Available online at www.sciencedirect.com
SCIENCE DIRECT®
Computers & Fluids 33 (2004) 1305–1333
computers & fluids
www.elsevier.com/locate/compfluid

A finite volume method to solve the 3D Navier–Stokes equations on unstructured collocated meshes

Sébastien Perron [a,*,1], Sylvain Boivin [b], Jean-Marc Hérard [c,d]

[a] *ARDC, Alcan, Applied Science Research Group, 1955 Mellon Blvd, P.O. Box 1250, Jonquiere, Quebec, Canada G7S 4K8*
[b] *Université du Québec à Chicoutimi, 555 Boulevard de l'université, Chicoutimi, Quebec, Canada G7H 2B1*
[c] *DRD, Électricité de France, 6 quai Watier Chatou, France*
[d] *CMI, Université de Provence, 39 rue Joliet et Curie, 13453 Marseille Cedex 13, France*

Received 20 September 2002; received in revised form 23 June 2003; accepted 6 October 2003

Abstract

A new method to solve the Navier–Stokes equations for incompressible viscous flows and the transport of a scalar quantity is proposed. This method is based upon a fractional time step scheme and the finite volume method on unstructured meshes. The governing equations are discretized using a collocated, cell-centered arrangement of velocity and pressure. The solution variables are stored at the cell-circumcenters. Theoretical results and numerical properties of the scheme are provided. Predictions of lid-driven cavity flow, flows past a cylinder and heat transport in a cylinder are performed to validate the method.

1. Introduction, mathematical model

When the density of a viscous fluid is constant, the transport of a scalar quantity by an incompressible flow in a domain $\Omega \times [0, T]$ is governed by the Navier–Stokes equations:

$$\nabla \cdot \mathbf{v} = 0 \tag{1}$$

* Corresponding author. Tel.: +1-418-699-6585x2417; fax: +1-514-699-3148.
E-mail addresses: sebastien.perron@alcan.com, crda@alcan.com (S. Perron), sylvain-h_boivin@uqac.ca (S. Boivin), jean-marc.herard@edf.fr (J.-M. Hérard).
[1] Partially supported by NSERC, Canada.

0045-7930/$ - see front matter © 2004 Elsevier Ltd. All rights reserved.
doi:10.1016/j.compfluid.2003.10.006

$$\frac{\partial \mathbf{v}}{\partial t} + \nabla \cdot [\mathbf{v} \otimes \mathbf{v} - v\nabla \mathbf{v}] + \nabla P = \mathbf{f} \tag{2}$$

$$\frac{\partial \phi}{\partial t} + \nabla \cdot [\mathbf{v}\phi - \alpha \nabla \phi] = s \tag{3}$$

where

- $P = p/\rho$, P being the kinematic pressure, ρ the density and p the pressure;
- $v = \mu/\rho$, v being the kinematic viscosity and μ the dynamic viscosity;
- $\mathbf{f}$ is a source term such as buoyancy force;
- ϕ is a scalar quantity such as a concentration;
- α is the diffusivity and s a source term.

In order to solve these equations, they must come with appropriate boundary conditions and, for a non-permanent flow, a suitable initial condition. By adjusting the source terms to the problem being considered, this model can be extended to many applications such as heat transfer or the transport of a pollutant. A discretization method that is well suited for the numerical simulation of such flows is the finite volume method. It can be used in complex geometries and often leads to robust and cheap schemes. It also allows local conservativity of the numerical flux, which is quite attractive for the modeling of problems for which the flux is of importance.

Incompressible viscous flow solvers based on a finite volume discretization of the domain have been used as early as the sixties by Harlow and Welch [1]. Due to the lack of a time-evolution equation for the pressure and the occurrence of the continuity equation, it is well known that on collocated grids, straight forward discretization of the incompressible flow equations leads to an unphysical odd–even coupling of the pressure (often called spurious checkerboard modes). The oldest and most straight forward approach to avoid this problem is to use different grid points for the pressure and the velocity (such grids are usually referred as ''staggered grids''). In their scheme, Harlow and Welch put the pressure at the cell centers and the normal components of the velocity vector were located at the middle of the faces (this staggering of the variables on Cartesian grids has been described in details by Patankar [2]). More recent developments of staggered grids for unstructured meshes include the method proposed by Hwang [3] where the pressure is stored at the centroids and the velocity vector at the cell faces of triangular meshes. Koshizuka et al. [4] have proposed a finite volume scheme for 2D grids where the velocity is stored at the centroids of the cells and the pressure on the vertices. For Delaunay 2D meshes, Perot [5] has proposed a finite volume scheme where the normal velocity components are stored at the face mid-points and the pressure at the circumcenters of the triangles.

Usually, the extension of staggered unstructured meshes to general 3D unstructured grids is not straightforward and, for general meshes, most unstructured finite volume incompressible flow solvers avoid the false pressure modes by implicitly (or explicitly) adding a small diffusion term in the continuity equation. The most common approaches are those based on the pressure weighted interpolation of Rhie and Chow [6]. For those schemes using colocated grids, a stabilizing term is implicitly introduced when the cell face velocity components are interpolated in the continuity equation. Unfortunately, in the first method proposed by Rhie and Chow, the final result could

depend on certain relaxation factors and the iterative procedure. Later on, a more effective pressure-weighted interpolation (PWI) have been proposed by Miller and Griffiths [7]. More recently, results on unstructured collocated meshes have been published by other researchers using the PWI method [8–10].

Often, those incompressible flow solvers use temporal discretizations based on pressure-correction methods where a time step is split up in sub steps (these are often referred as fractional step methods). The system (1)–(3) is not solved as it stands, but in a decoupled manner. First a prediction of a starred velocity field that dot not satisfy the continuity equation is made. Then a correction involving the pressure is computed such that the continuity equation is satisfied. The pressure-correction method has been formulated and studied by numerous authors [1,11–13]. Notwithstanding the space discretization, the temporal accuracy for the velocity is of the same order as the order of accuracy of the underlying time stepping method (first order for Euler, second order for Cranck–Nickolson, ...). But the accuracy for the pressure evolution is only first order.

In this paper, a cell centered finite volume scheme is presented. The time discretization of the Navier–Stokes equations is based upon the fractional time step method constructed in the late 1960s by Chorin [11] and Teman [14]. The spatial discretization is an application of recent theoretical results on finite volume methods published by Gallouët and coworkers [15] and a generalization of the recent work of Boivin et al. [16,17]. In these papers, those authors proposed a finite volume scheme for 2D unstructured triangular meshes and gave an extension to two phase flows. The scheme presented here after is a generalization of this work to 3D general meshes where a control volume can be an assembly of tetrahedrons.

2. Time discretization

The time discretization is semi-implicit and based upon a variation of the projection scheme originally proposed by Chorin [11] and Teman [14] and often called "projection-2" scheme. Let,

$$\left.\frac{\partial \phi}{\partial t}\right|_{t=t_{n+1}} = \begin{cases} \dfrac{\phi^{(n+1)} - \phi^{(n)}}{\delta t}, & \text{permanent flow} \\ \dfrac{3\phi^{(n+1)} - 4\phi^{(n)} + \phi^{(n-1)}}{2\delta t}, & \text{otherwise} \end{cases}$$

be the approximation of the temporal derivative and

$$\mathbf{v}(t = t_{n+1}) = \begin{cases} 2\mathbf{v}^{(n)} - \mathbf{v}^{(n-1)}, & \text{momentum equations} \\ \mathbf{v}^{(n+1)}, & \text{other equations} \end{cases}$$

the approximation of the velocity field at time $t = t_{n+1}$, the time discretization of the governing equations is the following:

- Prediction:

$$\left.\frac{\partial \mathbf{v}}{\partial t}\right|_{t=t_{n+1}} + \nabla \cdot \left[\mathbf{v}(t = t_{n+1}) \otimes \mathbf{v}^{(n+1/2)} - \nu \nabla \mathbf{v}^{(n+1/2)}\right] + \nabla P^n = \mathbf{f}^n \tag{4}$$

- Projection:

$$\frac{\mathbf{v}^{(n+1)} - \mathbf{v}^{(n+1/2)}}{\delta t} = -\beta \nabla \left(\delta P^{(n+1)}\right), \quad \delta P^{(n+1)} = P^{(n+1)} - P^{(n)} \tag{5}$$

$$\nabla \cdot \mathbf{v}^{(n+1)} = 0 \tag{6}$$

- Convection and diffusion of other scalar variables:

$$\left.\frac{\partial \phi}{\partial t}\right|_{t_{n+1}} + \nabla \cdot \left[\mathbf{v}(t = t_{n+1})\phi^{(n+1)} - \alpha \nabla \phi^{(n+1)}\right] = s^{(n)} \tag{7}$$

The parameter $\beta \in (0, 2]$ is a used for relaxation and its value is not arbitrary. When the flow is steady, the final solution does not depend on its value. But the rate of convergence does so. It was found that a value of $\beta = \frac{2}{3}$ often gives the best rate of convergence. Hence this value is the one retained for all stationary results shown in this article. For a transient flow, the value $\beta = 2$ was chosen accordingly with the results published by Shen et al. on projection schemes [12,13,18]. For a transient flow, the linear parabolic operator is approximated with a second-order backward scheme known as "Gear's scheme". As for the non-linear convective terms, a semi-implicit time discretization is preferred. The convective field in the momentum equations is extrapolated with the Adams–Moulton scheme. At every time step, the vector field $\mathbf{v}^{(n)}$ is divergence free. Thus, the convective field for all equations is a linear combination of divergence free vector fields.

3. Space discretization

3.1. Geometrical elements

The initial mesh is built with tetrahedrons. For each tetrahedron K, the reference position X_K is the circumcenter of the element. This point is allowed to be inside K, outside K, or on its boundary (see Fig. 1).

Before continuing, a remark about this unusual choice of reference position must be added Many cell centered finite volume methods whose interpolation functions are constant by control volume use the center of the cells as a reference position to compute the diffusion flux. [2] Let ϕ be a scalar variable, K and L be adjacent control volumes whose centers are Xc_K and Xc_L, many computer codes use the following approximation of the normal gradient at the cell interfaces:

$$\nabla \phi \cdot \mathbf{n}_K = \frac{\phi_L - \phi_K}{d_{K\backslash L}}$$

where $d_{K\backslash L}$ is the distance between the centers of cells K and L. For non-orthogonal meshes, this approximation of the flux is not consistent, which can lead to important errors. Even the approximation

[2] In this paper, the diffusion flux is the physical flux (or viscous flux for the momentum equations).

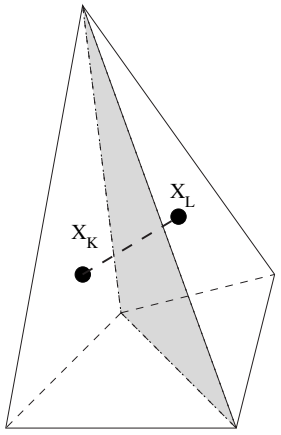

Fig. 1. Circumcenters for tetrahedrons K and L.

$$\nabla\phi \cdot \mathbf{n}_K = \frac{\phi_L - \phi_K}{(Xc_K - Xc_L) \cdot \mathbf{n}_K}$$

does not lead to a consistent approximation of the flux and there is a singularity when $(Xc_L - Xc_K) \cdot \mathbf{n}_K$ tends to zero. For those discretization schemes, accurate computations of reference quantities related to the diffusion flux (Nusselt number, drag coefficient, lift coefficient, friction coefficient, ...) can be tedious. Finally, it is important to point out that for viscous flows, if the approximation of the diffusion flux is not consistent, the error does not vanish with grid refinement [19]. With a cell based gradient reconstruction using a least-square method or the Green–Gauss theorem, it is also possible to approximate the gradient at an interface from a simple average [20,21]. However, it can lead to a wider stencil with a less favorable weight distribution of the coefficients [22,23]. In this latter case, the discretization of the Laplacian can be non-positive. Moreover, since the approximation of $\nabla\phi \cdot \mathbf{n}_K$ on $K|L$ is not necessarily equal to the approximation $-\nabla\phi \cdot \mathbf{n}_L$ that is used in the cell L, this approach also raises another difficulty about the conservativity of the diffusion flux.

3.2. Control volumes and approximation of functions

Let $\sigma_{K,L}$ be the interface between tetrahedrons K and L and $\mathbf{n}_{K,L}$ the outward normal unit vector of the interface $\sigma_{K,L}$ of K. Then, the following quantity called "transmittivity" is introduced [15]:

$$\tau_{K,L} = \frac{m(\sigma_{K,L})}{(X_L - X_K) \cdot \mathbf{n}_{K,L}}$$

where $m(\sigma_{K,L})$ is the area of $\sigma_{K,L}$. Four categories of meshes can be encountered:

- Category M_1: For all interfaces $\sigma_{K,L}$, $\tau_{K,L} > 0$ and for all tetrahedrons K, $X_K \in K$. The control volumes are the tetrahedrons.

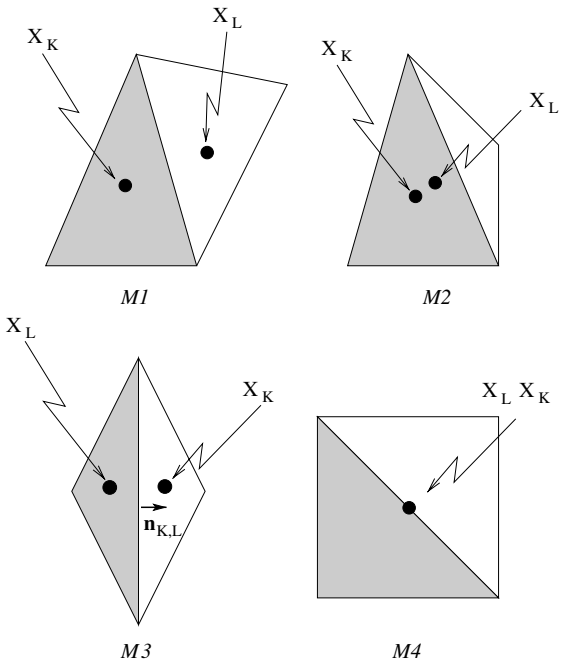

Fig. 2. Mesh categories.

- Category M_2: For all interfaces $\sigma_{K,L}$, $\tau_{K,L} > 0$, but there exist at least one tetrahedron K for which $X_K \notin K$. The control volumes are the tetrahedrons.
- Category M_3: For at least one interface $\sigma_{K,L}$, $\tau_{K,L} < 0$, the Delaunay condition is not fulfilled. The tetrahedrons K and L are combined together to form a new macro-element and the control volume is the macro-element. At least two positions (X_K and X_L) are associated to this control volume.
- Category M_4: For at least one $\sigma_{K,L}$, $|\tau_{K,L}| \to \infty$, at least two circumcenters occupy the same position. The elements K and L are combined to form a new macro-element and the control volume is the macro-element. At least two positions (X_K and X_L) are associated to this control volume.

These categories are the same in 2D when triangles are the elements, they are shown on Fig. 2. In the 2D case, triangles in category M_4 are such that $X_K = X_L$. For meshes M_1 and M_2, the Delaunay condition is satisfied.

For categories M_3 and M_4, tetrahedrons are combined together in a large macro-element. This macro-element is used as the control volume. In practice, it is almost impossible to construct meshes in category M_1 or M_2 in 3D.

For all variables, piecewise constant functions by control volumes are used for approximation. Even when there are more than one position associated with a control volume (meshes of categories M_3 and M_4), there is only one degree of freedom (DOF) or unknown per control volume.

3.3. Discrete equations, convection–diffusion operator $\mathscr{C}\mathscr{D}$

Let us consider the following equation on the domain Ω:

$$\left.\frac{\partial \phi}{\partial t}\right|_{t=t_{n+1}} + \nabla \cdot \left(\mathbf{v}(t_{n+1}) \otimes \phi^{n+1}\right) - \nabla\left(\alpha \nabla \phi^{n+1}\right) = s^n \tag{8}$$

where

- ϕ can be any scalar variable or the components of the velocity vector (v_x, v_y, v_z);
- $\left.\frac{\partial \phi}{\partial t}\right|_{t=t_{n+1}}$ was defined in Eq. (2);
- $s^{(n)}$ is a source term;
- $\mathbf{v}(t_{n+1})$ was defined in Eq. (2), it is such that $\nabla \cdot \mathbf{v}(t_{n+1}) = 0$.

The discrete equations are obtained by integrating (8) over each control volume K and applying the Gauss theorem:

$$\int_K \left.\frac{\partial \phi}{\partial t}\right|_{t=t_{n+1}} \mathrm{d}V + \int_{\partial K} \left(\mathbf{v}(t_{n+1}) \otimes \phi^{(n+1)}\right) \cdot \mathbf{n}\,\mathrm{d}S - \int_{\partial K} \nabla\left(\alpha \nabla \phi^{(n+1)}\right) \cdot \mathbf{n}\,\mathrm{d}S = \int_K s^{(n)}\,\mathrm{d}V \tag{9}$$

The quantities $\phi^{(n+1)}$ and $s^{(n)}$ are assumed constant over any given volume K.

The discrete equation for K is given by this expression:

$$m(K) \left.\frac{\partial \phi}{\partial t}\right|_{t=t_{n+1}} + \sum_{\sigma \in \mathscr{E}_K} F^C_{K,L} - F^D_{K,L} = m(K) s^{(n)}_K \tag{10}$$

where $\mathscr{E}_K$ is the set of interfaces which belong to the boundary of volume K and $m(K)$ is the measure of the hyperplane K (the volume of the macro-element K in 3D). $F^C_{K,L}$, $F^D_{K,L}$ are the discrete approximation of the convection and diffusion flux and these are given by the following expressions:

$$\int_{K/L} (\mathbf{v}\phi(\mathbf{x})) \cdot \mathbf{n}(\mathbf{x})\,\mathrm{d}S \approx m(\sigma_{K,L}) v_{K,L} \phi_{K,+} = F^C_{K,L} \tag{11}$$

$$\int_{K/L} \alpha \nabla \phi(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x})\,\mathrm{d}S \approx \alpha_{K,L} \tau_{K,L} (\phi_L - \phi_K) = F^D_{K,L} \tag{12}$$

- $v_{K,L}$ is an approximation of the speed normal to the interface $\sigma_{K,L}$, $\mathbf{n}(\mathbf{x})$ is the outward normal unit vector;
- $\phi_{K,+} = \begin{cases} \phi_K, & v_{K,L} \geqslant 0 \\ \phi_L, & \text{otherwise} \end{cases}$
- $\alpha_{K,L}$ is a discrete approximation of the diffusivity $\alpha(\mathbf{x})$ at the interface $\sigma_{K,L}$ (more details on this approximation will be given in Section 3.4).

$\tau_{K,L}$ is the transmittivity of the interface $\sigma_{K,L}$.

The system (10) is linear but not symmetric and the associated matrix is a diagonal dominant M-matrix. This implies that A^{-1} has all its coefficients greater or equal to zero and as a consequence, for suitable source terms s_K ($s_K = 0$ for example), the discrete maximum principle will hold for ϕ_K.

The system (10) is used to solve all scalar variables. The solution of this system will be denoted as $\phi_K^{(n+1)} = CD\left(\phi_K^{(n)}\right)$. Thanks to the time discretization, the components of the velocity vector are solved in a decoupled manner. Therefore, the convection and diffusion of the velocity field $\mathbf{v}^{(n)}$ can be considered as the convection and diffusion of its three components: $\mathbf{v}_K^{(n+1/2)} = CD\left(\mathbf{v}_K^{(n)}\right)$ (lets not forget that the convection and diffusion of the velocity vector components is only an intermediate solution). It is worth mentioning that for the momentum equations, the pressure gradient is taken into account with the following term:

$$\int_{K/L} p\mathbf{n}(\mathbf{x})\,\mathrm{d}S \approx m(\sigma_{K,L})p_{K,L}\mathbf{n}_{K,L} \tag{13}$$

where $p_{K,L}$ is the discrete approximation of the pressure at the interface $\sigma_{K,L}$. After convecting and diffusing the velocity field, the solution $\mathbf{v}^{(n+1/2)}$ is not divergence free and a projection has to be made.

3.4. Discrete approximation at an interface

Meshes where all the reference positions lie in their associated control volume are first considered. For those meshes, the discrete approximation at an interface is straight forward. Consider the interface $\sigma_{K,L}$, the value for ϕ at this interface can be approximated with the linear interpolation

$$\phi_{K,L} = (1 - t_{K,L})\phi_K + t_K\phi_L \tag{14}$$

$$t_{K,L} = \frac{(X_K - X_{K,L}) \cdot (X_L - X_K)}{(X_L - X_K) \cdot (X_L - X_K)} \tag{15}$$

Unfortunately, the most common meshes are those for which there can exist reference positions that do not lie in their associated control volume. This is the case for meshes of category M_2 (see Fig. 3). When $t_{K,L} \gg 1$ or $t_{K,L} \ll 0$, the minimum or the maximum of the approximative solution cannot be preserved at the interface. Preliminary numerical results showed that this linear interpolation could even make the scheme unstable when $t_{K,L}$ did not satisfied the inequality $0 \leqslant t_{K,L} \leqslant 1$, which is rather frequent in 3D.

In order to preserve the maxima and the minima of the solution when interpolating a scalar quantity other that the diffusivity at an interface, the mean value is used as an approximation

$$\phi_{K,L} = \tfrac{1}{2}(\phi_K + \phi_L) \tag{16}$$

Care must be given to the approximation of the diffusivity. In the report [24], numerical results show the rate of convergence of the diffusion operator $F_{K,L}^D$ is highly dependent on the approximation of the diffusivity when this physical property is discontinuous. For general meshes, where reference positions do not necessarily lie in their associated control volume, care must be given to the computation of the harmonic average. First, the following "signed" distance is defined:

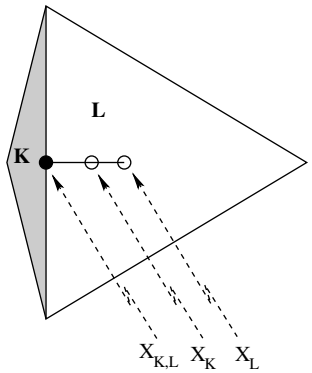

Fig. 3. Interpolation at an interface.

- $d_{K,\sigma} = (X_\sigma - X_K) \cdot \mathbf{n}_{K,L}$,
- $d_{K,L} = (X_L - X_K) \cdot \mathbf{n}_{K,L}$,
- $d_{\sigma,L} = (X_L - X_\sigma) \cdot \mathbf{n}_{K,L}$,

where $d_{K,L} = d_{K,\sigma} + d_{\sigma,L}$. Let $\alpha_K = \alpha(X_K)$, the harmonic average of α at the interface $\alpha_{K,L}$ is defined as:

$$\alpha_{K,L} = \frac{\alpha_L \alpha_K}{\frac{d_{\sigma,L}}{d_{K,L}}\alpha_K + \frac{d_{K,\sigma}}{d_{K,L}}\alpha_L} \tag{17}$$

3.5. Correction of the diffusion coefficient

It is well known that the upwind scheme implicitly introduces too much diffusion. In order to gain more precision when needed, the diffusion coefficient is corrected. This correction is based upon the power law scheme introduced by Patankar [2]. Let $\alpha_{K,L}$ be an approximation of the diffusion coefficient at the interface $\sigma_{K,L}$, it is corrected as follows:

$$\widehat{\alpha}_{K,L} = \alpha_{K,L} \cdot \max\left(0, (1 - 0.1Re_l)^5\right) \tag{18}$$

where $Re_l = \frac{|v_{K,L}|\|X_K - X_L\|}{\alpha_{K,L}}$ is called the "local Reynolds number". Since $\lim_{h\to 0} Re_l = 0$, the approximation of the diffusion flux is still consistent (at the limit, there is no more "correction" of the diffusion coefficient). Furthermore, the stability of the convection scheme is preserved. Here, it is important to understand that this correction does not improve the order of the approximation of the convection–diffusion operator. But, as it will be shown later with the numerical results, the accuracy of the scheme is effectively improved with this treatment. For now on, to simplify the notation, the hat over $\alpha_{K,L}$ will be dropped.

4. Projection

In order to compute a velocity field that fulfills the incompressibility constraint, a projection has to be made. It is a combination of two operators: the extension operator $\mathscr{E}$ and the projection

operator $\mathscr{P}$. The goal of the first operator is to compute an intermediate normal component of the cell velocity field. The projection operator computes both an update of the pressure and the normal component of the cell velocity field. The update of the normal velocity field is later used to correct the velocity field within the cells.

4.1. Extension operator $\mathscr{E}$

This operator is applied on each interface of the control volumes to compute an intermediate normal velocity $(\mathbf{v}\cdot\mathbf{n})_{\sigma}^{(n+1/2)}$:

$$E:\left(\mathbf{v}_K^{(n+1/2)},\mathbf{v}_K^{n}\right)\mapsto\left((\mathbf{v}\cdot\mathbf{n})_{\sigma}^{(n+1/2)}\right) \tag{19}$$

To define this operator, it is assumed that the variation of the normal velocity component agrees with the variation of the cell velocity computed in the predictor step:

$$(\mathbf{v}\cdot\mathbf{n})_{K,L}^{(n+1/2)}=(\mathbf{v}\cdot\mathbf{n})_{K,L}^{(n)}+\frac{1}{2}[\delta\mathbf{v}_K+\delta\mathbf{v}_L]\cdot\mathbf{n}_{K,L} \tag{20}$$

$$\delta\mathbf{v}=\mathbf{v}^{(n+1/2)}-\mathbf{v}^{(n)} \tag{21}$$

When $\sigma_{K,b}$ lies on the domain boundary, the intermediate normal velocity is computed with this expression:

$$(\mathbf{v}\cdot\mathbf{n})_{K,b}^{(n+1/2)}=\mathbf{v}_{K,b}^{(n)}+\delta\mathbf{v}_K\cdot\mathbf{n}_{K,b} \tag{22}$$

Hence, *the normal velocity is not directly computed at this stage, but updated*. It is worth mentioning that the boundary condition for the normal velocity must not be considered at this stage. If it were so, it would be possible to construct a non-constant velocity field for which the discrete divergence approximation is zero. In this case, the solution would exhibit spurious pressure oscillations that have no physical meaning (false pressure modes).

The application of the extension operator on the velocity field $\mathbf{v}^{(n+1/2)}$ will be denoted $(\mathbf{v}\cdot\mathbf{n})_{\sigma}^{(n+1/2)}=E\left(\mathbf{v}^{(n+1/2)},\mathbf{v}_K^{(n)}\right)$.

4.2. Projection operator $\mathscr{P}$

This operator applies to both the pressure and the velocity:

$$\mathscr{P}:\left(P_K^{(n)},(\mathbf{v}\cdot\mathbf{n})_{\sigma}^{(n+1/2)}\right)\mapsto\left(P_K^{(n+1)},\mathbf{v}_K^{(n+1)}\right) \tag{23}$$

This operation is carried out in two steps. Eq. (5) is first written under this form:

$$\mathbf{v}^{(n+1)}=-\delta t\beta\nabla\left(\delta P^{(n+1)}\right)+\mathbf{v}^{(n+1/2)} \tag{24}$$

This expression is then substituted into the continuity equation, which is then discretized as:

$$\beta\delta t\sum_{\sigma\in\mathscr{E}_K}m(\sigma)\left(\nabla\delta P^{(n+1)}\cdot\mathbf{n}\right)_{\sigma}=\sum_{\sigma\in\mathscr{E}_K}m(\sigma)(\mathbf{v}\cdot\mathbf{n})_{\sigma}^{(n+1/2)} \tag{25}$$

Eq. (25) enables us to correct the pressure field. After applying Eq. (24) to all interfaces, the equation

$$\sum_{\sigma\in\mathscr{E}_K} m(\sigma)v_\sigma^{(n+1)} = 0 \tag{26}$$

is satisfied for all control volumes.

Before solving (25), appropriate boundary conditions have to be given. When the normal velocity $v_{K,b}$ is imposed (at an inlet, a wall or on a symmetry plane), the following Neumann boundary condition holds:

$$\delta t\beta\nabla\left(\delta P^{(n+1)}\right)\cdot\mathbf{n} = v_{K,b}^{(n+1/2)} - v_{K,b} \tag{27}$$

The only other case considered is an imposed pressure (such as at an outlet). In this case, the boundary condition for the pressure correction is a Dirichlet boundary condition:

$$\delta P_{K,b}^{(n+1)} = g\left(X_{K,b}, t^{n+1}\right) - g(X_{K,b}, t^n) \tag{28}$$

where $g(X_{K,b}, t^{n+1})$ is the given pressure at time $t = t^{n+1}$ at the position $X_{K,b}$ associated to the interface $\sigma_{K,b}$ of volume K.

After updating the velocity at all interfaces, a velocity correction is also made on the cells. Let $\sigma_{K,L}$ be an interface between the cells K and L, this last correction has to be compatible with the velocity update that has been made on this interface:

$$\left(\mathbf{v}^{(n+1)} - \mathbf{v}^{(n+1/2)}\right)_K\cdot\mathbf{n}_{K,L} = W_{K,L}\left[(\mathbf{v}\cdot\mathbf{n})_{K,L}^{(n+1)} - (\mathbf{v}\cdot\mathbf{n})_{K,L}^{(n+1/2)}\right] \tag{29}$$

where $W_{K,L}$ is a parameter such that

$$W_{K,L} = \begin{cases} \text{big number : (i.e. } 10^6) & \sigma_{K,L}\in\text{wall} \\ 1, & \text{otherwise} \end{cases}$$

For each control volume K, a linear system of equations is built by applying (29) to all interfaces which belong to K. This linear system can be inconsistent and its solution is always approximated with a least square method. The parameter W has to be introduced to insure that the cell velocity always agrees with the wall boundary conditions.

Both the velocity–pressure formulation and the projection operator fall into the same class than the operators presented in [25]. Hence, the proof of the uniqueness of the solution for the pressure given in [25] also applies to this scheme.

Finally, the application of the projection operator on the velocity $(\mathbf{v}\cdot\mathbf{n})_\sigma^{(n+1/2)}$ and pressure $P_K^{(n)}$ field will be denoted as

$$\left(P_K^{(n+1)}, \mathbf{v}_K^{(n+1)}\right) = \mathscr{P}\left(P_K^{(n)}, (\mathbf{v}\cdot\mathbf{n})_\sigma^{(n+1/2)}\right)$$

This section ends with a short remark for the special case where there is no macro-element. For this type of mesh, all control volumes are tetrahedrons and it is possible to use the zero degree Raviart–Thomas finite element which contains the following polynomials [26,27]:

$$\left.\begin{array}{l} a+\mathrm{d}x \\ b+\mathrm{d}y \\ c+\mathrm{d}z \end{array}\right\} \quad \forall (a,b,c) \in \mathfrak{R}^3$$

With this family of elements, it is possible to compute a velocity field $\mathbf{v}_K$ which satisfies the following equations:

$$\mathbf{v}_K \cdot \mathbf{n}_{K,L} = -\mathbf{v}_L \cdot \mathbf{n}_{L,K}$$
$$\sum_{\sigma \in \mathscr{E}_{\mathscr{K}}} \mathbf{v}_K \cdot \mathbf{n}_\sigma = 0$$

In this case, it is possible to use an extension and a projection operators for which the normal components of the cell velocity field on each interface is continuous and satisfies the divergence free constraint [16,17,28]. But for the scheme presented in this paper, macro-elements are frequent and the normal components of the cell velocity field cannot directly be computed from the cell velocity. As a consequence, the zero degree Raviart–Thomas finite elements are not used.

5. The complete algorithm

This first part of the article ends with the presentation of the whole algorithm. This algorithm contains a start-up step and care must be given to the notation of this initial step:

- $\mathbf{v}_K^{(-1)}$ stands for an initial velocity field which is always null;
- $\mathbf{v}_K^{(-1/2)}$ is the initial condition for the velocity field;
- $P_K^{(-1)}$ is the initial condition for the pressure;
- $\phi_K^{(0)}$ is the initial condition for the scalar variables.

The algorithm is as follows:

- Given the initial condition $\mathbf{v}_K^{(-1)} = \mathbf{0}$, $\mathbf{v}_K^{(-1/2)}$, $P_K^{(-1)}$ and $\phi_K^{(0)}$, apply the extension and the projection operators:

$$(\mathbf{v} \cdot \mathbf{n})_\sigma^{(-1/2)} = E\left(\mathbf{v}_K^{(-1)}, \mathbf{v}_K^{(-1/2)}\right)$$

$$\left(P_K^{(0)}, \mathbf{v}_K^{(0)}\right) = \mathscr{P}\left(P_K^{(-1)}, (\mathbf{v} \cdot \mathbf{n})_\sigma^{(-1/2)}\right)$$

- Given a solution $\mathbf{v}^{(n)}$, $P^{(n)}$ and $\phi^{(n)}$,
  (1) apply the convection–diffusion operator to all components of the velocity vector:

$$\mathbf{v}_K^{(n+1/2)} = \mathscr{C}\mathscr{D}\left(\mathbf{v}_K^{(n)}\right)$$

  (2) apply the extension and the projection operators:

$$(\mathbf{v} \cdot \mathbf{n})_\sigma^{(n+1/2)} = E\left(\mathbf{v}_K^{(n+1/2)}, \mathbf{v}_K^{(n)}\right)$$

$$\left(P_K^{(n+1)}, \mathbf{v}_K^{(n+1)}\right) = \mathscr{P}\left(P_K^{(n)}, (\mathbf{v}\cdot\mathbf{n})_\sigma^{(n+1/2)}\right)$$

(3) apply the convection–diffusion operator to all scalar quantities:

$$\phi_K^{(n+1)} = \mathscr{C}\mathscr{D}\left(\phi_K^{(n)}\right)$$

6. Theoretical results

In this section a very brief summary of the theoretical results that apply to this scheme are presented. Those results were proven by Gallouët and coworkers [15].

6.1. Steady problems

Consider the steady problem:

$$\begin{cases} \nabla\cdot(\mathbf{v}\phi) - \nabla(\alpha\nabla\phi) = s \\ \text{Boundary conditions} \end{cases} \tag{30}$$

where $\nabla\cdot\mathbf{v} = 0$ and $s \in L^2(\Omega)$.

The properties shown in [15] depend on the quality of the triangulation. Meshes of the categories M_1, M_2 and M_4 (with macro-elements) are called "admissible meshes". For these meshes, the following properties were proven:

(1) *Convergence*. Let $\mathscr{T}$ be an admissible mesh and $\phi_{\mathscr{T}}(\mathbf{x}) = \phi_K$ for any $K \in \mathscr{T}$. $\phi_{\mathscr{T}}$ converges to the unique variational solution ϕ of problem (30) as $h \to 0$, h being the diameter of the largest volume.

(2) *Error estimate*. Let $\mathscr{T}$ be an admissible mesh and $\phi \in H^2(\overline{\Omega})$ the unique variational solution of (30). The following error estimate holds:

$$\|e_K\|_{L^2(\Omega)} \leqslant Ch \tag{31}$$

where $e_K = \phi_K - \phi(X_K)$, is the error and C is a positive constant which is independent of the mesh size h.

(3) *Maximum principle.* Let $\mathscr{T}$ be an admissible mesh and

$$s_K = \frac{1}{m(K)}\int_K s(\mathbf{x})\,\mathrm{d}\Omega$$

If $s_K \geqslant 0$ for all $K \in \mathscr{T}$ and positive Dirichlet boundary condition apply for all $\sigma \in \partial\Omega$, then the solution ϕ_K satisfies $\phi_K \geqslant 0$ for all $K \in \mathscr{T}$.

6.2. Transient problems

Consider the transient problem,

$$\begin{cases} \dfrac{\partial\phi}{\partial t} + \nabla\cdot(\mathbf{v}\phi) - \nabla(\alpha\nabla\phi) = s \\ \text{Boundary conditions} + \text{Initial condition} \end{cases} \tag{32}$$

where $\nabla \cdot \mathbf{v} = 0$ and $s \in L^2(\Omega)$. For a first order Euler time discretization, the following error estimate holds:

$$\sqrt{\sum_{K \in \mathscr{T}} (\phi(X_K) - \phi_K)^2 m(K)} \leqslant C(h + \delta t) \tag{33}$$

h being the diameter of the largest volume, δt the time step and $C > 0$ a constant independent from the time step and the mesh size h.

6.3. Remarks

For meshes of category M_3, the approximation of the flux between two control volumes is consistent. However, it is assumed that ϕ takes the same value for at least two different positions associated to a control volume. In this case, locally for some atypical edges, the order of the local approximation at the positions associated to a control volume could not be enough to ensure that the latter properties are fulfilled. Nevertheless, in [15], it is shown that even for meshes where atypical edges are found, such as meshes of category M_3 with macro-elements, the scheme still converges to the true solution if the length of the atypical edges is not too large compared with the total length of all edges.

In the report [29], the numerical rate of convergence of the diffusion operator was studied with extensive numerical experiments. These numerical results show that the observed rate of convergence of the diffusion operator can reach order 2 at the circumcenters. It is not a contradiction with the theoretical results: it shows that the theoretical rate of convergence is probably not optimal.

7. Numerical results

In this section, numerical results to validate the method are given. All computations were carried out in three dimensional domains, even for the 2D flows.

Before showing these results, a few words must be said about the resolution of the systems of discrete of equations. Lets first recall that these systems of equations are linear and the scheme does not require any non-linear solver. To speed-up the computations, the non-zero elements of the matrix of coefficients is always stored. The linear systems are solved with the Orthomin2 algorithm. Which is not as well known as the conjugate gradient method. It requires more operations (two matrix vector multiplication by iteration), but it does not require the linear system to be symmetric. The performance of this algorithm was compared to GMRES and the conjugate gradient. For the problems that were considered, Orthomin2 was a better choice than the others or the combination GMRES-CG (conjugate gradient for the symmetric systems associated with the projection operator and GMRES for the other systems). For more information on these solvers, see [30].

Finally, for the steady flows, the convergence curves of the residuals norms of the discrete equations are given. Those residuals are defined as follows. For any unknown variable ϕ, the discrete equations can be written in the following matrix form:

$$\mathbf{A}\phi = \mathbf{b}$$

Thus, the L^2 norm of residual can be defined as

$$\mathbf{r} = \mathbf{b} - \mathbf{A}\phi$$

$$\|r\|_{L^2} = \frac{\sqrt{\sum_i^N r_i^2}}{N}$$

where N is the number of equations.

7.1. Lid-driven cavity flow

This problem deals with a confined flow in a square cavity. The flow is driven by the constant displacement of the upper wall where the tangent velocity is imposed. The set-up for this problem is the following:

Domain: $[0.0, 0.1] \times [0.0, 1.0] \times [0.0, 1.0]$.

Boundary conditions: $\mathbf{v} \cdot \mathbf{n}|_{\partial\Omega} = 0$, $\mathbf{v} \cdot \boldsymbol{\tau} = \begin{cases} 1, & z = L, \\ 0, & \text{elsewhere.} \end{cases}$

Initial condition: $\mathbf{v}(\mathbf{x}) = \mathbf{0}$, $P(\mathbf{x}) = 0$.

Physical properties: $Re = \dfrac{v_{\max} L}{v} = 1000$, $v_{\max} = 1$, $L = 1.0$.

Time step: $\delta t = 0.1$.

This problem has been studied in details by Ghia et al. [31] for different Reynolds numbers. Hence, the solutions computed with this method could be compared to those obtained by others using a second order discretization scheme. This problem was solved using three different unstructured meshes made of 4015, 6783 and 10339 control volumes (a plane cut of the coarsest mesh is shown in Fig. 4). Two simulations were also carried out for each mesh: one with the power-law scheme, the other without a correction of the diffusion coefficient. The velocity

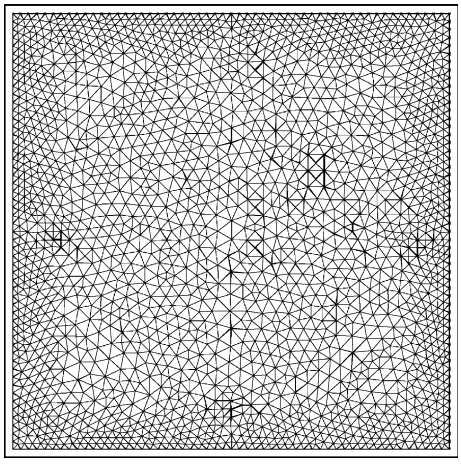

Fig. 4. Lid-driven flow: coarsest mesh (4015 cells).

components on the mid-planes were given in [31] and some local characteristics of the flow are presented in Table 1. In this table, the maxima of the velocity components and the location where they occur are presented. From these results it appears that

(1) with mesh refinement, the solutions converge toward the benchmark *n*;
(2) the quality of the results are improved with the power-law scheme.

In Fig. 5, a plane cut of the benchmark solution and the numerical solution obtains with the finest mesh and the power-law scheme are given.

Finally, the convergence curves for the computation made with the finest mesh and the power-law scheme are shown in Fig. 6.

The other results presented in the next sections were all computed using the power-law scheme.

Table 1
Lid-driven flow: comparative results for three meshes

Mesh	Power-law scheme	Min v ($z=$)	Min w ($y=$)	Max w ($y=$)
4015 volumes	No	−0.322, (0.201)	−0.460, (0.892)	0.308, (0.189)
4015 volumes	Yes	−0.359, (0.201)	−0.503, (0.914)	0.344, (0.189)
6783 volumes	No	−0.334, (0.202)	−0.473, (0.906)	0.322, (0.190)
6783 volumes	Yes	−0.368, (0.202)	−0.511, (0.906)	0.356, (0.176)
10339 volumes	No	−0.348, (0.193)	−0.496, (0.898)	0.334, (0.187)
10339 volumes	Yes	−0.379, (0.183)	−0.522, (0.906)	0.367, (0.164)
Ghia et al. 10,000 nodes	–	−0.383, (0.172)	−0.516, (0.906)	0.371, (0.156)

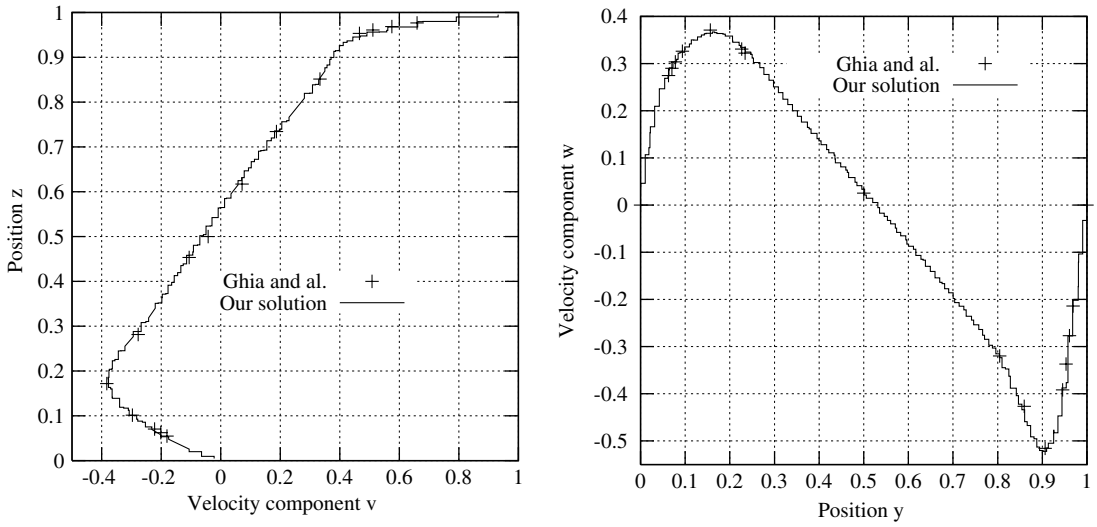

Fig. 5. Lid-driven flow: velocity components on the mid-planes $y = 1/2$ and $z = 1/2$.

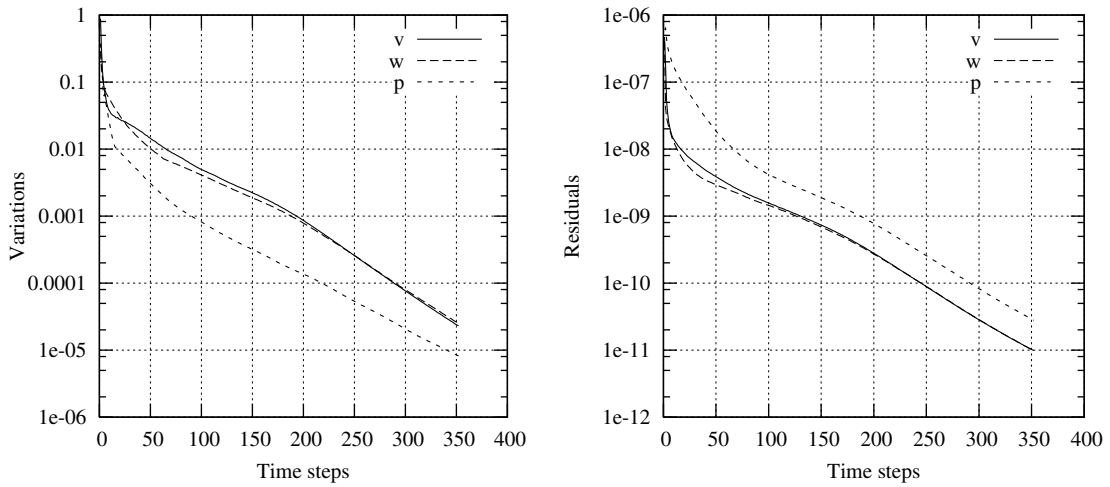

Fig. 6. Lid-driven flow: the graph on the left shows the norms of the variations $\left|\phi^{n+1}-\phi^{n}\right|_{\infty}$, the norms $\|r\|_{L^2}$ of the residuals are shown on the right.

7.2. Transient 2D flow around a cylinder

This problems deals with an internal flow between two parallel planes. A cylinder is present near the inlet and its center is slightly above the mid-section. Hence, the flow is not symmetric and lift is produced. For this Reynolds number ($Re = 100$), the flow is periodic due to vortex shedding. The setup for this problem is as follows:

Domain: see Fig. 7.
Boundary conditions:
Inlet: $v(x, y = 0, z) = \dfrac{4v_{\mathrm{m}}z(H-z)}{H^2}$, $v_{\mathrm{m}} = 1.5$, $w(x, y = 0, z) = 0$.
Outlet: $\left.\dfrac{\partial \mathbf{v}}{\partial n}\right|_{x,y,z=2.2} = 0$, $P(x, y = 2.2, z) = 0$.
Walls: $\mathbf{v}\cdot\mathbf{n} = 0$, $\mathbf{v}\cdot\boldsymbol{\tau} = 0$.
Initial condition: $\mathbf{v}(\mathbf{x}) = \mathbf{0}$, $P(\mathbf{x}) = 0$.

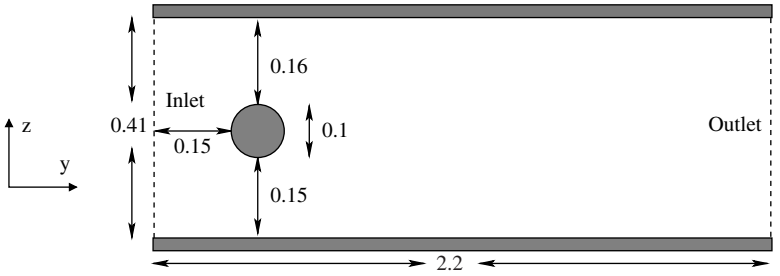

Fig. 7. 2D flow around a cylinder: geometry.

Physical properties: $Re = \frac{v_{\text{ref}} D}{\nu} = 100$, $v_{\text{ref}} = 1.0$, $D = 0.1$.

Time steps: $\delta t = \{0.008, 0.004, 0.002\}$.

For this problem, the numerical results are compared to the benchmark quantities published by Schafer and Turek [32]. In this report, the drag, lift, difference of pressure and Strouhal number are presented. Let t_0 be the time when the maximum value of lift is reached and f the frequency of vortex shedding, the following quantities are provided:

(1) Drag coefficient:

$$C_D(t_0) = \frac{2 \cdot F_D(t_0)}{\rho \cdot v_{\text{ref}}^2 \cdot D \cdot H \cdot L}$$

$$F_D(t_0) = \int_{\partial S} \left(0, \nu \frac{\partial v_\tau}{\partial n}, P\right) \cdot \boldsymbol{\tau} \, \mathrm{d}A$$

(2) Lift coefficient:

$$C_L(t_0) = \frac{2 \cdot F_L(t_0)}{\rho \cdot v_{\text{ref}}^2 \cdot D \cdot H \cdot L}$$

$$F_L(t_0) = -\int_{\partial S} \left(0, \nu \frac{\partial v_\tau}{\partial n}, P\right) \cdot \mathbf{n} \, \mathrm{d}A$$

(3) Pressure difference:

$$\Delta P\left(t_0 + \frac{1}{2f}\right) = P(0, 0.15, 0.2) - P(0, 0.25, 0.2)$$

(4) Strouhal number:

$$St = \frac{Df}{v_{\text{ref}}}$$

where

- $L = 0.10$ is the depth in the 3rd dimension (the flow being solved in a 3D domain);
- $\boldsymbol{\tau}$ is a tangent vector to the cylinder surface;
- $v_\tau = \mathbf{v} \cdot \boldsymbol{\tau}$ is the tangential speed at the cylinder surface;
- $\mathbf{n}$ is the unit normal vector to the surface cylinder;
- ∂S is the area of the cylinder.

In order to show that grid refinement and smaller time steps lead to more accurate results, the computations were on three different meshes and three different time steps on the finest grid. The benchmark quantities provided by Schafer and Turek [32] and our results are presented in Table 2. Those results show that systematic grid refinement lead to more accurate results. As for time independent results, the convergence is clear for the lift and drag coefficients. For the other

Table 2
2D flow around a cylinder: maximum drag, lift and other comparative results

δt	Cells	C_D $t = t_0$	C_L $t = t_0$	$St = \frac{D \cdot f}{v_m}$	ΔP $t = t_0 + \frac{1}{2f}$
0.002	22387	3.21	0.93	0.282	2.42
0.002	32411	3.21	0.96	0.287	2.47
0.002	46761	3.23	1.01	0.292	2.48
0.004	46761	3.24	1.02	0.294	2.49
0.008	46761	3.28	1.03	0.291	2.50
Benchmark quantities		3.22–3.24	0.99–1.01	0.295–0.305	2.46–2.50

benchmark quantities, the convergence is not so evident, the "best" Strouhal number not being obtained with the smallest time step.

When constructing the extension and projection operators, some hypotheses on the time evolution of the velocity components were made and these could have had a dramatic effect on the time accuracy of the scheme. Thereby, this problem is very important for the validation of the scheme. Fortunately, these results tend to show that these hypotheses were justified. Thus the scheme can also be considered to predict unsteady flows.

7.3. 3D flow around a cylinder

This flow is similar to the previous one. It is a 3D steady flow around a cylinder confined in a square duct (Fig. 8). Once again, a complete description of the data needed to set-up this problem is given.

Domain: see Fig. 8.
Boundary conditions:

$$\text{Inlet: } v(x, y = 0, z) = \frac{16 v_m xz(H - x)(H - z)}{H^4},\ v_m = 0.45.$$

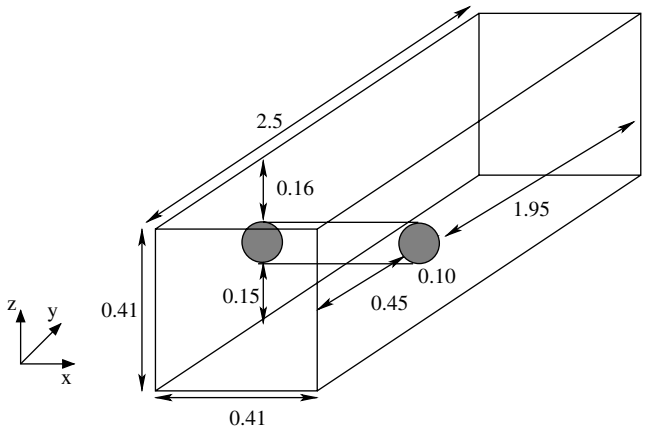

Fig. 8. 3D flow around a cylinder: geometry.

$u(x, y = 0, z) = w(x, y = 0, z) = 0.$

Outlet: $\left.\dfrac{\partial \mathbf{v}}{\partial n}\right|_{x,y=2.5,z} = 0,\ P(x, y = 2.5, z) = 0.$

Walls: $\mathbf{v} \cdot \mathbf{n} = 0,\ \mathbf{v} \cdot \boldsymbol{\tau} = 0.$

Initial condition: $\mathbf{v}(\mathbf{x}) = \mathbf{0},\ P(\mathbf{x}) = 0.$

Physical properties and dimensions: $Re = \dfrac{v_{\text{ref}} D}{\nu} = 20,\ v_{\text{ref}} = 0.2\ D = 0.10,\ H = 0.41.$

Time step: $\delta t = 0.1.$

As for the previous flow, this problem is part of a benchmark problems presented in [32]. The solutions were not provided. Instead, the lift coefficient, the drag coefficients and the pressure variation between two points were given. These quantities can be computed with the following expressions:

$$F_D = \int_{\partial S} \left(\nu \frac{\partial v_\tau}{\partial n} n_z - P n_y \right) \mathrm{d}S \tag{34}$$

$$F_L = -\int_{\partial S} \left(\nu \frac{\partial v_\tau}{\partial n} n_y + P n_z \right) \mathrm{d}S \tag{35}$$

$$C_D = \frac{2F_D}{\rho v_{\text{d}}^2 DH}, \quad C_L = \frac{2F_L}{\rho v_{\text{ref}}^2 DH} \tag{36}$$

$$\Delta P = P(0.45, 0.20, 0.205) - P(0.55, 0.20, 0.205) \tag{37}$$

S is the surface of the cylinder, $\mathbf{n}$ the normal unit vector on S and v_τ the tangent speed at the surface of the cylinder.

In order to make an extensive validation, three computations were carried out on different meshes. The 3D meshes were built by translating a 2D mesh composed of cut-quadrangles and triangles. Which gives semi-structured grids that fall into the M_3 and M_4 category of meshes. Hence, in these regions, the mesh used for the computations is composed of hexahedrons (the tetrahedrons being combined together). At the inlet, around the cylinder and at the outlet, the grid is a semi-structured mesh and the control volumes are hexahedrons. Further from the cylinder, the mesh is completely unstructured. Parts of the 2D mesh used to built the second 3D mesh are shown on Figs. 9 and 10.

In Table 3, the references quantities obtained with this method are compared to those presented by Schafer and Turek [32]. The results are in good agreement with the benchmark solutions. As for the other steady flows, the convergence curves for all variables are showed on Fig. 11.

In Table 4, some statistics relative to the performance of the computer code are provided. This computer code was written in C++ and ran on single processor PC equipped with an AMD1000 processor. [3] All the real numbers were stored in double precision format.

[3] The MFlop rate for the LINPACK1000 benchmark for this computer is 74.

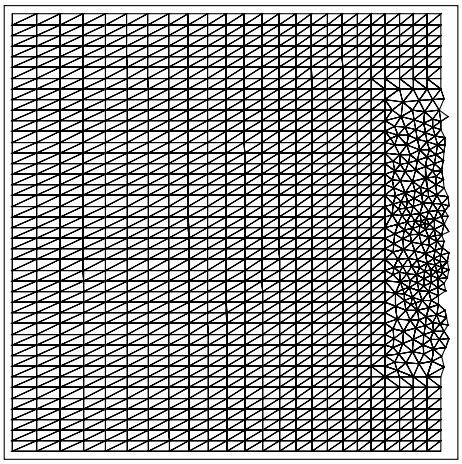

Fig. 9. 3D flow around a cylinder: mesh at the inlet.

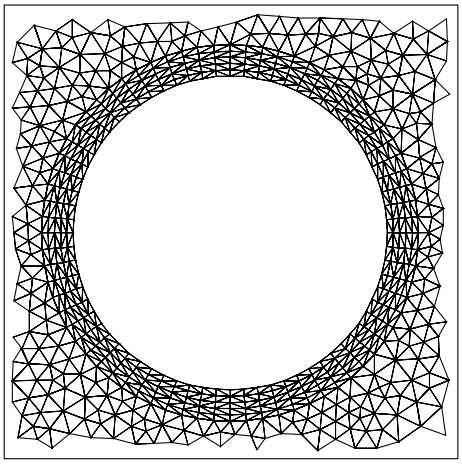

Fig. 10. 3D flow around a cylinder: mesh around the cylinder.

Table 3
3D flow around a cylinder: lift, drag and pressure difference

	Unknowns	C_L	C_D	ΔP
Mesh 1	646600	0.0078	6.19	0.170
Mesh 2	794000	0.0133	6.178	0.169
Mesh 3	1069000	0.0095	6.17	0.170
Benchmark quantities	753664 to 12582912	0.008 to 0.010	6.05 to 6.25	0.165 to 0.175

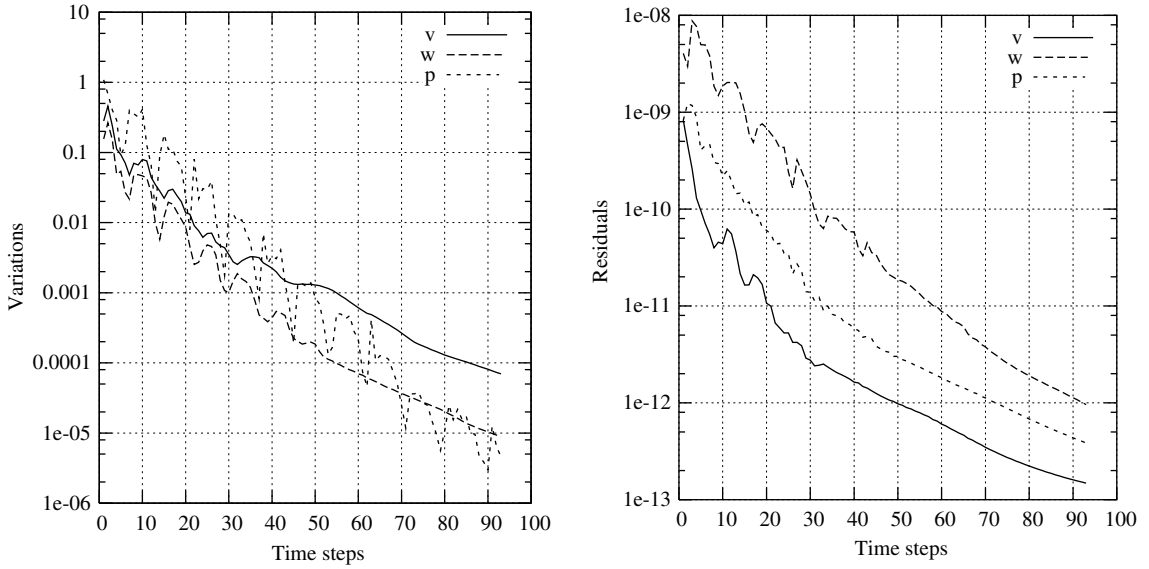

Fig. 11. 3D flow around a cylinder: the graph on the left shows the norms of the variations $\left|\phi^{n+1}-\phi^{n}\right|_{\infty}$, the norms $\|r\|_{L^2}$ of the residuals are shown on the right.

Table 4
3D flow around a cylinder, statistics

Unknowns	Iterations	Time (s)	Time per iteration	Memory usage (megabytes)
646600	102	8731	85.6	394
794000	99	10412	105.17	482
1069000	93	13599	146.23	607

7.4. 3D thermal flow in a cylinder

This problem deals with a forced thermal flow in a cylinder. The temperature is imposed both at the inlet and on the cylinder's surface. The flow is not developed at the inlet, a constant velocity being imposed at this location. All the data needed to solve this problem are presented below.

Domain: $\partial\Omega = \begin{pmatrix} x(\theta) \\ y(\theta) \\ z \end{pmatrix} = \begin{pmatrix} 0.05\cdot\cos(\theta) \\ 0.05\cdot\sin(\theta) \\ z \end{pmatrix},\ 0 \leqslant \theta \leqslant \pi/2,\ 0 \leqslant z \leqslant 1.2.$

Boundary conditions:

Inlet: $u(x,y,z=0)=0,\ v(x,y,z=0)=0,\ w(x,y,z=0)=1,\ T(x,y,z=0)=1.$

Outlet: $\left.\dfrac{\partial \mathbf{v}}{\partial n}\right|_{z=1.2}=0,\ P(x,y,z=1.2)=0,\ \left.\dfrac{\partial T}{\partial n}\right|_{z=1.2}=0.$

Walls: $\mathbf{v}\cdot\mathbf{n}=0,\ \mathbf{v}\cdot\boldsymbol{\tau}=0,\ T=0.$

Initial condition: $\mathbf{v}(\mathbf{x})=\mathbf{0},\ P(\mathbf{x})=0,\ T(\mathbf{x})=0.$

Physical properties and dimensions: $Re_D=\dfrac{v_{\mathrm{ref}}D}{v}=120,\ w_{\mathrm{ref}}=1.0,\ Pr=\dfrac{vc_p}{k}=1.0,\ D=0.1,\ c_p=1.0.$

Time step: $\delta t=0.1.$

w_{ref} is the average of the speed for a section of the duct and D the diameter. This flow being symmetric, the computations were carried out only on one quarter of the domain. The 3D mesh was built using the extrusion of a 2D mesh composed of triangles.

For this type of flow, there is an analytical solution in the region of the domain where the flow is fully developed. For a laminar flow, $Re_D \leqslant 2300$, the location at which the flow starts to be fully developed is approximated as follows:

$$\left(\frac{L}{D}\right)_{\text{lam}} \approx 0.05 Re_D$$

In this region, the velocity component w and the pressure gradient can be computed from these equations:

$$w\left(r = \sqrt{x^2 + y^2}\right) = -\frac{1}{4\mu}\frac{\partial P}{\partial z}\left(\frac{D}{2}\right)^2\left(1 - \left(\frac{r}{D/2}\right)^2\right) \tag{38}$$

$$\frac{\partial P}{\partial z} = -\frac{8 \cdot \mu \cdot v_{\text{ref}}}{(D/2)^2} \tag{39}$$

As for the temperature, the length at which the flow is thermally developed is given by the empirical expression:

$$\left(\frac{L}{D}\right)_{\text{lam},T} \approx 0.05 Re_D \cdot Pr$$

In this region, the exact solution for the temperature is not known. But in this part of the domain, there is no variation along the cylinder of the dimensionless temperature:

$$\frac{\partial}{\partial z}\left(\frac{T_s - T(\mathbf{x})}{T_s - T_{\text{m}}(z)}\right) = 0 \tag{40}$$

$T_{\text{m}}(z)$ being the mean axial temperature in a given section. For each cross-section area, this mean axial temperature is often called mixed mean fluid temperature and is defined as [33]:

$$T_{\text{m}} = \frac{1}{A_{\text{c}} w_{\text{a}}}\int_{A_{\text{c}}} w(\mathbf{x}) T(\mathbf{x})\, \mathrm{d}S$$

A_{c} being the area of a cross-section and w_{a} the average speed.

There is another important result for thermally developed flow in a circular tube: it can be shown that the Nusselt number at the surface is constant [33]:

$$Nu_D = 3.657$$

For the computation of Nu_D, the length of reference is $L_{\text{ref}} = D = 0.1$ and the reference temperature is the mean axial temperature.

The graph in Fig. 12, shows that there is no significant difference between the computed velocity component w and the analytical solution given by equation (38). In Figs. 13

and 14, pressure component $\partial P/\partial z$ and the adimensional temperature are plotted along the curves

$$\mathbf{X}(z) = \begin{pmatrix} r \cdot \cos(45°) \\ r \cdot \sin(45°) \\ z \end{pmatrix}, \quad r = \{0.005, 0.02, 0.04\}, \ 0 \leqslant z \leqslant 1.2$$

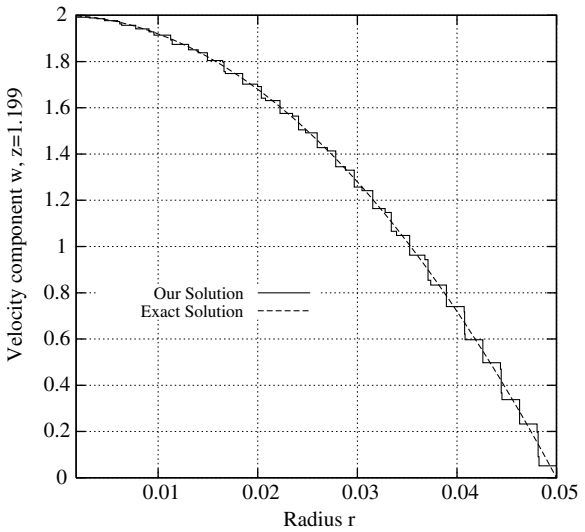

Fig. 12. Thermal flow in a cylinder: velocity profile near the outlet.

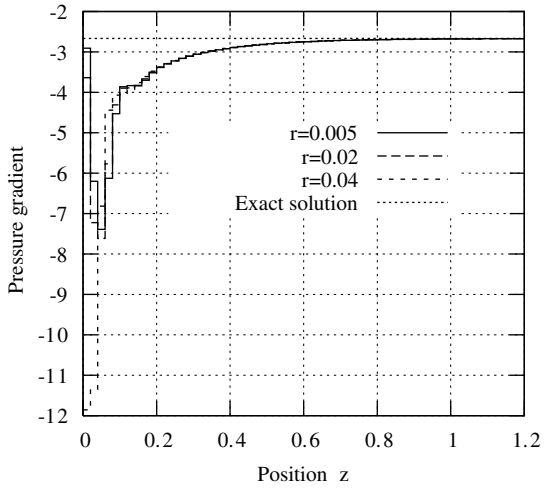

Fig. 13. Thermal flow in a cylinder: pressure gradient along the cylinder, component $\partial P/\partial z$.

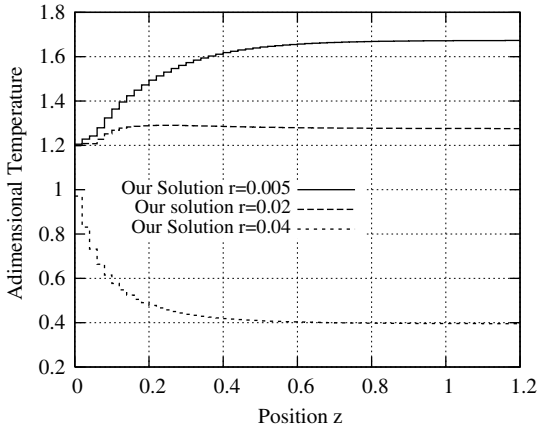

Fig. 14. Thermal flow in a cylinder: adimensional temperature profile along the cylinder.

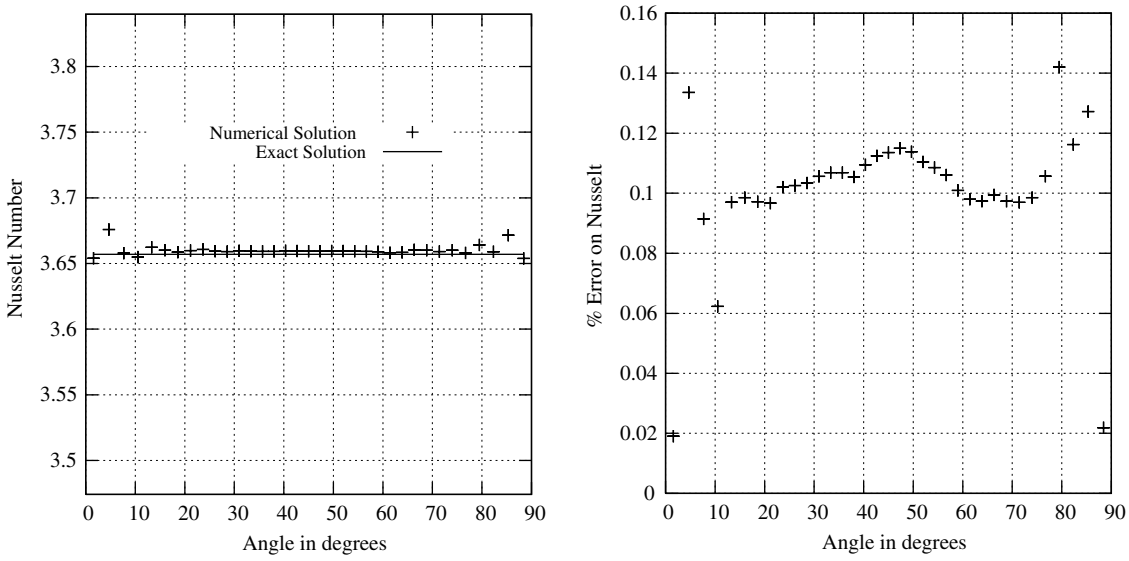

Fig. 15. Thermal flow in a cylinder: the graph on the left compares the computed Nusselt number near the outlet to its exact value, percentage of error is shown on the right.

These plots show that the pressure gradient and the temperature behave accordingly with the theoretical solution. The pressure gradient becomes constant near the outlet and the adimensional temperature also becomes constant for a given radius. Finally, the Nusselt number at the outer surface of the cylinder near its outlet was computed. Those values are given in Fig. 15 with the percentage of error at each location.

As for the other stationary flows, the convergence curves are given in Fig. 16.

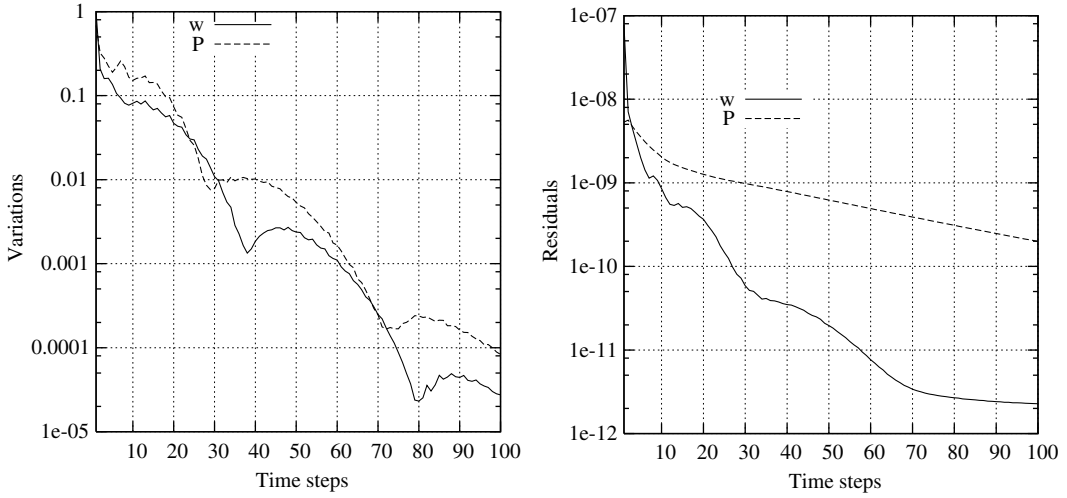

Fig. 16. Thermal flow in a cylinder: the graph on the left shows the norms of the variations $\left|\phi^{n+1}-\phi^{n}\right|_{\infty}$, the norms $\|r\|_{L^2}$ of the residuals are shown on the right.

8. Conclusion

A numerical method has been proposed to solve the 3D Navier–Stokes equations for incompressible viscous flows and the convection–diffusion of scalar quantities. This solver is based upon a fractional step method and a cell centered finite volume scheme for unstructured tetrahedral meshes. This solver allows the local conservation of mass and scalar quantities and the numerical preservation of the maximum principle for scalar quantities on general meshes do not necessarily satisfy the Delaunay condition.

The convection–diffusion scheme is very robust and easy to implement. By choosing the circumcenters as the reference positions, a consistent approximation of the normal gradients to the cell interfaces can be made with a simple differential quotient. Being of order one space accurate, it is a low order scheme. Nevertheless, it was showed that by introducing an efficient correction of the diffusivity coefficient (the power-lay scheme), more accurate solutions could be obtained. The overall accuracy of the scheme could also be improved with a more sophisticated computation of the convective terms based on a MUSCL type reconstruction.

In order to preserve the maximum principle, tetrahedrons that do not respect the Delaunay condition are combined together to form new elements that are called macro-elements. In its present form, the scheme is not optimal for meshes where many elements have high-aspect ratios. Thereby, the extension to Navier–Stokes compressible flows where shocks need adapted meshes aligned with contact discontinuities and shocks is not straightforward.

Numerical solutions for laminar steady and unsteady flows were presented. For all cases, the solutions computed with this scheme were in good agreement with those presented by other researchers or exact solutions. For all stationary flows, the convergence curves for both the variations and the residuals were given. This scheme is under active development, numerical results for a turbulent flow using the k–ϵ model have already been presented in [34,35]. Current

developments of the scheme include: a second order approximation of the convection term, more exhaustive computations of turbulent flows with heat transfer and an extension to compressible flows.

Appendix A. Boundary conditions

Let $\sigma_{K,b}$ be a boundary interface which is a face of volume K. When the boundary condition is a Neumann boundary condition, the numerical diffusion flux $F^D_{K,b}$ is

$$F^D_{K,b} = \int_{K,b} f^D(\mathbf{x})\,\mathrm{d}S \tag{A.1}$$

f^D being the given flux by unit area. Let $X_{K,b}$ be the intersection of the orthogonal bisectors of the interface $\sigma_{K,b}$. When a Dirichlet boundary condition $g(\mathbf{x})$ applies, the value $\phi_{K,b} = g(X_{K,b})$ is imposed at the interface. In this case, the numerical diffusion flux is

$$F^D_{K,b} = \alpha_{K,b}\tau_{K,b}(\phi_{K,b} - \phi_K) \tag{A.2}$$

As for the convective flux, the value of the variable $\sigma_{K,b}$ at the interface is only needed when the fluid is incoming ($v_{K,b} < 0$). The convective flux is given by

$$F^C_{K,b} = m(\sigma_{K,b})v_{K,b}\phi_{K,b}$$

where $\phi_{K,b} = g(\mathbf{x}_{K,b})$.

For meshes of category M_3, it is possible that for at least on boundary interface $\sigma_{K,b}$, $\tau_{K,b} < 0$. Then, there exist $X_K \notin \Omega$, where Ω is the computational domain. Hence there exist at least on position associated to a control volume that is outside the computational domain and the function being approximated can be undefined at such location. In the report [29], numerical results show that for diffusion problems, the observed rate of convergence is second order for regular function and of order one when Dirac functions are considered. For convection–diffusion problems, numerical results have shown poor convergence behavior when negative transmittivities were found on boundaries where Dirichlet boundary conditions apply. In order to enhance the convergence behavior, we propose a simple treatment that can be easily implemented.

Let ϕ be a scalar variable such has a concentration or a velocity vector component, $\sigma_{K,b}$ an interface of volume K for which $\tau_{K,b} < 0$ and the Dirichlet boundary condition $\phi(\mathbf{x}) = g(\mathbf{x})$ is provided. The cell value ϕ_K is imposed equal to the boundary value: $\phi_K = g(X_{K,b})$, $X_{K,b}$ being the position associated to the interface $\sigma_{K,b}$. Even though this approximation has shown to be satisfactory for practical cases (such has the lid-driven flow in Section 7.1), we must say that locally the order of the approximation for the scalar variable ϕ could be insufficient to ensure global convergence of the scheme toward the true solution.

References

[1] Harlow FH, Welch JE. Numerical calculation of time-dependant viscous incompressible flow of fluid with a free surface. Phys fluids 1965;8:2182–9.

[2] Patankar SV. Numerical heat transfer and fluid flow. Hemisphere Publishing Corporation; 1996.
[3] Hwang YH. Calculations of incompressible flow on a staggered triangle grid, part I: mathematical formulation. Num Heat Transfer B 1995;27:323–1995.
[4] Koshizuka S, Oka Y. Moving particle semi-implicit method for fragmentation of incompressible fluid. Nucl Eng Sci 1996;123:421–34.
[5] Perot JB. Conservation properties of unstructured staggered mesh schemes. J Comput Phys 2000;159:58–89.
[6] Chow WL, Rhie CM. Numerical study of the turbulent flow past an airfoil with trailing edge separation. AIAA J 1983;21:1525–32.
[7] Miller TF, Griffiths DF. Use of a pressure-weighted interpolation method for the solution of the incompressible Navier–Stokes equations on a nonstaggered grid system. Num Heat Transfer 1988;14:213–33.
[8] Davidson L. A pressure correction method for unstructured meshes with arbitrary control volumes. Int J Numer Meth Fluids 1996;22:265–81.
[9] Foy B, Dawes W. Unstructured pressure-correction solver based on a consistent discretization of the Poisson equation. Int J Numer Meth Fluids 2000;34:463–78.
[10] Lien FS. A pressure-based unstructured grid method for all-speed flows. Int J Numer Meth Fluids 2000;33:355–74.
[11] Chorin AJ. Numerical solution of the Navier–Stokes equations. Math Comput 1968;22:745–62.
[12] Shen J. On error estimates on projection methods for Navier–Stokes equations: first order schemes. SIAM J Numer Anal 1992;29:55–77.
[13] Shen J. On error estimates on projection methods for the Navier–Stokes equations: second order schemes. Math Comput 1996;65:1039–65.
[14] Teman R. Sur l'approximation de la solution des equations de Navier–Stokes par la méthode des pas fractionnaires II. Arch Rat Mech Anal 1969;33:377–85.
[15] Eymard R, Gallouet T, Herbin R. In: Ciarlet PG, Lions JL, editors. Handbook of numerical analysis, vol. 7. North-Holland; 2000. p. 713–1020.
[16] Boivin S, Cayré F, Hérard JM. A finite volume method to solve the Navier–Stokes equations for incompressible flows on unstructured meshes. Int J Thermal Sci 2000;39:806–25.
[17] Boivin S, Cayré F, Hérard JM. Un schéma Volumes Finis pour la simulation d'écoulements diphasiques sur maillages triangulaires. Int J Finite Elements 2001;10(5):539–74.
[18] Lopez JM, Shen J. Numerical simulation of incompressible flows in cylindrical geometries using a spectral projection method. Int J Appl Sci Comput 1998;5:35–40.
[19] Faille I. A control volume method to solve an elliptic equation on a 2D irregular meshing. Comp Meth Appl Mech Engrg 1992;100:275–90.
[20] Barth TJ. Recent developments in high order K-exact reconstruction on unstructured meshes. AIAA-93-0668, January 1993.
[21] Coirier WJ. An adaptively-refined, Cartesian, cell-based scheme for the Euler and Navier–Stokes Equations. NASA TM-106754, 1994.
[22] Haselbacher A, Blasek J. On the accurate and efficient discretisation of the Navier–Stokes equations on mixed grids. AIAA J 2000;38:2094–102.
[23] Mavriplis DJ. Three-dimensional multigrid Reynolds-averaged Navier–Stokes solver unstructured meshes. AIAA J 1995;33:445–53.
[24] Davroux A, Archambeau F, Hérard JM. Tests numériques sur quelques méthodes de résolution d'une équation de diffusion en volumes finis. Technical Report HI-83/00/027/A. Électricité de France, Direction des Études et Recherche, 2000 (Unpublished).
[25] Boivin S, Hérard JM. Un schéma de Volumes finis pour résoudre les équations de Navier–Stokes sur une triangulation. Revue Européenne des Éléments Finis 1996;5:461–90.
[26] Brezzi F, Fortin M. Mixed and finite element methods. Springer-Verlag; 1991.
[27] Girault V, Raviart PA. Finite element methods for Navier–Stokes equations. Spinger-Verlag; 1986.
[28] Cayré F. Méthodes de volumes finis pour maillages non structutés pour la simulation numérique des coulements incompressibles monophasiques et diphasiques. Master Thesis. Université Laval, Québec, 1999.

[29] Cayré F. Schémas volumes finis pour un problème elliptique sur un maillage triangulaire, étude numérique de convergence. Technical Report HE-41/97/057/A Électricité de France, Direction des Études et Recherche 1997 (Unpublished).
[30] Greenbaum A. Iterative methods for solving linear systems. SIAM 1997.
[31] Ghia U, Ghia KN, Shin CT. High-*Re* solutions for incompressible flow using the Navier–Stokes equations and a multigrid method. J Comput Phys 1982;48:387–411.
[32] Schafer M, Turek S. Benchmark computations of laminar Flow around cylinder. Proceedings DFG Priority Research Program, Flow Simulation on High Performance Computers, Vieweg, 1992.
[33] Kays WM, Crawford ME. Convective heat and mass transfer. New York: McGraw-Hill; 1980.
[34] Boivin S, Hérard JM, Perron S. A finite volume method to solve the Navier–Stokes equations for incompressible flows on unstructured meshes. In: Schneider GE, editor. Proceedings, 9th Annual Conference of the CFD Society of Canada; 2001. p. 56–62.
[35] Perron S. Résolution numérique d'écoulements 3 dimensions avec une nouvelle méthode de volumes finis pour maillages non structurés. PhD thesis, Université du Québec à Chicoutimi, Chicoutimi, 2001.