

Particle Collisions in a Lumped Particle Model

Omar al-Khayat^{1,2,*}, Are Magnus Bruaset^{1,2} and
Hans Petter Langtangen^{2,3}

¹ *Computational Geosciences, CBC, Simula Research Laboratory, P.O. Box 134, NO-1325 Lysaker, Norway.*

² *Department of Informatics, University of Oslo, P.O. Box 1080, Blindern, NO-0316 Oslo, Norway.*

³ *Center for Biomedical Computing, Simula Research Laboratory, P.O. Box 134, NO-1325 Lysaker, Norway.*

Received 29 January 2010; Accepted (in revised version) 26 November 2010

Available online 13 June 2011

Abstract. This paper presents an extension of the lumped particle model in [1] to include the effects of particle collisions. The lumped particle model is a flexible framework for the modeling of particle laden flows, that takes into account fundamental features, including advection, diffusion and dispersion of the particles. In this paper, we transform a binary collision model and concepts from kinetic theory into a collision procedure for the lumped particle framework. We apply this new collision procedure to investigate numerically the role of particle collisions in the hindered settling effect. The hindered settling effect is characterized by an increase in the effective drag coefficient C_D that influences each particle in the flow. This coefficient is given by $C_D = (1 - \phi)^{-n} C_D^*$, where ϕ is the volume fraction of particles, C_D^* is the drag coefficient for a single particle, and $n \simeq 4.67$ for creeping flow. We obtain an approximation for C_D / C_D^* by calculating the effective work done by collisions, and comparing that to the work done by the drag force. In our numerical experiments, we observe a minimal value of $n = 3.0$. Moreover, by allowing high energy dissipation, an approximation for the classical value for creeping flow, $n = 4.7$, is reproduced. We also obtain high values for n , up to $n = 6.5$, which is consistent with recent physical experiments on the sedimentation of sand grains.

AMS subject classifications: 76M28, 70F35

Key words: Lumped particle model, particle collision, hindered settling.

*Corresponding author. *Email addresses:* omark@simula.no (O. al-Khayat), arem@simula.no (A. M. Bruaset), hpl@simula.no (H. P. Langtangen)

1 Introduction

Particle collisions are of paramount importance in dense particle laden flows. Such fluid flow appears in many processes in Nature, as well as in many industrial applications. Examples range from chemical synthesis engines to highly turbulent sand-laden subaquatic flow [27]. The complete understanding of the physics in these systems is of great scientific and economical value, and a large body of literature has been dedicated to studying these effects. In many of these physical phenomena, however, the exact effect of particle collisions is still under debate. This is especially the case for highly turbulent sand-laden flows. In a recent comprehensive review, Maiburg and Kneller [19] wrote: "Researchers will have to undertake high-resolution numerical simulations that track large numbers of individual particles to gain insight into the influence of particle-particle interactions". As stated in [19], there is still a need to develop new numerical models for particle collisions in these fluid flows, and the present paper describes the development of such a model.

There are numerous ways of modeling the effects of particle collisions. The most commonly used approach is the discrete particle methods. Here, each particle's position and velocity are obtained by the application of Newton's second law of motion [24]. This equation is coupled with variants of the Navier-Stokes equation to obtain a description of the dense flow [20]. Discrete particle methods, however, are computationally expensive.

An alternative way is to model the dense particle laden flow as a continuum [5], using a multiphase computational fluid dynamics (CFD) approach. In these approaches, one of the big challenges lie in correctly obtaining the constitutive relations for the modelling of two-phase flows [26]. Concepts from kinetic theory [11] has been used to obtain constitutive relations for certain particle laden flows [15], but has of yet not been applied to highly fluidized beds, or highly turbulent sand laden flows [19]. Continuum models also have the conceptual difficulty of simulating discrete granular effects, such as particle segregation [6].

It is beyond the scope of this paper to give a full survey of the available approaches for the modelling of particle collisions. Readers interested in discrete particle modelling are referred to [6]. For a detailed review of multiphase CFD, see [4,26]. The goal of this paper is to extend a recently developed numerical framework, the lumped particle model originally presented in [1], with a new hybrid continuum-particle model for particle collisions.

The lumped particle model is a flexible and numerically efficient framework for the modelling of particle transport in fluid flow, that takes into account fundamental features of particle flow, including advection, diffusion and dispersion of the particles. This framework reproduces particle flow properties inherent in both continuum and discrete approaches, and correctly reproduces advection and diffusion phenomena as special cases [1]. There are, however, some particle flow features not included in the framework. Currently, the lumped particle model is applicable to dilute particle laden flows only. In this paper, we want to expand the framework to dense particle flows. When the local volume fraction of particles increases, it is no longer reasonable to ignore inter-particle

collisions. Hence, we present an expansion of the lumped particle model to include the effects of particle collisions.

The goal of this paper is twofold. First, we want to extend the lumped particle framework such that dense particle flows can be modelled. This extension implies the introduction of a collision procedure. Second, by applying the newly developed procedure, we want to investigate the role of particle collisions on the hindered settling effect. In this setting, the hindered settling effect refers to the fact that particle sedimentation is interfered by the presence of other particles within the flow [21].

The structure of this paper is as follows: Section 2 gives a brief overview of the lumped particle modelling framework. In Section 3, we review some aspects from kinetic theory as applied to particle collisions. Moreover, we describe how these concepts are to be applied to the lumped particle framework. This enables us to design a collision procedure in Section 3.1. We continue in Section 3.2 to recount a specific binary collision model, which includes the effects of rotational energy dissipation. In Section 4, we briefly discuss the hindered settling effect, and show how to apply the newly developed collision procedure to this setting. Finally, in Section 4.1, we present a set of numerical experiments on hindered settling.

2 Overview of the lumped particle model

We will now give a short account of the lumped particle modelling framework. The framework is based on a mesoscopic hybrid continuum-particle approach, where groups of particles constitute a particle lump. Instead of tracking the individual dynamics of each particle, a weighted spatial averaging procedure is used to evolve the particles in the computational domain. The external forces are applied to the lump of particles, from which an average position and velocity is derived. Hence, the particles are in a sense considered as a continuum, but where the particle nature heavily influence the dynamics. In the following description, we will restrict our attention to a two-dimensional regular lattice.

When computing the evolution of the particles, the particle lumps are partitioned into smaller entities, known as quasi-particles, which are then transported according to local physical effects. These smaller entities recombine into new particle lumps at the target destinations. We partition the computational domain into a regular lattice, with physical spacing parameters Δx and Δy . Time is discretized into increments of Δt . As shown in Fig. 1, the lattice defines a set of grid cells, each having a center point \mathbf{x}_p . The distribution $N(\mathbf{x}_p, t)$, and the velocity $\mathbf{V}(\mathbf{x}_p, t)$ are defined as the number of particles inside the cell and the average velocity of these particles, respectively. The computation of the temporal evolution of these variables consists of three distinct steps which replace the conventional full particle-tracking approach.

The first step is a dispersion step, where the particle lump is split into smaller parts, the quasi-particles. Each of these quasi-particles have a dispersion velocity \mathbf{c}_m , which

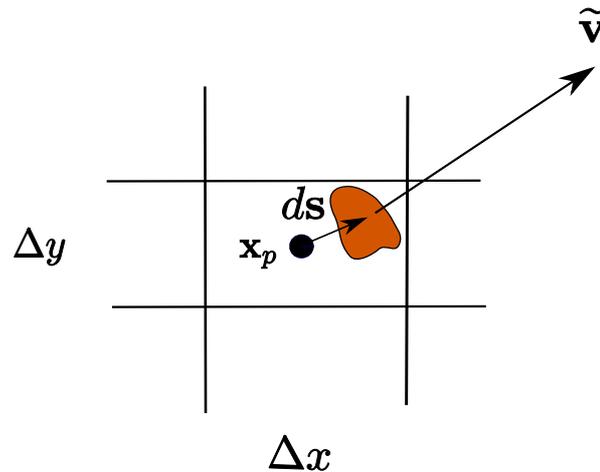


Figure 1: The lumped representation of particles. Particles within a grid cell are treated as a single entity. Here, $\tilde{\mathbf{v}}$ is the average velocity and ds is the offset from the particle centroid to the cell center.

quantifies the direction of the quasi-particle movement. It is in this step that the kinematics of these quasi-particles is calculated. The second step is the recombination step, where quasi-particles are recombined to form a new particle lump at the destination sites. In preparation for the next time step, the dispersion velocities are calculated at this stage as well. The third step is a diffusion step, which enables the modeling of Brownian motion and similar phenomena. In this paper, however, we will not discuss the diffusion step further. Interested readers are referred to [1].

In the dispersion step, we compute the force acting on the particle lump in each grid cell. Using the average velocity as a basis, an acceleration is calculated numerically from Newton's second law applied to the particle lump. In this context, Newton's second law can be written as

$$\frac{d\mathbf{v}_p}{dt} = -\frac{1}{\tau_p}(\mathbf{v}_p - \mathbf{u}) + \left(1 - \frac{\rho_f}{\rho_p}\right)\mathbf{g}. \tag{2.1}$$

By solving the above equation, the particle position \mathbf{x} can be found from

$$\frac{d\mathbf{x}}{dt} = \mathbf{v}_p. \tag{2.2}$$

Here, we have defined

$$\tau_p = \frac{d_p^2 \rho_p}{18\mu}, \tag{2.3}$$

where the particle relaxation time τ_p is a measure of the particle's response to a changing fluid velocity. In Eq. (2.1), \mathbf{v}_p and \mathbf{u} are the particle and fluid velocities respectively, while d_p is the particle diameter, and \mathbf{g} is the gravitational acceleration. Furthermore, ρ_f and ρ_p are the fluid and particle densities respectively, and μ is the dynamic fluid viscosity.

The first term on the right hand side is the drag force on the particle, and the second term represents the combined effect of gravity and buoyancy. In our approximation, the relaxation time can be written as

$$\tau_p = \frac{8m_p}{\pi d_p \mu_f} (C_D \text{Re}_p)^{-1}, \tag{2.4}$$

where C_D is called the drag coefficient. Moreover, Re_p is the particle Reynolds number defined as

$$\text{Re}_p = \frac{\rho_f |\mathbf{v}_p - \mathbf{u}| d_p}{\mu}. \tag{2.5}$$

For low particle Reynolds number, that is $\text{Re}_p < 1000$, the drag coefficient can be approximated by

$$C_D \simeq \frac{24}{\text{Re}_p} (1 + 0.15 \text{Re}_p^{0.687}). \tag{2.6}$$

The drag coefficient we use at this stage is that of a single particle. That is, we assume that the particles are unperturbed by each other's presence. Using the above equations, the drag force can be written as

$$\mathbf{F}_d = -\frac{\pi d_p^2 \rho_f}{8} C_D (\mathbf{v}_p - \mathbf{u}) |\mathbf{v}_p - \mathbf{u}|. \tag{2.7}$$

Observe that we have neglected some physical effects like the virtual mass force [24] and rotational forces. When adding these effects, the above equation is better known as the Bassinet-Boussinesq-Oseen (BBO) equation.

A discrete solution of Eq. (2.1) is obtained by an explicit forward Euler method in time. Let $t^\ell = \ell \Delta t$ with ℓ being the time step index. A temporal discretization of Eq. (2.1) becomes

$$\frac{\Delta \mathbf{v}^\ell}{\Delta t} = -\frac{1}{\tau_p} (\mathbf{V}^{\ell-1} - \mathbf{u}^{\ell-1}) + \left(1 - \frac{\rho_f}{\rho_p}\right) \mathbf{g}, \tag{2.8}$$

where $\Delta \mathbf{v}^\ell / \Delta t$ is the acceleration of each individual particle in the current grid cell. We will apply this acceleration to the quasi-particles. The average velocity $\mathbf{V}^{\ell-1}$ is used as the basis for the force calculation, modeling the overall drift of the particle lump. Whereas the fluid velocity $\mathbf{u}^{\ell-1}$ is assumed to be known, either analytically or as a numerical approximation generated by a separate solver for fluid flow.

The displacement $\Delta \mathbf{x}^\ell$ of the quasi-particles will depend on the dispersion velocities. The above acceleration is then applied to the quasi-particles within a grid cell, thereby changing their dispersion velocities \mathbf{c}_i . Since the k' th quasi-particle attains the new velocity $\mathbf{c}_k + \Delta \mathbf{v}^\ell$, we can numerically integrate Eq. (2.2) to obtain the displacement,

$$\Delta \mathbf{x}_k^\ell = \int_{t^{\ell-1}}^{t^\ell} \mathbf{v}_p dt = \left(\mathbf{c}_k + \frac{\Delta \mathbf{v}^\ell}{2} \right) \Delta t. \tag{2.9}$$

As a result, the quasi-particles are transported to their target cells, corresponding to traveling with their respective velocities in the given time increment Δt .

The distance a quasi-particle travels does not usually correspond to an integer multiple of one grid cell. As a means to track this error, an error correction vector $d\mathbf{s}(\mathbf{x}_p)$ is introduced, which quantifies the offset of the centroid of the particle lump relative to \mathbf{x}_p , and can also serve as a measure of positional error. A quasi-particle displacement algorithm is used to calculate the target grid cell and the new error correction vector $d\mathbf{s}^+$. Let I_j be the initial grid cell and I_k be the target grid cell, where \mathbf{k} and \mathbf{j} are grid cell index vectors. If ds_i and k_i denotes the i 'th component of $d\mathbf{s}$ and the grid cell index vector respectively, then these three are updated at each time step by

$$d\tilde{s}_i = s_i - \left[\frac{s_i}{\Delta x_i} \right] \Delta x_i, \quad (2.10a)$$

$$k_i = j_i + \left[\frac{s_i}{\Delta x_i} \right] + \left[\frac{ds_i + d\tilde{s}_i}{\Delta x_i} \right], \quad (2.10b)$$

$$ds_i^+ = ds_i + d\tilde{s}_i - \left[\frac{ds_i + d\tilde{s}_i}{\Delta x_i} \right] \Delta x_i, \quad (2.10c)$$

where s_i is i -th component of the displacement vector calculated by Eq. (2.9). Here, $[\cdot]$ denotes the closest integer value. The full derivation of this algorithm is given in [1].

In the recombination step, the quasi-particles entering the grid cells are recombined into a new lumped particle. A new mean velocity \mathbf{V} and error measure $d\mathbf{s}$ are calculated as the averages of the respective velocities and error measures of the quasi-particles. Consider a grid cell where several quasi-particles have entered, where the i 'th quasi-particle consists of N^i particles and has a velocity \mathbf{v}_i . Assuming that there are m quasi-particles entering the grid cell, the magnitude of the average velocity for the next time step becomes

$$\bar{v}^+ = \sqrt{\frac{N^1 \mathbf{v}_1^2 + N^2 \mathbf{v}_2^2 + \dots + N^m \mathbf{v}_m^2}{\sum N^i}}, \quad (2.11)$$

which is the root mean square (rms) of the quasi-particles' velocities. To find the direction, we first calculate the weighted average velocity vector

$$\hat{\mathbf{v}} = \frac{N^1 \mathbf{v}_1 + N^2 \mathbf{v}_2 + \dots + N^m \mathbf{v}_m}{\sum N^i}, \quad (2.12)$$

which implies the new velocity

$$\mathbf{V} = \bar{v}^+ \frac{\hat{\mathbf{v}}}{|\hat{\mathbf{v}}|}. \quad (2.13)$$

This averaging approach has the distinct advantage of conserving the total kinetic energy of the particles. Moreover, the error correction $d\mathbf{s}$ is calculated as the weighted average of the incoming quasi-particles' offsets

$$d\mathbf{s} = \frac{N^1 d\mathbf{s}_1 + N^2 d\mathbf{s}_2 + \dots + N^m d\mathbf{s}_m}{\sum N^i}. \quad (2.14)$$

Note that each ds_i is obtained from Eqs. (2.10a)-(2.10c). Moreover, the number of particles constituting the new lump is $N(\mathbf{x}_p, t + \Delta t) = \sum N^i$.

To compensate for the possibly varying velocities of the quasi-particles, a set of dispersion velocities \mathbf{c}_i are defined. These are derived from a momentum balance calculation, where each direction is weighted according to its fraction of the total momentum within the grid cell. That is, in preparation for the next time step, these weights determine the number of particles each quasi-particle will consist of. A standard deviation $\sigma = (\sigma_x, \sigma_y)$ from the mean value $\hat{\mathbf{v}} = (\hat{v}_x, \hat{v}_y)$ is defined to quantify the general spread of particle velocities within the grid cell. The standard deviation is then

$$\sigma_x = \sqrt{\frac{N^1(\hat{v}_x - v_x^1)^2 + N^2(\hat{v}_x - v_x^2)^2 + \dots + N^m(\hat{v}_x - v_x^m)^2}{\sum N^i}}, \tag{2.15a}$$

$$\sigma_y = \sqrt{\frac{N^1(\hat{v}_y - v_y^1)^2 + N^2(\hat{v}_y - v_y^2)^2 + \dots + N^m(\hat{v}_y - v_y^m)^2}{\sum N^i}}. \tag{2.15b}$$

The directions of dispersion are given by

$$\tilde{\mathbf{c}}_0 = \hat{\mathbf{v}}, \quad \tilde{\mathbf{c}}_1 = \hat{\mathbf{v}} + \sigma, \quad \tilde{\mathbf{c}}_2 = \hat{\mathbf{v}} - \sigma. \tag{2.16}$$

Furthermore, we define a fractional number distribution q_i , which quantifies the fraction of particles at each grid cell that are transported in direction \mathbf{c}_i . So, q_i is given by

$$q_i = \frac{\tilde{c}_i^2}{\sum_j \tilde{c}_j^2}. \tag{2.17}$$

Furthermore, the magnitude of the dispersion directions must be adjusted to conserve the total kinetic energy within the grid cell. This property is guaranteed by setting the magnitude equal to the rms velocity \mathbf{V} , which defines the dispersion vectors

$$\mathbf{c}_0 = v^+ \frac{\hat{\mathbf{v}}}{|\hat{\mathbf{v}}|}, \quad \mathbf{c}_1 = v^+ \frac{\hat{\mathbf{v}} + \sigma}{|\hat{\mathbf{v}} + \sigma|}, \quad \mathbf{c}_2 = v^+ \frac{\hat{\mathbf{v}} - \sigma}{|\hat{\mathbf{v}} - \sigma|}. \tag{2.18}$$

Further details about the lumped particle modelling framework can be found in [1].

3 Modeling particle collisions with the lumped particle framework

Particle collisions serve to redistribute the mechanical energy of the individuals involved. This usually changes the direction and magnitude of particle velocities. For *conservative* collisions, the total kinetic energy is conserved. For *dissipative* collisions, however, some of the kinetic energy is lost through frictional interaction between the particles. We will only consider the effects of two-particle collisions.

In this section, we will first show in general how the lumped particle framework can be extended to include the effects of particle collisions. Once the formalism is well

established, we will then give an explicit example of a specific particle model, which will be used for the remainder of this paper. We will also assume that the particle angular velocity is negligible, meaning that any rotational buildup will dissipate faster than the collision time scale. Furthermore, only two-particle collisions will be considered.

The modeling of collision within the lumped particle framework will be accomplished by a re-calculation of the dispersion velocities. Recall that in the dispersion phase, the particle lump is partitioned into three quasi-particles, each with a dispersion velocity. These dispersion velocities give a measure of the distribution of velocities of the particles within a grid cell, and are consequently necessary variables for Newton's laws of motion. It is therefore natural to add a collision procedure before Eqs. (2.1)-(2.2) are applied on the particles. The collision procedure will consist of an algorithm for updating the dispersion velocities, which will be based on a number of factors, like the average number of collisions occurring within a grid cell.

The collision procedure is as follows; Consider a particle lump in a grid cell consisting of N real particles. We first calculate the number of colliding particles N_C , which we will refer to as the collision number. Recall that in the dispersion phase, the lumped particle is partitioned into three quasi-particles, each with their own dispersion velocity. An amount of N_C real particles are proportionally removed from these quasi-particles, which are used to create two new ones. These two colliding quasi-particles, which each consist of $N_C/2$ real particles, will be given dispersion velocities according to collisional effects. To this end, we need to calculate the average impact velocity \mathbf{v}_Δ of the N_C colliding particles. Here, the impact velocity is the relative velocity between the colliding particles, which can be obtained by a statistical mechanics approach. Once \mathbf{v}_Δ is calculated, a specific collision model can be applied to find the colliding particles' new velocities, which will be the dispersion velocities of the new colliding quasi-particles. The three original quasi-particles, which now consist of fewer particles, will continue unperturbed by the collisions and retain their dispersion velocities. Hence, we are left with five quasi-particles, two of which encode collisional effects.

It is important to note that with the term "collision model", we mean a specific rule that relates the impact velocity of the particles with the new perturbed velocities of the particles. In essence, a collision model will describe the momentum transfer and dissipation of the interacting particles. This is equivalent to specifying the constitutive law for the system. Furthermore, particle collision number can also be affected by the amount of turbulence in the flow. Although we are not considering turbulence in this paper, the formalism developed here is compatible with modeling effects such as these in future developments.

3.1 The collision procedure

We will now give a detailed account on the collision procedure, which entails the calculation of the collision number N_C and the average impact velocity \mathbf{v}_Δ . This will consist of studying the particle lump using a statistical mechanics approach based on the book [3]

and results in [10]. Moreover, we describe how the new quasi-particles are constructed, and give a complete algorithm for the re-calculation of the dispersion velocities.

Consider a set of N particles suspended in a fluid or gas within a grid cell. In statistical mechanics, the particle velocity \mathbf{v}_p can be expressed as a sum of the local ensemble average velocity \mathbf{V} , and a random fluctuating velocity $\mathbf{C}=(C_x, C_y, C_z)$. This can be written as [10]

$$\mathbf{C} = \mathbf{v}_p - \mathbf{V}. \tag{3.1}$$

Moreover, it is usual to define a velocity distribution function $f(\mathbf{v}_p, \mathbf{x}, t)$, which is the probability of finding a particle at position \mathbf{x} within the volume $d\mathbf{x}$ with the velocity \mathbf{v}_p at time t . A well known approximation for the distribution function is the Maxwellian distribution, which is based on the assumption that the particle velocity is isotropically distributed around a mean velocity. In our setting, the velocity distribution is given by

$$f(C_x, C_y, t) = 4\pi N' \frac{C_x^2 + C_y^2 + C_z^2}{(2\pi)^{3/2} \theta_x^{1/2} \theta_y^{1/2} \theta_z^{1/2}} \exp \left[-\frac{C_x^2}{2\theta_x} - \frac{C_y^2}{2\theta_y} - \frac{C_z^2}{2\theta_z} \right], \tag{3.2}$$

where θ_i is the granular temperature defined as

$$\theta_x = \langle C_x, C_x \rangle, \quad \theta_y = \langle C_y, C_y \rangle, \quad \theta_z = \langle C_z, C_z \rangle. \tag{3.3}$$

Here, N' is the number density defined as the number of particles per unit volume. It is also convenient to define the directionally averaged granular temperature $\theta \equiv D^{-1} \sum_i \theta_i$, where D is the number of dimensions of the system.

To study collisional interactions, we can define the pair distribution function $f_{12}(\mathbf{v}_{p,1}, \mathbf{x}_1, \mathbf{v}_{p,2}, \mathbf{x}_2, t)$, which is the probability of finding a pair of particles with velocities $\mathbf{v}_{p,1}$ and $\mathbf{v}_{p,2}$ at the position \mathbf{x}_1 and \mathbf{x}_2 within the volumes $d\mathbf{x}_1$ and $d\mathbf{x}_2$ at time t . By assuming that the velocities of the particles are statistically uncorrelated, it is shown in [3] that the pair distribution f_{12} can be approximated by

$$f_{12}(\mathbf{v}_{p,1}, \mathbf{x}_1, \mathbf{v}_{p,2}, \mathbf{x}_2, t) = g_0(\mathbf{x}_1 - \mathbf{x}_2) f(\mathbf{v}_{p,1}, \mathbf{x}_1, t) f(\mathbf{v}_{p,2}, \mathbf{x}_2, t). \tag{3.4}$$

Here, $g_0(\mathbf{x}_1 - \mathbf{x}_2)$ is the radial distribution function, which quantifies the effect of the volume occupied by the two particles on their movement. We will assume that the radial function depends only on the volume fraction of particles. Physically, this means that the degree of anisotropy in particle collisions is small [10]. The radial function can then be written as a function of the volume fraction of a grid cell $\phi = NV_p / (\Delta x \Delta y)$. That is, $g_0(\mathbf{x}_1 - \mathbf{x}_2) \equiv g_0(\phi)$. Here, V_p is the particle volume. Many different expressions for g_0 exist in the literature. The most commonly used expression from [23] is given by

$$g_0(\phi) = \left[1 - \left(\frac{\phi}{\Phi} \right)^{1/3} \right]^{-1}, \tag{3.5}$$

where Φ is the maximum volume fraction of the particles, which in general is dependent on particle shape. Another similar expression is

$$g_2(\phi) = \frac{3}{5} \left[1 - \left(\frac{\phi}{\Phi} \right)^{1/3} \right]^{-1}, \tag{3.6}$$

which was first proposed in [7]. We will mention one more radial function given by [16], which according to [10] gives the best fit for experiments on elastic spheres [2]. This is given by

$$g_3(\phi) = 1 + 4\phi \frac{1 + 2.5\phi + 4.5904\phi^2 + 4.515439\phi^3}{[1 - (\phi/\Phi)^3]^{0.67802}}. \quad (3.7)$$

In Section 4, we will investigate what differences these three radial functions will have on the collisions. We will, unless we state otherwise, use Eq. (3.5) in the numerical experiments. Furthermore, we assume that the particles are spherical in shape. Experiments on random packing [12] have shown a maximum volume fraction of $\Phi \sim 0.64$. Note that $g_0(\phi)$ approaches 1 for dilute flows.

By calculating the first moment of the pair distribution f_{12} , we can find an estimate for the number of colliding particles per unit volume and unit time \bar{N}_C [3]. This calculation gives

$$\bar{N}_C = 4N'^2 d_p^2 g_0(\phi) \sqrt{\pi\theta}. \quad (3.8)$$

We will now apply the above results to the lumped framework. Restricting our attention to the two dimensional case, the particle lump is partitioned into three quasi-particles, each consisting of q_0N , q_1N , and q_2N particles, respectively. We remove q_0N_C , q_1N_C , and q_2N_C from each of the respective quasi-particles; these will be the colliding particles. Two new quasi-particles are created, consisting of $N_C/2$ particles each. These quasi-particles are given new post-collision dispersion velocities, \mathbf{c}_6 and \mathbf{c}_7 . Since we are studying the averaged collision effect, the pre-collision velocities will be set equal to the average dispersion velocities of the particle lump \mathbf{c}_1 and \mathbf{c}_2 . This implies that the average impact velocity is given by

$$\mathbf{v}_\Delta = \mathbf{c}_2 - \mathbf{c}_1. \quad (3.9)$$

By the principle of momentum conservation, the velocities after the collision gives

$$\mathbf{c}_6 = \mathbf{c}_1 + \frac{\Delta J(\mathbf{v}_\Delta)}{m_p}, \quad \mathbf{c}_7 = \mathbf{c}_2 - \frac{\Delta J(\mathbf{v}_\Delta)}{m_p}, \quad (3.10)$$

where $J(\mathbf{v}_\Delta)$ is the impulse between the particles. In general, the impulse is a function of the impact velocity, and must be specified by a collision model. As mentioned earlier, we will give an example of such a model in Section 3.2. In summary, we are left with three quasi-particles with dispersion velocities given by Eqs. (2.18), and the two colliding ones with dispersion velocities \mathbf{c}_6 and \mathbf{c}_7 . This also implies a recalculation of the velocity of the particle lump \mathbf{V} . Observe that when the particle collision number N_C becomes high, some or all of the non-colliding quasi-particles can be left with no particles.

What remains to be quantified is the granular temperature θ for the lumped particle in each grid cell. Observe that the definition of the standard deviation σ from Eq. (2.15a) can be viewed as a discrete approximation to $\sqrt{\theta}$. Hence, we can assume the following

relations

$$\theta = \frac{1}{2}(\sigma_x^2 + \sigma_y^2), \quad \theta_x = \sigma_x^2, \quad \theta_y = \sigma_y^2. \quad (3.11)$$

Given the above equations, we obtain

$$\bar{N}_C = 2N^{/2}d_p^2g_0(\phi)\sqrt{2\pi(\sigma_x^2 + \sigma_y^2)}. \quad (3.12)$$

From the above results, we can calculate the collision number N_C by

$$N_C = \bar{N}_C \Delta x \Delta y \Delta t, \quad (3.13)$$

which is the number of collisions that occur in a grid cell during one time step.

3.2 A dissipative binary collision model

We will now discuss a binary collision model from the literature, and illustrate how this model can be applied in the lumped particle framework. The collision model shown here is based on the work of [10], and is used frequently in studying granular flows [9, 14, 28]. Although there exist many variations of this model, we will use a variant that has the advantage of being able to model the effects of rotational energy dissipation. It should be noted, however, that only binary collisions is considered in this model.

Three parameters are integral in describing the collisions; The coefficient of normal restitution e , the coefficient of tangential restitution β , and the Coulomb friction coefficient μ_C . The coefficient of normal restitution e represents the ratio between the relative normal velocities of the colliding particles before and after the collision. Here, the relative normal velocity is the component of the impact velocity along the radial distance between the colliding particles. A value of $e = 1$ is equivalent to a perfect elastic impact, while $e = 0$ represents a collision of perfectly plastic particles [14]. The latter case implies a significant loss in kinetic energy. Moreover, the tangential restitution coefficient β represents the loss of kinetic energy in the tangential direction of impact. Finally, the coefficient μ_C models the sliding friction between the particles. We have that $0 \leq e \leq 1$, $0 \leq \beta \leq 1$, and $\mu_C \geq 0$.

In this model it is usual to distinguish between two distinct types of collisions. A sliding collision, and a sticking collision. The collision is of the sliding type if the angle of impact γ is greater than the critical angle

$$\gamma_0 = \arctan \left[\frac{7\mu_C(1+e)}{2(1+\beta)} \right]. \quad (3.14)$$

Otherwise, we have a collision of the sticking type. In the sliding collision case, the impulse can be written as

$$\Delta J = -\frac{m_p}{2}(1+e)\mathbf{v}_\Delta^n - \frac{m_p}{2}\mu_C(1+e)\cot\gamma|\mathbf{v}_\Delta^n|\frac{\mathbf{v}_\Delta^t}{|\mathbf{v}_\Delta^t|}, \quad (3.15)$$

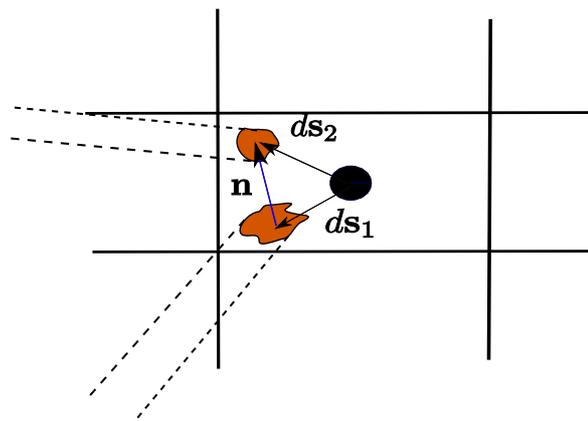


Figure 2: Estimation of the normal impact vector for two quasi-particles.

where \mathbf{v}_{Δ}^n and \mathbf{v}_{Δ}^t is the normal and tangential component of \mathbf{v}_{Δ} relative to the direction of impact, respectively. The direction of impact is defined as the unit vector between the particle positions \mathbf{r}_1 and \mathbf{r}_2

$$\mathbf{n} = \frac{\mathbf{r}_1 - \mathbf{r}_2}{|\mathbf{r}_1 - \mathbf{r}_2|}. \quad (3.16)$$

This implies that [28],

$$\mathbf{v}_{\Delta}^n = (\mathbf{v}_{\Delta} \cdot \mathbf{n}) \mathbf{n}, \quad (3.17a)$$

$$\mathbf{v}_{\Delta}^t = (\mathbf{v}_{\Delta} \cdot \mathbf{n}) \frac{\mathbf{v}_{\Delta} - (\mathbf{v}_{\Delta} \cdot \mathbf{n}) \mathbf{n}}{|\mathbf{v}_{\Delta} - (\mathbf{v}_{\Delta} \cdot \mathbf{n}) \mathbf{n}|}. \quad (3.17b)$$

Note that since the Coulomb friction force is proportional to the normal impact velocity component, $(\mathbf{v}_{\Delta} \cdot \mathbf{n})$ has been used as the magnitude of the tangential component \mathbf{v}_{Δ}^t . For the sticking collision, the impulse is given by

$$\Delta J = -\frac{m_p}{2}(1+e)\mathbf{v}_{\Delta}^n - \frac{m_p}{7}(1+\beta)\mathbf{v}_{\Delta}^t. \quad (3.18)$$

To apply this collision model to our framework, we need to find an estimate for the average normal vector $\bar{\mathbf{n}}$ for the lumped particle. This calculation will be done in the recombination step during each time increment. The weighted average impact normal $\bar{\mathbf{n}}$ is given by

$$\bar{\mathbf{n}} = \frac{N^1 \mathbf{ds}_1 - N^2 \mathbf{ds}_2 - N^3 \mathbf{ds}_3 - \dots - N^m \mathbf{ds}_m}{\sum N^i}, \quad (3.19)$$

where N^i is the number of real particles constituting quasi-particle i . Furthermore, m is the number of quasi-particles entering the grid cell. Fig. 2 shows an illustration of this calculation for two quasi-particles. In Section 4, we will investigate what consequences this collision model has on freely evolving particle flow.

4 Hindered settling experiments

In the previous section we discussed a specific three parameter collision model within the lumped particle framework. We will now consider a physical setting where collisions are highly relevant. This case will involve a highly dense distribution of particles within a fluid, which is often referred to as a fluidized bed [13]. Many physical forces affect these particles, such as gravity, bubble lifting, fluid drag force, and in some cases electrostatic forces [17]. To be able to pinpoint the exact effect of the collisions, however, we will focus on fluid-particle interaction alone. Therefore, buoyancy and the other physical effects will be omitted. The analysis and results presented here can be viewed as a proof of concept for the simulation of particle collisions within the lumped particle approach.

Recall that in Eq. (2.1), we use the single particle description for the particle lump. Hence, we want to investigate what effect collisions have on the evolution of the particles in the flow. Particularly, we will calculate the effective increase of the drag force. In this setting, it is usual to compare the effective drag coefficient C_D on a single particle in the lump compared to the drag coefficient C_D^* of an isolated particle. One common parameterization is given by [13]

$$C_D = C_D^* f(\epsilon)^m. \quad (4.1)$$

The above equation is known as a drag law, with $f(\epsilon)$ being the voidage function and m is an integer. Here, ϵ is the voidage fraction within a control volume. That is, we have that $\epsilon = 1 - \phi$. In Eq. (4.1), m lies between 1 and 2, depending on the particle Reynolds number Re_p . For low values of Re_p , $m = 1$ [14]. The voidage function $f(\epsilon)$ is often approximated by a power law correlation ϵ^{-n} [8,21]. This gives

$$C_D = C_D^* \epsilon^{-n}, \quad (4.2)$$

where n is the drag correlation index. A value of $n = 4.67$ reproduces experimental data for flow of spherical particles with low particle Reynolds number [18,29]. For particles in turbulent flows, n is usually between 2 and 4. The drag law described by Eq. (4.2) has been successfully used in many numerical approaches for particle flows. It should be noted, however, that the experimental values obtained for n results from the averaging of many different physical effects, including many-particle collisional events. It is therefore unclear how the different physical forces contribute to the drag law. Moreover, the value of n may also depend on the particles studied. For instance, it has been recently shown in [25] that the drag index n can become as large as 6.5 for the sedimentation of sand grains.

In this paper, we will investigate if the lumped particle framework can reproduce the drag law described above. Hence, we need to calculate the effect of collisions on the drag law as described by the ratio C_D/C_D^* . Since the configuration of the particles' position and velocities is complex, a calculation of an effective drag force is faces considerable challenges. Instead, we will estimate the drag law by calculating the total work done

by the drag force, and compare that to an effective drag force which takes the particle collisions into account.

Recall that during each time step, the particle lump is partitioned into three quasi-particles, each of which are influenced by Eq. (2.1). If each quasi-particle consists of M_i real particles, then the work done by the fluid on each quasi-particle is given by

$$W_i = -m_p M_i \frac{1}{\tau_p} (\mathbf{v}_p - \mathbf{u}) \cdot \mathbf{s}_i, \quad (4.3)$$

where \mathbf{s}_i is the displacement resulting from the drag force. These particles are then moved to their target grid cells where a collision event occurs. Consider one target grid cell. If no collisions take place, the total work done, W , is given by $\mathbf{F}_d \cdot \mathbf{s}_i$. Using Eq. (2.7), we obtain

$$W = -\frac{\pi d_p^2 \rho_f}{8} C_D^* M_i \sum_i (\mathbf{v}_i - \mathbf{u}) \cdot \mathbf{s}_i |\mathbf{v}_i|. \quad (4.4)$$

If a collision event occurs, however, we can write down the effective drag force as

$$W + W_C = -\frac{\pi d_p^2 \rho_f}{8} C_D M_i \sum_i (\mathbf{v}_i - \mathbf{u}) \cdot \mathbf{s}_i |\mathbf{v}_i|, \quad (4.5)$$

where W_C is the work done by the particle collision. By combining these two equations, we get

$$\frac{C_D}{C_D^*} = 1 + \frac{W_C}{W}. \quad (4.6)$$

We obtain a value for W_C by calculating the difference in kinetic energy before and after the collision event. Our aim is to plot the values for C_D/C_D^* against the values of the void fraction ϵ .

Observe that the available experimental correlations for C_D/C_D^* are conducted in a strictly three dimensional setting. Our numerical experiments are carried out in two dimensions. Hence, we require a procedure to make our results comparable with the three dimensional data. To accomplish this, we will use a pseudo three-dimensional concept where the characteristic length in the third dimension is set equal to the particle diameter. Following the procedure from Helland et al. in [13], we transform the value of ϵ from our calculations to an effective three dimensional void fraction ϵ_{3D} by

$$\epsilon_{3D} = 1 - \frac{2}{3}(1 - \epsilon). \quad (4.7)$$

We will restrict our attention to the low Stokes number (St) domain. Recall that this paper is focused on the particle motion alone, and does not include the dynamics of the ambient fluid. Hence, we will set the fluid velocity $\mathbf{u} = 0$. We use the physical domain $[0.0, 5.0] \times [0.0, 5.0]$, which is partitioned in 151×151 grid cells. This division corresponds to a grid spacing of $\Delta x = \Delta y = 0.033m$. Moreover, the relaxation time is calculated to

$\tau_p=2.34s$, and the length of the time steps is chosen to be $\Delta t=1.0s$. The particles, of which there are 28900, are distributed randomly in the rectangular area $[1.0,2.0] \times [1.0,2.0]$ within the domain. Moreover, the particles have a radius $r=0.002m$, and density $\rho_p=2630kg/m^3$. Furthermore, the filling rate varies from $\phi = 0.58$ to $\phi = 0.21$ with an average of $\bar{\phi} = 0.4$, where the maximum filling rate is set to $\Phi=0.64$. Using the above expression for the void fraction, we get $\bar{\epsilon}_{3D}=0.73$.

Each particle is initialized with a velocity

$$\mathbf{v}_p = v_0(\cos\eta, \sin\eta), \tag{4.8}$$

where η is a uniformly distributed random variable in $[-\pi/3, \pi/3]$ and $v_0=0.1m/s$. From the particle velocities, the average velocity \mathbf{V} and the dispersion directions are calculated as described earlier in Section 2. The particles are then evolved one time step using our model.

At grid cells where a collision event occurs, the value of ϕ and C_D/C_D^* is recorded. We partition the interval $[\phi_{min}, \phi_{max}]$ into 15 equal parts, where a measured value of ϕ is associated with the midpoint of one of these intervals. Hence, we associate each of the measured drag correlations with the corresponding interval. Once the entire computational domain has been traversed, the average value of C_D/C_D^* is calculated for each ϕ -class. From the measured values of the filling rate, we calculate the void fraction ϵ , which henceforth will refer to the transformed three dimensional value.

4.1 The effect of collisions on the particles

We have performed a series of numerical experiments to explore the parameter space spanned by μ_C , e , and β . Recall that in this specific collision model, rotational dissipation is included. We will, however, focus mainly on the variation of the normal restitution coefficient e , since the simulation results indicate a higher sensitivity to this parameter. For each choice of values for the model parameters e , μ_C and β , we obtain a relation between C_D/C_D^* and ϵ . This allows us to calculate the logarithm of C_D/C_D^* and ϵ , respectively. For convenience, we set $D = \log(C_D/C_D^*)$ and $E = \log\epsilon$. If a power law relation is present, then we should have

$$\log \frac{C_D}{C_D^*} = -n \log \epsilon. \tag{4.9}$$

To quantify the degree of linearity of the data sets, we calculate the correlation coefficient [22] given by,

$$r = \frac{\sum_k (D_k - \bar{D})(E_k - \bar{E})}{[\sum_k (D_k - \bar{D})^2 \sum_k (E_k - \bar{E})^2]^{1/2}}, \tag{4.10}$$

where $\{E_k\}$ is the set of values of $\log\epsilon$, and $\{D_k\}$ is the set of values of $\log(C_D/C_D^*)$. Moreover, \bar{D} and \bar{E} are the mean values of $\{D_k\}$ and $\{E_k\}$, respectively. Recall that a value of $r = \pm 1$ implies a perfect linear relationship between the data sets.

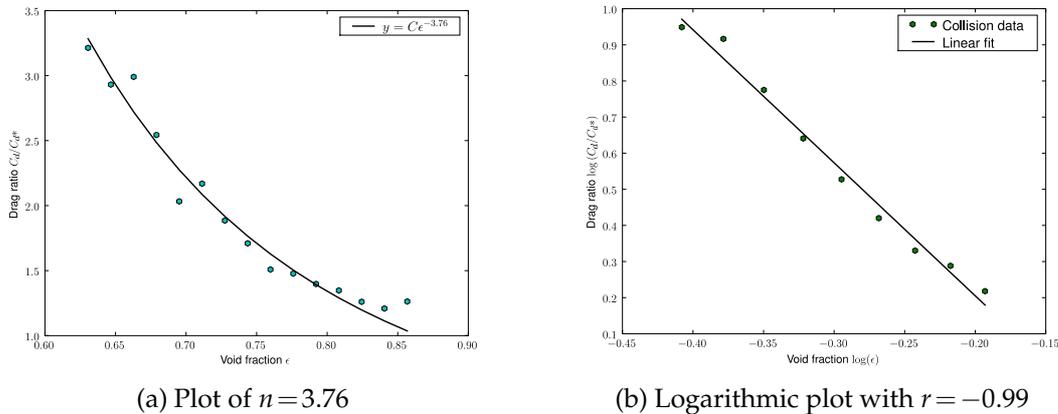


Figure 3: A power law correlation between C_D/C_D^* and ϵ . We have used the values $e = 0.95$, $\beta = 0.1$ and $\mu_C = 0.1$. This data plotted reproduces the value of $n = 3.76$ for the drag correlation index.

Fig. 3 shows the first of these experiments. What can be observed first is that there is indeed a power law correlation between the measured results. The plot of Fig. 3(a) shows the value $n = 3.76$ for the drag law above, where we have plotted a reference curve $y = C\epsilon^{-3.76}$. Moreover, Fig. 3(b) shows the logarithmic plot for three sets of parameter values, with a calculated correlation coefficient $r = -0.99$. We will shortly return to the significance of the constant C , which will illuminate an interesting property of our simulated data. But first, let us study some additional simulations.

In all of the simulations, we observe that the correlation factor r was within the range $0.99 - 0.98$. Fig. 4 shows three instances of numerical trials, where we have varied the restitution parameter e from 1.0 to 0.1. We can see a definitive decrease in the index n with lower values of e . This is consistent with a higher rate of collision energy dissipation with lower values of e . Within the range of the simulated values of e , we obtain values of n between 3.2 and 6.0.

The value of n , however, seem to fluctuate slightly for the same choice of parameters. This is expected, since we have randomized both the particle position and velocities. Fig. 5 shows three instances of the drag correlation for the same parameter values. By averaging over many simulations, however, we obtain a convergence towards $\bar{n} = 4.7$ for the parameter values $\beta = 0.1$, $\mu_C = 0.1$, and $e = 0.8$. This is a very good approximation to the classical result of $n = 4.67$. Moreover, we observe a decrease in the standard deviation of the average index, as illustrated in Fig. 6.

In the expression for the collision number N_C , Eq. (3.8), we used the radial function

$$g_0(\phi) = \left[1 - \left(\frac{\phi}{\Phi} \right)^{1/3} \right]^{-1}. \tag{4.11}$$

For completeness, we compare the three radial functions described in Section 3.1. That is, we compare the above function with the two defined by Eq. (3.6) and Eq. (3.7). We perform a series of 10 experiments for parameter values $e \in [0.1, 0.9]$ for each of the radial

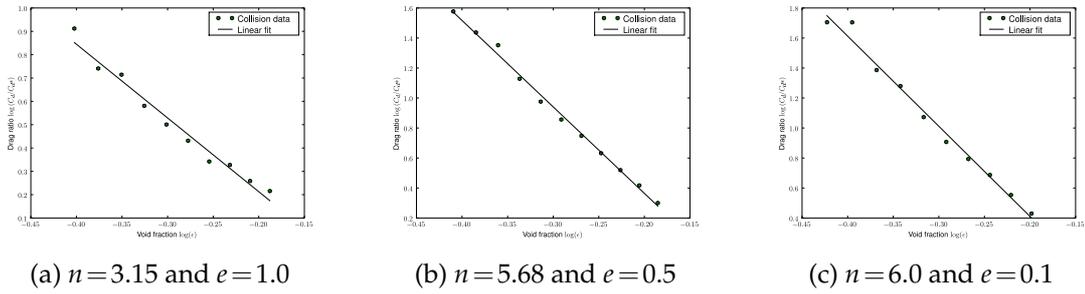


Figure 4: Correlations between $\log[C_D/C_D^*]$ and $\log \epsilon$. In all the plots, we have that $\beta = 0.1$ and $\mu_c = 0.1$.

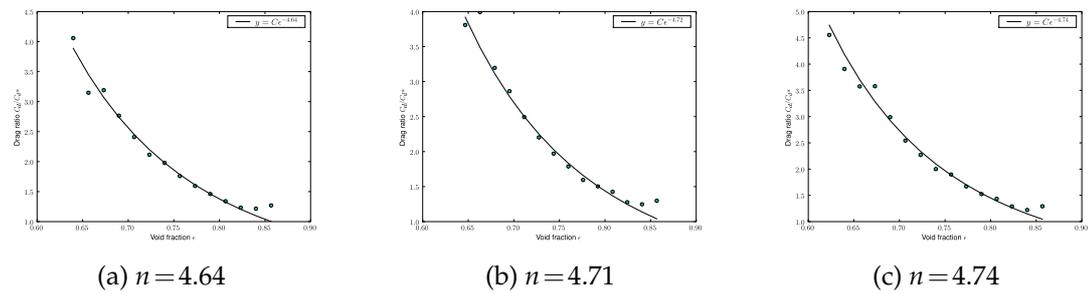


Figure 5: Three simulations of C_D/C_D^* and ϵ with $\beta = 0.1$, $\mu_c = 0.1$, and $e = 0.8$. In each experiment, the calculated value of n varies slightly.

functions. Fig. 7 shows the result of these numerical trials. We can observe a slight difference in the drag correlation index. The function g_2 gives the highest index values, while g_1 is correlated with the lowest index values.

There are three important aspects to consider. First, for the types of particles studied here, we do not expect that the normal restitution coefficient to become much lower than $e = 0.9$. Moreover, many-particle collision events should occur in these high concentrations, which we have not accounted for. Furthermore, the remaining physical effects valid for these particle laden flows are very difficult to simulate. But as we showed earlier, our results successfully reproduce physical experiments in hindered settling. Recall that we obtained a good approximation for the classical value of $n = 4.67$ for creeping flow. By allowing the normal restitution parameter to decrease below what is usual, we can conclude that we can actually simulate the complete hindered settling effect by a binary collision model. This means that we can compensate for not including these physical aspects into our framework by increasing the collisional effect.

From our simulations, we obtain that the lowest value for the drag correlation index is $n = 3.0$. Moreover, we can even obtain higher values of n than 4.7, up to $n = 6.0$. Using the radial function g_3 , we get the maximum of $n = 6.5$. Hence, our simulations indicate that the hindered settling effect can be even more profound than previously expected. This has recently been shown in [25], where experiments on sand grains produced high values of the drag correlation index, up to $n = 6.0$.

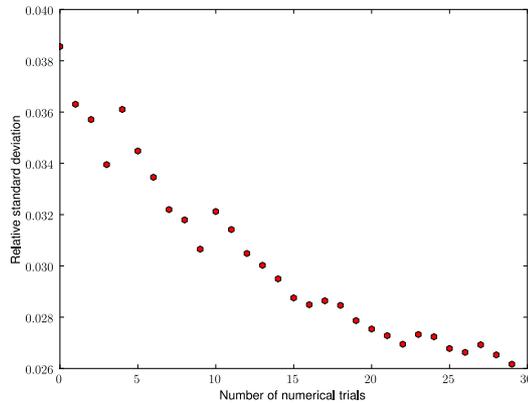


Figure 6: The standard deviation of the simulated data for $\bar{n} = 4.7$.

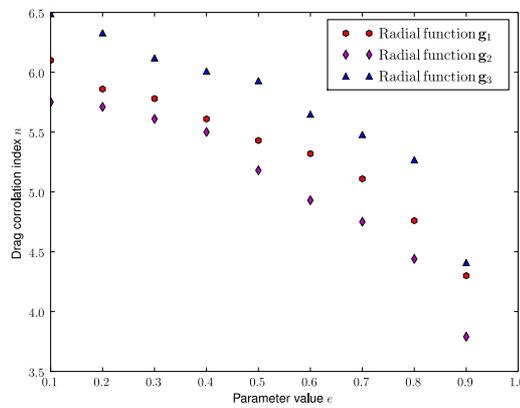


Figure 7: The value of the correlation index n as a function of the restitution coefficient e . All of these experiments has been done with $\mu_C = 0.1$ and $\beta = 0.1$. Three different radial functions has been used; The function g_0 defined by Eq. (4.11), g_1 defined by Eq. (3.6) and g_3 defined by Eq. (3.7)

We now return to the constant C mentioned briefly above. The fact that $C \neq 1$ shows that our data is biased in some fashion. However, the value of $C \sim 0.5$ for all the simulations we have carried out, which represents the average value over the computational domain. This biasing factor has a very simple explanation. Observe that we have shown with the above analysis that

$$C_D = C\epsilon^{-n}C_D^*. \tag{4.12}$$

Recall that in each simulation, $\bar{\epsilon} = 0.73$, with $\bar{\epsilon}^2 = 0.53$. The value of C varies from $\exp(-0.60)$ to $\exp(-0.77)$, which is approximately ~ 0.5 , indicating that $C \simeq \epsilon^2$. To test this, we simulated some additional cases with $\bar{\epsilon} = [0.5, 0.8]$. We consistently observe that the constant $C \simeq \bar{\epsilon}^2$. Moreover, it is independent of the radial function used in the simulations. This indicates that $C_D = \bar{\epsilon}^2\epsilon^{-n}C_D^*$. One way of interpreting this bias is that the

effective drag force, Eq. (4.2), must be modified to

$$\mathbf{F}_d = -\frac{\pi d_p^2 \rho_f}{8} \epsilon^2 \epsilon^{-n} C_D^* (\mathbf{v} - \mathbf{u}) |\mathbf{v}|. \quad (4.13)$$

This is, however, the exact form of the drag force used in most applications with granular flow [13]. Even though that the above analysis is not an exact a proof, we find it very pleasing that the extra ϵ^2 factor in the drag force is reproduced by our data. This acts as an additional verification of our results, and reveals some of the potential of the lumped particle framework.

5 Conclusions

In this paper we have extended the lumped particle framework in [1] to include particle collisions. Using concepts from kinetic theory, we introduced a collision procedure where we used a binary collision model from [10] which includes rotational energy dissipation. By testing this framework on the hindered settling effect, we observed that the lowest value of the drag correlation index is $n=3.0$. We also showed that we can compensate for the many physical effects not included in the model by allowing high collisional energy dissipation: numerical simulations show that we can obtain $n=4.7$, which is very close to the classical value from physical experiments on creeping flow. Moreover, the experiments show that the slight bias in the simulated data give rise to a modified effective drag force which includes a factor of ϵ^2 . This is consistent with a widely used expression for the drag force in granular flows. Furthermore, high values of the drag index $n=6.5$ is also possible to simulate, which is consistent with recent physical experiments on the sedimentation of sand grains.

To conclude, the proposed binary-based collision procedure, embedded in the lumped particle framework seems very promising, as it quantitatively captures several key features observed in physical experiments on dense particle flows. Moreover, since our model works with lumped particles, the computational procedure is also possibly much more efficient than models based on pure discrete particle methods.

Acknowledgments

The presented work was funded by a research grant from Statoil. The work has been conducted at Simula Research Laboratory as part of CBC, a Center of Excellence awarded by the Research Council of Norway.

References

- [1] O. al Khayat, A. M. Bruaset and H. P. Langtangen, A lumped particle modeling framework for simulating particle transport in fluids, *Commun. Comput. Phys.*, 8(1) (2010), 115–142.

- [2] B. J. Alder and T. E. Wainright, Studies in molecular dynamics ii, behaviour of small number of elastic spheres, *J. Chem. Phys.*, 33 (1960), 1439–1451.
- [3] S. Chapman and T. G. Cowling, *The Mathematical Theory of Non-Uniform Gases*, Cambridge University Press, 3rd edition, 1970.
- [4] C. Crowe, M. Sommerfield and T. Yutaka, *Multiphase Flows with Droplets and Particles*, CRC Press, 1998.
- [5] J. S. Curtis and B. van Wachem, Modeling particle-laden flows: a research outlook, *AIChE J.*, 50(11) (2004), 2638–2645.
- [6] N. G. Deen, M. Van Sint Annaland, M. A. Van der Hoef and J. A. M. Kuipers, Review of discrete particle modeling of fluidized beds, *Chem. Eng. Sci.*, 62 (2007), 28–44.
- [7] J. Ding and D. Gidaspow, A bubbling fluidization model using kinetic theory of granular flow, *AIChE J.*, 36 (1990), 523–538.
- [8] R. Felice, The void function for fluid-particle interaction systems, *Int. J. Multiphase. Flow.*, 20 (1994), 153–159.
- [9] S. Foerster, M. Louge, H. Chang and K. Allia, Measurements of the collision properties of small spheres, *Phys. Fluids.*, 6 (1994), 1108–1115.
- [10] M. J. V. Goldschmidt, R. Beetstra and J. A. M. Kuipers, Hydrodynamic modelling of dense gas-fluidised beds: comparison of the kinetic theory of granular flow with 3d hard-sphere discrete particle simulations, *Chem. Eng. Sci.*, 57(11) (2002), 2059–2075.
- [11] T. I. Gombosi, *Gaskinetic Theory*, Cambridge University Press, 1994.
- [12] T. C. Hales, Proof of the kepler conjecture, Internet, 1998, <http://www.math.pitt.edu/~thales/kepler98/>.
- [13] E. Helland, H. Bournot, R. Occelli and L. Tadrict, Drag reduction and cluster formation in a circulating fluidised bed, *Chem. Eng. Sci.*, 62(1-2) (2007), 148–158.
- [14] E. Helland, R. Occelli and L. Tadrict, Computational study of fluctuating motions and cluster structures in gas-particle flows, *Int. J. Multiphase. Flow.*, 28 (2002), 199–223.
- [15] C. Lun, S. Savage, D. Jefferey and N. Chepurini, Kinetic theories for granular flow: inelastic particles in couette flow and slightly inelastic particles in a general flowfield, *J. Fluid. Mech.*, 140 (1984), 223–256.
- [16] D. Ma and G. Ahmadi, An equation of state for dense rigid sphere gases, *J. Chem. Phys.*, 84 (1960), 3449–3450.
- [17] Y. T. Makkawi and P. C. Wright, The voidage function and effective drag force for fluidized beds, *Chem. Eng. Sci.*, 58 (2003), 2035–2051.
- [18] A. D. Maude and R. L. Whitmore, A generalized theory of sedimentation, *British J. Appl. Phys.*, 9 (1958), 477–482.
- [19] E. Meiburg and B. Kneller, Turbidity currents and their deposits, *Ann. Rev. Fluid. Mech.*, 42 (2010), 135–156.
- [20] Y. Nio and M. Garca, Using lagrangian particle saltation observations for bedload sediment transport modelling, *Hydrol. Process.*, 12(8) (1998), 1197–1218.
- [21] J. F. Richardson and W. N. Zaki, Sedimentation and fluidization: part I, *Trans. Institute. Chem. Eng.*, 32 (1954), 35–53.
- [22] J. Lee Rodgers and W. Alan Nicewander, Thirteen ways to look at the correlation coefficient, *Am. Stat.*, 42(1) (1988), 59–66.
- [23] N. Oshima, S. Ogawa and A. Unemura, On the equations of fully fluidized granular materials, *Zeitschrift fuer Angewandte Mathematik und Mechanik Physics*, 31 (1989), 483–493.
- [24] J. S. Shirolkar, C. F. M. Coimbra and M. Queros McQuay, Fundamental aspects of modeling turbulent particle dispersion in dilute flows, *Pro. Energy. Combust. Sci.*, 22 (1996), 363–399.

- [25] M. R. Tomkins, T. E. Baldock and P. Nielsen, Hindered settling of sand grains, *Sedimentology*, 52 (2005), 1425–1432.
- [26] B. G. M. van Wachem and A. E. Almstedt, Methods for multiphase computational fluid dynamics, *Chem. Eng. J.*, 96(1-3) (2003), 81–98.
- [27] R. G. Walker, The origin and significance of the internal sedimentary structures of turbidites, *P. Yorks. Geol. Soc.*, 35 (1965), 1–32.
- [28] S. Wang, X. Li, H. Lu, L. Yu, J. Ding and Z. Yang, DSMC prediction of granular temperatures of clusters and dispersed particles in a riser, *Powder. Tech.*, 192 (2009), 225–233.
- [29] C. Y. Wen and Y. H. Yu, Mechanics of fluidization, *Chem. Eng. P. Symp. Ser.*, 62 (1966), 100–111.