A spatially resolved fluid-solid interaction model for dense granular packs/Soft-Sand.

Paula A. Gago · Ali Q. Raeini · Peter King

Abstract Fluid flow through dense granular packs or soft sands can be described as a Darcy's flow for low injection rates, as the friction between grain-grain and grain-walls dominate the solid system behaviour. For high injection rates, fluid forces can generate grain displacement forming flow channels or "fractures", which in turn modify local properties within the system, such as permeability and stress distribution. Due to this kind of "self organized" behaviour, a spatially resolved model for these interactions is required to capture the dynamics of these systems. In this work, we present a resolved model based on the approach taken by the $CFDEM$ open source project which uses $LIGGGHTS$ – a discrete elements method (DEM)– to model the granular behaviour and $OpenFoam$ finite volume library for computational fluid dynamics (CFD), to simulate the fluid behaviour. The capabilities provided by the DEM engine allows the properties of the solid phase, such as inter-grain cohesion and solid confinement stress to be controlled. In this work the original solver provided by the $CFDEM$ project was modified so as to deal with dense granular packs more effectively. Advantages of the approach presented are that it does not require external "scaling parameters" to reproduce well known properties of porous materials and that it inherits the performance provided by the $CFDEM$ project. The model is validated by reproducing the well-known properties of static porous materials, such as its permeability as a function of the system porosity, and by calculating the drag coefficient for a sphere resting inside a uniform flow. Finally, we present fracture patterns obtained when modelling water injection into a Hele-Shaw cell, filled with a dense granular pack.

Paula A. Gago
Department of Earth Science and Engineering, Imperial College London, SW7 2AZ, UK.
E-mail: p.gago@imperial.ac.uk

Ali Q. Raeini
Department of Earth Science and Engineering, Imperial Colleg London, SW7 2AZ , UK.
E-mail: a.qaseminejad-raeini09@imperial.ac.uk

Peter King
Department of Earth Science and Engineering, Imperial College London, SW7 2AZ, UK.
E-mail: peter.king@imperial.ac.uk

Declaration of interest

Declarations of interest: none.

1 Introduction

Many physical and engineering systems involve the interaction of fluid with "moving" particulate systems. Examples of such systems goes from the swimming of bacteria [1,2], where the movement of the bacteria flagellum moves the body of the organism, to the pneumatic fracturing created when gas leakage from underground reservoirs as it migrates through the soil before escaping into the atmosphere. This type of fluid-solid interaction is of interest also in oil industry, when water in injected into oil reservoirs in soft-sand formations to enhance production. It is easy to notice that these interactions are of different types. The bacteria swimming can be described as a dilute suspension with little interaction among solid particles, while for water injection into soft sand, the solid-solid interactions are the predominant ones as they affect strongly the system dynamics.

Accurate experimental [3–5] and numerical [2,6,7] models are of value in characterising the behaviour of these systems. Spatially resolved approaches have been successfully developed to deal with the detailed movement of dilute systems [2,7,8], however, when the solid-solid interaction becomes dominant, these models become less accurate. In this work we present a spatially resolved model to simulate fluid-solid interaction for dense granular systems.

Zhang et. at [6] have modeled dense systems using a Discrete Element Method (DEM) coupled with a pore network fluid flow scheme [9]. The model has successfully reproduced experimental results on the morphology of the fractures developed into soft-sand due to fluid injection, but this model requires external calibration factors to scale the macroscopic system properties to obtain suitable microscopic model parameters.

In this work we present a numerical model based on the $CFDEM$ [7,10,11] open source project that enables us to deal with the grain scale behaviour of fluid injection into weakly consolidated (deformable) dense granular packs. The $CFDEM$ project couples two well established (open source) engines, $LIGGGHTS$ [12], a DEM engine specialised in simulation of particulate systems, and $OpenFOAM$ [13], a finite volume method for computational fluid dynamics.

The present model uses the Immersed Boundary method [8] to locate the solid boundary within the fluid domain and introduces a discontinuous density field into the CFD domain to account for solid presence. It represents a variation of the "Immersed Boundary Solver" provided by the $CFDEM$ project [7,11].

In Section 2, we describe our approach and discuss its differences with the $CFDEM$ Immersed Boundary Solver [7]. In Section 3, we show how the model captures the behaviour of the permeability variation in a porous material as a function of the system porosity, without the need of any external calibration parameters. We then analyse the drag coefficient for a spherical grain immersed in

a constant fluid stream. Finally, we show the dynamic behaviour of water injection into soft-sand by modelling water injection into a granular pack contained in a radial Hele-Shaw cell with a central inlet. Section 5 summarises the model capabilities and limitations and discusses its future applications.

2 Theory

The basis of the procedure followed by the $CFDEM$ project to couple the CFD engine (based on $OpenFOAM$ open source project) and the DEM engine (based on $LIGGGHTS$ open source project) is shown in Fig 1. Each engine resolves interaction in its particular domain and then information is passed between them. The algorithm starts by getting particle positions and velocities from the DEM engine. Information is then passed to the CFD solver which solves the fluid equation to obtain the velocity and pressure fields compatible with the given boundary conditions. Fluid forces acting on each solid object are then calculated based on these pressure and velocity fields. These forces are sent to the DEM engine, where they are added to the intrinsic solid forces calculated by the DEM engine. Particle positions and velocities are then updated and the cycle starts again.
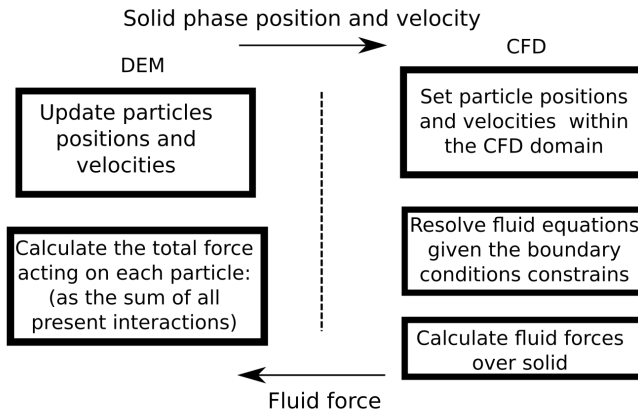
Fig. 1 $CFDEM$ coupling procedure. In each time-step the DEM engine determines grains positions and velocities and send this information to the CFD engine. The CFD engine solves the fluid equations compatible with the given boundary conditions. Velocity and a pressure fields are obtained and forces F_g on each grain due to the fluid are calculated. F_g is sent to the DEM engine which adds it to the contact forces and any other grain interaction forces set within the DEM model and updates particle positions and velocities.

Particles are placed within the CFD domain using the Immersed Boundary method [8]. It discretises the continuum boundary of the solid object into the CFD mesh, as shown in Fig. 2 (a). Grid-blocks (in the CFD domain) that are completely "covered" by particles are assigned a "void fraction" of zero ($v_f = 0$) and those grid-blocks which are partially covered by particles are assigned a void fraction with a value proportional to the covered area. Grid-blocks containing just

fluid are assigned a void fraction of one ($v_f = 1$). The application of this method requires the CFD mesh refinement to be few times smaller than the mean particle diameter. Figure 2 (b) shows three different mesh refinements (from left to right 7, 8 and 9 grid-blocks per particle diameter) for the same system setup. It is possible to see how the "mass distribution" gets more localised as the refinement increases. Due to the different time scales involved in the fluid and solid dynamics calculations, different time-steps are set for each engine. It was shown [14] that to resolve accurately a grain-grain contact at least 100 steps are required, and hence the time-step used in the CFD solution is usually 100 times bigger than that used in the DEM engine.
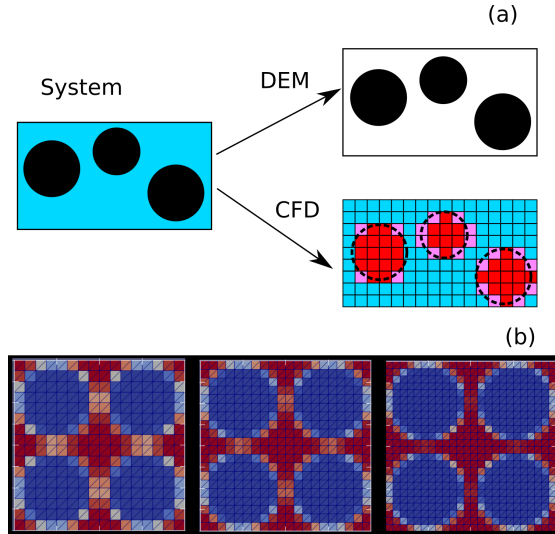
Fig. 2 (a) Decomposition of the whole system into the different domains. The granular dynamics is resolved using a DEM engine ($LIGGGHTS$) while the fluid dynamic is solved using a CFD solver based on $OpenFOAM$. (b) Different degrees of the CFD mesh refinement. From left to right, 7, 8 and 9 grid-blocks per particle diameter.

Since particles are immersed in the fluid, the particle velocity at the fluid-particle interface should be passed to the CFD engine. This, in addition to the intrinsic boundary conditions given to the fluid system at the domain boundaries, form the system boundary conditions.

The $CFDEM$ open source project provides a spatially resolved solver (called the "Immersed Boundary Solver" [7]) which has been proven effective for dilute systems. The $CFDEM$ implementation resolves the fluid fields as follows.

It first uses the Pressure-Implicit with Splitting Operator ($PISO$) algorithm to solve the discretized Navier-Stokes equations over the whole domain without considering the presence of particles:

$$\rho \frac{\bar{u} - u^{n-1}}{\Delta t} + \rho(u^{n-1}\nabla)u^{n-1} = -\nabla\bar{p} + \mu\nabla^2 u^{n-1} \tag{1}$$

$$\nabla\bar{u} = 0 \tag{2}$$

where ρ is the fluid density, μ is the fluid viscosity, u^{n-1} is the velocity solution at the previous time-step and $\bar{u}$ and $\bar{p}$ denotes interim velocity and pressure solutions, respectively.

The interim $\bar{u}$ solution is then corrected by forcing those grid-blocks covered by particles (i.e. those grid-blocks with void fraction smaller than 1) to have a velocity equal to the corresponding particle velocity (or proportion of it) in order to satisfy the boundary condition constraint imposed by the grain movement to obtain a "corrected" velocity $\bar{u}_c$. It is possible to see that this step is equivalent to adding a force term f to the Eq. (1):

$$f = \rho \frac{\bar{u}_c - \bar{u}}{\Delta t}, \tag{3}$$

and that the divergence free condition (Eq. (2)) is lost for $\bar{u}_c$.

The divergence free condition is recovered and the final velocity solution u^n is obtained by correcting $\bar{u}_c$ as:

$$u^n = \bar{u}_c - \nabla \theta \tag{4}$$

where θ is a scalar field satisfying:

$$\nabla^2 \theta = \nabla \bar{u}_c \tag{5}$$

A more detailed explanation of the $CFDEM$ model implementation can be found in [7,15].

The approach described above is accurate for dilute systems, where the change in particle velocities is due to fluid interaction and hence, the fluid-solid interaction can be considered as the only action-reaction pair in the system. For dense particle packs, where the particle movement does not only depend on the fluid interaction, the force term in Eq. (3) can be divergent as a function of the time-step size due to the denominator of Eq. (3). Take for example a static particle, fixed by external means, in a uniform flow stream. Hence, $\bar{u}_c = 0$ in grid blocks occupied by the particle. Due to the loose coupling between the particle and the fluid equations (1) and (2), the force applied to fluid, Eq. (3), will depend on the time step size.

Therefore, the above approach is not accurate for dense granular packs where the particles are often static (or highly constrained in their movement) as a consequence of the grain-grain interaction.

In the model presented in this paper to improve the coupling between particles and fluid velocities, we first set an artificial "denser" phase to those grid-blocks containing particles and incorporate particle velocities into the CFD domain before solving the momentum predictor step of the $PISO$ algorithm:

$$\rho^* \frac{u - u^{n-1}}{\Delta t} + \rho^* (u^{n-1}\nabla)u^{n-1} = -\nabla p + \mu \nabla^2 u^{n-1} \tag{6}$$

where $\rho^* = \rho(1 - \Theta(1 - v_f))$ with $\Theta \gg 1$ is a parameter that accounts for the presence of particles and v_f the block void fraction. As ρ^* is higher for "solid" blocks it will tend to conserve the associated particle momentum with respect to less dense fluid filled block surrounding them, i.e. the fluid blocks near solid blocks will keep the particle velocities, to balance the momentum equation.

Following the implementation provided by [11], the fluid forces acting on each particle are then calculated from the computed fluid fields by:

$$f_g = \sum_{c \in T_g} (\mu \nabla^2 u^n - \rho \nabla p^n) v_c \qquad (7)$$

where T_g is the set of cells representing the given particle in the CFD framework and v_c is the volume of the cell. These forces are then passed to the DEM engine where they are added to the intrinsic DEM forces before particle positions and velocities are updated.

In summary, each time step involves:

– Setting the particle positions in the CFD domain through the void fraction, v_f, field (Fig. 2).
– Imposing the particle velocities on the CFD mesh (in proportion to the v_f field).
– Solving Eqs. (6) and (2) using the Pressure-Implicit with Splitting Operator ($PISO$) algorithm.
– Calculating the forces acting on grains using equation (7).
– Calculating the total forces acting on each grain and updating the particle positions and velocities.

In the following section we will show the performance provided by this approach and we will show that the solution becomes convergent for Θ values bigger than 10^6.

3 Validation

In the following, we first show the performance of this solver as a function of the parameter Θ in Eq. (6), the time-step and the mesh refinement. Then, we show that the model can reproduce well known properties of solid-fluid interaction without the need for external scaling parameters.

3.1 Parameter dependence

To assess the solution convergence, a constant flow through a porous material (composed of a dense pack of spheres held together by external confining forces) was simulated. The pressure drop Δp through the material was analysed as a function of the model parameters: Θ, the time-step size and the CFD mesh refinement.

The system setup is shown in Fig. 3 (a). It consists of a single layer of spheres of diameter 0.1mm. The sample is confined inside a rigid rectangular box (not showed in the figure for simplicity) so that the granular pack is forming a solid-like porous material with none of little particle movement. The box dimensions are $L = 1$mm, $h = 2$mm and a thickness $\gamma = 0.11$mm (slightly bigger than one sphere diameter). The initial granular configuration was obtained by pouring grains inside the container and then moving the top and the bottom walls in the vertical direction while applying a constant force to the grains until reaching a static condition. The porosity of the system obtained using this protocol is about $\phi = 40\%$.
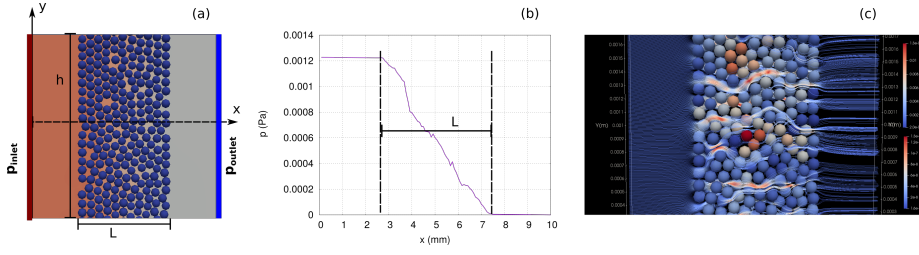
Fig. 3 (a) System setup for pressure drop convergence analysis. A dense pack of mono-sized spheres of radius 0.1mm confined inside a rectangular box of fixed walls (not shown for simplicity). Box dimensions: length $L = 1.05$mm, height $h = 2$mm, thickness $\gamma = 0.11$mm. A constant flow rate q is applied at the left wall while $p = p_{outlet}$ is applied at the right wall. (b) Pressure behaviour along the x-axis of the same system. Pressure through the porous material decays smoothly as expected for a Darcy flow. (c) Fluid streamlines through the granular system obtained with the present model.

A constant flow rate, q, is applied at the left wall of the cell and the right side is set to constant pressure p_{outlet}. The remaining walls of the system are set with zero velocity constrains.

Within the $LIGGGHTS$ implementation, we used a $hertz$ [14] model for the grain-grain and grain-wall contact forces with Young's modulus $Y = 5 \times 10^6 \mathrm{Nm}^{-2}$, Poisson ratio $n = 0.15$, restitution coefficient $\epsilon = 0.7$ and friction coefficient $f_\mu = 0.7$. For the fluid domain we used a non-compressible linear Newtonian model with fluid density $\rho_{fluid} = 10^3 \mathrm{Kg\ m}^{-3}$ and fluid viscosity $\mu = 10$cPa·s.

Figure 3 (b) shows the pressure behaviour along the x axis. As expected, pressure decays linearly through the porous material, i.e. the flow can be described with Darcy's law and a permeability K can be associated with the system:

$$K = -\frac{q\mu L}{\Delta p} \tag{8}$$

where L is the length of the porous medium, μ is the fluid viscosity, q is the Darcy flow velocity and Δp is the pressure drop through the material.

Figure 3 (c) shows the flow streamlines through the porous material and how the fluid flow is confined to move through the inter-grain pore spaces.
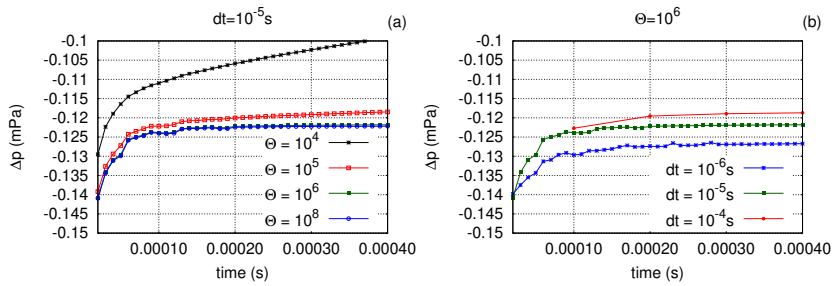
Fig. 4 Pressure drop through the porous material of Fig. 3 (a) as a function of time for (a) different values of the parameter Θ (Eq. (6)) and (b) different time steps dt.

Figure 4 (a) shows the evolution of the pressure drop along the porous material, for different values of the model parameter Θ. It is possible to see that the model converges for values of $\Theta > 10^6$. Figure 4 (b) shows the evolution of Δp for three different time-steps, dt. As expected the solution does not depend on the time-step more than it would be expected to for this kind of discrete numerical solvers [16].

Results in Fig. 4 were obtained using a mesh refinement of $\Delta x = 6$ grid-blocks per particle diameter. In the next sub-section we analyse how mesh refinement affects the pressure behaviour.

3.2 System permeability

For the rest of this work the value $\Theta = 10^6$ will be fixed and, unless stated otherwise, a time-step of $dt = 10^{-5}$s will be used.

We use Eq. (8) and the stationary value reached by Δp to obtain the system permeability K.

Figure 5 (a) shows K as a function of the CFD mesh refinement. The blue dashed line in Fig. 5 (a) shows a solution obtained using a single-phase CFD solver [17], where only the void space is meshed (a mesh resolution corresponding to 16 grid-blocks per particle diameter is used) to create the CFD boundaries and set an explicit zero velocity boundary condition over all the boundaries between the fluid and solids. We can see that both approaches provide permeability values within the same range, although the strict zero velocity boundary condition imposed in the single-phase CFD solver results in a slightly lower permeability for the same mesh refinement. Figure 5 (a) shows that the system permeability increases as the mesh refinement increases but seems to converge as the mesh refinement increases. This dependence is expected since permeability depends strongly on the pore and throat size distribution and different degrees of mesh refinement affect the "effective" inter-grain pore-space, as can be seen in Fig. 2 (b).

According to the Kozeny-Carman equation [18,19] permeability dependence on porosity can be described as:

$$K = \frac{\phi^3 d^2}{180(1 - \phi)^2} \tag{9}$$

where d is the mean particle diameter and ϕ is the system porosity. If measured from the DEM configuration the system porosity results about $\phi \sim 0.515$, and as particle diameters is $d = 0.1$mm, this would result in a system permeability of ≈ 32D (using Eq. (9)). Although this value is still twice the value obtained with the higher refinement studied here, it is in the same order of magnitude. Depending on the system studied a cost-benefit regarding to the mesh refinement and the required simulation times should be analysed in each case.

In order to evaluate permeability as a function of system porosity in the present model, granular packs having different porosities were created. A set of 10 different packs were obtained by using different grains sizes, different inter-grain friction coefficient and different initial pack compaction pressures. Different porosity values obtained as a consequence of mesh refinement were also considered. System porosity is defined as the fraction of grid-blocks with a void fraction value bigger than 0.9 (essentially just fluid). The variation between porosity defined in this way and the obtained from the DEM geometry was around the [-20-20]% interval.
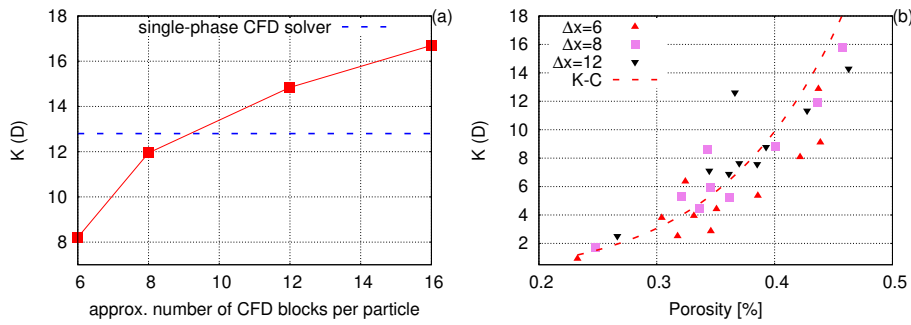
Fig. 5 (a) System permeability K as a function of CFD mesh refinement. Dashed blue line corresponds to the solution obtained using a single-phase CFD solver [17] where only the void space is meshed and an explicit no-slip boundary condition is applied over all the boundaries between the fluid and solids. A mesh refinement of 16 grid-blocks per particle diameter was used. (b) System permeability K as a function of system porosity, defining system porosity as the proportion of grid-blocks having a void fraction $v_f > 0.9$. Dashed red line corresponds to the theoretical prediction $K = (\phi^3 d^2)/(180(1-\phi)^2)$, where ϕ is the system porosity and d the particle diameter.

Figure 5 (b) shows the behaviour of K as a function of system porosity. The red dashed line in the figure shows the behaviour predicted by the Kozeny-Carman equation (Eq. (9)). We see good agreement between the results obtained and theoretical predictions. We can also see that the results have the same trend independently of the way in which the given porosity was obtained; i.e. similar results are obtained by changing the actual pore-space size or slightly changing the mesh refinement. These results suggests that, if the system porosity is considered as that perceived by the CFD solver, our results are in good agreement with other CFD methods.

3.3 Drag-force

In this section we analyse the forces acting on the solid phase due to the fluid flow. To simplify the setup, the fluid force acting over a single static particle is considered.

First the dynamics of a free particle moving under the action of a flow stream of constant velocity inside a rectangular pipe is analysed. Figure 6 (a) shows a lateral cut representation of the system setup. A constant velocity boundary condition of $U = 2\text{mm s}^{-1}$ in the x direction and a constant pressure at the right hand boundary. The CFD domain is a rectangular box with $h \times h$ square transversal area ($h = 10\text{mm}$) and no gravity is considered. A particle of radius 1mm is placed in the centre of the fluid domain.

Figure 6 (b) shows the particle velocity v in the x direction as a function of time for three different time-steps. The mesh used has a resolution of 6 grid-blocks per particle diameter. We see that the particle velocity converges to the fluid system velocity, independent of the time-step size, as is expected since there are no other forces acting on the grain.
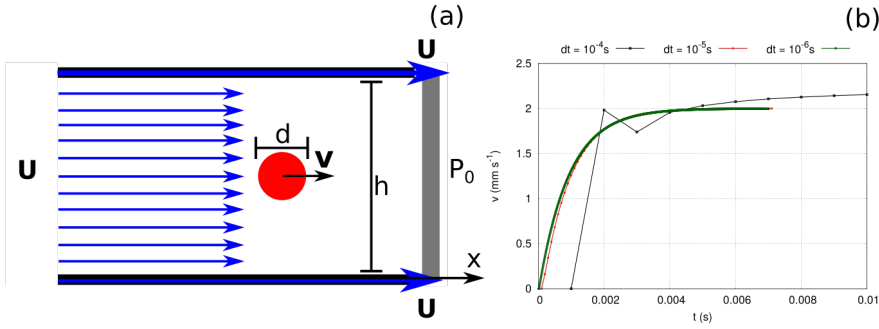
Fig. 6 (a) Lateral view of the $(3D)$ system used to characterise particle movement due to fluid flow: a particle of $d = 1$mm diameter is placed in the centre of a rectangular cell (with square cross section h, $h = 10$mm) that contains a fluid stream flowing at a velocity $U = 2$mm s^{-1}. The lateral (upper and lower sides in the figure) wall velocities were set equal to U to avoid finite size effects. (b) Particle velocity v as a function of time for three different time-steps values. The CFD mesh used has a resolution of 6 blocks per particle diameter.

If we now prevent the particle from moving (by external means) we can measure the drag force F_p exerted by the fluid stream on the particle and hence calculate the drag coefficient, $C_d = 2F_p/(\rho_{fluid}V^2 a)$, where ρ_{fluid} is the fluid density, $V = U$ is the relative velocity between the flow and the particle and $a = 2\pi r^2$ is the cross sectional area of the particle (of radius r).

Figure 7 (a) shows C_d as a function of Reynolds number $Re = V\rho_{fluid}l/\mu$, with μ being the fluid viscosity and $l = h$ is the width of the domain perpendicular to the direction of flow.
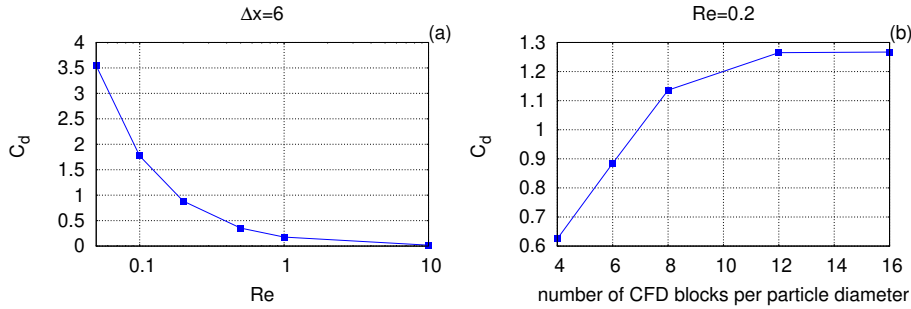
Fig. 7 Drag coefficient (for the setup represented in Fig. 6) for a particle forced to remain static (by external means) as a function of (a) Reynolds number (Re) and (b) mesh refinement.

The values of C_d lie within the expected range given in Ref. [20]. The values in Fig. 7 (a) significantly improve the values reported by the $CFDEM$ open source project implementation [7]. As for the system permeability, the drag coefficient slightly varies with Δx but converges for mesh refinements larger than 12 blocks per particle diameter, as can be seen from Fig. 7 (b), which shows C_d as a function of the CFD mesh refinement.

4 Grain-displacement driven by fluid: fractures in soft-sand

Finally, we show the capability of the model to model fracture growth in of soft-sand samples due to water injection. We performed simulations of fluid injection into a rigid radial Hele-Shaw cell with central inlet, filled with granular material, as it is a commonly used geometry for experimental setups [3,6].

We created the quasi-two-dimensional Hele-Shaw cell by compacting $N = 5 \times 10^4$ spherical grains initially loosely confined between two fixed horizontal parallel plates (separated a distance $h \sim 2.4$mm) by 8 vertical piston-like walls. This methodology was chosen so as to reduce solid stress anisotropy and to minimise the creation of preferential stress directions. Each piston-like wall applied a constant force $F_{wall} = 4$N to the system. After reaching a static configuration all the system walls were fixed in place and remained fixed during the fluid injection simulations. The diameter of the cell at this point was approximately $D \sim 9$cm, $D \gg h$, (Fig. 8 (a)). The central inlet diameter was 8mm.

Grain diameters were poly-dispersed distributed, following a truncated normal distribution of mean $< d_m >= 1.35$mm and standard deviation $\sigma_d = 0.3$mm, truncated to a range of $[0.9 - 1.8]$mm. The same parameter values introduced in section 3.1 for the solid and the fluid numerical implementation were used and a weak inter-grain cohesion was included by using a simplified Johnson-Kendall-Roberts [21,22] (part of $LIGGGHTS$ [12] implementation), with cohesion energy density 3×10^5J/m^3.

Different fluid pressure injection p_i were simulated starting from the same granular initial configuration. The pressure at the octagonal perimeter of the cell was set constant ($p_{outlet} = 0$) and a zero velocity boundary condition was imposed on the upper and lower walls of the cell. A mesh refinement of 6 grid-blocks per mean particle diameter was used. Although this refinement is in the lower limit of the refinements analysed in the previous sections, we find that this gives us an acceptable compromise between the required system size and the simulated time and the required computational time to achieve it. A total of 26 inlet pressures were simulated for 0.01s each (that time was enough to allow the system reach a new static equilibrium state). Each simulation, on average, took eight hours running on 32 processors (at the Imperial College Research Computing Service [23]).

Figure 8 shows the system behaviour for three different inlet pressure values (increasing from (a) to (c)). The colour scale corresponds to the trace of the stress tensor $\sigma_\alpha^\beta = (1/V) \sum_c r_c^\alpha f_c^\beta$ of each particle [24], where V is the particle volume, c runs over all the contacts of the particle, r_c is the vector from the center of the grain to the contact point c and f_c is the corresponding contact force.

It can be seen that at low inlet pressures, the system maintains its structure as a solid porous system (Fig. 8 (a)). The sample permeability $K_s \sim 30$D was measured from the flow rate obtained in this configuration and using Darcy's law.

From Fig. 8 (b) we see that at a "critical" inlet pressure, the system starts to deform initiating the formation of flow channels or fractures. For higher inlet pressures flow channels development increases (Fig. 8 (c)). From Fig. 8 it is also possible to distinguish the regions of accumulation of stresses in the granular system while the fracture develops. This stress accumulation and its relation with the fracture geometry is a phenomenon that we aim to study extensively in future works.

Finally, figure 9 shows the area of the fracture connected with the fluid inlet, as a function of the inlet pressure p_i. Here, the existence of the two regimes, a Darcy-like flow and a "fractured" one is clear.

Although a more statistically significant range of variations is necessary to properly characterise the fracture growth initialisation and behaviour after initialization, the present example captures one of the main expected behaviours related to fracturing soft sand [3,25], showing the capabilities of the presented model.

5 Conclusions

In this work we presented a spatially resolved fluid-solid model based on the coupled interaction of a CFD and a DEM engine, each a highly optimized software and with well established in their respective independent applications. The model is based on the $CFDEM$ [15] open source framework and hence inherits its key aspects of performance.

We showed that the macroscopic parameters describing the porous material, such as the system permeability, can be obtained without need for any external scaling factors and that the fluid-solid force obtained is accurate and provides values for the drag coefficient within the range of values reported [20].

Although the results obtained depends on the CFD mesh refinement, we showed that this dependence is mostly due to changes in the system effective porosity and that it does not strongly influence the model predictions. Efforts have been made by other authors [26,27] to improve the sharpness of the solid-fluid interface for the immersed boundary condition that could in the future be incorporated to the model.

The model is capable of simulating the generation of fractures in soft-sand samples due to water injection and reproducing the expected features in these systems, for instance the existence of a critical fluid injection pressure for fracture initialization.

In future work we will use the present model to characterise the fracture growth initialisation and behaviour after initialization and to assess the effect that the granular properties (e.g. grain-grain friction and cohesion) have on these systems.

References

1. T. Goto, K. Nakata, K. Baba, M. Nishimura, and Y. Magariyama. A fluid-dynamic interpretation of the asymmetric motion of singly flagellated bacteria swimming close to a boundary. *Biophysical Journal*, 89(6):3771–3779, 2005.
2. A. A. Shirgaonkar, M. A. MacIver, and N. A. Patankar. A new mathematical formulation and fast algorithm for fully resolved simulation of self-propulsion. *J. Comput. Phys.*, 228(7):2366–2390, 2009.
3. H. Huang, F. Zhang, P. Callahan, and J. Ayoub. Granular fingering in fluid injection into dense granular media in a hele-shaw cell. *Phys. Rev. Lett.*, 108(25):258001, 2012.

4. G. Varas, J-C. Géminard, and V. Vidal. Air invasion in a granular layer immersed in a fluid: morphology and dynamics. *Gran. Matt.*, 15(6):801–810, 2013.
5. F. K Eriksen, R. Toussaint, A. L. Turquet, K. J Måløy, and E. G Flekkøy. Pressure evolution and deformation of confined granular media during pneumatic fracturing. *Phys. Rev. E.*, 97(1):012908, 2018.
6. F. Zhang, B. Damjanac, and H. Huang. Coupled discrete element modeling of fluid injection into dense granular media. *J. Geophys. Res. Solid Earth*, 118(6):2703–2722, 2013.
7. A. Hager, C. Kloss, S. Pirker, and C. Goniva. Parallel open source cfd-dem for resolved particle-fluid interaction. In *Proceedings of 9th International Conference on Computational Fluid Dynamics in Minerals and Process Industries*, pages 1–6, 2012.
8. C. S Peskin. The immersed boundary method. *Acta numerica*, 11:479–517, 2002.
9. M. J Blunt. Flow in porous mediapore-network models and multiphase flow. *Current opinion in colloid & interface science*, 6(3):197–207, 2001.
10. C. Goniva, C. Kloss, N. G Deen, J. AM Kuipers, and S. Pirker. Influence of rolling friction on single spout fluidized bed simulation. *Particuology*, 10(5):582–591, 2012.
11. A. Hager, C. Kloss, and C. Goniva. Towards an efficient immersed boundary method within an open source framework. In *Proc. of 12th International Conference on Multiphase Flow in Industrial Plants, Ischia (Napoli), Italy*, 2011.
12. C. Kloss, C. Goniva, A. Hager, S. Amberger, and S. Pirker. Models, algorithms and validation for opensource dem and cfd–dem. *Progress in Computational Fluid Dynamics, an International Journal*, 12(2-3):140–152, 2012.
13. H. G Weller, G. Tabor, H. Jasak, and C. Fureby. A tensorial approach to computational continuum mechanics using object-oriented techniques. *Computers in Physics*, 12(6):620–631, 1998.
14. N. V Brilliantov, F. Spahn, J-M Hertzsch, and T. Pöschel. Model for collisions in granular gases. *Physical review E*, 53(5):5382, 1996.
15. C Goniva, C Kloss, A Hager, G Wierink, and S Pirker. A multi-purpose open source cfd-dem approach. In *8th International Conference on CFD in Oil & Gas, Metallurgical and Process Industries SINTEF/NTNU. Trondheim, Norway*, pages 21–23, 2011.
16. Raad I Issa. Solution of the implicitly discretised fluid flow equations by operator-splitting. *Journal of computational physics*, 62(1):40–65, 1986.
17. B. P Muljadi, M. J Blunt, A. Q Raeini, and B. Bijeljic. The impact of porous media heterogeneity on non-darcy flow behaviour from pore-scale simulation. *Advances in Water Resources*, 95:329–340, 2016.
18. J. Kozeny. Uber kapillare leitung der wasser in boden. *Royal Academy of Science, Vienna, Proc. Class I*, 136:271–306, 1927.
19. P. C. Carman. Fluid flow through granular beds. *Trans. Inst. Chem. Eng.*, 15:150–166, 1937.
20. M. Uhlmann. An immersed boundary method with direct forcing for the simulation of particulate flows. *J. Comput. Phys.*, 209(2):448–476, 2005.
21. K. L. Johnson, K. Kendall, and AD Roberts. Surface energy and the contact of elastic solids. *Proc. R. Soc. Lond. A*, 324(1558):301–313, 1971.
22. KL Johnson. Continuum mechanics modeling of adhesion and friction. *Langmuir*, 12(19):4510–4513, 1996.
23. 10.14469/hpc/2232.
24. S. Henkes and B. Chakraborty. Statistical mechanics framework for static granular matter. *Physical Review E*, 79(6):061301, 2009.
25. Ø Johnsen, R Toussaint, K J Måløy, and E G Flekkøy. Pattern formation during air injection into granular materials confined in a circular hele-shaw cell. *Phys. Rev. E.*, 74(1):011301, 2006.
26. J. H. Seo and R. Mittal. A sharp-interface immersed boundary method with improved mass conservation and reduced spurious pressure oscillations. *J. Comput. Phys.*, 230(19):7347–7363, 2011.
27. M Kumar. *Sharp Interface Immersed Boundary Method with Improved Mass Conservation for Simulation of Moving Boundary Flows*. PhD thesis, IIT Patna, 2016.
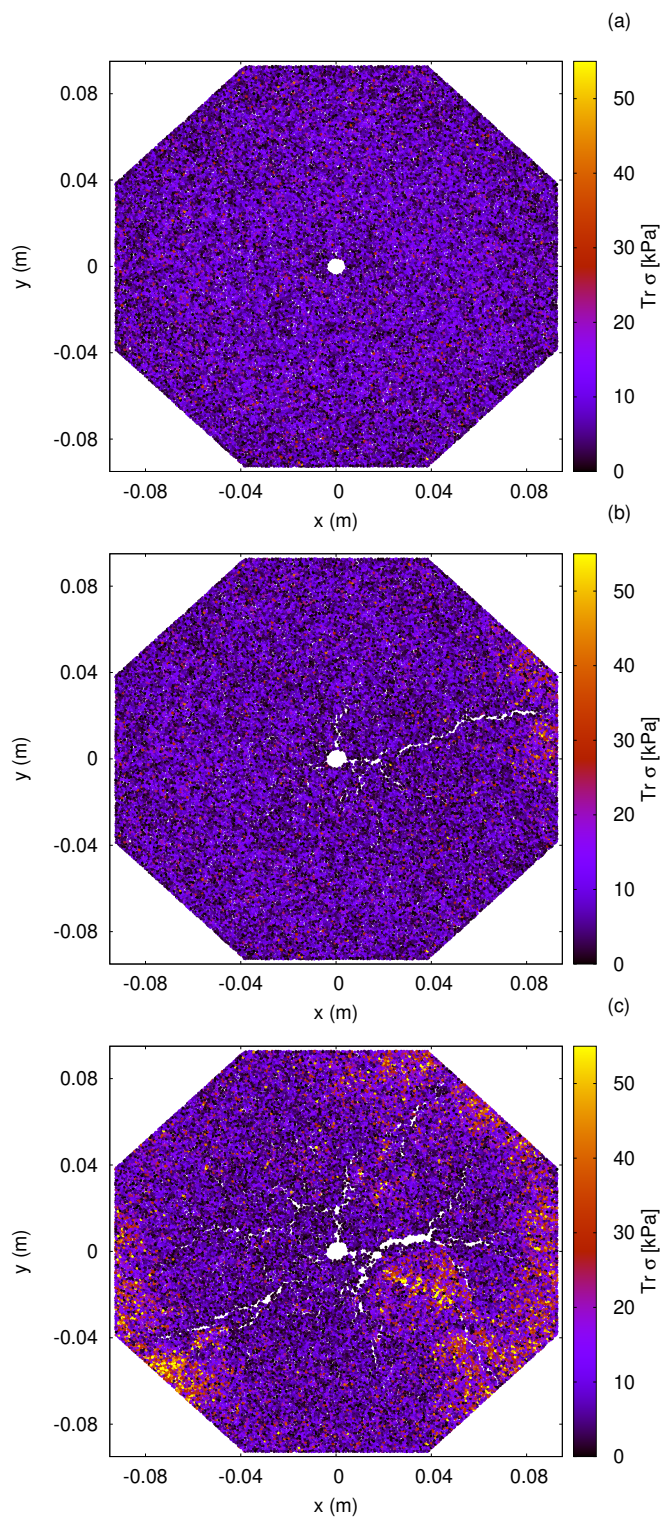
Fig. 8 (a) Numerical setup: $N = 50000$ spherical particles with sizes dispersed in the range of $1.8 - 0.9$mm and a mean diameter $< d_m >= 1.35$mm contained in an octagonal Hele-Shaw cell. Initial granular pack was created by compressing a loose granular material confined between two parallel plates (upper and lower plates) separated by a distance of about 2.5 times the mean particle diameter, by 8 piston like walls applying a constant force $F_{wall} = 4$N each, until the system reaches a mechanically stable static condition. Fluid is injected from the central inlet at different pressures, (a) $P_i = 8$kPa (b) $P_i = 22$kPa (c) $P_i = 28$kPa. Fluid pressure was zero at the outlet (octagonal perimeter) and a zero velocity boundary condition was set in the upper and lower plates.
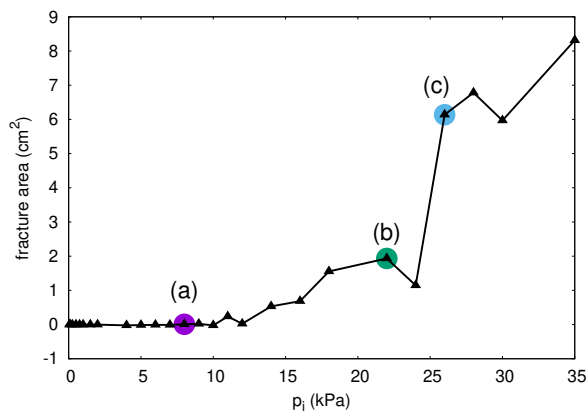
Fig. 9 Area of the "fracture" connected to the inlet as a function of the inlet pressure p_i. (a), (b) and (c) labels in the figure refer to the sub figures of Fig. 8