(NASA-TM-78426) FAST METHODS INCORPORATING DIRECT ELLIPTIC SOLVERS FOR NONLINEAR APPLICATIONS IN FLUID DYNAMICS (NASA) 19 p HC AC2/MF A01 CSCL 20D

N77-30410

Unclas

G3/34 42082

NASA TECHNICAL MEMORANDUM

NASA TM 78,426

NASA TM 78,426

FAST METHODS INCORPORATING DIRECT ELLIPTIC SOLVERS FOR NONLINEAR APPLICATIONS IN FLUID DYNAMICS

E. Dale Martin

Ames Research Center
Moffett Field, Calif. 94035

August 1977

FAST METHODS INCORPORATING DIRECT ELLIPTIC SOLVERS FOR NONLINEAR APPLICATIONS IN FLUID DYNAMICS*

E. Dale Martin
NASA-Ames Research Center, Moffett Field, Calif.

1. INTRODUCTION

This paper describes some developments in fast elliptic solution algorithms and methods for their application in computational fluid dynamics. The description concerns mostly past work, but includes brief remarks on work in progress and some expectations for the future of similar methods.

The primary importance of the fast, direct elliptic solvers lies in the possibilities for their application in complex practical problems in fluid dynamics as well as in other areas of computational physics. The work summarized here is motivated by the need for the efficient solution of practical problems.

The partial differential equations governing fluid-dynamic problems are generally nonlinear. The approach emphasized here for using direct elliptic solvers in the numerical solution of nonlinear problems is referred to as a semidirect method, which is a globally implicit iteration scheme that uses a direct elliptic solver as the driving algorithm of the iteration.

The following topics are discussed in connection with the semidirect method: (a) the equations of fluid dynamics that have been treated or that are expected to be treated, (b) the use of direct solvers for the globally implicit iteration, (c) a Cauchy-Riemann solver and its relation to Poisson

---

*Contribution to the GAMM Workshop on Fast Solution Methods for the Discretized Poisson Equation, held at the Kernforschungszentrum, Karlsruhe, Germany, March 3-4, 1977.

problems, (d) some developments needed for application of the semidirect method to the mixed elliptic-hyperbolic equations of steady, inviscid transonic flow, and (e) the treatment of interior boundary conditions.

2. SEMIDIRECT ITERATIVE METHODS

Consider the equations representing four levels of approximation of fluid flows: (I) Navier-Stokes equations, (II) Euler equations, (III) Potential-flow equations, and (IV) Small-disturbance-potential-flow equations. The Navier-Stokes equations (I) describe a large class of fluid flows. However, when the transport processes including viscosity and heat conduction are negligible, the Euler equations (II) govern the "inviscid flow" adequately. If there are no sources of rotationality, such as strong curved shock waves, in the inviscid flow, the potential-flow approximation (III) is valid. The equations are further simplified (IV) if only small disturbances affect the flow. Fromm /35/ was the first to apply a direct elliptic solver in nonlinear fluid dynamics. He used Buneman's /12/ Poisson solver to treat part of the Navier-Stokes equations for an unsteady, nearly incompressible, laminar flow. Since then, many others have used Poisson solvers in similar ways. Roache /97/ then used a Poisson solver in a different way, as the total driving algorithm in an iterative scheme (i.e., a semidirect method), to solve the Navier-Stokes equations for a simple viscous flow. At about the same time, Widlund /141/ and Concus and Golub /18/ were exploring similar iterative methods for nonseparable (but linear) elliptic equations.

To further describe the semidirect method, let us consider the equations of steady potential flow (III),

$$U_x + V_y = -(1/\rho)(U\rho_x + V\rho_y) \ , \quad U_y - V_x = 0 \ , \qquad (2.1a,b)$$

$$\rho = F(\rho^2U^2+\rho^2V^2) = [1+(1/2)(\gamma-1)M_\infty^2(1-U^2-V^2)]^{1/(\gamma-1)} \qquad (2.1c)$$

where ρ is the dimensionless mass density, U and V are the dimensionless Cartesian velocity components, and γ and M_∞

are constants. An equivalent system for the stream function Ψ defined by $\rho U = \Psi_y$, $\rho V = -\Psi_x$ is

$$\Psi_{xx} + \Psi_{yy} = (1/\rho)(\rho_x \Psi_x + \rho_y \Psi_y) \; , \qquad \rho = F\left(\Psi_x^2 + \Psi_y^2\right) \; . \; (2.2a,b)$$

Another equivalent system can be obtained from eqs. (2.1) for a velocity potential Φ defined by $U = \Phi_x$, $V = \Phi_y$.

If the right side of either eq. (2.1a) or (2.2a) were known, these would be just either nonhomogeneous Cauchy-Riemann equations or a Poisson equation. A simple form of the semidirect iterative method treats the right side as known from a previous iteration and uses a direct elliptic solver for the left side; the iteration is continued until the solution converges. This method has been used (in a preliminary version of /73/*) to solve a finite-difference form of eqs. (2.2) for purely subsonic flow over lifting airfoils. In a similar way, one could use a direct Cauchy-Riemann solver /61,74,71/ to solve instead eqs. (2.1). In these subsonic applications, central differences are used everywhere because the equations are elliptic.

In steady, inviscid transonic flow, on the other hand, the equations are elliptic in the subsonic portions of the flow, but hyperbolic in the supersonic portions. In addition, discontinuities in the variables must be allowed for because the transition from a supersonic region to a subsonic region may occur through a shock wave. Jameson /53/ has solved the so-called "full-potential equation" (III) in terms of the velocity potential for transonic flows using a line-relaxation iterative method (with a direct Poisson solver used to accelerate convergence, but not as the driving algorithm). Work is currently in progress to solve eqs. (2.1) for steady transonic flow over lifting airfoils using a form of the semidirect method.

The semidirect method has been used /73,67-69/ to solve the transonic-small-disturbance equations (IV):

*AIAA Paper 74-11, Jan. 1974.

$$u_x + v_y = M_\infty^2 u_x + \tau(\gamma+1)M_\infty^2 u u_x \ , \qquad u_y - v_x = 0 \ . \tag{2.3a,b}$$

where the perturbation velocities u and v are given by

$$U = 1 + \tau u \ , \qquad V = \tau v \ , \tag{2.4}$$

τ is a small parameter (e.g., airfoil thickness ratio), and M_∞ is the free-stream Mach number. Eqs. (2.3) are locally elliptic or hyperbolic depending on whether $u \lessgtr u^*$, the critical value. If the flow is "subcritical" the semidirect method with central differences is straightforward. In supercritical cases, the hyperbolic regions and shock-wave discontinuities require special treatment; the additional developments needed are described in section 4.

3. DIRECT CAUCHY-RIEMANN SOLVERS

A direct solution algorithm for the elliptic equations

$$u_x + v_y = s(x,y) \ , \qquad u_y - v_x = -\omega(x,y) \ , \tag{3.1a,b}$$

is called a Cauchy-Riemann solver /61,74,71/. (The earliest "direct" Cauchy-Riemann solver appears to have been given by Gates and von Rosenberg /37/, but the more recent "fast" methods do not have some of the apparent limitations of the approach used in /37/.) For the applications, additional terms can be added to the left side of eqs. (3.1) /74,71/.

If $\omega = 0$ everywhere, one can define a function ϕ (e.g., velocity potential) by $u = \phi_x$, $v = \phi_y$, and obtain a Poisson equation for ϕ. Or if $s = 0$ everywhere, one can define a function ψ (e.g., stream function) by $u = \psi_y$, $v = -\psi_x$, and obtain a Poisson equation for ψ. The ϕ-formulation allows point sources (s), but not point vortices. The ψ-formulation allows point vortices (ω), but not point sources. But the u-v formulation allows both s and ω to be nonzero. Furthermore, conditions such as at solid boundaries or across shock waves in the semidirect method are specified most naturally in terms of u and v. For a given problem, the boundary conditions may make the Cauchy-Riemann formulation preferable over a Poisson formulation.

For given Poisson problems, one can solve equivalent Cauchy-Riemann problems and, in the process, obtain the Poisson solution. To illustrate this while formulating the Cauchy-Riemann algorithm /71/, let us consider the discretized problems as follows. Let the indices for u be i and j', the indices for v be i' and j, and let the points where u and v are defined be oriented in a staggered grid as in Fig. 1. The "continuity equation" (3.1a), central differenced about the point i',j', and the "rotationality equation" (3.1b), central differenced about i,j, are

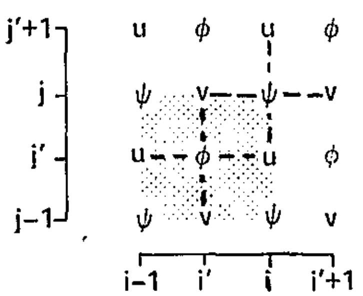

Fig. 1.- Indices for discretization on staggered grid.

$$(u_{i,j'} - u_{i-1,j'})/\Delta x + (v_{i',j} - v_{i',j-1})/\Delta y = s_{i',j'} \ , \quad (3.2a)$$

$$(u_{i,j'+1} - u_{i,j'})/\Delta y - (v_{i'+1,j} - v_{i',j})/\Delta x = -\omega_{i,j} \ . \quad (3.2b)$$

On the computation field in Fig. 2, u is defined at points on the left and right boundaries and v is defined on the top

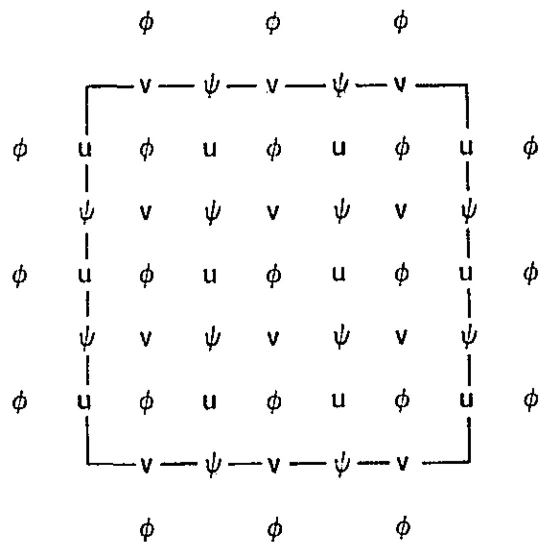

Fig. 2.- Meshes for Poisson-Dirichlet, Poisson-Neumann, and equivalent Cauchy-Riemann problems.

and bottom boundaries. With this mesh, boundary values of u and v are specified respectively on those boundary segments. Equation (3.2a) is to be satisfied at all interior points designated by ϕ on Fig. 2, and eq. (3.2b) is to be satisfied at all interior points designated by ψ. Let us relate this configuration and Cauchy-Riemann (CR) problem to both the Poisson-Dirichlet (P-D), or first boundary-value problem ($n_{bvp} = 1$), and the Poisson-Neumann (P-N), or second boundary-value problem ($n_{bvp} = 2$). If all $s_{i',j'} = 0$, define

$$u_{i,j'} = (\psi_{i,j} - \psi_{i,j-1})/\Delta y \ , \quad v_{i',j} = (\psi_{i-1,j} - \psi_{i,j})/\Delta x \ . \tag{3.3a,b}$$

Equations (3.2) are then equivalent to a discretized Poisson equation for ψ, with values of ψ specified on the boundary ($n_{bvp} = 1$). Thus, a P-D problem can be solved by using an algorithm for eqs. (3.2) on the mesh of Fig. 2, and the values of $\psi_{i,j}$ are then obtained from eqs. (3.3). If, on the other hand, all $\omega_{i,j} = 0$, define

$$\left.\begin{aligned} u_{i,j'} &= (\phi_{i'+1,j'} - \phi_{i',j'})/\Delta x \ , \\ v_{i',j} &= (\phi_{i',j'+1} - \phi_{i',j'})/\Delta y \ . \end{aligned}\right\} \tag{3.4a,b}$$

Eqs. (3.2) are then equivalent to a discretized Poisson equation for φ, with normal differences of ϕ centered on each boundary; and specifying u and v on the boundaries is equivalent to specifying the normal differences of ϕ ($n_{bvp} = 2$). A P-N problem can therefore be solved by using the algorithm for eqs. (3.2) on the mesh of Fig. 2 (with the boundary values of u and v specified in the <u>same</u> configuration as for $n_{bvp} = 1$), and $\phi_{i',j'}$ is obtained from eqs. (3.4).

With an algorithm for solving eqs. (3.2) on Fig. 2, there is no essential difference in the treatments for $n_{bvp} = 1$ and 2 except that in the one case all $s_{i',j'}$ are zero and (3.3) gives the final solution, whereas in the other case all $\omega_{i,j}$ are zero and (3.4) gives the final solution. In previous formulations of Poisson solvers, a different coefficient matrix results for the two problems, and the P-N solution has usually been more time consuming.

When eqs. (3.2) are written for all i,j, an equation is obtained /71/ that reduces to the very regular block-tridiagonal matrix equation

$$\underset{\sim}{T}\underset{\sim}{V} = \underset{\sim}{F} , \qquad (3.5)$$

in which $\underset{\sim}{T}$ is a block-tridiagonal matrix having entries $(-\underset{\sim}{I}, \underset{\sim}{C}, -\underset{\sim}{I})$, $\underset{\sim}{I}$ is a unit matrix, $\underset{\sim}{C}$ is a scalar-tridiagonal matrix having entries $(-\alpha, \beta_i, -\alpha)$, $\alpha = (\Delta y/\Delta x)^2$, $\beta_i = 2 + \alpha$ for the first and last diagonal elements, and otherwise $\beta_i = 2 + 2\alpha$. The column vector $\underset{\sim}{V}$ is composed of all the $v_{i',j}$ and the column vector $\underset{\sim}{F}$ has elements that are simple combinations of the $s_{i',j'}$ and $\omega_{i,j}$.

Note that (a) all the $u_{i,j'}$ have been eliminated from the equation to be solved, (b) eq. (3.5) can be solved by <u>any</u> fast block-tridiagonal-equation solver, (c) after all the $v_{i',j}$ are known, one can obtain the $u_{i,j'}$ quickly from eq. (3.2a) for $j' = n_1$ † and from eq. (3.2b) for all other j'. If only a Poisson solution is wanted, one need not determine all $u_{i,j'}$ but may simply find either the P-D or the P-N solution from eq. (3.3b) or (3.4b).

A FORTRAN computer program (UVSOLV) was written for testing this algorithm and for comparison with other fast Poisson solvers. The "standard problem" for the comparison was the discretized Poisson equation with either Dirichlet ($n_{bvp} = 1$) or Neumann ($n_{bvp} = 2$) conditions. That problem was equivalent to eqs. (3.2) with either (3.3) or (3.4) and

$$s_{i',j'} = (n_{bvp}-1)\lambda_{k,\ell}\phi_{i'j'}, \quad \omega_{i,j} = (n_{bvp}-2)\lambda_{k,\ell}\psi_{i,j} , \qquad (3.6a,b)$$

$$\lambda_{k,\ell} = 2[\cos(ka) + \cos(\ell b) - 2] , \quad a = \pi/m_1 , \quad b = \pi/n_1 , \qquad (3.7)$$

where the functions put into eqs. (3.6) are the exact discretized solutions for $n_{bvp} = 1$ and 2, respectively,

† On the vertical boundaries, $i = 0$ and m_1; on the horizontal, $j = 0$ and n_1.

$$\psi_{i,j} = \sin(ika)\ \sin(j\ell b)\ , \tag{3.8a}$$

$$\phi_{i',j'} = \cos[(i' - 1/2)ka]\ \cos[(j' - 1/2)\ell b]\ . \tag{3.8b}$$

For both cases, the integer index i (or i') ranges from 0 to $M + 1$ and j (or j') ranges from 0 to $N + 1$ where $M = m_1 + n_{bvp} - 2$, $N = n_1 + n_{bvp} - 2$. Zero boundary values for u and v are specified.

The computer program solved eq. (3.5) by recursive cyclic reduction, with the inner loops (scalar tridiagonal solutions) done by Gaussian elimination. The program was run on the Control Data 7600 computer at Ames Research Center, and a slightly modified program was then run on the IBM 370/168 at the Kernforschungszentrum at Karlsruhe, Germany. The test cases used $m_1 = 128$, $n_1 = 32$ for both the P-D and P-N problems. For the 128X32 mesh, the CPU time on the 7600 (FTN 4.5 compiler) for both the P-D and P-N problems was 0.054 (±0.001) sec. On the 370/168 (H compiler), the solution times were 0.18 (±0.01) sec and 0.19 (±0.01) sec. Different values of k and ℓ were used, some of which provided a severe test of the accuracy. Single-precision truncation errors on the IBM machine were excessive for a few cases. Of 18 cases, 6 had maximum errors larger than 10^{-3}, one larger than 10^{-2} (6.64×10^{-2}). (A subsequent modification reduced the largest error to 3×10^{-3}.) However, on the 7600, all errors were less than 10^{-9}, with all but two being less than 10^{-10}. On an IBM 360/67 at Ames Research Center (having the same precision as the 370/168) no accuracy difficulties have been encountered in applications to physical problems. It has also been pointed out that a full double-precision conversion of the program would increase the computing time by only 20% and would eliminate any problems of inaccuracy.

4. SOME ADVANCES MADE FOR THE APPLICATION OF SEMIDIRECT METHODS

We consider further now the use of direct elliptic solvers in semidirect methods and some further developments needed for applications in nonlinear fluid dynamics. Specifically considered are the application to steady, inviscid transonic flow and the treatment of interior conditions.

4.1 Transonic Flow. It is convenient to transform the perturbation velocities u, v (eqs. (2.4)), according to:

$$u = (\bar{a}/\beta)[1 + \bar{u}(\bar{x},\bar{y})] , \quad v = \bar{a}\bar{v}(\bar{x},\bar{y}) , \qquad (4.1a,b)$$

$$y = \bar{y}/\beta , \; x = \bar{x} , \; \beta = (1 - M_\infty^2)^{1/2} , \; \bar{a} = \beta^3/[\tau(\gamma + 1)M_\infty^2]. \qquad (4.1c)$$

Then eqs. (2.3), with $f = -(1/2)\bar{u}^2$, become

$$\bar{u}_{\bar{x}} + \bar{v}_{\bar{y}} = \bar{u}_{\bar{x}} - f_{\bar{x}} , \quad \bar{u}_{\bar{y}} - \bar{v}_{\bar{x}} = 0 . \qquad (4.2a,b)$$

For a nonlifting symmetrical thin airfoil, the chord can be located on one boundary ($\bar{y} = 0$), and the surface tangency condition can be approximated by

$$v = \bar{a}\bar{v} = F'(\bar{x}) \quad \text{on} \quad \bar{y} = 0 , \qquad (4.3)$$

where $\bar{y} = \beta\tau F(\bar{x})$ is the equation of the airfoil surface between the leading and trailing edges, and $F(\bar{x}) = 0$ off the chord. Far from the airfoil, $\bar{u} \to -1$ and $\bar{v} \to 0$ as $\bar{x}^2 + \bar{y}^2 \to \infty$.

If $\bar{u} < 0$ everywhere, eqs. (4.2) are elliptic everywhere. But as M_∞ increases from zero toward unity, both β and $\bar{a}$ become smaller, and then the boundary values of $\bar{v}$ become sufficiently large that $\bar{u}$ attains positive values in some regions, where eqs. (4.2) are then hyperbolic. Shock-wave discontinuities (in $\bar{u}$) can exist where the flow decelerates from supersonic to subsonic velocity. Two related important factors are then the accuracy of a finite-difference representation and the iterative convergence of a scheme to solve the difference equations.

For purely subsonic flow, (4.2) can be central differenced, and the semidirect method converges rapidly. But for transonic flow, the pure central differencing is not appropriate in the embedded hyperbolic region, and the iteration with all central differences does not converge. Murman and Cole (see /76/) introduced type-dependent differencing in their line-relaxation method. In /76/, Murman refined the difference operators to make a fully conservative calculation, which is important for capturing a shock wave in the proper location and for having the proper strength of the jump discontinuity. Such a scheme uses upwind differencing in the hyperbolic regions to account for the domain of dependence; and transition operators for switching between the elliptic and hyperbolic operators provide the needed conservation property. To use type-dependent differencing in the semidirect method, note first that eq. (4.2a) is just $-\bar{u}\bar{u}_{\bar{x}} + \bar{v}_{\bar{y}} = 0$, in which the first term is $f_{\bar{x}}$. Since the sign of $\bar{u}$ in $f_{\bar{x}}$ determines the type for eqs. (4.2), only $f_{\bar{x}}$ needs type-dependent differencing. All other terms can be central differenced. With subscript C denoting central differences and subscript T denoting a type-dependent difference operator, a finite-difference representation of eqs. (4.2) is

$$(\bar{u}_{\bar{x}})_C + (\bar{v}_{\bar{y}})_C = (\bar{u}_{\bar{x}})_C - (f_{\bar{x}})_T , \tag{4.4a}$$

$$(\bar{u}_{\bar{y}})_C - (\bar{v}_{\bar{x}})_C = 0 . \tag{4.4b}$$

Although the use of type-dependent differencing increases the range of conditions (e.g., higher M_∞) for which the semidirect iteration converges, it is not sufficient for flows with large supersonic zones. It is necessary /67,68/ to desymmetrize (with an "upwind bias") the iteration matrix representing the left side of eqs. (4.4). This can be done by adding a term, $(-\bar{\alpha}/\Delta\bar{x})\bar{u}_{i-1,j'}$, to both the left and right sides of eq. (4.4a) for use in the iteration, where the central differences in (4.4a) are centered at point i',j' on Fig. 1. The effects on the stability and convergence of the iteration were explored in /67/, and appropriate values of $\bar{\alpha}$ were determined in /68/.

With simple central differences on a uniform mesh, all the terms other than $(f_{\bar{x}})_T$ in eqs. (4.4) are second-order-accurate. The specification of $(f_{\bar{x}})_T$, however, determines (a) the formal order of accuracy of the solutions, (b) whether the desired conservation property is fulfilled, and (c) the capability of the difference equations to properly describe both the elliptic and hyperbolic portions of the flow field and the transitions (such as through an abrupt "shock wave").

For the finite differences in eqs. (4.4), refer to Fig. 1. Equation (4.4b) is centered at the point i,j and the result is eq. (3.2b) with $\omega_{i,j} = 0$. The shaded area on Fig. 1 is a mesh cell for the difference equation (4.4a), which represents the conservation equation $f_{\bar{x}} + \bar{v}_{\bar{y}} = 0$. The central differences in eq. (4.4a) are the same as in eq. (3.2a). Let

$$(f_{\bar{x}})_T = (\hat{f}_{i,j'} - \hat{f}_{i-1,j'})/\Delta\bar{x} , \quad \hat{f}_{i,j'} = -(1/2)(\hat{u}_{i,j'})^2 . \tag{4.5a,b}$$

For central differencing, $\hat{u}_{i,j'} = \bar{u}_{i,j'}$ for both $\hat{f}_{i,j'}$ and $\hat{f}_{i-1,j'}$. For the general type-dependent differencing derived in /68/, an expression for $\hat{u}_{i,j'}$ has been given in /69/ that produces $(f_{\bar{x}})_T$ in eqs. (4.5) to either first- or second-order accuracy and automatically produces all four types of operators (elliptic or hyperbolic and the transition operators, "parabolic" and "shock-point"). The result (dropping the primes from i' and j' from here on) is

$$\hat{u}_{i,j} = (1 - \sigma_{i,j})\bar{u}_{i,j} + \sigma_{i,j}[\lambda\bar{u}_{i-1,j} + (1 - \lambda)\bar{u}_{i-2,j}] , \tag{4.6}$$

where $\sigma_{i,j}$ is 0 or 1 depending on whether $\tilde{u}_{i,j} < 0$ or > 0, that is, $\sigma_{i,j} = 0.5(1 + \operatorname{sgn} \tilde{u}_{i,j})$, where

$$\tilde{u}_{i,j} = (1/2)(\bar{u}_{i-1,j} + \bar{u}_{i,j}) ; \tag{4.7}$$

and λ is either 1 or (about) 2, corresponding to the formal order of accuracy desired. Note that $\tilde{u}_{i,j}$ is effectively the transformed velocity at the center of the shaded mesh cell but its sign (i.e., whether subsonic or supersonic) is used to determine how the flux $\hat{f}_{i,j}$ is represented at the downstream boundary of that mesh cell in (4.5a). "Artificial viscosity" in the second-order-accurate case is included by using $\lambda = \mu + (1 - \mu)\kappa\Delta\bar{x}$, where μ is 1 or 2 for first- or second-order accuracy and κ is a positive number, $O(1)$. The result from (4.5) is then

$$\begin{aligned}(\Delta\bar{x})(f_{\bar{x}})_T = -(1/2)\{&(1 - \sigma_{i,j})\bar{u}_{i,j} + \sigma_{i,j}[\lambda\bar{u}_{i-1,j} \\ &+ (1 - \lambda)\bar{u}_{i-2,j}]\}^2 + (1/2)\{(1 - \sigma_{i-1,j})\bar{u}_{i-1,j} \\ &+ \sigma_{i-1,j}[\lambda\bar{u}_{i-2,j} + (1 - \lambda)\bar{u}_{i-3,j}]\}^2\end{aligned} \tag{4.8}$$

for use in eqs. (4.4) which become

$$\bar{u}_{i,j} - (1 + \bar{\alpha})\bar{u}_{i-1,j} + (\Delta\bar{x}/\Delta\bar{y})(\bar{v}_{i,j} - \bar{v}_{i,j-1})$$
$$= \bar{u}_{i,j} - (1 + \bar{\alpha})\bar{u}_{i-1,j} - (\Delta\bar{x})(f_{\bar{x}})_T \ , \qquad (4.9a)$$
$$\bar{u}_{i,j+1} - \bar{u}_{i,j} - (\Delta\bar{y}/\Delta\bar{x})(\bar{v}_{i+1,j} - \bar{v}_{i,j}) = 0 \ , \qquad (4.9b)$$

In the semidirect iteration, extended direct Cauchy-Riemann solvers /74,71/ that can include the extra term (with coefficient $\bar{\alpha}$) solve the left side of eqs. (4.9) at all points in the field simultaneously at iteration n, with the right side evaluated at iteration $n - 1$. The four types of difference operators are implicitly included, and shock-wave discontinuities, represented by rapid changes over a few mesh points, are captured automatically.

One can accelerate the iterative convergence by using shifting and scaling transformations /18/ and relaxation factors (see /68/) and by a special technique for using the Aitken extrapolation formula /70/.

Some results have been computed for a simple biconvex airfoil using an earlier inefficient version of this method. The outer conditions, at infinity, were replaced by prescribing the "Prandtl-Glauert" solution on boundaries at 1/2-chord length fore and aft of the airfoil and at five chord lengths above the airfoil. Representative results for the thin-airfoil approximation to the surface pressure coefficient ($C_p \approx -2\tau u$) over the chord are shown in Fig. 3. The mesh for Fig. 3(a) had 20 mesh points on the airfoil chord, and the results compare well with those from a line-relaxation program /77/, which used a refined variable mesh. All values of $-C_p$ above C_p^* are in a supersonic (hyperbolic) region that is terminated downstream by a shock-wave discontinuity. The shock is captured well by the finite-difference scheme. Figure 3(b) illustrates that the second-order-accurate results approximate the pressure distribution well even on a very coarse mesh on which the first-order results are not accurate.

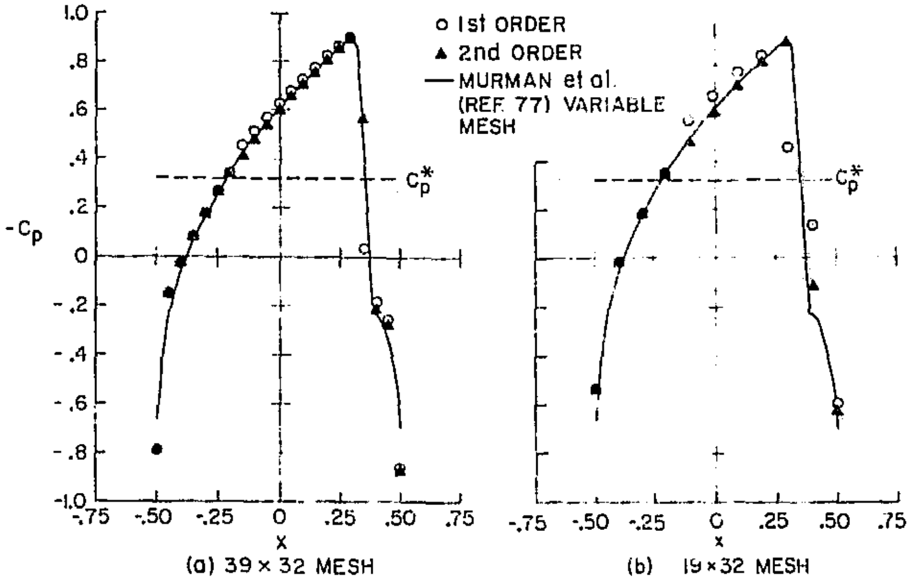

Fig. 3.- Pressure on a thin biconvex airfoil ($M_\infty = 0.85$, $\tau = 0.10$).

The short computing time required is the most significant property of the semidirect method. The measured CPU time per iteration on a Control Data 7600 computer for the 39X32 mesh was 40 ms for an inefficiently coded program of an inefficient version of the algorithm. It is estimated for various reasons, discussed in /68/, that the time can be reduced at least to 20 ms per iteration. The direct solver /74/ used 14 ms on this mesh. The subcritical cases converged in 3 iterations or less, and a slightly supercritical case (using an extrapolation) required only 6 iterations. The first-order-accurate case in Fig. 3(a) required 35 iterations. These methods are continually being improved, and more efficient computations are expected.

4.2 Interior conditions. The above application to transonic flow was for a nonlifting biconvex airfoil. Because of symmetry, the airfoil chord could be located on one boundary of the computation field, and the airfoil surface condition could be transferred to $\bar{y} = 0$. However, the real problem of interest in aerodynamics is the lifting airfoil (or wing). Then there is generally no symmetry condition, and one needs to consider the airfoil in the interior of the flow field.

For treating interior conditions with direct elliptic solvers, the approaches can be grouped into two categories: (a) coordinate transformations, and (b) interior boundary techniques without transformations. When conditions are to be applied on an interior surface such as on a two-dimensional airfoil, a transformation can put the interior surface on one boundary of the computation field of the transformed variables. Such transformations can be either analytical or numerical. These techniques are being vigorously pursued in other solution methods, but have not been extensively employed with direct solvers, although Jameson (e.g., /53/) has used an analytical transofrmation when using a direct Poisson solver to accelerate SOR iterations. On the basis of work currently in progress, it is believed that general coordinate transformations, including those done numerically, hold significant promise for use with direct solvers and semidirect methods.

The other approach is to leave the geometry undistorted and deal with the interior conditions directly. Hockney /48/ adapted the classical capacity-matrix concept to numerical computation. That numerical technique, with all the subsequent refinements and generalizations that have been made for its applications, is referred to as the capacity-matrix technique (CMT). In /66/ procedures were outlined for applying the technique with more general conditions than had been applied previously. The motivating physical problem was flow over a lifting airfoil. Specific procedures were given for applying conditions at points on an airfoil surface not coinciding with mesh points and for including the determination of the circulation (which produces the lift) about the airfoil to satisfy the Kutta condition at the trailing edge. In aerodynamics, that determination is normally done iteratively, but the "generalized" CMT provided the direct solution for the flow including the Kutta condition without iteration. One extra row and one extra column in the capacity matrix account for the Kutta condition and the circulation.

Although the examples used in /66/ were for Laplace's equation, the same technique has been used for calculation of nonlinear compressible flow over a lifting airfoil using a semidirect method for eqs. (2.2) in a preliminary version of /73/, with a condition $\Psi = 0$ on the airfoil surface.

Recently various methods were studied /72/ for treating conditions imposed on an interior slit. One method is the straightforward application of CMT with a Poisson operator. Extensions of this in a semidirect method can use combinations of CMT and shifting of matrix elements from one side of the iteration equation to the other. These techniques are useful for problems containing both a slit (where, e.g., a Neumann condition is applied) and a cut behind the slit on which a discontinuity in the solution is enforced. In /72/ the CMT concept is also extended to use with first-order-elliptic equations, and various methods for applying discontinuous conditions on a slit are also explored. Such treatments can be useful in aerodynamic flows, in which either an airfoil (wing) can be approximated by a slit (sheet) or transformations can be used that represent parts of airplanes (wings, fuselage, etc.) as sheets in a three-dimensional configuration.

5. CONCLUDING REMARKS

The future prospects are encouraging for the use of direct elliptic solvers within semidirect methods for practical problems. On the basis of developments in progress it is anticipated that the full potential-flow equations (III, see section 2) will be treated successfully by semidirect methods for general inviscid transonic flows. Looking further into the future, based on some preliminary developments, it is believed that the most general fluid-dynamic equations (II and I) may eventually be solvable, for practical problems, using generalized forms of the semidirect method.

REFERENCES*

12. Buneman, O.: A Compact Non-Iterative Poisson Solver, SUIPR Rep. 294, Institute for Plasma Research, Stanford Univ., May 1969.

18. Concus, P.; and Golub, G. H.: Use of Fast Direct Methods for the Efficient Numerical Solution of Non-separable Elliptic Equations. SIAM J. Numer. Anal., vol. 10, Dec. 1973, pp. 1103-1120.

35. Fromm, J. E.: A Numerical Study of Buoyancy Driven Flows in Room Enclosures. Proc. Second. Intern. Conf. on Numerical Methods in Fluid Dynamics, Sept. 15-19, 1970, Univ. of Calif., Berkeley, Maurice Holt, ed., Lecture Notes in Physics, vol. 8, Springer-Verlag (Berlin), 1971, pp. 120-126.

37. Gates, W. J.; and von Rosenberg, D. U.: An Improved Method of Numerical Solution for Flux Components in Potential Flow. Chemical Engineering Science, vol. 25, 1970, pp. 535-547.

48. Hockney, R. W.: Formation and Stability of Virtual Electrodes in a Cylinder. J. Appl. Phys., vol. 39, 1968, pp. 4166-4170.

53. Jameson, A.: Numerical Solution of Nonlinear Partial Differential Equations of Mixed Type, SYNSPADE 1975, Univ. of Maryland, May 1975.

61. Lomax, H.; and Martin, E. D.: Fast Direct Numerical Solution of the Nonhomogeneous Cauchy-Riemann Equations. J. Comp. Phys., vol. 15, May 1974, pp. 55-80.

66. Martin, E. D.: A Generalized-Capacity-Matrix Technique for Computing Aerodynamic Flows. Presented at the Symposium on Application of Computers to Fluid Dynamics Analysis and Design, Polytechnic Institute of Brooklyn Graduate Center, Farmingdale, N. Y., Jan. 1973; also Computers and Fluids, vol. 2, March 1974, pp. 79-97.

---

* The book to which this article is contributed contains a master list of references. Only those pertinent to this paper are listed here.

67. Martin, E. D.: Progress in Applications of Direct Elliptic Solvers to Transonic Flow Computations. Aerodynamic Analyses Requiring Advanced Computers, Part II. NASA SP-347, 1975, pp. 839-870.

68. Martin, E. D.: A Fast Semidirect Method for Computing Transonic Aerodynamic Flows. AIAA 2nd Computational Fluid Dynamics Conf. Proc., Hartford, Conn., June 1975, pp. 162-174; also AIAA J., vol. 14, July 1976, pp. 914-922; and Errata, AIAA J., vol. 14, Nov. 1976, p. 1664.

69. Martin, E. D.: Advances in the Application of Fast Semidirect Computational Methods in Transonic Flow. Presented at the IUTAM Symposium Transsonicum II, Göttingen, Germany, Sept. 8-13, 1975. In Symposium Transsonicum II, K. Oswatitsch and D. Rues, eds., Springer-Verlag (Berlin), 1976, pp. 431-438.

70. Martin, E. D.: A Technique for Accelerating Iterative Convergence in Numerical Integration, with Application in Transonic Aerodynamics, in Padé Approximants Method and Its Applications to Mechanics. Lecture Notes in Physics, Vol. 47 (ed. by H. Cabannes), Springer-Verlag, Berlin, 1976, pp. 123-139.

71. Martin, E. D.: A Fast Direct Cauchy-Riemann Solver for the Poisson-Dirichlet and Poisson-Neumann Boundary-Value Problems. In preparation.

72. Martin, E. D.: On Computation of Lifting Flows Using Fast Direct Elliptic Solvers with Conditions on a Slit. In preparation.

73. Martin, E. D.; and Lomax, H.: Rapid Finite-Difference Computation of Subsonic and Slightly Supercritical Aerodynamic Flows. AIAA J., vol. 13, May 1975, pp. 579-586.

74. Martin, E. D.; and Lomax, H.: Variants and Extensions of a Fast Direct Numerical Cauchy-Riemann Solver, with Illustrative Applications. NASA TN D-7934, 1977.

76. Murman, E. M.: Analysis of Embedded Shock Waves Calculated by Relaxation Methods. AIAA J., vol. 12, May 1974, pp. 626-633.

77. Murman, E. M.; Bailey, F. R.; and Johnson, M. H.: TSFOIL-A Computer Code for 2-D Transonic Calculations Including Wind-Tunnel Wall Effects and Wave-Drag Evaluation, in Aerodynamic Analyses Requiring Advanced Computers, Part II. NASA SP-347, 1975, pp. 769-788.

97. Roache, P. J.: Finite-Difference Methods for the Steady-State Navier-Stokes Equations. Rep. SC-RR-73-0419, Sandia Labs., Albuquerque, N.M., Dec. 1972.

141. Widlund, O. B.: On the Use of Fast Methods for Separable Finite-Difference Equations for the Solution of General Elliptic Problems. Sparse Matrices and Their Applications, D. J. Rose and R. A. Willoughby, eds., Plenum Press, 1972, pp. 121-134.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No
NASA TM 78,426		

4. Title and Subtitle	5. Report Date
FAST METHODS INCORPORATING DIRECT ELLIPTIC SOLVERS FOR NONLINEAR APPLICATIONS IN FLUID DYNAMICS*	
	6. Performing Organization Code

7. Author(s)	8. Performing Organization Report No.
E. Dale Martin	A-7160
9. Performing Organization Name and Address	10. Work Unit No.
Ames Research Center Moffett Field, Calif. 94035	505-06-12-08
	11. Contract or Grant No.
12. Sponsoring Agency Name and Address	13. Type of Report and Period Covered
National Aeronautics and Space Administration Washington, D.C. 20546	Technical Memorandum
	14. Sponsoring Agency Code

15. Supplementary Notes

*Contribution to the GAMM Workshop on Fast Solution Methods for the Discretized Poisson Equation, held at the Kernforschungszentrum, Karlsruhe, German, March 3-4, 1977.

16. Abstract

Semidirect methods are discussed, including their present role and anticipated future prospects, as well as some developments for their application in computational fluid dynamics. A semidirect method is a computational scheme that uses a fast, direct, elliptic solver as the driving algorithm for the iterative solution of finite-difference equations. Specific subtopics include: direct Cauchy-Riemann solvers for first-order elliptic equations; application of the semidirect method to the mixed elliptic-hyperbolic problem of steady, inviscid transonic flow; and the treatment of interior conditions, such as those on an airfoil or wing, in semidirect methods.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Computational fluid dynamics Transonic aerodynamics Numerical analysis Direct elliptic solvers	Unlimited STAR Category - 64

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	19	$3.25

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