A face-centred finite volume method for laminar and turbulent incompressible flows

Luan M. Vieira [a,b,c], Matteo Giacomini [a,b], Ruben Sevilla [c,*], Antonio Huerta [a,b]

[a] *Laboratori de Càlcul Numèric (LaCàN), ETS de Ingeniería de Caminos, Canales y Puertos, Universitat Politècnica de Catalunya, Barcelona, Spain*
[b] *Centre Internacional de Mètodes Numèrics en Enginyeria (CIMNE), Barcelona, Spain*
[c] *Zienkiewicz Centre for Computational Engineering, Faculty of Science and Engineering, Swansea University, Swansea, SA1 8EN, Wales, UK*

ARTICLE INFO

ABSTRACT

This work develops, for the first time, a face-centred finite volume (FCFV) solver for the simulation of laminar and turbulent viscous incompressible flows. The formulation relies on the Reynolds-averaged Navier–Stokes (RANS) equations coupled with the negative Spalart–Allmaras (SA) model and three novel convective stabilisations, inspired by Riemann solvers, are derived and compared numerically. The resulting method achieves first-order convergence of the velocity, the velocity-gradient tensor and the pressure. FCFV accurately predicts engineering quantities of interest, such as drag and lift, on unstructured meshes and, by avoiding gradient reconstruction, the method is less sensitive to mesh quality than other FV methods, even in the presence of highly distorted and stretched cells. A monolithic and a staggered solution strategies for the RANS-SA system are derived and compared numerically. Numerical benchmarks, involving laminar and turbulent, steady and transient cases are used to assess the performance, accuracy and robustness of the proposed FCFV method.

1. Introduction

The simulation of incompressible flows is central to many areas of science and engineering, including aerodynamic design in the automotive industry, the modelling of water quality and the study of blood flow in the human circulatory system, just to name a few. The majority of computational fluid dynamics (CFD) tools – open-source, commercial, academic or industrial– are based on either second-order cell-centred or vertex-centred finite volume (FV) techniques [1,2].

Despite formally second-order accurate, one of the main drawbacks of classical cell-centred or vertex-centred FV methods is the need to perform a reconstruction of the gradient of the solution. In the presence of highly stretched or distorted cells, the reconstruction often leads to an important loss of accuracy [3,4].

The face-centred finite volume (FCFV) was initially proposed in [5] as an alternative to cell-centred or vertex-centred approaches, avoiding any reconstruction of the derivatives and results have proved that it is almost insensitive to mesh distortion. In the same way as second-order cell-centred FV methods can be seen as a low order discontinuous Galerkin (DG) with constant degree of approximation in each cell and second-order vertex-centred FV methods can be seen as a conforming piecewise linear continuous finite element method, the FCFV can be seen as the hybridisable DG (HDG) method [6–9] using the lowest order approximation for cell and face variables. As such, the method is almost second order, in the sense that the gradient of the velocity converges with order one without the need of reconstructions and, therefore, with an accuracy that is less sensitive to mesh distortion. Furthermore, the method allows the use of the same degree of approximation for velocity and pressure, circumventing the so-called Ladyzhenskaya-Babuška-Brezzi (LBB) condition [10]. Compared to classical second-order FV methods, the FCFV method provides the same order of accuracy in the gradient of the velocity without any reconstruction.

Since its introduction, the FCFV method has been applied to a variety of problems including solid mechanics [11], heat transfer and Stokes flows [12–15] and viscous laminar compressible flows [16,17]. This work presents, for the first time, the development and application of the FCFV method to viscous incompressible flows to simulate steady-state and transient phenomena, in both laminar and turbulent regime, using the one-equation Spalart–Allmaras model [18,19].

The FCFV formulation of the incompressible Reynolds-averaged Navier–Stokes (RANS) equations coupled with the negative Spalart–Allmaras (SA) turbulence model [19] is presented. The velocity–pressure formulation of the momentum equation is adopted to ensure that the vorticity, required in the SA model, can be computed from the mixed

---

* Corresponding author.
E-mail address: r.sevilla@swansea.ac.uk (R. Sevilla).

variable, which is the velocity-gradient tensor. This is in contrast with other mixed formulations for incompressible flow problems [20–22] that use the Cauchy formulation of the momentum equation and the symmetric gradient of the velocity as the mixed variable. In addition, three novel stabilisation tensors for the convective part of the incompressible RANS equations are proposed. They are derived following the unified analysis presented in [16,23] for the Riemann solvers in the context of hybrid discretisation methods. Extensive numerical examples are then used to assess the performance and robustness of the proposed FCFV methodology. The tests include benchmarks featuring highly distorted and stretched meshes. Finally, two approaches, one monolithic and one staggered, to solve the RANS-SA equations are discussed, implemented and compared in terms of their computational complexity, efficiency and robustness.

The remainder of the paper is organised as follows. Section 2 briefly recalls the governing RANS and SA equations. The mixed integral form of the RANS-SA system is detailed in Section 3 and the FCFV solution algorithm is described in Section 4. Section 5 presents six numerical examples, encompassing laminar and turbulent, steady and transient cases, solved using triangular and quadrilateral meshes. The examples include test cases to assess the optimal convergence properties of the FCFV approximations and more complex benchmarks where the results are compared to reference solutions in the literature. The conclusions of the work are summarised in Section 6, whereas the new convective stabilisations are derived in Appendix A and computational details, including a comparison between monolithic and staggered approaches, are presented in Appendix B.

2. Governing equations

Let us consider an open bounded domain $\Omega \subset \mathbb{R}^{n_{sd}}$, where n_{sd} is the number of spatial dimensions, and a time interval $(0, T)$, with $T > 0$. The non-dimensional incompressible RANS equations coupled with the negative SA turbulence model [19] can be written as

$$
\begin{cases}
\dfrac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{\nabla}\cdot(\boldsymbol{u}\otimes\boldsymbol{u}) - \boldsymbol{\nabla}\cdot\left(\dfrac{2(1+\nu_t)}{Re}\boldsymbol{\nabla}^s\boldsymbol{u} - p\mathbf{I}_{n_{sd}}\right) = \boldsymbol{s} & \text{in } \Omega\times(0,T], \\[2mm]
\boldsymbol{\nabla}\cdot\boldsymbol{u} = 0 & \text{in } \Omega\times(0,T], \\[2mm]
\dfrac{\partial \tilde{\nu}}{\partial t} + \boldsymbol{\nabla}\cdot(\tilde{\nu}\boldsymbol{u}) - \boldsymbol{\nabla}\cdot\left(\dfrac{1+\tilde{\nu}f_n}{\sigma Re}\boldsymbol{\nabla}\tilde{\nu}\right) = s & \text{in } \Omega\times(0,T],
\end{cases}
\tag{1}
$$

where $\boldsymbol{u}$ is the non-dimensional velocity vector field, p is the non-dimensional modified pressure field, which involves the density and the turbulent kinetic energy, the strain-rate tensor is $\boldsymbol{\nabla}^s\boldsymbol{u} = (\boldsymbol{\nabla}\boldsymbol{u}+\boldsymbol{\nabla}^T\boldsymbol{u})/2$, Re denotes the Reynolds number and $\boldsymbol{s}$ is the body force. The non-dimensional turbulent viscosity ν_t, introduced by the Spalart–Allmaras turbulence model, is computed in terms of the turbulent variable $\tilde{\nu}$ as $\nu_t = \tilde{\nu}f_{v1}$. The right-hand side of the SA turbulence model is given by

$$
s=\begin{cases}
c_{b1}(1-f_{t2})\tilde{S}\tilde{\nu}+\dfrac{c_{b2}}{\sigma Re}\boldsymbol{\nabla}\tilde{\nu}\cdot\boldsymbol{\nabla}\tilde{\nu}-\dfrac{1}{Re}\left(c_{w1}f_w-\dfrac{c_{b1}}{\kappa^2}f_{t2}\right)\left(\dfrac{\tilde{\nu}}{d}\right)^2 & \text{if } \tilde{\nu}\geq 0, \\[3mm]
c_{b1}(1-c_{t3})S\tilde{\nu}+\dfrac{c_{b2}}{\sigma Re}\boldsymbol{\nabla}\tilde{\nu}\cdot\boldsymbol{\nabla}\tilde{\nu}+\dfrac{c_{w1}}{Re}\left(\dfrac{\tilde{\nu}}{d}\right)^2 & \text{otherwise,}
\end{cases}
\tag{2}
$$

where the three terms correspond to production, cross-diffusion and destruction, respectively. In the production term, the modified vorticity magnitude, $\bar{S}$, is introduced to avoid negative values [19], namely

$$
\tilde{S}=\begin{cases}
S+\bar{S} & \text{if } \bar{S}\geq -c_{v2}S, \\[2mm]
S+\dfrac{S(Sc_{v2}^2+\bar{S}c_{v3})}{S(c_{v3}-2c_{v2})-\bar{S}} & \text{otherwise,}
\end{cases}
\tag{3}
$$

where $S=\sqrt{2\boldsymbol{S}:\boldsymbol{S}}$ is the magnitude of the vorticity, $\boldsymbol{S}=(\boldsymbol{\nabla}\boldsymbol{u}-\boldsymbol{\nabla}^T\boldsymbol{u})/2$ is the vorticity tensor, $\bar{S}=\tilde{\nu}f_{v2}/(Re\,\kappa^2 d^2)$ and d is the minimum distance

to the wall. The SA model is completed with the closure functions

$$
\begin{aligned}
&\chi = \tilde{\nu}, && f_{v1} = \dfrac{\chi^3}{\chi^3+c_{v1}^3}, && f_{v2} = 1-\dfrac{\chi}{1+\chi f_{v1}}, \\[2mm]
&r = \min\left(\dfrac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2}, r_{lim}\right), && g = r+c_{w2}(r^6-r), && f_w = g\left(\dfrac{1+c_{w3}^6}{g^6+c_{w3}^6}\right)^{1/6}, \\[2mm]
&f_{t2} = c_{t3}\exp(-c_{t4}\chi^2), && f_n = \dfrac{c_{n1}+\chi^3}{c_{n1}-\chi^3},
\end{aligned}
\tag{4}
$$

and closure constants $\sigma = 2/3$, $c_{b1} = 0.1355$, $c_{b2} = 0.622$, $\kappa = 0.41$, $c_{w1} = c_{b1}/\kappa^2 + (1 + c_{b2})/\sigma$, $c_{v1} = 7.1$, $c_{t3} = 1.2$, $c_{t4} = 0.5$, $c_{v2} = 0.7$, $c_{v3} = 0.9$, $r_{lim} = 10$, $c_{w2} = 0.3$, $c_{w3} = 2$, and $c_{n1} = 16$.

Remark 1. The conservation of momentum, see the first equation in (1), can be rewritten in the usual velocity–pressure formulation as

$$
\dfrac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{\nabla}\cdot(\boldsymbol{u}\otimes\boldsymbol{u}) - \boldsymbol{\nabla}\cdot\left(\dfrac{1+\nu_t}{Re}\boldsymbol{\nabla}\boldsymbol{u} - p\mathbf{I}_{n_{sd}}\right) - \dfrac{1}{Re}\boldsymbol{\nabla}^T\boldsymbol{u}\,\boldsymbol{\nabla}\nu_t = \boldsymbol{s},
\tag{5}
$$

where the term $(1+\nu_t)/Re\,\boldsymbol{\nabla}\cdot(\boldsymbol{\nabla}^T\boldsymbol{u})$ vanishes due to the point-wise divergence-free condition on the velocity.

The FCFV method described in this work employs the velocity–pressure formulation of the momentum equation (5) to ensure that the vorticity $\boldsymbol{S}$ can be computed from the components of the velocity-gradient tensor $\boldsymbol{\nabla}\boldsymbol{u}$. This strategy is adopted to avoid performing a reconstruction of the derivatives of the velocity, as commonly done in other FV methods, which induces a loss of accuracy when employing distorted meshes.

The strong form of the incompressible RANS equations and SA turbulence model is closed with appropriate boundary conditions, written in the abstract form

$$
\begin{cases}
\boldsymbol{B}(\boldsymbol{u}, p, \boldsymbol{\nabla}\boldsymbol{u}, \tilde{\nu}) = \boldsymbol{0} & \text{on } \partial\Omega\times(0,T], \\[1mm]
b(\boldsymbol{u}, \tilde{\nu}, \boldsymbol{\nabla}\tilde{\nu}) = 0 & \text{on } \partial\Omega\times(0,T],
\end{cases}
\tag{6}
$$

using the boundary operators $\boldsymbol{B}$ and b. In this work, Dirichlet, Neumann, outflow, and symmetry boundary conditions are considered. The portions of the boundary $\partial\Omega$ where such conditions are imposed, denoted by Γ_D, Γ_N, Γ_O, and Γ_S, respectively, are disjoint by pairs and such that $\partial\Omega = \Gamma_D \cup \Gamma_N \cup \Gamma_O \cup \Gamma_S$. The particular expression of the boundary operators $\boldsymbol{B}$ and b is

$$
\boldsymbol{B} := \begin{cases}
\boldsymbol{u} - \boldsymbol{u}_D & \text{on } \Gamma_D, \\[2mm]
\left(\dfrac{2(1+\nu_t)}{Re}\boldsymbol{\nabla}^s\boldsymbol{u} - p\mathbf{I}_{n_{sd}}\right)\boldsymbol{n} - \boldsymbol{g} & \text{on } \Gamma_N, \\[2mm]
\left(\dfrac{(1+\nu_t)}{Re}\boldsymbol{\nabla}\boldsymbol{u} - p\mathbf{I}_{n_{sd}}\right)\boldsymbol{n} & \text{on } \Gamma_O, \\[2mm]
\begin{cases}
\boldsymbol{t}_k \cdot\left(\dfrac{(1+\nu_t)}{Re}\boldsymbol{\nabla}\boldsymbol{u} - p\mathbf{I}_{n_{sd}}\right)\boldsymbol{n} \\[2mm]
\boldsymbol{u}\cdot\boldsymbol{n}
\end{cases} & \text{on } \Gamma_S,
\end{cases}
\tag{7a}
$$

and

$$
b := \begin{cases}
\tilde{\nu} - \tilde{\nu}_D & \text{on } \Gamma_D, \\[2mm]
\left(\dfrac{1+\tilde{\nu}f_n}{\sigma Re}\boldsymbol{\nabla}\tilde{\nu}\right)\cdot\boldsymbol{n} & \text{on } \Gamma_N \cup \Gamma_O \cup \Gamma_S,
\end{cases}
\tag{7b}
$$

where $\boldsymbol{n}$ is the outward unit normal and $\boldsymbol{t}_k$, for $k = 1, \ldots, n_{sd}-1$, are unit tangential vectors to the boundary (one vector in two dimensions and two vectors in three dimensions), $\boldsymbol{u}_D$ and $\tilde{\nu}_D$ are the imposed velocity and turbulent variable on the Dirichlet boundary, and $\boldsymbol{g}$ is the imposed traction on the Neumann boundary representing a material surface. A numerical comparison of the outflow boundary condition considered here and other boundary conditions imposed on an outlet can be found in [21].
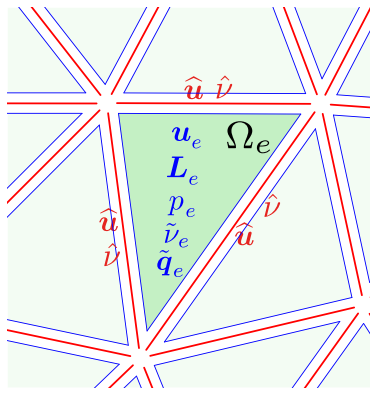
Fig. 1. Detail of a triangular mesh highlighting a cell Ω_e where the velocity $\boldsymbol{u}_e$, velocity gradient $\boldsymbol{L}_e$, mean pressure p_e, turbulent variable $\tilde{\nu}_e$ and gradient of the turbulent variable $\tilde{\boldsymbol{q}}_e$ are defined and internal edges where the hybrid velocity $\hat{\boldsymbol{u}}$ and hybrid turbulent variable $\hat{\nu}$ are defined.

Finally, an initial condition is given for the velocity and turbulent variable, namely

$$\begin{cases} \boldsymbol{u} = \boldsymbol{u}_0 & \text{in } \Omega \times \{0\}, \\ \tilde{\nu} = \tilde{\nu}_0 & \text{in } \Omega \times \{0\}. \end{cases} \tag{8}$$

It is worth noting that the divergence-free condition in the RANS equations induces the compatibility condition on the velocity field given by

$$\int_{\Gamma_D} \boldsymbol{u}_D \cdot \boldsymbol{n} \, d\Gamma + \int_{\partial\Omega \setminus \Gamma_D} \boldsymbol{u} \cdot \boldsymbol{n} \, d\Gamma = 0. \tag{9}$$

In particular, if only Dirichlet boundary conditions are considered it is necessary to impose an extra constraint to eliminate the indeterminacy of the pressure field [21].

3. Face-centred finite volume formulation

Let us consider a partition of the domain Ω in a set of non-overlapping cells $\{\Omega_e\}_{e=1,\dots,n_{el}}$, where n_{el} is the number of cells. The boundary of each cell, $\partial\Omega_e$, is formed by a set of faces $\{\Gamma_{e,j}\}_{j=1,\dots,n_{fa}^e}$, where n_{fa}^e is the number of faces of cell Ω_e. Finally the set of faces not on the boundary of the domain form the internal interface Γ. As shown in previous works [5,13], the FCFV allows for various cell types and hybrid meshes with cells of different types. Examples using triangular, quadrilateral, tetrahedral, hexahedral, pyramidal and prismatic cells, as well as hybrid meshes, have been previously considered in [5,13].

Following the rationale of FCFV [5,11–13,16] and HDG [24–27] methods, the incompressible RANS equations together with the SA turbulence model are written as a system of first-order equations after introducing the mixed variables $\boldsymbol{L} = -\boldsymbol{\nabla}\boldsymbol{u}$ and $\tilde{\boldsymbol{q}} = -\boldsymbol{\nabla}\tilde{\nu}$. This mixed formulation is solved in two steps. First, n_{el} *local problems* are defined, one per cell, to express the primal variables, $\boldsymbol{u}$, p and $\tilde{\nu}$, and the mixed variables, $\boldsymbol{L}$ and $\tilde{\boldsymbol{q}}$, as a function of new, independent, face variables, $\hat{\boldsymbol{u}}$ and $\hat{\nu}$. In addition, Dirichlet boundary conditions are enforced in the local problems, for the cells touching the portion Γ_D of the boundary $\partial\Omega$. Second, a *global problem* is defined to impose the inter-cell continuity of the velocity, the turbulent variable and the normal fluxes of the RANS and SA equations, across the internal interface Γ. In addition, non-Dirichlet boundary conditions on $\partial\Omega \setminus \Gamma_D$ are also accounted for in the global problem.

Fig. 1 shows a sketch of a triangular mesh with the variables defined on each cell and on each edge. As it will be shown in this section, the global system of equations to be solved only involves the variables on the edges and the cell variables can then be recovered cell-by-cell.

3.1. Integral form of the local problems

The local problems enforce, in each cell, the kinematic equations defining the mixed variables $\boldsymbol{L}$ and $\tilde{\boldsymbol{q}}$, the RANS momentum equation and the SA equation. Thus, conservation of mass and momentum is imposed cell-by-cell. The integral form of the local problems for each cell is

$$\begin{cases} -\displaystyle\int_{\Omega_e} \boldsymbol{L} \, d\Omega = \int_{\partial\Omega_e \cap \Gamma_D} \boldsymbol{u}_D \otimes \boldsymbol{n} \, d\Gamma + \int_{\partial\Omega_e \setminus \Gamma_D} \hat{\boldsymbol{u}} \otimes \boldsymbol{n} \, d\Gamma, \\[10pt] \displaystyle\int_{\Omega_e} \frac{\partial \boldsymbol{u}}{\partial t} \, d\Omega + \int_{\partial\Omega_e} \left(\widehat{\boldsymbol{u} \otimes \boldsymbol{u}}\right) \boldsymbol{n} \, d\Gamma + \int_{\Omega_e} \boldsymbol{\nabla} \cdot \left(\frac{1+\nu_t}{Re} \boldsymbol{L} + p\mathbf{I}_{n_{sd}}\right) d\Omega \\[10pt] \qquad + \displaystyle\int_{\partial\Omega_e} \left[\left(\overline{\frac{1+\nu_t}{Re} \boldsymbol{L} + p\mathbf{I}_{n_{sd}}}\right) - \left(\frac{1+\nu_t}{Re} \boldsymbol{L} + p\mathbf{I}_{n_{sd}}\right)\right] \boldsymbol{n} \, d\Gamma \\[10pt] \qquad\qquad\qquad - \displaystyle\int_{\Omega_e} \frac{1}{Re} \boldsymbol{L}^T \boldsymbol{q}_t \, d\Omega = \int_{\Omega_e} \boldsymbol{s} \, d\Omega, \\[10pt] \displaystyle\int_{\Omega_e} p \, d\Omega = |\Omega_e| \, \rho_e, \\[10pt] -\displaystyle\int_{\Omega_e} \tilde{\boldsymbol{q}} \, d\Omega = \int_{\partial\Omega_e \cap \Gamma_D} \tilde{\nu}_D \boldsymbol{n} \, d\Gamma + \int_{\partial\Omega_e \setminus \Gamma_D} \hat{\nu} \boldsymbol{n} \, d\Gamma, \\[10pt] \displaystyle\int_{\Omega_e} \frac{\partial \tilde{\nu}}{\partial t} \, d\Omega + \int_{\partial\Omega_e} (\widehat{\tilde{\nu}\boldsymbol{u}}) \cdot \boldsymbol{n} \, d\Gamma + \int_{\Omega_e} \boldsymbol{\nabla} \cdot \left(\frac{1+\tilde{\nu}f_n}{\sigma Re} \tilde{\boldsymbol{q}}\right) d\Omega, \\[10pt] \qquad + \displaystyle\int_{\partial\Omega_e} \left(\overline{\frac{1+\tilde{\nu}f_n}{\sigma Re} \tilde{\boldsymbol{q}}} - \frac{1+\tilde{\nu}f_n}{\sigma Re} \tilde{\boldsymbol{q}}\right) \cdot \boldsymbol{n} \, d\Gamma = \int_{\Omega_e} s \, d\Omega, \end{cases} \tag{10}$$

where $\hat{\boldsymbol{u}}$ and $\hat{\nu}$, known as *hybrid* variables, denote the face velocity and turbulent variable on $\Gamma \cup \partial\Omega \setminus \Gamma_D$, ρ_e is the mean pressure in the cell and $\boldsymbol{q}_t := -\boldsymbol{\nabla}\nu_t = (4 - 3f_{v1})f_{v1}\tilde{\boldsymbol{q}}$. Moreover, the right-hand side of the SA model, introduced in (2), is also written in terms of the mixed variable $\tilde{\boldsymbol{q}}$ as

$$s = \begin{cases} c_{b1}(1-f_{t2})\tilde{S}\tilde{\nu} + \dfrac{c_{b2}}{\sigma Re}\tilde{\boldsymbol{q}}\cdot\tilde{\boldsymbol{q}} - \dfrac{1}{Re}\left(c_{w1}f_w - \dfrac{c_{b1}}{\kappa^2}f_{t2}\right)\left(\dfrac{\tilde{\nu}}{d}\right)^2 & \text{if } \tilde{\nu} \geq 0, \\[10pt] c_{b1}(1-c_{t3})S\tilde{\nu} + \dfrac{c_{b2}}{\sigma Re}\tilde{\boldsymbol{q}}\cdot\tilde{\boldsymbol{q}} + \dfrac{c_{w1}}{Re}\left(\dfrac{\tilde{\nu}}{d}\right)^2 & \text{otherwise.} \end{cases} \tag{11}$$

Note that the second equation in (10) uses the rewriting of the momentum equation given by Remark 1.

In the local problems, the divergence theorem is applied twice to the diffusive terms of both the momentum and SA equations. This implies that the boundary integrals feature the difference between physical and numerical fluxes. This is simply done to guarantee the symmetry of the global system in the case of creeping flows [21,28], i.e., for Stokes problems.

As the local problems only contain Dirichlet boundary conditions, it is necessary to introduce an extra constraint to eliminate the indeterminacy of the pressure field. In this work, the constraint imposes the mean value of the pressure in each cell, as reported in the third equation in (10).

Therefore, the local problems provide an expression to compute $(\boldsymbol{L}, \boldsymbol{u}, p, \tilde{\boldsymbol{q}}, \tilde{\nu})$, in each cell, in terms of the global unknowns $(\hat{\boldsymbol{u}}, \boldsymbol{\rho}, \hat{\nu})$, where $\boldsymbol{\rho} = (\rho_1 \dots, \rho_{n_{el}})^T$.

3.2. Integral form of the global problem

The global problem imposes the continuity of the inter-cell fluxes of the RANS and SA equations and the non-Dirichlet boundary conditions.

The integral form of the global problem is

$$
\begin{cases}
\displaystyle\sum_{e=1}^{n_{el}}\left\{\int_{\partial\Omega_e\setminus\partial\Omega}\widehat{(\boldsymbol{u}\otimes\boldsymbol{u})}\boldsymbol{n}\,d\Gamma+\int_{\partial\Omega_e\setminus\partial\Omega}\left(\overline{\frac{1+v_t}{Re}\boldsymbol{L}+p\mathbf{I}_{n_{sd}}}\right)\boldsymbol{n}\,d\Gamma\right. \\[2mm]
\hspace{4cm}\left.+\int_{\partial\Omega_e\cap\partial\Omega\setminus\Gamma_{\tilde{D}}}\boldsymbol{B}(\boldsymbol{u},p,\boldsymbol{L},\tilde{v})d\Gamma\right\}=\boldsymbol{0}, \\[4mm]
\displaystyle\int_{\partial\Omega_e\setminus\Gamma_{\tilde{D}}}\hat{\boldsymbol{u}}\cdot\boldsymbol{n}\,d\Gamma=-\int_{\partial\Omega_e\cap\Gamma_{\tilde{D}}}\boldsymbol{u}_D\cdot\boldsymbol{n}\,d\Gamma\qquad\text{for }e=1,\dots,n_{el}, \\[4mm]
\displaystyle\sum_{e=1}^{n_{el}}\left\{\int_{\partial\Omega_e\setminus\partial\Omega}\widehat{(\tilde{v}\boldsymbol{u})}\cdot\boldsymbol{n}\,d\Gamma+\int_{\partial\Omega_e\setminus\partial\Omega}\left(\overline{\frac{1+\tilde{v}f_n}{\sigma Re}\tilde{\boldsymbol{q}}}\right)\cdot\boldsymbol{n}\,d\Gamma\right. \\[2mm]
\hspace{4cm}\left.+\int_{\partial\Omega_e\cap\partial\Omega\setminus\Gamma_{\tilde{D}}}b(\boldsymbol{u},\tilde{v},\tilde{\boldsymbol{q}})d\Gamma\right\}=0,
\end{cases}
\tag{12}
$$

where the second equation represents the compatibility condition enforcing the divergence-free condition on the velocity in each cell.

The continuity of the velocity and the turbulent variable across the internal interface Γ is not explicitly imposed in the global problem because it is automatically satisfied due to the Dirichlet boundary conditions $\boldsymbol{u}=\hat{\boldsymbol{u}}$ and $\tilde{v}=\hat{v}$ on $\partial\Omega_e\setminus\Gamma_{\tilde{D}}$ in the local problems and the unique value of the hybrid variables $\hat{\boldsymbol{u}}$ and $\hat{v}$ on each face.

3.3. Numerical fluxes and stabilisation

In the local and global problems, the numerical trace of the convective and diffusive fluxes in the RANS equations is given by

$$
\widehat{(\boldsymbol{u}\otimes\boldsymbol{u})}\boldsymbol{n}:=\begin{cases}(\boldsymbol{u}_D\otimes\boldsymbol{u}_D)\boldsymbol{n}+\tau^a(\boldsymbol{u}-\boldsymbol{u}_D)&\text{on }\partial\Omega_e\cap\Gamma_D,\\(\hat{\boldsymbol{u}}\otimes\hat{\boldsymbol{u}})\boldsymbol{n}+\tau^a(\boldsymbol{u}-\hat{\boldsymbol{u}})&\text{elsewhere,}\end{cases}
\tag{13a}
$$

and

$$
\overline{\left(\frac{1+v_t}{Re}\boldsymbol{L}+p\mathbf{I}_{n_{sd}}\right)}\boldsymbol{n}:=\begin{cases}\left(\dfrac{1+\tilde{v}_Df_{v1}(\tilde{v}_D)}{Re}\boldsymbol{L}+p\mathbf{I}_{n_{sd}}\right)\boldsymbol{n}+\tau^d(\boldsymbol{u}-\boldsymbol{u}_D)&\text{on }\partial\Omega_e\cap\Gamma_D,\\[3mm]\left(\dfrac{1+\hat{v}f_{v1}(\hat{v})}{Re}\boldsymbol{L}+p\mathbf{I}_{n_{sd}}\right)\boldsymbol{n}+\tau^d(\boldsymbol{u}-\hat{\boldsymbol{u}})&\text{elsewhere,}\end{cases}
\tag{13b}
$$

respectively. Similarly, the numerical trace of the convective and diffusive fluxes in the SA equation is given by

$$
\widehat{(\tilde{v}\boldsymbol{u})}\cdot\boldsymbol{n}:=\begin{cases}(\tilde{v}_D\boldsymbol{u}_D)\cdot\boldsymbol{n}+\tilde{\tau}^a(\tilde{v}-\tilde{v}_D)&\text{on }\partial\Omega_e\cap\Gamma_D,\\(\hat{v}\hat{\boldsymbol{u}})\cdot\boldsymbol{n}+\tilde{\tau}^a(\tilde{v}-\hat{v})&\text{elsewhere,}\end{cases}
\tag{14a}
$$

and

$$
\overline{\left(\frac{1+\tilde{v}f_n}{\sigma Re}\tilde{\boldsymbol{q}}\right)}\cdot\boldsymbol{n}:=\begin{cases}\left(\dfrac{1+\tilde{v}_Df_n(\tilde{v}_D)}{\sigma Re}\tilde{\boldsymbol{q}}\right)\cdot\boldsymbol{n}+\tilde{\tau}^d(\tilde{v}-\tilde{v}_D)&\text{on }\partial\Omega_e\cap\Gamma_D,\\[3mm]\left(\dfrac{1+\hat{v}f_n(\hat{v})}{\sigma Re}\tilde{\boldsymbol{q}}\right)\cdot\boldsymbol{n}+\tilde{\tau}^d(\tilde{v}-\hat{v})&\text{elsewhere,}\end{cases}
\tag{14b}
$$

respectively.

An appropriate stabilisation, given by τ^a, τ^d, $\tilde{\tau}^a$ and $\tilde{\tau}^d$, is required in order to ensure that the numerical scheme is well posed. The effect of the stabilisation has been extensively studied in the literature for HDG [9,20,23,25,29,30] and FCFV [5,11–13,16]. The diffusive stabilisation is taken as

$$
\boldsymbol{\tau}^d:=\frac{\beta\,(1+\hat{v}f_{v1}(\hat{v}))}{Re}\mathbf{I}_{n_{sd}}\quad\text{and}\quad\tilde{\tau}^d:=\frac{\beta\,(1+\hat{v}f_n(\hat{v}))}{\sigma Re},
\tag{15}
$$

where β is a numerical parameter, taken as 10 in all the examples presented in this work.

For the convective stabilisation, new definitions are proposed in this work. The derivation of three different convective stabilisations, based on the Lax–Friedrichs (LF), Roe and Harten-Lax-van Leer (HLL) Riemann solvers, is detailed in Appendix A, following the ideas presented in [16,17,23] for compressible flows.

4. Face-centred finite volume solution

To simplify the presentation, the set of indices for the faces of a cell is denoted by $\mathcal{A}_e:=\{1,\dots,n_{fa}^e\}$. In addition, the set of indices for the faces of a cell on the Dirichlet boundary, interior to the domain and on the boundary of the domain are denoted by $\mathcal{D}_e:=\{j\in\mathcal{A}_e\mid\Gamma_{e,j}\cap\Gamma_D\neq\emptyset\}$, $\mathcal{I}_e:=\{j\in\mathcal{A}_e\mid\Gamma_{e,j}\cap\partial\Omega=\emptyset\}$ and $\mathcal{E}_e:=\{j\in\mathcal{A}_e\mid\Gamma_{e,j}\cap\partial\Omega\neq\emptyset\}$, respectively. Given the imposition of Dirichlet boundary conditions in the local problems, it is also convenient to denote by $\mathcal{B}_e:=\mathcal{A}_e\setminus\mathcal{D}_e$ and $\mathcal{C}_e:=\mathcal{E}_e\setminus\mathcal{D}_e$ the set of indices for the faces of a cell not on a Dirichlet boundary and for the external faces not on a Dirichlet boundary. Finally, it is useful to introduce the indicator function of a set S, i.e.

$$
\chi_S(i)=\begin{cases}1&\text{if }i\in S,\\0&\text{otherwise.}\end{cases}
\tag{16}
$$

4.1. Spatial and temporal discretisation

The FCFV method introduces a constant approximation of the primal, $\boldsymbol{u}$, p, and $\tilde{v}$, and mixed, $\boldsymbol{L}$ and $\tilde{\boldsymbol{q}}$, variables in each cell as well as a constant approximation of the hybrid variables, $\hat{\boldsymbol{u}}$ and $\hat{v}$ on each cell face. The value of the primal and mixed variables in each cell is denoted by $\mathbf{u}_e$, ρ_e, $\tilde{v}_e$, $\boldsymbol{L}_e$ and $\tilde{\mathbf{q}}_e$ respectively, whereas the value of the hybrid variables on the jth face of a cell is denoted by $\hat{\mathbf{u}}_j$ and $\hat{v}_j$.

The time integration is performed in this work using implicit multi-step backward differentiation formulae (BDF) [31]. The first-order time derivative in the local problems is approximated as

$$
\left.\frac{\partial y}{\partial t}\right|^n\approx\sum_{s=0}^{n_s}a_s y^{n-s},
\tag{17}
$$

where $y^r(\boldsymbol{x}):=y(\boldsymbol{x},t^r)$, n_s is the number of stages and, to simplify the notation, the coefficients a_s include the dependence on the time step Δt.

For steady-state computations, the first-oder BDF method (BDF1) is employed as a pseudo-time marching scheme, which corresponds to $n_s=1$ with $a_0=1/\Delta t$ and $a_1=-1/\Delta t$. For transient simulations the second-oder BDF method (BDF2) is employed, which corresponds to $n_s=2$ with $a_0=3/(2\Delta t)$, $a_1=-2/\Delta t$ and $a_2=1/(2\Delta t)$.

Introducing the expression of the numerical fluxes given by Eqs. (13) and (14) into the local problem (10) and the approximation of the time derivative defined in (17), the fully discrete residuals of the local problems are

$$
\begin{cases}
\mathbf{R}_{e,L}^n:=|\Omega_e|\mathbf{L}_e^n+\displaystyle\sum_{j\in\mathcal{D}_e}|\Gamma_{e,j}|\boldsymbol{u}_{D,j}^n\otimes\boldsymbol{n}_j+\sum_{j\in\mathcal{B}_e}|\Gamma_{e,j}|\hat{\mathbf{u}}_j^n\otimes\boldsymbol{n}_j, \\[3mm]
\mathbf{R}_{e,u}^n:=|\Omega_e|\displaystyle\sum_{s=0}^{n_s}a_s\mathbf{u}_e^{n-s}+\sum_{j\in\mathcal{A}_e}|\Gamma_{e,j}|\boldsymbol{\tau}_j^{n-1}\mathbf{u}_e^n \\[3mm]
\hspace{1cm}-\dfrac{|\Omega_e|}{Re}(\mathbf{L}_e^n+[\mathbf{L}_e^n]^T)\mathbf{q}_{t,e}(\tilde{v}_e^n,\tilde{\mathbf{q}}_e^n)-|\Omega_e|\mathbf{s}_e^n \\[3mm]
\hspace{1cm}-\displaystyle\sum_{j\in\mathcal{D}_e}|\Gamma_{e,j}|(\boldsymbol{\tau}_j^{n-1}-(\boldsymbol{u}_{D,j}^n\cdot\boldsymbol{n}_j)\mathbf{I}_{n_{sd}})\boldsymbol{u}_{D,j}^n \\[3mm]
\hspace{1cm}-\displaystyle\sum_{j\in\mathcal{B}_e}|\Gamma_{e,j}|(\boldsymbol{\tau}_j^{n-1}-(\hat{\boldsymbol{u}}_j^n\cdot\boldsymbol{n}_j)\mathbf{I}_{n_{sd}})\hat{\mathbf{u}}_j^n, \\[3mm]
\mathbf{R}_{e,\tilde{q}}^n:=|\Omega_e|\tilde{\mathbf{q}}_e^n+\displaystyle\sum_{j\in\mathcal{D}_e}|\Gamma_{e,j}|\hat{v}_{D,j}^n\boldsymbol{n}_j+\sum_{j\in\mathcal{B}_e}|\Gamma_{e,j}|\hat{v}_j^n\boldsymbol{n}_j, \\[3mm]
\mathrm{R}_{e,\tilde{v}}^n:=|\Omega_e|\displaystyle\sum_{s=0}^{n_s}a_s\tilde{v}_e^{n-s}+\sum_{j\in\mathcal{A}_e}|\Gamma_{e,j}|\tilde{\tau}_j^{n-1}\tilde{v}_e^n \\[3mm]
\hspace{1cm}-\dfrac{|\Omega_e|}{\sigma Re}\tilde{\mathbf{q}}_e^n\cdot\tilde{\mathbf{q}}_e^n f_{n,e}(\tilde{v}_e^n)-\dfrac{|\Omega_e|}{\sigma Re}\tilde{\mathbf{q}}_e^n\cdot\tilde{\mathbf{q}}_e^n\tilde{v}_e^n\dfrac{\partial f_{n,e}(\tilde{v}_e^n)}{\partial\tilde{v}_e^n}-|\Omega_e|s_e(\tilde{v}_e^n,\tilde{\mathbf{q}}_e^n,\hat{\mathbf{u}}^n) \\[3mm]
\hspace{1cm}-\displaystyle\sum_{j\in\mathcal{D}_e}|\Gamma_{e,j}|(\tilde{\tau}_j^{n-1}-\boldsymbol{u}_{D,j}^n\cdot\boldsymbol{n}_j)\tilde{v}_{D,j}^n-\sum_{j\in\mathcal{B}_e}|\Gamma_{e,j}|(\tilde{\tau}_j^{n-1}-\hat{\mathbf{u}}_j^n\cdot\boldsymbol{n}_j)\hat{v}_j^n,
\end{cases}
\tag{18}
$$

where $\tau=\tau^a+\tau^d$, $\tilde{\tau}=\tilde{\tau}^a+\tilde{\tau}^d$, $|\Omega_e|$ is the area/volume of the cell Ω_e and $|\Gamma_{e,j}|$ is the length/area of the edge/face $\Gamma_{e,j}$.
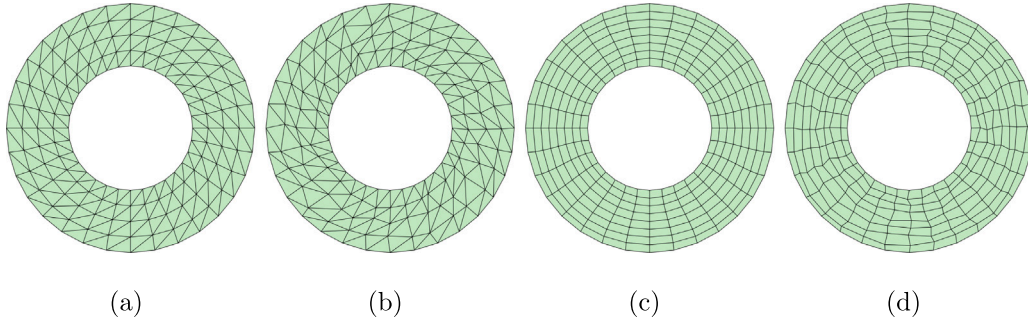
Fig. 2. Couette flow: (a, b) Triangular and (c, d) quadrilateral meshes used for the convergence study.

Remark 2. The right-hand side term s_e in the last equation of the local problem (18) requires the vorticity magnitude S. The vorticity can be computed from the mixed variable matrix $\mathbf{L}$ as

$$\mathbf{S}_e = (\mathbf{L}_e^T - \mathbf{L}_e)/2, \tag{19}$$

or in terms of the hybrid velocity $\hat{u}$ as

$$\mathbf{S}_e = \frac{1}{|\Omega_e|} \left(\sum_{j \in \mathcal{D}_e} |\Gamma_{e,j}|(\boldsymbol{u}_{D,j} \otimes \boldsymbol{n}_j - \boldsymbol{n}_j \otimes \boldsymbol{u}_{D,j}) + \sum_{j \in \mathcal{B}_e} |\Gamma_{e,j}|(\hat{\boldsymbol{u}}_j \otimes \boldsymbol{n}_j - \boldsymbol{n}_j \otimes \hat{\boldsymbol{u}}_j) \right). \tag{20}$$

The two expressions are equivalent and numerical tests have shown to provide an approximation that converges linearly as the mesh is refined. In all the numerical examples involving turbulent flows, the simpler expression given by Eq. (19) is employed.

The residuals of the global problem, denoted by $\mathbf{R}_{\hat{u}}$, $\mathbf{R}_\rho$ and $\mathbf{R}_{\hat{v}}$, are obtained by assembling the contributions

$$
\begin{cases}
\mathbf{R}_{e,i,\hat{u}}^n := |\Gamma_{e,i}| \Bigg\{ \left(\boldsymbol{\tau}_i^{n-1} \mathbf{u}_e^n + \frac{1+\hat{v}_i^n f_{v1}(\hat{v}_i^n)}{Re} \mathbf{L}_e^n \boldsymbol{n}_i + \rho_e^n \boldsymbol{n}_i - \tau_i^{n-1} \hat{\mathbf{u}}_i^n \right) \chi_{\mathcal{I}_e}(i) \\
\qquad\qquad + \hat{\mathbf{B}}_i(\mathbf{u}_e^n, \hat{\mathbf{u}}_i^n, \mathbf{L}_e^n, \rho_e^n, \hat{v}_i^n, \tau_i^{n-1}) \chi_{C_e}(i) \Bigg\}, \\
\mathbf{R}_{e,\rho}^n := \sum_{j \in \mathcal{B}_e} |\Gamma_{e,j}| \hat{\mathbf{u}}_j^n \cdot \boldsymbol{n}_j + \sum_{j \in \mathcal{D}_e} |\Gamma_{e,j}| \boldsymbol{u}_{D,j}^n \cdot \boldsymbol{n}_j, \\
\mathbf{R}_{e,i,\hat{v}}^n := |\Gamma_{e,i}| \Bigg\{ \left(\tilde{\tau}_i^{n-1} \tilde{v}_e^n + \frac{1+\hat{v}_i^n f_n(\hat{v}_i^n)}{\sigma Re} \tilde{\mathbf{q}}_e^n \cdot \boldsymbol{n}_i - \tilde{\tau}_i^{n-1} \hat{v}_i^n \right) \chi_{\mathcal{I}_e}(i) \\
\qquad\qquad + \hat{\mathbf{b}}_i(\tilde{v}_e^n, \hat{v}_i^n, \tilde{\mathbf{q}}_e^n, \tilde{\tau}_i^{n-1}) \chi_{C_e}(i) \Bigg\}.
\end{cases}
\tag{21}
$$

for all $i \in \mathcal{B}_e$, where the FCFV version of the boundary operators is given by

$$
\hat{\mathbf{B}} := \begin{cases}
\left(\frac{1+\hat{v} f_{v1}(\hat{v})}{Re}(\mathbf{L}+\mathbf{L}^T)+\rho \mathbf{I}_{\mathrm{n_{sd}}} \right)\mathbf{n}+\boldsymbol{\tau}(\mathbf{u}-\hat{\mathbf{u}})+\mathbf{g} & \text{on } \Gamma_N, \\
\left(\frac{1+\hat{v} f_{v1}(\hat{v})}{Re}\mathbf{L}+\rho \mathbf{I}_{\mathrm{n_{sd}}} \right)\mathbf{n}+\tau^d(\mathbf{u}-\hat{\mathbf{u}}) & \text{on } \Gamma_O, \\
\left\{ \begin{array}{l} \mathbf{t}_k \cdot \left[\left(\frac{1+\hat{v} f_{v1}(\hat{v})}{Re}\mathbf{L}+\rho \mathbf{I}_{\mathrm{n_{sd}}} \right)\mathbf{n}+\tau^d(\mathbf{u}-\hat{\mathbf{u}}) \right] \\ \hat{\mathbf{u}} \cdot \mathbf{n} \end{array} \right\} & \text{on } \Gamma_S,
\end{cases}
\tag{22a}
$$

and

$$
\hat{\mathbf{b}} := \begin{cases}
\left(\frac{1+\hat{v} f_n(\hat{v})}{\sigma Re} \tilde{\mathbf{q}} \right) \cdot \mathbf{n} + \tilde{\tau}(\tilde{v} - \hat{v}) & \text{on } \Gamma_N, \\
\left(\frac{1+\hat{v} f_n(\hat{v})}{\sigma Re} \tilde{\mathbf{q}} \right) \cdot \mathbf{n} + \tilde{\tau}^d(\tilde{v} - \hat{v}) & \text{on } \Gamma_O \cup \Gamma_S.
\end{cases}
\tag{22b}
$$

As the Dirichlet boundary conditions are accounted for in the local problems, the FCFV boundary operators are only required for Neumann, outflow and symmetry boundaries.

The third equation in (10) is omitted in (18) and the value of p_e in the global residual $\mathbf{R}_{e,i,\hat{u}}$ of Eq. (21) is directly taken as ρ_e.

Remark 3. The residuals in the local and global problems assume the stabilisation to be evaluated at time t^{n-1} to avoid the need to linearise the stabilisation tensors and scalars for the RANS and SA equations, respectively. This choice significantly facilitates the linearisation and,

as it will be shown in the numerical examples, it does not have a noticeable impact in the stability of the time marching scheme.

The resulting FCFV discrete problem consists of solving the nonlinear system of equations

$$\mathbf{R}(\mathbf{L}_e^n, \mathbf{u}_e^n, \dots, \mathbf{u}_e^{n-\mathrm{n_s}}, \tilde{\mathbf{q}}_e^n, \tilde{v}_e^n, \dots, \tilde{v}_e^{n-\mathrm{n_s}}, \hat{\mathbf{u}}^n, \hat{\mathbf{u}}^{n-1}, \rho^n, \hat{v}^n, \hat{v}^{n-1}) = \mathbf{0}, \tag{23}$$

at each time step, where the residual is obtained by assembling the cell contributions

$$
\mathbf{R}_{e,i} := \begin{Bmatrix}
\mathbf{R}_{e,L}(\mathbf{L}_e^n, \hat{\mathbf{u}}^n) \\
\mathbf{R}_{e,u}(\mathbf{L}_e^n, \mathbf{u}_e^n, \dots, \mathbf{u}_e^{n-\mathrm{n_s}}, \tilde{\mathbf{q}}_e^n, \tilde{v}_e^n, \hat{\mathbf{u}}^n, \hat{\mathbf{u}}^{n-1}, \hat{v}^{n-1}) \\
\mathbf{R}_{e,\tilde{q}}(\tilde{\mathbf{q}}_e^n, \hat{v}^n) \\
\mathbf{R}_{e,\tilde{v}}(\tilde{\mathbf{q}}_e^n, \tilde{v}_e^n, \dots, \tilde{v}_e^{n-\mathrm{n_s}}, \hat{\mathbf{u}}^n, \hat{\mathbf{u}}^{n-1}, \hat{v}^n, \hat{v}^{n-1}) \\
\mathbf{R}_{e,i,\hat{u}}(\mathbf{L}_e^n, \mathbf{u}_e^n, \rho_e^n, \hat{\mathbf{u}}^n, \hat{\mathbf{u}}^{n-1}, \hat{v}^n, \hat{v}^{n-1}) \\
\mathbf{R}_{e,\rho}(\hat{\mathbf{u}}^n) \\
\mathbf{R}_{e,i,\hat{v}}(\tilde{\mathbf{q}}_e^n, \tilde{v}_e^n, \hat{\mathbf{u}}^{n-1}, \hat{v}^n, \hat{v}^{n-1})
\end{Bmatrix}, \tag{24}
$$

for $e = 1, \dots, \mathrm{n_{el}}$ and $i \in \mathcal{B}_e$.

Following Remark 3, the dependence of the residuals on $\hat{\mathbf{u}}^{n-1}$ and $\hat{v}^{n-1}$, used to compute the stabilisation of the RANS and SA equations, is explicitly stated.

Remark 4. To simplify the presentation, an abuse of notation is introduced in Eq. (24), where $\mathbf{R}_{e,L}$ and $\mathbf{L}_e^n$ denote the vectorised version of the matrices appearing in Eqs. (18) and (21).

4.2. Newton–Raphson linearisation

The Newton–Raphson method is applied to linearise the residual of Eq. (23). After truncating the Taylor expansion at first order, the linear system to be solved at each Newton–Raphson iteration, m, can be written as

$$\begin{bmatrix} \mathbf{T}_{UU} & \mathbf{T}_{U\Lambda} \\ \mathbf{T}_{\Lambda U} & \mathbf{T}_{\Lambda\Lambda} \end{bmatrix}^{n,m} \begin{Bmatrix} \Delta\mathbf{U} \\ \Delta\Lambda \end{Bmatrix}^{n,m} = -\begin{Bmatrix} \mathbf{R}_U \\ \mathbf{R}_\Lambda \end{Bmatrix}^{n,m}, \tag{25}$$

where

$$
\mathbf{T}_{UU} = \begin{bmatrix} \mathbf{T}_{LL} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{T}_{uL} & \mathbf{T}_{uu} & \mathbf{T}_{u\tilde{q}} & \mathbf{T}_{u\tilde{v}} \\ \mathbf{0} & \mathbf{0} & \mathbf{T}_{\tilde{q}\tilde{q}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{T}_{\tilde{v}\tilde{q}} & \mathbf{T}_{\tilde{v}\tilde{v}} \end{bmatrix}, \quad
\mathbf{T}_{U\Lambda} = \begin{bmatrix} \mathbf{T}_{L\hat{u}} & \mathbf{0} & \mathbf{0} \\ \mathbf{T}_{u\hat{u}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{T}_{\tilde{q}\hat{v}} \\ \mathbf{0} & \mathbf{0} & \mathbf{T}_{\tilde{v}\hat{v}} \end{bmatrix},
$$

$$
\mathbf{T}_{\Lambda U} = \begin{bmatrix} \mathbf{T}_{\hat{u}L} & \mathbf{T}_{\hat{u}u} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{T}_{\hat{v}\tilde{q}} & \mathbf{T}_{\hat{v}\tilde{v}} \end{bmatrix}, \quad
\mathbf{T}_{\Lambda\Lambda} = \begin{bmatrix} \mathbf{T}_{\hat{u}\hat{u}} & \mathbf{T}_{\hat{u}\rho} & \mathbf{T}_{\hat{u}\hat{v}} \\ \mathbf{T}_{\rho\hat{u}} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{T}_{\hat{v}\hat{v}} \end{bmatrix},
$$

$$
\mathbf{U} = \begin{Bmatrix} \mathbf{L} \\ \mathbf{u} \\ \tilde{\mathbf{q}} \\ \tilde{v} \end{Bmatrix}, \quad
\Lambda = \begin{Bmatrix} \hat{\mathbf{u}} \\ \rho \\ \hat{v} \end{Bmatrix}, \quad
\mathbf{R}_U = \begin{Bmatrix} \mathbf{R}_L \\ \mathbf{R}_u \\ \mathbf{R}_{\tilde{q}} \\ \mathbf{R}_{\tilde{v}} \end{Bmatrix}, \quad
\mathbf{R}_\Lambda = \begin{Bmatrix} \mathbf{R}_{\hat{u}} \\ \mathbf{R}_\rho \\ \mathbf{R}_{\hat{v}} \end{Bmatrix}.
$$

$$\tag{26}$$

with $\mathbf{T}_{ab}$ denoting the tangent matrix obtained by assembling the contributions $(\mathbf{T}_{ab})_{e,i} := \partial \mathbf{R}_{e,i,a}/\partial b$ and $\Delta \circledcirc^{n,m} = \circledcirc^{n,m+1} - \circledcirc^{n,m}$ the solution increment from the Newton–Raphson iteration m to $m+1$.

Given the block-diagonal structure of the matrix $\mathbf{T}_{UU}$, the linear sytem (25), to be solved at each Newton–Raphson iteration, can be reduced to involve only the unknowns Λ, namely

$$\mathbb{K}^{n,m} \Delta \Lambda^{n,m} = \mathbb{F}^{n,m}, \tag{27}$$

where $\mathbb{K} = \mathbf{T}_{\Lambda\Lambda} - \mathbf{T}_{\Lambda U}\mathbf{T}_{UU}^{-1}\mathbf{T}_{U\Lambda}$ and $\mathbb{F} = -\mathbf{R}_\Lambda + \mathbf{T}_{\Lambda U}\mathbf{T}_{UU}^{-1}\mathbf{R}_U$.

After solving the reduced system (27), the local problems can be used to compute the solution in each cell as

$$\mathbf{T}_{UU}^{n,m} \Delta \mathbf{U}^{n,m} = -\mathbf{R}_U^{n,m} - \mathbf{T}_{U\Lambda}^{n,m} \Delta \Lambda^{n,m}. \tag{28}$$

It is worth emphasising that the block-diagonal structure of the matrix $\mathbf{T}_{UU}$ enables the solution of a small system of equations, cell-by-cell.

Computational aspects are discussed in detail in Appendix B, including the solution of local and global problems, the use of pseudo-time marching for steady state problems and a staggered approach to handle the turbulence, alternative to the monolithic approach presented in this section. The current implementation is done in Fortran 90. For the readers interested in the details of the FCFV method, a simpler implementation of the HDG method, for linear elliptic problems and using low and higher order approximations, in Matlab [32], is available.

5. Numerical examples

This section presents a set of numerical examples selected to test the optimal approximation properties of the proposed FCFV method and to compare the performance of the different convective stabilisations proposed. Examples include steady and transient cases, both in laminar and turbulent regimes.

When errors with respect to an analytical or reference solutions are reported, E denotes the relative error and $\|\cdot\|_{\mathcal{L}^2}$ denotes the $\mathcal{L}^2(\Omega)$ norm for velocity, pressure and gradient of velocity and the $\mathcal{L}^2(\Gamma)$ norm for the face velocity.

5.1. Couette flow

The first example is the so-called Couette flow, which involves the flow in an annulus with imposed angular velocity on the boundary. The inner and outer radii are $R_i = 1$ and $R_o = 2$, respectively, and the imposed angular velocities are $\Omega_i = 0$ and $\Omega_o = 0.5$, respectively. This example is used to test the optimal rate of convergence of the FCFV for laminar flows under mesh refinement. In polar coordinates, the exact solution is

$$\begin{cases} u_r = 0, \qquad u_\phi = C_1 r + C_2 \dfrac{1}{r}, \\ p = C_1^2 \dfrac{r^2}{2} + 2C_1 C_2 \log(r) - \dfrac{C_2^2}{2r^2} + C, \end{cases} \tag{29}$$

where $C_1 = (\Omega_o R_o^2 - \Omega_i R_i^2)/(R_o^2 - R_i^2)$, $C_2 = (\Omega_i - \Omega_o)R_i^2 R_o^2/(R_o^2 - R_i^2)$ and C is a constant such that the pressure at the outer radius is equal to 1. As the exact solution does not depend upon the viscosity, the Reynolds number is selected as $Re = 1$.

Triangular and quadrilateral structured meshes are considered and the effect of cell distortion on the accuracy of the computations is evaluated. Fig. 2 shows two structured triangular and quadrilateral meshes and the corresponding meshes where the internal nodes have been randomly moved to test the effect of the cell distortion. The strategy used to distort the meshes can be found in [5].

To perform a mesh convergence analysis, the meshes are uniformly refined and the error in all quantities, namely cell velocity, face velocity, gradient of the velocity and pressure is measured. The ith triangular and quadrilateral meshes have $(8\times2^i)\times(8\times2^i)$ cells. The results, reported in Fig. 3 for regular and distorted triangular and quadrilateral meshes, show the optimal convergence of the error for all quantities under mesh

refinement. The computations performed with distorted meshes show a very similar accuracy when compared to the computation on the corresponding regular mesh. It can be observed that a slightly more accurate result on the pressure and the gradient of the velocity is obtained in regular meshes, whereas the accuracy of the velocity is not influenced by the cell distortion. In addition, the effect of cell distortion on the accuracy of the pressure and the gradient of the velocity is more noticeable when using quadrilateral meshes.

The extra accuracy obtained using quadrilateral meshes in this example is simply due to the nature of the analytical solution, only dependent on the radial coordinate, and the selected meshes.

In this example, the HLL convection stabilisation is employed, but further numerical examples not reported for brevity, show that the results with the LF or the Roe stabilisation are almost identical. This is expected due to the diffusion dominated character of the solution.

Due to the simplicity of this example, the steady problem is directly solved in all cases, without marching in pseudo-time. A tolerance of 10^{-12} is imposed for the nonlinear Newton–Raphson iterations and quadratic convergence is observed in all cases. Details about the residuals used to check convergence and an example showing the quadratic convergence can be found in Appendix B.2.

5.2. Lid-driven cavity flow

The second problem is the lid-driven cavity flow, defined in $\Omega = (0,1)^2$. A constant horizontal velocity of magnitude one is imposed in the upper part of the domain, whereas homogeneous Dirichlet boundary conditions are imposed on the rest of the boundary. As no analytical solution is available, the accuracy of the computations is measured against a reference solution computed with the Taylor-Hood (Q2Q1) element and the streamline-upwind Petrov–Galerkin (SUPG) method [10]. The reference solution is computed on a mesh of 700×700 cells that introduces local refinement near the boundaries, with the height of the first cell being 9.6×10^{-5}. It is worth noting that the reference solution corresponds to the so-called leaky cavity, due to the strong imposition of the incompatible boundary conditions at the top left and right corners. With the FCFV approach, the incompatible boundary conditions do not represent an issue because the velocity is imposed at the barycentre of the faces. To account for this discrepancy between the FCFV and reference solutions, the $\mathcal{L}^2$ norm of the error, for all variables, is measured excluding the regions $[0, 0.05] \times [0.95, 1]$ and $[0.95, 1] \times [0.95, 1]$.

This example aims at comparing the accuracy of the different convection stabilisations derived in Appendix A and to assess the influence of the mesh distortion for convection dominated problems. To this end, the Reynolds number is taken as $Re = 1,000$ and regular and distorted triangular meshes are employed. As in the previous example, the steady problem is directly solved, without marching in pseudo-time, with a tolerance of 10^{-12}.

Five meshes are considered for the mesh convergence study. The ith mesh has $(24 \times 2^i) \times (24 \times 2^i) \times 2$ cells, the height of the first layer of boundary cells is h_0/i, where h_0 is taken as 10^{-2}, and the growth ratio that dictates the height of the next layer of cells varies from 1.06 in the coarse mesh to 1.003 in the finest mesh. Fig. 4 shows the first regular and distorted triangular meshes.

The results of the mesh convergence study are depicted in Fig. 5 and show again the expected first-order convergence of the error for all variables, as the mesh is refined. For this example, the Roe and HLL convective stabilisations provide very similar accuracy, whereas the LF stabilisation is less accurate. This is observed for all variables and on both regular and distorted meshes. It is also worth noting that the use of distorted meshes provides accuracy comparable to regular meshes, even for this example where resolving the boundary layers is crucial.

To further assess the accuracy of the results, Fig. 6 displays the profiles of the velocity and pressure fields along the centrelines, using the fourth regular mesh and different convection stabilisations. The

(a) Triangles   (b) Quadrilaterals
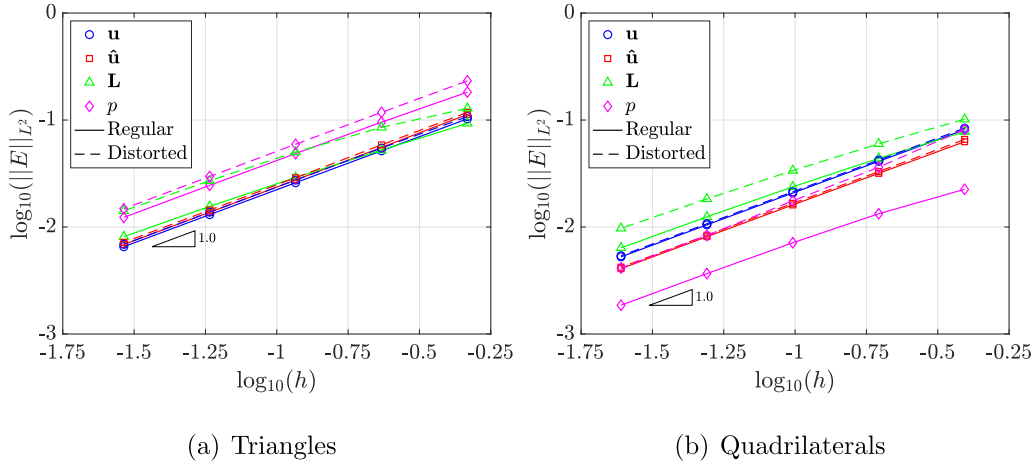
Fig. 3. Couette flow: Mesh convergence of the error of the cell velocity, face velocity, gradient of the velocity and pressure for regular and distorted triangular and quadrilateral meshes using the HLL convection stabilisation.

(a)   (b)
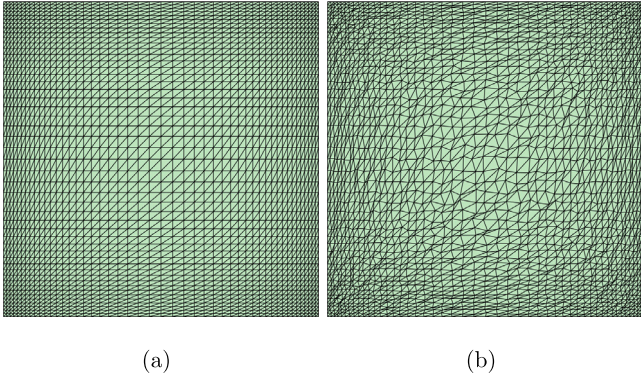
Fig. 4. Lid-driven cavity flow: (a) Regular and (b) distorted triangular meshes used for the convergence study.

Table 1
Relative error on $E(1)$ for three different triangular and quadrilateral meshes and using the three different convective stabilisations.

Mesh	Triangles			Quadrilaterals		
	LF	Roe	HLL	LF	Roe	HLL
1	8.5×10^{-2}	5.6×10^{-2}	2.8×10^{-2}	9.4×10^{-2}	6.6×10^{-2}	3.3×10^{-2}
2	4.6×10^{-2}	3.0×10^{-2}	1.5×10^{-2}	5.2×10^{-2}	3.7×10^{-2}	1.8×10^{-2}
3	2.4×10^{-2}	1.5×10^{-2}	7.7×10^{-3}	2.7×10^{-2}	1.9×10^{-2}	9.8×10^{-3}

results of the proposed FCFV formulation are compared to the reference solution and illustrate the extra accuracy obtained by using the Roe and HLL stabilisations, when compared to the LF stabilisation.

5.3. Assessment of the convective stabilisation

The next example, taken from [33], is used to evaluate the dissipative effect of the three convective stabilisations proposed in Appendix A.

The computational domain is $\Omega = (0, 1)^2$, the manufactured solution is given by

$$\begin{cases} u_1 = t^4 \sin(2\pi x_2)\big(1 - \cos(2\pi x_1)\big), \\ u_2 = -t^4 \sin(2\pi x_1)\big(1 - \cos(2\pi x_2)\big), \\ p = t^4\big(\cos(\pi x_1) + \cos(\pi x_2)\big), \end{cases} \quad (30)$$

and the Reynolds number is $Re = 10^5$. For such a high Reynolds number, the stabilisation of the FCFV is dominated by the convective stabilisation because the diffusive stabilisation is of the order of $1/Re$, as detailed in (15).

The solution is advanced until a final time $T = 1$ using the BDF2 time integrator scheme and $\Delta t = 0.001$. The time step has been selected to ensure that all the error is dominated by the spatial discretisation and a clear comparison of the three different stabilisations can be performed. To assess the dissipative effect of the stabilisation, the quantity

$$E(T) = \int_{\Omega} \boldsymbol{u}(\boldsymbol{x}, T) \cdot \boldsymbol{u}(\boldsymbol{x}, T)\, d\Omega, \quad (31)$$

proportional to the kinetic energy, is evaluated and compared to the exact value, which is 1.5.

Three meshes are considered to perform the study. The ith quadrilateral mesh is a structured mesh of $(32 \times 2^i) \times (32 \times 2^i)$ cells, whereas the ith triangular mesh is obtained after splitting in four triangles each quadrilateral of a mesh with $(16 \times 2^i) \times (16 \times 2^i)$ cells. Therefore, the ith triangular and quadrilateral meshes contain the same number of cells and a further comparison of the dissipative effect of the stabilisation for different element shape can be performed.

Table 1 reports the relative error on $E(1)$ for three different triangular and quadrilateral meshes and using the three different convective stabilisations. The results show that the HLL stabilisation consistently produces the most accurate results, whereas the LF stabilisation always produces the least accurate results. This is observed for all meshes and both element types.

More precisely the error using Roe stabilisation induce an error 1.5 times lower than the error using LF. When using the HLL stabilisation the error is two times lower than the error obtained with the Roe stabilisation. This conclusion is observed for both triangular and quadrilateral cells.

Comparing the element types marginal differences are observed, with the triangles providing a slightly better accuracy for all meshes and all three stabilisations.

5.4. Unsteady laminar flow past a circular cylinder

The next example considers the unsteady flow past a circular cylinder of diameter D at $Re = 100$. This classical benchmark is used to assess the performance of the proposed method with different convective stabilisations for a transient laminar flow. The setup of the problem is depicted in Fig. 7, including the relevant dimensions and the boundary conditions. The Dirichlet boundary consists of the inflow boundary, where a constant horizontal velocity of magnitude one is imposed, and the cylinder, where a no-slip condition is enforced.
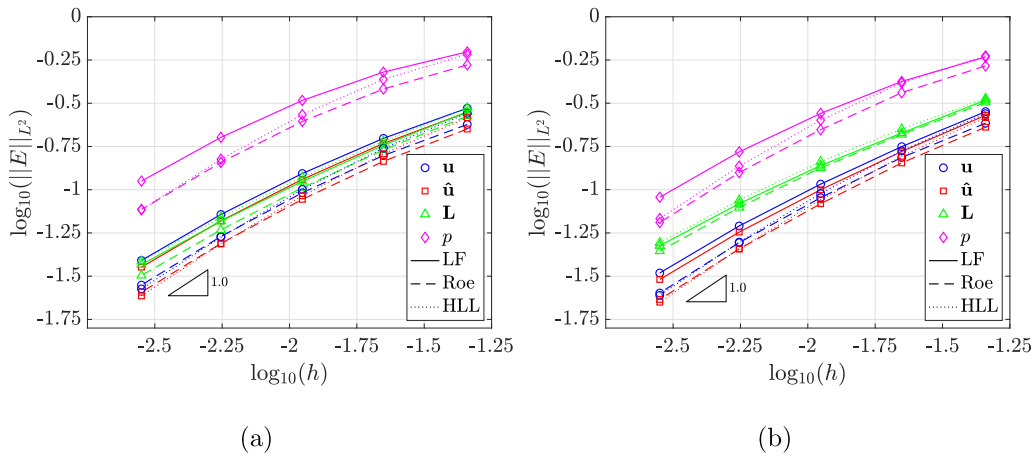
Fig. 5. Lid-driven cavity flow: Mesh convergence of the error of the cell velocity, face velocity, gradient of the velocity and pressure for (a) regular and (b) distorted triangular meshes using different convection stabilisations.
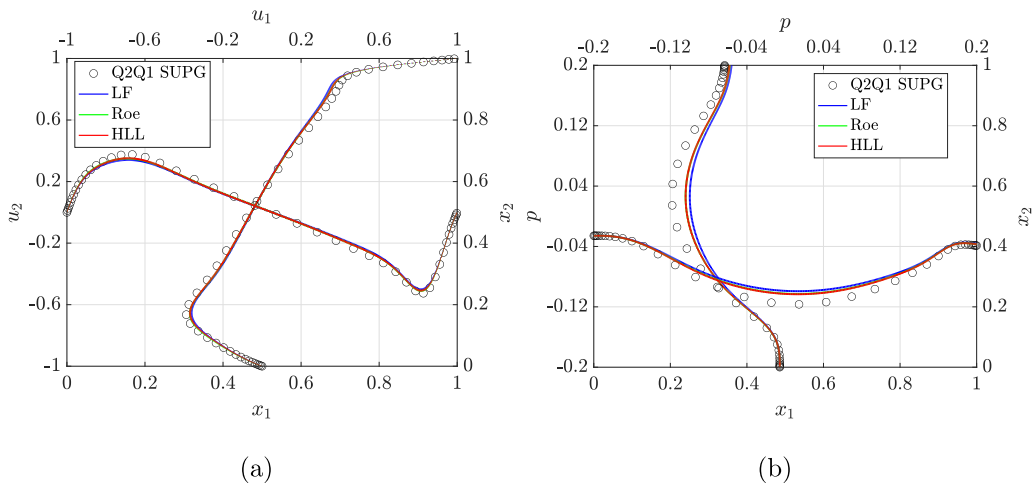

Fig. 6. Lid-driven cavity flow: Profiles of (a) velocity and (b) pressure along the centrelines, using the fourth regular mesh and different convection stabilisations.
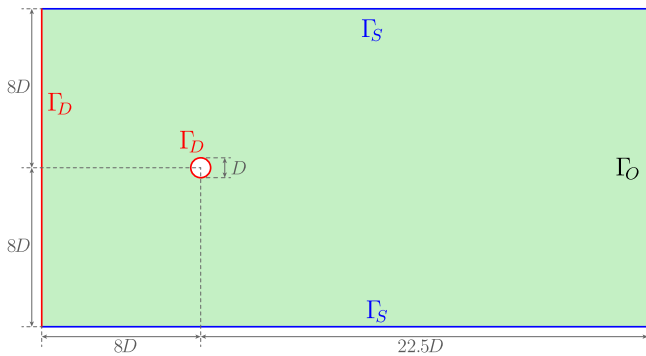

Fig. 7. Unsteady laminar flow past a circular cylinder: Problem setup.

Two unstructured triangular meshes are considered. For the first mesh, the inflation layer is defined using a height for the first cell on the cylinder equal to $0.01D$. A refinement behind the cylinder is introduced to capture the wake by imposing a cell size of $0.03D$ near the cylinder and $0.075D$ near the outflow boundary. For the second mesh, the height of the first cell in the inflation layer is $0.005D$ and the sizes to locally refine the region of the wake are $0.02D$ near the cylinder and $0.05D$ near the outflow. In both cases the growth ratio within the inflation layer is 1.1 and the maximum cell stretching near the cylinder is 10.

The first mesh has 73,251 cells whereas the second mesh, shown in Fig. 8, has 134,801 cells.

For the time-integration, the BDF2 scheme is employed with a time-step $\Delta t = 0.1$, and the initial condition is taken as the steady-state solution with $Re = 10$.

Fig. 9 shows a snapshot of the magnitude of the velocity and pressure fields at time $t = 65$, when the solution has reached a periodic state, illustrating the ability of the proposed method to capture the von Kármán vortex street. The displayed simulation corresponds to the FCFV solution in the second mesh using the HLL stabilisation.

A tolerance of 10^{-6} is used for the nonlinear problems and the average number of Newton–Raphson iterations across all time steps is two for the HLL stabilisation and one for the Roe and LF stabilisations.

To assess the accuracy of the simulations, the computed lift (C_L) and drag (C_D) coefficients and the Strouhal number (S_t) are compared against a number of results found in the literature. Fig. 10 shows the lift and drag coefficients as a function of time for the two meshes and the three convective stabilisations, where the shadowed area represents the range of values reported in the literature [34–39].

For the references considered the number of elements varies between 5,000 and 18,000, but it is important to note that in these references linear, quadratic and higher order elements are considered for the simulations. The results reported in the literature are obtained for a variety of numerical schemes including finite volumes, stabilised finite elements and immersed methods with B-Splines.
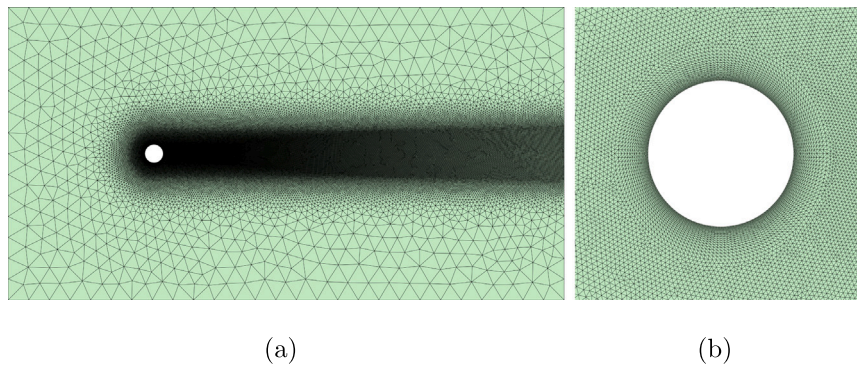
Fig. 8. Unsteady laminar flow past a circular cylinder: (a) Second mesh with 135,273 cells and (b) detail of the inflation layer.
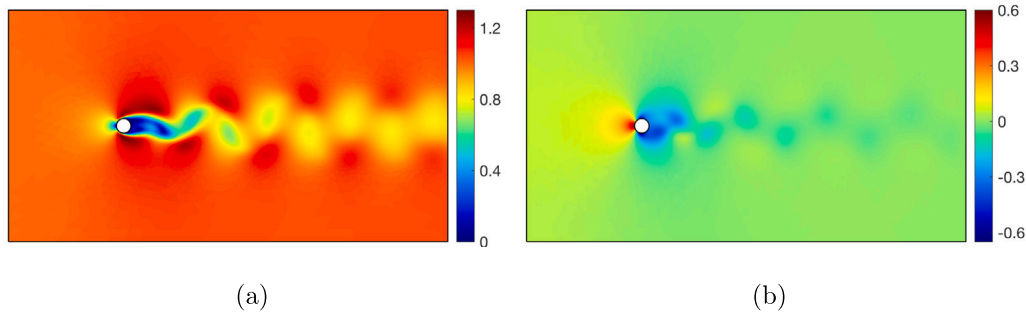

Fig. 9. Unsteady laminar flow past a circular cylinder: Snapshots of the (a) magnitude of the velocity and (b) pressure at $t = 65$.
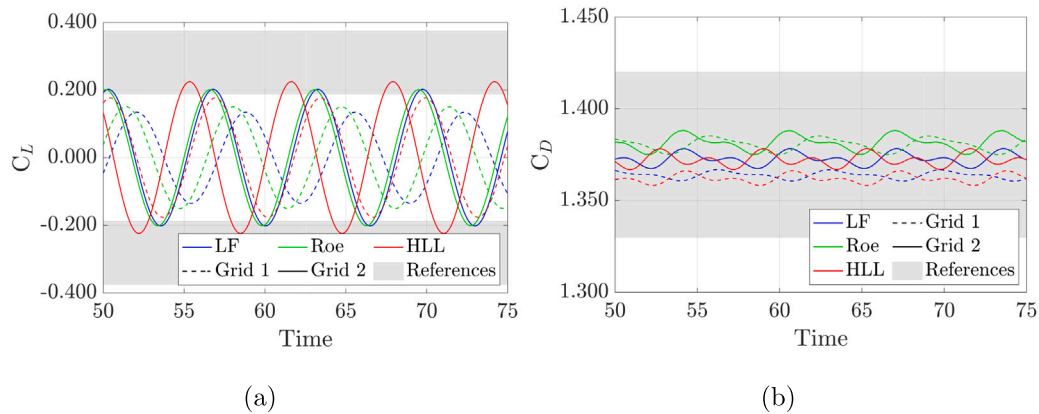

Fig. 10. Unsteady laminar flow past a circular cylinder: (a) Lift and (b) drag coefficients as a function of time for the two meshes and the three convective stabilisations.

A detailed comparison is provided in Table 2, showing the amplitude of the lift coefficient, the mean value of the drag and the Strouhal number using both meshes and the three different stabilisations. Results reported in the literature for the three quantities of interest are also included.

For this example the HLL stabilisation provides the most accurate results. With the first mesh the lift coefficient is almost within the range reported in the literature, whereas the Roe and LF stabilisations require the second grid to provide similar accuracy. Using the second mesh, the results of all convective stabilisations lie within the range reported in the literature. The extra accuracy of the HLL stabilisation is also observed when comparing the Strouhal number to reference values. The results obtained with the HLL stabilisation is almost within the range reported in the references and almost identical to the results obtained with LF and Roe in the second mesh. For the mean value of the drag, the results obtained on both meshes and with the three stabilisations lie within the range reported in the literature.

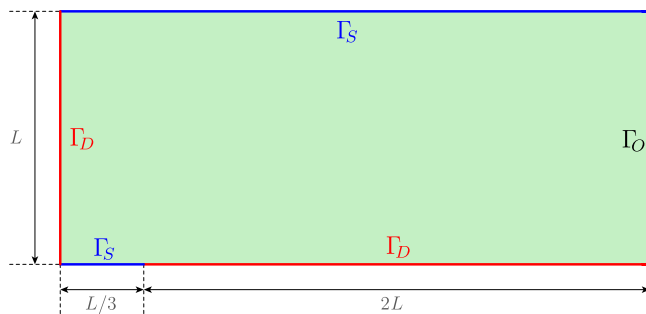### 5.5. Turbulent flow over a flat plate

The next example considers the steady turbulent flow over a flat plate at $Re = 5 \times 10^6$, a classical benchmark to test the accuracy and robustness of steady turbulent flow solvers. The setup of the problem is depicted in Fig. 11, including the relevant dimensions and the boundary conditions. The Dirichlet boundary consists of the inflow boundary, where a constant horizontal velocity of magnitude one is imposed, and the plate, where a no-slip boundary condition is imposed. The inflow SA viscosity is taken as $\tilde{\nu} = 3$, assuming that the flow is fully developed at the inlet [18,19].

Three triangular and quadrilateral meshes, typically employed with NASA solvers CLF3D and FUN3D [40,41], are used to test the accuracy and robustness of the proposed FCFV. The ith quadrilateral mesh has $(68 \times 2^i + 1) \times (48 \times 2^i + 1)$ cells, whereas the corresponding triangular meshes are obtained by splitting each quadrilateral into two triangles. The maximum aspect ratio of the cells near the wall in the first mesh

Table 2

Unsteady laminar flow past a circular cylinder: Amplitude of the lift coefficient, mean value of the drag and Strouhal number using both meshes and the three different stabilisations and reference results.

FCFV results

	C_L amplitude		Mean C_D		S_t	
Stabilisation	Mesh 1	Mesh 2	Mesh 1	Mesh 2	Mesh 1	Mesh 2
LF	0.142	0.202	1.365	1.373	0.148	0.154
Roe	0.159	0.201	1.381	1.382	0.150	0.155
HLL	0.177	0.225	1.362	1.372	0.154	0.159

Literature results

Reference	C_L amplitude	Mean C_D	S_t
[34]	[0.188, 0.375]	[1.35, 1.41]	[0.156, 0.171]
[35]	[0.370, 0.375]	[1.38, 1.40]	[0.166, 0.170]
[36]	[0.250, 0.330]	[1.34, 1.37]	[0.160, 0.170]
[37]	0.253	1.335	0.157
[38]	[0.338, 0.362]	[1.39, 1.42]	[0.165, 0.173]
[39]	[0.293, 0.338]	–	[0.159, 0.169]

Fig. 11. Turbulent flow over a flat plate: Problem setup.

is 2.14×10^5 and it is halved for each successive mesh refinement. The spacing is selected so that the average y^+ at the wall is close to 1 and 0.1 for the coarser and finest meshes, respectively. Fig. 12 displays the coarsest triangular and quadrilateral meshes.

The pseudo-time marching strategy described in Appendix B.4 is applied, using an initial condition corresponding to the free-stream conditions. The solution is advanced until the steady state is reached with a tolerance of 10^{-12}. As no time accuracy is required, only one Newton–Raphson iteration is performed per time step.

Fig. 13 reports the evolution of the residuals of the RANS and SA equations, measured in the maximum norm, and the CFL during the pseudo-time marching for the computation using the coarsest quadrilateral mesh and for the three different convective stabilisations. The results show a very similar performance for the three stabilisations considered. During the first two time steps a low CFL is required due to the non-physical transient effects caused by the initial condition not satisfying the boundary conditions, but after only six time steps the CFL is already near 10^3. The CFL stays between 10^3 and 10^5 until the solution develops and during the last time steps the CFL exceeds 10^{12}, showing the robustness and stability of the proposed FCFV method. The behaviour is very similar using finer meshes or triangular cells.

To assess the accuracy of the FCFV computations, the skin friction along the plate is compared in Fig. 14 to two references, corresponding to the Blasius solution and the numerical solution obtained with the CFL3D solver [40,41]. The FCFV computations are performed on the second triangular and quadrilateral meshes, with $273 \times 193 \times 2$ cells and with 273×193 cells, respectively, and employing the three different convective stabilisations. The numerical reference solution is computed on the finest quadrilateral mesh, featuring 545×385 cells, with the CFL3D solver. The FCFV results show an excellent agreement with the reference behaviour. For the quadrilateral mesh, the results using different convective stabilisations are almost indistinguishable,

Table 3

Turbulent flow over a flat plate: Drag coefficient computed using triangular and quadrilateral meshes and the three different stabilisations, and reference results.

FCFV results

	Triangles			Quadrilaterals		
Stabilisation	Mesh 1	Mesh 2	Mesh 3	Mesh 1	Mesh 2	Mesh 3
LF	0.00290	0.00290	0.00290	0.00281	0.00287	0.00289
Roe	0.00290	0.00291	0.00291	0.00280	0.00287	0.00289
HLL	0.00289	0.00290	0.00290	0.00281	0.00287	0.00289
HLL without SA	0.00099	0.00077	0.00062	0.00129	0.00098	0.00076

Reference results

Reference	Triangles			Quadrilaterals		
CFL3D	–	–	–	0.00287	0.00286	0.00286
FUN3D	–	–	–	0.00284	0.00285	0.00285

whereas on the triangular mesh a slightly better performance of the HLL stabilisatison is observed.

Similar differences between the stabilisations are obtained when comparing the dimensionless value of u^+ as a function of y^+ at $x_1 = 0.97008$, as shown in Fig. 15, and the turbulent viscosity at the same location, as shown in Fig. 16.

Figs. 14 and 15 also show the result obtained with the HLL stabilisation without activating the SA turbulence model. The results clearly show that a turbulent model is required to obtain accurate results, despite the extra dissipation of the first order FCFV method when compared to higher order methods.

To further analyse the accuracy of the results, Table 3 reports the computed drag coefficient using the three available triangular and quadrilateral meshes and the three different convective stabilisations. The results obtained using the CFL3D and FUN3D solvers [40,41] are also included and show that the FCFV results are less than one drag count away from the reference results. The extra accuracy of the FCFV results using triangular meshes is only due to the extra degrees of freedom introduced when splitting the quadrilaterals into two triangles.

To assess the robustness of the proposed FCFV method with respect to mesh quality, a random perturbation to the interior nodes is introduced. The resulting, coarse, distorted triangular and quadrilateral meshes are shown in Fig. 17. The skin friction using the HLL convective stabilisation on the second regular and distorted triangular and quadrilateral meshes are compared in Fig. 18. The results show very little influence of the substantial mesh distortion introduced, demonstrating the insensitivity of the FCFV method to mesh quality and the overall robustness of the proposed solver, even for high Reynolds number turbulent flows.

To further illustrate the robustness of the method with respect to cell distortion, Fig. 19 compares the dimensionless value of u^+ as a function of y^+ at $x_1 = 0.97008$ and the turbulent viscosity at the same location, when using regular and distorted triangular meshes. In both cases, the HLL convective stabilisation is used and the results on distorted meshes are almost identical to the ones obtained with regular meshes. The unnoticeable difference induced by the mesh distortion is corroborated by comparing the drag coefficient computed with distorted meshes. For instance, the drag coefficient obtained using the second distorted triangular mesh is 0.00291, which is less than one drag count away from the result with the second triangular mesh, 0.00290. The same conclusions are obtained for quadrilateral meshes and other levels of mesh refinement.

All above results have been computed using a monolithic strategy to solve the RANS-SA system. A comparison of the performance and robustness of this approach with a staggered algorithm, described in Appendix B.3, can be found in Appendix B.4.
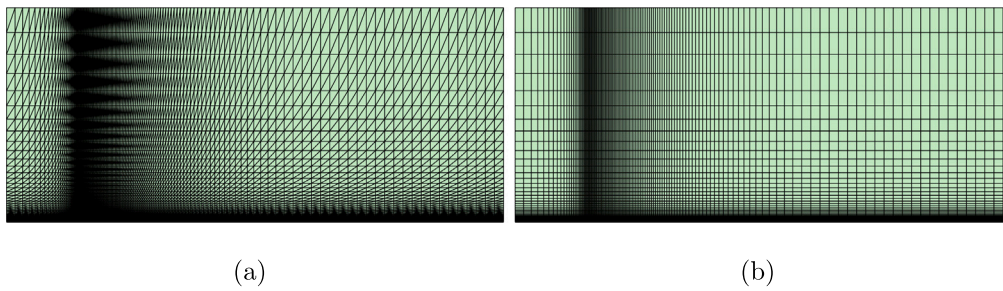
Fig. 12. Turbulent flow over a flat plate: (a) Triangular and (b) quadrilateral coarse meshes.
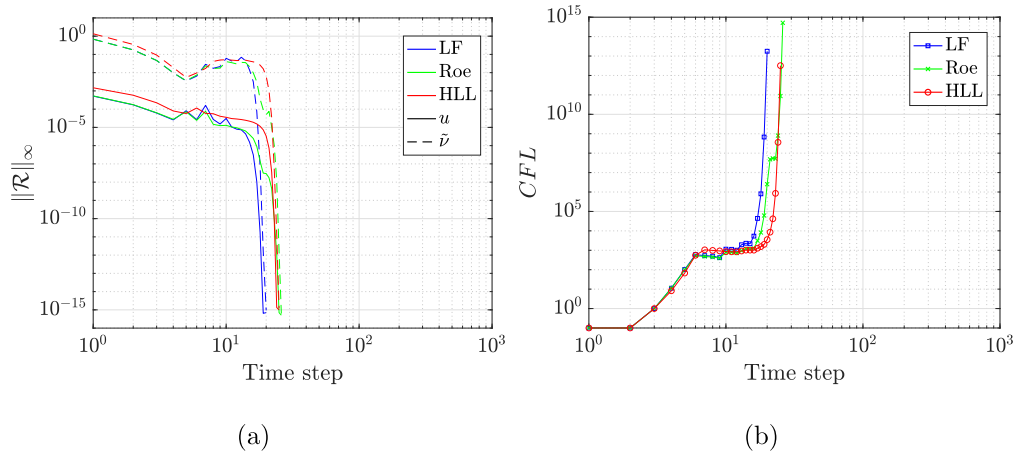

Fig. 13. Turbulent flow over a flat plate: Evolution of the (a) residuals and (b) CFL during the pseudo-time marching for the computation using the second quadrilateral mesh.
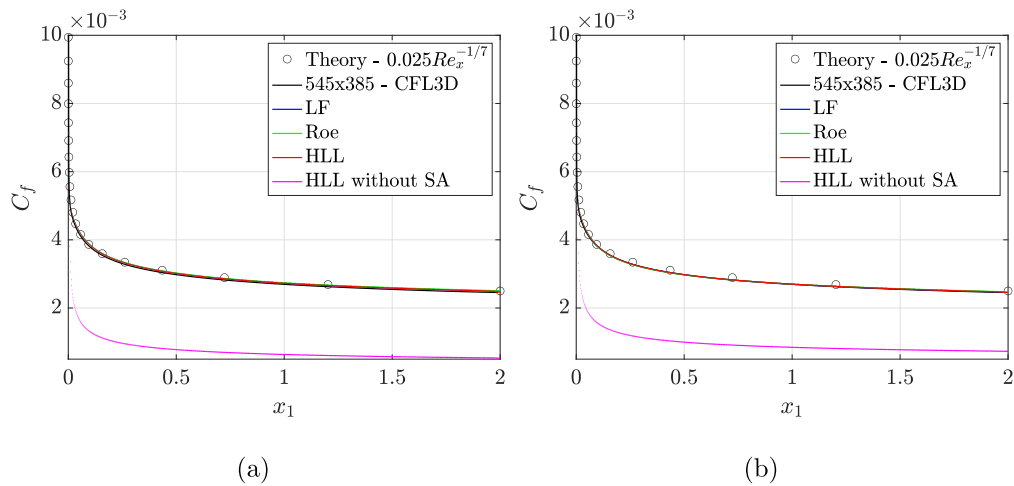

Fig. 14. Turbulent flow over a flat plate: Skin friction along the plate for (a) the second triangular mesh and (b) the second quadrilateral mesh, compared to the Blasius solution and a reference numerical solution. The solution for the HLL convective stabilisation without the SA model is also shown.

5.6. Unsteady turbulent flow past a circular cylinder

The last example considers the unsteady turbulent flow past a circular cylinder of diameter D at $Re = 10^4$. This example is used to assess the applicability of the proposed method to turbulent transient cases. The setup of the problem is depicted in Fig. 20, including the relevant dimensions and the boundary conditions.

Two unstructured triangular meshes are considered. For the first mesh, the inflation layer is defined using a height for the first cell on the cylinder equal to $0.002D$ and a growth ratio of 1.15. A refinement behind the cylinder is introduced to capture the wake by imposing a cell size of $0.015D$ near the cylinder and $0.075D$ near the outflow boundary.

For the second mesh, the height of the first cell in the inflation layer is $0.001D$ and the growth ratio is set to 1.1. The sizes to locally refine the region of the wake are $0.01D$ near the cylinder and $0.05D$ near the outflow. In both cases, the maximum cell stretching near the cylinder is 7.5. The first mesh has 52,094 cells, with a maximum y^+ of 1.6, whereas the second mesh, displayed in Fig. 21, has 98,319 cells, with a maximum y^+ of 0.8.

For the time-integration, the BDF2 scheme is employed with a time-step $\Delta t = 0.01$. The initial condition for the RANS equations is taken as the steady-state solution with $Re = 10$, and following [42], the turbulent variable $\tilde{\nu}$ is taken as 0.05.
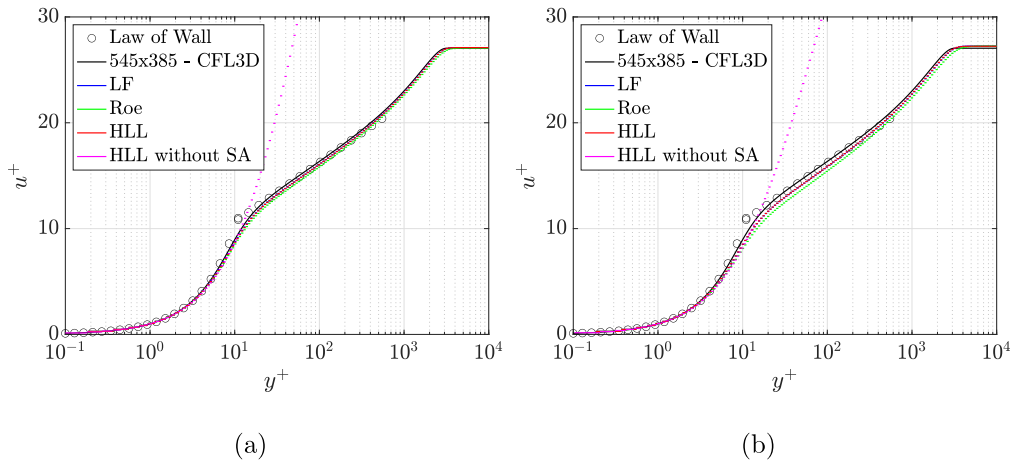
Fig. 15. Turbulent flow over a flat plate: Dimensionless value of u^+ as a function of y^+ at $x_1 = 0.97008$ for (a) the second triangular mesh and (b) the second quadrilateral mesh, compared to the law of wall and a reference numerical solution. The solution for the HLL convective stabilisation without the SA model is also shown.
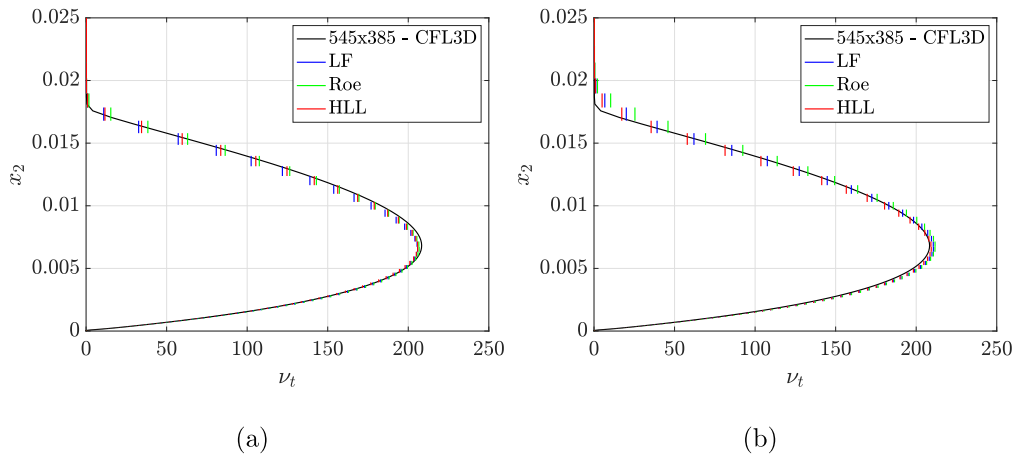

Fig. 16. Turbulent flow over a flat plate: Turbulent viscosity at $x_1 = 0.97008$ for (a) the second triangular mesh and (b) the second quadrilateral mesh, compared to a reference numerical solution.
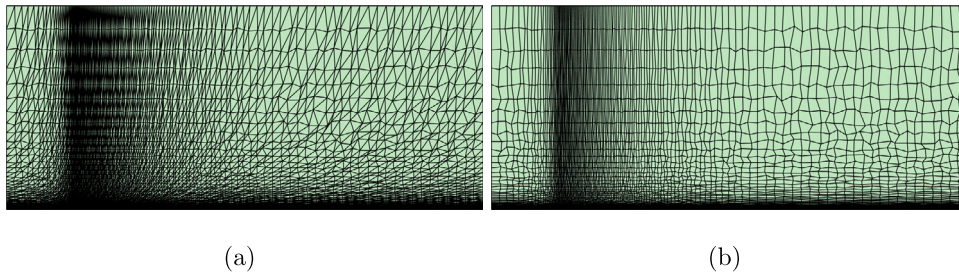

Fig. 17. Turbulent flow over a flat plate: Distorted (a) triangular and (b) quadrilateral coarse meshes.

A snapshot of the magnitude of the velocity, the pressure and the turbulent viscosity at time $t = 61$ is presented in Fig. 22. The simulation corresponds to the FCFV solution in the second mesh using the HLL stabilisation. The results show a more complicated pattern of the vortices, compared to the laminar flow of Section 5.4, and, given the moderate Reynolds number employed, it can be observed that the turbulent effects are confined to the wake.

As in the previous transient case, a tolerance of 10^{-6} is used for the nonlinear problems and the average number of Newton–Raphson iterations across all time steps is three in all cases. This is slightly higher than the number of iterations observed for the laminar flow, illustrating the extra difficulty in solving the coupled RANS-SA system.

The accuracy of the FCFV simulations is assessed by comparing the computed lift (C_L) and drag (C_D) coefficients and the Strouhal number (S_t) against values found in the literature. Fig. 23 shows the lift and drag coefficients as a function of time for the two meshes and the three convective stabilisations, where the shadowed area represents the range of values found in the literature [42–44].

For the references considered, the number of elements varies between 2,000 and 47,000, but it is important to note that in these references higher order approximations are considered for the simulations. The results reported in the literature are obtained for a variety of numerical schemes including finite volumes and stabilised finite elements.
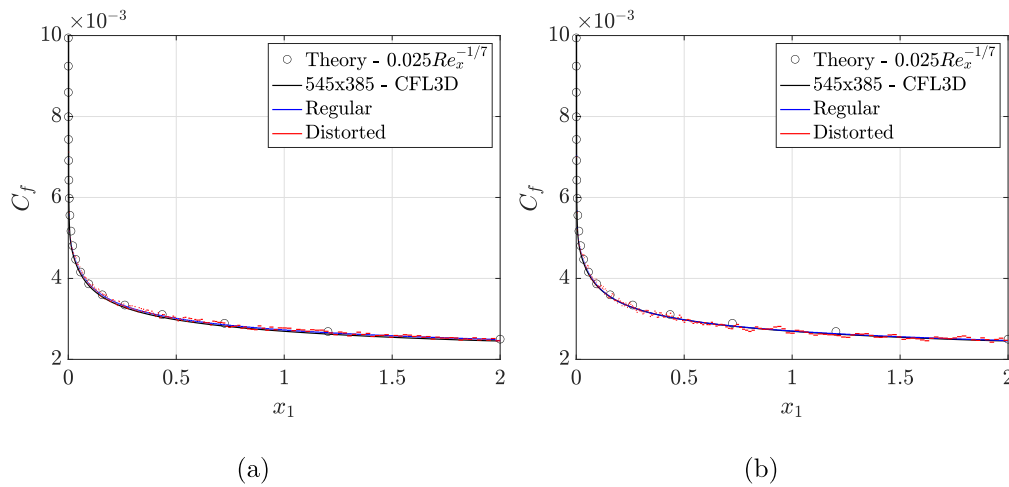
Fig. 18. Turbulent flow over a flat plate: Skin friction on the second (a) triangular and (b) quadrilateral distorted meshes compared to the computation with regular meshes, the Blasius solution and a reference numerical solution.
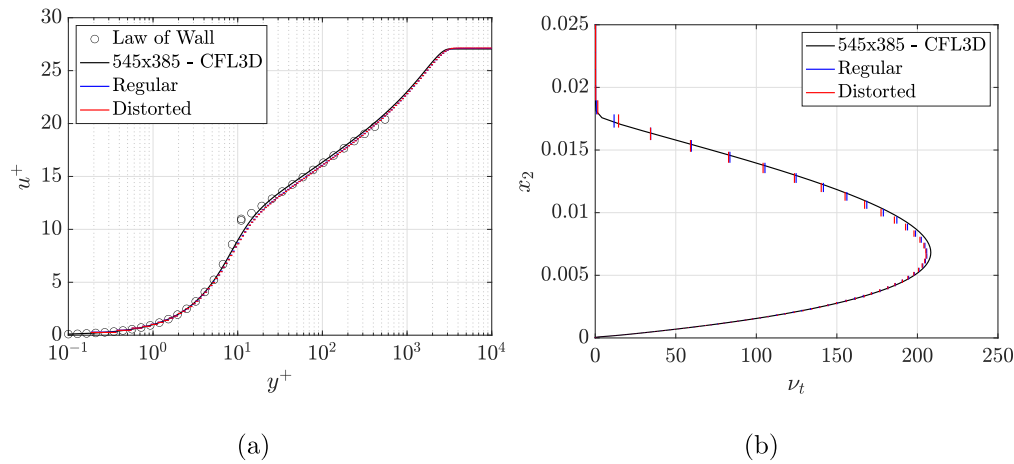

Fig. 19. Turbulent flow over a flat plate: (a) Dimensionless value of u^+ as a function of y^+ and (b) turbulent viscosity, at $x_1 = 0.97008$, on the second triangular distorted mesh compared to the computation with regular meshes and a reference numerical solution.
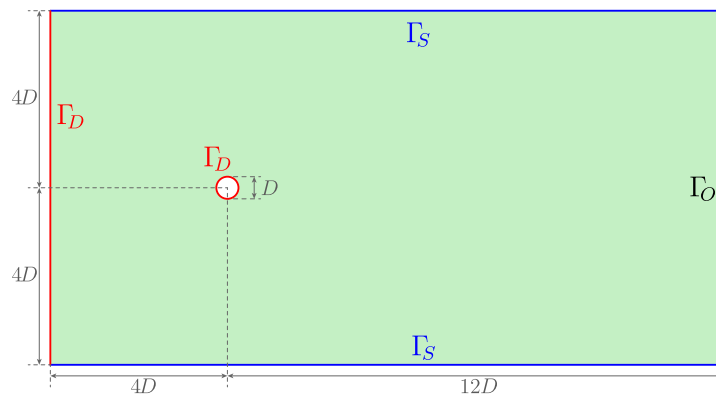

Fig. 20. Unsteady turbulent flow past a circular cylinder: Problem setup.

A more detailed comparison is provided in Table 4, reporting the amplitude of the lift coefficient, the mean value of the drag and the Strouhal number using both meshes and the three different stabilisations. Results reported in the literature for the three quantities of interest are also included.

To conclude, the simulation using the first mesh and the HLL flux is repeated using the laminar FCFV solver, without the SA turbulence

model. For this case, the C_L amplitude obtained is 1.74, the mean C_D is 1.71 and the Strouhal number is 0.230. These results substantially differ from the results obtained with the SA model, shown in Table 4, and are outside of the range of results reported in the Refs. [42–44]. This shows, again, the need to incorporate the SA turbulence model to accurately simulate the problems considered here, even if the
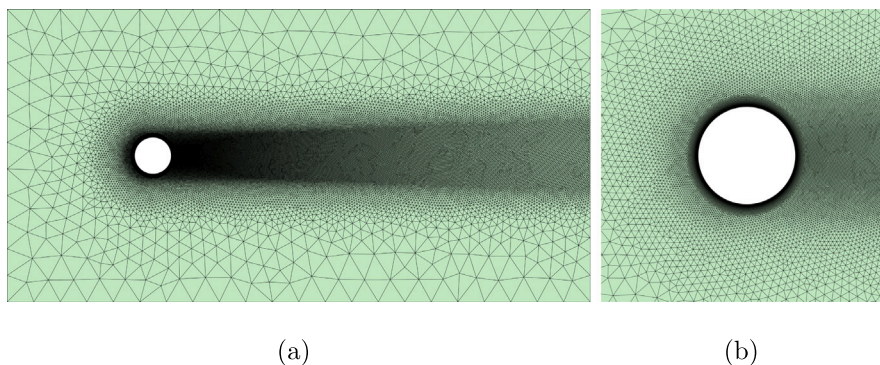
Fig. 21. Unsteady turbulent flow past a circular cylinder: (a) Second mesh with 98,193 cells and (b) detail of the inflation layer.
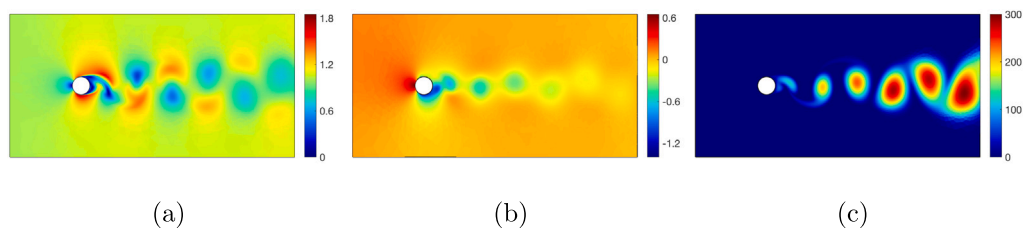

Fig. 22. Unsteady turbulent flow past a circular cylinder: Snapshots of the (a) magnitude of the velocity, (b) pressure and (c) turbulent viscosity at $t = 61$.
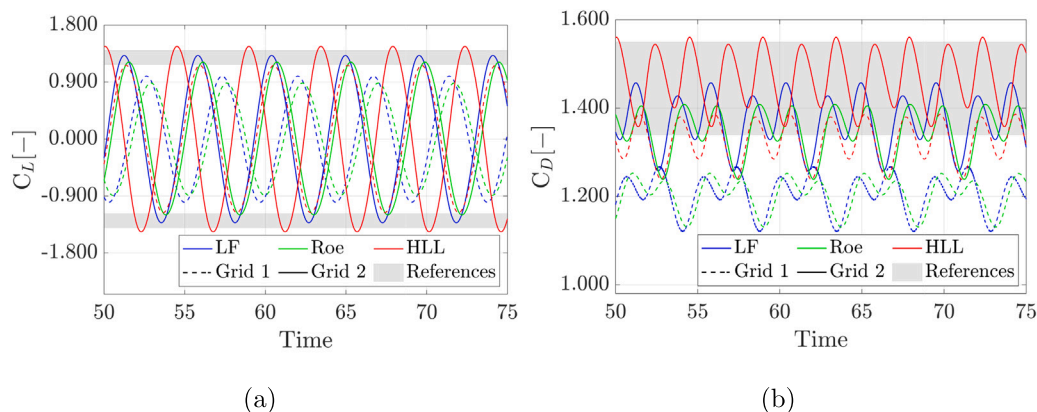

Fig. 23. Unsteady turbulent flow past a circular cylinder: (a) Lift and (b) drag coefficients as a function of time for the two meshes and the three convective stabilisations.

Table 4
Unsteady turbulent flow past a circular cylinder: Amplitude of the lift coefficient, mean value of the drag and Strouhal number using both meshes and the three different stabilisations and reference results.

FCFV results						
	C_L amplitude		Mean C_D		S_t	
Stabilisation	Mesh 1	Mesh 2	Mesh 1	Mesh 2	Mesh 1	Mesh 2
LF	1.00	1.32	1.21	1.37	0.213	0.219
Roe	0.90	1.20	1.20	1.34	0.213	0.218
HLL	1.18	1.46	1.32	1.46	0.218	0.224
Literature results						
Reference	C_L amplitude		Mean C_D		S_t	
[42]	1.40		1.35		0.167	
[43]	1.18		1.34		[0.16, 0.19]	
[44]	–		1.55		[0.217,0.244]	

FCFV method proposed is first-order and therefore introduces higher dissipation when compared to other, second-order, FV schemes.

6. Concluding remarks

The development and application of the FCFV method for laminar and turbulent viscous incompressible flows has been presented, including the introduction of three new stabilisations, inspired by classical Riemann solvers, for the convective part of the RANS and SA equations.

The Couette and the lid-driven cavity flow problems were used to test the convergence of the error under mesh refinement. In all cases, the results confirmed the first-order convergence of the error for all the variables, namely the cell velocity, the face velocity, the velocity-gradient tensor and the pressure. The performance of three convective stabilisations proposed in this work was also tested using the lid-driven cavity flow, showing a slightly superior performance of the HLL stabilisation, when compared to the LF and Roe stabilisation.

For a set of laminar and turbulent benchmarks, severe mesh distortion and stretching was introduced to test the robustness of the proposed FCFV method. In all the examples, it was observed that mesh distortion induces very little difference in the results, even when cell distortion is introduced in the boundary layer region, for high Reynolds number flows.

Monolithic and staggered approaches to handle the RANS-SA system are also discussed and numerically compared. Future work will focus on the parallel implementation of the FCFV method in three dimensions and the application to large scale problems. To this end, particular attention will be paid to the solution of the global system of equations and the potential of the staggered approach to reduce the memory footprint.

CRediT authorship contribution statement

Luan M. Vieira: Visualization, Validation, Software, Investigation, Formal analysis, Data curation. **Matteo Giacomini:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization. **Ruben Sevilla:** Writing – review & editing, Writing – original draft, Supervision, Methodology, Investigation, Formal analysis, Conceptualization. **Antonio Huerta:** Writing – review & editing, Supervision, Resources, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgements

Appendix A. Convective stabilisation

This appendix presents three new convective stabilisation tensors for the incompressible RANS equations and the SA turbulence model. The convective stabilisations τ^a and $\tilde{\tau}^a$ are used to define the full stabilisation tensors τ and $\tilde{\tau}$ in the local and global problems given by Eqs. (18) and (21), respectively. They are used and compared in several examples presented in Section 5.

The rationale is inspired by classical Lax–Friedrichs, Roe and Harten-Lax-van Leer Riemann solvers for compressible flows [45] and it is derived following the unified approach presented in [16,23].

The convective part of the RANS momentum equation can be written as

$$\frac{\partial \boldsymbol{u}}{\partial t} + \frac{\partial \mathbf{F}_k(\boldsymbol{u})}{\partial x_k} = \mathbf{0}, \tag{A.1}$$

where $\mathbf{F}_k(\boldsymbol{u}) = u_k \boldsymbol{u}$, or in quasi-linear form as

$$\frac{\partial \boldsymbol{u}}{\partial t} + \mathbf{A}_k \frac{\partial \boldsymbol{u}}{\partial x_k} = \mathbf{0}, \tag{A.2}$$

where $\mathbf{A}_k = \partial \mathbf{F}_k / \partial \boldsymbol{u}$.

To derive the stabilisation tensors τ^a, the system of equations is restricted to an arbitrary direction $\boldsymbol{n}$, which in the context of the FCFV will correspond to the normal direction to an edge/face of a cell. The resulting system is

$$\frac{\partial \boldsymbol{u}}{\partial t} + \mathbf{A}_n \frac{\partial \boldsymbol{u}}{\partial \boldsymbol{n}} = \mathbf{0}, \tag{A.3}$$

with $\mathbf{A}_n = (\boldsymbol{u} \cdot \boldsymbol{n}) \mathbf{I}_{\mathrm{n_{sd}}} + \boldsymbol{u} \otimes \boldsymbol{n}$.

The spectral decomposition of $\mathbf{A}_n$ leads to $\mathbf{A}_n = \mathbf{R} \boldsymbol{\Lambda} \mathbf{R}^{-1}$, where $\mathbf{R}$ is the matrix whose columns are the right eigenvectors of $\mathbf{A}_n$ and $\boldsymbol{\Lambda} = \mathrm{diag}\left(\lambda_1, \dots, \lambda_{\mathrm{n_{sd}}}\right)$ with $\lambda_1 = \dots = \lambda_{\mathrm{n_{sd}}-1} = v_n$, $\lambda_{\mathrm{n_{sd}}} = 2v_n$, and $v_n = \boldsymbol{u} \cdot \boldsymbol{n}$. To ensure a minimum value of the stabilisation, the regularised eigenvalues are considered, namely $\hat{\lambda}_1 = \dots = \hat{\lambda}_{\mathrm{n_{sd}}-1} = \max\left(|\boldsymbol{u} \cdot \boldsymbol{n}|, \varepsilon\right)$, $\hat{\lambda}_{\mathrm{n_{sd}}} = \max\left(2|\boldsymbol{u} \cdot \boldsymbol{n}|, \varepsilon\right)$, where $\varepsilon > 0$ is a real, user-defined numerical parameter.

Using the maximum eigenvalue of $\hat{\boldsymbol{\Lambda}} = \mathrm{diag}\left(\hat{\lambda}_1, \dots, \hat{\lambda}_{\mathrm{n_{sd}}}\right)$, the stabilisation inspired by the Lax–Friedrichs (LF) Riemann solver is defined as

$$\boldsymbol{\tau}^a_{\mathrm{LF}} = \max\left\{2|\hat{\boldsymbol{u}} \cdot \boldsymbol{n}|, \varepsilon\right\} \mathbf{I}_{\mathrm{n_{sd}}}. \tag{A.4}$$

The stabilisation inspired by the Roe Riemann solver is defined as

$$\boldsymbol{\tau}^a_{\mathrm{Roe}} = \mathbf{R} \, \hat{\boldsymbol{\Lambda}} \, \mathbf{R}^{-1}. \tag{A.5}$$

From an implementation point of view, it is convenient to rewrite the Roe stabilisation as

$$\boldsymbol{\tau}^a_{\mathrm{Roe}} = \begin{cases} \varepsilon \, \mathbf{I}_{\mathrm{n_{sd}}} & \text{for } |\hat{\boldsymbol{u}} \cdot \boldsymbol{n}| \leq \varepsilon/2, \\ \varepsilon \, \mathbf{I}_{\mathrm{n_{sd}}} + \mathrm{sign}(\hat{\boldsymbol{u}} \cdot \boldsymbol{n}) \, (2 - (\varepsilon/|\hat{\boldsymbol{u}} \cdot \boldsymbol{n}|)) \, (\hat{\boldsymbol{u}} \otimes \boldsymbol{n}) & \text{for } |\hat{\boldsymbol{u}} \cdot \boldsymbol{n}| \in (\varepsilon/2, \varepsilon), \\ \mathrm{sign}(\hat{\boldsymbol{u}} \cdot \boldsymbol{n}) \left[(\hat{\boldsymbol{u}} \cdot \boldsymbol{n}) \mathbf{I}_{\mathrm{n_{sd}}} + (\hat{\boldsymbol{u}} \otimes \boldsymbol{n})\right] & \text{for } |\hat{\boldsymbol{u}} \cdot \boldsymbol{n}| \geq \varepsilon, \end{cases} \tag{A.6}$$

The last stabilisation considered is inspired by the Harten-Lax-van Leer (HLL) Riemann solver. This stabilisation, defined as

$$\boldsymbol{\tau}^a_{\mathrm{HLL}} = \max\{2(\hat{\boldsymbol{u}} \cdot \boldsymbol{n}), \varepsilon\} \mathbf{I}_{\mathrm{n_{sd}}}, \tag{A.7}$$

employs an estimate of the largest wave speed of the Riemann problem.

For the SA turbulence model, the derivation of the convective stabilisation $\tilde{\tau}^a$ is simpler due to the scalar nature of the equation. The stabilisation inspired from both LF and Roe Riemann solvers is simply

$$\tilde{\tau}^a_{\mathrm{LF}} = \tilde{\tau}^a_{\mathrm{Roe}} = \max\left\{|\hat{\boldsymbol{u}} \cdot \boldsymbol{n}|, \varepsilon\right\}. \tag{A.8}$$

whereas the stabilisation inspired by the HLL Riemann solver is

$$\tilde{\tau}^a_{\mathrm{HLL}} = \max\left\{\hat{\boldsymbol{u}} \cdot \boldsymbol{n}, \varepsilon\right\}. \tag{A.9}$$

It can be observed that both the Lax–Friedrichs and the Roe stabilisations provide a unique value of the stabilisation as seen from the two cells sharing an edge/face. In contrast, the HLL stabilisation provides different values for the two cells sharing an edge/face.

The Roe Riemann solver of the RANS equations, given in Eq. (A.6), provides a continuous expression when $|\hat{\boldsymbol{u}} \cdot \boldsymbol{n}|$ tends to ε and $\varepsilon/2$, which numerical experiments have confirmed to be a key factor to guarantee convergence of the numerical scheme.

In all the examples, the numerical parameter ε is selected as 5×10^{-2} for the LF and HLL stabilisations of RANS equations, whereas a slightly higher value of 10^{-1} is used for the Roe stabilisation. For the SA equation, the value of ε is taken as 10^{-2} for all stabilisations.

Appendix B. Computational aspects

This appendix briefly discusses some computational aspects to be considered when implementing the proposed FCFV for solving the incompressible RANS equations with the SA turbulence model.

B.1. Solution of global and local problems

The cost of each Newton–Raphson iteration is clearly dominated by the cost of solving the global problem. The size of the sparse non-symmetric linear system of Eqs. (27) is $n_{\mathrm{eq}} = n_{\mathrm{fa}}(n_{\mathrm{sd}} + 1) + n_{\mathrm{el}}$, where n_{fa} is the total number of edges/faces in the mesh. In two dimensions and for a mesh with N nodes, $n_{\mathrm{eq}} \simeq 11N$ for a mesh of triangular cells and $n_{\mathrm{eq}} \simeq 7N$ for a quadrilateral mesh. In this work, the solution
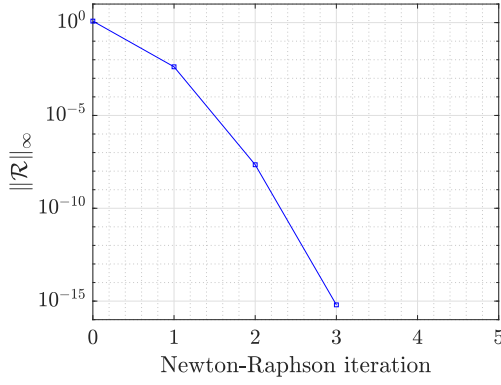
Fig. B.24. Residual of Eq. (B.1) as a function of the number of Newton–Raphson iterations for the Couette example.

of the global system is performed using the parallel direct method implemented in the Harwell subroutine library code `ma41` [46,47].

The size of each local problem is only $(n_{sd} + 1)^2$. The local problems can be trivially solved in parallel as there is no communication required between cells. Furthermore, the solution of a local problem does not actually require solving a linear system of equations of size $(n_{sd} + 1)^2$. After solving the global system (27), the increments of $\mathbf{L}_e$ and $\tilde{\mathbf{q}}_e$ are explicitly obtained from the increments of $\hat{\mathbf{u}}$ and $\hat{v}$, using the first and third equations in (18). With the updated values of $\hat{\mathbf{u}}$, $\hat{v}$ and $\tilde{\mathbf{q}}_e$, the last equation of (18) is used to obtain the increment of the turbulent variable $\tilde{v}_e$. Finally, the velocity can be updated using the second equation of (18). In summary, each local problem involves the solution of four independent equations, where three provide explicit expressions for the unknowns, whereas one requires the solution of a nonlinear problem with a scalar unknown.

B.2. Convergence of the Newton–Raphson method

To check the convergence of the Newton–Raphson iterations, the following residual is employed in all the examples

$$\|\mathcal{R}^n\|_\infty := \max \left\{ \frac{\|\mathbf{R}_{\hat{u}}^n\|_\infty}{\|\mathbf{f}_u^n\|_\infty}, \frac{\|\mathbf{R}_{\hat{v}}^n\|_\infty}{\|\mathbf{f}_{\tilde{v}}^n\|_\infty} \right\} \tag{B.1}$$

where the normalising factors are defined as

$$\mathbf{f}_{e,u}^n := |\Omega_e| \mathbf{s}_e^n + \sum_{j \in D_e} |\Gamma_{e,j}| \left(\boldsymbol{\tau}_j^{n-1} - (\boldsymbol{u}_{D,j}^n \cdot \boldsymbol{n}_j) \mathbf{I}_{n_{sd}} \right) \boldsymbol{u}_{D,j}^n \tag{B.2}$$

and

$$\mathbf{f}_{e,\tilde{v}}^n := \sum_{j \in D_e} |\Gamma_{e,j}| (\tilde{\tau}_j^{n-1} - \boldsymbol{u}_{D,j}^n \cdot \boldsymbol{n}_j) \tilde{v}_{D,j}^n \tag{B.3}$$

for the RANS and SA equations respectively.

To illustrate the quadratic convergence attained, Fig. B.24 shows the residual of Eq. (B.1) as a function of the number of Newton–Raphson iterations for the Couette example of Section 5.1.

B.3. Monolithic versus staggered solution

The solution strategy used in Sections 5.5 and 5.6 employs a monolithic coupling of the RANS and SA equations. An alternative approach would be to solve the RANS and SA equations in a staggered manner. Such an approach requires solving a global system to obtain $\hat{\mathbf{u}}$ and ρ followed by a global system to obtain $\hat{v}$. This is then repeated until convergence.

The staggered approach requires solving two global systems, associated with the RANS and SA equations, of size $n_{eq}^{RANS} = n_{fa} n_{sd} + n_{el}$ and $n_{eq}^{SA} = n_{fa}$, respectively. A priori, it is not feasible to compare the cost of the two approaches because it is not possible to know the

number of iterations required by the staggered scheme to converge, but some comments can be made about the memory requirements of both strategies.

Assuming that the number of exterior edges/faces in the mesh is negligible, when compared to the number of interior edges/faces, the number of non-zero entries in the global system of the monolithic approach is

$$n_{nz} \simeq \sum_{e=1}^{n_{el}} n_{fa}^e \left[n_{fa}^e (n_{sd}^2 + 1) + 2n_{sd}(n_{fa}^e + 1) \right] - n_{fa}(n_{sd} + 1)^2. \tag{B.4}$$

In contrast, the number of non-zero entries of the global problems associated to the RANS and SA equations in the staggered approach is

$$n_{nz}^{RANS} \simeq \sum_{e=1}^{n_{el}} n_{fa}^e n_{sd}(n_{fa}^e n_{sd} + 2) - n_{fa} n_{sd}^2, \qquad n_{nz}^{SA} \simeq \sum_{e=1}^{n_{el}} (n_{fa}^e)^2 - n_{fa}. \tag{B.5}$$

In two dimensions and for a mesh with N nodes, the global matrix of the monolithic approach has $n_{nz} \simeq 159N$ for a mesh of triangular cells and $n_{nz} \simeq 142N$ for a quadrilateral mesh. For the staggered approach, the global matrices have $n_{nz}^{RANS} \simeq 84N$ and $n_{nz}^{SA} \simeq 15N$ for triangular meshes and $n_{nz}^{RANS} \simeq 72N$ and $n_{nz}^{SA} \simeq 14N$ for quadrilateral meshes. This means that the staggered approach requires approximately 60% of the memory required by the monolithic approach to store the global matrices.

B.4. Pseudo-time marching for steady-state computations

When a steady-state simulation is of interest, the solution of a pseudo-transient problem, where t denotes the artificial or pseudo-time, is often preferred due to the difficulty to find an initial guess that guarantees convergence of the Newton–Raphson algorithm. In this scenario, time accuracy is not a requirement and, therefore, it is common to utilise a first-order time marching scheme to reach the steady state. Furthermore, strategies to vary the time step as the solution evolves are attractive because at the beginning of a simulation smaller time steps are required to guarantee convergence of the nonlinear problem, whereas as the solution approaches steady state it is possible to use extremely large time steps.

When time marching to a steady state, this work employs the low order BDF1 scheme. Several strategies have been proposed to automatically adjust the time step during steady state simulations, e.g. a simple exponential law [48,49]. More elaborated approaches such as the so-called switched evolution relaxation (SER) [48–50] approach use the ratio between the nonlinear residuals at two consecutive time steps to adjust the time step. This work adapts the SER approach to the proposed FCFV scheme.

At the beginning of a simulation, the CFL is initialised to a low value, namely $CFL^0 = 0.1$. This is particularly important to avoid the potential negative impact of performing an impulsive start with an initial condition not satisfying the boundary conditions, e.g. using freestream values for the initial condition. Following [48,51–54], after the first time step the CFL is updated according to the expression

$$CFL^{n+1} = \min \left\{ CFL^n / f^\gamma, CFL_{max} \right\}, \tag{B.6}$$

where

$$f = \max \left\{ \frac{\|\mathbf{R}_{u,s}^n\|_\infty}{\|\mathbf{R}_{u,s}^{n-1}\|_\infty}, \frac{\|\mathbf{R}_{\tilde{v},s}^n\|_\infty}{\|\mathbf{R}_{\tilde{v},s}^{n-1}\|_\infty} \right\} \tag{B.7}$$

and the exponent γ is taken as $\gamma_{max} > 1$ if both residuals decrease with respect to the previous time step, i.e., if $f \leq 1$, or as $\gamma_{min} < 1$ if at least one of the residuals increases, i.e., if $f > 1$.

In Eq. (B.7) the residual vector $\mathbf{R}_{u,s}$ and $\mathbf{R}_{\tilde{v},s}$ correspond to the result of assembling the elemental contributions of the second and fourth residual in Eq. (18), respectively, but without including the term corresponding to the discrete time derivative. It is worth noting that
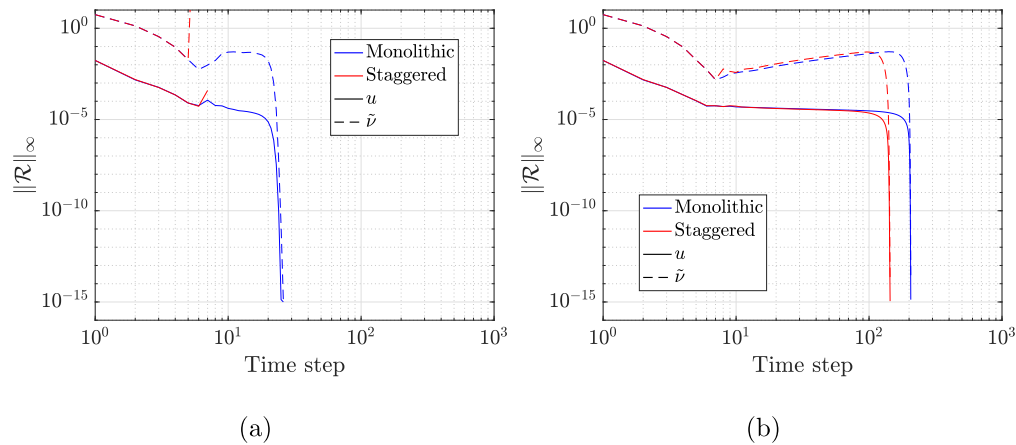
Fig. B.25. Turbulent flow over a flat plate: Evolution of the residuals with (a) $\gamma_{\max} = 2$ and (b) $\gamma_{\max} = 1.2$.

only two residuals of the local problem (18) are employed to define f in Eq. (B.7), because the other two residuals are linear. Furthermore, the residuals of the global problem (21) are also linear with respect to $\boldsymbol{u}$, $\hat{\boldsymbol{u}}$, $\boldsymbol{L}$, ρ and $\hat{\nu}$.

Following [51,52,54], in all the numerical examples the maximum allowed CFL is taken as $\mathrm{CFL}_{\max} = 10^{20}$. When employing the monolithic FCFV approach, the parameters to define the exponent γ are taken as $\gamma_{\max} = 2$ and $\gamma_{\min} = 0.1$. When using the staggered FCFV approach, numerical experiments show that it is necessary to employ a lower value for $\gamma_{\max}$. To illustrate this phenomenon, let us consider the turbulent flow over a flat plate problem presented in Section 5.5. Fig. B.25 shows the evolution of the residuals of the RANS and SA equations during the pseudo-time marching. The results correspond to the second quadrilateral mesh and using the HLL stabilisation. When using the value of $\gamma_{\max} = 2$ the monolithic approach converges in less than 30 time steps, whereas the staggered approach diverges after six time steps, as displayed in Fig. B.25(a). This illustrates a lack of stability induced by the staggered approach. A simple remedy consists of using a more conservative approach to increase the CFL number, for instance lowering the value of $\gamma_{\max}$. The results with $\gamma_{\max} = 1.2$ in Fig. B.25(b) show that the staggered approach converges. As expected, the more conservative increase of the CFL results in convergence, but with a substantially larger number of time steps.

References

[1] LeVeque RJ. Finite volume methods for hyperbolic problems. In: Cambridge texts in applied mathematics, Cambridge: Cambridge University Press; 2002, p. xx+558.

[2] Barth T, Herbin R, Ohlberger M. Finite volume methods: Foundation and analysis. In: Stein E, de Borst R, Hughes T, editors. Encyclopedia of computational mechanics second edition. John Wiley & Sons, Ltd; 2017, p. 1–60.

[3] Diskin B, Thomas JL, Nielsen EJ, Nishikawa H, White JA. Comparison of node-centered and cell-centered unstructured finite-volume discretizations: Viscous fluxes. AIAA J 2010;48(7):1326.

[4] Diskin B, Thomas JL. Comparison of node-centered and cell-centered unstructured finite-volume discretizations: Inviscid fluxes. AIAA J 2011;49(4):836–54.

[5] Sevilla R, Giacomini M, Huerta A. A face-centred finite volume method for second-order elliptic problems. Internat J Numer Methods Engrg 2018;115(8):986–1014.

[6] Cockburn B, Gopalakrishnan J. A characterization of hybridized mixed methods for second order elliptic problems. SIAM J Numer Anal 2004;42(1):283–301.

[7] Cockburn B, Gopalakrishnan J. Incompressible finite elements via hybridization. I. The Stokes system in two space dimensions. SIAM J Numer Anal 2005;43(4):1627–50.

[8] Cockburn B, Gopalakrishnan J. New hybridization techniques. GAMM-Mitt 2005;28(2):154–82.

[9] Cockburn B, Gopalakrishnan J, Lazarov R. Unified hybridization of discontinuous Galerkin, mixed, and continuous Galerkin methods for second order elliptic problems. SIAM J Numer Anal 2009;47(2):1319–65.

[10] Donea J, Huerta A. Finite element methods for flow problems. John Wiley & Sons, Ltd.; 2003.

[11] Sevilla R, Giacomini M, Huerta A. A locking-free face-centred finite volume (FCFV) method for linear elastostatics. Comput Struct 2019;212:43–57.

[12] Vieira LM, Giacomini M, Sevilla R, Huerta A. A second-order face-centred finite volume method for elliptic problems. Comput Methods Appl Mech Engrg 2020;358:112655.

[13] Giacomini M, Sevilla R. A second-order face-centred finite volume method on general meshes with automatic mesh adaptation. Internat J Numer Methods Engrg 2020;121(23):5227–55.

[14] Sevilla R, Duretz T. A face-centred finite volume method for high-contrast Stokes interface problems. Internat J Numer Methods Engrg 2023;124(17):3709–32.

[15] Sevilla R, Duretz T. Face-centred finite volume methods for Stokes flows with variable viscosity. Internat J Numer Methods Engrg 2024.

[16] Vila-Pérez J, Giacomini M, Sevilla R, Huerta A. A non-oscillatory face-centred finite volume method for compressible flows. Comput & Fluids 2022;105272.

[17] Vila-Pérez J, Giacomini M, Huerta A. Benchmarking the face-centred finite volume method for compressible laminar flows. Internat J Numer Methods Heat Fluid Flow 2023;33(6):2198–231.

[18] Spalart PR, Allmaras SR. A one-equation turbulence model for aerodynamic flows. In: 30th Aerospace sciences meeting and exhibit. American Institute of Aeronautics and Astronautics; 1992.

[19] Allmaras SR, Johnson FT, Spalart PR. Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model. In: Seventh international conference on computational fluid dynamics. 2012, p. 1902.

[20] Giacomini M, Karkoulias A, Sevilla R, Huerta A. A superconvergent HDG method for Stokes flow with strongly enforced symmetry of the stress tensor. J Sci Comput 2018;77(3):1679–702.

[21] Giacomini M, Sevilla R, Huerta A. Tutorial on hybridizable discontinuous Galerkin (HDG) formulation for incompressible flow problems. In: De Lorenzis L, Düster A, editors. In: Modeling in engineering using innovative numerical methods for solids and fluids, vol. 599, Cham: Springer International Publishing; 2020, p. 163–201.

[22] La Spina A, Kronbichler M, Giacomini M, Wall WA, Huerta A. A weakly compressible hybridizable discontinuous Galerkin formulation for fluid-structure interaction problems. Comput Methods Appl Mech Engrg 2020;372:113392.

[23] Vila-Pérez J, Giacomini M, Sevilla R, Huerta A. Hybridisable discontinuous Galerkin formulation of compressible flows. Arch Comput Methods Eng 2021;28(2):753–84.

[24] Cockburn B, Gopalakrishnan J. The derivation of hybridizable discontinuous Galerkin methods for Stokes flow. SIAM J Numer Anal 2009;47(2):1092–125.

[25] Nguyen NC, Peraire J, Cockburn B. A hybridizable discontinuous Galerkin method for Stokes flow. Comput Methods Appl Mech Engrg 2010;199(9–12):582–97.

[26] Cockburn B, Nguyen NC, Peraire J. A comparison of HDG methods for Stokes flow. J Sci Comput 2010;45(1–3):215–37.

[27] Nguyen NC, Peraire J, Cockburn B. An implicit high-order hybridizable discontinuous Galerkin method for the incompressible Navier-Stokes equations. J Comput Phys 2011;230(4):1147–70.

[28] Sevilla R, Huerta A. Tutorial on hybridizable discontinuous Galerkin (HDG) for second-order elliptic problems. In: Schröder J, Wriggers P, editors. Advanced finite element technologies. CISM international centre for mechanical sciences, vol. 566, Springer International Publishing; 2016, p. 105–29.

[29] Cockburn B, Dong B, Guzmán J. A superconvergent LDG-hybridizable Galerkin method for second-order elliptic problems. Math Comp 2008;77(264):1887–916.

[30] Sevilla R, Giacomini M, Karkoulias A, Huerta A. A superconvergent hybridisable discontinuous Galerkin method for linear elasticity. Internat J Numer Methods Engrg 2018;116(2):91–116.

[31] Ascher UM, Petzold LR. Computer methods for ordinary differential equations and differential-algebraic equations. vol. 61, SIAM; 1998.

[32] Giacomini M, Sevilla R, Huerta A. HDGlab: An open-source implementation of the hybridisable discontinuous Galerkin method in MATLAB. Arch Comput Methods Eng 2021;28(3):1941–86.

[33] Cheung SW, Chung E, Kim HH, Qian Y. Staggered discontinuous Galerkin methods for the incompressible Navier–Stokes equations. J Comput Phys 2015;302:251–66.

[34] Tezduyar TE, Mittal S, Shih R. Time-accurate incompressible flow computations with quadrilateral velocity-pressure elements. Comput Methods Appl Mech Engrg 1991;87(1–2):363–84.

[35] Tezduyar TE, Mittal S, Ray SE, Shih R. Incompressible flow computations with stabilized bilinear and linear equal-order-interpolation velocity-pressure elements. Comput Methods Appl Mech Engrg 1992;95(2):221–42.

[36] Kjellgren C. A semi-implicit fractional step finite element method for viscous incompressible flow. Comput Mech 1997;20:541–50.

[37] Rajani BN, Kandasamy A, Majumdar S. Numerical simulation of laminar flow past a circular cylinder. Appl Math Model 2009;33(3):1228–47.

[38] Kadapa C, Dettmer WG, Perić D. A fictious domain/distributed Lagrange multiplier based fluid-structure interaction scheme with hierachical B-spline grids. Comput Methods Appl Mech Engrg 2016;301:1–27.

[39] Kadapa C, Dettmer WG, Perić D. Accurate iteration-free mixed-stabilised formulation for laminar incompressible Navier-Stokes: Applications to fluid-structure interaction. J Fluids Struct 2020;97:103077.

[40] Rumsey C. SA expected results 2D zero pressure gradient flat plate. 2014, https://turbmodels.larc.nasa.gov/flatplate_sa.html.

[41] Jespersen DC, Pulliam TH, Childs ML. OVERFLOW: Turbulence modeling resource verification results. Technical report ARC-E-DAA-TN35216, NASA Ames Research Center; 2016.

[42] Nithiarasu P, Liu CB. An artificial compressibility based characteristic based split (CBS) scheme for steady and unsteady turbulent incompressible flows. Comput Methods Appl Mech Engrg 2006;195(23–24):2961–82.

[43] Panneer Selvam R. Finite element modelling of flow around a circular cylinder using LES. J Wind Eng Ind Aerodyn 1997;67–68:129–39.

[44] Saghafian M, Stansby PK, Saidi MS, Apsley DD. Simulation of turbulent flows around a circular cylinder using nonlinear eddy-viscosity modelling: Steady and oscillatory ambient flows. J Fluids Struct 2003;17(8):1213–36.

[45] Toro E. Riemann solvers and numerical methods for fluid dynamics. Springer-Verlag Berlin Heidelberg; 2009.

[46] Amestoy PR, Duff LS. Vectorization of a multiprocessor multifrontal code. Int J Supercomput Appl 1989;3(3):41–59.

[47] Amestoy PR, Duff lS. Memory management issues in sparse multifrontal methods on multiprocessors. Int J Supercomput Appl 1993;7(1):64–82.

[48] Kelley CT, Keyes DE. Convergence analysis of pseudo-transient continuation. SIAM J Numer Anal 1998;35(2):508–23.

[49] Bucker HM, Pullul B, Rasch A. On CFL evolution strategies for implicit upwind methods in linearized Euler equations. Internat J Numer Methods Fluids 2009;59:1–18.

[50] Mulder WA, Van Leer B. Experiments with implicit upwind methods for the Euler equations. J Comput Phys 1985;59(2):232–46.

[51] Crivellini A, D'Alessandro V, Bassi F. A spalart-allmaras turbulence model implementation in a discontinuos Galerkin solver for incompressible flows. J Comput Phys 2013;241:388–415.

[52] Crivellini A, Bassi F. An implicit matrix-free discontinuous Galerkin solver for viscous and turbulent aerodynamic simulations. Comput & Fluids 2011;50:81–93.

[53] Ceze M, Fidkowski K. Pseudo-transient continuation, solution update methods, and CFL strategies for DG discretizations of the RANS-SA equations. In: 21st AIAA computational fluid dynamics conference. 2013, p. 2686.

[54] Economon TD. The SU2 tutorial collection. 2019, https://su2code.github.io/tutorials/home/.