No. 684 July 2025

A Chimera domain decomposition method with weak Dirichlet-Robin coupling for finite element simulation of particulate flows

R. Münster, O. Mierka, D. Kuzmin, S. Turek

ISSN: 2190-1767

A Chimera domain decomposition method with weak Dirichlet-Robin coupling for finite element simulation of particulate flows

Raphael Münster, Otto Mierka, Dmitri Kuzmin*, Stefan Turek

Institute of Applied Mathematics (LS III), TU Dortmund University
Vogelpothsweg 87, D-44227 Dortmund, Germany

Abstract

We introduce a new multimesh finite element method for direct numerical simulation of incompressible particulate flows. The proposed approach falls into the category of overlapping domain decomposition / Chimera / overset grid meshes. In addition to calculating the velocity and pressure of the fictitious fluid on a fixed background mesh, we solve the incompressible Navier-Stokes equations on body-fitted submeshes that are attached to moving particles. The submesh velocity and pressure are used to calculate the hydrodynamic forces and torques acting on the particles. The coupling with the background velocity and pressure is enforced via (i) Robin-type boundary conditions for an Arbitrary-Lagrangian-Eulerian (ALE) formulation of the submesh problems and (ii) a Dirichlet-type distributed interior penalty term in the weak form of the background mesh problem. The implementation of the weak Dirichlet-Robin coupling is discussed in the context of discrete projection methods. Detailed numerical studies are performed for standard test problems involving fixed and moving immersed objects. A comparison with fictitious boundary methods illustrates significant gains in the accuracy of drag and lift approximations.

Keywords: particulate flows, fictitious domains, embedded boundaries, finite element methods, overlapping grids, Chimera domain decomposition, Dirichlet–Robin coupling

1. Introduction

Numerical methods for direct numerical simulation (DNS) of incompressible flows around moving rigid particles can be classified into fixed mesh and deforming mesh approaches [5]. Prominent representatives of the latter family include immersed boundary methods [11, 16, 23] and fictitious domain formulations in which the rigid body motion inside the particles is enforced using distributed Lagrange multipliers (DLM, [3, 4, 15]), fictitious (surrogate, shifted, unfitted) boundary methods

*Corresponding author

Email address: {raphael.muenster,otto.mierka,dmitri.kuzmin,stefan.turek}@math.tu-dortmund.de (Raphael Münster, Otto Mierka, Dmitri Kuzmin*, Stefan Turek)

[12, 25, 26, 27, 28], subspace projections [1, 17], and other non-DLM alternatives [24, 19]. The resolving power of such approaches can be greatly improved by using mesh deformation techniques [1, 14, 29] or overlapping domain decomposition with moving submeshes [2, 6, 7, 8, 10]. The coupling conditions in the overlap region can again be enforced in different ways. Houzeaux and Codina [8] propose a Chimera method with Dirichlet/Neumann(Robin) coupling such that Dirichlet-type conditions are strongly enforced at nodal points inside and around immersed objects. The multimesh finite element method developed by Dokken et al. [2] achieves the coupling effect by incorporating suitable stabilization terms into the discontinuous Galerkin (DG) weak forms of interacting subproblems.

A common disadvantage of all fixed-mesh algorithms that constrain the velocity of a fictitious fluid at discrete locations is the lack of continuous dependence on the data. Indeed, a small displacement of a particle can activate or deactivate the Dirichlet constraint. Moreover, the volume of the constrained region changes abruptly leading to nonphysical temporal oscillations in the pressure field and, as a consequence, in forces acting on the particles. In the present paper, we cure this deficiency of Chimera-type domain decomposition methods by using a distributed interior penalty term instead of strongly imposed nodal constraints. The weak form of our background mesh problem differs from the unfitted finite element method presented in [2] in the structure of employed stabilization terms and in the way in which they are incorporated into the discrete projection method for the Navier–Stokes system. Our algorithm is simpler than the DG method from [2] and requires fewer degrees of freedom because we are using a continuous approximation to the velocity field. The results of our numerical experiments show that our modification of the Chimera method from [8] is robust and capable of delivering accurate DNS results at a fraction of the cost that a fixed-mesh fictitious boundary method would require.

2. Fictitious domain formulation

Let $\Omega \subset \mathbb{R}^d,\ d \in \{2,3\}$ be a fixed fictitious domain. In our particulate flow model, Ω is filled with an incompressible Newtonian fluid that carries a suspension of N_p rigid particles (balls)

$$B_k(t) = \{\mathbf{x} \in \mathbb{R}^d \,:\, |\mathbf{x} - \mathbf{X}_k(t)| < R_k\}.$$

We subdivide Ω into $\Omega_p(t) := \bigcup_{k=1}^{N_p} B_k(t)$ and the subdomain $\Omega_f(t) := \Omega \backslash \bar{\Omega}_p(t)$ occupied by the fluid. The evolution of the fluid-particle mixture is governed by the system [14, 27]

$$\rho_f \left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} \right) = -\nabla p + \nabla \cdot (2\mu_f \mathbf{D}(\mathbf{u})) \quad \text{in } \Omega_f(t), \tag{1a}$$

$$\nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \tag{1b}$$

$$\mathbf{u} = \mathbf{U} \quad \text{on } \bar{\Omega}_p(t) \tag{1c}$$

of generalized incompressible Navier–Stokes equations. Here $\mathbf{u}$ is the velocity of the fluid, $\mathbf{U}$ is the velocity of rigid body motion inside the particles, p is the pressure, and

$$\mathbf{D}(\mathbf{u}) = \frac{1}{2}(\nabla \mathbf{u} + \nabla \mathbf{u}^\top)$$

is the deformation rate tensor. The constant density and dynamic viscosity of the fluid phase are denoted by ρ_f and μ_f, respectively. For $\mathbf{x} \in \bar{B}_k(t)$, the velocity of the fictitious fluid is given by

$$\mathbf{U}(\mathbf{x}, t) = \mathbf{U}_k(t) + \boldsymbol{\omega}_k(t) \times (\mathbf{x} - \mathbf{X}_k), \tag{2}$$

where $\mathbf{U}_k$ is the translational velocity and $\boldsymbol{\omega}_k$ is the angular velocity of the kth particle. The density, volume, and moment of inertia tensor of this particle are denoted by ρ_k, V_k, and $\mathbb{I}_k$, respectively.

For simplicity, we assume that the particles do not collide with each other or with solid walls. Therefore, we do not include repulsive or lubrication forces in the Newton–Euler equations

$$\rho_p V_k \frac{\mathrm{d}\mathbf{U}_k}{\mathrm{d}t} = \mathbf{F}_k + (\rho_p - \rho_f) V_k \mathbf{g}, \tag{3a}$$

$$\mathbb{I}_k \frac{\mathrm{d}\boldsymbol{\omega}_k}{\mathrm{d}t} = \mathbf{T}_k - \boldsymbol{\omega}_k \times (\mathbb{I}_k \boldsymbol{\omega}_k), \tag{3b}$$

where $\mathbf{g}$ is the gravitational acceleration. The hydrodynamic force $\mathbf{F}_k$ and torque $\mathbf{T}_k$ are defined by

$$\mathbf{F}_k = -\int_{\partial B_k} \boldsymbol{\sigma}\mathbf{n}\mathrm{d}\mathbf{s}, \qquad \mathbf{T}_k = -\int_{\partial B_k} (\mathbf{x} - \mathbf{X}_k) \times (\boldsymbol{\sigma}\mathbf{n})\mathrm{d}\mathbf{s}. \tag{4}$$

Here $\mathbf{n}$ denotes the unit outward normal and $\boldsymbol{\sigma} = -p\mathbf{I} + 2\mu_f \mathbf{D}(\mathbf{u})$, where $\mathbf{I}$ is the identity tensor.

The problem statement is completed by imposing appropriate initial and boundary conditions. By default, we prescribe a given velocity profile at the inlet, the zero-stress condition at the outlet, and the no-slip condition on solid walls. The rigid body motion constraint (1c) defines both the no-slip Dirichlet boundary data for $\partial\Omega_p(t)$ and the velocity of the fictitious fluid contained in $\Omega_p(t)$.

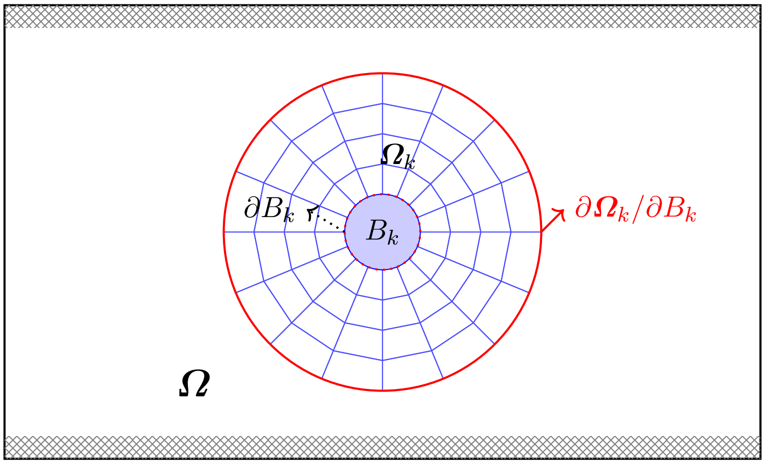


Figure 1: Schematic setup of the Chimera submesh.

3. Chimera domain decomposition

In our multimesh numerical method for solving the coupled problems (1) and (3), we calculate $\mathbf{F}_k$ and $\mathbf{T}_k$ using finite element approximations to $\mathbf{u}$ and p on the body-fitted subdomains

$$\hat{\Omega}_k(t) = \{\mathbf{x} \in \mathbb{R}^d \,:\, R_k < |\mathbf{x} - \mathbf{X}_k(t)| < R_k + H_k\},$$

which are embedded into Ω as shown in Fig. 1. In this work, we assume that $\hat{\Omega}_k(t) \cap B_j(t) = \emptyset$ for $k \neq j$. The general case of overlapping domains is considered in [2, 10].

If the particle $B_k(t)$ were a planet, the associated subdomain $\hat{\Omega}_k(t)$ could be interpreted as the atmosphere of that planet. We denote the atmospheric velocity and pressure fields by $\hat{\mathbf{u}}$ and $\hat{p}$, respectively. The parameter $H_k > 0$ determines the width of the atmospheric layer around $B_k(t)$.

Following Houzeaux and Codina [8], we perform *iteration by subdomains* using coupling conditions of Dirichlet–Robin type. That is, the background fields $\mathbf{u}$ and p influence the solution of

$$\rho_f\left(\frac{\partial \hat{\mathbf{u}}}{\partial t} + \hat{\mathbf{u}}\cdot\nabla\hat{\mathbf{u}}\right) = -\nabla\hat{p} + \nabla\cdot(2\mu_f\mathbf{D}(\hat{\mathbf{u}})) \quad \text{in } \hat{\Omega}_k(t), \tag{5a}$$

$$\nabla\cdot\hat{\mathbf{u}} = 0 \quad \text{in } \hat{\Omega}_k(t), \tag{5b}$$

$$\hat{\mathbf{u}} = \mathbf{U} \quad \text{on } \partial B_k(t), \tag{5c}$$

$$\hat{\boldsymbol{\sigma}}\mathbf{n} - \alpha(\hat{\mathbf{u}}\cdot\mathbf{n})\hat{\mathbf{u}} = \boldsymbol{\sigma}\mathbf{n} - \alpha(\mathbf{u}\cdot\mathbf{n})\mathbf{u} \quad \text{on } \partial\Omega_k(t)\backslash\partial B_k(t) \tag{5d}$$

by providing the data of the Robin/Neumann boundary condition (5d), where $\alpha \geq 0$ is an interior penalty parameter and $\hat{\boldsymbol{\sigma}} = -\hat{p}\mathbf{I} + 2\mu_f\mathbf{D}(\hat{\mathbf{u}})$ is the total stress. The consistency relation

$$\mathbf{u} = \hat{\mathbf{u}} \quad \text{on } \bar{\Omega}_k(t) \tag{6}$$

is satisfied by exact solutions to (1) and (5). Hence, it is appropriate to constrain a numerical approximation to the background velocity $\mathbf{u}$ using a strong or weak form of the Dirichlet condition (6).

4. Finite element discretization

We discretize the velocity and pressure fields in space using the inf-sup stable $\mathbb{Q}_2$-$\mathbb{P}_1^{\text{disc}}$ finite element pair. The atmospheric subproblems (5) are solved on body-fitted meshes using an Arbitrary-Lagrangian-Eulerian (ALE) formulation in the reference frame moving with the constant mesh velocity $\mathbf{w}_k(\mathbf{x},t) = \mathbf{U}_k(t)$. Details of such moving mesh finite element (FE) methods can be found elsewhere [1, 13, 29] and are not discussed here. The ALE-FE approximation to $(\hat{\mathbf{u}}, \hat{p})$ is denoted by $(\hat{\mathbf{u}}_h, \hat{p}_h)$.

The fictitious domain problem (1) is solved using a fixed background mesh $\mathcal{T}_h$ that consists of quadrilaterals (in 2D) or hexahedra (in 3D). Instead of enforcing the Dirichlet-type velocity constraints (1c) and (6) strongly at discrete locations (as in [8, 17, 26]), we incorporate them into a weak form of (1) using (a discrete counterpart of) the distributed interior penalty term

$$s(\hat{\mathbf{u}}, \mathbf{U}; \mathbf{u}, \mathbf{v}) = \gamma_{\max}\sum_{k=1}^{N_p}\left[\int_{\hat{\Omega}_k(t)}\beta_k(\mathbf{u}-\hat{\mathbf{u}})\cdot\mathbf{v}\mathrm{d}\mathbf{x} + \int_{B_k(t)}(\mathbf{u}-\mathbf{U})\cdot\mathbf{v}\mathrm{d}\mathbf{x}\right] \tag{7}$$

that depends on a penalty parameter $\gamma_{\max} \gg 1$ and a damping function $\beta_k : \hat{\varOmega}_k \times \mathbb{R}_0^+ \to [0,1]$ such as[1]

$$\beta_k(\mathbf{x},t) = \min\left(1, \max\left(0, \frac{R_k + 0.75H_k - |\mathbf{x} - \mathbf{X}_k(t)|}{0.25H_k}\right)\right).$$

The velocity $\mathbf{U}$ of rigid body motion is defined by (2). In the discrete version of (7), integration is performed over the regions $\varOmega_{k,h}$ and $B_{k,h}$ that are covered by / enclosed by the ALE submesh.

Let $\mathbf{V}_h$ and Q_h denote the finite element spaces for the velocity and pressure approximations on $\bar{\varOmega}$, respectively. We seek $\mathbf{u}_h \in \mathbf{V}_h$ and $p_h \in Q_h$ such that

$$\int_{\varOmega} \rho_f \left(\frac{\partial \mathbf{u}_h}{\partial t} + \mathbf{u}_h \cdot \nabla \mathbf{u}_h\right) \cdot \mathbf{v}_h \mathrm{d}\mathbf{x} + \frac{\mu_f}{2} \int_{\varOmega} \mathbf{D}(\mathbf{u}_h) : \mathbf{D}(\mathbf{v}_h) \mathrm{d}\mathbf{x} + s_h(\hat{\mathbf{u}}_h, \mathbf{U}_h; \mathbf{u}_h, \mathbf{v}_h)$$

$$- \int_{\varOmega} p_h \nabla \cdot \mathbf{v}_h \mathrm{d}\mathbf{x} = 0, \qquad \mathbf{v}_h \in \mathbf{V}_h, \tag{8a}$$

$$\int_{\varOmega} q_h \nabla \cdot \mathbf{u}_h \mathrm{d}\mathbf{x} = 0, \qquad \mathbf{q}_h \in Q_h. \tag{8b}$$

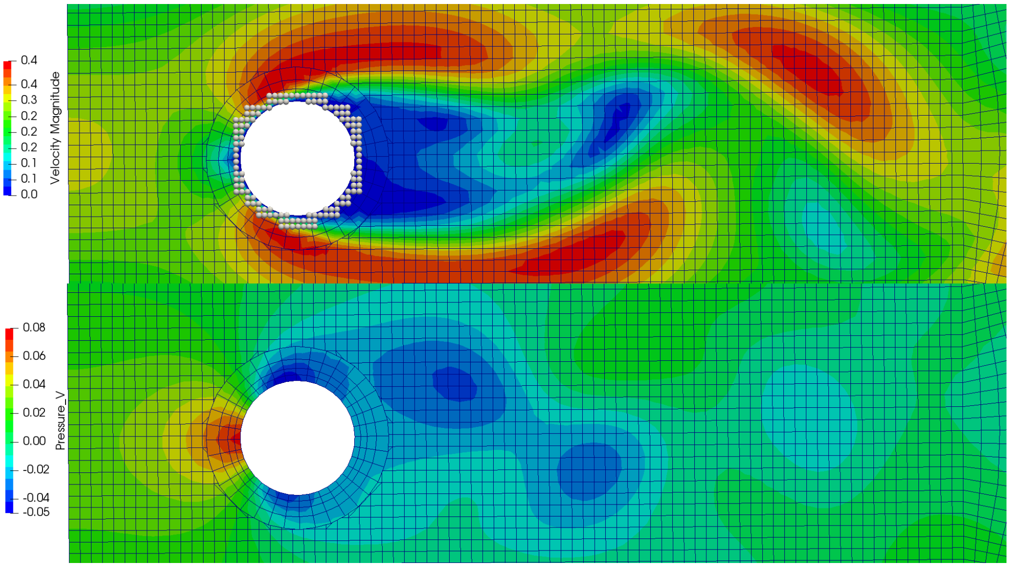


Figure 2: Snapshot of a Chimera multimesh computation for the flow around a cylinder configuration showing the background and particle submesh. Upper subfigure displays the velocity and the lower the respective pressure distribution.

Figure 2 shows a snapshot of the region in which the ALE mesh attached to $B_k(t)$ intersects the fixed background mesh. The former provides the Dirichlet data $\hat{\mathbf{u}}_h$ for calculating $s_h(\hat{\mathbf{u}}_h, \mathbf{U}_h; \mathbf{u}_h, \mathbf{v}_h)$

[1] The nonnegative function β_k determines the strength of velocity penalization. It should vanish in a neighborhood of the interface $\varGamma_k(t) = \partial\varOmega_k(t) \backslash \partial B_k(t)$ to avoid interference with the Robin boundary condition (5d).

defined by (7). The latter provides the Robin boundary data $\boldsymbol{\sigma}_h\mathbf{n} - \alpha(\mathbf{u}_h \cdot \mathbf{n})\mathbf{u}_h$ for updating $\hat{\mathbf{u}}_h$. The hydrodynamic force $\mathbf{F}_k$ and torque $\mathbf{T}_k$ are calculated using $\hat{\boldsymbol{\sigma}}_h = -\hat{p}_h\mathbf{I} + 2\mu_f\mathbf{D}(\hat{\mathbf{u}}_h)$ to approximate $\boldsymbol{\sigma}$ in (4). The simplicity and accuracy of integrating functions of the traction $\hat{\boldsymbol{\sigma}}_h\mathbf{n}$ over ∂B_k are a key advantage of the Chimera approach compared to one-mesh fictitious domain methods [17, 24, 26].

Remark 1. A remarkable property of the weak form (8) is that the interior penalty term (7) is well defined even for overlapping 'atmospheres' $\hat{\Omega}_k(t)$ and $\hat{\Omega}_j(t)$ of non-overlapping particles $B_k(t)$ and $B_j(t)$. The use of strongly imposed Dirichlet coupling conditions at background mesh nodes belonging to the overlap region $\hat{\Omega}_k(t) \cap \hat{\Omega}_j(t)$ would require artificial averaging of submesh data. The proposed approach performs such averaging automatically by using penalty parameters depending on β_k and β_j. An extension to the case $\hat{\Omega}_k(t) \cap B_j(t) \neq \emptyset$ is feasible but would require adding a Dirichlet penalty term to the submesh problem for $\hat{\Omega}_k(t)$. Such extensions can be carried out following [2, 10].

5. Fractional-step method

Let $t^n = n\Delta t$, where Δt is a constant time step and $n \in \mathbb{N}_0$. Approximate solutions at the time level t^n will be referred to using the superscript n. At the beginning of each time step, we use the old submesh approximation $(\hat{\mathbf{u}}_h^n, \hat{p}_h^n)$ to calculate the surface integrals (4) and the old background mesh approximation $(\mathbf{u}_h^n, p_h^n)$ to calculate the data of the Robin boundary condition (5d). Then we update the positions of the particles and solve the discrete saddle point problems associated with ALE submeshes for $\hat{\Omega}_k(t)$. These problems are small and can be solved efficiently, e.g., using the local Multilevel Pressure Schur Complement (MPSC) method [22, 20, 21]. The updated approximation $(\hat{\mathbf{u}}_h^{n+1}, \hat{p}_h^{n+1})$ can then be substituted into the interior penalty term of problem (8) for $(\mathbf{u}_h^{n+1}, p_h^{n+1})$.

Using the two-level θ-scheme to discretize (8a) in time, we obtain a nonlinear system of the form

$$\begin{bmatrix} A(u^{n+1}) + D^{n+1} & B \\ B^T & 0 \end{bmatrix} \begin{bmatrix} u^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} f^n + g^{n+1} \\ 0 \end{bmatrix}. \tag{9}$$

The contribution of the interior penalty term (7) is represented by $D^{n+1}u^{n+1} - g^{n+1}$, where D^{n+1} is a symmetric positive semi-definite matrix. In Appendix A below, we explicitly define all matrices and vectors that appear in (9), as well as the lumped approximation M_L to the consistent mass matrix M_C of the finite element space $\mathbf{V}_h$. We use M_L below for preconditioning purposes.

We solve the discrete problem (9) approximately using the following fractional-step algorithm:

1. Solve the viscous Burgers system

$$[A(\tilde{u}^{n+1}) + D^{n+1}]\tilde{u}^{n+1} = f^n + g^{n+1} - Bp^n. \tag{10}$$

2. Solve the pressure Poisson system

$$B^\top M_L^{-1} B(p^{n+1} - p^n) = \frac{1}{\Delta t} B^\top \tilde{u}^{n+1}. \tag{11}$$

3. Correct the intermediate velocity

$$[M_L + \Delta t D^{n+1}]u^{n+1} = [M_L + \Delta t D^{n+1}]\tilde{u}^{n+1} - \Delta t B(p^{n+1} - p^n). \tag{12}$$

Note that we penalize the velocity not only in the Burgers step but also in the final update. The nonlinear system (10) is linearized about u^n or solved using a fixed-point iteration method [20].

Remark 2. Rearranging (12) and invoking (11), we find that the corrected velocity u^{n+1} satisfies

$$B^\top u^{n+1} = \Delta t B^\top M_L^{-1} D^{n+1}(\tilde{u}^{n+1} - u^{n+1}).$$

Hence, u^{n+1} is approximately divergence-free. Moreover, the residual of the constraint $B^\top u^{n+1} = 0$ can be made as small as desired by performing additional outer iterations (cf. [20, 21]), in which the right-hand side of (10) is recalculated using the latest approximation to p^{n+1} instead of p^n.

6. Summary of the algorithm

for the sake of deeper numerical investigations we have implemented the Chimera multimesh method in two realizations differing in the mechanism of back-coupling of the submesh on the background mesh. In this regards we have generalized the multimesh method of Codina [8] to an instationary adaptive mesh-patch realization (further on we do refer to this realization as "Codina method") and its smoother alternative related to the previously described penalty method (further on we do refer to this realization as "Penalty method"). The use of the weak Dirichlet-Robin coupling is for both of these realizations identical, meaning that the coupling from the background mesh onto the submesh is in both cases identical. Now, let us declare the most relevant algorithmic steps in both of these realizations starting with the Codina method:

1. Solve (5) for the background mesh problem according to the classical Fictitious Boundary Method (in the first iteration the FBM mesh patch stands only for the particle itself, the FBM mesh patch is extended in higher number of iterations).
2. Solve (5) for the submesh problem using the given background velocity $\mathbf{u}_h$ and pressure p_h in the surface integral associated with the Robin boundary condition (5d).
3. Calculate $\mathbf{F}_k$ and $\mathbb{I}_k$ for each particle using the submesh stress $\hat{\boldsymbol{\sigma}}_h$ in the surface integrals (4).
4. Solve the discrete versions of (3) and update the particle position, velocity, angular position, and angular velocity.
5. Update the exchange mesh patch and interpolate the velocity to the extended mesh patch for coupling with the background mesh and go to Step 1 with an extended Fictitious Boundary mesh patch or continue (only for iteration 2 or more).

As it is visible above, there we re-iterate between the submesh and background mesh at least once, so that the extended mesh patch is applied according to the Codina method.

The most relevant algorithmic steps of the Penalty method can be summarized, as follows:

1. Calculate $\mathbf{F}_k$ and $\mathbb{I}_k$ for each particle using the submesh stress $\hat{\boldsymbol{\sigma}}_h$ in the surface integrals (4).
2. Solve the discrete versions of (3) and update the particle position, velocity, angular position, and angular velocity.
3. Solve (5) for the submesh problem using the given background velocity $\mathbf{u}_h$ and pressure p_h in the surface integral associated with the Robin boundary condition (5d).
4. Solve (5) for the background mesh problem using the penalty terms being responsible for the backcoupling of the submesh results onto the background mesh.
5. Solve the discrete version of (8) using the updated submesh velocity $\hat{\mathbf{u}}_h$ in $s_h(\hat{\mathbf{u}}_h, \mathbf{U}; \mathbf{u}_h, \mathbf{v}_h)$.

Compared to the Codina realization in this case the computational overhead is lower by one Pressure-Poisson system per time step. A momentum-like equation (8) is required to be solved in this realization which has a similar computational cost as the solution of the second momentum equation in the Codina realization.

7. Numerical examples

In this section, we perform numerical studies of the Chimera domain decomposition method with strong and weak imposition of the Dirichlet coupling condition on the background mesh. The proposed approach (Chimera-W) incorporates the interior penalty term (7) into the discretized momentum equation (8a). The alternative that we call Chimera-S is a fictitious boundary method that enforces pointwise Dirichlet constraints at *hole nodes* in $\hat{B}_k(t)$ and *fringe nodes* in $\hat{\Omega}_k$. The terminology that we use here is adopted from the paper by Houzeaux and Codina [8]. A node $\mathbf{x}_i$ of a mesh cell crossed by $\partial B_k(t)$ is treated as a hole node if $\mathbf{x}_i \in B_k(t)$ and as a fringe node if $\mathbf{x}_i \in \hat{\Omega}_k(t)$. We prescribe the fictitious boundary conditions $\mathbf{u}_h(\mathbf{x}_i) = \mathbf{U}(\mathbf{x}_i)$ and $\mathbf{u}_h(\mathbf{x}_i) = \hat{\mathbf{u}}_h(\mathbf{x}_i)$ at hole and fringe nodes, respectively. A detailed description of the original Chimera-S coupling can be found in [8].

7.1 DFG Benchmark 2D-2

The DFG Benchmark 2D-2 [18] is a well-established test case in computational fluid dynamics (CFD) designed to evaluate numerical algorithms for solving the incompressible Navier-Stokes equations under laminar flow conditions. This benchmark focuses on the two-dimensional flow around a circular cylinder at a Reynolds number (Re) of 100, where the flow exhibits periodic vortex shedding, known as the von Kármán vortex street. For an in depth description of the benchmark we refer to the publication [18]. We will summarize the most important parameters of the benchmark here briefly.The computational domain consists of a two-dimensional channel with a circular cylinder obstacle. The cylinder has a diameter of 0.1 units and is positioned at (0.2, 0.2) within a channel of length 2.2 units and height 0.41 units. At the inlet a parabolic velocity profile is applied at the inlet to simulate fully developed laminar flow. We measure the drag coefficient C_D and lift coefficient C_F which vary over time due to the periodic vortex shedding characteristic of this flow regime. The velocity profile at the inlet is defined as:

$$\mathbf{u}(0, y) = \left(\frac{4Uy(0.41 - y)}{0.41^2}, 0 \right)$$

This represents a parabolic velocity distribution across the channel height, characteristic of laminar flow conditions. The maximum velocity, U, occurs at the centerline ($y = 0.205$) in a channel with a height of 0.41. We compute forces with the help of the stress tensor:

$$\boldsymbol{\sigma} = \nu \left(\nabla \mathbf{u} + (\nabla \mathbf{u})^T \right) - p\mathbf{I}$$

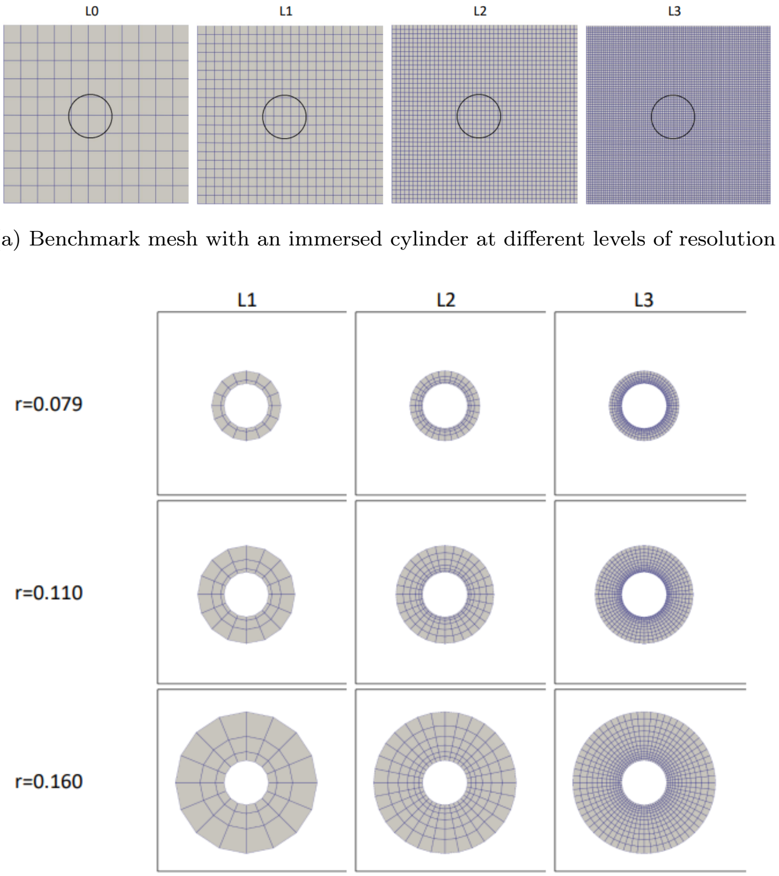
Here, ν denotes the kinematic viscosity, $\nabla \mathbf{u}$ is the velocity gradient, p represents the fluid pressure, and $\mathbf{I}$ is the identity matrix. The drag F_d and lift F_l forces are calculated as:

$$\begin{pmatrix} F_d \\ F_l \end{pmatrix} = \int_S \boldsymbol{\sigma} \cdot \boldsymbol{\eta} \, ds$$

In this equation, the surface integral is taken over the surface S of the cylinder, with $\boldsymbol{\eta}$ representing the outward-pointing normal vector on the surface. Using F_d and F_l we can compute the drag and lift coefficients C_d and C_l:

$$C_d = \frac{2F_d}{U_{\text{mean}}^2 L}, \qquad C_l = \frac{2F_l}{U_{\text{mean}}^2 L} \tag{13}$$

Where U_{mean} is the average velocity of the fluid flow, and L is the characteristic length.



(a) Benchmark mesh with an immersed cylinder at different levels of resolution

(b) Body-fitted submesh at different resolution levels.

Figure 3: Background mesh and submeshes for the DFG Benchmark 2D-2 configuration.

We simulated the DFG Benchmark 2D-2 using the standard fictitious boundary/domain approach and the Codina vadiant of the above described Chimera multimesh approach. The resulting drag and lift coefficients over time are presented in Figure 4. This approach typically leads to a sharp and often oscillatory force response.

To assess the advantages of the Chimera submesh method and provide insights on how to calibrate the mesh/submesh resolutions, we performed a second simulation where the cylinder is embedded in a body-fitted submesh. The results, shown in Figure 5 and 6, present both the drag and lift coefficients over time at various background/submesh resolutions and submesh radii. We have marked in the plots the reference C_d^{min}, C_d^{max}, C_d^{mean}, C_l^{min}, C_l^{max} and C_l^{mean} values and the evolution zone as determined in previous publications of our research group [18, 22]. The nomenclature for the cases is as follows L1_110L2 corresponds to background mesh resolution L1, submesh radius $r = 0.110$ and submesh resolution L2. From the simulation results we clearly see that all those three parameters have an influence on the accuracy of the computation. The Chimera method achieves smoother force evolution while preserving accuracy, demonstrating its potential for improved numerical stability and efficiency compared to the standard fictitious boundary approach.

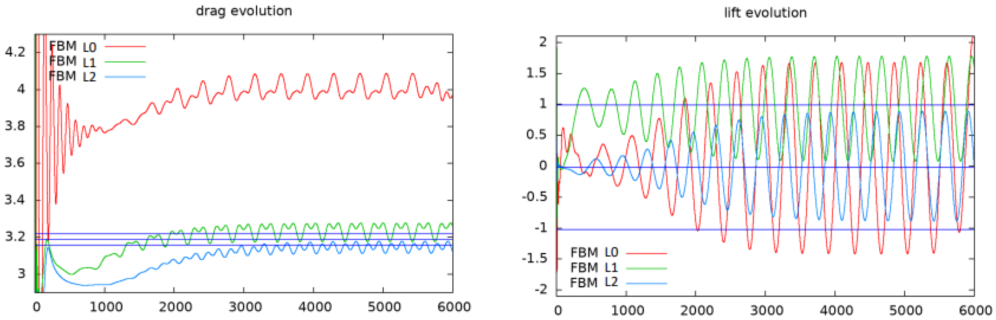


Figure 4: Drag and lift coefficients over time using the standard fictitious boundary approach at different levels of resolution.

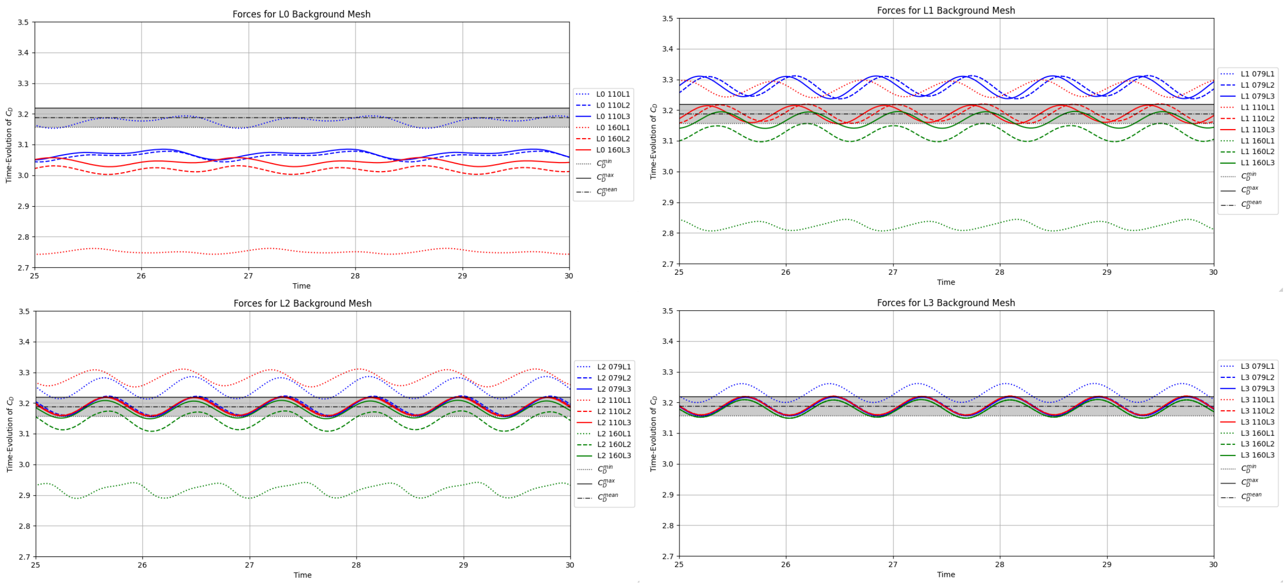


Figure 5: Drag coefficients over time using the Chimera submesh method with increasing submesh resolution.

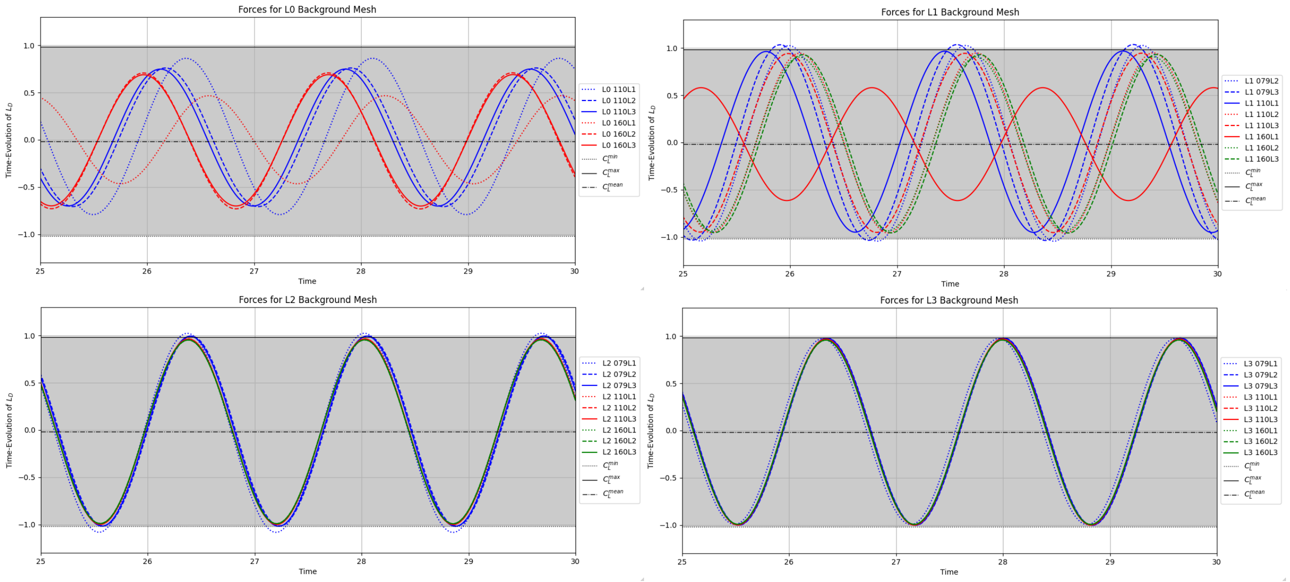


Figure 6: Lift coefficients over time using the Chimera submesh method with increasing submesh resolution.

7.2 Flow around a moving cylinder

This benchmark involves a cylinder oscillating horizontally in a rectangular channel. The primary quantities of interest are the time-dependent drag and lift forces and their corresponding coefficients. The cylinder motion is sinusoidal in the x-direction, while the y and z coordinates remain fixed.

Table 1: Cylinder and domain parameters

Parameter	Value
Cylinder diameter D	0.1
Initial position (X_0, Y_0, Z_0)	$(1.1,\ 0.2,\ 0.1025)$
Oscillation amplitude A	0.25
Oscillation frequency f	0.25
Domain size	$2.2 \times 0.41 \times 0.1025$

Table 2: Fluid properties

Property	Value
Density ρ	$1\,\mathrm{kg/m^3}$
Kinematic viscosity ν	$10^{-3}\,\mathrm{m^2/s}$
Initial velocity	Fluid at rest

The cylinder's x-position evolves as

$$X_c(t) = X_0 + A \cdot \sin(2\pi f t), \quad Y_c(t) = Y_0, \quad Z_c(t) = Z_0.$$

Due to the off-center y-placement, a non-zero lift force is generated. As a reference for this benchmark we use the 2D simulations conducted by Wan, Turek and Rivkind [28], where the drag and lift were calculated on a body-aligned mesh meaning that the circle in their case was built directly into the mesh. In order to make the results comparable with 3D simulations the drag and lift coefficients (see eq. 13) need to be scaled by $\frac{1}{T}$ where T is the thickness of the channel. For this particular numerical validation case we have performed mesh resolution convergence studies for both of the considered Chimera submesh implementations, namely the Codina and also for the Penalty method. The respective graphical representations are organized into plots displaying the evolution of drag and lift coefficients. These are the Figures 7 and 8 in case of the Codina submesh method and Figures 9 and 10 in case of the Penalty submesh method. In both cases we compare the Chimera submesh results to the body-fitted reference solution of Wan [28]. As clearly shown in the results, the Penalty method provides a smoother prediction of the evolution of the monitored force components. This behavior is indicative of its enhanced ability to dampen oscillations caused by the changing integration point patterns during the instationary simulation. In contrast, the Codina method reacts more stepwise, responding primarily to significant geometric changes, such as when the exchange patch of elements between submes and background mesh shifts. These larger changes introduce more abrupt variations, resulting in sharper fluctuations in the force components, which are less effectively damped than with the Penalty method.

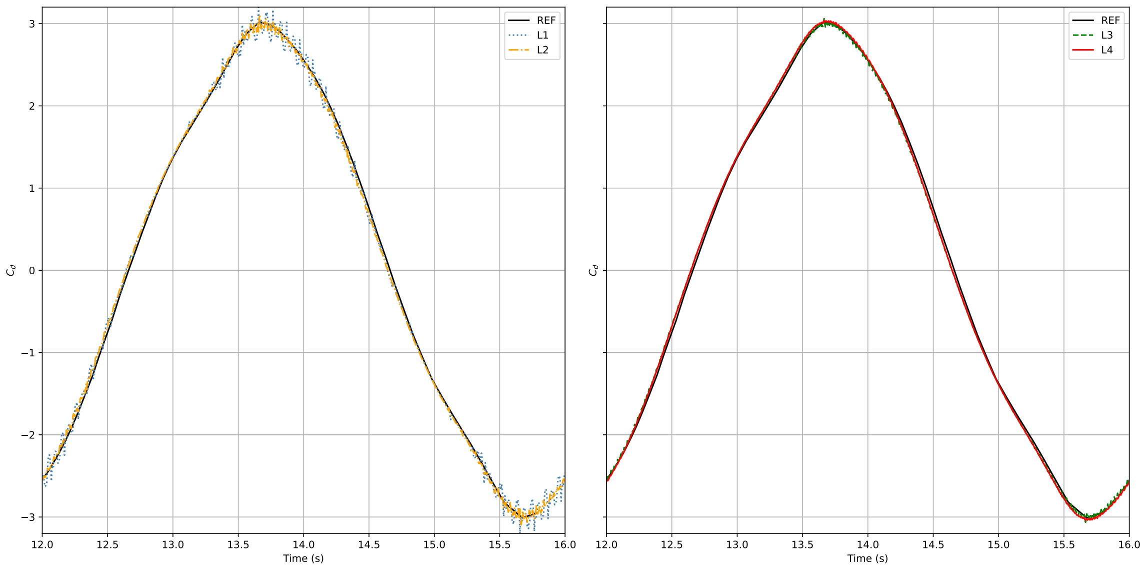


Figure 7: Time-dependent drag coefficients for the Chimera submesh method vs reference [28].

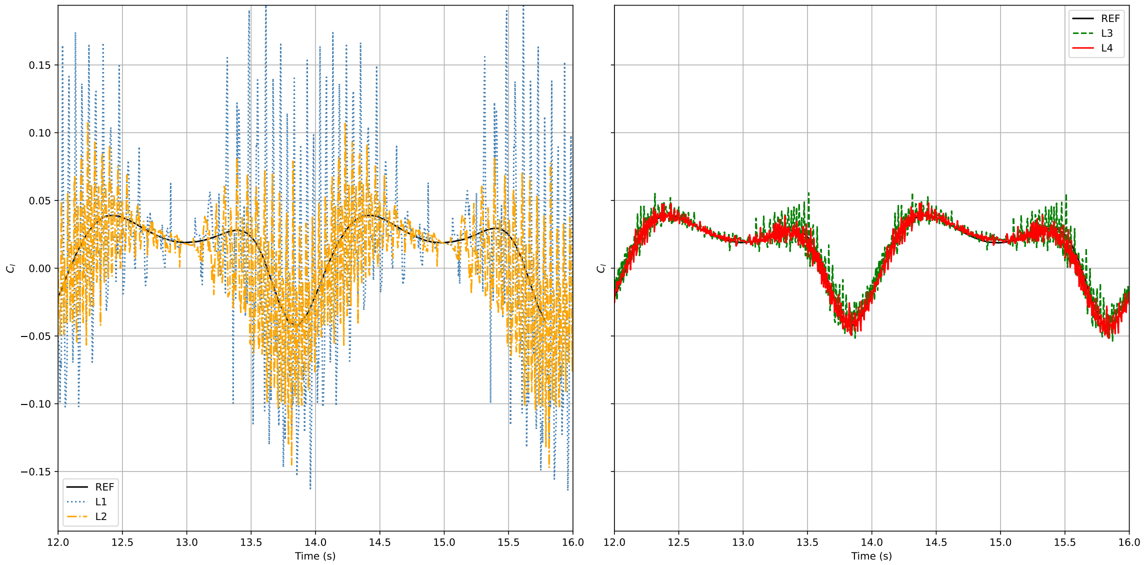


Figure 8: Time-dependent lift coefficients for the Chimera submesh method vs reference [28].

The first evident key fact that we can conclude is that our implementation of the Chimera submesh method shows systematic mesh convergence. On Levels L3 and L4 the results meet the reference within ±1 % for drag and ±5 % for lift over an entire oscillation cycle. On coarser Chimera meshes can predict the mean drag correctly (because the leading-order contribution is large), but lift requires at least an order of magnitude finer resolution. This is because the lift is two orders of magnitude smaller than the drag, so a percentage level error for C_d would show up a hundred-percent error for C_l.

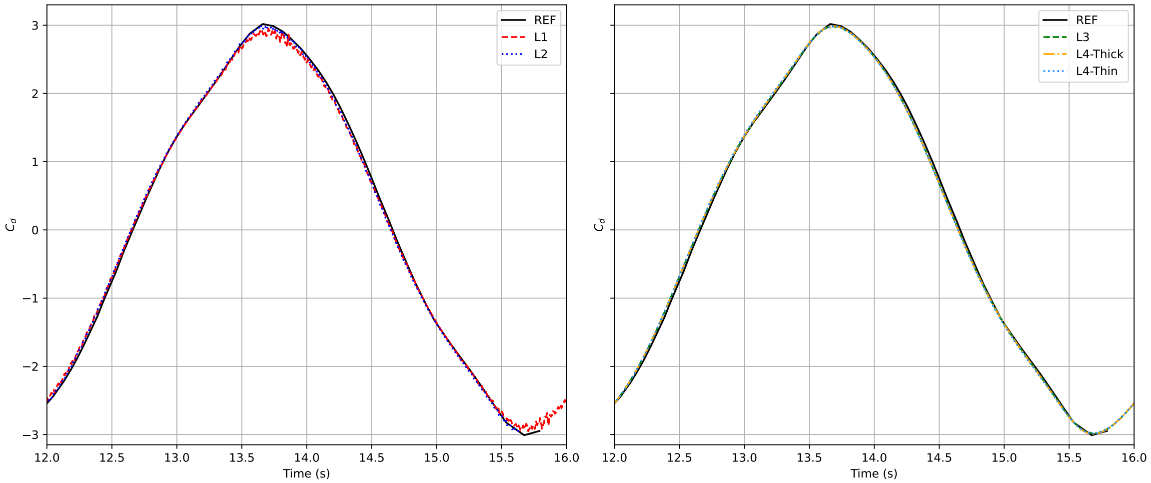


Figure 9: Time-dependent drag coefficients for the Chimera submesh penalty method vs reference [28].

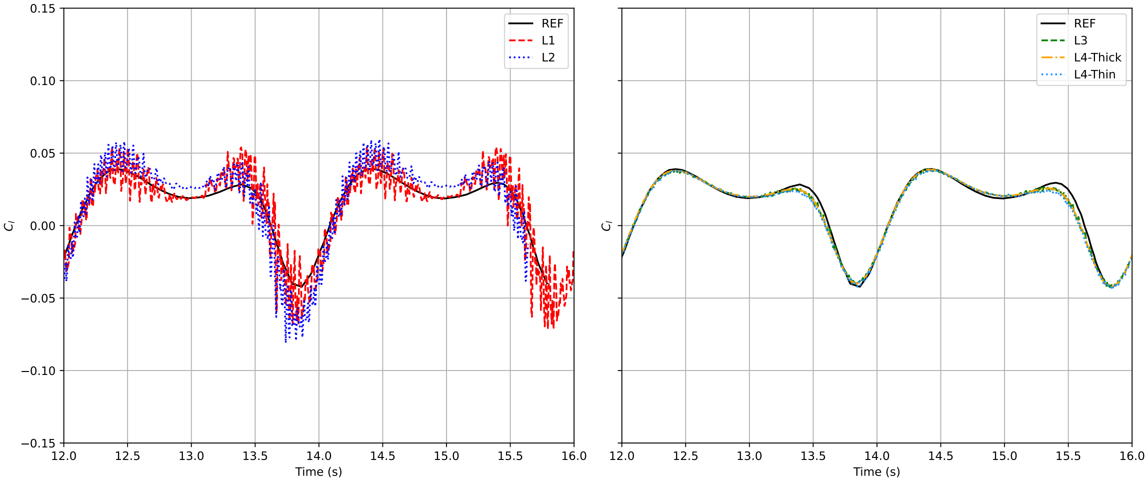


Figure 10: Time-dependent lift coefficients for the Chimera submesh penalty method vs reference [28].

7.3 Motion of an Elliptical Particle Immersed in a Viscous Fluid: Jeffrey Orbits

Jeffery's orbits describe the periodic rotational motion of ellipsoidal particles suspended in a viscous fluid under shear flow conditions. The theory was developed for a creeping flow, so the Reynolds number is taken to be effectively zero. In a cross-shear flow, such particles do not maintain a fixed orientation but instead their major axis undergoes continuous rotation that traces closed periodic trajectories on a sphere. The specific path of these orbits depends on the particle's initial orientation and its aspect ratio. In his work Jeffrey [9] developed an analytical solution for the time-dependent orientation of the particle. For our validation purposes of the Chimera submesh method we reduced the configuration to two dimensions. As a consequence the particle's major axis is confined to the shear plane and the rotation is reduced to the azimuthal angle (θ) which in our case is normal to the shear plane. We compared the results to Jeffrey's analytical solution and to the results of our own FBM particle implementation in order to access if the Chimera submesh method can indeed offer improved quality of result on the same background mesh. For an elliptical particle with semi-axes a and b (with aspect ratio $r = a/b$), Jeffery's analytical expressions for the time-dependent angle and angular velocity are

$$\theta(t) = \tan^{-1}\left(\frac{b}{a}\tan\left(\frac{ab\,\dot{\gamma}t}{a^2+b^2}\right)\right)$$

$$\dot{\theta}(t) = -\frac{\dot{\gamma}}{a^2+b^2}\left(a^2\cos^2\theta + b^2\sin^2\theta\right),$$

where $\dot{\gamma}$ is the shear rate and θ is the azimuthal angle. We have depicted the basic setup of the benchmark case in figure 11. A sample of the flow field is shown in figure 12, the numerical sizes and mesh resolution in table 3.

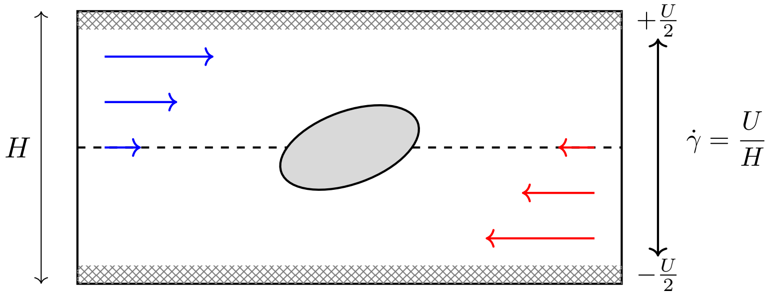


Figure 11: Schematic setup of the Jeffrey's Orbit case.

Table 3: Overview of Domain, Mesh Resolution, and Ellipse Sizes

(a) Domain Dimensions

Dir.	**Range**	Δ
X	-1 to 1	2
Y	-0.5 to 0.5	1

(b) Mesh Resolution

Level	**Elements**
0	3200
1	25600
2	204800

(c) Ellipse Sizes

Name	$\boldsymbol{a}$	$\boldsymbol{b}$
Big	0.05	0.025
Small	0.025	0.0125

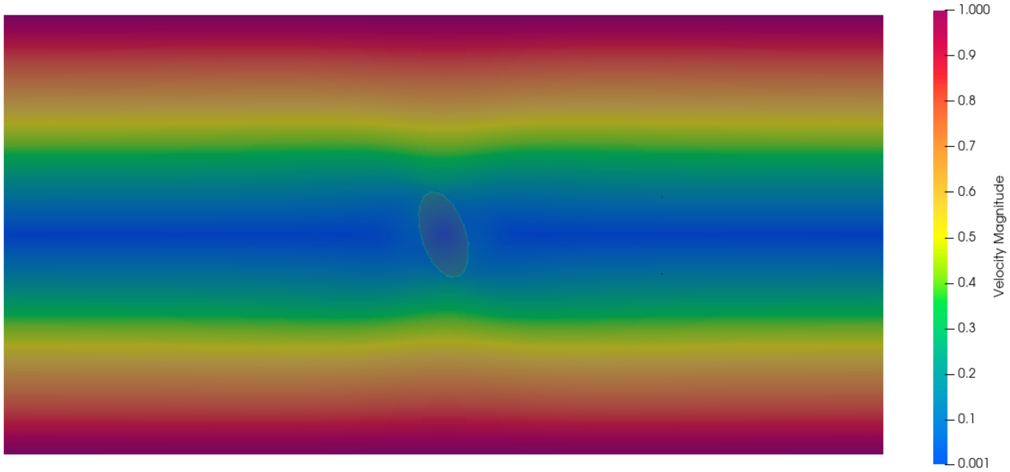


Figure 12: Sample of a flow field for the Jeffrey's Orbit simulation.

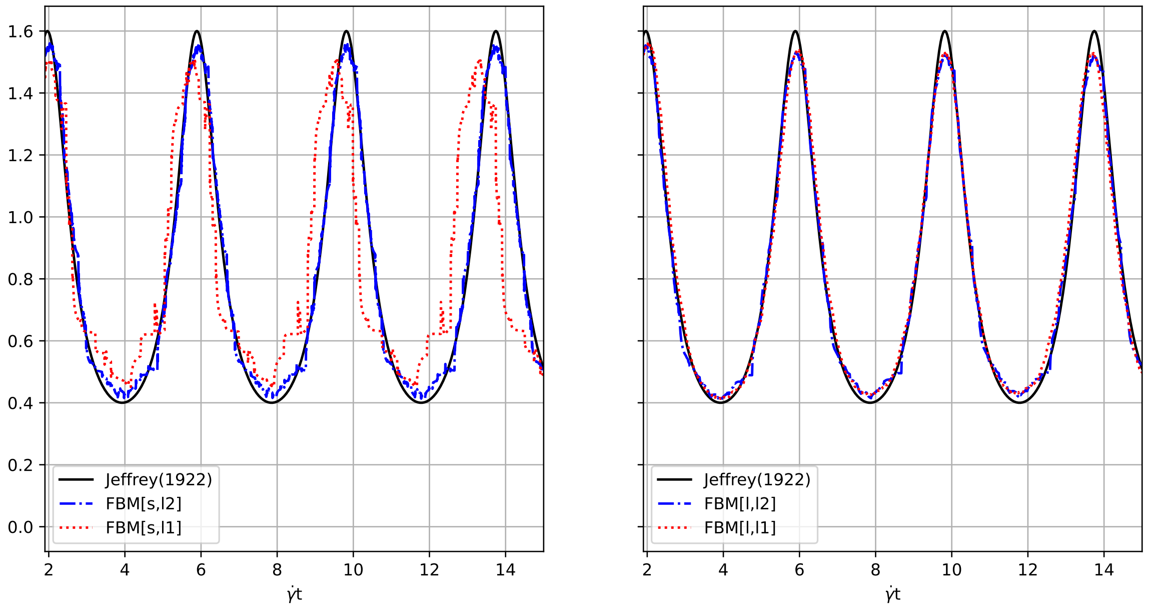


Figure 13: Fictious Boundary Method compared to Jeffrey's solution. Left: small particle, Right: big particle

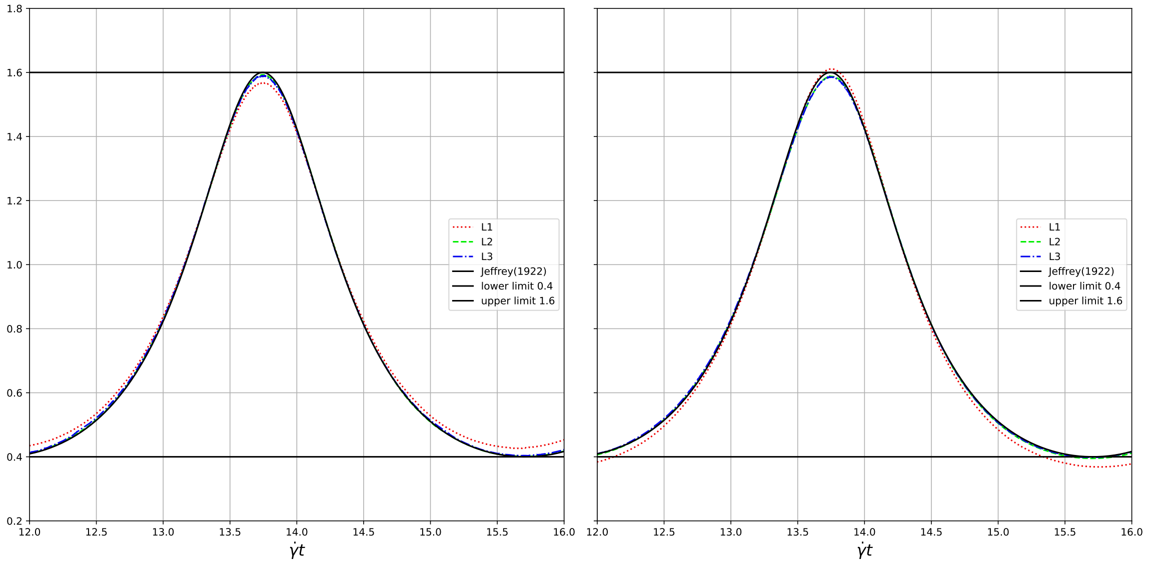


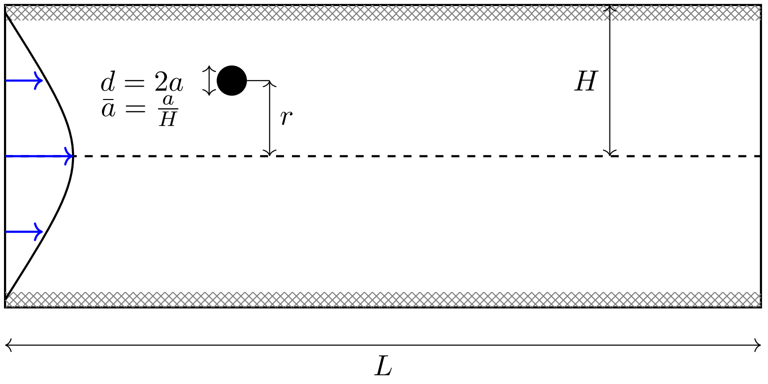
Figure 14: Chimera submesh method compared to Jeffrey's solution. Left: small particle, Right: big particle

In fig. 13 we show the results of the FBM in the Jeffrey's Orbit test case. The results were computed for two particle sizes (see tab. 3c) and for two different refinement levels each (see tab.

3b). As expected we notice characteristic features of the FBM: for the small particle and lower mesh resolution (FBM[s,l1]) there is a similarity to the reference solution, but there are also oscillations and severe jitter of the curve. The FBM solution overshoots the minimum of the reference and undershoots the maximum. A phase shift can also be observed. Then for all other FBM solutions FBM[s,l2], FBM[l,l1] and FBM[l,l2] we have added resolution by inscreasing the mesh refinement and/or the particle size. We see that with increased resolution the FBM solution tracks the curve better, offers reduced oscillation/jitter, eliminates the phase shift. Differences in the peak values are observable even at the higher refinement levels. In fig. 14 we test the Chimera submesh method for the Jeffrey's Orbit case. The shape of the curves in the Chimera case is much smoother this can be attributed to the fact that the method is essentially body-fitted and does not cut through mesh cells in the way the FBM does. The body-fitted Chimera submesh reproduces Jeffery's analytical orbit almost out of the box at a coarse resolution, whereas a fictitious-boundary approach needs at least one extra global refinement—and is still much more sensitive to particle size. This is clearly shown by the computations on the lower level of the FBM where the error for the peak value where we observe an $\approx 9\%$ error whereas the Chimera submesh method has an error of $\approx 0.5\%$. For the small particle the FBM has an $\approx 3\%$ error on a one level finer background mesh. This pattern is observable also for the larger particle where the FBM has a $\approx 3\%$ error w.r.t. the peak value on level 1 and a sub-percent error on level 2. The Chimera submesh method already has a sub-percent error on the first level.

7.4 Segre-Silberberg Study

The Segre-Silberberg effect describes the lateral migration of particles in shear flow, particularly in Poiseuille flow, where fluid moves through a tube or between parallel plates. Particles in such flows do not remain near the centerline or the walls but instead migrate to a stable equilibrium position, typically around 0.2 to 0.6 times the channel radius or half-height from the center. This migration occurs due to a balance between two opposing lift forces: the shear-gradient lift force, which pushes particles away from the center due to velocity variations, and the wall-induced lift force, which pushes them away from the boundaries due to interactions with slower-moving fluid. The effect significantly depends on the Reynolds number, where migration speed increases, and equilibrium positions change with different Reynolds numbers, particle sizes and wall-dependent parameters. The effect has been studied in depth by Yang et. al. [30] for a particle in a tubular channel and a circular particle between parallel planes. For varying configurations of the Chimera method we compute the equilibrium position of the particle for a Poiseuille flow between two parallel plates. For this case in the original publication it was found that the equilibrium position of the particle shifts closer to the centerline of the channel. This occurs because the shear-gradient lift force dominates, pushing the particle away from high shear regions near the walls.



For two different particle sizes $\bar{a} = \frac{a}{H}$, where a is the radius of the particle, and H is the half-height of the channel, we computed the equilibrium positions $\bar{r}_e$ which is stated in terms of a relative height $\bar{r} = \frac{r}{H}$ for a range of Reynolds numbers. To analyze the influence of the domain length, we also varied the channel length. We compared our results to those of the group Yang/Glowinski [30]. The results of these computations are summarized for both of the implemented submesh methods, namely for the Codina method (see Tables 4, 5) and also the for the Penalty method (see Tables 6, and 7). Due to the stronger convergence of the Codina method there are computations performed only for 3 subsequent resolution levels. For the Penalty method the numerical studies are extended also to higher resolution levels (even if the submeshes were not finer refined as L3) and also for different size of the submeshes. As visible from the results the two methods are converging to the same results what guarantees a validation of the method. Comparison of the results with available previously published references is provided in Table 9a and 9b which reference computational results of Glowinski [30]. In order to eliminate the dependence of the equilibrium position on the considered channel length we have also performed some low resolution level computations in order to determine the minimal channel length being necessary for the computations. The channel lengths have been varied in the range of $L : H = [20 : 1; 10 : 1; 5 : 1]$. The results of these channel length studies are available in Table 8, which confirm a minimal channel length on the order of $L : H = [10 : 1]$. The naming convention of the all the here applied background and submeshes is following the same conventions previously introduced in the numerical result section devoted to the moving-cylinder (see Section 7.1, particularly Figure 3).

Re	Background mesh Resolution	Submesh Resolution	Equilibrium position	%err
12	L1	160L1	0.40404	0.42
	L2	130L2	0.40654	0.19
	L3	110L3	**0.405**76	
80	L1	160L1	0.21944	1.48
	L2	130L2	0.21554	0.32
	L3	110L3	**0.216**24	
180	L2	130L2	0.16669	0.65
	L3	110L3	0.16704	0.86
	L4	060L3	**0.165**62	

Table 4: Equilibrium particle position data for varying Reynolds numbers and mesh resolutions for particle size $2\bar{a} = 0.10$ using the Codina submesh method.

Re	Background mesh Resolution	Submesh Resolution	Equilibrium position	%err
18	L1	130L1	0.44584	0.21
	L2	110L2	0.44676	0.00
	L3	079L3	**0.446**78	
45	L1	130L1	0.34320	0.69
	L2	110L2	0.34490	0.20
	L3	079L3	**0.345**60	
180	L1	130L1	0.22008	0.07
	L2	110L2	0.22002	0.10
	L3	079L3	**0.220**24	

Table 5: Equilibrium particle position data for varying Reynolds numbers and mesh resolutions for particle size $2\bar{a} = 0.15$ using the Codina submesh method.

Re	Background mesh Resolution	Submesh Resolution	Equilibrium position	%err	%err (vs Codina)
12	L1	160/L1	0.40112	0.58	1.14
	L2	160/L2	0.39952	0.97	1.54
	L3	160/L3	0.40298	0.11	0.69
	L4	160/L3	0.40122	0.55	1.12
	L4	110/L3	0.40344		0.57
80	L1	160/L1	0.21700	0.73	0.35
	L2	160/L2	0.21334	0.97	1.34
	L3	160/L3	0.21286	1.19	1.56
	L4	160/L3	0.21274	1.24	1.62
	L4	110/L3	0.21542		0.38
180	L1	160/L1	0.12324	26.19	25.59
	L2	160/L2	0.16533	0.98	0.17
	L3	160/L3	0.16619	0.47	0.34
	L4	160/L3	0.16566	0.79	0.02
	L4	110/L3	0.16697		0.82

Table 6: Equilibrium particle position data for varying Reynolds numbers and mesh resolutions for particle size $2\bar{a} = 0.10$ using the Penalty submesh method.

Re	Background mesh Resolution	Submesh Resolution	Equilibrium position	%err	%err (vs Codina)
18	L1	160/L1	0.44252	0.88	0.95
	L2	160/L2	0.44206	0.98	1.06
	L3	160/L3	0.44300	0.77	0.85
	L4	160/L3	0.44318	0.73	0.81
	L4	110/L3	0.44644		0.08
45	L1	160/L1	0.33550	2.36	2.92
	L2	160/L2	0.33848	1.49	2.06
	L3	160/L3	0.33920	1.28	1.85
	L4	160/L3	0.33904	1.33	1.90
	L4	110/L3	0.34360		0.58
180	L1	160/L1	0.21756	0.44	1.22
	L2	160/L2	0.21600	1.15	1.93
	L3	160/L3	0.21544	1.41	2.18
	L4	160/L3	0.21536	1.45	2.22
	L4	110/L3	0.21852		0.78

Table 7: Equilibrium particle position data for varying Reynolds numbers and mesh resolutions for particle size $2\bar{a} = 0.15$ using the Penalty submesh method.

$2\bar{a}$	Re	**SHORT**	**MEDIUM**	**LONG**	**M. vs. S [%]**	**M. vs. L [%]**
0.10	18	0.3330	0.3427	0.3426	-2.9	-0.1
	80	0.2175	0.2194	0.2189	-0.9	-0.2
	180	-0.1061	0.1667	0.1579	-163.6	-5.3
0.15	18	0.4437	0.4458	0.4458	-0.5	0.0
	45	0.3349	0.3432	0.3433	-2.4	0.0
	180	0.1426	0.2201	0.2184	-35.2	-0.8

Table 8: Domain length dependence study (SHORT, MEDIUM, LONG) with the Codina submesh method. Equilibrium position data for varying Reynolds numbers on L1 background mesh resolutions.

To gain further insights we not only compared the Chimera submesh method to our own FBM for the Segre-Silberberg configuration. We not only compare the value of the equilibrium position, we also look at oscillations in the solution to show important aspects of the Chimera submesh method. As we can see in Fig. 15 the Chimera solution (particularly, the results of the Codina method are visualized) has significantly less oscillations in five of the six comparision cases. The FBM converges to a value that is close to the reference solution, but shows a periodic oscillation of $O(10^{-3})$ to $O(10^{-2})$ around its mean value. The amplitude of the oscillation seems to be correlated to the Reynolds number. This is an intrinsic characteristic of the FBM because there is no guarantee that at the equilibrium position there is a perfectly symmetric alignment of nodal constraints and consequently some minor force imbalances remain even at $\bar{r}_e^{FBM}$.

It might be also important to point out that in this particular numerical example is the submesh method of Codina more superior to the penalty method, which might be devoted to the fact that in equilibrium position the exchange mesh patch (between submesh and background mesh) is not subjected to any further changes.

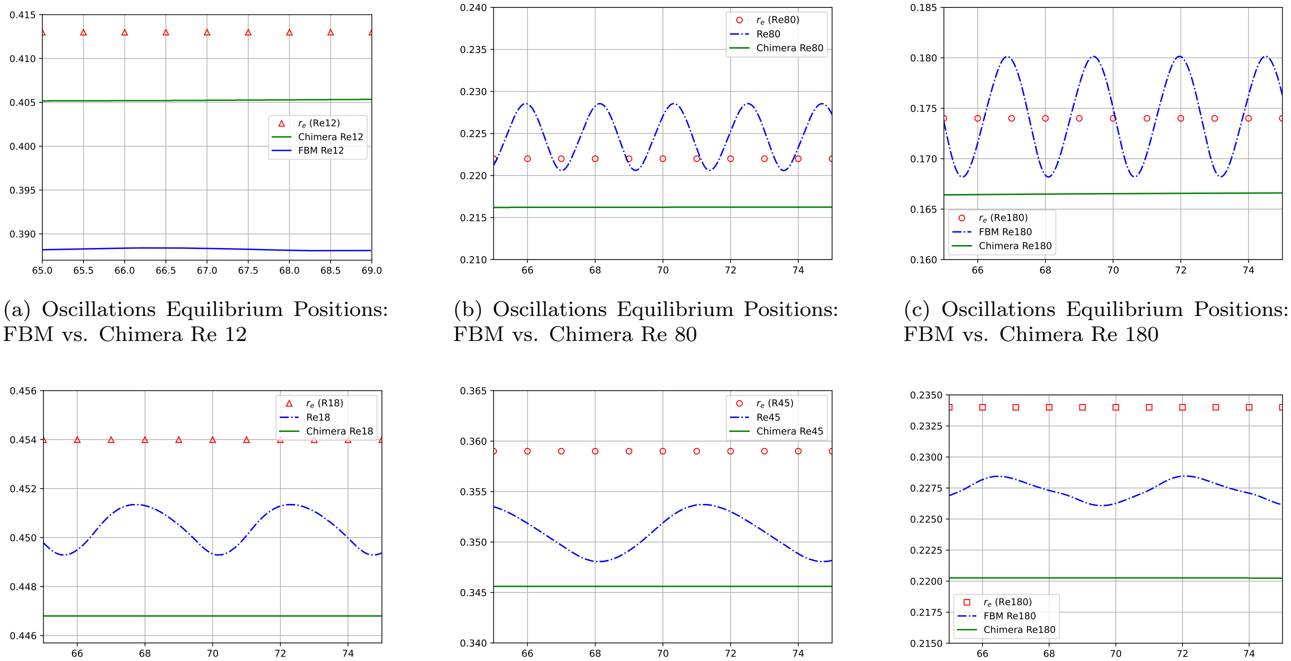
2a=0.10				
Re	ref [30]	pure FBM	Penalty	Codina
12	0.413	0.388	0.403	**0.405**8
80	0.222	0.225	0.215	**0.216**2
180	0.174	0.174	0.167	**0.165**6

(a) Lift-Off $2\bar{a} = 0.10$ equilibrium positions.

2a=0.15				
Re	ref [30]	pure FBM	Penalty	Codina
18	0.454	0.450	0.446	**0.446**8
45	0.359	0.351	0.344	**0.345**6
180	0.234	0.227	0.219	**0.220**2

(b) Lift-Off $2\bar{a} = 0.15$ equilibrium positions.

Table 9: Converged equilibrium positions in comparison with Glowinski [30].



(a) Oscillations Equilibrium Positions: FBM vs. Chimera Re 12

(b) Oscillations Equilibrium Positions: FBM vs. Chimera Re 80

(c) Oscillations Equilibrium Positions: FBM vs. Chimera Re 180

(d) Oscillations Equilibrium Positions: FBM vs. Chimera Re 18

(e) Oscillations Equilibrium Positions: FBM vs. Chimera Re 45

(f) Oscillations Equilibrium Positions: FBM vs. Chimera Re 180

Figure 15: Equilibrium Position Stability and Oscillations in Segre–Silberberg Migration

8. Conclusions

In this work, we explored the possibility of using a new weak form of Dirichlet-Robin coupling in the context of FE simulation tools for particulate flows. The benefits of the presented Chimera multimesh method include simplicity, efficiency, and continuous dependence of numerical solutions on the location of moving particles. The use of body-fitted submeshes eliminates the need for mesh deformation techniques designed to enhance the accuracy of fictitious boundary / subspace projection algorithms with strongly imposed Dirichlet constraints [1, 14, 29]. Extension to overlapping submeshes are feasible and can be performed by adapting the general framework developed in [2, 10] for projection schemes using discontinuous Galerkin weak forms and a different kind of interior penalization.

Acknowledgments

This work was supported by the German Research Foundation (DFG) under grant KU 1530/28-1 (TU 102/77-1) with project number 446888252. The authors gratefully acknowledge collaboration on this project with Prof. Yuliya Gorb (National Science Foundation) and Prof. Alexey Novikov (Pennsylvania State University).

References

[1] Steffen Basting and Rodolphe Prignitz. An interface-fitted subspace projection method for finite element simulations of particulate flows. *Computer Methods in Applied Mechanics and Engineering*, 267:133–149, Dec 2013.

[2] Jørgen S. Dokken, August Johansson, André Massing, and Simon W. Funke. A multimesh finite element method for the Navier–Stokes equations based on projection methods. *Computer Methods in Applied Mechanics and Engineering*, 368:113129, Aug 2020.

[3] R. Glowinski, T.-W. Pan, T.I. Hesla, and D.D. Joseph. A distributed Lagrange multiplier/fictitious domain method for particulate flows. *International Journal of Multiphase Flow*, 25(5):755–794, Aug 1999.

[4] R. Glowinski, T.W. Pan, T.I. Hesla, D.D. Joseph, and J. Périaux. A fictitious domain approach to the direct numerical simulation of incompressible viscous flow past moving rigid bodies: Application to particulate flow. *Journal of Computational Physics*, 169(2):363–426, May 2001.

[5] S. Haeri and J.S. Shrimpton. On the application of immersed boundary, fictitious domain and body-conformal mesh methods to many particle multiphase flows. *International Journal of Multiphase Flow*, 40:38–55, Apr 2012.

[6] William D. Henshaw. A fourth-order accurate method for the incompressible Navier-Stokes equations on overlapping grids. *Journal of Computational Physics*, 113(1):13–25, Jul 1994.

[7] William D. Henshaw and N. Anders Petersson. A split-step scheme for the incompressible Navier-Stokes equations. *Numerical Simulations of Incompressible Flows*, page 108–125, Jan 2003.

[8] Guillaume Houzeaux and Ramon Codina. A Chimera method based on a Dirichlet/Neumann(Robin) coupling for the Navier–Stokes equations. *Computer Methods in Applied Mechanics and Engineering*, 192(31–32):3343–3377, Aug 2003.

[9] George Barker Jeffery. The motion of ellipsoidal particles immersed in a viscous fluid. *Proceedings of the Royal Society of London. Series A, Containing Papers of a Mathematical and Physical Character*, 102(715):161–179, Nov 1922.

[10] August Johansson, Benjamin Kehlet, Mats G. Larson, and Anders Logg. Multimesh finite element methods: Solving PDEs on multiple intersecting meshes. *Computer Methods in Applied Mechanics and Engineering*, 343:672–689, Jan 2019.

[11] Jongho Lee, Jungwoo Kim, Haecheon Choi, and Kyung-Soo Yang. Sources of spurious force oscillations from an immersed boundary method for moving-body problems. *Journal of Computational Physics*, 230(7):2677–2695, Apr 2011.

[12] A. Main and G. Scovazzi. The shifted boundary method for embedded domain computations. Part II: Linear advection–diffusion and incompressible Navier–Stokes equations. *Journal of Computational Physics*, 372:996–1026, Nov 2018.

[13] B. Maury. Direct simulations of 2D fluid-particle flows in biperiodic domains. *Journal of Computational Physics*, 156(2):325–351, Dec 1999.

[14] R. Münster, O. Mierka, and S. Turek. Finite element-fictitious boundary methods (FEM-FBM) for 3D particulate flow. *International Journal for Numerical Methods in Fluids*, 69(2):294–313, May 2011.

[15] N.A. Patankar, P. Singh, D.D. Joseph, R. Glowinski, and T.-W. Pan. A new formulation of the distributed Lagrange multiplier/Fictitious domain method for particulate flows. *International Journal of Multiphase Flow*, 26(9):1509–1524, Sep 2000.

[16] Charles S. Peskin. The immersed boundary method. *Acta Numerica 2002*, page 479–518, Jul 2002.

[17] Rodolphe Prignitz and Bänsch Eberhard. Numerical simulation of suspension induced rheology. *Kybernetika*, 46(2):281–293, 2010.

[18] M. Schäfer, S. Turek, F. Durst, E. Krause, and R. Rannacher. Benchmark computations of laminar flow around a cylinder. *Notes on Numerical Fluid Mechanics (NNFM)*, page 547–566, 1996.

[19] Nitin Sharma and Neelesh A. Patankar. A fast computation technique for the direct numerical simulation of rigid particulate flows. *Journal of Computational Physics*, 205(2):439–457, May 2005.

[20] Stefan Turek. *Efficient Solvers for Incompressible Flow problems: An Algorithmic and Computational Approach.* Springer, 1999.

[21] Stefan Turek and Dmitri Kuzmin. Algebraic flux correction III: Incompressible flow problems. In Dmitri Kuzmin, Stefan Turek, and Rainald Löhner, editors, *Flux-Corrected Transport*, pages 239–297. Springer, 2 edition, 2012.

[22] Stefan Turek et al. *FeatFlow: Finite Element Software for the Incompressible Navier-Stokes Equations.* University of Dortmund, 2000. User manual.

[23] Markus Uhlmann. An immersed boundary method with direct forcing for the simulation of particulate flows. *Journal of Computational Physics*, 209(2):448–476, Nov 2005.

[24] C. Veeramani, P. D. Minev, and K. Nandakumar. A fictitious domain formulation for flows with rigid particles: A non-Lagrange multiplier version. *Journal of Computational Physics*, 224(2):867–879, Jun 2007.

[25] H. von Wahl. *Unfitted Finite Elements for Fluid-Rigid Body Interaction Problems*. Phd thesis, University of Magdeburg, 2021.

[26] Decheng Wan and Stefan Turek. Direct numerical simulation of particulate flow via multigrid fem techniques and the fictitious boundary method. *International Journal for Numerical Methods in Fluids*, 51(5):531–566, Dec 2005.

[27] Decheng Wan and Stefan Turek. Modeling of liquid-solid flows with large number of moving particles by multigrid fictitious boundary method. *Journal of Hydrodynamics*, 18(S1):93–100, Feb 2006.

[28] Decheng Wan and Stefan Turek. An efficient multigrid-FEM method for the simulation of solid–liquid two phase flows. *Journal of Computational and Applied Mathematics*, 203(2):561–580, Jun 2007.

[29] Decheng Wan and Stefan Turek. Fictitious boundary and moving mesh methods for the numerical simulation of rigid particulate flows. *Journal of Computational Physics*, 222(1):28–56, Mar 2007.

[30] B. H. Yang, J. Wang, D. D. Joseph, H. H. Hu, T.-W. Pan, and R. Glowinski. Numerical study of particle migration in tube and plane poiseuille flows. *Fluid Mechanics and Its Applications*, page 225–235.

Appendix A. Matrices and vectors of the discrete problem

The finite element approximations to the background velocity and pressure are given by

$$\mathbf{u}_h(\mathbf{x},t) = \sum_{j=1}^{N_h} u_j(t)\boldsymbol{\varphi}_j(\mathbf{x}), \quad p_h(\mathbf{x},t) = \sum_{k=1}^{M_h} p_k(t)\psi_k(\mathbf{x}),$$

where $\boldsymbol{\varphi}_j$ and ψ_k are basis functions spanning the spaces $\mathbf{V}_h$ and Q_h, respectively. In our description of discrete problems, $u = (u_j)$ and $p = (p_k)$ are vectors containing the coefficients of the above FE approximations. Recall that the systems of equations considered in Section 5 depend on

$$A = (a_{ij}), \qquad B = (b_{ik}), \qquad M_C = (m_{ij}), \qquad M_L = (\tilde{m}_{ij}), \qquad D = (d_{ij}).$$

In view of (7) and (8), the entries of these matrices and the components of $f = (f_i)$ are defined by

$$
\begin{aligned}
a_{ij} &= \frac{m_{ij}}{\Delta t} + \theta \left[\rho_f \int_\Omega (\mathbf{u}_h \cdot \nabla \boldsymbol{\varphi}_j) \cdot \boldsymbol{\varphi}_i \mathrm{d}\mathbf{x} + \frac{\mu_f}{2} \int_\Omega \mathbf{D}(\boldsymbol{\varphi}_j) : \mathbf{D}(\boldsymbol{\varphi}_i) \mathrm{d}\mathbf{x} \right], \\
b_{ik} &= -\int_\Omega \psi_k \nabla \cdot \boldsymbol{\varphi}_i \mathrm{d}\mathbf{x}, \qquad m_{ij} = \rho_f \int_\Omega \boldsymbol{\varphi}_j \cdot \boldsymbol{\varphi}_i \mathrm{d}\mathbf{x}, \qquad \tilde{m}_{ij} = \rho_f \boldsymbol{\varphi}_j(\mathbf{x}_i) \cdot \int_\Omega \boldsymbol{\varphi}_i \mathrm{d}\mathbf{x}, \\
d_{ij} &= \gamma_{\max} \sum_{k=1}^{N_p} \left[\int_{\hat{\Omega}_{k,h}} \beta_k \boldsymbol{\varphi}_j \cdot \boldsymbol{\varphi}_i \mathrm{d}\mathbf{x} + \int_{B_{k,h}} \boldsymbol{\varphi}_j \cdot \boldsymbol{\varphi}_i \mathrm{d}\mathbf{x} \right], \\
f_i &= \sum_{j=1}^{N_h} \left(\frac{m_{ij}}{\Delta t} - (1-\theta) \left[\rho_f \int_\Omega (\mathbf{u}_h \cdot \nabla \boldsymbol{\varphi}_j) \cdot \boldsymbol{\varphi}_i \mathrm{d}\mathbf{x} + \frac{\mu_f}{2} \int_\Omega \mathbf{D}(\boldsymbol{\varphi}_j) : \mathbf{D}(\boldsymbol{\varphi}_i) \mathrm{d}\mathbf{x} \right] \right) u_j.
\end{aligned}
$$

In the formula for $\tilde{m}_{ij}$, we denote by $\mathbf{x}_i$ the nodal point associated with the basis function $\boldsymbol{\varphi}_i$.

The data of the weakly imposed Dirichlet constraints is built into $g = (g_i)$ with

$$
g_i = \gamma_{\max} \sum_{k=1}^{N_p} \left[\int_{\hat{\Omega}_{k,h}} \beta_k \hat{\mathbf{u}}_h \cdot \boldsymbol{\varphi}_i \mathrm{d}\mathbf{x} + \int_{B_{k,h}} \mathbf{U}_h \cdot \boldsymbol{\varphi}_i \mathrm{d}\mathbf{x} \right].
$$