DE-AC21-95MC31346--01

MFIX Documentation Numerical Technique

Topical Report
January 1998

By:
Madhava Syamlal

Work Performed Under Contract No.: DE-AC21-95MC31346

For
U.S. Department of Energy
Office of Fossil Energy
Federal Energy Technology Center
P.O. Box 880
Morgantown, West Virginia 26507-0880

By
EG&G Technical Services of West Virginia, Inc.
3610 Collins Ferry Road
Morgantown, West Virginia 26505-3276

Disclaimer

Contents

Executive Summary

MFIX (Multiphase Flow with Interphase eXchanges) is a general-purpose hydrodynamic model for describing chemical reactions and heat transfer in dense or dilute fluid-solids flows, which typically occur in energy conversion and chemical processing reactors. The calculations give time-dependent information on pressure, temperature, composition, and velocity distributions in the reactors. The theoretical basis of the calculations is described in the MFIX Theory Guide (Syamlal, Rogers, and O'Brien 1993). Installation of the code, setting up of a run, and post-processing of results are described in MFIX User's manual (Syamlal 1994).

Work was started in April 1996 to increase the execution speed and accuracy of the code, which has resulted in MFIX 2.0. To improve the speed of the code the old algorithm was replaced by a more implicit algorithm. In different test cases conducted the new version runs 3 to 30 times faster than the old version. To increase the accuracy of the computations, second order accurate discretization schemes were included in MFIX 2.0. Bubbling fluidized bed simulations conducted with a second order scheme show that the predicted bubble shape is rounded, unlike the (unphysical) pointed shape predicted by the first order upwind scheme. This report describes the numerical technique used in MFIX 2.0.

1 Introduction

MFIX is a general-purpose hydrodynamic model for describing chemical reactions and heat transfer in dense or dilute fluid-solids flows, which typically occur in energy conversion and chemical processing reactors. MFIX is written in FORTRAN and has the following modeling capabilities: multiple particle types, three-dimensional Cartesian or cylindrical coordinate systems, uniform or nonuniform grids, energy balances, and gas and solids species balances. MFIX calculations give time-dependent information on pressure, temperature, composition, and velocity distributions in the reactors. With such information, the engineer can visualize the conditions in the reactor, conduct parametric studies and what-if experiments, and, obtain information for the design of multiphase reactors.

The theoretical basis of MFIX is described in a companion report (Syamlal, Rogers, and O'Brien 1993). The current version of MFIX uses a slightly modified set of equations as summarized in Appendix A, however. The installation of the code, the setting up of a run, and post-processing of results are described in MFIX User's manual (Syamlal 1994). The keywords used in the input data file are given in a *readme* file included with the code. This report describes the numerical technique used in MFIX 2.0, which resulted from work started in April 1996 to increase the execution speed and accuracy of the code.

To speed up the code, its numerical technique was replaced with a semi-implicit scheme that uses automatic time-step adjustment. The essence of the method used in the old version of MFIX was developed by Harlow and Amsden (1975) and was implemented in the K-FIX code (Rivard and Torrey 1977). The method was later adapted for describing gas solids flows at the Illinois Institute of Technology (Gidaspow and Ettehadieh 1983). In MFIX 2.0 that method was replaced by a method based on SIMPLE (SemiImplicit Method for Pressure Linked Equations), which was developed by Patankar and Spalding (Patankar 1980). Several research groups have used extensions of SIMPLE (e.g., Spalding 1980, Fogt and Peric 1994, Laux and Johansen 1997), and this appears to be the method of choice in commercial CFD codes (Fluent manual 1996, Witt and Perry 1996). Two modifications of standard extensions of SIMPLE have been introduced in MFIX to improve the stability and speed of calculations. One, MFIX uses a solids volume fraction correction equation (instead of a solids pressure correction equation), which appears to help convergence when the solids are loosely packed. That equation also incorporates the effect of solids pressure, which is a novel feature of the MFIX implementation that helps to stabilize the calculations in densely packed regions. Two, MFIX uses automatic time-step adjustment to ensure that the run progresses with the highest execution speed. In various test cases conducted MFIX 2.0 was found to run 3-30 times faster than the old version of the code.

To improve the accuracy of the code, second-order accurate schemes for discretizing convection terms were added to MFIX. Reducing the discretization errors is harder when first-order upwind (FOU) method is used for discretizing convection terms. For example, FOU method leads to the prediction of pointed bubble shapes in simulations of bubbling fluidized beds. This unphysical shape, caused by numerical diffusion, could not be corrected with certain affordable grid refinement. With the same grid, however, the use of a second-order accurate discretization scheme gave the physically realistic rounded bubble shape (Syamlal 1997).

This report is organized as follows: The discretization methods for the convection-diffusion terms are described in Section 2. That information is used in Section 3 to derive the discrete analog of the scalar transport equation, which is a prototype of the multiphase flow partial differential equation. The next step of solving the set of discretized equations is outlined in Section 4. Sections 5-9 describe the equations used in the various steps of the solution algorithm. Section 10 describes the final steps in the solution algorithm: the under relaxation procedure used for stabilizing the calculations, the linear equation solvers, the calculation of residuals used for judging the convergence of iterations, and the method of automatic time-step adjustment.

2 Discretization of Convection-Diffusion Terms

2.1 First-Order Schemes

The transport equations contain convection-diffusion terms of the form

$$\rho u \frac{\partial \phi}{\partial x} - \frac{\partial}{\partial x}\left(\Gamma \frac{\partial \phi}{\partial x}\right) \tag{1}$$

The stability and accuracy of the numerical scheme critically depend upon the method used for discretizing such terms. It is straightforward to discretize the terms using a Taylor series expansion. In fluid dynamics computations, however, a control volume (CV) method is usually preferred. CV method invokes the physical basis of the derivation of conservation equations and ensures the global conservation of mass, momentum and energy even on coarse grids (Patankar 1980). At a sufficiently fine grid resolution the two methods would yield the same, accurate solution. The CV method is more attractive in practical computations, since a fine grid is seldom affordable.

When the convection-diffusion (advection) term is integrated over a CV (shaded region in Figure 2.1)

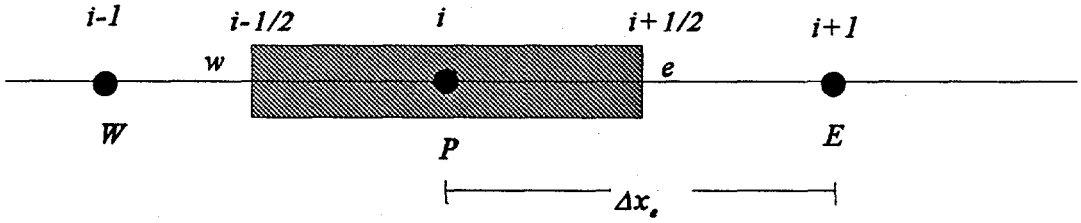


Figure 2.1 The control volume and node locations in x-direction

we get

$$\int \left[\rho u \frac{\partial \phi}{\partial x} - \frac{\partial}{\partial x}\left(\Gamma \frac{\partial \phi}{\partial x}\right)\right] dV = \left[\rho u \phi_e - \left(\Gamma \frac{\partial \phi}{\partial x}\right)_e\right] A_e - \left[\rho u \phi_w - \left(\Gamma \frac{\partial \phi}{\partial x}\right)_w\right] A_w \tag{2}$$

The calculation of diffusive fluxes at the CV faces is a relatively simple task: For example, the diffusive flux at the east-face can be approximated to a second order accuracy by

$$\left(\Gamma \frac{\partial \phi}{\partial x}\right)_e = \Gamma_e \frac{(\phi_E - \phi_P)}{\delta x_e} + O(\delta x^2) \tag{3}$$

The discretization of the convection terms is a more difficult task and the rest of this section will deal with that task.

From Equation 2.2 the discretization of the convection term is clearly equivalent to determining the value of ϕ at the CV faces (ϕ_e and ϕ_w). A simple interpolation such as

$$\phi_e = \frac{\phi_E + \phi_P}{2} \tag{4}$$

called central differencing, gives second order accuracy. However, in convection-dominated flows, typical of gas-solids flows, this method introduces spurious wiggles in the solution and leads to an unstable numerical scheme. A well-known remedy for stabilizing the calculations is the upwind discretization scheme

$$\phi_e = \begin{cases} \phi_P & u \geq 0 \\ \phi_E & u < 0 \end{cases} \tag{5}$$

This method is only first-order accurate and is diffusive.

The motivation for the upwind scheme can be found in the analytical solution for a steady, one-dimensional, source-free flow

$$\frac{\phi - \phi_P}{\phi_E - \phi_P} = \frac{\exp\left(\frac{P_e x}{\delta x_e}\right) - 1}{\exp(P_e) - 1} \tag{6}$$

where the cell Peclet number (P), the ratio of the convective flux to the diffusive flux, at the east face is given by

$$P_e = \frac{\rho u \Delta x_e}{\Gamma} \tag{7}$$

Low values of P show that diffusion is the dominant mechanism of transport at the scale of the grid size, and large values of P show that convection is dominant.

The analytical solution is plotted in Figure 2.2. At a large value of Peclet number (P_e =10) the cell face value of ϕ (at x = 0.5) is nearly identical to the upstream value of ϕ, and upwind differencing would be adequate to represent the face value of ϕ. At small Peclet numbers the upwind method is less accurate. An upwind bias, nevertheless, is evident in the solution, and it is a common feature of all practical discretization schemes for convection.

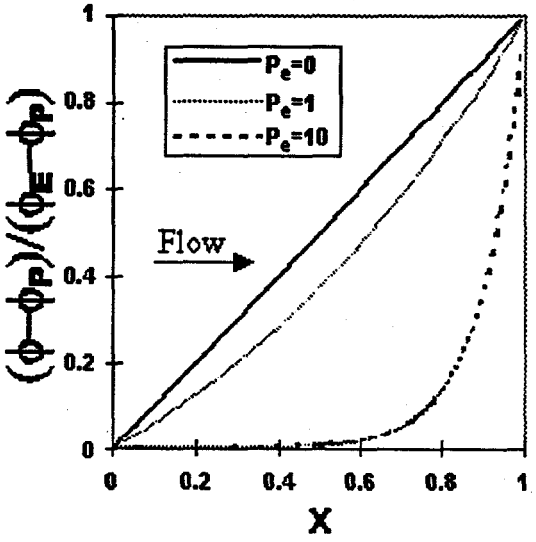


Figure 2.2. Analytical solution of a steady, 1-D, convection-diffusion equation

A more accurate discretization of the convection-diffusion flux, motivated by the analytical solution, is the exponential scheme

$$\rho u \phi_e - \left(\Gamma \frac{\partial \phi}{\partial x}\right)_e = \rho u \left(\phi_P + \frac{\phi_P - \phi_E}{\exp(P_e) - 1}\right) \tag{8}$$

The exponential scheme is computationally expensive, and, hence, cheaper approximations such as the hybrid scheme and the power-law scheme are used in practice. In these schemes upwind discretization is used for the convection term. The power-law discretization for the diffusive flux at the east face, for example, is given by

$$\left(\Gamma \frac{\partial \phi}{\partial x}\right)_e = A(|P_e|)\, \Gamma_e \frac{(\phi_E - \phi_P)}{\delta x_e} \tag{9}$$

where

$$A\ (|P|) = [(1-0.1|P|)^5] \tag{10}$$

which uses the definition of a double-brackets function

$$[R] = \begin{cases} 0 & R \le 0 \\ R & R > 0 \end{cases} \tag{11}$$

2.2 Higher-order schemes

For cell Peclet numbers larger than about 6, A(|P|) → 0, and the power-law (also, exponential) scheme is equivalent to first-order up winding for convection with physical diffusion switched off. The scheme is only first order accurate and does not give accurate results for flows in which the effects of transients, multi-dimensionality, or sources are important (Leonard and Mokhtari 1990). Higher order discretization methods for convection may be used to increase the accuracy. However, higher order schemes produce overshoots and undershoots near discontinuities. Such oscillations, apart from being undesirable in the final solution, will also hinder the convergence of iterations by producing physically unrealistic intermediate solutions (e.g., volume fractions greater than 1 or less than 0).

Resolving discontinuities in the solution has been a critical need in gas dynamics calculations and has motivated the development of higher order schemes that produce no spurious oscillations and total variation diminishing (TVD) schemes. Such schemes use a *limiter* that bounds the value of ϕ at the CV face, when the local variation in ϕ is monotonic. Thus, the discretization scheme is prevented from introducing any spurious extrema into the solution.

Leonard and Mokhtari describe a *universal limiter* expressed as a function of a normalized value of ϕ. Based on the notation for node locations along the flow direction given in Figure 2.3, the normalized value is given by

$$\tilde{\phi} = \frac{\phi - \phi_U}{\phi_D - \phi_U} \tag{12}$$

Then $\tilde{\phi}_U = 0$ and $\tilde{\phi}_D = 1$. The local distribution of ϕ is monotonic when

$$0 \le \tilde{\phi}_C \le 1 \tag{13}$$

Under monotonic conditions the *limiter* bounds ϕ_f in the following manner:

1. ϕ_f should be between ϕ_C and ϕ_D. That is

$$\tilde{\phi}_C \leq \tilde{\phi}_f \leq 1 \qquad \text{for} \quad 0 \leq \tilde{\phi}_C \leq 1 \tag{14}$$

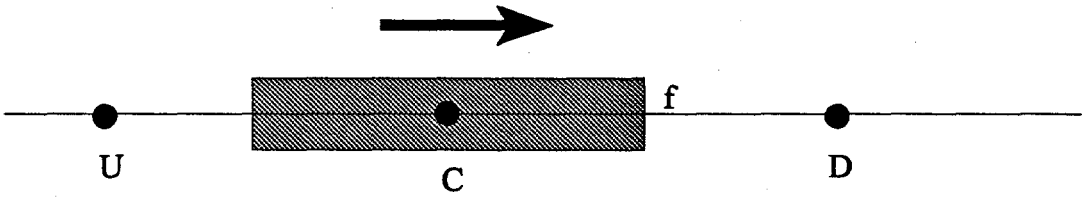


Figure 2.3. Notation for node locations based on the flow direction

This includes the special case $\phi_C = \phi_D$, in which case $\phi_f = \phi_C = \phi_D$. That is

$$\tilde{\phi}_f = 1 \qquad \text{for} \quad \tilde{\phi}_C = 1 \tag{15}$$

2. If $\phi_C = \phi_U$, we want $\phi_f = \phi_C = \phi_U$. That is

$$\tilde{\phi}_f = 0 \qquad \text{for} \quad \tilde{\phi}_C = 0 \tag{16}$$

3. To avoid nonuniqueness near $\tilde{\phi}_C \rightarrow 0$ define a steep boundary of a finite slope

$$\tilde{\phi}_f = \frac{\tilde{\phi}_C}{c} \qquad \text{for} \quad 0 \leq \tilde{\phi}_C \leq c \tag{17}$$

c is a constant about 0.01for steady state simulations. For time marching schemes c is the normal direction Courant number ($\frac{u\,\Delta t}{\Delta x}$).

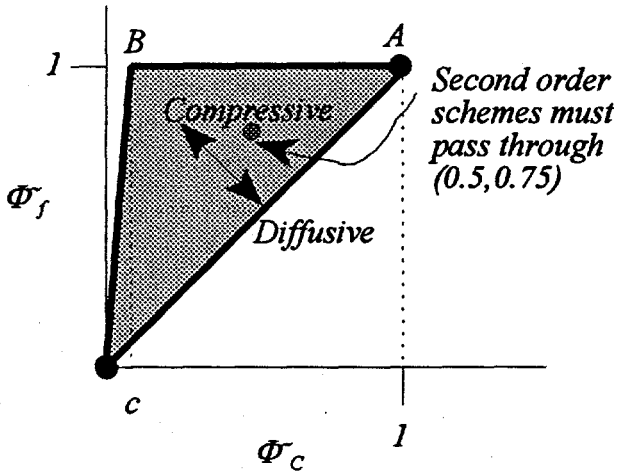


Figure 2.4. Normalized variable diagram

4. For non-monotonic behavior ($\tilde{\phi}_C < 0$ or $\tilde{\phi}_C > 1$), the limiter does not impose any constraint other than that the interpolations must be continuous with respect to $\tilde{\phi}_C$; that is, ϕ_f curve must pass through (0,0) and (1,1) with $\frac{\partial \tilde{\phi}_f}{\partial \tilde{\phi}_C} > 0$ and finite.

The above constraints may be represented on a normalized variable diagram (NVD) shown in Figure 2.4. The value of ϕ_f calculated by any higher order scheme should be constrained to pass through the shaded region to prevent overshoots and undershoots. Second order schemes must pass through the point (0.5, 0.75). Methods of order higher than two cannot be represented as a single curve on NVD.

Leonard and Mokhtari have proposed a *down wind factor* formulation, which simplifies the insertion of higher order methods into existing codes by not having to replace the septa-diagonal matrix structure of the discretized equations. The steps required for applying the formulation to an arbitrary order discretization method are the following:

1. Compute high-order, multidimensional, upwind biased estimate of ϕ_f.
2. Compute a tentative down wind weighting factor defined as

$$dwf^* = \frac{\phi_f - \phi_C}{\phi_D - \phi_C} = \frac{\tilde{\phi}_f - \tilde{\phi}_C}{1 - \tilde{\phi}_C} \qquad \textbf{(18)}$$

3. Limit dwf^* in the monotonic region to get dwf. The universal limiter expressed as a function of the down wind factor is shown in Figure 2.5.

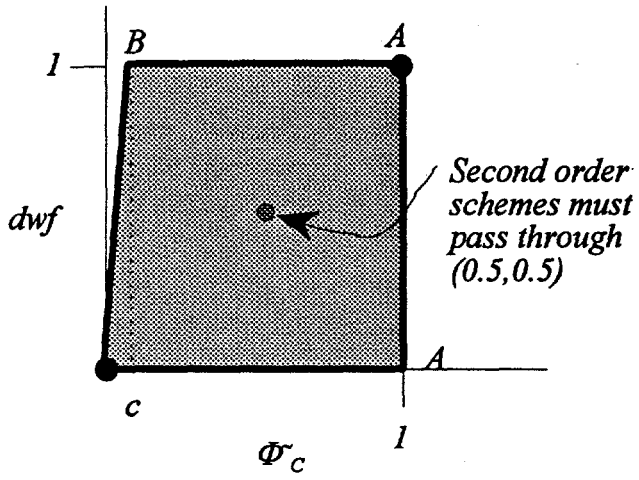


Figure 2.5. Down wind factor diagram

4. Compute the new estimate of ϕ_f as

$$\phi_f = dwf\ \phi_D + (1 - dwf)\ \phi_C \qquad (19)$$

Note that even for a higher-order method ϕ_f is calculated from the values of ϕ at adjacent nodes, the information from a wider stencil being contained in the factor *dwf*.

For several discretization schemes explicit formulas for the *down wind factors* may be derived. The formulas used in MFIX are shown in Table 2.I and are plotted in Figure 2.6. See Appendix C for some derivations.

Table I. Discretization formulas in terms of down wind factors

Discretization scheme	Down wind factor $\left(\theta \equiv \frac{\tilde{\phi}_C}{(1 - \tilde{\phi}_C)}\right)$
First order up winding	0
Central differencing	0.5
Second order up winding	½ θ
TVD schemes	0 if ($\tilde{\phi}_C < 0$ or $\tilde{\phi}_C > 1$) else

Discretization scheme	Down wind factor $\left(\theta \equiv \frac{\tilde{\phi}_C}{\left(1 - \tilde{\phi}_C\right)} \right)$
van Leer	$\tilde{\phi}_C$
Minmod	½ max[0, min(1, θ)]
MUSCL	½ max[0, min(2 θ, 0.5+0.5θ, 2)]
UMIST	½ max[0, min(2 θ, 0.75+0.25θ, 2)]
SMART	½ max[0, min(4 θ, 0.75+0.25θ, 2)]
Superbee	½ max[0, min(1, 2 θ), min(2, θ)]

Discretization scheme	Down wind factor $\left(\theta \equiv \frac{\tilde{\phi}_C}{\left(1 - \tilde{\phi}_C\right)} \right)$	
ULTRA-QUICK	$\frac{\theta}{2}$	$\tilde{\phi}_C < 0$
	$(\frac{1}{c} - 1)\,\theta$	$0 \le \tilde{\phi}_C < \frac{3c}{8-6c}$
	$0.375 + 0.125\,\theta$	$\frac{3c}{8-6c} \le \tilde{\phi}_C < \frac{5}{6}$
	1	$\frac{5}{6} \le \tilde{\phi}_C \le 1$
	0.5	$\tilde{\phi}_C > 1$

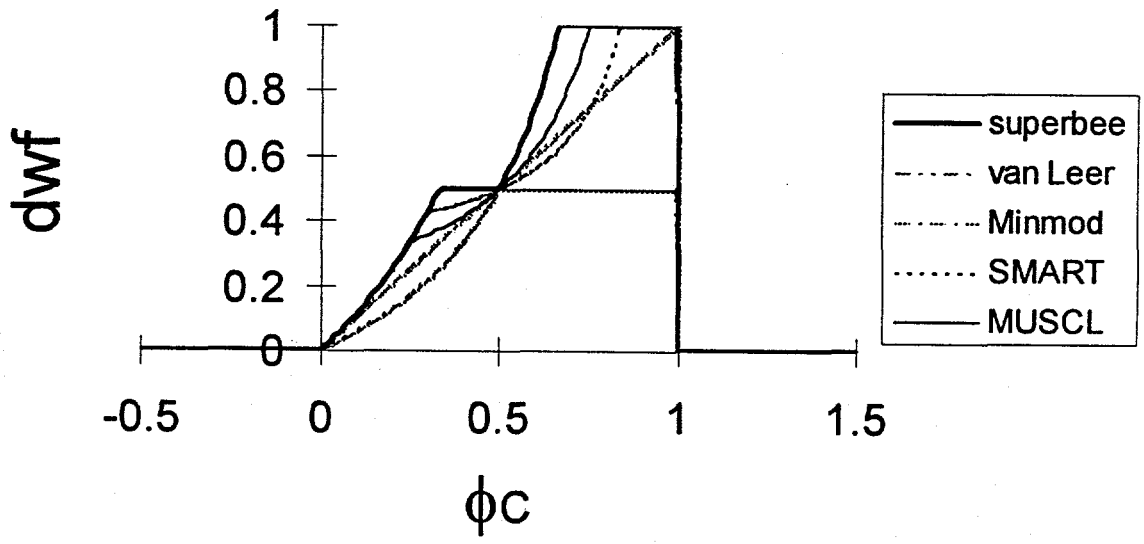


Figure 2.6. Downwind factors as a function of normalized ϕ

2.3 Usage of Downwind Factors

For the convenience of programming we calculate a convection weighting factor (ξ) from the down wind factors, which can be computed once and used without further checking the flow (wind) direction. The method is illustrated with the calculation of ξ at the east face. The node locations are shown in Figure 2.7. Also refer to Figure 2.3 for definitions of node locations C, D, and U.

1. Calculate $\tilde{\phi}_C$.

Algorithm 2.1

If $u_e \geq 0$ $(P \equiv C;\ E \equiv D;\ W \equiv U)$

$$\tilde{\phi}_C = \frac{\phi_P - \phi_W}{\phi_E - \phi_W} \tag{23}$$

else $(E \equiv C;\ P \equiv D;\ EE \equiv U)$

$$\tilde{\phi}_C = \frac{\phi_E - \phi_{EE}}{\phi_P - \phi_{EE}} \tag{24}$$

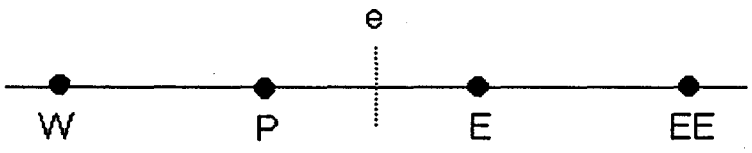


Figure 2.7. Node locations

2. Use $\tilde{\phi}_C$ in a formula from Table 2.I to calculate the down wind factor.

3. Recalling the definition

$$\phi_e = dwf_e\ \phi_D + \left(1 - dwf_e\right) \phi_C \tag{25}$$

calculate ξ as follows:

Algorithm 2.2

if $u_e \geq 0$ $\quad (P \equiv C;\ E \equiv D)$

$$\phi_f = dwf_e\ \phi_E + \left(1 - dwf_e\right) \phi_p \tag{27}$$

$$\xi_e = dwf_e$$

else $\quad (E \equiv C;\ P \equiv D)$

$$\phi_f = dwf_e\ \phi_P + \left(1 - dwf_e\right) \phi_E \tag{29}$$

$$\xi_e = 1 - dwf_e \tag{30}$$

The value of ϕ at the east face, for example, is then written as

$$\phi_e = \xi_e \phi_E + \overline{\xi}_e \phi_P \tag{31}$$

where $\overline{\xi}_e = 1 - \xi_e$.

In summary, Equation 2.31 is the discretization formula for the convection term and Equation 2.3 is the dicretization formula for the diffusion term. These formulas will be used in the next section to discretize a transport equation.

3 Scalar Transport Equation

In the previous section formulas for discretizing convection-diffusion terms were derived. In this section using those formulas we derive an algebraic (discretized) equation from a partial differential equation. For this demonstration we use the transport equation for a scalar ϕ:

$$\frac{\partial}{\partial t}\left(\epsilon_m \rho_m \phi\right) + \frac{\partial}{\partial x_i}\left(\epsilon_m \rho_m v_{mi} \phi\right) = \frac{\partial}{\partial x_i}\left(\Gamma_\phi \frac{\partial \phi}{\partial x_i}\right) + R_\phi \qquad (32)$$

The above equation has all the features of partial differential equations that form the multiphase flow model, except the interphase transfer term (Appendix A). The interphase transfer is an important aspect of the multiphase flow equations and deserves special attention in the algorithm. We postpone its discussion until Section 6.

3.1 Integration Over a Control Volume

We will integrate Equation 3.1 over a control volume (Figure 3.1) and write term by term, from left to right as follows:

- Transient term

$$\int \frac{\partial}{\partial t}(\epsilon_m \rho_m \phi) dV \approx \left[(\epsilon_m \rho_m \phi)_P - (\epsilon_m \rho_m \phi)_P^o\right]\frac{\Delta V}{\Delta t} \qquad (33)$$

where the superscript 'o' indicates old (previous) time step values.

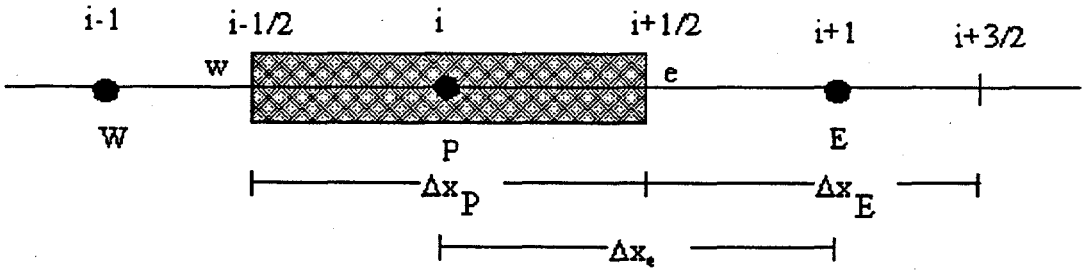


Figure 8 Control volume and node locations in x-direction

✿ Convection term

$$\int \frac{\partial}{\partial x_i}(\epsilon_m \rho_m v_{mi} \phi) dV \approx$$

$$\left\{\xi_e (\epsilon_m \rho_m \phi)_E + \bar{\xi}_e (\epsilon_m \rho_m \phi)_P\right\} (u_m)_e A_e$$

$$- \left\{\xi_w (\epsilon_m \rho_m \phi)_P + \bar{\xi}_w (\epsilon_m \rho_m \phi)_W\right\} (u_m)_w A_w$$

$$+ \left\{\xi_n (\epsilon_m \rho_m \phi)_N + \bar{\xi}_n (\epsilon_m \rho_m \phi)_P\right\} (v_m)_n A_n$$

$$- \left\{\xi_s (\epsilon_m \rho_m \phi)_P + \bar{\xi}_s (\epsilon_m \rho_m \phi)_S\right\} (v_m)_s A_s$$

$$+ \left\{\xi_t (\epsilon_m \rho_m \phi)_T + \bar{\xi}_t (\epsilon_m \rho_m \phi)_P\right\} (w_m)_t A_t$$

$$- \left\{\xi_b (\epsilon_m \rho_m \phi)_P + \bar{\xi}_b (\epsilon_m \rho_m \phi)_B\right\} (w_m)_b A_b \qquad \textbf{(34)}$$

where we have used Equation 2.31 from the previous section.

✿ Diffusion term

$$\int \frac{\partial}{\partial x_i}\left(\Gamma_\phi \frac{\partial \phi}{\partial x_i}\right) dV \approx \left(\Gamma_\phi \frac{\partial \phi}{\partial x}\right)_e A_e - \left(\Gamma_\phi \frac{\partial \phi}{\partial x}\right)_w A_w$$

$$+ \left(\Gamma_\phi \frac{\partial \phi}{\partial x}\right)_n A_n - \left(\Gamma_\phi \frac{\partial \phi}{\partial x}\right)_s A_s$$

$$+ \left(\Gamma_\phi \frac{\partial \phi}{\partial x}\right)_t A_t - \left(\Gamma_\phi \frac{\partial \phi}{\partial x}\right)_b A_b \qquad \textbf{(35)}$$

The diffusive fluxes are approximated using Equation 2.3 from previous section. For example, the diffusive flux through the east face is given by

$$\left(\Gamma_\phi \frac{\partial \phi}{\partial x}\right)_e \approx (\Gamma_\phi)_e \frac{\phi_E - \phi_P}{\Delta x_e} \qquad \textbf{(36)}$$

The cell face values of diffusion coefficients are calculated using a harmonic mean of the values defined at the cell centers (Patankar 1980). For example,

$$\left(\Gamma_\phi\right)_e = \left[\frac{1 - f_e}{\left(\Gamma_\phi\right)_P} + \frac{f_e}{\left(\Gamma_\phi\right)_E}\right]^{-1}$$

$$= \frac{\left(\Gamma_\phi\right)_P \left(\Gamma_\phi\right)_E}{f_e \left(\Gamma_\phi\right)_P + (1 - f_e) \left(\Gamma_\phi\right)_E} \tag{37}$$

where we use the definition

$$f_e = \frac{\Delta x_E}{\Delta x_P + \Delta x_E} \tag{38}$$

When the volume fraction of a certain phase changes to zero across a cell face, Equation 3.6 correctly sets the cell-face diffusion coefficient to zero. This is indeed the physically realistic limit as no diffusion can take place across such an interface. An arithmetic average, on the other hand, does not have such a physically realistic limit.

✿ Source term

Source terms are generally nonlinear and are first linearized as follows:

$$R_\phi \approx \bar{R}_\phi - R'_\phi \phi_P \tag{39}$$

For the stability of the iterations, it is essential that $R'_\phi \geq 0$. Also, when ϕ is a nonnegative field variable (such as, temperatures and mass fractions) it is recommended that $\bar{R}_\phi \geq 0$ (Patankar 1980). Then the integration of the source term over a control volume gives

$$\int R_\phi dV \approx \bar{R}_\phi \Delta V - R'_\phi \phi_P \Delta V \tag{40}$$

3.2 Discretized Transport Equation

Combining the equations derived above we get

$$\left(\frac{\left(\rho'_m\,\phi\right)_P - \left(\rho'_m\,\phi\right)^0}{\Delta t}\right)\Delta V$$

$$+\left\{\xi_e\left(\rho'_m\,\phi\right)_E + \bar{\xi}_e\left(\rho'_m\,\phi\right)_P\right\}(u_m)_e\,A_e - \left\{\xi_w\left(\rho'_m\,\phi\right)_P + \bar{\xi}_w\left(\rho'_m\,\phi\right)_W\right\}(u_m)_w\,A_w$$

$$+\left\{\xi_n\left(\rho'_m\,\phi\right)_N + \bar{\xi}_n\left(\rho'_m\,\phi\right)_P\right\}(v_m)_n\,A_n - \left\{\xi_s\left(\rho'_m\,\phi\right)_P + \bar{\xi}_s\left(\rho'_m\,\phi\right)_S\right\}(v_m)_s\,A_s$$

$$+\left\{\xi_t\left(\rho'_m\,\phi\right)_T + \bar{\xi}_t\left(\rho'_m\,\phi\right)_P\right\}(w_m)_t\,A_t - \left\{\xi_b\left(\rho'_m\,\phi\right)_P + \bar{\xi}_b\left(\rho'_m\,\phi\right)_B\right\}(w_m)_b\,A_b$$

$$= \left\{(\Gamma_\phi)_e\,\frac{\phi_E - \phi_P}{\Delta x_e}\,A_e - (\Gamma_\phi)_w\,\frac{\phi_p - \phi_w}{\Delta x_w}\,A_w\right\} + \left\{(\Gamma_\phi)_n\,\frac{\phi_N - \phi_P}{\Delta y_n}\,A_n - (\Gamma_\phi)_s\,\frac{\phi_p - \phi_s}{\Delta y_s}\,A_s\right\}$$

$$+\left\{(\Gamma_\phi)_t\,\frac{\phi_T - \phi_P}{\Delta z_t}\,A_t - (\Gamma_\phi)_b\,\frac{\phi_P - \phi_B}{\Delta z_b}\,A_b\right\} + \left(\bar{R}_\phi - \phi_P\,R'_\phi\right)\Delta V \qquad \textbf{(41)}$$

where we have defined the macroscopic densities as

$$\rho'_m = \epsilon_m\,\rho_m \qquad \textbf{(42)}$$

Equation 3.10 may be rearranged to get the following linear equation for ϕ

$$a_P\,\phi_P = \sum_{nb} a_{nb}\,\phi_{nb} + b \qquad \textbf{(43)}$$

where the subscript *nb* represents E, W, N, S, T, and B. Before using the above equation for determining ϕ, it is recommended that the discretized continuity equation multiplied by ϕ be subtracted from it.

The reason for the above manipulation is discussed in detail by Patankar (1980). The homogeneous part of the partial differential equation for ϕ has infinite number of solutions of the form $(\phi + c)$, where c is an arbitrary constant. The finite difference equation for ϕ must have the same number of solutions. Otherwise, small mass imbalances during the iterations may produce large fluctuations in the values of ϕ, and the convergence will be adversely affected. It is easy to show that the finite difference equations will have the desired property if

$$a_p = \sum_{nb} a_{nb} \qquad \textbf{(44)}$$

when the unsteady and source term contributions to a_p are discarded. Patankar calls this requirement *Rule 4*. An equation of the form 3.12 derived from Equation 3.10 will not satisfy *Rule 4*.

The discretized form of continuity equation can be easily obtained from Equation 3.10 by setting ϕ=1 and changing the source term to $\sum R_{ml}$. Then subtracting ϕ times the discretized continuity equation from Equation 3.10 we get a linear equation of the form 3.12, in which the coefficients are defined as follows:

$$a_E = D_e - \xi_e \left(\epsilon_m \rho_m\right)_E \left(u_m\right)_e A_e \tag{45}$$

$$a_W = D_w + \bar{\xi}_w \left(\epsilon_m \rho_m\right)_W \left(u_m\right)_w A_w \tag{46}$$

$$a_N = D_n - \xi_n \left(\epsilon_m \rho_m\right)_N \left(u_m\right)_n A_n \tag{47}$$

$$a_S = D_s + \bar{\xi}_s \left(\epsilon_m \rho_m\right)_S \left(u_m\right)_s A_s \tag{48}$$

$$a_T = D_t - \xi_t \left(\epsilon_m \rho_m\right)_T \left(w_m\right)_t A_t \tag{49}$$

$$a_B = D_b + \bar{\xi}_b \left(\epsilon_m \rho_m\right)_B \left(w_m\right)_b A_b \tag{50}$$

$$a_P = \sum_{nb} a_{nb} + a_P^0 + R'_\phi \Delta V + [\sum R_{ml}]\Delta V \tag{51}$$

$$b = a_P^0 \phi_P^0 + \bar{R}_\phi \Delta V + \phi_P [-\sum R_{ml}] \Delta V \tag{52}$$

$$a_p^0 = \frac{\left(\epsilon_m \rho_m\right)^0}{\Delta t} \Delta V \tag{53}$$

$$D_e = \frac{\left(\Gamma_\phi\right)_e A_e}{\Delta x_e} \tag{54}$$

Unlike single phase flow, multiphase continuity equations have a source term ($\sum R_{ml}$) that accounts for interphase mass transfer. Since ϕ times the continuity is equation is subtracted out, the term appears in discretized ϕ transport equation. Including this term in the source term would slow convergence, and including it in the center coefficient would destabilize the iterations when $\sum R_{ml} < 0$. Therefore, the term is manipulated as follows so that its contribution to a_p is nonnegative. Recall the definition of double brackets function:

$$[R] = \begin{cases} 0 & R \le 0 \\ R & R > 0 \end{cases} \tag{55}$$

From the above definition it follows that

$$R = [R] - [-R] \tag{56}$$

Then the interphase mass transfer term can be written as

$$\phi_P \sum_l R_{lm} = \phi_P [\![\sum_l R_{lm}]\!] - \phi_P [\![\sum_l -R_{lm}]\!] \tag{57}$$

The first term on the right-hand side of the above equation contributes to the source term (Equation 3.20) and the second coefficient contributes to the center coefficient (Equation 3.21).

If a power-law discretization is wanted, we will set the convection factor to zero (i.e., $\xi = 0$ and $\xi = 1$) and change D's to $D\,A(|P|)$. For example, replace D_e in Equation 3.14 by

$$D_e\, A\left(|P_e|\right) \tag{59}$$

where

$$A\left(|P_e|\right) = [(1 - 0{\cdot}1\ |P_e|)^5] \tag{60}$$

There are a couple of points to be noted regarding the usage of higher-order discretization schemes. In second or higher order discretization schemes the factors ξ are weak function of ϕ. Thus, the factors ξ in the discretized continuity equation may differ from the corresponding factors in the ϕ transport equation. Then the discretized equation for ϕ obtained by subtracting ϕ times the continuity equation will fail to satisfy *Rule 4*. Therefore, we make the assumption that the convection factors in the discretized continuity equation are the same as those in the ϕ transport equation to satisfy *Rule 4*.

The use of higher order methods may result in a violation of Patankar's *Rule 2* in some regions. *Rule 2* states that all the coefficients a_{nb} in Equation 3.11 must have the same sign, say positive. The physical basis for this rule is that an increase (or decrease) in the value of ϕ at a neighboring cell should cause an increase (or decrease) in the value of ϕ_P, not the other way around. This rule when combined with *Rule 4* also ensures that the discretization produces a diagonally dominant system of equations. The rule is strictly satisfied by the above equations only when first order upwinding is used. When higher order schemes are used, some coefficients may become negative when the local behavior of ϕ is monotonic. Such violations of *Rule 2* are of no concern, since the *limiter* uses more elaborate considerations to ensure that the solution is bounded and physically realistic (also see Appendix C).

4 An Outline of the Solution Algorithm

An extension of SIMPLE (Patankar 1980) is used for solving the discretized equations. Several issues need to be addressed when this algorithm, developed for single phase flow, is extended to solve multiphase flow equations. Spalding (1980) lists three issues, which he rates as "the first is obvious, the second rather less so, and the third may easily escape notice."

(i) There are more field variables, and hence more equations compared with single phase flow. This slows the computations, but does not in itself makes the algorithm any more complex.

(ii) Pressure appears in the three single phase momentum equations, but there is no convenient equation for solving the pressure field. The crux of SIMPLE algorithm is the derivation of such an equation for pressure -- the pressure correction equation. The pressure corrections give velocity corrections such that the continuity equation is satisfied exactly (to machine precision). There is no unique way to derive such an equation for multiphase flow, since there is more than one continuity equation in multiphase flow.

(iii) The multiphase momentum equations are strongly coupled through the momentum exchange term. Making this term fully implicit for the success of the numerical scheme is essential. This is the main idea in the Implicit Multifield Field (IMF) technique of Harlow and Amsden (1975), which is encoded in the K-FIX (Kachina- Fully Implicit Exchange) program of Rivard and Torrey (1977). In the MFIX algorithm the momentum equations are solved for the entire computational domain. To make the exchange term implicit all the equations for each velocity component (e.g., u-equations for gas and all solids phases) must be solved together, which leads to a nonstandard matrix structure. A cheaper alternative is to use the Partial Elimination Algorithm (PEA) of Spalding (1980), which is discussed in Section 6.

In granular multiphase flow two other issues critically determine the success of the numerical scheme. One is the handling of close-packed regions. The solids volume fraction ranges from zero to a maximum value of around 0.6 in close-packed regions. The lower limit is easily handled by formulating the linear equations such that nonnegative values of volume fraction are calculated. Constraining the solids volume fraction at or below the maximum value is more difficult. The formation of close-packed regions is analogous to the condensation of compressible vapor into an incompressible liquid. The reaction forces that resist further compaction of the granular medium result in a solids pressure, which must be distinguished from the fluid pressure. This situation was handled in the S^3 (Pritchett et al. 1978) and IIT (Gidaspow and Ettehadieh 1983) models by introducing a state equation that relates the solids pressure to the solids volume fraction (or the related void fraction). The solids pressure function increases exponentially as the solids volume fraction approaches the close-packed limit, and retards further compaction of the solids. This method allows the granular medium to be slightly compressible. The granular medium may also be considered incompressible as was done by Syamlal and O'Brien (1988). It is also the method used in FLUENT™ code (Fluent Users manual, 1996). In this method no state equation for solids pressure is needed. The solids volume fraction at maximum packing needs to be specified, which is also an implicit or explicit parameter in the state equation used for the slightly compressible case. In later versions of MFIX slight compressibility of packed granular medium was reintroduced to accommodate general frictional flow theories. The current numerical algorithm also requires that the granular medium be slightly compressible.

An equation similar to the fluid-pressure correction equation can be developed for the solids pressure. Such an equation is solved in FLUENT™ code (Fluent users manual, 1996). MFIX uses a solids volume fraction correction equation instead. The solids pressure correction equation requires that $\frac{\partial P_s}{\partial \epsilon_s}$ does not vanish when $\epsilon_s \to 0$. Solids volume fraction correction equation does not have such a restriction, but must account for the effect of solids pressure so that the computations are stabilized in close-packed regions.

A second issue is the difficulty in calculating field variables at interfaces across which a phase volume fraction goes to zero. The field variables associated with a phase are not defined in regions where the phase volume fraction is zero, and they may be set to arbitrary values. The computational algorithm must not use such arbitrarily set values, however. As an example, in the previous Section we showed that the use of a harmonic mean for calculating face values of diffusion coefficients will prevent the diffusion of ϕ into regions where the phase associated with it is absent. The calculation of velocity components at such interfaces is more difficult than scalar quantities because of the linearization of the nonlinear convection term. Across an interface where the phase volume fraction is nearly zero the normal component of velocity becomes very large. Since the product of the phase volume fraction and the velocity component is still nearly zero, the error in momentum conservation is negligible. However, the large phase velocities quickly destabilize the calculations, and a method is required to prevent such destabilization. MFIX uses an approximate calculation of the normal velocity at the interfaces (defined by a small threshold value for the phase volume fraction).

Gas-solids flows are inherently unstable. Steady state calculations are possible only for a few cases such as pneumatic (dilute) transport of solids. For vast majority of gas-solids flows, a transient simulation is conducted and the results are time-averaged. Transient simulations diverge, if too large a time-step is chosen. Too small a time step makes the computations very slow. Therefore, MFIX automatically adjusts the time steps, within user-specified limits, to reduce the computational time.

An out line of the computational technique is given below. The computational steps during a time step shown here are discussed in detail in the subsequent sections.

Algorithm 4.1

1. Start of the time step. Calculate physical properties, exchange coefficients, and reaction rates.

2. Calculate velocity fields based on the current pressure field: u_m^*, v_m^*, w_m^* (Sections 5 and 6).

3. Calculate fluid pressure correction P_g' (Section 7).

4. Update fluid pressure field applying an under relaxation: $P_g = P_g^* + \omega_{pg} P_g'$.

   Calculate velocity corrections from P_g' and update velocity fields: e.g., $u_m = u_m^* + u_m'$, m = 0 to M. (For solids phases, u_m calculated in this step is denoted as u_m' in Step 6).

5. Calculate the gradients $\frac{\partial P_m}{\partial x_m}$ for use in the solids volume fraction correction equation. Calculate solids volume fraction correction ϵ_m' (Section 8).

6. Update solids volume fractions ($\epsilon_m \rho_m$ in MFIX): $\epsilon_m = \epsilon_m^* + \omega_{Ps} \epsilon_m'$. Under relax only in regions where $\epsilon_0 < \epsilon_{cp}$ and $\epsilon_m' > 0$; i.e. where the solids are densely packed and the solids volume fraction is increasing.

   Calculate velocity corrections for the solids phases and update solids velocity fields: e.g., $u_m = u_m^* + u_m'$ (m = 1 to M).

7. Calculate the void fraction: $\epsilon_0 = \epsilon_v - \sum_{m \neq 0} \epsilon_m$. ($\epsilon_v$ is usually equal to 1).

8. Calculate the solids pressure from the state equation $P_m = P_m(\epsilon_m)$.

9. Calculate temperatures and species mass fractions (Section 9).

10. Use the normalized residuals calculated in Steps 2, 3, 5, and 9 to check for convergence. If the convergence criterion is not satisfied continue iterations (Step 2), else go to next time-step (Step 1).

5 Momentum Equation

The discretization of the momentum equations is similar to that of the scalar transport equation, except that the control volumes are staggered. As explained by Patankar (1980), if the velocity components and pressure are stored at the same grid locations a checkerboard pressure field can develop as an acceptable solution. A staggered grid is used for preventing such unphysical pressure fields. As shown in Figure 5.1, in relation to the scalar control volume centered around the filled circles, the x-momentum control volume is shifted east by half a cell. Similarly the y-momentum control volume is shifted north by half a cell, and the z-momentum control volume is shifted top by half a cell.

5.1 Discretized Momentum Equation

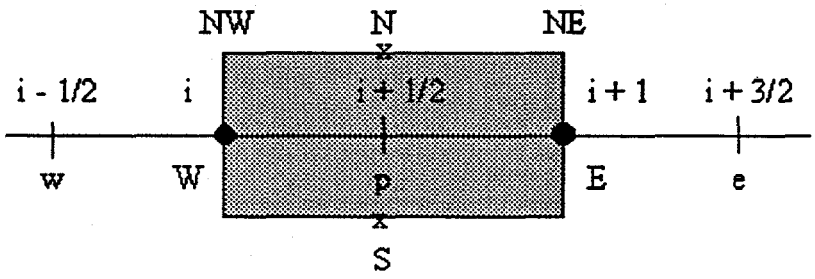


Figure 5.1. X-momentum equation control volume

For calculating the momentum convection, velocity components are required at the locations E, W, N, and S. They are calculated from an arithmetic average of the values at neighboring locations:

$$(u_m)_E = f_E (u_m)_p + (1 - f_E)(u_m)_e \tag{67}$$

$$(v_m)_N = f_p (v_m)_{NW} + (1 - f_p)(v_m)_{NE} \tag{68}$$

A volume fraction value required at the cell center denoted by p is similarly calculated.

$$(\epsilon_m)_p = f_p (\epsilon_m)_W + (1 - f_p)(\epsilon_m)_E \tag{69}$$

where

$$f_E = \frac{\Delta x_e}{\Delta x_p + \Delta x_e} \tag{70}$$

and

$$f_p = \frac{\Delta x_E}{\Delta x_W + \Delta x_E} \tag{71}$$

Now the discretized x-momentum equation can be written as

$$a_p(u_m)_p = \sum_{nb} a_{nb}(u_m)_{nb} + b_p$$

$$-A_p \left(\epsilon_m\right)_p \left((P_g)_E - (P_g)_W\right) + \left(\sum_l F_{lm} \left(u_l - u_m\right)_p\right) \Delta V_e \qquad (72)$$

The above equation is similar to the discretized scalar transport equation described in Section 3, except for the last two terms: The pressure gradient term is determined based on the current value of P_g (Step 2, Section 4) and is added to the source term of the linear equation set. The interface transfer term couples all the equations for the same component. A procedure for decoupling the equations is described in Section 6.

The definitions for the rest of the terms in Equation (5.6) are as follows:

$$a_e = D_E - \xi_E \left(\epsilon_m \rho_m\right)_e \left(u_m\right)_E A_E$$

$$a_w = D_W + \bar{\xi}_W \left(\epsilon_m \rho_m\right)_w \left(u_m\right)_W A_W$$

$$a_n = D_N - \xi_N \left(\epsilon_m \rho_m\right)_n \left(v_m\right)_N A_N$$

$$a_s = D_S + \bar{\xi}_S \left(\epsilon_m \rho_m\right)_s \left(v_m\right)_S A_S$$

$$a_t = D_T - \xi_T \left(\epsilon_m \rho_m\right)_t \left(w_m\right)_T A_T$$

$$a_b = D_B + \bar{\xi}_B \left(\epsilon_m \rho_m\right)_b \left(w_m\right)_B A_B$$

$$a_p = \sum_{nb} a_{nb} + a_p^0 + R'_{u_m} \Delta V_e + [\sum R_{lm}] \Delta V_e + S'$$

$$b = a_p^0 u_m^0 + \bar{R}_{u_m} \Delta V_e + u_m [-\sum R_{lm}] \Delta V_e + \left(\epsilon_m \rho_m\right)_e g_x \Delta V_e + \bar{S}$$

$$a_p^0 = \frac{\left(\epsilon_m \rho_m\right)^0 \Delta V_e}{\Delta t}$$

$$D_E = \frac{(\mu_m)_E A_E}{\Delta x_E} \qquad (73)$$

The center coefficient a_p and the source term b contain the extra terms S' and S, which account for the sources arising from cylindrical coordinates, porous media model, and shear stress terms. These are described in the next two subsections.

5.2 Cylindrical Coordinates

The MFIX cylindrical coordinate system is shown in Figure 5.2. The three momentum equations in MFIX notation are as follows:

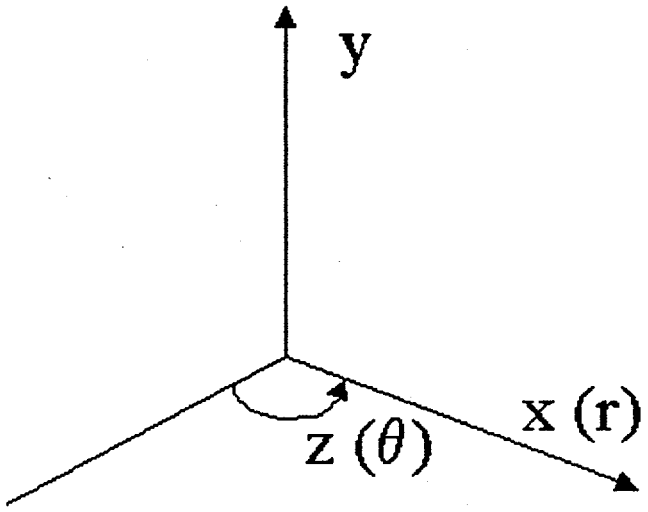


Figure 5.2 Coordinate labels in MFIX

- x-momentum equation:

$$\frac{\partial}{\partial t}\left(\rho'_m\, u_m\right) + \frac{1}{x}\frac{\partial}{\partial x}\left(\rho'_m\, x\, u_m^2\right) + \frac{\partial}{\partial y}\left(\rho'_m\, u_m\, v_m\right) + \frac{1}{x}\frac{\partial}{\partial z}\left(\rho'_m\, u_m\, w_m\right)$$

$$= -\epsilon_m \frac{\partial P_g}{\partial x} + \frac{\rho'_m\, w_m^2}{x} - \frac{\partial P_m}{\partial x} + \frac{\partial}{\partial x}\left[\lambda_m\, tr(D_m)\right] + \frac{1}{x}\frac{\partial}{\partial x}\left(x\,\tau_{xx}\right) + \frac{1}{x}\frac{\partial}{\partial z}\,\tau_{xz} \qquad (74)$$

$$-\frac{\tau_{zz}}{x} + \frac{\partial \tau_{xy}}{\partial y} + \rho'_m\, g_x + \sum_{\ell} F_{m\ell}\left(u_\ell - u_m\right)$$

- y-momentum equation:

$$\frac{\partial}{\partial t}\left(\rho'_m\, v_m\right) + \frac{1}{x}\frac{\partial}{\partial x}\left(\rho'_m\, x\, u_m\, v_m\right) + \frac{1}{x}\frac{\partial}{\partial z}\left(\rho'_m\, v_m\, w_m\right) + \frac{\partial}{\partial y}\left(p_m\, v_m^2\right)$$

$$= -\epsilon_m \frac{\partial P_g}{\partial y} - \frac{\partial P_m}{\partial y} + \frac{\partial}{\partial y}\left[\lambda_m\, tr\left(D_m\right)\right] + \frac{1}{x}\frac{\partial}{\partial x}\left(x\,\tau_{xy}\right) + \frac{1}{x}\frac{\partial \tau_{zy}}{\partial z} + \frac{\partial}{\partial y}\tau_{yy} \qquad (75)$$

$$+ \rho'_m\, g_y + \sum_{\ell} F_{m\ell}\left(v_\ell - v_m\right)$$

- z-momentum equation:

$$\frac{\partial}{\partial t}\left(\rho'_m\, w_m\right) + \frac{1}{x}\frac{\partial}{\partial x}\left(\rho'_m\, x\, u_m\, w_m\right) + \frac{1}{x}\frac{\partial}{\partial z}\left(\rho'_m\, w_m^2\right) + \frac{\partial}{\partial y}\left(\rho'_m\, v_m\, w_m\right)$$

$$= -\frac{\epsilon_m}{x}\frac{\partial P_g}{\partial z} - \frac{\partial P_m}{x\partial z} + \frac{\partial}{x\partial z}\left[\lambda_m\, tr\left(D_m\right)\right] - \frac{\rho'_m\, u_m\, w_m}{x} + \frac{1}{x}\frac{\partial}{\partial x}\left(x\,\tau_{xz}\right) + \frac{1}{x}\frac{\partial}{\partial z}\tau_{zz} \qquad (76)$$

$$+ \frac{\tau_{xz}}{x} + \frac{\partial}{\partial y}\tau_{zy} + \rho'_m\, g_z + \sum_{\ell} F_{m\ell}\left(w_\ell - w_m\right)$$

The equations in Cartesian coordinates are obtained from the above equations, by setting the value of x to 1 and terms specific to cylindrical coordinates to zero. Also, for the fluid phase P_m is equal to zero.

The stress tensor $\overline{\overline{\tau}}$ is defined as

$$\overline{\overline{\tau}} = 2\mu_m \overline{\overline{D}}_m \qquad (77)$$

The rate of strain tensor is

$$\overline{\overline{D}}_m = \begin{bmatrix} \frac{\partial u_m}{\partial x} & \frac{1}{2}\left[\frac{\partial v_m}{\partial x} + \frac{\partial u_m}{\partial y}\right] & \frac{1}{2}\left[\frac{\partial w_m}{\partial x} + \frac{1}{x}\frac{\partial u_m}{\partial z} - \frac{w_m}{x}\right] \\ \frac{1}{2}\left[\frac{\partial v_m}{\partial x} + \frac{\partial u_m}{\partial y}\right] & \frac{\partial v_m}{\partial y} & \frac{1}{2}\left[\frac{1}{x}\frac{\partial v_m}{\partial z} + \frac{\partial w_m}{\partial y}\right] \\ \frac{1}{2}\left[\frac{\partial w_m}{\partial x} + \frac{1}{x}\frac{\partial u_m}{\partial z} - \frac{w_m}{x}\right] & \frac{1}{2}\left[\frac{1}{x}\frac{\partial v_m}{\partial z} + \frac{\partial w_m}{\partial y}\right] & \left[\frac{1}{x}\frac{\partial w_m}{\partial z} + \frac{u_m}{x}\right] \end{bmatrix} \qquad (78)$$

The stress terms on the right-hand side of the x-momentum equation are as follows:

$$\text{R.H.S. of x-momentum eq.} = \ldots + \frac{1}{x}\frac{\partial}{\partial x}\left[x\, 2\mu_m \frac{\partial u_m}{\partial x}\right] + \frac{\partial}{\partial y}\left[\mu_m\left(\frac{\partial v_m}{\partial x} + \frac{\partial u_m}{\partial y}\right)\right]$$
$$+ \frac{1}{x}\frac{\partial}{\partial z}\left[\mu_m\left(\frac{\partial w_m}{\partial x} + \frac{1}{x}\frac{\partial u_m}{\partial z} - \frac{w_m}{x}\right)\right] \qquad (79)$$
$$- \frac{2\mu_m}{x}\left[\frac{1}{x}\frac{\partial w_m}{\partial z} + \frac{u_m}{x}\right]$$

which can be rearranged as

$$\text{R.H.S. of x-momentum eq.} = \ldots + \frac{1}{x}\frac{\partial}{\partial x}\left[x\,\mu_m \frac{\partial u_m}{\partial x}\right] + \frac{\partial}{\partial y}\left[\mu_m \frac{\partial u_m}{\partial y}\right] + \frac{\partial}{x\,\partial z}\left[\mu_m \frac{\partial u_m}{x\,\partial z}\right]$$
$$+ \frac{1}{x}\frac{\partial}{\partial x}\left[x\,\mu_m \frac{\partial u_m}{\partial x}\right] + \frac{\partial}{\partial y}\left[\mu_m \frac{\partial v_m}{\partial x}\right] \qquad (80)$$
$$+ \frac{1}{x}\frac{\partial}{\partial z}\left[\mu_m\left(\frac{\partial w_m}{\partial x} - \frac{w_m}{x}\right)\right] - \frac{2\mu_m}{x}\left[\frac{1}{x}\frac{\partial w_m}{\partial z} + \frac{u_m}{x}\right]$$

The first three terms appear in Equation (5.7) as the diffusion terms. The other terms are added as additional source terms S' and S.

$$\text{x-momentum sources} = \frac{\rho'_m\, w_m^2}{x} + \frac{\partial}{\partial x}\left[\lambda_m\, tr(D_m)\right] + \frac{1}{x}\frac{\partial}{\partial x}\left[x\ \mu_m \frac{\partial u_m}{\partial x}\right]$$
$$+ \frac{\partial}{\partial y}\left[\mu_m \frac{\partial v_m}{\partial x}\right] + \frac{1}{x}\frac{\partial}{\partial z}\left[\mu_m\left(\frac{\partial w_m}{\partial x} - \frac{w_m}{x}\right)\right] \qquad (81)$$
$$- \frac{2\mu_m}{x^2}\frac{\partial w_m}{\partial z} - \frac{2\mu_m\, u_m}{x^2}$$

Similarly the additional source terms for the y- and w-momentum equations can be determined and are shown below:

$$\text{y-momentum sources} = \frac{\partial}{\partial x}\left[\lambda_m\, tr\,(D_m)\right] + \frac{1}{x}\frac{\partial}{\partial x}\left[x\,\mu_m \frac{\partial u_m}{\partial y}\right] + \frac{\partial}{\partial y}\left[\mu_m \frac{\partial v_m}{\partial y}\right]$$
$$+ \frac{1}{x}\frac{\partial}{\partial z}\left[\mu_m \frac{\partial w_m}{\partial y}\right] \qquad (82)$$

$$\text{z-momentum sources} = -\frac{\rho'_m u_m w_m}{x} + \frac{\partial}{x\partial z}\left[\lambda_m \, tr\,(D_m)\right]$$
$$+ \frac{1}{x}\frac{\partial}{\partial x}\left[x \;\mu_m\left(\frac{1}{x}\frac{\partial u_m}{\partial z} - \frac{w_m}{x}\right)\right] + \frac{\partial}{\partial y}\left[\mu_m \frac{\partial v_m}{x\partial z}\right] \tag{83}$$
$$+ \frac{1}{x}\frac{\partial}{\partial z}\left[\mu_m\left(\frac{\partial w_m}{x\partial z} + \frac{2\,u_m}{x}\right)\right] + \frac{\mu_s}{x}\left[\frac{\partial w_m}{\partial x} + \frac{1}{x}\frac{\partial u_m}{\partial z} - \frac{w_m}{x}\right]$$

5.3 Discretization Formulas

The discretization formulas for the additional source terms are given below. Refer to the control volume dimensions in Appendix B.

- x-Momentum Equation:

$$\int \frac{\rho'_m w_m^2}{x}\, dV = \frac{(\rho'_m)_e (w_m)_e^2}{x_{i+1/2}}\,\Delta V_e \tag{84}$$

$$\int \frac{\partial}{\partial x}\left[\lambda_m \, tr\,(D_m)\right] dV = \left[\left(\lambda_m \, tr\,(D_m)\right)_E - \left(\lambda_m \, tr\,(D_m)\right)_W\right] A_p \tag{85}$$

$$\int \frac{1}{x}\frac{\partial}{\partial x}\left[x \;\mu_m \frac{\partial u_m}{\partial x}\right] dV = \left[\mu_m \frac{\partial u_m}{\partial x}\right]_E A_E - \left[\mu_m \frac{\partial u_m}{\partial x}\right]_W A_W$$
$$= [\mu_m]_{i+1} \frac{(u_m)_{i+3/2} - (u_m)_{i+1/2}}{\Delta x_{i+1}} A_{i+1} - [\mu_m]_i \frac{(u_m)_{i+1/2} - (u_m)_{i-1/2}}{\Delta x_i} A_i \tag{86}$$

$$\int \frac{\partial}{\partial y}\left[\mu_m \frac{\partial v_m}{\partial x}\right] dV = \left[\mu_m \frac{\partial v_m}{\partial x}\right]_{ne} A_{ne} - \left[\mu_m \frac{\partial v_m}{\partial x}\right]_{se} A_{se}$$
$$= [\mu_m]_{i+1/2,\, j+1/2,\, k} \frac{(v_m)_{i+1,\, j+1/2,\, k} - (v_m)_{i,\, j+1/2,\, k}}{\Delta x_{i+1/2}} A_{i+1/2,\, j+1/2,\, k} \tag{87}$$
$$- [\mu_m]_{i+1/2,\, j-1/2,\, k} \frac{(v_m)_{i+1,\, j-1/2} - (v_m)_{i,\, j-1/2,\, k}}{\Delta x_{i+1/2}} A_{i+1/2,\, j-1/2,\, k}$$

$$\int \frac{1}{x}\frac{\partial}{\partial z}\left[\mu_m\left(\frac{\partial w_m}{\partial x} - \frac{w_m}{x}\right)\right]dV = \left[\mu_m\left(\frac{\partial w_m}{\partial x} - \frac{w_m}{x}\right)\right]_{te} A_{te} - \left[\mu_m\left(\frac{\partial w_m}{\partial x} - \frac{w_m}{x}\right)\right]_{be} A_{be}$$

$$= \left[\mu_m\right]_{i+1/2,\, j,\, k+1/2}\left[\frac{\left(w_m\right)_{i+1,\, j,\, k+1/2} - \left(w_m\right)_{i,\, j,\, k+1/2}}{\Delta x_{i+1/2}} - \frac{\left(w_m\right)_{i+1,\, j,\, x+1/2} + \left(w_m\right)_{i,\, j,\, k+1/2}}{2\, x_{i+1/2}}\right] A_{i+1/2,\, j,\, k+1/2} \quad (88)$$

$$- \left[\mu_m\right]_{i+1/2,\, j,\, k-1/2}\left[\frac{\left(w_m\right)_{i+1,\, j,\, k-1/2} - \left(w_m\right)_{i,\, j,\, k-1/2}}{\Delta x_{i+1/2}} - \frac{\left(w_m\right)_{i+1,\, j,\, k-1/2} + \left(w_m\right)_{i,\, j,\, k-1/2}}{2\, x_{i+1/2}}\right] A_{i+1/2,\, j,\, k-1/2}$$

$$\int -\frac{2\,\mu_s}{x}\left[\frac{1}{x}\frac{\partial w_m}{\partial z} + \frac{u_m}{x}\right]dV = -\frac{2\left[\mu_s\right]_e}{x_{i+1/2}}\left[\frac{1}{x_{i+1/2}}\left[\frac{\partial w_m}{\partial z}\right]_e + \frac{u_m}{x_{i+1/2}}\right]\Delta V_e \quad (89)$$

where

$$\left[\frac{\partial w_m}{x\,\partial z}\right]_{i+1/2} = 0.5\left\{\frac{\left(w_m\right)_{i,\, j,\, k+1/2} - \left(w_m\right)_{i,\, j,\, k-1/2}}{x_i\,\Delta z_k} + \frac{\left(w_m\right)_{i+1,\, j,\, k+1/2} - \left(w_m\right)_{i+1,\, j,\, k-1/2}}{x_{i+1}\,\Delta z_k}\right\} \quad (90)$$

- y-Momentum:

$$\int \frac{\partial}{\partial y}\left[\lambda_m\, tr\left(D_m\right)\right]dV = \left[\lambda_m\, tr\left(D_m\right)\right]_N - \left[\lambda_m\, tr\left(D_m\right)\right]_S A_p \quad (91)$$

$$\int \frac{1}{x}\frac{\partial}{\partial x}\left[x\ \ \mu_m\frac{\partial u_m}{\partial y}\right]dV = \left[\mu_m\frac{\partial u_m}{\partial y}\right]_{ne} A_{ne} - \left[\mu_m\frac{\partial u_m}{\partial y}\right]_{nw} A_{nw}$$

$$= \left[\mu_m\right]_{i+1/2,\, j+1/2,\, k}\frac{\left(u_m\right)_{i+1/2,\, j+1,\, k} - \left(u_m\right)_{i+1/2,\, j,\, k}}{\Delta y_{j+1/2}} A_{i+1/2,\, j+1/2,\, k} \quad (92)$$

$$- \left[\mu_m\right]_{i-1/2,\, j+1/2,\, k}\frac{\left(u_m\right)_{i-1/2,\, j+1,\, k} - \left(u_m\right)_{i-1/2,\, j,\, k}}{\Delta y_{j+1/2}} A_{i-1/2,\, j+1/2,\, k}$$

$$\int \frac{\partial}{\partial y}\left[\mu_m \frac{\partial v_m}{\partial y}\right] dV = \left[\mu_m \frac{\partial v_m}{\partial y}\right]_N A_N - \left[\mu_m \frac{\partial v_m}{\partial y}\right]_S A_S$$

$$= \left[\mu_m\right]_{i,\,j+1,\,k} \frac{\left(v_m\right)_{i,\,j+3/2,\,k} - \left(v_m\right)_{i,\,j+1/2,\,k}}{\Delta y_{j+1}} A_{i,\,j+1,\,k} - \left[\mu_m\right]_{i,\,j,\,k} \frac{\left(v_m\right)_{i,\,j+1/2,\,k} - \left(v_m\right)_{i,\,j-1/2,\,k}}{\Delta y_j} A_{i,\,j,\,k} \tag{93}$$

$$\int \frac{1}{x} \frac{\partial}{\partial z}\left[\mu_m \frac{\partial w_m}{\partial y}\right] dV = \left[\mu_m \frac{\partial w_m}{\partial y}\right]_{nt} A_{nt} - \left[\mu_m \frac{\partial w_m}{\partial y}\right]_{nb} A_{nb}$$

$$= \left[\mu_m\right]_{i,\,j+1/2,\,k} \frac{\left(w_m\right)_{i,\,j+1,\,k} - \left(w_m\right)_{i,\,j,\,k}}{\Delta y_{j+1/2}} A_{i,\,j+1/2,\,k} \tag{94}$$

$$- \left[\mu_m\right]_{i,\,j+1/2,\,k-1} \frac{\left(w_m\right)_{i,\,j+1,\,k-1} - \left(w_m\right)_{i,\,j,\,k-1}}{\Delta y_{j+1/2}} A_{i,\,j+1/2,\,k-1}$$

- z-Momentum:

$$\int - \frac{\rho'_m\, u_m\, w_m}{x} dV = - \frac{\left(\rho'_m\right)_t \left(u_m\right)_t w_m}{x_i} \Delta V_t \tag{95}$$

$$\int \frac{\partial}{x\,\partial z}\left[\lambda_m\, tr\left(D_m\right)\right] dV = \left\{\left[\lambda_m\, tr\left(D_m\right)\right]_T - \left[\lambda_m\, tr\left(D_m\right)\right]_B\right\} A_p \tag{96}$$

$$\int \frac{1}{x} \frac{\partial}{\partial x}\left[x\ \mu_m\left(\frac{1}{x}\frac{\partial u_m}{\partial z} - \frac{w_m}{x}\right)\right] dV$$

$$= \left[\mu_m\left(\frac{1}{x}\frac{\partial u_m}{\partial z} - \frac{w_m}{x}\right)\right]_{te} A_{te} - \left[\mu_m\left(\frac{1}{x}\frac{\partial u_m}{\partial z} - \frac{w_m}{x}\right)\right]_{tw} A_{tw}$$

$$= \left[\mu_m\right]_{i+1/2,\,j,\,k+1/2} \left(\frac{\left(u_m\right)_{i+1/2,\,j,\,k+1} - \left(u_m\right)_{i+1/2,\,j,\,k}}{x_{i+1/2}\,\Delta z_{k+1/2}} - \frac{\left(w_m\right)_{i,\,j,\,k+1/2} + \left(w_m\right)_{i+1,\,j,\,k+1/2}}{2\,x_{i+1/2}}\right) A_{i+1/2,\,j,\,k+1/2} \tag{97}$$

$$- \left[\mu_m\right]_{i-1/2,\,j,\,k+1/2} \left(\frac{\left(u_m\right)_{i-1/2,\,j,\,k+1} - \left(u_m\right)_{i-1/2,\,j,\,k}}{x_{i-1/2}\,\Delta z_{k+1/2}} - \frac{\left(w_m\right)_{i,\,j,\,k+1/2} + \left(w_m\right)_{i-1,\,j,\,k+1/2}}{2\,x_{i-1/2}}\right) A_{i-1/2,\,j,\,k+1/2}$$

$$\int \frac{\partial}{\partial y}\left[\mu_m \frac{\partial v_m}{x\,\partial z}\right] dV = \left[\mu_m \frac{\partial v_m}{x\,\partial z}\right]_{tn} A_{tn} - \left[\mu_m \frac{\partial v_m}{x\,\partial z}\right]_{ts} A_{ts}$$

$$= \left[\mu_m\right]_{i,\,j+1/2,\,k+1/2} \left(\frac{\left(v_m\right)_{i,\,j+1/2,\,k+1} - \left(v_m\right)_{i,\,j+1/2,\,k}}{x_i\ \Delta z_{k+1/2}}\right) A_{i,\,j+1/2,\,k+1/2} \qquad (98)$$

$$- \left[\mu_m\right]_{i,\,j-1/2,\,k+1/2} \left(\frac{\left(v_m\right)_{i,\,j-1/2,\,k+1} - \left(v_m\right)_{i,\,j-1/2,\,k}}{x_i\ \Delta z_{k+1/2}}\right) A_{i,\,j-1/2,\,k+1/2}$$

$$\int \frac{1}{x}\frac{\partial}{\partial z}\left[\mu_m\left(\frac{\partial w_m}{x\,\partial z} + \frac{2\,u_m}{x}\right)\right] dV$$

$$= \left[\mu_m\left(\frac{\partial w_m}{x\,\partial z} + \frac{2\,u_m}{x}\right)\right]_T A_T - \left[\mu_m\left(\frac{\partial w_m}{x\,\partial z} + \frac{2\,u_m}{x}\right)\right]_B A_B$$

$$= \left[\mu_m\right]_{i,\,j,\,k+1} \left(\frac{\left(w_m\right)_t - \left(w_m\right)_p}{x_i\ \Delta z_{k+1}} + \frac{2\left[\left(u_m\right)_{i+1/2,\,j,\,k+1} + \left(u_m\right)_{i-1/2,\,j,\,k+1}\right]}{2\,x_i}\right) A_{i,\,j,\,k+1} \qquad (99)$$

$$- \left[\mu_m\right]_{i,\,j,\,k} \left(\frac{\left(w_m\right)_p - \left(w_m\right)_b}{x_i\ \Delta z_k} + \frac{2\left[\left(u_m\right)_{i+1/2,\,j,\,k} + \left(u_m\right)_{i-1/2,\,j,\,k}\right]}{2\,x_i}\right) A_{i,\,j,\,k}$$

$$\int \frac{\mu_s}{x}\left[\frac{\partial w_m}{\partial x} + \frac{1}{x}\frac{\partial u_m}{\partial z} - \frac{w_m}{x}\right] dV = \frac{\left(\mu_s\right)_p}{x_i}\left[\left(\frac{\partial w_m}{\partial x}\right)_p + \frac{1}{x_i}\left(\frac{\partial u_m}{\partial z}\right)_p - \frac{w_m}{x_i}\right] \Delta V_p \qquad (100)$$

$$\left(\frac{\partial w_m}{\partial x}\right)_p = \left(\frac{\partial w_m}{\partial x}\right)_{i,\,j,\,k+1/2} = \frac{1}{2}\left[\frac{\left(w_m\right)_{i+1} - \left(w_m\right)_i}{\Delta x_{i+1/2}} + \frac{\left(w_m\right)_i - \left(w_m\right)_{i-1}}{\Delta x_{i-1/2}}\right] \qquad (101)$$

$$\left(\frac{\partial u_m}{x_i \partial z}\right) = \frac{1}{2}\left[\frac{(u_m)_{i+1/2,\,j,\,k+1} - (u_m)_{i+1/2,\,j,\,k}}{x_{i+1/2}\,\Delta z_{k+1/2}} + \frac{(u_m)_{i-1/2,\,j,\,k+1} - (u_m)_{i-1/2,\,j,\,k}}{x_{i-1/2}\,\Delta z_{k+1/2}}\right] \tag{102}$$

$$\frac{(\mu_m)_t}{x_i}\left(\frac{\partial w_m}{\partial x}\right)_t \Delta V_t = \frac{(\mu_m)_t}{2\,x_i}\left[\frac{(w_m)_{i+1} - (w_m)_i}{\Delta x_{i+1/2}} + \frac{(w_m)_i - (w_m)_{i-1}}{\Delta x_{i-1/2}}\right]\Delta V_t \tag{103}$$

$$tr\,(D_m) = \frac{\partial u_m}{\partial x} + \frac{\partial v_m}{\partial y} + \frac{\partial w_m}{x\,\partial z} + \frac{u_m}{x} = \frac{1}{x}\,\frac{\partial (x u_m)}{\partial x} + \frac{\partial v_m}{\partial y} + \frac{\partial w_m}{x\,\partial z} \tag{104}$$

5.4 Zero Center Coefficient

The center coefficient of the discretized momentum equations may become zero, without the right-hand side becoming zero, at control volumes next to interfaces. An example of a typical y-momentum control volume outlined with bold lines is shown in Figure 5.3. The figure shows the grid near the surface of a fluidized bed. The bottom row of cells with dark shading shows the dense bed. The lightly shaded row of cells in the middle has a small amount of solids because of slight smearing of the interface. These cells did not have any solids before the iterations began. The situation shown occurs after the first few iterations. The top row of cells is still free of solids. We will examine the case of the solids velocity component in the y-direction, which is determined from the momentum control volume shown in the figure.

The average cell face velocities (in cm/s) from an actual case are shown in the figure. The solids viscosity values are zero at the six cell centers shown in Figure 5.3, because they are based on the conditions at the previous time step. When first order upwinding is used to discretize the equations, all the neighbor coefficients become zero; i.e.,

$$a_e = a_w = a_n = a_s = 0 \tag{105}$$

Since the cells are initially free of solids a_p^0 is also zero. Therefore, the center coefficient a_p is zero, when there are no momentum source terms.

The right-hand side of Equation 5.4, however, is non zero because of contributions from the fluid pressure gradient term

$$-A_N\,(\epsilon_m)_p\,\left((P_g)_N - (P_g)_S\right) = 1.76 \tag{106}$$

and from the gravity term

$$(\epsilon_m\,\rho_m)_p\,g_y\,\Delta V_n = -11. \tag{107}$$

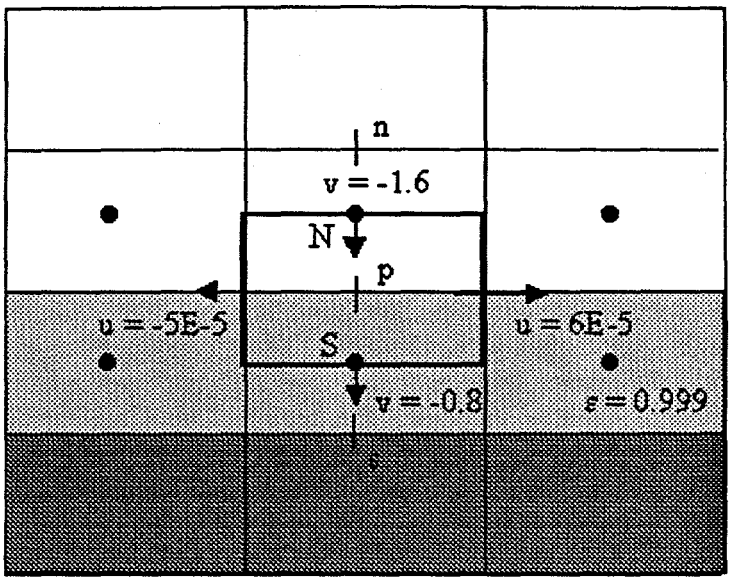


Figure 5.3. Example of conditions at an interface

By using the $(\epsilon_m)_p$ value (= 0) from the previous time step, we can make the right-hand side go to zero and, there by, avoid the singularity in the equations. This is not a useful solution, since this amounts to making the velocity component undefined, in a location where it is actually defined. Although the exact value of the velocity is not that important (considering the low value of solids volume fraction), the singularity in the discretized momentum equation must be removed to continue the computations. This is done by using an approximate momentum balance for such cells as illustrated below.

If (the right-hand side) $b > 0$ then $(v_m) > 0$. Now $(v_m)_N = f_N (v_m)_p + (1-f_N)(v_m)_n = (v_m)_p > 0$ because of the free-slip condition at the interface. And

$$(v_m)_S = f_S (v_m)_p + (1-f_S)(v_m)_s \approx f_S (v_m)_p > 0 \qquad (108)$$

assuming $(v_m)_p \gg (v_m)_s$. Then

$$a_p = a_s = (\epsilon_m \rho_m)_S f_S (v_m)_p A_s \qquad (109)$$

and we can solve for the velocity component as

$$(v_m)_p = \sqrt{\frac{b}{(\epsilon_m \rho_m)_S f_S A_s}} \qquad (110)$$

If $b < 0$ a similar argument will lead to

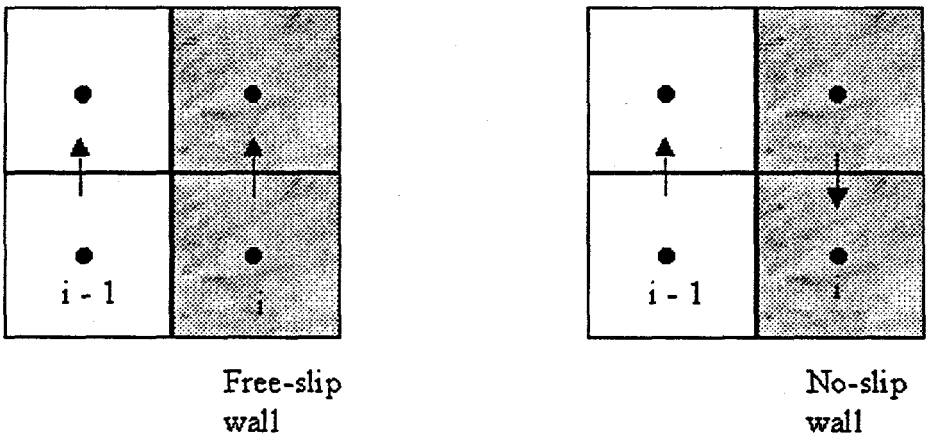
$$(v_m)_p = -\sqrt{\frac{-b}{(\epsilon_m \rho_m)_S f_S A_s}} \qquad (111)$$

5.5 Boundary Conditions

The implementation of wall boundary conditions in the linear equation solver is given below. The gas velocity component in the y-direction at an east-wall is used as an example (Figure 5.4). The implementation for the other components and locations is analogous.

1. Free-Slip wall

$$v_m\ (i, j+½, k) - v_m\ (i-1, j+½, k) = 0 \qquad (112)$$



Free-slip wall

No-slip wall

Figure 5.4. Free and no slip conditions at east-wall

2. No-Slip wall

$$v_m\ (i, j+½, k) + v_m\ (i-1, j+½, k) = 0$$

3. Partial-Slip wall

$$\frac{\partial v_m}{\partial n} + h_v (v_m - v_w) = 0$$

The discretized form of the above equation, for example, at east-wall is

$$v_m\ (i, j+½, k) \left(\frac{h_v}{2} + \frac{1}{\Delta x_E} \right) + v_m\ (i-1, j+½, k) \left(\frac{h_v}{2} - \frac{1}{\Delta x_E} \right) = h_v\ v_w$$

The above equation is a generalized slip condition, which can describe no-slip condition ($h_v \rightarrow \infty$, $v_w = 0$), free-slip condition ($h_v = 0$), and a specified wall velocity ($h_v \rightarrow \infty$, $v_w \neq 0$).

4. Velocity Boundary Condition at interfaces

At interfaces where the solids volume fraction goes to zero a free-slip condition is applied. For the conditions shown in Figure 5.5

$$-v_s(i, j+\tfrac{1}{2}, k) + v_s(i, j+1\tfrac{1}{2}, k) = 0 \qquad (116)$$

The following algorithm is used for setting this condition. The interface is identified with a threshold value of δ.

Algorithm 5.1

```
If (ε_s (i, j, k) < δ) then
        a_n = -1
        If (ε_s (i, j-1, k) > δ)
                a_s = 1
        else if (ε_s (i, j+1, k) > δ)
                a_n = 1
        else
                b = - v_s (i, j, k)
        endif
endif
```

This algorithm will fail in the rare occasion when two interfaces are separated by one numerical cell, however.

5. Internal Surfaces

MFIX allows the specification of internal surfaces that separate two adjacent cells with an infinitesimally thin wall. For impermeable surfaces the normal velocity is zero. For semipermeable internal surfaces the solids velocity is a user-defined constant and gas velocity is calculated as though the internal surface is a porous medium. No special treatment is needed for the convection terms. But always the diffusion across such surfaces is set to zero. This is done by first setting up the linear equations and then subtracting out the diffusion contributions for cells neighboring internal surfaces (two cells for scalar equations and four cells for velocity components).

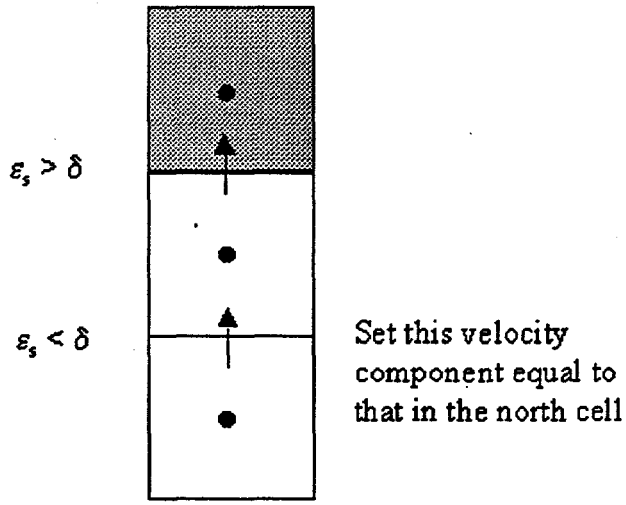


Figure 5.5. Free-slip condition at an interface

5.6 Linear Equation Setup

The linear equations for solving the momentum equation are set up as follows:

1. Calculate the average velocities at momentum cell faces.
2. Calculate the convection coefficients ξ.
3. Calculate the neighbor coefficients a_{nb}.
4. Modify the neighbor coefficients to account for the presence of internal surfaces (assumed to be free-slip walls).
5. Calculate the center coefficient and the source vector values. For impermeable walls and internal surfaces set all neighbor coefficients to zero and fix the normal velocity component at zero.

6 Partial Elimination of Interphase Coupling

As discussed earlier the presence of interphase transfer terms is a distinguishing feature of multiphase flow equations in comparison to single phase flow equations. Usually, the interphase transfer terms strongly couple the components of velocity and temperature in each phase to the corresponding variables in other phases. Decoupling of the equations by calculating the interphase transfer terms from the previous iteration values will make the iterations unstable or force the time step to be very small. The other extreme of solving all the discretized equations for a certain component together (e.g., equations for) will lead to a larger, nonstandard matrix. An effective alternative that maintains a higher degree of coupling between the equations while giving the standard septadiagonal matrix is the Partial Elimination Algorithm of Spalding (1980). The algorithm is illustrated with the following model equation:

$$\epsilon_m \rho_m \frac{\partial \phi_m}{\partial t} + \epsilon_m \rho_m (v_m)_i \frac{\partial \phi_m}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\Gamma_{\phi_m} \frac{\partial \phi_m}{\partial x_i}\right) + R_{\phi_m} - \phi_m \left(\sum R_{\ell m}\right) + \sum_{\ell=0}^{M} F_{\ell m} (\phi_\ell - \phi_m) \tag{117}$$

(Note that $F_{lm} = F_{ml}$ and $F_{mm} = 0$.)

The corresponding discretized equation is

$$(a_m)_P (\phi_m)_P = \sum_{nb} (a_m)_{nb} (\phi_m)_{nb} + b_m + \Delta V \sum_{\ell=0}^{M} F_{\ell m} \left[(\phi_\ell)_P - (\phi_m)_P\right] \tag{118}$$

which is similar in form to the discretized momentum equations discussed in Section 5.

We will first explain the problem with a straightforward decoupling of the equations. For example, consider the case of two-phase flow (M=1):

$$(a_0)_P (\phi_0)_P = \sum_{nb} (a_0)_{nb} (\phi_0)_{nb} + b_0 + \Delta V F_{10} \left[(\phi_1)_P - (\phi_0)_P\right] \tag{119}$$

$$(a_1)_P (\phi_1)_P = \sum_{nb} (a_1)_{nb} (\phi_1)_{nb} + b_1 + \Delta V F_{10} \left[(\phi_0)_P - (\phi_1)_P\right] \tag{120}$$

When $F_{10} \to 0$ the two equations are decoupled and the solution for $(\phi_0)_P$, for example, is

$$(\phi_0)_p = \frac{\sum_{nb} (a_0)_{nb} (\phi_0)_{nb} + b_0}{(a_0)_p} \tag{121}$$

When $F_{10} \to \infty$ the equations are strongly coupled and the solutions are

$$(\phi_0)_p = (\phi_1)_p = \frac{\sum_{nb} (a_0)_{nb} (\phi_0)_{nb} + b_0 + \sum_{nb} (a_1)_{nb} (\phi_1)_{nb} + b_1}{(a_0)_p + (a_1)_p} \quad (122)$$

An iteration scheme treating the interphase transfer term merely as a source term will give the correct solution for the case of small F_{10}, but will fail to give the correct solution for the case of $F_{10} \to \infty$. Therefore, in such an approach the time step must be made sufficiently small so that F_{10} is small in comparison to b0 and b1. For obtaining convergence while using large time steps, the iteration scheme must be designed such that it can calculate the above two limiting solutions. For this purpose, Spalding (1980) has suggested the following partial elimination algorithm:

Solve for $(\phi_1)_p$ from Equation 6.4 to get

$$(\phi_1)_p = \frac{\sum_{nb} (a_1)_{nb} (\phi_1)_{nb} + b_1 + \Delta V F_{10} (\phi_0)_p}{(a_1)_p + \Delta V F_{10}} \quad (123)$$

Substitute this in Equation 6.3 to get

$$\left[(a_0)_p + \frac{(a_1)_p \Delta V F_{10}}{(a_1)_p + \Delta V F_{10}}\right](\phi_0)_p = \sum_{nb} (a_0)_{nb} (\phi_0)_{nb} + b_0 + \frac{\Delta V F_{10}}{(a_1)_p + \Delta V F_{10}} \left\{\sum_{nb} (a_1)_{nb} (\phi_1)_{nb} + b_1\right\} \quad (124)$$

A similar procedure can be used to derive the equation for the other phase:

$$\left[(a_1)_p + \frac{(a_0)_p \Delta V F_{10}}{(a_0)_p + \Delta V F_{10}}\right](\phi_1)_p = \sum_{nb} (a_1)_{nb} (\phi_1)_{nb} + b_1 + \frac{\Delta V F_{10}}{(a_0)_p + \Delta V F_{10}} \left\{\sum_{nb} (a_0)_{nb} (\phi_0)_{nb} + b_0\right\} \quad (125)$$

The linear equation sets for ϕ_0 and ϕ_1 are decoupled by treating the last terms in the above equations (eq. 6.9 and 6.10) as a source term evaluated with $(\phi_1)_{nb}$ and $(\phi_0)_{nb}$ from the previous iteration. As $F_{10} \to 0$ and $F_{10} \to \infty$ we can recover the required limiting solutions from the above equations. Therefore, we expect an iteration scheme based on the above equation to converge for all values of F_{10}.

The above partial elimination procedure can be extended for multiple phases (M>1) to decouple multiphase equations. However, a matrix inversion is necessary for doing the partial elimination exactly. An approximate alternative (not yet tested in MFIX) is given in Appendix D.

7 Fluid Pressure Correction Equation

An important step in the algorithm is the derivation of a discretization equation for pressure (Step 3 in Algorithm 4.1), which is described in this section.

7.1 Formulation

The discretized x-momentum equations (see Section 5) for two phases, for example, are

$$a_{0p}\,(u_0)_p = \sum_{nb} a_{0nb}\,(u_0)_{nb} + b_0 - A_p(\epsilon_0)_p\left((P_g)_E - (P_g)_W\right) + F_{10}\left[(u_1)_p - (u_0)_p\right]\Delta V \tag{128}$$

and

$$a_{1p}\,(u_1)_p = \sum_{nb} a_{1nb}\,(u_1)_{nb} + b_1 - A_p(\epsilon_1)_p\left((P_g)_E - (P_g)_W\right) + F_{10}\left[(u_0)_p - (u_1)_p\right]\Delta V - A_p\left((P_s)_E - (P_s)_W\right) \tag{129}$$

where 0 denotes the fluid phase and $P_s = P_s\left(\epsilon_1\right)$ is the solids pressure.

As stated in Section 4, first we will solve Equations 7.1 and 7.2 using the pressure field P_g^* and the void fraction field ϵ_0^* from the previous iteration to calculate tentative values of the velocity fields -- u_0^* and u_1^* and other velocity components.

$$a_{0p}\,(u_0^*)_p = \sum_{nb} a_{0nb}\,(u_0^*)_{nb} + b_0 - A_p(\epsilon_0^*)_p\left((P_g^*)_E - (P_g^*)_W\right) + F_{10}\left((u_1^*)_p - (u_0^*)_p\right)\Delta V \tag{130}$$

$$a_{1p}\,(u_1^*)_p = \sum_{nb} a_{1nb}\,(u_1^*)_{nb} + b_1 - A_p(\epsilon_0^*)_p\left((P_g^*)_E - (P_g^*)_W\right) + F_{10}\left((u_0^*)_p - (u_1^*)_p\right)\Delta V - A_p\left((P_s^*)_E - (P_s^*)_W\right) \tag{131}$$

Let the actual values differ from the (starred) tentative values by the following corrections

$$\begin{aligned}(P_g)_E &= (P_g^*)_E + (P_g')_E\\ (P_s)_E &= (P_s^*)_E + (P_s')_E\\ (u_0)_p &= (u_0^*)_p + (u_0')_p\\ (u_1)_p &= (u_1^*)_p + (u_1')_p\end{aligned} \tag{132}$$

and similar formulas for other components of velocity.

Substitute the corrections (Equation 7.5) into Equations 7.1 and 7.2, and from the resulting equations subtract Equations 7.3 and 7.4 to get

$$a_{0p}\,(u_0')_p = \sum_{nb} a_{0nb}\,(u_0')_{nb} - A_p(\epsilon_0^*)_p\left((P_g')_E - (P_g')_W\right) + F_{10}\left((u_1')_p - (u_0')_p\right)\Delta V \quad (133)$$

$$a_{1p}\,(u_1')_p = \sum_{nb} a_{1nb}\,(u_1')_{nb} - A_p(\epsilon_1^*)_p\left((P_g')_E - (P_g')_W\right) + F_{10}\left((u_0')_p - (u_1')_p\right)\Delta V - A_p\left((P_s')_E - (P_s')_W\right) \quad (134)$$

To develop an approximate equation for fluid pressure correction, we drop the momentum convection and solids pressure terms to get

$$a_{0p}\,(u_0')_p = - A_p(\epsilon_0^*)_p\left((P_g')_E - (P_g')_W\right) + F_{10}\left((u_1')_p - (u_0')_p\right)\Delta V \quad (135)$$

$$a_{1p}\,(u_1')_p = - A_p(\epsilon_1^*)_p\left((P_g')_E - (P_g')_W\right) + F_{10}\left((u_0')_p - (u_1')_p\right)\Delta V \quad (136)$$

Note that the above simplifications would not affect the accuracy of the converged solution. They may, however, affect the rate of convergence of the iterations.

From Equation 7.9 we get

$$\left(a_{1p} + F_{10}\right)(u_1')_p = - A_p(\epsilon_1^*)_p\left((P_g')_E - (P_g')_W\right) + F_{10}\,(u_0')_p\,\Delta V \quad (137)$$

Substituting this in Equation 7.8 we get

$$a_{0p}\,(u_0')_p = -A_p(\epsilon_0^*)_p\left((P_g')_E - (P_g')_W\right) + F_{10}\left[\frac{-A_p(\epsilon_0^*)_p\left((P_g')_E - (P_g')_W\right) + F_{10}\,(u_0')_p\,\Delta V}{a_{1p} + F_{10}\,\Delta V} - (u_0')_p\right]\Delta V \quad (138)$$

Solving for $(u_0')_p$ we get

$$\left[a_{0p} + \frac{F_{10}\,\Delta V\,a_{10}}{a_{1p} + F_{10}\,\Delta V}\right](u_0')_p = - \left[A_p(\epsilon_0^*)_p + A_p(\epsilon_1^*)_p\,\frac{F_{10}\,\Delta V}{a_{1p} + F_{10}\,\Delta V}\right]\left((P_g')_E - (P_g')_W\right) \quad (139)$$

which can be written as

$$(u_0')_p = - d_{0p}\left((P_g')_E - (P_g')_W\right) \quad (140)$$

where

$$d_{0p} = \frac{A_p \left[(\epsilon_0^*)_p + \dfrac{(\epsilon_1^*)_p \, F_{10} \, \Delta V}{a_{1p} + F_{10} \, \Delta V} \right]}{\left[a_{0p} + \dfrac{F_{10} \, \Delta V \, a_{1p}}{a_{1p} + F_{10} \, \Delta V} \right]} \tag{141}$$

Similarly

$$(u_1')_p = - \, d_{1p} \left((P_g')_E - (P_g')_W \right) \tag{142}$$

where

$$d_{1p} = \frac{A_p \left[(\epsilon_1^*)_p + \dfrac{(\epsilon_0^*)_p \, F_{10} \, \Delta V}{a_{0p} + F_{10} \, \Delta V} \right]}{\left[a_{1p} + \dfrac{F_{10} \, \Delta V \, a_{0p}}{a_{0p} + F_{10} \, \Delta V} \right]} \tag{143}$$

$$(u_m)_p = (u_m^*)_p - d_{mp} \left((P_g')_E - (P_g')_W \right) \tag{144}$$

For the case of more than two phases an approximate formula is given in Appendix D. The velocity corrections are given by

Substituting the above equation and similar equations for other components of velocity into the fluid continuity equation (Equation 3.9 with $\phi = 1$), we get an equation for pressure correction.

$$\left(\frac{(\epsilon_0\rho_0)_P - (\epsilon_0\rho_0)_P^0}{\Delta t}\right) \Delta V$$

$$+ \left\{(\epsilon_0\rho_0)_E\, \xi_e + (\epsilon_0\rho_0)_P\, \bar{\xi}_e\right\} \left[(u_0^*)_e - d_{0e}\left((P_g')_E - (P_g')_P\right)\right] A_e$$

$$- \left\{(\epsilon_0\rho_0)_P\, \xi_w + (\epsilon_0\rho_0)_w\, \bar{\xi}_w\right\} \left[(u_0^*)_w - d_{0w}\left((P_g')_P - (P_g')_W\right)\right] A_w$$

$$+ \left\{(\epsilon_0\rho_0)_N\, \xi_n + (\epsilon_0\rho_0)_p\, \bar{\xi}_n\right\} \left[(v_0^*)_n - d_{0n}\left((P_g')_N - (P_g')_P\right)\right] A_n$$

$$- \left\{(\epsilon_0\rho_0)_P\, \xi_s + (\epsilon_0\rho_0)_s\, \bar{\xi}_s\right\} \left[(v_0^*)_s - d_{0s}\left((P_g')_P - (P_g')_S\right)\right] A_s$$

$$+ \left\{(\epsilon_0\rho_0)_T\, \xi_t + (\epsilon_0\rho_0)_p\, \bar{\xi}_t\right\} \left[(w_0^*)_t - d_0 t\left((P_g')_T - (P_g')_P\right)\right] A_t$$

$$- \left\{(\epsilon_0\rho_0)_P\, \xi_b + (\epsilon_0\rho_0)_B\, \bar{\xi}_b\right\} \left[(w_0^*)_b - d_{0b}\left((P_g')_P - (P_g')_B\right)\right] A_b$$

$$= \Delta V \sum_{\ell} (R_{\ell m})_P \qquad \textbf{(145)}$$

which can be written in the standard form

$$a_P\, (P_g')_P = \sum_{nb} a_{nb}\, (P_g')_{nb} + b \qquad \textbf{(146)}$$

where

$$a_E = \left\{(\epsilon_0\rho_0)_E\, \xi_e + (\epsilon_0\rho_0)_P\, \bar{\xi}_e\right\} d_{0e}\, A_e$$

$$a_W = \left\{(\epsilon_0\rho_0)_P\, \xi_w + (\epsilon_0\rho_0)_W\, \bar{\xi}_w\right\} d_{0w}\, A_w$$

$$a_N = \left\{(\epsilon_0\rho_0)_N\, \xi_n + (\epsilon_0\rho_0)_P\, \bar{\xi}_n\right\} d_{0n}\, A_n$$

$$a_S = \left\{(\epsilon_0\rho_0)_P\, \xi_s + (\epsilon_0\rho_0)_S\, \bar{\xi}_s\right\} d_{0s}\, A_s$$

$$a_T = \left\{(\epsilon_0\rho_0)_T\, \xi_t + (\epsilon_0\rho_0)_P\, \bar{\xi}_t\right\} d_{0t}\, A_t$$

$$a_B = \left\{(\epsilon_0\rho_0)_P\, \xi_b + (\epsilon_0\rho_0)_B\, \bar{\xi}_b\right\} d_{0b}\, A_b \qquad \textbf{(147)}$$

$$a_P = a_E + a_W + a_N + a_S + a_T + a_B \qquad \textbf{(148)}$$

$$
\begin{aligned}
b = - \Bigg\{ & \left(\frac{(\epsilon_0 \rho_0)_P - (\epsilon_0 \rho_0)_P^0}{\Delta t} \right) \Delta V \\
& + \left[(\epsilon_0 \rho_0)_E \, \xi_e + (\epsilon_0 \rho_0)_P \, \bar{\xi}_e \right] u_{0e}^* \, A_e \\
& - \left[(\epsilon_0 \rho_0)_P \, \xi_w + (\epsilon_0 \rho_0)_W \, \bar{\xi}_w \right] u_{0w}^* \, A_w \\
& + \left[(\epsilon_0 \rho_0)_N \, \xi_n + (\epsilon_0 \rho_0)_P \, \bar{\xi}_n \right] v_{0n}^* \, A_n \\
& - \left[(\epsilon_0 \rho_0)_P \, \xi_s + (\epsilon_0 \rho_0)_S \, \bar{\xi}_s \right] v_{0s}^* \, A_s \\
& + \left[(\epsilon_0 \rho_0)_T \, \xi_t + (\epsilon_0 \rho_0)_P \, \bar{\xi}_t \right] w_{0t}^* \, A_t \\
& - \left[(\epsilon_0 \rho_0)_P \, \xi_b + (\epsilon_0 \rho_0)_B \, \bar{\xi}_b \right] w_{0b}^* \, A_b \\
& - \Delta v \sum_{\ell} (R_{\ell m})_P \Bigg\}
\end{aligned}
\tag{149}
$$

After solving Equation 7.19 for the fluid-pressure corrections, the fluid and solids velocities are corrected. Note that when the tentative fluid velocity field satisfies the continuity equation, the pressure corrections will go to zero. Also the corrected fluid velocity field is such that it satisfies the continuity equation.

7.2 Mildly Compressible Flow

In compressible flows the term $\left(\frac{(\epsilon_0 \, \rho_0)_P - (\epsilon_0 \, \rho_0)_P^0}{\Delta t} \right) \Delta V$ in Equation 7.22 will make the calculations unstable. In mildly compressible flows this problem may be solved by accounting for the effect of pressure on fluid density.

$$
\begin{aligned}
\rho_0 &= \rho_0 (P_g) \\
&\approx \rho_0 (P_g^*) + \left(\frac{\partial \rho_0}{\partial P_g} \right)_{\rho_0^*} (P_g - P_g^*) \\
&\approx \rho_0^* + \left(\frac{\partial \rho_0}{\partial P_g} \right)^* P_g'
\end{aligned}
\tag{150}
$$

When this correction is inserted into the pressure correction equation, only the center-coefficient needs to be changed:

$$a_P = \sum_{nb} a_{nb} + \epsilon_0 \left(\frac{\partial \rho_0}{\partial P_g} \right)^* \frac{\Delta V}{\Delta t} \tag{151}$$

7.3 Boundary Conditions

The boundary conditions for the pressure correction equations at the inflow and outflow boundaries are formulated as follows. Figure 7.1 shows the fictitious (boundary) cell and the adjacent internal cell for two cases. The fictitious cells are shaded. Obviously, no pressure correction equation is available for the fictitious cells. The pressure correction equation for the adjacent internal cell is modified as follows by using information from the boundary conditions.

1. Specified Velocity

For the inflow condition shown in Figure 7.1, by substituting the specified velocity in Equation (7.18) we find that

$$a_S = 0$$

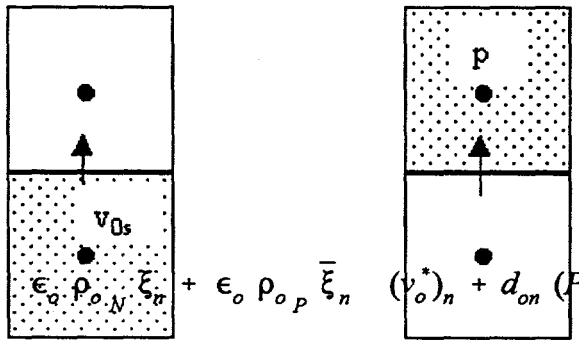
$$a_P = a_E + a_W + a_N + a_T + a_B \tag{152}$$

$(v_o^*)_s$ in b (Equation 7.22) is the same as $(v_o)_s$ specified at the inflow boundary.

The inflow boundaries at other locations (E, W, N, T, and B) are treated similarly.

The boundary condition at impermeable walls is similar to that of inflow boundaries since the normal velocity is specified as zero.

2. Specified Pressure



When the pressure is specified in the pressure correction in that cell is zero, and for the conditions shown in Figure 7.1 Equation (26) becomes

$$\ldots = \ldots + \epsilon_o \rho_{o_N} \bar{\xi}_n + \epsilon_o \rho_{o_P} \bar{\xi}_n \left((v_o^*)_n + d_{on} (P_g')_p \right) A_n + \ldots \tag{154}$$

Specified velocity

Specified pressure

Therefore, $a_N = 0$

$$\xi_n + (\epsilon_o \rho_o)_P \tag{155}$$

Figure 7.1. Flow boundary conditions

8 Solids Volume Fraction Correction Equation

The success of the numerical technique critically depends upon its ability to handle dense packing of solids. MFIX calculations in that limit are stabilized by including the effect of solids pressure, in the discretized solids continuity equation. This is accomplished by deriving a solids volume fraction correction equation as described in this section.

8.1 Convection Term

For this method to work we need a state equation that relates solids pressure to solids volume fraction

$$P_m = P_m(\epsilon_m) \tag{156}$$

and we define

$$K_m = \frac{\partial P_m}{\partial \epsilon_m} \tag{157}$$

Then, a small change in the solids pressure can be calculated as a function of the change in solids volume fraction:

$$P_m' = K_m \epsilon_m' \tag{158}$$

As discussed before, integrating the convection term over a control volume we get, for example,

$$\int \frac{d}{dx} (\epsilon_m \rho_m u_m)\, dV = (\rho_m \epsilon_m u_m)_e A_e - (\rho_m \epsilon_m u_m)_w A_w \tag{159}$$

We need to develop formulas for calculating fluxes such as $(\rho_m \epsilon_m u_m)_e$.

Denote the solids velocity obtained from the tentative solids pressure field and solids volume fraction field as $(u_m^*)_e$. This is the solids velocity field obtained at the end of Step 4 in Algorithm 4.1. The actual solids velocity can be represented as

$$(u_m)_e = (u_m^*)_e + (u_m')_e \tag{160}$$

where the correction $(u_m')_e$ is related to the correction in the solids pressure field as

$$(u_m')_e = e_e \left((P_m')_P - (P_m')_E \right) \tag{161}$$

which is derived similar to that described in Section 7. Now substituting from Equation (8.3) we get

$$\left(u_m'\right)_e = e_e \left[(K_m)_P \left(\epsilon_m'\right)_p - (K_m)_E \left(\epsilon_m'\right)_E\right] \tag{162}$$

Also, the volume fractions can be expressed as a sum of the current value plus a correction

$$\left(\epsilon_m\right)_e = \left(\epsilon_m^*\right)_e + \left(\epsilon_m'\right)_e \tag{163}$$

Combining Equations (8.7) and (8.8) we get

$$\begin{aligned}\left(\epsilon_m\right)_e \left(u_m\right)_e &\approx \left(\epsilon_m^*\right)_e \left(u_m^*\right)_e + \left(\epsilon_m'\right)_e \left(u_m^*\right)_e + \left(\epsilon_m^*\right)_e \left(u_m'\right)_e \\ &\approx \left(\epsilon_m^*\right)_e \left(u_m^*\right)_e + \left(\epsilon_m'\right)_e \left(u_m^*\right)_e + \left(\epsilon_m^*\right)_e e_e \left[(K_m)_P \left(\epsilon_m'\right)_P - (K_m)_E \left(\epsilon_m'\right)_E\right]\end{aligned} \tag{164}$$

where we have ignored the product of the corrections.

Recall that the cell face values can be written as a function of the cell center values using convection factors (Equation 2.31); e.g.,

$$\left(\epsilon_m\right)_e = \left(\epsilon_m\right)_E \xi_e + \left(\epsilon_m\right)_P \bar{\xi}_e \tag{165}$$

Now the flux $(\rho_m \epsilon_m u_m)_e$ can be expressed as (ρ_m is a constant in the current version of MFIX)

$$\begin{aligned}\left(\rho_m\right)_e \left(\epsilon_m\right)_e \left(u_m\right)_e \approx{}& \left(\rho_m\right)_e \left(\epsilon_m^*\right)_e \left(u_m^*\right)_e + \left(\rho_m\right)_e \left[\xi_e \left(\epsilon_m'\right)_E + \bar{\xi}_e \left(\epsilon_m'\right)_P\right] \left(u_m^*\right)_e \\ &+ \left(\rho_m\right)_e \left(\epsilon_m^*\right)_e e_e \left[(K_m)_p \left(\epsilon_m'\right)_P - (K_m)_E \left(\epsilon_m'\right)_E\right]\end{aligned} \tag{166}$$

which can be rearranged as

$$\begin{aligned}\left(\rho_m\right)_e \left(\epsilon_m\right)_e \left(u_m\right)_e \approx{}& \left(\rho_m\right)_e \left(\epsilon_m^*\right)_e \left(u_m^*\right)_e \\ &+ \left(\rho_m\right)_e \left[\bar{\xi}_e \left(u_m^*\right)_e + \left(\epsilon_m^*\right)_e (K_m)_p e_e\right] \left(\epsilon_m'\right)_P \\ &+ \left(\rho_m\right)_e \left[\xi_e \left(u_m^*\right)_e - \left(\epsilon_m^*\right)_e (K_m)_E e_e\right] \left(\epsilon_m'\right)_E\end{aligned} \tag{167}$$

8.2 Transient Term

$$\int \frac{\partial}{\partial t} (\epsilon_m \rho_m)_P \, dV = \frac{\left[(\epsilon_m)_P (\rho_m)_P - (\epsilon_m)_P^0 (\rho_m)_P^0\right]}{\Delta t} \Delta V$$

$$= \frac{\left[\left((\epsilon_m^*)_P + (\epsilon_m')_P\right)(\rho_m)_P - (\epsilon_m)_P^0 (\rho_m)_P^0\right]}{\Delta t} \Delta V \tag{168}$$

$$= \frac{(\epsilon_m')_P (\rho_m)_P \Delta V}{\Delta t} + \frac{\left[(\epsilon_m^*)_P (\rho_m)_P - (\epsilon_m)_P^0 (\rho_m)_P^0\right]}{\Delta t} \Delta V$$

8.3 Generation Term

The generation term is manipulated as described in Section 3.

$$\int \Sigma R_{\ell m} dV = \Delta V \, \Sigma R_{\ell m} = \Delta V \left\{[\Sigma R_{\ell m}] - [-\Sigma R_{\ell m}]\right\}$$

$$= \Delta V \left\{[\Sigma R_{\ell m}] - [-\Sigma R_{\ell m}] \frac{(\rho_m \epsilon_m)_P}{(\rho_m \epsilon_m)_P}\right\}$$

$$= \Delta V \left\{[\Sigma R_{\ell m}] - [-\Sigma R_{\ell m}] \frac{(\rho_m)_P (\epsilon_m^* + \epsilon_m')_P}{(\rho_m \epsilon_m^*)_P}\right\}$$

$$= \Delta V \left\{[\Sigma R_{\ell m}] - [-\Sigma R_{\ell m}] - [-\Sigma R_{\ell m}] \frac{(\rho_m)_P (\epsilon_m')_P}{(\rho_m \epsilon_m^*)_P}\right\}$$

$$= \Delta V \, \Sigma R_{\ell m} - [-R_{\ell m}] \frac{(\epsilon_m')_p (\rho_m)_P}{(\rho_m \epsilon_m^*)_p} \Delta V \tag{169}$$

8.4 Correction Equation

Collecting all the terms, an equation for volume fraction correction can be written as:

$$a_p (\epsilon_m')_P = \sum_{nb} a_{nb} (\epsilon_m')_{nb} + b \tag{170}$$

$$a_E = \left[(\rho_m \epsilon_m)_e^* e_e (K_m)_E - \xi_e (\rho_m)_E (u_m)_e^*\right] A_e \tag{171}$$

$$a_W = \left[(\rho_m \epsilon_m)_w^* e_w (K_m)_W + \bar{\xi}_w (\rho_m)_W (u_m^*)_w\right] A_w \tag{172}$$

$$a_N = \left[(\rho_m \epsilon_m)_n^* e_n (K_m)_N - \xi_n (\rho_m)_N (v_m^*)_n\right] A_n \tag{173}$$

$$a_S = \left[\left(\rho_m \, \epsilon_m \right)_s^* e_s \, (K_m)_S + \bar{\xi}_s \, (\rho_m)_S \left(v_m^* \right)_s \right] A_s \tag{174}$$

$$a_T = \left[\left(\rho_m \, \epsilon_m \right)_t^* e_t \, (K_m)_T - \xi_t \, (\rho_m)_T \left(w_m^* \right)_t \right] A_t \tag{175}$$

$$a_B = \left[\left(\rho_m \, \epsilon_m^* \right)_b e_b \, (K_m)_B + \bar{\xi}_b \, (\rho_m)_B \left(w_m^* \right)_b \right] A_b \tag{176}$$

$$\begin{aligned} a_P = {} & (\rho_m)_P \left[\bar{\xi}_e \left(u_m^* \right)_e A_e - \xi_w \left(u_m^* \right)_w A_w \right. \\ & + \bar{\xi}_n \left(v_m^* \right)_n A_n - \xi_s \left(v_m^* \right)_s A_s \\ & \left. + \bar{\xi}_t \left(w_m^* \right)_t A_t - \xi_b \left(w_m^* \right)_b A_b \right] \\ & + (K_m)_P \left[\left(\rho_m \, \epsilon_m^* \right)_e e_e A_e + \left(\rho_m \, \epsilon_m^* \right)_w e_w A_w \right. \\ & + \left(\rho_m \, \epsilon_m^* \right)_n e_n A_n + \left(\rho_m \, \epsilon_m^* \right)_s e_s A_s \\ & \left. + \left(\rho_m \, \epsilon_m^* \right)_t e_t A_t + \left(\rho_m \, \epsilon_m^* \right)_b e_b A_b \right] \\ & + (\rho_m)_P \frac{\Delta V}{\Delta t} + \left[-\Sigma \, R_{\ell m} \right] \frac{(\rho_m)_P \, \Delta V}{\left(\rho_m \, \epsilon_m^* \right)_P} \end{aligned} \tag{177}$$

$$\begin{aligned} b = {} & - \left(\rho_m \, \epsilon_m^* \right)_e \left(u_m^* \right)_e A_e + \left(\rho_m \, \epsilon_m^* \right)_w \left(u_m^* \right)_w A_w \\ & - \left(\rho_m \, \epsilon_m^* \right)_n \left(v_m^* \right)_n A_n + \left(\rho_m \, \epsilon_m^* \right)_s \left(v_m^* \right)_s A_s \\ & - \left(\rho_m \, \epsilon_m^* \right)_t \left(w_m^* \right)_t A_t + \left(\rho_m \, \epsilon_m^* \right)_b \left(w_m^* \right)_b A_b \\ & - \left[\left(\rho_m \, \epsilon_m^* \right)_P - \left(\rho_m \, \epsilon_m \right)_P^0 \right] \frac{\Delta V}{\Delta t} \\ & + \Delta V \, \Sigma \, R_{\ell m} \end{aligned} \tag{178}$$

After calculating the solids volume fraction correction from Equation (8.15), the solids velocities (Equation 8.5 and 8.7) and solids volume fractions (Equation 8.8) are corrected. No under relaxation is applied to such corrections, since we want to maintain the solids mass balance to machine precision during the iterations. One exception to this is a selective under-relaxation applied in densely packed regions.

8.5 Selective Under Relaxation for Packed Regions

The solids pressure is an exponentially increasing function of the solids volume fraction as the packing limit is approached (Figure 8.1). Under dense packed conditions, a small increase in the solids volume fraction will cause a large increase in the solids pressure. To moderate such rapid changes in the solids pressure that leads to numerical instability, solids volume fraction

corrections are under relaxed in packed regions when the solids volume fraction is increasing (Figure 8.1):

Algorithm 8.1

$$If \left(\epsilon_m\right)_{new} > \left(\epsilon\right)_{cp} \; and \; \epsilon_m' > 0$$

$$\epsilon_m' = \omega_{\epsilon m} \, \epsilon_m' \tag{179}$$

where $\omega_{\epsilon m}$ is an under relaxation factor.

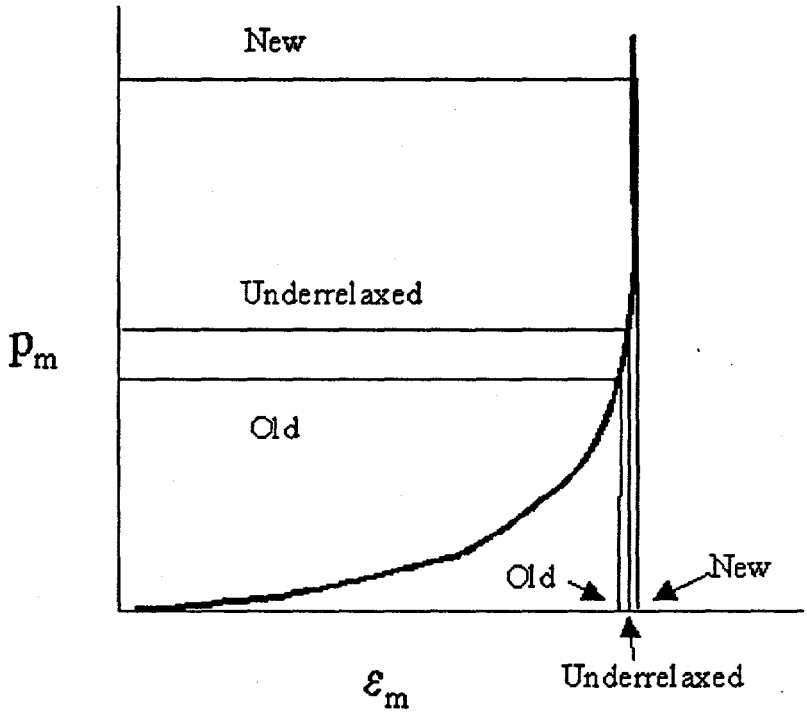


Figure 8.1. Iterative Adjustment of Solids Volume Fraction and Solids Pressure

9 Energy and Species Equations

The discretization of energy and species balance equations is similar to that of the scalar transport equation described in Section 3. The energy equations are coupled because of interphase heat transfer and are partially decoupled with the algorithm described in Section 6.

9.1 Heat Loss at the Wall

The wall boundary condition for energy equations is given by

$$\frac{\partial T_m}{\partial n} + h_m \left(T_m - T_w\right) = c_m \tag{180}$$

The heat loss can be calculated from

$$\begin{aligned} heat\ loss &= -\frac{K \partial T_m}{\partial n} \\ &= K_m \left(h_m \left(T_m - T_w\right) - c_m\right) \end{aligned} \tag{181}$$

When h becomes large the above method becomes inaccurate. Then, the heat loss is calculated from the temperature gradient at the wall:

west-wall at (i, j, k):

$$heat\ loss = \frac{K_m \left((T_m)_{i+2,\,j,\,k} - (T_m)_{i+1,\,j,\,k}\right)}{\Delta x_{i+1\frac{1}{2},\,j,\,k}} A_{i+\frac{1}{2},\,j,\,k} \tag{182}$$

east-wall at (i, j, k):

$$heat\ loss = \frac{-K_m \left((T_m)_{i-1,\,j,\,k} - (T_m)_{i-2,\,j,\,k}\right)}{\Delta x_{i-1\frac{1}{2},\,j,\,k}} A_{i-\frac{1}{2},\,j,\,k} \tag{183}$$

9.3 Radiation

A radiation source shown below is present in the MFIX energy equations:

$$S = \gamma_{Rm} \left(T_{Rm}^4 - T_m^4\right) \tag{184}$$

To ensure stability and help convergence, the term is discretized as follows:

$$S(T_m) = \gamma_{Rm}\left(T_{Rm}^4 - T_m^4\right)$$
$$= S\left(T_m^{\ell}\right) + \left.\frac{\partial S}{\partial T_m}\right|^{\ell}\left(T_m - T_m^{\ell}\right) \tag{185}$$

where superscript 'ℓ' indicates values at last iteration

$$\frac{\partial S}{\partial T_m} = -4\,\gamma_{Rm}\,T_m^3 \tag{186}$$

Then

$$\begin{aligned} S\left(T_m\right) &= \gamma_{Rm}\left[T_{Rm}^4 - \left(T_m\ell\right)^4\right] - 4\,\gamma_{Rm}\left(T_M^{\ell}\right)^3\left(T_m - T_m^{\ell}\right) \\ &= \gamma_{Rm}\left[T_{Rm}^4 - \left(T_m^{\ell}\right)^4 + 4\left(T_m^{\ell}\right)^4\right] - 4\gamma_{Rm}\left(T_m^{\ell}\right)^3 T_m \\ &= \underbrace{\gamma_{Rm}\left[T_{Rm}^4 + 3\left(T_m^{\ell}\right)^4\right]}_{\bar{S}} - \underbrace{4\,\gamma_{Rm}\left(T_m^{\ell}\right)^3 T_m}_{S'} \end{aligned} \tag{187}$$

The first term on the right-hand side is added to the source term and the second term is added to the center coefficient.

10 Final Steps

10.1. Under relaxation

All the discretized equations have the form

$$a_p \phi_p = \sum_{nb} a_{nb} \phi_{nb} + b \tag{188}$$

To ensure the stability of the calculations, it is necessary to underrelax the changes in the field variables during iterations.

$$\frac{a_p}{\omega_\phi} \phi_p = \sum_{nb} a_{nb} \phi_{nb} + b + \frac{(1 - \omega_\phi)}{\omega_\phi} a_p \phi_p^* \tag{189}$$

where $0 \leq \omega_\phi \leq 1$. When $\omega_\phi = 0$ the old value remains unchanged.

Applying the under relaxation factor first is better than to solve the equations first and then apply Under relaxation as $\left[\phi_p = \phi_p^* + \omega_\phi (\phi_p - \phi_p^*)\right]$ because of the better conditioning of the linear equation set and the consequent savings in the solution time.

10.2. Linear Equation Solvers

The final step in obtaining a solution is to solve the linear equations of the form 10.2 that result from the discretization of transport equations. The linear equation solver options available in MFIX are listed in Table 10.I. We use only iterative solvers as we always have a good initial guess for the solution. As initial guess the solution from the previous iteration is used for all equations, except the pressure and void fraction correction equations, which use zero as the starting guess.

During any iteration the linear equations need not be solved to a high degree of accuracy, because the solution gets modified in the next iteration and in the final iteration the initial guess is as good as the converged solution. A high degree of convergence in the linear equation solver will needlessly increase the computational time. On the other hand, poor convergence in the linear equation solver can increase the number of iterations and lead to nonconvergence of the iterations. An optimum degree of convergence has been determined from experience and is controlled by a specified number of iterations inside the linear equation solver. The user may change this value from the MFIX data file.

Table 10.I. Linear equation solver options

Method	Description	Source
SOR	Point successive over relaxation	-
IGCG	Idealized Generalized Conjugate Gradient	Kapitza and Eppel (1987)

Method	Description	Source
IGMRES	Incomplete LU Factorization + GMRES	SLAP (Seager and Greenbaum 1988)
DGMRES	Diagonal scaling + GMRES	SLAP (Seager and Greenbaum 1988)
BCGS	Incomplete LU Factorization + Biconjugate Gradient Square	SLAP (Seager and Greenbaum 1988)

A combination of SOR and IGCG was found to give the lowest run time and is set as the default in MFIX. IGCG is used for pressure and void fraction correction equations and energy and species balance equations. The momentum balance equations are solved with SOR. The user may change these settings from the MFIX data file

10.3. Calculation of Residuals

The convergence of iterations is judged from the residuals of various equations. The residuals are calculated before under relaxation is applied to the linear equation set. The standard form of the linear equation set is

$$a_P \phi_P = b + a_E \phi_E + a_W \phi_W + a_N \phi_N + a_S \phi_S + a_T \phi_T + a_B \phi_B \tag{190}$$

Denoting the current value as ϕ^*, the residual at point P is given by

$$R_{\phi P} = b + a_E \phi_E^* + a_W \phi_W^* + a_N \phi_N^* + a_S \phi_S^* + a_T \phi_T^* + a_B \phi_B^* - a_P \phi_P^* \tag{191}$$

Then a normalized residual for the whole computational domain is calculated from

$$\bar{R}_\phi = \frac{\sum_P |R_{\phi_P}|}{\sum_P |a_P \phi_P|} \tag{192}$$

For velocity components the denominator is replaced by $a_p \sqrt{u_P^2 + v_P^2 + w_P^2}$.

For fluid pressure and solids volume fraction correction equations, we know *a priori* that at convergence all the corrections must go to zero, which corresponds to the requirement that vector b becomes identically zero. Thus the convergence of those equations may be accurately judged from the norm of b, which turns out to be the residual of the continuity equations. This value, however, cannot be normalized as in Equation 10.5, since the denominator vanishes when convergence is achieved. Therefore, the norm of b is normalized with the norm of b for the first iteration.

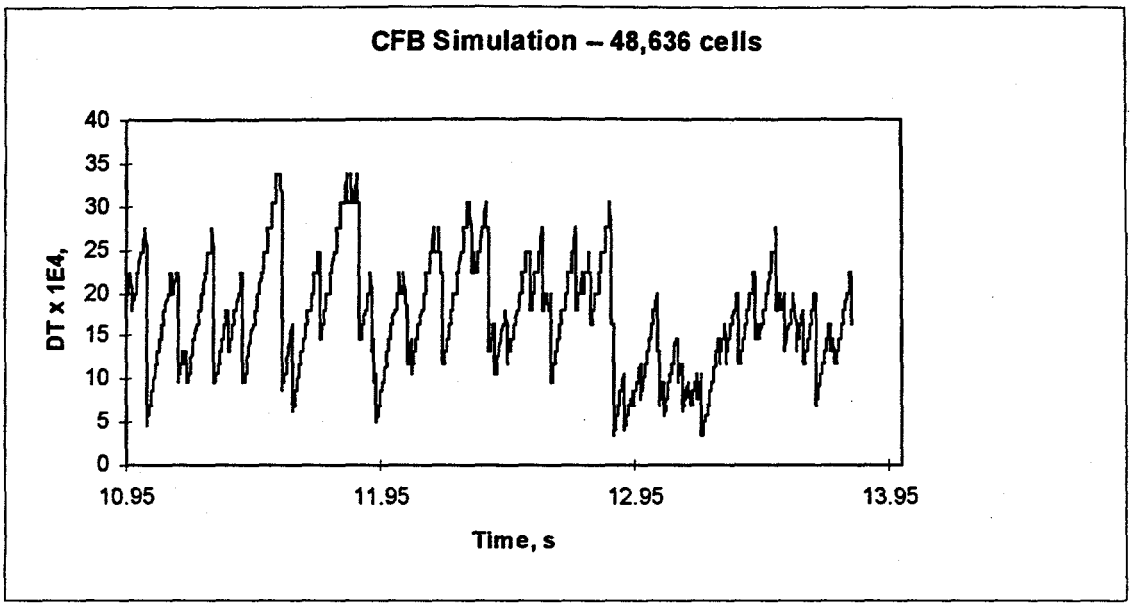


Figure 10.1. Time step adjustment history for a typical run

10.4. Time Step Adjustment

The semi-implicit algorithm imposes a time-step limitation that is particularly severe for dense gas-solids flow simulations. Too large a time step will make the calculations unstable. Too small a time-step, on the other hand, will make the calculations needlessly slow. A small time step is often needed to follow certain rapid changes in the flow field. After such events subside, the time steps may be increased. MFIX uses an automatic time step adjustment to reduce the run time. This is done by making small upward or downward adjustments in time steps and monitoring the total number iterations for several time steps. The adjustments are continued, if there is a favorable reduction in the number of iterations per second of simulation. Otherwise, adjustments in the opposite direction are attempted. Often the simulation will fail to converge, in which case the time step is decreased till convergence is obtained. Figure 10.1 shows the time step adjustment history for a circulating fluidized bed simulation. The large decreases in the time-step were caused by convergence failures.

Acknowledgement

I would like to thank Thomas J. O'Brien and Edward J. Boyle of DOE/FETC for their support, encouragement and fruitful discussions regarding this work. Thanks are also due to Debbie K. Sugg for typing all the equations.

11 References

Fluent User's Manual, 1996, Fluent, Inc., Lebanon, NH.

Fogt, H., and Peric, M., 1994, "Numerical calculation of gas-liquid flow using a two-fluid finite-volume method," FED-Vol. 185, Numerical Methods in Multiphase Flows, ASME, 73-80.

Gaskell, P.H. and A.K.C. Lau, 1988, "Curvature-compensated convective transport: SMART, a new boundedness-preserving transport algorithm," *Int. J. Numer. Methods in Fluids*, **8**, 617-641.

Gidaspow, D., and B. Ettehadieh, 1983, "Fluidization in Two-Dimensional Beds with a Jet; 2. Hydrodynamic Modeling," *I&EC Fundamentals*, **22**, 193-201.

Harlow, F.H., and A.A. Amsden, 1975, "Numerical Calculation of Multiphase Fluid Flow," *J. of Comp. Physics*, **17**, 19-52.

Kapitza, H., and D. Eppel, 1987, "A 3-D Poisson solver based on conjugate gradients compared to standard iterative methods and its performsnce on vector computers," *J. Computational Physics*, **68**, 474-484.

Laux, H., and Johansen, S.T., "Computer Simulation of Bubble Formation in a Gas-Fluidized Bed, 1997, Submitted to Fluidization IX conference.

Leonard, B.P., 1979, "A stable and accurate convective modeling procedure based on quadratic upstream interpolation," *Comput. Methods Appl. Mech. Eng.*, **19**, 59-98.

Leonard, B.P., S. Mokhtari, 1990, "Beyond first-order upwinding: the ultra sharp alternative for non-oscillatory steady-state simulation of convection," *Int. J. Numer. Methods Eng.*, **30**, 729(1990).

Patankar, S.V., 1980, Numerical Heat Transfer and Fluid Flow, Hemisphere Publishing Corporation, New York.

Pritchett, J.W., Blake, T.R., and Garg, S.K., 1978, "A Numerical Model of Gas Fluidized Beds," *AIChE Symp. Series No. 176*, **74**, 134-148.

Rivard, W.C., and M.D. Torrey, 1977, "K-FIX: A Computer Program for Transient, Two-Dimensional, Two-Fluid Flow," LA-NUREG-6623, Los Alamos National Laboratory, Los Alamos.

Seager, M.K., and A. Greenbaum, 1988, "The Linear Algebra Package SLAP Ver 2.0," Lawrence Livermore National Lab.

Spalding, D.B., 1980, "Numerical Computation of Multi-phase fluid flow and heat transfer," in Recent Advances in Numerical Methods in Fluids, C. Taylor et al., eds, Pineridge Press.

Syamlal, M., and O'Brien, T.J., 1988, "Simulation of Granular Layer Inversion in Liquid Fluidized Beds," *Int. J. Multiphase Flow*, **14**, 473-481.

Syamlal, M., W. Rogers, and T.J. O'Brien, 1993, "MFIX Documentation: Theory Guide," Technical Note, DOE/METC-94/1004, NTIS/DE94000087, National Technical Information Service, Springfield, VA.

Syamlal, M., 1994, "MFIX Documentation: User's Manual, 1994, " Technical Note, DOE/METC-95/1013, NTIS/DE95000031, National Technical Information Service, Springfield, VA.

Syamlal, M., 1997, "Higher order discretization methods for the numerical simulation of fluidized beds," presented at AIChE Annual Meeting, Los Angeles, CA.

Witt, P.J., and Perry, J.H., 1996, "A Study in Multiphase Modelling of Fluidised Beds," in Computational Techniques and Applications CTAC95, eds. R.L. May and A.K. Easton, World Scientific Publishing Co.

Appendix A: Summary of Equations

A.1 Equations

The equations solved in version 2.0 of MFIX are summarized in this section.

Gas continuity:

$$\frac{\partial}{\partial t}(\epsilon_g \rho_g) + \nabla\cdot(\epsilon_g \rho_g \vec{v}_g) = \sum_{n=1}^{N_g} R_{gn} \tag{198}$$

Solids continuity:

$$\frac{\partial}{\partial t}(\epsilon_{sm} \rho_{sm}) + \nabla\cdot(\epsilon_{sm} \rho_{sm} \vec{v}_{sm}) = \sum_{n=1}^{N_{sm}} R_{smn} \tag{199}$$

Gas momentum balance:

$$\begin{aligned}\frac{\partial}{\partial t}(\epsilon_g \rho_g \vec{v}_g) + \nabla\cdot(\epsilon_g \rho_g \vec{v}_g \vec{v}_g) = -\epsilon_g \nabla P_g + \nabla\cdot\bar{\bar{\tau}}_g + \sum_{m=1}^{M} F_{gm}(\vec{v}_{sm} - \vec{v}_g) + \vec{f}_g \\ + \epsilon_g \rho_g \vec{g} - \sum_{m=1}^{M} R_{0m}\left[\xi_{0m}\vec{v}_{sm} + \bar{\xi}_{0m}\vec{v}_g\right]\end{aligned} \tag{200}$$

Solids momentum balance:

$$\begin{aligned}\frac{\partial}{\partial t}(\epsilon_{sm} \rho_{sm} \vec{v}_{sm}) + \nabla\cdot(\epsilon_{sm} \rho_{sm} \vec{v}_{sm} \vec{v}_{sm}) = -\epsilon_{sm} \nabla P_g - \nabla\cdot\bar{\bar{S}}_{sm} \\ - F_{gm}(\vec{v}_{sm} - \vec{v}_g) + \sum_{l=1}^{M} F_{slm}(\vec{v}_{sl} - \vec{v}_{sm}) \\ + \epsilon_{sm} \rho_{sm} \vec{g} - \sum_{l=0}^{M} R_{ml}\left[\xi_{ml}\vec{v}_{sl} + \bar{\xi}_{ml}\vec{v}_{sm}\right]\end{aligned} \tag{201}$$

Gas energy balance:

$$\epsilon_g \rho_g C_{pg}\left(\frac{\partial T_g}{\partial t} + \vec{v}_g \cdot \nabla T_g\right) = -\nabla \cdot \vec{q}_g + \sum_{m=1}^{M} \gamma_{gm}(T_{sm} - T_g) - \Delta H_{rg} + \gamma_{Rg}(T_{Rg}^4 - T_g^4) \quad (202)$$

Solids energy balance:

$$\epsilon_{sm} \rho_{sm} C_{psm}\left(\frac{\partial T_{sm}}{\partial t} + \vec{v}_{sm} \cdot \nabla T_{sm}\right) = -\nabla \cdot \vec{q}_{sm} - \gamma_{gm}(T_{sm} - T_g) - \Delta H_{rsm} + \gamma_{Rm}(T_{Rm}^4 - T_{sm}^4) \quad (203)$$

Gas species balance:

$$\frac{\partial}{\partial t}(\epsilon_g \rho_g X_{gn}) + \nabla \cdot (\epsilon_g \rho_g X_{gn} \vec{v}_g) = R_{gn} \quad (204)$$

Solids species balance:

$$\frac{\partial}{\partial t}(\epsilon_{sm} \rho_{sm} X_{smn}) + \nabla \cdot (\epsilon_{sm} \rho_{sm} X_{smn} \vec{v}_{sm}) = R_{smn} \quad (205)$$

Gas-solids drag:

$$F_{gm} = \frac{3\epsilon_{sm}\epsilon_g \rho_g}{4 V_{rm}^2 d_{pm}}\left(0.63 + 4.8\sqrt{V_{rm}/Re_m}\right)^2 |\vec{v}_{sm} - \vec{v}_g| \quad (206)$$

$$V_{rm} = 0.5\left(A - 0.06Re_m + \sqrt{(0.06Re_m)^2 + 0.12Re_m(2B - A) + A^2}\right) \quad (207)$$

$$A = \epsilon_g^{4.14} \quad (208)$$

$$B = \begin{cases} 0.8\epsilon_g^{1.28} & if\ \epsilon_g \le 0.85 \\ \epsilon_g^{2.65} & if\ \epsilon_g > 0.85 \end{cases} \quad (209)$$

$$Re_m = \frac{d_{pm}|\vec{v}_{sm} - \vec{v}_g|\rho_g}{\mu_g} \tag{210}$$

Solids-solids drag:

$$F_{slm} = \frac{3(1+e_{lm})(\frac{\pi}{2} + \frac{C_{flm}\pi^2}{8})\epsilon_{sl}\rho_{sl}\epsilon_{sm}\rho_{sm}(d_{pl}+d_{pm})^2 g_{0_{lm}}|\vec{v}_{sl} - \vec{v}_{sm}|}{2\pi(\rho_{sl}d_{pl}^3 + \rho_{sm}d_{pm}^3)} \tag{211}$$

$$g_{0_{lm}} = \frac{1}{\epsilon_g} + \frac{3\left(\sum_{\lambda=1}^{M} \epsilon_{s\lambda}/d_{p\lambda}\right) d_{pl}d_{pm}}{\epsilon_g^2(d_{pl}+d_{pm})} \tag{212}$$

Gas-phase stress:

$$\bar{\bar{\tau}}_g = 2\mu_{gt}\bar{\bar{D}}_g - \frac{2}{3}\mu_{gt}tr(\bar{\bar{D}}_g)\bar{\bar{I}} \tag{213}$$

$$\mu_{gt} = Min(\mu_{gmax}, \mu_g + \mu_e) \tag{214}$$

$$\mu_e = 2l_s^2\epsilon_g\rho_g\sqrt{I_{2Dg}} \tag{215}$$

$$I_{2Dg} = \frac{1}{6}\left((D_{g11}-D_{g22})^2 + (D_{g22}-D_{g33})^2 + (D_{g33}-D_{g11})^2\right) + D_{g12}^2 + D_{g23}^2 + D_{g31}^2 \tag{216}$$

Porous media model:

$$\vec{f}_g = -\frac{\mu_g}{c_1}\vec{v}_g - \frac{c_2}{2}\rho_g|\vec{v}_g|\vec{v}_g \tag{217}$$

Granular stress:

$$\bar{\bar{S}}_{sm} = \begin{cases} -P_{sm}^{p}\bar{\bar{I}} + \bar{\bar{\tau}}_{sm}^{p} & if\ \epsilon_g \le \epsilon_g^* \\ -P_{sm}^{v}\bar{\bar{I}} + \bar{\bar{\tau}}_{sm}^{v} & if\ \epsilon_g > \epsilon_g^* \end{cases} \tag{218}$$

Plastic Regime:

$$P_{sm}^{p} = 10^{25}(\epsilon_m - \epsilon_m^{cp})^{10} \tag{219}$$

$$\bar{\bar{\tau}}_{s1}^{p} = 2\,\mu_{s1}^{p}\bar{\bar{D}} \tag{220}$$

$$\mu_{s1}^{p} = \frac{P_{sm}^{p}\sin\phi}{2\sqrt{I_{2Ds}}} \tag{221}$$

$$I_{2Ds} = \frac{1}{6}\left((D_{s11}-D_{s22})^2 + (D_{s22}-D_{s33})^2 + (D_{s33}-D_{s11})^2\right) + D_{s12}^2 + D_{s23}^2 + D_{s31}^2 \tag{222}$$

Viscous Regime:

$$P_{sm}^{v} = K_{1m}\epsilon_{sm}^2\Theta_m \tag{223}$$

$$\bar{\bar{\tau}}_{sm}^{v} = \lambda_{sm}^{v}\,tr(\bar{\bar{D}}_{sm})\bar{\bar{I}} + 2\mu_{sm}^{v}\bar{\bar{D}}_{sm} \tag{224}$$

$$\lambda_{sm}^{v} = K_{2m}\epsilon_{sm}\sqrt{\Theta_m} \tag{225}$$

$$\mu_{sm}^{v} = K_{3m}\epsilon_{sm}\sqrt{\Theta_m} \tag{226}$$

$$K_{1m} = 2(1+e_{mm})\rho_{sm}g_{0_{mm}} \tag{227}$$

$$K_{2m} = 4d_{pm}\rho_{sm}(1+e_{mm})\epsilon_{sm}g_{0_{mm}}/(3\sqrt{\pi}) - \frac{2}{3}K_{3m} \tag{228}$$

$$K_{3m} = \frac{d_{pm}\rho_{sm}}{2}\left\{\frac{\sqrt{\pi}}{3(3-e_{mm})}[1+0.4(1+e_{mm})(3e_{mm}-1)\epsilon_{sm}g_{0_{mm}}] + \frac{8\epsilon_{sm}g_{0_{mm}}(1+e_{mm})}{5\sqrt{\pi}}\right\} \quad (229)$$

$$g_{0_{mm}} = \frac{1}{\epsilon_g} + \frac{3\left(\sum_{\lambda=1}^{M}\frac{\epsilon_{s\lambda}}{d_{p\lambda}}\right)d_{pm}}{2\epsilon_g^2} \quad (230)$$

Gas-solids heat transfer:

$$\gamma_{gm} = \frac{C_{pg}R_{0m}}{\exp\left(\frac{C_{pg}R_{0m}}{\gamma_{gm}^0}\right)-1} \quad (231)$$

$$\gamma_{gm}^0 = \frac{6\,k_g\,\epsilon_{sm}\,Nu_m}{{d_{pm}}^2}\ , \quad (232)$$

$$Nu_m = (7-10\epsilon_g+5\epsilon_g^2)(1+0.7Re_m^{0.2}Pr^{1/3}) + (1.33-2.4\epsilon_g+1.2\epsilon_g^2)Re_m^{0.7}Pr^{1/3} \quad (233)$$

Gas and solids conduction:

$$\vec{q}_g = -k_g\,\nabla T_g \quad (234)$$

$$\vec{q}_{sm} = -k_{sm}\,\nabla T_{sm} \quad (235)$$

Granular energy equation:

$$\Theta_m = \left\{ \frac{-K_{1m}\epsilon_{sm} tr(\overline{\overline{D}}_{sm}) + \sqrt{K_{1m}^2 tr^2(\overline{\overline{D}}_{sm})\epsilon_{sm}^2 + 4K_{4m}\epsilon_{sm}[K_{2m} tr^2(\overline{\overline{D}}_{sm}) + 2K_{3m} tr(\overline{\overline{D}}^2_{sm})]}}{2\epsilon_{sm} K_{4m}} \right\}^2 \quad \textbf{(236)}$$

$$K_{4m} = \frac{12(1 - e_{mm}^2)\rho_{sm} g_{0mm}}{d_{pm}\sqrt{\pi}} \quad \textbf{(237)}$$

A.2 Nomenclature

a	-	Coefficients on the left hand side of a linear equation set
A	-	Area of control volume faces; m^2
b	-	Right-hand side vector of a linear equation set
c_1	-	Permeability of porous media; m^2
c_2	-	Inertial resistance factor of porous media; m^{-1}
C_{Ds}	-	Single particle drag function
C_{pg}	-	Specific heat of the fluid phase; $J/kg \cdot K$
C_{flm}	-	Coefficient of friction for solids phases l and m
C_{psm}	-	Specific heat of the m^{th} solids phase; $J/kg \cdot K$
d_{pm}	-	Diameter of the particles constituting the m^{th} solids phase; m
$\overline{\overline{D}}_g$	-	Rate of strain tensor, fluid phase; s^{-1}
$\overline{\overline{D}}_{sm}$	-	Rate of strain tensor, solids phase-m; s^{-1}
e_{lm}	-	Coefficient of restitution for the collisions of m^{th} and l^{th} solids phases
f	-	Ratio of cell sizes used in interpolation formulas
$\vec{f}_g$	-	Fluid flow resistance due to porous media; N/m^3
F_{gm}	-	Coefficient for the interphase force between the fluid phase and the m^{th} solids phase; $kg/m^3 \cdot s$
F_{slm}	-	Coefficient for the interphase force between the l^{th} solids phase and the m^{th} solids phase; $kg/m^3 \cdot s$
$\vec{g}$	-	Acceleration due to gravity; m/s^2
$g_{0_{lm}}$	-	Radial distribution function at contact
ΔH_{rg}	-	Heat of reaction in the fluid phase; $J/m^3 \cdot s$
ΔH_{rsm}	-	Heat of reaction in the m^{th} solids phase; $J/m^3 \cdot s$
I_{2Dg}	-	Second invariant of the deviator of the strain rate tensor for gas phase; s^{-2}
I_{2Ds}	-	Second invariant of the deviator of the strain rate tensor for solids phase-1; s^{-2}
k_g	-	Fluid-phase conductivity; $J/m \cdot K \cdot s$
k_{pm}	-	Conductivity of material that constitutes solids phase-m; $J/m \cdot K \cdot s$
k_{sm}	-	Solids phase-m conductivity; $J/m \cdot K \cdot s$
l	-	Index of the l^{th} solids phase; also used as a miscellaneous index

l_s	-	A turbulence length-scale parameter; m
m	-	Index of the m^{th} solids phase. "m=0" indicates fluid phase
M	-	Total number of solids phases
Mw	-	Average molecular weight of gas
n	-	Index of the n^{th} chemical species
N_g	-	Total number of fluid-phase chemical species
N_{sm}	-	Total number of solids phase-m chemical species
Nu_m	-	Nusselt number
P_g	-	Pressure in the fluid phase; Pa
P_{sm}^{p}	-	Pressure in Solids phase-m, plastic regime; Pa
P_{sm}^{v}	-	Pressure in Solids phase-m, viscous regime; Pa
Pr	-	Prandtl number
$\vec{q}_g$	-	Fluid-phase conductive heat flux; $J/m^2 \cdot s$
$\vec{q}_{sm}$	-	Solids-phase-m conductive heat flux; $J/m^2 \cdot s$
R	-	Universal gas constant; $Pa \cdot m^3/kmol \cdot K$
Re_m	-	m^{th} solids phase particle Reynolds number
R_{km}	-	Ratio of solids to fluid conductivity
R_{ml}	-	Rate of transfer of mass from m^{th} phase to l^{th} phase. l or m = 0 indicates fluid phase; $kg/m^3 \cdot s$
R_{gn}	-	Rate of production of the n^{th} chemical species in the fluid phase; $kg/m^3 \cdot s$
R_{smn}	-	Rate of production of the n^{th} chemical species in the m^{th} solids phase; $kg/m^3 \cdot s$
$\bar{\bar{S}}_g$	-	Fluid-phase stress tensor; Pa
$\bar{\bar{S}}_{sm}$	-	Solids phase-m stress tensor; Pa
t	-	Time; s
T_g	-	Thermodynamic temperature of the fluid phase; K
T_{sm}	-	Thermodynamic temperature of the solids phase-m; K
T_{Rg}	-	Gas phase radiation temperature; K
T_{Rm}	-	Solids phase-m radiation temperature; K
$\vec{v}_g$	-	Fluid-phase velocity vector; m/s

$\vec{v}_{sm}$ - m^{th} solids-phase velocity vector; m/s

ΔV - Control volume size; m^3

V_{rm} - The ratio of the terminal velocity of a group of particles to that of an isolated particle

X_{gn} - Mass fraction of the n^{th} chemical species in the fluid phase

X_{smn} - Mass fraction of the n^{th} chemical species in the m^{th} solids phase

GREEK LETTERS

γ_{gm} - Fluid-solids heat transfer coefficient corrected for interphase mass transfer; $J/m^3 \cdot K \cdot s$

γ^0_{gm} - Fluid-solids heat transfer coefficient not corrected for interphase mass transfer; $J/m^3 \cdot K \cdot s$

γ_{Rg} - Fluid-phase radiative heat transfer coefficient; $J/m^3 \cdot K^4 \cdot s$

γ_{Rm} - Solids-phase-m radiative heat transfer coefficient; $J/m^3 \cdot K^4 \cdot s$

γ_{Θ_m} - Granular energy dissipation due to inelastic collisions; $J/m^3 \cdot s$

Γ - A general diffusivity coefficient

ϵ_g - Volume fraction of the fluid phase (void fraction)

ϵ^{cp}_{sm} - Packed-bed (maximum) solids volume fraction

ϵ_{sm} - Volume fraction of the m^{th} solids phase

η - Function of restitution coefficient

Θ_m - Granular temperature of phase-m; m^2/s^2

λ_{rm} - Solids conductivity function

λ^v_{sm} - Second coefficient of solids viscosity, viscous regime; $kg/m \cdot s$

μ_e - Eddy viscosity of the fluid phase; $kg/m \cdot s$

μ_g - Molecular viscosity of the fluid phase; $kg/m \cdot s$

μ_{gmax} - Maximum value of the turbulent viscosity of the fluid phase; $kg/m \cdot s$

μ_{gt} - Turbulent viscosity of the fluid phase; $kg/m \cdot s$

μ^p_{sl} - Solids viscosity, plastic regime; $kg/m \cdot s$

μ^v_{sl} - Solids viscosity, viscous regime; $kg/m \cdot s$

ξ_{ml} - $\xi_{ml} = 1$ if $R_{ml} < 0$; else $\xi_{ml} = 0$.

ξ - Convection factor defined in Algorithm 2.2

ρ_g - Microscopic (material) density of the fluid phase; kg/m^3

ρ_g' - Macroscopic (effective) density of the fluid phase; kg/m^3

ρ_{sm} - Microscopic (material) density of the mth solids phase; kg/m^3

ρ_{sm}' - Macroscopic (bulk) density of the mth solids phase; kg/m^3

$\bar{\bar{\tau}}_g$ - Fluid phase deviatoric stress tensor; Pa

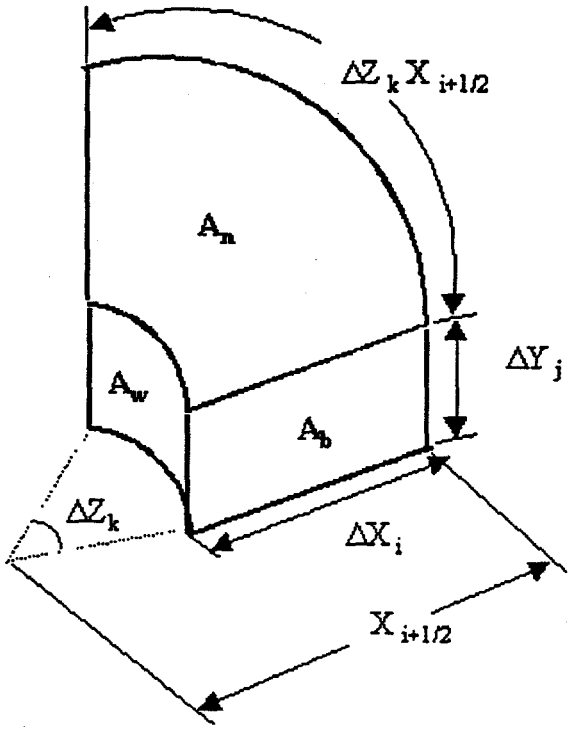
$\bar{\bar{\tau}}_{sm}^{p}$ - Solids phase-m deviatoric stress tensor, plastic regime; Pa

$\bar{\bar{\tau}}_{sm}^{v}$ - Solids phase-m deviatoric stress tensor, viscous regime; Pa

ϕ - Angle of internal friction, also used as general scalar

ϕ_k - Contact area fraction in solids conductivity model

Appendix B: Definition of Areas and Volume



Note that in Cartesian coordinates X_i, $X_{i+1/2}$ etc. are all equal to 1.

$$A_e = (\Delta Y_j)\ \left(X_{i+1/2}\ \Delta Z_k\right)$$

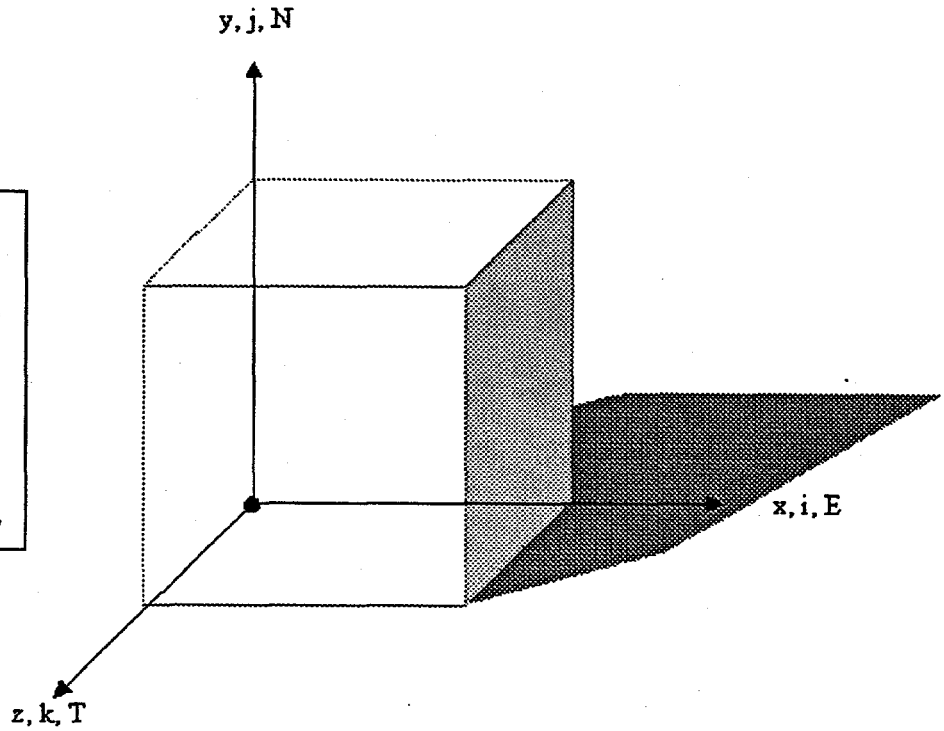
$$A_w = (\Delta Y_j)\ \left(X_{i-1/2}\ \Delta Z_k\right)$$

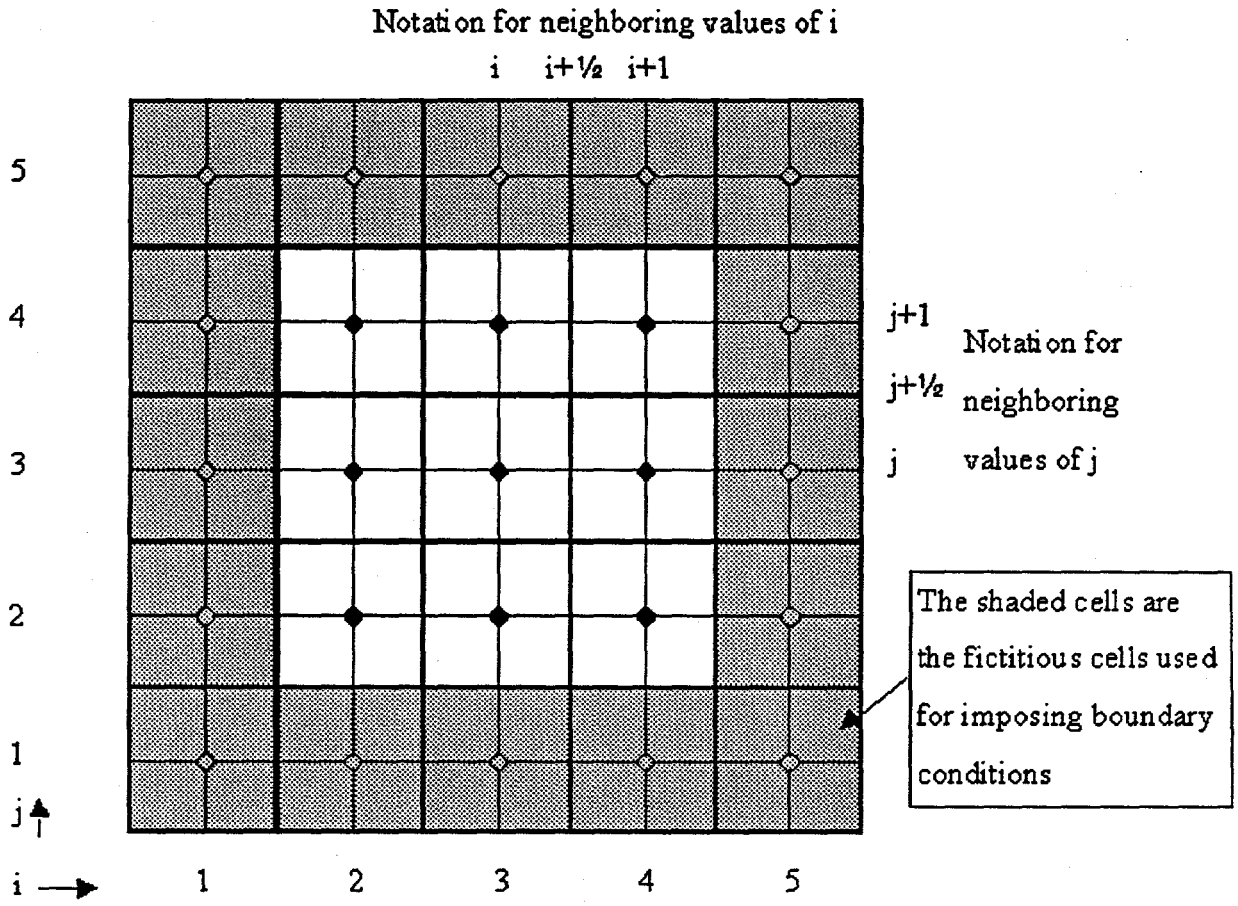
$$A_s = A_n = (\Delta X_i)\ \left(X_i\ \Delta Z_k\right)$$

$$A_t = A_b = (\Delta X_i)\ \ (\Delta Y_j)$$

$$\Delta V = (\Delta X_i)\ \ (\Delta Y_j)\ \left(X_i\ \Delta Z_k\right)$$

Coordinates			
Cartesian	x	y	z
Cylindrical	r	y	θ
Cell Index	i	j	k
+ve direction	E	N	T
-ve direction	W	S	B
Velocity	u	v	w




Notation for neighboring values of i
i
i+½
i+1
5
4
3
2
1
j
i
1
2
3
4
5
j+1
j+½
j
Notation for neighboring values of j
The shaded cells are the fictitious cells used for imposing boundary conditions

Appendix C: Notes on higher order discretization

C.1 Definitions

We will first define certain quantities that will be used in the derivations. Figure C.1 gives the notation used in the definitions and derivations.

$$\tilde{\phi}_C \equiv \frac{\phi_C - \phi_U}{\phi_D - \phi_U} \qquad \text{(239)}$$

Flow

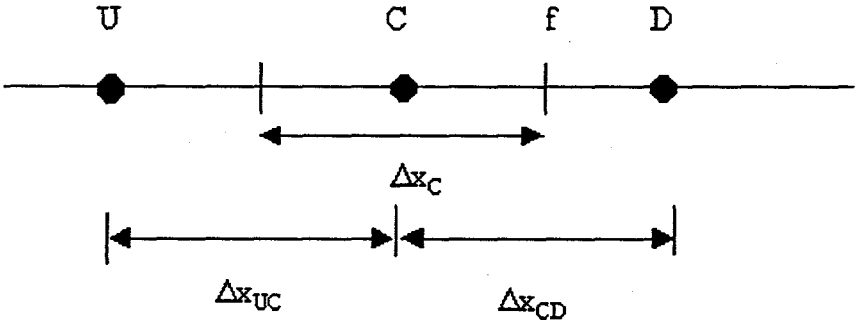


Figure C.1. Notation based on flow direction

$$\theta \equiv \frac{\tilde{\phi}_C}{1 - \tilde{\phi}_C} \qquad \text{(240)}$$

$$dwf \equiv \frac{\tilde{\phi}_f - \tilde{\phi}_C}{1 - \tilde{\phi}_C} \qquad \text{(241)}$$

$$\therefore \quad \frac{\phi_C - \phi_U}{\phi_D - \phi_C} = \frac{\tilde{\phi}_C}{1 - \tilde{\phi}_C} = \theta \qquad \text{(242)}$$

C.2 Central Differencing

$$\phi_f = \frac{1}{2}(\phi_C + \phi_D) \qquad \text{(243)}$$

$$dwf = \frac{\frac{1}{2}(\phi_C + \phi_D) - \phi_C}{(\phi_D - \phi_C)} = \frac{1}{2} \tag{244}$$

C.3 Second Order Upwinding

$$\phi_f = \frac{1}{2}(\phi_C + \phi_D) - \frac{1}{2}(\phi_D - 2\,\phi_C + \phi_U) \tag{245}$$

$$\begin{aligned} dwf &= \left\{\frac{1}{2}(\phi_C + \phi_D) - \frac{1}{2}(\phi_D - 2\,\phi_C + \phi_U) - \phi_C\right\} \Big/ (\phi_D - \phi_C) \\ &= \left\{\frac{1}{2}(\phi_D - \phi_C) - \frac{1}{2}(\phi_D - \phi_C) - \frac{1}{2}(\phi_U - \phi_C)\right\} \Big/ (\phi_D - \phi_C) \\ &= -\frac{1}{2}\frac{\phi_U - \phi_C}{\phi_D - \phi_C} = \frac{\theta}{2} \end{aligned} \tag{246}$$

C.4 QUICK (Leonard 1979)

$$\phi_f = \frac{1}{2}(\phi_C + \phi_D) - \frac{\Delta x_{CD}^2}{8}\frac{1}{\Delta x_C}\left[\frac{1}{\Delta x_{CD}}(\phi_D - \phi_C) - \frac{1}{\Delta x_{UC}}(\phi_C - \phi_U)\right] \tag{247}$$

On a uniform grid the above formula reduces to

$$\phi_f = \frac{1}{2}(\phi_C + \phi_D) - \frac{1}{8}(\phi_D - 2\,\phi_C + \phi_U) \tag{248}$$

$$dwf = \frac{\phi_f - \phi_C}{\phi_D - \phi_D}$$

$$\begin{aligned} &= \left\{\frac{1}{2}(\phi_C + \phi_D) - \frac{1}{8}\left[\frac{\Delta x_{CD}}{\Delta x_C}(\phi_D - \phi_C) - \frac{\Delta x_{CD}^2}{\Delta x_C\,\Delta x_{UC}}(\phi_C - \phi_U)\right] - \phi_C\right\} \Big/ (\phi_D - \phi_C) \\ &= \frac{1}{2} - \frac{1}{8}\left[\frac{\Delta x_{CD}}{\Delta x_C} - \frac{\Delta x_{CD}^2}{\Delta x_C \Delta x_{UC}}\frac{\phi_C - \phi_U}{\phi_D - \phi_C}\right] \end{aligned} \tag{249}$$

$$dwf = \frac{1}{2} - \frac{1}{8}\left[\frac{\Delta x_{CD}}{\Delta x_C} - \frac{\Delta x_{CD}^2}{\Delta x_C\,\Delta x_{UC}}\,\theta\right] \tag{250}$$

For a uniform grid the above formula reduces to

$$dwf = \frac{1}{2} - \frac{1}{8}(1 - \theta) = \frac{3}{8} + \frac{\theta}{8} \tag{251}$$

C.5 SMART (Gaskell and Lau 1988)

$$\tilde{\phi}_f = \begin{cases} \tilde{\phi}_C & \tilde{\phi}_C \notin [0,\ 1] \\ 3\tilde{\phi}_C & \tilde{\phi}_C \in \left[0, \frac{1}{6}\right) \\ 1 & \tilde{\phi}_C \in \left(\frac{5}{6},\ 1\right] \\ \frac{3}{8}\left(2\,\tilde{\phi}_C + 1\right) & \tilde{\phi}_C \in \left[\frac{1}{6},\ \frac{5}{6}\right] \end{cases} \tag{252}$$

Then

$$dwf = \begin{cases} 0 & \tilde{\phi}_c \notin [0,\ 1] \\ 2\theta & \tilde{\phi}_c \in \left[0, \frac{1}{6}\right] \\ \frac{3}{8} + \frac{1}{8}\,\theta & \tilde{\phi}_c \in \left[\frac{1}{6},\ \frac{5}{6}\right] \quad (\textit{same as QUICK}) \\ 1 & \tilde{\phi}_c \in \left(\frac{5}{6},\ 1\right] \end{cases} \tag{253}$$

which can be compactly written as

$$dwf = \frac{1}{2} \max\left[0,\ \min\left(4\,\theta,\ 0.75 + 0.25\,\theta,\ 2\right)\right] \tag{254}$$

C.6 Properties of Discretization Schemes

The following table taken from Gaskell and Lau (1988) summarizes properties of the several discretization schemes. The second column gives the truncation error term obtained from

Table C.I. Properties of discretization schemes

Discretization Scheme	Leading truncation error term	Convective stability	Boundedness		Critical Peclet number
			Interpolative	Computed	
First order Upwind	$\frac{u\Delta x}{2}\phi_f'$	$-\frac{u}{\Delta x}$ (*stable*)	Yes	Yes	∞
central differencing	$\frac{u\Delta x^2}{4}\phi_f''$	0 (*neutral*)	Yes	No	2
Second-order upwind	$\frac{3u\Delta x^2}{8}\phi_f'$	$-\frac{3u}{2\Delta x}$ (*stable*)	No	No	∞
QUICK	$\frac{u\Delta x^3}{16}\phi_f'''$	$-\frac{3u}{8\Delta x}$ (*stable*)	No	No	$\frac{8}{3}$

a Taylor series expansion. The error term is meaningful only for Fourier components of small wave numbers ($k_{min} = 2\pi / L$), where L is the length of the domain. No scheme can yield an accurate representation of components of high wave numbers, ($k_{max} = \pi / \Delta x$).

The third column of the table gives the convective stability, which is the rate of change of convective influx into a CV with respect to the value at the CV center ϕ_C (Leonard 1979). For stability this must be less than zero, which implies that schemes must have an upwind bias.

The fourth column shows the interpolative boundedness of the schemes. A scheme is bounded if the calculated face value ϕ_f lies in the range $[\phi_U, \phi_D]$, when the variation in ϕ is monotonic. If the interpolative boundedness criterion is not satisfied, the scheme allows the calculation of convective fluxes that exceed physically possible values.

When the variation in ϕ is monotonic, we want the computed solution not to have any spurious extrema. This property, called the computed boundedness of the scheme, is shown in the fifth column. Interpolative boundedness is a necessary, but not sufficient, condition for ensuring computed boundedness.

Diffusion has a stabilizing influence that can counterbalance the lack of convective stability. This stabilizing influence, however, diminishes as the cell Peclet number increases, and beyond a critical value (sixth column) the influence of diffusion is insufficient to prevent spurious oscillations in the solution.

Appendix D: Methods for Multiphase Equations

This appendix describes numerical methods for handling more than two phases. These have not been tested in MFIX and may be used in a future version of MFIX.

D.1 Partial elimination

The discretized momentum equation for phase l is

$$(a_l)_p (\phi_l)_p = \sum_{nb} (a_l)_{nb} (\phi_l)_{nb} + b_l + \Delta V \sum_{l'}^{M} F_{ll'} \left[(\phi_{l'})_p - (\phi_l) \right] \tag{255}$$

Solve for ϕ_l to get

$$\begin{aligned} (\phi_l)_p &= \frac{\sum_{nb} (a_l)_{nb} (\phi_l)_{nb} + b_l + \Delta V \sum_{l'} F_{ll'} (\phi_{l'})_p}{(a_l)_p + \Delta V \sum_{l'} F_{ll'}} \\ &= \frac{\sum_{nb} (a_l)_{nb} (\phi_l)_{nb} + b_l + \Delta V \sum_{l' \neq m} F_{ll'} (\phi_{l'})_p + \Delta V F_{lm} (\phi_m)_p}{(a_l)_p + \Delta V \sum_{l' \neq m} F_{ll'} + \Delta V F_{lm}} \end{aligned} \tag{256}$$

where we have purposely taken ϕ_m out of the summation sign. Substituting the above formula in Equation D.1 (with the subscripts l changed to m) we get

$$\begin{aligned} (a_m)_p (\phi_m)_p &= \sum_{nb} (a_m)_{nb} (\phi_m)_{nb} + b_m \\ &+ \Delta V \sum_{l} F_{lm} \left[\frac{\sum_{nb} (a_l)_{nb} (\phi_l)_{nb} + b_l + \Delta V \sum_{l' \neq m} F_{ll'} (\phi_{l'})_p + \Delta V F_{lm} (\phi_m)_p}{(a_l)_p + \Delta V \sum_{l' \neq m} F_{ll'} + \Delta V F_{lm}} - (\phi_m) \right] \end{aligned} \tag{257}$$

The last term can be simplified as

$$\begin{aligned} &\Delta V \sum_{l} F_{lm} \left[\frac{\sum_{nb} (a_l)_{nb} (\phi_l)_{nb} + b_l + \Delta V \sum_{l' \neq m} F_{ll'} (\phi_{l'})_p}{(a_l)_p + \Delta V \sum_{l' \neq m} F_{ll'} + \Delta V F_{lm}} \right. \\ &\left. + \frac{\Delta V F_{lm} (\phi_m)_p - \left[(a_l)_p + \Delta V \sum_{l' \neq m} F_{ll'} + \Delta V F_{lm} \right] (\phi_m)_p}{(a_l)_p + \Delta V \sum_{l' \neq m} F_{ll'} + \Delta V F_{lm}} \right] \end{aligned} \tag{258}$$

Thus, we get the partially eliminated equation

$$\left\{ \left(a_m\right)_p + \Delta V \sum_{\ell} \left(F_{\ell m} \frac{\left(a_\ell\right)_p + \Delta V \sum_{\ell' \neq m} F_{\ell\ell'}}{\left(a_\ell\right)_p + \Delta V \sum_{\ell' \neq m} F_{\ell\ell'} + \Delta V F_{\ell m}} \right) \right\} \left(\phi_m\right)_p = \sum_{nb} \left(a_m\right)_{nb} \left(\phi_m\right)_{nb}$$

$$+ b_m + \Delta V \sum_{\ell} \left(F_{\ell m} \frac{\sum_{nb} \left(a_\ell\right)_{nb} \left(\phi_\ell\right)_{nb} + b_\ell + \Delta V \sum_{\ell' \neq m} F_{\ell\ell'} \left(\phi'_\ell\right)_p}{\left(a_\ell\right)_p + \Delta V \sum_{\ell' \neq m} F_{\ell\ell'} + \Delta V F_{\ell m}} \right) \tag{259}$$

Note that $\left(\phi'_\ell\right)$ still appears in the last term on the right-hand side, which will be treated as a source term. Therefore, this method may not be as effective as an exact partial elimination for stabilizing the computations. The method will be faster than the exact method, however.

D.2. Pressure Correction Equation

An approximate pressure correction formula for multiple phases can be derived as follows:

$$a_{mp}\, u'_{mp} = -A_p \epsilon^*_{mp} \left((P_g)'_E - (P_g)'_W \right) + \Delta V \sum_{\ell} F_{\ell m} \left(u'_{\ell p} - u'_{mp} \right) \tag{261}$$

Solving for the 'ℓ'th component we get

$$\left(a_{\ell p} + \Delta V \sum_{\ell'} F_{\ell\ell'} \right) u'_{\ell p} = -A_p \epsilon^*_{\ell p} \left((P_g)'_E - (P_g)'_W \right) + \Delta V \sum_{\ell'} F_{\ell\ell'}\, u'_{\ell p} \tag{262}$$

Therefore

$$u'_{\ell p} = \frac{-A_p \epsilon^*_{\ell p} \left((P_g)'_E - (P_g)'_W \right) + \Delta V \sum_{\ell'} F_{\ell\ell'}\, u'_{\ell' p}}{a_{\ell p} + \Delta V \sum_{\ell'} F_{\ell\ell'}} \tag{263}$$

Substituting this in Equation 18 we get

$$
\begin{aligned}
{}'_p &= -A_p \epsilon^*_{mp} \left((P_g)'_E - (P_g)'_W\right) + \Delta V \sum_{\ell} \left\{ F_{\ell m} \left[\frac{-A_p \epsilon^*_{\ell p} \left((P_g)'_E - (P_g)'_W\right) + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} u'_{\ell' p} + \Delta V F_{\ell m} u'_{mp}}{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} + \Delta V F_{\ell m}} - \right.\right. \\
&= -A_p \epsilon^*_{mp} \left((P_g)'_E - (P_g)'_W\right) + \Delta V \sum F_{\ell m} \left[\frac{-A_p \epsilon^*_{\ell p} \left((P_g)'_E - (P_g)'_W\right) + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} u'_{\ell' p}}{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} + \Delta V F_{\ell m}} \right. \qquad (264) \\
&\left.\left. + \frac{\Delta V F_{\ell m} u'_{mp} - \left(a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} + \Delta V F_{\ell m}\right) u'_{mp}}{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} + \Delta V F_{\ell m}} \right] \right.
\end{aligned}
$$

Solving for the velocity component we get

$$
\left\{ a_{mp} + \Delta V \sum_{\ell} F_{\ell m} \left[\frac{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'}}{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} + \Delta V F_{\ell m}} \right] \right\} u'_{mp}
$$

$$
= -A_p \epsilon^*_{mp} \left((P_g)'_E - (P_g)'_W\right) + \Delta V \sum_{\ell} F_{\ell m} \frac{\left[-A_p \epsilon^*_{\ell p} \left((P_g)'_E - (P_g)'_W\right) + \Delta V \sum_{\ell' \neq m} \left(F_{\ell \ell'} u_{\ell' p}\right) \right]}{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} + \Delta V F_{\ell m}} \qquad (265)
$$

Dropping the term $\Delta V \sum_{\ell' \neq m} \left(F_{\ell \ell'} u_{\ell' p}\right)$ we get

$$
\begin{aligned}
u'_{mp} &= \frac{-A_p \left\{ \epsilon^*_{mp} + \Delta V \sum_{\ell} \frac{F_{\ell m} \epsilon^*_{\ell p}}{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} + \Delta V F_{\ell m}} \right\}}{a_{mp} + \Delta V \sum_{\ell} F_{\ell m} \left[\frac{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'}}{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} + \Delta V F_{\ell m}} \right]} \left((P_g)'_E - (P_g)'_W\right) \qquad (266) \\
&= - d_{mp} \left((P_g)'_E - (P_g)'_W\right)
\end{aligned}
$$

where

$$d_{mp} = \frac{-A_p \left\{ \epsilon^*_{mp} + \Delta V \sum_{\ell} \frac{F_{\ell m}\, \epsilon^*_{\ell p}}{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell} + \Delta V\, F_{\ell m}} \right\}}{a_{mp} + \Delta V \sum_{\ell} F_{\ell m} \left[\frac{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'}}{a_{\ell p} + \Delta V \sum_{\ell' \neq m} F_{\ell \ell'} + \Delta V\, F_{\ell m}} \right]} \tag{267}$$

Therefore,

$$u_{mp} = u^*_{mp} + u'_{mp} = u^*_{mp} - d_{mp} \left((P_g)'_E - (P_g)'_W \right) \tag{268}$$