INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids **28**: 917–943 (1998)

THREE-DIMENSIONAL INCOMPRESSIBLE NAVIER–STOKES EQUATIONS ON NON-ORTHOGONAL STAGGERED GRIDS USING THE VELOCITY–VORTICITY FORMULATION

F. BERTAGNOLIO[a,1] AND O. DAUBE[b,*]

[a] *Laboratoire d'Informatique pour la Mécanique et les Sciences de l'Ingénieur, B.P. 133, 91403 Orsay Cedex France*
[b] *LIMSI and CEMIF/Université d'Evry, 40 rue du Pelvoux, 91020 Evry Cedex France*

SUMMARY

This paper is concerned with the numerical resolution of the incompressible Navier–Stokes equations in the velocity–vorticity form on non-orthogonal structured grids. The discretization is performed in such a way, that the discrete operators mimic the properties of the continuous ones. This allows the discrete equivalence between the primitive and velocity–vorticity formulations to be proved. This last formulation can thus be seen as a particular technique for solving the primitive equations. The difficulty associated with non-simply connected computational domains and with the implementation of the boundary conditions are discussed. One of the main drawback of the velocity–vorticity formulation, relative to the additional computational work required for solving the additional unknowns, is alleviated. Two- and three-dimensional numerical test cases validate the proposed method. © 1998 John Wiley & Sons, Ltd.

KEY WORDS: Navier–Stokes; incompressible flow; velocity–vorticity formulation; generalized curvilinear co-ordinates

1. INTRODUCTION

Since the pionneer work by Fasel [1], the use of velocity and vorticity as dependent variables in solving the incompressible Navier–Stokes equations has appeared to many authors as an attractive alternative to the velocity–pressure formulation. The main expected advantage was the fact that with the pressure being removed from the set of equations, the difficulties associated with its resolution (see the review paper by Gresho [2]), would essentially disappear. Some other advantages were also expected, e.g. as an easier implementation of open boundary conditions to compute flows in unbounded domains. Also noteworthy is the fact quoted by Speziale [3], that in the context of a non-inertial frame of reference, the additional centrifugal and Coriolis terms can be integrated into the different terms of the formulation and explicitly appear only through the boundary conditions. Besides these advantages, it is natural to expect that some specific difficulties could arise when using the velocity–vorticity formulation. They fall into three main classes.

* Correspondence to: CEMIF/IUP 40 rue du Pelvoux, CE 1455 Courcouronnes, 91020 Evry Cedex, France. Tel.: + 33 169477550; fax: + 33 169477599; e-mail: daube@iup.univ-evry.fr
[1] Current address: Wind Energy and Atmospheric Physics Department, Risø National Laboratory, P.O. Box 49, DK-4000 Roskilde, Denmark.

CCC 0271–2091/98/160917–27$17.50

Received March 1997
Revised November 1997

Firstly, since the velocity–vorticity formulation is obtained by differentiating the original Navier–Stokes equations, special care has to be taken in the case of multiply connected domains to ensure the equivalence between both formulations. This point was addressed for instance by Daube *et al.* [4]. However, the choice of the discrete operators in order to ensure the equivalence between the discrete formulations has not been addressed and is one of the objectives of this paper. In particular, the case of non-orthogonal grids will be considered.

Secondly, in most problems, the boundary conditions involve only the velocity, usually through Dirichlet conditions. This fact precludes an easy decoupling between the resolution of the vorticity transport equation and the computation of the associated velocities. This difficulty can be overcome by means of Quartapelle's integral relationships or by means of influence matrix techniques [5].

Thirdly, the velocity–vorticity has been known to induce additional computational effort since six unknowns (three components of the vorticity, and three of the velocity) are involved, as opposed to four for the primitive equations. Nevertheless, by introducing a Helmholtz decomposition of the velocity vector field, in order to solve the kinematic equations [6], it has been shown that the overall computational work involved by the velocity–vorticity formulation can be reduced to approximately the same as that for the primitive equations [7]. This solution procedure is adopted herein.

Many authors [1,8–11] developed several methods in order to benefit from the above-mentioned advantages. In addition to the previously mentioned difficulties, all these works solve the equations in a decoupled manner in order to avoid the resolution of the fully coupled set of equations. The numerical procedure therefore, always reduces to a two-step problem:

1. compute the vorticity by solving a convection–diffusion equation with appropriate boundary conditions that are supposed to take into account the Dirichlet conditions for the tangential components of the velocity;
2. compute the divergence-free velocity vector field, the curl of which is the just-computed vorticity. The various proposed methods essentially differ by the way the velocity is computed: streamfunction in 2D problems, Cauchy–Riemann system, Laplacian vector operator or vector streamfunction in 3D problems. It must be pointed out that the normal component of the velocity is taken into account in this step.

However, a common point to these works is that the set of continuous equations that is discretized is the velocity–vorticity form of the Navier–Stokes equations. As a matter of fact, the discrete equivalence between the formulations is not necessarily ensured, resulting in a velocity field that is not clearly satisfying a set of discrete equations in primitive form. The purpose of this paper is to develop a methodology which consists of discretizing the primitive Navier–Stokes equations, and then deriving a discrete velocity–vorticity formulation by means of discrete differentiation operators. As a result, the velocity–vorticity formulation can be seen as a particular way to solve the discrete primitive equations.

In order to ensure an algebraic equivalence between the formulations, the differential operators must mimic the properties of the continuous ones. This remark suggests a staggered grid arrangement for the unknowns (see Harlow and Welch [12]). Furthermore, the discretization of the Navier-Stokes equations on three-dimensional generalized curvilinear co-ordinates systems is investigated. There are a lot of works that have been performed in this context (however, to the knowledge of the authors, none makes use of the velocity–vorticity formulation, except in the two-dimensional case [13]). The crucial point for the numerical consistency of the solution methods turns out to be the choice of the basis on which the vector unknowns, say the velocity, and/or the equations are projected. Whereas the use of Cartesian vector

components may present some drawbacks, the choice of the covariant and contravariant as dependent variables is natural (see the discussion by Shyy and Vu [14]). Since the primitive Navier–Stokes equations will be discretized (see below), the same difficulties will be faced as those of the people who have dealt with the Navier–Stokes equations on curvilinear co-ordinates systems. An important difficulty arises in the expression of the viscous terms. Indeed, it involves the spatial differentiation of the metric tensor yielding the well-known Cristoffel symbols. The storage of this third-order tensor is undoubtedly restrictive in the three-dimensional case, in terms of computational storage. Moreover, its computation may be inaccurate, and it is no longer straightforward to obtain a conservative form of the equation [15,16].

This paper is organized as follows. First, the conditions for the equivalence between the formulations in the continuous case are recalled. The discrete unknowns and differential operators are then introduced, yielding the discrete Navier–Stokes equations cast in primitive form. A discrete equivalent velocity–vorticity formulation is then derived. Special attention is paid to the discretization of both convective and viscous terms in order to avoid the use of Cristoffel symbols. In the third part, focus is on the methodology that is used to solve the resultant set of equations, emphasizing the way to enforce both the boundary conditions and the pressure uniformity condition in the case of multiply connected domains. Finally, some numerical tests are performed in two- and three-dimensional cases, in order to check the validity and the efficiency of the method.

2. GOVERNING EQUATIONS

Consider the flow of an incompressible viscous fluid in an open bounded domain Ω, with boundary $\partial\Omega$. The non-dimensional primitive Navier–Stokes equations read as

$$\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{\omega} \times \boldsymbol{v} = -\nabla p_t - \frac{1}{Re} \nabla \times \boldsymbol{\omega} \qquad \text{in } \Omega, \tag{1}$$

$$\nabla \cdot \boldsymbol{v} = 0 \qquad \text{in } \Omega, \tag{2}$$

$$\boldsymbol{v} \cdot \boldsymbol{n} = \boldsymbol{b} \cdot \boldsymbol{n} \qquad \text{on } \partial\Omega, \tag{3}$$

$$\boldsymbol{v} \times \boldsymbol{n} = \boldsymbol{b} \times \boldsymbol{n} \qquad \text{on } \partial\Omega, \tag{4}$$

where Re is the usual Reynolds number, $\boldsymbol{v}$ is the velocity, the vector $\boldsymbol{\omega}$ is the vorticity, i.e. the curl of the velocity, and $\boldsymbol{n}$ denotes the unit outer vector normal to the boundary $\partial\Omega$. The dynamic pressure p_t is related to the static pressure p by

$$p_t = p + \frac{1}{2}\boldsymbol{v}^2.$$

The prescribed velocity $\boldsymbol{b}$ on the boundary has to satisfy the following compatibility condition:

$$\iint_{\partial\Omega} \boldsymbol{b} \cdot \boldsymbol{n} \, \mathrm{d}S = 0. \tag{5}$$

In order to remove any explicit reference to the pressure, one can take the curl of (1) to obtain a convection–diffusion equation for the vorticity $\boldsymbol{\omega}$, which, together with the continuity equation and the definition of the vorticity, yields the so-called velocity–vorticity formulation, as

$$\frac{\partial \boldsymbol{\omega}}{\partial \tau} + \nabla \times (\boldsymbol{\omega} \times \boldsymbol{v}) = -\frac{1}{Re} \nabla \times (\nabla \times \boldsymbol{\omega}) \qquad \text{in } \Omega, \tag{6}$$

$$\nabla \times \boldsymbol{v} = \boldsymbol{\omega} \qquad \text{in } \Omega, \tag{7}$$

$$\nabla \cdot \boldsymbol{v} = 0 \qquad \text{in } \Omega, \tag{8}$$

$$\boldsymbol{v} \cdot \boldsymbol{n} = \boldsymbol{b} \cdot \boldsymbol{n} \qquad \text{on } \partial\Omega, \tag{9}$$

$$\boldsymbol{v} \times \boldsymbol{n} = \boldsymbol{b} \times \boldsymbol{n} \qquad \text{on } \partial\Omega, \tag{10}$$

which is always subject to the compatibility condition (5).

It was proved (see Daube *et al.* [4]) that this formulation is equivalent to the primitive equations (Equations (1)–(4)) provided that additional conditions, hereafter referred as pressure uniformity conditions, are satisfied in the case the domain Ω is p-multiply connected. For this purpose, let us define p independent arbitrary irreducible loops $\{\gamma_i\}_{i=1,\ldots,p}$. Along each of these loops, the following condition,

$$\oint_{\gamma_i} \left(\frac{\partial \boldsymbol{v}}{\partial t} + \boldsymbol{\omega} \times \boldsymbol{v} + \frac{1}{Re} \nabla \times \boldsymbol{\omega} \right) \cdot \mathrm{d}\boldsymbol{l} = 0, \tag{11}$$

has to be satisfied. They ensure that the pressure is a single valued function in the whole domain [4]. In this paper, it is assumed without loss of generality, that the domain is at most 1-multiply connected.

To numerically solve the problem, most of the authors discretize the $\boldsymbol{v}$–$\boldsymbol{\omega}$ set of equations. They are then able to compute the dynamics of the vorticity in a discrete sense. Nevertheless, it is not certain that this formulation is strictly equivalent to the discrete formulation of the primitive one. In other words, it is not enough just to recover the pressure since the equivalence between the discrete formulations is not ensured. The goal of this paper is to develop a methodology consisting of the discretization of the initial problem (1–4), and a proper discrete equivalence with the velocity–vorticity formulation.

3. DISCRETE PROBLEM

In this section the definition of the unknowns and of the differential operators are introduced. The choice of the basis on which the dependent vector variables are expressed and the momentum equation projected, is crucial to ensure the discrete equivalence with the velocity–vorticity formulation. The spatial discretization of the convective and viscous terms in the primitive formulation is performed independently of the velocity–vorticity formulation. The reader is referred Aris [17] for an introduction to curvilinear co-ordinate systems and tensor analysis.

3.1. Grid arrangement and discrete components

Assume that the domain Ω can be mapped onto a rectangle by a boundary fitted system of co-ordinates (ξ^1, ξ^2, ξ^3). Without loss of generality, the transformed domain is divided into uniform elementary cells with intervals of $\Delta\xi^i = 1$ $(i = 1, 2, 3)$.

A continuous vector field $\boldsymbol{v}$ may be defined either by its covariant components v_j or by its contravariant components v^i [17]. These two sets of components are related through the basis transformation, which reads in the continuous case as

$$v^i = g^{ij} v_j \qquad \text{(with summation over repeated indices)},$$

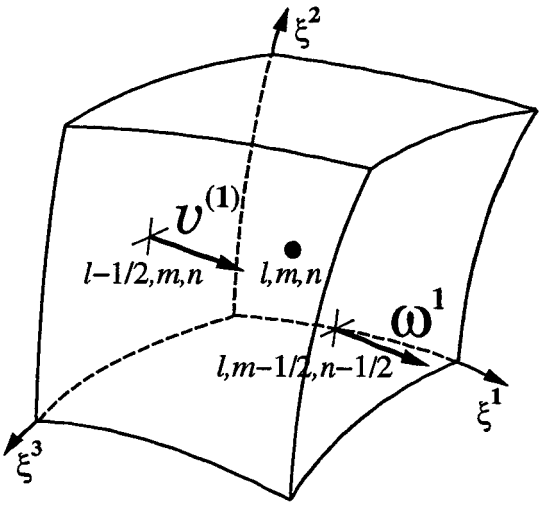

Figure 1. The definition of a cell.

where g^{ij} are the terms of the contravariant metric tensor [17].

In the discretization, let ${}_h\boldsymbol{v} = (v_1, v_2, v_3)$ denote the set of the covariant vector components, and let ${}^h\boldsymbol{v} = (v^1, v^2, v^3)$ denote the set of contravariant ones. A linear interpolation operator ${}^h\boldsymbol{P}$ must be introduced, which plays a role similar to the metric tensor g^{ij} such that

$${}^h\boldsymbol{v} = {}^h\boldsymbol{P}({}_h\boldsymbol{v}).$$

For a discrete vector field $\boldsymbol{v}$, this interpolation operator relates its discrete covariant components v_j to its discrete contravariant components v^i by

$$v^i = P^i(v_1, v_2, v_3), \quad i = 1, 2, 3.$$

In order to ensure a conservative discretization of the usual differential operators, a MAC-arrangement is adopted for the unknowns. A scalar field (for instance the pressure p) is

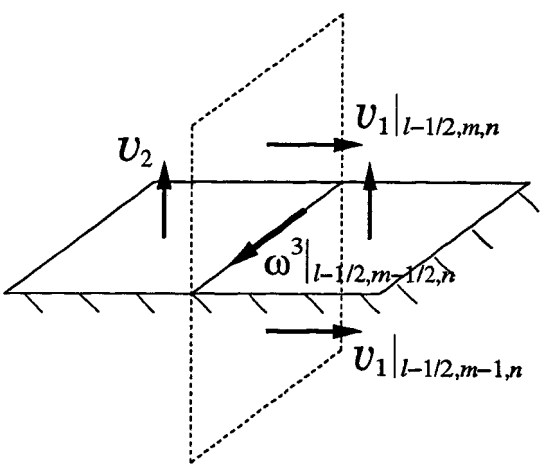

Figure 2. Velocity components at the boundary $\xi^2 = \text{constant}$.

located at cell centers that are referred by integer indices (l, m, n). Both covariant (v_j) and contravariant (v^i) components of the velocity vector field $\boldsymbol{v}$ are located at the centers of the faces $\xi^i = \text{constant}$ ($i = 1, 2, 3$), which are referred by the same indices, one being shifted by 1/2 (see Figure 1).

3.2. Discrete differentiation operators

Following the approach by Yang *et al.* [18] (also see Rosenfeld *et al.* [19]), the usual differential operators are introduced in strong conservative form.

First the gradient operator ${}_h\boldsymbol{G}$ acts on a scalar field ϕ. Its covariant components $G_j(\phi)$ are located at face centers; for instance, the ξ^1 component reads as

$$G_1(\phi)\big|_{l-1/2,m,n} = \phi\big|_{l,m,n} - \phi\big|_{l-1,m,n}.$$

This discretization obviously ensures that the discrete circulation of the gradient of ϕ along a path linking adjacent cells centers from a point A to a point B is equal to $\phi(B) - \phi(A)$.

To define the discrete divergence and curl operators, a finite volume-type discretization is adopted, which is obtained by writing the discrete analogs of the Gauss and Stokes theorems on an elementary cell. By construction, this discretization is conservative, i.e. the contribution of two adjacent cells on an interface identically cancel.

The curl operator ${}^h\boldsymbol{R}$ acts on a vector field $\boldsymbol{v}$ which is known by its covariant components v_j. The contravariant components ω^i of ${}^h\boldsymbol{R}({}_h\boldsymbol{v})$ are located at the edge centers (Figure 1); for instance, the ξ^1 component reads as

$$\begin{aligned} R^1({}_h\boldsymbol{v})\big|_{l,m-1/2,n-1/2} &= \omega^1\big|_{l,m-1/2,n-1/2} \\ &= [(v_3\big|_{l,m,n-1/2} - v_3\big|_{l,m-1,n-1/2}) - (v_2\big|_{l,m-1/2,n} - v_2\big|_{l,m-1/2,n-1})] \\ &\quad /\sqrt{g}\,\big|_{l,m-1/2,n-1/2}, \end{aligned}$$

where $\sqrt{g}$ is the discrete counterpart of the Jacobian of the co-ordinates transformation.

The divergence operator D acts on a vector field $\boldsymbol{v}$ which is known by its contravariant components v^i. The scalar $D({}^h\boldsymbol{v})$ is defined at the cell centers by

$$\begin{aligned} D({}^h\boldsymbol{v})\big|_{l,m,n} = (&\sqrt{g}\,v^1\big|_{l+1/2,m,n} - \sqrt{g}\,v^1\big|_{l-1/2,m,n} + \sqrt{g}\,v^2\big|_{l,m+1/2,n} \\ &- \sqrt{g}\,v^2\big|_{l,m-1/2,n} + \sqrt{g}\,v^3\big|_{l,m,n+1/2} - \sqrt{g}v^3\big|_{l,m,n-1/2})/\sqrt{g}\big|_{l,m,n}. \end{aligned}$$

These definitions imply that discrete counterparts of well-known vector identities are valid. It is a simple matter to show that the following relation,

$${}^h\boldsymbol{R}({}_h\boldsymbol{G}(\phi)) = 0,$$

is algebraically satisfied for every scalar function ϕ. It also implies that the circulation of the gradient along any loop γ linking adjacent cell centers identically cancel, i.e.

$$\sum_{\gamma} G_j(\phi) = 0, \tag{12}$$

where Σ_γ denotes the summation of the covariant components along γ.

There is also a need to define a second divergence operator $\tilde{D}$, which is similar to D but is computed at vertices ($l-1/2$, $m-1/2$, $n-1/2$), and which acts on the contravariant components of the vorticity ω^i. It can easily be shown that the relation

$$\tilde{D}({}^h\boldsymbol{R}({}_h\boldsymbol{v})) = 0 \tag{13}$$

algebraically holds.

Later on it will be shown that by applying these operators to the discretized form of the primitive Navier–Stokes equations, a readily equivalent discrete velocity–vorticity formulation of these equations can be obtained. However, the following proposition, which will be useful in achieving this goal, must first be proved.

Proposition 1

Consider a 1-multiply connected domain Ω. Let $\boldsymbol{W}$ be a discrete vector field defined at the face centers by its covariant components W_j, and let γ be an irreducible loop linking adjacent cells centers. The following equivalence holds:

$$\left({}^h\boldsymbol{R}({}_h\boldsymbol{W}) = 0 \quad \text{and} \quad \sum_{\gamma} W_j = 0 \right) \Leftrightarrow$$

(There exists a discrete scalar field ϕ such that ${}_h\boldsymbol{W} = {}_h\boldsymbol{G}(\phi)$).

Proof

Owing to the fact that the curl of the gradient identically cancels in the discrete sense, as shown before, the reverse implication is straightforward. The proof of the direct implication is exactly the same as in the continuous case, making use of finite sums instead of integrals. □

The consequence of this proposition is that a scalar function ϕ, whose discrete gradient is known, can be defined starting from an arbitrary value, ϕ_0, at an arbitrary point, M_0, by a simple summation along an arbitrary path, γ, linking M_0 to M, as

$$\phi(M) = \phi_0 + \sum_{\gamma} W_j.$$

The fact should be emphasized that, in the case of a multiply connected domain, this summation will no longer be independent of the path γ if the assumptions of Proposition 1 are violated, in particular if the discrete circulation of $\boldsymbol{W}$ does not identically cancel.

3.3. Discrete equations

In this section, concentration will be on the discretization of the primitive Navier–Stokes equations (1–4). Owing to the fact that an equation for the curl of the velocity that is known by its contravariant components ω^i must be written, and that the curl of the gradient must disappear, it seems natural to project Equation (1) on the covariant basis. This results in a set of three equations for the covariant velocity components v_j, located at the face centers, where they are defined as before. A discrete counterpart of the incompressiblity condition (2) is obtained simply by the cancellation of the operator D, and makes use of the contravariant components v^i. On the boundaries, the normal velocity component is prescribed by means of the contravariant components (Equation (3)), while the two tangential components are prescribed by means of the covariant components (Equation (4)). To close the set of equations, recall that covariant and contravariant components are related through the interpolation operator ${}^h\boldsymbol{P}$ introduced before. To summarize, the following system of equations results,

$$\frac{\partial v_j}{\partial t} + (\boldsymbol{\omega} \times \boldsymbol{v})_j = -G_j(p_t) - \frac{1}{Re}(\nabla \times \boldsymbol{\omega})_j, \quad j = 1, 2, 3 \qquad \text{in } \Omega, \tag{14}$$

$$D({}^h\boldsymbol{v}) = 0 \qquad \text{in } \Omega, \tag{15}$$

$$v^p = b^p, \quad p = 1, 2 \text{ or } 3 \qquad \text{on } \partial\Omega, \tag{16}$$

$$v_q = b_q, \quad q \neq p \qquad \text{on } \Omega, \tag{17}$$

$$\text{with } v^i = P^i(v_1, v_2, v_3), \quad i = 1, 2, 3 \tag{18}$$

where the boundary scalar field b^p is subject to the discrete compatibility condition,

$$\sum_{\partial\Omega} b^p = 0. \tag{19}$$

The spatial and temporal schemes used to evaluate the unsteady, convective and viscous terms of Equation (14) will be discussed later.

3.4. Implementation of the boundary conditions

Here, and in the following, the index p refers to the (contravariant) component normal to the boundary and q refers to the (covariant) components tangent to the boundary. Note that the no-slip condition (17) for the tangential velocity components is written on cell edges by summing the corresponding components located on each side of the boundary. For instance, Equation (17) for the ξ^1 component on the boundary $\xi^2 = \text{constant}$, reads as (see Figure 2)

$$\begin{aligned} v_1|_{l-1/2,m-1/2,n} &= \frac{1}{2}\left(v_1|_{l-1/2,m,n} + v_1|_{l-1/2,m-1,n}\right) \\ &= b_1|_{l-1/2,m-1/2,n}. \end{aligned}$$

Combining this relationship with the definition of the ξ^3 tangential component of the vorticity on the boundary

$$\sqrt{g}\,\omega^3|_{l-1/2,m-1/2,n} = (v_2|_{l,m-1/2,n} - v_2|_{l-1,m-1/2,n}) - (v_1|_{l-1/2,m,n} - v_1|_{l-1/2,m-1,n}),$$

yields the tangential velocity components on each side of the boundary as

$$v_1|_{l-1/2,m,n} = b_1|_{l-1/2,m-1/2,n} + \frac{1}{2}\left(-\sqrt{g}\,\omega^3|_{l-1/2,m-1/2,n} + (v_2|_{l,m-1/2,n} - v_2|_{l-1,m-1/2,n})\right), \tag{20}$$

$$v_1|_{l-1/2,m-1,n} = b_1|_{l-1/2,m-1/2,n} + \frac{1}{2}\left(+\sqrt{g}\,\omega^3|_{l-1/2,m-1/2,n} - (v_2|_{l,m-1/2,n} - v_2|_{l-1,m-1/2,n})\right), \tag{21}$$

where the covariant components of the velocity that are not parallel to the boundary (in this case v_2) are assumed to be known:

$$v_2|_{l,m-1/2,n} = b_2|_{l,m-1/2,n}. \tag{22}$$

3.5. Discrete equivalence

The existence and uniqueness of a solution for such a system, is not the concern of this paper and will be assumed herein. Now the following proposition, which establishes the discrete equivalence between the formulation, is stated.

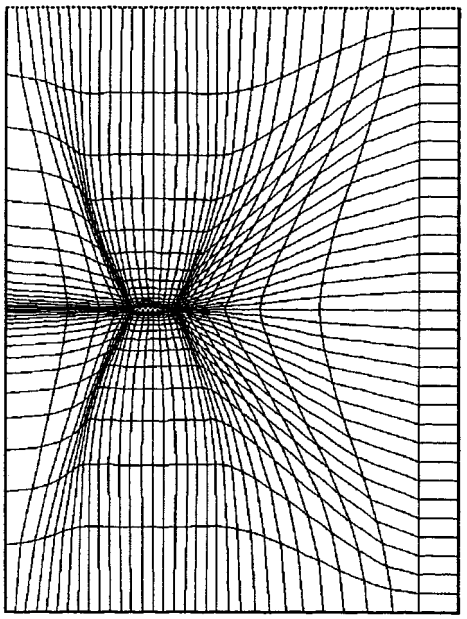

Figure 3. Mesh around the NACA12 airfoil (1 point over 4).

Proposition 2
The set of equations (14–18) is equivalent to the following set:

$$\frac{\partial \omega^i}{\partial t} + R^i({}_h(\boldsymbol{\omega} \times \mathbf{v})) = -\frac{1}{Re} R^i({}_h(\nabla \times \boldsymbol{\omega})), \quad i = 1, 2, 3 \qquad \text{in } \Omega, \tag{23}$$

$$R^i({}_h\mathbf{v}) = \omega^i, \quad i = 1, 2, 3 \qquad \text{in } \Omega, \tag{24}$$

$$D({}^h\mathbf{v}) = 0 \qquad \text{in } \Omega, \tag{25}$$

$$v^p = b^p, \quad p = 1, 2 \text{ or } 3 \qquad \text{on } \partial\Omega, \tag{26}$$

$$v_q = b_q, \quad q \neq p \qquad \text{on } \partial\Omega, \tag{27}$$

$$\text{with } v^i = P^i(v_1, v_2, v_3), \quad i = 1, 2, 3. \tag{218}$$

If the domain is 1-multiply connected, the following relationship

$$\sum_{\gamma} \left(\frac{\partial v_j}{\partial t} + (\boldsymbol{\omega} \times \mathbf{v})_j + \frac{1}{Re} (\nabla \times \boldsymbol{\omega})_j \right) = 0, \tag{29}$$

has to be enforced on one arbitrary irreducible loop, γ, defined as in Proposition 1.

Proof
Implication is straightforward and can be obtained by applying the discrete curl operator ${}^h\boldsymbol{R}$ to Equation (14).
Conversely, the circulation condition (29) for the quantity

$$W_j = -\frac{\partial v_j}{\partial t} - (\boldsymbol{\omega} \times \mathbf{v})_j - \frac{1}{Re} (\nabla \times \boldsymbol{\omega})_j,$$

reads as

$$\sum_{\gamma} W_j = 0.$$

Furthermore, Equation (23) reduces to

$${}^{h}\boldsymbol{R}({}_{h}\boldsymbol{W}) = 0.$$

Use can then be made of the Proposition 1, thus there exists a scalar field, namely the dynamic pressure p_t, satisfying

$${}_{h}\boldsymbol{W} = {}_{h}\boldsymbol{G}(p_t),$$

which is readily Equation (14) and proves the proposition. □

At this point, consider that the whole problem (23–29) has been solved numerically. It is now time to recover all the quantities of the primitive formulation. Since $(\boldsymbol{\omega} \times \boldsymbol{v})_j$ and $(\nabla \times \boldsymbol{\omega})_j$ are known, as well as v_j, the following can be set:

$$G_j(p_t) = W_j = -\frac{\partial v_j}{\partial t} - (\boldsymbol{\omega} \times \boldsymbol{v})_j - \frac{1}{Re}(\nabla \times \boldsymbol{\omega})_j.$$

The dynamic pressure p_t can be calculated over the domain by choosing arbitrarily its value at any point of the domain, and by integrating its gradient, i.e. $\boldsymbol{W}$. Notice that this method is less time consuming than the resolution of a Poisson equation for the dynamic pressure that can be obtained by taking the divergence of $\boldsymbol{W}$ (see Huang *et al.* [20]).

Furthermore, it is a simple matter to show that Equation (29) also holds on any arbitrary loop $\tilde{\gamma}$ defined as γ, since the discrete operators satisfy the Stokes theorem.

Note the fact that, the vorticity will remain solenoidal if it is initially solenoidal. Indeed, applying the divergence operator $\tilde{D}$ to Equation (23) yields

$$\frac{\partial \tilde{D}(\omega^i)}{\partial t} = 0,$$

which proves the result.

It is also noteworthy that the equivalence property stated in Proposition 2, is independent of the discrete spatial and temporal schemes for convective and viscous terms, and for the time derivative that will be introduced later.

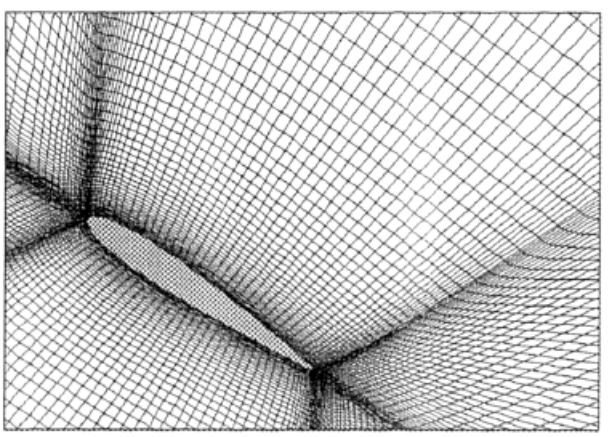

Figure 4. Partial view of the mesh.

3.6. Spatial scheme

This section will focus on the spatial discretization of both convective and viscous terms. The continuous covariant components of the cross product of two vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ are written in terms of their contravariant components a^k and b^l under the form of

$$(\boldsymbol{a} \times \boldsymbol{b})_j = \epsilon_{jkl} a^k b^l \qquad \text{(with summation over repeated indices)},$$

where ϵ_{jkl} is the Levy–Civita tensor [17]. The convective term $(\boldsymbol{\omega} \times \boldsymbol{v})_j$ can then be evaluated by the product of the contravariant components of the vector fields $\boldsymbol{\omega}$ and $\boldsymbol{v}$. Here an upwind Quick scheme is used, which is second-order-accurate in space (see Leonard [21]).

The viscous term involves $\nabla \times \nabla \times \boldsymbol{\omega}$, the discretization of which is not straightforward on non-orthogonal grids. Remember that the discrete curl operator acts on covariant components, and gives contravariant components. Therefore, discretizing the curl of a curl necessarily raises some specific difficulties.

A first solution would be to make use of a non-conservative expression of the curl operator, namely

$$(\nabla \times \boldsymbol{\omega})_j = \sqrt{g}\, \epsilon_{jkl} \left[g^{kp} \left(\frac{\partial \omega^l}{\partial \xi^p} + \Gamma^l_{pq} \omega^q \right) - g^{lp} \left(\frac{\partial \omega^k}{\partial \xi^p} + \Gamma^k_{pq} \omega^q \right) \right],$$

where g^{kp} is the contravariant metric tensor, and Γ^k_{pq} are the Cristoffel symbols that arise in the spatial differentiation of the metric tensor due to the variation of the curvilinear basis. Making use of this formulae overcomes the above mentioned difficulties. However, one now has to store, in addition to the contravariant metric tensor, which is already involved in the operator ${}^h\boldsymbol{P}$, the tensor Γ^k_{pq}, which involves 27 scalar quantities (half if use is made of symmetry property of this tensor). Therefore a simpler approach is developed.

In a first step, the contravariant components of the vorticity are projected onto the covariant basis (using the covariant metric tensor) by means of an operator ${}_h\boldsymbol{P}$. It is now possible to compute the curl of the vorticity in terms of the resulting covariant components. Because of the staggered arrangement of the unknowns, another curl operator, ${}^h\tilde{\boldsymbol{R}}$, is introduced which is similar to ${}^h\boldsymbol{R}$ but whose contravariant components are defined at face centers. These components will in turn be projected onto the covariant basis by means of a third projection operator ${}_h\tilde{\boldsymbol{P}}$. These successive operations may appear to be complicated, but they are easily performed and they only need the storage of the covariant metric tensor, which will be used in both projection operators ${}_h\boldsymbol{P}$ and ${}_h\tilde{\boldsymbol{P}}$. To sum up, achieved is:

$$(\nabla \times \boldsymbol{\omega})_j = \tilde{P}_j({}^h\tilde{\boldsymbol{R}}({}_h\boldsymbol{P}({}^h\boldsymbol{\omega}))).$$

Note that the differentiation of the metric tensor is implicitly done by the central differences involved by ${}^h\tilde{\boldsymbol{R}}$. The resultant operator reduces to the usual second-order central difference scheme in the case of an orthogonal co-ordinates system.

4. SOLUTION METHODOLOGY

In the above, it has been stated that the discrete equations are to be solved regardless of the time level at which the different equations may be written. Now the solution methodology is developed for this set of equations, including the boundary conditions and pressure uniformity, so that they are all satisfied at the same time level n.

Figure 5. Experimental visualization [23].

The resolution of these reduces to a sequence of two steps. First, the vorticity transport equation is solved with appropriate boundary conditions that will be discussed later. Then, the velocity components are calculated by solving the so-called div-curl problem, which consists of finding the velocity field when its curl, divergence and normal component on the boundary (and circulation around an arbitrary irreducible loop in case of a 1-multiply connected domain) are known. In the context of non-orthogonal curvilinear co-ordinates, this last problem has been successfully solved by the authors [6]. For the sake of completeness of the present paper, the employed methodology will be briefly recalled. Also, a method to achieve the pressure uniformity will be addressed, which couples the momentum equation with the kinematic div-curl problem.

4.1. Temporal scheme and transport equation

The momentum equation (14) at the present time level n reads as

$$\frac{\partial v_j^{(n)}}{\partial t} + (\boldsymbol{\omega} \times \boldsymbol{v})_j^{(n)} = -G_j(p_t^{(n)}) - \frac{1}{Re}(\nabla \times \boldsymbol{\omega})_j^{(n)}.$$

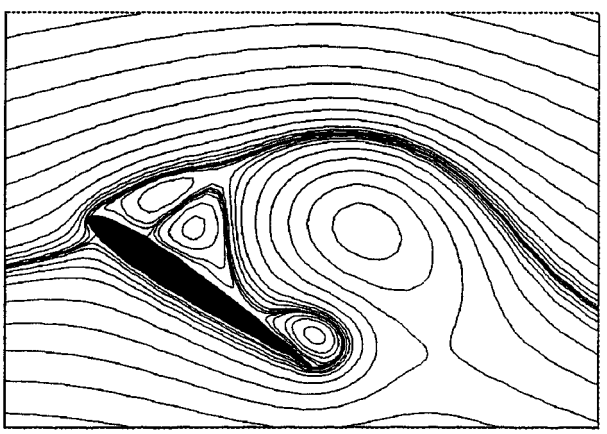

Figure 6. Explicit statement.

The time derivative is approximated by a second-order backward scheme, as

$$\frac{\partial v_j^{(n)}}{\partial t} = \frac{3v_j^{(n)} - 4v_j^{(n-1)} + v_j^{(n-2)}}{2\Delta t}, \tag{30}$$

where $v_j^{(m)}$ denotes the covariant velocity components at the time level m, and Δt is the time step. Since the value of $\boldsymbol{\omega}^{(n)}$ is unknown, as well as $\boldsymbol{v}^{(n)}$ (because it depends on $\boldsymbol{\omega}^{(n)}$ itself), it has been chosen to explicitly evaluate by the convective term using a second-order-accurate Adams–Bashforth extrapolation,

$$(\boldsymbol{\omega} \times \boldsymbol{v})_j^{(n)} \simeq 2(\boldsymbol{\omega} \times \boldsymbol{v})_j^{(n-1)} - (\boldsymbol{\omega} \times \boldsymbol{v})_j^{(n-2)}.$$

This term could also be linearized by extrapolating the velocity at time level n, the vorticity being implicitly treated in the course of the resolution of its transport equation which would yield

$$(\boldsymbol{\omega} \times \boldsymbol{v})_j^{(n)} \simeq \boldsymbol{\omega}^n \times ((2\boldsymbol{v})_j^{(n-1)} - (\boldsymbol{v})_j^{(n-2)}).$$

Taking the curl (${}^h\boldsymbol{R}$) of the momentum equation yields the following vorticity transport equation,

$$\frac{\partial \omega^{i(n)}}{\partial t} + R^i({}_h(\boldsymbol{\omega} \times \boldsymbol{v})^{(n)}) = -\frac{1}{Re} R^i({}_h(\nabla \times \boldsymbol{\omega})^{(n)}).$$

Owing to the fact that a MAC-grid is used and that the diffusion term is written in the form $\nabla \times \nabla \times \boldsymbol{\omega}$, this equation is well-posed if the tangential vorticity components $\boldsymbol{\omega}^{(n)} \times \boldsymbol{n}$ on the boundary are known. At this point, remember that the determination of this distribution is coupled with the resolution of the kinematic problem (24–26), in order to satisfy the no-slip condition (27). This coupling may be solved by means of a matrix influence technique [5]. In this case, the boundary vorticity is calculated so that the velocity at the present time level exactly satisfies the no-slip boundary condition, i.e.

$$\omega^{q+1,(n)} \text{ is prescribed on } \partial\Omega \text{ such that } v_q^{(n)} = b_q^{(n)},$$

where $q+1$ denotes the index of the contravariant vorticity component, tangent to the boundary and normal to the covariant velocity component v_q (see Section 3.4).

Unfortunately, this method is generally not affordable for 3D problems. Consequently, it was decided to decouple the resolution of the vorticity transport equation from the resolution of the kinematic problem. For this purpose, the first one is supplemented by Dirichlet-type conditions. As was done by several authors, the boundary vorticity is approached in an explicit manner, taking into account the no-slip condition (see E and Liu [22] for a review of those techniques). This relationship reads as

$$\omega^{q+1,(n)} \text{ is prescribed on } \partial\Omega \text{ such that } v_q^{(*)} = b_q^{(*)},$$

where $v_q^{(*)}$ denotes any temporal extrapolation of the tangential boundary velocity from previous time levels $(n-1, n-2, \ldots)$. In this paper, use is made of the following simple approximation $v_q^{(*)} = v_q^{(n-1)}$ first-order-accurate in time. It is introduced in either Equation (20) or (21) in order to compute the actual tangential vorticity component $\omega^{q+1,(n)}$. This evidently leads to a first-order error in the satisfaction of the no-slip condition at the present time level.

It will be checked in the 2D case (see Section 5.1.), that the use of this extrapolation does not yield significant discrepancies with the results obtained using an influence matrix technique.

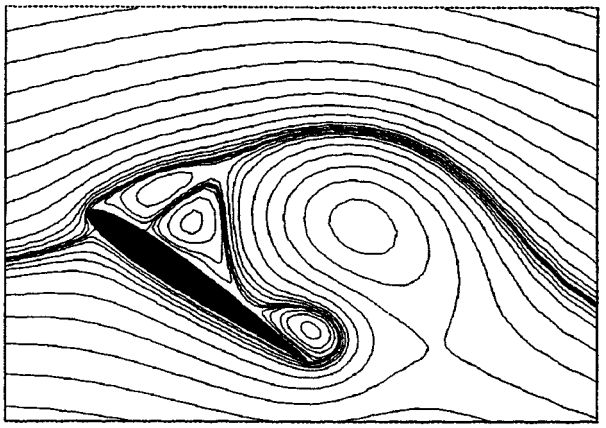

Figure 7. Influence matrix technique.

4.2. Pressure uniformity condition

If some approximations on the boundary vorticity distribution may be acceptable since they produce only truncation errors in the computation of the flow field, it is no longer the case for the pressure uniformity condition, which must be exactly (within machine accuracy) satisfied, in order to ensure the equivalence between the two discrete formulations. If it is not the case, the computation of the pressure variation between two points, A and B, may be dependent on the chosen integration path AB. In particular, the integration of the 'pressure gradient' along a loop starting from a point A could lead to a physically meaningless pressure gap when coming back to the starting point.

It was shown in the previous paragraph that the vorticity transport equation supplemented by Dirichlet-type conditions for the tangential components of the vorticity, is well posed. Therefore, the pressure uniformity condition (29) has to be included in the computation of the velocity field. So, consider that the vorticity transport equation has been solved. All the terms of Equation (29) are known except the time derivative, which is approximated by

$$\sum_{\gamma} \frac{\partial v_j^{(n)}}{\partial t} = \frac{3C_\gamma^{(n)} - 4C_\gamma^{(n-1)} + C_\gamma^{(n-2)}}{2\Delta t},$$

where $C_\gamma^{(m)}$ denotes the circulation of the (covariant) velocity along the loop γ at time level m. Since the first term on the right-hand side, namely the circulation at the present time level, has to be assigned a value C in the course of the resolution of the div-curl problem (see Section 4.3.), C_γ is chosen in order that the pressure uniformity is satisfied. Thus, the required value of C_γ is

$$3C_\gamma^{(n)} = 4C_\gamma^{(n-1)} - C_\gamma^{(n-2)} - 2\Delta t \sum_{\gamma} \left[(\boldsymbol{\omega} \times \boldsymbol{v})_j^{(n)} - \frac{1}{Re} (\nabla \times \boldsymbol{\omega})_j^{(n)} \right]. \tag{32}$$

4.3. Kinematic problem

To complete the solution of the set of equations, one has to find the velocity field $\boldsymbol{v}^{(n)}$ at the present time level. It is defined by the following div-curl problem and depends on the present vorticity and on the prescribed normal velocity on the boundaries

$$
\begin{aligned}
&{}^{h}\boldsymbol{R}({}_{h}\boldsymbol{v}^{(n)}) = {}^{h}\boldsymbol{\omega}^{(n)} \qquad \text{in } \Omega,\\
&D({}^{h}\boldsymbol{v}^{(n)}) = 0 \qquad \text{in } \Omega,\\
&v^{p,(n)} = b^{p,(n)} \qquad \text{on } \partial\Omega,\\
&\text{with } {}^{h}\boldsymbol{v}^{(n)} = {}^{h}\boldsymbol{P}({}_{h}\boldsymbol{v}^{(n)}).
\end{aligned}
$$

It has been shown [6] that the solution of this system exists and is unique if and only if, the compatibility relation (19) holds for the prescribed normal boundary velocity $b^{p,(n)}$ and the mesh satisfies some geometric conditions (which reduces to the fact that the cells are not greatly distorted). Note that it has already been shown that the vorticity field remains solenoidal (Section 3.5).

The resolution of the div-curl problem is performed in two steps by using a Helmholtz decomposition:

1. look for **one** vector field $\boldsymbol{w}$, the curl of which is equal to the vorticity $\boldsymbol{\omega}^{(n)}$.
2. The vector $\boldsymbol{w}$ is projected onto the space of divergence-free vector fields by adding the gradient of a suitable scalar function ϕ:

$$
{}_{h}\boldsymbol{v}^{(n)} = {}_{h}\boldsymbol{w} + {}_{h}\boldsymbol{G}(\phi). \tag{33}
$$

Moreover, if the domain is 1-multiply connected, the circulation of the velocity along one arbitrary irreducible loop γ' must be assigned a value $C_{\gamma'}^{(n)}$,

$$
\sum_{\gamma} v_j^{(n)} = C_{\gamma'}^{(n)}.
$$

The obvious choice for γ' and $C_{\gamma'}^{(n)}$ is the loop γ and the value defined in Equation (32), which enforces the pressure uniformity condition. It is clear that this value of the circulation along

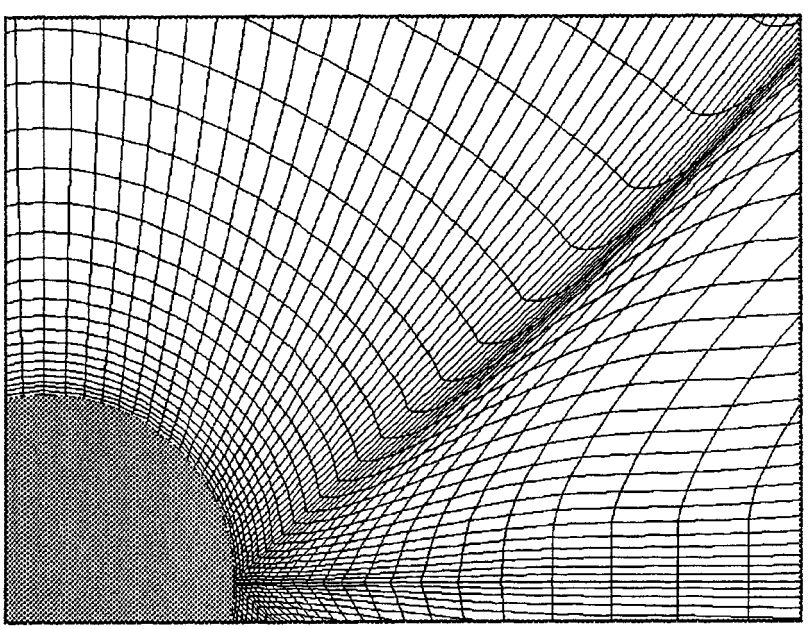

Figure 8. Mesh around the circular cylinder (partial view).

the loop γ has to be imposed in the first step since it will not be modified by adding the gradient of a scalar function.

More precisely, these two steps may be described as

Step 1. The determination of one such vector field $\boldsymbol{w}$ is very easily performed (see Lardat *et al.* [7]). Let $\boldsymbol{w}$ be the vector defined by:

$$\frac{3_h\boldsymbol{w} - 4_h\boldsymbol{v}^{(n-1)} + {}_h\boldsymbol{v}^{(n-2)}}{2\Delta t} = -{}_h(\boldsymbol{\omega} \times \boldsymbol{v})^{(n)} - \frac{1}{Re}{}_h(\nabla \times \boldsymbol{\omega})^{(n)} \qquad \text{in } \Omega,$$

$$w_p = b_p^{(n)} \qquad \text{on } \partial\Omega,$$

where the scalar field $b_p^{(n)}$ is given by Equation (22). It is straightforward to check that this vector field $\boldsymbol{w}$ has the desired property, i.e.

1. That ${}^h\boldsymbol{R}({}_h\boldsymbol{w}) = {}^h\boldsymbol{\omega}^{(n)}$.
2. Its circulation along the loop γ is equal to the value $C_\gamma^{(n)}$ defined by (32).

Also noteworthy is the fact that the computation of $\boldsymbol{w}$ is an explicit one, since the velocity at previous time steps and the vorticity at the actual time step have already been computed.

Step 2. The equation to be solved for ϕ is derived from the divergence-free condition (25) and reads as

$$D({}^h\boldsymbol{P}({}_h\boldsymbol{G}(\phi))) = -D({}^h\boldsymbol{P}({}_h\boldsymbol{w})), \tag{34}$$

supplemented with Neumann-type boundary conditions:

$$P^p({}_h\boldsymbol{G}(\phi)) = 0.$$

Making use of (33), results in $\boldsymbol{v}^{(n)}$ being divergence-free in terms of its contravariant components $v^{i,(n)}$ (Equation (25)) and that the transformation rules (28) hold. All details of this method can be found in Bertagnolio and Daube [6].

4.4. Complexity of the algorithms

The computational effort involved by the proposed method will now be examined in terms of the number of elliptic equations to be solved at each time step.

- The resolution of the vorticity transport equation involves three scalar Helmholtz-type equations, as in the case of the momentum equation when using the primitive variables.
- Noteworthy is the fact that the resolution of the kinematic problem reduces first to the explicit computation of the vector field $\boldsymbol{w}$, which requires a negligible amount of computational time, and then to the resolution of a Poisson equation for the scalar function ϕ.

The computational work required by this algorithm is thus similar to the one that is usually used by methods solving primitive equations. Moreover, this methodology can be seen as a particular kind of fractional step method (see the discussion by Lardat *et al.* [7]).

5. NUMERICAL RESULTS

In this section, some numerical calculations are presented that were performed in order to check the validity of the proposed method, considering different features:

- the ability to deal with strongly distorted grids;
- the ability to deal with non-simply connected domains;

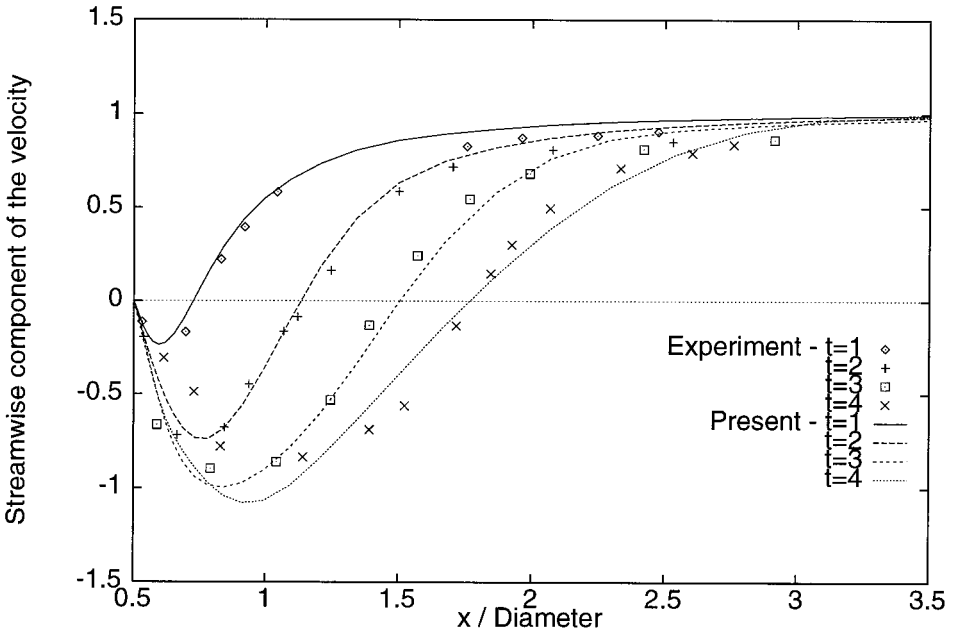

Figure 9. Streamwise velocity in section $z = 0.25$ (experiment by Pineau *et al.* [23]).

– the ability to reproduce specific features of the flows being computed;
– the validity of the explicit estimation of the boundary vorticity of Section 4.1. when compared with a fully coupled resolution by an influence matrix method.

These goals were achieved by comparing, whenever it was possible, with available experimental results. A more detailed assessment of the accuracy of the discrete operators can be found in Bertagnolio and Daube [6].

5.1. Two-dimensional flow around a NACA12 airfoil

The 2D-flow around a NACA12 airfoil has first been computed at Reynolds number of 1000 with an incidence of 34°. The computational mesh shown in Figure 3 involves 128 cells in each direction. A detailed view of this mesh (Figure 4) around the airfoil clearly shows the skewness of the cells.

Figures 6 and 7 display the streamlines, at the same non-dimensional time $t = 3.2$, which are computed using either the explicit method of Section 4.1, to determine the vorticity boundary distribution, or an influence matrix technique similar to the one described in Daube [5]. These results are compared with the experimental visualizations (Figure 5) which were reported by Coutanceau *et al.* [23]. Results show good agreement between the two methods and the experiments. This comparison shows the ability of the method to deal with multiply connected domains and highly distorted grids and also the validity of an explicit estimation of the vorticity boundary distribution.

5.2. Three-dimensional flow around a circular cylinder

The flow around a circular cylinder enclosed between walls has also been computed. The Reynolds number, based on the cylinder diameter and the velocity at the inlet, is $Re = 1000$.

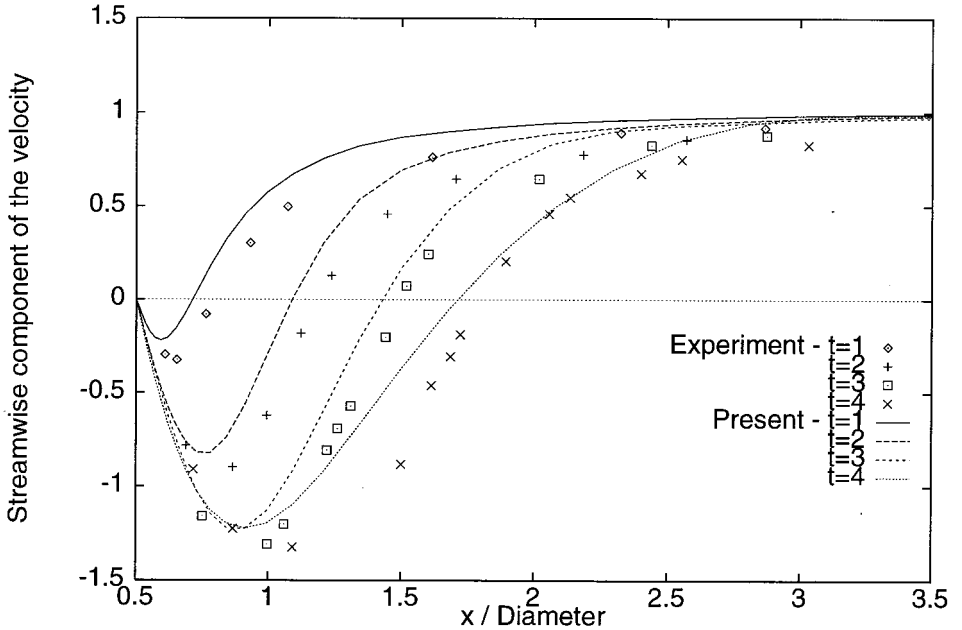

Figure 10. Streamwise velocity in section $z = 2.50$ (experiment by Pineau *et al.* [23]).

The spanwise length is ten times the diameter. This configuration was experimentally studied by Pineau *et al.* [23]

Use is made of a H-type mesh, which involves 88 cells in the streamwise direction, 88 in the spanwise direction, and 64 in the vertical direction. It is partly sketched in Figure 8. Note the

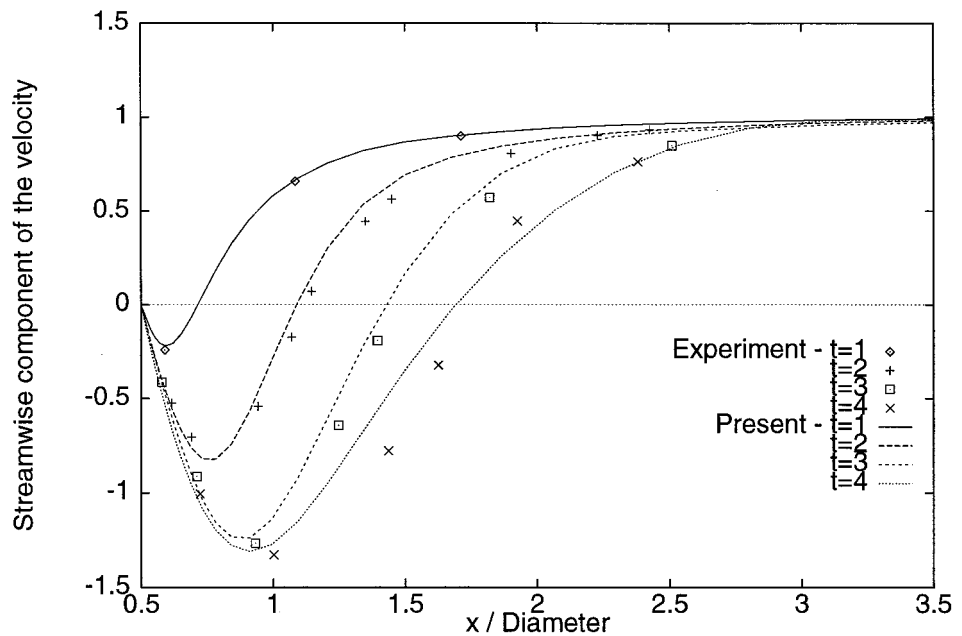

Figure 11. Streamwise velocity in section $z = 5.00$ (experiment by Pineau *et al.* [23]).

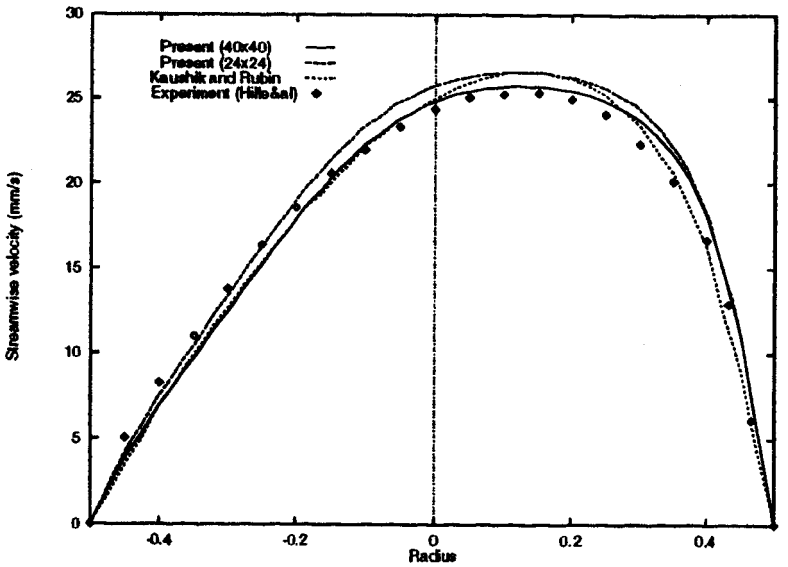

Figure 12. $\theta = 18°$.

distortion of the mesh cells in the vicinity of the trailing edge and of the leading edge of the cylinder. The fact that the domain is 1-multiply connected should be emphasized.

The computed streamwise velocity at different time levels ($t = 1$, 2, 3, 4) versus the distance from the cylinder in the streamwise direction is compared with the experimental data. The results are reported for three stations in the spanwise direction, $z = 0.25$, $z = 2.5$ and $z = 5$ (where z denotes the dimensionless spanwise co-ordinates) in Figure 9, Figure 10 and Figure 11, respectively. Results show good qualitative agreement, except at station $z = 2.5$ and at time level $t = 4$.

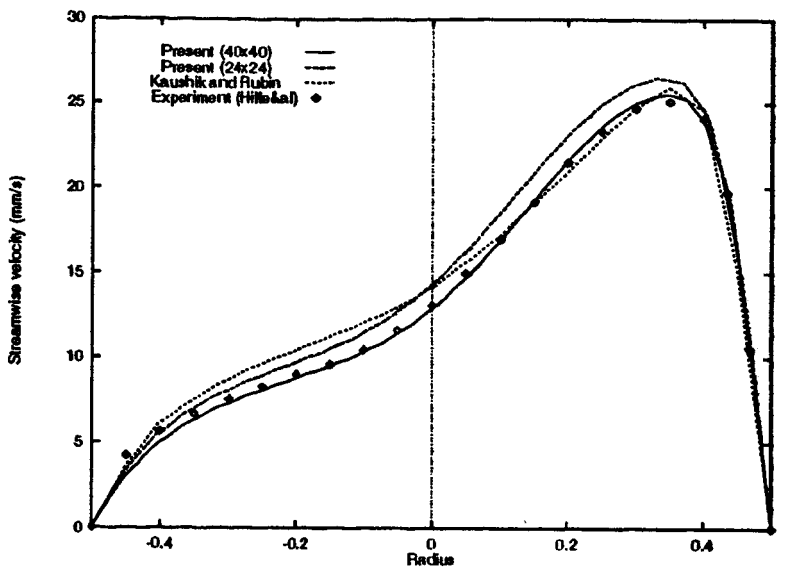

Figure 13. $\theta = 36°$.

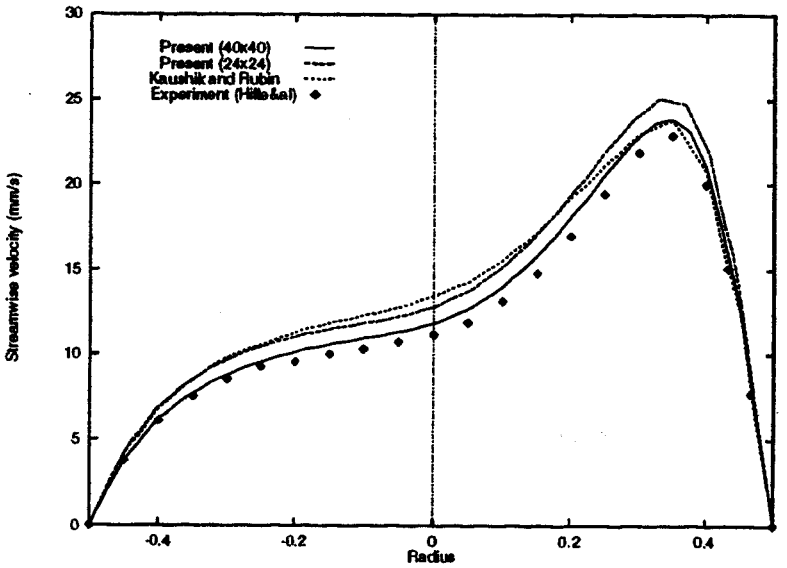

Figure 14. $\theta = 54°$.

5.3. Three-dimensional flow inside a curved duct

In order to test the accuracy and the prediction capability of the method in three dimensions, the flow inside a 180° turn-around bend of square cross-section has been computed, with a constant curvature ratio $\delta = 6.45$ ($\delta = R/d$), where R is the radius of curvature at the middle of the bend and d is the hydraulic diameter of the duct (viz. the width in the case of a square section). The Reynolds number of the flow, based on the hydraulic diameter and the inlet velocity, is $Re = 574$; thus, the Dean number given by $De = Re/\sqrt{\delta}$ is equal to 228.

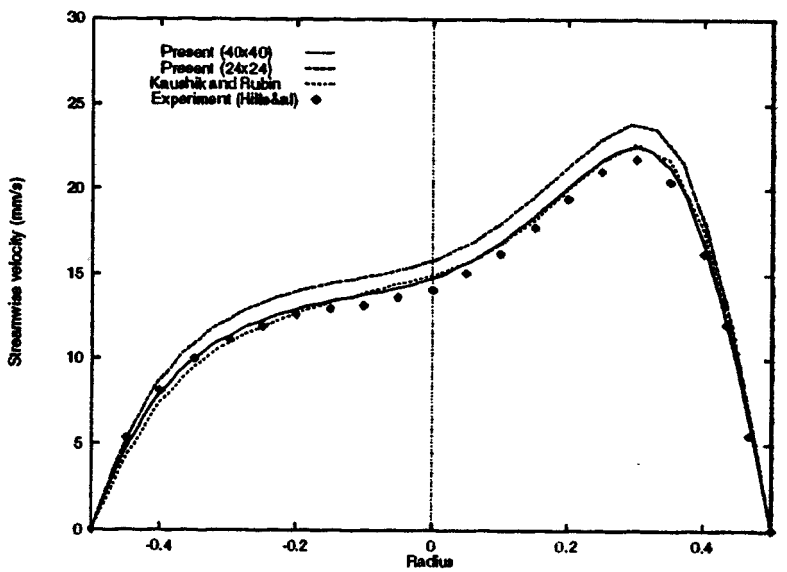

Figure 15. $\theta = 72°$.

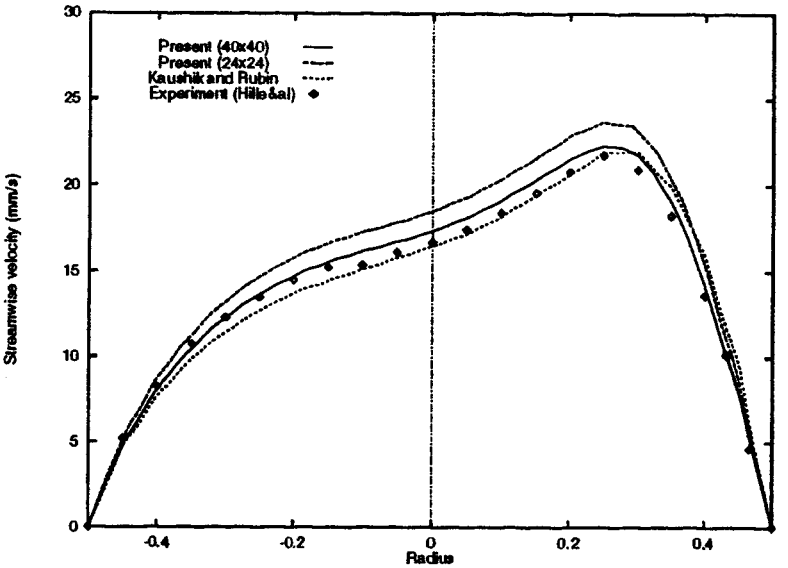

Figure 16. $\theta = 90°$.

The computations were performed on a coarse (fine) grid involving 24×24 (40×40) mesh points in the cross-section, and 160 points streamwise, including 100 inside the bend. Axial velocities, at the symmetry plane and at various streamwise stations (located by the angle θ from the start of the bend), are compared with experimental data from Hille *et al.* [24] in Figures 12–20. The centrifugal forces induced by the curvature of the main flow causes the peak of axial velocity to be shifted towards the outer wall.

The computational results indicates that a fine grid resolution is needed in order to accurately predict the flow field. On these same curves, are plotted the results that were

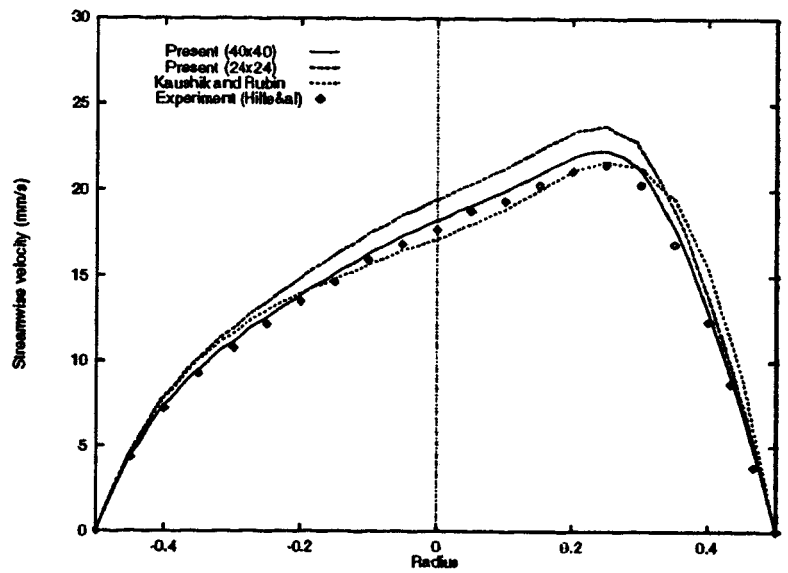

Figure 17. $\theta = 108°$.

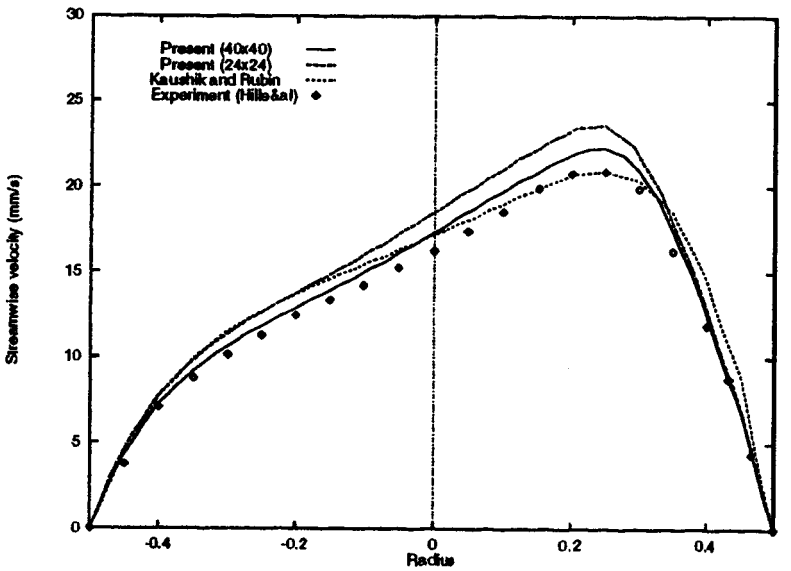

Figure 18. $\theta = 126°$.

computed by Kaushik and Rubin [25] by means of a Navier–Stokes solver using the primitive variables. The present results appear to be in somewhat better agreement with the experimental data at the beginning of the bend.

Furthermore, the curvature of the flow generates streamwise vorticity, resulting in the onset of Dean instabilities. As expected from the experiment, the computation predicts the emergence of two pairs of counter-rotating vortices. However, unlike the experimental flow, the

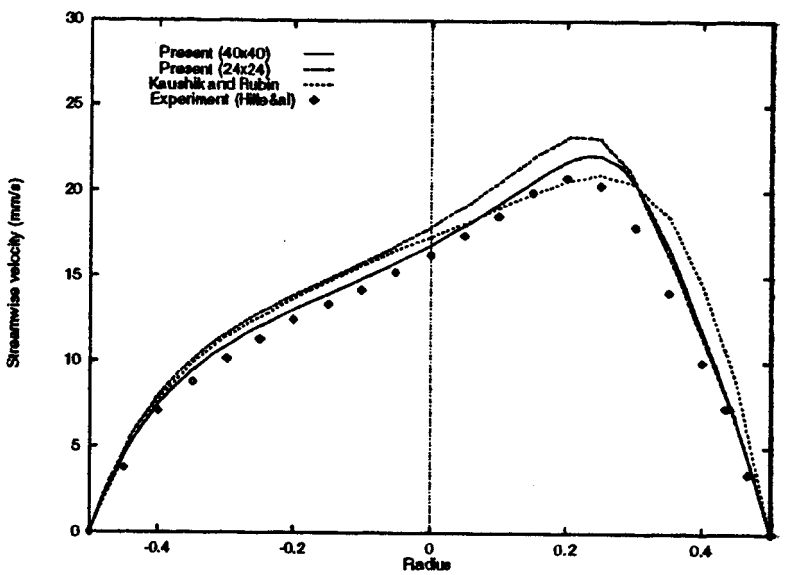

Figure 19. $\theta = 144°$.

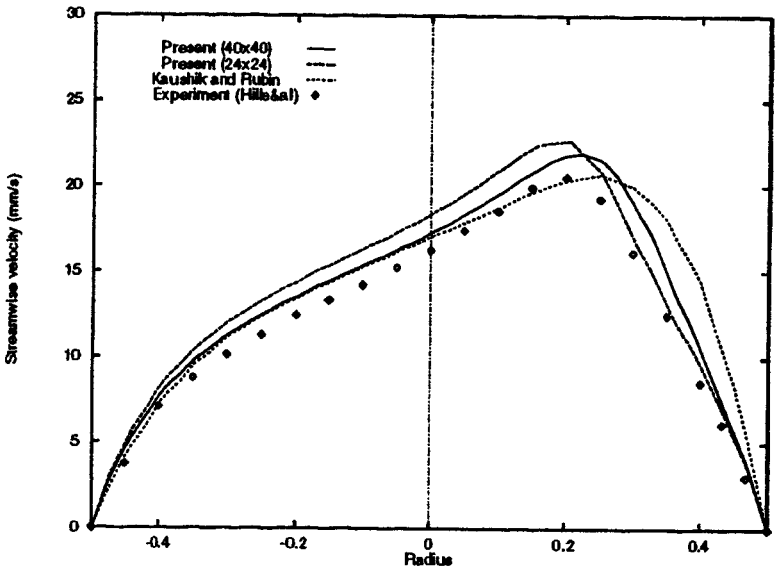

Figure 20. $\theta = 162°$.

computed one is distinguished by perfectly symmetric structures, as shown in Figures 21 and 22 by the streamlines in the cross-section at the station $\theta = 136°$ of the bend. Nevertheless, both computation and experiment are in good qualitative agreement.

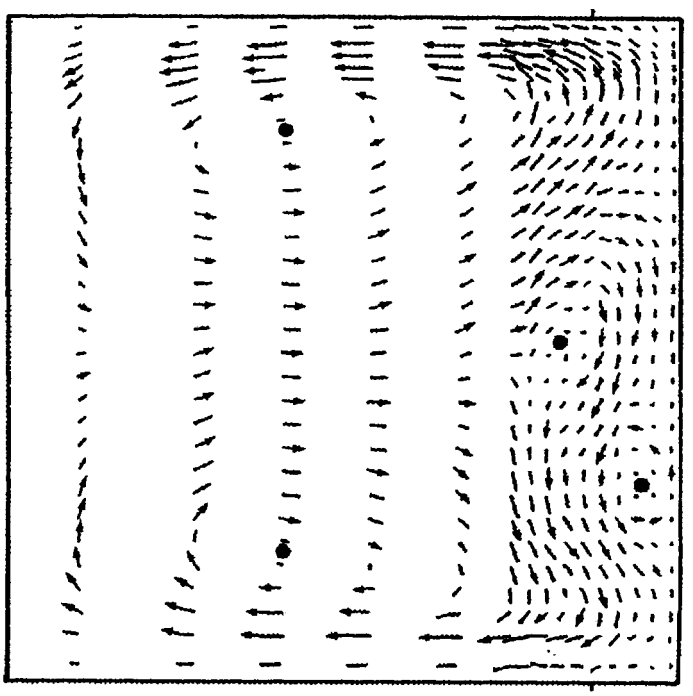

Figure 21. Cross-flow velocity vectors at $\theta = 136°$ (experiment by Hille *et al.* [24]).

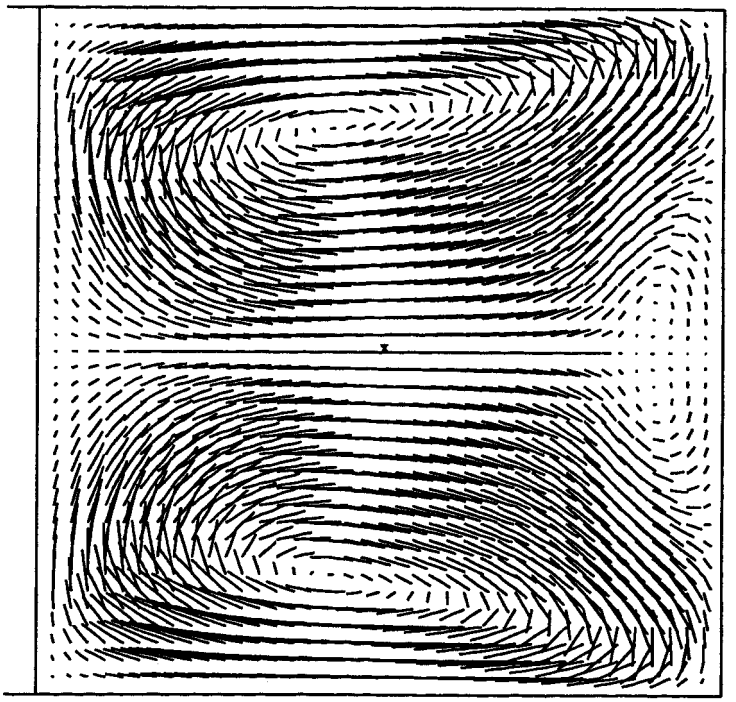

Figure 22. Cross-flow velocity vectors at $\theta = 136°$—Present (40×40).

5.4. 3D lid-driven cavity

The lid-driven cavity is a classical test problem for numerical methods. Presented in this section are some results that were obtained on the benchmak case which was proposed for a

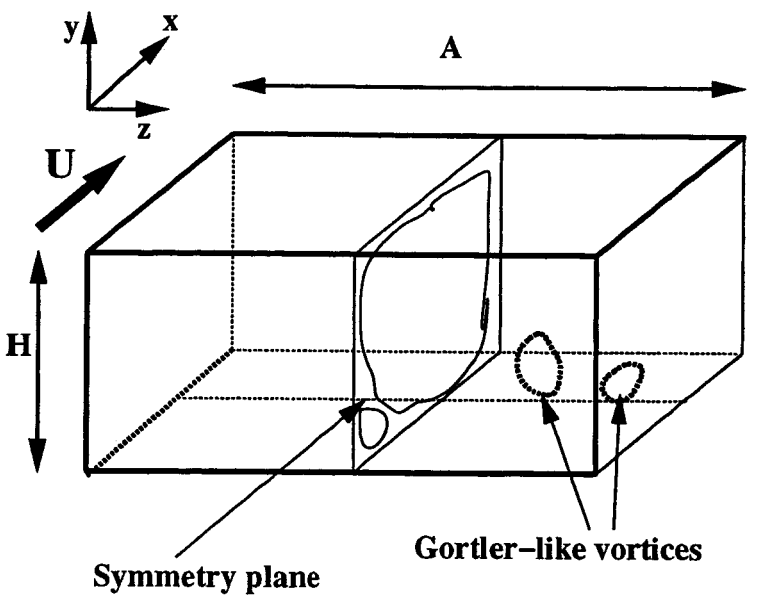

Figure 23. Sketch of the lid-driven cavity.

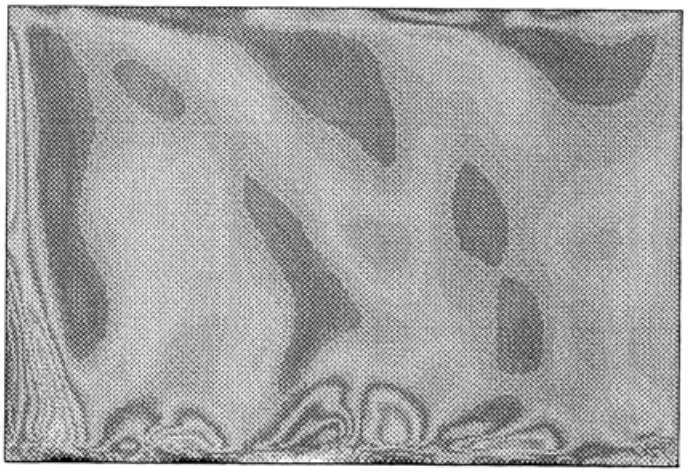

Figure 24. x-component of the vorticity in the plane $x = 0$.

GAMM Workshop in 1992 [26]. The studied configuration is defined by $Re = U_0 H/\nu = 3200$ and $A/H = 3$ (see Figure 23). The goal of this computation is only to check that the proposed method is able to compute some important physical features of the flow, in particular the onset of Görtler-like vortices that are due to the streamlines curvature. For this purpose, a uniform Cartesian grid of $65 \times 65 \times 97$ nodes was used and only one half of the cavity was discretized, thanks to the symmetry of the problem; therefore decreasing the computational effort. The spatial second-order-accuracy of the computations is ensured by the use of a staggered uniform grid in addition to the discretization of the non-linear terms by a classical centered second-order scheme.

These Görtler-like vortices are evidenced in Figure 24, which displays the x-component ω_x of the vorticity in the mid-plane $x = 0$. This picture is in good agreement with the results that were presented during the GAMM Workshop.

For information purposes, Table I displays the maximum, during the unsteady computations, of the divergence of both velocity and vorticity vectors. It can be seen that these two fields are solenoidal within machine accuracy.

6. CONCLUSION

This paper proposes a method for solving the Navier–Stokes equations in non-orthogonal curvilinear co-ordinates, which makes use of the velocity–vorticity formulation. This method is valid both in two and three dimensions. The problem associated with the possible multiply connectivity of the domain has been solved. The numerical results prove the efficiency of the method to deal with multiply connected domains and strong distorted grids, both in 2D and 3D cases.

Table I. Maximum of divergences

$\max\|\nabla\cdot\boldsymbol{v}\|$	$\max\|\nabla\cdot\boldsymbol{\omega}\|$
$2\cdot10^{-12}$	$2\cdot10^{-9}$

The discrete equivalence of the formulation with the primitive Navier–Stokes equations has been shown. The discretization and comments that have been introduced in this paper may thus be useful when dealing with the primitive equations in curvilinear co-ordinates. The algorithm for solving the div-curl problem proposed in Bertagnolio and Daube [6] leads to the resolution of a Poisson equation for the scalar field ϕ, that plays the role of the pressure since its resolution ensures the divergence-free condition for the velocity field. The solution method is then comparable with the usual methods that have been employed in the literature for solving the primitive equations and in particular, the overall amount of work involved by the velocity–vorticity formulation is equivalent to the primitive variables one. The advantages that may be expected by using the velocity–vorticity formulation are thus not so clear and deserves some additional investigations.

ACKNOWLEDGMENTS

The financial support of the Centre National d'Etudes Spatiales and of the Centre National de la Recherche Scientifique through a cosponsored BDI funding for FB is greatly acknowledged. Some of the computations were carried out at Institut du Développement et des Ressources en Informatique Scientifique (supercomputing center of the CNRS).

REFERENCES

1. H. Fasel, 'Investigation of the stability of boundary layers by a finite difference model of the Navier–Stokes equations', *J. Fluid Mech.*, **78**, 355 (1976).
2. P.M. Gresho, 'Incompressible fluid dynamics: some fundamental formulation issues', *Annu. Rev. Fluid Mech.*, **23**, 413–453 (1991).
3. C.G. Speziale, 'On the advantages of the vorticity–velocityof the equations of fluid dynamics', *J. Comput. Phys.*, **73**, 476–480 (1987).
4. O. Daube, J.L. Guermond and A. Sellier, 'Sur la formulation vitesse–tourbillon des équations de Navier–Stokes en écoulement incompressible', *CR Acad. Sci. Paris*, *Sér.II*, **313**, 377–382 (1991).
5. O. Daube, 'Resolution of the 2D Navier–Stokes equations in velocity–vorticity form by means of an influence matrix', *J. Comput. Phys.*, **103**, 402–414 (1992).
6. F. Bertagnolio and O. Daube, 'Solution of the div-curl problem in generalized curvilinear co-ordinates', *J. Comput. Phys.*, **138**, 121–152 (1997).
7. R. Lardat, F. Bertagnolio and O. Daube, 'La formulation vitesse–tourbillon en maillage décalé: une méthode de projection', *CR Acad. Sci. Paris*, *Sér. IIb*, **324**, 747 (1997).
8. O. Daube and Ta Phuoc Loc, 'Etude numérique d'écoulements instationnaires de fluide visqueux incompressible autour de corps profilés par une méthode combinée d'ordre $O(h^4)$ et $O(h^2)$', *J. de Mécanique*, **17**, 651 (1978).
9. S.C.R. Dennis, D.B. Ingham and R.N. Cook, 'Finite difference steady incompressible flows in three dimensions', *J. Comput. Phys.*, **33**, 325–339 (1979).
10. T.B. Gatski, C.E. Grotsch and M.E. Rose, 'The numerical solution of the Navier–Stokes equations for three-dimensional unsteady, incompressible flows by compact scheme', *J. Comput. Phys.*, **82**, 298–329 (1989).
11. G.A. Osswald, K.N. Ghia and U. Ghia, 'A direct algorithm for solution of incompressible three-dimensional unsteady Navier–Stokes equations', *AIAA Paper 871139*, 408 (1987).
12. R.S. Harlow and J.E. Welch, 'Numerical calculation of time-dependent viscous incompressible flow of fluid with free surface', *Phys. Fluids*, **8**, 2182 (1965).
13. G. Pascazio and M. Napolitano, 'A staggered grid finite volume method for the vorticity–velocity equations', *Comp. Fluids*, **25**, 433–446 (1996).
14. W. Shyy and T.C. Vu, 'On the adoption of velocity variable and grid system for fluid flow computation in curvilinear co-ordinates', *J. Comput. Phys.*, **92**, 82–105 (1991).
15. I. Demirdzic, A.D. Gosman, R.I. Issa and M. Peric, 'A calculation procedure for turbulent flow in complex geometries', *Comput. Fluids*, **15**, 251–273 (1987).
16. A. Segal, P. Wesseling, J. Van Kan, C.W. Oosterlee and K. Kassels, 'Invariant discretization of the incompressible Navier–Stokes equations in boundary fitted co-ordinates', *Int. J. Numer. Methods Fluids*, **15**, 411–426 (1992).
17. R. Aris, *Vectors, Tensors and the Basic Equations of Fluid Mechanics*, Prentice-Hall, Englewood Cliffs, NJ, 1962.
18. H.Q. Yang, S.D. Habchi and A.J. Przekwas, 'General strong conservation formulation of Navier–Stokes equations in non-orthogonal curvilinear co-ordinates', *AIAA J.*, **32**, 936–941 (1994).

19. M. Rosenfeld, D. Kwak and M. Vinokur, 'A solution method for the unsteady and incompressible Navier–Stokes equations in generalized co-ordinates systems', *AIAA Paper 88–0718*, Reno, CA, January 1988.
20. Y. Huang, U. Ghia, G.A. Osswald and K.N. Ghia, 'Velocity–vorticity simulation of unsteady 3D viscous flow within a driven cavity', in M. Deville, T.-H. Lê, Y. Morchoisne (eds.), *Numerical Simulation of 3D Incompressible Unsteady Viscous Laminar Flows: A GAMM Workshop*, *Notes on Numerical Fluid Mechanics*, vol. 36, Vieweg, Wiesbaden, pp. 54–66, 1992.
21. B.P. Leonard, 'A stable and accurate convective modelling procedure based on quadratic upstream interpolation', *Comput. Methods in Appl. Mech. Eng.*, **19**, 59 (1991).
22. W. E and J.G. Liu, 'Vorticity boundary condition and related issues for finite difference schemes', *J. Comput. Phys.*, **124**, 368–382 (1996).
23. T.P. Loc, W. Labidi, A. Dulieu, M. Coutanceau, G. Pineau and A. Texier, 'Simulation numérique d'écoulements instationnaires tridimensionnels par résolution des équations de Navier–Stokes sur un systéme multiprocesseur', *Rapport final de synthese de la convention DRET/LIMSI No. 88/047*, 1990.
24. P. Hille, R. Vehrenkamp and E.O. Schulz-DuBois, 'The development and structure of primary and secondary flow in a curved square duct', *J. Fluid Mech.*, **151**, 219–241 (1985).
25. S. Kaushik and S.G. Rubin, 'Incompressible Navier–Stokes solutions with a new primitive variable solver', *Comput. Fluids*, **24**, 24–40 (1995).
26. M. Deville, T.H. Le and Y. Morchoisne (eds.) ,'Numerical simulation of 3D incompressible unsteady viscous flows', *Notes on Numerical Fluid Mechanics*, vol. 36, Vieweg, Wiesbaden, 1992

