NASA Technical Memorandum 78495

On Improving the Iterative Convergence Properties of an Implicit Approximate-Factorization Finite Difference Algorithm

Jean-Antoine Desideri, J. L. Steger
and J. C. Tannehill

June 1978

National Aeronautics and
Space Administration

NASA Technical Memorandum 78495

On Improving the Iterative Convergence Properties of an Implicit Approximate-Factorization Finite Difference Algorithm

Jean-Antoine Desideri, Iowa State University, Ames, Iowa
J. L. Steger, Ames Research Center, Moffett Field, California
J. C. Tannehill, Iowa State University, Ames, Iowa

National Aeronautics and
Space Administration

Ames Research Center
Moffett Field, California 94035

TABLE OF CONTENTS

		Page
I.	INTRODUCTION	1
	A. Overview	1
	B. Governing Equations and Numerical Algorithm	2
II.	STABILITY ANALYSIS AND APPLICATIONS	6
	A. Generalities	6
	B. The Case of a Scalar Linear Equation	10
	1. Generalities	10
	2. The case of periodic boundary conditions	13
	3. The case of specified boundary data	20
	C. Application: Sequence of Parameters	26
III.	NUMERICAL EXPERIMENTATION ON STEADY-STATE CONVERGENCE	33
	A. Model Problem	33
	B. Results	35
IV.	ON THE EFFECT OF SPACIAL VARIATION OF THE JACOBIAN MATRICES	49
	A. Analysis	49
	B. Numerical Experiments on Scalar Model Equations with Variable Coefficients	54
	C. Further Comments	65
V.	CONCLUSION	68
VI.	REFERENCES	70
VII.	ACKNOWLEDGMENTS	72

VIII. APPENDIX A: MATRIX FORM OF THE FINITE-DIFFERENCE EQUATION FOR THE CASE OF A SCALAR DIFFERENTIAL EQUATION 73

A. Some Background on Kronecker Products and Sums 73

B. Application to the Finite-Difference Equation 74

IX. APPENDIX B: EIGENSYSTEMS OF TRIDIAGONAL MATRICES 77

X. APPENDIX C: STABILITY CONDITION FOR PERIODIC BOUNDARY CONDITIONS AND SECOND-ORDER SMOOTHING 82

XI. APPENDIX D: STABILITY CONDITION FOR PERIODIC BOUNDARY CONDITIONS AND FOURTH-ORDER SMOOTHING 86

XII. APPENDIX E: EFFECTIVE EIGENVALUES OF THE SMOOTHING OPERATORS AT LARGE COURANT NUMBERS 93

XIII. APPENDIX F: STABILITY CONDITION FOR SPECIFIED BOUNDARY DATA AND SMALL COURANT NUMBERS 98

A. Second-Order Smoothing 99

B. Fourth-Order Smoothing 102

XIV. APPENDIX G: OF THE EIGENVALUES OF THE MATRIX Γ 104

I. INTRODUCTION

A. Overview

Time-accurate implicit finite-difference schemes for the Euler and compressible Navier-Stokes equations are used to obtain steady as well as unsteady flow-field solutions. If only a steady-state solution is required, iterative paths that are not restricted to be time accurate can be sought to accelerate steady-state convergence. This is the concept of relaxation which has been used successfully for inviscid transonic flow.

The current time accurate implicit algorithms [1-6] for the Euler or compressible Navier-Stokes equations rely on approximate factorization or alternating direction (ADI) techniques to achieve computational efficiency. The same technique is the basis of many of the most successful relaxation procedures (e.g., [7-11]). As a consequence, it would seem that implicit algorithms developed for time accurate flow simulation could be adapted into successful relaxation procedures, and indeed this is the case.

In this work, the iterative convergence properties of a currently popular approximate-factorization implicit finite-difference algorithm are studied both analytically and experimentally. These studies are limited to the two-dimensional Euler equations, with emphasis on transonic flow computations. However, the major results are expected to apply to those flows that are governed by the complete Navier-Stokes equations, but in which the convection phenomena still play the most important role in the determination of the essential features of the numerical algorithm, at least from the standpoint of stability.

To achieve better numerical efficiency, large time-steps are often needed for problems that are unduly stiff. To permit this, modifications to the algorithm are made (in Section IB) in an attempt to enhance its stability properties. The success of this attempt is supported by a theoretical analysis (Chapter II), and a numerical experimentation (Chapter III). With achievement of stable large time-steps permitted, another technique is also employed to improve the iterative convergence rate. This technique, which consists of using a cyclic sequence of time-steps, appears promising after examination of a simple model problem (see Section IIC). In Chapter III, a variety of numerical experiments are conducted on the modified algorithm and the use of a sequence of time-steps.

Finally, it was observed in the course of this work, that the numerical algorithm could be subject to a particular form of instability due to variable coefficients. A discussion on this topic is presented in Chapter IV.

In Section IB, which follows, the definition of the base algorithm is recalled, and a modified algorithm is proposed.

B. Governing Equations and Numerical Algorithm

The conservative form of the Euler equations in Cartesian coordinates and for two-dimensional flow is given by:

$$\partial_t \vec{q} + \partial_x \vec{E} + \partial_y \vec{F} = 0 \qquad (1)$$

where

$$\vec{q} = \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ e \end{pmatrix}, \quad \vec{E} = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ u(e + p) \end{pmatrix}, \text{ and } \vec{F} = \begin{pmatrix} \rho v \\ \rho uv \\ \rho v^2 + p \\ v(e + p) \end{pmatrix}$$

Implicit finite-difference schemes developed for the Euler equations in nonconservative [1] or conservative form [2-3] share the essential features of central spacial differencing (for stability) and alternating-direction-like structure (for efficiency). The conservative differencing scheme is used here for transonic flow applications because it correctly captures shock waves, but the results obtained here should apply to the nonconservative differencing scheme as well.

The implicit finite-differencing scheme can be represented as

$$(I + h\delta_x A^n)(I + h\delta_y B^n)(\vec{q}^{n+1} - \vec{q}^n) = -\Delta t(\delta_x \vec{E}^n + \delta_y \vec{F}^n) - \epsilon_e[(\nabla_x\Delta_x)^2 + (\nabla_y\Delta_y)^2]\vec{q}^n \qquad (2)$$

where

δ_x and δ_y are central three-point difference operators

A, B are the 4×4 Jacobian matrices $\left[\frac{\partial E}{\partial q}\right]$, $\left[\frac{\partial F}{\partial q}\right]$

$h = \theta\Delta t$, with $\theta = 1$ or $1/2$ for Euler implicit, or trapezoidal time differencing;

$(\nabla\Delta)^2\vec{q}^n$ are fourth-order numerical dissipation terms with coefficient $\epsilon_e \geq 1/16$

$\vec{q}^n = (\vec{q}^n_{jk})$ with $x_j = (j-1)\Delta x$ and $y_k = (k-1)\Delta y$.

The operators δ and $\nabla\Delta$ are understood to operate on any product of terms that follow to their right and, for example,

$$\delta_y B^n \vec{q}^{n+1} = (B^n_{j,k+1}\vec{q}^{n+1}_{j,k+1} - B^n_{j,k-1}\vec{q}^{n+1}_{j,k-1})/(2\Delta y)$$

$$(\nabla_x\Delta_x)^2\vec{q}^n = \vec{q}^n_{j-2,k} - 4\vec{q}^n_{j-1,k} + 6\vec{q}^n_{jk} - 4\vec{q}^n_{j+1,k} + \vec{q}^n_{j+2,k}$$

Central difference operators are used because A and B usually have both positive and negative eigenvalues, for one sign of which the algorithm

is always unstable for only forward or only backward spacial differencing. The fourth-order numerical dissipation terms provide damping and are needed to control what is usually referred to as nonlinear instability. They also serve to smooth out small numerical inconsistencies especially due to slightly improper boundary conditions.

Although the basic differencing is stable for linear problems without dissipation added, the scheme given by Equation (2) has to be modified if the dissipation is to be allowed to increase with the use of large values of Δt. More precisely, the dissipation coefficient ϵ_e should vary directly as Δt to prevent nonlinear instability and to maintain steady-state consistency. It is only in this way that the steady-state solution can be independent of Δt. However, because the numerical dissipation is added explicitly, use of $\epsilon_e > 1/16$ would in itself cause linear instability. Consequently, ϵ_e cannot be maintained proportional to Δt and very large values of Δt cannot be taken without effectively reducing the amount of added numerical dissipation. In many flow-field problems, lack of sufficient numerical dissipation can cause instability.

Adding numerical dissipation implicitly would allow ϵ_e to assume any positive value, and in particular, ϵ_e could vary with Δt. Unfortunately, use of fourth-order implicit numerical dissipation requires the inversion of block pentadiagonal matrices which are twice as costly as the block tridiagonal inversions required for Equation (2). Use of second-order smoothing allows tridiagonal structure but is inaccurate. However one expects that less restricted values of ϵ_e could be obtained if a proper portion of the numerical dissipation that fits within the tridiagonal structure is treated

implicitly. This concept ultimately leads to the following modification of the numerical algorithm

$$(I + h\delta_x A^n - \epsilon_i \nabla_x \Delta_x)(I + h\delta_y B^n - \epsilon_i \nabla_y \Delta_y)(\vec{q}^{n+1} - \vec{q}^n)$$

$$= -\Delta t(\delta_x \vec{E}^n + \delta_y \vec{F}^n) - \epsilon_e[(\nabla_x \Delta_x)^2 + (\nabla_y \Delta_y)^2]\vec{q}^n \qquad (3)$$

which has now been implemented in recent flow-field codes [5,6].

In the following chapters, this modified algorithm is analyzed and compared to the base differencing scheme, Equation (2), from the standpoint of iterative convergence.

II. STABILITY ANALYSIS AND APPLICATIONS

The efficiency of a finite-difference algorithm for time-marching problems depends crucially on the stability limitations this algorithm is subject to. In Section IIA, the basic notions related to stability are recalled. In Section IIB, stability bounds are derived for the case where the numerical algorithm is applied to a scalar, linear, partial-differential equation, with constant coefficients and linear boundary conditions. Applications of the results of this analysis are considered in Section IIC.

A. Generalities

The importance of the stability condition is reviewed here for finite-difference methods in general, and for relaxation techniques in particular.

Recall that a numerical algorithm, for an initial-value problem, is said to be stable when, for arbitrary bounded starting solution $u^o = u(0)$, the solution u^n produced by n applications of this algorithm remains bounded as n tends to infinity. This limit may result from considering either one of two limiting processes which are: (1) a mesh refinement process, and (2) a search for a steady-state solution.

In the case of a mesh refinement process, one evaluates a sequence of solutions u^{n_i} $(i = 1,2, \ldots)$ which are all candidate approximations to the exact solution $u(t_f)$ of the initial-value problem, for some fixed final time t_f. At the ith step in this process, n_i applications of the algorithm are made, with initial solution $u^o = u(0)$ and using a time-step $\Delta t_i = t_f/n_i$ which tends to zero as i tends to infinity. P. D. Lax (see Richtmyer and Morton [12]) has shown that given a properly posed initial-value problem and a finite-difference approximation to it that satisfies

the consistency condition, the stability condition is the necessary and sufficient condition for convergence. Here the convergence is the one of $\|u^{n_i} - u(t_f)\|$ to zero as i tends to infinity (or Δt_i tends to zero); that is, the convergence of the finite-difference integral operator to the exact integral operator over a fixed domain in the limit of a mesh refinement.

In the case of the search for a steady-state solution, the time-step Δt may be fixed and n tends to infinity because the final time $t_f = n \Delta t$ should do so. There the stability condition is not sufficient for (steady-state) convergence, and one usually relies on numerical evidence to demonstrate the latter.

For both cases, violation of the stability condition produces an amplification of the various forms of errors that are present in the numerical solution. These are: truncation errors (due to inexact differentials), round-off errors (due to truncated arithmetics), errors due to slightly inconsistent boundary conditions, etc. For linear (constant coefficient) algorithms, the growth of the errors, if it happens, is generally exponential with n (sometimes polynomial, or a combination of the two), so that the numerical solution very rapidly becomes totally meaningless whenever computable. However, most schemes are stable when operating with a time-step that does not exceed a certain maximum allowable value Δt_{max} which unfortunately decreases with the mesh spacing parameters Δx and Δv, and also depends (for nonlinear schemes) on the solution u^n itself. For example, for usual explicit algorithms (e.g., [13]) applied to the Euler equations, stability is enforced by the well-known Courant, Friedrichs, Lewy (CFL) condition [14]. This condition considerably reduces the

efficiency of these algorithms when used as relaxation techniques. This is particularly true when, for an accurate resolution, a very fine mesh is required. On their part, implicit algorithms usually have the favorable property of being unconditionally stable, at least for some simple test equations. In practice, such unconditionality is rarely truly achieved, but time-steps that are significantly larger than those permitted by the CFL condition can be successfully used (see Chapter III). This is the reason that motivated the choice of an implicit algorithm in this work on relaxation.

These considerations suffice to explain the importance of stability for finite-difference time-marching techniques. However, when such a technique is employed as an artifice to solve a problem where time does not appear, one might want to relate the stability condition to the assumption of a (known) theorem dealing with relaxation techniques per se. This is the "contraction-mapping theorem" (see, e.g., [15]) which can be stated as follows: Given a closed domain D in a complete normed vector space (e.g., R^m), and an application f, with domain D and range included in D, which is contracting in the sense that

$$\forall u,v \in D \ , \qquad \|f(u) - f(v)\| \leq \rho \|u - v\| \tag{4}$$

for some real positive number $\rho < 1$, the following statements are true: (1) the equation $u = f(u)$ has a unique solution u^* (on D), and (2) for any $u^1 \in D$, the sequence u^n given by $u^0 = u^1$ and $u^{n+1} = f(u^n)$ $(n = 0,1,2, \ldots)$ is well defined and converges to u^*. Also, the following bound holds:

$$\|u^n - u^*\| \leq \frac{\rho}{1 - \rho} \|u^n - u^{n-1}\| \tag{5}$$

When a finite-difference method is used as a relaxation technique, the iterative formula can indeed be written as

$$u^{n+1} = f(u^n) \tag{6}$$

where $f(u)$ can generally be cast into the following quasi-linear form:

$$f(u) = L(u)u + b(u) \tag{7}$$

where $u^n = u(n\Delta t)$ is the (m-dimensional) solution-vector, $L(u)$ is an $m \times m$ coefficient matrix and $b(u)$ is an m-vector generally resulting from the application of boundary conditions. The unfortunate dependence on u of $L(u)$ and $b(u)$ renders the analysis very difficult in very general cases. For this reason, one is generally satisfied when successful in proving stability of the algorithm in the special case where the coefficients are frozen to some, perhaps arbitrary but fixed, nominal values L and b. Then, the boundedness of u^n, for arbitrary u^o, is equivalent to the following stability condition:

$$\|L\| \leq 1 \tag{8}$$

for some norm. Clearly, this condition is a weak form of the assumption that f is contracting of the cited theorem (see Equation (4)).

If the matrix L can be diagonalized, it is convenient to use the spectral norm for which Equation (8) reduces to:

$$\rho(L) \leq 1 \tag{9}$$

Matrices that are involved in finite-difference equations are usually band matrices. Despite this simplification, the determination of the eigensystem of L is usually difficult. For this reason, most analyses apply to simple test cases, such as the one of a scalar, linear partial-differential equation, with coefficients assumed constant in both time and space, and for some simple boundary conditions (generally periodic sometimes fixed, rarely

more sophisticated). In Section IIB, the stability analysis is developed for this simple case. Such analysis can be invalidated by either one of the following realities:

1. Dimensionality
2. Nonlinearity or time-dependence
3. Spacial dependence of the coefficients matrices
4. Complex boundary condition procedures

In Chapter IV, a form of instability due to spacial variation of the Jacobian matrices of the Euler equations (item 3) is discussed.

B. The Case of a Scalar Linear Equation

1. Generalities

If the Jacobian matrices A and B of the Euler equations commuted and were constant, the governing equations could be diagonalized into four scalar equations of the form:

$$u_t + au_x + bu_y = 0 \tag{10}$$

which is the first-order wave equation in two dimensions. Although these hypotheses are not satisfied by the Euler equations, the case of application of the numerical algorithm to Equation (10) is expected to reveal the essential properties of this algorithm. For this case, if linear boundary conditions[1] are assumed, it is convenient to rewrite the finite-difference equation (Equation (3)) in the following matrix form (derived in Appendix A):

$$A_x \otimes A_y(u^{n+1} - u^n) = -(B_x \otimes I_y + I_x \otimes B_y)u^n \tag{11}$$

---

[1]The analysis will be made for periodic or specified boundary conditions.

where the following definitions have been used

$$\left.\begin{aligned} A_x &= I_x + \nu_x C_x + \epsilon_i D_x \\ B_x &= \nu_x C_x + \epsilon_e D'_x \\ C_x &= (2\Delta x)\delta_x = \mathrm{Trid}(-1,0,1) \\ D_x &= \nabla_x \Delta_x = \mathrm{Trid}(-1,2,-1) \\ D'_x &= D_x \text{ or } D_x^2 , \quad \text{for second or fourth-order smoothing} \\ \nu_x &= a\Delta t/(2\Delta x) , \quad \text{half of a Courant number} \end{aligned}\right\} \qquad (12)$$

and A_y, B_y, C_y, D_y, D'_y, and ν_y are defined in a similar way. For a mesh containing $J \times K$ interior grid points, x-subscripted and y-subscripted matrices are of dimension $J \times J$ and $K \times K$, respectively. It is assumed that the $J \times K$ components of the solution vector u^n are conventionally ordered as follows:

$$u^n = (u^n_{11}, u^n_{12}, \ldots, u^n_{1K}, u^n_{21}, u^n_{22}, \ldots, u^n_{2K}, \ldots, u^n_{J1}, u^n_{J2}, \ldots, u^n_{JK})^t \qquad (13)$$

where as usual $u^n_{jk} = u(x_j, y_k, t_n)$. It is also assumed that this vector has been defined in such a way that $u^n = 0$ is the solution of the difference equation that one hopes to attain at the steady state. The homogeneity of Equation (11) results from this implicit convention. Finally, definitions and eigensystems of tridiagonal matrices for various boundary conditions are given in Appendix B.

Inverting Equation (11) yields the new equation:

$$u^{n+1} = Lu^n \qquad (14)$$

where $L = 1 - A_x^{-1}B_x \otimes A_y^{-1} - A_x^{-1} \otimes A_y^{-1}B_y$.

Let X and Y be two nonsingular matrices of sizes $J \times J$ and $K \times K$ respectively, to be chosen later. Upon defining

$$v^n = (X \otimes Y)^{-1} u^n \tag{15}$$

and making various applications of Equations (A3) through (A5) (see Appendix A), Equation (14) becomes:

$$v^{n+1} = \Lambda v^n \tag{16}$$

where the matrix Λ, which is defined by

$$\begin{aligned} \Lambda &= (X \otimes Y)^{-1} L (X \otimes Y) \\ &= I - X^{-1}A_x^{-1}B_x X \otimes Y^{-1}A_y^{-1}Y - X^{-1}A_x^{-1}X \otimes Y^{-1}A_y^{-1}B_y Y \\ &= I \quad \tilde{A}_x^{-1}\tilde{B}_x \otimes \tilde{A}_y^{-1} - \tilde{A}_x^{-1} \otimes \tilde{A}_y^{-1}\tilde{B}_y \end{aligned} \tag{17}$$

in which, for example,

$$\left.\begin{aligned} \tilde{A}_x &= X^{-1}A_x X \\ \tilde{B}_x &= X^{-1}B_x Y \end{aligned}\right\} \tag{18}$$

is similar to the matrix L.

Observe that for the simple case where no smoothing is applied ($\epsilon_e = \epsilon_i = 0$), the matrix Λ can be reduced to a diagonal form. For this, it suffices to choose X and Y to diagonalize the matrices C_x and C_y. Now, considering the general case where ϵ_e and ϵ_i are nonzero, and assuming the matrix Λ diagonalizable, an inspection of Equation (14) or (16) indicates that the solution u^n (alternately v^n) is bounded for arbitrary starting solution, if and only if the spectral radius of the matrix L (alternately Λ) is less than or equal to unity. It is desired to determine the conditions on ϵ_e, ϵ_i and θ under which this requirement is met for arbitrary values of the parameters ν_x and ν_y that control the time-step Δt ("unconditional stability"). This is done in the next two sections for some assumed boundary conditions.

2. The case of periodic boundary conditions

The case of periodic boundary conditions is of interest because it permits the development of a rigorous analysis of a pure initial-value problem. However, as one can anticipate by observing that the exact solution for this problem is given by

$$u(x,y,t) = u(x - at,\ y - bt,\ 0) \tag{19}$$

it does not provide a satisfactory test case for studying steady-state convergence. Nevertheless, the analysis of this case will be performed here as a guideline for the treatment of another case.

When periodicity conditions are applied, it is convenient to assume that u^n_{jk} is defined for all (positive or negative) integer values of j and k and that

$$u^n_{j+\nu J,\,k+\mu K} = u^n_{jk} \qquad (\nu \text{ and } \mu \text{ integers}) \tag{20}$$

The forward and backward shift operators (acting on either j or k) are then inverse of one another (see Appendix A); so are their matrix representations which thus, can be simultaneously diagonalized. As a consequence, the matrices in Equation (11) with the same subscript (x or y) which are linear combinations of their powers are also diagonalized by the same transformation. The (circulant) eigenvectors of these matrices are then chosen to construct X and Y (for details see Appendix B). This gives:

---

*Beam [16] originally showed the unconditional stability of the algorithm when applied (consistently) to the equation:

$$u_t + u_x + u_y = \epsilon(u_{xx} + u_{yy})$$

This analysis is extended here to the case where the dissipation terms may be fourth derivatives as well as second derivatives. Also, in the present analysis, these dissipation terms are differenced in a way that is not necessarily time-accurate.

$$\left.\begin{aligned} X_{mj} &= 1/\sqrt{J}\ \exp(mi\theta_j) \\ Y_{mk} &= 1/\sqrt{K}\ \exp(mi\theta_k) \end{aligned}\right\} \qquad (21)$$

where $\theta_j = 2\pi(j-1)/J$ $(j = 1,2,\ldots,J)$ and $\theta_k = 2\pi(k-1)/K$ $(k = 1,2,\ldots,K)$.[3] The eigenvalue of the j forward shift operator associated to the eigenvector $X_j = (X_{mj})$ is simply $X_{m+1,j}/X_{m,j} = \exp(i\theta_j)$. Similarly, the eigenvalue of the k forward shift operator associated to the eigenvector $Y_k = (Y_{mk})$ is $\exp(i\theta_k)$. The eigenvalues of other operators are obtained as linear combinations of the powers of $\exp(i\theta_j)$ or $\exp(i\theta_k)$. For example, the eigenvalues of the matrices A_x, B_x, C_x, D_x, and D_x' are given, respectively, by:

$$\left.\begin{aligned} a_j &= 1 + \theta\nu_x(ic_j) + \epsilon_i d_j \\ b_j &= \nu_x(ic_j) + \epsilon_i d_j \\ ic_j &= \exp(i\theta_j) - \exp(-i\theta_j) = 2i \sin\theta_j \\ d_j &= -\exp(-i\theta_j) + 2 - \exp(i\theta_j) = 2(1 - \cos\theta_j) \\ d_j' &= d_j \text{ or } d_j^2 \text{ — for second or fourth-order smoothing} \end{aligned}\right\} \qquad (22)$$

The eigenvalues of the matrices A_y, B_y, C_y, D_y, and D_y' are denoted by a_k, b_k, ic_k, d_k, and d_k', and are given by an equation similar to Equation (22).

---

[3] Remark: It is shown, in Appendix B, that X and Y are unitary, that is:

$$X^{-1} = X^* \text{ (adjoint of } X) = \bar{X}^t$$

$$Y^{-1} = Y^* \text{ (adjoint of } Y) = \bar{Y}^t$$

This is because X and Y represent (finite-dimensional, inverse) Fourier transforms.

With this choice of X and Y, the matrices A_x, B_x, A_y, and B_y in Equation (17) are all diagonal. As a result, the matrix Λ itself becomes a diagonal matrix with eigenvalues given by:

$$\lambda_{jk} = 1 - (a_j^{-1}b_j)a_k^{-1} - a_j^{-1}(a_k^{-1}b_k)$$
$$= \frac{a_j a_k - (b_j + b_k)}{a_j a_k} \tag{23}$$

This gives:

$$\lambda_{jk} = \frac{\alpha_{jk} + i\beta_{jk}}{\alpha_{jk} + i\beta_{jk}} \tag{24}$$

in which α_{jk}, β_{jk}, $\tilde{\alpha}_{jk}$ and $\tilde{\beta}_{jk}$ are real and given by:

$$\left.\begin{aligned} \alpha_{jk} &= (1 + \epsilon_1 d_j)(1 + \epsilon_1 d_k) - \theta^2 c_1 c_2 \\ \beta_{jk} &= \theta[(1 + \epsilon_1 d_k)c_1 + (1 + \epsilon_1 d_j)c_2] \\ \tilde{\alpha}_{jk} &= \alpha_{jk} - \epsilon_e(d_j' + d_k') \\ \tilde{\beta}_{jk} &= \beta_{jk} - (c_1 + c_2) \end{aligned}\right\} \tag{25}$$

where $c_1 = \nu_x c_j$ and $c_2 = \nu_y c_k$. The stability condition ($|\lambda_{jk}| \leq 1$) then becomes:

$$\tilde{\alpha}_{jk}^2 + \tilde{\beta}_{jk}^2 \leq \alpha_{jk}^2 + \beta_{jk}^2 \tag{26}$$

or

$$-2\alpha_{jk}\epsilon_e(d_j' + d_k') + \epsilon_e^2(d_j' + d_k')^2 - 2\beta_{jk}(c_1 + c_2) + (c_1 + c_2)^2 \leq 0$$

or

$$2\epsilon_e(d_j' + d_k')[(1 + \epsilon_1 d_j)(1 + \epsilon_1 d_k) - \theta^2 c_1 c_2] - \epsilon_e^2(d_j' + d_k')^2$$
$$+ 2\theta(c_1 + c_2)[(1 + \epsilon_1 d_k)c_1 + (1 + \epsilon_1 d_j)c_2] - (c_1 + c_2)^2 \geq 0$$

or

$$T + Q(c_1,c_2) \geq 0 \tag{27}$$

where

$$T = \varepsilon_e(d_j' + d_k')[2(1 + \varepsilon_i d_j)(1 + \varepsilon_i d_k) - \varepsilon_e(d_j' + d_k')] \tag{28}$$

and $Q(c_1,c_2)$ is a quadratic form in c_1,c_2 given by:

$$Q(c_1,c_2) = [2\theta(1 + \varepsilon_i d_k) - 1]c_1^2 + 2[\theta(2 + \varepsilon_i a_j + \varepsilon_i d_k) - \theta^2\varepsilon_e(d_j' + d_k') - 1]c_1c_2 + [2\theta(1 + \varepsilon_i d_j) - 1]c_2^2 \tag{29}$$

Note that if no smoothing is applied ($\varepsilon_e = \varepsilon_i = 0$), T vanishes identically, while $Q(c_1,c_2)$ reduces to $(2\theta - 1)(c_1 + c_2)^2$. From this, one concludes that the Euler explicit method ($\theta = 0$) is unconditionally unstable for this case, while the trapezoidal time-differencing method ($\theta = 1/2$) as well as the Euler implicit method ($\theta = 1$) are both unconditionally stable, as well known [2,3].

Now consider again the general case where ε_e and ε_i are nonzero. Since the wave speeds a and b as well as the time-step Δt ought to be arbitrary, the parameters ν_x and ν_y and consequently c_1 and c_2 must be considered as free parameters. The condition expressed in Equation (27) then breaks into two:

$$T \geq 0 \tag{30}$$

$$Q(c_1,c_2) \geq 0 \tag{31}$$

Equation (31) will be examined first. In view of Equation (29) it is apparent that its satisfaction requires, in particular, that the coefficients of c_1^2 and c_2^2 in $Q(c_1,c_2)$ be nonnegative. This gives the following necessary conditions:

$$2\theta(1 + \epsilon_i d_k) \geq 1$$
$$2\theta(1 + \epsilon_i d_j) \geq 1 \qquad (3.')$$

The definitions of d_j and d_k (Equation (22)) indicates that these eigenvalues are positive (dissipation), but tend to zero (for fixed values of j and k) in the limit of a mesh refinement $\Delta x, \Delta' \to 0$ (or $J,K \to \infty$). Hence, Equation (32) requires that

$$\theta \geq \frac{1}{2} \qquad (33)$$

Sufficiency is obtained by enforcing, also, that $Q(\epsilon_1, \epsilon_.)$ be nonfactorable. This gives the following condition:

$$[\theta(2 + \epsilon_i d_j + \epsilon_i d_k) - \theta^2 \epsilon_e (d_j' + d_k') - 1]^2$$
$$- [2\theta(1 + \epsilon_i d_k) - 1][2\theta(1 + \epsilon_i d_j) - 1] \leq 0 \qquad (34)$$

Expanding above quartic form in θ would reveal that θ^2 can be factored in it. For this reason, after a few simplifications, a condition equivalent to Equation (34) can be obtained in the form:

$$g_{jk}(\theta) \leq 0 \qquad (35)$$

where $g_{jk}(\theta)$ is a quadratic form in θ given by:

$$g_{jk}(\theta) = \theta^2 \epsilon_e{}^2 (d_j' + d_k')^2 - 2\theta\epsilon_e (d_j' + d_k')(\epsilon_i d_j + \epsilon_i d_k + 2)$$
$$+ 2\epsilon_e (d_j' + d_k') + \epsilon_i{}^2 (d_j - d_k)^2 \qquad (36)$$

Equation (35) must be enforced for all values of j and k and for the particular value of θ which corresponds to the chosen method ($\theta = 1/2$ for trapezoidal time-differencing, $\theta = 1$ for the Euler implicit method). In this way, a condition on ϵ_e and ϵ_i results. Before expliciting this condition, assume momentarily that ϵ_e, ϵ_i, and θ have been chosen to satisfy

precisely this condition. Then, certainly, for some arbitrarily chosen values of j and k, it is true that

$$g^*_{jk} \leq g_{jk}(\theta) \leq 0 \tag{37}$$

if one defines $g^*_{jk} = \text{Min}\, g_{jk}(\theta)$. But $g_{jk}(\theta)$ achieves its minimum for $\theta = (\varepsilon_i d_j + \varepsilon_i d_k + 2)/\{\varepsilon_e(d'_j + d'_k)\}$ so that:

$$\begin{aligned} g^*_{jk} &= -(\varepsilon_i d_j + \varepsilon_i d_k + 2)^2 + 2\varepsilon_e(d'_j + d'_k) + \varepsilon_i{}^2(d_j - d_k)^2 \\ &= -4\varepsilon_i{}^2 d_j d_k - 4(\varepsilon_i d_j + \varepsilon_i d_k + 1) + 2\varepsilon_e(d'_j + d'_k) \\ &= 2[\varepsilon_e(d'_j + d'_k) - 2(1 + \varepsilon_i d_j)(1 + \varepsilon_i d_k)] \\ &= -\frac{2T}{\varepsilon_e(d'_j + d'_k)} \end{aligned} \tag{38}$$

where T is given by Equation (28). Since $\varepsilon_e(d'_j + d'_k) > 0$, this shows that Equation (30) is redundant if Equation (31) is enforced. Consequently, the satisfaction of Equations (33) and (35) constitutes the necessary and sufficient condition for unconditional stability.

Expliciting Equation (35) requires some further algebraic treatment which is presented in Appendices C and D. From this, if one lets $\mu = (\theta - 1/2)/\theta^2$, the following stability conditions result:

$$\theta(\varepsilon_e - \sqrt{\mu\varepsilon_e}) \leq \varepsilon_i < \theta(\varepsilon_e + \sqrt{\mu\varepsilon_e}) \tag{39a}$$

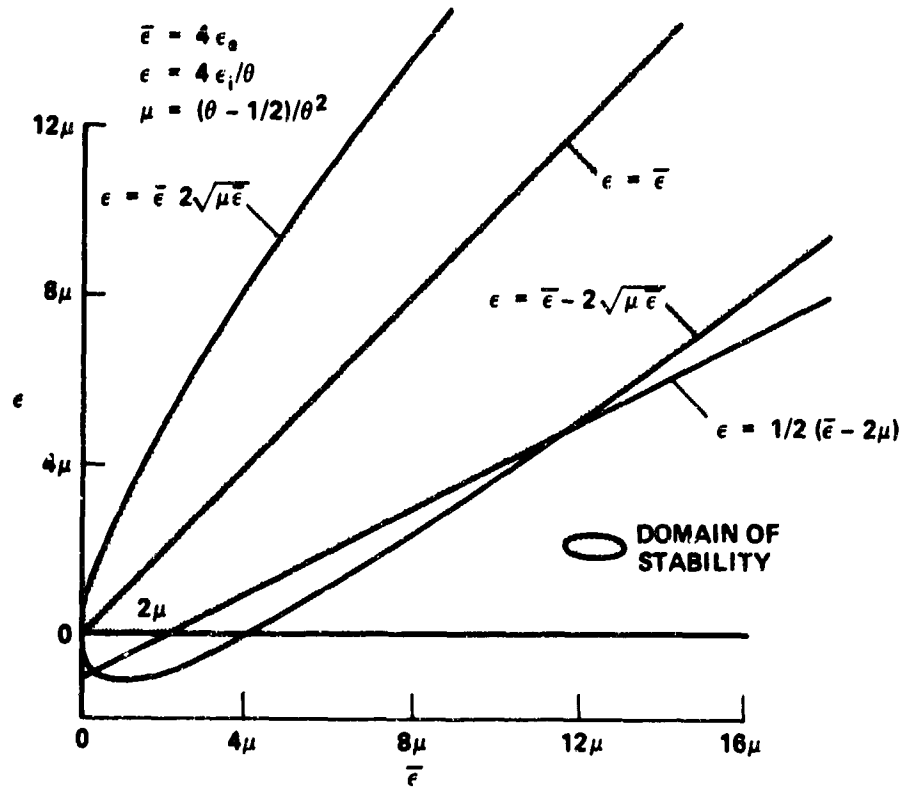
$$\frac{\theta}{2}(\varepsilon_e - 2\mu) \leq \varepsilon_i \tag{39b}$$

for the case of second-order smoothing (see Appendix C), and

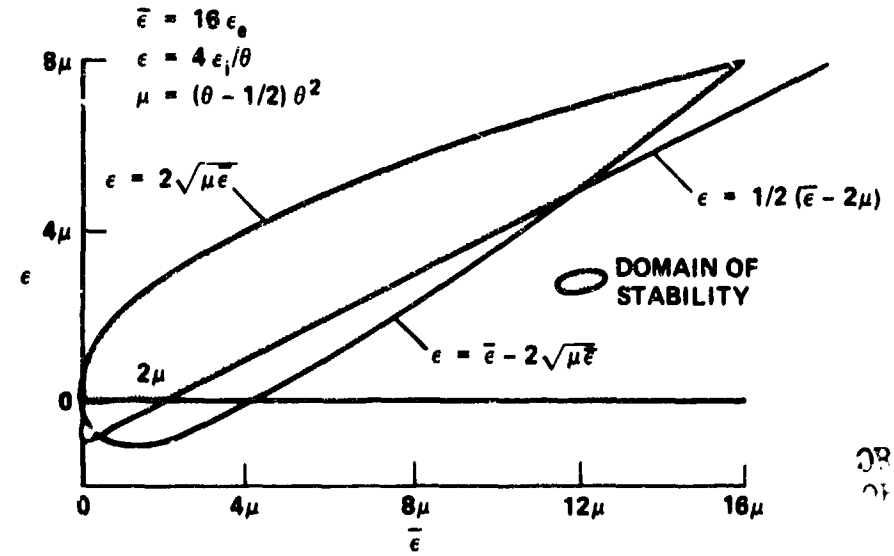
$$2\theta(2\varepsilon_e - \sqrt{\mu\varepsilon_e}) \leq \varepsilon_i \leq 2\theta\sqrt{\mu\varepsilon_e} \tag{40a}$$

$$2\theta\left(\varepsilon_e - \frac{\mu}{8}\right) \leq \varepsilon_i \tag{40b}$$

for the case of fourth-order smoothing (see Appendix D). The corresponding domains of "unconditional stability" are shown on Figure 1. Note that in



(a) Second-order smoothing applied explicitly.



(b) Fourth-order smoothing applied explicitly.

Figure 1.- Domain of unconditional stability for periodic boundary conditions.

case fourth-order smoothing is applied, this domain is bounded (Figure 1b). For the trapezoidal-time-differencing scheme, the domain reduces to the line $\varepsilon_i = \varepsilon_e/2$ in case second-order smoothing is applied, and collapses to the origin in case fourth-order smoothing is applied. These conservative results conflict with the authors' numerical experience (see Chapter III). This was attributed to the assumption of periodic boundary conditions which results in an inadequate model relaxation problem, as mentioned at the beginning of this section. For this reason, the case of specified boundary data is examined in the next section.

3. The case of specified boundary data

When the solution u^n is specified at the boundaries, the forward shift operator, Trid (0,0,1), and the backward shift operator, Trid (1,0,0), are no longer inverse of one another, and cannot be simultaneously diagonalized as they could for periodic boundary conditions. (In fact, they are both singular and in Jordan canonical form, or the transpose of it, with all the eigenvalues equal to zero.) As a consequence, in Equation (11), matrices with the same subscript (x or y) cannot be expressed as linear combinations of (positive or negative) integer powers of a unique (shift) operator, and cannot in general be simultaneously diagonalized. This is true, in general, for the matrices A_x and B_x together, and A_y and B_y together. (If second-order smoothing is applied, a case of exception occurs when $\varepsilon_i = \theta\varepsilon_e$.) However, approximate commutation of the right- and left-hand sides of Equation (11) occurs when the coefficients of the smoothing terms are either very small or very large compared to the coefficients of the convective derivative operators C_x and C_y. This situation corresponds to

$$\frac{\varepsilon}{\nu} \ll 1 \text{ or } \gg 1 \tag{41}$$

where ε is either ε_i or ε_e and ν is either ν_x or ν_y.

In practice, implicit smoothing is introduced in an attempt to keep the coefficient ε_e directly proportional to Δt, and still maintain unconditional stability. If one assumes a priori that this is possible,[4] and sets:

$$\left.\begin{aligned} \varepsilon_e &= \alpha_e \, \Delta t \\ \varepsilon_i &= \alpha_i \theta \, \Delta t \end{aligned}\right\} \tag{42}$$

for some constants α_e and α_i, Equation (41) becomes:

$$\frac{\Delta x}{a} \text{ and } \frac{\Delta y}{b} \ll 1 \text{ or } \gg 1 \tag{43}$$

The mesh spacing parameters Δx and Δy can certainly be considered as very small and so are $\Delta x/a$ and $\Delta y/b$, in general. However, if this model problem is of any relevance for the Euler equations, the wave speeds a and b should play the roles of the eigenvalues of the Jacobian matrices A and B. Some of these eigenvalues can eventually become very small in some regions of a transonic flow field (see Chapter IV), so that both limits in Equation (43) are of interest. If one makes the assumption that these extreme situations ((1) $\Delta x/a$ and $\Delta y/b$ very small, and (2) $\Delta x/a$ and $\Delta y/b$ very large) are those that produce the binding conditions for stability, one is tempted to analyze the asymptotic properties of the algorithm in these two limits. This is precisely what is done in the remaining part of this section.

---

[4] To bring a theoretical support to this assumption is precisely the motivation for this analysis.

Consider first the case where $\Delta x/a$ and $\Delta y/b$ are both very small. This case will be referred to as the case of large Courant numbers (ν_x and ν_y). Choose X and Y to be the transformations that diagonalize the matrices C_x and C_y respectively. These are given by (see Appendix B):

$$\left.\begin{aligned} X_{mj} &= \sqrt{2/(J+1)}\; i^m \sin m\theta_j \\ Y_{mk} &= \sqrt{2/(K+1)}\; i^m \sin m\theta_k \end{aligned}\right\} \tag{44}$$

where now $\theta_j = j\pi/(J+1)$ $(j = 1,2, \ldots,J)$, and $\theta_k = k\pi/(K+1)$ $(k = 1,2, \ldots,K)$. As for the case of periodic boundary conditions, these transformations are unitary (C_x and C_y are skew-symmetric), so that:

$$\left.\begin{aligned} X^{-1}_{mj} &= X^*_{mj} = \sqrt{2/(J+1)}(-i)^j \sin j\theta_m \\ Y^{-1}_{mk} &= Y^*_{mk} = \sqrt{2/(K+1)}(-i)^k \sin k\theta_m \end{aligned}\right\} \tag{45}$$

As a result of this choice, the matrices C_x and C_y can be written as follows:

$$\left.\begin{aligned} C_x &= X(iK_x)X^{-1} \\ C_y &= Y(iK_y)Y^{-1} \end{aligned}\right\} \tag{46}$$

where $K_x = \mathrm{Diag}(c_j)$ and $K_y = \mathrm{Diag}(c_k)$, in which now, $c_j = \cos\theta_j$ and $c_k = \cos\theta_k$. On their part, the matrices $\tilde{A}_x$ and $\tilde{B}_x$ in Equation (18) become:

$$\left.\begin{aligned} \tilde{A}_x &= I_x + i\theta\nu_x K_x + \varepsilon_i(X^{-1}D_x X) \\ \tilde{B}_x &= i\nu_x K_x + \varepsilon_e(X^{-1}D_x X) \end{aligned}\right\} \tag{47}$$

and the matrices $\tilde{A}_y$ and $\tilde{B}_y$ are given by similar equations. In these equations, matrices proportional to ν_x or ν_y are now considered as principal parts, and the other matrices as perturbations. Recall that as a particular case of application of a general result of perturbation theory (see, e.g.,

[17]), the first-order perturbations on the eigenvalues of a diagonal matrix (with distinct eigenvalues) are simply the diagonal elements of the matrix by which it is perturbed. In view of Equations (17) and (47), it appears that the off-diagonal elements of Λ are themselves first-order perturbations and thus contribute to the eigenvalues of Λ by terms that are at least second-order perturbations. Such perturbations are neglected in the remaining part of this derivation. In this approximation, the matrices $\dot{A}_x$ and $\dot{B}_x$, for example, become diagonal matrices with eigenvalues given by

$$\left.\begin{aligned} a_j &= 1 + \epsilon\nu_x i c_j + \epsilon_1 i_j \\ b_j &= \nu_x i c_j + \epsilon_e d'_j \end{aligned}\right\} \tag{48}$$

where $\dot{d}_j$ and $\dot{d}'_j$ are defined by

$$\left.\begin{aligned} \dot{d}_j &= (X^{-1} D_x X)_{jj} \\ \dot{d}'_j &= (X^{-1} D'_x X)_{jj} \end{aligned}\right\} \tag{49}$$

and terms of order $(\epsilon/\nu_x)^2$ are neglected with respect to one. But Equation (48) is analogous to Equation (22). Thus, unconditional stability is enforced by Equation (35), provided d_j, d_k, d'_j, and d'_k are replaced by $\dot{d}_j$, $\dot{d}_k$, $\dot{d}'_j$, and $\dot{d}'_k$, respectively, which act as "effective eigenvalues" of the smoothing operators at large Courant numbers. These effective eigenvalues are evaluated in Appendix E. In particular

$$\dot{d}_j = \dot{d}_k = 2 \tag{50}$$

(which is the average value of d_j or d_k); making the corresponding substitutions in Equation (35) yields, after some simplifications:

$$\epsilon_e^{0.2}(d'_j + d'_k) - 4\theta(2\epsilon_1 + 1) + 2 \geq 0$$

or

$$\varepsilon_i \geq \frac{\gamma\theta}{2}\left(\varepsilon_e - \frac{\mu}{\gamma}\right) \tag{51}$$

where again $\mu = (\theta - 1/2)/\theta^2$, and $\gamma = (1/4)\mathrm{Max}(\tilde{d}'_j + \tilde{d}'_k)$. In particular, if second-order smoothing is applied, $\tilde{d}'_j = \tilde{d}_j = \tilde{d}'_k = \tilde{d}_k = 2$, so that $\gamma = 1$ and Equation (51) becomes:

$$\varepsilon_i > \frac{\theta}{2}(\varepsilon_e - \mu) \tag{52}$$

If instead, fourth-order smoothing is applied, $\mathrm{Max}(d'_j)$ and $\mathrm{Max}(d'_i)$ converge to 8 in the limit of a mesh refinement (see Appendix E), so that $\gamma \to 4$ and Equation (51) becomes

$$\varepsilon_i \geq 2\theta\left(\varepsilon_e - \frac{\mu}{4}\right) \tag{53}$$

One can observe an analogy between Equations (52) and (53) and Equations (39b) and (40b).

Consider now the reverse situation where the Courant numbers ν_x and ν_y (alternately the wave speeds a and b) are small compared to the coefficients ε_e and ε_i of the smoothing terms. For this case, the matrices X and Y appearing in Equation (17), are chosen to be the orthogonal matrices ξ and η that diagonalize the (real symmetric) smoothing operators D_x and D_y (and also D'_x and D'_y), respectively. (The matrix ξ is explicited in Appendix B. It is found symmetric, but this property is not used here.) In this way, the smoothing operators D_x and D_y (and D'_x and D'_y), which produce the principal part of A, are represented by diagonal matrices, while the convective derivative operators $\nu_x C_x$ and $\nu_y C_y$, now sought as perturbations, are represented by the following similar matrices:

$$\left.\begin{aligned} \nu_x C_x &= \nu_x \xi^t C_x \xi \\ \nu_y C_y &= \nu_y \eta^t C_y \eta \end{aligned}\right\} \tag{54}$$

If second-order perturbations on the eigenvalues of λ are neglected, only the diagonal elements of C_x and C_y need to be retained. These are zero because C_x and C_y, and consequently $\hat{C}_x$ and $\hat{C}_y$ are skew-symmetric. Hence, first-order analysis and zeroth-order analysis of this case produce the same result. The latter one consists of treating the case of pure diffusion for which Equation (23) applies if, in the definitions that follow this equation (Equations (24) and (25)), the parameters, c_1 and c_3 are set equal to zero, and if θ_j and θ_k are defined as in Equation (45). Since these are the only modifications to bring to the analysis developed for periodic boundary conditions, the stability condition is given by Equation (30), or equivalently:

$$2(1 + \epsilon_1 d_j')(1 + \epsilon_1 d_k') - \epsilon_e(d_j' + d_k') \geq 0 \tag{55}$$

Enforcing that Equation (55) be satisfied for all values of j and k resulted in the following condition:

$$\left.\begin{aligned} \epsilon_1 &\geq \frac{1}{2}\left(\sqrt{\epsilon_e} - \frac{1}{2}\right) \quad \text{for} \quad \epsilon_e \geq 1 \\ \epsilon_1 &\geq \frac{1}{2}\left(\epsilon_e - \frac{1}{2}\right) \quad \text{for} \quad \epsilon_e \geq 1 \end{aligned}\right\} \tag{56}$$

for the case where second-order smoothing is applied, and in

$$\left.\begin{aligned} \epsilon_1 &\geq \sqrt{\epsilon_e} - \frac{1}{4} \quad \text{for} \quad \epsilon_e \geq 1 \\ \epsilon_1 &\geq 2\left(\epsilon_e - \frac{1}{8}\right) \quad \text{for} \quad \epsilon_e \geq 1 \end{aligned}\right\} \tag{57}$$

for the case where fourth-order smoothing is applied. (The derivation of Equations (56) and (57) is given in Appendix F).

In conclusion, an approximate definition of the domain of unconditional stability should be obtained by combining the conditions given in Equations (52) and (56) for the case where second-order smoothing is applied,

and in Equations (53) and (57) for the case where fourth-order smoothing is applied. These domains are shown in Figure 2 for the trapezoidal time differencing method ($\theta = 1/2$), and in Figure 3 for the Euler implicit method ($\theta = 1$). On the latter figure, the domains that have been previously obtained assuming periodic boundary conditions, are reproduced for comparison. It appears that in assuming specified boundary data instead of periodicity, results in less stringent stability limitations.

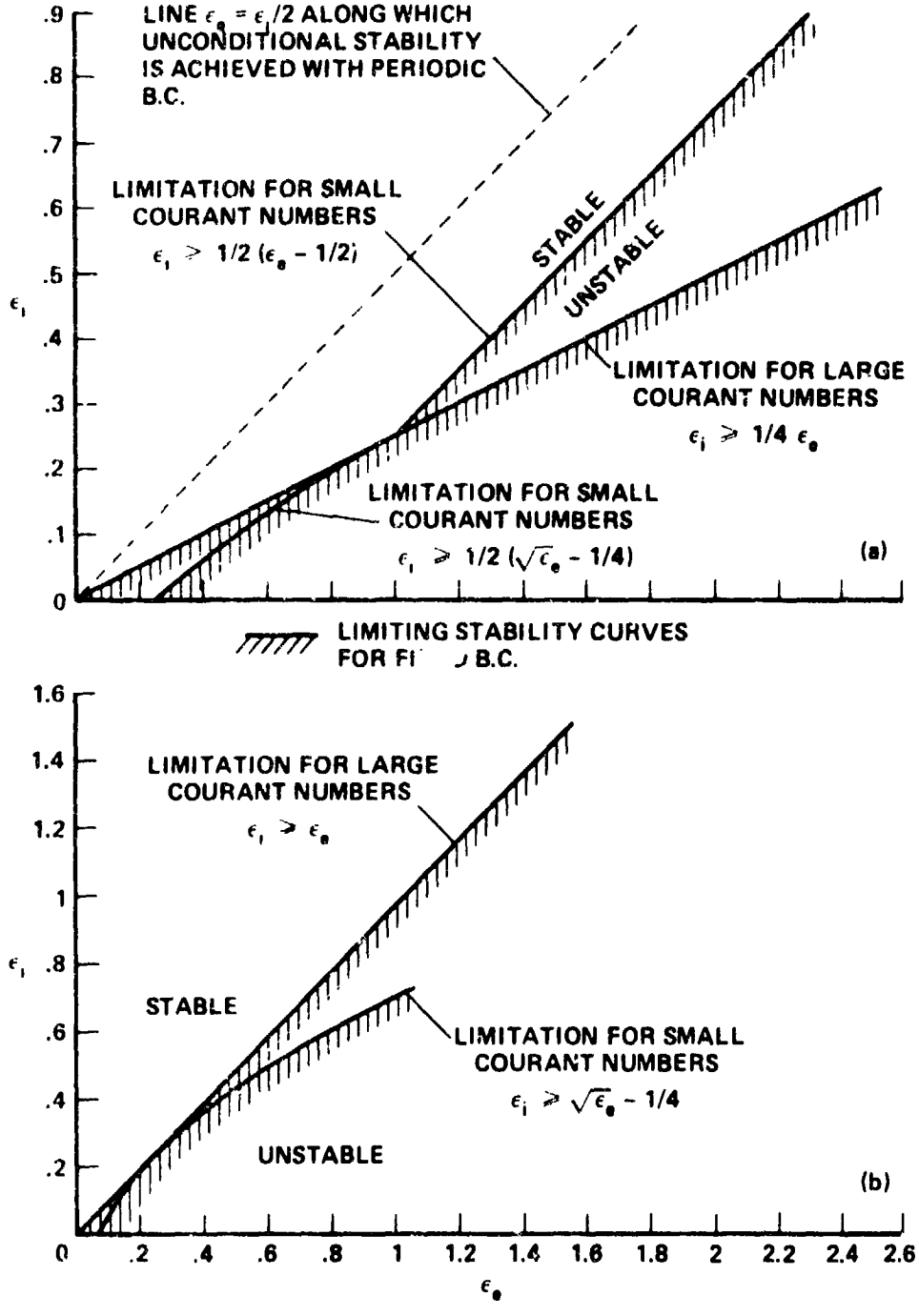
The key result of this analysis is that it suggests that the domain of unconditional stability in the $(\varepsilon_e, \varepsilon_i)$-plane is unbounded, allowing the use of arbitrary values of Δt and ε_e, provided ε_i is maintained sufficiently large. (This will be demonstrated in the next chapter by various numerical experiments that were conducted on a more complex problem.) In practice, it should be sufficient to let

$$\frac{\varepsilon_i}{\theta\varepsilon_e} = \frac{1}{2} \text{ or } 2 \tag{58}$$

when either second-order or fourth-order smoothing is employed. If now, ε_e is kept directly proportional to Δt, the inconsistency of the algorithm is removed, and this, theoretically, without violation of the property of unconditional stability, provided Equation (58) is enforced. This was not possible with the original formulation (Equation (2)).

C. Application: Sequence of Parameters

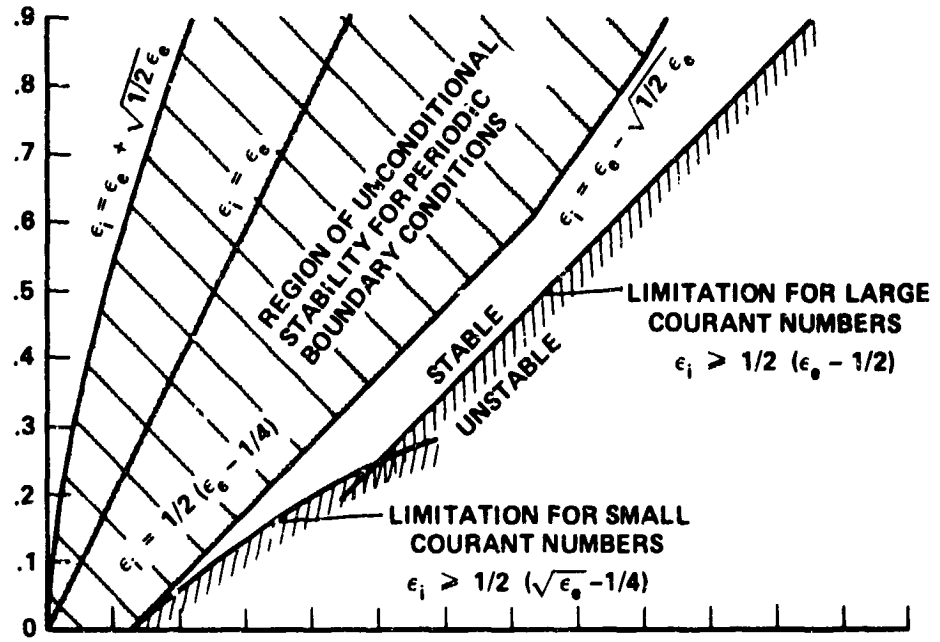
In Section IIB above, it was shown that in the modified differencing scheme, Equation (3), the numerical dissipation could be kept directly proportional to Δt. If this is done, the steady-state solution is independent of Δt, and much larger values of Δt can be taken without triggering



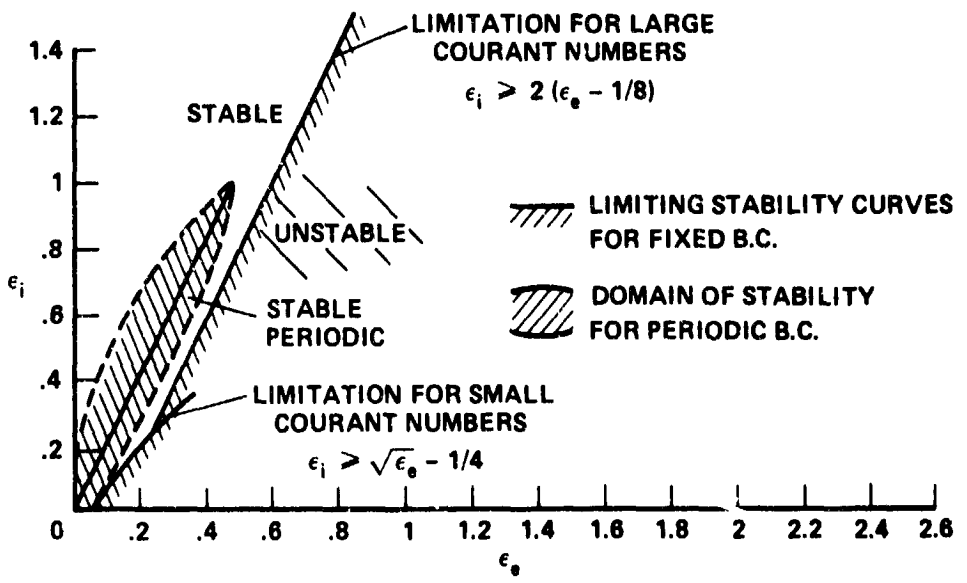
(a) Second-order smoothing applied explicitly.

(b) Fourth-order smoothing applied explicitly.

Figure 2.- Domain of unconditional stability of the trapezoidal time differencing scheme.



(a) Second-order smoothing applied explicitly.



(b) Fourth-order smoothing applied explicitly.

Figure 3.- Domain of unconditional stability of the Euler implicit method.

nonlinear instability. Large values of Δt, as well as a sequence of a small to a large Δt might thus be used to accelerate steady-state convergence. In this section, some motivations for using this technique are given. Consider again the simple model two-dimensional first-order wave equation, Equation (10). If the solution u is specified at the boundaries, and if numerical dissipation is not applied, the eigenvalues of the iteration matrix L can be obtained from an equation similar to Equation (23), and are given by:

$$\lambda_{jk} = \frac{1 - \nu_x\nu_y \cos\theta_j \cos\theta_k + i(\theta - 1)(\nu_x \cos\theta_j + \nu_y \cos\theta_k)}{1 - \nu_x\nu_y \cos\theta_j \cos\theta_k + \theta(\nu_x \cos\theta_j + \nu_y \cos\theta_k)} \tag{59}$$

where $\nu_x = ah/\Delta x$ and $\nu_y = bh/\Delta y$. The h^2-term which appears in the real parts of both numerator and denominator of λ_{jk} is the cross term that results from the approximate factorization of the left-hand side of the difference equation.

It is directly apparent that for the trapezoidal time differencing method ($\theta = 1/2$), the modulus of λ_{jk} is exactly equal to 1, whether approximate factorization is used or not. This means that no dissipative mechanism exists to permit the steady-state convergence of this method, unless smoothing is applied or boundary conditions are modified.

Consider now, the Euler implicit method ($\theta = 1$), and rewrite λ_{jk} as follows:

$$\lambda_{jk} = \frac{1}{1 + i\phi_{jk}} \tag{60}$$

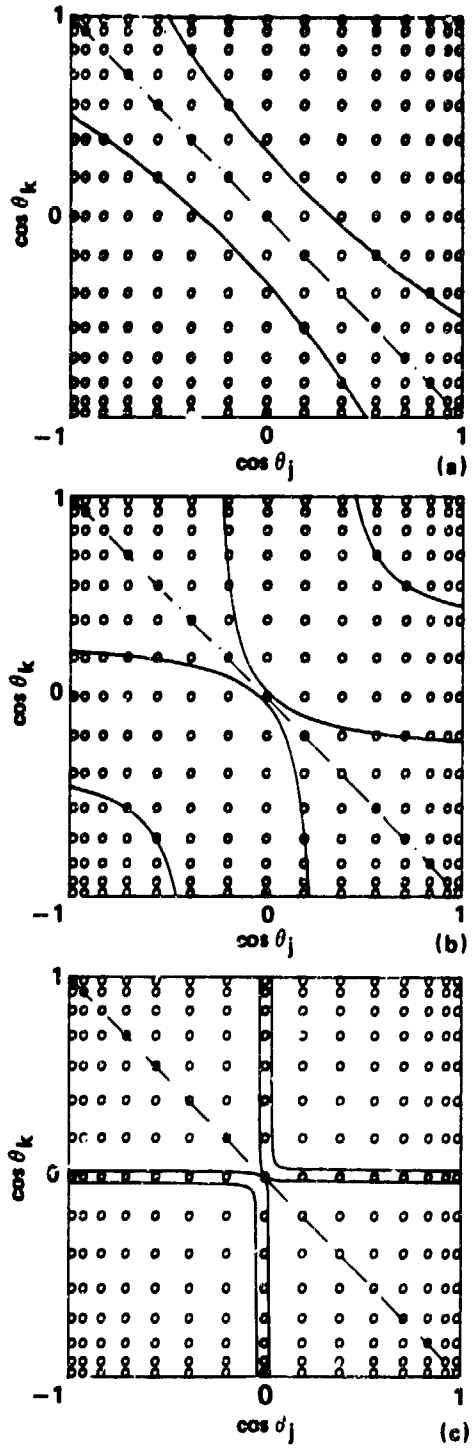
where

$$\phi_{jk} = \frac{h[(a/\Delta x)\cos\theta_j + (b/\Delta y)\cos\theta_k]}{1 - h^2(a/\Delta x)(b/\Delta y)\cos\theta_j \cos\theta_k} \tag{61}$$

Without approximate factorization, the h^2-term in the expression of ϕ_{jk} would disappear. Then $|\phi_{jk}|$ would be proportional to h, and $|\lambda_{jk}|$ would decrease with increasing h. In this case, the larger the time-step, the faster the convergence would be. However, a different conclusion can be drawn if approximate factorization is used. For that case, for given wave speeds (a and b), and frequency parameters (θ_j and θ_k), the modulus of λ_{jk} achieves a minimum when $|\phi_{jk}|$ is maximum, and this occurs when

$$h = \left| \frac{\Delta x}{a \cos \theta_j} \cdot \frac{\Delta y}{b \cos \theta_k} \right|^{1/2} \tag{62}$$

This shows that there exists an optimum time-step parameter h which not only depends on the wave speeds, a and b, and the mesh spacing parameters, Δx and Δy, but also on the frequency parameters, θ_j and θ_k. For small values of $\cos \theta_j$ and $\cos \theta_k$ (interpreted as low frequencies), or for small wave speeds a and b, a large time-step is desirable, as anticipated. However, for values of $|\cos \theta_j|$ and $|\cos \theta_k|$ of order 1 (interpreted as high frequencies), the Courant numbers $ah/\Delta x$ and $bh/\Delta y$ should themselves be of order 1 for a rapid reduction of the residuals. To illustrate this, the range of values of $\cos \theta_j$ and $\cos \theta_k$ for which $|\lambda_{jk}| \leq C$ (a constant taken to be 0.95), is represented in Figure 4 assuming $\Delta x/a = \Delta y/b$, for different values of the Courant number $\nu = ah/\Delta x = bh/\Delta y$. On this figure, the corners ($|\cos \theta_j| = 1$ or $|\cos \theta_k| = 1$) are interpreted as high frequency regions, and the neighborhood of the lines $\cos \theta_j = 0$ and $\cos \theta_k = 0$ as low frequency regions. The circles represent the values actually achieved by $\cos \theta_j$ and $\cos \theta_k$ for a mesh containing 15×15 interior grid points. The domain $|\lambda_{jk}| \leq C$ consists of two strips that converge towards the low frequency region when Δt increases.



(a) $\nu = 1$.

(b) $\nu = 10$.

(c) $\nu = 100$.

Figure 4.- The domain $|\lambda_{jk}| \leq C$ for a given Courant number.

From this, one concludes that a large time-step should be efficient at low frequencies, but also, that a sequence from a small to a large time-step should be efficient by not privileging any particular frequency band.

These concepts have served as a guideline for the numerical experimentation of the next chapter.

III. NUMERICAL EXPERIMENTATION ON STEADY-STATE CONVERGENCE

In this chapter, a model problem governed by the Euler equations is solved numerically to compare the iterative convergence properties of the modified algorithm, given by Equation (3), to those of the base algorithm, given by Equation (2).

A. Model Problem

A model transonic flow problem was selected to test the convergence of the modified differencing. In the past the transonic flow about a nonlifting biconvex airfoil with linearized boundary conditions has served as the prototype problem for relaxation algorithms and so this problem was used here. A variable grid with clustering was used to resolve flow-field gradients (see Figure 5), but the equations are solved on a uniform transform plane by introducing simple stretching transforms.

The solution procedure is as follows. The values of the conservative variables at interior points are first advanced from some starting solution, using either Equation (2) or Equation (3) with $h = \Delta t$. (The Euler implicit method ($\theta = 1$) is preferred here, to the trapezoidal time differencing method ($\theta = 1/2$) which is nondissipative (see Section IIC), because the emphasis, in these numerical tests is on steady-state efficiency.) Then, very simple boundary conditions are applied. Free-stream conditions are enforced at the inflow and upper boundaries. Along the body, the y component of velocity is obtained from thin airfoil theory:

$$v = U_\infty (dy/dx)_B \tag{63}$$

where U_∞ is the free-stream velocity, and $(dy/dx)_B$ the body slope which is a specified function of x. All other unknowns are obtained by

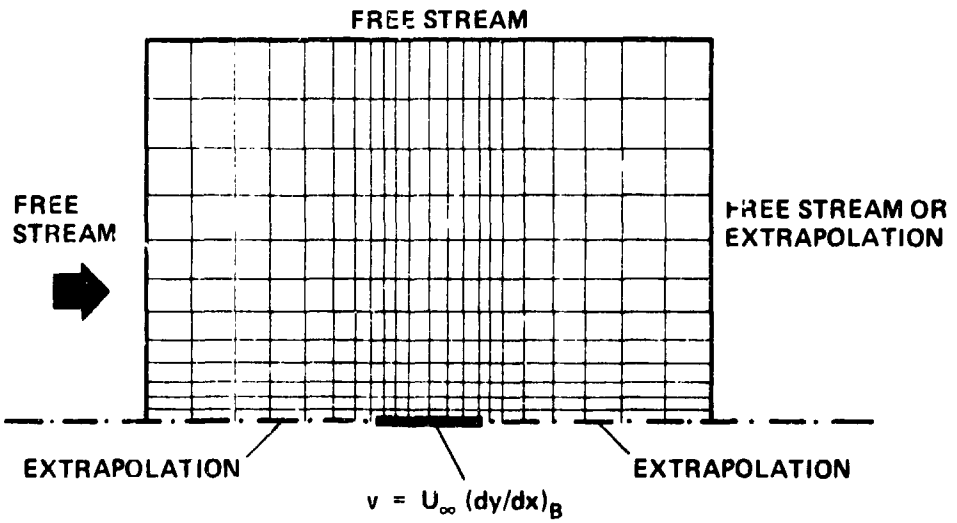


Figure 5.- Sketch of the computational domain.

(zeroth-order) extrapolations. Higher-order boundary conditions improve accuracy, but are deliberately avoided in this study because they can significantly degrade the stability and convergence properties that one would prefer to isolate.

B. Results

Results for a 10-percent-thick biconvex airfoil at $M_\infty = 0.84$ are shown in Figure 6, and compared to a potential solution by Holst [8]. It should be noted that a coarse grid and simplified boundary conditions have been used in order to test a variety of parameters. Much better solution accuracy is obtained by grid refinement and use of more accurate boundary conditions. Detailed solutions of this nature are available in [4].

The solution shown in Figure 6 was obtained using either Equation (2) with a nondimensional $\Delta t = 0.03$ and $\varepsilon_e = 0.03$, or Equation (3) with a nondimensional $\Delta t = 0.38$, $\varepsilon_e = 0.38$, and $\varepsilon_i = 2\varepsilon_e$. These values were each found to be close to optimum by a trial and error process. The convergence histories for both cases are shown in Figure 7 where root-mean-square residual error as well as the average difference between the converged and intermediate C_p distributions are indicated. Recall that the boundary conditions are applied in an explicit-like manner, which is expected to slow the more rapidly converging case, that is, Equation (3) more significantly than Equation (2), which uses more time-steps. Figure 7 shows that the modified differencing converges to steady state about 8 times faster than the original scheme. This experiment tends to verify the conclusion drawn from the model problem – a large value of Δt can be effective in achieving more rapid steady-state convergence.

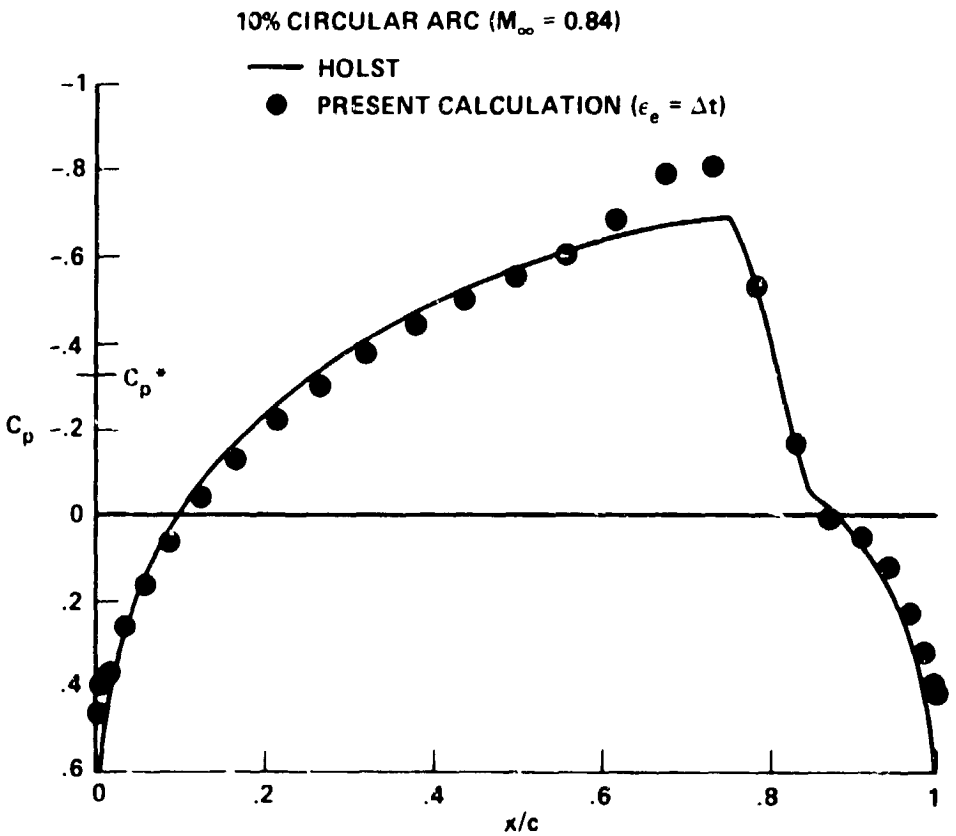


Figure 6.- Converged pressure distribution along the airfoil.

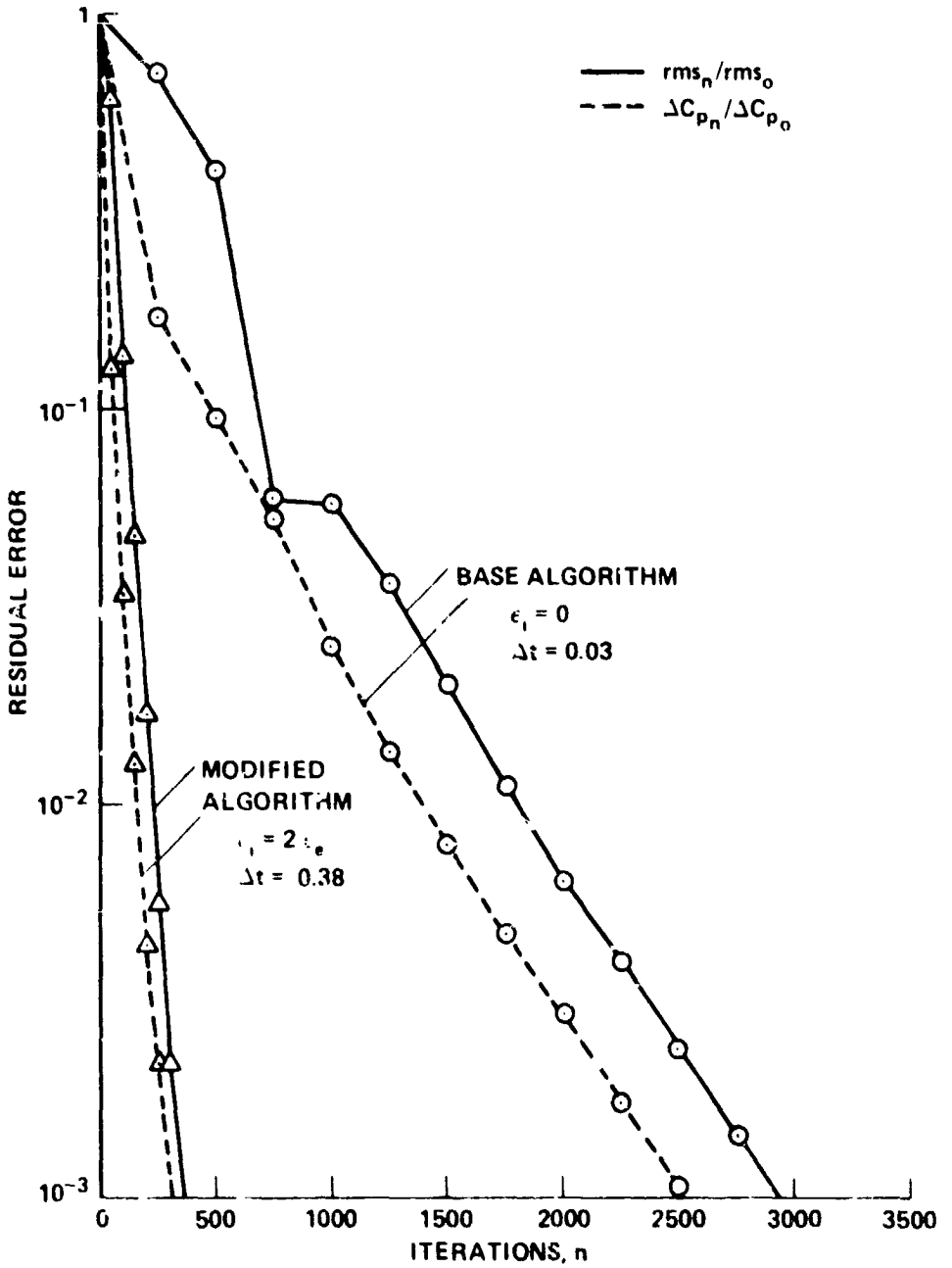


Figure 7.- Effect of using numerical dissipation implicitly (Euler implicit differencing scheme with $\epsilon_e = \Delta t$).

In these experiments, the starting solution was taken to be the one obtained after 25 applications of the base algorithm ($\Delta t = \varepsilon_e = 0.0$, $\varepsilon_i = 0$), with all the properties initialized to their free-stream values and gradually introducing the body by increasing with time the body slope $(dy/dx)_B$ from 0 to its correct value. In this way, impulsive starts were avoided.

It must be noted that the ratio of ε_i to ε_e can significantly influence the convergence rate. It was verified that for $\varepsilon_e = \Delta t$, and a single optimized time-step, this ratio could optimally be set equal to 2 (for the Euler implicit method), as indicated in Figure 8. For larger values of this ratio, the dissipation term added implicitly excessively stabilizes the transient behavior of the solution. For smaller values of this ratio, the coefficient ε_e, and consequently $\Delta t = \varepsilon_e$, must be reduced for stability (see Figure 3b), and this reduces the rate of convergence.

The effect of using a sequence of Δt is indicated in Figure 9. In order to simplify the optimization of the sequence, the following formula was used

$$\Delta t_n = \Delta t_1 + \left(\frac{n-1}{N-1}\right)^e (\Delta t_N - \Delta t_1)$$

where $n = 1,2,\ldots,N$ for a cycle of N time-steps, $e = 2$ in most experiments, and Δt_1 and Δt_N were optimized. The data show that a sequence of Δt is effective but not as much as one might expect. The sequence of 6 Δt is about 10 times more effective in steady-state convergence than the original scheme. The data shown are for optimum values of ε_e, ε_i, and Δt. In comparison to the same scheme based on using a single optimized time-step, a sequence of parameters saves 50 to 75 time-steps out of 150 to 250 time-steps,

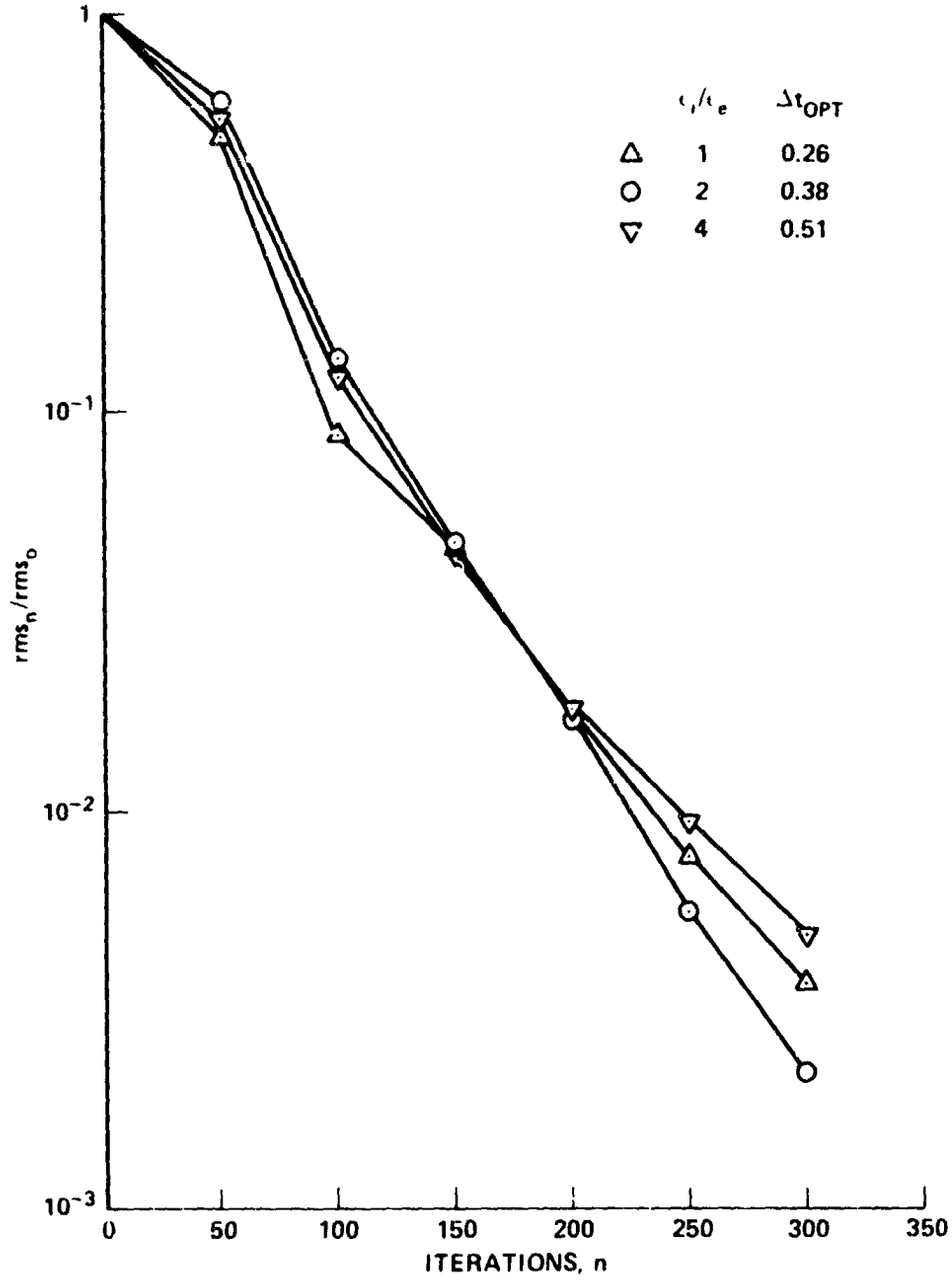


Figure 8.- Effect of the ratio of ϵ_i to ϵ_e (Euler implicit differencing scheme with $\epsilon_e = \Delta t$).

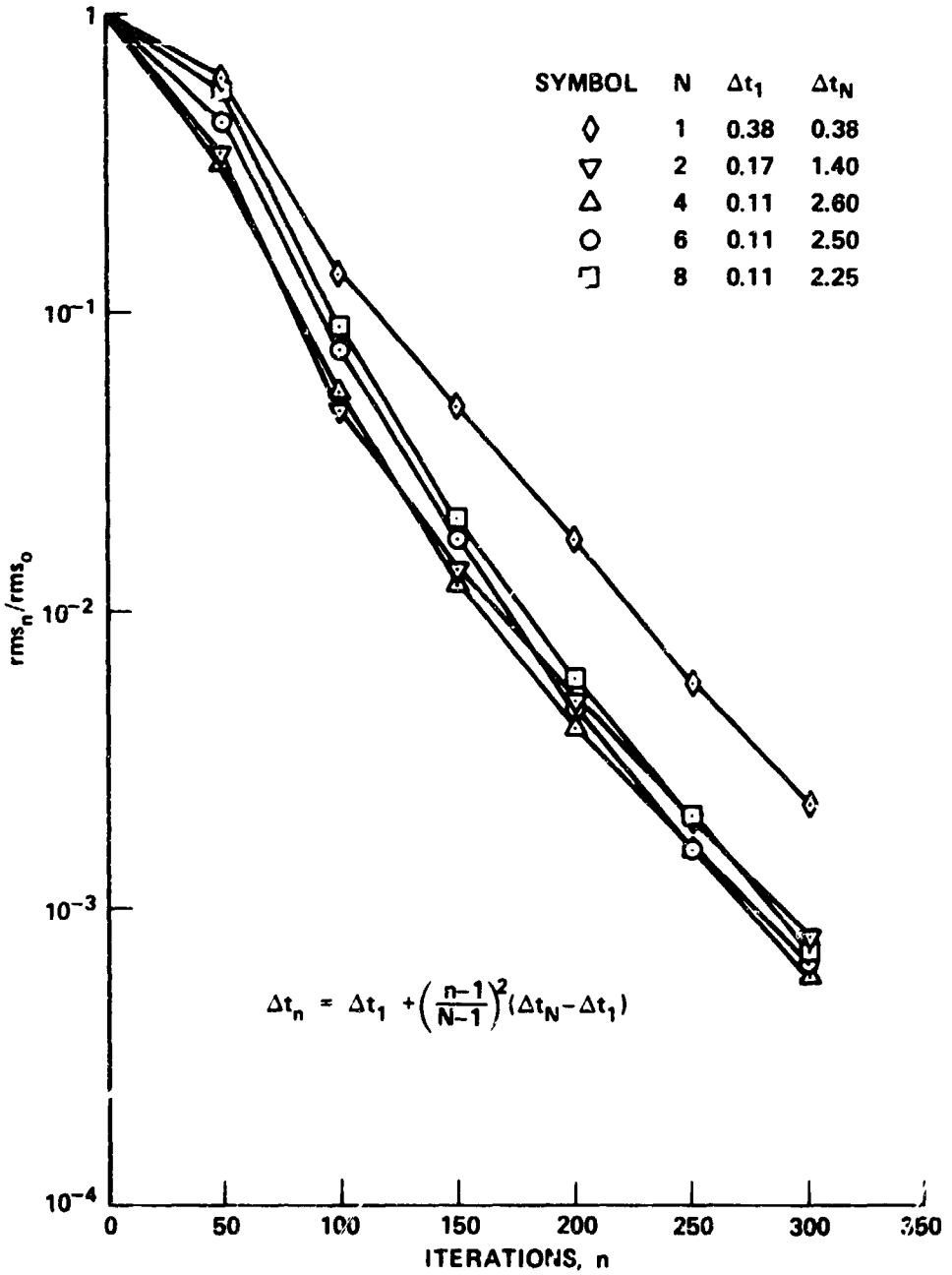


Figure 9.- Effect of using a sequence of time-steps (Euler implicit differencing scheme with $\varepsilon_e = \Delta t$ and $\varepsilon_i = 2\varepsilon_e$).

depending on what solution tolerance is desired. In this case, plottable accuracy is actually achieved after 150 time-steps with the most effective sequence. Again the model problem predicts the iterative convergence properties of the more complicated flow, and the use of a sequence of time-steps is also an effective way to accelerate steady-state convergence. In these tests, the optimum values of Δt_1 and Δt_N were found to correspond to a limit of stability. It was also noted, that at this limit, the average value of Δt for a cycle of N time-steps is about the same for all the cases shown in Figure 9.

Note that more sophisticated procedures for controlling various parameters would lead to better convergence rates. In particular, it was observed that a more rapid convergence could be obtained (for this problem) by setting $\varepsilon_i = 1.02\varepsilon_e$ (instead of $\varepsilon_i = 2\varepsilon_e$), $\varepsilon_e = \Delta t$ and choosing a sequence of time-steps that includes one or two that are sufficiently large for $(\varepsilon_i, \varepsilon_e)$ to fall in the unstable range. It also appears that the operational range $[\Delta t_1, \Delta t_N]$ should be optimized with the solution itself, that is, with the iteration counter n. It is most likely that these would produce better improvements. Nevertheless, they have been avoided here because of their lack of simplicity and generality.

Sensitivity in rate of convergence to nonoptimality is weaker if N is large. For example, Figure 10 shows that for a cycle of 6 time-steps, if Δt_1 and Δt_6 are set equal to half of their optimum values, Δt_1^* and Δt_6^*, the algorithm, over the first 300 steps, loses less than 20 percent in rate of convergence, and remains as efficient as it is for a single optimized time-step.

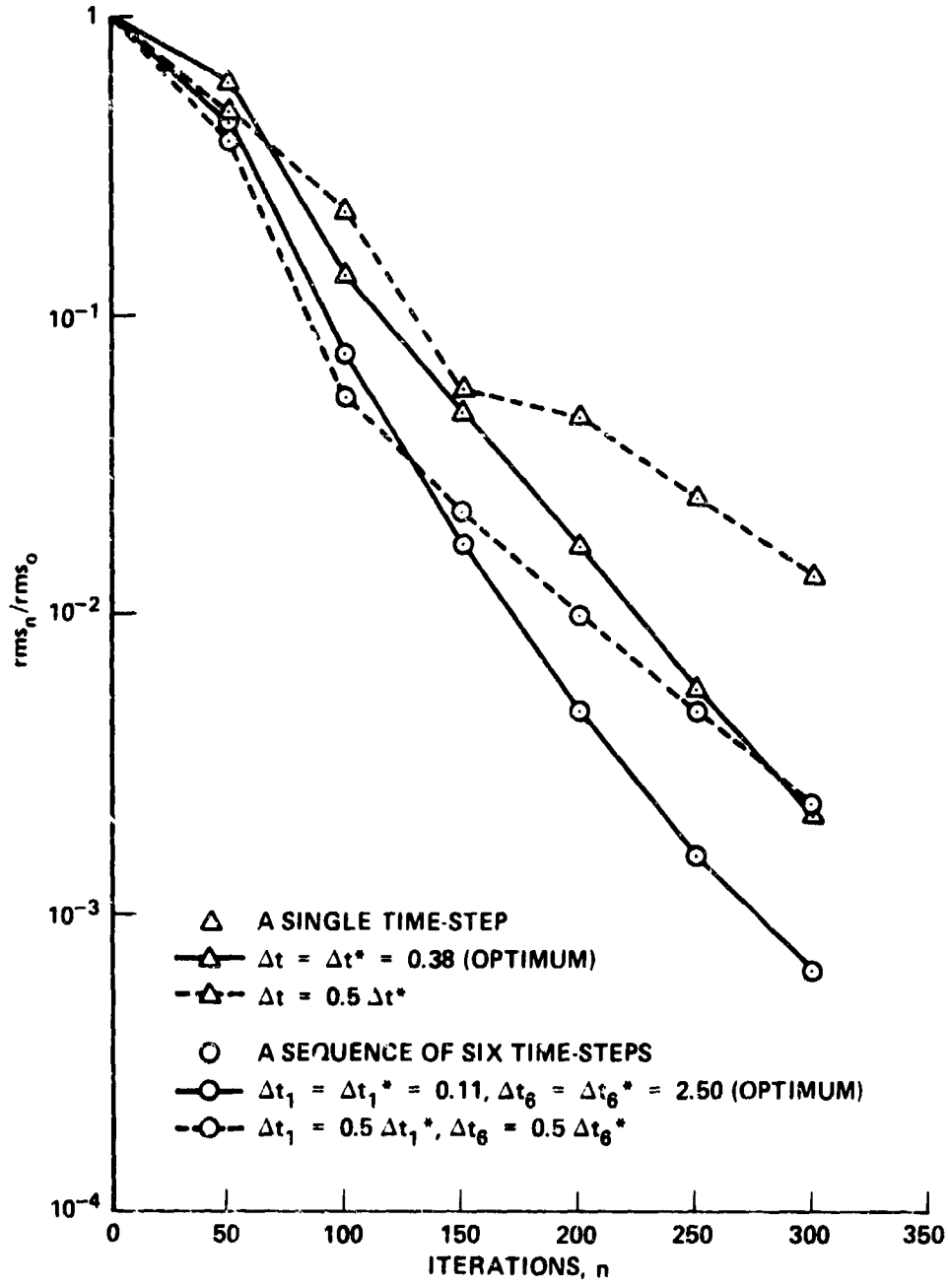


Figure 10.- Sensitivity to nonoptimality (Euler implicit differencing scheme with $\varepsilon_e = \Delta t$ and $\varepsilon_i = 2\varepsilon_e$).

The effect of varying the exponent e for fixed Δt_1 and Δt_6 is indicated in Figure 11. As e decreases, the average time-step increases and so does the rate of convergence until nonlinear instability occurs ($e = 1$).

Sensitivity in rate of convergence to free-stream Mach number M_∞ was also studied as indicated in Figure 12. Three cases were computed using the same sequence of time-steps; that is, the sequence was not optimized for M_∞. The data indicate that the implementation of implicit smoothing, and the use of large time-steps extend to subsonic and supersonic regimes as well as transonic regime.

The influence of the boundary conditions on the rate of convergence was also investigated. In this test, a sequence of six time-steps given by $\Delta t_n = 0.05$, 0.2, 0.45, 0.8, 1.25, and 1.8 was used, and $\varepsilon_e = \Delta t = \varepsilon_i/2$. This sequence was not optimal, but this is not believed to have had any importance. At first, a fully converged solution was obtained. The starting solution was then constructed by increasing by 5 percent the converged solution at interior points. The rate at which this disturbance could be eliminated, for some given boundary conditions, was then evaluated by computing the following estimate for the spectral radius of the iteration matrix:

$$\rho \sim \sqrt[50]{RMS_{300}/RMS_{250}}$$

where RMS_n is the root-mean-square residual error (the right-hand side of Equation (3)) after n applications of the algorithm. An estimate for the number n_{10} of time-steps required for a reduction of the residual errors by a factor of 10 was then computed according to the formula:

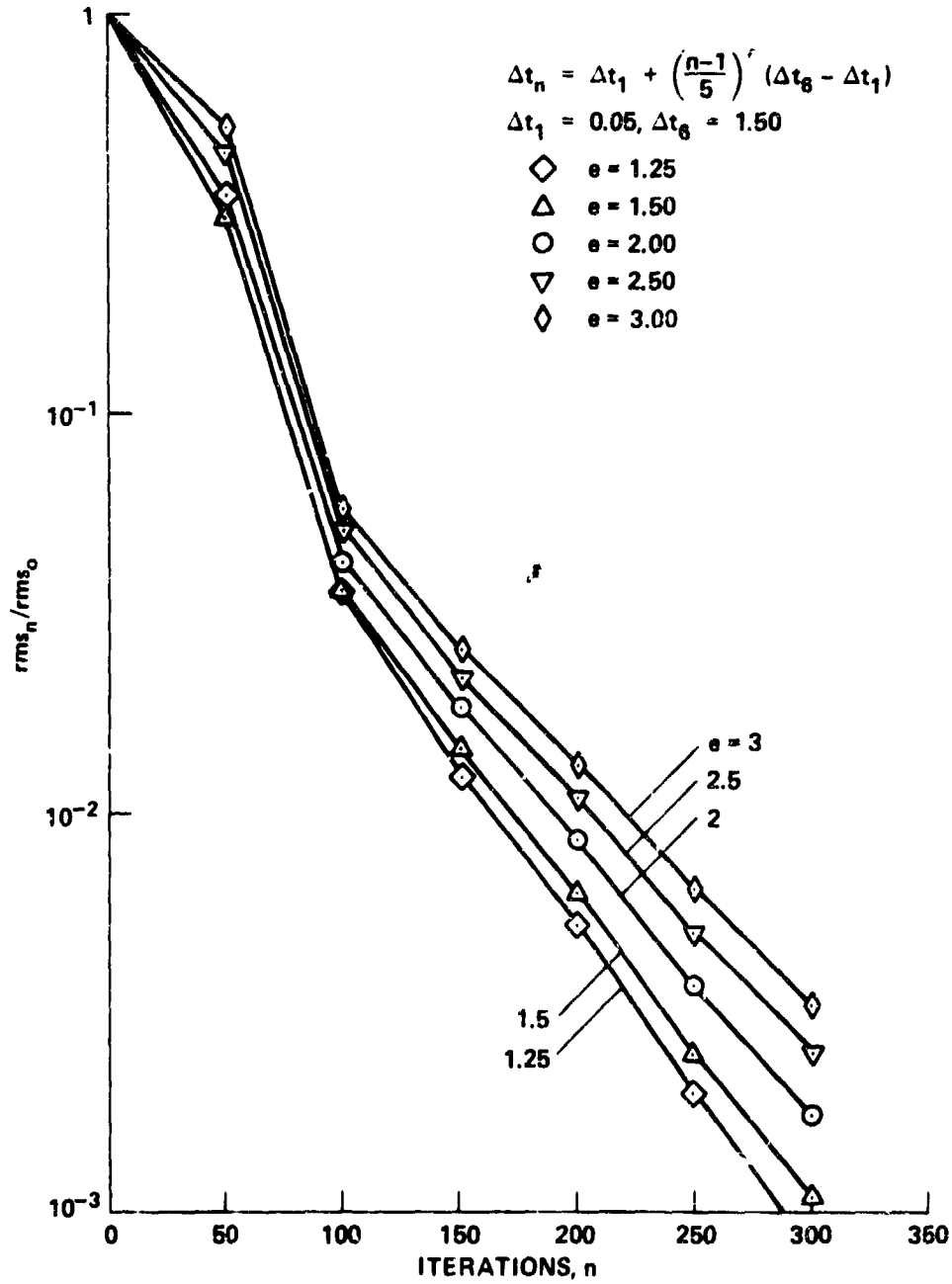


Figure 11.- Rate of convergence for various sequences of time-steps (Euler implicit differencing scheme with $\varepsilon_e = \Delta t$ and $\varepsilon_i = 2\varepsilon_e$).

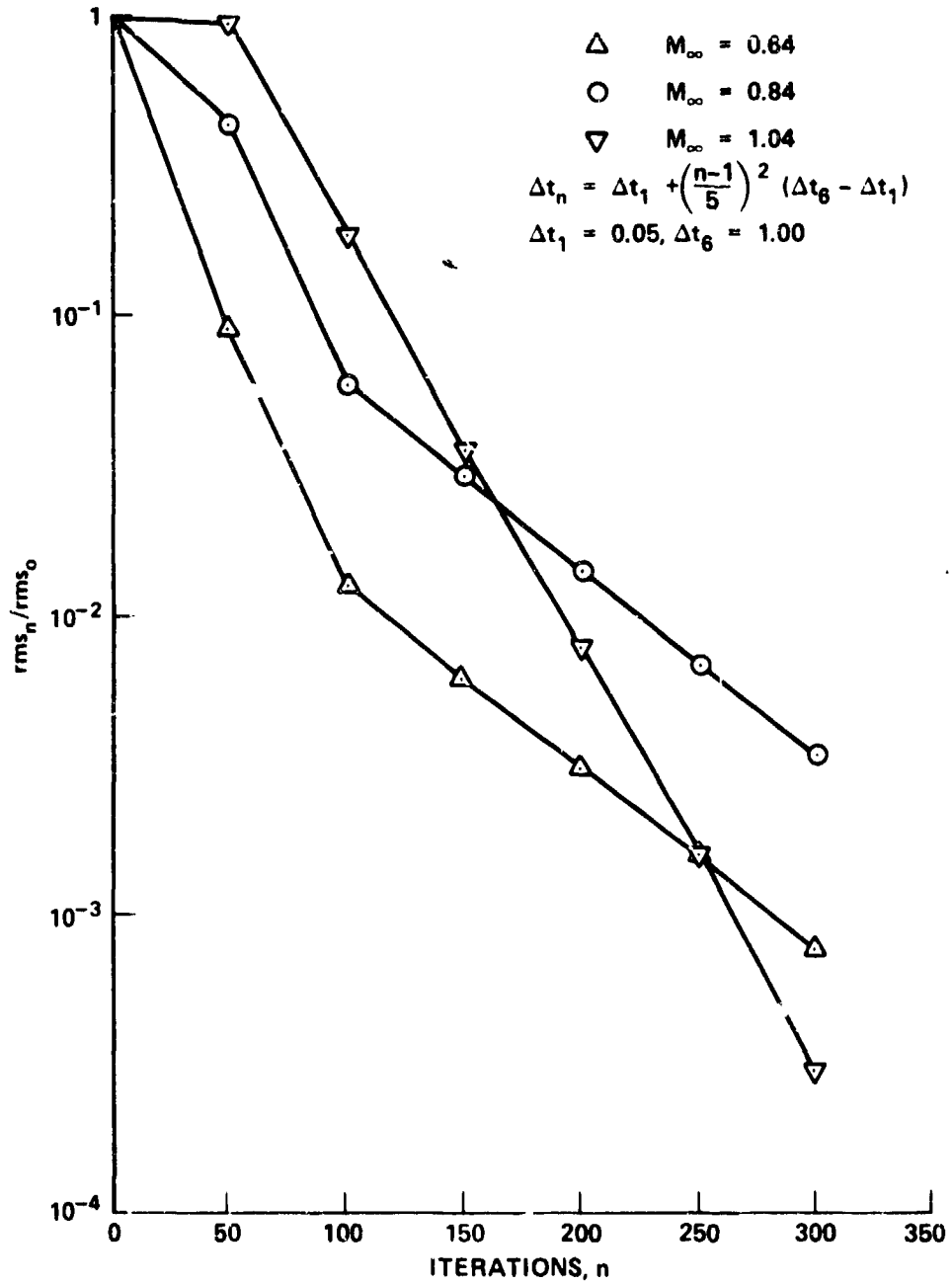


Figure 12.- Effect of the free stream Mach number (Euler implicit differencing scheme with $\varepsilon_e = \Delta t$ and $\varepsilon_i = 2\varepsilon_e$).

$$n_{10} = 1/\text{colog}_{10}(\rho)$$

Four tests were made. In these, the inflow and upper boundary values were fixed to free-stream conditions as normally done. However, the body boundary and outflow boundary values were either fixed to their converged values or variable, as permitted by the regular extrapolation procedure. The values of ρ and n_{10} obtained in the four cases are collected in Table 1. The results show that boundary conditions have a very strong effect on convergence properties. By fixing the body-boundary values (although this is impractical since it requires prior knowledge of the solution), the rate convergence doubled from what it was in the regular procedure. This favorable effect is even stronger if instead the outflow boundary values are fixed. This case is more practical for transonic flow applications where the properties at the outflow boundary can be fixed to free-stream values without significant degradation of the solution accuracy. If now all four boundaries are fixed, an improvement in rate of convergence by a factor of 4 is observed. This experiment indicates the strong dependence of the iterative convergence properties of the algorithm on boundary conditions, and opens a possible area of investigation for future work.

Table 1. Influence of the boundary conditions on the rate of convergence

Outflow boundary values	Body boundary values	ρ	n_{10}
Extrapolated[a]	Extrapolated[a]	0.98800	191
Extrapolated	Fixed	0.97633	96
Fixed	Extrapolated	0.96881	73
Fixed	Fixed	0.95064	45

[a]Regular procedure.

Finally, the effect of using only second-order smoothing, explicitly as well as implicitly, was investigated. For this test ε_e was set equal to $\Delta t/2$. The base algorithm ($\varepsilon_i = 0$, and $\varepsilon_e = \Delta t/2$) was found to operate optimally with $\Delta t = 0.22$. If instead, a sequence of time-steps is employed an improvement in rate of convergence by a factor of 2 or so is achieved, as shown in Figure 13. This test also shows that the use of second-order smoothing considerably increases the rate of convergence of the regular algorithm ($\varepsilon_i = 0$) itself. However, unacceptable losses in accuracy occur if this type of artificial dissipation is employed to calculate a flow field with a large change of gradient in the solution. Even in the simple biconvex airfoil calculation considered, the solution at the leading and trailing edges is noticeably degraded, although the shock wave is still adequately resolved.

One concludes in general that large Δt is very effective and that use of a sequence of Δt can be perhaps twice as good. The algorithm is not overly sensitive to nonoptimum features. However, better rates of improvement seem possible (e.g., the added effectiveness when only second-order dissipation is used, better boundary condition procedures).

One remarks that some ideas have proved effective as well in more complex flow calculations [4-6]. However, the additional sensitivity of the more complex flows to nonlinear instability forces the use of much smaller time-steps. Consequently, the improvement in rate of convergence is much less — typically a factor of 3 or 4 over the base algorithm.

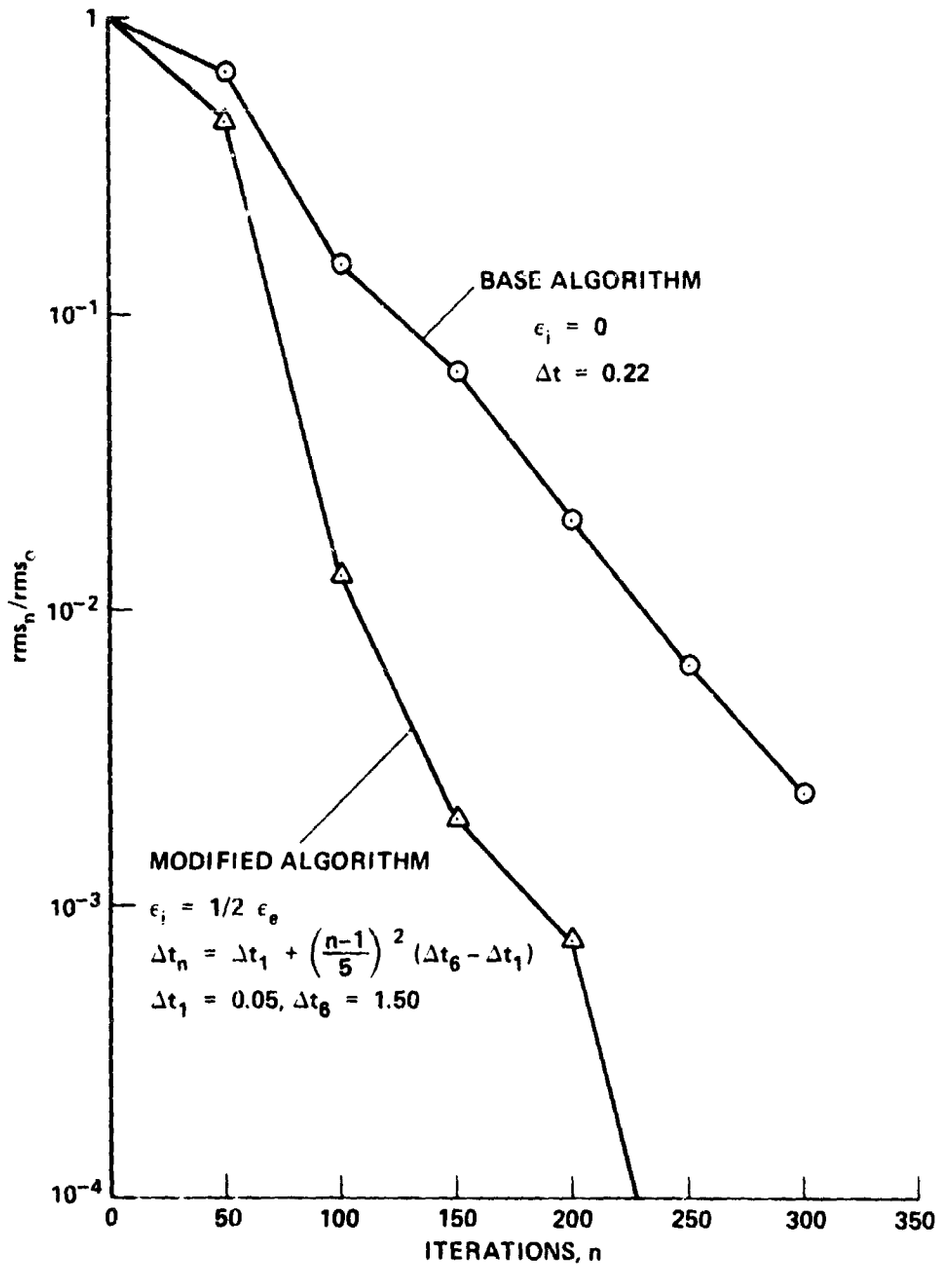


Figure 13.- Effect of using second-order smoothing explicitly as well as implicitly (Euler implicit differencing scheme with $\epsilon_e = \Delta t/2$).

IV. ON THE EFFECT OF SPACIAL VARIATION OF THE JACOBIAN MATRICES

In the computation of transonic as well as subsonic flows, the eigenvalues of the Jacobian matrices $A = [\partial E/\partial q]$ and $B = [\partial F/\partial q]$ are real, and of mixed sign. This is the reason for adopting a time-dependent approach, combined with the use of central space differencing which, in principle, produces purely imaginary eigenvalues for the convective derivative operators

$$\left.\begin{aligned} C_x &= (2\Delta x)\delta_x A \\ C_y &= (2\Delta y)\delta_y B \end{aligned}\right\} \tag{64}$$

The unconditional stability of the implicit algorithm, derived in Chapter II for a scalar, linear model equation, relies crucially on this property. In this chapter, the possibility of breakdown of this property for the Euler equations due to variable coefficients is examined.

A. Analysis

In this analysis, the matrix C_x is considered, in particular. A mesh containing $J\times K$ interior grid points is assumed, so that the dimension of the matrix C_x is $(J\times K\times p)^2$ for p dependent variables ($p = 4$ for two-dimensional flows). For convenience, it is here assumed that the $J\times K\times p$ components of the solution vector q are ordered as follows:

$$q = (q_{11}^t, q_{21}^t, \ldots, q_{J1}^t, q_{12}^t, q_{22}^t, \ldots, q_{J2}^t, \ldots, q_{1K}^t, q_{2K}^t, \ldots, q_{JK}^t)^t \tag{65}$$

where q_{jk} contains the p dependent variables evaluated at (x_j, y_k).

Then, making the simplifying assumption that the solution vector q is specified at the boundaries, permits us to write the matrix C_x as follows:

$$C_x = \text{BDiag}(C_{x,k}) \qquad (k = 1,2,\ldots,K) \tag{66}$$

where $C_{x,k}$ is the following $Jp \times Jp$ matrix:

$$C_{x,k} = \text{BTrid}(-A_{j-1,k},\ 0,\ A_{j+1,k}) \qquad (j = 1,2,\ldots,J) \tag{67}$$

in which A_{jk} is the $p \times p$ Jacobian matrix A evaluated at (x_j, y_k).

Clearly, if A was some symmetric constant matrix, the matrix C_x would be real skew-symmetric and would indeed have purely imaginary eigenvalues. However, one may question whether this property carries over to the general case where A is a nonsymmetric $p \times p$ matrix subject to appreciable variations from point to point, due to the nonuniformity of the mesh as well as the solution itself. The purpose of this analysis is precisely to bring some information about this question.

It first appears from Equations (66) and (67) that the eigenvalues of the matrix C_x are obtained by collecting those of the matrices $C_{x,k}$ together. For this reason, only one of these matrices will now be considered, with the subscript k omitted in what follows.

It is known that the Jacobian matrix A for the Euler equations can be diagonalized by a real transformation T, so that:

$$A = T\grave{A}T^{-1} \tag{68}$$

where $\grave{A} = \text{Diag}(a^m)$ and $m = 1,2,\ldots,p$. Explicit expressions for T and $\grave{A}$ can be found in [18]. For example, for a two-dimensional flow, if Cartesian coordinates are used: $a^1 = a^2 = u$, $a^3 = u + c$, and $a^4 = u - c$ where u is the x component of the velocity vector and c is the local speed of sound. For the same case, the eigenvalues of the Jacobian matrix B are $b^1 = b^2 = v$, $b^3 = v + c$, and $b^4 = v - c$, where v is the y component of the velocity vector.

Now, construct the following matrix

$$\mathcal{T} = \text{BDiag}(i^j T_j) \tag{69}$$

and perform on the matrix C_x(truly $C_{x,k}$ for some k) the following similarity transformation:

$$\begin{aligned} C_x' &= \mathcal{T}^{-1} C_x \mathcal{T} \\ &= \text{BDiag}[(-i)^j T_j^{-1}]\text{BTrid}(-A_{j-1},\ 0,\ A_{j+1})\text{BDiag}(i^j T_j) \\ &= i\sigma \end{aligned} \tag{70}$$

where

$$\sigma = \text{BTrid}(T_j^{-1}T_{j-1}\tilde{A}_{j-1},\ 0,\ T_j^{-1}T_{j+1}\tilde{A}_{j+1}) \tag{71}$$

It is desirable that all the eigenvalues of the matrix σ be real and for all those of the matrix C_x to be purely imaginary.

Note that if the flow variables are continuous, the matrices $T_j^{-1}T_{j-1}$ and $T_j^{-1}T_{j+1}$ depart from the identity matrix only by terms of $O(\Delta x)$. For this reason, one expects the eigenvalues of the matrix σ to be well represented by those of the following matrix:

$$\sigma' = \text{BTrid}(\tilde{A}_{j-1},\ 0,\ \tilde{A}_{j+1}) \approx \sigma \tag{72}$$

In making this approximation, one assumes the effect of variable coefficients to consist primarily of the variation of the eigenvalues of the matrix A_j rather than the variation of its eigenvectors. This assumption is made here, and the next step consists of rearranging the rows and the columns of the matrix σ' to collect the eigenvalues a_j^m (for given m) together. More precisely, for some nonsingular matrix P, which corresponds to a product of permutations, the matrix

$$\sigma'' = P^{-1}\sigma' P \tag{73}$$

which is similar to the matrix σ', becomes

$$\sigma'' = \text{BDiag}(\Gamma^m) \qquad (m = 1,2,\ldots,p) \tag{74}$$

where

$$\Gamma^m = \text{Trid}(a^m_{j-1},\ 0,\ a^m_{j+1}) \qquad (j = 1,2,\ldots,J) \tag{75}$$

Hence, the eigenvalues of the matrices σ' and σ'' are obtained by collecting those of the matrices Γ^m. These eigenvalues must be real for those of the matrix C_x to be purely imaginary. In this way, the analysis is reduced to the one of p independent scalar problems. Thus, one is lead to examine the eigenvalues of the matrix Γ^m for a particular value of m, and to omit this superscript in what follows. This is done in Appendix G, whose main results are repeated here without derivation.

It turns out, that for all the eigenvalues of the matrix Γ to be real, it suffices that the following condition holds:

$$a_j a_{j+1} \geq 0 \qquad (j = 1,2,\ldots,J-1) \tag{76}$$

This condition is met either when each eigenvalue a_j of the Jacobian matrix A has the same sign at all the grid points, or when it does change sign at one or more grid points but vanishes exactly at one or more grid points before changing sign. The condition given in Equation (76) is not, however, necessary for all the eigenvalues of Γ to be real. Nevertheless, this favorable result is most unlikely to be true if this condition is violated. To see this, another result of Appendix G is recalled. For this, define sequences of coefficients $\alpha^{(\nu)}_m$ and $\beta^{(\nu)}_m$ by the following recurrence formulas:

$$\left.\begin{aligned} \alpha^{(\nu)}_{m+1} &= \beta^{(\nu-1)}_m + a_{2m+1}a_{2m+2}\alpha^{(\nu)}_m \\ \beta^{(\nu)}_{m+1} &= \alpha^{(\nu)}_{m+1} + a_{2m+2}a_{2m+3}\beta^{(\nu)}_m \end{aligned}\right\} \tag{77}$$

where m is a natural integer and $\nu = 0,1,2,\ldots,m$, and the following conventions are adopted:

$$\left.\begin{aligned} \alpha_m^{(m)} &= \beta_m^{(m)} = 1 \\ \beta_m^{(-1)} &= 0 \end{aligned}\right\} \tag{78}$$

These definitions being made, it turns out that a necessary condition for all J eigenvalues of the matrix Γ to be real is that the coefficient $\alpha_m^{(\nu)}$, in case $J = 2m$, or $\beta_m^{(\nu)}$, in case $J = 2m + 1$, be positive for $\nu = 0,1,2,\ldots,m$. It is easy to calculate in particular $\alpha_m^{(o)}$ and $\beta_m^{(o)}$ which are given by:

$$\left.\begin{aligned} \alpha_m^{(o)} &= a_1 a_2 \ldots a_{2m} \\ \beta_m^{(o)} &= a_1 a_2 \ldots a_{2m+1} \left(\frac{1}{a_1} + \frac{1}{a_3} + \ldots + \frac{1}{a_{2m+1}}\right) \end{aligned}\right\} \tag{79}$$

In general, $\alpha_m^{(\nu)}$ and $\beta_m^{(\nu)}$ are polynomials of degree $2(m - \nu)$ of the coefficients a_j, and it is most unlikely that they all will remain positive if the condition given in Equation (76) is violated at one or more grid points. In particular, situations where $\alpha_m^{(o)}$, alternately $\beta_m^{(o)}$, is negative should be common. If this happens, the matrix Γ has an odd number of pairs of purely imaginary eigenvalues, say $i\,r_\ell$ and $-i\,r_\ell$ ($\ell = 1,2,\ldots,2g + 1$). To these correspond the real eigenvalues $-r_\ell$ and r_ℓ for the matrix C_x, half of which are negative. For trapezoidal time differencing and arbitrary time-step, or Euler implicit differencing and sufficiently large time-step, these real negative eigenvalues produce numerical instability unless they are balanced by a sufficient positive contribution coming from the smoothing operators. These unfavorable eigenvalues should, however, be of small modulus if one assumes that they are essentially determined by the entries

of the matrix C_x in the neighborhood of the point where the alternation of sign occurs (weak destabilizing effect). However, since the matrix C_x appears in the difference equation multiplied by Δt, the coefficients of the smoothing operators should themselves be kept proportional to Δt as required for consistency.

In conclusion, it appears that a particular form of instability due to variable coefficients may be triggered if central space differencing is used at a point where one of the eigenvalues of the Jacobian matrix A or B changes sign. However, use of numerical dissipation proportionally to Δt and in sufficient amount, should remedy this type of instability.

In the following section, some numerical examples of this phenomenon are presented for some scalar model problems.

B. Numerical Experiments on Scalar Model Equations with Variable Coefficients

In the numerical experiments, the trapezoidal time differencing method was used because this method is neutrally stable, that is nondissipative, for scalar linear problems with constant coefficients. In this way stability problems due to variable coefficients could be isolated more easily. In all the cases, scalar functions of the only two independent variables x and t were considered.

In the first test, the following problem was solved

$$\left.\begin{aligned} u_t + [a(x)(u-1)]_x - \varepsilon u_{xx} &= 0 \qquad (-1 \le x \le 1) \\ u(x,o) &= u_o(x) \quad [= 2\exp(-5x^2)] \end{aligned}\right\} \tag{80}$$

where the wave speed $a(x)$, chosen to be

$$a(x) = 4x/(1 + 27x^4)$$

had the sign of x. For $\varepsilon = 0$, this problem is a rarefaction wave problem. The characteristic curve in the (x,t) plane has the slope $dt/dx = 1/a(x)$ and is pointing outward of the domain of integration at the endpoints $x = \pm 1$. For this reason, specifying the solution at these boundary points would here be improper. Instead, in the numerical computation, (zeroth-order) extrapolation was used at these points. In this way, some small positive terms were introduced in the diagonal of the matrix C_x at the upper-left and lower-right corners. The numerical computation of the eigenvalues of the matrix C_x confirmed that these terms produce some positive contributions to the real parts of the eigenvalues (compared to the case where the solution is specified at the boundaries), and thus have a favorable stabilizing effect (outflow of residual errors). Despite this, with $\varepsilon = 0$, the trapezoidal time differencing method was found unstable when using a mesh with grid points located at $x_j = (j - 16.5)/15$ $(j = 1,2,\ldots,32)$, so that $a(x_{16})a(x_{17}) < 0$. This instability remained for values of ε less than 6×10^{-4}, or so. For larger values of ε, the numerical solution remained bounded and in fact convergent, at all the grid points, to the exact steady-state solution of this problem which is $u(x,\infty) = 1$. This steady-state solution was not altered by the smoothing terms in this ideal case where $u(x,\infty)$ is constant in x. It was also verified that a stable but not convergent (to steady-state) algorithm was obtained for $\varepsilon = 0$ by locating the grid points at $x_j = (x - 16)/15$ $(j = 1,2,\ldots,31)$ so that positive values of $a(x_j)$ were separated from negative ones by a true zero.

In the remaining tests, the following class of problems was considered:

$$\left.\begin{aligned} u_t + [\phi(u,x)]_x - \varepsilon u_{xx} &= [\phi(u_\infty(x),x)]_x \qquad (0 \le x \le 1) \\ u(0,t) &= u_\infty(0) \\ u(1,t) &= u_\infty(1) \\ u(x,0) &= u_0(x) \quad \text{(specified)} \end{aligned}\right\} \tag{81}$$

in which $u_\infty(x) = \exp(-5x^2)$, and the functional forms of ϕ and u_0 varied from case to case. The particular form of this equation was chosen anticipating that for sufficiently small ε, the stationary solution of this problem would approximate the function $u_\infty(x)$. This function was chosen rather arbitrarily but nonuniform so that $\varepsilon u_{xx} \neq 0$ at the steady state. Moreover, in the numerical computation, the term appearing on the right-hand side of the differential equation, that is the source term, was centrally differenced in the same way that the corresponding term of the left-hand side of the equation, that is, the flux term. In this way, the smoothing term εu_{xx} was entirely responsible for the discrepancies between $u(x,\infty)$ and $u_\infty(x)$.

Although, to the author's knowledge, specifying the solutions at the boundaries always leads to a well-posed problem for $\varepsilon > 0$ (assuming smooth data), this is not necessarily the case for $\varepsilon = 0$, as mentioned previously. In particular, for the latter case, the following inequalities

$$\left.\frac{\partial \phi}{\partial u}\right|_{x=0} \ge 0 \tag{82}$$

$$\left.\frac{\partial \phi}{\partial u}\right|_{x=1} \le 0 \tag{83}$$

should hold for the characteristic curve to point inward the domain at the boundary points. In all the cases that follow Equation (82) applied, but not necessarily Equation (83). This question will be discussed for specific examples.

Ten experiments were conducted on linear test equations that were obtained by letting the flux function ϕ be of the form $\phi(u,x) = a(x)u$. These experiments are defined in Table 2. For the first three cases, $a(x)$ was chosen strictly positive. For this reason, the implicit algorithm was found stable. However, adding artificial dissipation was found necessary to obtain a steady-state solution. For a rather small value of ε (Test No. 2) the numerical solution is very accurate as shown in Figure 14. For this case, the slightly improper boundary condition $u(1,t) = u_\infty(1)$, does not disrupt the stability of the algorithm (even for $\varepsilon = 0$), and does not seem to degrade the solution accuracy significantly (for $\varepsilon > 0$). This

Table 2. Numerical experiments for linear test equations

Test No.	$a(x)$	$u_0(x)$	ε	J	Comments
1	3.5-3x	exp(-5x)	0	52	Neutrally stable
2	3.5-3x	exp(-5x)	0.025	52	Convergent
3	3.5-3x	exp(-5x)	0.025	52	Convergent
4	1-3x	$\exp(-5x^2)$	0	52	Unstable
5	1-3x	$\exp(-5x^2)$	0	62	Neutrally stable
6	1-3x	exp(-5x)	0	62	Neutrally stable
7	1-3x	exp(-5x)	0.1	52	Convergent poor accuracy
8	1-3x	exp(-5x)	0.1	62	
9	1-3x	exp(-5x)	1.0	52	
10	1-3x	exp(-5x)	1.0	62	

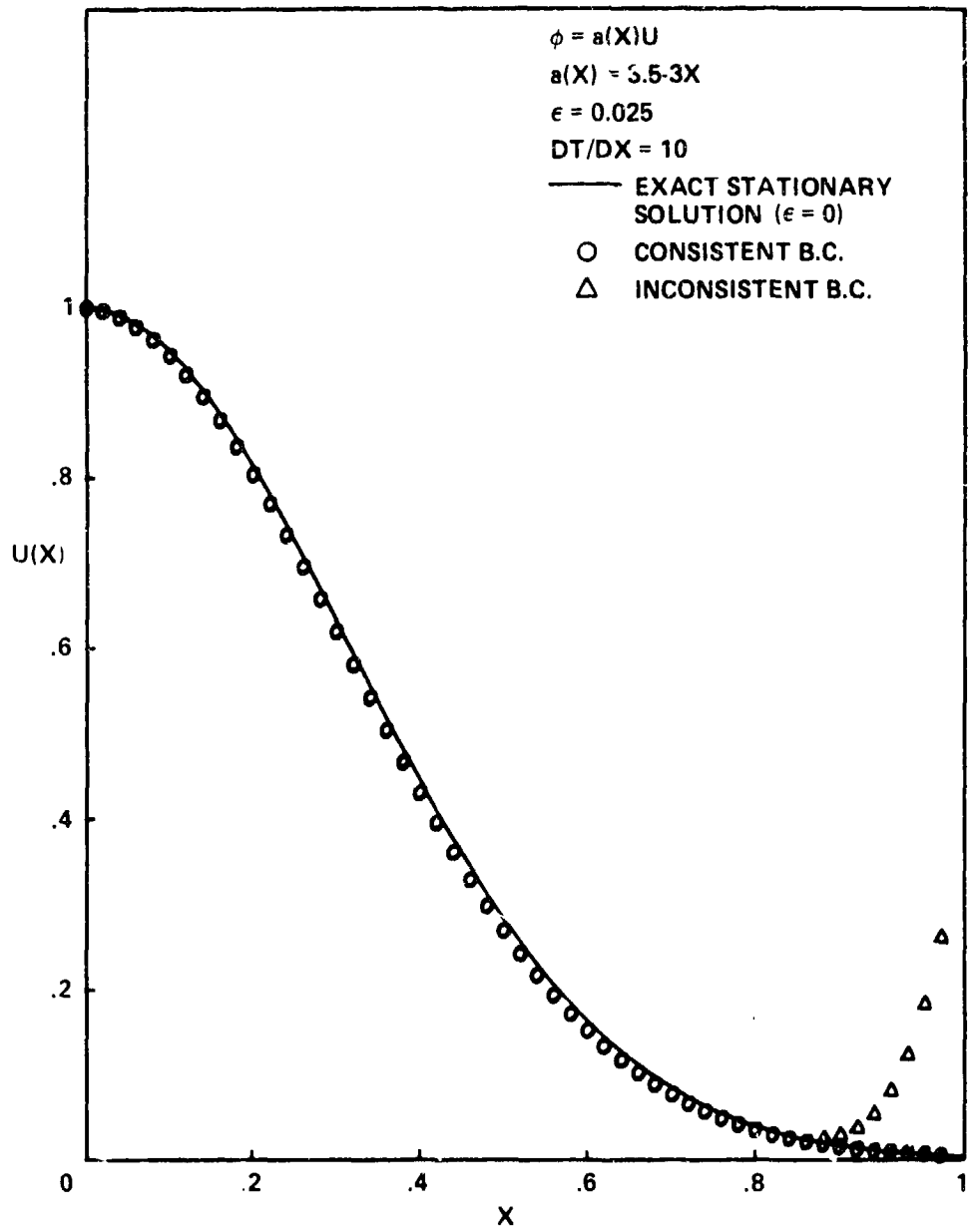


Figure 14.- Steady-state solution of a linear equation with a positive flux gradient and a small amount of artificial dissipation added.

shows that the well-conditioned nature of the algebraic system of difference equations is not necessarily equivalent to the well-posedness of the differential problem that one attempts to solve. The result of Test No. 3 is also indicated on Figure 14. It appears that in applying a quite erroneous boundary condition at $x = 1$ results a perturbation in the steady-state solution that is localized to a small neighborhood of this boundary. This is another aspect of the well-conditioned nature of this problem when $a(x) > 0$ everywhere.

In the experiments numbered 4-10, $a(x) = 1 - 3x$ so that the sign of $a(x)$ switched from positive to negative at $x = 1/3$. Without dissipation added ($\varepsilon = 0$), and a mesh of 52 grid points, the algorithm was found unstable even with the exact stationary solution for initial solution (Test No. 4). However if 62 grid points are used, the numerical solution remains bounded (Test No. 5). This is because positive and negative elements of the sequence $a_j = a(x_j)$ are separated by a true zero in the latter case. However, the solution does not converge to a steady state for a different initial solution (Test No. 6). If dissipation is now added ($\varepsilon > 0$), steady-state convergence is obtained but the accuracy of the steady-state solution is very poor, as shown on Figure 15 for Tests No. 8 and No. 10. One observes on this figure, that if ε is too small, say $\varepsilon = 0.1$, a peak appears in this solution near $x = 1/3$. The reduction of this peak requires excessive amounts of artificial dissipation which degrades severely the solution accuracy. An energy concept can explain the existence of this peak. For this purpose, define the following "energy" function:

$$E(t) = \int_0^1 u(x,t)dx \tag{84}$$

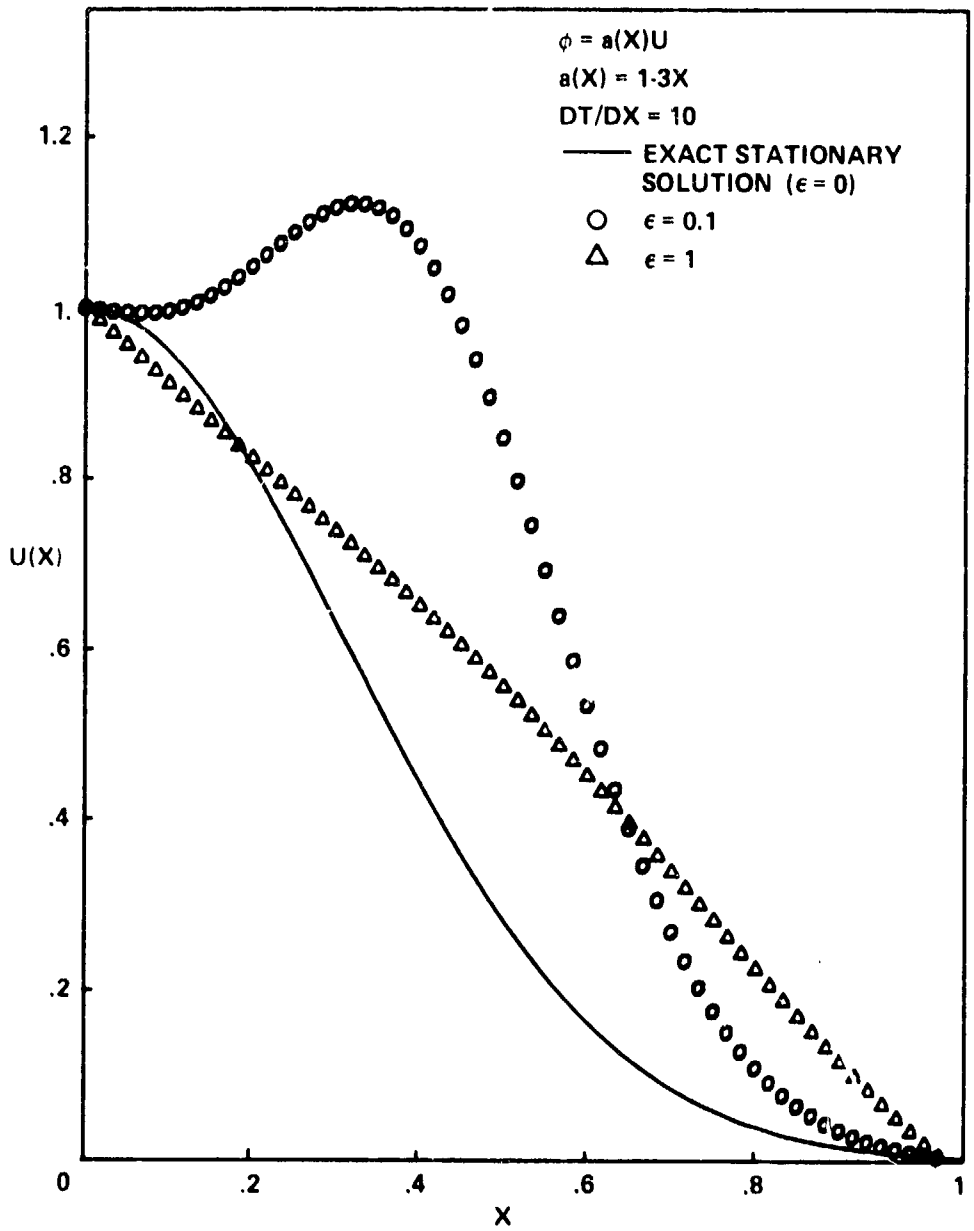


Figure 15.- Steady-state solution of a linear equation with a flux gradient changing sign and some artificial dissipation added.

and consider the case $\varepsilon = 0$ for which

$$E'(t) = \int_0^1 u_t(x,t)dx$$

$$= \int_0^1 \{\phi[u_\infty(x),x] - \phi(u,x)\}_x \, dx$$

$$= 0 \tag{85}$$

since $u(x,t)$ is constrained to equal $u_\infty(x)$ at $x = 0$ and $x = 1$. As a result, $E(t)$ is a constant (nondissipative phenomenon). Since, also, the characteristic curves are convergent at $x = 1/3$ (compression wave), this energy is accumulated at this point, in the limit $t \to \infty$. In fact, it can be obtained directly from the differential equation that

$$u(1/3,t) = u_\infty(1/3) + C \exp(3t) \tag{86}$$

for some constant C, while for $x \neq 1/3$, $u(x,t)$ converges to $u_\infty(x)$ in finite time. Hence this problem does not have a steady-state solution in the ordinary sense, unless the starting solution $u_0(x)$ is trivially chosen to be $u_\infty(x)$. These experiments indicate the difficulties encountered in attempting to achieve the stationary solution $u_\infty(x)$ by a viscosity method, when $a(x)$ changes sign.

Very similar results were obtained for nonlinear test equations. For these, six experiments, defined in Table 3, were conducted. Here, the starting solution $u_0(x)$ was obtained by adding to the stationary solution $u_\infty(x)$ a second-degree polynomial $q(x) = 5x(x - 1)$ that is zero at the boundary points and negative at interior points. In this manner, negative as well as positive values appeared in the initial solution. Here the flux function ϕ was chosen to depend on u only.

Table 3. Numerical experiments for nonlinear test equations

Test No.	$\phi(u)$	$u_o(x)-u_\infty(x)$	ε	Comments
11	$u + u^3/3$	$5x(x-1)$	0	Neutrally stable
12	$u + u^3/3$	$5x(x-1)$	0.025	Convergent
13	$u + u^3/3$	$5x(x-1)$	0.005	Convergent and very accurate
14	$u^2/2$	$5x(x-1)$	0	Unstable
15	$u^2/2$	$5x(x-1)$	0.025	Convergent
16	$u^2/2$	$5x(x-1)$	0.005	Convergent and very accurate

For the first three cases, the wave speed $a(u) = \partial\phi/\partial u = 1 + u^2 > 0$, and instability could not be triggered. Without dissipation added (Test No. 11), the solution does not converge (to steady state), but remains bounded. This is indicated by Figure 16 where an intermediate solution, obtained after 10^4 applications of the algorithm, is shown. On this figure, the values of u at the points $x_1, x_3, x_5, \ldots$ fall on a smooth curve, and so do the values of u at the points $x_2, x_4, x_6, \ldots$ but the two curves are distinct. This is because central space differencing does not couple the two subsequences $(u_1, u_3, u_5, \ldots)$ and $(u_2, u_4, u_6, \ldots)$. This is a known reason for requiring the use of artificial dissipation when a leap-frog type differencing is employed. However, if a small amount of dissipation is added, the numerical solution converges to a steady state. As an example, a very accurate solution obtained with $\varepsilon = 0.025$ (Test No. 12) is shown in Figure 17. For an even smaller value of ε (Test No. 13), exact stationary solution, $u_\infty(x)$, and numerical steady-state solution are indistinguishable to plottable accuracy.

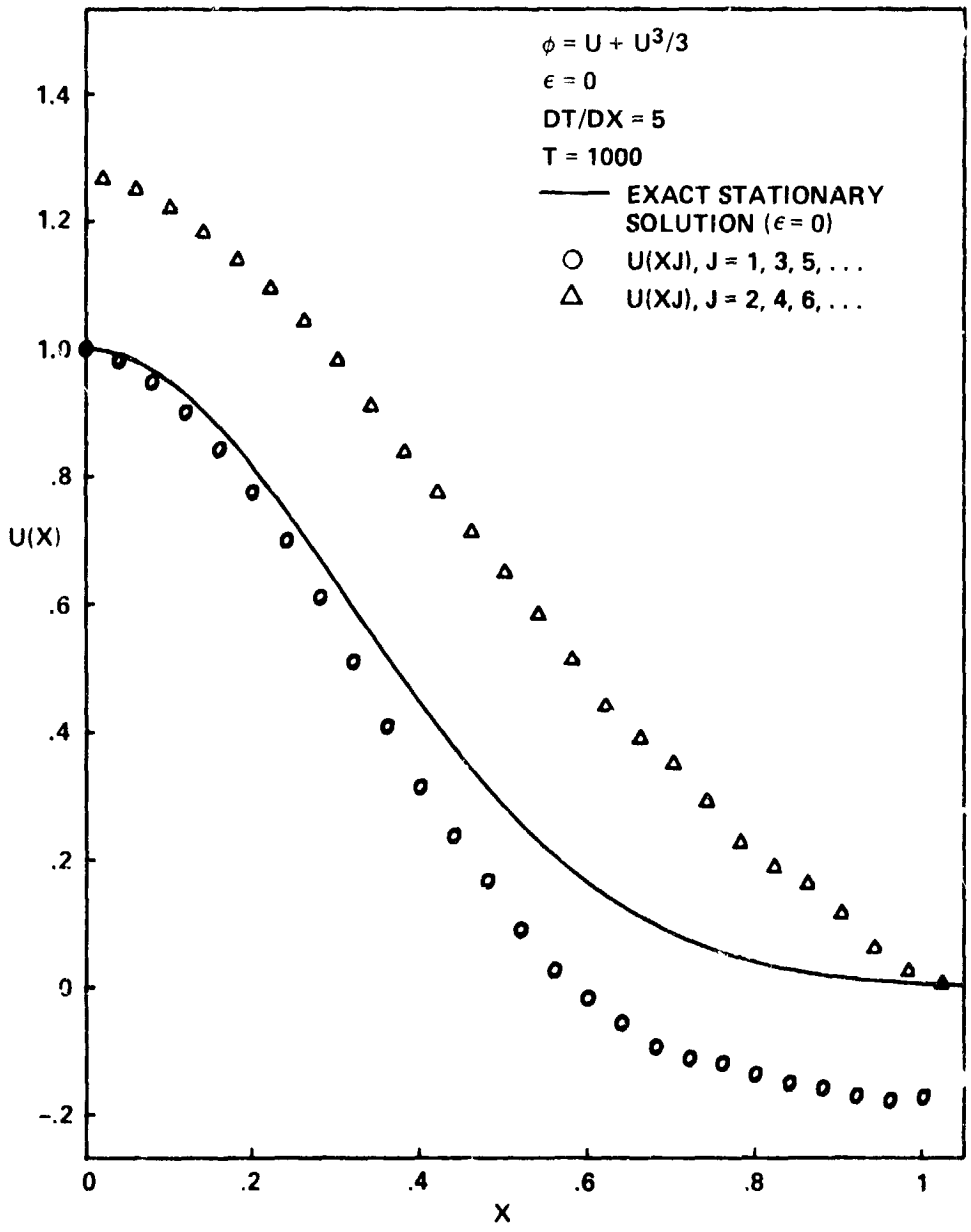


Figure 16.- An intermediate solution of a nonlinear equation with a positive flux gradient and no artificial dissipation added.

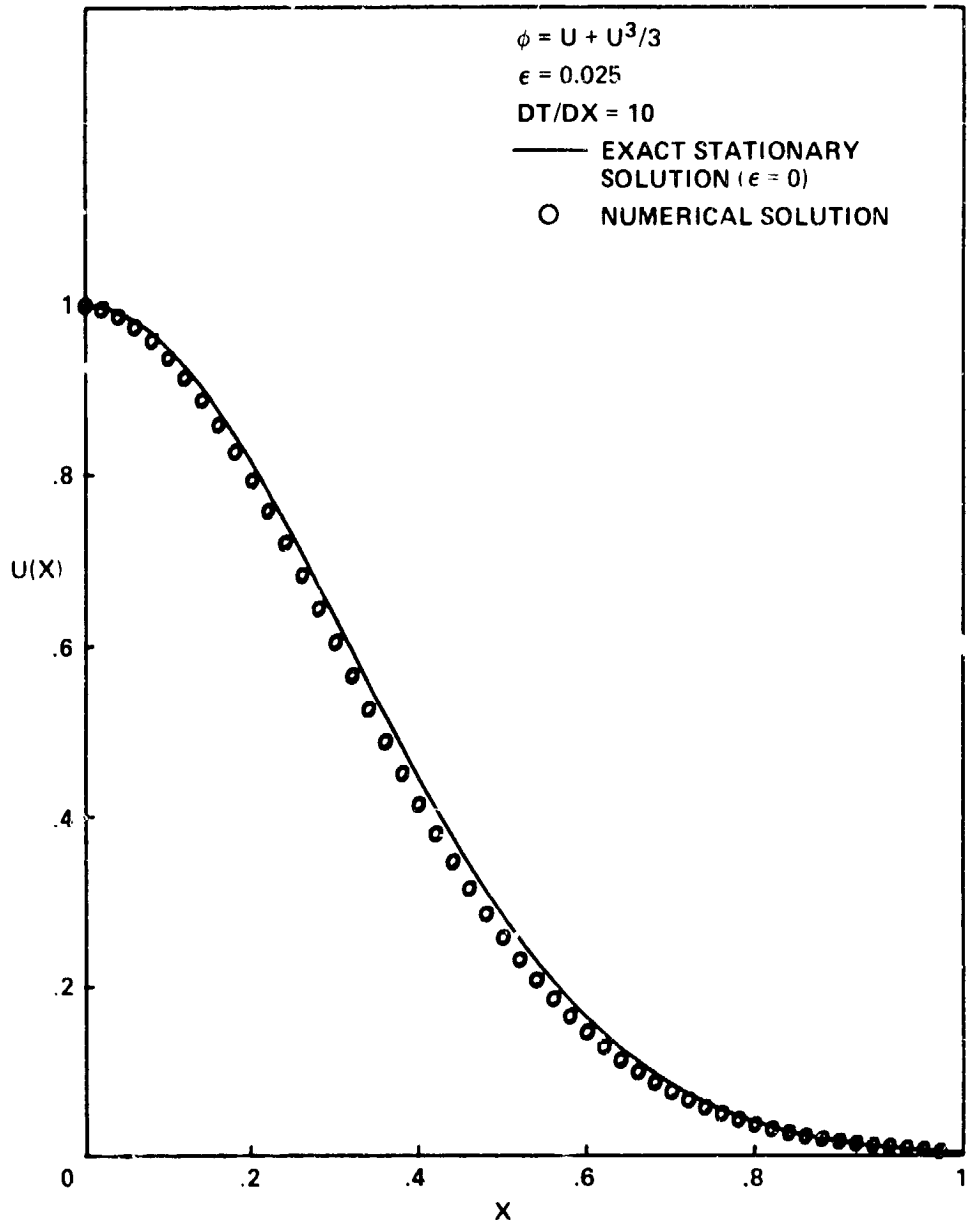


Figure 17.- Steady-state solution of a nonlinear equation with a positive flux gradient and some artificial dissipation added.

The last three experiments deal with a modified Burgers' equation obtained by letting $\phi = u^2/2$. In this case, the wave speed $a(u) = \partial\phi/\partial u = u$ changes sign twice at the initial time, but not at the steady state. If no dissipation is added, instability occurs (Test No. 14). However, for ε sufficiently large, the solution does converge. Figure 18 shows the steady-state solution which is obtained for $\varepsilon = 0.025$ (Test No. 15); again, an even more accurate solution can be obtained for a smaller value of ε, say $\varepsilon = 0.005$ (Test No. 16). Here the solution accuracy is not significantly degraded by the addition of artificial dissipation. This tends to indicate, for this simple problem at least, that viscosity methods converge when the wave speed $a = \partial\phi/\partial u$ does not change sign (at least) at the steady state.

In conclusion, the experiments do confirm that a particular form of instability can be triggered when the wave-speed $a(u,x) = \partial\phi/\partial u$, which plays the role of an eigenvalue of the Jacobian matrix of a flux vector, changes sign at some point, if central space differencing is used at this point. Severe solution accuracy degradation was experienced for a case where the nonuniform steady-state solution was such that the alternation of sign in the wave-speed remained at the steady state. This was to the extent of making the practicability of viscosity methods questionable for this case. However, the extension of this dramatic result to the solution of the Euler equations is uncertain.

C. Further Comments

The derivation of Section IVA above has brought a rationale, based on matrix analysis only, to explain one of the reasons for the necessity of using artificial dissipation. It also suggests that better stability properties

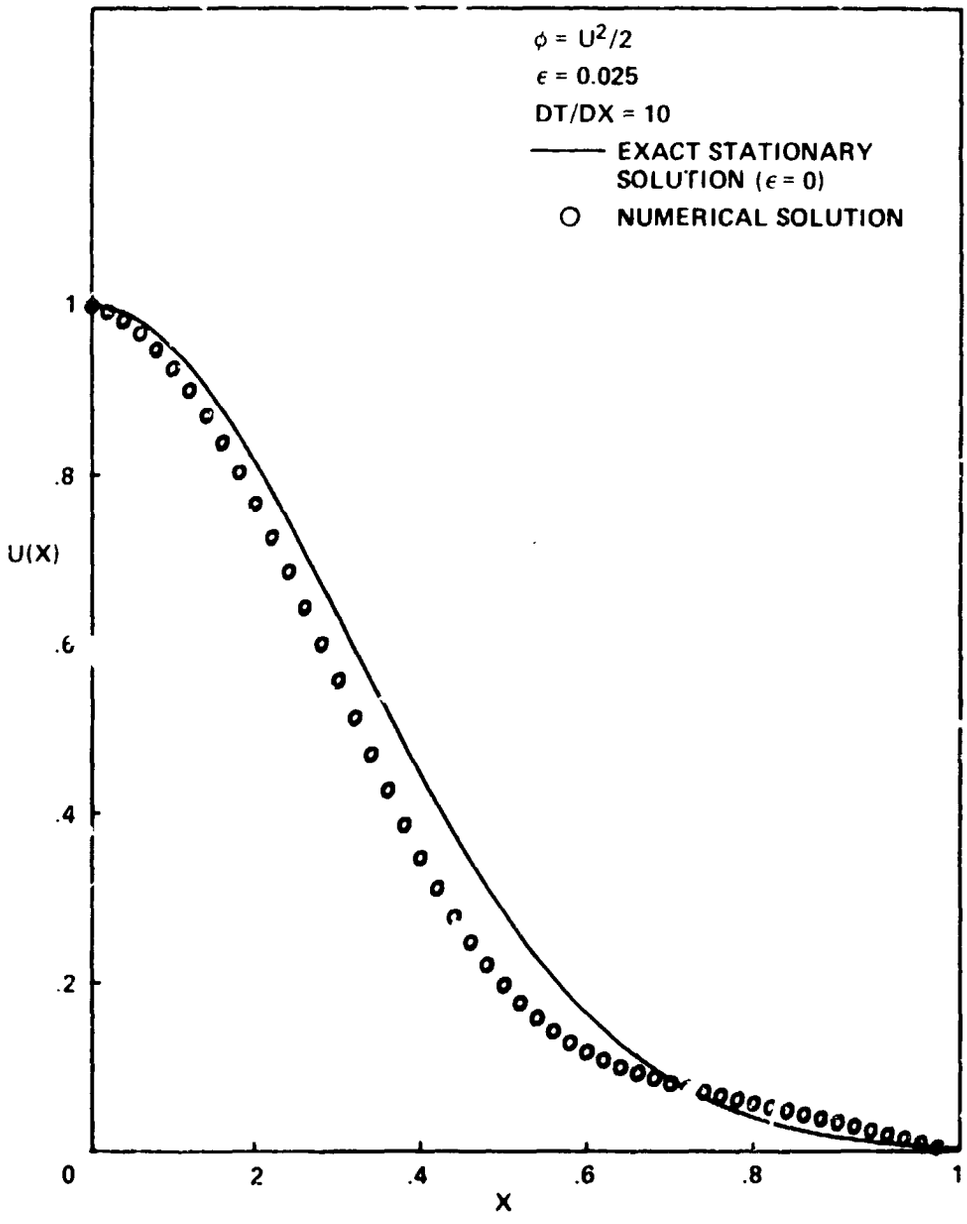


Figure 18.- Steady-state solution of Burgers equation with a source term and artificial dissipation added.

would perhaps result from simple modifications of the differencing at the points where one of the eigenvalues of the Jacobian matrix A or B changes sign.

For example, if the passage of say a_j^m through zero is smooth, the Jacobian matrix could be synthesized at one point from a modified eigensystem in which a_j^m would be set equal to zero exactly. This procedure ensures that the matrix C_x has purely imaginary eigenvalues only in the case of a scalar equation. However, a reduction in the required amount of added numerical dissipation would perhaps result from applying this technique.

Another procedure, applicable to the Euler implicit method, consists of averaging conservative with nonconservative differencing. This gives:

$$C_x = \Delta x(\delta_x A + A\delta_x)$$

$$= \text{BTrid}[-(A_{j-1} + A_j),\ 0,\ (A_{j+1} + A_j)] \qquad (87)$$

which would generate a real skew-symmetric matrix if A were symmetric, which is not, in general, for the Euler equations. However, a more favorable eigenvalue-spectrum can be anticipated from this. Using different arguments, Kreiss and Oliger [19] proposed essentially this for Burgers' equation. Since A and B are not themselves symmetric matrices, for the airfoil calculation of Chapter III, using a uniform mesh in this test, the algorithm was found to be stable with only twice larger time-steps when using this technique.

V. CONCLUSION

In this work, two conjectures were first made. First, it was claimed that the domain of unconditional stability of the base algorithm could be enlarged by introducing artificial dissipation in the implicit part of the differencing, as well as in the explicit part. Second, it was anticipated that the iterative convergence properties of the algorithm could be improved by the use of larger time-steps per se, but also by the use of a cyclic sequence of time-steps.

A heuristic stability analysis brought a theoretical support to the first conjecture. This analysis suggested that the time-step and the dissipation term added explicitly could both be arbitrary, provided the dissipation term added implicitly was kept sufficiently large. In particular, the two dissipation terms could be kept proportional to the time-step. In this way, the consistency condition was met, and the steady-state solution was independent of the time-step. This has been well confirmed by the numerical experiments that were conducted on a model transonic flow problem governed by the Euler equations. In fact, for this problem, it has never been possible to find a large enough value of the time-step, for which any adjustment of the dissipation terms would not remedy stability problems. However, for extremely large values of the time-step, the required amount of artificial dissipation was so large, that the iterative properties of the algorithm were degraded, although the numerical algorithm was stable. This was attributed to nonlinearities. For this reason, it was found that if a single time-step was used, this time-step could be optimized to a value roughly one order of magnitude larger than the one permitted by the base

differencing scheme. In this manner, the modified algorithm was found to converge about eight times faster than the base algorithm. Even more rapid convergence was obtained by using a sequence of time-steps. With the best sequence, an improvement in rate of convergence by a factor of 10 (over the base algorithm) was observed.

Various numerical experiments have shown that the modified algorithm was not very sensitive to nonoptimum parameters. In particular, approximately the same convergence rate was obtained when using a sequence of either four, six, or eight time-steps. Also, the nonoptimality of the sequence of time-steps, for a fixed number of them, did not seem to degrade the convergence rate severely.

Finally, a particular form of instability that the algorithm is subject to has been attributed to the spacial variation of the Jacobian matrices of the Euler equations. This instability was found to occur when central space-differencing is used at a point where one of the eigenvalues of either one of the Jacobian matrices changes sign. Addition of a sufficient amount of artificial dissipation remedies this type of instability. Nevertheless, two techniques have been proposed that could reduce the amount of required artificial dissipation. More conclusive results on this topic would, however, require further research.

VI. REFERENCES

1. W. R. Briley and H. McDonald. "An implicit numerical method for the multidimensional compressible Navier-Stokes equations." United Aircraft Research Laboratories, Report M911363-6, November 1973.

2. R. Beam and R. F. Warming. "An implicit finite-difference algorithm for hyperbolic systems in conservation-law form." Journal of Computational Physics, 22 (1976), 87-110.

3. R. Beam and R. F. Warming. "An implicit factored scheme for the compressible Navier-Stokes Equations." AIAA J., 16, No. 4 (1978), 393-402.

4. J. L. Steger. "Implicit finite-difference simulation of flows about arbitrary geometrics with application to airfoils," AIAA Paper 77-665, 1977.

5. T. H. Pulliam and J. L. Steger. "On implicit finite-difference simulations of three-dimensional flow." AIAA Paper 78-10, 1978.

6. P. Kutler, S. R. Chakravarthy, and C. K. Lombard. "Supersonic flow over ablated nosetips using an unsteady, implicit numerical procedure." AIAA Paper 78-213, 1978.

7. W. F. Ballhaus, A. Jameson, and J. Albert. "Implicit approximate-factorization schemes for the efficient solution of steady transonic flow problems." AIAA Paper 77-634, 1977.

8. T. L. Holst. "An implicit algorithm for the conservation transonic full potential equation using an arbitrary mesh." Paper 78-1113 to be presented at the AIAA 11th Fluid and Plasma Dynamics Conference, Seattle, 1978.

9. D. W. Peaceman and H. H. Rachford. "The numerical solution of parabolic and elliptic differential equations." J. Soc. Indust. Appl. Math., 3 (1955), 28-41.

10. J. Douglas. "On the numerical integration of $u_{xx} + u_{yy} = u_t$ by implicit methods." J. Soc. Indust. Appl. Math., 3 (1955), 42-65.

11. W. F. Ames. Numerical methods for partial differential equations. New York: Barnes & Noble, Inc., 1969.

12. R. D. Richtmyer and K. W. Morton. Difference methods for initial-value problems, Second Edition. New York: Interscience Publishers, a division of John Wiley & Sons, 1967.

13. R. W. MacCormack. "The effect of viscosity in hypervelocity impact cratering." AIAA Paper 69-354, 1969.

14. R. Courant, K. O. Friedrichs and H. Lewy. "Uber die partiellen Differenzengleichungen der mathematischen physik." In Math. Ann., Vol. 100, p. 32, 1928.

15. J. M. Ortega and W. C. Rheinboldt. Iterative solutions of nonlinear equations in several variables. New York and London: Academic Press, 1970.

16. R. Beam, private communication.

17. R. Bellman. Introduction to matrix analysis. Second Edition. New York: McGraw-Hill Book Company, 1970.

18. R. F. Warming, R. M. Beam, and B. J. Hyett. "Diagonalization and simultaneous symmetrization of gas-dynamic matrices." Mathematics of Computation, 29 (1975), 1037-1045.

19. H. Kreiss and J. Oliger. "Methods for the approximate solution of time dependent problems." Garp Publications Series No. 10, 1973.

20. F. W. Byron, Jr. and R. W. Fuller. Mathematics of classical and quantum physics, Vol. 1. Reading, Massachusetts: Addison-Wesley Publishing Company, 1969.

VII. ACKNOWLEDGMENTS

This work was supported by NASA-Ames Research Center under Grant NCA2-OR340-706 and the Engineering Research Institute, Iowa State University, Ames, Iowa.

VIII. APPENDIX A: MATRIX FORM OF THE FINITE-DIFFERENCE EQUATION FOR THE CASE OF A SCALAR DIFFERENTIAL EQUATION

In this appendix, the (matrix) definition of Kronecker products and sums, and some of their properties are first recalled. With this background, the finite-difference equation for the case where the implicit algorithm is applied to the two-dimensional first-order wave equation is derived in a form particularly convenient for the stability analysis of Section IIB.

A. Some Background on Kronecker Products and Sums

The definitions and the essential properties of Kronecker products and sums can be found in most books on matrix theory (e.g., [17] or [20]). The properties that are used in Section B of this appendix are repeated here, without proof, for the reader's convenience.

Let A and B be two square matrices of dimension $J\times J$ and $K\times K$ respectively. The Kronecker product of A and B is denoted by $A \otimes B$ and defined as the square matrix of dimension $JK\times JK$ given by:

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1J}B \\ a_{21}B & a_{22}B & \cdots & a_{2J}B \\ \vdots & \vdots & & \vdots \\ a_{J1}B & A_{J2}B & \cdots & a_{JJ}B \end{bmatrix} \tag{A1}$$

The Kronecker sum of A and B is defined as the matrix $A \otimes I_K + I_J \otimes B$ where I_m is the $m\times m$ identity matrix.

The following properties are true:

$$(A \otimes B) \otimes C = A \otimes (B \otimes C) \tag{A2}$$

$$(A + A') \otimes (B + B') = A \otimes B + A' \otimes B + A \otimes B' + A' \otimes B' \tag{A3}$$

$$(A \otimes B)(A' \otimes B') = (AA') \otimes (BB') \tag{A4}$$

where A' and B' have the same dimensions as A and B, respectively.

It follows from Equation (A4) that if A and B are nonsingular, then so is $A \otimes B$ and:

$$(A \otimes B)^{-1} = A^{-1} \otimes B^{-1} \tag{A5}$$

An important result concerning the eigensystems of Kronecker products and sums can be stated as follows: If $\lambda_1, \lambda_2, \ldots, \lambda_J$ are the eigenvalues of A and $\mu_1, \mu_2, \ldots, \mu_K$ are the eigenvalues of B, then the eigenvalues of $A \otimes B$ are the numbers $\lambda_j \mu_k$ and the eigenvalues of the Kronecker sum of A and B are the numbers $\lambda_j + \mu_k$. For both, the corresponding eigenvectors have the form:

$$Z_{jk} = \begin{bmatrix} x_1^j & y^k \\ x_2^j & y^k \\ \cdot & \\ \cdot & \\ \cdot & \\ x_J^j & y^k \end{bmatrix} \tag{A6}$$

where x_m^j is the mth component of the eigenvector x^j of A associated to λ_j, and y^k is the eigenvector of B associated to μ_k.

B. Application to the Finite-Difference Equation

In this section, the implicit algorithm is applied to the two-dimensional first-order wave equation (Equation (10)). The calculations are made assuming the solution u equal to zero at the boundaries, but the

result would hold for other linear boundary conditions as well. The equivalence between operator notation (Equation (3)) and matrix notation (Equation (11)) is explicited.

For this purpose, the components of the solution vector u are conventionally ordered as follows:

$$u = (u_{11}, u_{12}, \ldots, u_{1K}, u_{21}, u_{22}, \ldots, u_{2K}, \ldots, u_{J1}, u_{J2}, \ldots, u_{JK})^t \tag{A7}$$

where, as usual, $u_{jk} = u(x_j, y_k)$, $j = 1, 2, \ldots, J$ and $k = 1, 2, \ldots, K$.

Let E_x^+ and E_x^-, E_y^+, and E_y^- be the forward and backward shift operators for the x and y directions. More precisely:

$$\left.\begin{aligned} E_x^+u &= (u_{21}, u_{22}, \ldots, u_{2K}, u_{31}, u_{32}, \ldots, u_{3K}, \ldots, 0, 0, \ldots, 0)^t \\ E_x^-u &= (0, 0, \ldots, 0, u_{11}, u_{12}, \ldots, u_{1K}, \ldots, u_{J-1,1}, u_{J-1,2}, \ldots, u_{J-1,K})^t \\ E_y^+u &= (u_{12}, u_{13}, \ldots, 0, u_{22}, u_{23}, \ldots, 0, \ldots, u_{J2}, u_{J3}, \ldots, 0)^t \\ E_y^-u &= (0, u_{11}, \ldots, u_{1,K-1}, 0, u_{21}, \ldots, u_{2,K-1}, \ldots 0, u_{J1}, \ldots, u_{J,K-1})^t \end{aligned}\right\} \tag{A8}$$

where boundary conditions have been taken into account.

For $m = J$ or K, let I_m be the $m \times m$ identity matrix, and E_m be the following $m \times m$ matrix

$$E_m = \begin{bmatrix} 0 & 1 & & & \\ & 0 & 1 & & \\ & & \ddots & \ddots & \\ & & & 0 & 1 \\ & & & & 0 \end{bmatrix} \tag{A9}$$

If the matrix representations of the operators E_x^+, E_x^-, E_y^+, and E_y^- are denoted by the same symbols as the corresponding operators, it is apparent that:

$$E_x^+ = E_J \otimes I_K \tag{A10}$$

$$E_x^- = E_J^t \otimes I_K \tag{A11}$$

$$E_y^+ = I_J \otimes E_K \tag{A12}$$

$$E_y^- = I_J \otimes E_K^t \tag{A13}$$

Clearly, these equations also hold for periodic boundary conditions, if the lower left corner element of E_m in Equation (A9) is set equal to one. In this case, $E_m^t = E_m^{-1}$ so that forward and backward shift operators with the same subscript are inverse of one another.

Linear combinations of Equations (A10) through (A13) and of their powers and use of Equations (A3) and (A4) yield the desired finite-difference equation (Equation (11)).

IX. APPENDIX B: EIGENSYSTEMS OF TRIDIAGONAL MATRICES

In this appendix, the eigenvalues and eigenvectors of tridiagonal matrices are recalled. This is done in two cases: (1) periodic boundary conditions, and (2) specified boundary data.

Consider first the case where periodicity is enforced at the boundaries, so that the general tridiagonal $J \times J$ matrix has the following form:

$$A = \mathrm{Trid}(a,b,c) = \begin{bmatrix} b & c & & & & a \\ a & b & c & & & \\ & a & b & c & & \\ & & \ddots & \ddots & \ddots & \\ & & & a & b & c \\ c & & & & a & b \end{bmatrix} \tag{B1}$$

Define a sequence of vectors X_j $(j = 1, 2, \ldots, J)$ by

$$X_{mj} = 1/\sqrt{J}\, \exp(mi\theta_j) \tag{B2}$$

where $\theta_j = 2\pi(j - 1)/J$ and $m = 1, 2, \ldots, J$. If Equation (B2) is also applied for $m = 0$ and $m = J + 1$, the periodicity boundary conditions:

$$\left.\begin{aligned} X_{o,j} &= X_{J,j} \\ X_{1,j} &= X_{J+1,j} \end{aligned}\right\} \tag{B3}$$

are found automatically satisfied by X_j. Now compute AX_j:

$$\begin{aligned} (AX_j)_m &= A_{mk}X_{kj} \\ &= aX_{m-1,j} + bX_{mj} + cX_{m+1,j} \\ &= \lambda_j X_j \end{aligned} \tag{B4}$$

where

$$\lambda_j = a\, \exp(-i\theta_j) + b + c\, \exp(i\theta_j) \tag{B5}$$

this shows that X_j is an eigenvector of A associated to the eigenvalue λ_j. Now compute the following inner product:

$$\langle X_j, X_k \rangle = \bar{X}_j{}^t X_k = \frac{1}{J} \sum_{m=1}^{J} \exp(mi\alpha) \tag{B6}$$

where $\alpha = \theta_k - \theta_j = 2\pi(k - j)/J$. This gives

$$\langle X_j, X_j \rangle = 1 \tag{B7}$$

for all j, and for $j \neq k$:

$$\langle X_j, X_k \rangle = \frac{1}{J} \exp(i\alpha) \frac{\exp(Ji\alpha) - 1}{\exp(i\alpha) - 1} = 0 \tag{B8}$$

Hence, the eigenvectors X_j ($j = 1, 2, \ldots, J$) are orthonormal. The matrix X that diagonalizes A, which contains these eigenvectors for column vectors, is thus unitary:

$$X^{-1} = X^*(\text{adjoint of } X) = \bar{X}^t \tag{B9}$$

Consider now the case where the components of the solution vector are constrained to be zero at the boundaries. For this case, the general $J \times J$ tridiagonal matrix is given by Equation (B1) in which the upper-right and lower-left corner elements of the matrix on the right-hand side of this equation are set equal to zero. Then, define a sequence of vectors X_j ($j = 1, 2, \ldots, J$) by:

$$X_{mj} = \sqrt{2/(J + 1)}(\sqrt{a/c})^m \sin m\theta_j \tag{B10}$$

where $\theta_j = \pi j/(J + 1)$ and $m = 1, 2, \ldots, J$. If Equation (B10) is also applied for $m = 0$ and $m = J + 1$, the boundary conditions

$$X_{o,j} = X_{J+1,j} = 0 \tag{B11}$$

are found automatically satisfied by X_j. Now compute AX_j:

$$(AX_j)_m = A_{mk}X_{kj}$$

$$= aX_{m-1,j} + bX_{mj} + cX_{m+1,j}$$

$$= \lambda_j X_{mj} \tag{B12}$$

where:

$$\lambda_j = a(\sqrt{a/c})^{-1} \frac{\sin(m-1)\theta_j}{\sin m\theta_j} + b + c(\sqrt{a/c}) \frac{\sin(m+1)\theta_j}{\sin m\theta_j} \tag{B13}$$

$$= b + \sqrt{ac}\cos\theta_j$$

which is found independent of m as expected. This shows that X_j is an eigenvector of A associated to the eigenvalue λ_j.

Now consider the particular case where a, b, and c are real, with also $a = c$, and redefine the eigenvectors of A to be ξ_j. Clearly, those are (real) orthogonal since A is real symmetric in this case, so that:

$$\langle \xi_j, \xi_k \rangle = 0 \qquad (\text{for } j \neq k) \tag{B14}$$

Er plus:

$$\langle \xi_j, \xi_j \rangle = \frac{2}{J+1} \sum_{m=1}^{J} \sin^2 m\theta_j$$

$$= \frac{1}{J+1} \sum_{m=1}^{J} (1 - \cos 2m\theta_j)$$

$$= \frac{J-R}{J+1} \tag{B15}$$

where

$$\left.\begin{aligned} R &= \mathrm{Re}(S) \\ S &= \sum_{m=1}^{J} \exp(m i\alpha) \end{aligned}\right\} \tag{B16}$$

in which $\alpha = 2\pi j/(J + 1)$. Computing S gives:

$$S = \exp(i\alpha) \sum_{m=0}^{J-1} \exp(mi\alpha)$$

$$= \exp(i\alpha) \frac{\exp(Ji\alpha) - 1}{\exp(i\alpha) - 1}$$

$$= \exp i(J + 1) \frac{\alpha}{2} \frac{\sin \frac{J\alpha}{2}}{\sin \frac{\alpha}{2}} \tag{B17}$$

so that:

$$R = \frac{\cos(J + 1) \frac{\alpha}{2} \sin \frac{J\alpha}{2}}{\sin \frac{\alpha}{2}} \tag{B18}$$

Also:

$$\cos(J + 1) \frac{\alpha}{2} = \cos \pi j = (-1)^j$$

$$\sin \frac{J\alpha}{2} = \sin \left[\pi j - \frac{\alpha}{2}\right] = (-1)^{j+1} \sin \frac{\alpha}{2}$$

so that $R = -1$ and $(\xi_j, \xi_j) = 1$. As a result, the matrix ξ that diagonalizes A and contains the eigenvectors ξ_j for column-vectors is orthogonal:

$$\xi^t \xi = I \tag{B19}$$

This matrix is en plus symmetric:

$$\xi^t = \xi \tag{B20}$$

In particular, ξ diagonalizes the smoothing operator D_x of Equation (12):

$$D_x = \xi(2I + 2K)\xi = 2(I + \xi K \xi) \tag{B21}$$

where $K = \text{Diag}(c_j)$ and $c_j = \cos \theta_j$.

Now consider the matrix $C_x = \text{Trid}(-1, 0, 1)$, in Equation (12). In view of Equation (B10), it appears that this matrix is diagonalized by

$$X = D\xi \tag{B22}$$

where D is the diagonal matrix with mth eigenvalue equal to i^m. Hence,

$$X^{-1} = \xi^{-1}D^{-1} = \xi\bar{D} = \xi^*D^* = X^* \tag{B23}$$

showing that X is then unitary as expected, since C_x is real skew-symmetric. The eigenvalues of C_x are given by Equation (B13), and this permits us to write C_x in the following form:

$$\begin{aligned} C_x &= X(iK)X^{-1} \\ &= iD\xi K\xi\bar{D} \end{aligned} \tag{B24}$$

where the diagonal matrix K is defined in Equation (B21).

X. APPENDIX C: STABILITY CONDITION FOR PERIODIC BOUNDARY CONDITIONS AND SECOND-ORDER SMOOTHING

In this appendix, Equation (39) which expresses the stability condition for the case of periodic boundary conditions and second-order smoothing, is derived. For this purpose, it is recalled that the satisfaction of Equations (33) and (35) constitutes the necessary and sufficient condition for stability.

For the case considered,

$$\left.\begin{aligned} d_j &= d'_j = 2(1 - \cos \theta_j) \\ d_k &= d'_k = 2(1 - \cos \theta_k) \end{aligned}\right\} \tag{C1}$$

where $\theta_j = 2\pi(j - 1)/J$ $(j = 1, 2, \ldots, J)$ and $\theta_k = 2\pi(k - 1)/K$ $(k = 1, 2, \ldots, K)$, and it is convenient to let:

$$\left.\begin{aligned} d_j &= d'_j = 4\xi \\ d_k &= d'_k = 4\eta \\ \varepsilon_i &= \varepsilon/4 \\ \varepsilon_e &= \bar{\varepsilon}/4 \end{aligned}\right\} \tag{C2}$$

In this manner, the new variables ξ and η, which are not subscripted for notational simplicity, take their values in the interval $[0,1]$. Equation (36) then becomes:

$$\begin{aligned} g_{jk}(\theta) &= \omega^2\bar{\varepsilon}^2(\xi + \eta)^2 - 2\omega\bar{\varepsilon}(\xi + \eta)[\theta\varepsilon(\xi + \eta) + 2] \\ &\quad + 2\bar{\varepsilon}(\xi + \eta) + \theta^2\varepsilon^2(\xi - \eta)^2 \\ &= \omega^2[(\bar{\varepsilon} - \varepsilon)^2(\xi + \eta)^2 - 4\varepsilon^2\xi\eta - 4\mu\bar{\varepsilon}(\xi + \eta)] \end{aligned} \tag{C3}$$

where $\mu = (\theta - 1/2)/\theta^2$. As a result, Equation (35) becomes:

$$f(\xi,\eta) \leq \mu\bar{\varepsilon} \tag{C4}$$

where

$$f(\xi,\eta) = \left(\frac{\bar{\varepsilon} - \varepsilon}{2}\right)^2 (\xi + \eta) - \varepsilon^2 \frac{\xi\eta}{\xi + \eta} \tag{C5}$$

The satisfaction of Equation (C4) for all feasible values of ξ and η is equivalent to the condition:

$$S(\varepsilon,\bar{\varepsilon}) \leq \mu\bar{\varepsilon} \tag{C6}$$

where

$$\left.\begin{aligned} S(\varepsilon,\bar{\varepsilon}) &= \text{Sup } f(\xi,\eta) \\ &0 \leq \xi, \eta \leq 1 \end{aligned}\right\} \tag{C7}$$

Some simple conclusions can be drawn at first. For this, note that, if $\varepsilon = \bar{\varepsilon}$, $f(\xi,\eta) \leq 0$ and Equation (C6) is satisfied. Thus, the algorithm is unconditionally stable for every value of θ (provided Equation (33) holds) for the case where the equation

$$u_t + au_x + bu_y = c(u_{xx} + u_{yy}) \tag{C8}$$

is differenced in a time-accurate manner. Also observe that $f(1,0) = (\bar{\varepsilon} - \varepsilon)^2/4 \geq 0$, so that, unless perhaps this value is zero, $S(\varepsilon,\bar{\varepsilon})$ is strictly positive. Consequently, for trapezoidal time-differencing ($\theta = 1/2$), since $\mu = 0$, letting $\varepsilon = \bar{\varepsilon}$ constitutes the only way of enforcing unconditional stability (see Equation (C6)). For this reason, $\theta > 1/2$ and $\mu > 0$ are assumed in the remaining.

The next step consists of the determination of $S(\varepsilon,\bar{\varepsilon})$, that is, the maximization of $f(\xi,\eta)$ for fixed values of ε and $\bar{\varepsilon}$. Inspection of Equation (C5) indicates that $f(\xi,\eta)$ is a homogeneous function of ξ and η

of degree one. Applications of Euler's theorem for homogeneous functions gives the following identity:

$$f(\xi,\eta) = \frac{\partial f}{\partial \xi}\,\xi + \frac{\partial f}{\partial \eta}\,\eta \tag{C9}$$

This shows that if there exists a local maximum of $f(\xi,\eta)$ at a point of the open square $]0, 1[\times]0, 1[$, this maximum is equal to zero. In view of Equation (C4), it appears that such maximum, if it exists, does not yield any binding condition for stability. Hence, the maximization of $f(\xi,\eta)$ can be reduced to the one over the boundaries of the square. Since, en plus, $f(\xi,\eta)$ is symmetric in ξ and η, only two of these boundaries need to be considered. These are the segments: (1) $0 \leq \xi \leq 1$ and $\eta = 0$, and (2) $\xi = 1$ and $0 \leq \eta \leq 1$.

For $\eta = 0$,

$$f(\xi,0) = \left(\frac{\bar{\varepsilon} - \varepsilon}{2}\right)^2 \xi \tag{C10}$$

It is maximum at the point $\xi = 1$, which also belongs to the second segment. Consequently,

$$\left.\begin{aligned} S(\varepsilon,\bar{\varepsilon}) &= \operatorname{Sup}\ \phi(\eta) \\ &\qquad 0 \leq \eta \leq 1 \end{aligned}\right\} \tag{C11}$$

where

$$\begin{aligned} \phi(\eta) &= f(1,\eta) \\ &= \left(\frac{\bar{\varepsilon} - \varepsilon}{2}\right)^2 (\eta + 1) + \frac{\varepsilon^2}{\eta + 1} - \varepsilon^2 \end{aligned} \tag{C12}$$

Differentiating Equation (C12) gives:

$$\phi'(\eta) = \left(\frac{\bar{\varepsilon} - \varepsilon}{2(\eta + 1)}\right)^2 [(\eta + 1)^2 - \omega^2] \tag{C13}$$

where $\omega = 2\varepsilon/|\bar{\varepsilon} - \varepsilon|$. If $\omega \leq 1$ or $\omega \leq 2$. $\phi'(\eta)$ does not change sign when η increases from 0 to 1, and $\phi(\eta)$ is maximum at either one of the

two endpoints If $1 < \omega < 2$, $\phi'(\eta) \leq 0$ for $0 \leq \eta \leq \omega - 1$ and $\phi'(\eta) \geq 0$ for $\omega - 1 \leq \eta \leq 1$, so that $\phi(\eta)$ is again maximum at either one of the two endpoints. Finally:

$$S(\epsilon,\bar{\epsilon}) = \text{Max}[\phi(0), \phi(1)] \tag{C14}$$

One obtains:

$$\left.\begin{aligned} \phi(0) &= \left(\frac{\bar{\epsilon} - \epsilon}{2}\right)^2 \geq 0 \\ \phi(1) &= \bar{\epsilon}\left(\frac{\bar{\epsilon}}{2} - \epsilon\right) \end{aligned}\right\} \tag{C15}$$

Equation (C6) then breaks into the following three inequalities:

$$\left.\begin{aligned} -\sqrt{\mu\bar{\epsilon}} \leq \frac{\epsilon - \bar{\epsilon}}{2} \leq \sqrt{\mu\bar{\epsilon}} \\ \frac{\bar{\epsilon}}{2} - \mu \leq \epsilon \end{aligned}\right\} \tag{C16}$$

Using the definitions of ϵ and $\bar{\epsilon}$ given in Equation (C2) and making a few simplifications yields Equation (39) (Q.E.D.).

Remark: In this derivation, no case has been made of the shape of the function $h(\theta)$ for which $d_j = d'_j = h(\theta_j)$ and $d_k = d'_k = h(\theta_k)$. Hence, Equation (C16) applies to all the cases where the same type of smoothing is applied implicitly and explicitly, provided ϵ and $\bar{\epsilon}$ are defined by:

$$\epsilon = \theta\epsilon_i/\lambda_{max}$$

$$\bar{\epsilon} = \epsilon_e/\lambda_{max}$$

where it is assumed that the eigenvalues of the smoothing operator vary from 0 to λ_{max}. In particular, for fourth-order smoothing applied explicitly and implicitly one would let:

$$\epsilon = \frac{\theta}{16}\epsilon_i$$

$$\bar{\epsilon} = \frac{1}{16}\epsilon_e$$

and apply Equation (C16).

XI. APPENDIX D: STABILITY CONDITION FOR PERIODIC BOUNDARY CONDITIONS AND FOURTH-ORDER SMOOTHING

In this appendix, Equation (40), which expresses the stability condition for the case of periodic boundary conditions and fourth-order smoothing, is derived. For this purpose, it is recalled that the satisfaction of Equations (33) and (35) constitutes the necessary and sufficient condition for stability.

For the case considered,

$$\left.\begin{aligned} d_j &= 2(1 - \cos\theta_j) , \quad & d_j' &= d_j^2 \\ d_k &= 2(1 - \cos\theta_k) , \quad & d_k' &= d_k^2 \end{aligned}\right\} \tag{D1}$$

where $\theta_j = 2\pi(j - 1/J$ $(j = 1, 2, \ldots, J)$ and $\theta_k = 2\pi(k - 1)/K$ $(k = 1, 2, \ldots, K)$, and it is convenient to let:

$$\left.\begin{aligned} d_j &= 4\xi , \quad & d_j' &= 16\xi^2 \\ d_k &= 4\eta , \quad & d_k' &= 16\eta^2 \\ \epsilon_i &= \theta\epsilon/' \\ \epsilon_e &= \bar{\epsilon}/16 \end{aligned}\right\} \tag{D2}$$

In this manner, the new variables ξ and η, which are not subscripted for notational simplicity, take their values in the interval [0, 1]. Equation (36) then becomes:

$$\begin{aligned} g_{jk}(\theta) = {} & \theta^2\bar{\epsilon}^2(\xi^2 + \eta^2)^2 - 2\theta\bar{\epsilon}(\xi^2 + \eta^2)[\theta\epsilon(\xi + \eta) + 2] \\ & + 2\bar{\epsilon}(\xi^2 + \eta^2) + \theta^2\epsilon^2(\xi - \eta)^2 \end{aligned} \tag{D3}$$

Define:

$$g(\xi,\eta) = \frac{\bar{\epsilon}(\xi^2 + \eta^2) - \epsilon(\xi + \eta)}{2} \tag{D4}$$

and

$$f(\xi,\eta) = g^2(\xi,\eta) - \mu\bar{\varepsilon}(\xi^2 + \nu^2) - \varepsilon^2\xi\eta \tag{D5}$$

where $\mu = (\theta - 1/2)/\theta^2$, so that

$$g_{jk}(\theta) = 4\theta^2 f(\xi,\eta) \tag{D6}$$

As a result, Equation (35) becomes:

$$f(\xi,\eta) \leq 0 \tag{D7}$$

The satisfaction of Equation (D7) for all feasible values of ξ and η is equivalent to the condition:

$$S(\varepsilon,\bar{\varepsilon}) \leq 0 \tag{D8}$$

where:

$$\left.\begin{aligned} S(\varepsilon,\bar{\varepsilon}) &= \text{Sup } f(\xi,\eta) \\ &0 \leq \xi,\ \eta \leq 1 \end{aligned}\right\} \tag{D9}$$

Thus, the problem consists of maximizing $f(\xi,\eta)$ over the closed square $[0, 1] \times [0, 1]$. For this purpose, one first looks for stationary points of $f(\xi,\eta)$ that belong to the open square. At these points, if any exists,

$$\left.\begin{aligned} \frac{\partial f}{\partial \xi} &= 2g(\xi,\eta)\,\frac{\partial g}{\partial \xi} - 2\mu\bar{\varepsilon}\xi - \varepsilon^2\eta = 0 \\ \frac{\partial f}{\partial \eta} &= 2g(\xi,\eta)\,\frac{\partial g}{\partial \eta} - 2\mu\bar{\varepsilon}\eta - \varepsilon^2\xi = 0 \end{aligned}\right\} \tag{D10}$$

so that, in particular

$$\begin{aligned} 0 &= \frac{\partial f}{\partial \eta} - \frac{\partial g}{\partial \xi} \\ &= 2g(\xi,\eta)\left(\frac{\partial g}{\partial \eta} - \frac{\partial g}{\partial \xi}\right) + (2\mu\bar{\varepsilon} - \varepsilon^2)(\xi - \eta) \\ &= (\xi - \eta)[-2\bar{\varepsilon}g(\xi,\eta) + 2\mu\bar{\varepsilon} - \varepsilon^2] \end{aligned} \tag{D11}$$

which requires the satisfaction of at least one of the following two equations:

$$\xi = \eta \tag{D12}$$

$$2\mu\bar{\varepsilon} - 2\bar{\varepsilon}g(\xi,\eta) = \varepsilon^2 \tag{D13}$$

If Equation (D10) holds, it is also true that:

$$\begin{aligned} 0 &= \eta \frac{\partial f}{\partial \eta} - \xi \frac{\partial f}{\partial \xi} \\ &= 2g(\xi,\eta)\left[\eta \frac{\partial g}{\partial \eta} - \xi \frac{\partial g}{\partial \xi}\right] + 2\mu\bar{\varepsilon}(\xi^2 - \eta^2) \\ &= 2g(\xi,\eta)[\bar{\varepsilon}(\eta^2 - \xi^2) - \frac{\varepsilon}{2}(\eta - \xi)] + 2\mu\bar{\varepsilon}(\xi^2 - \eta^2) \\ &= (\xi^2 - \eta^2)[2\mu\bar{\varepsilon} - 2\bar{\varepsilon}g(\xi,\eta)] + \varepsilon g(\xi,\eta)(\xi - \eta) \end{aligned} \tag{D14}$$

Now, assume that $f(\xi,\eta)$ is stationary at a point (ξ,η) of the open square that does not belong to the line $\xi = \eta$. Then, at this point, Equations (D13) and (D14) must hold simultaneously, so that:

$$0 = \varepsilon(\xi + \eta) + g(\xi,\eta) = \frac{1}{2}[\bar{\varepsilon}(\xi^2 + \eta^2) + \varepsilon(\xi + \eta)] \tag{D15}$$

If the trivial case $\varepsilon = \bar{\varepsilon} = 0$ is eliminated, the satisfaction of Equation (D15) is impossible for $\xi > 0$ and $\eta > 0$. This brings a contradiction to the assumption $\xi \neq \eta$. Consequently, if $f(\xi,\eta)$ is stationary at a point (ξ,η) of the open square, this point belongs to the line $\xi = \eta$. Hence, it suffices to enforce Equation (D7) on the boundaries of the square and its diagonal segment $0 \leq \xi = \eta \leq 1$. Observing that $f(\xi,\eta)$ is symmetric in ξ and η permits us to further reduce its maximization to the one over the following three segments: (1) $0 \leq \xi = \eta \leq 1$, (2) $0 \leq \xi \leq 1$, $\eta = 0$, and (3) $\xi = 1$, $0 \leq \eta \leq 1$. For this reason, three cases will now be examined separately.

Case 1: $0 \leq \xi = \eta \leq 1$

Define

$$\phi(\xi) = f(\xi,\xi)/\xi^2 = (\bar{\varepsilon}\xi - \varepsilon)^2 - (2\mu\bar{\varepsilon} + \varepsilon^2) \tag{D16}$$

so that

$$\phi'(\xi) = 2\bar{\varepsilon}(\bar{\varepsilon}\xi - \varepsilon) \tag{D17}$$

When ξ increases from 0 to $\varepsilon/\bar{\varepsilon}$, $\phi(\xi)$ decreases, and since $\phi(0) = -2\mu\bar{\varepsilon} \leq 0$ remains negative. For values of ξ greater than $\varepsilon/\bar{\varepsilon}$, $\phi(\xi)$ is an increasing function of ξ. The satisfaction of Equation (D7) over the considered segment is hence equivalent to the condition

$$\phi(1) \leq 0 \tag{D18}$$

This gives the following necessary condition for stability:

$$\varepsilon \geq \frac{1}{2}(\bar{\varepsilon} - 2\mu) \tag{D19}$$

Case 2: $0 \leq \xi \leq 1$, $\eta = 0$

For this case, Equation (D7) takes the following particular form:

$$\left(\frac{\varepsilon\xi^2 - \varepsilon\xi}{2}\right)^2 - \mu\bar{\varepsilon}\xi^2 \leq 0 \tag{D20}$$

which is equivalent to:

$$(\bar{\varepsilon}\xi - \varepsilon)^2 \leq 4\mu\bar{\varepsilon} \tag{D21}$$

or

$$\bar{\varepsilon}\xi - 2\sqrt{\mu\bar{\varepsilon}} \leq \varepsilon \leq \bar{\varepsilon}\xi + 2\sqrt{\mu\bar{\varepsilon}} \tag{D22}$$

The binding case for the inequality on the left corresponds to $\xi = 1$. For the inequality on the right, the binding case corresponds to $\xi = 0$. This gives the following necessary condition for stability:

$$\bar{\varepsilon} - 2\sqrt{\mu\bar{\varepsilon}} \leq \varepsilon \leq 2\sqrt{\mu\bar{\varepsilon}} \tag{D23}$$

Case 3: $\xi = 1$, $0 \le \eta \le 1$

For this case, one defines

$$\psi(\eta) = f(1, \eta)$$

$$= \frac{1}{4}\left[\bar{\varepsilon}(\eta^2 + 1) - \varepsilon(\eta + 1)\right]^2 - \mu\bar{\varepsilon}(\eta^2 + 1) - \varepsilon^2\eta \qquad \text{(D24)}$$

which is a fourth-degree polynomial in η. The satisfaction of Equation (D7) over the considered segment is equivalent to the condition

$$\psi(\eta) \le 0 \qquad \text{(D25)}$$

for $0 \le \eta \le 1$. It is assumed that Equations (D19) and (D23) hold, so that $\psi(0)$ and $\psi(1)$ are both nonpositive. Hence, if an additional condition must be enforced, it must be of the form:

$$\psi(\eta^*) \le 0 \qquad \text{(D26)}$$

where $0 < \eta^* < 1$, and

$$\psi'(\eta^*) = 0 \qquad \text{(D27)}$$

(for a local maximum). Dividing the polynomial $\psi(\eta)$ by its derivative $\psi'(\eta)$, according to decreasing powers of η, produces a quadratic remainder, $q(\eta)$. This gives the following identity:

$$\psi(\eta) = (a\eta + b)\psi'(\eta) + q(\eta) \qquad \text{(D28)}$$

where

$$q(\eta) = \alpha\eta^2 + \beta\eta + \gamma \qquad \text{(D29)}$$

The calculation of the coefficients a, b, α, β, and γ gives:

$$a = 1/4$$

$$b = -\varepsilon/(8\bar{\varepsilon}) \qquad \text{(D30)}$$

and,

$$\alpha = \frac{1}{16}\left\{4\bar{\varepsilon}(\bar{\varepsilon} - \varepsilon - 2\mu) - \varepsilon^2\right\}$$

$$\beta = \frac{\varepsilon}{16\bar{\varepsilon}}\left\{(\varepsilon^2 - 4\mu\bar{\varepsilon}) - 4\bar{\varepsilon}^2 - 8\varepsilon\bar{\varepsilon}\right\} \qquad \text{(D31)}$$

$$\gamma = \frac{1}{4}\left\{\left[(\bar{\varepsilon} - \varepsilon)^2 - 4\mu\bar{\varepsilon}\right] - \frac{\varepsilon^2(\varepsilon + \bar{\varepsilon})}{4\bar{\varepsilon}}\right\}$$

where terms between brackets [ ] are nonpositive, as a consequence of Equation (D23). Hence, it is directly apparent that $\beta \leq 0$ and $\gamma \leq 0$. This implies that $q(0) = \gamma \leq 0$. Now compute $q(1)$:

$$q(1) = \alpha + \beta + \gamma$$
$$\leq \frac{1}{16}(4\bar{\varepsilon}^2 - 4\bar{\varepsilon}\varepsilon - 8\mu\bar{\varepsilon}) + \frac{\varepsilon}{16\bar{\varepsilon}}(-4\bar{\varepsilon}^2 - 8\varepsilon\bar{\varepsilon}) + \frac{1}{4}(\bar{\varepsilon} - \varepsilon)^2 - \mu\bar{\varepsilon} \tag{D32}$$

where some nonpositive terms have been neglected. After a few simplifications, one obtains:

$$q(1) \leq \frac{\bar{\varepsilon}}{2}(\bar{\varepsilon} - 2\varepsilon) - \frac{3}{2}\mu\bar{\varepsilon} \tag{D33}$$

This indicates that in case $\bar{\varepsilon} \leq 2\varepsilon$, $q(1) \leq 0$. If now, $\bar{\varepsilon} - 2\varepsilon > 0$, the satisfaction of Equation (D19) requires that $\bar{\varepsilon} - 2\varepsilon \leq 2\mu$, so that $q(1) \leq -\mu\bar{\varepsilon}/2 \leq 0$. Thus, in all cases, $q(1) \leq 0$. Now, if $\alpha \leq 0$, since $\beta \leq 0$ and $\gamma \leq 0$, $q(\eta) \leq 0$ for $0 \leq \eta$. If $\alpha > 0$, $q(\eta)$ achieves a minimum at the point

$$\bar{\eta} = -\frac{\beta}{2\alpha} \geq 0 \tag{D34}$$

Whether $\bar{\eta}$ belongs to the interval [0, 1] or not, since $q(0)$ and $q(1)$ are both nonpositive, so is $q(\eta)$ for all values of η in this interval. Hence, in both cases, $q(\eta) \leq 0$ for $0 \leq \eta \leq 1$.

Hence, if it happens that $\psi(\eta)$ admits a local maximum at a point η^* of the open interval]0, 1[, then, as a consequence of Equations (D27) and (D28), the following is true:

$$\psi(\eta^*) = q(\eta^*) \leq 0 \tag{D35}$$

This shows that $\psi(\eta) \leq 0$ for all values of η in the open interval]0, 1[, provided Equations (D19) and (D23) hold (sufficiency).

In conclusion, the satisfaction of Equations (D19) and (D23) constitutes the necessary and sufficient condition for the unconditional stability of the algorithm for the case considered. Replacing, in these equations, ε and $\bar{\varepsilon}$ by their definitions, given in Equation (D2), yields the desired equation (Equation (40)).

Remark 1:

In this derivation, no case has been made of the shape of the function $h(\theta)$ for which $d_j = h(\theta_j)$, $d'_j = h^2(\theta_j)$, $d_h = h(\theta_k)$, $d'_k = h^2(\theta_k)$. Hence, Equations (D19) and (D23) apply to all the cases where the explicit smoothing operator D'_x (or D'_y) is the square of the implicit smoothing operator D_x (or D_y), provided ε and $\bar{\varepsilon}$ defined by

$$\left.\begin{aligned} \varepsilon &= \theta\varepsilon_i/\lambda_{max} \\ \bar{\varepsilon} &= \varepsilon_e/\lambda^2_{max} \end{aligned}\right\} \qquad \text{(D36)}$$

where it is assumed that the eigenvalues of D_x (or D_y) vary from 0 to λ_{max}. However, this has apparently no application, since the next step after the combination second-order implicit/fourth-order explicit smoothing would be the combination fourth-order implicit/sixteenth-order explicit smoothing, which is impractical.

Remark 2:

Note that Equations (D19) and (D23) are equivalent to Equation (C16). This is because the same boundary values of ξ and η give rise to the stability conditions for fourth-order smoothing as for second-order smoothing. Hence the two cases only differ by scale factors which have been eliminated from Equations (C16), (D19), and (D23) by appropriate definitions of the parameters ε and $\bar{\varepsilon}$.

XII. APPENDIX E: EFFECTIVE EIGENVALUES OF THE SMOOTHING OPERATORS AT LARGE COURANT NUMBERS

The "effective eigenvalues," $\tilde{d}_j$ and $\tilde{d}'_j$, of the smoothing operators, D_x and $D'_x = D_x$ or D_x^2, were introduced in the development of the stability analysis for the case of specified boundary data and large Courant numbers (Section IIB3). These effective eigenvalues are evaluated in this appendix.

For this purpose, one recalls that:

$$\tilde{d}_j = (X^{-1} D_x X)_{jj} \tag{E1}$$

$$\tilde{d}'_j = (X^{-1} D'_x X)_{jj} \tag{E2}$$

where the various matrices are of dimensions $J \times J$ and defined by:

$$\left.\begin{aligned} D_x &= \text{Trid}(-1, 2, -1) \\ D'_x &= D_x \text{ or } D_x^2 \\ X_{mj} &= \sqrt{2/(J+1)}\; i^m \sin m\theta_j \\ X^{-1}_{mj} &= \sqrt{2/(J+1)}\; (-1)^j \sin j\theta_m \end{aligned}\right\} \tag{E3}$$

where $\theta_j = j\pi/(J+1)$ and $m, j = 1, 2, \ldots, J$.

Applying Equation (B13) to the case where:

$$\left.\begin{aligned} a &= -1 - \varepsilon \\ b &= 2\varepsilon \\ c &= 1 - \varepsilon \end{aligned}\right\} \tag{E4}$$

gives

$$\begin{aligned} \lambda_j &= 2\varepsilon + \sqrt{-(1-\varepsilon^2)}\cos\theta_j \\ &= i\cos\theta_j + 2\varepsilon + O(\varepsilon^2) \end{aligned} \tag{E5}$$

This shows that when the matrix $D_x = \text{Trid}(-1, 0, 1)$ is perturbed by matrix $\varepsilon D_x = \text{Trid}(-\varepsilon, 2\varepsilon, -\varepsilon)$, where ε is a small parameter, the first-order

perturbations brought to the eigenvalues of C_x have the coefficient 2. This is equivalent to saying that:

$$\tilde{d}_j = 2 \tag{E6}$$

as one could compute by expliciting Equations (E1). But according to Equations (B21) and (B22):

$$\begin{aligned} \tilde{D}_x &= X^{-1} D_x X \\ &= \xi \bar{D}(2I + 2\xi K \xi) D\xi \\ &= 2I + 2U^* K U \end{aligned} \tag{E7}$$

where ξ, D, and K are defined in Appendix B, and

$$U = \xi D \xi \tag{E8}$$

is another unitary matrix. One concludes that the diagonal elements of U^*KU are all equal to zero. Considering now the case where $D_x' = D_x^2$ and squaring Equation (E7) yields:

$$\begin{aligned} \tilde{D}_x' &= X^{-1} D_x^2 X \\ &= 4I + 8U^* K U + 4U^* K^2 U \end{aligned} \tag{E9}$$

Hence, for this case:

$$\tilde{d}_j' = 4(1 + w_{jj}) \tag{E10}$$

where:

$$\begin{aligned} w_{jj} &= (U^* K^2 U)_{jj} \\ &= (\bar{U}^t)_{jm} K^2_{m\ell} U_{\ell j} \\ &= \bar{U}_{mj} K^2_{m\ell} U_{\ell j} \\ &= \sum_{m=1}^{J} |U_{mj}|^2 c_m \end{aligned} \tag{E11}$$

in which $c_m = \cos \theta_m$. It has not been possible to evaluate w_{jj} explicitly. However, it appears from Equation (E11) that w_{jj}, and consequently

$\tilde{d}'_j$ is real and positive. Moreover, since U is unitary and symmetric, the following is true for every value of j:

$$\sum_{m=1}^{J} |U_{mj}|^2 = 1 \tag{E12}$$

Since $|c_m| \leq 1$, the following bound holds:

$$0 \leq w_{jj} \leq \sum_{m=1}^{J} |U_{mj}|^2 \cdot 1 = 1 \tag{E13}$$

and consequently:

$$4 \leq \tilde{d}'_j \leq 8 \tag{E14}$$

The remainder of this appendix shows that the maximum value of $\tilde{d}'_j$ does converge to 8 in the limit of a mesh refinement. For this, let

$$w = \underset{j}{\text{Max}}\, w_{jj} \tag{E15}$$

to make the claim equivalent to the following statement:

$$\lim_{j\to\infty} w = 1 \tag{E16}$$

The simplifying assumption that J is odd is made, and one lets

$$J + 1 = 2\nu \tag{E17}$$

so that $\theta_\nu = \pi\nu/(J+1) = \pi/2$. Since $w \leq 1$ (see Equation (E13)), it is sufficient to show that:

$$\lim_{J\to\infty} w_{\nu\nu} = 1 \tag{E18}$$

To evaluate $w_{\nu\nu}$, one first computes $U_{m\nu}$ using Equation (E8):

$$U_{m\nu} = (\xi D \xi)_{m\nu} = \xi_{mj} D_{jj} \xi_{j\nu}$$

$$= \frac{2}{J+1} \sum_{j=1}^{J} i^j \sin m\theta_j \sin j\theta_\nu \tag{E19}$$

where the values of ξ and D were taken from Appendix B. Recall that $\theta_\nu = \pi/2$ to simplify Equation (E19) as follows:

$$U_{m\nu} = \frac{2}{J+1} \sum_{k=1}^{\nu} i^{2k-1} \sin(m\theta_{2k-1}) \cdot (-1)^{k-1}$$

$$= \frac{2i}{J+1} \sum_{k=1}^{\nu} \sin[(2k-1)\theta_m]$$

$$= \frac{2i}{J+1} I_m \qquad \text{(E20)}$$

where

$$\left.\begin{aligned} I_m &= \mathrm{Im}(\sigma_m) \\ \sigma_m &= \sum_{k=1}^{\nu} \exp[i(2k-1)\theta_m] \end{aligned}\right\} \qquad \text{(E21)}$$

One first computes σ_m as follows:

$$\sigma_m = \exp(i\theta_m) \sum_{q=0}^{\nu-1} \exp[q(2i\theta_m)]$$

$$= \exp(i\theta_m) \frac{\exp(2\nu i\theta_m) - 1}{\exp(2i\theta_m) - 1}$$

$$= \frac{\exp(2\nu i\theta_m) - 1}{2i \sin \theta_m} \qquad \text{(E22)}$$

Note that $2\nu\theta_m = \pi m$, so that

$$\sigma_m = \frac{(-1)^m - 1}{2i \sin \theta_m} \qquad \text{(E23)}$$

$$I_m = \frac{1 + (-1)^{m+1}}{2 \sin \theta_m} \qquad \text{(E24)}$$

and

$$J_{m\nu} = \frac{i}{J+1} \frac{1 + (-1)^{m+1}}{\sin \theta_m} \tag{E25}$$

One now applies Equation (E11) to the case $j = \nu$ to get:

$$\begin{aligned} w_{\nu\nu} &= \sum_{m=1}^{J} |U_{m\nu}|^2 \cos^2 \theta_m \\ &= \sum_{m=1}^{J} |U_{m\nu}|^2 - \sum_{m=1}^{J} |U_{m\nu}|^2 \sin^2 \theta_m \\ &= 1 - \frac{1}{(J+1)^2} \sum_{m=1}^{J} \left[1 + (-1)^{m+1}\right]^2 \\ &= 1 - \frac{2}{J+1} \end{aligned} \tag{E26}$$

where Equations (E12) and (E17) have been used. Equation (E26) validates the statement made in Equation (E18) (Q.E.D.).

XIII. APPENDIX F: STABILITY CONDITION FOR SPECIFIED BOUNDARY DATA AND SMALL COURANT NUMBERS

The stability analysis developed in Section IIB3, has shown that when the data are specified at the boundaries, the following approximate stability condition applies to the case where the Courant numbers ν_x and ν_y are small:

$$2(1 + \varepsilon_i d_j)(1 + \varepsilon_i d_k) - \varepsilon_e(d_j' + d_k') \geq 0 \tag{F1}$$

where

$$\left.\begin{aligned} d_j &= 2(1 + \cos\theta_j) \\ d_k &= 2(1 + \cos\theta_k) \end{aligned}\right\} \tag{F2}$$

in which $\theta_j = \pi j/(J+1)$ $(j = 1, 2, \ldots, J)$ and $\theta_k = \pi k/(K+1)$ $(k = 1, 2, \ldots, K)$, while

$$\left.\begin{aligned} d_j' &= d_j \text{ or } d_j^2 \\ d_k' &= d_k \text{ or } d_k^2 \end{aligned}\right\} \tag{F3}$$

depending on whether second-order or fourth-order smoothing is applied. In this appendix, the condition expressed in Equation (F1) is explicited for these two cases. For this, one lets

$$\left.\begin{aligned} d_j &= 4\xi \\ d_k &= 4\eta \\ \varepsilon_i &= \varepsilon/4 \end{aligned}\right\} \tag{F4}$$

and

$$\left.\begin{aligned} d_j' &= 4\xi \text{ or } 16\xi^2 \\ d_k' &= 4\eta \text{ or } 16\eta^2 \\ \varepsilon_e &= \bar{\varepsilon}/4 \text{ or } \bar{\varepsilon}/16 \end{aligned}\right\} \tag{F5}$$

In this manner, the new variables ξ and η, which are not subscripted for notational simplicity, take their values in the interval $[0,1]$, and Equation (F1) becomes:

$$2(1 + \varepsilon\xi)(1 + \varepsilon\eta) - \bar{\varepsilon}(\xi^{p/2} + \eta^{p/2}) \geq 0 \tag{F6}$$

where p is the order of the smoothing applied ($p = 2$ or 4).

A. Second-Order Smoothing

In the particular case where second-order smoothing is applied ($p = 2$), Equation (F6) becomes, after rearrangement:

$$\bar{\varepsilon} - 2\varepsilon \leq 2\,\frac{1 + \varepsilon^2\xi\eta}{\xi + \eta} \tag{F7}$$

Hence, we are led to determine the function:

$$\left.\begin{aligned} f(\varepsilon) = \text{Inf} \quad & g(\xi,\eta) \\ & 0 \leq \xi,\ \eta \leq 1 \end{aligned}\right\} \tag{F8}$$

where

$$g(\xi,\eta) = 2\,\frac{1 + \varepsilon^2\xi\eta}{\xi + \eta} \tag{F9}$$

and to write the stability conditions as follows:

$$\bar{\varepsilon} - 2\varepsilon \leq f(\varepsilon) \tag{F10}$$

For this, one lets

$$\left.\begin{aligned} x &= \xi + \eta \\ y &= \eta - \xi \end{aligned}\right\} \tag{F11}$$

In this way, the domain of study (see Figure 19) is now defined by:

$$\left.\begin{aligned} 0 \leq x \leq 2 \\ |y| \leq y_{max}(x) \end{aligned}\right\} \tag{F12}$$

where $y_{max}(x) = x$ or $2 - x$ depending on whether $x \leq 1$ or $x \geq 1$. Computing

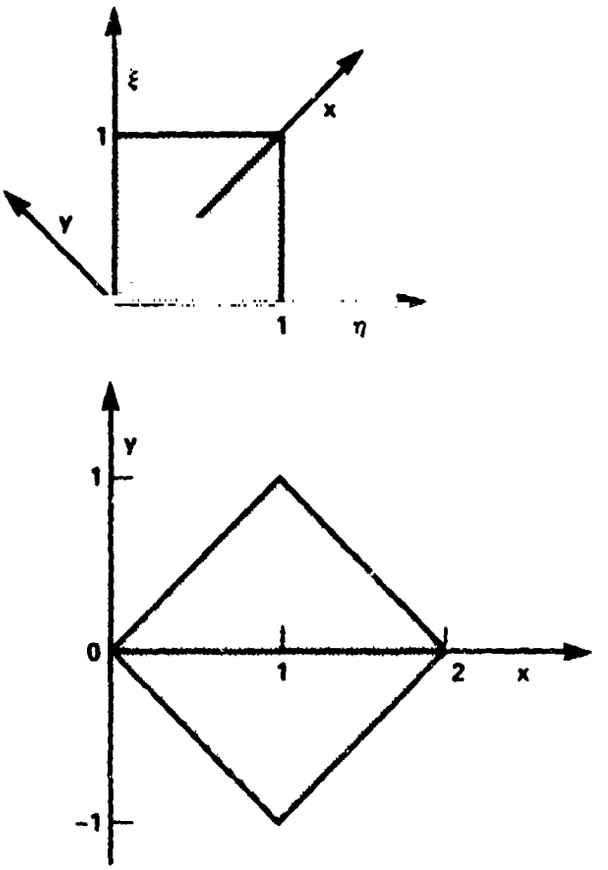


Figure 19.- Domain of study in the (x,y) plane.

$$g(\xi,\eta) = h(x,y) = \frac{4 + \varepsilon^2(x^2 - y^2)}{2x} \tag{F13}$$

indicates that for given x, $h(x,y)$ is minimum at $y = y_{max}(x)$. Let $\phi(x) = h[x,y_{max}(x)]$. For $0 \le x \le 1$, $y_{max}(x) = x$ so that $\phi(x) = 2/x$ which achieves a minimum value of 2 at $x = 1$. For $x \ge 1$, $y_{max}(x) = 2 - x$ so that:

$$\phi(x) = \frac{4 + \varepsilon^2[x^2 - (2 - x)^2]}{2x}$$

$$= \frac{2(1 - \varepsilon^2)}{x} + 2\varepsilon^2 \tag{F14}$$

Hence, if $\varepsilon \ge 1$, $\phi(x)$ is nondecreasing over the interval $[1,2]$, so that $\phi(x)$ is actually minimum at $x = 1$. This gives

$$f(\varepsilon) = 2 \quad \text{for} \quad \varepsilon \ge 1 \tag{F15}$$

If now, instead, $\varepsilon < 1$, $\phi(x)$ decreases when x increases from 1 to 2, and $\phi(x)$ achieves its minimum at $x = 2$, so that:

$$f(\varepsilon) = 1 + \varepsilon^2 \quad \text{for} \quad \varepsilon < 1 \tag{F16}$$

Combining these results with Equation (F10) yields the following conditions for stability:

$$\left.\begin{aligned} \varepsilon &\ge \sqrt{\bar{\varepsilon}} - 1 \quad \text{for} \quad \bar{\varepsilon} \le 4 \\ \varepsilon &\ge \frac{\bar{\varepsilon}}{2} - 1 \quad \text{for} \quad \bar{\varepsilon} \ge 4 \end{aligned}\right\} \tag{F17}$$

Replacing ε and $\bar{\varepsilon}$ by their expressions in terms of ε_i and ε_e given in Equations (F4) and (F5), yields the desired equation (Equation (56)). Finally, the remark that was made on Equation (C16) in Chapter IX also applied to Equation (F16).

B. Fourth-Order Smoothing

In the particular case where fourth-order smoothing is applied ($p = 4$), Equation (F6) becomes, after rearrangement:

$$\bar{\varepsilon} \leq \frac{2(1 + \varepsilon\xi)(1 + \varepsilon\eta)}{\xi^2 + \eta^2} \tag{F18}$$

Hence, if one defines, for this case, a function $g(\xi,\eta)$ by

$$g(\xi,\eta) = \frac{2(1 + \varepsilon\xi)(1 + \varepsilon\eta)}{\xi^2 + \eta^2} \tag{F19}$$

and a function $f(\varepsilon)$ by Equation (F8) the stability condition takes a form similar to Equation (F10), which is

$$\bar{\varepsilon} \leq f(\varepsilon) \tag{F20}$$

New variables x and y are also defined as in Equation (F11), and the domain of study is still given by Equation (F12). Here,

$$g(\xi,\eta) = h(x,y) = \frac{4 + \varepsilon^2(x^2 - y^2) + 4\varepsilon x}{x^2 + y^2} \tag{F21}$$

Note that for given x, $|y| \leq x$ in the domain, so that the numerator of $h(x,y)$ is positive and decreases when $|y|$ increases. The denominator is also positive but it increases with $|y|$. Thus again:

$$\left.\begin{aligned} f(\varepsilon) &= \text{Inf } \phi(x) \\ 0 &\leq x \leq 2 \end{aligned}\right\} \tag{F22}$$

where

$$\phi(x) = h[x, y_{max}(x)] \tag{F23}$$

For $0 \leq x \leq 1$, $y_{max}(x) = x$, so that:

$$\phi(x) = \frac{2}{x^2} + \frac{2\varepsilon}{x} \tag{F24}$$

which achieves a minimum value of $2(1 + \varepsilon)$ at $x = 1$. For $1 \leq x \leq 2$, $y_{max}(x) = 2 - x$, so that:

$$\phi(x) = \frac{4 + \varepsilon^2[x^2 - (2 - x)^2] + 4\varepsilon x}{x^2 + (2 - x)^2}$$

$$= 2\,\frac{(\varepsilon^2 + \varepsilon)x + 1 - \varepsilon^2}{x^2 - 2x + 2} \qquad \text{(F25)}$$

and

$$\frac{(x^2 - 2x + 2)^2}{2}\,\phi'(x) = (\varepsilon^2 + \varepsilon)(x^2 - 2x + 2) - 2(x - 1)[(\varepsilon^2 + \varepsilon)x + 1 - \varepsilon^2]$$

$$= -(\varepsilon^2 + \varepsilon)x^2 + 2(\varepsilon^2 - 1)x + 2(1 + \varepsilon) \qquad \text{(F26)}$$

Note that $\phi'(x)$ has two real zeros given by:

$$x_1, x_2 = \frac{\varepsilon^2 - 1 \pm \sqrt{(\varepsilon^2 - 1)^2 + 2\varepsilon(1 + \varepsilon)^2}}{\varepsilon^2 + \varepsilon} \qquad \text{(F27)}$$

Clearly $x_1 < 0$ and $x_2 > 0$, and $\phi(x)$ changes sign at most one time (at x_2) in the interval $[1,2]$. Since, also, $\phi(1) = 2\varepsilon(\varepsilon + 1) \geq 0$, $\phi(x)$ achieves its minimum at either $x = 1$ or $x = 2$. Computing

$$\left.\begin{aligned} \phi(1) &= 2(\varepsilon + 1) \\ \phi(2) &= (\varepsilon + 1)^2 \end{aligned}\right\} \qquad \text{(F28)}$$

indicates that $\phi(1) \leq \phi(2)$ when $\varepsilon \geq 1$. In view of Equations (F20) and (F22), the stability condition is written as follows:

$$\left.\begin{aligned} \varepsilon &\geq \sqrt{\bar{\varepsilon}} - 1 \quad \text{for} \quad \bar{\varepsilon} \leq 4 \\ \varepsilon &\geq \frac{\bar{\varepsilon}}{2} - 1 \quad \text{for} \quad \bar{\varepsilon} \geq 4 \end{aligned}\right\} \qquad \text{(F29)}$$

Replacing ε and $\bar{\varepsilon}$ by their expressions in terms of ε_i and ε_e given in Equations (F4) and (F5), yields the desired equation (Equation (57)).

The remark that was made on Equations (D19) and (D23) in Chapter X (Remark 1) also applied to Equation (F29). Also note that Equation (F29) is identical to Equation (F16) for a reason given in Chapter X (Remark 2).

XIV. APPENDIX G: ON THE EIGENVALUES OF THE MATRIX Γ

The stability analysis of Section IIB was developed for the case of a scalar, linear partial-differential equation with constant coefficients. In this way, the convective derivative operator C_x, then (real) skew-symmetric, had purely imaginary eigenvalues, and this property was crucial to the derivation. In Chapter IV it was shown that for the Euler equations, this property is most likely to be true when the following matrix

$$\Gamma_J = \begin{bmatrix} 0 & a_2 & & & & \\ a_1 & 0 & a_3 & & & \\ & a_2 & 0 & a_4 & & \\ & & \ddots & \ddots & \ddots & \\ & & & a_{J-2} & 0 & a_J \\ & & & & a_{J-1} & 0 \end{bmatrix} \tag{G1}$$

has real eigenvalues. (In Equation (G1), a_j represents the value at $x_j = (j-1)\Delta x$ $(j = 1, 2, \ldots, J)$ of any one of the four eigenvalues of the Jacobian matrix A (for 2-D flows), and Γ_J is subscripted to indicate its dimension.) In this appendix, the question of whether Γ_J has real eigenvalues is investigated.

Three cases will be examined successively. They are: Case I — $a_1, a_2, \ldots, a_J$ are all nonzero and of the same sign; Case II — there is at least one true zero in the sequence a_j at every alternation of sign; and Case III — there is at least one value of j for which $a_j a_{j+1} < 0$.

All three cases could be studied by analyzing the characteristic polynomial of Γ_J, that is

$$\Delta_J(\lambda) = \det(\Gamma_J - \lambda I_J) \tag{G2}$$

where I_J is the $J \times J$ identity matrix, but for Cases I and II simpler arguments, which yield the desired information, are preferred here.

For Case I, define a sequence δ_j by:

$$\begin{aligned} \delta_1 &= 1 \\ \delta_{j+1} &= \delta_j \sqrt{a_j / a_{j+1}} \qquad (j = 1, 2, \ldots, J-1) \end{aligned} \tag{G3}$$

Clearly the sequence δ_j is well defined, and none of its elements is zero, so that the diagonal matrix

$$\nabla = \mathrm{Diag}(\delta_j) \tag{G4}$$

is nonsingular, and

$$\nabla^{-1} = \mathrm{Diag}(\delta_j^{-1}) \tag{G5}$$

Then perform on the matrix Γ_J the following similarity transformation:

$$\begin{aligned} \tilde{\Gamma}_J &= \nabla^{-1} \Gamma_J \nabla \\ &= \mathrm{Diag}(\delta_j^{-1}) \mathrm{Trid}(a_{j-1}, 0, a_{j+1}) \mathrm{Diag}(\delta_j) \\ &= \mathrm{Trid}(\delta_j^{-1} a_{j-1} \delta_{j-1}, 0, \delta_j^{-1} a_{j+1} \delta_{j+1}) \qquad \text{(G6)} \\ &= \mathrm{Trid}\left(\sqrt{a_{j-1} a_j},\ 0,\ \sqrt{a_j a_{j+1}}\right) \qquad \text{(G6)} \end{aligned}$$

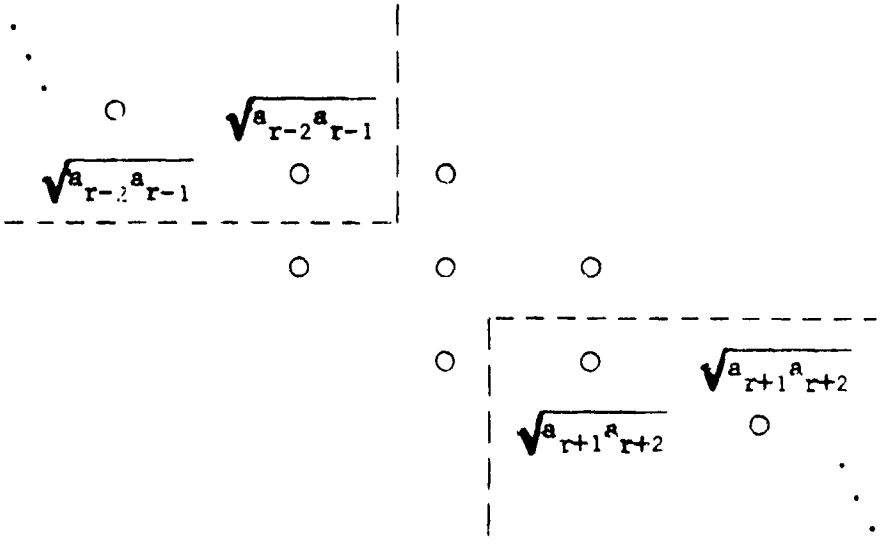
The matrix $\tilde{\Gamma}_J$ is found real symmetric. As so, it is diagonalized by an orthogonal transformation Ω and has real eigenvalues. These eigenvalues are also those of the matrix Γ_J which is similar to the matrix $\tilde{\Gamma}_J$. This case is hence favorable to the stability of the implicit algorithm. (More information about the eigenvalues of the matrix Γ_J, in Case I, is given in the remarks at the end of this appendix.)

Using a continuity argument, Case II can be treated as an extension of Case I. For this, an undetermined scalar parameter x is substituted in place of every zero appearing in the original sequence a_j, all the other

elements of this sequence being maintained the same. In this way, the matrix Γ_J becomes a first-degree polynomial of x whose eigenvalues for $x = 0$ are required. The coefficients of the characteristic polynomial $\Delta_J(\lambda)$ are simply some polynomials of x, that is, continuous functions of x, so that:

$$\Delta_J(\lambda)\Big|_{x=0} = \lim_{x\to 0} \Delta_J(\lambda) \tag{G7}$$

But for $x \neq 0$, Equations (G3) through (G6) apply. Hence, for $x = 0$, the matrix $\tilde{\Gamma}_J$ (here defined by the last line in Equation (G6)) and the matrix Γ_J may not be similar, but have the same characteristic polynomial, and thus the same eigenvalues. To analyze these eigenvalues, suppose, for example, that $a_r = 0$ for some r. In the neighborhood of this element, the matrix $\tilde{\Gamma}_J$ has the following structure:



Here, the matrix $\tilde{\Gamma}_J$ is found block-diagonal. Any of these blocks is a real symmetric (tridiagonal) matrix (because consecutive nonzero elements of the sequence a_j are of the same sign), and separated from the next one by at least one row and one column of zeros. In view of this, let Ω_k be

the (orthogonal) matrix that diagonalizes the kth block, and Ω be the (orthogonal) block-diagonal matrix where the blocks are taken to be Ω_1, I_{m_1}, Ω_2, I_{m_2}, Ω_3, etc., where I_m is the $m \times m$ identity matrix and m_k the number of zeros that separate positive from negative elements of the sequence a_j at the kth alternation of sign. Clearly the matrix Ω diagonalizes the matrix $\tilde{\Gamma}_J$ whose eigenvalues are then found to be those of the blocks in the "diagonal" of the matrix $\tilde{\Gamma}_j$ taken together, with en plus the eigenvalue 0 added Σm_k times. These eigenvalues, which are also those of the matrix Γ_J, are all real as is desirable for stability.

For Case III the characteristic polynomial $\Delta_J(\lambda)$ of the matrix Γ_J needs to be analyzed. For this analysis, expand the determinant of the matrix

$$\Gamma_{J+1} - \lambda I_{J+1} = \left[\begin{array}{c|c} \Gamma_J - \lambda I_J & \\ & a_{J+1} \\ \hline a_J & -\lambda \end{array}\right] \tag{G8}$$

along its last column to get:

$$\Delta_{J+1}(\lambda) = -\lambda\Delta_J(\lambda) - a_{J+1} \det\left[\begin{array}{c|c} \Gamma_{J-1} - \lambda I_{J-1} & \\ & a_J \\ \hline & a_J \end{array}\right]$$

and finally:

$$\Delta_{J+1}(\lambda) = -\lambda\Delta_J(\lambda) - a_J a_{J+1}\Delta_{J-1}(\lambda) \tag{G9}$$

From this result, one can derive the following formulas:

$$\Delta_{2m}(\lambda) = P_m(\lambda^2) \tag{G10a}$$

$$\Delta_{2m+1}(\lambda) = -\lambda Q_m(\lambda^2) \tag{G10b}$$

where m is a natural integer, and $P_m(x)$ and $Q_m(x)$ are polynomials of degree m, with leading term x^m, and which satisfy the following recurrence formulas:

$$P_{m+1}(x) = xQ_m(x) - a_{2m+1}a_{2m+2}P_m(x) \tag{G11a}$$

$$Q_{m+1}(x) = P_{m+1}(x) - a_{2m+2}a_{2m+3}Q_m(x) \tag{G11b}$$

To see this, use induction. Compute

$$\Delta_2(\lambda) = \begin{vmatrix} -\lambda & a_1 \\ a_2 & -\lambda \end{vmatrix}$$

$$= \lambda^2 - a_1a_2 \tag{G12}$$

and

$$\Delta_3(\lambda) = \begin{vmatrix} -\lambda & a_2 & 0 \\ a_1 & -\lambda & a_3 \\ 0 & a_2 & -\lambda \end{vmatrix}$$

$$= \lambda(\lambda^2 - a_2a_3) - a_2(-a_1\lambda)$$

$$= -\lambda[\lambda^2 - (a_1a_2 + a_2a_3)] \tag{G13}$$

Clearly, if one defines:

$$\left.\begin{aligned} \Delta_0(\lambda) &= 1 \\ P_0(x) &= Q_0(x) = 1 \\ P_1(x) &= x - a_1a_2 \\ Q_1(x) &= x - (a_1a_2 + a_2a_3) \end{aligned}\right\} \tag{G14}$$

Equation (G10) holds for $m = 0$ and $m = 1$, and Equation (G11) holds for $m = 0$. Now suppose that Equation (G10) holds for $m = r$ for some $r \geq 1$, and Equation (G11) holds for $m = r - 1$. Then applying Equation (G9) with $J = 2r + 1$ gives:

$$\Delta_{2r+2}(\lambda) = -\lambda\Delta_{2r+1}(\lambda) - a_{2r+1}a_{2r+2}\Delta_{2r}(\lambda)$$

$$= \lambda^2 Q_r(\lambda^2) - a_{2r+1}a_{2r+2}P_r(\lambda^2) \qquad \text{(G15)}$$

Hence, Equation (G10a) holds for $m = r + 1$ provided $P_{r+1}(x)$ is defined according to Equation (G11a) in which m is set equal to r. It also appears from this equation, that since $P_r(x)$ and $Q_r(x)$ are polynomials of degree r, with leading term x^r, $P_{r+1}(x)$ is a polynomial of degree $r + 1$, with leading term x^{r+1}. Similarly,

$$\Delta_{2r+3}(\lambda) = -\lambda\Delta_{2r+2}(\lambda) - a_{2r+2}a_{2r+3}\Delta_{2r+1}(\lambda)$$

$$= -\lambda[P_{r+1}(\lambda^2) + a_{2r+2}a_{2r+3}Q_r(\lambda^2)] \qquad \text{(G16)}$$

Hence, Equation (G10b) holds for $m = r + 1$ provided $P_{r+1}(x)$ is defined according to Equation (G11b) in which m is set equal to r. It also appears from this equation that since $P_{r+1}(x)$ and $Q_r(x)$ are polynomials of degrees $r + 1$ and r, respectively, with leading terms x^{r+1} and x^r, respectively, $Q_{r+1}(x)$ is a polynomial of degree $r + 1$ with leading term x^{r+1}. This shows that Equation (G10) holds for $m = r + 1$ and Equation (G11) holds for $m = r$; hence, they both hold for every value of m. (Q.E.D.)

In view of Equation (G10), the following two conclusions can be drawn: (1) if λ is an eigenvalue of the matrix Γ_J, then $-\lambda$ is also an eigenvalue; and (2) the eigenvalues of the matrix Γ_J are all real if and only if the roots of $P_m(x)$, in case $J = 2m$, or $Q_m(x)$, in case $J = 2m + 1$, are all real positive.

The first result could be derived directly for the block matrix with the same structure. The second one implies that a necessary condition for the matrix Γ_J to have only real roots, is that the coefficients of $P_m(x)$,

in case $J = 2m$, or $Q_m(x)$, in case $J = 2m + 1$, alternate in sign, or equivalently, that the coefficients of $\alpha_m^{(\nu)}$ or $\beta_m^{(\nu)}$, respectively, given by

$$\left.\begin{aligned} \alpha_m^{(\nu)} &= (-1)^{m-\nu} P_m^{(\nu)}(0)/\nu! \\ \beta_m^{(\nu)} &= (-1)^{m-\nu} Q_m^{(\nu)}(0)/\nu! \end{aligned}\right\} \tag{G17}$$

where $\nu = 0, 1, 2, \ldots, m$, be all positive. Recurrence formulas for these coefficients can easily be obtained using Equation (G11). In particular, setting $x = 0$ in this equation and multiplying the result by $(-1)^{m+1}$ gives:

$$\alpha_{m+1}^{(o)} = a_{2m+1} a_{2m+2} \alpha_m^{(o)} \tag{G18a}$$

$$\beta_{m+1}^{(o)} = \alpha_{m+1}^{(o)} + a_{2m+2} a_{2m+3} \beta_m^{(o)} \tag{G18b}$$

Similarly, differenciating Equation (G11) ν times ($\nu \geq 1$) with respect to x, setting $x = 0$, and multiplying the result by $(-1)^{m+1-\nu}/\nu!$ give:

$$\alpha_{m+1}^{(\nu)} = \beta_m^{(\nu-1)} + a_{2m+1} a_{2m+2} \alpha_m^{(\nu)} \tag{G19a}$$

$$\beta_{m+1}^{(\nu)} = \alpha_{m+1}^{(\nu)} + a_{2m+2} a_{2m+3} \beta_m^{(\nu)} \tag{G19b}$$

which in fact, in view of Equation (G18), can be applied with $\nu = 0$ if one defines $\beta_m^{(-1)} = 0$. Equation (G19) should be completed by the following "boundary" conditions:

$$\alpha_m^{(m)} = \beta_m^{(m)} = 1 \tag{G20}$$

which simply state that x^m is the leading term of both $P_m(x)$ and $Q_m(x)$.

From this, it is easy to derive explicit formulas for $\alpha_m^{(\nu)}$ and $\beta_m^{(\nu)}$ for $\nu = 0$ and 1, now denoted more simply by α_m, β_m, α_m', and β_m', respectively. In particular, applying Equation (G18a) recursively gives:

$$\begin{aligned}\alpha_m &= a_{2m-1}a_{2m}\alpha_{m-1} \\ &= a_{2m-3}a_{2m-2}a_{2m-1}a_{2m}\alpha_{m-2} \\ &\cdot \\ &\cdot \\ &\cdot \\ &= a_1a_2 \cdot \cdot \cdot a_{2m}\alpha_o \\ &= a_1a_2 \cdot \cdot \cdot a_{2m} \end{aligned} \tag{G21}$$

where Equation (G20) has been used with $m = 0$. Now, write Equation (G18b) with $m = k - 1$ ($k = 1, 2, \ldots, m$) and multiply the result by $a_{2k+2}a_{2k+3} \cdot \cdot \cdot a_{2m+1}$. This gives

$$a_{2k+2}a_{2k+3} \cdot \cdot \cdot a_{2m+1}\beta_k = a_{2k}a_{2k+1} \cdot \cdot \cdot a_{2m+1}\beta_{k-1} + \frac{a_1a_2 \cdot \cdot \cdot a_{2m+1}}{a_{2k+1}} \tag{G22}$$

where Equation (G21) has been used. (For $k = m$, the coefficient of β_m in this equation is understood to be one.) Then, write Equation (G22) for $k = 1, 2, \ldots, m$, add the resulting relationships, use Equation (G20) with $m = 0$, and simplify the result to get:

$$\beta_m = a_1a_2 \cdot \cdot \cdot a_{2m+1}\sum_{k=0}^{m}\frac{1}{a_{2k+1}} \tag{G23}$$

Similarly, write Equation (G19a) with $m = k - 1$ ($k = 1, 2, \ldots, m$) and $\nu = 1$, and multiply the result by $a_{2k+1}a_{2k+2} \cdot \cdot \cdot a_{2m}$ to get:

$$a_{2k+1}a_{2k+2} \cdot \cdot \cdot a_{2m}\alpha'_k = a_{2k-1}a_{2k} \cdot \cdot \cdot a_{2m}\alpha'_{k-1} + \frac{a_1a_2 \cdot \cdot \cdot a_{2m}}{a_{2k}}\sum_{j=1}^{k}\frac{1}{a_{2j-1}} \tag{G24}$$

where Equation (G23) has been used. Then, write Equation (G24) for $k = 1, 2, \ldots, m$ and add the resulting relationships to get:

$$\alpha'_m = a_1 a_2 \cdots a_{2m} \sum_{k=1}^{m} \frac{1}{a_{2k}} \sum_{j=1}^{k} \frac{1}{a_{2j-1}} \tag{G25}$$

where the fact that $\alpha'_0 = 0$ has been used. Finally, write Equation (G19b) with $m = \ell - 1$ ($\ell = 1, 2, \ldots, m$) and $\nu = 1$, and multiply the result by $a_{2\ell+2} a_{2\ell+3} \cdots a_{2m+1}$ to get:

$$a_{2\ell+2} a_{2\ell+3} \cdots a_{2m+1} \beta'_\ell = a_{2\ell} a_{2\ell+1} \cdots a_{2m+1} \beta'_{\ell-1} + \frac{a_1 a_2 \cdots a_{2m+1}}{a_{2\ell+1}} \sum_{k=1}^{\ell} \frac{1}{a_{2k}} \sum_{j=1}^{k} \frac{1}{a_{2j-1}} \tag{G26}$$

where Equation (G25) has been used. Then write Equation (G26) for $\ell = 1, 2, \ldots, m$ and add the resulting relationships to get:

$$\beta'_m = a_1 a_2 \cdots a_{2m+1} \sum_{\ell=1}^{m} \frac{1}{a_{2\ell+1}} \sum_{k=1}^{\ell} \frac{1}{a_{2k}} \sum_{j=1}^{k} \frac{1}{a_{2j-1}} \tag{G27}$$

where the fact that $\beta'_0 = 0$ has been used.

In view of the Equations (G21), (G23), (G25), and (G27), it appears that in the event one or several alternations of sign occur in the sequence a_j, without separation of positive from negative values by at least one zero, it is most unlikely that the coefficients $\alpha_m^{(\nu)}$ or $\beta_m^{(\nu)}$ are all positive. It suffices that only one of these coefficients be negative, for $P_m(x)$ or $Q_m(x)$ to have at least one root x^* which is not real positive; then, the matrix Γ_J has, in particular, the eigenvalues $\pm\sqrt{x^*}$; these are complex and one of the two has a positive imaginary part; to that one corresponds an eigenvalue of the matrix C_x (see Equation (70)) with a negative real part

which has a destabilizing effect, since it acts like a negative smoothing. More precisely, suppose for example, that $J = 2m$ and that $\alpha_m < 0$. This occurs for example when $a_1, a_2, \ldots, a_{2r}$ are negative for some odd value of r. Then let $r_1, r_2, \ldots$ be the real roots of $P_m(x)$ and $z_1, \bar{z}_1, z_2, \bar{z}_2, \ldots$ be its complex roots. The coefficient α_m is simply the product of these roots so that:

$$\begin{aligned} \alpha_m &= r_1 r_2 \ldots (z_1\bar{z}_1)(z_2\bar{z}_2) \ldots \\ &= r_1 r_2 \ldots |z_1|^2 \, |z_2|^2 \ldots < 0 \end{aligned} \tag{G28}$$

This shows that $P_m(x)$ then has an odd number of real negative roots $n_1, n_2, \ldots$ to which correspond an odd number of pairs of purely imaginary eigenvalues $i\sqrt{-n_1}$ and $-i\sqrt{-n_1}$, $i\sqrt{-n_2}$, and $-i\sqrt{-n_2}$, $\ldots$ for the matrix Γ_J and the real eigenvalues $-\sqrt{-n_1}$ and $\sqrt{-n_1}$, $-\sqrt{-n_2}$ and $\sqrt{-n_2}$, $\ldots$ for the matrix C_x. A similar situation occurs if $J = 2m + 1$ and $\beta_m < 0$.

Remark 1:

The fact that the eigenvalues of the matrix Γ_J are real for Cases I and II can be derived from Equation (G10) and (G11). In fact, this result was originally obtained in this way. In that first analysis, the following separation properties were found to be true for Case I:

$$\left.\begin{aligned} &0 < x_1 < y_1 < x_2 < y_2 < \ldots < x_m < y_m \\ &0 < x_1' < x_1 < x_2' < x_2 < \ldots < x_m' < x_m < x_{m+1}' \\ &0 < y_1' < y_1 < y_2' < y_2 < \ldots < y_m' < y_m < y_{m+1}' \end{aligned}\right\} \tag{G29}$$

where $\{x_j\}$, $\{y_j\}$ $(j = 1, 2, \ldots, m)$, $\{x_j'\}$ and $\{y_j'\}$ $(j = 1, 2, \ldots, m + 1)$ are the roots of $P_m(x)$, $Q_m(x)$, $P_{m+1}(x)$ and $Q_{m+1}(x)$, respectively. The derivation of Equation (G29) is omitted here, since no particular application of it was found.

However, Equation (G29) shows that the eigenvalues of the matrix Γ_J and thus of the matrix C_x are simple, which implies that both matrices are diagonalizable, an already known fact.

Remark 2:

Consider the case where the Euler implicit method is applied to the following one-dimensional (generalized) wave-equation:

$$u_t + [a(x,t)u]_x = 0 \tag{G30}$$

If no smoothing is applied, the solution-vector u^{n+1} at the $n+1$st time-step is given by:

$$u^{n+1} = Lu^n \tag{G31}$$

where the following definitions are made:

$$\left.\begin{aligned} L &= \left(I + \frac{\Delta t}{2\Delta x} C_x^{n+1}\right)^{-1} \\ C_x^{n+1} &= \mathrm{Trid}\left(-a_{j-1}^{n+1},\ 0,\ a_{j+1}^{n+1}\right) \\ &= iD\Gamma_J^{n+1}\bar{D} \\ D &= \mathrm{Diag}(i^j) \end{aligned}\right\} \tag{G32}$$

Suppose that a uniform bound on u^n is required. For this, assume that $a(x,t) > 0$ so that the results of Case I are applicable. In particular, the transformations that diagonalize the matrices $\tilde{\Gamma}_J^{n+1}$, Γ_J^{n+1}, and C_x^{n+1} (or L) are, respectively, Ω, $\nabla\Omega$, and $X = D\nabla\Omega$ where Ω is some orthogonal matrix, and the diagonal matrices ∇ and D are given by Equations (C4) and (G32), respectively. Consider first the case where $a(x,t)$ does not depend on time, so that L does not either. Then

$$\begin{aligned} \|u^n\| &= \|(X\Lambda X^{-1})^n u^o\| = \|X\Lambda^n X^{-1} u^o\| \\ &\le \nu \|\Lambda\|^n \|u^o\| \end{aligned} \tag{G33}$$

where $\nu = \|X\| \cdot \|X^{-1}\|$ is a condition number, and Λ is the diagonalized form of the matrix L whose eigenvalues are given by:

$$\lambda_j = \frac{1}{1 + i\gamma_j} \tag{G34}$$

where γ_j is an eigenvalue of the matrix Γ_J, that is, a real number, so that $|\lambda_j| \leq 1$. If the euclidean norm is selected, $\|\Lambda\| \leq 1$ so that

$$\|u^n\| \leq \nu \|u^o\| \tag{G35}$$

which is the bound we were looking for. The value of ν has no importance in this case.

However, if now $a(x,t)$ does depend on time, it appears inevitable to use the following bound for L:

$$\|L\| = \|X\Lambda X^{-1}\| \leq \bar{\nu} \tag{G36}$$

where $\bar{\nu}$ is an upper bound for ν (which depends on n). This gives:

$$\|u^n\| \leq \bar{\nu}^n \|u^o\| \tag{G37}$$

It hence appears of interest to evaluate $\bar{\nu}$. For this, recall that Ω is orthogonal and D unitary, so that:

$$\left.\begin{aligned} \|X\| &= \|D\nabla\Omega\| = \|\nabla\| = \operatorname*{Max}_j |\delta_j^n| \\ \|X^{-1}\| &= \|\Omega^t \nabla^{-1} \bar{D}\| = \|\nabla^{-1}\| = 1/\operatorname*{Min}_j |\delta_j^n| \end{aligned}\right\} \tag{G38}$$

where euclidean norm has been used. This gives

$$\begin{aligned} \bar{\nu} &= \operatorname*{Sup}_n [\operatorname*{Max}_j |\delta_j^n| / \operatorname*{Min}_j |\delta_j^n|] \\ &= \operatorname*{Sup}_n \sqrt{\operatorname*{Max}_j |a_j^n| / \operatorname*{Min}_j |a_j^n|} \end{aligned} \tag{G39}$$

where Equation (G3) has been used. Clearly, $\bar{\nu} \geq 1$ unless $a(x,t)$ only depends on time, and no uniform bound for u^n can be derived from Equation (G37), which, to be rigorous, only reveals the failure of this attempt.

However, for a practical problem, one anticipates $\bar{\nu}$ to be very large if the eigenvalues a_j^n are themselves subject to large variations. This suggests that despite the fact that Equation (G36) is a conservative estimate, the operator L of Equation (G31) is unlikely to be contracting as one would wish in order to apply the contraction-mapping theorem cited in Section IIA (i.e., contracting in the same norm sense for all n).

1 Report No	2. Government Accession No	3 Recipient's Catalog No
NASA TM-78495		

4 Title and Subtitle	5 Report Date
ON IMPROVING THE ITERATIVE CONVERGENCE PROPERTIES OF AN IMPLICIT APPROXIMATE-FACTORIZATION FINITE-DIFFERENCE ALGORITHM	6 Performing Organization Code

7 Author(s)	8 Performing Organization Report No
Jean-Antoine Desideri,* J. L. Steger,† and J. C. Tannehill*	A-7474
	10 Work Unit No: 505-15-31
9 Performing Organization Name and Address: *Iowa State University, Ames, Iowa 50011; †Ames Research Center, NASA, Moffett Field, Calif. 94035	11 Contract or Grant No: NCA2-OR340-706
	13 Type of Report and Period Covered: Technical Memorandum
12 Sponsoring Agency Name and Address: National Aeronautics and Space Administration, Washington, D. C. 20546	14 Sponsoring Agency Code

15 Supplementary Notes

16 Abstract

The iterative convergence properties of a currently popular approximate-factorization implicit finite-difference algorithm are analyzed both theoretically and numerically. Modifications to the base algorithm are made to remove the inconsistency in the original implementation of artificial dissipation. In this way, the steady-state solution becomes independent of the time-step, and much larger time-steps can be used stably. To accelerate the iterative convergence, large time-steps and a cyclic sequence of time-steps are used. For a model transonic flow problem governed by the Euler equations, convergence is achieved with 10 times fewer time-steps with the modified differencing scheme. Finally, a particular form of instability due to variable coefficients is also analyzed.

17 Key Words (Suggested by Author(s))	18 Distribution Statement
Transonic flows, Implicit finite difference schemes, Stability, Artificial dissipation, Steady-state convergence rate	Unlimited STAR Category - 64

19 Security Classif (of this report)	20 Security Classif (of this page)	21 No of Pages	22 Price*
Unclassified	Unclassified	119	$5.50

*For sale by the National Technical Information Service, Springfield, Virginia 22161