Meso-scale coupling of FEM/DEM for fluid-particle interactions

S. Srivastava, K. Yazdchi and S. Luding*

Multi Scale Mechanics, MESA+ Institute for Nanotechnology, Faculty of Engineering Technology, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

Abstract

A new method for two-way fluid-particle coupling on an unstructured mesoscopically coarse mesh is presented. In this approach, we combine a (higher order) finite element method (FEM) on the moving mesh for the fluid with a soft sphere discrete element method (DEM) for the particles. The novel feature of the proposed scheme is that the FEM mesh is a dynamic Delaunay triangulation based on the positions of the moving particles. Thus, the mesh can be multipurpose: it provides (i) a framework for the discretization of the Navier-Stokes equations, (ii) a simple tool for detecting contacts between moving particles, (iii) a basis for coarse graining or up-scaling and (iv) coupling with other physical fields (viz. temperature, electromagnetic, etc.). This approach is suitable for a wide range of dilute and dense particulate flows, since the mesh resolution adapts with particle density in a given region. Two-way momentum exchange is implemented using semi-empirical drag laws akin to other popular approaches, e.g. the discrete particle method, where a finite volume solver on a coarser, fixed grid is utilized. We validate the methodology with several basic test cases, including single- and double-particle settling with analytical and empirical expectations, and flow through ordered and random porous media, as compared against finely resolved FEM simulations of flow through fixed arrays of particles.

Keywords: Multiphase flow; Multi-scale methods; Delaunay tessellation; Numerical analysis; FEM; DEM two-way coupling

* - Corresponding authors: K. Yazdchi (kyazdchi@gmail.com), S. Luding (s.luding@utwente.nl), Tel: +31534894212

1. Introduction

Fluid flow through particulate media is pivotal in many industrial processes, e.g. in fluidized beds, granular storage, industrial filtration and medical aerosols. Flow in these types of media is inherently complex and challenging to simulate, especially when the particulate phase is mobile. For the past two decades, particulate flows have been an active area of research and two widely used approaches are now considered state of the art. The first approach is based on an Eulerian continuum model of two phase flows, which only describes the averaged behavior of the multiphase media, see for example Kuipers et al. [1]. The second approach is based on an Eulerian-Lagrangian approach using finite volume/finite difference methods on a fixed grid as a fluid solver and either immersed boundary (IB) [2], fictitious domain (FD) [3], marker and cell (MAC) [4] or discrete element method (DEM) [5] for the particles. Both one-way and two-way couplings have been explored using these methods. While many fluid solvers are based on a staggered grid finite difference method, others e.g. Ladd [6, 7], Han et al. [8] and Feng et al. [2] have successfully utilized the lattice Boltzmann method (LBM) as a fluid solver for particle-fluid suspensions. The LBM is an attractive alternative due to its ease of implementation and parallelization; however, the selection of the lattice resolution, the collision kernel and accurate numerical implementation of various boundary conditions are still challenging [9].

A detailed description of flow through particulate media and accurate particle tracking can be obtained using discrete particle modeling (DPM) as proposed by Tsuji et al. [10], Kuipers et al. [11], Xu et al. [5] and Wu et al. [12]. In DPM, individual particles are tracked using Newton's laws of motion and particle-particle/wall interactions are taken into account on the (smallest) scale of the contacts. These models invariably couple a continuum solver for fluid with DEM, as originally proposed by Cundall & Strack [13], for particles. The coupling between fluid and particles is explicit and is achieved using semi-empirical drag laws or closure relations of fluid-particle interactions, e.g. Ergun et al. [14], Gidaspow [15], Drummond et al. [16], Gebart et al. [17]. In a recent study,

Yazdchi et al. [18, 19] proposed modified closure relations (drag laws) applicable to a wider range of porosities for both ordered and random porous media, valid for creeping two-dimensional flows. The DPM with hard sphere particle-particle interactions has been successfully applied to e.g. fluidized beds and slug formation in bubbly flows [20]. Those approaches represent a multi-scale method ranging from the smallest scale of the mechanical contacts via the micro-structure and particle-fluid interaction (drag laws) to the macroscopic scale of industrial applications. Todays challenges involve to develop methods that work for all ranges of densities (from dilute, Knudsen like regimes to dense sediments or even compressed particle packings) and keeping as much as possible the information of smaller scales (e.g. the micro-structure) available on the larger scales. For this the multi-scale, multi-phase model presented in this study features an intrinsic resolution of the fluid that adapts to the particle density and is of the order of the particle-particle distance.

Furthermore, for dense particulate flows, efficient contact detection in a DPM approach requires additional data structures and specialized algorithms adding to its computational overhead. On the other hand, the grid size for flow resolution is often coarse, i.e. they are several times bigger than the mean particle diameter. Thus, most DPM models ignore the sub-grid scale flow characteristics and this affects the small scale particle dynamics. Xu et al. [21] have recently proposed including sub-grid scale features to better capture the particle dynamics. However, fully resolved methods are often too slow to be able to reach macroscopic scales, so that in this study we propose as compromise to use the resolution of the (dynamically changing) particle distance. Note that this, like all the preceding methods [1-5], is based on explicit coupling between fluid- and particle-solvers through empirical drag relations. In contrast, in many finely resolved approaches an implicit coupling is present. For example the distributed Lagrange multiplier (DLM) method of Glowinski et al. [22] has been successfully applied to simulate fluid-particle interaction in porous media and fluidized beds. Due to an additional set of Lagrange multipliers, DLM is more computationally expensive than DPM. However, similar to DPM, the particles are not modeled geometrically in this approach, but the flow in the vicinity of particles is better resolved. Using DLM, Pan et al. [23] simulated the behavior of fluidized beds; however, they ignored particle-particle interactions to keep the

computational costs low. More recently, Kanarska et al. [24] have coupled DLM with DEM for particle-particle interactions. Fully resolved simulations of particle laden flows using FD by Avci et al. [3] is in spirit similar to DLM, except that coupling forces are computed by integrating the stress field at the surface of the particles. In essence, the two methods are exact as no drag correlations are required to couple the two phases.

Interest in using a deforming mesh for fluid-structure/particle interactions has persisted for some time now. Tezduyar et al. [25] developed the so-called Deforming Spatial Domain/Deforming Space Time (DSD/DST)-FEM for flow problems with deforming interfaces using the so-called arbitrary Lagrangian-Eulerian (ALE) methods and space-time finite element method. In this approach, particles are geometrically modeled in the mesh and the flow is fully resolved around each particle, hence leaving it as computationally expensive for dense flows [2].

In this paper, we introduce a new method for fluid-particle interaction based on a two-way coupling between a higher order FEM and a soft sphere DEM approach on a deforming unstructured mesh. The main feature of our approach is a deforming Delaunay triangulation, which is utilized as an efficient contact detection tool for the moving particles as well as a finite element mesh for discretizing the Navier-Stokes equation. It is known that the nearest neighbor property of the Delaunay edges renders it an attractive algorithm for contact detection, see Ferrez et al. [26] and references therein. To better resolve the flow around the particles, we apply the interaction forces as point forces at the particle locations or at their surface (for different version see Section 3.1). To our knowledge, this study is the first attempt to apply a moving Delaunay triangulation (particle based) for both contact detection and finite element fluid solver. Coupling with FEM as fluid solver has advantages; it allows to build upon very well developed numerical and mechanical methods and thus may provide the leverage of higher order interpolations for simulating flow to the desired accuracy and scales, even when the mesh is relatively coarse. Another motivation to use FEM is that for packed beds and dense particulate flows, the mesh can also be used as a coarse graining or up-scaling tool for stress and strain fields, which are quantity of interest for a macro-scale description of the particle-fluid system.

Despite the advantages of using FEM for higher accuracies, Wu et al. [12] have pointed out several issues associated with implementing fluid-particle coupling on an unstructured mesh. The most restrictive one pertains to computing the particle volume fraction in a given cell, since particles may be shared between neighboring cells and thereby add to the computational complexity. We circumvent this issue in the present methodology by resorting to a moving mesh and an ALE formulation. In this way, the particles have finite radius and always lie at the element vertices and consequently, due to their repulsive forces and excluded volume, a coarse mesh in moderately dense system generally remains robust with respect to element degeneration; in other cases, re-meshing is deployed whenever necessary, or additional features have to be added to the method. Like the particle-contacts, the particle-particle/wall interactions are modeled using a linear spring contact model with dissipation and friction, but the triangulation needs special attention in the vicinity of the walls in order to generate a "healthy" grid.

This paper is organized as follows. We start with an introduction to the mathematical model applicable to viscous, incompressible flow through an isotropic random or ordered porous media in Section 2. Also the drag force model used for coupling FEM and DEM and the contact force model used in DEM are discussed in details. In Section 3, we detail the underlying finite element formulation and discuss the methodology for approximating the porosity field and its impact on numerical computations. This is followed by numerical examples in Section 4, demonstrating the most basic flow situations in static and moving particulate media. Finally, Section 5 presents conclusions drawn in this paper and an outlook for future studies.

2. Mathematical model

The governing equation for the multiphase flow is a set of porosity scaled Navier-Stokes equations, which define the flow of fluid in a particulate porous media (see Anderson and Jackson [27], Deen et al. [28], Xu et al. [5]). Considering an incompressible fluid (i.e. the density, ρ is constant) in an Eulerian flow domain, Ω, we can write the equations of both fluid and solid phase as

Fluid phase:

$$\left.\begin{aligned} &\rho\frac{\partial\left(\varepsilon\vec{u}\right)}{\partial t}+\rho\nabla\cdot\left(\varepsilon\vec{u}\vec{u}\right)=-\varepsilon\nabla p+\nabla\cdot\left(\varepsilon\boldsymbol{\tau}\right)+\varepsilon\rho\vec{g}-\sum_i\vec{f}_i^{\,D}\\ &\boldsymbol{\tau}=\mu\left(\nabla\vec{u}+\left(\nabla\vec{u}\right)^T\right)-\frac{2}{3}\mu\left(\nabla\cdot\vec{u}\right)\mathbf{I}\\ &\frac{\partial\varepsilon}{\partial t}+\nabla.\left(\varepsilon\vec{u}\right)=0 \end{aligned}\right\}\ \text{in } \Omega \tag{1}$$

Solid phase:

$$\left.\begin{aligned} &m_i\frac{\mathrm{d}\vec{u}_i}{\mathrm{d}t}=\vec{F}_i^{\,D}+\sum_j\vec{F}_{ij}^{\,C}-V_i\nabla p+m_i\vec{g}\\ &I_i\frac{\mathrm{d}\vec{\omega}_i}{\mathrm{d}t}=\vec{T}_i^{\,D}+\sum_j r_i\hat{n}_{ij}\times\vec{F}_{ij}^{\,C} \end{aligned}\right\}, \tag{2}$$

where ε, μ, $\vec{u}$, p, $\boldsymbol{\tau}$ and $\vec{g}$ are the porosity, viscosity, fluid velocity vector, pressure, shear stress and the acceleration due to gravity, respectively. For the particles m, I_i, r_i, V_i, $\vec{u}_i$ and $\vec{\omega}_i$ represent particle mass, moment of inertia, radius, volume, translational and angular velocity, respectively. The $\vec{F}_{ij}^{\,C}$ represents the inter-particle/wall contact force and $\hat{n}_{ij}$ is the unit vector pointing from the center of the particle to the contact point (with particle *j*). Finally, $\vec{f}_i^{\,D}$ and $\vec{F}_i^{\,D}$ represent the drag force per unit volume on the fluid due to interaction with the i^{th} particle and the total drag force acting on the i^{th} particle, defined in the following section. In the angular momentum equation, $\vec{T}_i^{\,D}$ represents the torque experienced by the i^{th} particle due to fluid drag when flow around the particle becomes asymmetric, as shown in Section 4.2. The pressure gradient term in Eq. (2) accounts for the net buoyancy force on each particle acting on its center of mass. Since Eq. (2) is a system of ordinary differential equations in time, for good accuracy and conservation properties, we use the velocity-Verlet time integrator, which is

second order accurate in time[1]. Note that the indices i and j do not represent the tensorial components of respective fields in the above equation, instead i represents particle number and j is the index for the contacts of the i^{th} particle. In the rest of this section, we introduce the model for the drag force density, used to explicitly couple the fluid and particle dynamics.

2.1. Drag force model

The drag force accounts for the resistance to the flow through a porous media, and is inversely related to its permeability, K. The permeability is the proportionality constant in Darcy's equation

$$\vec{U} = \frac{-K\nabla p}{\mu} \ , \tag{3}$$

where U is the *superficial* (discharge) fluid velocity, relative to the particle speed, defined as

$$\vec{U} = \frac{1}{V}\int_{V_f} \vec{u}\,\mathrm{d}V = \varepsilon\left\langle \vec{u} \right\rangle \ , \tag{4}$$

where V and V_f are total available volume and the volume of fluid. On the other hand, the *intrinsic* average flow velocity $\left\langle \vec{u} \right\rangle = \frac{1}{V_f}\int_{V_f} \vec{u}\,\mathrm{d}V$ is defined only over the fluid volume. Following Yazdchi et al. [18], the permeability, K is related to the friction coefficient

$$\beta = \frac{\mu\varepsilon^2}{d^2\lambda} \ , \tag{5}$$

where $\lambda = K/d^2$ represents the non-dimensional permeability and is often used instead of K in literature. Several existing correlations for λ are listed in Table 1. Henceforth,

[1] Since the forces between particles can be dissipative the choice of an integrator does not have a major impact on either solution quality or the performance, thus will not be discussed in detail in this paper.

the drag force density in the fluid, $\vec{f}_i^{\,D}$ is defined at a point x_e, (see next section for details). The force density is modeled as

$$\vec{f}_i^{\,D} = \beta\left(\langle\vec{u}\rangle - \vec{u}_i\right)\psi\left(x - x_e\right), \tag{6}$$

where $\vec{u}_i$ is the instantaneous velocity of the i^{th} particle and ψ is a smoothing function describing the influence of the force density on its neighborhood. While for ψ several possibilities exist, e.g. a Gaussian function, in this paper we restrict ourselves to $\psi\left(x - x_e\right) = \delta\left(x - x_e\right)$, i.e. the Dirac delta function, for reasons that are discussed in the next section. Eq. (6) is a model of the drag force density in the fluid in the neighborhood of the particle. The drag from fluid to particle is proportional to the relative velocity between particle and fluid. In other words, a particle moving in the direction of the flow in its neighborhood with the average fluid velocity does not experience any drag.

Author	Porosity range	λ
Ordered arrays (square configuration)		
Drummond et al. [16]	$\varepsilon \geq 0.6$	$\lambda_D^s = \frac{1}{32\phi}\left(\ln\left(\frac{1}{\phi}\right) - 1.476 + \frac{2\phi - 0.796\phi^2}{1 + 0.489\phi - 1.605\phi^2}\right),\ \phi = 1-\varepsilon$
Gebart [17]	$\varepsilon < 0.6$	$\lambda_G^s = C\left(\sqrt{\frac{1-\varepsilon_c}{1-\varepsilon}} - 1\right)^{5/2},\quad C = \frac{4}{9\pi\sqrt{2}},\ \varepsilon_c = 1 - \pi/4$
Yazdchi et al. [18]	$0.16 \leq \varepsilon \leq 0.95$	$\lambda_Y^s = \lambda_{G2}^s + \frac{\left(\lambda_D^s - \lambda_{G2}^s\right)}{2}\left(1 + \tanh\left(\frac{\varepsilon - 0.75}{0.037}\right)\right),\ \lambda_{G2}^s = \frac{\lambda_G^s}{1 + 0.336\left(\varepsilon - 0.21\right)}$
Random arrays		
Ergun [14]	$0.4 \leq \varepsilon \leq 0.8$	$\lambda_E^r = \frac{\varepsilon^3}{150\left(1-\varepsilon\right)^2}$
Yazdchi et al. [19]	$0.16 \leq \varepsilon \leq 0.95$	$\lambda_Y^r = \lambda_G^r + \frac{\left(\lambda_{CD}^r - \lambda_G^r\right)}{2}\left(1 + \tanh\left(\frac{\varepsilon - 0.67}{0.1}\right)\right),\ \lambda_{CD}^r = 0.97\lambda_D^r\left(1 + 0.18\varepsilon\right),$ $\lambda_D^r = \frac{1}{32\phi}\left(-\ln\left(\phi\right) - 1.497 + 2\phi - \frac{\phi^2}{2} - 0.739\phi^4 + \frac{2.534\phi^5}{1 + 1.2758\phi}\right),\ \phi = 1-\varepsilon,$ $\lambda_G^r = 0.035\left(\sqrt{\frac{0.84}{1-\varepsilon}} - 1\right)^{5/2}$

Table 1: Different non-dimensional permeabilities for monodisperse systems as a function of porosity (ε) and particle diameter (d) in the creeping flow (Re = 1) regime.

Ergun's equation[2] is a commonly used drag law, which is a non-linear function of porosity, fluid velocity and particle size. It accurately predicts the total drag force for a limited range of porosities in 3D. Using this relation, one can derive the macroscopic permeability of the media and use Darcy's equation to determine the average flow velocity through the media. An aptly modified version of this equation applicable in 2D is deployed as suggested in [19]. Accordingly, a more general form of β, taken directly from Ergun et al. [14], applicable towards inertial regimes can be written as

$$\beta = \frac{\mu\varepsilon^2}{d^2\lambda} + 1.75\rho\frac{\left(1-\varepsilon\right)\left(\left\langle\vec{u}\right\rangle - \vec{u}_i\right)}{d} \quad , \tag{7}$$

but we do not follow this further and use instead the laminar relation by Yazdchi et al. [18] due to its widest regime of valid porosities. Having specified the drag law, in the following, we introduce a simple contact force model to account for inter-particle/wall forces.

2.2. Contact force model

We take into account the particle-particle/wall interactions and therefore, the contact forces are essential in order to integrate the particles equations of motion. As elsewhere [31], we use a linear spring-dashpot model for the contact force

$$\vec{F}_{ij}^{C} = \kappa\delta_{ij}^{p}\hat{n}_{ij} + \eta\vec{v}_{ij} \, , \tag{8}$$

where $\kappa, \eta, \vec{v}_{ij}$ and δ_{ij}^{p} are contact stiffness, viscous damping coefficient, relative velocity between particle i and j and the overlap, respectively. A similar model can also be implemented in the tangential direction along with a sliding spring based on tangential overlap, for cases where rotation and friction are relevant (but is not used in this paper).

[2] Ergun equation is essentially a correction to the Carman-Kozeny [29] drag relation for creeping flows, which also takes into account the inertial drag at higher Reynolds numbers [30].

The contact stiffness, κ and overlap, δ_{ij}^{p} set a limit value for the DEM time step as $\Delta t_{\mathrm{DEM}} \cong \frac{1}{50}\pi/\varpi$ and $\varpi = \sqrt{2\kappa/m - (4\eta/m)^2}$ for numerical simulations. A particle may also have more than one contact at any given time, in this case the total contact force is found by summing over all the contacts. For further details and state of the art in DEM contact models, see the review paper by Luding [31] and references therein.

3. Finite element formulation

Let us assume we have suitably defined discrete finite element (polynomial) spaces V^h, S^h for trial and test solutions and let $\vec{u}^h$, p^h denote the trial solution of Eq. (1). Further, we divide our domain Ω into non-overlapping triangles Ω_k such that $\mathrm{U}_k \Omega_k = \Omega$. The weak form is obtained by multiplying the Eq. (1) with appropriate test functions ($\vec{v}^h$, q^h) and performing integrating by parts on the diffusion term. This yields a mixed Galerkin formulation for ($\vec{u}^h$, p), which reads as

Find $\left(\vec{u}^h, p^h\right) \in V^h \times S^h$ such that $\forall \left(\vec{v}^h, q^h\right) \in V^h \times S^h$,

$$\begin{aligned} &\rho\left(\frac{\vec{u}^h \partial\varepsilon}{\partial t} + \frac{\varepsilon \partial \vec{u}^h}{\partial t}, \vec{v}^h\right) + \left(\varepsilon\left(\vec{u}^h - \vec{u}_M\right)\vec{u}^h, \nabla\vec{v}^h\right) + \left(\varepsilon\boldsymbol{\tau}^h, \nabla\vec{v}^h\right) = \left(\varepsilon p^h, \nabla\cdot\vec{v}^h\right) - \left(\vec{f}_i^{\,D} - \varepsilon\rho\vec{g}, \vec{v}^h\right) + \sum_{\Omega_k}\int_{\Omega_k}\left(\mathrm{S.T.}\right), \\ &\left(-\frac{\partial\varepsilon}{\partial t}, q^h\right) = \left(\varepsilon\nabla\cdot\vec{u}^h + \vec{u}^h\cdot\nabla\varepsilon, q^h\right) \end{aligned} \quad (9)$$

where $(x, y) = \int_{\Omega} xy \mathrm{d}\Omega$ denotes the standard inner product on V^h and S^h. Notice that for the moving mesh we replace the convection velocity $\vec{u}^h$ by ($\vec{u}^h - \vec{u}_M$), where $\vec{u}_M$ is the mesh velocity in ALE formulation and essentially takes into account the convection of the fluid momentum due to mesh motion. To compute $\vec{u}_M$ at quadrature points inside a triangle, we interpolate the velocities of the nodal particles of that triangle. Using nodal velocities ensures that the Geometric Conservation Law (GCL) for the ALE formulation is satisfied [32, 33], since a constant solution is reproduced trivially. The above

formulation requires a-priori knowledge of the porosity field at every point inside the domain, which can be computed using SPH interpolation, see Section 3.2.

For additional robustness and stability in our formulation, we add streamline-upwind/Petrov Galerkin (SUPG), pressure stabilized/Petrov Galerkin (PSPG) and other terms similar to least square incompressibility constraint (LSIC) as discussed in [34], to the above variational formulation. Forster et al. [35] have investigated that such stabilization is also effective when simulating on distorted meshes. Henceforth, following [25] we add residual based stabilization terms (S.T.) given by

$$\text{S.T.} = \left[\boldsymbol{\tau}_{SUPG} \vec{u}^h \cdot \nabla \vec{v}^h + \frac{1}{\rho} \boldsymbol{\tau}_{PSPG} \nabla q^h \right] \cdot \vec{r}_u \left(\vec{u}^h, p^h \right) + \boldsymbol{\tau}_{LSIC} \left(\nabla \cdot \vec{v}^h \right) r_p \left(\vec{u}^h \right), \tag{10}$$

where $\vec{r}\left(\vec{u}^h, p^h\right)$ denotes the residual of continuity equation, Eq. (1)

$$\begin{aligned} \vec{r}_u \left(\vec{u}^h, p^h \right) &= \rho \left(\frac{\partial \left(\varepsilon \vec{u}^h \right)}{\partial t} + \nabla \cdot \left(\varepsilon \vec{u}^h \vec{u}^h \right) \right) + \varepsilon \nabla p - \nabla \cdot \left(\varepsilon \boldsymbol{\tau}^h \right) - \varepsilon \rho \vec{g}, \\ r_p \left(\vec{u}^h \right) &= \frac{\partial \varepsilon}{\partial t} + \nabla \cdot \left(\varepsilon \vec{u}^h \right) \end{aligned} . \tag{11}$$

The stabilization parameters are fixed using the following expressions

$$\begin{aligned} \boldsymbol{\tau}_{SUPG} = \boldsymbol{\tau}_{PSPG} &= \left[\left(\frac{2}{\Delta t} \right)^2 + \left(2 \frac{\left\| \vec{u}^h \right\|}{h_e} \right)^2 + \left(4 \frac{\mu}{\rho h_e^2} \right)^2 \right]^{-1/2}, \\ \boldsymbol{\tau}_{LSIC} &= \frac{h_e}{3} \left\| \vec{u}^h \right\| \end{aligned} \tag{12}$$

where h_e is the length of the smallest edge of the element. Eq. (12) has shown to be a convenient choice in computations [34].

Stable discretization of Eq. (9) can be difficult to construct and solutions are well studied in literature, see [36] for a detailed theory. Classical methods forbid equal interpolation of both velocity and pressure variables in the above setting. Stable solutions can usually be obtained if $P_p \subset P_u$ (polynomial spaces for p and u) in numerical approximations.

Furthermore, it is known that the incompressibility constraint is not strongly enforced when using a continuous approximation for the pressure field [37]. To circumvent this problem, we adopt a discontinuous polynomial space for pressure discretization. In this paper, unless stated otherwise, we choose stabilized P1/P0 or P2/P1 elements with continuous velocities and discontinuous pressure polynomials. However, this formulation is not restricted at all in choosing higher order FE spaces.

3.1. The mesh and drag force computation

The FE mesh adapted in the above formulation is a Delaunay triangulation based on the particle locations. This implies that all interior vertex nodes of the mesh are occupied by particles at all times, while the boundary nodes are inserted only for the convenience of computation and application of boundary conditions, see Fig. 1.

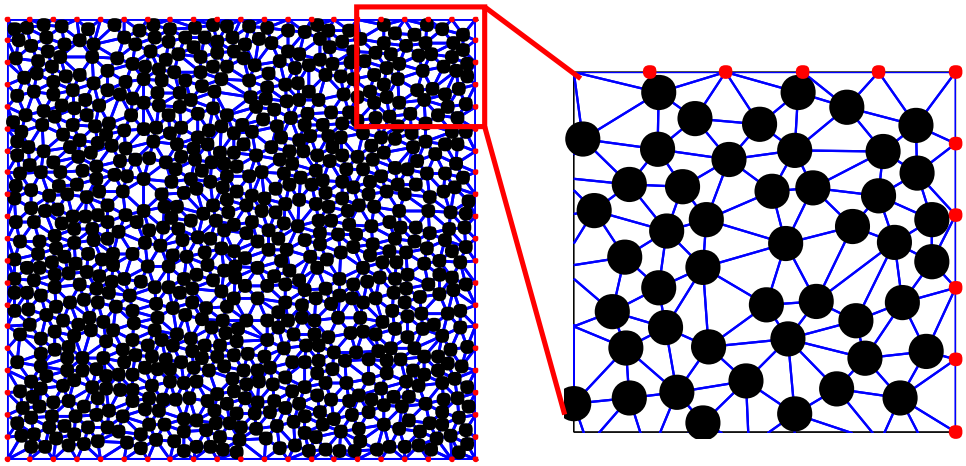


Figure 1: Finite element mesh based on 800 randomly distributed particles at porosity $\varepsilon = 0.6$. (Right) Complete Mesh; (Left) Zoomed in section at the upper right corner, which shows the added boundary nodes (red points) to define the geometry, spaced at equal distances of about twice the particle diameter (i.e. ~ 2d).

For moving particles the mesh vertices move with the particles, thereby deforming the mesh. Currently, we re-mesh at fixed (short) time intervals in order to maintain the quality of triangles and also use the triangulation for contact detection. This implies that a

new triangulation is created from the current particle positions and the solution from the old mesh is transferred to the new mesh using a simple projection scheme. To remain focused, we will not discuss the projection scheme in detail.

In Fig. 2(a) the particle overlaps with an element are shown for particles of different sizes. While we do not address polydisperse particles in a fluid flow in this paper, different size particles are shown to highlight the generality of the proposed method. Fig. 2(b) shows the drag force contribution from each element. The total drag force and torque acting on the i^{th} particle is considered as a sum of contributions from all the overlap elements

$$\vec{f}_e^{\,D} = \frac{1}{A_e}\sum_{p=1}^{3}\vec{F}_e^{\,p}\delta\left(x-x_e\right), \quad \vec{F}_p^{\,D} = \sum_{e=1}^{E_e}\vec{F}_e^{\,p}, \quad T_p^{D} = \sum_{e=1}^{E_e} r_p\hat{n}\times\vec{F}_e^{\,p}, \tag{13}$$

where e is the index counting the number of triangular overlaps, E_e of the i^{th} particle (for example E_e=5 in Fig. 2(b)).

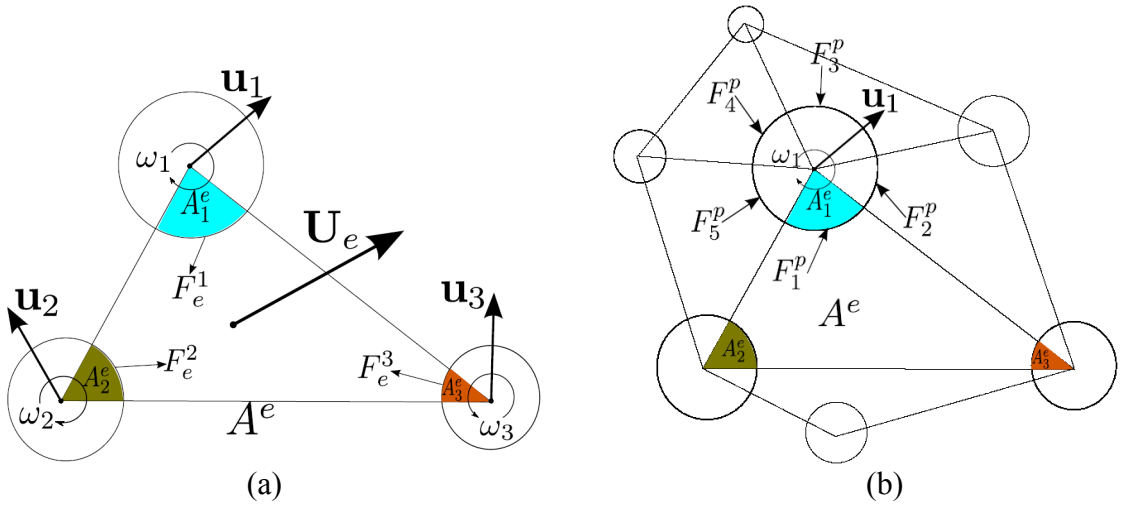


Figure 2: (a) An element from the mesh is shown with the 3 particles occupying its vertices. The particle translational and angular velocities are represented by u_1, u_2, u_3, ω_1, ω_2, ω_3, respectively. U_e represents the superficial (relative) velocity in the cell. A^e represents the area of the element e and A_i^e the area of the respective overlaps. (b) The drag force contributions to a particle from all its neighboring/touching fluid elements e are shown as arrows with F^p_e.

An important modeling aspect from the numerical point of view is the location of the drag forces, x_e computed from Eq. (6). Here we list a few possibilities for application of the $\vec{f}_i^D$ (as shown in Fig. 3):

(a) At the mid point of the chord of the respective overlaps;
(b) At the mid point of the arc in respective overlaps (as in Fig. 2);
(c) At the intersection of particles circumference with element edges;
(d) At the nodal location of the respective particle.

Fig. 3 shows several possible sites for the application of the drag force. Unless specified otherwise, for simplicity, we choose the mid point of the chord (i.e. Fig. 3(a)), as it lies close to the fluid solid interface, where the momentum exchange occurs. We did not experience an impact of the choice of application of force on the numerical results for the various cases tested, however, this issue remains a task for future studies.

A force equal in magnitude but opposite in direction is applied to the fluid, i.e. $\vec{F}_e^1 = -\vec{F}_1^p$, at exactly the same point in the cell, thereby providing a consistent point-force based coupling. The total force on the particle due to the fluid also contains the buoyancy force, which is computed based on the local pressure gradient.

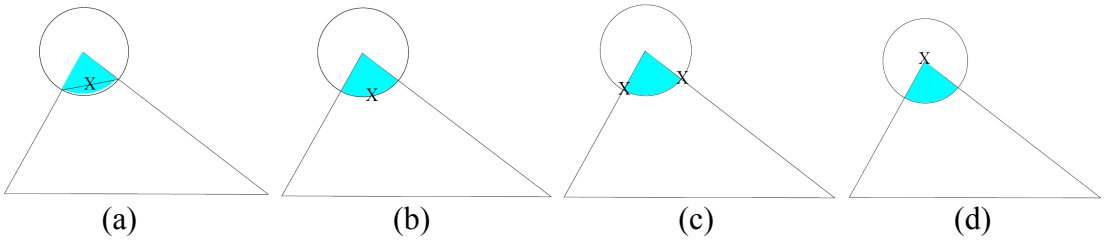


Figure 3: Point of application of the drag force; (a), (b), (c) and (d) show four distinct possibilities (marked with 'X') for x_e.

3.2. Local porosity calculation

At this point the general variational form, i.e. Eq. (9), can be solved using various assumptions for the porosity field. If the particles are fixed and are relatively

homogeneously distributed, one can simplify Eq. (9) by making the assumption that the porosity is a constant for $\forall\Omega_e$ and there is no temporal variation. Thus, for a locally averaged formulation, one could take a simpler approach and define a porosity for each triangle in the mesh, see Fig. 2, as

$$\varepsilon^e = 1 - \frac{\sum_{i=1}^{3} A_i^e}{A^e} \,. \tag{14}$$

Although Eq. (14) is computationally efficient and simple to compute, this definition may lead to high fluctuations in the porosity field, thereby adding to the numerical instabilities especially for dynamic meshes. Therefore, in this paper, we utilize Eq. (14) only for static particles. To remedy this issue, we interpolate the particle number density using a smooth particle hydrodynamics (SPH) kernel function as given by

$$W\left(\vec{r},h\right) = \frac{4}{\pi h^8}\begin{cases}\left(h^2 - r^2\right)^3 & 0 \le r < h \\ 0 & r \ge h\end{cases}, \tag{15}$$

where h is the smoothing length. Following Xu et al. [21] one can evaluate the porosity and its gradient at an arbitrary point r as

$$\begin{aligned}\varepsilon\left(\vec{r}\right) &= 1 - \frac{\pi}{6} d^2 \sum_j W\left(\left|\vec{r} - \vec{r}_j\right|, h\right) \\ \nabla\varepsilon\left(\vec{r}\right) &= \frac{\pi}{6} d^2 \sum_j \nabla W\left(\left|\vec{r} - \vec{r}_j\right|, h\right)\end{aligned} \,. \tag{16}$$

This definition yields a smoother porosity field on a length-scale h that can be considerably larger than the particle diameter; however, it incurs additional computation at each numerical quadrature point. Furthermore, special attention is required at the boundaries, e.g. Shepard correction is required [38], but details are to be studied in the future.

The issue of the validity of Eq. (1) for different length-scales is another open issue for future research. Note that a SPH-DEM code, as introduced in Ref. [38] was working reliably well for smoothing lengths as small as $h \sim 2d$. Furthermore, the relevance and

validity of drag laws like Erguns' in the case of moving particles and for rather high resolution is an open issue. For a preliminary study on the local field quantities for different resolution see Ref. [40], which considered only static particles but provides vast details on the local and non-local flow behavior across different scales.

3.3. Time integration

After performing spatial integration, a second order finite difference scheme is utilized for time integration of the resulting system of equations. In a general form this can be written as

$$\ddot{\vec{u}} = \frac{\partial \vec{u}}{\partial t} = \frac{3\vec{u}^{n+1} - 4\vec{u}^{n} + \vec{u}^{n-1}}{2\Delta t}. \tag{17}$$

Using the necessary polynomial approximations of test and trial functions, the finite element matrices for each element in the mesh are assembled and the algebraic form of the equations is written as

$$\begin{matrix} \left[M\right]\left\{\ddot{\vec{u}}^{h}\right\} + \left[C\left(\vec{u}^{h}\right)\right]\left\{\vec{u}^{h}\right\} - \left[B\right]^{T}\left\{p^{h}\right\} + \left[A\right]\left\{\vec{u}^{h}\right\} = \left\{\vec{f}\right\} \\ \left[B\right]\left\{\vec{u}^{h}\right\} + \left[\gamma_{i}\right]\left\{p^{h}\right\} = \left\{\vec{f}_{\varepsilon}^{n}\right\} \end{matrix}, \tag{18}$$

where [*M*] represents the mass matrix, [*C*] is the matrix representing the convection term and [*B*] and [*A*] are the matrices due to pressure gradient and diffusion terms. The $\left[\gamma_{i}\right]$ matrix is due to pressure penalty terms on interior boundaries. The terms in {.} denote the corresponding coefficients of the FE solution. We discretize in time using a second order scheme, i.e. Eq. (17), and the θ-method (Crank-Nicolson method with $\theta = 0.5$) for linearizing the convection term as

$$\begin{matrix} \left(3\left[M\right] + 2\Delta t\theta\left[C^{n}\right] + 2\Delta t\left[A\right]\right)\left\{\vec{u}^{h}\right\}^{n+1} - 2\Delta t\left[B\right]^{T}\left\{p^{h}\right\}^{n+1} = \\ 2\Delta t\left\{\vec{f}\right\} + \left(4\left[M\right] + 2\Delta t\left(1-\theta\right)\left[C^{n}\right]\right)\left\{\vec{u}^{h}\right\}^{n} - \left[M\right]\left\{\vec{u}^{h}\right\}^{n-1} \\ \left[B\right]\left\{\vec{u}^{h}\right\} + \left[\gamma_{i}\right]\left\{p^{h}\right\} = \left\{\vec{f}_{\varepsilon}^{n}\right\} \end{matrix}, \tag{19}$$

where $\{f\}$ represents the sum of the forces and explicit RHS terms. This implies that the drag forces are explicitly calculated. A suitable time-step size for the FEM is chosen according to Courant–Friedrichs–Lewy (CFL) condition and the DEM time-step is computed based on the natural frequency of particle contacts. In order to allow that at every fluid-time step n DEM time steps are performed, where the integer $n = \Delta t_{FEM} / \Delta t_{DEM}$ is specified as input parameter together with Δt_{DEM}.

4. Numerical results

In this section numerical results will be presented for both verification and – to some extend – validation of the code. The computational framework described in the previous section will be used to simulate several basic test cases for both static and moving particles. In the following subsection, we first present results for static particles before continuing with deforming mesh simulations. The application to a real complex experimental situation is not shown in this study.

4.1. Static particles

This subsection deals with flow through static porous media for both ordered and disordered cases. The first example is a simplified model of flow through a homogeneous porous media, which verifies the compatibility between the present model and Darcy's law. In the second example, we compare our mesoscale resolution simulation with the average velocities obtained from fully resolved ANSYS simulations of flow through both ordered and disordered arrays of static particles [18, 19]. The fully resolved simulations were performed using a fine mesh with ~10^5 elements to accurately capture particle geometry and predict the flow around each particle. Our mesoscale approach, in contrast, contains elements of the same order of the number of particles (i.e. only a few hundreds). While the flow is not fully resolved, the comparison reassures that this scheme efficiently computes average velocities that are in the expected range and capture qualitatively the flow behavior at the macroscale while keeping details on the mesoscale.

4.1.1. Case 0: Homogeneous porous media and Darcy flow

A well defined multiphase model should also reproduce the limit behavior of single phase flow. When combined with a homogenized drag force (as opposed to point forces), the model must reproduce flow predictions from Darcy's law. As a preliminary verification case, we simulate the flow through a simple porous media using our formulation with a homogeneous body force (drag) in the test domain. We compare the average flow velocity from simulation with analytical results from Darcy's law. Recall that the permeability, K of the media describes the resistance to the flow and is intrinsically related to the drag coefficient, β via Eq. (5). Substituting Eq. (5) into (3) leads to

$$\varepsilon\left\langle \vec{u} \right\rangle = \vec{U} = -\frac{\varepsilon^2 \nabla p}{\beta}. \tag{20}$$

Setting $\beta = 1$ [kg/(m^{3}s)], $\varepsilon = 0.5$ and $\nabla p = -1$ [kg/(m^{2}s^2)], one obtains $\left\langle \vec{u} \right\rangle = 0.5$ [m/s]. For this special case, Eq. (1) can be simplified to $\frac{\mathrm{d}\vec{u}}{\mathrm{d}t} = -\frac{1}{\rho}\left(\nabla p + \frac{\beta}{\varepsilon}\vec{u} \right)$, with $\rho = 1$ [kg/m^3]. Assuming that the fluid is initially at rest $\left(\vec{u}\left(0\right) = 0 \right)$, the analytical, transient solution of the above equation is

$$\left\langle \vec{u}\left(t\right) \right\rangle = \frac{\varepsilon \nabla p}{\beta}\left(-1 + e^{\frac{-\beta}{\varepsilon\rho}t} \right). \tag{21}$$

We also numerically solve Eqs. (1) and (2), where $\vec{f}^{\,D} = \beta\left\langle \vec{u} \right\rangle$ acts as the distributed body force with stress-free boundary conditions. The problem setup is sketched in Fig. 4(a), and Fig. 4(b) shows that the flow quickly achieves a steady state value of $\left\langle \vec{u} \right\rangle = 0.5$ [m/s], in perfect agreement with the analytical solution above.

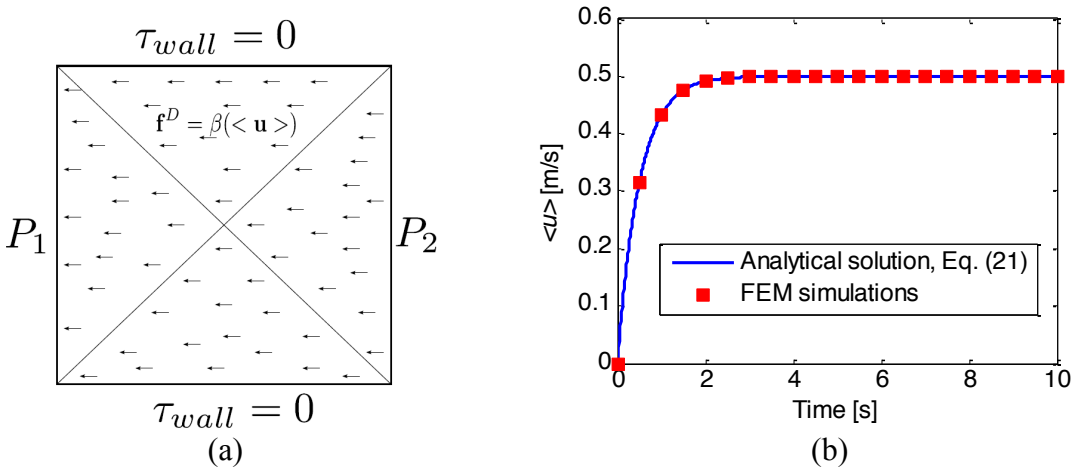
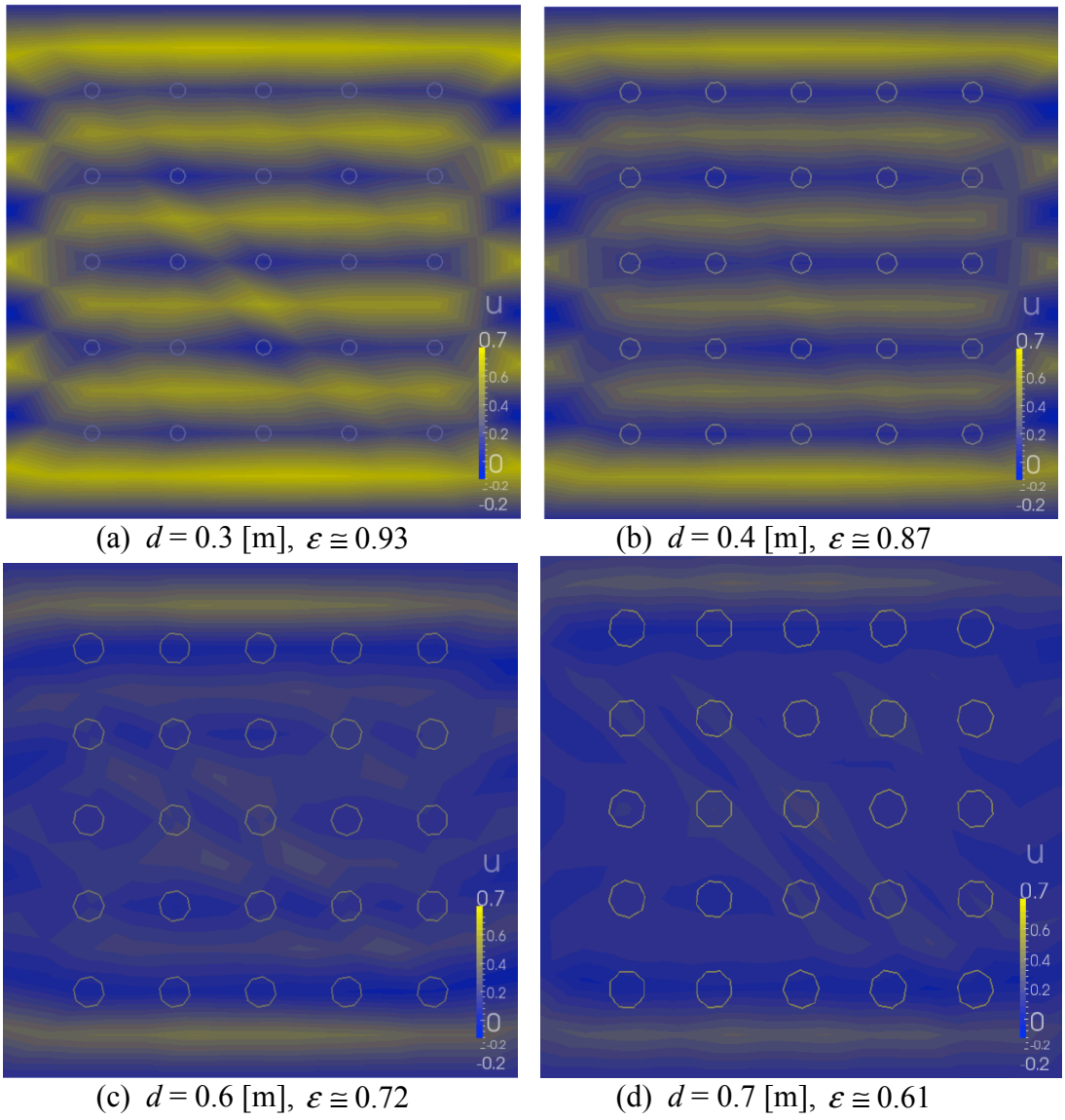


Figure 4: Darcy's homogenous flow case: (a) Setup with a coarse mesh (4 triangular elements), where the flow is driven by a pressure gradient in *x* direction. The arrows depict the homogenously smeared out resistive body force $\vec{f}^D$; (b) For the given parameters, the simulation predicts the correct transient and steady state average velocity.

4.1.2. Case 1: Flow through ordered and random porous media

For this problem in a square domain, the top and the bottom boundaries have no-slip boundary conditions, while the left and right boundaries maintain a pressure gradient of 5 [kg/(m^{2}s^2)]. In Figs. 5 and 6, the colors refer to the horizontal velocity in the ordered and random media, respectively. The blue regions indicate the slow flow region between the horizontal rows in the 5×5 particles array, while the predominant channel for the bulk flow lies between the two adjacent rows of particles. With decreasing porosity, the flow gradually confines itself between the walls and the top and bottom rows of particles as the interior becomes less and less permeable.



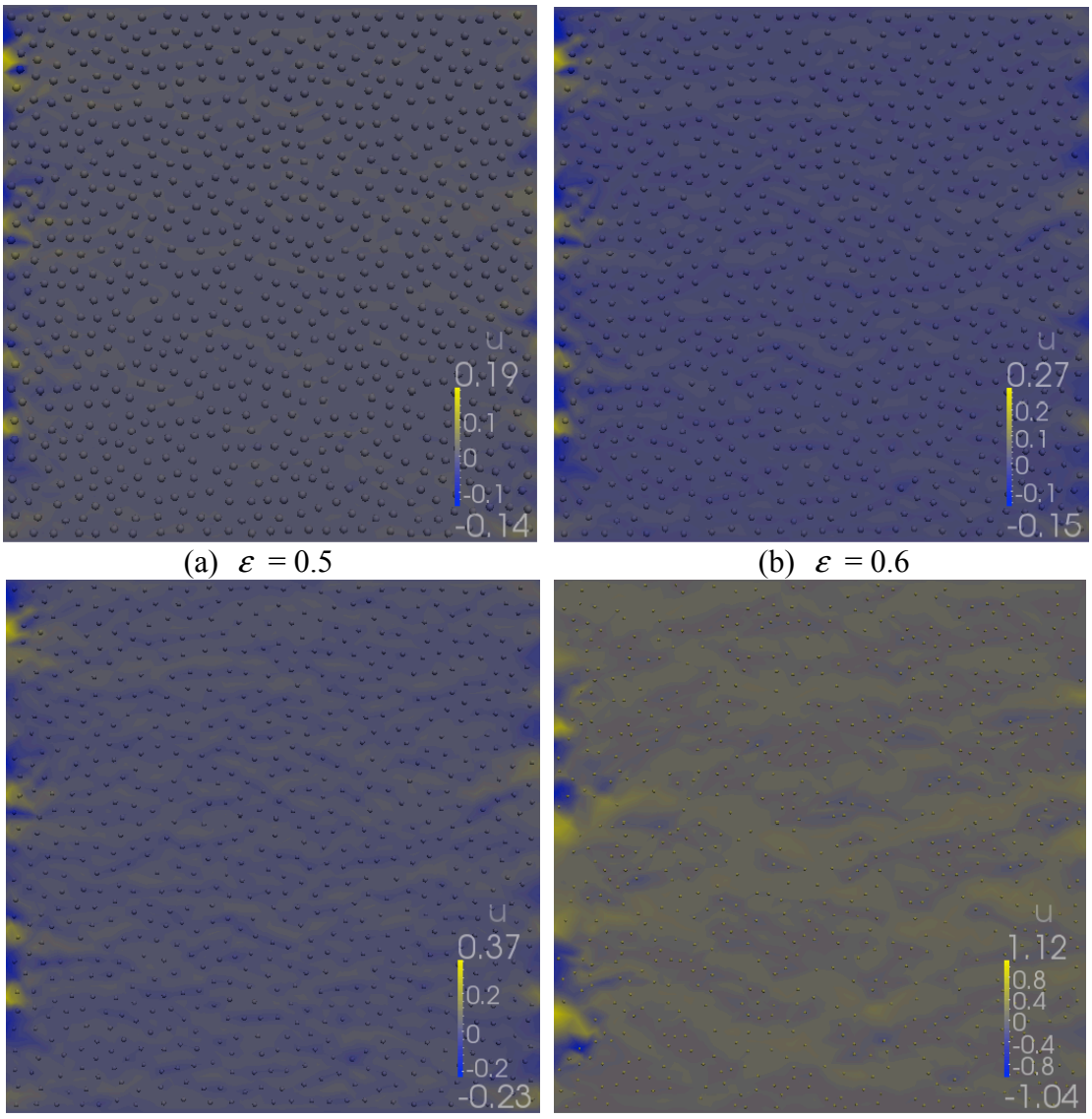
(a) $d = 0.3$ [m], $\varepsilon \cong 0.93$ (b) $d = 0.4$ [m], $\varepsilon \cong 0.87$

(c) $d = 0.6$ [m], $\varepsilon \cong 0.72$ (d) $d = 0.7$ [m], $\varepsilon \cong 0.61$

Figure 5: Horizontal velocity contours for ordered arrays (square configuration) of particles at different diameters, d, with ε given above.

For comparison purposes the average flow velocity is computed for the entire domain and compared with finely resolved FEM simulations. We utilized the drag law of Yazdchi et al. [18], from Table 1, in this simulation, which is valid for a wide range of porosities. The average flow predictions for both ordered and random cases agree very well with data from finely resolved FEM simulations (see Fig. 7). We must mention here that the fully resolved simulation is geometrically correct, i.e. particles are represented by holes

with no-slip boundary conditions and contains more than 10^5 degrees of freedom (dof). Our simulation, in contrast, relies on a few hundred dofs only.



(a) $\varepsilon = 0.5$ (b) $\varepsilon = 0.6$

(c) $\varepsilon = 0.7$ (d) $\varepsilon = 0.8$

Figure 6: Horizontal velocity contours in a random, homogeneous porous media with 800 randomly distributed particles at different porosities as given below the figures.

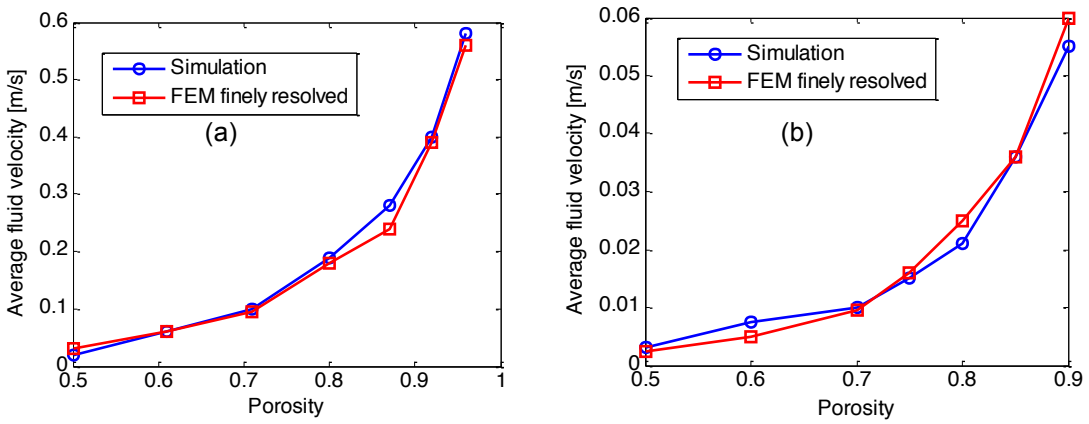


Figure 7: Average horizontal fluid velocity plotted against porosity through (a) ordered (square) and (b) random fibre arrays.

4.2. Moving particles

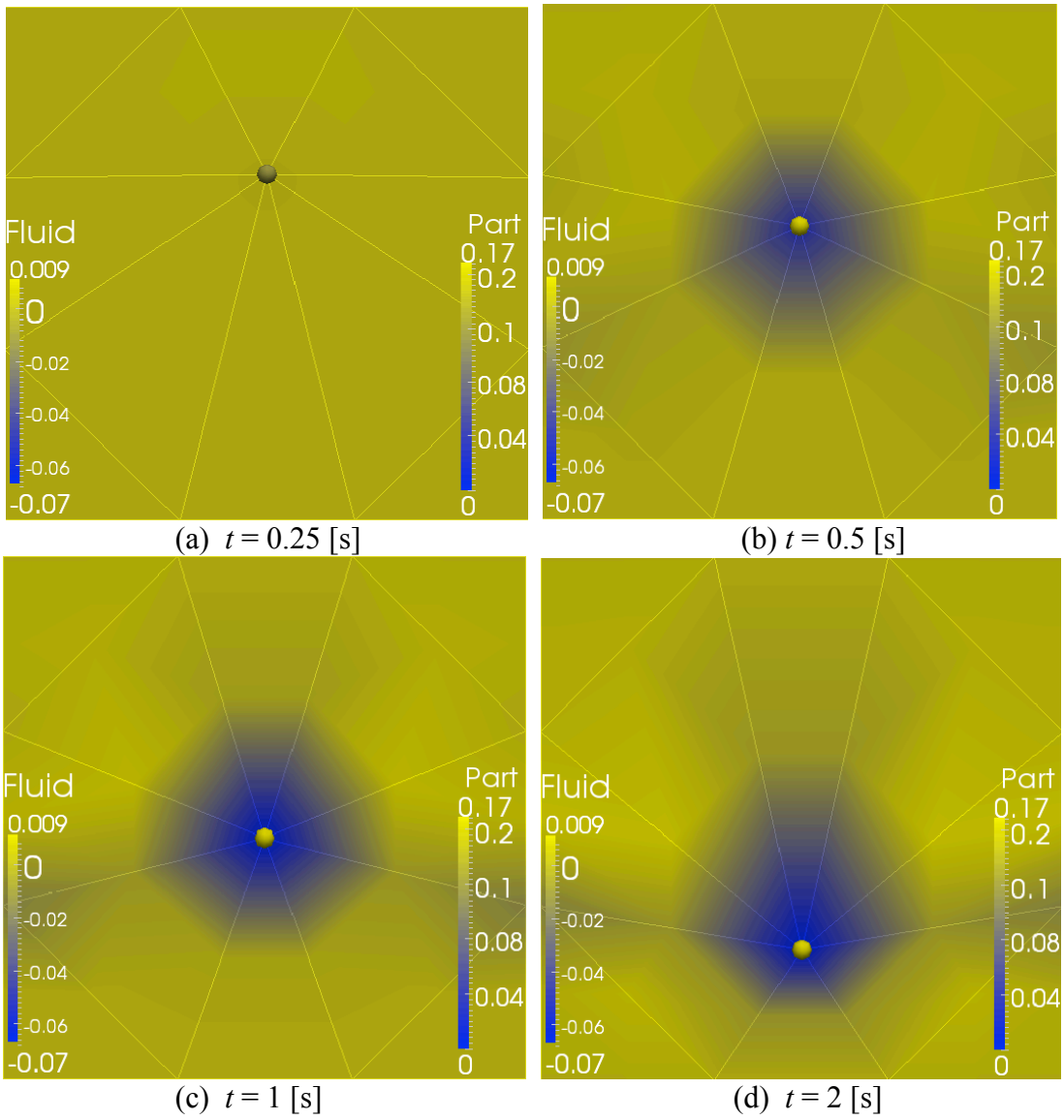
For the case of moving particles, e.g. fluidized beds, the underlying grid deforms as the particles (and mesh-nodes they occupy) move. This is an important feature of our methodology, since the particle positions are known at all times, it reduces the computational overhead associated with finding particles inside the correct cell [12]. For verification, we present two test cases of one and two particle sedimentation. To circumvent the solution degeneracy due to the deforming mesh, we re-mesh at fixed intervals. Re-meshing is essential in this approach since we wish to preserve the nearest neighbor property characteristic of the Delaunay triangulation for contact detection at all times. However, this is not too restrictive as the particles do not move much per Δt_{FEM} time step and contact detection with walls is handled separately in our code. Therefore, we do not address the particles escaping the fluid flow region in the present work, which remains a limitation to address in future work.

4.2.1. Case 1: Single particle settling

A particle under gravity in a viscous fluid, both initially at rest, will fall until it has reached the settling/terminal velocity, u_s calculated using the drag law prescribed in [39].

The parameters are $\mu = 1.14$ [kg/(m.s)], $\rho = 1.25\times10^{3}$ [kg/m^3], $\rho_p = 7.74\times10^{3}$ [kg/m^3], $d = 4.8\times10^{-3}$ [m] with drag force $\vec{f}^{D} = 4\pi\mu\vec{u}_s / \ln\left(7.4/\mathrm{Re}\right)$, and $\mathrm{Re} = \rho\vec{u}_s d/\mu$.

No slip boundary conditions are used at the top and bottom walls, while friction-less (no shear stress) boundary conditions are used along the left and right walls. The particle is released from $Z_0 = 0.6H$ [m], where $H = 2$ [m] is the height of the box. The mesh is based on the single particle location (corner points and two additional boundary points on each wall) and consists of only 12 triangular elements, which is rather coarse. As mentioned before, here, we switch to 4th order polynomials for an increased flow resolution. The settling velocity can be computed when the frictional force, $\vec{f}^{D}$, combined with the buoyancy force exactly balance the gravitational force ($m\vec{g}$) and is equal to $\vec{u}_s \simeq 0.17$ [m/s]. Fig. 8 shows the deforming mesh as the particle follows its trajectory. Near the particle surface a halo region with non-zero upwards fluid velocity appears due to the drag exerted by the falling particle. A trail of this halo is not evident since viscosity is large and our approach does not fully resolve the flow. Note that for this particular case no re-meshing was required as the mesh does not entangle throughout the simulation. The single-particle settling was studied in the context of another meso-scale SPH-DEM coupled method in Ref. [38] in much more detail and we refer to that study rather than testing different particle sizes and different fluid properties here.



(a) $t = 0.25$ [s] (b) $t = 0.5$ [s]

(c) $t = 1$ [s] (d) $t = 2$ [s]

Figure 8: Deforming mesh with velocity contours for 1 particle settling using 4th order basis functions. The velocity of the falling particle quickly attains its settling velocity.

4.2.2. Case 2: The Drafting, Kissing, Tumbling (DKT) problem

We illustrate another benchmark case, where two particles are initially separated vertically and start falling under gravity. As in the previous case, both fluid and particles are initially at rest and particles are then released. The simplest Stokes drag law, i.e. $\vec{f}^{D} = 3\pi\mu d\vec{u}_s$, with $\mu = 10^{-3}$ [kg/(ms)], $\rho = 10^{3}$ [kg/m^3], $\rho_p = 1.01\times10^{3}$ [kg/m^3] and $d = 4\times10^{-3}$ [m] is used in the simulation.

Similar to the previous example, no-slip boundary conditions are used at top and bottom walls, while friction-less (no shear stress) boundary conditions are used on the left and right walls. Fig. 9 depicts several snapshots of the 2 particle settling behavior. While the bottom particle center is aligned with the centerline of the box, the top particle's center location is offset to the centerline by 1% to the right to trigger the instability (a smaller offset would do the same job; if no offset is used the flow situation is determined by numerical errors). As the particles fall through the column of this fluid the top particle is observed to draft behind the first particle and catches up with the first particle (kissing) and then gets past it with a tumbling behavior. This behavior is very sensitive to flow resolution around the particles as the draft of one particle affects the other. This behavior is well captured in using 3rd order polynomials for fluid resolution in this approach.

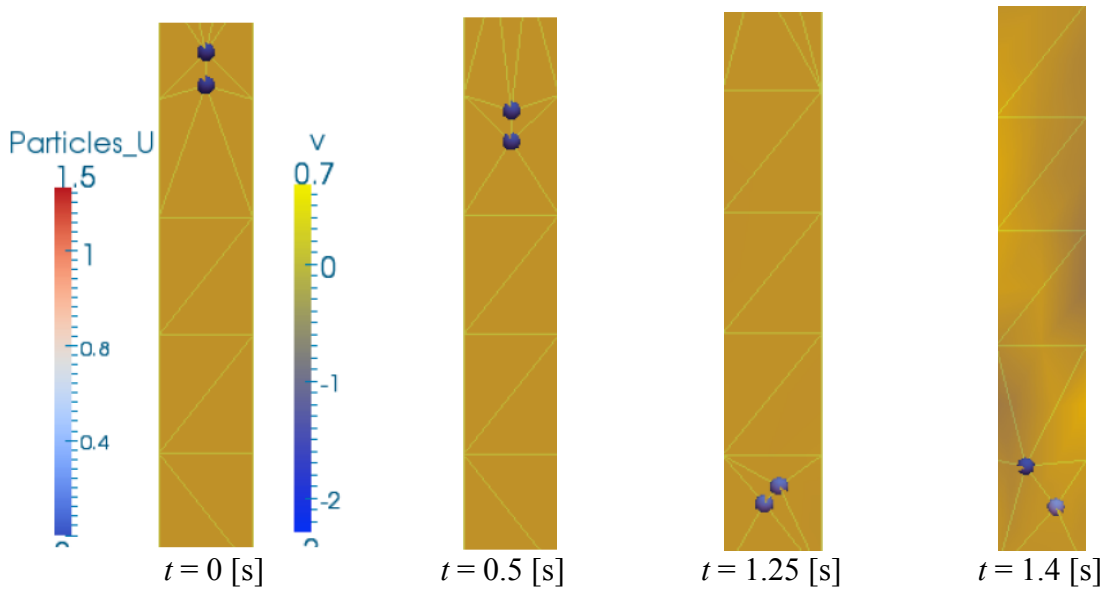


Figure 9: Snapshots of the Drafting, Kissing, Tumbling (DKT) problem. Triangles show the deforming mesh as the simulation progresses.

A more quantitative study of this test case and the comparison with fully resolved simulations [42, 43] is one of the next steps to be done. The present simple cases are rather an illustration that the method is qualitatively working, but does not yet represent a quantative validation which is work in progress.

5. Summary and conclusions

A meso-scale, two-way fluid-particle interaction framework based on coupling FEM and a soft particle DEM on an unstructured, moving mesh has been proposed. The key component in our approach is a Delaunay triangulation, which serves both as a contact detection tool and a FEM mesh generator. The triangulation deforms and changes with the particle motion. This design alleviates any computational overhead purported by existing methods for contact detections, particularly in dense particulate flows. Since particles always occupy nodal positions in our mesh, locating particles inside cells also becomes trivial while at the same time the particles help to keep the mesh from degenerating. Furthermore, duplication of data for storing the mesh and particles as well as their contact detection, has been avoided by defining a triangulation based on particle locations.

A FEM based fluid solver allows for a higher order interpolation when a better resolution of the flow is desired, whenever the underlying mesh is coarse. On the other hand, dense flows are resolved rather well with low order, since the mesh resolution is refining inversely proportional to the particle density. Different time scales in DEM and FEM can be coupled through inner iterations of DEM steps. The approach provides the dynamics of the particles and the fluid using a deforming mesh, while reasonably well resolving the fluid flow around the particles. The average velocities are accurately predicted when compared to fully resolved simulations, which indicates that the assumptions made are valid (for the drag laws, the continuum equations, and the choices made for mechanical and technical details) even though they might be at their limits. Many open issues remain for more detailed future studies of the algorithm and methodology, as stated in the previous sections.

The next step is to validate the model with more complex experimental test-cases like fluidized beds, batch-sedimentation, or particle dispersion [41], which is work in progress. Already now, we identified a challenge: the empty areas that are generated by particle-inhomogeneities require some type of mesh-refinement, since they leave too big cells, which could be addressed by adding “ghost”-particles that are used for mesh-generation, are repelled from each other and from real particles, but are neither affecting

the fluid nor the other particles. In future, another challenge will be to deal with multiple phases and to couple various other physical fields (e.g. temperature, electromagnetic, etc.), using the same data structure and to use the same data-structure also for coarse-graining or up-scaling [40] to achieve an even more versatile multi-scale model.

Acknowledgements:

We thank S. Turek for the discussions related to the incompressible solver. This work is sponsored by STW through the MUST Project Number 10120.

References

[1] J.A.M. Kuipers and K.J. Van Duin and F.P.H. Van Beckum and W.P.M. Van Swaaij, A numerical model of gas-fluidized beds, CES, 47 (1992) 1913-24.

[2] Z. Feng and E. E. Michaelides, The immersed boundary-lattice Boltzmann method for solving fluid-particles interaction problems, JCP, 195 (2004) 602-28.

[3] B. Avci, P. Wriggers, A dem-fem coupling approach for the direct numerical simulation of 3d particulate flows, J. Appl. Mech., 79 (2012) 10901.

[4] S. Schwarzer, K. Höfler, B. Wachmann, H.J. Hermann, Particle-level simulations of sedimentation and aggregation, 1998, John Wiley, New York.

[5] B.H. Xu and A.B. Yu, Numerical simulation of the gas-solid flow in a fluidized bed by combining discrete particle method with computational fluid dynamics, Chemical Engineering Science, 52 (1997) 2785 – 2809.

[6] A.J.C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 1. Theoretical foundation, JFM, 271 (1994) 285-309.

[7] A.J.C. Ladd, Numerical simulations of particulate suspensions via a discretized Boltzmann equation. Part 2. Numerical results, JFM, 271 (1994) 311-339.

[8] K. Han, Y.T. Feng, D.R.J. Owen, Coupled lattice Boltzmann and discrete element modelling of fluid-particle interaction problems, Computers & Structures, 85 (2007) 1080-88.

[9] A. Narvaez, K. Yazdchi, J. Harting, and S. Luding, From creeping to inertial flow in porous media: a lattice Boltzmann - Finite Element study, J. Stat. Mech., (2013) P02038.

[10] Y. Tsuji, T. Kawaguchi, T. Tanaka, Discrete particle simulation of two-dimensional fluidized bed, Powder Technology, 77 (1993) 79-87.

[11] J.A.M. Kuipers, K.J.V. Duin, P.H.V. Beckum, W.P.M.V. Swaaij, Computer simulation of the hydrodynamics of a two-dimensional gas-fuidized bed, Comput. Chem. Engrg., 17 (1993) 839-858.

[12] C.L. Wu, A.S. Berrouk, K. Nandakumar, Three-dimensional discrete particle model for gas-solid fluidized beds on unstructured mesh, Chem. Eng., 152 (2009) 514-29.

[13] P.A. Cundall, O.D.L. Strack, A discrete numerical model for granular assemblies, Geotechnique, 29 (1979) 47-65.

[14] S. Ergun, Fluid Flow through Packed Columns, Chem. Eng. Prog., 48 (1952) 89-94.

[15] D. Gidaspow, Multiphase flow in fluidized beds, (1994) Academic Press.

[16] J.E. Drummond, and M.I. Tahir, Laminar viscous flow through regular arrays of parallel solid cylinders, Int. J. Multiphase Flow, 10 (1984) 515-40.

[17] B.R. Gebart, Permeability of Unidirectional Reinforcements for RTM, Journal of Composite Materials, 26 (1992) 1100–33.

[18] K. Yazdchi, S. Srivastava and S. Luding, Microstructural effects on the permeability of periodic fibrous porous media, IJMF, 37 (2011) 956-66.

[19] K. Yazdchi, S. Srivastava and S. Luding, Micro-macro relations for flow through random arrays of cylinders, Composites Part A, 43 (2012) 2007-2020.

[20] B.P.B. Hoomans, J.A.M. Kuipers, W.J. Briels, W.P.M. van Swaaij, Discrete particle simulation of bubble and slug formation in a two dimensional gas-fluidised bed: A hard-sphere approach, Chem. Eng. Sci., 51 (1996) 99-118.

[21] M. Xu, W. Ge, J. Li, A discrete particle model for particle-fluid flow with considerations of sub-grid structures, Chem. Eng. Sci., 62 (2007) 2302-08.

[22] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, A distributed Lagrange multiplier/fictitious domain method for particulate flows, Int. J. of Multiphase Flow, 25 (1999) 755-794.

[23] T.W. Pan, D.D. Joseph, R. Bai, R. Glowinski, V. Sarin, Fluidization of 1204 spheres: simulation and experiment, JFM, 451 (2002) 169-191.

[24] Y. Kanarska, I. Lomov, T. Antoun, Mesoscale simulations of particulate flows with parallel distributed lagrange multiplier technique, Comput. Fluids, 48 (2011) 16-29.

[25] T. Tezduyar, M. Behr, J. Liou, A new strategy for finite element computations involving moving boundaries and interfaces-the deforming spatial- domain/space-

time procedure: I. the concept and the preliminary numerical tests, Comput. Meth. Appl. Mech. Eng., 94 (1992) 339-351.

[26] J.A. Ferrez, T. Liebling, D. Müller, Dynamic triangulations for granular media simulations, Lecture Notes in Physics, 554 (2000) 394-409.

[27] T.B. Anderson, R. Jackson, A fluid mechanical description of fluidized beds, I & EC fundamentals, 6 (1967) 89-102.

[28] N. Deen, M.V.S. Annaland, M.V. der Hoef, J.A.M. Kuipers, Review of discrete particle modeling of fluidized beds, Chem. Engineering Science, 62 (2007) 28-44.

[29] P.C. Carman, Fluid flow through granular beds, chemical engineering research and design. Transac. of the Inst. of Chem. Eng., 75 (1997) 32-48.

[30] K. Yazdchi, S. Luding, Towards unified drag laws for inertial flow through fibrous materials, CEJ, 207 (2012) 35-48.

[31] S. Luding, Cohesive frictional powders: Contact models for tension, Granular Matter, 10 (2008) 235-246.

[32] D. Wan, S. Turek, Fictitious boundary and moving mesh methods for the numerical simulation of rigid particulate flows, J. Comput. Phys., 222 (2007) 28-56.

[33] C. Forster, W.A. Wall, E. Ramm, On the geometric conservation law in transient flow calculations on deforming domains, IJNMF, 50 (2006) 1369-79.

[34] T. Tezduyar, S. Sathe, Stabilization parameters in SUPG and PSPG formulations, J. Comput. Appl. Math, 4 (2003) 71-88.

[35] C. Forster, W.A. Wall, E. Ramm, Stabilized finite element formulation for incompressible flow on distorted meshes, IJNMF, 60 (2009) 1103-26.

[36] F. Brezzi, M. Fortin, Mixed and Hybrid Finite Element Method, 1991, Springer Series in Computational Mechanics.

[37] S. Turek, Efficient Solvers for Incompressible Flow Problems: An Algorithmic and Computational Approach, 1999, Springer.

[38] M. Robinson, S. Luding and M. Ramaioli, Fluid-particle flow modelling and validation using two-way-coupled mesoscale SPH-DEM, Int. Journal of Multiphase Flow, 59 (2013) 121-134.

[39] A.P. Peskin, G.R. Hardin, Moving particles through a finite element mesh, J. Research of National Inst. Standards and Technology, 103 (1998) 77-91.

[40] K. Yazdchi, and S. Luding, Upscaling and microstructural analysis of the flow-structure relation perpendicular to random, parallel fiber arrays, Chem. Eng. Sci. 98 (2013) 173-185.

[41] M. Robinson, S. Luding, and Marco Ramaioli, SPH-DEM simulations of grain dispersion by liquid injection, in: Powders and Grains 2013, A. Yu, K. Dong, R. Yang, and S. Luding (Eds.), AIP Conf. Procs. #1542 (2013), pp. 1122-1125.

[42] J. Feng, H. H. Hu and D. D. Joseph, Direct simulation of initial value problems for the motion of solid bodies in a Newtonian fluid Part 1. Sedimentation, J. Fluid Mech. (1994), 277, 271-301.

[43] C.K. Aidun, Y. Lu and E.-J. Ding, Direct analysis of particulate suspensions with inertia using the discrete Boltzmann equation, J. Fluid Mech. (1998), 373, 287-311.