Development of a code for the direct numerical simulation of particles detaching from solid walls by hydrodynamical forces

Christiane Lechner, Hendrik Kuhlmann

Institute of Fluid Mechanics and Heat Transfer, Vienna University of Technology, Resselgasse 3, A-1040 Vienna, Austria

christiane.lechner@tuwien.ac.at, h.kuhlmann@tuwien.ac.at

Abstract

We report on our current progress in developing a code to be applied in the context of the cleaning of wafer surfaces by hydromechanical forces. Our goal is to study the detachment of (submicron) particles, exposed to a shear flow, from a wall by means of direct numerical simulation. The particles will be treated as rigid bodies with a two-way interaction with the fluid.

Our software is based on OpenFOAM. Following Uhlmann (2005) and Taira and Colonius (2007) we implement an immersed boundary method with direct forcing. The interpolation between the fixed Eulerian grid and the Lagrangian marker points is performed with the regularized δ-function introduced by Peskin (2002). For the time being we implemented this approach in 2D using OpenFOAM's standard solver `icoFoam` to solve the incompressible Navier-Stokes equations. To validate the implementation we present results on standard benchmark tests of the flow around a cylinder with prescribed motion.

Keywords: Immersed-boundary method, direct forcing, finite-volume method, OpenFOAM, Navier-Stokes equations, particulate flow

1 Introduction

The removal of impurities and particles from single crystal surfaces is an important technical problem in semi-conductor insdustry. We consider the detachment of (submicron) particles exposed to a fluid flow from a wafer surface. Due to their small size the particles essentially face a simple laminar shear-flow. We want to investigate the details of the flow around the particles and study the removal mechanisms numerically.

For this purpose we develop a code that can handle moving particles in a fluid. The particles are treated as rigid bodies with a resolved particle/fluid interface and a two-way interaction with the fluid. We start by considering circular respectively spherically particles, but the numerical method should be extendible to arbitrary shapes. Furthermore, we are seeking for an efficient numerical method that also allows to handle many-particle systems.

For the above reasons we selected the immersed boundary method with direct forcing. In this method the fluid equations are solved on a fixed Cartesian grid that does not conform with the particle surface and the particle's presence in the fluid is realized by a volume force entering the momentum equation.

Our code is based on the free and open-source software OpenFOAM that, besides providing a solver for the incompressible Navier-Stokes equations and many pre- and post-processing utilities, has MPI support built in on a low level which is necessary for 3D computations.

As we want to adapt the software to our needs on the top level of coding we follow the approach of Uhlmann (2005) who uses direct forcing with the discretized δ-function introduced by Peskin (2002) for the interpolation and spreading operations between the Cartesian grid and the Lagrangian "marker points" that are attached to the particle. For the numerical computation of the force we follow Taira and Colonius (2007) who treat the volume force as a Lagrangian multiplier similar to the pressure.

In this paper we describe our numerical method and validate the code for standard 2D test cases involving the flow over a cylinder with prescribed translational motion. In Section 2 we state the problem of a particle in a fluid modeled by a surface force. We present the equations of motion and, in particular, we derive the expressions for the total force and momentum the fluid exerts on the particle in terms of the surface force. In Section 3 we describe the numerical method we implemented by extending `icoFoam`. In Section 4 we provide numerical results and compare them to values from literature.

2 Representation of particles in a fluid by a surface force

2.1 Particles suspended in a fluid

We consider a rigid particle suspended in a fluid and denote the domain of the particle by $P(t)$ with boundary $\partial P(t)$ and the domain of the fluid by $D\backslash P(t)$. The fluid satisfies the incompressible Navier-Stokes equations on $D\backslash P(t)$ and is subject to the no-slip boundary condition $\boldsymbol{u}_{\text{fluid}} = \boldsymbol{u}_{\text{part}}$ at $\partial P(t)$ where $\boldsymbol{u}_{\text{part}}$ denotes the rigid body surface velocity of the particle

$$\boldsymbol{u}_{\text{part}}(\boldsymbol{x}, t) = \boldsymbol{U}(t) + \boldsymbol{\Omega}(t) \times (\boldsymbol{x} - \boldsymbol{X}(t)) \quad \text{at } \partial P(t), \tag{1}$$

$\boldsymbol{X}(t)$ being the particle's center of mass, and $\boldsymbol{U}(t) := d\boldsymbol{X}(t)/dt$ and $\boldsymbol{\Omega}(t)$ it's translational and angular velocities.

In the absence of external forces the equations of motion for the particle are given by

$$M\frac{d\boldsymbol{U}}{dt} = -\int\limits_{\partial P(t)} \boldsymbol{n} \,.\, \tilde{\boldsymbol{\Pi}}\, dO \qquad \boldsymbol{I} \,.\, \frac{d\boldsymbol{\Omega}}{dt} = -\int\limits_{\partial P(t)} (\boldsymbol{x} - \boldsymbol{X}) \times (\boldsymbol{n} \,.\, \tilde{\boldsymbol{\Pi}})\, dO \tag{2}$$

where $\boldsymbol{n}$ is the unit normal on $\partial P(t)$ pointing from the particle into the fluid, $\tilde{\boldsymbol{\Pi}}$ is the stress tensor of the fluid

$$\tilde{\Pi}^{ik} = \tilde{p}\,\delta^{ik} - \eta(\nabla^i u^k + \nabla^k u^i), \tag{3}$$

with η it's dynamic viscosity and $\boldsymbol{I}$ is the moment of inertia of the particle, $I^{ik} = \int_{P(t)} \rho(x)(\delta^{ik}\delta_{lm} - \delta^i_l \delta^k_m)\zeta^l \zeta^m dV$ with $\zeta^l := x^l - X^l$.

2.2 Representation of the particle by a surface force

In the immersed boundary method (originally introduced by Peskin (1972) for the flow around flexible membranes) the influence of the particle on the fluid is represented by an additional volume force in the momentum equation. The incompressible Navier-Stokes equations are solved in the whole domain D which includes the region of the fluid as well as the particle

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \nu \Delta \boldsymbol{u} + \boldsymbol{f} \qquad \text{in } D, \tag{4}$$

$$\nabla \cdot \boldsymbol{u} = 0 \qquad \text{in } D, \tag{5}$$

where $\boldsymbol{f}$ is a volume force that is constructed to enforce the no-slip boundary condition at the interface $\partial P(t)$. The constant density of the fluid ρ_0 has been absorbed into p, ν and $\boldsymbol{f}$.

There are various ways to formulate a force $\boldsymbol{f}$ with the above properties. Since we are only interested in the fluid flow outside of the particle we consider $\boldsymbol{f}$ being non-zero only at the particle surface, thereby mimicking a rigid hollow ball immersed in the fluid. We assume the surface of the particle to be described by $\boldsymbol{\xi}(q, s, t)$, with q, s being coordinates on $\partial P(t)$ and write the volume force $\boldsymbol{f}$ as

$$\boldsymbol{f}(x, t) = \int \boldsymbol{F}(q, s, t) \delta^3(\boldsymbol{x} - \boldsymbol{\xi}(q, s, t))\, dO_{(q,s)}, \tag{6}$$

where $\delta^3(\boldsymbol{x}) = \delta(x)\delta(y)\delta(z)$ with $\delta(x)$ being the Dirac δ-function satisfying

$$\int_{x_0-\epsilon}^{x_0+\epsilon} \delta(x - x_0)\, dx = 1, \qquad \int_{x_0-\epsilon}^{x_0+\epsilon} f(x)\, \delta(x - x_0)\, dx = f(x_0), \tag{7}$$

for any $\epsilon > 0$.

As an example in 2D, for a circular particle with radius r_0 centered at the origin the boundary is described by $\boldsymbol{\xi} = r_0(\cos\varphi, \sin\varphi)^T$ and expression (6) in polar coordinates gives

$$\boldsymbol{f}(x) = \boldsymbol{F}(\varphi)\delta(r - r_0). \tag{8}$$

Note, that the volume force $\boldsymbol{f}(x)$ diverges at $r = r_0$, whereas $\boldsymbol{F}(\varphi)$ is finite.

2.3 The total force on the particle

The total force exerted on the particle by the fluid is given by (2) as a surface integral of the fluid stress over the particle boundary $\partial P(t)$. Since in our formulation the volume force $\boldsymbol{f}$ is singular at the particle surface, care has to be taken when evaluating this integral. One rather has to consider the limit $\lim_{\epsilon\to 0} \int_{\partial P_{\epsilon^+}} \boldsymbol{n} \cdot \boldsymbol{\Pi}\, dO$ with ∂P_{ϵ^+} being any surface enclosing $\partial P(t)$ that tends to $\partial P(t)$ from outside in the limit $\epsilon \to 0$, i.e. $\lim_{\epsilon\to 0} \partial P_{\epsilon^+} = \partial P^+(t)$, see Fig. 1. The total force on the particle then is given by

$$\boldsymbol{F}_{\text{tot}} = -\lim_{\epsilon\to 0} \int_{\partial P_{\epsilon^+}} \boldsymbol{n} \,.\, \boldsymbol{\Pi}\, dO \tag{9}$$

In the above and in the following we denote by $\boldsymbol{\Pi}$ the stress tensor divided by the constant density ρ_0 and in all expressions involving the force we mean the force divided by the density.

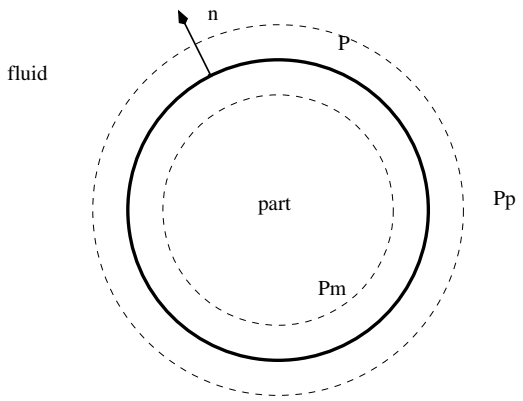

Figure 1: A circular particle with boundary ∂P (thick line) immersed in a fluid. ∂P is enclosed by the surfaces $\partial P_{\epsilon+}$ and $\partial P_{\epsilon-}$. The fluid equations are integrated over the volume V_ϵ enclosed by $\partial P_{\epsilon+}$ and $\partial P_{\epsilon-}$

Consider a second surface ∂P_{ϵ^-} that in the limit tends to $\partial P(t)$ from inside, i.e. $\lim_{\epsilon\to 0} \partial P_{\epsilon^-} = \partial P^-(t)$ as sketched in Fig. 1 and integrate the fluid equations over the volume V_ϵ enclosed by $\partial P_{\epsilon+}$ and $\partial P_{\epsilon-}$:

$$\int_{V_\epsilon(t)} (\partial_t \boldsymbol{u} + \nabla\cdot(\boldsymbol{uu}))dV = -\int_{V_\epsilon(t)} \nabla\cdot\boldsymbol{\Pi} dV + \int_{V_\epsilon(t)} \boldsymbol{f} dV \tag{10}$$

The second term on the left hand side and the first term on the right hand side can be converted into surface integrals by means of the Gauss theorem. Using (6) the volume integral over the force density also results in a boundary integral.

$$\begin{aligned}\int\limits_{V_\epsilon(t)} \partial_t \boldsymbol{u} dV + \int\limits_{\partial P_{\epsilon+}(t)} (\boldsymbol{n}\cdot\boldsymbol{u})\boldsymbol{u}\, dO - \int\limits_{\partial P_{\epsilon-}(t)} (\boldsymbol{n}\cdot\boldsymbol{u})\boldsymbol{u}\, dO = \\ - \int\limits_{\partial P_{\epsilon+}(t)} \boldsymbol{n}\cdot\boldsymbol{\Pi}\, dO + \int\limits_{\partial P_{\epsilon-}(t)} \boldsymbol{n}\cdot\boldsymbol{\Pi}\, dO + \int\limits_{\partial P(t)} \boldsymbol{F}(q,s,t)\, dO_{(q,s)}.\end{aligned} \tag{11}$$

Assume that $\boldsymbol{u}$ is continuous across ∂P and consider the limit $\epsilon \to 0$. Then the first term on the left hand side in (11) tends to zero because the integrand stays finite while $V_\epsilon \to 0$. The second and the third terms cancel. The only remaining terms are

$$\int\limits_{\partial P^+(t)} \boldsymbol{n}\cdot\boldsymbol{\Pi}\, dO = \int\limits_{\partial P(t)} \boldsymbol{F}(q,s,t)\, dO_{(q,s)} + \int\limits_{\partial P^-(t)} \boldsymbol{n}\cdot\boldsymbol{\Pi}\, dO. \tag{12}$$

Note, that the stress is discontinuous across ∂P. The total force the fluid exerts on the particle therefore can be written as

$$\boldsymbol{F}_{tot} = -\int\limits_{\partial P^+(t)} \boldsymbol{n}\cdot\boldsymbol{\Pi}\, dO = -\int\limits_{\partial P(t)} \boldsymbol{F}(q,s,t)\, dO_{(q,s)} - \lim_{\epsilon\to 0} \int\limits_{P_{\epsilon-}(t)} \nabla\cdot\boldsymbol{\Pi}\, dV. \tag{13}$$

An alternative way to express the last term on the right hand side is by using the momentum equation

$$-\lim_{\epsilon\to 0}\int\limits_{P_{\epsilon^-}(t)} \nabla\cdot\boldsymbol{\Pi}\,dV = \lim_{\epsilon\to 0}\int\limits_{P_{\epsilon^-}(t)} (\partial_t\boldsymbol{u}+(\boldsymbol{u}\cdot\nabla)\boldsymbol{u})dV = \frac{d}{dt}\int\limits_{P(t)}\boldsymbol{u}dV. \tag{14}$$

The volume force does not appear in this equation as it vanishes in $P_{\epsilon^-(t)}$. In the last step the limit $\epsilon\to 0$ was evaluated as $\boldsymbol{u}$ and its time derivative stay finite. So we can give the following expressions for the total force

$$\boldsymbol{F}_{tot} = -\int\limits_{\partial P^+(t)} \boldsymbol{n}\cdot\boldsymbol{\Pi}\,dO \tag{15}$$

$$= -\int\limits_{\partial P(t)} \boldsymbol{F}(q,s,t)\,dO_{(q,s)} - \lim_{\epsilon\to 0}\int\limits_{P_{\epsilon^-}(t)} \nabla\cdot\boldsymbol{\Pi}\,dV \tag{16}$$

$$= -\int\limits_{\partial P(t)} \boldsymbol{F}(q,s,t)\,dO_{(q,s)} + \frac{d}{dt}\int\limits_{P(t)} \boldsymbol{u}\,dV \tag{17}$$

Uhlmann (2004) shows for the two-dimensional case that the inertia force per fluid density of the fluid inside the particle domain is determined by the acceleration of the particle times its volume, i.e. $\frac{d}{dt}\int\limits_{P(t)}\boldsymbol{u}\,dV = \frac{d\boldsymbol{U}_{(t)}}{dt}V_{P(t)}$. In this work we will compute the total force by (17) and use the last term in (16) to check the convergence upon grid refinement.

Analogous computations can be carried out for the total torque per fluid density

$$\boldsymbol{M}_{tot} = -\int\limits_{\partial P^+(t)} (\boldsymbol{x}-\boldsymbol{X}(t))\times(\boldsymbol{n}\cdot\boldsymbol{\Pi})\,dO \tag{18}$$

$$= -\int\limits_{\partial P(t)} (\boldsymbol{\xi}(q,s,t)-\boldsymbol{X}(t))\times\boldsymbol{F}(q,s,t)\,dO_{(q,s)} - \lim_{\epsilon\to 0}\int\limits_{P_{\epsilon^-}(t)} (\boldsymbol{x}-\boldsymbol{X}(t))\times(\nabla\cdot\boldsymbol{\Pi})\,dV \tag{19}$$

$$= -\int\limits_{\partial P(t)} (\boldsymbol{\xi}(q,s,t)-\boldsymbol{X}(t))\times\boldsymbol{F}(q,s,t)\,dO_{(q,s)} + \frac{d}{dt}\int\limits_{P(t)} (\boldsymbol{x}-\boldsymbol{X}(t))\times\boldsymbol{u}\,dV. \tag{20}$$

2.4 Summary of the equations

Summarizing, we solve the following problem

$$\partial_t \boldsymbol{u} + \boldsymbol{u} \cdot \nabla \boldsymbol{u} = -\nabla p + \nu \Delta \boldsymbol{u} + \boldsymbol{f} \quad \text{in} \quad D, \tag{21}$$

$$\nabla \cdot \boldsymbol{u} = 0 \quad \text{in} \quad D, \tag{22}$$

$$\begin{aligned} \boldsymbol{\mathcal{U}}(q,s,t) &:= \int_D \boldsymbol{u}(x,t)\delta^3(\boldsymbol{x} - \boldsymbol{\xi}(q,s,t))dV \\ &= \boldsymbol{U}(t) + \boldsymbol{\Omega}(t) \times (\boldsymbol{\xi}(q,s,t) - \boldsymbol{X}(t)), \quad \boldsymbol{\xi}(q,s,t) \in \partial P(t), \end{aligned} \tag{23}$$

$$\frac{d\boldsymbol{X}(t)}{dt} = \boldsymbol{U}(t), \tag{24}$$

$$M\frac{d\boldsymbol{U}}{dt} = -\rho_0 \int_{\partial P} \boldsymbol{F}(q,s,t)\, dO_{(q,s)} + \rho_0 \frac{d}{dt} \int_{P(t)} \boldsymbol{u}\, dV, \tag{25}$$

$$\boldsymbol{I} \,.\, \frac{d\boldsymbol{\Omega}}{dt} = -\rho_0 \int_{\partial P(t)} (\boldsymbol{\xi}(q,s,t) - \boldsymbol{X}(t)) \times \boldsymbol{F}(q,s,t)\, dO_{(q,s)} + \rho_0 \frac{d}{dt} \int_{P(t)} (\boldsymbol{x} - \boldsymbol{X}) \times \boldsymbol{u}\, dV, \tag{26}$$

with $\boldsymbol{f}$ related to $\boldsymbol{F}$ by (6) All the numerical results presented in this paper are performed for a particle with prescribed motion. In the following we will therefore consider the discretization of (21) – (23).

3 Numerical Implementation

3.1 Direct forcing

The Navier-Stokes equations (21) and (22) are solved on a Cartesian grid. Following Uhlmann (2005) we evenly distribute N_L marker points at positions $\{\boldsymbol{X}_l\}$ over the surface of the particle, such that the distance between two marker points approximately equals the grid spacing h of the Cartesian grid, $\Delta X_l := |\Delta \boldsymbol{X}_l| \approx h$. The marker points move with the particle, whereas the Cartesian grid is fixed. The no-slip condition (23) will be imposed at the marker points by a suitable force field $\boldsymbol{F}(X_l)$. In the following we restrict ourselves to the two dimensional case.

As proposed by Peskin (2002) we use a distributed δ-function both to interpolate quantities from the Cartesian grid to the marker points as well as to spread the force $\boldsymbol{F}(X_l)$ back to the Eulerian grid:

$$\boldsymbol{U}(X_l) = \sum_{x_i} \boldsymbol{u}(x_i)\delta_h^2(\boldsymbol{x}_i - \boldsymbol{X}_l)h^2 \tag{27}$$

$$\boldsymbol{f}(x_i) = \sum_{X_l} \boldsymbol{F}(X_l)\delta_h^2(\boldsymbol{x}_i - \boldsymbol{X}_l)\Delta X_l \tag{28}$$

where $\delta_h^2(\boldsymbol{x}) = \delta(x)\delta(y)$ and $\delta_h(x) := \frac{1}{h}\phi(x/h)$.

As our solver is based on the finite volume method (FVM) on a collocated grid we use the 4-point approximation proposed by Peskin (2002) for the δ-kernel

$$\phi(r) = \begin{cases} \frac{1}{8}(3 - 2|r| + \sqrt{1 + 4|r| - 4r^2}) & \text{for} \quad 0 \le |r| < 1 \\ \frac{1}{8}(5 - 2|r| - \sqrt{-7 + 12|r| - 4r^2}) & \text{for} \quad 1 \le |r| < 2 \\ 0 & \text{for} \quad 2 \le |r| \end{cases} \tag{29}$$

For a uniform grid with $\Delta x = \Delta y = h$ the above approximation satisfies

$$\sum_{x_i} \delta_h(\boldsymbol{x}_i - \boldsymbol{X}_l)h^2 = 1, \qquad \sum_{x_i} (\boldsymbol{x}_i - \boldsymbol{X}_l)\delta_h(\boldsymbol{x}_i - \boldsymbol{X}_l)h^2 = 0 \qquad \forall \boldsymbol{X}_l \tag{30}$$

From the above property it follows that

$$\sum_{x_i} \boldsymbol{f}(x_i)h^2 = \sum_{X_l} \boldsymbol{F}(X_l)\Delta X_l \tag{31}$$

i.e. the total force representing the particle in the fluid can be computed either on the Cartesian grid or on the Lagrangian marker points.

Note that the composition of the interpolation and spreading operations (27) and (28) in combination with (29) does not yield the identity as it would for the continumm Dirac δ-function. E.g. interpolation of the force density $\boldsymbol{f}(x_i)$ to the marker points X_l is given by

$$\boldsymbol{G}(X_l) := \sum_{x_i} \boldsymbol{f}(x_i)\delta_h^2(\boldsymbol{x}_i - \boldsymbol{X}_l)h^2 = \underbrace{\sum_{X_m}\sum_{x_i} \delta^2(\boldsymbol{x}_i - \boldsymbol{X}_m)\delta^2(\boldsymbol{x}_i - \boldsymbol{X}_l)h^2\Delta X_m}_{\mathcal{M}_{lm}} \boldsymbol{F}(X_m) \neq \boldsymbol{F}(X_l), \tag{32}$$

as the $N_l \times N_l$ matrix $\mathcal{M}_{lm}$ is not the identity matrix.

There are several possibilities to numerically compute an approximation to the force density $\boldsymbol{f}$. In this paper we follow the approach of Taira and Colonius (2007), who consider the force as a Lagrange multiplier that guarantees no-slip at $\partial P(t)$. They use a fractional step scheme and solve a modified Poisson system simultaneously for the pressure p and the force $\boldsymbol{F}$.

The idea is the following: consider an intermediate velocity $\tilde{\boldsymbol{u}}$ obtained from solving the momentum equation

$$\mathsf{A}\tilde{\boldsymbol{u}} = \mathbf{rhs}^n, \tag{33}$$

where A is a discretization matrix that depends on the details of the scheme and $\mathbf{rhs}^n$ only depends on the variables at the previous time step t^n. An approximation to the velocity $\boldsymbol{u}^{n+1}$

$$\boldsymbol{u}^{n+1} = \tilde{\boldsymbol{u}} - \mathcal{A}^{-1}\nabla p + \mathcal{A}^{-1}\boldsymbol{f} \tag{34}$$

($\mathcal{A}^{-1}$ being a numerical approximation to the inverse of A) would be divergence free and would satisfy the no-slip condition at $\boldsymbol{X}_l$ if p and $\boldsymbol{F}$ solve the the following coupled system

$$\sum_{i,j} (\tilde{\boldsymbol{u}}(x_i) - (\mathcal{A}^{-1})_{ij}\nabla p(x_j))\delta^2(\boldsymbol{x}_i - \boldsymbol{X}_l) + \sum_m \sum_{i,j} (\mathcal{A}^{-1})_{ij}\delta^2(\boldsymbol{x}_j - \boldsymbol{X}_m)\delta^2(\boldsymbol{x}_j - \boldsymbol{X}_l)h^2\Delta X_m \boldsymbol{F}(X_m) = \boldsymbol{U}_{\text{part}}(X_l), \tag{35}$$

$$\nabla \cdot (\mathcal{A}^{-1}\nabla p - \mathcal{A}^{-1}\boldsymbol{f}) = \nabla \cdot \tilde{\boldsymbol{u}}. \tag{36}$$

$\boldsymbol{f}$ in (36) is understood to be expressed in terms of $\boldsymbol{F}$ by (28). Equation (35) represents N_L equations stating that $\boldsymbol{U}^{n+1}(X_l) = \boldsymbol{U}_{\text{part}}(X_l)$, the derivation being analogous to (32). Equation (36) is a Poisson equation, modified by the presence of the force, guaranteeing $\nabla \cdot \boldsymbol{u}^{n+1} = 0$.

3.2 `icoFoam` – OpenFOAM's standard solver for incompressible flow

We base our code on the free and open-source software OpenFOAM (Open Field Operation and Manipulation), in particular we use the `1.5-dev` version from the Project project. The software provides the finite volume method to solve partial differential equations on an unstructured grid.

The solver `icoFoam` for the incompressible Navier-Stokes equations is based on a fractional step method using the PISO algorithm (see e.g. Jasak, 1996). It involves a momentum predictor

$$\text{ddt}(\boldsymbol{u}) + \nabla(\boldsymbol{u}^n\boldsymbol{u}^*) - \nu\Delta\boldsymbol{u}^* = -\nabla p^n, \tag{37}$$

and a correction loop involving a Poisson equation for the pressure to compute a divergence free velocity $\boldsymbol{u}^{n+1}$. The correction loop roughly runs along the following lines: the left hand side of (37) is discretized as

$$\mathcal{A}(\boldsymbol{u}^n)\boldsymbol{u}^* - (\mathcal{H}(u^n)\boldsymbol{u}^* + \frac{\boldsymbol{u}^n}{\Delta t}) \tag{38}$$

where $\mathcal{A}$ and $\mathcal{H}$ are matrices that depend on the solution of the previous time-step. $\mathcal{A}$ is diagonal, whereas $\mathcal{H}$ only has off-diagonal entries. Equation (37) is solved and a series of corrections is applied by setting

$$\begin{aligned} \tilde{\boldsymbol{u}} &= \mathcal{A}^{-1}(\mathcal{H}\boldsymbol{u}^i + \boldsymbol{u}^n/\Delta t) \\ \nabla \cdot (\mathcal{A}^{-1}\nabla p) &= \nabla \cdot \tilde{\boldsymbol{u}} \\ \boldsymbol{u}^{i+1} &= \tilde{\boldsymbol{u}} - \mathcal{A}^{-1}\nabla p \end{aligned} \tag{39}$$

for $0 \leq i < \texttt{ncorr}$, with $\boldsymbol{u}^0 = \boldsymbol{u}^*$ and $\boldsymbol{u}^{\texttt{ncorr}} = \boldsymbol{u}^{n+1}$. In the spirit of Rhie-Chow care is taken with the discretization and interpolation in (39) in order to avoid unphysical oscillations in the solution (see e.g. Peng Kärrholm (2008)).

We choose $\text{ddt}(u) := (u^* - u^n)/\Delta t$ for all the runs in this paper and use linear interpolation when interpolating from cell centers to cell faces. On a Cartesian grid the above algorithm with these settings is second order in space and first order in time.

3.3 Modification of `icoFoam` to include direct forcing

As a first step we implemented the immersed boundary method with direct forcing in OpenFOAM in two spatial dimensions by extending `icoFoam`. To this end a new C++ class was defined to handle the particle, it's marker points as well as the interpolation and spreading operations.

The only modifications to the algorithm of `icoFoam` necessary to incorporate direct forcing are to add the force $\boldsymbol{f}$ to the momentum predictor (37) and to extend the correction loop (39) to the system (35)–(36). However, we do not solve (35)–(36) as a coupled system but decouple the equations by fixing p to it's latest value when solving (35) for $\boldsymbol{F}$ and vice versa. The correction loop is run through twice, i.e. $\texttt{ncorr} = 2$. In Sec. 4 we will demonstrate that this way the no-slip condition is fulfilled satisfactorily.

3.4 Discretization of the total force

The computation of the total force and moment on the particle is performed by discretizing (17) and (20), where the last term in (17) is replaced by $\frac{d\boldsymbol{U}_{(t)}}{dt}V_{P(t)}$.

We also compute the last term in (16) which should equal $\frac{d\boldsymbol{U}_{(t)}}{dt}V_{P(t)}$, in order to have an indicator for convergence of the method upon grid refinement close to the particle. However, as the stress jumps across ∂P we do not approximate the integral by a sum over the whole particle region but exclude the points close to the boundary, where the δ-kernel is non-zero. I.e. we perform the sum over a region $\tilde{P} \subset P(t)$ that results from subtracting the region of support of the δ-kernel from $P(t)$.

4 Numerical Results

In the following we present a number of two-dimensional test cases of the flow over a cylinder with prescribed motion immersed in a fluid. We introduce the particle Reynolds number $\mathrm{Re}_p = dV/\nu$, with d being the diameter of the cylinder and V being a characteristic velocity of the test case. The time unit is given by $T = d/V$. For all the examples the relative movement of the particle to the flow is in the x-direction. The drag and lift coefficients are given by

$$C_D = \frac{1}{\frac{1}{2}\rho_0 V^2 d}F^x_{tot}, \quad C_L = \frac{1}{\frac{1}{2}\rho_0 V^2 d}F^y_{tot}. \tag{40}$$

For all tests we selected a uniform grid close to the cylinder with a grid spacing h. The grid is stretched away from the cylinder up to some maximum grid spacing $\Delta x_{\max}$ at the outer boundaries (in general $\Delta x \neq \Delta y$ in the outer region).

4.1 Stationary cylinder at $\mathrm{Re}_p = 40$

We consider a stationary cylinder in uniform cross flow $u^x = V$. The computational domain is taken to be $D = 40d \times 40d$ with the cylinder centered at $\boldsymbol{X} = (20d, 20d)$. We set $u^x = V, u^y = 0$ at all boundaries except for the outlet. At these boundaries the normal gradient of the pressure is required to vanish. At the outlet a homogenous Neumann condition is applied for the velocity, i.e. $\partial_x u^x = \partial_x u^y = 0$ and the pressure is set to 0.

We consider three spatial resolutions with 210×210, 318×318 and 358×358 grid points. The grid spacing close to the cylinder is $h = 0.05d$, $h = 0.025d$ and $h = 0.0125d$, the maximal grid spacing at the outer boundary is $\Delta x_{max} = 0.8d$ for all runs. The time step is $\Delta t = 0.005T$ for all resolutions corresponding to a maximal Courant number of $0.116994, 0.233872$ and 0.4677.

At $\mathrm{Re}_p = 40$ the solution is stationary. Table 1 lists the drag coefficient for the three resolutions and compares them to results of Taira and Colonius (2007) as well as to an estimated mean value from the experimental results of Tritton (1959). Comparing to the numerical results of Taira and Colonius (2007) as well as those cited in their paper the present results show a slight over-prediction of the drag by $\sim 1\%$ for the highest resolution.

As the cylinder is stationary the integral of the divergence of the stress over the particle region should vanish in the continuum limit. Table 1 lists the numerical values of the integral. For

this test case the quantity converges roughly quadratically. The error in the no-slip boundary condition at the particle surface is less than 2×10^{-8} for all resolutions.

	C_D	$\int_{P_{\epsilon^-}} \nabla \cdot \mathbf{\Pi} dV$
$h = 0.05d$	1.593	2.27×10^{-3}
$h = 0.025d$	1.571	0.65×10^{-3}
$h = 0.0125d$	1.56	0.14×10^{-3}
Taira and Colonius (2007)	1.54	
Tritton (1959)	1.59	

Table 1: The drag coefficient for a stationary cylinder at $\mathrm{Re}_p = 40$. Comparisons to the numerical results of Taira and Colonius (2007), who use a domain size of $D = 60d \times 60d$ and a minimal grid spacing of $h = 0.02$ and an estimated mean value from the experimental results of Tritton (1959). The second column lists the contribution from the stress in the particle region that should vanish in the continuum limit.

4.2 Stationary cylinder at $\mathbf{Re}_p = 100$

At a particle Reynolds number $\mathrm{Re}_p = 100$ the flow is periodic in time. We compute the drag and lift coefficients and the Strouhal number $\mathrm{St} = fd/V$ with f being the frequency of oscillation.

We apply the same boundary conditions as in the previous subsection. Concerning the domain size and the resolution we use similar parameters as Uhlmann (2005) in order to compare to his results. The domain is taken to be $D = 26.67d \times 26.67d$ with the particle located at $\boldsymbol{X} = (6.17d, 13.33d)$. The grid spacing close to the cylinder is $h = 0.025d$, the maxium grid spacing at the outer boundaries is $\Delta x_{max} = 0.5d$ giving rise to a total number of 274×232 grid points. The time step is $\Delta t = 0.003T$ yielding a maxium Courant number of 0.1632.

Table 2 lists the mean drag coefficient, the variations in drag and lift as well as the Strouhal number. As can be seen the present results show an over-prediction of all the quantities (except for the variation of the lift) when compared to the results of Liu et al. (1998). The over prediction of mean drag and lift variation is more than 10%. This might be due e.g. to insufficient resolution and/or the domain size being to small.

	$\bar{C}_D$	C'_D	C'_L	St
$h = 0.025d$	1.475	± 0.012	± 0.379	0.171
Uhlmann (2005)	1.501	± 0.011	± 0.349	0.172
Liu et al. (1998)	1.350	± 0.012	± 0.339	0.165

Table 2: Mean value of the drag $\bar{C}_D$, variations of drag and lift C'_D and C'_L as well as the Strouhal number St for a cylinder in uniform flow at $\mathrm{Re}_p = 100$. Uhlmann (2005) uses the same domain and a uniform grid with $h = 0.026d$.

4.3 Oscillating cylinder in quiescent fluid at $\mathbf{Re}_p = 100$, $KC = 5$

We consider a circular cylinder oscillating in x-direction in an otherwise quiescent fluid. The cylinder oscillates with an amplitude $A = 5d/V$ and a frequency $f = V/5d$ such that the prescribed motion is given by $X(t) = X_0 - 5d\sin(2\pi Vt/(5d))/(2\pi)$. The characteristic velocity V corresponds to the maximal velocity attained by the cylinder. The value of A corresponds to a Keulegan-Carpenter number of $KC = 5$.

We choose a domain of $D = 50d \times 50d$ with the cylinder oscillating around $\boldsymbol{X}_0 = (25d, 25d)$. At all boundaries we set the velocity to zero and put homogenous Neumann conditions for the pressure.

We consider three grid resolutions with $h = 0.05d, h = 0.025d$ and $h = 0.0125d$ in the vicinity of the cylinder and a maximum grid spacing of $\Delta x_{\max} = 0.5d$ giving rise to a total number of $342 \times 342, 486 \times 486$ and 692×692 grid-points. The time step for the two coarser resolutions was $\Delta t = 0.003T$ leading to a maxium Courant number of 0.085, respectively 0.149. For the finest resolution the maximum Courant number was fixed to 0.2 resulting in the time step being in the interval $\Delta t \in [0.002T, 0.0035T]$.

Fig. 2 shows the drag coefficient as a function of time. As can be seen the drag exhibits quite large unphysical spikes close to the turning points. However these spikes decrease when increasing the resolution as can be seen more clearly in Fig. 3. The small oscillations clearly visible at the coarsest resolution stem from the movement of the marker points through the Cartesian grid. The amplitude of theses oscillations decreases with increasing resolution. The maximal value of the drag is over predicted by about 3% as compared to the results given in Kim and Choi (2006). Fig. 3 shows the error in $\lim_{\epsilon\to 0} \int\limits_{P_{\epsilon^-}(t)} \nabla \cdot \mathbf{\Pi}\, dV - \frac{d\boldsymbol{U}_{(t)}}{dt} V_{P(t)}$ that converges to zero linearly. In Fig. 4 we plot the maximal violation of the no-slip boundary condition. Convergence of the error is quadratic between the second and third resolution.

5 Conclusions and Outlook

In Section 4 we demonstrated that our numerical implementation of direct forcing in OpenFOAM gives good results for 2D flow problems over a cylinder with prescribed motion. The no-slip boundary condition at the particle surface is fulfilled satisfactorily. Drag and lift coefficients when compared to literature show a slight over-prediction with the exception of the problem in Section 4.2 where the over-prediction exceeds 10%. The latter needs further investigation. For the oscillating cylinder the drag coefficient exhibits an unphysical "spike" close to the turning point of the cylinder, which reduces with increasing grid resolution. We also showed that the error in $\lim_{\epsilon\to 0} \int\limits_{P_{\epsilon^-}(t)} \nabla \cdot \mathbf{\Pi}\, dV$ converges to zero.

As a next step for validating the implementation we will consider a rotating cylinder as well test cases for the fully coupled problem.

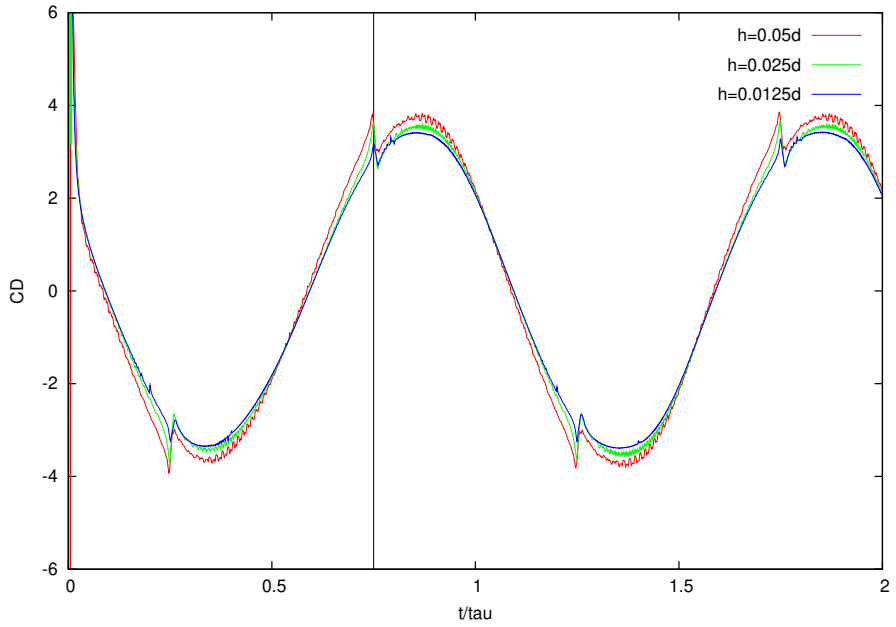

Figure 2: The drag coefficient for the three local resolutions as a function of time. τ denotes the period of the oscillation. The vertical line at $t/\tau = 0.75$ indicates one of the turning points. The drag coefficient shows unphysical spikes close to the turning point. As can be seen in the next figure these spikes decrease with increasing resolution.

6 Acknowledgments

Support from Lam Research Corporation and the Austrian Research Promotion Agency (FFG) under grant number 819320/1788 is gratefully acknowledged.

References

Hrvoje Jasak. *Error analysis and estimation for the Finite Volume method with applications to fluid flows*. PhD thesis, Imperial College, University of London, 1996.

Dokyun Kim and Haecheon Choi. Immersed boundary method for flow around an arbitrarily moving body. *J. Comput. Phys.*, 212:662 – 680, 2006.

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

OpenFOAM. URL `http://www.opencfd.co.uk/openfoam/`.

Fabian Peng Kärrholm. *Numerical Modelling of Diesel Spray Injection, Turbulence Interaction and Combustion*. PhD thesis, Chalmers University of Technology, Göteborg, 2008.

Charles S. Peskin. *Flow patterns around heart valves: a digital computer method for solving the equations of motion*. PhD thesis, Albert Einstein College of Medicine, 1972.

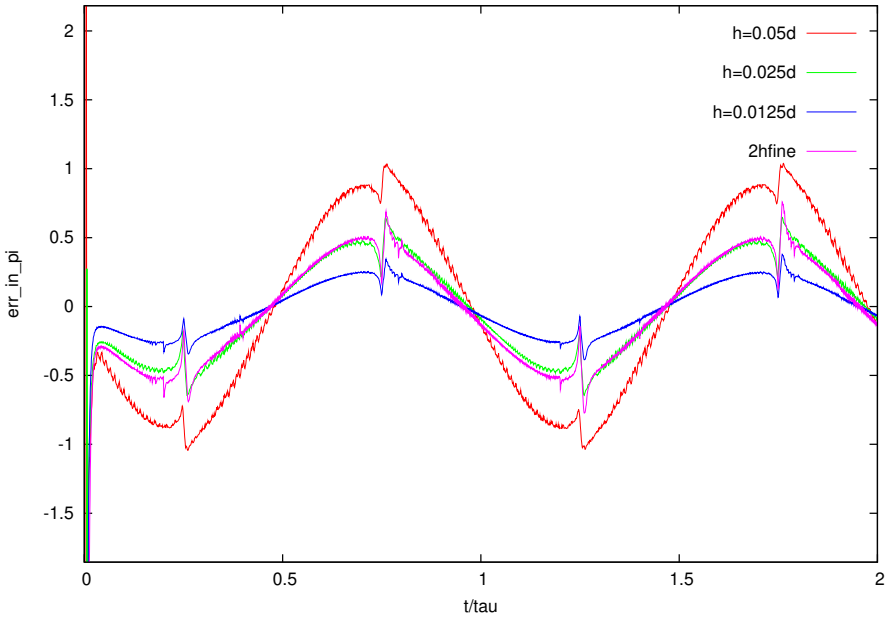

Figure 3: The error $\lim_{\epsilon \to 0} \int_{P_{\epsilon -}(t)} \nabla \cdot \mathbf{\Pi} \, dV - \frac{d\boldsymbol{U}(t)}{dt} V_{P(t)}$ for the three local resolutions. The smallest resolution is multiplied by a factor of 2 to lie on top of the middle resolution which demonstrates first order convergence of the error. As can be seen the magnitude of the unphysical 'spike' also decreases with increasing resolution.

Charles S. Peskin. The immersed boundary method. *Acta Numerica*, 11:479–517, 2002.

The OpenFOAM(R) Extend Project. http://sourceforge.net/projects/openfoam-extend/. URL `http://sourceforge.net/projects/openfoam-extend/`.

Kunihiko Taira and Tim Colonius. The immersed boundary method: A projection approach. *J. Comput. Phys.*, 225:2118 – 2137, 2007.

D. J. Tritton. Experiments on the flow past a circular cylinder at low reynolds numbers. *J. Fluid Mech.*, 6:547 – 567, 1959.

Markus Uhlmann. New results on the simulation of particulate flows. Technical report, CIEMAT, Madrid, 2004.

Markus Uhlmann. An immersed boundary method with direct forcing for the simulation of particulate flows. *J. Comput. Phys.*, 209:448–476, 2005.

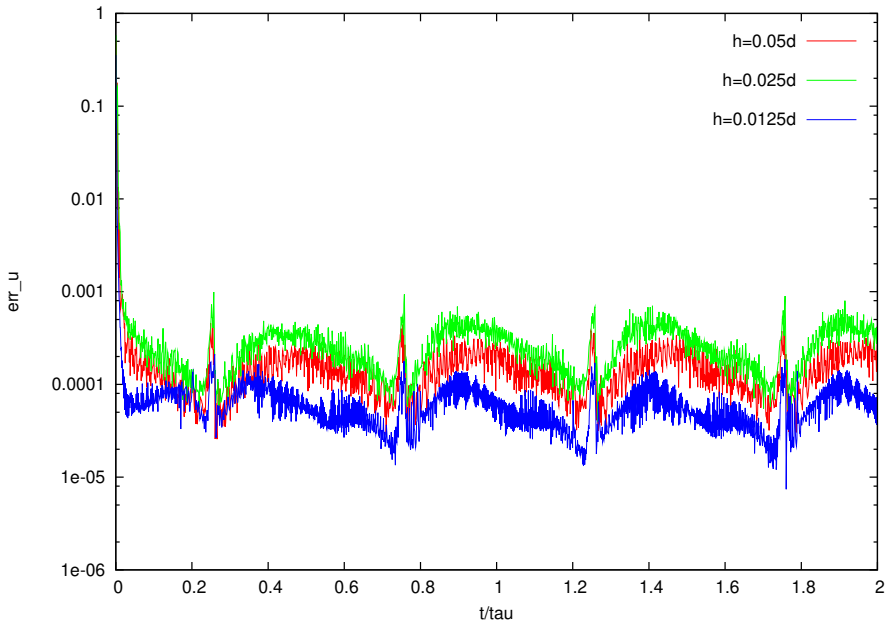

Figure 4: The violation of the no-slip boundary conditions for the three local resolutions. The error is maximal close to the turning points. Quadratic convergence can be observed between the second and third resolution.