Adaptive Stopping Criterion of Iterative Solvers for Efficient Computational Cost Reduction: Application to Navier–Stokes with Thermal Coupling

Aurélien Larcher, Ghaniyya Medghoul, Gabriel Manzinali, Elie Hachem

Adaptive stopping criterion of iterative solvers for efficient computational cost reduction: application to Navier–Stokes with thermal coupling

Ghaniyya Medghoul[a], Gabriel Manzinali[a], Elie Hachem[a], Aurélien Larcher[a,*]

[a] *Mines Paris, PSL - Research University, CEMEF - Centre for Material Forming, CNRS UMR 7635, CS 10207 rue Claude Daunesse, 06904 Sophia-Antipolis Cedex, France.*

Abstract

In this article, a strategy for efficient computational cost reduction of numerical simulations for complex industrial applications is developed and evaluated on multiphysics problems. The approach is based on the adaptive stopping criterion for iterative linear solvers previously implemented for elliptic partial differential equations and the convection–diffusion equation. Control of the convergence of iterative linear solvers is inferred from *a posteriori* error estimators used for anisotropic mesh adaptation. Provided that the computed error indicator provides an equivalent control on the discretization error, it is a suitable ingredient to assess when enough accuracy has been reached so that iterations of algebraic solvers can be stopped. In practice the iterative solution is stopped when the algebraic error is lower than a percentage of the estimated discretization error. The proposed method proves to be an effective cost-free strategy to reduce the number of iterations needed without degrading the accuracy of the solution. The discretization in the current work is based on stabilized finite elements, while the Generalized Minimal Residual method (GMRES) is used as iterative linear solver. Numerical experiments are performed of increasing complexity, from manufactured solutions to industrial configurations to evaluate the efficiency and the strengths of the proposed adaptive method.

Keywords: CFD, Adaptive stopping criterion, Anisotropic error estimator, Mesh Adaptation, Adaptive unstructured meshes, Navier–Stokes

1. Introduction

Numerical simulation of multiphysics systems encountered in engineering applications has become an essential tool for the control and optimization of industrial processes. However given the wide range of temporal and spatial scales involved in the different physical phenomena at hand, the computational cost can become prohibitive. This is especially true for the industrial applications in the frame of which this work as been carried out: hardening treatments of alloys using furnace-based thermal processes and quenching, which require capturing very small spatial scales over long periods of time. In both cases a steep initial unsteady phase driven by conjugate heat transfer with sharp gradients of temperature at the interface between manufactured parts (solids) and the fluid environment (furnace atmosphere, quenching bath) is then superseded by turbulent forced convection. The model equations in play consist therefore of the heat equation in solid parts, and the Navier–Stokes equations coupled with the thermal equation in the fluid environment. Additional models may be incorporated to take into account phase change in the fluid environment such as vaporization/boiling in the quenching bath, or phase transformation in the solid parts, but the current presentation is restricted to the coupling between incompressible fluid dynamics and heat transfer. Adaptive methods in terms of refinement in space and time of the discretization [1] are well known to be a reliable approaches to dynamically achieve better accuracy in the solution of PDEs, with a reduced computational cost. Anisotropic mesh adaptation specifically tackles the challenge of capturing directional features that characterize several physical problems, such as boundary of inner layers or heterogeneous material properties. These methods rely on *a posteriori* error estimates to identify

*Corresponding author.
Email address: aurelien.larcher@minesparis.psl.eu (Aurélien Larcher)

the regions of interest to be adapted so as to provide optimal local orientation and stretching of the elements. The adapted mesh is obtained applying local modifications of the initial mesh topology and geometry based on a discrete Riemannian metric field which is constructed from *a posteriori* error analysis. This subject has been addressed by several authors [2, 3, 4, 5, 6]. In [2] for example, the error analysis is done on the edges of the elements, while another possible choice is to rely on the recovered Hessian of the solution as a base to build the estimator [3, 4, 5]; the Hessian can be used also in combination with a PDE-dependent estimator to improve accuracy, as suggested in [6]. Using an Hessian-based metric field one can take advantage of the robustness and generality of the computational framework, and the relatively simple implementation.

The motivation of the present work is to exploit this general error estimation framework to extend its application to the control of the iterative algebraic solvers. Error analysis of solutions from the discretization of a continuous problem usually focuses on the approximation error which follows from the choice of time marching scheme and of discrete function spaces. In real-world applications however, the resulting system of linear equations can be solved efficiently only with an iterative procedure which introduces another level of error coined algebraic error. The accuracy of the approximation is therefore controlled by the stopping criteria used to drive the convergence of the iterative procedure.

As remarked by Becker et al. in their seminal work [7], *ad hoc* stopping criteria are commonly used; for example in their simplest form by requiring an initial residual norm to be reduced by a certain factor. These criteria are straightforward to implement but have no direct link to the actual error in the approximate solution. This could possibly affect the efficiency of the iterative procedure and the accuracy of the resulting solution. On the one side an highly accurate approximation is inefficient and most likely unnecessary, on the other side a poor approximation affects the accuracy of the solution and the convergence of the adaptation procedure. Several approaches regarding inexact iterative solvers and stopping criteria have been developed in an attempt to estimate the algebraic error in relation to the discretization error. In the framework of symmetric problems, Arioli [8] proposes an *a priori* stopping criterion and Picasso [9] suggests an *a posteriori* approach, as well as other methods presented in [10, 11, 12]. An interesting overview of the existing approaches is proposed by Arioli and co-workers [13].

In this work an adaptive stopping criterion is proposed that follows the strategy developed for convection–diffusion problems [14], extending the application to Navier–Stokes problems. It provides a cost-efficient automated adaptive control for the iterative solver, exploiting the information from the error analysis used by the mesh adaptation procedure. The method proves to be effective to drastically reduce the number of iterations needed, without spoiling at all the accuracy of the solution. The present work relies on discretization based on stabilized finite element schemes such the Variational MultiScale method (VMS). The implementation of the VMS approach is that described in [15, 16], which consists in the splitting of the solution into a resolved part (coarse scales) and an unresolved part (subscales) which is modeled to enrich the function spaces.

This paper is organized as follows. In Section 2, the governing equations are introduced as well as their discretization by the VMS method. In Section 3, the anisotropic mesh adaptation procedure is described, namely how to construct a discrete metric defined on the vertices of the mesh, based on interpolation operators using the star topology. In Section 4, the *a posteriori* error estimator framework and its relation to the construction of the metric used for the mesh adaptation is detailed. In Section 5, the adaptive stopping criterion to control the iterative solver is presented then followed in Section 6 by further discussions about the impact of the choice of preconditioner. Finally in Section 7, the performance of the adaptive strategy is evaluated on verification tests by means of manufactured solutions then on pre-industrial test cases which are representative of realistic multiphysics systems.

2. Governing equations

Let us introduce here the governing equations, namely the unsteady incompressible Navier–Stokes equations and their discretization by means of stabilized linear finite elements. A Variational MultiScale approach is used to derive the stabilized finite element scheme for the Navier–Stokes equations. Both the velocity and the pressure spaces are enriched, addressing the problems of spurious oscillations in the convection-dominated regime and the pressure instability induced by lack of discrete space compatibility.

2.1. The Navier–Stokes equations

Let $\Omega \subset \mathbb{R}^d$ be the fluid domain and $\partial\Omega$ its boundary, and $(0,T)$ is the time interval. The incompressible Navier–Stokes equations in strong form read

$$\begin{cases} \rho\ (\partial_t \mathbf{u} + (\mathbf{u}\cdot\nabla)\mathbf{u}) - \nabla\cdot\boldsymbol{\sigma} = \mathbf{f}, \\ \nabla\cdot\mathbf{u} = 0, \end{cases} \tag{1}$$

where ρ denotes the density, $\mathbf{u}$ the velocity field, p the pressure, $\mathbf{f}$ may be a given forcing term, and the Cauchy stress tensor for a Newtonian fluid is given by

$$\boldsymbol{\sigma} = 2\mu\ \boldsymbol{\varepsilon}(\mathbf{u}) - p\,\mathbf{I}_d, \tag{2}$$

with μ the dynamic viscosity, $\boldsymbol{\varepsilon}(\mathbf{u}) = 1/2(\nabla\mathbf{u} + \nabla\mathbf{u}^T)$ the rate-of-strain tensor, and $\mathbf{I}_d$ the d-dimensional identity tensor. In order to close the problem, Equations (1) are supplemented with boundary and initial conditions to be specified later.

The weak form of Problem (1) combined with Relation (2) can be obtained by multiplication by a test function and integration by parts to seek solutions in the sense of distributions. Let $H^1(\Omega)$ be the Sobolev space of square integrable functions the distributional derivatives of which are square integrable on Ω, that is the Lebesgue space

$$L^2(\Omega) = \left\{ u : \int_\Omega |u(x)|^2 \mathrm{d}\mathbf{x} < +\infty \right\} \tag{3}$$

endowed with its natural scalar product

$$(u,v)_{L^2} = \int_\Omega uv\mathrm{d}\mathbf{x}, \tag{4}$$

and let $V \subset \left[H^1(\Omega)\right]^d$ be a function space properly chosen according to the boundary conditions. Finally, let $Q = \left\{ q \in L^2(\Omega) : \int_\Omega q = 0 \right\}$. Let us define $(\cdot,\cdot)$ the scalar product of the space $L^2(\Omega)$ and assume homogeneous Dirichlet boundary conditions. The weak form of Problem (1) follows

$$\begin{cases} \text{Find } (\mathbf{u},p) \in V\times Q \text{ such that:} \\ \rho\ [(\partial_t\mathbf{u},\mathbf{v}) + ((\mathbf{u}\cdot\nabla)\mathbf{u},\mathbf{v})] + (2\mu\boldsymbol{\varepsilon}(\mathbf{u}),\boldsymbol{\varepsilon}(\mathbf{v})) - (p,\nabla\cdot\mathbf{v}) = (\mathbf{f},\mathbf{v}), \quad \forall \mathbf{v}\in V \\ (\nabla\cdot\mathbf{u},q) = 0, \quad \forall q \in Q. \end{cases} \tag{5}$$

Given an admissible triangulation $\mathscr{T}_h$ of Ω, the function 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 fulfill a compatibility condition [17]. 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-dominated problems (*i.e.* problems where the convection term $(\mathbf{u}\cdot\nabla)\mathbf{u}$ is much larger than the diffusion term $\nabla\cdot(2\mu\boldsymbol{\varepsilon}(\mathbf{u}))$) also lead to a loss of coercivity in Formulation (5). This phenomenon manifests itself as oscillations that pollute the solution. In this work, a Variational MultiScale method [18] is used to circumvent 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, fine scales are resolved in an approximate manner and then their effect is modelled into the large-scale equation. For the sake of completeness only an outline of the method is presented here.

Let us split the velocity and the pressure fields into resolvable coarse-scale and unresolved fine-scale components: $\mathbf{u} = \mathbf{u}_h + \mathbf{u}'$ and $p = p_h + p'$. The same decomposition can be applied to the weighting functions: $\mathbf{v} = \mathbf{v}_h + \mathbf{v}'$ 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 function 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:

$$\begin{cases} \text{Find}(\mathbf{u},p) \in V\times Q \text{ such that:} \\ \rho\,(\partial_t(\mathbf{u}_h+\mathbf{u}'),(\mathbf{v}_h+\mathbf{v}')) + \rho\,(((\mathbf{u}_h+\mathbf{u}')\cdot\nabla)(\mathbf{u}_h+\mathbf{u}'),(\mathbf{v}_h+\mathbf{v}')) \\ \qquad\qquad + (2\mu\boldsymbol{\varepsilon}(\mathbf{u}_h+\mathbf{u}'),\boldsymbol{\varepsilon}(\mathbf{v}_h+\mathbf{v}')) \\ \qquad\qquad - ((p_h+p'),\nabla\cdot(\mathbf{v}_h+\mathbf{v}')) = (\mathbf{f},(\mathbf{v}_h+\mathbf{v}')), \quad \forall \mathbf{v}\in V_0 \\ (\nabla\cdot(\mathbf{u}_h+\mathbf{u}'),(q_h+q')) = 0, \quad \forall q\in Q. \end{cases} \tag{6}$$

To derive the stabilized formulation, Equations (6) are split 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 [15], the fine-scale solutions $\mathbf{u}'$ and p' written in terms of the time-dependent large-scale variables using residual-based terms that are derived consistently. Consequently, static condensation can be used, that consists in substituting directly $\mathbf{u}'$ 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:

$$\begin{cases} \rho\left(\partial_t \mathbf{u}_h, \mathbf{v}_h\right) + \left(\rho(\mathbf{u}_h \cdot \nabla)\mathbf{u}_h, \mathbf{v}_h\right) \\ \qquad - \left(\tau_1 \mathscr{R}_M, \rho(\mathbf{u}_h \cdot \nabla)\mathbf{v}_h\right) + \left(2\mu\varepsilon(\mathbf{u}_h), \varepsilon(\mathbf{v}_h)\right) \\ \qquad - \left(p_h, \nabla \cdot \mathbf{v}_h\right) - \left(\tau_2 \mathscr{R}_C, \nabla \cdot \mathbf{v}_h\right) = \left(\mathbf{f}, \mathbf{v}_h\right), \quad \forall \mathbf{v}_h \in V_{h,0} \\ \left(\nabla \cdot \mathbf{u}_h, q_h\right) - \left(\tau_1 \mathscr{R}_M, \nabla q_h\right) = 0, \quad \forall q_h \in Q_h \end{cases} \tag{7}$$

where $\mathscr{R}_M$ and $\mathscr{R}_M$ denote respectively the residuals of the momentum equation and of the continuity equation,

$$\begin{aligned} \mathscr{R}_M &= \mathbf{f} - \rho\partial_t \mathbf{u}_h - \rho(\mathbf{u}_h \cdot \nabla)\mathbf{u}_h - \nabla p_h \\ \mathscr{R}_C &= -\nabla \cdot \mathbf{u}_h \end{aligned} \tag{8}$$

and the quantities τ_1 and τ_2 are element-wise stabilization parameters defined hereafter.

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 the following definition proposed in [19] is adopted, for any $K \in \mathscr{T}_h$,

$$\tau_1|_K = \left[\left(\frac{2\rho\|\mathbf{u}_h\|_K}{h_K} \right)^2 + \left(\frac{4\mu}{h_K^2} \right)^2 \right]^{-\frac{1}{2}}, \tag{9}$$

$$\tau_2|_K = \left[\left(\frac{\mu}{\rho} \right)^2 + \left(\frac{c_2\|\mathbf{u}_h\|_K}{c_1 h_K} \right)^2 \right]^{\frac{1}{2}}, \tag{10}$$

where h_K is the characteristic length of the element and c_1 and c_2 are algorithmic constants; the values $c_1 = 4$ and $c_2 = 2$ are chosen for linear elements [19].

2.2. Element size measures in stabilization parameters

Recall that the coefficients τ_1 and τ_2 act as weights to the stabilization terms added to the weak formulation (7). 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 [20], while others always use the triangle diameter (see [21] 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 [22, 19]. In [23], the authors examine in detail the effect of different element length definitions on distorted meshes. In [24], 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, the definition proposed in [25] is implemented to compute h_K as the size of K in the direction of the velocity using the finite element advection operator as

$$h_K = 2\|\mathbf{u}_h\|_K \left(\sum_{\varphi_i \in \mathrm{span}(V_h(K))} |(\mathbf{u}_h \cdot \nabla\varphi_i)(\mathbf{x}_K)| \right)^{-1} \tag{11}$$

where $\varphi_1, \ldots, \varphi_{N_K}$ are the basis functions of $P_1(K)$ mapped onto K and $\mathbf{x}_K$ is the centroid of element K.

2.3. Semi-discrete scheme

The Navier–Stokes equations are discretized in time by a semi-implicit scheme using either backward Euler or Crank–Nicolson. The viscous term and the pressure term in the momentum equation, as well the divergence term in the continuity equation, are integrated implicitly, while the stabilization terms are evaluated semi-implicitly and the source term is explicit. In the numerical framework two ways of treating the convective term implicitly are implemented: the first one is the usual Picard iteration where at the k-th inner iteration the advection velocity is taken at the previous inner iteration so that $[(\mathbf{u}\cdot\nabla)\mathbf{u}]^k \approx (\mathbf{u}^{k-1}\cdot\nabla)\mathbf{u}^k$, and the second is a classic Newton method with the linearization $[\mathbf{u}\cdot\nabla)\mathbf{u}]^k \approx (\mathbf{u}^k\cdot\nabla)\mathbf{u}^{k-1} + (\mathbf{u}^{k-1}\cdot\nabla)\mathbf{u}^k - (\mathbf{u}^{k-1}\cdot\nabla)\mathbf{u}^{k-1}$. In the present work the convective term was treated semi-implicitly (*i.e.* doing only one Picard iteration) to avoid any influence of the choice of the stopping criterion of nonlinear iterations on the performance metrics.

3. General metric

First of all, let us introduce the definition of the discrete Riemannian metric field and some useful notation. Let us consider a triangulation $\mathscr{T}_h$ of topological dimension d, in a Euclidean space of the same dimension. Given a non-degenerate element $\hat{K}$, the discrete metric field M_K can be defined such that $\hat{K}$ is a unit element in the Riemannian space associated with M_K. Indeed, applying an affine transformation T_K from the physical space with associated finite element (K,P,Σ), to the unit ball of the Euclidean metric space, mapping element K to a reference unit element $\hat{K}$, we can determine M as follows

$$\begin{array}{rccc} T_K & : & \hat{K} & \to & K \\ & & \hat{\mathbf{x}} & \mapsto & \mathbf{x} = \mathbf{x}_0 + A_K(\hat{\mathbf{x}} - \hat{\mathbf{x}}_0) \end{array} \tag{12}$$

where $\hat{\mathbf{x}}_0$ and $\mathbf{x}_0$ denotes the origin of the associated local coordinate system and A_K is the Jacobian matrix of the affine mapping. Unlike the usual finite element definition, the reference element $\hat{K}$ in this case is the most balanced simplex, such that all edges are of unit length: for instance in two dimensions of space it is the unitary equilateral triangle. The mapping T_K is sometimes coined the equiaffine simplex mapping. To perform the mesh adaptation procedure, the inverse operation is required to pull back from the physical space to the reference element. Therefore, T_K^{-1} the inverse transform of T_K reads

$$\begin{array}{rccc} T_K^{-1} & : & K & \to & \hat{K} \\ & & \xi & \mapsto & \hat{\xi} = A_K^{-1}\xi \end{array}$$

where ξ and $\hat{\xi}$ denote respectively vectors in the mapped element K and in the reference element $\hat{K}$. Let us denote by $\mathscr{E}(K) = \{\xi_{ij} = \mathbf{x}_j - \mathbf{x}_i : i,j = 0,\dots,d, i \neq j\}$ the set of edge vectors of the d-simplex K. Therefore the set edge vectors $\{\hat{\xi}\}_{ij}$ of the unitary equilateral simplex $\hat{K}$, satisfies therefore the relation

$$|\hat{\xi}_{ij}| = |A_K^{-1}\xi_{ij}| = 1,$$

and equivalently

$$(A_K^{-1}\xi_{ij}, A_K^{-1}\xi_{ij}) = (A_K^{-T}A_K^{-1}\xi_{ij}, \xi_{ij}) = 1,$$

so that the M-conjugate scalar product can be introduced formally,

$$\forall K \in \mathscr{T}_h, \quad (M_K\xi_{ij}, \xi_{ij}) = 1.$$

Subsequently, the metric tensor on the element K can be identified as

$$M_K = A_K^{-T}A_K^{-1} \tag{13}$$

so that M_K is a constant tensor per element if the mesh is affine, and is uniquely defined on the element K. Repeating the same construction process for all the elements in the mesh, a piecewise constant tensor field is obtained.

3.1. Vertex-based discrete metric field

In the following, the discrete metric is now based on the mesh vertices so that M^i is now constructed for all $i \in \mathscr{V}(\mathscr{T}_h)$, the set of vertices of the triangulation, using the star topology. Let $\Gamma(i) = \left\{ j \in \mathscr{V}(\mathscr{T}_h), \exists K \in \mathscr{T}_h, \xi_{ij} \in \mathscr{E}(K) \right\}$ be the set of edges in the mesh connected to vertex i. The aim is to define a metric on vertex i such that the length of each edge connected to vertex i equals one with respect to the metric field M^i. An averaging process is used to construct a unique tensor at $\mathbf{X}^i$ that gathers all the data defined on the edges sharing this node. Hence, the discrete metric field M^i is sought as defined on vertex i such that the lengths of all edges in $\Gamma(i)$ are one in the norm induced by the M-conjugate scalar product. Taking $M^i = A_{ij}^{-T} A_{ij}^{-1}$, the M-conjugate product reads

$$(M^i \xi_{ij}, \xi_{ij}) = 1, \quad \forall j \in \Gamma(i) \tag{14}$$

which after summing up over j so that all incident edges contributions are taken into account,

$$\sum_{j \in \Gamma(i)} (M^i \xi_{ij}, \xi_{ij}) = \sum_{j \in \Gamma(i)} 1 = \mathrm{card}(\Gamma(i)). \tag{15}$$

Moreover, using the Einstein notation for tensor scalar product $A : B = A_{ij} B_{ij}$, Equation (15) can be equivalently written as

$$M^i : \left(\sum_{j \in \Gamma(i)} \xi_{ij} \otimes \xi_{ij} \right) = |\Gamma(i)| \tag{16}$$

then

$$M^i : \underbrace{\left(\frac{1}{|\Gamma(i)|} \sum_{j \in \Gamma(i)} \xi_{ij} \otimes \xi_{ij} \right)}_{\mathbf{X}^i} = 1. \tag{17}$$

Note that $\mathbf{X}^i$ is a symmetric positive definite matrix if the triangulation is admissible so that it does not contain degenerate elements. Under this assumption, $\mathbf{X}^i$ is non-singular,

$$(\mathbf{X}^i)^{-1} \mathbf{X}^i = \mathbf{I}_d, \quad \text{and also} \quad Tr((\mathbf{X}^i)^{-1} \mathbf{X}^i) = (\mathbf{X}^i)^{-1} : \mathbf{X}^i = d$$

which, by identification using the expression of the trace, yields

$$M^i = \frac{1}{d} (\mathbf{X}^i)^{-1}. \tag{18}$$

Therefore, the family of tensors defined by (18) at nodes i generates a discrete metric which is a good approximation of the natural metric transforming the edges in the mesh into unit edges. The idea is then to use interpolation operators allowing to reconstruct functions from $P_0(\mathscr{T}_h)$ to $P_1(\mathscr{T}_h)$.

3.2. Gradient construction

The finite element gradient of the piecewise linear function u_h is well-defined at the elements' interiors. A smoothing procedure should be applied to construct nodal gradient values. The gradient reconstruction used in the current work is based on a least squares approximation of ∇u_h along the edges of the mesh, similarly to the element-wise Zienkiewicz–Zhu (ZZ) interpolation operator [26]. In fact, this method is used on P_1 finite element spaces. The gradient is constant on the elements and discontinuous from one element to the other. In order to recover the gradient at the nodes X_i of the mesh, a patch consisting of the elements sharing the node X_i is constructed. Then a linear function is defined fitting in a least square sense gradient values at the centers of mass of the elements in the patch. Using the length distribution tensor, a continuous gradient will be defined directly at the nodes of the mesh and depending only on the solution's interpolation values. For each node X_i, the proposed gradient reconstruction $G^i \in \mathbb{R}^d$ is sought satisfying the least-squares minimization

$$G^i = \underset{G_h}{\mathrm{argmin}} \sum_{j \in \Gamma(i)} |(G_h - \nabla u_h) \cdot \xi_{ij}|^2 = \underset{G_h}{\mathrm{argmin}} \sum_{j \in \Gamma(i)} |(G_h \xi_{ij} - U_{ij}|^2 \tag{19}$$

where $\xi_{ij} \in \mathscr{E}(\mathscr{T}_h)$ is the edge vector of the pair of vertices (i, j).
The minimum can by obtained by setting the derivative of Equation (19) to zero *i.e.*

$$\sum_{j\in\Gamma(i)} \left(\xi_{ij}^T G^i \xi_{ij} - \xi_{ij}^T U_{ij}\right) = 0,$$

or equivalently,

$$G^i \cdot \underbrace{\sum_{j\in\Gamma(i)} \xi_{ij} \otimes \xi_{ij}}_{\text{card}(\Gamma(i))\mathbf{X}^i} - \xi_{ij}^T U_{ij} = 0.$$

The length distribution tensor is defined as

$$U^i = \frac{1}{\text{card}(\Gamma(i))} \sum_{j\in\tau_j} X_{ij}^T U_{ij}$$

which appears as a characteristic quantity of the local mesh resolution, so that the following compact form

$$G^i = (\mathbf{X}^i)^{-1} U^i \tag{20}$$

is used to compute the recovered gradient.

Note that the present gradient recovery technique is directly computed on the nodes of discrete space, so that the only requirement for its implementation is the knowledge of the approximate solution at the nodes, which in this case of piecewise linear elements reduces to the vertices of the mesh, where the discrete metric field is colocated.

4. Element-wise error estimator and mesh adaptation

In this section, the anisotropic error estimator used to drive the stopping criterion and the mesh adaptation procedure are described. The local mesh sizes needed to define the metric field, that identifies the new adapted mesh, are provided solving explicitly a minimization problem, where the error estimator serves as a cost function. According to this metric field, the mesh is adapted to follow the behavior of the solution. In this article the Euclidean norm of the velocity $\|\mathbf{u}\|$ will be considered as the scalar field of interest to drive the adaptation procedure.

Let us consider a triangulation $\mathscr{T}_h$ of the physical domain $\Omega \in \mathbb{R}^d$, and define $u_h = \|\mathbf{v}_h\|$ the finite element approximation of scalar field $u = \|\mathbf{v}\|$, where $\mathbf{v}_h$ is the finite element approximation of the exact velocity solution $\mathbf{v}$ to the Navier–Stokes equation. Another approximation introduced is due to the use of an iterative procedure to solve the discretized system that stems from the finite element formulation: the approximation of u_h, obtained with this iterative procedure at the iteration n, is denoted by u_h^n.

Almeida et al. in [4] provide an upper bound for the interpolation error in L^p norm

$$\|u - u_h\|_{L^p(\Omega)} \leqslant C(1 + N_{\mathscr{T}_h}^{-\alpha}) \|H_R(u_h)(\mathbf{x})(\mathbf{x} - \mathbf{x}_K) \cdot (\mathbf{x} - \mathbf{x}_K)\|_{L^p(\Omega_h)} , \tag{21}$$

with

$$\|H_R(u_h)(\mathbf{x})(\mathbf{x} - \mathbf{x}_K) \cdot (\mathbf{x} - \mathbf{x}_K)\|_{L^p(\Omega_h)} = \left(\sum_{K\in\mathscr{T}_h} \|H_R(u_h)(\mathbf{x})(\mathbf{x} - \mathbf{x}_K) \cdot (\mathbf{x} - \mathbf{x}_K)\|_{L^p(K)}^p \right)^{1/p} , \tag{22}$$

where $\mathbf{x}_K$ is the centroid of the element K and $H_R(u_h)$ is the recovered Hessian of the solution, obtained applying a double gradient recovering technique as first proposed by Zienkiewicz and Zhu [27].

If the finite element solution u_h is a good approximation of u, the L^p norm of the recovered Hessian can be used as *a posteriori* error estimator. However, rather than u_h the solution computed is actually u_h^n, which is the one obtained at the iteration n at which the iterative solver is stopped, so that

$$\|u - u_h^n\|_{L^p(\Omega)} \approx C' \|H_R(u_h^n)(\mathbf{x})(\mathbf{x} - \mathbf{x}_K) \cdot (\mathbf{x} - \mathbf{x}_K)\|_{L^p(\Omega_h)} . \tag{23}$$

To ensure that this approximation will not affect the overall error, and its estimate, a reliable stopping criterion is required to control the convergence of the iterative algorithm. This will be the topic of Section 5.

This formulation highlights how the error is not isotropically distributed, and depends on the behavior of the second order derivative of the solution. For the sake of completeness, the main ingredients are described in a detailed manner here. At this point, it can be of practical interest to introduce a spectral decomposition of H_R, providing a symmetric positive definite matrix

$$\mathscr{H} = R\Lambda R^T \,, \tag{24}$$

where R is the orthogonal matrix of the eigenvectors of H_R and $\Lambda = diag\{|\lambda_1|, \cdots, |\lambda_d|\}$ is the diagonal matrix of the absolute values of the eigenvalues of H_R. Then the resulting tensor can be written as

$$\mathscr{H} = R\Lambda R^T = |\lambda_1| e_1 \otimes e_1 + \cdots + |\lambda_d| e_d \otimes e_d \,, \tag{25}$$

where e_i are the eigenvectors of H_R. Let us consider an element $K \in \mathscr{T}_h$, then using (24) the following local error estimator can be derived from relation (23) as

$$\eta_K^p = \int_K \big(\mathscr{H}(u_h^n(\mathbf{x}_K))(\mathbf{x}-\mathbf{x}_K)\cdot(\mathbf{x}-\mathbf{x}_K)\big)^p \, d\mathbf{x} \,, \tag{26}$$

so that, substituting the spectral decomposition (25) in (26) yields

$$\eta_K^p = \int_K \Big(\sum_{i=1}^{d} |\lambda_i(\mathbf{x}_K)| [e_i(\mathbf{x}_K)\cdot(\mathbf{x}-\mathbf{x}_K)]^2\Big)^p \, d\mathbf{x} \,. \tag{27}$$

which is the error estimator used in Section 5 to drive the convergence of the iterative solver.

The goal of this adaptive algorithm, detailed in [28], is to use the error estimator as a cost function of an explicit minimization problem, so in the following two main properties are introduced to provide a simple bound for the estimator. First, the projection of $\mathbf{x}-\mathbf{x}_K$ on the e_i direction is $[e_i\cdot(\mathbf{x}-\mathbf{x}_K)]^2 = \mathbf{x}_i^2 \leq h_i^2$, then injecting this bound in (26)

$$\eta_K^p \leq \int_K \Big(\sum_{i=1}^{d} |\lambda_i(\mathbf{x}_K)| h_i^2\Big)^p d\mathbf{x} \,. \tag{28}$$

Secondly, exploiting the main concept behind anisotropic mesh adaptation that identifies the optimal mesh as the one aligned with the solution u, implies that the error is locally equidistributed in the principal directions, *i.e.*

$$|\lambda_1| h_1^2 = \cdots = |\lambda_i| h_i^2 = \text{constant} \,. \tag{29}$$

Using this property the bound can be rewritten

$$\eta_K^p \leq |K| \big(d|\lambda_d(\mathbf{x}_K)| h_d^2\big)^p \,, \tag{30}$$

where $|K|$ is the volume of element K.

Finally, a local indicator can be defined using the proposed upper bound

$$\widetilde{\eta}_K \;=\; d|k|^{\frac{1}{p}} |\lambda_d(\mathbf{x}_K)| h_d^2 \,. \tag{31}$$

This indicator is used as a functional of the following minimization problem

$$\begin{cases} \text{Find } h_K = \{h_{1,K}, \cdots, h_{d,K}\}, K \in \mathscr{T}_h \text{ that minimizes the cost function} \,, \\ F(h_K) = \displaystyle\sum_{K\in\mathscr{T}_h} \widetilde{\eta}_K^p \,, \\ \text{under the constraint} \quad N_{\mathscr{T}_h'} = C_0^{-1} \displaystyle\sum_{K\in\mathscr{T}_h} \int_K \prod_{i=1}^{d} \frac{1}{h_{i,K}} \, d\mathbf{x} \,, \end{cases} \tag{32}$$

where C_0 is the volume of the reference regular tetrahedron, $\mathscr{T}_h'$ is the new triangulation, and the constraint is on the number of nodes $N_{\mathscr{T}_h'}$.

This optimization problem has a unique solution for any $d \geqslant 2$ and the generalized form can be written as

$$\begin{cases} h_{d,K} &= \left[\frac{\beta}{\frac{(2p+d)}{d} C_{1,K}} \int_K C_{2,K}\, d\mathbf{x} \right]^{\frac{1}{2(p+d)}}, \\ h_{i,K} &= \left(\prod_{k=i}^{d-1} s_{k,K} \right) h_{d\mathbf{x}}\,, \quad 1 \leqslant i \leqslant d-1\,, \end{cases} \tag{33}$$

where $\beta, C_{1,K}, C_{2,K}$ can be computed explicitly, and $s_{i,K} = h_i/h_{i+1} = (|\lambda_{i+1}|/|\lambda_i|)^{1/2}$ are the stretching factors for the element K.

Using these results the metric field used by the remeshing algorithm reads

$$\mathscr{M} = \frac{1}{h_1^2} e_1 \otimes e_1 + \cdots + \frac{1}{h_d^2} e_d \otimes e_d\,. \tag{34}$$

5. Adaptive stopping criterion

The focus of this work lies in combining the error estimator presented in Section 4 with the stopping criterion used to control the convergence of the iterative solver, resulting in a faster simulation.

The main idea is to stop the linear solver iterations when the algebraic error $\|u_h - u_h^n\|_{L^2}$ is very small compared to the Hessian-based estimator of the discretization error in L^p norm, derived in (26). Following this approach, the error estimator is computed using u_h^n, the solution obtained at the iteration n at which the iterative solver is stopped. The error estimator η is used as a bound for the algebraic error, scaled by a user defined factor c that defines the relative reduction to impose on the algebraic error compared to the estimated discretization error: in practice two orders of magnitudes lower is considered small enough,

$$\|u_h - u_h^n\|_{L^2} \leqslant c \left(\sum_{K \in \mathscr{T}_h} \eta_K^2 \right)^{1/2}. \tag{35}$$

However, to have a computable stopping criterion an estimate of the algebraic error norm should be available: in general a reliable estimate of this quantity can be nontrivial to provide. Arioli in [29] presents a review of several techniques that can be used for this purpose, in the framework of symmetric problems. In the same framework, Picasso in [9] shows that, using anisotropic adapted meshes, the Euclidean norm of the residual $\|\mathbf{r}^n\|$ can be considered as a good approximation for the norm of algebraic error. In a previous work [14] we performed several tests to assess the reliability of this approximation on a convection-diffusion problem. Due to the nature of our linearized approach for the modeling of the Navier–Stokes equations, we will apply the same approximation to this work. The residual vector $\mathbf{r}^n$ is here defined using only the velocity components of the solution, as a consequence to the choice to use the norm of the velocity for the error estimator procedure. Empirical observations show that the final residual norm related to the pressure component is usually several orders of magnitude lower. After these considerations the stopping criterion (35) becomes

$$\|u_h - u_h^n\|_{L^2} \approx \|\mathbf{r}^n\| \leqslant c \left(\sum_{K \in \mathscr{T}_h} \eta_K^2 \right)^{1/2}. \tag{36}$$

In the following section we will use this stopping criterion for the solution of several test cases, comparing the result obtained using the usual approach with a fixed stopping criterion.

6. Control of the discretization error and choice of preconditioning

Let us consider the linear system

$$\mathrm{A}\mathbf{u} = \mathbf{b} \qquad \mathrm{A} \in \mathbb{R}^{n\times n} \qquad \mathbf{u},\mathbf{b} \in \mathbb{R}^{n}. \tag{37}$$

A preconditioner is a matrix or transformation M the inverse of which M^{-1} stands as a good approximation of A^{-1}, and such that $\mathrm{M}^{-1}v$ can be computed efficiently. For nonsymmetric linear systems, a preconditioner may be applied either to the left or to the right of A. With a left preconditioner, instead of solving (37), one solves the linear system

$$\mathrm{M}^{-1}\mathrm{A}\mathbf{u} = \mathrm{M}^{-1}\mathbf{b}, \tag{38}$$

and the initial residual is

$$\mathbf{r} = \mathrm{M}^{-1}(\mathbf{b} - \mathrm{A}\mathbf{u}) \tag{39}$$

by using the Krylov subspace $\mathscr{K}(\mathrm{M}^{-1}\mathrm{A}, \mathrm{M}^{-1}b)$ instead of $\mathscr{K}(\mathrm{A}, \mathbf{b})$. For a right preconditioner, one solves the linear system for $\mathbf{v} \in \mathbb{R}^n$

$$\mathrm{A}\mathrm{M}^{-1}\mathbf{v} = \mathbf{b}\ , \tag{40}$$

and the initial residual is

$$\mathbf{r} = \mathbf{b} - \mathrm{A}\mathbf{u}, \tag{41}$$

by using the Krylov subspace $\mathscr{K}(\mathrm{A}\mathrm{M}^{-1}, \mathbf{b})$, and the final solution is therefore

$$\mathbf{u} = \mathrm{M}^{-1}\mathbf{v}. \tag{42}$$

The convergence of a preconditioned KSP (Preconditioned Krylov subspace) method is then determined essentially by the condition number of the system, therefore it depends on the eigenvalues of $\mathrm{M}^{-1}\mathrm{A}$ [30], which are the same than those of $\mathrm{A}\mathrm{M}^{-1}$ as well as their eigenvectors. The left and the right preconditioners have similar asymptotic behavior, but they can behave differently in practice. The use of right, instead of left, preconditioners is recommended for two reasons [30]. Firstly, the stopping criterion of a Krylov subspace method is typically based on the norm of the residual of the preconditioned system, which may be significantly different from the true residual. Figure 1 shows an example where the norm of the preconditioned residual is significantly larger than the true residual. Secondly, which is a consequence of the first one, if the iteration terminates with a relatively large residual $\mathbf{r}$, then the error of the solution is bounded by

$$\|e\|_{L^2(\Omega)} = \|u_h - u\|_{L^2(\Omega)} = \|\mathrm{A}^{-1}(\mathrm{A}\mathbf{u} - \mathbf{b})\| \leq \|-\mathrm{A}^{-1}\mathbf{r}\| \leq \|\mathrm{A}^{-1}\|\|\mathbf{r}\| \tag{43}$$

with $\|\cdot\|$ the Euclidean norm (ℓ^2–norm) on $\mathbb{R}^n$

$$\|\mathbf{r}\| = \left(\sum_{i=1}^{n} r_i^2\right)^{\frac{1}{2}}. \tag{44}$$

Dividing both sides by $\|u_h\|_{L_2(\Omega)}$ and using the definition of the condition number $\kappa(\mathrm{A})$, then

$$\frac{\|e\|_{L_2(\Omega)}}{\|u_h\|_{L_2(\Omega)}} \leq \kappa(\mathrm{A})\frac{\|\mathbf{r}\|}{\|\mathrm{A}\|\|u_h\|_{L_2(\Omega)}} \tag{45}$$

so that a large residual r implies a large relative error in the solution even if the matrix A is well conditioned. The stability analysis of PDE discretization depends on the boundedness of $\|\mathrm{A}^{-1}\|$ in (17), therefore, one should not change the residual unless the preconditioner is derived based on *a priori* knowledge of the PDE discretization. One could overcome these issues by computing the true residual $\|\mathbf{r}\|$ at each step, but it would incur additional costs with the left preconditioner.

Developed by Saad and Schultz [31], the General Minimum Residual Method (GMRES) is one of the best-known iterative methods for solving large, sparse, nonsymmetric systems. The algorithm is based on the Arnoldi iteration

to minimize $\|\mathbf{r}_k\|$ in $\mathscr{K}_k(\mathrm{A},\mathbf{b})$ at the k-th step. Equivalently, it finds an optimal k-polynomial $P_k(\mathrm{A})$ such that $\mathbf{r}_k = P_k(\mathrm{A})\mathbf{r}_0$ and $\|\mathbf{r}_k\|$ is minimized, supposing the approximate solution has the form

$$\mathbf{u}^k = \mathbf{u}^0 + \mathrm{Q}_k\mathbf{z} \tag{46}$$

where Q_k is given by $\mathrm{Q}_k = [\mathbf{q}_1|\mathbf{q}_2|....|\mathbf{q}_k]$ which is an orthonormal basis of the Krylov subspace $\mathscr{K}_k(\mathrm{A},\mathbf{v})$ and

$$\mathrm{AQ}_k = \mathrm{Q}_{k+1}\tilde{\mathrm{H}}_k \ , \tag{47}$$

where $\tilde{\mathrm{H}}_k$ is a $(k+1)\times k$ upper Hessenberg matrix. Note that GMRES implemented in PETSc supports both left and right preconditioning, but the default is left preconditioning. An extension of GMRES, called *Flexible* GMRES (FGMRES), allow adapting preconditioner from iteration to iteration, and it only supports right preconditioners. Therefore, if FGMRES is available, one can use it as GMRES with right preconditioning by setting the preconditioner across iterations. When either symmetric or right preconditioners are used, the solution of the preconditioned

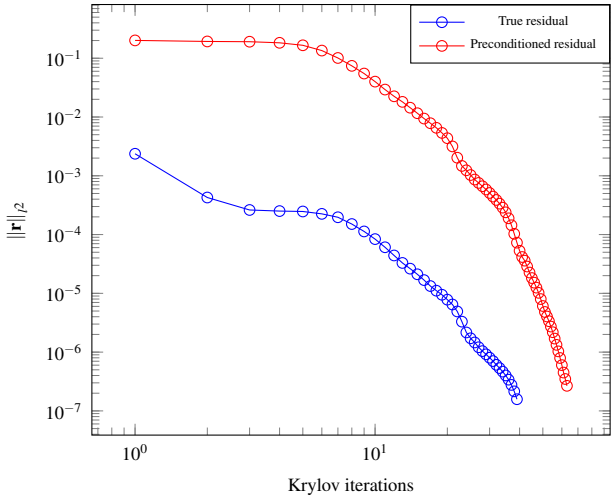


Figure 1: Convergence history of the GMRES residual for left and right preconditioners for one time-step of a Navier–Stokes manufactured solution (Section 7.1)

problem is evaluated. The solution of the original problem (the unpreconditioned one) is obtained using the right scaling operation used in order to scale the preconditioned system solution to a real solution. If preconditioning on the left is used nothing is done since the norm of the solution is the real one. For GMRES, such approach is employed to rewind the preconditioned residual obtained with left preconditioner to an unpreconditioned one also known as true residual using the left scaling.

Finally, the norm of the inverse operator appears as a bound constant, which reflects the scaling effect of the condition number of the discrete operator related to a given Krylov iterator. In the present work this constant is evaluated using a fast estimate of the extremal singular values during the Krylov iterations. Special care should be taken to have a reliable estimate: firstly a deviation calculated on a sliding average of extremal singular values is used to ensure enough eigenmodes are taken into account, and also the adaptive criterion is deactivated for several iterations after each GMRES restart.

7. Numerical results

7.1. Manufactured solution (2D)

As a verification test case, the Navier–Stokes problem is considered in the unit square $\Omega = [0,1]^2$ presented in [32]. This subject has been addressed by several authors ([33], [34]). The well-known method of manufactured

solutions (MMS) consists of evaluating the performance of numerical solvers on analytical solutions for a set of partial differential equations [35],[36] by constructing the corresponding forcing term. This method has been used with great success for the verification of compressible flows [37] or eddy-viscosity models [38, 39]. The source term and the boundary conditions are determined so that the chosen velocity–pressure solution satisfies relation

$$\rho \frac{\partial \mathbf{u}}{\partial t} + \rho(\mathbf{u}\cdot\nabla)\mathbf{u} - \nabla\cdot(\mu(\nabla\mathbf{u} + \nabla\mathbf{u}^T)) + \nabla p = \mathbf{f}\,, \tag{48}$$

and additionally the analytical expression for the velocity is solenoidal

$$\mathbf{u}(x,y) = \begin{pmatrix} \sin(\pi x)\sin(\pi y)\sin(t) \\ -\cos(\pi x) sin(\pi y)\sin(t) \end{pmatrix} \tag{49}$$

so that $\nabla\cdot\mathbf{u} = 0$ is satisfied pointwise, while the pressure is defined as

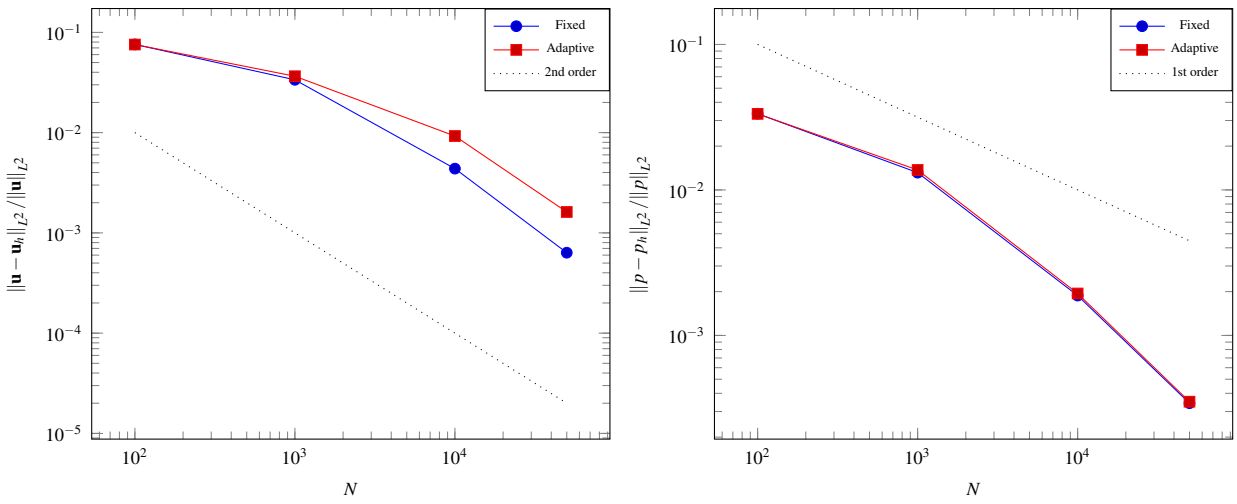
$$p(x,y) = (\sin(\pi x)\sin(\pi y)\cos(t))\,, \tag{50}$$

and model constants as $\rho = 1$, $\mu = 10^{-6}$.
Consequently, the source term is given by the following expression

$$\begin{aligned} \mathbf{f}(x,y) = &\begin{pmatrix} \sin(\pi x)\cos(\pi y)\cos(t) + \pi\cos(\pi x)\sin(\pi x)\cos(\pi y)^2\sin(t)^2 \\ -\cos(\pi x)\sin(\pi y)\sin(t) + \pi\sin(\pi x)^2\cos(\pi y)\sin(\pi y)\sin(t)^2 \end{pmatrix} \\ &+ \begin{pmatrix} \pi\cos(\pi x)\sin(\pi x)\sin(\pi y)^2\sin(t)^2 + \pi\cos(\pi x)\sin(\pi y)\cos(t) \\ \pi\cos(\pi x)^2\sin(\pi y)\cos(\pi y)\sin(t)^2 - 2\mu\pi^2\cos(\pi x)\sin(\pi y)\sin(t) \end{pmatrix} \\ &+ \begin{pmatrix} 2\mu\pi^2\sin(\pi x)\cos(\pi y)\sin(t) \\ \pi\cos(\pi y)\sin(\pi x)\cos(t) \end{pmatrix}. \end{aligned} \tag{51}$$

As previously specified, the linear system which stems from the finite element discretization of this problem is solved using the GMRES algorithm [31] with ILU factorization as a preconditioner. The final simulation time $T = 0.1s$ is reached with a time step $\delta t = 10^{-4}s$.

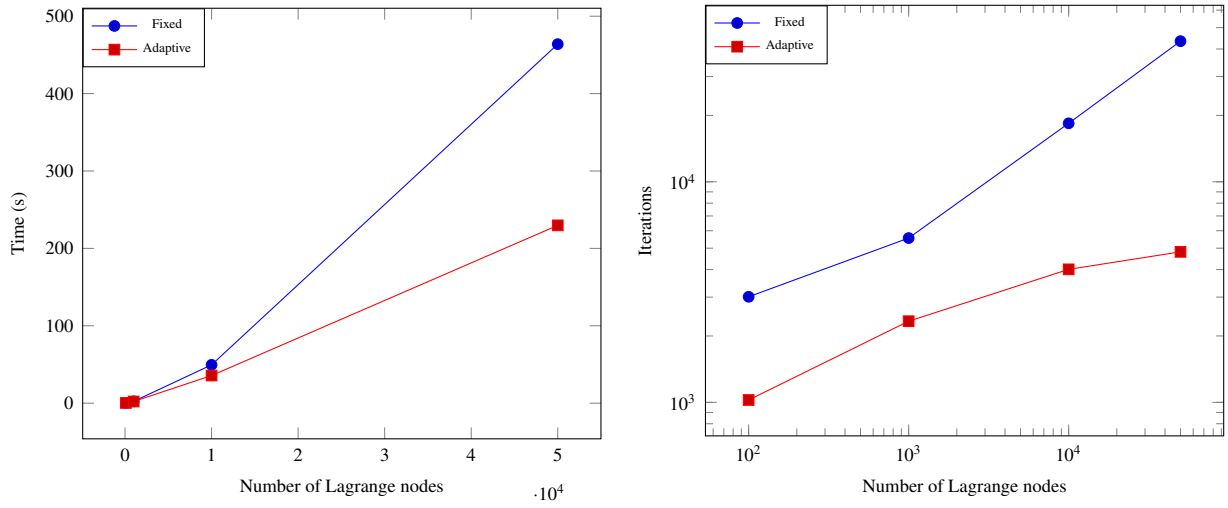
In Figure 2 and 3 are reported respectively the results in terms of convergence in the L^2 norm, and number of iterations of the linear solver used for the entire computation.



(a) Velocity *(b) Pressure*

Figure 2: Relative approximation error in the L^2 norm for fixed precision and adaptive stopping criterion based on the General metric estimator

Figure 2 left, shows that the use of the proposed stopping criterion does not affect much the convergence of the method, resulting in almost the same accuracy between the compared results. When the proposed adaptive stopping criterion (36) is activated, the total number of GMRES iterations decreases considerably; this behavior is highlighted in Figure 3 right.

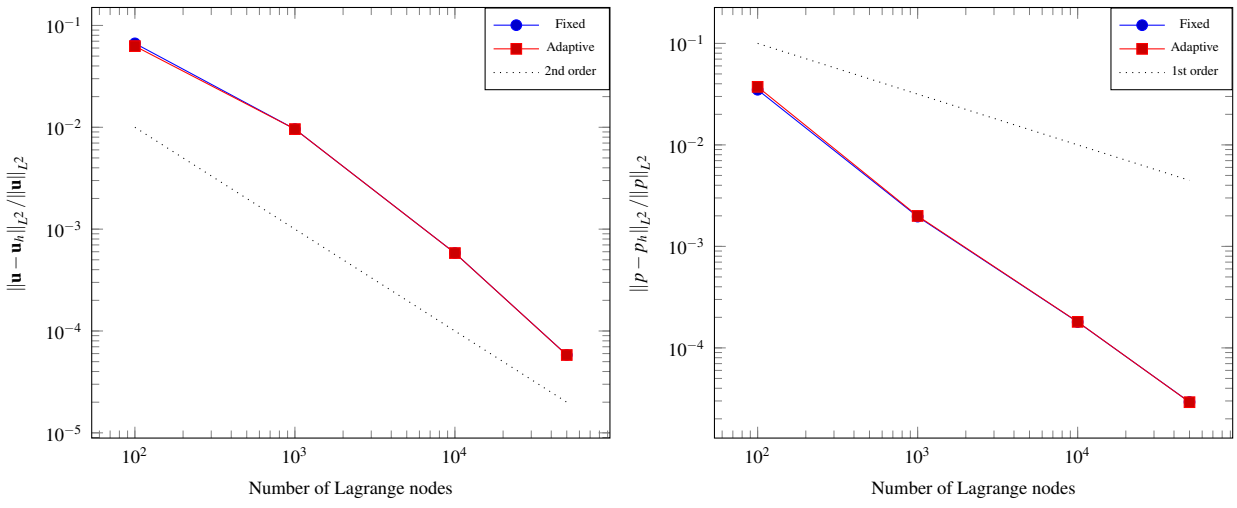


(a) Resolution time *(b) Number of Krylov iterations (log scale)*

Figure 3: Performance metrics for fixed precision and adaptive stopping criterion based on the General metric estimator

Finally, in Figure 3, are reported the results in term of computational time for three different adaptations with increasing refinement. Note that the adapted stopping criterion allows to reduce the required computational time by approximately 50% which is a significant gain in terms of efficiency.

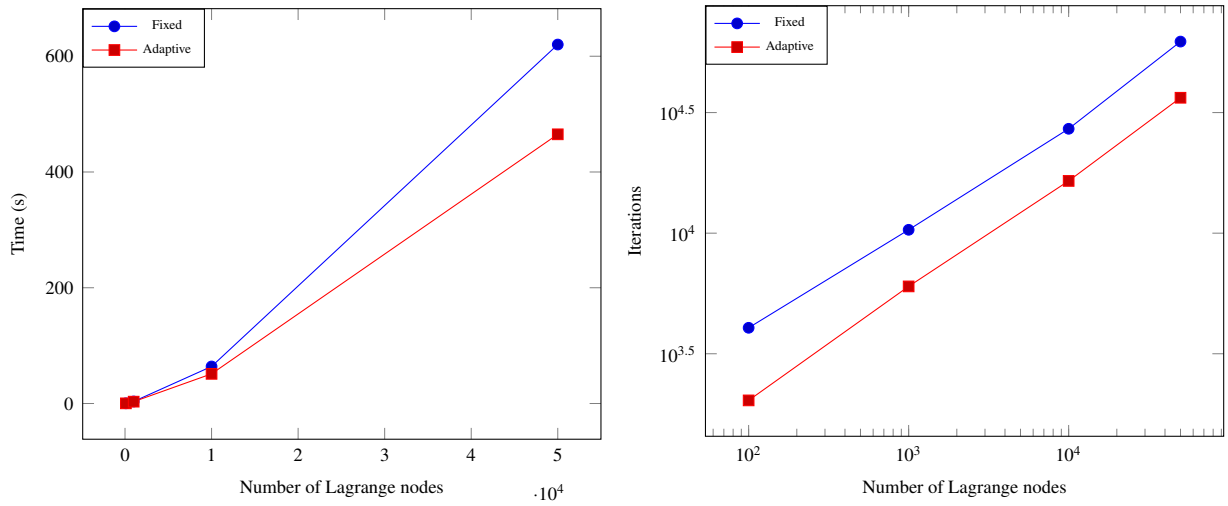
In Figures 4 and 5 are reported the results in terms of iterations, the error norm and the computational time for the Navier–Stokes problem and this time the Hessian metric is used to calculate the estimator.



(a) Velocity *(b) Pressure*

Figure 4: Relative approximation error in the L^2 norm for fixed precision and adaptive stopping criterion based on the Hessian metric estimator

Notice that the gain in terms of overall computational time is approximately 25%. Figure 4 right, depicts that the use of the Hessian metric does not affect the convergence of the method, resulting in almost the same accuracy between the compared results. In this test case, the general metric is to be preferred in terms of computational time reduction.



(a) Resolution time *(b) Number of Krylov iterations*

Figure 5: Performance metrics for fixed precision and adaptive stopping criterion based on the Hessian metric estimator

7.2. Flow past a square cylinder (2D)

To evaluate the proposed method on an incompressible flow problem, the well known square cylinder test case is considered, treated amongst others by [40, 41, 42, 43]. A two-dimensional square cylinder is placed in a computational domain, with his center in the origin of the coordinate system. The cylinder is exposed to a constant free-stream velocity U. As shown in Figure 6, the square has a length D and the distances from the upstream and downstream

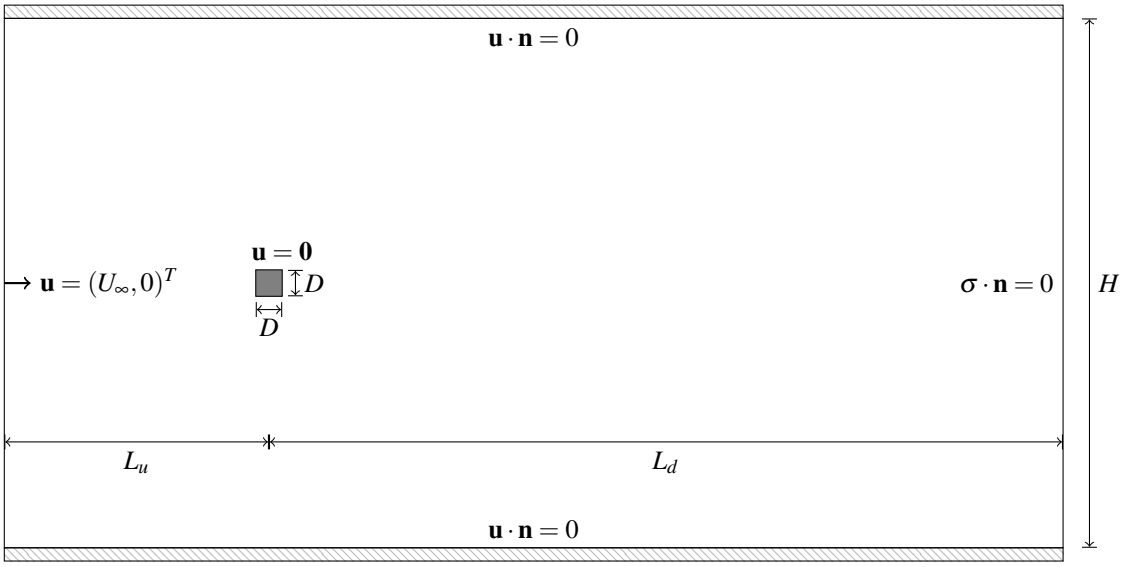


Figure 6: Problem definition

boundaries are respectively L_u and L_d, H is the distance between the sidewalls. Sohankar et al. [44] studied the effects of the placing of the boundaries on the flow. Following their conclusions the nondimensionalizing length scale D is chosen with $H = 20$ for a blockage $B \leq 5\%$, and the horizontal lengths are increased to $L_u = 10D$, $L_d = 30D$ as a safety measure.

7.2.1. Mesh sensitivity analysis

For the simulations performed on a fixed mesh, the computational domain is discretized with an isotropic mesh of triangles, where the zone around the cylinder has been refined to capture the boundary layer and the wake, as shown in Figure 7. To validate the choice of the mesh, the simulation is performed on five meshes M1, M2, M3, M4, M5.

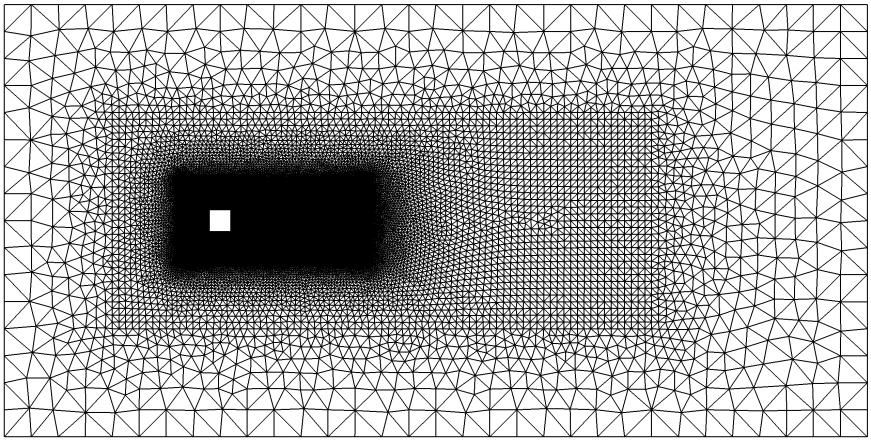


Figure 7: Isotropic triangular mesh M1.

Each mesh is obtained from the previous one applying a global uniform refinement, reducing the element size by 25%. The simulations performed with the mesh adaptation technique presented in Section 4 are carried out using an initial

Table 1: Details of the different meshes used for the convergence study and flow parameters at Re=100.

Mesh	Elements	Cd_{avg}	Cl_{rms}
M1	35892	1.5034	0.2034
M2	50968	1.4989	0.2011
M3	79270	1.4966	0.2006
M4	142202	1.4954	0.2004
M5	315336	1.4952	0.2002
Adapted[1]	30000	1.4947	0.1999

coarse mesh and performing one adaptation step every three time steps. An example of one of the resulting adapted meshes is shown in Figure 8, where the two snapshots (b) and (c) are taken with an interval of roughly one-half of the shedding period.

The details of the meshes and the obtained results are listed in Table 1, in Figure 9 we show the convergence on the values of drag and lift coefficients (Cd, Cl). It can be observed that the step of refinement from M4 to M5 has a negligible effect of the results, so mesh M4 is used for all the following tests. The comparison of the integral flow parameters obtained on the M4 mesh with earlier results available in the literature is presented in Table 2. The time averaged drag coefficient (Cd_{avg}) is in good agreement with the references and the Strouhal number ($St = fD/U$, where f is the frequency of shedding) is within 3%. The r.m.s. value of the lift coefficient (Cl_{rms}) is slightly higher than the other references, nonetheless it is within 4% from the one reported by Sen et al. [40]. Figure 10 shows the contours of velocity with isovalues of vorticity, at different time steps, obtained on the M4 fixed mesh. In (a) the solution has yet to develop the instability that is highlighted in (b) and (c), where we can see the typical Karman vortex street; the two snapshots are taken with an interval of roughly one-half of the shedding period.

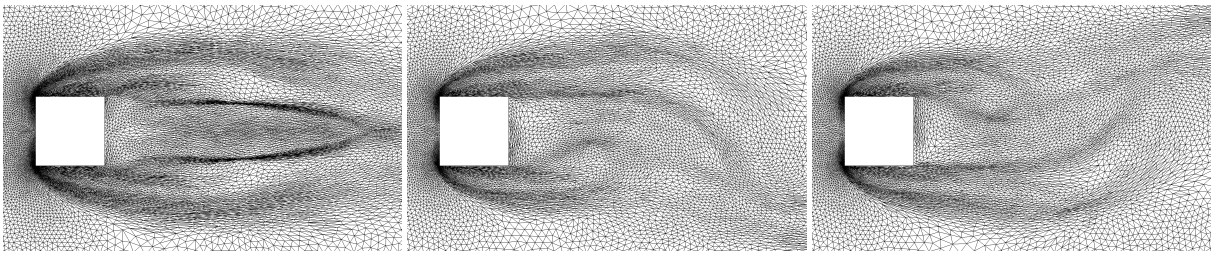


Figure 8: Adapted meshes, zoom on the object proximities.

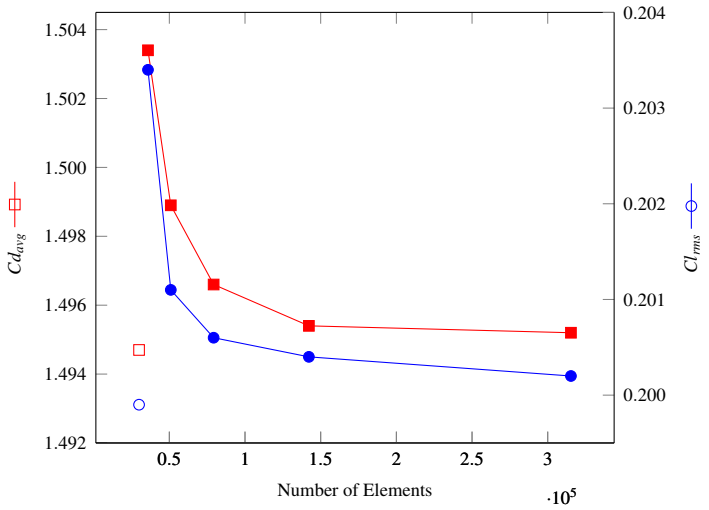


Figure 9: Mesh convergence results at Re=100: solid lines for the simulations using a fixed mesh, empty markers for the simulation with mesh adaptation.

Table 2: Comparison of integral flow parameters with references at Re=100.

Reference	B	Cd_{avg}	Cl_{rms}	St
Sharma and Eswaran [42]	0.0500	1.4936	0.1922	0.1488
Darekar and Sherwin [43] (3D)	0.0230	1.4860	0.1860	0.1460
Sahu et al. [41]	0.0500	1.4878	0.1880	0.1486
Sen et al. [40]	0.0500	1.5287	0.1928	0.1452
Present (Fixed)	0.0500	1.4966	0.2006	0.1416
Present (Adaptive)	0.0500	1.4947	0.1999	0.1416

7.2.2. *Stopping criteria comparison. Unsteady flow at Re=100.*

The efficiency of the adaptive stopping criterion proposed in Section 5 is investigated on the unsteady flow at Re=100 past a stationary square cylinder test case, presented above. Simulations were carried out on a fixed mesh and using mesh adaptation with *a posteriori* error estimation, comparing the proposed adaptive stopping criterion

$$\|\mathbf{r}^n\| \leqslant c \left(\sum_{K \in \mathcal{T}_h} \eta_K^2 \right)^{1/2}, \tag{52}$$

with a fixed precision stopping criterion, where a small enough fixed precision is imposed, here $\varepsilon = 10^{-6}$, so that the iteration are stopped whenever

$$\|\mathbf{r}^n\| \leq 10^{-6}. \tag{53}$$

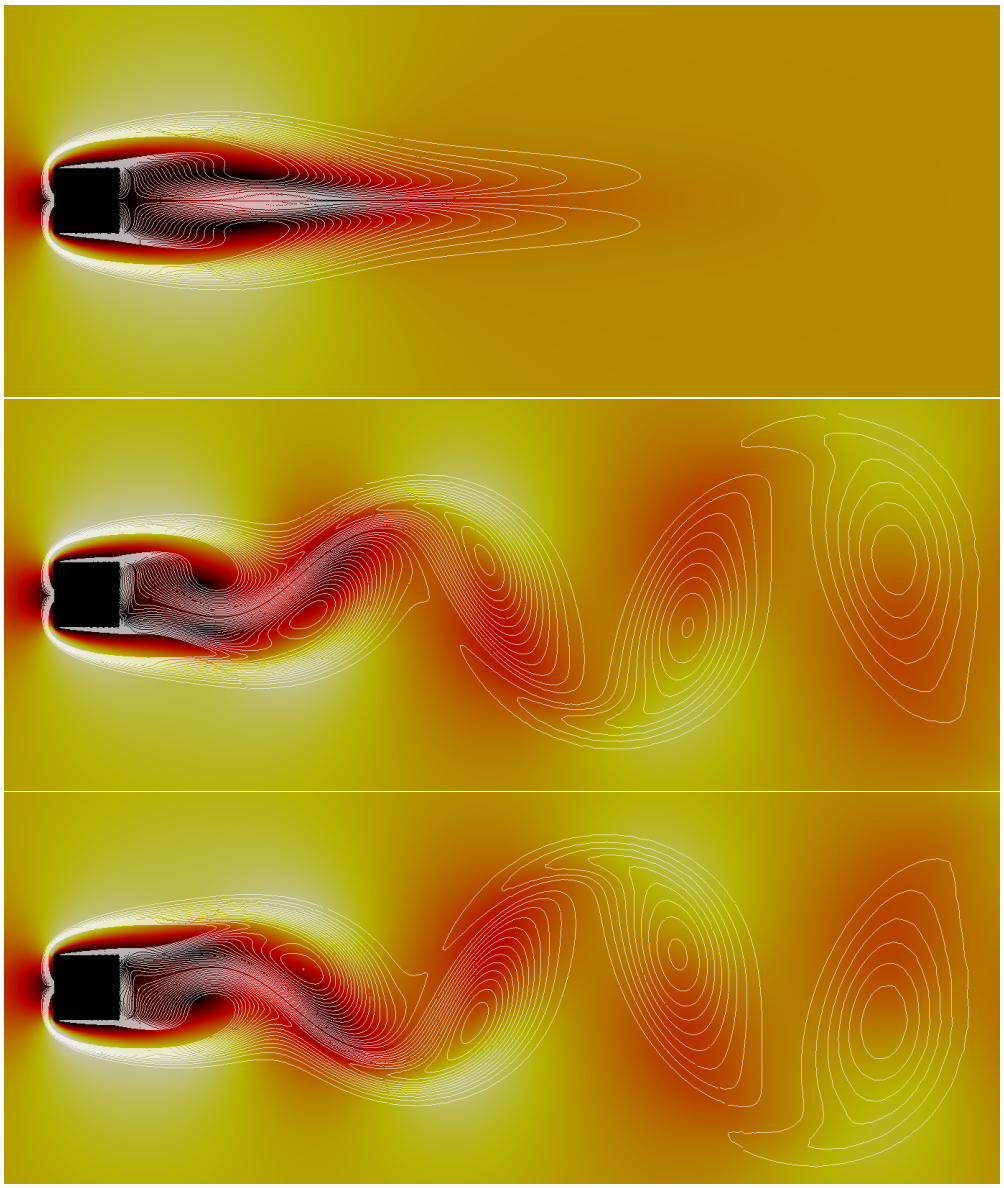


Figure 10: Velocity contours $\{0,1.4\}$, with isovalues of vorticity

Let us point out here that the chosen precision is not extremely strict (far from machine precision) but is typically used by the final users of commercial software, although no theory is behind this choice, but only the specific experience of the specific user, when its influence is deemed negligible on the resulting discrete solution. In Table 3 the results of the comparison in terms of precision are provided together with the computational time:

- *SC*: stopping criterion used. F (fixed) A (adaptive);
- *Scaling*: scaling factor for the stopping criterion, from Equation (36);
- Cd_{avg}, Cl_{rms}: average drag and r.m.s lift coefficients;
- *Walltime*: total computational time spent for the iterative solution for each simulation;

Table 3: Comparison of integral flow parameters with references at Re=100.

Mesh	SC	Scaling	Cd_{avg}	Cl_{rms}	Walltime(min)	
M4	F	/	1.4966	0.2006	84.9	
M4	A	c=0.1	1.4955	0.2004	39.5	-65%
M4	A	c=1.0	1.4955	0.2004	15.2	-82%
Adapted	F	/	1.4948	0.1999	10.3	
Adapted	A	c=0.1	1.4946	0.1997	4.8	-53%
Adapted	A	c=1.0	1.4937	0.1996	2.1	-79%

The results show that the application of the proposed stopping criterion for the iterative solver does not affect the precision of the measured integral values as the relative error is always below 0.1%. while a greatly reducing the computational time required, up to 82% compared to a fixed stopping criterion. Moreover, the estimated error in L^2-norm is in the order of magnitude of 10^{-4} for both fixed and adapted mesh simulations.

7.3. Workpiece cooling in quenching bath (3D)

The test case considered here is a three-dimensional configuration devised to model the cooling of high-temperature rectangular workpiece inside a quenching bath, where the incompressible Navier–Stokes equations are coupled with the thermal convection–diffusion equation [14]. The domain consists of a rectangular enclosure with three inlets on the front wall and two outlets, one on the left wall and the other one on the right wall see Figure 11. Inside the rectangular domain, an object is subjected to the thermal treatment: the initial temperature of the workpiece is equal to 150°C and that of the surrounding atmosphere is equal to 20°C. The air is introduced into the furnace at a constant velocity $U = 1m/s$. On all the walls, a temperature of 20°C is imposed. The simulation is run to a final time of $125s$, with a time step $\delta t = 0.1$s.

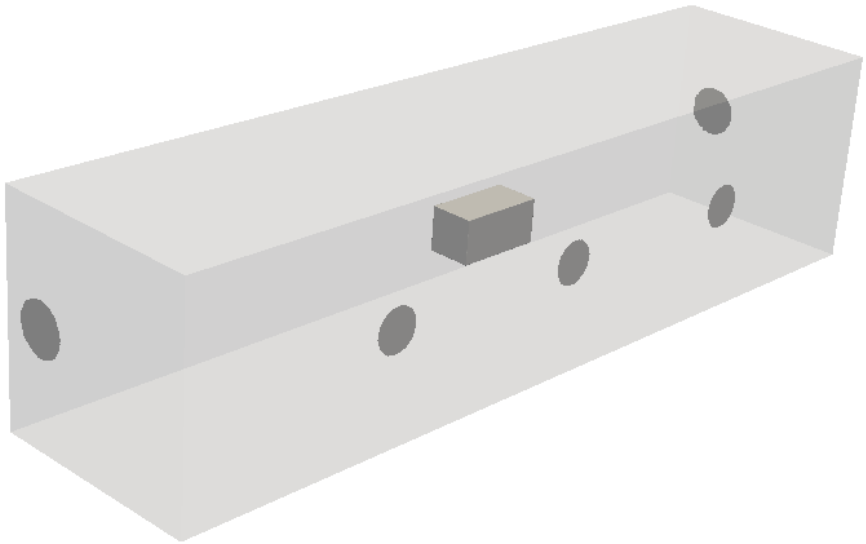


Figure 11: Computational domain for the quenching bath of a rectangular workpiece with three inlets on the side wall, two inlets at front and back walls.

A comparison on a fixed pre-adapted mesh made of 640112 elements is performed to eliminate the impact of dynamic remeshing. In Figure 12, are compared the time averaged values of the temperature in a point located in the submerged solid, using fixed and adaptive stopping criterion. The fixed precision have been set respectively to 10^{-6}

for the Navier–Stokes solver and to 10^{-5} for the thermal equation solver: these values are typical choices for a good compromise between accuracy and performance. The plot in Figure 13 shows that the difference between the two results is negligible as the relative error between the temperatures computed with the two stopping strategies reaches a maximum of 2.5 percent towards the final time of the simulation. Therefore the proposed method does not degrade the accuracy of the quantity of interest.

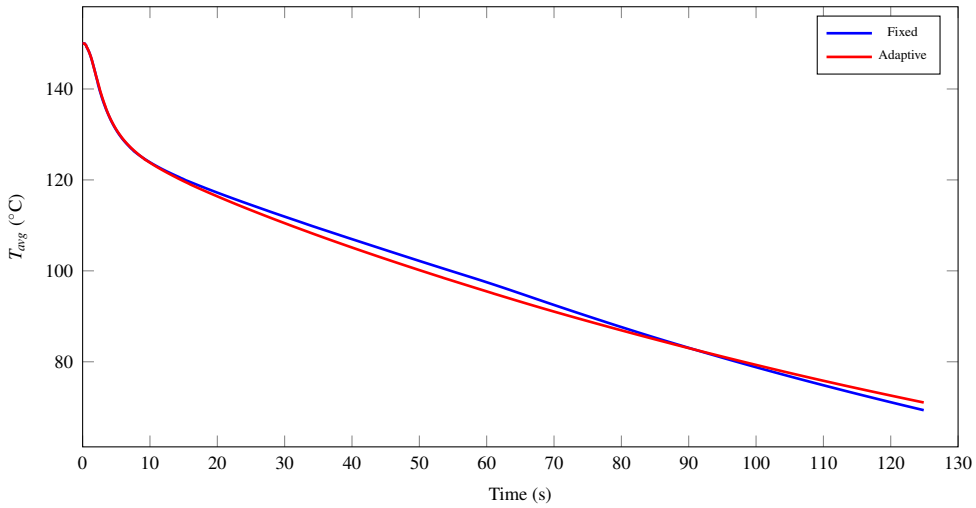


Figure 12: Evolution of the time averaged temperature on the immersed solid.

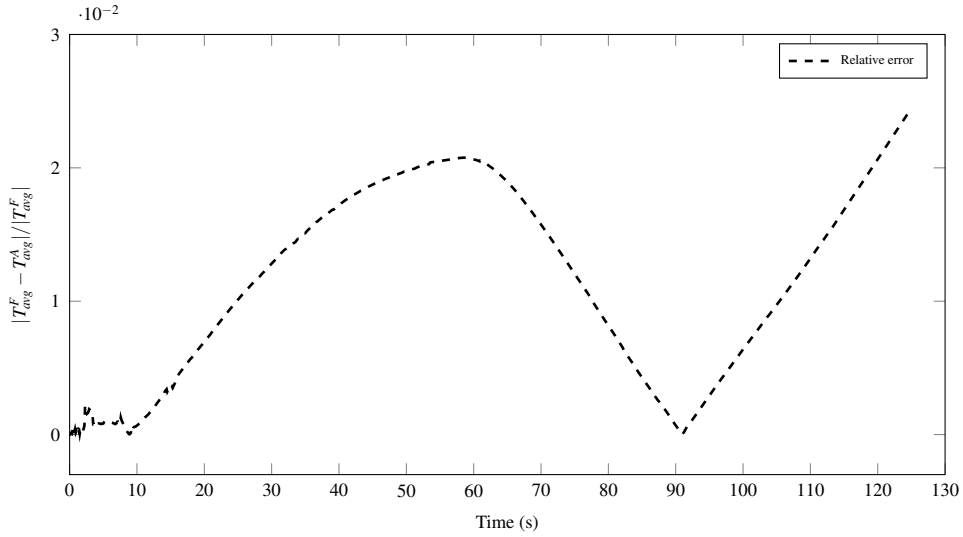


Figure 13: Evolution of the relative error between temperatures T^F_{avg} and T^A_{avg} obtained with fixed and adaptive strategies.

Additionally, the difference in temperature bounds in the entire computational domain does not exceed 0.5°C at the final time of the simulation, as depicted in the median cutplane Figure 14.

The results of the comparison between the fixed and adaptive stopping criterion in terms of number of iterations are provided in Figure 15 as well as the computational time needed in Figure 16. Regarding the number of iterations, the adaptive stopping criterion allows a reduction of approximately 65% which is noteworthy in terms of efficiency. However the impact on the resolution time for the thermal convection-diffusion-reactionequation, consists of a reduction up to 15% see Figure 16 left compared to the Navier–Stokes equations for which reduction is approximately 7%

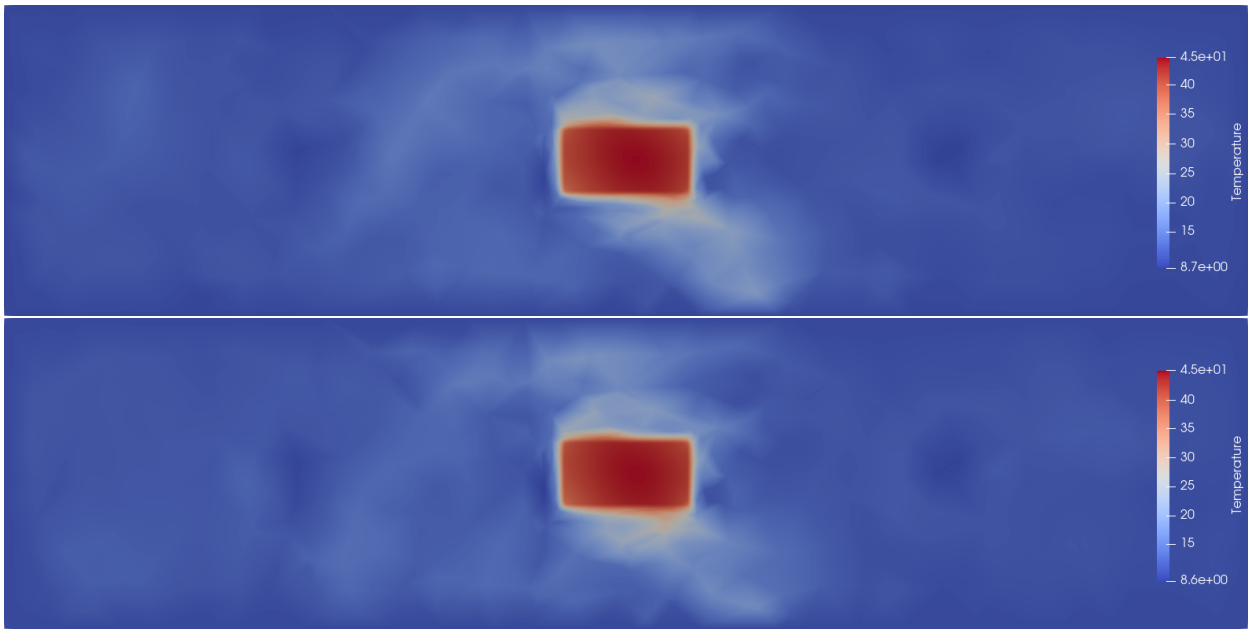


Figure 14: The final temperature map at $t = 125s$ in the vertical cutplane at the middle of the enclosure for fixed precision (top) and adaptive stopping criteria (bottom).

as depicted on Figure 16 right. Given the conservative fixed precision criteria chosen, the current approach allows a reasonable reduction in computational time. More importantly removes the need for user-defined input, which may lead to under-resolved approximate solutions.

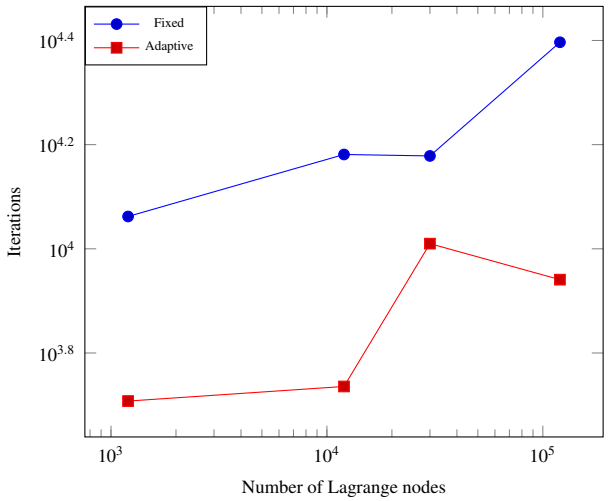
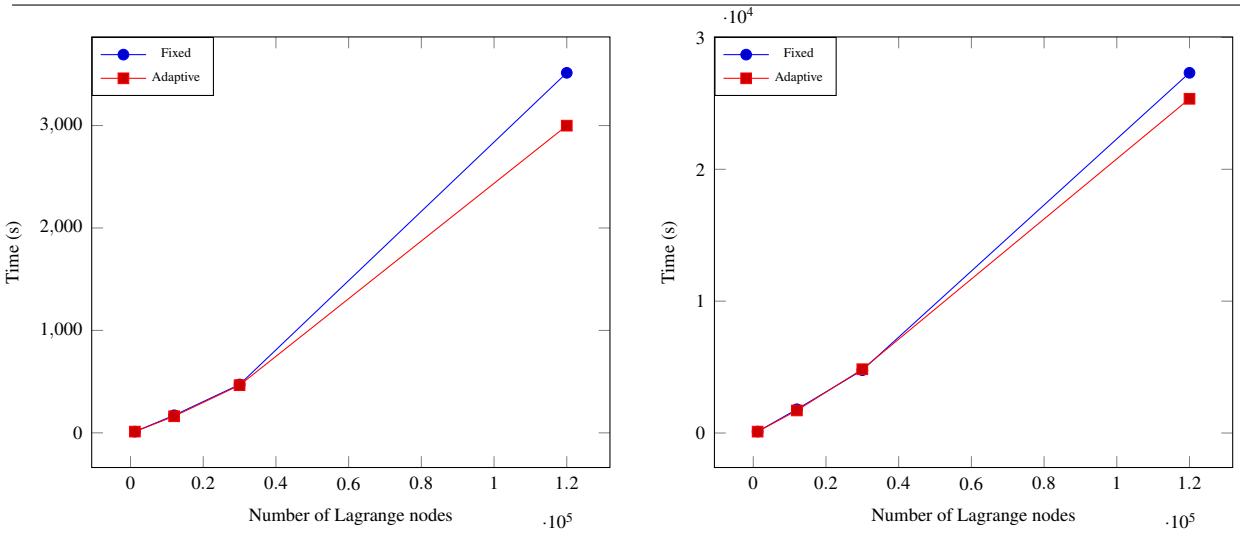


Figure 15: Total number of Krylov iterations for the workpiece cooling problem: fixed versus adapted stopping criterion

7.4. 3D fluid flow with thermal coupling

To assess further performance of the proposed method in realistic configurations, a three-dimensional test case is devised to model the fluid flow with heat transfer inside an industrial furnace. The incompressible Navier–Stokes equations are coupled with the thermal convection-diffusion-reaction equation as in the previous test case.



(a) Thermal equation solver *(b) Navier–Stokes solver*

Figure 16: Resolution time for the workpiece cooling problem: fixed versus adaptive stopping criterion

As shown in Figure 17, the problem is considered in a cubic domain with one inlet on the lower wall and two outlets on the top wall, where five cylinders are placed to be subjected to the thermal treatment. The air is introduced into the furnace at a constant velocity $U = 0.5m/s$ with a temperature of 400°C. On all the walls we impose an adiabatic condition for the temperature. The final time of the simulation is set to $t = 150s$, with a time step $\delta t = 0.01s$.

A comparison is first performed on a fixed mesh with 3.5M tetrahedral elements. Figure 18 shows the contours of temperature for different time steps. Time averaged values of temperature on a diagonal line that lays on the horizontal plane over the cylinders are compared using fixed and adaptive stopping criteria. The plot in Figure 20 shows that the difference between the two results are negligible, and the proposed method does not degrade the accuracy of the solution.

Table 4 presents the comparison of the stopping criteria in terms of computational time: The results show a great

Table 4: Comparison of integral flow parameters with references at Re=100.

Mesh	SC	Scaling	Walltime(h)	
Fixed 3.5M	F	/	80.6	
Fixed 3.5M	A	c=0.01	47.4	-41%

impact on the computational time needed for the iterative solution, with a reduction up to 41% compared to a fixed stopping criterion, with no sensible influence on the quantity of interest.

8. Conclusions

In this article an automated adaptive stopping criterion for iterative solvers was proposed in the framework of stabilized finite elements with anisotropic remeshing, and applied to Navier–Stokes problems with thermal coupling. The formulation takes advantage of the information computed in the mesh adaptation procedure, which provides an estimate of the discretization error with no additional computational cost. Krylov solver iterations can therefore be stopped when the algebraic error is sufficiently small compared to the estimated discretization error. Numerical tests have been performed for a benchmark on 2D and 3D problems ranging from manufactured solutions and an academic benchmark for incompressible flow, to industrial configurations with heat transfer. For the manufactured solutions the measure of accuracy considered chosen naturally as the approximation error showed no degradation, while the benchmark offers a point of comparison with existing results in the literature. For realistic configurations, as no objective

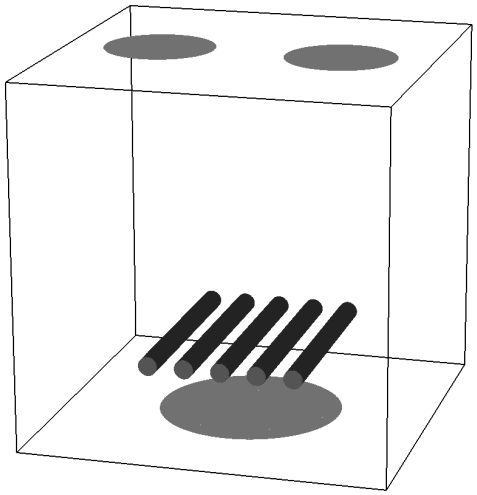


Figure 17: Problem definition of a prototypical furnace for pre-industrial simulations of heat treatments

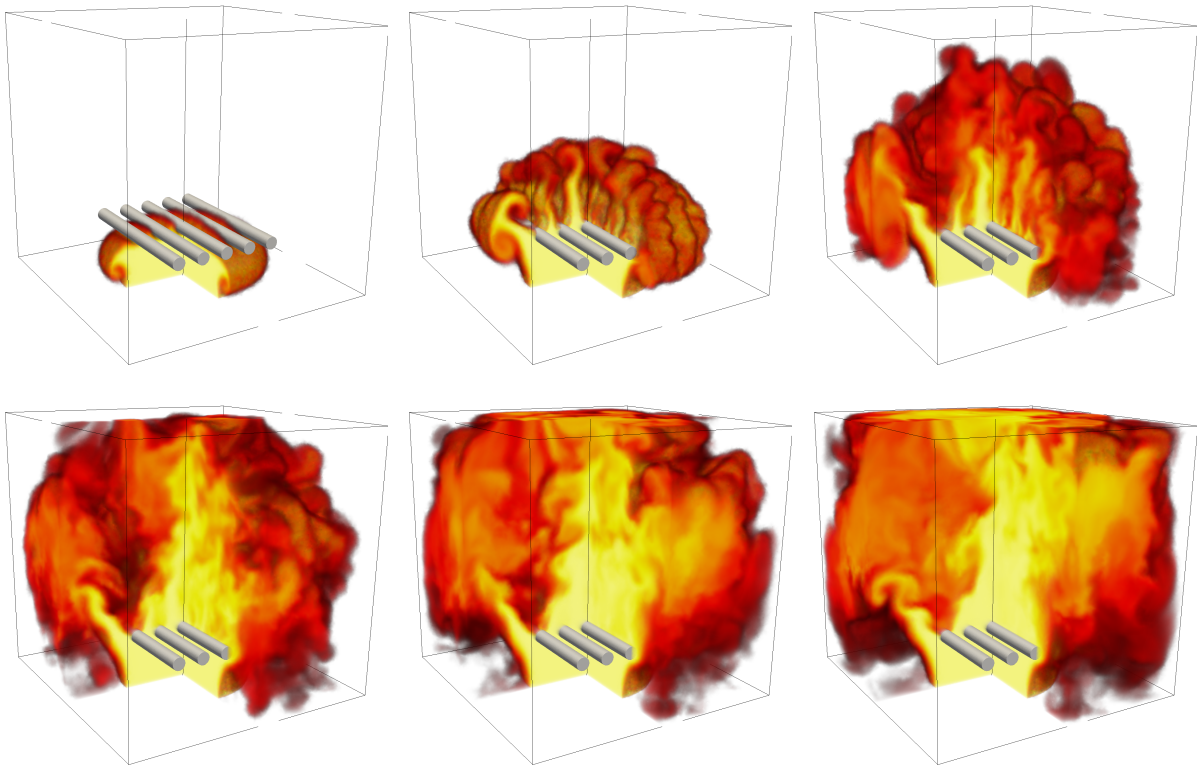


Figure 18: Temperature contours with one quarter section removed, values from 300° *to* 400° *Celsius*

280 measure of quality exists, the quantities of interests where chosen as space/time averaged values representative of the engineering problems considered. In all cases the proposed *a posteriori* error estimation framework with adaptive stopping criterion showed the potential of efficient computational time reduction up to 50 percent, with no significant effect on the accuracy of the computed solution with respect to the chosen metric. Future work will focus on the extension of the framework to versatile multicriteria error estimators able to cope better with a wide-range of multiphysics

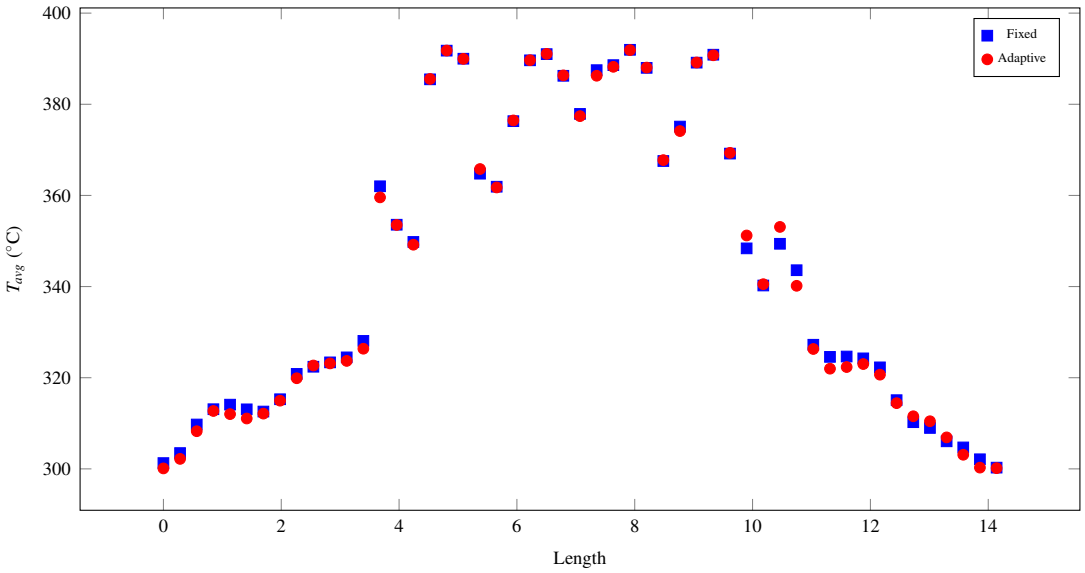


Figure 19: Time averaged temperature on the diagonal above the cylinders.

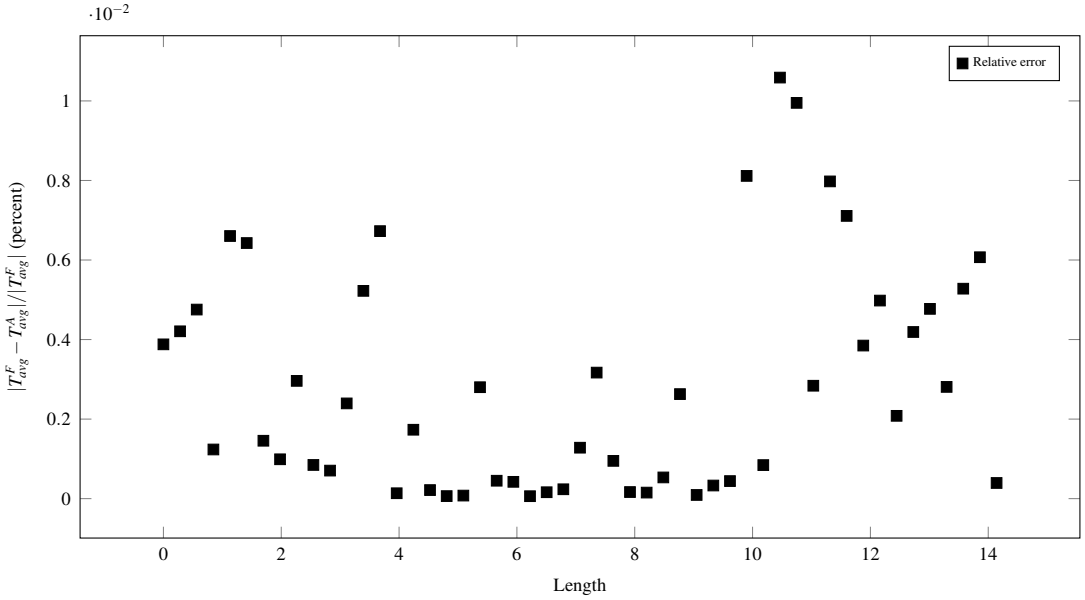


Figure 20: Relative error in time averaged temperatures T^F_{avg} and T^A_{avg} obtained with fixed and adaptive strategies.

problems. However they need to be tailored to the set of partial differential equations involved. In particular, while the current approach is expected to provide a meaningful stopping criterion for a wide-range of dissipative equations, the extension to hyperbolic systems is not direct. Finally a refined analysis of the lower bound of the *a posteriori* error estimate is considered to improve the robustness of the approach in the case of strongly anisotropic adapted meshes, where the overestimation of error indicators may result in under-resolving due to premature termination of linear iterations.

Acknowledgments

The authors acknowledge the support of the French Agence Nationale de la Recherche (ANR), under grant ANR-17-CHIN-0003 (project INFINITY).

References

[1] G. Jannoun, E. Hachem, J. Veysset, T. Coupez, Anisotropic meshing with time-stepping control for unsteady convection-dominated problems, Applied Mathematical Modelling 39 (7) (2015) 1899 – 1916.

[2] E. Hachem, G. Jannoun, J. Veysset, T. Coupez, On the stabilized finite element method for steady convection-dominated problems with anisotropic mesh adaptation, Applied Mathematics and Computation 232 (2014) 581 – 594.

[3] P. Frey, F. Alauzet, Anisotropic mesh adaptation for CFD computations, Computer Methods in Applied Mechanics and Engineering 194 (48) (2005) 5068 – 5082, Unstructured Mesh Generation.

[4] R. C. Almeida, R. A. Feijóo, A. C. Galeão, C. Padra, R. S. Silva, Adaptive finite element computational fluid dynamics using an anisotropic error estimator, Computer Methods in Applied Mechanics and Engineering 182 (3) (2000) 379 – 400.

[5] A. Agouzal, Y. V. Vassilevski, Minimization of gradient errors of piecewise linear interpolation on simplicial meshes, Computer Methods in Applied Mechanics and Engineering 199 (33) (2010) 2195 – 2203.

[6] A. Bazile, E. Hachem, J. Larroya-Huguet, Y. Mesri, Variational multiscale error estimator for anisotropic adaptive fluid mechanic simulations: Application to convection-diffusion problems, Computer Methods in Applied Mechanics and Engineering 331 (2018) 94 – 115.

[7] R. Becker, C. Johnson, R. Rannacher, Adaptive error control for multigrid finite element, Computing 55 (4) (1995) 271–288.

[8] M. Arioli, A stopping criterion for the conjugate gradient algorithm in a finite element method framework, Numerische Mathematik 97 (1) (2004) 1–24.

[9] M. Picasso, A stopping criterion for the conjugate gradient algorithm in the framework of anisotropic adaptive finite elements, Communications in Numerical Methods in Engineering 25 (4) (2009) 339–355.

[10] A. Ern, M. Vohralík, Adaptive inexact Newton methods with a posteriori stopping criteria for nonlinear diffusion PDEs, SIAM Journal on Scientific Computing 35 (4) (2013) A1761–A1791.

[11] P. Jiránek, Z. Strakoš, M. Vohralík, A posteriori error estimates including algebraic error and stopping criteria for iterative solvers, SIAM Journal on Scientific Computing 32 (3) (2010) 1567–1590.

[12] R. Stevenson, Optimality of a standard adaptive finite element method, Foundations of Computational Mathematics 7 (2) (2007) 245–269.

[13] M. Arioli, J. Liesen, A. Miçdlar, Z. Strakoš, Interplay between discretization and algebraic computation in adaptive numerical solution of elliptic PDE problems, GAMM-Mitteilungen 36 (1) (2013) 102–129.

[14] G. Manzinali, E. Hachem, Y. Mesri, Adaptive stopping criterion for iterative linear solvers combined with anisotropic mesh adaptation, application to convection-dominated problems, Computer Methods in Applied Mechanics and Engineering 340 (2018) 864–880.

[15] E. Hachem, B. Rivaux, T. Kloczko, H. Digonnet, T. Coupez, Stabilized finite element method for incompressible flows with high Reynolds number, Journal of Computational Physics 229 (23) (2010) 8643–8665.

[16] T. J. Hughes, L. Mazzei, K. E. Jansen, Large eddy simulation and the variational multiscale method, Computing and Visualization in Science 3 (1-2) (2000) 47–59.

[17] A. Ern, J.-L. Guermond, Theory and Practice of Finite Elements, Vol. 159, Springer Science & Business Media, 2004.

[18] T. J. Hughes, Multiscale phenomena: Green's functions, the Dirichlet-to-Neumann formulation, subgrid scale models, bubbles and the origins of stabilized methods, Computer Methods in Applied Mechanics and Engineering 127 (1-4) (1995) 387–401.

[19] R. Codina, Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods, Computer Methods in Applied Mechanics and Engineering 190 (13-14) (2000) 1579–1599.

[20] S. Mittal, On the performance of high aspect ratio elements for incompressible flows, Computer Methods in Applied Mechanics and Engineering 188 (1-3) (2000) 269–287.

[21] S. Micheletti, S. Perotto, 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 (3) (2003) 1131–1162.

[22] I. Harari, T. J. Hughes, What are c and h?: Inequalities for the analysis and design of finite element methods, Computer Methods in Applied Mechanics and Engineering 97 (2) (1992) 157–192.

[23] C. Förster, W. Wall, E. Ramm, Stabilized finite element formulation for incompressible flow on distorted meshes, International Journal for Numerical Methods in Fluids 60 (10) (2009) 1103–1126.

[24] A. Cangiani, E. Süli, The residual-free-bubble finite element method on anisotropic partitions, SIAM Journal on Numerical Analysis 45 (4) (2007) 1654–1678.

[25] T. E. Tezduyar, Y. Osawa, Finite element stabilization parameters computed from element matrices and vectors, Computer Methods in Applied Mechanics and Engineering 190 (3-4) (2000) 411–430.

[26] J. Z. Zhu, O. C. Zienkiewicz, Superconvergence recovery technique and a posteriori error estimators, International Journal for Numerical Methods in Engineering 30 (1990) 1321–1339.

[27] O. C. Zienkiewicz, J. Z. Zhu, A simple error estimator and adaptive procedure for practical engineering analysis, International Journal for Numerical Methods in Engineering 24 (2) (1987) 337–357.

[28] Y. Mesri, M. Khalloufi, E. Hachem, On optimal simplicial 3D meshes for minimizing the hessian-based errors, Applied Numerical Mathematics 109 (2016) 235 – 249.

[29] M. Arioli, E. H. Georgoulis, D. Loghin, Stopping criteria for adaptive finite element solvers, SIAM Journal on Scientific Computing 35 (3) (2013) A1537–A1559.

[30] A. Ghai, C. Lu, X. Jiao, A comparison of preconditioned Krylov subspace methods for large-scale nonsymmetric linear systems, Numerical Linear Algebra with Applications 26 (1) (2019) e2215.

[31] Y. Saad, M. Schultz, GMRES: a generalized minimal residual algorithm for solving nonsymmetric linear systems, SIAM Journal on Scientific and Statistical Computing 7 (1986) 856–869.

[32] B. Blais, F. Bertrand, On the use of the method of manufactured solutions for the verification of CFD codes for the volume-averaged Navier–Stokes equations, Computers & Fluids 114 (2015) 121–129.

[33] D. Zhou, High-order numerical methods for Pressure Poisson equation reformulations of the incompressible Navier–Stokes equations, 2014.

[34] E. Celledoni, B. K. Kometa, O. Verdier, High order semi-Lagrangian methods for the incompressible Navier–Stokes equations, Journal of Scientific Computing 66 (2016) 91–115.

[35] W. Oberkampf, C. Roy, Verification and Validation in Scientific Computing, Cambridge University Press, 2010. doi:10.1017/CBO9780511760396.

[36] P. Roache, Verification and Validation in Computational Science and Engineering, Hermosa Publishers, 1998.
URL `https://books.google.fr/books?id=ENRlQgAACAAJ`

[37] J. Waltz, T. Canfield, N. Morgan, L. Risinger, J. Wohlbier, Manufactured solutions for the three-dimensional Euler equations with relevance to inertial confinement fusion, Journal of Computational Physics 267 (2014) 196–209.

[38] L. Eça, M. Hoekstra, A. Hay, D. Pelletier, On the construction of manufactured solutions for one and two-equation eddy-viscosity models, International Journal for Numerical Methods in Fluids 54 (2) (2007) 119–154.

[39] L. Eça, M. Hoekstra, A. Hay, D. Pelletier, Verification of RANS solvers with manufactured solutions, Eng. Comput. (Lond.) 23 (2007) 253–270. doi:10.1007/s00366-007-0067-9.

[40] S. Sen, S. Mittal, G. Biswas, Flow past a square cylinder at low Reynolds numbers, International Journal for Numerical Methods in Fluids 67 (9) (2011) 1160–1174.

[41] A. K. Sahu, R. Chhabra, V. Eswaran, Two-dimensional unsteady laminar flow of a power law fluid across a square cylinder, Journal of Non-Newtonian Fluid Mechanics 160 (2-3) (2009) 157–167.

[42] A. Sharma, V. Eswaran, Heat and fluid flow across a square cylinder in the two-dimensional laminar flow regime, Numerical Heat Transfer, Part A: Applications 45 (3) (2004) 247–269.

[43] R. M. Darekar, S. J. Sherwin, Flow past a square-section cylinder with a wavy stagnation face, Journal of Fluid Mechanics 426 (2001) 263–295.

[44] A. Sohankar, C. Norberg, L. Davidson, Low-Reynolds-number flow around a square cylinder at incidence: study of blockage, onset of vortex shedding and outlet boundary condition, International Journal for Numerical Methods in Fluids 26 (1) (1998) 39–56.