HAL open science

Anisotropic adaptive stabilized finite element solver for RANS models

Jessica Sari, Francesco Cremonesi, Mehdi Khalloufi, François Cauneau, Philippe Meliga, Youssef Mesri, Elie Hachem

To cite this version:

Jessica Sari, Francesco Cremonesi, Mehdi Khalloufi, François Cauneau, Philippe Meliga, et al.. Anisotropic adaptive stabilized finite element solver for RANS models. International Journal for Numerical Methods in Fluids, 2018, 86 (11), pp.717-736. 10.1002/fld.4475 . hal-02115828

HAL Id: hal-02115828

https://minesparis-psl.hal.science/hal-02115828v1

Submitted on 30 Apr 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Anisotropic adaptive stabilized finite element solver for RANS models

J. Sari[1], F. Cremonesi[1], M. Khalloufi[1], F. Cauneau[1], P. Meliga[1,2], Y. Mesri[1] and E. Hachem[1,*,†]

[1] *MINES ParisTech, PSL - Research University, CEMEF - Centre for material forming, CNRS UMR 7635, CS 10207 rue Claude Daunesse, 06904 Sophia-Antipolis Cedex, France.* [2]*Aix-Marseille Univ., CNRS, Centrale Marseille, M2P2, Marseille, France*

SUMMARY

Aerodynamic characteristics of various geometries are predicted using a finite element formulation coupled with several numerical techniques to ensure stability and accuracy of the method. First, an edge based error estimator and anisotropic mesh adaptation are used to detect automatically all flow features under the constraint of a fixed number of elements, thus controlling the computational cost. A Variational MultiScale stabilized finite element method is employed to solve the incompressible Navier-Stokes equations. Finally, the Spalart-Allmaras turbulence model is solved using the Streamline Upwind Petrov-Galerkin (SUPG) method. This paper is meant to show that the combination of anisotropic unsteady mesh adaptation with stabilized finite element methods provides an adequate framework for solving turbulent flows at high Reynolds numbers. The proposed method was validated on several test cases by confrontation with literature of both numerical and experimental results, in terms of accuracy on the prediction of the drag and lift coefficients as well as their evolution in time for unsteady cases.

KEY WORDS: Anisotropic mesh adaptation; Turbulent flows; Stabilized finite element method; Spalart-Allmaras; Boundary layers; Reynolds Averaged Navier-Stokes.

1. INTRODUCTION

In the aerospace industry, aerodynamic efficiency is one of the main components of the overall aircraft performance, and has thus a significant impact on the commercial success of aircraft.

†E-mail: elie.hachem@mines-paristech.fr
*Correspondence to: Elie Hachem, Mines ParisTech, France.

Consequently, airplane and airship designers are often concerned with accurately predicting aerodynamic forces and moments as a function of design parameters. In particular, the capability to determine the drag and lift coefficients for a given configuration is instrumental in the process of aircraft design. For this purpose, it is essential to reach a high level of accuracy and reliability in the description of the physical phenomena involved. For instance, one drag count (i.e. 10^{-4} in drag coefficient) is often cited as the desired accuracy for the drag prediction of subsonic transport aircraft [19] or the shape optimization of boats [30, 12]. However, the measurement of such quantities through experimental data can be often time and budget consuming as well as difficult to generalize, as such quantities depend strictly on the geometry at hand. For this reason, numerical simulations have become very important for the conceptual and design phases in the aerospace industry [22].

The high accuracy requirement for the prediction of integral coefficients in aerodynamic simulations implies the necessity to correctly resolve local flow features such as boundary layers or flow separation. For this purpose, locally-refined meshes are usually used. They incur a computational cost that is several orders of magnitude lower than globally-refined meshes [31]. However, the generation of locally-refined grids is an extremely time- and effort-consuming process that relies on the know-how and intuition of experts. It is not uncommon that engineers and technicians must re-create adapted meshes when simulations prove locally under-resolved. Mesh generation may amount to 80% of the overall analysis time in the automotive, aerospace, and ship building industries [21].

One remedy to these challenges lies in algorithms for automatic adaptation of unstructured meshes. A large variety of methods were proposed to optimize the compromise between accuracy and computational cost in CFD (Computational Fluid Dynamics). One of the most successful approaches involves a metric field that locally modifies the evaluation of spatial lengths depending on an error estimator. With this technique, the mesh can be adapted in a classical manner, that is, the local mesh size can be varied while maintaining element shapes as regular as possible. Above all, it also makes it possible to perform anisotropic adaptation: the element shapes can be controlled to fulfill a local mesh size requirement that varies depending on the direction. This capability is necessary for obtaining optimal meshes in fluid dynamics problems, where anisotropic flow features such as boundary layers are ubiquitous. Nevertheless, it implies that the numerical scheme perform adequately with highly stretched elements or cells, which is not the case of all numerical methods.

Methods exploiting a priori error bounds through the evaluation of the Hessian or the gradient of the solution have been developed in both the isotropic and anisotropic setting in [16, 32, 8]. Another class of anisotropic error estimates have also been obtained in [13, 14, 6], leading to the standardization of the anisotropic metric construction procedure. However, dynamic anisotropic mesh adaptation remains a challenge in the simulation of high Reynolds number flows past complex geometries. In such cases, Reynolds-Averaged Navier-Stokes (RANS) models are most often used, due to the prohibitive cost of turbulent scale-resolving approaches. With such models, local flow features such as boundary layers and separation need to be well captured by mesh refinement, and therefore the use of a priori adapted meshes often proves inefficient.

In this paper, we propose to solve RANS models by a stabilized finite element method combined with an efficient dynamic anisotropic mesh adaptation procedure. In particular, we focus on the Spalart-Allmaras [37] turbulence model. The mesh adaptation, driven by an efficient error estimator,

automatically captures thin boundary layers and flow separation points at high Reynolds number ($Re \sim 10^6$). The procedure makes it possible to accurately simulate three-dimensional turbulent flows around complex geometries with a reasonable computational cost.

In this work, particular attention is paid to the automatic character of the mesh adaptation process, with the objective of providing a useful and practical tool for aerodynamic applications. The proposed algorithm enables adaptive meshing under the constraint of a fixed number of nodes, which spares the end user from dealing with a case-dependent parameter linked to the error analysis. Moreover, the choice of the appropriate input for the error estimator is usually difficult when unknowns of different natures are solved simultaneously. Thus, we describe in this paper a multi-criteria functional that estimates the error on a combination of several variables including the velocity norm, all the velocity components as well as the turbulent viscosity.

In principle, the mesh adaptation procedure can be coupled to different spatial discretization schemes such as Finite Volumes or Finite Elements. However, the numerical method must be able to handle highly stretched simplicial elements. It is notoriously complicated to find schemes that exhibit the level of robustness required by automatic anisotropic mesh adaptation, especially among Finite Volume methods. We show here that stabilized Finite Element methods using appropriate stabilization parameters are adequate to deal with elements of aspect ratio up to the order of 1000. A Variational Multiscale formulation for the incompressible Navier-Stokes equations, as well as a Streamline Upwind Petrov-Galerking method used to solve the Spallart-Allmaras turbulence equation, are presented for this purpose [26].

The proposed framework is first verified using the well known NACA0012 airfoil benchmark. The results obtained for turbulent flows around cylinders with three different geometries are also presented and compared to data from the literature. The solver is then validated by comparing numerical predictions for the flow past an Unmanned Aerial Vehicle (UAV) with experimental measurements.

This paper is organized as follows: we start in section 2 with the description of the anisotropic mesh adaptation technique. Then, we present in Section 3 the governing equations and their discretization. Section 4 describes the numerical test cases showing the efficiency and the accuracy of the proposed method. Finally, Section 5 is dedicated to the conclusion and the future work.

2. ANISOTROPIC MESH ADAPTATION

The mesh adaptation strategy presented in this work relies on the *a posteriori* definition of a metric field able, in some specified sense, to drive the re-meshing procedure so that the interpolation error on the finite element solution is minimized. It relies on a statistical representation of the distribution of edges sharing a vertex, a quantity we call length distribution tensor. In order to relate the length distribution tensor to the interpolation error, following [10] we define an edge based error estimator based on a gradient recovery procedure. Once the optimal metric has been obtained, the mesh generation and adaptation procedure described in [27], based on a topological representation, is applied to obtain the new mesh.

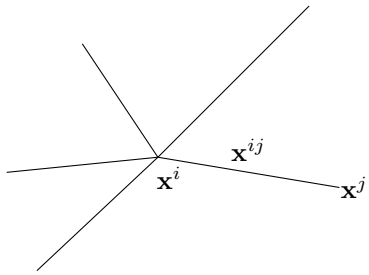


Figure 1. Patch associated with node $\mathbf{x}^i$.

2.1. *Definition of the length distribution tensor: a statistical representation*

Let $\Omega \subset \mathbb{R}^d$ be a polygonal domain, we consider a triangulation $\Omega = \bigcup_{K \in \mathcal{K}} K$ where K is a simplex such as a triangle or tetrahedron. Let $\Gamma(i)$ be the "patch" associated to a mesh vertex $\mathbf{x}^i$, defined as the set of nodes $\mathbf{x}^j$ which share one edge with $\mathbf{x}^i$, and let us denote by $\mathbf{x}^{ij}$ the edge connecting $\mathbf{x}^i$ to $\mathbf{x}^j$ as in Figure 1. The problem of finding a unitary metric $\mathbf{M}^i$ associated to the ith node can be formulated as the least squares problem [10]

$$\mathbf{M}^i = \underset{\mathbf{M} \in \mathbb{R}^{d \times d}_{sym}}{\operatorname{argmin}} \left(\sum_{j \in \Gamma(i)} \mathbf{M}\mathbf{x}^{ij} \cdot \mathbf{x}^{ij} - |\Gamma(i)| \right)^2 . \tag{1}$$

Provided that the vertices of $|\Gamma(i)|$ form at least d non co-linear edges with vertex $\mathbf{x}^i$ (which is the case in a valid mesh), then an approximate solution of (1) is given by

$$\mathbf{M}^i = \frac{1}{d} \left(\mathbf{X}^i \right)^{-1}, \tag{2}$$

where, denoting by $\otimes$ the tensor product between two vectors, we have introduced the *length distribution tensor*

$$\mathbf{X}^i = \frac{1}{|\Gamma(i)|} \left(\sum_{j \in \Gamma(i)} \mathbf{x}^{ij} \otimes \mathbf{x}^{ij} \right) \tag{3}$$

whose purpose is to give an "average" representation of the distribution of edges in the patch.

2.2. *Gradient recovery error estimator*

Let v_h be a P1 finite element approximation obtained by applying the Lagrange interpolation operator to a regular function $v \in C^2(\Omega)$. At node $\mathbf{x}^i$, we seek the recovered gradient $\mathbf{g}^i$ of v_h defined by

$$\mathbf{g}^i = \underset{\mathbf{g} \in \mathbb{R}^d}{\operatorname{argmin}} \sum_{j \in \Gamma(i)} \left| (\mathbf{g} - \nabla v_h) \cdot \mathbf{x}^{ij} \right|^2 = \underset{\mathbf{g} \in \mathbb{R}^d}{\operatorname{argmin}} \sum_{j \in \Gamma(i)} \left| \mathbf{g} \cdot \mathbf{x}^{ij} - \left(v_h(\mathbf{x}^j) - v_h(\mathbf{x}^i) \right) \right|^2 . \tag{4}$$

The solution to (4) can be expressed in terms of the length distribution tensor introduced before as

$$\mathbf{g}^i = \frac{1}{|\Gamma(i)|} \left(\mathbf{X}^i \right)^{-1} \sum_{j \in \Gamma(i)} \left(v_h(\mathbf{x}^j) - v_h(\mathbf{x}^i) \right) \mathbf{x}^{ij}. \tag{5}$$

It can be shown [10] that the quantity $|\mathbf{g}^i \cdot \mathbf{x}^{ij}|$ gives a second order accurate approximation of the second derivative of v along the edge $\mathbf{x}^{ij}$. Motivated by the fact that, for $P1$ finite elements on anisotropic meshes, edge residuals dominate a posteriori errors, as proved in [25], it is therefore suitable to define an error indicator function associated to the edge $\mathbf{x}^{ij}$ as

$$e^{ij} = \left|\mathbf{g}^{ij} \cdot \mathbf{x}^{ij}\right|. \tag{6}$$

Moreover, this quantity can be easily extended to account for several sources of error, instead of just the scalar field v_h, by applying formula (6) to each component separately.

2.3. Metric construction

In order to relate the error indicator defined in (6) to a metric suitable for a mesh adaptation procedure, we introduce the concept of stretching the factor s^{ij} defined as the ratio between the length of the edge $\mathbf{x}^{ij}$ after the adaptation procedure and before the adaptation procedure. The new metric, denoted by $\widetilde{\mathbf{M}}^i$, will then be given by

$$\widetilde{\mathbf{M}}^i = \frac{1}{d}\left(\widetilde{\mathbf{X}}^i\right)^{-1}; \quad \widetilde{\mathbf{X}}^i = \frac{1}{|\Gamma(i)|}\left(\sum_{j\in\Gamma(i)} (s^{ij})^2 \mathbf{x}^{ij} \otimes \mathbf{x}^{ij}\right) \tag{7}$$

To relate the metric to the interpolation error, following [23] we set a target total number of nodes N. Because of the quadratic behavior of the error as a function of the scaling factor, denoting by $\tilde{e}^{ij}$ the quantity defined in (6) computed after the mesh adaptation process we have

$$s^{ij} = \left(\frac{\tilde{e}^{ij}}{e^{ij}}\right)^{1/2}. \tag{8}$$

Moreover, the number of nodes in the new mesh resulting from the application of the scaling factor s^{ij} to the edge will be roughly equal to

$$N^{ij} = \frac{1}{s^{ij}}, \tag{9}$$

so that the total contribution of node i (in the old mesh) to the number of nodes in the new mesh will be given by

$$N^i = \det\left(\left(\mathbf{X}^i\right)^{-1} \sum_{j\in\Gamma(i)} N^{ij}\mathbf{x}^{ij} \otimes \mathbf{x}^{ij}\right) \tag{10}$$

By combining (8) and (9), it is possible to consider N^i in (10) is a function of $\tilde{e}^{ij}$. Assuming that the total error is equidistributed among all edges such that each edge contributes a constant error e to the total, one can see that $N^{ij} = \left(\frac{e^{ij}}{e}\right)^{1/2}$, which results in the relation

$$N^i(e) = e^{-\frac{d}{2}} N^i(1). \tag{11}$$

This article is protected by copyright. All rights reserved.

By summing over all the nodes of the old mesh, an expression for the total error as a function of the number of nodes in the new mesh is obtained. Inverting this equality yields

$$e = \left(\frac{\sum_i N^i(1)}{N} \right)^{\frac{2}{d}}, \tag{12}$$

and injecting this into (8) provides an expression for the scaling factors

$$s^{ij} = \frac{\left(\sum_i N^i(1) \right)^{\frac{1}{d}}}{N^{\frac{1}{d}} (e^{ij})^{\frac{1}{2}}}. \tag{13}$$

2.4. *Mesh adaption criteria*

In the preceding sections, we have explained how a metric filed driving the mesh adaptation can be derived from an edge-based error estimator under the constraint of a constant number of nodes. Nevertheless, this work is dedicated to the solution of Fluid Dynamics problems, which involve several unknowns, namely the velocity field $\mathbf{v}$ and the turbulent eddy viscosity $\tilde{\nu}$ (see Section 3).

The common way to adapt a mesh to several variables is to compute the metrics corresponding to each of them and then to produce a unique metric by intersection. The resulting metric should yield the largest mesh size in any direction that still fulfills the size constraint given by all the metrics. The intersection operation between several metric fields is not uniquely defined. The most common technique, consisting of a sequence of simultaneous reductions of two metrics [1], does not provide the optimal metric, is not commutative and incurs a relatively high computational cost. Robust techniques, based on the optimization problems, are even more costly.

In this work, we simplify this operation by using one field that accounts for different variables. Applying the error estimator (6) makes it possible to define a metric that to account for several sources of error. A vector of error sources is locally defined in two dimensions as

$$\mathbf{Y}(\mathbf{x}^i) = \left\{ \frac{v_x^i}{\|\mathbf{v}^i\|}, \frac{v_y^i}{\|\mathbf{v}^i\|}, \frac{\|\mathbf{v}^i\|}{\max_j \|\mathbf{v}^j\|}, \frac{\tilde{\nu}^i}{\max_j \tilde{\nu}^j} \right\}. \tag{14}$$

In three dimensions, a component $v_z^i / \|\mathbf{v}^i\|$ is added. The field used as input for the error estimator (6) is the norm $\left(\sum_k Y_k^2\right)^{1/2}$ of $\mathbf{Y}$, with Y_k the components of $\mathbf{Y}$. This definition measures the error in the norm as well as in the direction of the velocity vector $\mathbf{v}$, in addition to the error on the turbulent eddy viscosity $\tilde{\nu}$. Because all fields are normalized (the velocity components v_x, v_y and v_z by the local velocity norm, the velocity norm $\|\mathbf{v}\|$ and the turbulent eddy viscosity $\tilde{\nu}$ by their respective global maxima), a field that is much larger in absolute value does not dominate the error estimator, and the variations of all variables are fairly taken into account.

3. GOVERNING EQUATIONS AND NUMERICAL SCHEME

In this section, we present the governing equations, namely the unsteady incompressible Navier-Stokes equations coupled to the Spalart-Allmaras turbulence model, and their discretization. Time splitting is applied to the system, so that the Navier-Stokes and the Spalart-Allmaras transport equation can be solved separately using different numerical methods. At each time step, the Navier-Stokes equations are solved in a first stage using the value of the turbulent variable from the preceding time step, and the resulting velocity filed is used to solve the Spallart-Allmaras equation in a second stage.

The stabilized Finite Element scheme for the Navier-Stokes equations is derived from a variational multiscale point of view. Both the velocity and the pressure spaces are enriched which cures the spurious oscillations in the convection-dominated regime and deals with the pressure instability. The Spalart-Allmaras transport equation is solved through a Streamline Upwind Petrov-Galerkin (SUPG) method. In both schemes, particular attention is paid to the determination of the stabilization parameters in the presence of mesh anisotropy, through the use of a directional element size.

3.1. The Navier-Stokes equations

To fix notations, let $\Omega \subset \mathbb{R}^d$ be the fluid domain, where d is the space dimension, and $\partial\Omega$ its boundary. The strong form of the incompressible Navier Stokes equations reads:

$$\begin{cases} \rho\left(\partial_t \mathbf{v} + \mathbf{v}\cdot\nabla\mathbf{v}\right) - \nabla\cdot\boldsymbol{\sigma} = \mathbf{f} \\ \nabla\cdot\mathbf{v} = 0 \end{cases} \tag{15}$$

where $t \in [0, T]$ is the time, $\mathbf{v}(\mathbf{x}, t)$ the velocity, $p(\mathbf{x}, t)$ the pressure and ρ the density. The Cauchy stress tensor for a Newtonian fluid is given by:

$$\boldsymbol{\sigma} = 2\mu\,\boldsymbol{\epsilon}(\mathbf{v}) - p\,\mathbf{I}_d, \tag{16}$$

with $\mathbf{I}_d$ the d-dimensional identity tensor and μ the dynamic viscosity. In order to close the problem, Equations (15) are subject to the boundary and initial conditions to be specified later.

The Reynolds-Averaged Navier-Stokes (RANS) equations are obtained by applying a time-averaging procedure to the Navier-Stokes equations, which gives rise to an additional stress term (the Reynolds stress tensor) taking into account the effect of turbulent fluctuations on the averaged flow. The Boussinesq approximation can then be used to model this additional stress as a turbulent eddy viscosity μ_{t}. The RANS equations, with Reynolds-averaged velocity and pressure as unknowns, have the same form as the Navier-Stokes equations, except that the molecular viscosity μ is replaced by the sum $\mu + \mu_{\mathrm{t}}$. The eddy viscosity μ_{t} is obtained from a model involving additional partial differential equations. We choose for this work the one-equation model described in Section 3.2. Hereafter, we use the same notations for the Navier-Stokes and RANS equations and denote the total viscosity as μ for the sake of simplicity.

The weak form of problem (15) combined with (16) can be obtained by multiplication of a test function and integration by parts. Let $H^1(\Omega)$ be the Sobolev space of square integrable functions whose distributional derivatives are square integrable, and let $V \subset \left[H^1(\Omega)\right]^d$ be a functional space

properly chosen according to the boundary conditions. Finally, let $Q = \left\{q \in L^2(\Omega) : \int_\Omega q = 0\right\}$. By denoting $(\cdot,\cdot)$ the scalar product of the space $L^2(\Omega)$, the weak form of problem (15) on $\partial\Omega$ reads, under the assumption of homogeneous Dirichlet boundary conditions:

$$\begin{cases} \text{Find } (\mathbf{v}, p) \in V \times Q \text{ such that:} \\ \rho\ [(\partial_t \mathbf{v}, \mathbf{w}) + (\mathbf{v} \cdot \nabla \mathbf{v}, \mathbf{w})] + (2\mu \boldsymbol{\epsilon}(\mathbf{v}) : \boldsymbol{\epsilon}(\mathbf{w})) - (p, \nabla \cdot \mathbf{w}) = (\mathbf{f}, \mathbf{w}), \quad \forall \mathbf{w} \in V \\ (\nabla \cdot \mathbf{v}, q) = 0, \quad \forall q \in Q. \end{cases} \tag{17}$$

where ρ and μ are the density and the dynamic viscosity, respectively. Based on a mesh $\mathcal{K}_h$ of Ω made of N_{el} elements K, the functional spaces for the velocity V and for the pressure Q are approximated by the finite dimensional spaces V_h and Q_h respectively. It is well known that the stability of the semi-discrete formulation requires an appropriate choice of the finite element spaces V_h and Q_h, that must to fulfill a compatibility condition [11]. Accordingly, the standard Galerkin method with the P1/P1 element (i.e. the same piecewise linear space for V_h and Q_h) is not stable. Moreover, convection-dominant problems (i.e. problems where the convection term $\mathbf{v} \cdot \nabla \mathbf{v}$ is much larger than the diffusion term $\nabla \cdot (2\mu\, \boldsymbol{\epsilon})$) also lead to a loss of coercivity in formulation (17). This phenomenon manifests itself as oscillations that pollute the solution. In this work, we use a Variational MultiScale method [20] which circumvents both problems through a Petrov-Galerkin approach. The basic idea is to consider that the unknowns can be split into two components, a coarse one and a fine one, corresponding to different scales or levels of resolution. First, we solve the fine scales in an approximate manner and then we replace their effect into the large-scale equation. We present here only an outline of the method, and the reader is referred to [17] for extensive details about the formulation.

Let us split the velocity and the pressure fields into resolvable coarse-scale and unresolved fine-scale components: $\mathbf{v} = \mathbf{v}_h + \mathbf{v}'$ and $p = p_h + p'$. The same decomposition can be applied to the weighting functions: $\mathbf{w} = \mathbf{w}_h + \mathbf{w}'$ and $q = q_h + q'$. Subscript h is used hereafter to denote the finite element (coarse) component, whereas the prime is used for the so called subgrid scale (fine) component of the unknowns. The enrichment of the functional spaces is performed as follows: $V = V_h \oplus V'$, $V_0 = V_{h,0} \oplus V_0'$ and $Q = Q_h \oplus Q'$. Thus, the finite element approximation for the time-dependent Navier-Stokes problem reads:

$$\begin{cases} \text{Find}(\mathbf{v}, p) \in V \times Q \text{ such that:} \\ \rho\left(\partial_t(\mathbf{v}_h + \mathbf{v}'), (\mathbf{w}_h + \mathbf{w}')\right) + \rho\left((\mathbf{v}_h + \mathbf{v}') \cdot \nabla(\mathbf{v}_h + \mathbf{v}'), (\mathbf{w}_h + \mathbf{w}')\right) \\ \qquad\qquad + \left(2\mu\boldsymbol{\varepsilon}(\mathbf{v}_h + \mathbf{v}') : \boldsymbol{\varepsilon}(\mathbf{w}_h + \mathbf{w}')\right) \\ \qquad\qquad - \left((p_h + p'), \nabla \cdot (\mathbf{w}_h + \mathbf{w}')\right) = \left(\mathbf{f}, (\mathbf{w}_h + \mathbf{w}')\right), \quad \forall \mathbf{w} \in V_0 \\ \left(\nabla \cdot (\mathbf{v}_h + \mathbf{v}'), (q_h + q')\right)_\Omega = 0, \quad \forall q \in Q. \end{cases} \tag{18}$$

To derive the stabilized formulation, we split Equations (18) into a large-scale and a fine-scale problem. The fine-scale problem is defined on element interiors. Under several assumptions about the time-dependency and the non-linearity of the momentum equation of the subscale system detailed in [17], the fine-scale solutions $\mathbf{v}'$ and p' written in terms of the time-dependent large-scale variables using residual-based terms that are derived consistently. Consequently, we can use static condensation, that consists in substituting directly $\mathbf{v}'$ and p' into the large-scale problem. This

gives rise to additional terms in the Finite Element formulation, that are tuned by a local stabilizing parameter. These terms are responsible for the enhanced stability compared to the standard Galerkin formulation. The large-scale system finally reads:

$$\begin{cases} \rho\left(\partial_t \mathbf{v}_h, \mathbf{w}_h\right)_\Omega + \left(\rho \mathbf{v}_h \cdot \nabla \mathbf{v}_h, \mathbf{w}_h\right)_\Omega \\ \qquad - \sum_{K\in\mathcal{T}_h}\left(\tau_1 \boldsymbol{\mathcal{R}}_{\mathrm{M}}, \rho \mathbf{v}_h \nabla \mathbf{w}_h\right)_K + \left(2\mu \boldsymbol{\varepsilon}(\mathbf{v}_h) : \boldsymbol{\varepsilon}(\mathbf{w}_h)\right)_\Omega \\ \qquad - \left(p_h, \nabla \cdot \mathbf{w}_h\right)_\Omega - \sum_{K\in\mathcal{T}_h}\left(\tau_2 \mathcal{R}_{\mathrm{C}}, \nabla \cdot \mathbf{w}_h\right)_K = \left(\mathbf{f}, \mathbf{w}_h\right)_\Omega, \quad \forall \mathbf{w}_h \in V_{h,0} \\ \left(\nabla \cdot \mathbf{v}_h, q_h\right)_\Omega - \sum_{K\in\mathcal{T}_h}\left(\tau_1 \boldsymbol{\mathcal{R}}_{\mathrm{M}}, \nabla q_h\right)_K = 0, \quad \forall q_h \in Q_h \end{cases} \tag{19}$$

where $(\cdot,\cdot)_\Omega$ represents the scalar product on the whole domain omega while $(\cdot,\cdot)_K$ is the scalar product on Element K. The quantities τ_1 and τ_2 are stabilization parameters defined hereafter. The momentum residual $\boldsymbol{\mathcal{R}}_{\mathrm{M}}$ and the continuity residual $\mathcal{R}_{\mathrm{C}}$ are expressed as:

$$\begin{aligned} \boldsymbol{\mathcal{R}}_{\mathrm{M}} &= \mathbf{f} - \rho \partial_t \mathbf{v}_h - \rho \mathbf{v}_h \cdot \nabla \mathbf{v}_h - \nabla p_h \\ \mathcal{R}_{\mathrm{C}} &= -\nabla \cdot \mathbf{v}_h \end{aligned} \tag{20}$$

Compared to the standard Galerkin method, the proposed stable formulation involves additional integrals that are evaluated element-wise. These additional terms represent the stabilizing effect of the sub-grid scales and are introduced in a consistent way in the Galerkin formulation. They make it possible to avoid instabilities caused by both dominant convection terms and incompatible approximation spaces. All of these terms are controlled by the stabilization parameters τ_1 and τ_2, for which we adopt the definition proposed in [9]:

$$\tau_1 = \left[\left(\frac{2\rho\|\mathbf{v}_h\|_K}{h_K}\right)^2 + \left(\frac{4\mu}{h_K^2}\right)^2\right]^{-\frac{1}{2}}, \tag{21}$$

$$\tau_2 = \left[\left(\frac{\mu}{\rho}\right)^2 + \left(\frac{c_2\|\mathbf{v}_h\|_K}{c_1 h_K}\right)^2\right]^{\frac{1}{2}} \tag{22}$$

where h_K is the characteristic length of the element and c_1 and c_2 are algorithmic constants. We take them as $c_1 = 4$ and $c_2 = 2$ for linear elements [9].

Equations (20) are discretized in time by a semi-implicit scheme. The convective term, the viscous term and the pressure term in the momentum equation, as well the divergence term in the continuity equation, are integrated implicitly through a backward Euler scheme. All other contributions (i.e. the source term and the stabilization terms) are integrated explicitly by a forward Euler scheme. Due to the split between the Navier-Stokes and the Spalart-Allmaras equations, the dependence of the total viscosity μ on the turbulent viscosity μ_t is also explicit. Among the implicit terms, the convection operator $\mathbf{v}_h \cdot \nabla \mathbf{v}_h$ is non-linear, thus a Newton root finding algorithm is applied to the system at each time step. Using the superscript n to denote the value of the quantities at time t_n and $n+1, i$ to denote the value of the quantities for time t_{n+1} at the i-th iteration of the Newton

method, the system to be solved for $(\mathbf{v}_h^{n+1,i+1}, p_h^{n+1,i+1})$ at each iteration i reads:

$$\begin{cases} \rho\left(\frac{\mathbf{v}_h^{n+1,i+1}-\mathbf{v}_h^n}{\Delta t}, \mathbf{w}_h\right)_\Omega \\ \quad + \left(\rho\left[\mathbf{v}_h^{n+1,i}\cdot\nabla\mathbf{v}_h^{n+1,i+1} + \mathbf{v}_h^{n+1,i+1}\cdot\nabla\mathbf{v}_h^{n+1,i} - \mathbf{v}_h^{n+1,i}\cdot\nabla\mathbf{v}_h^{n+1,i}, \mathbf{w}_h\right]\right)_\Omega \\ \quad - \sum_{K\in\mathcal{T}_h}\left(\tau_1^n\boldsymbol{\mathcal{R}}_{\mathrm{M}}^n, \rho\mathbf{v}_h^n\nabla\mathbf{w}_h\right)_K + \left(2\mu^n\boldsymbol{\varepsilon}(\mathbf{v}_h^{n+1,i+1}) : \boldsymbol{\varepsilon}(\mathbf{w}_h)\right)_\Omega \\ \quad - \left(p_h^{n+1,i+1}, \nabla\cdot\mathbf{w}_h\right)_\Omega - \sum_{K\in\mathcal{T}_h}\left(\tau_2^n\mathcal{R}_{\mathrm{C}}^n, \nabla\cdot\mathbf{w}_h\right)_K = (\mathbf{f}^n, \mathbf{w}_h)_\Omega, \quad \forall\mathbf{w}_h\in V_{h,0} \\ \left(\nabla\cdot\mathbf{v}_h^{n+1,i+1}, q_h\right)_\Omega - \sum_{K\in\mathcal{T}_h}\left(\tau_1^n\boldsymbol{\mathcal{R}}_{\mathrm{M}}^n, \nabla q_h\right)_K = 0, \quad \forall q_h\in Q_h. \end{cases} \tag{23}$$

We usually precondition the linear system with a block Jacobi method supplemented by a incomplete LU factorization, and solve it by means of a GMRES algorithm.

3.2. The Spalart-Allmaras turbulence model (SA)

The turbulence model chosen to compute the eddy viscosity is the one-equation Spalart-Allmaras (SA) turbulence model. The SA model [37] represents the evolution of the kinematic eddy viscosity $\tilde{\nu}$ by a a non linear advection-diffusion-reaction equation:

$$\frac{\partial\tilde{\nu}}{\partial t} + \mathbf{v}\cdot\nabla\tilde{\nu} - c_{b1}(1-f_{t2})\tilde{S}\tilde{\nu} + \left[c_{w1}f_w - \frac{c_{b1}}{\kappa^2}f_{t2}\right]\left(\frac{\tilde{\nu}}{d}\right)^2 - \frac{c_{b2}}{\sigma}\nabla\tilde{\nu}\cdot\nabla\tilde{\nu} - \frac{1}{\sigma}\nabla\cdot[(\nu+\tilde{\nu})\nabla\tilde{\nu}] = 0 \tag{24}$$

The eddy viscosity can then be obtained from $\mu_{\mathrm{t}} = \rho\tilde{\nu}f_{v1}$, with:

$$\begin{gathered} f_{v1} = \frac{\chi^3}{\chi^3+c_{v1}^3}, \qquad \chi = \frac{\tilde{\nu}}{\nu}, \qquad f_{v2} = 1 - \frac{\chi}{1+\chi f_{v1}} \qquad f_{t2} = c_{t3}e^{-c_{t4}\chi^2} \\ f_w = g\left[\frac{1+c_{w3}^6}{g^6+c_{w3}^6}\right]^{\frac{1}{6}}, \qquad g = r + c_{w2}(r^6 - r), \qquad r = \frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2}, \\ \tilde{S} = S + \frac{\tilde{\nu}}{\kappa^2 d^2}f_{v2}, \qquad S = \sqrt{2\boldsymbol{\varepsilon}(\mathbf{v}) : \boldsymbol{\varepsilon}(\mathbf{v})}. \end{gathered} \tag{25}$$

where d is the shortest distance to the wall, $\kappa = 0.4$ is the von Kármán constant, and the remaining model coefficients are:

$$\begin{gathered} c_{b1} = 0.1355, \qquad c_{b2} = 0.622, \qquad \sigma = 2/3, \qquad c_{v1} = 7.1, \qquad c_{v2} = 0.7, \qquad c_{v3} = 0.9, \\ c_{w1} = \frac{c_{b1}}{\kappa} + \frac{1+c_{b2}}{\sigma}, \qquad c_{w2} = 0.3, \qquad c_{w3} = 2, \qquad c_{t3} = 1.2, \qquad c_t = 0.5. \end{gathered} \tag{26}$$

In order to improve accuracy and convergence, some modifications may be performed, in particular to avoid the generation of negative viscosity values. There exist in the literature many variations for the SA model, most of which are collected in NASA's *turbulence modeling resource* webpage [33]. In this work, the *Negative Spalart Allmaras Model* is used because of its capability to avoid the generation of negative turbulent viscosity without the use of clipping [2]. It consists in

replacing (24) when $\tilde{\nu}$ is negative by:

$$\frac{\partial \tilde{\nu}}{\partial t} + \mathbf{v} \cdot \nabla \tilde{\nu} - c_{b1}(1 - c_{t3}) S \tilde{\nu} - c_{w1} \left(\frac{\tilde{\nu}}{d} \right)^2 - \frac{c_{b2}}{\sigma} \nabla \tilde{\nu} \cdot \nabla \tilde{\nu} - \frac{1}{\sigma} \nabla \cdot [(\nu + f_n \tilde{\nu}) \nabla \tilde{\nu}] = 0, \quad (27)$$

with $f_n = (c_{n1} + \chi^3)/(c_{n1} - \chi^3)$ and $c_{n1} = 16$. Moreover, the turbulent eddy viscosity μ_{t} is set to zero when $\tilde{\nu}$ is negative.

A stabilized Finite Element discretization of the Spallart-Allmaras model is proposed in [24]. Following a similar idea, we recast Equation (24) into a convection-diffusion-reaction form, and apply a backward Euler time discretization:

$$\frac{\tilde{\nu}^{n+1} - \tilde{\nu}^n}{\Delta t} + \underbrace{\left(\mathbf{v}^{n+1} - \frac{c_{b2}}{\sigma} \nabla \tilde{\nu}^{n+1} \right) \cdot \nabla \tilde{\nu}^{n+1}}_{\text{convection}} - \underbrace{\frac{1}{\sigma} \nabla \cdot \left[(\nu + \tilde{\nu}^{n+1}) \nabla \tilde{\nu}^{n+1} \right]}_{\text{diffusion}}$$
$$- \underbrace{\left[c_{b1}(1 - f_{t2}^{n+1}) \tilde{S}^{n+1} + \left(c_{w1} f_w^{n+1} - \frac{c_{b1}}{\kappa^2} f_{t2}^{n+1} \right) \frac{\tilde{\nu}^{n+1}}{d^2} \right] \tilde{\nu}^{n+1}}_{\text{reaction}} = 0, \quad (28)$$

where $\tilde{\nu}^n$ stands for the value of $\tilde{\nu}$ at discrete time t_n. We recall that the value of the velocity field at time t_{n+1} is computed before the Spallart-Allmaras equation at the same time step, so the quantities $\mathbf{v}^{n+1}$, $\tilde{S}^{n+1}$, f_{t2}^{n+1} and f_w^{n+1} are explicitly available when solving Equation (28).

Equation (28) is non-linear. Following [24], we apply the non-linear root finding algorithm at semi-discrete level, but we chose to work with a simpler Picard-like linearization instead of the Newton method employed in [24]. Using the superscript $\tilde{\nu}^{n+1,i}$ for the value of $\tilde{\nu}^{n+1}$ at the i-th iteration, an iteration of the non-linear root search reads:

$$\frac{\tilde{\nu}^{n+1,i+1} - \tilde{\nu}^n}{\Delta t} + \left(\mathbf{v}^{n+1} - \frac{c_{b2}}{\sigma} \nabla \tilde{\nu}^{n+1,i} \right) \cdot \nabla \tilde{\nu}^{n+1,i+1} - \frac{1}{\sigma} \nabla \cdot \left[(\nu + \tilde{\nu}^{n+1,i}) \nabla \tilde{\nu}^{n+1,i+1} \right]$$
$$- \left[c_{b1}(1 - f_{t2}) \tilde{S}^{n+1} + \left(c_{w1} f_w - \frac{c_{b1}}{\kappa^2} f_{t2} \right) \frac{\tilde{\nu}^i}{d^2} \right] \tilde{\nu}^{n+1,i+1} = 0, \quad (29)$$

Equation (29) is then discretized in space using a Streamline Upwind Petrov-Galerkin (SUPG) method. The Galerkin formulation is obtained by multiplying (29) by appropriate test functions $\omega_h \in W_h$, applying the divergence theorem to the diffusion terms and integrating by parts. Following the lines in [3] on the use of stabilization methods for transient convection-diffusion-reaction equations, the stabilized weak form of equation (29) reads:

$$\left(\frac{\tilde{\nu}^{n+1,i+1} - \tilde{\nu}^n}{\Delta t}, \omega_h \right)_\Omega +$$
$$\left(\left[\mathbf{v}^{n+1} - \frac{c_{b2}}{\sigma} \nabla \tilde{\nu}^{n+1,i} \right] \cdot \nabla \tilde{\nu}_h^{n+1,i+1}, \omega_h \right)_\Omega - \left(\frac{1}{\sigma} (\nu + \tilde{\nu}_h^i) \nabla \tilde{\nu}_h^{n+1,i+1}, \nabla \omega_h \right)_\Omega$$
$$- \left(\left[c_{b1}(1 - f_{t2}^{n+1}) \tilde{S}_h^{n+1} + \left(c_{w1} f_w^{n+1} - \frac{c_{b1}}{\kappa^2} f_{t2}^{n+1} \right) \frac{\tilde{\nu}_h^{n+1,i}}{d^2} \right] \tilde{\nu}_h^{n+1,i+1}, \omega_h \right)_\Omega$$
$$+ \sum_K \left(\mathcal{R}(\tilde{\nu}^{n+1,i}), \tau_3^{n+1,i} \left[\mathbf{v}_h^{n+1} - \frac{c_{b2}}{\sigma} \nabla \tilde{\nu}_h^{n+1,i} \right] \cdot \nabla \omega_h \right)_K = 0, \quad \forall \omega_h \in W_h. \quad (30)$$

This article is protected by copyright. All rights reserved.

where $\mathcal{R}(\tilde{\nu})$ is the finite element residual of (29). The stabilization parameter τ_3, is computed within each element as:

$$\tau_3 = \left(\frac{c_2}{h} \|\alpha_c\|_K + \frac{c_1}{h_K^2} \alpha_d + \alpha_r \right)^{-1} \tag{31}$$

where α_c, α_d and α_r are respectively the convection, diffusion and reaction coefficients in Equation (30), h_K is the element size, $\|\alpha_c\|_K$ a characteristic norm of the convection term and $c_1 = 4$, $c_2 = 2$ for linear elements.

The linear system arising from Equation (30) is solved using the same numerical method as for the Navier-Stokes equations (see Section 3.1).

3.3. Element size measures in stabilization parameters

Recall that the coefficients ($\tau_{1,2,3}$) weight the stabilization terms added to the weak formulations (19) and (30). They are defined for each element K of the triangulation and depend on the local mesh size h_K. Many numerical experiments show that good results can be obtained when using the minimum edge length of K [29], while others always use the triangle diameter (see [28] for details).

However, in the case of strongly anisotropic meshes with highly stretched elements, the definition of h_K is still an open problem and plays a critical role in the design of the stabilizing coefficients [18, 9]. In [15], the authors examine deeply the effect of different element length definitions on distorted meshes. In [7], anisotropic error estimates for the residual free bubble (RFB) method are developed to derive a new choice of the stabilizing parameters suitable for anisotropic partitions. In this work, we adopt the definition proposed in [39] to compute h_K as the size of K in the direction of the velocity:

$$h_K = \frac{2|\mathbf{v}_h|}{\Sigma_{i=1}^{N_K} |\mathbf{v}_h \cdot \nabla \varphi_i|} \tag{32}$$

where N_K is the number of vertices of K and $\varphi_1, ..., \varphi_{N_K}$ are the usual basis functions of $P_1(K)$ mapped onto K.

4. NUMERICAL EXPERIMENTS

This section describes the numerical tests conducted to assess the correctness and the efficiency of the method. We compare results obtained for steady and unsteady 2D classical benchmarks such as the flow around a prismatic cylinder and the NACA 0012 airfoil with data from the literature. For all test cases, we use different type of meshes and number of elements to make adequate comparisons. Finally, we present a more complex application, namely the flow past the 3D geometry of a UAV.

4.1. Evaluation of aerodynamic forces

The aerodynamic forces, which are the quantities of prime engineering interest in an aerodynamic simulation, are most often expressed in terms of the lift coefficient C_L and the drag coefficient C_D

defined as

$$
\begin{aligned}
C_D &= \frac{2F_D}{\rho \bar{U}^2 A} \\
C_L &= \frac{2F_L}{\rho \bar{U}^2 A}
\end{aligned}
\tag{33}
$$

where A is a reference area, F_D is the drag force oriented in the direction $\mathbf{e_x}$ parallel to the flow and F_L is the lift force oriented in a direction $\mathbf{e_y}$ normal to the flow. In 2D, these coefficients are usually expressed per unit of span length by substituting a reference length L for the reference area A.

The total aerodynamic force exerted by the flow on an object of boundary $S \in \partial\Omega$ is given by

$$
\mathbf{F}_{aero} = \int_S \boldsymbol{\sigma}\mathbf{n}
$$

so

$$
\begin{aligned}
F_D &= \mathbf{F}_{aero} \cdot \mathbf{e_x} = \int_S (\boldsymbol{\sigma}\mathbf{n}) \cdot \mathbf{e_x} \\
F_L &= \mathbf{F}_{aero} \cdot \mathbf{e_y} = \int_S (\boldsymbol{\sigma}\mathbf{n}) \cdot \mathbf{e_y}
\end{aligned}
\tag{34}
$$

We use the variational technique described in [41] to compute the components F_D and F_L of $\mathbf{F}_{aero}$. Starting with the drag force, let us consider the Sobolev space $[H^1_{\partial\Omega\setminus S}(\Omega)]^D$ of functions supported in Ω that are equal to $\mathbf{e_x}$ on S and that vanish on all other boundaries of Ω. A variational form of the momentum conservation Equation (15) can be obtained by multiplying by a test function and integrating by parts:

$$
\int_\Omega \partial_t \mathbf{v} \cdot \mathbf{w} + \int_\Omega \boldsymbol{\sigma} : \nabla \mathbf{w} + \int_{\partial\Omega} (\boldsymbol{\sigma}\mathbf{n}) \cdot \mathbf{w} + \int_\Omega (\mathbf{v} \cdot \nabla)\, \mathbf{v} \cdot \mathbf{w} = \mathbf{0}, \quad \forall \mathbf{w} \in [H^1_{\partial\Omega\setminus S}(\Omega)]^D \tag{35}
$$

where $\mathbf{n}$ is the normal pointing inside the computational domain. Considering that $\mathbf{w}|_{\partial\Omega\setminus S} = \mathbf{0}$, the third term in (35) reduces to F_D, and rearranging the terms yields

$$
F_D = -\int_\Omega \partial_t \mathbf{v} \cdot \mathbf{w} - \int_\Omega \boldsymbol{\sigma} : \nabla \mathbf{w} - \int_\Omega (\mathbf{v} \cdot \nabla)\, \mathbf{v} \cdot \mathbf{w}, \quad \forall \mathbf{w} \in [H^1_{\partial\Omega\setminus S}(\Omega)]^D. \tag{36}
$$

In practice, we use for $\mathbf{w}$ the piecewise linear function that is equal to $\mathbf{e_x}$ at the nodes lying on S and $\mathbf{0}$ at every other nodes, i.e. the function that vanishes everywhere except on the first layer of elements around S. The lift force can be obtained in the same manner, substituting $\mathbf{e_y}$ for $\mathbf{e_x}$.

4.2. Flow past a 2D cylinder

As a first numerical test, we disregard the turbulence model ($\mu_t = 0$) and consider the classical benchmark proposed in [34], that is, the laminar flow past a cylinder located slightly asymmetrically in a channel. The main point of this test is to demonstrate the ability of the mesh adaptation framework to capture local flow features. Also, it assesses the capacity of the method described

Table I. Drag and lift coefficient values with literature for fixed mesh (Test 1) and adaptive mesh (Test 2).

	C_D	C_L
Schäfer and Turek [34]	5.58 ± 0.01	0.0107 ± 0.0003
Test 1	5.57	0.0098
Test 2	5.56	0.0099

in Section 4.1 to correctly evaluate the aerodynamic coefficients C_D and C_L on highly anisotropic meshes. The detailed geometry, parameters, flow conditions and boundary conditions are described in [34].

The first set of simulations corresponds to the steady case with Reynolds number $Re = 20$. Two tests are performed, starting with a uniform solution and marching in time until the steady state is reached. In the first test, a fixed fine mesh is used. In the second one, dynamic mesh adaptation is triggered every a given frequency (i.e. here we fixed it at 10 iterations) under the constraint of a fixed number of nodes $N = 40,000$, starting from an arbitrary mesh.

Recall that a multi-criteria adaptivity is used in this case. As explained in Section 2.4, each component of the velocity field and its norm were used to account for several sources of error.

The final drag and lift coefficients are listed in Table I, where they are compared with the reference data from [34]. Very good agreement is found (about 0.2% in drag and 8% in lift).

To highlight the capability of the proposed adapted method, we repeated the same case with a Reynolds number equal to 100. The same dynamic mesh adaptation parameters as for the steady case are used. The velocity norm and the corresponding mesh at several points in time are shown in the upper part of Fig. 2. As expected, the boundary layers on the cylinder and on the channel walls, as well as the flow detachments, are captured automatically. The maximum drag and lift (i.e. 3.21 and 0.94 respectively) are again in good agreement with the reference.

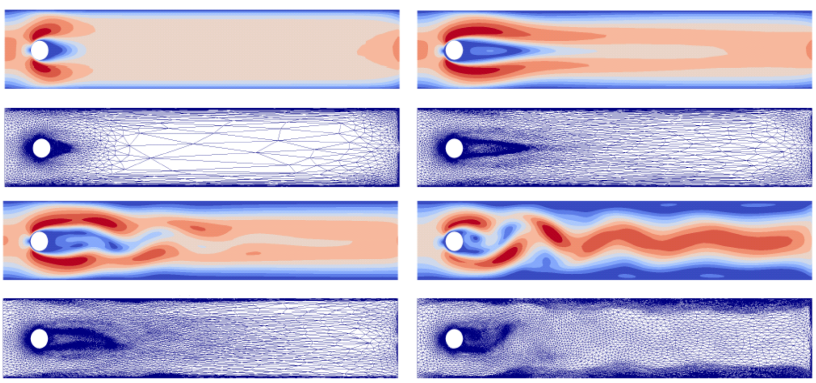


Figure 2. Dynamic anisotropic mesh adaptation and fluid flow past a fixed cylinder.

4.3. Turbulent flow around prismatic cylinders

We turn now to a case with fully turbulent flow. Non-smooth cylindrical geometries are often employed to assess stabilized finite element methods, because the sharp and localized flow features that they develop challenge the stabilization techniques. For the same reason, they represent a good test for mesh adaptation procedures. In this Section, we focus on the test cases proposed in [4]. They

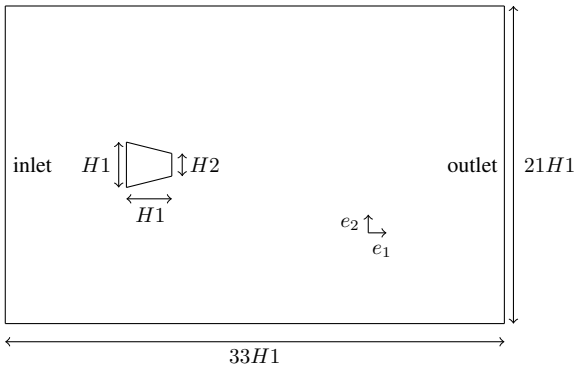


Figure 3. Geometry for cylinder case in section 4.3.

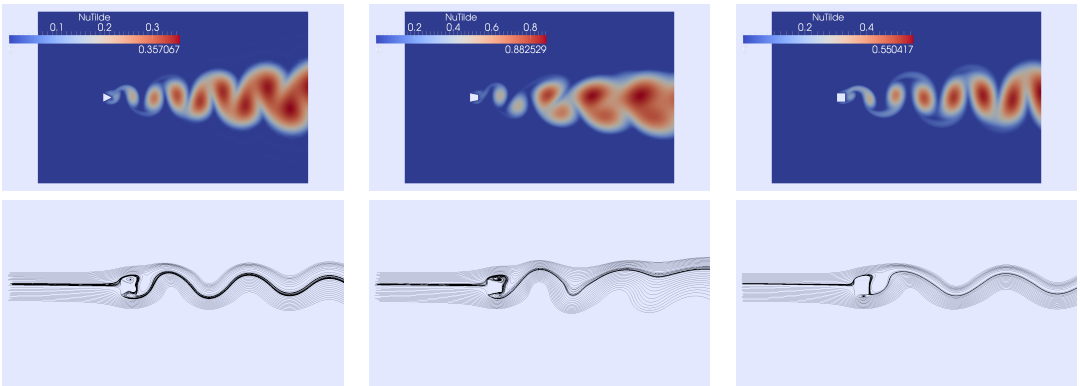


Figure 4. Plots of $\tilde{\nu}$ (*top*) and streamlines (*bottom*) for Configurations (a) (left), (b) (center) and (c) (right).

involve geometries of 2D primatic cylinders that are parametrized by lengths H_1 and H_2 as shown in Figure 3. The computational domain is $33H_1$ long and $21H_1$ wide. The cylinder is centered in width, and its front side is located $8H_1$ downstream of the inlet boundary. Three configurations (a), (b) and (c), corresponding to aspect ratios H_2/H_1 of 0, 0.6 and 1 respectively, are considered. The values of the inlet velocity V_{in} and the dynamic viscosity μ are chosen to yield a Reynolds number based on H_1 equal to $Re = 2.2 \times 10^4$.

The inflow boundary conditions are $\mathbf{v} = (V_{in}, 0), \tilde{\nu} = 3\nu$, as recommended in [38]. Symmetry conditions are imposed on the outer boundaries parallel to the flow direction as $v_y = 0$, $\partial v_x/\partial y = 0$ and $\partial\tilde{\nu}/\partial y = 0$. For the outflow, $p \cdot \mathbf{n} - (\mu + \mu_{\mathrm{t}}) \left(\nabla\mathbf{v} + \nabla^T\mathbf{v}\right) \cdot \mathbf{n} = 0$ and $\nabla\tilde{\nu} \cdot \mathbf{n} = 0$ are prescribed, whereas for the rigid body no slip conditions are imposed as $\mathbf{v} = \mathbf{0}$ and $\tilde{\nu} = 0$. The non-dimensional time step is chosen equal to $\Delta t = 0.002$. We are interested in assessing the evolution of the drag and lift coefficients over time, once a condition of periodic shedding of vortices has been reached. These quantities are computed by means of the method described in Section 4.1.

The plots in the upper part of Figure 4 depict the turbulent variable $\tilde{\nu}$ at a time in the periodic regime of the flow for the three different geometries with fixed meshes. It is clear that the turbulence model is only active in the restricted area of interest, and the effects of the averaging process and damping function seem to be correctly taken into account. The streamlines, shown in the bottom part of Figure 4, are similar to those given in [4] in the cylinder wake.

Table II. Comparisons of drag and lift coefficients with the literature for Configurations (b) and (c).

	mean C_D	r.m.s. C_L
Bearman and Obasaju (c) [5]	2.10	1.20
Sohankar et al. (c) [36]	2.25	1.50
Shimada and Ishihara (c) [35]	2.05	1.43
Bao et al. (c) [4]	2.04	1.24
Present work (c)	2.08	1.57
Bao et al. (b) [4]	2.50 ± 0.125	1.7 ± 0.1
Present work (b)	2.57	1.79

The results in Table II are in good agreement with the literature presented in [4] and the references therein. Some differences may however be spotted (for instance in the lift coefficient C_L with Configuration (c)). This may be due to the fineness of the spatial discretization, but also to the time discretization, as pointed out in [40]. In addition to the use of a first-order implicit time integration scheme, the time splitting error may contribute to this inaccuracy.

As in the previous case, the effects of mesh adaptation are investigated on Configuration (b). The adaptive process always starts with an arbitrary uniform mesh and is iterated every 5 time steps. It considers again a multi-criteria adaptivity taking into account the velocity components, its norm and the turbulent viscosity. The number of nodes N is set to 10^5. As shown in Figure 5, the dynamic mesh adaptation capture well the evolution of the solution. Note the concentration of the resolution not only along the boundary layers but also in the wake regions. This reflects well the anisotropy of the solution caused by the nature of the flow past the cylinder. The snapshots in Figure 5 explains also, how for a controlled number of nodes, the mesh is refined around the newly appeared vortices and automatically is coarsened in the regions far from the cylinder.

The zoom-in in Figure 6 highlights how sharply the layers are captured. It shows that the shape but also the size and the orientation of the elements matches the directional features of the flow (boundary layers, flow detachments). The zoom-out in Figure 6 shows how the anisotropic adaptive procedure modifies the mesh in a way that the local mesh resolutions become adequate in all directions. The mean aspect ratio for the anisotropic elements is O(1:1000). Again, the developed stabilized finite element methods show to be very efficient and robust for solving flows at high Reynolds number using highly stretched elements.

The evolution of drag and lift coefficients for different aspect ratios H_2/H_1 is shown in Figure 7. Both reach as expected a steady oscillating state, and compare well to the plots shown in [4].

4.4. NACA0012 airfoil at $Re = 6 \times 10^6$

The objective of this test case is to demonstrate the capability of the method to simulate turbulent flows at Reynolds numbers typical of aeronautical applications, as well as to assess the multi-criteria adaptation feature presented in Section 2. We consider the flow around a NACA0012 airfoil [33], that is a standard benchmark for aerodynamic simulations. The geometry of the NACA airfoil is placed in a domain that is 8 chord lengths wide and 20 chord lengths long, which is considered sufficiently large to avoid any influence of the boundaries on the flow in the vicinity of the airfoil

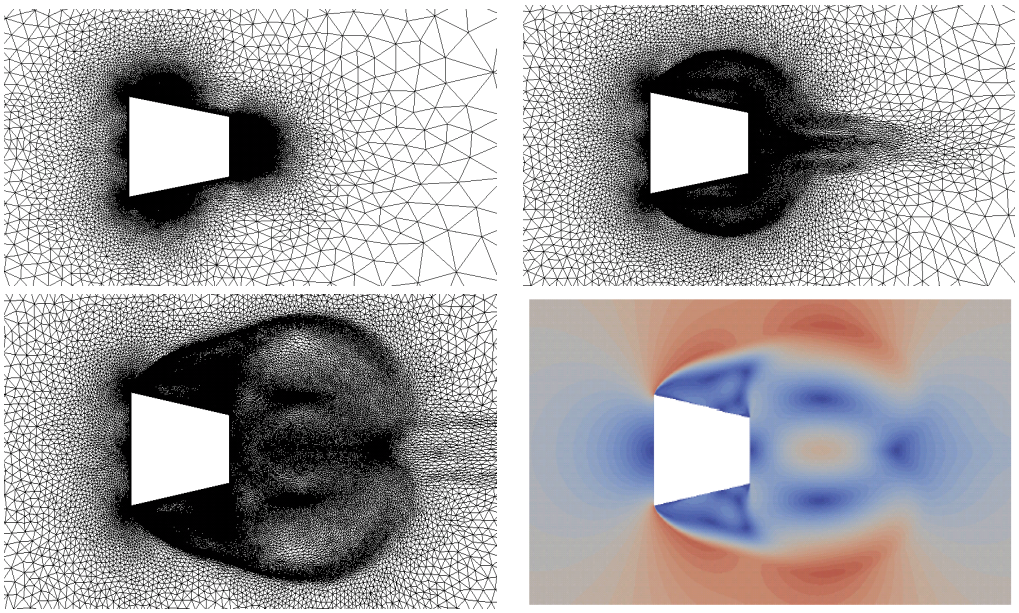


Figure 5. Subsequently adapted meshes and the corresponding velocity norm.

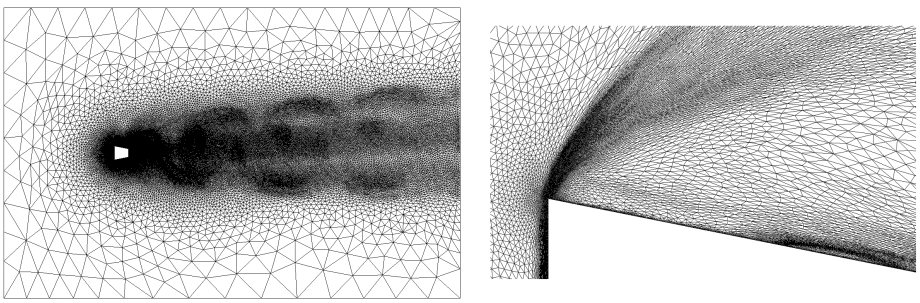


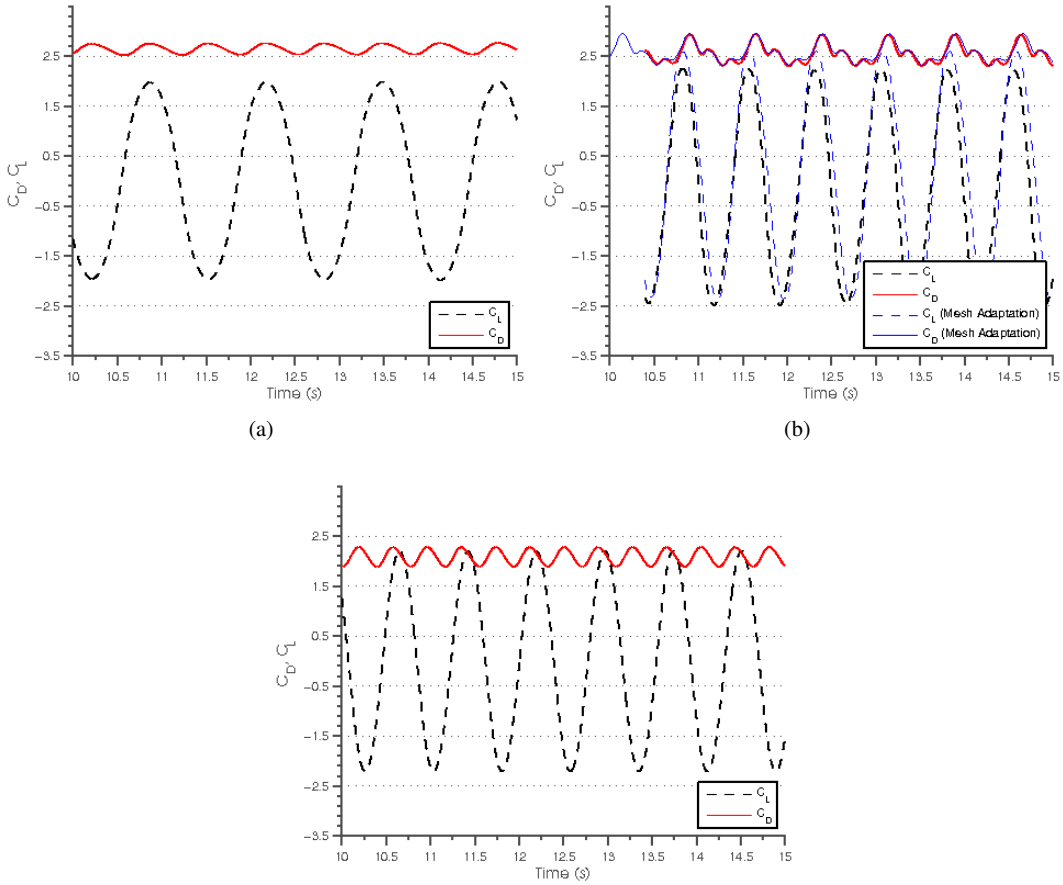
Figure 6. Zoom-out with vortex shedding (*left*) anf zoom-in the adapted mesh (*right*).

(see Figure 8). The Reynolds number based on the unitary cord length is $Re = 6 \times 10^6$ and the Mach number is $Ma = 0.2$, thus ensuring the validity of the incompressibility hypothesis.

Flat velocity and turbulent viscosity ($\tilde{\nu} = 3\nu$) profiles are imposed at the inlet boundary. Slip conditions are prescribed on the boundaries parallel to the flow as $\mathbf{v} \cdot \mathbf{n} = 0$. On the profile, no-slip boundary conditions for the velocity as well as homogeneous Dirichlet conditions for the SA working variable $\tilde{\nu}$ are imposed. Time marching is performed until the steady-state is reached. The non-dimensional time step is set to $\Delta t = 10^{-4}$.

As for the preceding test cases, the effects of the mesh adaptation process described in Section 2 are investigated, in particular with regard to the variables driving the adaptation and the use of a fixed number of elements. Mesh adaptation is triggered every 5 time steps.

In a first phase, two cases are studied with a constant number of $N = 100,000$ nodes in the mesh. In the first one, only the velocity (norm and directions) is included in the adaptation criterion, whereas in the second case, the combination of the velocity and the turbulent viscosity is considered. The meshes obtained are depicted in Figure 9. It is clear that adapting on both the velocity and the turbulent viscosity increases mesh resolution in the wake.



(a) (b) (c)

Figure 7. Drag (lines) and lift (dots) coefficients for aspect ratios (a) H_2/H_1=0.0 ,(b) H_2/H_1=0.6 and (c) H_2/H_1=1.0 .

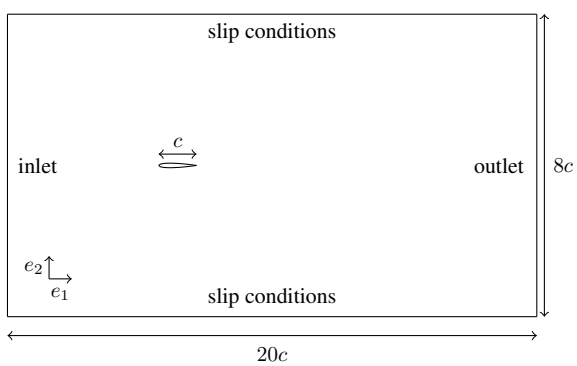


Figure 8. Geometry for the NACA test case in section 4.4.

In a second phase, the adaptation criterion including both velocity and turbulent viscosity is chosen, and three tests are carried out, with a constraint of $N = 50,000$, $N = 100,000$ and $N = 150,000$ nodes respectively. The meshes obtained at the stationary state are shown in Figure 10. We notice that pressure gradients are too localized to drive the adaptation procedure in a way that

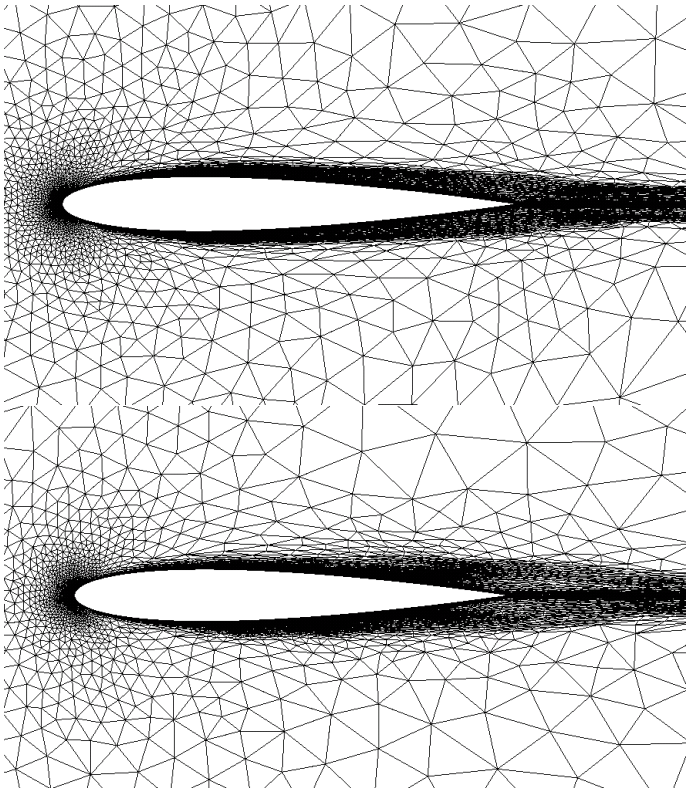


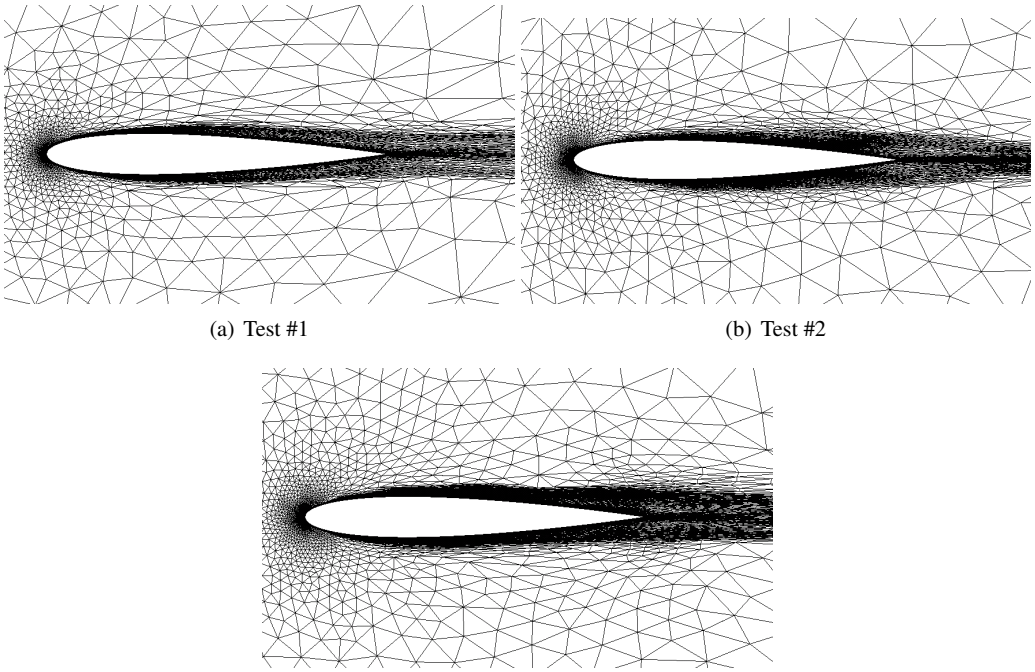
Figure 9. Adaptation driven by the velocity (*left*) and combination of velocity and turbulent viscosity (*right*).

would be beneficial to the resolution of the complete flow. As expected, a high degree of mesh anisotropy is reached in all the cases because of the steep local variations of the velocity and the turbulent viscosity. Note the high resolution in the wake: the element size increases anisotropically when moving away from the trailing edge.

The effect of the mesh resolution and adaptation on the drag coefficient is summarized and compared to results obtained by different codes [33] in Table III. The influence of the mesh size is clearly visible. Under the constraint of a fixed number of elements, the anisotropic mesh adaptation algorithm, based on the principle of equi-distribution of the error, increases the mesh size at the interface to better capture and refine the mesh in the wake. As may be seen from Table III, all tests except the first one are in a 1% range of error, which confirms that it is appropriate to solve the RANS equations by combining stabilized Finite Element methods and mesh adaptation. The influence of the mesh adaptivity criteria is also well highlighted. Indeed, using only the velocity field as a criteria shows higher error for the drag coefficient. Finally, the solutions and meshes obtained in Test 2 are shown in Figure 11 and seem to be in agreement with similar tests in the literature. The mean aspect ratio for the anisotropic elements is O(1:1000).

4.5. Analysis of turbulent flow past a drone

In the last numerical test case, the aim is to assess the capability of our numerical method to simulate the turbulent flow past a complex 3D geometry, namely a UAV. We seek to reproduce experimental results obtained for several angles of attack (0, 5, 10, 15) with a set-up pictured in Figure 12. The



(a) Test #1

(b) Test #2

(c) Test #3

Figure 10. The obtained meshes using (a)$50,000$, (b)$100,000$ and (c) $150,000$ elements.

Table III. Summary of test cases.

	Reference	Test #1	Test #2	Test #3	Test #4
No. elements $\times 10^5$		0.5	1.0	1.5	1.5
Adaptation using		$\underline{\mathbf{v}}$ & ν_t	$\underline{\mathbf{v}}$ & ν_t	$\underline{\mathbf{v}}$ & ν_t	$\underline{\mathbf{v}}$
$C_D \times 10^{-3}$	8.155 ± 0.35	7.90	8.12	8.16	7.7
Relative error		3%	0.3%	0.1%	4.5%

model has a span of 300 mm and a total length of 300 mm. The inlet velocity is set to 10 m/s, yielding a span-based Reynolds number of $Re = 2 \times 10^6$.

The model is placed in a computational domain of length $3m$ and width $1m$. Similarly to the preceding test cases, flat velocity and turbulent viscosity ($\tilde{\nu} = 3\nu$) profiles are prescribed at the inlet boundary, a pressure condition is imposed at the outlet, and slip conditions are used on the boundaries parallel to the flow. At the UAV wall, no-slip boundary conditions for the velocity as well as homogeneous Dirichlet conditions for the turbulent viscosity $\tilde{\nu}$ are specified. Again, time marching is performed until the steady-state is reached. The non-dimensional time step is set to $\Delta t = 10^{-4}$. Figure 13 gives a general view of the computational domain.

All the 3D simulations have been conducted in parallel on 64, 2.4Ghz Opteron cores and have required 12.5 hours of computational time. On average, 4 nonlinear iterations were required to reach convergence.

In the initial configuration, the UAV is simulated at zero angle of attack. The mesh is adapted every 10 time steps while marching in time. When steady state is reached, the configuration is instantaneously rotated to the following prescribed angle of attack. Figure 14 shows the steady-state mesh for zero angle of attack in the vicinity of the UAV.

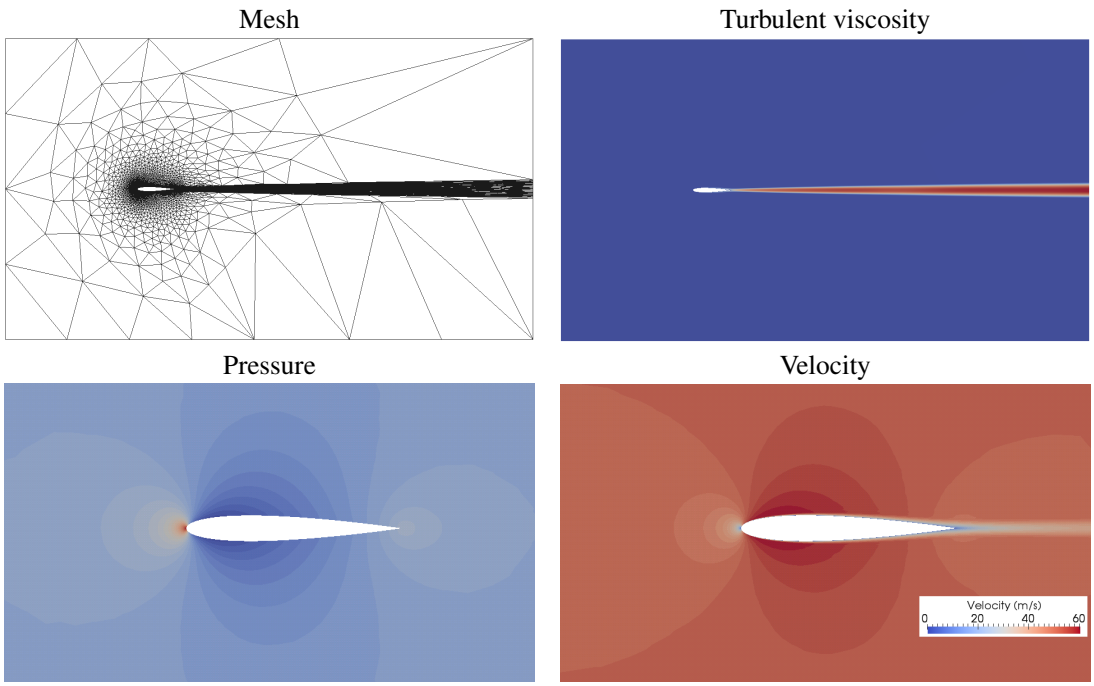


Figure 11. Final plots for Test # 2.

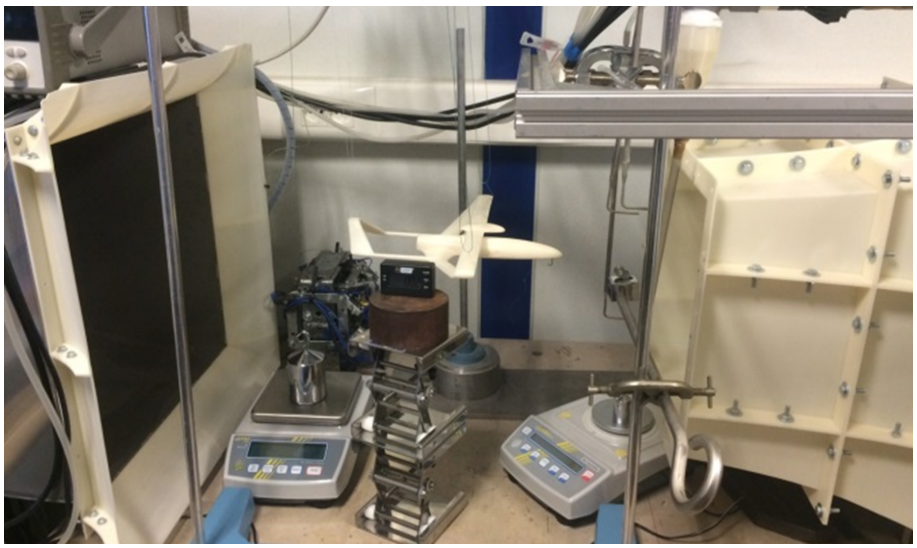


Figure 12. The experimental setup: the drone inside a wind tunnel

Results obtained for the drag are presented in Figure 15. The agreement between the numerical and the experimental results is satisfying despite the discrepancies between the experimental setup and the numerical simulations. Note that three experiments were made for each case and the averaged value is considered here. In particular, the surface roughness of the model obtained by additive manufacturing may be the cause of the higher drag, due to the increased turbulent skin friction. Figure 16 shows a visualization of the flow past the UAV by means of streamlines and pressure distribution.

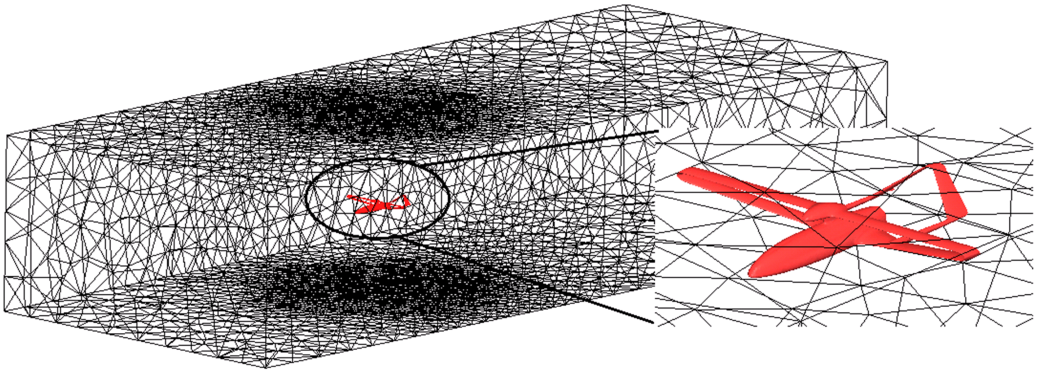


Figure 13. The immersed drone inside the numerical wind channel

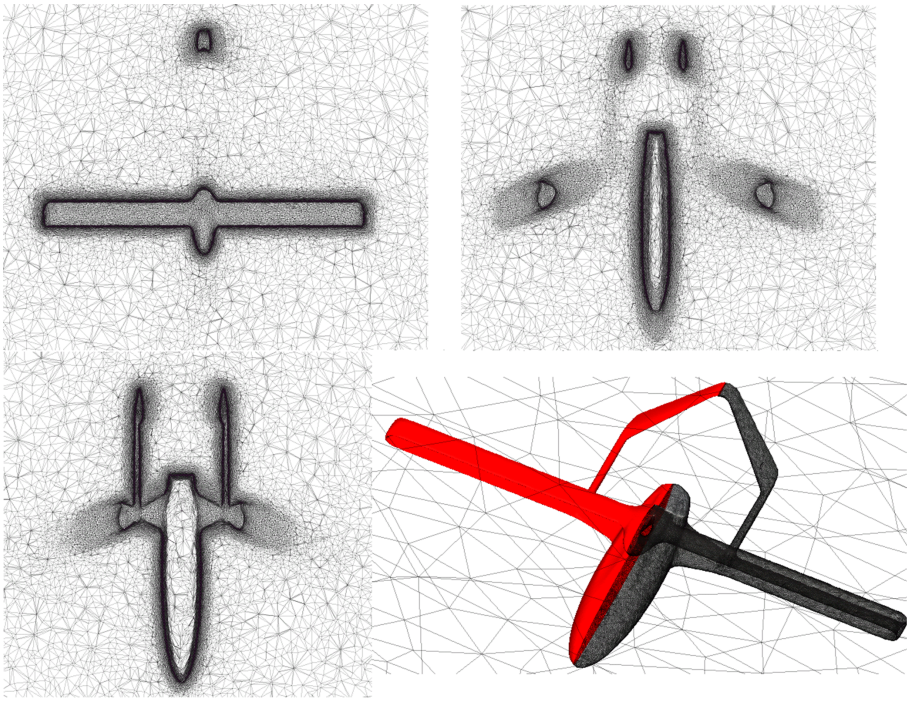


Figure 14. Zoom on the adapted mesh around the immersed drone

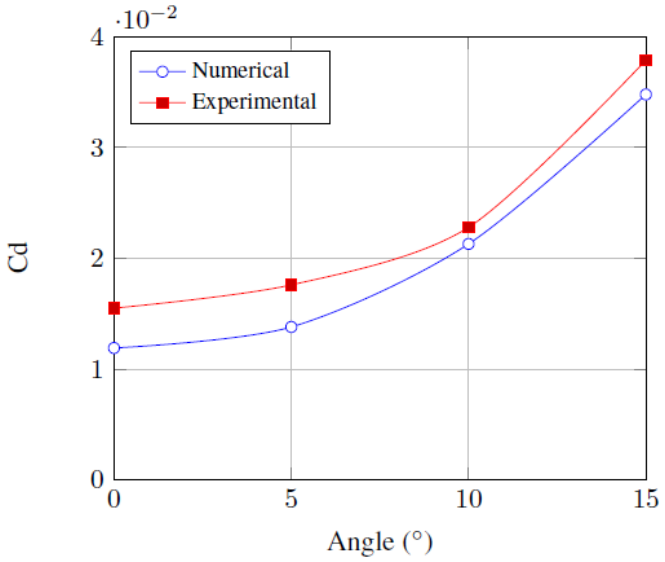


Figure 15. Comparisons for the drag of the UAV at 10m/s

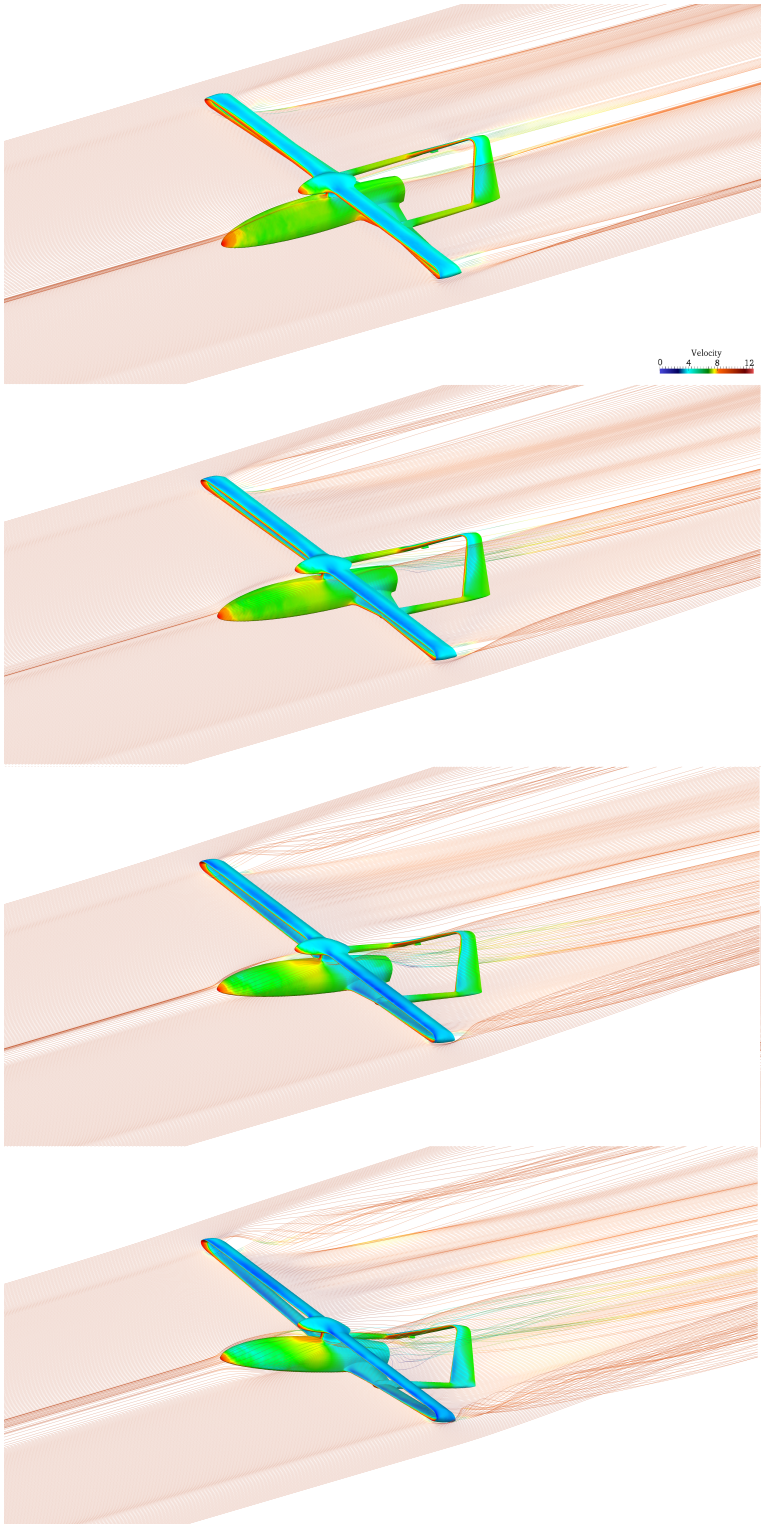


Figure 16. Streamlines past the UAV from $0\,\mathrm{deg}$ (top) to $15\,\mathrm{deg}$ angle of attack (bottom)

5. CONCLUSION

An adaptive anisotropic mesh methodology for performing accurate numerical simulations of turbulent flows past complex geometries was presented. It couples a stabilized Variational Multiscale Navier-Stokes solver, a Spalart-Allmaras turbulent model with a dynamic anisotropic mesh adaptation algorithm. The benefit from using an edge-based error estimator with a multi-criteria functional that estimates the error on a combination of several variables including the velocity norm, all the velocity components as well as the turbulent viscosity is shown on different challenging cases in 2 and 3 dimensions. All the main directional features, characterizing the turbulent flows past immersed solids, are well detected and captured. The comparison with the literature show that despite a low number of nodes, simulations do not suffer any instabilities and a good accuracy is obtained. Further investigations will take into account the performance of the presented adaptive method in terms of the computational cost, the speed-up and the frequency of remeshing in a massively parallel environment.

REFERENCES

1. F. Alauzet. *Adaptation de maillage anisotrope en trois dimension. Applications aux simulations instationnaires en Mcanique des Fluides*. PhD thesis, Universit de Montpellier II, 2003.
2. S. R. Allmaras and F. T. Johnson. Modifications and clarifications for the implementation of the spalart-allmaras turbulence model. In *7th International Conference on Computational Fluid Dynamics, Big Island, Hawaii*, 9-13 July 2012.
3. S. Badia and R. Codina. Analysis of a stabilized finite element approximation of the transient convection-diffusion equation using an ale framework. *Journal on Numerical Analysis*, 44:2159–2197, 2006.
4. Yan Bao, Dai Zhou, Cheng Huang, Qier Wu, and Xiang-Qiao Chen. Numerical prediction of aerodynamic characteristics of prismatic cylinder by finite element method with spalart–allmaras turbulence model. *Computers & structures*, 89(3):325–338, 2011.
5. PW Bearman and ED Obasaju. An experimental study of pressure fluctuations on fixed and oscillating square-section cylinders. *Journal of Fluid Mechanics*, 119:297–321, 1982.
6. A. Belme, A. Dervieux, and F. Alauzet. Time accurate anisotropic goal-oriented mesh adaptation for unsteady flows. *Journal of Computational Physics*, 231(19):6323 – 6348, 2012.
7. A. Cangiani and E. Süli. The residual-free-bubble finite element method on anisotropic partitions. *SIAM Journal on Numerical Analysis*, 45(4):1654–1678, 2007.
8. Diaz Castro, F Hecht, and B Mohammadi. New progress in anisotropic grid adaptation for inviscid and viscous flows simulations. In *In 4th Annual Intl. Meshing Roundtable*. Citeseer, 1995.
9. R. Codina. Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 190:1579–1599, 2000.
10. T. Coupez. Metric construction by length distribution tensor and edge based error for anisotropic adaptive meshing. *Journal of Computational Physics*, 230:2391–2405, 2011.
11. A. Ern and J.L. Guermond. *Theory and Practice of Finite Elements*. Applied Mathematical Sciences. Springer New York, 2004.
12. L. Formaggia, E. Miglio, A. Mola, and N. Parolini. Fluidstructure interaction problems in free surface flows: Application to boat dynamics. *International Journal for Numerical Methods in Fluids*, 56(8):965–978, 2008.
13. L. Formaggia and S. Perotto. New anisotropic a priori error estimates. *Numerische Mathematik*, 89(4):641–667, 2001.
14. L. Formaggia and S. Perotto. Anisotropic error estimates for elliptic problems. *Numerische Mathematik*, 94(1):67–92, 2003.
15. C. Förster, W. A. Wall, and E. Ramm. Stabilized finite element formulation for incompressible flow on distorted meshes. *International Journal for Numerical Methods in Fluids*, 60(10):1103–1126, 2009.
16. PASCAL-JEAN Frey and Frédéric Alauzet. Anisotropic mesh adaptation for cfd computations. *Comput. Meth. Appl. Mech. Engrg.*, 194(48):5068–5082, 2005.

17. E. Hachem, B. Rivaux, T. Kloczko, H. Digonnet, and T. Coupez. Stabilized finite element method for incompressible flows with high reynolds number. *Journal of Computational Physics*, 229(23):8643–8665, 2010.
18. I. Harari and T. J. R. Hughes. What are c and h?: inequalities for the analysis and design of finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 97:157–192, 1992.
19. Michael J. Hemsch. Statistical Analysis of Computational Fluid Dynamics Solutions from the Drag Prediction Workshop. *Journal of Aircraft*, 41(1):95–103, 2004.
20. T. R. J. Hughes. Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origin of stabilized methods. *Computer Methods in Applied Mechanics and Engineering*, 127:387–401, 1995.
21. T.J.R. Hughes, J.A. Cottrell, and Y. Bazilevs. Isogeometric analysis: CAD, finite elements, NURBS, exact geometry and mesh refinement. *Computer Methods in Applied Mechanics and Engineering*, 194(39-41):4135–4195, October 2005.
22. A. Jameson. A perspective on computational algorithms for aerodynamic analysis and design. *Progress in Aerospace Sciences*, 37(2):197–243, 2001.
23. G. Jannoun, E. Hachem, J. Veysset, and T. Coupez. Anisotropic meshing with time-stepping control for unsteady convection-dominated problems. *Applied Mathematical Modelling*, 39(7):1899 – 1916, 2015.
24. R. A Khurram, Y. Zhang, and W. G. Habashi. Multiscale finite element method applied to the spalart–allmaras turbulence model for 3d detached-eddy simulation. *Computer Methods in Applied Mechanics and Engineering*, 233:180–193, 2012.
25. Gerd Kunert and R Verfürth. Edge residuals dominate a posteriori error estimates for linear finite element methods on anisotropic triangular and tetrahedral meshes. *Numer. Math.*, 86(2):283–303, 2000.
26. Erb F. Lins, Renato N. Elias, Gabriel M. Guerra, Fernando A. Rochinha, and Alvaro L. G. A. Coutinho. Edge-based finite element implementation of the residual-based variational multiscale method. *International Journal for Numerical Methods in Fluids*, 61(1):1–22, 2009.
27. Y. Mesri, M. Khalloufi, and E. Hachem. On optimal simplicial 3d meshes for minimizing the hessian-based errors. *Applied Numerical Mathematics*, pages –, 2016.
28. S. Micheletti, S. Perotto, and M. Picasso. Stabilized finite elements on anisotropic meshes: A priori error estimates for the advection-diffusion and the stokes problems. *SIAM Journal on Numerical Analysis*, 41:1131–1162, 2004.
29. S. Mittal. On the performance of high aspect ratio elements for incompressible flows. *Computer Methods in Applied Mechanics and Engineering*, 188:269–287, 2000.
30. Nicola Parolini and Alfio Quarteroni. Mathematical models and numerical simulations for the americas cup. *Computer Methods in Applied Mechanics and Engineering*, 194(911):1001 – 1026, 2005.
31. R. Rannacher. *Finite element methods for the incompressible Navier-Stokes equations*. Springer, 2000.
32. Jean-François Remacle, Xiangrong Li, S Mark Shephard, and E Joseph Flaherty. Anisotropic adaptive simulation of transient flows using discontinuous Galerkin methods. *Int. J. Num. Meth. Engrg.*, 62(7):899–923, 2005.
33. C Rumsey. Nasa turbulence modeling resource page. http://turbmodels.larc.nasa.gov/.
34. M. Schäfer, S. Turek, F. Durst, E. Krause, and R. Rannacher. Benchmark computations of laminar flow around a cylinder. *Notes on numerical fluid mechanics*, 52:547–566, 1996.
35. K. Shimada and T. Ishihara. Application of a modified $k-\epsilon$ model to the prediction of aerodynamic characteristics of rectangular cross-section cylinders. *Journal of Fluids and Structures*, 16(4):465 – 485, 2002.
36. Ahmed Sohankar, Lars Davidson, and Christoffer Norberg. Large eddy simulation of flow past a square cylinder: comparison of different subgrid scale models. *Journal of Fluids Engineering*, 122(1):39–47, 2000.
37. P. Spalart and S. Allmaras. A one-equation turbulence model for aerodynamic flows. *AIAA journal*, 1992.
38. Philippe R Spalart and Christopher L Rumsey. Effective inflow conditions for turbulence models in aerodynamic calculations. *AIAA journal*, 45(10):2544–2553, 2007.
39. T. E. Tezduyar and Y. Osawa. Finite element stabilization parameters computed from element matrices and vectors. *Computer Methods in Applied Mechanics and Engineering*, 190(3-4):411–430, 2000.
40. J. Volker. Reference values for drag and lift of a two-dimensional time-dependent flow around a cylinder. *International Journal for Numerical Methods in Fluids*, 44(7):777–788, 2004.
41. John Volker. Reference values for drag and lift of a two-dimensional time-dependent flow around a cylinder. *International Journal for Numerical Methods in Fluids*, 44(7):777–788, 2004.