JOHANNES KEPLER
UNIVERSITÄT LINZ | JKU

TNF

Technisch-Naturwissenschaftliche
Fakultät

CFD-DEM on Multiple Scales – An Extensive Investigation of Particle-Fluid Interactions

DISSERTATION

zur Erlangung des akademischen Grades

Doktorin

im Doktoratsstudium der

Technischen Wissenschaften

Eingereicht von:
Alice Hager

Angefertigt am:
Christian Doppler Labor für partikuläre Strömungen

Beurteilung:
Priv.-Doz. Dipl.-Ing. Dr. Stefan Pirker (Betreuung)
Ao. Univ.-Prof. Dipl.-Ing. Dr. Christian Weiß

Mitwirkung:
Dipl.-Ing. Dr. Christoph Goniva

Linz, April, 2014

Abstract

The objective of the presented thesis was to find and apply numerical methods for calculating different particle laden flows.

For this purpose a coupling between Computational Fluid Dynamics (CFD) and Discrete Element Method (DEM) was used. Depending on the flow situation either a resolved or an unresolved CFD-DEM method was applied. With resolved methods the behaviour of few, large objects can be investigated in much detail. A parallelized Fictitious Domain Method (FDM) was developed, implemented and validated. Unresolved CFD-DEM comes to application when the global behaviour of a larger number of particles is of interest. In the presented work this method was used for analysing the behaviour of blast air induced cavities, termed raceways, inside blast furnaces.

After the development and validation phase the FDM was applied to a case that involved some 70 particles. The example was inspired by the findings of Boycott, who could show that the tilting of a test tube, filled with blood, had an effect on the settling behaviour of red blood cells.

The investigation on the formation of raceways was divided into two parts: in a first step a small scale pseudo 2D lab model was built up and a detailed experimental and numerical study was carried out. The results of the lab-tests and the simulations were in good accordance, yet remarkable influences on the process by the 2D nature of the setup could be observed. This led to step two of the project, which consisted of the simulation of a full size blast furnace part in 3D.

For both the implementation of the resolved method and the application examples CFDEM-project was used. The newly developed tools were included in the Open Source project, and solvers provided by the toolbox were applied.

Kurzfassung

Das Ziel der voliegenden Arbeit war es, numerische Verfahren für die Berechnung unterschiedlicher partikelbeladener Strömungen zu finden und anzuwenden.

Zu diesem Zweck wurden computergestütze Strömungssimulationen (CFD) mit Partikelsimulationen (DEM) gekoppelt. Abhängig von der jeweiligen Strömungssituation wurde entweder eine "aufgelöste" oder eine "nicht aufgelöste" CFD-DEM-Methode verwendet. Im ersten Fall handelt es sich um ein Verfahren, bei dem die Dynamik weniger, großer Objekte detailliert untersucht wird. Es wurde eine parallelisierte "Methode fiktiver Bereiche" (FDM) entwickelt, implementiert und validiert. Nicht aufgelöste CFD-DEM hingegen kommt zur Anwendung, wenn das globale Verhalten einer großen Anzahl kleiner Partikel betrachtet werden soll. Dieses Verfahren wurde zur Untersuchung der Entstehung und des Verhaltens von Windformen im Hochofen verwendet.

Nach der Entwicklungs- und Validierungsphase wurde die FDM auf einen Fall angewandt, der in etwa 70 Partikel beinhaltete. Das Beispiel war von Boycotts Entdeckung hinsichtlich des geänderten Sedimentationsverhaltens von Erythrozyten bei veränderter Reagenzglasposition inspiriert.

Die Hochofen-Berechnungen gliederten sich in zwei Teile: zunächst wurde ein skaliertes pseudo-2D Modell erstellt und sowohl experimentall als auch numerisch untersucht. Die Ergebnisse der Labor-Tests und der Simulationen lieferten gute Übereinstimmungen. Es zeigte sich allerdings, dass die Zweidimensionalität einen starken Einfluss auf den Enstehungsprozess der Windformen hat. Der zweite Teil des Projekts bestand deshalb aus der Simulation eines dreidimensionalen Hochofenteils.

Sowohl für die Implementierung des aufgelösten Teils als auch für die Anwendungsbeispiele wurde das Programmpaket CFDEMproject benutzt. Die neu entwickelten Komponenten wurden in das Open Source Projekt CFDEMcoupling eingebunden und für die nicht aufgelösten Rechnungen wurden die von der Software zur Verfügung gestellten Solver verwendet.

Acknowledgements

I would like to express my gratitude to all those, who supported me writing this thesis.

My thanks go to Priv.-Doz. Stefan Pirker, head of the Christian Doppler Laboratory on Particulate Flow Modelling and of the Department of Particulate Flow Modelling, for giving me the chance to join his research-group and for his supervision. I would like to thank Dr. Christoph Goniva for all the valuable discussions and the great cooperation within the projects. Also, I am grateful to Dr. Christoph Kloß for sharing his expertise in the field of discrete particle simulations. Furthermore, I would like thank Prof. Christian Weiß for reviewing my thesis. All my colleagues from Christian Doppler Laboratory on Particulate Flow Modelling and the Institute of Fluid Mechanics and Heat Transfer provided a very pleasant working-atmosphere that I am most grateful for.

My thanks also go to the Christian Doppler Society and all the industrial partners of the Christian Doppler Laboratory on Particulate Flow Modelling, especially Siemens VAI Metals Technologies and voestalpine Stahl GmbH, whose financial support made this project possible.

I would also like to thank all my friends for accompanying on this journey, for providing the distraction one sometimes needs and for putting up with me in the harder times.

Finally, my biggest thanks go to my family, who has always supported me in every possible way and who gave me the opportunity to choose this way of life and study.

Publications

Paper in Refereed Journal

Hager A., Kloß C., Pirker S., Goniva C. (2013). Parallel Open Source CFD-DEM for Resolved Particle-Fluid Interaction. Journal of Energy and Power Engineering, Volume 7, Number 9, Page(s) 1705-1712.
Hager A., Kloß C., Pirker S., Goniva C. (2014). Parallel Resolved Open Source CFD-DEM: Method, Validation and Application. Journal of Computational Multiphase Flow, Volume 6, Number 1, Page(s) 13-27.

Proceedings

Hager A., Kloß C., Pirker S., Goniva C. (2012), Parallel Open Source CFD-DEM for Resolved Particle-Fluid Interaction, Proc. 9th International Conference on CFD in the Minerals and Process Industries (CFD 2012).
Goniva C., Kloß C., Hager A., Wierink G., Pirker S. (2011). A multi-purpose Open Source CFD-DEM approach, Proceedings of the 8th International Conference on CFD in Oil and Gas, Metallurgical and Process Industries (CFD 2011).
Hager A., Kloß C., Pirker S., Goniva C. (2011). Efficient Realization of a resolved CFD-DEM Method within an open source framework, in: Proc. Open Source CFD International Conference 2011 (OSCIC2011).
Amberger S., Goniva C., Hager A., Kloß C. (2011). Strategies for Simulation Software Quality Assurance applied to Open Source DEM, in: Proc. II International Conference on Particle-based Methods - Fundamentals and Applications (PARTICLES 2011).
Hager A., Kloß C., Goniva C. (2011). Towards an Efficient Immersed Boundary Method Within an Open Source Framework, in: Proc. International Conference on Multiphase Flow in Industrial Plants (MFIP 2011).
Goniva C., Kloß C., Hager A., Pirker S. (2010). An Open Source CFD-DEM Perspective, in: Proc. of OpenFOAM Workshop, Series OpenFOAM Workshop.

Publication in non-refereed Conference Proceedings

Hager A., Doppelhammer N., Kloß C., Pirker S., Goniva C. (2013). On the formation of blast furnace raceways - a combined experimental and open source CFD-DEM investigation, in: MEI (Eds.): Proceedings of Computational Modelling '13.
Hager A., Kloß C., Pirker S., Goniva C. (2012). Improvement of a Resolved Open Source Fluid-Particle Interaction Method, in: ECCOMAS 2012, September 10-14, Vienna, Austria.

Eidesstattliche Erklärung

Ich erkläre an Eides statt, dass ich die vorliegende Dissertation selbstständig und ohne fremde Hilfe verfasst, andere als die angegebenen Quellen und Hilfsmittel nicht benutzt bzw. die wörtlich oder sinngemäß entnommenen Stellen als solche kenntlich gemacht habe.

Die vorliegende Dissertation ist mit dem elektronisch übermittelten Textdokument identisch.

Contents

Introduction

Philosophers and scholars in ancient Greece already stated: „Panta rhei" , which - directly translated - means "everything flows". Whichever interpretation one chooses, it is undisputed that fluids are encountered everywhere. Their motion impacts every aspect of life, no matter if one regards phenomena in nature or man-made machines and processes. The investigation and prediction of events of all kinds has been in the centre of interest ever since. First observations have led to fundamental theories, and in the course of time the methods have become more and more sophisticated. Experiments are nowadays often performed with the aid of complex apparatuses and analytical equations for describing different systems have been developed. At the beginning of the 20th century, first approaches towards numerical methods were made. In the early days, the calculations were still carried out manually, but the introduction of computers opened a completely new prospect and led to remarkable advances in the field. In contrast to the beginnings, where only small test cases could be evaluated, today high-performance computers are used that allow for the simulation of real-world problems.

There are only few practical applications that involve pure single-phase problems. Often more than one fluid is involved, or solids interact with the flow. It is the latter point that this thesis focuses on: the interaction of fluids and solid particles, or rather the treatment of particle laden flow. Countless examples for the presence of such regimes can be given, ranging from colliding billiard balls over blood cells in plasma to coke and iron ore chunks in steel moulding devices such as blast furnaces. The diversity of these examples already suggests that the ways which lead to "the optimal solution" in each case might be different ones: some of the problems involve few particles and require a very detailed resolution, others contain huge numbers of particles and the main focus lies on the global behaviour. In some approaches the particle phase is treated like a second fluid phase with different properties, in others the particles are influenced by the flow but not vice versa.

The objective of this work was to find a way for handling a possibly wide range of fluid-particle interaction problems. For accomplishing this task, two different methods were chosen. They both calculate the motion of the fluid and the particles separately and account for their interdependency. The treatment of the particle phase is in both cases taken care of by a software package that calculates the trajectory of each particle in the domain individually. Influences such as the gravity or collisions with other particles and walls are considered. This approach is called Discrete Element Method (DEM). For the fluid phase, Computational Fluid Mechanics (CFD) was used. The basic governing equations were modified in two different ways, which then resulted in resolved and unresolved CFD-DEM.

This thesis consists of two parts, each dedicated to one of the methods. The resolved CFD-DEM approach was implemented in the course of this thesis, thus a strong focus in this

part lies on its realization and validation. The unresolved CFD-DEM method had already been implemented. Part II mainly deals with the theoretical background and its application to an industrial size problem. Chapter 1 - termed "Vision" - portrays the thesis, giving the reader an idea about its contents.

The calculations were carried out with the software package CFDEMcoupling [2012]. It combines OpenFOAM [2011]-based finite volume solvers and the DEM toolbox LIGGGHTS [2012]. The software is fully Open Source, and the added features, i.e. the resolved CFD-DEM solver and the models developed in this context, have been made accessible to the public.

1 Vision

The investigation of granular materials, their properties and their behaviour in general has been of major interest for a long time. The methods that can be used for analysing the materials are almost as numerous as the different forms bulk solids can take. A very important approach in this context is the experimental investigation of processes. Unfortunately, in many cases difficult environmental conditions hinder the extraction of data, or the conduction of multiple test runs would require too many resources. An alternative to experiments are numerical simulations. Since the introduction of digital computers they experienced extensive developments. Even the world's first electronic digital computer, ENIAC, was used for the simulation of granular materials (cf., Weik [1961]).

The simulation methods range from Monte Carlo methods (cf., Rosato et al. [1986]) over event driven methods (cf., Luding et al. [1994]) to cellular automaton models (cf., Baxter and Behringer [1991]). Recently the most-used methods are those based on the idea of molecular dynamics (MD), such as for example the **Discrete Element Method (DEM)**. In the literature it is also referred to as Distinct Element Method. In this method, the trajectory of each particle is calculated separately, using Newton's second law. Collisions amongst particles or between particles and walls are accounted for. During the calculations, position and velocity of each particle are saved. In other words, the observer follows each particle, which makes the method a Lagrangian one. A brief introduction to the method and its governing equations is given in Appendix B. Two ways for modelling particle-contacts were established, namely the hard and the soft sphere approach. While the hard sphere model is restricted to applications, where binary collisions prevail, the soft sphere approach can be employed to all kinds of regimes.

In most applications the granular materials are surrounded by fluid that influences their behaviour. In some cases its effect might be negligible, but usually an accurate depiction of a process depends on the fluid's consideration. A widely used procedure for processing pure fluid problems is **Computational Fluid Dynamics (CFD)**. The method uses a decomposition of the computational domain, termed mesh, and solves the discretized governing

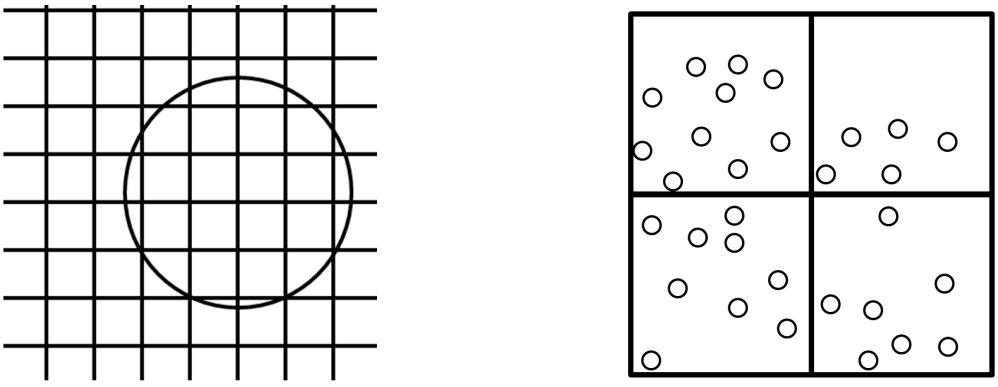
equations on each cell of the mesh. The fluid motion in this case is regarded at fixed locations. Such an approach is termed Eulerian. Appendix A is concerned with the governing equations and their numerical consideration in the framework of a finite volume method.

If one wants to work on problems that involve interacting fluids and solid particles, a coupling between CFD and soft sphere DEM is advisable: in both methods force terms account for external influences, so why not extend each of the equations by another term, that accounts for the interaction between fluid and solid. This idea is based on Newton's third law, which states "actio est reactio". The resulting class of **coupled CFD-DEM** methods is of Eulerian-Lagrangian nature.

Fluid-particle problems occur in most different forms. On the one hand, the particles themselves can range from sticky powders over almost spherical granules of few millimetres in diameter to non-spherical chunks of coke and even further to stars and planets. On the other hand, also the ratio of particle size to total problem size matters: in some cases one wants to investigate on the exact behaviour of few relatively large particles, in others the global behaviour of a big amount of rather small objects is of interest. Due to the varying challenges one differs - amongst others - between resolved and unresolved CFD-DEM.

In **resolved CFD-DEM** the particles are remarkably larger than the fluid cells, which means that when represented within the mesh a particle covers several cells at once (cf., Fig. 1.1a). This method is used for cases, where only small amounts of particles - say a couple of hundreds or thousands - need to be considered. The fluid field around each particle is resolved is with much detail, and the force on each particle is calculated individually. This method belongs to the class of Direct Numerical Simulations (DNS).

In contrast to that **unresolved CFD-DEM**is designed for handling large amounts of particles. The considered particles are significantly smaller than the mesh cells, consequently a cell can contain several particles at once (cf., Fig. 1.1b). In this method, the bigger part of the computational costs arises on the DEM side. Depending on the available computational resources, systems containing several millions of objects can be handled.



(a) Resolved CFD-DEM *(b)* Unresolved CFD-DEM

Fig. 1.1: Fluid-particle methods in the presence of a fluid mesh.

If a system contains more particles, than classical unresolved CFD-DEM can deal with, an approach termed "coarse graining" can be applied. In coarse graining, several particles are grouped together and treated as a single object (e.g., Sakai and Koshizuka [2009]; Sakai et al.

[2010, 2012]; Sundaresan et al. [2013]). The properties of these objects are in accordance with the properties of the previously replaced group. This results in a fluid-particle system that is suitable for the application of unresolved CFD-DEM. Fig. 1.2 shows a simple 2D system before and after coarse graining. The trade-off of assuming equal properties for all particles of a group is a loss of resolution and accuracy. In order to remedy this, additional models, as proposed by Radl et al. [2011] were developed. The matter of coarse graining was not addressed.

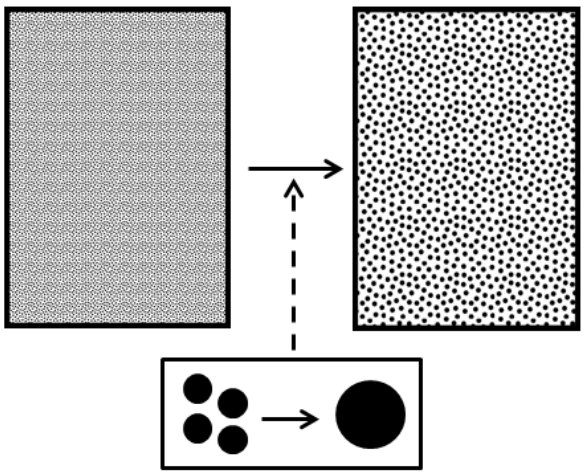


Fig. 1.2: A dense particle bed before (left) and after (right) coarse graining.

This thesis comprises two parts, one dealing with resolved and one with classical unresolved CFD-DEM. The overall structure of the two parts looks alike: topics such as governing equations, implementation and validation/application are covered. All calculations as well as the new developments were carried out in the framework of CFDEMcoupling [2012]. CFDEMcoupling is a rather young Open Source coupling software. It was mainly developed by Christoph Goniva (cf.,Goniva et al. [2012]), and later on extended with some outcomes of this thesis. CFDEMcoupling combines OpenFOAM-based finite volume solvers and LIGGGHTS, an Open Source DEM code (cf., LIGGGHTS [2012]; Kloss et al. [2012a,b]). All mentioned software components are available for free download, and a growing community makes use of it.

As mentioned above, this thesis consists of two distinct parts:

Part I is dedicated to resolved CFD-DEM. Resolved methods gained more and more importance over the years, and several approaches are documented in the literature. The first task was to identify a method that is suitable for the incorporation to the existing CFDEMcoupling framework. As the fluid phase is calculated with the aid of a CFD solver, the whole computational domain is discretized, resulting in a fluid mesh. The objects move through the domain, thus mesh-conformance or the consideration of sharp surfaces could be outruled from the beginning. Shirgaonkar et al. [2009] presented an approach, which involved an interface condition between fluid and solid. While initially given as stress directly at the particle surface they suggested to transform it to a volume force, which can easily be

represented within a finite volume method. The proposed algorithm was used as guideline for the implementation of a resolved solver within CFDEMcoupling.

After first validations of the basic version it became clear that some improvements were necessary: as the resolved method belongs to the class of Direct Numerical Simulations (DNS), a high resolution of the fluid mesh in the area of the particles was required. Especially for 3D cases the fine meshes caused enormous computational costs for only small problems. The first remedy was provided by OpenFOAM itself, by means of **dynamic local mesh refinement**. The tool can refine the mesh in a desired region, as for example around the particles. When the particles move on, the cells are coarsened again. Especially for dilute particle systems this has a large effect. Another feature an efficient coupled code needs to provide is the possibility to **parallelization**. Both OpenFOAM and LIGGGHTS can be run in parallel using the Message Passing Interface (MPI). In the newly developed solver the particle transition from one processor to another caused troubles. This issue could be overcome by the introduction of an innovative void fraction distribution model.

In its first version, particles were either treated as fluid or as solid. This zero-one representation was rather simple to implement and very efficient in terms of computational costs. Unfortunately this method also had two major drawbacks: on the one hand the representation of the particle was inaccurate and in many cases changed in time, on the other hand fluid regions that were actually physically connected to the remaining fluid were cut off, which caused instabilities in the calculations. Fig. 1.3 illustrates the two phenomena.

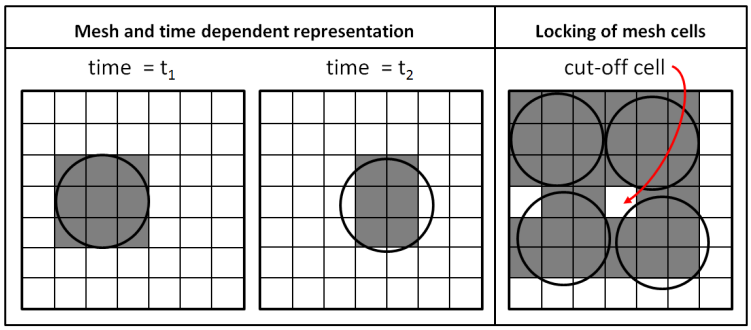


Fig. 1.3: The two drawbacks of the initial particle representation within the mesh: inaccurate and time dependent representation of the particle's volume and instabilities caused by the cutting-off of physically connected mesh regions.

As a remedy to that a **smooth particle representation** was introduced. Here a weighting factor accounted for the amount of fluid/solid inside a cell, increasing both stability and accuracy of the method. The three features that were added to the resolved CFD-DEM solver are summarized in Fig. 1.4.

The complete tool was validated against a number of examples that aimed at different aspects of the method. The problems considered involved both translational and rotational motion. A further example served the investigation of the effect of different initial configurations on the settling behaviour of a group of particles. The last example was inspired by an effect discovered by Boycott in the context of investigating on the settling of red blood cells. Around 70 spheres were released in a rectangular 3D domain, two different configurations were considered: in one case the domain was aligned with gravity, in the second case it

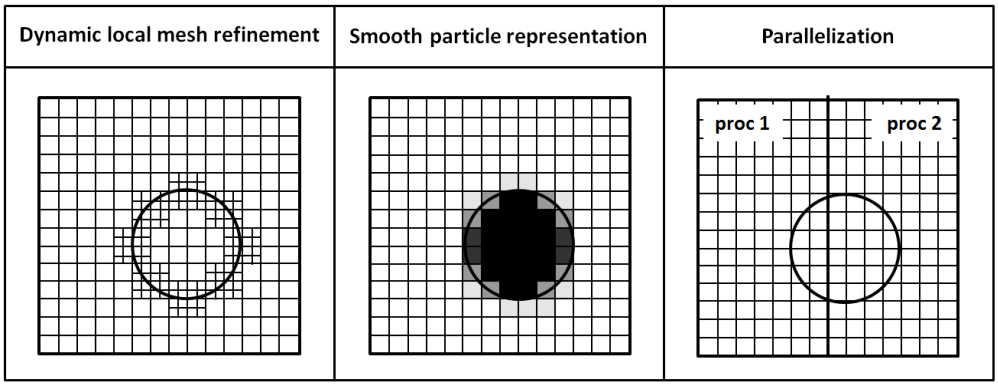


Fig. 1.4: Dynamic local mesh refinement, a smooth particle representation model and the ability for parallelization completed the resolved CFD-DEM solver.

was tilted. The investigation concentrated on the effect, this difference had on the settling behaviour.

In conclusion, the main task in Part I of this thesis was the implementation and validation of a resolved CFD-DEM solver. With the last example the method's applicability to problems involving up to one hundred objects was shown. A further extension to larger problems is possible, yet the fact that the objects are resolved within the mesh imposes certain limits. For larger cases, as the one considered in Part II, a shift to another approach is advisable:

Part II of this thesis is concerned with classical unresolved CFD-DEM. In contrast to the previous case, where a lot of code development was necessary, the required models had already been implemented within CFDEMcoupling. The focus of this part lay on finding a way to simulate an industrial scale problem.

After a brief description of the algorithm itself and a study on time step widths, the body of this part, i.e. the investigations on raceway formation, commences: in steel production, blast furnaces are used for smelting iron. For doing so, blast air enters the system through tuyeres, and causes the break of cavities, termed raceways. Fig. 1.5 contains a schematic sketch of a furnace.

The process itself is very complex, involving all kinds of multiphase flows and chemical reactions. The simulation of the whole operation is a long-term objective, whose first interim goals were to be accomplished within the course of this thesis. More specifically, the work should result in a "cold", i.e. iso-thermed, 3D simulation of a real-size blast furnace. As one cannot simply conduct measurements inside a furnace for validating the results of the calculations, a road map leading from a small lab experiment to a full-scale simulation was established (cf., Fig. 1.6):

Initially a small scale pseudo 2D test rig was set up. The dimensions for the test set-up and the different materials in use were determined by scaling a real-world problem. A number of experiments with different boundary conditions and tuyere diameters were run. For being able to run according simulations, the simulation parameters of the granular materials had to be determined. As some of them cannot be measured directly, an established particle-calibration routine was employed. In both the experiments and the simulations several factors had to be accounted for: In the real process smelting and gasification of the material

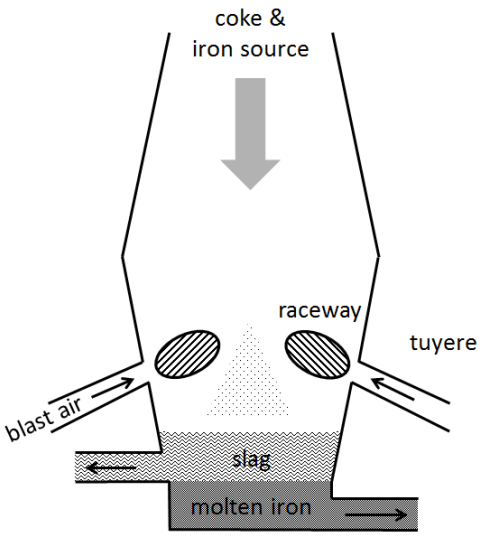


Fig. 1.5: Schematic sketch of a blast furnace.

makes the particle bed move downwards. This motion has a strong impact on the properties of the bed. For reproducing this effect as much as possible, material was extracted from underneath the raceway. When filling an acrylic glass box with granular material, the walls experience a deformation that induces stresses onto the bed. This effect had to be accounted for in the simulations.

After the simulations of the pseudo 2D case, a 3D simulation was set up. The simulation parameters were extracted from data provided by the industrial partners, from in-house experiments and from the literature. A blast furnace is an axial-symmetric device, thus the simulation of a wedge suffices for depicting the effects. At the time being, CFDEMcoupling did not provide axial-symmetric periodic boundary conditions. Thus, for out-ruling wall effects a region containing three tuyeres was regarded.

Fig. 1.6 shows the road map that was followed, starting with a real application, taking a "detour" over the small scale model, and finally resulting in a real-scale simulation.

Altogether, the two parts of the thesis comprise all aspects of numerical modelling: the work in the first part consisted of rather fundamental tasks, such as identifying suitable governing equations, finding a way to implement them, improve the first, running version of the code and finally validate the outcome. By releasing the code on the online platform CFDEMcoupling, the solver was exposed to the critical eyes of the community. In the second part experiment and simulation were developed alongside, and finally a problem of realistic size could be regarded.

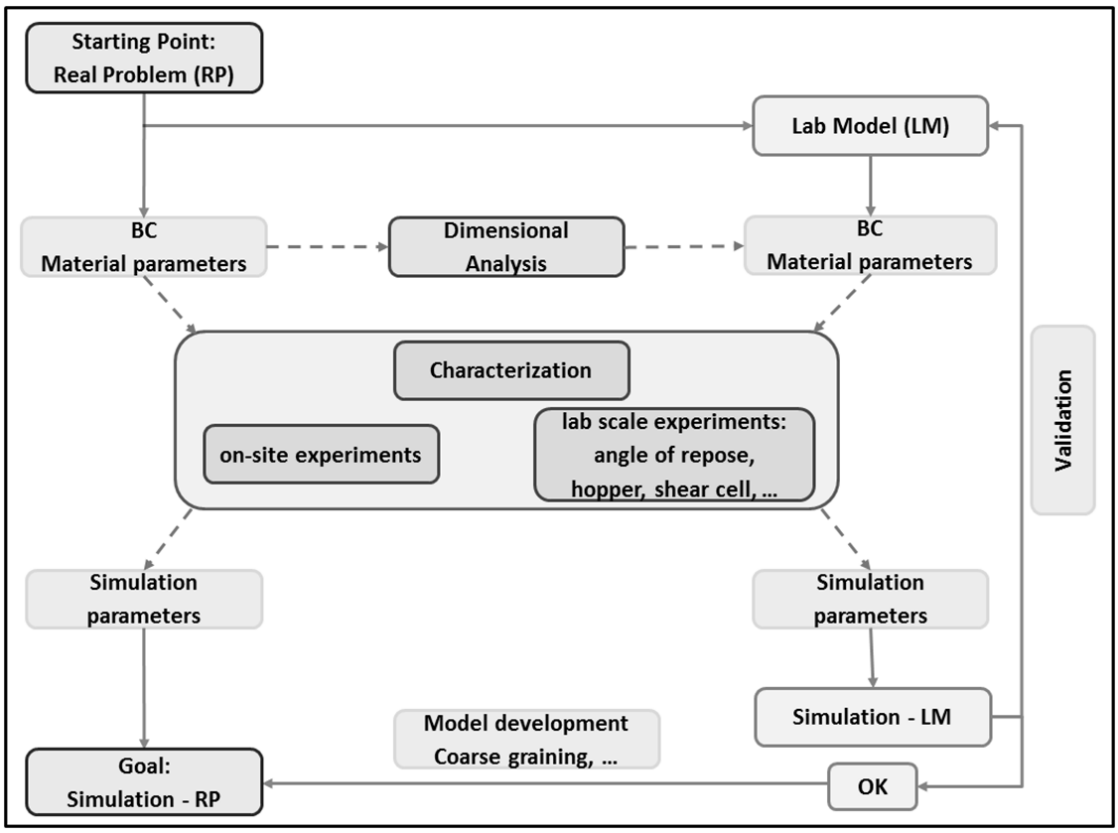


Fig. 1.6: Road map of the procedure for obtaining a cold 3D simulation of a full scale blast furnace.

Part I

RESOLVED CFD-DEM

2
Resolved CFD-DEM

With increasing computational power the Direct Numerical Simulation (DNS) of fluid-particle problems gained more and more importance. The first one to use such a method was Peskin [1972]. In the methods of this class each particle covers several cells of the fluid mesh. One can distinguish between two approaches, one involving body conformal meshes and one where fixed meshes are used. Body conformal mesh methods are of high accuracy but also of high complexity in terms of realization and computational costs (cf., Haeri and Shrimpton [2012]). In the fixed mesh approach the whole computational domain is discretized at once, and both fluid and solid share a single velocity and pressure field. This allows for efficient calculations and thus a fixed mesh method was chosen in this thesis. Fig. 2.1 shows a schematic representation of a body within a 2D fluid domain before and after discretization.

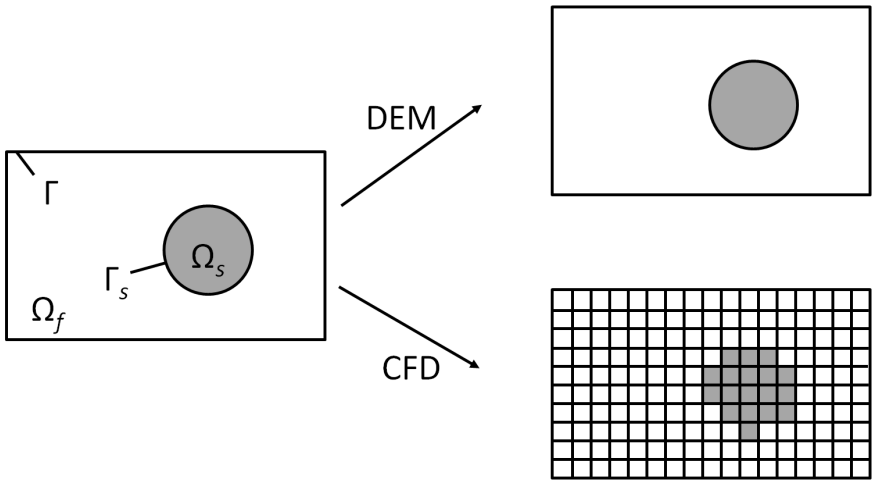


Fig. 2.1: Particle in the fluid domain (left), representation of the particle in DEM (top) and CFD (bottom). Notation: Ω_F denotes the fluid domain, Ω_s the particulate phase, $\Omega = \Omega_F \cup \Omega_s$. The boundary of Ω is given by Γ; Γ_S is the interface between fluid and solid.

The generic term "fictitious domain method" summarizes all kinds of methods that are suitable for describing complex structures within flow fields whose boundary conditions are imposed as force terms. One of these approaches became known as immersed boundary method. It is rather for historical reasons than for substantial ones that this method keeps being mentioned separately. This custom will be maintained in this thesis; hence a brief introduction to the immersed boundary method is given before the presented fictitious method is discussed.

2.1 Immersed Boundary Method

The main issue of this method is adding a force term to the Navier-Stokes equations in order to account for the presence of the bodies. Generally one can distinguish between **continuous** and **discrete forcing** (cf., Mittal and Iaccarino [2005]).

In the continuous approach the force term is added to the undiscretized governing equations. As a consequence this method does not depend on the discretization scheme. It is well suited for the representation of the interaction between fluids and immersed elastic boundaries, as documented by various publications (e.g., Peskin [1972],Lai and Peskin [2000]). Since for rigid bodies the equations tend to become ill-posed, further assumptions are necessary to make this approach applicable for those cases (e.g., Beyer and Leveque [1992]; Stockie and Wetton [1998]; Goldstein et al. [1993]; Uhlmann [2005]).

In case of discrete forcing the governing equations are already discretized when the force terms are introduced. Clearly this evokes a dependence on the chosen discretization, which allows a certain control over the method in terms of accuracy and stability. A detailed discussion of different approaches within the range of the definition are presented by Mittal and Iaccarino [2005].

The immersed boundary method is focused on the treatment of the surface of the objects; the flow that develops in the mesh regions, where the solid is located, is neglected in the calculations. In this area non-physical behaviour can occur, thus a more general perspective of the matter would be of interest.

2.2 Fictitious Domain Methods with Lagrangian Multipliers

Glowinski et al. [1998, 1999, 2001] were the first ones to introduce FDM that made use of Distributed Lagrange Multipliers (DLM). In their concept they integrated Lagrangian multipliers as constraint to the weak form of the Navier-Stokes equations, enforcing the boundary conditions of the rigid body in the system. In their work, the Lagrangian multiplied needed to be calculated explicitly, which made the method rather inefficient. Patankar et al. [2000]; Patankar [2001], Diaz-Goano et al. [2003] and Yu and Shao [2007] improved the method by using body forces and therefore avoiding the necessity of calculating the Lagrangian Multiplier explicitly. Instead, they introduced body forces that enforced the rigidity constraint.

This idea was picked up in the presented solver: according to Patankar a fictitious domain method generally consists of three steps: In a first run, the whole domain is considered as

fluid phase. Secondly, the motion of the bodies is imposed, and finally the body motion is relaxed. In the present case, the procedure is the following:

1. The incompressible Navier-Stokes equations are solved on the whole computational domain.
2. Information about the rigid motion of the bodies - evaluated with the DEM - is incorporated in the form of a body force.
3. The velocity field of the fluid is modified in order to satisfy the conservation equation and a correction of the pressure field takes place.

This method can be used for arbitrary multi-body problems. However, for the sake of clarity, all theoretical considerations are shown for only one body. The collisions between two bodies or a body and the wall are resolved correctly as being part of the DEM calculation. For multi-body examples see Chapter 4.

After this brief description of the procedure, the governing equations of the problem are discussed in the following section.

2.3 Governing equations

The problem is described by a number of equations and conditions that account for the dynamics of the fluid and the particles:

$$\begin{aligned}
\rho\frac{\partial u}{\partial t} + \rho(u\cdot\nabla)u &= -\nabla p + \mu\Delta u \text{ in } \Omega, && (2.1)\\
\nabla\cdot u &= 0 \text{ in } \Omega, && (2.2)\\
u &= u_\Gamma \text{ on } \Gamma, && (2.3)\\
u(x, t=0) &= u_0(x) \text{ in } \Omega, && (2.4)\\
u &= u_i \text{ on } \Omega_s, && (2.5)\\
\sigma\cdot\widehat{n} &= t_{\Gamma_s} \text{ on } \Gamma_s, && (2.6)\\
m_p\frac{du_p}{dt} &= m_p g + f_{p,f} + \sum_{N_p} f_{p,p} + \sum_{N_w} f_{p,w}. && (2.7)
\end{aligned}$$

σ denotes the stress tensor of the fluid field, $\widehat{n}$ is the outer normal vector of the particle and t_{Γ_s} is the traction vector of the fluid, acting on the particle. $f_{p,f}$ is the particle-fluid interaction force, $f_{p,p}$ and $f_{p,w}$ denote the forces, resulting from particle-particle and particle wall contacts.

Eq. (2.1) and (2.2) are the equations of motion and the continuity equation for an incompressible fluid, also termed Navier Stokes equations. These equations hold on the whole domain. The Dirichlet boundary condition (2.3) and the initial condition (2.4) complete the set of flow equations. Eq. (2.7) describes the motion of the Lagrangian particles. Eq. (2.5) and (2.6) are responsible for the coupling between the fluid and the solid phase. Expression 2.5 is used to transfer the body's velocity on the velocity field of the fluid. The second condition, eq. (2.6), referred to as interface condition, describes the stress between fluid and solid. It can be transformed to a drag force term, which will be demonstrated in Sec. 2.4.

The calculation procedure

The algorithm follows the routine introduced in the previous section:

1. First of all an interim velocity field $\widehat{u}$ is calculated by solving the Navier-Stokes equations over the whole domain:

$$\rho\frac{\partial\widehat{u}}{\partial t}+\nabla\cdot(\rho\widehat{u}\widehat{u})=-\nabla p+\mu\Delta\widehat{u} \tag{2.8}$$

   For accomplishing this task, a finite volume based CFD solver comes to application. A PISO routine (cf., Ferziger and Perić [2002]) is employed for evaluating the system of equations. Appendix A.2 gives an overview over numerical details.

2. In the next step the interim velocity field $\widehat{u}$ is corrected in the solid area by imposing the particle's velocity provided by the DEM calculation, leading to a new velocity field $\tilde{u}$.

   Remark: The manipulation of the velocity field is equivalent to adding the body force term

$$f=\rho\frac{\partial}{\partial t}(\tilde{u}-\widehat{u}) \tag{2.9}$$

   to eq. (2.8).

3. The new velocity field $\tilde{u}$ is not divergence-free, thus a corrected field $\overline{u}$ is defined as

$$\overline{u}=\tilde{u}-\nabla\phi. \tag{2.10}$$

   As $\overline{u}$ is defined in such a manner that

$$\nabla\cdot\overline{u}=0 \tag{2.11}$$

   is fulfilled, one automatically obtains a Poisson equation for the correction factor ϕ:

$$\Delta\phi=\nabla\cdot\tilde{u}. \tag{2.12}$$

   With its aid, the velocity field can be corrected, and only the calculation of the final pressure field is left.

4. If one now replaces $\tilde{u}$ in eq. (2.9) with $\overline{u}+\nabla\phi$, this yields

$$\rho\frac{\partial\widehat{u}}{\partial t}=\rho\frac{\partial\overline{u}}{\partial t}+\rho\frac{\partial(\nabla\phi)}{\partial t}-f. \tag{2.13}$$

   Thus if one replaces the transient term in eq. (2.8), this leads to

$$\rho\frac{\partial\overline{u}}{\partial t}+\nabla\cdot(\rho\widehat{u}\widehat{u})=-(\nabla p+\rho\frac{\partial\nabla\phi}{\partial t})+\mu\Delta\widehat{u}+f. \tag{2.14}$$

   So consequently the corrected pressure is expanded by the term $\frac{\partial\phi}{\partial t}$.

The routine of the presented solver follows the ideas of Patankar [2001] and Shirgaonkar et al. [2009].

So far, only the steps conducted by the CFD solver were discussed. The Lagrangian solver for the particulate phase uses Verlet integration (cf., Verlet [1967]) for calculating the trajectories of each particle. Force models are used to account for the contributions stemming from collisions with other particles and walls. The calculation of the force, the fluid exerts on the particles, is discussed.

2.4 Force calculation

The starting point for the calculation of the force, the fluid exert on the particle is the interface condition (2.6). In a first step, the condition is integrated over the body's boundary Γ_s:

$$\int_{\Gamma_s} \sigma \cdot \widehat{n}\, d\Gamma_s = \int_{\Gamma_s} t_{\Gamma_s}\, d\Gamma_s \tag{2.15}$$

Applying the divergence theorem (cf., Appendix C.1) yields

$$\int_{\Omega_s} \nabla \cdot \sigma\, d\Omega_s = \int_{\Gamma_s} t_{\Gamma_s}\, d\Gamma_s. \tag{2.16}$$

As the fluid is incompressible, the relation

$$\sigma = -p\, I + \tau \tag{2.17}$$

holds for the stress tensor. Inserting eq. (2.17) into eq. (2.16) then leads to

$$\int_{\Omega_s} -\nabla p + \nabla \cdot \tau d\Omega_s = \int_{\Gamma_s} t_{\Gamma_s}\, d\Gamma_s. \tag{2.18}$$

For Newtonian fluids τ is given by

$$\tau = 2\mu D(u) = \mu(\nabla u + (\nabla u)^T). \tag{2.19}$$

As a consequence, (2.18) becomes

$$\int_{\Omega_s} -\nabla p + \nabla \cdot (\mu(\nabla u + (\nabla u)^T))\, d\Omega_s = \int_{\Gamma_s} t_{\Gamma_s}\, d\Gamma_s. \tag{2.20}$$

It can be shown (cf., Appendix C.2) that

$$\nabla \cdot (\nabla u + (\nabla u)^T) = \Delta u + \nabla(\nabla \cdot u), \tag{2.21}$$

so eq. (2.20) can be rewritten as

$$\int_{\Gamma_s} t_{\Gamma_s}\, d\Gamma_s = \int_{\Omega_s} -\nabla p + \nu\rho\Delta u\, d\Omega_s. \tag{2.22}$$

One can now see that the force acting on the body consists of a pressure and a viscous term. The pressure and velocity information from the fluid phase is given on the discrete fluid mesh. This rises the need for a discretization of (2.22).

In the first step, the integral over the body in (2.22) is extended to an integral over the whole domain Ω:

$$\int_{\Omega_s} -\nabla p + \nu\rho\Delta u \; d\Omega_s = \int_{\Omega} (-\nabla p + \nu\rho\Delta u) \; \xi_\Omega \; d\Omega, \tag{2.23}$$

given that ξ_Ω is defined as

$$\xi_{\Omega_s} = \begin{cases} 1 & \text{if } \mathbf{x} \in \mathbf{\Omega_s}, \\ 0 & \text{else .} \end{cases}$$

In this case $\mathbf{x}$ denotes an arbitrary element from Ω. Let us now assume a decomposition T_h of Ω, consisting of cells c. Then the integral in (2.23) can be written as

$$\int_{\Omega} (-\nabla p + \nu\rho\Delta u) \; \xi_\Omega \; d\Omega = \sum_{c \in T_h} \int_{V(c)} (-\nabla p + \nu\rho\Delta u) \; \xi_\Omega \; dV(c), \tag{2.24}$$

where $V(c)$ denotes the volume of cell c. Numerical integration with the aid of a 3D-version of the rectangle rule then yields

$$f_{p,f} = \sum_{c \in \overline{T}_h} (-\nabla p + \nu\rho\Delta u)(c) \cdot V(c), \tag{2.25}$$

where $\overline{T}_h$ denotes the set of all solid-covered cells. For cases with more than one body, the fluid force on each particle is calculated separately.

3

Implementation of the Resolved Method

The implementation of the resolved CFD-DEM method was carried out within the Open Source framework CFDEMcoupling [2012]. CFDEMcoupling combines finite volume based CFD solvers and LIGGGHTS calculations. The CFD solvers use the tools and data structures provided by OpenFOAM. Each solver consists of a main structure, which takes care of solving the governing equations, and has access to a number of models that carry out specific tasks. The following sections give an overview over the algorithm as well as details concerning the models that were developed or modified for the given application.

3.1 cfdemSolverIB[1] *- The algorithm*

CFDEMcoupling is a C++ code that uses OpenFOAM functionalities and provides a coupling interface to the DEM code LIGGGHTS, which is written in C++ as well. The solver `cfdemSolverIB` itself contains commands that

(1) create a data structure for the particle handling,

(2) import,

(3) handle and

(4) export particle data,

(5) solve the fluid equation and carry out a

(6) velocity correction.

[1] As described in the previous section the solver does not directly apply an immersed boundary method, but rather a fictitious domain method. Its name was chosen rather early in the development phase, and kept for historical reasons.

Tasks (1)-(4) are carried out by submodels. Some of these models belong to the general framework of CFDEMcoupling, others were especially developed for the presented application.

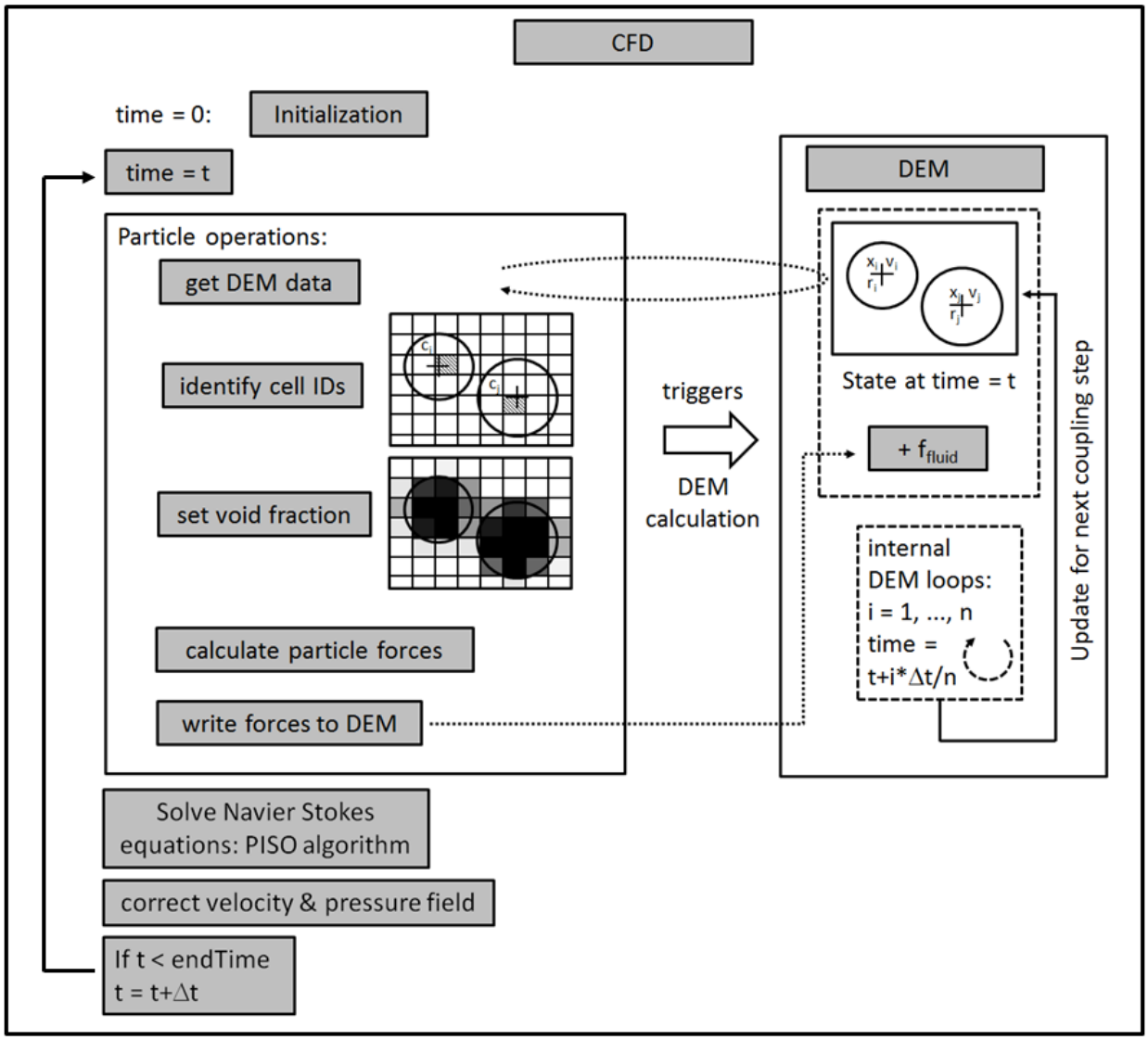


Fig. 3.1: Procedure of the CFDEMsolver, one iteration step.

Fig. 3.1 shows the main structure of the solver. The CFD calculation is the master process, which is launched by the user.

Before actual calculations take place, the data structure for handling the DEM data is created. Then time iteration starts. In each time step the following routine is carried out:

- First of all, all particle-related operations are taken care of:
  - A function retrieves position, velocity and radius of each particle in the domain from the DEM calculation at the current state.

  - A "location model" is used to assign the particles to the cells that contain their centres.
  - Another model, termed "void fraction[2] model", is used for determining the void fraction in each cell.
  - With the aid of the pressure and velocity information in the regions marked by the void fraction, the drag force on each particle is computed as given in Sec. 2.4.
  - The resulting forces for each particle are returned to the DEM solver.
  - A "data exchange model" triggers the run of a DEM script. The outcomes from these calculations are used as input in the next iteration.

- While the DEM solver updates the particle positions, the CFD-solver proceeds with the PISO-loop, which solves the Navier-Stokes equations on the whole computational domain.
- The newly gained velocity and pressure field are then subject to the correction step: first, the particles' velocities are incorporated in the marked regions, then the continuity is restored. Furthermore, the before mentioned pressure correction is carried out.
- As customary for solvers using the OpenFOAM structure, last task in each time step consists in the adjustment according to the boundary conditions.

In most cases, DEM calculations require smaller time steps than CFD computations do. The code accounts for that fact by allowing for an independent choice of the time step width in CFD and DEM. One only has to make sure that time steps and coupling steps match.

`cfdemSolverIB` requires a number of specific models, which are discussed in the following sections.

3.2 Grid independence and dynamic local mesh refinement

When a particle moves through a meshed domain, its representation by cells changes with the its position. If the mesh is rather coarse in comparison to the particle diameter, the representation of the object might change remarkably from one time step to another. Fig. 3.2 shows an example for such an unfavourable case.

From Fig. 3.2a to Fig. 3.2b the particle's volume in the mesh changes by 33 %. As this volume is used for the force calculation, the outlined inaccuracy leads to wrong results. The independence of the particle's position with respect to the mesh cells as well as grid independence in general are crucial for obtaining high-quality results. A series of investigations showed that a minimum of 8 cells per particle diameter is necessary for guaranteeing mesh independent results. In the investigation, a sphere was released in a rectangular domain. As being exerted to gravity, it started settling. Fig. 3.3 illustrates the heights of the sphere centres for cases with identical settings, but meshes of different grid size.

[2] The void fraction referred to in this context is only used as marker for defining the solid region within the mesh. It has no impact on the solution of the Navier-Stokes equations.

(a) position of the particle at time t *(b)* position of the particle at time $t + \Delta t$

Fig. 3.2: Representation of a body moving through a fixed grid.

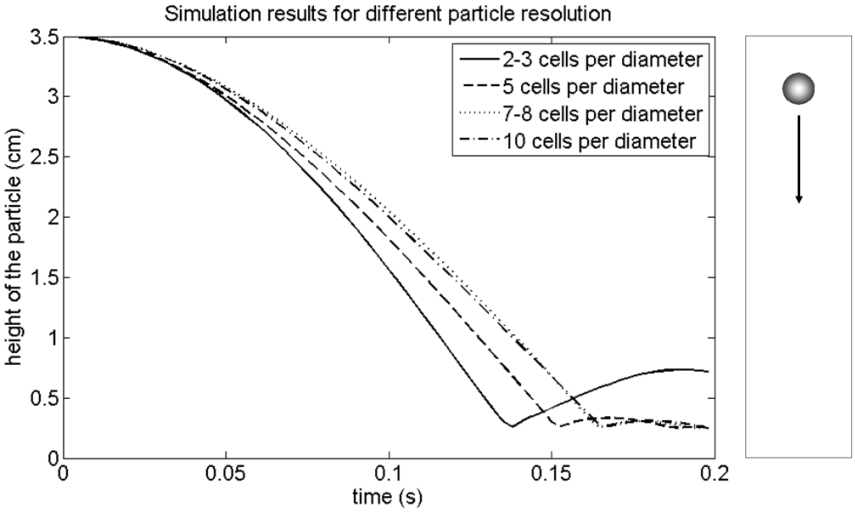


Fig. 3.3: Particle centre height over time for different cell-size/particle-diameter ratios.

The results for the two coarser meshes (2-3 and 5 cells per particle diameter) differ significantly from the results with the finer meshes. The lines representing the finer meshes show only minor deviations from each other.

Especially in dilute regimes the claim for rather small cells in the particle area often leads to an unnecessarily fine mesh in the pure fluid regions, and consequently to high computational costs. A remedy to this issue is provided by dynamic local mesh refinement. In this approach the basic mesh is chosen according to the requirements of the CFD calculation. Regions, which are covered by solids an thus need more cells, can be refined during the calculations. As soon as the particles leave the area, a re-coarsening operation can be carried out. OpenFOAM provides a refinement-plug-in which was included in the solver. It allows choosing the "degree of refinement" and then divides the affected cells into a number of smaller ones. For avoiding large aspect ratios between neighbouring cells, the refinement is carried out in layers.

Fig. 3.4 illustrates the idea of the process: a marker field indicates the cells that need refinement. After being identified they are sub-divided layer by layer.

The realization within the calculation is shown in Fig. 3.5. The resolution of the area in close proximity of the two spheres is higher than in the remaining domain.

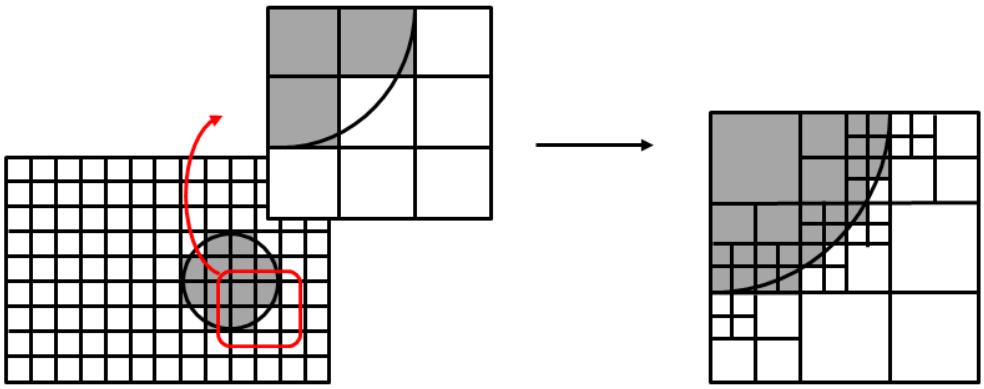


Fig. 3.4: Dynamic local mesh refinement in the area of the particles.

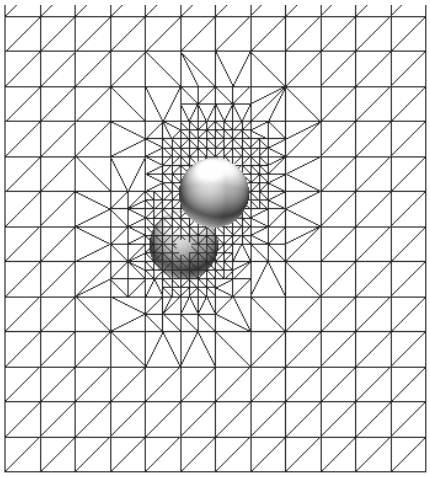


Fig. 3.5: Realization of dynamic local mesh refinement within OpenFOAM.

3.3 Void fraction models

`cfdemSolverIB` reverts to the void fraction for calculating the particle forces and correcting the fluid velocity field. The question of how to choose and treat this field, that maps the Lagrangian particles to the Eulerian mesh, was therefore a main issue. In the following two possibilities for setting the field are introduced.

One way to represent a particle within a non-conforming mesh is to denote all cells as "solid", whose centre is covered by the body, and the remaining ones "fluid". This yields a void fraction field that only contains the values zero and one. This method is rather fast, but at the same time rather inaccurate, as the particle's volume will usually be misrepresented, depending on the discretization of the domain. For cases involving fine meshes the error might be negligible, but as soon as densely packed beds are considered, stability problems occur. As illustrated in Fig. 3.6, physically existing gaps between particles might vanish in the calculation due to the misrepresentation of the particles. This leads situations, where certain fluid-containing areas are cut off from the remaining void space.

A new model was designed to overcome this weakness: in comparison to the stair-step method it accounts for the degree, to which a cell is covered by the particle at the present time

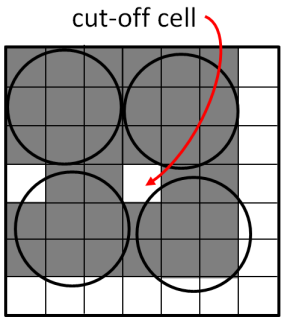
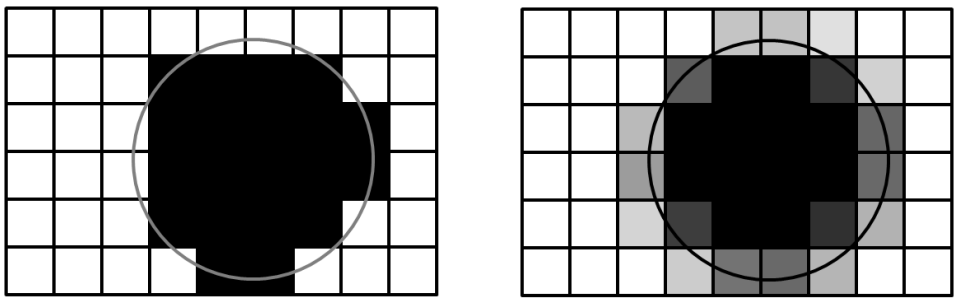


Fig. 3.6: Closing of gaps due to mis-representation of particles.

step. This model is referred to as "smooth representation". In Fig. 3.7 the two approaches are compared to each other.



(a) stairstep representation *(b)* smooth representation

Fig. 3.7: Representation of a particle within the mesh - different void fraction models.

The cells are coloured according to the void fraction. While in Fig. 3.7a only solid - black - and fluid - white - are distinguished, there are grey scales in Fig. 3.7b, accounting for the degree to which the cell is filled with fluid or solid. For determining the solid and fluid fraction of a cell the following algorithm was implemented:

3.3.1 Smooth particle representation algorithm

The task was to find an efficient way for calculating the solid volume within a cell that contains both phases. A volume integral would be the solution to the problems, but calculating it analytically for each affected cell would have been by far too complicated. Instead, a suitable numerical method had to be found.

The presented algorithm was developed for 3D meshes with hexahedral cells. Lifting this limitation and generalizing it to a form that can be applied to arbitrary meshes is possible, but as the used grids were all structured this step was not taken.

The algorithm starts with dividing the spherical particle into a core and a corona. (cf. Fig. 3.8).

Afterwards the following procedure is carried out for each particle:

All cells that are detected to be in the area of the particle are checked for their relative position with respect to the particle. If they are located inside the core, their volume fraction is set to one straight ahead. If they are covered by the corona, a loop over all

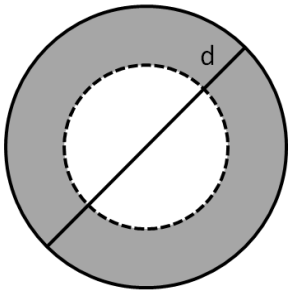


Fig. 3.8: The particle is divided into a core (white) and a corona (grey).

vertices belonging to the cell starts. If the vertex is inside the particle, the volume fraction of the cell is increased by one eighth (as each cell possesses eight vertices in the given case). If the vertex lies outside the corona, the intersecting point of particle hull and the connection between cell centre and edge is computed. The relative length of the line between cell centre and intersection times one eighth is then added from the present volume fraction of the cell.

Fig. 3.9 shows an example for determining the void fraction of a 2D cell: in (1) a vertex outside the particle is found. In (2) a connection between cell centre and edge is established, which is used for the calculation of the intersection and also the distance between intersection and cell centre. The procedure is repeated for the left upper vertex, the two others - lying inside the particle - contribute to the volume fraction with 0.25 each. The final contribution is marked with a red line.

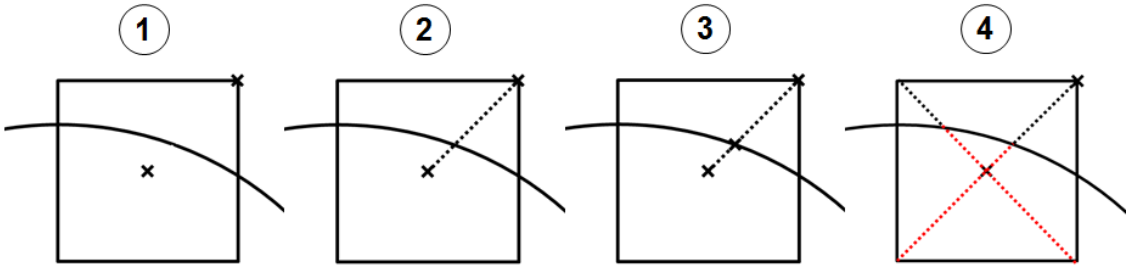


Fig. 3.9: Determination of the volume fraction: (1) vertex lies outside the particle (2) + (3) intersection point is found (4) summing up the contribution from each vertex.

Before applying the model, its accuracy had to be tested. For this purpose it was implemented in matlab and applied to variation of body-cell configurations. The outcome was compared to results calculated with a Monte Carlo Method. Fig. 3.10 shows how the different cell-body configurations were generated.

The cell centre was moved towards the surface of the particle, following the broken line, thus cases for all percentages of coverage were considered. Between the calculations the angle α was varied. In Fig. 3.11 the relative errors between Monte Carlo method and new algorithm are illustrated for three angles. The images below the x-axis indicate the relative position of the cell with respect to the particle surface.

In the worst case the offset amounts to 10%. Given the simplicity of the model and the low computational costs this result is satisfying.

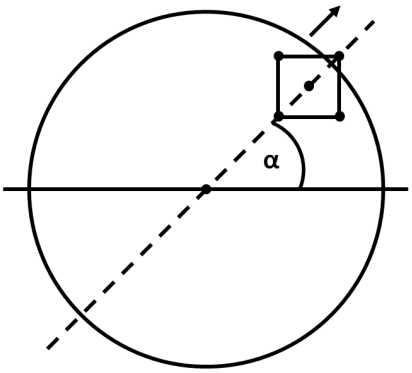


Fig. 3.10: Comparative cases for the validation of the void fraction model.

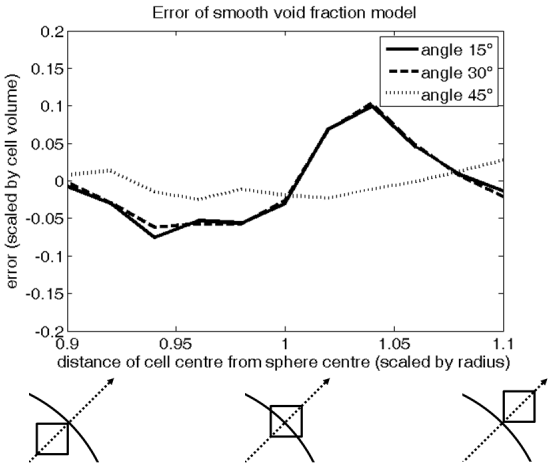


Fig. 3.11: Deviation from the Monte Carlo results for $\alpha = 15\,\text{deg}, 30\,\text{deg}$ and $45\,\text{deg}$.

3.3.2 Usage of the smooth void fraction field

The void fraction field is then finally used in the step where the velocity correction takes place. In the cells that contain both phases, the velocity is evaluated as combination of fluid and particle velocity. With this remedy the beforehand mentioned numerical instabilities can be avoided and the representation of the particles is more accurate.

3.4 Parallelization of the method

As discussed in the previous sections, rather fine meshes are required for representing the particles accurately. Thus also the computational costs grow fast when the number of particles considered is increased. Both OpenFOAM and LIGGGHTS can be operated in parallel using the Message Passing Interface (MPI). Running the presented solver in parallel is, however, not possible right out of the box. The reason therefore lies in the particle location method: the particles are detected via their centre, and if a body covers cells on two processors at once, only the cells sharing the processor with the centre-cell are found.

In a first step an inter-processor communication routine was introduced during the particle location process. The particles could be depicted correctly, but at the same the communication slowed down the calculation, reducing the speed up effect remarkably. After further investigations, the following idea emerged:

In the non-parallel version the particle location starts at the centre cell. However, the algorithm is written in such a way, that any other cell, covered by the particle, can be used as starting point as well. Due to the nature of the parallelization applied, each processor sees all particles at any time, also those that are located on other processors. Therefore, in the parallel version of the void fraction model, the algorithm first checks, whether the particle centre can be found on the processor. If this is not the case, an additional number of "satellite points", which are located on the surface of each particle, is checked. In Fig. 3.12 one can see that the particle location process on the processor with the centre starts at the centre-cell, just as usual. The particle also covers cells on the neighbouring processor, which are located starting from the rightmost satellite point.

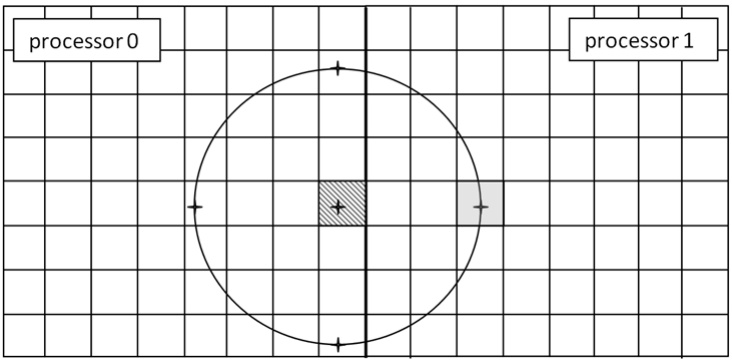


Fig. 3.12: Particle location process in the parallel IB - grey shaded cells mark the starting positions of the locate algorithm on the two processors respectively.

The number of satellite points can be chosen arbitrarily. No unnecessary inter-processor communication has to take place during a calculation step, which contributes to improvements of the code-performance. A small benchmark study showed the scaling behaviour for a case with about 110,000 cells (cf. Tab. 3.1).

number of processors	computational time
2	1
4	0.59
8	0.37
12	0.24

Tab. 3.1: Performance behaviour of a problem with approx. 110,000 cells.

The resolved CFD-DEM method is designed for problems involving relatively small numbers of particles ($O(10^3)$), while classical CFD-DEM methods often handle some millions of particles. The limiting factor of a resolved CFD-DEM problem is the cell number of the CFD mesh.

In addition to the solver and the presented code details two force models were generated. They accounted for the drag force that the fluid exerts on the particle, and the buoyancy of the particles. After the implementation was completed, all features were tested thoroughly; the results of the tests are documented in the following chapter.

4

Validation Examples and Application

For validating the solver and demonstrating its applicability, some basic examples were chosen. Both 2D- and 3D-cases were considered. The simulation results were compared with the literature or analytics.

4.1 Flow around a cylinder - 2D Example

This first example was used to show the correctness of the resolved method. For this purpose the drag force onto a cylinder moving through a resting fluid was compared against the drag force onto a fixed cylinder in a steady fluid flow. For comparing the results under different conditions, the drag coefficient was considered. The correlation between drag force f_{drag} and drag coefficient C_d is given by

$$f_{drag} = \frac{\rho_F u^2}{2} A C_d, \tag{4.1}$$

where ρ_F is the fluid density, u the relative velocity between cylinder and fluid and A the cross section area of the cylinder. In the present example the cylinder was either fixed or moving at a prescribed velocity, wherefore no DEM calculation was necessary. The drag information was calculated and written to a file for post processing. No interaction with LIGGGHTS took place in this case. A series of calculations with varying Reynolds numbers was carried out. A comparison of the results to values from the literature con be found in Tab. 4.1.

The obtained results were in good accordance with the literature, yet the values for fixed and moving cylinder were not identical, as indicated by Tab. 4.2. One explanation for this behaviour lies in the rather narrow computational domain. The wall-effect was pronounced strong enough influencing on the results. Another reason for the deviation lies in the fact that this investigation was carried out at a rather early state of development. At this time, the smooth void fraction model was not available. So while in the case of the fixed cylinder

Re	Method	C_d
100	fixed cylinder, present calculation	1.438
	fixed cylinder, Uhlmann [2005] (enlarged domain)	1.453
	fixed cylinder, Lai and Peskin [2000]	1.447
	moving cylinder, present calculation	1.500
	moving cylinder, Uhlmann [2005] (narrow domain)	1.559
150	fixed cylinder, present calculation	1.3304
	fixed cylinder, Liu et al. [1998]	1.3340

Tab. 4.1: Comparison of drag coefficients at different Reynolds numbers.

its representation remained the same over the whole computational time, this case not the case for the moving cylinder case.

Re	C_d - fixed cylinder	C_d - moving cylinder
0.1	321.9	266.25
1.0	19.3	17.8
4.0	5.0	5.5

Tab. 4.2: Comparison of the drag coefficient C_d for the fixed and the moving cylinder.

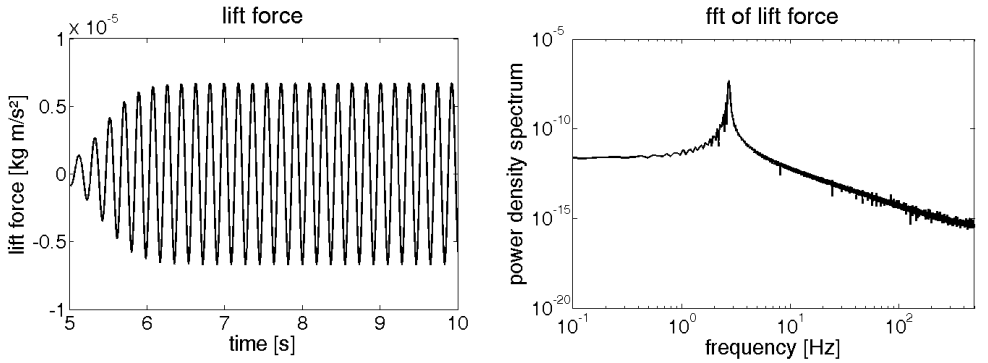
In fluid mechanics, there are several non-dimensional quantities that allow for the characterization of flow situations. One of them is the Reynolds number, given as

$$Re = \frac{u \cdot L}{\mu}, \tag{4.2}$$

where u is the flow velocity, L a characteristic length and μ the dynamic viscosity. While for low Reynolds numbers, the flow is of a laminar nature, turbulences occur in high-Reynolds-regimes. For the considered set-up vortices start shedding behind the cylinder in a certain range between laminarity and turbulence. The emerging chain of vortices is termed Kármán vortex street. In the given example its development was observed at a Reynolds number of 150 (cf. Fig. 4.1).

Fig. 4.1: Vortex street developing in the wake of a cylinder, Re = 150.

Due to the vortex shedding an oscillating lift force acted onto the cylinder. Fig. 4.2 shows the lift force and its fast Fourier transform (fft). The peak visible in the fft-diagram indicates the frequency of the vortex shedding. A quantity to measure this oscillating flow



(a) lift force on the cylinder

(b) fast Fourier transform of the lift force

Fig. 4.2: Vortex shedding causes an oscillating lift force onto a cylinder at $Re = 150$.

mechanism is the Strouhal number St, defined as

$$St = \frac{fd}{u}. \tag{4.3}$$

Here f is the frequency of the vortex shedding and d the diameter of the cylinder. For a moving cylinder with Reynolds number $Re = 100$ the Strouhal number was determined as 0.205, compared to Uhlmann's result of 0.17.

4.2 Flow past a stationary sphere

In this first 3D example the flow around a sphere was considered. Again the Reynolds number was used to classify the developing flow regime. In the present study, the Reynolds numbers ranged from 100 to 1000, the different phenomena observed are presented in the following:

- For $Re = 100$, the flow behaviour in the wake of the sphere was symmetric. The fluid evaded the obstacle and only a few sphere diameters downstream it continued almost undisturbed. Downstream the sphere a small backflow area developed. Fig. 4.3 and Fig. 4.4 contain the flow field and stream lines for this and three further regimes.

- At a Reynolds number of 250, the backflow area was remarkably larger, yet the overall flow behaviour was still symmetric.

- This changed at Re = 300, where the Kármán vortex street started to develop. This observation was reconfirmed by the findings of Sakamoto and Haniu [1990]. Fig. 4.5 shows the drag and lift coefficient for this case.

  After a certain stabilization phase the drag oscillated around 0.638. This result is in good accordance with the results by Kim and Choi [2002] (cf. Tab. 4.3). Furthermore, Tab. 4.3 contains comparative values for the lift coefficient C_l and the Strouhal number St. The lift coefficient was calculated analogously to the drag-coefficient, where the drag force is replaced by the lift force f_{lift}:

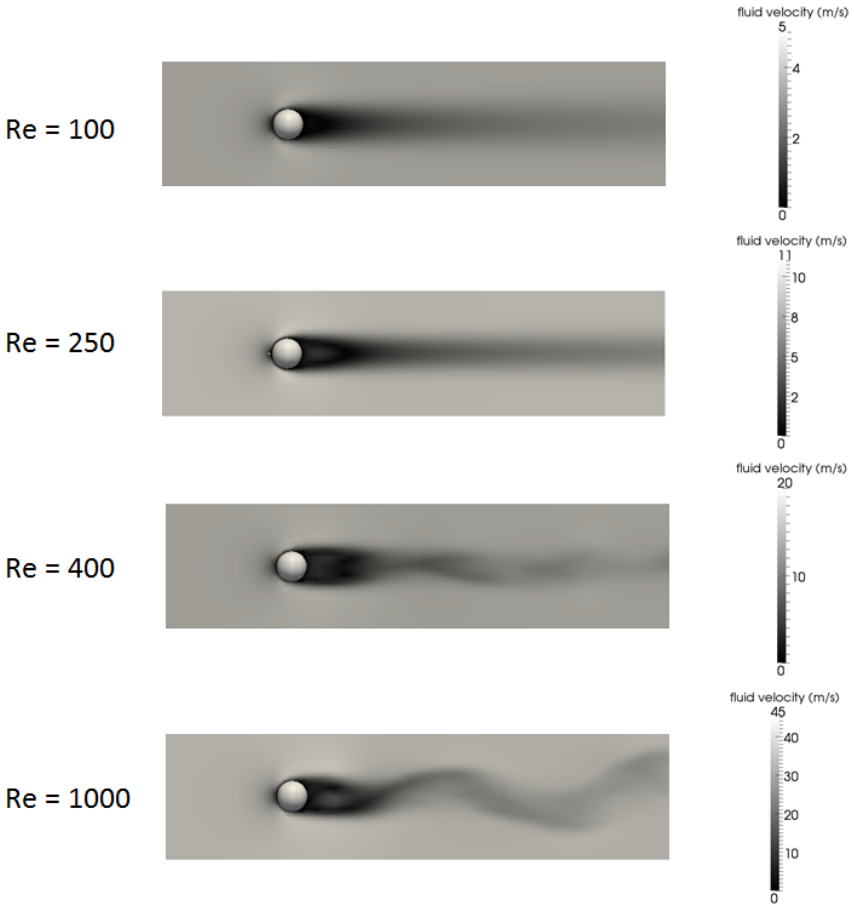


Fig. 4.3: Flow fields of flow around a sphere at different Reynolds numbers.

$$C_l = \frac{f_{lift}}{\frac{1}{2}\rho_F u^2 A}. \tag{4.4}$$

	recent results	Kim and Choi (2002)
C_d	0.638	0.657
C_l	0.065	0.067
St	0.117	0.134

Tab. 4.3: Comparison of drag coefficient C_d, lift coefficient C_l and Strouhal number St for a sphere at $Re = 300$.

- For Reynolds numbers larger than 300, the vortex shedding frequencies became irregular. The flow in the wake of the sphere showed large turbulent regions. In Fig. 4.3 the flow fields for $Re = 400$ and $Re = 1000$ are displayed the according streamlines can be found in Fig. 4.4. For the latter case, drag and lift coefficient are illustrated by Fig. 4.6.

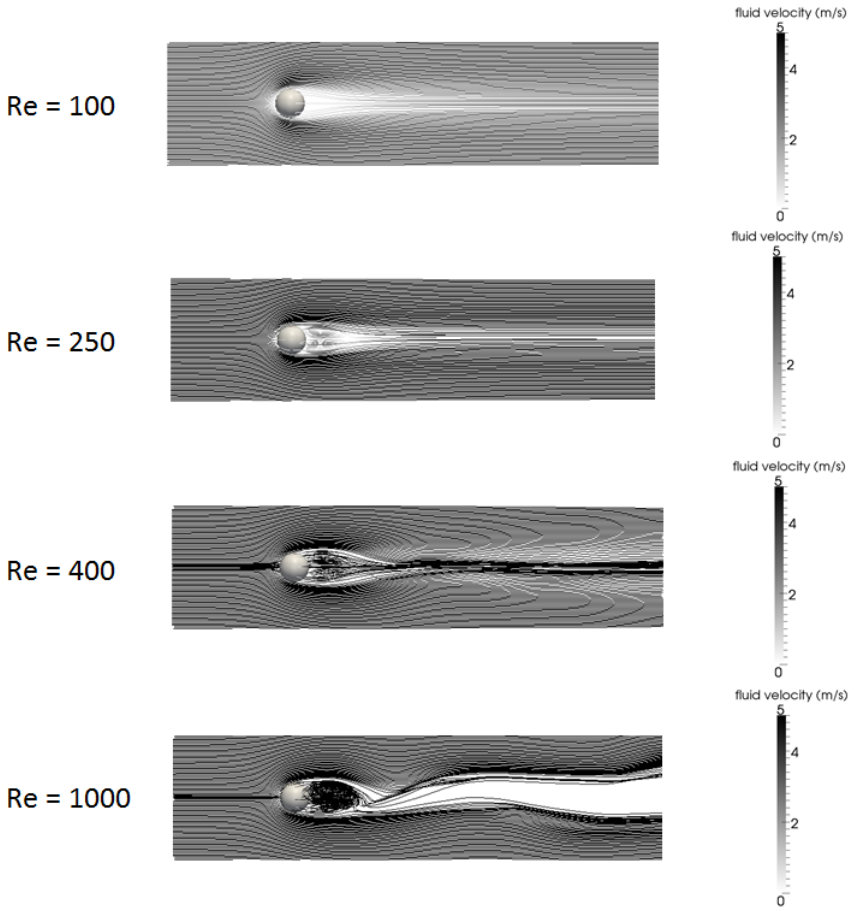


Fig. 4.4: Stream lines of flow around a sphere at different Reynolds numbers.

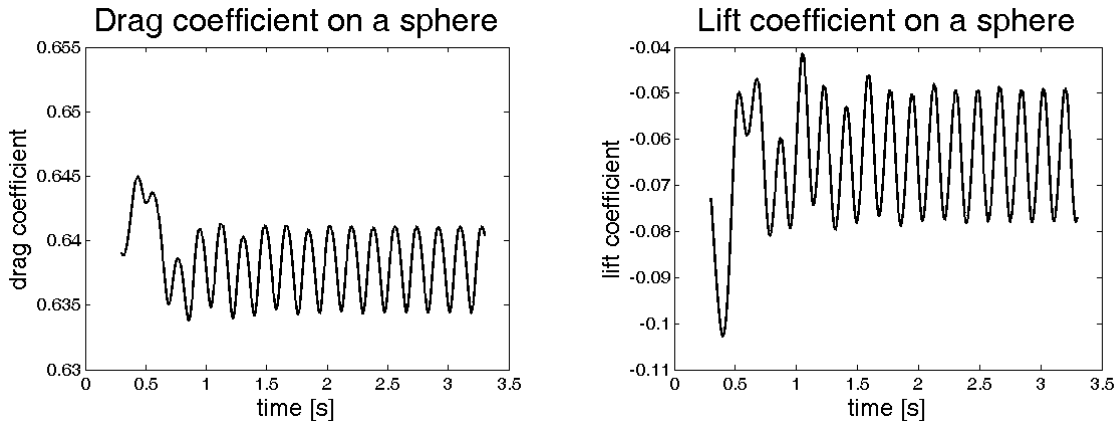


Fig. 4.5: Drag and lift coefficient over time on a sphere for Re = 300.

In Tab. 4.4 the drag coefficients of all cases regarded are collected. The drag coefficients are in good accordance with the heuristic results presented in the literature (cf., e.g., Brauer [1971]), as illustrated by Fig. 4.7.

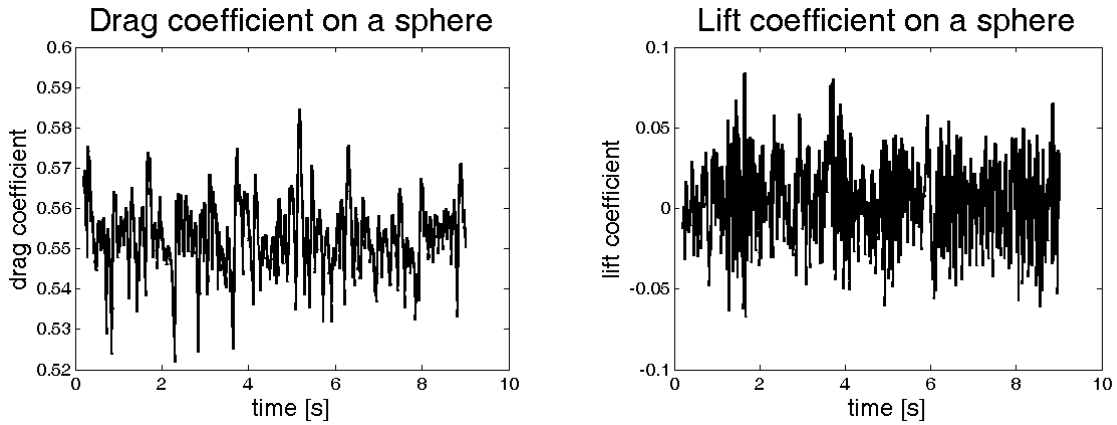


Fig. 4.6: Drag and lift coefficient over time on a sphere for Re = 1000.

Reynolds number	Drag Coefficient C_d
100	1.01
200	0.767
250	0.679
300	0.638
400	0.609
1000	0.534

Tab. 4.4: Drag coefficients C_d on a sphere at different Reynolds numbers.

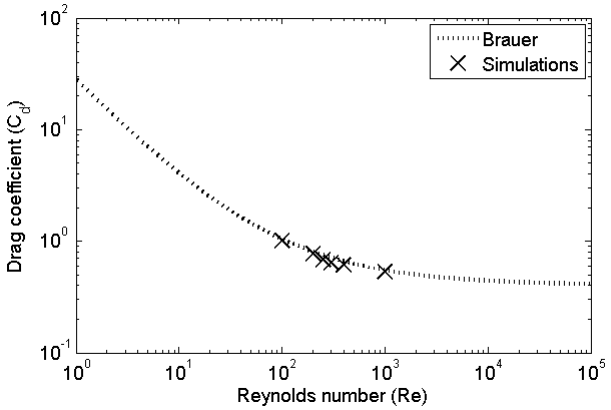


Fig. 4.7: Drag coefficient for a sphere for different Reynolds numbers, comparison of simulation results and the literature (cf., Brauer [1971], pp. 199/200).

When a fluid, faster than creeping flow, passes by an obstacle, one can observe its detachment from the objects surface. In all cases this critical velocity was exceeded, and it could be observed that the break-off points moved upstream for increasing Reynolds numbers (cf., Fig. 4.8).

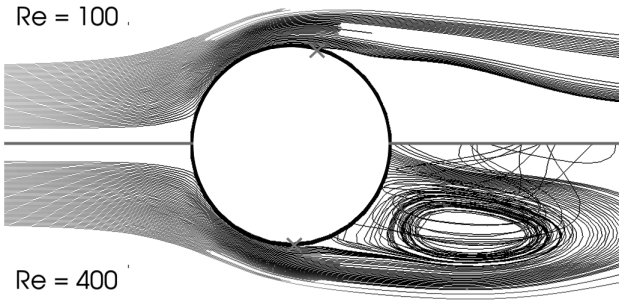


Fig. 4.8: Break-off points (x) for Re = 100 and Re = 400.

4.3 Flow past a rotating sphere

So far, only fixed objects or objects displaying a constant translational motion were considered. However, asides from translation, a sphere can also experience rotation, as its motion is given by

$$u = U + \omega \times r. \tag{4.5}$$

Here U is the translational velocity, r is a position vector with respect to the centre of the sphere and ω denotes the angular velocity.

In the current example the focus lay on the impact of a rotating sphere on the surrounding flow. In order to account for the rotation, the velocity correction operation was equipped with an additional term.

The rotation of a sphere in a flow causes an increase of the velocity in the area, where the sphere surface moves in flow direction, and a reduction on the opposite side. According to Bernoulli this causes a pressure drop. The different pressure levels result in a lift force, termed the Magnus force.

Fig. 4.9 shows the stream lines of a sphere the rotating with $\omega* = 2$ at a Reynolds number of 100. $\omega*$ is the dimensionless angular velocity given by $r\omega/u$, with r being the radius of the sphere, ω its angular velocity and u the fluid velocity.

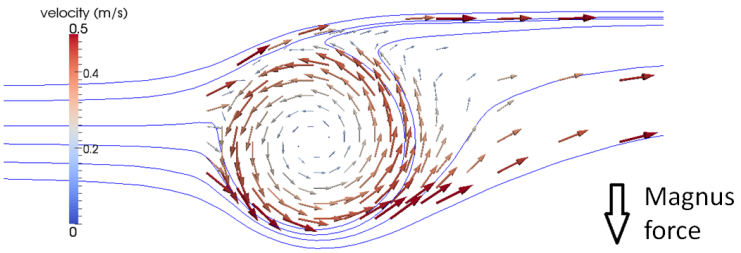


Fig. 4.9: Streamlines around a rotating sphere, the Magnus force acts in the direction of the arrow.

For Reynolds numbers between 10 and 140, Oesterlé and Dinh [1998] derived a relation between Reynolds number and lift coefficient. It reads as follows:

$$C_L \cong 0.45 + (2\omega^* - 0.45)\exp(-0.075{\omega^*}^{0.4} Re^{0.7}). \tag{4.6}$$

This relation was used to validate the simulation results for spheres at different rotating velocities. In Fig. 4.10 the results of Oesterlé are compared to the results of the recent calculations. Especially for higher angular velocities the results are in good accordance.

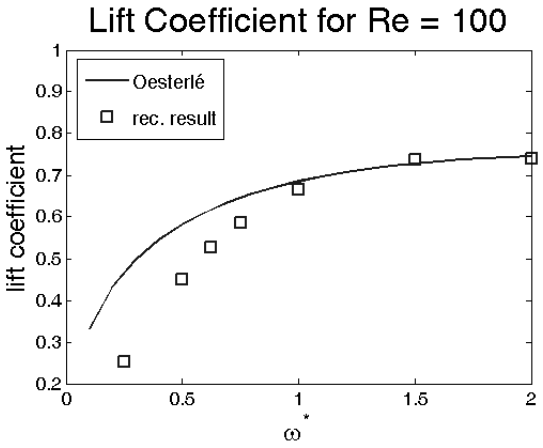


Fig. 4.10: Comparison of the lift coefficients.

In most cases the translational velocity components dominate the flow behaviour, nevertheless, the consideration of the rotation allows for more accurate computations.

4.4 Settling of disks (2D) and spheres (3D)

After a number of cases with prescribed particle motion, the solver was finally applied to cases, where the particle's path was calculated with LIGGGHTS. In the presented cases either one or two disks - in 2D - or spheres - in 3D - were released in a rectangular 2D or 3D domain. The contained fluid was at rest. Both gravity and buoyancy acted onto them, causing them to settle. Tab. 4.5 gives an overview over the calculated cases.

	2D		3D	
	1 disk	2 disks	1 sphere	2 spheres
∅ disk/sphere (cm)	0.25	0.25	0.25	0.167
ρ disk/sphere	1.25 1.50	1.5	1.14 1.50	1.14 1.50
geom. of the box ($cm^{2,3}$)	$[0,2] \times [0,6]$	$[0,2] \times [0,6]$	$[0,2] \times [0,6] \times [0,0.5]$	$[0,2] \times [0,6] \times [0,0.5]$
centre disk/sphere	(1,4)	(1,4.5) (1.0,5.0)	(1.0,5.0, 0.25)	(0.5, 0.5, 3.5) (0.5, 0.5, 3.16)
fluid viscosity ν (cm^2/s)	0.01 0.1	0.01 0.1	0.01	0.01
gravity (cm/s^2)	(0 -981)	(0 -981)	(0 -981 0)	(0 -981 0)

Tab. 4.5: 2D and 3D testcases for the settling of one or two disk(s)/sphere(s).

Figures 4.11, 4.12, 4.13 and 4.14 show the comparison of centre positions and settling velocities obtained by calculations with the immersed boundary solver to results by Glowinski et al. [1998, 1999, 2001].

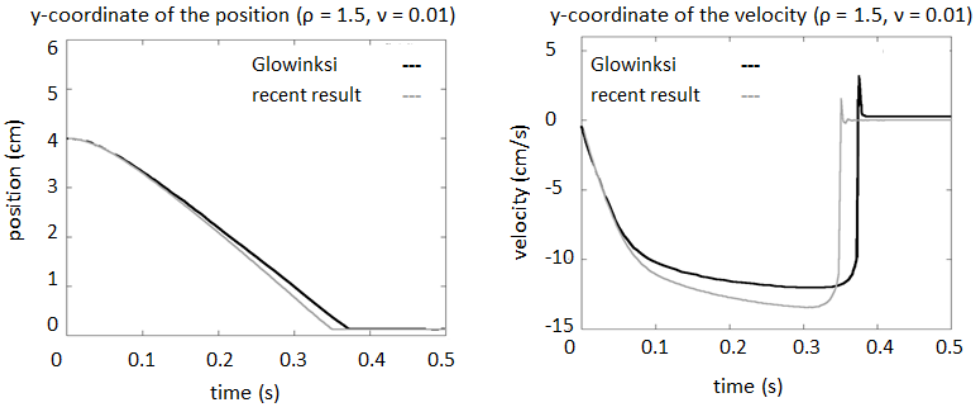


Fig. 4.11: Position (left) and velocity (right) of a settling particle of density 1.25 in a fluid with viscosity 0.1.

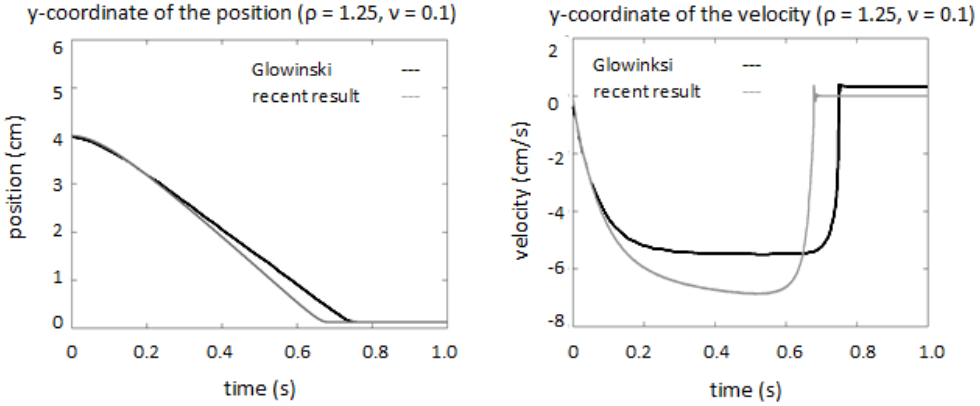


Fig. 4.12: Position (left) and velocity (right) of a settling particle of density 1.5 in a fluid with viscosity 0.01.

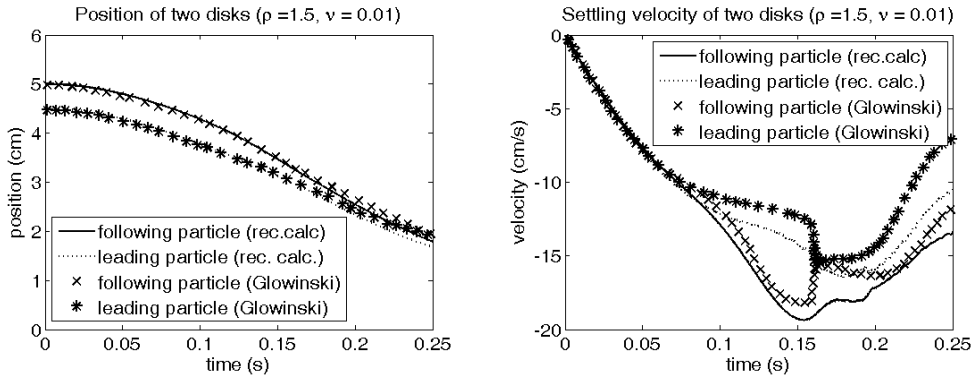


Fig. 4.13: Comparison of centre positions and settling velocities of two disks - computed results vs. Glowinski et al. [2001].

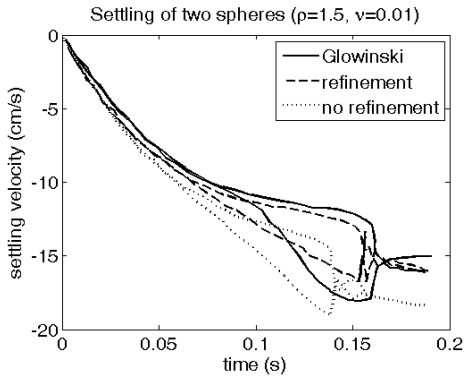
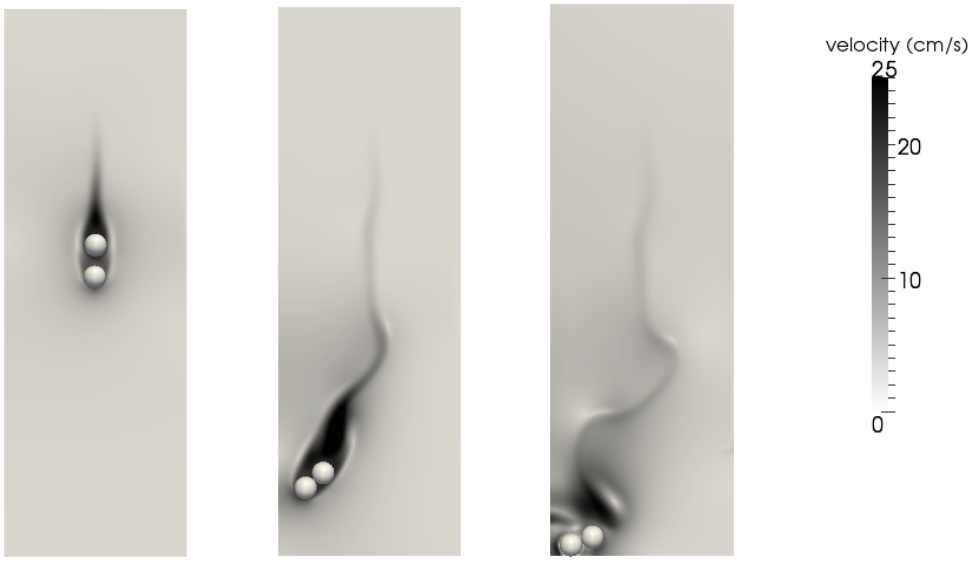


Fig. 4.14: Velocity of two settling spheres of density 1.5 in a fluid with viscosity 0.01. Computed results (with and without dynamic local mesh refinement) vs. results by Glowinski et al. [2001].

4.4.1 Drafting, kissing and tumbling

When two bodies are released above each other one can observe the following three states: first the two particles start descending independently. As soon as the upper particle reaches the wake of the lower one, it gets accelerated and "follows" the leading particle. This phenomenon is called drafting (cf., Fig. 4.15a). In Fig. 4.15b one sees the collision of two particles; until this point the behaviour of the particles can still be predicted. Finally, in Fig. 4.15c the particles reach the state of tumbling; in this state a strict prediction is not possible. These effects occur both for 2D and 3D simulations.



(a) drafting *(b)* kissing *(c)* tumbling

Fig. 4.15: Three states during the settling of two disks (2D).

4.5 Eight settling spheres

In this case the number of particles in the domain was extended from two to eight. Special attention was given to the effect; different initial configurations have on the general settling behaviour. The particles were inserted as two layers of four objects. In one case the objects were arranged in such a way that the centres of the spheres formed the vertices of a cube, in the second configuration they formed a parallelepiped (cf. Fig. 4.16). The two cases will further on be referred to as "cubic" and "askew".

The computational domain had the dimensions 0.4 cm × 0.4 cm × 3.2 cm. With 0.05 cm the vertical distance of the two layers was equal to the minimal distance of the spheres from each other. The fluid's viscosity was given as 0.4765 cm^2/s, the density of the particles was 2.5 g/cm^3. The eight spheres had a diameter of 0.1 cm each.

At $t = 0$ the spheres were placed in a resting fluid bed and released. Fig. 4.16 shows the initial configuration, the positions during the settling process at different time steps and in (1.3) and (2.3) the positions after 1.8 seconds.

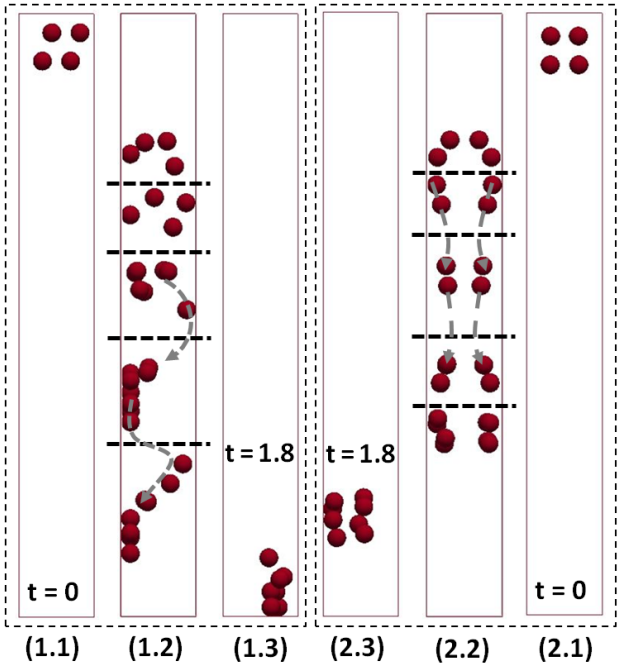


Fig. 4.16: Settling of eight spheres, (1.1) and (2.1) initial position, (1.2) and (2.2) after 0.3, 0.6, 0.9, 1.2 and 1.5 seconds and (1.3) and (3.3) final configuration at t = 1.8 seconds.

The settling behaviour differed strongly in the two set-ups: In the "cubic" case the particles divided into a left and a right group, and started settling four by four. The paths of the respective groups were rather symmetric (cf. 4.16, (2.3)). In the "askew" case the initial setting prevented any kind of symmetric behaviour. As illustrated by Fig. 4.16 (1.2) the particles soon formed a single group. A "leading" particle started its descent first, taking the remaining objects along in its wake. This caused an overall acceleration. The comparison of (1.3) and (2.3) at $t = 1.8$ s shows that while the settling of the particles in the "askew" case was completed, the process was still on-going for the "cubic" case.

This example pointed out that the initial configuration of a group of objects has a strong impact on its settling behaviour.

4.6 Settling of particles within a suspension - The Boycott effect

In this last test case for the resolved CFD-DEM solver the dynamics of a larger amount of particles was studied. The idea for setting up this case as described stemmed from the "Boycott-effect": Investigating the settling behaviour of blood cells, Boycott discovered in the 1920s that the sedimentation of the cells can be accelerated by tilting the test tube (cf., Boycott [1920]). This effect is evoked by the fact that the settling particles first approach the sloping wall and leave a region with pure liquid above them. The region with high particle concentration has a significantly higher density than the remaining fluid and therefore the particles settle faster (cf., Xu and Yu [1998]).

Figure 4.17 gives a schematic representation of the settling of particles in a tilted test tube. This regime develops after a certain time; in the upper region the suspension is already cleared of particles, while in the lower region one can distinguish between a dilute and a dense suspension. In the given figure the (purely imaginary) boarder between the two domains is marked by a thin line.

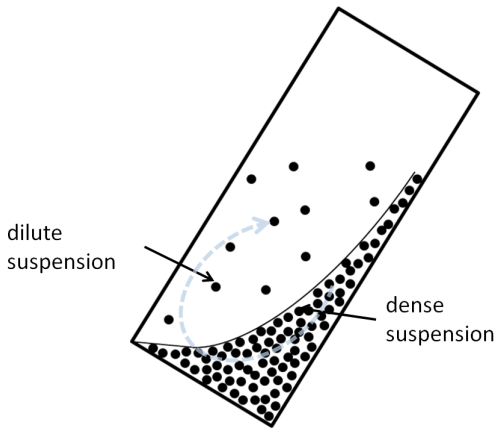


Fig. 4.17: Settling of particles in a tilted tube, dense suspension, dilute suspension and pure fluid.

In order to reproduce this effect, the following test case was defined: A cuboid with square base of length 1 cm and with a height of 5 cm was chosen. A dense packing of 50 particles with a diameter of 0.167 cm was generated in the upper part of the domain and released at $t = 0$. Then two calculations were launched, one where the gravity vector pointed straight in the direction of the negative z-axis (i.e. $g = (0\ 0\ -981)$), and one, where the gravity vector was shifted by an angle of 45°(i.e., $g = 981 * (-0.707\ 0\ -0.707)$). Fig. 4.18 shows a screen shot of the velocity fields of the two cases at the same time step: On the left hand side one can see the glyphs for the vertical case. Here, the dominating velocities directly occur in the proximity of the particles. The picture on the right hand side stems from the tilted tube, where the predicted circulating flow can be seen.

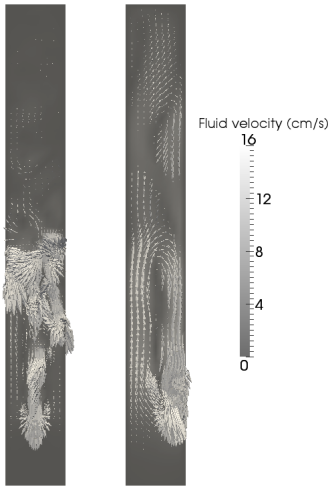


Fig. 4.18: Settling of a dense packing of spheres - gravity vertical (left) vs. gravity vector shifted by 45° (right).

Fig. 4.19 shows the particles before the settling takes place and during the sedimentation process at two different time steps. While the two central figures already show the tendency of the faster settling in the tilted case, the screen shots for $t = 1.1$ make it apparent: while the sedimentation is still on-going for the vertical tube, it is already completed in the shifted tube.

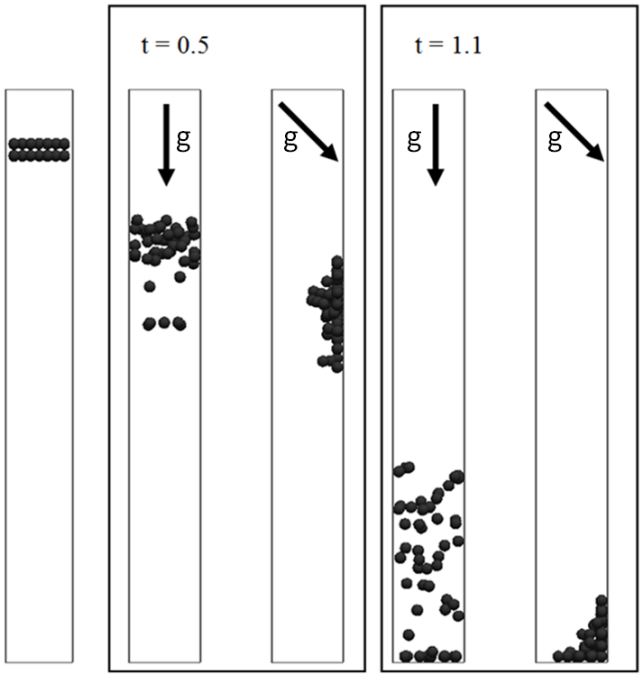


Fig. 4.19: Initial position of the spheres, settling of the packing at two different time steps. Arrow points in the direction of the gravity.

Part II

UNRESOLVED CFD-DEM

5 Unresolved CFD-DEM

The governing equations of unresolved CFD-DEM are a combination of the continuum approach for the fluid phase and a discrete approach for the solids. The literature provides a number of different formulations, some of which were collected by Zhou et al. [2010]. The presented introduction to the method and its governing equations follows this publication.

5.1 Governing equations

Particle-fluid flows can be treated in many ways. One popular method is the two fluid model (TFM). In the TFM, both fluid and solid phase are treated with the continuum approach, using the locally averaged Navier-Stokes equations. The equations of motion of the fluid and the particulate phase are formulated as follows (cf., Anderson and Jackson [1967]):

$$\frac{\partial(\alpha_f \rho_f u)}{\partial t} + \nabla \cdot (\alpha_f \rho_f u u) = \nabla \cdot \sigma - n f_i + \alpha_f \rho_f g, \tag{5.1}$$

$$\frac{\partial(\alpha_s \rho_s u_p)}{\partial t} + \nabla \cdot (\alpha_s \rho_s u_p u_p) = n\Phi - \nabla \cdot S + n f_i + \rho_s \alpha_s \rho_s g. \tag{5.2}$$

u denotes the fluid velocity, u_p the velocity of the solid phase, α_f and α_s are the fluid and solid fraction, with $\alpha_f = 1 - \alpha_s$. The densities of the fluid and the solid phase are given by ρ_f and ρ_s respectively. σ is the fluid stress tensor, n is the number of particles and f_i the local mean value of the force that the fluid exerts on the particle. Φ represents the particle-particle interaction forces, and S stands for the Reynolds stresses.

The fluid phase underlies the conservation of mass, given by

$$\frac{\partial \alpha_f}{\partial t} + \nabla \cdot (\alpha_f u) = 0. \tag{5.3}$$

The TFM does not resolve the dynamics of each particle individually. Closure relations are used to describe the solid phase and account for the inter-phase force exchange. Accuracy and applicability of the method strongly depend on these relations and many regimes cannot be treated accordingly. An approach that allows for the handling of all kinds of regimes is coupled CFD-DEM. Here, the trajectories of each particle are calculated separately. The fluid phase is in analogy to the TFM described by the locally averaged Navier-Stokes equations, given by (5.1). For obtaining the final set of governing equations a number of steps are necessary:

Zhou et al. [2010] state that the fluid-particle interaction force term nf_i can be considered as the sum of two distinct factors, namely the change of the fluid stress tensor in the area of the solids and the small-scale changes due to the flow around the particles. The expression can be denoted as

$$nf_i = \alpha_s \nabla \cdot \sigma - R_{sl}, \tag{5.4}$$

where R_{sl} denotes the liquid-solid interaction force. The considered fluid is Newtonian, thus the stress tensor σ is given as

$$\sigma = -pI + \tau, \tag{5.5}$$

with τ being the shear stress (cf. Appendix, A.1, eq. (A.11)):

$$\tau = \mu(\nabla u + (\nabla u)^T) - \frac{2}{3}\mu(\nabla \cdot u)I \tag{5.6}$$

In summary the governing equations for the fluid phase are the following:

$$\frac{\partial(\alpha_f \rho_f u)}{\partial t} + \nabla \cdot (\alpha_f \rho_f uu) - \nabla \cdot (\alpha_f \tau) = -\alpha_f \nabla p + R_{sl} + \alpha_f \rho_f g. \tag{5.7}$$

The force exchange term R_{sl} is given as

$$R_{sl} = K_{sl}(u - u_p). \tag{5.8}$$

In the implemented approach one first calculates the interaction forces for each particle. For obtaining the coefficients of K_{sl} the forces of all particles in each fluid cell are summed up. As the contributions of the pressure gradient and the viscous term are already included in nf_i via the stress tensor (cf. eq. (5.4)), only the drag force needs to be accounted for, i.e.

$$K_{s,l} = \frac{1}{\Delta V} \sum f_{d,i}, \tag{5.9}$$

where $f_{d,i}$ are the drag force values for all particles in the considered cell. ΔV is the volume of a computational cell.

A CFD-DEM approach, where both the fluid and the solid pressure are included in the fluid equation (5.7), is called "model A" in the literature. It was initially published by Ishii [1975] for the application to dispersed flows involving bubbles and was later on applied to particle laden flows by Enwald et al. [1996]. In contrast to that an alternative approach termed "model B" was presented by Anderson and Jackson [1967] and Jackson [1997]. In this model the pressure purely applied to the fluid phase. According to van Wachem et al.

[2001], the differences of the models do not result in differing macroscopic flow patterns. CFDEMcoupling provides both formulations.

The governing equations of the solid phase are the classical DEM governing equations (cf. Appendix B, eq. (B.3).), extended by a force term $f_{p,f}$, which accounts for the fluid-particle interaction:

$$m_p \frac{du_p}{dt} = m_p g + f_{p,f} + \sum_{N_p} f_{p,p} + \sum_{N_w} f_{p,w}. \tag{5.10}$$

$f_{p,p}$ and $f_{p,w}$ denote the forces caused by particle-particle and particle-wall contacts. The fluid-solid interaction term $f_{p,f}$ summarizes all particle-fluid interaction forces

$$f_{p,f} = f_d + f_{\nabla p} + f_{\nabla \cdot \tau} + f_{vm} + f_B + f_{Saff} + f_{Mag}, \tag{5.11}$$

namely the drag force f_d, the pressure gradient force $f_{\nabla p}$, the viscous force $f_{\nabla \cdot \tau}$, the virtual mass force f_{vm}, the Basset force f_B and lift forces such as the Saffman force f_{Saff} and the Magnus force f_{Mag}. In many applications the predominant forces are the drag force, the pressure gradient force and the viscous force. The remaining forces are often neglected. Also in the presented case the fluid particle interaction force is calculated as

$$f_{p,f} = f_d + f_{\nabla p} + f_{\nabla \cdot \tau}. \tag{5.12}$$

The forces are taken into account by different force models. A commonly used drag-model by Gidaspow et al. [1992] combines the drag relations by Ergun and Orning [1949] and Wen and Yu [1966]. Another approach was published by Felice [1994]. This model uses a voidage function that depends on the void fraction in the system and the particle Reynolds number Re_p which is defined as

$$Re_p = \frac{\alpha_s \rho_f |u - u_p| d_p}{\mu}. \tag{5.13}$$

One of the models used for the presented calculations was developed by Koch and Hill [2001] and Koch and Sangani [1999]. In the formulation used by Buijtenen et al. [2011] the force term for the drag is given as

$$f_d = \frac{V_p \beta}{\alpha_s}(u - u_p). \tag{5.14}$$

V_p is the particle volume and β denotes the inter phase momentum exchange term, being defined as

$$\beta = \frac{18 \mu \alpha_f \alpha_s}{d_p^2} F. \tag{5.15}$$

F is defined in dependence of the particle Reynolds number:

$$F = \alpha_f \left(F_0(\alpha_s) + \frac{1}{2} F_3(\alpha_s) Re_p\right) \tag{5.16}$$

The functions F_0 and F_3 are given as

$$F_0(\alpha_s) = \begin{cases} \frac{1+3\sqrt{\frac{\alpha_s}{2}}+\frac{135}{64}\alpha_s \ln \alpha_s + 16.14\alpha_s}{1+0.681\alpha_s - 8.48\alpha_s^2 + 8.16\alpha_s^3} & \alpha_s < 0.4 \\ \frac{10\alpha_s}{\alpha_f^3} & \alpha_s \geq 0.4 \end{cases} \tag{5.17}$$

and

$$F_3(\alpha_s) = 0.0673 + 0.212\alpha_s + \frac{0.0232}{\alpha_f^5}. \tag{5.18}$$

The pressure gradient force $f_{\nabla p}$ is given as

$$f_{\nabla p} = -V_p \nabla p, \tag{5.19}$$

and the viscous force $f_{\nabla \cdot \tau}$ is calculated as

$$f_{\nabla \cdot \tau} = -V_p \nabla \cdot \tau. \tag{5.20}$$

The governing equations are completed by initial and boundary conditions for both phases. Tab. 5.1 contains a collection of the governing equations used to compute the unresolved problems in the further course of this thesis.

Fluid phase:
Continuity equation: $\frac{\partial \alpha_f}{\partial t} + \nabla \cdot (\alpha_f u) = 0$ Momentum equation: $\frac{\partial (\alpha_f \rho_f u)}{\partial t} + \nabla \cdot (\alpha_f \rho_f uu) - \nabla \cdot (\alpha_f \tau) = -\alpha_f \nabla p + R_{sl} + \alpha_f \rho_f g.$ $R_{s,l} = \frac{1}{\Delta V} \sum f_{d,i}$
Solid phase:
$m_p \frac{du_p}{dt} = m_p g + f_{p,f} + \sum_{N_p} f_{p,p} + \sum_{N_w} f_{p,w}$ $f_{p,f} = f_d + f_{\nabla p} + f_{\nabla \cdot \tau}$ Drag force, pressure gradient force and viscous force are computed according to the presented relations.

Tab. 5.1: Summary of the unresolved CFD-DEM governing equations (Type A).

5.2 Time step width

The equations presented in Tab. 5.1 are discretized for the numerical computations (cf. Appendix A). In most cases time step widths in DEM are smaller than in CFD. For CFD calculations the Courant-Friedrichs-Lewy number (also CFL number or Courant number) establishes a connection between time step with Δt_f, grid size Δx and the fluid velocity u:

$$CFL = \frac{u \cdot \Delta t_f}{\Delta x}. \tag{5.21}$$

It indicates the number of cells a "fluid particle" passes in one time step. Consequently, a calculation can only be considered stable if the Courant number is smaller or equal to one.

In DEM calculations the time step widths depends on the mean free path lengths of the particles, i.e. the distance a particle can move before colliding with another particle or the wall. Especially in densely packed beds this quantity is by far smaller than the average cell size of the CFD mesh, thus also a smaller time step width is needed for obtaining accurate results.

In coupled calculations also the ability of a particle to follow the fluid flow plays a role. A measure for this quantity is the particle relaxation time, given by

$$\tau_p = \frac{\rho_p d_p^2}{18\mu}(1 + 0.15Re_p^{0.687})^{-1}, \tag{5.22}$$

whereas in this context the particle Reynolds number is defined as

$$Re_p = \frac{\rho_f(u - u_p)d_p}{\mu}. \tag{5.23}$$

Particles with large relaxation times show stronger resistance to the flow motion as particles with small relaxation times do. For a stable calculation the DEM time step width must not exceed the particle relaxation time.

A common practice for decreasing the computational costs and guaranteeing accurate results at the same time is the use of different time scales for the CFD and the DEM calculations. The later on used code supports this approach. The time step widths can be chosen independently and the inter-phase data exchange takes place at specific coupling steps.

With numerical methods one generally approximates the solutions of the regarded problems instead of solving them in an exact way. This involves small errors, whose behaviour is of great interest. Goal of the presented study is on the one hand to show how the particle relaxation time affects the numerical error and on the other hand to investigate on the effect of the coupling interval size. Besides the method that is used in CFDEMcoupling (Base Case) three further approaches (Case I-III) were implemented and tested.

Prerequisites

For the investigation the force model by Koch and Hill (cf., Sec. 5.1) was implemented in matlab. The code was written for a 1D case. The fluid's motion was prescribed with the aid of a velocity profile, the only influence on the particle motion was assumed to be the exerted drag force.

Generally, the CFD time step width is assumed to be larger than the DEM time step width. Thus, with regard to the time line, new fluid velocity information is available at a coarser grid than DEM data. The points in time at which CFD calculations take place are denoted with t_i. Between two such CFD-data points an arbitrary number of DEM steps are carried out at the times $t_{i,1}, ..., t_{i,-\#DEM}$. $\#DEM$ is the total number of DEM steps per CFD time step. Fig. 5.1 illustrates the described timeline.

The particle's motion was calculated using eq. (5.10). All force terms apart from the drag force were omitted. The latter is a the product of the relative velocity and the drag

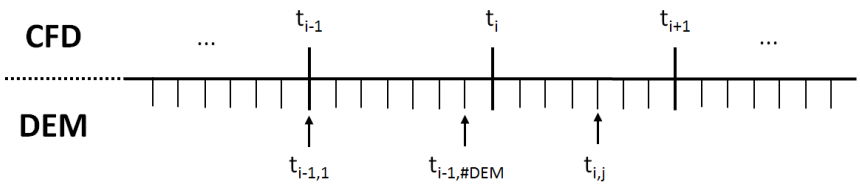


Fig. 5.1: In this test, the coupling is conducted at each CFD time step. Between two couplings, an arbitrary number of DEM calculations takes place.

coefficient dependent quantity β. The replacement of the time derivative by the difference quotient lead to

$$\frac{u_{p,new} - u_{p,1}}{\Delta t} = \frac{V_p}{m_p \alpha_p} \beta(u_{p,2})(u_f - u_{p,3}), \tag{5.24}$$

where $u_{p,1}$, $u_{p,2}$, $u_{p,3}$ and u_f are particle and fluid velocities, which might stem from different points in time. $u_{p,new}$ is the desired particle velocity at $t = t_{i,j}$. If calculated in an explicit manner, $u_{p,new}$ is evaluated as

$$u_{p,new} = u_{p,1} + \Delta t \left(\frac{V_p}{m_p \alpha_p} \beta(u_{p,2})(u_f - u_{p,3}) \right). \tag{5.25}$$

In CFDEMcoupling, the fluid velocity and the particle velocities for the force term are only updated at the coupling times. This case will further on be referred to as "Base Case". For the sake of comparison three further configurations were investigated:

- Case I: In order to increase the accuracy the particle velocity used for the relative velocity, $u_{p,3}$, was updated every DEM step. β was sill calculated at the coupling steps and kept constant for the DEM calculations.
- Case II: As second remedy, all particle velocity values $u_{p,1}$, $u_{p,2}$ and $u_{p,3}$ were updated at each DEM time step.
- Case III: The three previously described cases were fully explicit. In this last case, the a semi-implicit approach was applied: the relative velocity was chosen in dependence on the desired particle velocity. The new particle velocity was thus determined as

$$u_p(t_{i,j}) = (1 + \frac{V_p \Delta t}{m_p \alpha_p} \beta(u_{p,2}))^{-1} (u_{p,1} + \frac{V_p \Delta t}{m_p \alpha_p} \beta(u_{p,2}) u_f(t_i)). \tag{5.26}$$

  Following the example of the Base Case, the particle velocity in the difference quotient $u_{p,1}$ was updated at each DEM step, while β was calculated using the particle velocity known at the last coupling step.

The respective particle and fluid velocities assigned to $u_{p,1}$, $u_{p,2}$ and $u_{p,3}$ are collected in Tab. 5.2.

The fluid in the calculations was assumed to be air at a temperature of 20°C. The solid material was glass. For obtaining different relaxation times, the particle diameter was varied between 10^{-3} and $2 \cdot 10^{-7}$ m. The material properties are collected in Tab. 5.3.

	$u_{p,1}$	$u_{p,2}$	$u_{p,3}$
Base Case	$u_p(t_{i,j-1})$	$u_p(t_{i-1,\#DEM})$	$u_p(t_{i-1,\#DEM})$
Case I	$u_p(t_{i,j-1})$	$u_p(t_{i-1,\#DEM})$	$u_p(t_{i,j-1})$
Case II	$u_p(t_{i,j-1})$	$u_p(t_{i,j-1})$	$u_p(t_{i,j-1})$
Case III	$u_p(t_{i,j-1})$	$u_p(t_{i-1,\#DEM})$	$u_p(t_{i,j})$

Tab. 5.2: Particle and fluid velocities for the different cases, to be inserted in eq. (5.24).

Fluid - Air	
density	1.1885 kg/m^3
kinematic viscosity	$1.532 \cdot 10^{-5}$ m^2/s
Bulk solid - Glass	
density	2500 kg/m^3
diameter	0.001 m - $2 \cdot 10^{-7}$ m

Tab. 5.3: Material properties.

For being able to investigate on several aspects, two different velocity profiles were used: in one case a constant velocity of 10 m/s was assumed and in the other case a constantly accelerating velocity profile was prescribed.

Investigation I

A resting particle was placed in a constantly moving fluid. With the aid of the particle diameter the particle's relaxation time was varied between 7.62 s and $3.05 \cdot 10^{-7}$ s. As the flow velocity was constant, the coupling interval size had no effect in this case. The calculation was stopped as soon as the particle reached 90% of the flow velocity. The error was defined as the normalized difference between the integrals over the velocities:

$$error_1 = \frac{\int u_{coupling} - \int u_{comparative}}{\int u_{comparative}}. \tag{5.27}$$

Fig. 5.2 illustrates the resulting relative errors for particle relaxation times between 7.62 s and $7.62 \cdot 10^{-4}$ s, Tab. 5.4 contains the errors for particle relaxation times between 7.62 s and $3,05 \cdot 10^{-7}$ s. Case I already shows a significant improvement in comparison to the Base Case. Further accuracy is gained in Case II, where also the drag-related term β was updated at each DEM step. The particle velocity influences β via the particle Reynolds number. The Reynolds number depends on the relative velocity between particle and fluid, which which is larger for cases with bigger relaxation times. In all cases the accuracy of Case I and Case III are in accordance. Generally, as predicted by the theory, larger particle relaxation times lead to smaller numerical errors.

Tab. 5.4 contains a number of empty cells for those cases where no valid result could be obtained. The first method to become unstable for decreasing particle relaxation times is the Base Case, followed by Case I and Case II. As being semi-implicit, Case III shows the best stability behaviour, even though it was shown to be of the same accuracy as Case I previously.

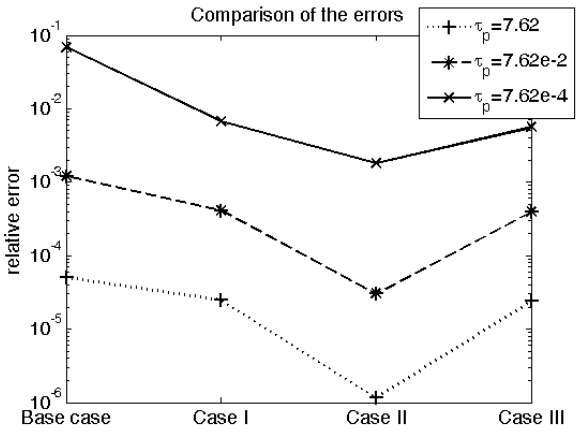


Fig. 5.2: Relative errors for different particle relaxation times.

	Base Case	**Case I**	**Case II**	**Case III**
$\tau_p = 7.63$ $d = 1 \cdot 10^{-3}$	t_{90}: 1.77 s error: $5.03 \cdot 10^{-5}$	t_{90}: 1.77 s error: $2.45 \cdot 10^{-5}$	t_{90}: 1.77 s error: $1.15 \cdot 10^{-6}$	t_{90}: 1.77 s error: $2.39 \cdot 10^{-6}$
$\tau_p = 7.63 \cdot 10^{-2}$ $d = 1 \cdot 10^{-4}$	t_{90}: 0.07 s error: $1.2 \cdot 10^{-5}$	t_{90}: 0.07 s error: $4.08 \cdot 10^{-4}$	t_{90}: 0.07 s error: $3.04 \cdot 10^{-5}$	t_{90}: 0.07 s error: $2.93 \cdot 10^{-4}$
$\tau_p = 7.63 \cdot 10^{-4}$ $d = 1 \cdot 10^{-5}$	t_{90}: $9.0 \cdot 10^{-4}$ s error: $6.85 \cdot 10^{-2}$	t_{90}: $1.1 \cdot 10^{-3}$ s error: $6.7 \cdot 10^{-3}$	t_{90}: $1.1 \cdot 10^{-3}$ s error: $1.8 \cdot 10^{-3}$	t_{90}: $1.1 \cdot 10^{-3}$ s error: $5.6 \cdot 10^{-3}$
$\tau = 7.62 \cdot 10^{-6}$ $d = 1 \cdot 10^{-6}$	- -	$t_{90} < 1 \cdot 10^{-4}$ error: $1.7 \cdot 10^{-2}$	$t_{90} < 1 \cdot 10^{-4}$ error: $1.6 \cdot 10^{-2}$	$t_{90} < 1 \cdot 10^{-4}$ error: $5.8 \cdot 10^{-3}$
$\tau_p = 3.05 \cdot 10^{-7}$ $d = 2 \cdot 10^{-7}$	- -	- -	- -	$t_{90} < 1 \cdot 10^{-4}$ error: $5.1 \cdot 10^{-3}$

Tab. 5.4: Relative errors for relaxation times between 7.62 s and $3,05 \cdot 10^{-7}$ s.

Investigation II

The next investigation dealt with the effect of the coupling interval size on the accuracy of the results. In the first test-series a sphere with a diameter of 10^{-4} m was placed in fluid with a velocity of 10 m/s. During the total run time of 0.1 s the velocity was linearly increased to 12 m/s. The DEM time step width was chosen as 10^{-6} s, the coupling interval sizes were varied between $5 \cdot 10^{-6}$ and 10^{-3} s.

Fig. 5.3 shows the numerical error, calculated according to eq. (5.27). It is plotted over the ratio of the particle relaxation time and the coupling interval width for all four cases. The accuracy of all methods increased with decreasing coupling interval sizes and growing particle relaxation times. Switching from the Base Case to Case I and III yielded an improvement of roughly one order of magnitude, Case II lead to even better results.

For drawing general conclusions from the presented results, some considerations were necessary. The basis for the further steps in this investigation were the previously obtained errors for the Base Case. However, the outcomes are of general nature and also apply for the other cases. In Fig. 5.3 each ratio of relaxation time and coupling interval size is identified with a specific error. Consequently, if the particle diameter and thus the particle relaxation time changes, the coupling interval needs to be adjusted for keeping the error constant. This allows for the development of a time step choice criterion, which starts with the relation

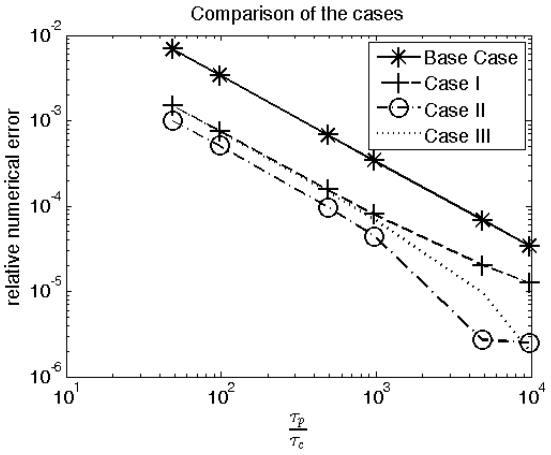


Fig. 5.3: Comparison of the numerical error for Base Case and Case I-III. Plotted over the ratio of the particle relaxation time τ_p and the coupling interval size τ_c.

$$\frac{\tau_{p,1}}{\tau_{c,1}} = \frac{\tau_{p,2}}{\tau_{c,2}}. \tag{5.28}$$

$\tau_{p,1}$ and $\tau_{c,1}$ are original particle relaxation time and coupling interval size respectively, $\tau_{p,2}$ and $\tau_{c,2}$ are the corresponding quantities for the particle with the changed diameter. The particle relaxation time is chosen according to (5.22), thus $\tau_{c,2}$ can be calculated as

$$\tau_{c,2} = \left(\frac{d_{p,2}}{d_{p,1}}\right)^2 \frac{1 + 0.15(\frac{u_1 - u_{p,1}}{\nu_f} d_{p,1})^{0.687}}{1 + 0.15(\frac{u_2 - u_{p,2}}{\nu_f} d_{p,2})^{0.687}} \tau_{c,1}. \tag{5.29}$$

The values for $u_{p,1}$ and $u_{f,1}$ are the particle and fluid velocity at the last time step of the initial calculation. For the new particle these quantities are unknown. One possibility consists in approximating them with the values of the original particle, which leads to

$$\tau_{c,2} = \left(\frac{d_{p,2}}{d_{p,1}}\right)^2 \frac{1 + 0.15(\frac{u_1 - u_{p,1}}{\nu_f} d_{p,1})^{0.687}}{1 + 0.15(\frac{u_1 - u_{p,1}}{\nu_f} d_{p,2})^{0.687}} \tau_{c,1}. \tag{5.30}$$

This relation was used to determine corresponding coupling interval sizes for particles with diameters of 1, 3 and $5 \cdot 10^{-5}$ m. The values are displayed in Tab. 5.5.

Original particle: $d_p = 10^{-4}$	Particle 1: $d_p = 10^{-5}$	Particle 2: $d_p = 3 \cdot 10^{-5}$	Particle 3: $d_p = 5 \cdot 10^{-5}$	Number of CFD steps:
10^{-3}	$1.4 \cdot 10^{-5}$	$1.13 \cdot 10^{-4}$	$2.9 \cdot 10^{-4}$	100
$5 \cdot 10^{-4}$	$7.0 \cdot 10^{-6}$	$5.65 \cdot 10^{-5}$	$1.45 \cdot 10^{-4}$	200
10^{-4}	-	$1.13 \cdot 10^{-5}$	$2.9 \cdot 10^{-5}$	1000
$5 \cdot 10^{-5}$	-	$5.65 \cdot 10^{-6}$	$1.45 \cdot 10^{-5}$	2000
10^{-5}	-	-	$2.9 \cdot 10^{-6}$	10000

Tab. 5.5: Coupling interval sizes that correspond to the intervals of the comparative case in terms of numerical errors.

The calculations were then run for the same number of CFD steps as the original calculations. This allowed for a direct comparison of the numerical errors. The newly calculated coupling interval sizes were used as CFD time step widths. As shown in Fig. 5.4, the errors for the new calculations match the errors of the original case.

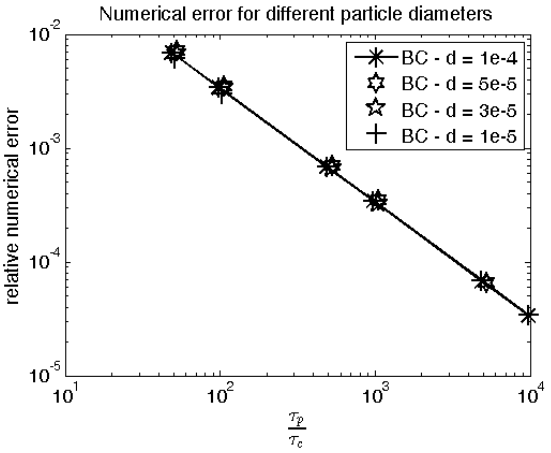


Fig. 5.4: Relative numerical errors for different particle diameters (BC ... base case).

In conclusion it can be stated that smaller particles follow the fluid flow faster than larger particles. Decreasing particle diameters lead to increasing numerical errors, thus problems involving smaller particles require a finer time resolution. For cases with non-constant velocity profiles it could be shown that the coupling interval size needs to be adjusted to the particle size, if a certain accuracy should be obtained. The modified methods in Case I, II and III show significant improvements in terms of error-reduction and increased stability. Even though this results could not be incorporated in the current implementation of the software, this knowledge is valuable for future work in this field.

6 Implementation of the Method

The Open Source framework CFDEMcoupling [2012] provides the solvers `cfdemSolverPiso` and `cfdemSolverPimple` for handling unresolved CFD-DEM problems. The implementation of the solvers is based on OpenFOAM's `pisoFoam` and `pimpleFoam` respectively. The solvers and most of the models were developed by Christoph Goniva. The following section will give a brief overview over the algorithmic structure.

6.1 cfdemSolverPiso and cfdemSolverPimple - The algorithm

The basis of the coupled tool is a CFD solver that uses functions to obtain DEM data and trigger the particle calculations. Initially a CFD mesh with initial and boundary conditions and the initial state of the particulate phase is required. The main structure of a coupled CFD-DEM solver is illustrated by Fig. 6.1:

In the first step of the calculation a data structure that will later on contain all particle data is created. As soon as this is accomplished, the overall time loop starts. At the beginning of each time step the particle-related operations are executed:

- First of all the **DEM data** from the present time step is **fetched**. The information is stored in the prepared data structure.
- In a next step the **particles** are **located** within the fluid mesh. The cell ID of the centre position of each particle is saved.
- After this location process the **void fraction field** is generated: depending on the amount of matter in a cell it is assigned a value between 0 (solid only) and 1 (fluid only). The force models used in these solvers only hold for cases where the particles are smaller than the fluid cells, i.e. a void fraction of 0 cannot be handled correctly. As to make sure that this prerequisite is met, a minimal void fraction is set. If the void fraction is corrected, a minimal error occurs.

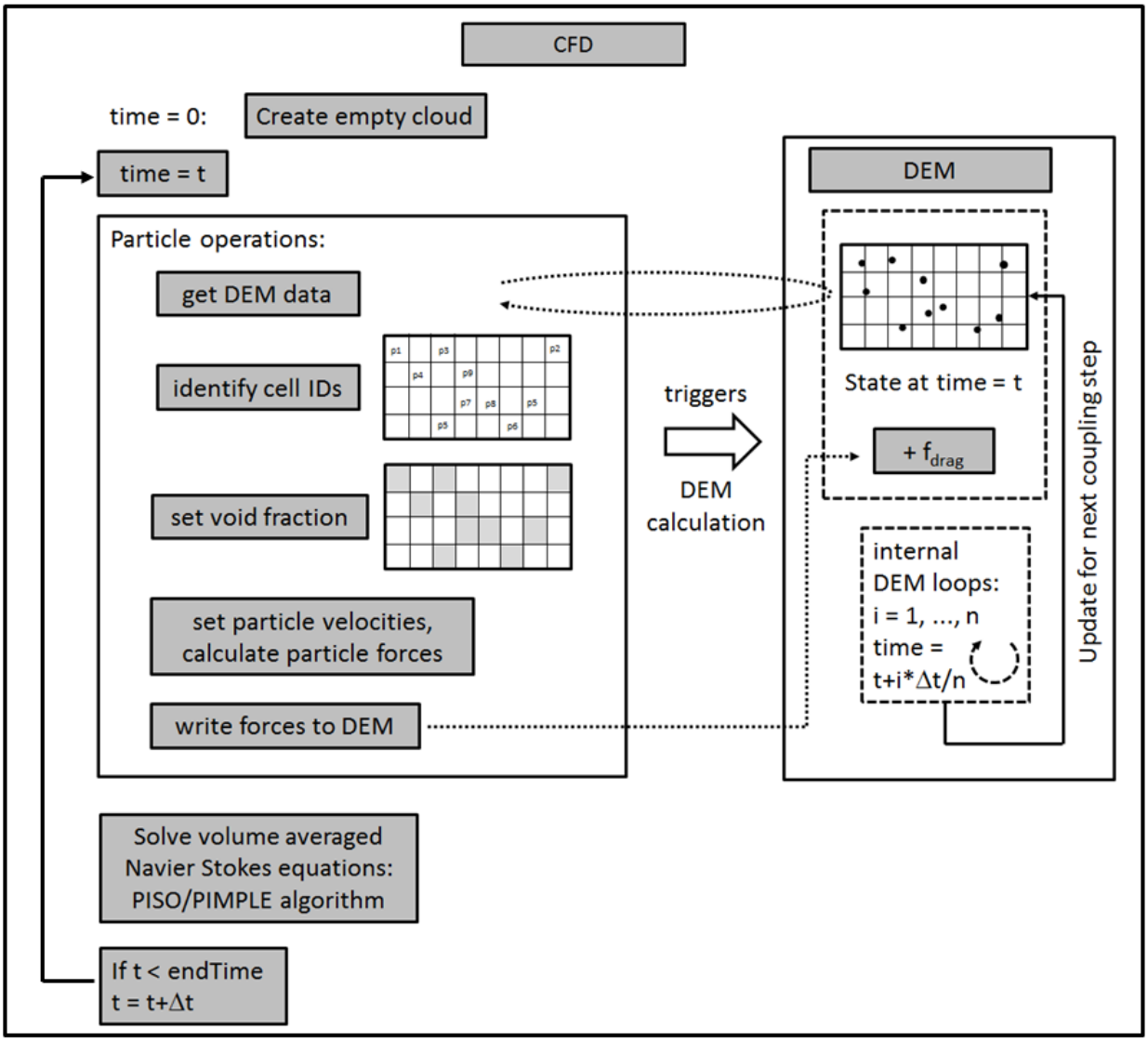


Fig. 6.1: Structure of the unresolved CFD-DEM algorithm.

- Based on the particle velocity, the void fraction field and the fluid velocity field a **particle velocity field** is calculated.
- This data is used to **evaluate** the **drag force** that acts onto each **particle**.
- Finally the **force** information for each particle is **communicated to the DEM solver**.

The CFD solver triggers the DEM solver, which starts its calculations. If the DEM time step width is smaller than the CFD time step width, multiple DEM calculations take place until the next coupling is established.

On the CFD side the volume averaged Navier-Stokes equations are solved for the present time step. Depending on the chosen solver either a PISO or a PIMPLE algorithm is applied.

The latter one is a combination of PISO and SIMPLE. Before the solver is ready for the next time step, the boundary conditions are adjusted.

The different model classes are described in the following:

- Averaging model: This model is responsible for transferring data from the Lagrangian particle cloud to the Eulerian mesh structure. The possible settings are "dilute" and "dense" for the different handling of dilute and dense particle phases.
- Clock model: The clock model provides a timer function that provides information about the run time of certain parts of the calculation.
- Data exchange model: In most cases CFD and DEM calculation are coupled in a two-way-MPI manner. Nevertheless, other options such as data transfer via files or one-way coupling can be chosen.
- Force model: In the literature one can find a big number of models that can be used to compute the forces, the fluid exerts onto the particles. CFDEMcoupling provides a selection, comprising the ones discussed in Sec. 5.
- IO model: If the IO model "write" is activated, a folder named "lagrangian" is generated in each time step folder of the CFD calculation. Position, velocity and radius of each particle allow for a direct visualization of the Lagrangian particle cloud within the postprocessor paraview.
- LIGGGHTS command model: As customary for CFDEMcoupling, the CFD solver is the master process that calls a LIGGGHTS input script. This is evoked by the command "runLiggghts" in the dedicated LIGGGHTS-command dictionary.
- Locate model: When the particle information is written to the mesh, the position of each particle, namely the ID of the mesh cell that contains the object, needs to be determined. As unresolved CFD-DEM calculations usually comprise large numbers of particles that need to be located within the CFD mesh, it is crucial to have an efficient model at hands. Apart from the rather slow standard search model a tree search approach was implemented, termed "engine search". Improved versions were added later on.
- Momentum couple model: It is responsible for the coupling between CFD and DEM. `cfdemSolverPiso` and `cfdemSolverPimple` contain an implicit force term, meaning that the option "implicitCouple" has to be chosen in the "couplingProperties" -dictionary. The alternative is "explicitCoupling", which is applied when a solver involves an explicit source term. If the solver permits it, also combinations of the two coupling modes are possible.
- Region model: In some cases only parts of the computational domain contain particles, allowing for a partially coupled calculation. The implemented class provides the basic framework for such a model, so far only the default model "allRegion" is available.

- Smoothing model: Smoothing models can be applied to the data exchange fields. Basically a diffusion equation is used to get smoother exchange fields. Especially when the mesh cells are small in comparison to the particle diameter instabilities can be avoided with their aid.

- Void fraction model: After the particles were located in the mesh, their impact onto the flow field needs to be determined. This can only be accomplished, if the solid covered volume is known. Various void fraction models were implemented for that purpose, differing in efficiency and exactness.

Further information about the models and the available options can be found in the documentation of CFDEMcoupling. A big advantage of CFDEMcoupling lies in the fact that solvers and models are independent of each other: the user can decide, which model to use, and change one or several models from one case to another. The code remains completely untouched; the model settings are mostly made in the dictionary `couplingProperties`. The only exception so far is the LIGGGHTS command model that requires an own input file.

Apart from the free choice of the models for the handling of the particle data one can choose whether the equations solved are the ones of "model A" or "model B" (c.f., 5). So in each time step - after the coupling operation is carried out (cf., Chap. 3) - one of the two equation systems is solved. All solvers can be run in parallel. A brief benchmark study (cf., App. D.4) demonstrates the code's good performance.

7

Application: Raceway Formation Inside a Blast Furnace

In steel production, blast furnaces are used for smelting iron ore to obtain iron. Coke, iron ore and other components are charged from the top, while blast air enters the system through tuyeres. Tuyeres are nozzles that are arranged in a circle around the furnace. Fig. 7.1 shows a schematic sketch of a blast furnace.

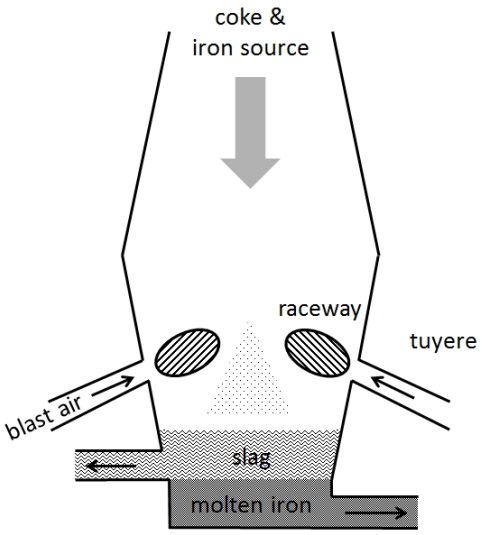


Fig. 7.1: Schematic sketch of a blast furnace.

In the following the process of iron making is discussed. The comments are based on a description by Biswas [1981].

Inside the furnace iron ore smelts and molten iron and slag are extracted at the bottom.

The coke reaches the tuyere level in solid state and gasifies in front of the tuyeres. In front of the tuyeres the blast air leads to the formation of cavities, termed raceways. Their roof is formed by rather loose pieces of coke while the walls are densely packed. Inside the tuyeres loose coke particles circulate. Even though molten material enters them eventually, the momentum of the air jet keeps the area in front of the tuyeres clear of material.

The combustion process consists of several steps: first the coke oxidises due to the contact with the blast air and carbon dioxide is produced. In this step energy or rather heat is released. Due to the high temperatures the CO_2 is unstable and thus another oxidation leads to the emergence of carbon monoxide. Neither the oxygen nor the carbon dioxide vanishes completely. The gas composition varies depending on the distance to the air inlet. The region with high CO_2 concentration is often called oxidizing zone as entering metal can be re-oxidised. The area where CO predominates is called reducing zone.

Melting and combustion cause a downwards movement of the granular material. The lowering should be as evenly as possible. This can be achieved by a big total raceway area. The size of the area depends both on the number of tuyeres and the size of the individual raceway. The main factors that influence the size of the individual raceway are given by

- a blast parameter,
- the specifications of the tuyere,
- the grain size distribution,
- the flow pattern inside the bed and
- the composition of the blast air.

The blast parameter is dimensionless and comprises blast-air related information such as temperature, pressure, volumetric flow rate and density, coke-related information like density and diameter and the tuyere dimensions. It accounts for the relation between raceway depth, momentum of the gas and gravitational acceleration of the coke particles. In industrial furnaces average penetration depths between 1 and 2 metres are common, bigger depths will be found in larger furnaces. Increasing the kinetic energy of the blast leads to bigger combustion areas in the inter-tuyere area. High flow rates cause raceways with smaller vertical and higher horizontal expansion. Decreasing the tuyere diameter increases the kinetic energy, i.e. the raceway depth increases. Increased tuyere diameters lead to raceways with a higher vertical expansion. Combustion areas that reach deep inside the furnace lead to high temperatures close to the axis and a loosening of the material in the dead man - an area in the centre of the furnace, where almost not material motion is observed. The penetration depth can also be influenced by the length of the tuyere and also the shapes of the tuyere nose have a remarkable effect on the formation of the raceways. Another important influence parameter is the flow pattern inside the furnace: high gas flow rates close to the walls loosen the material in this area and cause smaller combustion areas than a more uniform gas flow. Furthermore blast air modifications by oxygen enrichment or the addition of natural gas were used to influence the furnace performance. Biswas [1981] provides more details on the topic.

For classifying processes in terms of efficiency, detailed information about shape and size of the developing raceways is required. Direct measurements in the real system are difficult to impossible, and so the application of numerical methods comes into play. The objective of this work was finding a simulation tool that is capable of depicting raceways in specific regimes and that can be validated against experiments.

For obtaining such a model a simple roadmap was followed: First a small scale cold 2D-model was built up in the laboratory. The main geometry consisted of acrylic glass; the granular phase was represented by glass spheres or polypropylene beads. The size relations of the different components were chosen to fit those in an actual furnace. This model allowed detailed measurements of the raceway formation. A complying unresolved CFD-DEM calculation was designed. While the boundary conditions for the CFD computations could be measured in the experiment, the determination of the material properties was more complex: additional experiments were necessary to establish numerical parameters such as inter-particle or wall-particle friction, restitution etc. The visual data gained in both cases was post processed with the same software tools, guaranteeing a direct comparison. In a second step the numerical tool was applied to a full size 3D simulation. The new geometry depicted a wedge of a full size blast furnace. The results of this investigation could be compared to heuristic models used in industry.

7.1 Literature research and theoretical background

The formation of raceways is – as the number of recent publications show – a hot topic (cf., Malmberg et al. [2007]; Natsui et al. [2011a]; Sakai et al. [2012]; Sterneland et al. [2003]; Umkage et al. [2007]). Starting from advanced analytical models that consider a balance between bed weight and the momentum of the blast air over heuristically determined correlations based on data gained through experiments to numerical methods: the approaches used are most different.

The process involves chemical reactions as well as the interaction of at least four phases and therefore it is of great importance to focus on certain aspects. One technique for gaining knowledge about the formation of raceways consists in the conduction of experiments.

Experiments

Experiments can either be hot, if combustion is involved or cold otherwise.

Cold experiments Cold experiments are less complicated and a lot cheaper than hot experiments. Usually downscaled geometries are used, which makes the handling easier. As the combustion process is not part of the investigation, the original materials can be replaced by handier ones. Sarkar et al. [2003] for example used glass beads, MacDonald and Bridgewater [1997] reported using mustard seeds. Due to the absence of chemical reactions also the melting of material is not accounted for, which reduces the number of phases by one. Also the emergence of fines can often be neglected. This leads to a remarkable diminution of the number of process parameters.

These simplifications have a large benefit: in most cases investigations consist of both a practical and a theoretical part. For the theoretical part, models have to be found, no

matter if numerical or analytical, and their complexity strongly depends on the number of input parameters. Examples of this approach can be found in publications by Rajneesh et al. [2004], Sastry et al. [2003] and Sarkar et al. [2007]. Most of these experiments are 2D or pseudo-2D, allowing for optical measurement techniques.

Hot experiments Hot experiments are a lot more complex. There are approaches, where down-scaled test-reactors were built up, especially designed for experimental purposes (e.g., Sterneland et al. [2003]). As their purpose lies in yielding data rather than working most efficiently at all times, it is possible to run them in different operational states to study the influences of the different process parameters closely.

Another method is to gain results by on-site measurements at a real blast furnace. High temperatures and the presence of fumes and dust usually rule out conventional measurement techniques where either sensors are inserted into the area of interest or optical measurements are made from the outside. A method that has been applied in other disciplines before and made an appearance in the discussed field as well is microwave technology. The advantage of the electromagnetic waves is that they can propagate through matter and are insensitive to fines. The pattern of wave propagation changes depending on the properties of the material, the waves pass through, and thus microwave technology is well suited for measurements within the steel and metal industries. Malmberg et al. [2007] give an overview over the technique, further investigations are presented by Matsui et al. [2005b,a].

Heuristic and analytic models

In many cases experimental data is not only used for directly extracting information, but also for validating and building up analytical and numerical models. Especially simplified models are well suited for that purpose. Before setting up a model, influence properties have to be determined. Their number clearly depends on the complexity of the given setting; generally they can be assigned to three different groups, in the present case of a cold pseudo 2D model the groups could be called geometry-related, gas-related and particle-related.

The *geometry-related* influences describe the dimensions of the system, such as total height and diameter of the overall geometry, properties of the tuyere, etc. Amongst the *gas properties* for example density, inlet velocity at the nozzles and viscosity can be found. In more general cases also temperature plays a role here. The *particle-related* properties depend strongly on the model chosen, in the presented case, slightly poly-disperse spheres of equal density were regarded. In more complex models, several classes of particles that account for different materials or non-spherical objects can be used. The group of solid-related quantities comprises

- particle diameter or grain size distribution,
- density,
- angle of repose,
- packing density and
- bed height.

For guaranteeing the comparability of a preferably large number of cases, dimensionless quantities can be used. Those can stem from geometric relations such as the relation of the tuyere diameter to the particle diameter or the tuyere diameter to the bed height. Furthermore established quantities such as the (particle-) Reynolds number and the Froude number are used in that context. Several attempts for finding models to predict the raceway penetration depth for specific conditions were made. An example for a joint analytic and heuristic parameter choice is given in the following. It was published by Rajneesh et al. [2004] and gives a step-by step description for developing a model.

First the set of influence parameters is determined and raceway diameter is expressed as unknown function of those quantities:

$$d_r = f(\rho_s, \rho_g, \epsilon, v_b, g, d_p, \phi, \mu_g, d_T, H, W, \mu_w, \mu_p). \tag{7.1}$$

ρ_s	...	solid density (kg/m^3)
ρ_g	...	gas density (kg/m^3)
ϵ	...	gas volume fraction
v_b	...	blast velocity (m/s)
g	...	gravitational constant (m/s^2)
ϕ	...	sphericity of the particles
μ_g	...	viscosity of the gas phase (kg/(ms))
d_T	...	tuyere diameter (m)
H	...	bed height (m)
W	...	bed width (m)
μ_w	...	particle-wall coefficient of friction
μ_p	...	particle-particle coefficient of friction

As granular beds are considered, the solid density is replaced by the bulk density ρ_{bulk}:

$$\rho_{bulk} = \epsilon\rho_g + (1-\epsilon)\rho_s \tag{7.2}$$

For spherical particles the sphericity coefficient is equal to one, otherwise the diameter d_{eff} used in the computations is $\phi \cdot d_p$. This reduces the total number of variables from 14 to 12. The number of fundamental physical dimensions is three (kg, m and s). The goal is to obtain a dimensionless formulation. The Buckingham-π-theorem (cf., Spurk [1992]) states that the number of dimensionless expressions is equal to the number of original variables minus the number of independent physical quantities, which in this case leaves 9 terms. Rajneesh et al. [2004] identify the dimensionless expressions and combine them. This results in

$$\frac{d_r}{d_T} = k \left(\frac{\rho_g}{\rho_{bulk}}\right)^a \left(\frac{v_b^2}{g d_p}\right)^b \left(\frac{\rho_g v_b d_p}{\mu}\right)^c \left(\frac{d_T}{W}\right)^d \left(\frac{d_T}{H}\right)^e (\mu_w)^f (\mu_p)^g \left(\frac{d_p}{d_T}\right)^h. \tag{7.3}$$

They claim that the raceway formation is a continuum process and thus drop the last term. By arguing that the particle wall-friction dominates the particle-particle friction the $(\mu_p)^g$ is omitted as well. In the next step they use a regression analysis for adjusting this simplified expression to a set of data. It turns out that c is small in comparison to the other

values and that the coefficients a, b, d and e are of the same order of magnitude. Thus a second simplification is carried out, leading to

$$\frac{d_r}{d_T} = k \left(\frac{\rho_g v_b^2 d_T^2}{\rho_{bulk} g d_{eff} HW} \right)^a (\mu_w)^b . \tag{7.4}$$

They now use two different sets of data for another regression analysis. In one case the inlet velocity is increasing, in the other case it is decreasing. This leads to two different final relations, as the decreasing case accounts for the hysteresis. Methods like these are rather popular, however, in many cases the resulting relations only hold for a very limited class of applications.

The hysteresis effect is mentioned in most publications and is also clearly displayed by the investigations carried out in the course of this thesis: for cases where the inlet velocity of the blast air is first increased up to a certain maximum and then decreased again, the raceway remains constant for a while despite decreasing blast velocity. This can be explained by inter-particle and particle-wall forces: during the development of the raceway a certain amount of energy is necessary to overcome the resistance stemming from those influences. As soon as the raceway is fully developed, these forces stabilize the existing cavity.

Gupta et al. [2005]; Gupta and Rudolph [2006] argue that pressure force, bed weight and frictional forces form a force balance. They introduce and analytic approach, where they first determine the flow field by dividing the whole domain into a radial part, where the raceway is located and a Cartesian part everywhere else. By equalizing the volumetric flow rate at the tuyere-outlet, the boundary of the radial domain and the surface of the Cartesian domain (which is equal to the surface of the particle bed) they are able to define the velocity profile analytically. With this information they express the pressure field. A force balance in the radial and Cartesian region is set up and its solution yields the normal stresses in the respective direction. The particle-particle and particle-wall forces are then determined by integrating the shear stresses over an appropriate distance.

Numerical models

After discussing experiments and analytical approaches for predicting shape and size of raceways inside blast furnaces, the class of numerical methods is discussed now. Here one can mainly differ between multi-fluid models and combined continuous and discrete models. When it comes to multi-fluid models, all phases are represented as continua. The averaged Navier-Stokes equations are solved for each phase, and a void fraction indicates the existence or non-existence of a certain species in a particular region. In Section 5.1 the idea of the TFM was discussed. Mondal et al. [2005] used such an approach in combination with a κ-ϵ-model for describing the turbulence of the gas phase. A review by Yu and Xu [2003] points out that the outcomes strongly depend on the closure relations used for the solid phase. Developing a general model of this kind is considered complex.

Over the past years, thanks to the increasing capacity of computers, combined continuous and discrete methods have become more and more important. The main idea behind these methods consists of the separate treatment of fluid and particulate phase and information interchange at runtime. The fluid phase is – just as in the previous case – governed by the

averaged Navier-Stokes equations. On the particulate side, Newton's equation of motion is solved for each particle and particle-particle and particle-wall collisions are taken into account (cf., Sec. 5.1).

Even though state-of-the-art computers are able to handle large systems, the costs for simulating an original-size blast furnace in every detail are rather high. A number of different approaches for overcoming the complexity-issue can be found in literature: Natsui et al. [2011a,b] for example represent a number of smaller particles by a bigger one, reducing the number of objects in the simulation domain significantly. This approach is known as parcel approach or coarse graining (cf., publications by Radl et al. [2011] or Sakai and Koshizuka [2009]; Sakai et al. [2012]. Another way of decreasing the size of the problems is simplifying them to 2D, like for example Yu and Xu [2003] or to compute a wedge only (cf., Umekage et al. [2005]; Umkage et al. [2007]) and for example apply periodic boundary conditions.

In the basic version of discrete element method, the particles are considered to be spherical. With the aid of simulation parameters such as the coefficient of restitution, friction or rolling friction one can account for a certain non-sphericity. Truly non-spherical particles can either be represented by a number of spheres that are lumped together ("multi-sphere method", cf., Ueda et al. [2010]) or superquadrics (cf., Hilton and Cleary [2011, 2012]). These methods require special drag laws for the depiction of the fluid forces onto the particles.

The following two sections present the work that was conducted within the course of this thesis.

7.2 A pseudo-2D investigation

Within the course of this study, experiments and simulations of a pseudo 2D model were conducted.

7.2.1 Experiments

For investigating the phenomenon of raceway development experimentally, a small scale test rig was built up in the institute's laboratory[1] (cf., Fig. 7.2). With its aid comparative data could be gained that would be used for the validation of the numerical model. For being able to conduct the simulations without the use of arbitrary tuning parameters, additional experiments for characterizing the material were necessary.

A detailed description of the experimental setting and the different operational modes is given in the first part of this section; in the second part the results are presented.

7.2.1.1 Cold Pseudo 2D-model

The main body of the pseudo 2D-model consisted of a acrylic glass box with the dimensions 600 mm × 50 mm × 1000 mm. The diameter of the granular material was between 2 and 4 mm, so the number of particles in depth always exceeded 10. The original configuration provided the possibility for two counter current air inlets, but during the first calibration experiments it turned out that the two air-jets do not influence each other. Therefore only

[1] The used test rig was built up by Christoph Kloss for a previous study. Parts of the measurements were conducted by Nikolaus Doppelhammer.

one air inlet was used and in the simulations only half of the domain was regarded. Blast air entered the domain through a nozzle at a height of 135 mm. The red arrow in Fig. 7.2 shows its position. For investigating the influence of the tuyere diameter, different nozzles with diameters between 13 and 20 mm were used for the test runs.

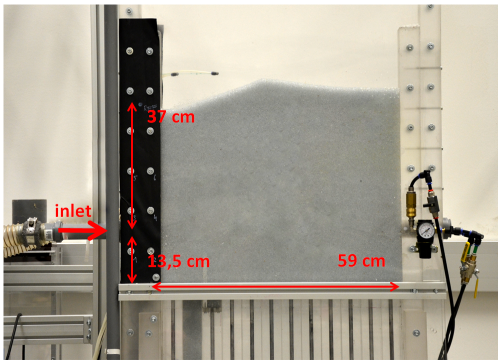


Fig. 7.2: Experimental set up, filled with 3 mm glass particles.

In order to emulate the gasification of the solid particles in the area of the raceway, material was removed underneath the cavity in the left section of the box. The material was extracted with a screw conveyor, as indicated in sketch 7.3.

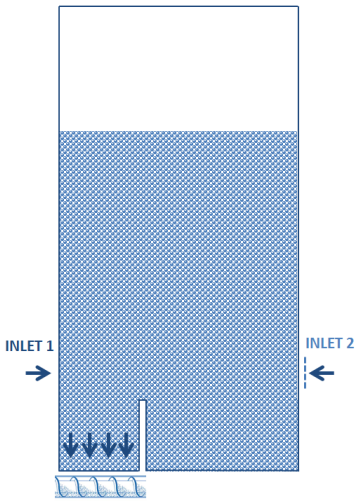


Fig. 7.3: Sketch of the pseudo 2D experiment. Air enters the setup through a nozzle at the left hand side and leaves at the top, while material is extracted in the lower left part.

The gap next to the material extraction region (cf., Fig. 7.3) indicates an acrylic glass plate that was inserted for ensuring that the material was extracted from the raceway area.

The particles used in the experiments were on the one hand glass spheres with a diameter of 2 and 3 mm respectively and on the other hand non-spherical polypropylene particles with an average diameter of 4 mm (cf., Fig. 7.4). In preliminary tests 4 mm glass spheres proved to be unsuitable, as the maximal momentum of the air jet did not suffice to trigger the development of a raceway. Tab. 7.1 contains a list with the material properties of the granular matter used.

name	material	shape	Ø (mm)	ρ (kg-m^3)	ρ_{bulk} (kg-m^3)	void fraction
Sili2mm	glass	spherical	2	2500	1500	0.38-0.4
Sili3mm	glass	spherical	3	2500	1500	0.39-0.4
Rounded	polypropylene	rounded (cf., Fig. 7.4)	4	828	558	0.33

Tab. 7.1: Material properties.

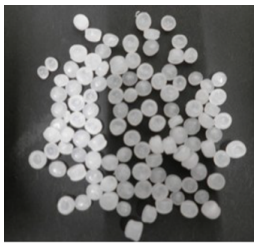


Fig. 7.4: Rounded4mm - rounded polypropylene cylinders with an average diameter of 4mm

The following four experimental configurations were considered:

- TEST1: The pure hysteresis test

  The hysteresis test was intended for determining the change of the system in dependence of the history. The fan engine was run with a performance of 60 and 80 percent alternately. The inlet velocities at the respective power settings differed slightly, as the void fraction of the bed increased slightly with the air passing through it and caused a change in the pressure drop.

- TEST 2: Increasing and decreasing the velocity gradually

  The boundary conditions for this series of experiments were inspired by the work of Sarkar et al. [2003]. For each test run the inlet velocity was first incremented gradually until a maximum was reached and then decremented correspondingly.

- TEST 3: Constant inlet velocities

  This test series was designed for a preliminary investigation of the effect of the screw conveyor onto the behaviour of the raceway. Both the volumetric flow rate of the blast air and particle removal rate of the screw conveyor were kept constant, while the development of the raceway was monitored.

- TEST 4: Increasing and decreasing velocity with screw conveyor

  In this series two of the previous tests were combined: the inlet velocity was varied just as in TEST 2 while at the same time material was extracted as in TEST 3.

Fig. 7.5 contains a schematic sketch of the different engine-power profiles of the fan.

Dimensional analysis

For showing that the parameters used in the test rig can be related with real systems, three dimensionless quantities were compared, namely

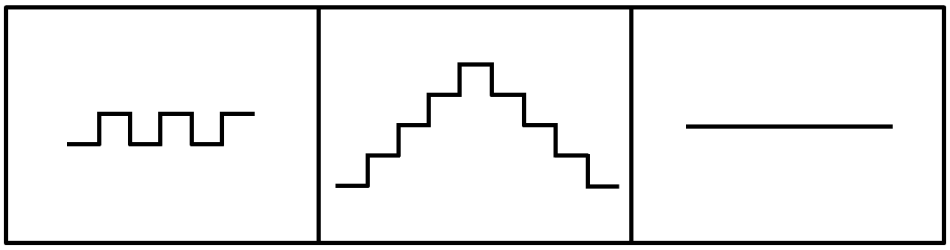


Fig. 7.5: Velocity profiles of the different experiments (left: TEST 1, middle: TEST 2 and 4, right: TEST3).

- the relation between particle diameter and tuyere diameter

$$\frac{d_p}{d_T}, \tag{7.5}$$

- the relation between the raceway diameter and the tuyere diameter

$$\frac{\sqrt{\frac{4 \cdot A_R}{\pi}}}{d_T} \tag{7.6}$$

- and the relation between particle weight and momentum flow

$$\frac{d_p^3 \rho_p g}{d_T^3 v_b^2 \rho_g}. \tag{7.7}$$

Table 7.2 contains the values that apply for industrial size blast furnaces and the according quantities for the test rig.

	blast furnace[2]	test rig
particle diamtere d_p	5-40 mm	2-4 mm
tuyere diameter d_T	120-200 mm	13-20 mm
raceway area A_R	5000 cm^2	20-180 cm^2
particle density ρ_p	1651 kg/m^3	828 - 2500 kg/m^3
gas density ρ_g	1.145 kg/Nm^3	1.145 kg/m^3
blas velocity v_b	59 Nm/h	40-140 m/s

Tab. 7.2: Geometrical and process parameters in an industrial size blast furnace and the experimental test rig.

Tab. 7.3 contains a comparison of the dimensionless quantities computed from the given values. The covered data-ranges are in accordance.

[2] The quantities for the blast furnace were provided by industrial partners of the Christian Doppler Laboratory on Particulate Flow Modelling.

	blast furnace	test rig
d_p/d_T	0.025 - 0.2	0.1 - 0.3
$\frac{\sqrt{4 \cdot A_R/\pi}}{d_T}$	4.0-6.6	3.9-7.5
$\frac{d_p^3 \rho_p g}{d_T^3 v_b^2 \rho_g}$	1.3e-05-0.0065	7.2e-06-0.0051

Tab. 7.3: Comparison of the dimensionless quantities.

Data extraction

The use of acrylic glass for the test rig in combination with the translucent glass and plastic particles allowed for a relatively simple optical post processing: a spotlight was mounted either in front of the setup or behind it and the experiments were recorded with a standard camcorder. The single frames of the video were processed with an image processing routine realized in the matlab toolbox. Mean value, variation and standard deviation of the raceway area were determined. Furthermore the mean diameter of the equivalent circle is calculated. This quantity was used for the comparison of the results and is termed "equivalent raceway diameter" from now on. From the resulting images information about the shape of the raceway, i.e. horizontal vs. vertical diameter could be extracted.

7.2.1.2 Results

Global bed movement

Due to the extraction of material in the raceway-region, a global bed motion developed. The removed material was re-introduced to the system. In comparison to un-recycled particles the used ones showed a slightly darker colouring. Fig. 7.6 shows the colour pattern of the bed after a number of tests.

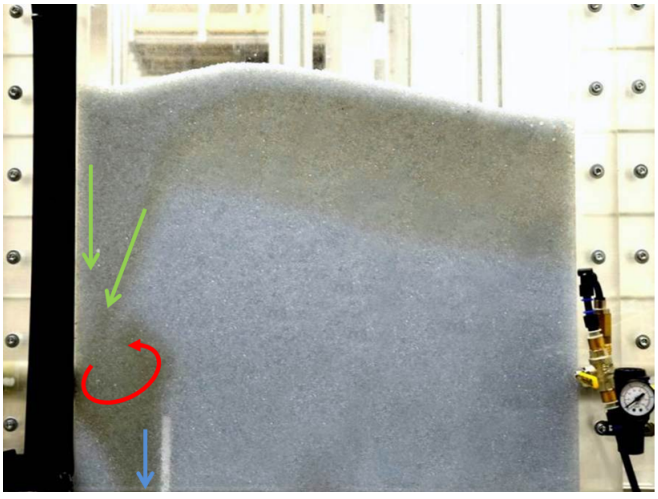


Fig. 7.6: Global bed movement. The raceway (red arrow) "aspires" material from regions close to the wall (green arrows), extraction beneath the raceway (blue arrow).

It turned out that only a rather narrow particle layer close to the left wall moved down-

wards and entered the raceway. The granular material in the centre of the system remained almost unaffected. A further investigation of the bed movement with particle image velocimetry (PIV) reinforced that observation. The vector field in Fig. 7.7 shows that only a small part of the bed participates in the downwards motion.

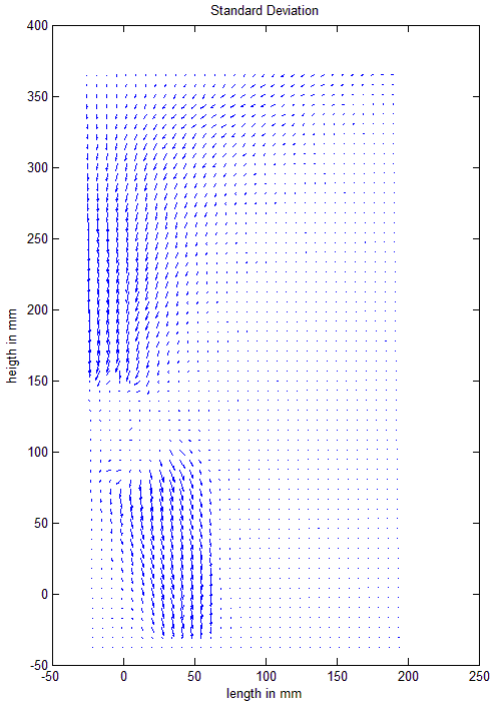


Fig. 7.7: PIV-measurement of the global bed movement.

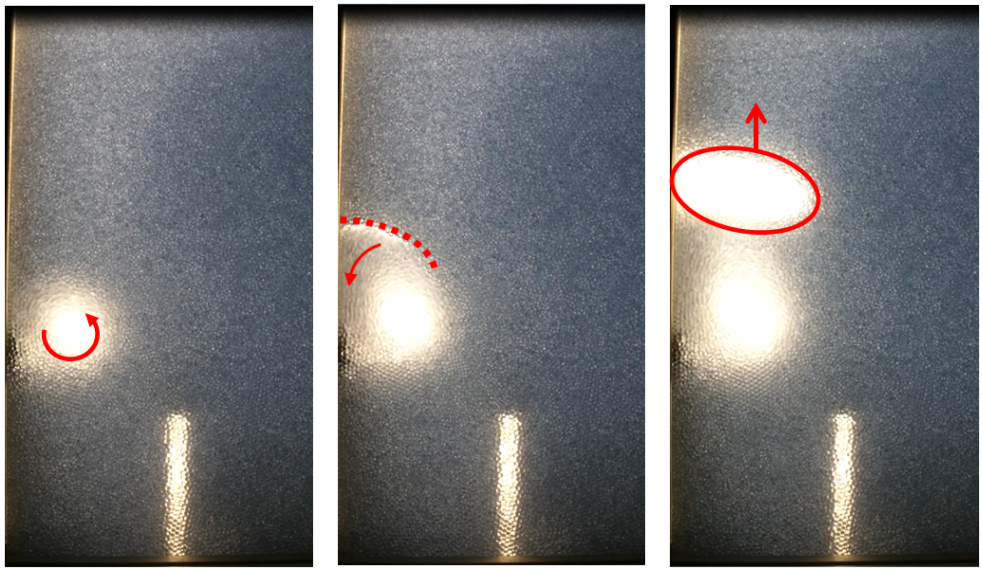
This result was rather surprising and suggested a strong 2D effect of the acrylic glass walls.

Operational states

During the experiments, different states could be observed. Fig. 7.8 shows the behaviour of a system at different points of time. A typical process of raceway formation, dynamics and breakdown is described in the following:

As soon as air enters the system, a small cavity forms. The rising gas leads to a slight reduction of the void fraction in the whole area above the tuyere and thus an interlocking of the particles. The momentum of the penetrating jet causes the breakup of a bigger cavity - the actual raceway (cf., Fig. 7.8a). Inside it a number of particles circulate. The raceway grows in vertical direction. Above the raceway another hollow space forms (cf., Fig. 7.8b). Until now the momentum of the blast air sufficed to support the upper wall. Yet, at some point the stabilizing force is too weak, and the hole breaks down. At the same time an air bubble starts rising towards the surface of the bed (cf., Fig. 7.8c).

The states described above do not appear equal in all cases: While for example for rather small inlet velocities and only few or no material extraction a rather steady raceway forms,



(a) (b) (c)

Fig. 7.8: (a) breakup of the raceway, particles rotate (b) formation of a second cavity (c) rise of a bubble.

high inlet velocities can cause a pulsating behaviour.

Comparison to results from the literature

In the literature a number of publications can be found that deal with similar experiments (e.g., Sarkar et al. [2007]; Gupta et al. [2005]; Sastry et al. [2003]). Even though the test rig and the material parameters are not identical, a qualitative comparison for demonstrating the plausibility of the measurement results was possible. In Fig. 7.9 the equivalent raceway diameter in two experiments is compared to results published by Gupta et al. [2005]. The boundary conditions were chosen according to the TEST2-configuration, the red arrows indicate whether the diameters were observed during the increasing of decreasing phase.

Discussion of the results from the different TEST series

A large number of test runs with the settings described beforehand were conducted. In the following an overview over the outcomes of TEST1 - TEST4 (cf., Sec. 7.2.1.1) will be given.

TEST1 Fig. 7.10, 7.11 and 7.12 contain a collection of the data recorded at the different test runs.

While the diagrams on the left hand side (Fig. 7.10a, 7.11a and 7.12a) show the inlet velocities at the times $t = 0 - 5$, the figures on the right hand side contain the corresponding information about the equivalent raceway diameter.

Especially for larger tuyere diameters the first increase of the inlet velocity influenced the equivalent raceway diameter remarkably. Afterwards, variations of the inlet velocity by about 20% showed almost no short-term impact on the raceway size. One exception can be

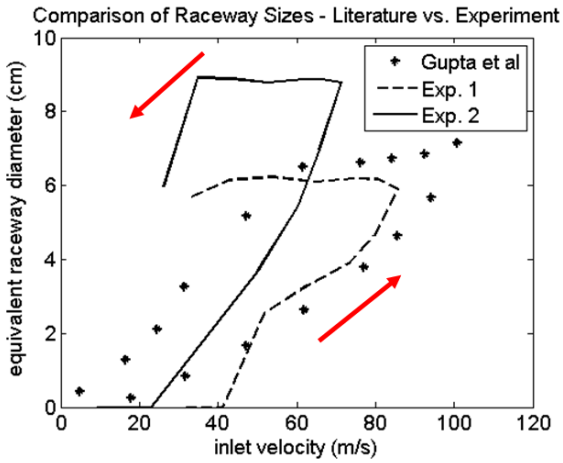
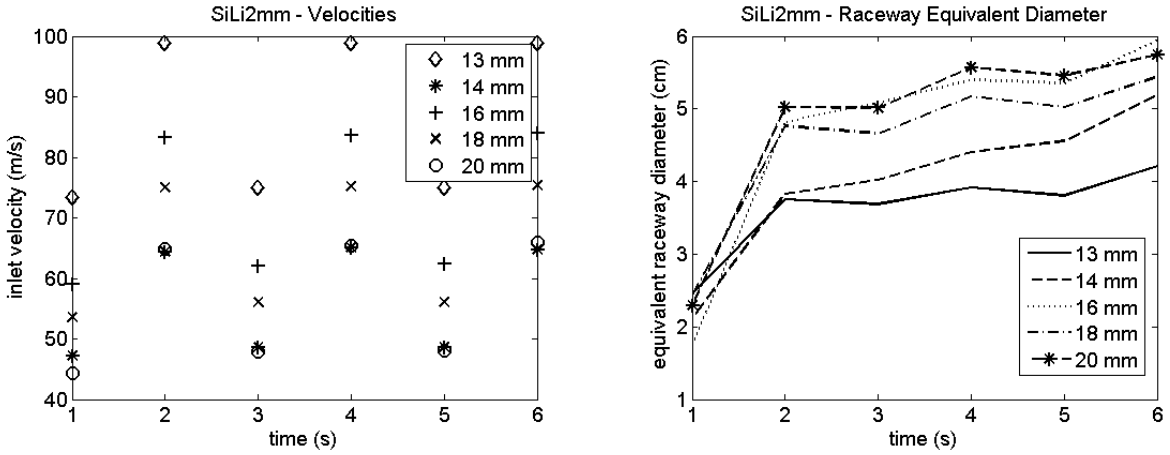


Fig. 7.9: Qualitative comparison of raceway size measurements - Gupta et al. [2005], Exp. 1: tuyere diameter 16 mm, Exp. 2: tuyere diameter 18mm. material: polypropylene beads (Gupta), Rounded4mm (Exp. 1 and 2, material properties acc. to Tab. 7.1).



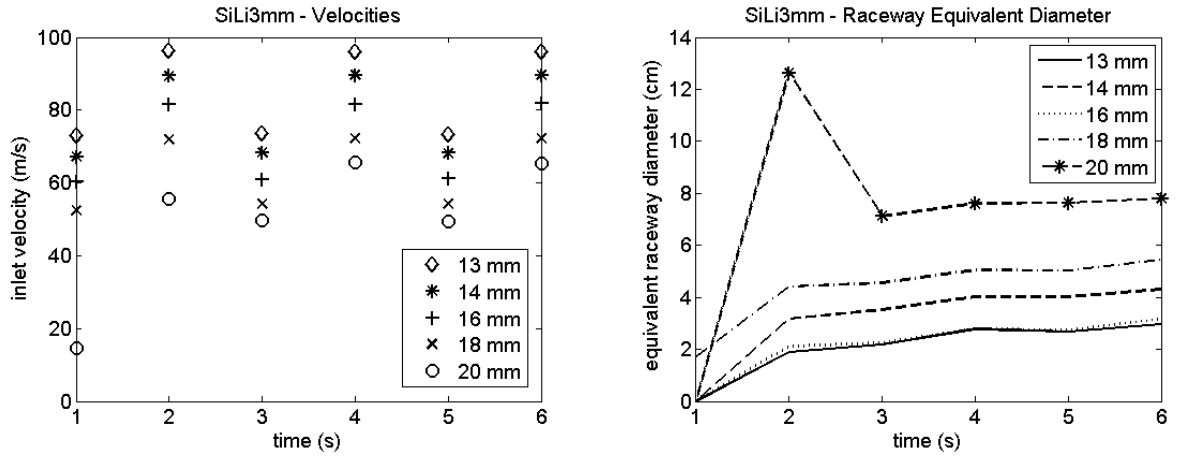
(a) *(b)*

Fig. 7.10: Results from the hystersis investigations (TEST1, SiLi2mm).

seen in Fig. 7.11b, where the results for a tuyere with a diameter of 20 mm are illustrated. The large initial size and the following decrease can be explained by the low inlet velocity in the first place (cf., Fig. 7.11a): the sudden increase provoked an over-proportionally large raceway that stabilized as soon as the inlet velocity was reduced again.

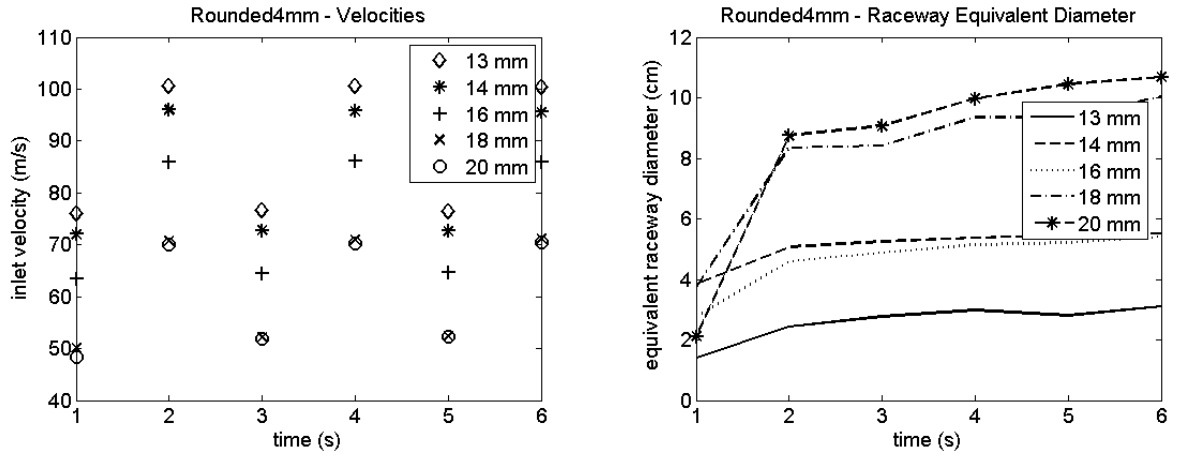
TEST2 The inlet velocity profile used in this case was symmetric with respect to time (cf., Fig 7.13a), but not so the measured equivalent raceway diameter (cf., Fig. 7.13b): again, the size of the raceway depended on its history, and a strong hysteresis showed. The presented example contains the results for 2mm glass beads in combination with a nozzle of 16 mm in diameter.

Fig. 7.14 shows a comparison of the equivalent raceway diameters for different tuyere diameters. At equivalent flow rates in the increasing case, similar raceway sizes can be



(a) *(b)*

Fig. 7.11: Results from the hystersis investigations (TEST1, SiLi3mm).



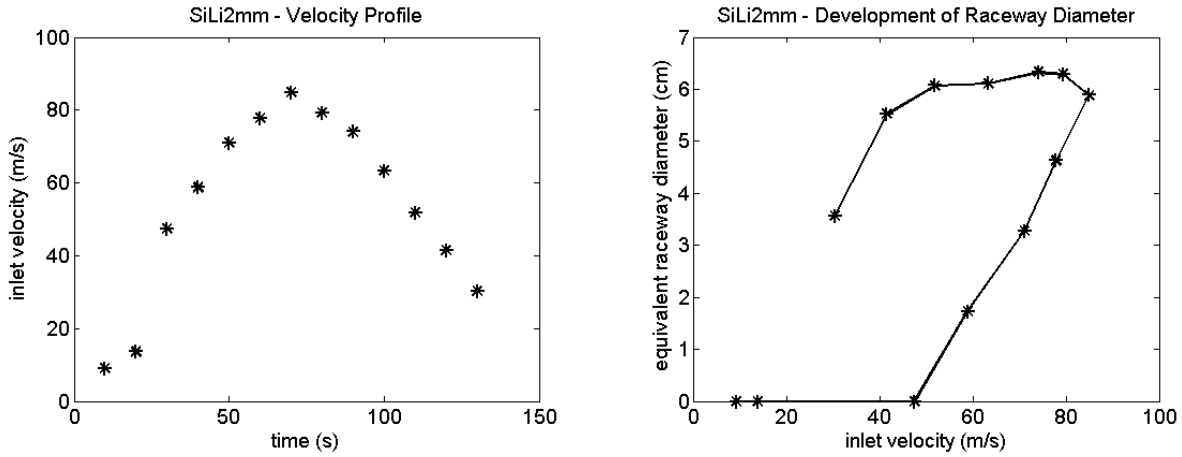
(a) *(b)*

Fig. 7.12: Results from the hystersis investigations (TEST1, Rounded4mm).

observed, especially for the material Rounded4mm. Due to the reduced pressure drop, higher flow rates could be obtained with larger tuyere diameters, thus the maximal raceway sizes were larger for larger tuyere diameters.

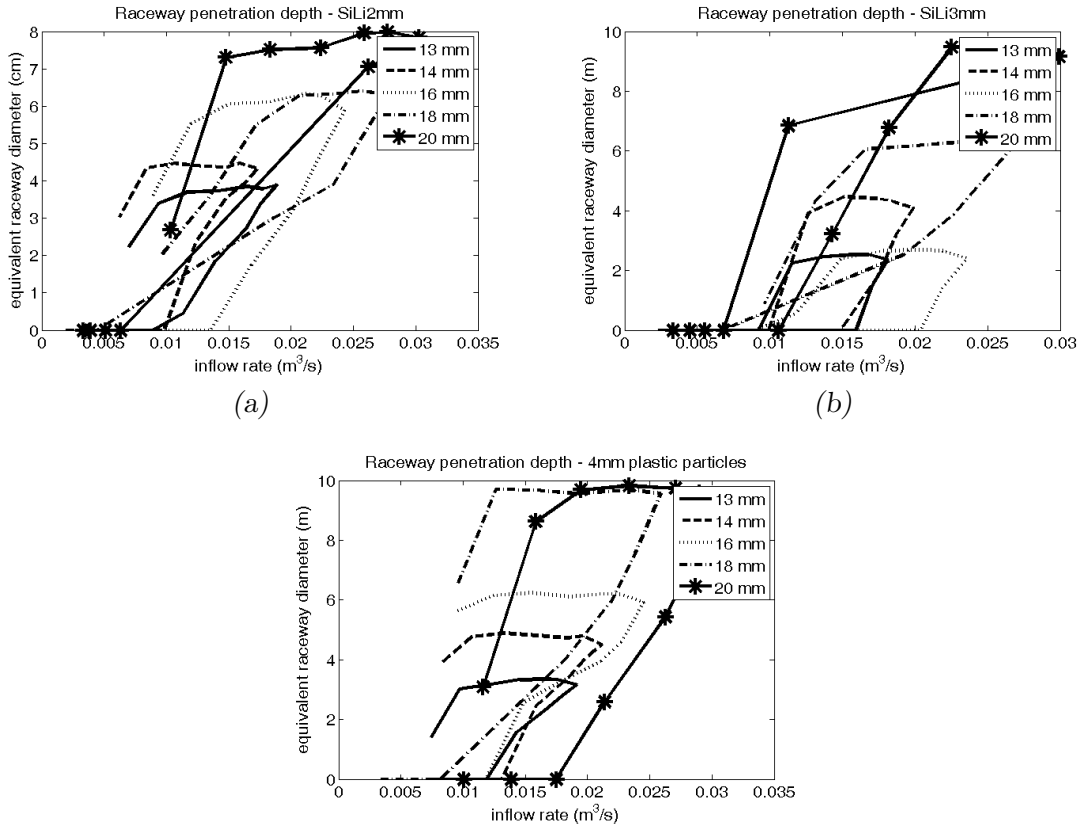
TEST3 The goal of this series was to determine the time after which the raceway reaches a steady state. For most cases a steady raceway was reached after 60-80 seconds. Fig. 7.15a indicates that for a tuyere of 16 mm in diameter, stable raceways formed at the latest after 80 seconds. For the combination of a large tuyere diameter and a relatively low mass flow rate an unstable raceway was obtained, as illustrated by Fig. 7.15b. In the latter case the momentum of the air jet did not suffice for obtaining a stable raceway.

TEST4 This investigation focused on the influence of the material extraction onto the raceway size and behaviour. In Fig. 7.16 cases with active and inactive conveyor are compared



(a) *(b)*

Fig. 7.13: Equivalent raceway diameter for increasing and decreasing velocity. (a) velocity profile (m/s) (b) raceway area(m^2).



(a) *(b)*

(c)

Fig. 7.14: Raceway penetration depth for 2 and 3 mm glass spheres and rounded plastic cylinders of an equivalent diameter of 4 mm. Comparison of the results for tuyere diameters between 13 and 20 mm.

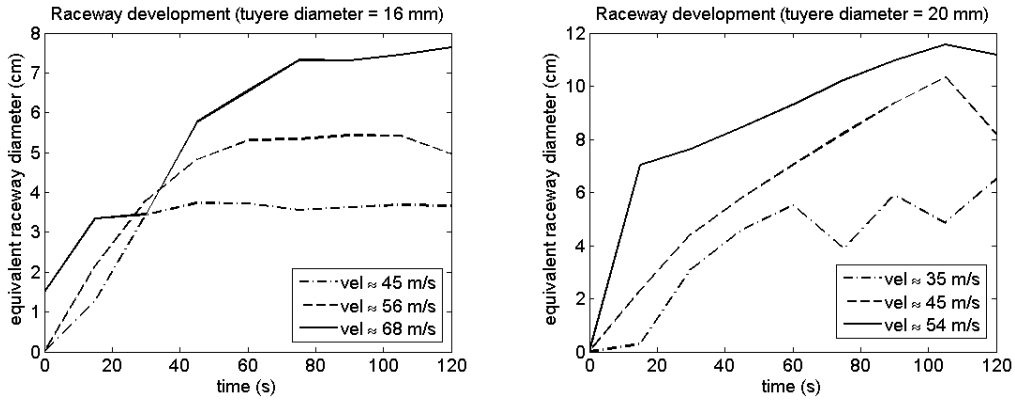


Fig. 7.15: Development of the raceway over time while granular material (SiLi3mm) is extracted. (a) Constant raceway after 10-60 s (b) Unstable raceway for high tuyere diameters.

against each other. In the cases where material was extracted the hysteresis was by far less pronounced than in the others. For the spherical glass particles this effect was even stronger than for the slightly non-spherical plastic cylinders.

In a separate study different extraction rates were considered. In Fig. 7.17 one can see that the hysteresis reduced for increasing material outflow velocities, while at the same time the raceway area became larger.

In conclusion it can be stated that the hysteresis was caused by two main factors: on the one hand the inter-locking of the particles within the bed and on the other hand the stabilization of the bed by the walls. By the continuous material flow in TEST4 the bed-stresses were reduced and both the hysteresis was reduced and the raceways became larger. For non-spherical particles (Rounded4mm) the interlocking effect of the particles was pronounced stronger than for spherical particles, thus the hysteresis reduced slower.

7.2.2 Simulations

In the first part of this section the necessary conditions for the simulations are presented. Afterwards the setup and the results are discussed.

7.2.2.1 Determination of the simulation parameters

Generally one can distinguish between two classes of parameters: those that can be measured directly and model parameters that are important for accurate simulations. The first group comprises

- particle diameter or grain size distribution,
- density and
- void or volume fraction.

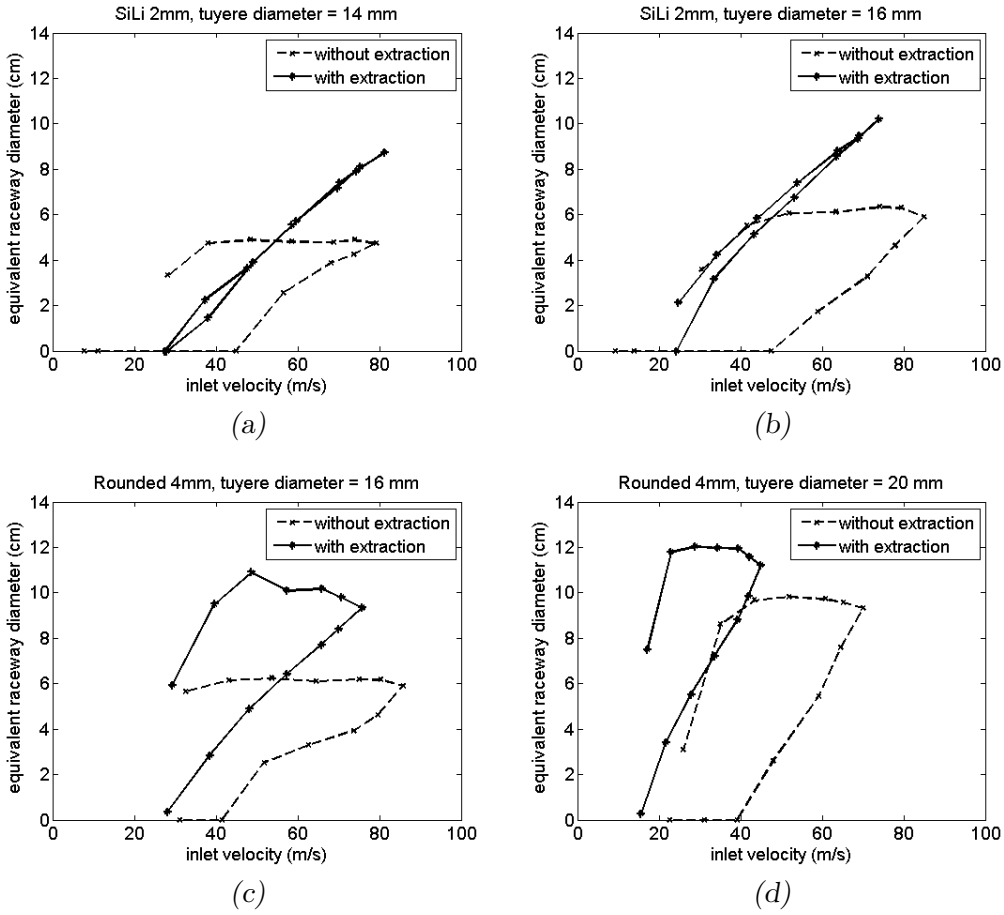


Fig. 7.16: Comparison of TEST2 and TEST4 for 14 and 16 mm (SiLi2mm) and 16 and 20 mm (Rounded4mm) tuyere diameter.

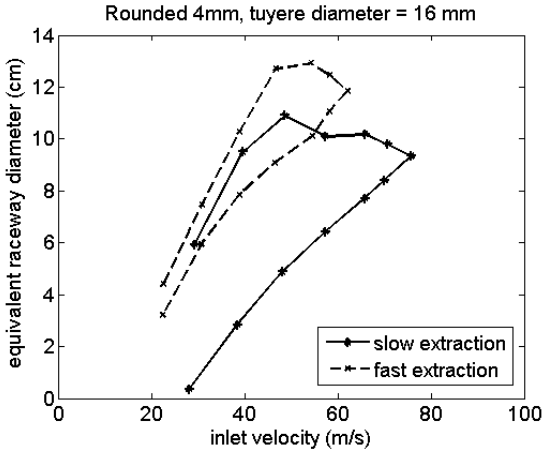


Fig. 7.17: Comparison of the raceway equivalent diameter for different material extraction rates.

Additional quantities like the viscosity of a fluid under certain conditions can be looked up in a reference work such as for example VDI-Wärmeatlas (cf., Verein Deutscher Ingeneure

[2010]). The group of model parameters deals with properties that describe the physical behaviour of the material. These quantities are usually given as a set of coefficients, for example the

- coefficient of friction,
- the coefficient of rolling friction or
- the coefficient of restitution.

These parameters can be deduced from observed material behaviour. In the present case a set of well-established characterizing experiments was conducted, where each of the experiments mainly depended on only one of the parameters.

Inclined plate

In this experiment a small number of particles is placed on a horizontal plate. The plate can be inclined with a crank. Starting from a certain angle of inclination the particles either start to roll or to slide down the plate, depending on whether sliding or rolling friction between the particles and the plate is stronger. Fig. 7.18 shows a schematic sketch of the setting.

Fig. 7.18: Few particles are placed on a plate, which is inclined until the particles start rolling or sliding.

Especially for materials consisting of (slightly) non-spherical particles the particle movement starts within a certain range of angles. For the simulation an average angle is determined and used as "target angle". In the simulation a single sphere is placed on a plate. The target angle is used as input parameter, a loop over the coefficient of rolling friction helps to determine the according parameter. In the simulation a constant directional rolling friction model is used.

The experiments for the rounded 4mm plastic particles yielded an average angle of 11 degrees where the particles start moving. For this input value the according coefficient of rolling friction was determined as 0.2.

Angle of repose

If granular matter is poured onto an even surface the resulting pile shows an angle of repose that is characteristic for this specific material. This angle is directly related to the rolling friction of the particles amongst each other.

In the presented experiment a hollow steel cylinder is placed on top of an elevated circular disk and filled with the particles. The hollow cylinder is lifted carefully, and the material cone forms.

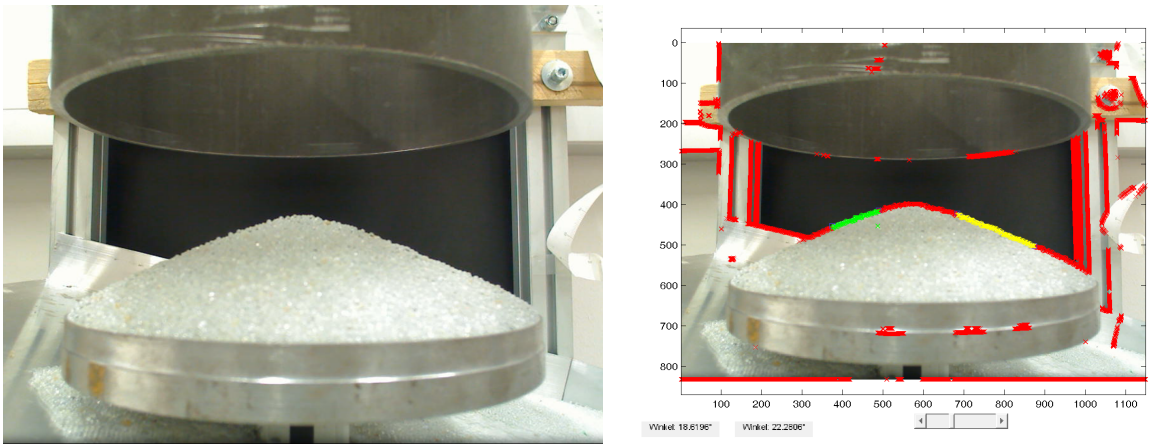


Fig. 7.19: Angle of repose - experiment and processed image.

Fig. 7.19 is an image of the angle of repose that forms when using glass spheres with a diameter of 2 mm. Fig. 7.19b illustrates the post processing of the experimental data. The experiments were repeated for a couple of times, the resulting average angle of repose was then used as target angle for the simulations.

For directly comparing the results of experiment and simulation, the data from the simulation were post-processed in the same manner (cf. Fig. 7.20).

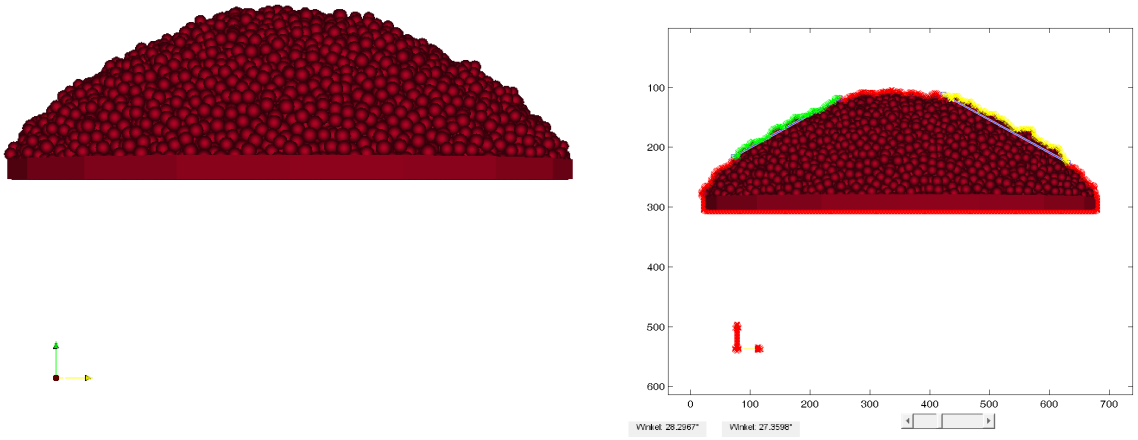


Fig. 7.20: Angle of repose - simulation and processed image.

Table 7.4 contains the measured angles for different materials.

name	Sili2mm	Sili3mm	Rounded
angle (°)	20.8	20.3	29.2

Tab. 7.4: Angle of repose for different materials.

Jenike shear cell

The inner friction of granular material can be determined with a shear cell. The one used in the present investigations resembles the design of the shear tester proposed by Jenike [1964]: the bottom of the rig is formed by a fixed hollow cylinder. On top of it a movable hollow cylinder is placed. The lower hollow cylinder contains bulk solid that also fills the movable hollow cylinder to a certain degree. A weight placed on top of the granular material exerts a normal force, which causes a compressive stress. The movable hollow cylinder is now pulled away in tangential direction. The force required for moving the cylinder is measured with a load cell. Fig. 7.21 gives a rough idea of the concept. From the measured force a starting value for the coefficient of friction can be guessed.

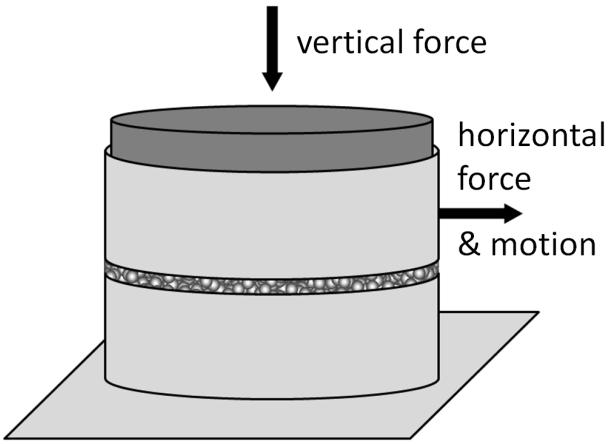


Fig. 7.21: Shear cell for measuring the inner friction of bulk material.

In the simulation a number of coefficients of friction are tested, until the value for which the forces in simulation and experiment match is found. Fig. 7.22 contains the result of a shear cell experiment and the according simulation for the rounded 4mm polypropylene cylinders.

Hopper outflow

This experiment serves the purpose of validating the previously determined parameters. A certain amount of granular matter is filled in an acrylic glass hopper. Different apertures can be mounted at the bottom of the device leading to different outflow rates.

In Fig. 7.24 the hopper outflow rates from simulation and experiment are compared. The material used are the rounded plastic cylinders of an average diameter of 4mm (cf., Tab. 7.1). The mass flow rates are in good accordance for the whole range of orifice diameters tested.

The described set of experiments and the according DEM simulations were developed at the Christian Doppler Laboratory for Particulate Flow modelling. For the coefficient of restitution we relied on values from reference works and former investigations.

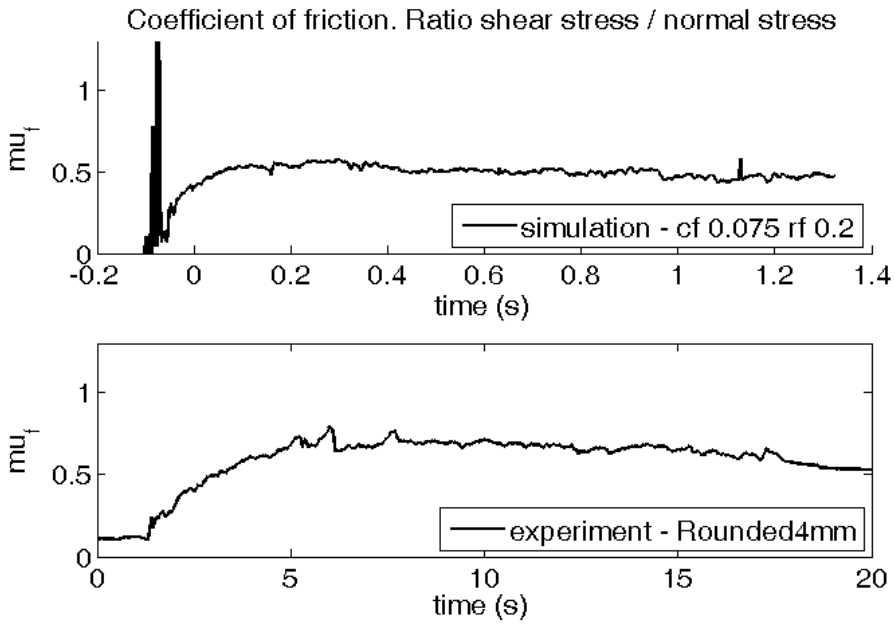


Fig. 7.22: Simulated (top) and measured (bottom) coefficient of friction for shear cell setup (material: Rounded4mm).

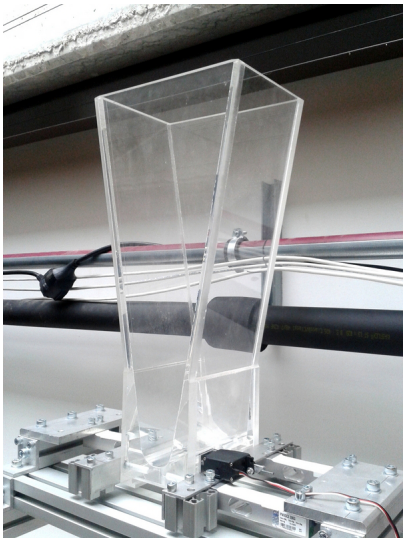


Fig. 7.23: Hopper for the outflow experiments.

7.2.2.2 Case setup

The goal of the investigation was to generate a coupled CFD-DEM tool for simulating the presented experimental setup. Generally a CFD-DEM case consists of two distinct folders. One contains the files required for the CFD calculation and the other one the LIGGGHTS input scripts and further material required for the DEM calculation. The basic structure of a coupled case, here termed Raceway case, is presented in Fig. 7.25 and will be described in the following.

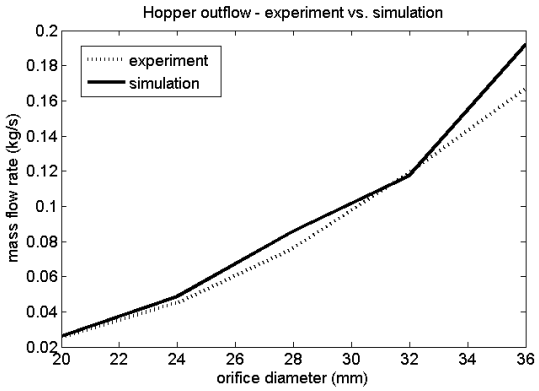


Fig. 7.24: Comparison of experimental data and simulation results (material: Rounded4mm).

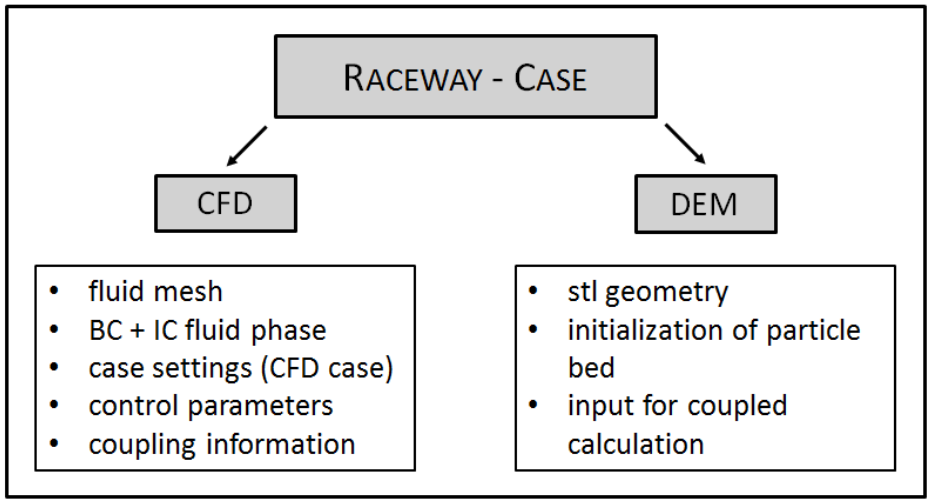


Fig. 7.25: Outline of a coupled CFD-DEM case.

Calculation of the fluid phase

The CFD-folder contains the fluid mesh and files determining initial and boundary conditions for the fluid fields. Furthermore case-control and coupling settings are made here. The input data is organized in the folders `0`, `constant` and `system`.

- **The mesh** : in the presented cases the basic geometry was created with Gambit, a commercial mesher that can be used to generate Fluent-meshes. With the command `fluentMeshToFoam` they can be converted to OpenFoam-meshes.

  For a successful application of the unresolved method it is crucial to work with a mesh whose cells are larger than the contained particles. For the raceway case this requirement lead to a relatively coarse mesh of $\approx$ 35,000 cells (approx. 0.006 $\times$ 0.006 $\times$ 0.006 m^3).

  In the experimental setup the velocity inlet consisted of a round tuyere, which was represented by a square within the mesh. This slight change led to a better mesh quality and a better stability behaviour. The results remained practically unaffected

by this small change as the areas of the round and the square inlet coincided. For each tuyere diameter in the experiment an according mesh was created. Fig. 7.26 shows a 2D-projection of the mesh, numbers mark the different boundary regions. The dimensions of the domain are 0.3 m × 0.05 m × 0.65 m, the gap represents the acrylic glass plate in a distance of 0.1 m from the left wall.

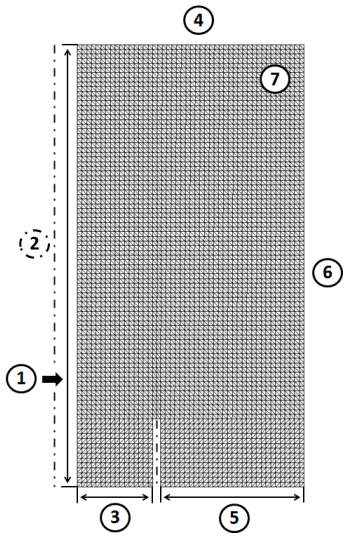


Fig. 7.26: Boundary fields: 1 ... inlet, 2 ... wall (4 faces, as the small acrylic plate is included), 3 ... sink, 4 ... top, 5 ... bottom, 6 ... internal, 7 ... sides (2 faces, front and back).

- **Initial and boundary conditions**: In the experiments the air entered the domain through the tuyere. In the simulations a velocity inlet was placed at this position. The inlet velocities were prescribed with data files, the profiles were chosen according to the experimental settings. A pressure outlet was installed at the top of the rig, being responsible for the discharge of the air. At the remaining walls a no-slip condition was imposed. For $t = 0$ the container was assumed to be filled with fluid only. Given that the particle bed had been initialized with a pure DEM calculation (see below) this false assumption was corrected at the first time step. Tab. 7.6 contains a complete list of the boundary settings for the different quantities. Further information such as the prescription of a kinematic viscosity, the gravity and turbulence properties was provided as well (cf., Tab. 7.7).

- **Case control**: As customary for CFD-cases that are calculated with OpenFOAM based solvers, the case control is taken over by the `controlDict`. Starting time, time step width and end time are defined in there. Furthermore it is used to specify the times, at which data is written to files. In `fvSolution` and `fvSchemes` the discretization schemes and numerical solvers for the different equations are specified. In the presented case the standard settings were chosen.

- **Coupling**: As the CFD-solver is the master process that calls the DEM solver, it also needs access to a script with the settings for the coupled calculation. In `couplingProperties` one can choose between a number of models (cf., Sec. 6.1) and

configure their settings. As discussed in Sec. 5, the literature provides different sets of governing equations for describing the problem. The `modelType` keyword allows to choose between "model A" and "model B" (see Section 5.1). For all calculations "model A" was applied. Tab. 7.5 contains a list of all models that were used for the calculations.

model name	model
`modelType`	`A`
`couplingInterval`	`10`
`locateModel`	`engine`
`momCoupleModels`	`implicitCouple`
`turbulenceModelType`	`RASProperties`
`IOModel`	`basicIO`
`meshMotionModel`	`noMeshMotion`
`regionModel`	`allRegion`
`dataExchangeModel`	`twoWayMPI`
`averagingModel`	`dense`
`clockModel`	`standardClock`
`smoothingModel`	`off`

`forceModels`	`KochHillDrag`
	`gradPForce`
	`viscForce`

Tab. 7.5: Properties for the coupled case.

The `couplingInterval` defines the number of DEM-steps, after which a coupling with the fluid calculation is established. This has to be taken into account when the CFD and LIGGGHTS time step widths are chosen.

Calculation of the particulate phase

All files required for the Lagrangian calculations are collected in the DEM-folder. LIGGGHTS permits the incorporation of stl-files as granular walls. A straight forward method for generating them from an existing mesh is by applying the OpenFOAM-command `surfaceMeshTriangulate`. The core of the DEM case consists of one or more input scripts. In the raceway-investigations a three-step DEM computation was carried out (cf. Fig. 7.27), which comprises the following three steps:

- **Initialization**: the solid bed was initialized with an uncoupled LIGGGHTS calculation. The particles were generated in a domain that was 3-4 times higher than the final bed. The inserted particles were slightly poly disperse for avoiding a non-physical packing behaviour.

  After the completion of the settling process a restart file was written. Restart files contain all particle data from the last time step.

	U	Us	p	rho	void fraction	Ksl
inlet	cf. Tab. D.1, Appendix	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
wall	fixedValue, (0 0 0)	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
sink	fixedValue, (0 0 0)	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
top	zeroGradient	zeroGradient	fixedValue, 1.0e5	zeroGradient	zeroGradient	zeroGradient
bottom	fixedValue (0 0 0)	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
internal	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
sides	fixedValue (0 0 0)	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
internal field	(0 0 0.1)	(0 0 0)	1.0e5	1.1	1	0

Tab. 7.6: Boundary conditions for the CFD-case.

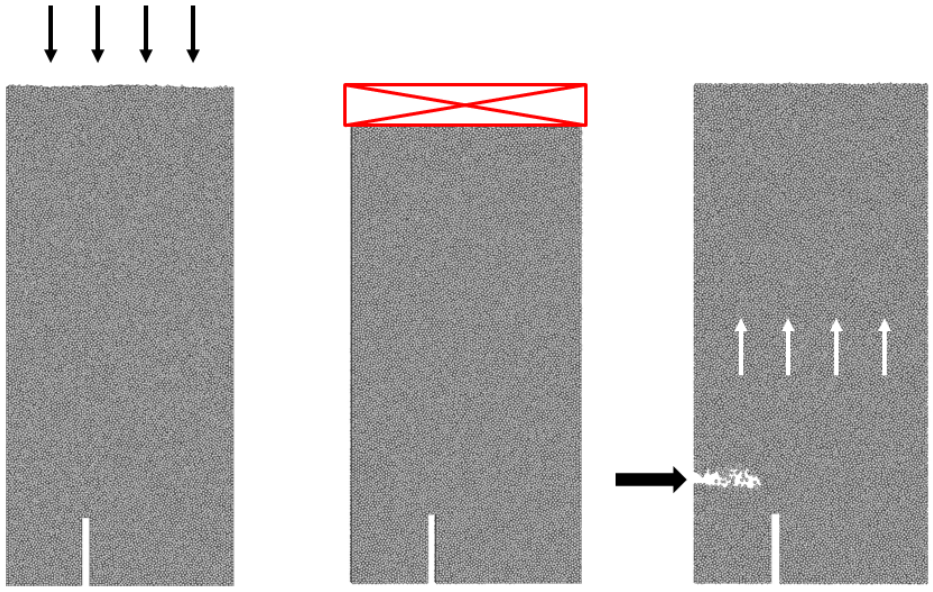


Fig. 7.27: The DEM calculation is carried out in three steps: initialization, adjustment of the bed and bed movement due to the impact of the air jet.

- **Adjusting the bed**: as the particles were allowed to settle in an uncontrolled manner, the surface of the packing became uneven. For guaranteeing identical starting conditions for all cases a delete command was used to remove the particles above the desired filling height.

- **Coupled input**: This input file is called by the CFD solver during the coupled calculation. During the case studies this file underwent a number of modifications. Initially the base geometry consisted of a rigid container. As will be discussed later on (cf. Sec. 7.2.2.3), this assumption was not suitable for depicting the behaviour of the experiment accordingly. Therefore the two big plates - forming the front and the back of the box - were uncoupled from the remaining geometry and replaced by **servo walls** (cf., e.g., Aigner et al. [2013a,b]). The servo walls have the capability of exerting a force in a certain direction. With their aid the interaction between granular material and the deformable acrylic glass walls in the experiment could be represented. Fig. 7.28 illustrates the functionality of the servo walls in case of the raceway simulations.

  Furthermore the file was designed in a way that permitted material extraction, both in the actual raceway region, and in the region where the material was extracted during the experiments. More detailed discussions on the different options can be found in Sec. 7.2.2.3.

  The coupling was indicated by two fix commands, namely:

```
fix cfd all couple/cfd couple every 10 mpi
fix cfd2 all couple/cfd/force
```

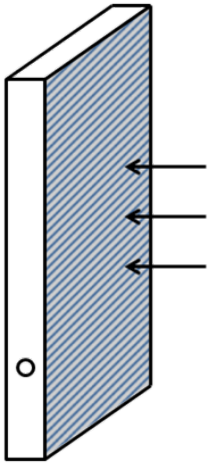


Fig. 7.28: The side walls are replaced by servo walls that exert a pressure onto the particle bed.

Parallel computation

CFDEMcoupling offers the capability to run cases in parallel. For doing so, the CFD case needs to be decomposed. Decomposition method as well as the number of subdomains are defined in the dictionary `decomposeParDict`. OpenFOAM provides a number of different decomposition methods. As the used geometry is rather close to a cuboid, it sufficed to apply the "simple" method. LIGGGHTS only needs one additional line in the input script for recognizing a case as parallel:

```
processors noXProc noYProc noZProc,
```

with `noXProc`, `noXProc`and `noXProc`being the number of processors in the respective direction. Even though the numbers of processors for the CFD and the DEM case have to coincide, the composition of the domains can differ. This is especially interesting when it comes to cases where load balancing is advantageous. Load balancing is a capability that is available in LIGGGHTS and that moves processor boundaries according to the number of particles located in a certain area. It increases the efficiency of the parallelization significantly.

7.2.2.3 Results

Based on the data gained by the experiments, simulations were run. The boundary conditions were chosen according to the test cases described in Sec. 7.2.1.1. All simulations were carried out for the 4 mm rounded polypropylene cylinders. A total of 165.000 particles were necessary to fill the box. The material parameters used are summarized up in Tab. 7.7.

For the determination of the coefficient of friction and rolling friction the combined experimental and simulation routine discussed in Sec. 7.2.2.1 was used. 0.8 is a common value for the coefficient of restitution for polypropylene. The Young's modulus indicates the material stiffness. In comparison to the values given in the literature, the one used in the presented simulations is rather low. Underestimating the Young's modulus is common in

Fluid	name: fluid
	density: 1 kg/m^3
	viscosity: $1.8 \cdot 10^{-5}$ m^2/s
Granular material	name: Rounded4mm
	particle density: 878 kg
	Young's modulus: $5 \cdot 10^6$
	Poisson ratio: 0.45
	coefficient of restitution: 0.8
	coefficient of friction: 0.075
	coefficient of rolling friction: 0.2

Tab. 7.7: Properties of fluid and solid in the coupled simulations.

DEM calculations: large moduli require a time resolution that is over-proportionally fine, where at the same time they have no strong impact on the behaviour of the whole system. The parameter check with the hopper example yielded a satisfying accordance between simulation results and experiments (cf., Fig. 7.24).

Rigid domain

In the first test setup the acrylic glass box was modelled by a rigid stl-box. During the initial runs no material was extracted and the velocity profile of the blast air was chosen according to the boundary conditions in TEST2. The results regarded more closely stem from the case with a tuyere diameter of 16 mm. First of all the developing raceways were compared directly. Fig. 7.29 displays the images of the systems in simulation and experiment for an inlet velocity of 74 m/s. Clearly the size of the raceway was strongly over-predicted in the simulation. The upper series of images was taken at a time, where the inlet velocity was still increasing while the lower series shows the raceways at decreasing gas velocity. In the simulations the raceway size is more or less in accordance while raceway in the experiments is almost twice the size for the decreasing velocity.

Both the over-prediction of the raceway-area and its different behaviour can be observed during the whole computation and at all velocities. Fig. 7.30 illustrates these two effects.

The strong hysteresis that was monitored in the experiments was by far less pronounced in the simulation. A further analysis of the results showed that while the raceway was rather stable in the experiments, it deformed during the simulations. Even at rather low inlet velocities the instabilities were big enough to cause bubbles rising upwards in the bed (cf., Fig. 7.31). In the experiments far higher inlet velocities were necessary to reproduce a similar behaviour.

The experiments showed that material extraction reduced the hysteresis. Therefore the following case was equipped with a material sink. Again the inlet velocity was chosen in accordance to the blast velocity, measured in the experiment. In Fig. 7.32 the equivalent raceway diameters occurring in experiment and simulation are compared for tuyere diameters of 16 and 18 mm.

In these cases the equivalent raceway diameters can be depicted a lot better than in the previous cases.

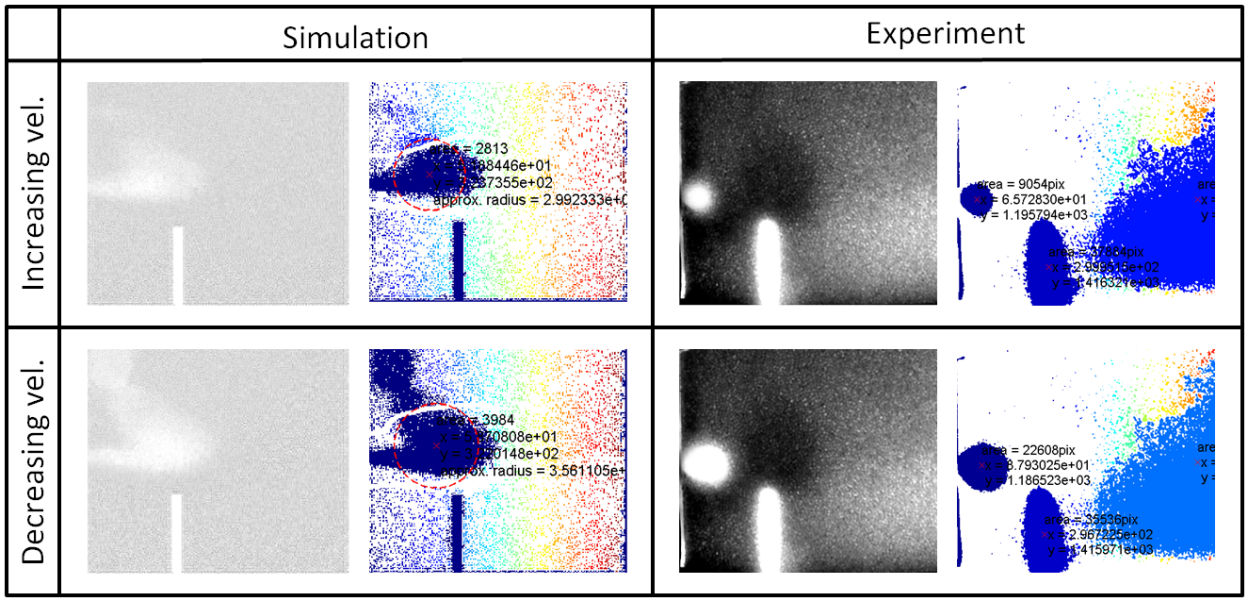


Fig. 7.29: Comparison of experiment and simulation for Rounded 4mm, tuyere diameter = 16 mm. Each column shows raw data on the left and processed images on the right.

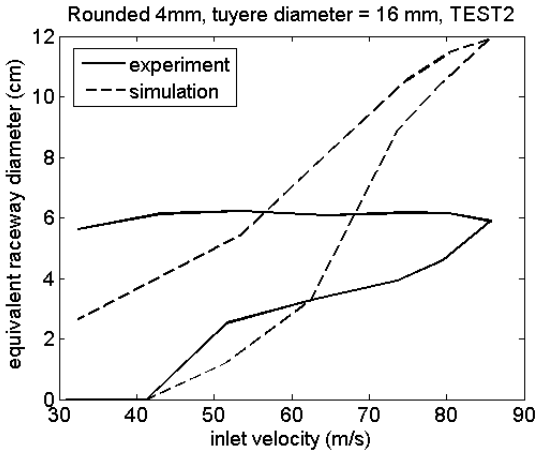


Fig. 7.30: Comparison of the equivalent raceway diameters for experiment and simulation of TEST2.

In the experiments one could observe that the hysteresis became less as soon as material was removed from the bottom of the container underneath the raceway. This removal led to a loosening of the bed, or, in other words, a reduction of the inner stresses. The fact that the hysteresis could not be depicted in the simulations so far thus indicated that the inner bed stresses were under-represented. A closer look at the test rig itself provided a rather simple explanation: even though the container was a cuboid when manufactured, its walls were subject to slight deformation when the box was filled with granular material. The two large side walls started acting like springs under tension, exerting pressure to the particle bed. In order to verify this assumption, the DEM geometry was modified.

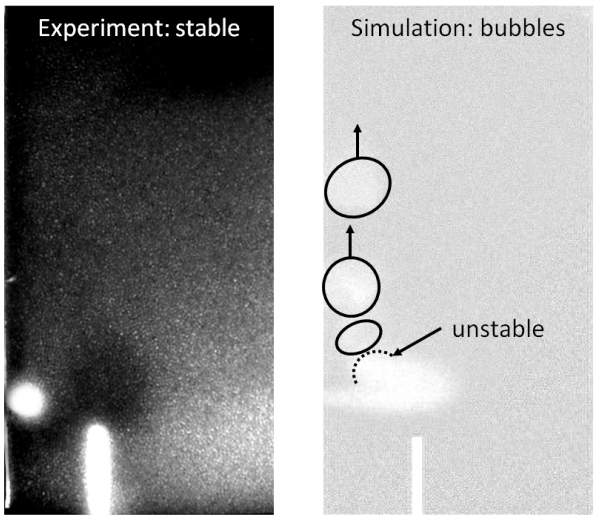


Fig. 7.31: Bed stability in the experiment and the simulation (Rounded 4mm, tuyere diameter 16mm, inlet velocity 80 m/s.

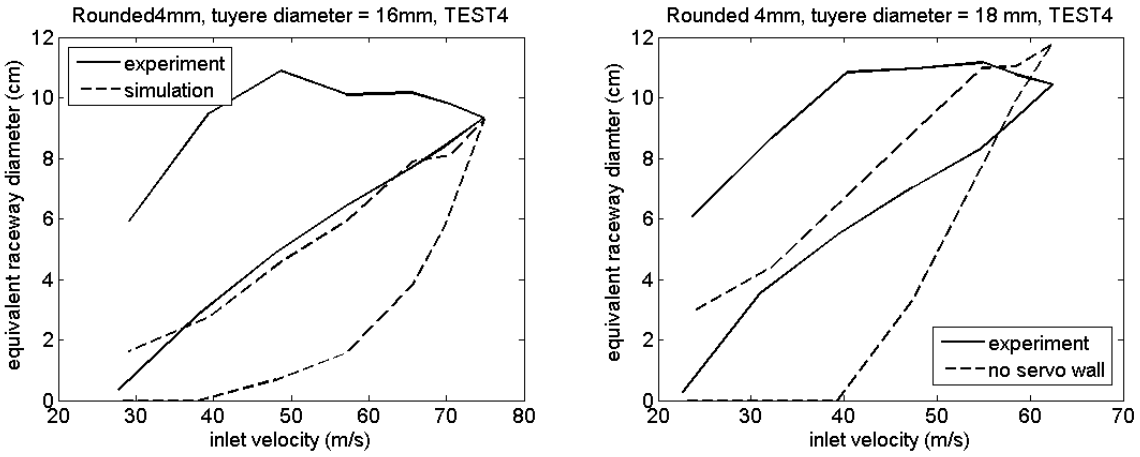


Fig. 7.32: Comparison of the equivalent raceway diameters for experiment and simulation of TEST4 (tuyere diameters 16 and 18mm).

Deformable domain

In order to model a deformation of the walls, the rigid stl-box needed to be changed: the side planes were detached from the remaining geometry and replaced by servo walls. The force, acting perpendicular to the particle bed, was chosen in such a way that the walls remained in their original position. First of all the calculation for the "TEST2"-case was repeated with this additional feature. The result is displayed in Fig. 7.33.

The result from the simulation with the newly introduced servo wall was in good accordance with the raceway diameters occurring in the experiments. Fig. 7.34 compares the raceway for a simulation with a rigid box against a simulation with the servo walls. In the first case the forming cavity was unstable, air bubbles ascended in the bed. Furthermore its expansion was strongly over predicted. The picture on the right hand side shows the

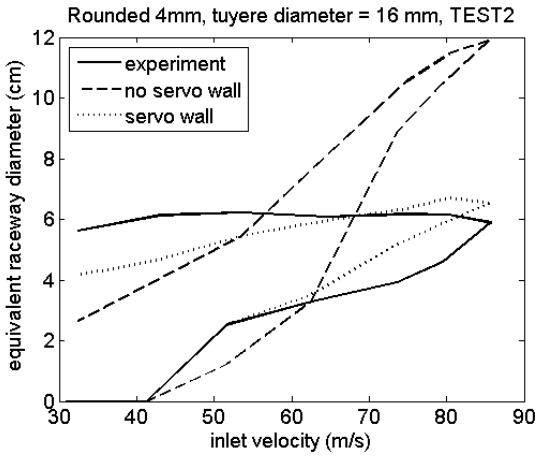


Fig. 7.33: Comparison of the equivalent raceway diameters for experiment and simulation with and without servo walls. Case: TEST2.

raceway that formed when the sides consisted of servo walls. It was stable in shape and its size coincided with the experimentally predicted size. In the presented case the boundary conditions were chosen according to TEST3 (inlet velocity 80 m/s).

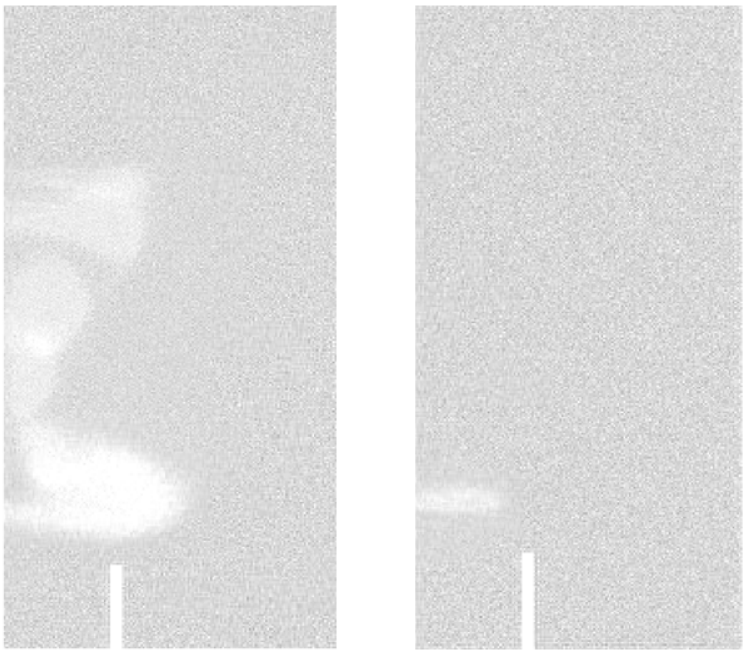


Fig. 7.34: Simulation of the raceway using a rigid box (left) and servo walls (right).(tuyere diameter = 16mm, inlet velocity 88 m/s, material: Rounded4mm).

In the next step the concept of the servo walls was used for simulating TEST4-cases. The results are given in Fig. 7.35.

The material loosening effect of the material extraction was reduced over-proportionally by the pressure exerted by the walls. So while the general behaviour, especially in terms of hysteresis, could be depicted nicely, the raceway was now under-predicted. Adjustments of

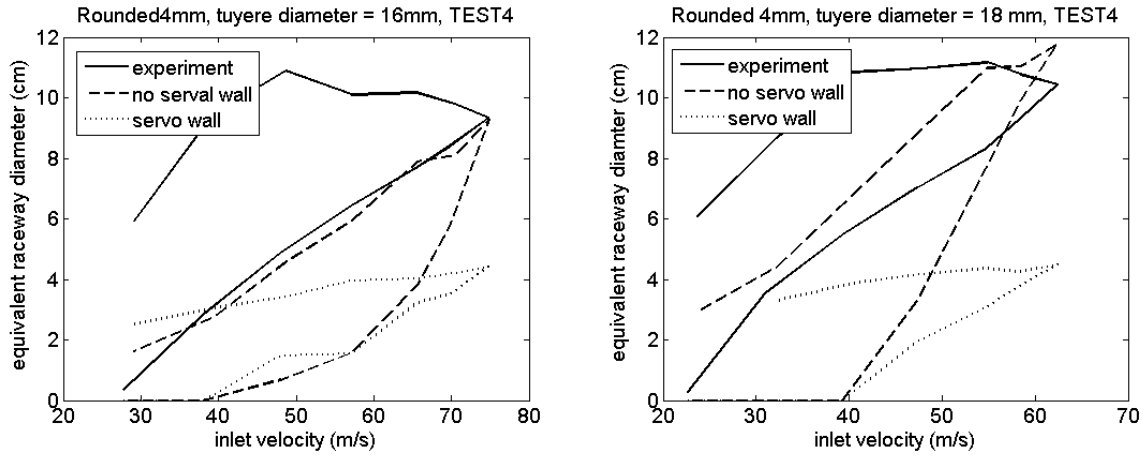


Fig. 7.35: Comparison of the equivalent raceway diameters for experiment and simulation with and without servo walls (TEST4, tuyere diameters 16 and 18mm).

the forces acting on the bed would have yielded improvement, but as the goal of this investigation was rather to verify the explanation given for the different behaviour in simulation and experiment than to get a perfect match of the data, further calculations were omitted.

Global bed movement in the simulation

When material is extracted from the bottom of a container, the particle bed starts moving downwards. In Sec. 7.2.1.2 the nature of the motion was discussed for the experiments. During those tests the material was extracted through an opening of the acrylic glass box in a region underneath the raceway. In a real-life blast furnaces most of the material is gasified in the raceway area itself. In the simulations it was easily possible to remove material from arbitrary regions in the domain.

The goal of this study was to compare the bed behaviour of experiment and simulation and to find out, whether extracting material underneath the raceway has the same effect as if removing it directly from the raceway. For this purpose two cases with different material removal areas were set up. The results are compared in Fig. 7.36, displayed side by side with the experiment.

It turned out that the different extraction regions did not influence the global bed motion. Furthermore it could be shown that the material displays the same overall behaviour for both experiment and simulation.

This investigation showed that the application of CFD-DEM simulations is suitable for depicting the effects found in the experiments. The results showed that the formation of the raceway size strongly depends on the pressure that is exerted by the walls of the test rig. This leads to the conclusion that the present results cannot be used to make assumptions concerning the general development of raceways in the 3D case.

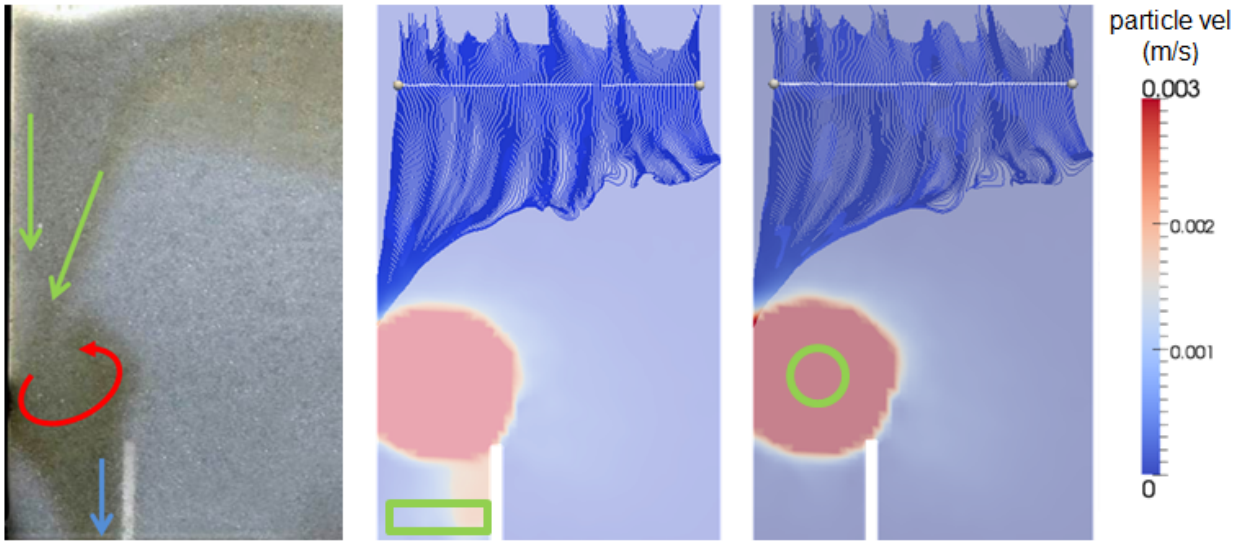


Fig. 7.36: Global movement of the particle bed due to material extraction. Left hand side: experiment, centre and right: particle velocity field and streamlines of the particles, starting at the given lines. Material extraction areas are marked with the green box / circle.

7.3 Simulation of a 3D wedge

The previous investigation showed that the pseudo 2D nature has a strong influence on the formation of the raceways. Consequently a correct depiction of the process requires a three dimensional model. In the following chapter such a model is presented and the results of the numerical calculations are discussed.

7.3.1 Case setting

The presented model is based on an operating blast furnace. The data was kindly provided by industrial partners of the Christian Doppler Laboratory on Particulate Flow Modelling.

7.3.1.1 Geometry

A blast furnace is an axial-symmetric system. Instead of simulating the whole device, it would be enough to consider a wedge only and apply periodic boundary conditions. At the time where the simulations were set up periodic boundary conditions were unavailable in CFDEMcoupling. As a workaround a larger wedge comprising three tuyeres was used and the internal walls were defined as frictionless. This ensured that the central raceway was unaffected by the presence of walls. In this investigation the area of interest was located at the raceway level. Therefore it was considered sufficient to calculate the dynamics of the system up to a certain height. The influences from the bed above this level were accounted for by a pressure exerting plate.

Just as in the pseudo 2D case the mesh was generated with the aid of Gambit. The resulting Fluent-mesh was the transformed with the command `fluentMeshToFoam`. The surface meshes for the DEM calculations were extracted from the OpenFOAM-mesh by

applying `surfaceMeshTriangulate`. The mesh consisted of a total of approx. 15,000 cells (approx. $0.1 \times 0.1 \times 0.1$ m^3) and is displayed in Fig. 7.37.

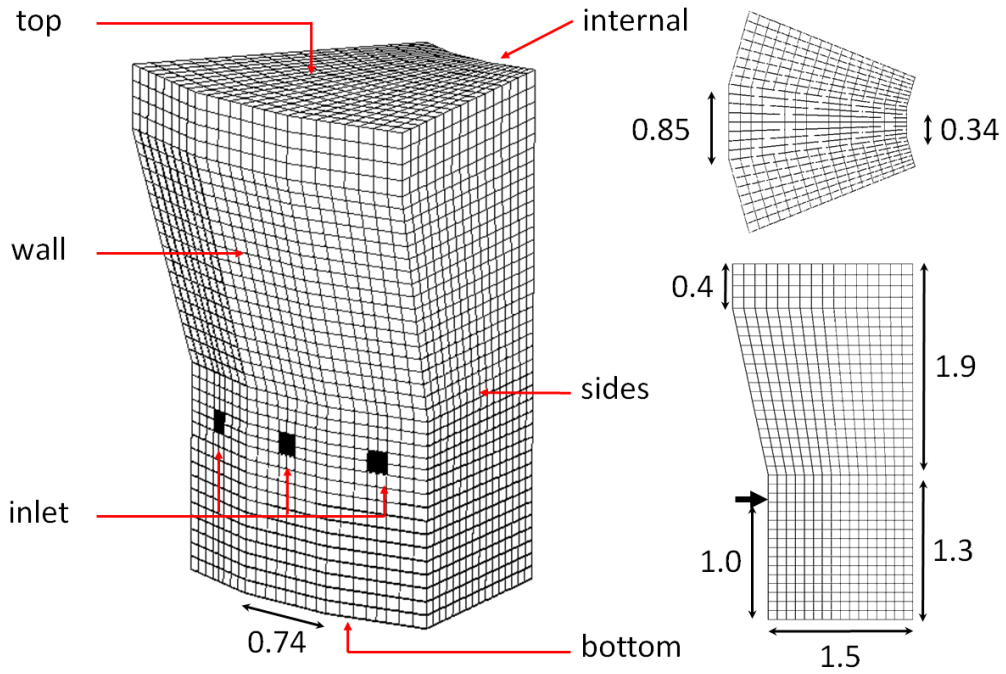


Fig. 7.37: Fluid mesh of the 3D wedge of a blast furnace.

7.3.1.2 Boundary conditions

Blast air entered the system through 3 tuyeres, thus the inlet now consisted of 3 distinct regions. In accordance with the provided data the inlet velocity was 100 m/s. The standard values of air at 20°C were used as fluid density and viscosity (1.1885 kg/m^3 and $1.5{\cdot}10^{-5}$ mr/s). At the external walls ("wall") a no-slip condition was applied, and in the DEM calculation they were subject to friction. The walls inside the furnace, termed "internal" and "sides" in Fig. 7.37, were regarded as frictionless and in the CFD calculation a slip-condition came to application. The top was defined as pressure outlet.

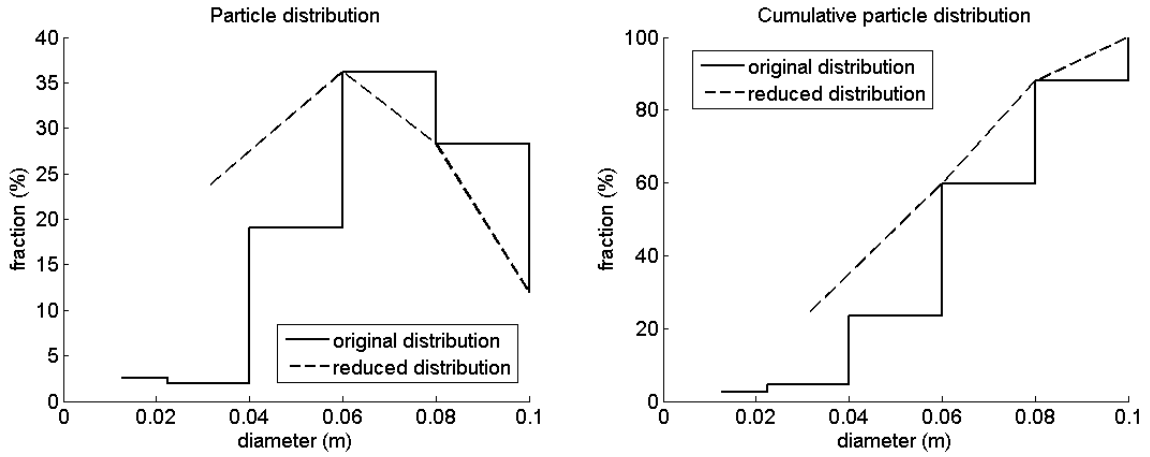
7.3.1.3 Material properties

The production of a ton of steel requires 1.6 tons of iron source and 360 kg of coke. The raw materials are charged from the top, and it can be assumed that bed in the upper region consists of that blend. Based on different data sheets the average bulk density in this region was determined as 1350 kg/m^3 (cf., Tab. 7.8). An important topic is the void fraction of the bed inside the blast furnace. The pressure onto the particle bed was determined with the aid of the Janssen equation (cf., e.g., Schulze [2006]).

As the iron source liquidizes on its way down the furnace, almost only coke particles reach tuyere-level. Therefore the bulk density of the particles in the simulation was chosen with 560

Material	Bulk density
1600 kg iron source	
- 650 kg sinter	1800 kg/m^3
- 600 kg pellets	2300 kg/m^3
- 200 kg lump ore	2850 kg/m^3
- 150 kg fines (s+p+lo)	2400 kg/m^3
360 kg coke	520 kg/m^3
Total bulk density	1350 kg/m^3

Tab. 7.8: Bulk densities for different materials.



(a) Grain size distribution *(b)* Cumulative particle distribution

Fig. 7.38: Particle distribution: measurement data vs. simplified distribution.

kg/m^3. No experimental void fraction data inside a blast furnace could be provided, thus the particle density was adjusted with the void fraction data of preliminary simulations, resulting in a density of 760 kg/m^3 for the spheres in the DEM calculation. Tab. 7.9 summarizes all boundary conditions of the case.

The particle size distribution used in the recent calculations was based on measurements conducted by one of our industrial partners. For numerical reasons and also for keeping the computational costs in an acceptable range, the fines were removed, resulting in a total of about 160,000 particles. In Fig. 7.38 the original particle distribution and the one used in the simulations are compared. The cumulative particle distribution in Fig. 7.38b indicates that the simplified distribution was a good approximation of the more complex original one. The values are displayed in Tab. 7.10.

The previously presented particle characterization routine could not be employed one to one, as the average particle size exceeds the possible maximal size of our laboratory devices. External measurements conducted by Benvenuti et al. [2014] yielded 0.8 and 0.4 for the coefficient of friction and rolling friction respectively. A typical value for the coefficient of restitution for coke is 0.5. Tab. 7.11 shows all simulation parameters used. As customary for DEM calculations, the Young's modulus in the simulations was chosen to be smaller than

	U	Us	p	rho	void fraction	Ksl
inlet	fixedValue, (100 0 0)	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
wall	fixedValue, (0 0 0)	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
top	zeroGradient	zeroGradient	fixedValue, 1.0e5	zeroGradient	zeroGradient	zeroGradient
bottom	fixedValue (0 0 0)	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
internal	slip	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
sides	slip	zeroGradient	zeroGradient	zeroGradient	zeroGradient	zeroGradient
internal field	(0 0 0)	(0 0 0)	1.0e5	1.1885	1	0

Tab. 7.9: Boundary conditions for the CFD-case.

diameter (cm)	ratio (%)
3	23.5
6	36.2
8	28.3
10	12.0

Tab. 7.10: Particle size distribution in the simulations.

the Young's modulus of the real material. This practice has little effect on the results but allows for larger DEM time steps (cf., Muguruma et al. [2000]).

Young's modulus	$5 \cdot 10^6$
Poisson ratio	0.45
coefficient of restitution	0.5
coefficient of friction	0.8
coefficient of rolling friction	0.4

Tab. 7.11: Material parameters for the 3D simulation.

7.3.1.4 Additional case settings

The DEM time step width was chosen as $5 \cdot 10^{-6}$ s, and the CFD time step width as $5 \cdot 10^{-5}$ s,. The coupling was carried out each 10 DEM steps. The case itself consisted of two parts:

- **Initialization**: First a pure DEM calculation was launched. The desired amount of particles was inserted in a domain three times higher than the target-domain. The settling process was accelerated with a servo wall that compressed the particles. The few particles that managed to escape the domain were deleted. In order to speed up the process of cavity breakup, the removal of particles from the area where later on the raceways would form was already commenced. Finally the data was stored in a LIGGGHTS restart file.

- **Coupled calculation**: The DEM part was initialized with the LIGGGHTS restart file, for the CFD calculations the previously discussed boundary and initial conditions were used. The computations were carried out with the incompressible, iso-thermal solver `cfdemSolverPimple`. The inlet velocity was increased gradually until the desired final blast velocity of 100 m/s was reached. The material removal in front of the tuyere was continued. The properties set in the `couplingProperties`-file are collected in Tab. 7.12.

  Instead of the previously used drag model by Koch and Hill models by Di Felice (cf., Zhou et al. [2010]) and Gidaspow (cf., Zhu et al. [2007]) were used.

  The computations were carried out on cores with 8 to 12 nodes. The approximate calculation time for one second of real time was about eight hours, slight variations depending on the hardware could be observed. The cases were run for 10-20 seconds.

model name	model
modelType	A
couplingInterval	10
locateModel	engine
momCoupleModels	implicitCouple
turbulenceModelType	RASProperties
IOModel	basicIO
meshMotionModel	noMeshMotion
regionModel	allRegion
dataExchangeModel	twoWayMPI
averagingModel	dense
clockModel	standardClock
smoothingModel	off
forceModels	DiFeliceDrag or GidaspowDrag gradPForce viscForce

Tab. 7.12: Properties for the coupled case.

7.3.2 Results

In the 3D case a direct comparison of simulation and experiment was not possible, lacking experimental data. On the one hand measurement data from inside a blast furnace is hard to obtain and on the other hand the simulation set-up was still simplified in comparison to the real process. Thus in the present case the focus lay on a qualitative depiction of the global behaviour. Matters such as global bed behaviour, raceway penetration depth and pressure drop over the bed will be discussed in the following.

7.3.2.1 Global bed movement

In the previous study the global bed movement of simulation and experiment were compared. In both cases a strong wall effect showed, which caused the particles to move downwards along the wall where the tuyere was mounted (cf., Fig. 7.36). A large part of the particle bed was at rest. In the three dimensional simulations a different bed motion could be observed: the whole particle bed displayed a downwards motion, particles entered the raceway from all directions. Fig. 7.39 compares the particle-motion patterns of the pseudo 2D and the 3D simulation.

The analysis of the time-averaged particle velocity field showed that particles close to the blast furnace wall are at rest. The strongest bed motion could be observed directly above the raceway, a slight decrease of the bed velocity became visible towards the centre (cf., Fig. 7.40). Altogether this behaviour is in far better accordance to the documented process than the behaviour shown in the pseudo 2D case.

An effect that can be observed in real blast furnaces is the formation of the "dead man zone". In this area - located in the centre of the furnace - the coke is almost stagnant. This behaviour could not be observed in the presented model. One reason might be its relative

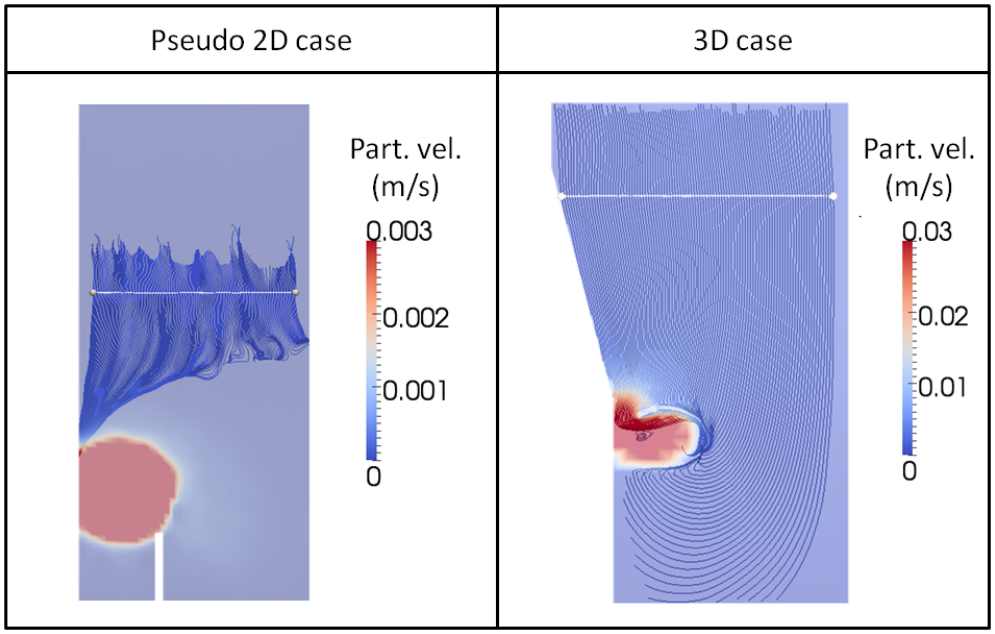


Fig. 7.39: Comparison of the global bed movement - Pseudo 2D case vs. 3D case.

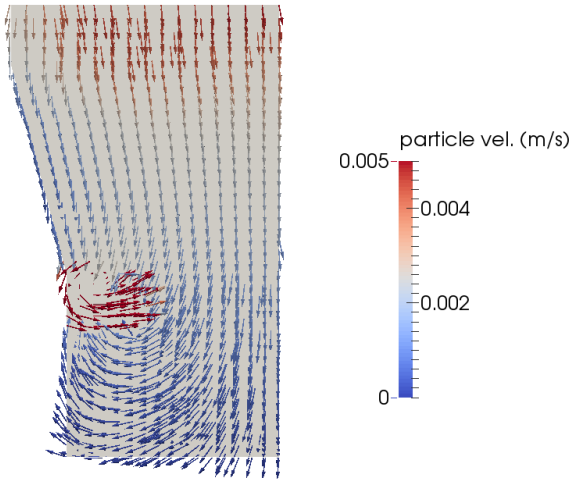


Fig. 7.40: Vectors coloured according to the time averaged particle velocities.

simplicity: from all chemical and thermal reactions only the gasification of the coke inside the blast furnace is accounted for - by the simple removal of particles. Furthermore, as periodic boundary conditions for axial-symmetric objects were unavailable, the very centre of the furnace was not included in the wedge geometry. The combination of the sharp angle and the rather large particles would lead to misleading results. An adaption of the particle rolling friction would help pronouncing this stagnant zone.

7.3.2.2 Fluid velocity

The next point concerned the behaviour of the fluid flow through the particle bed. On the left hand side of Fig. 7.41 the stream lines of the flow field are displayed. They distribute uniformly over the whole geometry. The second image in Fig. 7.41 emphasizes this claim: the image contains the velocity profiles at different levels (broken lines).

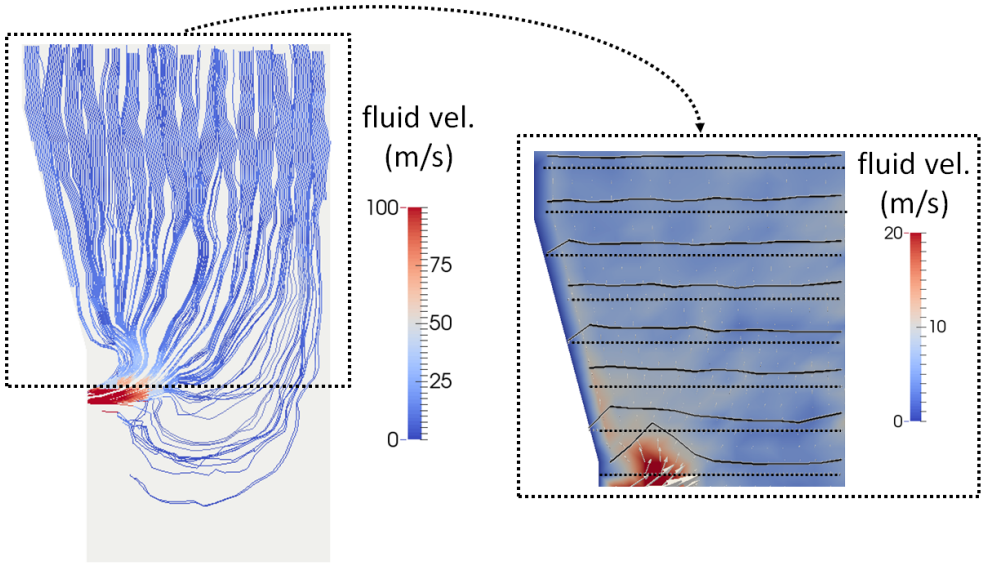


Fig. 7.41: Stream lines and velocity profile of the fluid.

7.3.2.3 Raceway penetration depth

One of the main issues in this investigation was the formation of the raceways inside the blast furnace. The literature does not provide a clear definition of "the raceway". Previously, in the pseudo 2D case one could compare simulation and experiment directly, which was not possible in the present case. One method for distinguishing between bed and raceway is given in Fig. 7.42. It shows a series of snapshots of a thin layer of particles at the central tuyere. The images were taken at the same time step in the same simulation, the colouring was chosen according to the particle velocity. Those, moving faster than the given limit-velocity are depicted in red, the slower ones in light grey. The left most image shows that the particle velocity of 4 m/s is only exceeded in exceptional cases. The second and third image show that there is a remarkable amount of spheres moving faster than 0.1 and 0.01 m/s respectively. It becomes visible that the raceway follows the shape of the outer blast furnace wall, which is curves outwards. The right most image shows that almost the whole particles bed is in a slow motion.

Another perspective on the particle bed is given by Fig. 7.43. Here a horizontal layer of particles at tuyere level is regarded. The left image stems from a calculation, where the drag force model of Di Felice was used and the right one was taken in a simulation where the drag model of Gidaspow was employed. The particles are coloured with respect to their velocity, the green arrow marks the approximate raceway penetration depth. A close look on

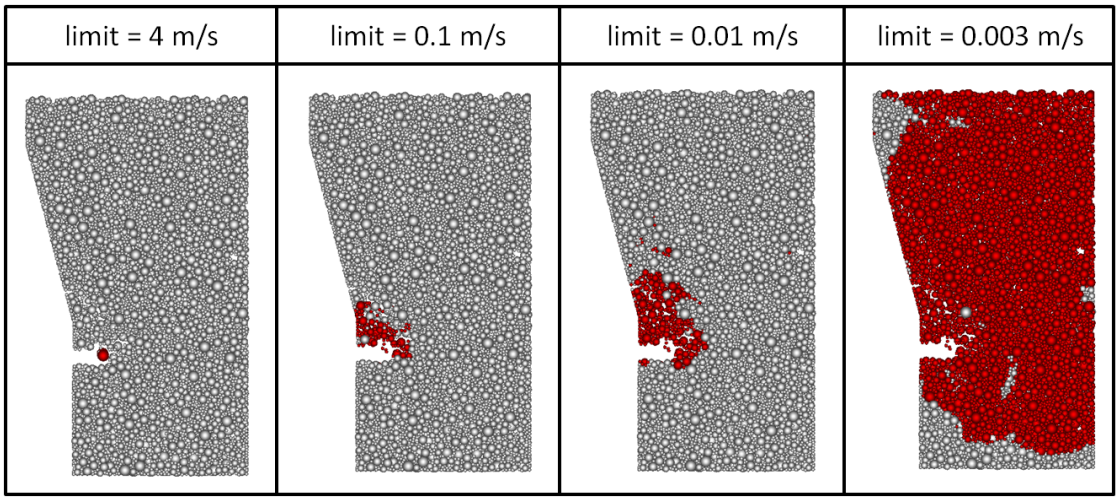


Fig. 7.42: The red particles move faster than the given velocity limit, the grey ones slower.

the two images shows that the results differ slightly. Yet, the average raceway penetration was around 0.5 metres in both cases.

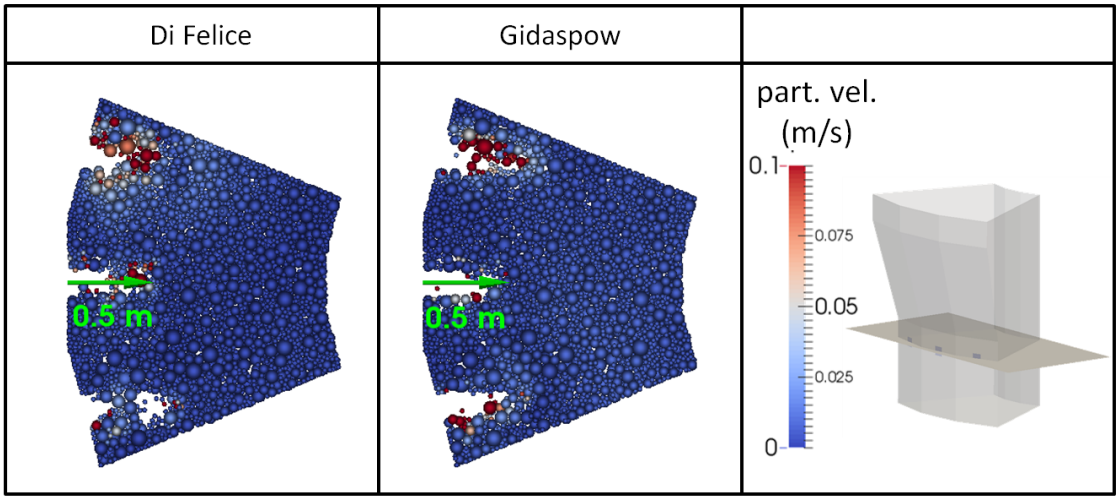


Fig. 7.43: Raceways at tuyere level.

In practice, one often uses heuristic models for predicting the raceway penetration depth. One commonly applies the model by Merry, which is given as

$$L = 5.2 \cdot \left(\frac{\rho_f d_T}{\rho_s d_p}\right) 0.3 \cdot d_T \cdot (1.3 \cdot \left(\frac{u^2}{g d_T}\right)^{0.2} - 1), \tag{7.8}$$

where d_T is the tuyere diameter, d_p the particle diameter, ρ_f and ρ_s fluid and particle density, u the inlet velocity and g the gravitational constant. For the presented simulations a penetration depth of 0.8 m was evaluated. Thus the simulation seems to under-predict the raceway size.

Two main factors can be responsible for this deviation: on the one hand, the smallest

fraction of particles could not be depicted in the simulations. On the other hand, no data about the void fraction of the bed inside the operating furnace was available. The chosen particle bed resulted in a void fraction of about 0.26, and the assumption that it is smaller in reality is admissible. The void fraction impacts strongly on the pressure drop over the bed and thus also influences the dynamics of the particles, which lead to the following investigation:

7.3.2.4 Pressure drop investigation

An industrial partner provided pressure measurement data along a blast furnace. The pressure data was extracted at seven points along the height of the furnace, starting closely beneath the tuyere level. Fig. 7.44 shows the pressure distribution inside the described system.

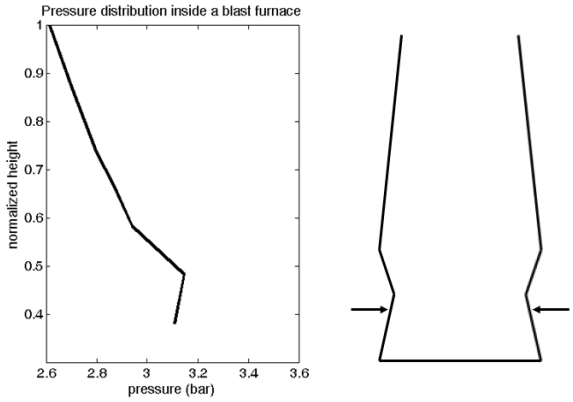


Fig. 7.44: Average pressure distribution inside a blast furnace of normalized height.

In the area of the tuyere the pressure first increases and then decreases rapidly. A comparison of the single pressure measurement showed strong variations. However, in a certain height above the raceways the pressure drop becomes almost constant.

The system for which the data was provided differed from the one investigated previously. However, the question that arose now, was whether or not the previously used granular material was suitable for obtaining the observed pressure drop under the present conditions. In the following numerical investigation particles of the same diameter distribution and material properties were filled into a cylinder and compressed accordingly. As the goal was the depiction of the pressure drop in the upper region of the blast furnace (cf., Fig. 7.45), the flow properties in this area were deduced from the provided data (see Tab. 7.13).

fluid velocity (m/s)	0.3
pressure (bar)	3.75
fluid density (kg/m^3)	2.5
kinematic viscosity (m^2/2)	6.95 $\cdot 10^{-6}$

Tab. 7.13: Fluid properties in the cylinder test case.

A steady air flow from bottom to top was imposed, and the simulations were run until a steady solution was reached.

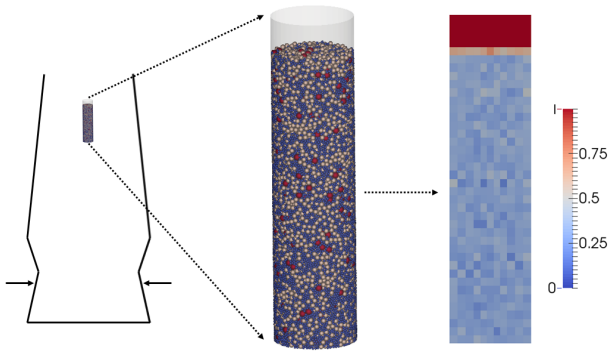


Fig. 7.45: Simulation of a cylinder, filled with the granular material from the previous simulations - run with the boundary and initial conditions of the real size blast furnace.

The values obtained by the measurements lead to an average pressure drop of 2300 Pa/m in the upper furnace region. The initial simulation only resulted in a pressure drop of about 100 Pa/m, thus with the previously used material the pressure drop was under-predicted by a factor of 23. This behaviour is an additional indicator of an over prediction of the void fraction. CFDEMcoupling allows for an artificial manipulation of the void fraction by changing the particle's volume in the representation in the CFD mesh (cf., Fig. 7.46).

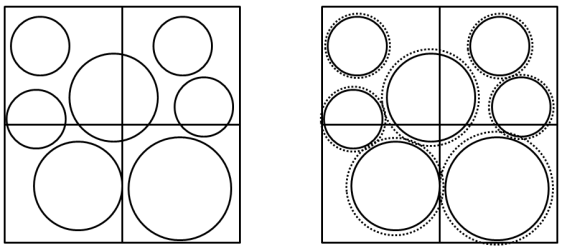


Fig. 7.46: Artificially decreasing the void fraction by increasing the particle diameter on the CFD side.

Due to the increased particle volume, the drag force that acts onto each object is increased. Different weighting factors were applied, and finally a 10 % increase of the diameter lead to the same pressure drop values as in the blast furnace. The void fraction in this case was 0.1. In the `couplingProperties`-dictionary this change corresponded to a correction factor "`weight`" of 1.3 in the `dividedProps`, as this factor relates to the particle volume.

7.3.2.5 *Final calculations*

For finding out about the impact of the changed void fraction on the raceway formation, a new simulation was set up. The settings were chosen as in the previous investigations with the exception of the corrected void fraction field. In Fig. 7.47 the raceways resulting in an uncorrected and a corrected case are compared. One can see that both the cavity and the

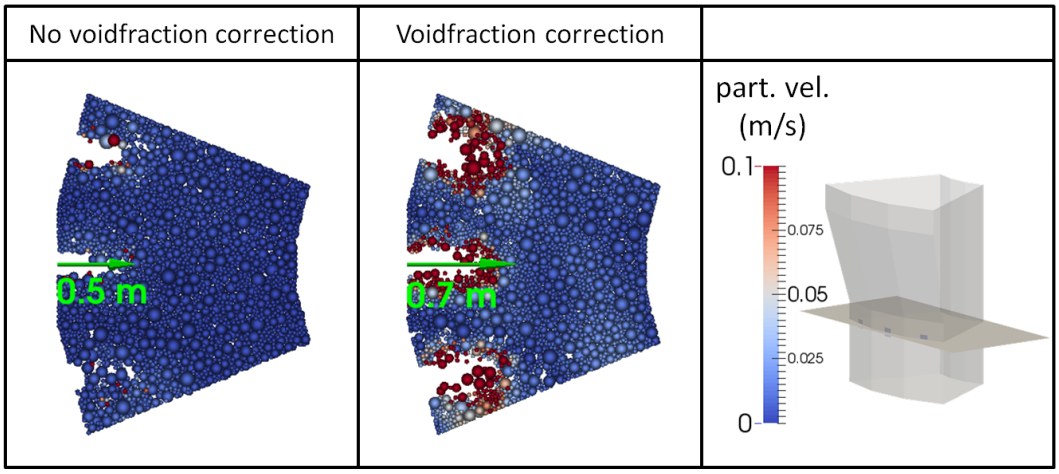


Fig. 7.47: Raceway formation with and without void fraction correction (Di Felice drag model).

layer of moving particles increased significantly in the case with the corrected void fraction. While the old calculations yielded a penetration depth of around 0.5 meters, the new ones lead to about 0.7 m. The calculations were carried out for both previously employed drag models (cf., Fig. 7.48). The usage of the two models yielded slightly different results, yet the qualitative outcome was similar.

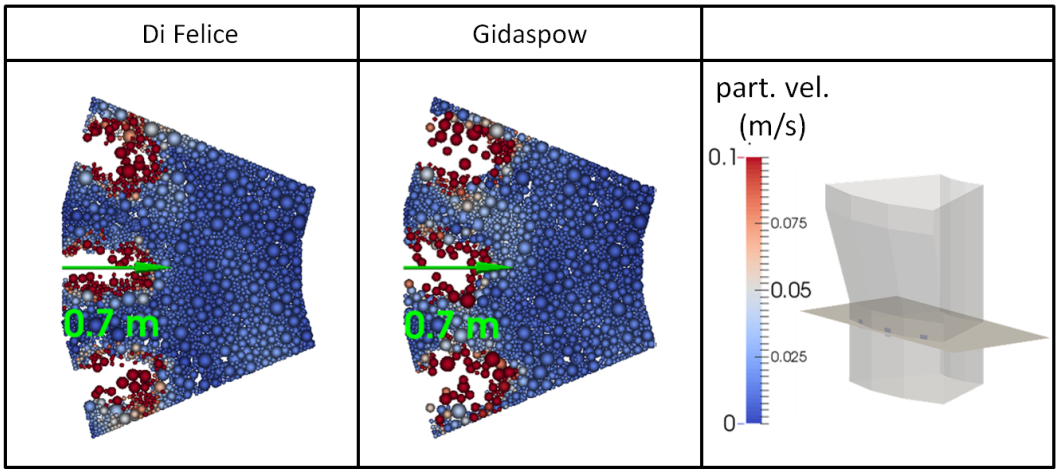


Fig. 7.48: Raceway formation with corrected void fraction (Di Felice vs. Gidaspow drag).

When displaying a vertical layer of particles the changed behaviour of the particle bed became visible (cf., Fig. 7.49). Just an in Fig. 7.42 before, the particles faster than the limit velocity are coloured red and the slower ones are grey. The increased drag force did not only lead to a larger raceway, but also lead to a loosening of the bed above the raceway area.

In conclusion one can state that the void fraction of the particle bed has a strong impact on the results. By artificially decreasing the void fraction both shape and size of the raceway change strongly. The assumption that the particle bed inside a blast furnace is rather dense is admissible, but nevertheless further measurement data would be necessary for future investigations.

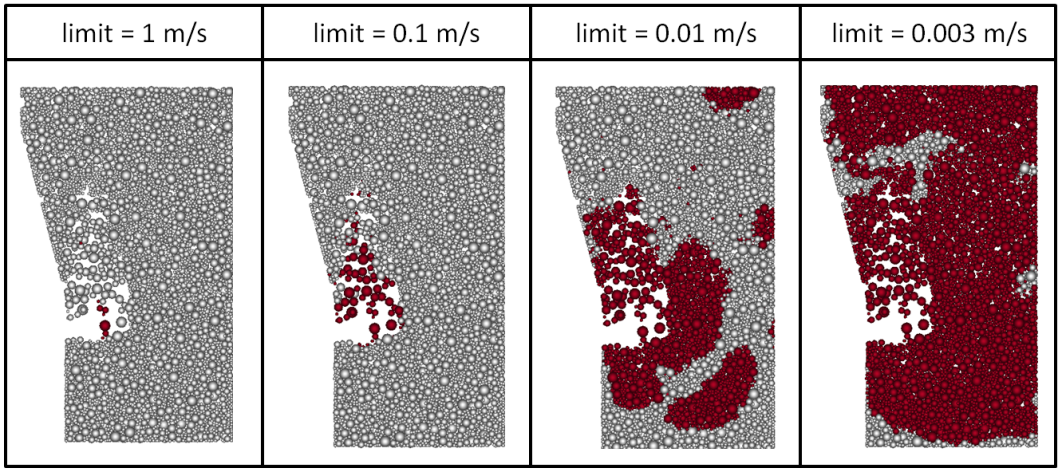


Fig. 7.49: Particle velocities inside the bed in the simulation with corrected void fraction. The red particles move faster than the limit values, the grey ones slower.

8

Summary and Conclusion

The goal of this thesis was to deal with of a possibly wide range of particle-fluid problems. Discrete Element Methods (DEM) have proved to be capable of handling systems of particles with great accuracy, thus possibilities to couple this method with an approach for solving pure fluid problems were searched. In the field of coupled methods CFD-DEM gained more and more relevance over the years, and it was selected due to the various forms it can take. Within the course of this work, two different coupling modes were used, namely resolved and unresolved CFD-DEM. Each of the topics was discussed in a designated part:

Part I was dedicated to resolved CFD-DEM. First of all a new solver was implemented. After the validation of its basic functionality a number of improvements were made. The final version was fully parallel and the last examples showed the method's applicability to multi-body problems. The solver itself is able to handle arbitrary meshes, but as all examples considered so far involved hexahedral cells only, the submodels were designed for those cases. However, thanks to the modular structure of the code an extension to the general case is possible and does not require fundamental structural modifications. The developments were carried out in the framework of CFDEMproject, an online platform that is specialised on the development and coupling of Open Source codes. The outcomes of this thesis have been made accessible to the community, and - to the author's knowledge - have been applied by other research groups.

In Part II unresolved CFD-DEM was chosen to build a simulation tool for an application example. In contrast to the previous case, where a lot of fundamental development took place, an already existing solver was used, which enabled the start from a - numerically speaking – higher level. The given task was the simulation of raceways inside blast furnaces, which was too complex for a one-step approach. For getting an idea of the process a small scale pseudo 2D lab model was built up. After first practical investigations the process was simulated. During the extensive study several adjustments were made for getting both a more realistic depiction of the process by the experiment and more accurate numerical results. However, the walls turned out to influence the forming raceways drastically, thus

the next step lead to a 3D model. The new test case was built up according to a full size blast furnace, using parameters provided by the industry. Only the void fraction of the particle bed, a quantity of great importance, was not available. First simulations suggested that the initially assumed value was too high. The parameter could be calibrated with the aid of a pressure drop study, and the final simulation results were then in good accordance with the estimates of the heuristic models. Nevertheless, even though the created model was able to depict the formation of raceways inside blast furnaces rather realistically, a lot remains to be done in this area, for instance the introduction of non-isothermal, compressible reacting models. The whole process is of high complexity, and many aspects such as chemical reactions remained untouched. The presented work is to be understood as a first step in this direction.

APPENDIX

A

Computational Fluid Dynamics

Computational Fluid Dynamics (CFD) combines classical fluid mechanics and numerical methods to allow for the solution of complex flow problems. First advances in fluid mechanics date back to Ancient Greece, where e.g. Archimedes was concerned with hydrostatics. Numerical methods developed a lot later. First algorithms were found by Richardson [1911] and executed by "human computers". Around 1960, the introduction of computes enabled the consideration of larger problems.

This chapter will give an overview over the governing equations, their deduction and their treatment with numerical tools. Before going into detail, a few preliminary remarks are made:

A fluid is defined as material that requires a to zero tending force for a to zero tending deformation. The continuum hypothesis states that a fluid can be considered as continuum, even though its smallest structure consists in distinct molecules (e.g., Streeter and Wylie [1985]). The matter itself is characterized by parameters such as viscosity or density. Both properties can be used to deduce criteria that allow for the classification of materials. If one can for example establish a linear relation between applied shear force and deformation, a fluid is called Newtonian, otherwise non-Newtonian. Fluids that change their density over time, on the other hand, are categorized as compressible in contrast to the incompressible ones. A fluid can change its properties along with the operating conditions: it can for instance become compressible due to motion or temperature. The Mach number (Ma), being defined as the ratio of speed to the specific sound speed of the fluid, divides flow regimes into a subsonic and a supersonic section. If $Ma < 0.3$ holds, a flow can be considered incompressible. Another way to differentiate flow regimes with respect to their velocity is the division into creeping flow, laminar flow, transitional regimes and turbulent flow.

The equations that will now be deduced provide an exact description of a flow problem, no matter if the fluid in question is compressible or incompressible or if the flow is laminar or turbulent. Nevertheless, when it comes to numerics, models are used to represent the turbulence. The reason is that turbulent structures develop on very small scales, which

cannot be depicted with sufficient precision. The applications dealt with later on will exclusively involve incompressible flows, meaning that from now on the density will be considered constant.

A.1 Governing equations

The computational domain Ω is defined as subdomain of $\mathbb{R}^3$. When regarding the dynamics of the flow problem considered, one could choose a "fluid particle" and track its motion through the domain. This approach is very popular in continuum mechanics where this "particle" - referred to as control mass - can easily be defined. As soon as fluids are involved, it turns out to be more intuitive to use a fixed control volume instead of a mass. Reynold's theorem is used to establish a connection between these two ideas. According to Spurk and Aksel [1997] it relates the change of a transport quantity φ in the volume of the control mass at time t ($V(t)$) to the change of the quantity in a fixed control volume (V) and its flux over the boundaries (S):

$$\frac{d}{dt}\int_{V(t)} \varphi dV = \int_V \frac{\partial \varphi}{\partial t} dV + \int_S \varphi u \cdot n \, dS, \tag{A.1}$$

where u is the fluid velocity and n the outer normal vector.

The governing equations for the fluid system can be set up for a time dependent control volume and then be transformed to a fixed element. The equations that describe the system are based on two main assumptions, which are

- the conservation of mass m and
- the conservation of momentum $m\ u$.

The first one can be represented by

$$\frac{dm}{dt} = 0 \mid_{V(t)}, \tag{A.2}$$

or, using the transport theorem

$$\frac{dm}{dt} = \frac{d}{dt}\int_{V(t)} \rho \, dV = \int_V \frac{\partial \rho}{\partial t} \, dV + \int_S \rho u \cdot n \, dS = 0. \tag{A.3}$$

Due to incompressibility $\rho = const$ holds over the whole domain. Since this assumption is satisfied for all control volumes in the domain, the application of Gauss' theorem yields the following divergence-free condition:

$$\nabla \cdot u = 0. \tag{A.4}$$

The basis for the conservation of momentum is formed by Newton's second law, which states that the change of momentum equals the sum of all forces f acting onto the volume. It can be denoted as

$$\frac{d(mu)}{dt} = \sum f. \tag{A.5}$$

The momentum mu can be expressed by the volume integral $\int_{V(t)} \rho u \; dV$. Inserting it into eq. (A.5) and applying eq. (A.1) yields

$$\frac{d}{dt}(mu) = \frac{d}{dt}\int_{V(t)} rhou \; dV = \int_V \frac{\partial}{\partial t}(\rho u) \; dV + \int_S \rho u u \cdot n \; dS. \tag{A.6}$$

The forces $\sum f$ can be divided into body forces and surface forces. Body forces are of the form

$$F_{body} = \int_V (t)\rho f \; dV, \tag{A.7}$$

with f being a specific force density. Examples are the gravity ($f = (0, 0, g)$, where $g \approx 9.81$) or electromagnetic forces. Surface forces can be described by a stress vector t, called Cauchy stress vector:

$$F_{surface} = \int_{S(t)} t \; dS. \tag{A.8}$$

The stress vector can be represented as the product of the Cauchy stress tensor and the normal vector:

$$t = \sigma \cdot n \tag{A.9}$$

.

The conservation of momentum consists of three equations, and together with the conservation equation this yields a total of four. In order to be able to determine velocity, pressure and shear forces, the system needs to be extended. As already mentioned in the introductory part, only Newtonian fluids are considered in this thesis. These are characterized by the fact that the Cauchy stress tensor is of the following form:

$$\sigma = -pI + \tau, \tag{A.10}$$

where I is the identity matrix and τ is the shear stress, given as

$$\tau = \lambda \nabla \cdot u \; I + 2\mu\epsilon(u), \tag{A.11}$$

with ϵ defined as

$$\epsilon(u) = \frac{1}{2}(\nabla u + (\nabla u)^T). \tag{A.12}$$

μ and λ are coefficients of viscosity, and using the Stokes hypothesis that states that

$$\lambda = \frac{2}{3}\mu \tag{A.13}$$

one gets

$$\sigma = -(p + \frac{2}{3}\mu\nabla \cdot u)I + 2\mu\epsilon(u) \tag{A.14}$$

for the stress tensor σ. Collecting all the terms that were deduced so far leads to

$$\int_V \frac{\partial(\rho u)}{\partial t} dV + \int_S \rho u u \cdot n \; dS = \int_V \rho f \; dV + \int_S \sigma \cdot n \; dS. \tag{A.15}$$

.

Just as in the case of the mass conservation Gauss's theorem can be applied, resulting in

$$\frac{\partial(\rho u)}{\partial t} + \nabla \cdot (\rho u u) = \rho f + \nabla \cdot \sigma. \tag{A.16}$$

This form is called conservative. Transforming the second term on the right hand side, using eq. (A.4), yields the convective form

$$\frac{\partial(\rho u)}{\partial t} + \rho u \cdot \nabla u = \rho f + \nabla \cdot \sigma. \tag{A.17}$$

According to eq. (A.14) the last term, $\nabla \cdot \sigma$, can be transformed as follows:

$$\begin{aligned}
\nabla \cdot \sigma &= \nabla \cdot (-(p + \frac{2}{3}\mu \underbrace{\nabla \cdot u}_{=0})I + 2\mu\epsilon(u)) && \text{(A.18)}\\
&= -\nabla p + \mu\nabla \cdot (\nabla u + (\nabla u)^T) && \text{(A.19)}\\
&\overbrace{=}^{Appendix} -\nabla p + \mu(\Delta u + \nabla(\underbrace{\nabla \cdot u}_{=0})) && \text{(A.20)}\\
&= -\nabla p + \mu\Delta u && \text{(A.21)}
\end{aligned}$$

In order to obtain a complete description of the flow situation, initial and boundary conditions have to be prescribed. Several kinds of boundary conditions are known, two typical ones are the Dirichlet and the Neumann type boundary condition. When applying the Dirichlet boundary condition, a certain velocity is defined in the designated boundary region $\partial\Omega_D$. In case of a Neumann boundary condition a surface force is prescribed on the affected area $\partial\Omega_N$. One often encounters a combination of both boundary conditions.

So, in summary, the dynamics of an incompressible, Newtonian fluid without gravity is described by the following set of equations, termed Navier-Stokes equations:

$$\begin{aligned}
\nabla \cdot u &= 0 \text{ in } \Omega && \text{(A.22)}\\
\frac{\partial(\rho u)}{\partial t} + \rho u \cdot \nabla u &= -\nabla p + \mu\Delta u \text{ in } \Omega && \text{(A.23)}\\
u &= u_D \text{ on } \partial\Omega_D && \text{(A.24)}\\
\sigma \cdot n &= t_N \text{ on } \partial\Omega_N && \text{(A.25)}\\
u &= u_0 \text{ in } \Omega \text{ for } t = 0 && \text{(A.26)}
\end{aligned}$$

Eq. (A.22) is referred to as continuity equation, Eq. (A.23) as momentum equation. The system of equations is completed by the boundary conditions (A.24) and (A.25) end the initial condition (A.26), which describes the state of the system at $t = 0$.

A.2 Numerics

In the previous chapter a rough overview over the deduction of the governing equations was given. The resulting differential equations are transient, non-linear and of second order. The Navier-Stokes equations do not possess a known solution, even worse, its existence and uniqueness could not be proven so far [1]. However, it has been proven admissible to solve them with the aid of numerical methods.

Several approaches for solving differential equations numerically have been developed over the years, examples are the finite volume method, the boundary element method, the finite difference method or the finite volume method. As is customary for fluid dynamical methods, the latter one will be applied in this thesis. It is based on the integral form of the conservative Navier-Stokes equations.

Apart from the precisely defined analytical problem the application of a numerical algorithm requires the discretization of the problem. In the following two sections an overview of obtaining such a discretization and an example for a solution method are given.

A.2.1 Discretization of the governing equations

As the problem considered is three dimensional and transient, the solution space can be described as space-time-cylinder. Apart from a full discretization (e.g. space-time finite element methods, e.g., Heinrich and Pepper [1999]) a method termed semi-discretization can be applied. In this case, spacial and temporal are carried out separately. In this context two approaches are known (e.g., Knabner and Angermann [2003]):

- The horizontal method of lines, also known as Rothe method: Here one first discretizes with respect to the time variable and then proceeds with the spacial discretization.
- The vertical method of lines: in complement to the Rothe method the equations are first discretized with respect to space, leading to a Galerkin approximation with time dependent coefficients.

In the present case the horizontal method of lines was chosen. Time and space require different treatment in the given equations. While an event at a certain point in time can only affect the future state of the system, while a local event can have an effect anywhere in space. The latter statement requires an elliptic system of equations. If this applied, the whole problem is parabolic with respect to time. Further intel on the classification of partial differential equations can for example be found in Evans [1998].

[1] Proving the existence and uniqueness of a solution of the Navier-Stokes equations is one of seven Millennium Prize Problems, named by the Clay mathematics institute in 2000. (cf. http://www.claymath.org/millennium/)

The time space was divided into a number of time intervals, the results for each time step were calculated consecutively. The spacial domain Ω is divided into a a set of subdomains Ω_i, $i = 1, ..., n$. These subdomains have to be distinct from each other and must - in sum - add up to the whole domain. One such domain is called cell, their collective is the computational mesh.

The main ideas of the method will now be developed for a specific spacial domain within the mesh, called control volume. Fig. A.1 illustrates such a control volume. For the sake of simplicity a 2D element will be considered. The extension to 3D works rather straight forwards. P marks the centre of the element, N, S, E and W the centres of the neighbouring volumes in the north, south, east and west. The position of the centre will later on be referred to as x_P. n_N, n_S, n_E and n_W denote the outer normal vectors in the designated direction.

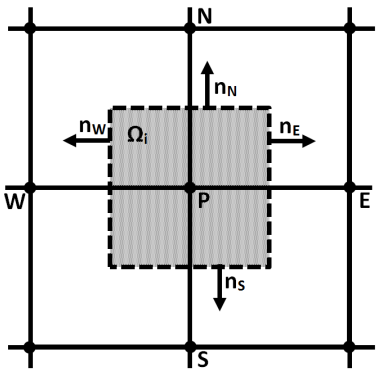


Fig. A.1: Control volume Ω_i in the finite volume method.

The control volume is chosen in such a way that

$$\int_{\Omega_i} (x - x_P)\, dV = 0 \tag{A.27}$$

holds for all $i = 1, ..., n$.

In the following we consider the term wise discretization of the integral form of the conservative momentum equation, eq. (A.16). The expression consists of an transient term, a convective term, a diffusive term and a source term:

$$\underbrace{\int \frac{\partial}{\partial t}(\rho u)\, dV}_{\text{instationary term}} + \underbrace{\int \nabla \cdot (\rho u u)\, dV}_{\text{convective term}} - \underbrace{\int \nabla \cdot (\mu \nabla u)\, dV}_{\text{diffusive term}} = \underbrace{- \int \nabla p\, dV}_{\text{source term}} \tag{A.28}$$

A.2.1.1 Temporal discretization

As the horizontal method of lines comes to application, the equation is first discretized with respect to time. The time interval $[0, T)$ is divided into a number of time intervals of length Δt, starting at t_0:

$$[0,T) = [0,\Delta t] \cup [\Delta t, 2\Delta t] \cup ... \cup [T-\Delta t, T). \tag{A.29}$$

We consider the equation for the interval $[t, t+\Delta t)$:

$$\int_t^{t+\Delta t} \frac{\partial}{\partial t} \int \rho u \, dV \, dt = \int_t^{t+\Delta t} \left[-\int \nabla \cdot (\rho u u) \, dV + \int \nabla \cdot (\mu \nabla u) \, dV - \int \nabla p \, dV \right] dt \tag{A.30}$$

In the unsteady term integration and differentiation were interchanged. According to Wong [1999] (cf., Theorem 1.1) this operation is feasible, taken that the required preconditions are met. Before solving (A.30), the spacial discretization of the volume integrals over the control volumes is considered.

A.2.1.2 Spatial discretization

As ρ is constant, it will be omitted in the following. First of all, numerical integration is applied to each term:

- Transient term: In this case u is replaced by an approximation of the Taylor series expansion around the centre of the cell x_P:

$$\int_{\Omega_i} u \, dV = \int_{\Omega_i} u_P + (x - x_P)(\nabla u)_P \, dV = u_P \int_{\Omega_i} 1 \, dV + (\nabla u)_P \int_{\Omega_i} (x - x_P) \, dV = u_P |\Omega_i|, \tag{A.31}$$

  where u_P is $u(x_P)$ and $(\nabla u)_P$ is $\nabla u(x_P)$. The latter equality follows from assumption (A.27).

- Convective term: In order to discretize the convective term, the divergence theorem is used to obtain

$$\int_{\Omega_i} \nabla \cdot (uu) \, dV = \int_{\partial\Omega_i} uu \cdot n \, dS. \tag{A.32}$$

  The transformed term is then used for discretization:

$$\int_{\partial\Omega_i} uu \cdot n \, dS = \sum_{j \in \{1,...,n\}} \int_{\partial\Omega_{i,j}} uu \cdot n \, dS \approx \sum_{j \in \{1,...,n\}} \partial\Omega_{i,j} u_{\partial\Omega_{i,j}} u_{\partial\Omega_{i,j}} = \sum_{j \in \{1,...,n\}} F_{\partial\Omega_{i,j}} u_{\partial\Omega_{i,j}}, \tag{A.33}$$

  where $\partial\Omega_{i,1}, ..., \partial\Omega_{i,n}$ are the single faces of the volume element Ω_i and $u_{\partial\Omega_{i,j}}$ stands for an interpolated value at the corresponding face. $F_{\partial\Omega_{i,j}}$is the flux across the according face. Fig. A.1 illustrates the position of the faces etc. for a 2D element.

- Dissipative term: Just as in the previous case, the divergence theorem is applied to transfer the volume integral to a surface integral:

$$\int_{\Omega_i} \nabla \cdot (\mu \nabla u) \, dV = \int_{\partial\Omega_i} \mu \nabla u \cdot n \, dS \tag{A.34}$$

Numerical integration leads to

$$\int_{\partial\Omega_i} \mu\nabla u \cdot n \, dS = \sum_{j\in\{1,\dots,n\}} \int_{\partial\Omega_{i,j}} \mu\nabla u \cdot n \, dS \approx \sum_{j\in\{1,\dots,n\}} \mu(\nabla u)_{\partial\Omega_{i,j}} |\partial\Omega_{i,j}|, \tag{A.35}$$

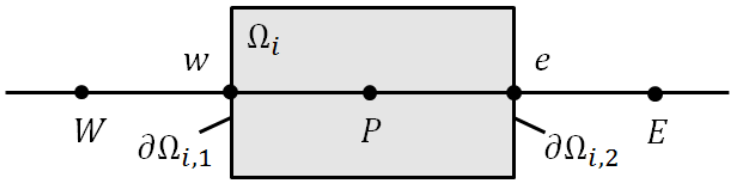
where $(\nabla u)_{\partial\Omega_{i,j}}$ is the velocity gradient at the face $\Omega_{i,j}$.

- Pressure term: For calculating the according contribution of the pressure term, the following steps are necessary:

$$\int_{\Omega_i} \nabla p \, dV \approx (\nabla p)_P \int_{\Omega_i} 1 \, dV = (\nabla p)_P |\Omega_i|, \tag{A.36}$$

where $(\nabla p)_P$ is $\nabla p(x_P)$.

For evaluating the integrals, values at faces etc. need to be determined. The calculation of $u_{\partial\Omega_{i,j}}$ requires the choice of an interpolation scheme, as the values of the velocity are saved at the cell centres only. The gradients $(\nabla u)_{\partial\Omega_{i,j}}$ and $(\nabla p)_P$ are computed with the aid of a numerical differentiation. Fig. A.2 gives a very simple example for the computation of the terms in question in 1D.



$$\partial\Omega_i = \partial\Omega_{i,1} \cup \partial\Omega_{i,2}$$

- $u_{\partial\Omega_{i,1}} = u_w \approx \frac{u_W + u_P}{2}$
- $(\nabla u)_{\partial\Omega_{i,1}} \approx \frac{u_W - u_p}{\Delta t}$ or
  $(\nabla u)_{\partial\Omega_{i,1}} \approx \frac{(\nabla u)_W + (\nabla u)_P}{2}$
- $(\nabla p)_P = \frac{p_{\partial\Omega_{i,1}} - p_{\partial\Omega_{i,2}}}{\Delta t} = \frac{\frac{p_W + p_P}{2} - \frac{p_P + p_E}{2}}{\Delta t}$
  $= \frac{p_W - p_E}{2\Delta t}$

Fig. A.2: Simplest way to numerically evaluate $u_{\partial\Omega_{i,j}}$, $(\nabla u)_{\partial\Omega_{i,j}}$ and $(\nabla p)_P$

.

The pressure gradient is evaluated from the neighbour values only, i.e. the grid for the pressure calculations is coarser than the grid for the remaining operations. Consequently a

chequered pressure field as given in Fig. A.3 is treated as uniform field. This phenomenon is known as checker boarding.

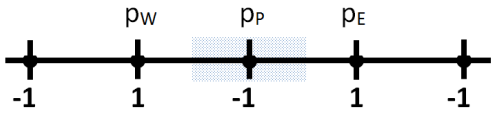


Fig. A.3: Checker boarding on a 1D pressure field (control volume is marked grey)

More advanced schemes are used to circumvent such troubles and to obtain higher accuracy. The CFD software package OpenFOAM, on which the later on used solvers are based, for example uses the Rhie-Chow interpolation scheme (cf., Kärrholm [2008]).

The discretization with respect to time can be carried out in different ways. One option is the Crank-Nicholson scheme, where the temporal derivative is replaced by the backwards difference quotient and the integration is a simple trapezoidal rule. It is unconditionally stable, but the boundedness of the solution cannot be guaranteed.

A.2.2 Solving the Navier-Stokes equations

When it comes to solving the transient Navier-Stokes equations, two main difficulties have to be accounted for: on the one hand, the equations are non-linear due to the convective term, and on the other hand the pressure-velocity coupling needs special treatment. In 1986, Issa [1986] published the PISO-algorithm (Pressure Implicit with Splitting of Operator), a procedure which is now commonly used in CFD. Since its introduction the algorithm experienced a number of improvements, like as for example discussed by Ferziger and Perić [2002] and Jasak [1996]. As the latter version is the one implemented in OpenFOAM, it will be discussed briefly in the following.

The linearization of the convective term has the following form:

$$\nabla \cdot (UU) = a_P U_P + \sum_N a_N U_N, \tag{A.37}$$

where a_P and a_N are functions of U. While a_P is the matrix coefficient in the considered point, a_N accounts for the neighbours. The discretized expression in eq. (A.37) is linearized, and for the calculation of the coefficients an existing velocity field is used that satisfies the divergence-free condition. As discussed by Jasak, the whole equation system can be formulated as

$$a_P U_P = H(U) - \sum_f S(p)_f, \tag{A.38}$$

$$\sum_f S\left[\left(\frac{1}{a_P}\right)_f (\nabla p)_f\right] = \sum_f S\left(\frac{H(U)}{a_P}\right)_f. \tag{A.39}$$

In $H(U)$ all the matrix coefficients velocity contributions from the neighbours as well as source terms from the transient term and other source terms are collected. S denotes the outward pointing face vector, the index f indicates that the values at the faces are used.

As velocity and pressure influence each other linearly, the equations (A.38) and (A.39) need to be coupled. The presented method is segregated, in contrast to simultaneous methods, meaning that the equations are solved in sequence.

In Jasak [1996] the algorithm is described by the following three steps:

1. Momentum predictor: The momentum equation is solved using the pressure field from the previous time step. This leads to a preliminary velocity field.

2. Pressure solution: After the fluxes at the faces are determined the pressure equation is solved, using the preliminary velocity field.

3. Explicit velocity correction: With the aid of the new pressure field, an explicit velocity correction is carried out. Furthermore the boundary conditions are corrected with respect to the new data.

Steps 2 and 3 are repeated for an arbitrary number of times. The whole procedure is repeated for each time step.

B

Discrete Element Method

The Discrete Element Method (DEM) is a Lagrangian method that is used for calculating the dynamics of large granular systems. It was introduced by Cundall and Strack [1979]. In this method the trajectory of each particle is calculated under the consideration of influences by other particles, walls or present forces. The motion of a particle consists of a rotational and a translational component. Using Newton's second law one can describe the translation of an object with respect to its centre of mass by

$$m\frac{du_p}{dt} = F \tag{B.1}$$

$$v = \frac{dx}{dt} \tag{B.2}$$

, where m denotes the mass, u_p the velocity, x the centre of mass and F the forces acting onto the object (cf., Landau and Lifshitz [1976]). In case of granular systems (cf., Luding [2008])- and related to a single particle - the force term consists of

- ... a gravitational component $F_g = m_p g$,
- ... the forces stemming from particle-particle collisions $\sum_{N_p} F_{p,p}$,
- ... and the forces stemming from particle-wall interactions $\sum_{N_w} F_{p,w}$.

N_p and N_w denotes the number of particles and walls in the system respectively. In conclusion this leaves us with

$$m_p\frac{du_p}{dt} = m_p g + \sum_{N_p} F_{p,p} + \sum_{N_w} F_{p,w}. \tag{B.3}$$

The rotational motion of a particle can be described as follows (cf., Landau and Lifshitz [1976]):

$$I_p \frac{d\omega_p}{dt} = T_p, \tag{B.4}$$

where I_p is the inertial tensor, T_p stands for the torques and ω_p is the angular velocity. An alternative formulation of the balance equations can be found in Kloss et al. [2012a].

When a particle's behaviour is simulated with DEM, the equations are solved numerically. On the one hand this means that the equations are evaluated on a discrete time grid, and on the other hand models are used to determine the force terms on the right hand side of equation (B.3). Especially the modelling of the contact forces $F_{p,p}$ and $F_{p,w}$ plays an important role: when in reality either two particles collide with each other or a particle hits a wall, an elastic deformation takes place. Two different approaches, namely the hard sphere and the soft sphere approach, are common to represent the collision.

In the hard sphere model the particles are impenetrable and the contacts are instantaneously and perfectly rigid. Furthermore they are purely binary and long-distance particle forces cannot be considered. In dilute systems the number of binary collisions prevail, making the hard sphere model a good choice. This model was for example used in computer simulations on one of the world's first electronic digital computers, ENIAC, at Los Alamos Science Laboratory (cf. Weik [1961]). An extensive discussion of the method can be found in Mulero [2008].

In the soft sphere approach colliding particles are assumed to overlap to a certain degree. The simplest way to represent the collision force is a linear spring-dash-pot model of the form

$$F = k\delta + \gamma v, \tag{B.5}$$

where k and γ are the spring constant and the damping coefficient respectively, δ is the overlap and v the deformation velocity. Influences from both normal and tangential forces are accounted for.

Generally the computational costs of a soft sphere model are higher than for a hard sphere model. The presented linear spring-dash-pot approach is rather simple and thus cannot capture all effects of the complex collision process as stated in Kloss [2011]. A study comparing a number of different contact models can be found in Di Renzo and Di Maio [2004]. However, the presented approach is a good choice for determining the behaviour of rather large, densely packed particle beds and thus is the model of choice in the following DEM calculations.

C Mathematical Details

C.1 Divergence Theorem

Divergence theorem (cf., Wesseling [2001])

For any volume $V \subset \mathbb{R}^d$ with piecewise smooth closed surface S and any differentiable vector field $\mathbf{u}$ we have

$$\int_V \text{div}\mathbf{u} dV = \int_S \mathbf{u} \cdot \mathbf{n} dS,$$

where $\mathbf{n}$ is the outward unit normal on S.

C.2 Proof of $\nabla \cdot (\nabla u + (\nabla u)^T) = \Delta u + \nabla div\, u$

$$\begin{aligned}
\nabla \cdot (\nabla \mathbf{u} + (\nabla \mathbf{u})^{\mathbf{T}}) &= \begin{pmatrix} \frac{\partial}{\partial x} & \frac{\partial}{\partial y} & \frac{\partial}{\partial z} \end{pmatrix} \begin{pmatrix} 2\frac{\partial u}{\partial x} & \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} \\ \frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} & 2\frac{\partial v}{\partial y} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} \\ \frac{\partial u}{\partial z} + \frac{\partial w}{\partial x} & \frac{\partial v}{\partial z} + \frac{\partial w}{\partial y} & 2\frac{\partial w}{\partial z} \end{pmatrix} \\
&= \begin{pmatrix} 2\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} + \frac{\partial^2 v}{\partial x \partial y} + \frac{\partial^2 u}{\partial z^2} + \frac{\partial^2 w}{\partial x \partial z} \\ \frac{\partial^2 u}{\partial x \partial y} + \frac{\partial^2 v}{\partial x^2} + 2\frac{\partial^2 v}{\partial y^2} + \frac{\partial^2 v}{\partial z^2} + \frac{\partial^2 w}{\partial y \partial z} \\ \frac{\partial^2 u}{\partial x \partial z} + \frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 v}{\partial y \partial z} + \frac{\partial^2 w}{\partial y^2} + \frac{\partial^2 w}{\partial z^2} \end{pmatrix}^T \\
&= \begin{pmatrix} \Delta u + \frac{\partial}{\partial x}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\ \Delta v + \frac{\partial}{\partial y}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \\ \Delta w + \frac{\partial}{\partial z}\left(\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}\right) \end{pmatrix}^T \\
&= \Delta \mathbf{u} + \nabla (\nabla \cdot \mathbf{u})
\end{aligned}$$

D
Lists and tables

D.1 Velocity boundary conditions

case name	material	tuyere diam. (mm)	veloc. (m-s)
rounded_13_race	Rounded	13	55-73-90-109-127-136-145-134-127-109-91-73-56
rounded_14_race	Rounded	14	51-68-85-103-120-129-138-129-121-104-86-70-53
rounded_16_race	Rounded	16	59-74-89-105-114-107-92-76-61-46
rounded_20_race	Rounded	20	32-44-56-68-84-92-100-92-87-74-62-50-37
rounded_13_hys	Rounded	13	109-143-110-144-109-143
rounded_14_hys	Rounded	14	103-137-104-137-104-137
rounded_16_hys	Rounded	16	91-123-92-123-92-123
rounded_20_hys	Rounded	20	69-100-74-100-75-100
sili2mm_13_race	Sili2mm	13	49-67-86-105-124-133-142-133-124-107-88-70-52
sili2mm_14_race	Sili2mm	14	4-6-12-64-81-87-94-87-81-70-57-45-33
sili2mm_16_race	Sili2mm	16	13-20-68-84-101-111-121-112-106-90-74-59-43
sili2mm_18_race	Sili2mm	18	11-13-18-78-93-101-110-102-95-81-67-52-37
sili2mm_20_race	Sili2mm	20	11-12-16-20-83-89-97-89-83-71-58-46-32
sili2mm_13_hys	Sili2mm	13	105-141-107-141-107-141
sili2mm_14_hys	Sili2mm	14	67-92-70-93-70-92
sili2mm_18_hys	Sili2mm	18	76-107-80-107-80-107
sili2mm_20_hys	Sili2mm	20	63-92-68-93-69-94

Tab. D.1: Inlet velocities for the different cases

D.2 Materials

Name	SiLi2mm	SiLi3mm	SiLi4mm	Rounded4mm
Material	glass	glass	glass	Polyenthylen
Density (kg/m^3)	2500	2500	2500	828
Bulk density (kg/m^3)	1500	1500	1500	558
Void fraction	0.38-0.40	0.39-0.40	0.38-0.40	0.33
Diameter (m)	0.002±0.0002	0.003±0.0002	0.004±0.0002	0.004
Shape	spherical	spherical	spherical	rounded cylinders

D.3 Experimental settings

name: hys	engine power*: 60-80-60-80-60-80
	tuyere diameters: 13, 14, 16, 18, 20 materials: SiLi2mm, SiLi3mm, Rounded4mm screw conveyor: off
name: race	engine power*: 30-40-50-60-70-75-80-75-70-60-50-40-30
	tuyere diameters: 13, 14, 16, 18, 20 materials: SiLi2mm, SiLi3mm, Rounded4mm screw conveyor: off
name: mot	engine power*: 45, 55, 65
	tuyere diameters: 13, 14, 16, 18, 20 materials: SiLi2mm, SiLi3mm, Rounded4mm screw conveyor: on
name: ramp	engine power*: 30-40-50-60-70-75-80-75-70-60-50-40-30
	tuyere diameters: 14, 16 materials: SiLi2mm screw conveyor: on
name: press	engine power*: 40, 50, 60, 70, 80
	tuyere diameters: 16 materials: SiLi2mm screw conveyor: off
name: motpress	engine power*: 45, 55, 65
	tuyere diameters: 16 materials: SiLi2mm screw conveyor: on

* engine power listed in percent of max. instead of volumetric/mass flow rate, as the latter differs depending on the material.

D.4 Benchmark of CFDEMcoupling

A brief benchmark study compares the performance of CFDEMcoupling to the performance of OpenFOAM 2.0.0.

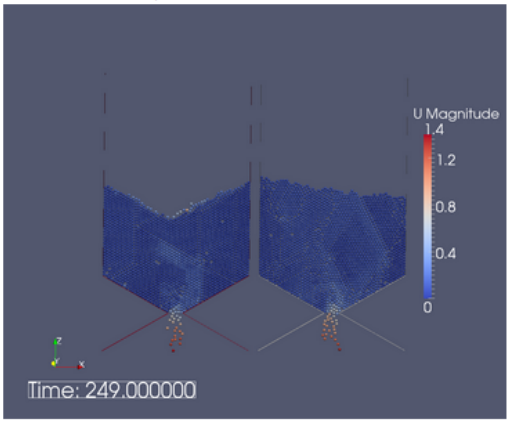
Involved in benchmarking and testing:
CD Lab on Particulate Flow Modelling, JKU Linz

We would like to motivate you to perform your own benchmarks and provide your results to the user community.

BENCHMARK CASES

Hopper Emptying
The setup of this test case was inspired by a tutorial case available in OpenFOAM(R)-2.0.0 by OpenCFD.
Although we tried to keep the cases as similar as possible, a direct comparison is neither desired nor reasonable. It can although give a rough estimate of the performance and results.

test case: 4-way coupled CFD-DEM
Hopper flow with LIGGGHTS (left), compared to original case performed with OpenFOAM(R)-2.0.0. As the models applied are not the the same, this results cannot be used for detailed comparison of the codes!!!



A rough estimate on the performance performed from our side:

TOOL	#procs	simTime [s]	execTime [s]	final#particles	fluid coupling	speedup
OF2.0.0	2	5	39900	2016	no	1
LIGGGHTS	2	5	1799	1858	no	~22
CFDEM	2	1	556	3070	4-way	~14

We would highly appreciate test and benchmark results from your side to verify or falsify above numbers.

NOTE:
OpenFOAM® is a registered trade mark of OpenCFD Limited
This offering is not approved or endorsed by OpenCFD Limited, the producer of the OpenFOAM®software and owner of the OpenFOAM® and OpenCFD® trade marks.

Fig. D.1: source: www.cfdem.com / Dec. 2, 2013

Bibliography

Aigner, A., Schneiderbauer, A., Kloss, C., and Pirker, S. (2013a). Comparison of simulation and experimental results of a simplified Jenike shear tester. In *6th International Conference on Discrete Element Methods (DEM6)*, Denver.

Aigner, A., Schneiderbauer, S., Kloss, C., and Pirker, S. (2013b). Determining the coefficient of friction by shear tester simulation. In *Proceedings of the 3th International Conference on Particle-Based Methods*, pages 335–342, Stuttgart.

Anderson, T. and Jackson, R. (1967). A fluid mechanical description of fluidized beds. *Ind. Eng. Chem. Fundam.*, 6:527–539.

Baxter, G. and Behringer, R. (1991). Cellular automata models for the flow of granular materials. *Physica D*, 51:465–471.

Benvenuti, L., Aigner, A., Queteschiner, D., Combarros, M., Pirker, S., and Kloss, C. (2014). Establishing predictive capability of DEM simulations: sliding and rolling friction coefficient of non-spherical particles. In *Proc. 10th Int. Conf. on CFD in Oli and Gas, Metallurgical and Process Industries*, Trondheim, Norway.

Beyer, R. and Leveque, R. (1992). Analysis of a one dimensional model for the immersed boundary method. *SIAM J. Numer. Anal.*, 29:332–364.

Biswas, A. (1981). *Principles of blast furnace iron making*. Cootha Publishing House.

Boycott, A. (1920). Sedimentation of blood corpuscles. *Nature*, 104:532.

Brauer, H. (1971). *Grundlagen der Einphasen- und Mehrphasenströmungen (transl: Fundamentals of single and multiphase flows)*. Sauerländer Verlag.

Buijtenen, M., Dijk, W.-J., Deen, N., Kuipers, J., Leadbeater, T., and Parker, D. (2011). Numerical and experimental study on multiple-spout fluidized beds. *Chem. Eng. Sci*, 66:2368–2376.

CFDEMcoupling (2012). CFDEM: CFDEM - Open Source CFD, DEM and CFD-DEM. URL http://www.cfdem.com.

Cundall, P. and Strack, O. (1979). A discrete numerical model for graular assemblies. *Geotechnique*, 21:47–65.

Di Renzo, A. and Di Maio, F. (2004). Comparison of contact-force models for the simulation of collisions in DEM-based granular flow codes. *Chem. Eng. Sci.*, 59:525–541.

Diaz-Goano, C., Minev, P., and K., N. (2003). A fictitious domain/finite element method for particulate flows. *J. Comput. Phys.*, 192:105–123.

Enwald, H., Peirano, E., and Almstedt, A.-E. (1996). Eulerian two-phase flow theory applied to fluidization. *Int. J. Multiphase Flow*, 22:21–66.

Ergun, S. and Orning, A. (1949). Fluid flow through randomly packed columns and fluidized beds. *Ind. and Chem. Eng.*, 41(6):1179–1184.

Evans, L. C. (1998). *Partial Differential Equations*. American Mathematical Society, Providence.

Felice, R. D. (1994). The voidage function for fluid-particle interaction systems. *Int. J. Multiph. Flow*, 20:153–159.

Ferziger, J. and Perić, M. (2002). *Computational Methods for Fluid Dynamics*. Springer, USA.

Gidaspow, D., Bezburuah, R., and Ding, J. (1992). Hydrodynamics of circulating beds: Kinetic theory approach. *Fluidization VII, Proceedings of the 7th Engineering Foundation Conference on Fluidization*, pages 75–82.

Glowinski, R., Pan, T.-W., Hesla, T., Joseph, D., and Periaux, J. (1998). A ficitious domain method with distributed Langrange multipliers for the numerical simulation of particulate flow. *Contemporary Mathematics*, 218:121–137.

Glowinski, R., Pan, T.-W., Hesla, T., Joseph, D., and Periaux, J. (1999). A distributed Lagrange multiplier/fictious domain method for the flows around moving rigid bodies: application to particulate flow. *Int. J. Numer. Meth. Fluids*, 20:1043–1066.

Glowinski, R., Pan, T.-W., Hesla, T., Joseph, D., and Periaux, J. (2001). A fictitious domain apprach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: application to particulate flow. *J. Comput. Phys.*, 169:363–426.

Goldstein, D., Handler, R., and Sirovich, L. (1993). Modeling a no-slip flow boundary with an external force field. *J. Comput. Phys.*, 105:354–366.

Goniva, C., Kloss, C., Hager, A., and Pirker, S. (2012). An Open Source CFD-DEM perspective. In *5th OpenFOAM Workshop*, Gothenburg. Chalmers.

Gupta, G., Rajneesh, V., Singh, V., Sarkar, S., and J.D., L. (2005). Mechanics of raceway hysteresis in a packed bed. *Metall. and Material. Transactions*, 36B:755–764.

Gupta, G. and Rudolph, V. (2006). Comparison of blast furnace raceway size with theory. *ISIJ Int. Journal*, 46:195–201.

Haeri, S. and Shrimpton, J. (2012). On the application of immerserd boundary, fictitious domain and body-conformal mesh methods to many particle multiphase flows. *Int. J. Multiphase Flow*, 40:38–55.

Heinrich, J. and Pepper, D. (1999). *Intermediate Finite Element Method: Fluid Flow and Heat Transfer Applications*. Series in Computational and Physical Processes in Mechanics and Thermal Sciences. Taylor & Francis.

Hilton, J. and Cleary, P. (2011). Three dimensional coupled discrete element-CFD modelling of high speed gas-particle raceway dynamics. In *Proc. 8th Int. Conf. on CFD in Oli and Gas, Metallurgical and Process Industries*, Trondheim, Norway.

Hilton, J. and Cleary, P. (2012). Raceway formation in laterally gas-driven particle beds. *Chem. Eng. Sci.*, 80:306–316.

Ishii, M. (1975). Thermo-fluid dynamic theory of two-phase flow. *Direction des etudes et recherches d'electricité de France*.

Issa, R. (1986). Solution of the implicit discretized fluid-flow equations by operator-splitting. *J. Comp. Phys*, 62:40–65.

Jackson, R. (1997). Locally averaging equations of motion for a mixture of identical spherical particles and a Newtonian fluid. *Chem. Eng. Sci.*, 52:2457–2468.

Jasak, H. (1996). *Error Analysis and Estimation for the Finite Volume Method with Applications to Fluid Flows*. PhD thesis, Imperial College of Science, Technology and Medicine, London, United Kingdom.

Jenike, A. (1964). *Storage and flow of solids*. Bulletin of the University of Utah. University of Utah.

Kärrholm, F. (2008). *Numerical Modelling of Diesel Spray Injection, Turbulence Interaction and Combustion*. PhD thesis, Chalmers University of Technology, Sweden.

Kim, D. and Choi, H. (2002). Laminar flow past a sphere rotating in the streamwise direction. *J. Fluid Mech.*, 461:365–386.

Kloss, C. (2011). *Numerical Simulation of Granular Flow and Coupled Fluid-Granular Systems*. PhD thesis, Johannes Kepler University Linz, Austria.

Kloss, C., Goniva, C., Hager, A., Amberger, S., and Pirker, S. (2012a). Models, algorithms and validation for Open Source DEM and CFD-DEM. *Progress in Comp. Fluid Dyn.*, 12:140–152.

Kloss, C., Goniva, C., and Pirker, S. (2012b). Open Source DEM and CFD-DEM with LIGGGHTS and OpenFOAM. In *Proc. Open Source CFD International Conference*, Munich.

Knabner, P. and Angermann, L. (2003). *Numerical Methods for Elliptic and Parabolic Partial Differential Equations*. Texts in Applied Mathematics. Springer.

Koch, D. and Hill, R. (2001). Inertial effects in suspension and porous media flows. *Ann. Rev. Fluid Mech.*, 33:619–647.

Koch, D. and Sangani, A. (1999). Particle pressure and marginal stability limits for a homogeneous monodisperse gas fluidized bed: kinetic theory and numerical simulations. *J. Fluid Mech.*, 400:229–263.

Lai, M. and Peskin, C. (2000). An immersed boundary method with formal second-order accuracy and reduced numerical viscosity. *J. Copmut. Phys.*, 160:705–719.

Landau, L. and Lifshitz, E. (1976). *Mechanics*. Course of Theoretical Physics. Elsevier.

LIGGGHTS (2012). LIGGGHTS: LAMMPS improved for general granular and granular heat transfer simulations. URL http://www.liggghts.com.

Liu, C., Zheng, X., and Sung, C. (1998). Preconditioned multigrid methods for unsteady incompressible flows. *J. Comput. Phys.*, 139:35–57.

Luding, S. (2008). Introduction to discrete element methods - basics of contact force models and how to perform the micro-macro transition to continuum theory. *European Journal of Environmental and Civil Engineering*, pages 785–826.

Luding, S., Clement, E., Blumen, A., Rajchenbach, J., and Duran (1994). Studies of columns of beads under external vibrations. *Phys. Rev. E*, 49:1634–1646.

MacDonald, J. and Bridgewater, J. (1997). Void formation in stationary and moving beds. *Chem. Eng. Sci.*, 52:677–691.

Malmberg, D., Hahlin, P., and Nilsson, E. (2007). Micowave technology in steel and metal industry, an overview. *ISIJ Int. Journal*, 47:533–538.

Matsui, Y., Tanaka, M., Sawayama, M., Imai, T., and Goto, A. (2005a). Analyses on dynamic solid flow in blast furnace lower part by deadman shape and reaceway depth measurement. *ISIJ Int. Journal*, 45:1445–1451.

Matsui, Y., Yamaguchi, Y., Sawayama, M., Kitano, S., Nagai, N., and Imai, T. (2005b). Analyses on blast furnace raceway formation by micro wave refelection gunned through tuyere. *ISIJ Int. Journal*, 45:1432–1438.

Mittal, R. and Iaccarino, G. (2005). Immersed boundary methods. *Ann. Rev. Fluid Mech.*, 37:239–261.

Mondal, S., Som, S., and Dash, S. (2005). Numerical predictions on the influences of the air blast velocity, initial bed porosity and bed height on the shape and size of raceway zone in a blast furnace. *Journal of physics D: Applied Physics*, 38:1301–1307.

Muguruma, Y., Tanaka, T., and Tsuji, Y. (2000). Numerical simulation of particulate flow with liquid bridge between particles (simulation of centrifugal tumbling granulator). *Powder Technology*, 109:49–57.

Mulero, A. (2008). *Theory and Simulation of Hard-Sphere Fluids and Related Systems*. Lecture Notes in Physics. Springer.

Natsui, S., Nogami, H., Ueda, S., Kano, J., Inoue, R., and Ariyama, T. (2011a). Simultaneous three-dimensional analysis of gas-solid flow in blast furnace by combining discrete element method and computational fluid dynamics. *ISIJ*, 51:41–50.

Natsui, S., Ueda, S., Nogami, H., Kano, J., Inoue, R., and Ariyama, T. (2011b). Dynamic analysis of gas and solid flows in blast furnace with shaft gas injection by hybrid model of DEM-CFD. *ISIJ*, 51:51–58.

Oesterlé, B. and Dinh, D. B. (1998). Experiments on the lift of a spinning sphere in a range of intermediate Reynolds numbers. *Experiments in Fluids*, 25:16–22.

OpenFOAM (2009-2011). OpenFOAM - The Open Source CFD toolbox. URL: http://www.openfoam.com.

Patankar, N. (2001). A formulation for fast computations of rigid particulate flows. *Center for turbulence research annual research briefs 2001*, pages 185–196.

Patankar, N., Singh, P., Joseph, D., Glowinski, R., and Pan, T.-W. (2000). A new formulation of the distributed Lagrange multiplier/fictitious domain method for particulate flows. *Int. J. Multiphase Flow*, 26:1509–1524.

Peskin, C. (1972). *Flow patterns around heart valves: a digital computer method for solving the equations of motion*. PhD thesis, Albert Einstein Coll. Med.

Radl, S., Radeke, C., Khinast, J., and Sundaresan, S. (2011). Parcel-based approach for the simulation of gas-particle flows. In *Proc. 8th Int. Conf. on CFD in Oil and Gas, Metallurgical and Process Industries*, Trondheim, Norway. SINTEF/NTNU.

Rajneesh, S., Sarkar, S., and Gupta, G. (2004). Prediction of raceways size in blast furnace from two dimensional experimental correlations. *ISIJ International*, 44:1298–1307.

Richardson, L. (1911). The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam. *Philos. Trans. R. Soc. Lond., A*, 210:307–357.

Rosato, A., Prinz, F., Strandburg, K., and Swendsen, R. (1986). Monte Carlo simulation of particle matter segregation. *Powder Technology*, 49(1):59–69.

Sakai, M. and Koshizuka, S. (2009). Large-scale discrete element method in pneumatic conveying. *Chem. Eng. Sci.*, 64:53–539.

Sakai, M., Takahashi, H., Pain, C., Latham, J.-P., and Xiang, J. (2012). Study on large-scale discrete element model for fine particles in a fluidized bed. *Advanced Powder Technology*, 23:673–681.

Sakai, M., Yamada, Y., Shigeto, Y., Shibata, K., Kawasaki, V., and Koshizuka, S. (2010). Large-scale discrete element modeling in a fluidized bed. *Int. J. Numer. Meth. Fluids*, 64:1319–1335.

Sakamoto, H. and Haniu, H. (1990). A study on vortex shedding from spheres in a uniform flow. *J. Fluids Eng.*, 112:386–392.

Sarkar, S., G.S., G., Litster, J., Rudolph, V., White, E., and Choudhary, S. (2003). A cold model study of the raceway hysteresis. *Metall. and Material. Transactions*, 34B:183–191.

Sarkar, S., Gupta, G., and Kitamura, S. (2007). Prediction of raceway shape and size. *ISIJ International*, 47:1738–1744.

Sastry, G., Gupta, G., and Lahiri, A. (2003). Void formation and breaking in a packed bed. *ISIJ Int. Journal*, 43:153–160.

Schulze, D. (2006). Stresses in silos. URL http://www.dietmar-schulze.de/spannepr.html.

Shirgaonkar, A., MacIver, M., and Patankar, N. (2009). A new mathematical formulation and fast algorithm for fully resolved simulation of self-propulsion. *J. Comput. Phys.*, 228:2366–2390.

Spurk, J. (1992). *Dimensionsanalyse in der Strömungslehre (transl: Dimensional analysis of fluid dynamics)*. Springer.

Spurk, J. and Aksel, N. (1997). *Fluid Mechanics*. Springer.

Sterneland, J., M.A.T., A., and Joïsson, P. (2003). Comparison of iron ore reduction in experimental blast furnace and laboratory scale simulation of blast furnace process. *Ironmaking and Steelmaking*, 30:313–327.

Stockie, J. and Wetton, B. (1998). Analysis of stiffness in immersed boundary method and implications for time-stepping. *J. Comput. Phys.*, 154:41–64.

Streeter, V. and Wylie, E. (1985). *Fluid Mechanics*. Eighth edition. McGraw-Hill.

Sundaresan, S., Radl, S., Milioli, C., and Milioli, F. (2013). Coarse-grained models for momentum, energy and species transport in gas-particle flows. In *The 14th International Conference on Fluidization – From Fundamentals to Products*.

Ueda, S., Natsui, S., Nogami, H., Yagi, J., and Ariyama, T. (2010). Recent progress and future perspective on mathematical modeling of blast furnace. *ISIJ International*, pages 914–923.

Uhlmann, M. (2005). An immersed boundary method with direct forcing for the simulation of particulate flows. *J. Comput. Phys.*, 209:448–476.

Umekage, T., Yuu, S., and Kadowaki, M. (2005). Numerical simulation of blast furnace raceway depth and height, and effect of wall cohesive matter on gas and coke particle flows. *ISIJ*, 45:1416–1425.

Umkage, T., Kadowaki, T., and Yuu, S. (2007). Numerical simulation of effect of tuyere angle and wall scaffolding on unsteady gas and particle flows including raceway blast furnace. *ISIJ*, 47:659–668.

van Wachem, B., van Schouten, J., van den Bleek, C., Krishna, R., and Sinclair, J. (2001). Comparative analysis of dense gas-solid systems. *AIChE Journal*, 45(5):1035 –1051.

Verein Deutscher Ingeneure, V. (2010). *VDI Heat Atlas*. 2nd Edition. Springer, Heidelberg.

Verlet, L. (1967). Computer "experiments" on classical fluids. I. thermodynamical properties of lennard-jones molecules. *Phys. Rev.*, 159:98–103.

Weik, M. (1961). The ENIAC story.

Wen, C. and Yu, Y. (1966). Mechanics of fluidization. *Chem. Eng. Process Symposium*, Series 62:100–111.

Wesseling, P. (2001). *Principles of Computational Fluid Dynamics*.

Wong, M. W. (1999). *An Introduction to Pseudo-Differential Operators*. World Scientific.

Xu, B. and Yu, A. (1998). Numerical simulation of the gas-solid flow in a fluidized bed by combining discrete particle method with computational fluid dynamics. *Chem. Eng. Sci.*, 52:2785–2809.

Yu, A. and Xu, B. (2003). Particle-scale modelling of gas-solid flow in fluidization. *J. Chem. Technol. Biotechnol.*, 78:111–121.

Yu, Z. and Shao, X. (2007). A direct-forcing fictitious domain method for particulate flows. *J. Comput. Phys.*, 227:292–314.

Zhou, Z., Kuagn, S., Chu, K., and Yu, A. (2010). Discrete particle simulation of particle-fluid flow: model formulations and their applicability. *J. Fluid Mech.*, 661:482–510.

Zhu, H., Zhou, Z., Yang, R., and Yu, A. (2007). Discrete particle simulation of particulate systems: Theoretical developments. *Chem. Eng. Sci.*, 62:3378–3396.

List of Figures

List of Tables

Curriculum Vitae Alice Hager

Contact Information

Name	Dipl.-Ing. Alice Hager
Address	Dinghoferstraße 48 4020 Linz Austria
Cell phone	+43 680 3060009
Email	alice.m.hager@gmail.com

Education

October 2010 - now	Doctoral program in engineering sciences Topic: CFD-DEM on multiple scales Main focus: development of numerical models for the simulation of particle laden fluid flows for problems with both resolved and unresolved bodies Christian Doppler Laboratory on Particulate Flow Modelling Johannes Kepler University, Linz (JKU)
February 2009 – July 2010	Master studies Industrial Mathematics (passed with distinction) Thesis: Parameter choice criteria for the regularization of inverse problems Main focus: Numerics, Financial Mathematics JKU
October 2005 – January 2009	Bachelor studies Technical Mathematics Bachelor thesis in algebra and mathematical modelling of economic processes JKU
September 1997 – June 2005	Bundesgymnasium und Bundesrealygymnasium Steyr Werndlpark Matura (passed with distinction)

Professional experience

January 2011 - now	Research assistant at Christian Doppler Laboratory on Particulate Flow Modelling Tasks: development, implementation and application of numerical models in CFDEMcoupling (www.cfdem.com), simulations with OpenFOAM® and validation of the results by comparison to the literature, analytics or experimental data Co-host of international software trainings
December 2010	Simulations in Fluent Institute of Fluid Mechanics and Heat Transfer (JKU)

International experience

	Talks at international conferences (cf. list of publications)
1-15 September 2010	NUMAP – FOAM, OpenFOAM Summer school Zagreb, Croatia
September 2003	Student exchange Le Mans, France

Internships

October 2010 – November 2010	Applied Computational Fluid Mechanics d-tech Steyr
August 2009 – September 2009	Optimization of computational models and structures for numerical analysis d-tech Steyr
Further occupations	Retail, gastronomy (Summer 2005 – 2008) Tutor for mathematics, English, Latin, physics Fair representative for Johannes Kepler University

Special Knowledge

First language	German
Further languages	English (fluent) French (conversant) Spanish (basic knowledge)
Software knowledge	OpenFOAM®: Simulations and Programming CFDEMcoupling LIGGGHTS Fluent® Gambit, Ansa Paraview C++, Prolog Matlab®, Mathematica Microsoft Office (Word, Excel, PowerPoint), LaTex
Driver's license	Class B Touring Motor Glider License

Date and Place

Linz, April, 2014