Available online at www.sciencedirect.com
SCIENCE DIRECT®
Comput. Methods Appl. Mech. Engrg. 195 (2006) 5857–5876
Computer methods in applied mechanics and engineering
www.elsevier.com/locate/cma

Subgrid stabilized projection method for 2D unsteady flows at high Reynolds numbers

J.-L. Guermond [a,1], A. Marra [b], L. Quartapelle [b,*]

[a] Department of Mathematics, Texas A&M University, College Station, TX 77843-3368, USA
[b] Dipartimento di Ingegneria Aerospaziale, Politecnico di Milano, via La Masa 34, 20156 Milano, Italy

Received 9 November 2004; received in revised form 28 May 2005; accepted 16 August 2005

Abstract

A subgrid stabilization technique is developed for solving the two-dimensional incompressible Navier–Stokes equations at high Reynolds numbers. The time marching algorithm is based on a well-established fractional-step pressure-correction projection method. The advection–diffusion step is enriched by an implicit subgrid stabilizing term and by an explicit dissipative shock capturing term. The former is calculated by means of a hierarchical finite element setting, the latter is included to avoid Gibbs' phenomenon in the boundary layer. Convergence tests on prototypical two-dimensional examples are reported and the method is used to simulate the viscous incompressible flows around the airfoil NACA0012 at zero incidence and Reynolds numbers ranging from 10^5 to 10^6.

Keywords: Subgrid stabilization; Projection methods; Incompressibility; Navier–Stokes equations; Finite elements; 2D flows at high Reynolds numbers

1. Introduction

The capability of finite element-based Galerkin methods to solve the incompressible Navier–Stokes equations in a wide range of Reynolds numbers (*Re*) is limited by the capability of current computers to handle extremely large sets of degrees of freedom. From the mathematical point of view, the reason for this difficulty is that the stability of Galerkin methods solely relies on the coercivity of the second-order dissipative operator in the momentum equation. As *Re* grows, the coercivity constant goes to zero and the problem becomes dominated by the non-coercive first-order non-linear transport operator. Accuracy can be recovered if the cell-Reynolds number Vh/ν is of order one, where h is the typical meshsize and V is the typical velocity. Working with cell-Reynolds numbers larger than some units usually yields unrealistic spurious wiggles spreading throughout the entire computational domain.

The currently known alternatives to pure Galerkin methods are either to change the model (i.e., e.g., inviscid potential flow theory plus boundary layer corrections) or to use so-called stabilized methods. One of the most popular class of stabilized method is the Galerkin/least squares method (GaLS) and its many variants (streamline diffusion, streamline upwind Petrov Galerkin, residual free bubbles) [1,2,13,15]. Another set of stabilization techniques are the so-called discontinuous Galerkin methods [5,19,22]. GaLS, GaLS-like and DG methods are quite efficient for solving first-order PDEs and they all yield the same quasi-optimal $\mathcal{O}(h^{k+\frac{1}{2}})$ convergence rate in the L^2-norm on transport equations. However, these methods (except RFB) require tuning coefficients measuring the relative importance of first-order terms to second-order terms when (possibly small) second-order differential operators are present in the equation. Moreover, these methods do not easily

* Corresponding author.
E-mail addresses: guermond@limsi.fr (J.-L. Guermond), marra@aero.polimi.it (A. Marra), quartapelle@aero.polimi.it (L. Quartapelle).
[1] On long leave from LIMSI (CNRS-UPR 3251), BP 133, 91403 Orsay, France.

0045-7825/$ - see front matter
doi:10.1016/j.cma.2005.08.016

extend to time-dependent problems unless using non-trivial discontinuous Galerkin approximation in time. These reasons have led us to prefer the subgrid stabilization technique which does not suffer from the two problems mentioned above, [7–9], and the objective of this paper is to show how the subgrid stabilization technique can be applied in dimension two to simulate the Navier–Stokes equations over a wide range of Reynolds number.

The paper is organized as follows. The basic concepts of the subgrid stabilization method are recalled in Section 2. Application of the stabilization technique to two-dimensional advection equations and convergence tests are reported in Section 3. It is shown in Section 4 how the subgrid viscosity method can be used to solve the Navier–Stokes equations. Implementation details are given and convergence tests are reported. In Section 5 we show numerical solutions of flows around airfoil NACA0012 for $10^5 \leqslant Re \leqslant 10^6$. We explore the transition of the two-dimensional flow beyond the laminar regime.

2. Basic concepts of the subgrid stabilization method

The mathematical theory that yields to the subgrid stabilization method employed in this work has been already presented in detail in a number of papers [8,9]. Therefore the objective of this section is simply to give a brief summary of its most important features.

2.1. Abstract setting

Let us introduce a general setting for non-coercive time-dependent problems. Let $V \subset L$ be two separable Hilbert spaces (V being dense and continuously embedded in L). Let $A : V \to L$ be a positive maximal linear operator. The maximality hypothesis here means that there are $c_1 > 0$ and $c_2 \geqslant 0$ such that

$$\forall u \in V, \quad \sup_{v \in L} \frac{(Au, v)_L}{\|v\|_L} \geqslant c_1 \|u\|_V - c_2 \|u\|_L. \tag{2.1}$$

For applications one may think of L as being $L^2(\Omega)$ and A as being a first-order differential operator, for instance A can be a transport operator. Consider the following problem: For $f \in \mathscr{C}^1([0,T];L)$ and $u_0 \in V$, seek $u \in \mathscr{C}^1([0,T];L) \cap \mathscr{C}^0([0,T];V)$ such that

$$d_t u + Au = f, \quad u(0) = u_0, \tag{2.2}$$

which is well known to be well posed owing to the Hille–Yosida theorem [28, p. 248]. Whereas the Galerkin method is known for not being suitable to solve this problem when A is a first-order differential operator, the subgrid viscosity technique has been tailored for this purpose.

2.2. Abstract discrete setting

The key ideas of the subgrid stabilization method are twofold. First, the approximation space is split into resolved scales and subgrid scales so that a discrete counterpart of the inf–sup condition (2.1) holds uniformly with respect to this decomposition. Second, the Galerkin approximation is slightly modified by adding an asymptotically consistent artificial diffusion term on the subgrid scales.

Let us briefly restate the setting introduced in [8,9]. We assume that there is a family of pairs of finite-dimensional subspaces $\{X_H, X_h\}_{\{H,h\}}$ such that

(i) $X_H \subsetneq X_h \subset V$.

(ii) There is $X_h^H \subset X_h$ such that $X_h = X_H \oplus X_h^H$ and the projection $P_H : X_h \to X_H$ induced by this decomposition is stable in L:

$$\exists c > 0 \quad \forall v_h \in X_h, \quad \|P_H v_h\|_L \leqslant c \|v_h\|_L. \tag{2.3}$$

For all v_h in X_h, set $v_H = P_H v_h$ and $v_h^H = (1 - P_H) v_h$.

(iii) There is $c_i > 0$ such that

$$\forall v_h \in X_h, \quad |v_h|_V \leqslant c_i H^{-1} \|v_h\|_L. \tag{2.4}$$

This hypothesis holds for finite elements based on quasi-uniform mesh families provided $c_1 H \leqslant h \leqslant c_2 H$.

(iv) There exist a dense subspace of V, say W, a linear interpolation operator $I_H \in \mathscr{L}(W; X_H)$, an integer k, and a constant $c > 0$ such that

$$\forall v \in W, \quad \|v - I_H v\|_L + H \|v - I_H v\|_V \leqslant c H^{k+1} \|v\|_W. \tag{2.5}$$

(v) There is a subgrid viscosity bilinear form $b_h \in \mathscr{L}(X_h^H \times X_h^H; \mathbb{R})$ satisfying the following continuity and coercivity hypotheses:

$$\forall v_h^H, w_h^H \in X_h^H, \quad \begin{cases} b_h(v_h^H, w_h^H) \leqslant c_{b2} H |v_h^H|_b |w_h^H|_b, \\ b_h(v_h^H, v_h^H) \geqslant c_{b1} H |v_h^H|_b^2, \end{cases} \tag{2.6}$$

where the seminorm $|\cdot|_b$ is such that there are $c_{e1} > 0$ and $c_{e2} > 0$ for which

$$\forall v_h^H \in X_h^H, \quad c_{e1} |v_h^H|_V \leqslant |v_h^H|_b \leqslant c_{e2} H^{-1} \|v_h^H\|_L. \tag{2.7}$$

(vi) There are $c_a > 0$ and $c_\delta \geqslant 0$ such that

$$\forall v_h \in X_h, \quad \sup_{\phi_h \in X_h} \frac{(A v_H, \phi_h)}{\|\phi_h\|_L} \geqslant c_a |v_H|_V - c_\delta \|v_h\|_L. \tag{2.8}$$

All the constants involved in (2.3)–(2.8) do not depend on $\{H, h\}$.

Hypothesis (2.8) is the keystone of the theory. It is simply the discrete counterpart of (2.1). In general, (2.1) has no uniform discrete counterpart in the standard Galerkin framework. In actual applications, we will choose X_H to be a finite element space defined on a mesh of size H, whereas X_h will be defined on the once-refined mesh, i.e., $h = H/2$.

Set $a_h(u_h, v_h) = (A u_h, v_h) + b(u_h^H, v_h^H)$ and also introduce $a_s(u, v) = \frac{1}{2}[(Au, v) + (Av, u)]$. Since A is positive, a_s is positive and symmetric. As a result, a_s induces a seminorm on V; let $|\cdot|_{V_s}$ be this seminorm. Assume that $u_0 \in W$ so that u_0 can be approximated by $I_H u_0$. The discrete problem we consider hereafter is the following: Seek $u_h \in \mathscr{C}^1([0, T]; X_h)$ such that

$$(d_t u_h, v_h)_L + a_h(u_h, v_h) = (f, v_h)_L \quad \forall v_h \in X_h, \quad u_h(0) = I_H u_0. \tag{2.9}$$

This problem has clearly a unique solution since it is a linear system of ODEs. The major result proved in [9] is the following:

Theorem 2.1. *Under hypotheses (i)–(vi), if the solution u to (2.2) is in $\mathscr{C}^2([0, T]; W)$, the solution u_h to (2.9) satisfies*

$$\|u - u_h\|_{\mathscr{C}^0([0,T];L)} + |u - u_h|_{L^2(]0,T[;V_s)} \leqslant c_1 H^{k+\frac{1}{2}}, \tag{2.10}$$

$$\frac{1}{\sqrt{T}} \|u - u_h\|_{L^2(]0,T[;V)} \leqslant c_2 H^k, \tag{2.11}$$

where $c_1 = c[H + T(1 + T)]^{\frac{1}{2}} \|u\|_{\mathscr{C}^2([0,T];W)}$, and $c_2 \leqslant c[1 + T]\|u\|_{\mathscr{C}^2([0,T];W)}$.

Note that in the above error estimates H can be replaced by h, since owing to the inverse inequality hypothesis (iii) h and H must be of the same order.

2.3. A singular perturbation result

The subgrid viscosity technique is tailored for first-order differential operators. In practice, we have often to deal with situations where $B = A + \epsilon D$, where A is a first-order differential operator and D is a coercive second-order differential operator. From the mathematical point of view, the coercivity of D implies that the evolution equation is parabolic. If ϵ is $\mathscr{O}(1)$, the standard Galerkin theory applies. If ϵ is small, the coercivity is not strong enough to guarantee that the Galerkin approximation is satisfactory, since, in first approximation, one has $B \approx A$. We now show that the subgrid viscosity technique can easily be extended to treat this situation.

Use the notation introduced above. In addition to the two Hilbert spaces already considered, L and V, we introduce a new Hilbert space, say X, with dense and continuous embedding in V. We also introduce a bilinear form $d \in \mathscr{L}(X \times X; \mathbb{R})$, and we assume that there is a seminorm $|\cdot|_X$ in X so that $d(u, v) \leqslant c_d |u|_X |v|_X$ for all u, v in X. In practice, D can be a degenerate elliptic operator. We assume the following coercivity property with respect to the seminorm $|\cdot|_X$:

$$\forall v \in X, \quad |v|_X^2 \leqslant (Av, v) + d(v, v). \tag{2.12}$$

Consider the following problem: For $f \in C^1([0, T]; L)$, $\epsilon \geqslant 0$, and $u_0 \in X$, seek u in $\{v \in L^2(]0, T[; X);\ d_t v \in L^2(]0, T[; X')\}$ such that for all $v \in X$ and for all $t \geqslant 0$

$$(d_t u, v)_L + (Au, v) + \epsilon d(u, v) = (f, v)_L, \quad u(0) = I_H u_0. \tag{2.13}$$

Assume that (2.13) is normalized so that $\epsilon \leqslant 1$ and that there is $c > 0$ so that $\|v\|_X \leqslant c(\|v\|_L + |v|_X)$. The consequence of this hypothesis is that problem (2.13) is parabolic and has a unique solution. In the framework of unsteady advection–diffusion equations, ϵ is the reciprocal of the Péclet number.

To illustrate the above setting, let us think for instance of an advection diffusion equation with homogeneous Dirichlet boundary conditions. Let ϵ be the viscosity coefficient and β be the advection field, and let Γ^- be the inflow boundary. Then $L = L^2(\Omega)$, $V = \{v \in L^2(\Omega); \beta \cdot \nabla v \in L^2(\Omega); v|_{\Gamma^-} = 0\}$, and $X = H_0^1(\Omega)$. Moreover, $Av = \beta \cdot \nabla v$, $Dv = -\Delta v$, $a(u,v) = \int_\Omega v(\beta \cdot \nabla u)$, and $d(u,v) = \int_\Omega \nabla u \cdot \nabla v$.

We now turn our attention to the approximation of problem (2.13). We use the two-scale discrete setting already introduced above to construct an approximate solution to (2.13). Introduce a sequence of pairs of finite-dimensional spaces $\{X_H, X_h\}_{\{H,h\}}$, both conforming in X and satisfying hypotheses (i)–(vi) stated in Section 2.2. Furthermore, assume that the following inverse inequality holds:

$$|v_h|_X \leqslant cH^{-1}\|v_h\|_L. \tag{2.14}$$

In practice, hypothesis (2.14) means that the norms of X and V involve derivatives of the same order (see the advection diffusion example considered above).

Assume $u_0 \in W$ so that $I_H u_0$ is an optimal approximation to u_0. The discrete problem is: Seek u_h in $C^1([0,T]; X_h)$ such that, $u_h(0) = I_H u_0$ and for all $t \geqslant 0$ and $v_h \in X_h$

$$(d_t u_h, v_h)_L + (Au_h, v_h) + \epsilon d(u_h, v_h) + b_h(u_h^H, v_h^H) = (f, v_h)_L. \tag{2.15}$$

Problem (2.15) is well posed since it is a linear system of ODEs. The following result is proved in [9]:

Theorem 2.2. *In the framework of the above assumptions, if the solution u to (2.13) is in $\mathscr{C}^2([0,T]; W)$, then the solution u_h to (2.15) satisfies*

$$\|u - u_h\|_{\mathscr{C}^0([0,T];L)} + \epsilon^{\frac{1}{2}}\|u - u_h\|_{L^2([0,T];X)} \leqslant c_1(T,u)\left(H^{k+\frac{1}{2}} + \epsilon^{\frac{1}{2}}H^k\right), \tag{2.16}$$

$$\frac{1}{\sqrt{T}}\|u - u_h\|_{L^2([0,T];V)} \leqslant c_2(T,u)H^k, \tag{2.17}$$

where $c_1 = c[H + T(1+T)]^{\frac{1}{2}}\|u\|_{\mathscr{C}^2([0,T];W)}$, and $c_2 = c[1+T]\|u\|_{\mathscr{C}^2([0,T];W)}$.

Note that contrary to GaLS-like methods or DG techniques nothing in the setting of the subgrid viscosity method has to be tuned to account for the presence of the dissipative bilinear form $\epsilon d(u,v)$. In particular, the subgrid viscosity constant c_b (see (2.24)) does not depend on ϵ. Furthermore any reasonable type of time-stepping technique can be used to approximate the time derivative (for instance any A-stable method is suitable).

The subgrid stabilization technique described above shares some features with the heuristic setting of the so-called variational multiscale method [12]. One may think of the present subgrid method as one particular mathematical formalization of the variational multiscale concept.

2.4. Particular instances of two-level settings

We now present two realizations of the two-level Lagrange finite elements setting satisfying the hypotheses (2.3)–(2.8) when A is a first-order differential operator (possibly with reasonably smooth non-constant coefficients, see [8] for details).

Since our goal is to simulate two-dimensional flows using triangular meshes, we assume that the solution domain Ω is a polygon of $\mathbb{R}^2$. We denote by $(\mathscr{T}_H)_{\{H>0\}}$ a shape-regular family of triangulations of Ω. All that is said below can be extended to dimension three provided the computer resource is available.

2.4.1. Two-level $\mathbb{P}_1$ interpolation

Let X_H be the function space of the resolved scales defined by

$$X_H = \left\{\Phi \in \mathscr{C}^0(\overline{\Omega}), \Phi|_{K_H} \in \mathbb{P}_1(K_H)\ \forall K_H \in \mathscr{T}_H\right\}. \tag{2.18}$$

For each macro-triangle $K_H \in \mathscr{T}_H$, we create four new triangles by connecting the midpoints of the three sides of K_H. Set $h = H/2$ and denote by $\mathscr{T}_h$ the resulting triangulation. For each macro-triangle K_H, we define $\mathbb{P}_{(1)}(K_H)$ to be the space of functions that are continuous on K_H, are piecewise $\mathbb{P}_1$ on each sub-triangle of K_H, and vanish at the three vertices of K_H. We define the subgrid scale space to be

$$X_h^H = \left\{\varphi \in \mathscr{C}^0(\overline{\Omega}), \varphi|_{K_H} \in \mathbb{P}_{(1)}\ \forall K_H \in \mathscr{T}_H\right\}. \tag{2.19}$$

Letting $X_h = X_H \oplus X_h^H$, it is clear that we can characterize X_h by

$$X_h = \left\{\phi \in \mathscr{C}^0(\overline{\Omega}), \phi|_{K_h} \in \mathbb{P}_1(K_h)\ \forall K_h \in \mathscr{T}_h\right\}. \tag{2.20}$$

Fig. 1 shows the two-level spatial discretization and the degrees of freedom for X_H, X_h and X_h^H.

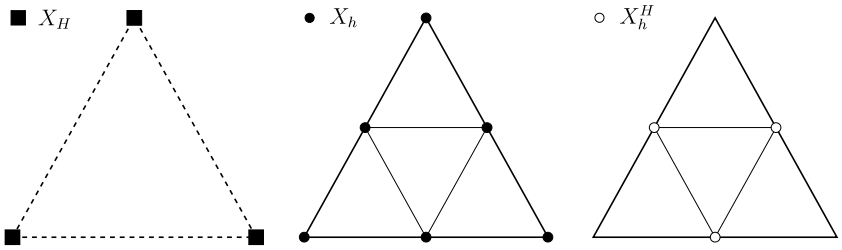

Fig. 1. Two-level $\mathbb{P}_1$ setting. Left: $\mathbb{P}_1$ resolved scale space X_H (■, dof); Center: four $\mathbb{P}_1$ triangles of X_h generated from the macro-triangle (●, dof). Right: subgrid scale space X_h^H (○, dof).

2.4.2. Two-level $\mathbb{P}_2$ interpolation

The construction of hierarchical bases for quadratic polynomials mimics that for the linear case. For the sake of completeness, we repeat the construction of the spaces. Since a $\mathbb{P}_2$ subparametric interpolation is employed for the velocity in the Navier–Stokes problem, the sides of the triangles of $\mathscr{T}_h$ are assumed to be straight.

Again we set $h = H/2$ and we denote by $\mathscr{T}_h$ the triangulation obtained by dividing each macro-triangle of $\mathscr{T}_H$ into four equal triangles (Fig. 2). For each triangle $K_h \in \mathscr{T}_h$ we denote by φ_1, φ_2, φ_3 the three nodal $\mathbb{P}_2$ shape functions associated with the midpoints of the three sides of K_h. We set the resolved scale space to be

$$X_H = \left\{ \Phi \in \mathscr{C}^0(\overline{\Omega}), \Phi|_{K_H} \in \mathbb{P}_2(K_H) \ \forall K_H \in \mathscr{T}_H \right\} \tag{2.21}$$

and we define the space of the subgrid scale to be

$$X_h^H = \left\{ \varphi \in \mathscr{C}^0(\overline{\Omega}), \varphi|_{K_h} \in \operatorname{span}(\varphi_1, \varphi_2, \varphi_3) \ \forall K_h \in \mathscr{T}_h \right\}. \tag{2.22}$$

The space $X_h = X_H \oplus X_h^H$ is then characterized by

$$X_h = \left\{ \phi \in \mathscr{C}^0(\overline{\Omega}), \phi|_{K_h} \in \mathbb{P}_2(K_h) \ \forall K_h \in \mathscr{T}_h \right\}. \tag{2.23}$$

2.4.3. The subgrid viscosity bilinear form b_h

Assuming that one of the two realizations of the two-level Lagrange finite element setting described above is chosen, we now define the bilinear form b_h introduced in the hypothesis (v) as follows:

$$b_h(u_h^H, v_h^H) = c_b \sum_{K \in \mathscr{T}_h} \sqrt{|K|} \int_K \nabla u_h^H \cdot \nabla v_h^H, \tag{2.24}$$

where $|K|$ is the measure of K, $u_h^H = (1 - P_H)u_h$ and $v_h^H = (1 - P_H)v_h$. A three-dimensional rendering of the graph of nodal shape functions associated with a vertex of the coarse mesh $\mathscr{T}_H$ is shown in Fig. 3. In the figure we show the fine scale v_h, the resolved scale $v_H = P_H v_h$, and the fluctuation $v_h^H = (1 - P_H)v_h$ for the two-level $\mathbb{P}_1$ setting (left) and the two-level $\mathbb{P}_2$ setting (right). c_b is a free constant; henceforth we set $c_b = 0.015$, unless specified otherwise.

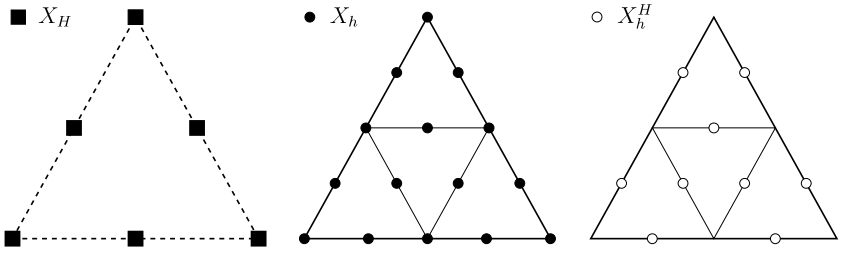

Fig. 2. Space discretization and different set of degrees of freedom for the two-level $\mathbb{P}_2$ interpolation. Left: $\mathbb{P}_2$ macro-triangle of the resolved scale X_H. Center: four $\mathbb{P}_2$ triangles of X_h generated from the macro-triangle. Right: subgrid scale X_h^H whose degrees of freedom are indicated by empty circles.

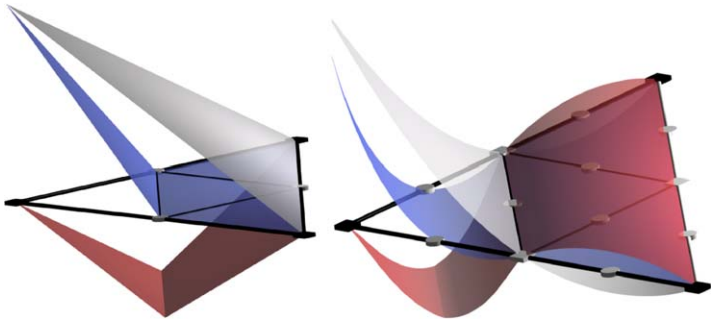

Fig. 3. 3D rendering of the graph of nodal shape functions associated with a vertex of the grid $\mathscr{T}_H$: $v_H \in X_H$ (grey), $v_h \in X_h$ (blue) and $v_h^H \in X_h^H$ (red). Left: two-level $\mathbb{P}_1$. Right: two-level $\mathbb{P}_2$. (For interpretation of colours in this figure legend the reader is referred to the web version of this article.)

3. Subgrid stabilized solution of 2D advection equation

3.1. Smooth initial data

To quantitatively appreciate the behavior of the subgrid stabilization technique, we test it on a two-dimensional advection equation with smooth initial data in a circular domain $\Omega = \left\{(x,y) \in \mathbb{R}^2 | \sqrt{x^2+y^2} < 1\right\}$. The velocity field is a rigid rotation with unit angular velocity; accordingly we set $\boldsymbol{\beta}(\boldsymbol{r}) = 2\pi(-y,x)$. The initial value problem is as follows:

$$\partial_t u + \boldsymbol{\beta}(\boldsymbol{r}) \cdot \nabla u = 0 \quad \text{in } \Omega, \qquad u|_{t=0} = u_0(\boldsymbol{r}) = \exp\left(-\frac{|\boldsymbol{r}-\boldsymbol{r}_0|^2}{a^2}\right), \tag{3.1}$$

where $\boldsymbol{r}_0 = (0.5, 0)$ and $a = 0.2$. Thus, the initial Gaussian hill rotates around the origin without deformation.

The time discretization is based on the second-order accurate three level BDF scheme. The time step used in the numerical tests reported below is either $\Delta t = 10^{-3}$ or $\Delta t = 2.5 \times 10^{-4}$. We have chosen very small time steps to make sure that the time error is significantly smaller than the spatial error so that the only error observed in the tests is that induced by the space discretization. In practice such small time steps are not necessary since the method is unconditionally stable. To simplify notations, for every sequence $(\phi^0, \phi^1, \ldots)$ we set

$$D\phi^{k+1} = \begin{cases} \frac{1}{2}(3\phi^{k+1} - 4\phi^k + \phi^{k-1}) & \text{if } k \geqslant 1, \\ \phi^{k+1} - \phi^k & \text{if } k = 0. \end{cases} \tag{3.2}$$

Using the notation introduced in Sections 2.4.1 and 2.4.2, the fully discrete subgrid stabilized counterpart of problem (3.1) is as follows: Set $u_h^0(\boldsymbol{r}) = I_H u_0(\boldsymbol{r})$, where I_H is the Lagrange interpolation operator, and for $k \geqslant 0$ seek u_h^{k+1} in X_h such that

$$\left(\frac{Du_h^{k+1}}{\Delta t}, v_h\right) + \left(\boldsymbol{\beta}(\boldsymbol{r}) \cdot \nabla u_h^{k+1}, v_h\right) + b_h(u_h^{H,k+1}, v_h^H) = 0, \tag{3.3}$$

where $u_h^{H,k+1} = (1 - P_H)u_h^{k+1}$ is the fluctuating (or subgrid) part of u_h^{k+1}.

3.1.1. Numerical results

The numerical results obtained with ($c_b = 0.015$) and without ($c_b = 0$) subgrid stabilization and using $\Delta t = 10^{-3}$ are reported in Figs. 4 and 5, for $\mathbb{P}_1$ and $\mathbb{P}_2$ approximations, respectively.

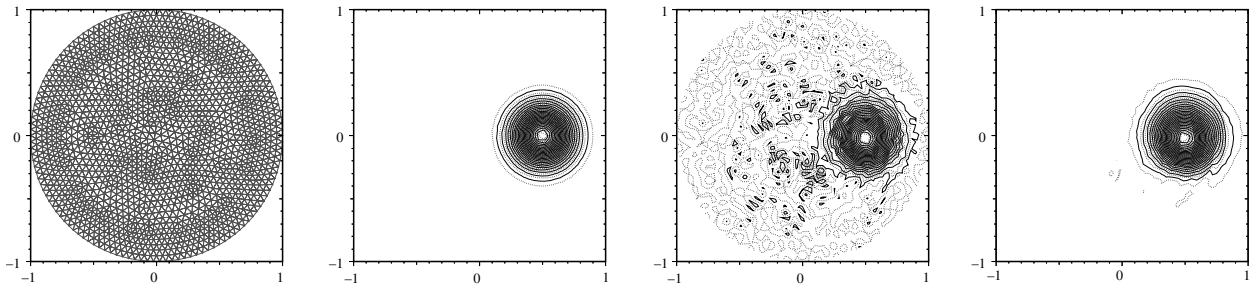

Fig. 4. Gaussian hill: $T = 5$, $\mathbb{P}_1$ approximation, $\Delta t = 10^{-3}$. From left to right: grid ($h = 0.05$); $\mathbb{P}_1$ interpolant of solution; Galerkin solution ($c_b = 0$); subgrid stabilized solution ($c_b = 0.015$).

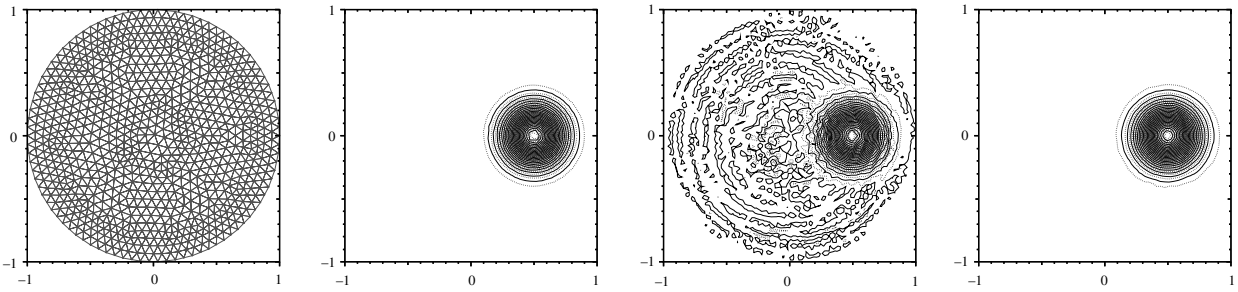

Fig. 5. Gaussian hill: $T = 5$, $\mathbb{P}_2$ approximation, $\Delta t = 10^{-3}$. From left to right: grid ($h = 0.07$); $\mathbb{P}_1$ interpolant of solution; Galerkin solution ($c_b = 0$); subgrid stabilized solution ($c_b = 0.015$).

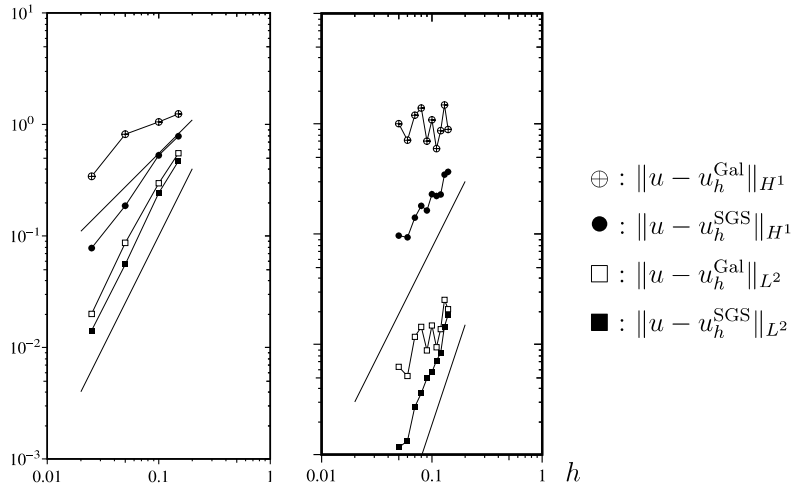

Fig. 6. Spatial convergence tests. Error versus meshsize h for $\mathbb{P}_1$ (left) and $\mathbb{P}_2$ (right) approximations. Empty markers refer to standard Galerkin approximation. Full markers refer to the subgrid viscosity approximation.

The leftmost panel in Fig. 4 shows the grid employed consisting of 3220 triangles ($h = 0.05$). The second panel shows the isovalues of the exact solution at $T = 5$, i.e., after five complete revolutions. The solution obtained by using the Galerkin method is shown in the third panel and is found to be affected by strong numerical wiggles spreading throughout the computational domain. By contrast, the subgrid stabilized solution shown in the rightmost panel is obviously a better approximation.

Similarly, Fig. 5 shows the results obtained by the two-level $\mathbb{P}_2$ approximation at $T = 5$ and using $\Delta t = 10^{-3}$. The grid consisting of 1612 triangles ($h = 0.07$) is shown in the leftmost panel. The Lagrange interpolant of the exact solution is shown in the second panel. The other two panels in the figure demonstrate clearly the improvement achieved by the subgrid stabilization technique over the standard Galerkin method.

Finally, we collect in Fig. 6 the results from convergence tests using $\mathbb{P}_1$ and $\mathbb{P}_2$ approximations, respectively. The curves with full markers correspond to the results obtained using the subgrid stabilization technique. The most accurate results are those from the stabilized $\mathbb{P}_2$ solution. Note that for both the $\mathbb{P}_1$ and the $\mathbb{P}_2$ approximations the L^2-norm of the error is monotone with respect to the meshsize. To materialize the convergence rates, the lines of slope 1, 2 and 3 are also drawn. The results show that the stabilized method based on the $\mathbb{P}_1$ approximation is $\mathcal{O}(h^2)$ in the L^2-norm and $\mathcal{O}(h)$ in the H^1-norm, while the $\mathbb{P}_2$ stabilized method is $\mathcal{O}(h^3)$ in the L^2-norm and $\mathcal{O}(h^2)$ in the H^1-norm. We used $\Delta t = 2.5 \times 10^{-4}$ to avoid saturation of the error on very fine meshes due to the time discretization.

3.2. Discontinuous initial data

We now consider a problem with an initial datum that is not smooth, say discontinuous. Referring to (2.2), we no longer assume $u_0 \in V$. Consequently the quasi-optimal convergence results stated in Theorem 2.1 no longer hold; although, owing to the density of V into L, convergence is still ensured in the L-norm whenever the data is only in L.

A very similar situation arises when computing incompressible flows at high Reynolds numbers. As *Re* grows, boundary layers appear and grow thinner. If the grid that is used to approximate the solution is not fine enough to capture these boundary layers, then, at the meshsize scale the exact solution can be considered to be discontinuous. We want to show

in this section that the subgrid stabilization technique can handle this situation provided an additional shock capturing term is added to the formulation.

Consider the following advection problem:

$$\partial_t u + \boldsymbol{\beta}(\boldsymbol{r}) \cdot \nabla u = 0 \quad \text{in } \Omega, \qquad u|_{t=0} = \begin{cases} 1 & \text{in } \Omega_1, \\ 0 & \text{elsewhere,} \end{cases} \tag{3.4}$$

where $\Omega_1 = \{(x,y) \in \mathbb{R}^2 | 0.4 \leqslant x \leqslant 0.8, |y| \leqslant 0.2\}$ and $\boldsymbol{\beta}(\boldsymbol{r}) = 2\pi(-y,x)$. Since the initial datum is discontinuous, the exact solution is also discontinuous with respect to space at all times. When trying to approximate this function one then observes the well-known Gibbs phenomenon characterized by spurious oscillations. This behavior is the numerical manifestation of a far-reaching theorem of analysis that states that truncated Fourier series of a given function does not converge uniformly to the function in question unless the function is very smooth (continuity is not enough) [23].

A possible solution to avoid Gibbs oscillations is to introduce a strong local dissipation where the solution is not smooth [14,16]. Within our two-level space discretization, an indicator of the lack of regularity of the solution is its fluctuation $u_h^H = (1 - P_H)u_h$. Therefore we introduce the following non-linear shock capturing form:

$$c_h^H(u_h, w_h, v_h) = c_{\text{sc}} \sum_{K_H \in \mathcal{T}_H} \sqrt{|K_H|} \frac{\|\nabla u_h^H\|_{\boldsymbol{L}^2(K_H)}}{\|\nabla u_h\|_{\boldsymbol{L}^2(K_H)}} \int_{K_H} \nabla w_h \cdot \nabla v_h. \tag{3.5}$$

This form is linear with respect to its second and third arguments and non-linear with respect to u_h. The need for a non-linear form is rooted in the Godunov theorem [20] which states that a linear monotone scheme cannot be more accurate than $\mathcal{O}(h)$.

The approximate counterpart of (3.4) is obtained in weak form by adding to the Galerkin formulation the (linear) subgrid stabilization form b_h and the (non-linear) shock capturing form c_h. The discrete problem is the following: Set $u_h^0(\boldsymbol{r}) = I_H u_0(\boldsymbol{r})$ and for $k \geqslant 0$ seek u_h^{k+1} in X_h such that for all $v_h \in X_h$

$$\left(\frac{Du_h^{k+1}}{\Delta t}, v_h\right) + \left(\boldsymbol{\beta}(\boldsymbol{r}) \cdot \nabla u_h^{k+1}, v_h\right) + b_h(u_h^{k+1,H}, v_h^H) + c_h^H(u_h^k, u_h^k, v_h) = 0. \tag{3.6}$$

3.2.1. Numerical results

The computations are performed on the grid shown in Fig. 4: it consists of 3220 triangles ($h = 0.05$) and 1674 $\mathbb{P}_1$ degrees of freedom. The results for $T = 1$ and $\Delta t = 10^{-3}$ are shown in Fig. 7. By considering the $\mathbb{P}_1$-interpolant of the exact solution shown in the leftmost panel, it is clear that due to the lack of regularity, the Galerkin solution after just one period (second panel) is more affected by spurious oscillations than the Gaussian hill after five turns (Fig. 4). The linear subgrid stabilization alone (Fig. 7, third panel) eliminates most of them but cannot get rid of Gibbs' oscillations. By employing both the subgrid stabilization and the shock capturing term the spatial oscillations disappear (rightmost panel) at the price of increased smearing of the discontinuity.

Fig. 8 shows the graph of the Galerkin solution and of the two stabilized solutions to put in evidence the large oscillations close to the prism and their removal by using the non-linear shock capturing term.

The results obtained by means of the quadratic approximation are presented in Figs. 9 (isovalues) and 10 (three-dimensional rendering of the graph). The grid consisting of 1612 triangles ($h = 0.07$) and 3315 $\mathbb{P}_2$ degrees of freedom is shown in Fig. 5.

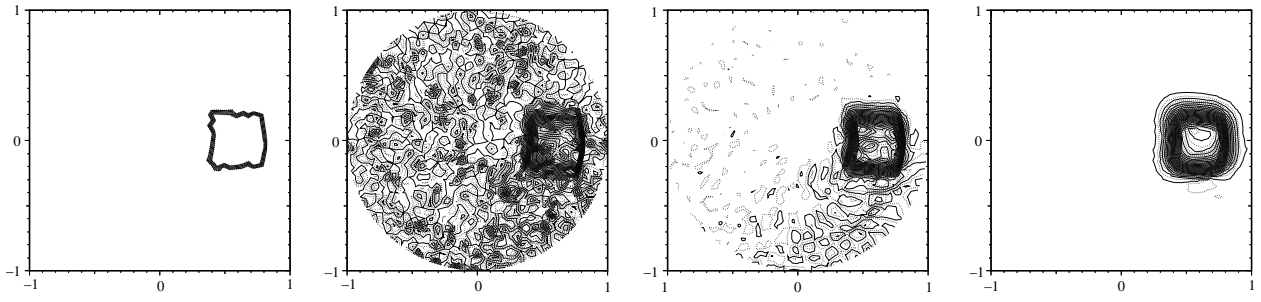

Fig. 7. Prism: $T = 1$, $\mathbb{P}_1$ approximation, $\Delta t = 10^{-3}$. From left to right: $\mathbb{P}_1$ interpolant of solution; Galerkin solution ($c_b = 0$); subgrid stabilized solution ($c_b = 0.015$); shock capturing + subgrid stabilized solution ($c_b = 0.015$, $c_{\text{sc}} = 0.09$).

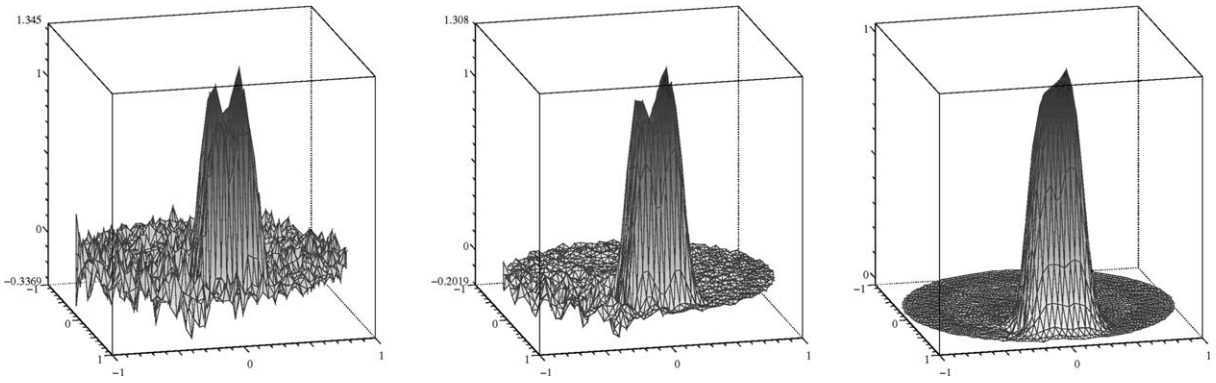

Fig. 8. Prism: $T = 1$, $\mathbb{P}_1$ approximation, $\Delta t = 10^{-3}$. From left to right: Galerkin solution ($c_b = 0$); subgrid stabilized solution ($c_b = 0.015$); shock capturing + subgrid stabilized solution ($c_b = 0.015$, $c_{sc} = 0.09$).

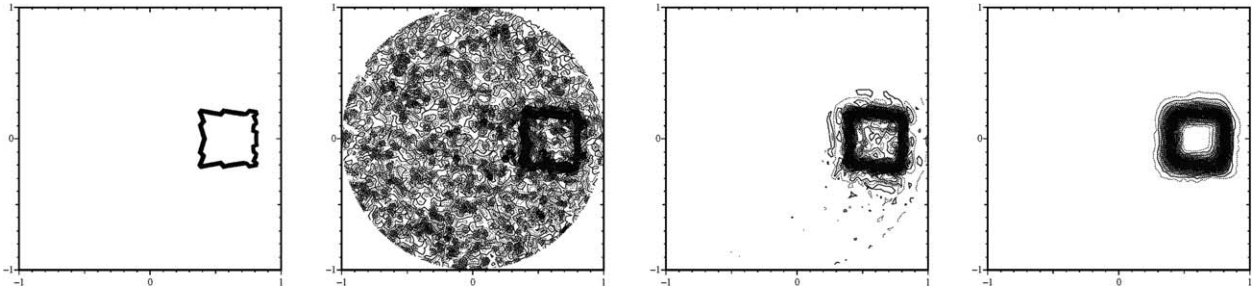

Fig. 9. Prism: $T = 1$, $\mathbb{P}_2$ approximation, $\Delta t = 10^{-3}$. From left to right: $\mathbb{P}_2$ interpolant of solution; Galerkin solution ($c_b = 0$); subgrid stabilized solution ($c_b = 0.015$); shock capturing + subgrid stabilized solution ($c_b = 0.015$, $c_{sc} = 0.06$).

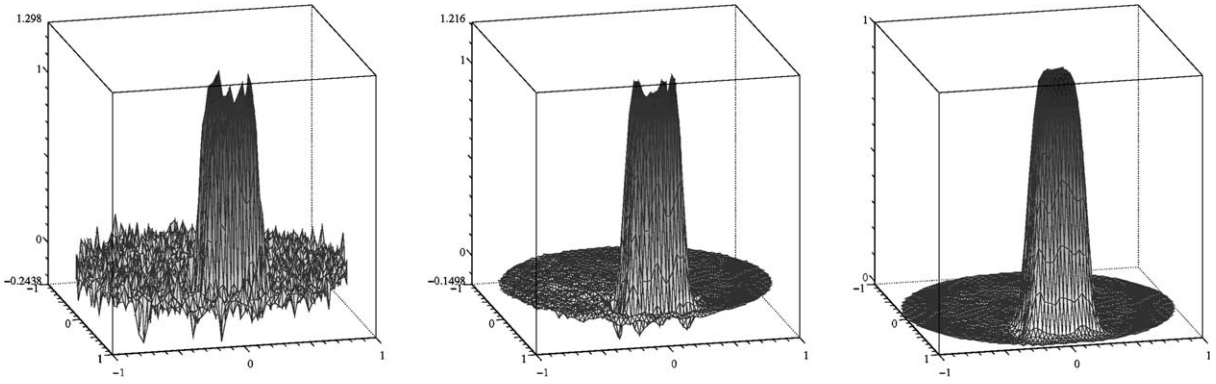

Fig. 10. Prism: $T = 1$, $\mathbb{P}_2$ approximation, $\Delta t = 10^{-3}$. Left: Galerkin solution ($c_b = 0$). Center: subgrid stabilized solution ($c_b = 0.015$). Right: shock capturing + subgrid stabilized solution ($c_b = 0.015$, $c_{sc} = 0.06$).

4. Subgrid stabilized incremental projection method

In this section we introduce the time-dependent incompressible Navier–Stokes equations and we describe our solution technique. The algorithm is based on a pressure-correction method combined with the subgrid technique introduced above. Convergence tests for the stabilized projection method are reported at the end of the section.

4.1. Time-dependent incompressible Navier–Stokes equations

We consider the motion of an incompressible viscous fluid occupying a domain Ω assumed to be a two-dimensional bounded region. One (or more) airfoil is immersed in the fluid. The set of equations modeling this situation is the incompressible Navier–Stokes equations expressed in terms of the primitive variables, namely the velocity $\boldsymbol{u}$ and the pressure p:

$$\begin{cases} \partial_t \boldsymbol{u} + (\boldsymbol{u}\cdot\boldsymbol{\nabla})\boldsymbol{u} = -\nabla p + \nu\nabla^2\boldsymbol{u}, \quad \nabla\cdot\boldsymbol{u} = 0, \\ \boldsymbol{u}_{|t=0} = \boldsymbol{u}_0(\boldsymbol{r}), \quad \boldsymbol{u}_{|\Gamma} = \boldsymbol{b}(\boldsymbol{r}_\Gamma, t), \end{cases} \tag{4.1}$$

$\boldsymbol{u}_0(\boldsymbol{r})$ is the initial velocity field and $\boldsymbol{b}(\boldsymbol{r}_\Gamma, t)$ is the velocity prescribed at the boundary Γ of Ω. The initial and boundary data, assumed to be smooth enough for the purpose of our subsequent analysis, must satisfy the following set of compatibility conditions:

$$\nabla\cdot\boldsymbol{u}_0(\boldsymbol{r}) = 0, \quad \int_\Gamma \boldsymbol{n}\cdot\boldsymbol{b}(\boldsymbol{r}_\Gamma, t) = 0, \quad \boldsymbol{n}\cdot\boldsymbol{u}_0(\boldsymbol{r})|_\Gamma = \boldsymbol{n}\cdot\boldsymbol{b}(\boldsymbol{r}_\Gamma, 0). \tag{4.2}$$

These three compatibility conditions guarantee existence and uniqueness of weak solutions of minimal regularity ($\boldsymbol{u} \in L^2(0,T;H^1)\cap L^\infty(0,T;\boldsymbol{L}^2)$), see e.g. [26, p. 253] or [18, p. 88].

4.2. Second-order BDF incremental projection method

A major difficulty for the numerical simulation of incompressible flows is that the velocity and the pressure are coupled by the incompressibility constraint. Projection methods are fractional-step time-marching algorithm that overcome this difficulty [3,4,25,26]. The most attractive feature of projection methods is that, at each time step, one only needs to solve a sequence of decoupled elliptic equations for the velocity and the pressure, making it very efficient for large scale numerical simulations. The method implemented here is based on the incremental projection method of Guermond and Quartapelle [10]. Since a large body of literature has already been dedicated to the development and the analysis of methods belonging to this class, we are just going to show how the subgrid viscosity method can be implemented using the so-called rotational form of the pressure-correction algorithm [27,11].

We approximate the time derivative in (4.1) by using a second-order BDF2 time-stepping scheme and we adopt the rotational form of the pressure-correction algorithm. Let Δt be the time step and set $t_k = k\Delta t$ for $k \in \{0,1,\ldots,[T/\Delta t]\}$. Let $\boldsymbol{u}_\star^{k+1}$ be the second-order linear extrapolation of the advection velocity at time t_{k+1} (for $k \geqslant 1$) defined as follows:

$$\boldsymbol{u}_\star^{k+1} = 2\boldsymbol{u}^k - \boldsymbol{u}^{k-1}. \tag{4.3}$$

If $k = 0$ we set $\boldsymbol{u}_\star^1 = \boldsymbol{u}^0$. The non-linear advection term is put in skew-symmetric form $(\boldsymbol{u}\cdot\boldsymbol{\nabla})\boldsymbol{u} + \frac{1}{2}(\nabla\cdot\boldsymbol{u})\boldsymbol{u}$ (see e.g., [26]) and is semi-implicitly discretized in time. To avoid long notation we introduce

$$\mathrm{NL}(\boldsymbol{u}_\star, \boldsymbol{u}) = (\boldsymbol{u}_\star\cdot\boldsymbol{\nabla})\boldsymbol{u} + \frac{1}{2}(\nabla\cdot\boldsymbol{u}_\star)\boldsymbol{u}. \tag{4.4}$$

We initialize the algorithm by setting $\boldsymbol{u}^0 = \boldsymbol{u}_0$, $p_0 = p(t=0)$, and $\phi^0 = 0$. Then for $k \geqslant 0$ we set

$$p_\star^{k+1} = \begin{cases} p^0 & \text{for } k = 0, \\ p^1 + 2\phi^1 & \text{for } k = 1, \\ p^2 + \frac{4}{3}\phi^2 - \frac{1}{2}\phi^1 & \text{for } k = 2, \\ p^k + \frac{4}{3}\phi^k - \frac{1}{3}\phi^{k-1} & \text{for } k \geqslant 3, \end{cases} \tag{4.5}$$

where ϕ^k, $k = 1,2,\ldots$, is the solution of a Poisson equation that will be defined below. Note that the initial pressure field p_0 is not an initial data but can be computed by solving $-\nabla^2 p^0 = \nabla\cdot[(\boldsymbol{u}_0\cdot\boldsymbol{\nabla})\boldsymbol{u}_0]$ with $\partial_n p^0|_\Gamma = \boldsymbol{n}\cdot[-(\boldsymbol{u}_0\cdot\boldsymbol{\nabla})\boldsymbol{u}_0 - \nu\nabla\times\nabla\times\boldsymbol{u}_0]|_\Gamma$. In the present case, we assumed $\boldsymbol{u}_0$ to be the potential flow field $\boldsymbol{u}_0 = \boldsymbol{\nabla}\psi_0$ with $\nabla^2\psi_0 = 0$, $\partial_n\psi_0|_\Gamma = \boldsymbol{n}\cdot\boldsymbol{b}(\boldsymbol{r}_\Gamma, 0)$, so that $\boldsymbol{\nabla}\times\boldsymbol{u}_0 = 0$ and the boundary condition for p^0 simplifies to $\partial_n p^0|_\Gamma = -\boldsymbol{n}\cdot(\boldsymbol{u}_0\cdot\boldsymbol{\nabla})\boldsymbol{u}_0|_\Gamma$. Then using the identity $(\boldsymbol{u}_0\cdot\boldsymbol{\nabla})\boldsymbol{u}_0 = \boldsymbol{\nabla}(\frac{1}{2}\|\boldsymbol{u}_0\|^2)$ since $\nabla\times\boldsymbol{u}_0 = 0$, we infer $p_0 = -\frac{1}{2}\|\boldsymbol{u}_0\|^2 + \text{constant}$. See [21] for other details.

The first step of the algorithm accounts for the viscous effects; it consists of seeking $\boldsymbol{u}^{k+1}$ such that

$$\frac{D\boldsymbol{u}^{k+1}}{\Delta t} - \nu\nabla^2\boldsymbol{u}^{k+1} + \mathrm{NL}(\boldsymbol{u}_\star^{k+1}, \boldsymbol{u}^{k+1}) = -\nabla p_\star^{k+1}, \quad \boldsymbol{u}^{k+1}|_\Gamma = \boldsymbol{b}^{k+1}. \tag{4.6}$$

Note that the Cartesian components of the unknown $\boldsymbol{u}^{k+1}$ are fully uncoupled. The second step of the algorithm accounts for the incompressibility constraint; it is the so-called projection step. It is written in the form of a Poisson equation supplemented with Neumann boundary conditions:

$$-\nabla^2\phi^{k+1} = -\frac{\beta_{k+1}}{\Delta t}\nabla\cdot\boldsymbol{u}^{k+1}, \quad \partial_n\phi^{k+1}|_\Gamma = 0, \tag{4.7}$$

where $\beta_1 = 1$ and $\beta_{k+1} = \frac{3}{2}$ if $k \geqslant 1$. Then we update the pressure field as follows:

$$p^{k+1} = \phi^{k+1} + p_\star^{k+1} - \nu\nabla\cdot\boldsymbol{u}^{k+1}. \tag{4.8}$$

4.3. Weak formulation and spatial discretization

To develop the fully discretized solution algorithm, the two problems (4.6) and (4.7) are recast in discrete weak forms using the Galerkin method. To simplify notation we use the inner product notation for scalar functions $(q,p)\equiv\int_\Omega qp$ and for vector-valued functions $(\boldsymbol{u},\boldsymbol{v})\equiv\int_\Omega \boldsymbol{u}\cdot\boldsymbol{v}$. Moreover, we define $(\nabla\boldsymbol{u},\nabla\boldsymbol{v})\equiv\int_\Omega\sum_{i=1}^d\sum_{j=1}^d\frac{\partial u_j}{\partial x_i}\frac{\partial v_j}{\partial x_i}$, where $d=2$ is the space dimension.

To account for the possible loss of coercivity in the momentum equation when the Reynolds number becomes very large and the grid is not fine enough, we approximate the velocity by using the two-level $\mathbb{P}_2$ finite element setting defined in Section 2.4.2. Using the definitions of X_H, X_h^H, and X_h in (2.21)–(2.23), we set

$$\boldsymbol{X}_H=X_H\times X_H,\quad \boldsymbol{X}_h^H=X_h^H\times X_h^H,\quad \boldsymbol{X}_h=X_h\times X_h. \tag{4.9}$$

For approximating the pressure we introduce

$$M_h=\left\{q_h\in\mathscr{C}^0(\overline{\Omega});q_h|_{K_h}\in\mathbb{P}_1(K_h)\ \forall K_h\in\mathscr{T}_h;\ \int_\Omega q_h=0\right\}. \tag{4.10}$$

The fully discrete weak form of the viscous step (4.6) is written as follows: For $k\geqslant 0$, seek $\boldsymbol{u}_h^{k+1}\in\boldsymbol{X}_h$, with $\boldsymbol{u}_h^{k+1}|_\Gamma=\mathbf{I}_h\boldsymbol{b}^{k+1}$ ($\mathbf{I}_h$ is the Lagrange interpolation operator in $\boldsymbol{X}_h$), such that for all $\boldsymbol{v}_h\in\boldsymbol{X}_h$, with $\boldsymbol{v}_h|_\Gamma=0$,

$$\left(\frac{D\boldsymbol{u}_h^{k+1}}{\Delta t},\boldsymbol{v}_h\right)+\nu\left(\nabla\boldsymbol{u}_h^{k+1},\nabla\boldsymbol{v}_h\right)+\left(\mathrm{NL}(\boldsymbol{u}_{h,\star}^{k+1},\boldsymbol{u}_h^{k+1}),\boldsymbol{v}_h\right)+\boldsymbol{b}_h(\boldsymbol{u}_h^{k+1,H},\boldsymbol{v}_h^H)=-\boldsymbol{c}_h^H(\boldsymbol{u}_h^k,\boldsymbol{u}_h^k,\boldsymbol{v}_h)-\left(\nabla p_{h,\star}^{k+1},\boldsymbol{v}_h\right) \tag{4.11}$$

where, using definitions (2.24) and (3.5), we set $\boldsymbol{b}_h(\boldsymbol{u}_h^H,\boldsymbol{v}_h^H)=\sum_{i=1}^2 b_h(u_{i,h}^H,v_{i,h}^H)$ and $\boldsymbol{c}_h^H(\boldsymbol{u}_h,\boldsymbol{w}_h,\boldsymbol{v}_h)=\sum_{i=1}^2 c_h^H(u_{i,h},w_{i,h},v_{i,h})$.

The fully discrete form of the projection step (4.7) is set as follows: Find $\phi_h^{k+1}\in M_h$ such that for all $q_h\in M_h$

$$\left(\nabla\phi_h^{k+1},\nabla q_h\right)=-\frac{\beta_{k+1}}{\Delta t}\left(\nabla\cdot\boldsymbol{u}_h^{k+1},q_h\right). \tag{4.12}$$

The pressure is updated by solving the following mass problem: Seek $p_h^{k+1}\in M_h$ such that for all $q_h\in M_h$

$$\left(p_h^{k+1},q_h\right)=\left(\phi_h^{k+1}+p_{h,\star}^{k+1}-\nu\nabla\cdot\boldsymbol{u}_h^{k+1},q_h\right). \tag{4.13}$$

4.4. Post-processing procedure

Since the time discretization of the projection method induces a time-splitting error, we will resort to a method introduced in [21] to a posteriori compute a pressure field with a lower splitting error. We briefly recall the method here. We first observe that the momentum equation can be written as follows:

$$\partial_t\boldsymbol{u}+\mathrm{NL}(\boldsymbol{u},\boldsymbol{u})=-\nabla p-\nu\nabla\times\nabla\times\boldsymbol{u} \tag{4.14}$$

and can be regarded as an equation for ∇p, assuming $\boldsymbol{u}$ is known. Then, multiplying (4.14) by ∇v, $\forall v\in H^1(\Omega)$, integrating by parts the evolutionary term, using the incompressibility condition $\nabla\cdot\boldsymbol{u}=0$ together with the boundary condition on the normal component of the velocity, we obtain

$$(\nabla p,\nabla v)=-(\mathrm{NL}(\boldsymbol{u},\boldsymbol{u}),\nabla v)-\nu(\nabla\times\nabla\times\boldsymbol{u},\nabla v)-\int_\Gamma\partial_t(\boldsymbol{n}\cdot\boldsymbol{b})v \tag{4.15}$$

$\forall v\in H^1(\Omega)$. Finally the viscous term can be reduced to a surface integral by using again an integration by parts and using the fact that the curl of gradients is zero. In the present two-dimensional setting, we obtain

$$(\nabla p,\nabla v)=-(\mathrm{NL}(\boldsymbol{u},\boldsymbol{u}),\nabla v)+\nu\int_\Gamma\hat{\boldsymbol{z}}\cdot\nabla\times\boldsymbol{u}\ \boldsymbol{\tau}\cdot\nabla v-\int_\Gamma\partial_t(\boldsymbol{n}\cdot\boldsymbol{b})v \tag{4.16}$$

$\forall v\in H^1(\Omega)$, where $\boldsymbol{\tau}$ is the unit tangential vector and $\hat{\boldsymbol{z}}\cdot\nabla\times\boldsymbol{u}=\partial_x u_y-\partial_y u_x$. This is a Poisson problem for the pressure with a Neumann boundary conditions involving the tangential trace of vorticity.

To compute the post-processed pressure we introduce

$$N_h^{\mathrm{post}}=\left\{q_h\in\mathscr{C}^0(\overline{\Omega});q_h|_{K_h}\in\mathbb{P}_\ell(K_h)\ \forall K_h\in\mathscr{T}_h;\ \int_\Omega q_h=0\right\}, \tag{4.17}$$

where ℓ is either 1 or 2. Using the fully discrete setting described in the previous section, the post-processed pressure at the time level t_k is computed by solving for $p_h^{k,\mathrm{post}}\in N_h^{\mathrm{post}}$ such that for all $q_h\in N_h^{\mathrm{post}}$

$$\left(\nabla p_h^{k,\mathrm{post}},\nabla q_h\right)=-\left(\mathrm{NL}(\boldsymbol{u}_h^k,\boldsymbol{u}_h^k),\nabla q_h\right)+\nu\int_\Gamma\hat{\boldsymbol{z}}\cdot\nabla\times\boldsymbol{u}_h^k\ \boldsymbol{\tau}\cdot\nabla q_h-\int_\Gamma\partial_t(\boldsymbol{n}\cdot\boldsymbol{b}^k)q_h. \tag{4.18}$$

Whether this strategy can be proved to deliver full $\mathcal{O}(h^{\ell+1})$ convergence on the pressure in the L^2-norm is an open problem, but convergence tests reported below suggest that the post-processed pressure is more accurate than p_h^k. The post-processed pressure is computed only when needed; it is never used to march in time.

4.5. Convergence tests

In this section we test the accuracy of the subgrid stabilized projection method and that of the aforementioned post-processing technique. The domain is the unit square $\Omega = [0,1]^2$. The reference solution is as follows:

$$\begin{cases} u_{\text{ex}} = -\cos x \sin y\, g(t), \quad v_{\text{ex}} = \sin x \cos y\, g(t), \\ p_{\text{ex}} = -\frac{1}{4}[\cos(2x) + \cos(2y)]g^2(t), \end{cases} \tag{4.19}$$

where $g(t) = \sin(2t)$. Writing the velocity in the form $\boldsymbol{u}_{\text{ex}} = \bar{\boldsymbol{u}}(x,y)g(t)$, the source term corresponding to the Navier–Stokes equation is

$$\boldsymbol{f} = \bar{\boldsymbol{u}}(x,y)\left[g'(t) + \frac{2g(t)}{Re}\right].$$

The Reynolds number is set to 100,000 and the time convergence tests are performed on two meshes composed of 2×20^2 and 2×40^2 triangles, respectively. Since the analytical solution we use is very smooth, the non-linear stabilizing term is not employed in the numerical test considered in this section.

4.5.1. Convergence test for the subgrid stabilized projection method

We use the algorithm (4.11)–(4.13). The pressure is approximated using $\mathbb{P}_1$ elements and the velocity is approximated using $\mathbb{P}_2$ elements. The full markers refer to the subgrid stabilized technique whereas the empty markers refer to the non-stabilized technique. Fig. 11 shows the maximum value over the time interval $[0,1.5]$ of the L^2-norm of the error in the pressure and the H^1- and L^2-norms of the error in the velocity. The graph on the left panel shows the results obtained on the 2×20^2 grid and that on the right panel shows the results obtained on the 2×40^2 grid. The range of time steps explored is $\Delta t \in [10^{-3}, 10^{-1}]$.

The results show that, when the meshsize of the grid is fixed, second-order accuracy in time is achieved, as expected from theoretical considerations. The different saturation levels we observe as $\Delta t \to 0$ are due to the spatial discretization error, which is $\mathcal{O}(h^2)$ and $\mathcal{O}(h^3)$ for the velocity in the H^1-norm and the L^2-norm, respectively, and is $\mathcal{O}(h^2)$ for the pressure in the L^2-norm.

When comparing the stabilized solution curves with the non-stabilized ones it is evident that the saturation of the error occurs at a significantly higher level for the non-stabilized solution than for the stabilized one. Although only the advection–diffusion step is stabilized, the subgrid stabilization significantly increases the accuracy of the pressure also.

Note that refining the grid increases the error in the non-stabilized solution. To illustrate this behavior we have performed a space convergence test fixing $\Delta t = 10^{-3}$. The results are shown in Fig. 12. The main conclusion is that, for this

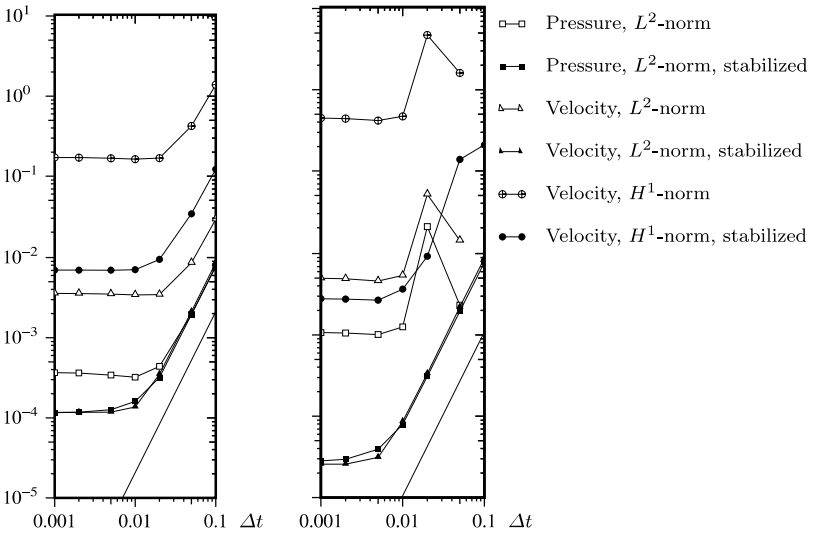

Fig. 11. Time convergence test, $Re = 100{,}000$. Left: 2×20^2 grid. Right: 2×40^2 grid. Curves with full (empty) markers refer to the (un)stabilized technique. The solid line without markers materializes the second-order slope.

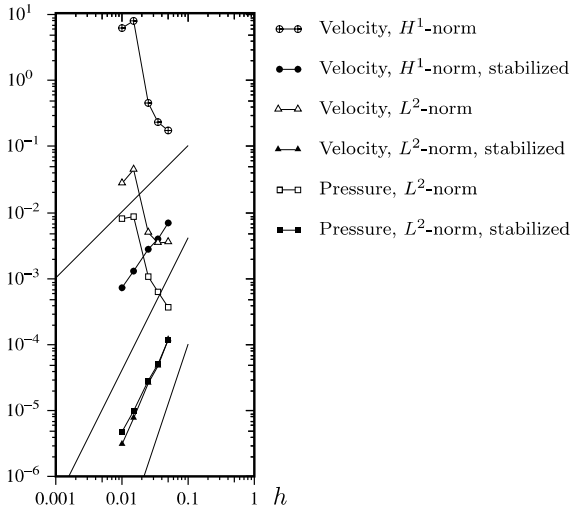

Fig. 12. Spatial convergence test, $Re = 100{,}000$. Error versus meshsize h. Curves with full (empty) markers refer to the (un)stabilized technique. The solid lines without markers materialize first-, second-, and third-order slopes.

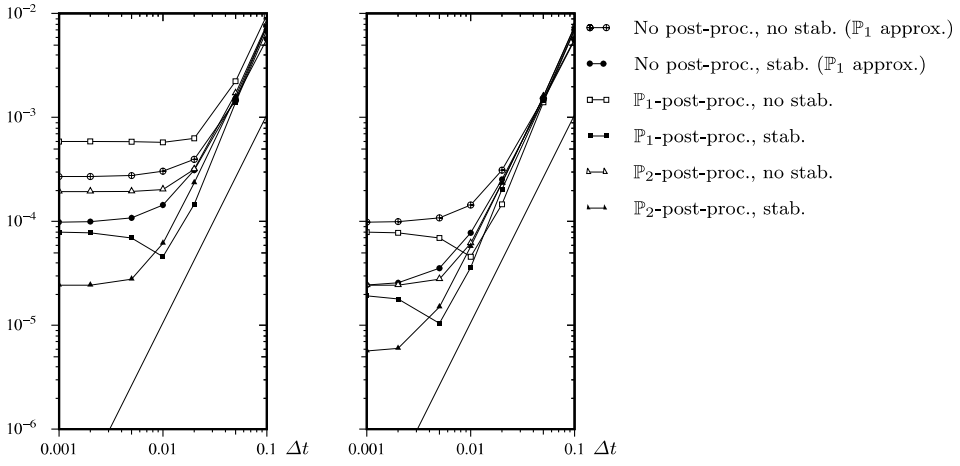

Fig. 13. Time convergence test. The L^2-norm of the error in the pressure field versus Δt. Left: 2×20^2 grid. Right: 2×40^2 grid. The line without markers materialize second-order slope.

high Reynolds number and for the range of meshsizes explored $h \in [10^{-2}, 10^{-1}]$, the non-stabilized solution diverges whereas the stabilized solution maintains the correct convergence rates.

4.5.2. Convergence test for the post-processing technique

In this section we test the accuracy of our proposed post-processing technique using either $\mathbb{P}_1$ or $\mathbb{P}_2$ approximation for the pressure, i.e., $\ell = 1$ or $\ell = 2$ (see (4.17)). In Fig. 13 we report the L^2-norm of the difference between the post-processed pressure and the exact pressure at time $t = 1$, i.e., $\|(p_h^{\text{post}} - p)|_{t=1}\|_{L^2}$. This error is plotted as a function of the time step Δt. The computation is done with $Re = 100{,}000$. We also show in this figure the L^2-norm of the error in the pressure that is provided by the projection method; this pressure field is henceforth referred to as the non-post-processed pressure.

It is clear that the post-processing technique substantially improves the accuracy of the pressure whether the $\mathbb{P}_1$ or $\mathbb{P}_2$ interpolation is used. Improvement of almost one order of magnitude is observed when using $\mathbb{P}_2$ approximation for post-processing. For $\mathbb{P}_1$ post-processing the results are qualitatively similar but the improvement is definitely less significant as Δt decreases.

5. Numerical results on NACA0012 at zero incidence

In this section we use the algorithm described in the previous sections to simulate the flow around the NACA0012 airfoil at zero incidence starting from the inviscid potential flow. Three different Reynolds numbers are simulated: $Re = 100{,}000$, $Re = 500{,}000$, $Re = 1{,}000{,}000$.

5.1. The setting

The velocity of the incoming flow at infinity is the reference velocity scale. The chord of the airfoil is the reference length scale. The computational domain is the rectangle $[-5,6]\times[-5,5]$. The initial flow field is that of the inviscid potential flow. The upstream velocity is enforced on $\{x=-5;\ -5\leqslant y\leqslant 5\}\cup\{-5\leqslant x\leqslant 6;\ y=\pm 5\}$. Natural boundary conditions are enforced at the downstream boundary Γ_{out}, i.e., we impose $\boldsymbol{\tau}\cdot\boldsymbol{u}|_{\Gamma_{\text{out}}}=0$ and $\nabla\cdot\boldsymbol{u}|_{\Gamma_{\text{out}}}=0$, and the latter, by virtue of the former, becomes $\partial_n(\boldsymbol{n}\cdot\boldsymbol{u})|_{\Gamma_{\text{out}}}=0$.

The grid we used has been generated taking the simulation at $Re=100{,}000$ as a guideline for establishing the necessary space and time discretization parameters. Due to our limited hardware resources (LINUX PC) and the single-processor architecture of the program, we used the same grid when simulating flows at Reynolds numbers higher than 100,000. The grid consists of about 172,000 nodes for the quadratic interpolation of velocity and 43,500 nodes for the linear pressure. Two enlargement of the coarse triangulation $\mathscr{T}_H$ around the airfoil and near its leading edge are shown in Fig. 14; each triangle is divided into four subtriangles to yield the mesh $\mathscr{T}_h$. The spatial discretization inside the boundary layer is $h\approx 10^{-3}$. Since in dimension two, the thickness of boundary layers scale like $Re^{\frac{1}{2}}$, our mesh captures the viscous boundary layer for $Re\lesssim 10^6$. However, when $Re=10^6$, h is just of the same order as the thickness of the boundary layer, and our simulation may not represent the complete real physics inside the boundary layer at this extreme Reynolds number, i.e., the results should be considered to be only qualitative.

As far as the initialization of the computation is concerned, even when respecting the compatibility conditions (4.2) as we did, the impulsive start yields singularities on the higher order derivatives of the velocity and the pressure. This difficulty has been dealt with by proceeding as follows. We employ $\Delta t=10^{-3}$ and $Re=1000$ for the first 20 time steps and we use $\Delta t=10^{-3}$ and $Re=10{,}000$ for the next 20 time steps. For the subsequent time steps the Reynolds number is set to its correct value and Δt is set to 10^{-4}.

The values of the subgrid viscosity coefficient c_b and that of the shock capturing term c_{sc} that we used in our computations are reported in Table 1. The criteria for choosing these values has simply been based on a trial and error strategy. We have tried to eliminate the macroscopic oscillations while keeping c_b and c_{sc} as small as possible so as to depart as little as possible from the original non-stabilized problem. Note that owing to the scalings chosen for the subgrid viscosity (2.24) and the non-linear shock capturing viscosity (3.5), the coefficients c_b and c_{sc} scale like the advection velocity.

As a first introductory result we show in Fig. 15 the effect of our stabilization strategy by comparing the Galerkin solution and the stabilized one. We show the isolines of the Euclidean norm of the velocity at time $t=0.5$ and $Re=100{,}000$. The Galerkin solution is in the left panel and the stabilized one is in the right panel. It is clear that our stabilization strategy removes the spurious oscillations plaguing the Galerkin solution and the stabilized solution has the expected qualitative

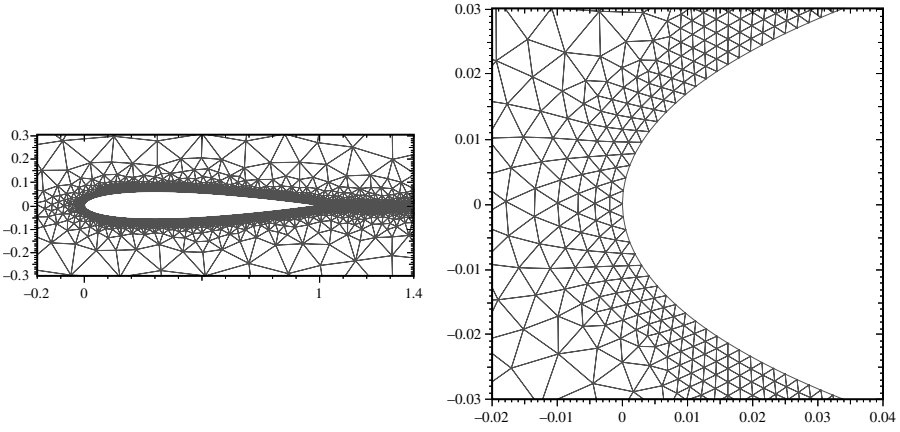

Fig. 14. Coarse mesh $\mathscr{T}_H$ around the airfoil (left) and near its leading edge (right).

Table 1
Values of the stabilizing coefficients c_b and c_{sc} used in the simulations

	Re		
	100,000	500,000	1,000,000
c_b	0.2	0.2	0.25
c_{sc}	20	20	30

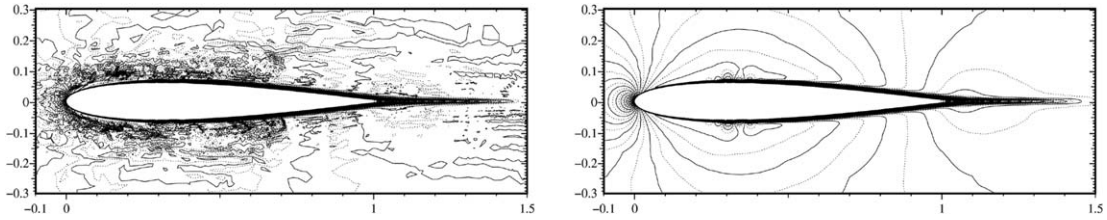

Fig. 15. Isovalues of the Euclidean norm of the velocity at $t = 0.5$, $Re = 100{,}000$. Left: Galerkin solution. Right: stabilized solution.

behavior. For the sake of brevity, we do not show more comparisons between stabilized and non-stabilized solutions at higher Reynolds numbers or later times since the non-stabilized solutions are of an increasingly worse quality.

5.2. Flow at Re = 100,000

We show in this section computations done at $Re = 100{,}000$. The choice for this particular first Reynolds number is based on the fact that the boundary layer of the 3D flow starts to become turbulent at slightly higher Reynolds numbers ($200{,}000 \leqslant Re \leqslant 300{,}000$). The flow is laminar at $Re = 100{,}000$.

Henceforth we compare our results with steady-state solutions computed using the `Xfoil` software, see [6]. The solution method implemented in `Xfoil` couples the inviscid potential flow theory with a model of turbulent boundary layer. We have selected this software since it is easily available and the models it uses are generally believed to fit well the zero incidence situation in the range of Reynolds numbers considered in this work. As described in [6], `Xfoil` uses the e^n method to determine the position of the transition point. This method is reasonable as long as the 2D Tollmien–Schlichting instability is the principal mechanism that leads to transition. This is an acceptable hypothesis for the airfoil problem in a free stream configuration with very low disturbance levels. The exponent n is the logarithm of the maximum amplification ratio of the most amplified frequency. This coefficient reaches the critical value n_{crit} at the laminar–turbulent transition. The value of n_{crit} depends on the level of disturbance of the experiment: for instance, in a clean wind tunnel $n_{\text{crit}} \in [10, 12]$, while in a dirty one $n_{\text{crit}} \in [4, 8]$. A value commonly used in the literature is $n_{\text{crit}} = 9$ and we picked this value for our comparisons.

We show in Fig. 16 the pressure coefficient c_p at time $t = 5$ on the upper and lower surfaces of the NACA0012 airfoil ($c_p = (P - P_\infty)/\frac{1}{2}\rho V_\infty^2$, with $\rho = 1$, $P_\infty = 0$). The reference (symmetric) solution given by the `Xfoil` code is the solid line with markers. The results from the subgrid stabilized projection method are shown in the left panel. The results obtained by means of the $\mathbb{P}_1$ and $\mathbb{P}_2$ post-processing technique are shown in the middle and right panels, respectively. We observe that the non-post-processed pressure field differs noticeably from the reference solution. Both the $\mathbb{P}_1$ and $\mathbb{P}_2$ post-processed pressure fields seem to match more accurately the reference solution.

To explain the origin of the non-symmetric pressure profiles observed in Fig. 16, we show in Fig. 17 snapshots of the $\mathbb{P}_2$ post-processed pressure coefficient at times $t = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0$, and 4.5. Oscillations starting from the leading-edge and traveling down to the trailing-edge are clearly visible. These oscillations are induced by vortices that have been generated at the leading-edge and are transported downstream. The symmetry of the flow is almost perfectly preserved until the vortices reach the trailing-edge, then the vortices trigger the von Karman instability. The vertical symmetry is progressively lost as the von Karman wake develops. This behavior is strongly related to the two-dimensional character of our simulation. In a hypothetical three-dimensional simulation we would not have observed a von Karman wake at $Re = 100{,}000$ because a vortex instability would have broken the two-dimensional vortices into smaller structures with spanwise features.

The vorticity fields at times $t = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5$, and 5.0 are shown in Fig. 18.

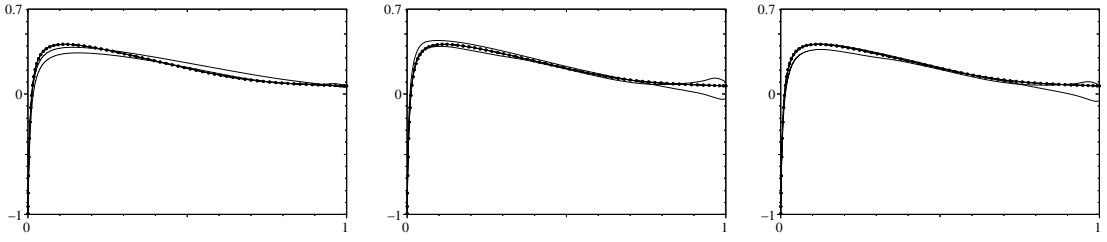

Fig. 16. c_p profiles on upper and lower sides of NACA0012 airfoil at zero incidence, $Re = 100{,}000$, $t = 5.0$. Left: non-post-processed pressure. Center: $\mathbb{P}_1$ post-processing. Right: $\mathbb{P}_2$ post-processing. Curve with markers is `Xfoil` solution.

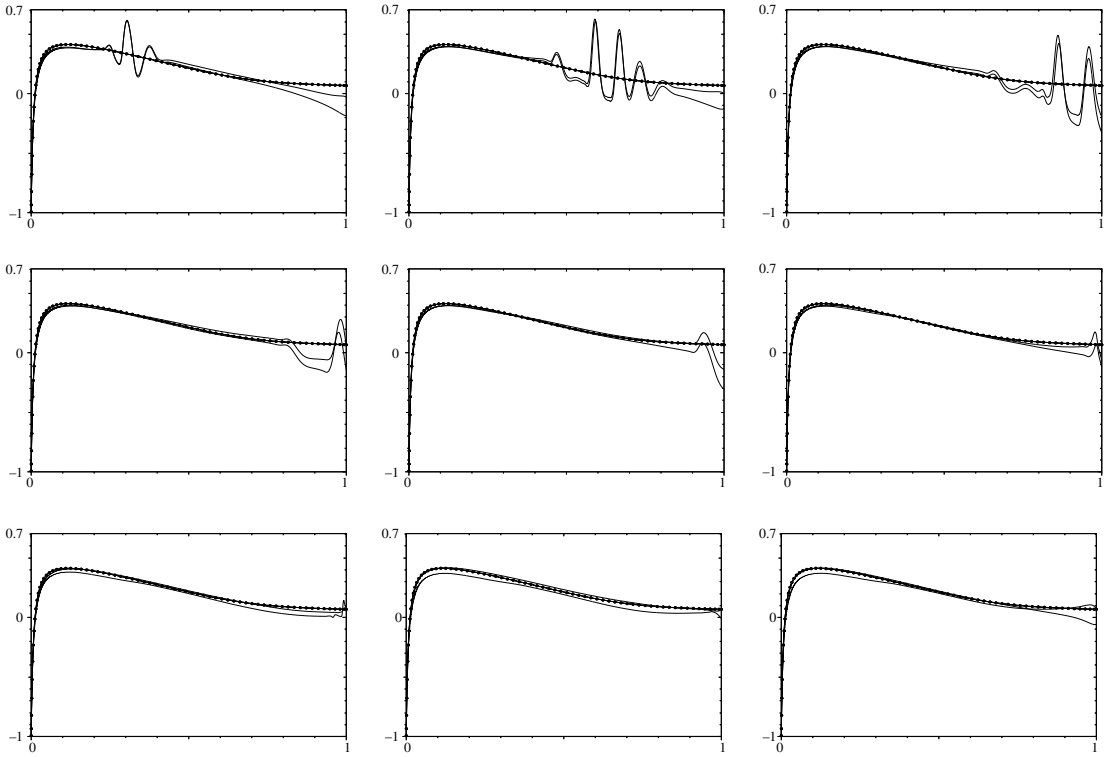

Fig. 17. From left to right and top to bottom: $\mathbb{P}_2$ post-processed pressure coefficient at $t = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0$, and 4.5, $Re = 100{,}000$. The curve with markers is the `Xfoil` reference solution.

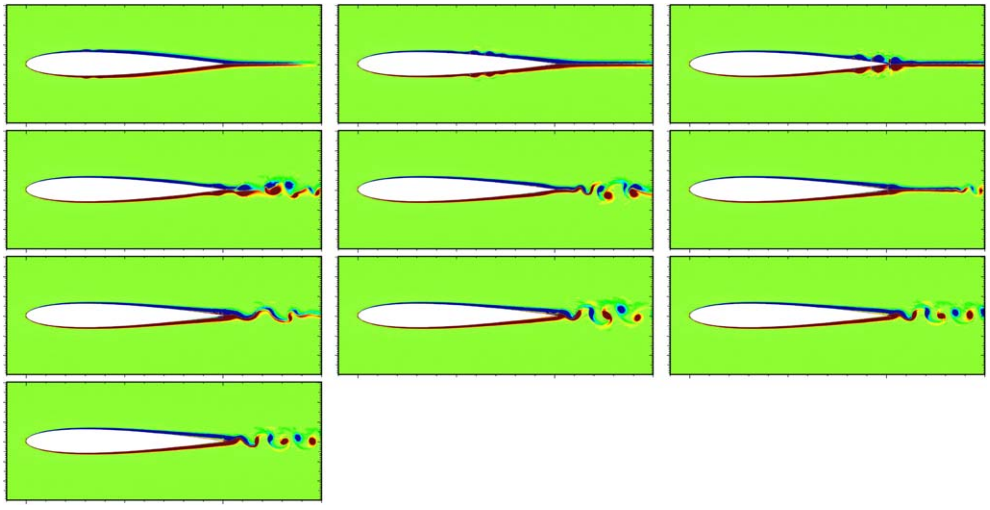

Fig. 18. From left to right and top to bottom: vorticity field at $t = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5$, and 5.0, $Re = 100{,}000$.

5.3. Flow at Re = 500,000

We now simulate the flow around the NACA0012 airfoil at $Re = 500{,}000$ keeping the incidence angle to be zero.

Contrary to the $Re = 100{,}000$ configuration, the flow near the airfoil does not tend to a quasi-steady regime over its entire surface. The reference numerical solution obtained by `Xfoil` predicts the laminar–turbulent transition point at 79% of the chord from the leading-edge. Our simulation yields an unsteady solution as can be seen in Fig. 19 where we show snapshots of the $\mathbb{P}_2$ post-processed pressure coefficient. After a transient time period lasting from $t = 0$ to $t = 3$, during which vortices generated at the leading-edge at $t = 0$ are advected downstream, we observe pressure oscillations occur-

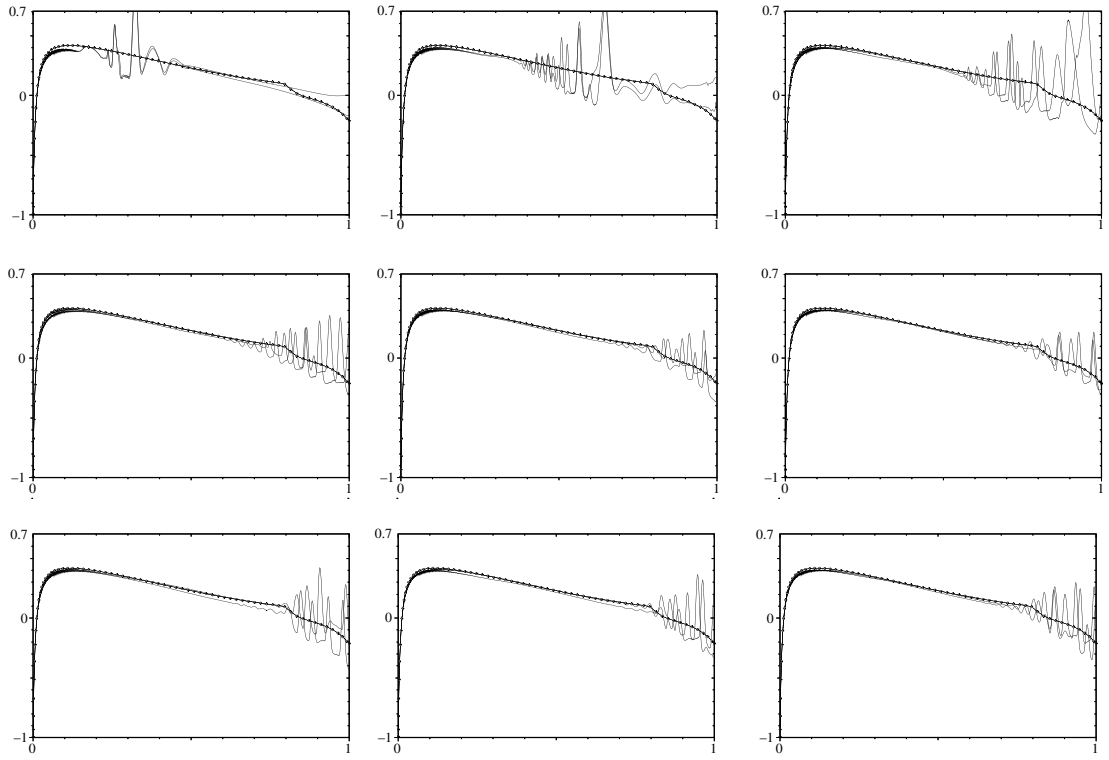

Fig. 19. From left to right and top to bottom: $\mathbb{P}_2$ post-processed pressure coefficient at $t = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0$, and 4.5, $Re = 500{,}000$. The curve with markers is the `Xfoil` reference solution.

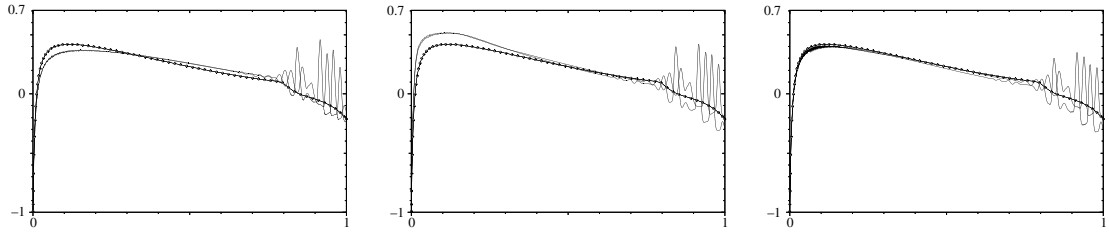

Fig. 20. c_p profiles on upper and lower sides of NACA0012 airfoil at zero incidence, $Re = 500{,}000$, $t = 5.0$. Left: non-post-processed pressure. Center: $\mathbb{P}_1$ post-processing. Right: $\mathbb{P}_2$ post-processing. Curve with markers is `Xfoil` solution.

ring in close proximity to the transition point predicted by `Xfoil`. We conjecture that these oscillations are the manifestation of the Tollmien–Schlichting instability [24] occurring in the viscous boundary layer, but since there are not many degrees of freedom in the boundary layer we cannot claim it with certainty. Nevertheless, it is extremely satisfactory to observe that our stabilized simulation predicts quite well the position of the transition point.

We show in Fig. 20 the distribution of the pressure coefficient at $t = 5$. The non-post-processed pressure is shown in the left panel, while the post-processed pressure results with $\mathbb{P}_1$ and $\mathbb{P}_2$ interpolations are shown in the center and right panels, respectively. The two solutions with linear approximation show a comparable level of mismatch when compared with the `Xfoil` results, the non-post-processed pressure being slightly better than the post-processed one. The $\mathbb{P}_2$ post-processed pressure profile, instead, better agrees with that from `Xfoil`. All these results are consistent with the convergence tests reported in Section 4.5.2.

Snapshots of the vorticity field are shown in Fig. 21. Once the small vortices generated at the transition point reach the trailing-edge, they alternatively leave the upper side and the lower side of the airfoil. The wake thus created is composed of small alternating vortices interacting according to the standard inverse cascade scenario [17]: in two-dimensional flows, the transfer of energy goes from the smaller scales to the larger ones, preventing the small scales from surviving for long periods.

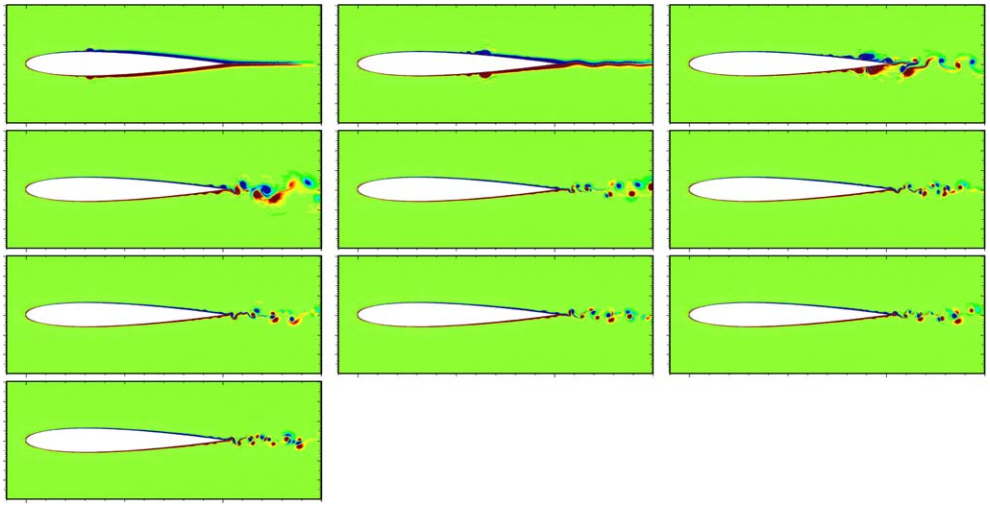

Fig. 21. From left to right and top to bottom: vorticity field at $t = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5$, and 5.0, $Re = 500{,}000$.

5.4. Flow at Re = 1,000,000

We now simulate the flow around the NACA0012 airfoil at $Re = 1{,}000{,}000$ keeping the zero incidence configuration. The behavior of the flow at $Re = 1{,}000{,}000$ is qualitatively similar to that at $Re = 500{,}000$, the main difference being that the laminar–turbulent transition point predicted by `Xfoil` is now at 69% of the chord from the leading-edge.

We show in Fig. 22 snapshots of the $\mathbb{P}_2$ post-processed pressure coefficient. We observe a transient time period between $t = 0$ and $t = 2.5$. During this transient, the vortices that are generated at the leading-edge at the initial time are advected downstream and eventually shed in the wake. After the transient period, we observe pressure oscillations occurring in close proximity to the transition point predicted by `Xfoil`. Like the simulation at $Re = 500{,}000$, we think that these oscillations

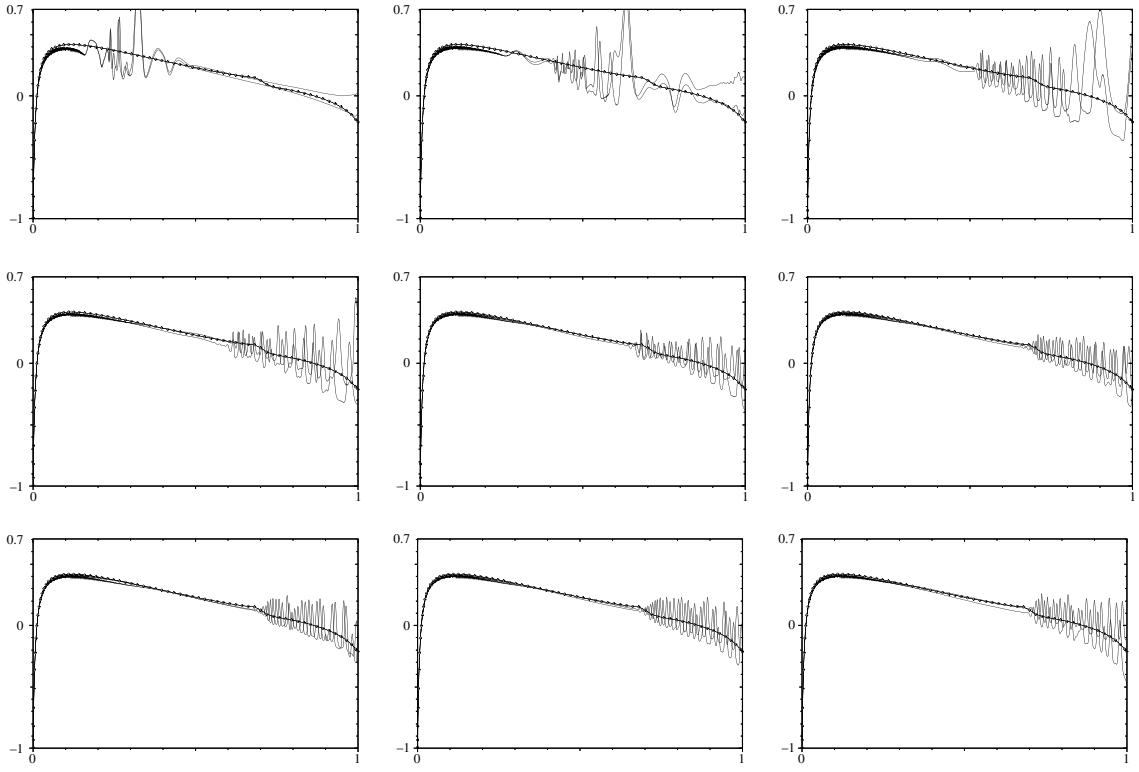

Fig. 22. From left to right and top to bottom: $\mathbb{P}_2$ post-processed pressure coefficient at $t = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0$, and 4.5, $Re = 1{,}000{,}000$. The curve with markers is the `Xfoil` reference solution.

are the manifestation of the Tollmien–Schlichting instability [24] occurring in the viscous boundary layer, but once again we cannot make a definitive claim in this respect since the number of degrees of freedom in the boundary layer is small. Nevertheless, considering the extremely high Reynolds number we are working with, it is satisfactory to observe that the simulation accurately predicts the position of the transition point.

We compare in Fig. 23 the non-post-processed and the post-processed pressure fields at $t = 5$. We observe again that the $\mathbb{P}_2$ post-processed pressure field is that which matches the most accurately the `Xfoil` reference solution.

We report in Fig. 24 snapshots of the vorticity field at times $t = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5$, and 5.0.

5.5. Concluding remarks

When carefully looking at Figs. 20 and 23, we observe residual oscillations of the pressure coefficient in the fore part of the airfoil. Since these oscillations have no counterpart in the calculation done at $Re = 100{,}000$, see Fig. 16, we are led to conclude that the mesh we have used is barely fine enough to fully resolve the boundary layer in the leading edge region (where it is the finest) at $Re = 500{,}000$ and $Re = 1{,}000{,}000$. These oscillations could have been entirely eliminated by slightly increasing the values of the coefficients c_b and c_{sc}, but we did not play this game.

Nevertheless, despite the residual oscillations, the present simulations are very satisfactory when compared to the `Xfoil` averaged solutions.

We finish by showing in Fig. 25 the pressure profiles at $t = 5$ for $Re = 300{,}000$, $Re = 500{,}000$, and $Re = 1{,}000{,}000$. This figure summarizes quite well the content of the paper. The fact that the method is able to compute accurately the laminar–turbulent transition point at $Re = 300{,}000$, $Re = 500{,}000$ and $Re = 1{,}000{,}000$ on the same grid proves that the stabilizing terms do exactly what they are meant for: that is, they are strong enough to remove the spurious oscillations while being small enough not to overwhelm the physical viscosity.

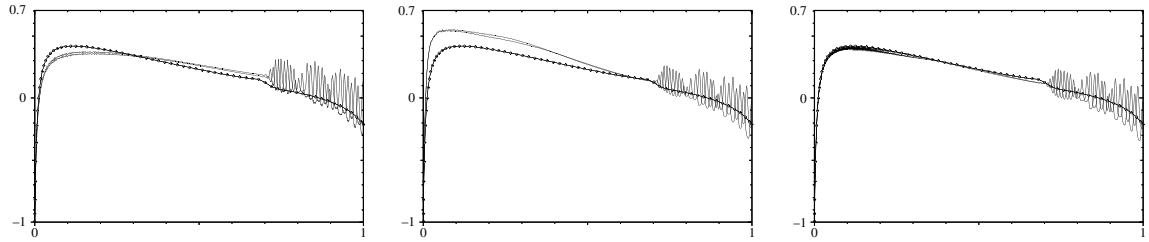

Fig. 23. c_p profiles on upper and lower sides of NACA0012 airfoil at zero incidence, $Re = 1{,}000{,}000$, $t = 5.0$. Left: non-post-processed pressure. Center: $\mathbb{P}_1$ post-processing. Right: $\mathbb{P}_2$ post-processing. Curve with markers is `Xfoil` solution.

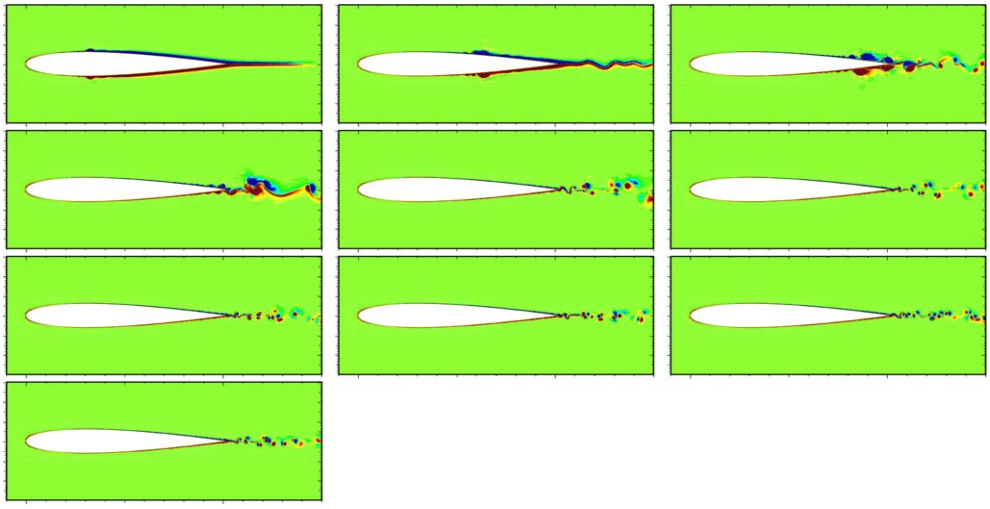

Fig. 24. From left to right and top to bottom: vorticity field at $t = 0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0, 4.5$, and 5.0, $Re = 1{,}000{,}000$.

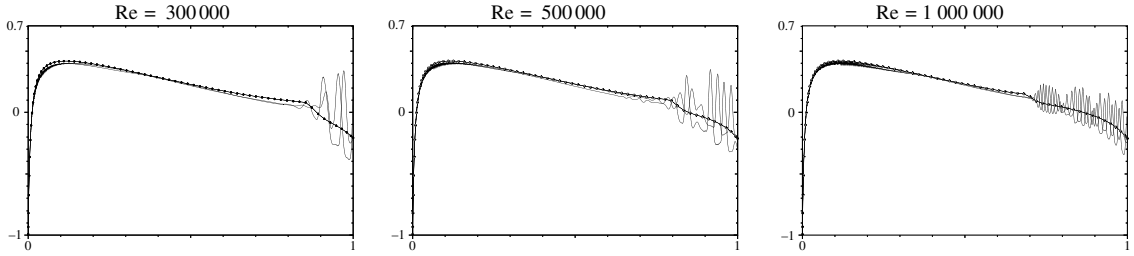

Fig. 25. Position of the transition point for $Re = 300{,}000$, $Re = 500{,}000$, and $Re = 1{,}000{,}000$ at $t = 5$.

References

[1] C. Baiocchi, F. Brezzi, L.P. Franca, Virtual bubbles and Galerkin–least-squares type methods (GaLS), Comput. Methods Appl. Mech. Engrg. 105 (1993) 125–141.
[2] A.N. Brooks, T.J.R. Hughes, Streamline upwind/Petrov–Galerkin formulations for convective dominated flows with particular emphasis on the incompressible Navier–Stokes equations, Comput. Methods Appl. Mech. Engrg. 32 (1982) 199–259.
[3] A.J. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comput. 22 (1968) 745–762.
[4] A.J. Chorin, On the convergence of discrete approximations to the Navier–Stokes equations, Math. Comput. 23 (1969) 341–353.
[5] B. Cockburn, G.E. Karniadakis, C.W. Shu, in: Discontinuous Galerkin Methods—Theory, Computation and Applications, Lecture Notes in Computer Science and Engineering, vol. 11, Springer, 2000.
[6] M. Drela, M.B. Giles, Viscous-inviscid analysis of transonic and low Reynolds number airfoils, AIAA J. 25 (10) (1987) 1347–1355.
[7] A. Ern, J.-L. Guermond, in: Theory and Practice of Finite Elements, Series on Applied Mathematical Sciences, vol. 159, Springer-Verlag, Berlin, 2004.
[8] J.-L. Guermond, Stabilization of Galerkin approximations of transport equations by subgrid modeling, M2AN Math. Model. Numer. Anal. 33 (6) (1999) 1293–1316.
[9] J.-L. Guermond, Subgrid stabilization of Galerkin approximations of linear contraction semi-groups of class C^0 in Hilbert spaces, Numer. Methods Partial Diff. Equ. 17 (2001) 1–25.
[10] J.-L. Guermond, L. Quartapelle, On the approximation of the unsteady Navier–Stokes equations by finite element projection methods, Numer. Math. 80 (1998) 207–238.
[11] J.-L. Guermond, J. Shen, On the error estimates for the rotational pressure-correction projection methods, Math. Comput. 73 (248) (2004) 1719–1737 (electronic).
[12] T.J.R. Hughes, G.R. Feijóo, L. Mazzei, J. Quincy, The variational multiscale method—a paradigm for computational mechanics, Comput. Methods Appl. Mech. Engrg. 166 (1–2) (1998) 3–24.
[13] T.J.R. Hughes, L.P. Franca, G.M. Hulbert, A new finite element formulation for computational fluid dynamics. VIII. The Galerkin/Least-Squares method for advection-diffusive equations, Comput. Methods Appl. Mech. Engrg. 73 (1989) 173–189.
[14] T.J.R. Hughes, M. Mallet, A new finite element formulation for computational fluid dynamics. IV. A discontinuity-capturing operator for multidimensional advective-diffusive systems, Comput. Methods Appl. Mech. Engrg. 58 (1986) 329–336.
[15] C. Johnson, U. Nävert, J. Pitkäranta, Finite element methods for linear hyperbolic equations, Comput. Methods Appl. Mech. Engrg. 45 (1984) 285–312.
[16] C. Johnson, A. Szepessy, P. Hansbo, On the convergence of shock-capturing streamline diffusion finite element methods for hyperbolic conservation laws, Math. Comput. 54 (189) (1990) 107–129.
[17] R.H. Kraichnan, Inertial ranges in two-dimensional turbulence, Phys. Fluids 10 (1967) 1417–1423.
[18] O.A. Ladyzhenskaya, The Mathematical Theory of Viscous Incompressible Flow, second English ed., revised and enlarged, Translated from the Russian by Richard A. Silverman and John Chu, Mathematics and its Applications, vol. 2. Gordon and Breach Science Publishers, New York, 1969.
[19] P. Lesaint, P.-A. Raviart, On a finite element method for solving the neutron transport equation, in: C. de Boors (Ed.), Mathematical Aspects of Finite Elements in Partial Differential Equations, Academic Press, 1974, pp. 89–123.
[20] R.J. LeVeque, in: Numerical Methods for Conservation Laws, Lectures in Mathematics ETH Zürich, Birkhäuser Verlag, Basel, 1990.
[21] A. Marra, A. Mola, L. Quartapelle, L. Riviello, Calculation of impulsively started incompressible viscous flows, Int. J. Numer. Methods Fluids 46 (2004) 877–902.
[22] W.H. Reed, T.R. Hill, Triangular mesh methods for the neutron transport equation. Technical Report LA-UR-73-479, Los Alamos Scientific Laboratory, Los Alamos, NM, 1973.
[23] W. Rudin, Analyse réelle et complexe, Masson, Paris, 1987, fourth ed.
[24] H. Schlichting, Boundary layer theory Translated by, in: J. Kestin (Ed.), McGraw-Hill Series in Mechanical Engineering, fourth ed., McGraw-Hill Book Co., Inc., New York, 1960.
[25] R. Temam, Sur l'approximation de la solution des équations de Navier–Stokes par la méthode des pas fractionnaires ii, Arch. Rat. Mech. Anal. 33 (1969) 377–385.
[26] R. Temam, in: Navier–Stokes Equations, Studies in Mathematics and its Applications, vol. 2, North-Holland, 1977.
[27] L.J.P. Timmermans, P.D. Minev, F.N. van de Vosse, An approximate projection scheme for incompressible flow using spectral elements, Int. J. Numer. Methods Fluids 22 (1996) 673–688.
[28] K. Yosida, in: Functional Analysis, Classics in Mathematics, Springer-Verlag, Berlin, Germany, 1995, Reprint of the sixth (1980) edition.