Jens Andreas Melheim

Computer simulation of turbulent electrocoalescence

Thesis for the degree of philosophiae doctor

Trondheim, March 2007

Norwegian University of
Science and Technology
Faculty of Engineering Science and Technology
Department of Energy and Process Engineering

NTNU
Innovation and Creativity

NTNU
Norwegian University of Science and Technology

Thesis for the degree of philosophiae doctor

Faculty of Engineering Science and Technology
Department of Energy and Process Engineering

ISBN 978-82-471-0881-9 (printed ver.)
ISBN 978-82-471-0895-6 (electronic ver.)
ISSN 1503-8181

Doctoral Theses at NTNU, 2007:41

Printed by Tapir Uttrykk

Abstract

Offshore wells produce some water, and the ratio of water increases during the lifetime of a well, in particular when water is injected to increase the extraction rate. Hence, oil companies demand techniques that enhance the separation of oil and water. A speed-up of the separation process is achieved by applying electric fields to turbulent-flow water-in-oil emulsions. The electric field gives rise to attractive forces between close droplets and increases the probability of coalescence at contact, while the turbulence enhances the frequency of droplet collisions.

To improve the understanding of the mutual interaction between the turbulence and the electric field, this thesis presents a framework for computer simulation of turbulent electrocoalescence. The framework is based on the Eulerian-Lagrangian approach where each droplet is tracked and the electric and the hydrodynamic interactions between the droplets are handled.

The forces working between two droplets in stagnant oil are modelled and compared with experimental data. It was found that the electric dipole-dipole forces and the film-thinning forces dominate at small droplet spacings.

The turbulence felt by the droplets is modelled by a stochastic differential-equation model. A new model is proposed to correlate the fluid velocities seen by close droplets, and this is important for the prediction of the collision velocity, the collision frequency, and the clustering of droplets.

Two algorithmic improvements are made: An adaptive cell structure and the cluster integration method. The proposed adaptive cell structure adapts to the number density of droplets and ensures an efficient computation without any input from the user regarding the cell structure. The cluster integration method assembles clusters of droplets that interact and integrates each cluster separately using a variable step-size Runge-Kutta method. A significant speed-up compared to traditional approaches is reported.

Finally, the results obtained by computer simulations of turbulent electrocoalescence agree qualitatively with experimental observations in the literature.

Samandrag

Frå oljebrønnar til havs kjem det opp ein del vatn saman med oljen. Mengda av vatn aukar i løpet av produksjonstida, særleg dersom oljeselskapa pumpar vatn attende i brønnen for å få ut endå meir olje. Oljeselskapa spør derfor etter teknologi som skil vatn og olje meir effektivt enn dagens utstyr. Å nytte elektriske felt på turbulente olje-vatn-blandingar er ein lovande separasjonsteknologi. Det elektriske feltet skapar tiltrekkjande krefter mellom nære vassdråpar og aukar sannsynet for samanslåing ved kontakt, medan turbulensen gjev hyppige dråpekollisjonar.

Avhandlinga presenterer eit rammeverk for numeriske simuleringar som kan nyttast til betre å forstå vekselverknaden mellom turbulens og elektrisk felt. Rammeverket byggjer på ein dråpefølgjingsmetode, der ein modellerer kreftene som verkar på kvar enkelt dråpe og reknar ut dråpebanene. Arbeidet legg vekt på modelleringa av kreftene som verkar mellom dråpane, modelleringa av turbulensen og på utvikling av eigna algortimar.

Dei viktigaste nyvinningane i arbeidet er ein modell finn samanheng mellom rørslene til nære dråpar i turbulensen og ein algoritme som er vesentleg mindre reknekrevjande enn eksisterande algoritmar.

Numeriske simuleringar av turbulente olje-vatn-blandingar syner dei same trendane som eksperimentelle forsøk i litteraturen. Det føreslegne rammeverket kan derfor nyttast under utforming av framtidig separasjonsutstyr.

Acknowledgements

I am very grateful for the three-year personal PhD grant and the additional overseas fellowship I received from the Research Council of Norway.

I would like to thank my supervisor, Professor Inge R. Gran for initializing an interesting research topic for my doctoral work and for inspiring advises. I appreciate the freedom I have been given to develop my own ideas. I thank my second supervisor, Professor Morten Chr. Melaaen at Telemark College for reading and remarking the papers that have been sent him. I also would like to thank Professor Martin Sommerfeld at Martin-Luther Universität Halle-Wittenberg for inviting me to stay half a year together with his group in Merseburg.

At the Department of Energy Processes Engineering at NTNU and the co-located SINTEF Energy Research department, Energy Processes, there is always someone to ask. First of all, I would like to thank my colleague Svend Tollak Munkejord for the many discussions and exchanges of views we had, for reading and commenting my manuscripts and for patiently correcting my English. It is always inspiring to work with Dr Matteo Chiesa and I have enjoyed the journeys we had together. I also appreciate the discussions with Dr Erik B. Hansen, Dr Ola Maurstad, Dr Roar Meland and last, but not least, Dr Robert Olsen, who also helped me with computer and compiler problems.

In the beginning of my PhD study, I had the joy and honour to work with Dr Anders Gjelsvik at SINTEF Energy Research, who also introduced me to scientific writing and some programming pearls.

Another source of inspiration has been the Electrocoalescence project at SINTEF Energy Research. I thank Mr Lars Lundgaard for letting me participate. The Electrocoalescence project has been an arena for fruitful discussions with partners from the industry and other NTNU departments. The discussions with Dr Pål Jahre Nilsen from Vetco Aibel have been particularly valuable.

At Martin-Luther Universität Halle-Wittenberg, I would like to thank Mr Stefan Blei, Dr Frank Schaub, Dr Stefan Horender and family, and my office-mate,

Mr Thomas Schulz, for helping me getting installed in Merseburg and for making the stay in Merseburg a memorable time. Dr Stefan Horender should also be acknowledged for the discussions we had on particle trajectories in turbulence.

My friends on the 'sognalag-mailing-list' and 'kæmmisene' deserve a special thank-you. It really helps to have social life outside of the doctoral-work sphere.

Many thanks to my family for always supporting me. Finally, I would like to thank my wife Turid and our daughters Brita and Sigrid for helping me remember what is important in life.

Jens Andreas Melheim

Bergen, February 2007

Contents

1

Introduction

1.1 Motivation for the thesis

Oil in the forms of fuel and petrochemical products is important in our everyday life. Transportation requires fossil fuels, and wardrobe, sporting equipment, etc. are to a large extent made of oil-based products.

There is no doubt about the fact that oil is a finite resource, but until now, the world's production has been ever increasing. However, this situation will change. 'In our life time, we will have to deal with a peak in the supply of cheap oil', says Robert A. Kaufmann (Appenzeller, 2004). The point in time where the global oil production reaches its maximum is often called *peak oil*. Peak oil is a watershed moment, then we will really feel that oil is a finite resource. As long as substitutes are not easily available, the price of oil will increase and the oil companies can then allow themselves to produce oil at a higher cost. Increased extraction of oil from already existing wells can extend the moment of peak oil and make the decline in production less steep. The Norwegian government wants to increase the extraction rate in the North Sea from 47% (2004) to beyond 50% (Storting proposition no 38 , 2003-2004).

In the tail production of oil wells in the North Sea, an increasing amount of water is produced, and it has to be separated from the oil before sale. The wells contain some water initially, and injection of water is a widely used technique for increasing the oil extraction rate. To keep the production rate stable on already existing platforms, one has to separate the emulsion of oil and water more efficiently than today.

There are several techniques to de-emulsify or separate a stable water-in-oil emulsion, where the water droplets are so fine and the oil is so viscous that the droplets hardly sediment (Eow et al., 2001). *Heating* changes the properties of the oil, the emulsion turns unstable and the droplets sediment faster. The drawback is that one has to use a significant amount of produced oil or natural gas for heating, and that has a considerable cost. *Chemical de-emulsifiers* can be added to de-stabilise the emulsion. These chemical de-emulsifiers are

expensive and they have to be separated from the water and oil afterwards. Another approach, which is cheap and has no environmental impact, is electrostatic destabilisation, *electrocoalescence*. One benefits from the very different electric properties of oil and water; oil is an insulator and water is a conductor (Lundgaard et al., 2005). Electrocoalescence has been known since the beginning of the 20th century when Cottrell patented it (Eow et al., 2001). An electric field has traditionally been applied on a stagnant or laminar flow emulsion. Atten (1993) found that it is much more efficient to apply an electric field to a turbulent-flow emulsion. Turbulence mixes the emulsion very well and brings the droplets close to each other, while the electric field defeats the film-drainage forces and increases the probability of successful coalescence at contact.

There is little work on the effect of turbulence on electrocoalescence in the literature. Atten (1993) suggested the use of high shears to increase the number of droplet meeting. Urdahl et al. (2001) reviewed the effect on the collision rate of different types of shears and turbulence. The mutual effect of turbulence and an electric field has been studied experimentally by Harpur et al. (1997) and Friedemann et al. (2001). The aim of this thesis is to investigate numerically, by using computational fluid dynamics (CFD), the interaction between turbulent flow and electric field.

1.2 Electrocoalescence

Water with a salt content of 3.5 wt% NaCl behaves like an ideal conductor compared to crude oil, which can be considered as an ideal insulator. When an electric field is applied to water droplets suspended in oil, the water droplets are polarised. The positive and negative ions travel towards the surface to keep the potential over the droplet close to zero. The electric field is influenced by the presence of the droplets. Two droplets in a homogeneous background field $\boldsymbol{E}_0$ are sketched in Figure 1.1. The resulting field is inhomogeneous and gives rise to dielectrophoretic forces between the droplets, dipole-dipole forces $\boldsymbol{F}_e$. When the droplets are aligned with the field, the dipole-dipole forces are attractive.

Eow et al. (2001) divided the process of electrocoalescence into three stages:

1. The droplets' approaching each other.
2. The process of film-thinning.
3. Film rupture leading to droplet–droplet coalescence.

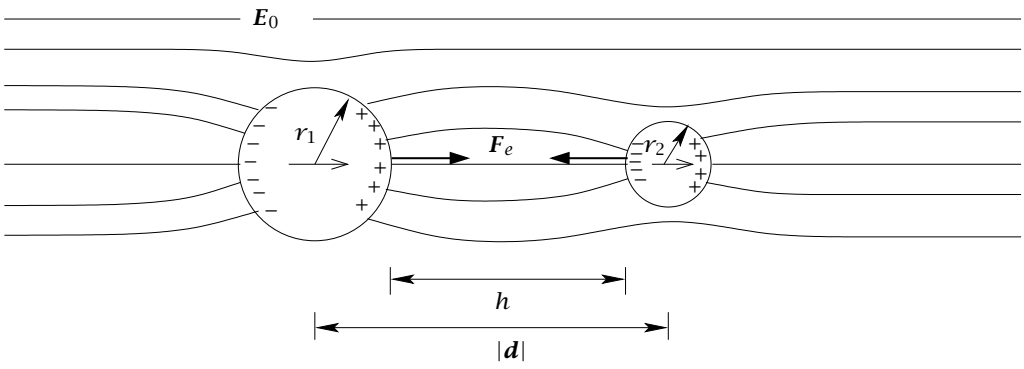


Figure 1.1: Electric forces F_e between two conductive spheres with radii r_1 and r_2, separation vector d, and surface to surface distance h. E_0 is the homogenous background field. From Melheim et al. (2004).

The three stages are distinguished by the inter-surface distance h: Stage 1 is when the distance between two droplets is larger than the radius of the smaller droplet, $h > r_2$, and the drag force is the most pronounced force describing the motion of the droplets. Stage 2 is the intermediate range, $0.1\,\mu\text{m} < h \leq r_2$, where the attractive electric forces suppress the oil film. Stage 3 is at even smaller separation distances, $h < 0.1\,\mu$m, where film-rupture and coalescence take place.

Eow et al. (2001) reviewed the literature on electrocoalescence thoroughly and discussed the effect of different types and strengths of the electric field, of voltage shape and frequency, and of the volume fraction of droplets. Urdahl et al. (2001) gave a historical review and an overview of different mechanisms bringing the droplets closer. The dimensional parameters for droplet collisions in turbulence discussed by Urdahl et al. (2001) is based on the expression for collision frequency of Saffman and Turner (1956), which is valid when the characteristic response time of the droplets, τ_p, is much less than the Kolmogorov time scale of the turbulence, τ_η. In turbulent electrocoalescers, strong electric forces may increase the frequency of collisions and only large droplets may have response times comparable to the Kolmogorov time scale. Lundgaard et al. (2005) focused on the instabilities of droplets and surfaces in addition to give an introduction to basic electrostatics. Hansen (2005) used a level-set method to calculate Stages 2 and 3 numerically. In the field of electro-rheology (Klingenberg et al., 1991), conductive metal spheres suspended in stagnant and laminar-flow oil are investigated. The physics regarding the dipole-dipole forces and the film-thinning forces are equal for metal spheres and small droplets and the expressions for the forces on metal

spheres can be adapted to water droplets. However, there is an important difference between rigid spheres and water droplets; the surface of the water droplets deforms and starts to move when the shear-stresses on the droplet surface get very high.

In this work, attention is given to describe numerically the behaviour of suspended water droplets in a turbulent-flow oil with an applied electric field. At contact (Stage 3), instantaneous coalescence is assumed. Different numerical approaches to calculate turbulent electrocoalescence are discussed in the next section.

1.3 Numerical approaches

To describe the behaviour of particles suspended in a fluid, there are several approaches both for the fluid phase and the particle phase. The term 'particle' denotes the dispersed phase that can be solid particles, droplets or bubbles. The fluid phase is the continuous phase. In the following, various techniques for the fluid flow will be briefly described. Then approaches for the particle phase and possible combinations with the fluid-phase methods will be discussed with respect to turbulent electrocoalescence. A more complete overview of numerical approaches for dispersed multi-phase flows is given by Loth (2000).

Fluid-phase approaches

The fluid phase is either laminar or turbulent, and usually handled in the Eulerian way, which means that the Navier-Stokes equations for the fluid are solved on a grid, contrary to Lagrangian methods. Laminar flows are rarely found in industrial processes, but they occur in large separation vessels and in thin films, as for instance the oil film between two close water droplets. There are mainly three levels in computation of turbulence; direct numerical simulations (DNS), large-eddy simulations (LES) and Reynolds-averaged Navier-Stokes equations (RANS) simulations.

In DNS, all time and length scale of the turbulence are solved without modelling. Unfortunately, DNS is computationally expensive and one is limited to homogeneous turbulence and simple shear flows with low Reynolds numbers and cyclic boundaries in the flow direction.

LES is, as DNS, unsteady simulations, but the smallest scales are filtered and modelled. One assumes that the smallest scales are isotropic and that energy only travels from the larger scales to the smaller, modelled scales. LES is

more robust than DNS, but one is still limited to small and moderate Reynolds numbers.

In RANS, the governing equations for the fluid flow are averaged, and the Reynolds stresses ($-\overline{u_i u_j}$) in the momentum transport equations are modelled. The Reynolds stresses can be found either by solving transport equations for them, Reynolds-stress models (RSM), or by employing the eddy-viscosity hypothesis. The most popular way to estimate the eddy viscosity is by using the k-ε-model, where transport equations are solved for the turbulent kinetic energy k and its dissipation ε. RANS is best suited for high-Reynolds-number turbulence.

Particle-phase descriptions

The particle phase can be handled both by an Eulerian and a Lagrangian formulation. The Eulerian approach is suited both for very detailed calculations of systems with two different fluids and for the coarse multi-fluid approach, where the particle phase is handled as a fluid. In the Lagrangian approach, the particles are tracked and the motion is described by the forces working on the particles.

The DNS methods for Eulerian treatment of droplets and bubbles handle the fluids as one single fluid with a jump in the fluid properties at the position of the interface. There are several methods to handle the interface, examples are the front-tracking method (Tryggvason et al., 2001), the level-set method (Osher and Fedkiw, 2001; Hansen, 2005) and the volume-of-fluid (VOF) (Scardovelli and Zaleski, 1999). The strengths of these methods are seen in the study of phenomena like break-up and coalescence. Laminar flow is most often considered, but these techniques have been combined with low-Reynolds-number turbulent flows, in particular the VOF method.

For large scales, the dispersed phase is handled as one or more separate fluid phases and one has the two-fluid or multi-fluid models. The governing equations for the particle phase are found by volume averaging (Nigmatulin, 1991, Chap. 1.2) or ensemble averaging (Drew and Passman, 1999, Chap. 9). The inter-phase terms and Reynolds-stress-like terms have to be modelled to close the equations. Empirical relations (Ergun, 1952) and kinetic theory (Gidaspow, 1994) have been used to make closure models. A more recent closure approach for dilute flows is the probability-density-function (PDF) approach (Simonin et al., 1993). The closure models in the PDF approach are related to a Lagrangian description of the particle motion, where it is easier to model for instance turbulent dispersion. Because it can handle an innumerable amount of particles, the multi-fluid approach is implemented in commercial

CFD-codes. However, the modelling of different particle sizes requires one particle phase for each particle size and the interactions between the particles and particle sizes are complex to model. Multi-fluid models are most widely combined with RANS for the fluid flows, but some combinations with LES are also seen.

The great advantage of the Lagrangian approach is that the forces working on each particle can be written down straightforwardly, which makes it easy to handle complex physics.

The most detailed of the Lagrangian methods with freely moving particles is the immersed boundary method, where the forces on a solid particle from the fluid phase and vice versa are calculated on area segments of each particle.

While the particles are larger than the grid size in immersed boundary methods, the particles are much smaller than the grid in point-particle methods. The modelled forces are provided with a particular physical meaning and they are assumed to give separate contributions to the force balance, and they are added together. In the discrete particle model (DPM), one numerical particle represents one physical particle, and the inter-particle forces like dipole-dipole forces, hydrodynamic interactions and collision forces can be modelled directly. The drawback with DPM is the limited number of particles that can be handled due to computational resources. The computational barrier can be passed by letting one numerical particle (parcel) represent many real particles with equal physical properties; the particle cloud model (PCM). Collisions are then modelled by a stochastic model (Sommerfeld, 2001). DPM can be combined with DNS calculations of the fluid phase (Patankar and Joseph, 2001), LES (Laviéville et al., 1995) and RANS (Tsuji et al., 1993). PCM is best suited for stationary calculations and therefore only combined with RANS.

Conclusion on numerical approach

Table 1.1 summarises possible[1] combinations of numerical approaches for the fluid phase and the particle phase, where the asterisk $*$ indicates the approach in this work.

Recalling the aim of the present work, that is to investigate numerically the interaction between turbulence and the electric field in coalescence, poses limitations to the numerical methods. First, the flow is turbulent and many droplets are needed, which eliminates the DNS methods for the particle phase. The inter-droplet physics are complex, one has to deal with both dipole-dipole forces and hydrodynamic interactions and the droplet size spans from 10 μm to 1 mm. Such complex physics cannot be captured without simplifications

[1] Other combinations are indeed possible, but they are not rational.

Table 1.1: Combinations of numerical approaches.

Fluid phase		**Particle phase**				
		Eulerian		Lagrangian		
		DNS	Multifluid	Big particles	Point particles	
					DPM	PCM
Laminar		•	•	•	•	•
Turbulent	DNS	•		•	•	
	LES		•		•	
	RANS		•		∗	•

in the multi-fluid framework and the particle cloud model (PCM). Then one is left with the discrete particle model (DPM). Due to the droplet growth, it is not possible to use DNS for the fluid flow calculations. LES is a good candidate because it captures the large-scale mixing that brings the droplets together and the turbulence is supposed to be moderate. Because of the filtering, the smallest scales of the turbulence are not 'seen' by the particles in LES. LES is therefore inaccurate when the droplet response time is shorter than the smallest time scale resolved. Nevertheless, RANS was chosen, mainly because it already was available. The advantages of RANS are its robustness and low computational costs, in particular for steady-state calculations. The drawback is that the instantaneous fluid velocities seen by the particles have to reproduced from the mean values provided by the RANS. Challenges in the chosen numerical approach are pointed out in the following sections.

1.4 Discrete Particle Model

A discrete particle model tracking each particle in a domain is a powerful tool, in particular when complex physics is considered. Examples are simulations of molecules (Allen and Tildesley, 1987), granular flows (Cundall and Strack, 1979), fluidised beds (Hoomans et al., 1996; Xu and Yu, 1997), sedimentation of clays (Anandarajah, 1999), electro-rheology (Lukkarinen and Kaski, 1998), particle collisions in turbulence (Laviéville et al., 1995; Wang et al., 2000), and combustion synthesis of nanoparticles (Zuccaro et al., 2004).

Because it is assumed that forces describing different physics can be superpositioned, it is easy to add new forces. DPM in combination with RANS requires a careful handling of the turbulence. RANS provides only the mean values of the Reynolds stresses and the dissipation of turbulent kinetic energy.

The instantaneous velocities must therefore be modelled. This is crucial, because the dispersion of particles and the particle collision rate depend on the turbulence.

However, physical models are only half of a discrete particle model; efficient algorithms, suitable data structures and good numerical solvers are also needed to get reliable results in a reasonable time. For further studies it is convenient to break the discrete particle model into four fundamental elements:

1. An appropriate description of the *forces working on the particles,* suited to the case one likes to study.
2. A sufficiently physical *realisation of the instantaneous velocity* at the particle position due to turbulence.
3. *Algorithms* that can handle many particles efficiently.
4. Stable and accurate *numerical solvers*.

These four topics are treated in the following, with respect to numerical simulations of turbulent electrocoalescence.

Forces working on droplets

The forces working on the particle is analytically expressed by the following surface integral, appearing from a volume average of the momentum equation (Nigmatulin, 1991):

$$\boldsymbol{F}_{\text{fluid}} = \int_{\mathcal{A}_\text{d}} (-p_s \boldsymbol{n}_\text{d} + \tau_d \cdot \boldsymbol{n}_d)\, \text{d}A, \tag{1.1}$$

where $\mathcal{A}_\text{d}$ is the particle surface area, p_s is the relative surface pressure, τ_d is the stress tensor on the surface, and $\boldsymbol{n}_\text{d}$ is the surface normal vector.

A Lagrangian formulation of the equations governing the particle motion is employed in the discrete particle model and the models for the forces are related to different physical phenomena such as drag, lift, added mass, buoyancy, etc. The sum of those forces that have to do with particle-fluid interaction should be consistent with the surface integral in equation (1.1). Reviews of models for forces working on point particles, including droplets and bubbles, are given by Crowe et al. (1998), Loth (2000) and Michaelides (2003).

The position $\boldsymbol{x}$ and velocity $\boldsymbol{V}$ of the particles are calculated by

$$\frac{d\boldsymbol{x}}{dt} = \boldsymbol{V}, \tag{1.2}$$

$$\frac{d\boldsymbol{V}}{dt} = \frac{1}{m_p} \sum \boldsymbol{F}_p, \tag{1.3}$$

where m_p is the particle mass. The sum of the forces working on a particle $\sum \boldsymbol{F}_p$ can be written as:

$$\sum \boldsymbol{F}_p = \boldsymbol{F}_D + \boldsymbol{F}_g + \boldsymbol{F}_L + \boldsymbol{F}_H + \boldsymbol{F}_A + \boldsymbol{F}_{P-P} + \boldsymbol{F}_W, \tag{1.4}$$

where $\boldsymbol{F}_D$ is the drag force, $\boldsymbol{F}_g$ is the gravity force, $\boldsymbol{F}_L$ is the lift forces, $\boldsymbol{F}_H$ is the history force, $\boldsymbol{F}_A$ is the added-mass force, $\boldsymbol{F}_{P-P}$ is the inter-particle forces and $\boldsymbol{F}_W$ is the particle-wall forces.

The drag force is generally given by

$$\boldsymbol{F}_D = \frac{1}{2}\rho C_\mathrm{d} A \left|\boldsymbol{U} - \boldsymbol{V}\right| \left(\boldsymbol{U} - \boldsymbol{V}\right), \tag{1.5}$$

where ρ is the fluid density, A is the projected particle area, and $\boldsymbol{U}$ is the fluid velocity. The drag coefficient C_d depends on the particle Reynolds number $Re_p = \rho \left|\boldsymbol{U} - \boldsymbol{V}\right| d_p/\mu$, the viscosity ratio between the droplet phase and the continuous phase, $\lambda = \rho/\rho_p$, and the mobility of the droplet surface. In paper A (Melheim et al., 2004), a small droplet falling on a bigger droplet in a stagnant oil under the influence of an electric field was considered. It was found that the expression of LeVan (1981) for the drag coefficient gave reasonable results. The LeVan expression reads:

$$C_d = \frac{24}{Re_p} \frac{3\lambda + 2 + 4\kappa(\mu d_p)^{-1} + 2/3\gamma_1(\mu|\boldsymbol{U} - \boldsymbol{V}|)^{-1}}{3\lambda + 3 + 4\kappa(\mu d_p)^{-1}}, \tag{1.6}$$

where the surface dilational viscosity was neglected ($\kappa = 0$) and a magnitude in the variation of the surface tension $\gamma_1 = 10^{-5}\,\mathrm{N/m}$ was used.

The gravity (buoyancy) force is given by

$$\boldsymbol{F}_g = \left(\rho_\mathrm{p} - \rho\right) g V_p \boldsymbol{e}_g, \tag{1.7}$$

where g and $\boldsymbol{e}_g$ are the modulus and the direction of the gravity, and V_p is the particle volume.

The added-mass force is due to the displacement of fluid and is given by

$$\boldsymbol{F}_A = \frac{m_p}{2}\left(\frac{D\boldsymbol{U}}{Dt} - \frac{d\boldsymbol{V}}{dt}\right), \tag{1.8}$$

where $\frac{1}{2}\frac{d\boldsymbol{V}}{dt}$ can be put on the left-hand side of the momentum equation (1.3).

The inter-particle forces $\boldsymbol{F}_{P-P}$ can be divided into a dipole-dipole force and a film-thinning force: $\boldsymbol{F}_{P-P} = \boldsymbol{F}_{d-d} + \boldsymbol{F}_f$. Different models for the film-thinning force and the dipole-dipole force were tested against experimental data in paper A (Melheim et al., 2004). Melheim et al. (2004) concluded that the analytical dipole-dipole force expression of Davis (1964) and the film-thinning

force expression of Vinogradova (1995) gave best results. However, the analytical solution of Davis requires too much computational effort to be used in a multi-droplet system . The dipole-induced-dipole model of Siu et al. (2001) is numerically efficient and gives good results for an inter-surface spacing h larger than $0.1r_2$, where r_2 is the radius of the smallest droplet. The dipole-induced-dipole model is written as

$$F_r = \frac{12\pi\beta^2\varepsilon_{\text{oil}}|\boldsymbol{E}_0|^2 r_2^3 r_1^3}{|\boldsymbol{d}|^4}\left(3K_1\cos^2\theta - 1\right), \tag{1.9}$$

$$F_t = -\frac{12\pi\beta^2\varepsilon_{\text{oil}}|\boldsymbol{E}_0|^2 r_2^3 r_1^3}{|\boldsymbol{d}|^4}K_2\sin(2\theta), \tag{1.10}$$

where ε_{oil} is the permittivity of the oil, $\boldsymbol{d}$ is the particle-centre separation vector and β is defined as

$$\beta = \frac{\varepsilon_w - \varepsilon_{\text{oil}}}{\varepsilon_w + 2\varepsilon_{\text{oil}}}, \tag{1.11}$$

where ε_w is the permittivity of water. The coefficients K_1 and K_2 are given by

$$\begin{aligned} K_1 &= 1 + \frac{\beta r_1^3|\boldsymbol{d}|^5}{\left(|\boldsymbol{d}|^2 - r_2^2\right)^4} + \frac{\beta r_2^3|\boldsymbol{d}|^5}{\left(|\boldsymbol{d}|^2 - r_1^2\right)^4} + \frac{3\beta^2 r_1^3 r_2^3\left(3|\boldsymbol{d}|^2 - r_1^2 - r_2^2\right)}{\left(|\boldsymbol{d}|^2 - r_1^2 - r_2^2\right)^4}, \\ K_2 &= 1 + \frac{\beta r_1^3|\boldsymbol{d}|^3}{2\left(|\boldsymbol{d}|^2 - r_2^2\right)^3} + \frac{\beta r_2^3|\boldsymbol{d}|^3}{2\left(|\boldsymbol{d}|^2 - r_1^2\right)^3} + \frac{3\beta^2 r_1^3 r_2^3}{\left(|\boldsymbol{d}|^2 - r_1^2 - r_2^2\right)^3}. \end{aligned} \tag{1.12}$$

The coefficients K_1 and K_2 approach unity as $|\boldsymbol{d}| \to \infty$, and the well-known point-dipole model is recovered.

The expression for the film-thinning force of Vinogradova (1995) is written as

$$\boldsymbol{F}_{\text{f}} = -\frac{6\pi\mu a^2(\boldsymbol{V}_r \cdot \boldsymbol{e}_{\text{r}})}{h}\left\{\frac{2h}{6b}\left[\left(1 + \frac{h}{6b}\right)\ln\left(1 + \frac{6b}{h}\right) - 1\right]\right\}\boldsymbol{e}_r, \tag{1.13}$$

where $a = r_1 r_2/(r_1 + r_2)$ is the reduced radius, $\boldsymbol{V}_{\text{r}} \equiv \boldsymbol{V}_2 - \boldsymbol{V}_1$ is the relative velocity vector, and $\boldsymbol{e}_{\text{r}}$ indicates the direction of the relative motion. b can regarded as a slip factor and $b = 10^{-6}\,\text{m}$ was shown to give good results for $d_p \approx 220\,\mu\text{m}$ droplets, while $b = 10^{-7}\,\text{m}$ was used for $d_p \approx 115\,\mu\text{m}$ droplets.

Vojir and Michaelides (1994) concluded that the history force can be neglected in random velocitiy fields, e.g. turbulent flows, when there is no monotonical change in the velocity. Further, in the present work, particle flows close to walls have not been considered. The lift forces $\boldsymbol{F}_L$ and the wall forces $\boldsymbol{F}_W$ are therefore neglected.

Turbulent dispersion

Turbulent flows are irregular, seemingly random and highly diffusive. The length scales in the turbulence are limited downwards by viscosity and upwards by the geometry of the flow domain. The scales of the turbulence vary in space and time and are impossible to predict *a priori*, but it is possible to extract time-averaged mean values (Ertesvåg, 2000, Chap. 1).

By doing numerical simulations of the Reynolds-averaged Navier-Stokes equations (RANS), one calculates the mean values of the turbulent flow, e.g. the mean velocities, the mean turbulence quantities, etc. In the derivation of the RANS, it is assumed that the fluid velocity $\boldsymbol{U}$ can be split into a mean part $\overline{\boldsymbol{U}}$ and a fluctuating part $\boldsymbol{u}$ as follows:

$$\boldsymbol{U} \equiv \overline{\boldsymbol{U}} + \boldsymbol{u}, \tag{1.14}$$

where $\overline{\boldsymbol{u}} \equiv \boldsymbol{0}$. Then the turbulence appears in the RANS equations in terms of the Reynolds stresses $-\overline{u_i u_j}$, that have to be modelled. The most common modelling approaches are solving transport equations for the Reynolds stresses or employing the eddy viscosity hypothesis. The particles, however, see the instantaneous fluid velocity $\boldsymbol{U}$, for instance in the model for drag force, equation (1.5). When the mean flow is simulated by a RANS model, one therefore has to calculate the instantaneous fluid velocity $\boldsymbol{U}$ from the mean values provided by the RANS. Pope (1985, 1994, 2000) proposed stochastic differential-equation models for the instantaneous velocity based on the Langevin equation for Brownian motion. Written in its most simple form, called the simplified Langevin model (SLM), the Langevin model of Pope is given by

$$dU_i = -\frac{\partial \langle P \rangle}{\partial x_i} dt + \frac{1}{T_{L,i}} u_i dt + \sqrt{C_0 \varepsilon} dW_i(t), \tag{1.15}$$

where $\langle P \rangle$ is the mean pressure, $T_{L,i}$ is the Lagrangian time scale of the turbulence, dt is the time step, $dW_i(t)$ is the Wiener process, and C_0 is known as the Lagrangian Kolmogorov constant. The value of C_0 varies from 2.1 (Pope, 1985) to 7 (Yeung and Borgas, 2004) in the literature. The Wiener process $dW_i(t)$ is a stochastic term that is sampled from a normal distribution with zero mean and dt as the root-mean-square; $\mathcal{N}(0, dt)$.

Inertial particles do not follow the fluid particles completely, as sketched in Figure 1.2, which poses challenges to the modelling. The Lagrangian time scale has to be changed (Thomas and Oesterlé, 2005) and a mean 'crossing trajectory term' is often added to equation (1.15). In paper D (Melheim et al.,

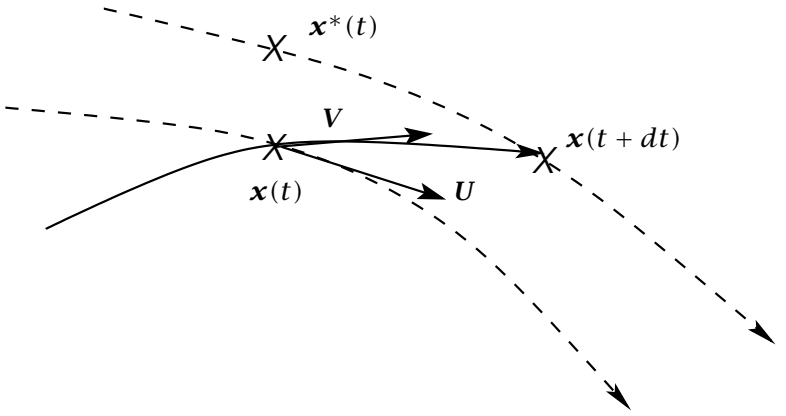


Figure 1.2: Fluid and particle paths. (From Melheim et al. (2005b)).

2005b), a slightly modified version of the expression of Minier (1999) is compared with experimental data in a horizontal mixing-layer. The mixing-layer is anisotropic and the gravity works perpendicularly to the stream-wise direction, and that makes the dispersion difficult to predict. It was shown that the model of Minier (1999) predicts the dispersion and the particle fluctuating velocities well, but a drift was seen in the mean particle velocity.

The fluid velocities seen by droplets located within the same turbulent vortex are correlated. Sommerfeld (2001) and Berlemont et al. (2001) included this effect in stochastic collision models by correlating the velocity of the fictitious collision partner with the velocity of the particle in question. Such an approach is impossible in DPM, where the collision between two particles is deterministic. The fluid velocities seen by close particles should be correlated due to the properties of the turbulence. A physically correct model should also be able to capture a *preferential concentration* for Stokes numbers around unity (Fessler et al., 1994). An attempt was made in paper E (Melheim, 2006), where the velocities were correlated via the stochastic term. To capture the variation in time and space of the scales of the turbulence, an additional stochastic differential equation for the turbulence frequency (Pope and Chen, 1990) was solved for each particle.

Algorithms

So far we have been concerned about physical issues. However, a discrete particle model is useless without appropriate algorithms and numerics. There are two topics that require special effort, the search for neighbour particles

and the number of force calculations needed per unit time in order to achieve the desired numerical accuracy.

The most convenient way to keep track of neighbour particles is the cell method (Eastwood et al., 1980). By using a uniform grid, it is easy to map the particles to the cells. Then one only has to search for neighbours in the cell where a particle is located, and in the neighbouring cells. The complexity of the search is then reduced from $O(N^2)$ to $O(N)$, where N is the number of particles.

For particles following rectilinear trajectories between collisions with an instantaneous change in velocities at collision (hard-spheres), the algorithm can be written purely *event-driven*. In an optimal event-driven simulation, the position and velocity of a particle are only calculated when a particle takes part in a collision (Lubachevsky, 1991; Sigurgeirsson et al., 2001). The efficiency of an event-driven algorithm depends on the number of particles, the number of collisions and the resolution of the grid. The number of particle cells has typically to be set before the computation starts. In paper B (Melheim and Gjelsvik, 2005), an adaptive cell structure is proposed, that adapts to the local number density of particles and thereby ensures an efficient computation.

In most particle-laden flows, there is a considerable interaction between the fluid and the particles, and the particles do not move along straight lines. The drag force requires time-steps in the order of the particle response time τ_p and the interactions between particles in a very viscous fluid cannot be handled as an instantaneous change in velocity. Small time steps are therefore needed, and a more sophisticated integrator than the forward Euler scheme, which is used in event-driven simulations, should be employed. Because of the small time steps needed, for instance for computation of the collision forces in soft-sphere collision models (Tsuji et al., 1993), the algorithm is *time-driven*. Time-driven algorithms are generally slow compared to event-driven billiards algorithms.

An algorithm that tries to include the advantages of both event-driven and time-driven algorithms is proposed in paper C (Melheim, 2005) and is called the cluster integration method (CIM). In CIM, the particles are equipped with a virtual radius, and close-particle lists are made of particles with overlapping virtual radii. A 'test move' with the forward Euler scheme, following the patterns of an event-driven algorithm, is performed in order to identify potential collision partners in the direction of motion. Potential collision partners are added to each other's close-particle lists. Clusters are then made from the close-particle lists and the different clusters are integrated separately using a variable step-size Runge-Kutta scheme (Hairer et al., 1992, Chap. II.4). Hence, the single-particle clusters and clusters with weak interactions between the

particles can be integrated with a much longer time step than those clusters with strong particle-particle interactions. CIM was reported to give a significant speed-up compared to a traditional time-driven algorithm.

Numerical methods

The equations describing the motion of the particles are ordinary differential equations (ODEs), while the equations for the fluctuating fluid velocities and turbulent frequency are stochastic differential equations (SDEs). There are two numerical approaches to solve an ODE; multi-step methods and one-step methods (Hairer et al., 1992), where the simplest version of both approaches is the forward Euler scheme. Furthermore, a third approach is seen in the literature, a quasi-analytical solution, e.g. in Coimbra et al. (1998).

The Runge-Kutta schemes belong to the one-step methods, and there are several reasons that make them attractive for evolving particles in time:

- The Runge-Kutta schemes are easy to implement, there is no starting difficulties and at the end of a time step one has one value.
- The Runge-Kutta schemes are well studied, one can choose a scheme based on desired properties.
- It is easy to employ variable step size, so-called embedded Runge-Kutta methods.
- By using Runge-Kutta schemes, one has to separate the numerics and the physics, which is good for code maintenance.

The model for the turbulent dispersion is given by a stochastic differential equation for the fluctuating velocities, that contains both a dt term and a stochastic Wiener process dW. The Runge-Kutta schemes for ordinary differential equations cannot in general be used directly on SDEs because they do not approximate an Ito-Taylor expansion of an SDE (Kloeden and Platen, 1992). The forward Euler scheme approximates the first deterministic and stochastic term of an Ito-Taylor expansion and can therefore be used, but it is inaccurate. Burrage and Burrage (1996) made an explicit Runge-Kutta scheme for SDEs, where the classical 4th order Runge-Kutta scheme is recovered when the stochastic terms are neglected. Hence, the scheme of Burrage and Burrage (1996) can be used for the fluctuating fluid velocity together with the classical RK4 scheme for the particle position and velocity.

A variable step-size Runge-Kutta scheme also exists for SDE (Burrage and Burrage, 2002), but such schemes are complex. It is assumed that the stiffness of the velocity equation (1.3) decides the time-step and a variable step-size scheme can therefore be applied to the position and the velocity only. The fluctuating velocities, found by the SDE, are updated at the end of every successful local time-step dt_l with a fixed step-size SDE scheme. Minier and Peirano (2001) stated that the time-step for calculation of the fluctuating velocity should be much larger than the Kolmogorov time scale τ_η and much less than the integral time scale T_L. The upper bound is satisfied by limiting the global time step.

1.5 Fluid-flow equations

Due to the presence of particles, the Reynolds-averaging strategy cannot be employed on the Navier-Stokes equations. However, the averaged continuity and momentum equations appearing from volume averaging (Nigmatulin, 1991) or ensemble averaging (Drew and Passman, 1999) are, except for the particle source terms and the volume fraction, equal to the Reynolds-averaged equations. Hence, they are frequently called the Reynolds-averaged Navier-Stokes equations (RANS). The averaged Navier-Stokes equations for the continuous phase in a dispersed particle flow can be written as

$$\frac{\partial}{\partial t}\left(\alpha\rho\right) + \frac{\partial}{\partial x_i}\left(\alpha\rho\overline{U}_i\right) = 0, \tag{1.16}$$

$$\frac{\partial}{\partial t}\left(\alpha\rho\overline{U}_i\right) + \frac{\partial}{\partial x_j}\left(\alpha\rho\overline{U}_i\overline{U}_j\right) = -\alpha\frac{\partial\overline{P}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(-\alpha\rho\overline{u_i u_j}\right) + \frac{\partial}{\partial x_j}\left(\alpha\mu\left(\frac{\partial\overline{U}_i}{\partial x_j} + \frac{\partial\overline{U}_j}{\partial x_i}\right)\right) + \mathcal{F}_p, \tag{1.17}$$

where $\overline{P}$ is the mean pressure and $\mathcal{F}_p$ is the mean force of the particles working on the fluid. Mass transfer is neglected. The momentum equation (1.17) is in the present work closed by the turbulent viscosity approach, where the Reynolds stresses $-\overline{u_i u_j}$ are expressed by (Pope, 2000, Chap. 4)

$$-\rho\overline{u_i u_j} = \mu_t\left(\frac{\partial\overline{U}_i}{\partial x_j} + \frac{\partial\overline{U}_j}{\partial x_i}\right) - \rho\frac{2}{3}k\delta_{ij}, \tag{1.18}$$

where k is the turbulent kinetic energy ($k = \frac{1}{2}\overline{u_i u_i}$) and μ_t is the turbulent viscosity, modelled by a slightly modified version of the standard k-ε-model

(Launder and Spalding, 1974). Launder & Spalding modelled the turbulent viscosity as follows:

$$\mu_t = C_\mu \rho \frac{k^2}{\varepsilon}, \tag{1.19}$$

where $C_\mu = 0.09$ is a model constant and ε is the dissipation rate of turbulent kinetic energy. Transport equations are solved for the turbulent scalars k and ε, wherein the volume fraction α is included. The transport equation for the turbulent kinetic energy k reads:

$$\frac{\partial}{\partial t}(\alpha\rho k) + \frac{\partial}{\partial x_j}\left(\alpha\rho k\overline{U}_j\right) = \frac{\partial}{\partial x_j}\left(\alpha\left(\mu + \mu_t\right)\frac{\partial k}{\partial x_j}\right) + \alpha\rho P_k - \alpha\rho\varepsilon + S_k, \tag{1.20}$$

where the production of turbulent kinetic energy P_k is defined by

$$\rho P_k = \mu_t\left(\frac{\partial \overline{U}_i}{\partial x_j} + \frac{\partial \overline{U}_j}{\partial x_i}\right)\frac{\partial \overline{U}_i}{\partial x_j}. \tag{1.21}$$

S_k accounts for the influence of the particles on the turbulent kinetic energy. The transport equation for the dissipation rate of turbulent kinetic energy ε is given by:

$$\frac{\partial}{\partial t}(\alpha\rho\varepsilon) + \frac{\partial}{\partial x_j}\left(\alpha\rho\varepsilon\overline{U}_j\right) = \frac{\partial}{\partial x_j}\left(\alpha\left(\mu + \frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right) + C_{\varepsilon 1}\alpha\frac{\varepsilon}{k}\rho P_k - C_{\varepsilon 2}\alpha\rho\frac{\varepsilon}{k}\varepsilon + S_\varepsilon, \tag{1.22}$$

where $\sigma_\varepsilon = 1.3$, $C_{\varepsilon 1} = 1.44$, $C_{\varepsilon 2} = 1.92$, and S_ε is a source term due to the particles.

Inter-phase transfers

A criterion for the inter-phase transfers from the particle phase to the fluid phase based on the macroscopic momentum-coupling parameter is given by (Loth, 2000):

$$\Pi_L = \frac{\alpha_p L^3 \left|\rho_p - \rho\right| g}{\rho V_L^2 L^2}, \tag{1.23}$$

$$\Pi_\Lambda = \frac{\alpha_p \Lambda^3 \left|\rho_p - \rho\right| g}{\rho V_\Lambda^2 \Lambda^2}, \tag{1.24}$$

where L equals the largest length scale in the flow and V_L is the largest velocity scale. Λ and V_Λ denote the integral scales. The coupling is, according to Loth (2000), negligible for the mean flow when $\Pi_L \ll 1$. The source terms in the k and ε equations can be neglected when $\Pi_\Lambda \ll 1$. A low volume fraction of water droplets ($\alpha_p < 10\%$) in oil and a moderate turbulence level $Re_\lambda \approx 25$ typically gives $\Pi_L < 1$ and $\Pi_\lambda \ll 1$, which means that the droplets might have an influence on the mean flow, but not on the turbulence. Hence, the source terms S_k and S_ε can be neglected in the present work.

Solution methods

The momentum equation (1.17) and the transport equations related to the turbulence models, equations (1.20) and (1.22), are discretised using the finite-volume method on a grid. Incompressible flow is assumed and the pressure field is found by the SIMPLE algorithm (Patankar, 1980, Chap. 6). The governing equations for the fluid motion can be solved both for transient and steady-state flows. A transient solution requires a volume-averaging of the particle source terms over each control volume. To gain stability, the drag force is split into a term dependent on the mean velocity at the centre of the control volume and an independent term. Steady simulations are more stable, but require that time-averaged values can be extracted from a large number of particle trajectories. An iteration process is performed until convergence: The single phase fluid flow is calculated, then a large number of particles are tracked and the mean force on the fluid in each control volume is calculated. Then the flow is calculated again with updated source terms. The iteration is performed until the number of iterations the fluid solver needs to converge reached a 'steady number' (Kohnen et al., 1994).

1.6 Present contribution

The main contribution of the present work is given in six articles, published in or submitted to the scientific community.

Together, the contents of articles A–E fill the gap between a general Eulerian-Lagrangian particle code and a code that can perform numerical simulation of turbulent electrocoalescence. Figure 1.3 shows how the gap is filled: the particle-pair forces, an adaptive cell structure and a model for turbulent dispersion were implemented in the basic code. The cluster integration method uses the cell structure and speeds up the computation of the inter-particle forces. The model for correlating the motion of close particles requires a

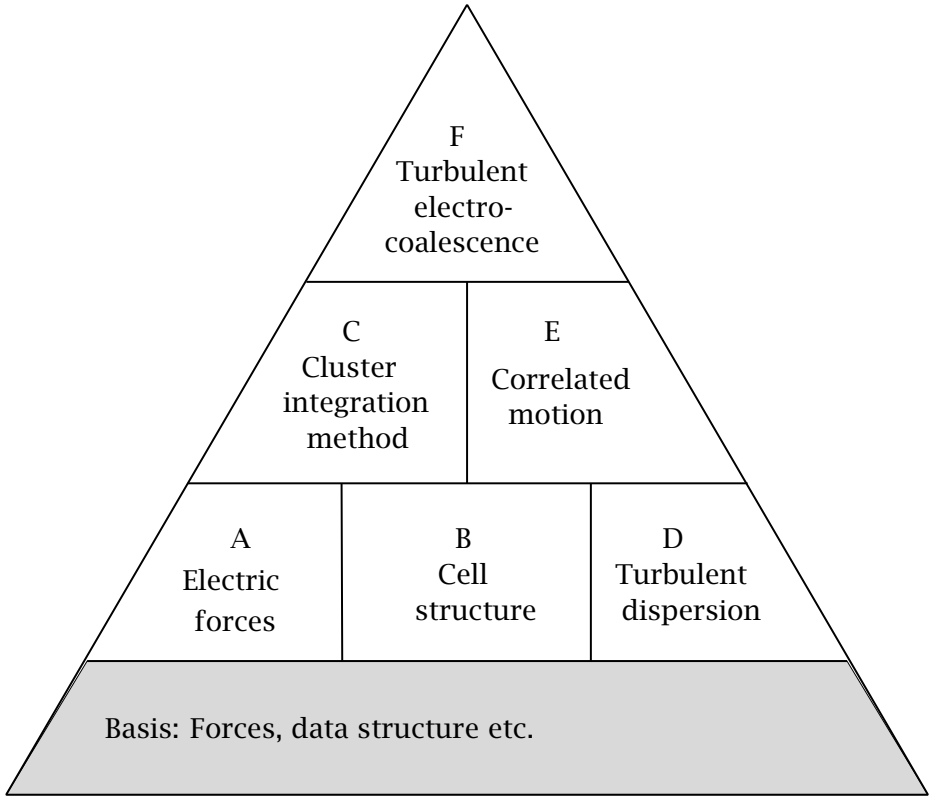


Figure 1.3: Map of articles.

cell structure and a turbulent dispersion model for each single particle. Paper F (Melheim and Chiesa, 2006) on turbulent electrocoalescence concludes the work and shows that turbulent electrocoalescence can be computed numerically by using a discrete particle method for the droplets and Reynolds-averaged Navier-Stokes equations for the fluid phase.

As is apparent from the summaries of the articles that follow, the articles themselves have a wider audience than those interested in numerical simulations of turbulent electrocoalescence. Everyone interested in Lagrangian particle simulations, in particular those working with deterministic particle-particle interactions, will hopefully find something interesting and useful.

Paper A – Forces between two water droplets in oil under the influence of an electric field

Co-authors: Matteo Chiesa, Stian Ingebrigtsen, and Gunnar Berg

The paper outlines the forces that influence the kinematics of a droplet falling towards a stationary droplet exposed to an electric field. Experiments and numerical simulations were performed. The experimental setup consisted of two vertically placed electrodes. One droplet was at rest at the lower electrode and a smaller droplet was released some millimetres above. The droplet motion was recorded with a high-speed CMOS camera. The optical observations were compared with the results from numerical simulations where the governing equations for the droplet motion were solved by the RK45 Fehlberg method with step-size control and low tolerances. It was found that the droplet motion mainly was due to buoyancy, drag, film-drainage, and dipole-dipole forces. Attention was paid to internal circulations, non-ideal dipoles, and the effects of surface tension gradients. The best results were obtained by using the drag-force model of LeVan (1981), the analytical dipole-dipole force expression of Davis (1964) and the film-thinning force model of Vinogradova (1995). It should be mentioned that a better physical interpretation of the *slip-factor* in the film-thinning model was given in Chiesa et al. (2005b). The analytical expression of Davis contains numerical solution of complex series, which is computationally heavy, and the expression is valid for two droplets only. Therefore, the dipole-induced-dipole model of Siu et al. (2001) was recommended for multi-droplet systems.

Paper B – Adaptive cell structure for efficient detection of neighbouring particles in collisional particle-laden flows

Co-author: Anders Gjelsvik

An adaptive algorithm for particle-particle and particle-wall collision detection in the two-dimensional case was presented. The algorithm ensures efficient computations of colliding particle flows without any input from the user regarding the particle cell structure. The physical domain was hierarchically divided and structured as a quadtree. The algorithm was intended for particle-laden flows, which require small time steps, but within each time step, the algorithm is event-driven. Tests with uniform particle distribution and fixed cell structures with uniform cell sizes showed that the number of particles inside a particle cell should be kept in the range of 0.25 to 3.0 by the adaptive algorithm. Test cases with non-uniform particle distributions were

performed and showed that the proposed adaptive cell structure is not more efficient than an optimal uniform cell structure. The advantages are therefore the user-friendliness and the time not used to search for the optimal uniform cell structure, and that is difficult to measure.

Paper C – Cluster integration method in Lagrangian particle dynamics

An efficient and robust approach for numerical simulations of collisional particle dynamics in the Lagrangian framework was proposed. The algorithm handles efficiently the interactions between particles and droplets; clusters of interacting droplets are assembled and integrated separately, using a variable step-size Runge-Kutta solver. The inter-particle interactions can be both contact forces, hydrodynamic forces, and important for electro-coalescence, the electric dipole-dipole forces. The cluster integration method was applied on sedimentation of 5000 particles in a two-dimensional box. A significant speed-up was achieved. Compared to a traditional discrete element method with the forward Euler scheme, a speed-up factor of three orders of magnitude in a dilute regime and two orders of magnitude in a dense regime were observed.

Paper D – Modeling of the vortex-structure in a particle-laden mixing-layer

Co-authors: Stefan Horender and Martin Sommerfeld

Numerical calculations of a particle-laden turbulent horizontal mixing-layer based on the Eulerian-Lagrangian approach were presented in this paper. This test case was chosen because Dr Stefan Horender did his doctoral work on it. It is a challenging test case; the turbulence is anisotropic and the gravity causes a drift in the span-wise direction. A slightly modified version of the Langevin model of Minier (1999) was used to determine the fluctuating fluid velocity seen by the particles. The Reynolds-averaged Navier-Stokes equations for the fluid phase were closed by the standard k-ε-model. The numerical results were compared with particle image velocimetry (PIV) measurements. The concentration profile and the particle velocity fluctuations were predicted reasonably well, but a drift was seen in the mean particle velocity. Further studies are therefore needed.

Paper E – Correlated motion of inertial particles in turbulent flows

The instantaneous fluid velocity is modelled by evolving in time a stochastic differential equations for the fluctuating fluid velocity, as described in Paper D (Melheim et al., 2005b). In this paper (Paper E), a model for correlating the motion of close particles in turbulent flows was proposed. The model correlated the fluid velocity *seen* by the particles via the random term in the stochastic equation for the fluctuating fluid velocity, using the spatial auto-correlation functions. The intermittency of the turbulence was modelled by employing an additional stochastic differential equation for the turbulence frequency. Simulations of fluid-particle pair dispersion showed that the results followed the classical Richardson t^3-law. N inertial particles in isotropic and homogeneous turbulence were simulated and the results were compared with empirical correlations based on direct numerical simulations. A deviation was seen at low Stokes number, but the gap decreased when more particles, that means particles further away from the particle in question, were included in the correlation calculations. Both the DNS and the presented calculations allowed particle overlaps, hence too much weight might be given to viscous effects, which were neglected in the proposed correlation model. A comparison with DNS-data with real particle-particle collisions should be made before conclusions can be drawn.

Paper F – Simulation of turbulent electrocoalescence

Co-author: Matteo Chiesa

Eulerian-Lagrangian simulations of water droplets in a turbulent oil flow influenced by electric fields were presented. A two-dimensional channel (3 cm $\times$ 6 cm), downstream of a perforated plate was considered, where the electric fields were directed perpendicularly to the stream-wise direction. In these simulations, we used the models and algorithms that were presented in the previous papers. The results show that the mutual interaction of turbulence and electric fields strongly enhances the droplet growth. A 0.2 s treatment increases the droplet diameter with a factor three, which corresponds to a ten-fold higher terminal velocity. The results agree qualitatively well with experimental observations on a macroscopic scale. Quantitative comparisons, however, require a three-dimensional code and large computational resources.

1.7 Conclusions and further work

The papers A–F deal with different topics, and they have their separate conclusions. In this section, a main conclusion from each paper is drawn. Suggestions for the direction of further work on computer simulation of turbulent electrocoalescence are also given.

Conclusions

Paper A. The point-particle approach with appropriate models for the inter-droplet forces can be used to calculate electrocoalescence.

Paper B. The advantage of the proposed adaptive cell structure is its user-friendliness rather than computational savings.

Paper C. The proposed algorithm, called cluster integration method, is efficient for a wide range of particle-dynamics applications.

Paper D. The Langevin-equation approach for the fluctuating fluid velocity 'seen' by the particles provides a reasonable prediction of the dispersion of particles in anisotropic turbulence.

Paper E. The proposed model for correlating the fluid velocities felt by close particles gave better results than existing N-particle models in the literature.

Paper F. The results from numerical simulations of turbulent electrocoalescence in two dimensions showed the same trends as experimental results in the literature.

Further work

- The code should be extended to three dimensions to get more reliable results. A parallelisation of the code might then be necessary.
- A criterion for the coalescence probability is needed.
- Large-eddy simulations of the fluid phase may improve the results.
- Laboratory experiments on turbulent electrocoalescence should be carried out for validation of the numerical models.

Bibliography

Allen, M. P., Tildesley, D. J., 1987. Computer Simulation of Liquids. Oxford University Press, Oxford, ISBN 0-19-855645-4.

Anandarajah, A., 1999. Multiple time-stepping scheme for discrete element analysis of colloidal particles. Powder Technol. 106, 132–141.

Appenzeller, T., June 2004. The end of cheap oil. National Geographic.

Atten, P., 1993. Electrocoalescence of water droplets in an insulating liquid. J. Electrostat. 30, 259–270.

Berlemont, A., Achim, P., Chang, Z., 2001. Lagrangian approaches for particle collisions: The colliding particle velocity correlation in multiple particle tracking method and in the stochastic approach. Phys. Fluids 13 (10).

Burrage, K., Burrage, P. M., 1996. High strong order explicit Runge-Kutta methods for stochastic ordinary differential equations. Appl. Numer. Math. 22, 81–101.

Burrage, P. M., Burrage, K., 2002. A variable stepsize implementation for stochastic differential equations. Siam J. Sci. Comput 24 (3), 848–864.

Chiesa, M., Mathiesen, V., Melheim, J. A., Halvorsen, B., 2005a. Numerical simulation of particulate flow by the Eulerian-Lagrangian and Eulerian-Eulerian approach with application to fluidised bed. Comput. Chem. Eng. 29, 291–304.

Chiesa, M., Melheim, J. A., Pedersen, P. A., Ingebrigtsen, S., Berg, G., 2005b. Forces acting on water droplets falling in oil under the influence of an electric field: numerical predictions versus experimental observations. Eur. J. Mech. B-Fluid 24, 717–732.

Chiesa, M., Olsen, R., Melheim, J. A., 2005c. Euler/Lagrange simulations of a two-dimensional fluidised bed. In: CFD 2005, 4th International conference on Computational Fluid Dynamics in the Oil and Gas, Metallurgical & Process Industries. Trondheim, Norway.

Coimbra, C. F. M., Shirolkar, J. S., Queiroz MeQuay, M., 1998. Modeling particle dispersion in a turbulent multiphase mixing layer. J. Wind Eng. Ind. Aerod. 73, 79–97.

Crowe, C., Sommerfeld, M., Tsuji, Y., 1998. Multiphase flows with droplets and particles. CRC Press, Florida, ISBN 0-8493-9469-4.

Cundall, P. A., Strack, O. D. L., 1979. A discrete numerical model for granular assemblies. Geotechnique 29 (1), 47–65.

Davis, M., 1964. Two charged spherical conductors in a uniform electric field: Forces and field strength. Rand. Corp. Memorandum RM-3860-PR.

Drew, D. A., Passman, S. L., 1999. Theory of Multicomponent Fluids. No. 135 in Applied Mathematical Sciences. Springer-Verlag, New York, ISBN 0-387-98380-5.

Eastwood, J. W., Hochney, R. W., Lawrence, D. N., 1980. P3M3DP – The three dimensional periodic particle-particle/particle-mesh program. Comput. Phys. Commun. 19, 215–261.

Eow, J. S., Ghadiri, M., Sharif, A. O., Williams, T. J., 2001. Electrostatic enhancement of coalescence of water droplets in oil: A review of current understanding. Chem. Eng. J. 84, 173–192.

Ergun, S., 1952. Fluid flow through packed columns. Chem. Eng. Prog. 48, 89.

Ertesvåg, I. S., 2000. Turbulent strøyming og forbrenning. Tapir akademisk forlag, Trondheim, ISBN 82-519-1568-6. In Norwegian.

Fessler, J. R., Kulick, J. D., Eaton, J. K., 1994. Preferential concentration of heavy particles in a turbulent channel flow. Phys. Fluids 6 (11).

Friedemann, J. D., Nilsen, P. J., Sæther, O., Sjöblom, J., 2001. An alternative method for establishing parameters for novel coalescer designs. In: 4th International Symposium on the Chemistry and Physics of Petroleum–Water Emulsions II.

Gidaspow, D., 1994. Multiphase flow and fluidization: Continuum and kinetic theory description. Academic Press, San Diego, ISBN 0-12-282470-9.

Hairer, E., Nørsett, S., Wanner, G., 1992. Solving Ordinary Differential Equations I, 2nd Edition. Springer, Berlin, ISBN 3-540-56670-8.

Hansen, E. B., 2005. Numerical simulation of droplet dynamics in the presence of an electric field. Ph.D. thesis, Department of Energy and Process Engineering, Norwegian Univeristy of Techology and Science (NTNU), Trondheim, Norway.

Harpur, I. G., Wayth, N. J., Bailey, A. G., Williams, T. J., Urdahl, O., 1997. Destabilisation of water-in-oil emulsions under the influence of an A.C. electric field: Experimental assessment of performance. J. Electrostat. 40 & 41, 135–140.

Hoomans, B. P. B., Kuipers, J. A. M., Briels, W. J., van Swaaij, W. P. M., Jan. 1996. Discrete particle simulation of bubble and slug formation in a two-dimensional gas-fluidised bed: A hard-sphere approach. Chem. Eng. Sci. 51 (1), 99–118.

Horender, S., Melheim, J. A., Sommerfeld, M., 2005. Modelling non-isotropic particle dispersion in a shear layer. In: 11th Workshop on Two-Phase Flow Predictions. Merseburg, Germany.

Klingenberg, D. J., Frank van Swol, Zukoski, C. F., 1991. The small shear rate response of electrorheological suspensions. II. Extensions beyond the point-dipole limit. J. Chem. Phys. 94 (9), 6170–6178.

Kloeden, P. E., Platen, E., 1992. Numerical Solution of Stochastic Differential Equations. Springer-Verlag, Berlin, ISBN 3-540-540062-8.

Kohnen, G., Rüger, M., Sommerfeld, M., 1994. Convergence behaviour for numerical calculations by the Euler/Lagrange method for strongly coupled phases. In: Numerical Methods in Multiphase Flow. Vol. 185 of ASME FED. pp. 191–202.

Launder, B. E., Spalding, D. B., 1974. The numerical computation of turbulent flow. Comput. Method. Appl. M. 3, 269–289.

Laviéville, J., Deutsch, E., Simonin, O., 1995. Large eddy simulations of interactions between colliding particles and a homogeneous isotropic turbulence field. In: Gas-Particle Flows. Vol. 228 of ASME FED.

LeVan, D. M., 1981. Motion of droples with a Newtonian interface. J. Colloid Interf. Sci. 83 (1), 11–17.

Loth, E., 2000. Numerical approaches for motion of disperesed particles, droplets and bubbles. Prog. Energ. Combust. 26, 161–223.

Lubachevsky, B. D., 1991. How to simulate billiards and similar systems. J. Comput. Phys. 94, 255–283.

Lukkarinen, A., Kaski, K., Feb. 1998. Simulation studied of electrorheological fluids under shear, compression, and elongation loading. J. Appl. Phys. 83 (3), 1717–1725.

Lundgaard, L. E., Berg, G., Ingebrigtsen, I., Atten, P., 2005. Electrocoalescence in oil-water separation: Fundamental aspects. In: Sjöblom, J. (Ed.), Emulsions and Emulsion Stability, 2nd Edition. CRC Press, ISBN 978-0824726959.

Melheim, J. A., 2005. Cluster integration method in Lagrangian particle dynamics. Comput. Phys. Commun 171, 155–161.

Melheim, J. A., 2006. Correlated motion of inertial particles in turbulent flows. Submitted to Eur. J. Mech. B Fluid.

Melheim, J. A., Chiesa, M., 2006. Simulation of turbulent electrocoalescene. Chem. Eng. Sci. 61, 4540–4549.

Melheim, J. A., Chiesa, M., Gjelsvik, A., 2005a. Formulation of numerical performance of an adaptive algorithm for efficient collision detection. In: 2005 ASME FEDSM, Algorithmic Developments in CFD. Houston, Texas, USA.

Melheim, J. A., Chiesa, M., Ingebrigtsen, S., Berg, G., 2004. Forces between two water droplets in oil under the influcence of an electric field. In: 5th International Conference on Multiphase Flow. Yokohama, Japan, paper # 126.

Melheim, J. A., Gjelsvik, A., 2005. Adaptive cell structure for efficient detection of neighbouring particles in collisional particle-laden flows. Submitted to Comput. Fluids.

Melheim, J. A., Gran, I. R., 2005. Cluster integration method – an efficient and flexible algorithm for DEM simulations. In: 11th Workshop on Two-Phase Flow Predictions. Merseburg, Germany.

Melheim, J. A., Horender, S., Sommerfeld, M., 2005b. Modeling of the vortex-structure in a particle-laden mixing-layer. In: 2005 ASME FEDSM, Gas-Particle Flows. Houston, Texas, USA.

Michaelides, E. E., 2003. Hydrodynamic force and heat/mass transfer from particles, bubbles, and drops - the Freeman scholar lecture. J. Fluid. Eng.-T. ASME 125, 209–238.

Minier, J.-P., 1999. Closure proposals for the Langevin equation model in Lagrangian two-phase flow modelling. In: Proceedings of the 3rd ASME/JSME Joint Fluids Engineering Conference. San Francisco, California, USA.

Minier, J.-P., Peirano, E., 2001. The pdf approach to turbulent polydispersed two-phase flows. Phys. Rep. 352, 1–214.

Munkejord, S. T., Mølnvik, M. J., Melheim, J. A., Gran, I. R., Olsen, R., 2005. Prediction of two-phase pipe flow using simple closure relations in a 2d two-fluid model. In: CFD 2005, 4th International conference on Computational Fluid Dynamics in the Oil and Gas, Metallurgical & Process Industries. Trondheim, Norway.

Nigmatulin, R. I., 1991. Dynamics of Multiphase Media. Vol. 1. Hemisphere Publishing Corporation, New York, ISBN 0-89116-316-6.

Osher, S., Fedkiw, R. P., 2001. Level set methods: An overview and some recent results. J. Comput. Phys. 169, 463–502.

Patankar, N., Joseph, D., 2001. Modeling and numerical simulation of particulate flows by Eulerian-Lagrangian approach. Int. J. Multiphase Flow 27, 1659–1684.

Patankar, S. V., 1980. Numerical heat transfer and fluid flow. McGraw-Hill, New York, ISBN 0-07-048740-5.

Pope, S. B., 1985. Pdf methods for turbulent reactive flows. Prog. Energ. Combust. 11, 119–192.

Pope, S. B., 1994. Lagrangian pdf methods for turbulent flows. Annu. Rev. Fluid Mech. 26, 23–63.

Pope, S. B., 2000. Turbulent Flows. Cambridge University Press, Cambridge, United Kingdom, ISBN 0 521 59886.

Pope, S. B., Chen, Y. L., 1990. The velocity-dissipation probability density function model for turbulent flows. Phys. Fluids 2 (8), 1437–1449.

Saffman, P. G., Turner, J. S., 1956. On the collision of drops in turbulent clouds. J. Fluid Mech. 1, 16–30.

Scardovelli, R., Zaleski, S., 1999. Direct numerical simulations of free-surface and interfacial flow. Annu. Rev. Fluid Mech. 31, 567–603.

Sigurgeirsson, H., Stuart, A., Wan, W.-L., 2001. Algorithms for particle-field simulations with collisions. J. Comput. Phys. 172, 766–807.

Simonin, O., Deutsch, E., Minier, J.-P., 1993. Eulerian prediction of the fluid/particle corrolated motion in turbulent two-phase flows. Appl. Sci. Res. 53, 275–283.

Siu, Y. L., Jones, T. W., Yu, K., 2001. Interparticle force in polydisperse electrorheological fluid: Beyond the dipole approximation. Comput. Phys. Commun. 142, 446–452.

Sommerfeld, M., 2001. Validation of a stochastic Lagrangian modelling approach for inter-particle collisions in homogeneous isotropic turbulence. Int. J. Multiphase Flow 27, 1829–1858.

Storting proposition no 38 (2003-2004), 2004. In Norwegian.
URL `http://odin.dep.no/repub/03-04/stmeld/38/index.html`

Thomas, L., Oesterlé, B., 2005. An investigation of crossing trajectory effects in turbulent shear flow. In: 2005 ASME FEDSM, Gas-Particle Flows. Houston, Texas, USA.

Tryggvason, G., Bunner, B., Esmaeeli, A., Juirc, D., Al-Rawahi, N., Tauber, W., Han, J., Nas, S., Jan, Y.-J., 2001. A front-tracking method for the computations of multiphase flow. J. Comput. Phys. 169, 708–759.

Tsuji, Y., Kawaguchi, T., Tanaka, T., 1993. Discrete particle simulation of two-dimensional fluidized bed. Powder Technol. 77 (1), 79–87.

Urdahl, O., Wayth, N. J., Førdedal, H., Williams, T. J., Bailey, A. G., 2001. Compact electrostatic coalescer technology. In: Encyclpedic Handbook of Emulsion Techology. Marcel Dekker, New York, pp. 679–694, ISBN-0824704541.

Vinogradova, O. I., 1995. Drainage of a thin liquid film confined between hydrophobic surfaces. Langmuir 11, 2213–2220.

Vojir, D. J., Michaelides, E. E., 1994. Effect of the history term on the motion of rigid spheres in a viscous fluid. Int. J. Multiphase Flow 20 (3), 547–556.

Wang, L.-P., Wexler, A. S., Zhou, Y., 2000. Statistical mechanical description and modelling of turbulent collision of inertial particles. J. Fluid Mech. 415, 117–153.

Xu, B. H., Yu, A. B., 1997. Numerical simulation of the gas-solid flow in a fluidized bed by combining discrete particle method with computational fluid dynamics. Chem. Eng. Sci. 52 (16), 2785–2809.

Yeung, P. K., Borgas, M. S., 2004. Relative dispersion in isotropic turbulence. Part 1. Direct numerical simulations and Reynolds-number dependence. J. Fluid Mech. 503, 93–124.

Zuccaro, G., Lapenta, G., Maizza, G., 2004. Particle in cell simulations of combustion synthesis of TiC nanoparticles. Comput. Phys. Commun. 162, 89–101.

Paper A

Forces between two water droplets in oil under the influence of an electric field

Melheim J. A., Chiesa, M., Ingebrigtsen, S. and Berg G.,
In 5th International Conference on Multiphase Flow
ICMF'04, Yokohama, Japan., June 2004

5th International Conference on Multiphase Flow, ICMF'04
Yokohama, Japan, May 30-June 4, 2004
Paper No.126

Forces between two water droplets in oil under the influence of an electric field

Jens A. Melheim[1], **Matteo Chiesa**[2], **Stian Ingebrigtsen**[3], **Gunnar Berg**[4]

1: Dept. of Energy and Process Engineering, NTNU , Norway, jensm@pvv.ntnu.no
2: Dept. of Energy Processes, SINTEF Energy Research, Norway, matteo.chiesa@sintef.no
3: Dept. of Electric Power Technology, SINTEF Energy Research, Norway, stian.ingebrigtsen@sintef.no
4: Dept. of Electric Power Technology, SINTEF Energy Research, Norway, gunnar.berg@sintef.no

Abstract The combination of an electric field and a moderate turbulent flow is a promising technique for separating stabile water-oil emulsions. Charges induced on the water droplets will cause adjacent droplets to align with the field and attract each other. The present work outlines the forces that influence the kinematics of a droplet falling towards a stationary droplet. Mathematical models for these forces are presented and discussed with respect to the implementation in a multi-droplet Lagrangian framework. The droplet motion is mainly due to buoyancy, drag, film-drainage, and dipole-dipole forces. Attention is paid to internal circulations, non-ideal dipoles, and the effects of surface tension gradients.

Experiments are performed to observe the behavior of the falling droplet exposed to an electrical field. The experimental setup consists of two vertically placed electrodes. One droplet is at rest at the lower electrode and a smaller droplet is released some millimeters above. The droplet motion is recorded with a high-speed CMOS camera. The optical observations are compared with the results from numerical simulations where the governing equations for the droplet motion are solved by the RK45 Fehlberg method with step size control and low tolerances. Results, using different models, are compared and discussed in details. Furthermore, a strategy that allows to properly describe both the kinematics of a falling rigid sphere particle and of a fluid droplet under the influence of an electric field is outlined.

Nomenclature

A, A_d	Particle surface	r_2	Radius of stationary particle
a	Reduced radius	Re_d	Particle Reynolds number
b	Slip length, see Fig. 1	$\boldsymbol{u}$	Continuous phase velocity vector
C_d	Drag coefficient	$\boldsymbol{v}$	Particle velocity vector
$\boldsymbol{d}$	Separation vector	$\boldsymbol{v}_i$	Velocity vector of particle i
$\boldsymbol{E}$,$\boldsymbol{E}_0$,	Electric field vector	$\boldsymbol{v}_\mathrm{r}$	Relative particle velocity vector
$\boldsymbol{e}_\mathrm{r}$	Relative motion vector	V_d	Particle volume
$\boldsymbol{F}_\mathrm{e}$	Dielectrophoretic force vector	$\boldsymbol{x}_\mathrm{i}$	Particle position vector
$\boldsymbol{F}_\mathrm{b}$	Body force vector	γ_1	Magnitude of the surface tension gradient
$\boldsymbol{F}_\mathrm{f}$	Film thinning force vector	ε_0	Vacuum dielectric constant
$\boldsymbol{F}_\mathrm{d\text{-}d}$	Inter-droplet force vector	ε_d	Relative dielectric constant of droplet/particle
$\boldsymbol{F}_\mathrm{d}$	Drag force vector	ε_oil	Relative dielectric constant of oil
$\boldsymbol{F}_\mathrm{ext}$	External-droplet force vector	θ	Angle between $\boldsymbol{E}$ and $\boldsymbol{d}$
$\boldsymbol{F}_\mathrm{fluid}$	Fluid-droplet force vector	λ	Viscosity ratio ($\lambda = \mu_d/\mu_c$)
h	Least distance between two particles	μ_c	Viscosity of oil
i	Particle i	μ_d	Viscosity of droplet
r_d	Particle radius	ρ_c	Density of oil
r_1	Radius of falling particle	ρ_d	Density of droplet/particle

1 Introduction

The oil extracted from offshore reservoirs will normally contain a large and, during the reservoir lifetime, increasing percentage of water emulsified in the oil. Electric fields are used to extract the water from the crude oil [1] . The combination of an electric field and a moderate turbulent flow is a promising and compact technique for separating stabile water-oil emulsions [2]. Charges induced on the water droplets will cause adjacent droplets to align with the field, attract each other and eventually coalesce. The sedimentation velocity increases proportionally to the square of the diameter, and therefore one wishes to get the smallest water droplets to coalesce together or into larger droplets. The present work outlines the forces that influence the kinematics of a spheric droplet falling towards a stationary one. Mathematical models for these forces are presented and discussed with respect to the implementation in a multi-droplet Lagrangian framework. The spherical droplet motion is mainly due to buoyancy, drag, film-drainage, and dipole-dipole forces. General and physically meaningful models for these forces are needed. The trajectory of a spherical droplet i is calculated by integrating Newton's second law. The law equates the droplet inertia with the forces acting on it, and reads:

$$\frac{d\boldsymbol{x}_i}{dt} = \boldsymbol{v}_i \tag{1}$$

$$m_i \frac{d\boldsymbol{v}_i}{dt} = \boldsymbol{F}_{\text{fluid}} + \boldsymbol{F}_{\text{ext}} + \boldsymbol{F}_{\text{d-d}}, \tag{2}$$

where m_i, $\boldsymbol{x}_i$, and $\boldsymbol{v}_i$ are the mass, position, and velocity of the droplet. $\boldsymbol{F}_{\text{fluid}}$ represents the vector of forces acting from the fluid on the droplet, $\boldsymbol{F}_{\text{ext}}$ is the external force vector, and $\boldsymbol{F}_{\text{d-d}}$ represents the inter-droplet force vector.

Droplet tracking with droplet-droplet interaction has a high computational cost. It is therefore important to keep the computational work necessary to calculate the particle forces as low as possible since the forces have to be calculated for each particle. Finally, models should be easily implementable in a numerical code. The computation and implementation costs of each force have to be evaluated in relation to the accuracy achieved. In such evaluation it is also important to asses the contribution of each force on the total force account.

Experiments are performed to observe the behaviour of the falling droplet exposed to an electrical field. The experimental setup consists of two vertically placed electrodes. One droplet is at rest at the lower electrode and a smaller droplet is released some millimetres above. The droplet motion is recorded with a high-speed CMOS camera. In the present work a first series of experiments is carried out releasing a rigid sphere particle with density and permittivity similar to water. Electric fields with different magnitudes are applied and the velocity of the particle is recorded as a function of the distance between the electrode and the particle itself. A second series of experiments is performed releasing water droplets and observing their motion when electric fields of different magnitudes are applied. The velocity of the droplet is recorded as a function of its distance from the electrode.

The experimental observations are compared with the results obtained numerically solving the governing equations for the droplet motion. The comparison between observed and predicted velocities versus normalized particle surface distance for rigid particle when no electric field is applied shows a good agreement. The forces dictating the motion of the particles such as buoyancy, drag, and film-drainage are properly modelled in our numerical framework. The prediction of the motion of a rigid particle under the influence of an electric field agrees well with the experimental observations. The electric force between two conductive particles is modelled with the analytical expression obtained by Davis [3]. It is possible to conclude that the accuracy of the predictions obtained using the DID model by Siu et al. [4] also is satisfactory as long as the falling particle is not too close to the stationary particle.

When comparing the predictions and the observations of the motion of a falling droplet, the effect

of internal circulation induced in the droplet has to be taken into account. Internal circulation reduces the viscous part of the drag force and therefore the drag coefficient needs to be corrected in order to account for this reduction [5]. Furthermore the surface tension varies over the droplet surface by the effect of surfactants on the interface and by elongation of the droplet, caused by the electric field. This leads to interfacial stresses that inhibit the creation of internal circulation. LeVan [6] suggested how to take into account the effect of surface tension gradient. Barnocky and Davis [7] show that the drainage between fluid spheres with arbitrary viscosity is different from rigid particles. The model proposed by Vinogradova [8] takes into account the slip between the liquid film and the approaching spheres. The comparison between observed and predicted velocities versus normalized droplet surface distance h/r_1 for fluid droplet shows a good agreement when all the effects influencing the droplet kinematics are modelled.

2 Modeling the fluid-droplet and body forces

Fluid droplet forces are transfered from the fluid to the droplets through friction and pressure difference. These forces are expressed exactly by the following surface integral:

$$\frac{1}{V_\mathrm{d}}\boldsymbol{F}_{\text{fluid}} = \frac{1}{V_\mathrm{d}}\int_{\mathcal{A}_\mathrm{d}} (-p_s\boldsymbol{n}_\mathrm{d} + \boldsymbol{\tau}_d \cdot \boldsymbol{n}_d)\ \mathrm{d}A \tag{3}$$

where V_d is the volume of the droplet. p_s is the pressure at the droplet surface, $\boldsymbol{n}_\mathrm{d}$ represents the unit outward normal vector and $\boldsymbol{\tau}_\mathrm{d}$ is the shear stress tensor at the droplet surface.

The pressure and the friction on the interface are unknown and Eq. (3) has to be modelled. In the Lagrangian framework the models for the surface integral attempt to provide particular physical meanings.

Drag force
The 'steady-state' drag force acts on a droplet in a uniform pressure field when there is no acceleration of the relative velocity between the droplet and the conveying fluid. The force reads:

$$\boldsymbol{F}_\mathrm{d} = \frac{1}{2}\rho_\mathrm{c} C_\mathrm{d} A|\boldsymbol{u}-\boldsymbol{v}|(\boldsymbol{u}-\boldsymbol{v}), \tag{4}$$

For a droplet Reynolds number Re_d below 1, the drag coefficient C_d for a rigid sphere is given by:

$$C_\mathrm{d} = \frac{24}{Re_d} \tag{5}$$

An internal circulation is induced in fluid spheres, that reduces the viscous part of the drag. For spherical clean bubbles and droplets, the induced internal circulation is accounted by the Hadamard-Rybczynski formula [9]:

$$C_\mathrm{d} = \frac{24}{Re_d}\frac{\lambda+2/3}{\lambda+1} \tag{6}$$

where $\lambda = \mu_d/\mu_c$ is the viscosity ratio. Surfactants on the interface and elongation of the droplet, caused by the electric field, give a variation in the surface tension. The surface tension gradient leads to interfacial stresses that inhibit the creation of internal circulation. The surface tension gradient is included in the formula by LeVan [6]:

$$C_d = \frac{24}{Re_d}\frac{3\lambda+2+2\kappa(\mu_c r_d)^{-1}+2/3\gamma_1(\mu_c|\boldsymbol{u}-\boldsymbol{v}|)^{-1}}{3\lambda+3+2\kappa(\mu_c r_d)^{-1}}, \tag{7}$$

where also the surface dilational viscosity κ is taken into account. However, in the present work surface dilational viscosity is neglected, $\kappa = 0$. In Eq. (7) it is assumed that the interfacial tension varies as follows:

$$\gamma = \gamma_0 + \gamma_1 cos\psi \tag{8}$$

where ψ is measured from the front stagnation point.

Virtual mass force
The virtual mass force $\boldsymbol{F}_{\text{vm}}$ is an unsteady force that describes the acceleration of fluid when a particle and the fluid have a relative acceleration. It reads:

$$\boldsymbol{F}_{\text{vm}} = \frac{\rho_{\text{c}} V_{\text{d}}}{2} \left(\frac{D\boldsymbol{u}}{Dt} - \frac{d\boldsymbol{v}}{dt} \right) \tag{9}$$

External body forces
We assume that the droplets have no net charge, hence the electric field as a far field force can be neglected. On the other hand, the electric field gives rise to dipole-dipole interactions between the droplets, which are modelled as inter-droplet forces. Then the gravity is the only external force and the buoyancy force is given by:

$$\boldsymbol{F}_{\text{b}} = (\rho_{\text{d}} - \rho_{\text{c}})\, g V_{\text{d}} \boldsymbol{e}_{\text{g}} \tag{10}$$

where g and $\boldsymbol{e}_{\text{g}}$ are the modulus and the direction of the gravity.

Observations
In the present work, the effects of the pressure gradient, the Basset history force and the lift forces have been neglected. The pressure difference over a small droplet is negligible due to the size of the droplets. The contribution from the gravity is handled separately. Lift forces are due to droplet rotation and shear forces, and can therefore be neglected when a rigid sphere or droplet is falling in a stagnant fluid. Due to the small size of the spheres and the high viscosity of the oil, the particle time scale is very small. Thereby follows that the Stokes number is small and the Basset history force can therefore be neglected [10].

3 Modeling droplet-droplet forces

The inter-droplet forces are the film thinning forces, due to the drainage of the fluid between the droplets, and the electric forces due to polarisation of the conductive water droplets.

Film-thinning force
The film-thinning force is caused by drainage of the liquid film between two approaching droplets. The derivation of the formulas usually requires that the gap between the particles is small $h \ll a$ and that the flow is within Stokes regime $Re_d h \ll a$. $a = (r_1 r_2)/(r_1 + r_2)$ is the reduced radius. For rigid spheres the film-thinning force is written as [11]:

$$\boldsymbol{F}_{\text{f}} = -\frac{6\pi\mu_{\text{c}} a^2 (\boldsymbol{v}_r \cdot \boldsymbol{e}_{\text{r}})}{h} \boldsymbol{e}_r, \tag{11}$$

where $\boldsymbol{v}_{\text{r}}$ is their relative velocity vector and $\boldsymbol{e}_{\text{r}}$ indicates the direction of the relative motion. When the particles are very close, a slip will occur and avoid a zero impact velocity. The formula of Vinogradova [8] includes a slip distance b, as shown in Fig. 1, and is written as:

$$\boldsymbol{F}_{\text{f}} = -\frac{6\pi\mu_{\text{c}} a^2 (\boldsymbol{v}_r \cdot \boldsymbol{e}_{\text{r}})}{h} \left\{ \frac{2h}{6b} \left[\left(1 + \frac{h}{6b} \right) \ln \left(1 + \frac{6b}{h} \right) - 1 \right] \right\} \boldsymbol{e}_r. \tag{12}$$

Barnocky and Davis [7] studied the drainage between fluid spheres with arbitrary viscosity, from bubbles to rigid particles. The influence of surfactants and the surface tension gradient were not paid any effort. In our case, the droplet at rest is covered with surfactants and is therefore handled as a rigid sphere. The film force formula of Barnocky and Davis [7] for a fluid sphere approaching a rigid sphere reads:

$$\boldsymbol{F}_{\mathrm{f}} = -\frac{6\pi\mu_{\mathrm{c}}a^2(\boldsymbol{v}_r\cdot\boldsymbol{e}_{\mathrm{r}})}{h}\frac{1+0.47\xi+0.0047\xi^2}{1+1.13\xi+0.19\xi^2}\boldsymbol{e}_r, \tag{13}$$

where $\xi = \lambda^{-1}\sqrt{(a/h)}$.

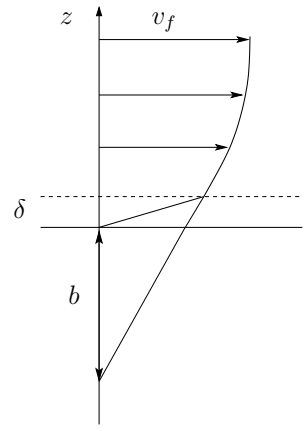


Figure 1: Illustration of the slip length b.

Electric forces acting on the spherical droplets
Consider an uncharged spherical droplet placed in an insulating medium. The droplet is furthermore subjected to an electric field $\boldsymbol{E}_0$. The field outside a dielectric sphere of permittivity ε_d corresponds exactly to the electric field of a dipole located at the sphere centre. The value of this dipole moment $\boldsymbol{p}$ depends on the sphere size, permittivity and the strength of the electric field. Due to the polarisation of the droplet, the poles will have charges of same magnitude but opposite polarity, preserving zero net charge. In a homogeneous field the net force on the droplet is zero. Subjected to an inhomogeneous field the droplet will experience a stronger field at one pole than at the other, resulting in a net force acting on the droplet in the direction of the field gradient. A phenomenon called dielectrophoresis. The resulting force is given by $\boldsymbol{F} = (\boldsymbol{p}\cdot\boldsymbol{\nabla})\,\boldsymbol{E}$. If the permittivity of the drop ε_d is higher than the permittivity of the surrounding medium ε_{oil}, the drop will move towards the high field region. An inhomogeneous electric field may for instance be set up by nearby point charge or another dielectric droplet, see Fig. 2. In the latter case the electrostatic force attracts the two droplets, given that $\varepsilon_d > \varepsilon_{oil}$.

Point dipole model
For large droplet distances $|\boldsymbol{d}|/r_d \gg 1$ we can approximate the electrostatic interaction between two droplets as the force between two dipoles located at the sphere centres, see Fig. 2. This is frequently

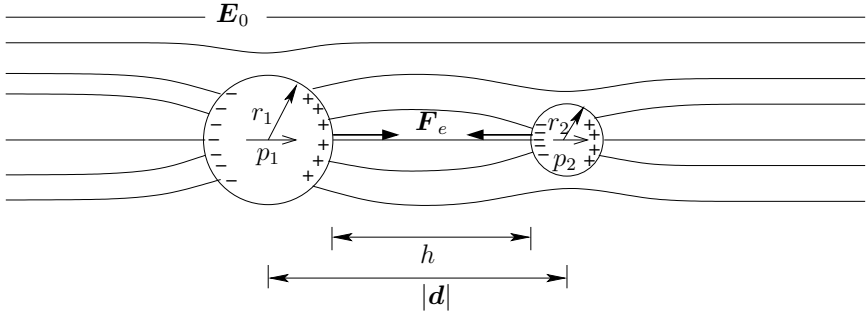


Figure 2: Electric forces between two conductive spheres.

referred to as the *point-dipole* approximation. The forces in radial direction F_r and tangential direction F_t read [12]:

$$F_\mathrm{r} = \frac{12\pi\beta^2\varepsilon_\mathrm{oil}|\boldsymbol{E}_0|^2 r_2^3 r_1^3}{|\boldsymbol{d}|^4}\left(3\cos^2\theta - 1\right) \tag{14}$$

$$F_\mathrm{t} = -\frac{12\pi\beta^2\varepsilon_\mathrm{oil}|\boldsymbol{E}_0|^2 r_2^3 r_1^3}{|\boldsymbol{d}|^4}\sin(2\theta), \tag{15}$$

where θ is the angle between the direction of the electrical field $\boldsymbol{E}_0$ and the relative droplet position vector $\boldsymbol{d}$. β is defined as:

$$\beta = \frac{\varepsilon_d - \varepsilon_\mathrm{oil}}{\varepsilon_d + 2\varepsilon_\mathrm{oil}}, \tag{16}$$

The Dipole induced dipole model (DID)

The above point-dipole model is not valid when the droplets are approaching each other. In the literature there are different approaches to find the dipole-dipole forces beyond the point dipole approximation for multiple particles of arbitrary size and position. Clercx and Bossis [13] presented a multi-pole expansion method that gives good results, but the calculation is complex. A more promising method, the *multiple image method*, was presented by Yu et al. [14]. The two first terms in the multiple image method gives the dipole induced dipole model (DID) [4], which is simple and numerical efficient. Siu et al. [4] show that the DID model is in good agreement with the experimental values obtained by Klingenberg et al. [12] for $|\boldsymbol{d}|/r_1 > 0.1$ for equally sized conductive particles. It is customary to write the dipole forces on the form:

$$F_r = \frac{12\pi\beta^2\varepsilon_\mathrm{oil}|\boldsymbol{E}_0|^2 r_2^3 r_1^3}{|\boldsymbol{d}|^4}\left(3K_1\cos^2\theta - 1\right) \tag{17}$$

$$F_t = -\frac{12\pi\beta^2\varepsilon_\mathrm{oil}|\boldsymbol{E}_0|^2 r_2^3 r_1^3}{|\boldsymbol{d}|^4}K_2\sin(2\theta). \tag{18}$$

For the DID model the coefficients K_1 and K_2, are written as:

$$K_1 = 1 + \frac{\beta r_1^3|\boldsymbol{d}|^5}{\left(|\boldsymbol{d}|^2 - r_2^2\right)^4} + \frac{\beta r_2^3|\boldsymbol{d}|^5}{\left(|\boldsymbol{d}|^2 - r_1^2\right)^4} + \frac{3\beta^2 r_1^3 r_2^3\left(3|\boldsymbol{d}|^2 - r_1^2 - r_2^2\right)}{\left(|\boldsymbol{d}|^2 - r_1^2 - r_2^2\right)^4}, \tag{19}$$

$$K_2 = 1 + \frac{\beta r_1^3|\boldsymbol{d}|^3}{2\left(|\boldsymbol{d}|^2 - r_2^2\right)^3} + \frac{\beta r_2^3|\boldsymbol{d}|^3}{2\left(|\boldsymbol{d}|^2 - r_1^2\right)^3} + \frac{3\beta^2 r_1^3 r_2^3}{\left(|\boldsymbol{d}|^2 - r_1^2 - r_2^2\right)^3}, \tag{20}$$

In the limit $|\boldsymbol{d}| \to \infty$ the coefficients K_1 and K_2 approach unity and we recover the point dipole model given by Eq. (14) and (15).

The analytical solution

Davis [3] found an analytical solution to Laplace's equation for two conducting spheres of arbitrary size, displacement and net charge, using bi-spherical coordinates. The exact solution for uncharged spheres is given by:

$$F_r = 4\pi\varepsilon_\mathrm{oil}|\boldsymbol{E}_0|^2 r_2^2\left(F_1\cos^2\theta + F_2\sin^2\theta\right) \tag{21}$$

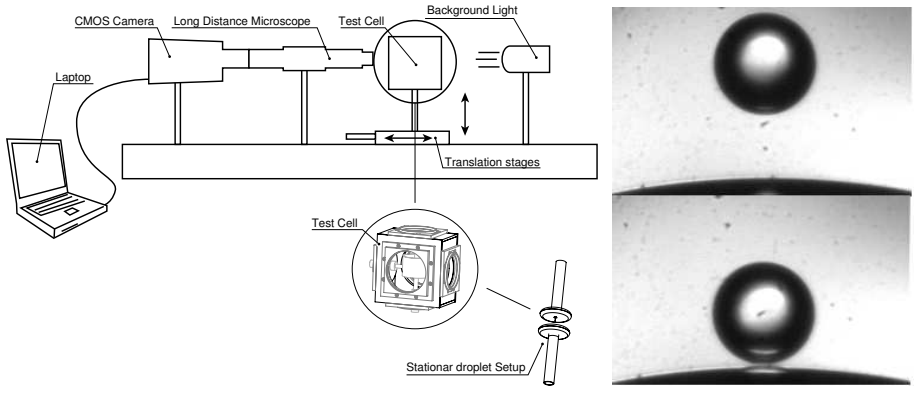
$$F_t = 4\pi\varepsilon_\mathrm{oil}|\boldsymbol{E}_0|^2 r_2^2 F_3\sin(2\theta), \tag{22}$$

where the parameters F_1, F_2, and F_3 are complicated series depending on the ratios $|\boldsymbol{d}|/r_2$ and r_1/r_2. Unfortunately, the computational cost required for calculating $F_1 - F_3$ is high in a multi-droplet

situation. However, the exact solution is excellent for benchmarking other models in cases with two particles/droplets. For large drop separations $|\boldsymbol{d}|/r_1 \gg 1$ the force components $F_1 - F_3$ approach the values of the point-dipole Eq. (14) and (15). For small separations $|\boldsymbol{d}|/r_1 < 1$, F_2 and F_3 takes constant values while F_1 diverges. Atten [2] showed that F_1 takes the asymptotic value:

$$F_1 = \frac{4\left(r_2/|\boldsymbol{d}|\right)^{0.8}}{3\left(1 + r_2/2r_1\right)^4}. \tag{23}$$

4 Experimental setup



(a) Experimental set-up (b) Experimental observation

Figure 3: (a) Experimental set-up designed for visual observation of the behaviour of drops in oil emulsion exposed to the effect of an external electric field. (b) An experimental observation where a small droplet comes into contact with a bigger one.

Experiments are designed for visual observation of water drops in oil under the influence of electric field stress. A vertical $15\,\mathrm{mm}$ electrode-gap arrangement is placed inside a cubic test cell with side lengths $150\,\mathrm{mm}$. The cell is placed in an optical bench to obtain a shadow-graphic representation of the water drops as shown in Fig. 3(a). One drop of $4\,\mathrm{mm}$ in diameter is positioned at rest and in direct contact with the lower high voltage electrode, and sub-millimetre sized droplets are released within the upper grounded electrode from a glass capillary. Electric fields are vertical thus parallel to the drop-drop impact vectors and in the range $250 - 400\,\mathrm{V/mm}$. Bipolar square voltages with frequencies $10\,\mathrm{Hz}$ and $100\,\mathrm{Hz}$ are used. Drop interactions and coalescence are recorded with a Phantom V4 high speed CMOS camera capable of 1000 frames per second at $512 \times 512\,\mathrm{pixels}$ resolution. The position and the velocity of the droplets are digitally extracted from the sequential frames. Uncertainty in measured droplet diameter is less than $5\,\mu\mathrm{m}$. Water is distilled and a small amounts of salt is added (3.5 $H_20\,\mathrm{wt}\%$ NaCl), and the oil used is Nynäs Nytro 10X transformer oil. A similar experiment is performed with un-deformable rigid spheres that are designed to have similar density and permittivity to salt water, see Table 1. The falling rigid spheres are of Ugelstad type, $\approx 150\,\mu\mathrm{m}$ diameter polystyrene base spheres ($\rho = 1050\,\mathrm{kg/m^3}$) covered with $400\,\mathrm{nm}$ silver coating. Glass capillaries are made hydrophobic to ease the release of generated droplets, and coated with gold to avoid static charge transfer from glass to water. At $20°C$ the oil viscosity is $13.7\,\mathrm{Pa \cdot s}$, however some series with rigid spheres are performed at a slightly higher temperature ($23 - 25°C$) and an interpolated

Table 1: Properties of Nytro 10X, water with a small salt content and rigid sphere particle at 20°C.

	Oil	Water	Rigid sphere
Density ρ	$875\,\mathrm{kg/m^3}$	$1022\ \mathrm{kg/m^3}$	$1206 - 1210\,\mathrm{kg/m^3}$
Viscosity μ	$13.7 \cdot 10^{-3}\,\mathrm{Pa \cdot s}$	$0.98 \cdot 10^{-3}\,\mathrm{Pa \cdot s}$	∞
Relative permittivity ε	2.2	1000	1000

Table 2: Experimental series.

Electric field	Rigid sphere radius	Droplet radius
$0\,\mathrm{V/mm}$	$82.5\,\mu\mathrm{m}$	
$250\,\mathrm{V/mm}$	$70.5\,\mu\mathrm{m}$	$105\,\mu\mathrm{m}$
$300\,\mathrm{V/mm}$	$70.5\,\mu\mathrm{m}$	$110\,\mu\mathrm{m}$ and $58\,\mu\mathrm{m}$
$400\,\mathrm{V/mm}$	$72\,\mu\mathrm{m}$	$110\,\mu\mathrm{m}$

viscosity of 12.5 $\mathrm{Pa \cdot s}$ is used for these simulations. Fig. 3(b) shows a typical collision between a 200 $\mu\mathrm{m}$ diameter falling water droplet and a 4 mm diameter stationary drop.

5 Results and Discussions

Numerically, the governing equations (1) and (2) are solved with a Runge-Kutta Fehlberg 4-5 solver with step size control, see for instance Hairer et al. [15]. Accurate simulations are ensured by using a relative tolerance of 10^{-5} and an absolute tolerance of 10^{-25}. The expression for the forces described in Section 2 are used. An ideal bipolar squared voltage is assumed.

Figure 4 shows a comparison between experimental observations and numerical predictions of the kinematics of a rigid spherical particle falling towards an electrode. No electric field is applied. The b constant in the expression of Vinogradova Eq. (12) used in the numerical calculation is $10^{-7}\,\mathrm{m}$. The velocity of the rigid sphere is plotted versus the normalized distance h/r_1. The good agreement between observations and predictions shows that drag Eq. (4), buoyancy Eq. (10), and film-drainage Eq. (12) are well modelled in our numerical framework.

A comparison between observed and predicted velocities versus normalized particle surface distance h/r_1 for rigid spheres under the influence of an electric field is plotted in Fig. 5. The buoyancy, drag, and film-drainage are modelled as in the previous numerical exercise presented in Fig. 4. Figure 5(a) shows the effect of different electric field magnitudes on the falling velocity of the particle. The analytical model by Davis Eq. (21) and (22) is used in the numerical calculations of the particle velocity. A good agreement between numerical results and experimental observations is obtained. The velocity of the falling droplet observed when the magnitude of the electric field is greatest $|\boldsymbol{E}_0| = 400\,\mathrm{V/mm}$, is higher than for weaker field magnitude. This follows expectations. Figure 5(b) shows a comparison between observed and predicted velocities obtained using different models for the induced electrical forces. The magnitude of the applied electric field is $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$. The accuracy of the results obtained by using the DID model is high as long as the droplets are not too close to each other. The point dipole does not provide satisfactory results.

When comparing predictions and the observations of the motion of a falling fluid droplet, the effect of internal circulation induced in the droplet has to be taken into account. Internal circulation reduces the viscous part of the drag force and therefore the drag coefficient needs to be corrected in order to account for this reduction as outlined in Section 2. Furthermore the surface tension of

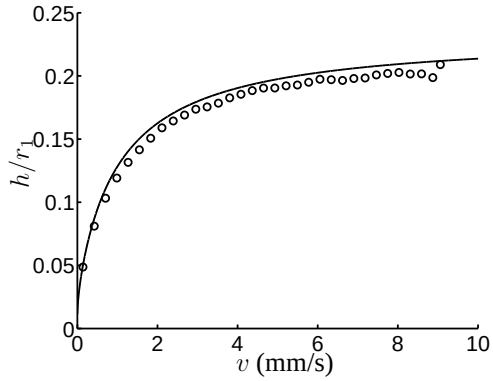
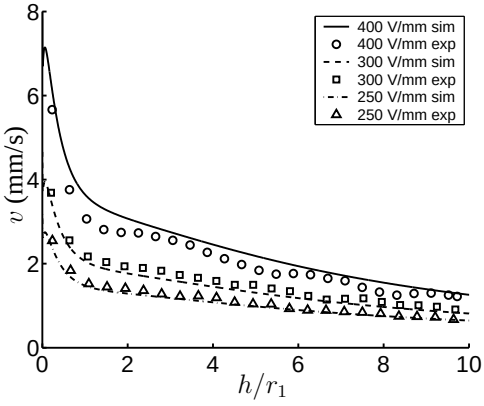


Figure 4: Predicted and observed velocity versus normalized particle surface distance for a falling rigid sphere with radius $r_d = 82.5\,\mu$m. No electric field is applied.

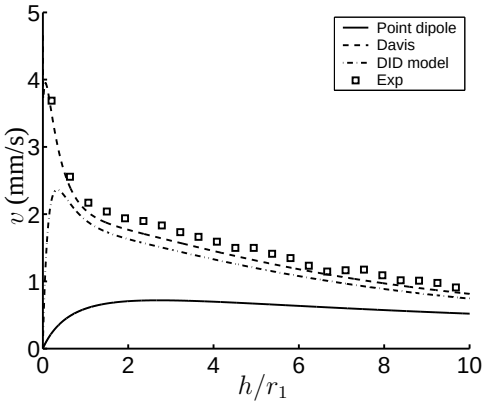
the droplet is varied by the effect of surfactants on the interface and by elongation of the droplet, caused by the electric field. This leads to interfacial stresses that inhibit the creation of internal circulation. LeVan [6] suggests how to take into account the effect of surface tension gradient. Barnocky and Davis [7] derived expressions for the drainage between fluid spheres with arbitrary viscosity. The model proposed by Vinogradova [8] takes interfacial slip effects into account. Although Vinogradova [8] proposed a correction for hydrophobic surfaces, the present work shows that the Eq. (12) can be used with good results under other conditions. A comparison between observed and predicted velocities versus normalized particle surface distance h/r_1 for fluid droplets under the influence of an electric field is plotted in Fig. 6. The average radius of the droplets is $r_1 \approx 110\mu m$. The slip distance used in the Vinogradova equation (12) is $b = 10^{-6}\,\mathrm{m}$ and the magnitude of the interfacial tension gradient in Eq. (7) is $\gamma_1 = 10^{-5}\,\mathrm{N/m}$. Figure 6(a) shows the effect of different electric field magnitudes on the falling droplet velocity. The analytical model by Davis Eq. (21) and (22) is used in the numerical calculations of the particle velocity and a good agreement between numerical results and experimental observations is obtained. Figure 6(b) shows a comparison between observed and predicted velocities obtained using different models for the induced electrical forces. The magnitude of the applied electric field is $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$. The same trend as previously observed for the rigid spheres is observed.

Figure 7 shows a comparison between observed and predicted velocities versus normalized droplet surface distance h/r_1 for water droplets of averaged radius $r_1 \approx 110\mu m$. Figure 7(a) shows the results obtained by different drag force models. The electric field is constant $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$. The Davis' analytical expressions, the Vinogradova's model with $b = 10^{-6}\,\mathrm{m}$ and Eq. (7) with $\gamma_1 = 10^{-5}\,\mathrm{N/m}$ are used in the calculations. The best agreement is obtained when the equation proposed by LeVan [6] is used in the numerical calculations. Figure 7(b) shows the results obtained for different drainage models. The electric field is kept constant $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$. The Davis' analytical expressions and the LeVan drag force model are used in the calculations. In the formula of Vinogradova, $b = 10^{-6}\,\mathrm{m}$. The best agreement is obtained when the model proposed by Vinogradova Eq. (12) is used. The viscous sphere model Eq. (13), provides a severe overestimation of the velocity, and shows that the surface cannot be handled as ideal. The results obtained from the hard sphere model seems to well agree with the experimental observations up until the droplets are very close.

Figure 8 shows a comparison between observed and predicted velocities versus normalized droplet surface distance h/r_1 for water droplets of different radii. In Fig. 8(a) two different droplet sizes $r_{\text{big}} = 110\,\mu$m, $r_{\text{small}} = 58\,\mu$m are considered. The electric field is constant $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$. The slip distance in the Vinogradova model is $b = 10^{-6}\,\mathrm{m}$ for the biggest droplet and $b = 10^{-7}\,\mathrm{m}$ for the

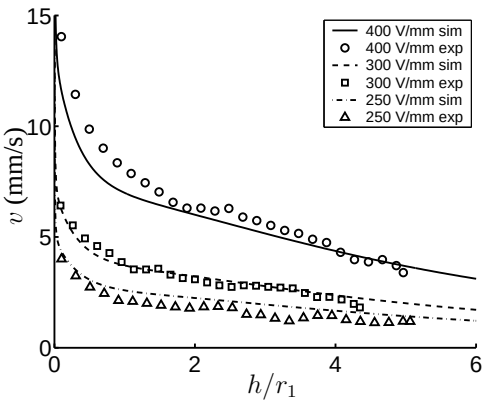


(a) Different electric field magnitudes are applied in the numerical calculations of the particle kinematics. Davis' analytical expressions are used.

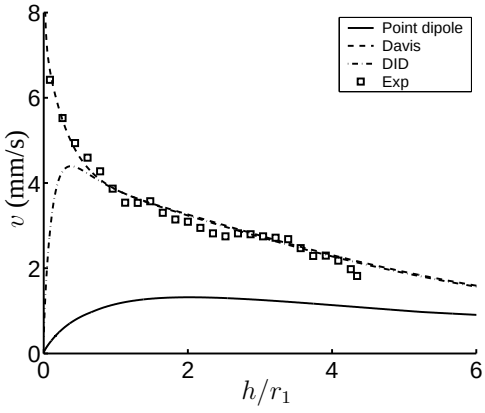


(b) Constant electric field magnitude is applied $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$ and different models for the induced electrical forces are used in the numerical calculations of the particle kinematics

Figure 5: Observed and predicted velocities versus normalized particle surface distance h/r_1 for rigid spheres. The radius of the sphere is $r_1 \approx 70\mu m$ and the slip length in Eq. (12) is $b = 10^{-7}\,\mathrm{m}$.

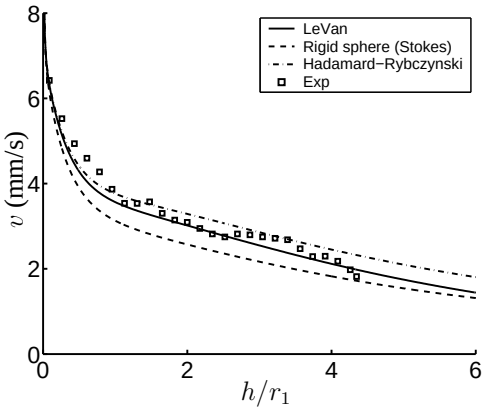


(a) Different electric field magnitudes are applied and in the numerical calculations of the droplet kinematics. Davis' analytical expressions are used in the numerical calculations.

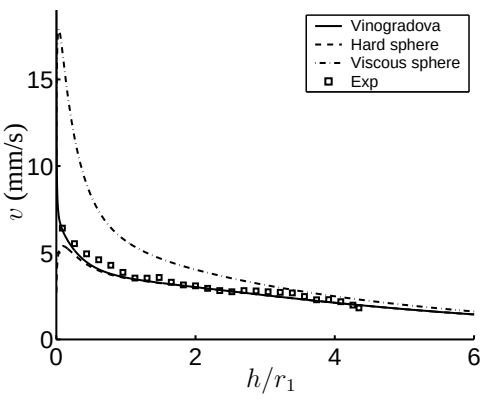


(b) Constant electric field magnitude is applied $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$ and different models for the induced electrical forces are used in the numerical calculations of the droplet kinematics

Figure 6: Observed and predicted velocities versus normalized droplet surface distance h/r_1 for water droplets. The radii of the droplets are $r_1 \approx 110\mu m$. The slip distance in Eq. (12) is $b = 10^{-6}\,\mathrm{m}$ and the magnitude of the interfacial tension gradient in Eq. (7) is $\gamma_1 = 10^{-5}\,\mathrm{N/m}$.

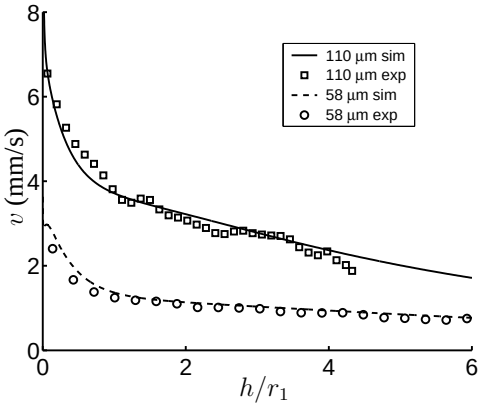


(a) The results obtained by different drag force models are compared to experimental observations. $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$. Davis' analytical expressions, Vinogradova's model with $b = 10^{-6}\,\mathrm{m}$, and an interfacial tension variation of $\gamma_1 = 10^{-5}\,\mathrm{N/m}$ are used in the calculations.

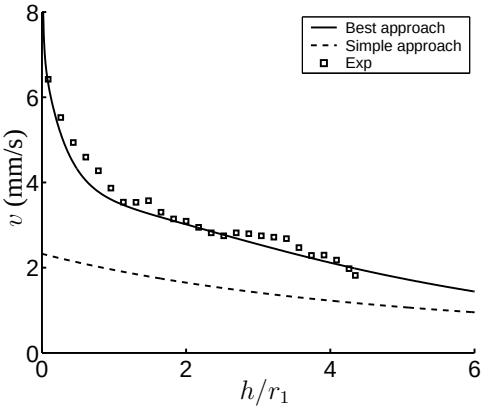


(b) The results from different drainage models are compared to experimental observations. $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$. In Vinogradova's model $b = 10^{-6}\,\mathrm{m}$. Davis' analytical expressions are used.

Figure 7: Observed and predicted velocities versus normalized droplet surface distance h/r_1 for water droplets. The radius of the droplet is $r_1 = 110\mu m$.



(a) Different droplet size $r_{\text{big}} = 110\,\mu\mathrm{m}$, $r_{\text{small}} = 58\,\mu\mathrm{m}$. $|\boldsymbol{E_0}| = 300\,\mathrm{V/mm}$. the slip distance in the Vinogradova model is $b = 10^{-6}\,\mathrm{m}$ for r_{big} and $b = 10^{-7}\,\mathrm{m}$ for r_{small}. Furthermore $\gamma_1 = 10^{-}5\,\mathrm{N/m}$ for r_{big} and $\gamma_1 = 2 \cdot 10^{-5}\,\mathrm{N/m}$ for r_{small}.



(b) 'Simple' versus 'best' modeling at $|\boldsymbol{E_0}| = 300\,\mathrm{V/mm}$. Droplet size $r = 110\,\mu\mathrm{m}$. The simple modeling strategy includes: point dipole, rigid sphere drag and no film force. The best modeling strategy employs the models of Davis, LeVan, and Vinogradova

Figure 8: Observed and predicted velocities versus normalized droplet surface distance h/r_1 for water droplets

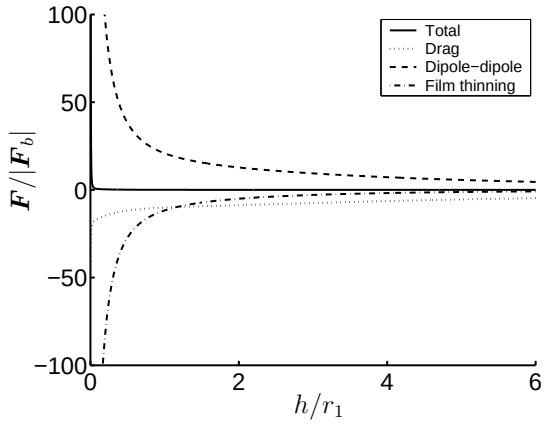


Figure 9: Contribution of the force components, scaled with the buoyancy force

smallest one. Furthermore the value of γ_1 for the big droplet is $\gamma_1 = 10^{-5}\,\mathrm{N/m}$ and $\gamma_1 = 2{\cdot}10^{-5}\,\mathrm{N/m}$ for the small one. The numerical predictions agree very well with the experimental observations. In Fig. 8(b) numerical prediction obtained with two different approaches are compared to experimental observations. The 'simple approach' adopts point-dipole, rigid sphere drag and no film-thinning force while the 'best approach' employs the models of Davis, LeVan, Vinogradova. The agreement between observations and predictions obtained by means of the 'simple approach' is not satisfactory. Figure 9 shows the contribution of the different forces normalized by the buoyancy force for the $|\boldsymbol{E}_0| = 300\,\mathrm{V/mm}$ case. The film-thinning and drag forces almost balance the electric force until the droplets are very close. One also sees that the film-thinning force gives a significant contribution to the force account far outside the range considered in the derivation.

6 Observations

The comparison between observed and predicted velocities versus normalized droplet surface distance h/r_1 for fluid droplet shows a good agreement when all the effects influencing the droplet kinematics are modelled.

In the case of rigid particle the models used to describe drag Eq. (4), buoyancy Eq. (10), and film-drainage Eq. (12) give good agreement with the experimental observations. The effect of the electric forces on the motion of the rigid particles are well taken into account when the Davis model Eq. (21) and (22) are used in the numerical calculations. It was also observed that the numerical efficient DID model Eq. (17) provides good results as long as normalized droplet surface distance is not too small.

When the motion of fluid droplets is to be predicted, the simple model for drag, buoyancy, and film-drainage used for the rigid particle are not longer valid. The effect of internal circulation induced in the droplet has to be taken into account together with the variation of the surface tension of the droplet due to the electric field. In the present work different models are assessed. The use of the model proposed by LeVan Eq. (7) for the drag force, by Vinogradova Eq. (12) for the film-thinning force and the Davis' analytical expression Eq. (21) and (22), provide numerical predictions that well agree with the experimental observations.

Acknowledgments

The authors are thankful to ABB, Statoil and the Research Council of Norway for supporting the present work. In particular we are grateful to Pål Jahre Nilsen and Lars Lundgaard for the fruitful discussion we had concerning the many challenges we were faced with during the present work.

References

[1] J. S. Eow, M. Ghadiri, A. O. Sharif, and T. J. Williams. Electrostatic enhancement of coalescence of water droplets in oil: a review of current understanding. *Chem. Eng. J.*, 84:173–192, 2001.

[2] P. Atten. Electrocoalescence of water droplets in an insulating liquid. *J. Electrostatics*, 30:259–270, 1993.

[3] M. H. Davis. Two charged spherical conductors in a uniform electric field: forces and field strength. Rand. Corp. Memorandum RM-3860-PR, January 1964.

[4] Y. L. Siu, T. K. Wan Jones, and K. W. Yu. Interparticle force in polydisperse electrorheological fluid: Beyond the dipole approximation. *Comput. Phys. Commun.*, 142:446–452, 2001.

[5] J. Happel and H. Brenner. *Low Reynolds number hydrodynamics*. Martinus Nijhoff Publishers, The Hague, The Netherlands, 1983.

[6] D. M. LeVan. Motion of droples with a Newtonian interface. *J. Colloid Interface Sci.*, 83(1):11–17, 1981.

[7] G. Barnocky and R. H. Davis. The lubrication force between spherical drops, bubbles and rigid particles in a viscous fluid. *Int. J. Multiphase Flow*, 15:627–638, 1989.

[8] O. I. Vinogradova. Drainage of a thin liquid film confined between hydrophobic surfaces. *Langmuir*, 11:2213–2220, 1995.

[9] C. Crowe, M. Sommerfeld, and Y Tsuji. *Multiphase flows with droplets and particles*. CRC Press, Boca Raton, USA, 2nd edition, 1998. ISBN 0-8493-9469-4.

[10] E. E. Michaelides. Hydrodynamic force and heat/mass transfer from particles, bubbles, and drops - the Freeman scholar lecture. *J. Fluids Eng. - Trans. ASME*, 125:209–238, March 2003.

[11] R. H. Davis, J. A. Schonberg, and J. M. Rallison. The lubrication force between two viscous drops. *Phys. Fluids A*, 1(1):77–81, January 1989.

[12] D. J. Klingenberg, F. van Swol, and C. F. Zukoski. The small shear rate response of electrorheological suspensions. II. extensions beyond the point-dipole limit. *J. Chem. Phys.*, 94(9):6170–6178, 1991.

[13] H. J. H. Clercx and G. Bossis. Many-body electrostatic interactions in electrorheological fluids. *Phys. Rev. E*, 48(4):2721–2737, October 1993.

[14] K. W. Yu and T. K. Wan Jones. Interparticle forces in polydisperse electrorheological fluids. *Comput. Phys. Commun.*, 129:177–184, 2000.

[15] E. Hairer, S. P. Nørsett, and G. Wanner. *Solving Ordinary Differential Equations I*. Springer, Berlin, 2nd edition, 1992. ISBN 3-540-56670-8.

Paper B

Adaptive cell structure for efficient detection of neighbouring particles in collisional particle-laden flows

Melheim, J. A. and Gjelsvik, A.
Submitted to Computers and Fluids, 2005

Adaptive cell structure for efficient detection of neighbouring particles in collisional particle-laden flows

Jens A. Melheim [a,*] , Anders Gjelsvik [b]

[a] *Norwegian University of Science and Technology (NTNU), Department of Energy and Process Engineering, Kolbjørn Hejes veg 1A, NO-7491 Trondheim, Norway*

[b] *SINTEF Energy Research, Sem Sælandsvei 11, NO-7465 Trondheim, Norway*

Abstract

An adaptive algorithm for particle-particle and particle-wall collision detection in the two-dimensional case is presented. The algorithm ensures an efficient computation of colliding particle flows without any input from the user regarding the particle cell structure. The physical domain is hierarchically divided and structured as a quadtree. The quadtree structure allows us to have variable cell width in space and time. We have implemented the entire tree with pointers. Lists of particles and boundary elements are linked to the particle cells, which constitute the leaf nodes of the quadtree. The algorithm is intended for particle-laden flows, which require small time steps, but within each time step, the algorithm is event-driven. From tests with uniform particle distribution and fixed cell structures with uniform cell sizes we infer that the number of particles inside a particle cell should be in the range of 0.25 to 3.0. The adaptive algorithm keeps the number of particles per cell within the given range and garantee an efficient computation. The advantages and performance of the presented adaptive algorithm are shown in two cases: Particles in a gravity field and in a 90° bend.

Key words: collision detection algorithm, hard sphere dynamics, adaptive, cell structure, quadtree, particle laden flow, discrete element method
PACS: 02.70.Ns, 45.50.-j, 47.11.+j, 47.55.Kf

* Corresponding author
Email address: jensm@pvv.ntnu.no (Jens A. Melheim).
URL: http://www.pvv.ntnu.no/~jensm/research.html (Jens A. Melheim).

1 Introduction

Flows with solid particles, droplets or bubbles occur in a wide variety of industrial applications, engineering problems, and science. Computer simulations are increasingly used to analyse these flows, for instance when chemical process equipment is designed.

During the last decade, the 'Discrete Element Method' (DEM), where every single particle is tracked has been used by many investigators and gives promising results, Refs. [1,2,3].

A crucial point when using the DEM model is the CPU-time consumption of detections of particle collisions. In very dilute flows, where the volume fraction of particles, α_p, is less than 10^{-6}, the inter-particle collisions are negligible. When the particles are more densely packed in the flow or in regions of the flow, the particle-particle interaction becomes important, Ref. [4]. Then, the algorithm for collision detection is of vital importance for the overall CPU-time consumption. An efficient solution strategy for the multi-particle problem was first addressed by Alder and Wainwright [5]. An efficient method for finding neighbouring particles is the particle-cell method. Either a uniform raster is used, Refs. [2,6,7], or a tree structure, Refs. [8,9]. Lubachevsky [6] made a thorough study of billiards simulations and pointed out some important properties. Sigurgeirsson et al. [7] followed up and presented a suitable data structure. The algorithm of Sigurgeirsson et al. [7] was almost optimal for billiards simulations where the particles are uniformly distributed. For non-uniformly distributed particle systems, Sigurgeirsson et al. presented a strategy for updating the particle cell mesh under computations and decreased the computational time compared with a fixed structure.

We have implemented an algorithm for simulating hard sphere particle dynamics in two dimensions, along the lines of Sigurgeirsson et al. [7], using a cell system for collision detection. Instead of a uniform cell structure (a *raster*), we have developed an automatic adaptive particle cell structure based on a quadtree representation. An adaptive cell structure attempts to optimise the particle cell size locally and ensures efficient computations. By *automatic* we mean that no information about the cell structure needs to be specified by the user. Several test runs in order to find the optimal uniform fixed cell-structure are therfore avoided.

The adaptive cell system is the main contribution of the present paper. However, to make the paper self-contained, we also describe the overall algorithm and its aspects. The paper is organised as follows: Section 2 gives an overview of the problem. The quadtree cell system and the col-

lision detection algorithm are described, and estimates of the computational complexity are given. In Section 3, computer experiments with a 'billiards' system are described. We have also simulated the repeated initialisation of the event list at each time step that is required if a particle-laden flow is to be simulated. From the billiards simulations we establish the control parameters for the adaptive cell structure Section 4 describes the strategy for the adaptive cell-size control, and simulation results from cases were the particle density is highly non-uniform. Section 5 contains a brief discussion of the performance of the proposed algorithm, and Section 6 gives the conclusions.

2 Background

2.1 Particle physics

Consider n spherical particles that follow a trajectory between the collisions that is given by:

$$m_i \frac{\mathrm{d}\boldsymbol{u}_i}{\mathrm{d}t} = \boldsymbol{F}_{\text{fluid}} + m_i \boldsymbol{g} \tag{1}$$

$$\frac{\mathrm{d}\boldsymbol{x}_i}{\mathrm{d}t} = \boldsymbol{u}_i, \tag{2}$$

where $i = 1, \ldots, n$. m_i is the mass, $\boldsymbol{u}_i$ is the velocity and $\boldsymbol{x}_i$ is the position of particle i. $\boldsymbol{F}_{\text{fluid}}$ are forces from the surrounding fluid and $\boldsymbol{g}$ is the gravitational field.

The collisions are considered to be instantaneous and the particles are assumed to be rigid spheres. The impulse equation for a collision is then given by:

$$m_1 \left(\boldsymbol{u}_1 - \boldsymbol{u}_1^0 \right) = -m_2 \left(\boldsymbol{u}_2 - \boldsymbol{u}_2^0 \right) \tag{3}$$

where subscripts 1 and 2 denote particle number and superscript 0 denotes the state before collision. For an elastic collision we have:

$$\left(\boldsymbol{u}_1^0 - \boldsymbol{u}_2^0 \right) \cdot \boldsymbol{n} = e \left(\boldsymbol{u}_1 - \boldsymbol{u}_2 \right) \cdot \boldsymbol{n} \tag{4}$$

where e is the restitution coefficient and $\boldsymbol{n}$ is the surface normal vector at the point of impact for particle 1. The velocities after a collision could

therefore be written as:

$$\boldsymbol{u}_1 = \boldsymbol{u}_1^0 - \boldsymbol{n}\left(\boldsymbol{n}\cdot(\boldsymbol{u}_1^0 - \boldsymbol{u}_2^0)\right)(1+e)\frac{m_2}{m_1+m_2} \tag{5}$$

$$\boldsymbol{u}_2 = \boldsymbol{u}_2^0 + \boldsymbol{n}\left(\boldsymbol{n}\cdot(\boldsymbol{u}_1^0 - \boldsymbol{u}_2^0)\right)(1+e)\frac{m_1}{m_1+m_2}, \tag{6}$$

where the tangential collision forces have been neglected. Further, collisions with the wall change the velocity perpendicular to the wall:

$$\boldsymbol{u}\cdot\boldsymbol{n}_w = -e_w\boldsymbol{u}^0\cdot\boldsymbol{n}_w \tag{7}$$

where $\boldsymbol{n}_w$ is the wall normal vector and e_w is the particle-wall restitution coefficient.

The governing equations for the particle's trajectory between collision (1) and (2) are solved by the forward Euler scheme, given by

$$\boldsymbol{y}_{n+1} = \boldsymbol{y}_n + dt\boldsymbol{y}_n', \quad \boldsymbol{y}_0 = \boldsymbol{y}(t_0), \tag{8}$$

where dt is the time step and $\boldsymbol{y}$ is a vector with a given initial condition, $\boldsymbol{y}(t_0)$.

When assuming that the spherical particles move along straight trajectories, the time until a collision between particle i and particle j is given by [2]:

$$t_{ij} = \frac{-\boldsymbol{r}_{ij}\cdot\boldsymbol{u}_{ij} - \sqrt{(\boldsymbol{r}_{ij}\cdot\boldsymbol{u}_{ij})^2 - |\boldsymbol{u}_{ij}|^2\left(|\boldsymbol{r}_{ij}|^2 - (R_i+R_j)^2\right)}}{|\boldsymbol{u}_{ij}|^2}, \tag{9}$$

where $\boldsymbol{r}_{ij} = \boldsymbol{x}_j - \boldsymbol{x}_i$, $\boldsymbol{u}_{ij} = \boldsymbol{u}_j - \boldsymbol{u}_i$ and R is the particle radius.

2.2 Cell system for collision detection

If we have n particles, checking for collisions by checking every possible pair of particles would lead to $n(n-1)/2$ checks. To avoid this, we use the cell method. The spatial domain is partitioned into rectangular cells, and the idea is that if we keep track of which particles are in each cell, and if the cells are larger than the largest particle, then for each particle we need only check for collisions with particles in the neighbouring cells.

There is a considerable literature on a collision detection algorithm based on a cell system. Lubachevsky [6] and Sigurgeirsson et al. [7] use simple cell systems of uniform grid size. Kim et al. [9] employed a bounded-balanced tree where the cells are at the leaves. They also describe how to

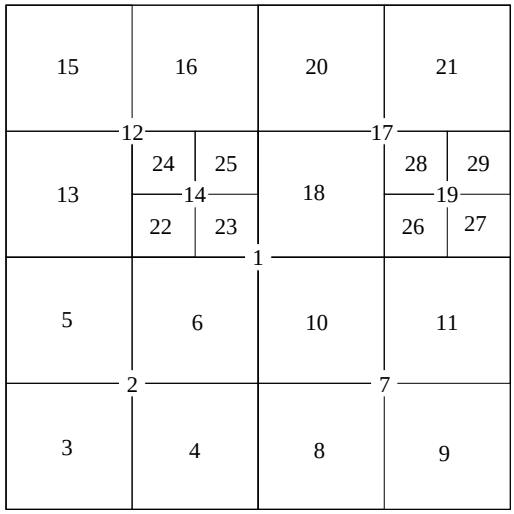


Figure 1. An example of a subdivision in cells

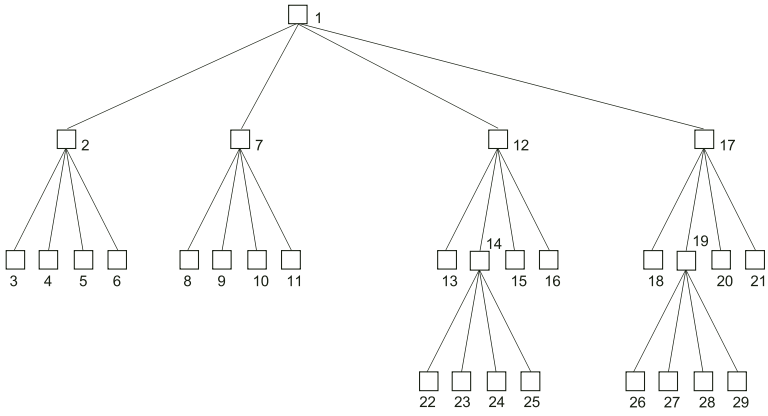


Figure 2. The quadtree corresponding to Fig. 1.

use a hierarchy of grids with different cell sizes to improve the performance for varying particle sizes. Vemuri et al. [8] used an *octree* (3D) for holding the cell structure. It seems that they use a storage scheme where only the nodes for the cells that contain particles are stored.

We have chosen to use a *quadtree* to represent the cells (working in two dimensions). A quadtree is the two-dimensional analogue of an octree [8], and is frequently used for handling geometrical data and in picture processing. There is a rich literature on quadtrees and octrees, examples are Refs. [10,11,12,13].

Each node in the quadtree represents a square spatial area. Each node that

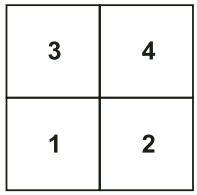


Figure 3. Numbering of quadrants

is not a leaf node, has four sons, which represent a subdivision of the cell in quadrants to give smaller cells, according to fixed rules. Each leaf node in the quadtree represents a spatial cell that is not further subdivided. It is these cells that we use for collision detection. Since the cells are represented by (leaf) nodes, we shall sometimes refer to them as nodes. An example of a cell partition is shown in Fig. 1, where node 1 represent the root, that is divided into four quadrants; node 2, 7, 12, and 17. These quadrants are in turn subdivided. At this level, only the nodes 14 and 19 are subdivided. The corresponding quadtree is shown in Fig. 2.

A quadtree representation is more complicated to implement than a straightforward rectangular grid of equal-sized cells (a *raster*). In trivial cases, the computing time may also increase. However, using a quadtree, we obtain increased flexibility of the cell system. The quadtree cell structure can easily adapt to complicated geometries, and the cell size can vary with the spatial variations in the number density of particles.

The root of the quadtree corresponds to a square with side length 1 covering the area of interest. The physical dimensions are scaled to obtain unit side length. A key feature of the quadtree is that the subdivision of cells/nodes follows a fixed labelling sequence for the quadrants, for instance as shown in Fig. 3. This defines the *sontype* of a node. For the cell in the southeast (SE) quadrant in Fig. 3, the sontype is 2. With the sontype recorded at each node, it is possible to calculate the position of a cell by following links up to the root node. Going downwards, a record of the sontypes encountered defines a unique *path* from the root to each node. Geometrically, to obtain a quadtree with $N+1$ levels, we start with a square with side one and repeat the subdivision procedure N times. This gives a subdivision in squares of the initial square. The smallest squares have side length $1/2^N$. The root node is said to be at level N, while the leaves are at level 0. For instance, the tree corresponding to the subdivision in Fig. 1 has $N = 3$ levels and is shown in Fig. 2.

There are several ways of representing a quadtree, see Samet [10]. One way is to represent the quadtree as a tree, with pointers. This is easy to implement as a linked list of nodes. The disadvantage is that the non-leaf nodes require much space. Other methods avoid pointers. Typically

in Refs. [10,11], the nodes are sorted according to their paths from the root node and stored with the paths. A path can be coded in two bits per sontype number (4 possible values). Finding a given node requires a search among the stored nodes.

We have chosen to work with the full tree representation, with pointers. This makes it easier to dynamically change the cell structure during computations, for instance to use a finer subdivision in places where the density of particles becomes high. To avoid restructuring we also store leaf nodes that contain no particles, since this will change as particles enter and leave during simulation.

We use a linked list of node records to store the quadtree. If the tree grows, the necessary nodes can be added at the end. At each node we record 6 items: A pointer to father node, a pointer to each of the four sons, and the sontype. At the leaf nodes, the first son pointer is set to zero to indicate that this is a leaf node. The pointers to the three other sons are not needed, so we use two of these spaces for pointers into lists, as will be described later.

2.2.1 Particle lists

Before the simulation starts, lists of particles contained in each cell must be established. These lists are set up as linked lists. Since a particle's centroid can be in only one cell, all the lists can be contained in a single array of length n, the number of particles. Each cell has a pointer to the first particle in the list (the space of the second son pointer is used for this), and for each particle there is a pointer to the next particle in the list. The lists are established on basis of the initial particle positions, and they are updated during the simulations.

2.2.2 Finding the neighbour-particles lists

In the collision detection, there are three cases where neighbour particle lists are required, as described by Sigurgeirsson et al. [7].

(1) When we establish the initial event list, all particles are considered systematically. The whole quadtree is traversed, beginning with the leaf node in the South West corner. For a given cell, only particles in neighbour cells that are in the North, North East, East and South East directions are checked for collisions. Particles in neighbour cells in the other directions will have appeared earlier during the tree traversal and are already checked for collisions. This can be termed a 'forward' search.

(2) Immediately after a particle has made a transfer to a new cell, it has new neighbour cells in the direction of transfer. Only particles in the new neighbour cells need to be checked for collisions.
(3) After a collision or a check event (described in Section 2.3), particles in neighbour cells in all directions have to be checked for collisions.

In all three cases, a list of neighbour cells is produced, and from the particle lists of these neighbour cells a list of particles to be checked is set up. Therefore, one of the most basic operations in the quadtree is that of finding neighbour cells under various conditions, see Samet [10].

2.2.3 Handling boundary segments

The boundary of the spatial domain is assumed to be approximated by piecewise linear segments. In order to check for boundary interaction, we keep for each cell a list of the boundary segments that are within reach of particles contained in the cell. These lists are kept as linked lists. They differ from the particle lists in that a given segment can occur in more than one list. A son pointer of the corresponding node is used to point at the beginning of the cell's boundary segment list.

2.2.4 Merging and splitting cells

We use an adaptive cell structure that may change during the simulations, as will be described in Section 4. We then need the operations of merging and splitting cells.

Merging cells means that four cells with a common father node are merged to one. The father node then becomes a leaf node, corresponding to the area of the four cells that were merged. The particle lists and the boundary segment lists of the merged cells are merged, and the the necessary pointers are updated.

Splitting a cell means that a leaf node becomes the father of four leaf nodes. First, the pointers in the quadtree are updated, then the particle list is split with respect to the position of the particles. Similarly, boundary segment lists for the new leaf nodes are set up from the list of their father, including only segments that are within reach from each of the new cells.

We shall now consider the case of coupled particle-fluid flows where the particle trajectories and the fluid flow are calculated separately using some time step dt. The mutual interaction between the particles and the fluid is taken into account by transferring information between the particles and fluid between every time step. It is necessary to divide the calculation period into very small time steps dt to obtain accurate and reliable solutions. Sigurgeirsson et al. [7] suggested the use of a *billiards* algorithm within each time step and thus rebuilding the event queue at the beginning of each time step. A similar framework was also used by Hoomans et al. [2].

Sigurgeirsson et al. [7] identified the following three primary data structures:

1. **The particle information**. Arrays with information about the particles, their positions, velocities, size, etc.
2. **The event queue** which is a collection of events, each with an event time and information to carry out the event. In this queue, a maximum of one event is stored per particle. Possible events are:
   - A collision with another particle.
   - A wall collision.
   - A transfer (the particle moves to a neighbour cell).
   - A check (the particle is to be checked for new events). This is a trick introduced by Sigurgeirsson et al. [7] for handling the double entry of collisions in the event queue and for labelling the third party if the new collision partner after a collision or a transfer already is scheduled.
3. **The cell structure** which contains information to make a neighbour-particle list in an efficient way.

The algorithm for performing one time step dt reads:

S1. Initial operations. These operations are for instance adding and initialising new particles, if there are new particles to add. If the cell structure is chosen to be adaptive, any splitting or merging of cells is carried out before the event queue is built.
S2. Calculate the forces on the particles.
S3. Calculate the times to the first collision and the first transfer and build the event queue of the events that are scheduled within the time step. A detailed description is given in Section 2.3.1.
S4. Until the event queue is empty, we carry out the following loop:
   L1 Handle event. Details are given in Section 2.3.2.

L2 Update the event queue if the new events are scheduled within the time step dt.
L3 Return to step L1.

S5. Update particle positions at $t = t_0 + dt$.
S6. Calculate information for the fluid code.

2.3.1 Build event queue

In the general case of a particle-laden flow, the particle positions and velocities and the fluid velocity are updated at each time step. New collision and transfer times are calculated and the event queue is built from the bottom. The particles-lists are updated during the simulation and are therefore, except at the first time step, up-to-date.

As outlined in Section 2.2.2, a loop is made over all cells, and neighbour particle lists are made. From the neighbour-particle lists we are able to find:

- Time to transfer from the present cell to one of the neighbouring cells by using the quadtree structure.
- Time to collision with the wall by using the list boundary elements belonging to the particle cell.
- Time to collision with the particles in the neighbour particle list, given by Eq. (9).

Both the time to transfer and the least of the time to wall collision and the time to particle collision are stored. For administration of the event queue, the least of the time to transfer and the time to collision is stored separately.

A binary heap structure is used to find the first event to process and the following sequence of events. The heap is an efficient data structure for priority queues and is elegant and easy to maintain, Ref. [14, p.138]. The heap is also recommended by Lubachevsky [6] and Sigurgeirsson et al. [7]. The length of the queue is limited to the number of particles, but we restrict this further by only scheduling the events that will happen within the time step dt.

2.3.2 Handle events

The operation *Handle event* includes finding the first event to handle, executing the operations given by the event type, and searching for new events.

First, the time to transfer and the time to collision are compared. If the time to transfer is less than the time to collision, a transfer is handled. Otherwise, the next event type could be either a collision, a wall collision, a check, or a removal. The actions of the different events are:

Transfer :

- The particle is removed from the old cell's particle list and added to the new cell's particle list.
- A neighbour-particle list is made by using the information from the new neighbour cells only, see Section 2.2.2
- The particle is checked for potential new collisions and transfers.

Particle-particle collision :

- The positions and velocities of involved particles are recalculated.
- The updated positions and velocities are checked. A 'touch', but no collision is possible.
- In case of a collision, the velocities immediately after the collision, given by Eq. (5) and Eq. (6), are computed.
- The forces on the particles are recalculated.
- The collision partner event type is changed to *check*.
- A neighbour particle list is made from the information in all neighbour cells and the cell itself, and the particle is checked for potential new collisions and transfers.

Wall collision :

- The particle position is recalculated.
- The updated positions and velocities are checked.
- In case of a collision, the boundary properties are checked. The boundary could be:
  1) A solid wall.
  2) A flow boundary, the particle is then removed.
  3) A flow boundary with a 'particle proof grating', which behaves like a solid wall for the particles.
- The particle velocity immediately after the collision, given by Eq. (7), is calculated.
- The forces on the particle are recalculated.
- The neighbour-particle list is made from the particle lists belonging to all neighbour cells and the cell itself and the particle is checked for potential new collisions and transfers

Check :

- The particle in question is checked for collisions and transfers. This requires that a neighbour particle list be made from the particle lists belonging to the neighbour cells in all directions, and the cell itself.

2.3.3 Computational complexity

We shall now analyse the complexity of the proposed algorithm. The computational complexity of the time-consuming operations is estimated and presented in Table I.

To find a neighbour-particle cell in the quadtree, different neighbour-finding algorithms are used, depending on the direction. Computational complexity for these basic algorithms are given in Ref. [12], as bounds to the average number of nodes visited. The bound averages are constant (the largest is $6\frac{2}{27}$ in the case of a corner-corner-neighbour) and therefore of $O(1)$. To check for transfers after a collision event, the position of the cell containing the colliding particle is needed (it is not stored). This requires an ascent to the quadtree root, and therefore the work is of $O(\log n_{\text{cell}})$. This term is avoided with a raster system. It is of the same order as other terms, though.

For a particle-laden flow, the calculation period is divided into n_t time steps, as mentioned earlier. The event queue is rebuilt at the beginning of each time step, which requires a traversal through all n_{cell} cells. For each cell, there is a cost of finding the forward neighbours $O(1)$ and the collision times with the neighbouring particles $O(n_{\text{nb}}^2)$. Here the cell position is kept track of, so an ascent to the root is not necessary. The cost of building the heap structure is $n_h \log n_h$, where n_h is the number of events in the heap. Because we only store one event per particle, the maximum number of elements in the heap is limited to the number of particles, n.

During a simulation, n_{tot} events will happen. n_{tot} is the sum of particle-particle collisions n_c, particle-wall collisions n_{wc}, and transfers n_{tr}. For each event there are costs related to extracting the event and updating the heap $O(\log n_h)$, finding the cell position $O(\log n_{\text{cell}})$, and checking neighbour particles for new collisions, $O(n_b)$. Since the number of particles per cell typically is of order 1 or less, we assume that

$$n_{\text{cell}} \approx n. \tag{10}$$

It follows from Eq. (10) that the number of neighbour particles n_{nb} is independent of the number of particles n. If the particles are uniformly spatially distributed, the number of neighbour particles is of the same order as the number of neighbouring cells, $O(1)$.

Further, we assume that the number of particle-particle collisions is much larger than the number of particle-wall collisions $n_c \gg n_{\text{wc}}$. When a collision happens, the event type of the collision partner is changed to a check. If the new collision partner after a collision, transfer, or check has a third party as its collision partner, the event type of the third party is

Table I
Overview of the computational work.

Operation	Cost	Calls	Complexity
Build event queue			
Find forward neighbours	$< 4 \cdot 6\frac{2}{27}$	$n_t \cdot n_{\text{cell}}$	$O(n_t \cdot n_{\text{cell}})$
Find cell position	$\log n_{\text{cell}}$	$n_t \cdot n_{\text{cell}}$	$O(n_t n_{\text{cell}} \cdot \log n_{\text{cell}})$
Search for collision partners	n_{nb}^2	$n_t \cdot n_{\text{cell}}$	$O(n_t n_{\text{cell}} \cdot n_{\text{nb}}^2)$
Build heap	$n \log n$	n_t	$O(n_t \cdot n \log n)$
Handle event			
Extract first event and update queue	$\log n$	n_{tot}	$O(n_{\text{tot}} \log n)$
Transfer	n_{nb}	n_{tr}	$O(n_{\text{tr}} \cdot n_{\text{nb}})$
Collision	n_{nb}	n_c	$O(n_c \cdot n_{\text{nb}})$
Wall collision	n_{nb}	n_{wc}	$O(n_{\text{wc}} \cdot n_{\text{nb}})$
Check	n_{nb}	n_{ch}	$O(n_{\text{ch}} \cdot n_{\text{nb}})$
Find cell position	$\log n_{\text{cell}}$	n_{tot}	$O(n_{\text{tot}} \cdot \log n_{\text{cell}})$

changed to a check. However, according to Sigurgeirsson [15, p. 22] and our own results, the number of checks n_{ch} is of the order the number of collisions n_c. Therefore, the total number of events can be written as:

$$n_{\text{tot}} \approx 2n_c + n_{\text{tr}} \tag{11}$$

We sum the last column in Table I and insert the assumptions in Eq. (10) and Eq. (11). The complexity of the algorithm can then be written as:

$$O\left(n_t n \log n\right) + O\left((n_c + n_{\text{tr}}) \log n\right) \tag{12}$$

3 Billiards

By 'billiards' we understand a system of n chaotically colliding hard spherical particles. Each particle moves along straight lines until it collides with another particle or a wall. The collision is instantaneous, hence the probability of more than two particles involved in the same collision is zero. Simulating billiards is a straightforward case that has been studied by many investigators, for instance Refs. [6,7]. We are not mainly interested in billiards, but the simplicity of the billiards case makes it well-suited for testing the algorithm.

The governing equations (1) and (2) for the particles can be rewritten as:

$$m_i \frac{\mathrm{d}\boldsymbol{u}_i}{\mathrm{d}t} = 0 \tag{13}$$

$$\frac{\mathrm{d}\boldsymbol{x}_i}{\mathrm{d}t} = \boldsymbol{u}_i, \tag{14}$$

where $i = 1, \ldots, n$. It follows from (13) and (14) that the particles move along straight lines with constant velocity between the collisions. This implies that the forward Euler scheme, given in Eq. (8), calculates the position and the velocity of the particles to machine precision. No energy is lost in the collisions, hence the restitution coefficients e and e_w are equal to unity.

3.1 Test method

It is of basic interest to know as much as possible about how the code works. We would like to know how the computational work scales with the number of particles and the number of particle-particle interactions. Further, we need to establish criteria for splitting and merging cells automatically in the adaptive cell structure. Within the frame of billiards simulations it should be possible to investigate:

A. How the CPU cost varies with the number of particles n and the number of collisions n_c.
B. The influence of multiple time steps.

This has been investigated as follows:

1. Billiards simulations with a single time step, where
   (i) The volume fraction α_p is constant and the number of particles varies for some volume fractions.
   (ii) The number of particles is constant and the volume fraction α_p varies.

   This addresses Item A.
2. Billiards simulations with several time steps, where
   (i) The volume fraction α_p and the number of time steps n_t are constant and the number of particles varies for some volume fractions.
   (ii) The number of particles n and the number of time steps n_t are constant and the volume fraction α_p varies.
   (iii) The number of particles ,n, and volume fraction ,α_p, are constant and the number of time steps varies. The physical simulation time is kept constant.

   This addresses Item B.

3.1.1 Test case setup

The billiards simulations were carried out in a two-dimensional box where the length of the sides was 1 m. The mean velocity in both directions was 0 m/s, while the root mean square of the velocity components was 1 m/s. Initially, the particle velocity was given a normal distribution. The particles were initially uniformly distributed within an inner square in the box, so there were no particles less than 10 cm from the walls at the start of the simulations.

Since the geometry is fixed, the radius of the particles has to change for different volume fractions α_p and numbers of particles n. A correlation between area fraction A_p and volume fraction α_p is given by $\alpha_p = \frac{2}{3}\frac{A_p}{A}$, where A_p is the particle area and A is the total area. The particle radius is then given by $R = \sqrt{\frac{3}{2}\frac{\alpha_p A}{n\pi}}$. An absolute requirement is that there be room for a particle entirely within a cell in the cell structure. This gives the minimum cell size and thus the maximum number of cells. In the case of uniform cell size, the number of cells n_{cell} in the quadtree is $n_{\text{cell}} = 4^N$, where $N + 1$ is the number of levels in the quadtree.

The test cases that have been carried out are described in Table II. The calculation period is three seconds and is divided into n_t time steps. Measurements of the CPU time were done by profiling the code on a Compaq Alpha 677MHz 21264a computer. It is not possible to both fully optimise and profile the code, but we assumed that the results regarding how the CPU time scales in different variables are unaffected by the optimisation level.

Table II
Test scheme.

Case	# particles n	Volume fraction α_p	n_{cell}	# time steps n_t
I	5000 – 200000	1%, 5%, and 15%	$4^4 - 4^{10}$	1
II	5000 – 200000	1%, 5%, and 15%	$4^4 - 4^{10}$	30000
III	50000	0.1% – 15%	$4^7 - 4^9$	1 and 30000
IV	50000	5%	$4^7 - 4^9$	1 – 30000

3.2 Results

First, we ensured that we got the same results with a fine particle cell structure resolution as with a naïve n^2 search for collision partners in the entire domain. The velocity distributions at the end of the simulation

time were compared and we found that the velocity distributions were equal within sampling variations and that they corresponded to Maxwell distributions.

To find the best cell size for a given number of particles n and a given volume fraction α_p, we ran each case with different numbers of particle cells. Because the number of cells is given by $n_{\text{cell}} = 4^N$, the cell number steps are coarse. We did therefore not find *the optimal cell size*. However, we belive that the average of the best cell size for several numbers of particles may give us an idea of what the optimal cell size is.

Particle simulations, where the position of a particle is calculated only when the particle is involved in a collision, are *event-driven*. According to Sigurgeirsson et al. [7] the CPU cost for event-driven billiards simulations could be written as:

$$\text{CPU time} = a\,(n_c \cdot \log n)^{b_c}\,, \tag{15}$$

where b_c should be close to unity for an efficient algorithm.

On the other hand, if we assume that the first term in Eq. (12), related to the search for collision partners and building of the event queue, dominates the computational cost, the CPU time will scale linearly in $n \log n$. Then we could write the CPU time as:

$$\text{CPU time } = a(n \log n)^{b_n}, \tag{16}$$

where b_n should be close to unity for an efficient algorithm. Because of the dependency of the number of time steps, the simulations are *time-driven*, but within each time step the algorithm is still event-driven.

3.2.1 Case I – Plain billiards

Case I is billiards where the time step dt is equal to the entire calculation period. The simulations were therefore event-driven and the CPU time should scale as $n_c \log n$. Figure 4 shows the CPU time versus $n_c \log n$ for particle volume fraction $\alpha_p = 1\%$ according to Case I in Table II. The best fit of the form given by Eq. (15) has b_c=1.15. Table III summarises the simulations for each volume fraction. The exponents b_c are close to one for all cases and the average number of particles per cell is between 0.54 and 0.74 for different volume fractions α_p. The average number of transfers per collision n_{tr}/n_c varies from 1.33 for $\alpha_p = 15\%$ to 6.75 for $\alpha_p = 1\%$.

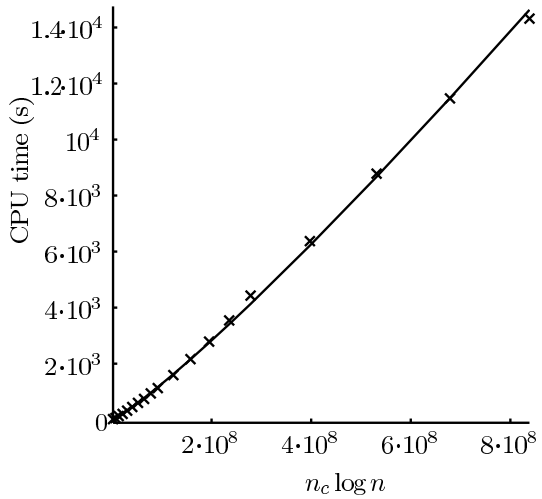


Figure 4. Plain billiards. Particle volume fraction $\alpha_p = 1\%$. CPU time versus $n_c \log n$. The best fit of the form $a(n_c \log n)^{b_c}$ has $b_c = 1.15$.

Table III
Summary of Case I - Plain billiards.

α_p	n_t	b_c	Average n/n_{cell}	Average n_{tr}/n_c
1%	1	1.15	0.74	6.75
5%	1	1.14	0.54	3.14
15%	1	1.17	0.57	1.33

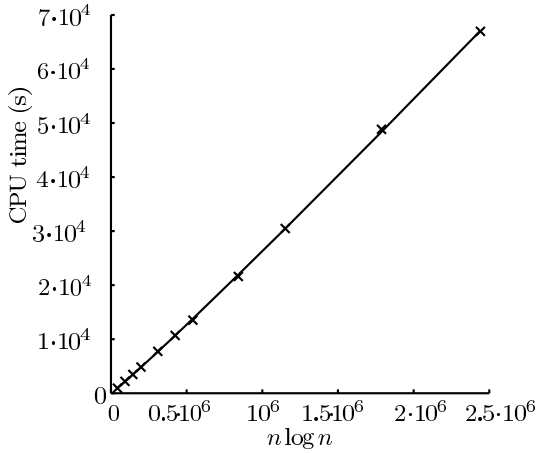


Figure 5. Billiards with time stepping. Particle volume fraction $\alpha_p = 1\%$. CPU time versus $n \log n$. The best fit on the form $a(n \log n)^{b_n}$ has $b_n = 1.05$.

3.2.2 Case II – Billiards with time stepping

Case II is equal to Case I, except that the calculation period was divided into 30000 time steps of 10^{-4} s. Figure 5 shows the CPU time versus $n \log n$ for 1% particle volume fraction α_p. The best fit on the from given by Eq. (16) has b_n=1.05. Table IV summarises the results. b_n is close to unity for all cases, although a slight increase is seen for the 15% volume fraction case. A linear scaling in $n \log n$, indicates that the first term in Eq. (12) is larger than the second term, hence the simulations were time-driven. The average number of particles per cell n/n_{cell} for is within the range $n/n_{\text{cell}} \in (0.79, 1.02)$, which is slightly higher than with a single time step. As in Case I, the average number of transfers per collision n_{tr}/n_c varies considerably with the volume fraction.

Table IV
Summary of Case II - Billiards with time stepping.

α_p	n_t	b_n	Average n/n_{cell}	Average n_{tr}/n_c
1%	30000	1.05	1.02	5.76
5%	30000	1.06	1.02	2.30
15%	30000	1.12	0.79	1.13

3.2.3 Case III and IV

Figure 6 shows the results for 50000 particles where the volume fraction α_p varied from 0.1% to 15%, hence the number of collisions n_c also varied. The linearity between the CPU time and the number of collisions n_c is

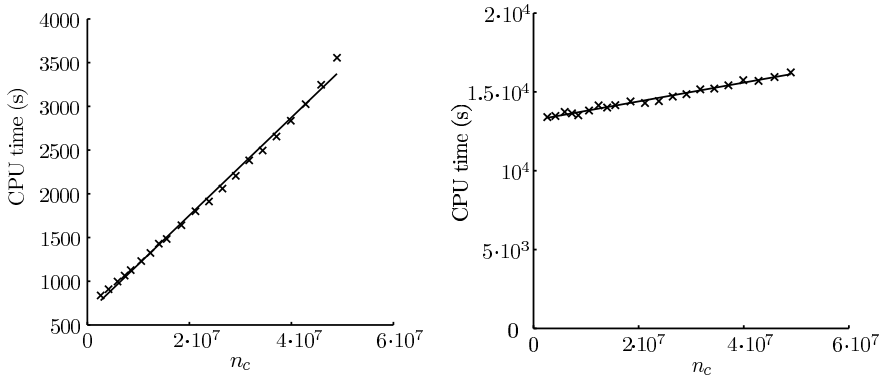


Figure 6. CPU time for constant number of particles, $n = 50000$. CPU time versus number of collisions. The best fit on the form $a_0 + a_1 \cdot n_c$ have $a_1 = 5.60 \cdot 10^{-5}$ s for $n_t = 1$ and $a_1 = 5.96 \cdot 10^{-5}$ s for $n_t = 30000$.

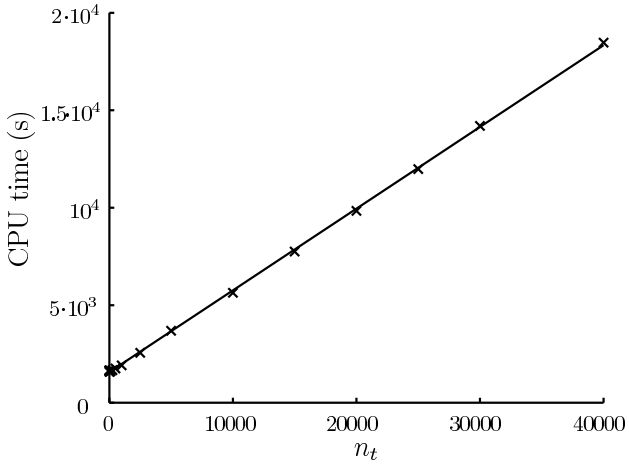


Figure 7. Billiards with time stepping and a constant number of particles, $n = 25000$ and constant volume fraction $\alpha_p = 5\%$, hence the number of collisions n_c is constant. CPU time versus number of time steps n_t.

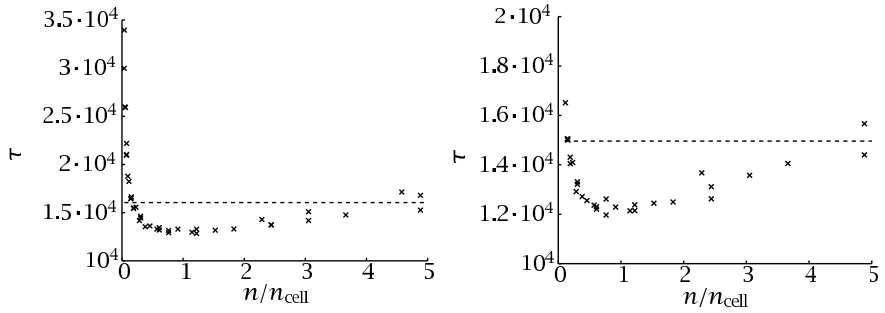


Figure 8. Normalised CPU time τ versus particles per cell ratio n/n_{cell} for α_p=1% and α_p= 5%. The dashed line indicates a level 25% above the minimum value of τ.

clear in the tested range, both without and with time stepping. One also sees from Fig. 6 that recalculation of the event queue every time step takes most of the calculation time when many time steps are carried out.

Figure 7 shows the CPU time versus the number of time steps, Case IV, where the calculation period is fixed and the number of particles and the number of collisions are constant. The linearity between CPU time and the number of time steps n_t is evident.

3.3 Optimal cell size

Based on the results from the simulations carried out above, we will now try to find the optimal number of particles per cell. The number of particles per cell is nearly constant for the best particle cell structures and the simulations are time-driven for small time steps. We therefore use the results given in Table IV to 'normalise' the CPU time by making use of the best fit solution the following way:

$$\tau = \frac{\text{CPU time}}{(n \log n)^{b_n}}, \tag{17}$$

where b_n is given in Table IV. Figure 8 shows the normalised CPU time τ versus the particles-per-cell ratio n/n_{cell}, where all simulations with a particles-per-cell ratio below five are shown. The dashed line indicates where the computational cost is 25% above the minimum value of τ. Figure 8 indicates that the optimal number of particles per cell n/n_{cell} is of order of 0.2 – 4.0.

4 Adaptive cell structure

In this section, we describe a simple strategy for locally adapting the cell size to variations in particle density. The cell size may then vary in space and with time. This applies to cases of particle-laden flows, since it is not practical to change the cell sizes at times other than those where the event list is recalculated, at intervals dt. Cell sizes need not be checked at every dt; for a given class of problems, the adaptation frequency can be determined from experience to give the best reduction of CPU time.

The adaptation strategy is based on the results regarding the optimal number of particles per cell in the previous section. Figure 8 shows that the computational cost is acceptable for a particles-per-cell ratio n/n_{cell} within the range $n/n_{\text{cell}} \in (0.2, 4.0)$. These numbers are averages. Considering single cells and observing that a cell can only be split in four cells and vice versa, we propose the following strategy:

- Split a cell in four if it contains 3 particles or more.
- Merge four cells that have the same father node if they together contain 0 or 1 particle.

For cells that contain particles, this will give an average number per cell within the range given above. To carry out this strategy, it is necessary to count the particles in each cell when the update is to be considered.

4.1 Update interval

With the criterion for splitting and merging of cells given above, we have to find how often the adaption should be done and to check how the computational work is affected. This is done by running billiards simulations similar to those described in Section 3. We found that updating the particle cell structure every time has an acceptably low cost, less than 10% above that of a fixed cell structure, for a tested number of particles between 5000 and 50000.

The simplicity of updating the particle cell structure every time step makes it attractive: We may use the instantaneous particle distribution directly and it works for all kind of particle flows with small time steps.

4.2 Description of the numerical experiments

Simulations of particles in a gravity field and a 90° bend were carried out with the adaptive cell structure and several fixed cell structures for each number of particles. The number of particles was in the range 5000 – 50000. The density of particles was kept constant and equal to 5% by varying the particle size.

In Fig. 9 and Fig. 11 we plot three curves:

Adaptive, that refers to the adaptive cell structure as described above.
Optimal, that refers to the fixed cell structure that gives the shortest CPU time. At least three simulations were run to find the best fixed cell structure.
Billiards, that refers to the cell structure that was the best fixed cell structure in the billiards case, Section 3.2.2, with equal number of particles and the same density of particles α_p. Billiards can therefore be regarded as a 'qualified guess' for cases where the cell structure not is known.

A fixed cell structure means that the cell size is given initially and that the cell size is uniform in the entire domain.

4.3 Particles in a gravity field

We ran a test case with particles falling in a gravity field, with the influence of the fluid neglected. This has many physical aspects in common with industrial applications, for instance the filling of a silo or a hopper. Initially, the particles were uniformly distributed in the domain and the initial velocities were zero. Due to the gravity, the particles accelerated towards the bottom of the box. Because energy was lost in the collisions, the particles gathered in the bottom of the box. The particle-particle restitution coefficient e and the particle-wall restitution coefficient e_w were chosen to be 0.9 and 0.99, respectively. The simulations were carried in a square box with side lengths 1 m. Initially, the particles were uniformly distributed, as in the billiards case, and the velocity of the particles was zero. 20000 time steps of 10^{-4} s each were computed.

Figure 9 shows the CPU time versus number of particles n for two fixed cell structures and the adaptive cell structure. A small difference between the optimal fixed cell structure and the adaptive cell structure is observed, only slightly in favour of the adaptive cell structure. A significantly higher computational cost was obtained with the cell structures that were optimal for the billiards simulations, which had a coarser resolution

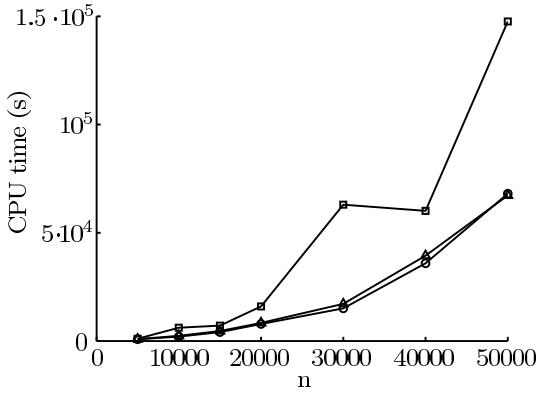


Figure 9. CPU time versus number of particles n for particles in a gravity field. The CPU-time consumption of the adaptive cell structure (O), the case-optimal fixed cell structure (△), and the fixed cell structure that was best in billiards case (□) are compared.

than the optimal cell structure in the sedimentation case.

4.4 Bend

90° bends appear in many processing applications, for instance in pneumatic transport systems. This case will show how the algorithm works in complex geometries where the particles are bound to be in a small part of the area spanned out by the extreme values of the geometry.

The bend with outer radius 1 m and inner radius 0.95 m is shown in Fig. 10. The particles entered the bend at the left boundary with a mean velocity of 4 m/s in the x direction and 0 m/s in the y direction. The velocities were Gaussian distributed with a standard deviation of 0.1 m/s in both x and y direction. During the particles' travel through the bend, only gravity and collision forces were acting on the particles. The restitution coefficients e and e_w were 1.0.

The physical simulation time was 2.0 s, with 20000 time steps of 10^{-4} s each. Initially, no particles were inside the bend. The particles started entering the bend when the simulations started and the number of particles reached a steady state after 0.4 s.

Figure 10 shows how the cell structure adapted to the regions with a high density of particles. The quadrant at bottom left is not divided while the densely packed regions have a cell width of the same order as the particle diameter.

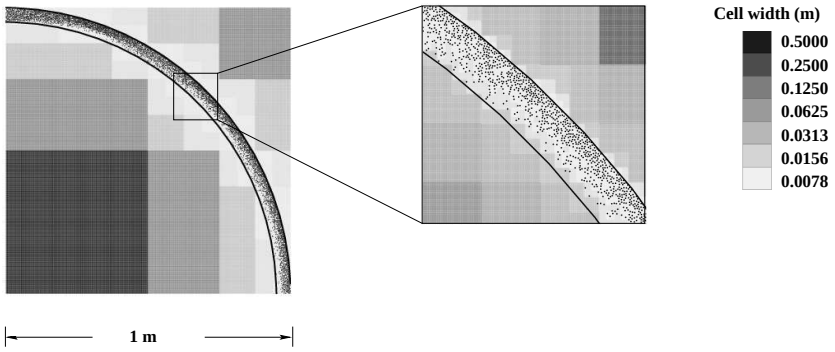


Figure 10. Cell structure adapted to bend.

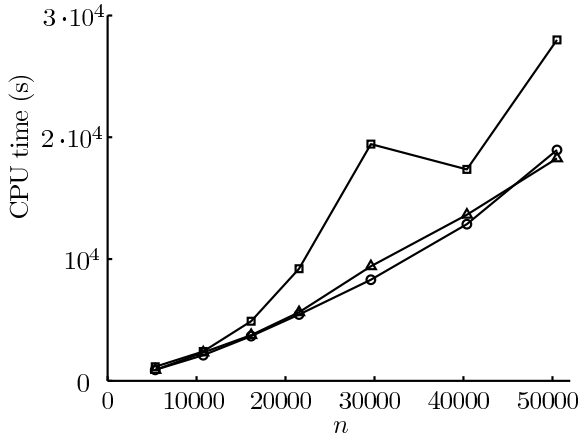


Figure 11. CPU time versus number of particles n for a 90° bend. The CPU-time consumption of the adaptive cell structure (O), the case-optimal fixed cell structure (△), and the fixed cell structure that was best in billiards (□) are compared.

Figure 11 shows the CPU time versus number of particles n for two fixed cell structures and the adaptive cell structure. The best fixed cell structure and the adaptive cell structure are equally fast. The adaptive cell structure was more efficient than the best fixed cell structure for 10000 to 40000 particles, but when 50000 particles are computed, the fixed cell structure is fastest. The reason is probably a shift in the maximum resolution for the adaptive cell structure, which increases the number of levels in the quadtree by one. As for the simulations of sedimenting particles, the best fixed cell structure resolutions for the bend are not those that were best for billiards.

5 Discussion

An adaptive algorithm for efficient detection of neighbouring particles in the frame of the Eulerian-Lagrangian approach is presented and tested. The algorithm uses a quadtree structure in two dimensions and the cell size may vary in space and time. Attention is also paid to link piecewise linear segments in the arbitrary shaped flow boundary to the particle cell structure.

Billiards simulations were done to ensure that the proposed algorithm is close to optimal for simple cases. From the simulations with uniform particle distribution in space (billiards), criteria for splitting and merging cells was established. We found that the number of particles per cell should be in the range of 0.25 to 3.0 for efficient computations, independent of the number of particles and the density of particles. In the adaptive cell structure, cells are split and merged between each time step to keep the number of particles within the proposed range. Two cases: particles in a gravity field and in a 90° bend, have been simulated with the new adaptive algorithm and the computational time has been compared with the simulations with a fixed cell structure. The best fixed cell structure and the adaptive cell structure seemed to be equally fast. In cell structures with uniform cells, many cells may be in regions with few or no particles, but the cost of visiting an empty cell is small. The computational costs are mainly due to calculations of potential collisions with neighbour particles, that is minimised by finding the cell size that minimise the product of the number of transfer and the number of neighbouring particles squared. It is not straightforward to find the best fixed uniform cell size. In order to find the 'optimal' fixed cell structure in Fig. 9 and Fig. 11 there was run at least three simulations for each number of particles. This means that the total consumed CPU-time for the best fixed cell-structure is much larger than the total CPU-time needed by the adaptive cell structure. For cases where the optimal fixed cell structure not is known *a priori*, a cell size

based on the knowledge about billiards simulations or from the literature might be a proper choice. In the presented test cases we observe that the fixed cell structure that was best for billiards is up to three times slower than the adaptive cell structure.

6 Conclusion

In this paper we have presented and tested an algorithm that uses an adaptive quadtree structure for neighbour particle detections. We have found that:

- A quadtree structure provides a flexible particle cell structure for efficient neighbour-particle detection that also handles complex flow boundary geometries.
- The criterion for splitting and merging cells should be based on the number of particles per cell.
- A number of particles per cell between 0.25 and 3 is reasonable and establishes suitable lower and upper limits for the presented automatic adaptive cell structure.
- The advantages of an adaptive algorithm appear in flows where the particles are not uniformly distributed, in complex geometries, and otherwise where information about the best cell size is not known *a priori*. Several simulations to find the optimal cell size resolution is avoided by using an adaptive algorithm.

Acknowledgements

The authors thank Inge Røinaas Gran and Svend Tollak Munkejord for reading and commenting the manuscript.

This work has been sponsored by the Research Council of Norway.

References

[1] Y. Tsuji, T. Kawaguchi, T. Tanaka, Discrete particle simulation of two-dimensional fluidized bed, Powder Technology 77 (1) (1993) 79–87.

[2] B. P. B. Hoomans, J. A. M. Kuipers, W. J. Briels, W. P. M. van Swaaij, Discrete particle simulation of bubble and slug formation in a two-dimensional gas-

fluidised bed: A hard-sphere approach, Chem. Eng. Sci. 51 (1) (1996) 99–118.

[3] E. Helland, R. Occelli, L. Tadrist, Computational study of fluctuating motions and cluster structures in gas-particle flows, Int. J. of Multiphase Flow 28 (2002) 199–223.

[4] M. Sommerfeld, Analysis of collision effects for turbulent gas-particle flow in a horizontal channel: Part I. particle transport, International Journal of Multiphase Flow 29 (2003) 675–699.

[5] B. J. Alder, T. E. Wainwright, Studies in molecular dynamics. I. General method, Journal of Chemical Physics 31 (2) (1959) 459–466.

[6] B. D. Lubachevsky, How to simulate billiards and similar systems, Journal of Computational Physics 94 (1991) 255–283.

[7] H. Sigurgeirsson, A. Stuart, W.-L. Wan, Algorithms for particle-field simulations with collisions, Journal of Computational Physics 172 (2001) 766–807.

[8] B. C. Vemuri, L. Chen, L. Vu-Quoc, X. Zhang, O. Walton, Efficient and accurate collision detection for granular flow simulation, Graphical models and image processing 60 (1998) 403–422.

[9] D.-J. Kim, L. J. Guibas, S.-Y. Shin, Fast collision detection among multiple moving spheres, IEEE Transactions on visualization and computer graphics 4 (3) (1998) 230–242.

[10] H. Samet, Application of spatial data structures, Addison-Wesley, Reading, Massachusetts, 1990.

[11] I. Gargantini, An effective way to represent quadtrees, Communications of the ACM 25 (12) (1982) 905–910.

[12] H. Samet, Neighbor finding techniques for images represented by quadtrees, Computer Graphics and Image Processing 18 (1982) 37–57.

[13] G. Schrack, Finding neighbors of equal size in linear quadtrees and octrees in constant time, CVGIP: Image understanding 55 (3) (1992) 221–230.

[14] T. A. Cormen, C. E. Leirserson, R. L. Rivest, C. Stein, Introduction to Algorithms, 2nd Edition, The MIT Press, Cambridge, 2001, ISBN 0-07-013151-1.

[15] H. Sigurgeirsson, Particle-field models: Algorithms and applications, Ph.D. thesis, Standford University (Nov. 2001).

Paper C

Cluster integration method in Lagrangian particle dynamics

Melheim, J. A.
Computer Physics Communications 171 (2005) 155-161

Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computer Physics Communications 171 (2005) 155–161
Computer Physics Communications
www.elsevier.com/locate/cpc

Cluster integration method in Lagrangian particle dynamics

Jens A. Melheim *

Norwegian University of Science and Technology (NTNU), Department of Energy and Process Engineering, Kolbjørn Hejes veg 1A, NO-7491 Trondheim, Norway

Received 4 November 2004; received in revised form 20 February 2005; accepted 5 May 2005

Available online 13 June 2005

Abstract

An efficient and robust approach is proposed in order to conduct numerical simulations of collisional particle dynamics in the Lagrangian framework. Clusters of particles are made of particles that interact or may interact during the next global time-step. Potential collision partners are found by performing a test move, that follows the patterns of a hard-sphere model. The clusters are integrated separately and the collisional forces between particles are given by a soft-sphere collision model. However, the present approach also allows longer range inter-particle forces. The integration of the clusters can be done by any one-step ordinary differential equation solver, but for dilute particle systems, the variable step-size Runge–Kutta solvers as the Dormand and Prince scheme [J. Comput. Appl. Math. 6 (1980) 19] are superior. The cluster integration method is applied on sedimentation of 5000 particles in a two-dimensional box. A significant speed-up is achieved. Compared to a traditional discrete element method with the forward Euler scheme, a speed-up factor of three orders of magnitude in the dilute regime and two orders of magnitude in the dense regime were observed. As long as the particles are dilute, the Dormand and Prince scheme is ten times faster than the classical fourth-order Runge–Kutta solver with fixed step size.

PACS: 02.70.Ns; 45.50.-j; 47.11.+j

Keywords: Discrete element method; Lagrange simulations; Soft-sphere dynamics; Particle-cell structure; Runge–Kutta; Variable time-step

1. Introduction

Computer simulations of particle-laden flows, where the Newtonian equations of motion of n spherical particles are solved numerically, constitute a useful framework for computation of many industrial applications. A time-consuming part of 'discrete element method' (DEM) simulations where every particle is tracked, is the calculations of the collisional forces. The collisional forces may result in a change in the velocity of the particles involved in the collision and a collision can last for a very short period of time. There are two main approaches for handling collisions, the *hard-sphere* approach [2], where the collisions are handled as an instantaneous change of velocities, and

* Tel.: +47 41215517; fax: +47 73592889.
E-mail address: jens.a.melheim@ntnu.no (J.A. Melheim).

0010-4655/$ – see front matter
doi:10.1016/j.cpc.2005.05.003

the *soft-sphere* approach [3], where the actual forces acting during the collision are modeled. The advantage of the hard-sphere approach is the numerical efficiency in dilute systems. In cases were the interaction with the fluid is neglected, the position and the velocity of a particle are updated only when the particle is involved in a collision, so-called *event-driven* simulations. Unfortunately, the hard-sphere approach fails in semi-dense and dense systems, when the assumption of instantaneous binary collisions is invalid and the *inelastic collapse* occur [4]. The soft-sphere model does always work, but one has to use very small time-steps [5].

In present work, the numerical efficiency of the hard-sphere approach is combined with the robustness of the soft-sphere approach. The particles within a *cluster* of close neighboring particles are integrated simultaneously, while each cluster is integrated separately, using different time-steps. The time-step size is decided by the estimated error, by using an embedded Runge–Kutta (RK) method [1]. However, one is free to choose another one-step ordinary differential equation (ODE) solver. The presented algorithm is quite similar to that of Reed and Flurchick [6] designed for molecular dynamics. Reed and Flurchick grouped the molecules in monomers and polymers and achieved a speedup up to a factor 7 compared to a Verlet algorithm [7]. The monomers (molecules with no close neighbors) are moved along straight trajectories until the collision radius r_{coll} hits the collision radius of another molecule or a group of molecules. The polymers are calculated with a Verlet algorithm. A main difference between the algorithm of Reed and Flurchick and the algorithm presented here is the way single particles are handled. In the presented algorithm, particles that travel far from other particles are integrated with the same ODE solver as the particles within clusters, instead of being moved along straight trajectories. It is therefore possible to handle particle-fluid forces and far-field forces acting on single particles in a proper way.

The proposed algorithm is tested on 5000 particles settling in a box. As long as most of the particles are flying around in the box, the variable step size ODE solver is faster than the classical fourth-order Runge–Kutta method (RK4) with fixed time-steps [8]. During the first time-steps a speed-up factor 10 is observed. When most particles are at rest at the bottom of the box, the variable step and fixed step solver are about equally fast. Compared to a soft-sphere algorithm, that uses a cell method for finding neighbors and the forward Euler scheme, the present approach is superior.

2. Cluster integration method

The basic idea is to control the error of the particle simulation by varying the time-step, for instance by employing an embedded Runge–Kutta method with self-adjusting step-size [8]. For a given maximum error, the greatest allowed time-step for integration of the particles varies over several orders of magnitude, depending on the actual forces working on the particles. To save computational time, the particles should be integrated with the largest possible time-step. Therefore, only particles that interact or may interact during the next global time-step Δt are integrated simultaneously. Particles that are far from other particles can be integrated alone.

Clusters of particles that are very close are formed, and the particles in a cluster are integrated simultaneously one global time-step Δt. A particle-cell structure [2] is used to detect neighbor particles and wall segments. It is an absolute requirement that there be room for a particle and its close neighborhood, the virtual radius, in each cell in the cell structure, as it is in Fig. 1. Sigurgeirsson et al. [9] found that the number of cell should be of the same order as the number of particles for efficient computations. A list of particles (`cpl`) and a list of wall segments (`cwl`) are connected to each cell, as sketched in Fig. 2.

As shown in Fig. 1, only particles that at the beginning of a global time-step have overlapping virtual radii r_v, the dashed circles, are considered as close neighbors. Close particles are added to each others close neighbor particle list (`pnpb`). In Fig. 1, the black particle A has initially the close neighbors B, C, and D. The shaded particles are also within the checked cells when particle A is evaluated, but the virtual radii of these particles do not overlap the virtual radius of particle A. However, the non-close neighbors are checked for a potential collision during the time-step by a *test move*. By doing a test move, one time-step is executed with the forward Euler scheme, as illustrated in Fig. 3(left). During the test move potential collisions with other particles are detected and transfers to

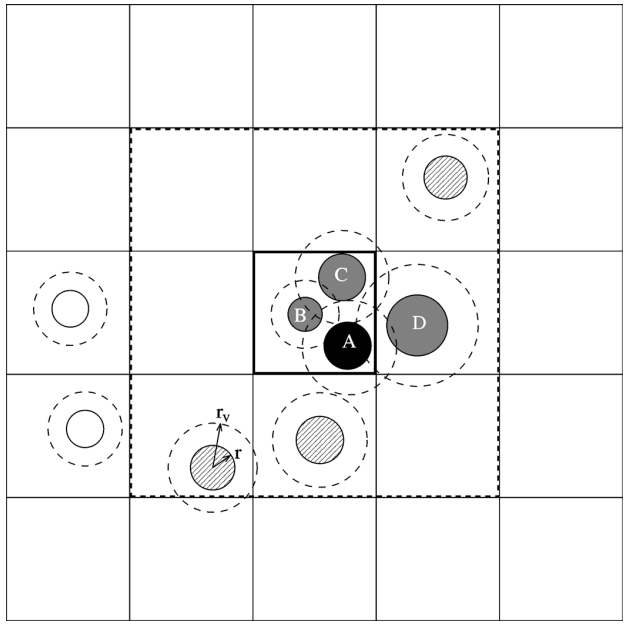


Fig. 1. Particles in a particle-cell structure at the beginning of a global time-step, where particle A, B, C, and D form a cluster.

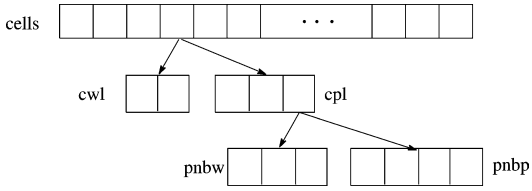


Fig. 2. The relationships between the list of cells (`cells`), the cell-particle list (`cpl`), the cell-wall list (`cwl`), the close particle (`pnbp`), and close wall (`pnbw`) lists.

a neighboring particle-cells are handled. A potential particle–particle collision is detected when the virtual radii of two particles collide during a test move, as illustrated in Fig. 3 (right). The particles are then added to each others close particle lists (`pnbp`). If the centroid of a particle crosses a particle-cell boundary during the test motion, the particle is transferred to the particle list of the new cell and the particles in the new neighboring cells are checked for potential collisions, similarly to what is done in hard-sphere codes [9,10]. The queue of collisions/transfers is sorted by the time to collision/transfer estimated by the forward Eulers scheme. The entire particle has to move inside the checked area when the actual particle path is calculated in order to avoid instabilities. The virtual radius r_v is estimated similarly as the cut-off radius of Verlet [11]:

$$r_v - r \geqslant \Delta t v_c, \tag{1}$$

where r is the particle radius and v_c is a *critical velocity* related to the expected deviation from a straight lined trajectory, due to collisions and the ballistic behavior. The critical velocity v_c is typically of same order as the fluctuating particle velocity v'.

After the transfers and collisions are handled and the close-particles lists (`pnbp`) are complete, the clusters are assembled. First, duplicate entries are deleted from the close particles lists. Then the lists are traversed recursively, as illustrated by the solid arrows in Fig. 4. The first place in the close-particle lists (`pnbp`) is used to label already visited lists in order to avoid infinite loops.

Note that a particle can only be part of one cluster. Thereby it follows that the maximum number of particles in a cluster equals the number of particles n and that the maximum number of clusters $K_{\max}$ equals n. A list of length n contains the particles, where parti-

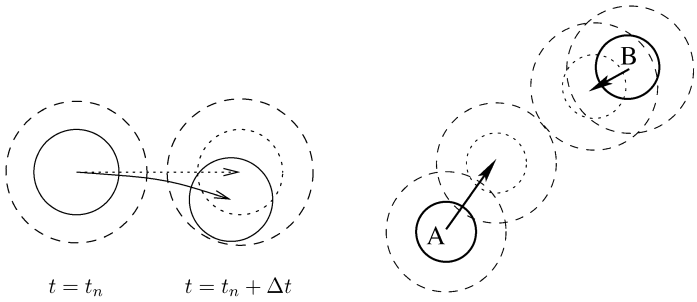


Fig. 3. (*Left*) Test motion, the dotted circles, to check the area the particle will move within. The solid line shows the accurately computed particle trajectory. (*Right*) A situation where the test motion predicts a collision during the time-step.

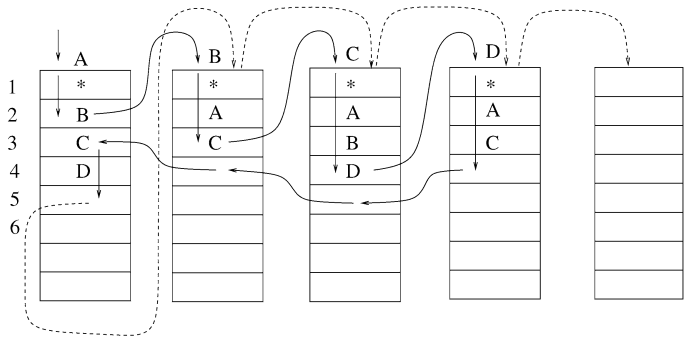


Fig. 4. The recursive algorithm working on the close particle lists that corresponds to the situation in Fig. 1.

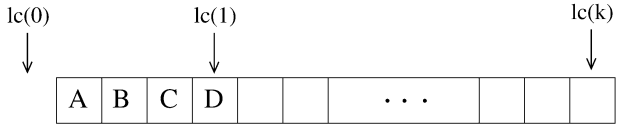


Fig. 5. The list of particles, sorted by cluster, where another list `lc` keeps the last particle in each cluster.

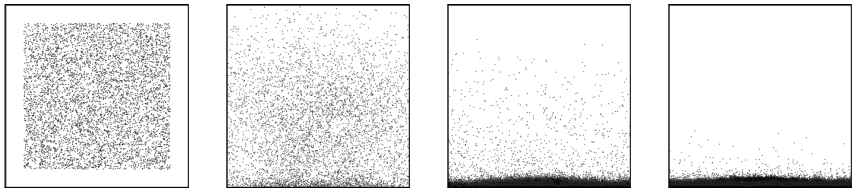
cles belonging to the same cluster are placed after each other. Another list (`lc`) keeps the position of the last particle in each cluster, as shown in Fig. 5.

The clusters, including the single-particle clusters, are now integrated over the global time-step Δt separately by using any one-step ODE solver. A significant reduction in the CPU time and an accurate solution is possible by employing an embedded Runge–Kutta scheme [8], that automatically adjusts the local time-step in order to keep the local error less than the given tolerances. When a fixed time-step ODE solver is employed, the time-step is calculated *a priori*, for instance by

$$\delta t = \min\left(\Delta t, \frac{1}{a} t_{c,\min}\right), \tag{2}$$

where $t_{c,\min}$ is the shortest collision period within the cluster and a is a chosen constant. The properties of the chosen ODE solver and the desired accuracy decide the value of a.

During the integration, the close particle lists are used for calculation of the inter-particle forces. The wall segments belonging to the particle cells that were visited by the particle during the test move are added



(a) (b) (c) (d)

Fig. 6. Particles settling in a box. (a) $t = 0.0$ s; (b) $t = 0.2$ s; (c) $t = 0.4$ s; (d) $t = 0.6$ s.

to the neighboring wall segment list (pnbw). The wall segments in pnbw are handled as the close-neighbor particles. After the integration, the position of the particles are checked against the particle-cell structure and the mesh for the calculation of the fluid equations. Due to the possible different results from the test move and the final computation, particles might be mapped to wrong particle cell. Incorrectly mapped particles are transferred to the correct particle cell.

A summary of the cluster integration algorithm reads:

(1) Add particles with overlapping virtual radii to their respective close-particles list.
(2) Perform a test motion with the forward Euler scheme and handle transfers. Add particles that will become close during the time-step to their respective close-particle lists.
(3) Trim the close-particle lists.
(4) Build the particle clusters.
(5) Integrate each cluster separately.
(6) Check new particle position versus the particle-cell structure.

The idea of particle clusters is to some extent similar to the *polymers* of Reed and Flurchick [6]. In the present work, contrary to Reed and Flurchick, a particle-cell structure is used, the clusters/polymers are rebuilt every global time-step, and one ODE solver is employed for all particles.

3. Description of the test cases

The performance of the algorithm was tested by the settling of 5000 spherical particles in a square two-dimensional box with edge lengths 1 m. The particle diameter was $d = 3.568$ mm and the density was $\rho = 2400$ kg/m^3. This gave an area fraction of particles $A_p/A = 5\%$. Initially, the particles were randomly placed within an inner square of 0.9×0.9 m, as shown in Fig. 6(a). The initial particle velocities were sampled from a normal distribution function with zero mean $\bar{v} = 0$ m/s and variance $\overline{v'^2} = 1$ m^2/s^2. The governing equations for the particles are written as:

$$\frac{d\boldsymbol{x}_i}{dt} = \boldsymbol{v}_i, \tag{3}$$

$$\frac{d\boldsymbol{v}_i}{dt} = \boldsymbol{g} + \frac{1}{m_i}\boldsymbol{F}_C, \tag{4}$$

where the contact forces $\boldsymbol{F}_{Cn}$ and $\boldsymbol{F}_{Ct}$ were modeled as [5]:

$$\boldsymbol{F}_{Cn} = (-k\delta - \eta \boldsymbol{v}_r \cdot \boldsymbol{n})\boldsymbol{n}, \tag{5}$$

$$\boldsymbol{F}_{Ct} = \begin{cases} (-\eta \boldsymbol{v}_r \cdot \boldsymbol{t})\boldsymbol{t}, & |\eta \boldsymbol{v}_r \cdot \boldsymbol{t}| \leqslant \mu_f |\boldsymbol{F}_{Cn}|, \\ -\mu_f |\boldsymbol{F}_{Cn}| \boldsymbol{t}, & |\eta \boldsymbol{v}_r \cdot \boldsymbol{t}| > \mu_f |\boldsymbol{F}_{Cn}|, \end{cases} \tag{6}$$

where the tangential spring has been neglected. The contact forces only take place when there is an overlap $\delta > 0$. $\boldsymbol{v}_r$ is the relative velocity, $\boldsymbol{n}$ and $\boldsymbol{t}$ are the normal and tangential unit vectors, η is the damping coefficient, and μ_f is the coefficient of friction. In the test case, the restitution coefficient e and collision time t_c were chosen to take the values 0.95 and $5 \cdot 10^{-6}$ s, respectively. The corresponding stiffness is $k = 1.12 \cdot 10^7$ N/m and damping coefficient is $\eta = 0.60$ N/ms. The particle-wall restitution coefficient e_w and stiffness k_w were chosen equal to the particle–particle parameters. A global time-step of $\Delta t = 10^{-4}$ s and a rather conservative virtual radius $r_v = 1.2r$ were used, which corresponds, according to Eq. (1), to a critical velocity $v_c < 3.57$ m/s.

An initial test with two particles approaching each other with a relative velocity $v_r = 2$ m/s was done to check the accuracy of the different ODE solvers, the forward Euler scheme, the classical RK4 scheme [8], and the Dormand and Prince 5(4) scheme [1]. The

error is defined by the reproduction of the specified restitution coefficient:

$$\text{error} = \frac{|(|v|/|v^0| - e)|}{e}, \tag{7}$$

where v^0 is the velocity before the collision. To compute a collision within a relative tolerance of 1%, the forward Euler scheme needs 500 sub-steps (500 evaluations of $\boldsymbol{F}_C$), RK4 needs 4 sub-steps (16 evaluations of $\boldsymbol{F}_C$) and the Dormand and Prince scheme needs 3 sub-steps (21 evaluations of $\boldsymbol{F}_C$). The parameter a in Eq. (2) equals the number of needed sub-steps for the respective ODE solver. In the case of a variating time-step, the shortest time-step was $t_{\min} = t_c/a$.

4. Results

Fig. 6 shows the situation after $t = 0.0$ s, $t = 0.2$ s, $t = 0.4$ s, and $t = 0.6$ s. Initially, the particles were spread over the entire area of the box and most of the particles could be integrated separately. At $t = 0.4$ s most particles were settled and they belonged to the same cluster. The measured CPU time shown in Fig. 7 reflects this behavior. The Dormand and Prince scheme with a relative tolerance of 10^{-4} and an absolute tolerance of 10^{-25} gave a speed-up factor of 10 the first 0.2 s compared with the fixed-step RK4 solver. The relative and absolute tolerance is defined by the relative and absolute difference between the 4th and the 5th order Rung–Kutta scheme embedded in the Dormand and Prince scheme. At $t = 0.4$ s, no speed-up was achieved by employing a variable step size ODE solver. The traditional soft-sphere method refers to a method that searches for neighbor particles in a fixed particle-cell structure and the motion of the particles is calculated with the forward Euler scheme. The cluster integration method with variable step size was up to three orders of magnitude faster than the traditional method, while RK4 with fixed time-step was about 100 times faster. The CPU time was measured by using *time* on a Compaq Alpha 677 MHz 21264a computer. Each case was run three times and the maximum relative standard deviation of the measured CPU time was $1.63 \cdot 10^{-2}$. Due to the very long computational time only the first 0.3 s were calculated with the traditional soft-sphere method. A potential platform dependency was also checked. The simulations were run once on a 32 bit Linux computer with the Portland Group F77 compiler and the trends were confirmed.

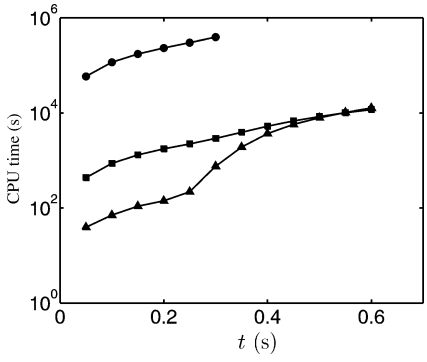


Fig. 7. CPU time versus elapsed time. CPU time consumed by the traditional soft-sphere approach (●) and the cluster integration method with the classical RK4 (■), and the variable time-step Dormand and Prince scheme (▲).

5. Discussion

An algorithm that gives efficient and accurate integration of the governing equations in Lagrangian particle dynamics is presented. The main ideas are that only particles that interacts are integrated simultaneously and that the integration of should be done with the largest possible time-step. By forming clusters of close particles and by integrating the clusters separately by a variable step-size embedded Runge–Kutta scheme, a significant speed-up can be achieved. A settling process is simulated and the CPU-time consumption is measured at different stages in the settling process. Initially, when the particles are flying around, a speed-up of factor of 10 is seen compared to a fixed time-step Runge–Kutta solver, that also uses the cluster integration method. When the particles are settled, most of the particles are in the same cluster, and the fixed and variable time-step approach is about equally efficient. In the selected case, where the collision time was quite short, the traditional soft-sphere method, that uses the particle-cell structure directly to form close-neighbor-lists and the forward Euler scheme, is very slow compared with the presented approach. There are three reasons for this: First, a drawback with the forward Euler scheme is the number of time-steps needed to compute a collision within a given toler-

ances. Second, the used soft-sphere code updates the neighbor-particles lists every time-step by traversing the particle-cell structure, that has a significant costs. Third, the number of particles to check for potential collisions is reduced by applying a close-particle-list in addition to the particle-cell structure.

Besides a significant speed-up of Lagrangian particle dynamic calculations, the cluster integration method is robust and flexible. The inelastic collapse is handled and collision parameters that represent the material properties can be directly. The algorithm is suitable for a wide range of applications; fluid forces can easily be added and by increasing the virtual radius, it is possible to calculate semi-long and long-range inter-particle forces.

Acknowledgements

The author thanks Roar Meland for providing his DEM code and for introducing me to the field of soft-sphere modeling. Thanks also to Svend Tollak Munkejord for reading and commenting the manuscript. This work has been sponsored by the Research Council of Norway.

References

[1] J.R. Dormand, P.J. Prince, A family of embedded Runge–Kutta formulae, J. Comput. Appl. Math. 6 (1980) 19–26.
[2] B.D. Lubachevsky, How to simulate billiards and similar systems, J. Comput. Phys. 94 (1991) 255–283.
[3] P.A. Cundall, O.D.L. Strack, A discrete numerical model for granular assemblies, Geotechnique 29 (1) (1979) 47–65.
[4] L.P. Kadanoff, Built upon sand: Theoretical ideas inspired by granular flows, Rev. Modern Phys. 71 (1) (1999) 435–444.
[5] Y. Tsuji, T. Kawaguchi, T. Tanaka, Discrete particle simulation of two-dimensional fluidized bed, Powder Technol. 77 (1) (1993) 79–87.
[6] M.S.C. Reed, K.M. Flurchick, Hybrid molecular dynamics: an approach to low density simulations, Comput. Phys. Comm. 81 (1994) 56–64.
[7] M.P. Allen, D.J. Tildesley, Computer Simulation of Liquids, Oxford University Press, Oxford, 1987.
[8] E. Hairer, S.P. Nørsett, G. Wanner, Solving Ordinary Differential Equations I, second ed., Springer, Berlin, ISBN 3-540-56670-8, 1992, pp. 129–353 (Chapter II).
[9] H. Sigurgeirsson, A. Stuart, W.-L. Wan, Algorithms for particle-field simulations with collisions, J. Comput. Phys. 172 (2001) 766–807.
[10] J.A. Melheim, A. Gjelsvik, Adaptive cell structure for efficient detection of neighboring particles, Computers and Fluids, submitted for publication.
[11] L. Verlet, Computer “experiments” on classical fluids. I. Thermodynamical properties of Lennard-Jones molecules, Phys. Rev. 159 (1) (1967) 98–103.

Paper D

Modeling of the vortex-structure in a particle-laden mixing-layer

Melheim J. A., Horender, S., Sommerfeld, M.
In 2005 ASME FEDSM, Gas-Particle Flows, Houston, Texas, USA, June 2005

Proceedings of FEDSM2005
2005 ASME Fluid Engineering Division Summer Meeting and Exhibition
Houston, TX, USA, June 19-23, 2005

FEDSM2005-77040

MODELING OF THE VORTEX-STRUCTURE IN A PARTICLE-LADEN MIXING-LAYER

Jens A. Melheim
Department of Energy and Process Engineering
Norwegian University of Science and Technology
NO-7491 Trondheim, Norway
E-mail: jens.a.melheim@ntnu.no

Stefan Horender
Martin Sommerfeld
Institut fur Verfahrenstechnik
Martin-Luther-Universität Halle-Wittenberg
D-06099 Halle (Saale) Germany
E-mail: martin.sommerfeld@iw.uni-halle.de

ABSTRACT

Numerical calculations of a particle-laden turbulent horizontal mixing-layer based on the Eulerian-Lagrangian approach are presented. Emphasis is given to the determination of the stochastic fluctuating fluid velocity seen by the particles in anisotropic turbulence. The stochastic process for the fluctuating velocity is a "Particle Langevin equation Model", based on the Simplified Langevin Model. The Reynolds averaged Navier-Stokes equations are closed by the standard k-epsilon turbulence model. The calculated concentration profile and the mean, the root-mean-square (rms) and the cross-correlation terms of the particle velocities are compared with particle image velocimetry (PIV) measurements. The numerical results agree reasonably well with the PIV data for all of the mentioned quantities. The importance of the modeled vortex structure "seen" by the particles is discussed.

INTRODUCTION

Prediction of the particle concentration and mean and fluctuating particle velocities are important in the modeling of the particle collision frequency and for the interaction between the particle and the fluid phase in turbulent particle-laden flows. Stochastic models for particle collisions depend directly on the particle velocity fluctuations and the number density of particles [1–3]. The trajectories of the particles are especially important when evaporation takes place, where the temperature and the chemical composition of the particles depend on their history. In the Lagrangian particle tracking approach the instantaneous fluid velocity is unknown. However, the Eulerian approach for the turbulent fluid phase gives mean velocities and some average turbulence parameters that can be used to model the instantaneous fluid velocity. Minier [4] suggested a Langevin equation model for the instantaneous fluid velocity working on the particles. Minier extended the "Simplified Langevin Model" (SLM) for fluid particles, see Pope [5] and references therein, to inertial particles. Hereafter we will call the model of Minier the "Particle Langevin Model" (PLM). Previous models for the turbulent structure, such as the "eddy-life time" approach [6] and the "random walk" models [7] have shown good results for particles in grid turbulence. Coimbra *et al.* [8] calculated the mean and the rms of the fluid velocities in the two-dimensional mixing layer of Hishida *et al.* [9] with an eddy life-time model. However, the prediction of the rms of the particles velocity fluctuations failed and showed that the eddy-life time approach is insufficient in anisotropic turbulent flows. The combination of the Eulerian approach for the fluid flow and the stochastic differential equation of Minier [4] for the fluctuating velocity seen by the particles was recently presented by Naud *et al.* [10]. Naud *et al.* closed the Reynolds averaged Navier-Stokes equations by the Simplified Langevin Model. Liu *et al.* [11] combined a Langevin equation model for the instantaneous fluid velocity with a Reynolds stress model for the continuous phase and got reasonable results for the average velocities and the velocity fluctuations in a particle-laden swirl flow.

In this work we have written the PLM for the fluctuating fluid velocities and a standard k-ε model has been used for the continuous phase. The numerical simulations were compared

with the horizontal mixing-layer experiments of Horender [12]. The particles with mass mean diameter 90 μm where injected slightly above the splitter plate on the low-speed side. The mixing layer has a clear vortex structure and due to gravity the dispersion of the particles depends on the modeled life-time, frequency and strength of the vortex structure. We observe that the PLM was able to reproduce the measured concentration profile, the rms of the particle velocity and the particle velocity cross-correlations terms in anisotropic and non-homogeneous turbulence. For comparison, we showed that a random walk model [7] could not predict the concentration profile and the rms of the particle velocities correctly. The reason is the insufficient modeling of the cross-correlations terms, while the rms of the fluid velocity fluctuations were predicted well.

In the following, attention is first given to the modeling of the continuous phase, then the modeling of the fluid velocity seen by the particles. The models determining the motion of heavy particles in gas are outlined and the experimental setup is briefly presented. Then the results for the turbulent horizontal mixing-layer, laden with glass particles are presented and discussed.

MODELS FOR THE FLUID FLOW

The fluid flow was assumed to be incompressible and the particle volume fraction α_p was very small ($\alpha_p \ll 1$). Then the Reynolds averaged continuity and momentum equations are:

$$\frac{\partial \overline{U}_i}{\partial x_i} = 0 \tag{1}$$

$$\frac{\partial}{\partial t}\left(\overline{U}_i\right) + \frac{\partial}{\partial x_j}\left(\overline{U}_i \overline{U}_j\right) = -\frac{1}{\rho_f}\frac{\partial \overline{P}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(-\overline{u_i u_j}\right) + \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial \overline{U}_i}{\partial x_j} + \frac{\partial \overline{U}_j}{\partial x_i}\right)\right] + \frac{1}{\rho_f}\mathcal{F}_p, \tag{2}$$

where ρ_f is the fluid density and $\overline{U}_i$ is the mean fluid velocity defined by $U_i = \overline{U}_i + u_i$. $\overline{P}$ is the mean pressure, ν is the kinematic viscosity and $\mathcal{F}_p$ is the mean force of the particles working on the fluid. The momentum equation (2) was closed by the turbulent viscosity approach, where the Reynolds stresses $-\overline{u_i u_j}$ were expressed by [13]:

$$-\overline{u_i u_j} = \nu_t\left(\frac{\partial \overline{U}_i}{\partial x_j} + \frac{\partial \overline{U}_j}{\partial x_i}\right) - \frac{2}{3}k\delta_{ij}, \tag{3}$$

where k is the turbulent kinetic energy ($k = \frac{1}{2}\overline{u_i u_i}$) and ν_t is the turbulent kinematic viscosity, modeled by the standard k-ε-model of Launder & Spalding [14]. Launder & Spalding modeled the turbulent kinematic viscosity as follows:

$$\nu_t = C_\mu \frac{k^2}{\varepsilon}, \tag{4}$$

where $C_\mu = 0.09$ is a model constant and ε is the dissipation rate of turbulent kinetic energy. Transport equations are solved for the turbulent scalars k and ε. The transport equation for the turbulent kinetic energy k reads:

$$\frac{\partial k}{\partial t} + \overline{U}_j\frac{\partial k}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\left(\nu + \nu_t\right)\frac{\partial k}{\partial x_j}\right) + P_k - \varepsilon, \tag{5}$$

where the production of turbulent kinetic energy P_k is defined by:

$$P_k = \nu_t\left(\frac{\partial \overline{U}_i}{\partial x_j} + \frac{\partial \overline{U}_j}{\partial x_i}\right)\frac{\partial \overline{U}_i}{\partial x_j}. \tag{6}$$

The transport equation for the dissipation rate of turbulent kinetic energy ε is given by:

$$\frac{\partial \varepsilon}{\partial t} + \overline{U}_j\frac{\partial \varepsilon}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\left(\nu + \frac{\nu_t}{\sigma_\varepsilon}\right)\frac{\partial \varepsilon}{\partial x_j}\right) + C_{\varepsilon 1}\frac{\varepsilon}{k}P_k - C_{\varepsilon 2}\frac{\varepsilon}{k}\varepsilon, \tag{7}$$

where $\sigma_\varepsilon = 1.3$, $C_{\varepsilon 1} = 1.44$, and $C_{\varepsilon 2} = 1.92$. The direct influence of the particles on the turbulent kinetic energy was neglected.

The momentum equation (2) and the transport equations related to the turbulence models, Eqn. (5) and (7), were discretized using the finite-volume method on a staggered grid. The convective terms were discretized by a central differencing scheme and the pressure field was found by the SIMPLE algorithm. The calculations were performed on a grid with $100 \times 48 \times 49$ cells. Tests were carried out on a finer grid ($200 \times 98 \times 99$) to ensure a grid independent solution of the flow field. The relative deviation of the shear-layer growth between the coarse and the fine grid was 6%, which is less than the deviation between the numerical and experimental results. After the fluid flow had converged, the particles were tracked and the mean force on the fluid in each control volume was calculated. Then the flow was calculated with updated source terms. The iteration was performed until the number of iterations the fluid solver needed to converge reached a "steady number" [15].

FLUID VELOCITY SEEN BY THE PARTICLES

In the Eulerian-Lagrangian methodology, the mean values of the fluid velocity components and the turbulent scalars are given by the Eulerian approach for the fluid flow. However, the particles see the instantaneous fluid velocity $\overrightarrow{U}$. The instantaneous

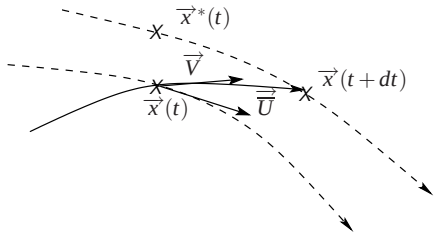


Figure 1. FLUID AND PARTICLE PATHS.

velocity is stochastic, and should not only obey the correct turbulence statistics, but it should also represent the large scale vortex structures. Pope [5,13] presented the Simplified Langevin Model (SLM) for the instantaneous fluid velocities in the Lagrangian approach following a fluid particle. By using this modeling approach, the velocity of the fluid particles are given by a stochastic differential equation for each velocity component. These models have a clear correspondence to the well established Reynolds Stress Models. Inertial particles do not follow the fluid particle, as illustrated in Fig. 1. Minier [4] suggested a model based on the Simplified Langevin Model that takes into account the drift between the inertial particles and the fluid particles, the Particle Langevin Model (PLM). An elaborated presentation of PLM is given by Minier and Peirano [16] and recently Minier *et al.* [17] introduced a two-way coupling term. In the present formulation, the two-way coupling is handled in the momentum equation (2) and is therefore not considered in the Particle Langevin Model for the fluctuating velocity. Written for the fluctuating velocity u_i, the Particle Langevin Model is given by:

$$du_i = \underbrace{-u_j\frac{\partial \overline{U}_i}{\partial x_j}dt + \frac{\partial \overline{u_i u_j}}{\partial x_j}dt}_{I} + \underbrace{(V_j - U_j)\frac{\partial \overline{U_i}}{\partial x_j}dt}_{II} \underbrace{-\frac{1}{T_{L,i}}u_i dt}_{III} + \underbrace{\sqrt{\varepsilon\left(\left(C_0+\frac{2}{3}\right)b_i\tilde{k}/k - \frac{2}{3}\right)}dW_i(t)}_{IV}. \quad (8)$$

The first term (I) on the right hand side is a fluid drift term, the change of the fluctuation due to the changes of the fluid flow. In the present work, the Reynolds stresses are calculated from Eqn. (3). The second term (II) handles the drift between the particle and the fluid velocity field. Minier [4] proposed this term as a *mean-drift-term* and used the ensemble averaged particle velocity, that is known when solving a pdf for the particle phase in an Eulerian framework. As mentioned by MacInnes & Bracco [6], the average particle velocity $\langle V_j \rangle$ at the particle position is not known *a priori*. Therefore, the instantaneous particle velocity V_j and fluid velocity U_j are used in the current work, as proposed by Simonin *et al.* [18]. A consequence of using the instantaneous velocities also in the calculations of the Csanady's factors is a possible over-prediction of those factors in low-speed regions. The third term (III) is the memory term, relating the new fluctuating fluid velocity to the previous by the Lagrangian time scale $T_{L,i}$. The Lagrangian time scale in an arbritrary direction is given by:

$$T_{L,i} = \frac{1}{\left(\frac{1}{2}+\frac{3}{4}C_0\right)\frac{\varepsilon}{k}b_i}, \quad (9)$$

where the constant C_0 takes the value $C_0 = 2.1$ [19]. The difference between the stream-wise and the transversal time scales are handled by the Csanady's factors b_i, given by:

$$b_i = b_\perp + \left(b_\parallel - b_\perp\right)\frac{(V_i - U_i)^2}{\left|\overrightarrow{V} - \overrightarrow{\overline{U}}\right|^2}, \quad (10)$$

$$b_\parallel = \left(1 + \beta^2\frac{\left|\overrightarrow{V} - \overrightarrow{U}\right|^2}{2k/3}\right)^{0.5}, \quad (11)$$

$$b_\perp = \left(1 + 4\beta^2\frac{\left|\overrightarrow{V} - \overrightarrow{U}\right|^2}{2k/3}\right)^{0.5}, \quad (12)$$

where β is the ratio between the Lagrangian T_L and Eulerian T_E time scales $\beta = T_L/T_E = 1/C_L$. In the current work $C_L = 1.1$ is used. Term III has a clear correspondence to the *linear-return-to-equilibrium* model for the redistribution term in a Reynolds Stress Model. The *linear-return-to-equilibrium* model is indeed a simple model for the redistribution term, but it seems not to be crucial in the modeling of the motion of inertial particles. The fourth term (IV) is the diffusion term, that includes randomness through the Wiener process $dW_i(t)$. The diffusion in the different directions is independent, but anisotropic. $\tilde{k}$ is a dimensionless turbulent kinetic energy weighted by the Csanady's factors that was introduced by Minier [4] and is defined by:

$$\tilde{k} = \frac{3}{2}\frac{\sum_{i=1}^{3} b_i \overline{u_i^2}}{\sum_{i=1}^{3} b_i}. \quad (13)$$

For comparison, the instantaneous fluid velocity was also calculated by the random walk model (RWM) of Sommerfeld *et al.* [7], that also can been seen as a Langevin model for an *Ornstein-Uhlenbeck* process written in finite difference form [5]. The model should therefore work well in isotropic and homogeneous turbulence. The fluctuating velocity at the next time step

$u_{n+1,i}$ is given by:

$$u_{n+1,i} = R_{P,i}(dt)u_{n,i} + \sigma_f\sqrt{1-R_{P,i}^2}\xi_i, \tag{14}$$

where ξ_i denotes a normal distribution with zero mean and a variance equal unity. σ_f is given by the turbulent kinetic energy as follows $\sigma_f = \sqrt{2/3k}$. $R_{P,i}$ is the velocity correlation function $R_{P,i} = R_L R_{E,i}$. R_L is the Lagrangian velocity correlation given by:

$$R_L = \exp(-\frac{dt}{\widehat{T}_L}), \tag{15}$$

where the Lagrangian time scale $\widehat{T}_L$ is determined from:

$$\widehat{T}_L = c_T\frac{\sigma_f^2}{\varepsilon}, \tag{16}$$

where $c_T = 0.24$ is consistent with $C_0 = 2.1$ in the PLM given in Eqn. (8). The drift between the inertial particles and the fluid, the so-called "crossing-trajectory-effect", is handled by an additional Eulerian velocity correlation function $R_{E,i}$, by using the longitudinal and transverse correlation coefficients, f and g

$$R_{E,i} = (f-g)\frac{r_i r_i}{|\overrightarrow{r}|^2} + g, \tag{17}$$

where $\overrightarrow{r}$ is the displacement between the inertial particle and fluid particle that were at the same position at the previous time step. The longitudinal and transverse correlation coefficients for homogeneous and isotropic turbulence are given by:

$$f = \exp\left(-\frac{|\overrightarrow{r}|}{L_E}\right), \tag{18}$$

$$g = \left(1-\frac{|\overrightarrow{r}|}{2L_E}\right)\exp\left(-\frac{|\overrightarrow{r}|}{L_E}\right), \tag{19}$$

where the integral length scale L_E is determined from:

$$L_E = C_L\widehat{T}_L\sigma_f. \tag{20}$$

PARTICLE MOTION

The particles were tracked individually in the Lagrangian framework and were assumed to be spherical and adiabatic. Due to the large density ratio between the particles ($\rho_p = 2950\,\mathrm{kg/m^3}$) and the carrier phase ($\rho_f = 1.2\,\mathrm{kg/m^3}$) only the drag and the gravity forces were considered. Collisions between the particles were neglected. Then the Lagrangian equations that determine the motion of N particles could be written as:

$$\frac{d\overrightarrow{X}_n}{dt} = \overrightarrow{V}_n \tag{21}$$

$$\frac{d\overrightarrow{V}_n}{dt} = -\frac{1}{\tau_n}\left(\overrightarrow{V}_n - \overrightarrow{U}\right) + \overrightarrow{g} \tag{22}$$

where $n = 1,\ldots,N$. The particle response time τ_n is defined by:

$$\tau_n = \frac{4}{3}\frac{\rho_p}{\rho_f}\frac{d_p}{C_D}\frac{1}{|\overrightarrow{V}_n - \overrightarrow{U}|} \tag{23}$$

where the drag coefficient C_D is given by the standard correlation:

$$C_D = \begin{cases} \frac{24}{Re_p}\left(1+0.15(Re_p)^{0.687}\right) & , \quad Re_p < 1000 \\ 0.44 & , \quad Re_p \geq 1000 \end{cases} \tag{24}$$

with the particle Reynolds number $Re_p = (d_p|\overrightarrow{V} - \overrightarrow{U}|)/\nu$. The fluid properties in the position of the particles were found by linear interpolation between the fluid cell centers. The motion of the parcels was calculated sequentially. Each parcel represented a number of real particles with equal properties. Data were sampled in each fluid cell for the back-coupling of the drag forces. For comparison with experimental results, data were sampled in the same way as by the PIV, described below. That is in boxes of $3.2\times3.2\times2\,\mathrm{mm}$. In order to obtain statistically reliable data also in the outskirts of the mixing layer, a huge number of parcels (10^6) was needed. The time step was limited upwards by the particle response time τ , the Lagrangian time scale T_L and the size of the control volumes for the fluid flow calculations l_{cv} as follows:

$$dt = \min\left(0.2\tau, 0.1T_L, 0.5\frac{l_{\mathrm{cv}}}{\left|\overrightarrow{V}\right|}\right). \tag{25}$$

EXPERIMENTAL SETUP

The wind tunnel was thoroughly explained by Hardalupas & Horender [20] and only the key features will be mentioned here. Figure 2 shows the horizontal setup, where the cross section of the tunnel is $0.3\times0.3\,\mathrm{m}$ and the distance from the edge of the splitter plate to the end of the channel is $1\,\mathrm{m}$. The splitter plate is $300\,\mathrm{mm}$ long and 0.5 mm tick at the edge. A pipe with inner diameter $5\,\mathrm{mm}$ and outer diameter $7\,\mathrm{mm}$, with the outlet

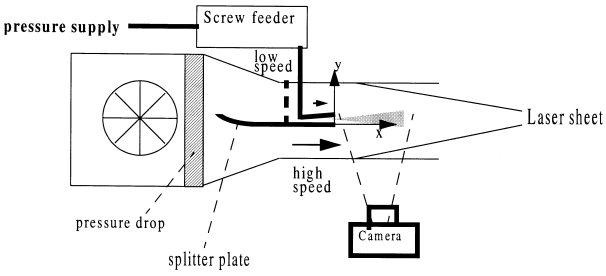


Figure 2. DRAWING OF THE EXPERIMENTAL SETUP.

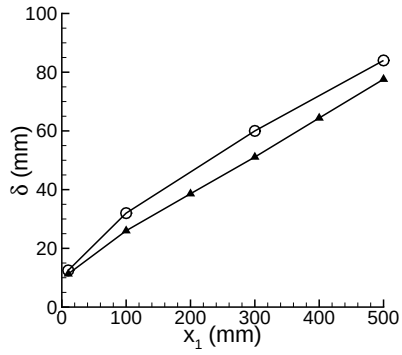


Figure 3. MOMENTUM THICKNESS δ IN STREAM-WISE DIRECTION x_1. ○ DENOTES THE EXPERIMENTS AND ▲ DENOTES THE NUMERICAL SIMULATION.

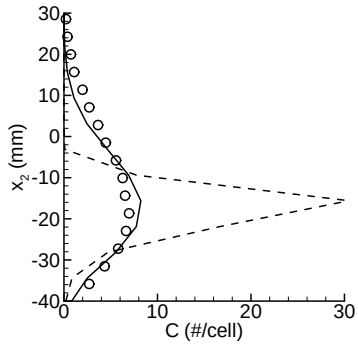


Figure 4. MEAN CONCENTRATION OF PARTICLES C. THE SYMBOLS (○) DENOTE THE EXPERIMENTAL RESULTS, THE SOLID LINE DENOTES THE PLM MODEL, EQN. (8), AND THE DASHED LINE DENOTES THE RWM MODEL, EQN. (14)

placed on the upper edge of the splitter plate, was used to inject the particles. The air velocities were $5.5\,\mathrm{m/s}$ and $0.8\,\mathrm{m/s}$ on the the high-speed and low-speed side respectively. The turbulence intensities were around 3% on the undisturbed high-speed side. The properties of the air were density $\rho_f = 1.2\,\mathrm{kg/m^3}$ and kinematic viscosity $\nu = 1.5 \cdot 10^{-5}\,\mathrm{m^2/s}$. The injection rate of particles was $7.0 \cdot 10^{-4}\,\mathrm{kg/s}$ that corresponds to a local mass loading of 12%. The density of the particles was $2950\,\mathrm{kg/m^3}$ and the particles were in the range 70-110 μm with a mass mean diameter 90 μm. The inlet particle velocity was $3.0\,\mathrm{m/s}$ with rms of stream-wise and cross-stream velocity fluctuations of $0.3\,\mathrm{m/s}$ and $0.2\,\mathrm{m/s}$ respectively. A particle image velocimeter (PIV) measured the particles over a cross section 300 mm downstream of the edge of the splitter plate. The size of the interrogation windows was 3.2×3.2 mm and 400 images were recorded with a sampling frequency of 3 Hz. The laser sheet was approximately 1 mm thick. The uncertainties were 2% and 7% for the mean particle velocity and the rms of the velocity fluctuations, 6% for the mean particle concentration and 15% for the cross-correlation between the stream-wise and cross-stream velocities. The uncertainties were expected to be somewhat larger in regions with few particles.

RESULTS

The development of the momentum thickness δ for the single-phase flow is shown in Fig. 3. The experiments shows a non-linearity in the mixing-layer growth close to the splitter plate that was not obtained by the numerical simulations. A reason might be the low turbulent Reynolds number and the intermittency of the flow. Further down-stream, the growth of the mixing layer was well predicted by the standard *k*-ε-model, and we may conclude that the *k*-ε-model is suited for mixing-layer calculations, which agrees with the findings of Coimbra *et al.* [8].

The results for the particle-laden flow are presented 300 mm downstream of the edge of the splitter plate. At this location the mixing-layer should be undisturbed by the presence of the particle injection pipe and the results should therefore be representative for the entire mixing-layer. Figure 4 shows the mean number concentration profile and we observe a good agreement between the measured number density and the PLM model. The difference between the PLM and the random walk model (RWM) is significant. Probably reasons for the difference in the dispersion of particles will be addressed in the discussion. One also sees that few particles are present for $x_2 > 20$ mm, which increases the uncertainties in this region.

The mean particle velocities, shown in Fig. 5, are generally well predicted by the RWM while the PLM over-predicts the particle velocity, in particular on the low-speed side. Figure 6 shows

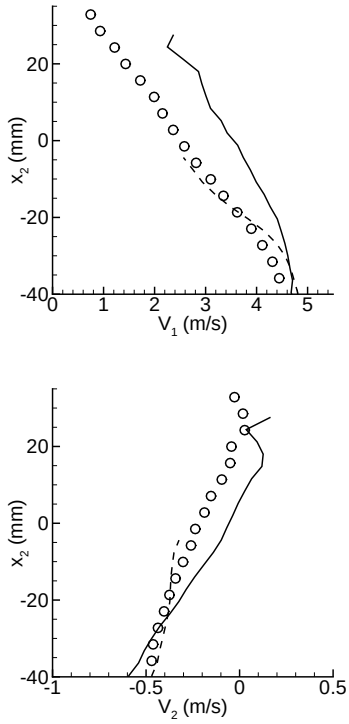


Figure 5. THE STREAM-WISE (UPPER) AND CROSS-STREAM (LOWER) MEAN PARTICLE VELOCITY AT $x_1 = 300\,\mathrm{mm}$. THE SYMBOLS (○) DENOTE THE EXPERIMENTAL RESULTS, THE SOLID LINE DENOTES THE PLM MODEL, EQN. (8), AND THE DASHED LINE DENOTES THE RWM MODEL, EQN. (14)

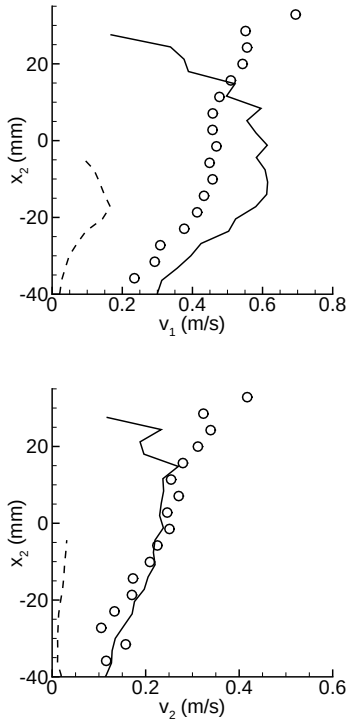


Figure 6. THE STREAM-WISE (UPPER) AND CROSS-STREAM (LOWER) PARTICLE FLUCTUATING VELOCITY AT $x_1 = 300\,\mathrm{mm}$. THE SYMBOLS (○) DENOTE THE EXPERIMENTAL RESULTS, THE SOLID LINES DENOTE THE PLM MODEL, EQN. (8), AND THE DASHED LINE DENOTES THE RWM MODEL, EQN. (14).

that the rms of the fluctuating particle velocities are far better predicted by the PLM than by RWM. We also observe that the PLM slightly over-predicts the rsm of the fluctuating particle velocity in the stream-wise direction. However, Horender [12] pointed out that the rms velocities might have been measured too low by the PIV. The results of the modeled large scale vortex structure are shown by the particle velocity cross-correlation $\langle v_1 v_2 \rangle$ in Fig. 7. The isotropic model does not correlate the particle fluctuation velocity in different directions, hence no vortex structure is predicted.

DISCUSSION

The turbulent dispersion, shown by the number density profile in Fig. 4, depends on several parameters. For instance was the particle size distribution used in the experiments measured rather roughly. In the numerical simulations a particle diameter of $90\,\mu$m was considered, hence only one particle relaxation time was present, whereas in the experiments the particle relaxation time varied with a factor two. It should also be mentioned the calculations of the presented statistics were based on the number of particles in the irrogations windows, which gives the small particles present in the experiments too much weight. Despite the uncertainties in the particle size distribution was the particle number density profile well predicted by the PLM. The calculated dispersion of particles into the low-speed side shows that vortex structure has been modeled well.

Compared to the measurements, the PLM gives too high values for the stream-wise mean particle velocity. Two reasons are the modeling of the "crossing trajectory effect" and the particle size distribution. Firstly, due to the gravity and the mean velocity gradient, term II in Eqn. (8) is in average positive, hence

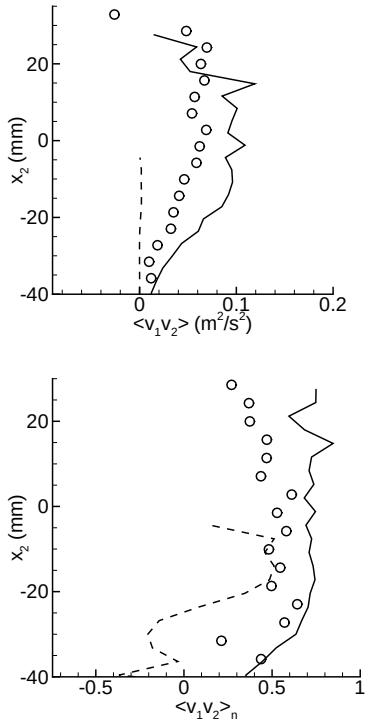


Figure 7. THE PARTICLE CROSS-CORRELATIONS $\langle v_1 v_2 \rangle$ AND THE NORMALIZED ONES $\langle v_1 v_2 \rangle_n$.

the mean fluid velocity seen by the particles is larger than the mean fluid velocity. A larger "seen" than real fluid velocity is physical in this case, but the difference might be over-predicted. Secondly, the particle size distribution might be important for the mean particle velocity on the low-speed side. Small particles travel more easily towards the gravity into the low-speed region and they furthermore retard faster to the low-speed side velocity, due to a shorter particle relaxation time.

Figure 6 shows that the rms of the fluctuating particle velocities are close to zero for the RWM. However, Fig. 8 shows that the RWM reproduce the diagonal terms in the Reynolds stress matrix well. This indicates that the integral time scale $\widehat{T}_L$ was modeled too short. Figure. 8 also shows that the cross correlation $\langle u_1 u_2 \rangle$ was not reproduced by the RWM, hence a vortex structure was not predicted.

The large scale vortex structure transports particles with high velocity from the high-speed side to the low-speed side and vice versa. The over-prediction of the rms of the particle velocity fluctuations v_1 and v_2 by the PLM can therefore be addressed to the modeling of the vortex-structure. Possible reasons for the over-prediction of v_1, v_2 and $\langle v_1 v_2 \rangle$ are over-prediction of Csanady's factors and the "crossing trajectory effect" due to the use of the instantaneous particle velocity instead of the average particle velocity, but there might also be a shortcoming of the model on this point. Another reason might be the effect of preferential concentration. The particles were tracked sequentially and the fluctuating fluid velocities related to each particles were independent, therefore a preferential concentration, that was measured by Hardalupas & Horender [20], cannot be predicted. In turbulent flows with a particle Stokes number around unity the particles spin out of the eddies into regions where straining dominates vortices [22, 23]. It is not necessarily anything wrong with the reproduction of the Reynolds stresses (Fig. 8), but the the Reynolds stresses seen by the fluid are not those seen by the particles, because the particles are accumulated in straining dominating regions. This may lead to an over-prediction of the particle fluctuating cross-correlation terms, which is seen in Fig. 7. If we normalize the cross-correlation, $\langle v_1 v_2 \rangle_n = \langle v_1 v_2 \rangle / (v_1 v_2)$, the numerical results agrees well with the experimental results. Hishida [9] also reported values of $\langle v_1 v_2 \rangle_n$ around 0.7.

CONCLUSION

In this paper numerical calculations of a particle-laden, turbulent mixing-layer based on the Eulerian-Lagrangian approach have been presented. The mixing-layer was horizontal and the particles were injected from a jet placed just above the splitter-plate, on the low-speed side. The stochastic process for the fluctuating velocity was modeled by the Particle Langevin equation Model (PLM). The Particle Langevin Model is based on the corresponding Simplified Langevin Model for single-phase flows, that again has a clear correspondence to the Reynolds Stress Models. The Reynolds averaged Navier-Stokes equations for the fluid flow were closed by the standard k-epsilon turbulence model without additional source terms. It was found that the PLM is able to reproduce the anisotropic turbulence and the vortex structure in the mixing-layer quite well. The PLM was compared with a random walk model that does not model the vortex structure. It was found that the modeling of the vortex structure is important for the prediction of the particle number density and the rsm of the particle velocity fluctuations.

ACKNOWLEDGEMENT

Thanks to Prof. Schäfer at the Technical University of Darmstadt for providing the fluid solver FASTEST. The work was made possible for the first author by an overseas fellowship from the Research Council of Norway.

REFERENCES

[1] Abrahamson, J., 1975. "Collision rated of small particles in a vigorously turbulent fluid". *Chem. Eng. Sci.*, **30** , pp. 1371–1379.

[2] Sundaram, S., and Collins, L. R., 1997. "Collision statistics in an isotropic particle-laden turbulent suspension. part 1. direct numerical simulations". *J. Fluid Mech.*, **335** , pp. 75–109.

[3] Sommerfeld, M., 2001. "Validation of a stochastic Lagrangian modelling approach for inter-particle collisions in homogeneous isotropic turbulence". *Int. J. of Multiphase Flow*, **27** , pp. 1829–1858.

[4] Minier, J.-P., 1999. "Closure proposals for the langevin equation model in lagrangian two-phase flow modelling". In Proceedings of the 3rd ASME/JSME Joint Fluids Engineering Conference.

[5] Pope, S. B., 1994. "Lagrangian pdf methods for turbulent flows". *Annu. Rev. Fluid Mech.*, **26** , pp. 23–63.

[6] MacInnes, J. M., and Bracco, F. V., 1992. "A stochastical model of turbulent drop dispersion in dilute sprays". In Proceedings of the 8th annual European Conference on Liquid Atomisation and Spray Systems, M. Afghan, Ed., pp. 79–88.

[7] Sommerfeld, M., Kohnen, G., and Rüger, M., 1993. "Some open questions and inconsistencies of lagrangian particle dispersion models". In Ninth Symposium on "Turbulent Shear Flows".

[8] Coimbra, C. F. M., Shirolkar, J. S., and MeQuay, M. Q., 1998. "Modeling particle dispersion in a turbulent multiphase mixing layer". *J. of Wind Eng. and Ind. Aerodynam.*, **73** , pp. 79–97.

[9] Hishida, K., Ando, A., and Maeda, M., 1992. "Experiments on particle dispersion in a turbulent mixing layer". Int. J Multiphase Flow, **18** (2) , pp. 181–194.

[10] Naud, B., Jiménez, C., and Roakaerts, D., 2004. "Velocity-position error of tracer particles in stochastic lagrangian modelling of polydispersed sprays based on the seen velocity". In 5th International Conference on Multiphase Flow.

[11] Liu, Z. H., Zheng, C. G., and Zhou, L. X., 2001. "A second-order-moment-monte carlo model for simulating swirling gas-particle flows". *Powder Technology*, **120** , pp. 216–222.

[12] Horender, S., 2002. *Experiments and Simulations of Particle-laden Turbulent Shear Flows*. PhD thesis, Department of Mechanical Engineering, Imperial College of Science, Technology and Medicine, London.

[13] Pope, S. B., 2000. *Turbulent Flows*. Cambridge University Press, Cambridge, United Kingdom. ISBN 0 521 59886.

[14] Launder, B. E., and Spalding, D. B., 1974. "The numerical computation of turbulent flow". *Computer Methods in Appl. Mech. and Eng.*, **3** , pp. 269–289.

[15] Kohnen, G., Rüger, M., and Sommerfeld, M., 1994. "Convergence behaviour for numerical calculations by the euler/lagrange method for stongly coupled phases". In FED-Vol. 185, Numerical Methods in Multiphase Flow, AMSE, pp. 191–202.

[16] Minier, J.-P., and Peirano, E., 2001. "The pdf approach to turbulent polydispersed two-phase flows". *Physics Reports*, **352** , pp. 1–214.

[17] Minier, J.-P., Peirano, E., and Chibbaro, S., 2004. "PDF model based on the langevin equation for polydispersed two-phase flows applied to a bluff-body gas-solid flow". Physics of Fluids, **16** (7) , pp. 2419–2431.

[18] Simonin, O., Deutsch, E., and Minier, J.-P., 1993. "Eulerian prediction of the fluid/particle corrolated motion in turbulent two-phase flows". *Appl. Sci. Res.*, **53** , pp. 275–283.

[19] Pope, S. B., 1985. "PDF methods for turbulent reactive flows". *Prog. Energy Combust. Sci.*, **11** , pp. 119–192.

[20] Hardalupas, Y., and Horender, S., 2003. "Fluctuations of particle concentration in a turbulent two-phase shear layer". *Int. J Multiphase Flow*, **29** , pp. 1645–1667.

[21] Iliopoulos, I., and Hanratty, T. J., 1999. "Turbulent dispersion in a non-homogeneous field". *J. Fluid Mech.*, **392** , pp. 45–71.

[22] Squires, K. D., and Eaton, J. K., 1991. "Preferential concentration of particles by turbulence". Phys. Fluids, **3** (5) , pp. 1169–1178.

[23] Fessler, J. R., Kulick, J. D., and Eaton, J. K., 1994. "Preferential concentration of heavy particles in a turbulent channel flow". Phys. Fluids, **6** (11) .

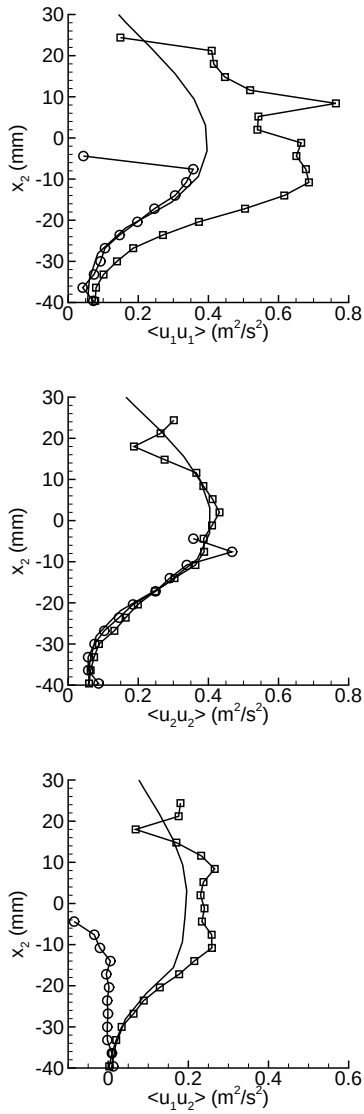


Figure 8. THE REYNOLDS STRESSES, SEEN BY THE FLUID (SOLID LINES) AND SEEN BY THE PARTICLES (○=RWM, □=PLM).

Paper E

Correlated motion of inertial particles in turbulent flows

Melheim, J. A.
Submitted to European Journal of Mechanics B/Fluids, 2006

Correlated motion of inertial particles in turbulent flows

Jens A. Melheim *

Norwegian University of Science and Technology (NTNU), Department of Energy and Process Engineering, Kolbjørn Hejes veg 1A, NO-7491 Trondheim, Norway

Abstract

A model for correlating the motion of N inertial particles in turbulent flows in the Eulerian-Lagrangian framework is proposed. The Reynolds-averaged Navier-Stokes equations provide only a mean fluid velocity field and a mean turbulence field. One therefore has to reproduce the instantaneous fluid velocity *seen* by the particles to obtain realistic particle trajectories. This is often done by evolving in time a stochastic differential equations for fluid particles, where one fluid particles follows each inertial particle. The velocity of close fluid particles is correlated, and in the proposed model, the velocity is correlated via the random term in the stochastic differential equations. Turbulence is characterized by time and length scales varying in time and space, and this intermittency is modeled by employing a stochastic differential equation for the turbulence frequency. The present results of fluid-particle pairs simulations yield the classical Richardson t^3-law. N inertial particles in isotropic and homogeneous turbulence are simulated and the results are compared with empirical correlations based on direct numerical simulations.

Key words: Particles, Turbulence, Lagrangian, Collisions,
PACS: , 47.11.+j, 47.27.Eq, 47.55.Kf

1 Introduction

Particle contacts, resulting in collision, agglomeration or coalescence, take place in many particle-laden turbulent flows. The number of particle contacts and the outcome of such contacts are influenced by the concentration of particles and the relative particle velocity at the time of contact. For particles with a response time τ_p comparable to some time scales of the fluid, the velocity of

* Tel.: +47 41215517, Fax: +47 55 57 43 31, E-mail: jensm@pvv.ntnu.no

the particles located in the same turbulent eddy will be correlated. Furthermore, if the particle response time τ_p is of same order as the Kolmogorov time scale τ_η, the particles concentrate in regions with high strain [1].

In Eulerian-Lagrangian simulations of turbulent particle-laden flows, where the mean velocity and turbulence parameters calculated, the instantaneous fluid velocity has to be modeled. The Langevin-equation model [2] is a stochastic model for the fluid velocity working on inertial particles, and it gives satisfactory results in flows like wall-jet [3] and mixing-layer [4], that include anisotropic turbulence. However, the Langevin equation determines independent particle trajectories. It is reasonable to assume that particles that are close to each other will see almost the same velocity field.

The aim of this work is to find a correlation for the fluid velocity at the position of inertial particles that are spatially close to each other. By correlating the velocity correctly, it should be possible to calculate the collision *frequency*, the collision *impact velocity* and the *preferential concentration* of particles in turbulent flows.

There is a considerable literature on the collision frequency in various turbulent flows [5,6,7,8]. Sundaram and Collins [9] made direct numerical simulations (DNS) of a homogeneous turbulent flow and found a correlation between the collision frequency and the particle Stokes number based on the Kolmogorov time scale τ_η. The Kolmogorov Stokes number is defined by

$$St_\eta \equiv \frac{\tau_p}{\tau_\eta}. \tag{1}$$

Reade and Collins [10] connected the collision frequency to the effect of preferential concentration, or clustering, of particles. Clustering of particles has been observed both in numerical calculations [1] and experimentally [11]. The centrifugal forces in a vortex try to move the particles out of the vortex. These forces are most effective at Kolmogorov Stokes numbers St_η around unity. Large-eddy simulations of homogeneous and isotropic turbulence laden with heavy particles showed a smaller impact velocity than estimated by kinetic theory [12].

Sommerfeld [13] and Berlemont et al. [14] suggested models that correlate the velocity of the fictive collision particle to the velocity of the real colliding particle in a stochastic collision model. Berlemont et al. ran also simulations with multiple particles and detection of every collision. Then the fluid velocities of the colliding particles were correlated *after* the collision. As mentioned by the authors, the models of Sommerfeld and Berlemont et al. cannot predict clustering of particles.

Another family of models handles the dispersion of fluid particle pairs by a

stochastic differential equation (SDE). Models have been made for the inter-particle distance [15] and the inter-particle velocity [16]. These models handle the relative motion of two particles well and are for instance used to close probability density function (pdf) transport equations.

Kaplan and Dinar [17] correlated the motion of N particles via the random term in the Langevin-equation model by using the spatial autocorrelation function and a fixed integral length scale L_E. An advantage of using the random term for correlating the motion is that the equations are valid for any number of particles. The model of Kaplan and Dinar fails to reproduce Richardson's t^3-law. In the current work, the motion of N inertial particles also are correlated via the random term by using the spatial autocorrelation function. Contrary to what was done by Kaplan and Dinar, the spatial correlation is based on the *instantaneous integral length scale* L_E^*, calculated by evolving in time a stochastic differential equation for the turbulence frequency [18]. Hence the intermittency of the turbulence is modeled, and, as we will show in Sec. 4, the classical Richardson t^3-law is successfully reproduced by the new model.

The paper is organized as follows. In Section 2, models for the instantaneous fluid velocity and the turbulence frequency are presented. A simple correlation function based on the instantaneous integral length scale is proposed. Section 3 briefly presents the governing equations for the motion of heavy particles in a gas without gravity. The results for dispersion of fluid-particle pairs in homogeneous turbulence are shown in Section 4. Section 5 shows the results of the collision rate, the relative velocity at contact and the radial distribution function for heavy particles suspended in isotropic and homogeneous turbulence. The results are compared with the correlations obtained from DNS calculations [19]. A discussion of the results follows in Section 6 and Section 7 concludes the paper.

2 Models for the fluid velocity

To determine realistic particle paths in cases where only mean fluid flow values are provided, one has to reproduce the instantaneous velocity *seen* by the particles. A Reynolds-averaged Navier-Stokes simulation usually gives back the mean velocity $\overline{\boldsymbol{U}}$ and mean turbulence parameters like the turbulent kinetic energy k, its dissipation ε and the Reynolds stresses $-\overline{u_i u_j}$. In non-adiabatic cases, one also has the mean temperature and the chemical composition of the fluid.

Pope [20,21,22] proposed a model framework for fluid particles. In the models of Pope, stochastic differential equations for the instantaneous fluid velocity $\boldsymbol{U_i}$ are solved in time. From many fluid-particle trajectories one can make statistics, which can be used to close pdf-equations. In homogeneous and isotropic turbulence, the *Simplified Langevin Model* (SLM) of Pope reads:

$$\mathrm{d}U_i = -\frac{1}{\rho}\frac{\partial \langle p \rangle}{\partial x_i}\,\mathrm{d}t - \frac{1}{T_L}u_i\,\mathrm{d}t + \sqrt{C_0\varepsilon}\,\mathrm{d}W_i, \tag{2}$$

where U_i is the instantaneous fluid velocity, u_i is the fluctuating fluid velocity, $\mathrm{d}W_i$ is the Wiener process, and C_0 is the Kolmogorov constant. T_L is the Lagrangian time scale determined by

$$T_L = \frac{1}{\frac{1}{2} + \frac{3}{4}C_0}\frac{k}{\varepsilon}. \tag{3}$$

A shortcoming of the SLM is the lack of information about the instantaneous length and time scales; the intermittency of the turbulence cannot be determined. Pope and Chen [18] proposed an additional model equation for the turbulence frequency ω. The average turbulence frequency is defined by

$$\langle \omega \rangle \equiv \frac{e}{k}, \tag{4}$$

and the stochastic differential equation model for the turbulent frequency ω is written as:

$$\begin{aligned}\mathrm{d}\omega = -C_\chi \omega \langle \omega \rangle \left(\ln(\omega/\langle \omega \rangle - \frac{1}{2}\sigma^2\right)\,\mathrm{d}t \\ +\omega\sqrt{2C_\chi \langle \omega \rangle \sigma^2}\,\mathrm{d}W,\end{aligned} \tag{5}$$

where the constant $C_\chi = 1.6$. σ^2 is the variance of $\ln(\omega/\langle \omega \rangle)$ and is assumed to take the value 1 in the current work. In the original model [18], there is an extra constant S_ω, but this constant is neglected in later works [22]. Eq. (5) obeys a log-normal distribution of the turbulence length and time scales. The modeled intermittency influences the model instantaneous velocity in the following way [18]:

$$\mathrm{d}U_i = -\frac{1}{\rho}\frac{\partial \langle p \rangle}{\partial x_i}\,\mathrm{d}t - \frac{1}{T_L^*}u_i\,\mathrm{d}t + \sqrt{C_0 k\omega}\,\mathrm{d}W_i, \tag{6}$$

where the instantaneous integral time scale T_L^* is given by

$$T_L^* = \frac{1}{\frac{1}{2}\langle \omega \rangle + \frac{3}{4}C_0\omega}. \tag{7}$$

Eqs. (5-7) constitute the *Refined Langevin Model* [18].

2.2 Correlated motion

The linear combination theorem and central limit theorem applied on M independent normal distributions with zero mean and equal variance gives [23]:

$$\mathcal{N}^c = \frac{\sum_{m=1}^{M} \alpha_m \mathcal{N}_m}{\sqrt{\sum_{m=1}^{M} \alpha_m^2}}. \tag{8}$$

Then $\mathcal{N}^c$ has zero mean and equal variance to that of $\mathcal{N}_m$. $\mathcal{N}^c$ approaches a normal distribution as $M \to \infty$. In the following, this property is used to correlate the random terms in Eq. (5) and Eq. (6). The Wiener processes are correlated by using the spatial autocorrelation function for isotropic and homogeneous turbulence:

$$f_{mn} = \frac{1}{l_{mn}} \sqrt{\left(l_{mn}^{\parallel} \exp\left(-l_{mn}/L_{E,n}^{*}\right)\right)^2 + \left(l_{mn}^{\perp} \exp\left(-2l_{mn}/L_{E,n}^{*}\right)\right)^2} \tag{9}$$

where $l_{mn} = \|\boldsymbol{x}_m - \boldsymbol{x}_n\|$. $l_{mn}^{\parallel}$ and $l_{mn}^{\perp}$ is the separation distance between particle n and particle m in the stream-wise and span-wise direction, respectively. The longitudinal and transverse autocorrelation functions are assumed to be equal. The instantaneous integral length scale associated with particle n, $L_{E,n}^{*}$, is determined by

$$L_{E,n}^{*} = C_l \frac{1}{\frac{1}{2} + \frac{3}{4} C_0} \frac{1}{\omega_n} \sqrt{\frac{2}{3} k} \frac{1}{e}, \tag{10}$$

where C_l is the ratio between the Eulerian and Lagrangian time scales $C_l = T_E/T_L$. e is defined by $e \equiv \exp(1)$ such that

$$\begin{aligned} L_E = \left\langle L_{E,n}^{*} \right\rangle &= \left\langle C_l \frac{1}{\frac{1}{2} + \frac{3}{4} C_0} \frac{1}{\omega_n} \sqrt{\frac{2}{3} k} \frac{1}{e} \right\rangle \\ &= C_l \frac{1}{\frac{1}{2} + \frac{3}{4} C_0} \frac{1}{\langle \omega \rangle} \sqrt{\frac{2}{3} k}, \end{aligned} \tag{11}$$

given $\sigma^2 = 1$ in Eq. (5). The correlated motion model, written for the fluctuating velocity by using the continuity, becomes:

$$\mathrm{d}u_{n,i} = -\frac{1}{T_{L,n}^{*}} u_{n,i} \, \mathrm{d}t + \sqrt{C_0 k \omega_n} \, \mathrm{d}W_{n,i}^{c}, \tag{12}$$

$$\begin{aligned} \mathrm{d}\omega_n = &-C_\chi \omega_n \langle \omega \rangle \left(\ln(\omega_n / \langle \omega \rangle - \frac{1}{2} \right) \mathrm{d}t \\ &+ \omega_n \sqrt{2 C_\chi \langle \omega \rangle} \, \mathrm{d}W_{n,\omega}^{c}, \end{aligned} \tag{13}$$

where

$$\mathrm{d}W_n^c = \frac{\sum_{m=1}^{N} f_{mn}\,\mathrm{d}W_m^*}{\sqrt{\sum_{m=1}^{N} f_{mn}^2}}, \tag{14}$$

where N is the number of particles and $\mathrm{d}W_m^*$ are independent Wiener processes. Four Wiener processes are related to each particle, one for each velocity component $u_{n,i}$ and one for the turbulence frequency ω. f_{mn} is given by Eq. (9). To ensure stable simulations, ω is kept within the range $\omega \in (0.1\,\langle\omega\rangle\,, 40\,\langle\omega\rangle)$ in the present work. Calculation of $\mathrm{d}W_n^c$ is computationally expensive for systems with many particles; the complexity is of order $O(N^2)$. A speed-up can be achieved by looping over the closest neighbors in Eq. (14); those which give a significant contribution to $\mathrm{d}W_n^c$. The closest neighbors can be found for instance by employing a particle mesh [24]. The complexity of calculating $\mathrm{d}W_n^c$ is then reduced to $O(N)$.

3 Lagrangian particle model

The particles were tracked individually in the Lagrangian framework and they were assumed to be spherical and adiabatic. Due to the large density ratio between the particles ($\rho_p = 9 - 550\,\mathrm{kg/m^3}$) and the surrounding fluid ($\rho_f = 1.0\,\mathrm{kg/m^3}$) only the drag force was considered. The drag force is modeled by the Schiller and Naumann correlation, which is reasonably good for particle Reynolds numbers, $Re_p = (d_p|\boldsymbol{V} - \boldsymbol{U}|)/\nu$, lower than 800 [25]. In order to compare with DNS data, the gravity was neglected. The Lagrangian equations that determine the motion of N particles can then be written as:

$$\begin{aligned} \frac{d\boldsymbol{X}_n}{dt} &= \boldsymbol{V}_n, \\ \frac{d\boldsymbol{V}_n}{dt} &= -\frac{1 + 0.15 Re_p^{0.687}}{\tau_{p,n}} \left(\boldsymbol{V}_n - \boldsymbol{U}\right), \end{aligned} \tag{15}$$

where $n = 1, \ldots, N$. The particle response time $\tau_{p,n}$ is defined by

$$\tau_{p,n} = \frac{\rho_p d_p^2}{18\mu}. \tag{16}$$

4 Dispersion of fluid particle pairs

To validate the proposed model for the correlated motion, dispersion of two fluid particles in homogeneous isotropic turbulence is performed. From a small initial separation l_0, the average dispersion, measured by the average of the

separation distance squared $\langle l^2 \rangle$, undergoes three regimes. This first is the viscous range ($l \ll \eta$), where the turbulence has little or no influence on the separation. The proposed model is made for real particles with diameter of order $O(\eta)$, hence very small separation distances are avoided. In the inertial range ($\eta \ll l \ll L_E$), the relative dispersion is unaffected by both the viscosity and the large scales of the turbulence. Recently, Yeung and Borgas [26] did DNS and reported that the separation distance in this region is highly intermittent. The average dispersion in the inertial range follows Richardson's t^3-law:

$$\left\langle l^2 \right\rangle = a_R \varepsilon t^3, \tag{17}$$

where a_R is a non-dimensional constant. By inclusion of an equation for the turbulence frequency in our model, the first moment of the intermittency, that was observed by Yeung and Borgas, should be captured. At large separation distances ($l \gg L_E$), the motion is uncorrelated and the average separation distance goes as $\langle l^2 \rangle \sim k T_L t$.

A background turbulence field with turbulent kinetic energy $k = 0.345\,\mathrm{m}^2/\mathrm{s}^2$ and dissipation $\varepsilon = 1.0\,\mathrm{m}^2/\mathrm{s}^3$ was considered. The fluid properties were density $\rho = 1.2\,\mathrm{kg}/\mathrm{m}^3$ and viscosity $\mu = 1.8 \cdot 10^{-5}\,\mathrm{Pa} \cdot \mathrm{s}$. The values above correspond to the Reynolds number $Re_\lambda = 230$ and the Kolmogorov scales $\eta = 2.42 \cdot 10^{-4}\,\mathrm{m}$ and $\tau_\eta = 3.873 \cdot 10^{-3}\,\mathrm{s}$. The model constants were $C_0 = 5.33$ and $C_L = 1.26$, such that the turbulence integral time and length scale were $T_L = 7.67 \cdot 10^{-2}\,\mathrm{s}$ and $L_E = 4.65 \cdot 10^{-2}\,\mathrm{m}$. 2000 pairs for each initial separation distance were found to be sufficient. Initial separations $l_0/\eta \in (1, 4, 16, 64)$ were considered. A relative long time step, $dt = 0.1 T_L$ was used and the total simulation time was $\approx 10 T_L$. Tests were done with smaller time steps without any significant difference in the results. The initial fluid velocities were picked from a Gaussian distribution with zero mean and variance $2/3k$ and then correlated similarly to the Wiener process in Eq. (14). The turbulence frequencies for the first pair were picked from a Gaussian distribution with mean $\langle \omega \rangle = \varepsilon/k$ and a standard deviation $0.1\,\langle \omega \rangle$, thereafter the last values from the previous simulated pair were used.

Fig. 4 and Fig. 4 show the 1/3 power of the mean square separation versus time for different initial separations, where only the first T_L is shown in Fig. 4. A linear scaling is clearly seen when $\eta \ll l < L_E$ and this shows that Richardson's t^3-law can be reproduced. The dimensionless Richardson's constant a_R is about 0.83 which is in reasonably good agreement with recent DNS results ($a_R \approx 0.87$) [26]. At larger separation distances, $|l| > 2L_E$, the fluid particles are separating as $\langle l^2 \rangle \sim t$, which indicates uncorrelated trajectories.

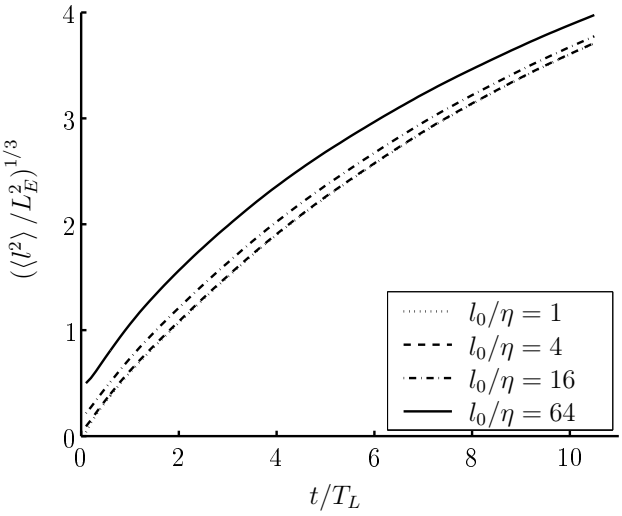


Figure 1. The 1/3 power of mean squared separation for pair of fluid particles.

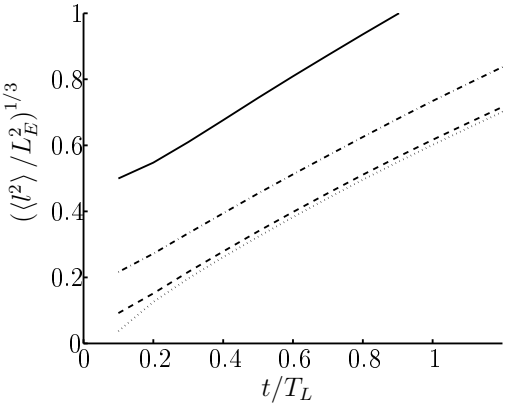


Figure 2. Same as Fig. 4, but the first T_L in time only.

5 Inertial particles in homogeneous turbulence

5.1 Collision counting

In the present work, the actual collisions are counted but not handled, hence it is often called the *ghost collision* approach. Because the forward Euler scheme is used to update the particle position and velocity, the time where a collision

between particle i and j will occur can be estimated by

$$t_{ij} = \frac{-\boldsymbol{r}_{ij} \cdot \boldsymbol{V}_{ij} - \sqrt{(\boldsymbol{r}_{ij} \cdot \boldsymbol{V}_{ij})^2 - V_{ij}^2 \left(r_{ij}^2 - (R_i + R_j)^2\right)}}{V_{ij}^2}, \tag{18}$$

where $\boldsymbol{r}_{ij} = \boldsymbol{X}_j - \boldsymbol{X}_i$, $\boldsymbol{V}_{ij} = \boldsymbol{V}_j - \boldsymbol{V}_i$ and R is the particle radius. If two particles do not overlap, but an overlap is scheduled during the next time step, $0 < t_{ij} \leq dt$, a collision is counted.

Two parameters are used to identify the behavior of the particles; the Lagrangian pair relative velocity statistics $\langle |w_r| \rangle$ and the radial distribution function $g(r)$. According to Wang et al. [19], $\langle |w_r| \rangle$ measures the turbulent transport effect and $g(r)$ measures the accumulation effect, which is closely connected to preferential concentration. The spherical formulation of the collision kernel is determined by $\langle |w_r| \rangle$ and $g(r)$ as follows:

$$\Gamma^{\text{sph}} = 2\pi d_p^2 \langle |w_r| \rangle g(r), \tag{19}$$

and the collision rate per unit volume is given by

$$\dot{\mathcal{N}}_c = \Gamma n_0^2 / 2, \tag{20}$$

where $n_0 = N/\mathcal{V}$ and $\mathcal{V}$ is the actual volume. The way $\langle |w_r| \rangle$ and $g(r)$ are computed is described in Wang et al. [19] Particle pairs are found when the distance between two particle centroids r is within $(d_p - \delta/2) \leq r \leq (d_p + \delta/2)$ where $\delta = 0.02 d_p$. The particle pairs are counted and the relative radial velocity w_r is calculated by:

$$w_r = \frac{\boldsymbol{V}_{ij} \cdot \boldsymbol{r}_{ij}}{|\boldsymbol{r}_{ij}|}. \tag{21}$$

After a simulation, the obtained data are processed to find $\langle |w_r| \rangle$ and $g(r)$. $g(r)$ is computed by [19]

$$g(r) = \frac{\text{Total number of particle pairs detected} \times V_{\text{box}}}{V_s N_t N(N-1)/2}, \tag{22}$$

where $V_s = 4/3\pi \left((d_p + \delta/2)^3 - (d_p - \delta/2)^3\right)$ is the shell volume, V_{box} is the domain volume and N_t is the number of samples taken to detect particle pairs.

5.2 Empirical correlations

It is difficult to measure the collision frequency and impact velocities in turbulent particle-laden flows. One therefore has to trust theoretical analysis and DNS, where the latter approach in general is based on fewest assumptions

and simplifications. Saffman and Turner [5] assumed light particles in turbulent flow ($St_\eta \ll 1$) and derived the following collision kernel:

$$\Gamma^{\text{Saff}} = d_p^3 \left(\frac{8\pi\varepsilon}{15\nu}\right)^{1/2}. \tag{23}$$

Wang et al. [19] made an empirical correlation based on the spherical formulation of the collision kernel, Eq. (19), from DNS of particles in isotropic and homogeneous turbulence. As mentioned above, Wang et al. measured the relative velocity statistics $\langle |w_r| \rangle$ and the relative distribution function at contact, $g(r)$. The correlation for $g(r)$ was made by curve-fitting and reads:

$$\begin{aligned} g(r) = & 1 + 18St_\eta^2 \left(1 - z_0^2\right) \\ & + Re_\lambda z_0^2 \left(y_1 \left(1 - z_1\right) + y_2 z_1 + y_3 z_2\right), \end{aligned} \tag{24}$$

where

$$\begin{aligned} y_1 &= 0.36 St_\eta^{2.5} \exp\left(-St_\eta^{2.5}\right), \\ y_2 &= 0.24 \exp\left(-0.5 St_\eta\right), \\ y_3 &= 0.013 \exp\left(-0.07 St_\eta\right), \\ z_0 &= \frac{1}{2}\left(1 + \tanh\left(\frac{St_\eta - 0.5}{0.25}\right)\right), \\ z_1 &= \frac{1}{2}\left(1 + \tanh\left(\frac{St_\eta - 1.25}{0.1}\right)\right), \\ z_2 &= \frac{1}{2}\left(1 + \tanh\left(\frac{St_\eta - 6.5}{2.5}\right)\right). \end{aligned}$$

Both turbulent shear and inertia effects (acceleration) contribute to the relative velocity statistics $\langle |w_r| \rangle$, which is modeled by

$$\langle |w_r| \rangle = \left(\frac{2}{\pi}\left(w_{r,\text{accel}}^2 + w_{r,\text{shear}}^2\right)\right)^{1/2}. \tag{25}$$

Wang et al. found that the model of Kruis and Kusters [7], written for heavy particles in gaseous flows, gave a good approximation to the DNS results when it was multiplied with a factor C_w. The modified Kruis and Kusters model reads:

$$\begin{aligned} w_{r,\text{accel}}^2 = & C_w \frac{2\gamma\theta\overline{u^2}}{\gamma - 1}\left(1 - \frac{(1+2\theta)^{1/2}}{1+\theta}\right) \\ & \times \left(\frac{1}{(1+\theta)^2} - \frac{1}{(1+\gamma\theta)^2}\right), \end{aligned} \tag{26}$$

where $\theta = \tau_p/T_L$ and $\gamma = 0.183 Re_L$. The integral-scale Reynolds number Re_L is given by $Re_L = uL_E/\nu$. Consistency with the theory of Saffman and Turner when $\tau_p \to 0$ was achieved by introducing the relative velocity at contact due to pure shear as follows:

$$w_{r,\text{shear}}^2 = \frac{1}{15}\frac{d_p^2}{\tau_\eta^2}. \tag{27}$$

5.3 Test-case setup

A cubic domain with edge lengths $2 \cdot 10^{-2}\,\text{m}$ and cyclic boundaries in all directions was considered. The turbulence was given by a constant background field of $k = 1.5\,\text{m}^2/\text{s}^2$ and $\varepsilon = 244\,\text{m}^2/\text{s}^3$. A fluid density $\rho = 1\,\text{kg}/\text{m}^3$ and kinematic viscosity $\nu = 1.825 \cdot 10^{-5}\,\text{ms}$ were used, which gave the Reynolds number $Re_\lambda \approx 58$, the Eulerian integral time scale $T_E = (2/3k)/\varepsilon \approx 4.1 \cdot 10^{-3}\,\text{s}$ and the integral length scale is $L_E = (2/3k)^{3/2}/\varepsilon \approx 4.1 \cdot 10^{-3}\,\text{m}$. The Kolmogorov scales were $\tau_\eta = 2.73 \cdot 10^{-4}\,\text{s}$ and $l_\eta = 70 \cdot 10^{-6}\,\text{m}$. The particle diameter was chosen equal to the Kolmogorov length scale, $d_p = 70 \cdot 10^{-6}\,\text{m}$, such that the point-particle approximation held. The density varied from $9.12\,\text{kg}/\text{m}^3$ to $583\,\text{kg}/\text{m}^3$, hence Kolmogorov Stokes numbers in the range $0.5 - 32$ were tested. 24000 particles were simulated, which corresponded to a particle volume fraction $\alpha_p = 0.054\%$.

Initially, the particle velocities and the fluctuating velocities related to each particle were sampled from a Gaussian distribution with zero mean and variance $2/3k$. The particles were uniformly spread in the entire domain. The system was assumed to equilibrate during the first $2.5 \cdot 10^{-2}\,\text{s}$, which equaled $3\tau_p$ of the heaviest particles. After the equilibration period, the collisions were counted and collision parameters were sampled, as described above, for about $25T_E$. A particle mesh [24] was provided to keep track of close particles in the N particle system. A $20 \times 20 \times 20$ mesh was used, that gave about 3 particles per cell. The 20^3 resolution was supposed to be a reasonable compromise between computational efficiency and accuracy in the computation of the correlated Wiener process. The effect of the mesh has been tested separately in Section 5.5.

5.4 Results

Fig. 3 shows the collision frequency $\dot{\mathcal{N}}_c$ for different Kolmogorov Stokes numbers St_η. The results for particles with correlated motion is compared with the uncorrelated results, the expression of Wang et al., and that of Saffman and Turner. There is little difference in the collision frequency obtained with and without correlated Wiener processes.

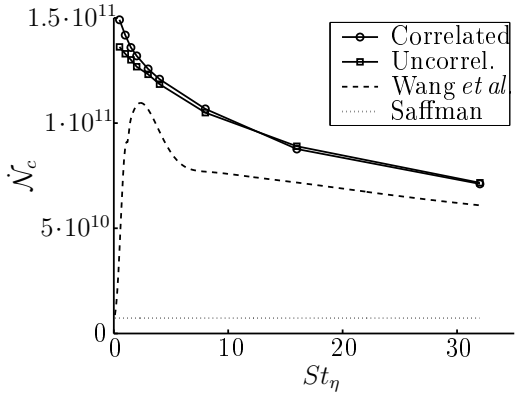


Figure 3. Number of collisions $\dot{\mathcal{N}}_c$ versus Kolmogorov Stokes number St_η.

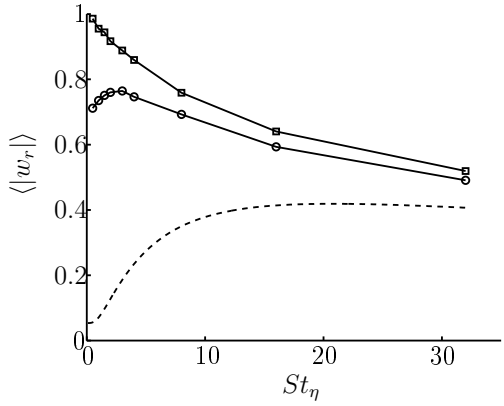


Figure 4. Relative velocity statistics $\langle |w_r| \rangle$ versus Kolmogorov Stokes number St_η. See Fig. 3 for legend.

The relative velocity at contact is shown in Fig. 4, where Wang et al. refers to the correlation given in Eq. (25), where the factor C_w takes the value $C_w = 1.63$ for $Re_\lambda = 58$ [19]. It is seen that the proposed correlation model does not equilibrate the velocity of approaching particles sufficiently prior to a collision. The results with correlated Wiener processes are almost equal to the uncorrelated results, except for low Stokes numbers where a peak in $\langle |w_r| \rangle$ is seen when the Wiener processes were correlated.

Fig. 5 shows that the proposed model did not capture the peak in the radial distribution function $g(r)$, which is estimated by Eq. (24). The uncorrelated simulations gave $g(r) \approx 1.0$, which shows that the particles really were uncorrelated.

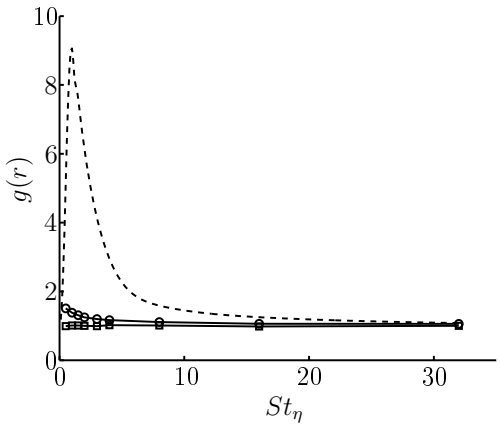


Figure 5. Radial distribution $g(r)$ versus Kolmogorov Stokes number St_η. See Fig. 3 for legend.

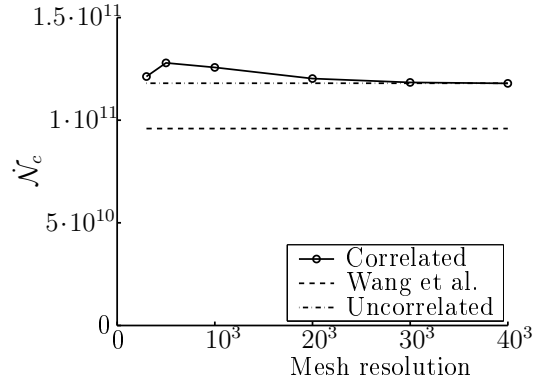


Figure 6. Number of collisions $\dot{\mathcal{N}}_c$ versus particle mesh resolution.

5.5 Effect of the particle mesh

The complexity of finding neighboring particles is reduced from $O(N^2)$ to $O(N)$ by employing a particle mesh. The size of a particle cell is typically slightly bigger than the particles for efficient computations when contact forces are the only interaction between the particles [27]. Particle interactions via the vortex structure has a typical range of L_E, decreasing as $\exp(-r)$. Due to Eq. (11), L_E^* varies considerably and the effect of different particle mesh resolutions has to be addressed.

The results with different mesh resolutions, 3^3–40^3 were obtained at a Kolmogorov Stokes number $St_\eta = 4.0$. 3^3 cells means that the entire domain, every particle, was taken into account in the calculation of the correlated Wiener processes, Eq. (14). Fig. 6 shows the collision rate per volume for dif-

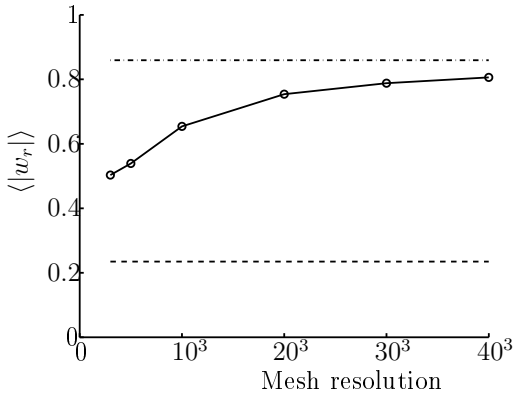


Figure 7. Relative velocity statistics $\langle |w_r| \rangle$ versus particle mesh resolution. See Fig. 6 for legend.

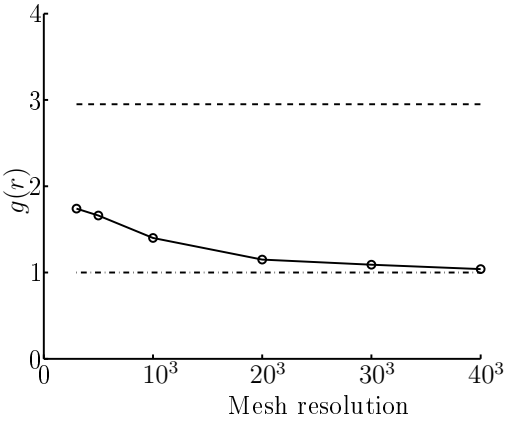


Figure 8. Radial distribution $g(r)$ versus particle mesh resolution. See Fig. 6 for legend.

ferent mesh resolutions. The collision rate seems to be almost independent of the mesh. The relative velocity at contact, shown in Fig. 7, changes significantly when coarser meshes are employed. A coarse mesh gives the particles more time to equilibrate their velocities. A smaller relative velocity gives rise to a higher value of the radial distribution function, which is observed in Fig. 8.

Fig. 9 shows the CPU-time consumption for different mesh resolutions. The CPU-time is normalized with the CPU-time of the 20^3 mesh. Because more cells have to be visited, only a slight speed-up is achieved by using a finer mesh than 20^3. A finer mesh also increases the possibility of a particle traveling more than one mesh-edge-length Δ during one time-step, which causes errors. A 5^3 mesh corresponds to $\Delta \approx L_E$, which was observed to give reasonably good results. The drawback is that a 5^3 mesh requires a factor 20 more

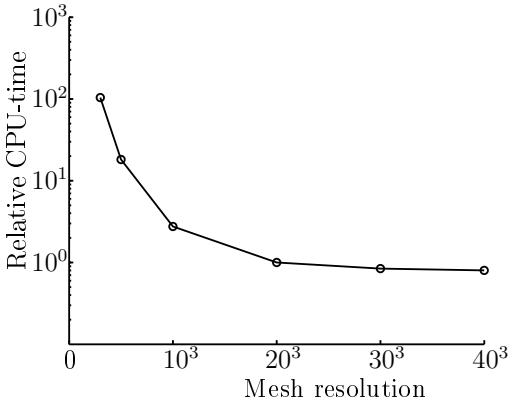


Figure 9. Relative CPU-time versus particle mesh resolution.

computational work than a 20^3 mesh.

6 Discussion

A model for correlating the motion of inertial particles in turbulent flows is proposed. The particles are evolved in time in the Lagrangian framework and a fluid particle is connected to each intertial particle to provide the instantaneous fluid velocity. The motion of close inertial particles is correlated via the random term in the stochastic differential equation for the connected fluid particle. A spatial correlation function determines the strength correlation. To model the intermittency of the turbulence, a stochastic differential equation for the turbulent frequency is solved. The turbulent frequency obeys log-normal distribution and can take any positive value, but limitations were applied to avoid very low and very high values.

Initial tests with fluid particle pairs show that the separation of particles obeys the well-know Richardson t^3-law in isotropic and homogeneous turbulence. The separation factor in the log-range was shown to be in good agreement with recent DNS data [26].

Many inertial particles were investigated in a box with isotropic and homogeneous turbulence. The collision frequency, the radial collision velocity and the radial distribution function were measured and compared with empirical expressions based on DNS data. To get comparable data, the present results were obtained under similar conditions as the DNS. This means that the collisions were not handled, only counted, and overlaps were therefore accepted. The present results reports a higher collision frequency than that observed in the DNS, in particular for $St_\eta < 5$. Also the relative velocity at contact

was overpredicted by the presented model compared to empirical correlations. The gap in the radial collision velocity decreased when coarser meshes were applied. Although the correlation most likely is small when the fluid particles are far away from each other, it seems that the velocity of the particles start approaching each other at large inter-particle spacings. Unfortunately, coarse particle meshes, which includes many particles in a large neighbourhood of the particle in question when calculating the correlated Wiener processes, lead to a high computational cost. A possible approach to overcome the computational demand is to employ techniques from the field of astrophysics, where forces decrease as $1/r$, e.g. the Barnes and Hut algorithm [28].

Collisions were only counted when particles were approaching each other, but data for the relative velocity statistics and the radial distribution function were sampled both when the particles were approaching and when the particles were separating. Overlaps give the particles more time to equilibrate their velocities and in the viscous subrange, fluid particles separate slowly [26]. It is therefore reasonable to assume that the relative separation velocity is smaller than the relative collision velocity. Hence a smaller relative velocity and more particles were sampled when the particles were separating than when the particles were approaching. Accordingly, the viscosity might played a disproportionately big role in DNS in which ghost collisions were considered [9,19]. The effect of the viscosity was neglected in the proposed model and one therefore does not expect the model to give good results for very close and overlapping particles. In a collisional system, the size of the particles ($d_p \approx \eta$) limits the minimum particle spacing and hence the viscous subrange is avoided. Sigurgeirsson and Stuart [29] compared collisional and non-collisional particle-laden flows in synthetic turbulence and found that collisions increased the diffusivity of the particles. Larger diffusivity means a larger relative particle velocity and a smaller radial distribution function. Reade and Collins [10] showed exactly that hard-sphere collisions decrease the radial distribution function.

7 Conclusions

A model for correlating the velocities of close inertial particles is proposed. The model provides some promising results. The separation of two fluid particles followed Richardson's t^3-law, which is not captured by previous N-particle models. The collision frequency, the relative particle velocity statistics and the radial distribution function of inertial particles in an isotropic and homogeneous turbulence were compared with DNS-based empirical correlations. A discrepancy, especially in the relative velocity statistics, was seen at low Stokes number, but the gap decreased when more particles, that means particles further away from the particle in question, were included in the correlation calculations. The proposed model does not include the effects of viscosity, which can

be important when correlating the fluid velocities seen by very close and overlapping particles. It would therefore be interesting to compare the proposed model with DNS-data with real particle-particle collisions, without overlaps.

8 Acknowledgments

This work was sponsored by the Research Council of Norway. Thanks to Svend Tollak Munkejord for fruitful discussions during the preparation of the manuscript and for reading and commenting.

References

[1] K. D. Squires, J. K. Eaton, Preferential concentration of particles by turbulence, Phys. Fluids 3 (5) (1991) 1169–1178.

[2] J.-P. Minier, Closure proposals for the langevin equation model in lagrangian two-phase flow modelling, in: Proceedings of the 3rd ASME/JSME Joint Fluids Engineering Conference, San Francisco, California, USA, 1999.

[3] J.-P. Minier, E. Peirano, The pdf approach to turbulent polydispersed two-phase flows, Physics Reports 352 (2001) 1–214.

[4] J. A. Melheim, S. Horender, M. Sommerfeld, Modeling of the vortex-structure in a particle-laden mixing-layer, In 2005 ASME FEDSM, *Gas-Particle Flows*, Houston, Texas, USA.

[5] P. G. Saffman, J. S. Turner, On the collision of drops in turbulent clouds, J. Fluid Mech. 1 (1956) 16–30.

[6] J. J. E. Williams, R. I. Crane, Particle collision rate in turbulent flow, Int. J. Multiphase Flow 9 (4) (1983) 421–435.

[7] F. E. Kruis, K. A. Kusters, The collision rate of particles in turbulent flow, Chem. Eng. Comm. 158 (1997) 201–230.

[8] Y. Zhou, L.-P. Wang, A. S. Wexler, On the collision rate of small particles in isotropic turbulence. II. Finite-inertia case, Phys. Fluids 10 (5) (1998) 1206–1216.

[9] S. Sundaram, L. R. Collins, Collision statistics in an isotropic particle-laden turbulent suspension. Part 1. Direct numerical simulations, J. Fluid Mech. 335 (1997) 75–109.

[10] W. C. Reade, L. R. Collins, Effect of preferential concentration on turbulent collision rates, Phys. of Fluids 12 (10) (2000) 2530–2540.

[11] J. R. Fessler, J. D. Kulick, J. K. Eaton, Preferential concentration of heavy particles in a turbulent channel flow, Phys. Fluids 6 (11).

[12] J. Laviéville, E. Deutsch, O. Simonin, Large eddy simulations of interactions between colliding particles and a homogeneous isotropic turbulence field, In ASME FED vol 228, *Gas-Particle Flows*.

[13] M. Sommerfeld, Validation of a stochastic Lagrangian modelling approach for inter-particle collisions in homogeneous isotropic turbulence, Int. J. Multiphase Flow 27 (2001) 1829–1858.

[14] A. Berlemont, P. Achim, Z. Chang, Lagrangian approaches for particle collisions: The colliding particle velocity correlation in multiple particle tracking method and in the stochastic approach, Physics of Fluids 13 (10).

[15] P. A. Durbin, A stochastical model of two-particle dispersion and concentration fluctuations in homogeneous turbulence, J. Fluid Mech. 100 (1980) 279–302.

[16] D. J. Thompson, A stochastic model for the motion of particle pairs in isotropic high-Reynolds-number turbulence, and its application to the problem of concentration variance, J. Fluid Mech. 210 (1990) 113–153.

[17] H. Kaplan, N. Dinar, A three-dimensional stochastic model for concentration fluctuation statistics in isotropic homogeneous turbulence, J. Comput. Phys. 79 (1988) 317–335.

[18] S. B. Pope, Y. L. Chen, The velocity-dissipation probability density function model for turbulent flows, Phys. Fluids 2 (8) (1990) 1437–1449.

[19] L.-P. Wang, A. S. Wexler, Y. Zhou, Statistical mechanical description and modelling of turbulent collision of inertial particles, J. Fluid Mech. 45 (2000) 117–153.

[20] S. B. Pope, pdf methods for turbulent reactive flows, Prog. Energ. Combust. 11 (1985) 119–192.

[21] S. B. Pope, Lagrangian pdf methods for turbulent flows, Annu. Rev. Fluid Mech. 26 (1994) 23–63.

[22] S. B. Pope, Turbulent Flows, Cambridge University Press, Cambridge, United Kingdom, 2000.

[23] D. T. Gillespie, Markov processes: an introduction for physical scientits, Academic Press, San Diego, 1992, ISBN 0-12-283955-2.

[24] J. W. Eastwood, R. W. Hochney, D. N. Lawrence, P3M3DP – The three dimensional periodic particle-particle/particle-mesh program, Comput. Phys. Commun. 19 (1980) 215–261.

[25] C. Crowe, M. Sommerfeld, Y. Tsuji, Multiphase flows with droplets and particles, CRC Press, Florida, 1998, ISBN 0-8493-9469-4.

[26] P. K. Yeung, M. S. Borgas, Relative dispersion in isotropic turbulence. Part 1. Direct numerical simulations and Reynolds-number dependence, J. Fluid Mech. 503 (2004) 93–124.

[27] H. Sigurgeirsson, A. Stuart, W.-L. Wan, Algorithms for particle-field simulations with collisions, J. Comput. Phys. 172 (2001) 766–807.

[28] J. E. Barnes, P. Hut, A hierarchical $O(N \log N)$ force-calculation algorithm, Nature 324 (1986) 446–449.

[29] H. Sigurgeirsson, A. M. Stuart, A model for preferential concentration, Phys. Fluids 14 (12) (2002) 4352–4361.

Paper F

Simulation of turbulent electrocoalescene

Melheim, J. A. and Chiesa, M.
Chemical Engineering Science 61 (2006) 4540 - 4549

ELSEVIER

Chemical Engineering Science 61 (2006) 4540–4549

Chemical Engineering Science

www.elsevier.com/locate/ces

Simulation of turbulent electrocoalescence

Jens A. Melheim[a,*], Matteo Chiesa[b]

[a]*Department of Energy and Process Engineering, Norwegian University of Science and Technology (NTNU), Kolbjørn Hejes veg 1A, NO-7491 Trondheim, Norway*

[b]*SINTEF Energy Research, Sem Sælandsvei 11, NO-7465 Trondheim, Norway*

Received 2 November 2005; received in revised form 14 February 2006; accepted 15 February 2006
Available online 28 February 2006

Abstract

The combination of an electric field and a moderate turbulent flow is a promising technique for enhancing the separation of water from oil. In this work, a numerical framework based on the Eulerian–Lagrangian approach is presented, where the turbulent dispersion and the inter-droplet hydrodynamic and electrical forces are carefully handled. Water-in-oil emulsions are studied in a channel flow with almost isotropic, decaying turbulence. The results obtained agree qualitatively with experimentally data reported in the literature. Our simulations show that the collision frequency is mainly controlled by the turbulence, but strong electric fields may increase the collision rate at low turbulence levels. It is also observed that turbulent electrocoalescence works equally well for all simulated volume fractions of water droplets.

PACS: 45.50.−j; 82.70.Kj; 83.10.Rs; 83.95.Pt

Keywords: Emulsion; Turbulence; Simulation; Multiphase flow; Electrocoalescence; Eulerian–Lagrangian

1. Introduction

The separation of water from oil is a bottleneck in offshore oil production. The oil–water mixture produced from oil wells often consists of oil as the continuous phase with water dispersed as small droplets. Traditionally, the separation of water from oil is done in several stages and in large vessels. The separation relies on buoyancy as the driving force; the heavier water droplets will fall and form a dense-packed layer at the bottom of the separation vessels. There, the droplets coalesce and form a continuous water phase that is drained out of the separation vessel. The speed of the sedimentation process is controlled by the terminal velocity, which in the Stokes regime is given by

$$V_s = \frac{\rho_d d_d^2}{18\mu}\left(1 - \frac{\rho}{\rho_d}\right) g, \tag{1}$$

where the falling droplet is assumed to behave like a rigid sphere in an infinite stagnant fluid. We observe that the terminal velocity of the droplets is proportional to the square of the droplet diameter. The only way to speed up the sedimentation process is to make small water droplets coalesce into larger ones. Effective coalescence is achieved through the combined effect of an electric field that induces attractive forces between the water droplets and a turbulent flow regime that provides a high collision frequency.

Traditionally, one has introduced the electric field within the last-stage separation vessel by using bare electric grids (Sams and Zaouk, 2000). These grids are sensitive to water bridging, resulting in unwanted short-circuits. Another problem is the large droplet spacing in dilute mixtures, where the attractive electric forces among the droplets become almost insignificant. In the last years, equipment vendors have shown a renewed interest in electrocoalescence. They have realized that the inefficiency of traditional electrocoalescers is mainly due to the fact that large droplet spacings hinder the electrical forces to have a large impact on the coalescence. To improve the efficiency, Atten (1993) suggested to combine large shears and electrical forces. A more efficient electrocoalescence was easily achieved by placing the electrodes at the inlet piping of the first-stage separation vessel, see Fig. 1. This had a huge impact on separation

* Corresponding author. Tel.: +47 41215517; fax: +47 73592889.
E-mail address: jensm@pvv.ntnu.no (J.A. Melheim).

0009-2509/$ - see front matter © 2006 Elsevier Ltd. All rights reserved.
doi:10.1016/j.ces.2006.02.022

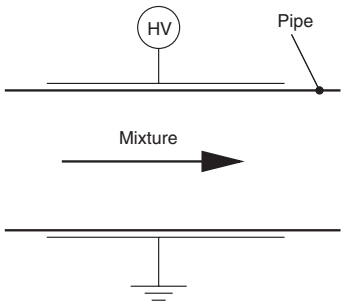


Fig. 1. Schematics of the modern in-line electrostatic coalescers, from Hansen (2005).

efficiency, and laboratory results reported an up to 50-fold reduction in separation time (Friedemann et al., 2001) and a 10-fold increase in the volume-weighted mean diameter (Harpur et al., 1997). Even though many years of research have been put into the electrocoalescence process, more detailed understanding is needed. The lack of knowledge of the electrocoalescence process demands numerical tools able to predict the droplet behavior in a turbulent flow under the influence of an electric field.

Turbulent flows are irregular, seemingly random and highly diffusive. The droplet paths in a turbulent velocity field depend on the droplet response time τ_d and the velocity field they have been exposed to. Droplet collisions, which may lead to coalescence, are due to motion of droplets relative to each other. To characterize the behavior of droplets in turbulent flows, it is convenient to use the Kolmogorov Stokes number, St, which is defined by

$$St \equiv \frac{\tau_d}{\tau_\eta}, \tag{2}$$

where τ_d is the droplet response time $\tau_d = \frac{1}{18}\rho_\eta d_\eta^2/\mu$ and τ_η the Kolmogorov time scale $\tau_\eta = (\nu/\varepsilon)^{1/2}$. Only very small droplets ($St \ll 1$) follow the flow perfectly, and big droplets ($St \gg 1$) are not influenced by the turbulence at all. The collision frequency between very small droplets yields the expression of Saffman and Turner (1956) and is small compared to the collision frequency between bigger droplets with uncorrelated velocities (Abrahamson, 1975). There is a considerable literature on the collision frequency of droplets between these two extremes: Williams and Crane (1983) and Kruis and Kusters (1997) based their expressions on an analytical approach while the expressions of Sundaram and Collins (1997) and Wang et al. (2000) were supported by direct numerical simulations. For Stokes numbers St around unity, a preferential concentration is experienced (Squires and Eaton, 1991). The droplets accumulate in regions with high strain between the vortices, and this enhances the collision frequency. There is a consensus in literature that the collision frequency decreases when the Stokes number becomes less than one. To benefit from the turbulence, one should, therefore, try to keep the droplet response time τ_d of the same order as, or larger than, the Kolmogorov time scale τ_η.

In the present work, the "steady-state" Reynolds-averaged Navier–Stokes equations (RANS) closed with the k–ε turbulence model is combined with a discrete particle model (DPM) to calculate turbulent electrocoalescence. The instantaneous fluid velocity "seen" by the droplets is realized by evolving in time a stochastic differential equation for the fluctuating fluid velocity at the positions of the droplets. The fluid velocity seen by close droplets is correlated as proposed by Melheim (2006). Another Lagrangian method is the cloud particle model, where one numerical particle represents many physical ones and the collisions are determined by stochastic models, see for instance Sommerfeld (2001). The advantage of the cloud particle model is the opportunity to handle many droplets, but it is not straightforward to model inter-droplet hydrodynamic and electric forces in a stochastic collision model. Takeda et al. (2004) combined the DPM with large-eddy simulations (LES) and simulated bubble coalescence and break-up in a bubble column. LES predicts unsteady flows better than RANS, as for instance bubble columns and fluidized beds, but LES is more unstable. The smallest time scales, which are important for the droplet collisions, are still filtered away. The DPM was successfully used by Chen et al. (1994) to simulate chain-formation and electrocoalescence of water droplets in a stagnant oil.

In the following, the basics equations governing the particle motion and the fluid flow in the Eulerian–Lagrangian framework are reviewed. A channel flow with almost isotropic, decaying turbulence is calculated, and the effects of the electric field, turbulence level and droplet volume fraction are investigated. The results show that turbulent electrocoalescence is very effective and the trends that have been observed experimentally are confirmed.

2. Modeling framework

2.1. Discrete particle model

In the discrete particle model, each droplet is tracked individually. The position $\boldsymbol{x}$ and velocity $\boldsymbol{V}$ of the droplets are calculated by

$$\frac{\mathrm{d}\boldsymbol{x}}{\mathrm{d}t} = \boldsymbol{V}, \tag{3}$$

$$\frac{\mathrm{d}\boldsymbol{V}}{\mathrm{d}t} = \frac{1}{m_d}\sum \boldsymbol{F}_d, \tag{4}$$

where $\sum \boldsymbol{F}_d$ is the sum of the forces acting on a droplet. The realization of the instantaneous velocity field "seen" by the droplets is crucial for the droplet trajectories, and thereby the collision frequency and collision velocity. The instantaneous fluid velocity is modeled by employing stochastic differential equations for the fluctuating fluid velocities at the particle position. Efficient and accurate calculations are ensured by employing an algorithm proposed by Melheim (2005), called the cluster integration method.

2.1.1. Forces acting on the droplets

In a turbulent water-in-oil emulsion influenced by an electric field, the following forces are assumed to affect the droplet motion:

$$\sum \boldsymbol{F}_p = \boldsymbol{F}_D + \boldsymbol{F}_B + \boldsymbol{F}_H + \boldsymbol{F}_A + \boldsymbol{F}_{\mathrm{Film}} + \boldsymbol{F}_{\mathrm{El}}, \tag{5}$$

where $\boldsymbol{F}_D$ is the drag force, $\boldsymbol{F}_B$ is the buoyancy force, $\boldsymbol{F}_H$ is the history force, $\boldsymbol{F}_A$ is the added-mass force, $\boldsymbol{F}_{\mathrm{Film}}$ is the film-thinning forces and $\boldsymbol{F}_{\mathrm{El}}$ is the inter-droplet electric forces. The history force is assumed to be negligible in very random flows (Vojir and Michaelides, 1994). Lift forces and wall effects are neglected because we assume spherical droplets and we are only interested in the flow in the middle of the channel, where the turbulence is nearly isotropic and the mean velocity gradients are small.

Small contaminated water droplets in oil behave almost as rigid spherical particles. In the present work, the droplet Reynolds number $Re_d = \rho|\boldsymbol{U} - \boldsymbol{V}|d_d/\mu$ is around unity and below and the maximum Bond number $Bo = |\rho_d - \rho|gd_d^2/\sigma$, where σ is the surface tension, is of order $O(10^{-3})$. Hence, the droplets are in the spherical region in the shape map of Fan and Tsuchiya, see Loth (2000). Furthermore, Loth (2000) showed that the expressions Clift et al. (1978) proposed for "grossly contamined" droplets are almost identical to that of rigid spheres for droplet Reynolds numbers Re_d less than 80. Expressions valid for rigid spheres are therefore used to model the drag force in this work. Chiesa et al. (2005) studied forces between two water droplets under the influence of an electric field and recommended expressions for the different forces. The expressions given below for the film-thinning and dipole–dipole forces that were suggested by Chiesa et al. are used in the present work.

The drag force is written as

$$\boldsymbol{F}_D = \tfrac{1}{2}\rho C_d A|\boldsymbol{U} - \boldsymbol{V}|(\boldsymbol{U} - \boldsymbol{V}), \tag{6}$$

where C_d is given by

$$C_d = \frac{24}{Re_d}\left(1 + \frac{3}{16}Re_d\right) \quad \text{for } Re_d \leqslant 1, \tag{7}$$

$$C_d = \frac{24}{Re_d}\left(1 + \frac{1}{6}Re_d^{2/3}\right) \quad \text{for } Re_d > 1. \tag{8}$$

Because of the applied electric field, close droplets will coalesce quickly, and "swarm" effects are therefore neglected.

The buoyancy force is given by

$$\boldsymbol{F}_B = (\rho_d - \rho)gV_d\boldsymbol{e}_g, \tag{9}$$

where $\boldsymbol{e}_g$ is the direction of the gravity.

The mean fluid flow is stationary and there is no steep spatial velocity gradients away from the walls. Hence, the added-mass force becomes

$$\boldsymbol{F}_A = \frac{1}{2}\rho V_d\left(\frac{\mathrm{d}\boldsymbol{u}}{\mathrm{d}t} - \frac{\mathrm{d}\boldsymbol{V}}{\mathrm{d}t}\right), \tag{10}$$

where $\boldsymbol{u}$ is the fluctuating fluid velocity.

Vinogradova (1995) proposed an expression for the film-thinning force between hydrophobic spheres that reads

$$\boldsymbol{F}_{\mathrm{Film}} = -\frac{6\pi\mu a^2(\boldsymbol{V}_r \cdot \boldsymbol{e}_r)}{h} \times \left\{\frac{2h}{6b}\left[\left(1 + \frac{h}{6b}\right)\ln\left(1 + \frac{6b}{h}\right) - 1\right]\right\}\boldsymbol{e}_r, \tag{11}$$

where $a = r_1r_2/(r_1 + r_2)$ is the reduced radius, $\boldsymbol{V}_r \equiv \boldsymbol{V}_2 - \boldsymbol{V}_1$ is the relative velocity vector, and $\boldsymbol{e}_r$ indicates the direction of the relative motion, h is the smallest surface-to-surface distance and b is a fitting parameter, that can be regarded as a slip factor. b was given the value $b = 10^{-7}$ m in this work.

Water droplets are polarized by an applied electric field. Polarization results in an inhomogeneous electric field, which again gives rise to electric dipole–dipole forces. The dipole-induced-dipole (DID) model of Siu et al. (2001) is computationally efficient and suited for multi-droplet systems. Chiesa et al. (2005) showed that the DID model was superior to the point-dipole model at small droplet spacings. The DID model is written as

$$F_r = \frac{12\pi\beta^2\varepsilon_{\mathrm{oil}}|\boldsymbol{E}_0|^2r_2^3r_1^3}{|\boldsymbol{d}|^4}(3K_1\cos^2\theta - 1), \tag{12}$$

$$F_t = -\frac{12\pi\beta^2\varepsilon_{\mathrm{oil}}|\boldsymbol{E}_0|^2r_2^3r_1^3}{|\boldsymbol{d}|^4}K_2\sin(2\theta), \tag{13}$$

where β is defined as

$$\beta = \frac{\varepsilon_w - \varepsilon_{\mathrm{oil}}}{\varepsilon_w + 2\varepsilon_{\mathrm{oil}}}. \tag{14}$$

The coefficients K_1 and K_2 are given by

$$K_1 = 1 + \frac{\beta r_1^3|\boldsymbol{d}|^5}{(|\boldsymbol{d}|^2 - r_2^2)^4} + \frac{\beta r_2^3|\boldsymbol{d}|^5}{(|\boldsymbol{d}|^2 - r_1^2)^4} + \frac{3\beta^2r_1^3r_2^3(3|\boldsymbol{d}|^2 - r_1^2 - r_2^2)}{(|\boldsymbol{d}|^2 - r_1^2 - r_2^2)^4},$$

$$K_2 = 1 + \frac{\beta r_1^3|\boldsymbol{d}|^3}{2(|\boldsymbol{d}|^2 - r_2^2)^3} + \frac{\beta r_2^3|\boldsymbol{d}|^3}{2(|\boldsymbol{d}|^2 - r_1^2)^3} + \frac{3\beta^2r_1^3r_2^3}{(|\boldsymbol{d}|^2 - r_1^2 - r_2^2)^3}. \tag{15}$$

We observe that the coefficients K_1 and K_2 approach unity as $|\boldsymbol{d}| \to \infty$, and the point-dipole model is recovered. It is assumed that the droplets are uniformly distributed over the cross-section, such that the background field $\boldsymbol{E}_0$ remains unchanged.

2.1.2. Turbulence

The instantaneous velocity "seen" by the droplets is needed in the above models for the forces. Melheim (2006) proposed a model for the fluctuating fluid velocity that correlates the fluid velocities seen by close droplets. The model of Melheim is based on the *Refined Langevin Model* of Pope and Chen (1990). Written for the fluctuating velocity in isotropic and

Table 1
Constants in models describing the turbulence working on the droplets

C_0	C_χ	σ^2	C_l
5.0	1.6	1.0	4.0

homogeneous turbulence, the Refined Langevin Model becomes

$$\mathrm{d}u_i = -\frac{1}{T_L^*}u_i\mathrm{d}t + \sqrt{C_0 k\omega}\,\mathrm{d}W_i, \tag{16}$$

where the instantaneous integral time scale T_L^* is given by

$$T_L^* = \frac{1}{\frac{1}{2}\langle\omega\rangle + \frac{3}{4}C_0\omega}. \tag{17}$$

The average turbulent frequency $\langle\omega\rangle$ is defined by $\langle\omega\rangle \equiv \varepsilon/k$ and the instantaneous turbulence frequency ω is determined by the following stochastic differential equation:

$$\mathrm{d}\omega = -C_\chi\omega\langle\omega\rangle\left(\ln(\omega/\langle\omega\rangle) - \tfrac{1}{2}\sigma^2\right)\mathrm{d}t + \omega\sqrt{2C_\chi\langle\omega\rangle\sigma^2}\,\mathrm{d}W. \tag{18}$$

The model constants used in the present work are given in Table 1.

The fluid velocity seen by close droplets are correlated via the Wiener process $\mathrm{d}W$ as follows (Melheim, 2006):

$$\mathrm{d}W_n = \frac{\sum_{m=1}^{N} f_{mn}\mathrm{d}W_m^*}{\sqrt{\sum_{m=1}^{N} f_{mn}^2}}, \tag{19}$$

where N is the number of droplets and $\mathrm{d}W_m^*$ represents independent Wiener processes. f_{mn} is given by

$$f_{mn} = \frac{1}{l_{mn}}\sqrt{(l_{mn}^{\parallel}\exp(-l_{mn}/L_{E,n}^*))^2 + (l_{mn}^{\perp}\exp(-2l_{mn}/L_{E,n}^*))^2}, \tag{20}$$

where $l_{mn} = \|\boldsymbol{x}_m - \boldsymbol{x}_n\|$. $l_{mn}^{\parallel}$ and $l_{mn}^{\perp}$ are the separation distances between droplet n and droplet m in the stream-wise and span-wise directions, respectively. In Eq. (20), the longitudinal and transversal autocorrelation functions are assumed to be equal. The instantaneous integral length scale associated with particle n, $L_{E,n}^*$, is determined by

$$L_{E,n}^* = C_l\frac{1}{\frac{1}{2}+\frac{3}{4}C_0}\frac{1}{\omega_n}\sqrt{\frac{2}{3}k}\frac{1}{\exp(1)}, \tag{21}$$

where C_l is the ratio between the Eulerian and Lagrangian time scales $C_l = T_E/T_L$ and is given in Table 1.

2.1.3. Cluster integration method

An efficient and accurate computation is achieved by using the *cluster integration method*, which is described in Melheim (2005). Each droplet is equipped with a virtual radius and clusters are assembled of droplets with overlapping virtual radii as shown in Fig. 2. A test move following the pattern of an event-driven *hard-sphere* algorithm (Sigurgeirsson et al., 2001) is employed to search for droplets that might have overlapping virtual radii during the next global time step Δt. Then the clusters are integrated separately one global time step Δt using the embedded Runge–Kutta scheme of Dormand and Prince (1980). After a local time step is successfully performed for the positions and velocities of the droplets within a cluster, the fluctuating velocities and the turbulence frequencies of those droplets are updated using a Runge–Kutta scheme for stochastic differential equations (Burrage and Burrage, 1996).

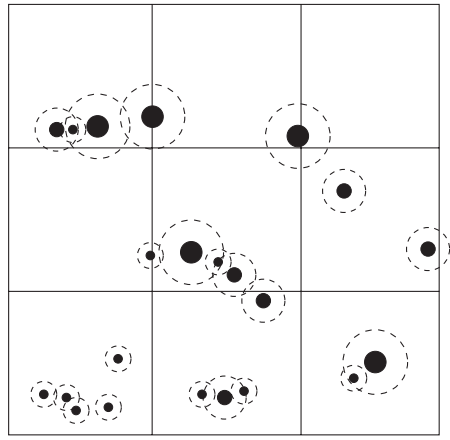


Fig. 2. Particle cells and droplets which are equipped with a virtual radius.

An adaptive cell structure is applied to optimize the search for close droplets (Melheim et al., 2005). The cell structure adapts to the number density of droplets and to the droplet size. Because the cell structure is also used to correlate the Wiener process in Eq. (19), the minimum cell width $\Delta x_{\min}^d$ is of order $O(L_E)$. Furthermore, there should be room for a droplet and its virtual neighborhood in a cell. Accordingly, the cell structures used in this work were coarser than what was stated as optimal for hard-sphere simulations by Sigurgeirsson et al. (2001).

2.2. Fluid-flow equations

The volume-averaged Navier–Stokes equations for an incompressible and isentropic fluid are given by

$$\frac{\partial\alpha}{\partial t} + \frac{\partial\alpha\overline{U}_i}{\partial x_i} = 0, \tag{22}$$

$$\frac{\partial}{\partial t}(\alpha\overline{U}_i) + \frac{\partial}{\partial x_j}(\alpha\overline{U}_i\overline{U}_j) = -\alpha\frac{1}{\rho}\frac{\partial\overline{P}}{\partial x_i} + \frac{\partial}{\partial x_j}(-\alpha\overline{u_i u_j}) + \frac{\partial}{\partial x_j}\left[\alpha\nu\left(\frac{\partial\overline{U}_j}{\partial x_j} + \frac{\partial\overline{U}_j}{\partial x_i}\right)\right] + \frac{1}{\rho}\mathscr{F}_d, \tag{23}$$

where $\overline{U}_i$ is the mean fluid velocity, $\overline{P}$ is the mean pressure and $\mathscr{F}_d$ is the mean force of the particles working on the fluid. The

momentum equation (23) is closed by the turbulent viscosity approach, where the Reynolds stresses $-\overline{u_i u_j}$ are expressed by (Pope, 2000)

$$-\overline{u_i u_j} = v_t \left(\frac{\partial \overline{U}_i}{\partial x_j} = \frac{\partial \overline{U}_j}{\partial x_i} \right) - \frac{2}{3} k \delta_{ij}, \tag{24}$$

where k is the turbulent kinetic energy ($k = \frac{1}{2}\overline{u_i u_i}$). The turbulent kinematic viscosity v_t is modeled by the standard k–ε model of Launder and Spalding (1974):

$$v_t = C_\mu \frac{k^2}{\varepsilon}, \tag{25}$$

where $C_\mu = 0.09$ is a model constant and ε is the dissipation rate of turbulent kinetic energy. Transport equations are solved for the turbulent scalars k and ε. The transport equation for the turbulent kinetic energy k, where the volume fraction α is included, reads

$$\frac{\partial}{\partial t}(\alpha k) + \frac{\partial}{\partial x_j}(\alpha k \overline{U}_j) = \frac{\partial}{\partial x_j}\left(\alpha(v + v_t) \frac{\partial k}{\partial x_j} \right) + \alpha P_k - \alpha \varepsilon, \tag{26}$$

where the production of turbulent kinetic energy P_k is defined by

$$P_k = v_t \left(\frac{\partial \overline{U}_i}{\partial x_j} + \frac{\partial \overline{U}_j}{\partial x_i} \right) \frac{\partial \overline{U}_i}{\partial x_j}. \tag{27}$$

The transport equation for the dissipation rate of turbulent kinetic energy ε is given by

$$\frac{\partial}{\partial t}(\alpha \varepsilon) + \frac{\partial}{\partial x_j}(\alpha \varepsilon \overline{U}_j) = \frac{\partial}{\partial x_j}\left(\alpha \left(v + \frac{v_t}{\sigma_\varepsilon} \right) \frac{\partial \varepsilon}{\partial x_j} \right) + C_{\varepsilon 1} \alpha \frac{\varepsilon}{k} P_k - C_{\varepsilon 2} \alpha \frac{\varepsilon}{k} \varepsilon, \tag{28}$$

where $\sigma_\varepsilon = 1.3$, $C_{\varepsilon 1} = 1.44$, and $C_{\varepsilon 2} = 1.92$.

The momentum equation (23) and the transport equations related to the turbulence models, Eqs. (26) and (28) are discretized using the finite-volume method on a non-staggered grid. The convective terms are discretized by a power-law scheme and the pressure field is found by the SIMPLE algorithm (Patankar, 1980). After the fluid flow has converged, a continuous flow of droplets are tracked for a certain period and the mean force on the fluid and the mean volume fraction of droplets in each control volume are calculated. Then the flow is calculated with updated source terms and volume fractions. This iteration process is performed until the fluid flow converged quickly with updated source terms, this means that a stationary solution is reached.

3. Channel for the numerical experiments

A two-dimensional channel, sketched in Fig. 3, was considered for the numerical experiments. The flow-field down stream of a perforated plate was assumed. An electric field was applied perpendicularly to the flow direction and the direction of gravity was against the flow. An ideal bipolar square voltage was considered. Plug profiles were given on the inlet, and due to the short channel length, the flow in the middle of the channel was considered as decaying isotropic turbulence. Three different set of inflow data were considered, with Taylor-scale Reynolds number Re_λ in the range 9–15, which is rather low. The inlet data are given in Table 2. Not only the flow was varied, but also the effects of different electric fields and the volume fractions of droplets in the channel were studied. The test matrix is presented in Table 3, where Case 1 is the reference case.

A fluid with density $\rho = 800\,\mathrm{kg/m^3}$ and dynamic viscosity $\mu = 4 \times 10^{-3}\,\mathrm{kg/m/s}$ models a relatively light crude oil. The crude oil was assumed to be a perfect insulator, with a relative permittivity $\varepsilon_{\mathrm{oil}} = 2.2$. A surface tension of $20 \times 10^{-3}\,\mathrm{N/m}$ is

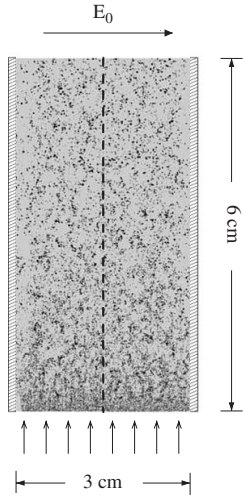


Fig. 3. Sketch of the channel used in the numerical experiment.

Table 2
Inflow data

Re_λ	$\overline{U}_{y,\mathrm{inlet}}$ (m/s)	k_{inlet} ($\mathrm{m^2/s^2}$)	$\varepsilon_{\mathrm{inlet}}$ ($\mathrm{m^2/s^3}$)
9	0.15	0.0013	0.027
125	0.30	0.005	0.22
15	0.45	0.011	0.73

Table 3
Test matrix

Case	E_0 (V/mm)	α_d (%)	Re_λ
1	250	2	12.5
2	0	2	12.5
3	125	2	12.5
4	500	2	12.5
5	250	2	9
6	250	2	15
7	250	1	12.5
8	250	5	12.5

a typical value for emulsion produced from offshore oil-fields (Sjöblom et al., 2003).

Water droplets were inserted at the inlet every time step, so that the volume fraction α_d given in Table 3 was fulfilled. A salt content of 3.5 wt% was assumed, so that the water can be regarded as a perfect conductor, $\varepsilon_w = 1000$. The water density was $\rho_d = 1000\,\mathrm{kg/m^3}$. The inserted droplets were given the mean fluid velocity plus a random velocity v, which was picked from a normal distribution with zero mean and a standard deviation of $(\frac{2}{3}k_{\mathrm{inlet}})^{1/2}$. The inlet droplet diameter was sampled from a normal distribution with mean 20 μm and a standard deviation of 2 μm. To capture the dipole–dipole forces and the film-thinning forces, the virtual radius in the cluster integration method was set to $r_{v,i} = 3r_i$.

A global time step of $\Delta t = 10^{-4}$ s was used, where the local time step was determined by a relative tolerance of 10^{-4} in the variable step-size Runge–Kutta scheme of Dormand and Prince (1980). The local time step δt was limited downwards to $\frac{1}{3} \times 10^{-7}$ s. 10^4 global time steps were carried out in each droplet tracking. The physical time of one droplet tracking was then 1 s. During 1 s, typically 10^6 droplets were inserted at the inlet.

4. Results

The mean fluid velocity along the centerline of the channel is plotted in Fig. 4. The dashed line represents the fluid velocity when the droplets are taken into account and the continuous line represents the fluid velocity without droplets. These curves are compared with the mean droplet velocity represented by the small triangles in Fig. 4. The mean-fluid velocity is slightly increased due to the presence of the droplets. The droplets follow the fluid motion and move at the same velocity as the fluid. Fig. 5 shows that the turbulent kinetic energy k of the fluid phase does not change when the droplets are added to the fluid. The continuous and the dashed lines in Fig. 5 are almost identical. The difference between the turbulent kinetic energy "seen" by the droplets and the turbulent kinetic energy of the fluid phase is due the assumption of homogeneous turbulence in Eq. (16). The effect of this difference is considered negligible within the present investigation.

4.1. Effect of the electric field

The mean droplet diameter along the channel centerline is plotted in Fig. 6, where different electric field magnitudes are applied perpendicularly to the fluid flow. The high turbulence level and the large number of droplets in the proximity of the channel inlet give a high collision frequency and a rapid droplet growth. Further downstream ($y > 1$ cm), where the turbulence is less intense, the calculated mean droplet diameter seems to be independent of the applied electric field for moderate field magnitudes $E_0 \leqslant 250\,\mathrm{V/mm}$. A significantly faster droplet growth is experienced for the strongest field. Similar results were observed experimentally by Harpur et al. (1997). A dimensioning parameter for the gravity separator vessel placed after the in-line electrocoalescer in the process chain is the diameter $d_{V0.5}$.

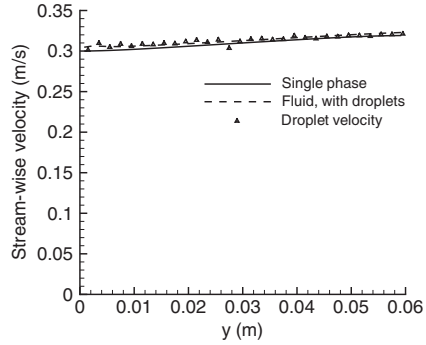


Fig. 4. Mean velocities along the centerline.

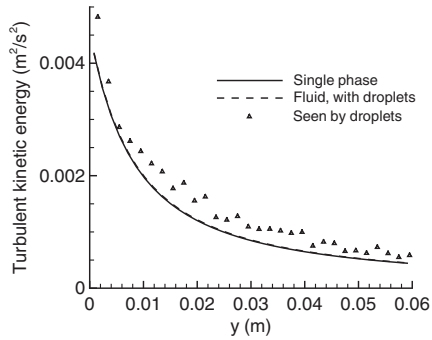


Fig. 5. Turbulent kinetic energy k along the centerline, with and without droplets.

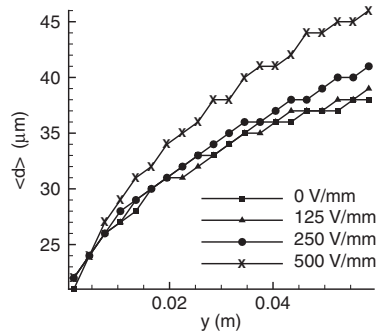


Fig. 6. Mean droplet diameter along the centerline for different electric field.

The $d_{V0.5}$ diameter is the largest diameter satisfying the following expression:

$$\frac{\text{The sum of the volume of all droplets with } d_d \leqslant d_{V0.5}}{\text{Total volume}} < 0.5\%. \tag{29}$$

Fig. 7 shows the effect of the applied electric field on the size of the $d_{V0.5}$ droplet at the channel outlet. The dashed

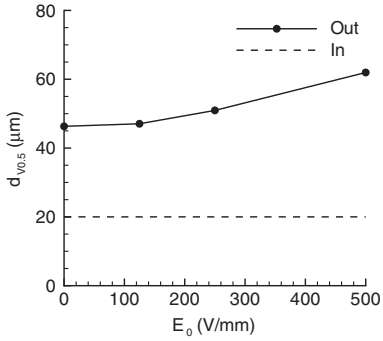


Fig. 7. $d_{V0.5}$-droplet versus electric field.

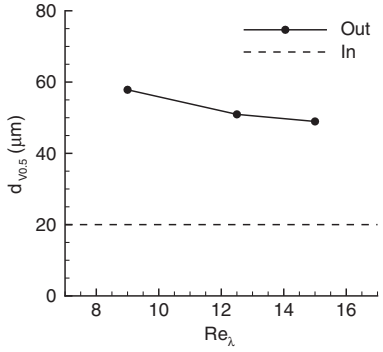


Fig. 9. $d_{V0.5}$ versus the turbulence level at the inlet.

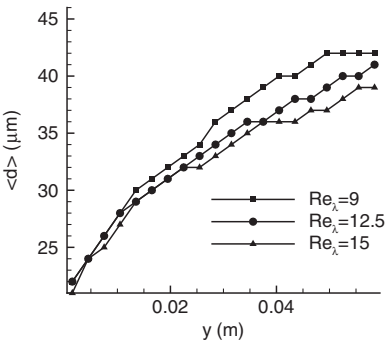


Fig. 8. Mean droplet diameter along the centerline for different turbulence levels.

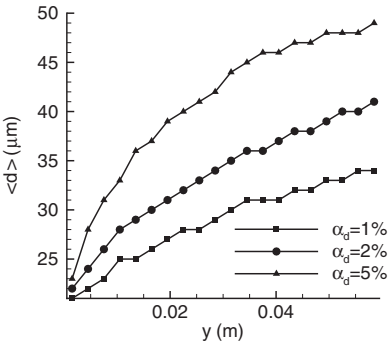


Fig. 10. Mean droplet diameter along the centerline or different droplet volume fractions α_d.

line shows the mean diameter of the in-flowing droplets and the solid line the droplet with diameter $d_{V0.5}$ at the outlet. The $d_{V0.5}$ droplet diameter increases for stronger electric fields.

4.2. Effect of turbulence

Fig. 8 shows the mean diameter along the centerline for different turbulence levels. An electric field of 250 V/mm perpendicular to the direction of the emulsion flow was applied. We observe that the droplet growth is fastest for the lowest turbulence level. Higher turbulence regime corresponds to larger emulsion flow velocity, and thus at $Re_\lambda = 12.5$ and 15 the emulsion flow is two and three times larger than in the lowest turbulence case, $Re_\lambda = 9$. The emulsion residence time under the influence of the electric field is longer at the low turbulent regime. This explains why the droplet growth is fastest at low turbulence regime. Towards the outlet of the channel, the droplet diameter increases faster at higher turbulence levels. The droplet diameter $d_{V0.5}$, shown in Fig. 9, decreases only slightly for a higher flux of mixture, due to the increase in the turbulence level.

4.3. Effect of volume fraction

The droplet growth at different droplet volume fractions α_d is shown in Fig. 10. An electric field of 250 V/mm perpendicular to the direction of the emulsion flow was applied. Unsurprisingly, the droplets grow faster when they are close to each other at high droplet volume fractions. Close to the inlet, where there are many small droplets and a high turbulence level, the growth is particularly fast in the $\alpha_d = 5\%$ case. The change in the diameter $d_{V0.5}$, shown in Fig. 11, is more interesting for the performance of the electrocoalescer. In the $\alpha_d = 1\%$ case, one only needs to separate 50% of the dispersed phase in order to achieve an oil of export quality, while when the volume fraction is $\alpha_d = 5\%$, one has to get rid of 90% of the droplets to achieve $\alpha_d = 0.5\%$. We observe in Fig. 11 that the droplet diameter $d_{V0.5}$ is almost independent of droplet volume fraction.

5. Discussion

In the present work, the "steady-state" RANS closed with the k–ε turbulence model was combined with a DPM to investigate the combined effect of turbulence and electric dipole–dipole forces in two dimensions. The instantaneous fluid velocity

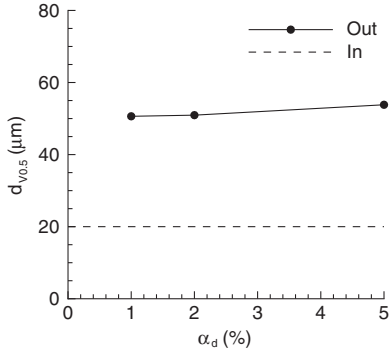


Fig. 11. Terminal velocity for the $d_{V0.5}$-droplet versus the volume fraction of droplets α_d.

"seen" by the droplets is realized by evolving in time a stochastic differential equation for the fluctuating fluid velocity at the positions of the droplets, where the fluid velocities seen by close droplets are correlated. The viscous effects are neglected in the model that is used to correlate close fluid velocities, Eq. (20). Hence, the velocities of very close droplets are not so well-correlated as they should be in the viscous sub-range. As a consequence, frequencies of collisions and collision impact velocities might have been over-predicted. This partially explains the fast droplet growth when no electric field was applied. Strong attractive electric forces will determine the motion of close droplets, and thereby reduce the importance of the modeling of the viscous sub-range when high voltages are applied.

The coalescence efficiency is the probability of two droplets coalescing when they are in near contact. There are several parameters involved in this process, such as the impact velocity, the electrostatic forces and properties of the droplet surfaces. Due to the lack of a precise model for coalescence efficiency, each collision leads to coalescence in our approach. This might be an adequate approximation when electric fields are applied and when the water-droplet surfaces are not too contaminated. The validity of the approximation is more doubtful in the case with no applied electric field.

A channel flow with almost isotropic, decaying turbulence is calculated and the effects of electric field, turbulence level and droplet volume fraction are investigated. The macroscopical behavior of the water-in-oil emulsion obtained numerically shows the same trends as the experimental results obtained by Harpur et al. (1997). Unfortunately, there is a lack in the literature of experimental work that provide more detailed data.

In the present investigation, the electric field magnitude is varied. This exercise shows, as expected, that the strongest field gives the best coalescence rate and therefore the best efficiency of the turbulent electrocoalescer. From the dipole–dipole force expression, we observe that the electric forces are proportional to the square of the electric field magnitude. Although the results without an electric field are not reliable due to the assumption of successful coalescence at every contact, it is observed that the weakest electric fields do not increase the number of droplet contacts. Weak electric fields seem only to assist the coalescence process, an observation which agrees with the conclusions of Harpur et al. (1997).

The influence of the turbulence level on the coalescer efficiency is investigated by varying the fluid velocity. The reason why the droplet growth is slightly faster for the lowest turbulence regime $Re_\lambda = 9$ is due that the emulsion residence time under the influence of the electric field is longer. One can still observe that the increased turbulence level to some extent balances the shorter residence time. There is almost no difference in the most turbulent region ($y < 2$ cm). One should therefore try to keep the turbulence level high. However, turbulent break-up provides limitations to the maximum stable droplet size, and that is of order $d_{p,\max} = O(\sigma^{0.6}\varepsilon^{-0.4}\rho^{-0.6})$ (Hinze, 1955). Because the aim of turbulent electrocoalescence is to remove the smallest droplets by letting them coalesce, break-up of droplets larger than $\approx 250\,\mu$m does not matter. A maximum stable droplet diameter of 250 µm gives a maximum dissipation ε of order $10^2\,\mathrm{m}^2/\mathrm{s}^3$, which is far higher than the values in the present work ($\varepsilon_{\max} = 0.73\,\mathrm{m}^2/\mathrm{s}^3$).

Given a constant electric field magnitude of 250 V/mm applied perpendicularly to the direction of the emulsion flow, one can observe that it is slightly favorable to treat a $\alpha_d = 5\%$ emulsion compared to a $\alpha_d = 2\%$ or a $\alpha_d = 1\%$ emulsion. This is due to the fact that the water droplets are closer to each other at high droplet volume fractions than at small droplet volume fractions for emulsions with the same size distribution. However, contrary to traditional methods with a laminar-flow emulsion, which need a certain volume fraction of droplets to be efficient (Bailes and Larkai, 1984), the turbulent coalescer tested here is equally efficient over the range of tested volume fractions.

6. Conclusions

A model of a turbulent electrocoalescer is studied numerically in the Eulerian–Lagrangian framework proposed here. The following has been observed:

- Turbulence intensity is an important parameter to achieve frequent droplet collisions, which may lead to coalescence and, therefore, to an efficient electrocoalescer.
- Strong electric fields increased the number of droplet contacts, while weaker electric fields only assisted the process of coalescence.
- Even a short treatment with electric field increased the droplet diameter $d_{V0.5}$ considerably.
- Turbulent electrocoalescence worked equally well for the tested range of droplet volume fractions.

Notation

A	projected droplet area, see Eq. (6), m^2
C_0	Kolmogorov constant, dimensionless

C_d	drag coefficient, dimensionless
C_l	ratio between the Lagrangian and Eulerian time scale, dimensionless
$\boldsymbol{d}$	inter-droplet distance vector, m
$\langle d \rangle$	number mean droplet diameter, m
$d_{V0.5}$	dimensioning diameter, see Eq. (29), m
d_d	droplet diameter, m
$\mathrm{d}W_i$	Wiener process, $\mathrm{s}^{1/2}$
$\boldsymbol{E}_0$	electric background field, V/m
$\boldsymbol{F}_A$	added mass force, see Eq. (10), N
$\boldsymbol{F}_B$	buoyancy force, see Eq. (9), N
$\boldsymbol{F}_D$	drag force, see Eq. (6), N
$\boldsymbol{F}_{\mathrm{El}}$	electric dipole–dipole force, see Eq. (11), N
$\boldsymbol{F}_{\mathrm{Film}}$	film-thinning force, see Eq. (11), N
g	standard Earth gravity, $g = 9.81\,\mathrm{m/s}^2$
k	turbulent kinetic energy, $\mathrm{m}^2/\mathrm{s}^2$
L_E	Eulerian integral length scale, m
L_E^*	instantaneous Eulerian integral length scale, see Eq. (21), m
m_d	droplet mass, kg
N	number of droplets, dimensionless
r	droplet radius, m
Re_λ	Taylor-scale Reynolds number, $Re_\lambda = ((20/3)\, k^2/\nu\varepsilon)^{1/2}$, dimensionless
St	Stokes number, see Eq. (2), dimensionless
t	time, s
Δt	global time step, s
T_L	Lagrangian integral time scale, s
T_L^*	instantaneous Lagrangian integral time scale, see Eq. (17), s
u_i	fluctuating fluid velocity component, m/s
$\overline{U}_i$	mean fluid velocity component, m/s
$\boldsymbol{U}$	fluid velocity vector, m/s
$\boldsymbol{V}$	droplet velocity vector, m/s
V_d	droplet volume, m^3
V_s	droplet terminal velocity, see Eq. (1), m/s
$\boldsymbol{x}$	droplet position vector, m

Greek letters

α	volume fraction of fluid phase, dimensionless
α_d	droplet volume fraction, dimensionless
δ_t	local time step, s
ε	dissipation rate of turbulent kinetic energy, $\mathrm{m}^2/\mathrm{s}^3$
$\varepsilon_{\mathrm{oil}}$	permittivity of oil, dimensionless
ε_w	permittivity of water droplets, dimensionless
θ	angle between $\boldsymbol{E}_0$ and $\boldsymbol{d}$, dimensionless
μ	dynamic viscosity of continuous phase, kg/m/s
ν	kinematic viscosity of continuous phase, m^2/s
ν_t	turbulent viscosity, see Eq. (25), m^2/s
ρ	density of continuous phase, $\mathrm{kg/m}^3$
ρ_d	droplet density, $\mathrm{kg/m}^3$
σ	surface tension, N/m
τ_d	droplet response time, s
τ_η	Kolmogorov time scale, s
ω	turbulence frequency, s^{-1}
$\langle \omega \rangle$	mean turbulence frequency, $\langle \omega \rangle \equiv \varepsilon/k$, s^{-1}

References

Abrahamson, J., 1975. Collision rates of small particles on a vigorously turbulent fluid. Chemical Engineering Science 30, 1371–1379.

Atten, P., 1993. Electrocoalescence of water droplets in an insulating liquid. Journal of Electrostatics 30, 259–270.

Bailes, P.J., Larkai, S.K.L., 1984. Influence of phase ratio on electrostatic coalescence of water-in-oil dispersions. Chemical Engineering Research and Design 62 (1), 33–38.

Burrage, K., Burrage, P.M., 1996. High strong order explicit Runge–Kutta methods for stochastic ordinary differential equations. Applied Numerical Mathematics 22, 81–101.

Chen, T.Y., Mohammed, R.A., Bailey, A.I., Luckham, P.F., Taylor, S.E., 1994. Dewatering of crude oil emulsions. 4. Emulsion resolution by the application of an electric field. Colloids and Surfaces A: Physicochemical and Engineering Aspects 83, 273–284.

Chiesa, M., Melheim, J.A., Pedersen, P.A., Ingebrigtsen, S., Berg, C., 2005. Forces acting on water droplets falling in oil under the influence of an electric field: numerical predictions versus experimental observations. European Journal of Mechanics B-Fluids 24, 717–732.

Clift, R., Grace, J.R., Weber, M.E., 1978. Bubbles, Drops, and Particles. Academic Press, New York.

Dormand, J.R., Prince, P.J., 1980. A family of embedded Runge–Kutta formulae. Journal of Computational and Applied Mathematics 6, 19–26.

Friedemann, J.D., Nilsen, P.J., Sæther, O., Sjöblom, J., 2001. An alternative method for establishing parameters for novel coalescer designs. In: Fourth International Symposium on the Chemistry and Physics of Petroleum–Water Emulsions II.

Hansen, E.B., 2005. Numerical simulation of droplet dynamics in the presence of an electric field. Ph.D. Thesis, Norwegian University of Technology and Science (NTNU).

Harpur, I.G., Wayth, N.J., Bailey, A.G., Williams, T.J., Urdahl, O., 1997. Destabilisation of water-in-oil emulsions under the influence of an A.C. electric field: experimental assessment of performance. Journal of Electrostatics 40 & 41, 135–140.

Hinze, J.O., 1955. Fundamentals of the hydrodynamic mechanism of splitting in dispersion processes. A.I.Ch.E. Journal 1, 289–295.

Kruis, F.E., Kusters, K.A., 1997. The collision rate of particles in turbulent flow. Chemical Engineering Communications 158, 201–230.

Launder, B.E., Spalding, D.B., 1974. The numerical computation of turbulent flow. Computational Methods in Applied Mathematics 3, 269–289.

Loth, E., 2000. Numerical approaches for motion of dispersed particles, droplets and bubbles. Progress in Energy and Combustion Science 26, 161–223.

Melheim, J.A., 2005. Cluster integration method in Lagrangian particle methods. Computer Physics Communications 171, 155–161.

Melheim, J.A., 2006. Correlated motion of inertial particles in turbulent flows. European Journal of Mechanics B-Fluids, submitted for publication.

Melheim, J.A., Chiesa, M., Gjelsvik, A., June 2005. Formulation of numerical performance of an adaptive algorithm for efficient collision detection. In: 2005 ASME FEDSM, Algorithmic Developments in CFD, Houston, TX, USA.

Patankar, S.V., 1980. Numerical Heat Transfer and Fluid Flow. McGraw-Hill, New York, pp. 79–131.

Pope, S.B., 2000. Turbulent Flows. Cambridge University Press, Cambridge, UK, p. 93.

Pope, S.B., Chen, Y.L., 1990. The velocity–dissipation probability density function model for turbulent flows. Physics of Fluids 2 (8), 1437–1449.

Saffman, P.G., Turner, J.S., 1956. On the collision of drops in turbulent clouds. Journal of Fluid Mechanics 1, 16–30.

Sams, G.W., Zaouk, M., 2000. Emulsion resolution in electrostatic processes. Energy and Fuels 14, 31–37.

Sigurgeirsson, H., Stuart, A., Wan, W.-L., 2001. Algorithms for particle-field simulations with collisions. Journal of Computational Physics 172, 766–807.

Siu, Y.L., Jones, T.W., Yu, K., 2001. Interparticle force in polydisperse electrorheological fluid: beyond the dipole approximation. Computer Physics Communications 142, 446–452.

Sjöblom, J., Aske, N., Auflem, I.H., Brandal, Ø., Havre, T.E., Sæther, Ø., Westvik, A., Johnsen, E.E., Kallevik, H., 2003. Our current understanding of water-in-crude oil emulsions. Recent characterization techniques and high pressure performance. Advances in Colloid and Interface Science 100–102, 399–473.

Sommerfeld, M., 2001. Validation of a stochastic Lagrangian modelling approach for inter-particle collisions in homogeneous isotropic turbulence. International Journal of Multiphase Flow 27, 1829–1858.

Squires, K.D., Eaton, J.K., 1991. Preferential concentration of particles by turbulence. Physics of Fluids 3 (5), 1169–1178.

Sundaram, S., Collins, L.R., 1997. Collision statistics in an isotropic particle-laden turbulent suspension. Part 1. Direct numerical simulations. Journal of Fluid Mechanics 335, 75–109.

Takeda, H., Esaki, N., Doi, K., Murakami, H., Yamasaki, K., Kawase, Y., 2004. Flow simulation in bubble columns in regard to bubble coalescence and break-up utilizing LES and DEM. Journal of Chemical Engineering of Japan 37, 976–989.

Vinogradova, O.I., 1995. Drainage of a thin liquid film confined between hydrophobic surfaces. Langmuir 11, 2213–2220.

Vojir, D.J., Michaelides, E.E., 1994. Effect of the history term on the motion of rigid spheres in a viscous fluid. International Journal of Multiphase Flow 20 (3), 547–556.

Wang, L.-P., Wexler, A.S., Zhou, Y., 2000. Statistical mechanical description and modelling of turbulent collision of inertial particles. Journal of Fluid Mechanics 415, 117–153.

Williams, J.J.E., Crane, R.I., 1983. Particle collision rate in turbulent flow. International Journal of Multiphase Flow 9 (4), 421–435.