Available online at www.sciencedirect.com
SCIENCE DIRECT®
International Journal of Thermal Sciences 43 (2004) 833–848
International Journal of Thermal Sciences
www.elsevier.com/locate/ijts

A new finite volume discretization scheme to solve 3D incompressible thermal flows on unstructured meshes

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

[a] *ARDC, Alcan, Applied Science Research Group, 1955 Mellon Blvd, P.O. Box 1250 Jonquiere, Quebec G7S 4K8, Canada*
[b] *Université du Québec à Chicoutimi, 555 Boulevard de l'université, Chicoutimi, Québec G7H 2B1, Canada*
[c] *DRD, Électricité de France, 6, quai Watier 78400, Chatou, France*

Received 24 October 2003; accepted 1 December 2003

Available online 13 February 2004

Abstract

We present a new method to solve incompressible thermal flows and the transport of scalar quantities. It is a finite volume scheme for unstructured meshes whose time discretization is based upon the fractional time step method. The governing equations are discretized using a collocated, cell-centered arrangement of velocity and pressure. The solution variables are stored at the cell-circumcenters. This scheme is convergent, stable and allows computing solutions that does not violate the maximum principle when it applies. Theoretical results and numerical properties of the scheme are provided. Predictions of Boussinesq fluid flow, flow past a cylinder and heat transport in a cylinder are performed to validate the method.

Keywords: Navier–Stokes; Incompressible flow; Finite volume method; Fractional time step; Projection scheme; Unstructured mesh; Boussinesq fluid flow; Heat transfer prediction

1. Introduction, mathematical model

In this paper we consider the following mathematical model. When the density of the fluid is constant, the flow in a domain Ω is governed by the Navier–Stokes equations:

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

$$\frac{\partial \mathbf{v}}{\partial t} + \nabla \cdot (\mathbf{v} \otimes \mathbf{v}) + \nabla P - \nabla(\nu \nabla \mathbf{v}) = \mathbf{f} \tag{2}$$

where

- $P = p/\rho$, P being the kinematic pressure, ρ the density and p the pressure;
- $\nu = \mu/\rho$, ν being the kinematic viscosity and μ the dynamic viscosity;
- $\mathbf{f}$ is a source term such as buoyancy force.

When the specific heat c_p of the fluid is considered constant,[2] a model for the transport of heat by the fluid is:

$$\frac{\partial T}{\partial t} + \nabla \cdot (\mathbf{v}T) - \nabla(\alpha \nabla T) = s \tag{3}$$

where $\alpha = \frac{k}{c_p \rho}$ is the thermal diffusivity, k the thermal conductivity 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 applied to many industrial problems that deal with the transport of a scalar quantity by an incompressible flow. Even though many stable and globally convergent schemes are already available for specific cases such as Euler flows or Stokes flows, only a few fulfill the following physical principles for more general problems:

- local conservation of mass and scalar quantities;
- numerical preservation of the maximum principle for the scalar quantities.

* Correponding author
E-mail addresses: sebastien.perron/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.

[2] In general, the specific heat can be variable. We make this hypothesis to ease the presentation of the scheme.

1290-0729/$ – see front matter
doi:10.1016/j.ijthermalsci.2003.12.002

Nomenclature

c_p	specific heat	$J{\cdot}kg^{-1}{\cdot}K^{-1}$	
$\mathcal{E}_K$	set of interfaces surrounding volume K		
$\mathcal{E}_{\text{ext}}$	set of interfaces surrounding the domain		
$\mathcal{E}_{\text{int}}$	set of interfaces inside the domain		
$m(\sigma)$	measure of a general hyperplane σ		
$\mathbf{n}$	unit normal vector		
p	pressure	$kg{\cdot}m^{-1}{\cdot}s^{-2}$	
P	kinematic pressure	$m^2{\cdot}s^2$	
s	general source term		
t	time	s	
T	temperature	K	
v	speed	$m{\cdot}s^{-1}$	
$\mathbf{v}$	velocity vector	$m{\cdot}s^{-1}$	
$v_{K,L}$	speed normal to the interface of volumes K and L	$m{\cdot}s^{-1}$	
$\mathbf{x}$	coordinate	m	
X_K	position associated to an element K	m	

Greek symbols

α	diffusivity	$m^2{\cdot}s^{-1}$
μ	dynamic viscosity	$kg{\cdot}m^{-1}{\cdot}s^{-1}$
ν	kinematic viscosity	$m^2{\cdot}s^{-1}$
Ω	computational domain	
ϕ	general scalar variable	
ρ	density	$kg{\cdot}m^{-3}$
$\sigma_{K,L}$	interface between elements (or volumes) K and L	
$\tau_{K,L}$	transmittivity of interface $\sigma_{K,L}$	m

Some numerical schemes using the cell centered finite volume method on structured meshes satisfy these requirements. However, only a few theoretical results are available for unstructured triangular or tetrahedral meshes (see Gallouët et al. [1]). The scheme presented hereafter fulfills the previous requirements.

There are few cell centered finite volume methods that enable the use of unstructured meshes and accurate computations of the diffusion flux on non-orthogonal grids. 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. Let ϕ be a scalar variable,[3] K and L be adjacent control volumes whose centers are X_K and X_L, many computer codes use the following approximation of the normal gradient to the cells interface:

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

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

$$\nabla\phi\cdot\mathbf{n_K} = \frac{\phi_L - \phi_K}{(X_K - X_L)\cdot\mathbf{n}_K}$$

does not lead to a consistent approximation of the flux. Furthermore, there here is also a singularity when $(Xc_K - Xc_L)\cdot\mathbf{n}_K$ 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. Moreover, it is important to point out that when the approximation of the diffusion flux is not consistent, the error does not vanish with grid refinement (see [2] for more details).

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 [3,4]. However, it can lead to a wider stencil with a less favorable weight distribution of the coefficients [5,6]. In this latter case, the discretization of the Laplacian can be nonpositive. Moreover, since the approximation of $\nabla\phi\cdot\mathbf{n}_K$ on $K|L$ is not necessarily equal to the approximation $-\nabla\phi\cdot\mathbf{n}_L$ in the cell L, this approach raises another difficulty about the conservativity of the diffusion flux.

In this paper, we propose a novel approach where the solution variables are located at the cell-circumcenters. This allows computing a consistent approximation of the viscous fluxes on non-orthogonal meshes without gradient reconstruction. Which leads to discretization of the Laplacian that is always positive. With an appropriate discretization of the convective fluxes, this scheme is convergent, stable and allows computing solutions that does violate the maximum principle when it applies.

First of all, the time discretization of the Navier–Stokes equations. will be presented. This time discretization is based upon the fractional time step method (or projection method) constructed in the late 60s by Chorin [7] and Teman [8].

Second of all, the spatial discretization of the governing equations will be detailed and some theoretical results will be provided. This spatial discretization is an application of recent theoretical results on finite volume methods published by Gallouët et al. [1] and a generalization of the recent work of Boivin et al. [9,10]. In these papers, Boivin et al. proposed a finite volume scheme for 2D 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 is not always a tetrahedron but can be an assembly of tetrahedras. This section covers:

[3] In this paper, when an equation or relation applies to any scalar variable (the temperature, a concentration, a component of the velocity vector), we use the variable ϕ.

- the spatial discretization of the convective and diffusive fluxes,
- the discrete interpolation of scalar variables and the diffusivity coefficients at cell interfaces,
- the approximation of the cell gradient of a scalar,
- application of boundary conditions.

Afterward, we will discuss the projection scheme used to compute a divergence-free vector field and update the pressure field. Finally, we present the whole algorithm used to solve the Navier–Stokes equations on unstructured grids.

In the second part of the article, we present some numerical results that were used to validate the present finite volume scheme. These results include: Boussinesq flow in cavity, permanent flow past a cylinder and heat transport in a cylinder. For all test cases, convergence history toward steady states are shown and computations on several grids are performed.

2. Time discretization

The time discretization is semi-implicit and based upon a variation of the projection scheme originally proposed by Chorin [7] and Teman [8] and often called "projection-2" scheme. Let,

$$\left.\frac{\partial \phi}{\partial t}\right|_{t=t_{n+1}} = \frac{\phi^{n+1}-\phi^{n}}{\delta t} \tag{4}$$

be the approximation of the temporal derivative for a scalar variable ϕ and

$$\mathbf{v}(t=t_{n+1}) = \frac{3}{2}\mathbf{v}^{n} - \frac{1}{2}\mathbf{v}^{n-1} \tag{5}$$

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

- Prediction:

$$\left.\frac{\partial \mathbf{v}}{\partial t}\right|_{t=t_{n+1}} + \nabla\cdot\left[\mathbf{v}(t=t_{n+1})\otimes\mathbf{v}^{n+1/2} - \nu\nabla\mathbf{v}^{n+1/2}\right] + \nabla P^{n} = \mathbf{f}^{n} \tag{6}$$

- Projection:

$$\frac{\mathbf{v}^{n+1}-\mathbf{v}^{n+1/2}}{\delta t} = -\beta\nabla\left(\delta P^{n+1}\right) \tag{7}$$

$$\delta P^{n+1} = P^{n+1} - P^{n}$$

$$\nabla\cdot\mathbf{v}^{n+1} = 0 \tag{8}$$

- Convection and diffusion of other scalar variables such as the temperature:

$$\left.\frac{\partial T}{\partial t}\right|_{t_{n+1}} + \nabla\cdot\left[\mathbf{v}(t=t_{n+1})T^{n+1} - \alpha\nabla T^{n+1}\right] = s^{n} \tag{9}$$

The parameter $\beta \in]0, 2]$ is 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. We 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 computations shown in this article. When transient flows are considered, a second order time discretization is more appropriate and β must be chosen accordingly. Since only permanent flows are considered in paper, we will not give more information on this matter. But the reader is invited to read some papers published by Shen et al. on projection schemes [11–13] and the references [14,15] where transient flows are solved with the scheme presented in this paper.

3. Space discretization

The convection and diffusion schemes are presented thereafter. The convection scheme is simple and robust. The diffusion scheme is based upon theoretical results recently published by Gallouët et al. [1]. A numerical study of this diffusion scheme was performed in the report [17] and some of those results were also published in [9]. Finally, this numerical scheme has given satisfactory results for the resolution of incompressible two-phase flows in a 2D frame (see [10]).

3.1. Geometrical elements

The initial mesh is built with tetrahedras. For each tetrahedron K, let X_K be its circumcenter. This point is allowed to be inside K, outside K, or on its boundary. Let $\sigma_{K,L}$ be the interface between tetrahedras K and L, X_K and X_L be the circumcenters of tetrahedras K and L. The straight line going through these points is always perpendicular to the cell interface $\sigma_{K,L}$ (Fig. 1). Hence, by storing the variables at these locations, we can compute a consistent approximation of the diffusive flux with a minimal stencil. For this scheme, the circumcenters will be the reference positions associated to the control volumes.

3.2. Control volumes and approximation functions

Let $\sigma_{K,L}$ be the interface between tetrahedras K and L and $\mathbf{n}_{K,L}$ the outward normal unit vector of the interface $\sigma_{K,L}$ of K. We then introduce the following quantity called "transmittivity" (see [1]):

$$\tau_{K,L} = \frac{m(\sigma_{K,L})}{(X_L - X_K)\cdot\mathbf{n}_{K,L}} \tag{10}$$

where $m(\sigma_{K,L})$ is the area of $\sigma_{K,L}$. All meshes can fall into one these four categories of meshes:

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

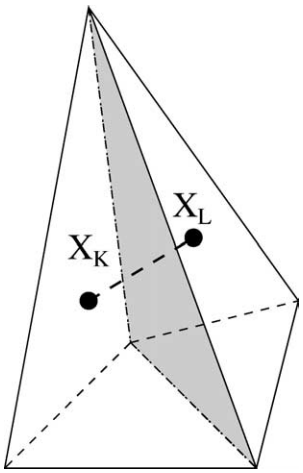

Fig. 1. Circumcenters for tetrahedras K and L.

- 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 tetrahedras.
- Category M_3: For at least one interface $\sigma_{K,L}$, $\tau_{K,L} < 0$, the Delaunay condition is not fulfilled. The tetrahedras 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 lie at the same location. 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 $X_K = X_L$. Meshes of categories M_1 and M_2 fulfill the Delaunay condition.

For categories M_3 and M_4, tetrahedras 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 to 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. The convection scheme

We start with the approximation of the convective term

$$\int \nabla \cdot (\phi \mathbf{v}) \, \mathrm{d}\Omega \tag{11}$$

Its discrete form is obtained by performing an integration on a cell K and applying the divergence theorem. When the flux is approximated with an upwind scheme, the integral at an interface $\sigma_{K/L}$ is given by the following expression:

$$\int_{K/L} \big(\mathbf{v}\phi(\mathbf{x})\big) \cdot \mathbf{n}(\mathbf{x}) \, \mathrm{d}S \approx m(\sigma_{K,L}) v_{K,L} \phi_{K,+} \tag{12}$$

where:

- $v_{K,L}$ is an approximation of the speed normal to the interface $\sigma_{K,L}$, $\mathbf{n}(\mathbf{x})$ is the outward normal unit vector;
- $\phi_{K,+} = \begin{cases} \phi_K, & v_{K,L} \geqslant 0 \\ \phi_L, & \text{otherwise} \end{cases}$

ϕ_K and ϕ_L being the unknowns associated to the volumes K and L.

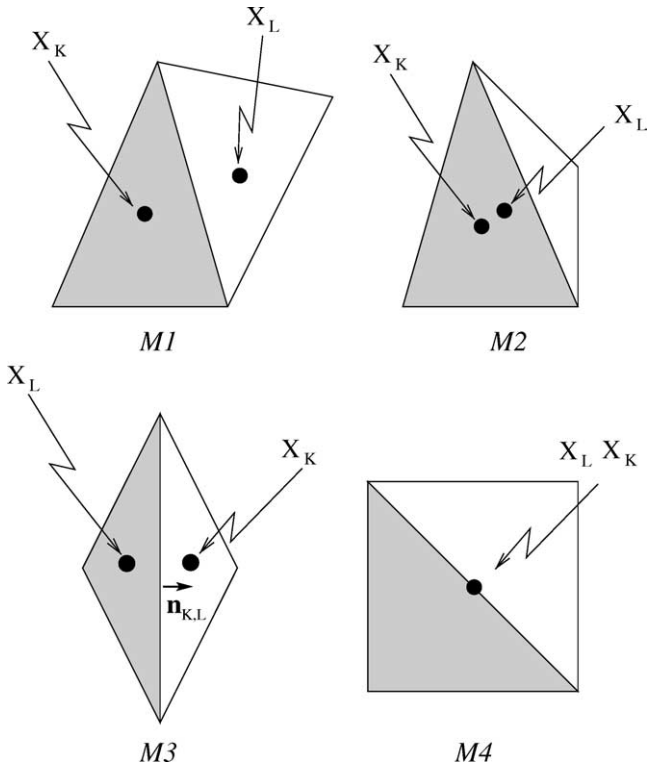

Fig. 2. Mesh categories.

3.4. The diffusion scheme

The approximation of the diffusion term

$$\int \nabla \cdot (\alpha \nabla \phi) \, \mathrm{d}\Omega \tag{13}$$

is quite similar to the convective term. We first integrate it on a cell K and use the divergence theorem to obtain the integral of the diffusion flux between volumes K and L. Which gives the following expression:

$$\int_{K/L} \alpha \nabla \phi(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x}) \, \mathrm{d}S \approx \alpha_{K,L} \tau_{K,L} (\phi_L - \phi_K) \tag{14}$$

where $\alpha_{K,L}$ is a discrete approximation of the diffusivity $\alpha(\mathbf{x})$ at the interface $\sigma_{K,L}$, $\tau_{K,L}$ is the transmittivity of the

interface $\sigma_{K,L}$ (see Eq. (10)). In this paper, for sake of simplicity, the diffusivity $\alpha(\mathbf{x})$ will be considered constant ($\alpha(\mathbf{x}) = \alpha$). For problems where diffusivity is variable, see [15,16].

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, we propose a correction on the diffusion coefficient. This correction is based upon the power law scheme introduced by Patankar [28]. Let $\alpha_{K,L}$ be an approximation of the diffusion coefficient at the interface $\sigma_{K,L}$, it is corrected as follows:

$$\widehat{\alpha}_{K,L} = \alpha_{K,L} \cdot \max\bigl(0, (1 - 0.1Pe_l)^5\bigr) \tag{15}$$

where $Pe_l = \frac{|v_{K,L}|\|X_K - X_L\|}{\alpha_{K,L}}$ is called the "local Peclet number". As the mesh size tends to zero, there is no more "correction" of the diffusion coefficient ($\lim_{h\to 0} Pe_l = 0$) and the approximation of the diffusion flux stays consistent. 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 scheme. But, as it will be shown later with the numerical results, it does improve the accuracy of the scheme.

For now on, to simplify the notation, we will drop the hat and only keep $\alpha_{K,L}$.

3.6. Boundary conditions

Let $\sigma_{K,b}$ be a boundary interface which is a face of volume K and $X_{K,b}$ the intersection of the orthogonal bisectors of this interface. When a Neumann boundary condition applies, the numerical diffusion flux is equal to the exact flux.

When a Dirichlet boundary condition applies, the value $\phi_{K,b} = \phi(X_{K,b})$ is imposed at the interface. In this case, the numerical diffusion flux is given by the following expression

$$\int_{\sigma_{K,b}} \alpha \nabla\phi(\mathbf{x}) \cdot \mathbf{n}(\mathbf{x})\,\mathrm{d}S \approx \alpha_{K,b}\tau_{K,b}(\phi_{K,b} - \phi_K) \tag{16}$$

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 then given by:

$$\int_{\sigma_{K,b}} \bigl(\mathbf{v}\phi(\mathbf{x})\bigr) \cdot \mathbf{n}(\mathbf{x})\,\mathrm{d}S \approx m(\sigma_{K,b}) v_{K,b}\phi_{K,b} \tag{17}$$

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 $X_K \notin \Omega$, Ω being the computational domain. For this configuration, 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 [17], numerical results show that for diffusion problems, the observed rate of convergence of the diffusion operator is second order for regular functions and of order one when Dirac functions are considered. Nevertheless, when negative transmittivities were located on boundaries where Dirichlet boundary conditions are applied, we found that for convection–diffusion problems the scheme showed poor convergence behavior. 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, 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.

3.7. Theoretical results

In this section we present a very brief summary of the theoretical results that were proven in [1].

3.7.1. Steady problems

Consider the steady problem:

$$\begin{cases} \nabla \cdot (\mathbf{v}\phi) - \nabla(\alpha\nabla\phi) = s \\ \text{Boundary conditions} \end{cases} \tag{18}$$

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

The properties shown in [1] 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 $\mathcal{T}$ be an admissible mesh and $\phi_{\mathcal{T}}(\mathbf{x}) = \phi_K$ for any $K \in \mathcal{T}$. $\phi_{\mathcal{T}}$ converges to the unique variational solution ϕ of problem (18) as $h \to 0$, h being the diameter of the largest volume.

(2) *Error estimate*. Let $\mathcal{T}$ be an admissible mesh and $\phi \in H^2(\overline{\Omega})$ the unique variational solution of (18). The following error estimate holds:

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

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

In theory, those properties insure that:

- Systematic grid refinement enables the computation of a solution that is globally more accurate.
- A converged solution does not violate the discrete maximum when it applies. Hence, this solution should not exhibit any non-physical behavior.

3.7.2. Transient problems

Consider the transient problem,

$$\begin{cases} \frac{\partial \phi}{\partial t} + \nabla \cdot (\mathbf{v}\phi) - \nabla(\alpha \nabla \phi) = s \\ \text{Boundary conditions} + \text{Initial condition} \end{cases} \tag{19}$$

where $\nabla \cdot \mathbf{v} = 0$ and $s \in L^2(\Omega)$. For a first order Euler time discretization, the following error estimate was also proven in [1]:

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

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.

3.7.3. 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 [1], it is shown that even for meshes where atypical edges are found, such as meshes of category M_3 with macro-elements, the numerical solution can still converge to the true solution if the number of atypical edges is not too large.

Here it is important to bring forward the main weakness of this finite volume scheme. For highly anisotropic meshes were the elements have very high aspect ratios (such as most adapted meshes on shocks), the number of atypical meshes can be too high to ensure global convergence to the true solution. This suggests using mesh adaptation with a strict constraint on the aspect ratio of the elements or using isotropic grid refinement. We also want to mention that this constraint does not apply to computations of Euler flows: for such flows there is no diffusive flux.

In the report [17], the numerical rate of convergence of the diffusion scheme was studied with extensive numerical experiments. These numerical results show that the observed rate of convergence of the diffusion scheme is of order 2 at the circumcenters. It is not a contradiction with the theoretical results: it shows that the theoretical rate of convergence is not optimal.

3.8. Discrete approximation at an interface

We first consider meshes where all reference positions lie in their associated control volume. Consider an interface

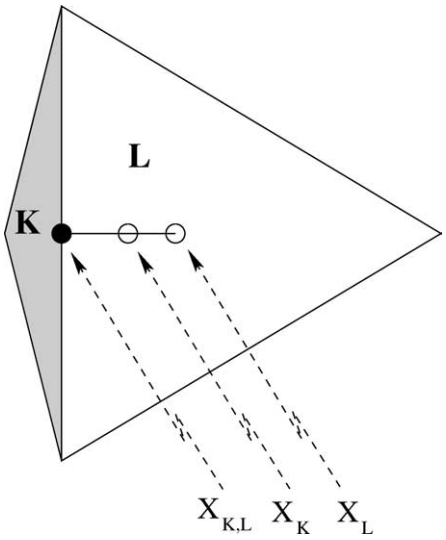

Fig. 3. Interpolation at an interface.

$\sigma_{K,L}$, the value for ϕ at this interface could be approximated with the linear interpolation

$$\phi_{K,L} = (1 - t_{K,L})\phi_K + t_K \phi_L$$
$$t_{K,L} = \frac{(X_K - X_{K,L}) \cdot (X_L - X_K)}{(X_L - X_K) \cdot (X_L - X_K)}$$

With this linear interpolation, the minimum or the maximum of the approximative solution is preserved at the interfaces when $0 \leqslant t_{K,L} \leqslant 1$. Which is only the case when the reference positions lie in their associated control volumes (meshes of category M_1). Unfortunately, the most common meshes are those for which there exist reference positions that do not lie in their associated control volume. This is the case for meshes of category M_2 (Fig. 3). Preliminary numerical results showed that the linear interpolation could make the scheme unstable when the $t_{K,L}$ 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, we use the geometrical average

$$\phi_{K,L} = \frac{m(K)\phi_K + m(L)\phi_L}{m(K) + m(L)} \tag{20}$$

where $m(K)$ is the measure of the hyperplane K (the volume of the cell K in 3D). Unfortunately, in theory, this approximation does not give as much precision than the linear interpolation.

3.9. Discrete approximation of the gradient

Since our approximation functions are constant by control volume, the gradient of a function cannot be directly computed, but only approximated. We first assume that the gradient $(\nabla\phi)_K$ of a scalar ϕ over the volume K is a constant vector. Let $(\nabla\phi \cdot \mathbf{n})_{K,L}$ be the normal gradient to the interface $\sigma_{K,L}$ (it being a known quantity), we suppose that

the projection of $(\nabla\phi)_K$ over $\sigma_{K,L}$ should be closed to $(\nabla\phi\cdot\mathbf{n})_{K,L}$:

$$(\nabla\phi)_K\cdot\mathbf{n}_{K,L}\approx(\nabla\phi\cdot\mathbf{n})_{K,L} \tag{21}$$
$$(\nabla\phi\cdot\mathbf{n})_{K,L}=\frac{\tau_{K,L}}{m(\sigma_{K,L})}(\phi_L-\phi_K)$$

To obtain a closed system of linear equations, (21) is applied to all interfaces which belong the cell K:

$$\begin{bmatrix}\mathbf{n}_{K,1}\\ \vdots\\ \mathbf{n}_{K,n}\end{bmatrix}(\nabla\phi)_K\approx\begin{bmatrix}(\nabla\phi\cdot\mathbf{n})_{K,1}\\ \vdots\\ (\nabla\phi\cdot\mathbf{n})_{K,n}\end{bmatrix}$$

or, in a more compact way:

$$\mathbf{N}(\nabla\phi)_K\approx(\nabla\phi\cdot\mathbf{n})_\sigma \tag{22}$$

The solution to this system of equations is the approximated gradient.

In general, the over-constraint system of Eq. (22) is not compatible. The "best" solution is approximated with a least square method, the linear system

$$\mathbf{N}^t\mathbf{N}(\nabla\phi)_K=\mathbf{N}^t(\nabla\phi\cdot\mathbf{n})_\sigma$$

being solved. It is worth to mention that this approximation is used to calculate the pressure gradient in the momentum equations.

There is also a particular case when the control volume K is a tetrahedron and the gradient of a scalar variable is divergence free:

$$\sum_{\sigma\in\mathcal{E}_K}(\nabla\phi\cdot\mathbf{n})_\sigma=0$$

$\mathcal{E}_K$ being the set of interfaces surrounding the control volume K. Based upon a geometrical property of tetrahedras, we have the following linear combination:

$$0=\nabla\phi_K\cdot\sum_{\sigma\in\mathcal{E}_K}m(\sigma)\mathbf{n}_\sigma=\sum_{\sigma\in\mathcal{E}_K}m(\sigma)\nabla\phi_K\cdot\mathbf{n}_\sigma$$
$$=\sum_{\sigma\in\mathcal{E}_K}m(\sigma)(\nabla\phi\cdot\mathbf{n})_\sigma$$

In this case, given this linear combination, there is a unique solution to (22).

3.10. Discrete equations, convection–diffusion operator $\mathcal{CD}$

In this section, to discretize the convections–diffusion equations, we put together the operators that were defined in the previous sections. Let us consider the following equation on the domain Ω:

$$\left.\frac{\partial\phi}{\partial t}\right|_{t=t_{n+1}}+\nabla\cdot\left(\mathbf{v}(t_{n+1})\otimes\phi^*\right)-\nabla\left(\alpha\nabla\phi^*\right)=s^n \tag{23}$$

where

- ϕ can be any scalar variable or the components of the velocity vector;
- $\left.\frac{\partial\phi}{\partial t}\right|_{t=t_{n+1}}=\frac{\phi^{n+1}-\phi^n}{\delta t}$;
- s^n is a source term;
- $\mathbf{v}(t_{n+1})=\frac{3}{2}\mathbf{v}^n-\frac{1}{2}\mathbf{v}^{n-1}$ is an approximation of $\mathbf{v}$ at time $t=t_{n+1}$, it is such that $\nabla\cdot\mathbf{v}(t_{n+1})=0$.

The discrete equations are obtained by integrating (23) over each control volume K and applying the Gauss theorem:

$$\int_K\left.\frac{\partial\phi}{\partial t}\right|_{t=t_{n+1}}\mathrm{d}V+\int_{\partial K}\left(\mathbf{v}(t_{n+1})\otimes\phi^*\right)\cdot\mathbf{n}\,\mathrm{d}S$$
$$-\int_{\partial K}\nabla\left(\alpha\nabla\phi^*\right)\cdot\mathbf{n}\,\mathrm{d}S=\int_K s^n\,\mathrm{d}V \tag{24}$$

The quantities ϕ^*, ϕ^n and s^n are assumed constant over any given volume K. When both the convective and the diffusive schemes presented in the previous section are applied to the surface integrals, the discrete equation for K is given by this expression:

$$m(K)\left.\frac{\partial\phi}{\partial t}\right|_{t=t_{n+1}}+\sum_{\sigma\in\mathcal{E}_K}m(\sigma)v_\sigma\phi^*_{\sigma,+}$$
$$-\sum_{\sigma\in\mathcal{E}_K}\alpha_\sigma\tau_\sigma\left(\phi^*_L-\phi^*_K\right)=m(K)s^n_K \tag{25}$$

where $\mathcal{E}_K$ is the set of interfaces which belong to the boundary of volume K. The system (25) is linear but not symmetric and the associated matrix is a diagonal dominant M-matrix. This imply 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 (25) is used to solve all scalar variables. The solution of this system will be denoted as $\phi^*_K=CD(\phi^n_K)$. Thanks to the time discretization, the components of the velocity vector are solved in a decouple 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: $\mathbf{v}^*_K=CD(\mathbf{v}^n_K)$. It is worth mentioning that for the momentum equations, the source term for the components of the velocity vector takes into account the pressure gradient:

$$\mathbf{s}^n_K=\mathbf{f}^n_K-(\nabla P)^n_K$$

$\mathbf{f}^n_K$ being the discrete approximation of a force on the control volume K and $(\nabla P)^n_K$ the pressure gradient over K. When convecting and diffusing the velocity field, the solution $\mathbf{v}^*$ is not divergence free and a projection has to be made.

3.11. 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 $\mathcal{E}$ and the projection operator $\mathcal{P}$. The goal of the first operator

is to compute an intermediate normal component of the cell velocity on the cell interfaces. The projection operator computes both an update of the pressure and the normal component of the cell velocity field at the cell interfaces. The update of the normal velocity field is latter used to correct the velocity field within the cells.

3.11.1. Extension operator $\mathcal{E}$

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

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

To define this operator, we make the assumption that the variation of the normal velocity component must agree with the variation of the cell velocity computed in the predictor step. To interpolate the variation of the normal velocity, we use the geometrical average:

$$(\mathbf{v}\cdot\mathbf{n})_{K,L}^{n+1/2}=(\mathbf{v}\cdot\mathbf{n})_{K,L}^n+\frac{[m(K)\delta\mathbf{v}_K+m(L)\delta\mathbf{v}_L]\cdot\mathbf{n}_{K,L}}{m(K)+m(L)}\tag{26}$$

$$\delta\mathbf{v}=\mathbf{v}^{n+1/2}-\mathbf{v}^n$$

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

$$(\mathbf{v}\cdot\mathbf{n})_{K,b}^{n+1/2}=\mathbf{v}_{K,b}^n+\delta\mathbf{v}_K\cdot\mathbf{n}_{K,b}\tag{27}$$

It is worth to mention 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 would be zero. In this case, the solution would exhibit spurious pressure oscillations that have no physical meaning (false pressure modes).

Later on, the computation of the velocity field $\mathbf{v}^{n+1/2}$ with the extension operator will be denoted as:

$$(\mathbf{v}\cdot\mathbf{n})_\sigma^{n+1/2}=E\left(\mathbf{v}^{n+1/2},\mathbf{v}_K^n\right)\tag{28}$$

3.11.2. Projection operator $\mathcal{P}$

The projection operator actually computes a velocity field that is divergence free, it is applied to both the pressure and the velocity

$$\mathcal{P}:\left(P_K^n,(\mathbf{v}\cdot\mathbf{n})_\sigma^{n+1/2}\right)\mapsto\left(P_K^{n+1},\mathbf{v}_K^{n+1}\right)\tag{29}$$

and it is carried out in two steps. Eq. (7) is first written under this form:

$$\mathbf{v}^{n+1}=-\delta t\beta\nabla\left(\delta P^{n+1}\right)+\mathbf{v}^{n+1/2}\tag{30}$$

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

$$\beta\delta t\sum_{\sigma\in\mathcal{E}_K}m(\sigma)\left(\nabla\delta P^{n+1}\cdot\mathbf{n}\right)_\sigma=\sum_{\sigma\in\mathcal{E}_K}m(\sigma)(\mathbf{v}\cdot\mathbf{n})_\sigma^{n+1/2}\tag{31}$$

The solution to Eq. (31) gives the correction to the pressure field. As for the correction of the velocity field on the cell interfaces, it is given by Eq. (30). Which is discretized as

$$v_\sigma^{n+1}=-\delta t\beta\frac{\tau_\sigma}{m(\sigma)}(\delta P_L-\delta P_K)+v_\sigma^{n+1/2}\tag{32}$$

on the cell interfaces.

Before solving (31), 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:

$$\delta t\beta\nabla\left(\delta P^{n+1}\right)\cdot\mathbf{n}=v_{K,b}^{n+1/2}-v_{K,b}\tag{33}$$

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:

$$\delta P_{K,b}^{n+1}=g\left(X_{K,b},t^{n+1}\right)-g\left(X_{K,b},t^n\right)\tag{34}$$

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 cell $\sigma_{K,b}$ 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:

$$\left(\mathbf{v}^{n+1}-\mathbf{v}^{n+1/2}\right)_K\cdot\mathbf{n}_{K,L}=W_{K,L}\left[(\mathbf{v}\cdot\mathbf{n})_{K,L}^{n+1}-(\mathbf{v}\cdot\mathbf{n})_{K,L}^{n+1/2}\right]\tag{35}$$

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 (35) to all interfaces which belong to K. This linear system can be inconsistent, its solution is always approximated with a least square method. In order to make sure that the cell velocity always agrees with an adjacent wall boundary condition, the parameter W had to be introduced.

Both the velocity-pressure formulation and the projection operator fall into the same class than the operators presented in [18]. Hence, the proof of the unicity of the solution for the pressure given in [18] also applies to this scheme.

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

$$\left(P_K^{n+1},\mathbf{v}_K^{n+1}\right)=\mathcal{P}\left(P_K^n,(\mathbf{v}\cdot\mathbf{n})_\sigma^{n+1/2}\right)\tag{36}$$

We end this section by making a short remark for the special case where there are no macro-elements. For this type of mesh, all control volumes are tetrahedras and it is possible to use the zero degree Raviart–Thomas finite element which contains the following polynomials [19,20]:

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

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

$$\begin{aligned}&\mathbf{v}_K \cdot \mathbf{n}_{K,L} = -\mathbf{v}_L \cdot \mathbf{n}_{L,K}\\ &\sum_{\sigma \in \mathcal{E}_\mathcal{K}} \mathbf{v}_K \cdot \mathbf{n}_\sigma = 0\end{aligned} \tag{37}$$

Then, 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 [9,10,21]. For our scheme, where macro-elements are frequent, it is not the case: the normal components of the cell velocity field cannot directly be computed from the cell velocity and we do not make use of the Raviart–Thomas finite elements.

3.12. The complete algorithm

We end this first part of the article by giving the whole algorithm for solving the Navier–Stokes equations coupled with other scalar transport equations such as the energy equation. We always use a start-up step and care must be given to the notation for the 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;
- ϕ_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 ϕ_K^0, apply the extension and the projection operators:

  $$(\mathbf{v}\cdot\mathbf{n})_\sigma^{-1/2} = E\big(\mathbf{v}_K^{-1}, \mathbf{v}_K^{-1/2}\big)$$
  $$\big(P_K^0, \mathbf{v}_K^0\big) = \mathcal{P}\big(P_K^{-1}, (\mathbf{v}\cdot\mathbf{n})_\sigma^{-1/2}\big)$$

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

  $$\mathbf{v}_K^{n+1/2} = \mathcal{CD}\big(\mathbf{v}_K^n\big)$$

  (2) apply the extension and the projection operators:

  $$(\mathbf{v}\cdot\mathbf{n})_\sigma^{n-1/2} = E\big(\mathbf{v}_K^{n+1/2}, \mathbf{v}_K^n\big)$$
  $$\big(P_K^{n+1}, \mathbf{v}_K^{n+1}\big) = \mathcal{P}\big(P_K^n, (\mathbf{v}\cdot\mathbf{n})_\sigma^{n-1/2}\big)$$

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

  $$\phi_K^{n+1} = \mathcal{CD}\big(\phi_K^n\big)$$

4. Numerical results

In this section, we present some numerical tests that were conducted to validate this numerical scheme. All computations were carried out in three dimensional domains, even for the 2D flows.

Before showing our results, we must say a few words about the resolution of the systems of discrete of equations. First, we recall that these systems of equations are linear and the scheme does not require any non-linear solver. To speed-up the computations, we always store the matrix of coefficients associated to the discrete systems of equations. In this paper, all results were obtain using the Orthomin2 algorithm. This algorithm is not as known as the conjugate gradient. It requires more operations (one matrix vector multiplication by iteration), but does not need the linear system to be symmetric. We compared the performance of this algorithm to GMRES and the conjugate gradient. For the problems we considered, Orthomin2 was a better choice than the others or the combination GMRES-CG (conjugate gradient for the symmetric systems associated the projection operator and GMRES for the other linear systems). As for the preconditioner, we only used a simple Jacobi diagonal preconditioner. For more information on these solvers, see [22].

For all problems, we give plots of the convergence history for the computations made on the finest grids. The goal is to verify the consistency of the extension-projection operator. A contradiction between convergence histories for the variations of variables and the residuals would show that the solution computed after the projection does not satisfy the Navier–Stokes equations. Thus showing that the projection operator is not appropriate.

4.1. Natural Convection in a square cavity

This problem deals with a confined Boussinesq 2D fluid flow in a square cavity. The free convection originates from buoyancy forces due to a fluid density gradient. We consider the body force $\mathbf{g}(\rho_\infty - \rho)$ where $\mathbf{g}$ is the local acceleration and ρ_∞ is the density for a reference state. For an incompressible flow with small temperature gradients, this force can be approximated with the *Boussinesq* approximation

$$\mathbf{g}(\rho_\infty - \rho) \simeq \mathbf{g}\rho_\infty\beta(T - T_\infty)$$

where β is the volumetric thermal expansion coefficient and T_∞ the temperature for the reference state. This source term is added to the momentum equations to model the body force.

We carried out computations for a fluid of Prandt number 0.71 and Rayleigh number of 10^6. The details for the set up for this problem are presented thereafter.

Domain:

$$[0.0, 0.1] \times [0.0, 1.0] \times [0.0, 1.0]$$

Boundary conditions:

$$\mathbf{v}\cdot\mathbf{n}|_{\partial\Omega} = 0, \qquad \mathbf{v}\cdot\tau|_{\partial\Omega} = 0$$
$$T(y = 0, z) = 0, \qquad T(y = L, z) = 1$$

$$\left.\frac{\partial T}{\partial n}\right|_{z=0,z=L} = 0$$

Initial condition: $\mathbf{v}(\mathbf{x}) = \mathbf{0}$, $P(\mathbf{x}) = 0$, $T(\mathbf{x}) = 0$
Physical properties and dimensions:

$$Ra = \frac{\rho\|\mathbf{g}\|\Delta T L^3}{k\mu} = 10^6, \qquad Pr = \frac{\mu}{c_p k} = 0.71$$

$$\rho = 1.0, \qquad = 1, \qquad L = 1.0, \qquad \mu = 0.71$$

We recall that k is the thermal conductivity and c_p the specific heat at constant pressure.
Time step:

$$\delta t = 0.1$$

Convergence criteria:

$$\left|v^{n+1} - v^n\right|_\infty < 1 \times 10^{-2} \quad \text{and}$$
$$\left|w^{n+1} - w^n\right|_\infty < 1 \times 10^{-2}$$

This problem has been studied in details by Vahl Davis et al. ([23,24] for different Rayleigh numbers). These authors solved this problem on several meshes with a second order difference method and extrapolated reference quantities with Richardson's extrapolation. Hence, our results are compared to those obtain by others with a second order discretization scheme. We solved this problem on three different unstructured meshes made of 902, 3656 and 8254 control volumes (a plane cut of the coarsest mesh is shown in Fig. 4). Two simulations were carried out for each meshes: one with the power-law scheme, the other without a correction of the diffusion coefficient.

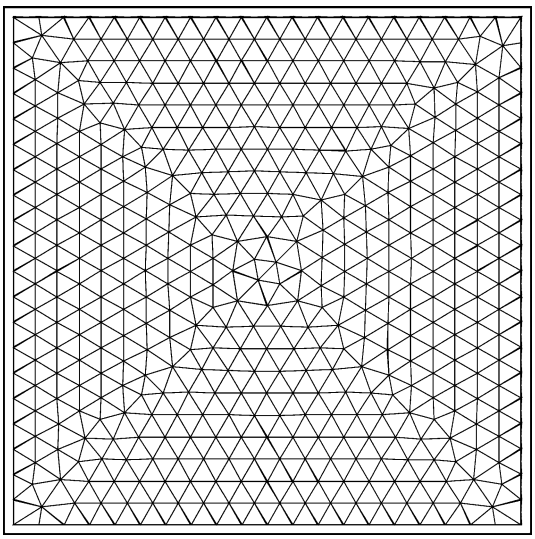

Fig. 4. 2D Boussinesq flow: coarsest mesh (902 cells).

In Table 1, we compare the maximum value of the velocity components on the mid-plane sections and their locations to the benchmark values published by Vahl Davis et al. Our functions of approximation being constant by control volume, we do not give an exact location. Instead, we give the interval in which they are located.

In Table 2, we present the maximum value of the Nusselt number and its average on the hot wall. For each control volume K adjacent to the hot wall, the approximation of the Nusselt number was computed as follows:

$$Nu_K = \frac{L_{\text{ref}} k_K}{k_{\text{ref}} \Delta T_{\text{ref}}} \cdot \frac{T_{\sigma_K} - T_K}{\|X_\sigma - X_K\|}$$

where, in this case, $L_{\text{ref}} = 1$, $k_{\text{ref}} = 1$, $\Delta T_{\text{ref}} = 1$. For the Nusselt number, we give the exact location where the flux was computed.

For all benchmark quantities, systematic grid refinement lead to more accurate results. Moreover, the power law scheme gave more accurate results than the upwind scheme. For all further computations presented in this paper, the power-law scheme is preferred to the simple upwind scheme.

In Figs. 5 and 6, we show plane cuts of the velocity components and the temperature for the solution obtained with the finest grid and the power-law scheme. The convergence history for this solution is shown on Fig. 7.

4.2. 2D flow around a cylinder

This problem 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. The computations were carried out for the Reynolds number $Re = 20$. For such a low Reynolds number, there is no vortex shedding and the flow is permanent. It is not a thermal flow. Nevertheless, it evaluates the capacity of the scheme for computing, on unstructured

Table 1
2D Boussinesq flow: maximum velocity components at mid-plane sections

Our results	Power-law scheme	$w_{\max}$ $z = 1/2$	$v_{\max}$ $y = 1/2$
902 cells	no	215.12 $y \in [0.0002, 0.0433]$	72.46, $z \in [0.827, 0.870]$
902 cells	yes	217.58 $y \in [0.0002, 0.0433]$	66.89, $z \in [0.827, 0.870]$
3656 cells	no	217.11, $y \in [0.0431, 0.0433]$	69.99, $z \in [0.870, 0.870]$
3656 cells	yes	218.83, $y \in [0.0416, 0.0433]$	65.89, $z \in [0.848, 0.868]$
8254 cells	no	217.82, $y \in [0.029, 0.433]$	68.095, $z \in [0.855, 0.856]$
8254 cells	yes	219.34, $y \in [0.029, 0.433]$	65.74, $z \in [0.855, 0.856]$
Reference solution		219.36, $y = 0.0379$	64.63, $z = 0.850$

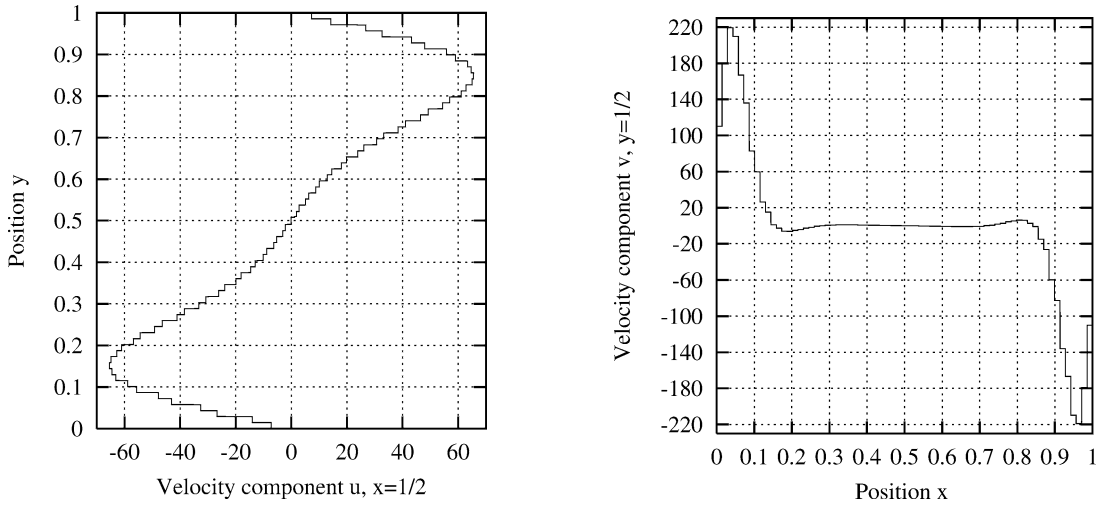

Fig. 5. 2D Boussinesq flow: Comparative results, velocity components on the mid-planes $y = 1/2$ and $z = 1/2$.

Table 2
2D Boussinesq flow: maximum and average Nusselt number at the hot wall

Our results	Power-law scheme	$Nu_{\max}$	$\overline{Nu}$
902 cells	no	22.419, $y = 0.0750$	11.152
902 cells	yes	20.785, $y = 0.0750$	10.632
3656 cells	no	20.077, $y = 0.0375$	9.883
3656 cells	yes	19.086, $y = 0.0375$	9.374
8254 cells	no	18.712, $y = 0.0417$	9.521
8254 cells	yes	18.443, $y = 0.0417$	9.105
Reference solution		17.925, $y = 0.0378$	8.800

non-orthogonal meshes, quantities related to the diffusion flux. The set-up for this problem is as follows:

Domain:

$[0.0, 0.1] \times [0.0, 2.2] \times [0.0, 0.41]$,

see Fig. 8.
Boundary conditions:

Inlet:

$$v(x, y = 0, z) = \frac{4 \cdot v_a \cdot (H - z)}{H^2}$$

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

Outlet:

$$\left.\frac{\partial v}{\partial n}\right|_{x,y,z=2.2} = 0$$

$$w(x, y = 2.2, z) = 0$$

$$P(x, y = 2.2, z) = 0$$

Walls:

$$\mathbf{v} \cdot \mathbf{n} = 0, \qquad \mathbf{v} \cdot \tau = 0$$

Initial condition:

$$\mathbf{v}(\mathbf{x}) = \mathbf{0}, \qquad P(\mathbf{x}) = 0$$

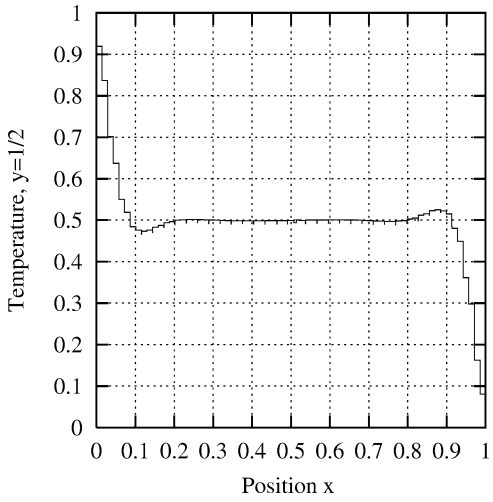

Fig. 6. 2D Boussinesq flow: Comparative results, temperature on the mid-plane section $z = 1/2$.

Physical properties:

$$Re = \frac{v_a D}{\nu} = 20, \qquad v_a = 0.2, \qquad D = 0.1$$

Time step:

$$\delta t = 0.05$$

Convergence criteria:

$$\left|v^{n+1} - w^n\right|_\infty < 1 \times 10^{-5} \quad \text{and}$$

$$\left|v^{n+1} - w^n\right|_\infty < 1 \times 10^{-5}$$

For this problem, we compare our results to benchmark quantities published by Turek et al. [25]. In this report, the drag, lift and difference of pressure between two positions on the disc are given. The following quantities are provided:

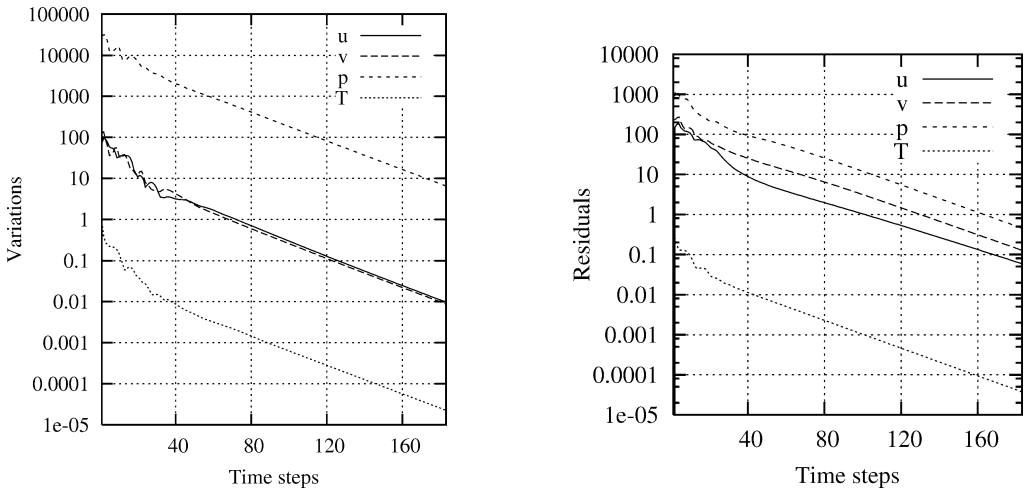

Fig. 7. 2D Boussinesq flow: Convergence, the graph on the left shows the norms of the variations $|\phi^{n+1} - \phi^n|_\infty$, the norms $\|\phi^{n+1} - \phi^n\|_{L^2}$ of the residuals are shown on the right.

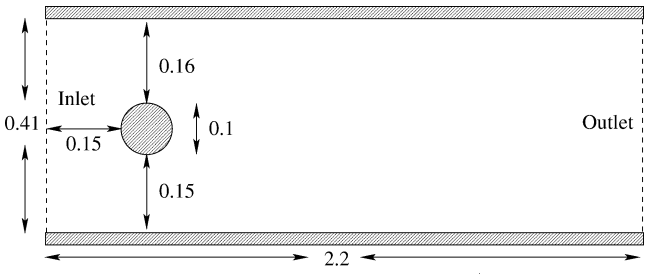

Fig. 8. 2D flow around a cylinder: Geometry.

(1) drag coefficient:

$$C_D = \frac{2 \cdot F_D}{\rho v_a^2 DHL}$$

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

(2) lift coefficient:

$$C_L = \frac{2 \cdot F_L}{\rho v_a^2 DHL}$$

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

(3) pressure difference:

$$\Delta P = P(0, 0.15, 0.2) - P(0, 0.25, 0.2)$$

where:

- $L = 0.10$ is the depth in the 3rd dimension (the flow being solved in a 3D domain);
- τ is a tangent vector to the cylinder surface;
- $v_\tau = \mathbf{v} \cdot \tau$ 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.

Table 3
2D flow around a cylinder: Maximum drag, lift and other comparative results

Cells	C_D	C_L	ΔP
16933	5.58	0.0073	0.119
36724	5.58	0.0124	0.117
49231	5.58	0.0118	0.116
Benchmark quantities	5.57–5.59	0.0104–0.0110	0.1172–0.1176

In order to show that we are able to obtain a solution which is independent from the grid size, computations were carried out on three different meshes. The benchmark quantities provided by Turek et al. [25] and our results are presented in Table 3.

We were able to predict the drag coefficient accurately for all three meshes. As for the lift, the difference between our results and the reference quantities diminish as we refine our grid. Finally, for the pressure difference, the value given by our scheme is slightly below the benchmark result. We give the convergence history for the finest mesh in Fig. 9.

4.3. 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:

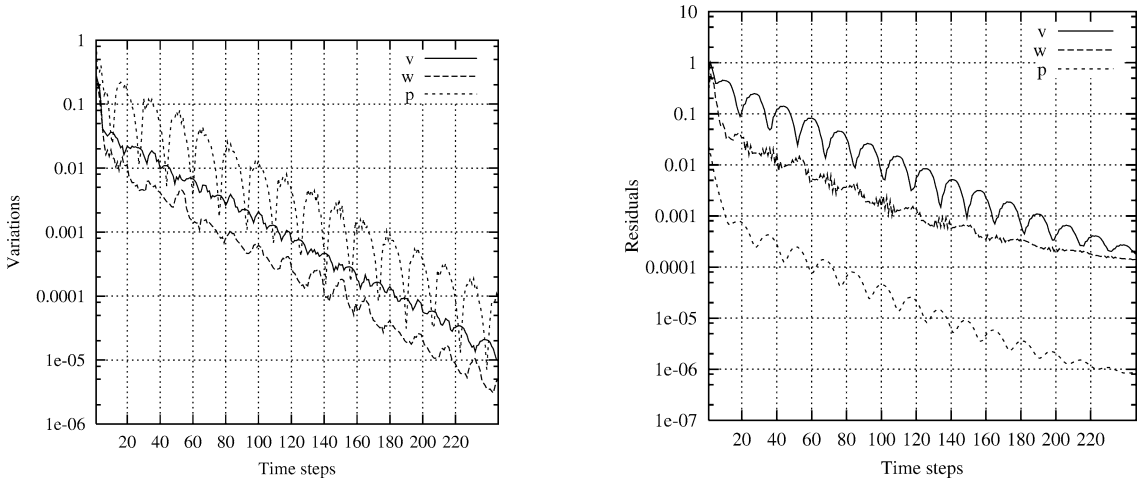

Fig. 9. 2D flow around a cylinder: Convergence, the graph on the left shows the norms of the variations $|\phi^{n+1} - \phi^n|_\infty$, the norms $\|\phi^{n+1} - \phi^n\|_{L^2}$ of the residuals are shown on the right.

Inlet:

$$u(x, y, z = 0) = 0, \qquad v(x, y, z = 0) = 0$$
$$w(x, y, z = 0) = 1, \qquad T(x, y, z = 0) = 1$$

Outlet:

$$u(x, y, z = 1.2) = v(x, y, z = 1.2) = 0$$
$$\left.\frac{\partial w}{\partial n}\right|_{z=1.2} = 0 \; P(x, y, z = 1.2) = 0, \qquad \left.\frac{\partial T}{\partial n}\right|_{z=1.2} = 0$$

Walls:

$$\mathbf{v} \cdot \mathbf{n} = 0, \qquad \mathbf{v} \cdot \tau = 0, \qquad T = 0$$

Initial condition:

$$\mathbf{v}(\mathbf{x}) = \mathbf{0}, \qquad P(\mathbf{x}) = 0, \qquad T(\mathbf{x}) = 0$$

Physical properties and dimensions:

$$Re_D = \frac{w_m D}{\nu} = 120, \qquad w_a = 1.0$$
$$Pr = \frac{\nu c_p}{k} = 1.0, \qquad D = 0.1, \qquad c_p = 1.0$$

Time step:

$$\delta t = 0.1$$

Convergence criteria:

$$\left|w^{n+1} - w^n\right|_\infty < 1 \times 10^{-5}$$

w_a 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 meshes were built using the extrusion of 2D meshes composed of triangles. The grids were made of 40, 60 and 80 sections, respectively.

For this 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 with these equations:

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

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

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, we do not have an exact solution for the temperature. Nevertheless, in this part of the domain, there is no variation along the cylinder of the dimensionless temperature:

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

$T_m(z)$ 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 (see Kays [27]):

$$T_m = \frac{1}{A_c w_a}\int_{A_c} w(\mathbf{x}) T(\mathbf{x})\, \mathrm{d}S$$

A_c being the area of a cross section.

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 [27]:

$$Nu_D = 3.657$$

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.

In order to verify the convergence behavior of our scheme toward the exact solution, we computed some estimates of the error for the velocity, pressure gradient and Nusselt number in the last section of the duct.

For all these variables, the L_2 norm

$$\|e\|_{L_2} = \sqrt{\sum_K m(K)\big(\phi_K - \phi(X_K)\big)^2}$$

Table 4
Thermal flow in a cylinder, norm $\|\phi_K - \phi(X_K)\|_{L_2}$ of the error

Variable cells	h^m/h^{m-1}	w	$\frac{\partial p}{\partial z}$	Nu
29080		7.75^{-6}	4.72^{-5}	2.89^{-4}
67500	0.667	3.50^{-6}	3.76^{-5}	1.95^{-4}
$\frac{\|e^m\|_{L_2}}{\|e^{m-1}\|_{L_2}}$		0.452	0.796	0.675
133200	0.750	2.77^{-6}	2.91^{-5}	1.37^{-4}
$\frac{\|e^m\|_{L_2}}{\|e^{m-1}\|_{L_2}}$		0.791	0.774	0.703

Table 5
Thermal flow in a cylinder, statistics

Statistics unknowns	Iterations	Time (seconds)	Time per iteration	Memory usage (megabytes)
145400	84	1612	19.2	67
337500	95	4688	49.3	150
666000	134	12526	93.5	294

was computed. Those norms are given in Table 4. We also provide the following ratios:

(1) h^m/h^{m-1}, the ratio between the diameters of the largest cell of mesh m and mesh $m-1$;
(2) $\|e^m\|_{L_2}/\|e^{m-1}\|_{L_2}$, the ratio between the norms of the error for mesh m and mesh $m-1$.

In Section 3.7.1 we gave an error estimate for the convection–diffusion operator. We also recall that the L_2 norm of this error estimate depends on the diameter of the largest cell. In this case, the results given in Table 4 show that the observed rate of convergence agrees with the theoretical results. Moreover, for the problem we considered, the rate of convergence of the approximative flux is also of the same order.

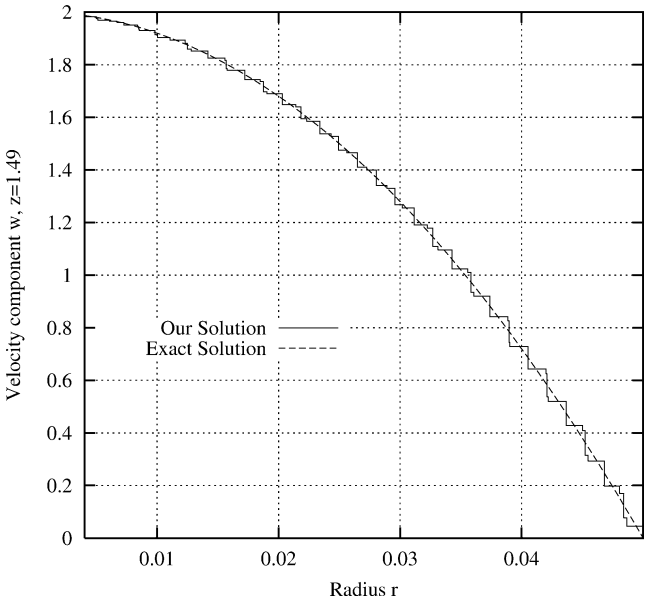

Fig. 10. Thermal flow in a cylinder: Velocity profile near the outlet.

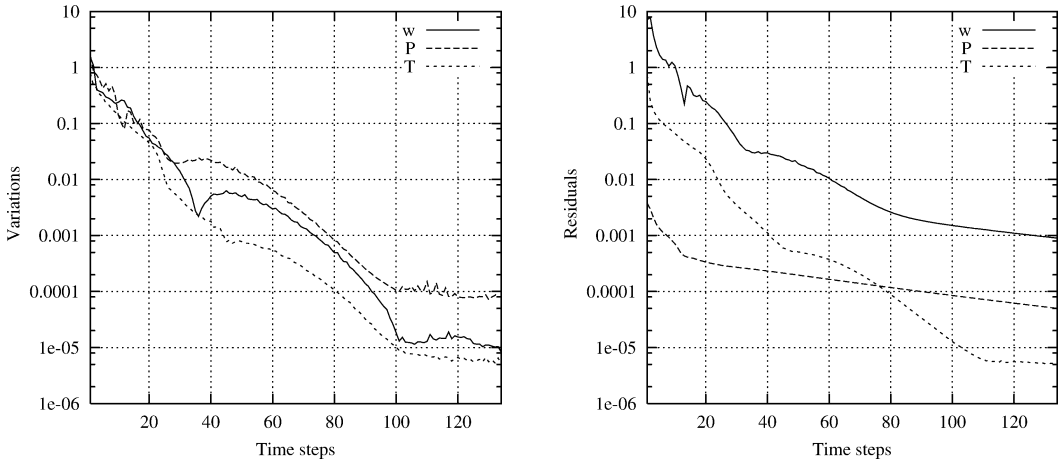

Fig. 11. Thermal flow in a cylinder: Convergence, the graph on the left shows the norms of the variations $|\phi^{n+1} - \phi^n|_\infty$, the norms $\|\phi^{n+1} - \phi^n\|_{L^2}$ of the residuals are shown on the right.

In Fig. 10, we show that there is no significant difference between the computed velocity component w and the analytical solution given by Eq. (38). As for the other problems, we show the convergence curves for the simulation on the finest grid in Fig. 11.

In Table 5, we provide some statistics relative to the performance of the computer code. We believe that those statistics are relevant to those who want to compare their code with others. This computer code was written in C++ and all the real numbers were stored in double precision format. All the problems considered in this paper were ran on a single processor PC equipped with AMD1000 thunderbird. We must say that those statistics show one odd behavior of the computer code: the number of time steps needed to reach convergence grows with the number of unknowns. Hence, the time taken to solve a problem does grow linearly with the number of unknowns. This behavior is mainly caused by the deterioration of the conditioning of the matrix associated to the projection operator (we recall that the projection operator requires solving a Laplacian equation). It can be shown that the condition number of the matrix associated to the discretized Laplacian equation is inversely proportional to the diameter h of the mesh [22]. Hence, the time needed to solve this system of equations increases as the mesh size decreases. This behavior is not unique to our scheme: it is one of the drawbacks of all incompressible flow solvers that use a projection, that includes SIMPLEs family of algorithms [28].

5. Conclusion

A numerical method has been proposed to solve the Navier–Stokes equations for incompressible thermal flows and the convection–diffusion of scalar quantities. This solver is based upon a fractional step scheme and the finite volume cell centered method on unstructured meshes. One of its main characteristics is the usage of the cell-circumcenters to store the unknowns. This leads to a very simple diffusion scheme on non-orthogonal unstructured meshes. By combining together elements of the mesh to form new cells, this scheme can be used on general unstructured meshes that do not satisfy the Delaunay condition. This solver allows the local conservation of mass and scalar quantities and the numerical preservation of the maximum principle for scalar quantities. However, 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.

The convection–diffusion scheme is very robust and easy to implement. Being of order one, it is a low order scheme. Nevertheless, the convection scheme accuracy's could be improved with a more sophisticated reconstruction method such as MUSCL.

Numerical solutions for laminar steady 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 problems, reference quantities associated with the diffusion flux were computed on non-orthogonal grids. To show that the solution computed after the projection satisfies the Navier–Stokes equations, we gave the convergence curves for both the variations of the variables and the residuals. This scheme is under active development, numerical results for a turbulent flow using the k–ε model have already been presented in [14, 26]. Current developments of the scheme include: a second order approximation of the convection flux, more exhaustive computations of turbulent flows with heat transfer and an extension to compressible flows.

References

[1] R. Eymard, T. Gallouet, R. Herbin, Finite volume methods, in: P.G. Ciarlet, J.L. Lions (Eds.), Handbook of Numerical Analysis, vol. 7, North-Holland, Amsterdam, 2000.
[2] I. Faille, A control volume method to solve an elliptic equation on a 2D irregular meshing, Comput. Methods in Appl. Mech. Engrg. 100 (1992) 275–290.
[3] T.J. Barth, Recent developments in high order K-exact reconstruction on unstructured meshes, AIAA-93-0668, January 1993.
[4] W.J. Coirier, An adaptively-refined, cartesian, cell-based scheme for the Euler and Navier–Stokes equations, NASA TM-106754, 1994.
[5] A. Haselbacher, J. Blasek, On the accurate and efficient discretization of the Navier–Stokes equations on mixed grids, AIAA J. 38 (2000) 2094–2102.
[6] D.J. Mavriplis, Three-dimensional multigrid Reynolds-averaged Navier–Stokes solver unstructured meshes, AIAA J. 33 (1995) 445–453.
[7] A.J. Chorin, Numerical solution of the Navier–Stokes equations, Math. Comp. 22 (1968) 745–762.
[8] R. Teman, Sur l'approximation de la solution des equations de Navier–Stokes par la méthode des pas fractionnaires II, Arch. Rat. Mech. Anal. 33 (1969) 377–385.
[9] S. Boivin, F. Cayré, J.M. Hérard, A finite volume method to solve the Navier–Stokes equations for incompressible flows on unstructured meshes, Internat. J. Therm. Sci. 39 (2000) 806–825.
[10] S. Boivin, F. Cayré, J.M. Hérard, Un schéma volumes Finis pour la simulation d'Écoulements diphasiques sur maillages triangulaires, Internat. J. Fin. Elem. 10 (5) (2001) 539–574.
[11] J.M. Lopez, J. Shen, Numerical simulation of incompressible flows in cylindrical geometries using a spectral projection method, Internat. J. Appl. Sci. Comput. 5 (1998) 35–40.
[12] J. Shen, On error estimates on projection methods for the Navier–Stokes equations: Second order schemes, Math. Comp. 65 (1996) 1039–1065.
[13] J. Shen, On error estimates on projection methods for Navier–Stokes equations: First order schemes, SIAM J. Numer. Anal. 29 (1992) 55–77.
[14] S. Perron, Résolution numérique d'écoulements 3 dimensions avec une nouvelle méthode de volumes finis pour maillages non structurés. Ph.D. Thesis, Université du Québec à Chicoutimi, Canada, 2001.
[15] S. Perron, S. Boivin, J.M. Hérard, A finite volume method to solve the Navier–Stokes equations for incompressible flows on unstructured meshes, Comput. Fluids, submitted for publication.
[16] J.M. Davroux, A. Archambeau, F. Hérard, 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.

[17] F. Cayré, 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.
[18] J.M. Boivin, S. Hérard, Un schéma de volumes finis pour résoudre les équations de Navier–Stokes sur une triangulation, Rev. Europ. Élém. Finis 5 (1996) 461–490.
[19] F. Brezzi, M. Fortin, Mixed and Finite Element Methods, Springer, Berlin, 1991.
[20] V. Girault, P.A. Raviart, Finite Element Methods for Navier–Stokes Equations, Springer, Berlin, 1986.
[21] F. Cayré, Méthodes de volumes finis pour Maillages Non Structutés pour la Simulation Numérique des écoulements Incompressibles Monophasiques et Diphasiques, Master Thesis, Université Laval, Canada, 1999.
[22] A. Greenbaum, Iterative Methods for Solving Linear Systems, SIAM, Philadelphia, PA, 1997.
[23] G. De Vahl Davis, I.P. Jones, Natural convection of air in a square cavity: A comparison exercise, Internat. J. Heat Mass Transfer 3 (1983) 227–248.
[24] G. De Vahl Davis, Natural convection of air in a square cavity: A benchmark solution, Internat. J. Heat Mass Transfer 3 (1983) 249–264.
[25] M. Schafer, S. Turek, Benchmark computations of laminar flow around cylinder, in: Proceedings DFG Priority Research Program, Flow Simulation on High Performance Computers, Vieweg, Braunschweig, 1992.
[26] S. Boivin, J.M. Hérard, S. Perron, A finite volume method to solve the Navier–Stokes equations for incompressible flows on unstructured meshes, in: G.E. Schneider (Ed.), Proceedings 9th Annual Conference of the CFD Society of Canada, 2001, p. 62.
[27] W.M. Kays, M.E. Crawford, Convective Heat and Mass Transfer, Mcgraw-Hill, New York, 1980.
[28] S.V. Patankar, Numerical Heat transfer and Fluid Flow, Hemisphere, Washington, DC, 1996.