INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 8, 1229–1240 (1988)

ON FINITE ELEMENT APPROXIMATIONS OF THE STREAMFUNCTION–VORTICITY AND VELOCITY–VORTICITY EQUATIONS

MAX D. GUNZBURGER

Department of Mathematics, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.

AND

JANET S. PETERSON

Computing and Communications Division, Los Alamos National Laboratory, Los Alamos, NM 87545, U.S.A.

SUMMARY

We consider finite element methods for vorticity formulations of viscous incompressible flows. In two-dimensional settings the familiar streamfunction–vorticity formulation is examined. We focus on its accuracy, especially when using low-order elements, and on its use with a variety of boundary conditions and in multiply connected domains. In three dimensions the velocity–vorticity formulation is shown to be preferable, and a promising algorithm using this formulation is presented. We close by considering the recovery of the pressure field once the streamfunction or velocity fields are known. In particular we describe and analyse an algorithm for recovering the pressure which is based on well known methods for the primitive variable formulation and which requires no boundary conditions on the pressure at solid walls.

KEY WORDS Streamfunction–vorticity approximations Velocity–vorticity approximations Finite element methods Navier–Stokes equations

STREAMFUNCTION–VORTICITY METHODS FOR PLANE FLOWS

The streamfunction–vorticity formulation has been very popular for the approximation of viscous incompressible plane or axially symmetric flows. The main advantages of using the streamfunction–vorticity variables compared to the use of the primitive variables, i.e. the velocity and pressure, are twofold: first, the incompressibility constraint, i.e. the continuity condition, is satisfied by definition, and secondly, there are two, instead of three, unknown fields. On the other hand, some boundary conditions, e.g. no-slip, are easier to enforce in the primitive variable setting; indeed, finding boundary conditions for the vorticity at solid boundaries is a major, and classical, problem connected with streamfunction–vorticity calculations. Furthermore, multiply connected domains, which pose no algorithmic difficulties for primitive variables, do require special treatment in the streamfunction–vorticity setting.

Here we are concerned with finite element methods for the streamfunction–vorticity formulation of viscous incompressible plane flows. Most of our discussion extends to the axially symmetric case as well. Because we are mainly concerned with spatial discretizations, we only consider steady flows. We first present a finite element algorithm, paying particular attention to

0271–2091/88/101229–12$06.00 *Received 11 April 1988*

the incorporation of a variety of boundary conditions into the algorithm. We then discuss an efficient means of treating multiply connected flow regions. Finally we investigate the accuracy of the finite element approximation. Algorithms similar to the one considered here are discussed in References 1–7 from both computational and implementational points of view.

Boundary conditions on the streamfunction and vorticity

The streamfunction–vorticity equations for plane steady flow are

$$-\Delta\psi=\omega \quad \text{in} \quad \Omega, \tag{1}$$

$$-\nu\Delta\omega+\left(\frac{\partial\psi}{\partial y}\frac{\partial\omega}{\partial x}-\frac{\partial\psi}{\partial x}\frac{\partial\omega}{\partial y}\right)=0 \quad \text{in} \quad \Omega, \tag{2}$$

where ψ denotes the streamfunction, ω the vorticity, ν the kinematic viscosity and Ω a bounded, possibly multiply connected, region in R^2 with boundary Γ. We denote by Γ_0 the exterior boundary of Ω and by Γ_i, $i=1,\ldots,m$, the remaining portions. For simply connected domains $m=0$. The exterior boundary may be a true physical boundary, an artificial boundary introduced in order to render the computational domain finite or a combination of both. In order to enable the specification of different boundary conditions on the outer boundary Γ_0, we subdivide it into three possibly disjoint segments: Γ_{0j}, $j=1,\ldots,3$.

Along the boundaries Γ_i, $i=1,\ldots,m$, and Γ_{01} the velocity is specified so that

$$\psi=q_i+a_i \quad \text{and} \quad \partial\psi/\partial n=-\mathbf{g}_i\cdot\boldsymbol{\tau} \quad \text{on} \quad \Gamma_i,\, i=1,\ldots,m, \tag{3}$$

$$\psi=q_0 \quad \text{and} \quad \partial\psi/\partial n=-\mathbf{g}_0\cdot\boldsymbol{\tau} \quad \text{on} \quad \Gamma_{01}, \tag{4}$$

where $\mathbf{g}_i(\mathbf{x})$, $i=0,\ldots,m$, is the prescribed velocity on the corresponding segment of the boundary, $q_i(\mathbf{x})$, $i=0,\ldots,m$, denote functions such that $\partial q_i/\partial\tau=\mathbf{g}_i\cdot\mathbf{n}$, and $\boldsymbol{\tau}$ and $\mathbf{n}$ denote the unit counterclockwise tangent and outer normal vector to the boundary respectively. Usually the functions $q_i(\mathbf{x})$ may be easily determined from $\mathbf{g}_i$ via simple integrations and in the general case may be determined to arbitrary accuracy through the use of numerical quadratures. The boundary conditions (3) or (4) apply, e.g. at walls, in which case the prescribed velocity vanishes. In (3) the constants a_i are to be determined as part of the solution and reflect the fact that the streamfunction may be specified at only one point on one boundary segment. In this work we have chosen to fix the streamfunction on $\Gamma_{01}\cup\Gamma_{02}$. Of course these boundary segments may be empty, in which case we may specify that $a_1=0$. Thus we need to consider two different cases:

Case I $\Gamma_{01}\cup\Gamma_{02}$ is not empty;

Case II $\Gamma_{01}\cup\Gamma_{02}$ is empty.

For Case I we set $m_1=1$ and for Case II we set $m_1=2$.

Additional conditions are required in order to fix the constants a_i. These can be deduced from the requirement that the pressure is a single-valued function and are easily derived from the momentum equation; indeed, one finds that

$$\int_{\Gamma_i}\left(\nu\frac{\partial\omega}{\partial n}-\omega\mathbf{g}\cdot\mathbf{n}\right)\mathrm{d}\tau=0, \quad i=m_1,\ldots,m. \tag{5}$$

On Γ_{02} we specify the vorticity and the normal component of the velocity; such boundary conditions are useful at, e.g. inflow boundaries. In this case we have

$$\psi=g_0 \quad \text{and} \quad \omega=\omega_0 \quad \text{on} \quad \Gamma_{02}, \tag{6}$$

where $\omega_0(\mathbf{x})$ is the prescribed vorticity and q_0 is defined as above. Finally, on the remaining boundary segment Γ_{03} the tangential component of the velocity and the normal derivative of the vorticity are specified; these boundary conditions are of use, e.g. on outflow or wake portions of a computational boundary. For simplicity we only consider homogeneous versions of these boundary conditions; we then have

$$\frac{\partial\psi}{\partial n}=0 \quad \text{and} \quad \frac{\partial\omega}{\partial n}=0 \quad \text{on} \quad \Gamma_{03}. \tag{7}$$

This completes the specification of our problem. Thus we seek approximations of the functions ψ and ω and the constants a_i, $i=m_1,\ldots,m$, which solve (1)–(7).

The finite element algorithm

We will define a finite element method in the context of quasi-regular triangulations of Ω and in the case where Ω is a polygon. Other element shapes, e.g. quadrilaterals, and curved boundaries can be treated in the standard manner. Throughout, h will denote the maximum of the diameters of the triangles. Also, P_l^h denotes the finite element space of functions which are continuous over Ω and which are polynomials of degree l in each triangle.

The streamfunction and vorticity trial sets, $\hat{S}^h$ and V^h respectively, are defined, for some integers k and l greater than zero, by

$$\begin{aligned}\hat{S}^h=(\psi\in P_l^h|\psi=q_0^h \quad &\text{on} \quad \Gamma_{01} \quad \text{and} \quad \Gamma_{02}, \text{ and}\\ \psi=q_i^h+a_i^h \quad &\text{on} \quad \Gamma_i, i=1,\ldots,m, a_i^h \text{ arbitrary constants}\\ &\text{except that } a_1^h=0 \text{ for Case II})\end{aligned} \tag{8}$$

and

$$V^h=(\omega\in P_k^h|\omega=\omega_0^h \quad \text{on} \quad \Gamma_{02}), \tag{9}$$

where q_i^h, $i=0,\ldots,m$, and ω_0^h are approximations to q_i, $i=0,\ldots,m$, and ω_0 respectively. For example, the former may be taken to be the boundary interpolants, with respect to appropriate boundary segments, of the latter. Similarly, the streamfunction and vorticity test spaces, $\mathscr{S}^h$ and $\mathscr{V}^h$ respectively, are defined, for two more integers r and j greater than zero, by

$$\begin{aligned}\mathscr{S}^h=(\psi\in P_r^h|\psi=0 \quad &\text{on} \quad \Gamma_{01} \quad \text{and} \quad \Gamma_{02}, \quad \text{and}\\ \psi=c_i^h \quad &\text{on} \quad \Gamma_i, i=1,\ldots,m, c_i^h \text{ arbitrary constants}\\ &\text{except that } c_1^h=0 \text{ for Case II})\end{aligned} \tag{10}$$

and

$$\mathscr{V}^h=(\omega\in P_j^h|\omega=0 \quad \text{on} \quad \Gamma_{02}). \tag{11}$$

The finite element algorithm we consider is defined as follows. We seek $\psi^h\in\hat{S}^h$, $\omega^h\in V^h$ and m real numbers $a_{m_1}^h,\ldots,a_m^h$ such that

$$\int_\Omega \omega^h\zeta^h \mathrm{d}\Omega-\int_\Omega \operatorname{grad}\psi^h\cdot\operatorname{grad}\zeta^h\mathrm{d}\Omega=\int_{\Gamma_{01}}\zeta^h\mathbf{g}_0\cdot\boldsymbol{\tau}\mathrm{d}\tau+\sum_{i=1}^m\int_{\Gamma_i}\zeta^h\mathbf{g}_i\cdot\boldsymbol{\tau}\mathrm{d}\tau \quad \text{for all } \zeta^h\in\mathscr{V}^h \tag{12}$$

and

$$\nu\int_\Omega \operatorname{grad}\omega^h\cdot\operatorname{grad}\phi^h\mathrm{d}\Omega+\int_\Omega\phi^h\left(\frac{\partial\psi^h}{\partial y}\frac{\partial\omega^h}{\partial x}-\frac{\partial\psi^h}{\partial x}\frac{\partial\omega^h}{\partial y}\right)\mathrm{d}\Omega=0 \quad \text{for all } \phi^h\in\mathscr{S}^h. \tag{13}$$

All the boundary conditions in (3), (4) and (7) involving normal derivatives are *natural* to the discrete problem (12), (13). Moreover, the conditions in (5) are natural as well. None of these conditions need be required of the test or trial functions.

By choosing bases for the various test and trial spaces, one may transform (12), (13) into a non-linear system of algebraic equations for the discrete streamfunction and vorticity and the unknown constants $a^h_{m_1}, \ldots, a^h_m$. Such a direct approach is described in References 6–8. However, in defining the bases for S^h and $\mathscr{S}^h$, particular basis functions must be defined which couple all the unknowns and equations corresponding to each of the boundary segments Γ_i, $i=m_1, \ldots, m$. Such *non-local* basis functions make for more complicated coding and more costly computations.

An algorithm for multiply connected domains

Non-local basis functions may be avoided if one solves (12), (13) with assumed values for the unknown constants $a^h_{m_1}, \ldots, a^h_m$ and with $c_i=0$, $i=m_1, \ldots, m$, in (10). Of course, arbitrary guesses for these constants will not result in a discrete solution satisfying (5), even in an approximate sense. One could subsequently make a new guess for these constants which hopefully will yield a discrete vorticity which better satisfies (5). There are many ways in which such an iteration can be coupled with a second iteration for solving the non-linear equations (12), (13). Here we describe a particularly simple and efficient method which takes advantage of the fact that any non-linear solution scheme involves the solution of a sequence of *linear* problems. Our discussion is placed in the context of Newton's method; however, any other non-linear method, e.g. quasi-Newton methods, would do just as well insofar as it concerns this technique for dealing with multiply connected domains.

To begin with, we need to redefine the streamfunction test and trial sets. For simplicity of exposition we describe the algorithm for Case I; the modifications necessary for Case II are obvious. Instead of (8) and (10), we now select a sequence of m-tuples $\alpha^{(n,s)}=\{\alpha_1^{(n,s)}, \ldots, \alpha_m^{(n,s)}\}$, $s=0, \ldots, m$ and $n=1, 2, \ldots$, and define the finite element sets

$$S^h_{n,s}=(\psi\in P^h_l|\psi=q^h_0 \quad \text{on} \quad \Gamma_{01} \quad \text{and} \quad \Gamma_{02}, \text{and } \psi=q^h_i+\alpha_i^{(n,s)} \quad \text{on} \quad \Gamma_i, i=1, \ldots, m) \tag{14}$$

for $s=0, \ldots, m$ and $n=1, 2, \ldots$, and

$$\mathscr{S}^h=(\psi\in P^h_r|\psi=0 \quad \text{on} \quad \Gamma_{01}, \Gamma_{02} \quad \text{and} \quad \Gamma_i, i=1, \ldots, m). \tag{15}$$

Here n is the counter for the Newton iteration and the choice of $\alpha^{(n,s)}$ is essentially arbitrary (See the next subsection). Then, for $n=1, 2, \ldots$, given the discrete functions $\psi^{(n-1)}$ and $\omega^{(n-1)}$ and the constants $\alpha^{(n,s)}$, $s=0, \ldots, m$, we define $\psi^{(n,s)}\in S^h_{n,s}$ and $\omega^{(n,s)}\in V^h$ to be the solutions of

$$\int_\Omega \omega^{(n,s)}\zeta \,\mathrm{d}\Omega-\int_\Omega \operatorname{grad}\psi^{(n,s)}\cdot\operatorname{grad}\zeta \,\mathrm{d}\Omega=\int_{\Gamma_{01}}\zeta\mathbf{g}_0\cdot\boldsymbol{\tau}\,\mathrm{d}\tau+\sum_{i=1}^m\int_{\Gamma_i}\zeta\mathbf{g}_i\cdot\boldsymbol{\tau}\,\mathrm{d}\tau \quad \text{for all } \zeta\in\mathscr{V}^h \tag{16}$$

and

$$\nu\int_\Omega \operatorname{grad}\omega^{(n,s)}\cdot\operatorname{grad}\phi \,\mathrm{d}\Omega+C(\psi^{(n-1)},\omega^{(n,s)},\phi)+C(\psi^{(n,s)},\omega^{(n-1)},\phi)=C(\psi^{(n-1)},\omega^{(n-1)},\phi) \quad \text{for all } \phi\in\mathscr{S}^h \tag{17}$$

for $s=0, \ldots, m$, where

$$C(\psi,\omega,\phi)=\int_\Omega\phi\left(\frac{\partial\psi}{\partial y}\frac{\partial\omega}{\partial x}-\frac{\partial\psi}{\partial x}\frac{\partial\omega}{\partial y}\right)\mathrm{d}\Omega.$$

In general none of $\omega^{(n,s)}$, $s=0,\ldots,m$, will satisfy even an approximation of (5). However, since (5) is linear in ω, there exist numbers $\beta_0^{(n)},\ldots,\beta_m^{(n)}$ such that

$$\beta_0^{(n)}+\ldots+\beta_m^{(n)}=1 \tag{18}$$

and, if

$$\omega^{(n)}=\sum_{s=0}^{m}\beta_s^{(n)}\omega^{(n,s)}, \tag{19}$$

$$\int_{\Gamma_i}\left(\nu\frac{\partial\omega^{(n)}}{\partial n}-\omega^{(n)}\mathbf{g}\cdot\mathbf{n}\right)\mathrm{d}\tau=0,\quad i=1,\ldots,m. \tag{20}$$

Indeed, upon substituting (19) into (20), one has that (18) and (20) constitute $m+1$ linear algebraic equations for the $m+1$ unknowns $\beta_0^{(n)},\ldots,\beta_m^{(n)}$. Moreover, because of (18) and the fact that each $\omega^{(n,s)}$ satisfies the boundary conditions for the discrete vorticity, $\omega^{(n)}$ defined by (19) satisfies these boundary conditions as well. Now let

$$\psi^{(n)}=\sum_{s=0}^{m}\beta_s^{(n)}\psi^{(n,s)}. \tag{21}$$

Then $\psi^{(n)}$ satisfies all the boundary conditions on the discrete streamfunction and, because (16) and (17) are *linear* equations, the pair $(\psi^{(n)},\omega^{(n)})$ satisfies the Newton equations

$$\int_\Omega\omega^{(n)}\zeta\mathrm{d}\Omega-\int_\Omega\mathrm{grad}\psi^{(n)}\cdot\mathrm{grad}\zeta\mathrm{d}\Omega=\int_{\Gamma_{01}}\zeta\mathbf{g}_0\cdot\boldsymbol{\tau}\mathrm{d}\tau+\sum_{i=1}^{m}\int_{\Gamma_i}\zeta\mathbf{g}_i\cdot\boldsymbol{\tau}\mathrm{d}\tau\quad\text{for all }\zeta\in\mathscr{V}^h \tag{22}$$

and

$$\nu\int_\Omega\mathrm{grad}\omega^{(n)}\cdot\mathrm{grad}\phi\mathrm{d}\Omega+C(\psi^{(n-1)},\omega^{(n)},\phi)+C(\psi^{(n)},\omega^{(n-1)},\phi)$$
$$=C(\psi^{(n-1)},\omega^{(n-1)},\phi)\quad\text{for all }\phi\in\mathscr{S}^h. \tag{23}$$

The unknowns $a_i^{(n)}$, $i=1,\ldots,m$, are approximated at the nth Newton step by

$$a_i^{(n)}=\sum_{s=0}^{m}\beta_s^{(n)}\alpha_i^{(n,s)},\quad i=1,\ldots,m. \tag{24}$$

Thus (16)–(21) indicate how the Newton iterates $[\psi^{(n)},\omega^{(n)},(a_i^{(n)},i=1,\ldots,m)]$ are updated.

Remarks

1. Each Newton step requires the solution of the $m+1$ linear systems (16), (17). However, the left-hand sides of these systems are all identical so the major portion of the computation, i.e. the Gauss elimination steps, need be carried out only once per Newton iteration.

2. This algorithm takes explicit advantage of the linearity of the boundary conditions and of the defining systems for the Newton iterates. Also, each Newton iterate $[\psi^{(n)},\omega^{(n)},(a_i^{(n)},i=1,\ldots,m)]$, $n>0$, is required to satisfy the auxiliary condition (5) or (20). Of course the individual Newton iterates do not satisfy the discrete system (12), (13); indeed, the Newton iteration is carried out until (12), (13) are satisfied to within some predetermined tolerance.

3. The algorithm may be started with arbitrary values of $\psi^{(0)}$ and $\omega^{(0)}$. However, it is usually a good idea to use the solution at lower values of the Reynolds number to help determine these initial guesses. Also, it is often desirable to perform a couple of steps of the simple iteration scheme

$$\int_{\Omega}\omega^{(n)}\zeta\mathrm{d}\Omega-\int_{\Omega}\mathrm{grad}\psi^{(n)}\cdot\mathrm{grad}\zeta\mathrm{d}\Omega=\int_{\Gamma_{01}}\zeta\mathbf{g}_0\cdot\boldsymbol{\tau}\mathrm{d}\tau+\sum_{i=1}^{m}\int_{\Gamma_i}\zeta\mathbf{g}_i\cdot\boldsymbol{\tau}\mathrm{d}\tau\quad\text{for all }\zeta\in\mathscr{V}^h \tag{25}$$

and

$$\nu\int_{\Omega}\mathrm{grad}\omega^{(n)}\cdot\mathrm{grad}\phi\mathrm{d}\Omega+C(\psi^{(n-1)},\omega^{(n)},\phi)=0\quad\text{for all }\phi\in\mathscr{S}^h \tag{26}$$

before switching to the Newton method. The reason for this is that the scheme (25), (26), although only linearly convergent, converges for a larger range of initial data than does the Newton scheme (22), (23). Of course, for multiply connected domains, the analogue of (16)–(21) may be defined for the scheme (25), (26). Also, through the use of a few flag variables, the Newton scheme and the simple iteration schemes may be easily implemented within the same code.

4. The guesses $\{\alpha_1^{(n,s)},\ldots,\alpha_m^{(n,s)}\}$, $s=0,\ldots,m$, needed for each n in order to define (16), (17), may be arbitrarily chosen except for the mild requirement that the $(m+1)\times(m+1)$ matrix whose sth row is given by $\{1,\alpha_1^{(n,s)},\ldots,\alpha_m^{(n,s)}\}$ be non-singular. For example, one may choose $\alpha_i^{(n,s)}=\delta_{is}$ for $i,s=1,\ldots,m$ and $\alpha_i^{(n,0)}=1$ for $i=1,\ldots,m$.

5. Since we are solving (12), (13) as a coupled system, no artificial boundary conditions on the vorticity need be specified at boundaries where the velocity is specified, e.g. solid walls.

6. With regard to the degree of the polynomials used in defining the various sets of trial and test functions, there are two choices which suggest themselves. First, we have $l=k=r=j\geqslant 1$, so that all test and trial sets employ the same degree polynomials. This choice has been used successfully by various researchers and has been subject to mathematical analyses. Secondly, for various practical as well as mathematical reasons, the choice $l=k+1\geqslant 2$ suggests itself. Here the streamfunction trial functions are polynomials of one degree higher than the vorticity trial functions. One cannot choose $r=l$ and $j=k$ since in this case, even for the linear Stokes problem, the discrete system is singular. However, in Reference 4 the choice $r=k=1$ and $j=l=2$ was used successfully in computations. Note that in this case the degrees of the polynomials in the test and trial sets are reversed.

7. We now briefly summarize what is known about the accuracy of finite element approximations of (1), (2). Rigourous results have been obtained only for the case of velocity boundary conditions, i.e. ψ and $\partial\psi/\partial n$ specified on all boundaries. For the case of all test and trial sets consisting of the same degree piecewise polynomials, we have[9] that, for sufficiently smooth exact solutions ψ and ω of (1), (2),

$$\left[\int_{\Omega}\left\{\left(\frac{\partial}{\partial x}(\psi-\psi^h)\right)^2+\left(\frac{\partial}{\partial y}(\psi-\psi^h)\right)^2\right\}\mathrm{d}\Omega\right]^{1/2}+\left[\int_{\Omega}(\omega-\omega^h)^2\mathrm{d}\Omega\right]^{1/2}\leqslant Ch^{l-1/2}|\ln h|^{\sigma}, \tag{27}$$

where $l=k=r=j\geqslant 1$ is the degree of the polynomials used, $\sigma=1$ for $l=1$ and $\sigma=0$ for $l>1$. Essentially, this estimate is not optimal with regard to the power of h for either the derivatives of the streamfunction, i.e. the velocity components, or for the vorticity.

8. There is both computational and theoretical evidence that the estimate (27) is not sharp. In References 6 and 7 some computational experiments are given which indicate that the derivatives of the streamfunction are optimally approximated by the finite element approximation in the sense that

$$\left[\int_{\Omega}\left\{\left(\frac{\partial}{\partial x}(\psi-\psi^h)\right)^2+\left(\frac{\partial}{\partial y}(\psi-\psi^h)\right)^2\right\}\mathrm{d}\Omega\right]^{1/2}=O(h^l).$$

This improved estimate is confirmed, in the case of the linear Stokes problem, in Reference 10.

9. We now present the results of some simple computational experiments illustrating the above algorithm in the case of doubly connected domains. In all cases the exterior boundary Γ_0 is the unit square. The single interior boundary Γ_1 is that of a rectangle located within the unit square. On Γ_1 we specify $q_1=0$ so that $\psi=a_1$ on Γ_1 for some unknown constant a_1. The three different problems considered are characterized by the boundary values of ψ on the left and right edges of Γ_0 and the position of the inner rectangle. We denote by Ω_1 the region bounded by Γ_1. We then consider the three problems

$$\begin{aligned}
&\text{A:}\quad \psi(0,y)=\psi(1,y)=2y^3-3y^2, && \Omega_1=(1/4,1/2)\times(1/2,3/4),\\
&\text{B:}\quad \psi(0,y)=\psi(1,y)=2y^3-3y^2, && \Omega_1=(1/4,1/2)\times(1/4,1/2),\\
&\text{C:}\quad \psi(0,y)=\psi(1,y)=y^2, && \Omega_1=(2/5,3/5)\times(2/5,3/5).
\end{aligned}$$

The boundary conditions on ψ at $y=0$ and $y=1$ are compatible constant values and $\partial\psi/\partial n=0$ on all boundaries. Note that owing to the symmetry of problem A in that case $a_1=-1/2$. All computations were carried out using a uniform mesh size. Results for the approximation of the value of the constant a_1 are summarized in Table I for different values of the grid size h and Reynolds number Re.

VELOCITY–VORTICITY FORMULATIONS IN THREE DIMENSIONS

The streamfunction–vorticity formulation may be extended to three-dimensional problems, where now both the streamfunction $\boldsymbol{\psi}$ satisfying $\mathbf{u}=\operatorname{curl}\boldsymbol{\psi}$ and the vorticity $\boldsymbol{\omega}=\operatorname{curl}\mathbf{u}$ are vector-valued functions. Thus in this setting the primitive variable formulation has less unknown fields, i.e. the velocity $\mathbf{u}$ and the pressure p, than does the streamfunction–vorticity formulation. Furthermore, unlike two-dimensional settings in which the streamfunction is uniquely determined except for an additive constant, in three-dimensional settings a gradient of any scalar function may be added to $\boldsymbol{\psi}$ without affecting the velocity field. This non-uniqueness may be removed in many ways; perhaps the most popular is to require that $\operatorname{div}\boldsymbol{\psi}=0$ throughout Ω. Note that the components of $\boldsymbol{\omega}$ are related, by definition, through the relation $\operatorname{div}\boldsymbol{\omega}=0$ as well.

Another problem is the transformation of velocity boundary conditions into boundary conditions for $\boldsymbol{\psi}$. This is a relatively easy task for two-dimensional problems. In three dimensions we have that if, e.g., $\mathbf{u}=\mathbf{0}$ on a portion of the boundary, then $\operatorname{curl}\boldsymbol{\psi}=\mathbf{0}$ there. The latter is not, in general, a boundary condition which discretizes easily.

Table I. Computational results for a_1 for doubly connected problems using piecewise linear functions

	h	$Re=0$	$Re=1$	$Re=10$
A	1/4	−0·417	−0·418	−0·429
	1/8	−0·466	−0·468	−0·487
	1/16	−0·482	−0·484	−0·501
B	1/4	−0·214	−0·216	−0·235
	1/8	−0·235	−0·238	−0·268
	1/16	−0·250	−0·253	−0·281
C	1/5	0·372	0·371	0·348
	1/10	0·377	0·374	0·347
	1/15	0·378	0·375	0·347

An alternative formulation uses the velocity and vorticity as variables and is given by

$$\operatorname{div}\mathbf{u}=0 \quad \text{in} \quad \Omega, \tag{28}$$

$$\operatorname{curl}\mathbf{u}=\boldsymbol{\omega} \quad \text{in} \quad \Omega, \tag{29}$$

$$\nu\Delta\omega=\mathbf{u}\cdot\operatorname{grad}\boldsymbol{\omega}-\boldsymbol{\omega}\cdot\operatorname{grad}\mathbf{u} \quad \text{in} \quad \Omega. \tag{30}$$

Of course velocity boundary conditions are easy to apply. On the other hand, unlike the streamfunction–vorticity formulation, one has to deal with the incompressibility constraint (28). However, it is not as difficult to deal with this constraint here as it is in the primitive variable setting.

If $\boldsymbol{\omega}$ is known, (28), (29) form a first-order system for $\mathbf{u}$; on the other hand, if $\mathbf{u}$ is known, (30) is a second-order system for $\boldsymbol{\omega}$. Unfortunately, on boundary segments where the velocity is specified, (28), (29) have too many boundary data and (30) not enough. For example, (29), (30) is uniquely solvable if only $\mathbf{u}\cdot\mathbf{n}$ is specified on the boundary. However, as a whole, (28)–(30) are well posed with just $\mathbf{u}$ specified at the boundary, e.g. at solid walls one need not specify $\boldsymbol{\omega}$.

On the other hand, one would like to avoid solving for the six scalar fields of $\mathbf{u}$ and $\boldsymbol{\omega}$ simultaneously, so that one would like to solve for $\mathbf{u}$ from (28), (29) using some approximation for $\boldsymbol{\omega}$, and then solve for $\boldsymbol{\omega}$ from (30) using some approximation for $\mathbf{u}$. To accomplish the first task, one must decide which, if any, of the boundary conditions for $\mathbf{u}$ are not to be enforced. To successfully solve (30) for $\boldsymbol{\omega}$, one must 'make up' a boundary condition for the vorticity on boundary segments where the velocity is specified.

Boundary conditions for the velocity–vorticity formulation

Some authors have chosen to only impose the normal component of the velocity in conjunction with the system (28), (29). The tangential components of the velocity are imposed in an indirect manner through an artificial boundary condition for the vorticity. Although, given $\boldsymbol{\omega}$, simply imposing the normal velocity is sufficient to solve this system, it seems wasteful not to use the known tangential components of the velocity field at the boundary in as direct a manner as possible. Therefore we suggest that the velocity field be made to satisfy the given boundary conditions directly. Thus, if for simplicity we are given that $\mathbf{u}=\mathbf{g}$ on the boundary Γ, we will require that all components of the discrete velocity field satisfy this condition or an approximation to this condition. Since, in some sense, we now have 'too many' boundary conditions for (28), (29), we solve these in a least-squares manner.

The above observations suggest that we solve the problem (28)–(30) with, for simplicity, $\mathbf{u}=\mathbf{g}$ on Γ through the following iterative process. Starting with some arbitrary guess $\mathbf{u}^{(0)}$, let $\boldsymbol{\omega}^{(0)}=\operatorname{curl}\mathbf{u}^{(0)}$; then solve, for $n=1, 2, \ldots,$

$$\operatorname{div}\mathbf{u}^{(n)}=0 \quad \text{in} \quad \Omega, \tag{31}$$

$$\operatorname{curl}\mathbf{u}^{(n)}=\boldsymbol{\omega}^{(n-1)} \quad \text{in} \quad \Omega, \tag{32}$$

$$\mathbf{u}^{(n)}=\mathbf{g} \quad \text{on} \quad \Gamma \tag{33}$$

for $\mathbf{u}^{(n)}$ and

$$\nu\Delta\omega^{(n)}=\mathbf{u}^{(n)}\cdot\operatorname{grad}\boldsymbol{\omega}^{(n)}-\boldsymbol{\omega}^{(n)}\cdot\operatorname{grad}\mathbf{u}^{(n)} \quad \text{in} \quad \Omega \tag{34}$$

for $\boldsymbol{\omega}^{(n)}$. We still need to define boundary conditions for $\boldsymbol{\omega}^{(n)}$ so that (34) may be uniquely solved. Here there are a least two choices which suggest themselves. One may simply require that $\boldsymbol{\omega}^{(n)}=\operatorname{curl}\mathbf{u}^{(n)}$ on the boundary Γ. Another choice is

$$\operatorname{div}\boldsymbol{\omega}^{(n)}=0 \quad \text{on} \quad \Gamma, \tag{35}$$

$$\boldsymbol{\omega}^{(n)}\times\mathbf{n}=\operatorname{curl}\mathbf{u}^{(n)}\times\mathbf{n} \quad \text{on} \quad \Gamma. \tag{36}$$

It is well known that because of (31) the data $\mathbf{g}$ must have zero mean over Γ; we assume that this is the case. However, for (31)–(33) to have a solution, we must also have that $\operatorname{div}\boldsymbol{\omega}^{(n-1)}=0$ on Ω. This motivates the choice of boundary condition (35). Indeed, by taking the divergence of (34), we find that $\Delta(\operatorname{div}\boldsymbol{\omega}^{(n)})-\mathbf{u}^{(n)}\cdot\operatorname{grad}(\operatorname{div}\boldsymbol{\omega}^{(n)})=0$ in Ω, so that together with (35) this implies the desired result $\operatorname{div}\boldsymbol{\omega}^{(n)}=0$ in Ω. Thus all the vorticity iterates are divergence-free. The boundary condition (36) simply fixes the tangential components of the vorticity.

Finite element discretizations

We now turn to finite element discretizations of the two problems (31)–(33) and (34)–(36). We now use $\mathbf{u}^{(n)}$ and $\boldsymbol{\omega}^{(n)}$ to denote the *discrete* velocity and vorticity respectively, and $\mathbf{g}$ now denotes some approximation to the given data along the boundary. Sets of finite element spaces for the velocity and vorticity are defined in the usual manner; we denote these by $\mathscr{V}$ and $\mathscr{W}$ respectively. We define the sets $\mathscr{V}_g=(\mathbf{v}\in\mathscr{V}\mid\mathbf{v}=\mathbf{g}\text{ on }\Gamma)$ and $\mathscr{W}_g^{(n)}=(\boldsymbol{\zeta}\in\mathscr{W}\mid\boldsymbol{\zeta}\times\mathbf{n}=\operatorname{curl}\mathbf{u}^{(n)}\times\mathbf{n}\text{ on }\Gamma)$ and the spaces $\mathscr{V}_0=(\mathbf{v}\in\mathscr{V}\mid\mathbf{v}=\mathbf{0}\text{ on }\Gamma)$ and $\mathscr{W}_0=(\boldsymbol{\zeta}\in\mathscr{W}\mid\boldsymbol{\zeta}\times\mathbf{n}=\mathbf{0}\text{ on }\Gamma)$. Then, starting with the initial guess $\boldsymbol{\omega}^{(0)}$ as above, for $n=1, 2, \ldots$, we compute $\mathbf{u}^{(n)}\in\mathscr{V}_g$ such that

$$\int_\Omega(\operatorname{curl}\mathbf{u}^{(n)}\cdot\operatorname{curl}\mathbf{v}+\operatorname{div}\mathbf{u}^{(n)}\operatorname{div}\mathbf{v})\mathrm{d}\Omega=\int_\Omega\boldsymbol{\omega}^{(n-1)}\cdot\operatorname{curl}\mathbf{v}\mathrm{d}\Omega \quad \text{for all } \mathbf{v}\in\mathscr{V}_0 \tag{37}$$

and then $\boldsymbol{\omega}^{(n)}\in\mathscr{W}_g^{(n)}$ such that

$$\int_\Omega(\operatorname{curl}\boldsymbol{\omega}^{(n)}\cdot\operatorname{curl}\boldsymbol{\zeta}+\operatorname{div}\boldsymbol{\omega}^{(n)}\operatorname{div}\boldsymbol{\zeta})\mathrm{d}\Omega+\int_\Omega(\mathbf{u}^{(n)}\cdot\operatorname{grad}\boldsymbol{\omega}^{(n)}-\boldsymbol{\omega}^{(n)}\cdot\operatorname{grad}\mathbf{u}^{(n)})\cdot\boldsymbol{\zeta}\mathrm{d}\Omega=0 \quad \text{for all } \boldsymbol{\zeta}\in\mathscr{W}_0. \tag{38}$$

The discrete problems (37) and (38) are respectively based on a least-squares variational formulation of (31)–(33) and a Galerkin variational formulation of (34)–(36). Individually, these discretization methods are known to be well suited for the corresponding problems. Note that the boundary condition (35) is natural to the weak formulation (38).

The weak form (37) is identical to that which would be obtained from a standard Galerkin discretization of the system

$$-\Delta\mathbf{u}^{(n)}=\operatorname{curl}\boldsymbol{\omega}^{(n-1)} \quad \text{in} \quad \Omega, \qquad \mathbf{u}^{(n)}=\mathbf{g} \quad \text{on} \quad \Gamma,$$

which is easily obtainable from (31)–(33) by differentiating. This last system has exactly the right boundary conditions for $\mathbf{u}^{(n)}$ and thus it is not surprising that (31)–(33) are solvable by a least-squares technique, even though at first glance there seem to be too many boundary conditions.

We have not carried out extensive calculations based on the above algorithm, nor have we attempted to analyse its convergence properties; we hope to do both in the near future.

RECOVERY OF THE PRESSURE FIELD

Once the velocity field is known, either directly or by differentiating the streamfunction, the pressure is commonly recovered by the following process. First one takes the divergence of the momentum equation to obtain

$$-\Delta p=\operatorname{div}(\mathbf{u}\cdot\operatorname{grad}\mathbf{u}-\nu\Delta u), \tag{39}$$

where p denotes the pressure and the constant density has been absorbed into p. Corresponding to (39), one then sets up a weak formulation

$$\int_{\Omega} (\operatorname{grad} p \cdot \operatorname{grad} q) \mathrm{d}\Omega = G(q; \mathbf{u}), \tag{40}$$

where $G(\cdot; \mathbf{u})$ is a linear functional. Various forms for this functional have been suggested by different authors; see Reference 11 for a complete discussion. In (40) neither the trial functions p nor the test functions q are required to satisfy any boundary conditions. Of course (40) may be discretized by standard finite element techniques. Comparing (39) and (40) shows that the latter implies that p satisfies a natural boundary condition whose particular form depends on the specific choice made for $G(\cdot; \mathbf{u})$. After much confusion and misunderstanding concerning which natural boundary condition is physically correct, this question has recently been settled in Reference 11 where the correct choice for $G(\cdot; \mathbf{u})$ is discussed.

However, the use of (39) or (40) is still problematical when used in conjunction with a streamfunction–vorticity or velocity–vorticity calculation. The velocity field is only approximately known, and thus the differentiation process used to derive (39) introduces unnecessary additional errors. (Indeed, if the velocity is derived from a discrete streamfunction, the right-hand side of (39) involves the *second* derivatives of that streamfunction.) Related to this is the fact that, in a strict mathematical sense, the right-hand side of (40) may not be well defined for discrete velocities belonging to the usual finite element spaces.

We briefly describe an alternate method of recovering the pressure which does not encounter any of the above difficulties; in particular, absolutely no boundary conditions on the pressure are needed at solid walls. Furthermore, only the discrete velocity and vorticity are needed and not derivatives of the velocity. This method is based on one given in Reference 12 for the fourth-order streamfunction formulation.

We begin by noting that the momentum equation may be written in the form

$$\operatorname{grad} H = -\nu \operatorname{curl} \boldsymbol{\omega} + \mathbf{u} \times \boldsymbol{\omega}, \tag{41}$$

where $H = p + (\mathbf{u} \cdot \mathbf{u})/2$. (Again, the constant density is absorbed into p.) Then, for any sufficiently smooth vector-valued field $\mathbf{v}$ vanishing on Γ, we have

$$\int_{\Omega} H \operatorname{div} \mathbf{v} \, \mathrm{d}\Omega = \int_{\Omega} (\boldsymbol{\omega} \cdot \mathbf{v} \times \mathbf{u} - \nu \boldsymbol{\omega} \cdot \operatorname{curl} \mathbf{v}) \mathrm{d}\Omega. \tag{42}$$

At this point we could try to discretize (42); however, this will in general not lead to a practical algorithm. Roughly speaking, the problem is that discrete versions of (41) do not usually have right-hand sides which are in the range of the discrete approximation to the gradient.

This problem is easily remedied by introducing the auxiliary variable $\mathbf{w}$ and then considering the *linear* Stokes problem for $\mathbf{w}$ and H:

$$\Delta \mathbf{w} + \operatorname{grad} H = -\nu \operatorname{curl} \boldsymbol{\omega} + \mathbf{u} \times \boldsymbol{\omega} \quad \text{in } \Omega, \tag{43a}$$

$$\operatorname{div} \mathbf{w} = 0 \quad \text{in } \Omega, \tag{43b}$$

$$\mathbf{w} = \mathbf{0} \quad \text{on } \Gamma. \tag{43c}$$

If the right-hand side of (43a) is in the range of the gradient, as is the case for the exact solutions $\boldsymbol{\omega}$ and $\mathbf{u}$, then it is easily shown that $\mathbf{w} = \mathbf{0}$ and that therefore H satisfies (41).

Thus, given $\boldsymbol{\omega}$ and $\mathbf{u}$, we will use (43) to determine H, after which p is easily determined. Our procedure does not require any boundary conditions on H or p, except for fixing one of these at a single point in the flow. Of course this is perfectly natural since the pressure is determined only up to an additive constant.

A discretization of (43) is effected by using stable finite element spaces for the primitive variable formulation of the Navier–Stokes equations. Thus we seek approximations to H and $\mathbf{w}$ belonging to a pressure finite element space and a velocity finite element space respectively, such that the standard weak form of (43) holds. Specifically, we choose a pressure finite element space $\mathscr{P}^h$ and a velocity space $\mathscr{V}^h$, the latter satisfying homogeneous boundary conditions, and then seek $\mathbf{w}^h \in \mathscr{V}^h$ and $H^h \in \mathscr{P}^h$ such that

$$-\int_\Omega \operatorname{grad} \mathbf{w}^h \cdot \operatorname{grad} \cdot \mathbf{v}^h \, d\Omega + \int_\Omega H^h \operatorname{div} \mathbf{v}^h \, d\Omega = \int_\Omega (\boldsymbol{\omega} \cdot \mathbf{v}^h \times \mathbf{u} - \nu \boldsymbol{\omega} \cdot \operatorname{curl} \mathbf{v}^h) \, d\Omega \quad \text{for all } \mathbf{v}^h \in \mathscr{V}^h,$$

$$\int_\Omega q^h \operatorname{div} \mathbf{w}^h \, d\Omega = 0 \quad \text{for all } q^n \in \mathscr{P}^h. \tag{44}$$

There are many such primitive variable spaces known (e.g. see References 8 and 13) which result in accurate approximations to H. We note that if a stable finite element pair is used for approximating H and $\mathbf{w}$, then $\mathbf{w}^h \to \mathbf{0}$ as $h \to 0$ and, of course, $H^h \to H$.

In general the right-hand side of (43) or (44) is only known discretely, e.g. $\boldsymbol{\omega}$ and $\mathbf{u}$ are found from a streamfunction–vorticity or velocity–vorticity calculation. Thus one cannot expect to be able to find the pressure to arbitrary accuracy. In fact the pressure can be approximated only as accurately as the vorticity used in the right-hand side of (44). For example, if peicewise linear elements are used for the streamfunction and vorticity, the vorticity approximation is at best $O(h)$ accurate and thus one may use piecewise constant pressure spaces and likewise compute an $O(h)$ approximation to the pressure.

ACKNOWLEDGEMENT

The work of M. D. Gunzburger was supported by the United States Air Force Office of Scientific Research under Grant Number AFOSR-83-0101.

REFERENCES

1. A. Campion-Renson and M. Crochet, 'On the streamfunction–vorticity finite element solutions of Navier–Stokes equations', *Int. j. numer. methods eng.*, **12,** 1809–1818 (1978).
2. G. Dhatt, B. Fomo and C. Bourque, 'A ψ–ω finite element formulation for the Navier–Stokes equations', *Int. j. numer. methods eng.*, **17,** 199–212 (1981).
3. K. Barrett, 'A variational principle for the streamfunction–vorticity formulation of the Navier–Stokes equations incorporating no-slip conditions', *J. Comput. Phys.*, **26,** 153–161 (1978).
4. M. Peeters, W. Habashi and E. Dueck, 'Finite element solutions of the Navier–Stokes equations', *Proc. 4th Int. Conf. on Numerical Methods in Laminar and Turbulent Flow*, Pineridge Press, Swansea, 1985, pp. 433–444.
5. W. Habashi, M. Peeters, G. Guevremont and M. Hafez, 'Finite element solutions of the compressible Navier–Stokes equations', *AIAA J.*, **25,** 944–948 (1987).
6. M. Gunzburger, and J. Peterson, 'On finite element approximations of the streamfunction–vorticity equations', *Computer Methods for PDE, Vol. V*, IMACS, 1984, pp. 47–56.
7. M. Gunzburger and J. Peterson, 'Finite element methods for the streamfunction–vorticity equations: boundary condition treatments and multiply connected domains', *SIAM J. Sci. Stat. Comput.*, **9,** 650–668 (1988).
8. V. Girault and P. Raviart, *Finite Element Approximations of the Navier–Stokes Equations*, Springer, Berlin 1979 and 1986.
9. F. Brezzi, J. Rappaz and P. Raviart, 'Finite dimensional approximation of nonlinear problems. Part I: Branches of nonsingular solutions', *Numer. Math.*, **36,** 1–25 (1980).

10. G. Fix, M. Gunzburger, R. Nicolaides and J. Peterson, 'Mixed finite element approximations for the biharmonic equation', *Proc. 5th Int. Symp. on Finite Element Flow Problems*, University of Texas, 1984, pp. 281–286.
11. P. Gresho and R. Sani, 'On pressure boundary conditions for the incompressible Navier–Stokes equations', *Int. j. numer. methods fluids*, **7**, 1111–1145 (1987).
12. M. Cayco and R. Nicolaides, 'Finite element techniques for optimal pressure recovery from the streamfunction formulation of viscous flows', *Math. Comput.*, **46**, 371–377 (1986).
13. M. Gunzburger, 'Mathematical aspects of finite element methods for incompressible viscous flows', *ICASE Report 86-62*, Hampton, 1986.