Laminar Flow Around a Cylinder

Bice Marzagora, Riccardo Selis, Mara Tortorella

July 16, 2024

Incompressible Navier-Stokes Equations

$$\begin{cases} \dfrac{\partial \mathbf{u}}{\partial t} - \nu \Delta \mathbf{u} + (\mathbf{u} \cdot \nabla)\mathbf{u} + \nabla p = \mathbf{f}, & \text{in } \Omega \\ \nabla \cdot \mathbf{u} = 0, & \text{in } \Omega \\ \mathbf{u} = \mathbf{0}, & \text{on } \Gamma_D \\ \nu \dfrac{\partial \mathbf{u}}{\partial \mathbf{n}} - p\mathbf{n} = \mathbf{d}, & \text{on } \Gamma_N \\ \mathbf{u}|_{t=0} = \mathbf{u}_0, & \text{in } \Omega \end{cases} \tag{1}$$

Where:

- $u \in \mathbf{R}^d$: velocity vector field
- $p \in \mathbf{R}$: pressure scalar field
- ν : kinematic viscosity
- $\mathbf{f} \in \mathbf{R}^d$: forcing term

Chosen Spaces and Weak Formulation

The chosen spaces function are:

- $V = \left[H^1_{\Gamma_D}(\Omega)\right]^d = \left\{\mathbf{v} \in \left[H^1(\Omega)\right]^d : \mathbf{v}\big|_{\Gamma_D} = 0\right\}$
- $Q = L^2(\Omega)$

Find $(u(t), p(t)) \in V \times Q, u(0) = u_0, u = \mathbf{0}$ on Γ_D s.t. for $t > 0$

$$\begin{cases} \int_\Omega \frac{\partial \mathbf{u}}{\partial t} \cdot \mathbf{v}\, d\Omega + \int_\Omega \nu \nabla \mathbf{u} : \nabla \mathbf{v}\, d\Omega + \int_\Omega [(\mathbf{u} \cdot \nabla)\mathbf{u}] \cdot \mathbf{v}\, d\Omega \\ -\int_\Omega p \nabla \cdot \mathbf{v}\, d\Omega \quad = \int_\Omega \mathbf{f} \cdot \mathbf{v}\, d\Omega + \int_{\Gamma_N} \mathbf{d} \cdot \mathbf{v}\, ds \quad \forall \mathbf{v} \in \mathbf{V}, \\ \int_\Omega q \nabla \cdot \mathbf{u}\, d\Omega = 0 \quad \forall q \in Q \\ \mathbf{u}|_{t=0} = \mathbf{u}_0, \qquad \text{in } \Omega \end{cases} \tag{2}$$

Taylor-Hood Finite Elements Space

We approximate Navier-Stokes using **Taylor-Hood** finite elements, so the spaces are defined such as:

$$\mathbf{V}_h = \left\{ \mathbf{v}_h \in [C^0(\Omega)]^d,\ \mathbf{v}_h|_K \in (\mathbb{P}^{k+1})^d\ \forall K \in \mathcal{T}_h,\ \mathbf{v}_h|_{\Gamma_D} = \mathbf{0} \right\}$$

$$Q_h = \left\{ q_h \in C^0(\Omega),\ q_h|_K \in \mathbb{P}^k\ \forall K \in \mathcal{T}_h,\ \int_\Omega q_h = 0 \text{ if } \Gamma_D = \partial\Omega \right\}$$

The Taylor-Hood finite elements are *inf-sup stable* due to their specific construction of using higher-order polynomials for the velocity field and lower-order polynomials for the pressure field, leading to a stable and accurate numerical solution for the Navier-Stokes equations without spurious pressure modes.

Galerkin Discretization

Find $(\mathbf{u}_h, p_h) \in \mathbf{V}^h \times Q^h$ such that for all $(\mathbf{v}_h, q_h) \in \mathbf{V}^h \times Q^h$:

$$\begin{cases} \displaystyle\int_\Omega \frac{\partial \mathbf{u}_h}{\partial t} \cdot \mathbf{v}_h \, d\Omega + \nu \int_\Omega \nabla \mathbf{u}_h : \nabla \mathbf{v}_h \, d\Omega + \int_\Omega [(\mathbf{u}_h \cdot \nabla)\mathbf{u}_h] \cdot \mathbf{v}_h d\Omega \\ \displaystyle -\int_\Omega p_h (\nabla \cdot \mathbf{v}_h) \, d\Omega = \int_\Omega \mathbf{f} \cdot \mathbf{v}_h \, d\Omega + \int_{\Gamma_N} \mathbf{d} \cdot \mathbf{v}_h \, d\Gamma, \\ \displaystyle\int_\Omega q_h (\nabla \cdot \mathbf{u}_h) \, d\Omega = 0 \\ \mathbf{u}_h|_{t=0} = \mathbf{u}_{0,h}, \qquad \text{in } \Omega \end{cases} \tag{3}$$

Semi-Implicit Backward Euler

We consider an **semi-implicit backward Euler scheme** for the time-discretization, in which the linear part of the equation is advanced implicitly, while nonlinear terms explicitly:

$$\begin{cases} \dfrac{\mathbf{u}^{n+1}-\mathbf{u}^{n}}{\Delta t} - \nu\Delta\mathbf{u}^{n+1} + (\mathbf{u}^{n}\cdot\nabla)\mathbf{u}^{n+1} + \nabla p^{n+1} = \mathbf{f}^{n+1}, \quad \text{in } \Omega \\ \nabla\cdot\mathbf{u}^{n+1} = 0 \end{cases} \tag{4}$$

The main advantage of this approach, compared to the fully implicit method, is that it results in a linear system that needs to be solved at each time step.

Fully Discretized Problem

The approximation of $(\mathbf{u}, p)$ at time t_n is given by $(\mathbf{u}_h^n, p_h^n) \in V_h \times Q_h$. Thanks to the Galerkin projection of $V_h \times Q_h$, the discrete system is obtained as follows:

$$
\begin{cases}
\dfrac{1}{\Delta t}\displaystyle\int_{\Omega_h} \mathbf{u}_h^{n+1}\varphi_i \, d\Omega + \int_{\Omega_h} \nu \nabla \mathbf{u}_h^{n+1} : \nabla \varphi_i \, d\Omega \\
\quad + \displaystyle\int_{\Omega_h} [(\mathbf{u}_h^n \cdot \nabla)\mathbf{u}_h^{n+1}] \cdot \varphi_i \, d\Omega - \int_{\Omega_h} p_h^{n+1} \nabla \cdot \varphi_i \, d\Omega \\
= \displaystyle\int_{\Omega_h} \mathbf{f}(t_{n+1}) \cdot \varphi_i \, d\Omega + \int_{\partial\Omega_h} \mathbf{d}(t_{n+1}) \cdot \varphi_i \, ds \qquad (5) \\
+ \dfrac{1}{\Delta t}\displaystyle\int_{\Omega_h} \mathbf{u}_h^n \varphi_i \, d\Omega, \qquad \forall i = 1, \ldots, N_u, \\
\displaystyle\int_{\Omega_h} \phi_j \nabla \cdot \mathbf{u}_h^{n+1} \, d\Omega = 0, \qquad \forall j = 1, \ldots, N_p.
\end{cases}
$$

Matrix Representation and Stability

At each time step, the linear system to be solved is given by:

$$\begin{bmatrix} \frac{M}{\Delta t} + A + N(\mathbf{U}^n) & B^T \\ -B & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u}^{n+1} \\ p^{n+1} \end{bmatrix} = \begin{bmatrix} \mathbf{G} \\ 0 \end{bmatrix} \tag{6}$$

The resulting matrices are **non-symmetric** due to the nature of the convective term, which can be more complex to solve but still more efficient compared to handling nonlinear systems, typically solved using iterative methods like Newton's method. The semi-implicit method is conditionally stable, the stability restriction on the time step takes the following form

$$\Delta t \leq \frac{C}{\max_{\mathbf{x}\in\Omega} |\mathbf{u}^n(\mathbf{x})|}. \tag{7}$$

Solver structure

The implementation of the Navier-Stokes Solver using deal.II is structured as follow:

- Setup
- Assemble constant part of the matrix

At each time step:

- Assemble time dependent part of the matrix, assemble matrices needed for preconditioners, apply boundary conditions
- Solve linear system
- Calculate Drag and Lift force

Linearization: Skew-symmetric form

The nonlinearity in the Navier-Stokes equations can be written in several ways, which lead to discretizations with different algorithmic costs, conserved quantities, and approximation accuracy. In the solver we implemented also the skew-symmetric form, given by:

$$\frac{1}{2}((\mathbf{u}\cdot\nabla)\mathbf{u} + (\nabla\mathbf{u})^T\mathbf{u})$$

Simulations: Skew symmetric form vs semi-implicit treatment of the convective term

Figure: Without skew symmetric form (3D)

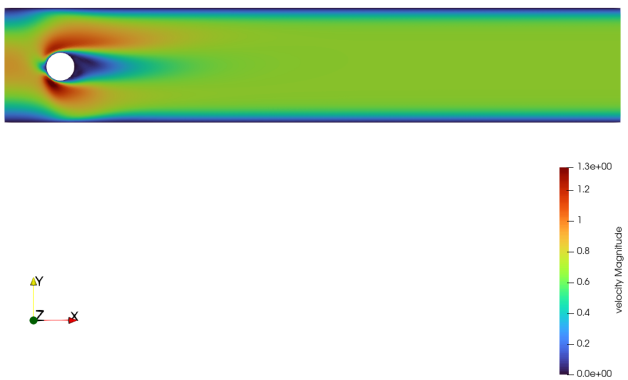


Figure: With skew symmetric form (3D)

Case study: flow past a cylinder

The problem geometry is represented in the following images[1]:

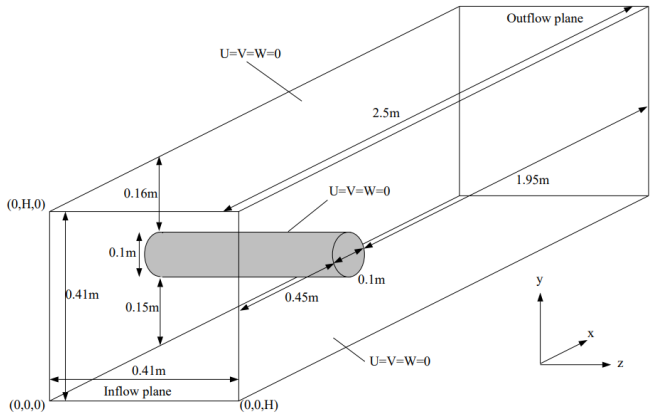


Figure: 3D case

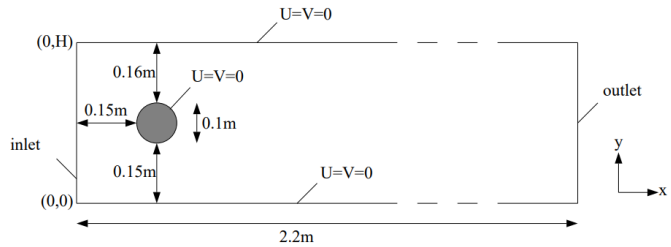


Figure: 2D case

The corresponding meshes are:

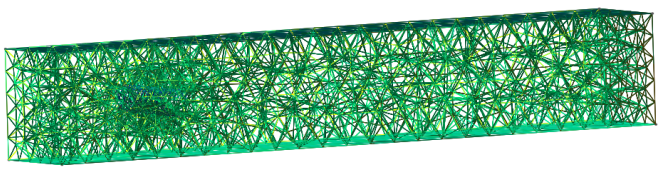


Figure: 3D coarse mesh

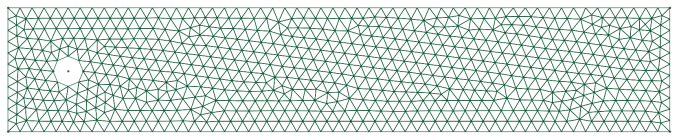


Figure: 2D coarse mesh

[1] F. Durst, E. Krause, R. Rannache, M. Schäfer, S. Turek, *Benchmark computations of laminar flow around a cylinder*, Vieweg+Teubner, pp. 547–566, 1996.

Simulations: Re = 20

Results of velocity and pressure with Reynolds Number 20 using skew-symmetric form, velocity = 0.3 m/s, viscosity = 0.001 m^2/s.. The flow is laminar. Near the upstream stagnation point, the fluid velocity decreases, resulting in a maximum pressure. As the fluid moves around the cylinder, it accelerates along the surface, leading to a reduction in pressure.

Simulation: Re = 60

As the Reynolds number increases, the flow should becomes more unstable and transitional, eventually leading to turbulence. However, at a Reynolds number of 60, the flow still exhibits laminar characteristics.

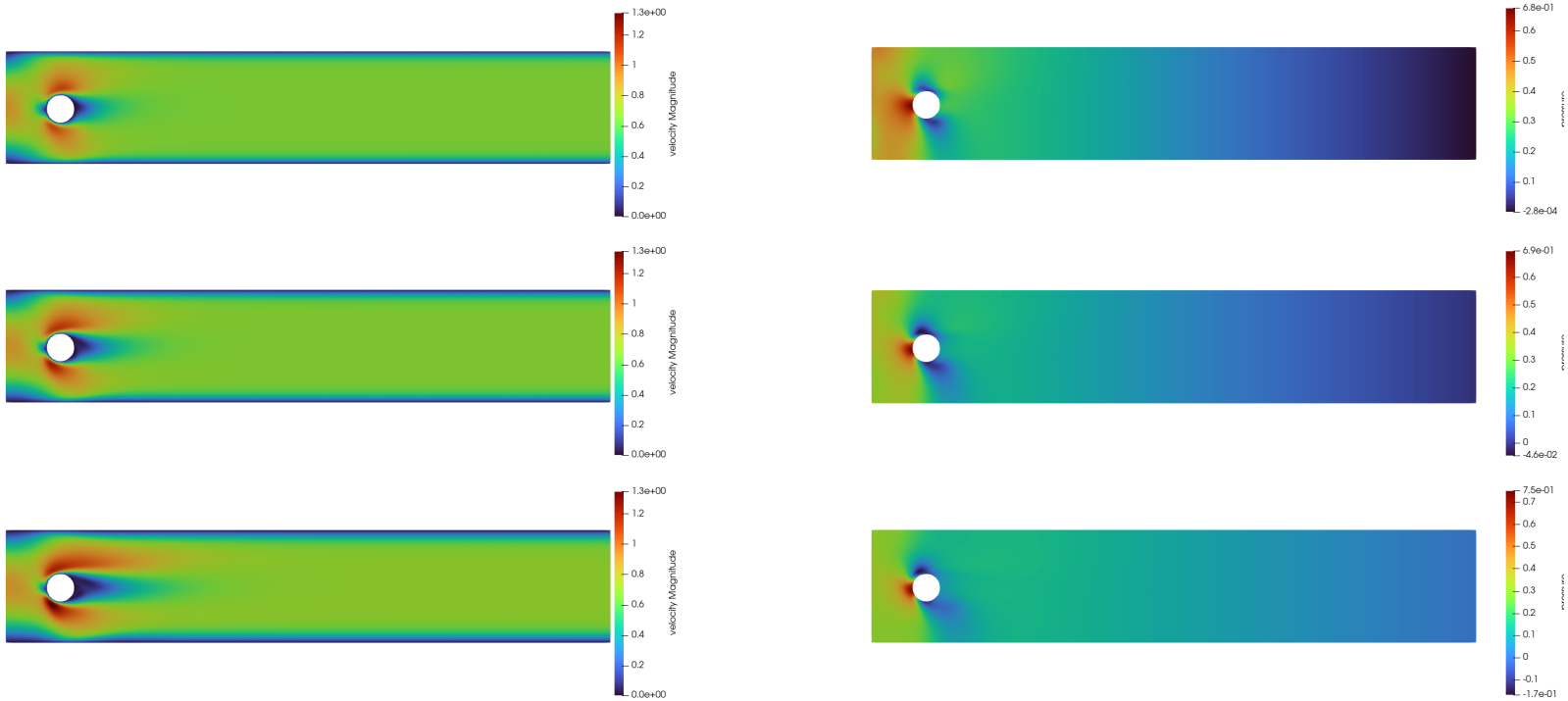
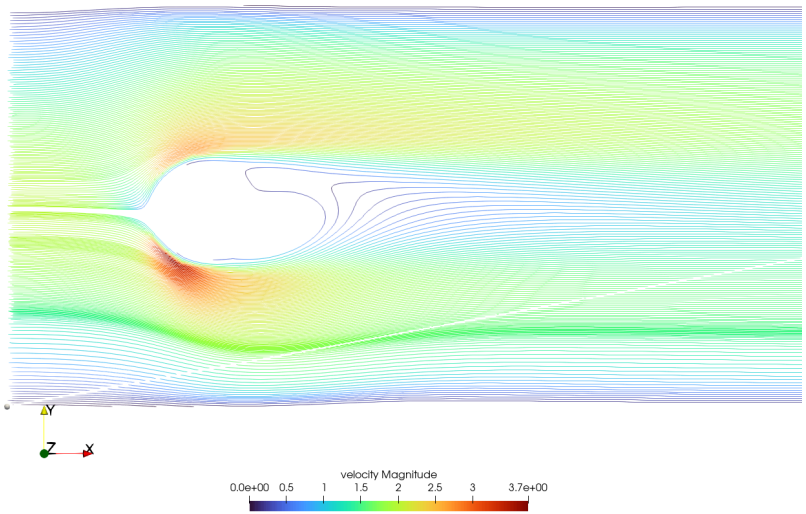
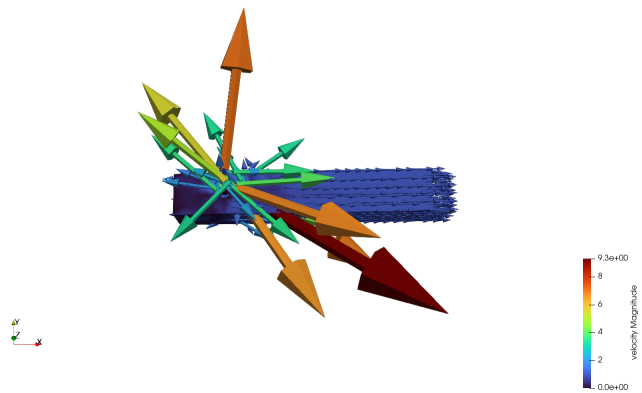


Figure: Time evolution with $Re = 60$, 3D

Instability of the solution

For high Reynolds numbers the solution becomes unsteady and the solvers blows up even with smaller time steps and finer meshes.





Alternative approaches may be necessary to achieve stable and accurate results.

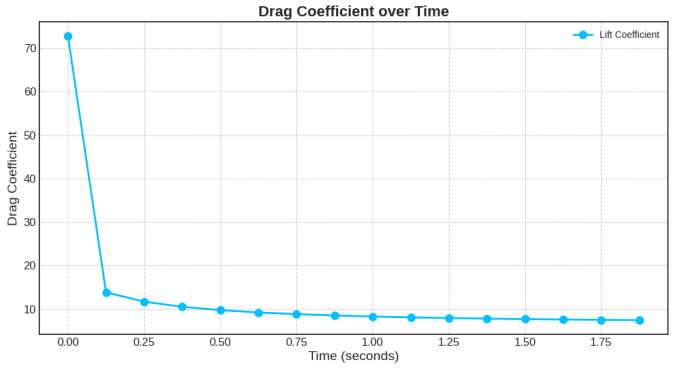
Drag and lift coefficients results

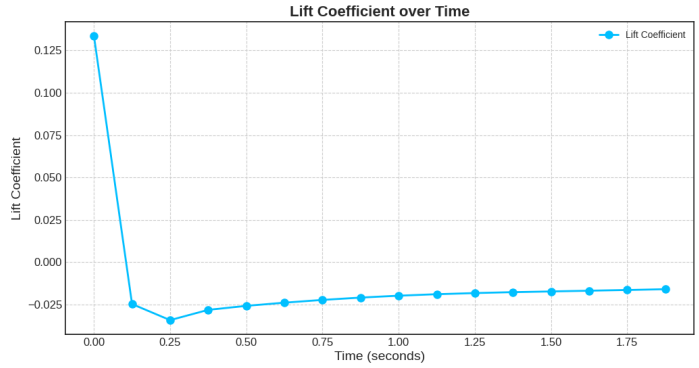
Drag and lift coefficient are dimensionless parameters and are directly proportional to the lift and drag forces experienced by an object. These forces are due to pressure and viscous forces acting on a body surface.

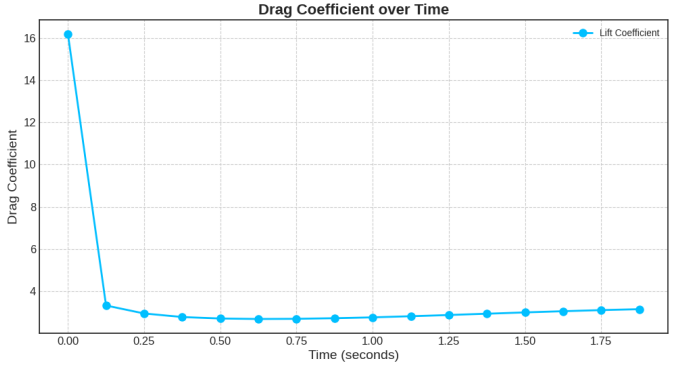
- Drag: component parallel to flow direction.
- Lift: component normal to flow direction.

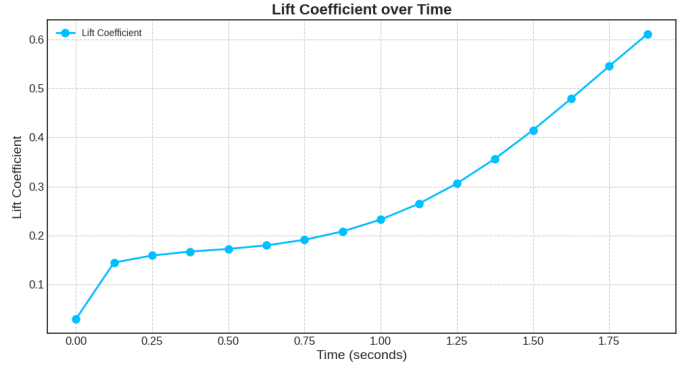
$$C_D = \frac{2}{U^2 L} F_D \quad C_L = \frac{2}{U^2 L} F_L,$$

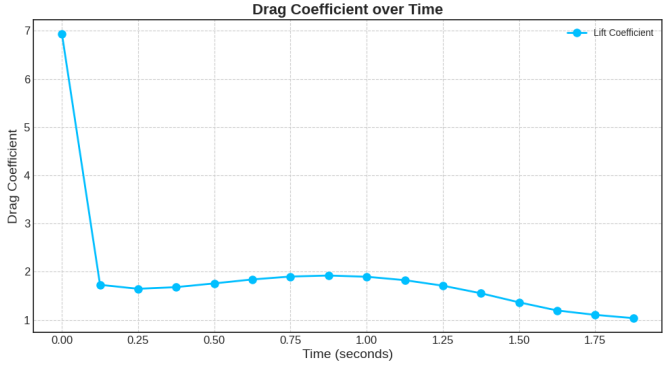
In the next slide drag and lift coefficients are plotted over time for $\mathrm{Re} = 4, 20, 60$

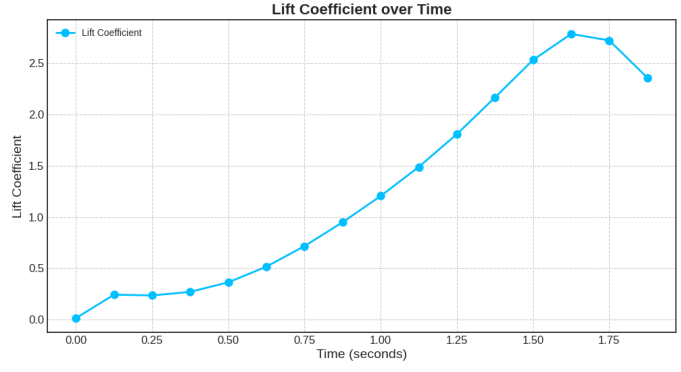

Drag Coefficient over Time
Lift Coefficient
Drag Coefficient
Time (seconds)


Lift Coefficient over Time
Lift Coefficient
Lift Coefficient
Time (seconds)


Drag Coefficient over Time
Lift Coefficient
Drag Coefficient
Time (seconds)


Lift Coefficient over Time
Lift Coefficient
Lift Coefficient
Time (seconds)


Drag Coefficient over Time
Lift Coefficient
Drag Coefficient
Time (seconds)


Lift Coefficient over Time
Lift Coefficient
Lift Coefficient
Time (seconds)

Preconditioners: SIMPLE

$$P_{SIMPLE} = \begin{bmatrix} F & 0 \\ B & -\tilde{S} \end{bmatrix} \begin{bmatrix} I & D^{-1}B^T \\ 0 & \alpha I \end{bmatrix} \tag{8}$$

- D: diagonal of F,
- α: parameter that damps the pressure update,
- $\tilde{S} = BD^{-1}B^T$.

Preconditioners: Yosida

$$P_Y = \begin{bmatrix} F & 0 \\ B & -\tilde{S} \end{bmatrix} \begin{bmatrix} I & F^{-1}B^T \\ 0 & I \end{bmatrix} \tag{9}$$

- $\tilde{S} = \Delta t B M_u^{-1} B^T$,
- M_u: mass matrix associated to the velocity shape functions.

Limitations of SIMPLE and Yosida

The efficiency of both these preconditioners deteriorates as Δt increases:

- SIMPLE: $\tilde{S}$ becomes a poor approximation of the exact Schur complement S of the approximation of the Schur complement;
- Yosida: the approximation of becomes less accurate, and the dominance of the mass matrix M_u in F diminishes, leading to decreased performance.

Preconditioners: aSIMPLE

$$P^{-1}_{\text{aSIMPLE}} = \begin{bmatrix} D^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} I & -B^T \\ 0 & I \end{bmatrix} \begin{bmatrix} D & 0 \\ 0 & \frac{1}{\alpha} I \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & \hat{S}^{-1} \end{bmatrix} \begin{bmatrix} I & 0 \\ -B & I \end{bmatrix} \begin{bmatrix} \hat{F}^{-1} & 0 \\ 0 & I \end{bmatrix}$$

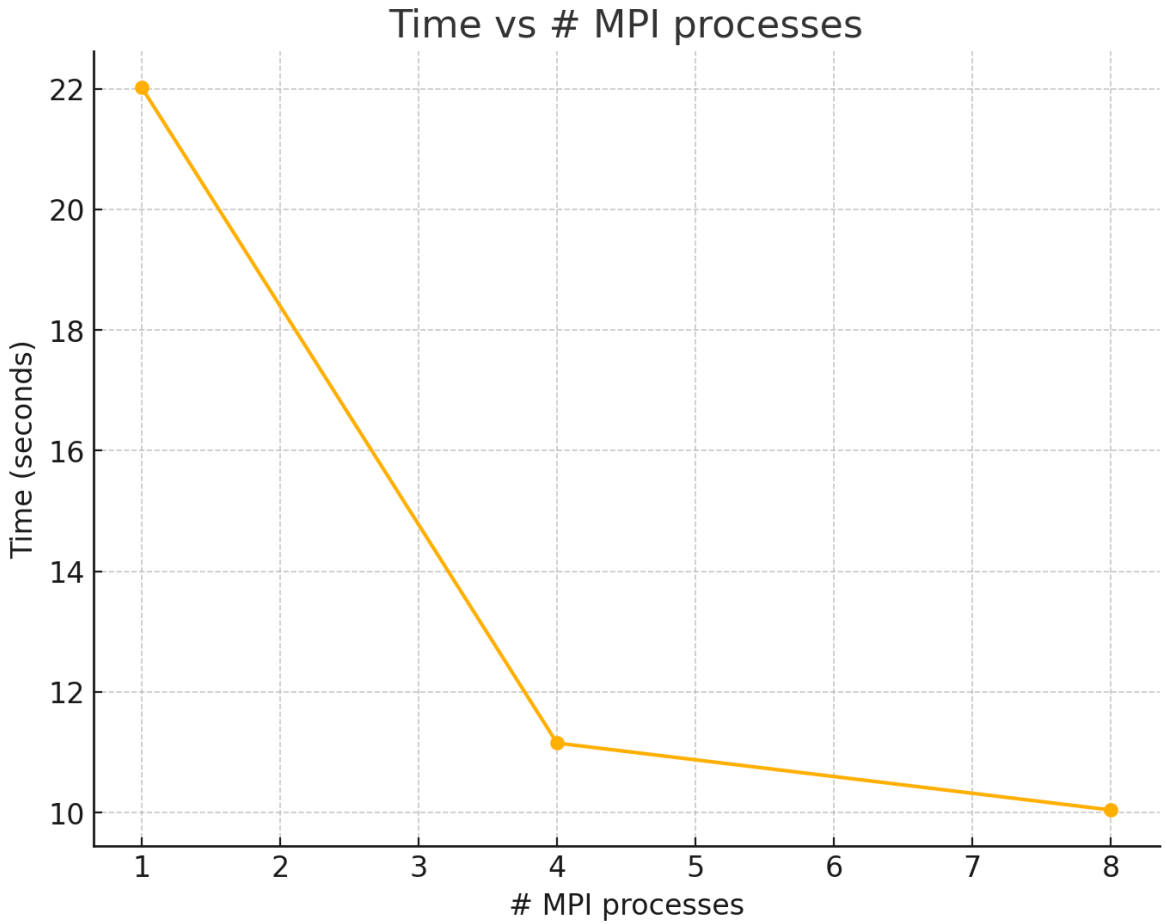
- Starting point: factorization of the SIMPLE preconditioner
- $\hat{F}^{-1}$: approximation of the inverse of matrix F,
- $\hat{S}^{-1}$: approximation of the inverse of the Schur complement $\tilde{S}$

Results

Method	Time (seconds)	Total number of iterations
SIMPLE	4.790718	170
aSIMPLE	4.040719	139
YOSIDA	10.790526	201

- aSIMPLE shows better performances in terms of iterations and time
- SIMPLE has similar performances with respect to aSIMPLE,
- both outperform the Yosida preconditioner in the number of iterations and in the execution time.

Parallelization I



Parallelization II