Efficient Algorithm for Solution of the Unsteady Transonic Small-Disturbance Equation

John T. Batina*
NASA Langley Research Center, Hampton, Virginia

A time-accurate approximate factorization (AF) algorithm is formulated for solution of the three-dimensional unsteady transonic small-disturbance equation. The AF algorithm consists of a Newton linearization procedure coupled with an internal iteration technique. Superior stability characteristics of the new algorithm are demonstrated through applications to steady and oscillatory flows at subsonic and supersonic freestream conditions for an F-5 fighter wing. For steady flow calculations, the size of the time step is cycled to achieve rapid convergence. For unsteady flow calculations, the AF algorithm is sufficiently robust to allow the step size to be selected based on accuracy rather than on stability considerations. Therefore, accurate solutions are obtained in only several hundred time steps yielding a significant computational cost savings when compared to alternative methods.

Nomenclature

c = airfoil chord
c_r = wing reference chord
C_p = pressure coefficient
C_p^* = critical pressure coefficient
$\tilde{C}_p$ = unsteady pressure coefficient normalized by oscillation amplitude
f = function defining instantaneous position of wing
k = reduced frequency, $\omega c_r/2U$
M = freestream Mach number
t = time, nondimensionalized by freestream velocity and wing reference chord
U = freestream velocity
x,y,z = nondimensional Cartesian coordinates in streamwise, spanwise, and vertical directions, respectively
α_o = mean angle of attack
α_1 = amplitude of pitch oscillation
γ = ratio of specific heats
Γ = circulation
Δt = nondimensional time step
$\bar{\eta}$ = fractional semispan
Λ = leading-edge sweep angle
ϕ = disturbance velocity potential
ω = angular frequency

Subscripts
j = index of grid points in spanwise direction
J = index of grid point at wing root
k = index of grid points in vertical direction
TE = trailing edge

Introduction

PRESENTLY, considerable research is being conducted to develop finite-difference computer codes for calculating transonic unsteady aerodynamics for aeroelastic applications.[1] The computer codes are being developed to provide accurate methods of calculating unsteady airloads for the prediction of aeroelastic phenomena such as flutter and divergence. The most fully developed U.S. code for transonic aeroelastic analysis of isolated planar wings is XTRAN3S.[2] The code uses an alternating-direction implicit (ADI) finite-difference algorithm to calculate steady and unsteady transonic flows in a time-marching fashion. Several terms of the ADI algorithm are treated explicitly, though, which leads to a time step restriction based on numerical stability considerations. Experience with the code has shown that for applications to practical wings with moderate to high sweep and taper, very small time steps are required for the algorithm to be numerically stable.[3-6] This stability restriction typically results in thousands of time steps required to obtain converged solutions, which is generally many more time steps than is necessary to resolve the physics of the problem. Such solutions are computationally expensive and, thus, aeroelastic applications of XTRAN3S have generally been limited. An algorithm is therefore desired which is robust in comparison with the ADI algorithm to allow for efficient application to aeroelastic problems.

The purpose of this paper is to describe the development of a time-accurate approximate factorization (AF) algorithm for solution of the unsteady transonic small-disturbance (TSD) equation. The AF algorithm involves a Newton linearization procedure coupled with an internal iteration technique based on work recently reported by Shankar et al.[7] and Shankar and Ide.[8] In Refs. 7 and 8, these concepts were developed for the full-potential equation, and the resulting algorithm was shown to be very robust for application to either steady or oscillatory transonic flow problems. The objectives of the present research were to 1) develop a similar AF algorithm for solution of the unsteady TSD equation, 2) validate the new algorithm by making comparisons with available experimental data as well as with results computed using the XTRAN3S ADI algorithm, and 3) investigate the robustness and efficiency of the new solution procedure. The paper presents a description of the AF algorithm along with results and comparisons assessing this new capability.

Governing Equations

In this section, the equations governing the flow are briefly described, including the TSD equation and boundary conditions. Details of the coordinate transformation to map the physical domain into the computational domain are also given.

TSD Equation

The flow is governed by the modified TSD equation, which may be written in conservation law form as

$$\frac{\partial f_0}{\partial t}+\frac{\partial f_1}{\partial x}+\frac{\partial f_2}{\partial y}+\frac{\partial f_3}{\partial z}=0 \tag{1}$$

Presented as Paper 87-0109 at the AIAA 25th Aerospace Sciences Meeting, Reno, NV, Jan. 12–15, 1987; received Dec. 4, 1986; revision received July 10, 1987.

*Research Scientist, Unsteady Aerodynamics Branch, Loads and Aeroelasticity Division. Senior Member AIAA.

where

$$f_0 = -A\phi_t - B\phi_x \tag{2a}$$

$$f_1 = E\phi_x + F\phi_x^2 + G\phi_y^2 \tag{2b}$$

$$f_2 = \phi_y + H\phi_x\phi_y \tag{2c}$$

$$f_3 = \phi_z \tag{2d}$$

The coefficients A, B, and E are defined as

$$A = M^2, \qquad B = 2M^2, \qquad E = 1 - M^2 \tag{3}$$

Several choices are available for the coefficients F, G, and H depending upon the assumptions used in deriving the TSD equation.[2] In this study, the coefficients are defined as

$$F = -\tfrac{1}{2}(\gamma + 1)M^2 \tag{4a}$$

$$G = \tfrac{1}{2}(\gamma - 3)M^2 \tag{4b}$$

$$H = -(\gamma - 1)M^2 \tag{4c}$$

Boundary Conditions

Boundary conditions imposed upon the flowfield are

Far upstream: $$\phi = 0 \tag{5a}$$

Far downstream: $$\phi_x + \phi_t = 0 \tag{5b}$$

Far above and below: $$\phi_z = 0 \tag{5c}$$

Far spanwise: $$\phi_y = 0 \tag{5d}$$

Symmetry plane: $$\phi_y = 0 \tag{5e}$$

Trailing wake: $$[\phi_z] = 0 \tag{5f}$$

$$[\phi_x + \phi_t] = 0 \tag{5g}$$

where [ ] indicates the jump in the indicated quantity across the wake. The wing flow-tangency boundary condition is

$$\phi_z^{\pm} = f_x^{\pm} + f_t \tag{6}$$

which is imposed at the mean plane of the wing. The plus and minus superscripts indicate the upper and lower wing surfaces, respectively.

Coordinate Transformation

The finite-difference grids in both the physical and computational domains are contained within rectangular regions and conform to the wing planform. Regions in the physical domain such as the swept and/or tapered wing are mapped into rectangular regions in the computational domain using the shearing transformation

$$\xi = \xi(x,y), \qquad \eta = y, \qquad \zeta = z \tag{7}$$

where ξ, η, and ζ are the nondimensional computational coordinates in the x, y, and z directions, respectively. The TSD equation, Eq. (1), may then be expressed in computational coordinates as

$$-\frac{\partial}{\partial t}\left[\frac{A}{\xi_x}\phi_t + B\phi_\xi\right] + \frac{\partial}{\partial \xi}\left[E\xi_x\phi_\xi + F\xi_x^2\phi_\xi^2 + G(\xi_y\phi_\xi + \phi_\eta)^2 + \frac{\xi_y}{\xi_x}(\xi_y\phi_\xi + \phi_\eta) + H\xi_y\phi_\xi(\xi_y\phi_\xi + \phi_\eta)\right] + \frac{\partial}{\partial \eta}\left[\frac{1}{\xi_x}(\xi_y\phi_\xi + \phi_\eta) + H\phi_\xi(\xi_y\phi_\xi + \phi_\eta)\right] + \frac{\partial}{\partial \zeta}\left[\frac{1}{\xi_x}\phi_\zeta\right] = 0 \tag{8}$$

Alternating-Direction Implicit Algorithm

An alternating-direction implicit algorithm was developed in Ref. 2 for solution of the modified TSD equation [Eq. (1)]. This algorithm forms the basis of the XTRAN3S computer code, which can be used to calculate steady and unsteady transonic flowfields about planar wings, including aeroelastic deformation effects. The program is capable of treating either forced harmonic or aeroelastic transient-type motions. Details of the algorithm development and solution procedure are given in Ref. 2. The original XTRAN3S code used a simple global shearing transformation that restricted applications to wings with mild sweep and taper. Alternative coordinate transformations[3–5] were subsequently developed to allow application to practical wings. Although the code can now be applied to more general planforms, several of the terms of the TSD equation are treated explicitly in the ADI algorithm, which leads to a time step restriction based on numerical stability considerations. For application to wings with moderate to high sweep and taper, very small time steps are required for the algorithm to be numerically stable. For example, a summary of selected time steps and number of steps per cycle is listed in Table 1; the values are taken from Refs. 3–6. Cases are tabulated for three wings, the F-5 wing,[3,4] the RAE tailplane,[5] and the B-1 wing.[6] (The calculations for these wings were performed using different computational grids and different coordinate transformations.) As shown in the table, the stability restriction typically results in thousands of time steps required to obtain converged steady-state solutions and thousands of steps per cycle of forced harmonic or aeroelastic motion. These solutions are computationally expensive and, thus, aeroelastic applications of XTRAN3S have generally been limited.

To investigate the stability problem of the ADI algorithm, a numerical stability analysis was performed for the F-5 wing. Shown in Fig. 1 are the resulting stability boundaries calculated using the algorithm started from an initial undisturbed flow. The boundaries were determined by first selecting a step size of $\Delta t = 0.05$ and running the algorithm until the solution diverged, causing program failure. The step size was then successively reduced and the calculations repeated until the boundary was determined. The analysis was performed for freestream Mach numbers of $M = 0.6$, 0.8, 0.9, and 0.95. As shown in the figure, the ADI algorithm is unstable in the region above and to the right of the boundaries; the algorithm is stable in the region below and to the left. The stability boundaries indicate that a very small step size is required to obtain stable solutions using the ADI algorithm. In fact, this step size is excessively small, as demonstrated by the following example. The number of steps per cycle N for oscillatory flow is determined for a given time step by

$$N = \frac{\pi}{k\,\Delta t} \tag{9}$$

If the maximum allowable time step is 0.01 and the reduced frequency k is 0.1, then Eq. (9) indicates that over 3000 steps per cycle are required for numerical stability. This is an order-of-magnitude more steps than is necessary for accurate aeroelastic calculations.

To investigate the problem further, stability boundaries were obtained for several leading-edge sweep angles by shearing the F-5 planform aft. Values selected for the sweep angle include $\Lambda = 32$ deg, the actual sweep of the F-5 wing, and $\Lambda = 45$ and 60 deg. Shown in Fig. 2 are the resulting stability boundaries for the freestream Mach number of 0.9. The boundaries indicate that as the sweep increases, smaller time steps are required for numerical stability. For the $\Lambda = 60$ deg case, for example, the maximum allowable time step is approximately 0.002.

Approximate Factorization Algorithm

An approximate factorization algorithm was developed as an alternative solution to the modified TSD equation [Eq. (1)].

Table 1 Summary of selected time steps and steps per cycle using the XTRAN3S ADI algorithm reported in Refs. 3–6

Reference	Wing	Leading-edge sweep angle, deg	Taper ratio	Aspect ratio	Steady		Unsteady	
					Δt	Total steps	Case	Steps/cycle
3	F-5	31.9	0.28	3.16	0.01	2000	40 Hz, pitch	1200
4	F-5	31.9	0.28	3.16	0.01	4000	20 Hz, pitch	2160
5	RAE	50.2	0.27	2.41	0.0075	1500	70 Hz, pitch	1000
							33 Hz, pitch	2000
6	B-1	67.5	0.38	1.85	0.002	not reported	aeroelastic	6000

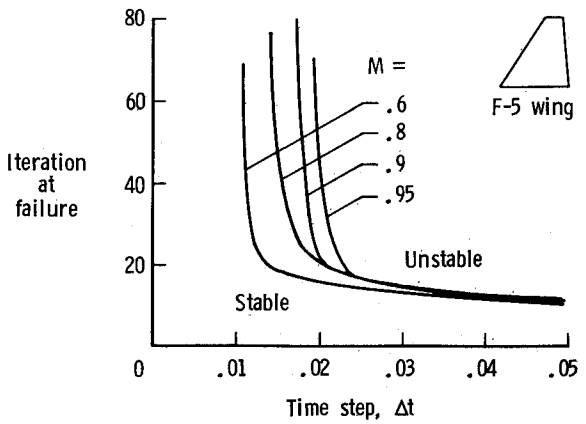


Fig. 1 Numerical stability boundaries of the ADI algorithm computed from an initial undisturbed flow past the F-5 wing.

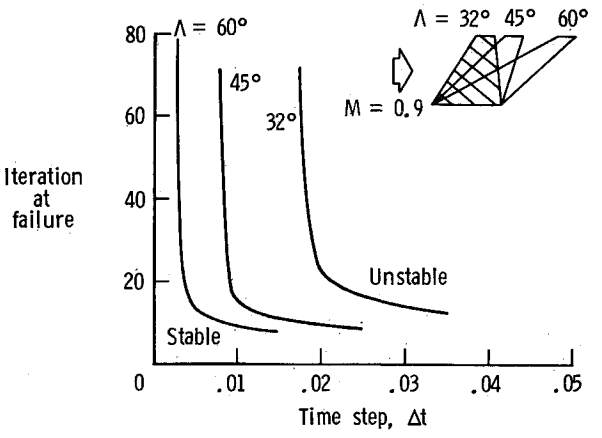


Fig. 2 Numerical stability boundaries of the ADI algorithm computed from an initial undisturbed flow past the F-5 wing for various sweep angles.

In this section, the AF algorithm is described in detail, including the mathematical formulation, finite-difference discretization, boundary conditions, and solution procedure.

General Description

The AF algorithm consists of a Newton linearization procedure coupled with an internal iteration technique. For unsteady flow calculations, the solution procedure involves two steps. First, a time linearization step (which will be described) is performed to determine an estimate of the potential field. Second, internal iterations are performed to provide time accuracy. More specifically, the TSD equation is written in general form as

$$R(\phi^{n+1}) = 0 \tag{10}$$

where ϕ^{n+1} represents the unknown potentials at time level $(n+1)$. The solution to Eq. (10) is then given by the Newton linearization of Eq. (10) about ϕ^*:

$$R(\phi^*) + \left(\frac{\partial R}{\partial \phi}\right)_{\phi=\phi^*} \Delta\phi = 0 \tag{11}$$

In Eq. (11), ϕ^* is the currently available value of ϕ^{n+1} and $\Delta\phi = \phi^{n+1} - \phi^*$. During convergence of the iteration procedure, $\Delta\phi$ will approach zero so that the solution will be given by $\phi^{n+1} = \phi^*$. In general, only one or two iterations are required to achieve acceptable convergence. For steady flow calculations, iterations are not used, since time accuracy is not necessary when marching to steady-state.

Mathematical Formulation

The AF algorithm is formulated by first approximating the time derivative terms (ϕ_{tt} and ϕ_{xt}) by second-order-accurate finite-difference formulas. By substituting $\phi = \phi^* + \Delta\phi$ into the TSD equation and neglecting squares of derivatives of $\Delta\phi$ [which is equivalent to applying Eq. (11) term by term], each term of Eq. (1) may then be rewritten as

$$\frac{\partial f_0}{\partial t} = -A\frac{2\phi^* - 5\phi^n + 4\phi^{n-1} - \phi^{n-2}}{\Delta t^2} - \frac{2A}{\Delta t^2}\Delta\phi - B\frac{3\phi_x^* - 4\phi_x^n + \phi_x^{n-1}}{2\Delta t} - \frac{3B}{2\Delta t}\Delta\phi_x \tag{12a}$$

$$\frac{\partial f_1}{\partial x} = \frac{\partial}{\partial x}(E\phi_x^* + F\phi_x^{*2} + G\phi_y^{*2}) + \frac{\partial}{\partial x}(E\,\Delta\phi_x + 2F\phi_x^*\,\Delta\phi_x + 2G\phi_y^*\,\Delta\phi_y) \tag{12b}$$

$$\frac{\partial f_2}{\partial y} = \frac{\partial}{\partial y}(\phi_y^* + H\phi_x^*\phi_y^*) + \frac{\partial}{\partial y}(\Delta\phi_y + H\phi_x^*\,\Delta\phi_y + H\phi_y^*\,\Delta\phi_x) \tag{12c}$$

$$\frac{\partial f_3}{\partial z} = \frac{\partial}{\partial z}(\phi_z^*) + \frac{\partial}{\partial z}(\Delta\phi_z) \tag{12d}$$

Summing these four terms and rearranging gives

$$\frac{2A}{\Delta t^2}\Delta\phi + \frac{3B}{2\Delta t}\Delta\phi_x - \frac{\partial}{\partial x}(E\,\Delta\phi_x + 2F\phi_x^*\,\Delta\phi_x + 2G\phi_y^*\,\Delta\phi_y) - \frac{\partial}{\partial y}(\Delta\phi_y + H\phi_x^*\,\Delta\phi_y + H\phi_y^*\,\Delta\phi_x) - \frac{\partial}{\partial z}(\Delta\phi_z)$$
$$= -A\frac{2\phi^* - 5\phi^n + 4\phi^{n-1} - \phi^{n-2}}{\Delta t^2} - B\frac{3\phi_x^* - 4\phi_x^n + \phi_x^{n-1}}{2\Delta t} + \frac{\partial}{\partial x}(E\phi_x^* + F\phi_x^{*2} + G\phi_y^{*2}) + \frac{\partial}{\partial y}(\phi_y^* + H\phi_x^*\phi_y^*) + \frac{\partial}{\partial z}(\phi_z^*) \tag{13}$$

The right-hand side of Eq. (13) is simply the TSD equation, which may be evaluated using known potentials ϕ^*, ϕ^n, ϕ^{n-1}, and ϕ^{n-2}. The left-hand side of Eq. (13) consists of terms containing $\Delta\phi$ and its spatial derivatives. At convergence, ϕ^* approaches ϕ^{n+1} so that $\Delta\phi$ vanishes and the left-hand side becomes zero.

Equation (13) is transformed into computational coordinates using Eq. (7) and is rewritten in conservation form for solution by approximate factorization. Each term is multiplied by $\Delta t^2/(2A)$, and the left-hand side is approximately factored into a triple product of operators, yielding

$$L_\xi L_\eta L_\zeta \Delta\phi = -R(\phi^*,\phi^n,\phi^{n-1},\phi^{n-2}) \tag{14}$$

where

$$L_\xi = 1 + \frac{3B}{4A}\xi_x \Delta t \frac{\partial}{\partial\xi} - \xi_x \frac{\Delta t^2}{2A}\frac{\partial}{\partial\xi} F_1 \frac{\partial}{\partial\xi} \tag{15a}$$

$$L_\eta = 1 - \xi_x \frac{\Delta t^2}{2A}\frac{\partial}{\partial\eta} F_2 \frac{\partial}{\partial\eta} \tag{15b}$$

$$L_\zeta = 1 - \xi_x \frac{\Delta t^2}{2A}\frac{\partial}{\partial\zeta} F_3 \frac{\partial}{\partial\zeta} \tag{15c}$$

$$F_1 = E\xi_x + 2F\xi_x^2\phi_\xi^* + 2G\xi_y(\xi_y\phi_\xi^* + \phi_\eta^*) + \frac{\xi_y^2}{\xi_x}(1 + H\xi_x\phi_\xi^*) + H\xi_y(\xi_y\phi_\xi^* + \phi_\eta^*) \tag{15d}$$

$$F_2 = \frac{1}{\xi_x}(1 + H\xi_x\phi_\xi^*) \tag{15e}$$

$$F_3 = \frac{1}{\xi_x} \tag{15f}$$

$$\begin{aligned} R = -\xi_x \frac{\Delta t^2}{2A}\Bigg\{ & -\frac{A}{\xi_x}\frac{2\phi^* - 5\phi^n + 4\phi^{n-1} - \phi^{n-2}}{\Delta t^2} \\ & - B\frac{3\phi_\xi^* - 4\phi_\xi^n + \phi_\xi^{n-1}}{2\Delta t} \\ & + \frac{\partial}{\partial\xi}\Bigg[E\xi_x\phi_\xi^* + F\xi_x^2\phi_\xi^{*2} + G(\xi_y\phi_\xi^* + \phi_\eta^*)^2 \\ & + \frac{\xi_y}{\xi_x}(\xi_y\phi_\xi^* + \phi_\eta^*) + H\xi_y\phi_\xi^*(\xi_y\phi_\xi^* + \phi_\eta^*)\Bigg] \\ & + \frac{\partial}{\partial\eta}\Bigg[\frac{1}{\xi_x}(\xi_y\phi_\xi^* + \phi_\eta^*) + H\phi_\xi^*(\xi_y\phi_\xi^* + \phi_\eta^*)\Bigg] \\ & + \frac{\partial}{\partial\zeta}\Bigg[\frac{1}{\xi_x}\phi_\xi^*\Bigg]\Bigg\} \end{aligned} \tag{15g}$$

Equation (14) is solved using three sweeps through the grid by sequentially applying the operators L_ξ, L_η, and L_ζ as

$$\xi \text{ sweep: } L_\xi \Delta\bar{\phi} = -R \tag{16a}$$

$$\eta \text{ sweep: } L_\eta \Delta\bar{\bar{\phi}} = \Delta\bar{\phi} \tag{16b}$$

$$\zeta \text{ sweep: } L_\zeta \Delta\phi = \Delta\bar{\bar{\phi}} \tag{16c}$$

Spatial Discretization

Central difference formulas are employed for all of the derivatives on the left-hand sides of Eqs. (16) except for the second term in the L_ξ operator (from the ϕ_{xt} term), which is backward-differenced to maintain stability, and the third term in the L_ξ operator, which is split into streamwise and spanwise components. The resulting terms are centrally differenced at subsonic points, and the streamwise terms are upwind-biased at supersonic points using the Murman[9] type-dependent mixed-difference operator. The terms on the right-hand side of the ξ sweep are also approximated using central-difference operators except for the ϕ_{xt} term, which is backward differenced, and for terms in the streamwise direction, which are upwind biased at supersonic points based on the sign of $E + 2F\phi_x^*$.

In the present coding of the AF algorithm, the time derivatives are implemented for variable time stepping to allow for step-size cycling to accelerate convergence to steady-state. In these calculations, the step size is cycled using a standard geometric sequence. Also, since the L_ξ, L_η, and L_ζ operators only contain derivatives in their respective coordinate directions, all three sweeps may be vectorized.

Time-Linearization Step

An initial estimate of the potentials at time level $(n+1)$ is required to start the iteration process. This estimate is provided by performing a time-linearization calculation. The equations governing the time-linearization step are derived in a similar fashion as the equations for iteration. The only difference is that the equations are formulated by linearizing about time level (n) rather than the iterate level (*). So by substituting $\phi = \phi^n + \Delta\phi$ into the TSD equation [Eq. (1)] and neglecting squares of derivatives of $\Delta\phi$, the time-linearization step may be written as

$$L_\xi L_\eta L_\zeta \Delta\phi = -R(\phi^n,\phi^{n-1},\phi^{n-2}) \tag{17}$$

where the operators L_ξ, L_η, and L_ζ are defined by Eqs. (15a), (15b), and (15c), respectively, with ϕ^* replaced by ϕ^n in the definitions of F_1 and F_2 [Eqs. (15d) and (15e)].

Boundary Conditions

The boundary conditions are numerically imposed by redefining the L_ξ, L_η, and L_ζ operators in Eq. (14) as well as the right-hand side R at the appropriate grid points. The equation to be solved at boundary grid points may then be written symbolically as

$$\tilde{L}_\xi \tilde{L}_\eta \tilde{L}_\zeta \Delta\phi = -\tilde{R} \tag{18}$$

where the tilde indicates that the quantity has been modified or rewritten to account for the boundary conditions. In the following subsections, the numerical treatment of each of the boundary conditions is briefly described.

Wing

The wing flow-tangency boundary condition [Eq. (6)] is imposed within the differencing of the $\partial(\phi_\zeta^*)/\partial\zeta$ term, which appears in R [Eq. (15g)] on the right-hand side of Eq. (14) and within the L_ζ operator in the time-linearization step for unsteady calculations. In general, the $\partial(\phi_\zeta^*)/\partial\zeta$ term is approximated by

$$\frac{\partial}{\partial\zeta}(\phi_\zeta^*)_k = \frac{2}{\zeta_{k+1} - \zeta_{k-1}}(\phi^*_{\zeta_{k+\frac{1}{2}}} - \phi^*_{\zeta_{k-\frac{1}{2}}}) \tag{19}$$

where the derivatives on the right-hand side are written about half-node points. The wing is located equidistantly between grid lines so that in the plane directly above the wing, the $\phi^*_{\zeta_{k-\frac{1}{2}}}$ derivative in Eq. (19) is replaced by $(f_x^+ + f_t)^{n+1}$, and in the plane below the wing, the $\phi^*_{\zeta_{k+\frac{1}{2}}}$ derivative in Eq. (19) is replaced by $(f_x^- + f_t)^{n+1}$. The L_ζ operator is similarly modified in the time-linearization step, which results in a bi-diagonal ζ-sweep equation.

Wake

The wake boundary condition [Eqs. (5f) and (5g)] is similar to the wing flow-tangency boundary condition in that it is imposed within the $\partial(\phi_\zeta^*)/\partial\zeta$ term and the right-hand side of the ζ sweep. The wake circulation Γ is calculated from Eq. (5g), which is equivalent to

$$\Gamma_x + \Gamma_t = 0 \tag{20}$$

Starting from the trailing-edge value given by $\Gamma_{TE} = \phi^+_{TE} - \phi_{TE}$, the circulation is convected downstream by integrating

Eq. (20) using second-order-accurate finite-difference approximations for Γ_x and Γ_t. The wake circulation is incorporated within the solution procedure by requiring that the perturbation velocity in the vertical direction be continuous across the wake [Eq. (5f)].

Symmetry Plane

The symmetry condition [Eq. (5e)] is imposed at the plane of the wing root $j = J$ by requiring that

$$(\xi_y \phi^*_\xi + \phi^*_\eta)_{J-\frac{1}{2}} = -(\xi_y \phi^*_\xi + \phi^*_\eta)_{J+\frac{1}{2}} \tag{21a}$$

$$(\xi_y \phi^*_\xi + \phi^*_\eta)_J = 0 \tag{21b}$$

Equations (21a) and (21b) affect terms that appear on the right-hand side of Eq. (14). Equation (21b) causes two terms in F_1 [Eq. (15d)] to vanish. The L_η operator [Eq. (15b)] is also modified at $j = J$ by requiring that

$$\left(F_2 \frac{\partial}{\partial \eta} \Delta\bar{\phi}\right)_{J-\frac{1}{2}} = -\left(F_2 \frac{\partial}{\partial \eta} \Delta\bar{\phi}\right)_{J+\frac{1}{2}} \tag{22}$$

which results in an upper bidiagonal η-sweep equation at the wing root. The condition at the far spanwise boundary [Eq. (5d)] is identical to the symmetry condition [Eq. (5e)] and thus is treated similarly.

Farfield

The farfield boundary conditions [Eqs. (5a–5c)] are imposed by writing finite-difference approximations for these equations, casting them in the form of Eq. (18), and including them with the system of simultaneous equations that results from Eq. (14) for solution.

Results and Discussion

Calculations were performed for the F-5 wing to assess the accuracy and efficiency of the AF algorithm. The wing has a full-span aspect ratio of 3.16, a leading-edge sweep angle of 31.9 deg, and a taper ratio of 0.28. The airfoil section of the F-5 wing is a modified NACA 65A004.8 airfoil that has a drooped nose and is symmetric aft of 40% chord. The AF results are compared with parallel calculations performed using the ADI algorithm. In these calculations, identical grids and coordinate transformations [Eq. (7)] were used to allow for a direct comparison between AF and ADI results. The grid and transformation were identical to those used in Ref. 4. Furthermore, the results are compared with the experimental data from an F-5 wing model tested by Tijdeman et al.[10] In Ref. 10, steady and oscillatory pressure distributions were measured at eight span stations along the wing. The stations were located at $\bar{\eta} = 0.18$, 0.36, 0.51, 0.64, 0.72, 0.82, 0.88, and 0.98. In this paper, comparisons are presented at the first, third, fifth, and seventh span stations. The tests covered the Mach number range from $M = 0.6$–1.35. In this study, two Mach number cases were selected for application of the new algorithm. The first case, herein referred to as case 1, was chosen to have the same freestream conditions as investigated in Refs. 3 and 4. In case 1, the freestream Mach number was 0.9 and the mean angle of attack was 0 deg. At these conditions, both steady and unsteady results were obtained. The unsteady calculations were performed for the rigid wing pitching harmonically about a line perpendicular to the root at the root midchord. The second case, herein referred to as case 2, was chosen to assess the performance of the AF algorithm for supersonic freestream conditions. In case 2, the freestream Mach number was 1.1 and the mean angle of attack was 0 deg. No ADI calculations were attempted for this case since the algorithm is unstable for supersonic freestream cases for swept wings.

Case 1: $M = 0.9$ and $\alpha_0 = 0$ deg

For case 1, steady calculations were performed using both the ADI and AF algorithms. The ADI results were obtained using a constant step size of $\Delta t = 0.01$, which is the same as that reported in Refs. 3 and 4. These calculations were performed for a total of 4000 time steps. The AF results were obtained by cycling the step size through a range of values between $\Delta t = 0.05$ and 5.0. A total of 400 time steps were run. A comparison of steady-state convergence between the ADI and AF algorithms is shown in Fig. 3. After 4000 time steps, the maximum $|\Delta\phi|$ in the ADI solution has been reduced to approximately $10^{-6.5}$, whereas the AF solution has achieved similar convergence in less than 400 steps. The total lift and moment in the AF solution were converged after approximately 100 steps. Steady calculations were also performed for the high sweep case of $\Lambda = 60$ deg to assess the convergence characteristics of the two algorithms further. The ADI results were obtained using a constant step size of $\Delta t = 0.002$, and the AF results were obtained by cycling the step size through a range of values between $\Delta t = 0.05$ and 0.5. A comparison of the steady-state convergence histories between the two algorithms is shown in Fig. 4. The ADI solution converges very slowly such that after 4000 steps, the maximum $|\Delta\phi|$ has been reduced to approximately 10^{-5}. The AF solution, however, has converged very quickly to maximum $|\Delta\phi| < 10^{-7}$ in only 250 steps.

A comparison of the ADI and AF steady pressure distributions for the F-5 wing is shown in Fig. 5. These pressure distributions indicate that there is an embedded supersonic region on the wing upper surface and that the flow is slightly supercritical along the lower surface. The AF pressures are nearly identical to the ADI pressures, and both sets of results are in good general agreement with the experimental data. The AF algorithm thus produced the same steady-state solution for this case as the ADI algorithm at one-tenth of the computational cost.

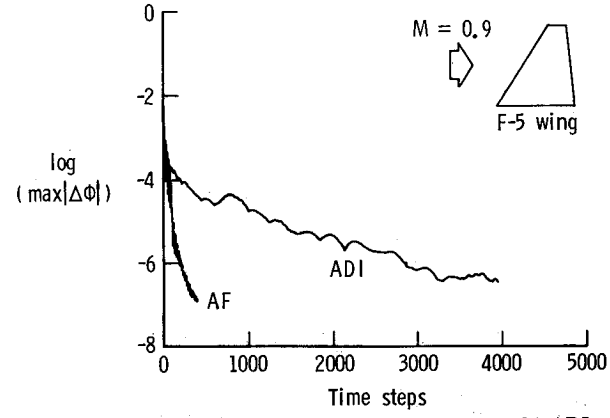


Fig. 3 Comparison of steady-state convergence between the ADI and AF algorithms for the F-5 wing at $M = 0.9$ and $\alpha_0 = 0$ deg.

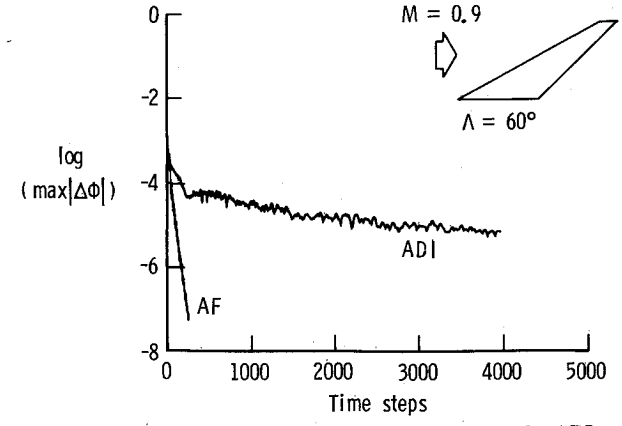


Fig. 4 Comparison of steady-state convergence between the ADI and AF algorithms for the F-5 wing sheared to $\Lambda = 60$ deg at $M = 0.9$ and $\alpha_0 = 0$ deg.

Unsteady results were obtained using both algorithms for the wing oscillating with amplitude $\alpha_1 = 0.109$ deg at a reduced frequency of $k = 0.137$ for comparison with the experimental data. The ADI calculations were performed using 2000 steps per cycle of motion, which corresponds to a step size of $\Delta t = 0.0115$. (In Ref. 4, similar calculations were performed using 2160 steps per cycle with a step size of $\Delta t = 0.0106$.) For unsteady calculations with the ADI algorithm, the step size is typically selected based on numerical stability considerations rather than on accuracy. Because of the stability restriction inherent in the ADI algorithm, this generally leads to using many more time steps per cycle than is necessary to resolve the physics of the problem. The AF algorithm, however, allows the step size to be selected based on accuracy rather than on stability. Consequently, a convergence study was performed using the AF algorithm to determine the largest step size (fewest number of steps per cycle) that would produce converged answers. Unsteady results were obtained for 100, 200, 300, and 400 steps per cycle, which required $\Delta t = 0.2293, 0.1147, 0.0764$, and 0.0573, respectively. Only one iteration per time step was required to satisfy the convergence criteria of maximum $|\Delta\phi| < 10^{-6}$ for the latter three cases. For the 100 steps per cycle case, two iterations per time step were required during approximately 60% of the cycle of motion. Selected results from this convergence study are plotted in Fig. 6 for 200 and 300 steps per cycle. The unsteady pressure coefficients along the upper surface are shown in Fig. 6a; the unsteady pressure coefficients along the lower surface are shown in Fig. 6b. These coefficients are plotted as real and imaginary components corresponding to the in-phase and out-of-phase unsteady pressure distributions normalized by the amplitude of motion. The calculation for 100 steps per cycle of motion produced reasonable results, but fairly large differences were observed in comparison with the 200 steps per cycle calculation. As shown in Fig. 6, the results for 200 and 300 steps per cycle are very similar with the largest differences occurring near the leading edge and in the shock pulse region on the upper surface. The pressure distributions computed using 400 steps per cycle are identical, to plotting accuracy, to those computed using 300 steps per cycle and therefore are not shown. A converged solution hence requires approximately 300 steps per cycle of motion for this case, although the results computed using 200 steps may be acceptable for engineering purposes.

Figure 7 shows a comparison between the ADI and AF unsteady pressures for case 1 along with the experimental data. Upper-surface pressure distributions are shown in Fig. 7a; lower-surface pressure distributions are shown in Fig. 7b. The two sets of calculated results are very similar, and both sets generally agree well with the experimental data. On the upper surface, however, the agreement is better in the inboard region of the wing ($\bar{\eta} = 0.18$, for example). The AF algorithm (using 300 steps per cycle and one iteration per time step) thus produced approximately the same unsteady solution for this case as the ADI algorithm (using 2000 steps per cycle) at 30% of the computational cost.

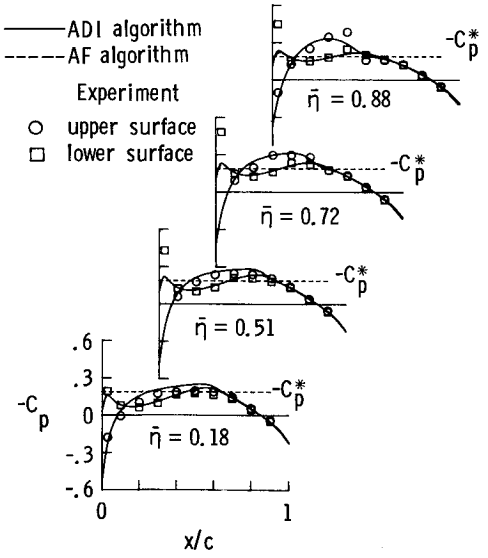
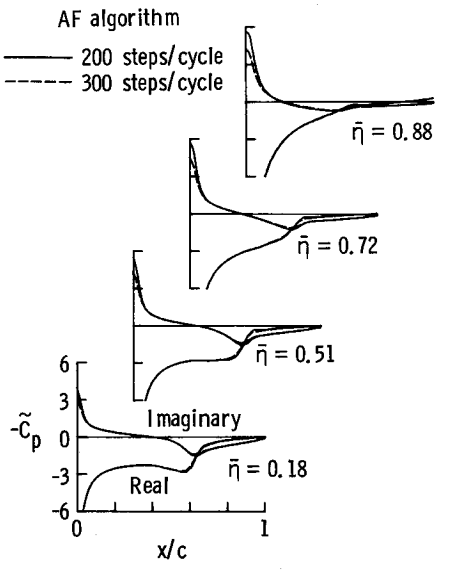


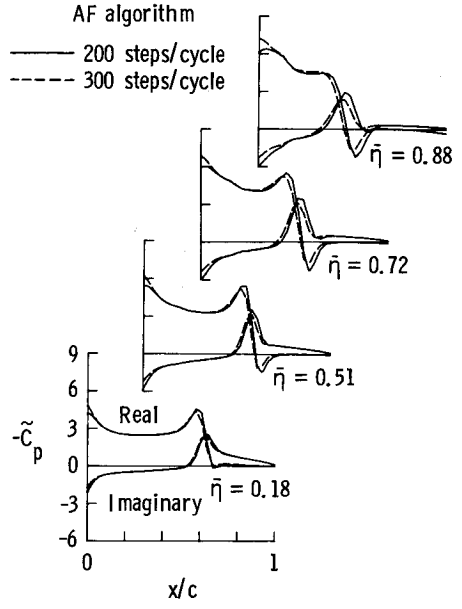
Fig. 5 Comparison of F-5 wing steady pressure distributions at $M = 0.9$ and $\alpha_0 = 0$ deg.

Case 2: $M = 1.1$ and $\alpha_0 = 0$ deg

For case 2, calculations were performed using only the AF algorithm. The steady results were obtained by cycling the step size through a range of values between $\Delta t = 0.01$ and 1.0. The total lift and moment were converged after approximately 150 steps, and the maximum $|\Delta\phi|$ was reduced to approximately $10^{-6.5}$ after 400 steps. The resulting steady pressure distributions are compared with the experimental data in Fig. 8. For this supersonic freestream case, the upper- and lower-surface pressure levels are well predicted except near the leading edge. In general, the pressure distributions computed using the AF algorithm are in good agreement with the experimental data.

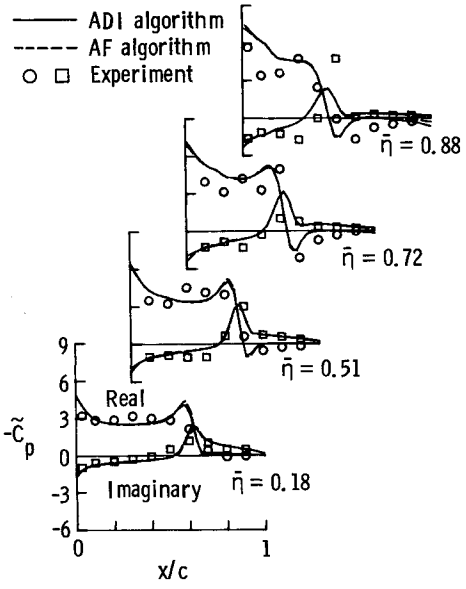


a)



b)

Fig. 6 Convergence study using AF algorithm for F-5 wing unsteady pressure distributions due to wing pitching at $M = 0.9$, $\alpha_0 = 0$ deg, $\alpha_1 = 0.109$ deg, and $k = 0.137$: a) upper surface; b) lower surface.



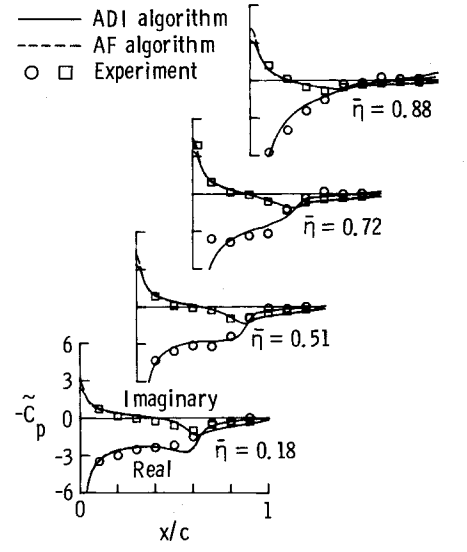


Fig. 7 Comparison of F-5 wing unsteady pressure distributions due to wing pitching at $M = 0.9$, $\alpha_0 = 0$ deg, $\alpha_1 = 0.109$ deg, and $k = 0.137$: a) upper surface; b) lower surface.

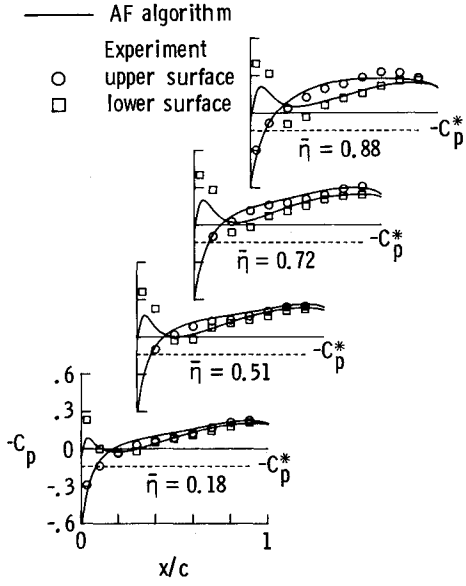
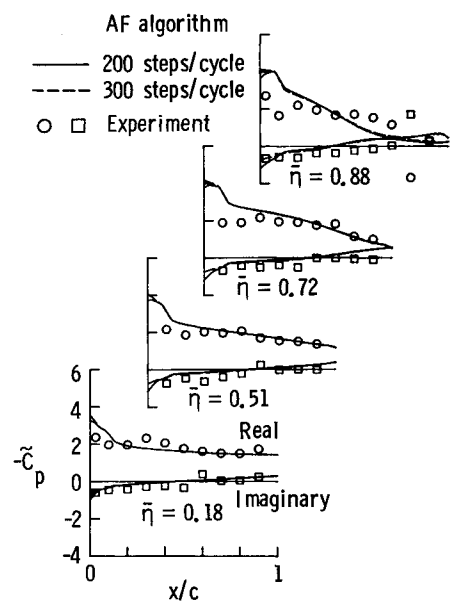


Fig. 8 Comparison of F-5 wing steady pressure distributions at $M = 1.1$ and $\alpha_0 = 0$ deg.



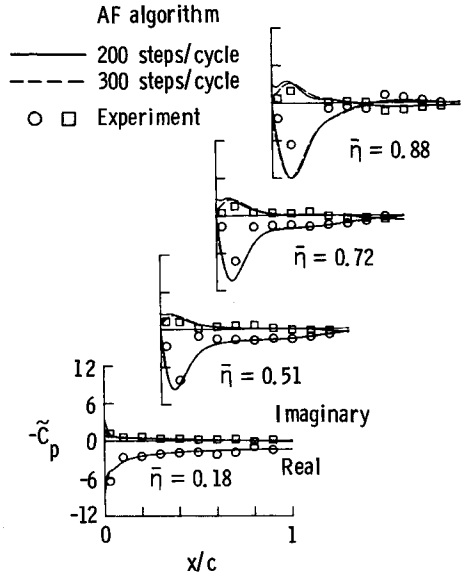


Fig. 9 Comparison of F-5 wing unsteady pressure distributions due to wing pitching at $M = 1.1$, $\alpha_0 = 0$ deg, $\alpha_1 = 0.267$ deg, and $k = 0.116$: a) upper surface; b) lower surface.

Unsteady results were obtained using the AF algorithm for the wing oscillating with amplitude $\alpha_1 = 0.267$ deg at a reduced frequency of $k = 0.116$. A convergence study was performed using 100, 200, 300, and 400 steps per cycle, which corresponds to $\Delta t = 0.2708$, 0.1354, 0.0903, and 0.0677, respectively. Unsteady pressure distributions for 200 and 300 steps per cycle are plotted in Fig. 9 along with the experimental data. The upper-surface pressures are shown in Fig. 9a and the lower-surface pressures are shown in Fig. 9b. Similar to the unsteady pressures of case 1, approximately 300 steps per cycle are required in case 2 for convergence. Along the upper surface, the AF pressures generally compare well with the experimental data except in the real part near the leading edge and outboard near the tip. Along the lower surface, the AF results again compare well with the data, and the calculation here predicts the large change in the real part near 10% chord at $\bar{\eta} = 0.51$, 0.72, and 0.88. Therefore, the AF algorithm is accurate and efficient for application to supersonic as well as subsonic freestream cases.

Concluding Remarks

A time-accurate approximate factorization (AF) algorithm has been developed for the solution of the unsteady transonic

small-disturbance equation. The new algorithm was developed as an alternative solution procedure to the alternating-direction implicit (ADI) algorithm. The AF algorithm consists of a Newton linearization procedure coupled with an internal iteration technique. For unsteady flow calculations, the solution procedure involves two steps. First, a time-linearization step is performed to determine an estimate of the potential field. Second, internal iterations are performed to provide time accuracy. In general, only one or two iterations are required to achieve acceptable convergence. For steady flow calculations, iterations are not used since time accuracy is not necessary when marching to steady-state.

The AF algorithm has been used to calculate steady and oscillatory transonic flows at subsonic and supersonic freestream conditions. These calculations were performed for the F-5 wing, and comparisons were made with experimental data to assess the accuracy of the algorithm. The results were further compared with pressure distributions computed using the ADI algorithm for the subsonic freestream case. In general, the steady pressures calculated using the two algorithms were nearly identical, and both sets of results compared well with the experimental data. The ADI results were computed using 4000 time steps, whereas the AF results were computed using only 400 time steps to achieve similar convergence. The new algorithm, therefore, gives an order-of-magnitude decrease in computational cost for steady-state calculations. Unsteady pressures due to harmonic wing pitching were very similar for the two algorithms, and again, both sets of results agreed well with the experimental data. The ADI results were computed using 2000 steps per cycle with a time step based on numerical stability considerations. The AF results, however, were computed using only 300 steps per cycle since the step size is selected for accuracy rather than for stability. Hence, for application to unsteady flow problems, the AF algorithm again yields a significant decrease in computational cost.

For a supersonic freestream case, the pressure distributions computed using the AF algorithm were in good agreement with the experimental pressures for both steady and oscillatory flows. (An ADI solution for this case was not obtained since the algorithm is unstable for swept-wing cases with supersonic freestream conditions.) Similar to the subsonic freestream case, the steady-state AF results were obtained using only 400 time steps, and the oscillatory AF results were obtained using only 300 steps per cycle. Therefore, the AF algorithm is robust and efficient for application to steady or unsteady transonic flows with subsonic or supersonic freestream conditions. The new algorithm can provide accurate solutions in only several hundred time steps, yielding a significant computational cost savings when compared to the ADI method.

References

[1]*Proceedings of the AGARD Meeting on Transonic Unsteady Aerodynamics and its Aeroelastic Applications*, AGARD-CP-374, Sept. 1984.

[2]Borland, C. J. and Rizzetta, D. P., "Nonlinear Transonic Flutter Analysis," *AIAA Journal*, Vol. 20, Nov. 1982, pp. 1606–1615.

[3]Guruswamy, G. P. and Goorjian, P. M., "Efficient Algorithm for Unsteady Transonic Aerodynamics of Low-Aspect-Ratio Wings," *Journal of Aircraft*, Vol. 22, March 1985, pp. 193–199.

[4]Borland, C. J. and Sotomayer, W. A., "An Algorithm for Unsteady Transonic Flow About Tapered Wings," AIAA Paper 84-1567, June 1984.

[5]Bennett, R. M., Wynne, E. C., and Mabey, D. G., "Calculation of Transonic Steady and Oscillatory Pressures on a Low Aspect Ratio Model", *Journal of Aircraft*, Vol. 24, June 1987, pp. 392–398.

[6]Guruswamy, G. P., Goorjian, P. M., Ide, H., and Miller, G. D., "Transonic Aeroelastic Analysis of the B-1 Wing," *Journal of Aircraft*, Vol. 23, July 1986, pp. 547–553.

[7]Shankar, V., Ide, H., Gorski, J., and Osher, S., "A Fast Time-Accurate Unsteady Full-Potential Scheme," AIAA Paper 85-1512, July 1985.

[8]Shankar, V. and Ide, H., "Treatment of Steady and Unsteady Flows Using a Fast, Time-Accurate Full Potential Scheme," AIAA Paper 85-4060, Oct. 1985.

[9]Murman, E. M., "Analysis of Embedded Shock Waves Calculated by Relaxation Methods," *Proceedings of AIAA Computational Fluid Dynamics Conference*, AIAA, New York, July 1973, pp. 27–40.

[10]Tijdeman, H. et al., "Transonic Wind Tunnel Tests on an Oscillating Wing with External Stores," AFFDL-TR-78-194, Dec. 1978.