NASA Contractor Report 178127

ICASE REPORT NO. 86-37

ICASE

PSEUDO-TIME ALGORITHMS FOR THE NAVIER-STOKES EQUATIONS

R. C. Swanson

E. Turkel

Contract Nos. NAS1-17070, NAS1-18107

May 1986

INSTITUTE FOR COMPUTER APPLICATIONS IN SCIENCE AND ENGINEERING
NASA Langley Research Center, Hampton, Virginia 23665

Operated by the Universities Space Research Association

National Aeronautics and
Space Administration

Langley Research Center
Hampton, Virginia 23665

PSEUDO-TIME ALGORITHMS FOR THE NAVIER-STOKES EQUATIONS

R. C. Swanson
NASA Langley Research Center

E. Turkel
Tel-Aviv University, Israel
and
Institute for Computer Applications in Science and Engineering

Dedicated to Milton E. Rose
on Occasion of his 60th Birthday

ABSTRACT

A pseudo-time method is introduced to integrate the compressible Navier-Stokes equations to a steady state. This method is a generalization of a method used by Crocco and also by Allen and Cheng. We show that for a simple heat equation that this is just a renormalization of the time. For a convection-diffusion equation the renormalization is dependent only on the viscous terms. We implement the method for the Navier-Stokes equations using a Runge-Kutta type algorithm. This enables the time step to be chosen based on the inviscid model only. We also discuss the use of residual smoothing when viscous terms are present.

Research was supported in part by the National Aeronautics and Space Administration under NASA Contract Nos. NAS1-17070 and NAS1-18107 while the second author was in residence at ICASE, NASA Langley Research Center, Hampton, VA 23665-5225.

I. INTRODUCTION

The solution of the compressible Navier-Stokes equations for flow about two- and three-dimensional complex aerodynamic configurations is still a time consuming problem on today's supercomputers. The resolution of the boundary layers requires the use of very fine meshes in the neighborhood of solid bodies. For a typical viscous flow the mesh can be several orders of magnitude finer (depending on the Reynolds number) than that required for an inviscid calculation. As an example, using a C-type mesh about an NACA 0012 airfoil, a typical mesh spacing near the body in the normal direction for an inviscid calculation is 1×10^{-2} chords. For a laminar viscous calculation with $Re = 5 \times 10^{3}$, this minimum cell height would be about 6×10^{-4} chords. For a turbulent calculation using an algebraic turbulence model and with $Re \simeq 3 \times 10^{6}$, the minimum cell height would be about 8×10^{-5} chords. In all cases a typical chordwise spacing at the midsection of the airfoil is about 5×10^{-2} chords.

Using an explicit method this fine mesh reduces the time step, due to stability requirements, that can be used. The time step restriction is caused by two factors. One contribution is due to the effect of the finer mesh on the inviscid portion of the calculation. When using an explicit method this reduction of the time step cannot be avoided without using a coarser mesh. It follows strictly from the need to include the entire domain of dependency in the numerical algorithm. Use of a local time step allows faster convergence to a steady state, but it does not remove the requirement to satisfy the convection stability condition in a local sense. A second difficulty is caused by the viscous terms. For an explicit method the time step is now dependent on the square of the mesh size rather than just the mesh size as

occurs for inviscid flow. Thus, even for a high Reynolds number flow the viscous time step will dominate when the mesh is sufficiently fine. In all these cases the use of an implicit scheme will alleviate the difficulties. In some ADI methods the Jacobian of the viscous terms is not used in the implicit portion of the code in order to improve the speed of the calculation [7]. We thus conclude that for both explicit and many implicit codes it is advantageous to account for the dependence of the time step on the viscous terms.

In this study we shall only discuss steady state problems which are solved by a pseudo time-dependent method. Hence, we can change all time derivatives as long as the steady state solution is not affected. One common device is to use a different time step in each zone. It is easier to calculate this local time step based on the inviscid equations. This provides an additional reason to eliminate the dependence of the time step on the viscous terms.

In this study we shall analyze a method used by Crocco [4] and also by Allen and Cheng [2]. They claim that the new scheme is unconditionally stable for a simple diffusion equation. We will show that in effect the scheme is a standard Euler forward-in-time central-in-space scheme. The time is artificially slowed down so as to satisfy the stability criterion. We then extend this scheme to the compressible Navier-Stokes equations using a Runge-Kutta scheme [9]. This modification enables us to choose our time step based on the inviscid equations. The modification automatically reduces the local time step in regions where the viscous time step is of importance. This enables us to use the inviscid time step in the far field while automatically accounting for viscous effects in the boundary layer. We will also look at residual smoothing for the heat equation.

II. SCALAR EQUATION

In this section we analyze and extend a scheme for the Navier-Stokes equations proposed by Crocco [4] and Allen-Cheng [2]. This scheme was also analyzed by Peyret and Viviand [6] and Roache [8], and we will extend their analysis.

We first consider the heat equation

$$w_t = \varepsilon w_{xx}. \tag{1}$$

The forward time centered space or Euler approximation to this scheme is given by

$$w_j^{n+1} = w_j^n + \frac{\varepsilon\Delta t}{(\Delta x)^2}\left(w_{j+1}^n - 2w_j^n + w_{j-1}^n\right). \tag{2}$$

This scheme is stable if

$$\nu = \frac{\varepsilon\Delta t}{(\Delta x)^2} \leq 1/2 \text{ , or } \Delta t \leq \frac{(\Delta x)^2}{2\varepsilon} . \tag{3}$$

Crocco, and Allen/Cheng introduce the inconsistent scheme

$$w_j^{n+1} = w_j^n + \frac{\varepsilon\Delta t}{(\Delta x)^2}\left(w_{j+1}^n - 2w_j^{n+1} + w_{j-1}^n\right). \tag{4}$$

This scheme is unconditionally stable. If we are only interested in the steady state, then (4) yields the correct steady-state solution. We now rewrite (4) as

$$\frac{w_j^{n+1} - w_j^n}{\Delta t} = \frac{\varepsilon}{(\Delta x)^2}\left(w_{j+1}^n - 2w_j^n + w_{j-1}^n\right) - \frac{2\varepsilon}{(\Delta x)^2}\left(w_j^{n+1} - w_j^n\right)$$

or

$$\left(\frac{1}{\Delta\tau} + \frac{1}{\Delta t}\right)\left(w_j^{n+1} - w_j^n\right) = \frac{\varepsilon}{(\Delta x)^2}\left(w_{j+1}^n - 2w_j^n + w_{j-1}^n\right) \tag{5}$$

with

$$\Delta\tau = \frac{(\Delta x)^2}{2\varepsilon} . \tag{6}$$

Thus, for this model problem the Crocco scheme is identical with the Euler scheme (2) with an artificial time step Δt_e given by

$$\frac{1}{\Delta t_e} = \frac{1}{\Delta t} + \frac{2\varepsilon}{(\Delta x)^2} . \tag{7}$$

Thus, the unconditional stability is achieved by slowing down the time process. Note that as $\Delta t \to \infty$, $\Delta t_e \to (\Delta x)^2/2\varepsilon$, i.e., the stability limit for the Euler method. So choosing a large time step for (4) is equivalent to choosing Δt_e at the stability limit for (2), and we have merely scaled the time. This can also be derived from the modified equation given in [6]. If ε or Δx is not constant, this also introduces a local time step.

We next consider the convection-diffusion equation

$$w_t = aw_x + \varepsilon w_{xx}. \tag{8}$$

The Crocco scheme now becomes

$$\frac{w_j^{n+1} - w_j^n}{\Delta t} = \frac{a\left(w_{j+1}^n - w_{j-1}^n\right)}{2\Delta x} + \frac{\varepsilon}{(\Delta x)^2}\left(w_{j+1}^n - 2w_j^{n+1} + w_{j-1}^n\right) \tag{9}$$

or

$$\left(\frac{1}{\Delta\tau} + \frac{1}{\Delta t}\right)\left(w_j^{n+1} - w_j^n\right) = \frac{a\left(w_{j+1}^n - w_{j-1}^n\right)}{2\Delta x} + \frac{\varepsilon}{(\Delta x)^2}\left(w_{j+1}^n - 2w_j^n + w_{j-1}^n\right) \tag{10}$$

with $\Delta\tau$ given by (6). Thus, again this is equivalent to the Euler scheme with a time scaling that depends only on the viscous terms. Allen and Cheng utilized this scheme within a time-marching scheme proposed by Brailovskaya [3]. We generalize this by considering a general N-stage Runge-Kutta scheme.

Consider the two-dimensional equation

$$w_t = Hw + \varepsilon_1 w_{xx} + \varepsilon_2 w_{yy} \tag{11}$$

where Hw describes the hyperbolic or first-order terms. In [9] we describe a Runge-Kutta scheme where the viscous terms are frozen for all the stages. This is similar in philosophy to the Brailovskaya scheme. Using the Crocco formulation the (K + 1)-st stage becomes

$$\begin{aligned}\frac{w_{j,k}^{(K+1)} - w_{j,k}^{n}}{\alpha_{K+1}\,\Delta t} = H_D\, w_{j,k}^{(K)} &+ \frac{\varepsilon_1}{(\Delta x)^2}\left(w_{j+1,k}^{n} - 2w_{j,k}^{(K+1)} + w_{j-1,k}^{n}\right) \\ &+ \frac{\varepsilon_2}{(\Delta y)^2}\left(w_{j,k+1}^{n} - 2w_{j,k}^{(K+1)} + w_{j,k-1}^{n}\right), \quad K=0,1,\cdots,N-1.\end{aligned} \tag{12}$$

This reduces to a Runge-Kutta scheme

$$w_{j,k}^{(K+1)} = w_{j,k}^{n} + \alpha_{K+1}\,\Delta t_e\left[H_D\, w_{j,k}^{(K)} + P_D\, w_{j,k}^{n}\right] \tag{13}$$

where H_D, P_D are the approximations to the hyperbolic and parabolic parts respectively and

$$\frac{1}{\Delta t_e} = \frac{1}{\Delta t} + \frac{2\varepsilon_1}{(\Delta x)^2} + \frac{2\varepsilon_2}{(\Delta y)^2}\,. \tag{14}$$

We slightly generalize (14) by redefining Δt_e by

$$\frac{1}{\Delta t_e} = \frac{1}{\Delta t} + 2\kappa\left(\frac{\varepsilon_1}{(\Delta x)^2} + \frac{\varepsilon_2}{(\Delta y)^2}\right) \tag{15}$$

where κ is a constant that we can choose. The form of (15) no longer follows directly from the Crocco formulation. Instead κ will be chosen based on a stability analysis.

We choose Δt in (12) or (15) based on the hyperbolic (inviscid) stability condition. We then find Δt_e from (15) and advance to stage $(K + 1)$ using the Runge-Kutta scheme (13).

The constant κ in (15) can be chosen so that we recover the parabolic stability limitation when $H_D = 0$. The exact value of κ depends on the coefficients α_K in the Runge-Kutta formula. In order to see this more clearly we revert to the one-dimensional convection-diffusion equation (8). We replace all space derivatives by second-order central differences while the time derivative is kept continuous. We therefore have

$$w_t = \frac{a\left(w_{j+1}^n - w_{j-1}^n\right)}{2\Delta x} + \frac{\varepsilon}{(\Delta x)^2}\left(w_{j+1}^n - 2w_j^n + w_{j-1}^n\right). \tag{16}$$

We Fourier transform (16) to get

$$\hat{w}_t = \lambda\hat{w} \tag{17}$$

with

$$\lambda(\xi) = -\frac{2\varepsilon}{(\Delta x)^2}(1 - \cos\xi) + \frac{ia}{\Delta x}\sin\xi \qquad 0 \le \xi \le 2\pi. \tag{18}$$

A Runge-Kutta scheme for (16) or (17) is stable whenever $z(\xi) = \lambda(\xi)\Delta t_e$ lies within the stability domain that depends on $\alpha_1,\cdots,\alpha_N$ for all $0 \le \xi \le 2\pi$.

We consider the stability domain for the four-step scheme with $\alpha_1 = 1/4$, $\alpha_2 = 1/3$, $\alpha_3 = 1/2$, $\alpha_4 = 1$. This scheme has a stability condition along the imaginary axis of $\max_{\xi}|z| \leq 2\sqrt{2}$, i.e., for a hyperbolic problem $(\varepsilon = 0)$ $\frac{a\Delta t_e}{\Delta x} \leq 2\sqrt{2}$. Along the negative real axis the stability condition is $|z| \lesssim 2.8$ and for a parabolic problem $(a = 0)$ $\frac{2\varepsilon\Delta t_e}{(\Delta x)^2} \lesssim 2.8$. Hence for this case we would choose κ in (15) as $\kappa = 1.4$. We define the cell Reynolds number as

$$R_h = \frac{a\Delta x}{\varepsilon}. \tag{19}$$

The previous analysis shows that the Runge-Kutta scheme is stable for $R_h = 0$ and $R_h = \infty$. We do not have any proof that the scheme is stable for all R_h.

III. NAVIER-STOKES EQUATIONS

We now discuss the implementation of these ideas to the two-dimensional, compressible, Navier-Stokes equations. The extension to three dimensions is straightforward. We first consider the conservation form in Cartesian coordinates. We express the equations in the following form

$$\begin{aligned}
\rho_t &= H_1 \\
(\rho u)_t &= H_2 + (\lambda + 2\mu)\frac{\partial^2 u}{\partial x^2} + (\lambda + \mu)\frac{\partial^2 v}{\partial x \partial y} + \mu\frac{\partial^2 u}{\partial y^2} \\
(\rho v)_t &= H_3 + \mu\frac{\partial^2 v}{\partial x^2} + (\lambda + \mu)\frac{\partial^2 u}{\partial x \partial y} + (\lambda + 2\mu)\frac{\partial^2 v}{\partial y^2}
\end{aligned} \tag{20}$$

$$(\rho E)_t = H_4 + \frac{\gamma\mu}{Pr}\left(\frac{\partial^2 e}{\partial x^2} + \frac{\partial^2 e}{\partial y^2}\right)$$

$$+ (\lambda + 2\mu)u \frac{\partial^2 u}{\partial x^2} + \mu v \frac{\partial^2 v}{\partial x^2}$$

$$+ (\lambda + \mu)\left[v \frac{\partial^2 u}{\partial x \partial y} + u \frac{\partial^2 v}{\partial x \partial y}\right]$$

$$+ \mu u \frac{\partial^2 u}{\partial y^2} + (\lambda + 2\mu)v \frac{\partial^2 v}{\partial y^2}$$

where

$$e = E - \frac{(\rho u)^2 + (\rho v)^2}{2\rho} ,$$

and H_j denote first derivative terms (including the artificial viscosity and also the viscous dissipation function). The coefficients of viscosity (μ and λ), γ the specific heat ratio, and the Prandtl number Pr are all assumed (for the analysis) to be locally constant.

In deriving our results we shall ignore all cross derivatives (see, e.g., [1], [2]). Based on our previous analysis we add the following terms to the standard Runge-Kutta scheme.

$$\Delta\rho = K_1$$

$$\Delta(\rho u) = K_2 - 2\left[\frac{\lambda + 2\mu}{(\Delta x)^2} + \frac{\mu}{(\Delta y)^2}\right] \frac{\Delta(\rho u)}{\rho} \alpha\Delta t$$

$$\Delta(\rho v) = K_3 - 2\left[\frac{\mu}{(\Delta x)^2} + \frac{\lambda + 2\mu}{(\Delta y)^2}\right] \frac{\Delta(\rho v)}{\rho} \alpha\Delta t \tag{21}$$

$$\Delta(\rho E) = K_4 - \frac{2u}{\rho}\left[-\frac{\gamma\mu}{2Pr}\left(\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2}\right) + \frac{(\lambda + 2\mu)}{(\Delta x)^2} + \frac{\mu}{(\Delta y)^2}\right]\Delta(\rho u)\alpha\Delta t$$

$$- \frac{2v}{\rho} \left[- \frac{\gamma\mu}{2Pr} \left(\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2} \right) + \frac{\mu}{(\Delta x)^2} + \frac{(\lambda + 2\mu)}{(\Delta y)^2} \right] \Delta(\rho v) \cdot \alpha \Delta t$$

$$- \frac{2\gamma\mu}{\rho Pr} \left(\frac{1}{(\Delta x)^2} + \frac{1}{(\Delta y)^2} \right) \Delta(\rho E) \cdot \alpha \Delta t$$

where $\Delta w = w^{n+1} - w^n$ and K_j denote the usual space derivative terms. For simplicity we have chosen $\kappa = 1$, and α denotes the constant in the Runge-Kutta scheme (28). Thus the density equation is unchanged. The second and third equations can be solved directly for $\Delta(\rho u)$, $\Delta(\rho v)$. Once $\Delta(\rho u)$, $\Delta(\rho v)$ are known the last equation can be solved for $\Delta(\rho E)$. As before these corrections imply an effective time step which automatically accounts for the viscous time step. In this case the effective time step differs for each equation.

We finally consider the Navier-Stokes equation in body fitted coordinates. This can be done either in a finite volume scheme or by using transformations. The result is the same in either case [9], and so we shall use a transformation for ease of presentation. Let $\xi = \xi(x,y)$, $\eta = \eta(x,y)$ be the body fitted coordinates. We choose the coordinate scaling so that $\Delta\xi = \Delta\eta = 1$. The Navier-Stokes equations (20) now become

$$\rho_t = \overline{H}_1$$

$$(\rho u)_t = \overline{H}_2 + \left[(\lambda + 2\mu)\xi_x^2 + \mu\xi_y^2 \right] \frac{\partial^2 u}{\partial \xi^2} + \left[(\lambda + 2\mu)\eta_x^2 + \mu\eta_y^2 \right) \frac{\partial^2 u}{\partial \eta^2}$$

$$+ (\lambda + \mu)\xi_x \, \xi_y \frac{\partial^2 v}{\partial \xi^2} + (\lambda + \mu)\eta_x \, \eta_y \frac{\partial^2 v}{\partial \eta^2} + \text{crossterms}$$

$$(\rho v)_t = \overline{H}_3 + \left[\mu\xi_x^2 + (\lambda + 2\mu)\xi_y^2 \right] \frac{\partial^2 v}{\partial \xi^2} + \left[\mu\eta_x^2 + (\lambda + 2\mu)\eta_y^2 \right] \frac{\partial^2 v}{\partial \eta^2}$$

$$+ (\lambda + \mu)\xi_x \, \xi_y \frac{\partial^2 u}{\partial \xi^2} + (\lambda + \mu)\eta_x \, \eta_y \frac{\partial^2 u}{\partial \eta^2} + \text{crossterms} \tag{22}$$

$$(\rho E)_t = \overline{H}_4 + \frac{\gamma\mu}{Pr}\left[\left(\xi_x^2 + \xi_y^2\right)\frac{\partial^2 e}{\partial \xi^2} + \left(\eta_x^2 + \eta_y^2\right)\frac{\partial^2 e}{\partial \eta^2}\right]$$

$$+ \left[(\lambda + 2\mu)u\xi_x^2 + (\lambda + \mu)v\xi_x \, \xi_y + \mu u\xi_y^2\right]\frac{\partial^2 u}{\partial \xi^2}$$

$$+ \left[(\lambda + 2\mu)u\eta_x^2 + (\lambda + \mu)v\eta_x \, \eta_y + \mu u\eta_y^2\right]\frac{\partial^2 u}{\partial \eta^2}$$

$$+ \left[\mu v\xi_x^2 + (\lambda + \mu)u\xi_x \, \xi_y + (\lambda + 2\mu)v\xi_y^2\right]\frac{\partial^2 v}{\partial \xi^2}$$

$$+ \left[\mu v\eta_x^2 + (\lambda + \mu)u\eta_x \, \eta_y + (\lambda + 2\mu)v\eta_y^2\right]\frac{\partial^2 v}{\partial \eta^2} + \text{crossterms}$$

where $\overline{H}_j$ are first derivative terms and we have ignored all second cross derivative terms. As before this generates an appropriate correction term to the Runge-Kutta scheme. Equation (21) is now replaced by

$$\Delta\rho = \overline{K}_1$$

$$\Delta(\rho u) = \overline{K}_2 - 2\left[(\lambda + 2\mu)\xi_x^2 + \mu\xi_y^2 + (\lambda + 2\mu)\eta_x^2 + \mu\eta_y^2\right]\frac{\Delta(\rho u)}{\rho}\,\alpha\Delta t$$

$$- 2(\lambda + \mu)(\xi_x \, \xi_y + \eta_x \, \eta_y)\frac{\Delta(\rho v)}{\rho}\,\alpha\Delta t$$

$$\Delta(\rho v) = \overline{K}_3 - 2(\lambda + \mu)(\xi_x \, \xi_y + \eta_x \, \eta_y)\frac{\Delta(\rho u)}{\rho}\,\alpha\Delta t$$

$$- 2\left[\mu\xi_x^2 + (\lambda + 2\mu)\xi_y^2 + \mu\eta_x^2 + (\lambda + 2\mu)\eta_y^2\right]\frac{\Delta(\rho v)}{\rho}\,\alpha\Delta t \tag{23}$$

$$\Delta(\rho E) = \overline{K}_4 - \frac{2\gamma\mu}{\rho Pr}\left(\xi_x^2 + \xi_y^2 + \eta_x^2 + \eta_y^2\right)\Delta(\rho E)\cdot\alpha\Delta t$$

$$- 2\Big[-\frac{u}{2}\frac{\gamma\mu}{Pr}\left(\xi_x^2 + \xi_y^2 + \eta_x^2 + \eta_y^2\right) + (\lambda + 2\mu)u\left(\xi_x^2 + \eta_x^2\right)$$

$$+ (\lambda + \mu)v(\xi_x\,\xi_y + \eta_x\,\eta_y) + \mu u\left(\xi_y^2 + \eta_y^2\right)\Big]\,\frac{\Delta(\rho u)}{\rho}\,\alpha\Delta t$$

$$- 2\Big[-\frac{v}{2}\frac{\gamma\mu}{Pr}\left(\xi_x^2 + \xi_y^2 + \eta_x^2 + \eta_y^2\right) + \mu v\left(\xi_x^2 + \eta_x^2\right)$$

$$+ (\lambda + \mu)u(\xi_x\,\xi_y + \eta_x\,\eta_y) + (\lambda + 2\mu)v\left(\xi_y^2 + \eta_y^2\right)\Big]\,\frac{\Delta(\rho v)}{\rho}\cdot\alpha\Delta t$$

where $\overline{K}_j$ represents the standard finite difference terms.

As before the density equation is unchanged by the viscous correction. Now, however, the two momentum equations are coupled together, unless the coordinate system is orthogonal. As we have two equations for $\Delta(\rho u)$ and $\Delta(\rho v)$, and we can easily solve these. To simplify the notation we define

$$z_1 = 1 + \frac{2\alpha\Delta t}{\rho}\left[(\lambda + 2\mu)\left(\xi_x^2 + \eta_x^2\right) + \mu\left(\xi_y^2 + \eta_y^2\right)\right]$$

$$z_2 = \frac{2\alpha\Delta t}{\rho}(\lambda + \mu)(\xi_x\,\xi_y + \eta_x\,\eta_y) \tag{24}$$

$$z_4 = 1 + \frac{2\alpha\Delta t}{\rho}\left[\mu\left(\xi_x^2 + \eta_x^2\right) + (\lambda + 2\mu)\left(\xi_y^2 + \eta_y^2\right)\right]$$

and

$$D = (1 + z_1)(1 + z_4) - z_2^2.$$

Then

$$\Delta\rho = \overline{K}_1$$

$$\Delta(\rho u) = \frac{\overline{K}_2\ z_4 - \overline{K}_3\ z_2}{D} \tag{25}$$

$$\Delta(\rho v) = \frac{\overline{K}_3\ z_1 - \overline{K}_2\ z_2}{D}\ .$$

As before given $\Delta(\rho u)$ and $\Delta(\rho v)$ we can solve for $\Delta(\rho E)$ directly from the energy equation in (23). We also note that if one uses the thin layer approximation (dropping all second ξ derivatives and cross derivatives in (22)) then these terms simplify slightly. In this case $\Delta\rho$, $\Delta\rho u$, $\Delta\rho v$ are still given by (25) with

$$z_1 = 1 + \frac{2\alpha\Delta t}{\rho}\left[(\lambda + 2\mu)\eta_x^2 + \mu\eta_y^2\right]$$

$$z_2 = \frac{2\alpha\Delta t}{\rho}(\lambda + \mu)\eta_x\ \eta_y$$

$$z_4 = 1 + \frac{2\alpha\Delta t}{\rho}\left[\mu\eta_x^2 + (\lambda + 2\mu)\eta_y^2\right] \tag{26}$$

$$\eta_x = -\frac{y_\xi}{J} \qquad \eta_y = +\frac{x_\xi}{J}$$

$$J = x_\xi\ y_\eta - x_\eta\ y_\xi$$

and

$$\left[1 + \frac{2\gamma\mu\alpha\Delta t}{\rho Pr}\left(\eta_x^2 + \eta_y^2\right)\right]\Delta(\rho E) = \overline{K}_4$$

$$- 2\left[-\frac{u}{2}\left(\frac{\gamma\mu}{Pr}\right)\left(\eta_x^2 + \eta_y^2\right) + (\lambda + 2\mu)u\eta_x^2 + (\lambda + \mu)v\eta_x\ \eta_y + \mu u\eta_y^2\right]\frac{\Delta(\rho u)}{\rho}\cdot\alpha\Delta t$$

$$- 2\left[-\frac{v}{2}\left(\frac{\gamma\mu}{Pr}\right)\left(\eta_x^2 + \eta_y^2\right) + \mu v\eta_x^2 + (\lambda + \mu)u\eta_x\ \eta_y + (\lambda + 2\mu)u\eta_y^2\right]\frac{\Delta(\rho v)}{\rho}\cdot\alpha\Delta t.$$

IV. RESIDUAL SMOOTHING

As an alternative method of reducing the effect of the parabolic terms on the stability of the scheme we consider residual smoothing. With this technique one post-processes an explicit method with an implicit method. In practice one post-processes each equation separately and each direction separately so that only scalar tridiagonal matrices need be inverted. When using a multistage Runge-Kutta method, one can apply the residual smoothing after each stage, or at the end of the entire process, or any intermediate permutation.

In [10] it is shown that one can construct such a scheme for a hyperbolic equation so that the total method is unconditionally stable. It is further shown in [10] that it is not efficient to use a very large Δt even ignoring splitting errors. An optimal Δt is about two to three times larger than the explicit time step. We now consider the process for a parabolic problem in order to see the effect of viscous terms.

We, therefore, consider the heat equation

$$u_t = bu_{xx}. \tag{27}$$

We solve this equation by a k-stage Runge-Kutta scheme

$$\begin{aligned} u^{(1)} &= u^n + \alpha_1 \, \Delta t Q u \\ &\vdots \\ u^{(\ell+1)} &= u^n + \alpha_{\ell+1} \, \Delta t Q u^{(\ell)} \\ &\vdots \\ u^{n+1} &= u^{(k)} \end{aligned} \tag{28}$$

where $\alpha_1,\cdots,\alpha_k$ are given coefficients with $\alpha_k = 1$. Q is a difference approximation to u_{xx}. The amplification factor corresponding to (28) is

$$G = 1 + \beta_1 \Delta t\hat{Q} + \beta_2(\Delta t)^2 \hat{Q}^2 + \cdots + \beta_k(\Delta t)^k \hat{Q}^k \tag{29}$$

where $\beta_1 = 1$ and $\beta_\ell = \beta_{\ell-1}\, \alpha_{k-\ell+1}$, $\ell = 2,\cdots,k$. $\hat{Q}$ is the Fourier transform of Q. Hence, for second-order central differencing

$$\hat{Q} = -\frac{4b \sin^2(\theta/2)}{(\Delta x)^2} . \tag{30}$$

Residual smoothing consists of updating a stage (ℓ) by

$$(1 - \sigma D_2)\Delta u^{(\ell)} = u^{(\ell)} - u^n \tag{31}$$

where D_2 is again a second-order central difference approximation to u_{xx}, i.e., $D_2 \to (1,-2,1)$. We now consider two possibilities. In the first we apply (31) only after the final stage. Then the new amplification factor is

$$G_1(\theta) = 1 + \frac{\beta_1 \Delta t\hat{Q} + \beta_2(\Delta t)^2 \hat{Q}^2 + \cdots + \beta_k(\Delta t)^k \hat{Q}^k}{1 + 4\sigma \sin^2(\theta/2)} . \tag{32}$$

The second case we consider is applying (31) after every stage. The resultant amplification factor is

$$G_2(\theta) = 1 + \beta_1 \Delta t\hat{R} + \beta_2(\Delta t)^2 \hat{R}^2 + \cdots + \beta_k(\Delta t)^k \hat{R}^k \tag{33}$$

with

$$\hat{R} = \frac{\hat{Q}}{1 + 4\sigma \sin^2(\theta/2)} . \tag{34}$$

We now investigate the possibility that either of these schemes is unconditionally stable. To investigate this we need only consider Δt sufficiently large. We thus consider $\Delta t \to \infty$ with $\sigma \to \infty$. Then (32) becomes

$$G_1(\theta) \longrightarrow 1 + \frac{(-1)^k \beta_k [\frac{4b\Delta t}{(\Delta x)^2} \sin^2(\frac{\theta}{2})]^k}{1 + 4\sigma \sin^2(\theta/2)} . \tag{35}$$

We thus see that for k even, $G_1(\theta) > 1$ and so (28) - (31) cannot be stable for Δt large. For $k = 1$ the scheme is identical with backward Euler for a scalar one-dimensional equation and, hence, unconditionally stable. For the second case we see that (33) has the same form as a standard Runge-Kutta method with $\hat{Q}$ replaced by $\hat{R}$, (34). Hence, it follows that the scheme is stable whenever $\Delta t\hat{R}$ is within the stability region of the scheme. As $\Delta t \to \infty$, so does σ and so there is a cancellation between the numerator and denominator; thus, $\Delta t\hat{R}$ remains bounded as Δt increases. We thus conclude that applying the residual smoothing after each stage can make the scheme unconditionally stable even for a Runge-Kutta method with an even number of stages.

We also see from the above argument that as Δt increases so must σ. In [9], [10] we show that for a hyperbolic equation

$$u_t + au_x = 0$$

that σ is proportional to $(a\Delta t/\Delta x)^2$. For the parabolic problem (27) it follows from (35) that σ should be proportional to $b\Delta t/(\Delta x)^2$. For the combined convection-diffusion equation σ will be related to the sum of two such contributions.

It follows from (33), (34) that if we apply residual smoothing after every stage then the stability polynomial has the same form as the original polynomial (29). The only difference is that $\hat{Q}$ is now replaced by $\hat{R}$. From (34) it follows that the ratio of $\hat{Q}$ to $\hat{R}$ is real. Hence, if $\hat{Q}$ is any complex number then $\hat{R}$ lies along the same ray in the complex plane but with a different amplitude. We therefore have shown that if the original scheme was unstable for a given direction then residual smoothing cannot stabilize the scheme. Furthermore, if the original scheme was conditionally stable then by choosing $\sigma = \sigma(\Delta t)$ sufficiently large we can make the scheme unconditionally stable. We have thus shown

Theorem: Let $\hat{Q}$ be the amplification factor for any approximation to the convection-diffusion equation and let (29) be the stability polynomial for a k stage Runge-Kutta scheme. We now apply residual smoothing, (31), after every stage of the scheme. If the original scheme was unconditionally unstable then the new scheme is still unconditionally unstable. If the original scheme was conditionally stable then the scheme with residual smoothing can be made unconditionally stable by choosing $\sigma(\Delta t)$ sufficiently large.

Hence, if the smoothing is applied at the end when solving a parabolic equation, then the scheme can be unconditionally stable only when using a multistage scheme with an odd number of stages. When the smoothing is done after each stage, the scheme can be stabilized for σ large. For a system with a hyperbolic portion and a small parabolic contribution, e.g., high Reynolds number Navier-Stokes, the residual smoothing is most effective with a

time step about twice that of the explicit convective portion. Hence, the question of unconditional stability is somewhat academic. In practice [8] the Runge-Kutta scheme for the Navier-Stokes equations is used with four stages and with the residual smoothing applied after each stage.

V. RESULTS

In this section we present some results for viscous flow obtained using the analysis of Sections II and III. We used a Runge-Kutta code to solve the Navier-Stokes equations for two flows about an airfoil section. The details of this code are discussed in [5], [9], [10], [11]. In these cases we considered only the thin-layer form of the Navier-Stokes equations.

For the first case we computed laminar flow over an NACA 0012 airfoil with a free-stream Mach number M_∞ of 0.5 and a Reynolds number Re_∞ of 5×10^3. The angle of attack (α) of the airfoil was zero degrees. Half-plane calculations were performed using a C-type grid consisting of 64 cells in the streamwise direction and 64 cells in the normal-like direction. The grid spacing at the airfoil surface was about 6×10^{-4} chords. The mesh spacing in the streamwise direction over the central part of the airfoil was $\Delta X = 0.05$ chords. Results for this case are shown in Figures 1a - 1c. As indicated in Figure 1b, the flow separates at $X = 0.817$ chords. The size of the recirculation zone is displayed in Figure 1c. The results are all independent of the time step procedure used to reach the steady state.

In Figure 1d convergence histories for this case for two calculations are shown. The residual displayed in this graph is the root mean square of the residual of the continuity equation. The calculations were started

impulsively by inserting the airfoil into a uniform flow and immediately enforcing the appropriate boundary conditions. Local time stepping and enthalpy damping (see [9]) were employed in each computation; no residual smoothing was used. For history A the Runge-Kutta scheme with the time step (Δt) limitation determined by convection was used; this required choosing a CFL = 1.0. For curve B a larger Courant-Friedrichs-Lewy (CFL) number was used by accounting for the diffusion limit on Δt with the pseudo-time algorithm. This allowed choosing CFL = 2.5 based on an inviscid criterion. There is additional work with the pseudo-time scheme. Nevertheless, the computational time required to reach a satisfactory level of convergence was reduced by a factor of 1.7.

In the second case we solved for turbulent flow over an NACA 0012 airfoil with $M_\infty = 0.5$, $Re_\infty = 2.89 \times 10^6$, and $\alpha = 0$ degrees. A 60×50 half-plane grid was used in the computations. The grid spacing at the surface was about 8.5×10^{-5} chords. The chordwise spacing at the midsection of the airfoil was about $\Delta X = 0.036$ chords. Numerical results for this case are presented in Figures 2a and 2b.

Figure 2c shows two convergence histories for this turbulent flow case. As in the laminar flow problem, the histories were obtained by computing without and with the effects on Δt due to diffusion. The pseudo-time algorithm was about 1.4 times faster in reaching steady state. This is close to the factor expected, since we were able to increase the CFL from 1.5 to 2.7, a factor of 1.8. We do not achieve this speedup of 1.8 since there is some reduction of the effective time step due to the diffusion terms.

VI. CONCLUSIONS

The use of the Crocco scheme for a scalar convection-diffusion equation introduces a scaling of the time step. This reduces the effective time step so that the viscous stability limit is automatically satisfied. As such the scheme cannot introduce any fundamental acceleration in reaching the steady state. The advantage of the scheme is that we do not need to explicitly account for the viscous time step restriction; it is done automatically. This can be done efficiently using Runge-Kutta type schemes. In addition, for variable coefficients or nonuniform meshes this introduces an effective local time step.

Using this scheme for a system of equations, e.g., Navier-Stokes, has the additional benefit that a different scaling is chosen for each equation. Thus each equation has its own appropriate (viscous) time step. This is equivalent to using a diagonal preconditioning [10] to accelerate the equations to a steady state. Computations demonstrate that we can gain a factor of between 1.5 and 2 with little programming effort.

We further show that if one uses residual smoothing to increase the time step then one must also account for the viscous terms. When the smoothing is applied after the completion of a Runge-Kutta cycle then unconditional stability is possible only if an odd number of stages is used. Applying the smoothing after each stage allows for unconditional stability for all multistage schemes provided σ is chosen sufficiently large.

REFERENCES

[1] S. Abarbanel and D. Gottlieb, "Optimal time splitting for two- and three-dimensional Navier-Stokes equations with mixed derivatives." J. Comput. Phys. 41, 1-33 (1981).

[2] J. S. Allen and S. I. Cheng, "Numerical solutions of the compressible Navier-Stokes equations for the laminar near wake in supersonic flow." Phys. Fluids 13, 37-52 (1970).

[3] Yu I. Brailovskaya, "A difference scheme for numerical solution of the two-dimensional nonstationary Navier-Stokes equations for a compressible gas." Soviet Phys. Dokl. 10, 107-110 (1965).

[4] L. Crocco, "A suggestion for the numerical solution of the steady Navier-Stokes equations." AIAA J. 3, 1824-1832 (1965).

[5] A. Jameson, W. Schmidt, and E. Turkel, "Numerical solutions of the Euler equations by finite volume methods using Runge-Kutta time-stepping schemes." AIAA paper 81-1259 (1981).

[6] R. Peyret and H. Viviand, "Computation of viscous compressible flows based on the Navier-Stokes equations." AGARD Report 212 (1975).

[7] T. H. Pulliam and J. L. Steger, "Recent Improvements in efficiency, accuracy, and convergence for implicit approximate factorization algorithms," AIAA paper 85-0360 (1985).

[8] P. J. Roache, Computational Fluid Dynamics, Hermosa Publishers, 1976.

[9] R. C. Swanson and E. Turkel, "A multistage time-stepping scheme for the Navier-Stokes equations." AIAA paper 85-0035 (1985).

[10] E. Turkel, "Acceleration to a steady state for the Euler equations." Numerical Methods for the Euler Equations of Fluid Dynamics, pp. 281-311, SIAM, Philadelphia, 1985.

[11] E. Turkel, "Algorithms for the Euler and Navier-Stokes equations on supercomputers." Progress and Supercomputing in Computational Fluid Dynamics (Edited by E. M. Murman and S. S. Abarbanel), 155-172, Birkhauser Publishing Co., Boston, 1985.

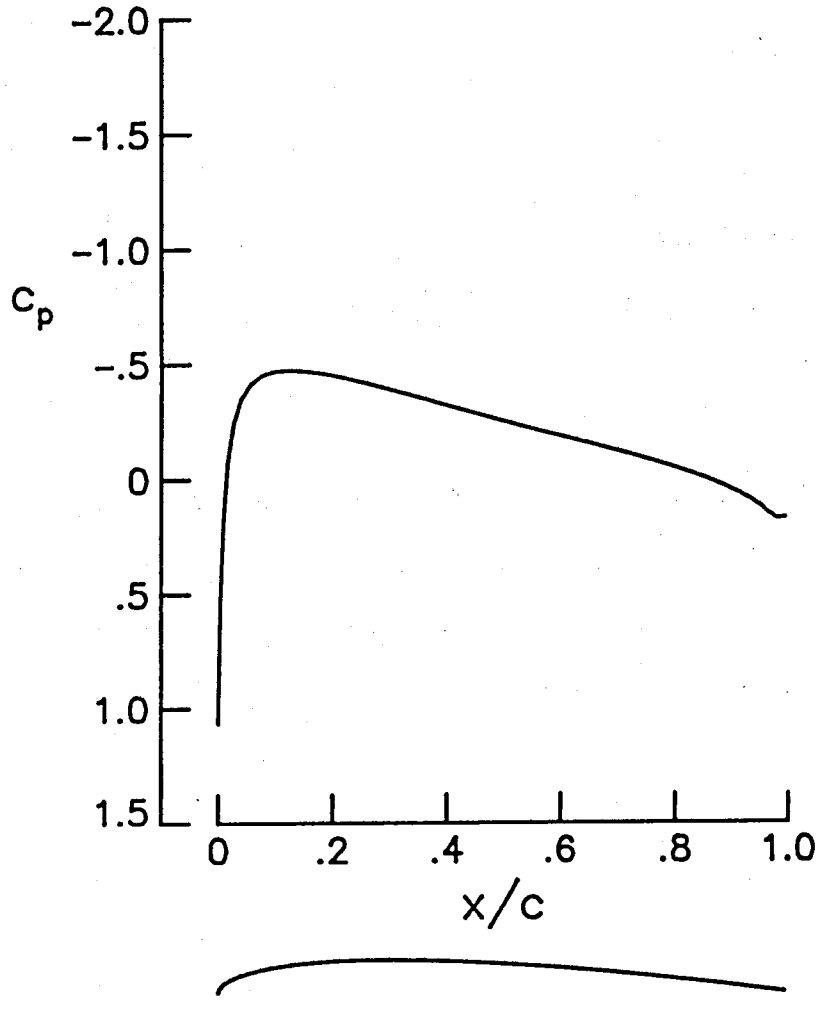

Figure 1a

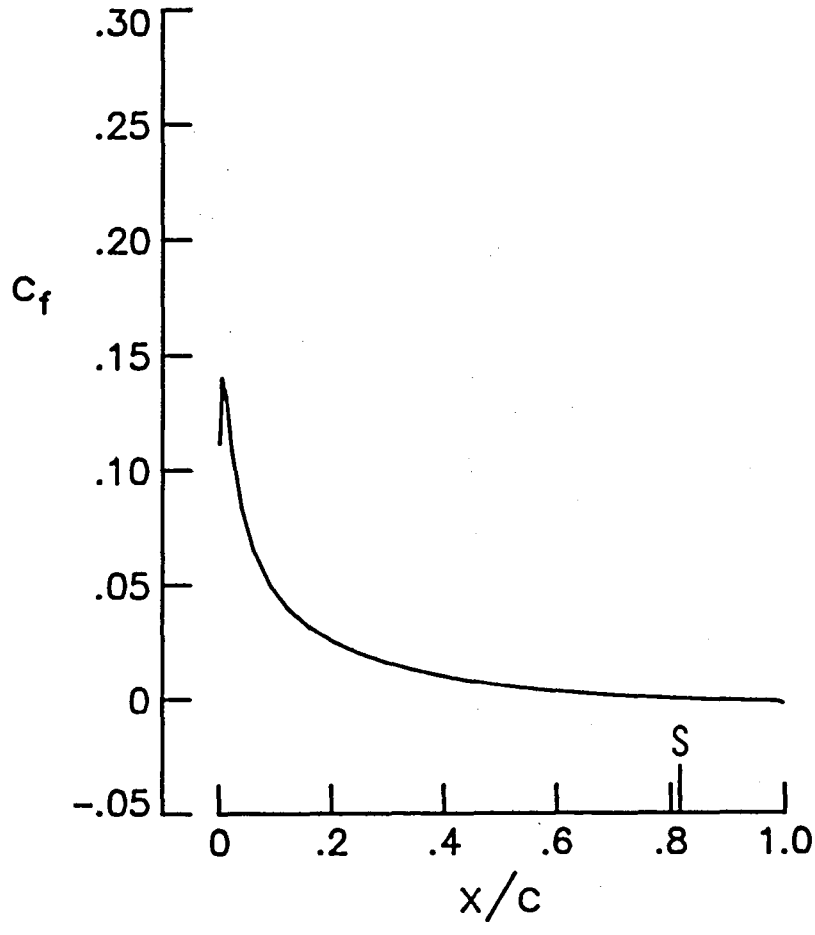

Figure 1b

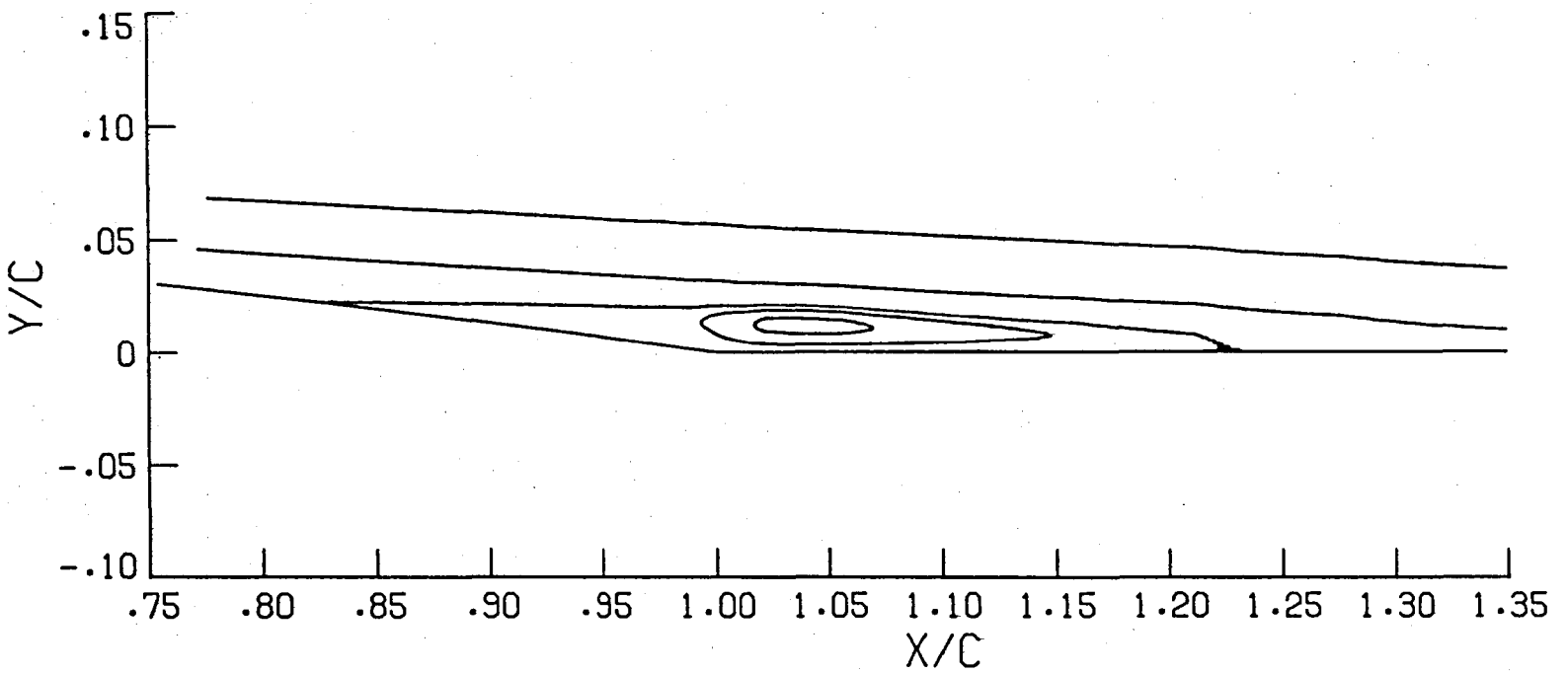

Figure 1c

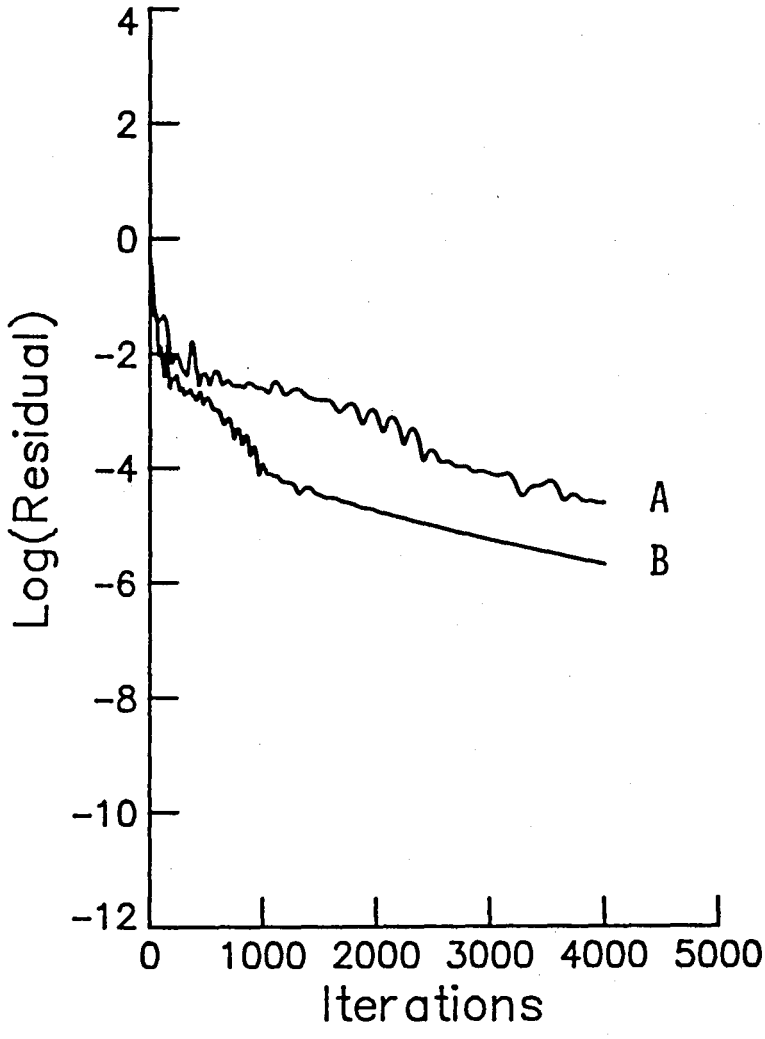
4
2
0
-2
-4
-6
-8
-10
-12
Log(Residual)
0
1000
2000
3000
4000
5000
Iterations
A
B

Figure 1d

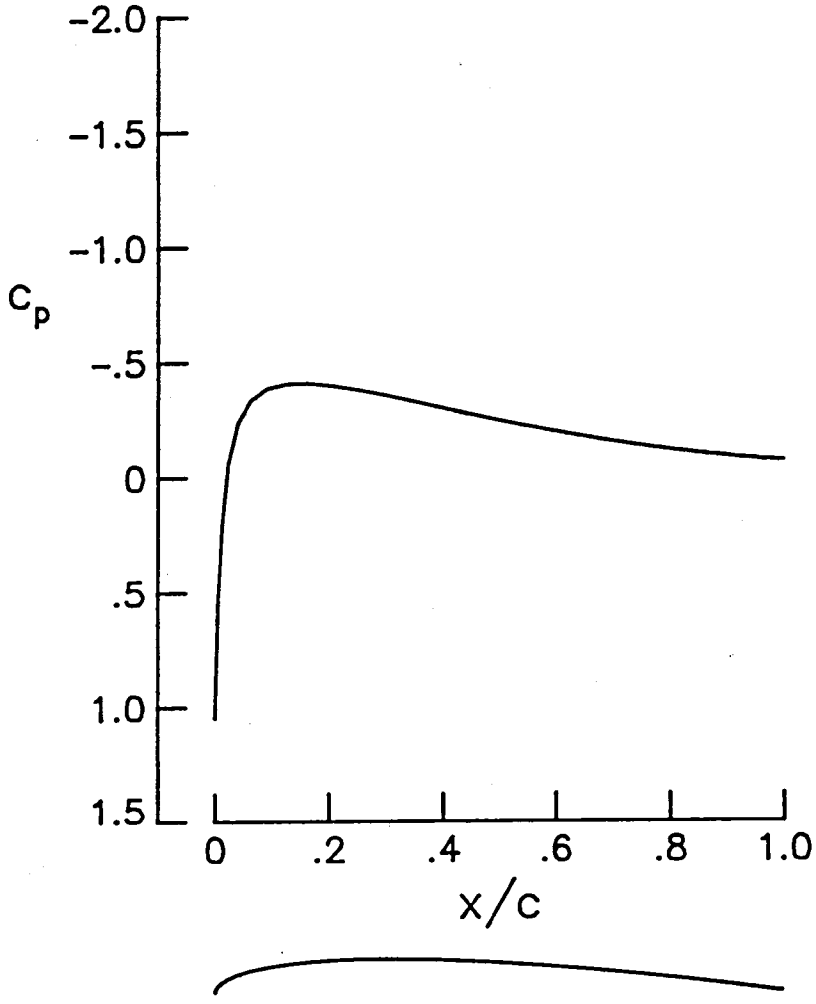

Figure 2a

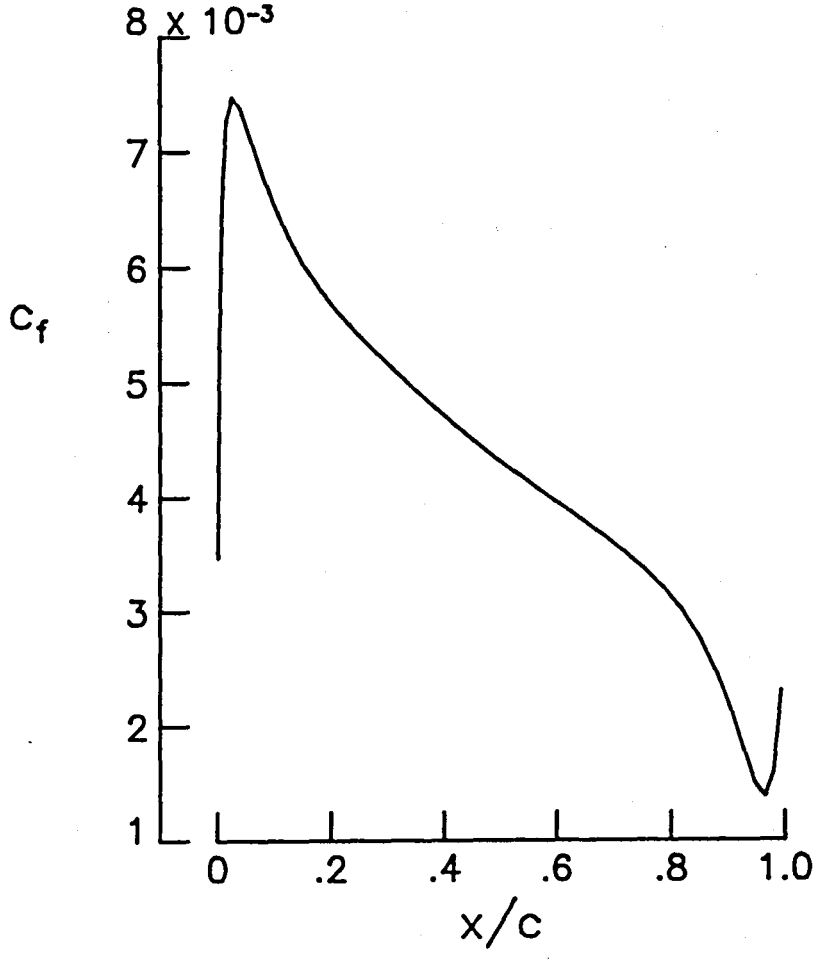

Figure 2b

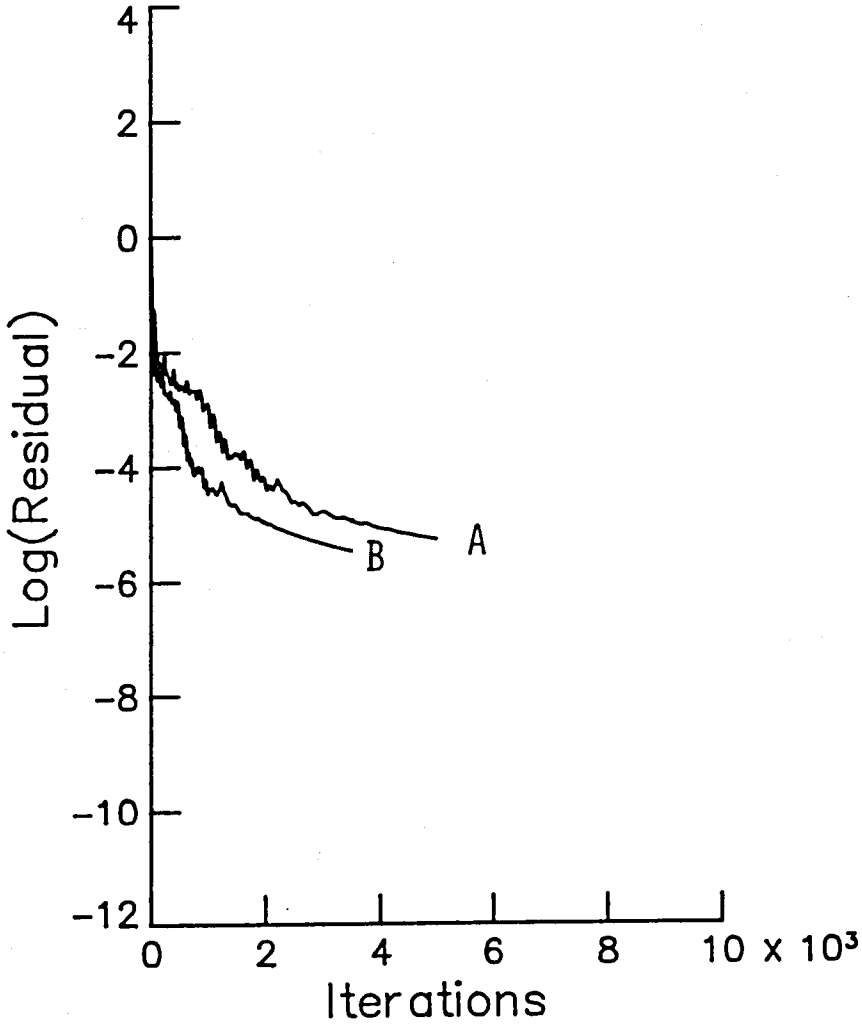

Figure 2c

Figure Captions

Figure 1a. Surface pressure distribution for laminar flow over an NACA 0012 airfoil ($M_\infty = 0.5$), $Re_\infty = 5 \times 10^3$, $\alpha = 0$ degrees).

Figure 1b. Skin-friction (based on free-stream conditions) distribution for laminar flow over an NACA 0012 airfoil ($M_\infty = 0.5$, $Re_\infty = 5 \times 10^3$, $\alpha = 0$ degrees).

Figure 1c. Streamlines for upper surface at the trailing edge ($M_\infty = 0.5$, $Re_\infty = 5 \times 10^3$, $\alpha = 0$ degrees).

Figure 1d. Convergence histories for laminar airfoil flow calculations.

A — Runge-Kutta scheme without pseudo-time algorithm (CFL number of 1.0).

B — Runge-Kutta scheme with pseudo-time algorithm (CFL number of 2.5).

Figure 2a. Surface pressure distribution for turbulent flow over an NACA 0012 airfoil ($M_\infty = 0.5$, $Re_\infty = 2.89 \times 10^6$, $\alpha = 0$ degrees).

Figure 2b. Skin-friction (based on free-stream conditions) distribution for turbulent flow over an NACA 0012 airfoil ($M_\infty = 0.5$, $Re_\infty = 2.89 \times 10^6$, $\alpha = 0$ degrees).

Figure 2c. Convergence histories for turbulent airfoil flow calculations.

A -- Runge-Kutta scheme without pseudo-time algorithm (CFL number of 1.5).

B -- Runge-Kutta scheme with pseudo-time algorithm (CFL number of 2.7).

Standard Bibliographic Page

1. Report No. NASA CR-178127 ICASE Report No. 86-37	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle PSEUDO-TIME ALGORITHMS FOR THE NAVIER-STOKES EQUATIONS		5. Report Date May 1986
		6. Performing Organization Code
7. Author(s) R. C. Swanson, E. Turkel		8. Performing Organization Report No. 86-37
9. Performing Organization Name and Address Institute for Computer Applications in Science and Engineering Mail Stop 132C, NASA Langley Research Center Hampton, VA 23665-5225		10. Work Unit No.
		11. Contract or Grant No. NAS1-17070, NAS1-18107
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		13. Type of Report and Period Covered Contractor Report
		14. Sponsoring Agency Code 505-31-83-01

15. Supplementary Notes

Langley Technical Monitor:
J. C. South

Final Report

Submitted to Applied Numerical Mathematics

16. Abstract

A pseudo-time method is introduced to integrate the compressible Navier-Stokes equations to a steady state. This method is a generalization of a method used by Crocco and also by Allen and Cheng. We show that for a simple heat equation that this is just a renormalization of the time. For a convection-diffusion equation the renormalization is dependent only on the viscous terms. We implement the method for the Navier-Stokes equations using a Runge-Kutta type algorithm. This enables the time step to be chosen based on the inviscid model only. We also discuss the use of residual smoothing when viscous terms are present.

17. Key Words (Suggested by Authors(s)) Navier-Stokes, Runge-Kutta, residual smoothing		18. Distribution Statement 02 - Aerodynamics 64 - Numerical Analysis Unclassified - unlimited	
19. Security Classif.(of this report) Unclassified	20. Security Classif.(of this page) Unclassified	21. No. of Pages 32	22. Price A03

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

NASA-Langley, 1986

End of Document