INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS, VOL. 5, 269–280 (1985)

A REVIEW OF SOME FINITE ELEMENT METHODS TO SOLVE THE STATIONARY NAVIER–STOKES EQUATIONS

A. SEGAL

Department of Mathematics and Inf., Delft University of Technology, Postbus 356, 2600 AJ Delft, The Netherlands

SUMMARY

In this paper the integrated solution approach, the penalty function approach and the solenoidal approach for the finite element solution of the stationary Navier–Stokes equations are compared. It is shown that both the penalty function approach and the solenoidal approach compare favourably to the integrated solution method. For fine meshes the solenoidal approach appears to be the cheapest method.

KEY WORDS Navier–Stokes Equations Finite Element Method Solenoidal Approach Penalty Function Approach

INTRODUCTION

The stationary two-dimensional Navier–Stokes equations are given by the momentum equations

$$-\sigma_{ij,j}+\rho(u_iu_j)_{,j}=\rho f_i, \quad i,j=1,2 \tag{1}$$

and the continuity equation

$$u_{i,i}=0, \quad i=1,2 \tag{2}$$

For a Newtonian fluid the Cauchy stress tensor can be written as

$$\sigma_{ij}=-p\delta_{ij}+\eta(u_{i,j}+u_{j,i}) \tag{3}$$

with η the dynamic viscosity, ρ the density of the fluid, p the pressure and u_i the velocity components.

Boundary conditions for equations (1) and (2) may be of the following types:

(a) u_n and u_t given
(b) u_n and σ_t or u_t and σ_n given
(c) σ_t and σ_n given

with

$$\sigma_t=\sigma_{ij}n_jt_i=\eta\left(\frac{\partial u_t}{\partial n}+\frac{\partial u_n}{\partial t}\right) \tag{4}$$

and

$$\sigma_n=\sigma_{ij}n_jn_i=-p+2\eta\frac{\partial u_n}{\partial n}$$

where **n** is the outward normal and **t** the tangential unit vector, along the boundary.

Along each part of the boundary exactly one of the conditions (a), (b) or (c) must be given.

0271–2091/85/03 0269–12$01.20

Received 4 October 1983
Revised 6 April 1984

Methods to solve (1), (2) with boundary conditions (4) numerically are, for example, the stream-function–vorticity approach, the stream-function approach and the primitive variables approach.[1]

In this paper we shall limit ourselves to the primitive variables approach and consider three methods commonly used to solve these equations by the finite element method.

THE PRIMITIVE VARIABLES APPROACH, INTEGRATED FORMULATION

In the integrated method equations (1), (2) and (4) are solved by applying the Galerkin method in a straightforward way.

The momentum equations are multiplied by test functions δu_i and the continuity equation by a test function δp, and integrated over the domain Ω. In this way we obtain

$$\int_\Omega (-\sigma_{ij,j} + \rho(u_i u_j)_{,j})\delta u_i \, dx = \int_\Omega \rho f_i \delta u_i \, dx \tag{5}$$

$$-\int_\Omega \delta p u_{i,i} \, dx = 0 \tag{6}$$

The Gauss theorem applied to (5) yields

$$\int_\Omega \sigma_{ij}\delta u_{i,j} \, dx + \int_\Omega \rho(u_i u_j)_{,j}\delta u_i \, dx = \int_\Omega \rho f_i \delta u_i \, dx + \int_{\partial\Omega} \sigma_{ij} n_j \delta u_i \, ds$$

We can write

$$\int_{\partial\Omega} \sigma_{ij} n_i \delta u_i \, ds = \int_{\partial\Omega} (\sigma_t \delta u_t + \sigma_n \delta u_n) \, ds \tag{7}$$

For a Newtonian fluid we have

$$\int_\Omega \sigma_{ij}\delta u_{i,j} \, dx = \int_\Omega \eta(u_{i,j} + u_{j,i})\delta u_{i,j} \, dx - \int_\Omega p \delta u_{i,i} \, dx \tag{8}$$

Hence we obtain

$$\int_\Omega \eta(u_{i,j} + u_{j,i})\delta u_{i,j} \, dx - \int_\Omega p\delta u_{i,i} \, dx + \int_\Omega \rho(u_i u_j)_{,j}\delta u_i \, dx$$

$$= \int_\Omega \rho f_i \delta u_i \, dx + \int_{\partial\Omega} \{\sigma_t \delta u_t + \sigma_n \delta u_n\} ds \tag{9}$$

$$-\int_\Omega \delta p u_{i,i} \, dx = 0 \tag{10}$$

This is the so-called weak formulation of (1)–(4)

The Galerkin equations are derived from (9) and (10) by substituting the approximations

$$u_i^h = \sum_{l=1}^{n_i} u_{il}\phi_l(x)$$

$$p_i^h = \sum_{l=1}^{m} p_l \psi_l(x)$$

and replacing δu_i and δp by the basis functions ϕ_k and ψ_k, respectively.

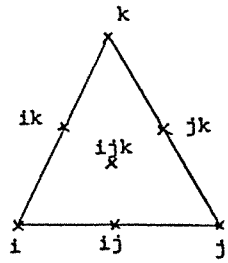

Figure 1. Element e

It is well known[2,3] that in order to prevent singular matrices and the so-called chequer-board modes for the pressure[4,5] the elements must satisfy the so-called Brezzi–Babuska condition.[6]

Among the successful elements in $\mathbb{R}^2$ we mention:

(i) The quadratic 6-node conforming triangle.[7] Velocity components are approximated quadratically; the pressure is approximated by piecewise linear polynomials, continuous over the element boundaries. This element satisfies the B–B condition. For a proof see Reference 8. The number of degrees of freedom in each element is 15.

(ii) The extended 7 point quadratic triangle[9] (see Figure 1). The velocity components are approximated by a quadratic polynomial plus a so-called 'bubble' third degree term, that is zero on the element boundaries. Approximation of the pressure is linear in each element and discontinuous over the element boundaries. In this form the number of degrees of freedom in each element is 17. However, writing the pressure in the elements as

$$p^h = p^h_c + (x_1 - x_{1c})p^h_{,x_1} + (x_2 - x_{2c})p^h_{,x_2} \tag{11}$$

with the subscript c referring to the centroid, the continuity equation (10) can be written elementwise as

$$\int_e u^h_{i,i}\,\mathrm{d}x = 0 \tag{12}$$

$$\int_e (x_j - x_{j,c})u^h_{i,i}\mathrm{d}x = 0, \quad i = 1,2; j = 1,2 \tag{13}$$

With (13) we can eliminate the velocity in the centroid and hence the gradient of the pressure, thus reducing the number of degrees of freedom to 13 in each element.

The quadrilateral counterparts of these elements are the biquadratic 9-node quadrilateral element,[4] with velocity components approximated biquadratically and bilinear pressure continuous over the element boundaries, and the biquadratic 9-node quadrilateral,[10] with linear pressure, discontinuous over the element boundaries.

All these elements have an accuracy of $O(h^3)$ for the velocity and $O(h^2)$ for the pressure.

One of the main difficulties associated with the integrated solution method is that the pressure does not appear in the continuity equation. Hence the system of equations to be solved has the following structure:

$$\begin{bmatrix} \mathbf{S(u)} & \mathbf{L}^T \\ \mathbf{L} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \mathbf{p} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ 0 \end{bmatrix} \tag{14}$$

Owing to the zeros on the main diagonal, pivoting may be necessary, resulting in a considerable increase of computation time and memory required.

Therefore one has tried to overcome these difficulties by constrained optimization techniques. In the following sections we shall consider two methods that separate the pressure computation from the velocity computation, thus avoiding the necessity of pivoting. These methods must be considered as efficient solvers for the system of equations (14) and hence must be applied only to elements satisfying the BB condition.

THE PENALTY FUNCTION APPROACH

One of the oldest methods to separate the pressure and velocity computation is the penalty function approach.[11] The method is based on techniques from constrained optimization[12] and the observation that the Navier–Stokes equations in the absence of convective terms can be formulated as a minimization problem.[9]

The idea is to replace the continuity equation (2) by a perturbed equation:

$$\varepsilon p - u_{i,i} = 0 \tag{15}$$

and then to apply the Galerkin method, resulting in

$$\varepsilon \int_\Omega p^h \delta p^h \, dx - \int_\Omega u^h_{i,i} \delta p^h \, dx = 0 \tag{16}$$

With (16) we can express the pressure in terms of the velocity and thus eliminate it from (9).

Using the notation of equation (14), the resulting system of equations can be formulated as

$$\begin{aligned} &\left\{\mathbf{S}(\mathbf{u}) + \frac{1}{\varepsilon}\mathbf{L}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{L}\right\}\mathbf{u} = \mathbf{f} \\ &\mathbf{p} = \frac{1}{\varepsilon}\mathbf{M}^{-1}\mathbf{L}\mathbf{u} \end{aligned} \tag{17}$$

For the Stokes equations one can prove[13] the convergence of the method as $\varepsilon \to 0$. Practical computations have shown that the method works for the Navier–Stokes equations as well. In practice ε should be chosen in the range 10^{-4} to 10^{-9} depending on the accuracy of the computer used.

The penalty method requires the computation of the matrix $(1/\varepsilon)\mathbf{L}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{L}$, which for the discontinuous pressure elements (iii) and (iv) can be carried out elementwise. Moreover for the continuous pressure elements (i) and (ii) the structure of the matrix $(1/\varepsilon)\mathbf{L}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{L}$ differs from the structure of $\mathbf{S}$ (larger bandwidth), and hence in general the penalty function approach is combined only with discontinuous pressure elements.

Computations have shown that the penalty function method can be 5 to 10 times faster than the integrated formulation. A disadvantage is the parameter ε, which for small values causes loss of accuracy and sometimes for too large values prevents convergence to the solution of system (14). Furthermore, the condition number of the system of equations (17) is large, since the matrix $\mathbf{L}^{\mathrm{T}}\mathbf{M}^{-1}\mathbf{L}$ is singular and ε small. Therefore this system cannot be solved by iterative matrix techniques.

An alternative form of the penalty function approach is to substitute the perturbed equation (15) into the Navier–Stokes equations (1) and (3), thus eliminating the pressure from the differential equations. In this way the perturbed Navier–Stokes equations can be written as

$$-\sigma_{i,j} + \rho(u_i u_j)_{,j} = \rho f_i \tag{18}$$

$$\sigma_{ij} = -\frac{1}{\varepsilon}u_{i,i}\delta_{ij} + \eta(u_{i,j} + u_{j,i}) \tag{19}$$

Application of the standard Galerkin method to equations (18) and (19) results in the following weak formulation:

$$\int_\Omega \eta(u_{i,j}+u_{j,i})\delta u_{i,j}\,\mathrm{d}x+\frac{1}{\varepsilon}\int_\Omega u_{j,j}\delta u_{i,i}\,\mathrm{d}x+\int_\Omega \rho(u_iu_j)_{,j}\delta u_i\,\mathrm{d}x$$
$$=\int_\Omega \rho f_i\delta u_i\,\mathrm{d}x+\int_{\partial\Omega}\{\sigma_t\delta u_t+\sigma_n\delta u_n\}\,\mathrm{d}s \tag{20}$$

Since in (20) no pressure terms are present, it seems as if (20) can be solved with any element even if it does not satisfy the B–B condition. However, computations have shown that the solution of (20) with most finite elements, suffers from chequerboard modes for the pressure when the penalty term (involving $1/\varepsilon$) is integrated exactly. Most authors suggest the use of reduced integration for this term (see for example Reference 14) in order to suppress the spurious pressure modes. It is easy to see that exact integration results in spurious pressure modes, since for small values of ε, (20) is equivalent to the solution of (9) under the constraint

$$\int_\Omega \delta u_{j,j}u_{i,i}\,\mathrm{d}x=0 \tag{21}$$

Hence the divergence of the velocity plays the role of the pressure. Now the divergence of the discretized velocity is discontinuous, and it is well known that discontinuous pressure elements with interpolation polynomials of the same degree for the pressure as for the divergence of the velocity do not satisfy the B–B condition. For example in the extended quadratic triangle the pressure must be linear, whereas the divergence of the velocity is linear plus some extra quadratic term. In fact reduced integration forces the divergence of the velocity to a lower degree term. From a practical point of view the penalty formulation (16, 17) is therefore preferred to the continuous formulation (20). In the penalty formulation the B–B condition is satisfied, whenever the original mixed interpolation element satisfies this condition.

SOLENOIDAL ELEMENTS

A more recent development is the construction of approximately divergence-free basis functions.[15] These basis functions are constructed so that they satisfy equation (10) exactly in the Galerkin sense, i.e.

$$-\int_\Omega \delta p^h u^h_{i,i}\,\mathrm{d}x=0 \tag{22}$$

Since the basis functions are divergence-free the term

$$\int_\Omega p^h\delta u^h_{i,i}\,\mathrm{d}x$$

in (9) is zero, and hence the pressure drops out from the momentum equations.

Divergence-free basis functions have only been constructed for discontinuous pressure elements. Here we give the derivation for the extended quadratic triangle; the 9-point biquadratic quadrilateral can be treated in the same way.

After elimination of the centroid by (13) the continuity equation (10) reduces in each element e to (12):

$$\int_e u^h_{i,i}\,\mathrm{d}x=\int_{\partial\Omega}\mathbf{u}^h\cdot\mathbf{n}\,\mathrm{d}s=0 \tag{23}$$

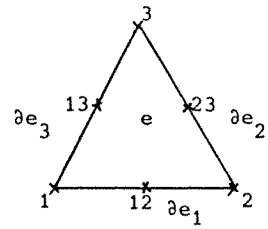

Figure 2. Element e

In order to construct a vector field with normal velocity components satisfying (23) a stream function ψ^h is introduced according to

$$\mathbf{u}^h = \left(\frac{\partial \psi^h}{\partial x_2}, -\frac{\partial \psi^h}{\partial x_1} \right)^{\mathrm{T}} \tag{24}$$

Since $\mathbf{u}^h \cdot \mathbf{n} = \nabla \psi^h \cdot \mathbf{t}$, for each side of the element we have (cf. Figure 2)

$$\int_{\partial e_i} \mathbf{u}^h \cdot \mathbf{n} \, \mathrm{d}s = \int_{\partial e_i} \nabla \psi^h \cdot \mathbf{t} \, \mathrm{d}s = \psi^h_{i+1} - \psi^h_i \quad (\psi_4 \equiv \psi_1) \tag{25}$$

The introduction of a function ψ^h by (25) forces the solution to satisfy (23). From (25) we can eliminate the normal velocity components at the midpoints of the sides and express them in terms of the stream function and other velocity components.

Figure 3 shows the effect of the introduction of the divergence-free basis functions.

Compared to the penalty function method the number of unknowns is not reduced per element. However, since the number of vertices is much smaller than the number of midpoints of sides, the total number of degrees of freedom is decreased.

For the practical implementation of the solenoidal approach we do not actually construct the divergence-free basis functions but express the new unknowns in terms of the old ones with the aid of a transformation matrix $\mathbf{R}$ such that

$$\mathbf{u} = \mathbf{R}\hat{\mathbf{u}} \tag{26}$$

with $\mathbf{u}$ the vector of velocity components in all nodal points before transformation and $\hat{\mathbf{u}}$ the vector of the new degrees of freedom, that is $\hat{\mathbf{u}}$ contains the velocity components and the stream function in the vertices and the tangential components at the midpoints of the sides of the elements.

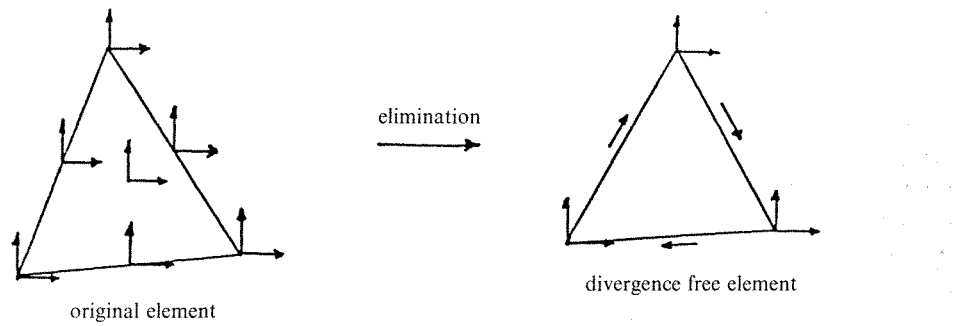

Figure 3. Original element: 1 pressure, 12 velocity components. Divergence free element: no pressure, 2 velocity components at vertices, tangential velocity components at midpoints of sides and stream-function unknowns at vertices

The matrix $\mathbf{R}$ is easily constructed elementwise from equations (25) and (13) and is such that

$$\mathbf{LR}\hat{\mathbf{u}} = 0 \quad \text{for each vector } \hat{\mathbf{u}} \tag{27}$$

Hence $\mathbf{LR} = 0$.

Equation (14) then reduces to

$$\mathbf{R}^{\mathrm{T}}\mathbf{S}(\mathbf{u})\mathbf{R}\hat{\mathbf{u}} = \mathbf{R}^{\mathrm{T}}\mathbf{f} \tag{28}$$

For a more detailed discussion see Reference 15.

It is important that the method can be carried out at element level using a standard element code, since it is sufficient to construct element matrices $\mathbf{R}^{e\mathrm{T}}\mathbf{S}^{e}\mathbf{R}^{e}$ and element vectors $\mathbf{R}^{e\mathrm{T}}\mathbf{f}^{e}$, where $\mathbf{S}^{e}$ and $\mathbf{f}^{e}$ are the classical element matrices and vectors for the Navier–Stokes equations.

Computation of the pressure

The pressure can be computed by substituting the eliminated basis functions for the test functions δu_i in the momentum equations (9).

Substitution of the 'bubble' function results in an expression for ∇p, that can be computed elementwise when the velocity and the pressure in the centroid are known. Substitution of the basis functions corresponding to the normal velocity components at the midpoints results in a relation between the pressures in the centroids of adjacent elements.[15] Hence, given the pressure in one element, it can be computed in all other elements by searching adjacent elements.

Once the pressure and its gradients are computed in each element they must be computed at the vertices of the elements. An averaging procedure must be used, since the pressure is discontinuous and hence has different values in different elements at the same nodal point.

Boundary conditions

The boundary conditions for the velocities must be transformed into boundary conditions for the new degrees of freedom, i.e. velocities and stream function. It is not possible to transform the boundary conditions at element level because the matrix $\mathbf{R}^{e}$ in (26): $\mathbf{u}^{e} = \mathbf{R}^{e}\mathbf{u}^{e}$ is singular, which can be easily verified by addition of equations (25) for one element. The singularity of this matrix is in agreement with the fact that the stream function is not unique.

For the transformation of velocities to stream-function variables, equation (25) can be used. Since the stream function is fixed except for an additive constant we can prescribe the stream function at one point of the boundary. Equation (25) can be used as long as the normal components of the velocities are given along the boundary. When the normal components are given except for one closed part of the boundary (for example an outflow part), we let the stream function be free along that part.

A more difficult situation arises when the normal components are given except for more than one (disjoint) part of the boundary, for example when we have an obstacle in the flow (Figure 4).

In that case we have the boundary conditions $\psi = c$ with c an unknown constant. For example in Figure 4 such a boundary condition is given along Γ_5.

The following method may be used to incorporate the boundary condition $\psi =$ constant in an easy way in existing codes.

The boundary condition $\psi =$ constant along a side Γ can be considered as the solution of the equation

$$\frac{\mathrm{d}^2\psi}{\mathrm{d}s^2} = 0 \tag{29}$$

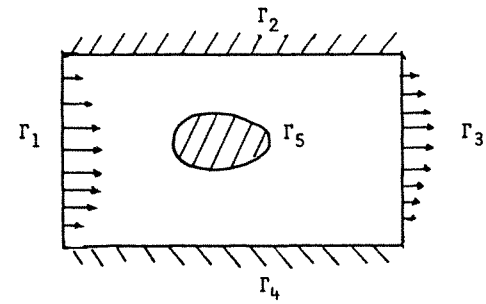

Figure 4. Flow around an obstacle

with boundary conditions

$$\frac{d\psi}{ds}(s_0) = \frac{d\psi}{ds}(s_n) = 0 \tag{30}$$

with s_0 the starting and s_n the endpoint of Γ, and s the direction along Γ.

Equation (29) can easily be solved using simple line elements for the Laplace equation. The boundary conditions are natural boundary conditions for this problem. If we add these line elements and multiply them by a large number (for example 10^4 to 10^9) we force the solution to satisfy the condition that ψ is constant. In fact we have used a penalty function method to satisfy the boundary conditions.

LINEARIZATION OF THE CONVECTIVE TERMS

In order to solve the Galerkin equations it is necessary to linearize the convective terms. Well-known linearization methods are, for example, Picard iteration (successive substitution), Newton linearization and quasi-Newton methods. Picard iteration is known to converge slowly with a linear convergence speed. The convergence of the Newton process is quadratic as soon as an iteration is in the neighbourhood of the solution. However, Newton is very sensitive to the initial guess; an inaccurate starting value may cause divergence of the iteration process. One of the methods to overcome this difficulty is to use a so-called continuation process, for example start with a low Reynolds number and increase the Reynolds number during the iteration process. Although Newton converges very fast (usually 5 to 10 iterations are sufficient) it requires the building of Jacobian matrices and the solution of linear systems of equations in each iteration step. Quasi Newton methods, using the Broyden update[16] as described by Engelman *et al.*[17] require the computation of Jacobian matrices only once in, for example, 5 steps. Their slower convergence behaviour (super-linearly) is regained by the fact that fewer Jacobian matrices have to be computed and fewer linear systems of equations have to be solved. Hence a good solution strategy seems to be: start with one Picard iteration as initial guess, and continue with the quasi-Newton methods.

The general approach to compute the Jacobian matrix is to discretize the Galerkin equations and to compute the partial derivatives of these discretized equations. However, a simpler way to derive the Jacobian matrix is to linearize the equations before application of the Galerkin method. The results of both methods are identical.

Let u_i^n be the result of the preceding iteration. Then the convective terms as, for example $(u_i u_j)_{,j}^{n+1}$, are linearized in the following way:

$$(u_i u_j)_{,j}^{n+1} \simeq (u_i^{n+1} u_j^n)_{,j} + (u_i^n u_j^{n+1})_{,j} - (u_i^n u_j^n)_{,j} \tag{31}$$

Equation (31) can be derived by substituting $u_i^{n+1} = u_i^n + \Delta u_i^n$ and neglecting quadratic terms in the correction Δu_i^n.

COMPARISON OF THE THREE METHODS

Considering the integrated solution method and comparing it with the segregated solution methods of discussed earlier we conclude that the last two are much more attractive, mainly because of the considerable savings in computing time and computer memory. Therefore we have limited ourselves to a comparison of the penalty function method and the solenoidal approach.

Advantages of the penalty function method

(i) The method is easily implemented; the computation of the pressure is straightforward.
(ii) The method can be extended without difficulty to three-dimensional problems.

Disadvantages of the penalty function method

(i) In some cases the value of ε may cause problems, however in general the range in which ε may be chosen is large.
(ii) Iterative solution methods to solve the linear systems of equations are not applicable because of the ill-condition due to the smallness of ε.

Advantages of the solenoidal approach

(i) The number of degrees of freedom is decreased, and hence the computing time and the required memory to solve the system of equations. However, the building of matrices and vectors requires more computing time, owing to the multiplications with the transformation matrices.
(ii) Since no special parameter has to be chosen, the resulting system of equations can be solved by iterative methods. Of course the boundary condition that ψ is constant has to be treated differently in that case.

Disadvantages of the solenoidal approach

(i) The implementation of the method is more complicated, since the boundary conditions must be transformed, the computed solution must be transformed back into the original variables,

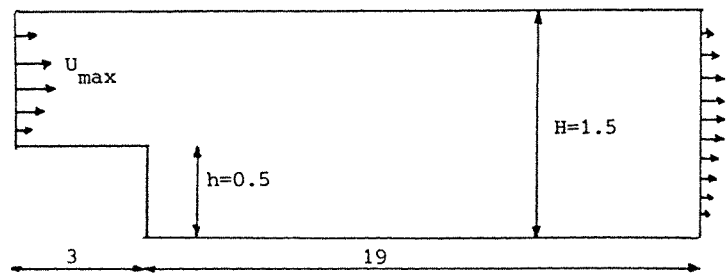

Figure 5. Backward facing step

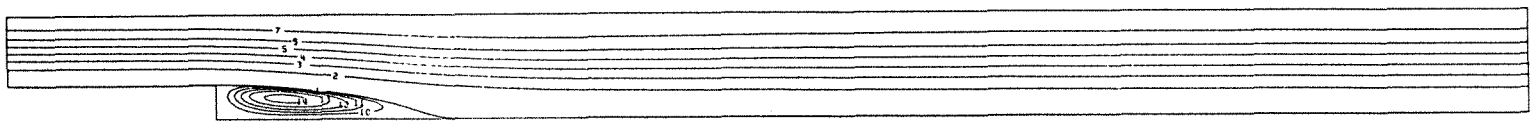

Figure 6. Streamline plot

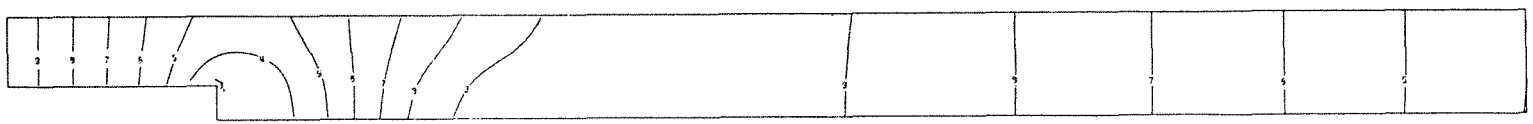

Figure 7. Pressure plot. Pressure level i: $p_i = -2 + (i-1)*2$

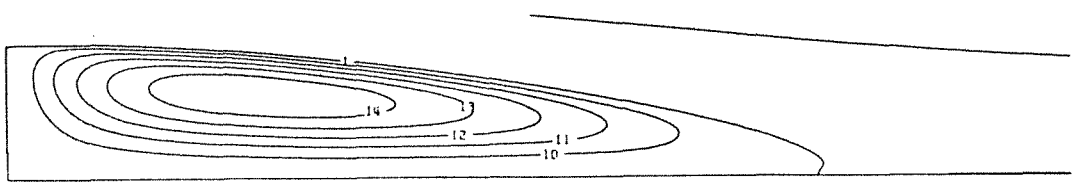

Figure 8. Streamline plot in recirculation zone: $\psi_{max} = 0{\cdot}0308$. Length of recirculation zone: 6·1

and also because the computation of the pressure requires a searching method for neighbouring elements.

(ii) The extension to three dimensions is not straightforward (except for axisymmetric cases). In Reference 18 some computations with solenoidal elements in $\mathbb{R}^3$ are reported and also Griffiths[3] treats a way to solve this problem.

PRACTICAL COMPARISON

In order to make comparisons for a practical problem we have computed the solution of the flow over a backward facing step with both methods. Figure 5 gives the definition of the problem.

The Reynolds number Re is defined as $Re = U_{\max}h/v$ with v the kinematic viscosity. The computations were performed for $Re = 150$, with boundary conditions:

fixed walls	:no-slip $\mathbf{u} = \mathbf{0}$
inlet	:given velocity profile
outlet	:$u_t = 0$ and $\sigma_n = 0$

In Figure 6 the streamlines, in Figure 7 the pressure contours and in Figure 8 an enlargement of the streamlines in the recirculation zone are given.

In Table I we give comparisons of the computing time to build the matrix and to solve the system of equations for one iteration. Furthermore the number of elements in the large matrix is given. The problem is solved with a profile method and took 5 Newton iterations to get a difference between two succeeding iterations of less than 10^{-5}. The extended quadratic triangle is used. These numbers show that the time to build the matrix is larger for the solenoidal approach, but the time to solve the system of equations for one iteration is considerably shorter. For coarse meshes the total computing time is comparable, since the building of the matrix takes as much time as the solution of the equations. For fine meshes however the solenoidal method is 25 per cent cheaper than the penalty function method.

Table I. Comparison of the penalty function method and the divergence free elements for 4 meshes.

	Divergence free elements				Penalty function method			
Mesh number	Number of degrees of freedom	Length of matrix	CPU 1 iter matrix	CPU 1 iter SOLVE	Number of degrees of freedom	Length of matrix	CPU 1 iter matrix	CPU 1 iter SOLVE
1	179	2960	0·40	0·09	222	4437	0·33	0·16
2	619	23,734	1·65	1·43	794	34,237	1·37	2·27
3	2291	190,718	6·87	21·10	2994	265,949	5·60	30·45
4	2435	217,408	7·38	25·50	3186	302,565	6·02	36·55

REFERENCES

1. D. H. Norrie and G. de Vries, 'A survey of the finite element applications in fluid mechanics', in *Finite element in Fluids*, Vol. 3, Wiley, Chichester, 1978, pp. 363–396.
2. M. Fortin, 'Old and new finite elements for incompressible flows', *Int. J. numer. methods fluids*, **1**, 347–364 (1981).
3. D. F. Griffiths, 'An approximately divergence-free 9-node velocity element (with variations) for incompressible flows', *Int. J. numer methods fluids*, **1**, 323–346 (1981).
4. R. L. Sani, P. M. Gresho, R. Lee and D. F. Griffiths, 'The cause and cure (?) of the spurious pressures generated by certain FEM solutions of the incompressible Navier–Stokes equations', Part 1, *Int. J. numer methods fluids*, **1**, 17–43 (1981); Part 2, **1**, 171–204 (1981).
5. M. S. Engelman, R. L. Sani, P. M. Gresho and M. Bercovier, 'Consistent vs. reduced integration penalty methods for incompressible media using several old and new elements', *Int. J. numer methods fluids*, **2**, 25–42 (1982).

6. F. Thomasset, *Implementation of Finite Element Methods for Navier–Stokes Equations*, Springer, Berlin, 1981.
7. C. Taylor and P. Hood, 'A numerical solution of the Navier–Stokes equations using the finite element technique', *Comput. Fluids*, **I**, 73–100 (1973).
8. A. Segal, 'On the numerical solution of the Stokes equations using the finite element method', *Comp. Meth. in Appl. Mech. and Eng.*, **19**, 165–185 (1979).
9. M. Crouzeix and P. A. Raviart, 'Conforming and non-conforming finite element methods for solving the stationary Stokes equations', *Rev. Francaise Autom. Informat. Recherche Opérationelle*, **7**, 33–76 (1973).
10. M. Bercovier and M. Engelman, 'A finite element for the numerical solution of viscous incompressible flows', *J. Comp. Physics*, **30**, 181–201 (1979).
11. O. C. Zienkiewiez and P. N. Godbole, 'Viscous incompressible flows with special reference to non-newtonian (plastic) fluids', in *Finite Elements in Fluids*, Vol. 1, Wiley, Chichester, 1975, pp. 25–56.
12. W. Murray and P. Gill, *Numerical Methods for Constrained Optimization*, Academic Press, London, 1974.
13. R. Temam, *Navier–Stokes Equations*, North Holland, Amsterdam, 1977.
14. D. S. Malkus and T. J. R. Hughes, 'Mixed finite element methods—reduced and selective integration techniques: a unification of concepts', *Comp. Methods Appl. Mech. and Eng.*, **15**, 63–81 (1978).
15. D. F. Griffith, 'Finite elements for incompressible flow', *Math. Meth. in Applied Science*, **1**, 16–31 (1979).
16. C. G. Broyden, 'A class of methods for solving nonlinear simultaneous equations', *Math. Comp.*, **19**, 577–593 (1965).
17. M. S. Engelman, G. Strang and K. J. Bathe, 'The application of quasi-Newton methods in fluid mechanics', *Int. J. numer. methods eng.*, **17**, 707–718 (1981).
18. F. Hecht, 'Construction d'une base de fonctions P_1 non conforme a divergence nulle dans $\mathbb{R}^3$', *R.A.I.R.O, Analyse Numerique*, **15**, (2), 119–150 (1981).