JOURNAL OF COMPUTATIONAL PHYSICS 51, 273–290 (1983)

Analysis of Some Low-Order Finite Element Schemes for the Navier–Stokes Equations

M. J. P. CULLEN

Forecasting Research Branch, Meteorological Office, Bracknell, Berkshire, England

Received January 26, 1982; revised December 7, 1982

Various low-order finite element schemes for the Navier–Stokes equations are analysed, including the commonly used bilinear velocity, piecewise constant pressure element, and the algorithms based on the finite element analogue of the MAC stencil discussed in a companion paper. By performing a truncation error analysis using a variety of different assumptions about how the nodal parameters represent the data, it is shown how a more complete picture of the behaviour of the algorithms can be obtained. In particular, differences in performance on coarse grids using algorithms which have the same asymptotic convergence rate can be explained.

1. INTRODUCTION

This paper is an attempt to gain a better understanding of the accuracy of finite element methods for the Navier–Stokes equations and how they relate to other methods. The standard analysis of finite element approximations to these equations, as set out, for instance, by Temam [1] and Heywood and Rannacher [2], gives error estimates with constants increasing with Reynolds number, and only applicable to a very restricted set of finite element approximations. However, experiments with finite element methods suggest that they can be just as capable as finite difference methods of giving good results at high Reynolds number; for instance, the vortex shedding computations of Gresho *et al.* [3], and the results discussed in the recent review paper of Hughes *et al.* [4]. These results, amongst others, also show that the estimates of [1] and [2] could be extended to a wider class of elements. There are, nevertheless, difficulties. An example of these is the vortex shedding experiment discussed by Gresho *et al.* ([3, 5] and private communication). It was found that a standard low-order finite element approximation using piecewise bilinear velocities and piecewise constant pressures gave inaccurate results on some meshes, but that if the piecewise constant pressures were replaced by bilinear pressures, then the velocity fields were much more accurate. The piecewise constant pressure element also yielded good results if the mesh was designed much more carefully round the cylinder. The same authors (private communication) had a similar experience with a steady Boussinesq flow over a wedge. The piecewise constant pressure element could not reproduce the

0021-9991/83 $3.00

basic hydrostatic pressure gradient and a steady state of rest could not be obtained. Though the problem could be alleviated by subtracting off the hydrostatic pressure for an isothermal fluid, the results were still judged to be unsatisfactory. The piecewise bilinear pressure approximation yielded the desired steady state at rest. It was also found, nevertheless, that the inaccurate velocity fields obtained using piecewise constant pressures converged to the state of rest at a second-order rate as the mesh was refined.

The standard approximation theory of [1] and [2] does not distinguish between the convergence rate in velocity of the schemes using bilinear and piecewise constant pressures. In order to explain the results a different type of analysis is required. It is shown that the observed behaviour on coarse grids is consistent with finite difference truncation error estimates for the schemes. This suggests that it is worth performing more than one type of error analysis. The range of validity of any form of asymptotic truncation error analysis is limited and cannot usually be determined mathematically because the constants are not known. Only practical experience will suggest what it is. The results in this paper, which show that the coarse grid behaviour described in [3] and [5] is more consistent with a finite difference analysis than the usual finite element analysis, suggest that the range of validity of the finite difference analysis is greater in these cases. When the mesh is refined and smoothed, the behaviour is as predicted by the finite element analysis, which predicts a more rapid convergence than the finite difference analysis.

In this paper the differences between the alternative analyses are first demonstrated for simple problems. A generalised truncation error analysis is then described, based on the ideas of [10], which includes both the standard analysis of [2] and a purely finite difference analysis as special cases. It is shown how this explains the results of [3] and [5]. In Section 4 it is shown how truncation error analysis can predict the behaviour of the schemes tested by the author in [17]. Since neither of the schemes is a standard Galerkin scheme, the analysis of [1] and [2] is not strictly applicable.

It should be made clear that the aim is not to replace the standard analysis, but to show that more information can be gained about the behaviour of schemes by performing a variety of alternative analyses.

2. Error Analysis of Simple Problems

2.1. *Alternative Analyses for the Heat Equation*

In order to motivate what follows, we first discuss briefly the simple case of the heat equation, which is well understood:

$$\begin{aligned} &\frac{\partial u}{\partial t} = K\nabla^2 u \qquad \text{on } \Omega, \\ &u = u_0 \qquad \text{at } t = 0, \\ &u \text{ given} \qquad \text{or } \partial\Omega. \end{aligned} \tag{2.1}$$

The semi-discrete finite element Galerkin approximation to this equation using piecewise bilinear functions to represent u is well known to converge to the exact solution at an $O(h^2)$ rate (e.g., Strang and Fix [6]). This estimate applies even on distorted grids where a local isoparametric coordinate transformation is used, and u is approximated by piecewise bilinear functions in the transformed coordinates (ξ, η). The approximation can be written in the form

$$\begin{aligned} \int_\Omega J \frac{\partial u_h}{\partial t} \chi_n \, d\xi \, d\eta &= -K \int_\Omega \nabla u_h \cdot \nabla \chi_n J \, d\xi \, d\eta \qquad \forall \text{ interior } n, \\ u_h &= u_{h0} \qquad \text{at} \quad t = 0, \\ \int (u_{h0} - u_0) \chi_n J \, d\xi \, d\eta &= 0, \end{aligned} \tag{2.2}$$

where the functions χ_n form a basis for the space of all piecewise bilinear functions in ξ and η. No equation is required for boundary nodes in this case, and the boundary integral terms vanish.

Consider the local patch of elements shown in Fig. 1. The system of equations (2.2) contains an equation associated with each interior node, where χ_n is a piecewise bilinear function equal to 1 at that node and zero at all the remaining nodes. The equation associated with the node numbered 5 in Fig. 1 is

$$\begin{aligned} &\frac{1}{32}\left(\frac{\partial u_1}{\partial t} + \frac{\partial u_3}{\partial t} + \frac{\partial u_7}{\partial t}\right) + \frac{1}{24}\frac{\partial u_9}{\partial t} + \frac{1}{8}\left(\frac{\partial u_2}{\partial t} + \frac{\partial u_4}{\partial t}\right) \\ &\quad + \frac{19}{144}\frac{\partial u_6}{\partial t} + \frac{7}{48}\frac{\partial u_8}{\partial t} + \frac{37}{72}\frac{\partial u_5}{\partial t} \\ &= \frac{1}{3}(u_1 + u_2 + u_3 + u_4 + u_7) + \left(1 - \frac{16}{3}\log 2\right) u_5 \\ &\quad + \left(\frac{16}{3}\log 2 - \frac{10}{3}\right) u_6 + \left(\frac{8}{3}\log 2 - \frac{4}{3}\right) u_8 + \left(2 - \frac{8}{3}\log 2\right) u_9. \end{aligned} \tag{2.3}$$

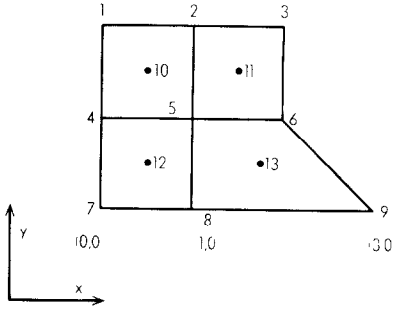

FIG. 1. Sample four element patch with node numbering.

If this were interpreted as a finite difference approximation to (2.1) then it has $O(1)$ error, as can be seen by substituting successively the polynomials $u = 1, x, y, x^2, xy$ and y^2. The coefficients of the left-hand side add to 13/12. Therefore the right-hand side should take the values, respectively, 0, 0, 0, 13/6, 0, 13/6. The right side of (2.3) gives 0, 0, 0, 2.3035, 0.0299, 2. A consistent first-order finite difference scheme has to reproduce these exactly. Thus the scheme (2.3) would have $O(1)$ error as a finite difference scheme. The reason for this is well known. It is not valid to regard nodal values of u_h as if they were interpolated values. They have to define the Ritz projection of u into the finite element space, given by

$$\int_\Omega \nabla(u_h - u) \cdot \nabla\chi_n J \, d\xi \, d\eta - \int_{\partial\Omega} \chi_n \frac{\partial}{\partial n}(u_h - u) \, ds = 0. \tag{2.4}$$

The standard finite element convergence proof starts by showing that $\|u_h - u\|_{L_2} = O(h^2)$ by using regularity properties of the Laplacian operator on Ω with the given boundary conditions. Thus the point values u_1 to u_9 of u_h in Fig. 1 are not interpolated values, but may depart by $O(h^2)$ from interpolated values. Second differences of these values, such as the right side of (2.3), may thus differ by $O(1)$, as is found in this case.

This standard example emphasises several points:

(i) A finite element scheme which converges at a second-order rate may be quite different from a finite difference scheme of second-order accuracy.

(ii) The error in the finite element scheme for (2.1) is not locally determined, since the projection (2.4) is global. This would be a serious disadvantage if, instead of (2.1), the second-order wave equation was being approximated.

(iii) The apparent truncation error of a scheme depends on the assumption made about how the data is represented.

2.2. *Alternative Analyses for the Advection–Diffusion Equation*

The advection–diffusion equation is the next standard system normally studied before the full Navier–Stokes equations. This can be written either as

$$\frac{\partial C}{\partial t} + \mathbf{u} \cdot \nabla C + P^{-1}\nabla^2 C = 0 \tag{2.5}$$

with suitable initial and boundary conditions, or as

$$\frac{\partial C}{\partial t} + P\mathbf{u} \cdot \nabla C + \nabla^2 C = 0. \tag{2.6}$$

The choices (2.5) and (2.6) are appropriate for high and low values, respectively, of the Peclet number P. There has been extensive recent analysis of the finite element Galerkin method applied to this problem, together with development of alternative

algorithms for it, (see, for instance, Barrett and Morton [7, 8] and Hughes and Brooks [9]). The projection (2.4) now becomes

$$\int_\Omega (\chi_n \mathbf{u} \cdot \nabla(C_h - C) - P^{-1}\nabla(C_h - C) \cdot \nabla\chi_n) J \, d\xi \, d\eta$$

$$- \int_{\partial\Omega} \chi_n \frac{\partial}{\partial n}(C_h - C) \, ds = 0. \tag{2.7}$$

This projection can still be used to prove that the Galerkin algorithm converges as $O(h^2)$, but the constant now includes the Peclet number P. Thus the convergence estimate for the steady problem and the truncation error estimate for the unsteady problem (2.5) are both very weak except for extremely small h. In the case of the steady problem it is common to change the algorithm, as discussed in [7–9]. The change is analogous to a change from centred to upwind finite differencing. For the unsteady problem (2.5) with large P, however, the equation is almost a pure advection equation, and in many applications the Galerkin method, or centred differencing, gives quite reasonable answers and the debate as to whether it, or "upwind"-type methods, should be used still continues. The behaviour of the Galerkin method for (2.5) can be explained by a truncation error analysis using not the Ritz projection (2.7) but a least-squares fit

$$\int_\Omega \chi_n (C_h - C) J \, d\xi \, d\eta = 0. \tag{2.8}$$

If we write (2.5) in the general form

$$\frac{\partial C}{\partial t} + LC = 0 \tag{2.9}$$

and the Galerkin approximation to it in the form

$$\frac{\partial C_h}{\partial t} + L_h C_h = 0, \tag{2.10}$$

then it was shown in Cullen and Morton [10] that the truncation error can be written in the form

$$(LC)_h - L_h C_h. \tag{2.11}$$

In the case of (2.5), using the bilinear approximation to C on quadrilaterals, and supposing for the moment that $\mathbf{u}$ is constant, this error is of the form

$$A_1 |\mathbf{u}| h + A_2 P^{-1}, \tag{2.12}$$

where A_1 and A_2 are constants independent of $\mathbf{u}$ and P, using basic approximation theory results (e.g., [6]). If P is large, the second term in (2.12) may be negligible

except in boundary layers. In addition, numerical experiments described by the author in [11] suggest that the first term of (2.12) is quite small for large h, provided that the data is represented by the least-squares fit (2.8). The reason for this is that the Galerkin method for a *first*-order operator L calculates the best least-squares fit to LC_h, and this can be expected to be close to the best least-squares fit to LC even on a coarse mesh. This would be reflected in the estimate (2.12) by a small value of the constant A_1.

In the boundary layer, where the second term of (2.12) is not negligible, the Galerkin method is well known to have problems, since, unless the boundary layer is resolved properly by a stretched coordinate, the solution oscillates. The remedy is either to increase the resolution, as advised by Gresho and Lee [12], or to use a Petrov Galerkin method as in [7–9]. In practice, the results obtained by Chan and Gresho [5], using the Galerkin method, and by Brooks [13], using the Petrov–Galerkin method, for vortex shedding at Reynolds number 100 are almost identical, which suggests that the distinction may not matter very much.

This example illustrates that by considering the least-squares representation (2.8) as well as the Ritz representation (2.5) more insight can be obtained into the behaviour of finite element schemes. In particular, error estimates can be obtained independent of global regularity constraints and dimensionless parameters. Such estimates, in general, will have smaller powers of h but also smaller constants. They may thus be a better guide to performance on coarse grids. For very small h the optimal estimates obtained using (2.7) take over.

3. Analysis of Schemes for the Navier–Stokes Equations

3.1. *Introduction*

Now consider incompressible viscous flow governed by the dimensionless Navier–Stokes equations

$$\frac{\partial \mathbf{u}}{\partial t} + \mathbf{u} \cdot \nabla \mathbf{u} + \nabla p = \frac{1}{R} \nabla^2 \mathbf{u}, \tag{3.1}$$

$$\nabla \cdot \mathbf{u} = 0, \tag{3.2}$$

on a region Ω, with boundary conditions

$$\mathbf{u} = \mathbf{f} \qquad \text{on } \partial\Omega, \tag{3.3}$$

and initial conditions

$$\begin{aligned} \mathbf{u} &= \mathbf{u}_0 \qquad \text{at} \quad t = 0, \\ \nabla \cdot \mathbf{u}_0 &= 0. \end{aligned} \tag{3.4}$$

R is the Reynolds number and the remaining notation is standard. We will make

frequent use of an alternative form of (3.1) and (3.2) discussed by Chorin and Marsden [14, p. 51]. It can be proved that any vector field $\mathbf{u}$ on Ω can be written as

$$P\mathbf{u} + \nabla q, \tag{3.5}$$

where $\nabla \cdot (P\mathbf{u}) = 0$; $P\mathbf{u} = \mathbf{f}$, on $\partial\Omega$, and q is a scalar field. Then (3.1)–(3.4) can be replaced by

$$\frac{\partial \mathbf{u}}{\partial t} = P\left(\frac{1}{R}\nabla^2\mathbf{u} - \mathbf{u}\cdot\nabla\mathbf{u}\right), \qquad \mathbf{u} = \mathbf{u}_0 \quad \text{at} \quad t = 0. \tag{3.6}$$

Numerical algorithms can be based on either system.

As in the previous sections, we will analyse some simple finite element approximations using a variety of assumptions about the way in which the data are represented. It is convenient to introduce some extra notation for this, in the manner of [10]. Only a brief summary of the construction is given here. For simplicity, first consider solutions of (3.6). This can be written in the form

$$\frac{\partial \mathbf{u}}{\partial t} = PL\mathbf{u}. \tag{3.7}$$

A Galerkin finite element approximation to (3.6) can be written in the form

$$\frac{\partial \mathbf{u}_h}{\partial t} = (PL)_h\,\mathbf{u}_h, \tag{3.8}$$

where $(PL)_h$ may or may not be separable as $P_h L_h$. To carry out an error analysis, assume that the method is attempting to represent the data by a projection r_h. Examples of this in Section 2.2 were the least-squares fit (2.8) or the Ritz projection (2.7). Then, as in [10], the error made in solving (3.7) can be split as follows:

$$\mathbf{u} - \mathbf{u}_h = (\mathbf{u} - r_h\mathbf{u}) + (r_h\mathbf{u} - \mathbf{u}_h). \tag{3.9}$$

Define

$$\mathbf{e}_h = (r_h\mathbf{u} - \mathbf{u}_h). \tag{3.10}$$

It obeys the equation

$$\frac{\partial \mathbf{e}_h}{\partial t} = (r_h(PL\mathbf{u}) - (PL)_h\, r_h\mathbf{u}) + ((PL)_h\, r_h\mathbf{u} - (PL)_h\,\mathbf{u}_h) \tag{3.11}$$

The first term on the right-hand side of (3.11) is the truncation error, and the second term is a growth term. A complete error estimate for the solution of (3.7) includes an estimate of the term $(\mathbf{u} - r_h\mathbf{u})$ and the evolution term $\mathbf{e}_h$. If r_h is the Ritz projection

for (3.1)–(3.4) we recover the standard convergence proof of [1, 2]. If r_h is a node point collocation operator, we obtain a finite difference error estimate as would be obtained by Taylor series methods.

In the steady case it is still possible to estimate the truncation error. In the case of (3.7), the steady problem is just

$$PL\mathbf{u} = 0 \tag{3.12}$$

so that the truncation error would be

$$-(PL)_h r_h \mathbf{u}. \tag{3.13}$$

In order to obtain the actual error, we then require a property of PL such that if

$$PLu_1 = f_1,$$

$$PLu_2 = f_2,$$

then

$$\|u_1 - u_2\| \leqslant C \|f_1 - f_2\|. \tag{3.14}$$

Such estimates for the Navier–Stokes equations are discussed in [1–2], and can only be obtained in special cases. Thus comparison of algorithms for the steady case is difficult, because whatever type of assumption is made about the data representation, the final error has to be calculated using (3.14).

3.2. *Analysis of Standard Mixed Interpolation Schemes in Two Dimensions*

We now seek an explanation of the results of [3] and [5]. It was found there that a Galerkin approximation to (3.1)–(3.4) using piecewise bilinear velocities and piecewise constant pressures could give inaccurate results. However, $O(h^2)$ convergence in velocity was observed in a steady problem and a transient problem could be solved accurately by careful mesh design. When the pressure approximation was made piecewise bilinear, and the mesh was otherwise unaltered, the results for the velocities were much more accurate, though the convergence rate was the same. Both types of approximation could give spurious pressure oscillations, especially with bilinear pressures.

Using the notation χ_n for piecewise bilinear functions on a quadrilateral mesh, with θ_n for piecewise constant functions, the standard mixed Galerkin approximation to (3.1)–(3.4) with these functions can be written as

$$\int_\Omega \left[\left(\frac{\partial \mathbf{u}_h}{\partial t} + \mathbf{u}_h \cdot \nabla \mathbf{u}_h \right) \chi_n - p_h \nabla \chi_n + \frac{1}{R} \nabla \mathbf{u}_h : \nabla \chi_n \right] J \, d\xi \, d\eta$$

$$+ \int_{\partial\Omega} (p_h \chi_n + \chi_n \nabla \mathbf{u}_h) \cdot ds = 0, \tag{3.15}$$

$$\int_\Omega \nabla \cdot \mathbf{u}_h \theta_n J \, d\xi \, d\eta = 0, \qquad \mathbf{u}_h \text{ given on } \partial\Omega \text{ and at } t = 0. \tag{3.16}$$

The correct way to enforce the boundary conditions is discussed by Engelman *et al.* [15], and the best choice of initial conditions depends on what view is taken of the method of data representation. In the same way as for the advection diffusion equation, the standard error analysis assumes that the data are represented by the Ritz projection $\mathbf{u} \to \mathbf{u}_h$ defined by

$$\int_\Omega \left[(\mathbf{u}_h \cdot \nabla \mathbf{u}_h - \mathbf{u} \cdot \nabla \mathbf{u}) \chi_n + (p - p_h) \nabla \chi_n + \frac{1}{R} \nabla(\mathbf{u}_h - \mathbf{u}) : \nabla \chi_n \right] J \, d\xi \, d\eta$$

$$+ \int_{\partial\Omega} [(p_h - p) \chi_n + \chi_n \nabla(\mathbf{u}_h - \mathbf{u})] \cdot ds = 0, \tag{3.17}$$

$$\int_\Omega \nabla \cdot (\mathbf{u}_h - \mathbf{u}) \theta_n J \, d\xi \, d\eta = 0.$$

A proof of convergence requires the Babuska–Brezzi compatibility condition, which involves

(i) a unique solution of (3.15), (3.16) for p, and

(ii) the property that a divergence free vector field $\mathbf{u}$ can be approximated to $O(h^2)$ by a piecewise bilinear field $\mathbf{u}_h$ which satisfies (3.16).

This condition can only be proved for a limited set of finite element approximations, not including this one. In this case (i) is known not to be true, and (ii) has not been proved. However, the computational tests of Gresho (private communication) suggest that (ii) is satisfied. The algorithm can be modified by filtering the spurious pressure solutions [16], and then (i) will also be satisfied. Convergence has been proved for a modified algorithm using reduced integration [20].

There are two problems with this analysis. The error estimate depends on R, the same difficulty as with the advection diffusion equation, and the same discussion applies. More seriously, the projection (3.17) involves $\mathbf{u}$ and p simultaneously. Thus it is not possible to extract an "energy" norm in which $\mathbf{u}_h$ is the best fit to $\mathbf{u}$ and p_h to p in any sense. Thus the possibility of recovering more accurate information is lost, which is a major advantage of finite element methods for self adjoint problems. An illustration of this is provided by the work of Gresho. A steady state of rest ($\mathbf{u} = 0$) with the pressure gradient balancing gravity and buoyancy terms could not be reproduced using bilinear velocities and piecewise constant pressures. It is clear that a best fit to a zero field in any norm is zero; so that in no sense does the Galerkin algorithm give a best fit to $\mathbf{u}$. The remainder of the analysis in this section is concerned with this second problem.

3.3. *Approximation of an Incompressible Field*

The extra difficulty in understanding solutions to the Navier–Stokes equations caused by the simultaneous approximation to $\mathbf{u}$ and p can be isolated by considering

the form (3.6). We consider simply how the projection P defined in (3.5) is represented. The truncation error extracted from (3.11) is

$$r_h(PL\mathbf{u}) - (PL)_h\, r_h\mathbf{u}. \tag{3.18}$$

Suppose for the present that $(PL)_h$ can be written in the form P_hL_h. Then (3.18) can be expanded, since P is linear, as

$$(r_hP - P_hr_h)(L\mathbf{u}) + P_h(r_hL\mathbf{u} - L_hr_h\mathbf{u}). \tag{3.19}$$

The important term at present is the first term in (3.19). It is sufficient to calculate the effect of

$$(r_hP - P_hr_h) \tag{3.20}$$

on a general vector field and to combine this with a truncation error estimate for the operator

$$L\mathbf{u} = \frac{1}{R}\nabla^2\mathbf{u} - \mathbf{u}\cdot\nabla\mathbf{u} \tag{3.21}$$

to give a complete estimate.

Equations (3.15), (3.16) can be written in this form, defining L_h by

$$\int_\Omega L_h\mathbf{u}_h\chi_n J\,d\xi\,d\eta = \int_\Omega \left[-\mathbf{u}_h\cdot\nabla\mathbf{u}_h\chi_n - \frac{1}{R}\nabla\mathbf{u}_h\cdot\nabla\chi_n\right] J\,d\xi\,d\eta$$

$$+ \int_{\partial\Omega}\chi_n\nabla\mathbf{u}_h\cdot ds \tag{3.22}$$

and P_h by

$$\int_\Omega P_h\mathbf{u}_h\chi_n J\,d\xi\,d\eta = \int_\Omega (\mathbf{u}_h\chi_n + q\nabla\chi_n)J\,d\xi\,d\eta - \int_{\partial\Omega} q_h\chi_n\cdot ds, \tag{3.23}$$

$$\int_\Omega \nabla\cdot(P_h\mathbf{u}_h)\,\theta_n J\,d\xi\,d\eta = 0. \tag{3.24}$$

The projection P_h is, however, only well defined if the Babuska–Brezzi condition is satisfied, allowing q to be uniquely determined.

Now evaluate the truncation error (3.20). While the standard error analysis of |1–2| does not use the decomposition (3.22)–(3.24), since an essential step in it uses a regularity property of the steady Stokes equations, it is clear that (3.20) can be no greater than the total truncation error (3.19). The computational results thus suggest that this error will be $O(h^2)$ for this element. As discussed in the Introduction, this leaves unexplained the behaviour on coarse grids. Therefore evaluate (3.20) using a grid point collocation definition for r_h. This is equivalent to performing a finite

difference truncation error analysis. Because of the implicit nature of the projection, there is a considerable amount of technical work to be done to convert an $O(h^\alpha)$ truncation error for the individual operators ∇ and $\nabla \cdot \mathbf{u}$ into an $O(h^\alpha)$ estimate for (3.20). This is discussed by Chorin [18] and is not repeated here. Henceforward we will assume that if errors can be estimated for the gradient and divergence operators separately, they can be combined into an estimate of (3.20).

Consider again the patch of elements in Fig. 1. The equation in (3.23) associated with node 5 for the y component of $\mathbf{u}$ is

$$\begin{aligned}\frac{1}{32}(Pv_1 + Pv_3 + Pv_7) + \frac{1}{24}Pv_9 + \frac{1}{8}(Pv_2 + Pv_4) + \frac{19}{144}Pv_6 + \frac{7}{48}Pv_8 + \frac{37}{72}Pv_5 \\ = \frac{1}{32}(v_1 + v_3 + v_7) + \frac{1}{24}v_9 + \frac{1}{8}(v_2 + v_4) \\ + \frac{19}{144}v_6 + \frac{7}{48}v_8 + \frac{37}{72}v_5 - \frac{13}{24}(q_{11} + q_{10} - q_{13} - q_{12}) \qquad (3.25)\end{aligned}$$

and the equation in (3.24) associated with node 13 is

$$\tfrac{1}{2}(u_6 - u_5 + u_9 - u_8) + \tfrac{1}{2}v_5 + v_6 - v_8 - \tfrac{1}{2}v_9 = 0. \qquad (3.26)$$

It is clear that the last term of (3.25) is not a consistent approximation to $\partial q/\partial y$ since it is non-zero if $q = x$. Equation (3.26) is a consistent approximation to $\partial u/\partial x + \partial v/\partial y$ since the terms in v can be treated as a difference between linearly interpolated values at $(1\frac{2}{3}, 1)$ and $(1\frac{2}{3}, 0)$ in Fig. 1. Thus the scheme has $O(1)$ error if regarded as a finite difference scheme.

Similar calculations, not repeated here, show that using piecewise bilinear velocities and pressures gives $O(h)$ error in this sense as do piecewise quadratic velocities with bilinear pressures. It can also be shown, after considerable algebra, that the same conclusion applies if, instead of a grid-point definition of r_h, we use a least-squares definition

$$\begin{aligned}\int_\Omega (r_h\mathbf{u} - \mathbf{u})\,\chi_n J\,d\xi\,d\eta = 0 \qquad \forall n, \\ \int_\Omega (r_h p - p)\,\theta_n J\,d\xi\,d\eta = 0 \qquad \forall n.\end{aligned} \qquad (3.27)$$

The second equation is exactly the same as the finite difference representation. Thus the observed behaviour of the schemes on coarse grids is consistent with either a finite difference or least-squares interpretation, while the ultimate convergence rate is consistent with the standard finite element analysis. This suggests that the range of validity of the finite difference analysis or the analysis using a least-squares representation is greater than the usual finite element analysis.

4. Approximation of the Navier–Stokes Equations Using Some Alternative Elements

4.1. *Introduction*

We now repeat the analysis of (3.20) using the finite element analogues of the MAC stencil introduced by DiCarlo and Piva [19]. The lowest-order element of this class uses discontinuous linear velocities defined by values at midside nodes, and piecewise constant pressures. A Galerkin-type algorithm using this element gives good results on a regular mesh, but fails to converge on a general rectangular mesh. In [19] it was proposed that the element should be used in conjunction with a global coordinate transformation, as with finite difference schemes. In a companion paper to this [17], it was shown how the element could be used with a local coordinate transformation if either finite differences were used for the pressure gradient term, or the algorithm was based on the vorticity. In this section we analyse these two algorithms.

In order to approximate the continuity equation properly, the unknowns are taken as normal mass fluxes across element boundaries, rather than velocities. Thus we write flux components (m, n) across sides $\xi = \text{constant}$ and $\eta = \text{constant}$. The component m depends only on ξ and is the total flux across the line $\xi = \text{constant}$ contained within one element (Fig. 2). This definition could be made in two or three dimensions, but in two dimensions, as used here, the method is really a disguised streamfunction method with (m, n) identified as $(-\partial\psi/\partial\eta, \partial\psi/\partial\xi)$, where ψ is a piecewise bilinear streamfunction.

The flux component m is assumed to be linear in ξ and independent of η and vice versa for n. The associated trial functions are written as λ_n and μ_n, respectively. Contours of a typical λ_n are shown in Fig. 2.

4.2. *Analysis of Scheme with Finite Difference Pressure Gradient*

The first algorithm used in [17] uses a Galerkin-type method for the operator L and a finite difference method for P, as follows:

$$\begin{aligned}
\int_\Omega \frac{\gamma}{J} L_h m_h \lambda_n \, d\xi \, d\eta &= \int_\Omega \left[\frac{\gamma}{J^2} (\alpha n_h + \beta m_h) \zeta_h + \frac{\gamma}{RJ} \zeta_{h\eta} \right] \lambda_n \, d\xi \, d\eta, \\
\int_\Omega \frac{\alpha}{J} L_h n_h \mu_n \, d\xi \, d\eta &= \int_\Omega \left[\frac{\alpha}{J^2} (\beta n_h - \gamma m_h) \zeta_h + \frac{\alpha}{RJ} \zeta_{h\xi} \right] \mu_n \, d\xi \, d\eta,
\end{aligned} \tag{4.1}$$

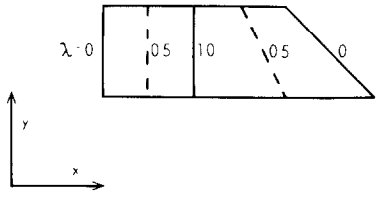

FIG. 2. Contours of trial function for ξ fluxes.

where

$$\int_\Omega \zeta_h \chi_n J \, d\xi \, d\eta = \int \frac{1}{J} |\alpha n_h \chi_{n\xi} - \gamma m_h \chi_{n\eta} - \beta(n_h \chi_{n\eta} - m_h \chi_{n\xi})| - \oint \chi_n \frac{\partial \psi_h}{\partial n} \, ds, \tag{4.2}$$

$$\begin{aligned} P_h m_h &= m_h + \delta_\xi q, \\ P_h n_h &= n_h + \delta_\eta q, \end{aligned} \tag{4.3}$$

$$\sum_{i=1}^{4} P_h \mathbf{m}_h \cdot \mathbf{d\hat{s}} = 0. \tag{4.4}$$

In (4.1), α, β and γ are the components of the metric tensor of the bilinear coordinate transformation defined by

$$\alpha = x_\eta^2 + y_\eta^2, \qquad \beta = x_\xi x_\eta + y_\xi y_\eta, \qquad \gamma = x_\xi^2 + y_\xi^2.$$

In (4.2), the test functions χ_n are the piecewise bilinear trial functions used in previous sections. The boundary integral term vanishes on no-slip boundaries. In (4.3), δ_ξ and δ_η are finite difference operators giving the gradients of q normal to element sides as defined in [17]. In (4.4), the summation is over the four sides of each individual element.

This is not the standard Galerkin algorithm, so we shall check the consistency of both L_h and P_h. The analysis depends, as usual, on the assumption made about how the data is represented. It is found that we must choose a projection r_h as follows:

$$\int_\Omega \frac{1}{J} |\alpha r_h n \chi_{n\xi} - \gamma r_h m \chi_{n\eta} - \beta(r_h n \chi_{n\eta} - r_h m \chi_{n\xi})| \, d\xi \, d\eta - \int_{\partial\Omega} \chi_n r_h \mathbf{m} \times \mathbf{ds} = \int_\Omega \zeta J \, d\xi \, d\eta \qquad \forall n, \tag{4.5}$$

$$\int_\Omega J(r_h p - p) \theta_n \, d\xi \, d\eta = 0 \qquad \forall n, \tag{4.6}$$

$$\sum_{i=1}^{4} r_h \mathbf{m}_i \cdot \mathbf{d\hat{s}}_i = 0 \qquad \text{around the perimeter of each element.} \tag{4.7}$$

In (4.5), ζ is the vorticity derived from the exact velocity field. This equation therefore states that the discrete vorticity ζ_h derived from $(r_h m, r_h n)$ using (4.2) is the best least-squares fit to ζ by piecewise bilinear functions. Therefore

$$\|\zeta_h - \zeta\|_{L_2} = O(h^2). \tag{4.8}$$

The definition (4.5), (4.7) of $r_h m$ and $r_h n$ can be shown to make sense, provided the

original problem was properly posed, since it is a definition of a velocity field from its rotational and divergent parts.

Because of (4.7), the flux components $(r_h m, r_h n)$ can be written as $(-\partial\psi_h/\partial\eta, \partial\psi_h/\partial\xi)$, where ψ_h is a discrete streamfunction. Substituting derivatives of ψ_h into (4.2) shows that the discrete vorticity ζ_h is derived from ψ_h by the standard Galerkin approximation to

$$\nabla^2\psi = -\zeta \tag{4.9}$$

using bilinear elements. We therefore have

$$\|\psi_h - \psi\|_{L_2} = O(h^2), \tag{4.10}$$

where the constant in the estimate involves only properties of the Poisson equation on Ω. This means, by direct differentiation, that

$$\|\mathbf{m}_h - \mathbf{m}\|_{L_2} = O(h) \tag{4.11}$$

and that therefore every term on the right-hand side of (4.1) is within $O(h)$ of its true value. Thus the truncation error

$$r_h L - L_h r_h = O(h), \tag{4.12}$$

where the constant in the estimate does not increase for large R.

We must now estimate $(r_h P - P_h r_h)\mathbf{u}$, for a general divergent field $\mathbf{u}$. Since the algorithm (4.3), (4.4) for P_h is a finite difference scheme, the difference operators δ_ξ, δ_η are constructed to be first-order accurate, and the approximation to the divergence operator in (4.4) is also at least first-order accurate, then the finite difference truncation error

$$s_h P - P_h s_h = O(h), \tag{4.13}$$

where s_h is a grid-point collocation projection such that $s_h\psi$ interpolates ψ at the nodes. This means that the discrete mass fluxes across element sides are exactly the integrals of the true mass fluxes. Because of (4.11), we have

$$\|r_h\mathbf{m} - s_h\mathbf{m}\|_{L_2} = O(h) \tag{4.14}$$

so that

$$\begin{aligned}(r_h P - P_h r_h)\mathbf{u} = (r_h - s_h)\,P\mathbf{u} + (s_h P - P_h s_h)\mathbf{u} \\ + P_h(s_h - r_h)\mathbf{u} = O(h).\end{aligned} \tag{4.15}$$

Thus the algorithm (4.1) to (4.4) is consistent. However, the proof requires a different representation of the data for P_h and L_h and thus the error includes two contributions of the form (4.14). These contributions are not locally determined, since they depend on the estimate (4.10). The results of [17] suggest that the scheme does converge to

the right answer, but is not very accurate on coarse grids. This is consistent with the difficulty in obtaining the error estimate.

The second algorithm of [17] cannot be naturally split into the form $P_h L_h$. However, it contains an approximation to P_h of the form

$$\int_\Omega \frac{1}{J} [\alpha(P_h n_h - n_h)\chi_{n\xi} - \gamma(P_h m_h - m_h)\chi_{nn} - \beta|(P_h n_h - n_h)\chi_{nn} - (P_h m - m_h)\chi_{n\xi}|]\, d\xi\, d\eta - \int_{\partial\Omega} \chi_n (P_h \mathbf{m}_h - \mathbf{m}_h) \times \mathbf{ds} = 0 \qquad \forall n, \quad (4.16)$$

$$\sum_{i=1}^{4} P_h \mathbf{m}_h \cdot \mathbf{d\hat{s}}_i = 0 \qquad \text{around the perimeter of each element.} \quad (4.17)$$

Inspection shows that (4.16) is exactly of the same form as (4.5) and so P_h preserves the discrete vorticity. Therefore, the truncation error (T.E.) contribution $(r_h P - P_h r_h)$ is zero; because both the rotational and divergent parts of the T.E. are zero.

The full algorithm can be written as follows, setting

$$\begin{aligned} Q_h = (PL)_h: \int_\Omega \frac{1}{J} [\alpha Q_h n_h \chi_{n\xi} - \gamma Q_h m_h \chi_{nn} - \beta(Q_h n_h \chi_{nn} - Q_h m_h \chi_{n\xi})]\, d\xi\, d\eta \\ = \int_\Omega \frac{(\alpha\gamma - \beta^2)}{J^2} \zeta_h [m_h \chi_{n\xi} + n_h \chi_{nn}] \\ + \frac{1}{RJ} [\zeta_{h\eta}(\alpha\chi_{n\xi} - \beta\chi_{nn}) - \zeta_{h\xi}(\gamma\chi_{nn} - \alpha\chi_{n\xi})]\, d\xi\, d\eta \qquad \forall n, \end{aligned} \quad (4.18)$$

where n runs over all interior nodes. The treatment of boundaries is discussed in [17]. On rigid boundaries the normal flux is specified and there is no need for an equation in (4.18) associated with boundary nodes. The algorithm is completed by using (4.2) to define ζ_h and (4.4) for the incompressibility condition. The pressure is recovered separately as discussed in [17].

The T.E. estimate proceeds as follows. The algorithm can be considered as first calculating components of $L\mathbf{u}$ as follows:

$$\begin{aligned} \xi \text{ component: } \frac{1}{J}(\alpha n_h + \beta m_h)\zeta_h + \frac{1}{R}\zeta_{h\eta}, \\ \eta \text{ component: } \frac{1}{J}(\beta n_h - \gamma m_h)\zeta_h + \frac{1}{R}\zeta_{h\xi}, \end{aligned} \quad (4.19)$$

and then projecting them, using (4.16) and (4.17). The preceding arguments giving the estimates (4.10) and (4.11) show that the components (4.19) are within $O(h)$ of

their true values. Thus, since there is no truncation error associated with the projection, we can expect that this estimate will still hold for the total truncation error. An attempt to prove this properly is outside the scope of this paper. It would require a regularity property of the exact projection P. The results of [17] suggest that this algorithm is more accurate than the hybrid finite difference scheme. This is probably due to the exact preservation of the discrete vorticity by (4.16), (4.17).

5. Fourier Analysis

It has been possible to explain most of the results of [17] and of Gresho by various truncation error estimates. However, all these estimates contain constants, many of which depend on global properties of the Poisson or Stokes equations in Ω. In particular, the estimate (4.10) of the error in the discrete streamfunction is globally dependent. The results of [17] showed that, on a stretched rectangular mesh, there were substantial errors in the results which were insensitive to the choice of algorithms used. All the algorithms used (4.4) to represent the incompressibility constraint. This suggests that the error associated with enforcing (4.4) may be significant, even though it is a locally consistent approximation to (3.2). To study it further, we state the results of a Fourier analysis of this constraint. Consider a rectangular domain $\Omega: 0 \leqslant x \leqslant L_x$, $0 \leqslant y \leqslant L_y$. Calculate the effect of enforcing (4.2) to (4.4) on the trial velocity field

$$\begin{aligned} u &= \tilde{u} \sin(kx) \cos(ly), \\ v &= \tilde{v} \cos(kx) \sin(ly), \end{aligned} \tag{5.1}$$

where $kL_x = m\pi$, $lL_y = n\pi$. This field satisfies the conditions $u = 0$ on $x = 0$, L_x and $v = 0$ on $y = 0$, L_y. Assume that Ω is subdivided into rectangular elements with dimensions Δx, Δy. Write $\kappa = k\,\Delta x$, $\lambda = l\,\Delta y$. Then the discrete incompressibility condition (4.4) on an approximate field with nodal values

$$\begin{aligned} &\tilde{u}_h \sin kx \cos ly, \\ &\tilde{v}_h \cos kx \sin ly \end{aligned} \tag{5.2}$$

requires

$$k\tilde{u}_h \frac{\sin \frac{1}{2}\kappa}{\kappa} + l\tilde{v}_h \frac{\sin \frac{1}{2}\lambda}{\lambda} = 0, \tag{5.3}$$

whereas the exact condition on (3.1) is

$$k\tilde{u} + l\tilde{v} = 0. \tag{5.4}$$

A lengthy calculation shows that satisfaction of (5.3) given (5.4) and the preservation of the vorticity requires

$$\tilde{u}_h = \frac{l\alpha(\kappa)\beta(\lambda)\alpha(\lambda)(l\tilde{u} - k\tilde{v})}{(l^2\alpha(\lambda) + k^2\alpha(\kappa))},$$
$$\tilde{v}_h = -\frac{k\alpha(\lambda)\beta(\kappa)(l\tilde{u} - k\tilde{v})}{(l^2\alpha(\lambda) + k^2\alpha(\kappa))}, \tag{5.5}$$

where

$$\alpha(\kappa) = \frac{12\sin^2(\frac{1}{2}\kappa)}{\kappa^2(2 + \cos(\kappa))},$$
$$\beta(\kappa) = 2\sin(\tfrac{1}{2}\kappa)/\kappa. \tag{5.6}$$

It is immediately seen that $\tilde{u}_h$ depends on $\tilde{v}$ and $\tilde{v}_h$ on $\tilde{u}$ and that, though the error in (5.5) can be shown to be $O(h^2)$, it has a very complex dependence on k and l. The error could only be reduced by replacing (4.4) by a more accurate approximation to $\nabla \cdot \mathbf{u} = 0$.

6. Discussion

This analysis shows how the results of both Gresho and the companion paper [17] can be understood by using a more flexible approach. This is achieved, as in [10], by carrying out truncation error estimates with a variety of different assumptions about how the data is represented. However, this brings the disadvantage that it is very hard to predict the performance of different algorithms in advance, since it may not be clear what assumption to use in the analysis. The results of [17] suggested that, in regular geometry, the accuracy depended more on the mesh design than on the choices of algorithm used in that paper. All the algorithms used there approximated the continuity equation by (4.4). This suggests that the error in the velocity field resulting from enforcing (4.4) may be an important contribution, even if the vorticity is preserved when it is enforced, as shown by the calculation in Section 5. In irregular geometry, the accuracy of the approximation to the projection operator P appears to be very important. The best results of Gresho and [17] are given by schemes which reduce the truncation error associated with it. This suggests that vorticity-based methods may have advantages.

It is clear that a great deal of further work is required to understand the relative merits of different methods for these equations. This discussion illustrates that error analysis cannot be a replacement for practical tests.

ACKNOWLEDGMENTS

This work was carried out while the author was visiting Dr. P. M. Gresho's group at the Lawrence Livermore Laboratory and the Department of Mathematics, University of California, both of whom provided much helpful discussion.

REFERENCES

1. R. TEMAM, "Navier–Stokes Equations," North-Holland, Amsterdam, 1977.
2. J. G. HEYWOOD AND R. RANNACHER, *SIAM J. Numer. Anal.* **19** (1982), 275–311.
3. P. M. GRESHO, R. L. LEE, AND C. D. UPSON, *Adv. Water Resources* **4** (1981), 175–184.
4. T. J. R. HUGHES, W. K. LIU, AND A. BROOKS, *J. Comput. Phys.* **30** (1979), 1–30.
5. S. T. CHAN AND P. M. GRESHO, *in* "Proceedings, 4th International Symposium on Finite Elements in Flow Problems," Tokyo, 1982.
6. G. STRANG AND G. J. FIX, "An Analysis of the Finite Element Method," Prentice–Hall, Englewood Cliffs, N. J., 1973.
7. J. W. BARRETT AND K. W. MORTON, *Internat. J. Numer. Methods Engrg.* **15** (1980), 1457–1474.
8. J. W. BARRETT AND K. W. MORTON, *I. M. A. J. Numer. Anal.* **1** (1981), 439–468.
9. T. J. R. HUGHES AND A. BROOKS, *in* "Finite Elements in Fluids" (R. H. Gallagher, Ed.), Vol. 4, pp. 47–66, Wiley, New York, 1982.
10. M. J. P. CULLEN AND K. W. MORTON, *J. Comput. Phys.* **34** (1980), 245–267.
11. M. J. P. CULLEN, *J. Comput. Phys.* **45** (1982), 221–245.
12. P. M. GRESHO AND R. L. LEE, *in* "Finite Element Methods for Convection Dominated Flows" (T. J. R. Hughes, Ed.), AMD Vol. 34, pp. 37–61, Am. Soc. Mech. Eng., 1979.
13. A. BROOKS AND T. J. R. HUGHES, *Comput. Methods Appl. Mech. Engrg.* **32** (1982), 199.
14. A. J. CHORIN AND J. E. MARSDEN, "A Mathematical Introduction to Fluid Mechanics," Springer-Verlag, New York, 1979.
15. M. S. ENGELMAN, R. L. SANI, AND P. M. GRESHO, *Internat. J. Numer. Methods Fluids* **2** (1981), 225–238.
16. R. L. SANI, P. M. GRESHO, R. L. LEE, AND D. F. GRIFFITHS, *Internat. J. Numer. Methods Fluids* **1** (1981), 17–45, and **2** (1981), 171–205.
17. M. J. P. CULLEN, *J. Comput. Phys.* **51** (1983), 291–312.
18. A. J. CHORIN, *Math. Comp.* **23** (1968), 341–354.
19. A. DICARLO AND R. PIVA, *Comput & Fluids* **8** (1980), 225–241.
20. C. JOHNSON AND J. PITKARANTA, *Math. Comp.* **38** (1982), 375–400.