Quantitative Analysis of Accuracy of Voidage Computations in CFD-DEM Simulations

H. A. Khawaja, S. A. Scott, M. S. Virk, and M. Moatamedi

Reprinted from

The Journal of
Computational Multiphase Flows

Volume 4 · Number 2 · 2012

Multi-Science Publishing

Quantitative Analysis of Accuracy of Voidage Computations in CFD-DEM Simulations

[1*]H. A. Khawaja, [2]S. A. Scott, [3]M. S. Virk, and [4]M. Moatamedi
[1]PhD Student, Department of Engineering, University of Cambridge, UK
[2]Lecturer, Department of Engineering, University of Cambridge, UK
[3]Associate Professor, High North Technology Centre, Narvik University College, Norway
[4]Professor, High North Technology Centre, Narvik University College, Norway

Received: 30 November 2011, Accepted: 18 May 2012

Abstract
CFD-DEM (Computational Fluid Dynamics – Discrete Element Modelling) is a two-phase flow numerical modelling technique, where the Eulerian method is used for the fluid and the Lagrangian method for the particles. The two phases are coupled by a fluid-particle interaction force (i.e. drag force) which is computed using a correlation. In a two-phase flow, one critical parameter is the voidage (or void fraction), which is defined as the ratio of the volume occupied by the fluid to the total volume. In a CFD-DEM simulation the local voidage is computed by calculating the volume of particles in a given fluid cell. For spherical particles, this computation is difficult when a particle is on the boundary of fluid cells. In this case, it is usual to compute the volume of a particle in a fluid cell approximately. One such approximation divides the volume of a particle into each cell in the same ratio as an equivalent cube of width equal to the particle diameter. Whilst this approach is computationally straight forward, the approximation introduces an error in the voidage computation. Here we estimate the error by comparing the approximate volume calculation with an exact (numerical) computation of the volume of a particle in a fluid cell. The results show that the error varies with the position of the particle relative to the cell boundary. A new approach is suggested which limits the error to less than 2.5 %, without significantly increasing the computational complexity.

1. INTRODUCTION

Two-phase granular flow is involved in many industrial applications: fluidised bed reactors, pneumatic conveying, etc. With the advancements in computational resources, researchers have developed various numerical models to simulate two-phase flow. The limitation and accuracy of the available two- phase flow models is discussed by Hoef et al. (2008). Table 1 gives an overview of the existing two-phase flow models, with accuracy increasing towards the bottom of the table, at the expense of computational effort.

Table 1: Two phase flow models [Hoef et al. (2008)]

Model	Fluid	Granular Particles	Fluid-particle interaction
Discrete Bubble Model (DBM)	Lagrangian	Eulerian	Drag closure on bubbles
Fluid-fluid Model (CFD)	Eulerian	Eulerian	Gas-mixture drag closure
Fluid Dynamics –Discrete Element Model (CFD-DEM)	Eulerian	Lagrangian	Gas-particle drag closure

*Corresponding Author:
Email address: hassan.abbas.khawaja@gmail.com

In CFD-DEM the particles are modelled as discrete entities, and their equations of motion solved (i.e. a Lagrangian approach), whilst the fluid flow is modelled using volume averaged fluid equations. This can be contrasted to the fluid-fluid model, where the particle phase is also treated as pseudo fluid. Volume averaged equations [Anderson and Jackson (1967), Crowe et al. (1998)] are used since it is impractical to solve the Navier-Stokes and continuity equation around every particle. One form of the volume averaged fluid equations are given below, i.e. the equations of momentum and continuity (eq. (1)-(2)). Here the equations are closed by using the ideal gas equation (Eqn.(3)) and a drag force correlation (Eqn.(6)),

$$\frac{\partial(\rho\epsilon\vec{u})}{\partial t} + \nabla.\left(\vec{u}(\rho\epsilon\vec{u})\right) + \epsilon\nabla P - \vec{F}_{drag} - \mu\nabla^2\epsilon\vec{u} - \rho\epsilon\vec{g} = 0 \tag{1}$$

$$\frac{\partial(\rho\epsilon)}{\partial t} + \nabla.(\rho\epsilon\vec{u}) = 0 \tag{2}$$

$$p = \rho RT \tag{3}$$

where ρ is the density of fluid, $\vec{u}$ is the velocity vector, ϵ is the voidage, t is time, R is the gas constant, $\vec{g}$ is the gravity constant and $\vec{F}_{drag}$ is the fluid-particle interaction force. These equations can be solved using standard techniques from computational fluid mechanics, with an extra term added to account for the drag force term and the voidage being derived from the known positions of the particles.

Most DEM simulations will assume (for the sake of simplicity) that the particles are perfect spheres, though it should be noted that simulations with non-spherical particles are possible [Zhong et al. (2009)]. Hard sphere models [Crowe et al. (1998)] track the particles between collisions, and assume that impact is instantaneous without modelling the contact forces directly, whilst soft-sphere models [e.g. Tsuji et al. (1992), Thronton and Kafui (2003)] use the overlap between particles to calculate the contact forces. In the latter case, the volume occupied by the particles can change, both in reality and in the simulation, as the particles overlap. However, the deformation at the contact is usually small if the particles are sufficiently stiff. Thus, in practice, the deformation is ignored when computing the voidage and there is no difference in the voidage computations between the soft or hard sphere approaches.

In addition to the importance of evaluating the voidage accurately for the volume averaged fluid equations, the voidage also plays a role in the drag force. One possible way to simulate the fluid-solid interaction in a two phase solid gas medium is to solve the equations for fluid dynamics around each particle by specifying no slip condition at a particle fluid interface; the interaction force then found by integration of pressure and shear forces over the particle surface. This method is good, but, as noted previously, too expensive for a large number of particles (e.g. above 100). A more feasible method is to use an empirical correlation to compute the drag on each particle. One example of such a drag correlation is given by Beetstra et al. (2007) and used by Müller et al. (2008), Khawaja et al. (2011), Khawaja and Scott (2011) as shown in Eqn. (4),

$$\vec{f}_p = \frac{V_p\beta_{Beetstra}}{1-\epsilon}\left(\vec{u} - \vec{v}_p\right). \tag{4}$$

Here, $\vec{f_p}$ is the drag force on a single particle, Vp is the volume of a particle, $\vec{u}$ is the fluid (volume averaged) velocity vector, $\vec{v_p}$ is the particle velocity, ϵ is the voidage, and $\beta_{Beetstra}$ is computed using Eqn.(5).

$$\beta_{Beetstra} = A\frac{\mu}{d_p^2}\frac{(1-\epsilon)^2}{\epsilon} + B\frac{\mu}{d_p^2}(1-\epsilon)Re \tag{5}$$

where

$$A = 180 + \frac{18\epsilon^4}{1-\epsilon}(1 + 1.5\sqrt{(1-\epsilon)}),$$

$$B = \frac{0.31\left(\epsilon^{-1} + 3(1-\epsilon)\epsilon + 8.4Re^{-0.343}\right)}{1+10^{3(1-\epsilon)}Re^{2\epsilon-2.5}},$$

$$Re = \frac{\epsilon d_p|\vec{u}-\vec{v}_p|\rho_f}{\mu}.$$

The force on the fluid in the cell is then the sum of drag forces on each particle in the cell as shown in Eqn.(6),

$$\vec{F}_{drag} = \frac{1}{V_{cell}} \sum_{n=1}^{n=N_p} \vec{f}_p \tag{6}$$

where V_{cell} is the cell volume and N_p is the number of particles in the cell. It should be noted that other correlations can also be used. Müller et al. (2008), for example, compared the effect of using other drag correlations [Ergun et al. (1952), Di Felice (1994)] on the granular temperature. One thing all these correlations have in common is that drag force depends on local value of voidage. Another point to note is that the summation in Eqn. (6) is analogous to the summation required for the voidage calculation. Thus, any improved method of distributing the volume of particles between cells, is also of relevance to the fluid-particle interaction force calculation.

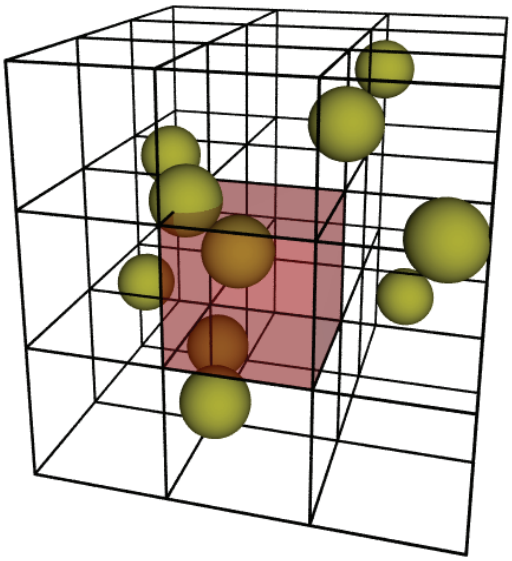

Figure 1: CFD-DEM fluid cells and spherical particles. To calculate the void fraction in the highlighted cell, the volume of all the particles and sections of particles in the cell must be computed: here, the cell contains one complete particle and a wedge shaped section of a second particle.

Finally, it should be noted, that volume averaging requires that there be a significant number of particles in a fluid cell, so that edge effects, where the particle straddles the boundary of a fluid cell, should not be important. In practice, the drag force calculation can be quite sensitive to local

voidage, and edge effects thus have a noticeable effect. Regardless of how accurately the voidage in a fluid cell is calculated, it is desirable to ensure that the voidage in adjacent cells varies smoothly as a particle crosses from one cell to the adjacent cell. For example, if the entire volume of a particle is assigned to the cell where the particle centre is located, there is a sudden jump in voidage when the particle crosses from one cell to another. This leads to unrealistic, high frequency noise in the fluid variables.

2. VOIDAGE COMPUTATION

In a DEM simulation, the voidage can be taken as the ratio of the volume occupied by the fluid in each cell, to the total volume of the fluid cell, i.e.

$$\epsilon = \frac{(volume\ of\ cell - volume\ of\ particles\ in\ the\ cell)}{volume\ of\ cell} \tag{7}$$

where, for the case of rectangular cuboid cells, the volume of the fluid cell is given by

$$volume\ of\ cell = length \times width \times height \tag{8}$$

The volume of the particles in the cell can then be computed by

$$volume\ of\ particles\ in\ cell = \sum_{n=1}^{n=N_p} \left(\frac{4}{3}\pi r^3 \times \phi_s\right) \tag{9}$$

where r is the radius of a particle and ϕ_s is the volume ratio whose value varies from 0 to 1 depending upon the position of the particle. A value of $\phi_s = 0$ means that the particle is completely outside the fluid cell; whilst $\phi_s = 1$ means that the particle is completely inside. An intermediate value indicates that the particle is at a boundary where its volume is divided between more than one cell, with ϕ_s equal to the volume of the particle in the cell divided by the total volume of the particle. In practice, the summation in Eqn. (9) can be restricted to only those particles which could have a contribution to the cell in question, rather than all particles in the system. However, exact calculation of ϕ_s is computationally expensive and an approximation is often used. Part of the difficulty arises from the fact that the fluid domain is divided in Cartesian coordinates, where as the particles are spheres. Particles can be in more than one fluid cell, as shown in Figure 2, where Figure 2(a) shows the particle cut in two by a plane normal to the y-direction, in (b) the particle is divided into four volumes by planes normal to the z and y directions, and in (c) the particle is divided into eight volumes since it is at the corner of a fluid cell.

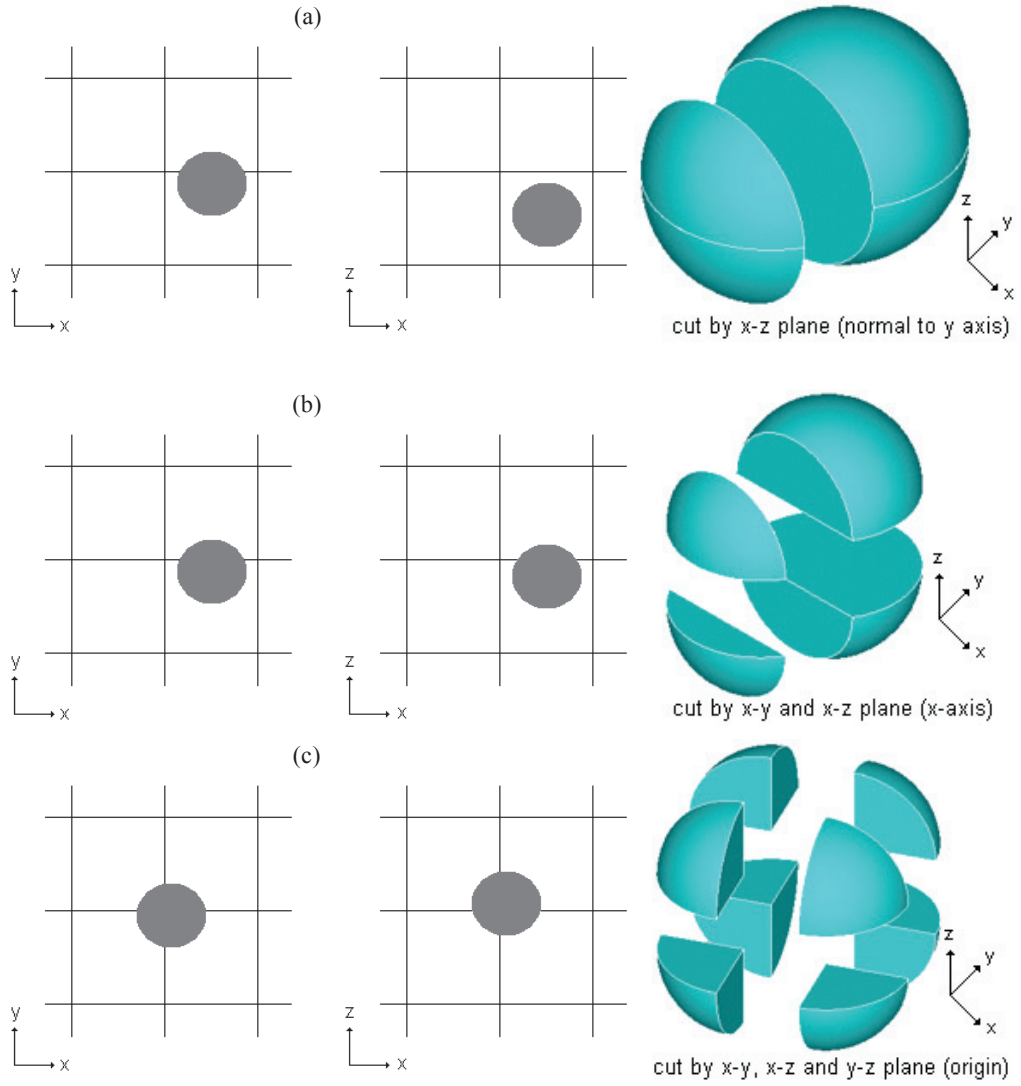

Figure 2: Particle at boundaries; (a) particle divided by a plane (b) particle divided by an edge (c) particle divided at a corner; exploded 3D views of particle for each case

For particles which are on fluid cell boundaries, eight difference cases are possible:

1. Particle divided by a plane (two volumes); given in Figure 2(a)
2. Particle divided by two planes (three volumes)
3. Particle divided by three planes (four volumes); given in Figure 4 (below)
4. Particle divided by an edge (four volumes); given in Figure 2(b)
5. Particle divided by an edge and a plane (five volumes)
6. Particle divided by two edges (six volumes)
7. Particle divided by three edges but not touching the corner (seven volumes)
8. Particle divided by the corner (eight volumes); given in Figure 2(c)

Cases 1 to 5 can be treated analytically, e.g. where an edge or planes slice the particle to produce a spherical cap or a wedge shaped volume as shown in Figure 2 (a) and (b). The other cases, however, require numerical integration. In general, it is possible to write the integral for the volume of the section of sphere contained within a cube as that of a suitable area with respect to a distance, i.e.

$$\iiint_{V\ in\ Cube} dV = \int A(s)\, ds \tag{10}$$

For example, to calculate the volume of the wedge section of the sphere in the cubic cell shown in Figure 3, the volume integral can be written as

$$V = \int_0^{z_i} A(h)dh \tag{11}$$

where, taking the corner of the cube as the origin, $A(h)$ is the area of the sector of a circle, of radius $R_s(h)$, formed by the intersection of the sphere and a plane at $z = h$ with normal in the z direction, minus the area formed by the triangles from $\overrightarrow{OPS}$ and $\overrightarrow{OPQ}$; i.e,

$$V = \int_0^{z_i} \frac{\theta(h)R_s^2(h)}{2} - \frac{X_i(h)[-Y_c]}{2} - \frac{Y_i(h)[-X_c]}{2} dh \tag{12}$$

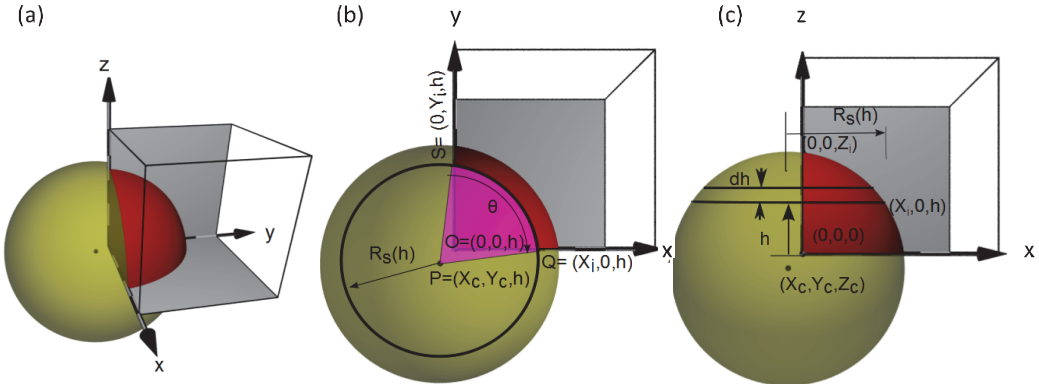

Figure 3: Calculation of the volume of the section of a sphere contained within a cubic volume ; (a) shows the part of volume in a cubic fluid cell, (b) and (c) show the projections looking down and from the side, respectively.

The angle $\theta(h)$, radius $R_s(h)$ and the intercept of the circle formed by the intersection of the sphere and a plane at $z = h$ with the x and y axis, $X_i(h)$ and $Y_i(h)$, are only functions of the distance. Thus, Eqn.(12) can be numerically integrated: here Simpson's method was used [Süli and Mayers (2003)].

One method which has been used previously to approximate the fraction of the volume of a particle falling in a fluid cell (i.e. ϕ_s) is to set it equal to that for a cube which circumscribes the particle, denoted here by ϕ_c (e.g. Figure 3(a)). When a particle is on the boundary of fluid cells (e.g. as in Figure 4 (b), where the particle is cut by three planes), the volume of a particle falling in to a cell is taken to be the fraction of the circumscribing cube's volume (which falls in the cell, ϕ_c multiplied with the volume of the spherical particle). Computing the approximate volume ratio, ϕ_c, using the circumscribing cube is straightforward and computationally efficient. However, this can lead to significant errors, for example, for the case shown in Figures 4 (b) and (c) the particle is cut into four pieces, whilst the cube is cut into eight pieces.

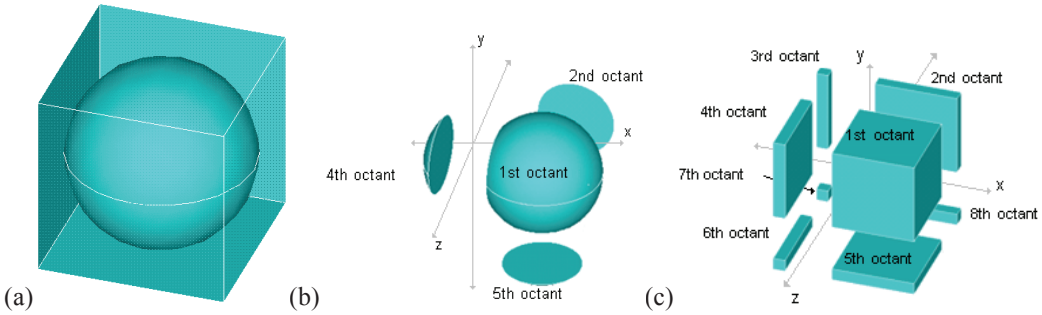

Figure 4: Particle and cube divided at same location; (a) the particle and the circumscribing cube, (b) the particle at a boundary of fluid cells (in this case cut by three planes) (c) the circumscribing cube cut by the same planes as the particle and divided into eight volumes.

The exact volume ratios for the spherical particle, ϕ_s and those estimated using the circumscribing cube, ϕ_c, for the case shown in Figure 4 are compared in Table 2. In this case, the largest error is in the first octant where using the cubic volume ratio would lead to a 20% deficit of volume.

Table 2: Difference between volume ratios of a cube and a sphere cut as shown in Figure 3(c); the corrected (Eqn. (14)) and normalised values are discussed in the error analysis

	Cube ϕ_c	**Sphere** ϕ_s	**Error** $\phi_c - \phi_s$	**Corrected** ϕ_s'	**Normalised** ϕ_s''	**Error** $\phi_s'' - \phi_s$
1st octant	0.67	0.87	-0.2	0.615	0.854	-0.016
2nd octant	0.096	0.043	0.053	0.032	0.045	0.002
3rd octant	0.0137	0	0.0137	0.003	0.004	0.004
4th octant	0.096	0.043	0.053	0.032	0.045	0.002
5th octant	0.096	0.043	0.053	0.032	0.045	0.002
6th octant	0.0137	0	0.0137	0.003	0.004	0.004
7th octant	0.00195	0	0.00195	0.000	0.001	0.001
8th octant	0.0137	0	0.0137	0.003	0.004	0.004

3. ERROR ANALYSIS

A computational study was conducted to calculate the possible range of error in using the cubic volume ratio, ϕ_c, to determine the actual volume ratio of a sphere divided amongst box-shaped cells, ϕ_s,. In this study, a spherical particle was displaced in all three Cartesian directions around the centre of eight cubic cells. These cubic cells had side length equal to the diameter of the particle. The particle was displaced in small steps to the extremities of the 8 cell complex to ensure that all possible divisions of the particle were accounted for (and noting that symmetry can be used to reduce the number of computations). At each step, the cubic volume ratio and the spherical volume ratio were recorded in all eight cells. The resulting data cloud (of more than 4,000,000 points) is shown in Figure 5. Any volume ratio other than unity or zero implies the particle is on a boundary. The mapping between cubic volume ratio, ϕ_c, and the spherical volume ratio, ϕ_s, is not unique (as can be seen in Table 3), since several different positions of the particle can give rise to the same ϕ_c, but result in different shaped sections of the sphere falling into the fluid cell (depending which of the above eight cases applies).

Table 3: Cubic and spherical volume ratios

Cubic volume ratio ϕ_c	Spherical volume ratio, ϕ_s		Middle of Range
	Max.	Min.	
0.0	0.0	0.0	0.0
0.1	0.0880	0.0280	0.058
0.2	0.2406	0.1040	0.172
0.3	0.3941	0.2160	0.305
0.4	0.5427	0.3520	0.447
0.5	0.6814	0.5000	0.591
0.6	0.8002	0.6480	0.724
0.7	0.8946	0.7840	0.840
0.8	0.9569	0.8960	0.926
0.9	0.9901	0.9720	0.981
1.0	1.0	1.0	1.0

Given the ease with which the cubic approximation to the volume ratio ϕ_c can be computed, one approach would be to apply a correction factor, bringing the value closer to ϕ_s. A simple, third order, polynomial was fitted using least squares fitting [Björck et al. (1996)] to the mid-point of the range of ϕ_s in Table 3: the form of the polynomial chosen so that it passed through the origin, and the minimisation of the least squares error was constrained to force ϕ_s =1 at ϕ_c = 1 as shown in Equation (13),

$$\phi_s' = \alpha\phi_c{}^3 + \beta\phi_c{}^2 + (1-\alpha-\beta)\phi_c \tag{13}$$

where α and β are the coefficients of the polynomial. Applying such a correction does not ensure that the volume of the particle is conserved, however this can be overcome by renormalising the values of ϕ_s' for a particle and dividing by their sum to give updated estimates of ϕ_s''. The initial fit to the values in Table 3 does not represent the optimal fit, once this renormalisation is applied, but does provide an good initial guess for the optimum values of α and β. The coefficients of the polynomial were optimised to reduce the maximum error between the corrected values of ϕ_s'' and true values of ϕ_s' for all the points in the data cloud (using the 'fminsearch' algorithm in Matlab®), giving

$$\phi_s' = -0.8457\phi_c{}^3 + 1.6625\phi_c{}^2 + 0.1832\phi_c. \tag{14}$$

The effect of applying this correction is shown in Table 2; as can be seen, the correction procedure reduces the error significantly. The corrected results are given in Figure 5, where it can be observed that the cubic volume ratio (blue) has significant deviation from spherical volume ratio (green). With the application of the Eqn. (14) and normalization, the deviation is significantly reduced (red). As shown in the Figure 5, the maximum error in the cubic volume ratios is about 20 % whereas after applying the correction the maximum error reduces to about 2.5 %.

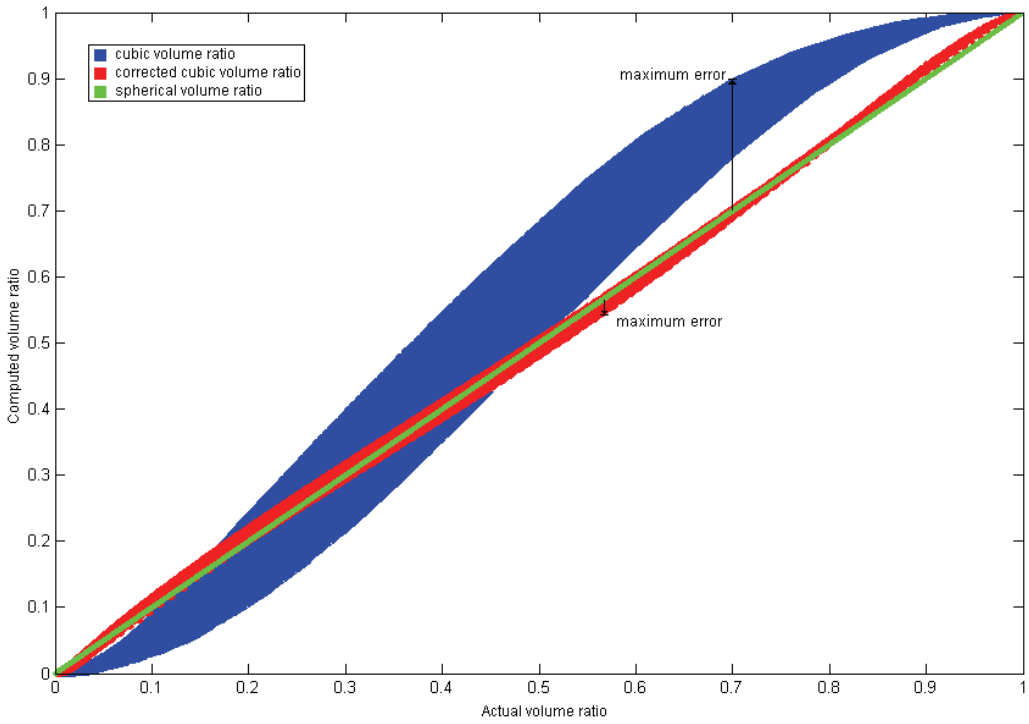

Figure 5: Actual (ϕ_s) and computed volume ratios vs. Cubic (ϕ_c), corrected (ϕ_s'') and spherical (ϕ_s) volume ratios; the distance from spherical volume ratio represents the error

It should be noted that the correction procedure does not distinguish between the different cases, and hence different shapes of sphere section. Thus, some of the remaining variance between the estimate ϕ_s'' and true value ϕ_s is due to the fact that the single correlation, which does not distinguish between the cases listed above, will not be able to fit all values perfectly. Although not shown in Figure 5, the cases do cluster together, and for the first case (when spherical particle cuts into two) there is a one to one mapping between ϕ_c and ϕ_s. Whilst it could be argued that separate correlations for each case should be developed, the possibility of sudden changes in voidage when particles cross cell boundaries and transition between cases would be difficult to exclude. The approach used here of using a single correlation across all cases, whilst not perfect, does significantly reduce the error, and ensures that the voidage varies smoothly.

CONCLUSION

Voidage is an important variable in CFD-DEM simulations. Here, an approximate method for computing the voidage using a circumscribing cube was compared with the exact method. This cubic approximation requires significantly less computational effort than exact computation, but introduces an error when the particle is on the boundary of a fluid cell. By sampling the space of possible particle position, it was found that the maximum error associated with this approximation was ~ 20%. An updated procedure was suggested that corrects the cubic estimate of voidage using a fitted polynomial; with the application of this correction, the error was reduced to less than 2.5%.

REFERENCES

[1] Anderson T. B, Jackson R. (1967). Industrial and Engineering Chemistry Fundamentals, 6, 4 527-539.

[2] Beetstra R., Van der Hoef, M. A., Kuipers J. A. M. (2007). Numerical study of segregation using a new drag force correlation for polydisperse systems derived from lattice Boltzmann simulations. Chemical Engineering Science, 62,246-255.

[3] Björck A. (1996). Numerical Methods for Least Squares Problems. SIAM.

[4] Crowe T., Sommerfeld M. & Tsuji Y. (1998) Multiphase flow with droplets and particles, CRC Press LLC.

[5] Di Felice R. (1994). The voidage function for fluid particle interaction systems, International Journal of Multiphase Flow, 20, 153-159.

[6] Ergun S. (1952). Fluid Flow through Packed Column, Chemical Engineering Progress, 48, 89-94.

[7] Hoef M., Ye M. & Kuipers J. (2005). The effects of particle and gas properties on the fluidisation of Geldart A particles, Chemical Engineering Science, 60, 4567-4580.

[8] Khawaja H. A. (2011). CFD-DEM simulation of minimum fluidisation velocity in two phase medium, International Journal of Multiphysics, 5, 2, 89-100.

[9] Khawaja H. A. & Scott S. A. (2011). CFD-DEM simulation of propagation of sound waves in a fluid particle fluidised medium, International Journal of Multiphysics, 5, 1, 47-60.

[10] Matlab® (2007a). Mathworks Corporation, Massachusetts, U.S.A.

[11] Müller, C., Holland D., Sederman A., Scott S., Dennis J. & Gladden L. (2008) Granular temperature: Comparison of Magnetic Resonance measurements with Discrete Element Model simulations, Powder Technology, 184, 2, 241-253

[12] Süli E. & Mayers D. F. (2003). An Introduction to Numerical Analysis, Cambridge University Press, Cambridge, UK.

[13] Thornton C. & Kafui D. (2003). 3D DEM simulations of gas–solid fluidised beds, Proceedings of the 3rd International Conference on Discrete Element Methods, Santa Fe, NM.

[14] Tsuji Y., Kawaguchi T. & Tanaka T. (1992). Discrete particle simulation of two-dimensional fluidised bed, Powder Technology, 77, 79-87.

[15] Zhong W., Zhang Y., Jin B. & Zhang M. (2009). Discrete Element Method Simulation of Cylinder-Shaped Particle Flow in a Gas-Solid Fluidized Bed, Chemical Engineering & Technology, 32, 3, 386-391