JOURNAL OF COMPUTATIONAL PHYSICS 82, 298–329 (1989)

The Numerical Solution of the Navier–Stokes Equations for 3-Dimensional, Unsteady, Incompressible Flows by Compact Schemes

T. B. GATSKI

NASA-Langley Research Center, Hampton, Virginia 23665–5225

C. E. GROSCH*

Old Dominion University, Norfolk, Virginia 23508–8508 and Institute for Computer Application in Science and Engineering, Hampton, Virginia 23665–5225

AND

M. E. ROSE†

MER Associates, Greensboro, North Carolina 27410

Received September 4, 1987; revised August 2, 1988

This paper describes a numerical method for solving the Navier–Stokes equations for unsteady, incompressible, 3-dimensional flows using velocity–vorticity variables and irregular Cartesian grids. The method involves solving Cauchy–Riemann type equations for the velocity and transport-diffusion equations for the vorticity whose solenoidal vorticity components are obtained by solving a Poisson equation for a suitably chosen scalar potential. The importance of boundary conditions is examined. The difference equations are solved by iterations in order to permit exploitation of parallel and vector computing methods. Numerical experiments confirm the second-order spatial and temporal accuracy of the method.

I. INTRODUCTION

During the past two decades, major advances have been reported about numerical methods for solving the Navier–Stokes (N–S) equations for incompressible flows using techniques based on finite-difference methods (FDM) [1], finite-element methods (FEM) [2, 3], boundary element methods (BEM) [4, 5], spectral

* Work supported under NASA Grant NAG1-530 and NASA Contract NAS1-18107.

† Work supported under NASA Contract NAS1-18240 to High Technology Corporation and under ONR Contract N000 14-83-K-0422 while the author was a Visiting Associate at the California Institute of Technology.

0021-9991/89 $3.00

methods [6], or combinations of these [7]. In addition, grid free vortex methods [8] have also produced revealing results.

This paper treats the velocity–vorticity formulation of the N–S equations for unsteady, 3-dimensional flows. In a previous paper [9], we reported on a finite difference solution of this problem in two dimensions. There, the transport of vorticity was governed by a simple advection–diffusion equation, and the only non-zero component of vorticity was normal to the plane of motion (velocity field). The governing equation set was closed by using the incompressibility condition and the kinematic definition of the vorticity. (These two equations thus constitute a form of the Cauchy–Riemann equations.) In this paper we extend this approach to 3-dimensional problems. As might be expected, additional complicating factors arise.

The most obvious of these is the appearance of the vortex stretching term in the vorticity transport equation. This term serves as the redistribution mechanism between the three vorticity components. Another complicating factor, although more subtle, is the requirement that the vorticity field which appears in the discrete Cauchy–Riemann equations be solenoidal (in some discrete sense). This, of course, is not an issue for the differential equations since it is a consequence of the fact that the vorticity is the curl of the velocity.

The idea of using a velocity–vorticity formulation itself is not new; however, the method of implementation presented in this paper and that presented previously [9] is unique in that we treat the Cauchy–Riemann equations directly as a system of first-order partial differential equations. In previous studies [10–13], the velocity field was obtained from Poisson equations for each of the velocity components. These were derived by taking the curl of the velocity and using the condition that the velocity is divergence free to eliminate the mixed derivatives. These results were for steady [10] and unsteady [11–13] flows in both two [11–13] and three dimensions [10]. The approach of solving the Cauchy–Riemann equations for the velocity field was also used in [14]. The method of solution employed matrix-inversion techniques rather than the iterative methods used in the present study. Finally, in a recent application of the boundary element method [15], it was indicated that a "vorticity–velocity" approach was utilized; however, a vector potential was introduced and the resulting Poisson equations were solved.

In the present study, the velocity field is obtained from the solution of the Cauchy–Riemann equations using a discrete scheme described in [16] in which the prescribed solenoidal vorticity field is first obtained by a discrete version of the Helmholtz decomposition theorem. The discrete approximation scheme for the vorticity transport equation is based upon methods used in [9], but modified so as to include the stretching term. These points are highlighted in the text and the details of the construction are presented in the Appendix.

In Section II the differential formulation of the problem is presented and an overall iterative solution strategy is developed. This enables us to identify four underlying problems, the numerical solutions of which enable us to treat the global problem itself. Section III describes the numerical treatment of three of these

problems using an iteration method proposed by Kaczmarz [17] (see also [18]). The 3-dimensional problem formulation is then completed by a discussion of boundary condition specifications (Section V). All numerical methods for the incompressible Navier–Stokes equations which require the calculation of boundary conditions for some of the variables in terms of computed values of the other variables can suffer a loss of accuracy at the boundary. Such methods include those based on the velocity–vorticity formulation as well as the velocity–pressure formulation. These issues, in the context of our method, are discussed in detail in Section V and numerical results for test problems designed to shed light on these issues are further discussed in Section VI. The two test problems of Section VI are steady stagnation flow [19] and unsteady vortex spinup [20]. The numerical results appear to validate our conclusion that the method yields second-order spatial and temporal accuracy.

The numerical schemes used in this paper are based on a domain decomposition method in which the relationship between data and solution values in each cell are appropriately expressed by algebraic relations and are extended throughout the underlying domain by continuity requirements. We call this a compact scheme: it can be interpreted as a special type of (non-conforming) finite-element method. A useful property of such schemes is that they can be expressed by using simple finite-difference notations whenever uniform or non-uniform Cartesian grids are used, as are the cases treated here. In such cases, the consistency of the schemes with the differential equations, when they are written as a first-order system, is evident. Although we plan to treat problems which involve non-rectangular elements elsewhere, the method for treating both Cartesian and general elements is presented in the Appendix to this paper. Emphasis has been placed on using element-by-element iterative solution techniques which can apply to the general problem and which facilitate either vector or parallel computation. However, no attempt has been made to seek the type of efficiency which a production code would require.

II. Formulation and Solution Strategy

The Navier–Stokes equations describing the flow of a Newtonian incompressible fluid can be written as

$$\nabla \cdot \mathbf{u} = 0 \tag{2.1a}$$

$$\nabla \times \mathbf{u} = \zeta \tag{2.1b}$$

$$\zeta_t + (\mathbf{u} \cdot \nabla)\zeta - (\zeta \cdot \nabla)\mathbf{u} = \nu \nabla^2 \zeta, \tag{2.1c}$$

where $\mathbf{u} = \mathbf{u}(\mathbf{x}, t)$ and $\zeta = \zeta(\mathbf{x}, t)$ are the velocity and vorticity fields, respectively, $\mathbf{x} = (x^1, x^2, x^3)$ is a point in $\mathbf{R}^3$, t is the time, and ν is the kinematic viscosity. Equation (2.1a) expresses the incompressibility condition, while (2.1b) identifies ζ as the vorticity. The vorticity transport equation, Eq. (2.1c), results from taking the

curl of the momentum equations. In addition, as a consequence of Eq. (2.1b), the vorticity field must also be solenoidal,

$$\nabla \cdot \zeta = 0. \tag{2.1d}$$

We seek a solution in a domain D which satisfies the initial conditions

$$\mathbf{u} = \mathbf{u}_0, \qquad \zeta = \nabla \times \mathbf{u}_0, \qquad \text{at } t = 0 \tag{2.2a), (2.2b}$$

and the boundary conditions

$$\mathbf{u} = \mathbf{u}_\Gamma, \qquad \zeta = (\nabla \times \mathbf{u})\,|_\Gamma \tag{2.3a), (2.3b}$$

on the boundary Γ of D. The task is to develop a second-order spatially and temporally accurate numerical algorithm for the differential system described in Eqs. (2.1) through (2.3). We assume the solution exists and is unique for $0 < t < T$. Problems with more general boundary conditions are discussed in Section V.

First note that the equations have the semigroup property, i.e., if the solution at $t = t_1$ can be obtained from the initial conditions at $t = 0$, then the solution at a later time $t = T$ can be obtained using the solution at $t = t_1$ as initial data. Since it is assumed that the solution exists for $0 < t \leqslant T$, it is only necessary to indicate a construction for an arbitrary subinterval.

We shall next consider the possibility of constructing a solution of (2.1)–(2.3) in any time interval by an iteration process each step of which will only require the solution of a linear problem. In Section III we will then examine how each of these problems can, in turn, be solved numerically.

SOLUTION METHOD. Consider the following iterative method for constructing a solution in any fixed time interval: Let l indicate an iteration index, $l = 0, 1, 2, \ldots$ and suppose that $\hat{\zeta}^l$ is a solenoidal vector which is given at the start of the lth step.

P(i) Since $\hat{\zeta}^l$ is solenoidal, $\nabla \cdot \hat{\zeta}^l = 0$. The Cauchy–Riemann system, Eqs. (2.1a) and (2.1b), with $\zeta = \hat{\zeta}^l$, can be solved with the boundary conditions $\mathbf{n} \cdot (\mathbf{u} - \mathbf{u}_\Gamma) = 0$ on Γ, where $\mathbf{n}$ denotes a unit exterior normal vector. Call this solution $\mathbf{u}^l$.

P(ii) Referring to Fig. 1, let D_ε indicate a small neighborhood of the boundary Γ. Construct $\mathbf{u}_\varepsilon^l$ such that $\mathbf{u}_\varepsilon^l = \mathbf{u}^l$ in $D - D_\varepsilon$ and is smoothly interpolated in D_ε so as to satisfy the boundary condition $\mathbf{u}_\varepsilon^l = \mathbf{u}_\Gamma$ on Γ. (Recall that $\mathbf{n} \cdot \mathbf{u} = \mathbf{n} \cdot \mathbf{u}_\Gamma$ was the only boundary condition enforced on $\mathbf{u}^l$ in step **P**(i).) Using Eq. (2.3b) construct the vorticity boundary condition

$$\zeta_\Gamma^l = \lim_{\varepsilon \to 0} (\nabla \times \mathbf{u}_\varepsilon^l)\,|_\Gamma. \tag{2.4}$$

P(iii) Solve

$$\zeta_t + (\mathbf{u}^l \cdot \nabla)\zeta - (\zeta \cdot \nabla)\mathbf{u}^l = \nu \nabla^2 \zeta \tag{2.5a}$$

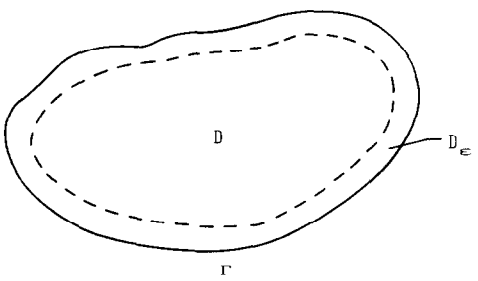


FIG. 1. Simply connected domain D with boundary Γ and boundary strip D_ϵ.

with the initial condition

$$\zeta = \nabla \times \mathbf{u}_0, \qquad t = 0 \tag{2.5b}$$

and boundary conditions Eq. (2.4). Since $\mathbf{u}^l$ was determined in step **P**(i), this is a linear transport-diffusion equation for ζ. Call the solution ζ^l.

P(iv) The solution ζ^l from step **P**(iii) need not be solenoidal, since the coefficient terms in (2.5a) involve a velocity field which, in step **P**(i), was determined by means of a solenoidal vector $\hat{\xi}^l$ which, so far, is unrelated to ξ^l. However, the Helmholtz theorem allows us to project ζ^l into a solenoidal component and an irrotational component. We call this solenoidal component the Helmholtz projection of ζ^l and denote it by $\hat{\zeta}^l$.

P(v) Setting $l \leftarrow l+1$, repeat the procedure. Assuming this describes a convergent process, denote the converged solution by $\mathbf{u}$ and ζ.

It is not difficult to conclude that the solution pair $(\mathbf{u}, \zeta)$ formally satisfies the velocity–vorticity form of the Navier–Stokes equations given by Eq. (2.1) as well as the initial conditions Eq. (2.2). Also, $\hat{\zeta} = \zeta$. However, it is more difficult to conclude that the boundary conditions (2.3) will be satisfied as well. By construction, the normal components of $\mathbf{u}^l|_\Gamma$ and $\mathbf{u}_\Gamma$ agree; moreover, from the definition of vorticity, Eq. (2.1b), it is evident that the normal component of $\zeta^l|_\Gamma$ is determined by the tangential components of $\mathbf{u}_\Gamma$ without appeal to the extrapolation involving $\mathbf{u}_\epsilon^l$ expressed in Eq. (2.4). Thus, Eq. (2.4) determines the tangential components of $\zeta|_\Gamma$ in terms of $\mathbf{u}_\Gamma$ and the field $\mathbf{u}_\epsilon^l$.

This outline indicates that, for a convergent process, the governing differential equations given in Eq. (2.1) will be satisfied as well as the initial conditions given in Eq. (2.2). In addition, the construction explicitly enforces the condition on the normal component of $\mathbf{u}_\Gamma$ and implicitly enforces, through the conditions on ζ_Γ, the tangential components of $\mathbf{u}_\Gamma$. Numerical evidence indicates that this construction does yield a solution pair $(\mathbf{u}, \zeta)$ which also satisfies the boundary conditions (2.3). Nevertheless, it can be anticipated that the treatment of boundary conditions will be a sensitive issue; this topic will be discussed in Section V. The next sections will

describe numerical schemes for solving each of the problems identified by **P**(i)–**P**(iv).

III. Numerical Schemes

In this section, the numerical schemes solving **P**(i), **P**(iii), **P**(iv) will be described. As presented in the description of the solution method, Eqs. (2.1a, 2.1b) are to be solved simultaneously as a Cauchy–Riemann-type system. In the remainder of the paper we refer to these as the velocity equations, since the system determines the velocity field in terms of a given solenoidal field. Obviously, Eq. (2.5a) corresponds to the vorticity transport equation (2.1c) and will be referred to as such, although, as indicated earlier, only its solenoidal component appears in the velocity equations.

The numerical methods used to develop the solution of these component problems are described in detail in the Appendix and only the resulting schemes will be presented in this section. These schemes arise by partitioning the fundamental domain D into volume elements, indicated by $\{e\}$, in each of which elementary approximate solutions of the differential equations are used to establish relationships between solution values on the faces of the element in the time interval. When Cartesian grids are employed, certain symmetries of the element allow conventional finite-difference notations to be used. In this case, the discrete problem can be formulated in a manner which makes its structural consistency with the Cartesian form of the differential equations obvious.

A. *Velocity Equations*

Given a solenoidal vector field $\hat{\boldsymbol{\zeta}}$, the problem arising in **P**(i) is to solve

$$\operatorname{div} \mathbf{u} = 0 \tag{3.1a}$$

$$\operatorname{curl} \mathbf{u} = \hat{\boldsymbol{\zeta}} \tag{3.1b}$$

in D with

$$\mathbf{n} \cdot \mathbf{u} = \mathbf{n} \cdot \mathbf{u}_\Gamma. \tag{3.2}$$

Consider a volume element e shown in Fig. 2. By integrating $\operatorname{div} \mathbf{u} = 0$ over e and the normal component of $\hat{\boldsymbol{\zeta}}$ over the faces $\gamma \in \partial e$, one obtains

$$\operatorname{div}_e \mathbf{u} = 0 \tag{3.3a}$$

$$\mathbf{n} \cdot \operatorname{curl}_\gamma \mathbf{u} = \mathbf{n} \cdot \hat{\boldsymbol{\zeta}}(\gamma). \tag{3.3b}$$

$e \in D$ in which div_e and $\operatorname{curl}_\gamma$ arise from the differential operators div and curl by using central divided difference operators in place of the differential operator.

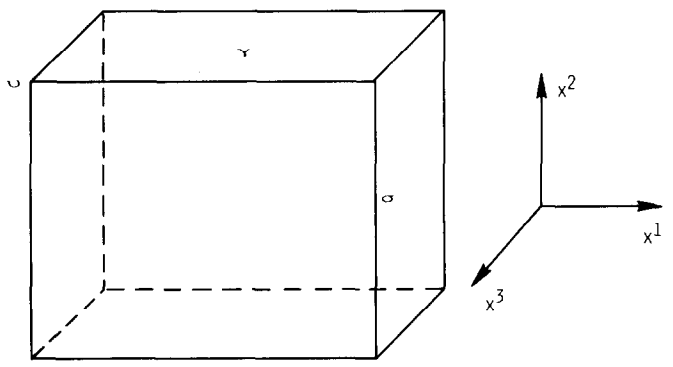


FIG. 2. Cartesian element e, with faces γ, edges σ, and vertices υ.

Writing $\mathbf{u} = (u_x, u_y, u_z)$, the discrete approximations to these equations are then given by (see Appendix for details)

$$\delta_x(\mu_y\mu_z u_x) + \delta_y(\mu_x\mu_z u_y) + \delta_z(\mu_x\mu_y u_z) = 0 \tag{3.4a}$$

$$\delta_y(\mu_z u_z) - \delta_z(\mu_y u_y) = \hat{\zeta}_x \tag{3.4b}$$

$$\delta_z(\mu_x u_x) - \delta_x(\mu_z u_z) = \hat{\zeta}_y \tag{3.4c}$$

$$\delta_x(\mu_y u_y) - \delta_y(\mu_x u_x) = \hat{\zeta}_z, \tag{3.4d}$$

where μ and δ are the respective centered average and divided difference operators. The variables u involved in these expressions are associated with the vertices of the volume element and are called box variables. Equations (3.4) constitute the velocity equations which are to be solved subject to the boundary conditions, Eq. (3.2). In Fix and Rose [16], a least squares formulation of Eq. (3.3) was considered and Eq. (3.4) was shown to provide a second-order accurate solution of the velocity Eqs. (3.1) and (3.2) provided that the prescribed field $\hat{\zeta}$ satisfies the discrete solenoidal condition $\mathrm{div}_e\,\hat{\zeta} = 0$ in each element e.

B. *Transport Equations*

Given a velocity field $\mathbf{u}^I$ satisfying Eqs. (3.1), the equations (2.5) governing the time evolution of the vorticity, ζ, are

$$\zeta_t + (\mathbf{u}^I \cdot \nabla)\zeta - (\zeta \cdot \nabla)\mathbf{u}^I = \nu\nabla^2\zeta \tag{3.5a}$$

with initial conditions

$$\zeta_0 = \nabla \times \mathbf{u}_0^I, \tag{3.5b}$$

and boundary conditions

$$\zeta_\Gamma = (\nabla \times \mathbf{u}^I)\,|_\Gamma. \tag{3.5c}$$

Concerning a time interval $|t - t_m| < \Delta t/2$ in which $t_m = m \cdot \Delta t/2$, $m = 1, 3, \ldots$, we interpret the initial conditions (3.5b) as those which have been determined at time $t_m - \Delta t/2$ from the previous time interval. Step **P**(iii) requires us to solve (3.5a) when, at each stage of the iteration cycle, a velocity field $\mathbf{u}^l$ has been determined by solving the velocity Eqs. (3.1). In this case (3.5) is a linear vector differential equation with variable coefficients.

A numerical scheme for solving these equations is described in the Appendix. It involves freezing the coefficients in any volume element and then constructing a solution in the volume element which approximately satisfies the initial condition from the previous time level and satisfies boundary conditions which match values obtained either from neighboring elements or from prescribed boundary conditions when the element adjoins the boundary Γ. In each volume element, the simple transformation of the vorticity variable

$$\zeta = \exp[\mathbf{b}(t - t_m)]\boldsymbol{\omega}, \tag{3.6}$$

where

$$\mathbf{b} = \nabla(\mathbf{u} + \mathbf{u}^{\mathrm{T}})^l/2 \tag{3.7}$$

leads to a simple advection–diffusion transport equation for the transformed variable $\boldsymbol{\omega}$, i.e.,

$$\boldsymbol{\omega}_t + (\mathbf{u}^l \cdot \nabla)\boldsymbol{\omega} = \nu \nabla^2 \boldsymbol{\omega}. \tag{3.8}$$

Note that at $t = t_m$, ζ and $\boldsymbol{\omega}$ are equal; therefore the specification of boundary conditions for ζ at $t = t_m$ also specifies $\boldsymbol{\omega}$. Also, even though the components of $\boldsymbol{\omega}$ are uncoupled in the transport Eqs. (3.8), the coupling among the components of the vorticity ζ is retained from one time interval to the next through the application of Eq. (3.6). These points will be highlighted further as the numerical scheme and solver for Eq. (3.8) are presented.

The difference scheme to solve (3.8) is most simply understood by writing (3.8) as a first-order system. Let

$$\boldsymbol{\Phi} = (\boldsymbol{\phi}_x, \boldsymbol{\phi}_y, \boldsymbol{\phi}_z);$$

then (3.8) can be written

$$\boldsymbol{\omega}_t = \frac{\partial \boldsymbol{\phi}_x}{\partial x} + \frac{\partial \boldsymbol{\phi}_y}{\partial y} + \frac{\partial \boldsymbol{\phi}_z}{\partial z} \tag{3.9a}$$

$$\boldsymbol{\Phi} = \nu \operatorname{grad} \boldsymbol{\omega} - \mathbf{u}^l : \boldsymbol{\omega}. \tag{3.9b}$$

Direct application and extension of the results in the Appendix (see Eqs. (B.43)), leads to the following compact scheme for the 3-dimensional advection–diffusion equations for $\boldsymbol{\omega}$:

$$\delta_t \omega = \delta_x \phi_x + \delta_y \phi_y + \delta_z \phi_z \tag{3.10a}$$

$$\mu_t \omega = c_\alpha \mu_\alpha \omega - \rho_\alpha \Delta_\alpha \omega - \sigma_\alpha \Delta_\alpha \phi_\alpha \tag{3.10b}$$

$$\mu_\alpha \phi_\alpha - \rho_\alpha \Delta_\alpha \phi_\alpha = \kappa_\alpha \lambda_\alpha^{-1} \Delta_\alpha \omega - \mu_\alpha \mathbf{u}_\alpha^l (c_\alpha \mu_\alpha \omega - \rho_\alpha \Delta_\alpha \omega) \tag{3.10c}$$

$\alpha = x, y, z$, and

$$c_\alpha = 1 - \Delta_\alpha u_\alpha^l (\mu_\alpha u_\alpha^l)^{-1} \rho_\alpha, \qquad \sigma_\alpha = \lambda_\alpha \kappa_\alpha^{-1} (\theta_\alpha^{-1} \rho_\alpha) \tag{3.11}$$

$$\theta_\alpha = (\mu_\alpha u_\alpha^l)\, h_\alpha \nu^{-1} \tag{3.12}$$

$$\lambda_\alpha = \tau h_\alpha^{-1} \tag{3.13}$$

$$\kappa_\alpha = \nu \tau h_\alpha^{-2}, \tag{3.14}$$

$h_x = \Delta x/2$, etc.; ρ_α is defined in Eq. (B.43d), and the operator notation is defined in the Appendix. As Eqs. (3.12) through (3.14) indicate, the scheme is parametized by cell Reynolds numbers, θ_α, CFL numbers, $(\mu_\alpha u_\alpha^l)\lambda_\alpha$, and diffusion parameters, κ_α (=CFL number/cell-Reynolds number).

These equations are used to calculate values of ω at time levels $t = t_m$ and $t = t_{m+1/2}$ from data at $t = t_{m-1/2}$ and boundary conditions at $t = t_m$. The values of ζ at corresponding times are obtained from an approximation to the exponential transformation (3.6) which is described below.

We now address the problem of determining the discrete solenoidal component of the numerical field ζ.

C. *Helmholtz Projection*

Consider the Helmholtz decomposition of the vorticity field ζ into a solenoidal part and an irrotational part,

$$\zeta = \hat{\zeta} + \nabla \chi \qquad \text{in } D. \tag{3.15}$$

A Poisson equation for χ is obtained by taking the divergence of Eq. (3.15). Let $\mathbf{p} = \nabla \chi$, then

$$\nabla \cdot \mathbf{p} = \nabla^2 \chi = \nabla \cdot \zeta \qquad \text{in } D \tag{3.16a}$$

which is solved with Neumann boundary conditions

$$\mathbf{n} \cdot \mathbf{p} = 0 \qquad \text{on } \Gamma \tag{3.16b}$$

so that

$$\mathbf{n} \cdot (\hat{\zeta} - \zeta) = 0 \qquad \text{on } \Gamma. \tag{3.17}$$

The discrete Helmholtz projection problem in **P**(iv) may be posed as follows: given

a vector field $\zeta(\gamma)$ defined on the faces of an element, determine the normal components of a vector field $\mathbf{p}(\gamma)$ on each face, such that (cf. Eq. (3.16))

$$\operatorname{div}_e \mathbf{p} = \operatorname{div}_e \zeta, \qquad e \in D \tag{3.18a}$$

$$\mathbf{n} \cdot \mathbf{p}(\gamma_\Gamma) = 0, \qquad \gamma_\Gamma \in \Gamma. \tag{3.18b}$$

in which $\operatorname{div}_e = (\delta_x, \delta_y, \delta_z)$. If

$$\hat{\zeta}(\gamma) = \zeta(\gamma) - \mathbf{p}(\gamma) \tag{3.19a}$$

then $\hat{\zeta}$ is a discrete solenoidal field since

$$\operatorname{div}_e \hat{\zeta} = 0, \qquad e \in D, \tag{3.19b}$$

while

$$\mathbf{n} \cdot (\hat{\zeta}(\gamma_\Gamma) - \zeta(\gamma_\Gamma)) = 0, \qquad \gamma_\Gamma \in \Gamma. \tag{3.19c}$$

A method for determining $\mathbf{p}(\gamma) \cdot \mathbf{n}$ is described in the Appendix. The compact approximation scheme for Eq. (3.16) is given by (cf. Eq. (B.9))

$$\delta_x p_x + \delta_y p_y + \delta_z p_z = \operatorname{div}_e \zeta \tag{3.20a}$$

$$\mu_x p_x = \delta_x \chi, \qquad \mu_y p_y = \delta_y \chi, \qquad \mu_z p_z = \delta_z \chi \qquad \text{(3.20b), (3.20c), (3.20d)}$$

$$\mu_x \chi - h_x^2 \delta_x p_x/2 = \mu_y \chi - h_y^2 \delta_y p_y/2 = \mu_z \chi - h_z^2 \delta_z p_z/2. \qquad \text{(3.20e), (3.20f)}$$

An examination of Eqs. (3.20) reveals that these six equations relate the six average values of χ on the faces γ of an element with the six average values of the normal components of $\mathbf{p}$ on the faces. Thus, the discrete problem for $\nabla^2 \chi = \operatorname{div} \zeta$ on e is solvable; that is, the discrete values $(\partial/\partial n)\, \chi(\gamma)$ can be determined from the discrete values $\chi(\gamma)$, $\gamma \in e$. Note, also, that Eqs. (3.20a)–(3.20d) result directly from a finite volume treatment of the system form of (3.16a), i.e., $\operatorname{div} \mathbf{p} = f$, $\mathbf{p} = \nabla \chi$ and are consistent with the differential equations. Equations (3.20e), (3.20f) provide the additional relationships required to solve the discrete problem on e. As indicated in the Appendix, the solution satisfies a discrete energy estimate and converges with second-order accuracy to the solution of the Poisson equation.

Finally, with a second-order accurate solution pair $(\chi, \mathbf{p})$, the construction of a solenoidal vorticity field $\hat{\zeta}$ can be obtained from

$$\mathbf{n} \cdot \hat{\zeta}(\gamma) = n \cdot (\zeta(\gamma) - \mathbf{p}(\gamma)), \qquad \gamma \in e,\ e \in D \tag{3.21}$$

or, in component form,

$$(\hat{\zeta}_x)_{i \pm 1/2, j, k} = (\zeta_x - p_x)_{i \pm 1/2, j, k} \tag{3.22a}$$

$$(\hat{\zeta}_y)_{i, j \pm 1/2, k} = (\zeta_y - p_y)_{i, j \pm 1/2, k} \tag{3.22b}$$

$$(\zeta_z)_{i, j, k \pm 1/2} = (\zeta_z - p_z)_{i, j, k \pm 1/2}. \tag{3.22c}$$

This concludes our description of the numerical schemes which solve the problems identified in steps **P**(i), **P**(iii), **P**(iv) of the overall iteration method described in Section II. The next section will describe how each of these problems can be solved by a uniform iteration method.

IV. Solution Methods for the Discrete Equations

We now describe methods used in solving the algebraic set of equations for the discrete forms of the velocity equations, the vorticity transport equations, and the Poisson equation. These will be discussed within the context of the five-step iteration procedure presented in Section II.

A. *Kaczmarz Algorithm*

The Kaczmarz method is an iteration method which is used to solve a system of m linear algebraic equations in n unknowns of the form

$$\mathbf{a}_i \mathbf{x}^{\mathrm{T}} = b_i, \qquad i = 1, 2, ..., m, \tag{4.1}$$

where $\mathbf{a}_i = (a_{1i}, a_{2i}, ..., a_{ni})$, $\mathbf{x} = (x_1, x_2, ..., x_n)$, and $\mathbf{a}^{\mathrm{T}}$ is the transpose of $\mathbf{a}$. We may suppose these equations have been normalized by the condition

$$\mathbf{a}_i \cdot \mathbf{a}_i^{\mathrm{T}} = 1. \tag{4.2}$$

Defining

$$r_i(\mathbf{x}) = \mathbf{a}_i \cdot \mathbf{x}^{\mathrm{T}} - b_i, \tag{4.3}$$

the Kaczmarz algorithm solves the least squares problem

$$\sum_i r_i^2(\mathbf{x}) = \min \tag{4.4}$$

as follows: given $\mathbf{x}_k^0$, set

$$\begin{aligned} &\mathbf{x}_{k+1}^{l+1} = \mathbf{x}_k^l - \alpha \mathbf{a}_k r_k(\mathbf{x}_k^l), \qquad k = 1, 2, ..., m, \; l = 0, 1, ... \\ &\mathbf{x}_1^{l+1} = \mathbf{x}_m^l \end{aligned} \tag{4.5}$$

in which α is a relaxation parameter. This scheme converges for $1 \leqslant \alpha < 2$ with an asymptotic convergence rate which is typical of SOR schemes and is independent of the manner in which the equations have been ordered. Thus, the equations in the discrete schemes described in Section III may be solved element-by-element either singly or in groups.

B. *Solution Procedures*

A numerical solution of the discrete Navier–Stokes equations is obtained by implementing the five-step solution procedure described in Section II using the

Kaczmarz method to solve each of the numerical schemes which were set forth in Section III.

Step **P**(i) requires the solution of the discrete Cauchy–Riemann system Eqs. (3.4) for a velocity field **u**. A requisite for such a solution is that the prescribed numerical vorticity field $\hat{\zeta}$ be solenoidal in the sense that $\mathrm{div}_e\,\hat{\zeta}=0$ in each element. These equations are subject to the boundary conditions $\mathbf{n}\cdot(\mathbf{u}-\mathbf{u}_\Gamma)=0$ on Γ.

Step **P**(ii) requires specifying the vorticity boundary conditions on Γ. This involves a method which will be described in the next section. Using the resulting vorticity boundary conditions and the velocity field **u** obtained from step **P**(i), step **P**(iii) involves solving the appropriate vorticity transport equations for ζ.

The discrete advective–diffusion equations (3.10a)–(3.10c) for $\boldsymbol{\omega}$ together with the relationship between $\boldsymbol{\omega}$ and ζ given by Eq. (3.6) enable values $\zeta^{m-1/2}$ to be determined from initial data $\zeta^{m-3/2}$ and boundary data in the time strip $|t-t_{m-1}|\leqslant \Delta t/2$. Since $\zeta^{m-1/2}$ provides initial data for the next time strip $|t-t_m|\leqslant \Delta t/2$, it is possible to relate the values ζ^m directly to ζ^{m-1}. The result, which applies in each element, can be written

$$\exp(-\mathbf{b}^m\tau)\,M_-\,\boldsymbol{\omega}^m=\exp(+\mathbf{b}^{m-1}\tau)\,M_+\,\boldsymbol{\omega}^{m-1}, \tag{4.6}$$

where

$$M_\pm=(\mu_t\pm\tau\delta_t),\qquad \tau=\Delta t/2,$$

to which are adjoined Eq. (3.10c). The operators $M_\pm$ are evaluated using the expressions for $\mu_t\boldsymbol{\omega}$, $\delta_t\boldsymbol{\omega}$ which are given in Eq. (3.10). Also, the approximation $\exp(\mathbf{b}\tau)\simeq 1+\mathbf{b}\tau$ is used, which is consistent with the second-order accuracy of the overall scheme. Equations (4.6) and (3.10c) are solved for $\boldsymbol{\omega}^m$ by the Kaczmarz algorithm using the boundary conditions imposed for $t=t_m$. From Eq. (3.6) we see that $\zeta^m=\boldsymbol{\omega}^m$ at $t=t_m$ so that the solution of Eqs. (4.6) and (3.10c) provides the discrete solution of (2.5a) at the $t=t_m$.

Finally, step **P**(iv) of the iteration cycle determines the discrete solenoidal component $\hat{\zeta}$ of the solution ζ just calculated. This is accomplished by solving the discrete Poisson equation system (3.20) for χ and then using Eq. (3.21) to obtain $\hat{\zeta}$.

Using the discrete values of $\hat{\zeta}$ just obtained, steps **P**(i)–**P**(iv) are repeated until convergence (step **P**(v)). The result, at $t=t_m$, is the solution pair of $(\mathbf{u}^m,\hat{\zeta}^m)$. The process is then repeated in the next time strip, using these values as starting approximations.

It is of interest to point out the ease with which the algorithm, as now constructed for the 3-dimensional case, degenerates to the 2-dimensional case. In the velocity equations, one component of the velocity is lost and two defining equations for the vorticity are dropped; since the Kaczmarz solver treats each equation separately throughout the domain D, it is obviously a simple matter to delete the appropriate component velocity and define the vorticity equations in the code. The situation is similar for treating the vorticity equations although more variables are dropped since gradients normal to the plane of flow are zero in the 2-dimensional

case. The remaining vorticity component is governed by a simple advection–diffusion equation for which no exponential transformation is needed and is discretely solenoidal so that the Helmholtz projection construction can be eliminated.

V. Boundary Conditions

In the original problem for the Navier–Stokes equations which was set forth in Section II, we assumed that the velocity was specified on the boundary of the computational domain (Eq. (2.3)). When, in addition, the normal derivatives of the velocity on the boundary have been determined (viz., by knowing the solution within the domain) then the values of the vorticity on the boundary can be determined, as indicated in Eq. (2.3).

In describing a solution strategy (steps **P**(i)–**P**(v)), we introduced in step **P**(ii) an interpolation process by which the velocity field which was constructred in step **P**(i) could smoothly link its interior values in the domain to values which were prescribed on the boundary by (2.3). As a result, both the normal as well as tangential derivatives of this velocity field were available in order to construct the boundary values of the vorticity (Eq. (2.4)) which are necessary in order to solve the vorticity–transport equation in step **P**(iii).

We can, of course, treat more general boundary conditions. The solution of the Cauchy–Riemann type equations described in step **P**(i) is possible when the value of a tangential component of the velocity is given on part of the boundary of the domain, and the normal component is given elsewhere. We can also replace certain velocity boundary conditions in Eq. (2.3) by vorticity conditions. For example, if the tangential components of the velocity were to be specified everywhere on a boundary, this would be equivalent to specifying the tangential derivatives of the velocity and thus the normal component of the vorticity. From this we may conclude that an appropriate set of three components taken from either velocity or vorticity boundary values also lead to a well-posed problem for the Navier–Stokes equations. This is of practical relevance, because it is often convenient to specify one or more components of the vorticity at inflow and/or outflow. However, not all of the components of ζ_Γ can be prescribed. Indeed, if the vorticity on rigid no-slip boundaries were known a priori then, in many cases, there would be no need to carry out the calculation. This follows from the facts that (1) the boundary shear stress is proportional to the tangential component of the vorticity and (2) in many engineering applications the main objective of the calculation is to find the shear stress on the rigid no-slip boundaries.

In order to apply this algorithm to realistic problems one must derive values of ζ_Γ, the boundary value of the vorticity. This difficulty is not unique to the velocity–vorticity formulation of the incompressible Navier–Stokes equations. If one uses the velocity–pressure formulation it is necessary to derive boundary condi-

tions for the pressure. This is by no means a trivial problem; see, for example, the recent study of pressure boundary conditions by Orszag, Israeli, and Deville [24].

In Section III we described how each of the steps **P**(i), **P**(iii), and **P**(iv) could be implemented numerically. Here we shall describe the interpolation used in step **P**(ii). The Cauchy–Riemann problem requires, as the boundary condition, one and only one component of $\mathbf{u}_\Gamma$ to be specified on the boundary, which, for simplicity, we take to be the normal component. In this case, the tangential components of the resulting solution, **u** will differ from the prescribed tangential components of u_Γ. At a rigid boundary there will be a velocity "slip."

This velocity "slip" generates vorticity at the boundary. The effect of step **P**(ii) is to generate an amount of vorticity at Γ so that the tangential component of **u** is equal to the tangential component of $\mathbf{u}_\Gamma$. This requires that we evaluate the tangential and normal derivatives of **u** on Γ. For a numerical construction the tangential derivatives of $\mathbf{u}_\Gamma$ can be evaluated exactly because the velocity $\mathbf{u}_\Gamma$ is given on Γ. The evaluation of the normal derivatives can be done with the use of an N point one-sided difference formula based on values of **u** interior to D. Similar considerations apply if one specifies a tangential component for the Cauchy–Riemann problem.

To summarize: for **P**(i) one and only one component of $\mathbf{u}_\Gamma$ gives the boundary condition; for **P**(ii) we use an N point extrapolation formula to compute $\partial\mathbf{u}/\partial n$ on Γ and the exact values of the tangential derivatives. Thus we have the boundary condition for **P**(iii). Finally for **P**(iv) we require the normal component of ζ, which is given in terms of the known derivative of the tangential components of **u** on Γ. This completes the specification of the boundary conditions for the method.

It is clear that the boundary conditions for the velocity problem in **P**(i) and the Helmholtz projection in **P**(iv) can be determined exactly from the specification of $\mathbf{u}_\Gamma$. In contrast, the boundary conditions for the vorticity problem requires the use of an N point formula to evaluate $\partial\mathbf{u}/\partial n$ on Γ.

The procedures used in **P**(i), **P**(iii), and **P**(iv) are second-order accurate. Thus we take $N=3$ and use a three point difference formula to evaluate $\partial\mathbf{u}/\partial n$ on Γ with accuracy $O(h^2)$. This is only formally correct. If the velocity, **u**, is accurate to $O(h^2)$, the numerical first derivatives of **u** will only be accurate to, at best, $O(h)$. This implies that the boundary conditions for the vorticity will be only accurate to $O(h)$. Clearly, the use of a higher order formula ($N>3$) to determine $\partial\mathbf{u}/\partial n$ will not improve the accuracy of the boundary conditions for ζ because the velocity field is only accurate to $O(h^2)$.

We conclude that this scheme yields boundary conditions for the vorticity which are $O(h)$. This is not surprising and, in fact, can give a more accurate boundary condition for ζ than is sometimes obtained for the pressure from a "primitive" variable, **u**, P formulation. This is because calculation of either a normal or tangential boundary condition for the pressure gradient requires the evaluation of the viscous terms in the momentum equation which requires the evaluation of second derivatives of **u**. Therefore, if the error in the *computed* value of **u** is $O(h^2)$, the *computed* value of the pressure gradient on Γ may be only accurate to $O(1)$.

The results of numerical experiments, given in the next section, will confirm the

fact that the velocity field $\mathbf{u}$ can be computed with second-order accuracy even though the vorticity boundary conditions are only first-order accurate. Given a desired level of accuracy of the vorticity solution we may choose a grid spacing, h, away from the boundaries with error $\varepsilon = O(h^2)$ and, near the boundaries, use a fine grid, h_B, such that $\varepsilon = O(h_B)$. Thus one can insure a uniform error.

The problem associated with prescribing downstream boundary conditions are well known and a full discussion of these is beyond the scope of this paper. Here we simply remark that in our earlier treatment of 2-dimensional problems we found that a simple advection of vorticity at the downstream boundary proved successful.

VI. Results of Test Problems

In this section we present results of calculations for two test problems; one, a 3-dimensional steady flow, and the other, a 3-dimensional, time-dependent flow. In both cases, exact solutions in closed form of the Navier–Stokes equations for an incompressible fluid are known. These problems thus provide tests of the spatial and temporal accuracy of the algorithm.

The first of these problems is Howarth's [19] solution for the 3-dimensional, steady, stagnation point flow (our notation is slightly different from that of [19]). In the Cartesian coordinate system (x^1, x^2, x^3) with velocity components (u_1, u_2, u_3), the (x^1, x^3) plane is an impenetrable wall upon which $\mathbf{u} \equiv 0$.

In the far field, as $x^2 \to \infty$, we have

$$\mathbf{u} \to U_0[(x^1/L_0)\hat{e}_1 - (1-\beta)(x^2/L_0)\hat{e}_2 + (\beta x^3/L_0)\hat{e}_3], \tag{6.1}$$

with U_0 a velocity scale, L_0 a length scale, $\{\hat{e}_j\}$ unit vectors, and β a dimensionless parameter, $0 \leqslant \beta \leqslant 1$. If $\beta = 0$, we have the 2-dimensional stagnation point flow and if $\beta = 1$ the flow is the axisymmetric stagnation point flow.

We then nondimensionalize with velocity scale U_0, length scale L_0, time scale $T_0 = L_0/U_0$, and pressure scale $P_0 = \rho U_0^2$, and ρ the density. Then define a Reynolds number $R = U_0 L_0/\nu$, with ν a kinematic viscosity, and let $\eta = R^{1/2}x^2$. It is now easy to show that, with the primes denoting differentiation,

$$u_1 = x^1 f'(\eta), \tag{6.2}$$

$$u_2 = -R^{-1/2}[f(\eta) + \beta g(\eta)], \tag{6.3}$$

$$u_3 = \beta x^3 g'(\eta), \tag{6.4}$$

$$P = P_1 - \tfrac{1}{2}[(x^1)^2 + \beta(x^3)^2] - R^{-1}[f'(\eta) + \beta g'(\eta) + \tfrac{1}{2}(f(\eta) + \beta g(\eta))^2] \tag{6.5}$$

with P_1 a constant, is the exact solution of the Navier–Stokes equations which satisfies the boundary conditions. The functions $\{f(\eta), g(\eta)\}$ are the solutions of

$$f''' + ff'' - (f')^2 + \beta g f'' + 1 = 0, \tag{6.6}$$

$$g''' + \beta g g'' - \beta(g')^2 + f g'' + \beta = 0, \tag{6.7}$$

with

$$f(0) = f'(0) = g(0) = g'(0) = 0 \tag{6.8}$$

and

$$f' \to g' \to 1 \qquad \text{as } \eta \to \infty. \tag{6.9}$$

It is trivial to compute the vorticity from the velocity. This solution describes the flow in the vicinity of the stagnation point of a nonaxisymmetric body.

If $\beta \neq (0, 1)$, this is a truly 3-dimensional flow which cannot be transformed to a 2-dimensional flow by a rotation about the x^2 axis because the projection of the velocity vector onto the (x^1, x^3) plane rotates as x^2 is varied from 0 to ∞. In our calculations we have chosen $\beta = \frac{1}{2}$ so that the flow is, in a sense, intermediate between a 2-dimensional and an axisymmetric flow.

The computational domain was held fixed at $0 \leqslant x^1 \leqslant 1$, $0 \leqslant x^3 \leqslant 1$, and $0 \leqslant \eta \leqslant 5$, in dimensionless units. We have solved this problem with grids of $(10)^3$, $(20)^3$, and $(30)^3$ cells using various boundary conditions.

In the first experiment both the velocity and vorticity on the computational boundaries were computed from Eqs. (6.2), (6.3), and (6.4). This gives exact boundary conditions for $\mathbf{u}$ and ζ. In Table I we list the values of the L_2 norms of the velocity, $\|\mathbf{u}\|$, and vorticity $\|\zeta\|$, the L_2 norms of the errors $\|E(\mathbf{u})\|$ and $\|E(\zeta)\|$, and their rations for grids of $(10)^3$, $(20)^3$, and $(30)^3$ cells.

The results given in this table clearly show the second-order spatial accuracy of this algorithm when exact boundary conditions for $\mathbf{u}$ and ζ are used. The error in $\mathbf{u}$ decreases slightly faster and the error in ζ decreases slightly slower with the grid spacing. Overall we can say that the relative errors in $\mathbf{u}$ and ζ decrease as h^2.

In the second experiment we solved the same problem with the exact velocity boundary conditions and, partially, computed vorticity boundary conditions. To be specific, all of the tangential values of ζ on the boundary were determined using a three-point difference formula. Thus, as stated earlier, these components of ζ on the boundary are expected to be only accurate to $O(h)$. The normal component of ζ on the boundary was obtained from the tangential derivatives of $\mathbf{u}$.

We list the values of the L_2 norms of $\mathbf{u}$, ζ, $E(\mathbf{u})$, $E(\zeta)$ and their ratios in Table II. It is clear from this data that the accuracy of the computed velocity field has deteriorated slightly, but is still approximately second order accurate. The error in the vorticity is now, however, only first-order accurate. We have examined in detail the field of errors in ζ. The relative errors in ζ are maximum near the solid boundary, the (x^1, x^3) plane, and are very small elsewhere. But the (x^1, x^3) plane is precisely where the maximum value of ζ occur so that the overall error in the vorticity field is only first-order accurate, even though the velocity field is still

TABLE I

Norms of the Velocity and Vorticity; the Errors in Both; the Relative Errors in Velocity and Vorticity; and the Ratios as Functions of the Grid Size

Member of cells	$\|\mathbf{u}\|$	$\|E(\mathbf{u})\|$	$\frac{\|E(\mathbf{u})\|}{\|\mathbf{u}\|}$	Ratio	$\|\zeta\|$	$\|E(\zeta)\|$	$\frac{\|E(\zeta)\|}{\|\zeta\|}$	Ratio
$(10)^3$	0.4008	2.882×10^{-3}	7.19×10^{-3}	—	1.402	9.372×10^{-3}	6.68×10^{-3}	—
$(20)^3$	0.3970	6.677×10^{-4}	1.68×10^{-3}	4.3	1.417	2.467×10^{-3}	1.74×10^{-3}	3.8
$(30)^3$	0.3959	2.904×10^{-4}	7.34×10^{-4}	9.8	1.420	1.131×10^{-3}	7.96×10^{-4}	8.4

Note. The ratios are that of the relative error on the coarsest grid to that on the grid. The data in this table was obtained from the solution of the 3-dimensional stagnation point flow ($\beta = \frac{1}{2}$) with exact boundary conditions.

TABLE II

Norms of the Velocity and Vorticity; the Errors in Both; the Relative Errors in Velocity and Voriticity; and the Ratios as Functions of the Grid Size

Number of cells	$\|\mathbf{u}\|$	$\|E(\mathbf{u})\|$	$\frac{\|E(\mathbf{u})\|}{\|\mathbf{u}\|}$	Ratio	$\|\zeta\|$	$\|E(\zeta)\|$	$\frac{\|E(\zeta)\|}{\|\zeta\|}$	Ratio
$(10)^3$	0.4008	2.998×10^{-3}	7.48×10^{-3}	—	1.410	2.117×10^{-2}	1.50×10^{-2}	—
$(20)^3$	0.3970	8.398×10^{-4}	2.12×10^{-3}	3.5	1.422	1.107×10^{-2}	7.78×10^{-3}	1.9

Note. The ratios are that of the relative error on the coarest grid to that on the grid. The data in this table was obtained from the solution of the three-dimensional stagnation point flow ($\beta = \frac{1}{2}$) with the tangential components of the vorticity on the boundary obtained from a (formally) second-order accurate, one-sided difference formula.

second-order accurate. In order to decrease the magnitude of the error in ζ, it is necessary to use a fine grid near the solid boundary.

The second test problem which we have used is Rott's [20] exact solution of the Navier–Stokes equations for the evolution of a time-dependent vortex. In appropriate dimensionless variables, the velocity and vorticity components in cylindrical coordinates (r, θ, z) are

$$u_r = -(2k^2/\mathrm{Re})r, \tag{6.10}$$

$$u_\theta = (k^2 R_0 r)^{-1}\,[1 - \exp(-k^2 r^2/F(t))], \tag{6.11}$$

$$u_z = (4k^2/\mathrm{Re})z, \tag{6.12}$$

$$\zeta_r = 0, \tag{6.13}$$

$$\zeta_\theta = 0, \tag{6.14}$$

$$\zeta_z = (2/F(t))\exp(-k^2 r^2/F(t)). \tag{6.15}$$

Here k is a numerical constant ($=1.12$), Re is a Reynolds number, R_0 is a Rossby number, and

$$F(t) = 1 + b\exp(-4k^2 t/\mathrm{Re}), \tag{6.16}$$

with b an arbitrary, real constant.

This flow is axisymmetric and is the solution for a radial inflow, rotation about the $z(x^3)$ axis, and an accelerating axial flow. The vortex along the x^3 axis spins up from $t=0$ to $t=\infty$.

Although this flow has axial symmetry, we solved for it in Cartesian coordinates. It is easy to rewrite the solution, (6.10) through (6.15), in Cartesian coordinates in order to compare the computed and exact solutions. The computational domain was chosen to be a box with $-3.1 \leqslant x^1 \leqslant 3.1$, $-3.1 \leqslant x^2 \leqslant 3.1$, $0 \leqslant x^3 \leqslant 1.0$. The size of the box in the (x^1, x^2) plane was chosen so that ζ_3 would be very nearly zero at all times on these boundaries. We choose $\Delta x^1 = \Delta x^2 = 0.2$, $\Delta x^3 = 0.1$, so that there were 31 cells in the x^1 and x^2 directions and 10 cells in the x^3 direction. Finally we chose $b = 2.0$, $\mathrm{Re} = 100.0$, and $R_0 = 0.7$. The time evolution of the flow was then computed, with $\Delta t = 0.2$, and boundary conditions for both the velocity and vorticity obtained directly from Eqs. (6.10) to (6.16), for $t=0$ to $t=20$. A number of measures were used to determine the accuracy of the computed solution for the vortex spin up.

Figure 3 is a plot of the time evolution of L_2 norms of the divergence of the vorticity $\|\nabla\cdot\zeta\|$ and the enstrophy $\|\zeta\cdot\zeta\|$, and the value of ζ_3 at the center of the computational domain. We desire to have $\|\nabla\cdot\zeta\|$ small for all time. We can compute the exact values of $\|\zeta\cdot\zeta\|$ and of ζ_3 at $x^1 = x^2 = 0$ and it is easy to show that the enstrophy norm varies in time as $F(0)/F(t)$ and that $\zeta_3(x^1 = x^2 = 0) = 2/F(t)$.

From the results shown in Fig. 3, it is obvious that the divergence of the vorticity is essentially constant in time with a mean value of about 5×10^{-3}. There are, of

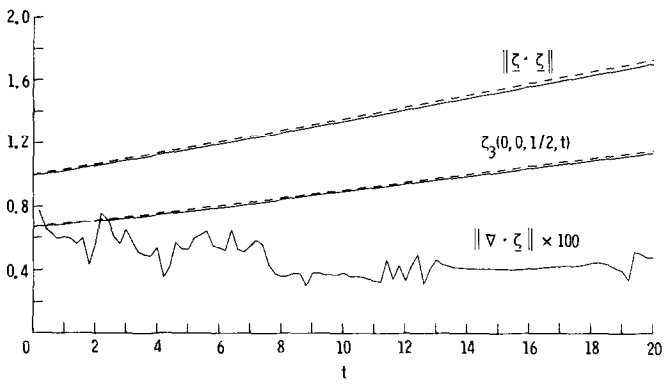


FIG. 3. The time variation of norm of the enstrophy, $\|\zeta \cdot \zeta\|$, the x^3 component of the vorticity at the center of the domain, $\zeta_3(0, 0, \frac{1}{2}, t)$, and the norm of the divergence of the vorticity, $\|\nabla \cdot \zeta\|$, for Rott's exact solution of the Navier–Stokes equations for vortex spinup. The Reynolds number is 100 and the Rossby number is 0.7. The dashed lines are from the exact solution and the solid lines are the computed results from the output of the code.

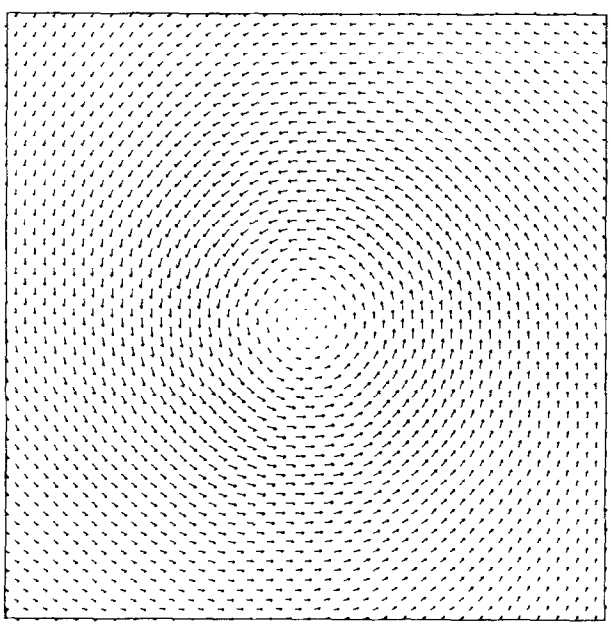
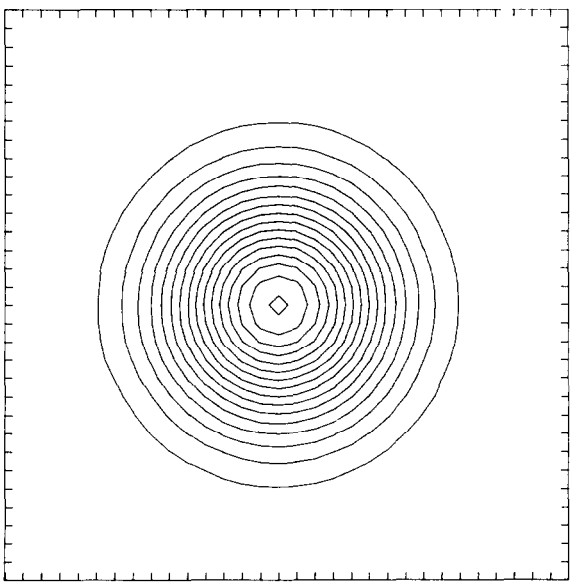


FIG. 4. Projection of the velocity vectors onto the plane $x^3 = 0.5$ at $t = 20$. The flow is the spinup of an axisymmetric vortex with Reynolds number of 100 and Rossby number of 0.7. The maximum value of the velocity in this plane is 0.615. The grid used in the calculation was $31 \times 31 \times 10$ cells.

course, fluctuations in time which lie in the range 3 to 7×10^{-3}. It is clear that the projection scheme preserves a divergence free vorticity field to within $O(h^2)$. The individual values of the vorticity, for example, $\zeta_3(0, 0, \frac{1}{2})$, as well as the norm of the enstrophy are both in excellent agreement with the exact solution.

We also have plotted the projection of the velocity field onto the plane $x^3 = 0.5$, as well as a contour map of ζ_3 in this plane, for $t = 20$. These are shown in Figs. 4 and 5. The velocity vectors shown in Fig. 4 are the projection of $\mathbf{u}$ onto the plane $x^3 = 0.5$ within the computational domain at $t = 20$. Figure 5 is a contour map of ζ_3 on the plane $x^3 = 0.5$ at $t = 20$. The tick marks shown on the border of Fig. 5 indicate the location of the computational grid (31 by 31 cells). An examination of these figures shows that the computed flow field is axisymmetric despite the use of a Cartesian grid in the calculations. But note that there is a very small left–right asymmetry in the velocity field in Fig. 4 and that the vortex has drifted very slightly to the left in Fig. 5. One might expect these very slight deviations from axial symmetry on a Cartesian grid. The results presented in these figures show that even an evolved flow field retains a second-order accuracy.

Finally in Table III, we give values of the error norms for each of the components of the velocity and vorticity at selected times from $t = 0$ to $t = 20$. These errors are relative to the magnitudes of the appropriate variables. It is clear from the results presented here that the error norms are bounded as the flow evolves in time and are approximately second order.



border indicate the size of the grid cells used in the calculation. The grid used was $31 \times 31 \times 10$ cells.

TABLE III

A Tabulation of the Relative Errors Norms of the Velocity (u_1, u_2, u_3) Components and the Vorticity $(\zeta_1, \zeta_2, \zeta_3)$ Components as a Function of Time

Time	$\|E(u_1)\|$	$\|E(u_2)\|$	$\|E(u_3)\|$	$\|E(\zeta_1)\|$	$\|E(\zeta_2)\|$	$\|E(\zeta_3)\|$
5.0	1.10×10^{-2}	1.04×10^{-2}	2.07×10^{-3}	9.21×10^{-4}	8.94×10^{-4}	9.68×10^{-3}
10.0	4.75×10^{-3}	4.36×10^{-3}	5.21×10^{-3}	1.16×10^{-3}	1.12×10^{-3}	8.70×10^{-3}
15.0	4.39×10^{-3}	4.15×10^{-3}	5.99×10^{-3}	1.11×10^{-3}	1.14×10^{-3}	9.93×10^{-3}
20.0	3.94×10^{-3}	3.77×10^{-3}	8.04×10^{-3}	1.38×10^{-3}	1.54×10^{-3}	1.91×10^{-2}

Note. The problem is Rott's exact solution of the Navier–Stokes equations for a vortex spinup, with $\mathrm{Re} = 100$ and $R_0 = 0.7$.

APPENDIX

A. *Approximation Schemes for* $\operatorname{div} \mathbf{u} = 0$, $\operatorname{curl} \mathbf{u} = \hat{\zeta}$

Let the planes $\mathbf{x} = \text{const}$ describe a Cartesian grid in $\mathbf{R}^3$, and denote by e_l a volume element with center at $\mathbf{x}_l$. Suppose that the fundamental domain D is the union of such elements. As Fig. 2 shows, each element e has faces γ, edges σ, and vertices υ and we identify these by a point associated with each. In addition, $|e|$, $|\gamma|$, $|\sigma|$ denote the respective volume, area, and length such that, if $|\sigma| = O(h)$, then $|\gamma| = O(h^2)$, and $|e| = O(h^3)$.

Consistent with this geometric construction, let $\mathbf{u}(e)$, $\mathbf{u}(\gamma)$, and $\mathbf{u}(\sigma)$ denote the average values of $\mathbf{u}$ on e, γ, and σ, respectively. The values $\mathbf{u}(\upsilon)$ associated with the vertices are sometimes called box-variables and are often useful for quadrature evaluations of $\mathbf{u}(e)$, $\mathbf{u}(\gamma)$, and $\mathbf{u}(\sigma)$. For simplicity of notation, these will also designate the quadrature evaluation of these average values in terms of box-variables.

Referring to Fig. 2, consider the faces of an element $e_{i,j,k}$, where the index i is associated with the x^1 or x-axis, j is associated with the x^2 or y-axis, and k is associated with the x^3 or z-axis. The defining relations for the average and difference operators on the $\gamma_{i \pm 1/2, j, k}$ faces are

$$\mu_1 \mathbf{u}(\gamma_{i,j,k}) \doteq [\mathbf{u}(\gamma_{i+1/2,j,k}) + \mathbf{u}(\gamma_{i-1/2,j,k})]/2 \tag{A.1}$$

$$\Delta_1 \mathbf{u}(\gamma_{i,j,k}) \doteq [\mathbf{u}(\gamma_{i+1/2,j,k}) - \mathbf{u}(\gamma_{i-1/2,j,k})]/2; \tag{A.2}$$

the operators μ_2, Δ_2 and μ_3, Δ_3 are similarly defined. The operators $\delta_i \mathbf{u}(\gamma_{i,j,k})$ are determined by

$$\Delta_i \doteq h^i \delta_i, \qquad i = 1, 2, 3 \tag{A.3}$$

where $h^i = \Delta x^i/2$. The defining relations for the average and difference operators relating sides and edges, and edges and vertices are similar.

With these operator definitions in mind, we can construct approximation schemes for both $\operatorname{div} \mathbf{u} = 0$ and $\operatorname{curl} \mathbf{u} = \hat{\zeta}$ as follows:

First, define the volume average of $\operatorname{div} \mathbf{u}$ over an element e as

$$\operatorname{div}_e \mathbf{u} \doteq \frac{1}{|e|} \int \operatorname{div} \mathbf{u} \, de. \tag{A.4}$$

Gauss's theorem evaluates $\operatorname{div}_e \mathbf{u}$ in terms of $\mathbf{u}(\gamma) \cdot \mathbf{n}$ on ∂e, where $\mathbf{n}$ is the unit outward normal, i.e.,

$$\begin{aligned} \operatorname{div}_e \mathbf{u} &\doteq \frac{1}{|e|} \oint_{\partial e} \mathbf{u} \cdot \mathbf{n} \, d|\gamma| \\ &= \frac{1}{|e|} \sum_{\gamma \in \partial e} \mathbf{u}(\gamma) \cdot \boldsymbol{\gamma}, \end{aligned} \tag{A.5}$$

where γ is the oriented area. By suitably arranging the order of summation and using Eqs. (A.2) and (A.3), Eq. (A.5) can be written as

$$\operatorname{div}_e \mathbf{u} = \delta_1 u_1(\gamma) + \delta_2 u_2(\gamma) + \delta_3 u_3(\gamma). \tag{A.6}$$

Using box-variables to evaluate the average values on the right-hand side of Eq. (A.6), one obtains

$$\operatorname{div}_e \mathbf{u} = (\delta_1 \mu_2 \mu_3 u_1 + \delta_2 \mu_1 \mu_3 u_2 + \delta_3 \mu_1 \mu_2 u_3). \tag{A.7}$$

In a similar fashion, define the surface average of the normal component of curl $\mathbf{u}$ over a surface γ as

$$\mathbf{n} \cdot \operatorname{curl}_\gamma \mathbf{u} = \frac{1}{|\gamma|} \int_\gamma \mathbf{n} \cdot \operatorname{curl}_\gamma \mathbf{u}\, d\,|\gamma|. \tag{A.8}$$

Stokes' theorem evaluates $\mathbf{n} \cdot \operatorname{curl}_\gamma \mathbf{u}$ in terms of $\mathbf{u}(\sigma) \cdot \boldsymbol{\sigma}$ on $\partial\gamma$, where $\boldsymbol{\sigma}$ is the unit tangent vector, i.e.,

$$\mathbf{n} \cdot \operatorname{curl}_\gamma \mathbf{u} = \frac{1}{|\gamma|} \sum_{\sigma \in \partial\gamma} \mathbf{u}(\sigma) \cdot \boldsymbol{\sigma}. \tag{A.9}$$

For a Cartesian element the opposite edges $\sigma_\pm$ on a face γ have equal lengths, and the associated unit tangent vectors have different signs. This allows one to rewrite the summation in Eq. (A.9) in terms of an operator acting on the edges. For example, the component on the x^1-face is given by

$$(\mathbf{n} \cdot \operatorname{curl}_\gamma \mathbf{u})\,|_1 = \delta_2 u_3(\sigma) - \delta_3 u_2(\sigma). \tag{A.10}$$

Using box-variables to evaluate the average values on the right-hand side of Eq. (A.10), one obtains

$$(\mathbf{n} \cdot \operatorname{curl}_\gamma \mathbf{u})\,|_1 = (\delta_2 \mu_3 u_3 - \delta_3 \mu_2 u_2). \tag{A.11}$$

With these defining relations, the general form for the (normal) vorticity components on the faces of an element e can be written as

$$(\mathbf{n} \cdot \operatorname{curl}_\gamma \mathbf{u})\,|_i = \delta_j(\mu_k u_k) - \delta_k(\mu_j u_j) \qquad (i, j, k) \tag{A.12}$$

where (i, j, k) indicates an even permutation of (1, 2, 3).

Using the definitions (A.7) and (A.12), Fix and Rose [16] have shown that the Cauchy–Riemann type equations described in **P**(i) in Section I can be solved by the least-squares solution of

$$\operatorname{div}_e \mathbf{u} = 0 \qquad \text{in } D \tag{A.12}$$

$$\operatorname{curl}_\gamma \mathbf{u} = \tilde{\zeta} \tag{A.13a}$$

$$\mathbf{n} \cdot \mathbf{u} = n \cdot u_\Gamma \qquad \text{on } \Gamma \tag{A.13b}$$

when $\operatorname{div}_e \tilde{\zeta} = 0$.

B. *Compact Schemes for Boundary and Initial-Boundary Value Problems*

The following discussion will outline the general development of the schemes that are used in Section III. These schemes will be seen to provide domain decomposition extensions of conventional boundary integral and boundary element methods. As a result, both boundary value and initial-boundary value problems can be handled.

Consider a vector function V having s components. A typical feature of boundary value problems for systems of elliptic equations involving V on a domain D is that $s-s'$ components of V on ∂D are determined, by means of the solution operator on D, by s' prescribed components of V on ∂D, $s' < s$. We may call the s' prescribed components the *primary variables* and the $s-s'$ components the *complementary variables*. In certain simple cases the relationship between these variables can be described by means of simple integral equations.

By constructing an approximate solution operator on D it may be possible to determine the relationship between the primary and complementary variables at N points of interpolation on ∂D. We may expect that these approximate values will converge, as $N \to \infty$, to the solution values $V_{\partial D}$ under reasonable precautions about the construction. This is the basis of discrete boundary integral methods.

Returning to the continuous problem on D, suppose D is partitioned into volume elements, $D = \{e\}$. With arbitrary values of the primary variables chosen on the boundary of each element (but consistent with the values prescribed on $\partial e \cap \partial D$) we can solve the boundary value problem in each element; the solution will be identical to the solution values of the boundary value problem in D in corresponding volume elements if they both have the same values on the boundary of each element and, thus, is continuous across interelement boundaries.

This suggests the following discrete approximation method: In each element e choose the center point of each face γ of e as an interpolation point and, using an appropriate solution operator in e, obtain the discrete boundary integral relationship between the primary and complementary variables at the interpolation points on the boundary of ∂e. We call this a *compact equation* on the element. Next impose continuity conditions at the interpolation points in D and use values prescribed by the problem on D when the interpolation point lies on ∂D. Then, solve the resulting algebraic problem. This, in effect, provides a domain decomposition extension of boundary element methods. (In this construction one may incorporate the continuity conditions quite simply by identifying the left and right limits at an interpolation point by their common value.)

We call this construction a *compact scheme*. It requires for its development only an element-by-element description of the discrete integral equations which relate the primary and complementary variables at the interpolation points on the boundary of each element e. As shown below, this idea may be applied to time dependent problems as well, (cf. [21, 22, 9]).

The weak-element method [23] implements this construction by using as an approximation basis a manifold of solutions of the differential equation (or an

approximation to it) in each element. Necessarily, then, the compact scheme which results is consistent with the differential equation in each element. This construction also leads immediately to a discrete energy estimate which approximates that which applies to the differential equation on D. Thus the convergence of the scheme is assured and leads to second-order accurate results.

We will now indicate how a simple Galerkin method can be used to obtain compact schemes for general volume elements.

A Boundary Value Problem

As an example, we will discuss the Poisson equation $\nabla^2 v = f$. Consistent with earlier notations, let $f(e)$ indicate the value at the center of e and $v(\gamma)$ the value at the interpolation point on a face γ. Define the bilinear boundary operator $B_e(v, w)$ by

$$B_e(v, w) \doteq \oint_{\partial e} \left(w \frac{\partial v}{\partial n} - v \frac{\partial w}{\partial n} \right) d\,|\gamma| \tag{B.1}$$

so that Green's theorem can be written

$$B_e(v, w) = \int_e (w\nabla^2 v - v\nabla^2 w)\, d\,|e|. \tag{B.2}$$

If w is any solution of the homogeneous problem ($\nabla^2 w = 0$) then

$$B_e(v, w) = \int_e wf\, d\,|e| \doteq (w, f)_e. \tag{B.3}$$

Second-order accurate quadrature approximations to Eqs. (B.1) and (B.3) yield

$$B_e^h(v, w) = \sum_{\gamma \in \partial e} \left[w(\gamma) \frac{\partial v(\gamma)}{\partial n} - v(\gamma) \frac{\partial w(\gamma)}{\partial n} \right] |\gamma| \tag{B.4}$$

$$(w, f)_e^h = w(e)\, f(e)\, |e|. \tag{B.5}$$

Suppose e has l faces: let w_i, $i = 1, 2, ..., l$ denote, say, the first l harmonic polynomials (i.e., $\nabla^2 w_i = 0$). Compact equations on e are given by

$$B_e^h(v, w_i) = (w_i, f)_e^h, \qquad i = 1, 2, ..., l \tag{B.6}$$

and provide an $O(h^2)$ truncation error.

These equations establish an algebraic relationship between the l values $v(\gamma)$ and $\partial v(\gamma)/\partial n$ on the faces of ∂e. The coefficients in the equation are determined by evaluating $w_i(\gamma)\,|\gamma|$ and $(\partial w_i(\gamma)/\partial n)\,|\gamma|$ on the faces, $i = 1, 2, ..., l$. For the Poisson

equation $\nabla^2 v = f$, application of these ideas is more straightforward if it is written as the system,

$$\nabla \cdot \mathbf{p} = f \tag{B.7a}$$

$$\mathbf{p} = \nabla v. \tag{B.7b}$$

Then, in the case of a Cartesian grid, use of the functions

$$w = (1, x, y, z, x^2 - y^2, x^2 - z^2) \tag{B.8}$$

in Eq. (B.6) leads to the compact equations

approximate solution v^e on each element as

$$v^e = \sum_{i=1}^{I} c_i v_i + \hat{v}, \tag{B.10}$$

where $\hat{v}$ is a particular solution of $\nabla^2 v = f(e)$. Equation (B.6) determines the coefficients by a Galerkin construction on each element. Since v^e is a solution of $\nabla^2 v^e = f(e)$ on e, it satisfies the energy equation

$$\oint_{\partial e} v^e \frac{\partial v^e}{\partial n} \, d|\gamma| = (v^e, f)_e + \int_e |\nabla v^e|^2 \, d|e|. \tag{B.11}$$

Using the second-order accurate quadrature formulas to evaluate the integral terms, one can approximate Eq. (B.11) as

$$\sum_{\gamma \in \partial e} v^e(\gamma) \frac{\partial v^e(\gamma)}{\partial n} |\gamma| = (v^2, f)_e + \int_e (|\nabla v^e|^2) \, d|e|. \tag{B.12}$$

Recalling the continuity conditions imposed by the compact construction and summing over elements in D, we obtain

$$\sum_{\gamma \in \partial D} v^e(\gamma) \frac{\partial v^e(\gamma)}{\partial n} |\gamma| = \sum_{e \in D} \left\{ (v^e, f)_e + \int_e (|\nabla v^e|^2) \, d|e| \right\}. \tag{B.13}$$

This is a discrete approximation to energy estimates for the solution in D, viz.,

$$\oint_{\partial D} v \frac{\partial v}{\partial n} \, d|\gamma| = \int_D (vf + |\nabla v|^2) \, d|e|. \tag{B.14}$$

This discrete energy estimate, together with the obvious fact that the approximation v^h is consistent with the differential equation, implies by standard arguments that the scheme converges and, in fact, with second-order accuracy.

It is, of course, possible to extend these ideas to more general boundary value problems of the form $Lv = f$. For sufficiently small volume elements, L can be approximated by an operator with constant coefficients in each element, which we call L_e. In this case Green's theorem can be written as (cf. Eq. (B.2))

$$B_e(v, w) = \int_e (wL_e v - vL_e{}^* w)\, d|e|, \tag{B.15}$$

where L_e^* is the adjoint of L_e. Solutions of the adjoint equation $L^* w = 0$, are easily generated in the form

$$w = \exp[\boldsymbol{\alpha} \cdot \mathbf{x}], \tag{B.16}$$

where $\boldsymbol{\alpha}$ satisfies the characteristic polynomial equation $L_e^*(\boldsymbol{\alpha}) = 0$. The compact equations which result,

$$B_e(v, w) = (w, f)_e \tag{B.17}$$

may now involve exponential factors. It is possible to avoid the use of exponentials by using polynomial solutions of $L^* w = 0$ which can be generated by

$$w_i = \frac{\partial^i}{\partial \alpha_{(j)}^i} (\exp[\boldsymbol{\alpha} \cdot \mathbf{x}])\,|_{\alpha = 0}, \qquad i = 0, 1, 2, \ldots; j = 1, 2, 3. \tag{B.18}$$

The weak-element construction just described is based upon using a projection on a manifold of solutions of the differential equation in each element. This same idea can be applied to time-dependent problems as well.

Initial-Boundary Value Problems

As an example, we will base our discussion on the diffusion equation

$$v_t = \nabla^2 v, \qquad \mathbf{x} \in D,\ 0 < t < T \tag{B.19a}$$

with initial and boundary conditions

$$v(\mathbf{x}, 0) = g(\mathbf{x}), \qquad t = 0, \tag{B.19b}$$

$$v(\mathbf{x}_\Gamma, t) = v_I(\mathbf{x}_\Gamma), \qquad \mathbf{x}_\Gamma \in \partial D. \tag{B.19c}$$

Recalling the discussion in Section II, it is sufficient to solve this problem in a time strip S^m: $|t - t_m| \leqslant \tau$; i.e., with initial data $v(\mathbf{x}, t_{m-1/2})$ and boundary data v on $\partial D \times S^m$, we seek to determine $\partial v(\mathbf{x}_\Gamma, t)/\partial n$ on $\partial D \times S^m$ as well as $v(\mathbf{x}, t_{m+1/2})$, $\mathbf{x} \in D$.

Introducing a domain decomposition $D=\{e\}$, we consider the same type of problem on each cylinder set $e\times S^m$. The general solution can be written as

$$v(\mathbf{x}, t; \boldsymbol{\alpha}) = \int A(\boldsymbol{\alpha}) \exp(\boldsymbol{\alpha}\cdot\mathbf{x} - \beta t)\, d\boldsymbol{\alpha}, \tag{B.20}$$

where β satisfies the dispersion relation

$$\beta = |\alpha|^2, \tag{B.21}$$

and $A(\boldsymbol{\alpha})$ is determined from the initial and boundary conditions in $e\times S^m$. If e has l faces, one can seek an approximate solution which interpolates to the initial and boundary conditions on the element in the form

$$v(\mathbf{x}, t) = \sum_{i=0}^{l} A(\boldsymbol{\alpha}_i) \exp(\boldsymbol{\alpha}_i\cdot\mathbf{x} - \beta_i t). \tag{B.22}$$

Once again, the relationship between the primary and complementary variables for the discrete problem can be determined by a Galerkin procedure using an appropriate form of Green's theorem. Let w indicate a solution of the adjoint equation

$$L^*(w) = w_t + \nabla^2 w = 0. \tag{B.23}$$

The application of Green's theorem to this problem leads to the relation (cf. (B.17))

$$\frac{d}{dt}(w, v)_e = B_e(v, w). \tag{B.24}$$

The approximation which results by interpolation is then (cf. Eq. (B.6))

$$\frac{d}{dt}(w, v)_e^h = B_e^h(v, w). \tag{B.25}$$

A time average of Eq. (B.25) on S^m produces the equation

$$\delta_t(w, v)_e^h = B_e^h(v^m, w^m), \tag{B.26}$$

where v^m, w^m indicate time averages over the strip S^m and

$$\delta_t v(t_m) = (v(t_m+\tau) - v(t_m-\tau))/\Delta t \tag{B.27a}$$

$$\mu_t v(t_m) = (v(t_m+\tau) + v(t_m-\tau))/2. \tag{B.27b}$$

Once again, the choice of $s'+1$ solutions v_i of the adjoint Eq. (B.23) will then determine the $s'+1$ complementary solution values at points of interpolation in

$e \times S^m$ in terms of the $s' + 1$ primary solution values. The results is a set of compact equations for the problem. Write (B.19a) as the first-order system

$$\begin{aligned} v_t &= \partial p/\partial x \\ p &= \partial v/\partial x. \end{aligned} \tag{B.28}$$

In this case, e_i is the interval $|x - x_i| < h$. The discrete mixed initial value problem on $e \times S^m$ can be stated as: given $v(e, t_{m-1/2})$ as initial data and $v^m(x_\Gamma)$, $x_\Gamma \in \partial e$, as boundary data, determine (i) $\partial v^m(x_\Gamma)/\partial x$, $x_\Gamma \in \partial e$, and (ii) $v(e, t_{m+1/2})$.

Three elementary polynomial solutions of the adjoint equation (B.23) are

$$w_i \doteq (w_0, w_1, w_2) = (1, x, t + x^2/2), \tag{B.29}$$

where the origin is taken at the center of $e \times S^m$. The compact equations (B.26) then lead to

$$\delta_t v(e, t_m) = B_e^h(v^m, w_0^m)/\Delta x \tag{B.30a}$$

$$0 = B_e^h(v^m, w_1^m)/\Delta x \tag{B.30b}$$

$$\mu_t v(e, t_m) = B_e^h(v^m, w_2^m)/\Delta x. \tag{B.30c}$$

These simplify to

$$\delta_t v = \delta_x p \tag{B.31a}$$

$$\mu_x p = \delta_x v \tag{B.31b}$$

$$\mu_t v = \mu_x v - \frac{h^2}{2}\,\delta_x p, \tag{B.31c}$$

where we have suppressed the reference to $e \times S^m$. The compact scheme results by requiring that v and p be continuous across endpoints of the intervals interior to D, using prescribed values of v on ∂D. (Note that the space operators in (B.31) apply to the face values of (v, p) on the cylinder $e \times S^m$, while the time operators apply to the values of v on the upper and lower bases.)

It is possible to obtain an energy estimate for the system described by Eq. (B.31). Multiply Eq. (B.31a) by $\mu_t v$ and use both Eqs. (B.31b), (B.31c); the resulting equation is

$$\frac{1}{2}\,\delta_t v^2 + (\delta_x v)^2 + \frac{h^2}{2}\,(\delta_x p)^2 = \delta_x(vp). \tag{B.32}$$

Summing over the elements in D, the discrete energy estimate for the approximation is

$$\frac{1}{2}\,\delta_t \sum_e v^2\,\Delta x + \sum_e \left[(\delta_x v)^2 + \frac{h^2}{2}\,(\delta_x p)^2 \right] \Delta x = vp\,|_{\partial D}. \tag{B.33}$$

This expression corresponds to that for the continuous problem

$$\frac{1}{2}\frac{d}{dt}\int_D v^2\,dx + \int_D (v_x)^2\,dx = vp\,|_{\partial D}. \tag{B.34}$$

The compact equations (B.31) are obviously consistent with the differential equation and, in view of the discrete energy estimate (B.33), the Lax equivalence theorem implies the convergence of the compact scheme. This same argument holds in three dimensions using an arbitrary partition of the domain D, $D = \{e\}$.

Finally, it is straightforward to extend these ideas to the advection–diffusion equation

$$v_t + \nabla\cdot(\mathbf{a}v) = \nu\nabla^2 v, \tag{B.35}$$

where $\mathbf{a}$ and ν are coefficients, which in the context of the physical problems of this paper, are associated with the velocity and viscosity of a fluid, respectively. Once again, writing this as a first-order system yields the equations

$$v_t = \nabla\cdot\mathbf{p} \tag{B.36a}$$

$$\mathbf{p} = \nu\nabla v - \mathbf{a}v. \tag{B.36b}$$

The corresponding adjoint equations are

$$-w_t = \nabla\mathbf{q} + \nu^{-1}\mathbf{a}\mathbf{q} \tag{B.37a}$$

$$\mathbf{q} = \nu\nabla w. \tag{B.37b}$$

The resulting form of Green's theorem is now

$$\frac{d}{dt}(w, v)_e = B_e(v, w) = \oint_{\partial e} (w\mathbf{p} - v\mathbf{q})\cdot\mathbf{n}\,d\gamma. \tag{B.38}$$

If, in the time-strip S^m, the coefficient $\mathbf{a}$ in each element e is frozen as $\mathbf{a} = \mathbf{a}^m$, then the elementary solution of (B.37) is

$$w = \exp[\boldsymbol{\alpha}\cdot\mathbf{x} - \beta t], \tag{B.39a}$$

where

$$\beta = \nu\,|\boldsymbol{\alpha}|^2 + \mathbf{a}^m\cdot\mathbf{x} \tag{B.39b}$$

and, again, the origin is taken as the centerpoint of $e\times S^m$. The discrete form of (B.37) is obtained by using appropriate interpolated values in the quadrature approximation to B_e and taking the time average over S^m.

The values $\alpha = 0$, $\alpha = -\nu^{-1}a^m$ in (B.38) lead to steady-state solutions, while

$x - a^m t$ is a simple time-dependent polynomial solution. Thus the appropriate approximation basis for this simple advection–diffusion equation is

$$w_i = (w_0, w_1, w_2) = \left(1, x - a^m t, \exp\left[-\frac{a^m x}{v}\right]\right). \tag{B.40}$$

The compact equations which result are

$$\delta_t v = B_e^h(v^m, w_0^m)/\Delta x \tag{B.41a}$$

$$-\mu_x a^m \mu_t v = B_e^h(v^m, w_1^m)/\Delta x \tag{B.41b}$$

$$[(\theta^m)^{-1} \sinh \theta^m]\, \delta_t v = B_e^h(v^m, w_2^m)/\Delta x, \tag{B.41c}$$

where

$$\theta^m = a^m h/v. \tag{B.42}$$

When Eqs. (B.41a) and (B.41c) are combined (using (B.30)), (B.41c) can be replaced by

$$0 = B_e^h(v^m, (\theta^m)^{-1} \sinh \theta_m - w_2^m). \tag{B.41d}$$

Expanding the right-hand sides of Eqs. (B.41a), (B.41b), and (B.41d), one obtains the compact scheme

$$\delta_t v = \delta_x p \tag{B.43a}$$

$$-\mu_x a^m \mu_t v = \mu_x p - \nu \delta_x v \tag{B.43b}$$

$$\mu_x p - h\rho\delta_x p = \nu\delta_x v - \mu_x a^m (c_x \mu_x v - h\rho\delta_x v), \tag{B.43c}$$

where

$$\rho = \coth \theta^m - (\theta^m)^{-1} \tag{B.43d}$$

and

$$c_x = 1 - \Delta_x a^m (\mu_x a^m)^{-1} \rho. \tag{B.43e}$$

The coefficient ρ given by (B.43d) controls the weighting given to upwind terms in the compact scheme. It can be consistently approximated by

$$\begin{aligned} \rho(\theta) &= \theta/3, & |\theta| < 3 \\ &= \operatorname{sgn} \theta, & |\theta| \geqslant 3 \end{aligned} \tag{B.44}$$

whose use allows us to employ an exponential-type scheme without having, in fact, to calculate exponential terms.

The extension to three dimensions results by using the basis

$$w_i = (1,\ x - a_1 t,\ y - a_2 t,\ z - a_3 t,\ \exp(-a_1 x/\nu),\ \exp(-a_2 y/\nu),\ \exp(-a_3 z/\nu)). \tag{B.45}$$

ACKNOWLEDGMENT

We appreciate helpful discussions with Drs. S. Abarbanel and H. Hassan in the course of our research, and thank Mr. Denis Bushnell for his support of this research.

REFERENCES

1. D. A. ANDERSON, J. C. TANNEHILL, AND R. H. PLETCHER, *Computational Fluid Mechanics and Heat Transfer* (McGraw-Hill, New York, 1984).
2. M. D. GUNZBURGER, *Finite Element Theory and Application—Proceedings, ICASE/NASA Langley Workshop*, edited by D. L. Dwoyer, M. Y. Hussaini, and R. G. Voigt (Springer-Verlag, New York/Berlin, 1987).
3. R. TEMAN, *Finite Element Theory and Application—Proceedings, ICASE/NASA Langley Workshop,*
4. by P. K. Banerjee and S. Makherjee (Elsevier Appl. Sci., London/New York, 1984), p. 137.
5. K. ONISHI, T. KUROKI, AND M. TANAKA, *Topics in Boundary Element Research, Vol. 2*, edited by C. A. Brebbia (Springer-Verlag, New York/Berlin, 1985), p. 209.
6. M. Y. HUSSAINI AND T. A. ZANG, *Annu. Rev. Fluid Mech.* **19**, 339 (1987).
7. A. T. PATERA, *J. Comput. Phys.* **54**, 468 (1984).
8. A. LEONARD, *Annu. Rev. Fluid Mech.* **17**, 523 (1985).
9. T. B. GATSKI, C. E. GROSCH, AND M. E. ROSE, *J. Comput. Phys.* **48**, 1 (1982).
10. S. C. R. DENNIS, D. B. INGHAM, AND R. N. COOK, *J. Comput. Phys.* **33**, 325 (1979).
11. H. F. FASEL, *Approximation Methods for Navier–Stokes Problems*, edited by R. Rautmann (Lecture Notes in Mathematics Vol. 771, Springer-Verlag, New York/Berlin, 1980), p. 209.
12. B. FAROUK AND T. FUSAGI, *Int. J. Num. Methods Fluids* **5**, 1017 (1985).
13. P. ORLANDI, *Comput. and Fluids* **15**, No. 2, 137 (1987).
14. G. A. OSSWALD, K. N. GHIA, AND U. GHIA, *AIAA 19th Fluid Dynamics, Plasma Dynamics and Lasers Conference, June 8–10, Honolulu, Hawaii, 1987.*
15. P. SKERGET AND A. ALUJEVIC, *Z. Angew. Math. Mech.* **65**, T245 (1985).
16. G. J. FIX AND M. E. ROSE, *SIAM J. Numer. Anal.* **22**, 250 (1985).
17. S. KACZMARZ, *Bull. Acad. Polon, Sci. Lett. A*, 355 (1937).
18. K. TANABE, *Num. Math.* **17**, 203 (1971).
19. L. HOWARTH, *Philos. Mag.* **42**, 1433 (1951).
20. N. ROTT, *Z. Angew. Math. Phys.* **9**, No. 6, 543 (1958).
21. R. B. PHILIPS AND M. E. ROSE, *SIAM J. Num. Anal.* **19**, 698 (1982).
22. M. E. ROSE, *J. Comput. Phys.* **49**, 420 (1983).
23. M. E. ROSE, *Num. Math.* **24**, 185 (1975).
24. S. A. ORSZAG, M. ISRAELI, AND M. O. DEVILLE, *J. Sci. Comput.* **1**, 75 (1986).