Development and Verification of a Novel Lagrangian, (Non-)Spherical Dirt Particle and Deposition Model to Simulate Fluid Filtration Processes using OpenFOAM®

DI Gernot Boiger
ICE Strömungsforschung GmbH, Austria

1st Advisor:
A.o. Univ. Prof. DI Dr. techn. Wilhelm Brandstätter
Department of Petroleum Engineering
University of Leoben, Austria

2nd Advisor:
Univ. Prof. DI Dr. mont. Werner Kepplinger
Institute of Process Technology for Industrial Environmental Protection
University of Leoben, Austria

Leoben, Oktober 2009

DEVOTEMENT/ACKNOWLEDGEMENTS

This work is dedicated to my beloved family and my girlfriend, because sometimes *thank you* is just not good enough!

I am particularily indebted to Ao. Univ. Prof. DI Dr. techn. Wilhelm Brandstätter without whom this work would not have been possible in the first place and whom I have come to estimate as a great mentor. Many thanks to the colleagues and friends of ICE Strömungsforschung GmbH. They made the office a creative and fun place to work at!

MAHLE Filtersystems GmbH. provided substantial financial support, for which I am very grateful.

Special, personal thanks to Ao. Univ. Prof. DI Dr. techn. Christian Weiss for being an advisor, a supporter and a friend. My additional thanks to o. Univ. Prof. DI Dr. mont. Werner Kepplinger for supervising this thesis.

I hereby declare in lieu of oath, that myself composed this PhD thesis and that the work contained therein is my own, except where stated.

DI Gernot Boiger

Abstract

The development of new, high performance filter media for Automotive oil filtration is an important issue for car suppliers. However, as of now knowledge of decisive, micro scale filtration processes is still limited and the relative importance of many static and dynamic process parameters remains unclear. This work represents an extensive attempt to push the field of fluid filter fibre design forward, away from being a strongly experimental based, trial and error scheme. Thus a micro scale, deterministic filtration solver has been developed using the Open Source, C++ based, computational fluid dynamics tool box OpenFOAM®. The new simulation tool models fluid, fibre and dirt particle interactions as well as dirt particle deposition processes within the framework of realistically reconstructed, microscopic fibre geometries. By statistically averaging the micro scale calculations, the filtration solver can derive some of the most important, macroscopic filtration parameters, such as pressure drop, particle penetration depth and filter fibre efficiency. While other, related publications [1, 30] deal with the simulation of fibre deformation effects, this thesis presents the novel Eulerian – Lagrangian dirt particle and deposition model behind the filtration solver. The particle model is capable of handling, spherical and non-spherical, discrete dirt particles as well as their relevant, dynamic interactions with the fibres, the fluid and among each other. Single particle hydrodynamics are resolved by several fluid calculation cells.

The software has already proven to be useful far beyond the field of filtration application and thus represents a completely new tool for Lagrangian, non-spherical particle simulation. In the course of this work the model is scientifically laid out and its physical as well as numerical background is explained.

In order to qualitatively and quantitatively validate the results, an extensive experimental set up has been created and a semi-empirical validation scheme has been devised. In addition to that a novel macroscopy method to visualize and digitally evaluate three dimensional dirt particle distributions in filter fibre samples can be presented.

To conclude, some revealing examples of solver functionality, plausibility and possible future application are given.

New insights provided by this development can now lead to a much better understanding of the filtration process as a whole and might define the direction an efficient, future, material development procedure will have to take.

Kurzfassung

Die Entwicklung neuer Hochleistungsfiltermedien zur Ölfiltration ist im Bereich der automobilzuliefernden Industrie ein brisantes Thema. Trotzdem ist bisher das Wissen um entscheidende, mikroskopische Filtrationsprozesse eingeschränkt und die relative Wichtigkeit statischer und dynamischer Prozessparameter unklar. Diese Arbeit stellt den umfassenden Versuch dar, die Möglichkeiten der Filterfaserentwicklung bedeutend zu erhöhen. Rein experimentelle Versuchs- und Irrtumsverfahren, sollen dabei durch computergestützte, zielgerichtete Entwicklung abgelöst werden. Aus diesem Grunde wurde ein, auf mikroskopischer Ebene arbeitender, deterministischer Filtrationssolver auf Basis des frei verfügbaren, thermofluiddynamischen Simulationspaketes OpenFOAM® entwickelt und programmiert.

Das neu entwickelte C++ Programm ist in der Lage Interaktionen von Fluid, Fasern und Schmutzpartikeln sowie deren Ablagerungsmechanismen innerhalb der Umgebung realistisch rekonstruierter, mikroskopischer Fasergeometrien hochdetailliert zu modellieren. Durch statistische Mittelung der mikroskopischen Rechenergebnisse kann dadurch auf einige der wichtigsten Prozessparameter der Filtration rückgeschlossen werden, zum Beispiel: Druckverlust, Partikeleindringtiefe und Filterfasereffizienz.

Während sich andere, verwandte Publikationen [1, 30] mit der Simulation von Faserdeformationseffekten beschäftigen, präsentiert diese Arbeit das neue Euler–Lagrangsche Partikelmodell hinter dem Filtrationssolver. Das Modell kann das Verhalten diskreter, sphärischer und nicht-sphärischer Schmutzpartikel, wie auch deren dynamische Interaktionen mit dem Fluid, den Fasern und untereinander berechnen. Die Hydrodynamik einzelner Partikel wird erstmals durch mehrere Fluidberechnungszellen aufgelöst. Die Software hat sich inzwischen auch schon in, über die Filtration hinausgehenden Anwendungsbereichen bewährt und stellt damit ein neues, Lagrangsches, nicht-sphärisches Partikelsimulationswerkzeug dar. Im Zuge dieser Arbeit wird das Modell wissenschaftlich aufbereitet und in seinen physikalischen wie numerischen Grundlagen Schritt für Schritt erläutert.

Um Simulationsergebnisse qualitativ und quantitativ zu validieren wurde außerdem eine umfassende Versuchsanordnung entwickelt und ein semi-analytisches Validationsschema hergeleitet. Zusätzlich kann hiermit die Erfindung einer neuartigen Makroskopiemethode, um drei dimensionale Schmutzpartikelverteilungen erfassen, digitalisieren und auswerten zu können, präsentiert werden.

Abschließend werden einige Anwendungsbeispiele des Simulators angeführt und mögliche, zusätzliche Anwendungsgebiete aufgezeigt. Die neuen Erkenntnisse, welche nun durch diese Entwicklung gewonnen werden können, werden zu einem besseren Verständnis von Filtrationsprozessen führen. Darüber hinaus könnte durch diese Arbeit die Richtung eines effizienten, zukünftigen Filtermaterialentwicklungsablaufes entscheidend geprägt werden.

Table of Contents

1 Introduction

High pressure and shear forces as well as cavitation effects close to the engine crank case cause local material rapture and hence, the creation of metallic particles. To avoid an accumulation of these particles and eventually damaging of bearings or piston/liner assemblies in the engine lubrication circuits, filtration elements have to be installed. Due to increasing maintainance intervalls, the Automotive industry currently undertakes considerable development efforts to increase the performance of existing filter elements and to create new, improved filter fibre materials. However, as of now the knowledge about many aspects of fluid filtration is still limited and the dynamic dependence of decisive micro scale filtration effects remains unclear. Any fluid filtration system consists of three obvious components to be considered: The incoming, highly viscous oil, the complex, interwoven fibre structure, which deforms due to the oil flow and the dirt particles which get entangled in the fibre and which, over filter life time, accumulate there. The latter effect leads to gradual, but macroscopic changes of important process parameters. Some of those parameters are pressure drop Δp_f over the filter element, filter permeability α_f, relative particle penetration depth P and filter fibre efficiency E. This basic situation unfolds remarkable complexity once the dynamic interaction between the individual components is considered:

The fluid hits the fibres and according to fluid-continuum mechanics, exerts pressure- and shear forces on the material. As a consequence the fibre structure deforms, following the laws of structural mechanics. The deformation in turn affects the flow pattern. Then the particles come in. According to their individual ratio between inertia and viscous forces, the particle relaxation time, they are more or less readily dragged towards- and into the filter medium. The dirt particles either hit or pass the fibres and either stick there, get sieved out or get blown off again. As more and more particles get entangled in the structure, their effect on the fluid flow field becomes more and more pronounced, thus gradually causing the pressure drop to rise. In addition to that, more and more oncoming particles get filtered out because of the *cake filtration effect*, being based on particles blocking each others flow path.

Due to the complexity of the issue, a change in fibre morphology (i.e. pore size diameter) can not be linearly linked to i.e. filter fibre efficiency, because it influences the whole hydrodynamic situation. As a consequence the field of fluid filter fibre design still relies on inefficient trial and error methods to create new materials. Therefore each development task still requires time consuming, costly experimental runs. Consequentially the motivation arises to enlist the aid of computational fluid dynamics (CFD).

This work represents an extensive attempt to create a tool which can increase the understanding of filter effects and dynamic parameter dependencies by means of computational engineering and simulation technology. A detailed, deterministic calculation model which simulates the most important filtration effects on a microscopic level has been created. The microscopic model results can be statistically averaged to yield the macroscopic parameters pressure drop, particle penetration depth, filter fibre efficiency and permeability. Figure 1 sketches out the basic concept behind this novel scheme.

In a first step, computer tomographic (CT) scans are conducted on “real life” filter fibre samples. The CT output data is compiled in stacks of two dimensional (2D) gray scale images of the fibre. Then the data is read in, -digitalized, and processed to a full 3D reconstruction of the microscopic filter element. The 3D object is then automatically meshed by a structured grid generator, so that the geometry can be utilized as boundary framework for oncoming CFD calculations. This is where the result of the main development task comes in. A CFD tool, designed and programmed in order to resolve the dynamic filtration situation for a user definable set of process variables, within the reconstructed fibre element. Produced simulation results can then be used to estimate the performance and suitability of the tested medium.

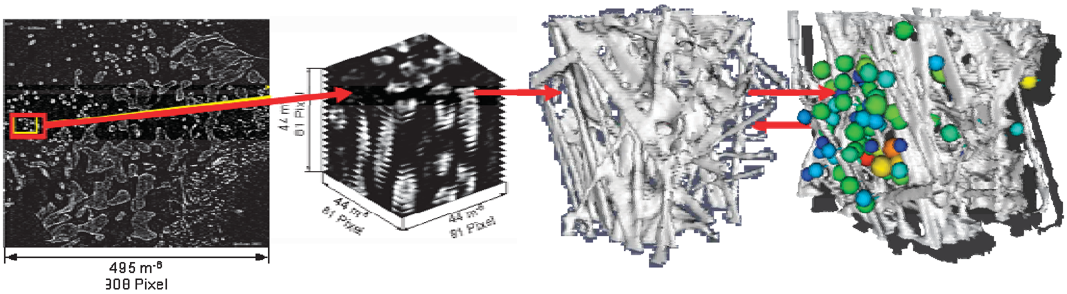


Figure 1: Sketch of the principle simulation concept. CT scans yield stacks of 2D grey scale images (left), which are transferred into 3D reconstructions of the fibre (middle). The 3D images are meshed and provide the geometry for the CFD filtration solver to be created (right).

Constant checks for result plausibility and validation have to be integral parts of any serious CFD development effort. In order to qualitatively and quantitatively validate the results, an extensive experimental set up has been created and a semi-empirical validation scheme has been devised. Figure 2 gives an overview of the underlying development scheme which links the experimental- and the simulation side. The application of this method leads to a continuous adjustment and improvement of the CFD model, according to the equivalent, experimental results.

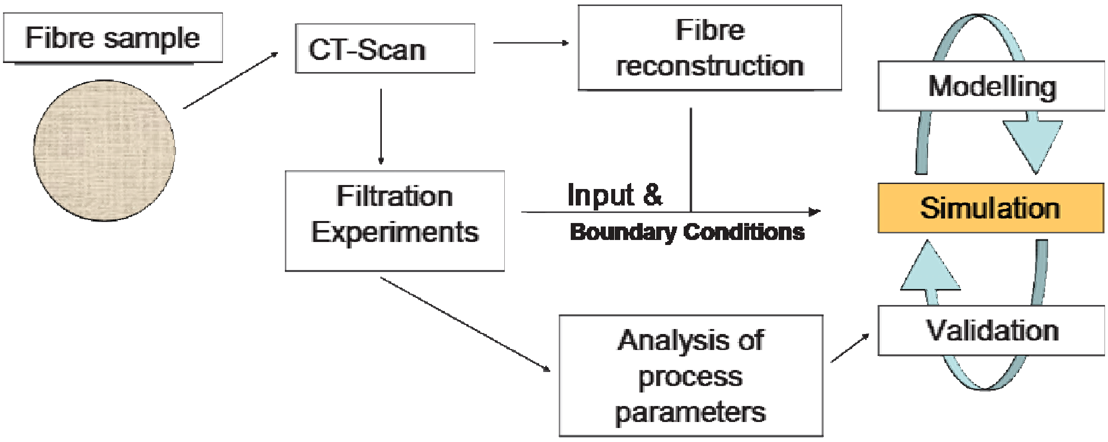


Figure 2: Experimental- and CFD development scheme.

From the beginning the development project was parted into four major working areas, as seen in Figure 3:

- Digital Fibre Reconstruction (DFR) from CT information, as well as the suitable meshing of the 3D data.

- Creation of a Fluid Structure Interaction (FSI) tool in order to handle the fibre deformation effects under the influence of fluid flow.

- Development of a detailed, dirt particle- and deposition model, capable of simulating spherical and non-spherical dirt particle behavior in and outside of the micro scale fibre vicinity.

- Validation of simulation results. Devise of an appropriate, experimental set up to verify solver functionality and to provide additional insight into filter fibre behavior and characteristics.

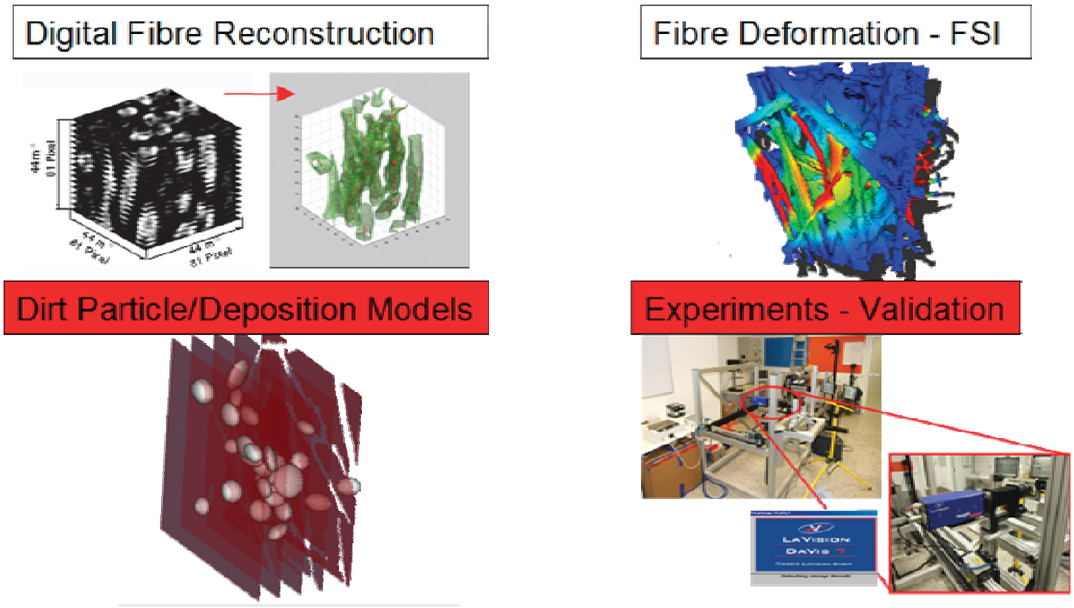


Figure 3: Overview of the four major areas of development behind the filtration solver project. The development of suitable dirt particle and deposition models as well as the experimental- and validation effort are at the focus of this thesis.

While other, related publications [1, 11] extensively deal with the creation of the FSI tool and the DFR utility, this thesis will only briefly discuss those two subjects. For it is mainly focused on the two latter areas of development:

- The introduction of a novel, deterministic Eulerian–Lagrangian [25], large spherical and non-spherical dirt particle and deposition model.
- The experimental validation of the numeric calculations.

The Open Source, finite volume based, CFD tool box OpenFOAM® (Open Field Operation And Manipulation) [36 - 39] was selected as environment for the development of the filtration solver. Two main reasons lead to this choice:

- OpenFOAM® is based upon the programming language C++ and therefore features a completely modular programming structure. This corresponds perfectly with the modular set up of the entire research project.

- The full source code is open to be altered as required by the developer. This provides a high degree of versatility and options, only limited by the imagination and capability of the user.

In the following a short review on the individual chapters of this thesis is given: Chapter 2 presents some basic fundamentals behind the work. Initially the simulation tool boxes OpenFOAM® and MatLab® as well as the interfacing software LabVIEW® are briefly introduced. Then the prevailing physical conditions as well as resulting model simplifications are discussed. In a next step the FSI tool and the DFR utility are described in short.

A main part of the entire development effort behind the project has gone into the creation of the novel, non-spherical dirt particle solver. Therefore chapter 2 also presents *three* important reasons as to *why* the consideration of particle shape effects in filtration simulation is believed to be imperative:

- The particle-inertia-to-fluid force ratio, represented by the particle relaxation time, is strongly shape dependent.

- Particles with small, angular particle relaxation times experience the non-spherical particle *slip effect*.

- Particles with large, angular particle relaxation times experience the non-spherical particle *bulk effect*.

Three fundamental concepts, which form the roots of the particle model, are discussed in chapter 3: the *Lagrangian* simulation approach, the *force-to-motion concept* and the *large particle model*.

Chapter 4 is the core part of this thesis and is about the intrinsics of the (non-) spherical dirt particle and deposition solvers. It is split in two. Chapter 4.1 presents the first, original version of the Lagrangian dirt particle solver. This first program is merely capable of handling spherical particles, but already contains many essential features. The refined, non-spherical model is described and laid out in high detail in the course of chapter 4.2. Basic, newly developed, non-spherical modeling concepts, as well as force-interaction implementations and drag/lift force calculation schemes are discussed. Benchmark examples of solver functionality are constantly given.

The decisive problem of numerical instability due to *Explicit Euler*, temporal particle movement discretization is addressed and amended in chapter 5. A possible solution, based on the development of a well founded, adaptive time stepping scheme is given.

Solver extension modules, namely the *bacteria module* and the *electro static module* are at the focus of chapter 6. In this context, the easy expandability of the source code is pointed out.

Chapter 7 provides an insight into the work-flow behind the code and into the C++ software design pattern of the relevant particle solver classes as well as into their embedding within the OpenFOAM® program structure. A complete description of all particle-solver specific, user-definable input parameters is given too.

Chapter 8 deals with the entire issue of solver validation and experimental verification and points out several significant developments in the field of dirt particle distribution detection. A macroscopic method for 3D digitalization and visualization of test particle distributions in filter fibre samples is presented and a newly developed, MatLab® based, reconstruction algorithm is described and verified.

Chapter 9 brings concrete examples of solver application and demonstrates how filter fibre engineering might look like in the near future.

The concluding chapter 10 sums up the main development achievements since 2005 and gives an outlook towards potential, future extensions of the solver.

2 Fundamentals and Modelling Task

Sub chapters 2.1 to 2.5 describe several aspects of the basic, physical and software-related fundament behind this thesis and the (non-)spherical dirt particle solver in particular. Thereby all general explanations are intentionally kept as brief as possible in order to focus on domestic development successes, described in oncoming chapters. In addition to that, sub chapter 2.6 justifies the extensive effort which was invested into the creation of the non-spherical particle model.

2.1 OpenFOAM®

The entire CFD related software development behind this thesis was conducted within the framework of the Open Source CFD package OpenFOAM®. Since various other sources (see [36-39] and [44]) describe OpenFOAM® very thoroughly, only a minimal introduction will be given in this context.

OpenFOAM® is an accumulation of flexible C++ modules that constitute a tool to solve any system of partial differential equations by applying finite volume numerics [87]. Fluid flow equations are thereby solved by a robust, implicit, pressure-velocity, iterative procedure [36]. Based on this framework, the CFD toolbox can simulate a wide variety of complex fluid flow problems in engineering mechanics. It provides a selection of solvers, utilities and libraries.

- Solvers are used for the actual simulation. They can be specifically selected according to the governing physics of the problem.

- Utilities fulfill various pre- and post processing tasks from output data processing to mesh manipulation.

- Libraries are repositories of function related software tools that can be accessed by solvers and utilities.

The source code of the program has been made *Open Source* and thus is publicly available to anyone. Every aspect of the software can be altered as required by the user. Constantly improved and updated versions of the OpenFOAM® package as well as additional utilities can be downloaded at Opencfd.co.uk [36], CFD-online.com [37] or at Openfoam-extend.svn.sourceforge.net [44].

Due to the *Open Source* character of the software framework any development created by using OpenFOAM® is subject to the General Public License (GNU) [45].

All numerical calculations within this work were conducted by OpenFOAM® version 1.4.1 which also served as programming framework that was extended as required. The meshing was conducted via the commercial FLUENT® mesh generator GAMBIT® [89] or via self written meshing utilities (see chapter 2.5). All results were post processed and visualized with the Open Source visualization tool ParaView version 3.2.0 [90] by Kitware®.

2.2 MatLab®

A large part of the experimental-result-evaluation related software development behind this thesis was conducted within the framework of the commercial, numerical computing environment MatLab® by *The Math Works* [88]. MatLab® is not just a mathematics tool but also a programming language. Some of the main capabilities of the software are matrix manipulation, data visualization and the possibility to create user interfaces. In addition to that MatLab® is equipped with a wide range of extension toolboxes. The graphics toolbox was essential for the programming of the optical evaluation algorithm (see chapter 8.3.3).

2.3 LabVIEW®

The software based interfacing between laboratory equipment and the control unit was realized with LabVIEW® (Laboratory Virtual Instrumentation Engineering Workbench) from *National Instruments* [89]. LabVIEW® is a platform and development environment for the visual programming language G. In terms of data acquisition, instrument control and industrial automation LabVIEW® has become a standard piece of software.

The control interface for correlating laser, drive set and digital camera of the particle distribution detection facility (see chapter 8.3.2) was created by using LabVIEW® version 7.5 whereas the data acquisition and control software for the oil-fibre test facility (see chapter 8.3.1) was set up under LabVIEW® version 8.1.

2.4 Prevailing Physical Conditions in Fluid Filtration

The prevailing physical situation in automotive oil filtration is characterized by the interaction of the three main components: fluid, filter fibre and dirt particles. It can be described as follows:

An oil pump pushes the lubricant towards the filtration device. The motor oil usually consists of a base component and of up to 25% of additives. While paraffin mineral oils or hydrocrack oils were mostly used as base component in the past, fully synthetic base components like polyether, silicones or synthetic hydro carbons are becoming increasingly important today [70]. In this work, representative oil properties are chosen. Thus the highly viscous, Newtonian oil fluid stream is stated to feature a kinematic fluid viscosity of $\nu_f \sim 2*10^{-5} m^2/s$ and fluid density of $\rho_f \sim 850 kg/m^3$.

The fluid stream comes in at relatively slow flow velocities of $u_f < 0.1 m/s$, hits the filter, exerts pressure and shear forces on the fibres and deforms them according to the laws of structural mechanics. Commonly used oil filter fibre materials are cellulose or glass fibre. In recent years polyester and polypropylene components have been inserted as well [70]. Single fibre diameters range from 5µm up to 50µm and usual pore size diameters range below 100µm. Because of the microscopic geometry range, the high viscosity and the low flow velocities, the local Reynolds numbers Re in the fibre vicinity are expected to be mostly below 1, but surely below 10. Accordingly the occurring particle Reynolds numbers Re_p are also expected to be well below 1, which means that calculations of particle hydrodynamics will have to be valid within the *Stokes flow* regime. Furthermore Knudsen numbers Kn are well below 0.015. Thus continuum equations are valid and the consideration of molecular fluid diffusion effects is unnecessary. The influence of diffusive motion on particle movement can be estimated via the Péclet number Pe. The Péclet number is a ratio of particle advection and diffusion effects. In the case of fluid filtration it has been found to be well above $5*10^7$ [70]. As a consequence, particle diffusion behaviour is negligible and Newtonian mechanics suffice to describe particle movement.

The bottom line at this point is that certain simplifications can be made in the modelling. A simple, incompressible, laminar and isothermal fluid solver can handle the situation. Thus the CFD fluid simulation relies on the standard solution of the temporarily and spatially discretized, incompressible *Continuity*- (Equ.1) [47] and *Navier Stokes* equation (Equ.2) for Newtonian fluids [46].

$$\nabla \cdot \vec{u}_f = 0 \tag{1}$$

$$\frac{\partial}{\partial t}\left(\vec{u}_f\right) + \left(\vec{u}_f \cdot \nabla\right)\vec{u}_f = \frac{1}{\rho_f}\left(-\nabla p + \mu_f \nabla^2 \vec{u}_f\right) + S_I \tag{2}$$

Whereby t stands for time, p is the pressure field, μ_f is the dynamic fluid viscosity and S_I is the source term for volumetric forces such as gravity. S_I can also work as the momentum source term for small, two-way coupled particles.

Certainly the prevailing physical situation is dominated by the presence of dirt particles and their interactions with the surroundings. The oil flow is laden with sparse accumulations of steel (ρ_p~7800kg/m³), quartz (ρ_p~2650kg/m³), and soot (ρ_p~1800kg/m³) particles, ranging from 5µm to 100µm in mass equivalent, spherical diameter [70]. They hit or pass the filter, some stick to the fibre due to adhesion, some get sieved out by the fibre itself or because of the *cake filtration effect* and some get blown off again. A slow particle accumulation takes place, which causes local plugging of the flow. Over filter life time the microscopic particle deposition effects give rise to macroscopic changes of filter characteristics, such as porosity, permeability, pressure drop, filter efficiency and particle penetration depth. In order to simulate the encountered phenomena, the implementation of a detailed particle model is necessary and the calculation of particle force interactions is essential. The effects with dominating influence on the changes of filter characteristics are fluid–particle, particle-fibre and particle–particle interactions.

2.5 Fibre Reconstruction and Fluid Structure Interaction

The ability to realistically model micro scale filtration processes in filter fibre materials is in large part based upon the realistic reconstruction of micro scale filter fibre geometries. Within the context of the development effort behind this work, a sophisticated method to digitally recreate real geometries is applied. In a first step, computer tomographic scans (CT) are conducted on the fibre material to be investigated. The data yielded by the CT scans are stacks of 2D grey scale images seen in Figure 4 (left). MatLab® based reconstruction algorithms have been programmed in order to process the CT data. The picture stacks can be uploaded and the individual slices are then analyzed. Local picture areas of higher grey scale intensities are recognized as fibre regions which can be clearly distinguished against the low-intensity background. Identified fibre slices are then quantified, their pixel area is calculated and their local centres and radii are determined. By applying a skeleton [48] algorithm the centres of consecutive fibre slices are interconnected to constitute the basic, local fibre framework. By applying the calculated radius information attached to each centre point, the actual fibre structure is recreated as 3D, digital data matrix. It can be visualized as seen in Figure 4 (right).

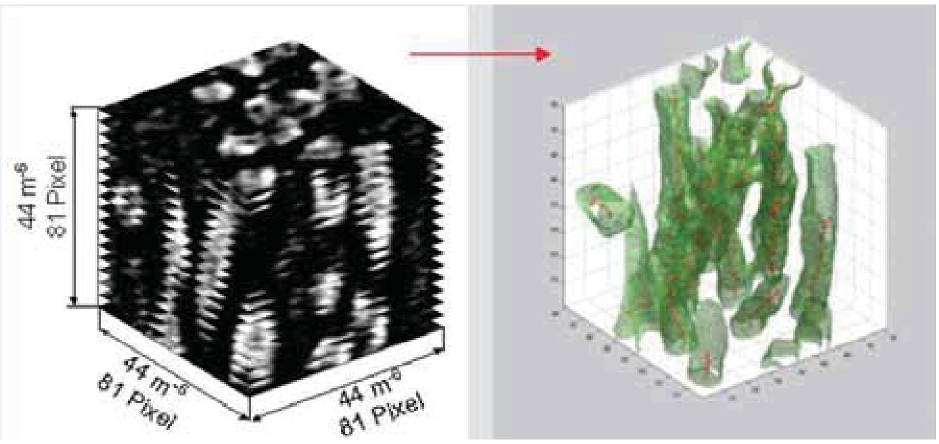


Figure 4: Fibre reconstruction and digitalization by MatLab® utilities. Stacks of grey scale images (left) out of CT scans are transferred to fully digitalized data matrices (right).

In a next step the digital data is automatically discretized into a structured, hexahedral grid mesh with a user definable cell-spacing-to-pixel ratio. This means that, if the CT scan resolution can be kept constant, a uniform spatial discretization rate for any filter fibre simulation can be guaranteed. Thereby one of two modes of spatial resolution can be chosen: Either the finer mode which features a spatial resolution of 0.625Pixel/µm or the coarser mode which features a resolution of 0.313Pixel/µm. The reconstruction utility yields perfectly interfaced grids, of both the fluid and the solid region of the fibre sample. Figure 5 shows an exemplary, structured, micro scale fibre grid mesh.

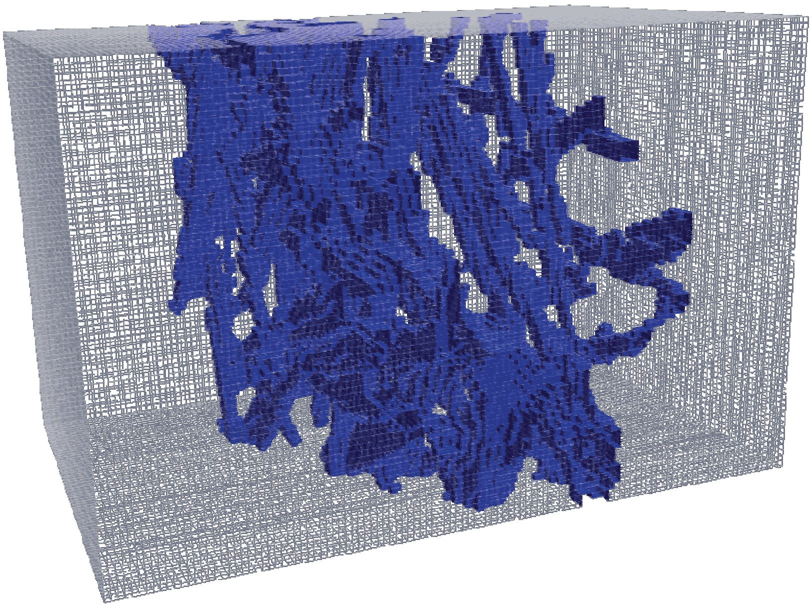


Figure 5: Filter fibre sample discretized into a structured fluid- and solid hexahedral grid mesh. Dimensions: 200µm*200µm*300µm. Number of cells: ~6.0*10^5.

The prepared, structured grid meshes serve as geometry boundary conditions for the simulator to be developed.

For quite some time, deformation effects have been suspected to have significant impact on the filter characteristics of a fibre. Therefore fluid structure interaction phenomena were included into the modelling. An extensive, detailed Fluid Structure Interaction utility was programmed. It features a stiff, explicit coupling between the fluid and the solid phase. A speciality of the code is that it uses only one, single finite volume solver to handle the governing fluid dynamics as well as the structural mechanics and deformation on the solid side. Figure 6 sketches out the basic scheme behind the FSI solver. Within every time loop the Navier Stokes equations along with the Continuity equation are solved in a PISO [91] loop to yield the fluid pressure- and velocity field. The pressure and the surface normal gradient of the fluid velocity field are then used to calculate pressure- and shear stresses respectively. The stress terms are explicitly passed as boundary conditions for the fibre. In a next step the Hook’s law structural mechanics equations [11] are solved on the solid side of the dual fluid/solid mesh. Then the local displacement values for the solid region are written out and the mesh is moved accordingly. In order to reduce numerical instabilities, a *semi-implicit implementation switch* can be used. Thus the fluid solution is recalculated within the same time step and compensates for changed flow geometries.

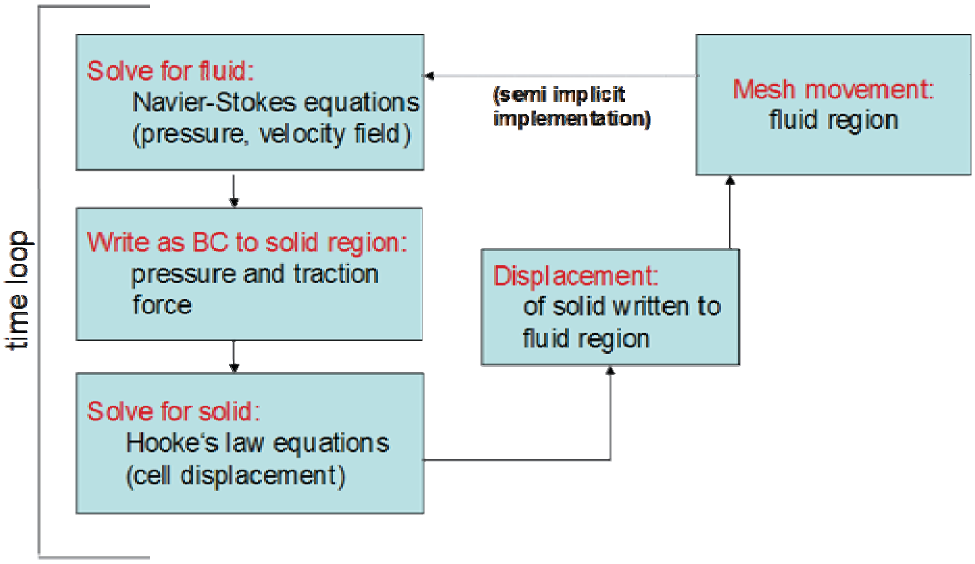


Figure 6: Basic solver scheme of the FSI tool for modelling deformation effects of the filter fibre structure under the influence of the oil stream.

Figure 7 shows an exemplary calculation result where the FSI utility has been applied on a realistically reconstructed piece of filtration fibre geometry which is hit and deformed by oil flow.

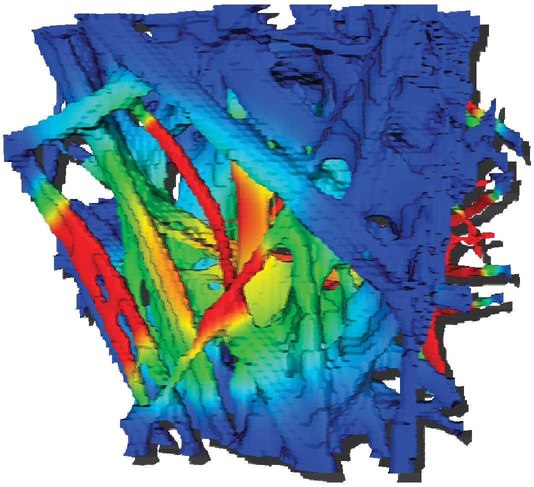


Figure 7: Filter fibre material, deformed by oil flow. Compact fibre regions show less deformation (blue), while thinner fibres are deformed more strongly (red).

Detailed descriptions regarding the back ground and the development successes on the FSI side of the filtration simulation model are laid out in [1] and [11].

2.6 Why Non-Spherical Particle Modeling?

A crucial part of this work is about the creation of a Lagrangian, non-spherical dirt particle solver. Its ability to consider particle-shape effects in high detail is supposed to set a new standard in the field of filtration simulation. However, taking into account the additional development and computational effort, the question regarding benefits and costs of a non-spherical particle model is valid. The following three sub-chapters are supposed to point out reasons why the consideration of non-spherical effects in filtration simulation is imperative.

2.6.1 Drag Forces and Particle Relaxation Times

The first, obvious reason to go from a spherical dirt particle description to a more realistic, non-spherical approach lies within a significant deviation in drag-force-to-mass-ratio. A good way to demonstrate the difference is to take a look at spherical and non-spherical particle relaxation times τ_p of mass equivalent particles. The parameter τ_p can be thought of as the ratio between particle inertia and fluid viscous forces.

Regardless of their shape, all particles of equal mass feature the same *diameter of a mass equivalent sphere* D_{sph}. With m_p being the particle mass and ρ_p being the particle density, D_{sph} can be written as:

$$D_{sph} = \sqrt[3]{\frac{6m_p}{\pi\rho_p}} \tag{3}$$

Since the particle Reynolds numbers under consideration range significantly below 1, Stokes drag conditions can be assumed. Thus the expression for the particle relaxation time for the translation of spherical particles $\tau_{p,sph}$ in the flow domain is given by:

$$\tau_{p,sph} = \frac{D_{sph}^2 \rho_p}{18\mu_f} \tag{4}$$

For non-spherical particles a drag force correlation, proposed by Hölzer & Sommerfeld [24], shall be chosen. It has been derived out of extensive Lattice Boltzmann (LB) simulations concerning the drag- and lift forces acting on non-spherical particles (see also chapter 4.2.5.2). In this case the definition of the non-spherical particle relaxation time $\tau_{p,nonsph}$ is more complex and reads:

$$\tau_{p,nonsph} = -\frac{C_0}{C_1 + C_2\sqrt{u_{rel}} + C_3 u_{rel}} \tag{5}$$

Here u_{rel} is the relative fluid–particle velocity and the constants C_0, C_1, C_2 and C_3 are:

$$C_0 = \frac{2m_p}{A_{f,ell}\rho_f} \tag{6}$$

Where $A_{f,ell}$ is the frontal area of an ellipsoid particle, projected onto a plane, perpendicular to the relative fluid-particle velocity vector.

$$C_1 = 8\frac{\nu_f}{D_{sph}}\frac{1}{\sqrt{\Phi_{length}}} + \frac{2}{\sqrt{\Phi}} \tag{7}$$

$$C_2 = 3\sqrt{\frac{\nu_f}{D_{sph}}}\frac{1}{\Phi^{3/4}} \tag{8}$$

$$C_3 = 0.421^{0.4210^{0.4(-\log\Phi)^{0.2}}\frac{1}{\Phi_{\mathrm{cross}}}} \tag{9}$$

In Equ.7 to Equ.9 ν_f is the kinematic fluid viscosity, Φ, Φ_{cross} and Φ_{length} are the shape dependent, overall sphericity, length-wise sphericity and cross-wise sphericity, respectively (for exact definitions see [24] or chapter 4.2.5.2). The comparison of Equ.4 and Equ.5 shows that the *Hölzer-Sommerfeld* drag correlation yields a surprising result. Even with the longest particle half axis being aligned along the fluid stream lines, non-spherical particle relaxation times are generally lower than those of mass equivalent spheres. This can be explained by the *low-Reynolds dominance* of shear forces over pressure forces and by the fact that for increasing non-sphericity the particle surface area increases as well. In [5] the parameter α_{ax} is introduced to measure deviation from spherical shape. It represents the medium, relative half axis deviation from D_{sph} and is defined as:

$$\alpha_{ax} = \frac{\sqrt{\left(2a - D_{sph}\right)^2 + \left(2b - D_{sph}\right)^2 + \left(2c - D_{sph}\right)^2}}{3D_{sph}} \qquad (10)$$

Here a, b and c are the lengths of the three particle half axes, whereby $a \geq b \geq c$. Using α_{ax} as a parameter, it becomes apparent that, the further the particle shape deviates from being a sphere (higher α_{ax}), the smaller $\tau_{p,nonsph}$ will be. A corresponding plot of the situation, as seen in Figure 8, reveals that non-spherical particle relaxation times show a dependency on local fluid conditions e.g. relative velocity u_{rel}, while spherical particle relaxation times do not.

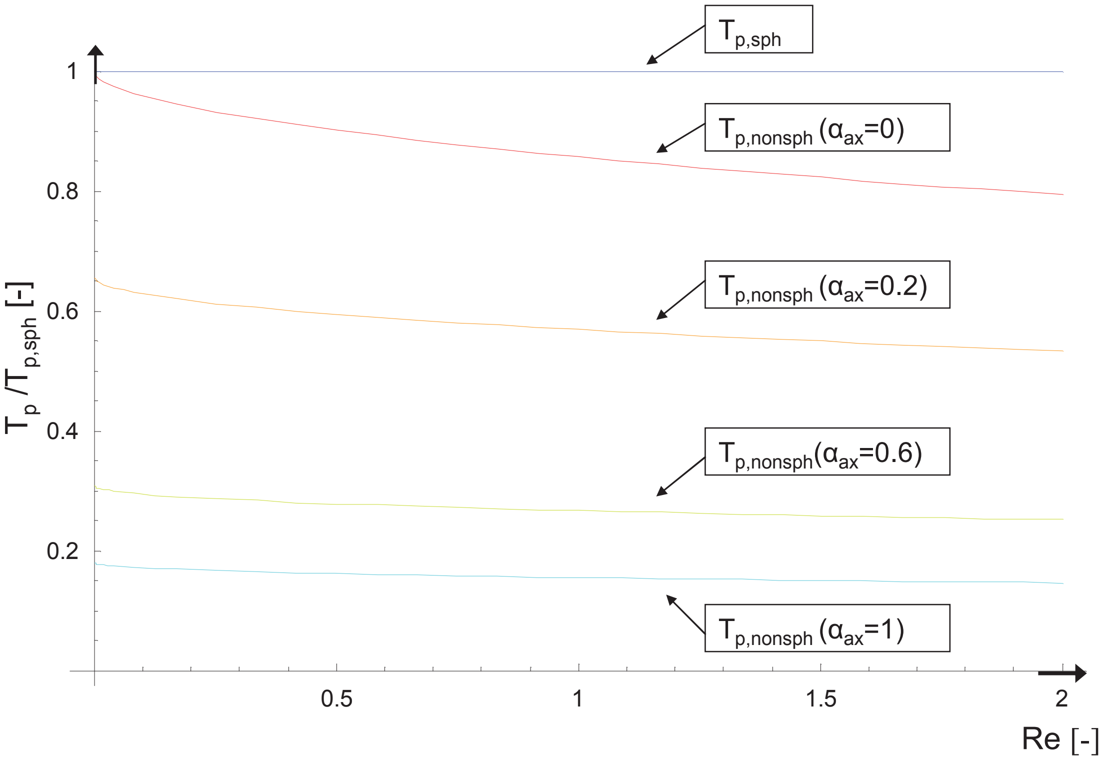


Figure 8: Spherical Stokes (blue) and non–spherical *Hölzer-Sommerfeld* (red, orange, yellow, turquoise) particle relaxation time behaviour against particle Reynolds number. Assumption: the longest half axis *a* is aligned along fluid stream lines.
Increasing α_{ax} (0.0-1.0) leads to lower $\tau_{p,nonsph}$. All values are scaled by $\tau_{p,sph}$(Re=0).

Furthermore the results in Figure 8 show that non-spherical particle relaxation times for highly non-spherical particles ($\alpha_{ax}\geq1$) amount to less than 1/5th of spherical relaxation times. The comparison of the results of the $\tau_{p,sph}$ and the $\tau_{p,nonsph}(\alpha_{ax}=0)$ line does reveal a certain discrepancy between the analytical Stokes drag solution for spherical particles and the LB based, semi-empirical drag formulation by Hölzer & Sommerfeld. This points to the fact, that the latter is only valid for $\alpha_{ax}>0$. However, the results clearly underline that a mere spherical particle model would significantly underestimate fluid *skin friction* and *form drag forces* on supposedly arbitrarily shaped dirt particles. One obvious consequence of disregarding particle shape effects for filtration simulation would be an overestimation of filter fibre efficiencies.

2.6.2 The Non-Spherical Particle Slip Effect

To characterize the rotational response of a non-spherical particle to torque effects, the rotational particle relaxation time $\tau_{p,\omega}$ can be introduced as [49]:

$$\tau_{p,\omega} = \frac{I_p(\Phi_p)}{c_{d,\omega}(\mathrm{Re}_p) \cdot \mu_f \cdot V_p} \tag{11}$$

Here the particle moment of inertia I_p depends on the particle orientation angle Φ_p towards the fluid stream. The rotational drag force coefficient $c_{d,\omega}$ depends on the local fluid conditions. V_p stands for particle volume.

Longish, non-spherical particles with *small*, rotational relaxation times show a tendency to easily align themselves along the streamlines of the surrounding fluid. If D_P, D_b, D_c denote the axis diameters of an ellipsoid, so that $D_c \leq D_b \leq D_P$, then the smallest possible, projected, frontal area of an ellipsoid is given by:

$$A_{p,\min} = \frac{D_b D_c}{4} \pi \tag{12}$$

Particle alignment increases the likelihood of slipping through a pore of diameter D_F in a direction, perpendicular to $A_{p,\min}$. Consequently the two smaller axes diameters define the minimal, equivalent, spherical pore size diameter $D_{F,\min}$ that an aligning, non-spherical particle can theoretically slip through:

$$D_{F,\min} = \sqrt{D_b \cdot D_c} \tag{13}$$

This means:

$$D_{F,\min} < D_P \tag{14}$$

The diameter of mass equivalent, spherical particles D_{sph} can also be written as:

$$D_{sph} = \sqrt[3]{D_P \cdot D_{F,\min}^2} \tag{15}$$

Which leads to the relation:

$$D_{F,\min} < D_{sph} \tag{16}$$

Thus the hydrodynamic *slip effect* of non-spherical dirt particles increases particle penetration depth and decreases filter fibre efficiency as compared to the case of mass equivalent, spherical particles.

For filtration simulation this means that a representation of non-spheres by mass equivalent spheres with diameter D_{sph} might lead to a considerable *overestimation* of the filter fibre efficiency. A representation by spheres of diameter D_p, on the other hand, will not only lead to an overestimation of filtration efficiencies, but will also result in wrong particle masses and consequently in wrong calculations of over all particle hydrodynamics. Figure 9 presents a basic sketch of the situation.

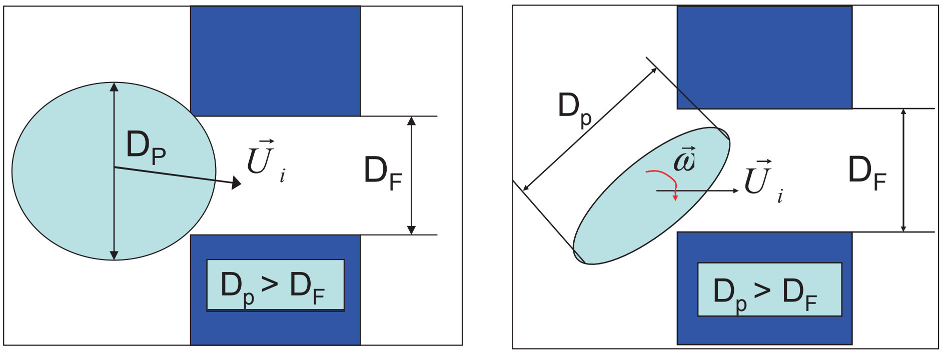


Figure 9: Non-spherical particle slip effect. Representation by spherical particles of diameter D_p (left) means overestimation of filtration efficiencies and particle mass as compared to the more realistic, non-spherical representation (right). The use of spheres with D_{sph} still leads to overestimation of filtration efficiencies because of Equ.16.

2.6.3 The Non-Spherical Particle Bulk Effect

Longish, non-spherical particles with *larger*, angular relaxation times show a tendency to hardly align themselves along the streamlines of the surrounding fluid. As a consequence, the two larger half axes D_p and D_b commonly define the minimal, spherical equivalent pore size diameter $D_{F,min}$ that the particle can slip through:

$$D_{F,\min} = \sqrt{D_P \cdot D_b} \qquad (17)$$

Since $D_{F,min} > D_{sph}$, a mass equivalent spherical particle might just slip through pores, that a bulky, non-spherical particle may not pass. Thus the hydrodynamic *bulk effect* of non-spherical dirt particles decreases particle penetration depth and increases filter fibre efficiency as compared to the case of mass equivalent, spherical particles.

In terms of filtration simulation this means that a representation of large non-spheres by mass equivalent spheres with diameter D_{sph} might lead to a considerable *underestimation* of the filter fibre efficiency. Figure 10 presents a basic sketch of the situation.

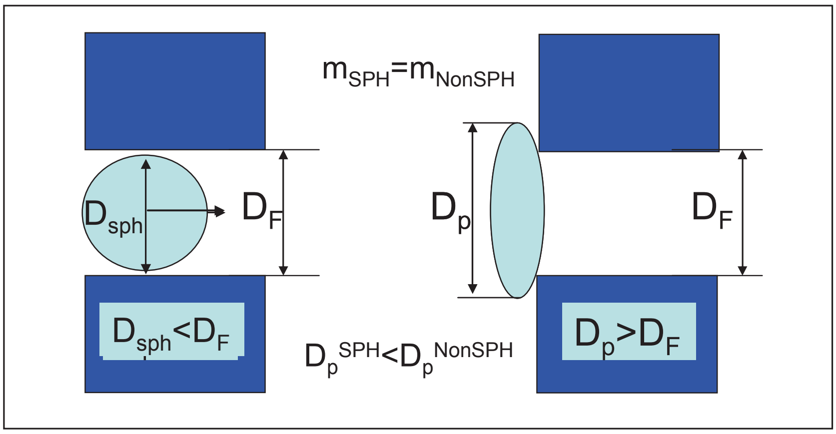


Figure 10: Non-spherical particle *bulk effect*. Representation by mass equivalent, spherical particles of diameter D_{sph} (left) means underestimation of filtration efficiencies as compared to the more realistic, non-spherical representation (right).

The bottom line at this point is that a non-spherical particle model, capable of including particle shape effects as well as rotational particle alignment, gives a much more realistic insight into detailed interaction behaviour in the particle-fibre vicinity, than any spherical representation ever can.

3 Basic Concepts of the Large, Lagrangian Dirt Particle and Deposition Model

This chapter deals with three important, conceptual corner stones of the developed model:

- The fact that a Lagrangian rather than an Eulerian particle simulation approach is chosen.

- The introduction of a strict *force-to-motion concept*.

- The solver's ability to realistically calculate *small* and *large* particles.

3.1 Lagrangian Particle Modelling Approach

A Lagrangian [25] approach to simulate particle behaviour considers particles to be individual entities, interacting with the surrounding environment. In general, this modelling concept is used to consider quite limited numbers of particles or parcels in higher levels of detail. Deterministic concepts are commonly applied for the calculation of individual particle behaviour. The Eulerian [16] approach, on the other hand requires a second, continuous particle phase along with the solution of corresponding conservation equations. This concept is rather used for modelling dense clouds of not individually resolved particles. Stochastic modelling concepts are common here.

The given task is to understand and simulate the dynamic interaction of individual micro scale effects that lead to dirt particle deposition in a filter fibre material. Therefore a high resolution of physically relevant details is required. As a consequence, a Lagrangian and not an Eulerian approach was chosen for the model.

In the Lagrangian implementation, all particle movement is based on the simple, Lagrangian equations of motion. The values for e.g. particle position X_p, orientation Φ_p, velocity u_p, angular velocity ω_p, acceleration a_p and angular acceleration α_p, depend on the sum of acting forces that come from interaction modelling and from the resulting velocity and pressure field out of the Eulerian fluid calculation. The Lagrangian equations of translatory motion are written out as Equ.18 and Equ.19.

$$\frac{d\vec{X}_p}{dt} = \vec{u}_p \tag{18}$$

$$\frac{d\vec{u}_p}{dt} = \vec{a}_p \tag{19}$$

The Lagrangian equations of rotational motion are:

$$\frac{d\vec{\Phi}_p}{dt} = \vec{\omega}_p \tag{20}$$

$$\frac{d\vec{\omega}_p}{dt} = \vec{\alpha}_p \tag{21}$$

The coupling of particle behaviour to the particle surroundings is performed by the acceleration terms a_p and α_p. They result from force- and torque vectors. Their formulation is the essential part of Lagrangian particle simulation.

3.2 The Force-to-Motion Concept

Translational and rotational force- and torque effects with influence on particle trajectory and deposition behaviour have to be accounted for. Those effects can be parted into three basic categories: particle-fluid (see chapter 4.1.2), particle-fibre (see chapter 4.1.4), and particle-particle (see chapter 4.1.5), interactions.

It is inaccurate to traditionally model e.g. one individual particle-deposition effect, without taking into account the interaction with other particles, or a changing flow field [12], [13]. Averaging, semi-empirical equations are therefore hard to define and inaccurate. Thus the *force-to-motion concept* is introduced here. It states that neither particle translation nor rotation can occur without previously, explicitly calculated force- and/or torque vectors. In this work all modelling is broken down to the level of individual force effects and their resulting torques. The following interaction forces are considered:

• particle–wall impact force,	F_{wall}
• particle–fibre interaction force,	F_{fibre}
• particle–particle impact force,	$F_{collision}$
• particle–fluid interaction (drag) force,	F_{fluid}
• force due to pressure gradient (form drag),	$F_{pressure}$
• force due to shear flow (shear drag),	F_{shear}
• gravity,	F_g

A simultaneous calculation of F_{fluid}, $F_{pressure}$ and F_{shear} would yield an overestimation of fluid–particle interaction forces. An important aspect of this work is the ambition to numerically resolve flow conditions for individual particles. Thus it becomes possible to break down all relevant force contribution terms to F_{fluid} to their essential causes: pressure and shear effects. Depending on the specific mode of operation, either F_{fluid} or $F_{pressure}$ & F_{shear} are calculated. Figure 11 illustrates an assembly of small, (non-) spherical particles and the corresponding system of acting forces and torques, which cause translation and rotation.

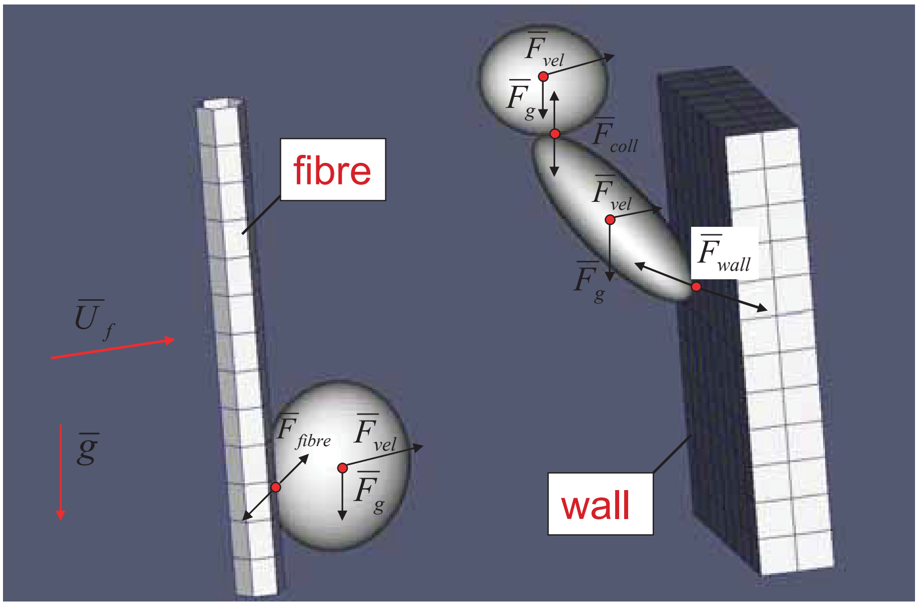


Figure 11: Illustration of acting forces and torques on an assembly of (non-) spherical particles.

3.3 The Large Particle Model

A special feature of the presented Lagrangian particle solvers is their ability to handle both *small* and *large* particles as shown in Figure 12. With the mean cell diameter being D_c, the term *large* refers to the case of $D_{sph}/D_c>1$.

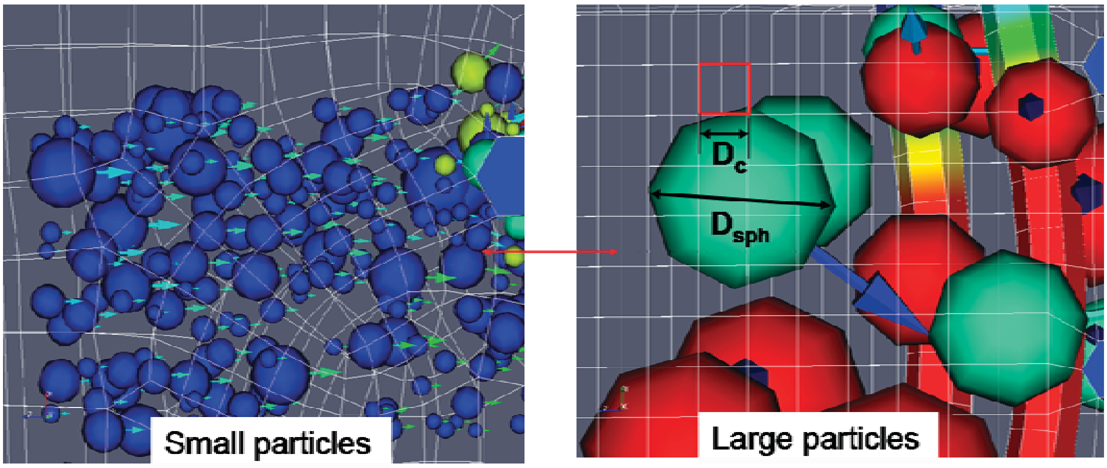


Figure 12: Particle simulation with small ($D_{sph}/D_c \leq 1$) particles (left) and large ($D_{sph}/D_c>1$) particles (right).

The modelling of *large* particles essentially entails three important adaptations concerning the interaction force implementation and the concept of fluid-particle interaction:

- The fluid flow field is uniform within each calculation cell. This is why for *small* particles the fluid drag and lift forces are calculated by using the uniform, relative fluid-particle velocity u_{rel}. For *large* particles the fluid velocity field has to be considered as generally non-uniform over the surface of the particle. Consequentially the changes of u_{rel} across the particle surface have to be considered. Here this is achieved by the introduction of pressure-velocity help points (see chapter 4.2.2.4.1).

- In the case of *large* particles, the calculation of pressure force effects can no longer rely on semi-analytical form drag formulations. In order to get a hold of pressure gradients forming across the surface of particles the pressure-velocity help point concept is useful as well.

- The two-way coupling of a *small* particle to the fluid field is, if at all, usually realized via a local momentum source term within the engulfing fluid cell. A large particle per definition causes more pronounced distortions to the fluid solution, since it affects not just one but multiple fluid calculation cells. Large particle-two-way coupling effects are hereby realized via the introduction of an elaborate plugging scheme (see chapter 4.1.2 and 4.2.6).

4 The (Non-) Spherical Dirt Particle Deposition Solvers

During the course of the past four years, two OpenFOAM® dirt particle and deposition simulation solvers have been created. Both are based on the basic solver concepts described in chapter 3 (Lagrangian, *large* particle model, *force-to-motion concept*). They can be used alternatively as required by the user.

- The spherical Lagrangian particle solver:

  This simulator is the original version. It has been created between 2005 and 2006 and is the basis for all later developments. This original particle model is a spherical, Lagrangian, fully deterministic (non-stochastic) approach with the capability to interact with the surrounding, Eulerian fluid–fibre framework. Each particle can extend well beyond the borders of a single calculation cell and can sense and affect fluid conditions within a multiple cell region of the fluid mesh.

  However, the implementation of many physically relevant effects, such as particle-fluid force calculation is conducted on a rather qualitative basis and the level of detail and accuracy is generally lower than in the more advanced, (non-)spherical solver. Calculation times and memory requirements are lower than for the advanced version, which is why this software is still an important tool. During the course of this thesis one relevant article on this subject has been published: [2].

- The (non-)spherical Lagrangian particle solver:

  This simulator is the advanced, final version. It has been created between 2007 and 2009 and constitutes the core part of this thesis. As an extension of the spherical particle model, it is a more sophisticated and more accurate tool which can handle spherical as well as non-spherical dirt particle and deposition behaviour within and without the vicinity of realistically reconstructed fluid filter fibre geometries. The main advantage over the original version is its ability to realistically handle non-spherical particle shape effects adding three rotational degrees of motional freedom. In addition to that it features much more detailed, quantitatively verified implementations of all physically relevant aspects of particle behaviour. Calculation times as well as memory requirements are higher but results are more accurate than those of the spherical solver. During the course of this thesis two relevant publications on this subject have appeared: [3, 4].

4.1 The Original, Spherical Particle Solver

The following sub chapters describe in detail how the prevailing physics has been implemented into the original, spherical, large, Lagrangian particle solver. Initially the focus is laid on the modelling of particle-fluid interaction forces. Secondly impact events are treated. Forces with influence on particle motion, that occur due to individual impact events, are hereby called *event forces*. They represent particle-wall, particle–particle and particle-fibre interaction effects. Finally the *large* particle aspects of the spherical implementation are discussed.

4.1.1 The Particle Momentum Equation behind the Spherical Solver

For merely spherical particles, rotational effects are irrelevant as long as *Magnus* forces [14] are neglected. Therefore only the translational, Lagrangian equations of motion (Equ.18 and Equ.19) need to be considered. The particle acceleration term a_p is given via the formulation of the particle momentum equation (PME). The PME behind the spherical solver considers fluid-particle drag F_{drag}, buoyancy, and three major categories of impact forces, which are summarized in Equ.22 as F_{pi}.

$$m_p \vec{a}_p = \vec{F}_{drag} + V_p \left(\rho_p - \rho_f\right)\vec{g} + \vec{F}_{pi} \qquad (22)$$

While buoyancy effects can be implemented just as shown in Equ.22, all other factors of influence need specific elaboration in terms of modelling.

As listed in Equ.23, the three components of the impact force term F_{pi} are:

- particle collision effects, F_{coll}
- particle-wall interaction forces, F_{wall}
- particle-fibre interaction forces, F_{fibre}

$$\vec{F}_{pi} = \vec{F}_{coll} + \vec{F}_{wall} + \vec{F}_{fibre} \tag{23}$$

The event forces summarized within F_{pi}, in general have a duration time Δt_i which is much shorter than any reasonable, discrete particle time step Δt_p. Consequentially they produce extremely high, time dependent impact forces $F_i(t)$. To correctly simulate the overall particle momentum change due to all n impact events, which occur during Δt_p, an adapted impact force $F_{i,adapted}$ has to be used as shown in Equ.24.

$$\vec{F}_{pi} = \vec{F}_{i,adapted} = \frac{1}{\Delta t_p} \cdot \sum_{i=1}^{n} \int_{t=0}^{\Delta t_i} \vec{F}_i(t) \cdot dt \tag{24}$$

The following chapters describe how the compositional terms of the spherical solver's PME: F_{drag}, F_{coll}, F_{wall} and F_{fibre} are derived and computed.

4.1.2 Particle – Fluid Interaction: Drag Forces on *Small* Particles

In the case of sparsely distributed, *small* particles in a highly viscous fluid, the drag force is the dominant factor on particle movement. The commonly used expression to model fluid drag on spherical particles is shown in Equ.25, where, A_f is the cross sectional particle area perpendicular to flow direction and c_d is the dimensionless drag coefficient.

$$\vec{F}_{drag} = \frac{1}{2} \cdot A_f \cdot c_d \cdot \rho_f \cdot \left(\vec{u}_f - \vec{u}_p\right) \cdot \left|\vec{u}_f - \vec{u}_p\right| \tag{25}$$

The order of both, particle diameter and kinematic fluid viscosity is approximately 10^{-4}. As a consequence the order of the corresponding particle Reynolds number Re_p depends directly on the order of relative fluid-particle velocity u_{rel}, which is well below 1, hence:

$$O\left(\mathrm{Re}_p = \frac{D_{sph} \cdot \left|\vec{u}_{rel}\right|}{\nu_f}\right) \sim O\left(\left|\vec{u}_{rel}\right|\right) < 1 \tag{26}$$

Therefore inertial flow field effects on drag can be neglected, simplifying the *Navier-Stokes* equations in tensorial notation to pressure and viscosity effects [15]:

$$\frac{\partial p}{\partial x_i} = \mu_f \cdot \frac{\partial^2 u_i}{\partial x_j \partial x_j} \tag{27}$$

In Equ.27 the indices i and j stand for the n vectorial components for n=3 dimensional vector space. Under these conditions the *Stokes* law for drag on spherical particles is applicable. It is shown in Equ.28. In contrast to other relations for $c_d = f(\mathrm{Re}_p)$, it yields an entirely analytical solution for the drag coefficient, which considers both, form drag and the shear stress.

$$c_d = \frac{24}{\mathrm{Re}_p} = \frac{24 \cdot \nu_f}{\left|\vec{u}_{rel}\right| \cdot D_{sph}} \tag{28}$$

For the limit of $\mathrm{Re}_p \rightarrow 0$ the form drag coefficient $c_{d,p}$ and the shear stress contribution $c_{d,shear}$ relate to c_d as:

$$c_d = c_{d,p} + c_{d,shear} = \frac{8}{\mathrm{Re}_p} + \frac{16}{\mathrm{Re}_p} \tag{29}$$

Should small particles experience higher particle Reynolds numbers ($\mathrm{Re}_p > 4$), the solver automatically switches to semi-empirical correlations like the *Abraham* equation [34] seen in Equ.30. It is valid up until $\mathrm{Re}_p < 6.000$. For particle Reynolds numbers $6.0 * 10^3 < \mathrm{Re}_p < 2.0 * 10^5$ the *Turton–Levenspiel* [34] equation is used. It is shown in Equ.31.

$$c_d = \frac{24}{\mathrm{Re}_p}\left(1 + 0{,}11 \cdot \sqrt{\mathrm{Re}_p}\right)^2 \tag{30}$$

$$c_d = \frac{24}{\mathrm{Re}_p} + \frac{5,48}{\mathrm{Re}_p^{0,573}} + 0,36 \tag{31}$$

Figure 13 shows the close accordance of the *Abraham* and the *Turton–Levenspiel* equation with experimental results.

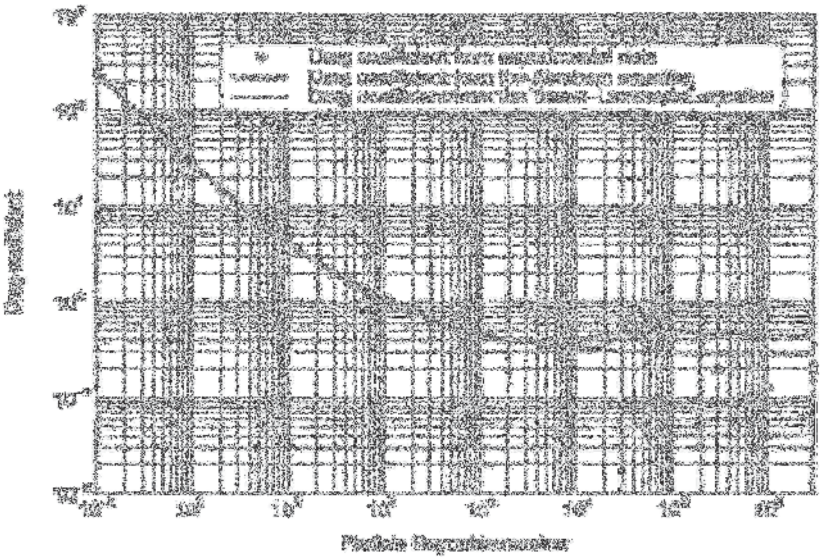


Figure 13: Drag coefficient of solid spheres plotted against particle Reynolds number. Comparison of experimental data with results from *Abraham* and *Turton-Levenspiel* equation [34, 50].

This simple drag implementation into the Eulerian-Lagrangian, OpenFOAM® solver already leads to very plausible results in terms of particle sizing effects. Due to a higher drag-force-to-inertia ratio (smaller particle relaxation time), smaller particles accelerate much more readily than larger particles in an otherwise steady state flow field. An exemplary result is shown in Figure 14.

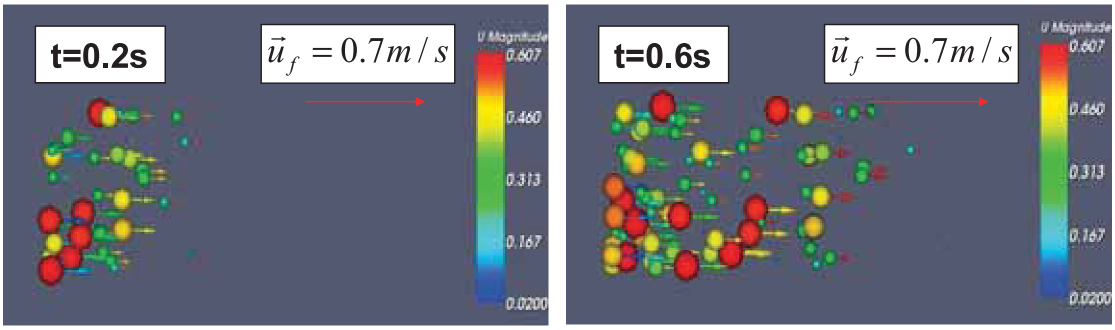


Figure 14: Particle sizing effect: smaller particles follow fluid motion more readily.

The qualitative impact and pass scenarios of two differently sized particles with a single fibre are depicted in Figure 15. The impact probability can be characterized by the *Stokes* number St (see chapter 8.1.2). Whilst the smaller particle, with St~0.15 follows the flow field in the neighbourhood of the fibre, the larger particle, with St~0.6 impacts on it and deviates accordingly. The influence of particle inertia effects is demonstrated in a physically plausible way. *Stokes* number values for typical dirt particles with relevance for fluid filtration that are considered within this work range from $St \sim 1.5*10^{-5}$ to $St \sim 1.5*10^{-3}$. According to [81] common *Stokes* number values for the entire field of fluid filtration range from $St \sim 10^{-9}$ to $St \sim 2*10^{-3}$.

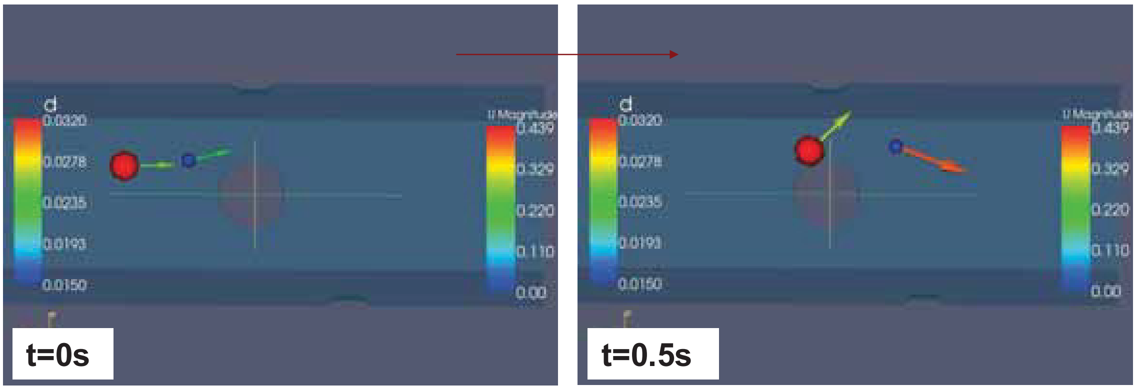


Figure 15: Inertial Impact effect: particles of higher *Stokes* number show higher probability of impact.

4.1.3 Spherical Particle Event Forces: Particle–Wall Interaction

A second, decisive factor for particle movement is their interaction with obstacles, like geometry boundary patches, in the stream line path.

There are two types of boundary patches: inlet/outlet patches, whereby the fluid enters/leaves the region and wall patches. The latter represent borders to neighbouring fibre regions that are not included into the calculation.

Particles hitting an inlet or outlet vanish from the calculation. Particles hitting a wall patch would in reality leave for the neighbouring fibre vicinity. From the statistical point of view just as many particles would enter through the wall boundaries. This leads to a relative conservation of particle cloud density ρ_{pc} perpendicular to the flow direction, as described by Equ.32.

$$\overline{\frac{\partial \rho_{pc}(r)}{\partial r}} = \frac{2\pi}{A_f} \cdot \int_{r=0}^{R} \frac{\partial \rho_{pc}(r)}{\partial r} \cdot r \cdot dr = 0 \tag{32}$$

Where r is the distance to the centre of the cross sectional flow area A_f. As a consequence, a simple elastic reflection rule has been implemented at the wall boundary patches of the geometry. Following Equ.24 the computed wall impact force F_{wall} has to account for the resulting momentum change of the particle mass m_p during the discrete force effect time Δt_p (=particle sub time step). For elastic reflection, the resulting momentum change should reverse the direction of the particle velocity components parallel to the normal vector n_w of the wall patch. The computed wall impact force is specifically calculated to have just that effect and is formulated as:

$$\vec{F}_{wall,el} = -m_p \frac{2 \cdot (\vec{u}_p \cdot \vec{n}_w) \cdot \vec{n}_w}{\Delta t_p} \tag{33}$$

For semi-elastic impact scenarios a user-defined coefficient of elasticity E_{pw} is introduced. It is 1 for totally elastic and 0 for totally inelastic particle-wall behaviour. To implement this option, the expression for the elastic impact force of Equ.33 has to be extended to describe the adaptable, semi-elastic wall force $F_{wall,iel}$. It can be written as:

$$\vec{F}_{wall,iel} = \vec{F}_{wall,el} \cdot \left(\frac{1 + E_{pw}}{2} \right) \tag{34}$$

This implementation, at the end of a wall impact time step Δt_p, results in the computation of the updated particle velocity $u_p(t+\Delta t_p)$ as shown in Equ.35.

$$\vec{u}_p(t + \Delta t_p) = \vec{u}_p(t) - (1 + E_{pw}) \cdot \lfloor \vec{u}_p(t) \cdot \vec{n}_w \rfloor \cdot \vec{n}_w \tag{35}$$

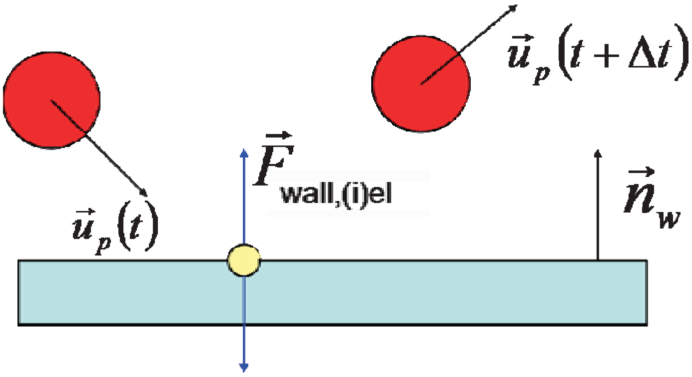


Figure 16: Sketch of spherical particle-wall impact event.

4.1.4 Spherical Particle Event Forces: Particle–Fibre Interaction

A formulation of particle-fibre interaction effects is essential for modelling the overall filter efficiency of the medium. Effective particle-fibre forces will vary depending on parameters such as surface roughness, particle-fibre adhesion or digital surface resolution. A model has been implemented in order to be valid for any kind of filter material. User-defined parameters can account for material specifics.

The procedure of particle immobilization on the fibre is broken down and simplified as follows:

1. Particle impact: When a particle hits the fibre, an impact of user-defined, material dependent elasticity occurs. It produces a particle-wall interaction force according to Equ.34. A completely inelastic impact scenario is chosen here.

2. Since in an inelastic impact scenario only velocity components parallel to the fibre patch remain, the particle glides along the fibre surface for a short time and is exposed to a fibre friction force F_{fibre}. The value of F_{fibre} is set in proportion to the sum of all other particle force components $F_{p,i}$ normal to the fibre surface. In addition to that a proportionality to material properties, expressed via η_{fp}[-] is implemented. F_{fibre} is always directed to point against the direction of the current particle velocity. Thus F_{fibre} is computed as seen in Equ.36, with n being the total number of discrete forces, affecting the particle during time step Δt_p.

$$\vec{F}_{fibre} = -\sum_{i=1}^{n-1} \left(\overrightarrow{F_{p,i}} \cdot \overrightarrow{n_w} \right) \cdot \left(\eta_{fp} \cdot \frac{\overrightarrow{u_p}}{\left| \overrightarrow{u_p} \right|} + \overrightarrow{n_w} \right) \tag{36}$$

3. If the value of the particle force components parallel to the fibre surface is smaller than those of F_{fibre}, the particle slows down. Should the resulting negative particle acceleration during Δt_p lead to reversing the glide direction, the particle velocity is set to *zero* and the fibre friction force is set according to Equ.37.

   $$\vec{F}_{fibre} = -\sum_{i=1}^{n-1} \overrightarrow{F}_{p,i} \tag{37}$$

   Thus the particle gets immobilized on the fibre surface.

4. A user-defined, material dependent sticking barrier F_{stick} is introduced. Only if surface parallel components of the external forces on the particle, e.g. induced by the fluid or by the hitting of other particles, get big enough again to overcome the sticking barrier, the particle can regain some motion.

The essential part about this implementation is the fact that immobilized particles are not just taken out of the calculation framework, but can still interact with their surroundings. Thus the simulation is enabled to model complex particle agglomeration effects or blow off mechanisms near the fibre. A qualitative example can be seen in Figure 17. As will be seen in chapter 4.2, the advanced (non-)spherical particle solver features a more refined version of the particle-fibre interaction model.

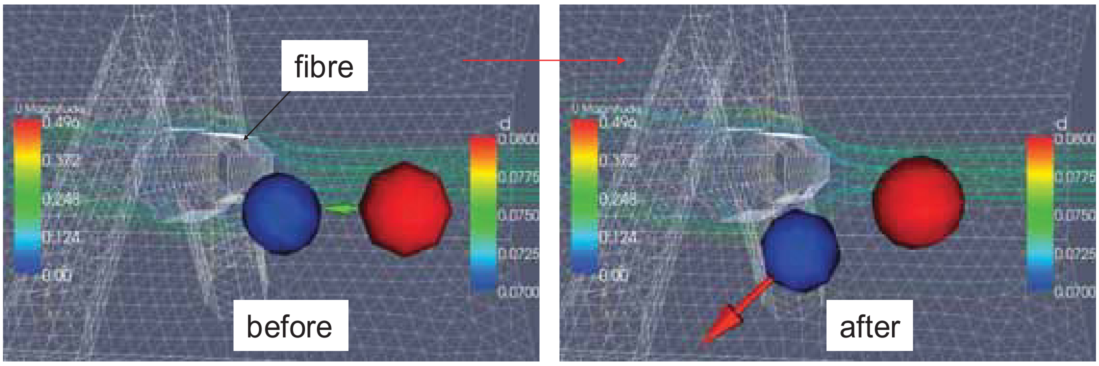


Figure 17: Two particles in fibre vicinity with velocity vectors. Blue particle gets immobilized on fibre (left) is hit by red particle and is blown off again (right).

4.1.5 Impact Forces: Particle–Particle Interaction

Particle agglomeration at the fibre, can lead from complex particle–particle interaction to plugging effects up to changes of overall permeability. Cake filtration effects stem from this very phenomenon. Thus a particle collision model is needed in the solver.

The usual Lagrangian collision model for a particle cloud of N particles requires additional calculation time in the order of ~N^2. Here a collision model was implemented which only considers those N_f particles, which either show fibre interaction, or interaction with other particles that are part of the *collision list*. Because N_f is k times smaller than N, the introduction of the collision list leads to k times faster calculations compared to a full collision model.

Following the *force-to-motion* concept, any particle collision interaction is handled via the calculation of collision forces F_{coll}.

There are two different cases to be considered when modelling the collision force of a particle A of mass m_a and velocity v_a with another particle B, of mass m_b and velocity v_b. The case of particle B being immobilized on a fibre can be handled just like particle–wall collision, given in Equ.33 and Equ.34. The difference is that the immobilized particle B does not just absorb the collision counter force like a wall boundary patch. If the collision force acting on particle B surpasses the fibre sticking barrier, B regains some motion. This constitutes the *blow-off* effect.

If particle B is still in motion when hit by particle A, the case to be dealt with is a collision of two mobile objects of user-defined elasticity. The adapted, elastic particle–particle collision force affecting particle A is described by:

$$\vec{F}_{coll,el} = \frac{2 \cdot m_a \cdot m_b}{\Delta t_p \cdot m_{ges}} \cdot \left(\vec{v}_{b,n} - \vec{v}_{a,n}\right) \tag{38}$$

where $v_{a,n}$ and $v_{b,n}$ are the velocity components along the collision direction between particle centre A and particle centre B. Figure 18 shows a simple sketch of a particle-particle impact event.

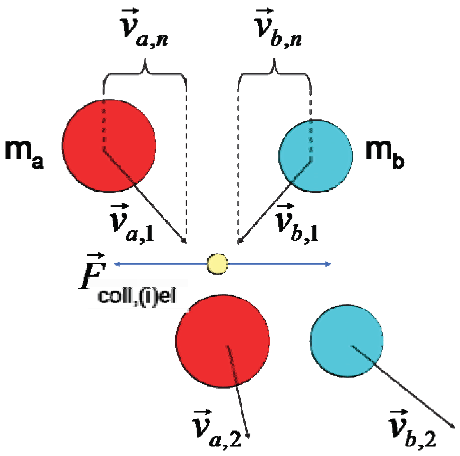


Figure 18: Simple sketch of particle-particle impact event.

In reality any scenario of collision is combined with some degree of object deformation along a certain deformation path Δs_{def}. In the case of total elasticity the deformation is reversed and 100% of the kinetic impact energy is regained by the object. In the case of total inelasticity, no reversed deformation (expansion) takes place. The overall deformation path is only ½ of the elastic case. For both cases the modelled impact takes place during time Δt_p with discrete, constant, relative particle-particle velocity $v_{p,rel}$. Because of that, the virtual deformation path $v_{p,rel} * \Delta t_p$ stays the same. This is why the occurring, modelled collision force $F_{coll,iel}$ has to account for differences in elasticity. Using the elasticity coefficient E_{pp} it can be described by:

$$\vec{F}_{coll,iel} = \vec{F}_{coll,el} \cdot \left(\frac{1 + E_{pp}}{2}\right) \tag{39}$$

The particle collision model can realistically simulate the interaction of large particle numbers in a filter fibre assembly. Figure 19 shows a qualitative example.

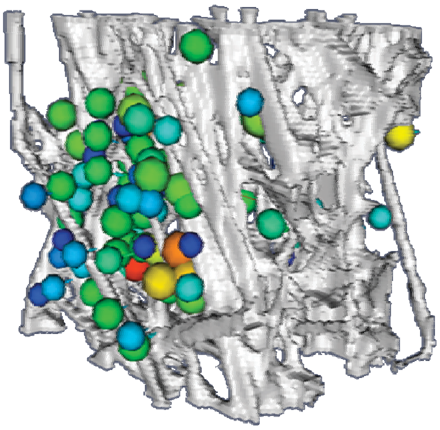


Figure 19: Particle cloud in digitally reconstructed fibre geometry. Large numbers of particle-particle impact events occur.

4.1.6 Spherical, Large Particle Effects: Drag Force via Pressure Gradient

Fluid-particle drag forces, calculated according to Stokes law, consider drag due to shear stress and form drag based upon particle surface normal gradients of the velocity field and pressure gradients over the particle volume, respectively. For *small* particles, uniform flow conditions can be assumed across the entire shape and the implementation of drag effects is simple (see Equ.25). In the case of *large* particles the situation becomes more complex because here the fluid-particle interaction force calculation has to account for non–uniform flow conditions over the particle surface.

The spherical, large particle solver does not explicitly calculate shear drag effects but only form drag effects due to pressure field non-uniformities across the particle surface. To achieve that, the concept of *pressure-velocity help points* has been introduced. *Pressure-velocity help points* are in essence small satellites, located on equally distributed positions on the particle surface. Those help points statically hold their relative position to the particle centre. Their main purpose is to sense pressure p_i and velocity u_i conditions on the particle surface and in the specific calculation cell they are located in.

The number of satellites is user defined. Of course higher help point numbers lead to higher calculation times but also to higher accuracy. Figure 20 shows a particle surrounded by 48 help points.

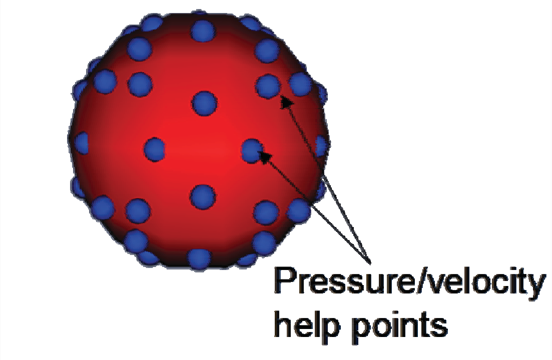


Figure 20: Particle surrounded by 48 enlarged pressure/velocity help points.

An averaging of the velocity values u_i at the help points results in the average, relative fluid-particle velocity used to calculate Stokes drag.

The surface fraction A_{hp}, is assigned to each help point and can be easily calculated according to:

$$A_{hp} = \frac{A_p}{M} = \frac{1}{M} \cdot D_{sph}^2 \cdot \pi \tag{40}$$

Where a_p is the total particle surface and M the number of help points.

The resulting pressure force contribution F_p can be approximated by choosing an appropriately large number of help points. An infinite number of surface help points leads to a perfect representation of the pressure force. This fact can be illustrated by:

$$\overrightarrow{F_p} = \lim_{M \to \infty} \left(-A_{hp} \cdot \sum_{i=1}^{M} p_i \cdot \vec{n}_{p,i} \right) = -\int^{A_p} p \cdot \vec{n}_p dA \tag{41}$$

Where $n_{p,i}$ is the particle surface normal vector at each help point. By applying Gauss' theorem the pressure force can be written as a volumetric term.

$$\overrightarrow{F_p} = -\int^{A_p} p \cdot \vec{n}_p dA = -\int^{Vp} \nabla p \cdot dV \tag{42}$$

Considering the *zero Reynolds limit Stokes relation* between shear and form drag (see Equ.29) the entire drag force on large, spherical particles is calculated as:

$$\vec{F}_{drag} = 3 \cdot \vec{F}_p \tag{43}$$

Thus the PME for large, spherical particles, implemented within the original, spherical solver reads:

$$\vec{a}_p = -3 \cdot \frac{A_{hp}}{m_p} \cdot \sum_{i=1}^{M} p_i \cdot \vec{n}_{p,i} + \left(1 - \frac{\rho_f}{\rho_p}\right) \cdot \vec{g} + \vec{a}_{coll} + \vec{a}_{wall} + \vec{a}_{fiber} = 0 \tag{44}$$

This drag formulation, based on the pressure detection method is not as refined as the one developed for the (non-)spherical solver (see chapter 4.2.6). However, it is useful in combination with the implementation of the *particle plugging effect* which is described in the following chapter.

4.1.7 Spherical, Large Particle Effects: Plugging Effect

If the simulated particles are large in comparison to the dimensions of local filter fibres, accumulation effects occur much more readily. Figure 21 shows the accumulation of large particles in a simplified fibre geometry.

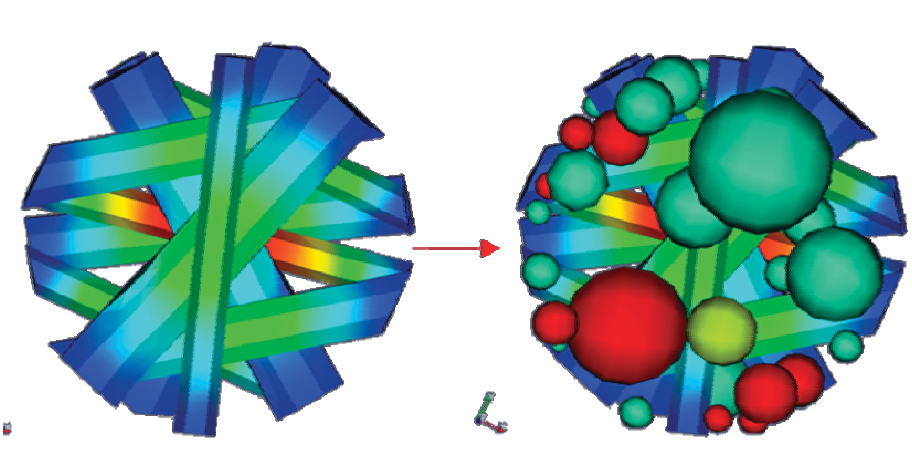


Figure 21: Accumulation of large particles in simplified fibre geometry.

Because of these accumulations, the over all fibre permeability decreases and pressure drop over the filter material sample increases. As a consequence the consideration of the *particle plugging effect* becomes imperative for realistic filtration simulation. In order to simulate particle plugging with effect on the fluid field, a two-way particle-fluid coupling becomes necessary. To achieve this, the vicinity of the particles is seen as a porous medium. Flow calculation in porous media demands an addition of the *Darcy* term to the governing momentum equations [62]. Some cases of flow in porous media require the addition of the *Forchheimer* term as well. This extension can be neglected here due the low flow velocities prevailing. The *Darcy* pressure gradient can thus be expressed via:

$$\nabla p_{Darcy} = \frac{\mu_f}{\alpha_f} \vec{u}_f \tag{45}$$

Where $\alpha_f[m^2]$ is the permeability. Using this expression, the *Navier-Stokes* equation for incompressible, Newtonian flow through porous media can be written as:

$$\rho_f \frac{\partial \vec{u}_f}{\partial t} + \rho_f \left(\vec{u}_f \cdot \nabla \right) \vec{u}_f = -\left(\nabla p + \nabla p_{Darcy} \right) + \mu_f \nabla^2 \vec{u}_f \tag{46}$$

To implement the porous concept in the numerical solver, a *Boolean* depot volume field has been created. Wherever it is set to 0 the permeability goes to ∞ and wherever it is set to 1 a total plugging occurs, changing the corresponding cell permeability to 0. Consequentially a high numeric constant is used in combination with the depot field to approximate the *Darcy* behaviour:

$$\nabla p_{Darcy} = \frac{\mu_f}{\alpha_f} \vec{u}_f = 10^6 \cdot depot(x_i, y_i, z_i) \cdot \vec{u}_f \tag{47}$$

As a starting condition the permeability is ∞ and the depot field is 0 throughout the entire volume. This means unhindered flow. As soon as particles get stuck in the fibre structure, the depot field in the fluid cells enclosed by those particles changes from 0 to 1. The plugging becomes effective.

Figure 22 contains chronological snapshots of a simple, multi-fibre case with a bunch of *large* particles getting stuck. The plugging effect on the fluid flow, which is represented through velocity vectors, is clearly visible.

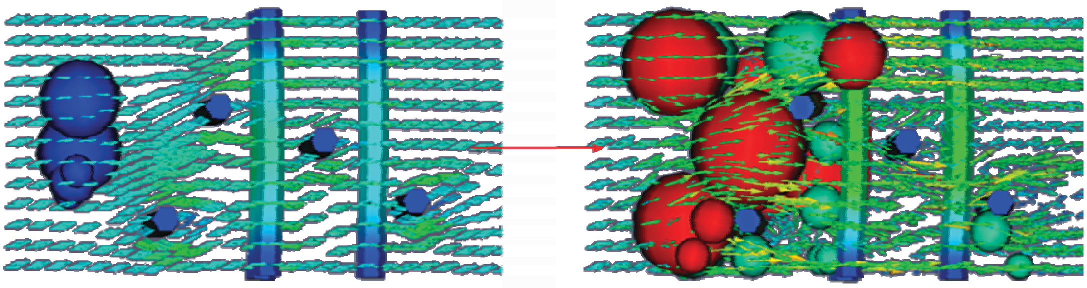


Figure 22: Simple filter fibre case with large particles before (left) and while (right) plugging effect occurs. The vector field symbolizes the fluid velocity field (0.2-0.4m/s).

Figure 23 shows the development of the pressure difference between inlet and outlet against run time. It corresponds to the plugging case shown in Figure 22. At a fixed volume flow rate the inlet-outlet pressure gradient increases over time, just as expected.

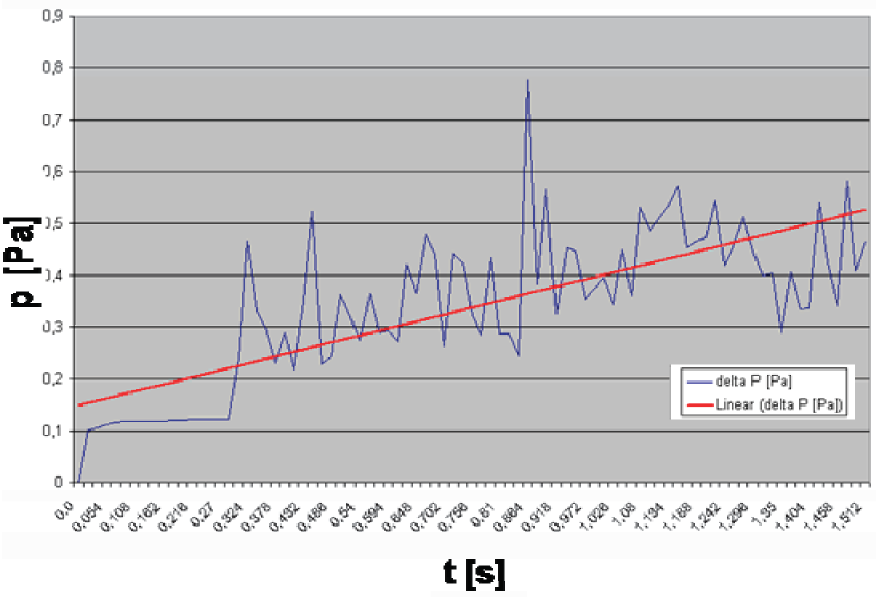


Figure 23: Pressure difference between inlet and outlet over run-time, corresponding to the qualitative, exemplary plugging case shown in Figure 22. Conditions: ρ_p=2000kg/m³, ρ_f=1000kg/m³, ν_f=1*10^{-6}m²/s.

4.1.8 Combined Spherical Filtration Solver

The chapters 4.1.1 to 4.1.7 have given a short overview of the principles and cornerstones of the original, spherical large Lagrangian particle solver. Even though some aspects of its implementation (i.e. drag force calculation) are rather qualitative in nature, it stands as one, compact, completely operational simulation tool. Due to the modular nature of the C++ programming language it can easily be combined with other, OpenFOAM® based simulators. A combination of icoFOAM [39] (the low Reynolds, incompressible fluid solver), the FSI solver (see [1] and chapter 2.5) and this spherical, Lagrangian particle solver yields the *icoLagrangianStructFOAM* simulator. The *icoLagrangianStructFOAM* can herby be presented as the first version, of a combined filtration solver capable of modelling the complex hydrodynamics of microscopic particle deposition processes in realistically reconstructed, deformable fibre media. The various solver modules can be switched on or off as required by the user. This means that flow in deformable media can be calculated without any particle injection on the one hand and that the particle solver can be used without FSI on the other hand.

Several combined solver runs on test benchmark cases and on actual technical applications have proven that the proposed, computational strategy is robust and stable. Moreover result plausibility has been very encouraging from the beginning. In the following some screenshots of successful runs on the *icoLagrangianStructFOAM* are presented.

Figure 24 is a screen shot from a case with simple horizontal and vertical filter fibres that are visibly deformed by an oil stream. A rather dense cloud of particles is injected and the particles get entangled in the fibre structure, which leads to a plugging of the stream path.

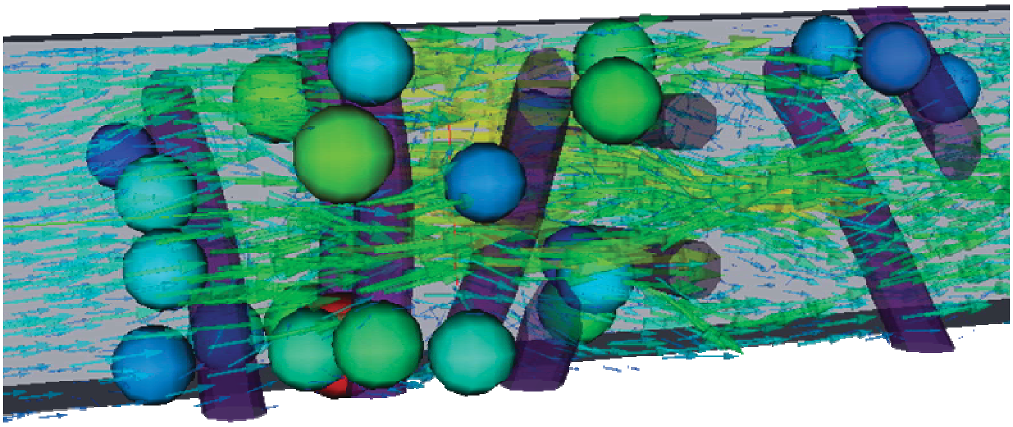


Figure 24: Simplified horizontal and vertical fibre structure (200µm*200µm*1000µm, ~2.5*10^5 cells) deformed by oil flow. Dense cloud of rather large particles (ρ_p=2000kg/m³, 40µm≤D_{sph}≤60µm) getting entangled in the structure and causing plugging effect, e.g. deviation of the flow (u_f~0.4m/s, ρ_f=1000kg/m³, ν_f=1*10^{-6}m²/s).

As seen in Figure 25, the solver could already be robustly applied to realistic geometries. In this example a dense cloud of smaller particles gets injected into the vicinity of a complex fibre structure. The filter fibre used here, has been digitally reconstructed from CT-scans as described in chapter 2.5.

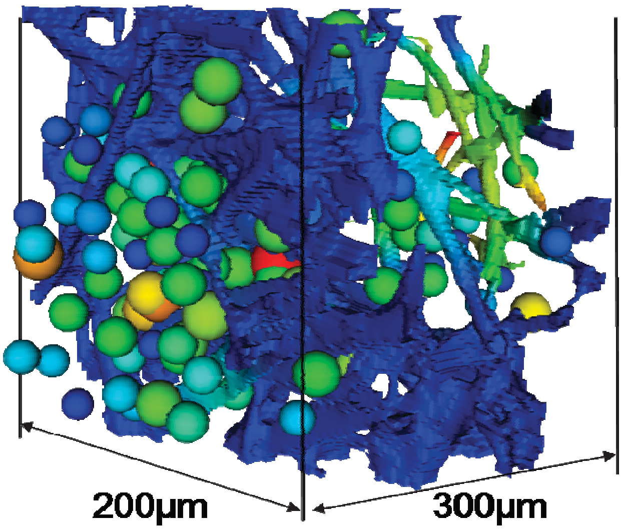


Figure 25: Realistic, microscopic (200µm*200µm*300µm, ~6.0*10^5 cells) fibre geometry reconstructed from CT-scan images. Dense cloud of rather small particles(ρ_p=2000kg/m³, 40µm≤D_{sph}≤60µm), suspended in flowing oil (u_f~0.4m/s, ρ_f=1000kg/m³, ν_f=1*10^{-6}m²/s), gets entangled in the deforming fibre.

4.2 The Advanced, Non-Spherical Particle Solver

It has been stated in chapter 2.6.1 that drag-force-to-mass-ratios and consequentially also particle relaxation times of non-spherical particles can dramatically deviate from the corresponding values, encountered in mass equivalent spheres. In addition to drag and lift force issues, it is mainly the physical- and hydrodynamic interaction situation between fluid, particles and fibres, that influences filter fibre efficiency. Right here, particle shape effects are most relevant. Non-spherical particle *slip-* and *bulk effects* (see chapter 2.6.2 and 2.6.3) can be observed in filtration. A direct comparison of spherical and non-spherical particle filtration behaviour leads to significant deviations in filter fibre efficiency (see also chapter 9.2) and particle penetration depth.
This is why, in extension of the previously presented, spherical particle model (chapter 4.1), a highly detailed, more sophisticated and more accurate, non-spherical particle model had to be created. In chapter 4.2 of this thesis, a significant extension of the original, spherical dirt particle model is laid out. It describes the basic concepts, the essential drag and lift force implementation method as well as the particle-surroundings-interaction schemes, behind a novel, realistic, Lagrangian, non-spherical particle solver.

During the course of development, the following, chronologically listed, main tasks turned out to be essential:

- Creation of an explicit, force and torque vector model, stating that any translational and/or rotational change in motion can only stem from a previously, explicitly calculated force and/or torque vector.

- Design of particles with basic, non-spherical abilities:
  - Creation of ellipsoids with three geometrical degrees of freedom.
  - Consideration of:
    position *and* orientation.

  mass *and* moment of inertia tensor.
  force *and* torque vectors.

- Device of help concepts:
  - Introduction of a co-rotational particle coordinate system
  - Particle surface help point method
  - Particle-panel model
  - Time step control to eliminate numerical instabilities
- Realistic calculation of non-spherical drag and lift forces.
- Event force handling of non-spherical particle interaction effects:
  - Particle-particle collisions
  - Particle-wall impacts
  - Particle-fibre interaction including deposition modelling
- Conditioning of output data to yield essential process parameters like:
  - Filter fibre efficiency
  - Particle penetration depth
  - Permeability
  - Pressure drop
- Validation of CFD results: Analytical and experimental verification.

Resulting from these tasks and underlying this thesis, a series of development successes can hereby be proudly presented:

- Design of the *Force-to-Motion* concept, that reduces the modelling to the mere formulation of single force effects ([2, 3] and chapter 3.2).
- Implementation of a *Six Degrees of Freedom* (DOF) solver for the Lagrangian particle momentum equations (PME) in OpenFOAM® ([3] and chapter 4.2.2.3).
- Introduction of an *Adaptive Time Stepping Scheme* for explicit Euler discretization of the PME ([5] and chapter 5).

- Device of a surface *Help Point Scheme* to account for large particle effects in terms of fluid–particle, particle–fibre and particle–particle interaction ([2, 3] and chapter 4.2.2.4.1).
- Development of a non-coupled drag force implementation that uses a combination of non-spherical, semi-empirical drag force formulas [24] and a panel method to consider free flow swirling effects: The *Free Flow Drag Module* ([3] and chapter 4.2.5).
- Creation and verification of an efficient particle–fluid, two-way coupling method: The *Fibre Vicinity Drag Module* which is a plugging method to consider inter-particle and particle–fibre hydrodynamics. It also includes a simple adoption of basic concepts known from the immersed boundary method [17]. ([3] and chapter 4.2.6).
- Implementation of a detailed particle–fibre interaction- and deposition model ([4] and chapter 4.2.7.2).
- Creation of a non-spherical particle collision model, including exact impact-point determination as well as the consideration of rotational collision effects ([4] and chapter 4.2.7.3).
- Programming of Python [11] based evaluation utilities, to extract essential data on result parameters from OpenFOAM® text file output.
- Device of a semi-analytical scheme to verify solver functionality and result quality, within the framework of simplified fibre geometries ([4] and chapter 8.1).
- An extensive, experimental set up to verify results (chapter 8.3).

Concerning these developments, three papers have been published. Article [3] focuses on the basic concepts as well as the essential drag force implementation method behind the particle model while article [4] mainly concerns itself with the handling of particle interaction with their surroundings. In addition to that [4] deals with the creation of a simplified, semi-analytical approach to verify solver functionality and result quality. The third article [5] presents an adaptive time stepping scheme for explicit Euler, temporal discretization of (non-)spherical particle movement.

4.2.1 Going from Spherical to Non-Spherical Particles

A supposedly insignificant upgrade of a spherical particle model to a full non-spherical approach soon turns out to be quite demanding. In direct comparison to the simpler, spherical model, the following aspects have to be considered:

- Three translational DOF have to be extended by three additional, rotational DOFs.
- Particle shape is not just characterized by one parameter, the diameter, but by three parameters, the half axes.
- Particle position *and* orientation will have to be known. Therefore an additional, co-rotational coordinate system, being aligned along the particle's main axes will have to be introduced.
- To calculate inertial effects, not just the particle mass, but also particle moments of inertia have to be considered.
- Not only forces, but also torques are relevant.
- Drag force implementation can not use standard, semi-empirical correlations for spheres.
- Qualitative particle impact modelling gets more complex since impact conditions can not be formulated that easily.
- Quantitative particle impact modelling becomes more complex since exact impact spots will have to be known to calculate resulting torques.

4.2.2 Crucial Concepts and Implementation Schemes

This chapter lists some basic concepts and innovative implementation schemes that had to be chosen and/or developed in order to create a suitable framework for the non-spherical particle model.

4.2.2.1 Particle Geometry: Ellipsoid Shape

The non-spherical particle shape representation is chosen to be an ellipsoid with three independent, geometrical degrees of freedom, - the three axis diameters: D_a, D_b, D_c or half axis diameters a, b, c. Note that, due to the three geometrical DOFs, there is no general rotational symmetry. The ellipsoid shape is selected in the awareness that there are still many arbitrarily shaped particle forms that can hardly be represented by a smooth ellipsoid. However, this choice constitutes a reasonable compromise between benefits and costs. It offers the versatility to approximate many shapes from sticks to plates (see Figure 26) on the one hand and can be mathematically described pretty easily, as seen in Equ.48, on the other hand [68].

$$\left(\frac{x'}{a}\right)^2 + \left(\frac{y'}{b}\right)^2 + \left(\frac{z'}{c}\right)^2 = 1 \qquad (48)$$

Here x', y' and z' are coordinates of a Lagrangian, co-rotational coordinate system, with base vectors being aligned along the particle's main axes (see chapter 4.2.2.2).

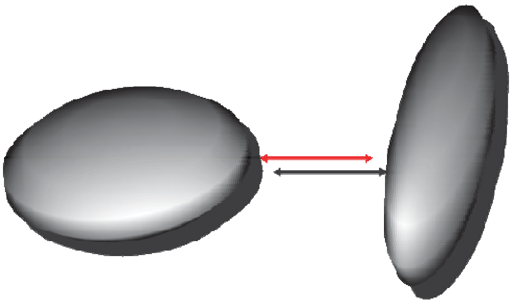


Figure 26: The ellipsoid shape can approximate a wide variety of geometries, e.g. plates and sticks.

The ellipsoid particle volume V_p is given by [9]:

$$V_p = \frac{4}{3}\pi \cdot abc \qquad (49)$$

An essential quantity for calculating skin friction forces on the particle is the particle surface area A_p. In this work A_p for ellipsoids is approximated by a comparatively simple formula, proposed by Thomsen [5]:

$$A_p \approx 4 \cdot \pi \cdot \left[\frac{1}{3} \left(a^p b^p + a^p c^p + b^p c^p \right) \right]^{\frac{1}{p}} \qquad (50)$$

With $p \approx 1.6075$, this formula is reported to yield a maximum of +/- 1.061 % deviation about the correct result.

It should be noted that Rosdahl [8] proposed the *super-elliptic* shape (Equ.51) as very versatile, non-spherical particle representation to be used in numerical solvers. The super-ellipsoid is rotationally symmetrical around its main axis and can also be defined by three independent geometrical DOFs, a (or b), β and n. It is described by the following shape function:

$$\begin{aligned} &\left(\frac{x'}{a} \right)^n + \left(\frac{y'}{b} \right)^n = 1 \\ &n \geq 2.0 \\ &\frac{b}{a} = \beta \geq 1 \end{aligned} \qquad (51)$$

The underlying programming structure of the particle solver, is strictly modular and such, that minor future adoptions could easily introduce e.g. the super-elliptic shape function instead of a standard ellipsoid.

4.2.2.2 Euler and Lagrange Coordinate System

The fluid and FSI calculations are based upon well known, Eulerian principles and require only one Cartesian coordinate system, with base vectors e_x, e_y and e_z and coordinates, x, y and z. In the course of the FSI calculation however, the fluid mesh actually works as Lagrangian mesh that adjusts itself to displaced fibres. Whereas the fibre structure mesh itself retains its original topology [1].

For the particle calculation, the partly Lagrangian character of the fluid mesh is completely irrelevant. The particle solver does not require separate meshing nor mesh movement. To account for particle position X_p *and* orientation, an additional co-rotational coordinate system is introduced. The particle coordinate system, with base vectors e_{px}, e_{py} and e_{pz} being aligned along the main particle axes, as seen in Figure 27, originates from the particle mass centre. Its coordinates are written as x', y' and z'.

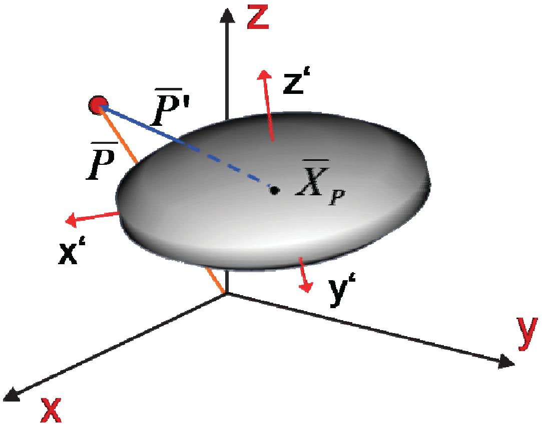


Figure 27: Exemplary ellipsoid particle with co-rotational coordinate system.

A similar *multiple coordinate system* approach is used by Rosdahl [8]. His solver uses a third, additional, co-moving coordinate system, which also originates from the mass centre of the particle and is aligned along the basis of the outer, inertial, Eulerian coordinate system.

The relationship of any single point P within the Eulerian system, to the corresponding point P' within the co-rotational, Lagrangian system, is given by the following formula:

$$\overline{P}' = \sum_{n=x,y,z} \left[\left(\overline{P} - \overline{X}_P \right) \cdot \overline{e}_{p,n'} \right] \cdot \overline{e}_n \qquad (53)$$

Here the index n denotes the axis directions x, y and z respectively and the base vectors of the particle coordinate system are given by:

$$\overline{e}_{p,n'} = \left(e_{n',1}, e_{n',2}, e_{n',3} \right) \qquad (54)$$

According to Equ.53 the transformation operation T can be defined as:

$$T\left(\bullet,\overline{X}_P\right)=\overline{\overline{A}}\cdot\left(\bullet-\overline{X}_P\right) \tag{55}$$

With A being the transformation matrix:

$$\overline{\overline{A}}=\begin{bmatrix} e_{x',1} & e_{x',2} & e_{x',3} \\ e_{y',1} & e_{y',2} & e_{y',3} \\ e_{z',1} & e_{z',2} & e_{z',3} \end{bmatrix} \tag{56}$$

Accordingly the re-transformation from P’ to P is computed as:

$$\overline{P}=\sum_{n=x,y,z}\left(\overline{P}^{'}\cdot\overline{e}_n\right)\cdot\overline{e}_{p,n'}+\overline{X}_P \tag{57}$$

The re-transformation operation T’ is formulated by using the transposed transformation matrix A^T:

$$T'\left(\bullet,\overline{X}_P\right)=\left(\overline{\overline{A}}^{T}\cdot\bullet\right)+\overline{X}_P \tag{58}$$

The co-rotational coordinate system helps to simplify the calculation of particle impact events, where the exact point of surface impact, as well as the particle moment of inertia is relevant. A change of angular velocity $\Delta\omega_p$ due to a particle impact force F_I, is always calculated by performing the following steps:

- If point of impact X_I and impact force F_I are already known, transform them into particle coordinates, otherwise transform relevant data such as particle help-point position of impact-partner particle into particle coordinates, to get X_I’, F_I’:

$$T(X_I)=X_I' \qquad T(F_I)=F_I' \tag{59}$$

- Calculate effect on particle (angular) velocity vector within particle inertial system, by using the simple, principal particle moment of inertia tensor I_P' (see chapter 4.2.2.3.2) and the direction of impact force effect n_I' :

$$\Delta \overline{u}_p' = \frac{\overline{F}_I'}{m_p} \cdot \Delta t_p \tag{60a}$$

$$\Delta \overline{\omega}_p' = \frac{\overline{X}_I' \times \overline{F}_I'}{\overline{\overline{I}}_P' \overline{n}_I' \cdot \overline{n}_I'} \cdot \Delta t_p \tag{60b}$$

- Re-transform new particle (angular) velocity vector into fluid coordinates:

$$T'(\omega_p')=\omega_p \tag{61}$$

Translation and rotation during time step Δt_p change the particle centre position X_p^i and the particle base vector orientation $e_{p,n}^i$, at time step i into their new arrangement at time step i+1. Applying a simple Euler discretization, the operations read:

$$\overline{X}_p^{i+1} = \overline{X}_p^i + \overline{u}_p^i \cdot \Delta t_p \tag{62a}$$

$$\overline{e}_{p,n}^{i+1} = \overline{e}_{p,n}^i + \left(\overline{\omega}_p^i \times \overline{e}_{p,n}^i \right) \cdot \Delta t_p \tag{62b}$$

Here u_p^i is the numerical particle velocity and ω_p^i is the numerical, angular velocity vector at time step i.

4.2.2.3 Six Degrees Of Freedom Solver

Non–spherical particle motion in general consists of three degrees of translational as well as three degrees of rotational freedom. This is why the original, spherical particle solver had to be transformed into a more general, six DOF solver.

4.2.2.3.1 The Lagrangian Equations of Motion for Ellipsoids

Considering all N external forces F_j, that act on the particle, the three translational, Lagrangian equations of motion for ellipsoids, can be written in vectorial form as:

$$\frac{d\overline{u}_p}{dt} = \frac{3}{4\pi\rho_p abc} \cdot \sum_{j=1}^{N} \overline{F}_j \qquad (63)$$

Thus Equ.63 is the specialization of Equ.19 for ellipsoids.

The specialization of Equ.21 for ellipsoid objects is not that simple. A change of angular velocity can be expressed by the three rotational equations of motion, where $I_{p,j}$ is the particle moment of inertia around the rotational axis n_j of any acting torque vector:

$$\frac{d\overline{\omega}_p}{dt} = \sum_j \frac{1}{I_{p,j}(\overline{n}_j)}\left(\overline{r}_j \times \overline{F}_j\right) \qquad (64)$$

The rotational axis of a single torque vector, is perpendicular to the acting force direction and to the direction of the lever, r_j. Thus a separate calculation of each particle moment of inertia $I_{p,j}$ for each torque effect r_j x F_j is necessary. An introduction of the particle moment of inertia tensor I_p helps to simplify the problem.

4.2.2.3.2 Moment of Inertia Tensor

In its generalized form the moment of inertia tensor can be written as [63]:

$$\bar{\bar{\mathrm{I}}}_p = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix} \tag{65}$$

Using a continuous, spatial density $\rho_{sp}(x, y, z)$, with V_{sp} describing the local space that completely encompasses the object, with r_k being the distance vector to the axis of rotation, E standing for the identity matrix and with $\otimes$ denoting the outer product, all tensor elements are defined by [65], [67]:

$$\bar{\bar{I}}_p = \iiint_{V_{sp}} \rho_{sp}(x,y,z)\left(\bar{r}_k^2 \bar{\bar{E}} - \bar{r}_k \otimes \bar{r}_k\right) dxdydz \tag{66}$$

The inertia tensor is symmetrical in nature, and it can be shown that it is always possible to find a Cartesian coordinate system where the off-diagonal elements vanish. The remaining, main diagonal elements I_x, I_y, I_z are then called the *principle moments of inertia*. Consequentially the principle moment of inertia tensor is:

$$\bar{\bar{\mathrm{I}}}_p = \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \tag{65}$$

To reduce the more general tensor formulation to the more specialized, scalar moment of inertia $I_{p,j}$, which is needed to express each torque effect $r_j \times F_j$ according to Equ.64, the following form can be used:

$$I_{p,j} = \left(\bar{\bar{I}}_p \cdot \bar{n}_j\right) \cdot \bar{n}_j \tag{67}$$

The expressions for standard, ellipsoid, *principle moments of inertia* read [64]:

$$I_x = \left(b^2 + c^2\right) \cdot \frac{m_p}{5} \tag{68}$$

$$I_y = \left(a^2 + c^2\right) \cdot \frac{m_p}{5} \quad (69)$$

$$I_z = \left(a^2 + b^2\right) \cdot \frac{m_p}{5} \quad (70)$$

Inserting Equ.68 to Equ.70 into Equ.65 and Equ.66 this amounts to:

$$I_{p,j} = \frac{m_p}{5}\left[\left(b^2 + c^2\right)n_{j,x}^2 + \left(a^2 + c^2\right)n_{j,y}^2 + \left(a^2 + b^2\right)n_{j,z}^2\right] \quad (71)$$

4.2.2.4 Non-Spherical Particle Shape Concepts

In order to consider rotational effects, collision–impact scenarios or other shape-related phenomena, the moving object has to extend beyond a simple, point-like representation. Thus the surface *help-point method (chapter 4.2.2.4.1)*, as well as a simple *panel method* (chapter 4.2.2.4.2) to discretize the particle surface are introduced.

4.2.2.4.1 Non-Spherical Surface and Pressure/Velocity Help Points

Within the advanced (non-)spherical solver a cloud of up to *M*= 68 help points per particle is used (see Figure 28). 18 surface help points are positioned directly at the surface of the particle to serve as collision detectors *and* pressure/velocity probes. An additional 48 *pressure/velocity help points* are located at crucial positions of the particle–panel model and detect local fluid field conditions. The help points surround the ellipsoid at constant positions HP_m‘ within the framework of the Lagrangian, co-rotational particle coordinate system. Thus each help point conserves its relative position to the particle centre, while the particle moves arbitrarily within the Eulerian fluid domain, “dragging along” the Eulerian help point positions HP_m.

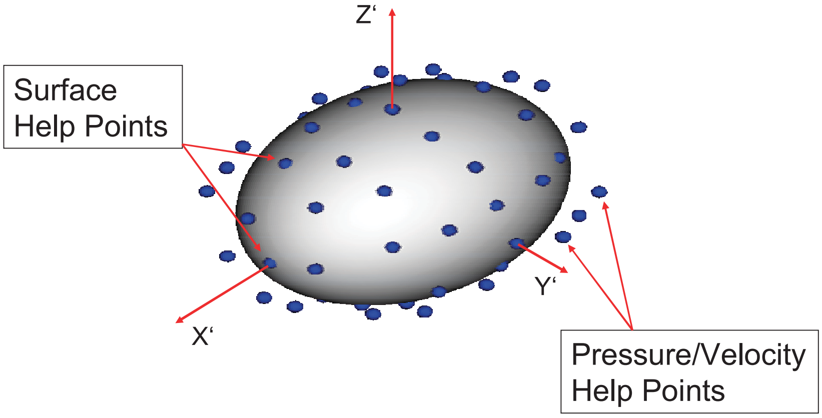


Figure 28: Non-spherical particle with 18 *surface help points* and 48 *pressure-/velocity help points*.

The help points essentially serve two purposes:

- In their function as pressure/velocity help points they detect local fluid conditions.
- They predefine the current particle movement by tracking the individual, projected trajectory, given by Equ.72 and by detecting any collision that might occur along this course.

Using a simple, temporal Euler discretization [5], the help point position HP_m^i at time i can be projected to its new position HP_m^{i+1}, at time i+1, after particle time step Δt_p. The new position is then:

$$\overline{HP}_m^{i+1} = \overline{HP}_m^{i} + \left(\overline{\omega}_p \times \overline{r}_{hp} + \overline{u}_p\right) \cdot \Delta t_p \qquad (72)$$

Here r_{hp} is the help point distance vector to the particle mass centre. This particle progression scheme is only used if collision events are to be expected. The linear trajectory $\overline{HP_m^{i+1} HP_m^{i}}$ is probed for obstacles. If a collision occurs at position X_{coll} before HP_m^{i+1} is reached, the help-point-specific fraction f_m is set to:

$$f_m = \frac{\left|\overline{X}_{coll} - \overline{HP}_m^{i}\right|}{\left|\overline{HP}_m^{i+1} - \overline{HP}_m^{i}\right|} \tag{73}$$

Then the new particle time step Δt_f^* is calculated using the minimum fraction of all help points:

$$\Delta t_f^* = \min\left(f_m\right)\cdot \Delta t_f \tag{74}$$

Now the actual particle movement is conducted.

If no collision events are to be expected, the solver uses an alternative particle progression scheme. The alternative scheme works by conducting translational and rotational operations merely on the particle mass centre X_p and on the particle base vectors $e_{p,n}$ as seen in Equ.62. To find the new help point positions at time i+1, a simple coordinate transformation suffices:

$$\overline{HP}_m^{i+1} = \sum_{n=x,y,z}\left(\overline{HP}'_m \cdot \overline{e}_n\right)\cdot \overline{e}_{p,n}^{i+1} + \overline{X}_p^{i+1} \tag{75}$$

Note the fact that the co-rotational help point positions HP_m‘ remain unchanged at all times.

4.2.2.4.2 Panel Method

While the *surface help point scheme* has been designed to aid in the modelling of collisions and in the detection of local flow field conditions, a simple panel method is introduced to get a hold of hydrodynamic drag and lift forces. Within the co-rotational coordinate system, the fixed help point positions are used as a framework to encase the ellipsoid with a system of edges and panels (see Figure 29).

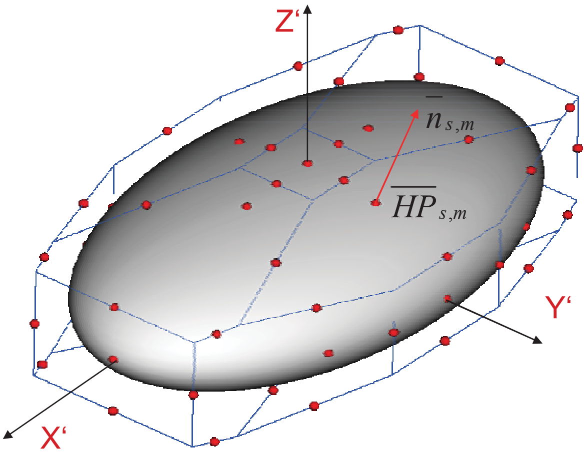


Figure 29: Non-spherical particle surrounded by help points and panels.

18 surface help points $HP_{s,m}$ are used. At their position each panel touches the ellipsoid and is positioned within a tangential plane, perpendicular to the local, ellipsoid surface-normal-vector $n_{s,m}$. Surface help points 1 to 6 ($HP_{s,1-6}$) are assigned to each one of the ellipsoid poles. Surface help points 7 to 18 ($HP_{s,7-18}$) are positioned within the principal planes of the ellipsoid, such that only two non-zero coordinate components, either X_{ell}, Y_{ell} or Z_{ell} exist. In addition to that arc $\tan[(HP_{s,7-18}{}'\ e_{p,j})/(HP_{s,7-18}{}'\ e_{p,i})]$ with $i \neq j$ always yields either $\pi/4$ or *zero* permutated for i=x’, y’ and z’. Thus the coordinates of the 18 surface help points can be calculated as seen in Eqn.76 to Eqn.79:

$$HP_{s,1-6} = \left({}^{+}\!/\!_{-}a \wedge 0, {}^{+}\!/\!_{-}b \wedge 0, {}^{+}\!/\!_{-}c \wedge 0\right) \tag{76}$$

$$HP_{s,7-10} = \left(0, {}^{+}\!/\!_{-}Z_{ell}, {}^{+}\!/\!_{-}Z_{ell}\right) \tag{77}$$

$$HP_{s,11-14} = \left({}^{+}\!/\!_{-}Y_{ell}, {}^{+}\!/\!_{-}Y_{ell}, 0\right) \tag{78}$$

$$HP_{s,15-18} = \left({}^{+}\!/\!_{-}X_{ell}, 0, {}^{+}\!/\!_{-}X_{ell}\right) \tag{79}$$

Hereby X_{ell}, Y_{ell} and Z_{ell} are given as:

$$X_{ell} = \frac{a \cdot c}{\sqrt{a^2 + c^2}} \tag{80}$$

$$Y_{ell} = \frac{a \cdot b}{\sqrt{a^2 + b^2}} \tag{81}$$

$$Z_{ell} = \frac{b \cdot c}{\sqrt{b^2 + c^2}} \tag{82}$$

Figure 30 shows the distribution of all $HP_{s,\,1\text{-}18}$ and the corresponding coordinate components. Note that all help point positioning is primarily conducted within the co-rotational particle coordinate system, only once within the entire simulation run.

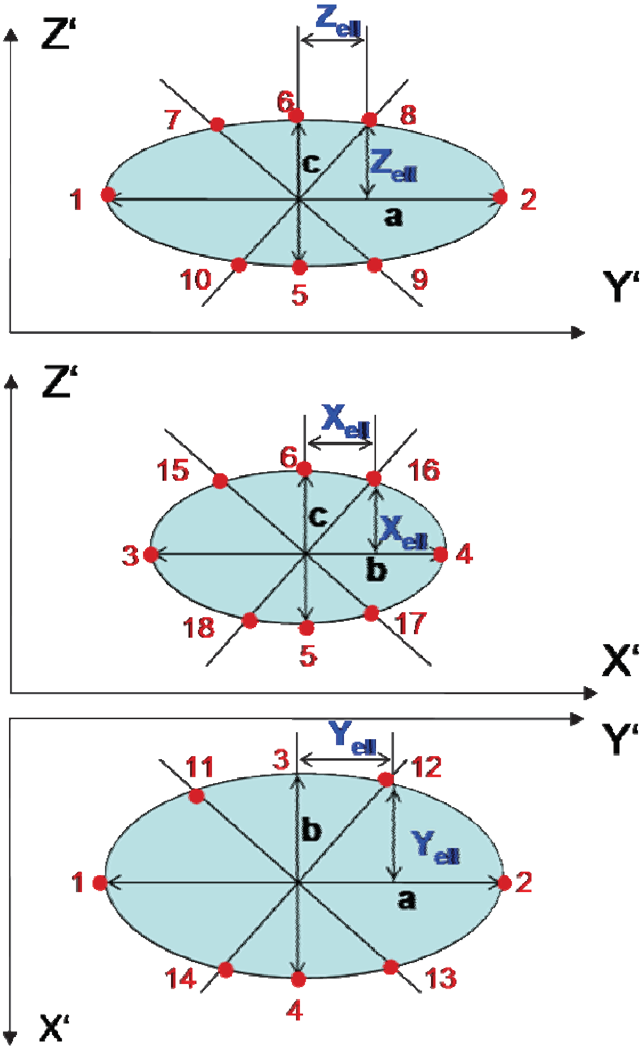


Figure 30: Positions of the 18 surface help points (red numbering) within the co-rotational particle coordinate system.

The exact calculation of many particle interaction events (e.g. collisions) requires knowledge of the local help point particle surface normal vectors $n_{p,m}$ with m=1-18. Exact $n_{p,m}$ coordinates are given by:

$$\vec{n}_{p,1-6} = \left({}^{+}\!/_{-}1 \wedge 0, {}^{+}\!/_{-}1 \wedge 0, {}^{+}\!/_{-}1 \wedge 0\right) \tag{83}$$

$$\vec{n}_{p,7-10} = \left(0, {}^{+}\!/_{-}Z_n, {}^{+}\!/_{-}Z_n\right) \tag{84}$$

$$\vec{n}_{p,11-14} = \left({}^{+}\!/_{-}Y_n, {}^{+}\!/_{-}Y_n, 0\right) \tag{85}$$

$$\vec{n}_{p,15-18} = \left({}^{+}\!/_{-}X_n, 0, {}^{+}\!/_{-}X_n\right) \tag{86}$$

Where X_n, Y_n and Z_n are:

$$X_n = \frac{\partial z'}{\partial x'} = \frac{c}{a^2} \cdot \frac{X_{ell}}{\sqrt{1 - \frac{X_{ell}{}^2}{a^2}}} \tag{87}$$

$$Y_n = \frac{\partial x'}{\partial y'} = \frac{a}{b^2} \cdot \frac{Y_{ell}}{\sqrt{1 - \frac{Y_{ell}{}^2}{b^2}}} \tag{88}$$

$$Z_n = \frac{\partial y'}{\partial z'} = \frac{b}{c^2} \cdot \frac{Z_{ell}}{\sqrt{1 - \frac{Z_{ell}{}^2}{c^2}}} \tag{89}$$

Figure 31 shows the exemplary position, orientation and relation to $\partial z'/\partial x'$, of the surface normal vector $n_{p,16}$.

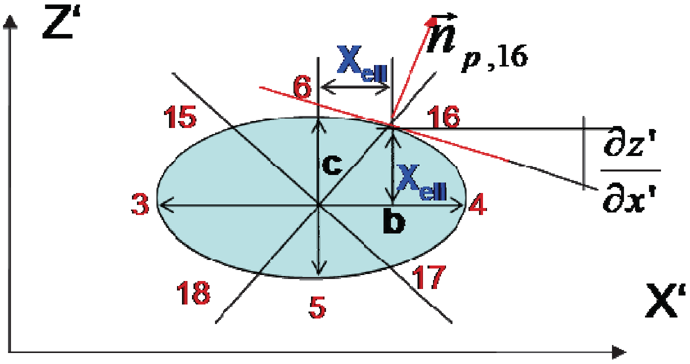


Figure 31: Position and orientation of surface normal vector $n_{p,16}$.

24 additional help points, hereby denoted as P_m, are used to mark panel corners at the principal axes of the ellipsoid or surrounding the 6 poles. Their coordinates are calculated as:

$$P_{1-8} = \left(\pm\left(X_{ell} + \frac{X_{ell} - c}{X_n} \right), \pm\left(Z_n \left(Z_{ell} - c \right) + Z_{ell} \right), \pm c \right) \tag{90}$$

$$P_{9-16} = \left(\pm\left(Y_n \left(Y_{ell} - b \right) + Y_{ell} \right), \pm b, \pm\left(Z_{ell} + \frac{Z_{ell} - b}{Z_n} \right) \right) \tag{91}$$

$$P_{17-24} = \left(\pm a, \pm\left(Y_{ell} + \frac{Y_{ell} - a}{Y_n} \right), \pm\left(X_n \left(X_{ell} - a \right) + X_{ell} \right) \right) \tag{92}$$

In addition to that, 8 meeting points M_m of panel edges have to be determined. They are given by:

$$M_{1-8} = \left(\pm X_m, \pm Y_m, \pm Z_m \right) \tag{93}$$

Where X_m, Y_m and Z_m are:

$$X_m = \frac{X_{ell} Y_n Z_n \left(X_n + 1 \right) + Y_{ell} \left(Y_n + 1 \right) - Z_{ell} Y_n \left(Z_n + 1 \right)}{X_n Y_n Z_n + 1} \tag{94}$$

$$Y_m = \frac{-Y_{ell}X_nZ_n(Y_n+1)-Z_{ell}(Z_n+1)+X_{ell}Z_n(X_n+1)}{X_nY_nZ_n+1} \quad (95)$$

$$Z_m = \frac{Z_{ell}X_nY_n(Z_n+1)+X_{ell}(X_n+1)-Y_{ell}X_n(Y_n+1)}{X_nY_nZ_n+1} \quad (96)$$

12 exemplary P-points and 4 exemplary M-points are shown in Figure 32. Neither P-points nor M–points serve as actual *surface* or *pressure/velocity help points*, but have only geometrical and visualization purposes. While surface help points are located within panel surfaces, the *pressure/velocity help points* are positioned at the centre of each defining edge of the structure.

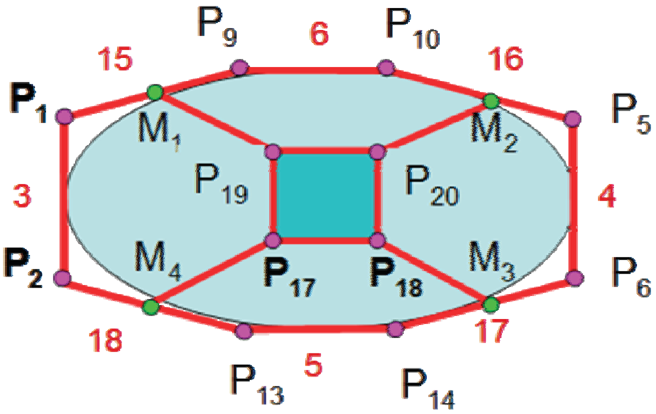


Figure 32: Side view of particle with panels, M/P-points and HP-points (red numbering). A total of 24 P-points and 8 M-points per particle serve as geometrical framework to define panel edges as seen in Figure 29.

The importance of the panels lies within their application to grasp free flow hydrodynamic as well as viscous drag and lift forces. Therefore the panel orientation $n_{s,m}$ and the panel surface area A_m, with m=1-18, are relevant. The panel normal vectors correspond with the HP-point normal vectors so that:

$$\vec{n}_{s,m} = \vec{n}_{p,m} \quad (97)$$

Following the defining M/P-points, the panel surface area is given by either Equ.98 or Equ.99.

$$\bigwedge_{a,b,c\le 24} \qquad A_m = \left|\overline{P_b P_a} \times \overline{P_c P_a}\right| \tag{98}$$

$$\bigwedge_{a,b,c\le 24;d,e\le 8} A_m = \left|\overline{P_a M_d} \times \overline{P_b M_d}\right| + \left|\overline{P_c P_a} \times \overline{P_b P_a}\right| \tag{99}$$

Inserting the corresponding expressions used to calculate M/P-point coordinates the panel surface areas amount to:
6 polar panels,

$$A_{1-2} = 4 \cdot \left(Y_{ell} + \frac{Y_{ell} - a}{Y_n}\right) \cdot \left(X_n\left(X_{ell} - a\right) + X_{ell}\right) \tag{100}$$

$$A_{3-4} = 4 \cdot \left(Z_{ell} + \frac{Z_{ell} - b}{Z_n}\right) \cdot \left(Y_n\left(Y_{ell} - b\right) + Y_{ell}\right) \tag{101}$$

$$A_{5-6} = 4 \cdot \left(X_{ell} + \frac{X_{ell} - c}{X_n}\right) \cdot \left(Z_n\left(Z_{ell} - c\right) + Z_{ell}\right) \tag{102}$$

and 12 lateral panels.

$$A_{7-10} = \left|\overrightarrow{P_{17} - P_3} \times \overrightarrow{P_2 - P_3}\right| + \left|\overrightarrow{P_{18} - M_2} \times \overrightarrow{P_2 - M_2}\right| \tag{103}$$

$$A_{11-15} = \left|\overrightarrow{P_4 - P_3} \times \overrightarrow{P_{10} - P_3}\right| + \left|\overrightarrow{P_3 - M_3} \times \overrightarrow{P_{10} - M_3}\right| \tag{104}$$

$$A_{15-18} = \left|\overrightarrow{P_{17} - P_{10}} \times \overrightarrow{P_{11} - P_{10}}\right| + \left|\overrightarrow{P_{17} - M_3} \times \overrightarrow{P_{10} - M_3}\right| \tag{105}$$

The panel method as a hole seems quite costly, considering that it has to be applied for each individual particle. However, since all help point positioning is conducted within the co-rotational particle coordinate system, each Lagrangian, geometrical particle attribute: HP_m‘, $n_{p,m}$’, P_m, M_m and A_m has to be calculated only once, within the particle constructor. HP_m‘ and $n_{p,m}$’ have to be constantly transformed to their Eulerian coordinates, which change due to translational and angular motion. Figure 33 shows an exemplary screenshot of a bunch of sinking, spherical particles, highlighting their surrounding *help point* and *panel* structure.

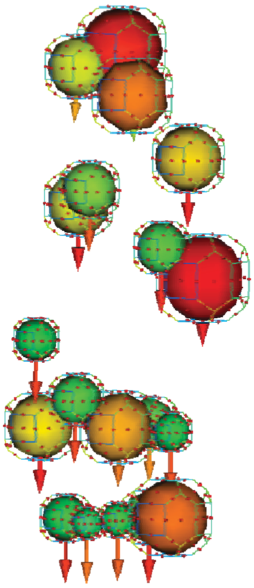


Figure 33: Sinking, spherical particles with velocity vectors as well as highlighted help point and panel structure.

4.2.3 The Particle Momentum Equation behind the Non-Spherical Solver

Applying Newton's second law, the translational PME for arbitrarily shaped particles and arbitrary flow conditions, is given. The full, generalized PME according to [14], presents the framework for all particle-motion modelling behind the non-spherical particle solver. Its implementation is much more refined than the one for the spherical solver (chapter 4.1.1) and can be written as:

$$m_p \frac{d\overline{u}_p}{dt} = \underbrace{\overline{F}_d + \overline{F}_h + \overline{F}_{Magnus} + \overline{F}_{Saffman} + \overline{F}_{Faxen} + \overline{F}_g + \overline{F}_b}_{\boxed{\text{Steady State Forces}}} + \underbrace{\overline{F}_{VM} + \overline{F}_{Basset}}_{\boxed{\text{Unsteady Forces}}} + \underbrace{\sum_{i=1}^{N} \overline{F}_{e,i}}_{\boxed{\text{Event Forces}}} \tag{106}$$

The individual force contributions, summarized in Equ.106 can be divided into three main categories and are denoted as follows [7], [15],and [16]:

Steady state forces:

- Drag force F_d: is the sum of the components of total form drag F_p due to pressure gradients and total shear drag F_τ due to viscous forces parallel to the main, relative flow direction. Calculated for uniform flow conditions and non-rotating, non-accelerating objects.
- Hydrodynamic lift force F_h: based upon unsymmetrical fluid deviation around arbitrarily shaped objects (e.g.: air foil) immersed in uniform flow field. It is *zero* for objects with symmetry plane parallel to relative particle–fluid velocity (e.g.: sphere). It is the sum of the components of total form drag due to pressure gradients and total shear drag due to viscous forces vertical to the main, relative flow direction. Calculated for uniform flow conditions and non-rotating, non-accelerating objects.
- Magnus force F_{Magnus}: additional hydrodynamic lift force based upon particle rotation within the fluid.
- Saffman force $F_{Saffman}$: additional hydrodynamic lift force based upon shear stress gradients across particle surface due to rotating, non-uniform flow field.
- Faxen force F_{Faxen}: corrects drag force for non-uniform flow field effects.
- Gravity force F_g: volumetric force proportional to particle mass.
- Buoyancy force F_b: based upon hydro-static pressure gradient across particle surface.

Unsteady forces:

- Added (virtual) mass effect F_{VM}: correction of particle inertia because of fluid mass that is accelerated / decelerated with particle. Relevant for high acceleration and non-coupled particle-fluid systems.
- Basset (history) force F_{Basset}: relevant for relatively accelerating/decelerating particles. Accounts for non-spontaneous

boundary layer adjustments due to viscosity. Relevant for highly unstationary flow conditions.

Event (Impact) forces:

- Particle–wall interaction F_{Wall}: models particle–wall impact of user definable elasticity.
- Particle–fibre interaction F_{Fibre}: models particle–fibre impact, adhesion forces and deposition with user definable probability variables.
- Particle-particle interaction $F_{Particle}$: models particle-particle collision scenarios of user-definable elasticity.

The specialization of Equ.106 for small, spherical particles, that are immerged into a uniformly flowing fluid, gives the *Basset Bousinesque Oseen* (BBO) equation [14], [26], without Faxen terms nor interaction with solids nor other particles. In this case the individual force contributions can be formulated as seen in Table 1.

Steady forces		Formulation
	Drag force	$\overline{F}_d = c_d\left(\mathrm{Re}_p\right)\frac{\pi}{4}D_{sph}^2\frac{\rho_f}{2}\left\lvert\overline{u}_f-\overline{u}_p\right\rvert\left(\overline{u}_f-\overline{u}_p\right)$
	Hydrodynamic lift force	$\overline{F}_h = \overline{0}$
	Magnus force	$\overline{F}_{Magnus} = \frac{\pi}{8}D_{sph}^3\rho_f\left(\left(\frac{1}{2}\nabla\times\overline{u}_f-\overline{\omega}_p\right)\times\left(\overline{u}_f-\overline{u}_p\right)\right)$
	Saffman force	$\overline{F}_{Saffman} = 1{,}615D_{sph}^2\frac{-\sqrt{\rho_f\nu_f}}{\sqrt{\left\lvert\nabla\times\overline{u}_f\right\rvert}}\left(\left(\nabla\times\overline{u}_f-\overline{\omega}_p\right)\times\left(\overline{u}_f-\overline{u}_p\right)\right)$
	Gravity force	$\overline{F}_g = \frac{\pi}{6}D_{sph}^3\rho_p\overline{g}$
	Buoyancy force	$\overline{F}_b = -\frac{\pi}{6}D_{sph}^3\rho_f\overline{g}$
Unsteady forces		
	Added mass	$\overline{F}_{Vm} = \frac{\pi}{12}D_{sph}^3\rho_f\left(\dot{\overline{u}}_f-\dot{\overline{u}}_p\right)$
	Basset force	$\overline{F}_{Basset} = \frac{3}{2}D_{sph}^2\sqrt{\pi\rho_f\nu_f}\left[\int\frac{\left(\dot{\overline{u}}_f-\dot{\overline{u}}_p\right)}{\sqrt{t-t'}}dt'+\frac{\left(\overline{u}_f-\overline{u}_p\right)_0}{\sqrt{t}}\right]$
Event forces		
	wall, particle, fibre interaction	$\sum_{i=1}^{N}\overline{F}_{e,i} = \overline{0}$

Table 1: Formulation of force contributions as they would look like in order to turn the PME according to Equ.106 into the classic BBO equation for small, spherical, non-coupled particles.

Usually a PME formulation like the one in Equ.106 is used for small particles and the classical Euler-Lagrange approach [25]. This work however, treats large particles that span multiple fluid cells, and still retains the typical Euler-Lagrange methodology. Thus a specifically adjusted, numerical scheme to model particle-fluid interaction becomes necessary.

Comparable programs, like that of Schütz [9], are using particle-related re-meshing of the fluid grid, which is duly avoided here.

4.2.4 Non-Spherical Particle–Fluid Interaction

To maximize calculation efficiency, a detailed drag implementation, specifically adapted to the case of non-spherical dirt particle filtration in lubricants has been created. The particle–fluid interaction model consists of two alternative modules:

- The *Free flow* particle–fluid module
- The *Fibre vicinity* particle–fluid module

Dirt particles are injected into the *free flow* regime upstream of the filter fibre geometry, where they occur in very low volume fractions. Particle-particle interaction and hydrodynamic particle impact on the fluid can be neglected here. As soon as the particles reach the *fibre vicinity*, the two-way coupling takes effect and inter-particle as well as full particle-fluid interaction becomes relevant. Those fundamentally different situations require separate drag modelling schemes in order to guarantee a good balance between accuracy and efficiency.

4.2.5 Free Flow Particle-Fluid Interaction Module

Within the *free flow* regime, all particle interactions with their surroundings are handled by the *free flow* module. In this zone, the most important aspects of the prevailing hydrodynamic situation are:

- The ratio between particle diameter D_{sph} and minimal distance to the nearest fibre (wall) boundary patch h_w can be considered as *small*. Thus wall proximity has no effect on particle drag.

- Due to very low particle volume fractions, the ratio between D_{sph} and the minimal distance between neighbouring particles h_p can be considered as *small*. No neither physical, nor hydrodynamic particle interaction takes place.

- Due to very low particle Reynolds number and very low particle volume fractions, the hydrodynamic particle effect on the fluid can be neglected. No *two-way-coupling* is necessary.

In the *free flow* regime it is primarily important to grasp torques, acting on the particle. Rotational effects due to non-uniform flow fields can lead to a pre-alignment of the particles, so that average penetration depth and filter fibre efficiencies are being influenced (see chapter 2.6). In this context the panel description (see chapter 4.2.2.4.2) of the ellipsoid shape is of special importance. The particle is enclosed by M panels and each panel j is subject to drag forces $F_{d,j}$ (which consist of pressure and shear flow contribution, $F^*_{p,j}$ and $F^*_{\tau,j}$ respectively) and hydrodynamic lift forces $F_{h,j}$. Note that $F_{h,j}$ can also be traced back to pressure $F^{**}_{p,j}$ and shear stress contributions $F^{**}_{\tau,j}$. Therefore the total form drag is $F_{p,j}=F^*_{p,j}+F^{**}_{p,j}$ and the total shear effect is $F_{\tau,j}=F^*_{\tau,j}+F^{**}_{\tau,j}$. However, for the sake of simplicity $F_{h,j}$ is hereby written out and calculated as one single force contribution term.

Forces which are better calculated by considering the entire particle are: gravity F_g, buoyancy F_b and N event forces $\Sigma F_{e,i}$. Thus the adapted PME within the *free flow* regime looks like Equ.107.

$$m_p \frac{d\overline{u}_p}{dt} = \sum_{j=1}^{M}\left(\overline{F}^*_{p,j} + \overline{F}^*_{\tau,j} + \overline{F}_{h,j}\right) + \overline{F}_g + \overline{F}_b + \sum_{i=1}^{N}\overline{F}_{e,i} \tag{107}$$

In analogy, torque effects on non-spherical rotation are described by:

$$I_p \frac{d\overline{\omega}_p}{dt} = \sum_{j=1}^{M}\left(\overline{r}_j \times \overline{F}^*_{p,j} + \overline{r}_j \times \overline{F}^*_{\tau,j} + \overline{r}_j \times \overline{F}_{h,j}\right) + \sum_{i=1}^{N}\overline{r}_i \times \overline{F}_{e,i} \tag{108}$$

Here r_j stands for the distance vector of each surface panel centre HP_j to the particle mass centre X_p and r_i denotes the distance vector from X_p to any particle help point HP_i that senses an impact event.

The Basset history force and the added, virtual mass are being neglected because of the lack of strongly un-stationary, relative particle-fluid flow. Comparing Equ.107 with Equ.106, the following parallels can be drawn:

$$\begin{aligned} \overline{F}_d &= \overline{F}_\tau^* + \overline{F}_p^* \\ \overline{F}_{Saffman} &\subseteq \left(\overline{F}_\tau^* \cup \overline{F}_h\right) \\ \overline{F}_{Magnus} &\subseteq \left(\overline{F}_\tau^* \cup \overline{F}_p^* \cup \overline{F}_h\right) \\ \overline{F}_{Faxen} &\subseteq \left(\overline{F}_\tau^* \cup \overline{F}_p^* \cup \overline{F}_h\right) \end{aligned} \tag{109}$$

Figure 34 shows a sketch of how the individual force contributions act on each panel and affect the particle.

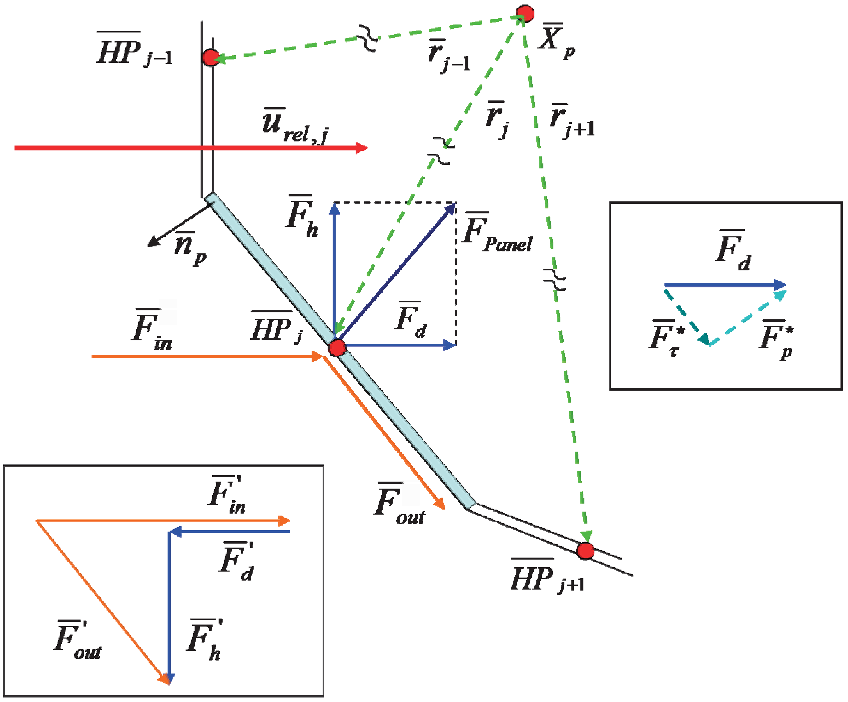


Figure 34: Sketch of local force balance and force effect on panel centre.

In Figure 34, F_{in} stands for the force contribution of the incoming stream, while F_{out} is the force contribution of the outgoing stream as it would look like if it were deviated by the panel surface (which it is not because of the non-coupled *free flow* momentum scheme). The total force acting on each panel F_{panel} is given by:

$$\overline{F}_{panel} = \overline{F}_d + \overline{F}_h \tag{110}$$

The following sub chapters describe the procedural calculation of *free flow* drag and lift force as well as torque effects within the *module*.

4.2.5.1 Free Flow Module Force Calculation

First the drag force contribution $F_{d,j}$ on each panel j has to be calculated. The drag force term consists of a form drag and a shear drag contribution, $F^*_{p,j}$ and $F^*_{\tau,j}$. The overall sum of drag force contribution and hydrodynamic lift contribution consists of $F_{p,j}$ and $F_{\tau,j}$. While $F_{\tau,j}$ and $F^*_{\tau,j}$ act perpendicular to the panel surface normal $n_{p,j}$, $F_{p,j}$ and $F^*_{p,j}$ act parallel to $n_{p,j}$ (see figure 35). Since $F_{d,j}$ is defined to act in the direction of $u_{rel,j}$, the following ratio has to hold:

$$\frac{\left|\overline{F}^*_{\tau,j}\right|}{\left|\overline{F}^*_{p,j}\right|} = \frac{\left|\overline{n}_{p,j} \times \overline{e}_{urel,j}\right|}{\overline{n}_{p,j} \cdot \overline{e}_{urel,j}} \tag{111}$$

Here $e_{urel,j}$ is the base vector of the relative fluid-particle-panel velocity encountered at the panel centre. The total panel drag coefficient $c_{d,panel}$ depends on the form drag coefficient $c_{d,p}$ and on the shear drag coefficient $c_{d,shear}$ and is given by:

$$c_{d,panel} = c_{d,p} + c_{d,shear} \tag{112}$$

In Equ.113 to Equ.117, form drag and shear drag vectors are listed, as well as the total panel drag vector and its dependence on form and shear contribution, $F_{p,j}$* and $F_{\tau,j}$* respectively and the auxiliary expressions $F_{p,j}$*' and $F_{\tau,j}$*'. Figure 35 shows a sketch of the situation.

$$\overline{F}^*_{p,j} = \left(c_{d,p} + c_{d,shear}\right) \cdot A_j \cdot \frac{1}{2}\rho_f \cdot \left|\overline{u}_{rel,j}\right|^2 \cdot \left(\overline{n}_{p,j} \cdot \overline{e}_{urel,j}\right) \cdot \overline{n}_{p,j} \tag{113}$$

$$\overline{F}^*_{\tau,j} = \left(c_{d,p} + c_{d,shear}\right) \cdot A_j \cdot \frac{1}{2}\rho_f \cdot \left|\overline{u}_{rel,j}\right|^2 \cdot \left[\overline{e}_{urel,j} - \left(\overline{n}_{p,j} \cdot \overline{e}_{urel,j}\right) \cdot \overline{n}_{p,j}\right] \tag{114}$$

$$\overline{F}^{*'}_{p,j} = c_{d,p} \cdot A_j \cdot \frac{1}{2}\rho_f \cdot \left|\overline{u}_{rel,j}\right| \cdot \overline{u}_{rel,j} \qquad (115)$$

$$\overline{F}^{*'}_{\tau,j} = c_{d,shear} \cdot A_j \cdot \frac{1}{2}\rho_f \cdot \left|\overline{u}_{rel,j}\right| \cdot \overline{u}_{rel,j} \qquad (116)$$

$$\overline{F}_{d,j} = \overline{F}^{*'}_{p,j} + \overline{F}^{*'}_{\tau,j} = c_{d,panel} \cdot A_j \cdot \frac{1}{2}\rho_f \cdot \left|\overline{u}_{rel,j}\right| \cdot \overline{u}_{rel,j} \qquad (117)$$

In Equ.113 through Equ.117, A_j is the panel surface area and $u_{rel,j}$ is the relative fluid-particle-panel velocity u_f-$u_{p,j}$. The particle-panel velocity is given by the velocity of the particle mass centre u_p and the rotational velocity contribution:

$$\overline{u}_{p,j} = \overline{u}_p + \overline{r}_j \times \overline{\omega}_p \qquad (118)$$

The panel Reynolds number is written as Re_j and is defined by using the hydraulic diameter $d_{h,j}$ of the panel and the kinematic fluid viscosity:

$$\mathrm{Re}_j = \frac{\overline{u}_{rel,j} \cdot d_{h,j}}{\nu_f} \qquad (119)$$

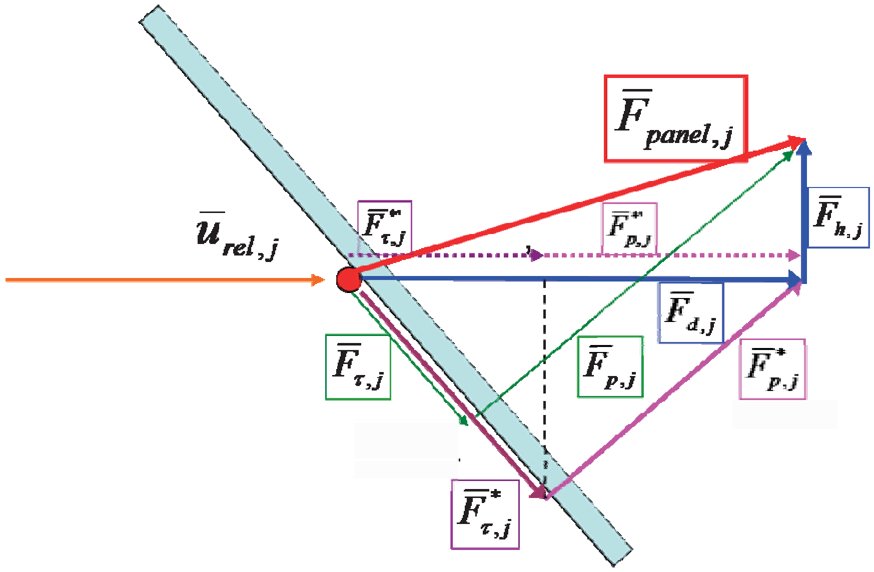


Figure 35: Sketch of complete form and shear force contribution situation to panel drag force.

Secondly the hydrodynamic lift force $F_{h,j}$, which stems from the deviation of the fluid at the panel, is calculated. The hydrodynamic lift is connected to $F_{in,j}$, $F_{out,j}$ and $F_{d,j}$ via a simple, local force balance (see Figure 34):

$$\overline{F}'_{in,j} - \overline{F}'_{d,j} - \overline{F}'_{h,j} = \overline{F}'_{out,j} \qquad (120)$$

Note that from here on the superscript ’ denotes the fact that force values are scaled by the acting surface area A_j and represent the forces acting within and/or onto the fluid. In addition to Equ.120, $F_{h,j}$ is defined to act perpendicular to $F_{d,i}$, so that:

$$\overline{F}'_{h,j} \cdot \overline{F}'_{d,j} = 0 \qquad (121)$$

The drag $F_{d,j}{}'$ is given by Equ.117, while $F_{in}{}'$ can be easily derived out of the local fluid field information, obtained by the *pressure/velocity help points*.

$$\overline{F}'_{in,j} = \frac{\rho_f}{2} \cdot \left|\overline{u}_{rel,j}\right| \overline{u}_{rel,j} \qquad (122)$$

While the value of F_{out}’ is not known in advance, its base vector $e_{out,j}$ is given because of panel orientation $n_{p,j}$ and relative panel-fluid velocity $u_{rel,j}$.

$$\overline{e}_{out,j} = \frac{\left(\overline{u}_{rel,j} - \left(\overline{n}_{p,j} \cdot \overline{u}_{rel,j}\right) \cdot \overline{n}_{p,j}\right)}{\left|\left(\overline{u}_{rel,j} - \left(\overline{n}_{p,j} \cdot \overline{u}_{rel,j}\right) \cdot \overline{n}_{p,j}\right)\right|} \qquad (123)$$

The expressions in Equ.120 and Equ.121 constitute a system of 4 equations and 4 unknowns: The three components of hydrodynamic lift $F_{h,j,x}$, $F_{h,j,y}$, $F_{h,j,z}$ and the absolute value of deviated flow momentum $|F_{out,j}|$. The solution yields the following expressions for the local, hydrodynamic lift force vector $F_{h,j}$‘ and the vector of deviated fluid momentum $F_{out,j}$’:

$$\overline{F}'_{h,j} = \left(1 - \frac{1}{2} c_{d,panel}\right) \rho_f \left|\overline{u}_{rel,j}\right| \left\{ \overline{u}_{rel,j} - \frac{\overline{u}_{rel,j}{}^2}{\overline{u}_{rel,j}{}^2 - \left(\overline{n}_{p,j} \cdot \overline{u}_{rel,j}\right)^2} \left[\overline{u}_{rel,j} - \left(\overline{n}_{p,j} \cdot \overline{u}_{rel,j}\right) \cdot \overline{n}_{p,j}\right] \right\} \qquad (124)$$

$$\overline{F}'_{out,j} = \left(1 - \frac{1}{2} c_{d,panel}\right) \rho_f \left|\overline{u}_{rel,j}\right| \frac{\overline{u}_{rel,j}{}^2}{\overline{u}_{rel,j}{}^2 - \left(\overline{n}_{p,j} \cdot \overline{u}_{rel,j}\right)^2} \left[\overline{u}_{rel,j} - \left(\overline{n}_{p,j} \cdot \overline{u}_{rel,j}\right) \cdot \overline{n}_{p,j}\right] \tag{125}$$

Due to non-coupling, the wake of the particle is not simulated in the *free flow module*. Therefore a panel has to face the stream in order to yield acceptable $F_{d,j}$ and $F_{h,j}$ results. The condition for calculating the individual force balance and for considering the panel is:

$$\overline{u}_{rel,j} \overline{n}_{p,j} \overset{!}{\leq} 0 \tag{126}$$

Consequentially the overall, unscaled drag force F_d^{unsc} and hydrodynamic lift force F_h^{unsc} are given by the contributions of all N considered panels:

$$\overline{F}_d^{unsc} = -\sum_{j=1}^{N} A_j \overline{F}'_{d,j} \tag{127}$$

$$\overline{F}_h^{unsc} = -\sum_{j=1}^{N} A_j \overline{F}'_{h,j} \tag{128}$$

4.2.5.2 Weighing Method and Torque Effect Calculation

The procedure of calculating each force effect on each panel that fulfils condition: Equ.126 is inaccurate because of two reasons:

- A surface description of 24 panels yields a limited, numerical approximation to a smooth particle surface.
- Panels facing the wake of the particle are not considered.

Yet the $F_{d,j}$ and $F_{h,j}$ calculation serves a useful purpose: to get an idea of the force distribution over the particle surface. This is necessary to grasp rotational fluid field effects on the aligning particle.

In order to improve the quantitative estimate on individual drag and lift force contributions, the results are being *scaled* to fit a newly found, empirical drag law for non-spherical particles: Recently Hölzer & Sommerfeld, [24] presented a new, simple relation for drag on non-spherical particles, which has been derived from extensive LB simulations on non-spherical particle shapes of varying sphericity Φ and particle-fluid alignment (length wise and cross wise sphericity, Φ_{length} and Φ_{cross}, respectively). The authors have compared their formula for the drag coefficient, shown in Equ.129, to a wide range of experimental results for spheres, isometric particles, cuboids, cylinders, disks and plates and report mean, relative deviations of 14.1%. This number compares to values of significantly more than 100% for several other non-spherical drag force formulations in use.

$$c_d^{somm} = \frac{3}{\mathrm{Re}_p}\frac{1}{\sqrt{\Phi_{length}}} + \frac{16}{\mathrm{Re}_p}\frac{1}{\sqrt{\Phi}} + \frac{3}{\mathrm{Re}_p}\frac{1}{\Phi^{3/4}} + 0.4210^{0.4(-\log\Phi)^{0.2}}\frac{1}{\Phi_{\mathrm{cross}}} \quad (129)$$

The three particle shape parameters Φ, Φ_{length} and Φ_{cross} are defined by Equ.130, Equ.131 and Equ.132 respectively.

$$\Phi = \frac{A_{sph}}{A_{ell}} \quad (130)$$

A_{sph} signifies the surface area of the volume equivalent sphere and A_{ell} stands for the surface area of the ellipsoid particle.

$$\Phi_{cross} = \frac{A_{f,sph}}{A_{f,ell}} \quad (131)$$

Here $A_{f,sph}$ is the cross sectional area of the volume equivalent sphere and $A_{f,ell}$ is the projected frontal area of the ellipsoid particle.

$$\Phi_{length} = \frac{A_{f,sph}}{\frac{1}{2}A_{ell} - A_{len,ell}} \tag{132}$$

In Equ.132, $A_{len,ell}$ stands for the mean longitudinal (e.g. parallel to the direction of relative flow), projected cross sectional area of the particle [24].

Using the *Hölzer-Sommerfeld* approach, the over all s*caled* drag force F_d^{sc} on the particle can be calculated as:

$$\overline{F}_d^{sc} = \frac{1}{2} c_d^{somm} \rho_f A_f \cdot \left|\overline{u}_{rel,med}\right| \overline{u}_{rel,med} \tag{133}$$

Hereby the average, acting, relative fluid-particle velocity $u_{rel,med}$ is calculated from the individual panel contributions:

$$\overline{u}_{rel,med} = \frac{\sum_{j=1}^{N} A_j \overline{u}_{rel,j}}{\sum_{j=1}^{N} A_j} \tag{134}$$

In order to get a hold of realistic, rotational torque effects, the originally calculated, unscaled force contributions are scaled by the ratio F_d^{sc}/ F_d^{unsc}, so that each panel contribution $F_{d,j}$ and $F_{h,j}$ (as defined by Equ.117 and Equ.124) is transformed to:

$$\overline{F}_{d,j}^{sc} = \frac{\left|\overline{F}_d^{sc}\right|}{\left|\overline{F}_d^{un}\right|}\overline{F}_{d,j} \tag{135}$$

$$\overline{F}_{h,j}^{sc} = \frac{\left|\overline{F}_d^{sc}\right|}{\left|\overline{F}_d^{un}\right|}\overline{F}_{h,j} \tag{136}$$

Then the more accurate, *scaled*, rotational torque effects are computed using Equ.108:

$$I_p \frac{d\overline{\omega}_p}{dt} = \sum_{j=1}^{M}\left(\overline{r}_j \times \overline{F}_{d,j}^{sc} + \overline{r}_j \times \overline{F}_{h,j}^{sc}\right) + \sum_{i=1}^{N}\overline{r}_i \times \overline{F}_{e,i} \tag{137}$$

Figure 36 shows a test case, where a longish, non-spherical particle approaches a small orifice of higher flow velocities and lower pressure. As physically plausible and expected, the given drag implementation models the occurring shear flow and pressure gradients over the particle surface in such a way that the particle aligns itself along the fluid stream lines. The translational and angular velocity vectors adapt to the local fluid field conditions which leads to a particle-slip effect.

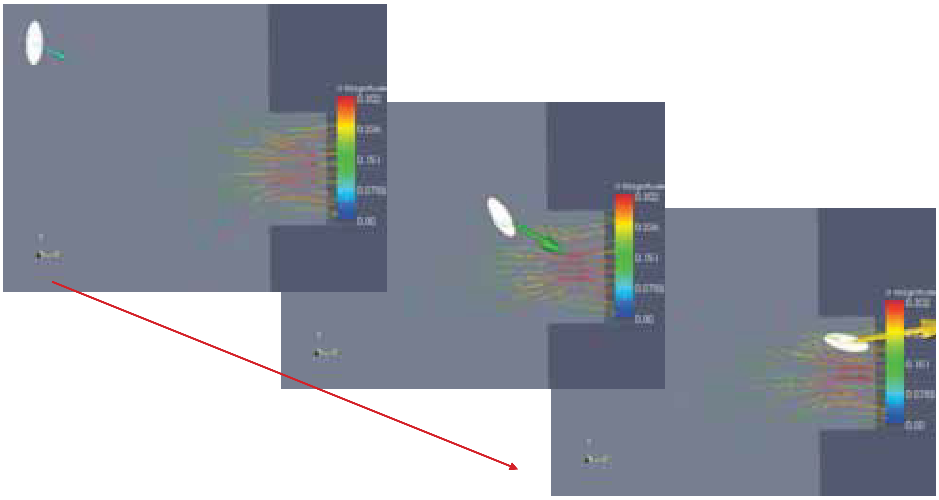


Figure 36: Ellipsoid particle accelerating towards valve. Alignment along the stream lines. Particle takes up its most stable position of least drag and lift forces. This behaviour causes the non-spherical slip effect with relevance for filtration efficiency and particle penetration depth (see chapter 2.6.2).

4.2.6 The Fibre Vicinity Particle-Fluid Interaction Module

As soon as a particle enters into the vicinity of the fibre geometry, the hydrodynamic situation changes completely and the *fibre vicinity drag module* takes over. The situation features the following characteristics:

- Particle–wall flow effects can no longer be neglected since the ratio between particle diameter and minimal particle-wall (fibre) distance h_p is per definition no longer small.

- Particles accumulate at the fibre in considerable volume fractions and the ratio between particle diameter and medium, minimal particle-particle distance h_{pp} is no longer small either. Particles interact hydro-dynamically and physically by plugging each others flow path.

- The high particle volume fractions lead to a plugging of the fluid flow path, diverting the flow and causing increased pressure drop over the filter.

Hydrodynamic particle impact on the fluid (two-way coupling) becomes essential.

Empirical expressions, describing every one of these effects individually, can be found in literature [14], [26]. Still, the dynamic combination of the phenomena to a highly complex, multi-parameter interaction situation can hardly be grabbed by any empirical, non-stochastic formula. Because of that, an approach was developed, that does not require the formulation of individual force contributions, but unites all relevant force terms cited in Equ.106 within expressions of pressure force and shear stress. This is realized by plugging local fluid cells, which are encompassed by the particle (see Fig.36b). The plugging method perceives the fluid mesh as a porous medium, where local permeability is introduced as relevant factor in the *Navier-Stokes* equations. By adding an additional, local pressure gradient via the *Darcy* term, a connection is made to a numeric deposition field which can be directly influenced by particle presence. In principle the procedure is analogous to the one presented in chapter 4.1.2 but has been extended to non-spherical particle shapes, refined and quantified as shall be seen in the following.

4.2.6.1 Fibre Vicinity Module Implementation

The plugging causes the fluid to be diverted around the fluid cell which leads to a local pressure build up p_i, that can be sensed by any of the N *pressure help points* HP_i at the particle surface (see Figure 37). Since each pressure help point represents 1/Nth of the entire particle surface area A_p and since pressure always acts perpendicular to the local particle surface normal $n_{p,i}$, the total pressure force F_p on the particle can be written as:

$$\overline{F}_p = -\sum_{i=1}^{N} p_i a_i \overline{n}_{p,i} \tag{138}$$

For infinitesimally fine grid spacing and an infinitely large number of pressure help points this expression amounts to:

$$\overline{F}_p = -\oint_{Ap} p \cdot \overline{n}_p dA = -\int_{V_p} \nabla p \ dV \tag{139}$$

The second, decisive force contribution results from viscosity effects (see Figure 38). Because of a lack of wall boundary conditions at the border between plugged and unplugged cells, no *"zero velocity"* condition can be introduced at the particle surface. What happens is that an effective *"zero velocity"* condition is imposed along a virtual surface including all cell centres just within the particle borders. Therefore local shear forces $F_{\tau i}$ at the help point positions can be approximated by using the velocity value of the nearest, unplugged fluid cell $u_{f,i}$, at distance $h_{u,i}$ perpendicular to the particle surface. This corresponds to a gradient approximation of 1st order accuracy. The overall shear force F_τ on the particle can thus be calculated as:

$$\overline{F}_\tau = \sum_{i=1}^{N} \mu_f \frac{\left|\overline{u}_{rel,i}\right|}{2 \cdot h_{u,i}} A_i \left(\overline{e}_{urel,i} - \left(\overline{e}_{urel,i} \cdot \overline{n}_{p,i}\right)\right) \tag{140}$$

For infinitesimally fine grid spacing and an infinitely large number of *pressure help points* this expression amounts to:

$$\overline{F}_\tau = \oint_{Ap} \mu_f \cdot \nabla \overline{u}_f \cdot \overline{n}_p dA \tag{141}$$

Where $\nabla \overline{u}_f$ is the Jacobian of u_f. This expression can be expanded and generalized. With τ being the viscous shear stress tensor, this amounts to Equ.142a (vectorial formulation with E being the identity matrix) or Equ.142b (tensorial formulation with m being an additional index and $\delta_{i,j}$ being the *Kronecker delta*):

$$\overline{F}_{\tau} = \oint_{Ap} \overline{\overline{\tau}} \cdot \overline{n}_p dA = \int_{V_p} \nabla \overline{\overline{\tau}} dV = \int_{V_p} \mu_f \left[\nabla \overline{u}_f + \left(\nabla \overline{u}_f \right)^T - \frac{2}{3} \cdot \left(\nabla \bullet \overline{u}_f \right) \overline{\overline{E}} \right] dV \qquad (142a)$$

$$F_{\tau,i} = \int_{V_p} \mu_f \left[\frac{\partial u_{f,j}}{\partial x_i} + \frac{\partial u_{f,i}}{\partial x_j} - \frac{2}{3} \cdot \frac{\partial u_{f,m}}{\partial x_m} \cdot \delta_{i,j} \right] dx_j \qquad (142b)$$

The local hydrodynamic situation resulting from an exemplary, plugging ellipsoid is shown in Figure 36 b.

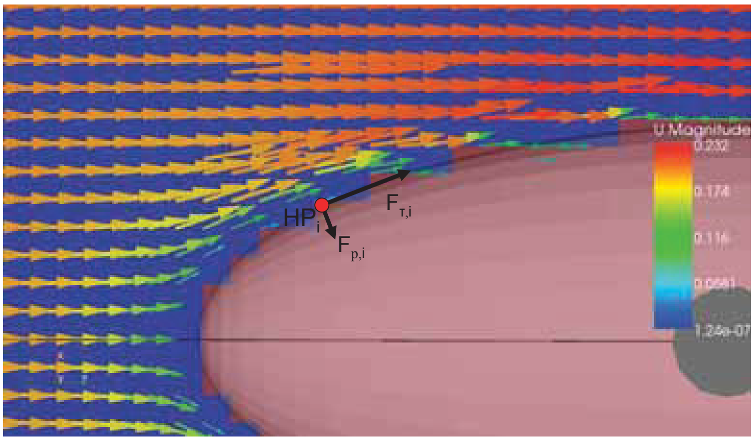


Figure 36 b: Ellipsoid with activated fibre vicinity module. Full two-way coupling is engaged. Flow field deviation around particle. Fluid imposes pressure and shear forces on particle surface.

Pressure and shear stress contributions to the overall drag force on an ellipsoid particle are represented by Figures 37 and 38.

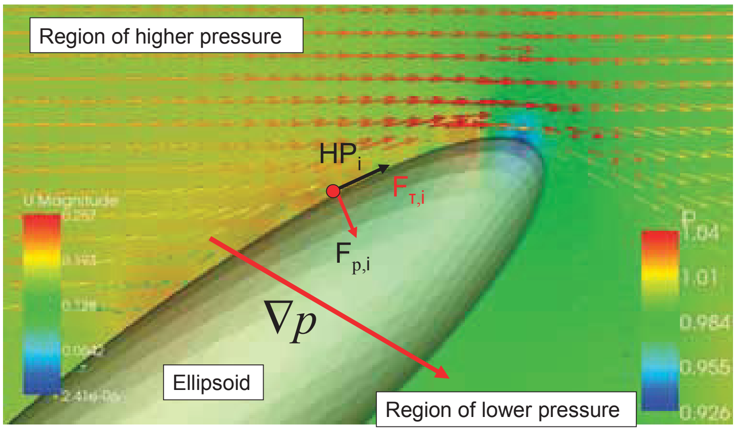


Figure 37: Pressure force contribution to over all fluid-particle force. Exemplary, two-way coupled ellipsoid. Pressure build up in frontal particle area. Formation of pressure gradient across particle surface.

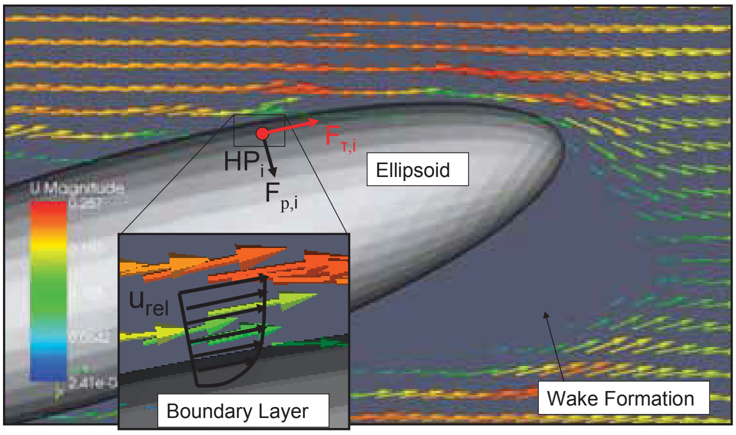


Figure 38: Shear stress contribution to over all fluid-particle force. Exemplary, two-way coupled ellipsoid. Plugging equals a *zero flow* velocity boundary condition at engulfed cell centres. Boundary layer is approximated by 1st order accurate gradient calculation. Shear stresses can be derived.

Figure 39 presents a full screen shot of another, exemplary, large, two-way coupled particle. Here the individual force contributions, resulting from pressure and shear effects are sketched out in more detail.

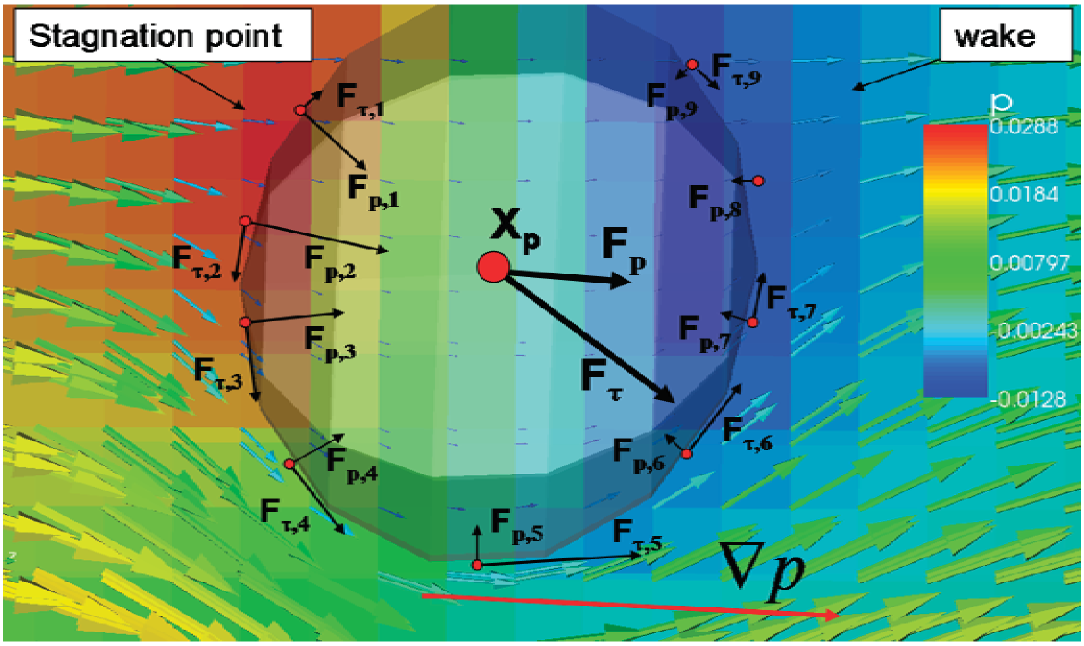


Figure 39: Flow field formation around large, two-way coupled particle. Background coloration gives pressure field. Fluid comes in from the left, pressure gradient forms from forward stagnation point to wake zone. Pressure and shear force contributions at help point positions are sketched out in their qualitative relation to one-another.

As a consequence of Equ.139 and Equ.142 the entire PME for the *fibre vicinity module* can be written as:

$$m_p \frac{d\bar{u}_p}{dt} = -\frac{m_p}{\rho_p}\left(-\nabla p + \nabla \underline{\underline{\tau}}\right) \qquad (143)$$

For a limited number of discretizing surface elements N this expression yields:

$$m_p \frac{d\bar{u}_p}{dt} = -\sum_{i=1}^{N} p_i A_i \bar{n}_i + \sum_{i=1}^{N} \mu_f \frac{\left|\bar{u}_{rel,i}\right|}{2 \cdot h_{u,i}} A_i \cdot \left[\bar{e}_{urel,i} - \left(\bar{e}_{urel,i} \cdot \bar{n}\right)\right] \qquad (144)$$

It has to be stated that this drag and lift force implementation is grid dependent and yields edgy objects with coarse surfaces. The applied meshes however, are structured grids with never changing resolution. An exact knowledge about particle shape and surface structure of actual dirt particles is not given. Yet plausibility commands the following statements to hold:

- Arbitrarily shaped dirt particles rather behave non-spherically than spherically.
- Dirt particles rather have rough surface structures than smooth surfaces.

Therefore the hereby presented simulation approach is considered to be valid.

4.2.6.2 Results and Verification

Figure 40 shows qualitatively how some two-way coupled, non-spherical particles can affect the surrounding fluid flow.

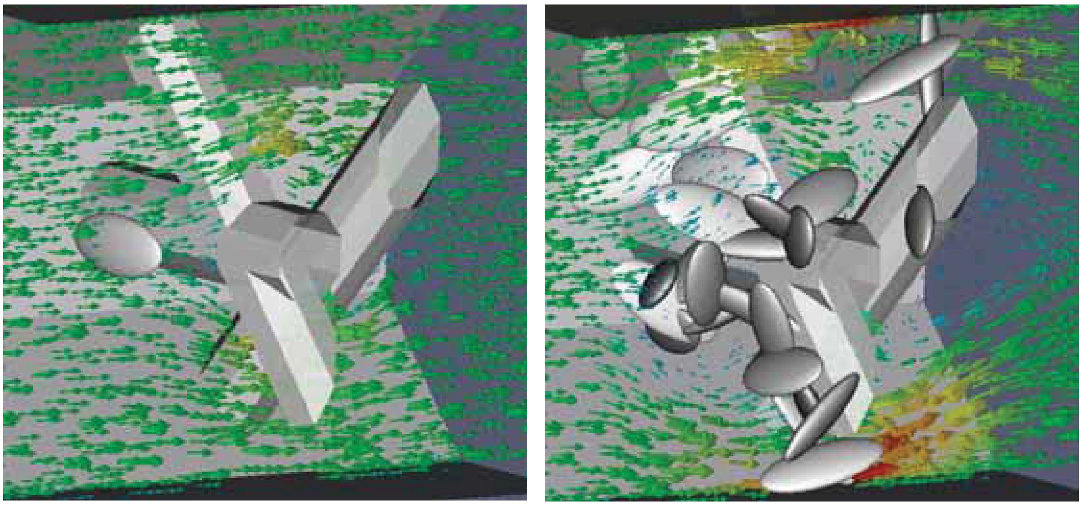


Figure 40: Flow field deviation by ellipsoid particles getting stuck in simplified fibre structure. Flow field before injection of multiple non-spherical particles (left). Deviated flow field after particle injection and impact on fibres (right).

To verify the results, an extensive fluid-particle force (pressure and shear force) evaluation within the *fibre vicinity module* has been conducted. Results have been compared to the corresponding values yielded by the *free flow module*, which is based on semi-empirical correlations (Hölzer & Sommerfeld, [24]), and to analytical formulations (Stokes drag). Here the outcome shall be discussed for the special case of equi-axed (=spherical) ellipsoids, as well as for ellipsoids of axe-ratio a:b:c=1.5:1:1.

4.2.6.2.1 Drag on Coarse Particles and Smooth Surface Correction

As symbolically depicted in Figure 36b, the *fibre vicinity drag module* gives a binary, coarse, grid spacing Δs dependent representation of the particle surface. Consequently the module yields higher fluid-particle forces than a comparable representation of smoothly surfaced objects. However, as will be shown in the oncoming chapters, the qualitative drag and lift force behaviour against Re_p is more than acceptable.

Figure 44 summarizes the drag force behaviour of simple Stokes-flow-spheres and *fibre-vicinity-module-* spheres. The ratio $S=\Delta s/D_{sph}$ is used as a parameter.

For supposedly, arbitrarily surfaced particles the CFD results can be expected to be more appropriate than any smooth surface representation. Still, correction functions have been introduced to compensate for surface roughness and numerical resolution effects on a user defined basis. Because of the good qualitative behaviour of the solution, the finding of a suitable correction function is comparatively simple. Possible parameters of dependence are the particle Reynolds number Re_p and the grid spacing ratio S. Exemplary cases within the parameter ranges $0.05 \le \mathrm{Re}_p \le 2.0$ and $0.05 \le S \le 0.5$ have been evaluated. Note that the particle model is, as of now, declared valid only for creeping flow conditions: $\mathrm{Re}_p<0.5$. The correction function ζ is defined via the c_d values of the analytical Stokes results $c_{d,\mathrm{Stokes}}$ and the model results $c_{d,\mathrm{model}}$:

$$\varsigma(S,\mathrm{Re}_p) = \frac{\log(c_{d,\mathrm{model}})}{\log(c_{d,Stokes})} \qquad (145)$$

An evaluation of the Re_p-influence shows, that for $\mathrm{Re}_p<0.5$ the correction ζ does hardly vary with Re_p, if compared to the local average ζ, as seen in Equ.146 and in Figure 41.

$$\frac{\partial \varsigma}{\partial \mathrm{Re}_p} \frac{1}{\bar{\varsigma}\Big|_{\mathrm{Re}_p=0}^{0.5}} \approx 0 \qquad (146)$$

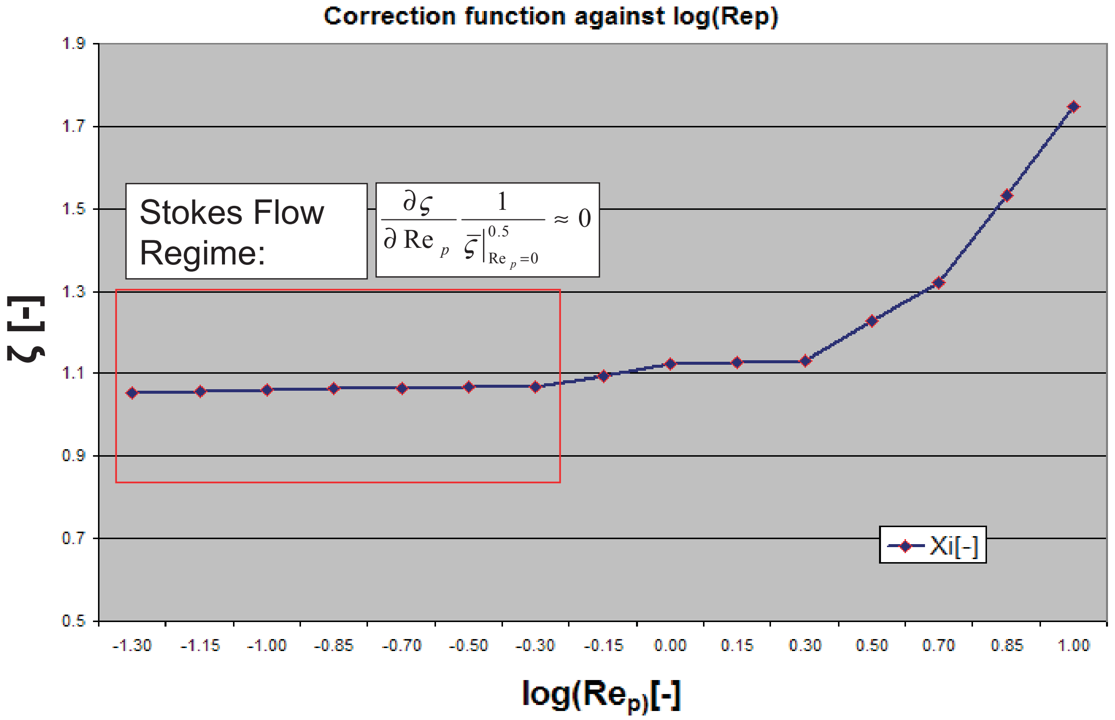


Figure 41: Plot of ζ against Re$_p$. For Re$_p$<0.5 there is no relevant result dependence on Re$_p$.

An evaluation of the S-dependence shows, that the formulation of a simple correction equation is possible, as seen in Figure 42.

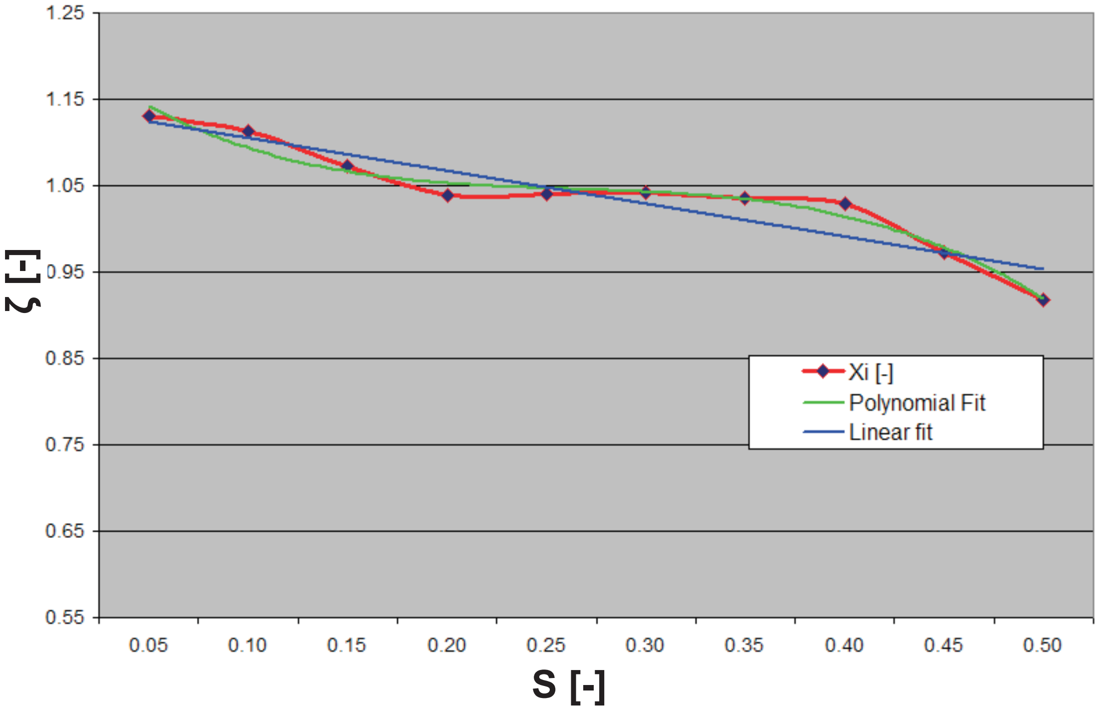


Figure 42: Plot of ζ against S. Results (red) are fitted linearly (blue) and with a 3rd order polynomial (green).

A linear fit to the ζ(S)-results gives:

$$\zeta_{\mathrm{Re}_p<0.5}(S') = -0.378S' + 1.1432 \qquad (147)$$

A 3rd order polynomial fit gives:

$$\zeta_{\mathrm{Re}_p<0.5}(S') = -8.0S'^3 + 6.480S'^2 - 1.7860S' + 1.2155 \qquad (148)$$

Here S' stands for S-S_0, with S_0=0.05. The smoothness correction ζ(S) is valid for Re_p<0.5 and 0.05≤S≤0.5. Within that region, the corrected c_d-values show an overall, relative, medium deviation from analytical results of ~8.1% (linear fit) and ~5.2% (polynomial fit). Similar results can be obtained for arbitrarily shaped ellipsoids. In the latter case the particle orientation is to be considered as well. Again the *fibre vicinity model* yields formidable outcomes (see Figure 46 and Figure 47).

A consideration of Figure 42, Equ.147 and Equ.148 yields the surprising result that ζ will have to be smaller for larger S-values than for fine grid spacing, even though the shape representation gets worse. The explanation for this can be seen in Figure 43 and is given by the fact that, with larger S, the closed fluid cell volume V_{block} decreases as compared to the analytic volume of the object V_a, until S~0.5. For S-values larger than 0.5, the “large” particle model forfeits its validity anyway.

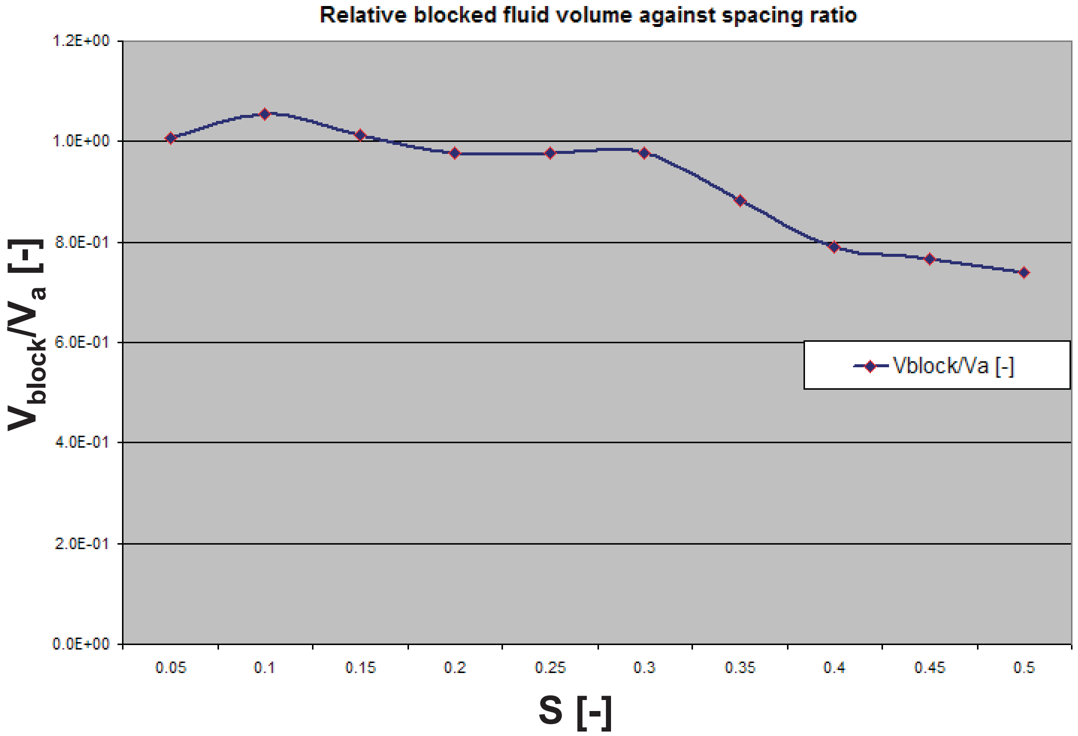


Figure 43: Plot of V_{block}/V_a against S . The closed fluid cell volume decreases with increased grid spacing. This effect outmatches increasing surface roughness and the overall drag force is reduced with increasing S.

Figure 44 shows analytical, un-corrected and smoothness-corrected model results in terms of c_d-values.

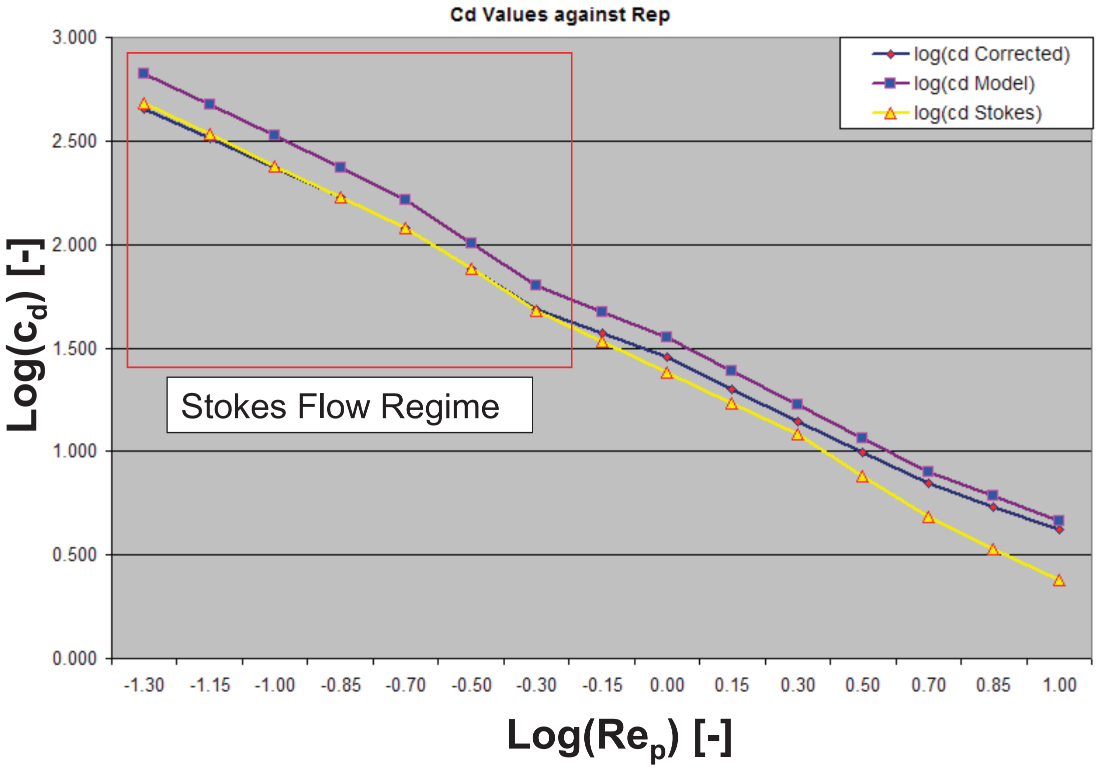


Figure 44: Plot of log(c_d) against log(Re_p). Averaged (over S range), original model results (purple) are fitted by polynomial smoothness correction using Equ.148 (blue) to analytical Stokes drag results (yellow). Model is valid within the Stokes drag regime, log(Re_p)≤-0.30.

4.2.6.2.2 Non-Spherical Drag and Lift Characteristic within the Fibre Vicinity Module

To provide a fully quantifiable basis for the essential *fibre vicinity drag module,* the drag and lift force characteristic for two way coupled, non-spherical particles has been worked out. In contrast to the spherical case, considered in the previous chapter, two main aspects of fluid-particle interaction will have to be accounted for in the non-spherical case:

- No general symmetry across planes through the object's mass centre is given. Therefore drag *and* lift forces will occur.

- Particle main axis orientation Φ to the main flow direction is no longer irrelevant.

Extensive simulation runs have been carried out to describe the full drag and lift behaviour of non-spherical particles within the *fibre vicinity module*. Ellipsoids with axis ratio a:b:c=1.5:1:1 and varying orientation have been systematically positioned within an otherwise uniform, unbounded flow field. It has been found that the modelled force results vary with varying particle positions within the fluid grid. To compensate, particle centre positions have been varied within the centre cell and results have been averaged.

Figure 45 shows some exemplary screenshots of the fluid being diverted around the ellipsoid at varying orientations.

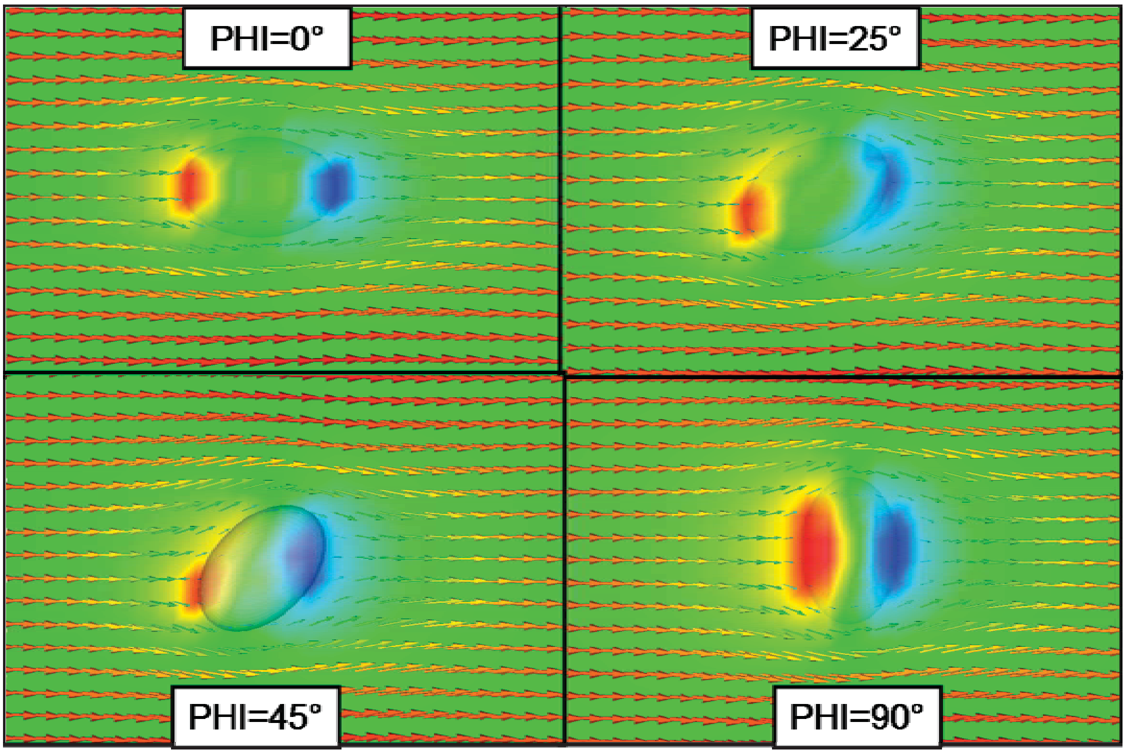


Figure 45: Ellipsoid with axis ratio a:b:c=1.5:1:1 with orientations Φ=0°, 25°, 45° and 90° to the relative flow velocity field with u_{rel}=0.4m/s. Re_p=0.3 (at Φ=0°). The background coloration represents the pressure field. Pressure build up at frontal stagnation points is clearly visible.

Resulting drag and lift forces have been used to calculate the corresponding drag and lift coefficients c_d and c_l respectively. The two characteristic interaction values are defined as [69]:

$$c_d = \frac{2 \cdot \left|\overline{F}_d\right|}{\rho_f \cdot u_{rel}{}^2 \cdot A_{Dsph}} \qquad (149)$$

$$c_l = \frac{2 \cdot \left|\overline{F}_l\right|}{\rho_f \cdot u_{rel}{}^2 \cdot A_{Dsph}} \qquad (150)$$

In Equ.149 and Equ.150, A_{Dsph} is the orientation independent, frontal area of a mass equivalent sphere. Since F_d and F_l are calculated via the orientation dependent frontal area $A_f(\Phi)$, the use of A_{Dsph} means that c_d and c_l contain not only hydrodynamic, but geometric orientation effects as well. In order to eliminate the geometric effects $A_f(\Phi)$ instead of A_{Dsph} can be used.

All in all the entire procedure of obtaining the drag/lift characteristic corresponds to the one used by Hölzer & Sommerfeld [69] who have derived the values from highly resolved LB simulations in terms of c_d and c_l according to Equ.149 and Equ.150. Comparability between the two methods is given, provided that the difference in particle surface roughness is taken into account. Figure 46 shows the results of the drag force/orientation analysis, as well as a simple sketch of the basic F_d and F_l situation. The results have been derived for Re_p=0.3 (at Φ=0°) and are directly compared to LB simulation results for smooth cuboids, cylinders and spheroids as well as to theoretical results for a smooth sphere and to *fibre vicinity model* results for a rough surfaced sphere.

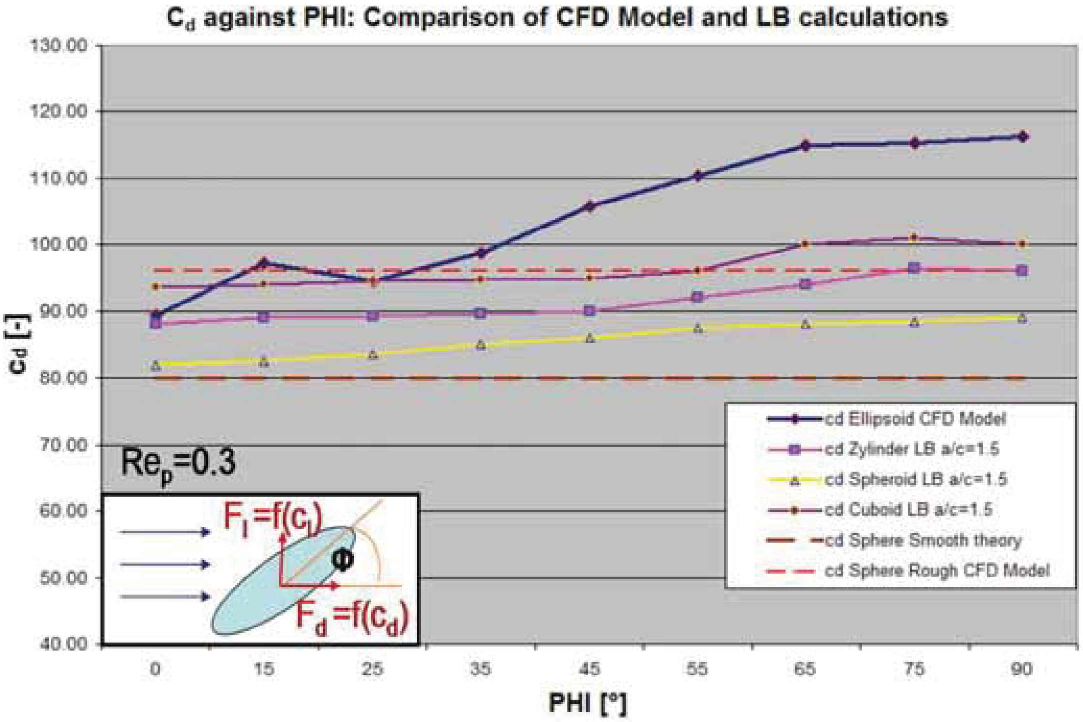


Figure 46: Drag force characteristic for Re_p=0.3 (at Φ=0°) for *fibre-vicinity-model*-coupled ellipsoid (blue). Results are directly compared to LB results [69] for smooth cuboid (purple), cylinder (pink) and spheroid (yellow) as well as to theoretical results for drag on smooth sphere (dashed, dark red) and to a *fibre-vicinity-model*-coupled, rough sphere (dashed, light red). All non-spherical objects feature: a:b:c=1.5:1:1.

The results in Figure 46 reveal the following facts:

- The *fibre-vicinity-model*-coupled ellipsoid results behave qualitatively very similar to comparable LB results.

- As expected and as commanded by plausibility c_d values for the coarse representation are generally larger than those of the smoothly surfaced objects.

- The coarse representation leads to larger differences between Φ=0° and Φ=90° as compared to the LB results.

- A comparison between a *fibre-vicinity-model*-coupled sphere and an ellipsoid yields plausible and consistent results throughout the entire orientation spectrum. At Φ=0° the sphere yields higher c_d values than the ellipsoid. With increasing Φ however, ellipsoid values quickly begin to outmatch those of the sphere.

The results shown in Figure 46 are based upon a first order surface normal gradient implementation of the shear force:

$$\overline{F}_\tau = \mu_f \frac{\overline{u}_f\left(\overline{x}_{Cellout,i}\right) - \overline{u}_f\left(\overline{x}_{CentreHP,i}\right)}{\left|\overline{x}_{Cellout,i} - \overline{x}_{CentreHP,i}\right|} \qquad (151)$$

Here $x_{CentreHP,i}$ is the fluid cell centre position of the cell containing help-point i and $x_{Cellout,i}$ is the fluid cell centre position of the neighbouring cell, which is first entered by moving outwards from $HP_{,i}$ along the local particle surface normal vector $n_{p,i}$.

In the course of development a second, less grid mesh dependent, more accurate shear force implementation has been devised, so that:

$$\overline{F}_\tau = \mu_f \frac{\overline{u}_f\left(\overline{x}_{5\%,i}\right) - \overline{u}_f\left(\overline{x}_{HP,i}\right)}{\left|\overline{x}_{5\%,i} - \overline{x}_{HP,i}\right|} \qquad (152)$$

Here $x_{HP,i}$ is the actual help-point position and $x_{5\%,i}$ is a position reached by moving outwards from $HP_{,i}$ along the local particle surface normal vector $n_{p,i}$ until $|x_{5\%,i} - x_{HP,i}| = D_{sph}*0.05$. The local fluid velocity vectors $u_f(x_{5\%,i})$ and $u_f(x_{HP,i})$ are obtained by interpolation between neighbouring fluid cell centre values. The obtained, overall c_d value characteristic is shown in Figure 47a. As compared to the previous results, the new shear force implementation yields:

- Generally smaller c_d values.

- Better agreement with comparable LB calculations.

- Smaller difference between Φ=0° and Φ=90°.

- For 60°< Φ<90°, a qualitative correspondence with LB results for a smooth cuboid, which actually features higher c_d values for Φ slightly below 90° than for Φ=90°.

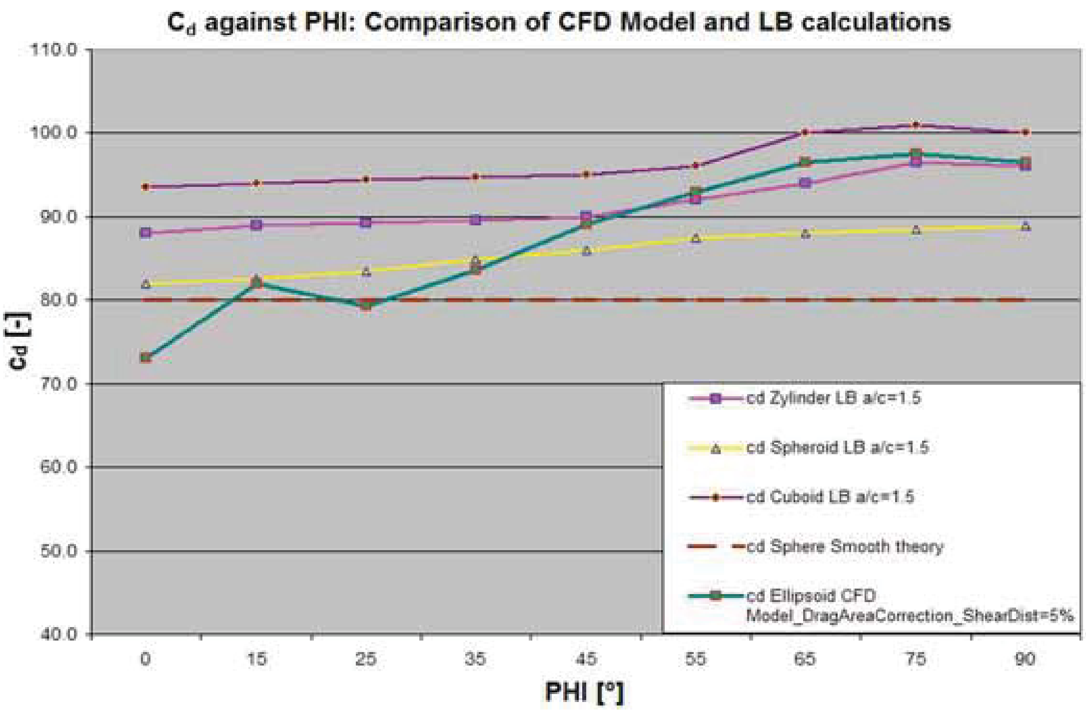


Figure 47a: Drag force characteristic for Re_p=0.3 (at Φ=0°) and *fibre-vicinity-model*-coupled ellipsoid using Equ.152 (green). Results are directly compared to LB calculations [69] for smooth cuboid (purple), cylinder (pink) and spheroid (yellow) as well as to theoretical results for drag on smooth sphere (dashed, dark red). All non-spherical objects feature: a:b:c=1.5:1:1.

Using A_f(Φ) instead of A_{Dsph} to calculate the specific resistence, Figure 47b can be produced. It shows that c_d, which contains only hydrodynamic information, actually decreases with increasing Φ. This is due to the positive correlation between Re_p and Φ for 0°< Φ<90° and for constant u_{rel}.

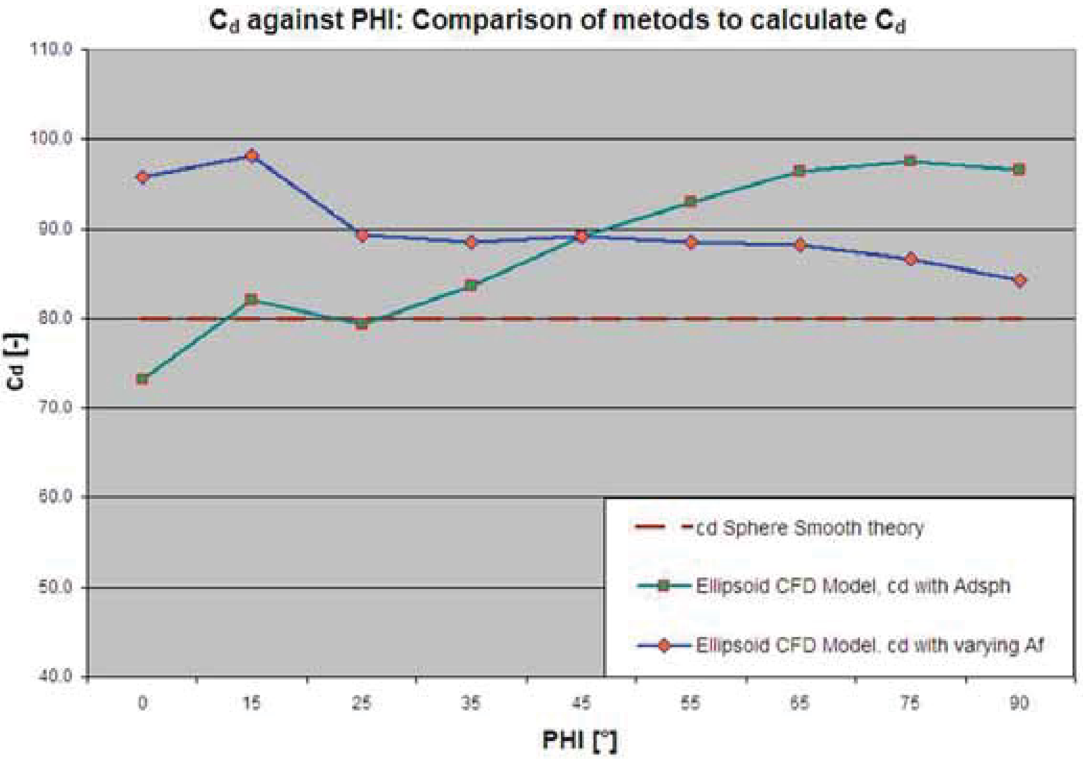


Figure 47b: Drag force characteristic for Re_p=0.3 (at $\Phi=0°$). Comparison between c_d results for smooth sphere (dashed red), *fibre-vicinity-model*-coupled ellipsoid using Equ.152 and A_{Dsph} to calculate c_d (green) and *fibre-vicinity-model*-coupled ellipsoid using Equ.152 and $A_f(\Phi)$ to calculate c_d (blue) and. All non-spherical objects feature: a:b:c=1.5:1:1.

Figure 48 shows the results of the lift force/orientation analysis. The results have been derived for Re_p=0.3 (at $\Phi=0°$) and are directly compared to LB simulations for smooth cuboids and spheroids. Note that most of the c_l values are negative. This means that F_l, which points upwards in the basic situation sketch bottom-right, for the given situation ($0° < \Phi < 90°$) actually points downwards. Simulations for $90° < \Phi < 180°$, which correspond to $0° > \Phi > -90°$, yield symmetrically equivalent results with $c_l \geq 0$.

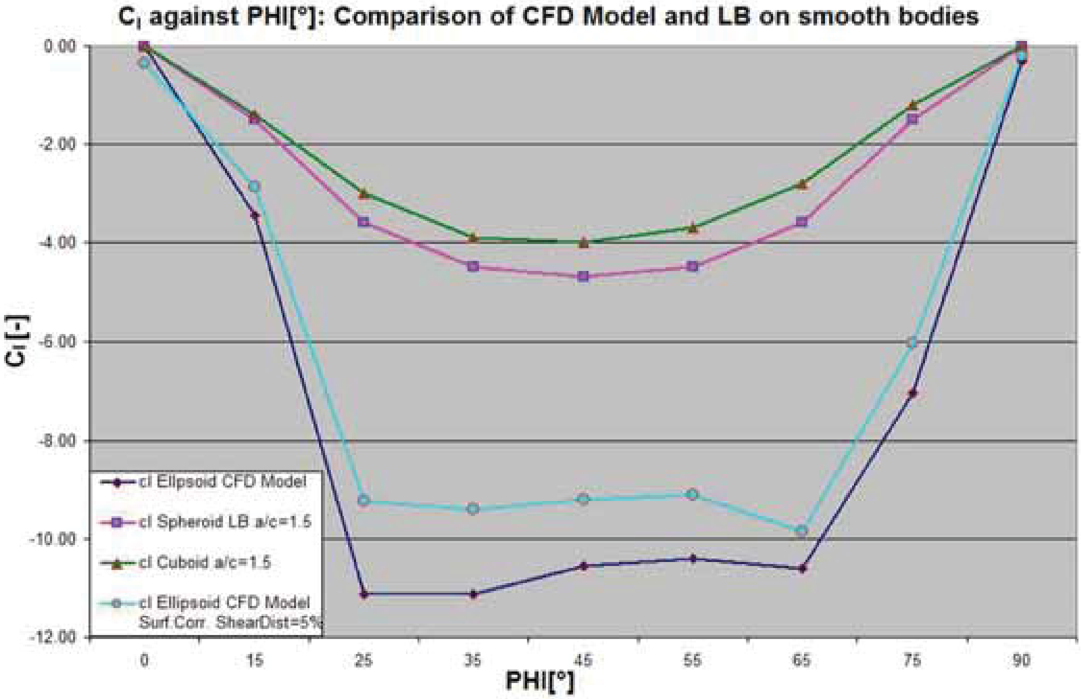


Figure 48: Lift force characteristic for Re_p=0.3 (at Φ=0°) for *fibre-vicinity-model*-coupled ellipsoid using Equ.151 (blue) and using Equ.152. Results are directly compared to LB calculations [69] for smooth cuboid (green) and spheroid (pink). All non-spherical objects feature: a:b:c=1.5:1:1.

The results in Figure 48 reveal the following facts:

- Both *fibre-vicinity-model*-coupled ellipsoid results behave qualitatively similar than comparable LB results.
- The absolute c_l values for the coarse representation are generally larger than those of the smoothly surfaced objects. Where the shear force implementation according to Equ.152 yields better correspondence.
- In contrast to the “LB objects”, both coarse ellipsoid characteristics show steep slopes at small deviations from either Φ=0° or Φ=90°, whereas c_l does not change significantly for 25°< Φ<65°.

Proposal for descriptive formulation of drag and lift force characteristics

Even though drag and lift forces stem from fluid pressure and shear effects, they are usually treated as separate phenomena. It is hereby proposed to introduce one, single compact expression for a fluid-particle interaction force F_{fp} which shall be proportional to the hydrodynamic interaction coefficient c_{fp}.

The referred situation is sketched out in Figure 49. Hereby a semi-axisymmetric, non-spherical particle, with a:b:c=x:1:1, whose main axes orientation deviates by an angle Φ around the relative particle-fluid velocity vector u_{rel} is considered.

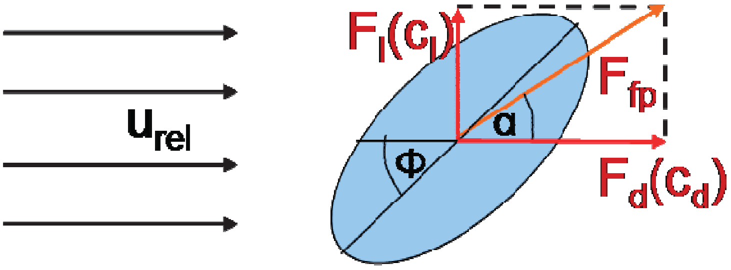


Figure 49: Non-spherical particle immersed in relative fluid flow, with acting drag and lift forces.

As described in the previous chapters, a drag force F_d and a lift force F_l will start to act on the particle. They depend on the characteristic drag force coefficient c_d and on the lift force coefficient c_l respectively. Thus F_{fp} is given by:

$$\overline{F}_{fp}\left(c_{fp}\right) = \overline{F}_d\left(c_d\right) + \overline{F}_l\left(c_l\right) \qquad (153)$$

Whereby c_{fp} is defined as:

$$c_{pf} = \frac{2 \cdot \left|\overline{F}_{pf}\right|}{\rho_f \cdot u_{rel}{}^2 \cdot A_{Dsph}} \qquad (154)$$

Which is equivalent to the definition of c_d and c_l (Equ.149 and Equ.150). Because of the vectorial nature of Equ.153, the c_{fp} relation to c_d and c_l is:

$$c_{pf}(\phi)^2 = c_d(\phi)^2 + c_l(\phi)^2 \qquad (155)$$

Considering Figure 49 and Equ.155, the orientation angle α of F_{fp} to u_{rel} can be calculated as:

$$\alpha(\phi) = \arctan\left(\frac{c_l(\phi)}{c_d(\phi)}\right) \tag{156}$$

Equ.155 and Equ.156 already indicate that c_d and c_l are functions of Φ, which can be directly derived from Figure 46 to Figure 48.

Since drag and lift force characteristics usually consider a 2D force-effect situation, it is the opinion of the author that an *"ansatz"*, based upon complex number methodology is suitable to elegantly describe vectorial, hydrodynamic force effects. Thus F_{fp} can be expressed in complex number notation as:

$$\overline{F}_{fp} = \left|\overline{F}_d\right| + i\left|\overline{F}_l\right| \tag{157}$$

Or as:

$$\overline{F}_{fp}(\phi) = \left|\overline{F}_{fp}\right|(\phi) \cdot e^{i\cdot\alpha(\phi)} = \sqrt{c_d(\phi)^2 + c_l(\phi)^2} \cdot K' \cdot e^{i\cdot\arctan\left(\frac{c_l(\phi)}{c_d(\phi)}\right)} \tag{158}$$

Where K' is the orientation independent constant:

$$K' = \frac{1}{2} \cdot \rho_f \cdot A_{Dsph} \cdot u_f^{\,2} \tag{159}$$

In order to mathematically characterize F_{fp}, based upon numerical results from i.e. Figure 47 and Figure 48, the $c_d(\Phi)$ and $c_l(\Phi)$ functions need to be approximated.

Adopting the example of the smooth LB spheroid, a valid characterization could look like this:

Approximation of the Spheroid’s $c_d(\Phi)$ function as seen in Figure 50:

$$c_d(\phi) = 85.5 - 3.5 \cdot \cos(2 \cdot \phi) \tag{160}$$

Approximation of the Spheroid’s $c_l(\Phi)$ function as seen in Figure 51:

$$c_l(\phi) = 4.7 \cdot \sin(2 \cdot \phi) \tag{161}$$

As a consequence of Equ.160 and Equ.161, the Spheroid’s, characteristic $|F_{fp}|(\Phi)$ function is given as:

$$\left|\overline{F}_{fp}\right|(\phi) = \sqrt{c_d(\phi)^2 + c_l(\phi)^2} = \sqrt{-(k_1 \cdot \cos(2 \cdot \phi) - k_2)^2 + k_3} \tag{162}$$

With k_1=3.1368, k_2=95.3998 and k_3=16433.50.

The Spheroid’s, characteristic $\alpha(\Phi)$ function is given as:

$$\alpha(\phi) = \arctan\left(\frac{c_l}{c_d}\right) = \arctan\left(\frac{\sin(2 \cdot \phi)}{k_{1'} - k_{2'} \cdot \cos(2 \cdot \phi)}\right) \tag{163}$$

with $k_{1'}$=18.1915 and $k_{2'}$=0.7447.

The procedure to characterize F_{fp} for the coarse, fibre-vicinity-coupled ellipsoid works accordingly, even though the result is more complex, considering the $c_d(\Phi)$ and $c_l(\Phi)$ approximation functions as seen in Figure 50 and Figure 51 respectively:

$$c_l(\phi) = 1.7 \cdot \left[5 \cdot \sin(-2 \cdot \phi) + \sin\left(-\frac{19}{2} \cdot \phi\right) - \sin(2 \cdot \phi)\right] \tag{164}$$

$$c_d(\phi) = 73.14 + 23.4 \cdot \sin(\phi) \tag{165}$$

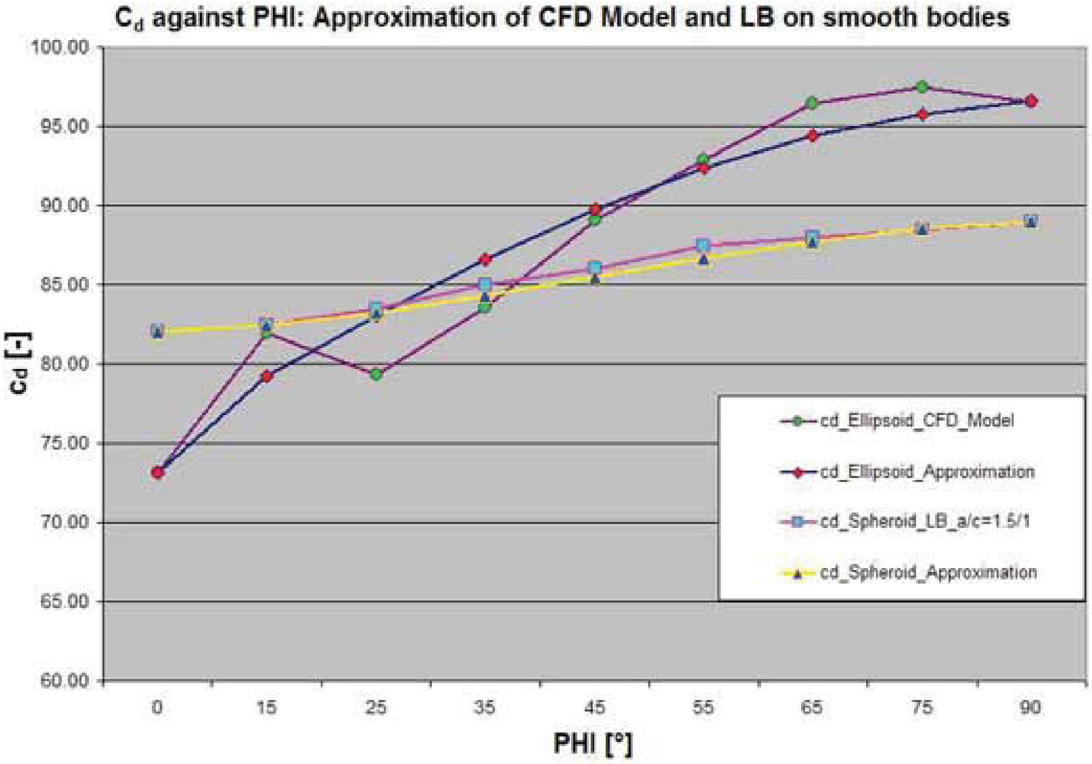


Figure 50: c_d values of ellipsoid according to CFD model (purple) and Spheroid according to smooth body LB simulations (pink) against Φ. Both objects feature a:b:c=1.5:1:1. Results are compared to approximation functions (blue and yellow, respectively).

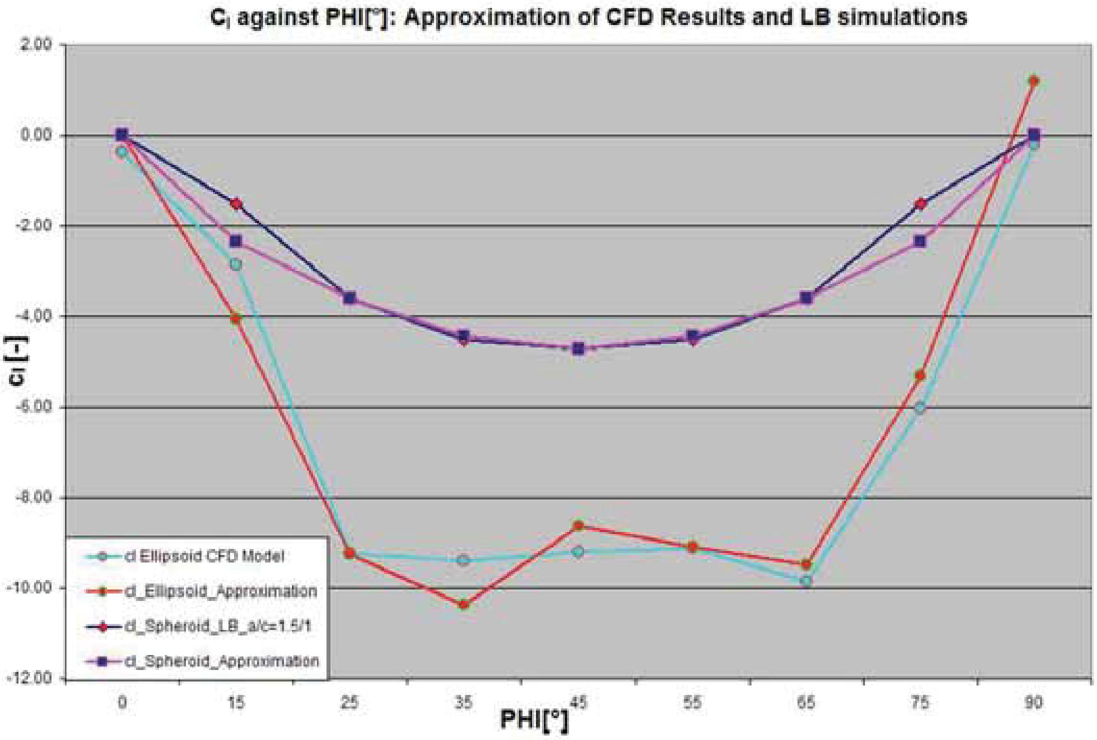


Figure 51: c_l values of ellipsoid according to CFD model (turquoise) and Spheroid according to smooth body LB simulations (blue) against Φ. Both objects feature a:b:c=1.5:1:1. Results are compared to approximation functions (red and pink, respectively).

4.2.6.2.3 Verification of the (Non-)Spherical Fibre Vicinity Drag Model: Terminal Settling Velocity

The classic approach to quantitatively verifying fluid-particle interaction models is to take a look at the single particle settling behavior in an otherwise quiescent fluid, [27]. Hereby the flow field formation, the development of the settling velocity u_p and the terminal, settling velocity u_s of exemplary, spherical and non-spherical particles has been qualitatively and quantitatively studied. Results have been compared to theoretical predictions.

Spherical Settling:

Initially the case of a simple, two-way coupled sphere of diameter D_{sph}, settling due to gravity in a cylindric, fluid filled structure of diameter D_{cyl}, where D_{sph}<<D_{cyl}, has been investigated.

Qualitative plausibility checks, concerning the flow field formation due to the settling particle, were conducted. An exemplary result can be seen in Figure 52. It shows the developing *counter swirl* velocity vector field as well as the expected pressure build-up at the forward stagnation point. All in all the qualitative flow field formation corresponds perfectly with real conditions and with results reported by Megahed [27].

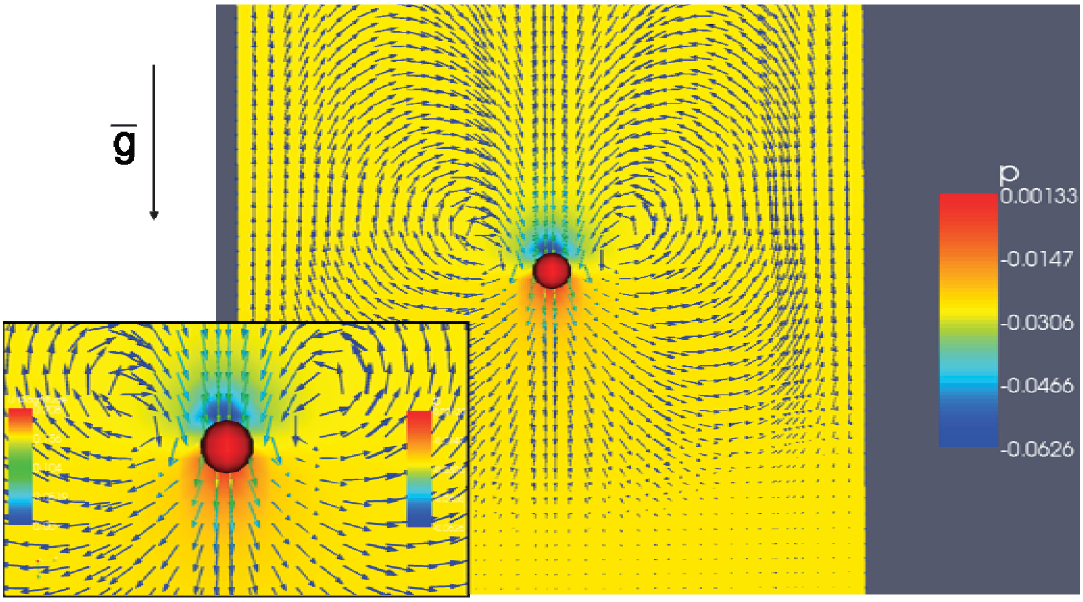


Figure 52: Fibre-vicinity-model-coupled particle, settling under the influence of gravity. Formation of a *counter swirl* fluid velocity vector field and the corresponding pressure field (background coloration) due to the settling particle.

Quantitative investigations, focussing on the development of u_p, have been conducted as well. The results have been compared to theoretical formulations for the terminal settling velocity and for settling velocity development. For spherical particles, which settle under *Stokes* conditions, the terminal settling velocity u_s can be calculated as:

$$u_s = \frac{D_{sph}{}^2 \cdot g \cdot (\rho_p - \rho_f)}{18 \cdot \mu_f} \tag{166}$$

Therefore, and by inserting for the particle relaxation time τ_p, the analytical development of the settling velocity can be written like this:

$$u_p(t) = \tau_p \cdot g \cdot \left(1 - \frac{\rho_f}{\rho_p}\right) \cdot \left(1 - e^{-\frac{t}{\tau_p}}\right) = u_s \cdot \left(1 - e^{-\frac{t}{\tau_p}}\right) \tag{167}$$

In its formulation for numerical calculation, using discrete particle time steps Δt_p this expression reads:

$$U_{p,i+1} = u_{p,i} + \left[g \cdot \left(1 - \frac{\rho_f}{\rho_p}\right) - 18 \cdot \frac{\mu_f \cdot u_{p,i}}{\rho_p \cdot D_{sph}^2}\right] \cdot \Delta t_p = U_{p,i} + \frac{1}{\tau_p}\left[u_s - u_{p,i}\right] \cdot \Delta t_p \tag{168}$$

Here i is the index for the current, numerical time step.

Figure 53 summarizes the results. It compares relative sinking velocities for rough surfaced spheres, which result from the developed CFD model, to:

- Analytically calculated values for the terminal settling velocity of mass equivalent, smooth spheres ($c_d=24/Re_p$).

- Rough spheres ($c_d=27/Re_p$), where the c_d value relation has been taken out of the previously simulated drag characteristic (see chapter 4.2.6.2.1).

- To the corresponding velocity developments calculated according to Equ.167 and Equ.168.

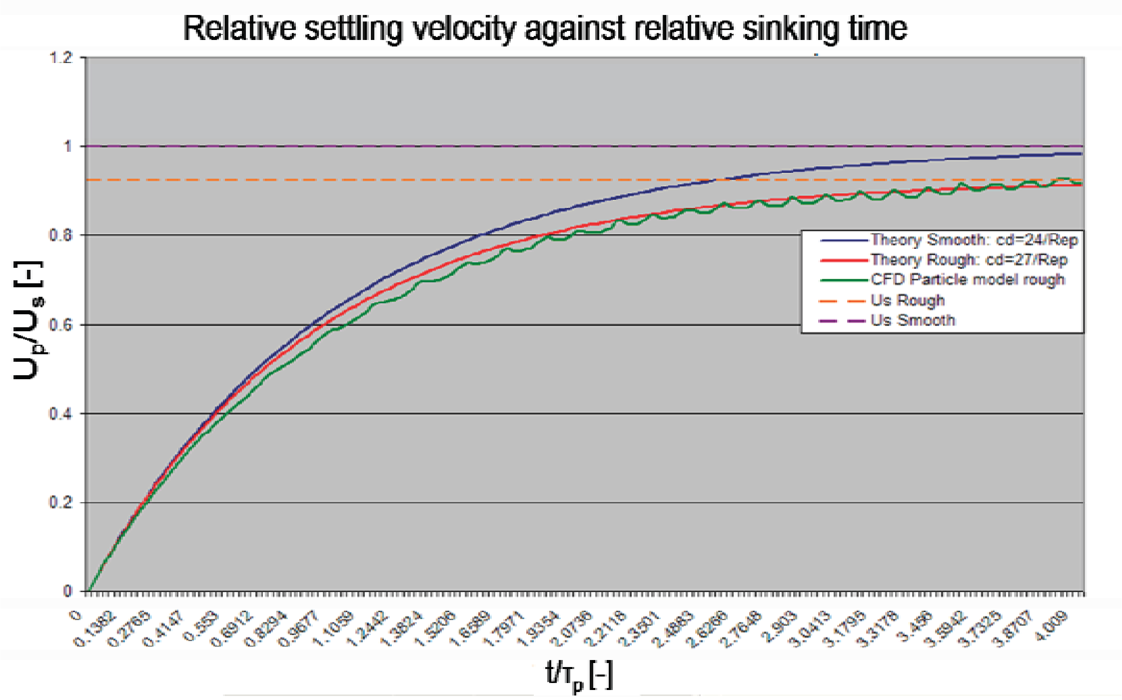


Figure 53: Relative settling velocity against relative settling time for spheres, settling in an otherwise quiescent, unbounded flow domain. Comparison of model results (green) with theoretical velocity development of smooth sphere (blue), rough sphere (red) as well as the terminal settling velocity for smooth (purple) and rough (orange) spheres.

The results presented in Figure 53 show very good agreements with theoretical predictions. As expected, the coarsely represented particles reach a slightly smaller, terminal settling velocity than theoretical, smooth spheres. The settling characteristic however, corresponds almost exactly with theoretical predictions for spheres that feature a previously (Figure 44) derived c_d correlation. One striking difference between the results is that the CFD modelled sphere's velocity development profile is not quite as smooth. The reason for that lies within slight, numerical fluctuations in the surrounding pressure field and within 3D flow and movement effects. In other words: the particle slowly starts to rotate due to slightly unsymmetrical drag force distributions. The latter effect is based upon the coarse surface representation. It is, however rather considered an asset than a problem, since real-life dirt particles will show very similar, rotational sinking behavior.

Non-Spherical Settling

The case of a theoretical and even more so a real-life, non-spherical dirt particle, settling under gravity is more complex. Non-spherical objects tend to turn towards an orientation angle of maximum drag resistance. The reason for this is based upon stagnation pressure effects. Whenever the object starts rotating towards a more streamlined position, the forward stagnation point shifts away from the particle mass centre and creates a new torque effect into the opposite direction of the rotation (see Figure 55), [69].

Therefore the first analysis of a settling ellipsoid has been conducted under the idealized assumption of rotation-less sinking in its most stable position (Φ=90°). An analytical reconstruction of the numerical velocity development looks like Equ.169 which can be brought to accordance with Equ.168 by inserting for a generalized particle relaxation time as seen in Equ.215:

$$U_{p,i+1} = U_{p,i} + \left[g \cdot \left(1 - \frac{\rho_f}{\rho_p} \right) - \frac{3}{4} \cdot \frac{\rho_f \cdot U_{p,i}{}^2}{\rho_p \cdot D_{sph}} \cdot c_d \right] \cdot \Delta t_p = U_{p,i} + \frac{1}{\tau_p} \left[u_s - u_{p,i} \right] \cdot \Delta t_p \qquad (169)$$

Hereby the roughness-corrected (additional 12.5% drag force) c_d relation from Equ.5, [24] has been inserted and compared to a fibre-vicinity-coupled ellipsoid as well as to a smooth, falling, mass equivalent sphere. The results of the velocity development have been put into relation to the terminal, spherical settling velocity and are shown in Figure 54.

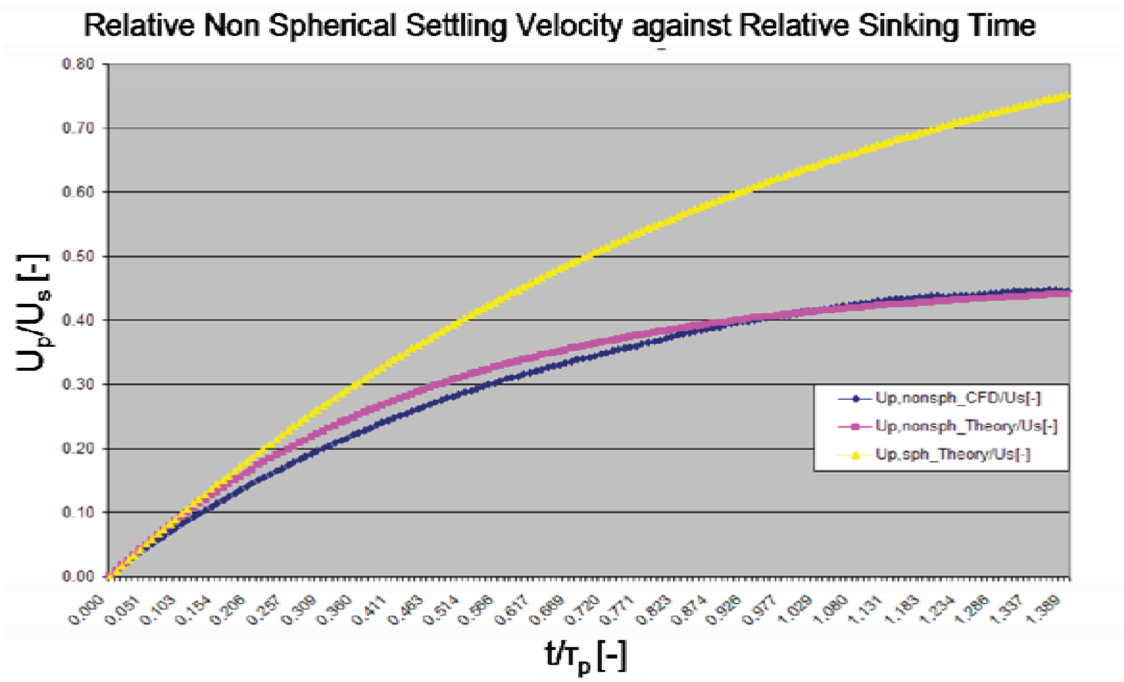


Figure 54: Relative settling velocity against relative settling time for ellipsoids with a:b:c=1,5:1:1, and a mass equivalent sphere settling in an otherwise quiescent, unbounded flow domain. The Archimedes number is Ar~8*10^5. Comparison of model results (blue) with theoretical velocity development of smooth sphere (yellow), and a rough ellipsoid (pink) calculated according to Equ.169 and Equ.5 with additional 12.5% drag force to compensate for rough surface.

Again the model results match comparable semi-analytical calculations. As expected, the terminal velocity difference between a smooth, settling sphere and a rough ellipsoid is even more significant, than in the purely spherical case. Since rotation has been suppressed in this example, the slightly unsteady behavior of the model particle, which has been observed before is almost gone here. With an Archimedes number of Ar~8*10^5 the ratio of the terminal particle settling velocity of smooth, longish non-spheres and the terminal particle settling velocity of a mass equivalent sphere is supposed to be ~0.53 [92]. Here the ratio is ~0.46, which is perfectly plausible since in our case rough, longish ellipsoids are compared to smooth spheres.

Secondly the model's qualitative ability to realistically simulate *real-life*, non-spherical dirt particle settling effects has been investigated. To do that the case of an ellipsoid, with a:b:c=1.5:1:1, initially oriented with Φ=45° towards the gravity vector has been chosen. The expected settling behavior is symbolically sketched out in Figure 55. At the beginning the particle will experience relatively small drag forces and, due to its orientation, relatively large lift forces, perpendicular to the direction of gravity. In addition to that, the off-centre formation of the forward stagnation point and the corresponding build-up of a relatively high (compared to the averaged particle surface pressure p_{med}) stagnation pressure p_{stag} will exert torque on the particle. As the ellipsoid accelerates it will rotate towards the stable $\Phi=90°$ position. Consequentially the F_d/F_l ratio will increase and the p_{stag}/p_{med} ratio will decrease. The higher the angular particle relaxation time τ_ω (ratio between angular inertia and fluid viscous forces), the more pronounced the particle's rotation past the stable position will be. A symmetrical, but dampened counter movement will follow.

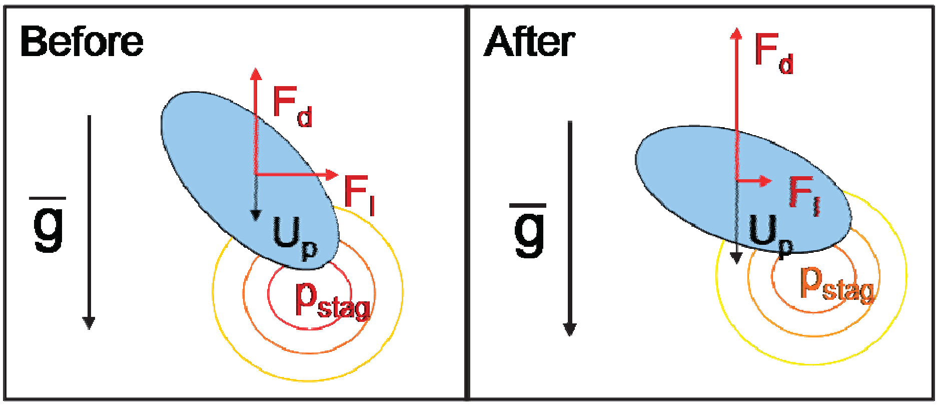


Figure 55: Sketch of the basic hydrodynamic situation of an ellipsoid settling under the effect of gravity.

The newly developed drag/lift model has been applied within a CFD simulation run for the situation, described above. A visualization (Figure 56) of the results shows staggering correspondence with the real-life phenomenon. Fluid field formation, pressure build-up, force and torque effects as well as the particle translational and rotational characteristic, behave just as anticipated. Up until now there is no other CFD simulation tool, known to the author that is just as capable of grapping the full complexity of the non-spherical, settling phenomenon, as the hereby presented simulator is.

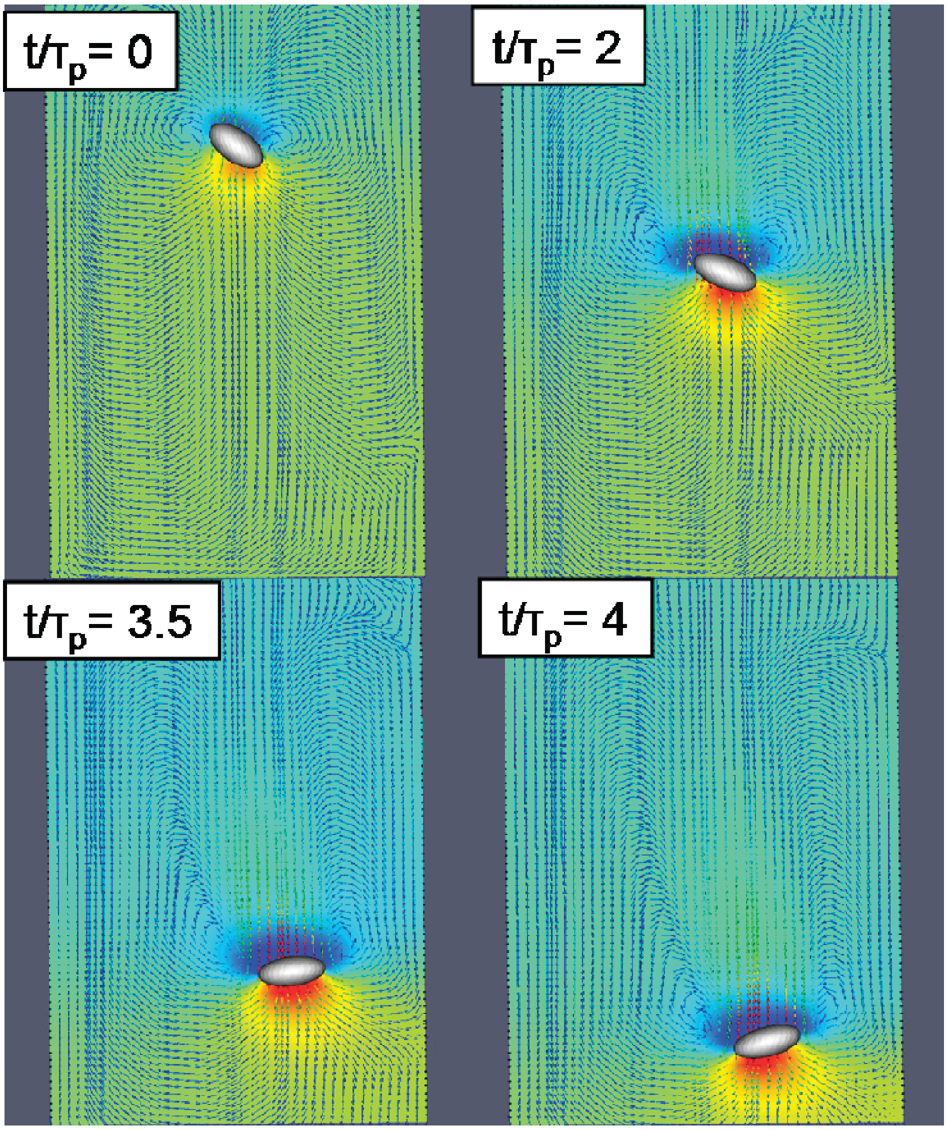


Figure 56: Ellipsoid (ρ_p=2000kg/m³) with initial orientation Φ=45° settling under the influence of gravity in otherwise quiescent fluid domain (ρ_f=1000kg/m³, ν_f=1*10^{-6}m²/s). Chronological screenshots of simulation results, using the newly developed large, Lagrangian dirt particle model.

4.2.7 Non-Spherical Particle Interaction Effects: Event Forces

Forces with influence on particle motion that occur due to individual impact events are also essential aspects of Lagrangian modelling. They are hereby called *event forces* and represent particle-wall, particle–particle and particle-fibre interaction effects.

As soon as a particle enters into the *fibre vicinity domain*, not only does the drag module change, but also the relevance of event forces. Impact events are not to be neglected anymore.

All N relevant impact forces on particle motion occur within infinitesimal time scales that are smaller than numerical fluid time steps Δt_f or particle time steps Δt_p. Via force effect modelling (see 4.1.1) their amount is averaged over the entire particle time step, in order to achieve equivalent outcome without infinitely high *"instant forces".*

4.2.7.1 Non-Spherical Event Forces: Particle-Wall Interaction

Boundary patches that are neither defined as "Inlet", "Outlet" nor "Fibre", are defined as "Walls". In case of the treatment of real fibre geometries, those patches constitute the open borders to neighbouring fibre-fluid regions. In general however, the wall boundary condition can be set to any solid object immersed in the fluid. The fluid boundary conditions for wall patches with normal n_w, are set to "*slip*". Let u and v denote velocity components parallel to the wall boundary patch and w denote velocity components perpendicular to the wall, then a *"slip"* boundary condition means:

$$\begin{aligned} &w = 0 \\ &\frac{\partial u}{\partial \overline{n}_w} = \frac{\partial v}{\partial \overline{n}_w} = 0 \end{aligned} \tag{170}$$

Particles of mass m_p that hit a wall boundary patch with translational velocity u_p and angular velocity ω_p are reflected with user-definable elasticity E_w. The reflection is performed by the use of a wall event force F_{wall}. Wall impact forces have already been introduced in chapter 4.1.3 where they handle spherical particle-wall interaction. Now the concept has to be extended in order to consider rotational effects, particle orientation, the moment of inertia I_p and the exact (or at least approximated) point of impact at the particle surface. In Figure 57 the basic difference in spherical and non-spherical particle-wall impact is sketched out.

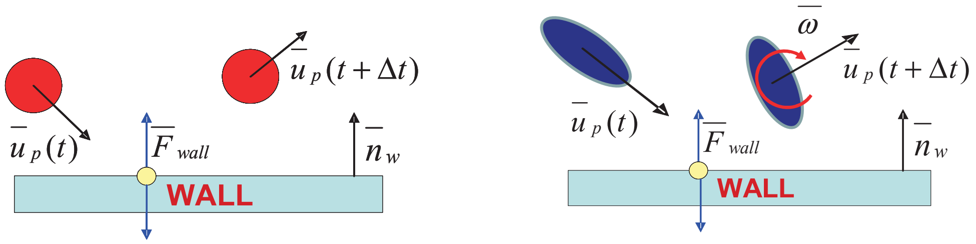


Figure 57: Sketch of spherical (left) and non-spherical (right) particle-wall impact. Non-spheres experience rotational as well as translational acceleration at impact.

To sense the impact location, the help points, with distance vector r_i to the particle centre are used and F_{wall} is modelled as:

$$\overline{F}_{wall} = -\frac{1+E_w}{\Delta t_p}\,\frac{(\overline{u}_p+\overline{\omega}_p\times\overline{r}_i)\cdot\overline{n}_w}{\frac{1}{m_p}+\frac{1}{I_p}[(\overline{r}_i\times\overline{n}_w)\times\overline{r}_i]\cdot\overline{n}_w}\cdot\overline{n}_w \qquad (171)$$

Note that F_{wall} always acts perpendicular to the wall boundary patch, whereas fibre forces F_{fibre} (see chapter 4.2.7.2) also have components parallel to the boundary patch. Impact elasticity E_w has decisive influence on the translational and rotational impact behaviour of the particle. In Figure 58 the situation before and after impact is sketched out. The difference between completely elastic (E_w=1) and completely plastic (E_w=0) impact scenario is shown.

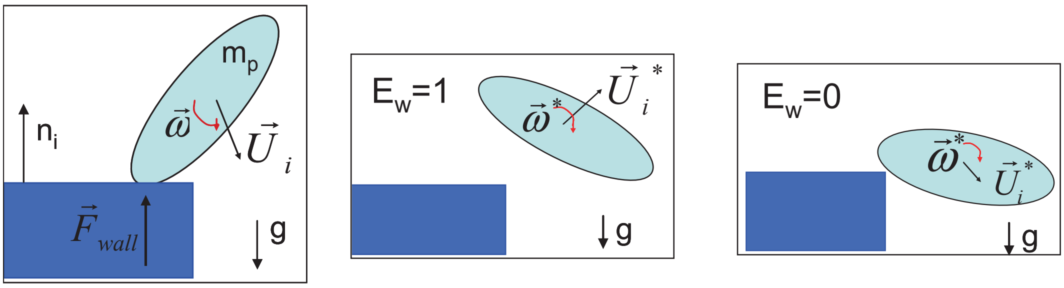


Figure 58: Qualitative sketch of the influence of non-spherical impact elasticity on particle-wall impact situation.

Screenshots of a qualitative benchmark case are shown in Figure 59. Here a simplified, fibre-like object in the particle flow path has been prescribed the *wall* *boundary condition* and thus serves as obstacle. An oval shaped ellipsoid hits the obstacle and is affected by the wall impact force. Consequently its translation and rotation is altered.

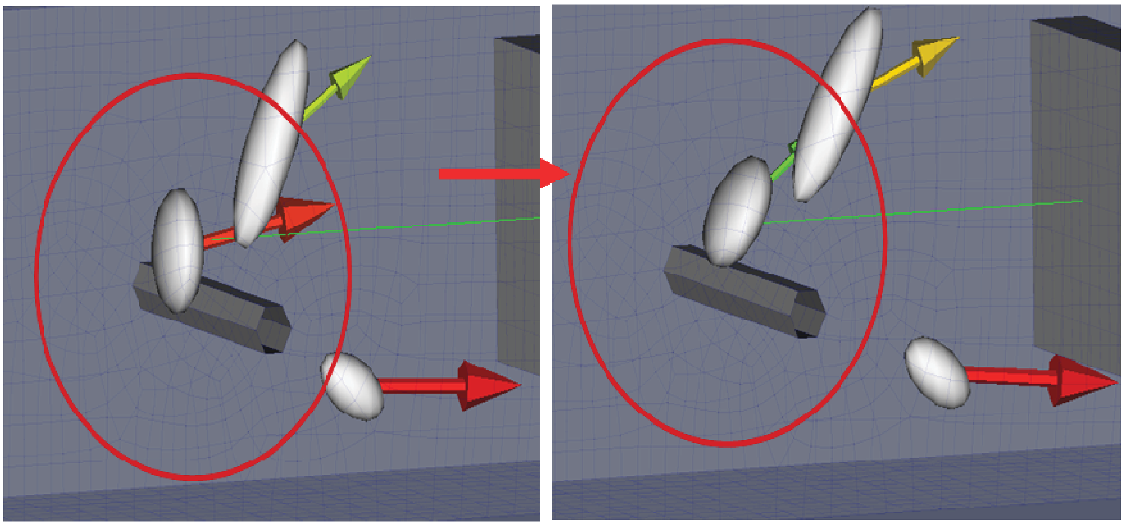


Figure 59: Screen shot of benchmark case. Ellipsoid particle just before (left) and after (right) hitting a wall boundary patch. Particle experiences positive, rotational and negative, translational acceleration.

4.2.7.2 Particle-Fibre Interaction and Particle Deposition Model

The modelling of particle interaction with filter fibres and deposition mechanisms is essential. Due to the non-stochastic nature of the particle solver, no simplifying deposition model as in [12] or [13] is applied. Each particle is treated as dynamically interacting individual, according to its momentum. Hydrodynamic or event forces as well as fibre forces, which restrain its movement are taken into account. Thus every increase of hydrodynamic or event forces can lead to a *particle blow-off*.

The amount of restraining fibre forces F_{fibre} on the particle naturally has a decisive impact on particle-fibre affinity and eventually filter fibre efficiency. Particle-fibre affinity is a quantity which can hardly be described analytically. As a consequence, one has to rely on experiments in order to obtain a better understanding of deposition mechanisms. Usually those experimental results come in the form of filter efficiency curves. These curves have to be studied at the particle diameter ranges, where intrinsic particle-fibre adhesion is *the* dominating filtration mechanism. Supposedly a fibre material which shows particle adhesion, can restrain any infinitesimally small particle (D_{sph}<<) hitting the fibre.

It is plausible to assume that there is a maximum, possible amount of fibre force on the particle and that it is directly proportional to the contact area, which is again proportional to D_{sph}^2. Fibre forces eliminate particle momentum, which is proportional to D_{sph}^3 and act against hydrodynamic drag which under *Stokes* flow conditions is roughly proportional to D_{sph}. As a consequence, particles will be harder to stop, as D_{sph} gets larger. Above a certain *“sticking diameter”* D_{stick}, particles will deposit according to a probability density distribution function (PDDF) that can be derived from experiments. The task within solver design is to integrate any, user definable PDDF of the form seen in Equ.178 into the concept of deterministic particle-fibre interaction-force-deposition.

To tackle this problem a special particle-fibre interaction module has been created. It distinguishes between three modes of interaction: the impact, the gliding and the full stop or deposition phase.

Impact phase

When a particle–fibre impact occurs, $F_{fibre,impact}$ acts just like F_{wall} for completely plastic (E_w=0) impact and can be written as:

$$\overline{F}_{fibre,impact} = -\frac{\left(\overline{u}_p + \overline{\omega}_p \times \overline{r}_i\right)\cdot \overline{n}_w}{\Delta t_p \cdot \left(\frac{1}{m_p} + \frac{1}{I_p}\left[\left(\overline{r}_i \times \overline{n}_w\right)\times \overline{r}_i\right]\cdot \overline{n}_w\right)} \cdot \overline{n}_w \qquad (172)$$

$F_{fibre,impact}$ is designed to momentarily remove the particle help point velocity component perpendicular to the wall $u_{p,i}^{\perp}$. For the particle mass centre this means that the velocity component perpendicular to the wall $u_p^{\perp}$ is reduced and that a particle moment of impact $M_{fibre,impact}$ causes the particle to rotate around the impact point (see Figure 60).

Gliding phase

As long as the particle detects fibre contact and still retains a velocity component parallel to the wall $|u_{p,||}|>0$, it glides along the fibre surface. The decelerating fibre counter force $F_{fiber,gliding}$ now acts in opposite direction of $u_{p,||}$ and is proportional to D_{sph}^2, the amount $|u_{p,||}|$ and a user definable material, fibre force constant D_{ff}[kg/m²s], so that:

$$\overline{F}_{fibre,gliding} = -D_{ff} \cdot D_{sph}^{2} \cdot \overline{u}_{p,\|} \tag{173}$$

The pre- and post impact situation is sketched out in Fig.60.

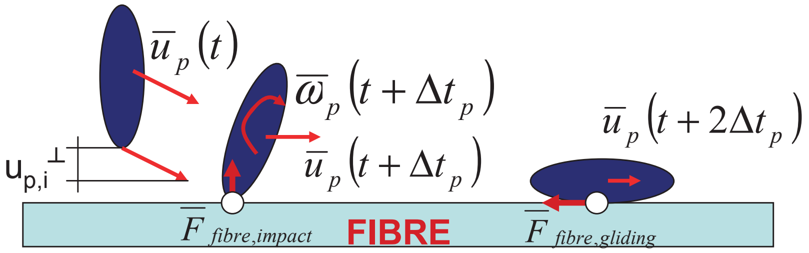


Figure 60: Sketch of a non-spherical particle-fibre impact/interaction situation. Ellipsoid just before (left) and after (middle, right) fibre impact. At impact particle is affected by $F_{fibre,impact}$, rotation around impact point, towards fibre sets in (middle). Particle then decelerates (glides) along surface, being restrained by $F_{fibre,gliding}$.

Full stop/deposition phase

Particle acceleration into the opposite direction, due to $F_{fiber,gliding}$ would be unphysical. Therefore the reversal of velocity direction, or the deceleration below a user defined, very low, *stop velocity* u_{stop} make a valid criterion for complete particle deposition. The reversal condition states that the ratio of numerical particle velocity-change $|\Delta u_{p,||}|$ due to fibre force during Δt_p, and $|u_{p,||}|$ equals 1, which means:

$$\frac{\left|\Delta \overline{u}_{p,\|}\right|}{\left|\overline{u}_{p,\|}\right|} \overset{!}{=} \frac{6 D_{ff} \Delta t_p}{D_{sph} \pi \rho_p} \overset{!}{=} 1 \tag{174}$$

A *stop number* N_{stop} is introduced, which contains both, the reversal condition and a *stop velocity* condition so that:

$$N_{stop} = \frac{6 D_{ff} \Delta t_p}{D_{sph} \pi \rho_p} + \frac{u_{stop}}{\left|\bar{u}_{p,||}\right|} \tag{175}$$

Here the *stop velocity* condition $u_{stop}/|u_{p,||}|$ merely serves as break up condition for the convergence of particle velocity to *zero*. Therefore u_{stop} should be set $<<u_f$. The final deposition condition is:

$$N_{stop} \overset{!}{\geq} 1 \tag{176}$$

Disregarding the *stop velocity* condition, this leads to the definition of the particle *stop diameter* D_{stop}:

$$D_{stop} = \frac{6 D_{ff} \Delta t_p}{\pi \rho_p} \tag{177}$$

A particle with $D_{sph} \leq D_{stop}$ will be stopped and deposited with deposition probability $\theta_d(D_{sph})=1$, as soon as a fibre is touched. A particle with $D_{stop} \leq D_{sph} \leq D_{siev}$ yields higher momentum and stronger hydrodynamic forces and therefore is deposited with a probability $0 \leq \theta_d \leq 1$. D_{siev} marks a particle diameter below which sieving effects are still irrelevant for particle deposition. The PDDF $\theta_d(D_{sph})$, given by Equ.178 is supposed to be user definable and is based upon a simple, Gaussian-Normal distribution [30], defined by the two parameters: average D_{stop} and standard deviation σ_d.

$$\bigwedge_{D_{stop} \leq D_{sph} \leq D_{siev}} \Theta_d\left(D_{sph}\right) = \frac{1}{\sigma_d \sqrt{2\pi}} e^{-\frac{\left(D_{sph} - D_{stop}\right)^2}{2\sigma_d^{\,2}}} \tag{178}$$

The standard deviation σ_d is given via a third, user definable deposition constant σ_{slope}. The solver basically features three optional deposition modules:

1.) No adhesion-based deposition for $D_{sph} \geq D_{stop}$:

$$\sigma_d \rightarrow 0 \tag{179}$$

2.) σ_d depends directly on the user defined σ_{slope}:

$$\sigma_d = \sqrt{\frac{\sigma_{slope}}{2}} \tag{180}$$

3.) σ_d depends not only on the user defined σ_{slope} but also on fluid/particle properties and D_{ff}:

$$\sigma_d = \sqrt{\frac{D_{ff} \cdot 10^{\sigma_{slope}}}{2 \cdot D_{sph} \cdot \rho_p \cdot \nu_f}} \tag{181}$$

In order to decide whether the individual particle with $D_{sph} \geq D_{stop}$, featuring a deterministic deposition probability $\theta_d(D_{sph})$, is actually deposited at fibre contact, the stop parameter P_{stop} is defined by comparing θ_d to a randomized, rectangularily distributed variable θ_{01}.

$$P_{stop} = \Theta_d\left(D_{sph}\right) - \Theta_{01} \cdot \frac{1}{\sigma_d \sqrt{2\pi}} \tag{182}$$

The deposition condition is then:

$$P_{stop} \overset{!}{\geq} 0 \tag{183}$$

The bottom line at this point is:

1) Any particle that touches a fibre experiences $F_{\text{fibre,impact}}$ and looses its velocity component perpendicular to the fibre surface.

2) Any particle that touches a fibre experiences $F_{\text{fibre,gliding}}$ which reduces its velocity parallel to the fibre surface.

3) There are three ways for a particle to get deposited due to adhesion at fibre contact:
   a) The diameter of a volume equivalent sphere is below the stop diameter:

$$D_{sph} \leq D_{stop}\left(D_{ff}, \Delta t_p, \rho_p\right) \;\rightarrow\; N_{stop} \geq 1 \qquad (184)$$

   b) Due to the effect of $F_{\text{fibre,gliding}}$ the particle velocity falls below the stop velocity:

$$u_{p,\|} \leq u_{stop} \;\rightarrow\; N_{stop} \geq 1 \qquad (185)$$

   c) The stop number is smaller than 1 but the particle fulfils the stop parameter condition:

$$N_{stop} < 1 \;\wedge\; P_{stop} \geq 0 \qquad (186)$$

Any particle which fulfils either condition 3a, 3b or 3c will be deposited and is subject to a fibre constraining force $F_{\text{fibre,constrain}}$ which brings the remaining movement of the particle-fibre contact point to a halt:

$$\overline{F}_{fibre,constrain} = -\frac{\left(\overline{u}_p + \overline{\omega}_p \times \overline{r}_i\right) \cdot \overline{e}_{up,i,\|}}{\Delta t_p \cdot \left(\frac{1}{m_p} + \frac{1}{I_p}\left[\left(\overline{r}_i \times \overline{e}_{up,i,\|}\right) \times \overline{r}_i\right] \cdot \overline{e}_{up,i,\|}\right)} \cdot \overline{e}_{up,i,\|} \tag{187}$$

Here $e_{up,i,\|}$ is the base vector for the velocity of contact point i, $u_{p,i,\|}$ and r_i is the distance vector of contact point i to the particle mass centre.

The occurrence of additional particle forces F_i can still speed up the particle, so that condition 3b is void and $F_{fibre,constrain}=0$ or so that $F_{fibre,constrain}$ still is >0, but is outmatched by the N-1 other acting forces. A *blow-off* would be the consequence. It is up to the user to decide whether the consideration of *blow-off* effects is relevant, or whether adhesional deposition probability should work strictly governed by θ_d. The latter choice requires a more rigid fibre constraining force $F_{fibre,constrain,rigid}$:

$$\overline{F}_{fibre,constrain,rigid} = -\frac{\left(\overline{u}_p + \overline{\omega}_p \times \overline{r}_i\right) \cdot \overline{e}_{up,i,\|}}{\Delta t_p \cdot \left(\frac{1}{m_p} + \frac{1}{I_p}\left[\left(\overline{r}_i \times \overline{e}_{up,i,\|}\right) \times \overline{r}_i\right] \cdot \overline{e}_{up,i,\|}\right)} \cdot \overline{e}_{up,i,\|} - \sum_{i=1}^{N-1} \overline{F}_i \tag{188}$$

Numerical evaluations of adhesion-based filter fibre efficiency curves, on simplified, single fibre geometries have been conducted as seen in Figure 61. The test cases already include all the important dynamic interactions between particles, fluid, walls and fibres. Results clearly show that the task of embedding externally imposed PDDFs θ_d, to include experimentally studied particle-fibre interaction properties, into an otherwise discrete, deterministic system of force interactions, has been accomplished. Up to 100 mono-disperse, non-spherical particles have been injected, just upstream of a simplified fibre geometry, seen in Figure 61. The injection position is adjusted to the Stokes number of each particle such that the hitting of a fibre is assured. Otherwise injection is arbitrary. A deposition probability curve has been pre-defined as seen in Figure 62. Repeating the run with particles, featuring varying, relevant diameters D_{sph}, deposition efficiencies E for each particle class have been evaluated. As shown in Figure 62, the results match quite well.

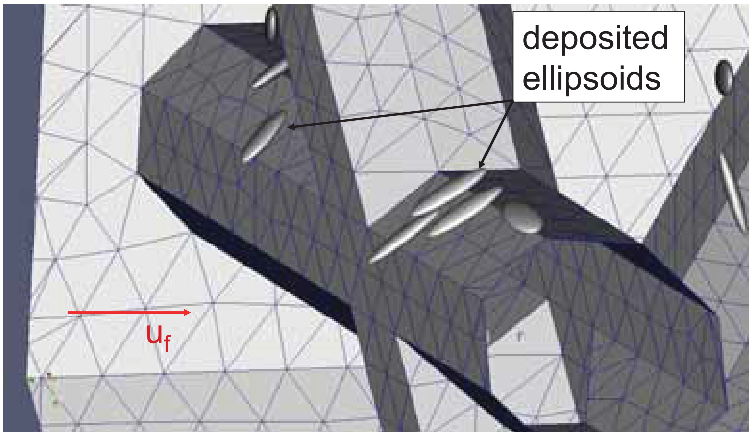


Figure 61: Benchmark case to test particle-fibre interaction module. Simple fibre geometry with ellipsoids being deposited due to particle-fibre adhesion.

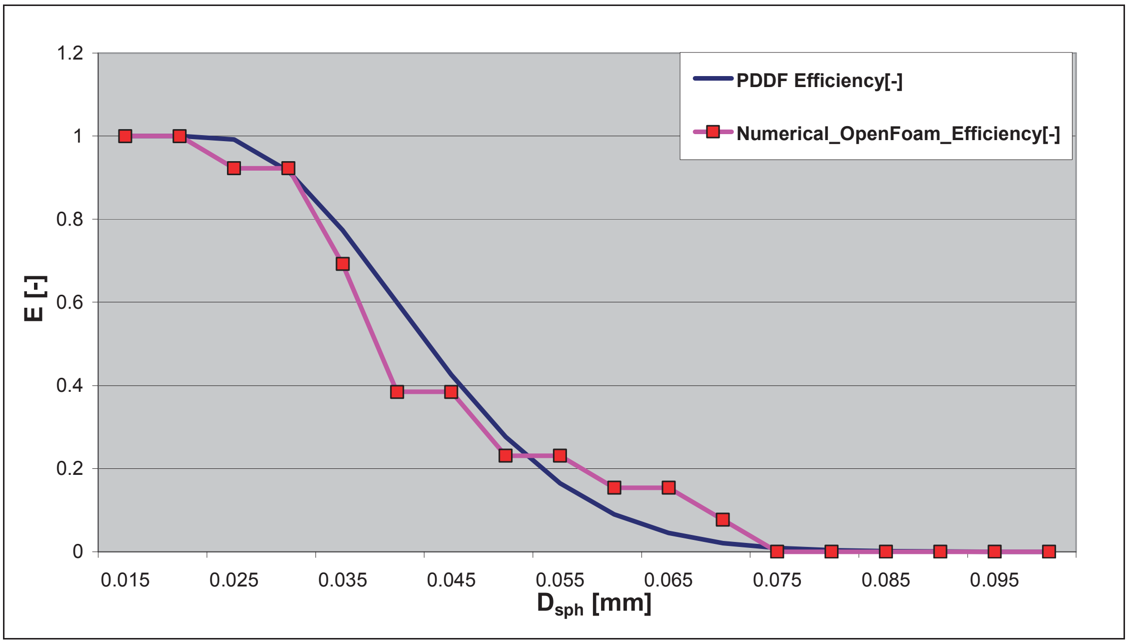


Figure 62: Adhesion-based, filter fibre efficiency for the simplified fibre structure case seen in Figure 61. Pre-defined PDDF for adhesion based deposition (blue) compared to numerical results (purple).

4.2.7.3 Particle–Particle Interaction

Within the fibre vicinity, high particle cloud densities occur. As more and more particles get entangled in the fibre, their volume fraction increases and particle-particle interactions become relevant. While hydrodynamic interaction is handled by the *fibre vicinity drag module* (local plugging, chapter 4.2.6), physical interactions (i.e. collisions) have to be considered as well.

As soon as a particle enters the fibre vicinity, its data (particle mass centre X_p, orientation of particle x- and y axes, e_x, e_y, particle mass m_p, and particle half axis diameters a, b, c) is stored in a dynamic *collision list*. The *collision list* is a data field with dynamic length M. Every one of the J help points of every one of the M particles within the fibre vicinity checks the entire collision list in M-1 steps and calculates whether or not its projected trajectory ends up within the volume of another particle. This amounts to a considerable calculation effort of the order O_1:

$$O_1[J \cdot M \cdot (M-1)] \tag{189}$$

A recently developed, more efficient approach towards the collision problem creates a *volume scalar list field* with one list entry for all N fluid cells. Here each particle is labelled and every help point stores the particle label in the local fluid cell entry. Thus a maximum of J*M scalar list field entries contain particle label-data and only J entries with L different particle labels will have to be checked per particle. The overall effort of this new method amounts to the order O_2:

$$O_2[J \cdot M \cdot L] \tag{190}$$

Since L is the number of other particles that could potentially collide with the checked particle per time step, L is usually considerably smaller than M, which means:

$$O_2 < O_1 \tag{191}$$

If any help point HP^A of particle A detects an impact with particle B, the following procedure is initiated:

1.) Calculation of impact point IP^B on particle B (not necessarily a particle help point HP^B). IP^B is determined as the intersection point of ellipsoid B and a straight line through HP^A and X_p^B. The problem is transformed into the co-rotational coordinate system of particle B where $X_p^B=0$, so that $IP^{B'}$ is:

$$IP^{B}{'} = HP^{A}{'} \frac{a^B \cdot b^B \cdot c^B}{\sqrt{x^{HP,A}{'} a^B c^B + y^{HP,A}{'} a^B c^B + z^{HP,A}{'} a^B b^B}} \tag{192}$$

Note that the superscript ‘ stands for coordinates within the co-rotational particle coordinate system. The concept of the co-rotational particle coordinate system is thoroughly described in chapter 4.2.2.2.

2.) Determination of collision vector n_{coll}. Due to the help point discretization of the ellipsoid particle surface, a calculation of surface normal vectors at impact points of particle A and B, would not yield two vectors of the same direction. An exact determination of ellipsoid surface normal n_B of particle B at IP^B is necessary to give the impact an appropriate direction. The calculation is performed within the co-rotational coordinate system of particle B and looks like:

$$\bar{n}_{coll}{'} = \bar{n}_B{'} = \frac{z^{IP,B}{'}}{\left|z^{IP,B}{'}\right|} \cdot \frac{1}{\sqrt{f_{Bxz}{}^2 + f_{Byz}{}^2 + 1}} \left(f_{Bxz}, f_{Byz}, 1\right) \tag{193}$$

Here f_{Bxz} and f_{Byz} are:

$$f_{Bxz} = -\frac{z^{IP,B}{'}}{\left|z^{IP,B}{'}\right|} \cdot \frac{c^2 \cdot x^{IP,B}{'}}{a^2} \sqrt{c^2 \left| 1 - \left(\frac{x^{IP,B}{'}}{a^2}\right)^2 - \left(\frac{y^{IP,B}{'}}{b^2}\right)^2 \right|} \tag{194}$$

$$f_{Byz} = -\frac{z^{IP,B'}}{\left|z^{IP,B'}\right|} \cdot \frac{c^2 \cdot y^{IP,B'}}{b^2} \sqrt{c^2 \left| 1 - \left(\frac{x^{IP,B'}}{a^2}\right)^2 - \left(\frac{y^{IP,B'}}{b^2}\right)^2 \right|} \tag{195}$$

3.) Modeling of the collision force $F_{collision}$. The collision force is designed to either reverse (100% elastic impact) or eliminate (100% plastic impact) the relative velocity $u_{p,rel}$, between HP^A and IP^B according to user-defined impact elasticity

E_{coll}. The direction of F_{coll} is given by n_{coll}. It obviously has to be reversed for collision force calculation for either one of the two collision partners. F_{coll} for particle A then reads:

$$\overline{F}^{A}_{Coll} = \frac{-(1+E_{coll})\cdot\left[\left(\overline{u}^{A}_{P}-\overline{u}^{B}_{P}\right)+\left(\overline{\omega}^{A}_{P}\times\overline{r}^{HP,A}-\overline{\omega}^{B}_{P}\times\overline{r}^{IP,B}\right)\right]\cdot\overline{n}_{coll}}{\Delta t_{p}\left[\frac{\overline{n}_{coll}}{m^{A}_{P}}+\frac{\overline{n}_{coll}}{m^{B}_{P}}+\frac{1}{I^{A}_{P}}\left(\overline{r}^{HP,A}\times\overline{n}_{coll}\right)\times\overline{r}^{HP,A}+\frac{1}{I^{B}_{P}}\left(\overline{r}^{IP,B}\times\overline{n}_{coll}\right)\times\overline{r}^{IP,A}\right]\cdot\overline{n}_{coll}}\cdot\overline{n}_{coll} \quad (196)$$

Here $r^{HP,A}$ stands for the distance vector of HP^{A} to $X_{P,A}$ and $r^{IP,B}$ stands for the distance vector of IP^{B} to $X_{P,B}$. All terms of Equ.196 have to be inserted with their last known values just before the impact. A similar collision force implementation is used by Shah & Megahed, [27]. A sketch of the non-spherical impact situation is shown in Figure 63.

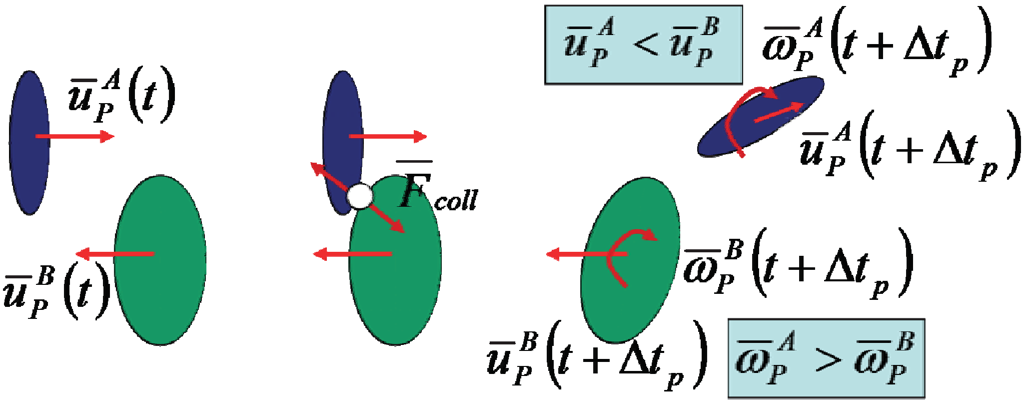


Figure 63: Sketch of impact situation with non-spherical particles A and B, before (left), during (middle) and after (right) collision. Post impact situation: The translational velocity of the smaller particle A is more strongly impaired and is smaller than that of particle B. The angular velocity of particle A is higher than that of particle B.

Figure 64 depicts screenshots from various stages of a simple collision benchmark case. Two differently sized ellipsoids are set on a slightly displaced collision course and interact just as predicted by the situation-sketch in Figure 63. Points of impact are correctly calculated, impact-forces are transferred and translational as well as rotational deceleration and acceleration can be observed just as expected.

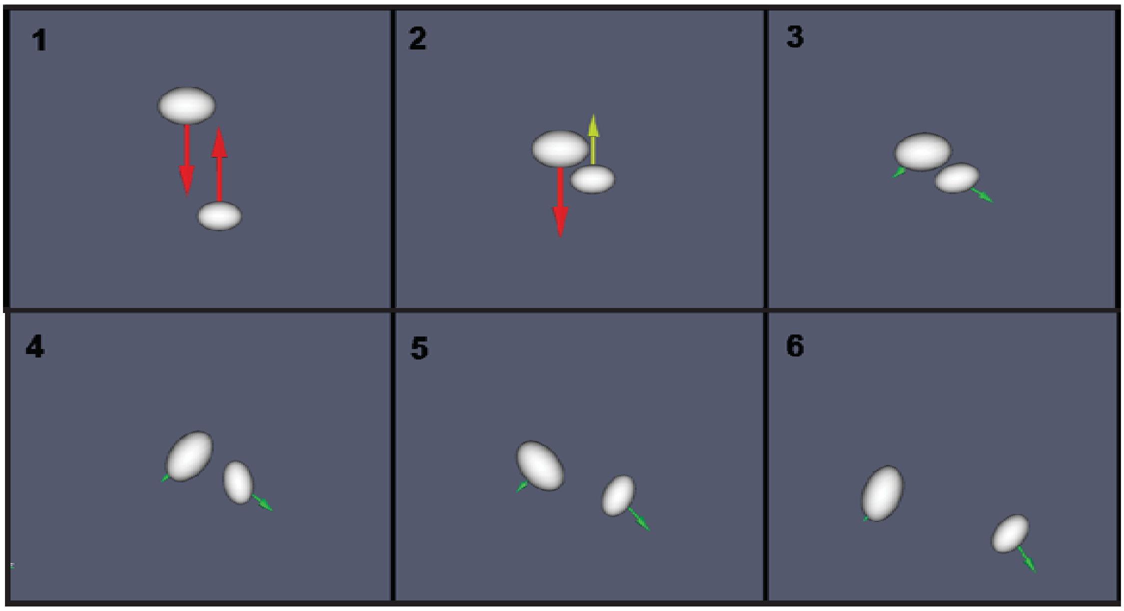


Figure 64: Six consequential screenshots of collision benchmark case. Two ellipsoids on displaced collision course collide and interact accordingly.

If larger accumulations of particles have to be dealt with, as is the case with *cake filtration* simulation, the situation gets more complex: More and more force vectors are exchanged, hardly any particle sub-time step passes without collisions and particles start colliding with more than one partner at a time. This is why, in those situations, not only the velocities, u_p and ω_p, are relevant for collision calculation, but also the entire spectrum of fluid and/or event forces momentarily acting on the individual particle. Consequently the scheme of mere collision detection and impact force calculation has to be extended by a suitable framework for transferring and accepting all occurring forces between the collision partners. Thus the transfer force term F_{trans} is introduced as:

$$\overline{F}_{trans} = \sum_{i=1}^{N-1} \overline{F}_i \qquad (197)$$

At initiation, the variable $F_{trans,0}$ comprises all N force terms acting on particle A during Δt_p, except for collision force terms where A serves as collision-initiator. This means that $F_{trans,0}$ *does* contain collision force terms that other particles have previously transferred to A (where A was the *non-initiating* collision partner).

Under the condition that the j^{th} collision event during Δt_p features a collision vector $n_{coll,j}$ so that $F_{trans,j-1}*n_{coll,j}<0$, the according collision force $F_{coll,A,j}$ is then calculated as an extension of the motion-based j^{th} collision force according to Equ.196, $F_{coll}^{A,j}$. (See Figure 65). It reads:

$$\overline{F}_{Coll,A,j} = \overline{F}_{Coll}^{A,j} - \left(\overline{F}_{trans,j-1} \cdot \overline{n}_{coll,j}\right) \cdot \overline{n}_{coll,j} \tag{198}$$

And $F_{trans,j}$ is consequentially set to:

$$\overline{F}_{trans,j} = \overline{F}_{trans,j-1} + \overline{F}_{coll,A,j} \tag{199}$$

With $r_{p,j}^{B}$ being the distance vector from the impact point to the mass-centre of the collision-partner-particle B, force and torque are then passed to B so that:

$$\overline{F}_{Coll,B,0} = -\overline{F}_{Coll,A,j} \tag{200}$$

$$\overline{M}_{Coll,B,0} = \overline{r}_{p,j}^{B} \times \left(-\overline{F}_{Coll,A,j}\right) \tag{201}$$

Figure 65 shows a sketch of an exemplary collision force transfer situation. The example assumes that a gravitational force F_g acts downwards and that collision event j=1 of centre particle A with particle C has already occurred. For this first collision two scenarios are possible:

- Particle C has already been calculated as collision centre before. Then $F_{trans,0}*n_{coll,1}<0$ and $F_{coll,C,1}= F_{coll,}^{C,1}+ f(F_{trans,0}(F_{g,C}))$.

- Particle C will be calculated as collision centre after particle A (as is assumed in the sketch). Then $F_{trans,0}= F_g$, $F_{trans,0}*n_{coll,1}>0$ and therefore $F_{coll,C,1}= F_{coll,}^{C,1}$. This means that Equ.198 and Equ.199 are not required, hence $F_{trans,0} = F_{trans,1}$.

The collision event j=2 of particle A with particle B is interesting, since $F_{trans,1}*n_{coll,2}<0$. Here $F_{coll,}^{B,2}$ is significantly enhanced by the reversed $F_{trans,1}$ component acting along $n_{coll,2}$.

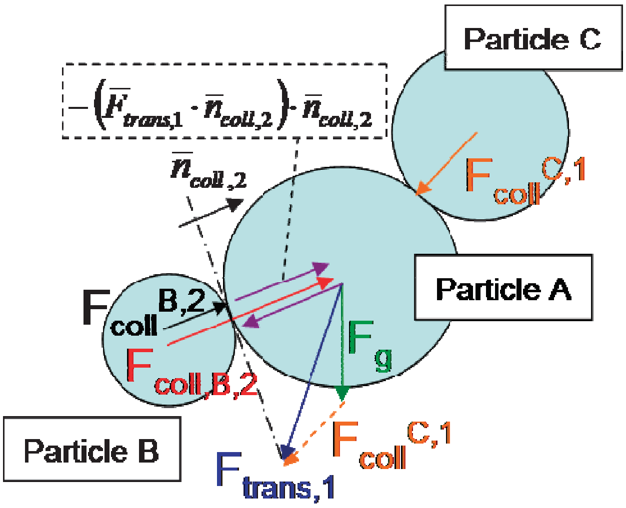


Figure 65: Sketch of exemplary collision force transfer situation.

This interaction implementation scheme constitutes the basis for getting a grip on any dense accumulation of (non-)spherical, arbitrarily sized particles. Two key problems within larger systems of dynamic particle interaction have turned out to be crucial in this context:

- Overall conservation of kinetic and potential energy: Local outbursts of rapid particle movement due to unphysically high force terms can occur.
- Lack of volumetric integrity: At times particles are pressed into each other and melt together.

Both problems can be addressed and amended by the herby described implementation scheme. By applying the given solution, a semi-dynamic equilibrium within any large particle collective can be achieved. If no external source of energy excites the dense particle accumulation, kinetic energy will slowly dissipate from the system, due to fluid drag forces. A final, settled, structurally integer state, governed by gravitation will emerge. Figures 66 and 67 show examples of densely packed, spherical and non-spherical particle collectives. Both cases converge to stable, physically plausible solutions.

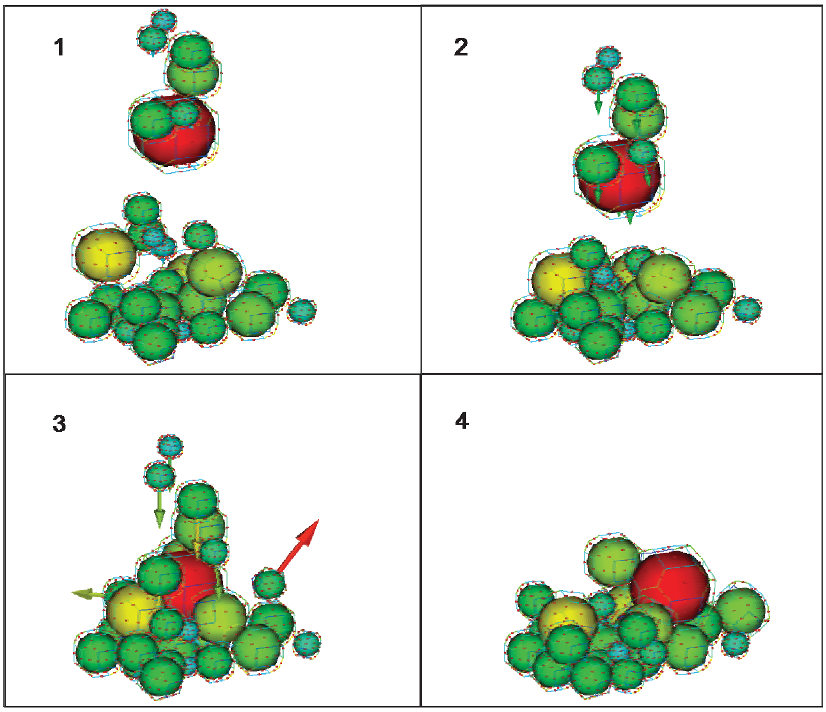


Figure 66: Four consequential screenshots of densely packed, arbitrarily sized, spherical particles driven downward by gravity. Velocity vectors, the particles' panel framework and their help points are also visible.

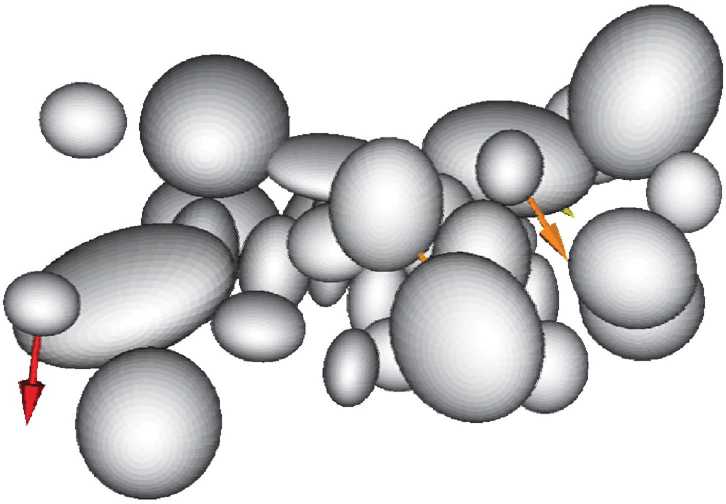


Figure 67: Screenshots of densely packed, arbitrarily sized, non-spherical particles. Gravity acts downward. Velocity vectors are visible. The collective has

already settled to a relatively dampened state. Remaining interaction forces are mainly based upon gravity and not upon particle movement anymore.

4.2.8 Qualitative Examples of Non-Spherical Dirt Particle Standard Filtration Solver Application

The features presented in chapters 4.2.2 through 4.2.7 constitute the physical framework for the most advanced and up-until now final version of the non-spherical dirt particle and deposition solver. In combination with the FSI-solver the particle simulator forms the unified filtration solver, i*coLagrangianNonSphericalStructFOAM*. In combination with the Digital Fibre Reconstruction utility, this new tool is capable of tracking (non-)spherical, four way coupled, Lagrangian dirt-particle clouds through a realistic reconstruction of a microscopic, deformable fibre geometry. It can be used to study process parameters like pressure drop, penetration depth and fibre efficiency.

In the following, some qualitative examples of application are shown. More intrinsic, quantitative, engineering applications are presented in chapter 9.

Figure 68 depicts an exemplary screenshot of the first benchmark case, which is a classic filtration application. It includes a dense cloud of non-spherical dirt particles, ranging from 30µm to 50µm in diameter. As they interact with fibre, fluid and amongst each other, the particles get entangled in a realistically reconstructed, deforming filter fibre geometry, measuring 180µm*180µm *200µm and with about $2.5*10^5$ fluid mesh cells. The screenshot gives a qualitative impression of the wide variety of effects which can be handled by the solver.

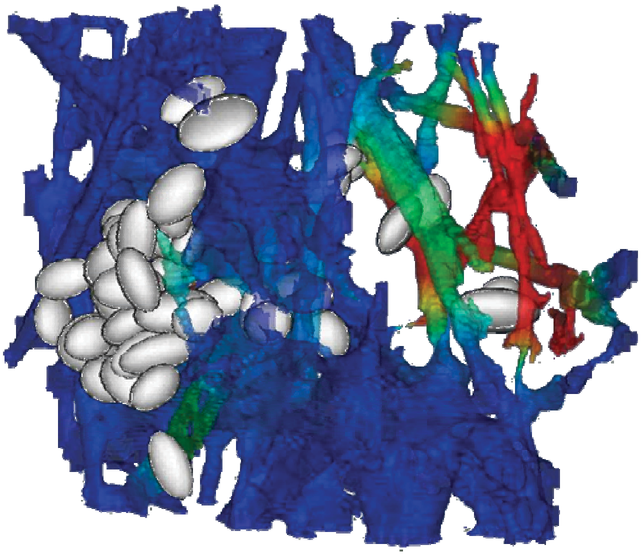


Figure 68: Screen shot of benchmark case *one*. Dense cloud of non-spherical particles (ρ_p=2000kg/m³, 20µm≤D_{sph}≤50µm) in viscous oil stream (u_f~0.1m/s, ρ_f=1000kg/m³, ν_f=1*10^{-6}m²/s) and realistic, deforming filter fibre geometry A, 200µm*200µm*200µm, ~6.0*10^5 cells. *Fibre vicinity drag module* is active, four-way coupling is engaged, deposition mechanisms are initiated and FSI utility is switched on.

The second benchmark case, shown in Figure 69, features a much larger, reconstructed, microscopic filter fibre sample, measuring 250µm * 250µm * 1000µm. The number of fluid mesh cells in this case is about 5*10^6. Again the sample is penetrated by an oil stream which is laden with spherical dirt particles. This geometry already represents an outtake of the entire real-life filter thickness. Results obtained by particle deposition calculation within a reconstructed fibre region of this size, have shown to be almost representative for some, homogeneous filter fibre media.

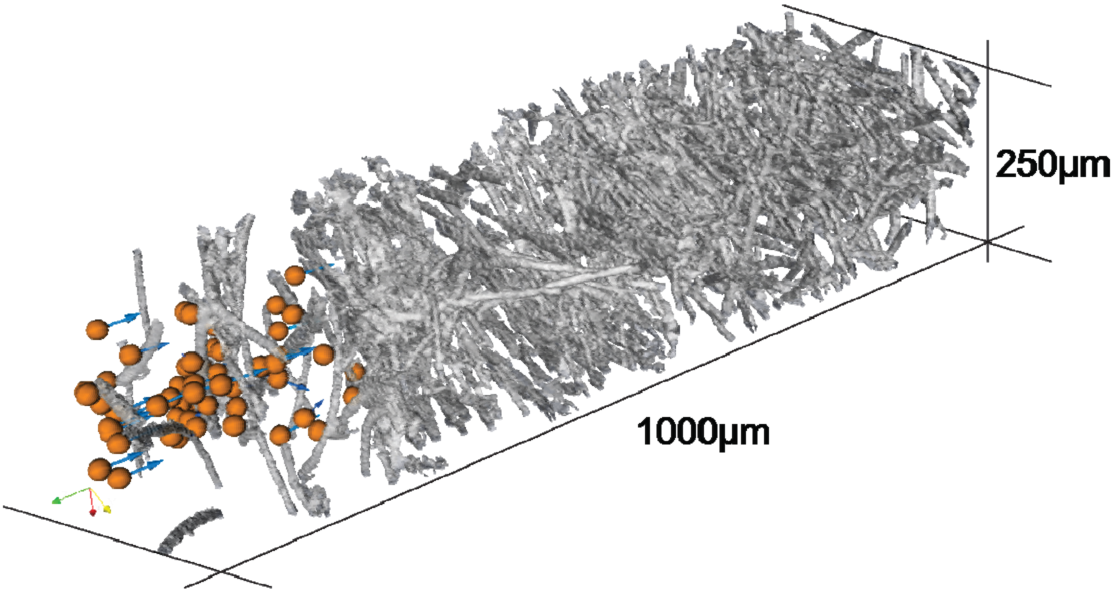


Figure 69: Screen shot of benchmark case *two*. Some spherical dirt particles (ρ_p=2000kg/m³, D_{sph}~25µm) are carried by oil stream (u_f~0.1m/s, ρ_f=1000kg/m³, $\nu_f=1*10^{-6}$m²/s) through large, realistic filter fibre geometry B, 250µm*250µm*1000µm, ~$5.0*10^6$ cells. *Fibre vicinity drag module* is active, four-way coupling is engaged and deposition mechanisms are initiated.

Simulation runs on the behaviour of specialized, shape-shifting, bacteria have been the focus of research behind benchmark case three. Initially spherical bacteria are hereby carried into a simplified fibre geometry. As soon as an entity hits one of the fibres, it deforms to the shape of a small plate as it attaches itself. Due to deposition and sub sequential reproduction, rather dense colonies start to form on the fibre surface. This case is supposed to show the versatility of the simulation tool.

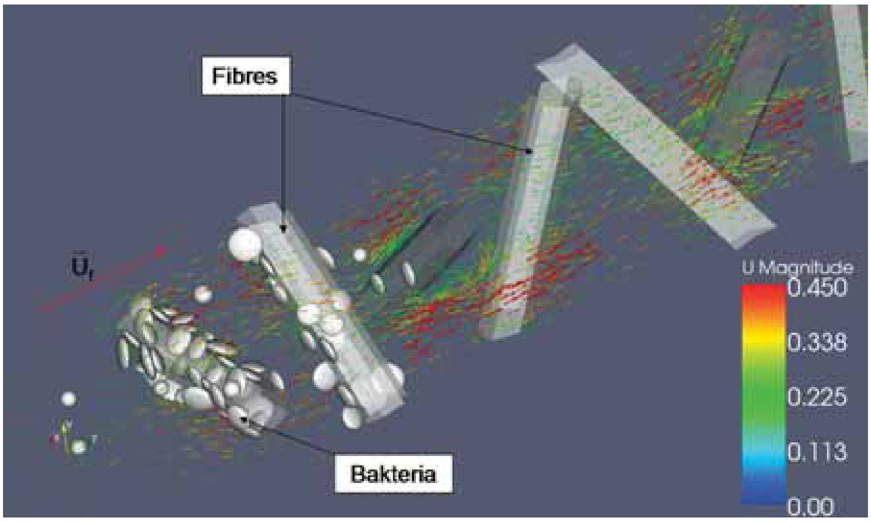


Figure 70: Screen shot of benchmark case *three*. Spherical and plate-like bacteria (D_{sph}~50µm, ρ_p=1100kg/m³) deposit in dense clusters within simplified fibre structure.

During the course of development of both particle solvers the matter of numerical instabilities in particle movement turned out to be an ever increasing problem. This is why an elaborate, adaptive time stepping scheme, suitable for spherical and non-spherical particles has been devised. The following chapter of this thesis intrinsically describes the problem, looks into particle shape-dependant influences on temporal discretization and finally presents a possible, quantifiable solution.

5 Adaptive Time Stepping for Explicit Euler Temporal Discretization of Spherical and Non-Spherical Particle Speed-Up

Numerical implementation schemes of drag force effects on Lagrangian particles can lead to instabilities or inefficiencies if static particle time stepping is used. Despite well known disadvantages, the programming structure of the particle solver led to the choice of an explicit *Euler*, temporal discretization scheme. To optimize the functionality of the *Euler* scheme, a method of adaptive time stepping, which adjusts the particle sub time step to the need of the individual particle is proposed here. A user-definable adjustment between numerical stability and calculation efficiency is sought and a simple time stepping rule is presented. Furthermore a method to quantify numerical instability is devised and the importance of the characteristic particle relaxation time as numerical parameter is underlined. All derivations are being conducted for (non-)spherical particles and finally for a generalized drag force implementation. Important differences in spherical and non-spherical particle behaviour are pointed out.

5.1 Introduction

Within the course of programming, all particle modelling was based on the calculation of individual, explicitly formulated force effects (see chapter 4.2.2). The application of commonly used, implicit discretization schemes such as the Runga Kutta Scheme [20], or semi-analytical approaches as described by Göz, Lain & Sommerfeld, [21], was found to be disadvantageous. Reasons for that are the wide variety of N individual forces $F_{i,j}$ acting during time step j, and the specific set-up of the code. Therefore an explicit *Euler* implementation of force effects on particle movement was selected. At constant time step Δt the *Euler* scheme constitutes an approximation of the semi-analytical approach of order $O(\Delta t)$, [20]. A well known problem of the simple, explicit *Euler* implementation is the occurrence of numerical instabilities. It has to be addressed.

So far the OpenFOAM® based, Lagrangian particle solvers have used a constant number of user-defined Subcycles J, to account for particle time scales, which are commonly smaller than fluid time scales. The simple relationship between fluid time step Δt_f, particle time step Δt_p and the number of particle Subcycles J then reads:

$$J = \left\lceil \frac{\Delta t_f}{\Delta t_p} \right\rceil = const \qquad (202)$$

With $J \in Z$, $\Delta t_p \leq \Delta t_f$, $J \geq 1$ and J=const. The ceiling function $\lceil \; \rceil$ is of course preferred here to a floor function $\lfloor \; \rfloor$, in order to be on the safe side. Per definition, particle time steps can not become greater than fluid time steps. Once the number of Subcycles is selected, it remains statically linked to the fluid time step, regardless of particle dimensions, material properties, or flow conditions. As a consequence, static time stepping can lead to inefficiencies and numerical instabilities.

Here a simple method of adaptive particle time stepping is proposed. It takes into account the properties of each individual particle, such as particle mass m_p, particle density ρ_p or particle velocity u_p, as well as local fluid and flow field properties, such as dynamic viscosity μ_f and fluid velocity u_f, so that:

$$J\left(\mu_f, u_f, u_p, \rho_p, \Delta t_f, m_p\right) = \left\lceil \frac{\Delta t_f}{\Delta t_p\left(\mu_f, u_f, u_p, \rho_p, m_p\right)} \right\rceil \qquad (203)$$

Furthermore this chapter presents a way to quantify the degree of numerical particle stability, which goes along with each chosen particle sub time step. Based on these results, an adaptive time stepping method is worked out, which allows the user to select accuracy and efficiency of the explicit, *Euler* temporal discretization.

5.2 Explicit Euler Temporal Discretization of Drag Force Effect on (Non-) Spherical Particles

The simple, explicit *Euler*, temporal discretization of particle movement under the influence of external forces for the jth time step can be written as:

$$\overline{u}_p\left(t_j = t_{j-1} + \Delta t_{p,j}\right) = \overline{u}_p\left(t_{j-1}\right) + \frac{1}{m_p} \cdot \sum_{i=0}^{N} \overline{F}_{i,j} \cdot \Delta t_{p,j} \qquad (204)$$

Issues concerning numerical instability, stemming from this implementation method, are frequently observed. Those instabilities turn out to be worst in connection with particle–fluid interaction force effects and are not as critical concerning momentary forces caused by events such as particle–particle, particle-wall or particle-fibre impacts (see chapter 4.2.7). Therefore particle-fluid forces have to be inspected primarily. Chapter 5.5 treats the consideration of other particle event forces as well.

5.2.1 Particle–Fluid Interaction: Drag Forces

The *Basset Bousinesque Oseen* (BBO) equation offers a complete, mathematical quantification of all possible interaction forces acting on objects being immersed in a fluid, [15, 21]. Going on from this generalized description, the problem needs to be simplified. Therefore the simple case of a particle, slowly speeding up in a uniform flow field shall be considered. The governing PME is then mainly dominated by fluid drag and lift forces, F_d and F_h respectively, so that:

$$m_p \frac{d\overline{u}_p}{dt} = \overline{F}_d + \overline{F}_h \qquad (205)$$

and for the jth time step:

$$\overline{u}_p\left(t_j\right) = \overline{u}_p\left(t_{j-1}\right) + \frac{1}{m_p} \cdot \left(\overline{F}_{d,j} + \overline{F}_{h,j}\right) \cdot \Delta t_{p,j} \qquad (206)$$

Lift is usually smaller than drag and is only relevant for arbitrarily positioned, non-spherical particles. It shall be disregarded here. As mentioned before, the general, well known formulation of the fluid drag force, for 1D, uniform flow conditions reads:

$$F_d = \frac{1}{2} c_d \left(\mathrm{Re}_p\right) \rho_f A_f \cdot u_{rel}^2 \qquad (207)$$

For spherical particles and low Reynolds numbers, Stokes' law is applicable to calculate the drag coefficient as seen in Equ.28. According to Haider & Levenspiel [23], there are in total well over 30 equations in literature, relating the drag coefficient of spherical particles to the Reynolds number.

The amount of equations in literature, describing the drag coefficient of non–spherical, e.g. ellipsoid, particles is significantly lower. Reviews on this subject have been conducted e.g. by Haider & Levenspiel [23] and Hölzer & Sommerfeld [24].

Hölzer & Sommerfeld have furthermore presented a c_d correlation formula for ellipsoids, which is reportedly valid over the entire range of Reynolds numbers. It has already been shown in Equ.129 and thoroughly presented in chapter 4.2.5.2. It is considered the most recent and so far the best, semi-analytical c_d correlation formula for arbitrarily positioned, non-spherical objects. Consequentially this work has adopted it for the discussion of non–spherical drag force implementation for *small* particles and the related time stepping problems.

However, the reader is instructed to keep in mind that the particle solver produces its own c_d and c_l characteristic, which is presented and compared to the Hölzer & Sommerfeld results in chapter 4.2.6.2.2.

5.2.2 Particle Speed Up

Previous works of e.g. Lain, Göz & Sommerfeld, [18]-[22], have considered the case of gravitational particle settling and specifically the value of the terminal particle settling velocity to study numerical instabilities. Since gravitational effects play a negligible role in the context of automotive oil filtration [70], a different method to numerically and analytically study time stepping and instability effects was chosen.

Hereby the case of a particle which speeds up in a uniform flow field shall be analyzed. The ratio between particle–wall distance and particle diameter shall be considered as large and any additional effects on the Lagrangian PME shall be disregarded as described above. Therefore, in extension of Equ.205, the simplified particle momentum equation reads:

$$\frac{du_p}{dt} = \frac{1}{2} c_d \frac{\rho_f A_f}{\rho_p V_p} u_{rel}^2 \tag{208}$$

5.2.2.1 Speed-Up of Spherical Particles

The characteristics of the speed up curve of any particle being inserted at velocity u_p=0 m/s, into a fluid flow of uniform velocity u_f, depend mainly on the implementation of the drag coefficient. For spherical particles and low Reynolds numbers the analytical solution for the development of particle velocity from *zero*, infinitely close to u_f, against time, can be found easily [16]. It is:

$$u_p = u_f - u_{rel,0} \cdot e^{-\frac{18\mu_f}{\rho_p D_{sph}^2} t} \tag{209}$$

Here $u_{rel,0}$ is the relative particle-fluid velocity at t=0s. Figure 71 shows the plot of an exemplary, spherical particle speed up curve and the usual, graphical interpretation of the characteristic, spherical particle relaxation time $\tau_{p,sph}$ (see chapter 5.3.1).

5.2.2.2 Speed-Up of Non–Spherical Particles

In the case of non–spherical particle speed up, an analytical solution for the speed up curve can not be found that easily. By inserting the c_d correlation of Equ.129 into the simplified PME of Equ.208 and by consequential integration over time and relative particle–fluid velocity, the following expression is reached:

$$t(u_{rel}) = \frac{C_0}{C_1}\left(\frac{2C_2 ArcTan\left[\frac{C_2 + 2C_3\sqrt{u_{rel}}}{\sqrt{-C_2^2 + 4C_1C_3}}\right]}{\sqrt{-C_2^2 + 4C_1C_3}} + \log(u_{rel}) - \log\left(C_1 + C_2\sqrt{u_{rel}} + C_3 u_{rel}\right) \right)\Bigg|_{U_{rel,0}}^{U_{rel}} \qquad (210)$$

The constants C_0, C_1, C_2 and C_3 are declared in Equ.6 through Equ.9. Equ.210 however, is transient in nature, therefore a solution for u_p=f(t) can only be obtained numerically. The Newton-Raphson procedure [31] is used to get a plot of the results of the explicit solution for u_p out of Equ.210 (see Figure 71).

One focus of chapter 5 is to point out the significant difference in speed up behaviour between spherical and non-spherical particles, which share the same volume equivalent spherical diameter D_{sph}. Figure 71 shows a direct comparison of the speed up behaviour of an exemplary spherical particle and a non-spherical particle of equal mass and volume, with arbitrarily chosen sphericity and under matching flow conditions. Assuming the qualitative, physical correctness of the non-spherical drag implementation of Equ.129, the decreased sphericity leads to decreased particle relaxation time, increased drag forces and thus to faster particle speed up.

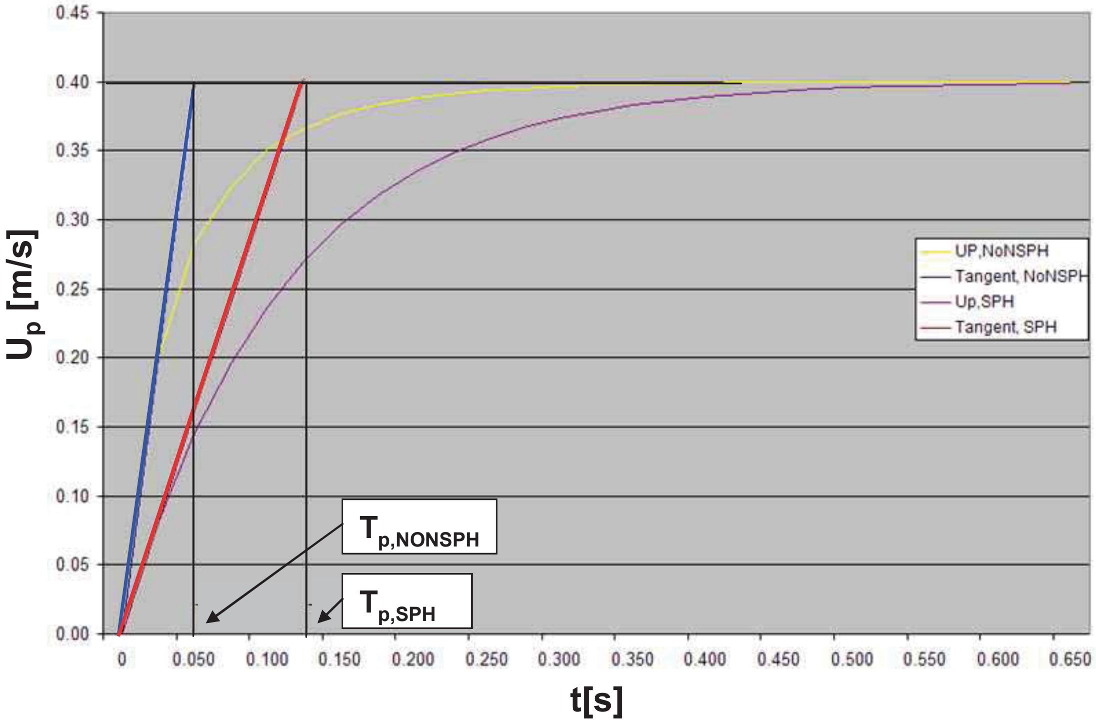


Figure 71: Plot of particle velocity u_p[m/s] against speed up time t[s]. Comparison of spherical and non-spherical particles with the same mass and D_{sph}=0.01m, speeding up under matching, uniform 1D flow conditions, μ_f=0.1Pas, u_f=0.4m/s. Graphical interpretation of spherical particle relaxation time $\tau_{p,sph}$ and non-spherical particle relaxation time, $\tau_{p,nonsph}$.

5.2.3 Numerical Instability of Explicit Euler Drag-Force-Effect-Implementation

For a given, exemplary fluid time step, fluid properties, spherical particle dimensions, particle density and flow conditions, a variation of particle Subcycles easily reveals the weakness of static particle time stepping. Figure 72 shows a plot of three numerically calculated, spherical particle speed up velocity curves, with the number of Subcycles J being the parameter. The results are compared to the correct, analytical solution of Equ.209.

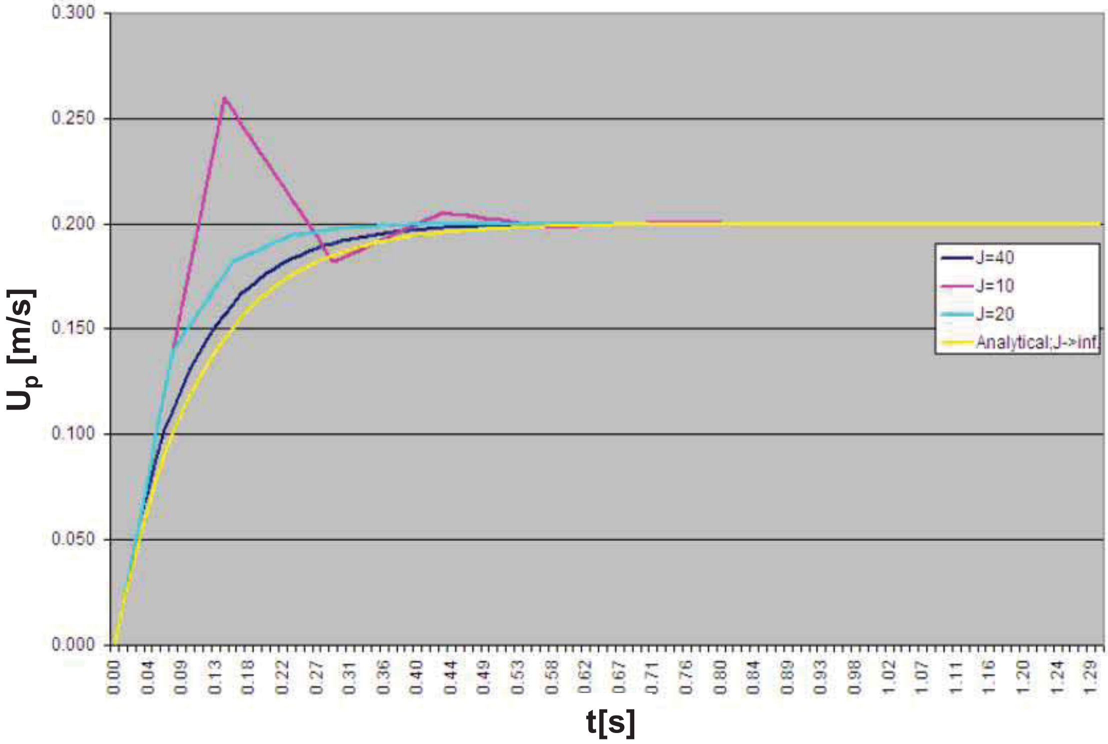


Figure 72: Plot of particle speed up curves for exemplary fluid time step Δt_f=1.333s, particle dimension D_{sph}=0.01m, dynamic fluid viscosity μ_f=0.1Pas, particle density ρ_p=2000 kg/m³, and uniform flow velocity u_f=0.2m/s. Three numerical speed up curves for static Subcycles J= 40 (blue), J=20 (turquoise) and J=10 (purple) are compared to the analytical solution (yellow).

At decreasing numbers of Subcycles, e.g. increasing particle time steps, the speed up curves show increasing deviation from the analytical solution (Figure 72, J=40, J=20). If a certain particle time step limit is exceeded, the particle starts experiencing unsteady acceleration (Figure 72, J=10), and for even larger time steps the numerical solution collapses altogether.

A similar behaviour of the results can be observed if the number of Subcycles is held constant, but the particle diameter is in turn decreased, or fluid viscosity is increased. When drag forces on non–spherical particles are considered, an additional parameter to be taken into account is the local, relative fluid–particle velocity u_{rel}. Here an increase of relative velocity has analogous effect to a decrease of Subcycles.

5.3 Particle Relaxation Time and Study of Non-Spherical Speed-Up Behaviour

In this chapter the well known definition of the particle relaxation time for spherical particles will be compared to an expression for the non–spherical particle relaxation time. Moreover the dependence of $\tau_{p,nonsph}$ on local fluid conditions and on the degree of non–sphericity will be investigated. Therefore a new quantification method to describe sphericity will be introduced.

5.3.1 Spherical Particle Relaxation Time

Out of Equ.209, a well known, essential parameter for the particle speed up characteristic can be derived. It defines the time scale for any individual, spherical particle under *Stokes* drag conditions, is called the spherical particle relaxation time and has already been shown in Equ.4. For the sake of consistency and to underline its importance it shall be written out again:

$$\tau_{p,sph} = \frac{D_{sph}^2 \rho_p}{18\mu_f} \tag{211}$$

The graphical interpretation of the parameter $\tau_{p,sph}$ is given by the intersection point of the speed up curve tangent at t=0s with the $u=u_f$ line and is depicted in Figure 71.

It is worth noting that, by using *Stokes'* law, $\tau_{p,sph}$ depends only on material properties, regardless of local flow conditions. By inserting Equ.211 into Equ.209, it can be rewritten as:

$$u_p = u_f - u_{rel,0} \cdot e^{-\frac{t}{\tau_{p,sph}}} \qquad (212)$$

5.3.2 Non-Spherical Particle Relaxation Time and Speed-Up Behaviour

As previously discussed, Equ.210 yields an expression for a non–spherical particle speed up curve, implicitly containing u_p (within u_{rel}). Equ.210 does not necessarily have to be evaluated numerically for $u_p=f(t)$, to obtain essential parameters of the speed up curve. Figure 73 shows an exemplary plot of Equ.210 and reveals that the characteristic non–spherical particle relaxation time can be extracted from this implicit expression for u_p as well.

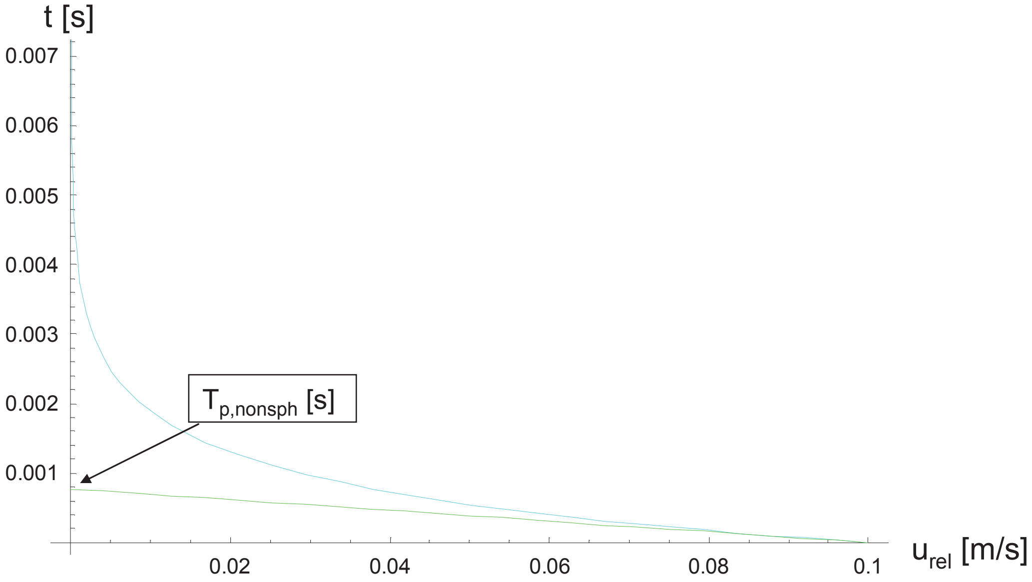


Figure 73: Exemplary plot of non-spherical particle speed up, for D_{sph}=0.001m, u_f=0.1m/s, μ_f=0.1Pas with tangent at t=0s and non–spherical particle relaxation time $\tau_{p,nonsph}$ =8*10^{-4} s.

Analogous to the spherical speed up case, the parameter $\tau_{p,nonsph}$ is given by the intersection point of the speed up curve tangent at t=0 s with the u_{rel}=0 m/s line. Therefore $\tau_{p,nonsph}$ is defined by:

$$\tau_{p,nonsph} = -u_{rel,0} \cdot \frac{dt(u_{rel})}{du_{rel}} \left(u_{rel} = u_{rel,0} \right) \qquad (213)$$

Uniting Equ.213 with the expression for t(u_{rel}) out of Equ.210 and with the definitions of C_0, C_1, C_2 and C_3, found in Equ.6 through Equ.9, the non-spherical particle relaxation time for *Hölzer-Sommerfeld drag* implementation can be found. It has already been shown in Equ.5. For the sake of consistency and to underline its importance it shall be written out again:

$$\tau_{p,nonsph} = \tau_{p,H/S} = -\frac{C_0}{C_1 + C_2\sqrt{u_{rel}} + C_3 u_{rel}} \tag{214}$$

The non–spherical particle relaxation time, based on Equ.214, depends not only on material properties and particle dimensions, but also on the local, relative fluid–particle velocity. As a consequence $\tau_{p,nonsph}$ depends on the local particle Reynolds number while $\tau_{p,sph}$ does not.

Even an actual particle sphericity of $\Phi=1$ does not eliminate the velocity dependence in Equ.214. The plot in Figure 74, which holds true for any set of particle properties, shows that, the larger the particle Reynolds number becomes, the worse the accordance of the two τ_p implementations for two identical, spherical particles will be. This reflects the fact that the *Stokes* version for spherical particle drag is only valid in the *zero* Reynolds limit.

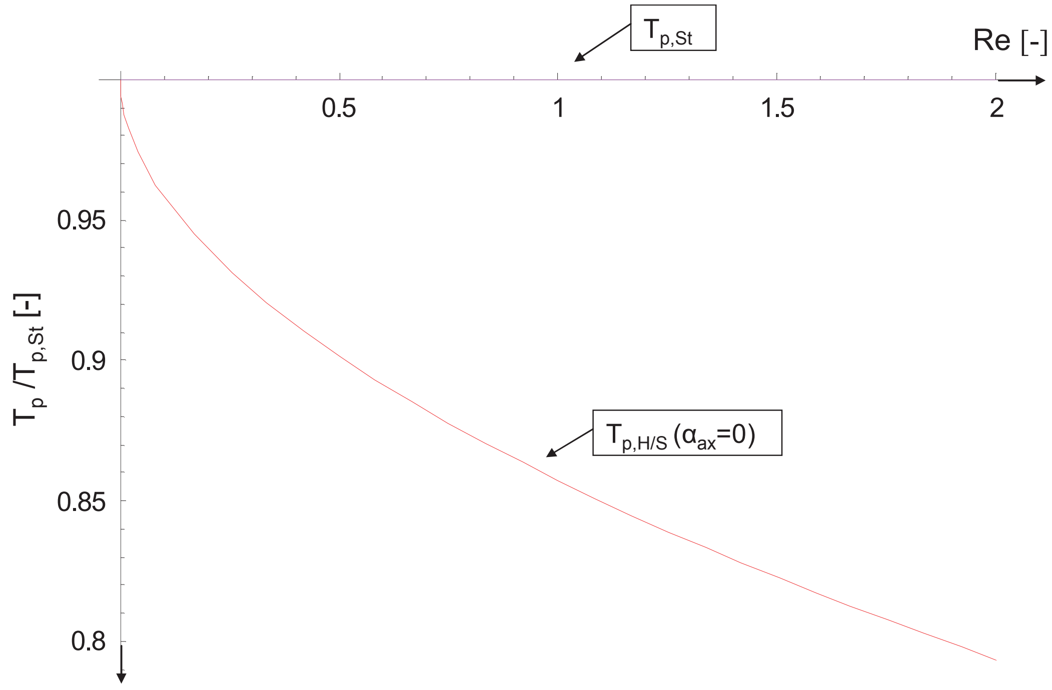


Figure 74: Ratio of spherical particle relaxation time according to *Hölzer-Sommerfeld* drag $\tau_{p,H/S}$ and spherical particle relaxation time according to Stokes' drag implementation $\tau_{p,St}$ plotted against particle Reynolds number. The higher the Reynolds number, the smaller $\tau_{p,H/S}$ compared to $\tau_{p,St}$. The Stokes' approach loses its validity.

Further evaluation of Equ.214 helps to get an idea of the relationship between particle shape and particle relaxation time. Figure 8 contains a plot of τ_p against Re_p for particles of varying sphericity, but constant D_{sph}. Furthermore Figure 8 demonstrates the basic difference between velocity dependence of "*Stokes' drag spheres"* and *"Hölzer-Sommerfeld ellipsoids"*. In addition to that, the plots in Figure 8 qualitatively show that, the further the particle shape is from being a sphere, the smaller τ_p becomes.

The sphericity can not fully describe the measure of similarity to spherical shape, - crosswise and lengthwise sphericity are needed as well. Therefore an alternative parameter to measure similarity to a sphere is introduced in this work. It is the ratio between the standard deviation of the half axes a, b and c around D_{sph}. This parameter is denoted as α_{ax} and has been shown in Equ.10.

Considering Figure 8 it becomes obvious that higher values of α_{ax} signify higher deviation from spherical shape and clearly lead to smaller $\tau_{p,nonsph}$ values. The

$\tau_{p,nonsph}$ values decrease further for higher particle Reynolds numbers. As has been stated in the introduction to this thesis, the plot in Figure 8 strongly makes the case for the consideration of particle shape effects in particle calculations. It shows that for values of $\alpha_{ax} \geq 1$, the non–spherical particle relaxation time becomes less than 1/5th of the relaxation time of a volume equivalent sphere.

Figure 75 is a plot of $\tau_{p,nonsph}/\tau_{p,sph}$ against α_{ax}. With the particle Reynolds number being used as parameter, this plot is valid, regardless of material properties or particle diameter D_{sph}. The plot shows that increasing deviation from spherical shape leads to a strong reduction of $\tau_{p,nonsph}$ values. The trend starts levelling of for $\alpha_{ax} \geq 1.5$. Higher particle Reynolds numbers lead to a $\tau_{p,nonsph}$ reduction as well. This effect is more pronounced for nearly spherical particles ($\alpha_{ax} \rightarrow 0$) and mainly within the creeping flow regime (Re<<1).

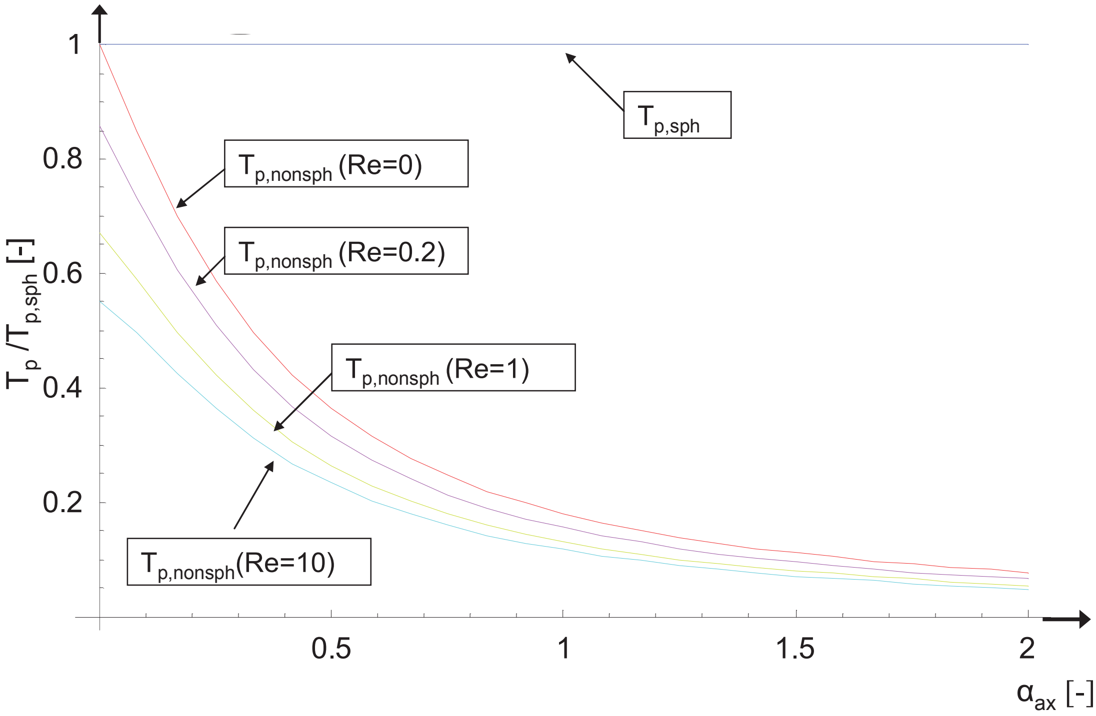


Figure 75: Spherical (blue) and non–spherical (red, orange, yellow, turquoise) particle relaxation time behaviour against relative half axes deviation around D_{sph}. Assumption: The longest particle half-axe *a* is aligned along fluid stream lines. Increasing Re_p (0-10) and increasing α_{ax} lead to smaller $\tau_{p,nonsph}$. All values are scaled by $\tau_{p,sph}(Re_p=0)$.

The analysis of Figure 75 shows that the non–spherical particle implementation leads to generally lower relaxation time values than that of the spherical case. Even if the ellipsoid half axes are of equal length ($\alpha_{ax}=0$), the non–spherical results deviate from the spherical implementation. Only if the particle Reynolds number reaches the limit $Re_p \rightarrow 0$ and $\alpha_{ax}=0$, a match is achieved.

In essence all those insights lead to the conclusion that non–spherical particles have smaller characteristic time scales than volume equivalent spheres. As a consequence, they require higher time step resolution. The stronger the non–sphericity, the smaller the time step will have to be to achieve numerical stability. If maximum efficiency is desired, non–spherical time steps can be increased as a particle accelerates, and relative velocities, as well as particle Reynolds numbers decline. However if, on the other hand, the chosen time step criterion is adjusted to the situation of highest possible particle Reynolds numbers, e.g. to the instant of particle injection, it will surely hold for the entire calculation.

5.3.3 Generalized Particle Relaxation Time

The expressions for spherical (Equ.211) and non–spherical (Equ.214) particle relaxation times can easily be extended to a generalized version. It holds for arbitrarily shaped particles and can be written as (Lain, Bröder, Sommerfeld, 1999, [18]):

$$\tau_p = 2\frac{m_p \cdot D_{sph}}{A_f \cdot \mu_f \cdot \mathrm{Re} \cdot cd(\mathrm{Re})} \tag{215}$$

5.4 Adaptive Time Stepping

After pin-pointing the problem, formulating expressions for particle relaxation times and after examining spherical as well as non-spherical speed up behaviour, a scheme of *adaptive time stepping* can be sought. First the multi-parameter character of the problem has to be reduced.

5.4.1 One Parameter to Define Numerical Stability

By inserting the generalized particle relaxation time τ_p (Equ.215) into the simplified PME (Equ.208) and by consequential, temporal discretization and substitution for $u_{rel}=u_f-u_p$ one obtains:

$$\frac{\Delta t_p}{\tau_p} = -\frac{\Delta u_{rel}}{u_{rel,0}} = \frac{u_{p,t} - u_{p,0}}{u_f - u_{p,0}} \qquad (216)$$

Here Δt_p, is the numerical particle sub time step and $u_{p,t}$ is the particle velocity after Δt_p. With Equ.216 a simple formula is given, which relates the ratio of the chosen particle time step and particle relaxation time to the ratio of relative particle–fluid velocity change, Δu_{rel} and relative particle–fluid velocity, $u_{rel,0}$ before Δt_p.

It is quite clear that if the ratio on the right hand side of Equ.216 gets larger than 1, the particle at its new velocity will in any case travel faster than the surrounding flow field. A result like this is not only wrong, but will cause the particle to accelerate in the opposite direction in the following time step. As will be shown, a $\Delta t_p/\tau_p$ ratio of ≥ 2 will collapse the particle calculation as a whole. The numerical stability of the calculation can only be guaranteed by substantially reducing the $\Delta t_p/\tau_p$ ratio. It turns out that the terms in Equ.216 are the single most important quantities to measure the extent of numerical (in-)stability of the calculation.

Equ.216 takes the subject away from being a multi parameter problem, which depends on particle dimension, fluid viscosity, fluid velocity and particle density, towards being a single parameter issue, which depends only on the ratio of particle sub time step to particle relaxation time $\Delta t_p/\tau_p$.

The first obvious conclusion is to start scaling the time axis by τ_p and to start expressing the degree of numerical stability by $\Delta t_p/\tau_p$.

5.4.2 Describing the Instabilities

To get a hold of the encountered instabilities, it is first necessary to thoroughly understand and describe them. The key to do that is to consider the iterational effects of the *Euler* scheme on velocity evolution. Using i as index for the specific iteration at runtime $t=\Delta t_p{*}i$, Equ.216 can be rewritten to:

$$u_i = u_{i-1} + \frac{u_{rel,i-1}}{\tau_p}\Delta t_p \qquad (217)$$

With particle velocity at iteration i=0 being $u_{p,0}=0.0$ m/s and with the relative fluid–particle velocity at that time consequentially being $u_{rel,0}=u_f$, the implicit statement of Equ.217 can be transferred into the following explicit expression for u_i:

$$u_i = u_f\left[(-1)^{i+1}\cdot\left(\frac{\Delta t_p}{\tau_p}-1\right)^i+1\right] \qquad (218)$$

The evaluation of Equ.218 for various ratios of $\Delta t_p/\tau_p$ is depicted in Figure 76. Obtained results immediately show that Equ.218 accurately explains the encountered instabilities which are partly shown in Figure 72.

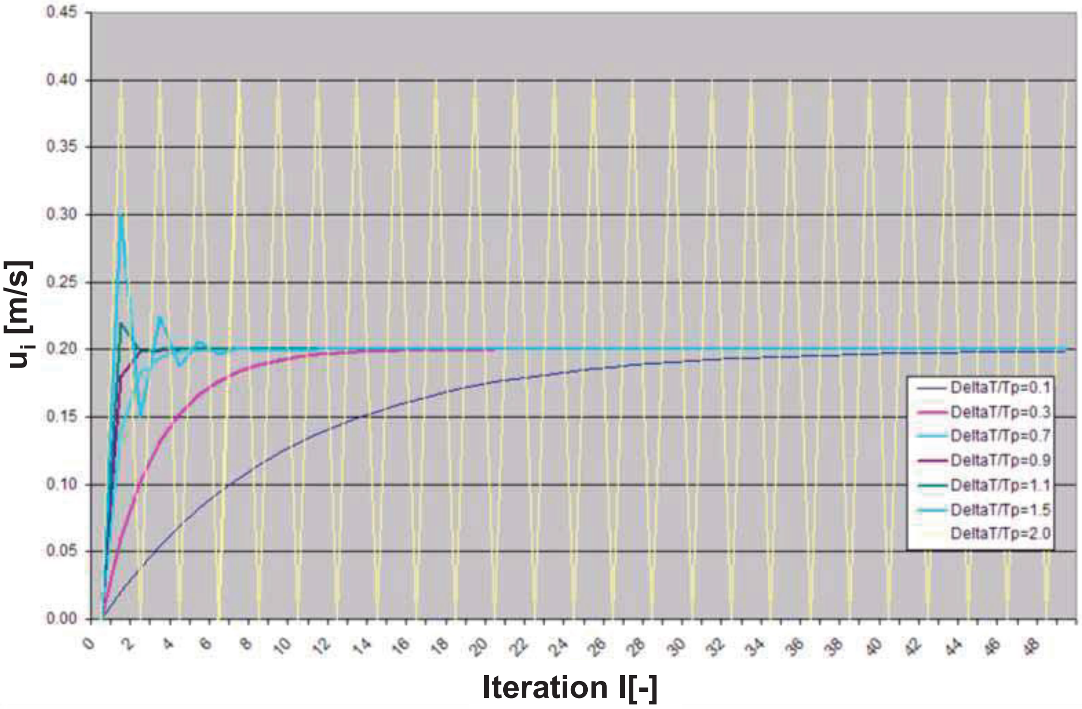


Figure 76: Evaluation of Equ.218 for varying $\Delta t_p/\tau_p$, with u_f=0.2 m/s. For $\Delta t_p/\tau_p \geq 1$ particle velocity evolution starts showing unsteady behaviour. For $\Delta t_p/\tau_p \geq 2$ the particle calculation collapses. The results accurately match the instability behaviour encountered in the OpenFOAM® solver (Figure 72).

By taking a look at Equ.218, the initial assumption, that $\Delta t_p/\tau_p$ is *the* decisive numerical parameter can be confirmed. The following facts can be stated:

1.) For $\Delta t_p/\tau_p \leq 1$ and for all $i \in Z$:

$$\lim_{i \to \infty} \left(\frac{\Delta t_p}{\tau_p} - 1 \right)^i = 0 \tag{219}$$

$$\lim_{i \to \infty} u_i = u_f \tag{220}$$

$$(-1)^{i+1} \cdot \left(\frac{\Delta t_p}{\tau_p} - 1 \right)^i \leq 0 \quad \rightarrow \quad u_i \leq u_f \tag{221}$$

The calculation will remain stable and the particle velocity will steadily converge to u_f. Overall numerical error will increase as $\Delta t_p/\tau_p$ increases.

2.) For $1<\Delta t_p/\tau_p<2$ and for all $i \in Z$:

$$\lim_{i \to \infty} \left(\frac{\Delta t_p}{\tau_p} - 1 \right)^i = 0 \tag{222}$$

$$\lim_{i \to \infty} u_i = u_f \tag{223}$$

For all $i \in Z_{even}$:

$$(-1)^{i+1} \cdot \left(\frac{\Delta t_p}{\tau_p} - 1 \right)^i > 0 \quad \rightarrow \quad u_i > u_f \tag{224}$$

For all $i \in Z_{uneven}$:

$$(-1)^{i+1} \cdot \left(\frac{\Delta t_p}{\tau_p} - 1 \right)^i < 0 \quad \rightarrow \quad u_i < u_f \tag{225}$$

In the $1<\Delta t_p/\tau_p<2$ regime the particle velocity will eventually converge to u_f, just as before, but it will show completely unsteady velocity jumps, oscillating around u_f.

3.) For $\Delta t_p/\tau_p= 2$ and for all $i \in Z$ the particle velocity oscillates unsteadily between $u_i= 0$ and u_f until the geometry boundaries are reached.

4.) For $\Delta t_p/\tau_p>2$ and for all $i \in Z$:

$$\lim_{i\to\infty}\left(\frac{\Delta t_p}{\tau_p}-1\right)^i=\infty \tag{226}$$

$$\lim_{i\to\infty} u_i=\infty \tag{227}$$

If $\Delta t_p/\tau_p>2$ the particle velocity will explode and the calculation will collapse. The consequence of this analysis is simple: $\Delta t_p/\tau_p$ must stay well below 1 to ensure steady evolution of particle velocity. What remains to be done is to quantify the extent of numerical error within the “regime of steady velocity evolution”.

5.4.3 Quantification of Numerical Error

Numerical error is best quantified by considering its effects. Here the resulting speed up curve for any specific $\Delta t_p/\tau_p$ shall be compared to the correct, analytical solution. An explicit, analytical solution is only known for spherical particles, accelerating under *Stokes*’ drag conditions (see Equ.209). For non–spherical particles, the $\Delta t_p/\tau_p$ speed up curve shall be compared to a numerically calculated reference curve, of small, yet basically variable $\Delta t_{p,0}/\tau_p$. As quantitative measure of the overall amount of deviation between numerical and *reference* results, the medium standard deviation σ shall be chosen. The medium standard deviation is calculated according to Equ.228.

$$\sigma\left(\Delta t_p / \tau_p\right)=\sqrt{\frac{\sum_{i=1}^{i_{\max}}\left[u_{n,i}\left(\Delta t_p / \tau_p\right)-u_{a,i}\right]^2}{i_{\max}}} \tag{228}$$

Here the index *i* indicates the individual, numerical time step, the index *n* indicates a result from the numerical solution for $\Delta t_p/\tau_p$ and index *a*, indicates a result from the reference (analytical) solution. Let the parameter M denote the last compared velocity point at runtime t_{end}, so that:

$$M=\frac{t_{end}}{\tau_p} \tag{229}$$

Then the total number of compared, discrete time steps i_{max} is:

$$i_{\max}=\left\lfloor\frac{M\cdot\tau_p}{\Delta t_p}\right\rfloor\in \mathrm{Z} \tag{230}$$

The exemplary plot of two compared speed up curves in Figure 77 illustrates the numerical error quantification scheme.

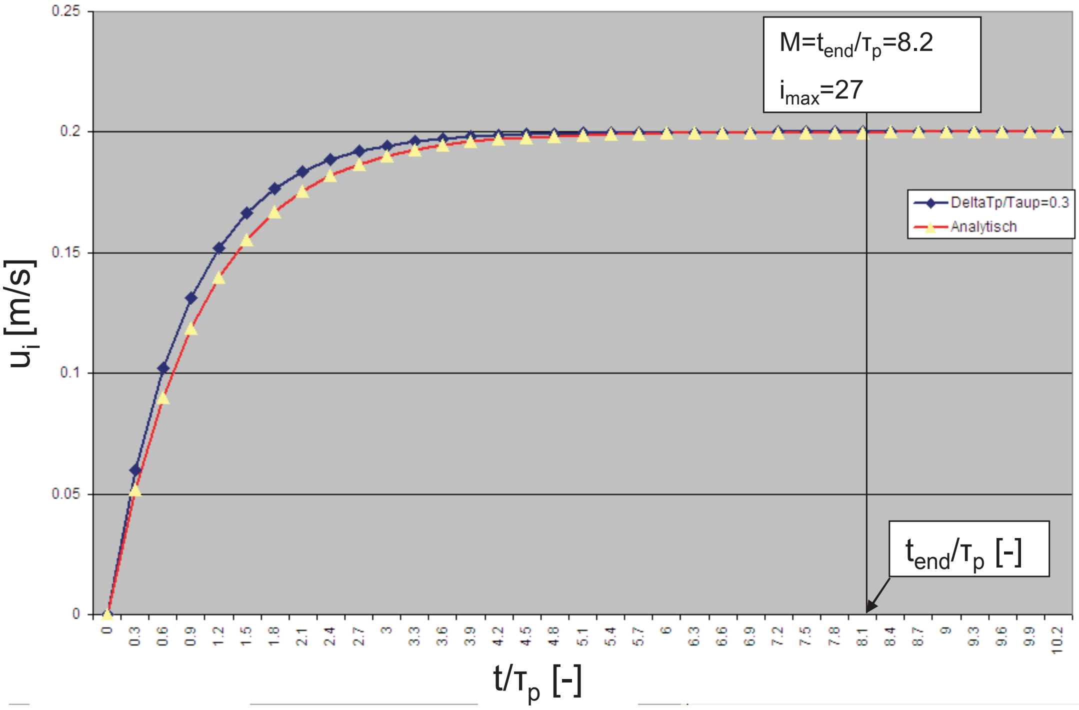


Figure 77: Comparison of analytical (red) and numerical (blue) speed up curve with $\Delta t_p/\tau_p$=0.3 and u_f=0.2m/s. The points of numerical evaluation are shown according to the chosen particle sub time step. To calculate σ, velocity points from t/τ_p =0.3 to M=t_{end}/τ_p =8.2 are chosen. In this case the number of compared velocity points is i_{max}=27.

5.4.3.1 Quantification of Spherical, Numerical Error

By using Equ.209 and by representing the particle runtime as t= Δt_p*i, the analytical solution for iterational particle velocity for spherical particles becomes:

$$u_i = u_f \left(1 - e^{-\frac{\Delta t_p i}{\tau_{p,sph}}} \right) \qquad (231)$$

Therefore, the resulting $\sigma(\Delta t_p/\tau_p)$ value for spherical particles can be calculated in accordance with Equ.228 which yields:

$$\sigma_{n-a}\left(\Delta t_p / \tau_p\right) = u_f \sqrt{\frac{1}{i_{\max}} \sum_{i=1}^{i_{\max}} \left[\left(-1\right)^{i+1} \left(\frac{\Delta t_p}{\tau_p} - 1 \right)^i + e^{-\frac{\Delta t_p \cdot i}{\tau_p}} \right]^2} \tag{232}$$

The index *n-a* represents the comparison between the numerical and the analytical solution. To get an idea of the relative deviation, compared to the uniform fluid velocity u_f, the relative medium standard deviation can be written as:

$$\sigma_{rel,n-a}\left(\Delta t_p / \tau_p\right) = \frac{\sigma_{n-a}\left(\Delta t_p / \tau_p\right)}{u_f} = \sqrt{\frac{1}{i_{\max}} \sum_{i=1}^{i_{\max}} \left[\left(-1\right)^{i+1} \left(\frac{\Delta t_p}{\tau_p} - 1 \right)^i + e^{-\frac{\Delta t_p \cdot i}{\tau_p}} \right]^2} \tag{233}$$

5.4.3.2 Quantification of Non–Spherical, Numerical Error

For non–spherical particles the reference curve shall be created by selecting another speed up curve, based on Equ.24. Therefore a very small $\Delta t_p/\tau_p$ ratio that serves as the reference value $\Delta t_{p,0}/\tau_p$ has to be chosen. Hence, the resulting $\sigma(\Delta t_p/\tau_p)$ value for non-spherical particles is calculated like this:

$$\sigma_{n-n}\left(\frac{\Delta t_p}{\tau_p}, \frac{\Delta t_{p,0}}{\tau_p} \right) = u_f \sqrt{\frac{1}{i_{\max}} \sum_{i=1}^{i_{\max}} \left[\left(-1\right)^{(n-1)\cdot i+1} \left(\frac{\Delta t_{p,0}}{\tau_p} - 1 \right)^{ni} + \left(\frac{\Delta t_p}{\tau_p} - 1 \right)^i \right]^2} \tag{234}$$

Here the index n-n represents the comparison between the inspected numerical speed up curve and the numerical reference solution. The new variable n stands for:

$$n = \frac{\Delta t_p}{\Delta t_{p,0}} \in Z \tag{235}$$

Of course the reference-value-based, $\sigma(\Delta t_p/\tau_p)$ calculation, shown in Equ.234, can also be applied for spherical particles, where an explicit, analytical reference solution is available. In this case the σ_{n-n} value converges to σ_{n-a} as the reference value $\Delta t_{p,0}/\tau_p$ converges to zero:

$$\lim_{\frac{\Delta t_{p,0}}{\tau_p} \to 0} \sigma_{n-n}\left(\frac{\Delta t_p}{\tau_p}, \frac{\Delta t_{p,0}}{\tau_p}\right) = \sigma_{n-a}\left(\frac{\Delta t_p}{\tau_p}\right) \tag{236}$$

5.4.3.3 Evaluation of Quantified, Numerical Error

Based on the quantification procedure described above and in particular based on Equ.233 and Equ.234, extensive parameter studies have been carried out. OpenFOAM® - CFD test runs, featuring spherical and non-spherical particles have been conducted. The particles were set to speed up in a *large* flow channel with *zero* wall friction and thus uniform flow conditions. Hereby the parameter $\Delta t_p/\tau_p$ was varied, speed up curves were monitored and the $\sigma_{rel}(\Delta t_p/\tau_p)$ values were written out. In parallel, equivalent evaluations, directly based on Equ.233 and Equ.234 were conducted. The results for spherical particles are shown in Figure 78, where the two $\sigma_{rel}(\Delta t_p/\tau_p)$ curves are plotted against $\Delta t_p/\tau_p$. Especially for $\sigma_{rel} \leq 0.2$ the two curves match almost exactly.

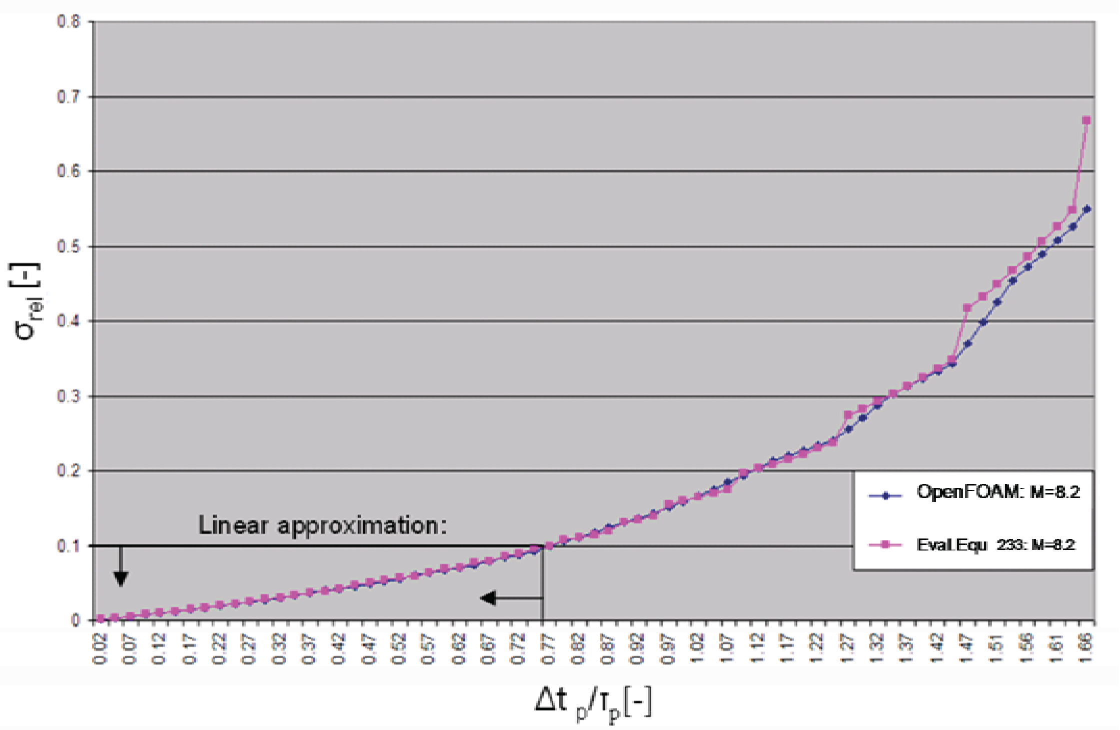


Figure 78: Plot of σ_{rel} against $\Delta t_p/\tau_p$ for a spherical particle of arbitrary size and composition which speeds up in an arbitrary fluid. Each data point is calculated by comparing the corresponding numerical speed up curve to the analytical speed up solution for spherical particles. Comparison of OpenFOAM® implementation (blue) and evaluation of Equ.233 (pink). Chosen M-value is 8.2. The equivalent procedure for an arbitrary, non-spherical particle yields the exact same result.

The very same figure can be produced for spherical and non–spherical particles, even though the non–spherical CFD calculation uses Hölzer-Sommerfeld drag instead of *Stokes'* drag, and the non–spherical, reference curve stems from Equ.234 instead of Equ.233.

A variation of the parameters: fluid velocity, dynamic fluid viscosity, volume equivalent, spherical particle diameter and particle density, confirms the derivations of chapter 5.4.3. The $\sigma_{rel}(\Delta t_p/\tau_p)$ results show absolutely no dependence on those factors and thus can be considered as universally suitable in terms of particle properties as well as fluid properties and conditions.

Considering Equ.233 and Equ.234, only three further sources of possible influence on the final result remain: the reference parameter $\Delta t_{p,0}/\tau_p$ (relevant for non–spherical particles), the parameter M that affects i_{max} over Equ.230 and the chosen $\Delta t_p/\tau_p$ range. For a discussion of those sources of influence see chapter 5.4.4.1 to 5.4.4.3.

The $\sigma_{rel}(\Delta t_p/\tau_p)$ curve shown in Figure 78 however, enables the user to choose a certain $\Delta t_p/\tau_p$ value and immediately get an estimate of the relative standard deviations of evolving, numerical particle velocities, compared to the correct result.

5.4.4 Simple, Linear Correlation for Deviation

Any serious simulation will use values of $\Delta t_p/\tau_p<0.8$ so that, according to Figure 78, the relative standard deviation to the *correct* speed up result, will be well below 0.1 (10% u_f). In that region the exponential character of the $\sigma_{rel}(\Delta t_p/\tau_p)$ curve is not yet fully developed and a linear correlation with a coefficient of determination, $R^2>0.99$ can be found. This means that a very simple, linear rule for σ_{rel} - $\Delta t_p/\tau_p$ dependence can be obtained. Since for $\Delta t_p/\tau_p=0$, also $\sigma_{rel}=0$, the linear correlation bears only *one* degree of freedom, the slope k_{rel}. Hence for $\sigma_{rel}<0.07$ (7% u_f) we find:

$$\sigma_{rel} = k_{rel}\frac{\Delta t_p}{\tau_p} \qquad (237)$$

Figure 79 can be plotted by evaluating the situation shown in Figure 78 for $\Delta t_p/\tau_p$ values that range from 0 to $\Delta t_{p,end}/\tau_p=0.55$. It shows the comparison of results yielded by OpenFOAM® and the evaluation of Equ.233. Hereby an almost exact match can be achieved. In this $\Delta t_p/\tau_p$ range a linear correlation with $R^2=0.9935$ can be drawn and the resulting slope value k_{rel} can be found to be:

$$k_{rel}= 0.1118 \qquad (238)$$

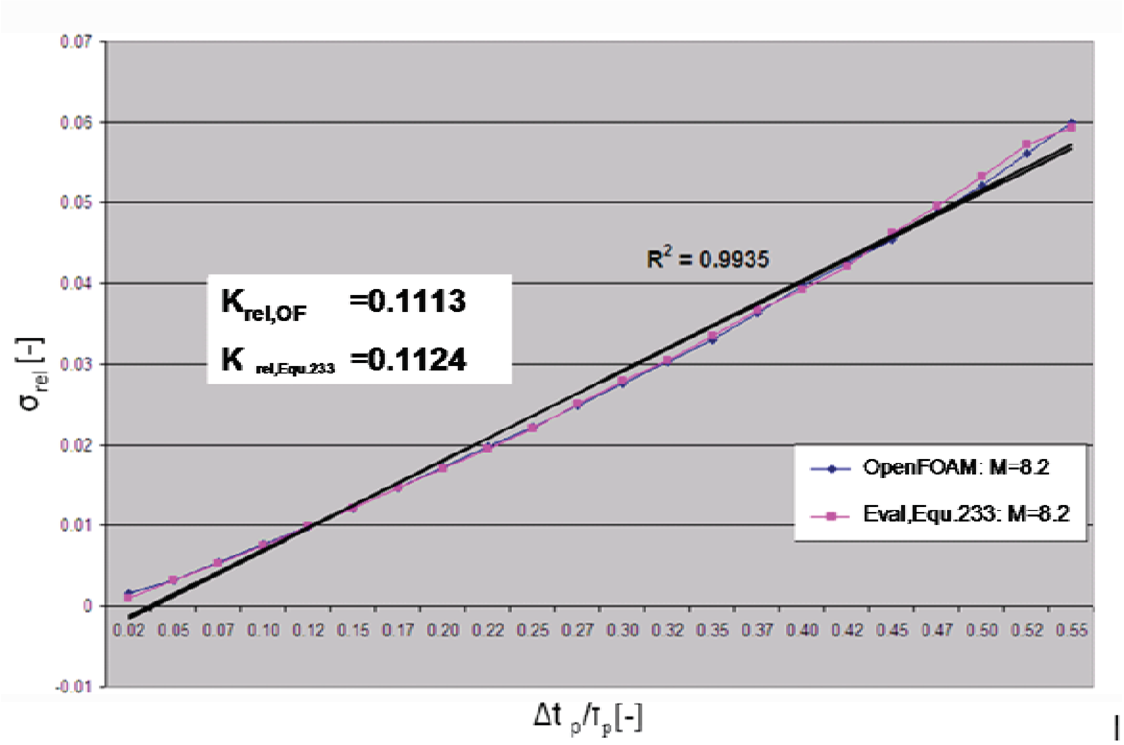


Figure 79: Plot of σ_{rel} against $\Delta t_p/\tau_p$ with $\Delta t_p/\tau_p$ ranging from 0.0 to $\Delta t_{p,end}/\tau_p$=0.55. Situation is equivalent to Figure 78. Linear correlation with coefficient of determination R^2=0.9935. Numerical speed up implemented in OpenFOAM® yields a slope of $k_{rel,OF}$=0.1113 and evaluation of Equ.39 yields slope $k_{rel,Equ.50}$=0.1124. Results are valid for spherical and non-spherical particles.

As a consequence the σ_{rel} - $\Delta t_p/\tau_p$ correlation for σ_{rel} <0.07 (7%u_f), M= 8.2 and $\Delta t_p/\tau_p$ values, ranging from 0.0 to $\Delta t_{p,end}/\tau_p$=0.55, can be written as:

$$\sigma_{rel} = 0.1118 \cdot \frac{\Delta t_p}{\tau_p} \qquad (239)$$

This holds true for any set of particle properties, for spherical and non–spherical particles and for any set of fluid properties and conditions. By inserting into Equ.239, the user can chose an appropriate $\Delta t_p/\tau_p$ value and immediately estimate its impact on overall numerical deviation to the analytical and/or reference solution, in relation to the given fluid velocity. On the other hand it is possible to choose a desired, maximum deviation $\sigma_{rel,max}$, and then to immediately estimate the maximum, allowed time step $\Delta t_{p,max}$ for any particle with particle relaxation time τ_p.

To finally decide on the universality of Equ.239, the dependence on parameters like M, the $\Delta t_p/\tau_p$ range and (for non-spherical particles) the reference parameter $\Delta t_{p,0}/\tau_p$ will have to be checked.

5.4.4.1 Slope Dependence on Reference Value, $\Delta t_{p,0}/\tau_p$

For non-spherical particles the σ_{rel} - $\Delta t_{p,0}/\tau_p$ curve can be calculated by using Equ.234 and by choosing an appropriate reference parameter $\Delta t_{p,0}/\tau_p$. Thus an additional parameter of possible result dependence is introduced. An inspection of magnitude of dependence is necessary.

By applying Equ.234 on spherical particles and by letting $\Delta t_{p,0}/\tau_p$ converge to 0.0, the result converges to that of Equ.233. Consequentially it can be concluded that, the lower the value for $\Delta t_{p,0}/\tau_p$ is chosen, the higher the quality of the result will be. To quantify this qualitative statement, a parameter study for non-spherical particles has been conducted. Therefore the parameter $\Delta t_{p,0}/\tau_p$ has been varied and for each value a full σ_{rel} - $\Delta t_p/\tau_p$ correlation, yielding k_{rel} values according to Figure 79, has been established. Using $\Delta t_{p,0}/\tau_p$=0.001 as starting point, k_{rel} values have been calculated for $\Delta t_{p,0}/\tau_p \leq 0.02$.

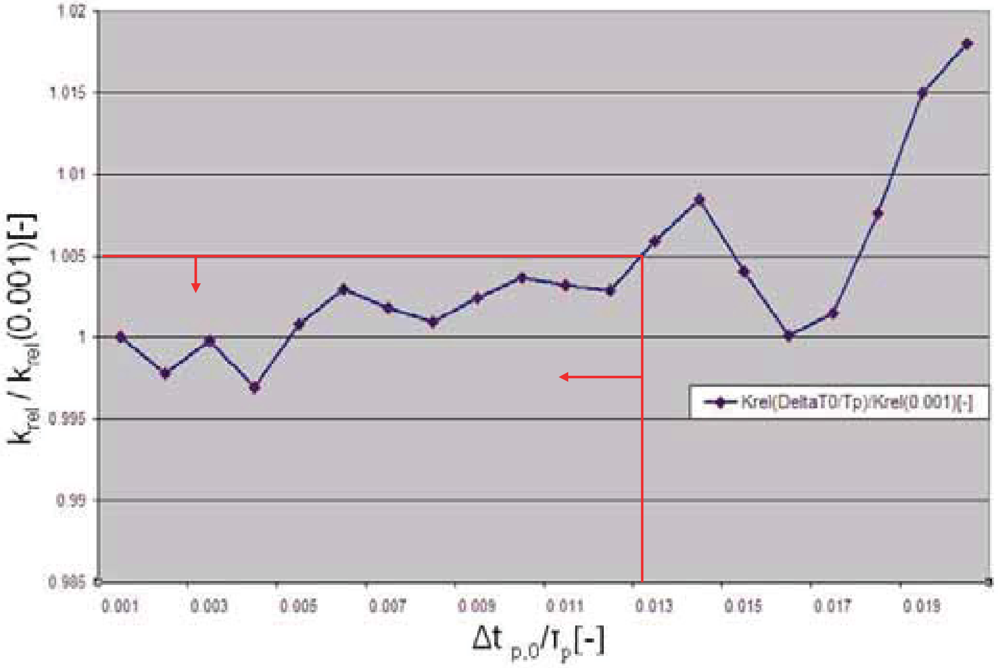


Figure 80: Ratio of $k_{rel}(\Delta t_{p,0}/\tau_p)/k_{rel}(\Delta t_{p,0}/\tau_p = 0.001)$ plotted against $\Delta t_{p,0}/\tau_p$.

The plot in Figure 80 shows that, for $\Delta t_{p,0}/\tau_p < 0.013$, the k_{rel} result deviates by only +/- 5‰ around the starting point result, which means that in this range k_{rel} can be considered to be completely independent of $\Delta t_{p,0}/\tau_p$.

5.4.4.2 Slope dependence on $M = t_{end}/\tau_p$

As seen in Figure 77, a variation of the parameter M will almost certainly lead to a change in the calculated, medium deviation between the compared curves. Not to mention the fact that the parameter $i_{max}(M)$ has a profound impact on Equ.233 and Equ.234. Qualitatively it can be stated that:

$$\lim_{M \to 0} \sigma_{rel} = 0 \qquad (240)$$

$$\lim_{M \to \infty} \sigma_{rel} = 0 \qquad (241)$$

Furthermore it is clear that σ_{rel} will show a maximum somewhere within the range $0 \leq M \leq \infty$. However, it must be noted, that the obvious σ_{rel} – M dependence does not change the numerical situation (e.g. stability) at all. It only brings about a different view point of one and the same numerical speed up curve and its analytical or reference solution.

To quantify the σ_{rel} – M dependence, and in particular the k_{rel} – M dependence, a parameter study has been conducted. Therefore the parameter M was varied and for each value a full σ_{rel} - $\Delta t_p/\tau_p$ correlation, yielding k_{rel} values according to Figure 79, was established. For each calculation of k_{rel} the $\Delta t_p/\tau_p$ value was varied between 0.0 and $\Delta t_{p,end}/\tau_p=0.15$. Figure 81 shows the resulting plot of k_{rel} against M. As expected: $k_{rel}=0.0$ for M=0 and also converges to 0.0 for $M \rightarrow \infty$. A maximum k_{rel} value $k_{rel,max}$ can be found for M=1.60. It is $k_{rel,max}=0.170$.

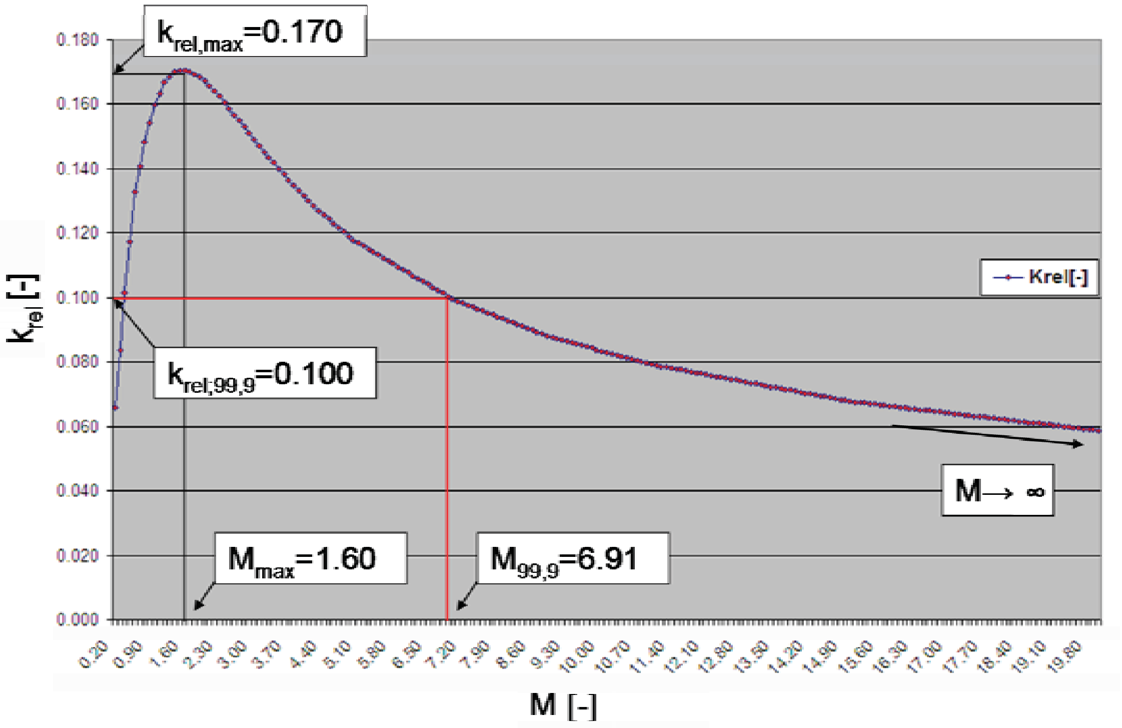


Figure 81: Plot of k_{rel} against M. Maximum k_{rel} value at M_{max}=1.60. Definition of $M_{99,9}$ value and $k_{rel;99,9}$ value.

Considering the facts stated above, a reasonable course of action in dealing with the k_{rel} – M dependence is to simply define a constant M value throughout the quantification procedure. Thus a constant, never changing frame of reference is established. A reasonably appropriate point to evaluate the behaviour of the entire speed up curve is the *M-time* $M_{99,9}$, when the accelerating particle has reached 99,9% of the fluid velocity u_f. In this case the ratio between relative fluid–particle velocity and fluid velocity is:

$$\frac{u_f - u_p}{u_f} = \frac{1}{10^p} \tag{242}$$

With, p=3.

Using the analytical speed up solution for spherical particles (Equ.209), $M_{99,9}$ can consequentially be defined as:

$$M_{99,9} = \frac{t_{end}}{\tau_p} = p \cdot \ln 10 \approx 6.91 \tag{243}$$

From Figure 12 the corresponding $k_{rel;99,9}$ value can be derived as:

$$k_{rel,99,9} = 0.100 \tag{244}$$

5.4.4.3 Slope Dependence on $\Delta t_{end}/\tau_p$

A third and final parameter with potential influence on the ultimate k_{rel} result is the $\Delta t_p/\tau_p$ range of possible relative time stepping width, or rather the upper time stepping limit $\Delta t_{p,end}/\tau_p$. While the $\Delta t_p/\tau_p$ values for the linear σ_{rel} - $\Delta t_p/\tau_p$ correlation in Figure 79 range from 0.0 to $\Delta t_{p,end}/\tau_p$=0.55, the k_{rel} – M curve in Figure 81 was calculated for $\Delta t_{p,end}/\tau_p$=0.15. A qualitative analysis of the σ_{rel} - $\Delta t_p/\tau_p$ curve in Figure 79 shows that the higher $\Delta t_{p,end}/\tau_p$, the steeper the “linear” slope k_{rel} will be. For values $\Delta t_{p,end}/\tau_p$>0.8 a linear correlation is neither appropriate nor necessary.

The basic situation is the same as for the parameters $\Delta t_{p,0}/\tau_p$ and M: a variation does not affect the numerical situation, but only the evaluation of one and the same status. Parameter studies, establishing k_{rel} – M curves (analogous to chapter 5.4.4.2) for two basic cases of $\Delta t_{p,end}/\tau_p$ have been conducted. The first case, where $\Delta t_{p,end}/\tau_p=0.15$, holds for $\sigma_{rel}\leq 0.012$ (=1,2% u_f) and the second case, where $\Delta t_{p,end}/\tau_p=0.40$ holds for $\sigma_{rel}\leq 0.04$ (=4,0% u_f). Figure 82 shows a direct comparison of the two k_{rel} – M curves.

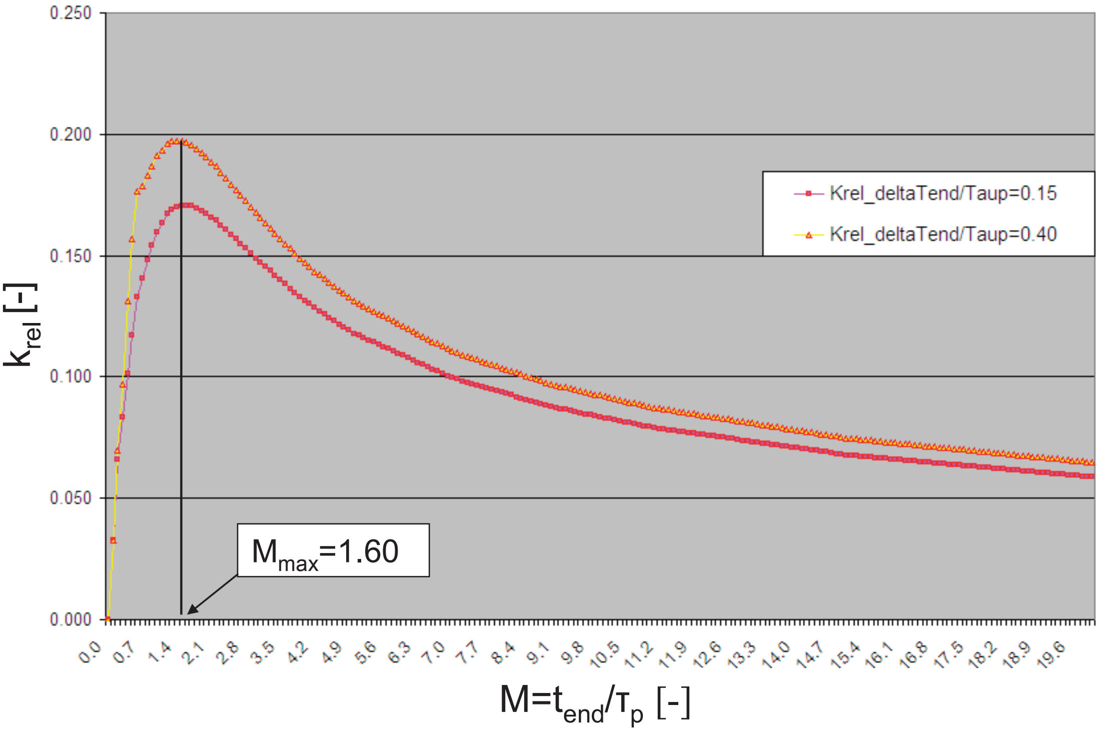


Figure 82: Plot of k_{rel} against M. Chosen parameter is $\Delta t_{p,end}/\tau_p=0.15$ (red) and $\Delta t_{p,end}/\tau_p=0.4$ (orange). Maximum of both curves lies at $M_{max}=1.60$. Difference between curves converges to 0.0 for $M\rightarrow\infty$.

As expected k_{rel} increases for increasing $\Delta t_{p,end}/\tau_p$, but the basic properties of the curve (convergence and maximum $k_{rel,max}$ at $M_{max}=1.60$) remain the same. For further applications of the quantification scheme, the $\Delta t_{p,end}/\tau_p=0.15$ curve and the $\Delta t_{p,end}/\tau_p=0.40$ curve will be chosen as alternative reference.

5.4.5 Adaptive Time Stepping of User-Defined Accuracy

Finally a simple, adaptive time stepping rule, for spherical and non-spherical particles, for any set of fluid and particle properties and for any given local flow field, can be presented.

For $M_{99,9}$=6.91 and with $\Delta t_{p,end}/\tau_p$=0.15 [σ_{rel}<0.012 (<1,2%u_f)] the k_{rel} value can be determined out of Figure 82 as:

$$k_{rel;M99,9}^{0.15} = 0.100 \tag{245}$$

So the linear σ_{rel} relation, using Equ.237 reads:

$$\sigma_{rel,M99,9}^{\max 0.012} = 0.100 \cdot \frac{\Delta t_p}{\tau_p} \tag{246}$$

For $M_{99,9}$=6.91 and with $\Delta t_{p,end}/\tau_p$ =0.40 [σ_{rel}<0.040 (<4%u_f)] the k_{rel} value can be determined out of Figure 82 as:

$$k_{rel;M99,9}^{0.40} = 0.112 \tag{247}$$

So the linear σ_{rel} relation, according to Equ.237 reads:

$$\sigma_{rel,M99,9}^{\max 0.04} = 0.112 \cdot \frac{\Delta t_p}{\tau_p} \tag{248}$$

The user can select a desired, medium standard deviation $\sigma_{rel,UD}$<0.040 (<4%u_f). Then the appropriate number of particle sub time steps J, which is specifically adapted to the particle as well as the local fluid properties and conditions, can be calculated by use of Equ.203.

For $\sigma_{rel,UD} \leq 1.2\%u_f$:

$$J = \left\lceil 0.100 \cdot \frac{\Delta t_f}{\sigma_{rel,UD} \cdot \tau_p \left(D_{sph}, \rho_p, u_f, \mu_f\right)} \right\rceil \tag{249}$$

For $1.2\%u_f < \sigma_{rel,UD} \leq 4.0\%u_f$:

$$J = \left\lceil 0.112 \cdot \frac{\Delta t_f}{\sigma_{rel,UD} \cdot \tau_p \left(D_{sph}, \rho_p, u_f, \mu_f\right)} \right\rceil \tag{250}$$

5.5 Adaptive Time Stepping and Event Forces

Up until this point, the discussion regarding necessary time step adaptation has only considered single, suspended particles in an unbounded flow domain. So far, event forces due to particle interactions have been neglected. Those additional forces create a more complex situation: relatively high, momentary particle acceleration in combination with static, or *particle fluid adaptive time stepping* (see 5.4) lead to an underestimation of fluid damping effects on the motion. Once again particles tend to unrealistically shoot out of the calculation domain.

In order to get a hold of these phenomena, a time stepping scheme, introducing the newly defined *particle event force relaxation time,* was developed.

5.5.1 The Particle-Event-Force Relaxation Time

For better distinction, the traditional, steady-state particle relaxation time shall hereby be denoted as $\tau_{p,0}$. As noted before, $\tau_{p,0}$ represents the ratio between particle inertia and viscous fluid forces, postulating quasi-steady state, particle-fluid drag. The effect of local event forces however, causes strongly unsteady particle behaviour. Therefore a new, particle event force oriented, non-steady relaxation time, the *particle event force relaxation time* $\tau_{p,EF}$ has been defined.

With the introduction of $\tau_{p,EF}$, the particle velocity change during Δt_p, under the influence of fluid drag and any acting event force F_i can be written as:

$$\frac{\Delta u_{p,j}}{\Delta t_p} = \frac{F_{i,j}}{m_p} + \frac{\left(u_{f,j} - u_{p,j}\right)}{\tau_{p,0}} = \frac{1}{\tau_{p,EF,j}} u_{f,j} - \frac{1}{\tau_{p,0}} u_{p,j} \qquad (251)$$

Here the index j denotes the coordinate components x, y, z. Thus $\tau_{p,EF,j}$ varies with varying force effects in the three spatial dimensions. To be on the safe side the smallest $\tau_{p,EF,j}$ component is chosen to introduce the *particle event force relaxation time* as follows:

$$\tau_{p,EF} = \min\left(\frac{1}{\left(\frac{F_{i,j}}{m_p \cdot u_{f,i}} + \frac{1}{\tau_{p,0}} \right)} \right) \qquad (252)$$

Equ.251 represents the extension of Equ.216. Whereas the previous formulation merely considers fluid-particle drag, the new form contains all possible acting forces.

The quantifiable, adaptive time stepping scheme of user-defined accuracy, presented in chapter 5.4, loses its general validity as soon as $F_{i,j} > F_d$. However, all in all, event forces turn out not to be as dangerous for the numerical integrity of the entire calculation as underestimated drag forces can be. The reason for this is that drag forces act continuously, whereas event forces, even though at times considerably higher, only occur momentarily.

Using the expression in Equ.252, two *event-force-adapted* time stepping schemes are proposed here:

- The spatially bounded scheme
- The temporally bounded scheme

5.5.1.1 The Spatially Bounded, Event-Force-Adapted Time Stepping Scheme

The spatially bounded, *event-force-adapted* time stepping scheme ensures that, during Δt$_p$, the event-force-accelerated particle proceeds only over a fixed fraction K of the local grid spacing Δx$_s$, so that:

$$\frac{\overline{u_p} \cdot \Delta t_p}{\Delta x_s} \overset{!}{=} \frac{1}{K} \tag{253}$$

Here the velocity development during Δt$_p$ shall be approximated with 1st order accuracy (linear development). Therefore the mean particle velocity $\overline{u_p}$ is:

$$\overline{u_p} = u_p + \frac{1}{2} \cdot \Delta u_p \tag{254}$$

In combination with Equ.251 and Equ.254, Equ.253 then yields:

$$\frac{\left(u_p + \frac{1}{2} \cdot \Delta u_p \right) \cdot \Delta t_p}{\Delta x_s} = \frac{\left(u_p + \frac{1}{2} \cdot \left(\frac{1}{\tau_{p,EF}} u_f - \frac{1}{\tau_{p,0}} u_p \right) \cdot \Delta t_p \right) \cdot \Delta t_p}{\Delta x_s} \overset{!}{=} \frac{1}{K} \tag{255}$$

Hence the spatially bounded, event-force adapted time step can be calculated as:

$$\Delta t_p = \frac{-u_{p,0} + \sqrt{u_{p,0}^2 + 2 \cdot \frac{\Delta x_s}{K} \left(\frac{1}{\tau_{p,EF}} u_f - \frac{1}{\tau_{p,0}} u_p \right)}}{\frac{1}{\tau_{p,EF}} u_f - \frac{1}{\tau_{p,0}} u_p} \tag{256}$$

Via a reduction of $\tau_{p,EF}$, the force effect leads to shorter time steps. The shorter the given particle path during each iteration, the smaller Δt_p will be. Note that varying time steps must not reduce the overall change of momentum due to the acting forces. This means that the shorter the time steps become, the more iterations will have to pass with F_i fully acting on the particle. This fact has to be taken into account for the spatially and the temporally bounded time stepping scheme alike.

5.5.1.2 The Temporally Bounded, Event-Force-Adapted Time Stepping Scheme

Compared to the spatially bounded scheme, the temporally bounded scheme is comparatively simple, once $\tau_{p,EF}$ is calculated via Equ.252. The spatially bounded scheme causes the particle to progress for equal amounts of distance during each time step. In contrast to that, the temporally bounded scheme calculates each time step as a fraction K of $\tau_{p,EF}$:

$$\Delta t_p = \frac{\tau_{p,EF}}{K} \tag{257}$$

This means that, the higher F_i is, the smaller Δt_p will be. In direct comparison to the spatial scheme this method turned out to be faster, simpler and just as accurate. Therefore it has been used for further work and will be referred to as *the* event-force-adapted time stepping scheme.

5.5.1.3 Event-Force-Adapted Time Stepping Scheme versus static time stepping

A thorough analysis concerning the impacts of the utilization of the event force adapted time stepping as compared to ordinary, static time stepping has been conducted. Thereby it has been assumed that a large, spherical particle with D_{sph}=0.001m, ρ_p=2800 kg/m³ speeds up under the influence of a viscous fluid stream with varying μ_f and u_f=0.4m/s. Local fluid grid spacing is $\Delta s=10^{-4}$m. Right at take-off the particle gets hit by the event force F_e, which acts during static particle sub-time step $\Delta t_{p,static}$=7,78*10^{-3}s and causes a momentum input of $\Delta I=F_e*\Delta t_{p,static}$. The speed-up and fluid-dampening effects have been investigated for both time stepping schemes. Figure 83 shows a comparison of the two resulting velocity curves. A definite difference in resulting, maximum velocity $u_{p,max}$ (which occurs directly after momentum input) can be observed. The adaptive time stepping scheme gives a much higher resolution of the interval during which F_e acts. Consequentially the fluid-damping effect is calculated more accurately.

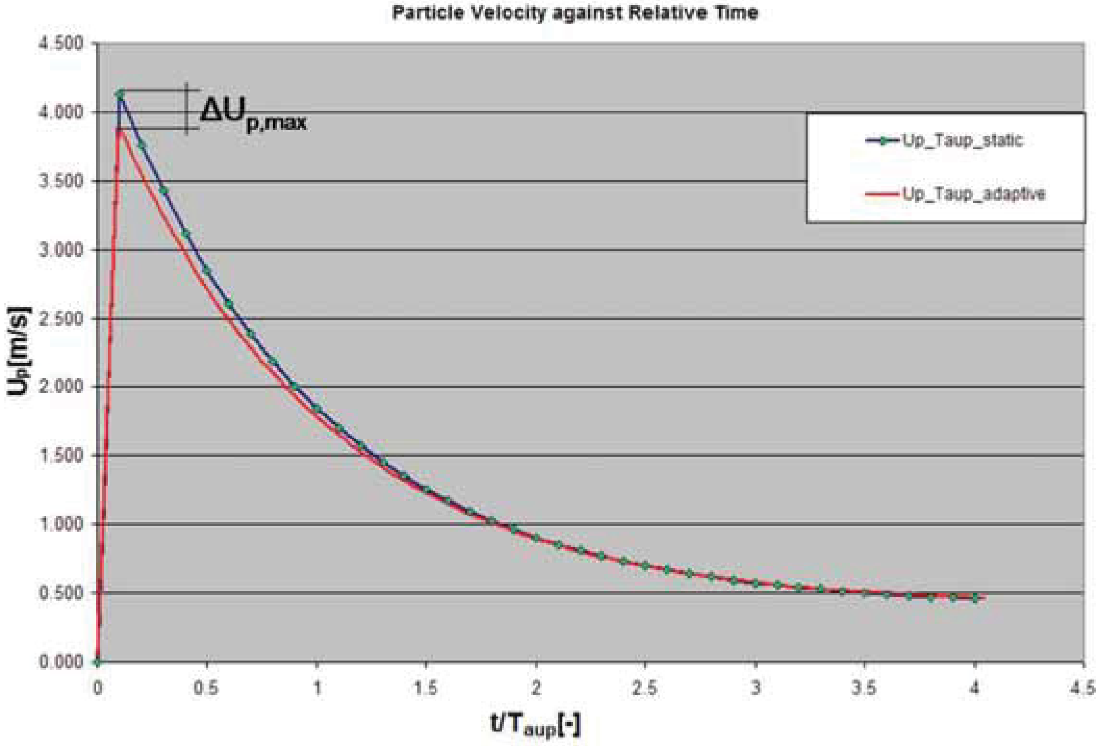


Figure 83: Comparison of exemplary, spherical particle velocity curves calculated for same physical conditions with static and event-force-adapted time stepping scheme. Conditions: $\tau_{p,0}$=0.0078s, μ_f=0.002Pas, ΔI=6*10^{-6}.kgm/s.

If the case shown in Figure 83 is altered by varying, the fluid viscosity then the static, spherical relaxation time $\tau_{p,0}$, changes along with it. In addition to that $\Delta u_{p,max}$ does not remain constant either, so that Figure 84a and Figure 84b can be produced. The two Figures present plots of $u_{p,max}$ and $\Delta u_{p,max}$ /$u_{p,max,static}$ respectively. They clearly show that, the smaller $\tau_{p,0}$ becomes the more necessary the *adaptive time scheme* will be. The reason for that is obvious:

Small $\tau_{p,0}$ means a smaller inertia-to-viscous-force ratio. Static time stepping means neglecting the fluid viscous force influence. Neglecting the viscous damping effect where viscous forces dominate, proves fatal. On the other hand the results make clear that for $\tau_{p,0} \rightarrow \infty$ the two curves converge to 1.0 and the relative deviation tends to *zero*.

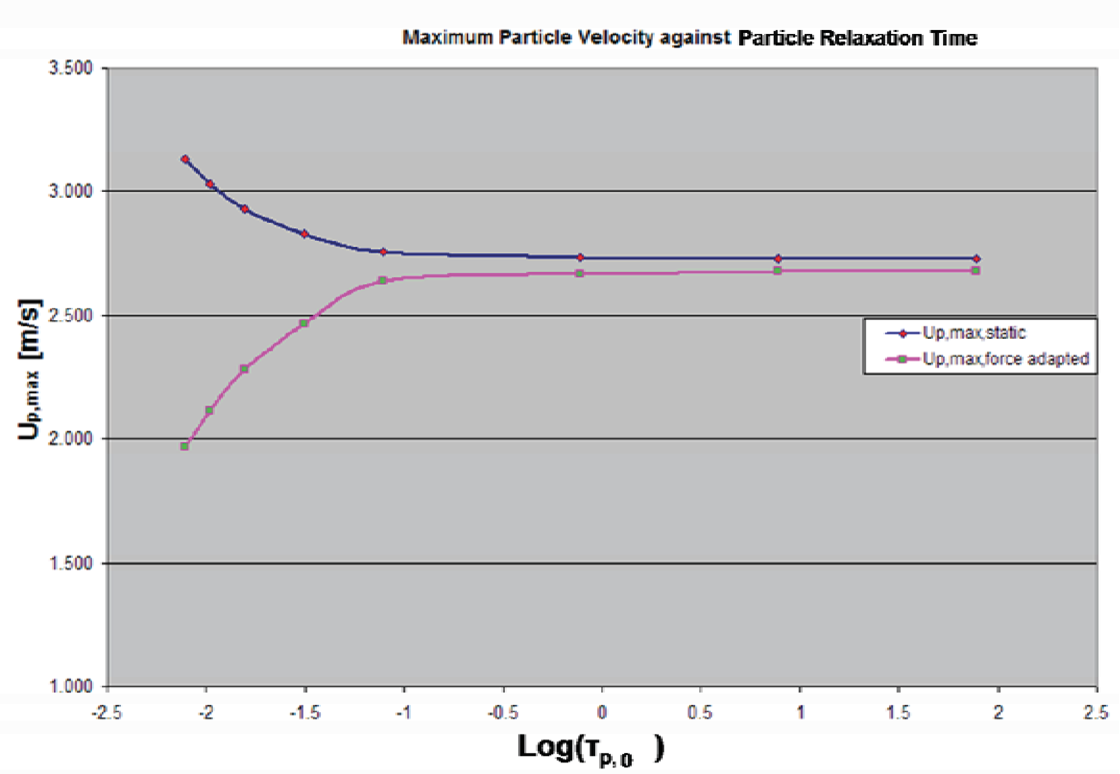


Figure 84a: Plot of $u_{p,max}$ against $\log(\tau_{p,0})$ for the case of $\Delta I = 6*10^{-6}$ kgm/s. Comparison of static time stepping scheme (blue) and adapted time stepping scheme (pink). At smaller $\tau_{p,0}$ the static scheme overestimates particle velocity development more dramatically.

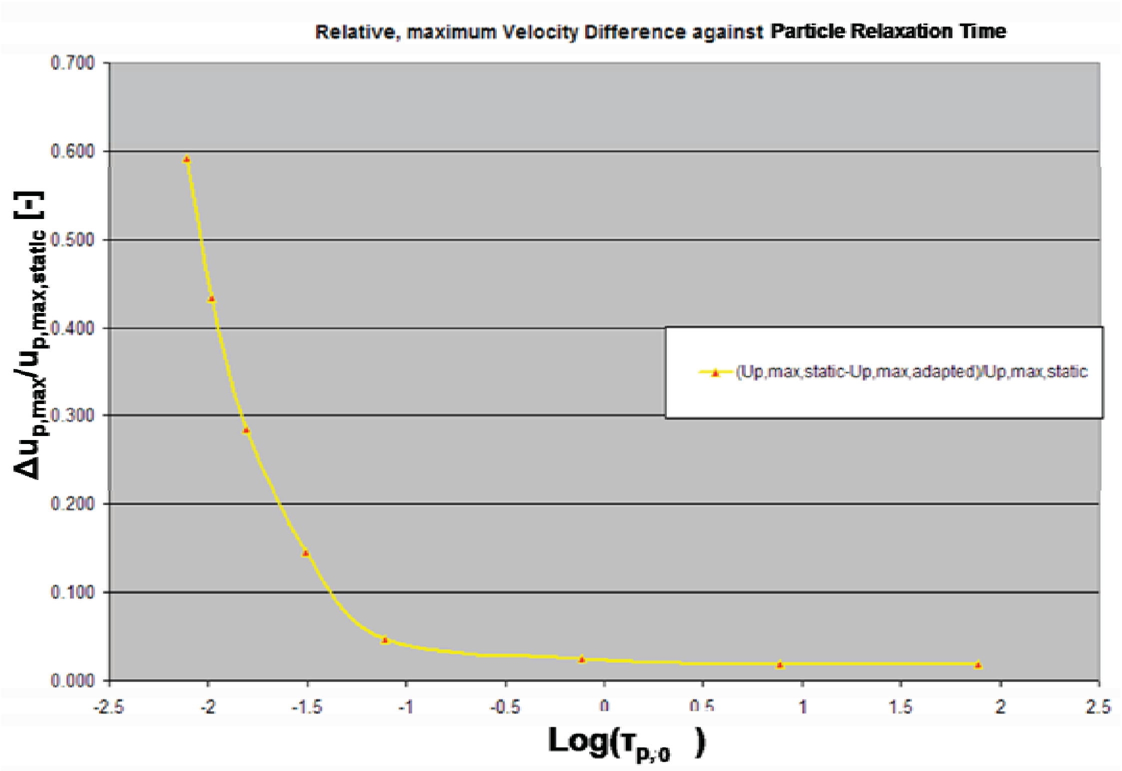


Figure 84b: Plot of $\Delta u_{p,max}/u_{p,max,static}$ (being the relative difference between the two $u_{p,max}$ curves shown in Figure 84a) against log($\tau_{p,0}$) for the case of $\Delta I=6*10^{-6}$ kgm/s. At smaller $\tau_{p,0}$ the deviation between the two results explodes, whereas for $\tau_{p,0}\rightarrow\infty$ the difference tends to *zero*.

Let us now consider the same case, but for fixed μ_f and $\tau_{p,0}=7.78*10^{-2}$ s ($\Delta t_{p,static}/\tau_{p,0}$ = 1/10). This time the momentum input shall be varied so that: $2.0*10^{-6}$ kgm/s$\leq\Delta I\leq 2.0*10^{-5}$kgm/s. If no fluid effects were present at all, this would amount to maximal, unhindered particle velocities between: 1.36m/s$\leq u_{p,max,unhindered}$ $\leq$13.64m/s. The results of $u_{p,max}$ and $\Delta u_{p,max}/u_{p,max,adaptive}$ against $u_{p,max,unhindered}$ are shown in Figure 85 and Figure 86 respectively. Even though the absolute result deviation increases for higher ΔI, the relative difference declines until it reaches a steady, asymptotic value. This is due to the following facts:

- The fluid has a decelerating impact on the particle as soon as $u_p>u_f=0.4$m/s.
- The viscous fluid damping effect is proportional to: $(u_f-u_p)/\tau_{p,0}$.

- During F_e impact ($0 \leq t < \Delta t_{p,static}$), u_p increases linearly with each adaptive time step $\Delta t_{p,adaptive}$. Thus the mean, fluid damping effect during each time step of force impact phase in the adaptive scheme is proportional to $(u_f - 1/2 * u_{p,max})/\tau_{p,0}$.
- The chosen amount of F_e, which acts per time step e.g. during $\Delta t_{p,static}$ relates to the maximum, unhindered outcome velocity as:

$$F_e = \frac{m_p \cdot u_{p,\max,\, unhindered}}{\Delta t_{p,static}} \tag{258}$$

- This is why:

$$u_{p,\max,\, adaptive} = \frac{u_{p,\max,\, unhindered} + \frac{\Delta t_{p,static}}{\tau_{p,0}} \cdot u_f}{1 + \frac{\Delta t_{p,static}}{2 \cdot \tau_{p,0}}} \tag{259}$$

- $u_{p,max,static}$ is in essence equal to $u_{p,max,unhindered}$ with the difference that during $\Delta t_{p,static}$ the fluid still has an accelerating effect on the particle because u_p is calculated as *zero* as long as $\Delta t_{p,static}$ lasts. So $u_{p,max,static}$ is actually a bit larger than $u_{p,max,unhindered}$ (see also Figure 85).

- The relative difference between static and adaptive solution can be described as:

$$\frac{\Delta u_{p,\max}}{u_{p,\max,\, adaptive}} = \frac{\frac{\Delta t_{p,static}}{\tau_{p,0}} \cdot \left(\frac{u_{p,\max,\, unhindered}}{2} - u_f \right)}{u_{p,\max,\, unhindered} + \frac{\Delta t_{p,static}}{\tau_{p,0}} \cdot u_f} \tag{260}$$

- For $u_{p,max,unhindered} \rightarrow \infty$ this expression converges to a constant:

$$\lim_{u_{p,\max,\,unhindered}\to\infty} \frac{\Delta u_{p,\max}}{u_{p,\max,\,adaptive}} = \frac{\Delta t_{p,static}}{2\cdot\tau_{p,0}} \tag{261}$$

This result states that, the larger the chosen static time step compared to the particle relaxation time, the larger the relative difference between static and adaptive time stepping scheme will be, as ΔI increases.

- For the given example $\Delta t_{p,static}/\tau_{p,0}$ = 1/10, thus $\Delta u_{p,max}/u_{p,max,adaptive}$ converges to: 0.05 just as seen in Figure 86.

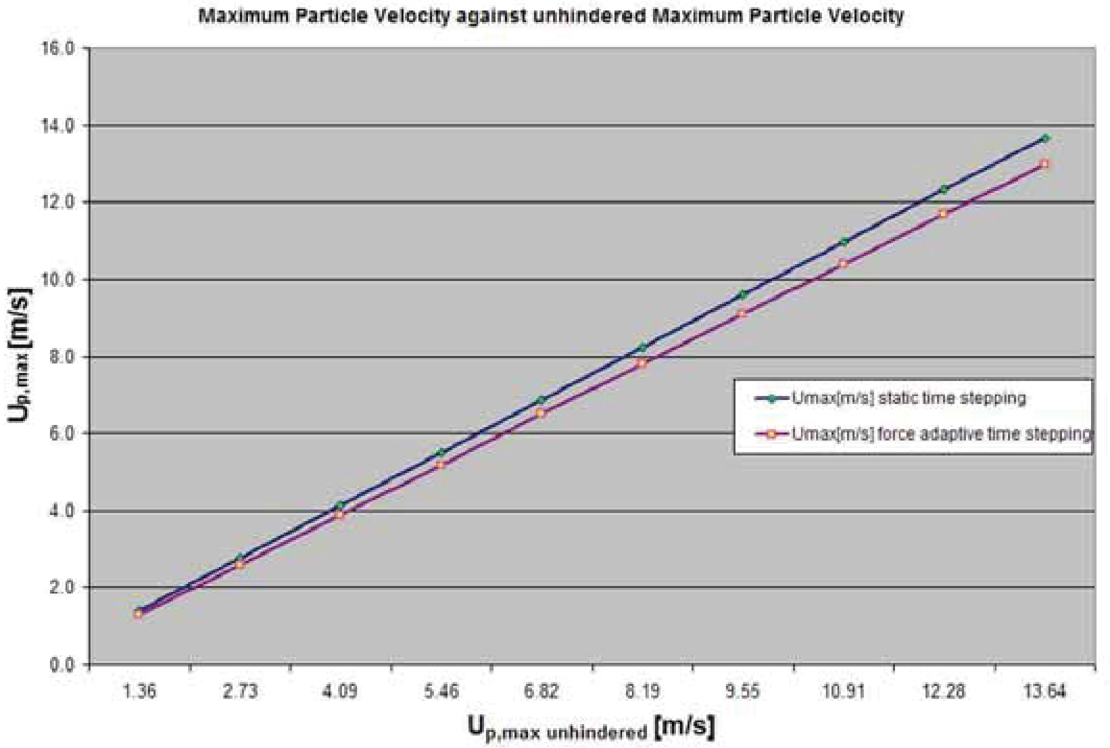


Figure 85: Plot of $u_{p,max}$ against $u_{p,max,unhindered}$ for the case of $\tau_{p,0}$=7.78*10^{-2} s. Comparison of static time stepping scheme (blue) and adaptive time stepping scheme (purple). The behaviour of both solutions is strictly linear and the absolute difference increases for higher $u_{p,max,unhindered}$ (higher momentum input).

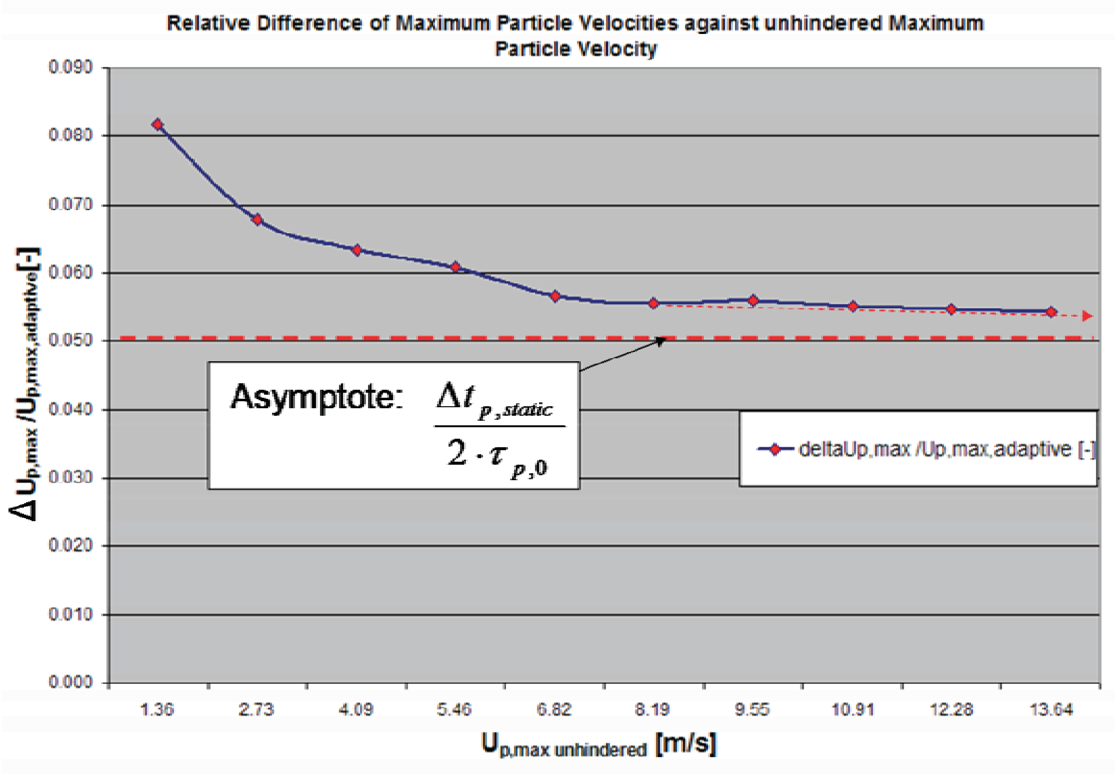


Figure 86: Plot of $\Delta u_{p,max}/u_{p,max,adaptive}$ (being the relative difference between the two $u_{p,max}$ curves shown in Figure 85) against $u_{p,max,unhindered}$ for the case of $\tau_{p,0}$=7.78*10^{-2}s and $\Delta t_{p,static}$=7.78*10^{-3}s. The relative deviation converges against $\Delta t_{p,static}/2*\tau_{p,0}$ =0.05 according to Equ.261.

5.6 Adaptive Time Stepping: Conclusion

Due to the specific programming structure of the code, both particle solvers use an explicit Euler discretization scheme to handle the particle momentum equation. The major draw back of this choice is that numerical instabilities occur more readily than with other discretization schemes such as the Runga Kutta method. In this chapter of the thesis the case of a (non-) spherical particle speeding up in an otherwise uniform, laminar flow field was chosen to describe, study and finally eliminate the encountered numerical instabilities.

The speed up behaviour of spherical and especially non–spherical particles was inspected in detail, and the necessity to consider particle shape deviations from spherical shape was pointed out again.

By identifying the parameter $\Delta t_p/\tau_p$ as single most decisive factor for the occurrence of instabilities, the complexity of the problem was dramatically reduced. Particle and fluid properties, as well as fluid conditions can be expressed by τ_p.

In addition to that a descriptive formulation for the instabilities was found, which accurately describes the problem.

A method to quantify the numerical stability of each speed up run was set up by comparing numerically calculated speed up curves to analytically obtained ones. By producing plots of relative, medium standard velocity deviations, against $\Delta t_p/\tau_p$, a simple, linear dependence for low $\Delta t_p/\tau_p$ values was encountered. Thus, by carefully eliminating any possible parameters of influence on the final result, a simple, linear σ_{rel} - $\Delta t_p/\tau_p$ relation could be defined which holds for any set of fluid and particle properties, as well as fluid conditions. This relation enables the user to choose a measure of accuracy (in terms of σ_{rel}) for the simulation run. Out of this choice, the appropriate particle sub time step (the number of particle Subcycles per fluid time step) for each, individual particle, immersed in any local fluid field can be calculated. An adaptive particle time stepping scheme to eliminate instabilities due to explicit Euler implementation could thus be presented. In extension of this original, drag force based time stepping scheme an event-force based scheme was introduced as well. It considers not only drag but all instantaneous forces on the particle and ensures either spatially or temporally bounded proceeding of the particle.

6 Extension Modules: Electro-Static Module and Bacteria Module

One major advantage of the particle code is that it can easily be extended by sub-modules, in order to extend its overall ability. Any sub-module can be switched on or off on a user-defined basis. Up until now, two larger, additional modules have been created for practical application. They shall hereby be introduced in brief:

- The Bacteria Module
- The Electro Static (E-Static) Module

6.1 The Bacteria Module

The bacteria module has been designed to get a qualitative impression of the settling, deposition and distribution behaviour of Epiterial bacteria [74] in porous media. The real-life bacteria are reported to be about spherical as long as they are immersed in the carrier fluid. As soon as they touch the surface of solid objects however, they change their shape by deforming perpendicular to the fibre surface normal n_w. The spheroid bacteria are thus transformed to small plates which stick to the surface. The deposition situation is sketched out in Figure 87. To achieve a plausible simulation of the hydrodynamically governed bacteria deposition, settling and deformation process, the (non-)spherical particle solver had to be slightly extended:

At particle-fibre impact, the particle coordinate system is rotated such that $e_y=n_w$. Then the length of half-axe b_0 is altered by the user-defined factor k_t so that $b_1=k_t*b_0$ with $0< k_t \leq 1.0$. The new half axes length a_1 and c_1 are set to be $a_1=c_1$ and are calculated via mass conservation as:

$$a_1 = c_1 = \sqrt{\frac{a_0 c_0}{k_t}} \tag{262}$$

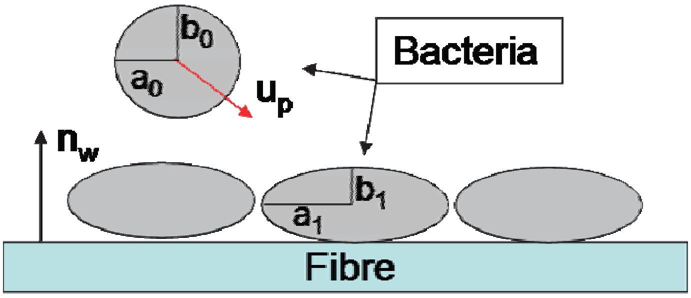


Figure 87: Sketch of the deposition and deformation situation of Epiterial bacteria on solid fibre surface.

Figure 88 shows a simplified fibre geometry with a bacteria-laden fluid passing by. As the spherical bacteria impact on the fibre, they change from being balls towards being small plates, which are usually immobilized at the position of impact. This simple benchmark case already gives an impression of the solver's capability to provide an idea about the deposition and distribution behaviour of bacteria in any geometry of fibrous or porous media. Accordingly Figure 89 shows a screenshot of a more realistic benchmark case where the bacterial deposition has been simulated within a realistically reconstructed micro-fibre geometry.

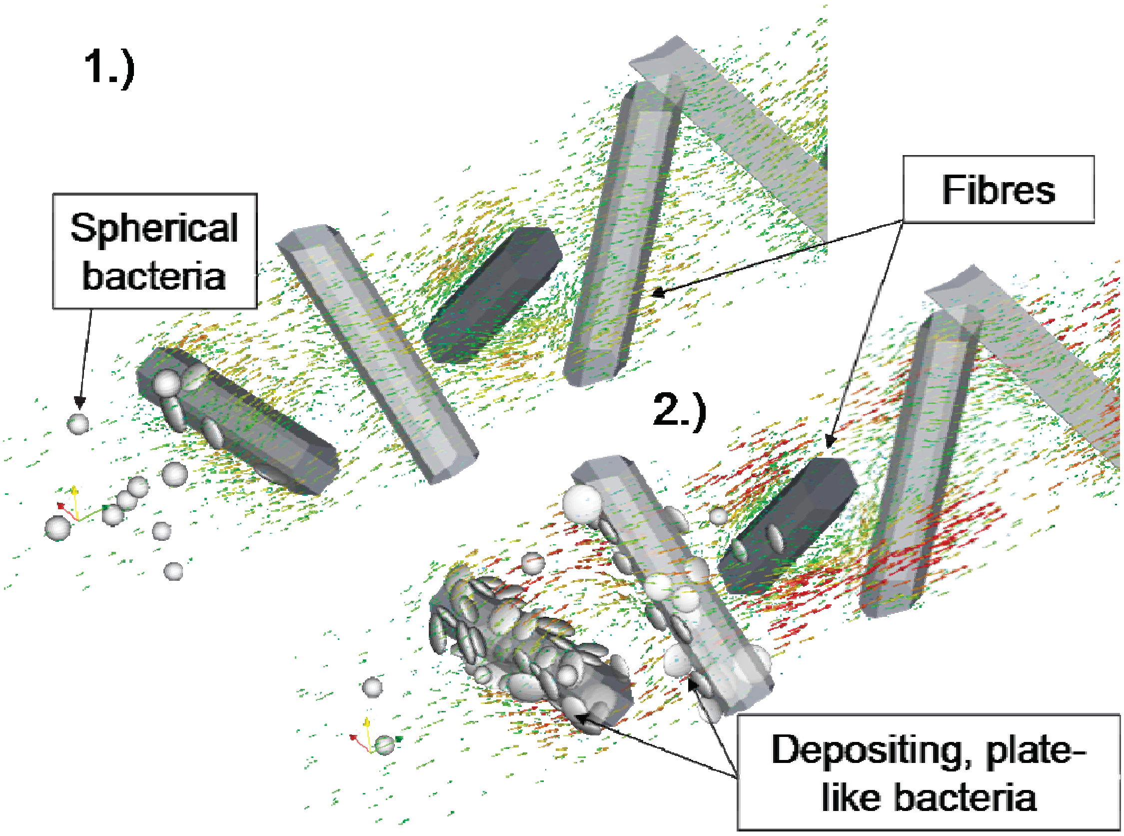


Figure 88: Screenshot of simulation run in simplified fibre geometry. Spherical bacteria, immersed in a watery fluid come in (1.), hit the fibres, deform and settle there (2.).

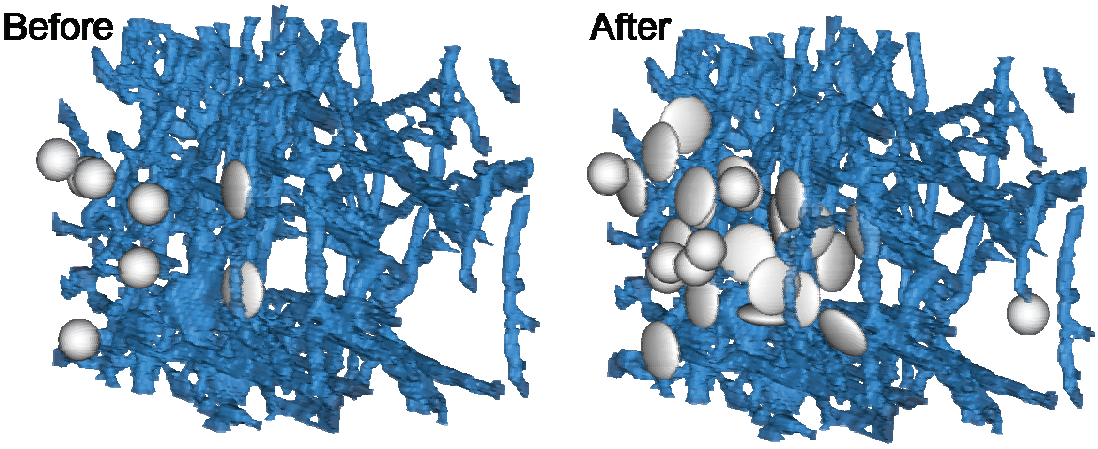


Figure 89: Screenshot of simulation run in realistic, microscopic fibre geometry. Spherical bacteria, immersed in a watery fluid come in (before) hit the fibres, deform and settle there (after).

6.2 The E-Static Module

The E-Static module has been designed to consider the influence of electro-static effects on particle motion and on filter fibre deposition behaviour. This model extension is supposed to move the filtration solver away from being a mere fluid-filtration-tool towards being applicable for air-filtration (where electrical effects are reported to be significant) simulation as well.

The E-Static module is essentially a simplified version of a full Maxwell equation [75] solver. Its physical framework has been derived by reducing the Maxwell equations to their stationary form [76]. In this case electro-magnetism can be neglected, thus Ampere's law and Gauss's law for magnetic fields become irrelevant. The remaining, governing equations are simplified formulations of Faraday's law of induction and Gauss's law for electric fields.

The generalized form of Faraday's law of induction gives the connection between electric field strength E and temporal flux of a local magnetic field B and reads [77]:

$$\oint_C \vec{E} \cdot d\vec{s} = -\frac{d}{dt} \oiint_A \vec{B} \cdot \vec{n}_A \cdot dA \qquad (263)$$

Here *s* denotes the path along a closed curve *C* which encloses the area *A* with local normal vector n_A. The integral form of the corresponding static version of Faraday's law is:

$$\oint_C \vec{E} \cdot d\vec{s} = 0 \qquad (264)$$

Its differential form can be obtained by applying Stokes' theorem [78] so that:

$$\nabla \times \vec{E} = \vec{0} \qquad (265)$$

Due to Equ.265 the electrical field can be expressed via a scalar potential field Ψ.

$$\vec{E} = -\nabla\Psi \tag{266}$$

Furthermore Gauss's law concerning electric fields states that the electric flux through any closed surface A enclosing the volume V, is directly proportional to the total amount of enclosed, electric charge $Q_{enclosed}$ [75]. The law can be written as:

$$\oint\!\!\!\oint_A \vec{E} \cdot \vec{n} \cdot dA = \iiint_V \frac{\rho}{\varepsilon} \cdot dV = \frac{Q_{enclosed}}{\varepsilon} \tag{267}$$

Where ρ is the specific, spatial charge and ε is the electric permittivity-constant of the medium. The corresponding, differential form can be obtained by applying Gauss' theorem [80] so that:

$$\nabla\vec{E} = \frac{\rho}{\varepsilon} \tag{268}$$

Inserting Equ.266 into Equ.268, a *Poisson equation*, is reached. It describes the distribution of the potential field within the vicinity of the solution domain:

$$(\nabla\nabla) \cdot \Psi = \Delta\Psi = -\frac{\rho}{\varepsilon} \tag{269}$$

The E-Static module allows the user to position positive or negative, specific charges ρ on arbitrary positions within the domain (e.g. static charges on fibres and moving charges on particles) and calculates the corresponding potential field according to the numerically implemented version of Equ.269 for each time step.

Then the vectorial, electric field is calculated via Equ.266 and the specific, electric force $F_{electric}$, which acts on each particle, can be obtained [79]:

$$\vec{F}_{electric} = Q_p \cdot \vec{E} \tag{270}$$

Here Q_p is the total, electrical charge of the particle.

Figure 90 shows a qualitative example of solver functionality. It is a screenshot of a benchmark case, where two oppositely charged particles have been inserted with *zero* initial velocity into an otherwise unbounded fluid domain. The Figure shows the distribution of the Ψ-field, its iso-lines, the vector plot of the resulting E-field and the initiated movement due to electrical attraction of the two particles.

A second example is shown in Figure 91, where two positively charged particles are placed within the vicinity of a simple, negatively charged fibre. The particles are clearly attracted by the fibre and stick to it. Development of the Ψ-field and of the E-field is shown too.

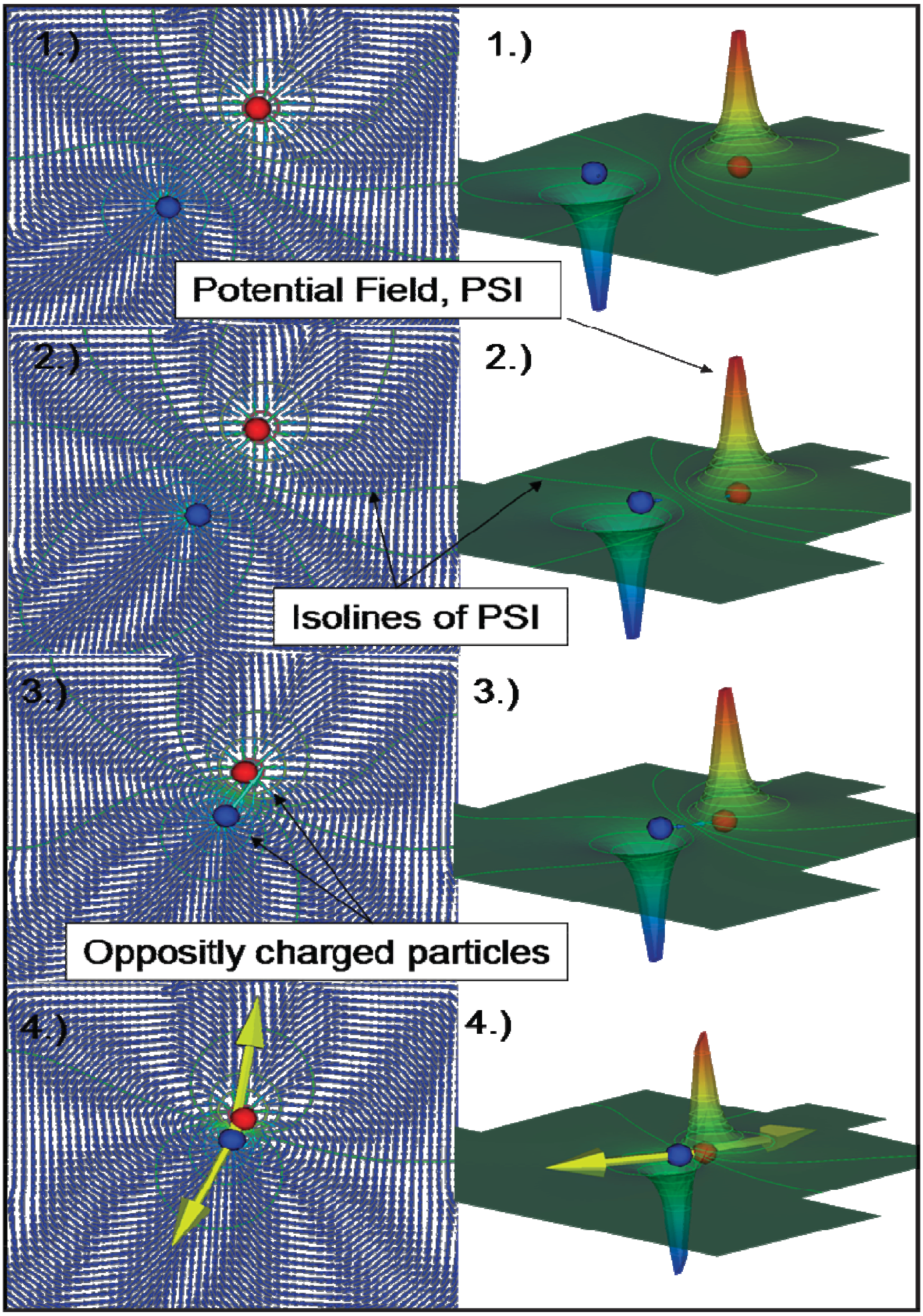


Figure 90: Four consequential screenshots from qualitative benchmark case, including two oppositely charged particles in an otherwise unbounded fluid domain. View from above (left) shows the vectorial E-field as well as the iso-lines from the Ψ-field. View from the side (right) shows the 3D, Ψ-field, dominated by particle charges.

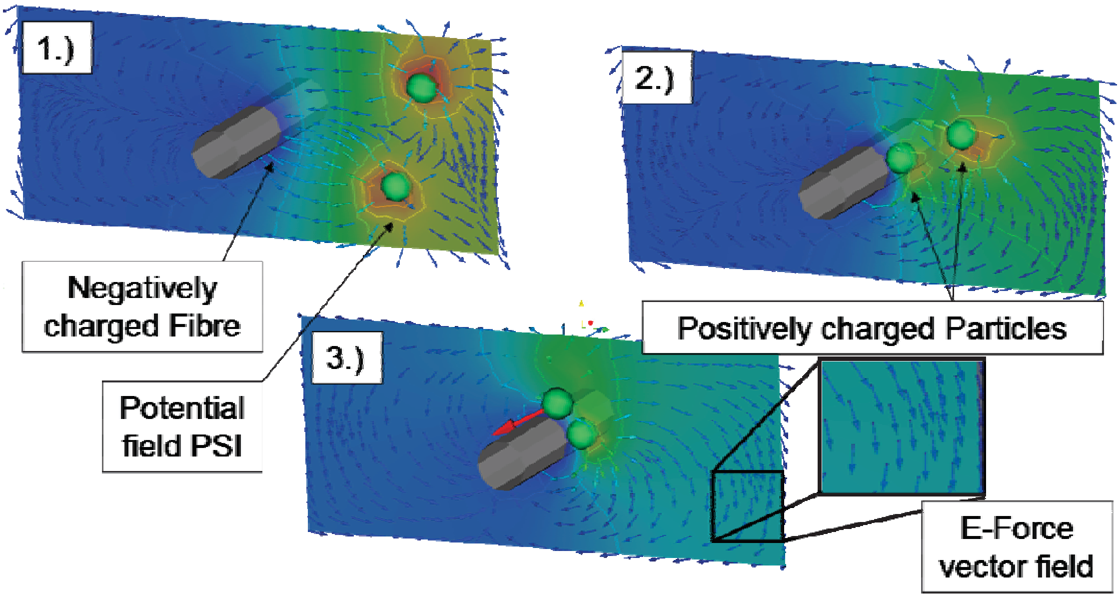


Figure 91: Three consequential screenshots from qualitative benchmark case including two positively charged particles and one negatively charged fibre. Background coloration insinuates Ψ-field and the vector field gives the resulting, E-field.

7 Workflow, C++ Program Structure and How to Use the Solver

This chapter gives an overview of the entire workflow from computer-tomographically scanning fibre samples to examining the final results of the CFD calculation. In addition to that, the workflow process behind the particle simulation is resolved in more detail. Thereafter the underlying C++ program structure is depicted as an inheritance diagram and the functionalities of essential program entities are listed. Finally all particle-code specific input dictionary parameters, which are crucial for the user to apply the solver, are laid out.

7.1 Overall Workflow

Figure 92 gives an overview of the entire workflow behind the filter fibre reconstruction and simulation project. The whole procedure comprises three main phases:

- Meshing and Pre-processing: CT-scan data is gathered from real-life fibre samples. A Digital Fibre Reconstruction utility digitalizes the CT-scan data, reconstructs a 3D image of the fibre structure and yields a structured grid mesh, suitable for OpenFOAM®. Then the user defines the physical starting and boundary conditions within the OpenFOAM® dictionaries.

- Processing: The flow field is calculated either in combination with the fibre-deformation phenomena (based on the FSI solver) and/or in combination with depositing dirt particles (based on the hereby presented particle solver). At the end of each time step the results are streamed out as text files.

- Post-processing: If necessary, the text file based data is processed by self-programmed Python® utilities [28] in order to extract information, such as filter fibre efficiency, particle penetration depth or kinetic particle energy. Compact, numerical results can thus be obtained. A conversion to the *VTK file format* [71] enables the full, 3D visualization of the simulation run using *ParaView* [72]. A Python® based visual filter has been programmed to enable the non-standard visualization of non-spherical particles.

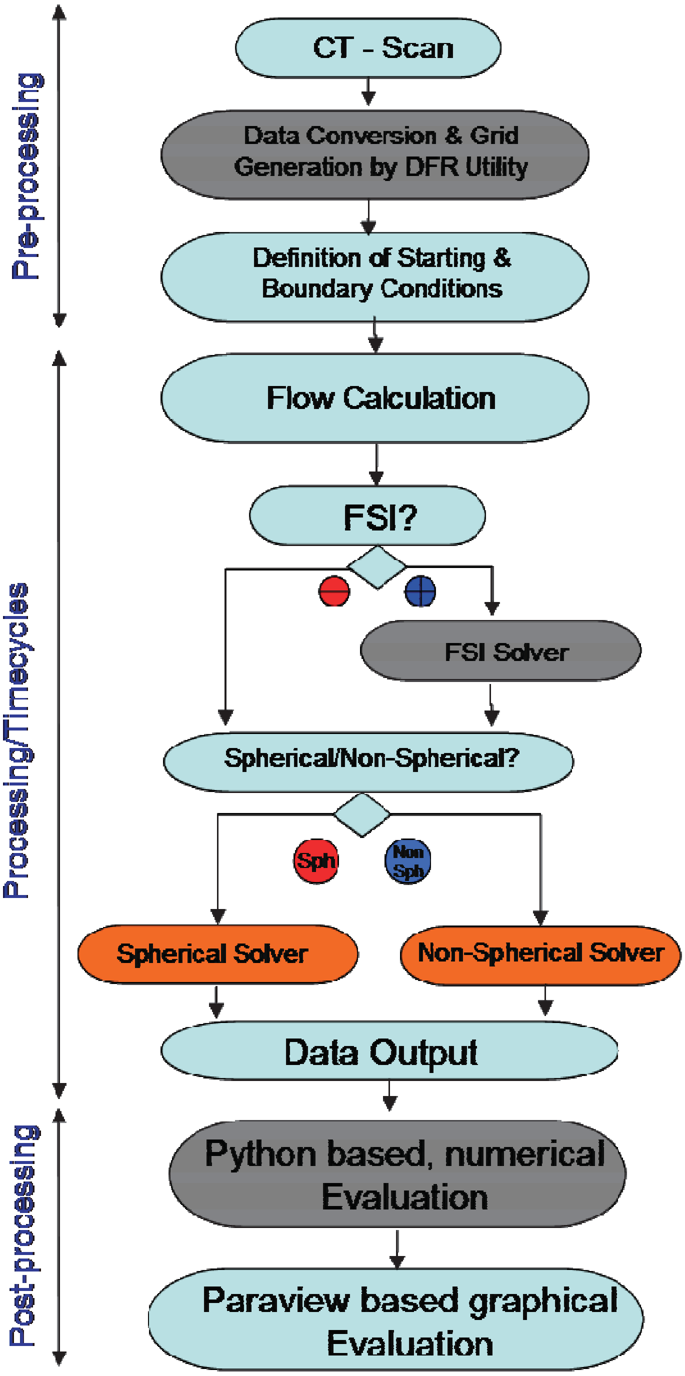


Figure 92: Activity diagram of total workflow, parted into pre-processing, processing and post-processing. Entities, colored in grey have been created through intense development effort in the course of the development project but are not discussed in detail in *this* work. Entities colored in orange are at the centre of attention of *this* thesis and are resolved in more detail in Figure 93.

7.2 Workflow for the Dirt Particle and Deposition Solvers

The workflow of the fluid calculation, in combination with the spherical or non-spherical solver shall hereby be depicted in more detail. Figure 93 contains the corresponding activity diagram. The following procedure is basically carried out during each time step:

- Decision whether or not *two way coupling* is to be initiated: If *negative*, the *Free Flow Drag* module becomes active, no deposition field is considered and the *Navier Stokes* equations are solved without the addition of a *Darcy term*. If *positive*, the *Fibre Vicinity Drag* module becomes active, the deposition field is considered and the *Navier Stokes* equations are solved with the addition of a *Darcy term*.

- Decision whether or not *the E-Static module* (see chapter 6.2) is to be initiated: Only if the *Fibre Vicinity Drag* module is active, the solver can switch to the *E-Static module.* If it is active then the distribution of a electric potential field is calculated within the spatial domain.

- Decision whether or not particles are to be injected at the current time step.

- Decision whether the spherical or the non-spherical solver is to be chosen: If particles are injected, the decision whether they are supposed to be spherical or non-spherical is made. Both particle solvers use the same basic structure:
  - Particle construction.
  - Particle movement during the particle time sub-cycles.
  - Decision whether or not the depot field is to be populated at particle positions.

- Output of particle-related data:
    - Position
    - Velocity vector
    - Rotation vector
    - Axe-Orientation vectors
    - Half axis diameters
    - Mass
    - Help point positions
    - Particle ID
    - Electric, particle load
    - Variable for current particle state

- Output of fluid/continuum-related data:
    - Fluid velocity field
    - Fluid pressure field
    - Particle momentum source field
    - Deposition field
    - Electro-static potential field
    - Electrical force field

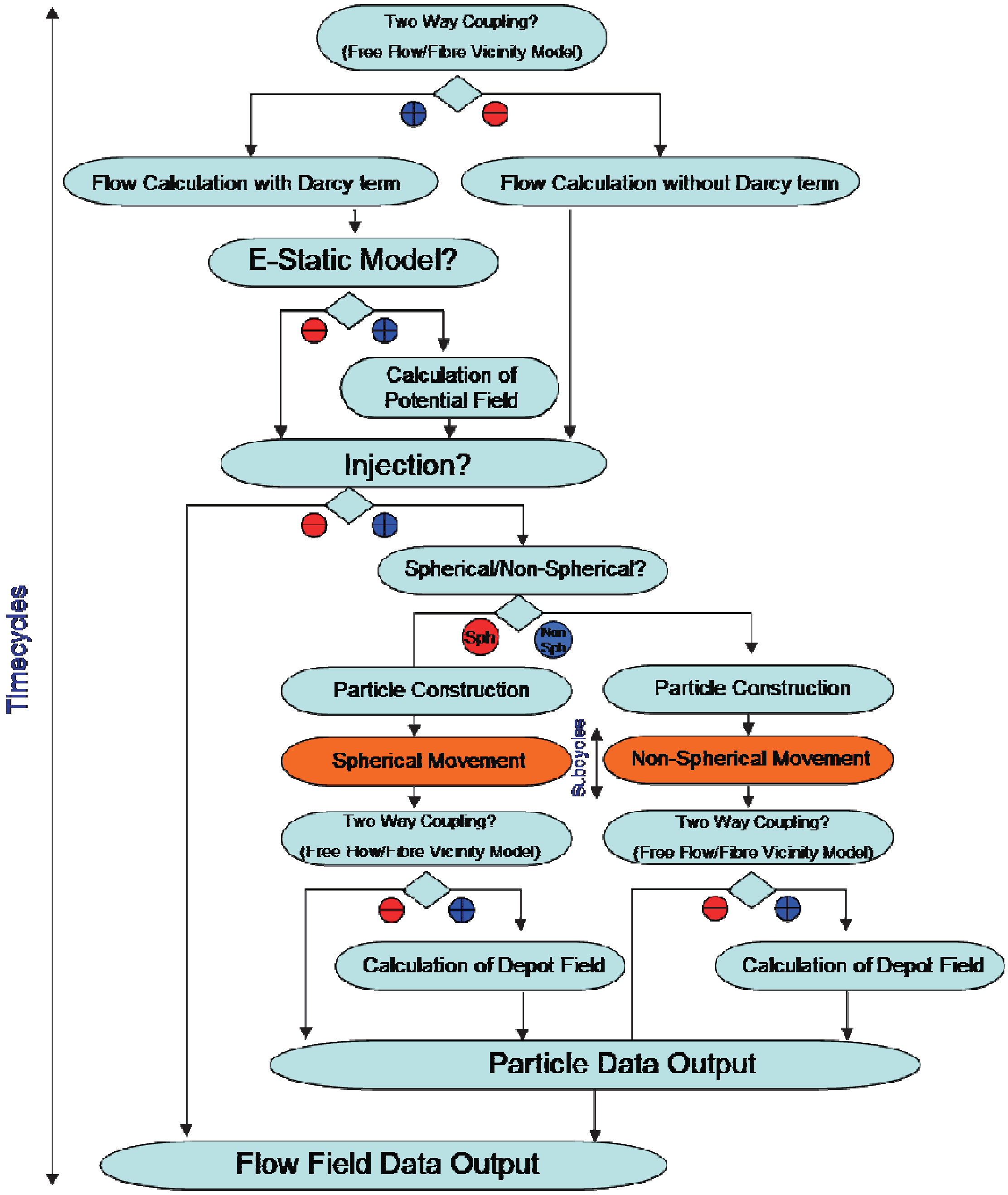


Figure 93: Activity diagram showing workflow of the fluid-particle simulation. Entities colored in orange are resolved in more detail in Figure 94.

7.3 Workflow for Particle Movement Calculation

By far the most intense development effort has been invested into the accurate particle movement and interaction calculation. A detailed workflow structure of the particle-movement method *(non)sphericalHardballParticle::move()* is depicted in Figure 94. It comprises essential steps like:

- Calculation of Help Point positions.
- Probing of the Help Points' projected path in order to detect obstacles.
- Detection of boundary patch impacts. If an impact occurs the decision between *wall*- or *fibre*- boundary patch is made and corresponding forces and/or moments are calculated.
- Conduction of actual particle movement.
- Calculation of gravity force.
- Calculation of fluid-particle interaction forces and/or moments.
- Detection of particle-particle collisions and calculation of corresponding force/moment effects.
- Calculation of total force/moment effects.
- Calculation of new translational and rotational velocity.
- Population of particle collision list for particle-particle collision model.
- Population of deposition field for fibre-vicinity drag model and/or E-static model.

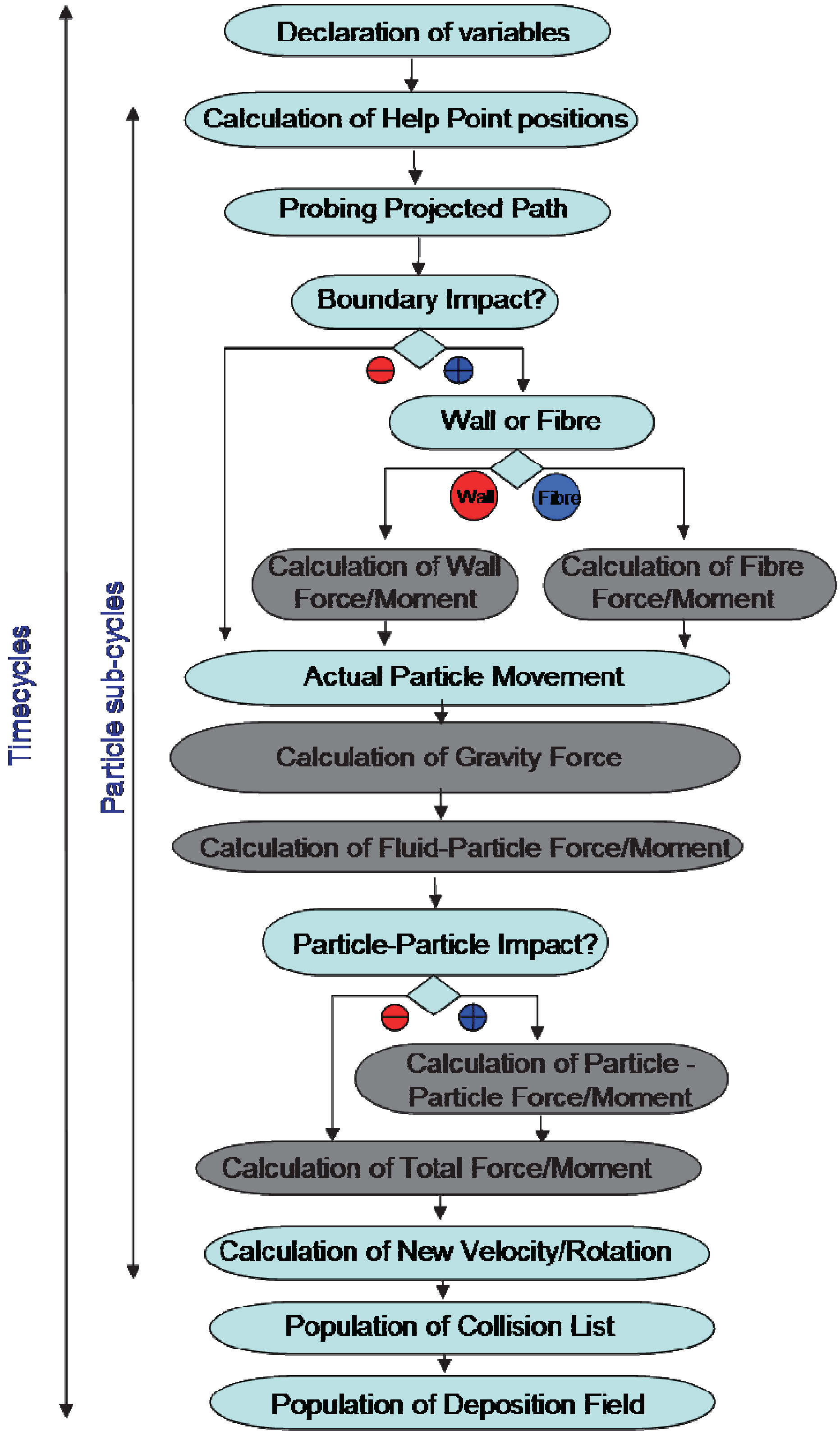


Figure 94: Activity diagram showing workflow of the particle movement algorithm. Entities colored in grey are individual force/moment contributions.

7.4 Inheritance Structure and Basic Functionality of Solver-Relevant C++ Classes

The software development included the creation and/or modification of several C++ classes [42], embedded within the OpenFOAM® programming framework. In the following those entities are cited and their basic functionality is described in brief.

- icoLagrangianFOAM.C:
  Location of main solver, fluid calculation, including finite volume matrix set-up, as well as PISO loop [73] and the main time loop.

- IncompressibleCloud.C, IncompressibleCloudI.C, IncompressibleCloudIO.C:
  Embodiment of the entire particle cloud. Injects and removes particles and stores data concerning the particle collective, such as the particle collision list.

- nonSphericalHardballParticle.C, nonSphericalHardballParticleI.C, non-SphericalHardballParticleIO.C:
  Store data, calculate interaction forces and handle movement of individual (non-)spherical particles. Core classes of advanced solver.

- sphericalHardballParticle.C, sphericalHardballParticleI.C, spherical-HardballParticleIO.C:
  Store data, calculate interaction forces and handle movement of individual spherical particles. Core classes of original solver.

- cloudDelegate.C, newCloudDelegate.C, sphericalCloudDelegate.C, non-sphericalCloudDelegate.C:
  Distinguish between the original spherical- and the more highly developed (non-)spherical particle solver.

- Cloud.C, CloudI.C, CloudIO.C: Template class for IncompressibleCloud.C. Includes several, important methods which help to track the particles through the fibre domain.

- Particle.C, ParticleI.C, ParticleIO.C: Template class for sphericalHardballParticle.C and nonSphericalHardballParticle.C.

The complex relation and the sharing of abilities between those classes, as well as their embeddement within the superior OpenFOAM® programming framework is best shown via an inheritance diagram which is depicted in Figure 95.

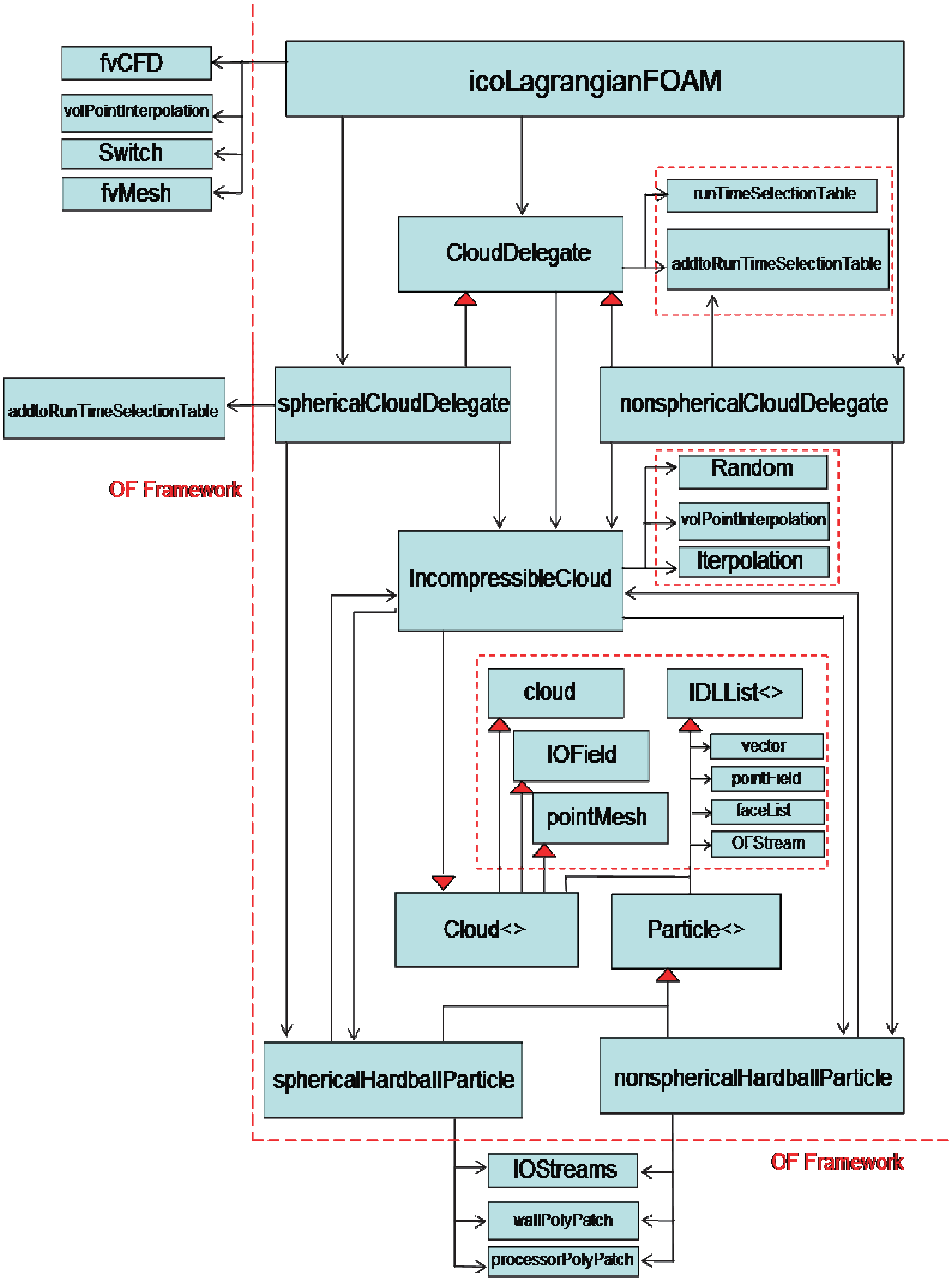


Figure 95: Inheritance diagram of C++ based, (non-)spherical particle solver. The dashed red lines separate the self-designed or modified solver classes from the standard OpenFOAM® program framework.

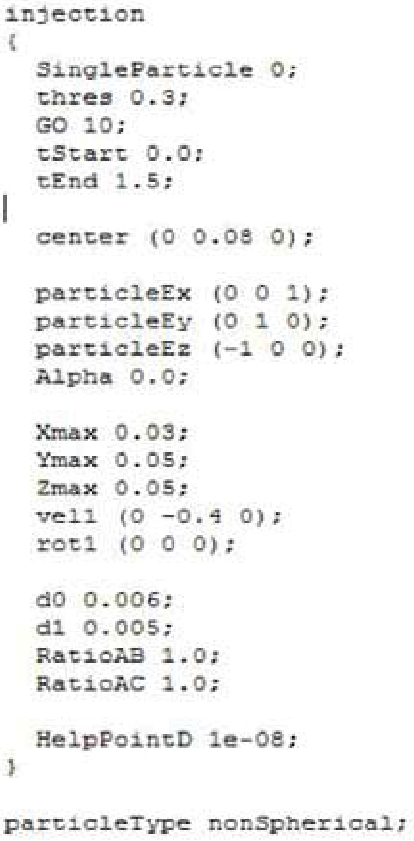
7.5 User Options and Dictionary

All decisive parameters for defining particle behaviour within their surroundings and amongst each other are specified within the *cloudProperties* dictionary, which is a sub-dictionary of the case-specific *constant* dictionary. Figures 96 through 100 show outtakes of the *cloudProperties* dictionary with some arbitrary input parameters. The input file is parted into several sub-sections which define categories of particle behaviour.

- Sub-section injection, shown in Figure 96. It defines location, distribution and time-span of particle injection. The following parameters can be specified here:

    - *SingleParticles*: type Boolean, values 0/1; States whether a single, accurately positioned test particle is injected (0) or whether a particle cloud with defined distribution is injected over a period of time (1).
    - *Thres, G0*: type scalar; Defines frequency and order of magnitude of particle number to inject.
    - *tStart, tEnd*: type scalar, values "0.0 - *maximum runtime*"; Define the period of injection time.
    - *Centre*: type vector, values "*within domain*"; Defines centre of particle injection.
    - *Xmax, Ymax, Zmax*: type scalar; Define absolute, maximum deviation of injection coordinates around injection centre.
    - *particleEx, particleEy, particleEz*: type vector; Define basic orientation of particle main axes a, b, c at injection.
    - *Alpha*: type scalar; Defines maximal orientation deviation around basic injection orientation.
    - *vel1, rot1*: type vector; Define velocity and rotation at injection.
    - *d0*: type scalar; Defines basic particle diameter;
    - *d1*: type scalar; Defines maximum axis deviation around basic particle diameter;

- *RatioAB, RatioAC*: type scalar; If not *zero* they define fixed, half axis ratios, overruling *d1*.
- *HelpPointD*: type scalar; Defines diameter of help points. Only relevant for visualization.



```
injection
{
  SingleParticle 0;
  thres 0.3;
  GO 10;
  tStart 0.0;
  tEnd 1.5;
|
  center (0 0.08 0);

  particleEx (0 0 1);
  particleEy (0 1 0);
  particleEz (-1 0 0);
  Alpha 0.0;

  Xmax 0.03;
  Ymax 0.05;
  Zmax 0.05;
  vel1 (0 -0.4 0);
  rot1 (0 0 0);

  d0 0.006;
  d1 0.005;
  RatioAB 1.0;
  RatioAC 1.0;

  HelpPointD 1e-08;
}

particleType nonSpherical;
```

Figure 96: Outtake of *cloudProperties* dictionary. Sub-section injection.

- Sub-section *wall*, shown in Figure 97. It defines particle-wall and particle-fibre interaction. The following parameters can be specified here:

  - *reflect*: type int, values 0, 1, 2; States whether a particle disappears (0), is reflected semi-elastically (1) or is reflected 100% elastically (2) at wall boundary patch impact.
  - *elasticity*: type scalar, values 0.0-1.0; Defines wall impact elasticity if reflect=1.

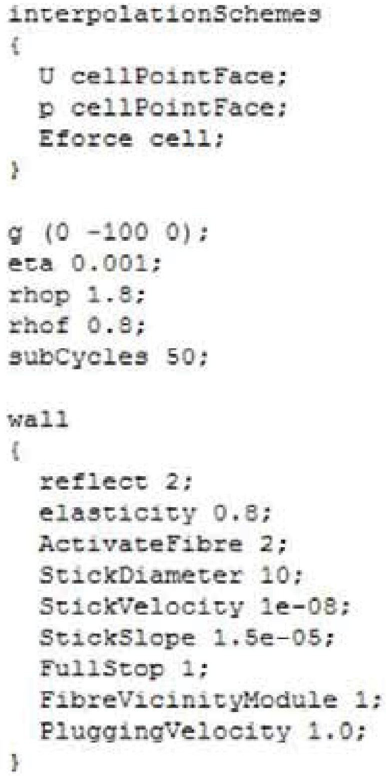
- *ActivateFibre*: type Boolean, values 0/1; States whether the fibre shows adhesion to the particles (1) or not (0).
- *StickDiameter*: type scalar. Defines particle diameter below which particles that hit a sticking fibre (*ActivateFibre*=1) stop at the surface.
- *Stickvelocity*: type scalar; Defines magnitude of velocity below which particles in contact with a fibre immediately stop at the surface.
- *StickSlope*: type scalar; Defines the shape of a Gaussian probability distribution deciding whether particles with larger diameter than *StickDiameter* stop at the fibre surface.
- *FullStop*: type Boolean, values 1/0; States whether a full stop due to adhesional sticking is allowed (1) or not (0).
- *FibreVicinityModule*: type Boolean, values 1/0; States whether the Fibre Vicinity module (see chapter 4.2.6) is active at all times (1) or whether the Free Flow module (see chapter 4.2.5) is active to start with and the Fibre Vicinity module becomes active if a particle decelerates below *PluggingVelocity* within fibre vicinity (0).
- *PluggingVelocity*: type scalar; Defines the switching velocity between Fibre Vicinity module and Free Flow module if *FibreVicinityModule=0;*

- Sub-section *interpolationSchemes*, shown in Figure 97. It defines the interpolation methods for interpolating field cell values to arbitrary coordinate-based values. The following parameters can be specified here:

  - *U*: Interpolation scheme for velocity field. For options see [39].
  - *p*: Interpolation scheme for pressure field. For options see [39].
  - *Eforce*: Interpolation scheme for electrical force field. For options see [39].

- Sub-section *general*, shown in Figure 97. It defines special physical and numerical properties of the solution which overrule those in other solver directories. The following parameters can be specified here:

- *g*: type vector; Defines gravity vector.
- *eta*: type scalar; Defines kinematic fluid viscosity.
- *rhop*: type scalar; Defines ratio of particle density to the density of water.
- *rhof*: type scalar; Defines ratio of fluid density to the density of water.
- *SubCycles*: type scalar; Defines number of particle sub-time steps per fluid time step. If set to *zero*, the adaptive time stepping scheme (see chapter 5) is activated.



```
interpolationSchemes
{
  U cellPointFace;
  p cellPointFace;
  Eforce cell;
}

g (0 -100 0);
eta 0.001;
rhop 1.8;
rhof 0.8;
subCycles 50;

wall
{
  reflect 2;
  elasticity 0.8;
  ActivateFibre 2;
  StickDiameter 10;
  StickVelocity 1e-08;
  StickSlope 1.5e-05;
  FullStop 1;
  FibreVicinityModule 1;
  PluggingVelocity 1.0;
}
```

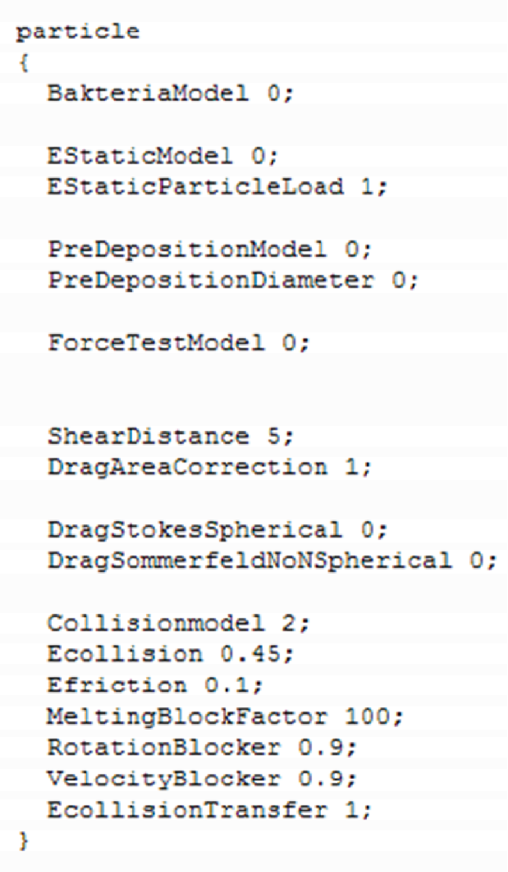
Figure 97: Outtake of *cloudProperties* dictionary. Sub-sections *interpolationSchemes*, *general* and *wall*.

- Sub-section *particle*, shown in Figure 98. It governs the activation of sub models. Particle-fluid interaction characteristics and particle-particle interaction. The following parameters can be specified here:

  - *BakteriaModel*: type Boolean, values 0/1; States whether the particles behave as Epiterial bacteria (1) or as solid objects (0) at fibre impact.

- *EStaticModel*: type Boolean, values 0/1; Activates (1) or deactivates (0) the E-Static module (see chapter 6.2).
- *EStaticParticleLoad*: type scalar; Defines the volume specific particle charge if *EstaticModule*=1.
- *PreDepositionModel*: type Boolean, values 0/1; Activates (1) or deactivates (0) the Pre-Deposition model which assumes a certain, immobilized particle distribution to be already present at the beginning of the simulation run.
- *PreDepositionDiameter*: type scalar; Defines the diameter of the pre-deposited particles if *PreDepositionModel=1.*
- *ForceTestModel*: type Boolean, values 0/1; Activates (1) or deactivates (0) the Force Test Model which injects one particle, permanently sets its velocity to *zero* and plots the fluid-particle drag- and lift forces that start to act on the particle in a stream.
- *ShearDistance*: type scalar; Defines the distance between particle surface and fluid velocity sample positions to calculate fluid-particle forces. Parameter is given in percentile fraction of half axis a. Valid only for activated Fibre Vicinity module.
- *DragAreaCorrection*: type Boolean, values 1/0; Activates (1) or deactivates (0) the surface area correction from actual, numerical particle surface to smooth, ellipsoid surface area. For fluid-particle force calculation.
- *DragStokesSpherical*: type Boolean, values 1/0; Activates (1) or deactivates (0) the simple Stokes drag law for fluid-particle force calculation.
- *DragSommerfeldNoNSpherical*: type Boolean, values 1/0; Activates (1) or deactivates (0) the *Hölzer-Sommerfeld* [24] drag law for fluid-particle force calculation.
- *CollisionModel*: type int, values 0, 1, 2; States whether particles shall not recognize each other at all (0), shall only recognize each other within the fibre vicinity (1) or shall recognize each other at all times (2).
- *ECollision*: type scalar, values 0.0-1.0; Defines the elasticity of particle-particle impact, where *Ecollision=0.0* means 100% plastic impact and *Ecollision=1.0* means 100% elastic impact.
- *EFriction*: type scalar, values 0.0-1.0; Defines the extent of particle-particle friction forces, where *EFriction=0.0* means no friction at all and *EFriction=1.0* means total stop of relative, horizontal particle movement at impact point.
- *MeltingBlockFactor*: type scalar; Defines the strength of an auxiliary force which prevents particles from melting into each other. Activation advisable in the case of dense particle accumulations.
- *VelocityBlocker*: type scalar, values 0.0-1.0; Defines the strength of an additional particle-particle, translational impact velocity blocker to reduce translational, kinetic energy of a particle collective. Activation advisable in the case of dense particle accumulations.

- *RotationBlocker*: type scalar, values 0.0-1.0; Defines the strength of an additional particle-particle, rotational impact velocity blocker to reduce rotational, kinetic energy of a particle collective. Activation advisable in the case of dense particle accumulations.
- *ECollisionTransfer*: type scalar, values 0.0-1.0; Defines the extent of force transfer within a particle collective. If *EcollisionTransfer>0* the particle-particle impact force is calculated not only on the basis of relative movement but also in dependence of other, external forces acting on the collision partners (see chapter 4.2.7.3). Activation advisable in the case of dense particle accumulations.



```
particle
{
  BakteriaModel 0;

  EStaticModel 0;
  EStaticParticleLoad 1;

  PreDepositionModel 0;
  PreDepositionDiameter 0;

  ForceTestModel 0;


  ShearDistance 5;
  DragAreaCorrection 1;

  DragStokesSpherical 0;
  DragSommerfeldNoNSpherical 0;

  Collisionmodel 2;
  Ecollision 0.45;
  Efriction 0.1;
  MeltingBlockFactor 100;
  RotationBlocker 0.9;
  VelocityBlocker 0.9;
  EcollisionTransfer 1;
}
```

Figure 98: Outtake of *cloudProperties* dictionary. Sub-sections *particle*.

7.6 The Graphical User Interface

The final version of the unified filtration solver will be equipped with an easy-to-use, graphical interface. This surface feature is supposed to direct the user through the entire process of fibre reconstruction, OpenFOAM® based FSI and/or dirt particle and deposition calculation as well as result evaluation. Furthermore it will enable the data transfer to an interlinked data-base where static and dynamic material properties can be stored. Figure 99 presents an overview of the main program modules e.g. the working steps and their relation within the workflow.

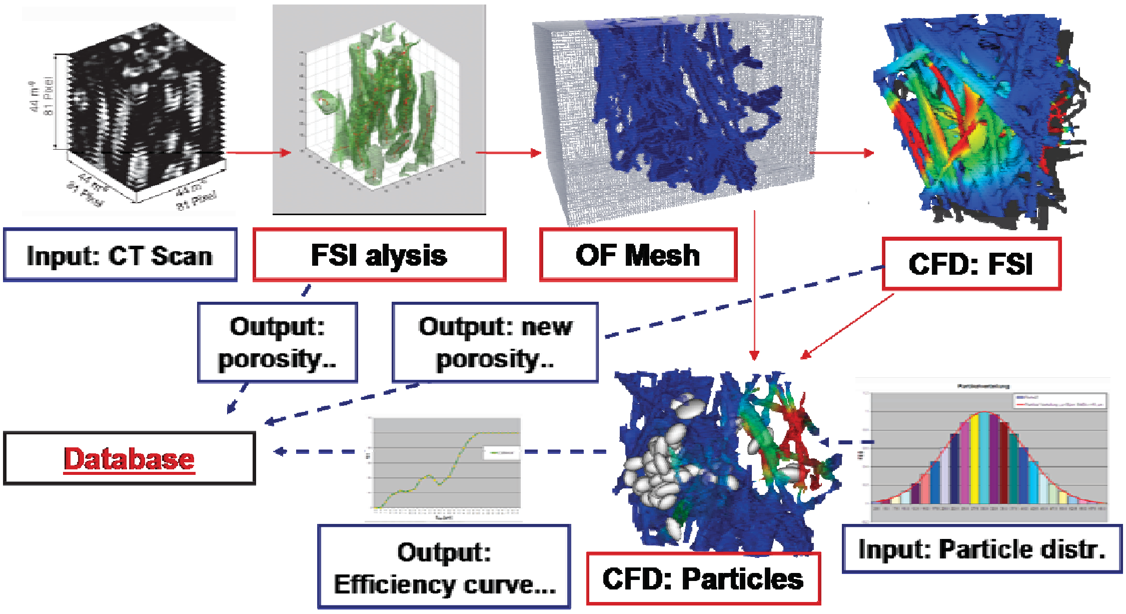


Figure 99: Sketch of the basic structure of the graphical user interface as it interlinks program modules according to the user workflow. Blue tags symbolize data in or output. Dashed blue arrows symbolize data flow. Red tags symbolize program modules e.g. working steps and red arrows symbolize the user workflow.

8 Experimental and (Semi-)Analytical Verification

In order to assure the validity of any simulation, qualitative and quantitative verification is imperative. This chapter is dedicated to prove qualitative solver functionality and to present quantitative evidence of result correctness. The simulation is hereby compared to a thoroughly constructed, semi-analytical verification scheme, to data from literature and to extensive experimental runs. In the given case, verification can only be based on comparison of statistically averaged results. Crucial process parameters such as pressure drop, filter fibre efficiency and particle penetration depth are the key to comparing calculations and experiment.

8.1 Semi-Analytical Verification Scheme for Simplified Geometries

A first, important step in validating qualitative aspects of solver functionality can be taken by comparing CFD calculations and semi-analytical results for artificially created, simplified fibre geometries. The process parameter to be chosen for benchmarking is filter fibre efficiency E. It is defined by:

$$E_i = n_{s,i}/n_i \qquad (271)$$

Here E_i is the fractional filter fibre efficiency of size class i, n_i is the total number of dirt particles per size class i and $n_{s,i}$ is the number of dirt particles retained by the filter. Note that throughout this work the filter fibre efficiency is defined by filter impact on *monodisperse* particle fractions. All conducted numerical calculations, experiments and evaluations take this definition into account.

8.1.1 Simplified Geometry

In order to be able to establish a well defined, semi-analytically derived filter fibre efficiency curve over the entire, relevant regime of particle sizes (2µm-100µm), a simplified fibre geometry has to be created. The main reasons for that are:

- Reduction of complexity to allow focus on main qualitative aspects of the solution.

- Lower computational cost for various test runs.

- Simpler determination of basic geometric parameters than with realistic geometries. Geometric parameters with relevance for the calculation are:

  - Frontal, free flow area due to pores, A_{pores}
  - Number of frontal, free flow channels, in other words: number of projected pores, n_{pores}
  - Pore size distribution: medium pore diameter (MPD) D_{pores} of projected, free flow area per pore, standard deviation around MPD, σ_{pores}
  - Medium fibre diameter D_{fibre}

Figure 100 shows an exemplary, simplified, 30.000 cell, fibre geometry, where all relevant geometric parameters are easily determinable.

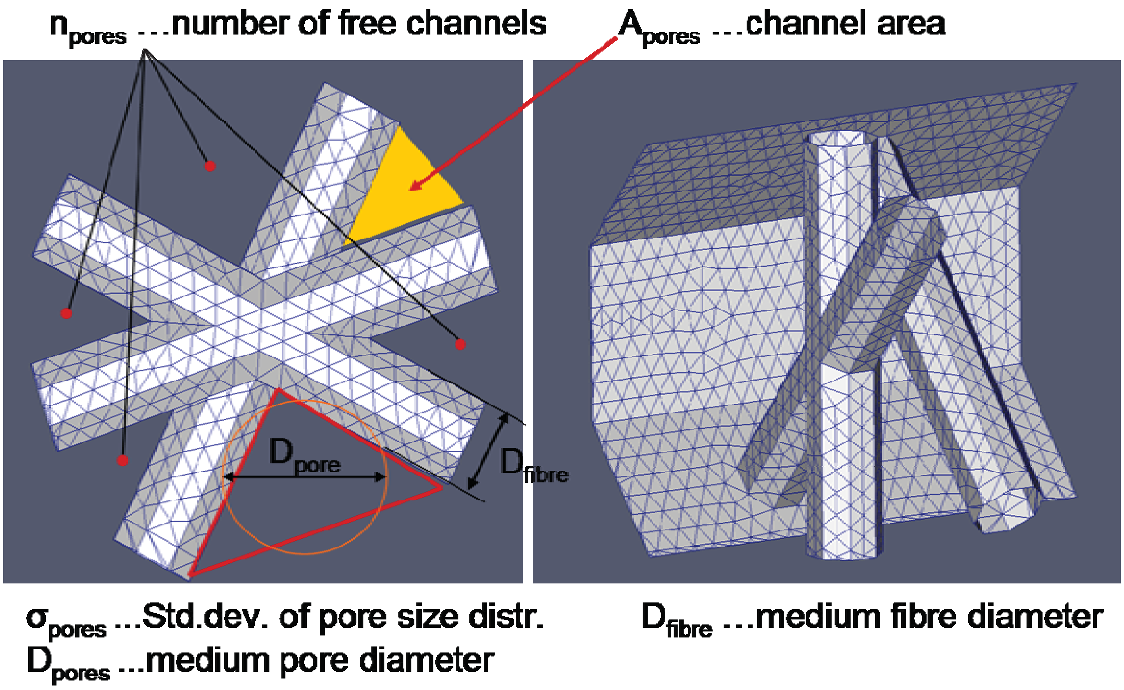


Figure 100: Exemplary image of simplified fibre geometry. Easy determination of geometric parameters A_{pores}, n_{pores}, D_{pores}, σ_{pores}, D_{fibre}.

8.1.2 Semi-Analytical Approach

Obviously the set up of a semi-analytical model for filter fibre efficiency calculation requires the consideration of all physically relevant filtration effects. Since Knudsen numbers are well below 0.015, fluid continuum conditions are predominant and Navier Stokes equations are valid.

The influence of diffusive motion on particle movement can be neglected as well, since Péclet numbers are relatively high ($Pe>5*10^7$). Therefore the main filtration effects to be considered over a dirt particle size-range of 2µm-100µm and particle Reynolds numbers $Re_p<1$ are:

- Inertial impact
- Particle–fibre adhesion
- Blow-off due to particle momentum or interactions

- Sieving

It should be noted that Banzhaf [70] concludes that inertial impact is negligible for fluid filtration because the prevailing Stokes numbers ($1*10^{-9} \leq St \leq 2*10^{-3}$,) are too low. The presented simulation model might very well yield the same result, it is however, set up in order not to exclude the possibility of inertial impact *apriori*.

Based on the consideration of the filtration effects, cited above, the total particle deposition probability P_{Dep} of any particle (which is equivalent to E) can be calculated as a function of various *"single effect"* deposition contributions (see Equ.272). Hereby the *inertial impact* contribution is described by the hitting channel probability P_{HCh} and by the *inertial impact* probability P_{Stokes} which is quantified via St (see Equ.274). The *particle–fibre adhesion* as well as *blow-off* effects can be considered via the particle sticking probability P_{Stick}. Sieving contributions to overall filter fibre efficiency are finally quantified over the pore size dependent, *flow path blocking probability* P_{Block}.

$$P_{Dep} = P_{HCh} P_{Stokes} P_{Stick} + \left(1 - P_{HCh} P_{Stokes} P_{Stick}\right) P_{Block} \qquad (272)$$

Inertial Impact Effects

In order for any particle to impact on a fibre surface, two conditions will have to be satisfied. The first condition states that the trajectory of the particle must define a particle-fibre collision course. The second condition is about the particle-fibre Stokes number. It states that inertial contributions must outweigh viscous force contributions within the PME, so that the particle will not follow the streamlines around the fibre, but will break out and collide with the obstacle.

In the context of this work a particle is defined to be on collision course with a fibre as long as at least the fraction n of D_{sph} overlaps with a part of the projected, cross-sectional, fibre-covered area A_{Fibre} of the entire flow channel A_{FCh}. With knowledge of the projected, cross-sectional, free flow area due to pores A_{Pores}

and the number of pores n_{Pores}, the hitting channel probability can be calculated as:

$$P_{HCh} = \frac{A_{fibre} + D_{sph} \cdot (1/2 - n) \cdot \sqrt{A_{pores} \cdot n_{pores} \cdot \pi}}{A_{FCh}} \tag{273}$$

Even if a particle is on collision course, impact is still not assured due to fluid deviation around the fibre. Larger particle inertia increases the likelihood of impacting anyway. The inertial impact probability P_{Stokes} can be characterized via the Stokes number, which gives a dimension-less relation between inertial and viscous forces on the particle:

$$St = \frac{\rho_p D_{sph}^2 |u_{rel}|}{18 \cdot \mu_f D_k} \tag{274}$$

Here D_k stands for the diameter of the collector, which in our case is the average fibre diameter D_{fibre}.

The minimal, critical particle-fibre Stokes number St_{min} is defined as the level below which any particle, even though on collision course, is deviated around the fibre (see Figure 101). The maximum, critical Stokes number St_{max}, on the other hand, is hereby defined as the level above which any particle, whose course overlaps with at least the fraction n of D_{sph} with a fibre-covered area, impacts on the fibre structure (see Figure 100).

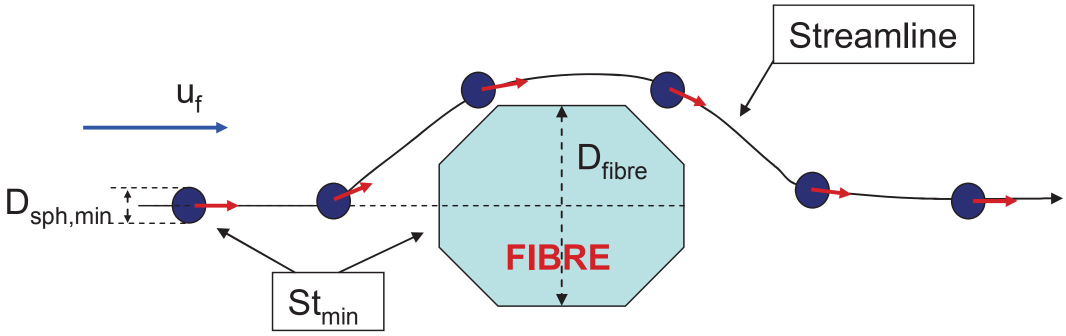


Figure 101: Definition of St_{min}. The largest particle on direct collision course that just passes by the fibre without hitting, defines St_{min}.

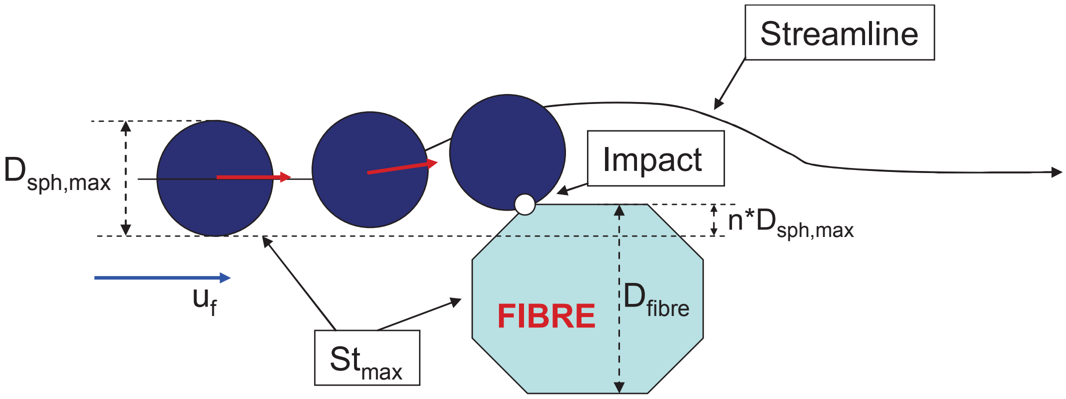


Figure 102: Definition of St_{max}. The smallest particle that just collides with the fibre, even though its collision course is just such that the n^{th} fraction of D_{sph} overlaps with a fibre, defines St_{max}.

Using those definitions, the inertial impact probability is in this work calculated as:

$$P_{Stokes} = \frac{St(D_{sph}) - St_{min}}{St_{max} - St_{min}} \tag{275}$$

In order to obtain values for St_{min} and St_{max} a simple, numerical simulation, where particles are set on a collision course with fibres of diameter D_{fibre} is used (equivalent to sketches in Figure 99 and Figure 100). The parameter n is set to 1.0%. The overall hitting probability P_{hit} can now be semi-analytically determined and is:

$$P_{Hit} = P_{HCh} P_{Stokes} \tag{276}$$

For “small” particles with $D_{sph} < D_{stop}$, the hitting probability is equivalent to the deposition probability. For “larger” particles with $D_{sph} > D_{stop}$, other effects have to be taken into account as well.

Particle–Fibre Adhesion and Blow-Off due to Particle Momentum or Interactions

The user-definable, adhesional deposition probability for “medium sized“ particles with $D_{stop} \leq D_{sph} \leq D_{siev}$ is deeply incorporated into the force interaction scheme of the CFD solver (see chapter 4.2.7.2). For the sake of semi-analytical verification of solver functionality, the underlying PDDF for the particle-adhesional sticking probability P_{stick} is chosen. The same parameters (D_{stop}, σ_d) that are used in the CFD run are inserted, thus P_{stick} is calculated as:

$$P_{stick} = \frac{1}{\sigma_d \sqrt{2\pi}} e^{-\frac{(D_{sph} - D_{stop})^2}{2\sigma_d^2}} \tag{277}$$

Sieving due to Pore Sizes

For “large” particles with $D_{sph}>D_{siev}$ the sieving effect is the most relevant reason for deposition. It is simply based upon the relation of particle diameter to pore size distribution $\Phi_{pores.}$ Underlying a Gauss-Normal distribution [30] of pore sizes, and using the geometric parameters medium pore size μ_{pores} and standard deviation σ_{pores}, the distribution function Φ_{pores} is given by:

$$\phi_{pores}(D_{sph}) = \frac{1}{\sigma_{pores} \sqrt{2\pi}} e^{-\left(\frac{(D_{sph} - \mu_{pores})^2}{2\sigma_{pores}^2}\right)} \tag{278}$$

The cumulative pore size distribution function is equivalent to the flow path blocking probability P_{Block} that can be calculated by use of the error-function *erf*:

$$P_{Block} = \int_{t=0}^{t=D_{sph}} \phi_{pores}(t)dt = \frac{1}{2}\left[1 + erf\left(\frac{D_{sph} - \mu_{pores}}{\sqrt{2}\sigma_{pores}}\right)\right] \tag{279}$$

Comparing CFD and Analytical Results

By using Equ.272 through Equ.279, the whole semi-analytical approach to describe the particle deposition probability within a simplified fibre geometry can be summarized by:

$$P_{Dep} =$$

$$\frac{(St-St_{\min})\left(A_{fibre}+2\cdot D_{sph}\cdot(1-n)\cdot\sqrt{A_{pores}\cdot n_{pores}\cdot\pi}\right)}{2\sqrt{2\pi}\sigma_d A_{FCh}(St_{\max}-St_{\min})}e^{-\frac{(D_{sph}-D_{stop})^2}{2\sigma_d{}^2}}\left\{\left[1-erf\left(\frac{D_{sph}-\mu_{pores}}{\sqrt{2}\sigma_{pores}}\right)\right]+\frac{1}{2}+\frac{1}{2}erf\left(\frac{D_{sph}-\mu_{pores}}{\sqrt{2}\sigma_{pores}}\right)\right\} \tag{280}$$

By inserting corresponding material, and geometric parameters into Equ.280, and into the input dictionaries of the CFD model, two comparable filter fibre efficiency curves are obtained. The results of the comparison are shown in Figure 103. Semi-analytical results and CFD results match qualitatively and quantitatively. A verification of qualitative solver functionality is hereby considered to be achieved.

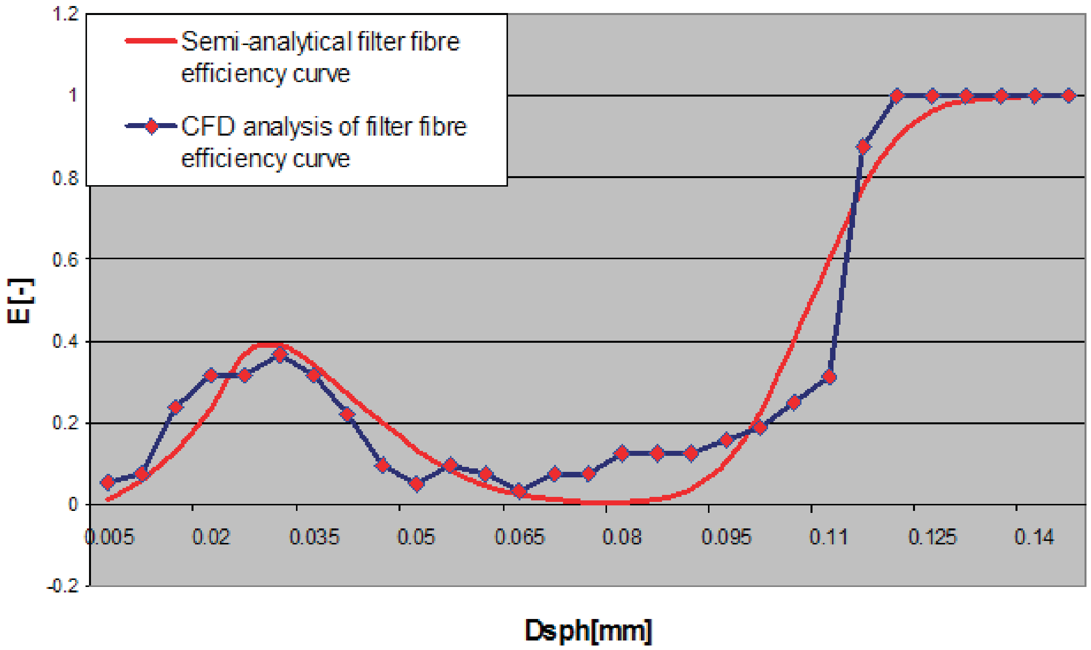


Figure 103: Comparison of filter fibre efficiency results for semi-analytical calculation (red) according to Equ.280 and for CFD results (blue) using the non-spherical particle solver. The underlying, simplified geometry is depicted in

Figure 100. Conditions: ρ_f=800kg/m³, μ_f=0.002Pas, ρ_p=1500kg/m³, u_f=0.02m/s, 5µm≤D_{sph}≤150µm.

8.2 Verification by Comparison to Data from Literature

A second method to verify simulation results is to retrieve comparable data from literature. Here results, published by Banzhaf [70], are used to achieve that. In his PhD thesis Banzhaf derives a semi-analytical model to predict fluid-fibre deposition efficiencies in relation to mean filter fibre diameter D_f and porosity ε_f. He evaluates the functions for the two parameters and compares the results to a “typical” filter fibre medium.

In order to produce comparable data, the digital reconstruction of a filter fibre medium with call sign “A43” has been used to conduct simulation runs. Even though the known, physical filter fibre properties (see below) match pretty well between the compared cases, total equivalence is not given. However, the results in Figure 104 clearly show that the output, yielded by the CFD model, ranges well within the bounds of the published, comparable data. The set-up behind the five E-curves shown in Figure 104 is as follows:

Blue curve: Efficiency data derived by the CFD model. Underlying fibre material is A43 with D_f~25µm, ε_f~86% and filter thickness s_f=1000µm.

Red curve: Efficiency data derived from measurements according to [70]. Underlying fibre material is described as “typical” fluid filter with D_f=30µm, ε_f=88% and s_f=800µm.

Dark-orange/orange/light-orange curves: Efficiency data derived from semi-analytical model according to [70]. Underlying, theoretical fibre materials feature D_f=20µm/30µm/20µm, ε_f=90%/85%/85% and s_f=800µm.

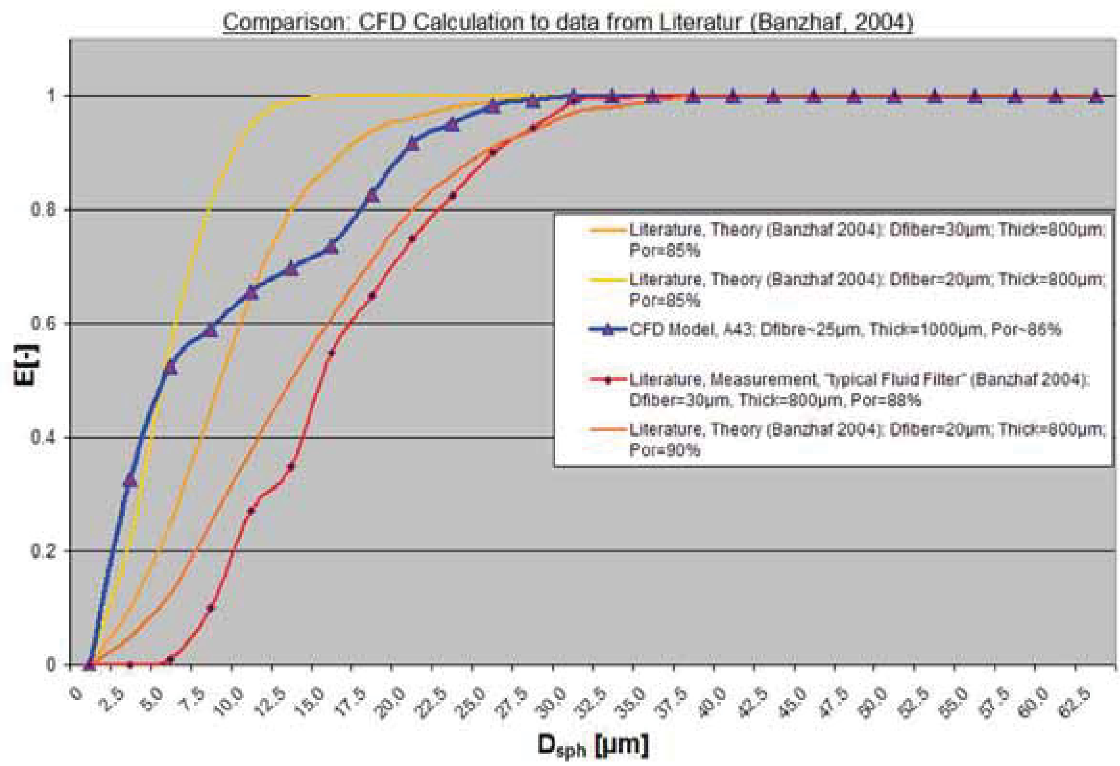


Figure 104: Comparison of filter fibre efficiency curves. CFD model results from A43 featuring D_f~25µm, ε_f~86% and s_f=1000µm (blue). Results from literature, based on experiments with "typical" filter fibre medium featuring D_f=30µm, ε_f=88% and s_f=800µm (red). Results from literature, based on semi-empirical model for theoretical filter fibre media featuring D_f=20µm/30µm/20µm, ε_f=90%/85%/85% and s_f=800µm (dark-orange/orange/light-orange).

The CFD model results in Figure 104 lie well within published data on materials with very similar properties. Throughout the entire, relevant particle-diameter spectrum the measured curve (red) shows lower efficiency values than the calculated curve (blue). This fact does correspond with the difference in physical properties between the two media underlying the plots: The A43 features lower porosity with $\Delta\varepsilon_f$~2%, smaller medium fibre diameter with ΔD_f~5µm, leading to smaller pore diameters and is thicker with Δs_f=200µm, than the presented reference medium.

8.3 Experimental Filter Fibre Analysis and Verification

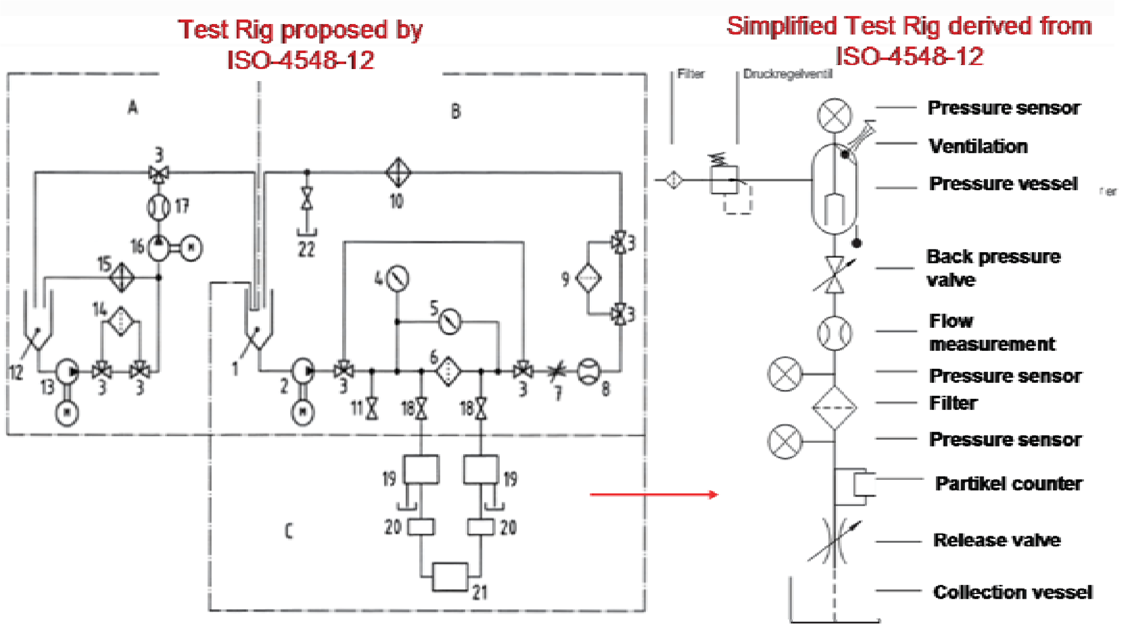
In parallel to the development of the CFD solver a comprehensive, experimental rig was devised and built. The experimental part of the work was initially intended to merely serve as means of verification but turned out to yield a new method for characterizing dirt particle distribution behaviour in any standard, fluid filter fibre medium. Two main devices constitute the corner stone of the experimental effort behind this work:

- The oil-fibre test facility.
- The particle distribution detection facility.

8.3.1 The Oil-Fibre Test Facility

The oil-fibre test facility is a fibre sample testing device designed to investigate the material's reaction to an oil stream. While exposing any circular fibre sample of diameter D_s=2.5cm to controlled flow conditions of test-particle laden oil, the development of decisive process parameters can be closely monitored, stored and electronically processed. The parameters are: pressure drop Δp_s, volumetric flow rate v_s and cumulative oil volume V_s over the sample.

The facility has been planned and constructed according to proposals within ISO 4548-12 [82] concerning the set-up of fluid filter fibre multi-pass tests. Figure 105 shows a comparison of the underlying test-rig process plan, proposed by ISO 4548 and the derived, simplified version which corresponds to the oil-fibre test facility.



Test Rig proposed by ISO-4548-12

Simplified Test Rig derived from ISO-4548-12

Figure 105: Process plan of multi-pass test rig proposed by ISO 4548-12 (left) opposed by process plan of simplified version (right) underlying the oil-fibre test facility.

Figure 106 presents an image of the actual testing facility, opposed by the previously introduced construction plan, highlighting its main components namely:

- Pressure Tube:

Effective Flow Diameter: 1.5cm, max. Pressure: 6.0bar.

- Compressor:

Mr.Tool TurboAir®, 25/180 with pressure control valve, safety valve and two separated pressure chambers. Maximum operating pressure: 8.0bar. Maximum test pressure: 12.0bar.

- Pressure vessel:

Krautzberger® MDC 10l with pressure control valve, safety valve, air-pressure mixing facility and electro-magnetic mixing facility. Maximum operating pressure: 6.0bar. Maximum test pressure: 8.6bar.

- Flow measurement device:

Bürkert® 8071, oval wheel flow sensor. Measurement range: 0.5l/h-500l/h. Measurement error: ≤1.0% of measured value.

- Magnetic back-pressure valve:

Bürkert® 2833, 2/2 ways. Range of functionality: 0.0-16.0bar.

- Pressure sensors:

JUMO MIDAS® 401001 capacitive sensor. Measurement range: 0.0-6.0bar. Measurement error: ≤0.5% of maximal, measured value.

- Sample holding device:

SANTORIUS® with flow diameter 2.5cm (=diameter of filter fibre sample). Maximum operating pressure: 5.0bar.

- Bus system and control software:

PCI/USB based bus system with analog-digital converter from National Instruments® and LabVIEW® based control software.

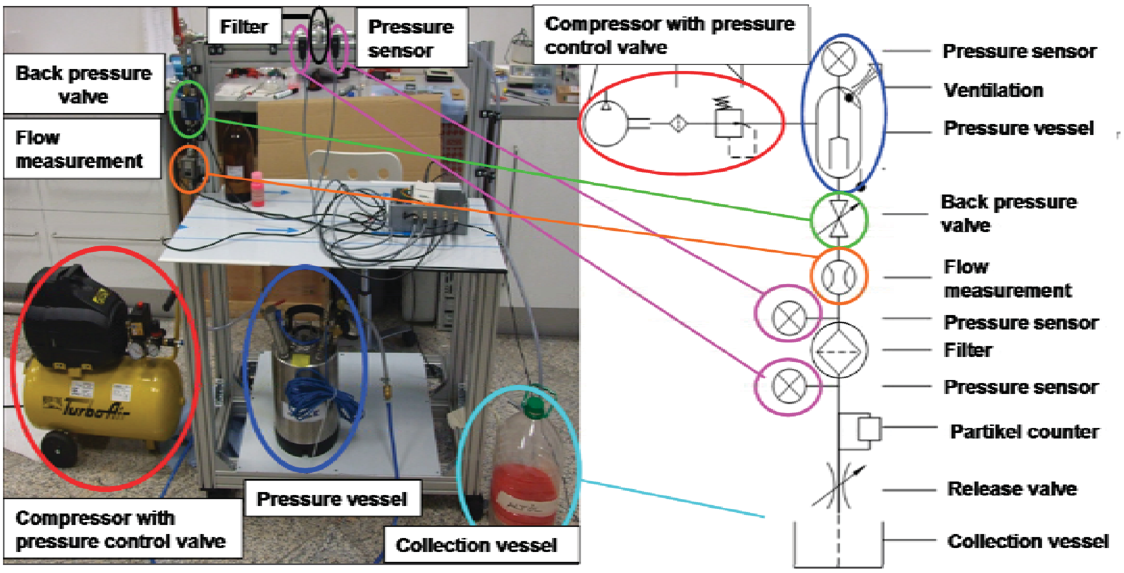


Figure 106: Image of actual oil-fibre test facility (left), opposed by process plan (right). Main components are highlighted.

Figure 107: Images of main components of test facility: 1.) compressor, 2.) pressure vessel, 3.) flow measurement device, 4.) magnetic valve, 5.) magnetic valve display, 6.) pressure sensor and 7.) sample holding device.

Experimental Procedure:

1. The pressure vessel is charged with silicone or paraffin oil and test particles. Test particles are Rhodamine-B (Rh-B) marked, polymethylmethacrylate

(PMMA) particles (see Figure 108) which show Laser Induced Fluorescence (LIV) under the influence of 532nm (green) light.

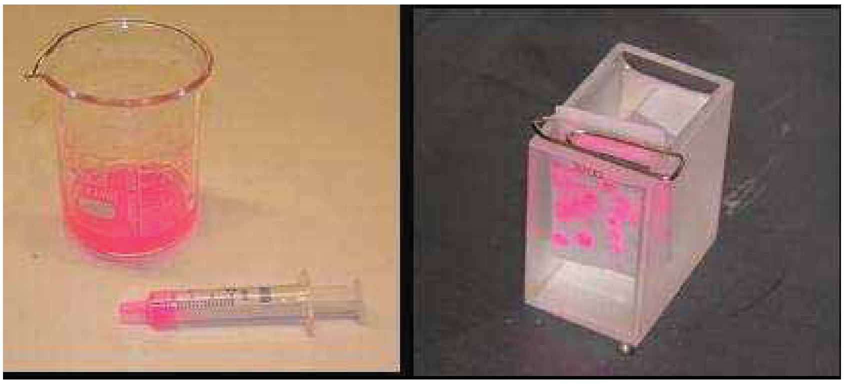
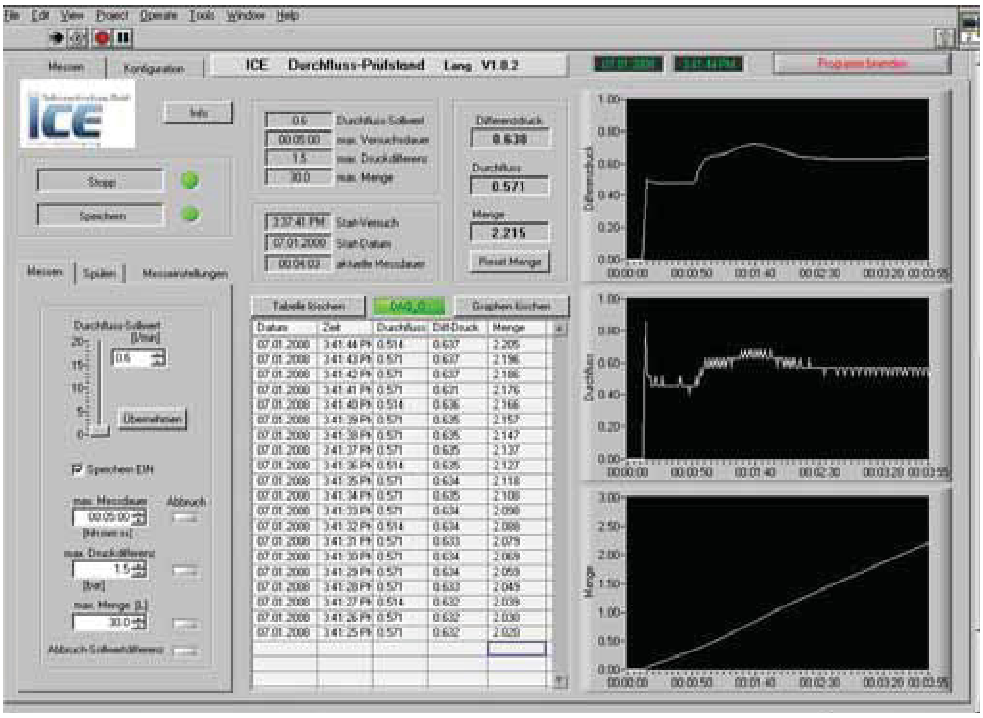


Figure 108: Rh-B marked PMMA particles in oil dispersion (left) and on filter fibre sample (right).

2. Pressurized air is provided by the compressor. Initially it is used to supply the air-pressure mixing facility at the pressure vessel. The vigorous mixing provides a homogenized oil-particle dispersion. After disabling the air-pressure mixing facility, the magnetic mixing device is activated. It ensures smooth, continuous mixing conditions throughout the entire experiment.
3. Pressurization of the pressure vessel up to p_v=6.0bar.
4. By-pass valve is switched and main flow valve is opened. Particle laden oil flows via flow measurement and flow regulation device into sample by-pass until steady flow conditions are reached. The process is monitored on-line from the beginning. Results are plotted and processed via the LabVIEW® [83] based, graphical-user interface (see Figure 109).



0.6 Durchfluss Sollwert
00:05:00 max. Versuchsdauer
1.5 max. Druckdifferenz
30.0 max. Menge

3:37:41 PM Start-Versuch
07.01.2008 Start-Datum
00:04:03 aktuelle Messdauer

Differenzdruck 0.638
Durchfluss 0.571
Menge 2.215

Durchfluss-Sollwert [l/min] 0.6
☑ Speichern EIN
max. Messdauer 00:05:00 [hh:mm:ss]
max. Druckdifferenz 1.5 [bar]
max. Menge [L] 30.0

Datum	Zeit	Durchfluss	Diff-Druck	Menge
07.01.2008	3:41:44 PI	0.514	0.637	2.205
07.01.2008	3:41:43 PI	0.571	0.637	2.196
07.01.2008	3:41:42 PI	0.571	0.637	2.186
07.01.2008	3:41:41 PI	0.571	0.631	2.176
07.01.2008	3:41:40 PI	0.514	0.636	2.166
07.01.2008	3:41:39 PI	0.571	0.635	2.157
07.01.2008	3:41:38 PI	0.571	0.635	2.147
07.01.2008	3:41:37 PI	0.571	0.635	2.137
07.01.2008	3:41:36 PI	0.514	0.635	2.127
07.01.2008	3:41:35 PI	0.571	0.634	2.118
07.01.2008	3:41:34 PI	0.571	0.635	2.108
07.01.2008	3:41:33 PI	0.571	0.634	2.098
07.01.2008	3:41:32 PI	0.514	0.634	2.088
07.01.2008	3:41:31 PI	0.571	0.633	2.079
07.01.2008	3:41:30 PI	0.571	0.634	2.069
07.01.2008	3:41:29 PI	0.571	0.634	2.059
07.01.2008	3:41:28 PI	0.571	0.633	2.049
07.01.2008	3:41:27 PI	0.514	0.632	2.039
07.01.2008	3:41:26 PI	0.571	0.632	2.030
07.01.2008	3:41:25 PI	0.571	0.632	2.020

Figure 109: Screenshot of LabVIEW® based graphical user interface of measurement and control software.

5. Sample by-pass valve is switched and oil flows past sample, passing up-stream and down-stream pressure detectors.
6. Experiment is monitored and stopped when prescribed, cumulative oil amount has passed the fibre sample.
7. Particle laden fibre sample is transferred to particle distribution detection facility.

8.3.2 The Particle Distribution Detection Facility

The Particle Distribution detection facility has been designed to determine the three dimensional test particle distribution in filter fibre samples coming out of the oil-fibre test facility. In combination with the optical evaluation algorithm (see chapter 8.3.3) a fully digitalized, 3D image of a 2.0mm*2.0mm*0.3mm region within a particle laden filter fibre sample can be obtained. The facility consists of three main components:

- Particle Image Velocimetry (PIV) camera with bellows (see Figure 110): LaVision ®, Image Intense. Maximal optical enhancement: 1:4.33. Resolution: ≤0.77Pixel/µm. Software: DaVis 7.0®.

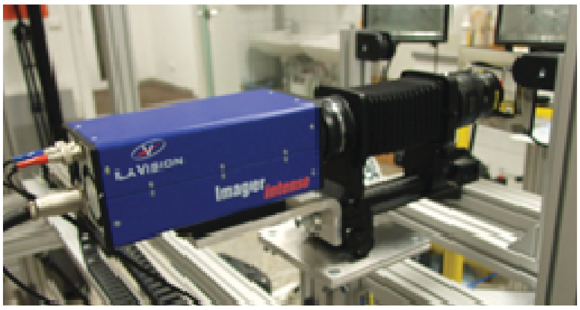
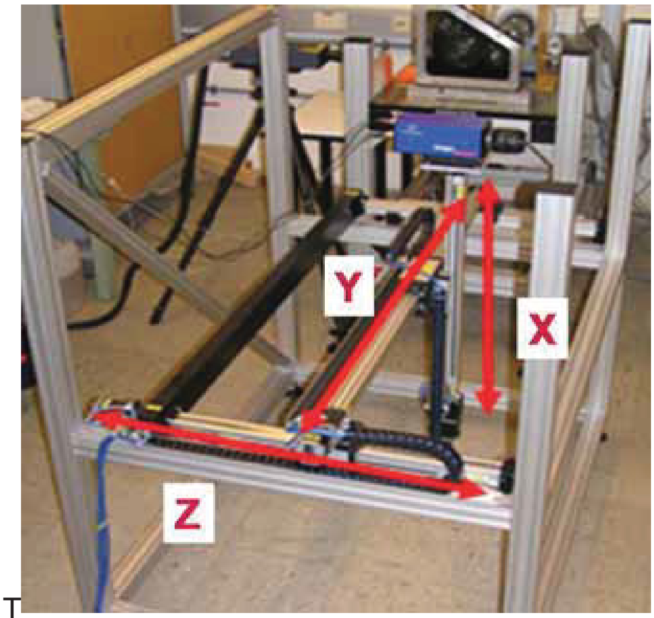


Figure 110: LaVision®, Image Intense, Particle Image Velocimetry camera with optical bellows.

- Drive Set: Systec®, SD Standard (see Figure 111). Three degrees of motional freedom. Minimal step width: 10µm. Software: Motion Basic®.



T

Figure 111: Systec ® DriveSet SD Standard with three degrees of motional freedom and PIV camera placed on it.

- Laser: New Wave Research®, Solo PIV (see Figure 112). Light frequency: 532nm; Pulse rate: 15Hz; Pulse length 3 – 5ns. Beam width: 3mm.

Figure 112: New Wave Research® laser with dispersive lens.

Measurement principle:

The experimental evaluation procedure is based upon the principle of *laser fluorescence macroscopy* [84]. An oil drenched, PMMA particle laden filter fibre sample is positioned in front of the camera. The camera uses minimal light exposition duration and the smallest f-number provided by the zoom lens. Thus the depth of focus is reduced to a minimum. The resulting picture will clearly distinguish between objects hit by the focal plane and the rest. As a consequence, the sharpness of the depicted objects will provide a quantifiable measure of distance to the focal plane. To obtain 3D information on the particle distribution within a fibre, the camera's focal plane is moved step by step through the relevant regions of the fibre as the camera takes series of depth images. As shown in Figure 113, along with a sketch of the basic measurement principle, the task also involves achieving a distinct, optical differentiation between particles and the surrounding fibre structure. Here the choice of test fluid in the oil-fibre test facility is decisive. The fluid is to be chosen such that an optimal match of indices of refraction between the oil and the fibre samples is achieved. The flow conditions are then adapted accordingly. A good, refractive index match leads to a semi-transparency of the filter fibre. Light penetration is thus enhanced. A second, decisive factor in highlighting the particles lies within the choice of test particles, in combination with the laser. When pictures are taken, the synchronized, 532nm laser shoots at the sample and the Rh-B marked particles start fluorescing. Resulting grayscale images show hardly any sign of the fibre structure, but only more or less sharply depicted, glowing particles. Figure 114 presents an exemplary picture.

At least four arbitrary x-y test positions are chosen for each sample (see Figure 115). At any one of the positions a series of up to 50 pictures is taken with the camera being moved by steps of 10µm-30µm within the z direction.

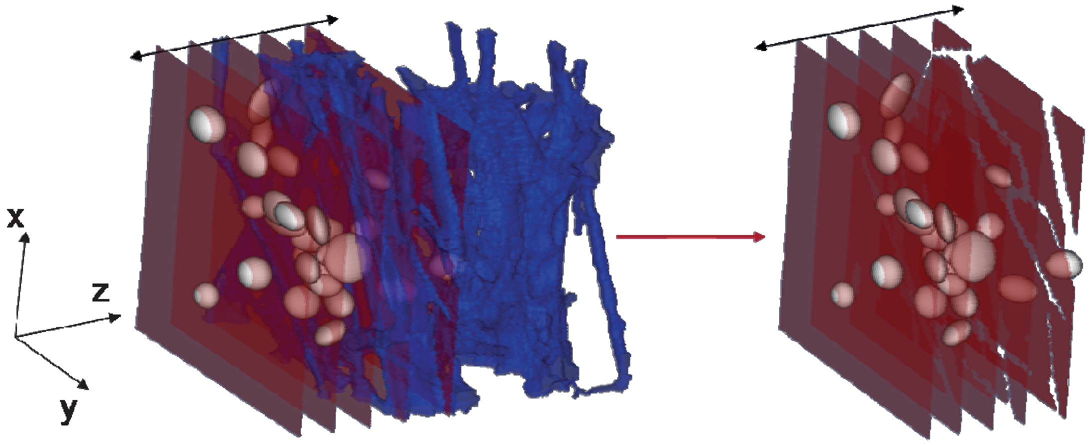


Figure 113: Measurement principle behind the *laser fluorescence macroscopy* method to determine 3D particle distribution. Focal plane (red) is moved through the fibre structure (blue) and particles (gray) are highlighted.

Figure 114: Exemplary picture of particle distribution within 2mm*2mm image region of filter fibre sample. Particles glow under influence of laser light. The closer to focal plane, the sharper the individual particle image becomes.

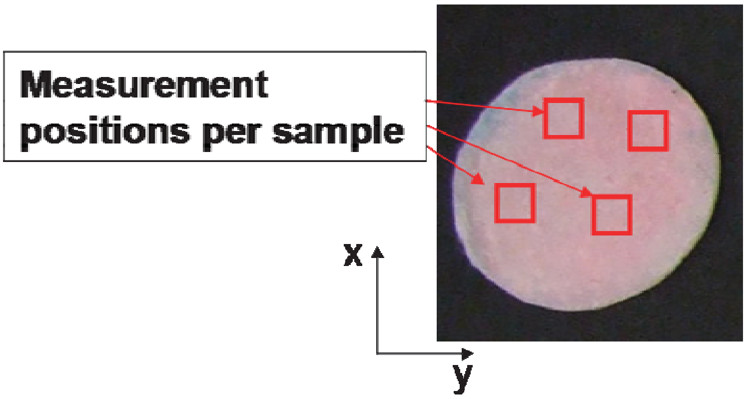


Figure 115: Fibre sample, D_S=2.5cm laden with test particles. Example of distribution of 4 measurement positions.

Experimental Procedure:

1. After being exposed to the PMMA particle laden flow in the oil-fibre test facility, the filter fibre sample is inserted into a sample holding device (see Figure 116) and placed in front of the PIV camera.

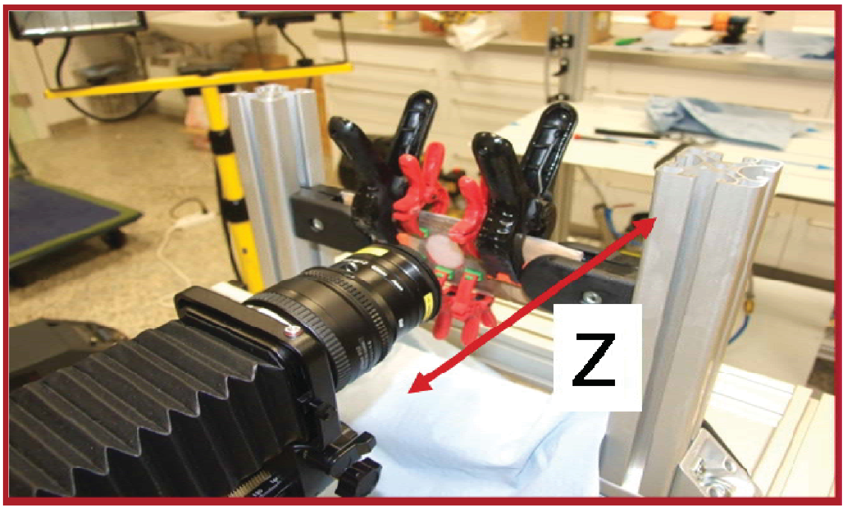


Figure 116: Sample holding device with fibre sample, positioned in front of camera which shifts position in z direction.

2. The PIV camera, the Drive Set and the laser are activated and programmed via a LabVIEW® based, graphical user interface. Thus the following parameters are set: laser intensity, laser beam frequency, light exposition time, initial camera position and Drive Set motion parameters to conduct scanning routine.
3. The automated evaluation run is being conducted: a) The Drive Set moves into starting position. b) The PIV camera initiates light exposition of the image. c) Laser shoots synchronously. d) Camera finishes light exposition of the image. e) The Drive Set sets the camera to its new position by shifting into z direction.
4. With each z-shift a picture is taken with the laser shooting synchronously. Up to 50 pictures are taken at any fixed x-y position. The pictures are digitally stored and written out as JPEG files at the end of the evaluation run.
5. The JPEG file series are passed on to the optical evaluation algorithm which determines the full, local 3D particle distribution.

8.3.3 The Optical Evaluation Algorithm

An optical evaluation algorithm to automatically recognize 2D dirt particle shapes and to reconstruct them as 3D objects has been created using MatLab®. The code is suitable to fully resolve and determine the dirt particle distribution from any given series of gray scale images yielded by the dirt particle distribution detection facility. It basically consists of three phases:

- 2D shape recognition:
  A standard application within the additional MatLab® image processing toolbox. The application has been extended by the ability to automatically remove non-isotropic, back ground light effects (with courtesy to Prof. Paul O'Leary, Institute of Automation, University of Leoben). Figure 117 shows an exemplary result of the mere 2D shape recognition function. Thereby five nuts are recognized as individual objects, counted and processed in terms of pixel-area.

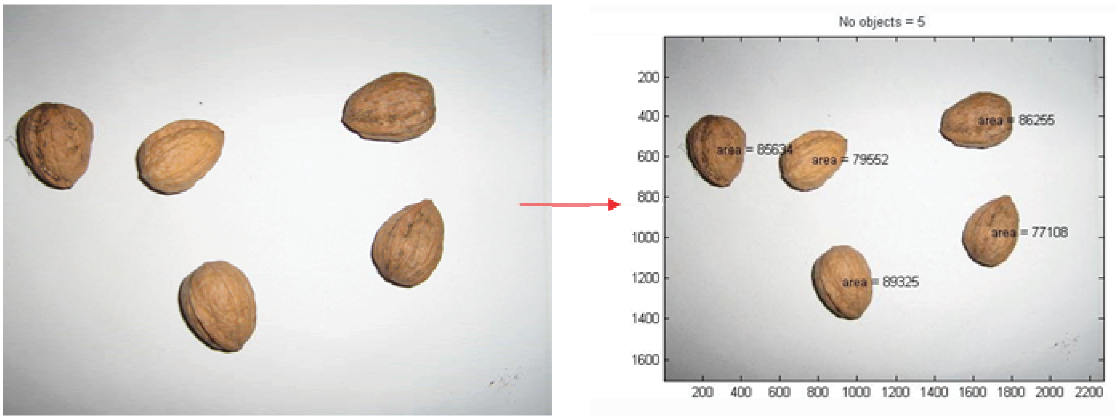


Figure 117: Exemplary result of 2D shape recognition function. Five nuts are recognized as individual objects, counted and their pixel area is evaluated.

- Evaluation of object specific sharpness value:
  Several focus measures, frequently used in *Multi Focus Image Fusion* procedures [85] have been implemented to assign specific sharpness values to the 2D objects. The focus measures, Energy of Image Gradient (EOG) (Equ.281) and Sum Modified Laplacian (SML) (Equ.282), [86] proved to be well suited for the given task.

$$EOG(x,y) = \left(f_x^2 + f_y^2\right) \qquad (281)$$

Here f_x and f_y are the local, spatial gray scale derivatives.

$$SML(x,y) = \sum_{i=x-1}^{x+1} \sum_{j=y-1}^{y+1} \left(\nabla^2{}_{ML} f(x,y) \right) \qquad (282)$$

Whereby the Sum Modified Laplacian operator is used according to [86]. Figure 118 shows a demonstration of how the EOG highlights sharply depicted objects and how it simultaneously forfeits blurred ones.

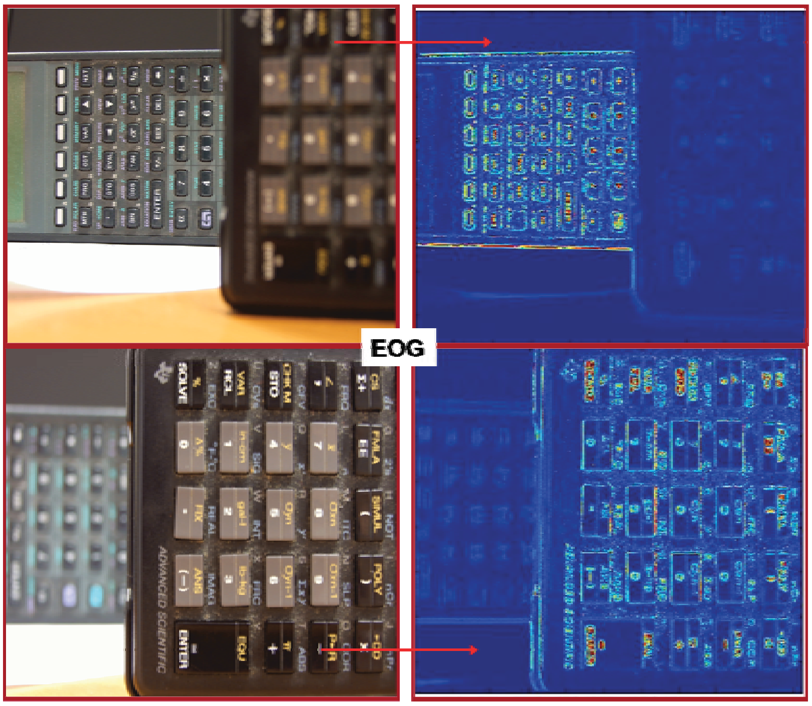


Figure 118: Example of the basic functionality of the EOG focus measure. It highlights regions of high grey-scale gradient (sharply depicted regions) and forfeits regions of smaller grey-scale gradient (blurred regions).

- Construction of digital 3D objects in vector space:

  In a third step, the recognized, focus measured 2D objects within the individual depth images are compared. Shapes belonging to one and the same, real life particle will feature a centre-to-centre distance which is notably smaller than the mean shape radius. Thus 2D shapes within different planes are recognized to belong to each other. Then the centre point of the shape with the highest focus measure is stated to be the 3D object's centre position. Accordingly the object's diameter is stated to be the centre-shape's medium diameter. The spherical dirt particle can then be positioned in 3D vector space. Figure 119 presents a sketch of the principle construction procedure.

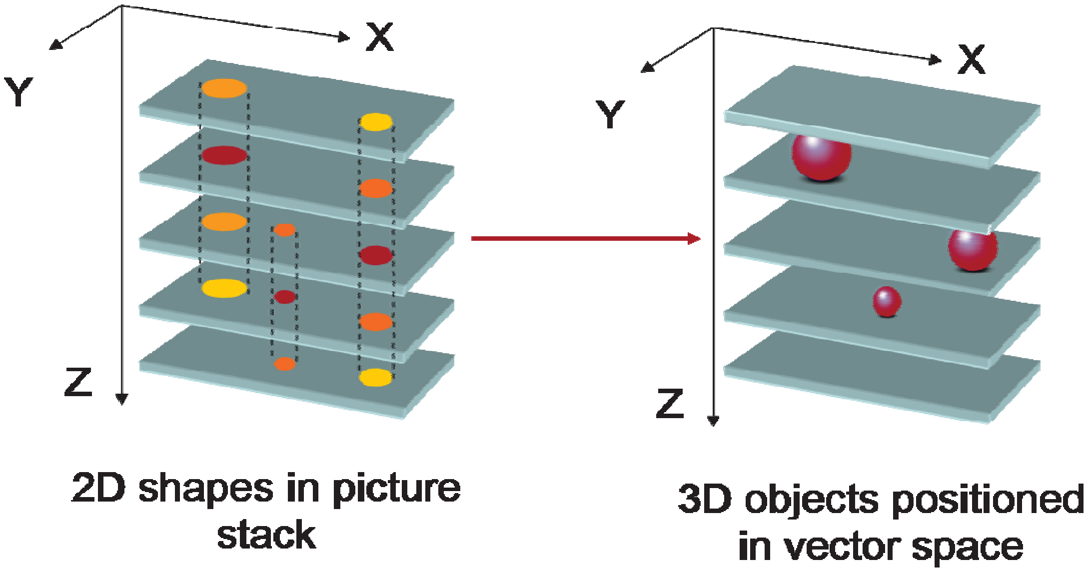


Figure 119: Methodology behind the construction of 3D objects, positioned in vector space (right) from 2D shapes (left) belonging together. Coloration of the 2D shapes corresponds to the intensity of the previously assigned focus measure.

Figure 120 shows a full, exemplary 3D particle distribution reconstruction of a probed 2.0mm*2.0mm*0.5mm region belonging to a fibre sample.

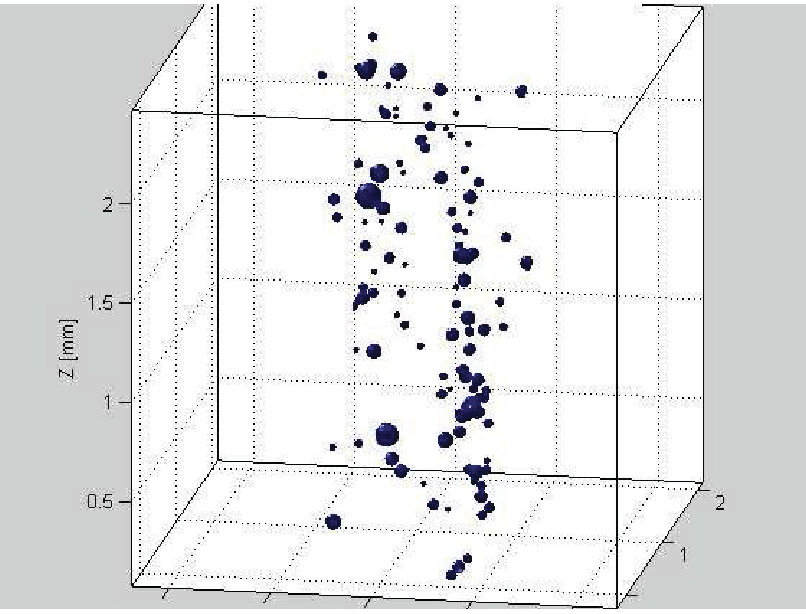


Figure 120: Final result of reconstructed picture series. 3D image of local test particle distribution.

The algorithm is capable of providing visualized results as well as numerical results like: a count of objects per depth plane, relative covered picture area per depth plane or medium object diameter of objects in depth plane. Figure 121 shows an exemplary, numerical evaluation.

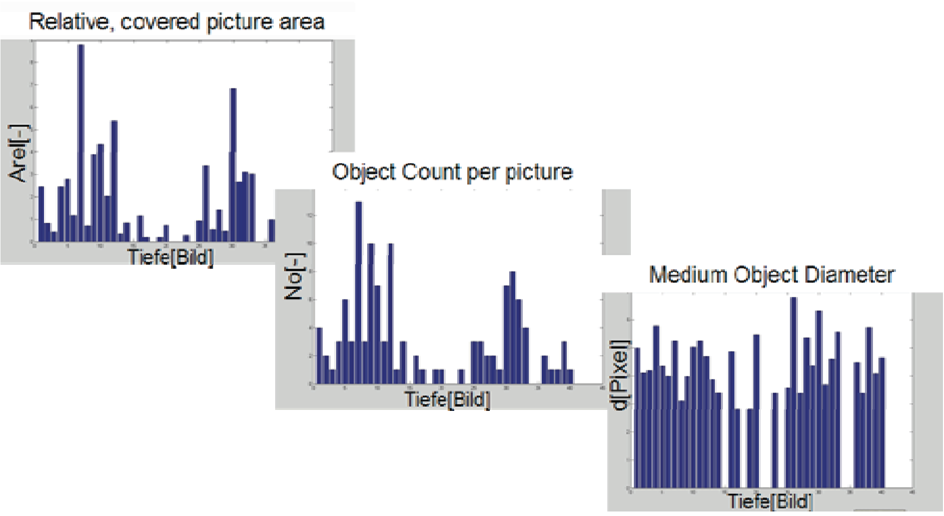


Figure 121: Exemplary, numerical evaluation of 3D test particle distribution. Output parameters are: relative covered picture area, object count and medium object diameter per depth plane (picture).

8.3.4 Qualitative Verification of the 3D Reconstruction Method

The functionality of the entire 3D reconstruction method has been qualitatively verified. To do this a simple experiment was set up. Five macroscopic spheres with diameters between 3cm and 8cm were hung on black thread and placed into an otherwise empty, black box (see Figure 122). Then the microscopic imaging procedure was duplicated as well as possible on the macroscopic level. An ordinary, digital camera was used and its depth of focus was switched to a minumum. As pictures were taken, the camera was moved in steps of 1cm, such that the focal plane was drawn through the spheres. The resulting gray scale images were then processed by the evaluation algorithm. A 3D reconstruction of the scene, created out of a series of 2D pictures was created.

Figure 123 presents a direct comparison of an ordinary 2D picture of the spheres and the three-dimensionalized, digital reconstruction. Due to the good agreement of results, a verification of qualitative measurement procedure functionality is considered to be achieved.

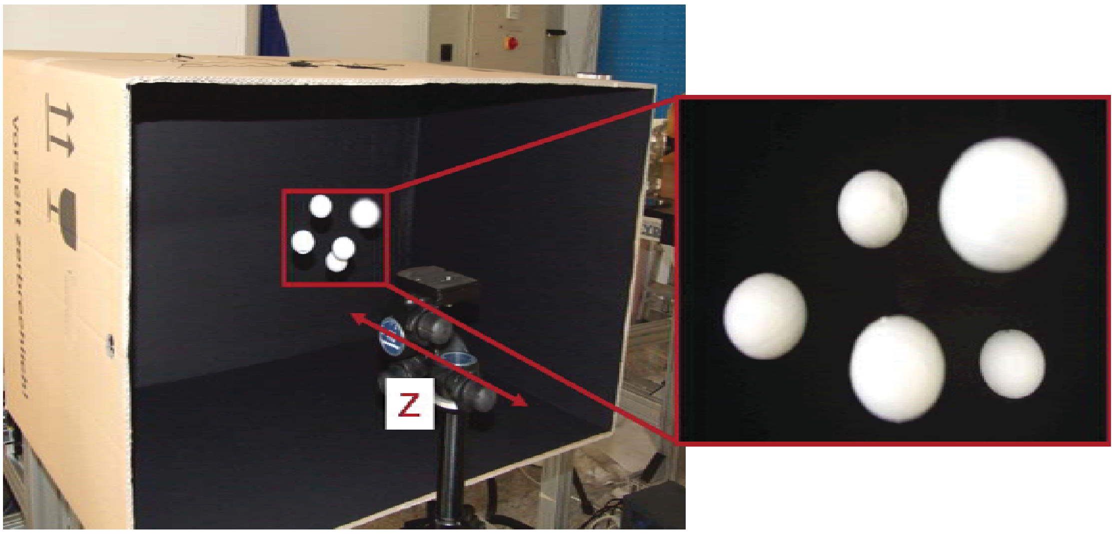


Figure 122: Set up of a simple scene to verify the functionality of the 3D particle reconstruction method. Five white spheres are hung on black thread against the black background of a simple box. An ordinary, digital camera is placed in front.

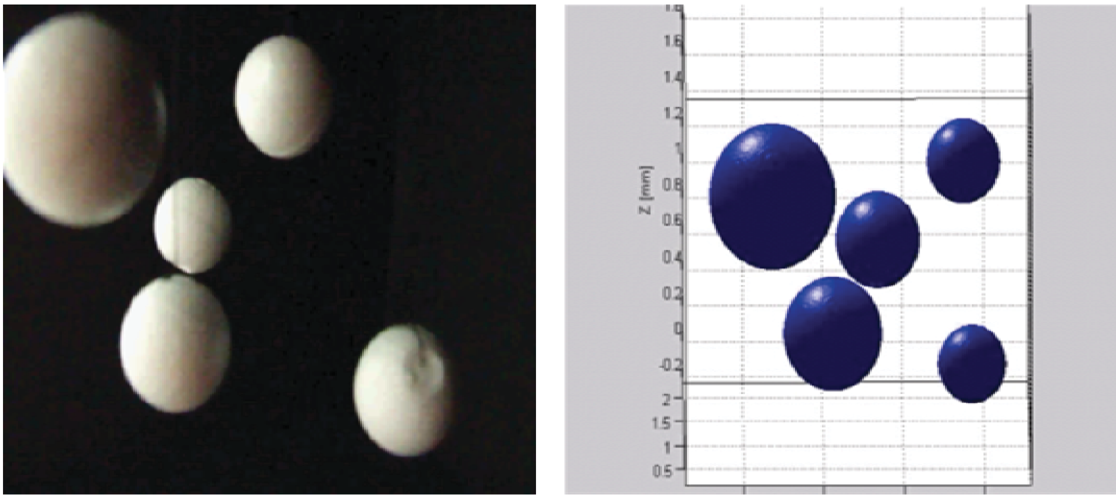


Figure 123: Direct comparison of an ordinar 2D picture taken from the five spheres (left) and a screen shot taken from the optical, three-dimensionalized reconstruction result yielded by the Matlab® algorithm.

8.3.5 Two Modes of Measurement

Based on the three experimental/evaluation tools: the oil-fibre test facility, the particle distribution detection facility and the evaluation algorithm, two optional measurement modes can now be performed:

- The particle distribution detection mode
- The filter fibre efficiency mode

Particle distribution detection mode

The particle distribution detection mode is what the equipment has been originally designed for. It constitutes a new procedure to obtain extensive information on 3D particle distributions in filter fibre samples. Its principles and the experimental approach have already been presented in chapter 8.3.3 and 8.3.6. It is important to note however, that for the particle distribution mode, poly disperse particle fractions are used in the oil-fibre test facility. Figure 124 shows an over view of the procedure.

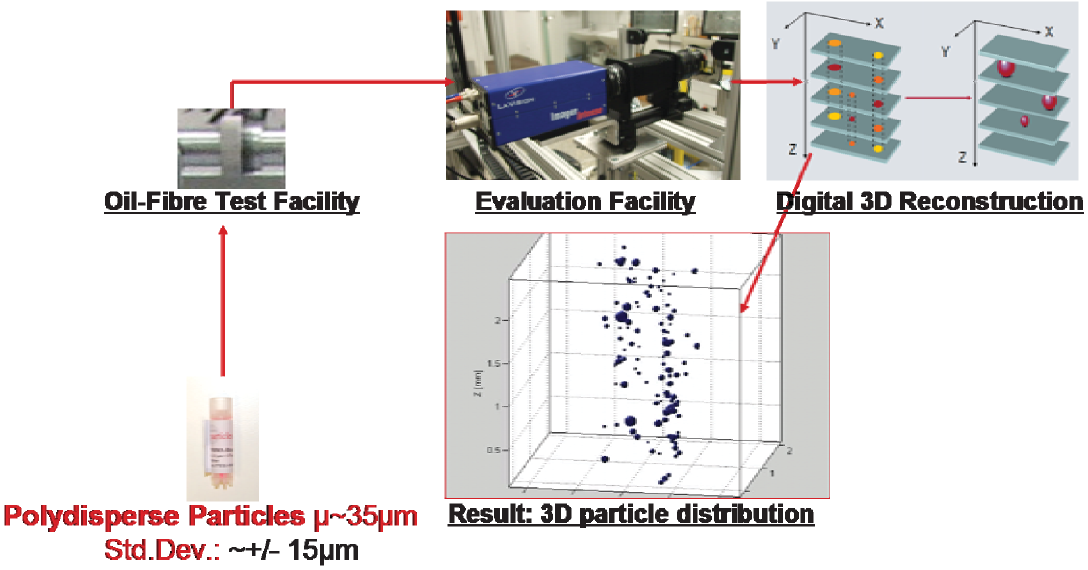


Figure 124: Overview of the particle distribution measurement mode.

Filter fibre efficiency mode

By slightly adjusting the measurement procedure used for the particle distribution mode, the filter fibre efficiency mode was designed. It basically serves as a tool for obtaining experimental verification of simulation runs concerning filter fibre efficiency curves. Experimental verification, concerning pressure drop, can be obtained via simply using the oil-fibre test facility, which is part of both measurement modes.

In the case of the filter fibre efficiency mode, mono-disperse fractions of test-particles are used for the oil-fibre test facility runs. The optical evaluation method is simpler here than for the distribution mode. The 3D distribution is no longer relevant and only the amount of particles, entangled in the fibre n_f, as compared to the total amount of particles in the fluid n_t, is of concern. Thus the depth of focus is switched to a maximum by increasing the f-number of the lens and only one picture per fibre sample is necessary. No request for reconstructing the entire 3D particle distribution has to be sent to the evaluation algorithm, since it merely has to automatically count the individual particle objects, detected in the fibre. Since the tests are conducted for fibre test samples and for samples of *absolute* filter material in parallel, the two results have to be compared in order to obtain the value E. Experiments and evaluations have to be conducted for all available particle fractions (10µm, 25µm,...100µm) in order to retrieve information on the efficiency curve. Figure 125 shows an overview of the entire measurement procedure.

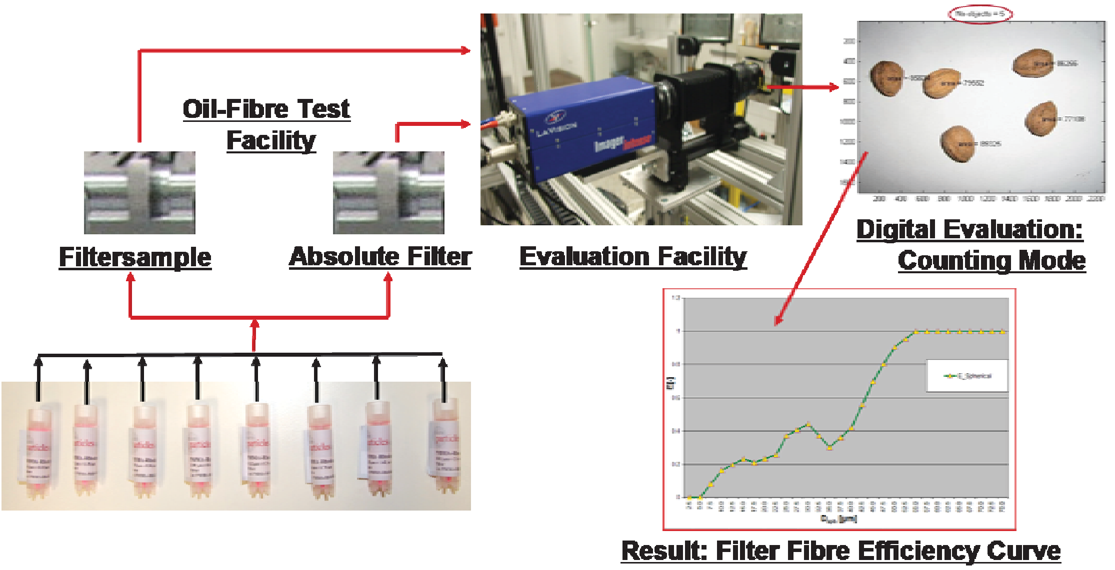


Figure 125: Overview of the filter fibre efficiency measurement mode.

8.3.6 Experimental Verification of Simulation Results

Using the presented laboratory equipment along the guidelines, developed for the oil-fibre test facility, the particle distribution detection mode and the fibre efficiency measurement mode, the following simulation parameters can be experimentally verified:

- Pressure drop over filter fibre: Verification by use of the oil-fibre test facility.
- Filter fibre efficiency curve: Verification by use of the fibre efficiency measurement mode.
- Particle penetration depth: Verification by use of the particle distribution detection mode.

During the course of this thesis, only the first two parameters have been thoroughly investigated in terms of verification while the particle penetration depth results have passed several plausibility checks. The particle distribution detection mode is hereby rather proposed as a stand-alone method to characterize particle-fibre interaction, than to merely serve as a verification tool.

8.3.6.1 Pressure Drop

Pressure drop over the filter fibre against flow rate variations can be monitored via the oil-fibre test facility. The test results can then be directly compared to CFD runs, conducted on digitally reconstructed geometries of the same filter fibre material. So far, an accordance of +/-10% between lab tests and CFD runs has been achieved. Whereby the CFD calculations tend to slightly underestimate the measured pressure drop. This is attributed to the fact that, even though the fibre geometry is reconstructed with a high degree of detail, surface structures below 1µm can still hardly be resolved. After all the fibre is depicted with a slightly decreased surface roughness causing smaller pressure drop. In terms of particle-fibre interaction this effect can be compensated: particle-fibre adhesion is simply increased. Figure 126 shows the comparison of two materials' (A43 and A55) pressure curves against volumetric flow rate. The curves are directly compared to the corresponding simulation outcome. The simulation results were obtained by calculating the averaged pressure drop over four arbitrarily chosen, reconstructed fibre sections, of the kind shown in Figure 127. Those samples represent a 140µm*140µm*(800µm-1000µm) portion of the real life sample and encompass the entire fibre thickness. Simulations and experiments were conducted under the following conditions: u_f=0.0067m/s-0.0204m/s, μ_f =1.93*10^{-3}Pas, ρ_f=800kg/m³.

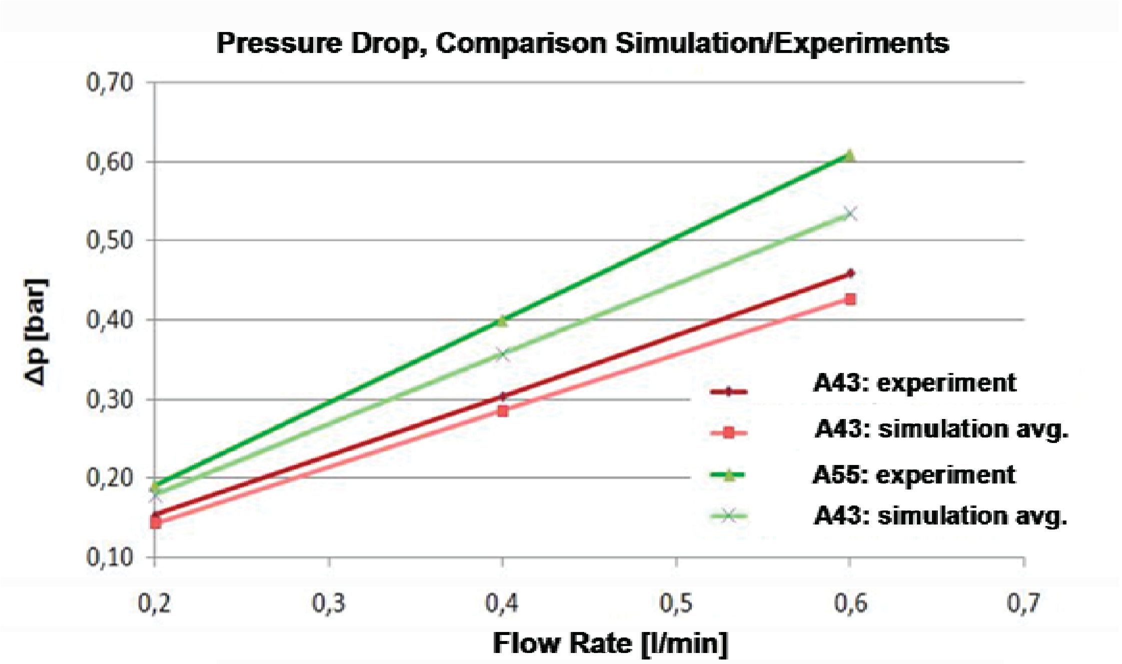


Figure 126: Pressure drop over filter fibre thickness against volumetric fluid flow rate. Absolute flow area of filter element: $A_f=4.91*10^{-4}m^2$. The values of A43 (red) and A55 (green) were compared to the corresponding simulations. Deviations amount to ≤10% and are explained by incomplete resolution of the surface roughness.

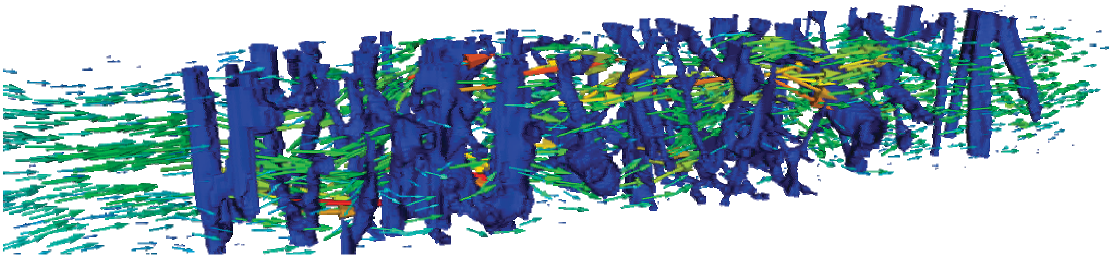


Figure 127: Screen shot of flow simulation to obtain pressure drop over filter fibre medium. 140µm*140µm*(800µm -1000µm) portions of the fibre structure (blue) were reconstructed. Four regions within the real life sample were tested and the results have been averaged.

8.3.6.2 Filter Fibre Efficiency Curve

Filter fibre efficiency curves can be experimentally obtained by applying the fibre efficiency measurement mode. The procedure is quite time consuming since the following steps have to be conducted for each particle size class of interest:

- Loading of the fibre sample with test particles in the oil-fibre test facility.
- Loading of a reference *absolute* filter in the oil-fibre test facility.
- Result evaluation by applying the optical particle detection facility plus the algorithm at 6 to 8 positions on both filter samples.

The procedure to obtain corresponding simulation results is quite similar, yet much more time efficient. For each particle size class i, a fixed amount of particles $n_{t,i}$ is injected during transient flow simulation within the fibre geometry. When kinetic particle energy in the system has dropped to a steady state, the final result is evaluated by comparing the remaining particles $n_{r,i}$ to $n_{t,i}$ so that:

$$E_i = \frac{n_{r,i}}{n_{t,i}} \qquad (283)$$

An extensive experimental and computational analysis of the filter fibre efficiency characteristic of the A43 filter material was conducted. Simulations and experiments were set up with the following conditions: u_f=0.0136m/s, μ_f=1.93*10^{-3}Pas, ρ_f=800kg/m³, ρ_p=1500kg/m³, no adhesional effects. The simulations were conducted on 200µm*200µm*300µm portions of the fibre structure, which represent only about 1/3rd of the total filter thickness. In order to compensate, the following procedure was used:

- The filter region to be modelled was selected such that it represents the average porosity and filter diameter throughout the filter depth as well as possible.
- It was supposed that the entire sample is composed of j=3 consecutive layers of representative material, each featuring a single fibre efficiency of $E_{i,s}$. The total, particle class specific fibre efficiency could then be calculated as:

$$E_i = 1 - \left(1 - E_{i,s}\right)^3 \qquad (284)$$

- Since the total fibre efficiency was obtained experimentally, the single fibre efficiency of the shorter fibre piece, which is comparable to the simulated piece, was obtained as:

$$E_{i,s} = 1 - \sqrt[3]{(1 - E_i)} \qquad (285)$$

A direct comparison of the simulated results and of the reduced, experimental results is shown in Figure 128.

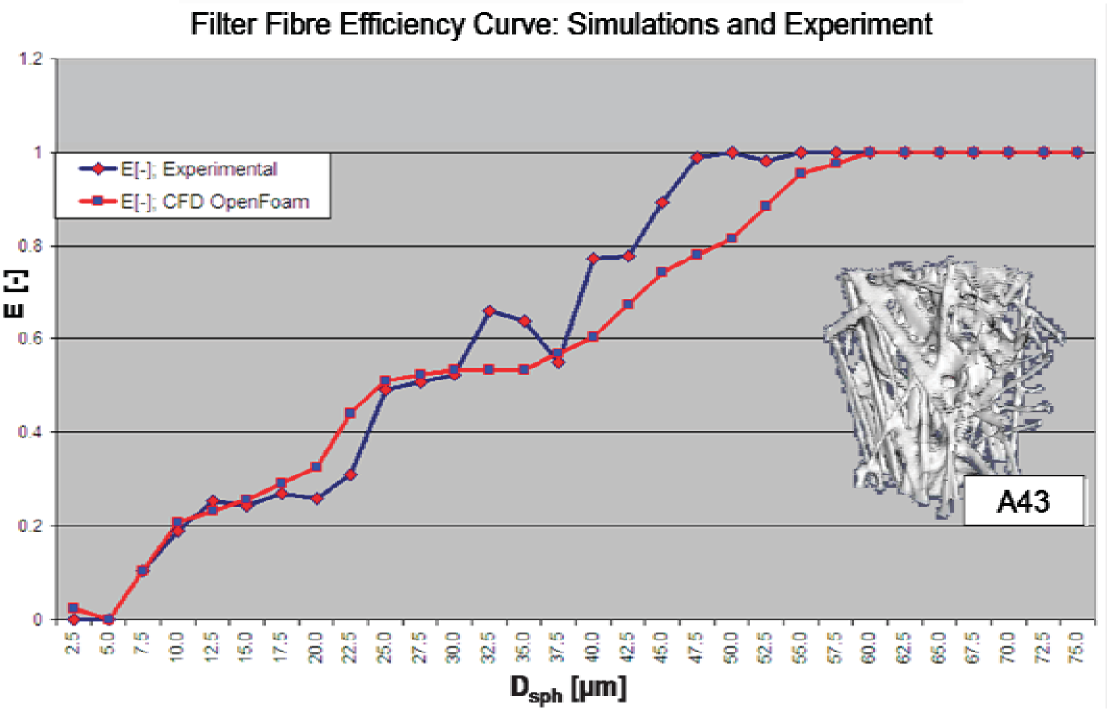


Figure 128: Comparison of simulated (red) and experimentally derived (blue) filter fibre efficiency curve for material A43. An exemplary screenshot of the reconstructed piece of fibre material is shown on the right.

The experimental results, shown in Figure 128, show very good agreement with calculations. Thereby the agreement is better for smaller particle diameters than for larger ones. This effect is attributed to the following facts:

The amount of pores, smaller than the actual particle diameter is decisive for deposition efficiency, especially if adhesional effects are negligible. The simulated portions of the fibre, feature a cross-section of 200μm*200μm. Hence a maximum of 100 pores with diameter D_{pore}~20μm can be theoretically considered per cross section slice. This constitutes a good base for averaging results over the entire filter medium. As a consequence, the deposition efficiency for particle diameters $D_{sph} \leq 20$μm is predicted very well. For larger particles the prediction is slightly worse. For pores with diameter D_{pore}~60μm a maximum of only 10 theoretical pores can be considered per cross section slice. Consequentially the statistical base of the results, as well as the agreement with experiments is worse.

Still the entire verification procedure is hereby considered to be successful and the high quality of solver results is highlighted. For further result improvement, larger fibre portions are to be examined in the future.

9 Application and Results: Filter Fibre Engineering

In practice the new CFD solver will rather be applied on realistically reconstructed, digitalized fibre geometries than on simplified models. The means of result evaluation are manifold. Important insight into qualitative and quantitative aspects of process parameter behaviour, over filter life time can be gained. Two of the most interesting results are filter fibre efficiency and particle penetration depth curves.

A typical example of solver application would be very similar to the procedure used in chapter 8.3.6.2. It would involve taking several, microscopic, digitally reconstructed portions out of a filter fibre sample in order to conduct deposition and penetration analysis for the entire range of relevant particle sizes. The results would then be averaged in order to obtain an idea of the basic particle deposition characteristic of the fibre.

All relevant solver output data (such as pressure/velocity field information, particle positions and orientation) is given in text files. Using Python® [28] scripting, several evaluation modules have been created. Those modules use the text-based output data to extract result parameters such as filter fibre efficiency, particle penetration depth, plugged flow channel volume or kinetic particle energy.

To give an impression of what the solver can do as of now, several application examples are presented in the following.

9.1 Comparison of Material with/without Adhesional Effects

The first example is supposed to point out the potential of the solver in terms of material design. A digitally reconstructed fibre geometry (see Figure 128) was tested in interaction with clouds of arbitrarily shaped, non-spherical particles with a half-axe standard deviation of $\alpha_{ax}=\sqrt{5}/9$. Filter fibre efficiency curves were evaluated for two cases of fibre quality. Case a) features fibres that do not show any particle-fibre interaction, while case b) has the very same fibre topology, but includes strong particle-fibre adhesion. The results, seen in Figure 129, show a dramatic increase in filter efficiency (at equivalent pressure drop) if adhesional effects (and not just pore size distributions) are considered in material selection. While real-life adhesional effects might not be quite as strong as those in the simulation, the potential of such effects is pointed out pretty well.

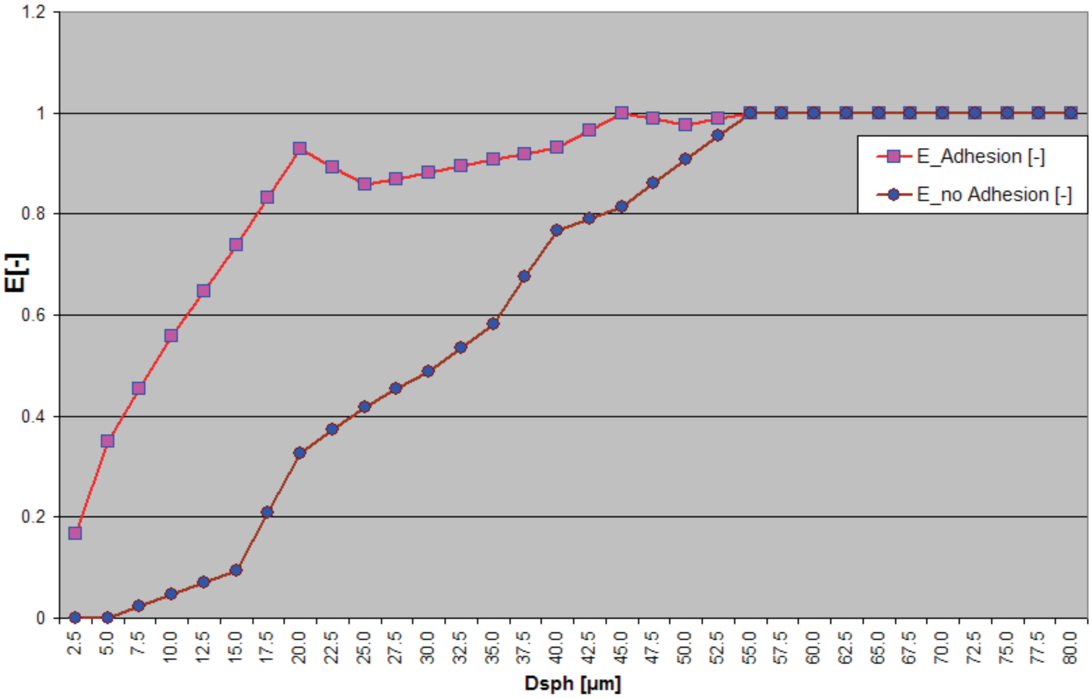


Figure 129: Comparison of filter fibre efficiency curves for one and the same fibre geometry. Case a) (brown) is calculated without adhesional effects and case b) is calculated by initiating strong adhesion. Test case parameters: geometry dimensions:200µm*200µm*300µm; fluid: u_f=0.01m/s, ρ_f=800kg/m³, ν_f=4*10^{-4}m²/s; particles: ρ_p=3000kg/m³, α_{ax}=√5/9.

9.2 Comparison of (Non-)Spherical Particle Filter Fibre Efficiency

The second example shows the importance of considering non-spherical particle shape effects. Filter fibre efficiency curves, using spherical and non-spherical particles on one and the same, non-adhesional filter fibre medium (see Figure 128), have been evaluated. The ellipsoids feature a half-axe standard deviation of α_{ax}=√5/9 and are always compared to their mass equivalent, spherical counterparts. The results, seen in Figure 130, point out the importance of considering particle shape effects. Relevant deviations in fibre efficiency are apparent.

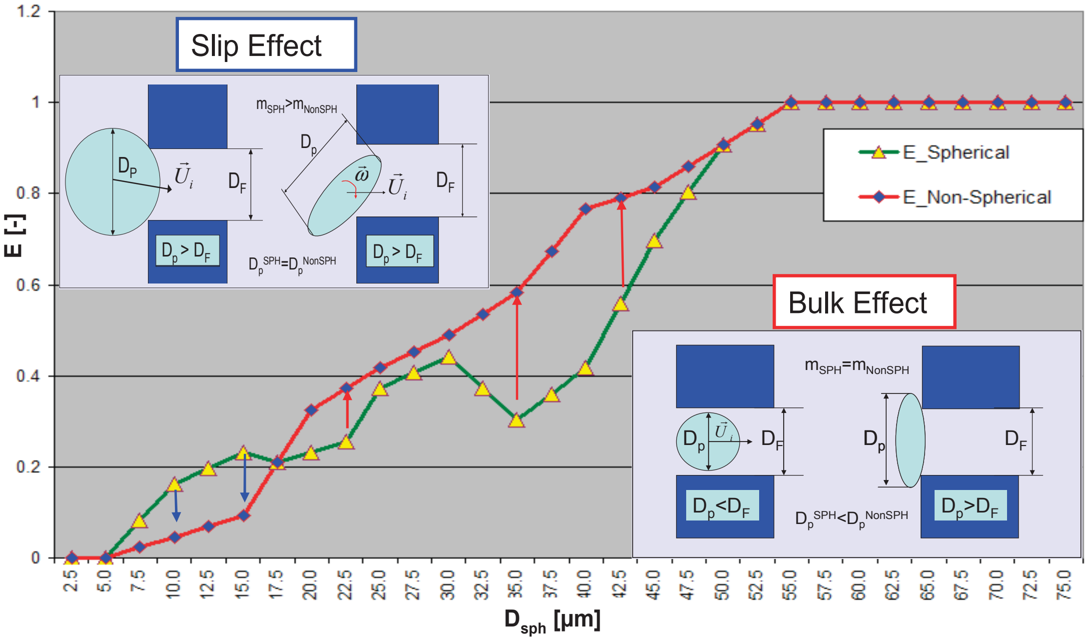


Figure 130: Comparison of filter fibre efficiency curves for one and the same fibre geometry without adhesional effects (equivalent to chapter 9.1, case a)) . Case a) (green) is calculated with spherical particles and case b) (red) is calculated with non-spherical particles of mass equivalent spheres. Test case parameters match those, given for Figure 129.

Fibre efficiency results, shown in Figure 130, reveal that smaller ellipsoids (D_{sph}<17.5µm) are deposited less easily than mass-equivalent spheres, whereas larger (D_{sph}>17.5µm), non-spherical particles rather get stuck. This behaviour is both plausible and to be expected. Ellipsoids with smaller, angular relaxation times experience the *slip effect*. They align easily along stream lines, which enables them to slip through pores, that mass-equivalent spheres get stuck in (see Figure 9 , chapter 2.6.2). Larger ellipsoids with larger, angular relaxation times do not align as readily and experience the *bulk effect*. They get stuck in pores that their spherical counterparts just fit through (see Figure 10 and chapter 2.6.3). Between 30µm and 37µm spherical filter fibre efficiency decreases. This phenomenon can be attributed to increasing particle-particle interactions due to higher particle-particle hitting probability. Momentum transfer leads to blow-off effects.

9.3 Comparison of (Non-)Spherical Particle Penetration Depth

The third example, shown in Figure 131, gives an evaluation of particle penetration depth for the spherical and non-spherical case shown in chapter 9.2 (both without adhesional effects) and for the non-spherical case, featuring adhesional effects, which is presented in chapter 9.1.

Hereby the particle penetration depth is represented via the relative penetration measure P which attributes the value 1 to particles slipping through the fibre mesh, and 0 to particles caught right at entry into the fibre domain.

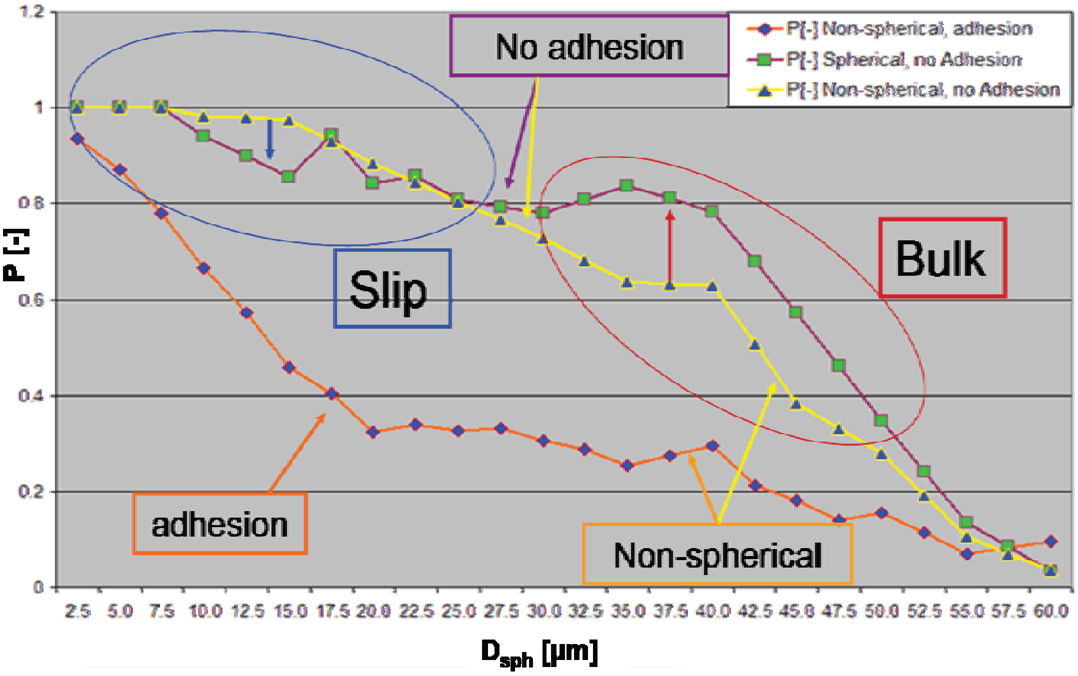


Figure 131: Comparison of particle penetration depth curves for one and the same fibre geometry (see Figure 128). Case a) (purple) is calculated with spherical particles and no adhesion, case b) (yellow) is calculated with non-spherical particles and no adhesion and case c) (orange) is calculated with non-spherical particles and with adhesion. Test case parameters match those, given for Figure 129.

As expected, the adhesional case (orange) shows remarkably lower P-values over the entire, relevant particle size regime, than the non-adhesional cases (yellow and purple). The relationship between particle penetration depth and filter fibre efficiency is demonstrated by comparing spherical and non-spherical, non-adhesional cases in Figures 130 and 131. For D_{sph}>20µm, non-spherical particles yield smaller penetration values than mass equivalent spheres. This corresponds with the fact, that filter fibre efficiencies for larger ellipsoids are higher than those for larger spheres.

9.4 Comparison of Fibre Materials: Ahlstrom A55 and Fulda A43

The fourth example, shown in Figure 132, is particularly interesting for filter fibre design applications. It gives the direct comparison of fibre efficiency characteristics of two similar but clearly distinguishable fibre materials: A55 and A43. A55 has slightly finer fibre structure and features a mean flow pore size MFP=22 +/-3µm while A43 has coarser fibres with MFP=31 +/- 3µm. Simulation runs were conducted under the conditions stated beneath Figure 129.

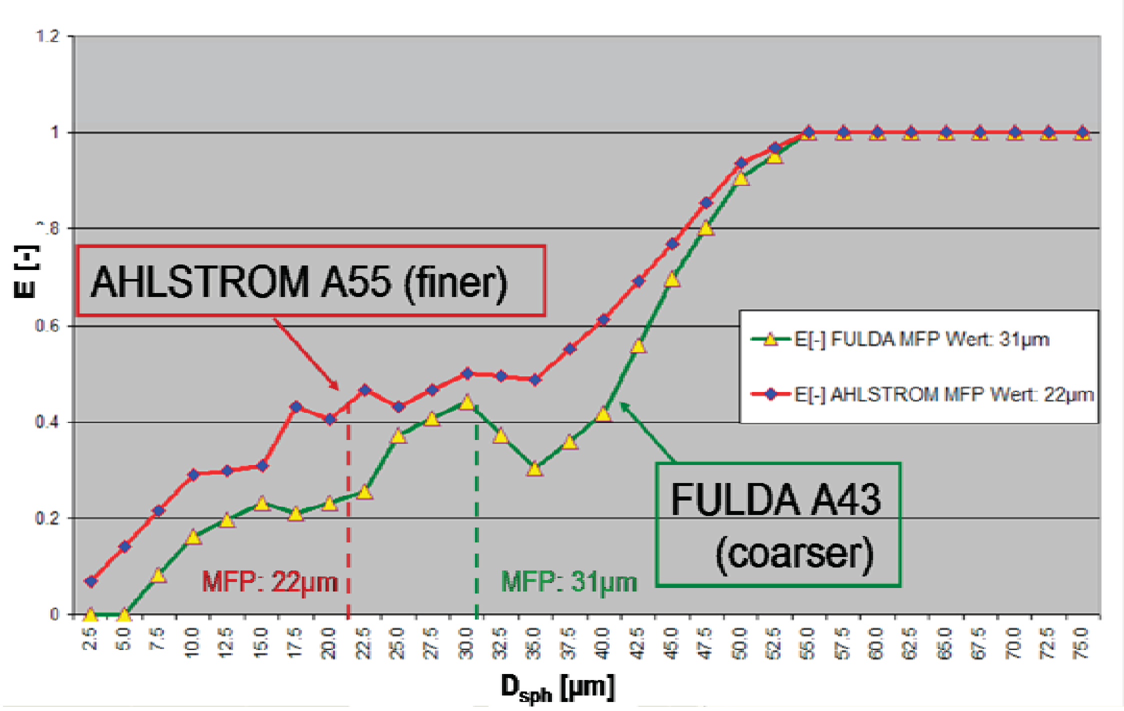


Figure 132: Comparison of filter fibre efficiency curves for different but similar materials: Ahlstrom A55 (red) with MFP=22µm and Fulda A43 (green) with MFP=31µm. Test case parameters match those, given for Figure 129.

The two efficiency curves, depicted in Figure 132, show very plausible results. Both curves are generally similar but efficiencies of the finer A55 material are larger than those of the coarser A43 throughout the entire, relevant particle size-spectrum. An evaluation of the corresponding particle penetration depth diagram, shown in Figure 133, underlines those results. Over the entire, relevant spectrum, particles penetrate slightly deeper into the coarser A43 material than into A55.

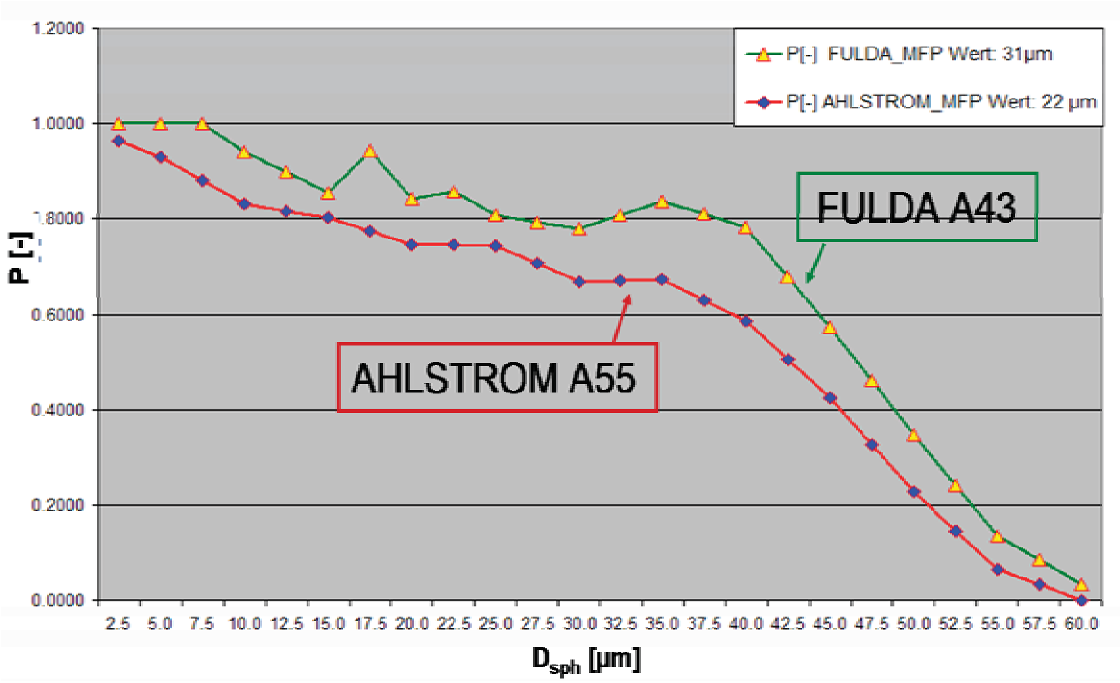


Figure 133: Comparison of particle penetration depth curves for Ahlstrom A55 (red) with MFP=22µm and Fulda A43 with MFP=31µm. Test case parameters match those, given for Figure 129.

Having obtained the efficiency curve data for the materials to be compared, further information on filter behavior can be retrieved relatively simple. Figures 134 and 135 show what A43 and A55 would probably do to a theoretical Gauss distributed, polydisperse particle collective with medium particle diameter μ=30µm and standard deviation σ=10µm. An estimation concerning the distribution of the deposited and the penetrating particle cloud is possible and values for depositing and penetrating fractions can be given. The results show that the overall deposited particle fraction f_{dep} for the given case will amount to

f_{dep}=50.85% for A55 and f_{dep}=38.28% for A43. Due to the demonstrational character of the calculations, the fact that the efficiency curves have been derived for monodisperse particle fractions is neglected here.

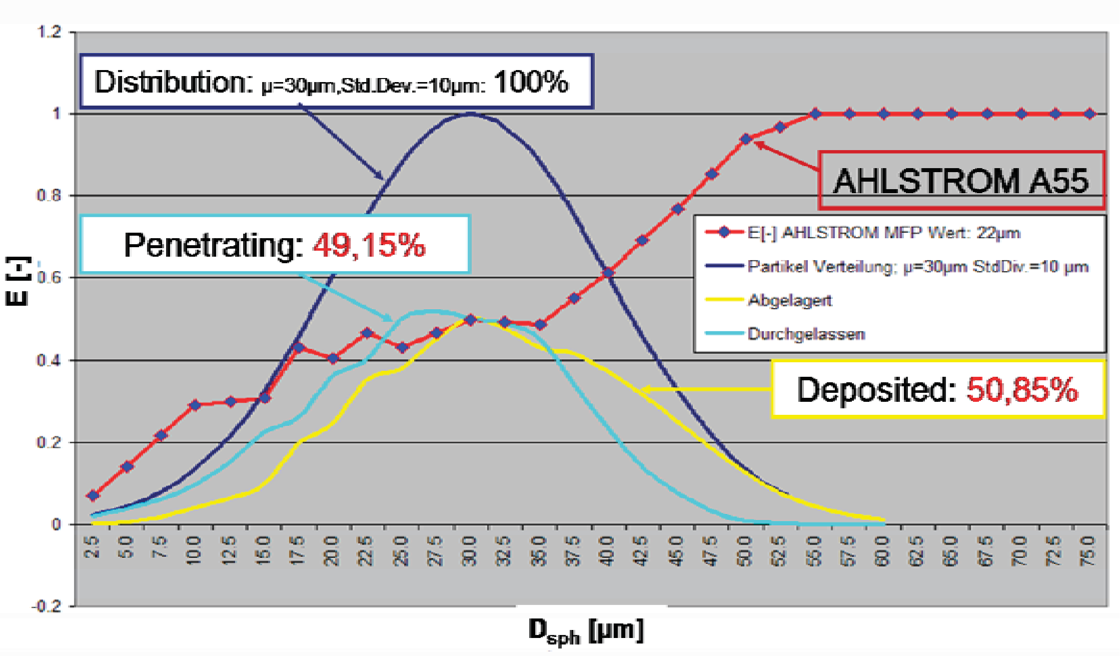


Figure 134: Filter fibre efficiency curve of A55 with superposed, Gauss distribution of theoretical dirt particle cloud with µ=30µm and σ=10µm. Particle distribution curves for penetrating and deposited fractions as well as fractional values can be calculated.

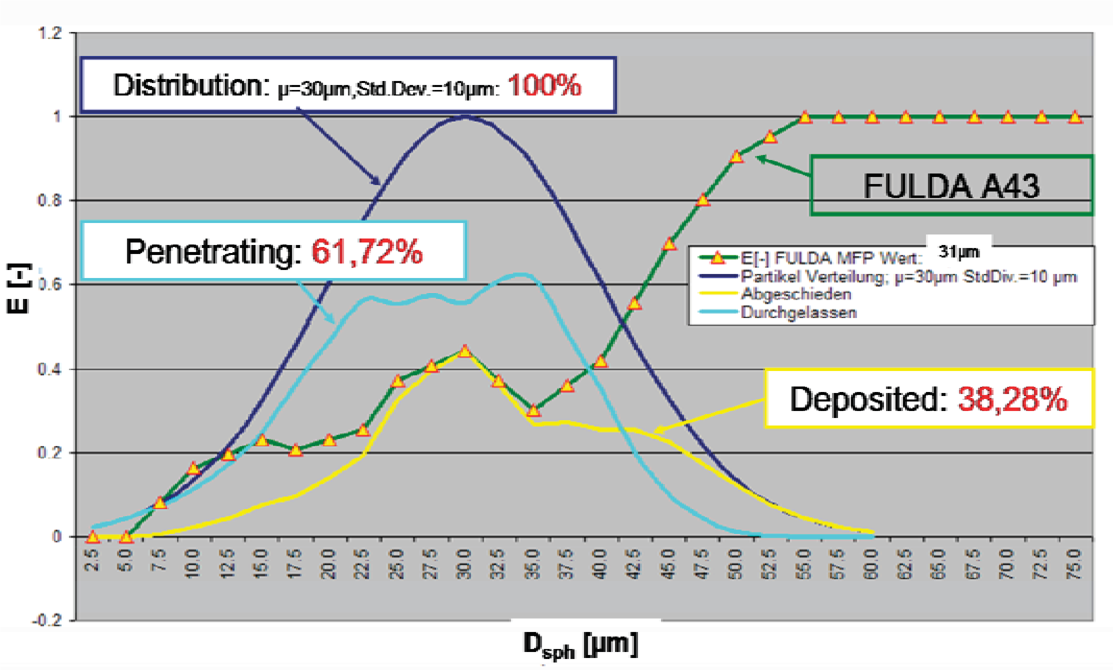


Figure 135: Filter fibre efficiency curve of A43 with superposed, Gauss distribution of theoretical dirt particle cloud with μ=30μm and σ=10μm. Particle distribution curves for penetrating and deposited fractions as well as fractional values can be calculated.

9.5 Effect of Dirt Pre-Deposition

The final application example is concerned with demonstrating the solver's capability of estimating the effect of increasing dirt deposition on the filter fibre efficiency. This ability stems from the particle-particle interaction feature which enables the simulation of cake filtration effects. In order to derive the results shown in Figure 136, previously conducted simulation runs on empty (clean) A43 fibres have been compared to pre-deposition runs that were set up as follows: An initial simulation on the empty fibre has been run and stopped when the desired amount of particles had settled in the fibre. A second run was started, using the particle laden geometry of the first run as initial condition. In addition to that, the pre-deposition utility was activated. It ensures that pre-deposited particles from the first run are not moved under any circumstances during the second run. Figure 136 has been produced by selecting the initial run such that a degree of 10% of the total porous fraction of the material was occupied with particles. Then consecutive runs were conducted with various particle size classes coming in after the initially deposited particles. The results in Figure 136 show that the pre-deposition primarily influences the deposition rate of larger particle classes. Smaller particles get deviated around the newly closed pores due to their lower Stokes numbers.

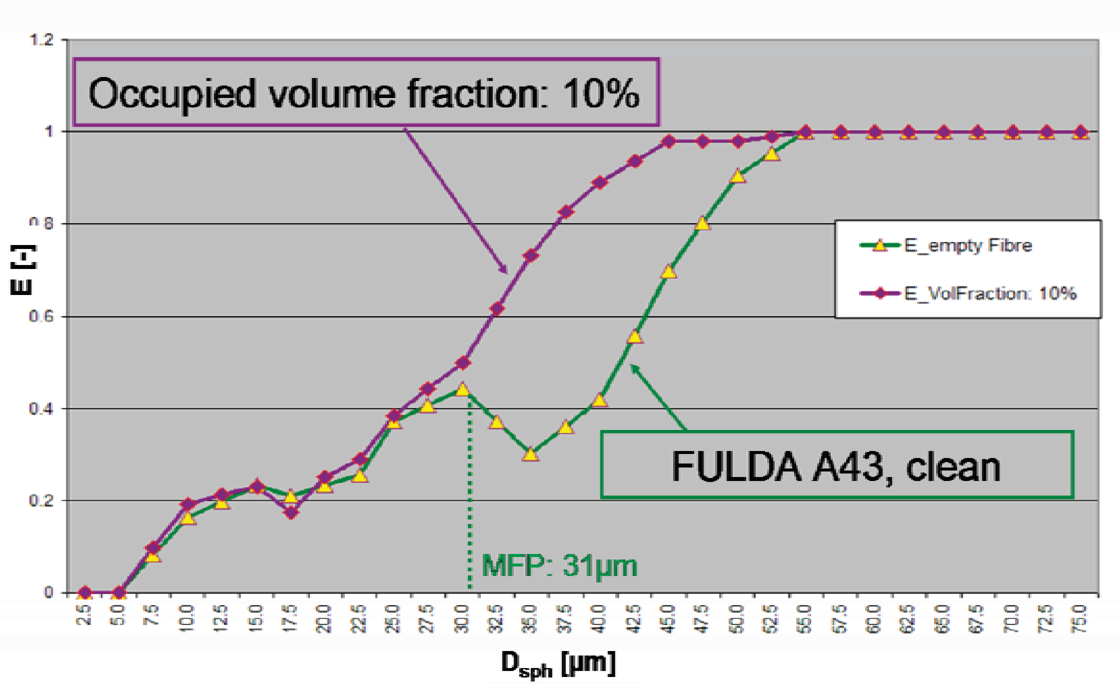


Figure 136: Filter fibre efficiency curves of A43. Comparison between results for empty (clean) material (green) and fibre with a volume fraction of 10% being occupied (purple).

10 Conclusion and Vision

A Lagrangian, (non-)spherical dirt particle model for filtration applications has been programmed using the Open Source CFD tool box OpenFOAM®. The new solver is capable of simulating the most relevant deposition effects that lead to the filtering of dirt particles out of the automotive oil circuit, on a microscopic level. It contains detailed sub-models concerning particle-fluid one and two way coupling, particle-fibre and particle-particle interaction. Based on those implementations and upon the application of digitally reconstructed, structured grid meshes, a realistic simulation of fluid filtration due to adhesion, inertial impact, cake and sieving effects is realized. A special feature of the code, setting it apart from comparable pieces of software is its ability to consider not only simplified, smooth, spherical dirt particles but also more realistically shaped, non-spherical particles. An important part of this thesis has involved the presentation of newly developed methods to cope with the generalized, ellipsoid shape of the dirt particles. Another focus has been laid on working out an adaptive time stepping concept in order to cope with numerical instability problems based on the implementation of an explicit Euler temporal particle movement discretization scheme. After a detailed presentation of the physical and mathematical background of the new model, the basic work flow and the code's programming structure have been explained. User input parameters have been shown and described as well.

Solver verification has been successfully conducted. The verification procedure was concentrated on the following process parameters: pressure drop over the filter fibre medium, filter fibre efficiency curves and particle penetration depth curves. Validation has been carried out on several levels:

- Plausibility checks

- Comparison with (semi-) analytical results

- Comparison with results found in literature

- Comparison with experimental data

Extensive measures have been taken to achieve this level of verification:

- A semi-analytical calculation scheme was devised to calculate the fibre efficiency of simple, artificially created fibre meshes. Thereby the results have been found to be in good agreement with corresponding, numerical calculations.

- Several realistic simulation runs have been set up in order to obtain results, comparable to data found in literature.

- An extensive, experimental set up was created, which enables the determination of parameters like pressure drop, fibre efficiency and particle penetration characteristic. The experimental set up consists of the oil-fibre and the particle detection facility and is concluded by the development of a 3D particle reconstruction algorithm.

Using the experimental equipment, solver results in terms of pressure drop and filter fibre efficiency could be verified. Results were found to be in very good agreement and deviations could be plausibly explained. Furthermore the application of the experimental facilities in particle distribution detection mode constitutes a new, innovative stand-alone method to characterize test particle distributions in filter fibre media on a full 3D basis.

Having concluded the model and program presentation and having succeeded in verifying many results, some realistic examples for practical solver application were given. In connection with those examples the following engineering tasks can probably be solved in the near future:

- Purely CFD based estimation of the influence of filter fibre quality (adhesion) and morphology (sieving) on filter characteristics.

- First time consideration of particle shape effects in a filter simulation application.

- Non-experimental comparison of filter materials.

- Estimation of the influence of dirt pre-deposition on filter performance (cake filtration effect).

- The virtual design of filter fibre media, leading to a dramatic cost and time reduction in the R&D process of filter producers by reducing time consuming, costly experimental runs.

One further benefit of the model lies within the C++ based, well structured code, which allows simple, modular extension by sub-models. Two sub-models, the bacteria model and the E-static model, have already been presented here. In the future several new models might be added to the code. Likely candidates are:

- A Brownian motion model.

- A turbulence model, probably based on the Langevin [18] approach.

- A particle agglomeration model.

- An extension of the E-static model to a full Maxwell equation solver.

- A chemical reaction model.

All in all the new development behind this thesis is seen as a significant advancement in the field of CFD based filtration simulation and concerning non-spherical, Lagrangian particle modelling. It is a new tool, based on a solid, physical, mathematical and numerical framework which constitutes a small puzzle piece on the road to promote computational engineering as the leading discipline among mankind's technical achievements of the 21st century.

11 List of Figures

Figure 19: Particle cloud in digitally reconstructed fibre geometry.

Figure 20: Particle surrounded by 48 enlarged pressure/velocity help points.

Figure 21: Accumulation of large particles in simplified fibre geometry.

Figure 22: Simple filter fibre case with large particles before and while plugging effect occurs.

Figure 23: Pressure difference between inlet and outlet over run-time, corresponding to the qualitative, exemplary plugging case shown in Figure 22.

Figure 24: Simplified horizontal and vertical fibre structure deformed by oil current. Dense cloud of rather large particles getting entangled in the structure and causing plugging effect, e.g. diversion of the flow.

Figure 25: Realistic, microscopic (200µm*200µm*300µm) fibre geometry reconstructed from CT-scan images.

Figure 26: The ellipsoid shape can approximate a wide variety of geometries, e.g. plates and sticks.

Figure 27: Exemplary ellipsoid particle with co-rotational coordinate system.

Figure 28: Non-spherical particle with 18 *surface help points* and 48 *pressure/velocity help points*.

Figure 29: Non-spherical particle surrounded by help points and panels.

Figure 30: Positions of the 18 surface help points within the co-rotational particle coordinate system.

Figure 31: Position and orientation of surface normal vector $n_{p,16}$.

Figure 32: Side view of particle with panels, M/P-points and HP-points.

Figure 33: Sinking, spherical particles with velocity vectors as well as highlighted help point and panel structure.

Figure 34: Sketch of local force balance and force effect on panel centre.

Figure 35: Sketch of form and shear force contribution to panel drag force.

Figure 36a: Ellipsoid particle accelerating towards valve. Alignment along the current stream lines.

Figure 36b: Ellipsoid with activated fibre vicinity module.

Figure 72: Plot of particle speed up curves for exemplary fluid time step Δt_f=1.333s, particle dimension D_{sph}=0.01m, dynamic fluid viscosity μ_f=0.1Pas, particle density ρ_p=2000 kg/m³, and uniform flow velocity u_f=0.2m/s.

Figure 73: Exemplary plot of non-spherical particle speed up, for D_{sph}=0.001m, u_f=0.1m/s, μ_f=0.1Pas with tangent at t=0s and non–spherical particle relaxation time $\tau_{p,nonsph}$ =8*10^{-4} s.

Figure 74: Ratio of spherical particle relaxation time according to *Hölzer-Sommerfeld* drag $\tau_{p,H/S}$ and spherical particle relaxation time according to Stokes' drag implementation $\tau_{p,St}$ plotted against particle Reynolds number.

Figure 75: Spherical and non–spherical particle relaxation time behavior against relative half axe deviation around D_{sph}.

Figure 76: Evaluation of Equ.218 for varying $\Delta t_p/\tau_p$, with u_f=0.2 m/s.

Figure 77: Comparison of analytical and numerical speed up curve with $\Delta t_p/\tau_p$=0.3 and u_f=0.2m/s.

Figure 78: Plot of σ_{rel} against $\Delta t_p/\tau_p$ for a spherical particle of arbitrary size and composition which speeds up in an arbitrary fluid.

Figure 79: Plot of σ_{rel} against $\Delta t_p/\tau_p$ with $\Delta t_p/\tau_p$ ranging from 0.0 to $\Delta t_{p,end}/\tau_p$=0.55.

Figure 80: Ratio of $k_{rel}(\Delta t_{p,0}/\tau_p)/k_{rel}(\Delta t_{p,0}/\tau_p$ =0.001) plotted against $\Delta t_{p,0}/\tau_{p.}$

Figure 81: Plot of k_{rel} against M. Maximum k_{rel} value at M_{max}=1.60. Definition of $M_{99,9}$ value and $k_{rel;99,9}$ value.

Figure 82: Plot of k_{rel} against M.

Figure 83: Comparison of exemplary, spherical particle velocity curves calculated for same physical conditions with static and event-force-adapted time stepping scheme.

Figure 84a: Plot of $u_{p,max}$ against log($\tau_{p,0}$) for the case of ΔI=6*10^{-6} kgm/s.

Figure 84b: Plot of $\Delta u_{p,max}/u_{p,max,static}$ (being the relative difference between the two $u_{p,max}$ curves shown in Figure 84a) against log($\tau_{p,0}$) for the case of ΔI=6*10^{-6} kgm/s.

Figure 85: Plot of $u_{p,max}$ against $u_{p,max,unhindered}$ for the case of $\tau_{p,0}$=7.78*10^{-2} s. Comparison of static time stepping scheme and adaptive time stepping scheme.

no adhesion and case c) is calculated with non-spherical particles and with adhesion.

Figure 132: Comparison of filter fibre efficiency curves for different but simular materials.

Figure 133: Comparison of particle penetration depth curves.

Figure 134: Filter fibre efficiency curve of A55 with superposed, Gauss distribution of theoretical dirt particle cloud with μ=30μm and σ=10μm.

Figure 135: Filter fibre efficiency curve of A43 with superposed, Gauss distribution of theoretical dirt particle cloud with μ=30μm and σ=10μm.

Figure 136: Filter fibre efficiency curves of A43.

12 References

1. M. Mataln, G.Boiger, W. Brandstätter, B. Gschaider, (2008). Simulation of Particle Filtration Processes in Deformable Media, Part 1: Fluid-Structure Interaction, ICE Stroemungsforschung GmbH., Montanuniversitaet Leoben. *Int.Journal of Multiphysics, Vol.2,(No.2), July 2008 , pp. 179-189(11)*;

2. G. Boiger, M. Mataln, W. Brandstätter, B. Gschaider, (2008). Simulation of Particle Filtration Processes in Deformable Media, Part 2: Large Particle Modelling, ICE Stroemungsforschung GmbH., Montanuniversitaet Leoben. *Int.Journal of Multiphysics, Vol.2,(No.2), July 2008 , pp. 191-206(16)8;*

3. G. Boiger, M. Mataln, W. Brandstätter, (2009). Simulation of Particle Filtration Processes in Deformable Media, Part 3.1: Basic concepts and particle-fluid force implementation of a non-spherical dirt particle solver, ICE Stroemungsforschung GmbH., Montanuniversitaet Leoben. *Article in review since Jan 2009. Int.Journal of Multiphysics;*

4. G. Boiger, M. Mataln, W. Brandstätter, (2009). Simulation of Particle Filtration Processes in Deformable Media, Part 3.2: Interaction modelling and solver verification of a non-spherical dirt particle solver, ICE Stroemungsforschung GmbH., Montanuniversitaet Leoben. *Article in review since Jan 2009. Int.Journal of Multiphysics;*

5. G. Boiger, M. Mataln, W. Brandstätter, (2009). Adaptive time stepping for explicit Euler implementation of spherical and non-spherical particle speed up. ICE Stroemungsforschung GmbH., Montanuniversitaet Leoben. *Int.Journal of Multiphysics; Vol.3,(No.3), August 2009 , pp. 267-291(25);*

6. K. Thomsen, (2004). Simplified Surface area of Ellipsoid. *Numericana.com, “Final Answers”.*

7. S.Schütz, M.Piesche, G.Gorbach, M.Schilling, C.Seyfert, P.Kopf, T.Deuschle, N.Sautter, E.Popp, T.Warth. (2007). CFD in der mechanischen Trenntechnik. *Chemie Ingeneur Technik 2007, 79, No.II. DOI: 10.1002/cite.200700109;*

8. S.Schütz, M.Schilling, M.Piesche. (2007). Bestimmung der Widerstandskraft und des Transportverhaltens kugelförmiger Partikel in Strömungen mit Hilfe moderner CFD-Werkzeuge. *Chemie Ingenieur Technik 2007, 79, No.12. DOI: 10.1002/cite.200700080;*

9. L.Rosendahl. (1999). Using a multi-parameter particle shape description to predict the motion of Non-Spherical particle shapes in swirling flow. Institute of Energy Technology, Aalborg University. *Applied Mathematical Modeling 24 (2000), 11-25.*

10. E.Loth. (2007). Drag of non-spherical solid particles of regular and irregular shape. Department of Aerospace Engineering, University of Illinois. *Powder Technology, 182 (2008), 342-353.*

11. M. Mataln, W. Brandstätter, (2004). A Unified Approach to Model Fluid-Structure Interactions. Montanuniversitaet Leoben, Austria. *Society of Petroleum Engineering, seminar notes, 2004.*

12. W.J.Kowalski, W.P.Bahnfleth, T.S.Whittam, (1999). Filtration of Airborne Microorganisms: Modeling and Prediction. Pennsylvania State University, 1999. ASHRAE Transactions 105(2), 4-17. http://www.engr.psu.edu/ae/iec/abe/publications .

13. C.L.Cox, E.W.Jenkins, P.J.Mucha, (2005). Modelling of Debris Deposition in an Extrusion Filter Medium. Clemenson University, Clemenson, SC. Proceedings of the 21st Annual Meeting of the Polymer Processing Society, Leipzig, Germany, June 19-23, 2005.

14. M.Sommerfeld, (2007). Fundamentals of Particles, Droplets and Bubbles. Martin Luther Universität, Halle-Wittenberg, Germany. SIAMUF-Seminar and workshop, Chalmers University of Technology, October 2007.

15. C.Crowe, M.Sommerfeld, Y.Tsuji, (1998). Multiphase Flows with Droplets and Particles, Boca Raton, FL: CRC Press 1998; ISBN-10: 0849394694/0-8493-9469-4.

16. R.P. King. Introduction to Particle Fluid Flow, Butterworth-Heinemann; (2002).

17. C.S. Peskin, (2002). The immersed boundary method, Cambridge University Press, Acta Numerica 2002, pp.1-39, DOI: 10.1017/S0962492902000077.

18. S. Lain, D. Bröder, & M. Sommerfeld, (1999). Experimental and numerical studies of the hydrodynamics in a bubble column. *Chemical Engineering Science, 54, 4913.*

19. S. Lain, & M. F. Göz, (2000). Instabilities in numerical simulations of dispersed two-phase flow. *Mechanical Research Communication, 27, 475.*

20. S. Lain, & M. F. Göz, (2001). Numerical instabilities in bubble tracking in two-phase flow. *International Journal of Bifurcation and Chaos, 11(4), 1169.*

21. S. Lain, & M. F. Göz, (2004). Study of the numerical instabilities in LaGrangian tracking of bubbles and particles in two-phase flow. *Computers and Chemical Engineering 28 (2004, 2727-2733.*

22. S. Lain, M.F. Göz,& M. Sommerfeld, (2006). Instabilities in LaGrangian Tracking of Bubbles and Particles in Two-Phase Flow numerical. *Wiley InterScience, 52: 469-477.*

23. A. Haider,& O. Levenspiel, (1988). Drag Coefficient and Terminal Velocity of Spherical and Non Spherical Particles. *Powder Technology, 58 (1989), 63-70.*

24. A. Hölzer,& M. Sommerfeld, (2007). New, simple correlation formula for the drag coefficient of Non – Spherical Particles. Martin-Luther-Universität, Halle-Wittenberg, Germany. *Powder Technology, Vol.184(3) (2007), 361-365.*

25. S. V. Apte, K. Mahesh, T. Lundgren, (2003). A Eulerian-Lagrangian model to simulate two-phase/particulate flows, Center for Turbulence Research, Annual Research Briefs, (2003).

26. G.Ahmadi, (2005). Particle Transport, Deposition & Removal II*, ME637 On-line Curriculum*, Potsdam, New York.

27. K.B.Shah, M.Megahed, (2005). Discrete and Chimera Particle Simulation: Novel LaGrangian Extensions and Applications, ESI-Group, Germany. Industrial two-phase flow CFD, 2005. VKI LS 2005-04; Vol.1.

28. M.Lutz, (2005). Python. Kurz&Gut, Köln: O'Reilly Verlag 2005. ISBN: 3-89721-511-X.

29. W.C.Hinds, (1999). Aerosol technology: properties, behavior, and measurement of airborne particles, New York: Wiley 1999. ISBN: 0471194107 9780471194101.

30. W.B.Jones, W.J.Thron, (1980). Continued Fractions: Analytic Theory and Applications, Cambridge University Press 2009. ISBN-10: 0521101522.

31. W.Brandstätter, (2005). Flow and Combustion Modelling. Montanuniversitaet Leoben, Austria. Lecture notes February 2005 – July 2005.

32. C.Crowe, M.Sommerfeld, Y.Tsuji, (1998). Multiphase Flows with Droplets and Particles, Boca Raton, FL: CRC Press 1998; ISBN-10: 0849394694/0-8493-9469-4.

33. A.Haider, O.Levenspiel, (1988). Drag Coefficient and Terminal Velocity of Spherical and Non Spherical Particles. *Powder Technology, 58 (1989), 63-70.*

34. W. Bohl, W.Elmendorf, (2005). Technische Strömungslehre, Vogel Fachbuch, Kamprath Reihe Aufl.:13.2005. ISBN-10: 3834330299.

35. C. Mundo, C. Tropea& M. Sommerfeld, (1997). Numerical and Experimental Investigation of Spray Characteristics in the Vicinity of a Rigid Wall. Universität Erlangen-Nürnberg & Martin-Luther-Universität, Halle-Wittenberg, Germany. *Experimental Thermal and Fluid Science, (2007),15:228-237.*

36. http://www.opencfd.co.uk/openfoam/

37. http://www.cfd-online.com/Wiki/CFD-Wiki:Introduction

38. OpenFOAM programmer's guide. Version 1.4.1. OpenCFD Limited, United Kingdom. (2007).

39. OpenFoam user guide. Version 1.5. OpenCFD Limited, United Kingdom. (2008).

40. O. Univ.-Prof. Dr. F.D.Fischer, Fluid Mechanics, lecture notes, Montanuniversitaet Leoben , Austria (2002)

41. Comet Version 2.00, User Manual, Institut of Computational Continuum Mechanics GmbH, (2002).

42. Bjarne Stroustrup, The C++ Programming Language, Special Edition (2000)

43. William H.Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, Numerical Recipies in C, Second Edition (1992).

44. openfoam-extend.svn.sourceforge.net.

45. T.Jaeger, O.Koglin, T.Kreutzer, A.Metzger, C.Schulz, (2005). Die GPL kommentiert und erklärt, O'Reilly Verlag, Köln, Germany. Aufl.:1.2005. ISBN: 3-89721-389-3.

46. W.Brandstätter, (2004). Advanced Simulation Techniques. Montanuniversitaet Leoben, Austria. Lecture notes February 2004 – July 2004.

47. C.W. Longest, C. Kleinstreuer& J.R. Buchanan, (2002). Efficient computation of micro-particle dynamics including wall effects. North Carolina State University. 3211 Broughton Hall, Raleigh, NC 27695-7910, USA. *Computers and Fluids,(2004),33:577-601.*

48. I.M. Spiliotis, B.G. Mertzios, (1997). A Fast Parallel Skeleton Algorithm on Block Represented Binary Images. Democritus University of Thrace. Greece. Elektrik, (*1997),Vol. 1, No. 1, pp. 161-173.*

49. D. Besnard, F. Harlow, (1986). Non-spherical particles in two phase flow. Cent. Études Limeil, France. International Journal of Multiphase Flow, (*1986),Vol. 12, No. 6, pp. 891-912(21).*

50. A.K. Majumder, J.P. Barnwal (2003). A computational Method to Predict Particles Free Terminal Settling Velocity. Regional Research Laboratory, Bhopal, India. IE(I) Journal-MN, August *2004,Vol. 85.*

51. P. Raiskinmäki, A. Shakib-Manesh, A. Koponen, A. Jäsberg, M. Kataja, J. Timonen (2000). Simulations of non-spherical particles suspended in a shear flow. University of Jyväskylä, Finland. *Computer Physics Communications, Vol.129,Issues 1-3, July 2000 , pp. 185-195; (Science Direct).*

52. E. Dodemand, R. Prud'Homme, P. Küntzmann (1995). Influence of unsteady forces acting on a particle suspension application to the sound propagation. *Int.Journal ofMultiphase Flow, Vol.21,(No.1), January 1995 , pp. 27-51(25)*; (Ingentaconnect.com).

53. Ho Chi Anh, M.Sommerfeld, R. Säuberlich, M. Bohnet (2004). Modellierung der Partikelaggloneration im Rahmen des Euler/Lagrange- Verfahrens und Anwendung zur Berechnung der Staubabscheidung im Zyklon. Dissertation. Martin-Luther Universität Halle-Wittenberg, Germany. Urn:nbn:de:gbv:3-000006169.(nbn-resolving.de).

54. C. Gamsjäger, Ph. Gittler, A. Mehrle, (2007). Numerische Untersuchung der Umströmung einer Flügel-Rumpf-Kombination mittels eines inkompressiblen Panelverfahrens. Diplomarbeit. Johannes Kepler Universität Linz, Austria.

55. M.W. Reeks, S. McKee (1984). The dispersive effects of Basset history forces on particle motion in turbulent flow. Nuclear Laboratories, Englang; Hertford College, Oxford, England. *Phys.Fluids, Vol.27, Issue 7, 1573(1984); DOI:10.1063/1.864812.*

56. M.Loewenberg (1993). Stokes resistance, added mass, and Basset force for arbitrarily oriented, finite length cylinders. University of Sydney, Australia. *Phys.Fluids A. Vol.5, Issue 3, 765(1993); DOI:10.1063/1.858660.*

57. J.F.Favier, M.H. Abbaspour-Fard, M. Kremmer, A.O. Raji (1999). Shape representation of axi-symmetrical, non-spherical particles in discrete element simulation using multi-element model particles. *Engineering Computations Vol.16, Issue 4, 467-480(13), (1993); DOI:10.1108/02644409910271894. (MCB UP Ltd., emeraldinsight.com)*

58. J.I. Jeong, C. Mansoo. (2003). A simple bimodal model for the evolution of non-spherical particles undergoing nucleation, coagulation and coalescence. Seoul National University. South Korea. *Journal of Aerosol Science Vol.34, Issue 8, 965-976(12), (2003); ISSN 0021-8502.*

59. B.W. Lee, J.I. Jeong, J.Y. Hwang, M. Choi, S.H. Chung. (2001). Analysis of growth of non-spherical silica particles in a counterflow diffusion flame considering chemical reactions, coagulation and coalescence. Seoul National University. South Korea. *Journal of Aerosol Science Vol.32, 165-185, (2001); ISSN 0021-8502/01.*

60. J.I. Jeong, M. Choi. (2000). A sectional method for analysis of growth of polydisperse non-spherical particles undergoing coagulation and coalescence. Seoul National

University. South Korea. *Journal of Aerosol Science Vol.32, 565-582, (2000); ISSN 0021-8502/01.*

61. A. Li, G. Ahmadi. (1992). Dispersion and Deposition of Spherical Particles from Point Sources in a Turbulent Channel Flow. Clarkson University, Potsdam, NY, USA. *Journal of Aerosol Science Vol.16, 209-226, (1992); ISSN 0278-6826/92.*

62. K. Vafai, C.L. Tien. (1980). Boundary and Inertia Effects on Flow and Heat Transfer in Porous Media. University of California, Berkley, CA, USA. *Int.Journal Heat Mass Transfer Vol.24, pp.195-203, (1980); ISSN 0017-9310/81/0201-0195.*

63. Goldstein H. (1980). *Classical Mechanics*. 2nd. ed., Addison-Wesley. ISBN 0-201-02918-9

64. L.D. Landau, E.M. Lifshitz. (1976). *Mechanics.* 3rd. ed., Pergamon Press. ISBN 0-08-021022-8 (hardcover) and ISBN 0-08-029141-4 (softcover).

65. J.B. Marion, ST. Thornton. (1995). Classical Dynamics of Systems and Particles. 4th. ed., Thomson. ISBN 0-03-097302-3.

66. KR. Symon. (1971). *Mechanics.* 3rd. ed., Addison-Wesley. ISBN 0-201-07392-7.

67. RA. Tenenbaum.(2004). *Fundamentals of Applied Dynamics.* Springer. ISBN 0-387-00887-X.

68. J. Bryant.(2007). Ellipsoid. Wolfram Demonstrations Project.

69. A. Hölzer, M. Sommerfeld. (2007). Lattice Boltzmann simulations to determine drag, lift and torque acting on non-spherical particles. PhD thesis. Martin-Luther Universität Halle-Wittenberg, Germany.

70. A. Hölzer, M. Sommerfeld. (2007). Analytische, numerische und experimentelle Untersuchungen zur Abscheidung von Partikeln aus Ölen mit Hilfe fasriger Tiefenfiltermedien. PhD thesis. Universität Stuttgart, Germany.

71. Will Schroeder, Ken Martin, Bill Lorensen.(2006). *The* Visualization Toolkit An Object-Oriented Approach To 3D Graphics. *4th Edition.* Kitware, Inc. Publishers. ISBN 1-930934-19-X.

72. J. Ahrens, B. Geveci and Ch. Law. (2005). ParaView: An End-User Tool for Large Data Visualization. In the Visualization Handbook. Edited by C.D. Hansen and C.R. Johnson. Elsevier. 2005.

73. P.J.Oliveira; R.I.Issa. (2001). AN IMPROVED PISO ALGORITHM FOR THE COMPUTATION OF BUOYANCY-DRIVEN FLOWS. *Numerical Heat Transfer, Part B: Fundamentals:* An International Journal of Computation and Methodology. 1521-0626, Volume 40, Issue 6, 2001, Pages 473 – 493.

74. R. J. Gibbons and J. van Houte. (1971). Selective Bacterial Adherence to Oral Epithelial Surfaces and Its Role as an Ecological Determinant**.** Forsyth Dental Center, Boston, Massachusetts 02115. Infect Immun. 1971 April; 3(4): 567-573.

75. R. Becker, F. Sauter. (1973). **Theorie der Elektrizität, Band 1: Einführung in die Maxwellsche Theorie, B.G. Teubner, Stuttgart (1973).**

76. A. Reiss, G. Boiger, W.Brandstätter. (2008). Numerical modelling of electro static effects in filtration applications. Bacc. thesis. Montanuniversität Leoben, ICE Strömungsforschung GmbH. Austria.

77. Purcell, Edward M. (1989). Elektrizität und Magnetismus. 4. Auflage. Translation: E. Gerstenhauer. Braunschweig/Wiesbaden: Vieweg. 1989.

78. Katz, Victor J. (May 1979). "The History of Stokes' Theorem", Mathematics Magazine 52 (3): 146–156.

79. Zhigang Li and Hai Wang (2003). Drag force, diffusion coefficient, and electric mobility of small particles. I., Phys. Rev. E 68(3), 061206 (2003) [9 pages]. DOI: 10.1103/PhysRevE.68.061206.

80. Arfken, G. "Gauss's Theorem." §1.11 in Mathematical Methods for Physicists. 3rd edition. Orlando, FL: Academic Press, pp. 57-61, 1985.

81. A. Rushton, A.S. Ward, R.G. Holdrich. (1996). Solid-Liquid Filtration and Separation Technology. VCH Weinheim. Weinheim Germany.

82. ISO 4548-12:2000-02. Methods of test for full-flow lubricating oil filters for internal combustion engines - Part 12: Filtration efficiency using particle counting, and contaminant retention capacity. 2000.

83. W. Georgi , E. Metin. (2007). Einführung in LabVIEW. Hanser Fachbuch, 2007. ISBN-10: 3446411097.

84. W Denk, JH Strickler, W.W. Webb. Two-photon laser scanning fluorescence microscopy. *Science* 6 April 1990:Vol. 248. no. 4951, pp. 73 – 76. DOI: 10.1126/science.2321027.

85. R. Maruthi, K. Sankarasubramanian. (2007). Multi Focus Image Fusion based on the Information Level in the Regions of the Images. Velammal Engineering College, Sri Sai Ram Engineering College. India. *Journal of Theoretical and Applied Information Technology. Vol.3.No.4.2007.*

86. W. Huang, Z. Jing. (2005). Evaluation of focus measures in multi-focus image fusion. Institute of Aerospace Information and Control, School of Electronics, Information and Electrical Engineering, Institute of Aerospace Science and Technology. China. Pattern Recognition Letters*. Vol.28, No.4,2007, Pages 493-500.*

87. W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery. Numerical Recipies, 3rd edition. NewYork, USA. Cambridge University Press, 2007;

88. O.Beucher. (2006). Matlab und Simulink. Germany. Pearson Studium, ISBN-10/-13: 3-8273-7206-2/97838273720622006.

89. Fluent. (2005). Fluent 6.2 User Guide. Fluent Inc., Centerra Resource Park, 10 Cavendish Court, Lebanon, NH 03766, USA.

90. A. Cedilnik, B. Geveci, K. Moreland, J. Ahrens, and J. Favre. (2006). Remote Large Data Visualization in the ParaView Framework. A. Heirich, B. Raffin, L. P. Santos (eds.)

editors, Eurographics Parallel Graphics and Visualization 2006, pages 162-170, May 2006.

91. S.V. Patankar, (1980). Numerical Heat Transfer and Fluid Flow. Hemisphere Pub, 1980; ISBN-10/-13: 0891165223/9780891165224.

92. W. Kepplinger, (2003). Skriptum: Mechanische Verfahrenstechnik. Montanuniversität Leoben, 2003. Kapitel: Fest-Flüssig Trennung, S-11.