Adaptive mesh finite-volume calculation of 2D lid-cavity corner vortices

João P.P. Magalhães, Duarte M.S. Albuquerque *, José M.C. Pereira, José C.F. Pereira

Universidade Técnica de Lisboa, Instituto Superior Técnico, Mechanical Engineering Department, LASEF, Av. Rovisco Pais, 1049-001 Lisbon, Portugal

ARTICLE INFO

ABSTRACT

An anisotropic refinement criterion suitable for Finite-Volume methods is presented and Navier–Stokes solutions are reported for the lid-driven cavity flow configuration at $Re = 1000$ with adaptive anisotropic meshes (*h*-refinement). The *a posteriori* error estimation criterion is based on the assessment of the goodness-of-fit of the least squares regression used to perform the variables profile reconstruction and it is capable of detecting both large-scale and small-scale flow phenomena. The criterion allowed to capture in detail the large-scale flow structure and also the sequence of creeping flow small sharp corner's [1] eddies, up to the fourth corner vortex in addition to the primary cavity vortex. The smallest corner vortex detected is $O(10^{-16})$ smaller in velocity magnitude compared with the cavity primary recirculation flow.

© 2013 Elsevier Inc. All rights reserved.

1. Introduction

There are a number of adaptive methods in Computational Fluid Mechanics to reduce and control the numerical error, either by modification of the mesh, *h*-refinement, or by changing the order of approximation of the numerical scheme, *p*-refinement. These methods require an appropriate error measure to detect the numerical error. There are several error estimation techniques and most of them have been widely used in the context of Finite-Element methods, see e.g. [2–6].

Despite the recent growth of adaptive grid refinement methods for Finite-Volume (FV) calculations, few investigations have been concerned with the chosen *a posteriori* error estimate, also known as refinement criterion. Usually, the refinement criterion is based on the analysis of the absolute or relative importance of the gradient and the Hessian matrix, or in some specific *ad hoc* heuristic measure of detection of important flow features, see [7]. Both approaches have the inconvenient of requiring some empirical manual tuning that are problem dependent, see [8]. Other methods involve the full integration of the governing equations with a low and high order method, see [9]. If the (normalized) difference between the two solutions is greater than a certain threshold then the corresponding cells are marked for refinement. This method involves the full integration of the governing equations under higher order requirements.

An alternative method consists in the estimation of the Taylor series truncation error (TSTE), but if we use this value directly, the threshold value will be dependent on the magnitude of the variables' values, thus requiring user input and knowledge. If we nondimensionalize it with the local resolved terms, the criterion will be overly sensible to variable values near zero, and it will also remain unbounded. Methods based on the estimate of the error of the variables' second moment have been proposed by Jasak and Gosman [10], but it tends to underperform in fine meshes. Moreover, with the exception of the

* Corresponding author.
E-mail addresses: jpmagalhaes@ist.utl.pt (J.P.P. Magalhães), duartealbuquerque@ist.utl.pt (D.M.S. Albuquerque), jose.chaves@ist.utl.pt (J.M.C. Pereira), jcfpereira@ist.utl.pt (J.C.F. Pereira).

TSTE, none of these methods have the ability to provide directional information, limiting the possibility of anisotropic refinement.

We use an anisotropic refinement criterion to reduce the outlined problems. The method is based on performing regression quality diagnostics on the dependent variables' local profiles, since this criterion directly tests the profile assumed by the FV numerical method by considering this as a problem of model building. The method potentially allows to use a suite of powerful model building and regression diagnostic tools. This method is possible to use simultaneously for several dependent variables and combinations, and most importantly, it has the ability to detect both small-scale and large-scale flow features present in the flow. Moreover, it is limited, normalized, while being easy to implement, inexpensive to compute and easy to calibrate.

This paper focuses on a FV implementation of the proposed anisotropic refinement criterion in an adaptive method for two dimensional incompressible flows. The application example is the standard lid-driven cavity flow for Reynolds number $Re = 1000$, see e.g. [11–16], for which a number of well documented benchmark solutions are available (e.g. [17–19]). Ham et al. [20] have considered the same case for $Re = 400$ for cartesian grid method with anisotropic adaptation driven by an error estimator based on the second derivative of the fluid velocities and mesh size.

The lid-driven cavity flow on the vicinity of the lower corners displays multiple eddies in creeping flow regime. Moffatt [1] has shown that under the assumption of 2D Stokes flow, the sharp corner flows formed by two intersection boundaries, where either the velocity or the tangential stress vanishes on each boundary with a critical angle opening, contain a sequence of vortices descending into the corner.

Corner eddies are very small compared to the cavity scale and tend to have little impact on the nature of the bulk flow. However, their description is relevant to understand the flow topology in increasingly slender cavities and the influence on the limiting cases of the aspect cavity ratio, approaching zero or infinite, see [21]. Other theoretical implications are related with 3D corner flows, see [22,23].

The numerical resolution of corner flows has been pursued and constitutes a challenging task in terms of numerical resolution, efficiency, robustness, adaptive mesh refinement and refinement criteria. One of the first attempts, and maybe the most impressive, was conducted by Gustafson and Leben [24] that computed the Stokes flow solution in an uniform grid in the whole domain and then projected it on finer local grids near the corner. A sequence of domains, in a zoom like fashion, allowed to predict up to 21 corner vortices with the local maximum stream function intensity of 10^{-92}. The lack of global interaction prevents their solution to be a benchmark test case for Navier–Stokes solvers. Several other authors combine the solutions of the Navier–Stokes equations with exponential mesh refinement of the cavity corners flow regions and asymptotic of the flow near corners to predict the series of vortices, see [25].

The main objective of this work is to present an adaptive criterion useful for problems with disparate length scales when adaptive meshing is often indispensable for resolving small flow features. To the Authors' knowledge, this work represents the first implementation of a grid refinement criterion for viscous flows within a FV framework that automatically captured multiple eddies, up to the fifth, in the vicinity of each of the lid-driven cavity bottom sharp corners.

The paper is organized as follows: the next section describes the FV method used to solve the flow in hybrid unstructured grids with cells of arbitrary topology. This is followed by the presentation of the refinement criterion and the application of the lid-driven cavity flow at $Re = 1000$. The paper ends with summary conclusions.

2. Mathematical and numerical model

The isothermal flow of an incompressible fluid is governed by the mass and momentum conservation laws, being expressed by the incompressibility constraint and the Navier–Stokes equations:

$$\nabla \cdot \mathbf{u} = 0, \tag{1}$$

$$\frac{\partial \mathbf{u}}{\partial t} + \nabla \cdot (\mathbf{u} \otimes \mathbf{u}) = \nabla \cdot \left(\nu \nabla \mathbf{u} + \nu \nabla^T \mathbf{u} \right) - \nabla p, \tag{2}$$

where $\mathbf{u}$ is the velocity vector, ν is the kinematic viscosity and p is the pressure density ratio. A fractional-step numerical method is used to numerically solve Eqs. (1) and (2). The method starts by computing from a previously existing velocity field $\mathbf{u}^n$ an initial approximation $\mathbf{u}^*$ of the new velocity field. This approximation satisfies the momentum equations without the pressure gradient contribution:

$$\frac{\mathbf{u}^* - \mathbf{u}^n}{\Delta t} = -\nabla \cdot (\mathbf{u}^n \otimes \mathbf{u}^n) + \nabla \cdot \left(\nu \nabla \mathbf{u}^n + \nu \nabla^T \mathbf{u}^n \right). \tag{3}$$

$\mathbf{u}^*$ is then projected onto the space of solenoidal fields, by removing its irrotational component:

$$\frac{\mathbf{u}^{n+1} - \mathbf{u}^*}{\Delta t} = -\nabla p^{n+1}. \tag{4}$$

Taking the divergence of Eq. (4) and setting $\nabla \cdot \mathbf{u}^{n+1} = 0$ results in the Poisson equation for the pressure field:

$$\nabla^2 p^{n+1} = \frac{\nabla \cdot \mathbf{u}^*}{\Delta t}. \tag{5}$$

2.1. Spatial discretization

The present numerical method works on the premise that the profile of the dependent variable within each cell is linear. Considering a generic variable ϕ, for cell P this is written as:

$$\phi(\mathbf{x}; P) = \phi_P + \nabla\phi_P \cdot (\mathbf{x} - \mathbf{P}), \tag{6}$$

where $\mathbf{P}$ is the cell center vector and ϕ_P and $\nabla\phi_P$ stand respectively for the value at the cell center and the gradient, which is assumed constant within the cell. This gradient is obtained by least-squares using the values of other cells in the vicinity of P.

To accomodate the multiple faces occurring in interfaces between refined and non-refined cells, an unstructured grid of arbitrary topology is used. Each cell P is a closed convex polyhedron, delimited by a boundary ∂P which consists in a set of faces $\mathcal{F}(P)$. The number of faces of P is arbitrary and variable within a grid, and is the cardinality of $\mathcal{F}(P)$. A face f is a convex plane polygon with arbitrary orientation in space, delimited by the consecutive line segments or edges connecting a set of vertices. Again, the number of vertices in each face is arbitrary and variable within the grid. Finally, a vertex is a point in three-dimensional space and is fully defined by its physical coordinates.

Connectivities in the grid are established by faces. Each face establishes in its entirety the boundary between two adjacent cells. As a consequence, the surface intersection of any two different cells must be, at most, a single face, in which case the cells are adjacent, and separated by that same face. Additionally, when face f_k belongs to both cells P_i and P_j, it cannot belong to any other cell than P_i or P_j.

The computational storage points are located at the geometrical center of each cell. This corresponds to the so-called "collocated" or "non-staggered" arrangement; the method devised by Mirtich [26] is used to compute the cell's centroid coordinates. Fig. 1 provides a sketch of the storage scheme and grid topology. To provide pressure–velocity coupling, the convective fluxes at each face are also stored; this issue is of great importance to the overall stability of the numerical method and is further developed below.

The usage of the divergence theorem for the governing equations in cell P yields:

$$\frac{1}{\Delta t}\int_P \mathbf{u}^* - \mathbf{u}^n \, dV = -\sum_{f\in\mathcal{F}(P)}\int_f \mathbf{u}^n \otimes \mathbf{u}^n \cdot \mathbf{dS} + \sum_{f\in\mathcal{F}(P)}\int_f \left(\nu\nabla\mathbf{u}^* + \nu\nabla^T\mathbf{u}^*\right) \cdot \mathbf{dS}, \tag{7}$$

$$\sum_{f\in\mathcal{F}(P)}\int_f \nabla p^{n+1} \cdot \mathbf{dS} = \frac{1}{\Delta t}\sum_{f\in\mathcal{F}(P)}\int_f \rho\mathbf{u}^* \cdot \mathbf{dS}. \tag{8}$$

The volume and surface integrals are evaluated using the Simpson rule, where the middle-point value of the integrand is multiplied by the integral domain measure. This yields second-order accuracy provided that the middle-point value is directly available. This is true for volume integrals since by definition the middle point is the cell center, but surface integrals require interpolation from the cell centers to the face centers. This interpolation must also be of second-order if accuracy is to be maintained. This becomes a problem in the interfaces between refined and unrefined cells, particularly for the diffusive term. Fig. 2 shows an example of such interface.

The convective term requires the value of ϕ in the center of face f. Two point schemes (using ϕ_{P_0} and ϕ_{P_1}) such as the simple arithmetic average (AVG) or distance-weighted linear interpolation (CDS or TRI) fail to incorporate variations of ϕ in the directional orthogonal to the line from P_0 to P_1. This error may be significant in refinement interfaces, and because
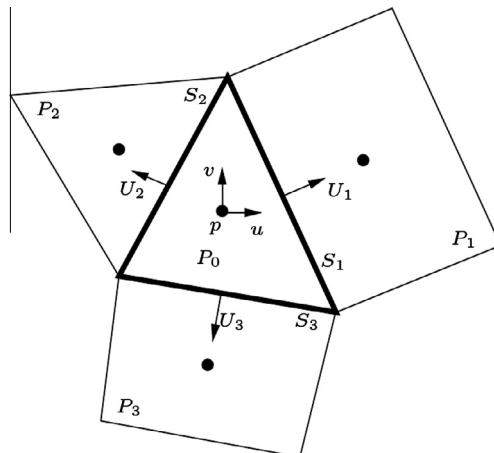

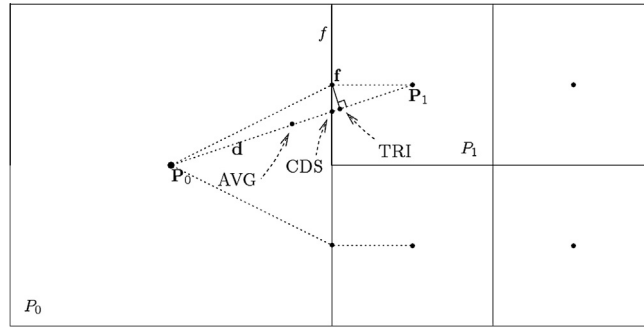**Fig. 1.** General 2D polyhedral control volume.

Fig. 2. Interface between cells of different refinement levels.

of this, we use the reconstructed cell profiles on each side of the face. On each side of f the reconstructed profile is used to compute the value of ϕ_f. These values are then conciled with an arithmetic average, this yields:

$$\phi_f = \frac{1}{2}\left\{(\phi_{P_0} + \nabla\phi_{P_0} \cdot (\mathbf{f} - \mathbf{P_0})) + (\phi_{P_1} + \nabla\phi_{P_1} \cdot (\mathbf{f} - \mathbf{P_1}))\right\}. \tag{9}$$

For the diffusive fluxes, the approach from Jasak and Gosman [10] was used. A central differences scheme (CDS) is used as the basis of the scheme, to which is added a contribution from the cell-centered gradients strictly proportional to the degree of non-orthogonality:

$$\widehat{\nabla\phi_f} = (\phi_{P_1} - \phi_{P_0})\Delta + \nabla\phi_f - (\mathbf{d}_f \cdot \nabla\phi_f)\Delta. \tag{10}$$

where $\mathbf{d}_f = \mathbf{P}_1 - \mathbf{P}_0$ is the distance vector between the centroids of P_0 and P_1, $\mathbf{S}_f$ is the normal of face f (with magnitude equal to its area), and $\Delta = \mathbf{S}_f/(\mathbf{S}_f \cdot \mathbf{d}_f)$ is a vector in the direction of $\mathbf{d}_f$ such that $\mathbf{S}_f$ is decomposed into Δ and another vector tangential to f. The value $\nabla\phi_f$ is computed as the average of the cell-centered gradients on both sides of the face:

$$\nabla\phi_f = \frac{\nabla\phi_{P_0} + \nabla\phi_{P_1}}{2}, \tag{11}$$

In Eq. (10), the first term on the right-hand side is the CDS, which acts only in the direction of $\mathbf{d}_f$; then the second and third terms add the cell-centered gradient contributions, from which the components along $\mathbf{d}_f$ are removed. Fig. 3 shows the vector decomposition of the orthogonal and non-orthogonal components.

The diffusive terms computed with recourse to (10), result in the presence of the sum of integrated face gradients:

$$\sum_{f\in\mathcal{F}(P)} \widehat{\nabla\phi_f} \cdot \mathbf{S}_f. \tag{12}$$

The diffusion scheme in Eq. (10), must be discretized implicitly so that it becomes fully available in the coefficient matrix, in order to avoid multiple correction steps. This procedure implies that for each face f the least squares regression to obtain both $\nabla\phi_{P_0}$ and $\nabla\phi_{P_1}$ must be embedded in the matrix. This is not easy to do and is further explained in appendix A.

To compute the reconstructed gradient $\nabla\phi_P$ of the dependent variables ϕ, we perform a least squares regression using the values of ϕ in the neighbor cells of the vicinity of P, see Muzaferija and Gosman [8]. These neighbors are obtained by visiting each face in the cell and querying the cell on the other side of the face. For each cell, we use the second order linear profile, see Kobayashi et al. [27] for details. The gradient components, g, are obtained by solving the linear system $\mathbf{Ag} = \mathbf{h}$, which is defined by:

$$a_{mn} = \sum_{f\in\mathcal{F}(P)} w_f d_{f,m} d_{f,n}, \tag{13}$$

$$h_m = \sum_{f\in\mathcal{F}(P)} \left(\phi_{P_k} - \phi_P\right) w_f d_{f,m}, \tag{14}$$
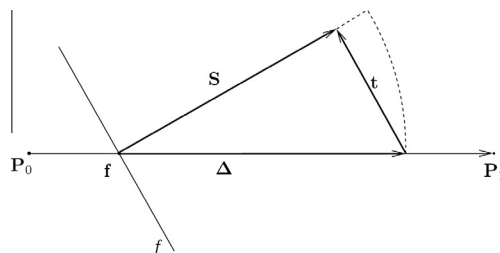

Fig. 3. Decomposition of orthogonal and non-orthogonal components for the diffusion scheme.

where m and n denote the Cartesian components and w_k is the weight of cell P_k, obtained as the inverse of the square of the magnitude of $\mathbf{d}$.

2.2. Temporal discretization

Because in this case we are only interested in the steady state solution of equations (1) and (2), a false-transient scheme is used, where the (steady) system is advanced in time as if it was unsteady. Since the time derivative is then not important, it could be discretized using the first order explicit Euler scheme. However, the small stability envelope of this scheme poses growing convergence problems as the refinement level increases. For this reason, the convective terms were discretized with the second-order Runge–Kutta scheme, whereas the diffusive terms were discretized with the Crank–Nicolson scheme. Being implicit, the Crank–Nicolson term has the added convenience of avoiding the need for the use of the diffusive stability criterion, which arises when explicit methods are used for the diffusive terms, which becomes especially stringent as the grid resolution increases, since it depends on the inverse square of the grid spacing.

2.3. Pressure–velocity coupling

The pressure–velocity coupling algorithm and the compatibility between the different numerical schemes affect the overall accuracy, stability and energy conservation proprieties of the results, see [28] for details. It is well known that the colocated grid arrangement for the calculation of incompressible fluid flow may induce unphysical oscillatory pressure profiles.

In this work, we use the procedure proposed by Mahesh et al. [29]. The Poisson equation for pressure has the discretized form:

$$\sum_{f\in\mathcal{F}(P)} \nabla p_f^{n+1} \cdot \mathbf{S}_f = \sum_{f\in\mathcal{F}(P)} \mathbf{u}_f^* \cdot \mathbf{S}_f, \tag{15}$$

(note that capital P stands for the cell, whereas small-case p stands for the pressure field). The pressure field obtained by solving this equation is such that by adding its gradient to the face velocity $U_f = \mathbf{u}_f \cdot \mathbf{S}_f$, mass conservation is attained. Recall that U_f is stored, per Fig. 1. That is, if U_f^{n+1} is given by

$$U_f^{n+1} = U_f^* - \Delta t \nabla p^{n+1} \cdot \mathbf{S}_f, \tag{16}$$

then usage of U_f^{n+1} as the transport velocity in the convective term ensures mass conservation:

$$\sum_{f\in\mathcal{F}(P)} \int_f \mathbf{u}^{n+1}\phi \cdot \mathbf{dS} = \sum_{f\in\mathcal{F}(P)} U_f^{n+1}\phi_f. \tag{17}$$

This is true up to the precision used to solve the system of algebraic equations resulting from the discretization of the pressure Poisson equation.

However, the cell center velocities still need to be corrected but, importantly, using the cell center pressure gradient. This is equivalent to:

$$\mathbf{u}_P^{n+1} = \mathbf{u}_P^* - \Delta t \nabla p_P^{n+1}. \tag{18}$$

The problem is that ∇p_P^{n+1} is not the same as ∇p_f^{n+1}. Exact continuity is satisfied only by virtue of ∇p_f^{n+1}. Because of this ∇p_P^{n+1} needs to be as close as possible to ∇p_f^{n+1} for $f \in \mathcal{F}(P_0)$. The objective should be such that by interpolating ∇p_P^{n+1} to the several faces of the cell, the overall difference should be as small as possible. Mahesh et al. [29] and Benhamadouche et al. [30] have proposed a least squares minimization, so that the above requirement is satisfied in a least squares sense:

$$\frac{\partial}{\partial(\nabla p_P)} \sum_{f\in\mathcal{F}(P)} \left(\nabla p_P \cdot \mathbf{S}_f - \nabla p_f \cdot \mathbf{S}_f\right)^2 \|\mathbf{S}_f\| = 0. \tag{19}$$

This scheme is indeed more robust from a stability point of view, although it does require more computational work.

During grid refinement, the coarse grids are saved in memory so this information can be used for a Multigrid method to solve the Poisson equation from the pressure–velocity coupling and improve the convergence speed. The implementation of this Multigrid method fully resembles the one described in Trotteberg et al. [31], using a V-cycle. As a restriction operator, the summation of the residuals from the descendants fine cells is used, the linear interpolation with the cell gradient from Thompson and Ferziger [9] is used as a prolongation operator and the solver (smother) employed in this method is the Gauss–Seidel with successive over-relaxation.

3. Refinement criterion and procedure

The error to be minimized by the adaptive process is the local truncation error. Since there is no possible way to accurately evaluate this error, we can observe the final error, defined as the difference between the computed and the exact profile. The average total error of cell P for a generic function Φ is given by

$$\epsilon_P = \frac{1}{V_P} \int_P |\Phi(\mathbf{x}) - \phi(\mathbf{x};P)| \, dV, \tag{20}$$

where $\phi(x;P)$ is the resolved profile. The discretization error is a parcel of this error; other parcels include nonlinear error sources, modeling errors and others, consequently, the discretization error is an approximation of the total error.

It should be noted that due to the nonlinear nature of the governing equations of fluid flow, it does not follow that the cells to be refined are those with the largest error because the error may be a symptom rather than a cause of inaccuracy. In other words, it is possible that cells with low error level become responsible for the appearance of high errors in other regions of the flow (downstream or even upstream, depending on the nature of the flow).

3.1. Taylor-series truncation error

The Taylor series expansion theorem states that any smooth function $\phi \in C^\infty$ in the vicinity of a point P can be approximated as a sum of its derivatives at P. In multidimensional form:

$$\phi(x_i; x_{i,P}) = \phi_P + \left(\frac{\partial \phi}{\partial x_i}\right)_P (x - x_{i,P}) + \frac{1}{2}\left(\frac{\partial^2 \phi}{\partial x_i \partial x_j}\right)_P (x_i - x_{i,P})(x_j - x_{j,P}) + \cdots, \tag{21}$$

Einstein summation is implied above. Since the present FV method implementation uses a linear profile, the truncation error is dominated by the second-order term. Using the definition of the average error:

$$\varepsilon_P = \frac{1}{2V_P} \int_P \left(\frac{\partial^2 \phi}{\partial x_i \partial x_j}\right)_P (x_i - x_{i,P})(x_j - x_{j,P}) dV, \tag{22}$$

and assuming that the Hessian matrix is constant inside the cell P, the integration (22) results in:

$$\varepsilon_P = \frac{1}{2V_P} \left|\left(\frac{\partial^2 \phi}{\partial x_i \partial x_j}\right)_P\right| (M_{ij})_P, \tag{23}$$

where $(M_{ij})_P$ is the inertia tensor of the cell P, computed using the methodology of Mirtich [26]. The error can be nondimensionalized by using the lower order terms T_l from the Taylor series:

$$T_l = \phi_P + l_i^P \left|\left(\frac{\partial \phi}{\partial x_i}\right)_P\right|, \tag{24}$$

where l_i^P is the length of cell P in the direction i. The isotropic version of the TSTE criteria is thus:

$$c_P = \frac{\left|\left(\frac{\partial^2 \phi}{\partial x_i \partial x_j}\right)_P\right| (M_{ij})_P}{2V_P T_l}. \tag{25}$$

whereas the anisotropic version of c_P is obtained by retaining only the relevant Hessian terms:

$$c_{P,i} = \delta_{ij} \frac{\left|\left(\frac{\partial^2 \phi}{\partial x_j \partial x_k}\right)_P\right| (M_{jk})_P}{2V_P T_l}. \tag{26}$$

3.2. Coefficient of multiple determination

Since the numerical method uses a least squares regression to reconstruct the dependent variable profile for each cell, it is natural to determine the goodness of fit of the regression. For clarity, we will use in this section the standard notation for regression analysis. With this notation, the profile in Eq. (6) is written as:

$$Y = \beta_0 + \sum_{k=1}^n \beta_k X_k, \tag{27}$$

where Y is a variable explained by a linear combination of n explaining variables X_k (in our case the coordinate moments). The parameters β are the ones which we want to compute, and consist in the intercept and in spatial derivatives of ϕ. For example, $\beta_0 = \phi_P$, $\beta_1 = (\partial \phi/\partial y)_P$, $\beta_2 = (\partial \phi/\partial x)_P$ and so on. As can be seen, the profile can be easily expanded to accommodate higher-order derivatives.

Considering all the observations, an overdetermined system of equations is formed, given by $\mathbf{y} = \mathbf{X}\mathbf{b} + \mathbf{e}$, where $\mathbf{y}$ is a vector with dimension $m \times 1$ containing the observations of Y, $\mathbf{X}$ is a matrix $m \times (n + 1)$ containing the values of the several explaining variables for each observation, $\mathbf{b}$ is a vector with dimension $n + 1$ with the parameters to be determined and $\mathbf{e}$ is the vector with the estimation residual for each of the observations.

The estimation residual is defined as the difference between the observed $\mathbf{y}$ and the estimated value $\hat{\mathbf{y}} = \mathbf{X}\hat{\mathbf{b}}$:

$$e_i = Y_i - \widehat{Y}_i. \tag{28}$$

The estimate $\hat{\mathbf{b}}$ of $\mathbf{b}$ obtained by least squares is the one that minimizes the error sum of squares (with zero mean error implied):

$$\min_{\hat{\mathbf{b}}} \sum_{i=0}^{m} e_i^2 \quad \Longleftrightarrow \quad \min_{\hat{\mathbf{b}}} \mathbf{e}^T \mathbf{e}. \tag{29}$$

The coefficient of multiple determination of the regression, R^2, expresses the extent to which the variance in the original data is explained by the variance of the predicted data. This is obtained by relating the sum of squares for the original data, SS_y and the sum of squares for the fitted data from the regression, SS_r. R^2 is then given by:

$$R^2 = \frac{SS_r}{SS_y}. \tag{30}$$

The quantity R^2 is limited between 0 and 1. Low values indicate a poor correlation and high values a good correlation. Since we want the criterion to behave the opposite way, the criterion is formed by the one's complement of R^2:

$$c = 1 - R^2. \tag{31}$$

R^2 always increases as the number of explaining variables n increases, even if it introduces no improvement in the regression. To account for this, the adjusted multiple determination coefficient (adjusted for the number of variables) can be used:

$$R_{adj}^2 = 1 - \frac{m-1}{m-n}(1 - R^2). \tag{32}$$

An alternative to the adjusted R^2, which penalizes more the loss in degrees of freedom when the number of regressors increases is called the alternative R^2:

$$R_{alt}^2 = 1 - \frac{m+n}{m-n}(1 - R^2). \tag{33}$$

The first comparisons of these three coefficients quickly revealed that the behavior of the criterion obtained by use of the adjusted and alternative varieties were less smooth, which created non-smooth grids, with less quality. This is explained by the fact that the number of neighbors used in the moving least squares approach is not high, and the correction in the number of degrees of freedom becomes important. For that reason, we only use the unmodified R^2.

3.3. Least-squares fit and anisotropy

The refinement criteria extracted from R^2 is isotropic because it contains information from all spatial directions. For an anisotropic version, these diagnostics are not useful since they refer to the basic profile without particular spatial direction. Other aggregate diagnostics, including detection diagnostics in parameter space, do not distinguish the spatial directions as well. To obtain this information, we pose this problem as a specification problem, which tries to identify the best set of explaining variables. This is done by comparing the diagnostics of different models. Namely we consider the model of the base linear profile and compare it with the model of a cubic profile which contains high order terms only in a given direction. For example, the R^2 criterion for the x direction is given by:

$$c_x = R^2(\phi_x(\mathbf{x}, P_0)) - R^2(\phi(\mathbf{x}, P_0)), \tag{34}$$

where $\phi(\mathbf{x}, P_0)$ is the base linear profile and $\phi_x(\mathbf{x}, P_0)$ is a profile which contains high order terms, but only those where the powers of x dominate:

$$\phi_x(x, y, z; P_0) = \overbrace{\beta_0 + \beta_1 x + \beta_2 y + \beta_3 z}^{\text{linear profile}} + \overbrace{\beta_4 x^2 + \beta_5 xy + \beta_6 xz + \beta_7 x^3 + \beta_8 x^2 y + \beta_9 x^2 z}^{\text{high order } x \text{ profile}}. \tag{35}$$

If the difference in R^2 (dR^2) of these two profiles is significant, then we have reasons to suspect that the inclusion of the high order terms in the x direction would improve the data fit. This highlights that the profile in the x direction is not linear, and induces the conclusion that cell P_0 does not possess sufficient resolution in the x direction and consequently should be refined in that direction. If this is performed for the remaining space directions, an anisotropic criterion is obtained.

3.4. Criterion filtering

Since any *a posteriori* refinement criterion is, by definition, applied to coarse grids, the results can be stiff and rapidly varying. This in turn has the effect of making the cell distribution also stiff, which in extreme situations causes numerical instability and ultimately divergence. This situation is avoided by employing the following concomitant procedures:

- The criterion is smoothed with a truncated arithmetic kernel;
- The selection of cells to be refined is expanded by a buffer layer (of one cell);
- The refinement directions are cross-checked in the vicinity, so that conflicting preferential directions are preemptively ruled out;
- As anisotropic cells become progressively stretched, it is made more difficult to refine them in counter directions, so that the degree of non-orthogonality does not become severe.

This is a multi-objective procedure and required some fine-tuning, so that the conflicting requirements of less cells vs. quality grids are both respected to some satisfactory degree. Of particular note is the impact that the last procedure has in the limit of high resolution: the anisotropy of the grid is diminished.

4. Results

4.1. Verification of the method for uniform mesh

The flow under consideration is the lid-driven square cavity flow. The boundary conditions consist of the no-slip condition in all boundaries, but the top boundary moves at a given constant velocity U_{lid}. The Reynolds number is given by $Re = \frac{U_{lid}L}{\nu}$ where L is the cavity length and was set to 1000.

The first results were obtained with an uniform mesh of $160 \times 160 = 25600$ hexahedrons using Central Differences scheme (CDS) for convection. Fig. 4 shows that the results agree very well with the benchmark data of the Finite-Difference predictions of Ghia et al. [17] or the spectral solutions by Botella and Peyret [18].

4.2. Comparison between the two error estimators

To simulate the differences between isotropic and anisotropic refinement a relative small number of cells was used. For the adaptive mesh calculations we start with a grid of 50×50 cells; after convergence, the criterion is evaluated and cells are marked for refinement. Using the TSTE criterion with Gauss estimation for the Hessian matrix, the grid was refined two times for each criterion, for cells with criterion values higher than 0.05. Fig. 5(a) shows the isotropically refined mesh, which contains 5300 cells and Fig. 5(b) shows the anisotropic mesh, comprising 3600 cells.

The accumulation of refined levels on the cavity's corners is considerable and allows a close resolution of the second vortex and its separation and attachment points. Fig. 6 compares the result after convergence on the present refinement level for both cases. The differences are not very significant, although the isotropic grid is in general closer to the benchmark results. The anisotropic grid contains less 33% cells than the isotropic mesh and thus cheaper by almost 40% (on account of the super-linear relation between cost and number of degrees of freedom). This argues for the efficiency of using anisotropic refinement as compared with isotropic refinement. In three-dimensional cases, in particular, for 3D the anisotropic adapted meshes may need much fewer cells than the isotropic one.

To study the dR^2 criteria capabilities an adaptive grid computation was done, it starts with a 10×10 grid and refines the cells with values higher than 0.1. After 4 refinement levels, we compare the results obtained for the isotropic and anisotropic criteria with the benchmark results of Botella and Peyret [18] which were obtained with an uniform mesh of 160×160, note
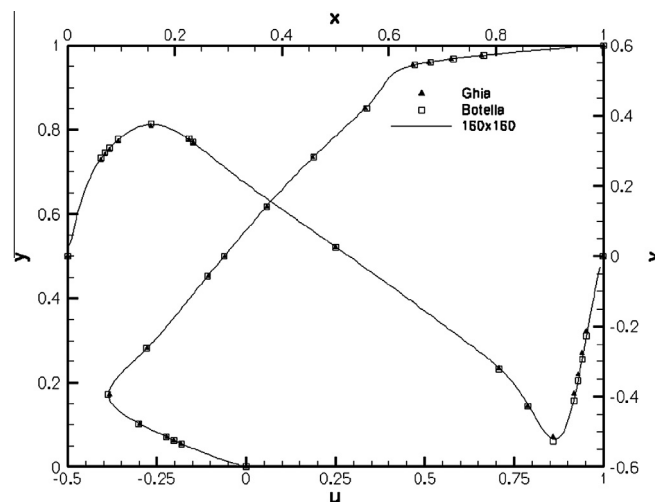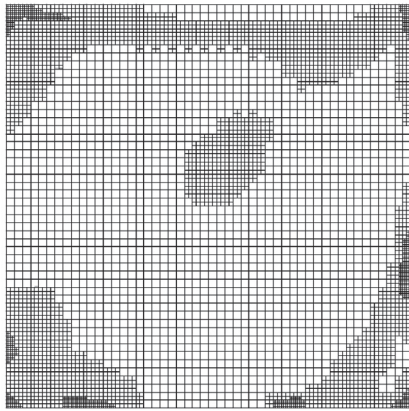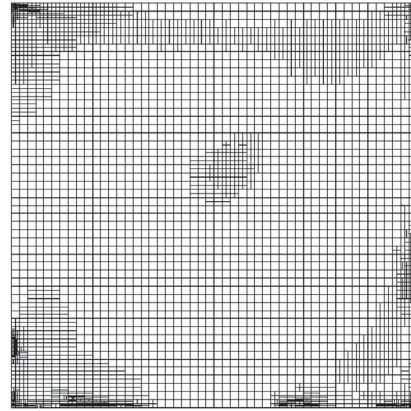

Fig. 4. Velocity profiles comparison with benchmark results.

(a) Lid-Driven cavity flow: Isotropically refined mesh: 5300 cells

(b) Lid-Driven cavity flow: Anisotropically refined mesh: 3600 cells

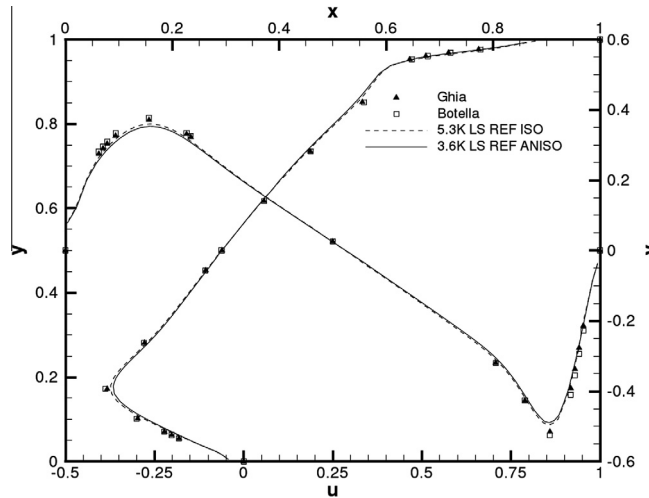**Fig. 5.** Lid-Driven cavity flow: Isotropic and Anisotropic mesh examples with TSTE.

Fig. 6. Velocity profiles comparison between benchmark values with isotropic and anisotropic adaptive refinement TSTE.

that the equivalent uniform mesh for the adaptive grids has the same grid size. Figs. 7(a) and (b) show the isotropic and anisotropic grid obtained, respectively.

The isotropic grid has 6910 cells and the anisotropic grid has 4407 cells. Fig. 8 compares the results obtained with both meshes and the benchmark solution. The differences between the curves of both grids are due to the anisotropic grid has lower quality than the isotropic one, although both results are close to the benchmark solution. The isotropic and anisotropic have approximately less 73% and 82.8% cells than the grid used by Botella and Peyret [18].

The dR^2 shows results closer to the benchmark data than the TSTE. The dR^2 is a more suitable criteria because it detects if the second and third order terms of the regression are not negligible when compared with the first order terms, while TSTE has low accuracy from the Hessian computation and does not work well for grids with a low number of cells.

4.3. Results for the adaptive mesh with lid-cavity corner vortices

The dR^2 criterion is sensitive to local maximums and minimums, because of the low correlation at these points. In flows such as the cavity flow, this is a useful feature. We have applied this criterion with a threshold of 0.01, starting from a grid of 10×10 and Fig. 9(a) shows the grid after 15 refinement levels. It is evident that large savings are achieved with the refinement. Fig. 9(b) shows the flow streamlines for $Re = 1000$. The grid is aligned with the flow, the boundary layers are well re-
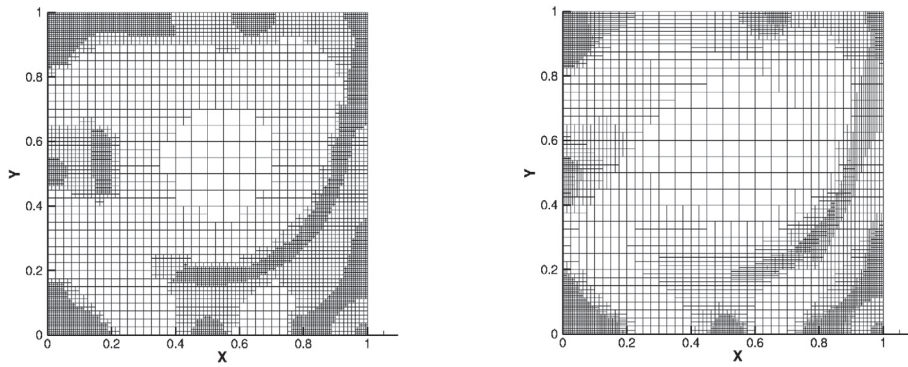
(a) Lid-Driven cavity flow: Isotropically refined mesh: 6910 cells

(b) Lid-Driven cavity flow: Anisotropically refined mesh: 4407 cells

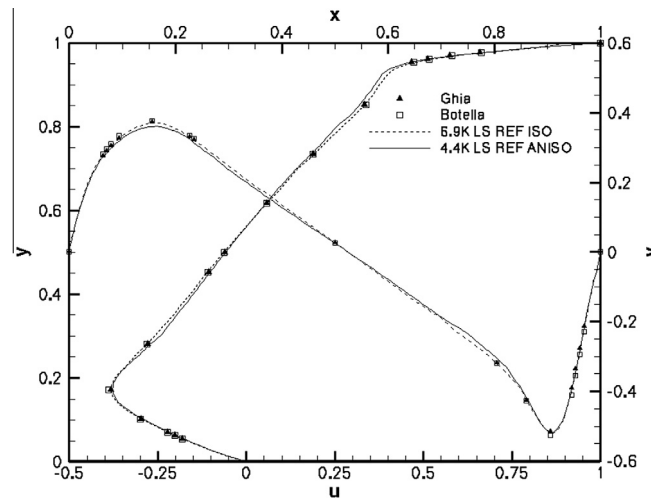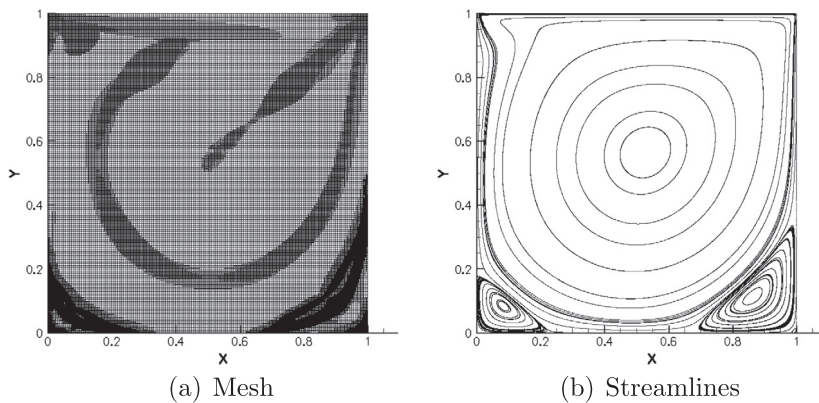**Fig. 7.** Lid-Driven cavity flow: Isotropic and Anisotropic mesh examples dR^2.

Fig. 8. Velocity profiles comparison between benchmark values with adaptive refinement dR^2.

(a) Mesh

(b) Streamlines

Fig. 9. Lid-Driven cavity flow: Final field and mesh after 15 refinement levels.
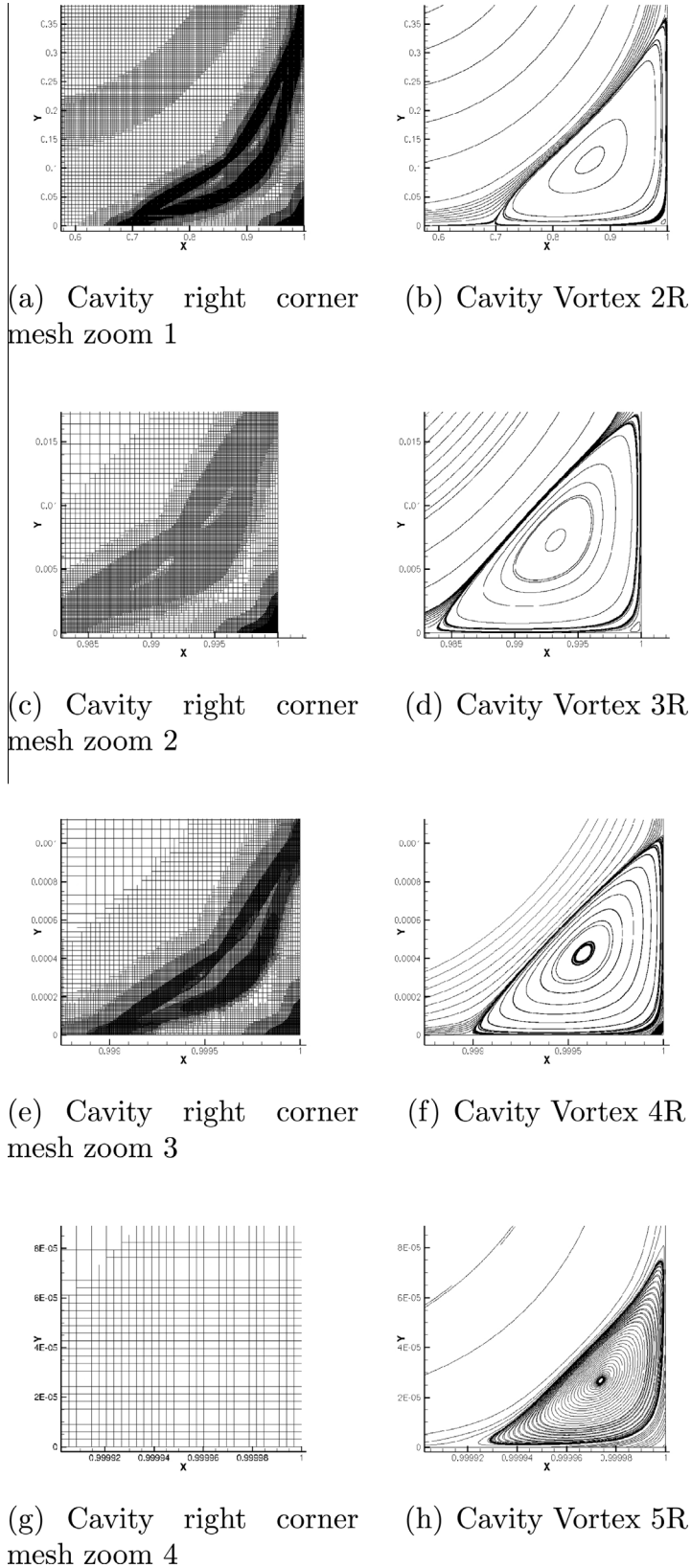
(a) Cavity right corner mesh zoom 1

(b) Cavity Vortex 2R

(c) Cavity right corner mesh zoom 2

(d) Cavity Vortex 3R

(e) Cavity right corner mesh zoom 3

(f) Cavity Vortex 4R

(g) Cavity right corner mesh zoom 4

(h) Cavity Vortex 5R

Fig. 10. Left corner vortices mesh and streamlines, sequence of vortices.

(a) Cavity left corner mesh zoom 1

(b) Cavity Vortex 2L

(c) Cavity left corner mesh zoom 2

(d) Cavity Vortex 3L

(e) Cavity left corner mesh zoom 3

(f) Cavity Vortex 4L

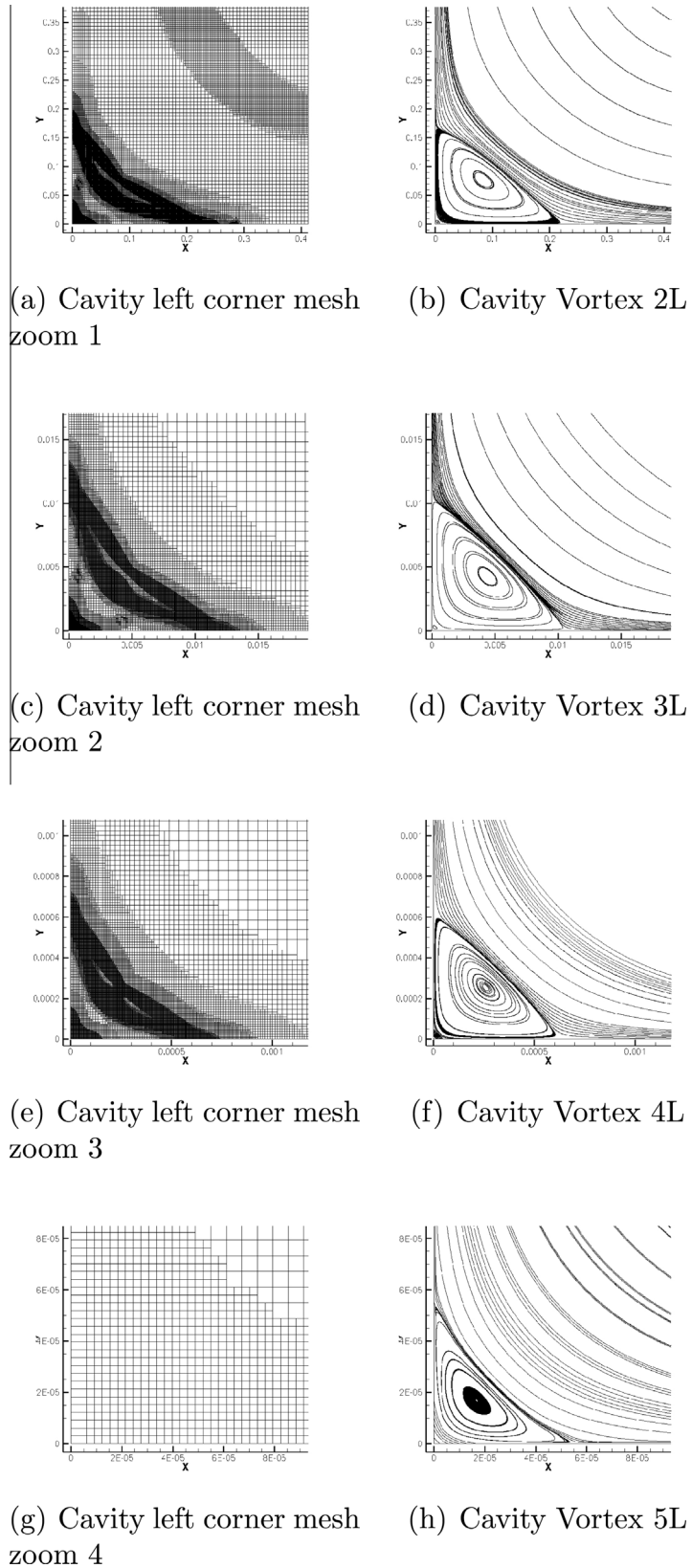(g) Cavity left corner mesh zoom 4

(h) Cavity Vortex 5L

Fig. 11. Left corner vortices mesh and streamlines, sequence of vortices.

Table 1
Cavity vortices parameters.

| Vortex	C_x, C_y	Vorticity	X,Y separation	Figure
1	0.53051,0.56531	−2.02727	−	9(b)
2R	0.86576,0.11364	0.822856	0.69495,0.36817	10(b)
3R	0.99323,0.00727	−0.00484733	0.98339,0.01759	10(d)
4R	0.99958,0.00042	3.50858E−5	0.99895,0.00107	10(f)
5R	0.99997,2.68E−5	−3.71581E−7	0.99991,8.25E−5	10(h)
2L	0.08317,0.07708	0.386846	0.22335,0.16800	11(b)
3L	0.00441,0.00424	−0.00198701	0.01068,0.01038	11(d)
4L	0.0025,0.0025	1.2416E−5	0.00065,0.00064	11(f)
5L	1.69E−5,1.69E−5	−1.42392E−7	5.530E−5,5.573E−5	11(h)
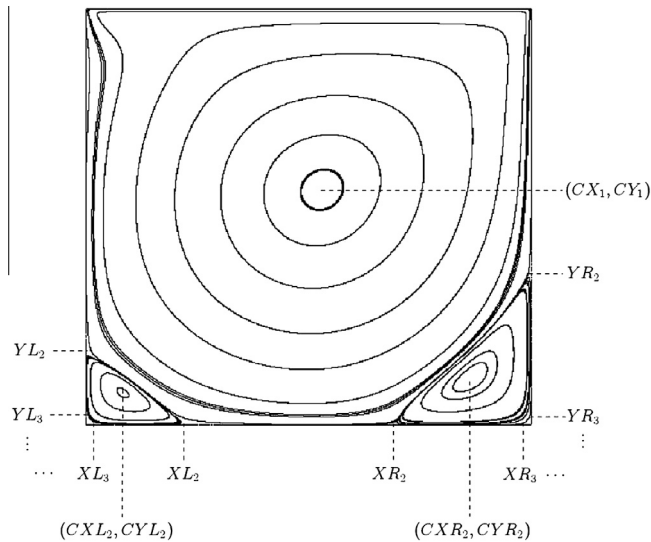
Fig. 12. Lid-Driven cavity: general layout, nomenclature.

solved, as well as the vortex core, the separation and attachment points. Out of the core of the secondary vortex, a belt of refined grid accompanies the inflexion of the streamlines, right where they have more pronounced curvature.

The adaptive computation of the lid-driven cavity flow allows the corner vortices to appear with successively refinement levels. This motivated progressively more thorough simulations to try to resolve relaying only on the refinement criterion as a way of detecting them. The dR^2 criterion is a logical choice for this application, as it is totally invariant in relation to scale.

For nomenclature purposes we will refer to the main cavity vortex as vortex number 1 and corner vortices number 2, 3, 4, 5, right or left vortices, *e.g.* vortex 3*R* denoting 3rd vortex on right corner.

Figs. 10(a)–(h) show 4 sequential zooms of the mesh and streamlines on the right corner vortices chain. The exponential decay in vortex size and strength can be qualitatively reduced from the figure zooms and obviously attention should be given to the coordinates values. Figs. 11(a)–(h) show 4 zooms of the mesh and streamlines on the left corner vortices chain.

Table 1 lists the vortices center locations, the attachment and separation points and also the vorticity magnitude. Fig. 12 explains the vortices nomenclature used.

The main cavity vortex is not exactly at the center of the cavity but slightly above this point, at $(0.530, 0.565)$, but the corner vortices are virtually centered at the cavity diagonals, see Table 1. The predicted vorticity decay is very strong with each consecutive vortex having less than one percent of the vorticity of the previous one. The vortices detachment length on the bottom and side walls decreases approximately with a contraction ratio of 5.0×10^{-2}.

Figs. 13 and 14 show the absolute velocity along the cavity diagonal (1,1) to (0,0) and diagonal (1,0) to (0,1) as a function of the distance along the diagonal, the *log* scale used can distinguish the eddies sequence that spread over 16 orders of velocity magnitude. The main vortex is displayed approximately from 1 to 0.1 and between 0.1 to 0.0 the sequence of vortices is shown. The length ratio between the secondary vortex and the main vortex is 0.1, and the velocity ratio is 4×10^{-2}; for the quaternary vortex, the length ratio is approximately 6×10^{-4} and the velocity ratio is 5×10^{-4}. These results are demanding from the numerical point of view and indeed are on the limit of machine precision. The spectral radius of the coefficient matrix is extreme because of the range of cell sizes, making this a difficult case to converge. Nevertheless, it can be seen that the signature of the 5th vortex on left and right corner is present in the results, despite of its very small velocity magnitude of
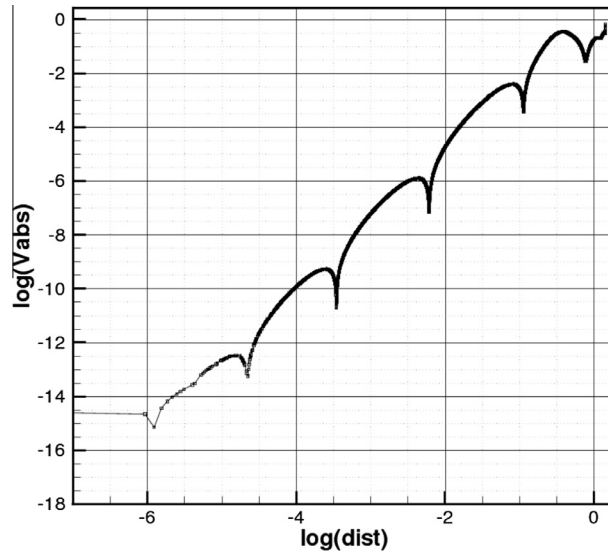
Fig. 13. Lid-Driven cavity: log plot of absolute velocity in the diagonal (0,0) to (1,1). The distance is measured along the diagonal starting from (0,0) and marked as the power of 10.
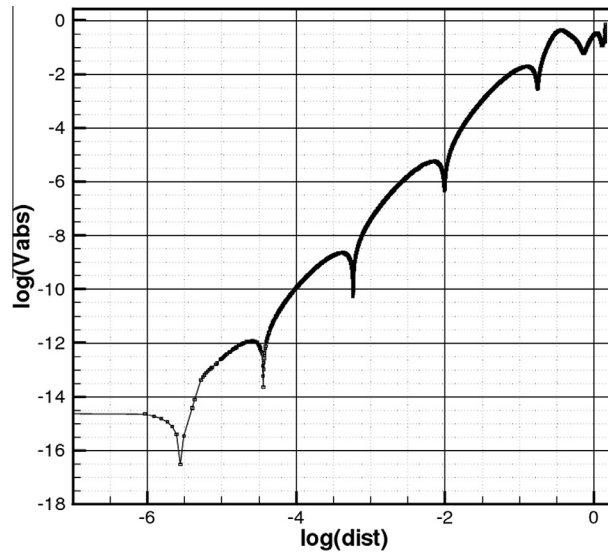

Fig. 14. Lid-Driven cavity: log plot of absolute velocity in the diagonal (1,0) to (0,1). The distance is measured along the diagonal starting from (1,0) and marked as the power of 10.

Table 2
Cavity vortices additional parameters.

Vortex	R_t	V_t	Re_{vortex}	Ratio	V_t decay
2R	5.2316E−2	6.85E−2	3.58371	0.82856	–
3R	4.09408E−3	2.88338E−5	1.18048E−4	0.94429	4.2093E−4
4R	2.45427E−4	1.22629E−8	3.00965E−9	0.98131	4.25296E−4
5R	1.0457E−5	5.44273E−12	5.69146E−14	1.09091	4.438371E−4
2L	4.23247E−2	2.22562E−2	9.41986E−1	1.32946	–
3L	3.45196E−3	1.15166E−5	3.97548E−5	1.0289	5.174558E−4
4L	2.25305E−4	5.19926E−9	1.17142E−9	1.01563	4.514579E−4
5L	1.2315E−5	2.74822E−12	3.38443E−14	0.95101	5.285791E−4

$O[10^{-16}]$. The last fifth vortex is not completely resolved. Any further refinement was prevented because the velocity magnitude is of the order of 10^{-16} and the truncation error saturates the solution. The proposed error criterion is able to detect phenomena over a wide range of scales without being overwhelmed by dynamically dominant phenomena.

According to [1] for a Reynolds number, based on the distance from the corner, less than the unity the inertia forces are negligible and the Stokes flow solution displays multiple eddies of decreasing size and rapidly decreasing intensity. All eddies (for a given corner angle) are geometrically and dynamically similar but with successive changes of length and velocity scales given by the factors present by Moffatt [1]. For sharp corners with 90 degrees, the factors are 0.0614 and 4.8566×10^{-4}. The present prediction obtained with a general error estimation procedure and without any particular adaptation to capture very small eddies in the flow and gave the length and velocity factors of 0.06001 and 4.5×10^{-4}.

The prediction display a ratio of the dimensions of third and fourth successive eddies, 0.065 and 0.054. The sixth right vortex center will be at a distance from the corner 3.0226×10^{-6} and will require a minimum mesh spacing of 1.907×10^{-7} for a similar accuracy of the present results.

Table 2 lists other parameters to study the successive vortices in the cavity corners. Where R_t is the lowest distance between the vortex center and boundary, V_t is the tangential velocity, "Ratio" stands for the geometric ratio of the separation points of the vortex X/Y, the Reynolds number $Re_{vortex} = \frac{V_t R_t}{\nu}$ and the V_t decay factor is the ratio of V_t between successive corners vortices. The predicted tangential velocity decay 4.3×10^{-4} is not far from the analytical value 4.8566×10^{-4} of [1]. The vortices Reynolds number are very small meaning $O(10^{-14})$ that will be very difficult to visualize them with experimental techniques.

The calculations at the last refinement level comprised 2×10^5 cells, the smallest mesh size is equal to 3.05×10^{-6}, meaning that the uniform mesh with the smallest mesh size would have 1.1×10^{11} cells, consequently, the adaptive mesh contains 0.00018% of the uniform mesh.

5. Conclusions

A Finite-Volume 2D Navier–Stokes solver was developed for unstructured polyhedral hybrid adaptive meshes with second order accurate spatial discretization. A new error estimation criterion was developed for adaptive space discretization (h-refinement).

The algorithm was applied to the lid-driven cavity flow at $Re = 1000$ and apart from the main cavity vortex the corner vortices chain was fully resolved up to the fourth cavity corner eddy and the fifth vortex was partially resolved, the mesh adaptation was fully driven by the mesh refinement criteria that allocated meshes on the cavity corners without any external prescription.

The main cavity eddies were well resolved and the velocity predictions are virtually identical to the benchmark data, [17,18]. The small eddies in the creep flow sharp corner region were also quantitatively very well captured in agreement with Moffatt [1]. The new error estimation criterion and the second order developed Navier–Stokes solver have allowed to calculate in detail the cavity eddies whose characteristic length scales are five order of magnitude lower and sixteen order in tangential velocity intensity. On a macroscopic inspection of the grid, the criterion does not present preferences for large scale in detriment of small-scale phenomena.

We are not aware of any benchmark results about these corner vortices and this, to the best of our knowledge, constitutes the first time these vortices were resolved from full solution of the Navier–Stokes equations without additional theoretical assumptions or forcing the location of meshes on the corner regions.

Acknowledgments

Appendix A. Adding diffusive terms to matrix

This appendix details the procedure employed to embed the diffusive term containing the LS regression for the cell-centered gradient in the coefficient matrix. The objective is to rewrite the Laplacian given by Eq. (12) in the form:

$$\sum_{f_k \in \mathcal{F}(P_0)} \widehat{\nabla \phi_{P_k}} \cdot \mathbf{S}_f = a_{P_0} \phi_{P_0} + \sum_{f_k \in \mathcal{F}(P_0)} a_{P_k} \phi_{P_k}, \tag{A.1}$$

where P_k stands for the neighbor cell of face f_k in relation to P_0. Eq. (12) is a non trivial relationship because of the incorporation of the gradient given by Eq. (13) in each neighbor cell.

Let us consider one element of the sum on the left hand side of Eq. (A.1), dropping the subscript f_k for convenience. Assuming that the cells containing f_k are P_0 and P_1, and remembering that $\mathbf{S}$ points from P_0 to P_1, the diffusive flux through f_k is

$$\widehat{\nabla \phi} \cdot \mathbf{S} = (\phi_{P_1} - \phi_{P_0}) \Delta \cdot \mathbf{S} + \frac{1}{2} \left[\nabla \phi_{P_0} - \left(\nabla \phi_{P_0} \cdot \mathbf{d} \right) \Delta \cdot \mathbf{S} \right] + \frac{1}{2} \left[\nabla \phi_{P_1} - \left(\nabla \phi_{P_1} \cdot \mathbf{d} \right) \Delta \cdot \mathbf{S} \right], \tag{A.2}$$

Looking at the right hand side of this expression it can be seen that the first term represents a strictly two-point local contribution, which can be trivially evaluated via a single loop over all grid faces. The second and third terms represent a contribution in spatial scope which spans the direct neighborhood of both cells P_0 and P_1 by way of the gradients $\nabla\phi_{P_0}$ and $\nabla\phi_{P_1}$, respectively. These are the terms whose implicit discretization we wish to address here.

In terms of implementation, both these terms are subject to the same treatment and can be evaluated in distinct iterations of a single loop. For this reason we need only address the first one, which pertains cell P_0. We want to recast it into implicit form:

$$\left[\nabla\phi_{P_0} - (\nabla\phi_{P_0} \cdot \mathbf{d})\,\Delta \cdot \mathbf{S}\right] \rightarrow a_{P_0}\phi_{P_0} + \sum_{f_k \in \mathcal{F}(P_0)} a_{P_k}\phi_{P_k}. \tag{A.3}$$

The least squares gradient $(\nabla\phi)_{P_0}$ is obtained by solving an overdetermined system for the differences: $\mathbf{Xg} = \mathbf{z}$,

$$\begin{bmatrix} x_{P_1}-x_{P_0} & y_{P_1}-y_{P_0} & z_{P_1}-z_{P_0} \\ x_{P_2}-x_{P_0} & y_{P_2}-y_{P_0} & z_{P_2}-z_{P_0} \\ \vdots & \vdots & \vdots \\ x_{P_m}-x_{P_0} & y_{P_m}-y_{P_0} & z_{P_m}-z_{P_0} \end{bmatrix} \begin{bmatrix} \frac{\partial\phi}{\partial x} \\ \frac{\partial\phi}{\partial y} \\ \frac{\partial\phi}{\partial z} \end{bmatrix} = \begin{bmatrix} \phi_{P_1}-\phi_{P_0} \\ \phi_{P_2}-\phi_{P_0} \\ \vdots \\ \phi_{P_m}-\phi_{P_0} \end{bmatrix}, \tag{A.4}$$

possibly with weights w_{P_1}, where $m = \sharp\mathcal{N}(P_0)$ is the number of face neighbors of P_0. We need to symbolically solve this system, so that we can obtain the solution as a function of the values of ϕ in $\mathcal{N}(P_0)$. Despite having worse conditioning properties, the normal form of the least squares system must be used to allow obtaining the symbolic solution in terms of the original matrix coefficients. The numerically more convenient form of the Moore–Penrose generalized inverse would be preferable, but the procedure would explicitly depend on the set of neighbors considered to perform the regression, which makes it impossible to carry out this task in a generalized way. The normal form of the system is given by

$$\mathbf{X'WXg} = \mathbf{X'Wz}, \tag{A.5}$$

or simply $\mathbf{Ag} = \mathbf{h}$, where

$$a_{ij} = \sum_{k\in N(P_0)} w_k \Delta x_{i,k} \Delta x_{j,k}, \tag{A.6}$$

and

$$h_i = \sum_{k\in N(P_0)} w_k \Delta x_{i,k}(\phi_{P_k} - \phi_P), \tag{A.7}$$

with

$$\Delta x_{i,k} = x_i(P_k) - x_i(P_0). \tag{A.8}$$

Let $\mathbf{Q} = \mathbf{A}^{-1}$. The components of $\mathbf{Q}$ are easily obtained in symbolic form using the Cramer rule, and are amenable to treatment provided they return their sum form. Then $\mathbf{g} = \mathbf{Q\,h}$ we have, after some algebra:

$$a_P\phi_P + \sum_{n=1}^{nc} a_n\phi_n = \{[\mathbf{Q} - (\mathbf{Q} \cdot \mathbf{d} \otimes \Delta)] \cdot \mathbf{S}\} \cdot \mathbf{h}. \tag{A.9}$$

Substituting $\mathbf{h}$ from (A.7) in the above expression, we finally obtain a general implicit expression like (A.9), with the vector $\mathbf{c}$ given by $[\mathbf{Q} - (\mathbf{Q} \cdot \mathbf{d} \otimes \Delta)] \cdot \mathbf{S}$:

$$\mathbf{c} = \begin{bmatrix} (Q_{11} - r_1\Delta_1)S_1 + (Q_{12} - r_1\Delta_2)S_2 + (Q_{13} - r_1\Delta_3)S_3 \\ (Q_{12} - r_2\Delta_1)S_1 + (Q_{22} - r_2\Delta_2)S_2 + (Q_{23} - r_2\Delta_3)S_3 \\ (Q_{13} - r_3\Delta_1)S_1 + (Q_{23} - r_3\Delta_2)S_2 + (Q_{33} - r_3\Delta_3)S_3 \end{bmatrix}, \tag{A.10}$$

where the vector $\mathbf{r}$ is given by $\mathbf{Q} \cdot \mathbf{d}$:

$$\mathbf{r} = \begin{bmatrix} Q_{11}d_1 + Q_{12}d_2 + Q_{13}d_3 \\ Q_{12}d_1 + Q_{22}d_2 + Q_{23}d_3 \\ Q_{13}d_1 + Q_{23}d_2 + Q_{33}d_3 \end{bmatrix}. \tag{A.11}$$

References

[1] H. Moffatt, Viscous and resistive eddies near a sharp corner, J. Fluid Mech. 18 (1963) 1–18.
[2] I. Babuska, W. Rheinbolt, A-posteriori error estimates for the Finite Element method, Int. J. Numer. Methods Eng. 12 (1978) 1597–1615.
[3] J. Peraire, M. Vahdati, K. Morgan, O. Zienkiewicz, Adaptive remeshing for compressible flow computations, J. Comput. Phys. 72 (1987) 449–466.
[4] M. Berger, P. Colella, Local adaptive mesh refinement for shock hydro dynamics, J. Comput. Phys. 52 (1989) 64–84.
[5] R. Verfurth, A Review of A Posteriori Error Estimation and Adaptive Mesh-Refinement Techniques, Teubner Verlag and J. Wiley, 1996.

[6] M. Aimsworth, J. Oden, A Posteriori Error Estimates in Finite Element Analysis, Wiley-Interscience, 2000.
[7] D. Venditti, D. Darmofal, Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows, J. Comput. Phys. 187 (2003) 22–46.
[8] S. Muzaferija, D. Gosman, Finite Volume CFD procedure and adaptive error control strategy for grids of arbitrary topology, J. Comput. Phys. 138 (1997) 766–787.
[9] M. Thompson, J. Ferziger, An adaptive multigrid technique for the incompressible Navier-Stokes equations, J. Comput. Phys. 82 (1989) 94–121.
[10] H. Jasak, D. Gosman, Automatic resolution control for the Finite-Volume method. Part 1: a-Posterior error estimates, Numer. Heat Trans. B 38 (2000) 237–256.
[11] R. Burggraf, Analytical and numerical studies of the structures of steady separated flows, J. Fluid Mech. 24 (1966) 113–151.
[12] J. Bozemann, C. Dalton, Numerical study of viscous flow in a cavity, J. Comput. Phys. 12 (1973) 348–363.
[13] P. Shankar, M. Deshpande, Fluid mechanics in the driven cavity, Ann. Rev. Fluid Mech. 32 (2000) 93–136.
[14] M. Cheng, K. Hung, Vortex structure of steady flow in a rectangular cavity, Comput. Fluids 35 (2006) 1046–1062.
[15] K. Gustafson, K. Halasi, Vortex dynamics of cavity flows, J. Comput. Phys. 64 (1986) 279–319.
[16] K. Gustafson, K. Halasi, Cavity flow dynamics at higher Reynolds number and higher aspect ratios, J. Comput. Phys. 70 (1987) 271–283.
[17] U. Ghia, K. Ghia, C. Shin, High-re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, J. Comput. Phys. 48 (1982) 387–411.
[18] O. Botella, R. Peyret, Benchmark spectral results on the lid-driven cavity flow, Comput. Fluids 27 (1998) 421–433.
[19] E. Erturk, T. Corke, C. Gökçöl, Numerical solutions of 2-D steady incompressible driven cavity flow at high Reynolds numbers, Int. J. Numer. Methods Fluids (2005).
[20] F.E. Ham, F.S. Lien, A.B. Strong, High-re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method, J. Comput. Phys. 179 (2002) 469–494.
[21] C.J. Heaton, On the appearance of Moffat eddies in viscous cavity flow as the aspect ratio varies, Phys. Fluids 20 (2008) 103102.
[22] W.M. Collins, S.C.R. Dennis, Viscous eddies near a 90 and a 45 corner in flow through a curved tube of triangular cross-section, J. Fluid Mech. 76 (1976) 417–432.
[23] P. Shankar, Moffat eddies in the cone, J. Fluid Mech. 539 (2005) 113–135.
[24] K. Gustafson, R. Leben, Vortex subdomains, in: R. Glowinski, G. Meurant, J. Periaux (Eds.), First International Symposium on Domain Decomposition Methods for Partial Differential Equations, SIAM, Philadelphia, 1988, pp. 370–380.
[25] A.V. Shapeev, P. Lin, An asymptotic fitting Finite Element method with exponential mesh refinement for accurate computation of corner eddies in viscous flows, SIAM J. Sci. Comput. 31 (2009) 1874–1900.
[26] B. Mirtich, Fast and accurate computation of polyhedral mass properties, J. Graphics Tools 1 (1996).
[27] M. Kobayashi, J.M.C. Pereira, J.C.F. Pereira, A conservative Finite-Volume second-order accurate projection method on hybrid unstructured grids, J. Comput. Phys. 150 (1999) 40–75.
[28] R. Verstappen, A. Veldman, Symmetry-preserving discretization of turbulent flow, J. Comput. Phys. 187 (2003) 343–368.
[29] K. Mahesh, G. Constantinescu, P. Moin, A numerical method for large-eddy simulation in complex geometries, J. Comput. Phys. 197 (2004) 215–240.
[30] S. Benhamadouche, K. Mahesh, G. Constantinescu, Colocated Finite-Volume schemes for large-eddy simulation on unstructured meshes, Center for Turbulence Research, in: Proceedings of the Summer Program, Proceedings, 2002.
[31] U. Trottenberg, C. Osterlee, A. Schöller, P. Oswald, Multigrid, Academic Press, 2001.