SIAM J. SCI. COMPUT.
Vol. 17, No. 2, pp. 358-369, March 1996

004

A NUMERICAL METHOD FOR THE INCOMPRESSIBLE NAVIER-STOKES EQUATIONS BASED ON AN APPROXIMATE PROJECTION*

ANN S. ALMGREN†, JOHN B. BELL†, AND WILLIAM G. SZYMCZAK‡

Abstract. In this method we present a fractional step discretization of the time-dependent incompressible Navier-Stokes equations. The method is based on a projection formulation in which we first solve diffusion-convection equations to predict intermediate velocities, which are then projected onto the space of divergence-free vector fields. Our treatment of the diffusion-convection step uses a specialized second-order upwind method for differencing the nonlinear convective terms that provides a robust treatment of these terms at a high Reynolds number. In contrast to conventional projection-type discretizations that impose a discrete form of the divergence-free constraint, we only approximately impose the constraint; i.e., the velocity field we compute is not exactly divergence-free. The approximate projection is computed using a conventional discretization of the Laplacian and the resulting linear system is solved using conventional multigrid methods. Numerical examples are presented to validate the second-order convergence of the method for Euler, finite Reynolds number, and Stokes flow. A second example illustrating the behavior of the algorithm on an unstable shear layer is also presented.

Key words. incompressible flow, projection methods

AMS subject classifications. 65-C20, 76-D05

1. Introduction.

In this paper we develop a projection method for the incompressible Navier-Stokes equations

$$U_t + (U \cdot \nabla)U = \varepsilon \Delta U - \nabla p + F, \tag{1.1}$$

$$\nabla \cdot U = 0, \tag{1.2}$$

which is formally second-order accurate. Here U represents the velocity field, p represents the hydrodynamic pressure, and F represents any external forces. We denote the x and y components of velocity by u and v, respectively. The method presented here is based on the algorithms presented by Bell, Colella, and Glaz [1] and Bell, Colella, and Howell [2]. As in those papers, we use a second-order upwind method for the treatment of the nonlinear convective terms in (1.1). The algorithms presented in those papers were motivated by a desire to apply higher-order upwind methods developed for inviscid, compressible flow to the incompressible Navier-Stokes equations. In particular, they used a specialized version of the unsplit second-order Godunov methodology introduced for gas dynamics by Colella [9]. The upwind methodology provides a robust discretization of the convective terms that avoids any cell-Reynolds-number stability restriction for high Reynolds-number flow.

The focus of this paper is on the discretization of the projection. The original projection method developed by Chorin [6] defines the projection by defining discrete operators D and G, approximating divergence and gradient, which are skew adjoint; i.e.,

$$D = -G^T.$$

With this definition the discrete projection

$$\mathbf{P} = I - G(DG)^{-1}D$$

*Received by the editors February 8, 1993; accepted for publication (in revised form) March 29, 1995.

†The work of these authors was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under contract W-7405-Eng-48. Support under contract W-7405-Eng-48 was provided by the Applied Mathematical Sciences Program of the Office of Energy Research and by the Defense Nuclear Agency under IACRO 92-825 and IACRO 93-817. Lawrence Livermore National Laboratory, Livermore, CA 94550 (almgren@.llnl.gov, gbb@llnl.gov).

‡Naval Surface Warfare Center, Silver Spring, MD 20903 (wszymcz@nswc-wo.navy.mil).

(with boundary conditions implicit in the boundary conditions of the flow problem) is a discrete orthogonal projection on the finite-dimensional space of vector fields defined on the mesh. In Chorin's formulation both pressure and velocity are specified at nodes and central differences are used for the definition of D and G. This results in an expanded five-point stencil for the discrete Laplacian, DG, that must be inverted to apply the projection. This expanded stencil produces a local decoupling of the mesh points with a 2^d-dimensional kernel for G where d is the dimension of the problem. Bell, Colella, and Glaz [1] use a discretization of the projection based on a finite element method due to Fortin [10] that uses pressure defined on cell centers with velocities given at nodes. This approach produces a more compact stencil but also generates a local decoupling of the grid and dim ker $G > 1$. Bell, Colella, and Howell [2] use a fully cell-centered analogue of Chorin's algorithm. This scheme exhibits a local decoupling, but in the presence of Dirichlet boundary conditions the cell-centered approximation eliminates the nonconstant elements in ker G.

Although the projections discussed above have been shown to be effective for solution of the incompressible Navier-Stokes equations, they exhibit a number of shortcomings as a result of the local grid decoupling. First, the schemes all use nonstandard discretizations of DG that require specialized iterative procedures that properly respect the stencil that is used. (See [2, 13] for a discussion of such a procedure.) For the schemes in which dim ker $G > 1$, the nonconstant elements in the kernel induce additional, artificial compatibility constraints similar to the physical condition

$$\int_{\partial\Omega} U \cdot \mathbf{n}\, ds = 0$$

on the boundary of the physical domain Ω. Local decoupling raises additional difficulties when more complex low Mach number processes are being modeled. Often in these cases a term that results from the additional physics, and is not in the range of the divergence operator, is added to the right-hand side of the Poisson equation for pressure. The presence of the additional source term can result in marked oscillations in the solution. Lai [16, 15] reports such oscillations when using projections with locally decoupled stencils for modeling low Mach number combustion. The approximate projection introduced here has successfully been used for this type of simulation; see [19].

The goal of the work reported here is to develop a projection that avoids the difficulties associated with locally decoupled stencils and is amenable to treatment by standard iterative methods. Several approaches have been proposed that address this issue. In a MAC discretization (cf. Harlow and Welch [11]), a discrete form of (1.2) is enforced in a context in which DG is the standard five-point finite difference Laplacian. However, in the MAC approach the components of velocity are not colocated, which makes the application of modern higher-order upwind techniques extremely cumbersome if not impossible. Strikwerda [20] proposes a regularized projection based on Chorin's original node-based algorithm. Strikwerda's method uses a skewed, higher-order third derivative perturbation to D and G to remove the local decoupling of Chorin's method. Colella [7] reports that although Strikwerda's algorithm is effective for homogeneous boundary conditions, noticeable anomalies arise from the asymmetry of the operators when inflow and outflow profiles are specified. Although the overall algorithm can be engineered around this shortcoming, the complexity of the method is considerably enhanced. In addition, the Strikwerda projection also generates a wide stencil requiring care in the treatment of boundary conditions.

We would like to develop a form of the projection compatible with a cell-centered discretization of velocity that avoids any local decoupling of the stencils, provides a symmetric discretization of the potential flow inherent in nonhomogeneous boundary conditions, and generates a linear system that fits the framework of conventional fast iterative techniques (e.g.,

multigrid) for second-order elliptic equations. The evidence to date suggests that this is not possible while simultaneously requiring that there is a discrete divergence D so that $DU = 0$ is satisfied. In this paper we will relax that condition and only require that $DU = O(h^2)$, where h is the mesh spacing. The approach will treat velocities as averages over grid cells and discretize pressure using a standard bilinear finite element discretization. We will show that relaxing the treatment of (1.2) does not have any deleterious effects on the stability or convergence properties of the algorithm; the advantages of the higher-order upwind methodology are retained and the complexity of the linear algebraic aspects of the method is substantially reduced.

In the next two sections we will review the basic fractional step scheme of Bell, Colella, and Glaz [1] and describe the discretization of the convection–diffusion step of the algorithm. The fourth section describes the approximation of the projection and briefly describes a multigrid algorithm for its solution. This section also contains a discussion of boundary conditions for the projection. In the fifth section we present computational results obtained with the method. The first sequence of results demonstrates second-order convergence of the algorithm for finite Reynolds number, Euler, and Stokes flows. The final example illustrates the application of the method to the computation of an unstable shear layer. The sixth section contains the conclusions.

2. Fractional step scheme. In this section we review the basic fractional step scheme used in the algorithm. In the presentation of the basic algorithm we will focus on homogeneous Dirichlet boundary conditions and no external forces. We will also assume that the mesh spacing on the base grid is uniform in the x- and y-directions, with $\Delta x = \Delta y = h$. These restrictions are not inherent limitations of the method; they have been adopted here for clarity of exposition. The reader is referred to [1, 4] for a more detailed description.

Our strategy for solving the system (1.1)–(1.2) is a fractional step scheme that has two parts: first we solve the advection–diffusion equations (1.1) without strictly enforcing the incompressibility constraint. Then, we project an intermediate vector field onto the space of discretely divergence-free vector fields.

For the diffusion–convection step we solve

$$\frac{U^* - U^n}{\Delta t} + [(U \cdot \nabla)U]^{n+\frac{1}{2}} = \varepsilon \Delta^h \left(\frac{U^n + U^*}{2} \right) - \nabla p^{n-\frac{1}{2}} \tag{2.1}$$

for the intermediate velocity U^*, where Δ^h is a second-order finite difference approximation to Δ. The pressure gradient is evaluated at $t^{n-1/2}$ and is treated as a source term in (2.1). (The pressure gradient is only computed at the 1/2-time levels.) The advection terms in (2.1), namely $[(U \cdot \nabla)U]^{n+1/2}$, are approximated at time $t^{n+1/2}$ to second-order in space and time using an explicit predictor–corrector scheme. This scheme uses only the available data at t^n; thus, the implicit part of (2.1) corresponds to two decoupled parabolic equation solves.

The velocity field U^* computed in the first step is not, in general, divergence-free. The projection step of the algorithm decomposes the result of the first step into a discrete gradient of a scalar potential and a discretely divergence-free vector field that correspond, respectively, to the new approximation to the pressure gradient and an update for the velocity. In particular, if $\mathbf{P}$ represents the composite grid projection then

$$\frac{U^{n+1} - U^n}{\Delta t} = \mathbf{P}\left(\frac{U^* - U^n}{\Delta t} \right), \tag{2.2}$$

$$\nabla p^{n+\frac{1}{2}} = \nabla p^{n-\frac{1}{2}} + (\mathbf{I} - \mathbf{P})\left(\frac{U^* - U^n}{\Delta t} \right).$$

Doesn't this mean that divergences can accumulate?

(Note that the vector field we project is not U^*; it is an approximation of U_t.)

3. Discretization of the diffusion-convection step. In this section we describe the algorithm for the diffusion-convection step in the fractional step scheme, namely (2.1). The algorithm is essentially the same as the second-order upwind method used by Bell, Colella, and Glaz [1]. Here, U_{ij}^n represents the value of the velocity field in cell B_{ij} at time t^n, and $Gp_{ij}^{n-1/2}$ represents the average value of ∇p in cell B_{ij} at time $t^{n-1/2}$. We note that although for the purposes of the projection the pressure is a bilinear function on each cell, we do not use that structure in the advection step. The algorithm is a predictor-corrector method. In the predictor, characteristic equations are used to extrapolate velocities to cell edges at the new half-time level $t^{n+1/2}$. In the corrector, the predicted values are used to compute a flux, which is then differenced to compute the advection terms. These two steps are described in more detail below.

Predictor. In the predictor we extrapolate the velocity and density to the cell edges at $t^{n+1/2}$ using a second-order Taylor series expansion. For edge $(i+\frac{1}{2}, j)$ this gives

$$U_{i+\frac{1}{2},j}^{n+\frac{1}{2},L} = U_{ij}^n + \frac{\Delta x}{2} U_{x,ij}^n + \frac{\Delta t}{2} U_{t,ij}^n$$

extrapolating from (i, j), and

$$U_{i+\frac{1}{2},j}^{n+\frac{1}{2},R} = U_{i+1,j}^n - \frac{\Delta x}{2} U_{x,i+1,j}^n + \frac{\Delta t}{2} U_{t,i+1,j}^n$$

extrapolating from $(i+1, j)$, with analogous formulae for the other edges. The differential equation (1.1) is then used to eliminate the time derivatives to obtain

$$U_{i+\frac{1}{2},j}^{n+\frac{1}{2},L} = U_{ij}^n + \left(\frac{\Delta x}{2} - \frac{u_{ij}\Delta t}{2}\right) U_{x,ij}^n - \frac{\Delta t}{2}(\widehat{vU_y})_{ij} + \frac{\Delta t}{2}\left(\varepsilon \Delta^h U_{ij}^n - Gp_{ij}^{n-\frac{1}{2}}\right), \tag{3.1}$$

$$\begin{aligned} U_{i+\frac{1}{2},j}^{n+\frac{1}{2},R} = {} & U_{i+1,j}^n - \left(\frac{\Delta x}{2} + \frac{u_{i+1,j}\Delta t}{2}\right) U_{x,i+1,j}^n \\ & - \frac{\Delta t}{2}(\widehat{vU_y})_{i+1,j} + \frac{\Delta t}{2}\left(\varepsilon \Delta^h U_{i+1,j}^n - Gp_{i+1,j}^{n-\frac{1}{2}}\right). \end{aligned} \tag{3.2}$$

Here, Δ^h is the standard five-point finite difference approximation to the Laplacian. Equations (3.1)-(3.2) represent the final form of the predictor. Analogous formulae are used to predict values at each of the other edges of the cell. In evaluating these terms the first-order derivatives normal to the edge (in this case U_x) are evaluated using a monotonicity-limited fourth-order centered-difference slope approximation [8]. The limiting is done on the components of the velocity individually.

The transverse derivative terms ($\widehat{vU_y}$ in this case) are evaluated by first extrapolating from above and below to construct edge states, using normal derivatives only, and then choosing between these states using the upwinding procedure defined below. In particular, we define

$$\widehat{U}_{i,j+\frac{1}{2}}^B = U_{ij}^n + \left(\frac{\Delta y}{2} - \frac{v_{ij}\Delta t}{2}\right) U_{y,ij},$$

$$\widehat{U}_{i,j+\frac{1}{2}}^T = U_{i,j+1}^n - \left(\frac{\Delta y}{2} + \frac{v_{i,j+1}\Delta t}{2}\right) U_{y,i,j+1},$$

where U_y are limited slopes in the y-direction, with similar formulae for the lower edge of B_{ij}. Using the upwinding procedure we first define the normal advective velocity on the edge:

$$\widehat{v}_{adv} = \begin{cases} \widehat{v}^B & \text{if } \widehat{v}^B > 0, \quad \widehat{v}^B + \widehat{v}^T > 0, \\ 0 & \text{if } \widehat{v}^B \le 0, \quad \widehat{v}^T \ge 0, \\ \widehat{v}^T & \text{otherwise.} \end{cases}$$

(We suppress the $i, j + \frac{1}{2}$ spatial indices on the bottom and top states here and in the next equation.) We now upwind U based on $\widehat{v}_{adv}$:

$$\widehat{U}_{i,j+\frac{1}{2}} = \begin{cases} \widehat{U}^B & \text{if } \widehat{v}_{adv} > 0, \\ \frac{1}{2}(\widehat{U}^B + \widehat{U}^T) & \text{if } \widehat{v}_{adv} = 0, \\ \widehat{U}^T & \text{if } \widehat{v}_{adv} < 0. \end{cases}$$

After constructing $\widehat{U}_{i,j-1/2}$ in a similar manner, we use these upwind values to form an approximation to the transverse derivative in (3.2):

$$\widehat{vU_y} \approx \frac{1}{2}(\widehat{v}_{i,j+\frac{1}{2}} + \widehat{v}_{i,j-\frac{1}{2}})(\widehat{U}_{i,j+\frac{1}{2}} - \widehat{U}_{i,j-\frac{1}{2}}).$$

Corrector. In the corrector we use the identical upwinding procedure to that used for the transverse derivatives to choose the appropriate states $U_{i+1/2,j}$ given the left and right states $U^{n+1/2,L}_{i+1/2,j}$ and $U^{n+1/2,R}_{i+1/2,j}$. We follow a similar procedure to construct $U_{i-1/2,j}$, $U_{i,j+1/2}$, and $U_{i,j-1/2}$. Finally, we use these values to form an approximation of the convective derivatives in (1.1)

$$uU_x + vU_y \approx \frac{1}{2}(u_{i+\frac{1}{2},j} + u_{i-\frac{1}{2},j})(U_{i+\frac{1}{2},j} - U_{i-\frac{1}{2},j}) + \frac{1}{2}(v_{i,j+\frac{1}{2}} + v_{i,j-\frac{1}{2}})(U_{i,j+\frac{1}{2}} - U_{i,j-\frac{1}{2}}).$$

The Godunov method is an explicit difference scheme and, as such, requires a time-step restriction. A linear, constant-coefficient analysis shows that for stability we must require

$$\max_{ij}\left(\frac{|u_{ij}|\Delta t}{\Delta x}, \frac{|v_{ij}|\Delta t}{\Delta y}\right) = \sigma \le 1,$$

where σ is the CFL number. The time-step restriction of the Godunov method is used to set the time step for the overall algorithm.

A weak nonlinear instability for $\sigma > 0.5$ has been observed [2]; it is believed to result from the use of the lagged pressure gradient in the predictor-corrector algorithm as described above. An alternate form of the corrector incorporates a MAC projection as in [2] to ensure that the edge velocities satisfy

$$D^{MAC}U = \frac{u_{i+1/2,j} - u_{i-1/2,j}}{\Delta x} + \frac{v_{i,j+1/2} - v_{i,j-1/2}}{\Delta y} = 0.$$

This projection is applied immediately before the construction of the convective derivatives and eliminates the instability for $.5 < \sigma \le 1$. Convergence results will be presented for calculations with and without the MAC projection.

Parabolic approximation. Once the advection terms are evaluated at every cell B_{ij}, to complete the solution of (2.1) we must solve the diffusion part of the system with the pressure and advective terms treated as source terms. For this step we use a standard five-point discretization of the Laplacian with a modification at the boundary that computes a second-order approximation of Laplacian that reflects the given boundary data. The resulting system for each velocity component is solved using multigrid.

4. Discretization of the projection. In this section we describe the numerical approximation of the projection. The projection is based on a finite element formulation. In particular, we consider the scalar pressure field to be a C^0 function that is a bilinear function over each cell; i.e., the pressure is in

$$S^h = M_0^1(x) \otimes M_0^1(y),$$

where $M_s^t(x)$ is the space of polynomials of degree t in the x-direction on each cell with C^s continuity at x-edges. For the velocity space we define

$$\mathbf{V}^h = \mathbf{V}^{h,x} \times \mathbf{V}^{h,y},$$

where $\mathbf{V}^{h,x} = M_{-1}^0(x) \otimes M_{-1}^1(y)$ and $\mathbf{V}^{h,y} = M_{-1}^1(x) \otimes M_{-1}^0(y)$; i.e., u is piecewise constant in x and a discontinuous linear function of y in each cell, with a similar form for v.

As noted above, our basic approximation represents U in terms of cell averages. The velocity space $\mathbf{V}^h$ contains additional functions that represent the linear variation within each cell. These additional degrees of freedom make $\mathbf{V}^h$ large enough to contain $\nabla\phi$ for $\phi \in S^h$. We establish a correspondence between these two representations by introducing an orthogonal decomposition of $\mathbf{V}^h$. In particular, for each $V \in \mathbf{V}^h$ we define a piecewise constant component

$$\overline{V}_{ij} = \frac{1}{Vol\ B_{ij}} \int_{B_{ij}} V\, d\mathbf{x}$$

and the variation

$$V^{\perp} = V - \overline{V}$$

so that for each cell B_{ij}, $\int_{B_{ij}} V^{\perp} d\mathbf{x} = 0$. By construction these two components are orthogonal in L^2, so they can be used to define a decomposition of $\mathbf{V}^h$ into two components

$$\mathbf{V}^h = \overline{\mathbf{V}}^h \oplus \mathbf{V}^{h^{\perp}}, \tag{4.1}$$

where $\overline{\mathbf{V}}^h$ and $\mathbf{V}^{h^{\perp}}$ represent the cell averages and the orthogonal linear variation, respectively. The decomposition of $\mathbf{V}^h$ induces a decomposition of $\nabla\phi$ for all $\phi \in S^h$; namely,

$$(\nabla\phi)_{ij} = (\overline{\nabla\phi})_{ij} + (\nabla\phi)_{ij}^{\perp},$$

where

$$G\phi_{ij} \equiv (\overline{\nabla\phi})_{ij} = \left(\frac{\phi_{i+1/2,j+1/2} + \phi_{i+1/2,j-1/2} - \phi_{i-1/2,j+1/2} - \phi_{i-1/2,j-1/2}}{2\Delta x}, \right.$$

$$\left. \frac{\phi_{i+1/2,j+1/2} + \phi_{i-1/2,j+1/2} - \phi_{i+1/2,j-1/2} - \phi_{i-1/2,j-1/2}}{2\Delta y} \right),$$

$$(\nabla\phi)_{ij}^{\perp} = (\phi_{i+\frac{1}{2},j+\frac{1}{2}} + \phi_{i-\frac{1}{2},j-\frac{1}{2}} - \phi_{i-\frac{1}{2},j+\frac{1}{2}} - \phi_{i+\frac{1}{2},j-\frac{1}{2}}) \left(\frac{y_{ij}}{\Delta x}, \frac{x_{ij}}{\Delta y} \right),$$

where $\phi_{i+1/2,j+1/2}$ represent the nodal values of ϕ. Here x_{ij} and y_{ij} are local variables, defined on each cell such that $x_{ij} = y_{ij} = 0$ at the center of B_{ij}, $x_{ij} = \pm\frac{1}{2}$ at the left and right edges of B_{ij}, and $y_{ij} = \pm\frac{1}{2}$ at the top and bottom edges of B_{ij}.

We now define a weak form of the projection on $\mathbf{V}^h$, based on a weak divergence on $\mathbf{V}^h$. In particular, we define a vector field V^d in $\mathbf{V}^h$ to be divergence-free if

$$\int_\Omega V^d \cdot \nabla\psi \, d\mathbf{x} = 0 \qquad \forall \psi \in S^h. \tag{4.2}$$

With this definition we can then project any vector field V onto a gradient $\nabla\phi$ and weakly divergence-free field V^d (with vanishing normal velocities on boundaries) by solving

$$\int_\Omega \nabla\phi(\mathbf{x}) \cdot \nabla\psi_{i+\frac{1}{2},j+\frac{1}{2}}(\mathbf{x}) \, d\mathbf{x} = \int_\Omega V \cdot \nabla\psi_{i+\frac{1}{2},j+\frac{1}{2}}(\mathbf{x}) \, d\mathbf{x} \qquad \forall \psi_{i+\frac{1}{2},j+\frac{1}{2}}(\mathbf{x}) \tag{4.3}$$

for

$$\phi(\mathbf{x}) = \sum_{i,j} \phi_{i+\frac{1}{2},j+\frac{1}{2}} \psi_{i+\frac{1}{2},j+\frac{1}{2}}(\mathbf{x})$$

and setting $V^d = V - \nabla\phi$. Here the ψ's are the standard basis functions for S^h; namely, $\psi_{i+1/2,j+1/2}(\mathbf{x})$ is the piecewise bilinear function having node values $\psi_{i+1/2,j+1/2} \cdot (\mathbf{x}_{k+1/2,\ell+1/2}) = \delta_{ik}\delta_{j\ell}$. We note that with a suitable normalization, the right-hand side of (4.3) defines a nodal value of the divergence of V at node $(i + \frac{1}{2}, j + \frac{1}{2})$; in particular, by using (4.3) we are implicitly defining the discrete divergence by

$$(DV)_{i+\frac{1}{2},j+\frac{1}{2}} = -\frac{1}{\overline{\psi}} \int_\Omega V \cdot \nabla\psi_{i+\frac{1}{2},j+\frac{1}{2}}(\mathbf{x}) \, d\mathbf{x}, \tag{4.4}$$

where $\overline{\psi} = \int_\Omega \psi_{i+1/2,j+1/2}(\mathbf{x}) \, d\mathbf{x}$.

When this approximate projection is applied in the context of our fractional step scheme, the vector field V to which the projection is applied is given by

$$V = \overline{V} \equiv \frac{U^* - U^n}{\Delta t}, \quad V^\perp = 0. \tag{4.5}$$

That's where Stefan and I deviate!

(In the predictor we have only determined the quantity $\overline{U}^*$, so we make the implicit assumption here that $U^{\perp,*} = U^{\perp,n}$.) In this case, since D operates on a "barred" vector, the divergence is simply

$$(D\overline{V})_{i+\frac{1}{2},j+\frac{1}{2}} = \frac{V^x{}_{i+1,j} + V^x{}_{i+1,j+1} - V^x{}_{i,j} - V^x{}_{i,j+1}}{2\Delta x} + \frac{V^y{}_{i,j+1} + V^y{}_{i+1,j+1} - V^y{}_{i,j} - V^y{}_{i+1,j}}{2\Delta y},$$

where V^x and V^y are the x- and y-components of V, respectively. With $\overline{V}$ defined by (4.5), $V^x = \frac{u^* - u^n}{\Delta t}$ and $V^y = \frac{v^* - v^n}{\Delta t}$. We solve (4.3) for ϕ to obtain an update for p; i.e.,

$$\phi(\mathbf{x}) = \delta p^n \equiv p^{n+\frac{1}{2}} - p^{n-\frac{1}{2}}.$$

For the purposes of the fractional-step scheme we then define

$$\overline{V}^d = \overline{V} - \overline{G\phi} \tag{4.6}$$

as the approximation of $\frac{\overline{U}^{n+1} - \overline{U}^n}{\Delta t}$ in (2.2).

The vector field $\overline{V}^d$ is only approximately divergence-free; i.e., $D\overline{V}^d \neq 0$ even in the weak sense of (4.2). The quantity that is weakly divergence-free is $V^d = V - \nabla\phi$; however, in general, $DV^{d^\perp} \neq 0$; consequently, $D\overline{V}^d \neq 0$. Effectively, what we have done in writing (4.6) is perform the vector field decomposition into a divergence-free component and the gradient of a scalar on $\mathbf{V}^h$ and project the weakly divergence-free component onto $\overline{\mathbf{V}}^h$ using the characterization of $\mathbf{V}^h$ given by (4.1).

In order to estimate the effect of only approximately enforcing discrete incompressibility we will rewrite the result of the projection as

$$\frac{\overline{U}^* - \overline{U}^n}{\Delta t} = \frac{U^{n+1} - U^n}{\Delta t} + \nabla\delta p^n, \tag{4.7}$$

$$DU^{n+1} = 0,$$

where we implicitly assume $DU^n = 0$, although $D\overline{U}^n \neq 0$. Since $D\overline{U}^n = -DU^{n^\perp}$, we can estimate how errors accumulate by examining the behavior of $U^{n^\perp}$. To accomplish this we break (4.7) into its component pieces:

$$\frac{\overline{U}^* - \overline{U}^n}{\Delta t} = \frac{\overline{U}^{n+1} - \overline{U}^n}{\Delta t} + \frac{U^{\perp,n+1} - U^{\perp,n}}{\Delta t} + \nabla(p^{n+\frac{1}{2}} - p^{n-\frac{1}{2}}).$$

If we then use the characterization of $\mathbf{V}^h$ given by (4.1), we obtain

$$\overline{U}^{n+1} = \overline{U}^n + \Delta t \left(\frac{\overline{U}^* - \overline{U}^n}{\Delta t} + Gp^{n-\frac{1}{2}} - Gp^{n+\frac{1}{2}} \right)$$

and

$$U^{n+1^\perp} = U^{n^\perp} + \Delta t((\nabla p^{n-\frac{1}{2}})^\perp - (\nabla p^{n+\frac{1}{2}})^\perp) \approx U^{n^\perp} - \Delta t^2 (\nabla p_t^n)^\perp.$$

This evolution equation for $U^\perp$ places limits on the growth of $U^\perp$. The quantity $(\nabla p)^\perp$ is uniquely defined by the node values $p_{i+1/2,j+1/2}$. As long as the pressure remains sufficiently smooth in space and time, $U^\perp$ is well behaved as well and scales with $\Delta t\ h = \sigma h^2$. The factor of Δt comes from the evolution equation for $U^\perp$; the factor of h comes from the fact that for smooth ψ, $|(\nabla\psi)^\perp| \leq O(h)$. Thus $DU^n = 0$ guarantees that $D\overline{U}^n = O(h^2)$ as long as the CFL condition is enforced. We monitored $D\overline{U}$ for the convergence study presented in the next section and verified the second-order accuracy computationally.

The linear system associated with the solution of (4.3) is the standard bilinear finite element stiffness matrix for Poisson's equations. We solve this system using standard multigrid methods (see [5]); in particular, we use the standard V-cycle with Jacobi relaxation. This procedure reduces the residuals by approximately a factor of five per V-cycle.

The left-hand side of (4.3) is, in discrete form, a nine-point stencil approximating the Laplacian of ϕ. We note that it is possible, using a similar development, to construct an approximate projection in which the left-hand side of (4.3) is the standard five-point finite difference Laplacian. This construction is based on approximating pressure as a piecewise linear function on triangles. We define a triangulation of the domain by connecting lower-left corners of each grid cell to upper-right corners. With this definition we define $\mathbf{V}^h$ to

TABLE 1
Convergence rates—MAC-less predictor.

Case	16-32	Rate	32-64	Rate	64-128	Rate	128-256
Euler	1.44e-2	2.22	3.08e-3	2.36	5.98e-4	2.17	1.33e-4
Re 100	6.70e-3	2.06	1.61e-3	1.94	4.20e-4	1.90	1.12-4
Stokes	4.95e-3	1.99	1.25e-3	1.99	3.14e-4	1.91	8.39-5

TABLE 2
Convergence rates—predictor using MAC.

Case	16-32	Rate	32-64	Rate	64-128	Rate	128-256
Euler	1.44e-2	2.26	3.00e-3	2.35	5.90e-4	2.18	1.30e-4
Re 100	6.10e-3	2.15	1.37e-3	1.91	3.63e-4	1.87	9.93e-5

be piecewise constant functions on each triangle. As in the bilinear case, we establish the relationship between the projection and the upwind procedure by defining a decomposition of $\mathbf{V}^h$ similar to (4.1); namely,

$$\mathbf{V}^h = \overline{\mathbf{V}}^h \oplus \mathbf{V}^{h^\perp},$$

where $\overline{\mathbf{V}}^h$ is defined as before and $\mathbf{V}^{h^\perp}$ is piecewise constant on each triangle and averages to zero over each of the original rectangular grid cells. This alternate form also leads to an approximate projection in which the vector field decomposition is performed in $\mathbf{V}^h$ and the result is projected onto $\overline{\mathbf{V}}^h$; however, in this case (4.3) for ϕ gives

$$\frac{\phi_{i+3/2,j+1/2} + \phi_{i-1/2,j+1/2} + \phi_{i+1/2,j+3/2} + \phi_{i+1/2,j-1/2} - 4\phi_{i+1/2,j+1/2}}{h^2} = (DV)_{i+\frac{1}{2},j+\frac{1}{2}},$$

where $DV_{i+1/2,j+1/2}$ is defined as in (4.4). We have verified the second-order accuracy of the method with this five-point stencil and observed that the L_2-errors between calculations with different resolution using the five-point stencil differ by less than 2% from the errors shown in Tables 1 and 2.

Boundary conditions for the projection. For homogeneous boundary conditions we want $V^d \cdot \mathbf{n} = 0$ on $\partial\Omega$. This condition is enforced (weakly) by including ψ's that are nonzero on the boundary of Ω in S^h, which imposes a natural treatment of the boundary conditions for the Poisson equation. If we formally integrate (4.3) by parts we see that ϕ satisfies

$$\Delta\phi = DV \quad \text{on } \Omega,$$

$$\frac{\partial\phi}{\partial n} = V \cdot \mathbf{n} \quad \text{on } \partial\Omega.$$

For inflow and outflow, we want to specify $V^d \cdot \mathbf{n}$ on part of the boundary (inflow), which we will refer to as Γ_1, and have outflow on Γ_2. On the outflow face we want to impose the condition that there are no net forces accelerating the fluid parallel to the outflow face. In this simple setting we can accomplish this by setting $\phi = 0$ on Γ_2. Thus, for outflow we restrict S^h to include only those functions that vanish at points on Γ_2.

We do allow ψ's to have support on Γ_1, so we solve for ϕ's on Γ_1, as in the homogeneous boundary condition case; however, we must augment (4.3) with boundary terms to reflect the

specified boundary data. In particular, we add a boundary term to (4.3) to obtain

$$\int_\Omega V \cdot \nabla\psi \, d\mathbf{x} = \int_\Omega \nabla\phi \cdot \nabla\psi \, d\mathbf{x} + \int_{\Gamma_1} \psi V^d \cdot \mathbf{n} \, ds.$$

We substitute the boundary conditions for V^d on Γ_1, subtract this quantity from the left-hand side, and proceed with the projection. As can be seen by formally integrating by parts, we now have

$$\frac{\partial\phi}{\partial n} = V \cdot \mathbf{n} - V^d \cdot \mathbf{n} \quad \text{on } \Gamma_1,$$

which enforces the desired condition. Note that the boundary condition for V^d is $V^d \cdot \mathbf{n} = (u_{in}^{n+1} - u_{in}^n)/\Delta t$, where u_{in} is the time-dependent inflow velocity.

5. Numerical results. In this section we first present numerical results to demonstrate the second-order convergence of the method presented here. We also apply the method to computation of a forced, unstable shear layer to illustrate the capabilities of the method in a more realistic setting.

For the convergence study we use the same initial data as was used in [1]. We initialize the data with a smooth velocity field

$$u(x, y) = -\sin^2(\pi x)\sin(2\pi y),$$

$$v(x, y) = \sin^2(\pi y)\sin(2\pi x),$$

in the unit square. The velocity satisfies homogeneous Dirichlet boundary conditions.

Calculations were run to $t = 0.5$ with $\Delta x = \Delta y = 1/2^n$ for $n = 4, 5, 6, 7, 8$. The time step was restricted by the stability criterion to be $\Delta t = .5\Delta x$, which approximates CFL $= 0.5$. To estimate the convergence rate, we find the L_2-norm of the difference between the velocity obtained on each grid and the velocity obtained on the next finer grid; then take $\log_2$ of the ratio of these norms. The convergence rates for velocity are given in Table 1 for the MAC-less predictor, and in Table 2 for the predictor that uses the MAC projection. (We also present, in Table 1, results for Stokes flow.)

The second set of calculations is a direct comparison with the experimental study of Ho and Huang [12] on the nonlinear instability of a forced shear layer. The forcing is introduced using an upstream perturbation based on the stability analysis of Monkewitz and Huerre [17]. We note that a number of other authors have also treated this problem; see, for example, Kuhl et al. [14], who include a survey of the other literature.

Our computations were performed on a 512×128 uniform grid, with $\varepsilon = 0$. In Figures 1 and 2 we present contours of vorticity for two cases corresponding to the fundamental perturbation frequency (ω_1) plus a subharmonic. The cases correspond to the first (ω_2) and second (ω_3) subharmonics, respectively. The largest perturbation amplitude was 1% for the fundamental with reduced amplitudes for the subharmonics as given in [17]. The numerical results show the instability of the shear layer and the downstream vortex merging pattern. The first case consists of a simple pairing; in the second case the vortices merge in groups of three with the latter two vortices merging first. The prediction of the correct merging patterns and the near periodicity of the results serve to validate the numerical method on a realistic flow problem.

We also compute the flow perturbed by the fundamental plus nine subharmonics. Computational results for this case are shown in Figure 3. In this case we see a substantially more complex pattern of merger, analogous to the broadly forced instability of this shear layer

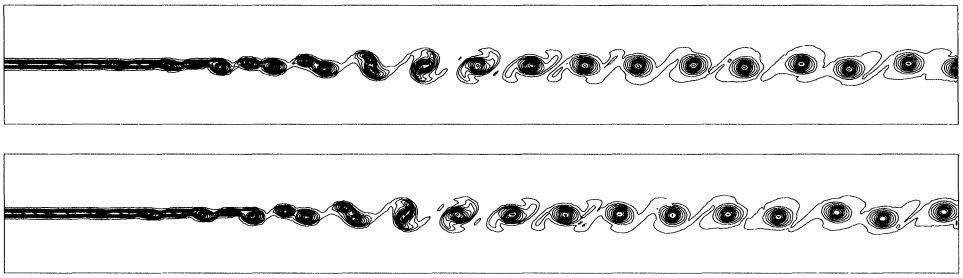

FIG. 1. $\omega_1 + \omega_2$ *forcing at times* T = 1333, 1369.

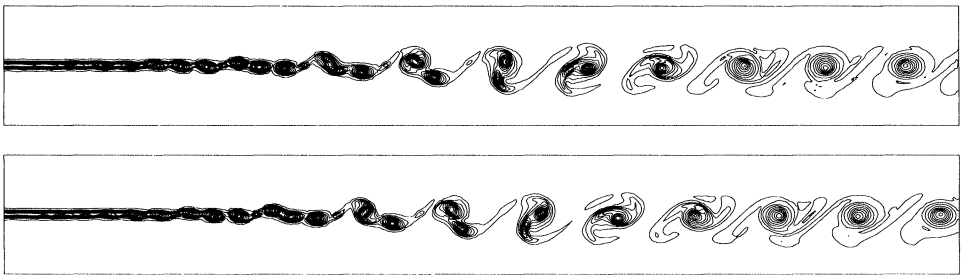

FIG. 2. $\omega_1 + \omega_3$ *forcing at times* T = 1319, 1354.

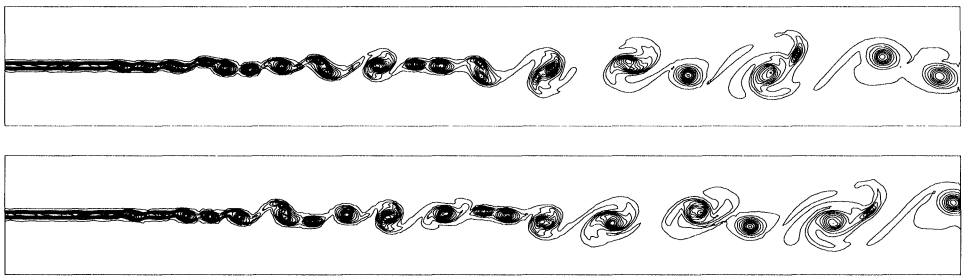

FIG. 3. *Ten frequency forcing forcing at times* T = 1281, 1351.

investigated by Oster and Wygnanski [18]. Although detailed comparisons for this case are beyond the scope of this paper, we note that the spreading rate in the computation agrees well with the experimental value of 11 degrees. Also note that we see no difficulties at inflow and outflow boundaries for either this case or the preceding cases.

6. Conclusions. Traditionally, projection methods for discretizing the time-dependent incompressible Navier-Stokes equations have exactly applied a discrete form of the projection; i.e., the velocity field returned from the projection satisfies a discrete constraint $DU = 0$ for some discrete divergence operator D. In this paper we have shown that this requirement can be relaxed. We have defined a projection based on a finite element construction that only enforces the divergence-free condition to second-order accuracy. The overall accuracy of the method is essentially no different than previous versions of the methodology that exactly imposed a discrete form of the constraint. By relaxing this requirement we have been able to define

a projection for which the linear system that must be solved corresponds to a conventional discretization of the Laplacian but for which the velocities are cell-centered, enabling the use of modern upwind techniques for treating the advective derivatives.

This simplified projection is easily generalized to three dimensions and to variable density flows (see [3]). This generalization will be discussed in a forthcoming paper. We are also using this approach to develop an adaptive mesh version of the projection method. In this context we can exploit the similarity with standard finite element techniques in order to use iterative methods developed in the elliptic community for adaptivity.

REFERENCES

[1] J. B. Bell, P. Colella, and H. M. Glaz, *A second-order projection method for viscous, incompressible flow*, in Proc. 8th AIAA Computational Fluid Dynamics Conference, Honolulu, HI, June 9-11, 1987.

[2] J. B. Bell, P. Colella, and L. H. Howell, *An efficient second-order projection method for viscous incompressible flow*, in Proc. 10th AIAA Computational Fluid Dynamics Conference, Honolulu, HI, June 24-27, 1991.

[3] J. B. Bell and D. L. Marcus, *A second-order projection method for variable-density flows*, J. Comput. Phys., 101 (1992), pp. 334-348.

[4] J. B. Bell, J. M. Solomon, and W. G. Szymczak, *A second-order projection method for the incompressible Navier–Stokes equations on quadrilateral grids*, in Proc. 9th AIAA Computational Fluids Dynamics Conference, Buffalo, NY, June 14-16, 1989.

[5] W. L. Briggs, *A Multigrid Tutorial*, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1987.

[6] A. J. Chorin, *Numerical solution of the Navier–Stokes equations*, J. Math. Comput., 22 (1968), pp. 745-762.

[7] P. Colella, *Private communcation*, 1992.

[8] ———, *A direct Eulerian MUSCL scheme for gas dynamics*, SIAM J. Comput., 6 (1985), pp. 104-117.

[9] ———, *A multidimensional second order Godunov scheme for conservation laws*, J. Comput. Phys., 87 (1990), pp. 171-200.

[10] M. Fortin, *Numerical solutions of the steady state Navier–Stokes equations*, in Numerical Methods in Fluid Dynamics, AGARD-LS-48, North Atlantic Treaty Org., Advisory Group for Aerospace Research & Development, Lecture Series 48, J. J. Smolderen, ed., Neuilly-sur-Seine, France, 1972.

[11] F. H. Harlow and J. E. Welch, *Numerical calculation of time-dependent viscous incompressible flow of fluids with free surfaces*, Phys. Fluids, 8 (1965), pp. 2182-2189.

[12] C. M. Ho and L. S. Huang, *Subharmonics and vortex merging in mixing layers*, J. Fluid Mech., 119 (1982), pp. 443-473.

[13] L. H. Howell and J. B. Bell, *An adaptive-mesh projection method for viscous incompressible flow*, SIAM J. Sci. Comput., to appear.

[14] A. L. Kuhl, K.-Y. Chien, R. E. Ferguson, H. M. Glaz, and P. Colella, *Inviscid Dynamics of Unstable Shear Layers*, Report RDA-TR-161604-004, R&D Associates, Marina del Rey, CA, April 1988.

[15] M. Lai, J. B. Bell, and P. Colella, *A projection method for combustion in the zero mach number limit*, in Proc. 11th AIAA Computational Fluid Dynamics Conference, Orlando, FL, July 6-9, 1993, pp. 776-783.

[16] M. F. Lai, *A Projection Method for Reacting Flow in the Zero Mach Number Limit*, Ph.D. thesis, University of California, Berkeley, CA, 1993.

[17] P. A. Monkewitz and P. Huerre, *Influence of the velocity ratio on the spatial instability of mixing layers*, Phys. Fluids, 25 (1982), pp. 1137-1143.

[18] D. Oster and I. Wygnanski, *The forced mixing layer between parallel streams*, J. Fluid Mech., 123 (1982), pp. 91-130.

[19] R. B. Pember, A. S. Almgren, J. B. Bell, P. Colella, L. H. Howell, and M. Lai, *A higher-order projection method for the simulation of unsteady turbulent nonpremixed combustion in an industrial burner*, in Proc. 8th International Symposium on Transport Phenomena in Combustion, San Francisco, CA, July 16-20, 1995.

[20] J. Strikwerda, *Finite difference methods for the Stokes and Navier–Stokes equations*, SIAM J. Sci. Statist. Comput., 5 (1984), pp. 56-67.