Report Numerical Methods

Numerical simulation of von Kármán vortex streets

V. Ballester and J. Nägerl

Paris, February 27th, 2024

Dauphine | PSL★
UNIVERSITÉ PARIS

Contents

1   Introduction

In the study of fluid dynamic systems, the *von Kármán vortex street* phenomenon stands a quintessential example of pattern formation in flows behind a bluff bodies. Characterized by alternating vortices it is not just an example of the complexity of fluid dynamics but also of great importance in various research fields influencing the design of objects in fluid- or aerodynamic systems. Simulating the Kármán vortex street plays a key role for advancements in aerodynamics, civil engineering, and environmental studies. For example the performance of a wing heavily depends on the specific flow, and therefore emergence of vortices can have an impact on the efficiency of the plane and its flight capabilities in various situations. The potential optimization of shape and coatings to achieve desired flight performances led alone in the field of aircraft engineering to a multitude of studies and research activities. Another very interesting occurrence of the phenomenon is shown in Figure 1, where the interplay of a mountain and wind leads to pattern formation in the clouds and forms alternating vortices.
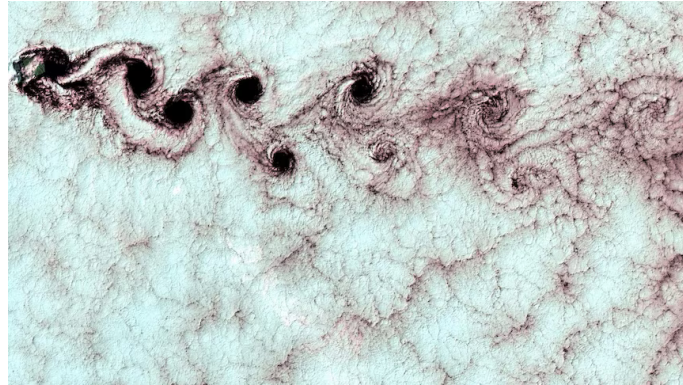
Figure 1: View of a von Kármán vortex street in the atmosphere, showcasing the mesmerizing pattern of swirling vortices caused by airflow around a mountain.

This project aims to numerically simulation of the phenomenon that accurately captures the dynamics of the the von Kármán vortex street behind bluff bodies in a two-dimensional flow field. By leveraging the *Chorin projection method* to decouple the velocity and pressure fields and using a *Semi-Langrangian* solver for the convective terms in the underlying equations, we aim to simplify the computational complexity inherent in fluid dynamics problems and to ensure stability and accuracy in the representation of vortex shedding and evolution over time. Therefore the plan for the following report is as follows HOW is it?

First we introduce the underlying equations in Section **??**, and present and motivate the solver in Section **??**. In Section **??** we numerically simulate the system (showing that we can observe it, test different forms and compare the observationss)

2   Methods

2.1   Posing the problem

To simplify our approach we look at a two-dimensional incompressible fluid. In this section we want to pose the problem by introducing the underlying mathematical equations and discuss the boundary conditions.

Underlying equations

In our approach we simplify the problem to a two-dimensional incompressible viscous fluid in a rectangular domain box flowing from one side of the box to the other, encountering an obstacle in its path. The motion of the fluid is described by the *incompressible Navier-Stokes equation*

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u}\right) - \mu\Delta\mathbf{u} + \nabla p = 0 \tag{1}$$
$$\nabla\cdot\mathbf{u} = 0$$

where ρ is the mass density (assumed constant), $\mathbf{u} = (u, v)$ is the fluid velocity with horizontal component u and vertical component v, μ is the dynamic viscosity of the fluid and p is the pressure. The first equation encapsulates the momentum balance within the fluid, incorporating the effects of convection (velocity interaction and movement), diffusion (velocity spreading due to viscosity), and the pressure gradient's influence on the fluid motion, whereas the second equation, often termed the continuity equation, asserts the incompressibility of the fluid by ensuring the volume conservation within the flow. The equations can be adimensionalized to obtain the dimensionless *Reynolds number* by looking at

$$\tilde{\mathbf{u}} = \frac{\mathbf{u}}{U}, \quad \tilde{p} = \frac{p}{\rho U^2}, \quad \tilde{\mathbf{x}} = \frac{\mathbf{x}}{L}, \quad \tilde{t} = \frac{U}{L}t \tag{2}$$

where the characteristic velocity U and length L are used. These quantities represent the typical velocity of the fluid and typical lengthscale. In practice they are taken to be the inflow velocity and the diameter of the object, respectively. Applying these change of variables to Eq. (1), we get (dropping the tilde for readability):

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u}\cdot\nabla\mathbf{u} - \frac{1}{\mathrm{Re}}\Delta\mathbf{u} + \nabla p = 0 \tag{3}$$
$$\nabla\cdot\mathbf{u} = 0$$

where the Reynolds number $\mathrm{Re} = \frac{\rho U L}{\mu}$ measures the ratio between the inertia of the flow and the viscosity of it.

Boundary conditions

The domain Ω is the rectangular box without the obstacle. We call the horizontal direction x and the vertical direction y-direction. In our study on the left side a laminar flow is coming into the domain. At the beginning the fluid inside the box is at rest. At the horizontal walls a slip condition is imposed and a free flow on the right side is assumed. Therefore we obtain as boundary conditions for the walls:

- At time $t = 0$, the fluid is at rest, and both velocity and pressure are zero.

- On the left side, the flow is incoming with a velocity equal to $U\mathbf{e}_x$, and the pressure satisfies the conditions $\frac{\partial p}{\partial x} = 0$.

- On the right side of the domain, the flow is free so that $\frac{\partial u}{\partial x} = \frac{\partial v}{\partial x} = 0$ and $p = 0$.

- On the horizontal sides, the walls are impenetrable, so that $v = 0$. A slip condition is imposed so that $\frac{\partial u}{\partial y} = 0$ and $\frac{\partial p}{\partial y} = 0$.

In order to accurately implement these boundary conditions, we made use of the *ghost cell* method. This technique involves the addition of a layer of cells outside the domain (see Fig. 2). These cells are used to approximate at first order the values of the velocity and pressure at the boundary for both Dirichlet and Neumann conditions
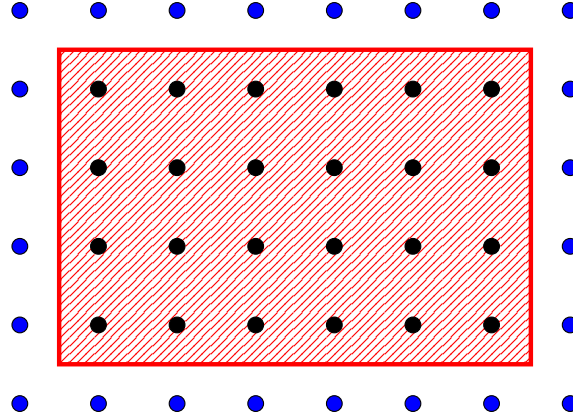
Figure 2: The ghost cell method. The domain Ω is represented in red and the dots represent the grid cells, the black ones being the inner points and the blue ones, the ghost cells.

As an example, if $u_{0,j}$ denotes the horizontal component of $\mathbf{u}$ at the left boundary and height j (counting from 0), then it will be approximated by $2U - u_{1,j}$, where $u_{1,j}$ is the horizontal component of $\mathbf{u}$ at the first cell inside the domain and the same height j. A straightforward Taylor expansion around the boundary point (say $u_{1/2,j}$) shows us that indeed this approximation is of second order. Similarly, we obtain values for $v_{0,j}$ and $p_{0,j}$ as

$$v_{0,j} = -v_{1,j}, \quad p_{0,j} = p_{1,j}$$

The other boundaries are treated in the same way. It should be noted than when

computing discrete derivatives, which involve the use of neighboring cells, the computations are carried out only with inner cells, which in turn use ghost cells. But the derivatives itself at ghost cells are never computed.

We discuss now the treatment of the boundary conditions on the object. In general dealing with the object is not an easy task. For simplicity, we model the presence of a solid body within a fluid by enforcing the fluid velocity to be zero at the boundaries that correspond to the shape of the body, which correspond to imposing a no-slip condition. It is very worth-mentioning that the way we integrate the body has an influence on the flow. Especially to study the impacts of coatings or surface structure, our method, is not capable of including these effects, but in our approach we only want to observe the effect, and therefore prioritize simplicity for understanding over complexity for small details in the effect. If the reader is interested in delving deeper into the fluid-structure interactions, including the effects of surface modifications, we recommend consulting works in the field, such as those by Peskin [Pes77] or Schlichting and Gersten [SG00].

2.2   Difference Scheme

In this section we discuss our numerical solver to integrate Eq. (1) together with the boundary conditions presented in above section. To simplify the equations, we use *Chorin's projection method*. It was originally introduced by Alexandre Chorin in 1967 [Cho67] as an efficient means of solving the incompressible Navier-Stokes equations.

The idea of the method is to first predict the velocities for the next time step by solving the advection and the diffusion term. With the intermediate velocity field we solve the Poisson equation with a finite difference method (described below). Finally, the *projection step* is preformed, and the velocity field is updated with the new pressure field. To ensure that the boundary conditions are met, we impose the conditions for the velocity right before solving the Poisson equation and at the end of each iteration. The boundary conditions for the pressure are imposed after solving the Poisson equation and before correcting the velocities in the projection step.

Next, we provide a summary of the steps in our method. Assuming we know the velocity field at time t^n, $\mathbf{u}^n$, we want to compute the velocity field at time t^{n+1}, $\mathbf{u}^{n+1}$.

1. $\dfrac{\mathbf{u}^a - \mathbf{u}^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{u}^n = 0$

2. $\dfrac{\mathbf{u}^* - \mathbf{u}^a}{\Delta t} = \dfrac{1}{\text{Re}} \Delta \mathbf{u}^n$

3. Set the boundary conditions for the intermediate velocity field $\mathbf{u}^*$ and impose set the velocity field $\mathbf{u}^*$ inside the object to be zero (forcing term).

4. $\Delta p^n = \dfrac{1}{\Delta t} \nabla \cdot \mathbf{u}^*$

5. Set the boundary conditions for the pressure p^n.

6. $\mathbf{u}^{n+1} = \mathbf{u}^* - \Delta t \nabla p^n$

7. Set the boundary conditions for the velocity field $\mathbf{u}^{n+1}$.

From these equations, one can easily check that we have:

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{u}^n - \frac{1}{\mathrm{Re}} \Delta \mathbf{u}^n + \nabla p^n = 0$$

$$\nabla \cdot \mathbf{u}^{n+1} = 0$$

In the following sections we will deepen in how do we compute each of the above steps in our method.

Semi-Lagrangian method

The Semi-Lagrangian method represents a powerful approach to solving fluid dynamics problems, particularly in the context of advection. Unlike Eulerian methods, which compute changes at fixed points in space, the Semi-Lagrangian (SL) method tracks the motion of fluid parcels. SL methods slightly differ from Lagrangian methods, as the word suggest. The second ones are rarely used in numerical methods because the particle trajectories become chaotic and wildly mixed in a short period of time. However, SL algorithms avoid this problem by 'reinitializing' the Lagrangian coordinate system after each time step [Boy01]. There are several ways to implement a SL method, and we will describe the one we used in our method. Our goal in this subsection is on solving:

$$\frac{\mathrm{D}\mathbf{u}}{\mathrm{D}t} = \frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} = 0$$

The idea is to somehow discretize the material derivative. To do that, if we call $\mathbf{u}^n$ the current velocity field and $\mathbf{u}^a$ the velocity field after the convection step, we can write a difference formula as:

$$\frac{\mathbf{u}^a - \mathbf{u}^n}{\Delta t} + \mathbf{u}^n \cdot \nabla \mathbf{u}^n = 0$$

The following discretization for the material derivative is used:

$$\frac{\mathbf{u}(\mathbf{x}_{i,j}, t^{n+1}) - \mathbf{u}(\mathbf{x}_{i,j} - \mathbf{u}(\mathbf{x}_{i,j}, t^n)\Delta t, t^n)}{\Delta t}$$

The reader may rapidly notice that the point $\mathbf{x}_{i,j} - \mathbf{u}(\mathbf{x}_{i,j}, t^n)\Delta$ will not in general be in the grid, and thus, we will not have the respective value of the velocity field. To solve this problem, we will make use of bilinear interpolation between the four (in 2D) nearest grid points to the point $\mathbf{x}_{i,j} - \mathbf{u}(\mathbf{x}_{i,j}, t^n)\Delta t$. This will give us an approximation of the velocity field at the point $\mathbf{x}_{i,j} - \mathbf{u}(\mathbf{x}_{i,j}, t^n)\Delta t$, which in turn will give us the convected velocity field $\mathbf{u}^a$.

Furthermore, to improve the accuracy of the method we use the back-and-forth method [DL07]. This method first advects the velocity field using the method described above, and then advects back with opposite velocity $\mathbf{u} \to -\mathbf{u}$. With this we can have an estimate of the error we are comitting when comparing the last result with the initial position. Using this information we can correct the velocity field and obtain a more accurate result. The steps are reproduced below. In order to make things more clear, we will consider the general case:

$$\frac{\mathrm{D}\boldsymbol{\psi}}{\mathrm{D}t} = \frac{\partial \boldsymbol{\psi}}{\partial t} + \mathbf{u} \cdot \nabla \boldsymbol{\psi} = 0$$

1. Advect the field $\boldsymbol{\psi}^n$ with velocity $\mathbf{u}$ to obtain $\tilde{\boldsymbol{\psi}}^a$.

2. Advect the field $\tilde{\boldsymbol{\psi}}^a$ with velocity $-\mathbf{u}$ to obtain $\bar{\boldsymbol{\psi}}^a$.

3. Advect the field $\boldsymbol{\psi}^n + \frac{1}{2}(\boldsymbol{\psi}^n - \bar{\boldsymbol{\psi}}^a)$ with velocity $\mathbf{u}$ to obtain $\boldsymbol{\psi}^a$.

Diffusion term

As a second step we add diffusion using a 2nd order central differences. The diffusion term is given by:

$$\frac{\mathbf{u}^* - \mathbf{u}^a}{\Delta t} = \frac{1}{\mathrm{Re}} \Delta \mathbf{u}^n$$

Once discretized we get, for each i, j in the grid:

$$\mathbf{u}^*_{i,j} = \mathbf{u}^a_{i,j} + \frac{\Delta t}{\mathrm{Re}} \left(\frac{\mathbf{u}^n_{i+1,j} - 2\mathbf{u}^n_{i,j} + \mathbf{u}^n_{i-1,j}}{(\Delta x)^2} + \frac{\mathbf{u}^n_{i,j+1} - 2\mathbf{u}^n_{i,j} + \mathbf{u}^n_{i,j-1}}{(\Delta y)^2} \right)$$

As before, we write concisely the discretization in vector form but in practice we do it for each component of the velocity field separately.

Solving for the Poisson equation

There are several efficient ways to compute the solution of the general Poisson equation $\Delta p = f$. In our case, we have to equip this equation with the boundary conditions $\partial_{\mathbf{n}} p = 0$ on the left, top and bottom boundaries, and $p = 0$ on the right boundary. We will use a finite difference method to solve this equation. The idea is to discretize the domain into a grid and use the finite difference method to solve the equation. We will use a 5-point stencil to approximate the Laplacian operator. The 5-point stencil is given by:

$$\Delta p_{i,j} = \frac{p_{i+1,j} - 2p_{i,j} + p_{i-1,j}}{(\Delta x)^2} + \frac{p_{i,j+1} - 2p_{i,j} + p_{i,j-1}}{(\Delta y)^2}$$

If we set $\mathbf{p} := (p_{11}, p_{12}, \ldots, p_{1n_y}, p_{21}, \ldots, p_{22}, \ldots, p_{2n_y}, \ldots p_{n_x n_y})$, we can write our problem in matrix form as $\mathbf{A}\mathbf{p} = \mathbf{f}$, where $\mathbf{f}$ contains the discrete values of the

right-hand side of the Poisson equation in the same order as $\mathbf{p}$, and $\mathbf{A}$ is a matrix that contains the coefficients of the Laplacian operator. Let's describe how we can build the matrix $\mathbf{A}$. Let $\mathbf{A} = \mathbf{X} + \mathbf{Y}$, where $\mathbf{X}$ and $\mathbf{Y}$ are the matrices that contain the coefficients of the Laplacian operator in the x and y directions, respectively. We can write $\mathbf{X}$ and $\mathbf{Y}$ as:

$$
\mathbf{X} = \begin{pmatrix}
-\mathbf{I}_{n_y} & \mathbf{I}_{n_y} & \mathbf{0} & \cdots & \cdots & \mathbf{0} \\
\mathbf{I}_{n_y} & -2\mathbf{I}_{n_y} & \mathbf{I}_{n_y} & \ddots & & \vdots \\
\mathbf{0} & \mathbf{I}_{n_y} & -2\mathbf{I}_{n_y} & \mathbf{I}_{n_y} & \ddots & \vdots \\
\vdots & \ddots & \ddots & \ddots & \ddots & \mathbf{0} \\
\vdots & & \ddots & \mathbf{I}_{n_y} & -2\mathbf{I}_{n_y} & -\mathbf{I}_{n_y} \\
\mathbf{0} & \cdots & \cdots & \mathbf{0} & \mathbf{I}_{n_y} & -3\mathbf{I}_{n_y}
\end{pmatrix}
\qquad
\mathbf{Y} = \begin{pmatrix}
\mathbf{B} & \mathbf{0} & \cdots & \mathbf{0} \\
\mathbf{0} & \mathbf{B} & \ddots & \vdots \\
\vdots & \ddots & \ddots & \mathbf{0} \\
\mathbf{0} & \cdots & \mathbf{0} & \mathbf{B}
\end{pmatrix}
$$

with

$$
\mathbf{B} = \begin{pmatrix}
-1 & 1 & 0 & \cdots & 0 \\
1 & -2 & 1 & \ddots & \vdots \\
0 & \ddots & \ddots & \ddots & 0 \\
\vdots & \ddots & 1 & -2 & 1 \\
0 & \cdots & 0 & 1 & -1
\end{pmatrix} \in \mathcal{M}_{n_y}(\mathbb{R})
$$

Both $\mathbf{X}$ and $\mathbf{Y}$ are block matrices, composed of n_x matrices of size $n_y \times n_y$ each, which results in a matrix of size $n_x n_y \times n_x n_y$. The boundary conditions (using ghost cells) on the pressure are applied at both the first and last block row of $\mathbf{X}$ and the first and last row of $\mathbf{Y}$.

In order to solve the resulting system, we have used the Cholesky decomposition, which requires the matrix to be symmetric and positive definite. The matrix $\mathbf{A}$ is symmetric, but it is not positive definite. In fact, it can be seen that it is negative definite. Thus, in our case, we will be solving the equivalent system $-\mathbf{A}\mathbf{p} = -\mathbf{f}$.

Updating velocities

As a final step, we compute the gradient of the pressure field and update the velocity field. The gradient of the pressure field is computed using a 2nd order central differences. The update of the velocity field is given by:

$$
\mathbf{u}_{i,j}^{n+1} = \mathbf{u}_{i,j}^* - \Delta t \left(\frac{p_{i+1,j}^n - p_{i,j}^n}{\Delta x}, \frac{p_{i,j+1}^n - p_{i,j}^n}{\Delta y} \right)
$$

3   Results

Implementing the numerical scheme outlined in Section 2.2, we have the opportunity to explore the dynamics of fluid flow around various objects. [1]   First, we want to illustrate the emergence of the phenomenon, before delving in different factors on the flow as differences in Reynolds numbers or in the shape of the object.

3.1   Observing Karman vortex street

To investigate how the phenomenon emerge, we look at an exemplary showcase of a fluid with a dynamic viscosity $\mu = 1.506 \times 10^{-5}$ which is similar to air. We simulate the fluid in a domain of the width $W = 1$ and the length $L = 5$. On the right side of the domain we put a circle with radius $r = 0.125$ as an object. With the Reynolds number Re We start the simulation with the fluid at rest. Imposing the boundary conditions as presented in Section 2.1, the fluid starts to move from the left side to the right side of the domain. As the simulation progresses, the flow dynamics evolve to reveal the organized structure known as a Karman vortex street. In Fig. 3, we present the outcomes of this prolonged simulation, showcasing the absolute velocity field and the vorticity $\omega = \frac{\partial v}{\partial x} - \frac{\partial u}{\partial y}$, to visualize the vortices. In the figure we can observe the characteristic periodicity between right hand sided and left hand sided vortices. In front of the body the flow seems to be laminar with an income velocity of approximately $v = 1$, which is consistent with our boundary conditions.

(a) Absolute velocity $|\vec{v}|$
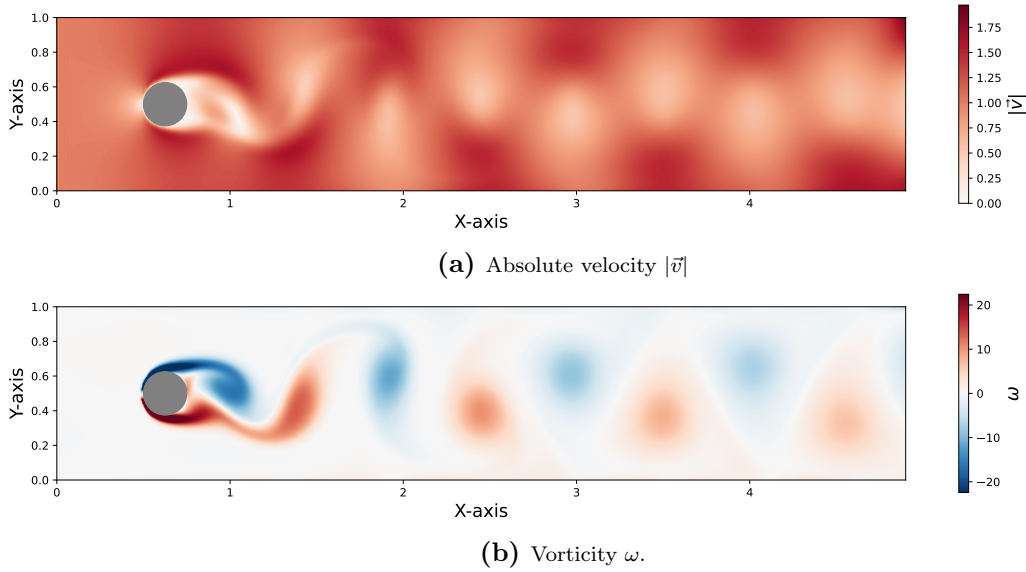
(b) Vorticity ω.

Figure 3: Flow dynamics around a circular obstacle in a fluid with dynamic viscosity similar to air ($\mu = 1.506 \times 10^{-5}$) at time $t = 25$. The top image displays the absolute velocity $|\vec{v}|$, and the bottom image focuses on the vorticity ω.

---

[1]For further details on the implementation, we shared our code in the accompanying C++ code documentation.

3.2   Influence of Reynolds number

The Reynolds number (Re) plays an important role in determining the flow characteristics around bodies in a fluid. As explained in Section 2.1, it is a dimensionless quantity that represents the ratio of inertial forces to viscous forces and serves as a key parameter in predicting the transition from laminar to turbulent flow. In this section, the object is the same circle as before and our characteristic length is therefore $L_c = 2r = 0.25$. Also the dynamical viscosity is set as before to $\mu = 1.506 \times 10^{-5}$. Therefore by varying the Reynolds number we mean to vary the incoming speed on the left side The impact of the Reynolds number can be categorized in different regimes.

plots to compare different reynolds numbers e.f. frequency of vortexes by taking the average in one column,

show that there is a maximum/ minimum re for observing the phenomen

3.3   Influence of the object

Compare two shapes at same reynolds numbers.

4   Discussion

References

[Boy01]   J. P. Boyd. *Chebyshev and Fourier Spectral Methods*. 2nd. PDF version. Dover Publications, 2001.

[Cho67]   A. J. Chorin. "The numerical solution of the Navier-Stokes equations for an incompressible fluid". In: *Bull. Am. Math. Soc.* 73.6 (1967), pp. 928–931. DOI: 10.1090/S0002-9904-1967-11853-6.

[DL07]   T. Dupont and Y. Liu. "Back and forth error compensation and correction methods for semi-Lagrangian schemes with application to level set interface computations". In: *Math. Comput.* 76 (Apr. 2007), pp. 647–668. DOI: 10.1090/S0025-5718-06-01898-9.

[Pes77]   C. S. Peskin. "The Immersed Boundary Method: Numerical Solutions of PDEs Involving Fluid Flow and Interacting Boundaries". In: *Journal of Computational Physics* something (1977), something.

[SG00]   H. Schlichting and K. Gersten. *Boundary-Layer Theory*. 9th. Springer, 2000.

5   Appendix