A coupled Discrete Element Lattice Boltzmann Method for the simulation of fluid-solid interaction with particles of general shapes

S.A. Galindo-Torres[a,b]

[a] *Geotechnical Engineering Centre. School of Civil Engineering, The University of Queensland, Brisbane QLD 4072, Australia*

[b] *National Centre for Groundwater Research & Training. School of Civil Engineering, The University of Queensland, Brisbane QLD 4072, Australia*

Abstract

In the present paper a coupling law between the Lattice Boltzmann Method (LBM) for fluids and the Discrete Element Method (DEM) is introduced to study problems involving the interaction between fluids and solid (rigid) structures. The DEM particles are modelled with the sphero-polyhedra method, allowing the representation of general shapes, even non-convex ones. The method is rigorously validated by an extensive simulation plan measuring the lift and drag coefficients for several geometries and comparing the results with previous studies. The method also includes some optimization techniques which are described in the paper. Finally, simulations of DEM particles with shapes of animals (dolphins and ducks) are carried out interacting with multiphase and multicomponent LBM fluids, and many realistic features of this interaction such as buoyancy, sonic wave propagation and fluid displacement are shown to qualitatively match observations. Some issues with the coupling law are identified for future research studies on the proposed method. The results show the potential of the method to be used in the simulation of a wide range of phenomena found in sectors such as the the mining and oil industries and disciplines such as medicine and biology.

Keywords: Lattice Boltzmann Method, Discrete Element Method

1. Introduction

It has always been an issue of any numerical modelling effort the fact that greater accuracy usually requires large amounts of details extracted from observations and fed into the model, which is often translated into larger computational requirements. Is not difficult to appreciate how with increasing computational power, the numerical models developed nowadays are including more and more details that were unmanageable years ago.

The computational power that researchers and engineers currently enjoy is enough to solve a broad range of models based on partial differential equations by means of methods

Email address: s.galindotorres@uq.edu.au (S.A. Galindo-Torres)

such as the Finite Element Method (FEM). Recently it has also been enough to abandon the continuous assumption and start studying the macroscopic response of many systems as a function of the microscopic interactions between the constitutive elements. Examples of such approaches are the Distinct Element Method (DEM) [1] introduced by Cundall for granular materials and the Molecular Dynamic Method (MD) [2] for gases and liquids. In such methods, a macroscopic constitutive equation is not introduced as in the case of FEM modelling. Instead, the modeller introduces the interactions between the particles and obtains the macroscopic response of the complex system. DEM and MD have been very successful tools for research since their results match, qualitatively, the response from controlled experiments.

Recently DEM has started to be used as a predictive tool [3, 4] despite the main difficulty of finding the value of the microscopic parameters from experimental set-ups. It is the author's opinion that as the computational power increases, DEM will be a viable option for engineering design at the field scale, and the advent of commercial DEM packages seems to follow this trend.

Some research efforts are focused in coupling DEM with other methods. One particular field of interest is the interaction between DEM and Computational Fluid Dynamics (CFD) methods. The need for such tool is evident for the modelling of a broad range of problems such as groundwater transport of colloid and contaminants, erosion and the stability of coastal foundations to name a few.

DEM has been successfully coupled with CFD schemes such as Smooth Particle Hydrodynamics (SPH) [5] and FEM solvers for the Navier Stokes equations [6]. These methods are great for large scale simulations, but lack some details of the fluid at the pore scale. A set of seminal papers by Owen and collaborators [7, 8, 9, 10] showed the coupling between DEM and the Lattice Boltzmann Method (LBM) by means of an immersed boundary method. LBM offers several advantages over other CFD schemes including the locality of the dynamic steps that facilitates the distribution of the domain into massive parallel systems. Such locality also facilitates the formulation of laws involving the interaction of the fluid with moving boundaries which at the ends translates into physically sound coupling schemes with the DEM [9]. It is easy to introduce multiphase and multicomponent systems into the LBM formalism by assigning different lattices to the different components and attractive or repulsive forces to model phase transition and immiscibility [11]. In this way many features at the pore scale such as the menisci and the capillary interaction are reproduced. It is also easy, due to the regular lattice configuration, to deal with obstacles of complex shapes [12] and therefore to cope with the realistic morphology of the porous medium.

For these reasons, the coupling DEM-LBM is a promising tool. It will potentially be able to tackle problems as complex as the mechanical response of unsaturated soils or to model industrial applications such as the hydraulic fracturing. But to be able to accurately address this kind of complexity, the classical DEM spherical element should be abandoned in favour of particles of general shape and with cohesion to model fracture phenomena. The author of the present work has implemented previously the sphero-polyhedra method to model the collision and fragmentation of particles with general shapes [13]. With Voronoi or Delaunay tessellations, solid blocks can be reproduced without voids and the cohesive parameters can

be tuned to reproduce any possible elastic moduli, Poisson's ratios and tensile strengths.

Due to the flexibility of the sphero-polyhedra method, it is a viable choice to expand the DEM-LBM formalism introduced by Owen *et al* . The presented research work was done with that aim. The sphero-polyhedra method generalises the widely used contact detection between spheres to the collision between polyhedra. In the same way it will be show how the sphero-polyhedra also generalises the contact law between the LBM fluid and the DEM particles introduced before in a simple and efficient way.

The paper is structured as follows: Sec. 2 describes independently the DEM and LBM and then finalizes with the coupling law between the two. Sec. 3 presents a series of validation examples for different geometries and compare the results with previous studies. In Sec. 4 some optimization techniques for the proposed law and their advantages are shown. A preliminary study combining DEM shaped like real animals and multiphase and multicomponent LBM fluids is shown in Sec. 5. Finally in Sec. 6 some conclusions and projections of the current work are presented.

2. The method

2.1. The DEM sphero-polyhedra approach

The sphero-polyhedra method was initially introduced by Pourning [14] for the simulation of complex-shaped DEM particles. Later, it was modified by Alonso Marroquin [15], who introduced a multi-contact approach in 2D allowing the modelling of non-convex shapes, and was extended to 3D by the author of this paper and his collaborators [16, 13, 17]. A sphero-polyhedron is a polyhedron that has been eroded and then dilated by a sphere element as seen in Fig. 1. The result is a polyhedron of similar dimensions but with rounded corners.

The advantage of the sphero-polyhedra technique is that it allows for an easy and efficient definition of contact laws between the particles. This is due to the smoothing of the edges of all geometric features by circles (in 2D) or spheres (in 3D).

To define the contact between two sphero-polyhedra P_1 and P_2, each solid is divided in a set of geometric features (edges, faces and vertices) and and overlapping check is performed between each possible pair. For simplicity, lets denominate the set of all the geometric features of the n-th particle as $\{G_n^i\}$. A distance function must be defined to check the overlapping between the i-th geometric feature of particle 1 and the j-th geometric feature of particle 2,

$$\mathrm{dist}(G_1^i, G_2^j) = \min\left(\mathrm{dist}(\vec{X}_i, \vec{X}_j)\right), \tag{1}$$

where $\vec{X}_i$ is a 3D vector that belongs to the set G_1^i. This means that the distance for two geometric features is the minimum Euclidean distance assigned to two points belonging to them.

Finding this distance between two geometric features is the core of the sphero-polyhedra method. The simplest case is the distance between two vertices which is the distance between two points in space. The distance between a vertex and an edge is the distance between the vertex and the point at the intersection of a ray passing trough the vertex and the edge and perpendicular to the latest. The distance between a face and a vertex is usually the distance

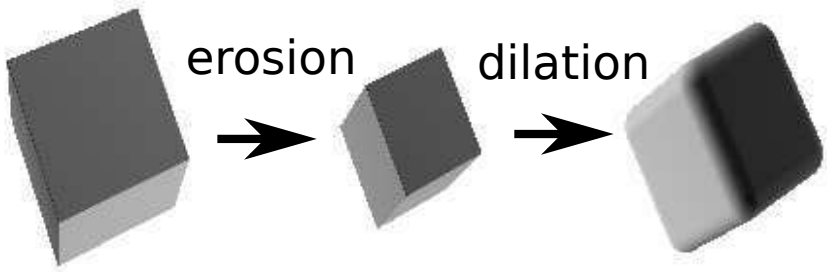


Figure 1: A 3D sphero-cube: Initially the cube is eroded or shrunk by a distance equal to the sphere radius, and then is dilated by the same sphere. After this morphological transformation the cube ends up having rounded corners.

between the vertex and a point that is connected to it by a line perpendicular to the face plane. In general, in this cases, finding the distance involves a linear or quadratic system of equations and no iterations are needed.

Since both particles are dilated by their sphero-radii R_1 and R_2, it can be said that there is a complete contact when the distance between the two geometric features is less than the addition of the corresponding radii used in the sweeping stage, i.e.:

$$\text{dist}(G_1^i, G_2^j) < R_1 + R_2, \tag{2}$$

and the overlapping distance $\delta(G_1^i, G_2^j) = R_1 + R_2 - \text{dist}(G_1^i, G_2^j)$ can be defined. Here, the advantage of the sphero-polyhedra technique becomes evident since this definition for contact based on the overlapping is similar to the one for the contact law of two spheres [3]. Fig. 2 shows the possible overlapping distances that are considered in the method between vertices and vertices, edges and edges and vertices and faces. With these definitions, the face face interaction is not needed.

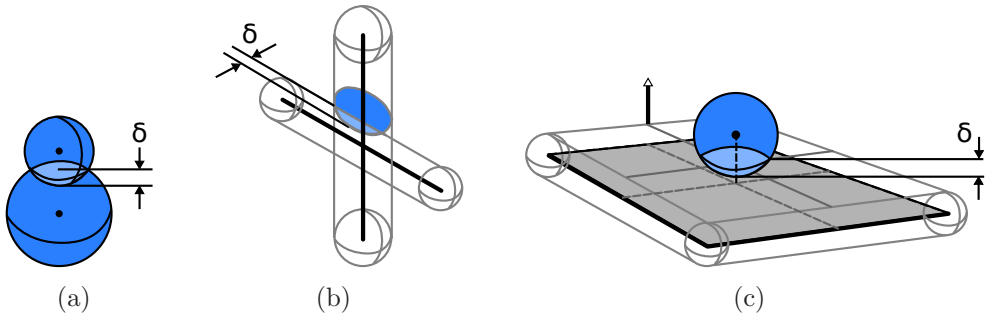


Figure 2: Definition of the overlapping distance $\delta(G_1^i, G_2^j)$: (a) vertex-vertex; (b) edge-edge; (c) vertex-face. No other contacts are needed to define the collision between polyhedra.

The minimum distance for the sets G_1 and G_2 is given by the Euclidean distance between two of their points $\vec{X_1}$ and $\vec{X_2}$. This allows for the definition of a normal vector $\vec{n}$ given by:

$$\vec{n}(G_1^i, G_2^j) = \frac{\vec{X_2} - \vec{X_1}}{\|\vec{X_2} - \vec{X_1}\|}. \tag{3}$$

With this definition, the same expression for the forces defined in conventional DEM codes can be easily employed in this method. For instance, a normal elastic force $\vec{F_n}$, proportional to the overlapping length, is defined according to

$$\vec{F_n}(G_1^i, G_2^j) = K_n \delta(G_1^i, G_2^j) \vec{n}(G_1^i, G_2^j), \tag{4}$$

where K_n is a parameter called the normal stiffness.

The previous normal force is defined for a pair of geometric features. The net elastic force is the addition of all these forces for each possible pair of geometric features. However, for general polyhedra, it is only necessary to consider the interactions between vertices and

faces and the interaction between edges and edges. The total force is the addition of all the forces for each pair of overlapping geometric features. The same applies for the total torque.

Friction and viscosity forces are also added to simulate inelastic collisions as explained in [13]. Finally, all these forces are added to obtain the net force over a particle. The forces that are going to be described in the following sections coming from the coupling with the LBM fluid are also added. Afterwards, Newton's second law is solved for which the Verlet algorithm is employed [18]. Each of these forces also provide the torque, where the point of application is at the middle of the line connecting the two contact points for each pair of geometric features that are in contact. With the torque, the angular dynamical equation of movement is solved with the Leap Frog algorithm [16, 19].

2.2. The Lattice Boltzmann Method

For the present study the Lattice Boltzmann D3Q15 scheme was used [20]. In this scheme the space is divided into a cubic domain where each cell has a set of probability distribution functions f_i, representing the density of fluid particles going through one of the 15 discrete velocities $\vec{\boldsymbol{e}}_i$ (see Fig. 3). The density ρ and velocity $\vec{\boldsymbol{u}}$ at each cell position $\vec{\boldsymbol{x}}$ can be determined by:

$$\begin{array}{ll} \rho(\vec{\boldsymbol{x}}) = & \sum_{i=0}^{15} f_i(\vec{\boldsymbol{x}}) \\ \vec{\boldsymbol{u}}(\vec{\boldsymbol{x}}) = & \frac{\sum_{i=0}^{15} f_i(\vec{\boldsymbol{x}})\vec{\boldsymbol{e}}_i}{\rho(\vec{\boldsymbol{x}})} \end{array} \tag{5}$$

Each distribution function has an evolution rule derived from the Chapman Enskog expansion of the Boltzmann equation [21],

$$f_i(\vec{\boldsymbol{x}} + \vec{\boldsymbol{e}}_i\delta t, t + \delta t) = f_i(\vec{\boldsymbol{x}}, t) + \Omega_{col}, \tag{6}$$

where $\vec{\boldsymbol{x}}$ is the position of the local cell, δ_t is the time step and Ω_{col} is a collision operator representing the relaxation processes due to the collision of the fluid particles. For this study, the widely accepted Bhatnagar-Gross-Krook (BGK) model for the collision operator [22] was used, which assumes that the collision processes drive the system into an equilibrium state described by an equilibrium function f_i^{eq},

$$\Omega_{col} = \frac{\delta t}{\tau}\left(f_i^{eq} - f_i\right) \tag{7}$$

with τ being the characteristic relaxation time. It has been shown that the Navier Stokes (NS) equations for fluid flow [23] are recovered if,

$$f_i^{eq} = \omega_i \rho \left(1 + 3\frac{\vec{\boldsymbol{e}}_i \cdot \vec{\boldsymbol{u}}}{C^2} + \frac{9(\vec{\boldsymbol{e}}_i \cdot \vec{\boldsymbol{u}})^2}{2C^4} - \frac{3u^2}{2C^2}\right) \tag{8}$$

and the kinetic viscosity of the fluid (ν) is given by,

$$\nu = (\tau - 0.5)\frac{\delta_x^2}{3\delta_t} \tag{9}$$

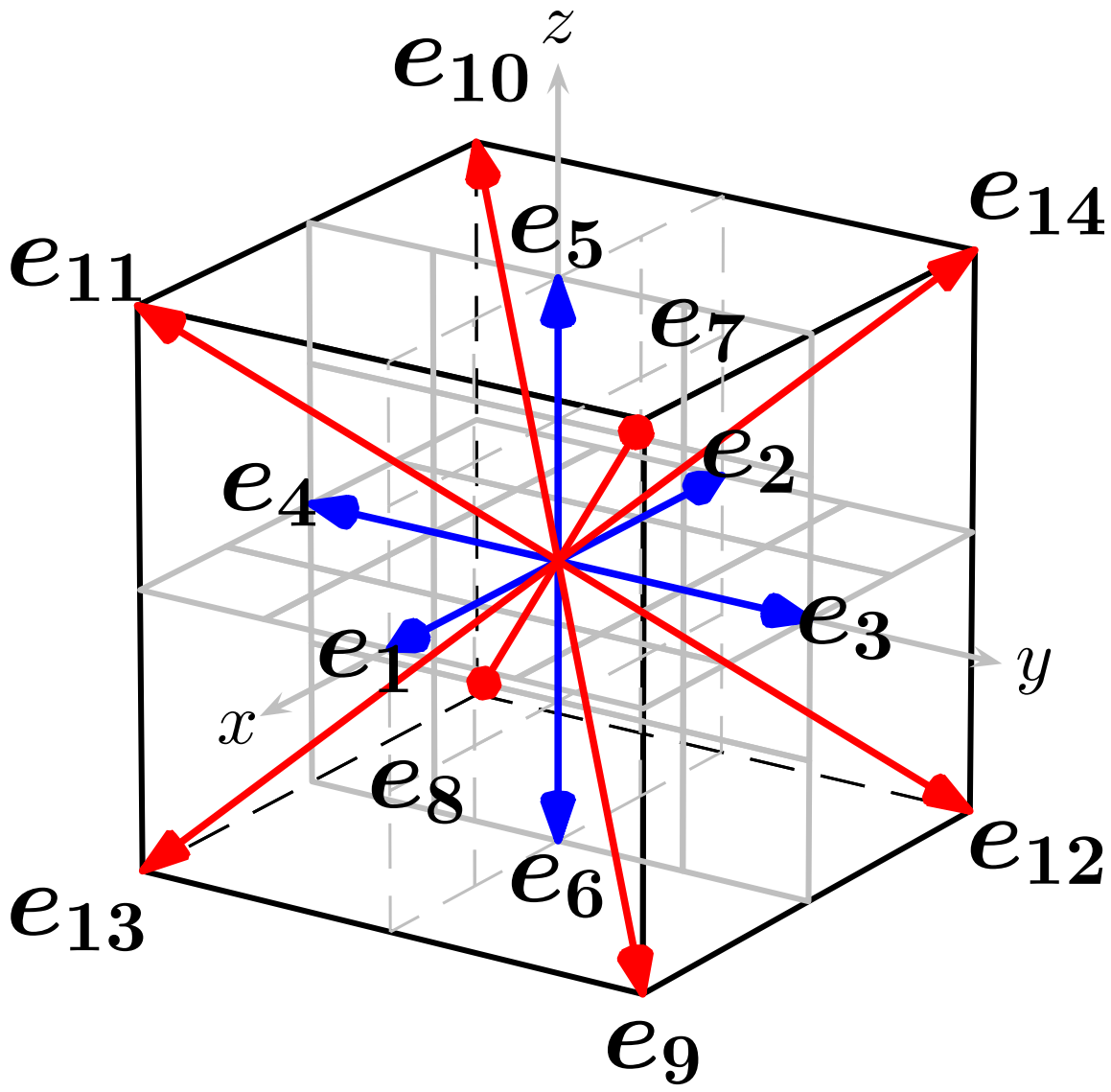


Figure 3: The LBM cell of the D3Q15 showing the direction of each one of the 15 discrete velocities

with $C = \delta_x/\delta_t$, a characteristic lattice velocity defined by the grid spacing δ_x. Eq. 9 imposes a constrain on the choice of τ, which must be greater than 0.5 for the viscosity to be physically correct. It is also known that values close to 0.5 produce unstable numerical behaviour [21] due to the non-linearity of the NS equations; hence it is advisable to keep its value close to one.

In order to model multicomponent and multiphase flows as well as body forces such as the gravity force [24], a net force is introduced for each cell. The net force $\vec{\boldsymbol{F}}$ modifies the velocity used in the calculation of the equilibrium function as follows,

$$\vec{\boldsymbol{u}}' = \vec{\boldsymbol{u}} + \frac{\delta_t \vec{\boldsymbol{F}}}{\rho}. \tag{10}$$

In the case of gravity, the force is simply $\vec{\boldsymbol{F}}_g = \rho\vec{\boldsymbol{g}}$ where $\vec{\boldsymbol{g}}$ is the gravitational acceleration [25].

Multiphase and multicomponent fluids are modelled in LBM with the Shan-Chen [26] approach by introducing attractive (for multiphase) or repulsive (for multicomponent) forces, representing the interaction between the fluid molecules. This forces are of the form:

$$\vec{\boldsymbol{F}}_{a,r} = -G_{a,r}\psi(\vec{\boldsymbol{x}}) \sum_{i=1}^{15} \omega_i \psi(\vec{\boldsymbol{x}} + \delta_t \vec{\boldsymbol{e}}_i)\vec{\boldsymbol{e}}_i, \tag{11}$$

where the sub-index r stands for repulsion and a for attraction, and the function ψ is a bonded potential for the case of multiphase simulations or the density value at the cell for multicomponent cases. Interested readers should refer to reference [11] for details on how to implement multiphase and multicomponent LBM simulations successfully.

2.3. The coupled method

With the LBM and DEM methods explained separately, the algorithm for the coupling can be explained in detail. The method is an extension of the one introduced by Owen *et al* [9] for spheres which is going to be introduced first.

2.3.1. Coupling law for spheres

One sphere moving through the lattice representing the fluid will intersect some LBM cells. Inside the spheres, the cells will be fully occupied by the DEM sphere. Outside of it, the volume fraction occupied by it will be zero. In the cells comprising the sphere surface, the volume fraction occupied by the DEM particle is between 0 and 1. The coupling law must ensure a smooth transition as the sphere invades and vacates the LBM cells during its movement. This coupling law modifies Eq. 6 to account for the volume occupation fraction ε,

$$f_i(\vec{\boldsymbol{x}} + \vec{\boldsymbol{e}}_i\delta t, t + \delta t) = f_i(\vec{\boldsymbol{x}}, t) + (1 - B_n)\left(\frac{\delta t}{\tau}\left(f_i^{eq} - f_i\right)\right) + B_n\Omega_i^s, \tag{12}$$

where B_n is a weight coverage function of ε_n for the $n-$th cell, and Ω_i^s is a collision operator representing the change of momentum due to the collision of the DEM sphere within the

LBM cell. The collision operator between LBM cells and DEM spheres takes the following form [27, 28],

$$\Omega_i^s = [f_{i'}(\vec{\boldsymbol{x}}, t) - f_{i'}^{eq}(\rho, \vec{\boldsymbol{v}}_p)] - [f_i(\vec{\boldsymbol{x}}, t) - f_i^{eq}(\rho, \vec{\boldsymbol{v}}_p)]\,, \tag{13}$$

with the symbol i' denoting the direction directly opposing the $\vec{\boldsymbol{e}}_i$ vector and $\vec{\boldsymbol{v}}_p$ the velocity of the DEM sphere at the cell position ($\vec{\boldsymbol{x}}$),

$$\vec{\boldsymbol{v}}_p = \vec{\boldsymbol{\omega}} \times (\vec{\boldsymbol{x}} - \vec{\boldsymbol{x}}_{CM}) + \vec{\boldsymbol{v}}_{CM}, \tag{14}$$

which depends on the sphere velocity $\vec{\boldsymbol{v}}_{CM}$, angular velocity $\vec{\boldsymbol{\omega}}$ and position $\vec{\boldsymbol{x}}_{CM}$ of the sphere's center of mass. This is a modification of the commonly used *bounce-back* condition within cells tagged as solids [21]. In fact if $\vec{\boldsymbol{v}}_p = \vec{\boldsymbol{0}}$ and $\varepsilon = 1$, the *bounce back* condition is recovered.

The weight coverage function has also a proposed form [9] in terms of the volume fraction ε,

$$B_n(\varepsilon) = \frac{\varepsilon_n(\tau - 1/2)}{(1 - \varepsilon_n) + (\tau - 1/2)}, \tag{15}$$

which has been empirically deduced to accurately simulate Poiseuille flow [28].

The total force $\vec{\boldsymbol{F}}$ over a DEM sphere is taken as the addition of the change of momentum given by each of the cells that the sphere covers,

$$\vec{\boldsymbol{F}} = \frac{\delta_x^3}{\delta_t} \sum_n B_n \left(\sum_i \Omega_i^s \vec{\boldsymbol{e}}_i \right). \tag{16}$$

And the torque $\vec{\boldsymbol{T}}$ is calculated similarly,

$$\vec{\boldsymbol{T}} = \frac{\delta_x^3}{\delta_t} \sum_n \left[(\vec{\boldsymbol{x}}_n - \vec{\boldsymbol{x}}_{CM}) \times B_n \left(\sum_i \Omega_i^s \vec{\boldsymbol{e}}_i \right) \right], \tag{17}$$

where it is assumed that the units of $\vec{\boldsymbol{e}}_i$ are the same units of the C parameter.

An important implementation issue is the calculation of the volume fraction occupied by the DEM spheres ε. This issue is extensively discussed in [9] where several schemes, and their strengths and drawbacks, are shown. The conclusion is that direct computation of the intersection volume is computationally expensive, and hence impractical for a coupled simulation framework. Several methods are proposed including the numerical integration of the intersection volume.

Herein an efficient way to calculate ε based on the computation of the portion of edges covered by the DEM sphere is illustrated and it will be validated in the following section. In Fig. 4 the algorithm to calculate the volume fraction approximately is shown. The algorithm is based on the length of an edge dwelling inside the sphere. If an edge is defined as the set of points between the two limits $\vec{\boldsymbol{p}}_0$ and $\vec{\boldsymbol{p}}_1$, then each of these points $\vec{\boldsymbol{p}}$ can be described in terms of a parameter s as,

$$\vec{\boldsymbol{p}} = \vec{\boldsymbol{p}}_0 + s(\vec{\boldsymbol{p}}_1 - \vec{\boldsymbol{p}}_0), \tag{18}$$

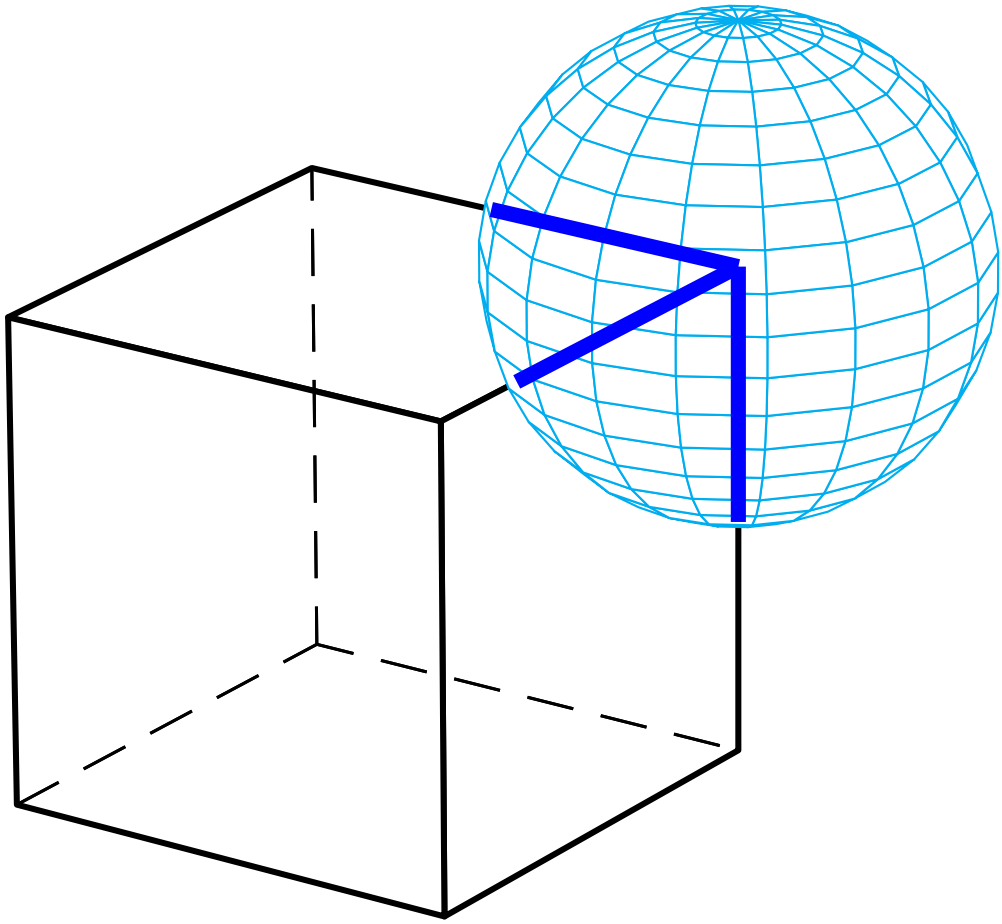


Figure 4: The DEM sphere interacting with an LBM cell (cube). The blue lines are the portion of edges covered by the sphere.

with s going from 0 to 1. Finding the length inside the sphere is reduced to finding the the intersection points with the sphere surface. This is expressed as the following equation:

$$(\vec{\boldsymbol{p}} - \vec{\boldsymbol{x}}_{CM}) \cdot (\vec{\boldsymbol{p}} - \vec{\boldsymbol{x}}_{CM}) = R^2, \tag{19}$$

with R the sphere radius. This equation gives a quadratic polynomial to solve for the parameter s with two solutions. If the solutions are imaginary then the sphere never intersects the edge. On the other hand if they are real and both between 0 and 1 then they represent the limit points of the intersection segment. Finally if one is less than 0 or greater than 1 then it is replaced by the edge limit points $\vec{\boldsymbol{p}}_0$ and $\vec{\boldsymbol{p}}_1$ accordingly. The intersection length l_e is computed for each of the 12 edges. Then it is compared with the total edge length to obtain the volume fraction,

$$\varepsilon = \frac{\sum_{e=1}^{12} l_e}{12\delta_x}. \tag{20}$$

2.3.2. Coupling law for sphero-polyhedra

One of the main advantages of the sphero-polyhedra method is the simplicity on how the collision law between to DEM spheres can be generalized to polyhedra. In the same way, the coupling law between a DEM sphere and a LBM cell can be extended to the sphero-polyhedra in a relatively easy way.

Following the reasoning of Eq. 1 there is a distance function between the center of the cell $\vec{\boldsymbol{x}}$ and a given geometric feature G^i of the DEM sphero-polyhedron,

$$\mathrm{dist}(\vec{\boldsymbol{x}}, G^i) = \min\left(\mathrm{dist}(\vec{\boldsymbol{x}}, \vec{\boldsymbol{X}})\right), \tag{21}$$

where $\vec{\boldsymbol{X}}$ is a point of the geometric feature. Finding the unknown point $\vec{\boldsymbol{X}}$ that minimizes the distance is reduced to solving a parametric equation of one parameter (similar to Eq. 19) for edges and two parameters for faces. In either case the solution does not require an iterative procedure improving efficiency.

Once the point of minimum distance $\vec{\boldsymbol{X}}$ is found, a sphere of radius equal to the particle's sphero-radius centred at $\vec{\boldsymbol{X}}$ is defined. From there, the algorithm follows the coupling law for spheres described in the previous section. Fig. 5 illustrates the procedure. The forces are supposed to be applied at the cell center and then the forces and torques follow Eqs. 16 and 17 in the same way.

Another fact that it is important to mention is that for most particles, checking the intersection of a given cell with each geometric feature may be redundant. In general, if a particle is a closed polyhedron (such as a sphero-cube or sphero-tetrahedron) then the intersection should only be calculated for the faces, ignoring the interaction with the edges. This is an important issue that needs to be taken into account in order to improve performance.

2.4. Details on the integration time step for the coupled method

As discussed in reference [9] the DEM-LBM framework suffers some of the drawbacks originated from the explicit integration schemes for both the DEM and the LBM. The

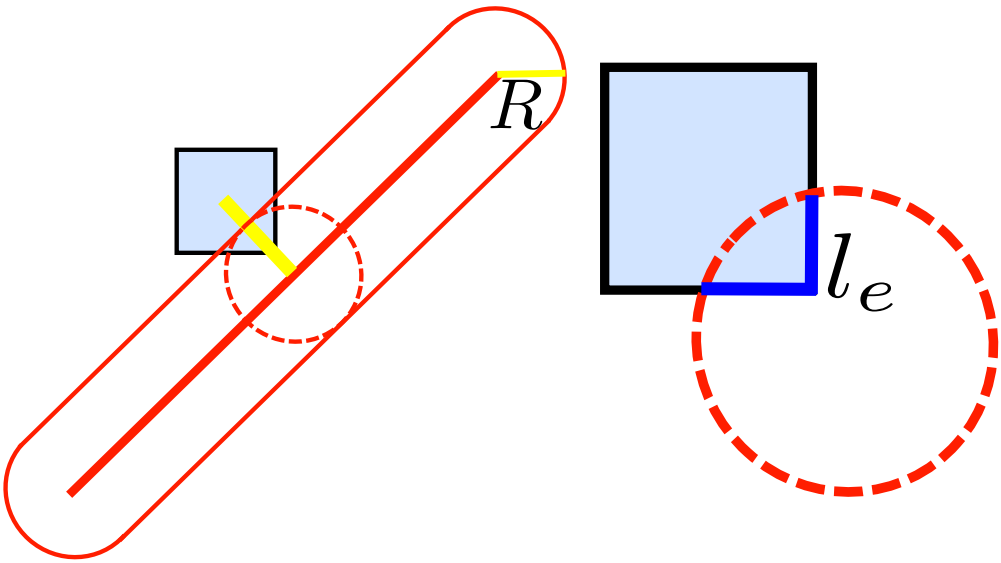


Figure 5: Interaction between an LBM cell and a sphero-edge. The first part of the algorithm (left) is to find the point inside the edge that minimizes the distance (yellow line) with the LBM cell center. Once found, a sphere is positioned at that point (right) with radius equal to the particle's sphero-radius. The coupling law for spheres is applied to that sphere by finding the length of intersection l_e between the cell edges and the sphere. Eq. 20 is then applied to find ε.

explicit nature of the Verlet integration for DEM provides a clear limit for the integration step [29],

$$\delta_t^{DEM} \sim \sqrt{\frac{M_{min}}{K_n}} \tag{22}$$

where M_{min} is the smallest value for the masses of all particles. However setting time steps smaller than this limit may not be enough, specially for particles with a shape significantly different than the sphere such as flat particles. As a general rule it is better to take a time step for the DEM integration that is one or possibly two orders of magnitude below this critical limit. On the other hand, the LBM time step is given by Eq. 9,

$$\delta_t^{LBM} = \left(\tau - \frac{1}{2}\right)\frac{\delta_x^2}{3\nu}, \tag{23}$$

where δ_x and ν are chosen to represent the physical fluid and τ determines the overall stability of the method. The true time step of the coupled method should be the smallest of the two. In the case that $\delta_t^{LBM} < \delta_t^{DEM}$ then the selection of the time step is straightforward since in DEM choosing a smaller time step improves the accuracy and does not alter the physical properties of the particles. On the other hand, the case when $\delta_t^{DEM} < \delta_t^{LBM}$ is more problematic, basically due to the fact that if the relaxation parameter τ takes values closer to 1/2, the LBM integration scheme becomes unstable. So in general it is a better strategy to avoid the last case and choose the critical LBM time as the overall time step by choosing a proper δ_x even at the cost of increasing the number of cells. In case this is impossible, a

sub cycle algorithm is also proposed in [9] where the DEM and LBM have different time steps and in the sub-cycles between LBM time steps the LBM is frozen and the dynamics of the DEM are integrated. For the simulations presented in this paper this last case was avoided since it was possible to always set $\delta_t^{LBM} < \delta_t^{DEM}$. However, this sometimes meant lowering the value of the stiffness K_n to unrealistic levels. A solution to this may be the use of an implicit integration scheme which is already proposed in the literature [30] where the DEM time step is independent from the stiffness. This should be the focus of future studies refining the approach introduced in this paper.

3. Validation

3.1. Validation for spheres

To validate the method, a comparison with the work carried out by Owen *et al* [9] is done to measure the drag coefficient of a sphere immersed into the fluid as a function of the Reynolds number Re. The LBM domain is formed by a lattice of $240 \times 60 \times 60$ cells. The lattice size step is $\delta_x = 0.004$m. The DEM sphere is placed at the domain's center and it has a radius of 0.036m which is equivalent to 9 LBM cells. The fluid density was taken as $\rho = 1000\text{kg/m}^3$, the kinematic viscosity ν is $10^{-4}\text{m}^2/\text{s}$ and the time step $\delta_t = 1.6 \times 10^{-2}$s. Periodic boundary conditions are applied to the three directions and a constant body acceleration of $7.81 \times 10^{-5}\text{m/s}^2$ is applied to the LBM cells allowing to measure the drag coefficient over a wide range of Reynolds numbers. Fig. 6 shows a plane slice plot with the velocity field surrounding the DEM sphere for a Reynolds number of 30 preventing the formation of eddies behind the sphere.

To calculate the Reynolds number, the sphere's diameter D is used as the characteristic length, and the velocity is taken as the average velocities of the unoccupied cells v_{ave},

$$Re = \frac{v_{ave} D}{\nu}. \tag{24}$$

The drag coefficient C_D is calculated by the reaction force x component, F_x over the sphere (Eq. 16) and the average fluid density ρ,

$$C_D = \frac{2F_x}{\rho v_{ave}^2 \pi D^2/4}. \tag{25}$$

Fig. 7 shows the results for the drag coefficient as a function of Re. For comparison, the results from the previous study of Owen *et al* [9] are shown as well. Additionally, an empirical correlation [31] found with experimental data is also presented:

$$C_D \approx \frac{24}{Re} + \frac{6}{1+\sqrt{Re}} + 0.4 \tag{26}$$

As can be seen, the method to calculate the volume fraction ε from Eq. 20 gives very similar results as the method used by Owen *et al* [9]. It is also relatively easy to implement requiring few calculations to compute ε. One point of concern is when the sphere intersects

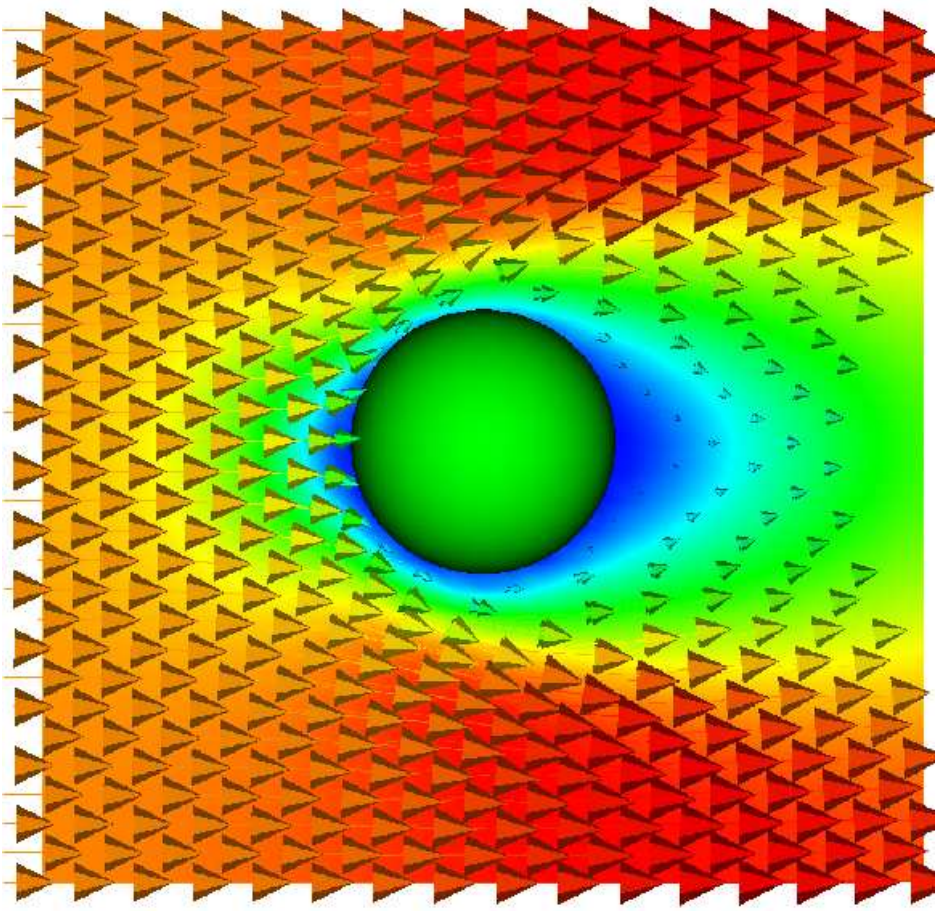


Figure 6: Cross sectional contour plot of the fluid velocity field surrounding a DEM sphere for $Re = 30$. The colourmap is proportional to the fluid velocity.

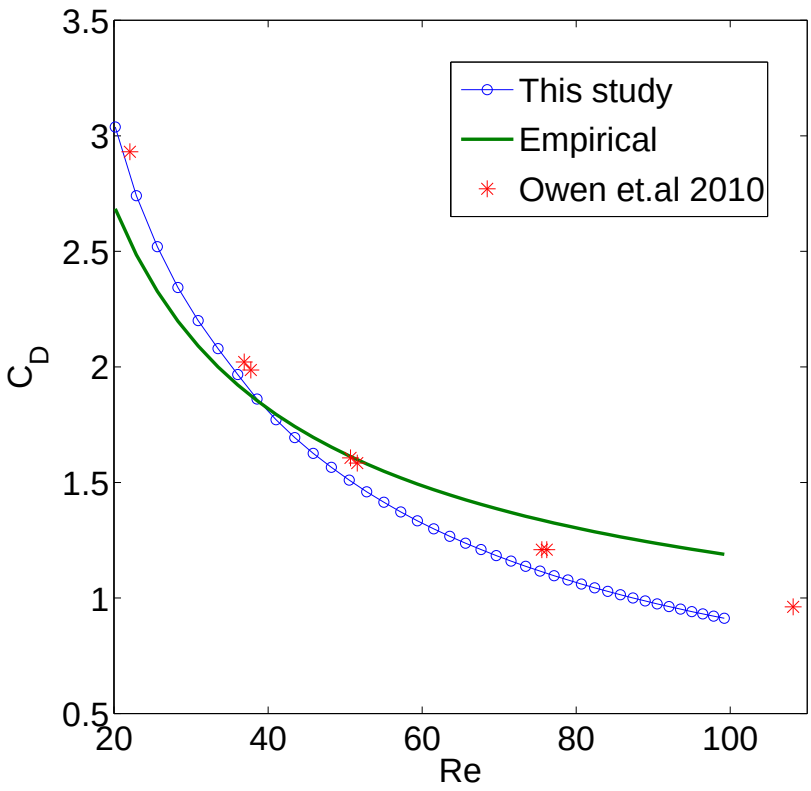


Figure 7: Sphere drag coefficient C_D as a function of the Reynolds number Re.

the LBM cell but no its edges. In this case although the real volume fraction is non-zero, Eq. 20 will give $\varepsilon = 0$. This is usually solved by ensuring that the sphere covers a wide range of LBM spheres. In the present case the radius of the sphere was 9 LBM cells which seems to be enough to reproduce reported results with similar methods, eliminating this problem.

Another validation for spheres is the simulation of the Magnus effect. The Magnus effect is a lift force applied over a spinning sphere due to the fluid flow asymmetry introduced by the angular velocity. Fig. 8 shows the velocity field disturbed by the spinning sphere. The sphere is spinning in a counter-clockwise direction, dragging the fluid close to it. The net effect is a decrease in the fluid's velocity at the top portion of the sphere due to this drag and a velocity increase at the bottom. If the system is considered as an inefficient air pump, air will build up at the top causing higher pressure at this point. This pressure gradient pointing downwards generates a force over the spheres in the same direction. The Magnus effect is commonly found in ball sports such as baseball and soccer where it gives the ball a distinct trajectory. With this force, a lift coefficient can be defined in complete analogy to the drag coefficient,

$$C_D = \frac{2F_y}{\rho v_{ave}^2 \pi D^2/4}, \tag{27}$$

Hölzer and Sommerfeld [32] conducted a number of simulations on the Magnus effect with a similar LBM scheme, without coupling it with the DEM. A dimensionless variable called the particle's spin number S_{Pa} is defined for comparison with the results of this previous study,

$$S_{Pa} = \frac{\omega D}{2v_{ave}}, \tag{28}$$

with ω the sphere's angular velocity.

Simulations varying S_{Pa} were conducted for $Re = 30$ a value that ensures that eddies do not appear in the simulation. Fig. 9 shows the dependence of the lift coefficient produced by the Magnus effect. It also shows the results obtained by Hölzer and Sommerfeld, and a fairly good match with the results from this study can be appreciated. For this small Reynolds number range the dependence of C_L on S_{Pa}, and therefore on the angular velocity, appears to be linear.

3.2. Validation for sphero-polyhedra

The same simulation domain was used to test the validity of the law for sphero-polyhedra. Initially a single sphero-cube is placed at the domain's center. Fig. 10 shows the simulation setup. The cube size is fixed to ensure that the equal volume sphere has a radius of 9 LBM cells. The sphero-radius of the particle is set at 5% of the cube length. As can be seen, if the cube is rotated along the z axis (perpendicular to the page) an angle θ, the flow loses its symmetry and a force appears along the y axis. A previous study conducted by Hölzer and Sommerfeld [32] has calculated the drag coefficient using LBM simulations, for several geometries including cubes. In this previous study a different coupling law (not a full DEM-LBM coupling) was used and the drag and lift coefficients for several values of Re are reported. Figs. 11 and 12 compares the results obtained in this study with the results from Hölzer and Sommerfeld for $Re = 30$ and different values of θ.

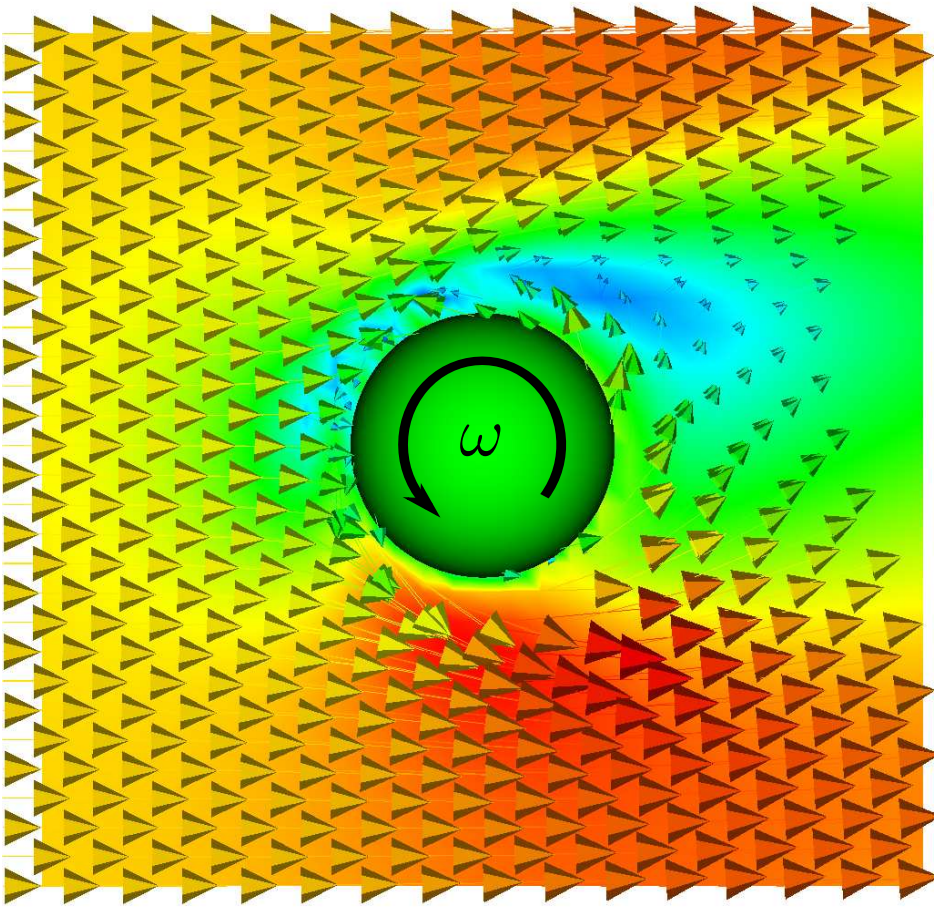


Figure 8: Cross sectional contour plot of the fluid velocity field surrounding a spinning DEM sphere with an angular velocity ω. The colourmap is proportional to the fluid velocity

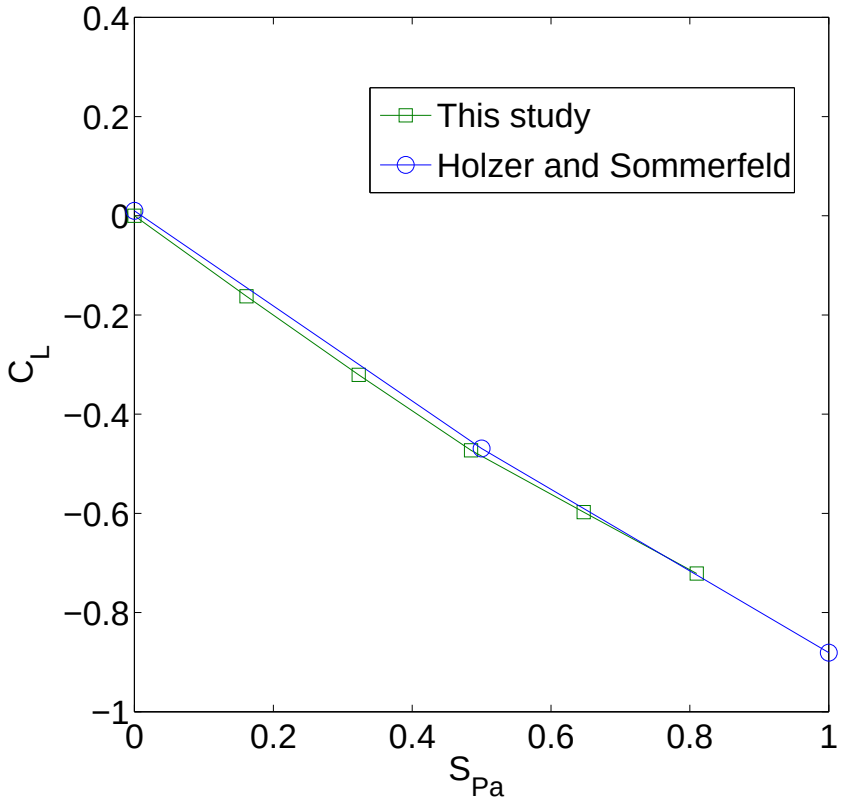


Figure 9: Sphere lift coefficient C_L as a function of the particle's spin number S_{Pa}.

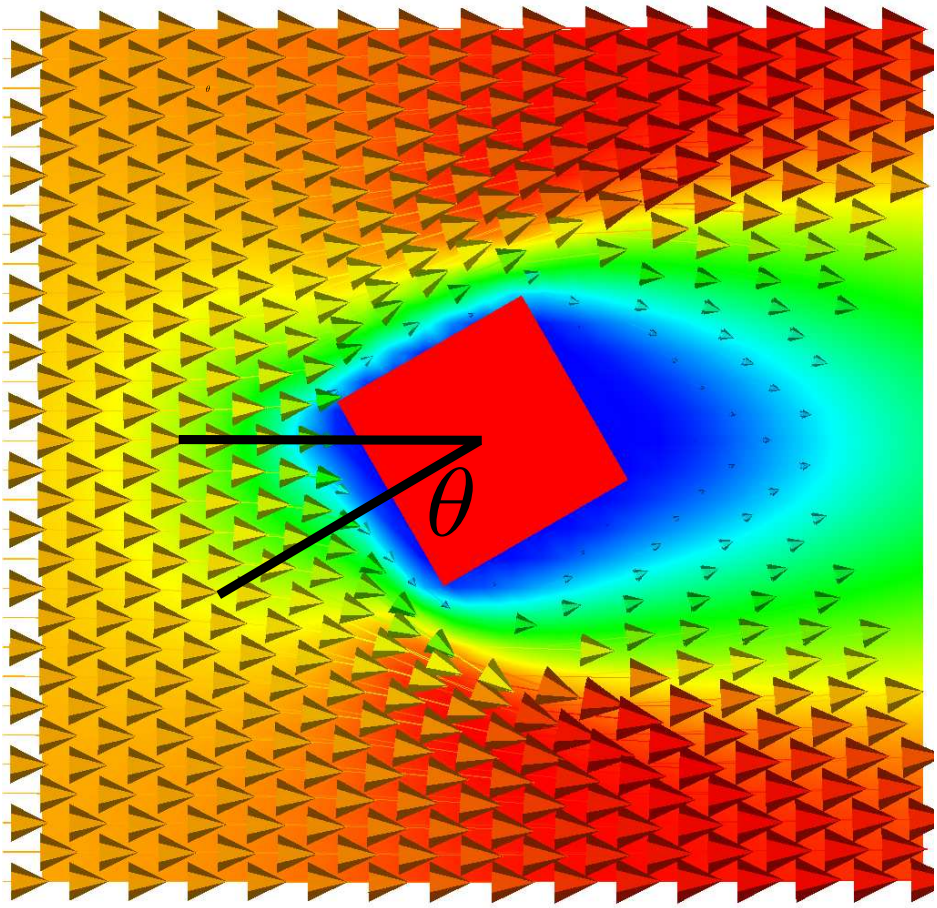


Figure 10: Cross sectional contour plot of the fluid velocity field surrounding a DEM sphero-cube of the xy plane (camera parallel to z axis). The colourmap is proportional to the fluid velocity.

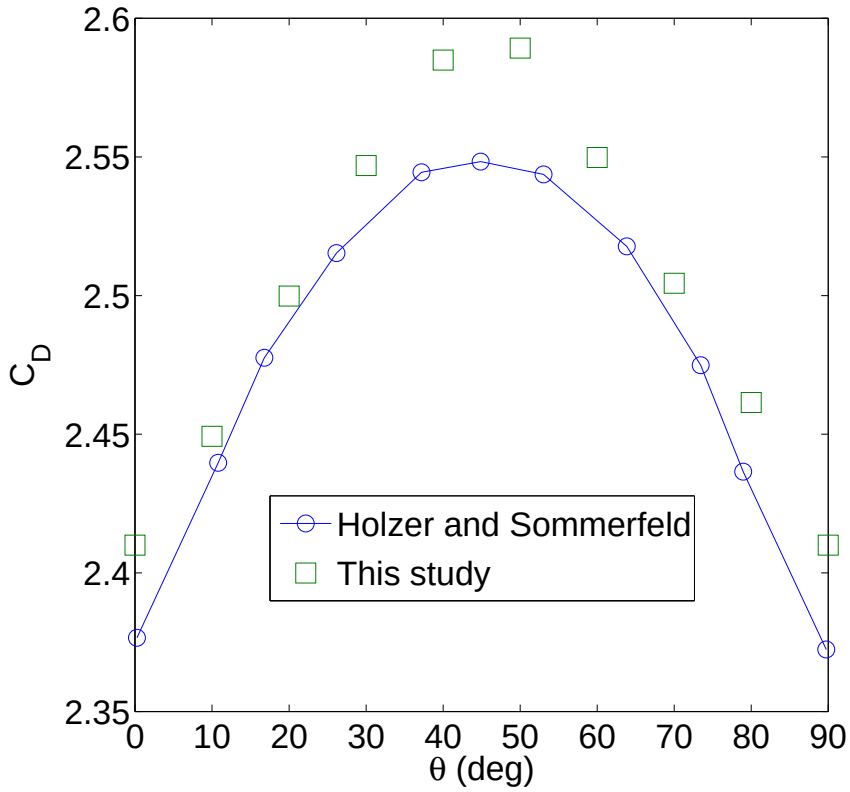


Figure 11: Sphero-cube drag coefficient C_D as a function of the θ angle for $Re = 30$.

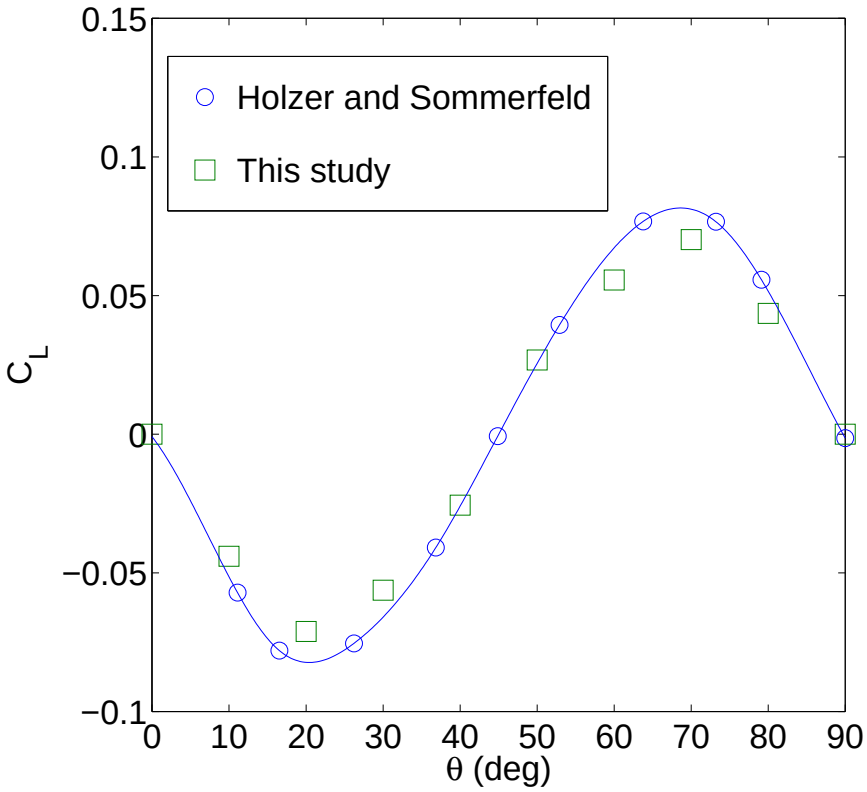


Figure 12: Sphero-cube lift coefficient C_L as a function of the θ angle for $Re = 30$.

Both studies show a similar value for both coefficients with a maximum difference of 3%. Both studies show a maximum value for C_D at $\theta = 45°$. At this point, the cube's area perpendicular to the flow direction is greater than when $\theta = 0°$ or $\theta = 90°$, increasing the overall drag. The lift coefficient is highly affected by θ. At three points is equal to 0 since, at this angles, the cube has a mirror symmetry with the x axis. For both studies, C_L take extreme values at approximately $\theta = 23°$.

A final simulation is presented of two sphero-cubes colliding with each other. The parameters are changed from the previous simulation to ensure that the dynamic viscosity of the fluid ($\mu = \nu\rho$) is kept small and the particles do not suffer much dissipation. The fluid density is set to 100kg/m^3 while the solid density is still 3000kg/m^3. The kinetic viscosity was lowered to $\nu = 1.5 \times 10^{-5}$m^2/s, the relaxation parameter was kept as $\tau = 0.6$ and the space step $\delta_x = 0.001$m. This gives a time step of $\delta_t^{LBM} = 2.2 \times 10^{-3}$s. The cubes have dimensions of 10 LBM cells (1 cm) and the collision stiffness $K_n = 21$N/m which gives a critical time step of $\delta_t^{DEM} = 0.01$s which is larger than the LBM time step. The parameters are set for the collision to be perfectly elastic in absence of fluid and the particles are frictionless. Both particles are given an initial velocity of 1cm/s in opposite directions. These velocities ensure a Reynolds number close to 10.

Fig. 13 shows four snapshots of the collision simulation at different time steps. As can be seen in the snapshots and the video attached as supplementary material, the movement of the sphero-cubes leaves a clear velocity trail behind them and a circulation vortex is formed at the sides of each cube. The collision suddenly decelerates both particles producing a shock wave that can be appreciated at the second snapshot. The wave travels at the lattice speed of sound ($C/\sqrt{3}$). Then the angular momentum over the particles gained by the collision produces a rather complex flow pattern characterised by further vortices. Finally at the last slide most of the energy is dissipated due to the viscosity of the fluid and the particles are drifting apart with a significant lower velocity.

As explained above, the only forces over the particles are the elastic collision force and the fluid forces due to the LBM. There are no artificial dissipative terms added to the particles' net force as is usually implemented in DEM simulations to ensure the convergence of the simulation and the presence of inelastic collisions [13]. Immersed into the LBM, the DEM particles do not need these dissipative terms since the fluid dissipates the DEM kinetic energy. Fig. 14 shows the total kinetic energy (linear and rotational) for both particles as a function of time. It decreases as is expected for a particle immersed in fluid with the higher dissipation occurring at the beginning due to the fact that the drag coefficient C_D is higher at lower Reynold numbers. The collision is clearly identified as the point where the kinetic energy has suddenly decreased since at this point the particles are in contact and are exchanging kinetic energy for potential energy at the collision spring. The rate of energy dissipation is lower around the collision time span due to the feedback provided over the particles by the two opposite flow directions. At the end of the simulation more than 80% of the initial energy has been dissipated by the LBM fluid.

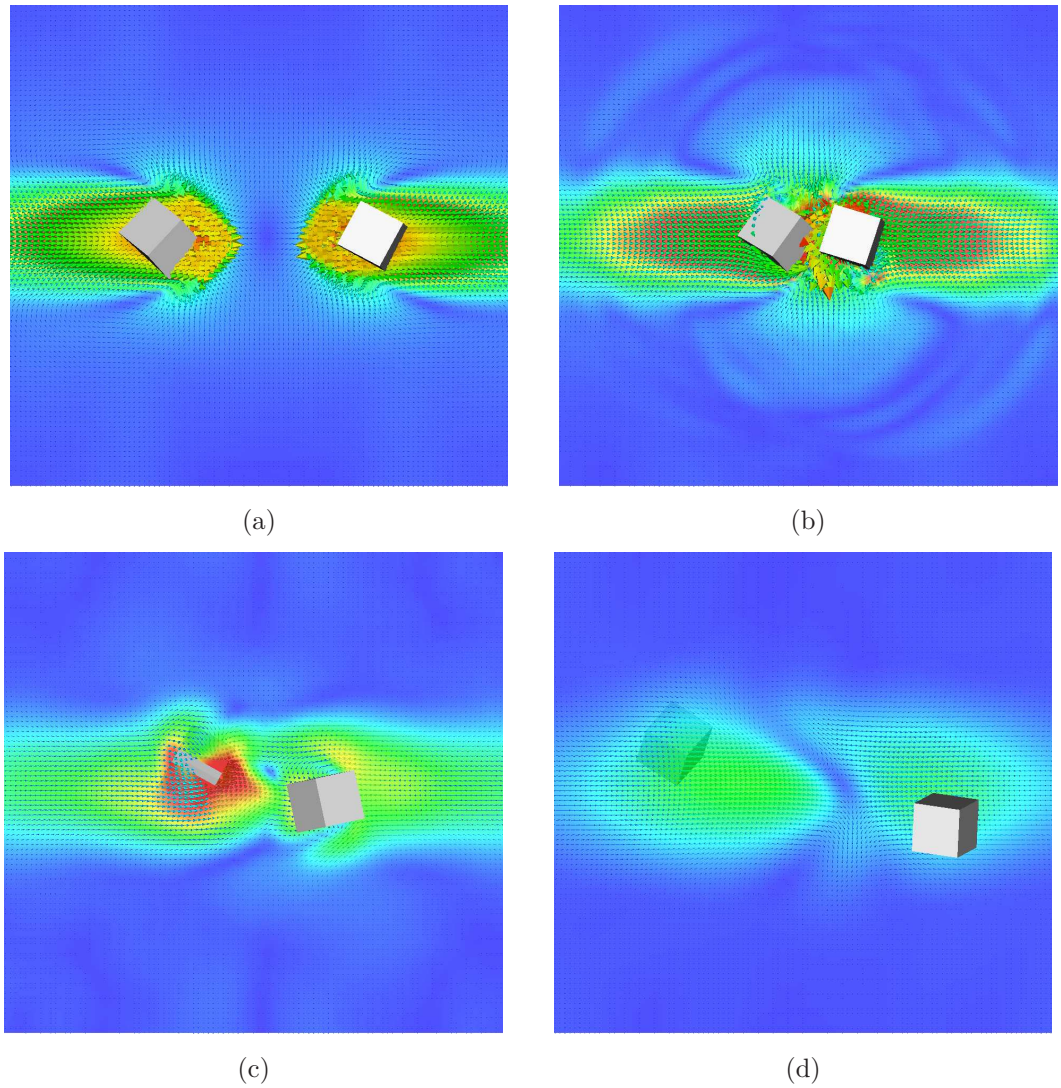


Figure 13: Snapshots at different times of two DEM sphero-cubes in a collision course. The color map and arrows represent the velocity field. The video is attached as supplementary material.

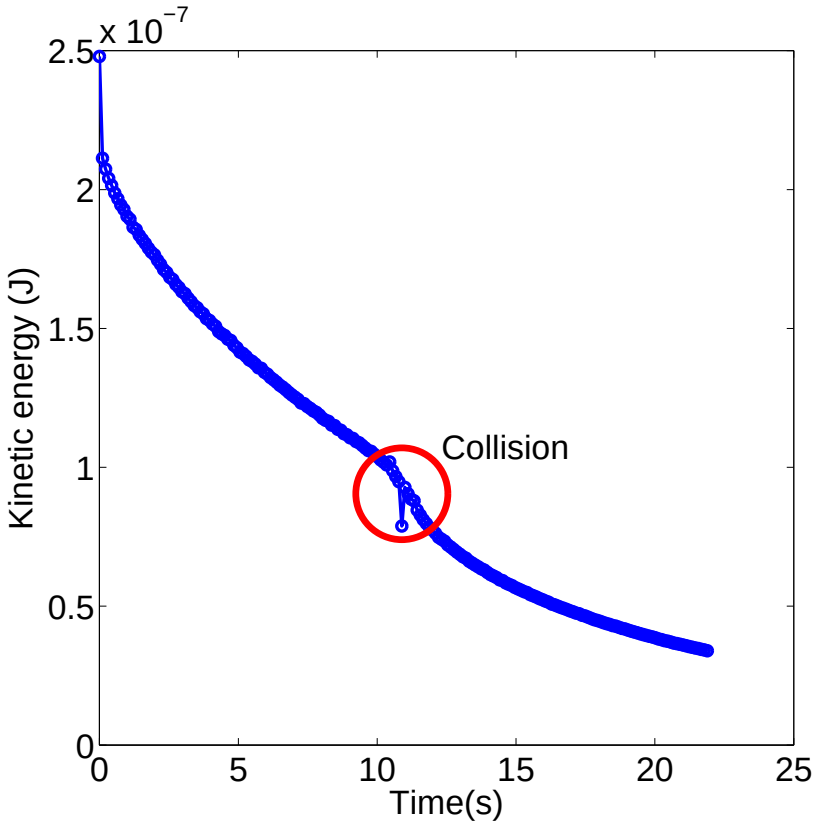


Figure 14: Total kinetic energy of the DEM sphero-cubes in time. The total kinetic energy comprises the linear and rotational energies. The time span where the collision occurs is marked with a circle in the graph.

4. Performance issues of the proposed coupling model

It is easy to identify the step of finding the distance between the LBM cell and the geometric feature of the sphero-polyhedron Eq. 21 as the efficiency bottleneck of the proposed coupling scheme. Finding the volume fraction ε for each cell and each geometric feature can quickly impair the algorithm performance. A Verlet list algorithm, similar as the one used to speed up DEM simulations, is suggested to address this issue [33]. Basically at the beginning of the simulation a list of pairs between cells and geometric features is formed by checking which pairs have a distance (defined by Eq. 21) smaller than a parameter α. During the simulation the only cell-geometric feature pairs checked for intersections are the ones belonging to this list. The list is updated if one of the particles' vertices moves more than $\alpha/2$ from the position were the list was updated the last time. In general the choice of α depends on the problem [33]. If the mean velocity of the particles is supposed to be high then a large value of α is advisable to avoid the computationally expensive Verlet list update step. Otherwise, for quasi stationary particles, a small value for α ensures few intersection checks per time step. In the previous simulation a value for α equal to 10 LBM cells was chosen. The method is illustrated in Fig. 15.

The Verlet list algorithm ensures that the computational complexity of the coupling scheme does not depend strongly on the DEM shape complexity. In Tab 1 the average time for the three geometries considered in the previous section is shown. The simulations were carried out in a 8 core Intel(R) Xeon(R) CPU L5520 @ 2.27GHz running for 60000 time steps with a $240 \times 60 \times 60$ LBM cells grid. The shape complexity sightly increases the computational requirements, but not by a significant amount due to the Verlet lists.

Table 1: Simulation times for different DEM shapes

Sphere	8h43m
Sphero-tetrahedron	9h54m
Sphero-Cube	10h7m

However, if the size of the particle is large enough to include many LBM cells in a particular list, then the computation speed is going to decrease. Fig. 16 shows the final times for a simulation using a sphero-cube with a variable diameter D for its volume equivalent sphere. As the size of the obstacle grow, more LBM cells are allocated into the Verlet list for each geometric feature increasing computational complexity. This study shows that for regular shapes such as sphero-cubes and sphero-tetrahedra a value of 9-10 LBM cells for the diameter of the equivalent volume sphere is a fair compromise between an accurate prediction of the flow effects and computational performance.

5. Implementation details of the DEM-LBM coupling method with multiphase and multicomponent fluids

In this section, a less rigorous simulation is going to be performed using a multiphase and multicomponent LBM. In the LBM literature it is common to express the relevant

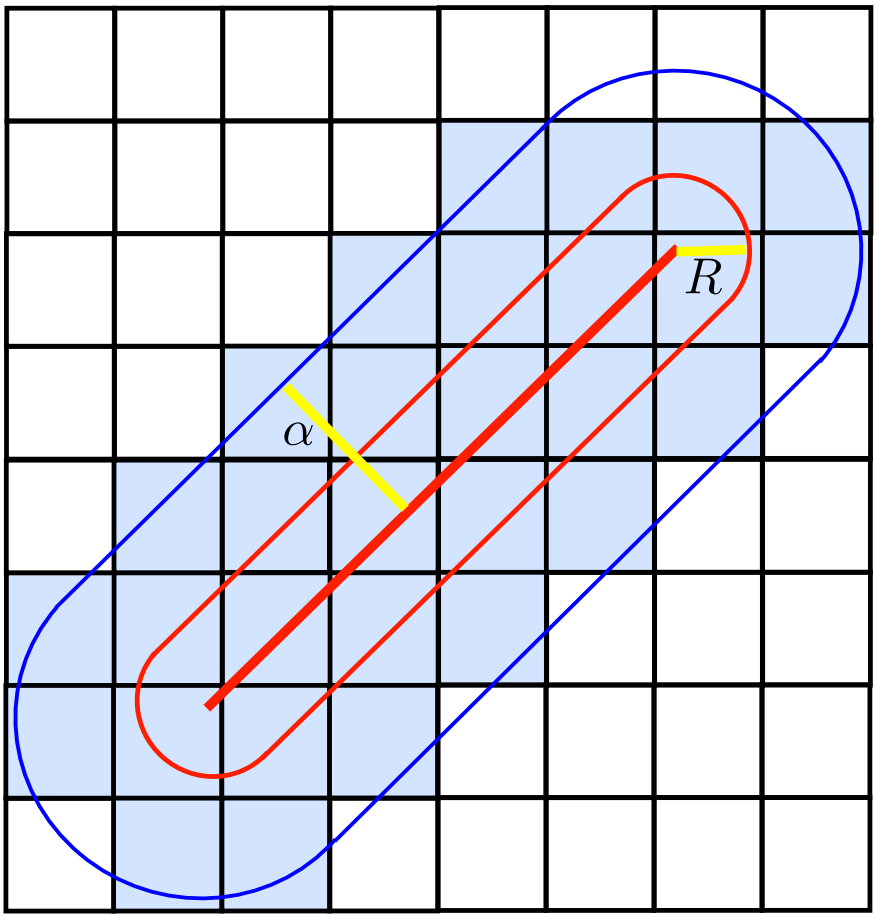


Figure 15: The Verlet list method applied to the coupling scheme. The sphero-edge (red line center) has a sphero-radius R and is immersed into the LBM grid. The method described in Sec. 2.3.2 is applied to the cells inside the Verlet *halo* (blue area). This Verlet *halo* has a characteristic distance α. The list are updated when at least one of the limit points of the sphero-edge moves further than $\alpha/2$ from the initial position.

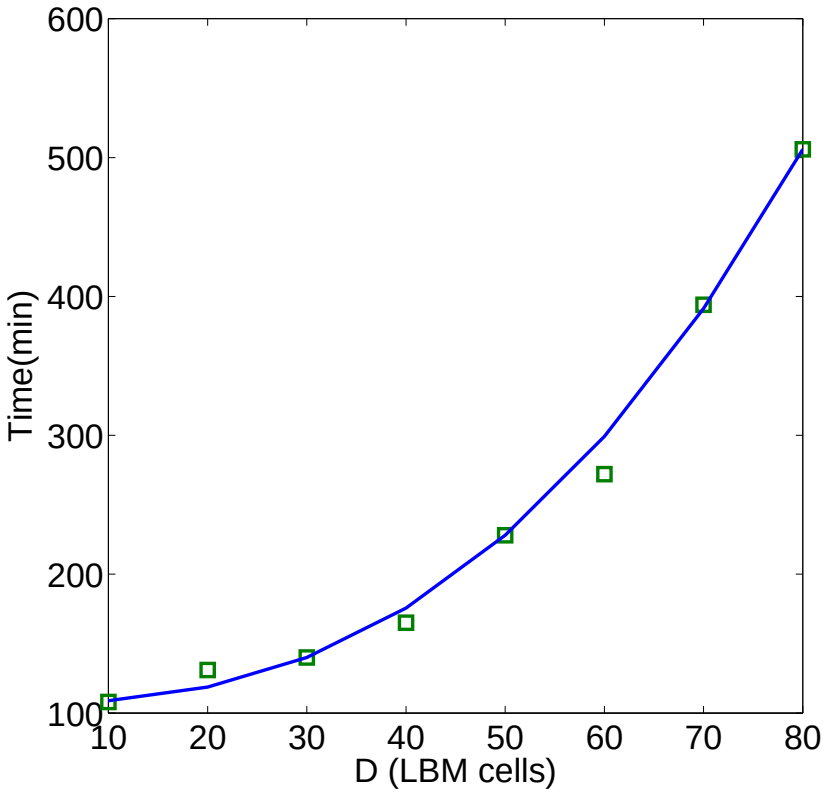


Figure 16: Computation time for a simulation of a sphero-cube with a varying diameter of its volume equivalent sphere D. The number of time steps is 6000. A power law is fitted with an exponent of 2.54.

quantities in terms of lattice units (the system in which $\delta_x = \delta_t = \rho_u = 1$ with ρ_u the units of density) instead of physical units. For easy comparison and implementation, the parameters of the LBM scheme used in this part are going to be expressed in both unit systems. The conversion between units is easy once the grid size δ_x, the time step δ_t and the density units ρ_u are properly defined with dimensional analysis. The simulation will involve a LBM *swimming pool* and a DEM particle with the shape of a dolphin diving into it.

First the system of two fluids (represented by one LBM grid each) is defined. The domain has dimensions of $100 \times 100 \times 200$ LBM cells, totalling 4000000 cells for both fluids with a grid size of $\delta_x = 0.02$m. Gravity is defined as pointing in the $-z$ direction and has a value of $0.001\delta_x/\delta_t^2 = 9.8\text{m/s}^2$ which defines the time step as $\delta_t = 0.0014$s and the speed parameter $C = 14.28$m/s. This gravity force is applied to both LBM fluids and to the DEM dolphin. The value for the relaxation constant $\tau = 1$ gives a kinetic viscosity of $4.7 \times 10^{-4}\text{m}^2/\text{s}$ which is two orders of magnitude greater than the real viscosity of water. One alternative to have a more realistic viscosity would be to define a finer grid size (two orders of magnitude finer) but then the number of cells will be one million times larger which is unmanageable with the average computational power found nowadays.

For each fluid the attractive constant G_a (Eq. 11) is defined, which helps to define its properties. For the fluid that is representing the liquid G_a^w is $-200\rho_u/\delta_x^2\delta_t^2$. In a previous study [11], the author found that this value for the attractive contact produces a liquid phase with an average density of $1119.2\rho_u$ (although it changes drastically due to the high compressibility of the liquid phase). Therefore, to obtain a liquid density of 1000kg/m^3, $\rho_u = 0.8935\text{kg/m}^3$. The attractive constant for the second fluid G_a^g is set to zero. The repulsion constant G_r (Eq. 11) controlling the immiscibility of both fluids is set to $0.001\delta_x/\delta_t^2$. The domain is filled up to half its height with the liquid fluid at a density of $2300\rho_u$. The reason that it is filled with a higher density than the equilibrium one ($1119.2\rho_u$) is due to the compressibility of the liquid phase under gravity which will be shown later on. In these cells the second fluid is also introduced but with a considerably lower density ($0.001\rho_u$) since due to stability of the LBM algorithm no cell can have zero density. In the upper part of the domain both fluids are also introduced but the gaseous fluid with $10\rho_u$ and the liquid fluid with $0.001\rho_u$.

The DEM sphero-dolphin is built from triangular mesh of 469 triangles measuring approximately 1.2m= $60\delta_x$ from tip to tail and with a density of 2000kg/m^3 and an sphero-radius of $1\delta_x$. The domain is enclosed by a box formed by six sphero-planes that can interact with the dolphin to prevent it from leaving the dolphin. The stiffness constant $K_n = 60000\rho_u\delta_x^3/\delta_t^2 = 219\text{kN/m}$ keeps the critical time step δ_t^{DEM} higher than the LBM time step although it makes for an extremely soft pool bottom. Fig. 17 shows the initial setup. The sphero-dolphin is initially performing a somersault by imposing an angular velocity of $\omega = 0.005\text{rad}/\delta_t = 3.5714\text{rad/s}$.

Fig. 18 shows four snapshots of the dolphin going into the water and colliding with the DEM sphero-plate at the bottom. For a more detailed view, please refer to the video attached as supplementary material.

One important issue is the question of fluid displacement by the DEM particle. In fully saturated incompressible conditions, as the ones imposed during the drag and lift coefficient

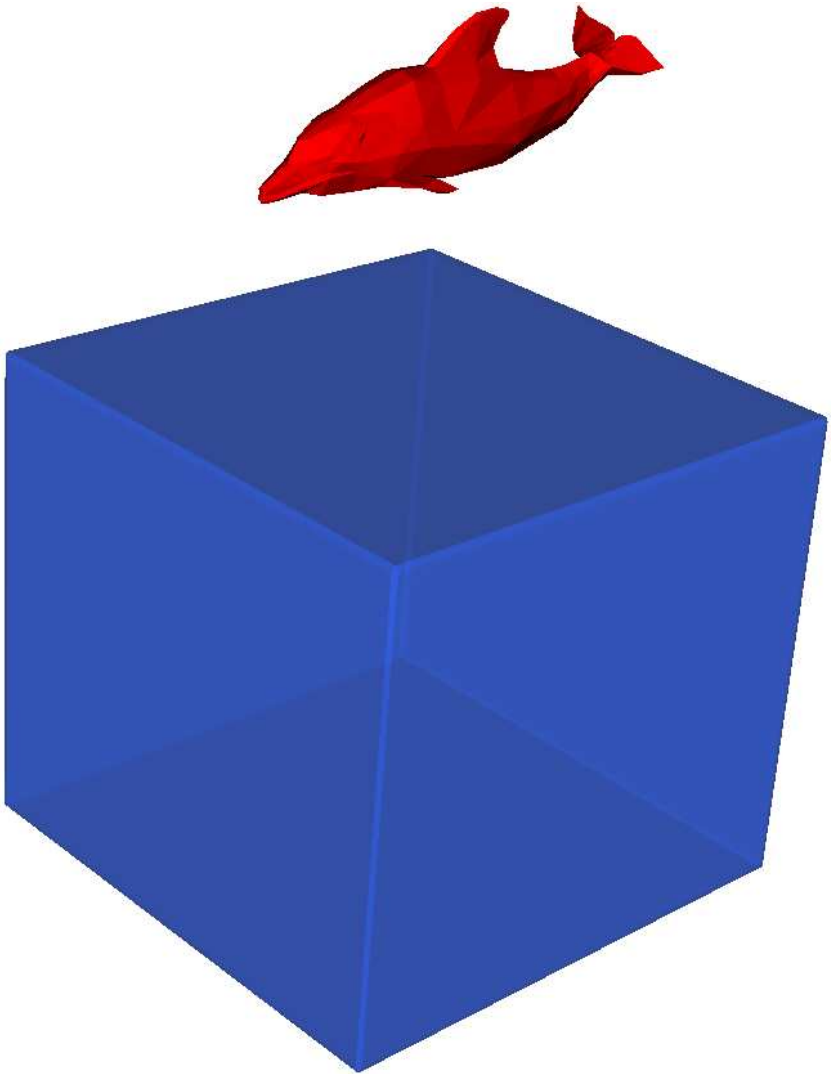
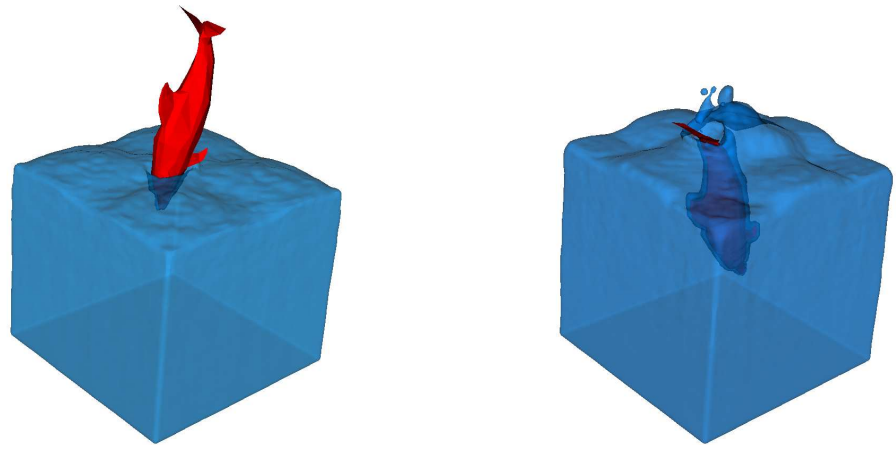


Figure 17: The simulation domain: A DEM dolphin is built from a triangular mesh comprising 469 triangular faces. As explained in the text the LBM domain is fill up to the middle height with the liquid fluids. There are five DEM plates (not visible) enclosing the LBM domain and preventing the dolphin from leaving it.



(a) (b)

(c) (d)

Figure 18: Snapshots at different times of the DEM sphero-dolphin simulation diving into the LBM pool. The video is attached as supplementary material.

measurements, fluid displacement is not an issue since most cells have roughly the same density with slight deviations. However in multiphase and multicomponent simulations the densities can change significantly across the small thickness of the interface separating both fluids. An unfortunate consequence of this is that the liquid fluid may appear inside the sphero-dolphin. Moreover, if the DEM particle moves too fast, Eqs. 12 and 13 will produce negative values for the LBM distribution functions, affecting the stability of the method. It has been verified empirically that one way to ensure these undesirable effects is to ensure that any point across the surface of the DEM particle moves considerably slower than the lattice speed parameter C. Therefore, even under free fall conditions, the simulation cannot run beyond the time when the DEM particle move at a speed comparable to C. Fig. 19 shows how during the dolphin simulation, fluid displacement occurred as expected although there is still a small quantity (negligible compared with the surrounding medium) of fluid permeating into the sphero-dolphin.

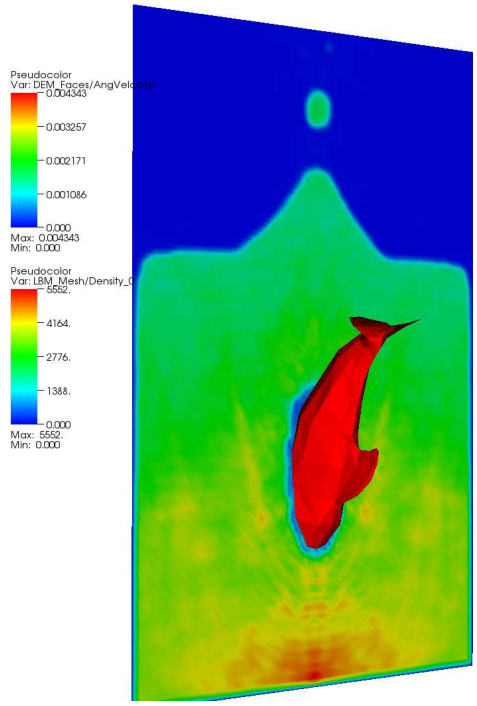
As mentioned before, one of the problems of the Shan-Chen multiphase and multicomponent formalism is that the liquid phase can be highly compressible which makes rigorous validation of the buoyancy forces difficult. Fig.19 shows the distribution of fluid density as the sphero-dolphin is diving. As can be seen, the displacement of the fluid by the sphero-dolphin produces density fluctuations travelling as waves at the lattice speed of sound in front of it. However the density can reach values as high as 5 times the original density which is unrealistic for real fluids like water. This is a problem which is caused by the high compressibility of the liquid phase in the Shan-Chen model which has been reported before [11].

Buoyancy is still present in the method despite the fact that it may not behave as in real water due to the high compressibility. A simulation of a DEM sphero-duck (video attached as supplementary material) is carried out with the same parameters as above but with a domain with half the height. Also de density of the sphero-duck ($600\rho_u$) is lower than the liquid fluid density at equilibrium. The sphero-duck is eventually risen above the liquid level and stays afloat. The video has been attached as supplementary material and some snapshots are shown in Fig. 20.

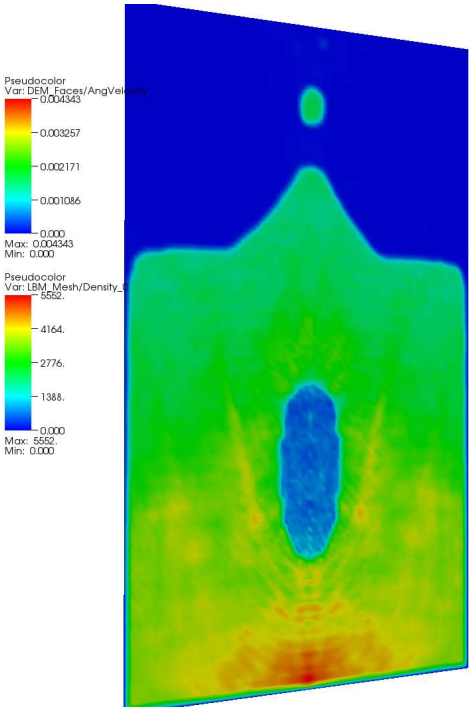
To conclude, The LBM Shan-Chen model combined with the DEM sphero-polyhedra shows a qualitatively agreement with several phenomena found in situations of solid-fluid interaction including: buoyancy, fluid displacement and sonic wave propagation. Two points need to be addressed before any rigorous validation effort can be undertaken with the Shan-Chen model combined with the DEM sphero-polyhedra and are listed as a proposal for future research endeavours:

- A correction to the Shan-Chen model to include the hard ball repulsion of a typical Van der Waals fluid. This will mitigate the high compressibility and increase the speed of sound to approach realistic values for water. It will also permit accurate measurements of the sonic waves produced by the fluid displacement.

- A correction to Eqs. 16 and 17 to account for the pressure given by a more realistic equation of state for the fluid. As it is expressed right now, the force over the particle is proportional to the density of the fluid. This is good enough for the simulations



(a)



(b)

Figure 19: Snapshot of the sphero-dolphin simulation middle cross section showing fluid displacement with (top) and without (bottom) the sphero-dolphin. The colour map is proportional to the density with blue representing 0 density and red the maximum value of $5552\rho_u = 4961\text{kg/m}^3$. As can be seen, the fluid density inside the dolphin is significantly smaller than at the surrounding space due to the fluid displacement.

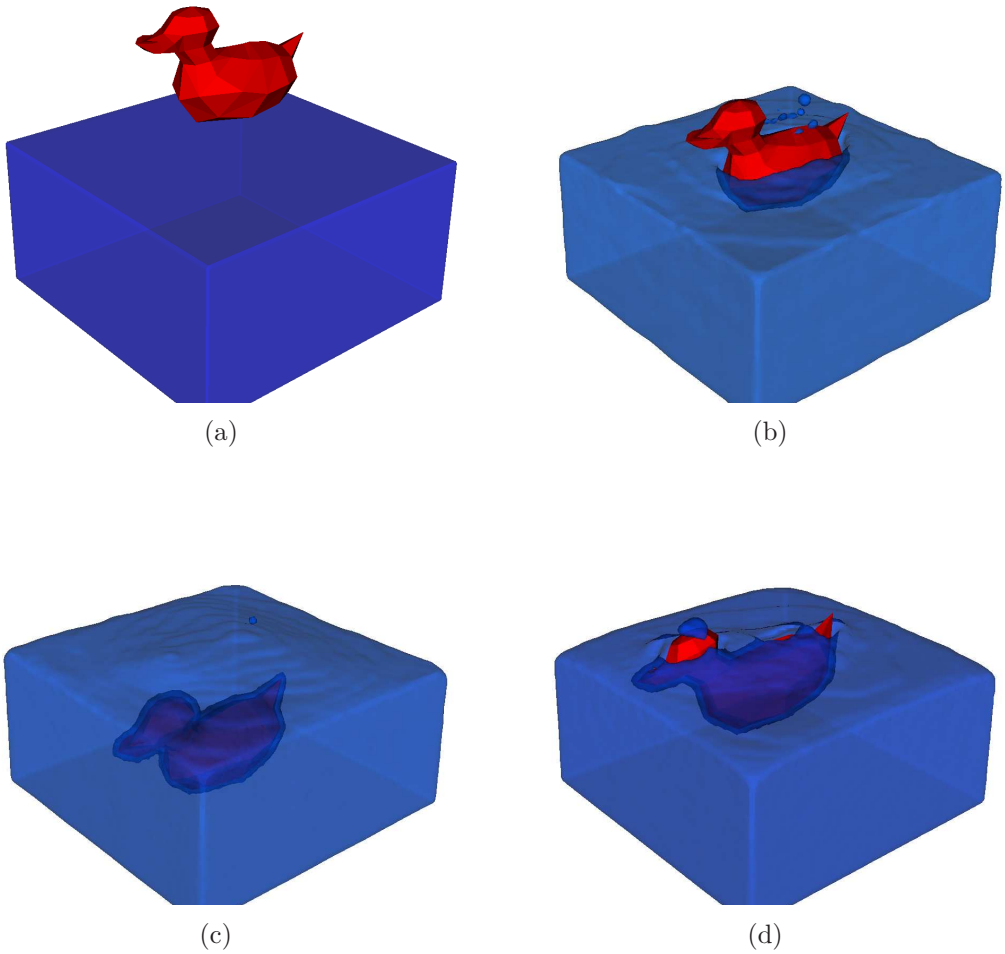


Figure 20: Snapshots at different times of the DEM sphero-duck simulation diving into the LBM pool. The video is attached as supplementary material.

carried out to measure the lift and drag coefficient since it was under incompressible conditions and there were no other phases or fluids. For these conditions the equation of state is the ideal gas law and the density is proportional to the pressure. However for more complex equations of state, the pressure is no longer proportional to the density.

6. Conclusions and projections of this work

An extension for the Owen *et al* [7, 8, 9, 10] coupling law for DEM spheres interacting with LBM fluids is introduced in the present work. The extension generalizes the coupling law for spheres to more complex particles called sphero-polyhedra [13, 33, 17, 16]. The sphero-polyhedra method simplifies the coupling law with the LBM fluid in the same way that it simplifies the collision law with DEM particles.

The method is validated with previous results obtained for spheres by Owen *et al* [9] and for general shaped bodies by Hölzer and Sommerfeld [32] in measurements of drag and lifting coefficients under a moderate Reynolds number regime ($Re = 20 - 100$). It is shown as well that the complexity of the method is not affected significantly with the complexity of the DEM shapes involved, as simulations with spheres, sphero-tetrahedra and sphero-cubes took similar times to finish. This independence of the computational complexity on the shape is achieved by the introduction of a contact list, similar to the Verlet list concept used in DEM simulations [33]. The lists store pairs of DEM features (face, edges or vertices) and LBM cells to check if they overlap and to apply the coupling law. However, the complexity grows in a quadratic manner with the size of the DEM particle since bigger particles will have larger contact lists and more checks per time step.

Finally, a plausibility check is carried out with multiphase and multicomponent LBM fluids using the widely used Shan-Chen [26, 11] method. The model reproduces qualitatively many real phenomena such as fluid displacement, sonic wave propagation and buoyancy. However, in order to undertake rigorous validation, some issues need to be addressed and the author have listed them for future studies. The first one is a modification of the Shan-Chen model to include the hard ball repulsion found in Van der Waals gases to reduce the high compressibility of the liquid phase. The second issue is a modification of the DEM-LBM coupling law (Eqs. 16 and 17) to provide a force over the DEM particle proportional to the fluid pressure and no to its density as it is formulated now.

The results shown in this paper illustrate the potential of the LBM-DEM sphero-polyhedra coupling. Although for this work the drag and lift coefficients were measured in a moderate range of Reynolds numbers, with corrections for turbulent flow as the ones proposed by Feng *et al* [7] larger values for Re can be reached. Therefore, the method could be used in aircraft design.

Another area where the method could be used is in the study of geomechanical processes involving the interaction with fluids. The author has introduced before a cohesion method to model fracture and damage phenomena with the DEM sphero-polyhedra [13]. By combining this method with the LBM, a fully simulation engine for the hydraulic fracturing [34] process could be developed. The hydraulic fracturing is a stimulation treatment used in several industrial sectors such as oil extraction, CO_2 sequestration and geothermal energy plants.

However, this treatment has risen many environmental concerns, and proper modelling tools are needed to learn how to mitigate undesirable collateral effects.

The LBM-DEM coupling scheme can be implemented to model contaminant and radio nuclei transport trough underground water networks, as the ones present in mining sites. It can also be used to simulate the movement of parasites and viruses across the blood vessels of the human body. With flexible DEM particles it can be applied to understand the propulsion methods of fishes, cetaceans and birds. And with a proper equation of state for the LBM fluid the dynamics of man made submerged objects could be studied.

In short, the proposed method could be used in a broad range of disciplines: Aero-space, civil, mining and environmental engineering, biology, medicine and physics. It is expected to have an important impact in future modelling efforts in both science and engineering.

Acknowledgement

The author wants to acknowledge the continuous support and encouragement from Alexander Scheuermann and Ling Li during the development of this numerical tool; to Dorival Pedroso for his advice on how to model animal shapes with triangular meshes; and to Alex Jerves for his useful comments which enriched the paper. This work was funded by The University of Queensland Early Career Research Grants Scheme (Grant No. RM2011002323).

[1] P. A. Cundall, O. D. L. Strack, A discrete numerical model for granular assemblages, Géotechnique 29 (1979) 47–65.

[2] D. Rapaport, The art of molecular dynamics simulation, Cambridge university press, 2004.

[3] N. Belheine, J. Plassiard, F. Donzé, A. Darve, F.and Seridi, Numerical simulation of drained triaxial test using 3D discrete element modeling, Computers and Geotechnics 36 (1-2) (2009) 320–331.

[4] J. Rojek, E. Oñate, C. Labra, H. Kargl, Discrete element modelling of rock cutting, Particle-Based Methods (2011) 247–267.

[5] A. V. Potapov, M. L. Hunt, C. S. Campbell, Liquid solid flows using smoothed particle hydrodynamics and the discrete element method, Powder Technology 116 (23) (2001) 204 – 213, modelling and Computational analysis of a collection of particulate and fluid-solid flow problems.

[6] L. Bierawski, S. Maeno, Dem-fem model of highly saturated soil motion due to seepage force, Journal of waterway, port, coastal, and ocean engineering 132 (5) (2006) 401–409.

[7] Y. Feng, K. Han, D. Owen, Coupled lattice boltzmann method and discrete element modelling of particle transport in turbulent fluid flows: Computational issues, International Journal for Numerical Methods in Engineering 72 (9) (2007) 1111–1134.

[8] Y. Feng, K. Han, D. Owen, Combined three-dimensional lattice boltzmann method and discrete element method for modelling fluid–particle interactions with experimental assessment, International Journal for Numerical Methods in Engineering 81 (2) (2010) 229–245.

[9] D. Owen, C. Leonardi, Y. Feng, An efficient framework for fluid–structure interaction using the lattice boltzmann method and immersed moving boundaries, International Journal for Numerical Methods in Engineering 87 (1-5) (2010) 66–95.

[10] C. Leonardi, D. Owen, Y. Feng, Numerical rheometry of bulk materials using a power law fluid and the lattice boltzmann method, Journal of Non-Newtonian Fluid Mechanics 166 (12) (2011) 628–638.

[11] S. Galindo-Torres, A. Scheuermann, L. Li, D. Pedroso, D. Williams, A lattice boltzmann model for studying transient effects during imbibitiondrainage cycles in unsaturated soils, Computer Physics Communications 184 (4) (2013) 1086 – 1093. doi:10.1016/j.cpc.2012.11.015.

[12] S. A. Galindo-Torres, A. Scheuermann, L. Li, Numerical study on the permeability in a tensorial form for laminar flow in anisotropic porous media, Phys. Rev. E 86 (2012) 046306. doi:10.1103/PhysRevE.86.046306.
[13] S. Galindo-Torres, D. Pedroso, D. Williams, L. Li, Breaking processes in three-dimensional bonded granular materials with general shapes, Computer Physics Communications 183 (2) (2012) 266 – 277. doi:10.1016/j.cpc.2011.10.001.
[14] L. Pournin, T. Liebling, A generalization of distinct element method to tridimentional particles with complex shapes, in: Powders & Grains 2005, Balkema, Leiden, 2005, pp. 1375–1478.
[15] F. Alonso-Marroquin, Spheropolygons: A new method to simulate conservative and dissipative interactions between 2d complex-shaped rigid bodies, Europhysics Letters 83 (1) (2008) 14001.
[16] S. Galindo-Torres, F. Alonso-Marroquín, W. Y. , D. Pedroso, J. Muñoz Castaño, Molecular dynamics simulation of complex particles in three dimensions and the study of friction due to nonconvexity, Physical Review E 79 (6) (2009) 60301.
[17] S. Galindo-Torres, D. Pedroso, Molecular dynamics simulations of complex-shaped particles using Voronoi-based spheropolyhedra., Physical review. E, Statistical, nonlinear, and soft matter physics 81 (6 Pt 1) (2010) 061303.
[18] L. Verlet, Computer experiments on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules, Phys. Rev 159 (1) (1967) 98–103.
[19] Y. Wang, S. Abe, S. Latham, P. Mora, Implementation of Particle-scale Rotation in the 3-D Lattice Solid Model, Pure and Applied Geophysics 163 (9) (2006) 1769–1785.
[20] M. Hecht, J. Harting, Implementation of on-site velocity boundary conditions for d3q19 lattice boltzmann simulations, Journal of Statistical Mechanics: Theory and Experiment 2010 (2010) P01018.
[21] M. Sukop, D. Thorne, Lattice Boltzmann modeling: An introduction for geoscientists and engineers, Springer Verlag, 2006.
[22] Y. Qian, D. d'Humieres, P. Lallemand, Lattice bgk models for navier-stokes equation, EPL (Europhysics Letters) 17 (1992) 479.
[23] X. He, L. Luo, Lattice boltzmann model for the incompressible navier–stokes equation, Journal of Statistical Physics 88 (3) (1997) 927–944.
[24] N. S. Martys, H. Chen, Simulation of multicomponent fluids in complex three-dimensional geometries by the lattice boltzmann method, Phys. Rev. E 53 (1996) 743–750. doi:10.1103/PhysRevE.53.743.
[25] J. Buick, C. Greated, Gravity in a lattice boltzmann model, Physical Review E 61 (5) (2000) 5307.
[26] X. Shan, H. Chen, Lattice boltzmann model for simulating flows with multiple phases and components, Phys. Rev. E 47 (1993) 1815–1819. doi:10.1103/PhysRevE.47.1815.
[27] D. Noble, J. Torczynski, A lattice-boltzmann method for partially saturated computational cells, International Journal of Modern Physics C 9 (08) (1998) 1189–1201.
[28] D. J. Holdych, Lattice boltzmann methods for diffuse and mobile interfaces, ProQuest Dissertations and Theses (2003) 165–165 p.
[29] S. Luding, Cohesive, frictional powders: contact models for tension, Granular Matter 10 (4) (2008) 235–246.
[30] K. Krabbenhoft, J. Huang, M. V. da Silva, A. Lyamin, Granular contact dynamics with particle elasticity, Granular Matter (2012) 1–13.
[31] F. White, Viscous fluid flow, Vol. 66, McGraw-Hill New York, 1991.
[32] A. Hölzer, M. Sommerfeld, Lattice boltzmann simulations to determine drag, lift and torque acting on non-spherical particles, Computers & Fluids 38 (3) (2009) 572–589.
[33] S. Galindo-Torres, J. Muñoz, F. Alonso-Marroquín, Minkowski-Voronoi diagrams as a method to generate random packings of spheropolygons for the simulation of soils, Physical Review E 82 (5) (2010) 56713.
[34] S. Galindo-Torres, J. Muñoz Castaño, Simulation of the hydraulic fracture process in two dimensions using a discrete element method, Physical review. E, Statistical, nonlinear, and soft matter physics 75 (6).