



Solution of high-Reynolds incompressible flow with stabilized finite element and adaptive anisotropic meshing



T. Coupez, E. Hachem*

MINES ParisTech, Center for Materials Forming (CEMEF), UMR CNRS 7635, BP 207, 06904 Sophia-Antipolis, France

ARTICLE INFO

Article history:

Received 23 September 2011

Received in revised form 6 August 2013

Accepted 8 August 2013

Available online 4 September 2013

Keywords:

Anisotropic meshing

Boundary layers

Navier–Stokes

High Reynolds number

2D & 3D lid-driven cavity

ABSTRACT

The objective of this paper is to show that anisotropic meshes with highly stretched elements can be used to compute high Reynolds number flows. In particular, it will be shown that boundary layers, flow detachments and all vortices are well captured automatically by the mesh. We present an anisotropic meshing based on a posteriori estimation for the incompressible Navier Stokes equations. The proposed a posteriori estimate is based on the length distribution tensor approach and the associated edge based error analysis. The Finite Element flow solver is based on a Variational MultiScale (VMS) method, which consists in here of decomposing both the velocity and the pressure fields into coarse/resolved and fine/unresolved scales. This choice of decomposition is shown to be efficient for simulating flows at high Reynolds number. The stabilization parameters are determined taking into account the anisotropy of the mesh using a directional element diameter. The adaptation algorithm is applied to high Reynolds number flows inside the 2D and 3D lid-driven cavities and compared to accurate reference solutions.

© 2013 Elsevier B.V. All rights reserved.

1. Introduction

Anisotropic mesh adaptation has proved to be a powerful strategy to improve the quality and efficiency of finite element/volume methods. It enables to capture scale heterogeneities that can appear in numerous physical or mechanical applications including those having boundary layers, shock waves, edge singularities and moving interfaces [1–5]. In these cases, discontinuities or gradients of the solution are highly directional and can be captured with a good accuracy using an anisotropic mesh with few additional elements.

Several approaches to build unstructured anisotropic adaptive meshes easily are often based on local modifications [6–9] of an existing mesh. In fact, it mainly requires extending the way to measure lengths following the space directions and that can be done using a metric field to redefine the geometric distances. In parallel, theories of anisotropic a posteriori error estimation (i.e. [10]) have been well developed, leading to some standardization of the adaptation process; production of metrics from the error analysis of the discretization error and steering of remeshing by these metrics.

It follows that most adaptive anisotropic meshing techniques take a metric map as input. In practice, it is preferable to have a nodal metric map. Indeed, during the remeshing operations, the elements are much more volatile than the mesh nodes and therefore defining fields on a continuous basis ease their reconstruction, interpolation or extrapolation.

The author in [1] has adopted a different route for the metric construction. The latter is done directly at the node of the mesh without any direct information from the element, neither considering any underlying interpolation. It is performed by introducing a statistical concept: the length distribution function. A second order tensor was introduced to approximate the

* Corresponding author. Tel.: +33 (0)4 92 95 74 58; fax: +33 (0)4 92 38 97 52.

E-mail addresses: thierry.coupez@mines-paristech.fr (T. Coupez), elie.hachem@mines-paristech.fr (E. Hachem).

distribution of lengths defined by gathering the edges at the node. Using such a technique, he computed the error along each edge and in the direction of each edge.

In this work, we pursue further the approach developed in [1] in the sense that we extend this theory to take into account multi-component fields (tensors, vectors, scalars). Rather than considering several metric intersections and thus having much computation to perform, we propose herein an easy way to account for different fields in an a posteriori analysis while producing a single metric field. We will show that the proposed approach seems to be a powerful tool for approximating the solution of the incompressible Navier–Stokes equations as it accounts for all components of the velocity field. It is based on controlling the interpolation error of the velocity field as well as the velocity norm. Note that the proposed algorithm is implemented in the context of adaptive meshing under the constraint of a fixed number of nodes. With such an advantage, we can avoid evaluating precisely the constant arising in the error analysis and provide a very useful tool for practical Computational Fluid Dynamic (CFD) applications.

The leading idea of this work is to show that by applying the proposed anisotropic meshing to the recently developed flow solver [11], based on a Variational MultiScale (VMS) method, we are able to produce very good accuracy properties for high Reynolds number flows. Precisely, on meshes with highly stretched elements we use an appropriate definition of the stabilization parameters using the directional element diameter.

For that purpose, we will address exhaustively the well known driven cavity benchmark in 2D and 3D. We demonstrate that using the proposed error estimate, all small vortices, flow detachments and boundary layers inside the cavity are captured successfully.

The paper is structured as follows. In Section 2, we introduce the node based metric framework and describe the anisotropic mesh adaptation procedure governed by the length distribution tensor. In Section 3, the interpolation edge error for multi-component fields is described. Section 4 presents the developed VMS Navier–Stokes solver. Section 5 provides some numerical results and examples showing the capability of the new highly parallelized anisotropic mesh adaptation.

2. Edge based error estimation

In [1], we have derived an a posteriori error estimate based on the length distribution tensor approach and the associated edge based error analysis. In particular, we have derived the node based metric framework and described the anisotropic mesh adaptation procedure under the constraint of a fixed number of nodes. In this section, we only recall those results, but we invite the interested reader to check [1] for a detailed description and proofs. Let us recall some useful notations and definitions.

We consider $u \in C^2(\Omega) = \mathcal{V}$ and \mathcal{V}_h a standard P^1 finite element approximation space:

$$\mathcal{V}_h = \left\{ w_h \in C^0(\Omega), w_h|_K \in P^1(K), K \in \mathcal{K} \right\}$$

where $\Omega = \bigcup_{K \in \mathcal{K}} K$ is a simplex (segment, triangle, tetrahedron, ...) and \mathcal{K} is the set of elements of the mesh.

Denote by $\mathbf{X} = \{\mathbf{x}^i \in \mathbb{R}^d, i = 1, \dots, N\}$ the set of nodes of the mesh.

We also denote the Lagrange interpolation operator from \mathcal{V} to \mathcal{V}_h by:

$$\begin{aligned} \Pi_h : \mathcal{V} &\mapsto \mathcal{V}_h \\ u &\mapsto \Pi_h u \end{aligned}$$

such that

$$\Pi_h u(\mathbf{x}^i) = u(\mathbf{x}^i) = \mathbf{U}^i, \quad \forall i = 1, \dots, N$$

with \mathbf{U}^i being the node value of u at the mesh node location. We define the set of nodes connected to node i as shown in Fig. 1 by: $\Gamma(i) = \{j, \exists K \in \mathcal{K}, \mathbf{x}^i, \mathbf{x}^j \text{ are nodes of } K\}$.

By introducing the following notation: $\mathbf{x}^{ij} = \mathbf{x}^j - \mathbf{x}^i$ and $\mathbf{U}^{ij} = \mathbf{U}^j - \mathbf{U}^i$ we have for a P^1 element:

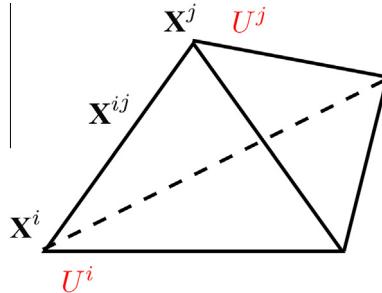


Fig. 1. length \mathbf{x}^{ij} of the edge joining nodes i and j .

$$\mathbf{U}^j = \mathbf{U}^i + \nabla u_h \cdot \mathbf{X}^{ij}$$

and

$$\nabla u_h \cdot \mathbf{X}^{ij} = \mathbf{U}^{ij} \quad (1)$$

Using the analysis carried in [1], we can set the following result:

$$\|\underbrace{\nabla u_h \cdot \mathbf{X}^{ij}}_{\mathbf{U}^{ij}} - \nabla u(\mathbf{X}^i) \cdot \mathbf{X}^{ij}\| \leq \max_{Y \in [\mathbf{X}^i, \mathbf{X}^j]} |\mathbb{H}(u)(Y) \mathbf{X}^{ij} \cdot \mathbf{X}^{ij}| \quad (2)$$

where $\mathbb{H}(u) = \nabla^{(2)}u$ is the associated Hessian of u . Recall that taking $u \in C^2(\Omega)$ we obtain $\nabla u \in C^1(\Omega)$ (Fig. 2).

Applying the interpolation operator on ∇u we get

$$\Pi_h(\nabla u)(\mathbf{X}^i) = \nabla u(\mathbf{X}^i) = \mathbf{g}^i.$$

where \mathbf{g}^i denotes the exact value of the gradient at node i . Such interpolation combined with (1) leads to the definition of $\mathbf{g}^{ij} = \mathbf{g}^j - \mathbf{g}^i$ given by:

$$\nabla^{(2)}u \mathbf{X}^{ij} = \mathbf{g}^{ij} \quad (3)$$

Finally by projecting (3) along the edge \mathbf{X}^{ij} we obtain:

$$(\nabla^{(2)}u) \mathbf{X}^{ij} \cdot \mathbf{X}^{ij} = \mathbf{g}^{ij} \cdot \mathbf{X}^{ij} \quad (4)$$

Consequently, we have from (4) a definition of the projected second derivative of u in terms of only the values of the gradient at the extremities of the edge. Recall also that due to the fact that $u_h|_K \in P^1(K)$ and u is continuous along the edge, the gradient projection along the edge is also continuous although it is a piecewise constant vector field.

2.1. Error along the edge

We denote the error along the edges using the following expression:

$$e_{ij} = \mathbf{g}^{ij} \cdot \mathbf{X}^{ij} \quad (5)$$

This error sampling is the exact interpolation error along the edge and enables us to evaluate the global L^1 error [1].

2.2. Gradient recovery

Eq. (5) can be evaluated only when the gradient of u is known and continuous at the nodes of the mesh. However the author in [1] proposes a recovery gradient operator defined by a local optimization problem as follows:

$$G^i = \operatorname{argmin}_G \sum_{j \in \Gamma(i)} |G \cdot \mathbf{X}^{ij} - \mathbf{U}^{ij}|^2 \quad (6)$$

This leads us to

$$G^i = (\mathbb{X}^i)^{-1} \sum_{j \in \Gamma(i)} \mathbf{U}^{ij} \mathbf{X}^{ij} \quad (7)$$

where $\mathbb{X}^i = \frac{1}{|\Gamma(i)|} \sum_{j \in \Gamma(i)} \mathbf{X}^{ij} \otimes \mathbf{X}^{ij}$ is what we call the length distribution tensor at node i . Additionally, the second order is preserved by using such a construction:

$$| (G^i - \nabla u(\mathbf{X}^i)) \cdot \mathbf{X}^{ij} | \sim | \mathbb{H}(u) \mathbf{X}^{ij} \cdot \mathbf{X}^{ij} |$$

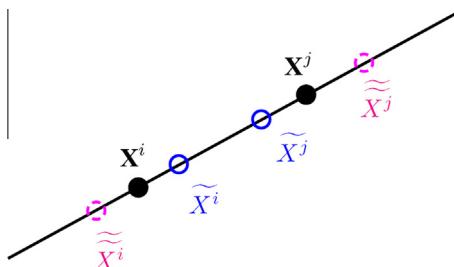


Fig. 2. Varying the edge in its own direction.

2.3. Error behavior due to varying the edge length

We examine now how the error behaves when we change the length of the edges. In fact the objective is to find the optimal length of all the edges of the mesh.

An interesting way of linking the error variations to the changes in edge lengths is by introducing a stretching factor $s \in \mathbb{R}$ such that

$$\begin{cases} \tilde{\mathbf{X}}^{ij} = s\mathbf{X}^{ij} \\ \tilde{e}_{ij} = s^2 e_{ij} = s^2 G^{ij} \cdot \mathbf{X}^{ij} \end{cases} \quad (8)$$

where \tilde{e}_{ij} and $\tilde{\mathbf{X}}^{ij}$ are the target error at edge ij and its associated edge length. Eqs. (8) mean that stretching the edge by a factor s changes the error quadratically and this corresponds to an optimal behavior for the P^1 elements.

On the other hand, from a meshing perspective, it is not sufficient to break the edge if the error is too large or to enlarge it if it is too small. Indeed, it is necessary to take into account the neighborhood of the node so that the best averaging representation is a metric defined at each node. Such a metric construction is the key point of [1].

2.4. Metric construction from the edge distribution tensor

The error analysis developed in the previous section together with the metric construction procedure leads to the “unit metric mesh”:

$$\mathbb{M}^i = \frac{1}{d} \left(\frac{1}{|\Gamma(i)|} \sum_{j \in \Gamma(i)} \mathbf{X}^{ij} \otimes \mathbf{X}^{ij} \right)^{-1}$$

and

$$\sum_{j \in \Gamma(i)} \mathbb{M}^i : \mathbf{X}^{ij} \otimes \mathbf{X}^{ij} = 1$$

For a complete justification of this result, the reader is referred to [1].

Consequently, for a set of stretching coefficients

$$\mathcal{S} = \{s_{ij} \in \mathbb{R}^+ \mid i = 1, \dots, N; j = 1, \dots, N; \Gamma(i) \cap \Gamma(j) \neq \emptyset\}$$

we can simply define the metric associated with \mathcal{S} as follows:

$$\mathbb{M}^i = \left(\frac{1}{d} \sum_{j \in \Gamma(i)} s_{ij}^2 \mathbf{X}^{ij} \otimes \mathbf{X}^{ij} \right)^{-1} \quad (9)$$

Note that \mathbb{M}^i is still positive definite when $s_{ij} \neq 0 \forall ij$.

2.5. Computing the stretching coefficient

We introduce $\eta_{ij} = s_{ij}^{-p}$ as being the number of created nodes as a result of reducing the edge by a factor s_{ij} . The exponent p intends to take into account the fact that the nodes are created not only along the edge but also implicitly from the crossing of the newly emerged edges. It is linked to the space dimension (2 or 3). However, any choice of p could be correct as long as the relation between the stretching s_{ij} factor and the number of created nodes tends to 1. In practice and for anisotropic meshes, we fixed $p = 1.5$ in all the following numerical experiments which gives in general a faster and satisfactory results.

Thus, if we fix the total number of edges to A , the optimal mesh will be derived using the following stretching factor definition:

$$s_{ij} = \left(\frac{\lambda}{e_{ij}} \right)^{\frac{1}{p+2}}$$

with

$$\lambda = \left(\frac{\sum_i \sum_{j \in \Gamma(i)} e_{ij}^{\frac{p}{p+2}}}{A} \right)^{\frac{p+2}{p}}$$

For further details about optimal factors, the reader is directed to [1].

To simplify and clarify the presentation, we propose in what follows the main steps needed for the metric construction at the nodes. The latter construction will subsequently lead to the required adaptive anisotropic meshing:

$$\mathbb{M}^i = \left(\frac{1}{d} \sum_{j \in \Gamma(i)} s_{ij}^2 \mathbf{X}^{ij} \otimes \mathbf{X}^{ij} \right)^{-1}$$

where

$$s_{ij} = \frac{\left(\sum_i \sum_{j \in \Gamma(i)} e_{ij}^{\frac{p}{p+2}} \right)^{\frac{1}{p}}}{(e_{ij})^{\frac{1}{p+2}}}$$

$$e_{ij} = G^{ij} \cdot \mathbf{X}^{ij}$$

and

$$G^i = (\mathbb{X}^i)^{-1} \sum_{j \in \Gamma(i)} \mathbf{U}^{ij} \mathbf{X}^{ij}$$

3. Extension to multi-component field

The main objective of this paper is to apply the adaptive anisotropic meshing to the Navier–Stokes equations. In that case we will adapt the mesh to the velocity fields. However the following remains valid for any number of components, i.e. it provides an extension of the way of adapting mesh to multiple criteria.

A common way to adapt a mesh to several components is to calculate the metrics corresponding to each of them and then to produce a unique metric by an operation known as the intersection of metrics.

In this work, we extend the anisotropic theory developed previously and we compute the error along the edges for a vectorial quantity. Therefore, by using an adequate scaling for all the components, we show in what follows that we can still produce only one metric.

Let us introduce $\mathbf{u} = \{u_1, u_2, \dots, u_n\}$,

$$\mathcal{V} = \mathcal{V} \times \mathcal{V} \times \dots \times \mathcal{V}$$

and

$$\mathcal{V}_h = \mathcal{V}_h \times \mathcal{V}_h \times \dots \times \mathcal{V}_h$$

The choice of using the above theory for each component of u turns out to be particularly handy with a view to a unique metric construction. We can still simply use the same theory developed previously without any necessary modification. It comes out immediately that the error is now a vector given by the following expression:

$$\mathbf{e}_{ij} = \{e_{ij}^1, e_{ij}^2, \dots, e_{ij}^n\}$$

Although there are other possibilities, we choose to produce a simple error value for each edge using the following:

$$1. L^2 \text{ case: } e_{ij} = \left(\sum_k (e_{ij}^k)^2 \right)^{\frac{1}{2}}$$

2. L^∞ case: $e_{ij} = \max_{k=1,\dots,n} e_{ij}^k$ and thus from a unique e_{ij} value, the scalar theory exposed in the previous sections remains valid and provides a unique metric. In the following numerical experiments, we used the L^2 case to compute the error. Indeed, for the velocity component we did not observe a significant difference between the two norms. However, in multiphysics application, the choice of the norms would affect the results and this requires further investigations.

Navier–Stokes at high Reynolds number exhibits the emergence of vortices of different sizes. In this work, we validate the proposed approach by comparing two and three dimensional results with very precise references on the well established lid driven cavity problem. The objective is then to adapt the mesh not only using the velocity norm, but also all the vector components.

Denote by v_h the finite element solution of the Navier–Stokes equations. In order to build a metric field and to construct then the corresponding mesh, we propose to use a normalised velocity and the proposed edge-based framework. However, in this work, we highlight the capability to account at the same time for the velocity and its components. Therefore, we consider the following nodal vector field $\mathbf{u}_h = \left(\frac{v_h}{|v_h|}, \frac{|v_h|}{\max|v_h|} \right)$ made of $d + 1$ components. We obtain then for every node i and direction j

$$\mathbf{u}_h(\mathbf{X}^i) = \mathbf{u}(\mathbf{X}^i) = \left\{ \frac{V^i}{|V^i|}, \frac{|V^i|}{\max_j |V^j|} \right\}$$

The edge-based metric technique as the Hessian-based metric one rely then on an interpolation error of a certain \mathbf{u} for which \mathbf{u}_h is the interpolant. Obviously the case $|V^i| = 0$ must be accounted for by using $\frac{V^i}{\max(e, |V^i|)}$ which means that $\mathbf{u}_h^i = 0$ when $|V^i| = 0$. All the following results will be using the error in the normalized vector field however the norm of the velocity will also be accounted for.

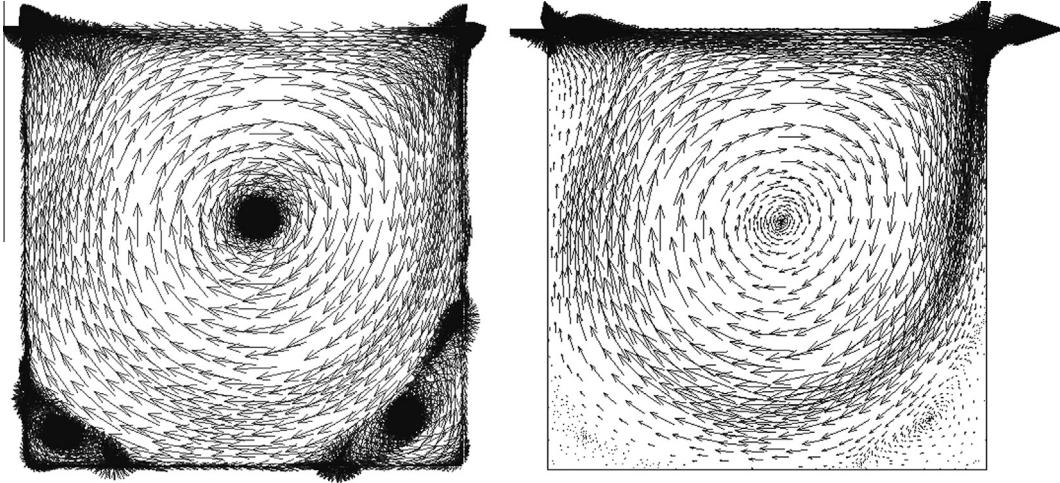


Fig. 3. The unit vector (left) and the velocity vector (right).

Note that adaptivity will account for both the change of direction of the velocity (as shown in Fig. 3) and its magnitude. Consequently, and as presented by the numerical results in the following section, even the small vortices developed by the solution will be very well captured. What is even more interesting is the capability of the method to automatically detect the boundary layers due to the anisotropically adapted mesh exhibiting highly stretched elements.

Finally, we recall that the used mesh technique referred as MTC is based on local modifications and on conformity controls through a minimal volume theorem. More details can be found in [6,5].

4. VMS: incompressible Navier–Stokes solver

In this section the general equations of time-dependent incompressible Navier–Stokes equation are solved. The stabilizing schemes from a variational multiscale point of view are revisited, described and presented [11]. 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 stabilization parameters will be determined rigorously taking into account the anisotropy of the mesh using a directional element diameter.

4.1. Basic principles of the multiscale approach

Following [12], we consider a direct sum decomposition of the velocity and the pressure fields into resolvable coarse-scale and unresolved fine-scale $\mathbf{v} = \mathbf{v}_h + \mathbf{v}'$ and $p = p_h + p'$. Likewise, we apply the same decomposition for the weighting functions $\mathbf{w} = \mathbf{w}_h + \mathbf{w}'$ and $q = q_h + q'$. The unresolved fine-scales are usually modelled using residual based terms that are derived consistently. The static condensation consists of substituting the fine-scale solution into the large-scale problem providing additional terms, tuned by a local stabilizing parameter. The latter enhances the stability and accuracy of the standard Galerkin formulation.

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 mixed-finite element approximation of the time-dependent Navier–Stokes problem can read:

Find a pair $\mathbf{v} : [0, T] \rightarrow V$ and $p : (0, T] \rightarrow Q$ such that : $\forall (\mathbf{w}, q) \in V_0 \times Q$

$$\begin{cases} \rho(\partial_t(\mathbf{v}_h + \mathbf{v}'), (\mathbf{w}_h + \mathbf{w}'))_\Omega + \rho((\mathbf{v}_h + \mathbf{v}') \cdot \nabla(\mathbf{v}_h + \mathbf{v}'), (\mathbf{w}_h + \mathbf{w}'))_\Omega \\ \quad + (2\mu \boldsymbol{\epsilon}(\mathbf{v}_h + \mathbf{v}') : \boldsymbol{\epsilon}(\mathbf{w}_h + \mathbf{w}'))_\Omega \\ \quad - ((p_h + p'), \nabla \cdot (\mathbf{w}_h + \mathbf{w}'))_\Omega = (\mathbf{f}, (\mathbf{w}_h + \mathbf{w}'))_\Omega \\ \quad (\nabla \cdot (\mathbf{v}_h + \mathbf{v}'), (q_h + q'))_\Omega = 0 \end{cases} \quad (10)$$

where ρ and μ are the density and the dynamic viscosity, respectively. To derive the stabilized formulation, we first solve the fine scale problem, defined on the sum of element interiors and written in terms of the time-dependent large-scale variables.

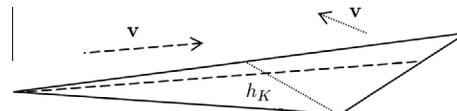


Fig. 4. Longest triangle lenght in the streamline direction.

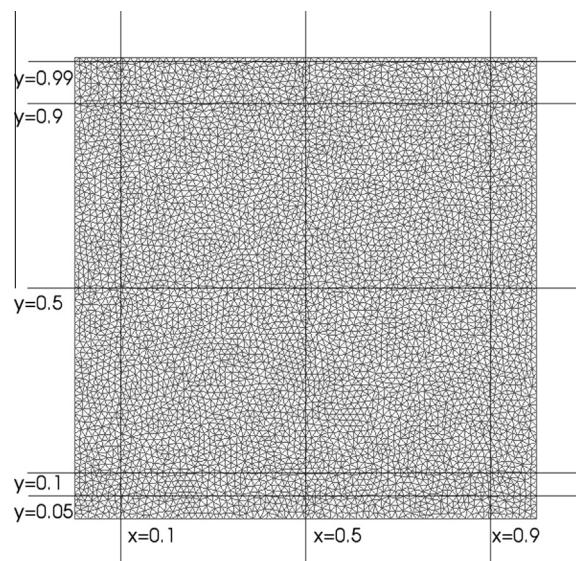


Fig. 5. The 2D computational domain and the corresponding plot axis.

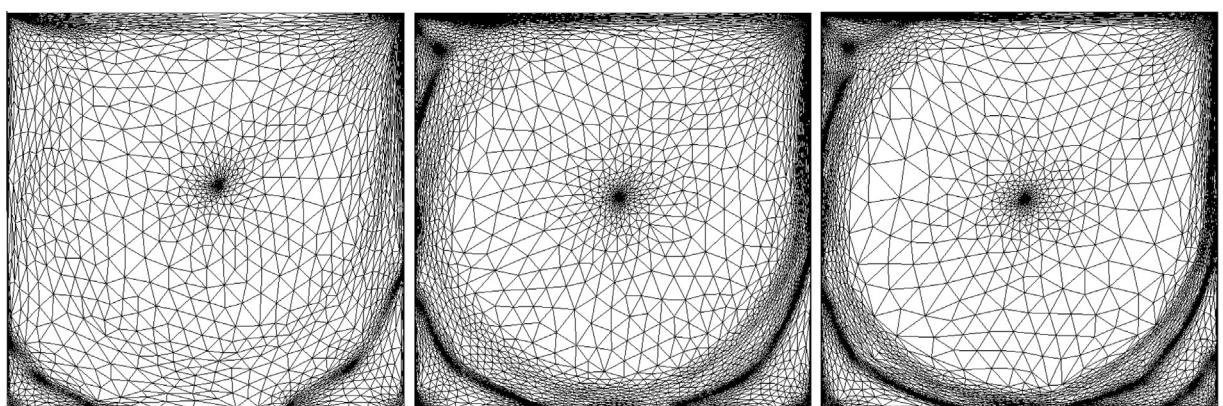


Fig. 6. Anisotropic meshes at Reynolds 1000, 5000 and 10,000.

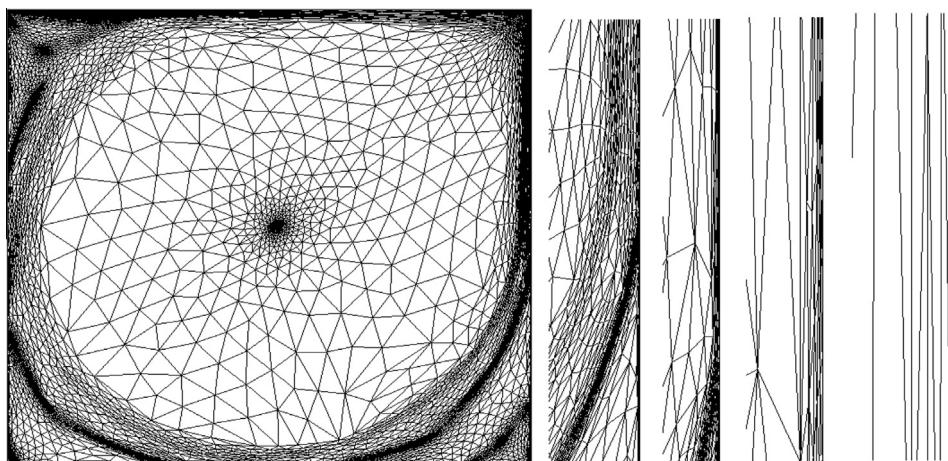


Fig. 7. Zooms on the mesh close to the right wall.

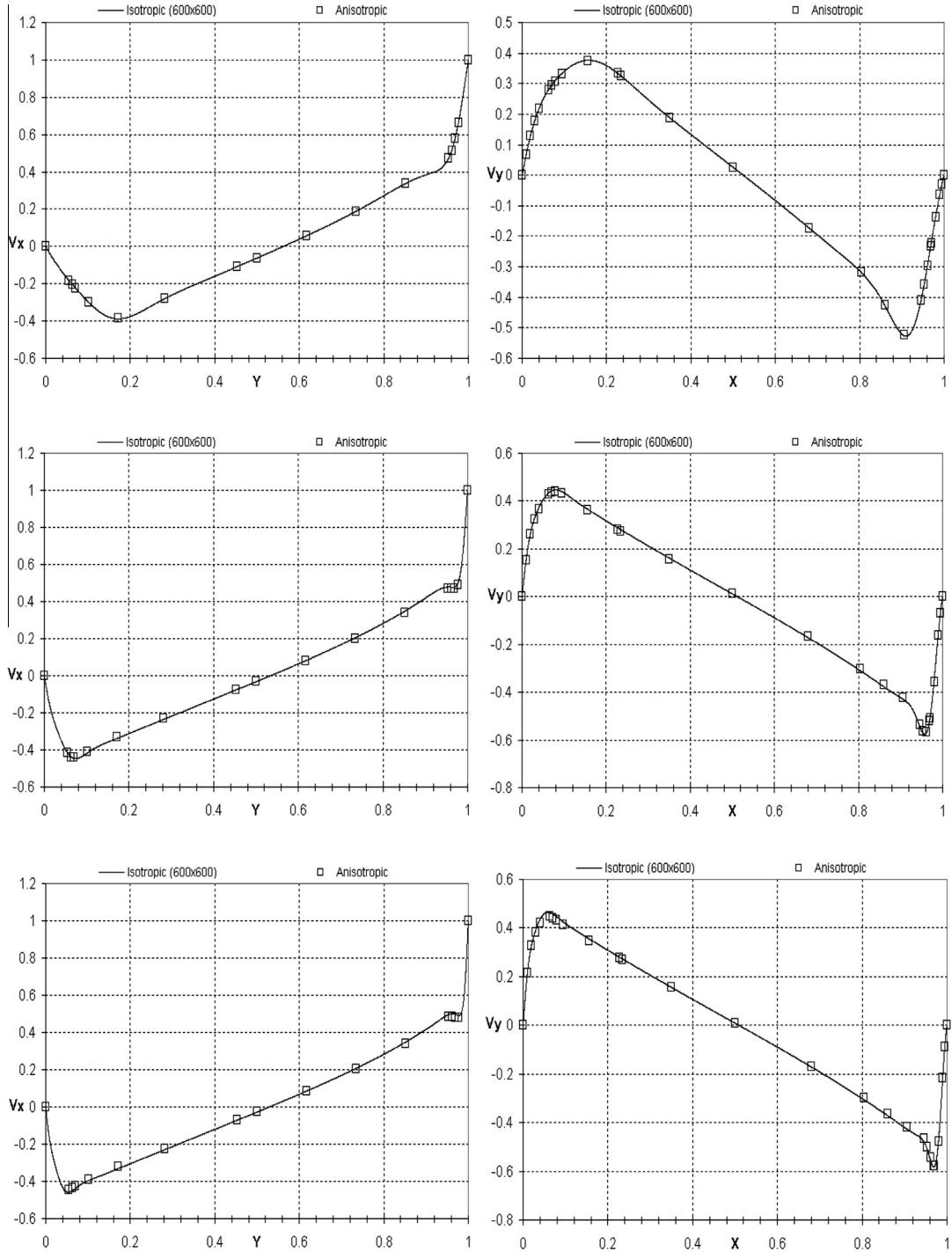


Fig. 8. Comparison of velocity profiles in the mid-planes for $Re = 1000$ (top), for $Re = 5000$ (middle) and for $Re = 10,000$ (bottom). Left: Velocity profiles for V_x along $x = 0.5$. Right: Velocity profiles for V_y along $y = 0.5$.

Then we substitute the fine-scale solution back into the coarse problem, thereby *eliminating the explicit appearance of the fine-scale while still modelling their effects*. It is worth mentioning that this would be equivalent to formulating the problem with a decomposition of linears (or bilinears) plus residual-free bubbles (see [13] for more details). At this stage, two important remarks have to be made in order to deal with the time-dependency and the non-linearity of the momentum equation of the subscale system:

Table 1Velocity profiles in median plane $x = 0.5$ at $Re = 1000$.

y	Ref. [27]	Ref. [28]	Isotropic	Anisotropic
1.0000	1.0000000	1.0000000	1.0000000	1.0000000
0.9766	0.6644227	0.6644194	0.6644520	0.6627950
0.9688	0.5808359	0.5808318	0.5808070	0.5798160
0.9609	0.5169277	0.5169214	0.5168170	0.5126750
0.9531	0.4723329	0.4723260	0.4721800	0.4709190
0.8516	0.3372212	0.3372128	0.3370270	0.3359990
0.7344	0.1886747	0.1886680	0.1885570	0.1883090
0.6172	0.0570178	0.0570151	0.0569487	0.0572112
0.5000	-0.0620561	-0.0620535	-0.0620790	-0.0616432
0.4531	-0.1081999	-0.1081955	-0.1082040	-0.1078430
0.2813	-0.2803696	-0.2803632	-0.2803280	-0.2808320
0.1719	-0.3885691	-0.3885624	-0.3883390	-0.3863410
0.1016	-0.3004561	-0.3004504	-0.3001060	-0.3009820
0.0703	-0.2228955	-0.2228928	-0.2225900	-0.2243200
0.0625	-0.2023300	-0.2023277	-0.2020430	-0.2041190
0.0547	-0.1812881	-0.1812863	-0.1810220	-0.1828680
0.0000	0.0000000	0.0000000	0.0000000	0.0000000

Table 2Velocity profiles in median plane $y = 0.5$ at $Re = 1000$.

x	Ref. [27]	Ref. [28]	Isotropic	Anisotropic
1.0000	0.0000000	0.0000000	0.0000000	0.0000000
0.995	–	–	-0.0291205	-0.0309492
0.99	–	–	-0.0616905	-0.0634961
0.98	–	–	-0.135746	-0.1383850
0.97	–	–	-0.217592	-0.2217140
0.9688	-0.2279225	-0.2279177	-0.2276530	-0.2288860
0.9609	-0.2936869	-0.2936814	-0.2935380	-0.2958380
0.9531	-0.3553213	-0.3553154	-0.3549680	-0.3577230
0.9453	-0.4103754	-0.4103691	-0.4099800	-0.4115380
0.9063	-0.5264392	-0.5264320	-0.5262860	-0.5240470
0.8594	-0.4264545	-0.4264492	-0.4263870	-0.4247730
0.8047	-0.3202137	-0.3202068	-0.3200460	-0.3188750
0.5000	0.0257995	0.0257987	0.0257945	0.0262780
0.2344	0.3253592	0.3253529	0.3252190	0.3249600
0.2266	0.3339924	0.3339860	0.3338450	0.3332520
0.1563	0.3769189	0.3769119	0.3767000	0.3750800
0.0938	0.3330442	0.3330381	0.3328310	0.3313150
0.0781	0.3099097	0.3099041	0.3097090	0.3076600
0.0703	0.2962703	0.2962650	0.2960800	0.2939980
0.0625	0.2807056	0.2807005	0.2805250	0.2796660
0.04	–	–	0.218777	0.2175300
0.03	–	–	0.179473	0.1773550
0.02	–	–	0.130519	0.1287840
0.01	–	–	0.0708388	0.0694771
0.0000	0.0000000	0.0000000	0.0000000	0.0000000

(i) the subscales are not tracked in time, therefore, quasi-static subscales are considered here (see [14] for a justification of this choice); however, the subscale equation remains quasi time-dependent since it is driven by the large-scale time-dependent residual. For time-tracking of subscales, see [15].

(ii) the convective velocity of the non-linear term may be approximated using only large-scale part so that $(\mathbf{v}_h + \mathbf{v}') \cdot \nabla(\mathbf{v}_h + \mathbf{v}') \approx \mathbf{v}_h \cdot \nabla(\mathbf{v}_h + \mathbf{v}')$ (see [11]).

Substituting the approximated \mathbf{v}' and p' into the large-scale equation and applying integration by parts we get:

$$\left\{ \begin{array}{l} \rho(\partial_t \mathbf{v}_h, \mathbf{w}_h)_\Omega + (\rho \mathbf{v}_h \cdot \nabla \mathbf{v}_h, \mathbf{w}_h)_\Omega - \sum_{K \in \mathcal{T}_h} (\tau_K \mathcal{R}_M, \rho \mathbf{v}_h \nabla \mathbf{w}_h)_K + (2\mu \boldsymbol{\varepsilon}(\mathbf{v}_h) : \boldsymbol{\varepsilon}(\mathbf{w}_h))_\Omega \\ - (\mathbf{p}_h, \nabla \cdot \mathbf{w}_h)_\Omega + \sum_{K \in \mathcal{T}_h} (\tau_C \mathcal{R}_C, \nabla \cdot \mathbf{w}_h)_K = (\mathbf{f}, \mathbf{w}_h)_\Omega \quad \forall \mathbf{w}_h \in \mathbf{V}_{h,0} \\ (\nabla \cdot \mathbf{v}_h, \mathbf{q}_h)_\Omega - \sum_{K \in \mathcal{T}_h} (\tau_K \mathcal{R}_M, \nabla \mathbf{q}_h)_K = \mathbf{0} \quad \forall \mathbf{q}_h \in \mathbf{Q}_h \end{array} \right. \quad (11)$$

with \mathcal{R}_M and \mathcal{R}_C the momentum and continuity residuals, respectively:

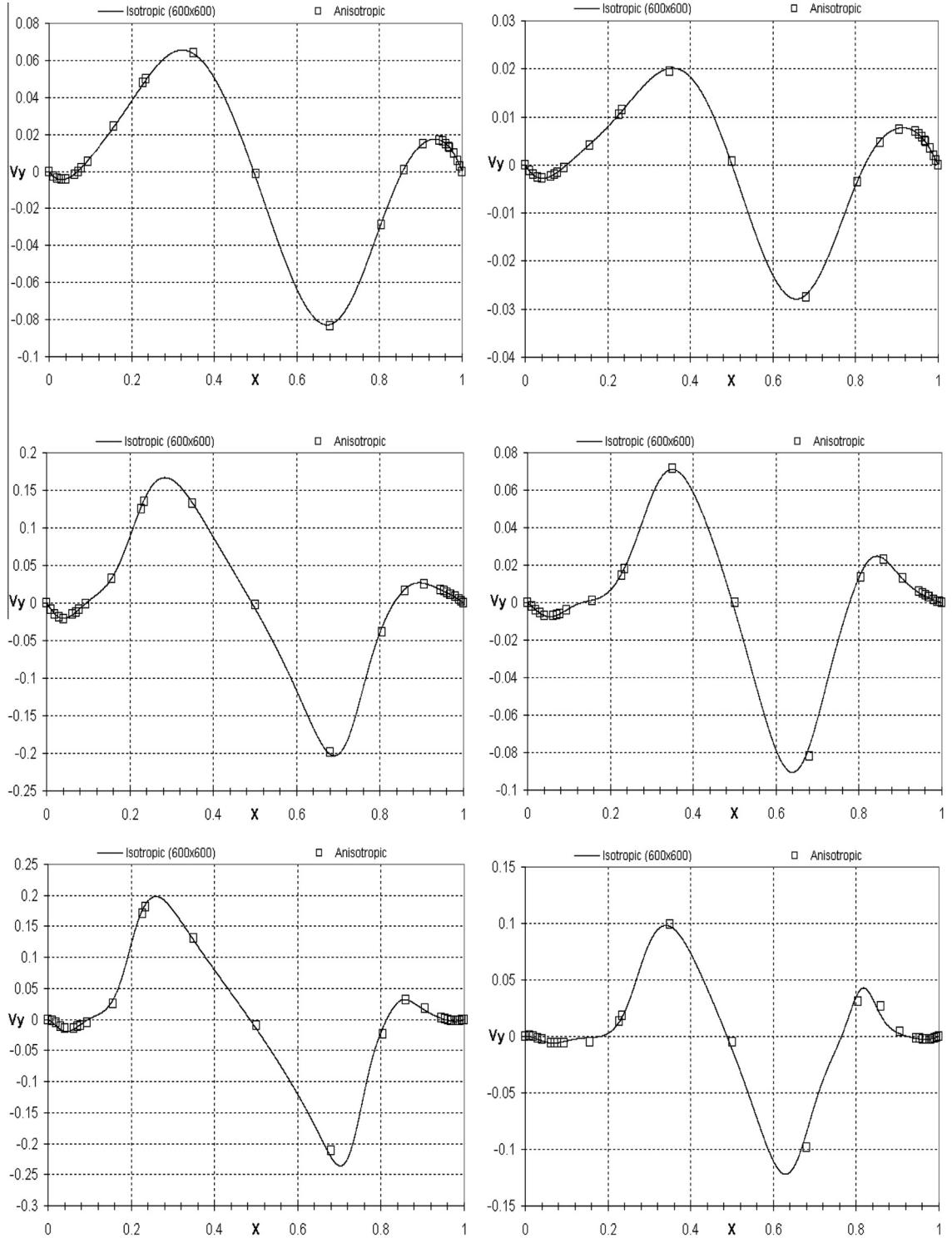


Fig. 9. Comparison of velocity profiles for $Re = 1000$ (top), for $Re = 5000$ (middle) and for $Re = 10,000$ (bottom). Left: Velocity profiles for V_y along $y = 0.1$. Right: Velocity profiles for V_y along $y = 0.05$.

$$\begin{aligned} \mathcal{R}_M &= \mathbf{f} - \rho \partial_t \mathbf{v}_h - \rho (\mathbf{v}_h + \mathbf{v}') \cdot \nabla \mathbf{v}_h - \nabla \mathbf{p}_h \\ \mathcal{R}_C &= -\nabla \cdot \mathbf{v}_h \end{aligned} \quad (12)$$

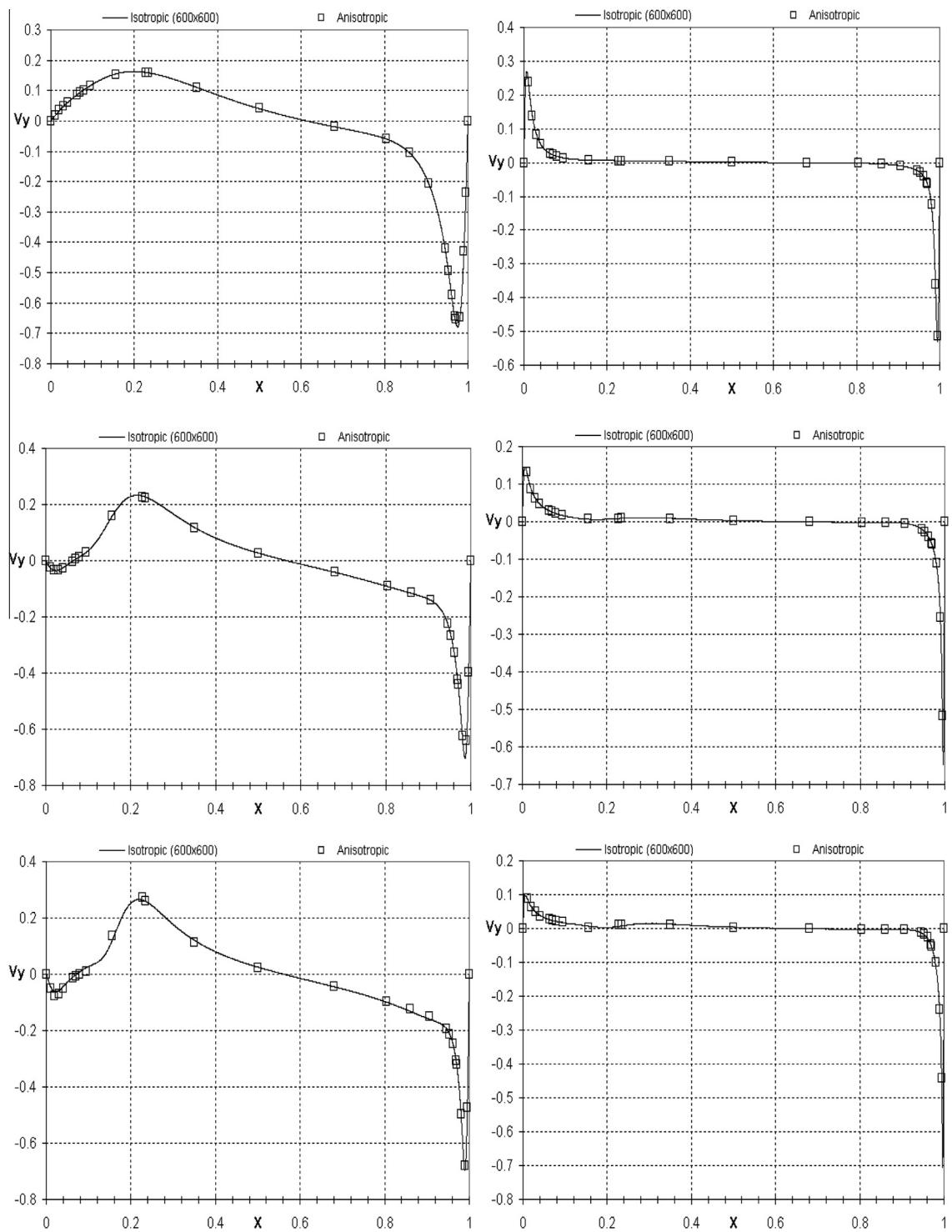


Fig. 10. Comparison of velocity profiles for $Re = 1000$ (top), for $Re = 5000$ (middle) and for $Re = 10,000$ (bottom). Left: Velocity profiles for V_y along $y = 0.9$. Right: Velocity profiles for V_y along $y = 0.99$.

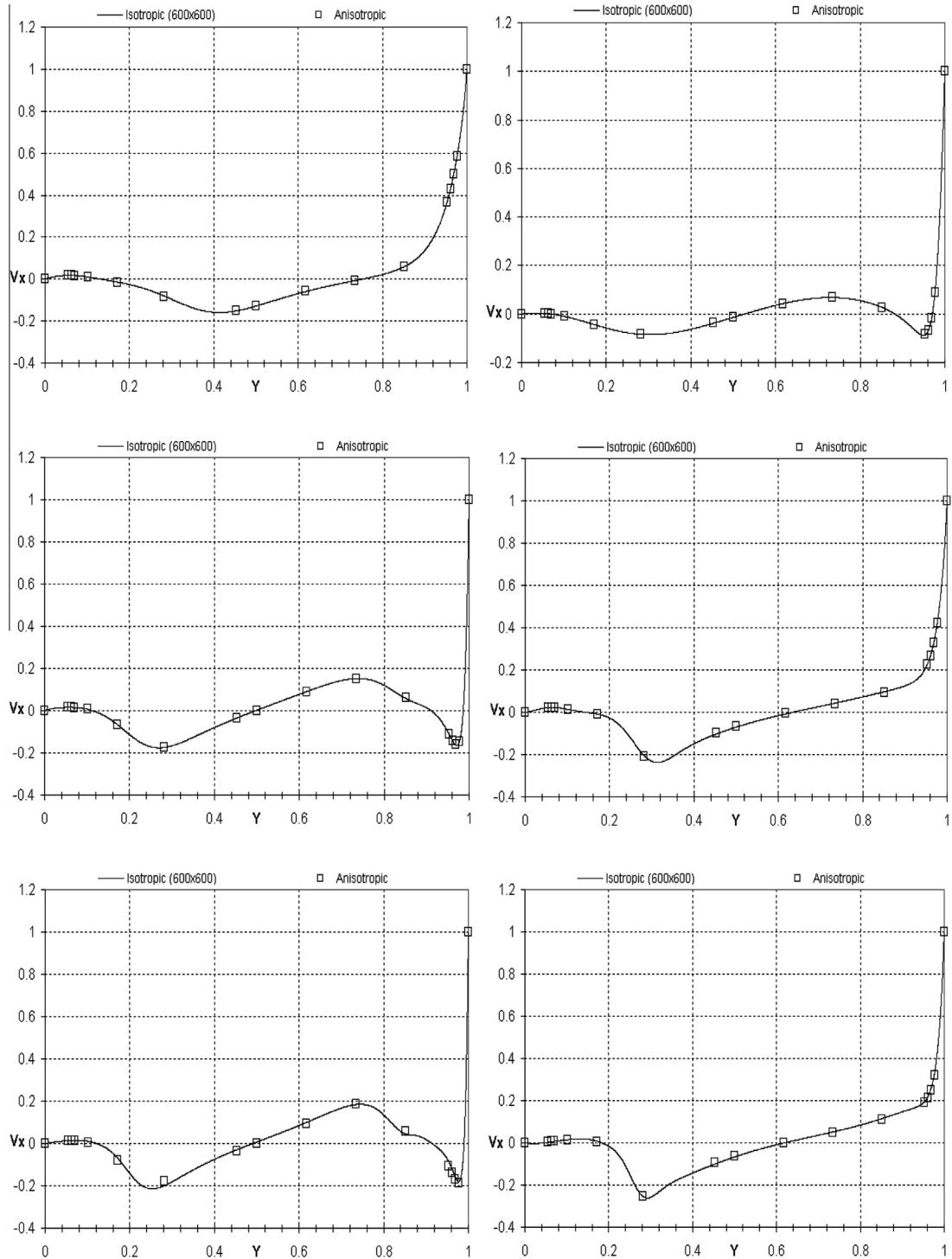


Fig. 11. Comparison of velocity profiles for $Re = 1000$ (top), for $Re = 5000$ (middle) and for $Re = 10,000$ (bottom). Left: Velocity profiles for V_x along $x = 0.1$. Right: Velocity profiles for V_x along $x = 0.9$.

When compared with the standard Galerkin method, the proposed stable formulation involves additional integrals that are evaluated element-wise. These additional terms, obtained by replacing the approximated \mathbf{v}' and p' into the large-scale equation, represent the effects of the sub-grid scales and they are introduced in a consistent way to the Galerkin formulation. All of these terms are multiplied by stabilizing parameters and enable to overcome the instability of the classical formulation arising in convection dominated flows and to deal with the pressure instabilities.

In this work, we adopt the following definitions for the stabilizing parameters τ_K and τ_C as proposed in [16] and in [17] respectively:

$$\tau_K = \left(\left(\frac{2\rho}{\Delta t} \right)^2 + \left(\frac{2\rho \|\mathbf{v}_h\|_K}{h_K} \right)^2 + \left(\frac{4\mu}{h_K^2} \right)^2 \right)^{-1/2}, \quad \tau_C = \left(\left(\frac{\mu}{\rho} \right)^2 + \left(\frac{c_2 \|\mathbf{v}_h\|_K}{c_1 h_K} \right)^2 \right)^{1/2} \quad (13)$$

where c_1 and c_2 are two constants independent from h_K [17] and h_K being the characteristic length of the element.

Note that the calculation of h_K is one of the main subjects of this paper. Recall that the stability coefficients weight the extra terms added to the weak formulation (11) and they are defined for each element K of the triangulation (13). Typically, these coefficients 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 [18], while others always use the triangle diameter (see [19] for details).

Nevertheless, 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 [20,17]. In [21] the authors examine deeply the effect of different element lenght definitions on distorted meshes. In [22] 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 adopted the definition proposed in [16] to compute h_K as the diameter of K in the direction of the velocity u (see Fig. 4):

$$h_K = \frac{2 |\mathbf{v}_h|}{\sum_{i=1}^{N_K} |\mathbf{v}_h \cdot \nabla \varphi_i|} \quad (14)$$

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

Note also that the use of the time step in the definition of the stabilizing parameter (13) is another important issue, in particular for small time steps. Several approaches can be found in the literature, in particular the use of dynamic subscales in [15] and the definition of the stabilization parameters computed from element matrices and vectors in [16]. These approaches have been used successfully and in both cases, the steady solution is independent of the time step and are stable for small time steps. In the following section, we show that we did not encounter these problems in the numerical results. Using adaptive mesh refinement, the first term in the definition of the stabilizing parameter (13) becomes small compared to the other terms. We also varied the time steps dynamically (dynamic adaptive meshing) to ensure that the stationary solution is stable and thus the contribution of the time steps in the stabilization parameter become negligible. However, the details of the stabilizing parameters are still very important and critical for successful use of stabilizing methods. Therefore, further research will focus on upgrading the existing code to take into account dynamic subscales from one hand and adaptive time stepping on the other hand in order to address mainly non stationary problem and turbulent flows.

5. Numerical examples

The performance of the method will be assessed by benchmarking the driven cavity at low and high Reynolds numbers and by comparing to very accurate reference solutions. In such simulations we show that boundary layers as well as vortices can be well captured by the mesh. Results are compared with the literature and show that the flow solvers based on stabilized finite element method are able to exhibit very good accuracy properties on anisotropic meshes with highly stretched elements. Additionally, we provide new measurements positioned near boundary layers as shown in Fig. 5. Such a new set of

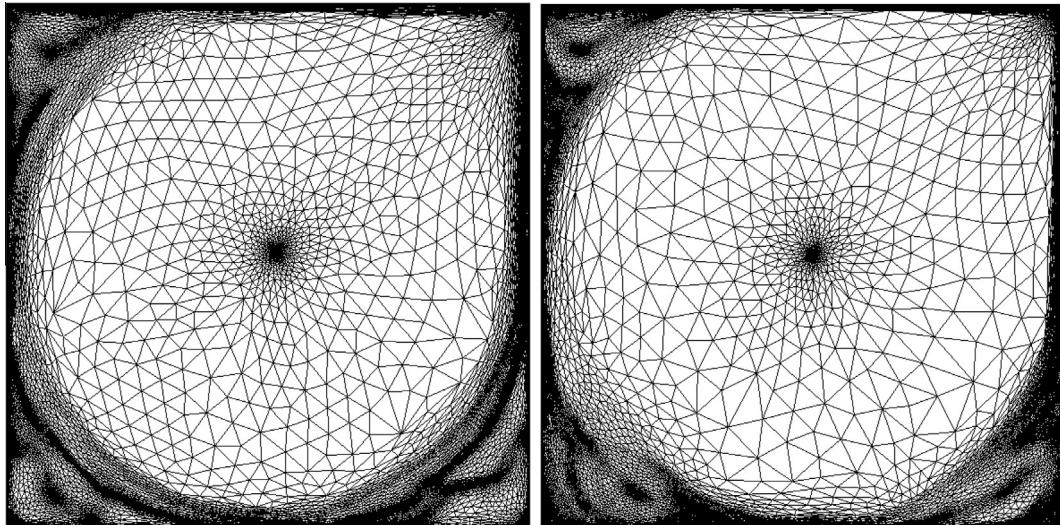


Fig. 12. Snapshots of the anisotropic meshes for Reynolds 2000 and 100,000.

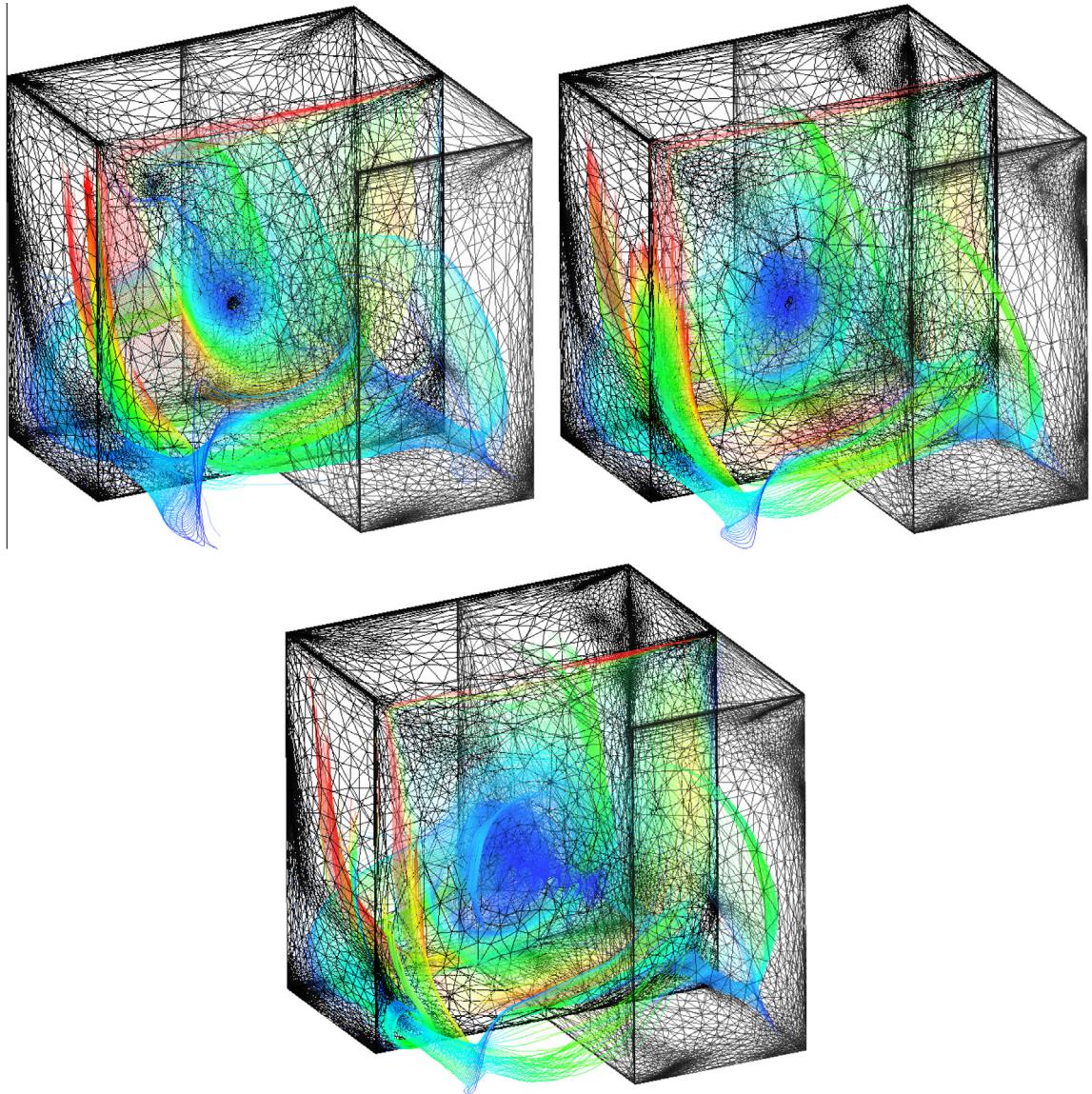


Fig. 13. Streamlines snapshots at Reynolds 1000, 3200 and 5000.

data can serve as a useful benchmark for comparisons of the results of the time dependent two and three dimensional incompressible Navier–Stokes equations with anisotropic mesh adaptation.

5.1. Driven flow cavity problem (2-D)

We begin to numerically solve the classical 2-D lid-driven flow problem. This test has been widely used as a benchmark for numerical methods and has been analyzed by a number of authors: [23,24] using finite difference and finite volume methods, and [25,26,11] using stabilized finite element method.

Homogeneous Dirichlet boundary conditions for the velocity are imposed equal to one on the upper boundary of the computational domain, i.e. at $y = 1$, and zero elsewhere on. The source term is identical to zero. The time step is fixed to 0.05. The density is set to one and the viscosity is adjusted in order to obtain Reynolds number of 1000, 5000, 10,000, 20,000, and 100,000.

An important point to mention before we start is that the following mesh adaptations are always performed for a fixed number of nodes. This turns out to be an important advantage because it avoids a drastic increase in the number of unknowns and in the computational cost. In other words, it provides a very useful tool in particular for industrial applications. Therefore, for Reynolds number of 1000, 5000 and 10,000 we fixed the number of nodes A to be equal to 10,000. The adaptive process always starts from the uniform mesh presented in Fig. 5 and is iterated every five time step.

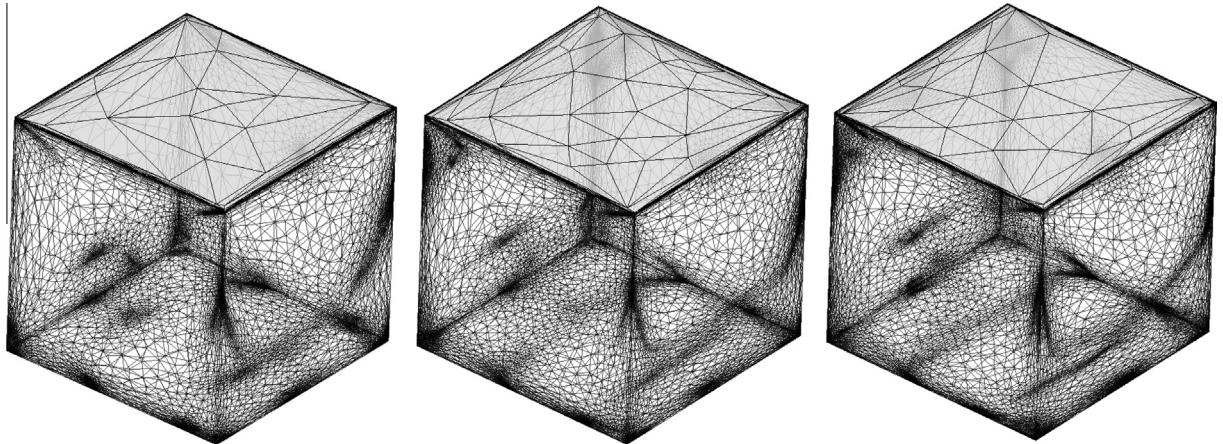


Fig. 14. 3D Anisotropic meshes at Reynolds 1000, 3200 and 5000.

The results on the three respective converged meshes ($\sim 10,000$ nodes) can be seen in Fig. 6. The exact number of nodes is 9771. Note the concentration of the resolution not only along all the boundary layers but also at the detachment regions. This reflects well the anisotropy of the solution caused by the discontinuity of the boundary conditions and the nature of the flow. The elements at the central bulk of the cavity region around the primary vortex are mostly isotropic and increase in size as the Reynolds number increases. Again, this reflects and explains how, for a controlled number of nodes, the mesh is naturally and automatically coarsened in that region with the goal of reducing the mesh size around the secondary vortices in the bottom corners.

As Reynolds increases, the capturing of different secondary vortices is very good. This is mainly due to the choice of the unit vector combining the velocity components and its norm. The proposed approach shows that it is capable of capturing all the vortices at all scales. Indeed, the small voricies are not intense enough to be captured accurately by classical adaptive strategies.

The zoom on the right side of the cavity in Fig. 7 highlights how sharply the layers can be captured. It shows the correct orientation and deformation of the mesh elements (longest edges parallel to the boundary). This yields a great reduction of the number of triangles. These results give confidence that the extension of the approach to take into account all the velocity components seems to work very well and plays an important role for transient flows.

The objective of these simulations is to compare now the newly obtained results to very accurate reference solutions. Therefore, we first plot the velocity profiles for V_x along $x = 0.5$ and for V_y along $y = 0.5$ at Reynolds 1000, 5000 and 10,000. Fig. 8 shows respectively that all the results are in very good agreement with a reference solution computed on a $600 \times 600 = 360,000$ points fixed mesh [11].

For more comparisons, we present in Table 1 and 2 values of the vertical and horizontal components of the velocity along $x = 0.5$ and $y = 0.5$, respectively. We report our results taken out from simulations made first on a fine isotropic mesh (600×600) and second from the converged anisotropic meshes ($\sim 10,000$ nodes). We compare these results to very precise computations found in [27,28] made on a uniform grid composed of 5000×5000 nodes. As shown in Table 1 and 2, a very good agreement is pointed between the present solution on isotropic 600×600 unstructured meshes and the references. The agreement is very good up to the fourth digit. However, also results from the anisotropic meshes compare well and agree up to the second digit with the highly accurate solutions computed on $5000 \times 5000 = 25,000,000$ points [28]. Again, results obtained by the VMS flow solver using either isotropic or, and in particular, anisotropic meshes compete very well to the literature.

It is worth to mention that there is a lack in the literature of highly resolved data at high Reynolds number. In particular, measurements taken near the walls does not exist yet in the literature. Thus, it makes the comparisons and validation of the code on anisotropic meshes at high Reynolds number difficult. Therefore, we will use in what follows our results carried out on the isotropic meshes (600×600) as references.

The newly introduced additional sensors near boundary layers are presented in Table 2. These sensors placed close to the walls are crucial for validating the efficiency in well capturing the boundary layers. Furthermore, we present the new plots taken near the corners in Figs. 9–11. We preview that the rich set of data presented herein could constitute a very interesting benchmark for purposes of validating the accuracy of the adaptive anisotropic meshing for incompressible Navier–Stokes near boundary layers.

We show in these figures the velocity profiles for V_y along $y = 0.1, 0.05, 0.9$ and 0.99 and for V_x along $x = 0.1$ and 0.9 at Reynolds 1000, 5000 and 10,000. All the results are in very good agreement with the proposed reference having 600×600 elements. It is clear that in regions of intense vortices and very close to the walls, the use of anisotropic meshes plays a critical role. All the boundary layers are sharply captured and automatically identified. We can clearly see from these plots that the creation of extremely stretched elements near the walls provides accurate results. The reference computed on a very fine

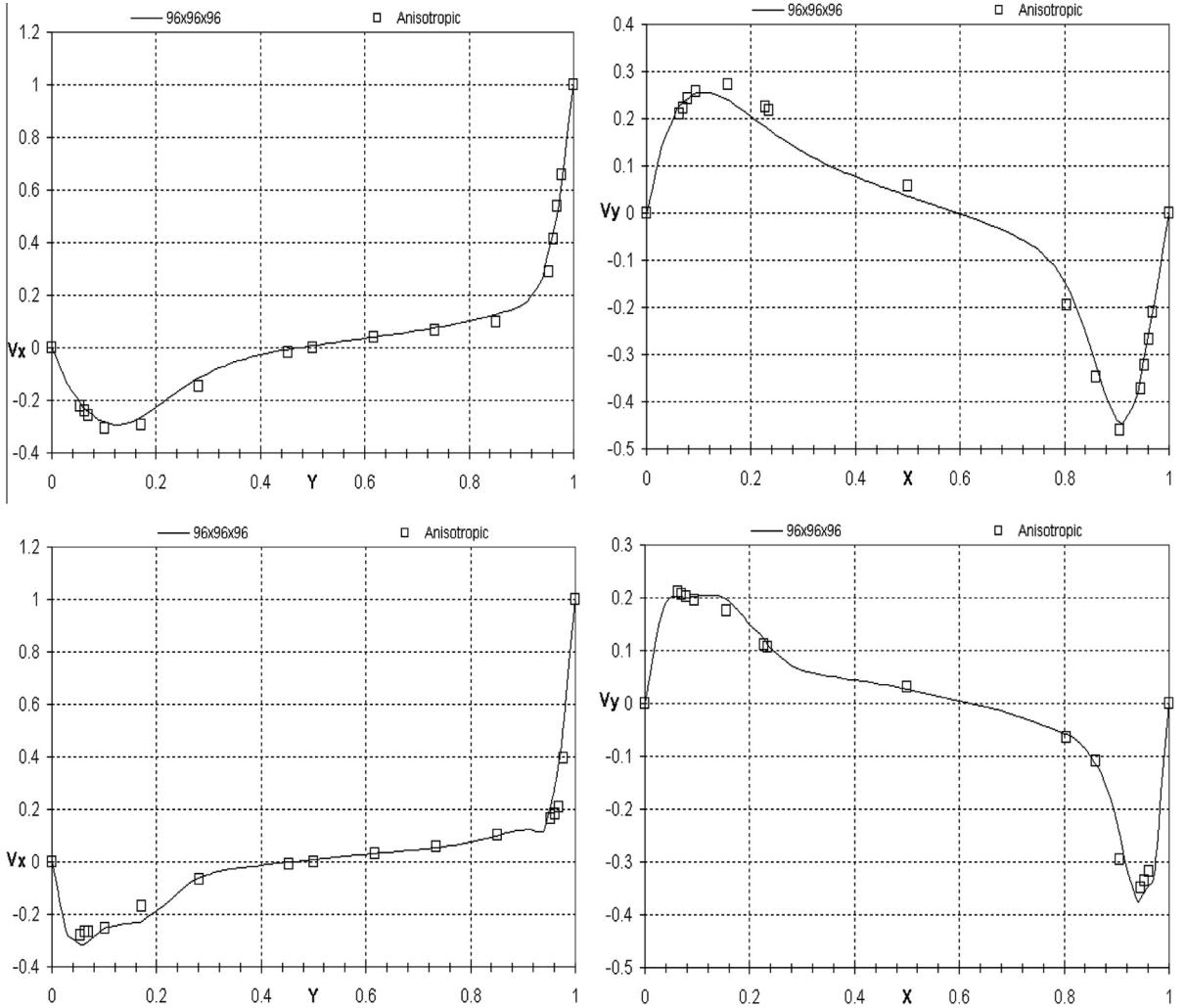


Fig. 15. Comparison of velocity profiles in the mid-planes for $Re = 1000$ (top) and for $Re = 3200$ (bottom). Left: Velocity profiles for V_x along $x = 0.5$. Right: Velocity profiles for V_y along $y = 0.5$.

isotropic mesh (600×600) agree with the anisotropic results using only 10,000 nodes up to the second digit and differ very slightly in the third digit at some points. We provide in Tables 3–9 in the Appendix the evaluation of all the velocity profiles at different benchmark locations and for different Reynolds number. Such a new set of data can serve as a useful benchmark for comparisons of the results.

Finally, in order to assess the capability of the VMS method to simulate high Reynolds number flows on anisotropic meshes, we repeated the simulation for Reynolds numbers of 20,000 and 100,000. Note that details on obtaining the steady solution for different Reynolds numbers is analyzed by the authors in [11] and will not be repeated here.

We increased the number of nodes to 20,000 in order to capture more accurately the characteristic of the unsteady solution. We present only in Fig. 12 snapshots of the meshes at a certain time step. As expected, all the main directional features characterizing the velocity inside the lid-driven cavity are detected and well captured by the anisotropic error estimator. The mesh elements are highly stretched along the direction of the layers, at the detachments regions and around all the newly appeared secondary vortices. Once again, the developed incompressible Navier–Stokes VMS solver shows to be very efficient and robust at high Reynolds numbers using highly stretched elements. Again, for a given constraint on the number of elements we can find the mesh that maximizes the accuracy of the numerical solution.

5.2. Driven flow cavity problem (3-D)

We continue to numerically solve the classical 3-D lid-driven flow problem. Same as in the 2-D case, Dirichlet boundary conditions are imposed on the velocity. At the upper wall at $z = 1$ it is equal to one, and zero elsewhere. The source term is

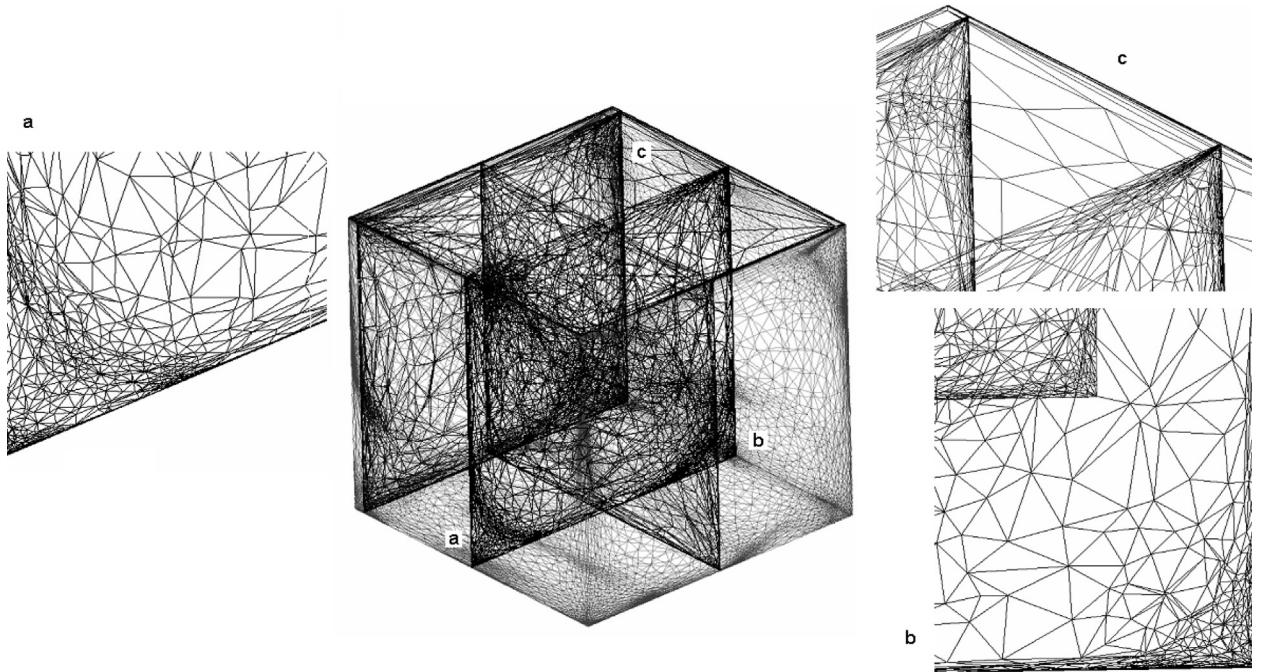


Fig. 16. Zooms on the mesh details inside the 3D cavity.

identical to zero. The time step is fixed to 0.05 s. The viscosity is adjusted in order to obtain Reynolds number of 1000, 3200 and 5000.

All numerical experiments are done with a fixed number of elements ($\sim 450,000$). Again, in 3-D where the flow characteristics are much more complicated, all the boundary layers as well as the vortices are sharply captured and identified.

Fig. 13 highlights the stretching of the elements near the walls, in the corners and even at the bifurcations of the vortices for Reynolds equal 1000, 3200 and 5000 respectively. The higher the Reynolds number is, the denser the mesh around these regions is as shown in Fig. 14. The velocity profiles for V_y along $y = 0.5$ and for V_x along $x = 0.5$ at Reynolds 1000 and 3200 are presented in Fig. 15. All the results are in good agreement with the reference having a twice finer mesh of $96 \times 96 \times 96$ grid points [29].

In Fig. 16 we present different zoomed captures on the mesh to show how the elements can be highly stretched along the direction of the layers. This reflects well the accuracy and details of the resolution caused by the discontinuity of the boundary conditions and the nature of the flow.

The anisotropic adaptive procedure modifies the mesh in a way that the local mesh resolutions become adequate in all directions. Recall again that these plots reflect for the given fixed number of elements ($\sim 450,000$) the mesh that maximizes the accuracy of the numerical solution. More analysis taking into account an increased number of elements and higher Reynolds number will be the subject of further investigations. The presented test cases are considered here in the objective of demonstrating the capability of the VMS method to simulate 2D and 3D high Reynolds number flows on anisotropic meshes.

6. Conclusions

We have showed in this paper that anisotropic adapted meshes with highly stretched elements can be used to compute high Reynolds number flows. We presented an a posteriori estimate based on the length distribution tensor approach and the associated edge based error analysis. We proposed an important extension of the anisotropic theory for computing the error along the edges for a vectorial quantity. It was addressed for the incompressible Navier–Stokes equations. All the meshes are obtained under the constraint of a fixed number of edges in the mesh. The Variational MultiScale (VMS) method is shown to be efficient for simulating flows at high Reynolds number. We demonstrated the efficiency of the anisotropic adaptive procedure to be capable of automatically producing boundary layer meshes inside the 2D and 3D lid-driven cavities. The numerical results show that the flow solvers based on stabilized finite element method is able to exhibit good stability and accuracy properties on anisotropic meshes. All the main directional features characterizing the velocity inside the cavities are detected and well captured by the anisotropic error estimator. The mesh elements are highly stretched along the direction of the layers, at the detachment regions and around all the vortices. 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.

Appendix A

See Tables 3-9.

Table 3

Velocity profiles along $y = 0.1$ at $Re = 1000, 5000$ and $10,000$.

Table 4

Velocity profiles along $y = 0.05$ at $Re = 1000, 5000$ and $10,000$.

Table 5

Velocity profiles along $y = 0.9$ at $Re = 1000, 5000$ and $10,000$.

Table 6

Velocity profiles along $y = 0.99$ at $Re = 1000, 5000$ and $10,000$.

Table 7

Velocity profiles along $x = 0.1$ at $Re = 1000, 5000$ and $10,000$.

Table 8

Velocity profiles along $x = 0.5$ at $Re = 1000, 5000$ and $10,000$.

Table 9

Velocity profiles along $x = 0.9$ at $Re = 1000, 5000$ and $10,000$.

References

- [1] T. Coupez, Metric construction by length distribution tensor and edge based error for anisotropic adaptive meshing, *J. Comput. Phys.* 230 (2011) 2391–2405.
- [2] F. Alauzet, Adaptation de maillage anisotrope en trois dimension, Applications aux simulations instationnaires en Mécanique des Fluides, Ph.D. thesis, Université de Montpellier II, 2003.
- [3] P. Frey, F. Alauzet, Anisotropic mesh adaptation for CFD computations, *Comput. Methods Appl. Mech. Engrg.* 194 (2005) 5068–5082.
- [4] F. Alauzet, P. Frey, P.-L. George, B. Mohammadi, 3D transient fixed point mesh adaptation for time-dependent problems: application to CFD simulations, *J. Comput. Phys.* 222 (2007) 592–623.
- [5] C. Gruau, T. Coupez, 3D tetrahedral, unstructured and anisotropic mesh generation with adaptation to natural and multidomain metric, *Comput. Methods Appl. Mech. Engrg.* 194 (2005) 4951–4976.
- [6] T. Coupez, A mesh improvement method for 3D automatic remeshing, *Numerical Grid Generation in Computational Fluid Dynamics and Related Fields*, Pineridge Press, 1994. 615–626.
- [7] X. Li, M.S. Shephard, M.W. Beall, 3D anisotropic mesh adaptation by mesh modification, *Comput. Methods Appl. Mech. Engrg.* 194 (2005) 4915–4950.
- [8] M.S. Shephard, J.E. Flaherty, K.E. Jansen, X. Li, X. Luo, N. Chevaugeon, J.F. Remacle, M.W. Beall, R.M. Obara, Adaptive mesh generation for curved-domains, *Appl. Numer. Math.* 52 (2005) 251–271.
- [9] J.F. Remacle, X. Li, M.S. Shephard, J.E. Flaherty, Anisotropic adaptive simulation of transient flows using discontinuous Galerkin methods, *Int. J. Numer. Methods Engrg.* 62 (2005) 899–923.
- [10] L. Formaggia, S. Perotto, New anisotropic a priori error estimate, *Numer. Math.* 89 (2001) 641–667.
- [11] E. Hachem, B. Rivaux, T. Kloczko, H. Digonnet, T. Coupez, Stabilized finite element method for incompressible flows with high Reynolds number, *J. Comput. Phys.* 229 (2010) 8643–8665.
- [12] T.J.R. Hughes, G.R. Feijoo, L. Mazzei, J.N. Quincy, The variational multiscale method – A paradigm for computational mechanics, *Comput. Methods Appl. Mech. Engrg.* 166 (1998) 3–24.
- [13] F. Brezzi, L. Franca, T. Hughes, A. Russo, $b = f g$, *Comput. Methods Appl. Mech. Engrg.* 145 (1997) 329–339.
- [14] T. Dubois, F. Jauberteau, R. Temam, *Dynamic Multilevel Methods and the Numerical Simulation of Turbulence*, Cambridge University Press, Cambridge, 1999.
- [15] R. Codina, J. Principe, Dynamic subscales in the finite element approximation of thermally coupled incompressible flows, *Int. J. Numer. Methods Fluids* 54 (2007) 707–730.
- [16] T.E. Tezduyar, Y. Osawa, Finite element stabilization parameters computed from element matrices and vectors, *Comput. Methods Appl. Mech. Engrg.* 190 (3–4) (2000) 411–430.
- [17] R. Codina, Stabilization of incompressibility and convection through orthogonal sub-scales in finite element methods, *Comput. Methods Appl. Mech. Engrg.* 190 (13–14) (2000) 1579–1599.
- [18] S. Mittal, On the performance of high aspect ratio elements for incompressible flows, *Comput. Methods Appl. Mech. Engrg.* 188 (2000) 269–287.
- [19] 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 J. Numer. Anal.* 41 (2004) 1131–1162.
- [20] I. Harari, T.J.R. Hughes, What are C and h?: inequalities for the analysis and design of finite element methods, *Comput. Methods Appl. Mech. Engrg.* 97 (1992) 157–192.
- [21] C. Förster, W.A. Wall, E. Ramm, Stabilized finite element formulation for incompressible flow on distorted meshes, *Int. J. Numer. Methods Fluids* 60 (10) (2009) 1103–1126.
- [22] A. Cangiani, E. Süli, The residual-free-bubble finite element method on anisotropic partitions, *SIAM J. Numer. Anal.* 45 (4) (2007) 1654–1678.
- [23] U. Ghia, K.N. Ghia, C.T. Shin, High-Re solutions for incompressible flow using the Navier–Stokes equations and a multigrid method, *J. Comput. Phys.* 48 (3) (1982) 387–411.
- [24] M. Sahin, R.G. Owens, A novel fully-implicit finite volume method applied to the lid-driven cavity problem. Part I: high Reynolds number flow calculations, *Int. J. Numer. Methods Fluids* 42 (2003) 57–77.
- [25] T. Hughes, L. Franca, M. Balestra, A new finite element formulation for computational fluid dynamics: V. Circumventing the Babuska–Brezzi condition: a stable Petrov–Galerkin formulation of the Stokes problem accommodating equal-order interpolations, *Comput. Methods Appl. Mech. Engrg.* 59 (1986) 85–99.
- [26] L. Franca, S. Frey, Stabilized finite element methods: II. The incompressible Navier–Stokes equations, *Comput. Methods Appl. Mech. Engrg.* 99 (1992) 209–233.
- [27] O. Botella, R. Peyret, Benchmark spectral results on the lid-driven cavity flow, *Comput. Fluids* 27 (1998) 421–433.
- [28] J.L. Guermond, P.D. Minev, Start-up flow in a three-dimensional lid-driven cavity by means of a massively parallel direction splitting algorithm, *International Journal for Numerical Methods in Fluids*, 2011. <http://dx.doi.org/10.1002/fld.2583> (online).
- [29] S. Albensoeder, H. Kuhlmann, Accurate three-dimensional lid-driven cavity flow, *J. Comput. Phys.* 206 (2005) 536–558.