Implicit Finite-Difference Procedures for the Computation of Vortex Wakes

Joseph L. Steger* and Paul Kutler*
NASA Ames Research Center, Moffett Field, Calif.

Implicit finite-difference procedures for the primitive form of the incompressible Navier-Stokes and the compressible Euler equations are used to compute vortex wake flows. The partial differential equations in strong conservation-law form are transformed to cluster grid points in regions with large changes in vorticity. In addition to clustering, fourth-order accurate, spatial difference operators are used to help resolve the flowfield gradients. The use of implicit time-differencing permits large time steps to be taken since temporal variations are typically small. Computational efficiency is achieved by approximate factorization. Both two-dimensional and preliminary three-dimensional calculations are described.

I. Introduction

THE concentrated vorticity in the near wake of a large aircraft can pose a destructive threat to smaller aircraft within the same airspace. Consequently, experimental and theoretical efforts have been under way to understand, predict (for use in avoidance systems), and possibly reduce the vortex wake hazard. Most theoretical models developed to treat this problem rely on tracing discrete vortices,[1-3] but an alternate and potentially more powerful approach is to use finite-difference procedures.[4,5] Computer programs based on such methods can ultimately account for flowfield nonlinear effects with few ad hoc assumptions.

Implicit finite-difference procedures are developed here to solve the incompressible Navier-Stokes equations and compressible Euler equations for simplified two- and three-dimensional, unsteady vortex wake flows. This paper is divided into five interdependent sections. The flowfield and its numerical implications are discussed in Sec. II. The incompressible equations are developed for simulation as a system of first-order partial-differential equations in Sec. III, and the finite-difference algorithms are described in Sec. IV. Simulation based on the compressible flow equations is discussed in Sec. V. and, finally, simple wake-vortex flow calculations using both kinds of modeling are presented in Sec. VI.

II. Preliminary Considerations to Numerical Simulation

A. The Flowfield

While the wake-vortex flowfield has been studied and described by various researchers, recent surveys by Donaldson and Bilanin[2] and Rossow[6,7] give illuminating accounts of this flow and contain comprehensive bibliographies. In Rossow's description, a typical wake-vortex flowfield is divided into three regions (Fig. 1). In the rollup region, the vorticity shed from the wing is rolled up into two or more vortices. This pattern is then almost stationary and may persist for some time throughout the plateau region. Finally, the vortices begin to diffuse because of viscous and turbulence effects – the decay region. The rollup and plateau regions are found to be inviscid phenomena. In the absence of vortex instabilities such as bursting, the decay region is characterized by time variations that can be small compared to the crossflow spatial variations.

Presented as Paper 76-385 at the AIAA 9th Fluid and Plasma Dynamics Conference, San Diego, Calif., July 14-16, 1976; submitted Oct. 1, 1976; revision received Jan. 18, 1977.

Index category: Jets, Wakes, and Viscid-Inviscid Flow Interactions.

*Research Scientist. Member AIAA.

In a quiescent atmosphere, the entire wake-vortex flowfield can persist for hundreds of wingspans downstream unless the vortices interact destructively. The vorticity shed by a wing typically rolls up into vortex pairs of opposite sign which translate downward under their own induced velocity. Consequently, the wake-vortex problem is characterized as a nonlinear flow that can persist for a long time and where concentrated gradients are convected through the field.

B. Division of the Flowfield for Numerical Simulation

Because the extent of the wake-vortex flowfield is so great, a three-dimensional, unsteady, finite-difference calculation of the entire flowfield is not feasible at this time. Fortunately, the flowfield can be decomposed into simpler problems. The inviscid rollup region, while subject to rapid time variations, is terminated by a region with small streamwise gradients. Consequently, the downstream flowfield cannot greatly influence the rollup region, and rollup can be solved as an independent problem.

Throughout the plateau and decay regions, changes of the dependent variables in the streamwise directions (or with time) can be small. Two approximations are used that make the solution of these persistent regions manageable. In the first, termed the "unsteady analogy," gradients in the streamwise or x direction are ignored and the equations are integrated in time for crossflow only. The applicability of the unsteady analogy is obviously limited, but it can yield useful results – indeed, even portions of the rollup region can be studied by this technique if changes in the streamwise direction are small compared to the crossflow variations. The second approach, used by Hackett and Evans,[3] encompassed a slab of fluid consisting of the crossflow plane and a finite length in the streamwise direction. Either periodic or zero derivative boundary conditions are imposed in the streamwise direction. The advantage of this "slab analogy" is that variations in the streamwise direction are permitted, and therefore a phenomenon like linking can be simulated.

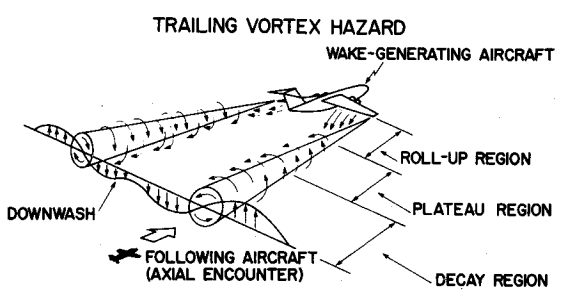

Fig. 1 Flowfield produced by lift-generated vortices; distances not drawn to scale.

However, the vorticity at the start of the slab is of the same age as that at the back of the slab.

C. Selecting a Numerical Algorithm

We describe here numerical procedures that use either the unsteady analogy or the slab analogy. The latter approach could ultimately treat the roll-up problem by extending the streamwise domain and changing the streamwise boundary conditions.

The primary difficulty in applying finite-difference procedures to solve the wake-vortex plateau and decay regions is that changes in the gradients of the dependent variables in the crossflow directions are large and must be resolved. This, in turn, requires a refined grid spacing. Changes of the dependent variables in time are frequently small, however, and the calculation may have to be computed over a long time interval before the solution of interest is obtained.

There are various ways to overcome the aforementioned difficulties, most of which govern the choice of the numerical algorithm. In this paper, an implicit numerical procedure is selected so that the time step is restricted only by the desired resolution of the time-varying dependent variables rather than the Courant number. Fourth-order spatial differencing is used, and grid points are clustered in regions containing large changes in the gradients of the dependent variables. Because the vortices can translate, the clustering transformation is written as a function of time. Computational efficiency is maintained with the implicit algorithms by using approximate factorization techniques[8,9] discussed in Sec. IV.

D. Compressible and Incompressible Equations

The wake-vortex problem of current practical interest occurs at aircraft landing speeds where the flow can be taken as incompressible. Under such conditions, one typically makes an incompressible assumption to remove the sound speed time scale, but in doing so, the solution of a Poisson equation is usually required at each time step to obtain the pressure, or vorticity components, or stream function (see, e.g., Roache[10] for a survey of such procedures). Such a scheme was tried but abandoned for an approximate factorization (alternating direction) implicit algorithm applicable to systems of first-order hyperbolic partial-differential equations. Adaptation of this approximate factorization method to the incompressible equations is described in Secs. III and IV. A lucid description of the algorithm is given by Beam and Warming,[11] who have applied it to the Euler equations. Once a fully implicit scheme is used, the stiffness problem that motivated the incompressible approximation no longer exists. The question then remains, when is it easier to solve the incompressible equations than the compressible equations? As discussed in Sec. V, we have no sure answer at this time.

III. Incompressible Navier-Stokes Equations

In this section, a contrivance is introduced that allows the numerical solution of the incompressible Navier-Stokes equations by the approximate factorization algorithm that Beam and Warming[11] recently used to solve the Euler equations. Clustering transforms are also introduced.

As a motivation for what follows, consider the compressible Euler equations

$$\partial_t \rho + v\partial_y \rho + w\partial_z \rho + \rho(\partial_y v + \partial_z w) = 0 \tag{1a}$$

$$\partial_t p + v\partial_y p + w\partial_z p + a^2(\partial_y v + \partial_z w) = 0 \tag{1b}$$

$$\partial_t(\rho v) + \partial_y(\rho v^2 + p) + \partial_z(\rho vw) = 0 \tag{1c}$$

$$\partial_t(\rho w) + \partial_y(\rho vw) + \partial_z(\rho w^2 + p) = 0 \tag{1d}$$

where v and w are the velocity components in the horizontal and vertical directions, y and z. As the incompressible limit is approached, density ρ is constant along any streamline, the speed of sound, a, approaches infinity, and Eqs. (1a) and (1b) degenerate to the single equation $\partial_y v + \partial_z w \to 0$. Without the time derivative of pressure, $\partial_t p$, the equations cannot be as easily integrated as a conventional hyperbolic time-dependent system of equations. So the following equation is fabricated

$$\partial_t p + \beta(\partial_y v + \partial_z w) = 0 \tag{2}$$

with $\beta \gg 1$. When β is very large, the condition $\partial_y v + \partial_z w \to 0$ is still satisfied.

If $u = (p, v, w)^t$, Eqs. (2), (1c), and (1d) form the conservative system of equations (for constant viscosity coefficient ν)

$$\partial_t u + \partial_y f + \partial_z g = \nu(\partial_{yy} + \partial_{zz}) Du \tag{3}$$

where the incompressible viscous terms have been appended and

$$f = \begin{bmatrix} \beta v \\ p + v^2 \\ vw \end{bmatrix}, \quad g = \begin{bmatrix} \beta w \\ vw \\ p + w^2 \end{bmatrix}, \quad D = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

A uniform constant value of density was assumed with ρ taken as unity.

The same concept carries over to three dimensions

$$\partial_t q + \partial_x \mathcal{E} + \partial_y \mathcal{F} + \partial_z \mathcal{G} = \nu(\partial_{xx} + \partial_{yy} + \partial_{zz}) Dq \tag{4}$$

with

$$q = \begin{bmatrix} p \\ u \\ v \\ w \end{bmatrix}, \quad \mathcal{E} = \begin{bmatrix} \beta u \\ u^2 + p \\ uv \\ uw \end{bmatrix}, \quad \mathcal{F} = \begin{bmatrix} \beta v \\ uv \\ v^2 + p \\ vw \end{bmatrix},$$

$$\mathcal{G} = \begin{bmatrix} \beta w \\ uw \\ vw \\ w^2 + p \end{bmatrix}, \quad D = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5}$$

and $\beta \gg 1$. The Jacobians of $\mathcal{E}$, $\mathcal{F}$, and $\mathcal{G}$ are

$$A = \frac{\partial \mathcal{E}}{\partial q} = \begin{bmatrix} 0 & \beta & 0 & 0 \\ 1 & 2u & 0 & 0 \\ 0 & v & u & 0 \\ 0 & w & 0 & u \end{bmatrix},$$

$$B = \frac{\partial \mathcal{F}}{\partial q} = \begin{bmatrix} 0 & 0 & \beta & 0 \\ 0 & v & u & 0 \\ 1 & 0 & 2v & 0 \\ 0 & 0 & w & v \end{bmatrix}, \quad C = \frac{\partial \mathcal{G}}{\partial q} = \begin{bmatrix} 0 & 0 & 0 & \beta \\ 0 & w & 0 & u \\ 0 & 0 & w & v \\ 1 & 0 & 0 & 2w \end{bmatrix} \tag{6}$$

and the eigenvalues of each matrix are real and given by

$$\sigma(A) = u, u, u \pm \sqrt{\beta + u^2} \tag{7a}$$

$$\sigma(B) = v, v, v \pm \sqrt{\beta + v^2} \tag{7b}$$

$$\sigma(C) = w, w, w \pm \sqrt{\beta + w^2} \tag{7c}$$

Each coefficient matrix has a complete set of eigenvectors. For large β, the condition number $(\sigma_{max}/\sigma_{min})$ of each matrix is large and the system of partial-differential equations is stiff.

To cluster grid points in regions where the vorticity changes rapidly, the following transforms are introduced

$$\xi = \xi(x) \tag{8a}$$

$$\eta = \eta(y) \tag{8b}$$

$$\zeta = \zeta(z,t) \tag{8c}$$

$$\tau = t \tag{8d}$$

Note that ζ is also a function of time, enabling the coordinates to translate in z with the moving vorticity. Such a transformation is needed for multiple vortex flows in which the vortices convect downward much like a vortex pair of opposite sign.

Under this transformation, the equations become (see Viviand[12])

$$\begin{aligned}&\partial_\tau(J^{-1}q) + \partial_\xi(\xi_x J^{-1}\mathcal{E}) + \partial_\eta(\eta_y J^{-1}\mathcal{F}) \\ &+ \partial_\zeta(\zeta_t J^{-1}q + \zeta_z J^{-1}\mathcal{G}) = \nu[\partial_\xi(\xi_x\partial_\xi(\xi_x J^{-1}Dq)) \\ &+ \partial_\eta(\eta_y\partial_\eta(\eta_y J^{-1}Dq)) + \partial_\zeta(\zeta_z\partial_\zeta(\zeta_z J^{-1}Dq))]\end{aligned} \tag{9}$$

where $J^{-1} = (\xi_x\eta_y\zeta_z)^{-1}$, $\zeta_t = -z_\tau\zeta_z$, while $\xi_x = (x_\xi)^{-1}$, etc. Equation (9) is computationally simplified by introducing the new variables $\hat{q} \equiv J^{-1}q$.

IV. Numerical Method

The numerical algorithm used to solve Eq. (9) is detailed here. Throughout it is assumed that the reader is acquainted with alternating direction implicit (ADI) or Douglas-Gunn finite-difference methods (see, e.g., Refs. 9 or 13). The same algorithm applies to the Euler equations with minor alterations (as discussed in Sec. V; see also Ref. 11).

A. Approximate Factorization

For large values of β, each coefficient matrix A, B, or C is ill-conditioned (stiff) and an implicit procedure is used to solve Eq. (9). Trapezoidal-rule time differencing of Eq. (9) is given by

$$\begin{aligned}&\hat{q}^{n+1} + \frac{h}{2}\left\{\delta_\xi\left[\xi_x(\hat{\mathcal{E}}^{n+1} + \hat{\mathcal{E}}^n)\right] + \delta_\eta\left[\eta_y(\hat{\mathcal{F}}^{n+1} + \hat{\mathcal{F}}^n)\right]\right. \\ &\left. + \delta_\zeta\left[\zeta_t(\hat{q}^{n+1} + \hat{q}^n) + \zeta_z(\hat{\mathcal{G}}^{n+1} + \hat{\mathcal{G}}^n)\right]\right\} = \hat{q}^n \\ &+ \frac{\nu h}{2}\left\{\delta_\xi\xi_x\delta_\xi\xi + \delta_\eta\eta_y\delta_\eta\eta_y + \delta_\zeta\zeta_z\delta_\zeta\zeta_z\right\} D(\hat{q}^{n+1} + \hat{q}^n)\end{aligned} \tag{10}$$

where spatial indices are to be inferred, δ_ξ, δ_η, and δ_ζ are centered difference approximations (specified later) to ∂_ξ, ∂_η, and ∂_ζ and are not necessarily the same operator for each term. Here $\hat{\mathcal{E}} \equiv J^{-1}\mathcal{E}$, etc., $\tau = n\Delta\tau$, and $h = \Delta\tau$. Finally, we evaluate ζ_t and ζ_z at time n and delete that superscript for convenience.

To avoid solving a nonlinear set of equations at each time level, the Taylor series linearizations are introduced

$$\hat{\mathcal{E}}^{n+1} = \hat{\mathcal{E}}^n + A^n(\hat{q}^{n+1} - \hat{q}^n) \tag{11a}$$

$$\hat{\mathcal{F}}^{n+1} = \hat{\mathcal{F}}^n + B^n(\hat{q}^{n+1} - \hat{q}^n) \tag{11b}$$

$$\hat{\mathcal{G}}^{n+1} = \hat{\mathcal{G}}^n + C^n(\hat{q}^{n+1} - \hat{q}^n) \tag{11c}$$

where A, B, and C are the Jacobians given by Eq. (6). Use of these linearizations in Eq. (10) with rearrangement yields

$$\begin{aligned}&\hat{q}^{n+1} + \frac{h}{2}\left[\delta_\xi\xi_x A^n + \delta_y\eta_y B^n + \delta_\zeta\left(\zeta_t + \zeta_z C^n\right)\right. \\ &\left. - \nu\left(\delta_\xi\xi_x\delta_\xi\xi_x + \delta_\eta\eta_y\delta_\eta\eta_y + \delta_\zeta\zeta_z\delta_\zeta\zeta_z\right) D\right]\hat{q}^{n+1} \\ &= \hat{q}^n - \frac{h}{2}\left[\delta_\xi\xi_x A^n + \delta_y\eta_y B^n + \delta_\zeta\left(\zeta_t + \zeta_z C^n\right)\right. \\ &\left. - \nu\left(\delta_\xi\xi_x\delta_\xi\xi_x + \delta_\eta\eta_y\delta_\eta\eta_y + \delta_\zeta\zeta_z\delta_\zeta\zeta_z\right) D\right]\hat{q}^n \\ &+ h\left[\delta_\xi\xi_x u^n + \delta_\eta\eta_y v^n + \delta_\zeta\zeta_z w^n\right] Dq^n\end{aligned} \tag{12}$$

where we have used the relations

$$\mathcal{G} = (A - uD)q, \qquad \mathcal{F} = (B - vD)q, \qquad \mathcal{G} = (C - wD)q$$

To avoid solving for a matrix of large bandwidth, the difference equations are factored into one-dimensional operators that approximately reproduce the original set of equations. A basic approximate factorization (AF) of Eq. (12) is

$$\begin{aligned}&\mathcal{L}_{+x}\mathcal{L}_{+y}\mathcal{L}_{+z}\hat{q}^{n+1} = \mathcal{L}_{-x}\mathcal{L}_{-y}\mathcal{L}_{-z}\hat{q}^n \\ &+ h\left(\delta_\xi\xi_x u^n + \delta_\eta\eta_y v^n + \delta_\zeta\zeta_z w^n\right) D\hat{q}^n\end{aligned} \tag{13}$$

where the operators

$$\mathcal{L}_{\pm x} \equiv \left[I \pm \frac{h}{2}\left(\delta_\xi\xi_x A^n - \nu\delta_\xi\xi_x\delta_\xi\xi_x D\right)\right], \quad \text{etc.} \tag{14}$$

are defined to avoid rewritting the lengthy expressions. Equation (13) reproduces Eq. (12) with the addition of higher-order, cross-differences such as

$$\frac{h^2}{4}\delta_\xi\xi_x A^n\delta_\eta\eta_y B^n(\hat{q}^{n+1} - \hat{q}^n)$$

and

$$\frac{h^3}{8}\delta_\xi\xi_x A^n\delta_\eta\eta_y B^n\delta_\zeta\zeta_z C^n(\hat{q}^{n+1} + \hat{q}^n)$$

all of which are formally second-order-accurate error terms. Alternately, one can arrange Eq. (12) in the delta form[14]

$$\begin{aligned}&\left\{I + \frac{h}{2}\left(\delta_\xi\xi_x A^n - \nu\delta_\xi\xi_x\delta_\xi\xi_x D\right) + \frac{h}{2}\left(\delta_\eta\eta_y B^n - \nu\delta_\eta\eta_y\delta_\eta\eta_y D\right)\right. \\ &\left. + \frac{h}{2}\left[\delta_\zeta\left(\zeta_t + \zeta_z C^n\right) - \nu\delta_\zeta\zeta_z\delta_\zeta\zeta_z D\right]\right\}(\hat{q}^{n+1} - \hat{q}^n) \\ &= -h\left[\delta_\xi\xi_x\hat{\mathcal{E}}^n + \delta_\delta\eta_y\hat{\mathcal{F}}^n + \delta_\zeta\left(\zeta_t\hat{q}^n + \zeta_z\hat{\mathcal{G}}^n\right)\right. \\ &\left. - \nu\left(\delta_\xi\xi_x\delta_\xi\xi_x + \delta_\eta\eta_y\delta_\eta\eta_y + \delta_\zeta\zeta_z\delta_\zeta\zeta_z\right) D\hat{q}^n\right]\end{aligned} \tag{15}$$

and then approximately factor the left-hand side

$$\mathcal{L}_{+x}\mathcal{L}_{+y}\mathcal{L}_{+z}(\hat{q}^{n+1}-\hat{q}^n)=\mathcal{R}^n \tag{16}$$

where $\mathcal{R}^n$ is defined to be the right-hand side of Eq. (15).

The AF schemes, Eqs. (13) and (16), differ in their higher-order cross-differences and each has its own advantages. The delta form, Eq. (16), is, for the most part, easier to code, requires fewer algebraic operations, and boundary conditions are easier to implement. The AF scheme given by Eq. (13), however, is considered to have better stability properties because analysis of the model convection problem in three dimensions shows that the delta AF form is weakly unstable because of the $\xi\eta\zeta$ cross-term (Beam and Warming[14]).

The boundary-condition problem alluded to above is readily illustrated from Eq. (13)

$$\mathcal{L}_{+x}\mathcal{L}_{+y}\mathcal{L}_{+z}\hat{q}^{n+1}=f^n \tag{17}$$

whith the right-hand-side terms incorporated into the vector f^n. Let

$$\bar{\hat{q}}=\mathcal{L}_{+y}\mathcal{L}_{+z}\hat{q}^{n+1} \tag{18}$$

then, in the inversion of

$$\mathcal{L}_{+x}\bar{\hat{q}}=f^n \tag{19}$$

$\bar{\hat{q}}$ and not q^{n+1} must be used as a boundary condition. In the delta form, the quantities

$$\mathcal{L}_{+y}\mathcal{L}_{+z}(q^{n+1}-q^n)$$

must be supplied as the boundary values, but these terms can be taken as zero with, at most, a first-order error.

In two dimensions, both AF algorithms were coded, but in three dimensions, only the delta form was coded because of its simplicity. The weak instability that should be present in this scheme in three dimensions was apparently detected (the issue is clouded because of the need to add dissipation for the nonlinear problem). However, experimentation shows that the instability can be controlled by adding a fourth-order dissipation term (if enough real viscosity is not present) and by keeping $\Delta t/\Delta x \le O(1)$. The ratios $\Delta t/\Delta y$ and $\Delta t/\Delta z$ are not so restricted and, for the problems considered, it is only in these directions that extensive grid clustering is needed.

B. Correction of the β Cross-Difference Error

Because the parameter β must be large to make the fictitious continuity equation accurate, the system of equations is very stiff. One can sometimes construct problems in which the solution of very stiff systems is sometimes more accurately captured by implicit procedures than well-conditioned systems. Approximate factorization schemes do not necessarily behave in the same way, however, and this is true of Eq. (13) or (16). Because of the cross-difference terms, the β coefficient is not just confined to the continuity equation. For example, in the term

$$\frac{h^2}{4}\delta_\xi\xi_x A^n\delta_\eta\eta_y B^n(\hat{q}^{n+1}-\hat{q}^n)$$

the product of A^nB^n occurs in a fashion and this matrix product is

$$AB=\begin{bmatrix} 0 & \beta v & \beta u & 0 \\ 0 & 2uv & \beta+2u^2 & 0 \\ u & v^2 & 3uv & 0 \\ 0 & wv & 2uv & uv \end{bmatrix} \tag{20}$$

Thus β contaminates the ξ-momentum equation and, for β very large, the differencing of the ξ-momentum equation can be inaccurate. The appearance of such terms in the continuity equation is not important since they are small compared to the spatial derivatives that have β as a coefficient.

To keep the accuracy of the AF solution from becoming meaningless, the size of β must be restricted, though clearly it must be large enough that continuity is satisfied. The choice $\beta=O(1/\Delta t)$ maintains first-order time accuracy. To achieve higher-order time accuracy, the AF algorithms are modified for iterative correction. The algorithms can be adjusted as follows, here illustrated for Eq. (16)

$$\mathcal{L}_{+x}\mathcal{L}_{+y}\mathcal{L}_z(\hat{q}^{n+1}-\hat{q}^n)=\mathcal{R}=(I-D)(\overline{q^{n+1}}-\hat{q}^n) \tag{21}$$

where D is defined as before by Eq. (5). In the first step, q^{n+1} is taken as $\bar{q}^n$, β is set to the $O(1/\Delta t)$, and $\hat{q}^{n+1}$ is found. In the iterative correction, q^{n+1} is then set equal to $\hat{q}^{n+1}$, β is set to $\sqrt{\beta/2}$, and a better estimate of $\hat{q}^{n+1}$ is computed. This iteration can be continued until $\beta\to\frac{1}{2}$ and second-order accuracy is achieved, but in many calculations, one iterative correction with β taken as 1 (not $\sqrt{\beta/2}$) will suffice.

C. Spatial Differencing, Block Tridiagonal Inversion, and Addition of Fourth-Order Dissipation

Central-difference operators are used to approximate the spatial first derivatives because their use with trapezoidal time differencing for the model linear convection problem leads to unconditional but neutral stability. Use of one-sided or biased spatial difference operators can be unconditionally stable for real eigenvalues of the correct sign, but the same operator is always unstable for eigenvalues of opposite sign. The coefficient matrices A, B, and C each have positive and negative real roots so, unless each matrix is split into parts with all real negative and all real positive eigenvalues, biased differencing must be avoided. The viscous terms are centrally differenced in the conventional manner for a diffusion equation.

The following spatial difference operators are used (illustrated in ξ). Near outer boundaries where the grid is highly stretched

$$2\Delta\xi\delta_\xi\xi_x=(\xi_x|_{j+1}E_\xi^{+1}-\xi_x|_{j-1}E_\xi^{-1}),\ O(\Delta\xi^3) \tag{22a}$$

$$2(\Delta\xi)^2\delta_\xi\xi_x\delta_\xi\xi_x=(\xi_x|_{j+1}+\xi_x|_j)(\xi_x|_{j+1}E_\xi^{+1}-\xi_x|_j)$$
$$-(\xi_x|_j+\xi_x|_{j-1})\times(\xi_x|_j-\xi_x|_{j-1}E_\xi^{-1})\ O[(\Delta\xi)^4] \tag{22b}$$

where $E_\xi^{\pm}q_j=q_{j\pm1}$. Elsewhere Hermite difference operators are used to maintain fourth-order accuracy of the convection terms

$$(\Delta\xi/3)\left(E_\xi^{+1}+4+E_\xi^{-1}\right)\delta_\xi\xi_x=\left(\xi_x|_{j+1}E_\xi^{+1}-\xi_x|_{j-1}E_\xi^{-1}\right),$$
$$O[(\Delta\xi)^5] \tag{23a}$$

$$[(\Delta\xi)^2/3]\left(E_\xi^{+1}+4+E_\xi^{-1}\right)\delta_\xi\xi_x\delta_\xi\xi_x$$
$$=\left(\xi_x|_{j+1}+\xi_x|_j\right)\left(\xi_x|_{j+1}E_\xi^{+}-\xi_x|_j\right)$$
$$-\left(\xi_x|_j+\xi_x|_{j-1}\right)\left(\xi_x|_j-\xi_x|_{j-1}E_\xi^{-1}\right),\ O[(\Delta\xi)^4] \tag{23b}$$

These terms can also be expressed in the operationally superior Pade form, e.g.

$$2\Delta\xi\delta_\xi\xi_x=\frac{\xi_x|_{j+1}E_\xi^{+1}-\xi_x|_{j-1}E_\xi^{-1}}{(1/6)[E_\xi^{+1}+4+E_\xi^{-1}]},\quad O[(\Delta\xi)^5] \tag{24}$$

At the outer boundaries where the grid is stretched exponentially, fourth-order differencing with its averaging

operator $(E_\xi^{+1}+4I+E_\xi^{-1})/6$ can be less "accurate" than second-order differencing. Near the outer boundaries, therefore, the second-order accurate form is used. Derivatives in η and ζ are everywhere treated in the same fashion.

There is not much point in bringing the viscous terms to fourth-order accuracy because ν is generally small. The truncation error of the first derivative term with fourth-order differencing is likely to be much larger than the truncation error term of the second-order-accurate differenced viscous term. Indeed, for very small values of ν, it is extremely difficult to keep the truncation error of the convection part from being much larger than the viscous terms themselves. The averaging operator $(E_\xi^{+1}+4+E_\xi^{-1})/6$ used with the second difference Eq. (23b) does not change the order of accuracy and it is used solely to simplify the inversion process when Eq. (23a) is used.

In the solution of Eq. (13) or Eq. (16), the right-hand side is formed, and the first part of the inversion task is represented by Eq. (19). If the second-order spatial differencing is used, the operator $\mathfrak{L}_{+x}$ generates a block tridiagonal matrix for each line of grid point in the ξ-direction. The blocks of this matrix have nonzero elements, x, as sketched below

$$\begin{bmatrix} 0 & x & 0 & 0 \\ x & x & 0 & 0 \\ 0 & x & x & 0 \\ 0 & x & 0 & x \end{bmatrix} \quad \begin{bmatrix} x & 0 & 0 & 0 \\ 0 & x & 0 & 0 \\ 0 & 0 & x & 0 \\ 0 & 0 & 0 & x \end{bmatrix} \quad \begin{bmatrix} 0 & x & 0 & 0 \\ x & x & 0 & 0 \\ 0 & x & x & 0 \\ 0 & x & 0 & x \end{bmatrix}$$

where the nonzero elements of A are detected in the blocks to the left and right of the diagonal. The viscous term contributes only to the diagonal of each block. The blocks of the tridiagonal are in reduced form and can be solved by a call to a 2×2 block tridiagonal solver and two calls of a scalar tridiagonal solver.

The Pade form of fourth-order-accurate differencing is used because the inversions are no more difficult to apply than what was just described (see Ref. 11). Both sides of Eq. (19) are multiplied by $(E_\xi^{+1}+4+E_\xi^{-1})/6$, which means that f is simply averaged in ξ. Once this step is complete, exactly the same tridiagonal as before is obtained except for new diagonal weights. The spatial derivatives on the right-hand side of Eq. (13) or (16) should be evaluated with consistent accuracy. For the basic AF algorithm, Eq. (13), the Hermite differencing has been used to evaluate these terms, but for Eq. (16), the conventional five-point central differencing is more easily used and is stable.

The trapezoidal rule AF scheme is neutrally stable in the model linear convection problem if central spatial difference operators are used. Consequently, for the nonlinear problem, one expects to add dissipation to stabilize the solution unless ν is sufficiently large, which is unlikely in practice. In order that the numerical dissipation not swamp the real viscosity, fourth-order differences $\epsilon(\delta_{\xi\xi\xi\xi}+\delta_{\eta\eta\eta\eta}+\delta_{\zeta\zeta\zeta\zeta})\hat{q}^n$ are appended to the right side of Eq. (10), where

$$\delta_{\xi\xi\xi\xi}=-\left(E_\xi^{-2}-4E_\xi^{-1}+6-4E_\xi^{+1}+E_\xi^{+2}\right)\xi_x \tag{25}$$

etc., and $\epsilon\simeq1/24$. To prevent storage and instability problems, these terms are factored into the algorithm in the form

$$\mathfrak{L}_{+x}\mathfrak{L}_{+y}(I+\epsilon\delta_{\xi\xi\xi\xi})^{-1}(I+\epsilon\delta_{\eta\eta\eta\eta})^{-1}(I+\epsilon\delta_{\zeta\zeta\zeta\zeta})^{-1}\mathfrak{L}_{+z}\hat{q}^{n+1}=f^n \tag{26}$$

for Eq. (13).

D. Translation of Grid

Vortices of opposite sign will translate downward (or upward) and can quickly move out of the clustered grid region. To compensate for this, the equations have been transformed so that the entire grid can move in the ζ-direction with time, or the grid can be locally reclustered in ζ with time. The latter approach proved to be unstable for the types of clustering functions used, so the entire grid was allowed to shift with the centroid of a spatially averaged value of vorticity. The initial clustering was kept fixed.

A simple control function was used to center the grid to the centroid of vorticity

$$\zeta_t^{n+1}=\zeta_t^n+\alpha(z_{\text{centroid of vorticity}}-z_{\text{center of grid}}) \tag{27}$$

with $\alpha<1$ and, typically, $\alpha\simeq1/4$. Thus the farther the centroid of vorticity is from the middle of the grid, the more ζ_t is adjusted to force realignment. Such a control function has a hysteresis effect, the grid center vacillates from behind to ahead of the centroid of vorticity. An improved control function is desirable because this motion can induce instability unless an adequate amount of dissipation is present.

V. Compressible Flow Equations

The compressible Euler equations have also been coded to treat the inviscid vortex problem, but only in two dimensions. While an assumption of compressibility is not generally needed (unless perhaps an effect such as a heated jet is taken into account), the compressible equations contain both computational advantages and disadvantages with respect to the incompressible equations. The equations can still be stiff and, because time variations tend to be smaller than spatial variations, an implicit scheme is still preferred.

The Euler equations in two dimensions and conservation law form are

$$\partial_t q+\partial_y\mathcal{F}+\partial_z\mathcal{G}=0 \tag{28}$$

with

$$q=\begin{bmatrix}\rho\\ \rho v\\ \rho w\\ e\end{bmatrix},\quad \mathcal{F}=\begin{bmatrix}\rho v\\ \rho v^2+p\\ \rho wv\\ v(e+p)\end{bmatrix},\quad \mathcal{G}=\begin{bmatrix}\rho w\\ \rho vw\\ \rho w^2+p\\ w(e+p)\end{bmatrix}$$

and the equations transform precisely as before. However, the Euler equations possess the identities

$$\mathcal{F}=\frac{\partial\mathcal{F}}{\partial q}q=Bq \tag{29a}$$

$$\mathrm{G}=\frac{\partial\mathcal{G}}{\partial q}q=Cq \tag{29b}$$

where B and C are the Jacobians given by

$$B=\begin{bmatrix} 0 & 1 & 0 & 0 \\ -v^2+\left(\frac{\gamma-1}{2}\right)(v^2+w^2) & -(\gamma-3)v & -(\gamma-1)w & \gamma-1 \\ -vw & w & v & 0 \\ b_{41} & b_{42} & -(\gamma-1) & \gamma v \end{bmatrix} \tag{30a}$$

$$C=\begin{bmatrix} 0 & 0 & 1 & 0 \\ -vw & w & v & 0 \\ -w^2+\left(\frac{\gamma-1}{2}\right)(v^2+w^2) & -(\gamma-1)v & (3-\gamma)w & \gamma-1 \\ c_{41} & -(\gamma-1)vw & c_{43} & \gamma w \end{bmatrix} \tag{30b}$$

$$b_{41}=-\gamma v(e/\rho)+(\gamma-1)v(v^2+w^2) \tag{30c}$$

$$b_{42}=\gamma(e/\rho)-((\gamma-1)/2)(3v^2+w^2) \tag{30d}$$

$$c_{41}=-\gamma w(e/\rho)+(\gamma-1)w(v^2+w^2) \tag{30e}$$

$$c_{43}=\gamma(e/\rho)-((\gamma-1)/2)(v^2+3w^2) \tag{30f}$$

Consequently, the time linearizations simplify to

$$\mathfrak{F}^{n+1}=B^n q^{n+1} \tag{31a}$$

$$\mathfrak{G}^{n+1}=C^n q^{n+1} \tag{31b}$$

and the basic AF scheme represented by Eq. (13) is simplified in the compressible case. The delta form of the AF scheme is not simplified by Eqs. (29).

The chief advantage of the compressible equations over the incompressible equations is in their lack of the β term. This term, in the incompressible case, usually requires a subtraction step using iteration. However, for low Mach numbers, the dependent variable e in dimensionless form is typically much larger in magnitude than the variables ρ, ρv, and ρw in dimensionaless form. Since e is an element of B and C, the unsteady solution accuracy may be significantly degraded because of terms such as the AF cross-term

$$(\Delta t/2)^2\delta_\eta\eta_y B^n\delta_\zeta\zeta_z C^n(\hat{q}^{n+1}-\hat{q}^n)$$

if "very" incompressible conditions are attempted. This situation can apparently occur because elements 2,1 and 2,3 of matrix BC are not small and coefficients containing e are no longer confined to the energy equation. However, a thorough study of the problem has not been made and the practical limitations of the AF scheme for the compressible equations have not yet been ascertained.

Compared to the two-dimensional incompressible equations, matrices B and C of the compressible equations are irreducible so the tridiagonal block solver must deal with the full 4×4 block. In contrast, the incompressible equations are reducible to a 2×2 block (as shown earlier). The compressible equations also require more storage to accommodate the extra dependent variable, and the elements of the coefficient matrices are not as easily formed. But unless the flow is characterized by very low Mach numbers the compressible inviscid equations are probably preferred over the incompressible inviscid equations when the AF implicit algorithms are used.

IV. Computational Results

A series of computations was run to verify the accuracy of the finite-difference procedures for both the compressible and incompressible equations. The numerical schemes were then used to compute more complex vortex-wake flowfields and to demonstrate their qualitative behavior. Calculations with the

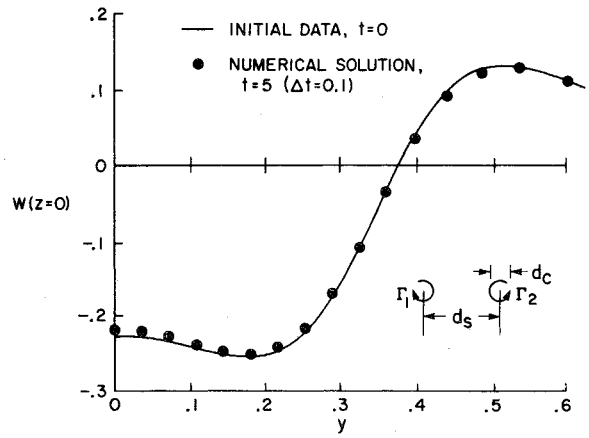

Fig. 3 Verification that Galilean transformation removes vortex-pair translational velocity; $d_s=2.47\ d_c$, $d_c=0.28284$, $\Gamma_1=\Gamma_2=0.25$, $\omega=0.0568$.

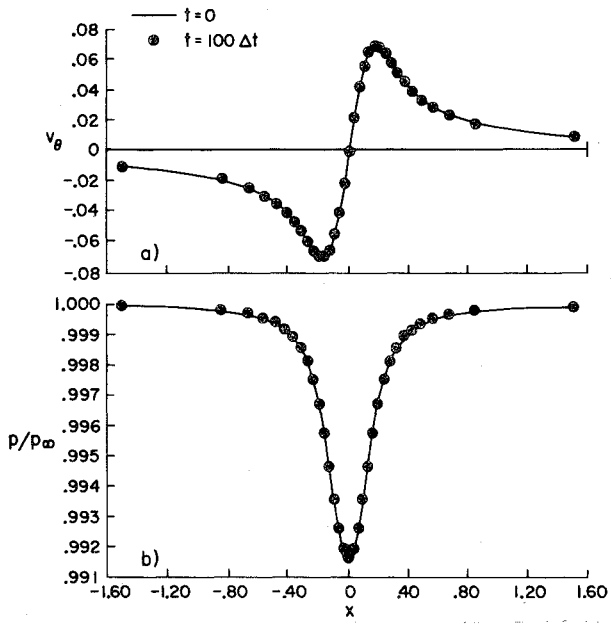

Fig. 2 Accuracy check of compressible equations for single Lamb vortex: a) pressure; b) radial velocity; $\Gamma=0.106$, $r_c=0.15$, CN = 5.

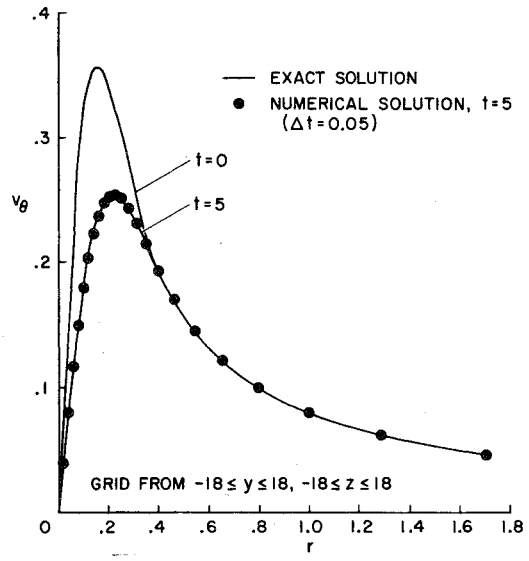

Fig. 4 Accuracy check of incompressible Navier-Stokes equations for single Lamb vortex; $\Gamma=0.5$, $d_c=0.28284$, $\nu=0.001$.

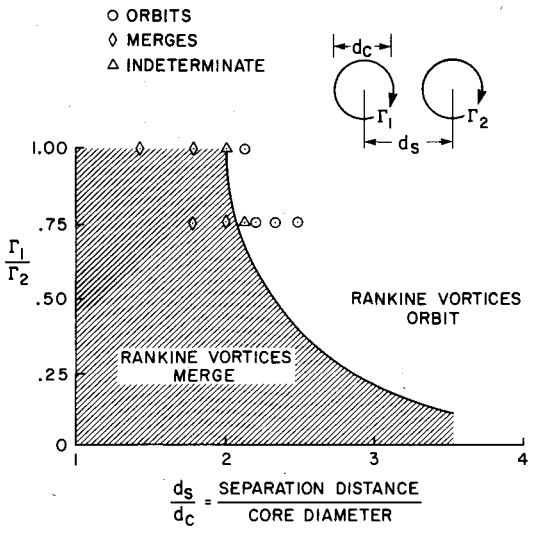

Fig. 5 Combination of strength and spacing required for Lamb vortices to merge.

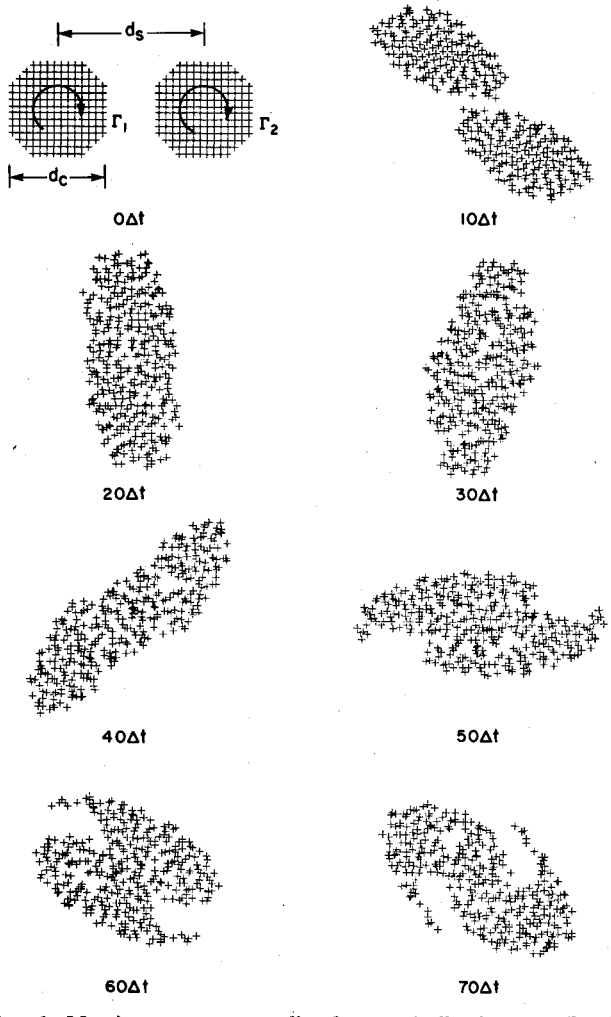

Fig. 6 Merging sequence predicted numerically for two Lamb vortices; $\Gamma_1 = \Gamma_2$, $d_s = 1.7\, d_c$.

compressible equations are restricted to inviscid, two-dimensional unsteady flow (unsteady analogy). The incompressible equations have been run in this manner as well as for simplified three-dimensional unsteady flow (slab analogy), usually with the additional laminar viscous terms.

A. Initial Data

The initial vorticity distribution is arbitrary and data measured experimentally can be used. In all results reported here, however, the initial vorticity distribution is generated by the super-position of Lamb (or spreading line) vortices given by

$$v_\theta = v/\cos\theta = w/\sin\theta$$

$$= \frac{\Gamma}{2\pi r}\left(1 - e^{-r^2/r_c^2}\right); \qquad r_c^2 = 4\nu t_0 \qquad (32)$$

(see Lamb,[15] p. 592, or Batchelor,[16] p. 204). In Eq. (32), r is the radial distance from the center of the vortex, Γ is the circulation constant, r_c is the approximate core radius, ν is the kinematic viscosity, and t_0 is initial time. In the three-dimensional case, u (the streamwise velocity component) must also be specified and can be arbitrary for periodic boundary conditions.

B. Unsteady Analogy

Verification runs for numerical accuracy used the unsteady analogy. The Lamb vortex is an exact solution of the incompressible Navier-Stokes equations and was used to test the accuracy of the numerical procedures. If the flowfield is initialized using a single Lamb vortex, inviscid theory requires that it remain indefinitely. This test serves as a steady-state verification of spatial differencing. A check on the unsteady accuracy is obtained by a solution of the viscous-flow case for which the exact solution is known.

The ability of the compressible flow procedure to capture the Lamb vortex solution and maintain it for 100 time steps at a Courant number of 5 (five times the maximum allowable time step for an explicity scheme) is shown in Figs. 2(a) and 2(b). As expected, the largest error in the pressure and velocity

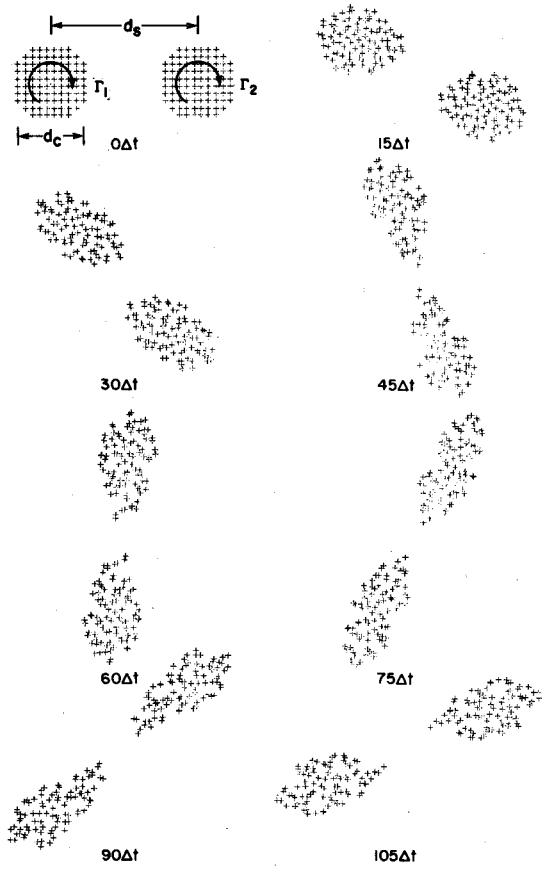

Fig. 7 Orbiting sequence predicted numerically for two Lamb vortices; $\Gamma_1 = \Gamma_2$, $d_s = 2.2\, d_c$.

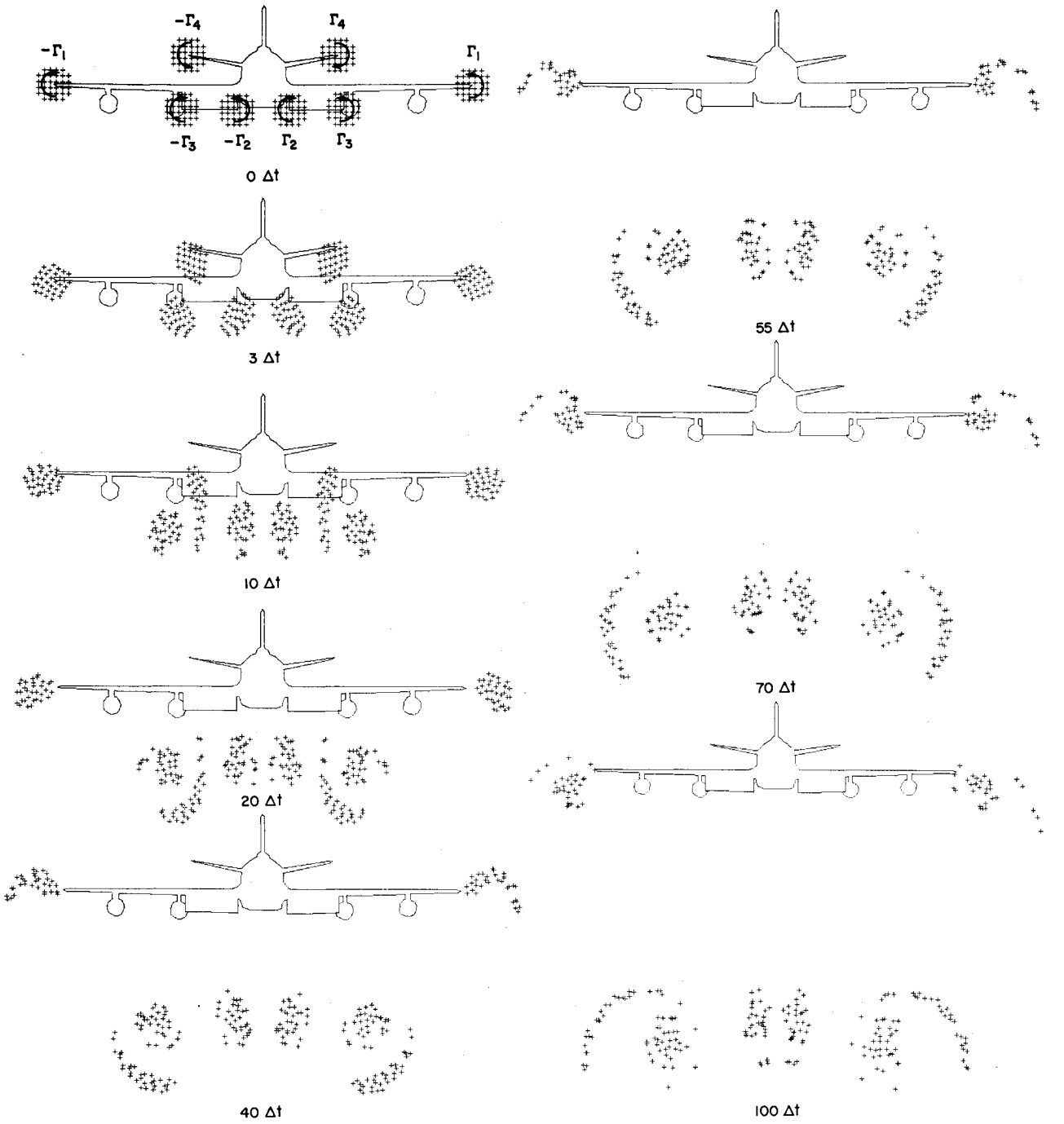

Fig. 8 Particle traces in vortex wake of B-747-like aircraft; $\Gamma_1 = 4\Gamma_4$, $\Gamma_2 = \Gamma_3 = 6\Gamma_4$.

plot occurs in the regions where the curvature of the dependent variables is the largest.

Another test of accuracy is the translation of a vortex pair of opposite signs. The sink rate of two such vortices is given by[17]

$$w = -\Gamma/2\pi d_s \equiv \omega \tag{33}$$

where d_s is the separation distance measured from the center of each core and call ω a Galilean constant. If the grid is translated at this velocity by setting $\zeta_t = \omega\zeta_z$ in Eq. (9), the vortex pair remains stationary and undamped in the computational plane according to inviscid theory. The results of such a calculation for the incompressible equations are shown in Fig. 3 where $\Delta t/\Delta x_{\min} = 1.0$ and $d_c = 2r_c$ is an input parameter to Eq. (32). The compressible procedure yields the same results and is slightly more efficient computationally than the incompressible algorithm.

A simple viscous decay result is presented in Fig. 4. The initial vortex is specified at $t_0 = 0$, and the computation proceeds through 100 times increments until $t = 5$ using a value of $\nu = 0.001$. The accuracy of the solution is very good for this case, and noticeable errors do not occur for this grid spacing until the value of ν is lowered by a factor of 10 or more. This accuracy should be sufficient to model the decay of vortices behind typical aircraft if an eddy viscosity model is used.

Conditions under which an inviscid vortex pair of the same sign merge are investigated and are used as the final accuracy check. The free parameters in the problem are the ratio of the strengths of the two vortices (Γ_1/Γ_2) and the ratio of the separation distance to the core diameter (d_s/d_c). Rossow[7] mapped merging conditions for the Rankine vortex, but, unfortunately, this vortex is a poor choice for finite-difference methods because the slope of the angular velocity is

discontinuous at the core radius. Consequently, since Lamb vortices are used in the present calculations, only qualitative comparison with Rossow's data is sought. Figure 5 shows the results of such a comparison. In come cases there is, as indicated, uncertainty as to whether merging occurs because dissipative terms have been added to the numerical algorithm to control nonlinear instabilities and, consequently, the vortex core diameters are slowly growing. If the integration is carried on too long, cases that should not merge will merge simply because of this numerical dissipation. Using the defined core diameter $d_c = 4\sqrt{\nu t_0}$, the results show that conditions for inviscid convective merging of Lamb vortices are nearly identical to those obtained by Rossow for Rankine vortices.

Passive particle traces at various time levels are shown in Figs. 6 and 7 for two vortices of equal strength. At $d_s/d_c = 1.7$, the two vortices merge (Fig. 6), while for $d_s/d_c = 2.2$, the two vortices orbit (Fig. 7).

As a final example of the capability of the algorithm, a multiple vortex wake representing the flowfield behind a large commercial jet aircraft in a landing configuration was simulated using the compressible equations. Inviscid Lamb vortices were positioned on both sides of the aircraft behind the wing tip, the horizontal stabilizer, and the inborad and outboard edges of the inboard flaps. The results of this calculation are shown in Fig. 8, which represent a sequence of pictures at progressive time levels of passive particles injected in the wake. In the figure, the weaker vortices off the stabilizer are immediately drawn down by the much stronger flap vortices. The flap vortex pairs translate downward at a much higher rate than the wing-tip pair. When qualitatively compared with experimental visualizations of the same configuration,[18,19] the numerical results exhibit the same behavior as those of the experiment.

C. Slab Analogy

The three-dimensional calculations using the previously described slab analogy must be considered qualitative at this point because a reasonably coarse grid was used – $7 \times 25 \times 25$ grid points in the x, y, and z directions – to remain in high-speed core on the CDC-7600 computer. The use of a plane of symmetry at the midspan location, $y = 0$, somewhat offsets the small dimension.

Figure 9 is a sketch of the computational domain. Periodic boundary conditions are imposed in x so that a particle

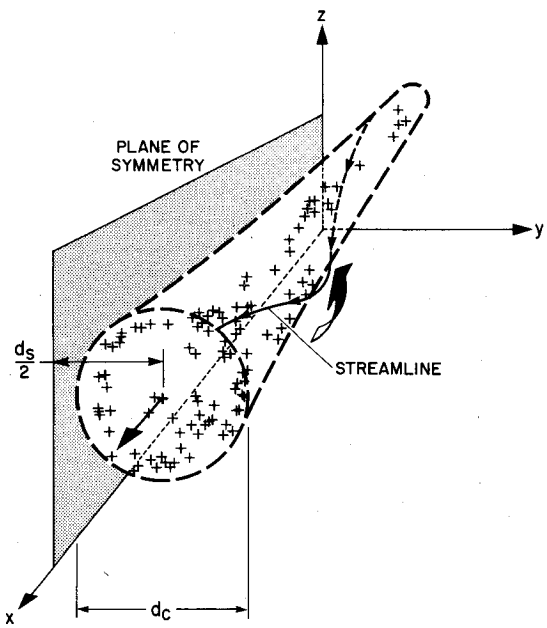

Fig. 9 Sketch of computational domain.

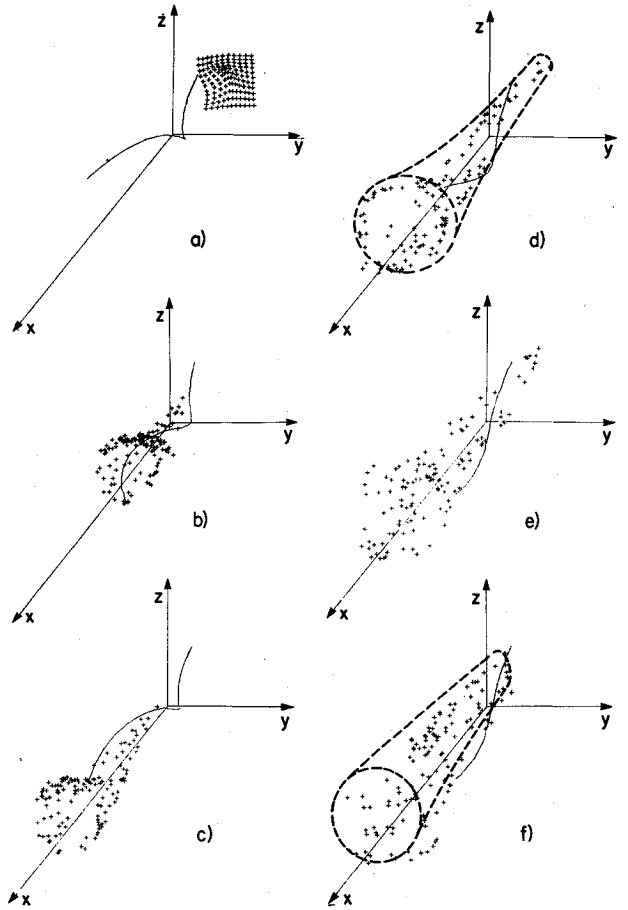

Fig. 10 Passive particle display of vortex pair in three-dimensional computational domain.

leaving at $x = x_{max}$ emerges at $x = 0$. A plane of symmetry is used at $y = 0.0$ so that any single vortex in the computational plane is part of a vortex pair and will thus translate in z. Boundary conditions at $y = 0.0$ are simply

$$v = 0 \tag{34a}$$

$$\partial_y u = \partial_y w = \partial_y p = 0 \tag{34b}$$

and second-order-accurate differencing is used to simulate the slope condition; e.g., in the transformed plane

$$(\eta_y / 2\Delta\eta)\,[-3u(0) + 4u(\Delta\eta) - u(2\Delta\eta)] = 0 \tag{35}$$

The entire flowfield is initialized with a distribution of the Lamb vortex with the streamwise velocity u taken as freestream less some factor of the crossflow vorticity. The streamwise velocity variation obtained in this manner attempts to model a velocity defect chiefly confined to the vortex core. Positive values of laminar viscosity are used to check a sometimes weak instability due to grid movement to stay centered on the centroid of vorticity as described in Sec. IV D.

Computational results from this simulation are displayed in Fig. 10. Passive particles are initially distributed in a plane at $x = 0$ [as shown in Fig. 10(a) after one time step was taken] and the particles are then allowed to convect downstream with time. The line plot shown is a streamline. The particles and streamline are all displayed in the transformed space, but the main display remains in an almost uniform portion of the grid and may be taken as a conventional visual display. At a later time [Fig. 10b], the outer particles have moved midway through the grid in x and, because the core velocity is less than

the freestream, the particles funnel out in x. In Fig. 10(c), the outer particles have reached the end of the x grid and are about to emerge in the upstream portion of the display at $x=0$. The same situation has occurred but at later times in Figs. 10(d), 10(e), and 10(f). The displays show an aging flow. The thickened core and less sinuous streamline indicate viscous decay of the radial and streamwise velocities.

VII. Concluding Remarks

Implicit finite-difference algorithms based on the method of approximate factorization have been used to compute simple vortex-wake flows. The algorithms are general and have the capability to simulate complex nonlinear flow interactions without the use of extensive simplifications. Any arbitrary vorticity distribution may be used as initial data, including those found experimentally. The numerical methods should be useful in testing turbulence models and in the exploration of vortex instabilities such as bursting. Work on a better means of clustering grid points near regions of concentrated vorticity and other improvements are under way.

References

[1]Rossow, V.J., "On the Inviscid Rolled-Up Structure of Lift-Generated Vortices," *Journal of Aircraft,* Vol. 10, Nov. 1974, pp. 647-650.

[2]Donaldson, C. duP. and Bilanin, A.J., "Vortex Wakes of Conventional Aircraft," AGARD-AG-204, 1975.

[3]Hackett, J.E. and Evans, M.R., "Vortex Wakes Behind High-Lift Wings," *Journal of Aircraft,* Vol. 8, 1971, pp. 334-338.

[4]Teske, M.E., "Turbulent Wakes in a Stratified Fluid, Part II," (User's Summary Guide to "Wake" Computer Program), Aeronautical Research Associates of Princeton, Inc., Princeton, N.J., ARAP Report 226, 1974.

[5]Bilanin, A.J., et al., "Viscous Effects in Aircraft Trailing Vortices," *NASA Symposium on Wake-Vortex Minimization,* Feb. 1976.

[6]Rossow, V.J., "Survey of Computational Methods for Lift-Generated Wakes, Aerodynamic Analyses Requiring Advanced Computers, Part II," NASA SP-347, 1975, pp. 879-923.

[7]Rossow, V.J., "Inviscid Modeling of Aircraft Trailing Vortices," *NASA Symposium on Wake-Vortex Minimization,* Feb. 1976.

[8]Douglas, J., Jr., and Gunn, J.E., "A General Formulation of Alternating Direction Methods," *Numerical Mathematics,* Vol. 6, 1964, pp. 428-453.

[9]Yanenko, N.N., *The Method of Fractional Steps,* Springer-Verlag, New York, 1971.

[10]Roache, P., *Computational Fluid Dynamics,* Hermosa, Albuquerque, 1972.

[11]Beam, R.M. and Warming, R.F., "An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation-Law Form," *Journal of Computational Physics,* Vol. 22, Sept. 1976, pp. 87-110.

[12]Viviand, H., "Conservative Forms of Gas Dynamic Equations," *La Recherche Aerospatiale,* No. 1, Jan.-Feb. 1974, pp. 65-68.

[13]Mitchell, A.R., *Computational Methods in Partial Differential Equations,* Wiley, New York, 1969.

[14]Beam, R. and Warming, R.F., "On the Construction and Application of Implicit Factored Schemes for Conservation Laws," (in preparation).

[15]Lamb, H., *Hydrodynamics,* 6th ed., Dover, New York.

[16]Batchelor, G.K., *An Introduction to Fluid Dynamics.* Cambridge University Press, Cambridge, England, 1970.

[17]Prandtl, L. and Tietjens, O.G., *Fundamentals of Hydro- and Aeromechanics.* Dover, New York, 1957.

[18]Corsiglia, V.R., Rossow, V.J., and Ciffone, D.L., "Experimental Study of the Effect of Span Loading on Aircraft Wakes," NASA TM X-62,431, 1975.

[19]Ciffone, D.L. and Lonzo, C., Jr., "Flow Visualization of Vortex Interactions in Multiple Vortex Wakes Behind Aircraft, NASA TM X-62,459, 1975.