Quadrature-centered Averaging Scheme for Accurate and Continuous Void Fraction Calculation in Computational Fluid Dynamics-Discrete Element Method Simulations

Toni El Geitani[a] and Bruno Blais[a,*]

[a] *Research Unit for Industrial Flows Processes (URPEI), Department of Chemical Engineering, École Polytechique de Montréal, PO Box 6079, Stn Centre-Ville, Montréal, QC, Canada, H3C 3A7*

E-mail: bruno.blais@polymtl.ca

Abstract

We propose a novel method to calculate the local void fraction in solid-fluid Euler-Lagrange models in an attempt to better predict the behavior of multi-phase flows in complex CFD-DEM simulations. This method is efficient, continuous in both time and space and valid in structured and unstructured meshes. Cheap and common methods such as the particle centroid method (PCM) are commonly used to determine the void fraction. When particles are partially located in a cell, their volume contribution is not evaluated accurately, leading to inaccurate prediction of the void fraction and, consequently, inaccurate simulations. This also affects the stability of the simulation and introduces constraints on the elements' size as well as the time step for the coupling

between CFD and DEM. There are several analytical approaches that accurately determine the void fraction but these methods might not be always feasible due to their high computational cost as well as their inability to properly function on unstructured meshes. The proposed method aims at resolving these problems. We demonstrate the capacity of this method using its implementation in Lethe, an open-source CFD-DEM software.

Introduction

Fluidized and spouted beds are often encountered in pharmaceutical, food processing, oil and gas industries in addition to many others.[1,2] They are often used in drying systems, pollutant control systems, and combustion systems.[3] Due to their design and operation complexity, these beds are difficult to build and scale up.[4] Therefore, investigating the dynamics of such beds becomes vital to understand the underlying phenomena governing the different processes and allowing to test different configurations and operational conditions.[2] Computational fluid dynamics-discrete element method (CFD-DEM) proved to be an effective method in capturing the macro- and micro-scale characteristics of two phase flows that provide information specifically at the particle scale.[5] In CFD-DEM, the coupling between the solid and the fluid phase is accomplished through the incorporation of the void fraction field defined as the fraction of fluid volume in an averaging volume and the solid-fluid interactions into the Volume Averaged Navier-Stokes (VANS) equations. These equations are presented as the continuity equation (Eq. 1) and the momentum equation (Eq. 2 for model A of the VANS equations) :

$$\rho_f \frac{\partial \left(\epsilon_f\right)}{\partial t} + \rho_f \nabla \cdot \left(\epsilon_f \boldsymbol{u}_f\right) = 0 \tag{1}$$

where ρ_f is the fluid's density, $\boldsymbol{u}_f$ is the fluid's velocity and ϵ_f is the fluid's void fraction.

$$\rho_f \left(\frac{\partial \left(\epsilon_f \boldsymbol{u}_f \right)}{\partial t} + \nabla \cdot \left(\epsilon_f \boldsymbol{u}_f \otimes \boldsymbol{u}_f \right) \right) = -\epsilon_f \nabla p + \epsilon_f \nabla \cdot \left(\boldsymbol{\tau}_f \right) + \boldsymbol{F}_{\boldsymbol{pf}}^{\boldsymbol{A}} + \rho_f \epsilon_f \boldsymbol{f} \tag{2}$$

with

$$\boldsymbol{\tau}_f = \mu_f \left(\left(\nabla \boldsymbol{u}_f \right) + \left(\nabla \boldsymbol{u}_f \right)^T \right) - \frac{2}{3} \mu_f \left(\nabla \cdot \boldsymbol{u}_f \right) \boldsymbol{I} \tag{3}$$

where p is the pressure, $\boldsymbol{\tau}_f$ the deviatoric stress tensor, $\boldsymbol{f}$ the external force, and $\boldsymbol{F}_{pf}$ is the momentum transfer term between the solid and fluid phases and includes forces such as drag, virtual mass, Basset force, Saffman lift, and Magnus lift.[6] $\boldsymbol{I}$ is the identity or unit tensor. These equations are the result of rigorous local averaging of local variables over averaging volumes that are large compared to the particle spacing and small compared to the complete system, to translate the Navier-Stokes equations for the fluid and the Newtonian equations of motion for the particles into continuum momentum balances for both the fluid and particles.[7] Theoretically, these averaging volumes do not need to coincide with the cells of the mesh. Yet, the majority of void fraction schemes use this assumption.

The accuracy of the void fraction calculation directly influences the stability as well as the quality of the CFD-DEM simulation.[8] As such, several attempts have been made to accurately and efficiently predict the void fraction in CFD-DEM simulations. There exists various approaches for the calculation of the void fraction in a fluid cell, some of which are analytical methods while the others are approximation methods such as the particle centered method (PCM) and its derivatives such as the satellite point method (SPM)[9] which are presented in Fig. 1 .

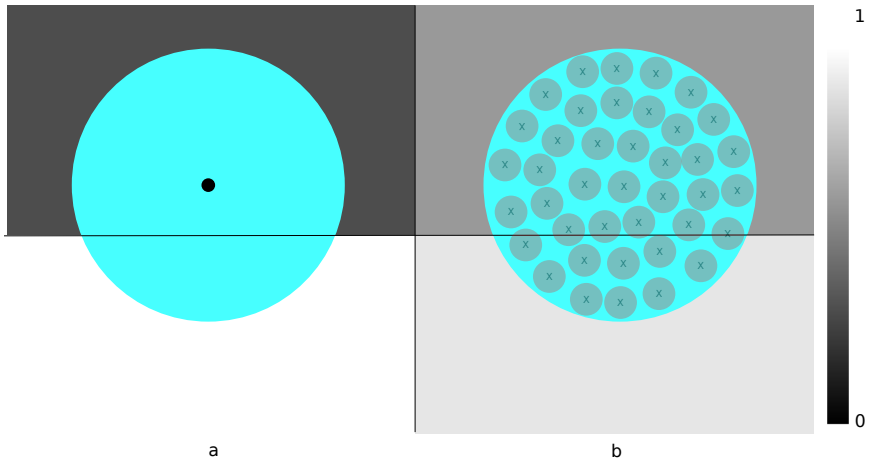


Figure 1: Representation of the particle volume fraction map for (a) PCM and (b) SPM.

In the context of the finite element method, the PCM and SPM are both calculated per element and then projected onto its nodes. The void fraction calculation using PCM and SPM is given by:

$$\epsilon_{f_{pcm}} = 1 - \frac{\sum^{N_p} V_p}{V_\Omega} \tag{4}$$

$$\epsilon_{f_{spm}} = 1 - \frac{\sum^{N_p} \sum^{N_{sp}} V_{sp}}{V_\Omega} \tag{5}$$

where V_p, V_Ω, V_{sp}, N_p and N_{sp} are respectively the particle volume, the element volume, the satellite point volume, the number of particles in an element and the number of satellite points in the particle that are also in the element. After finding the void fraction value in each element, we apply an $\mathcal{L}^2$ projection by solving the minimization problem of the following equation:

$$\min_{\epsilon \in \mathbb{R}} \frac{1}{2} \sum_i \Big(\sum_j \epsilon_j \varphi_j - \epsilon_{f_i} \Big)^2 \tag{6}$$

where ϵ_j and φ_j are the projected void fraction and the finite element shape function on each node j respectively and ϵ_{f_i} is the void fraction in each cell calculated previously. Thus, the $\mathcal{L}^2$ projection results in assembling and solving the following finite element system:

$$\int_\Omega \varphi_i \epsilon_j \varphi_j d\Omega = \int_\Omega \epsilon_{f_i} \varphi_i d\Omega \tag{7}$$

where φ_i is the void fraction test function. To the right hand side of Eq. 7, we add a smoothing term which allows to smooth the void fraction field and reduce high void fraction gradients resulting from discontinuities in the void fraction. It is represented as:

$$\iint_\Omega L^2 \nabla \varphi_i \nabla \varphi_j d\Omega \tag{8}$$

where L is the smoothing length specified by the user. The PCM is computationally inexpensive since it only searches for the centroid of the particles. If the centroid lies within a given cell, then the entire body of the particle is considered to be in this cell. However, this method leads to large errors when the centroid of the particle lies near a cell boundary [5] leading to discontinuities in time and space and resulting in significant instabilities in the simulation and unrealistic flow behaviors. Peng et al. [8] found that the PCM can achieve similar numerical stability and prediction accuracy as the analytical approach when the cell is beyond 3.82 times the particle diameter. Nonetheless, using such big cell sizes limits the accuracy in solving the fluid phase as capturing the meso-scale flow structures becomes difficult. This is because, by default, most methods assume the cell to be the averaging volume for the VANS equations where in fact, it does not have to be. As such, a smaller cell

size may satisfy the assumptions of CFD-DEM governing equations[8] rendering them valid if the methods implemented use an averaging volume independent from the cell size. In the SPM, each particle is divided into sub-particles and then the centroid of each sub-particle is treated similarly to the PCM method. A modified PCM method was developed by Peng et al.[5] where the particles are meshed into cells and the volume of the cells where added to the CFD cell in which their center is located. The SPM and the modified PCM suffers from the same limitations as the PCM. They are still discontinuous in space and time but the effect of this discontinuity is mitigated in space due to the refinement of the particle.

On the other hand, the analytical methods use geometrical-based approaches to determine the intersection between the particle and the cell and thus determine the exact volume of the particle that lies in that cell.[8] In the literature, there exist several approaches for analytically determining the void fraction: void fraction calculation in one dimensional classical soil mechanics problem,[10] two dimensional problem of dense gas-solid,[11] and three dimensional problems using structured meshes[12] as well as two and three dimensional solutions using unstructured meshes.[13] However, even though the analytical method is accurate and does not suffer from the same limitations of the PCM method, it is computationally very expensive as it involves the calculation of trigonometric functions at every CFD time step.[13,14] For example, since the discrete particle method (DPM) using the analytical method is computationally expensive, the authors used a look-up table which helped save only 2-5 % of CPU time.[13] This look-up table consists of pre-computing and storing the trigonometric analytical solutions of a particle's intersection allowing the interpolation of the particles' volume instead of calculating it at each CFD time step and thus speeding up calculations[13] Additionally, analytical methods are difficult to implement in unstructured meshes which are required to simulate complex geometries.[8]

In an attempt to reduce the computational cost of the analytical methods and to generalize its implementation to a wider range of geometries, several non analytical approaches were developed where approximate schemes instead of geometrical functions were used to relate

the particle to the cell.[8] The divided particle volume methods (DPVM)[15–17] represents the particle as a porous cube. This method adds small amounts of spatial smoothing in comparison to the analytical approach. To reduce the smoothing effect, the corrected cube DPVM was developed.[16,17] In these methods, spatial smoothing diffused the void fraction and thus, the presence of particles was weakly felt by the fluid.[5] Other applicable approaches included: grouping the fluid cell with its neighboring cells and calculating the void fraction for the group using the PCM method,[18] using a single volume-averaging sphere per CFD cell in which the void fraction was calculated and then mapped onto the CFD mesh,[19] implementing the statistical kernel method using a Gaussian distribution with a standard deviation of 1 d_p,[17,20,21] using cubes to subdivide particles which resulted in errors because cubes cannot smoothly represent curved boundaries,[22–24] and the use of a larger square grid as the averaging volume to calculate the void fraction before mapping it onto an arbitrary shaped CFD grid with a smaller cell size.[14] Another method used less often in CFD-DEM is coarse graining (CG) where neighboring particles are clumped together to form a single grain. This leads to better continuity and significant computational cost reduction but nonetheless introduces some inaccuracies in the numerical results.[25,26] In essence, CG is similar to SPM in the way particles are clustered. However, the difference between these methods is that SPM is a refining strategy where the refined particles are only used for void fraction calculation while CG is a coarsening strategy where the particles of the same cluster are linked by the forces they feel and the movements they undergo. Since these methods, in general use the cell as the averaging volume, they are limited by the type and the size of the mesh used and the results obtained are usually highly mesh dependent.

Most of these approaches were developed in the context of the finite volume method (FVM). In an attempt to overcome the limitations concerning computational efficiency, accuracy, and stability which many methods suffer from, we aim at decoupling the averaging volume from the computational cell size. We therefore develop a new analytical void fraction scheme in the context of the finite element method and implement it in the

open source CFD-DEM software Lethe.[27–30] The scheme calculates the void fraction in a volume-averaging sphere centered at the quadrature point which is where we are interested in determining the void fraction. Our developed scheme is analytical since it involves the calculation of intersections between particles and volume-averaging spheres and it is cheaper than current analytical methods since sphere-sphere intersections are much simpler to evaluate than sphere-polyhedron intersections. We demonstrate the space and time continuity of our scheme as well as its computational efficiency using a particle sedimentation case. We then prove the superiority of the QCM scheme over the other discussed methods through the study of the Boycott effect in the sedimentation of a group of particles in water. We later demonstrate its accuracy in predicting fluidization patterns as well as its mass conservation properties by simulating bubble formation in a flat spouted bed and comparing with experimental results as well as other void fraction schemes.

Description of the Method

We implement a new void fraction scheme that leverages the finite element method and which is available within the open-source software Lethe at `https://github.com/lethe-cfd/lethe`. In finite elements, equations are assembled through the integration over every cell by summing over the weak form of the residual calculated at quadrature points which are defined using quadrature rules (e.g. Gauss-Legendre quadratures). The number of quadrature points is determined by the equation and the degree of the polynomial we seek to integrate. For example, Gauss-Legendre quadrature in 1D can integrate polynomials of order $2n_q - 1$ using n_q quadrature points. The method we propose in this work involves constructing a volume-averaging sphere around each quadrature point and then calculating the void fraction inside of it. We call our new void fraction method the quadrature centered method (QCM) as it is based on calculating the void fraction directly at every quadrature point. It is important to note that our scheme is applicable when an analytical formulation is available

for the calculation of intersection between the particles and the volume-averaging spheres. In the present work, we limit our discussion to spherical particles for which the sphere-sphere intersections are easy to calculate.

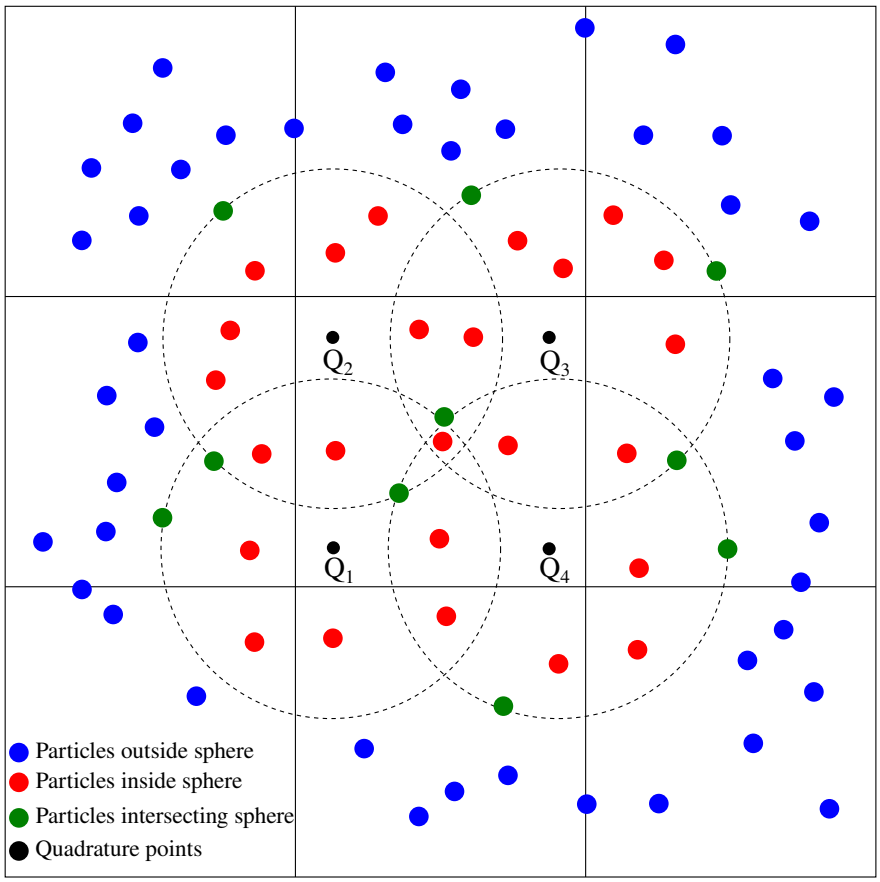


Figure 2: Volume-averaging spheres used in the calculation of the void fraction in computational cells for a 4 point quadrature rule.

Fig. 2 schematically demonstrates the volume-averaging spheres used for a given mesh. Instead of calculating the void fraction in a cell using the particle centered method (PCM) or the satellite point method (SPM) and then projecting it onto the quadrature points through $\mathcal{L}^2$ projection, we evaluate it directly at the quadrature point. This makes the quadrature points of the same element feel different void fraction values depending on the number of particles neighboring the quadrature point instead of feeling the same element value of the void fraction. Moreover, the volume-averaging spheres can exceed their mother cell boundaries and as such, can feel particles from neighboring cells. Since the calculation for the

sphere-sphere intersections are exact, this renders the scheme not only continuous in space but also in time. Thus, it results in better stability of the solver and is not limited by the size of the CFD cell as is the case with other methods such as the PCM. Furthermore, the intersection between two spheres (particle and volume-averaging sphere) is easier to calculate than the intersection between a sphere and a polyhedron (particle and cell) which makes the determination of the void fraction analytically more computationally efficient. Additionally, this method can be implemented on any finite element mesh whether structured or unstructured since we use volume-averaging spheres that are not bounded by the CFD cell. As such, implementing this method in unstructured meshes as well as in meshes with different cell sizes requires no modifications. The approach can thus work with tensor elements (quadrilaterals in 2D, hexahedron in 3D) or with simplex elements (triangles or tetrahedron). This method could also be extended to the finite volume approach by calculating the void fraction using the QCM at the centroid of the finite volume cells resulting in a method similar to that implemented by Kuang et al.[19] The only consequence would be the absence of intrinsic variations of the void fraction within the cells as occurs in FEM.

We determine the distance d between the center of the particle and the quadrature point in order to locate the particle with respect to the volume averaging sphere and we calculate the solid fraction of particles in the sphere accordingly. Fig. 3 shows the three possible cases between a particle and the sphere where R_s is the radius of the volume-averaging sphere and r_p is the radius of the particle.

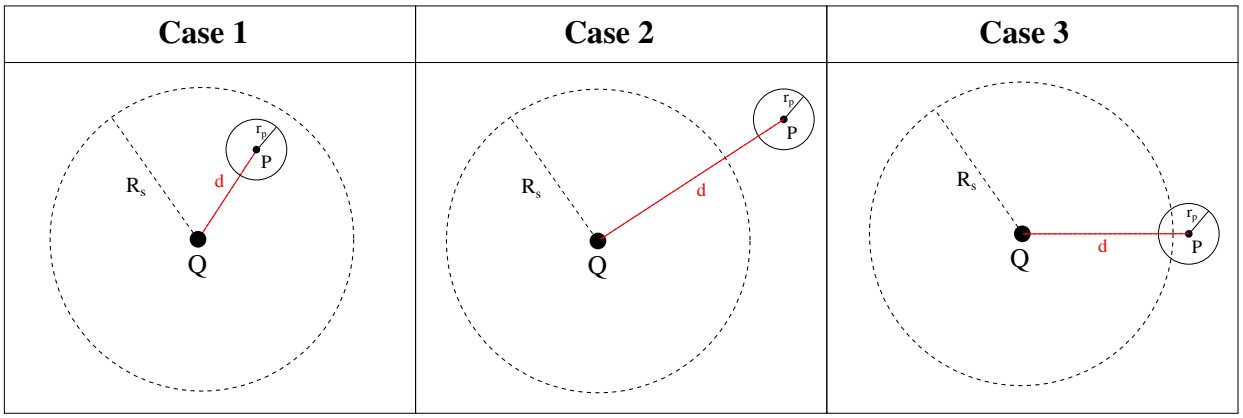


Figure 3: Volume-averaging sphere and particle intersection.

The void fraction at a quadrature point is calculated as follows:

$$\epsilon_f = 1 - \frac{\sum_j V_{s,j}}{V_{sphere}} \tag{9}$$

where $V_{s,j}$ represents the solid volume of the j^{th} particle in the volume-averaging sphere and V_{sphere} represents the volume of a single volume-averaging sphere. The choice of the sphere's volume can be adapted to the simulation requirements but for a general case, we aim to take it by respecting the following:

$$\frac{h_\Omega}{2} \leq R_s \leq h_\Omega \tag{10}$$

where h_Ω is the size of element Ω. This ensures that no particle inside the cell is missed by the spheres and that the volume of the spheres do not lie outside the neighboring cells of the current cell. To ensure adequate parallel efficiency, the parallel solver only generally keep one layer of ghost cells. Consequently, we wish to ensure that we only need to loop over the first layer of neighboring cells to detect the neighboring particles.

In our implementation, unless otherwise specified, we take the radius to be the maximum usable ($R_s = h_\Omega$) to ensure better spatial distribution. Whether in 2D or in 3D, three

cases exist to determine the particles volume in a volume-averaging sphere as shown if Fig. 2.

Case 1: particle j completely inside the volume-averaging sphere applicable when $d \leq R_s - r_p$. The volume of the particle (V_p) is entirely added to the volume of solids in sphere (V_s).

$$V_{s,j} = \frac{\pi d_p^{\mathrm{D}}}{2\mathrm{D}} \tag{11}$$

where d_p is the particle's diameter and D is the dimension of the problem.

Case 2: particle j completely outside the volume-averaging sphere applicable when $d \geq R_s + r_p$. We do nothing in this case as the particle does not contribute to the solids' volume inside the sphere.

Case 3: particle j is partially inside the volume-averaging sphere applicable when $R_s - r_p < d < R_s + r_p$. In this case, we calculate the intersection between the sphere and the particle and we add to the solids' volume, the volume of the portion of the particle that lies inside the sphere. In order to do so, we use the definition of a spherical cap to calculate the volume of intersection between the particle and the sphere. This volume of intersection is determined as the sum of the volume of the two spherical caps A and B shown in Fig. 4 that make up the intersection between the particle and the sphere.

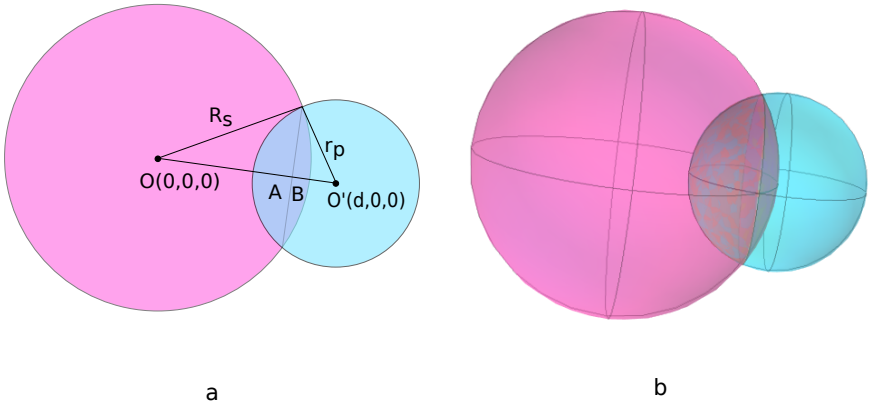


Figure 4: Intersection between two spheres in a) 2D and b) 3D

It is defined as:

$$V_{s,j} = \begin{cases} r_p^2 \cos^{-1}\left(\frac{d^2+r_p^2-R_s^2}{2dr_p}\right) + R_s^2 cos^{-1}\left(\frac{d^2-r_p^2+R_s^2}{2dR_s}\right) - \frac{1}{2}\sqrt{C}, & \text{in 2D} \\ \frac{\pi(R_s+r_p-d)^2(d^2+2dr_p-3r_p^2+2dR_s+6r_pR_s-3R_s^2)}{12d}, & \text{in 3D} \end{cases} \tag{12}$$

where C is:

$$C = (-d + r_p + R_s)(d + r_p - R_s)(d - r_p + R_s)(d + r_p + R_s) \tag{13}$$

Additionally, since there exist more than one sphere per cell and a particle can be felt by more than one sphere, we normalize our void fraction to ensure mass conservation. For the volume-averaging spheres, we multiply its volume by the weight of its respective quadrature point and then divide it by the sum of the weights of all quadrature points in the given cell. This ensures that the sum of all spheres' volumes in a cell is equal to the volume of the cell while also accounting for the fact that not all quadrature points contribute equally to a cell's volume. For the particle's volume in sphere, we divide by the total particle's contribution in all spheres in which it is felt and then multiply by the the particle's volume.

The normalization thus becomes:

$$\overline{V}_{sphere} = \frac{V_{sphere} w_q}{\sum_i w_i} \tag{14}$$

$$\overline{V}_s = \frac{\sum_j (V_{s,j} V_p)}{V_{contribution}} \tag{15}$$

where w_q is the weight of the quadrature point which represents the center of the current sphere, w_i is the weight of the i^{th} quadrature point in the cell and $V_{contribution}$ is the total of all the particle's volume contribution in the system.

This method is valid at the boundaries. Even though the volume-averaging spheres of boundary cells might exceed the domain, the fact that we normalize the spheres' volumes by the cell's volume and the solids' volume in the sphere by its total contribution makes the error due to the region out of the computational domain negligible. This ensures temporal and spatial continuity at the boundaries as well as mass conservation.

Particle Sedimentation

In order to demonstrate the spatial and temporal continuity of the quadrature centered method void fraction scheme, we simulate a test case of particle sedimentation in water using the CFD-DEM solver in Lethe.[30] We compare this method with two different void fraction schemes: the particle centered method and the satellite point method. Only these methods are evaluated as they represent well-known methods frequently used and which are relatively simple to implement in a software parallelized using distributed memory parallelization (MPI).

Simulation Setup

We simulate the particle sedimentation in a rectangular bed large enough so that the particle does not feel the walls. The mesh was chosen to have a non negligible void fraction ($\epsilon_f < 0.97$)

while being large enough to fit at least three particles so that the PCM is valid. We did not use smoothing for the void fraction as we want to showcase the behaviors of the methods without interference of artificial smoothing. The particle was placed at the top of the bed and allowed to fall due to gravity. Only drag and buoyancy forces were enabled for this simulation. The physical and numerical parameters for the simulation are presented in Table 1.

Table 1: Physical and numerical parameters for the particle sedimentation test case.

Simulation control			
End time (s)	0.25		
Coupling frequency	10^2	CFD time step (s)	10^{-3}
Geometry			
Bed Height (mm)	200	Bed Width and Length (mm)	100
Wall Thickness (mm)	0	Mesh Type	dealii::subdivided-hyper-rectangle
Mesh Refinement	0	Mesh Subdivisions in x-y-z	5-24-5
Particles			
Number	1	Diameter(m)	5×10^{-3}
Density(kg/m^3)	2500	Young Modulus (N/m^2)	1×10^6
Particle-particle poisson ratio	0.3	Particle-wall poisson ratio	0.3
Particle-particle restitution coefficient	0.2	Particle-wall restitution coefficient	0.2
Particle-particle friction coefficient	0.1	Particle-wall friction coefficient	0.1
Particle-particle rolling friction	0.2	Particle-wall rolling friction	0.3
Liquid phase			
Viscosity ($Pa \cdot s$)	1.002×10^{-3}	Density (kg/m^3)	997
Inlet Velocity (m/s)	0	Void fraction smoothing factor L^2	0
Linear Solver			
Method	GMRES	Max iterations	5000
Minimum residual	1×10^{-10}	Relative residual	10×10^{-3}
ILU preconditioner fill	1	ILU preconditioner absolute tolerance	1×10^{-12}
ILU preconditioner relative tolerance	1		
Non-linear solver			
Tolerance	1×10^{-9}	Max iterations	10

We perform the simulation using the three previously mentioned void fraction schemes and the Difelice drag model.[31] We perform the SPM for two different numbers of pseudo-particles (7 and 56). We track the sedimenting particle's velocity and compare it between the different methods implemented.

Results and Discussion

The instantaneous particle velocity is shown in Fig. 5 for the Difelice drag model.[31] The velocity profile of the particle follows the analytical solution until reaching the correct value of the settling velocity $v_r = 0.4276 m/s$ which occurs when the drag balances the gravitational force and buoyancy.[32] v_r is calculated as:

$$\boldsymbol{v}_r = \sqrt{\frac{V_p(\rho_p - \rho_f)\boldsymbol{g}}{\frac{1}{2}\rho_f C_D A_{ref}}} \tag{16}$$

where V_p is the volume of the particle, ρ_p and ρ_f are the particle's and fluid's densities respectively, $\boldsymbol{g}$ is the gravitational acceleration, A_{ref} is the particle's cross-sectional area and C_D is the drag coefficient given based on the drag model used.

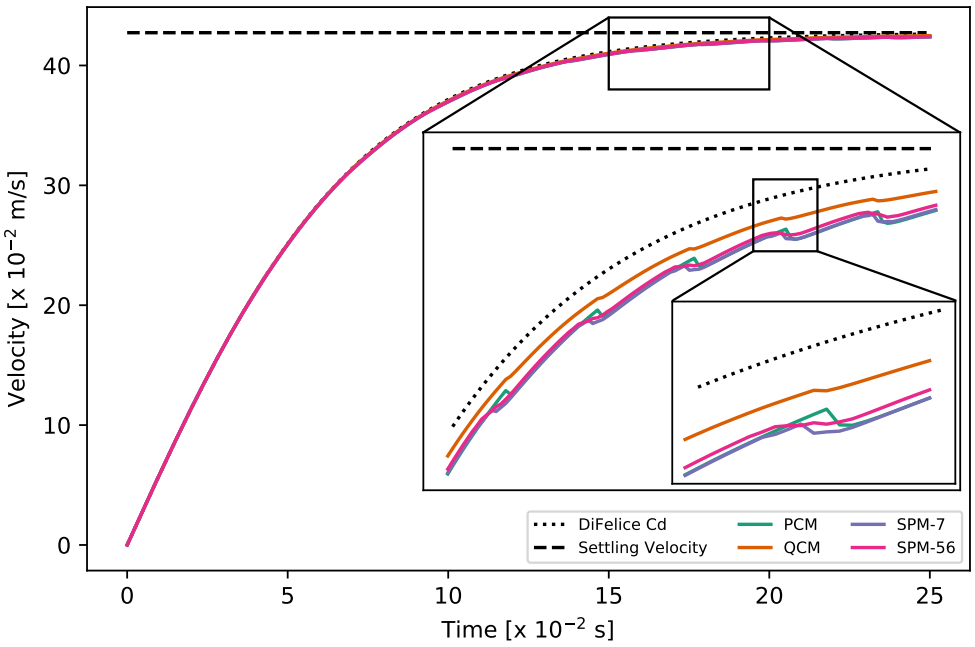


Figure 5: Instantaneous particle velocity during sedimentation for the different void fraction schemes.

The QCM leads to a slightly higher velocity values than the PCM and SPM methods which is closer to the expected derived value as it is not as sensitive to the size of the cells.

The PCM is discontinuous as the void fraction completely migrates from the first cell to its neighbor as soon as the particle's center passes between cells. We observe the same phenomenon with the SPM of 7 pseudo-particles. When we increase the number of pseudo-particles to 56, the discontinuity is reduced. The void fraction still diffuses largely among neighboring cells for the PCM and SPM methods as all quadrature points of the current cell feel the same void fraction. The QCM method leads to a continuous void fraction as its void fraction profile does not jump between elements in two consecutive CFD time steps but transitions progressively with time. This is because no new quadrature points from neighboring cells actually see the particle or start losing sight of the particle until it has moved sufficiently. This leads to a more realistic representation of the void fraction not only in time but also in space as the void fraction is not as diffused as the other methods.

Effect of the CFD Time Step

Additionally, we study the dependence of the void fraction scheme on the CFD time step for a simulation time of 0.2 s. For this, we rerun the simulation with smaller cells (mesh subdivision in x-y-z is 14-24-14). This allows to have a non negligible void fraction ($\epsilon_f < 0.96$) in the element resulting in more influence on the velocity and as such, the effect of the CFD time step would be clear. However, it is expected that the simulation will slightly under-predict the particle's velocity due to the large particle to cell size ratio. The case was simulated for CFD time steps of $10^{-3}\,s$, $10^{-4}\,s$, $10^{-5}\,s$ and $10^{-6}\,s$. Fig. 6 demonstrates the particle's velocity profile for the different cases. Only PCM and QCM are studied as the SPM is considered as a smoother PCM method. Therefore, it suffers from the same limitations of the PCM but due to its smoother nature, its instability is often delayed compared to the PCM.

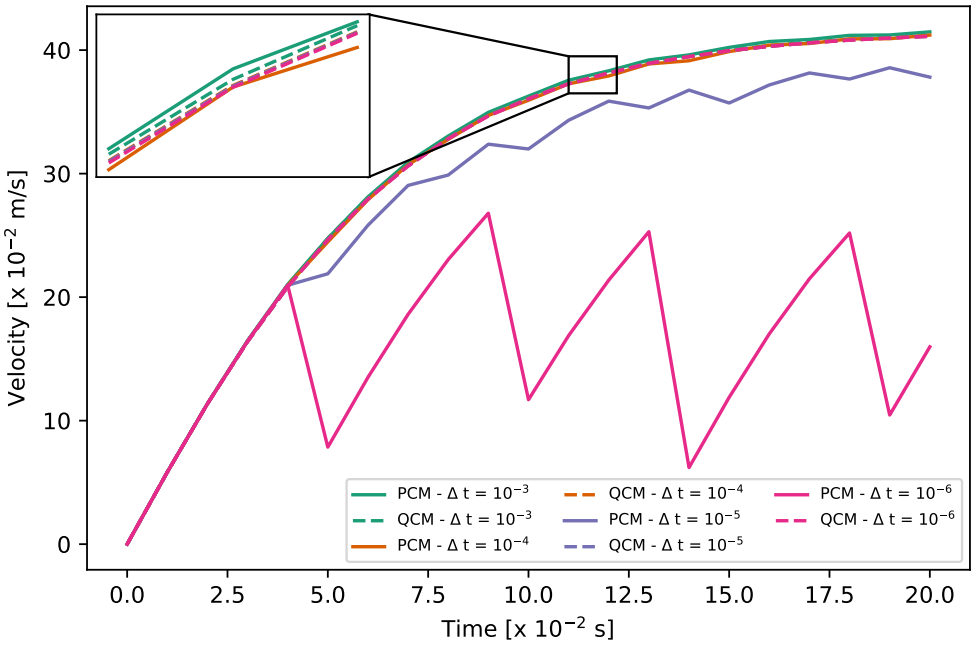


Figure 6: Effect of the CFD time step on the void fraction schemes.

A decrease in the CFD time step results in an increase in the instability and discontinuity of the PCM method. Below a certain time step, the PCM simulations become unrealistic. This is due to the term $\frac{\partial \epsilon_f}{\partial t}$ of the VANS equations[29] which tends towards infinity as the CFD time step tends to 0. Since the PCM is discontinuous, this term becomes far too large below a certain value of the time step. On the other hand, the QCM method does not result in under-prediction of the particle's velocity for all CFD time steps and it is invariant to the changes in the time step. Its stability is not affected and as such it is shown to be continuous with time. This is important especially in simulations where more temporal accuracy is desired. The continuity in time of the QCM scheme ensure that the CFD-DEM solver is well posed for all values of Δt as demonstrated by this test case. This is one of the main results of this work.

Effect of the Cell to Particle Size Ratio

We study the dependence of the void fraction schemes on the spatial discretization as well. We begin with a mesh subdivided in x-y-z into 2-4-2 subdivisions. In order to have a non-

negligible void fraction with such large cells, we increase the size of the particles to 0.01 m. We then refine the mesh three times simulating the particle sedimentation at each refinement. Similarly to the previous section, only PCM and QCM were compared. In QCM, R_s is a user defined constant and as such, we want to determine the influence of this choice on the overall performance of the method. Therefore, we consider two cases: the first case takes $R_s = h_\Omega$ while the second case takes $R_s = 2\ d_p$. The results obtained from both cases are shown in Figs. 7 and 8 respectively.

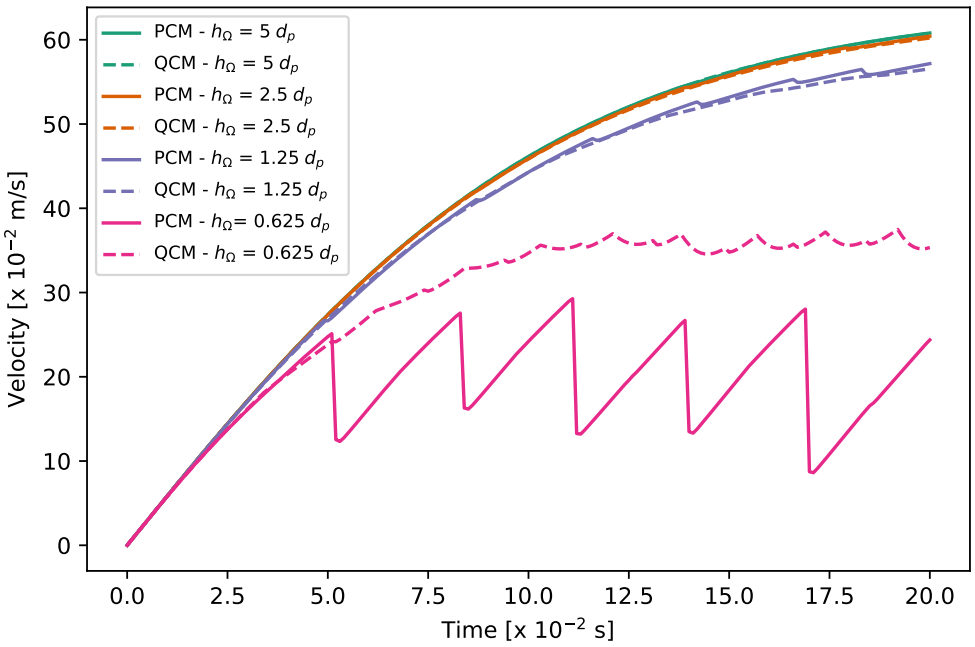


Figure 7: Effect of the element size on the void fraction schemes using QCM with $R_s = h_\Omega$.

When R_s depends on the size of the mesh elements, the QCM behaves somewhat similarly to the PCM as the mesh is refined. This is because the volume of the sphere around a quadrature point becomes smaller with every refinement. This leads to very high solid to fluid volume fractions in the sphere which results in unrealistically small values of the void fraction. This is the same in the case of the PCM. The last two refinements with the smallest cell sizes lie below the minimum cell size for which the governing equations are valid and for which the PCM is stable and accurate.[8] Thus, the bad prediction of the velocity. However, even the though the velocity values obtained by the QCM are not valid at smaller meshes,

they still represent a continuous scheme unlike the PCM where the velocity values oscillate nonphysically. This issue can be circumvented by defining a constant sphere radius. We took $R_s = 2\ d_p$.

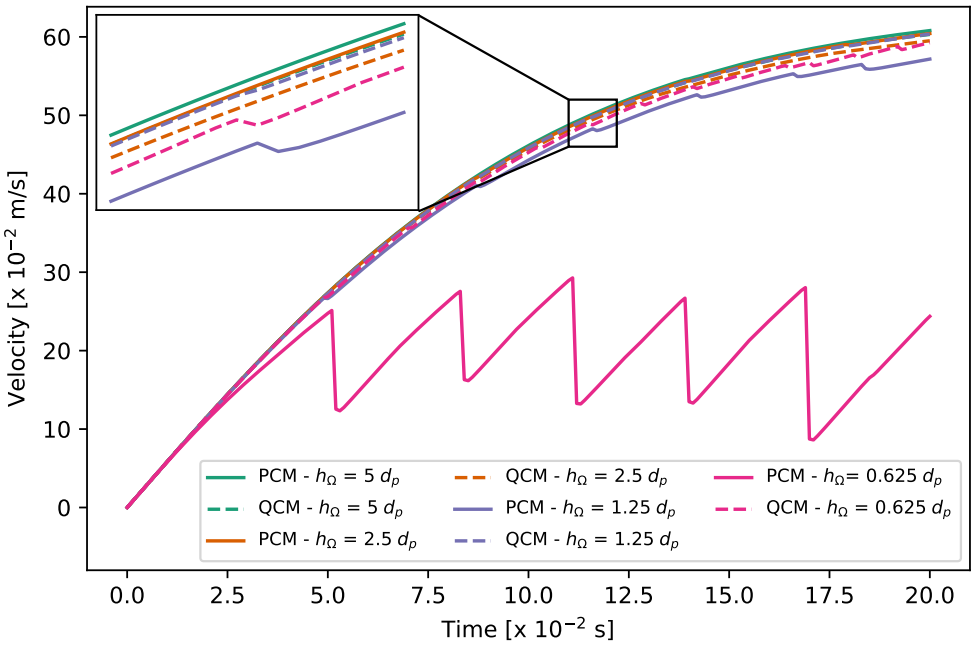


Figure 8: Effect of the element size on the void fraction schemes using QCM with $R_s = 2\ d_p$.

By defining a fixed radius R_s, the size of the virtual sphere becomes independent of the mesh. This ensures that no matter the size of elements, the void fraction is always calculated using a volume large enough for the VANS equations to be valid. This allows us to use a much finer mesh than normally allowed in CFD-DEM for most void fraction schemes going all the way to meshes smaller than the particle size as shown in Fig. 8 for the case of $h_\Omega = 0.625 d_p$. As such, we achieve higher accuracy on the fluid phase by capturing better the fluid flow without sacrificing the validity and stability of the void fraction scheme used. The initial mesh as well as the last two refinements do not satisfy the condition of Eq. (10) as $R_s = 0.4 h_\Omega$ for the initial mesh, $R_s = 3.2 h_\Omega$ for the last refinement and $R_s = 1.6 h_\Omega$ for the previous refinement. Nonetheless, the QCM works well. This is because in the case of a single sedimenting particle, even though the sphere might exceed the neighboring cells, there are no other particles in the system to be detected. Therefore, the method does not miss

any particle in the system. This is not true for dense solid-fluid systems. Additionally, for the initial mesh, where the sphere used is too small compared to the cell size, a discontinuity is noticed in the zoomed portion of the Fig. 8. In this case, parts of the particle are not captured by the small spheres which can lead to mass losses and discontinuities as the particle moves between spheres.

Numerical Efficiency

We determine the computational efficiency for the different methods as we increase the number of particles. In order to fit a large number of particles without changing the size of the geometry, we decrease the diameter of the particles from 5 mm to 2 mm. For the purpose of obtaining results faster, we increase the CFD time step from 10^{-3} s to 10^{-2} s as well as the coupling frequency from 100 to 500 to ensure a small enough DEM time step. Fig. 9 presents the average computational time of each method per CFD time step as a function of the number of particles N.

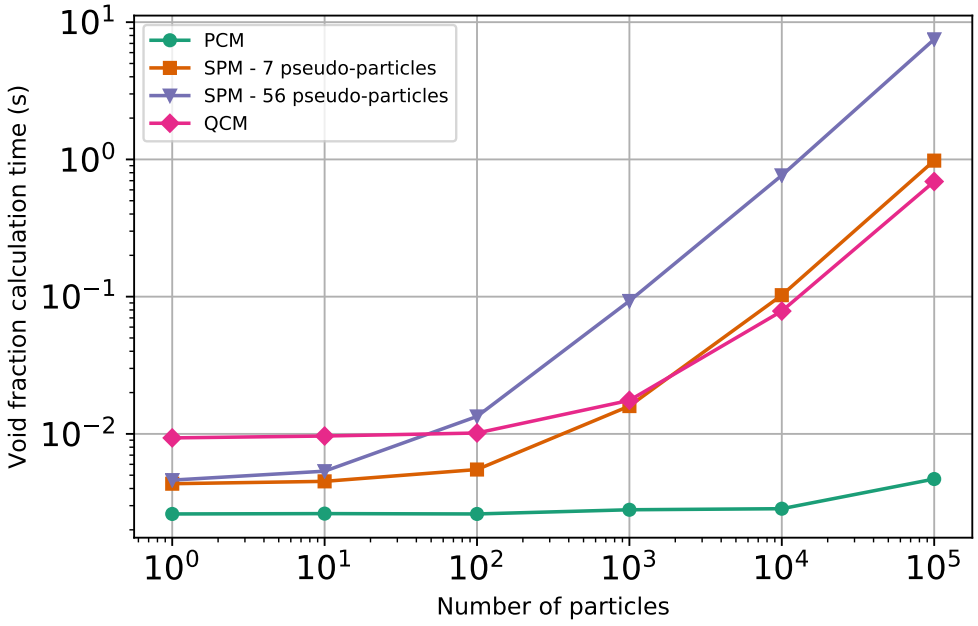


Figure 9: Average time required to calculate the void fraction in a CFD time step as a function of the number of particles.

The numerical efficiency of SPM decreases drastically as the number of pseudo-particles increases. At low number of particles N, the difference in computational time between the methods is not important. This slight difference becomes significant when the number of particles increases. For $N < 150$, the QCM is the most expensive. For $150 < N < 2000$, the QCM becomes cheaper than the SPM with 56 pseudo-particles. However, as the problem becomes larger and for $N > 2000$, the QCM becomes more efficient than the SPM method even when only 7 pseudo-particles are used. The PCM remains the most computationally efficient method as it is the simplest to implement. In addition to its spatial and temporal continuity, the QCM shows that for large problems, its computational cost is relatively good and comparable to other non-analytical methods. Overall, the added cost of the QCM did not increase the simulation run time significantly ($< 10\%$).

Boycott Effect

We compare the PCM, SPM with 7 pseudo particles, and QCM performances to determine their effect on the stability of a dense liquid-solid system and their accuracy is solving such systems.

Simulation Setup

The setup of the simulation is similar to that of the previous particle sedimentation case described in Table 1. The new bed height is 400 mm while the bed length and width were kept the same at 100 mm. The mesh subdivisions in x-y-z are 20-60-20 and were determined so that the PCM is valid. For QCM, we choose R_s as to obtain a volume averaging sphere with the same volume as the element (Ω) in which its center is located, so that $R_s = \sqrt[3]{\frac{3V_\Omega}{4\pi}}$. 100,000 particles with a diameter of 0.002 m and density of 2500 kg/m^3 were inserted at the top of the bed and were allowed to fall under the force of gravity. The simulation was run for 4 s with a CFD time step of 10^{-3} s and a coupling frequency of 50. More solid-

fluid forces were enabled for this simulation compared to the single particle sedimentation. These are drag, buoyancy, shear and pressure forces. The Difelice[31] drag model was used to calculate the drag force. Initially, the smoothing length was set to 0 for all the void fraction schemes. Three different inclination angles α of the bed relative to the vertical were studied, $\alpha = 0°,\ 5°,\ 10°$. The inclination of the bed was simulated by changing the direction of the gravity vector. It is given as $\vec{g} = (0, -9.81, 0)$ for $\alpha = 0°$, $\vec{g} = (-0.855, -9.773, 0)$ for $\alpha = 5°$ and $\vec{g} = (-1.703, -9.66, 0)$ for $\alpha = 10°$.

Results and Discussion

The simulation using the SPM void fraction scheme was unstable. We smoothed the void fraction field by using smoothing lengths $L^2 = 2d_p^2,\ 4d_p^2,\ 6d_p^2$ but the SPM method remained unstable and the linear solver failed to converge after few iterations for the different smoothing lengths used. For the PCM, the linear solver was also struggling to converge when no smoothing was implemented. Therefore, we reran the simulation with smoothing lengths $L^2 = 2d_p^2,\ 4d_p^2$. The PCM worked for all inclination angles when $L^2 = 4d_p^2$. However; increasing the smoothing length adds more artificial diffusion to the void fraction scheme. The failure of the linear solver is due to the first term $(\rho_f \frac{\partial(\epsilon_f)}{\partial t})$ of the continuity equation (Eq. 1) which is very stiff when the void fraction is discontinuous and as $\Delta t_{CFD} \to 0$. This leads to the presence of a singular matrix that prevents the convergence of the linear solver. In this case, $\Delta t_{CFD} = 10^{-3}$ which is a normal value for a CFD simulation. Normally, when the fluid used is a gas, this term does not affect the stability of the simulation due to the very small density of gases in general which significantly reduce the effect of this term, however; using a liquid as the fluid makes this term more dominant. The stiffness of this term is aggravated by the small CFD time step used and the temporal discontinuity of the void fraction scheme implemented. The QCM worked well with no smoothing for the different inclination angles where the term $\rho_f \frac{\partial(\epsilon_f)}{\partial t}$ was well-posed due to the time continuity of the QCM scheme. Table 2 summarizes the conditions under which the different void fraction schemes work for a dense

solid-liquid simulation where the term $\rho_f \frac{\partial(\epsilon_f)}{\partial t}$ of the continuity equation is enabled.

Table 2: Stability condition for the various void fraction schemes: unstable (■) and stable (■).

	$(\partial\epsilon_f/\partial t)$ enabled				$(\partial\epsilon_f/\partial t)$ disabled			
Method	Smoothing Length (L^2)				Smoothing Length (L^2)			
PCM	0	$2d_p^2$	$4d_p^2$	$6d_p^2$	0	$2d_p^2$	$4d_p^2$	$6d_p^2$
SPM	0	$2d_p^2$	$4d_p^2$	$6d_p^2$	0	$2d_p^2$	$4d_p^2$	$6d_p^2$
QCM	0	$2d_p^2$	$4d_p^2$	$6d_p^2$	0	$2d_p^2$	$4d_p^2$	$6d_p^2$

Fig. 10 shows time snapshots of a 2D slice of the void fraction profile during the sedimentation process at different inclination angles for the QCM void fraction scheme.

As the bed becomes more inclined, the particles' sedimentation becomes faster and the particles reach the bottom of the bed in a shorter time compared to the non inclined bed. This is because the particles gain momentum due to being hindered by the upward motion of the fluid. This upward motion of the fluid is a response to mass conservation represented by the continuity equation. As the particles are subjected to gravitational forces, they fall downwards. This requires the fluid to move upwards to give space for the particles falling. The void fraction during the sedimentation process is highly fluctuating as particles are falling fast.

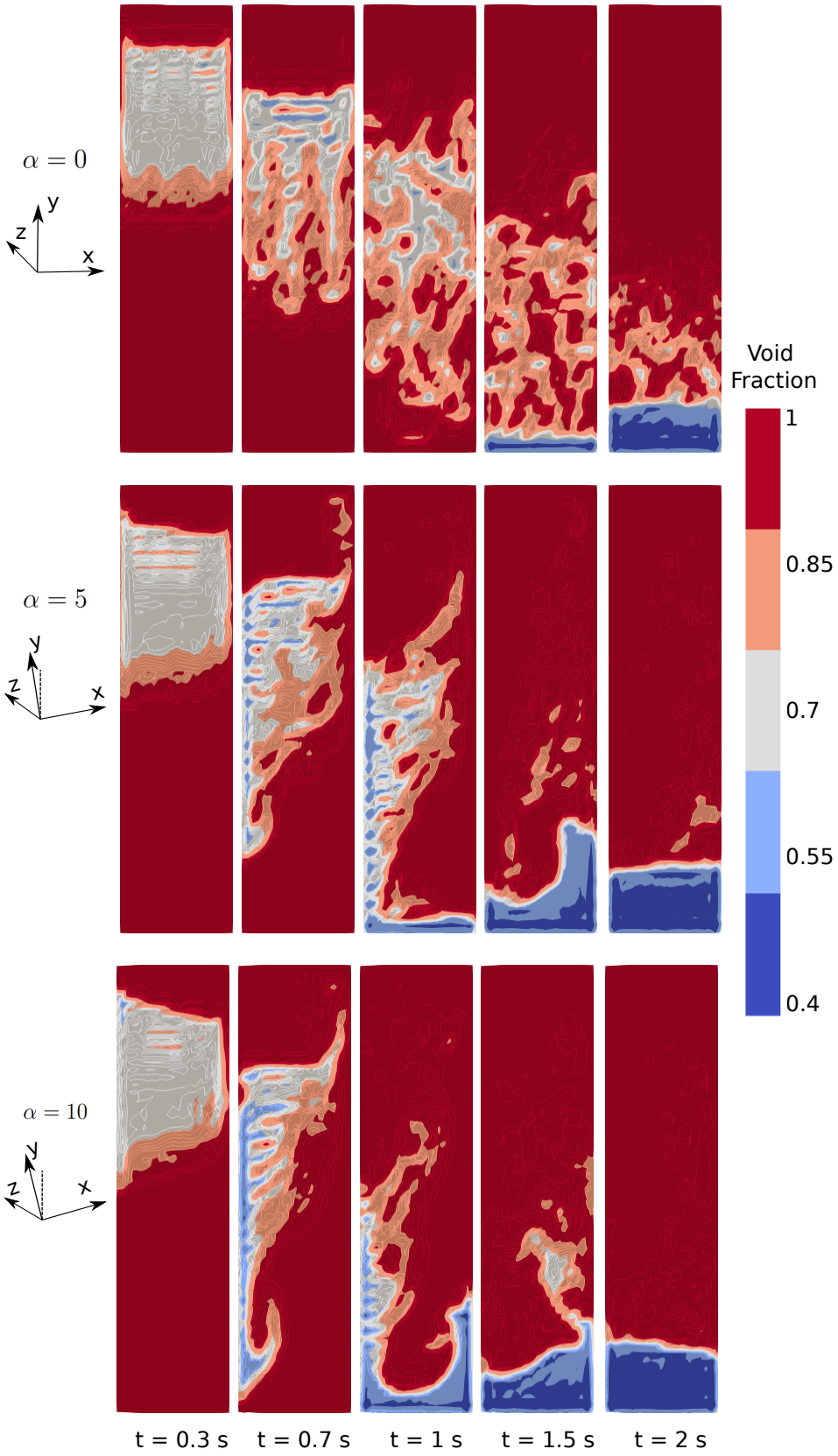


Figure 10: 2D snapshots of the void fraction profile during the sedimentation of the particles in water at different inclination angles for the QCM scheme.

The QCM was able to simulate the sedimentation with no difficulty on the linear and non linear solvers leading to fast convergence of the simulation. For PCM, some stiffness was noticed even with the smoothing of the void fraction where some iterations where taking longer to converge. The larger the inclination angle of the bed was, the longer the simulation with PCM took to converge at each time step. This is due to the increased particles' velocity as we increase the angle α, where now particles are travelling more per time step leading to a larger temporal discontinuity in the void fraction field which makes the term $\rho_f \frac{\partial(\epsilon_f)}{\partial t}$ more unstable. The simulation took twice as much time for PCM as for QCM when $\alpha > 0$ to reach the end time of 4 s. Using a smoothing length larger than $L^2 = 4d_p^2$ could help reduce the stiffness of the system being solved as it helps reduce the discontinuity of the void fraction scheme. The SPM failed since the first few iterations at simulating this case when the term $\rho_f \frac{\partial(\epsilon_f)}{\partial t}$ of the continuity equation was enabled as it resulted in a singular matrix which prevented the linear solver from converging. Even though the SPM is less discontinuous than the PCM, the refinement of each particle (in this case 7 pseudo particles) makes the discontinuities more frequent than with the PCM which explains the rapid explosion of the term $\rho_f \frac{\partial(\epsilon_f)}{\partial t}$.

Usually, in CFD-DEM simulations involving a liquid as the fluid, the first term $\rho_f \frac{\partial(\epsilon_f)}{\partial t}$ of the continuity equation (Eq. 1) is neglected as it is unstable and highly discontinuous. As the CFD time step decreases, this term is very large for the slightest discontinuity in time of the void fraction. The density of a liquid being much larger than that of a gas increases the effect of this term resulting in unstable systems. However, neglecting this term solely on a numerical basis without any physical explanation is not justifiable as we are no longer solving the volume averaged Navier-Stokes equations. In order to prove this, we monitored the y-velocity and the y-position of the center of mass of our system in time for the different void fraction schemes used. For this, we rerun the simulation using the different void fraction schemes for the stable cases with the least smoothing requirement as shown in Table 2. The results obtained are shown in Fig. 11 for the y-position and Fig. 12 for the y-velocity of the

center of mass.

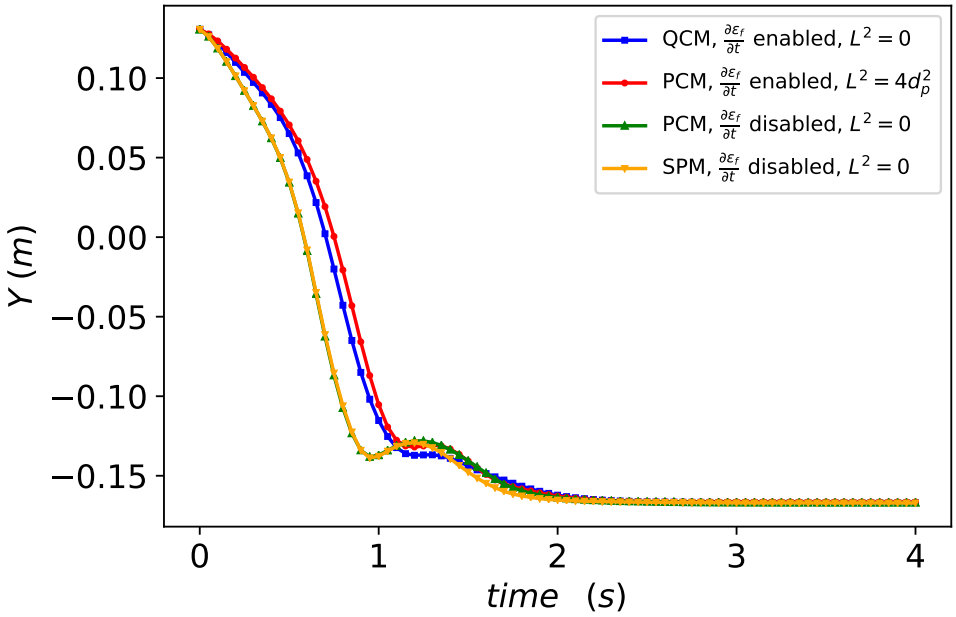


Figure 11: The y-position of the center of mass of the cluster of particles as a function of time.

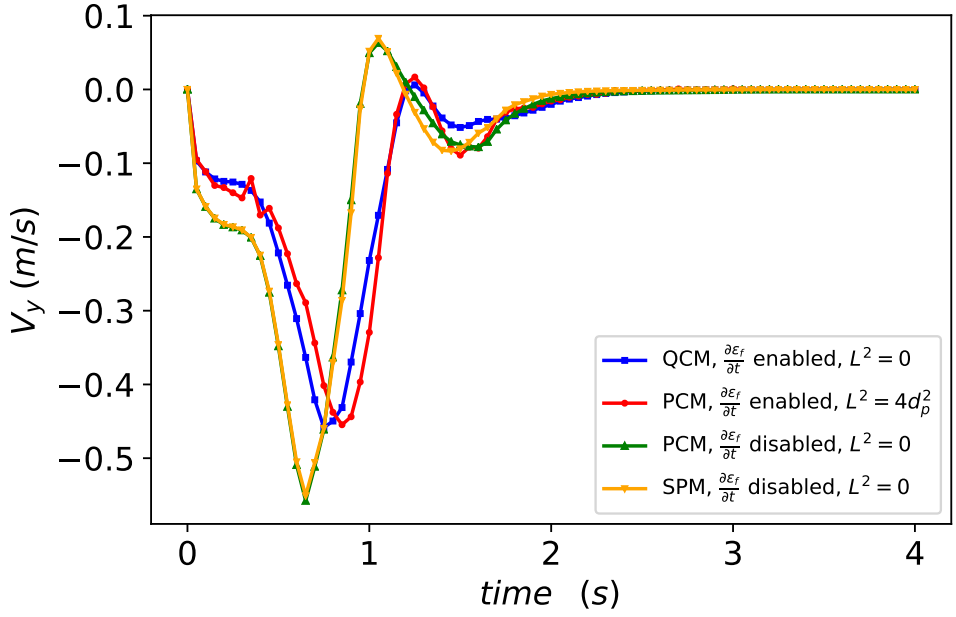


Figure 12: The y-velocity of the center of mass of the cluster of particles as a function of time.

As the term $\rho_f \frac{\partial(\epsilon_f)}{\partial t}$ is neglected, the particles fall faster. This is demonstrated by the center of mass's position and velocity where at the same simulation time, the simulations with the disabled term resulted in faster sedimentation due to larger particles' velocities. This becomes more obvious when comparing the two PCM simulations in Figs. 11 and 12. We explain this by the concept of mass conservation. As the particles move downwards, the term $\rho_f \frac{\partial(\epsilon_f)}{\partial t}$ enhances the upward movement of the fluid to give space for the particles to sediment. This upward movement slows down the particles that are in the way of the fluid. When this term is neglected, fluid is less affected by the difference in void fraction below and at the particles' locations resulting in less upward movement of the fluid and as a result faster particles.

Flat Bubbling Spouted Bed

We aim at validating our void fraction scheme by studying its effect on fluidization. For this purpose, we simulate using the CFD-DEM solver in Lethe[30] the same flat bubbling spouted bed as Wu et al.[13] and we compare with their simulations' results as well as the experimental results and the simulation results by Bokkers et al.[33] It is important to note that Wu et al.[13] and Bokkers et al.[33] used the hard sphere model in their simulations for the DEM. However, our solver uses the soft sphere model. Therefore, it is expected that there will be discrepancies in the results but the aim is not to compare the DEM models. For more information about these two models, we refer the reader to the book by Norouzi et al.[2] In this section, we compare the shape and size of the bubble formed as well as the height of the bed of particles 0.2 s after initial gas injection. We also look at the profile of the void fraction obtained from the different schemes.

Simulation Setup

The geometry as well as the physical properties of the particles are taken from Wu et al.[13] The mesh was chosen by taking the cell size to be slightly greater than 3 d_p so that the CFD-DEM simulation for the various void fraction schemes is valid.[2] We did not use smoothing of the void fraction in this case to show the exact behavior of each method. The physical and numerical parameters for this simulation are found in Table 3.

Table 3: Physical and numerical parameters for the spouted bed test case.

Simulation control			
End time (s)	0.2		
Coupling frequency	10^2	CFD time step (s)	10^{-4}
Geometry			
Bed Dimensions (m)	0.15 x 0.45 x 0.015	Wall Thickness (mm)	0
Mesh Type	GMSH	Mesh Subdivisions in x-y-z	15-45-1
Spout Width (m)	0.01	Spout Mesh Subdivisions in x-y-z	1-10-1
Particles			
Number	31050	Diameter(m)	25×10^{-3}
Density(kg/m^3)	2526	Young Modulus (N/m^2)	1×10^6
Particle-particle poisson ratio	0.25	Particle-wall poisson ratio	0.25
Particle-particle restitution coefficient	0.97	Particle-wall restitution coefficient	0.33
Particle-particle friction coefficient	0.2	Particle-wall friction coefficient	0.1
Particle-particle rolling friction	0.4	Particle-wall rolling friction	0.1
Gas phase			
Viscosity ($Pa \cdot s$)	1.81×10^{-5}	Density (kg/m^3)	1
Inlet Spout Velocity (m/s)	20	Inlet Background Velocity (m/s)	1.25
Void fraction smoothing factor L^2	0	Grad-div parameter (c^*)	0.005
Linear Solver			
Method	GMRES	Max iterations	1000
Minimum residual	1×10^{-10}	Relative residual	10×10^{-2}
ILU preconditioner fill	2	ILU preconditioner absolute tolerance	1×10^{-12}
ILU preconditioner relative tolerance	1		
Non-linear solver			
Tolerance	1×10^{-8}	Max iterations	10

For this simulation, we enable buoyancy, shear stress, pressure and drag forces. The Gidaspow model[32] was used to calculate the drag. This drag model is not the best model to be used in this study as it features a discontinuity in the calculation of the drag force at the void fraction value of $\epsilon_f = 0.8$. Since this value of the void fraction is present in our

bed, other drag models would be more appropriate. However, since we are comparing with a study in which the Gidaspow[32] model was implemented, we use it solely for comparison purposes. For more details on the description of the particle-fluid forces, we refer the reader to the Lethe CFD-DEM article.[30] We also choose R_s as to obtain a volume averaging sphere with the same volume as the element (Ω) in which its center is located, so that $R_s = \sqrt[3]{\frac{3V_\Omega}{4\pi}}$.

Results and Discussion

The formation of the bubble at the base of the flat bubbling spouted bed after 0.2 s using the Gidaspow drag model[32] is shown in Fig. 13 for the various void fraction schemes used.

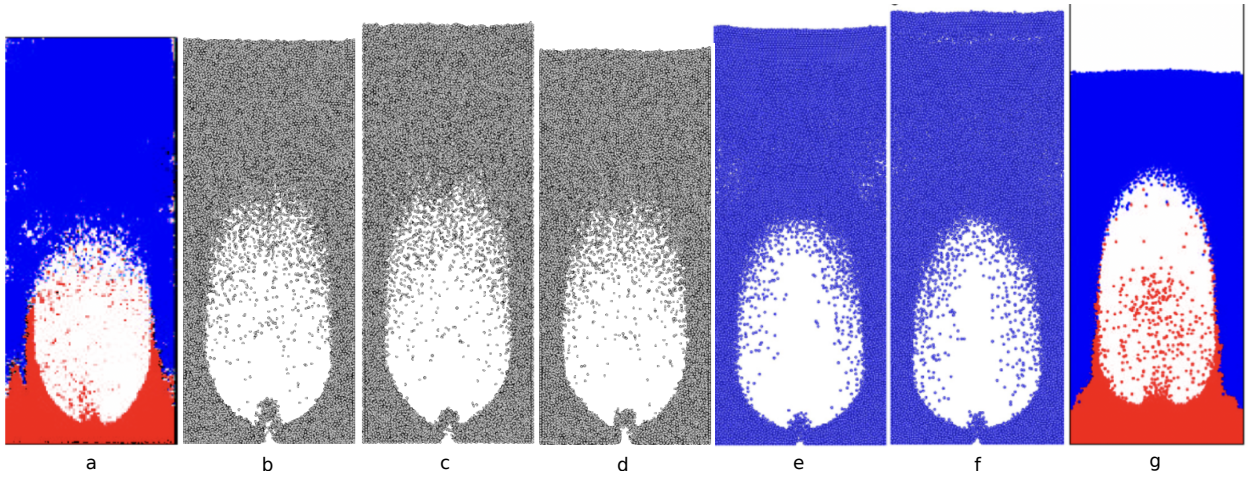


Figure 13: Particle distribution in a flat fluidized bed 0.2 s after central gas injection. (a) Experimental result by Bokkers et al;[33] (b) QCM; (c) PCM; (d) SPM; (e) DPM using analytical method;[13] (f) DPM using point approximate method (PAM) approach,[13] and (g) DPM[33]

.

Comparing between our three simulations, we notice that for the same simulation conditions, the QCM and SPM result in more accurate predictions than the PCM when compared with the experimental results. The PCM result in a larger bed height (about 5% increase) while the SPM under-predicts the bed height by 1.2 % compared to the QCM which predicts the bed height correctly. The PCM leads to few pockets of gas in the portion of the bed above the top of the bubble. This is also noticeable in the DPM using the point approximate

method (PAM) approach. Additionally, in all three methods we implemented, the shape and size of the bubble we obtain lies in between the results obtained by Bokkers et al.[33] and Wu et al.[13]

For the simulations performed by Wu et al.,[13] the analytical approach (Fig. 13 (e)) results in accurate bed height prediction and bubble formation contrary to the point approximate method where the bed height was over-predicted (Fig. 13 (f)). Additionally, the simulation by Bokkers et al.[33] (Fig. 13 (g)) leads to an over-prediction of the bubble size but to an under-prediction of the bed height. We find this unrealistic since generally a bigger bubble occupies a larger volume which should lead to a larger bed height to compensate for the volume occupied by the bubble. Even though Wu et al.[13] and Bokkers et al.[33] both use the hard sphere model and the Gidaspow drag correlation[32] in their simulations, their results are contradictory. The shape and size of the bubble as well as the bed height are completely different. The simulations by Wu et al.[13] results in particles falling down around the bubble contrary to the experimental results. This might be caused by the mesh size. Bokkers et al.[33] used a mesh with 675 elements while Wu et al.[13] used 5.6 times more cells (3780 elements). This leads to different values of the void fraction within a given cell which influences the value of the drag force in this cell. Additionally, the choice of solid-fluid forces applied can drastically change the results. Wu et al.[13] and Bokkers et al.[33] did not explicitly give details on all the solid-fluid forces they accounted for and as such, we cannot reproduce their simulations accurately.

In our simulation, we use the same mesh as the one used by Bokkers et al.[33] except for the inlet channel where we used a more refined mesh. Using the QCM method, we obtain results comparable to those of the experiment as well as to the results from the simulation of the DPM using the analytical method. QCM, SPM and DPM using the analytical method show good bubble size even though the DPM bubble is more compact. In our simulations, particles fall down through the bubble as also seen experimentally and in the simulation by Bokkers et al.[33] Moreover, the height of the bed in the simulations of both the QCM and the DPM

using the analytical approach is similar to the experimental bed height. Additionally, the particle spout seen at the base of the bubble is reproduced in all simulations even though the QCM leads to a slightly more pronounced spout than the other simulations. The occurrence of this particle spout is dependent on the grid resolution of the inlet channel. Using a finer mesh in the y direction led to the particle spout being reproduced.

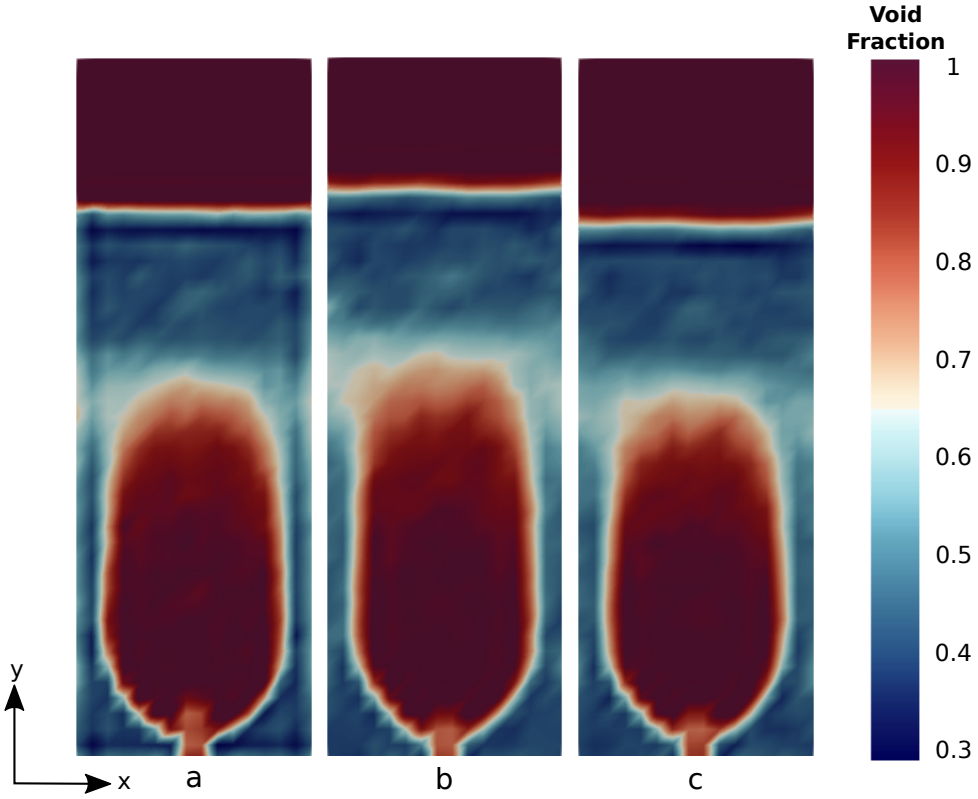


Figure 14: 2D slice showing the void fraction ϵ_f profile in a flat fluidized bed 0.2 s after central gas injection. (a) QCM; (b) PCM; and (c) SPM.

In the void fraction distribution of Fig. 14, the shape of the bubble at the base is similar between the three methods. The differences exist towards the top of the bubble. The PCM results in more air pockets in the upper packing represented by the whitish blue and light red colors on top of the bubble. This explains the difference in bed height between the methods even though the bubble size is similar. The QCM gives a good representation for the top of the bubble while for the SPM and the PCM, we notice a kink at the top of the bubble.

We demonstrated that the QCM is cheaper than the SPM when simulating a large number

of particles but still more expensive than the PCM. However, the void fraction is calculated only once per CFD time step, therefore, the cost is relatively modest. Also, the continuity in space and time of the QCM manifested itself in the convergence of the VANS equations where the number of Newton iterations was 6 ± 1 for the QCM, 8 ± 1 for the PCM, and 7 ± 1 for the SPM resulting in 25 % and 14.5 % reduction per CFD time step compared to PCM and SPM respectively. Overall, the total simulation time was 7700 s for PCM, 7400 for SPM, and 6640 s for QCM reducing the simulation time by 16 % compared to PCM and 11.5 % compared to SPM. This is in principle due to the enhancement of the condition number of the linear systems solved.

Moreover, we study the mass conservation properties of the three methods. We monitor the global continuity error (mass conservation error) that is the solution of the continuity equation as well as the maximum local continuity error in an element. The continuity error is defined as the continuity equation of the VANS equations:

$$E = \int_{\Omega} \rho_f \frac{\partial \left(\epsilon_f\right)}{\partial t} + \rho_f \nabla \cdot \left(\epsilon_f \boldsymbol{u}_f\right) d\Omega \tag{17}$$

where Ω is the simulation domain over which the error is being calculated.

Fig. 15 shows the continuity error fluctuations with time normalized by the spout inlet velocity of 20 m/s.

The QCM results in more homogeneous mass source term compared to the PCM and SPM. Among the compared methods, it is the most mass conservative. The order of magnitude of the normalized global mass source term is lowest for the QCM at 10^{-10} followed by the PCM and SPM at 10^{-7} and 10^{-5} respectively. The normalized maximum local mass conservation error order of magnitude for all methods is around 10^{-6}. In general, the QCM leads to the least fluctuations in the mass source term and the SPM leads to the most fluctuations specifically at certain CFD time steps when the mass source jumps four orders of magnitude and then drops instantly.

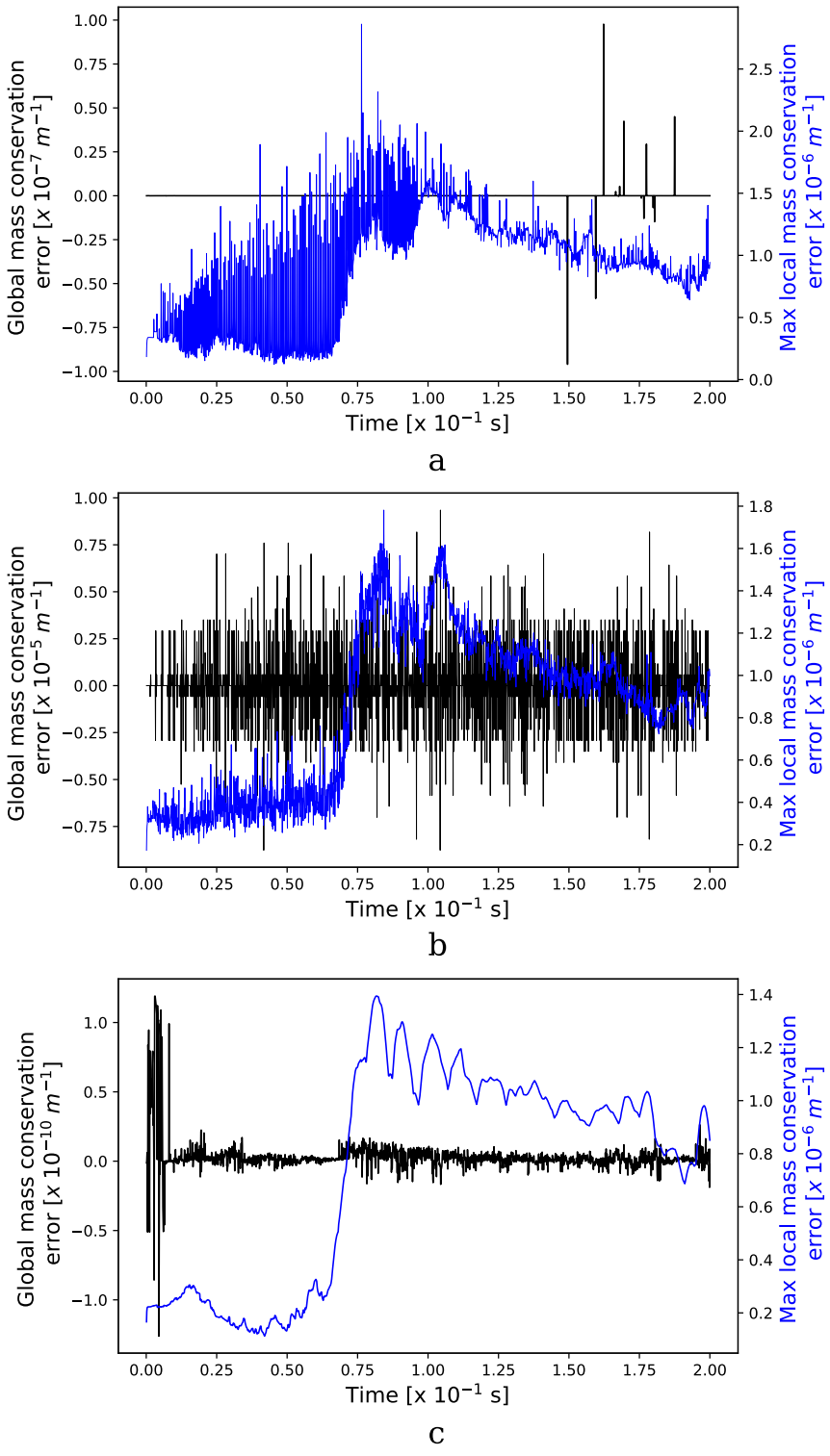


Figure 15: Global and maximum local continuity error for: (a) PCM; (b) SPM; and (c) QCM. The y-axes are scaled to the order of magnitude of the mass conservation error. The QCM results in a significantly lower mass conservation error in comparison to the PCM and SPM.

Conclusion

In this paper, we develop a new scheme to calculate the void fraction of solid-fluid systems in CFD-DEM simulations. It is mass conservative, stable and space and time continuous. Since no intersection between the particles and the cells are calculated, this method is valid on both structured and unstructured meshes. We demonstrate that the method is well-posed in the limit of $\Delta t \to 0$ and the element size when the radius of the virtual sphere is fixed. This leads to a very stable and robust void fraction scheme.

Additionally, a comparison of the results obtained by the different methods for the flat bubbling spouted bed demonstrate the significant error obtained from using the PCM where the bed height was higher than the experimental observation and the void fraction distribution was uneven. In comparison, the QCM correctly predicted the bed height and resulted in a compact distribution of the void fraction. Moreover, the improved accuracy and continuity in the calculation of the void fraction reduced the overall simulation time by 11-16 % by enhancing the condition number of the linear systems solved and reducing the number of non-linear iterations.

The implementation of the QCM void fraction scheme is advantageous in the simulation of applications with large void fraction fluctuations in time such as bubbling fluidized beds, high void fraction gradients such as packed beds, and dense solid-liquid systems. This method alleviates the need to artificially smooth the void fraction scheme when solving the $\mathcal{L}^2$ projection system as it is by nature continuous in both time and space, thus resulting in better representation of the void fraction field. Furthermore, the QCM scheme greatly reduces the limitations on unresolved CFD-DEM concerning the size of mesh elements and the CFD time step, thus ensuring better spatial and temporal accuracy. Finally, the continuity of QCM and its enhancement of the condition number of the linear systems solved allows it to be readily coupled with lesser robust high order velocity and pressure schemes without aggravating the stiffness of the linear system, thus additionally improving accuracy of results. Although we developed the QCM in the context of the Finite Element Method (FEM), it could readily

be extended to the Finite Volume Method (FVM) by calculating the intersection between the particles and the volume averaging sphere at the centroid of the cells to obtain the void fraction within the cells.

Acknowledgements

The authors would like to thank the deal.II community for their support. Bruno Blais would like to acknowledge the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) through the RGPIN-2020-04510 Discovery Grant. The authors would also like to acknowledge technical support and computing time provided by Compute Canada and Calcul Québec.

References

(1) Yeoh, G. H.; Tu, J. In *Computational Techniques for Multiphase Flows (Second Edition)*, second edition ed.; Yeoh, G. H., Tu, J., Eds.; Butterworth-Heinemann: Amsterdam, 2019; pp 227–319.

(2) Norouzi., H. R. *Coupled CFD-DEM Modeling : Formulation, Implementation and Application to Multiphase Flows.*, 1st ed.; Wiley: Newark, 2016.

(3) van der Hoef, M.; Ye, M.; van Sint Annaland, M.; Andrews, A.; Sundaresan, S.; Kuipers, J. Multiscale Modeling of Gas-Fluidized Beds. *Advances in Chemical Engineering* **2006**, *31*, 65–149.

(4) Cocco, R.; Karri, S. B. R.; Knowlton, T. Introduction to Fluidization. *Chemical engineering progress* **2014**, *110*, 21–29.

(5) Peng, Z.; Moghtaderi, B.; Doroodchi, E. A modified direct method for void fraction calculation in CFD–DEM simulations. *Advanced Powder Technology* **2016**, *27*, 19–32.

(6) Blais, B.; Bertrand, F. On the use of the method of manufactured solutions for the verification of CFD codes for the volume-averaged Navier–Stokes equations. *Computers & fluids* **2015**, *114*, 121–129.

(7) Anderson, T. B.; Jackson, R. Fluid Mechanical Description of Fluidized Beds. Equations of Motion. *Industrial & Engineering Chemistry Fundamentals* **1967**, *6*, 527–539.

(8) Peng, Z.; Doroodchi, E.; Luo, C.; Moghtaderi, B. Influence of void fraction calculation on fidelity of CFD-DEM simulation of gas-solid bubbling fluidized beds. *AIChE Journal* **2014**, *60*, 2000–2018.

(9) Blais, B.; Lassaigne, M.; Goniva, C.; Fradette, L.; Bertrand, F. Development of an unresolved CFD–DEM model for the flow of viscous suspensions and its application to solid–liquid mixing. *Journal of Computational Physics* **2016**, *318*, 201–221.

(10) Chen, F.; Drumm, E. C.; Guiochon, G. Coupled discrete element and finite volume solution of two classical soil mechanics problems. *Computers and Geotechnics* **2011**, *38*, 638–647.

(11) Li, J. Euler-Lagrange simulation of flow structure formation and evolution in dense gas-solid flows. Ph.D. thesis, University of Twente, 2002.

(12) Freireich, B.; Kodam, M.; Wassgren, C. An exact method for determining local solid fractions in discrete element method simulations. *AIChE journal* **2010**, *56*, 3036–3048.

(13) Wu, C.; Zhan, J.; Li, Y.; Lam, K.; Berrouk, A. Accurate void fraction calculation for three-dimensional discrete particle model on unstructured mesh. *Chemical Engineering Science* **2009**, *64*, 1260–1266.

(14) Boyce, C. M.; Holland, D. J.; Scott, S. A.; Dennis, J. S. Novel fluid grid and voidage calculation techniques for a discrete element model of a 3D cylindrical fluidized bed. *Computers & chemical engineering* **2014**, *65*, 18–27.

(15) Link, J.; Cuypers, L.; Deen, N.; Kuipers, J. Flow regimes in a spout–fluid bed: A combined experimental and simulation study. *Chemical Engineering Science* **2005**, *60*, 3425–3442.

(16) Khawaja, H. A.; Scott, S. A.; Virk, M. S.; Moatamedi, M. Quantitative analysis of accuracy of voidage computations in CFD-DEM simulations. *The Journal of Computational Multiphase Flows* **2012**, *4*, 183–192.

(17) Clarke, D. A.; Sederman, A. J.; Gladden, L. F.; Holland, D. J. Investigation of void fraction schemes for use with CFD-DEM simulations of fluidized beds. *Industrial & Engineering Chemistry Research* **2018**, *57*, 3002–3013.

(18) Wee Chuan Lim, E.; Wang, C.-H.; Yu, A.-B. Discrete element simulation for pneumatic conveying of granular material. *AIChE journal* **2006**, *52*, 496–509.

(19) Kuang, S.; Chu, K.; Yu, A.; Zou, Z.; Feng, Y. Computational investigation of horizontal slug flow in pneumatic conveying. *Industrial & Engineering Chemistry Research* **2008**, *47*, 470–480.

(20) Sun, J.; Battaglia, F.; Subramaniam, S. Hybrid Two-Fluid DEM Simulation of Gas-Solid Fluidized Beds. *Journal of Fluids Engineering* **2007**, *129*, 1394–1403.

(21) Xiao, H.; Sun, J. Algorithms in a robust hybrid CFD-DEM solver for particle-laden flows. *Communications in Computational Physics* **2011**, *9*, 297–323.

(22) Gui, N.; Fan, J. R.; Luo, K. DEM–LES study of 3-D bubbling fluidized bed with immersed tubes. *Chemical Engineering Science* **2008**, *63*, 3654–3663.

(23) Hilton, J.; Mason, L.; Cleary, P. Dynamics of gas–solid fluidised beds with non-spherical particle geometry. *Chemical Engineering Science* **2010**, *65*, 1584–1596.

(24) Hobbs, A. Simulation of an aggregate dryer using coupled CFD and DEM methods. *International Journal of Computational Fluid Dynamics* **2009**, *23*, 199–207.

(25) Lu, L.; Xu, Y.; Li, T.; Benyahia, S. Assessment of different coarse graining strategies to simulate polydisperse gas-solids flow. *Chemical Engineering Science* **2018**, *179*, 53–63.

(26) Di Renzo, A.; Napolitano, E. S.; Di Maio, F. P. Coarse-Grain DEM Modelling in Fluidized Bed Simulation: A Review. *Processes* **2021**, *9*, 1–30.

(27) Blais, B.; Barbeau, L.; Bibeau, V.; Gauvin, S.; Geitani, T. E.; Golshan, S.; Kamble, R.; Mirakhori, G.; Chaouki, J. Lethe: An open-source parallel high-order adaptative CFD solver for incompressible flows. *SoftwareX* **2020**, *12*, 100579–.

(28) Golshan, S.; Munch, P.; Gassmöller, R.; Kronbichler, M.; Blais, B. Lethe-DEM: An open-source parallel discrete element solver with load balancing. *Computational Particle Mechanics* **2022**, 1–20.

(29) El Geitani, T.; Golshan, S.; Blais, B. A High Order Stabilized Solver for the Volume Averaged Navier-Stokes Equations. *International Journal for Numerical Methods in Fluids* **2023**, 1–23.

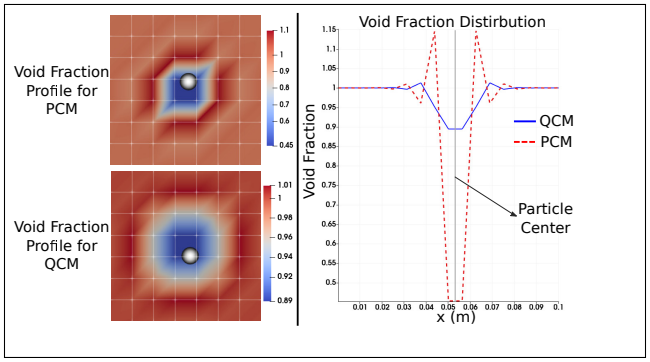
(30) El Geitani, T.; Golshan, S.; Blais, B. Toward High-Order CFD-DEM: Development and Validation. *Industrial & Engineering Chemistry Research* **2023**, *62*, 1141–1159.

(31) Di Felice, R. The voidage function for fluid-particle interaction systems. *International journal of multiphase flow* **1994**, *20*, 153–159.

(32) GIDASPOW, D. In *Multiphase Flow and Fluidization*; GIDASPOW, D., Ed.; Academic Press: San Diego, 1994; pp 1–29.

(33) Bokkers, G.; van Sint Annaland, M.; Kuipers, J. Mixing and segregation in a bidisperse gas–solid fluidised bed: a numerical and experimental study. *Powder Technology* **2004**, *140*, 176–186, 1st International Workshop on Granulation (Granulation across the length scales: linking microscopic experiments and models to real process operation).

TOC Graphic


Void Fraction Profile for PCM
Void Fraction Profile for QCM
Void Fraction Distirbution
Void Fraction
x (m)
QCM
PCM
Particle Center