

## PART 2: LARGE PARTICLE MODELLING

### Simulation of particle filtration processes in deformable media

**Gernot Boiger<sup>1</sup>, Marianne Mataln<sup>1</sup>, Bernhard Gschaider<sup>1</sup> and Wilhelm Brandstätter<sup>2</sup>**

<sup>1</sup>ICE Strömungsforschung GmbH, Hauptplatz 13, 8700 Leoben, Austria

<sup>2</sup>Montanuniversität Leoben, Franz-Josefstrasse 18, 8700 Leoben, Austria  
[www.ice-sf.at](http://www.ice-sf.at)

#### ABSTRACT

In filtration processes it is necessary to consider both, the interaction of the fluid with the solid parts as well as the effect of particles carried in the fluid and accumulated on the solid. While part 1 of this paper deals with the modelling of fluid structure interaction effects, the accumulation of dirt particles will be addressed in this paper. A closer look is taken on the implementation of a spherical, LAGRANGIAN particle model suitable for small and large particles. As dirt accumulates in the fluid stream, it interacts with the surrounding filter fibre structure and over time causes modifications of the filter characteristics. The calculation of particle force interaction effects is necessary for an adequate simulation of this situation. A detailed Discrete Phase Lagrange Model was developed to take into account the two-way coupling of the fluid and accumulated particles. The simulation of large particles and the fluid-structure interaction is realised in a single finite volume flow solver on the basis of the OpenSource software OpenFoam.

Keywords: Simulation; Filtration processes; Large particles; Lagrange particle; Discrete Phase Lagrange Model; Plugging effect; OpenFoam

#### 1. INTRODUCTION

High pressure and shear forces as well as cavitation effects close to the motor block cause local material rupture. To avoid an accumulation of dirt particles in the closed oil motor block lubrication circuits, filtration elements have to be installed. The design of filter fibres which perform this task still requires time consuming, costly experimental runs. This is why the motivation arises to enlist the aid of computational fluid dynamics (CFD) in order to simulate filtration processes in lubrication circuits.

It has been found that two major effects govern the change of filter fibre characteristics over filter life time. Firstly there is the considerable deformation of the filter fibre, caused by the forced oil convection in the stream. The simulation of these fluid structure interaction effects is described in detail in [1].

Secondly dirt particle behaviour has to be considered. Low density clouds of dirt particles, ranging from 5 to 50 microns in diameter, occur in the stream. They interact with the filter, get entangled in the fibre structure and slowly accumulate there, thus causing plugging of the

flow. Even though those effects occur at a microscopic level, they give rise to macroscopic changes of filter characteristics, such as porosity, pressure drop and over all filter efficiency. Following this logic, the simulation is carried out at a microscopic scale with the intention of statistically averaging results for several regions in order to retrieve macroscopic information in later phases of development.

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 that are considered to have the most influence on the changes of filter characteristics are fluid – particle drag, particle – wall, fibre and particle – particle interactions, as well as plugging effects. Prevailing physical conditions on the fluid side can be described as follows: oil currents are slow, highly viscous, incompressible and non turbulent, the geometry is of microscopic scale and the influence of heat flux is negligible. As a consequence certain simplifications can be made in the modelling.

The open source CFD tool box OpenFOAM [12] has been found to provide a suitable environment for the development of a fluid - structure interaction - and particle solver. Since OpenFOAM is based upon the programming language C++ it features a completely modular programming structure. In addition to that the source code can be altered as required by the user.

## 2. EULERIAN – LAGRANGIAN APPROACH TO SIMULATE PARTICLE BEHAVIOUR

Since the case to be considered entails very low particle distribution densities a LAGRANGIAN [6] and not an EULERIAN [7] approach was chosen to simulate particle behaviour. A LAGRANGIAN implementation considers particles to be single mass centres interacting with the surrounding environment. No second, continuous particle phase is necessary.

This simplification goes hand in hand with the fact that only slow motions of highly viscous fluids are modelled. As a consequence a simple, single phase, non turbulent, incompressible implementation can be used for the simulation of fluid flow conditions. Eqn.(1) and Eqn.(2) show the governing set of equations.

$$\frac{\partial}{\partial t}(u_{fi}) + \nabla(u_{fi}u_{fi}) = -\frac{1}{\rho_f}\nabla p + \frac{\mu_f}{\rho_f}\nabla^2 u_{fi} + S_I \quad (1)$$

$$\nabla \cdot \vec{u}_f = 0 \quad (2)$$

Where by  $u_f$  is the fluid velocity field,  $p$  the pressure,  $\mu_f$  the dynamic fluid viscosity,  $\rho_f$  fluid density and  $S_I$  the source term for volumetric forces such as gravity.

The values for particle acceleration  $A_p$ , velocity  $u_p$  and position  $x_p$  depend on the resulting velocity- and pressure field. They are described by the LAGRANGIAN Equations of motion that are written out as Eqn. (3) and Eqn. (4).

$$\frac{d}{dt}(\vec{x}_p) = \vec{u}_p \quad (3)$$

$$\frac{d}{dt}(\vec{u}_p) = \vec{A}_p \quad (4)$$

The essential coupling of particle behaviour to the particle surroundings is performed by the acceleration term  $\vec{A}_p$ . As can be seen in Eqn. (5), it contains acceleration effects of the local fluid velocity field, buoyancy force, particle collision effects  $\vec{A}_{coll}$ , particle-wall interaction  $\vec{A}_{wall}$  and particle-fibre interaction  $\vec{A}_{fibre}$ .

$$\vec{A}_p = D_p \left( \vec{u}_f - \vec{u}_p \right) + \left( 1 - \frac{\rho_f}{\rho_p} \right) \vec{g} + \vec{A}_{coll} + \vec{A}_{wall} + \vec{A}_{fibre} = 0 \quad (5)$$

While buoyancy effects can be implemented just as shown in Eqn. (5), all other factors of influence need specific elaboration in terms of modelling.

Throughout the solver a strict modelling rule has been adopted. It states that only forces can lead to a change of motion. Thus no direct particle interaction takes place to affect particles velocity or position. Instead this particle interaction produces impact forces that lead to acceleration, new velocity and changing positions.

Events like e.g. elastic collisions, in general have a duration time  $\Delta t_i$  that is much shorter than any reasonable discrete time step  $\Delta t$ . Consequentially they produce extremely high, time dependent impact forces  $\vec{F}_i(t)$ . To correctly simulate the overall momentum change of all  $n$  impact events on the particle, which occur during  $\Delta t$ , an adapted impact force  $\vec{F}_p$  has to be used as shown in Eqn. (6).

$$\vec{F}_p = \frac{1}{\Delta t} \sum_{i=1}^n \int_{t=0}^{\Delta t_i} \vec{F}_i(t) \cdot dt \quad (6)$$

Since all particle behaviour is based upon this overall adapted impact force  $\vec{F}_p$ , its correct calculation is of primary importance. The following chapters describe how the compositional terms of the particle acceleration vector  $\vec{A}_p$  are computed.

## 2.1. IMPACT FORCES: PARTICLE – FLUID INTERACTION

In the case of sparsely distributed, small particles in a highly viscous fluid, the drag force  $\vec{F}_{fp}$  will be the dominant factor on overall impact force. The commonly used expression to model fluid drag on particles is shown in Eqn. (7), where  $\rho_f$  is fluid density,  $A_c$  is the cross sectional particle area perpendicular to flow direction and  $C_w$  is the dimensionless drag coefficient.

$$\vec{F}_{fp} = \frac{1}{2} \cdot A_c \cdot C_w \cdot \rho_f \cdot \left( \vec{u}_f - \vec{u}_p \right) \cdot \left| \vec{u}_f - \vec{u}_p \right| \quad (7)$$

The order of both, particle diameter  $d_p$  and dynamic 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  $u_{fp}$ , which is well below 1. This fact is written out in Eqn. 8.

$$O(Re_p) \sim O\left(\left| \vec{u}_{fp} \right| \right) < 1 \quad (8)$$

Hence inertial flow field effects can be neglected altogether, simplifying the NAVIER-STOKES equations to pressure and viscosity effects as seen in Eqn. (9) and [11].

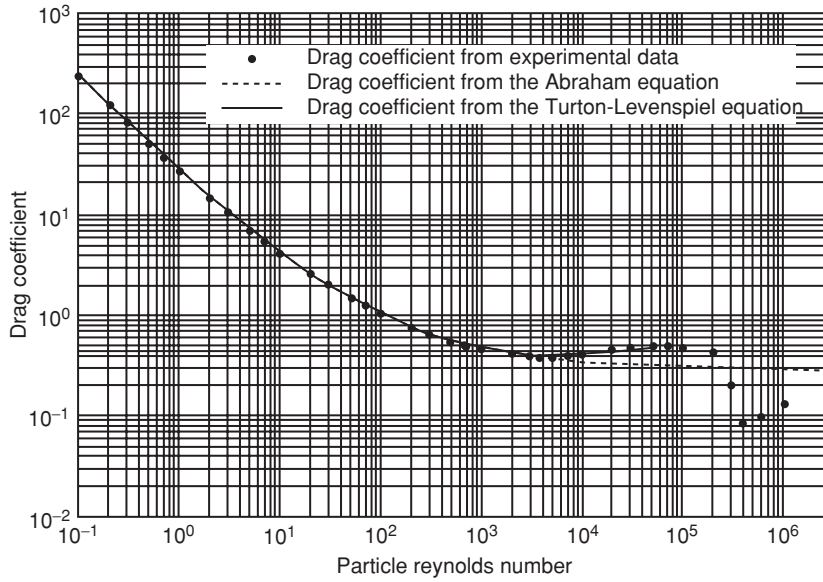


Figure 1 Drag coefficient of solid spheres plotted against particle Reynolds number, Comparison of experimental data and results from ABRAHAM and TURTON – LEVERSPIEL equation [11].

$$\frac{\partial p}{\partial x_i} = \mu_f \cdot \frac{\partial^2 u_i}{\partial x_j \partial x_j} \quad (9)$$

Under these conditions the STOKES law for drag on spherical particles is applicable. It is shown in Eqn. (10). In contrast to other relations for  $C_w = f(Re_p)$  it yields an entirely analytical solution for the drag coefficient, which considers both, form drag and the shear stress.

$$C_w = \frac{24}{Re_p} = \frac{24 \cdot \mu_f}{|\vec{u}_{fp}| \cdot d_p} \quad (10)$$

Should higher particle Reynolds numbers occur ( $Re_p > 4$ ), then the solver automatically switches to semi empirical correlations like the ABRAHAM equation [11] seen in Eqn. (11). It is valid for  $Re_p < 6000$ . For particle Reynolds numbers  $Re_p > 6000$  and  $Re_p < 2 \cdot 10^5$  the TURTON – LEVENSPIEL [11] equation is used. It is shown in Eqn. (12).

$$C_w = \frac{24}{Re_p} \left( 1 + 0, 11 \cdot \sqrt{Re} \right)^2 \quad (11)$$

$$C_w = \frac{24}{Re_p} + \frac{5, 48}{Re^{0, 573}} + 0, 36 \quad (12)$$

Figure (1) shows the close accordance of the ABRAHAM equation and the TURTON – LEVENSPIEL equation with experimental results.

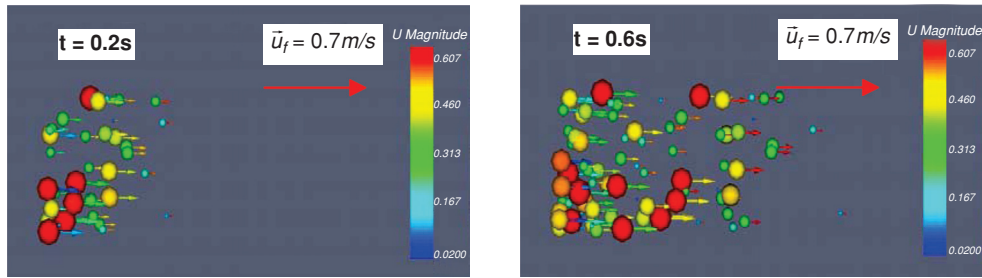


Figure 2 Sizing effect: Smaller particles follow current more readily.

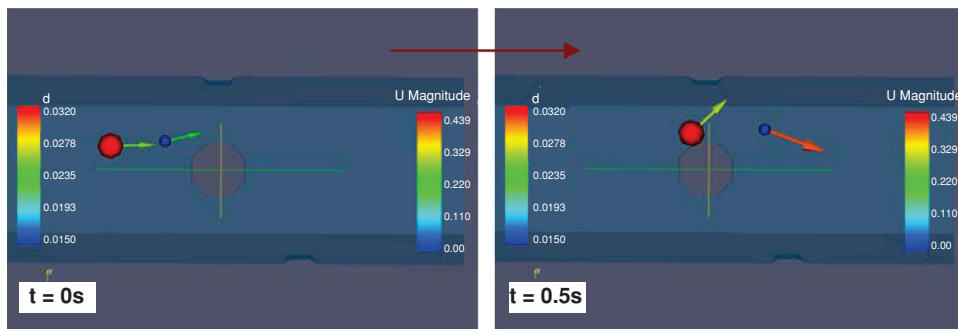


Figure 3 Impact inertial effect – Particles of higher STOKES number show higher probability of impact.

This simple drag implementation into the EULERIAN LAGRANGIAN solver already leads to very plausible results in terms of particle sizing effects. It is evident in Figure (2) that due to their lower inertia the smaller particles accelerate much faster than larger particles in an otherwise steady state flow field.

The interaction of differently sized particles with a single fibre is depicted in Figure (3). Whilst the smaller particle more or less follows the flow field in the neighbourhood of the fibre, the larger particle impacts on it and deviates accordingly.

Again the influence of particle inertia effects is demonstrated in a physically plausible way.

## 2.2. IMPACT FORCES: PARTICLE – WALL INTERACTION

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

There are two types of boundary patches: inlet/outlet patches, where by 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 on the other hand 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  $\rho_{pc}$  perpendicular to the flow direction, as described by Eqn. (13).

$$\frac{\overline{\partial \rho(r)}}{\partial r} = \frac{2\pi}{A_f} \cdot \int_{r=0}^R \frac{\partial \rho(r)}{\partial r} \cdot r \cdot dr = 0 \quad (13)$$

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 Eqn. (6) the computed wall impact force  $F_{pw}$  has to account for the resulting momentum change of the particle mass  $m_p$  during the discrete force effect time  $\Delta t$ . For elastic reflection the resulting momentum change should reverse the direction of the particle velocity components parallel to the normal vector  $\vec{n}_w$  of the wall patch. The computed wall impact force is specifically calculated to have just that effect and can be written as:

$$\vec{F}_{pw, el} = -m_p \frac{2 \cdot (\vec{u}_p \cdot \vec{n}_w) \cdot \vec{n}_w}{\Delta t} \quad (14)$$

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 Eqn. (14) has to be extended to describe the adaptable, semi elastic wall force  $F_{pw, iel}$ . It can thus be written as

$$\vec{F}_{pw, iel} = \vec{F}_{pw, el} \cdot \left( \frac{1 + E_{pw}}{2} \right) \quad (15)$$

This implementation, at the end of a wall impact time step  $\Delta t$ , results in the computation of the updated particle velocity  $\vec{u}_p(t + \Delta t)$  as shown in Eqn. (16).

$$\vec{u}_p(t + \Delta t) = \vec{u}_p(t) - (1 + E_{pw}) \cdot (\vec{u}_p(t) \cdot \vec{n}_w) \cdot \vec{n}_w \quad (16)$$

### 2.3. IMPACT FORCES: PARTICLE – FILTER FIBRE INTERACTION

The exact formulation of particle-fibre interaction effects is of course a very important aspect of modelling the overall filter efficiency of the medium. Effective particle-fibre forces will vary depending on parameters such as surface roughness and material stiffness. Determination of those parameters has to be the subject of separate research. This solver however is 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 Eqn. (15). A filter fibre medium in general shows strong cohesion to objects to be filtered out. Therefore it is sensible to choose a completely inelastic impact scenario.

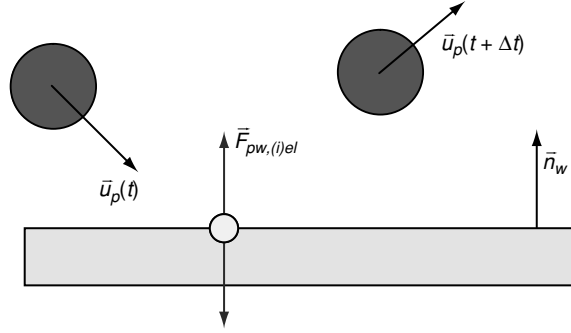


Figure 4 Sketch of particle-wall impact event.

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  $F_f$ . The value of  $F_f$  is proportional 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  $\eta_{fp}[-]$  is implemented.  $F_f$  is always directed to point against the direction of the current particle velocity  $u_p$ . Thus  $F_f$  is computed as seen in Eqn. (17), with  $n$  being the total number of discrete forces, affecting the particle during time step  $\Delta t$ .

$$\overrightarrow{F_f} = - \sum_{i=1}^{n-1} \left( \overrightarrow{F_{p,i}} \cdot \overrightarrow{n_w} \right) \cdot \left( \eta_{fp} \cdot \frac{\overrightarrow{u_p}}{|\overrightarrow{u_p}|} + \overrightarrow{n_w} \right) \quad (17)$$

3. If the value of the particle force components parallel to the fibre surface,  $F_{p,p}$  is smaller than  $F_f$ , the particle slows down. Should the resulting negative particle acceleration during time  $\Delta t$  lead to reversing the glide direction, the particle velocity is set to 0 and the fibre friction force  $F_f$  is set according to Eqn. (18).

$$\overrightarrow{F_f} = - \sum_{i=1}^{n-1} \overrightarrow{F_{p,i}} \quad (18)$$

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 ( $F_{p,p} > F_{stick}$ ), the particle can regain some motion.

What is essential in 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. An example can be seen in Figure (5).

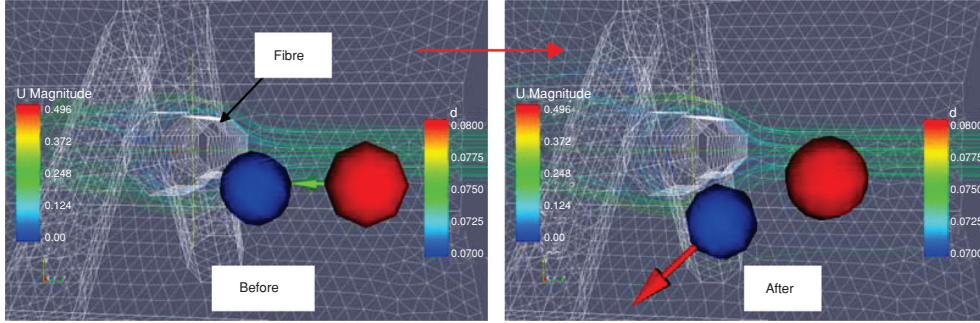


Figure 5 Two particles in fibre vicinity with velocity vectors. Blue particle gets immobilized on fibre (left) is hit by red particle and is blown off again (right).

#### 2.4. IMPACT FORCES: PARTICLE – PARTICLE INTERACTION

Particle agglomeration effects at the fibre, can lead from complex particle – particle interaction to plugging effects up to changes of overall permeability. Thus a 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  $\sim N^2$ . Here a collision model was implemented that 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 an overall collision model.

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

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 filter fibre can be handled just like particle – wall collision as given in Eqn. (15). The difference is that the immobilized particle does not just absorb the collision counter force like a wall boundary patch. It can surpass the fibre sticking barrier, thus regaining some motion. This is called 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

$$\vec{F}_{pp, el} = \frac{2 \cdot m_a \cdot m_b}{\Delta t \cdot m_{ges}} \cdot (\vec{v}_{b, n} - \vec{v}_{a, n}) \quad (19)$$

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 (6) shows a simple sketch of a particle-particle impact event.

In reality any scenario of collision is combined with some degree of object deformation along a certain deformation path  $\Delta s_{def}$ . 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 takes place, with the overall deformation path being only

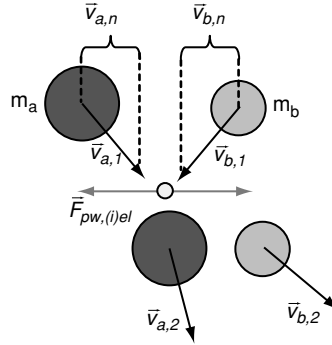


Figure 6 Simple sketch of particle-particle impact event.

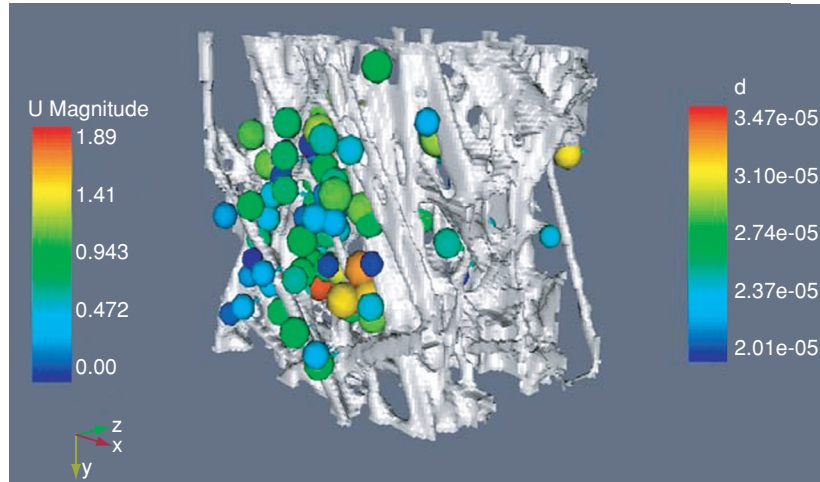


Figure 7 Particle agglomeration in digitally reconstructed fibre geometry.

1/2 of the elastic case. For both cases the modelled impact takes place during time  $\Delta t$  with discrete, constant, relative velocity  $v_{rel}$ . Because of that, the virtual deformation path  $v_{rel} * \Delta t$  stays the same. This is why the occurring, modelled collision force  $F_{pp, iel}$  has to account for differences in elasticity. Using the elasticity coefficient  $E_{pp}$  it can be described by:

$$\vec{F}_{pp, iel} = \vec{F}_{pp, el} \cdot \left( \frac{1 + E_{pp}}{2} \right) \quad (20)$$

The particle collision model can realistically simulate the agglomeration of large particle numbers in a filter fibre. Figure (7) shows an example.

#### Large particle model

A special feature of the presented Lagrangian particle solver is its ability to handle both small and large particles as shown in Figure (8). With the particle diameter being  $d_p$  and the mean cell diameter being  $d_c$ , the term large refers to a  $d_p/d_c$  ratio that is greater than 1.

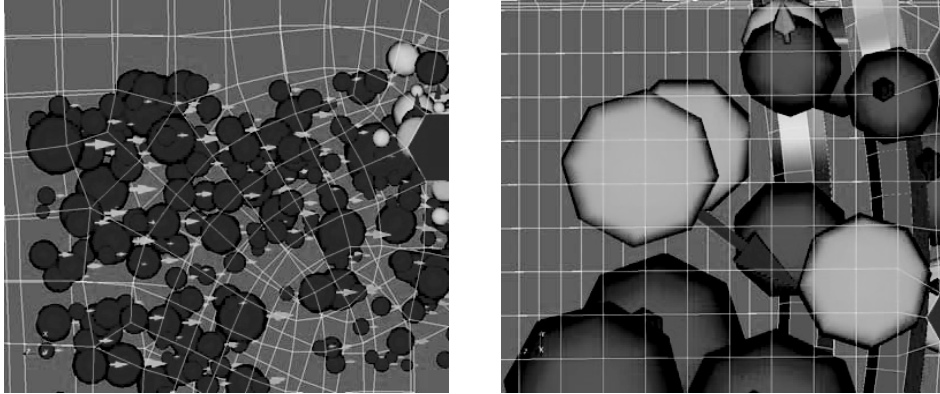


Figure 8 Particle simulation with small ( $d_p/d_c < 1$ ) particles and large ( $d_p/d_c > 1$ ) particles.

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

1. For small particles the fluid drag term is calculated just as described under chapter 2.1, using the uniform, relative fluid - particle velocity  $u_{fp}$ . For large particles the fluid velocity field has to be considered non uniform over the surface of the particle. Consequentially the relative fluid - particle velocity has to be averaged over all calculation cells enclosed by the particle. This is achieved by the introduction of pressure velocity auxiliary points (AP) (see chapter 2.4.1).
2. The pressure gradient across the particle surface can no longer be neglected (see chapter 2.4.1).
3. Large particles rather lead to flow field distortions than small ones and can cause considerable plugging effects in the fibre (see chapter 2.4.2).

#### *2.4.1. Large particles: pressure gradient*

Fluid - particle drag forces, calculated according to Stokes law, consider drag due to shear stress and form drag based upon pressure gradients over the particle boundary layer. Those pressure gradients are caused by the laminar, creeping flow around the spherical shape. Interaction force calculation for large particles has to consider additional macro scale pressure gradients caused by other obstacles, e.g. fibres, in the stream path.

To account for those macro scale pressure gradients across the particle surface, the concept of pressure velocity auxiliary points has been introduced. Pressure velocity auxiliary points are in essence small satellites stationed 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  $v_i$  conditions at the particle surface and in the specific calculation cell they are located in. The satellites' number is user defined. Of course higher help point numbers lead to higher calculation time consumption but also to higher accuracy. Figure (9) shows a particle surrounded by 48 help points.

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

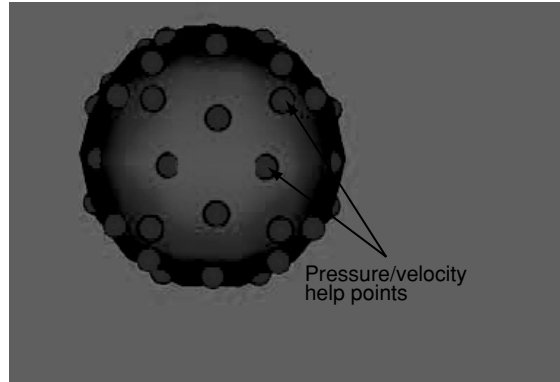


Figure 9 Particle surrounded by 48 enlarged help points.

An equal surface fraction  $A_{hp}$ , is assigned to each help point and can be easily calculated with

$$A_{hp} = \frac{A_p}{M} = \frac{4}{3} \cdot \frac{1}{M} \cdot r_p^2 \cdot \pi \quad (21)$$

where  $A_p$  is the total particle surface,  $r_p$  the particle diameter 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  $M$ . An infinite number of surface help points leads to a perfect representation of the pressure force. This fact can be illustrated by

$$\vec{F}_p = \lim_{M \rightarrow \infty} \left( -A_{hp} \cdot \sum_{i=1}^M p_i \cdot \vec{n}_{ip} \right) = - \int_{r=0}^{A_p} p \cdot \vec{n}_p dA \quad (22)$$

where  $\vec{n}_{ip}$  is the surface normal vector at each help point.

By applying the Gauss' theorem the pressure force can be written as a volumetric term.

$$\vec{F}_p = - \int_{A_p} p \cdot \vec{n}_p dA = - \int_{V_p} \nabla p \cdot dV \quad (23)$$

Eqn. (19) was previously used to describe the over all acceleration effects of particle interaction and can now be extended to account for the macro scale pressure gradient. The extended version can be written as

$$\vec{A}_p = D_p (\vec{u}_f - \vec{u}_p) - \frac{1}{m_p} \int_{V_p} \nabla p \cdot dV + \left( 1 - \frac{\rho_f}{\rho_p} \right) g + \vec{A}_{coll} + \vec{A}_{wall} + \vec{A}_{fiber} = 0 \quad (24)$$

This formulation for the additional pressure force is particularly useful in combination with the implementation of a particle plugging effect that is chosen in this context and described in the following chapter.

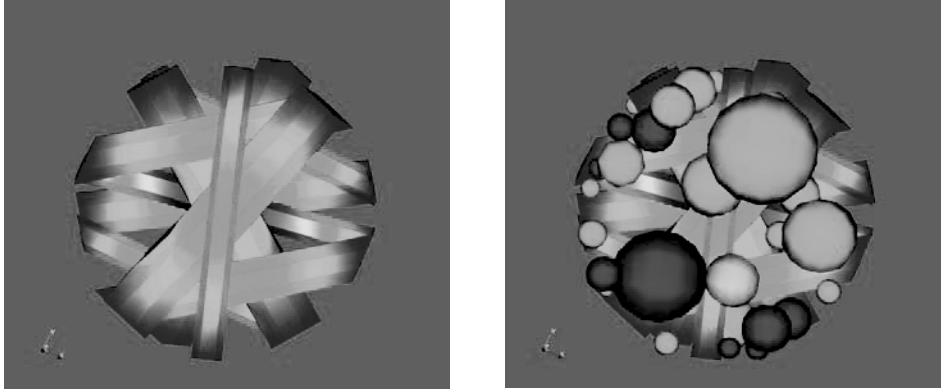


Figure 10 Accumulation of large particles in simplified fibre geometry.

#### 2.4.2. Large particles: plugging effect

If the simulated particles are large in comparison to the dimensions of filter fibres, accumulation effects occur much more easily. Figure (10) shows the 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 this plugging effect becomes imperative for realistic simulation. In order to implement particle plugging with effect on the fluid field, the vicinity of the particles is seen as a porous medium. To describe flow in porous media, the DARCY term has to be added to the governing momentum equations. 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 be expressed via

$$\nabla p_{Darcy} = \frac{\mu_f}{\alpha} u_f \quad (25)$$

where  $\alpha[m^2]$  is the permeability,  $\mu_f$  the dynamic fluid viscosity and  $u_f$  the fluid velocity field.

The NAVIER-STOKES equation for porous media can be written as

$$\frac{\partial}{\partial t}(\rho_f u_f) + \nabla(\rho_f u_f u_f) = -(\nabla p + \nabla p_{Darcy}) + \nabla(\mu_f D_c) \quad (26)$$

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  $\infty$  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 as

$$\nabla p_{Darcy} = \frac{\mu_f}{\alpha} u_f = 10^6 \cdot depot(x_i, y_i, z_i) \cdot u_f \quad (27)$$

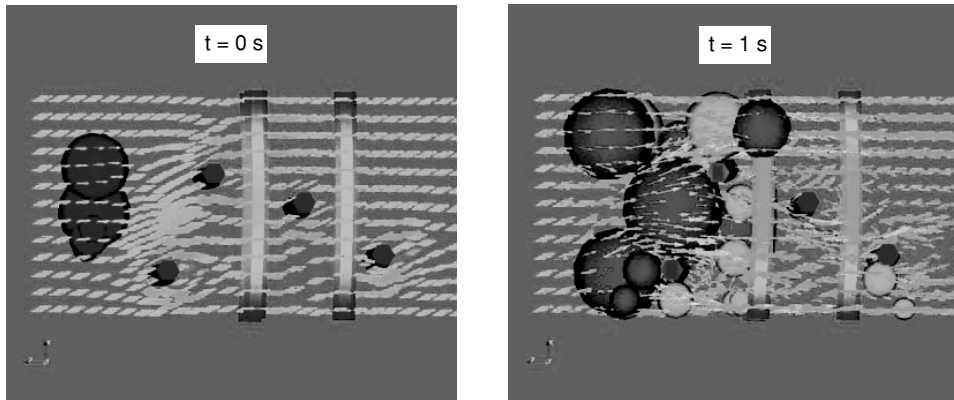


Figure 11 Simple filter fibre case with cloud of large particles before (left) and while (right) plugging effect occurs. The vector field symbolizes the fluid velocity field (0.2–0.4m/s).

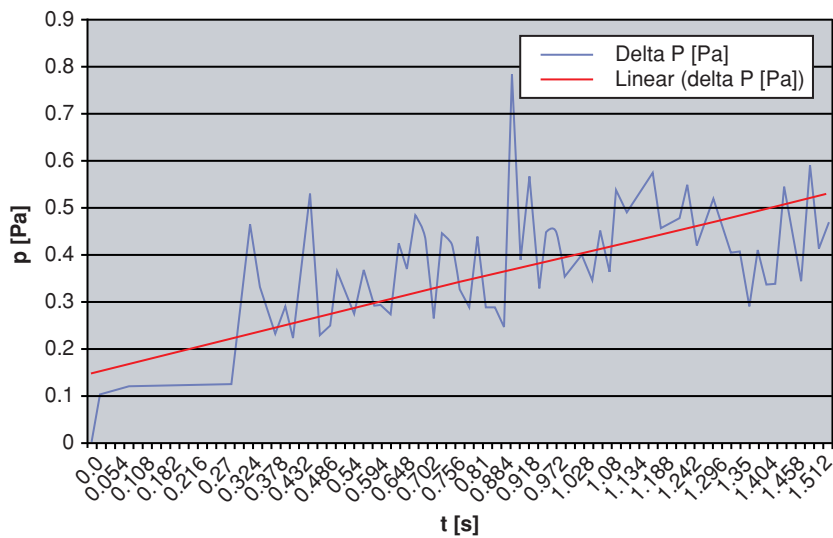


Figure 12 Pressure difference between inlet and outlet over time, corresponding to the plugging case shown in Figure (11).

As a starting condition the permeability  $\alpha$  is  $\infty$  and the depot field 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 (11) contains chronological snapshots of a simple, multi fibre case with a cloud of large particles getting stuck. The plugging effect on the fluid flow, which is represented through velocity vectors, is clearly visible.

Figure (12) shows the development of the pressure difference between inlet and outlet against run time. It corresponds to the plugging case shown in Figure (11). At a fixed volume flow rate the inlet - outlet pressure gradient increases over time, just as expected.

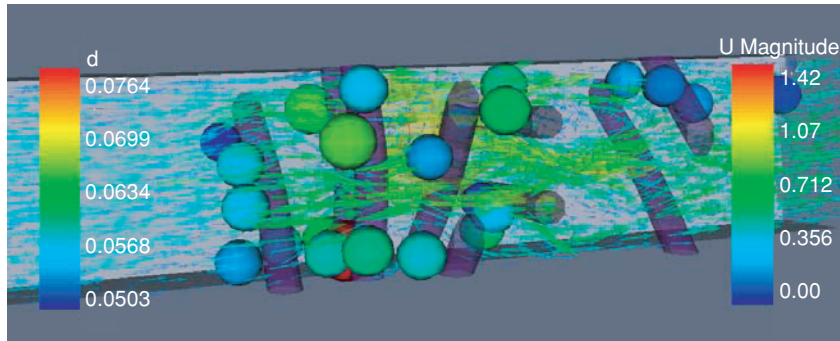


Figure 13 Simplified horizontal and vertical fibre structure deformed by oil current. Dense cloud of rather large particles getting entangled in the structure and causing plugging effect.

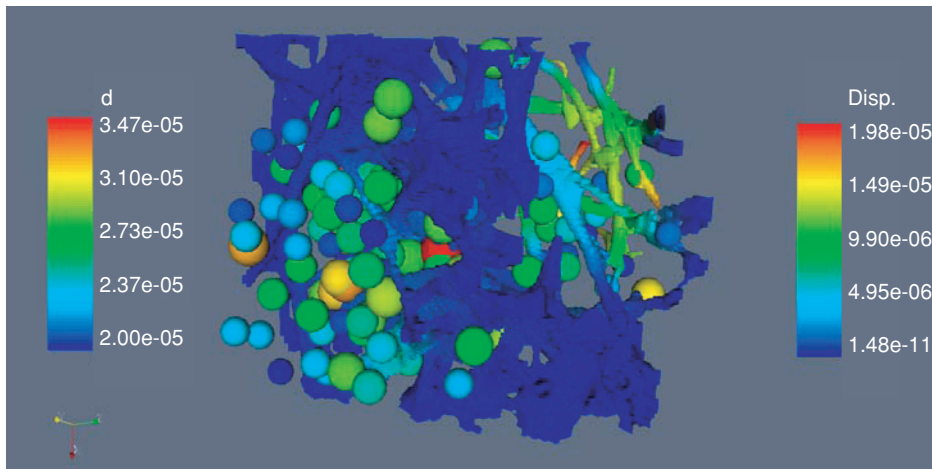


Figure 14 Realistic fibre geometry reconstructed from computer tomographic scan images. Dense cloud of rather small particles getting entangled in the fibre.

### 3. THE FILTRATION SOLVER: COMBINATION OF PARTICLE SOLVER AND FLUID - STRUCTURE MECHANISM

OpenFOAM features a strictly modular programming structure. This is why the fluid – structure solver [1] and the particle solver (chapter 2) could be developed almost separately. With only small additional efforts the two components were combined to a unified filtration solver. The various modules can be switched on or off as required by the user. This means that on the one hand flow in deformable media can be calculated without any particle injection and that on the other hand the particle solver can be used without fluid - structure interaction.

Several combined solver runs on test benchmark cases and on actual technical applications have already proven that proposed computational strategy is robust and stable. So far the plausibility of results is very encouraging. In the following some screenshots of successful runs are presented.

Figure (13) is a screen shot from a case with simple horizontal and vertical filter fibres that are visibly deformed by oil current. 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.

As seen in Figure (14), the solver has already been applied to realistic geometries as well. 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 computer tomographic scans and meshed by a self written meshing utility.

#### 4. CONCLUSION

OpenFOAM, the C++ based open source CFD platform, has served for the development of a filtration simulation tool. Given certain simplifications, the process of dirt particle filtration from hydraulic oils can now be simulated in terms of resulting filter efficiency and pressure drop. The fluid conditions, for which the applied physics are valid, are highly viscous, relatively slow, incompressible and isothermal fluid flows. The particle model has been designed for low density clouds of nearly spherically shaped dirt particles. Therefore an Eulerian-Lagrangian, non turbulent implementation for the fluid – particle – filter fibre interaction has been chosen. Fluid – structure interaction is considered in the modelling, so that small deformation effects of filter fibres in the current can be simulated. A detailed description of that part of our work can be found in [1].

As presented in this paper the program also contains an elaborate algorithm for dirt particle simulation. An important feature of the implementation is that large particles, enclosing several calculation cells can realistically be simulated. Consequentially a plugging effect of particles getting stuck in the filter fibre can be modelled. This is achieved by viewing the entire fluid geometry as a porous medium that calls for the addition of a Darcy term into the governing equations.

Since OpenFOAM features a strictly modular programming structure the stand alone development of a fluid - structure interaction solver and a particle motion and interaction simulator was possible. The two components can easily be coupled to the combined filtration solver. Considering the limitations, cited above, the solver presented here can be used to simulate filtration processes using any kind of hydraulic oil, filter fibre material and dirt particles.

Current development efforts aim towards increasing the range of applicability of the solver. The main goals are to be able to simulate:

- non spherical particle behaviour
- deformation of fibre material due to fluid forces and particle impact under working conditions.

#### 5. REFERENCES

- [1] M. Mataln, G. Boiger, W. Brandstätter, B. Gschaider, *Simulation of Particle Filtration Processes in Deformable Media, Part 1: Fluid-Structure Interaction*, ICE Stroemungsforschungs GmbH., Montanuniversitaet Leoben (2007)
- [2] M. Mataln, W. Brandstätter, *A Unified Approach to Model Fluid-Structure Interactions*, Montanuniversitaet Leoben (2004)
- [3] Ao. Univ.-Prof. Dr. Wilhelm Brandstätter, *Flow and Combustion Modelling*, lecture notes, Montanuniversitaet Leoben, Austria (2004/05)
- [4] O. Univ.-Prof. Dr. F. D. Fischer, *Fluid Mechanics*, lecture notes, Montanuniversitaet Leoben, Austria (2002)

- [5] *Comet Version 2.00*, User Manual, Institut of Computational Continuum Mechanics GmbH, (2002)
- [6] R. P. King, *Introduction to Particle Fluid Flow* (2002)
- [7] S. V. Apte, K. Mahesh, T. Lundgren, *A Eulerian-Lagrangian model to simulate two-phase/particulate flows*, Center for Turbulence Research, Annual Research Briefs (2003)
- [8] W.J.Kowalski, W.P.Bahnfleth, T.S.Whittam, *Filtration of Airborne Microorganisms: Modelling and Prediction*, (2003)
- [9] C. L. Cox, E. W. Jenkins, P. J. Mucha, *Modelling of Debris Deposition in an Extrusion Filter Medium* (2005)
- [10] W. Bohl, *Technische Strömungslehre*, (2002)
- [11] Clayton T. Crow, M. Sommerfeld, Y. Tsuji, *Multiphase Flows with Droplets and Particles*, Boca Raton, Fla. : CRC Press, (1998)
- [12] <http://www.open CFD.co.uk/openfoam/>
- [13] Bjarne Stroustrup, *The C++ Programming Language* (2000)
- [14] William H.Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, *Numerical Recipes in C*, Second Edition (1992)
- [15] <http://www.cfd-online.com/Wiki/CFD-Wiki:Introduction>