DEPARTMENT OF MECHANICAL ENGINEERING AND MECHANICS
COLLEGE OF ENGINEERING
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA 23508

SIMPLE NUMERICAL METHOD FOR PREDICTING STEADY COMPRESSIBLE FLOWS

By

Ernst von Lavante, Principal Investigator

and

N. Duane Nelson, Research Engineer

Progress Report
For the period ended November 14, 1986

Prepared for the
National Aeronautics and Space Administration
Langley Research Center
Hampton, VA 23665

Under
Research Grant NAG-1-633
Mr. Manuel D. Salas, Technical Monitor
TAD-Theoretical Aerodynamics Branch

February 1986

DEPARTMENT OF MECHANICAL ENGINEERING AND MECHANICS
COLLEGE OF ENGINEERING
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA 23508

SIMPLE NUMERICAL METHOD FOR PREDICTING STEADY COMPRESSIBLE FLOWS

By

Ernst von Lavante, Principal Investigator

and

N. Duane Nelson, Research Engineer

Progress Report
For the period ended November 14, 1986

Prepared for the
National Aeronautics and Space Administration
Langley Research Center
Hampton, VA 23665

Under
Research Grant NAG-1-633
Mr. Manuel D. Salas, Technical Monitor
TAD-Theoretical Aerodynamics Branch

Submitted by the
Old Dominion University Research Foundation
P.O. Box 6369
Norfolk, Virginia 23508

February 1986

SIMPLE NUMERICAL METHOD FOR PREDICTING STEADY COMPRESSIBLE FLOWS

E. von Lavante*
Old Dominion University
Norfolk, VA

and

N. Duane Melson**
NASA Langley Research Center
Hampton, VA

Abstract

A numerical method for solving the isenthalpic form of the governing equations for compressible viscous and inviscid flows was developed. The method was based on the concept of flux vector splitting in its implicit form. The method was tested on several demanding inviscid and viscous configurations. Two different forms of the implicit operator were investigated. The time marching to steady state was accelerated by the implementation of the multigrid procedure. Its various forms very effectively increased the rate of convergence of the present scheme. High quality steady state results were obtained in most of the test cases; these required only short computational times due to the relative efficiency of the basic method.

---

*Associate Professor, Member AIAA

**Research Engineer, Senior Member AIAA

I. Introduction

In recent years, considerable research has been performed towards the goal of computing three-dimensional inviscid and viscous transonic flows about realistic configurations. The complexity of these flowfields makes their numerical prediction very demanding in terms of the capabilities of the numerical method and the computer used for the calculation. Numerical methods for such problems must achieve high rates of convergence while providing results of good quality on reasonably sized computational grids. Computer hardware must have sufficient memory to perform the calculations and be fast enough to provide reasonable turnaround.

Many methods for predicting the inviscid compressible flows about realistic three-dimensional bodies have been developed. Perhaps the oldest is the explicit MacCormack [1] scheme dating back to 1972. Next came the implicit (three-factor ADI) method [2,3] using central difference for the spatial flux derivatives. The explicit, multistage Runge-Kutta method with central differences for the spatial derivatives [4] and multigrid acceleration [5] followed and is the method used widely around the world today.

The newest methods are the implicit schemes with flux-vector-splitting [6-10]. References [6-8] used a full formulation of the Euler equations. References [9] and [10] used an isenthalpic formulation which reduces the three-dimensional problem to a set of four partial differential equations. The energy equation was replaced by an algebraic expression.

The present effort continues the work by von Lavante and uses the isenthalpic assumption in two-dimensions. With the governing equations reduced to three partial differential equations, it is necessary to only solve 3x3 matrices in the block tridiagonal system of equations. This requires

about half as much work as solving the 4x4 block tridiagonal systems if the isenthalpic assumption was not made (9 versus 16 elements).

Jameson has pointed out the importance of conserving total enthalpy when solving the Euler equations (Ref. [11]). In his work, care must be taken to ensure its conservation. In the present method, with the isenthalpic assumption, the conservation of total enthalpy is assured, a priori.

The isenthalpic assumption is not without its drawbacks. First of all, it is limited to steady state calculations since the substantial derivatives of the total enthalpy and pressure are related. However, if the pressure is 'slowly' varying, the isenthalpic equations may be used. Second, for viscous calculations, the maximum freestream Mach number is limited to transonic and moderate supersonic values due to the requirement of no heat sources or sinks. Finally, viscous results can be only considered approximate, since in real flows the total enthalpy changes within the boundary layer. Notwithstanding these limitations, the present scheme worked well and produced good quality results in cases where the flow is steady or slowly varying.

II. The Equation of Motion

As noted previously, there is a large class of problems where only steady state solutions are of interest. For inviscid flows, the assumptions of steady state flow reduces the energy equation to the simple statement that in the absence of heat sources and sinks the total enthalpy will remain constant. The energy equation is therefore replaced by a simple algebraic equation, reducing the number of PDE's to be solved by one. In the case of viscous flows, the above statement is not true. However, it is well known that for the Prandtl Number Pr=1 and adiabatic walls, the total enthalpy will still be constant. Many investigators have applied the enthalpy damping acceleration

technique introduced by Jameson [5] and have effectively driven the total enthalpy to zero at the resulting (hopefully) steady state. Several investigators used this techique to predict very complex two- and three-dimensional configurations and reported results that were in good agreement with experimental data. Because of this experience, the present viscous formulation assumes that the total enthalpy is constant even in the presence of natural dissipation. Due to the obvious limitations of the present formulation, only the thin shear layer form of the viscous terms was implemented. Here the viscous terms in the normal direction are assumed to be much larger than those in the streamwise direction, which is consequently neglected. The two-dimensional Navier-Stokes equations for compressible flows in vector form for general, body fitted coordinates written in nondimensional strong conservation law form using the thin shear layer assumption are

$$\frac{\partial Q}{\partial t} + \frac{\partial F}{\partial \xi} + \frac{\partial (G+G_v)}{\partial \eta} = 0 \qquad (1)$$

where

$$Q = \frac{1}{J}\begin{bmatrix} \rho \\ \rho u \\ \rho v \end{bmatrix} \qquad F = \frac{1}{J}\begin{bmatrix} \rho U_\xi \\ \rho u U_\xi + p\xi_x \\ \rho v U_\xi + p\xi_y \end{bmatrix}$$

$$G = \frac{1}{J}\begin{bmatrix} \rho U_\eta \\ \rho u U_\eta + \rho\eta_x \\ \rho v U_\eta + \rho\eta_y \end{bmatrix} \qquad G_v = \begin{bmatrix} 0 \\ y_\xi \sigma_x - x_\xi \tau_{xy} \\ y_\xi \tau_{xy} - x_\xi \sigma_y \end{bmatrix}$$

Using the definition of the speed of sound at stagnation conditions, $c_o^2 = \gamma R T_o$, the non-dimensional total enthalpy is

$$h_t = \frac{h_o}{c_o^2} = \frac{1}{\gamma - 1} \qquad (2)$$

resulting in the following form of the equation of state

$$p = \frac{\rho}{\gamma} \{1 - \frac{\gamma-1}{2} (u^2 + v^2)\} \tag{3}$$

All variables are nondimensionalized by the stagnation values (for details, see Reference [10]): the primes denoting non-dimensional quantities are dropped for convenience. The viscous terms are shown after the application of the Stokes hypothesis for bulk viscosity. In the above equations, ρ is density, u and v are the cartesian velocity components and p is static pressure. The viscous terms included in G_v will be discussed later. The metric coefficients of the transformation of coordinates are defined as

$$\eta_y = J\, x_\xi \ , \quad \xi_y = -\, J\, x_\eta, \qquad \eta_x = -J\, y_\xi \ , \ \xi_x = J\, y_\eta \tag{4}$$

where J is the Jacobian of the transformation

$$J = 1/(x_\xi y_\eta - x_\eta y_\xi) \tag{5}$$

and U_ξ and U_η are the contravariant velocities

$$U_\xi = u\xi_x + v\xi_y \qquad\qquad U_\eta = u\eta_x + v\eta_y$$

The equivalent inviscid Euler equations are obtained from Eq. (1) by setting $G_v = 0$. The development of the solution algorithms for the Euler equations follows below.

III. Development of Inviscid Algorithm

An implicit Euler single step temporal scheme was selected for advancing the solution of Eq. (1) in time. After linearization in time using the Taylor

series expansions of the flux vectors F and G, and approximate factorization of the implicit operator (details are given in Reference [3]), the basic algorithm has the form

$$[I + \Delta t\, \partial_\xi A^n]\,[I + \Delta t\, \partial_\eta B^n]\,\Delta Q^n = -\Delta t\,(\partial_\xi F^n + \partial_\eta G^n) = R^n \qquad (7)$$

$$Q^{n+1} = Q^n + \Delta Q^n$$

where A and B are the Jacobian matrices

$$A = \frac{\partial F}{\partial Q}, \qquad B = \frac{\partial G}{\partial Q} \qquad (8)$$

The Jacobian matrices A and B are given in detail in Reference [10]. The special discretization of Eq. (7) can be carried out in many different ways. In the present method, the flux vector splitting approach applied to cell centered finite volume formulation was selected. The main reason was its superior ability to capture relatively strong shocks within at most two zones. It can be also shown that its truncation error provides the minimum necessary damping to limit spurious oscillations in the weak solutions to the Euler equations. Based on our previous experience reported in Ref. [10] as well as results presented to Ref. [8], it was decided to use the flux vector splitting introduced by van Leer [13] coupled with the so-called MUSCL type differencing. The van Leer splitting was selected because the split flux vectors are smooth and have smooth first derivatives with respect to the Mach number, so that their eigenvalues are also smooth.

The inviscid flux vectors F and G each have a complete set of three real eigenvectors and can be therefore split into two vectors, one with non-negative eigenvectors and one with non-positive eigenvectors. Following Reference [13], these are

$$F = F^+ + F^-, \qquad G = G^+ + G^- \tag{9}$$

where, for example, $F^+ = (F_1^+, F_2^+, F_3^+)^T$. In more detailed, they are in cartesian coordinates (denoted here by $\hat{}$)

$$\hat{F}_1^+ = \rho\, c_{*,x} \{\tfrac{1}{2}(M_{*,x} + 1)\}^2 \qquad \hat{F}_1^- = \rho\, c_{*,x} \{\tfrac{1}{2}(M_{*,x} - 1)\}^2$$

$$\hat{F}_2^+ = c_{*,x} \tfrac{\gamma+1}{\gamma} \hat{F}_1^+ \qquad \hat{F}_2^- = - c_{*,x} \tfrac{\gamma+1}{\gamma} \hat{F}_1^- \tag{10}$$

$$\hat{F}_3^+ = v\, \hat{F}_1^+ \qquad \hat{F}_3^- = v\, \hat{F}_1^-$$

where $c_{*,x}^2 = \frac{2\beta}{\gamma+1} [\frac{1}{\beta} - \frac{1}{2} v^2]$, $\beta = \gamma - 1$ and $M_{*,x} = u/c_{*,x}$. Similarly,

$$\hat{G}_1^+ = \rho\, c_{*,y} \{\tfrac{1}{2}(M_{*,y} + 1)\}^2 \qquad \hat{G}_1^- = - \rho\, c_{*,y} \{\tfrac{1}{2}(M_{*,y} - 1)\}^2$$

$$\hat{G}_2^+ = u\, \hat{G}_1^+ \qquad \hat{G}_2^- = u\, \hat{G}_1^- \tag{11}$$

$$\hat{G}_3^+ = c_{*,y} \tfrac{\gamma+1}{\gamma} \hat{G}_1^+ \qquad \hat{G}_3^- = - c_{*,y} \tfrac{\gamma+1}{\gamma} \hat{G}_1^-$$

with $c_{*,y}^2 = \frac{2\beta}{\gamma+1} [\frac{1}{\beta} - \frac{1}{2} u^2]$, $M_{*,y} = v/c_{*,y}$.

The above split fluxes $\hat{F}^+$ and $\hat{G}^+$ have, by definition, one zero eigenvalue and two positive eigenvalues; similarly, the fluxes $\hat{F}^-$ and $\hat{G}^-$ have one zero and two negative eigenvalues; these are in the case of F:

$$\lambda_{1,2}^+ = \tfrac{1}{2} c_* (M_*+1) \{\tfrac{2\gamma+1}{2\gamma} \pm \sqrt{(\tfrac{2\gamma+1}{2\gamma})^2 - (M_*+1) [\tfrac{\gamma+1}{2\gamma} + \tfrac{1}{4} (1-M_*)]}\}$$

$$\lambda_3^+ = 0$$

$$\lambda_{1,2}^- = \frac{1}{2} c_* (M_*-1) \left\{\frac{2\gamma-1}{2\gamma} \pm \sqrt{\left(\frac{2\gamma+1}{2\gamma}\right)^2 + (M_*-1)\left[\frac{\gamma+1}{2\gamma} + \frac{1}{4}(1+M_*)\right]}\right\} \quad (12)$$

$$\lambda_3^- = 0$$

where c_* and M_* are either $c_{*,x}$ and $M_{*,x}$ or $c_{*,y}$ and $M_{*,y}$.

These split fluxes are, unfortunately, formulated in cartesian coordinates only. They have to be transformed into general coordinates ξ and η, which is accomplished by simply rotating the local coordinates at a given point in the flowfield to a direction parallel with one of the covariant vectors $\vec{\xi}$ and $\vec{\eta}$. This procedure is described in some detail in Reference [8]; the resulting transformation is

$$F^+ = T_\xi^{-1} \hat{\bar{F}}^+$$

where $\hat{\bar{F}} = \hat{F}^+ (\bar{Q})$ (13)

and

$$T_\xi^{-1} = \begin{bmatrix} \sqrt{x_\eta^2 + y_2^2} & 0 & 0 \\ 0 & y_\eta & x_\eta \\ 0 & -x_\eta & y_\eta \end{bmatrix}$$

The new dependent variable vector $\bar{Q}$ is obtained from Q by replacing the cartesian velocity components u and v by the physical velocity components $\bar{u}$ and $\bar{v}$ in the covariant direction $\vec{\xi}$. These are, respectively

$$\bar{u} = (y_\eta u - x_\eta v)/\sqrt{x_\eta^2 + y_\eta^2}$$

$$\bar{v} = (x_\eta u + y_\eta v)/\sqrt{x_\eta^2 + y_\eta^2}$$

Knowing the eigenvalues of the split fluxes, it is now obvious that in the special differences in Eq. (7) F^+ and G^+ have to be backward differenced and F^- and G^- have to be forward differenced. This is accomplished by the application of the MUSCL type differencing, described in more detail in References [10, 8]. Here, instead of using the traditional backward or forward finite differences operating on F^+, G^+, F^- and G^-, the dependent variables Q, which are better differentiable than the flux vectors, are extrapolated to the cell faces in positive or negative direction, depending on the sign of the eigenvalues. The RHS of Eq. (7) becomes

$$R^n = -\Delta t \,(F^+_{i+1/2,j} - F^+_{i-1/2,j} + F^-_{i+1/2,j} - F^-_{i-1/2,j} + G^+_{i,j+1/2} - G^+_{i,j-1/2} + G^-_{i,j+1/2} - G^-_{i,j-1/2}) \qquad (14)$$

where, for example,

$$F^+_{i+1/2,j} = F^+(Q^+_{i+1/2,j}) \quad ; \quad F^-_{i+1/2,j} = F^-(Q^-_{i+1/2,j})$$

$$G^+_{i,j+1/2} = G^+(Q^+_{i,j+1/2}) \quad ; \quad G^-_{i,j+1/2} = G^-(Q^-_{i,j+1/2})$$

and

$$Q^-_{i+1/2,j} = Q_{i+1,j} - k_s \frac{1}{2}(Q_{i+2,j} - Q_{i+1,j}),$$

$$Q^+_{i+1/2,j} = Q_{i,j} + k_s \frac{1}{2}(Q_{i,j} - Q_{i-1,j})$$

etc., with similar expressions in the j-direction.

The parameter k_s switches between first order formulation ($k_s = 0$) and second order formulation ($k_s = 1$).

The present formulation, when applied to transonic and low supersonic

flows, did not require the use of flux limiters for essentially oscillation free shocks. This was noticed by Anderson, Thomas and van Leer [8] and von Lavante and Haertl [10] and was explained in more detail by van Leer [14]. The favorable behavour of the present formulation is due to the fact that at transonic speeds, the backward running characteristic variable that is being extrapolated from downstream of the shock is much smaller than the forward running characteristic variable. Despite the above linear extrapolation, no or very small overshoots were encountered.

The implicit left hand side of Eq. (7) underwent similar modifications as the right hand side. The Jacobian matrices A and B were replaced by the corresponding Jacobians of the split flux vectors, yielding following form of the left hand side of Eq. (7).

$$[I + \partial_\xi^b A^+ + \partial_\xi^f A^-] [I + \partial_\eta^b B^+ + \partial_\eta^f B^-] \Delta Q^n = R^n \qquad (15)$$

where $A^+ = \frac{\partial F^+}{\partial Q}$, $A^- = \frac{\partial F^-}{\partial Q}$, $B^+ = \frac{\partial F^+}{\partial Q}$, $B^- = \frac{\partial F^-}{\partial Q}$ and ∂_ξ^b and ∂_ξ^f are first order backward and forward differences, respectively. The exact form of these Jacobian matrices will be given in the full paper. A standard block tridiagonal solver was used to solve the system of algebraic equations given by Eq. (15).

An acceleration of the convergence to steady state conditions was achieved by the use of local time steps. In this procedure, the time step used in each of the cells was determined from the maximum local eigenvalue after each iteration. The two-factor, block tridiagonal form of the resulting algorithm, given by Eqs. (15) and (14) is relatively easy to vectorize.

The Eq. (7) is not the only possible Euler implicit form. The factorization of the left hand side can be carried out in many different ways.

One alternative factorization that seemed to yield more efficient method for execution on scalar computers is a four-factor form

$$[I + \partial_\xi^b A^+] [I + \partial_\xi^f A^-][I + \partial_\eta^b B^+] [I + \partial_\eta^f B^-] \Delta Q^n = R^n \qquad (16)$$

This form of the implicit operator results in block-bidiagonal L-U systems of equations that can be solved very efficiently, each of them in one sweep. The implicit operator based on Eq. (16) was therefore also tested in the present investigation. The directions of sweep were permutated in order to preserve the symmetry of the algorithm. However, the L-U scheme was only marginally better than the block-tridiagonal scheme, since its maximum stable CFL number was lower, probabily due to its larger splitting error. Attention was therefore focused on the scheme based on Eq. (15), since, in addition to being slightly more efficient, it also makes the inclusion of central difference viscous terms much easier than the L-U scheme.

IV. Development of Viscous Algorithm

The simplicity of the present viscous scheme is striking. This is mainly due to the assumption of constant total enthalpy and application of the thin shear layer form of the Navier-Stokes equations. Equation (10) already indicated that only two lines have to be added to the flux vector G, in addition to the necessary changes in the implicit operator.

After some simple manipulations, G_v can be rewritten as

$$G_v = \mu J \begin{bmatrix} 0 \\ -(\frac{4}{3} y_\xi^2 - x_\xi^2)\, u_\xi + \frac{1}{3} x_\xi y_\xi v_\eta \\ \frac{1}{3} x_\xi y_\xi u_\eta - (\frac{4}{3} x_\xi^2 + y_\xi^2)\, v_\eta \end{bmatrix} \qquad (17)$$

It should be noted that in this case this is exactly the same form as obtained by Swanson and Turkel [12] after evaluating the first order derivatives in the viscous terms using Green's theorem and then using following center difference discretization at the cell face at j+1/2:

$$u_\eta = u_{i,j+1} - u_{i,j}; \qquad v_\eta = v_{i,j+1} - v_{i,j}$$

$$x_\xi = x_{i,j} - x_{i-1,j}; \qquad y_\xi = y_{i,j} - y_{i-1,j} \tag{18}$$

$$\mu = \frac{1}{2}\left(\mu_{i,j} + \mu_{i,j+1}\right) \qquad J = \frac{2}{1/J_{i,j+1} + 1/J_{i,j}}$$

In the case of the implicit part, the full viscous flux Jacobian matrix is relatively complicated and time consuming to evaluate. However, the basic underlying assumption of the validity of the present method was that only steady state results were required. From the delta form of the governing equations, given in Eqs. (17), (15) and (16), it is obvious that if the method converges to steady state, the steady state results will not be effected by the implicit operator. The dissipative part of the implicit operator can be therefore simplified, as long as the stability limits of the scheme are not reduced.

It has been shown previously (References [15, 16] that the implicit part of the Navier-Stokes solver (15) with (17) can be significantly simplified by substituting the correct Jacobian matrix of the viscous flux by a diagonal matrix $I\nu$, where I is the unity matrix and ν is the maximum diagonal component of the matrix M that is obtained from the thin shear layer Navier-Stokes equations in the following delta form

$$[I + \partial_\xi A]\,[I + \partial_\eta B + \partial_\eta M\,\partial_\eta]\,\Delta Q^n = R^n \qquad (19)$$

$$\nu = \frac{1}{\rho}\frac{4}{3}\,\mu/(x_\eta^2 + y_\eta^2)$$

The final form of the present viscous scheme is given by

$$[I + \partial_\xi^b A^+ + \partial_\xi^f A^-]\,[I + \partial_\eta^b B^+ + \partial_\eta^f B^- + \partial_\eta\, I\nu\,\partial_\eta]\,\Delta Q^n = R^n \qquad (20)$$

where, for example, $G^+_{i,j+1/2} \rightarrow G^+_{i,j+1/2} + G_{v;i,j+1/2}$ in order to account for the viscous effects in the explicit part R^n. The viscous terms are centrally differenced in both the explicit and implicit parts of Eq. (20).

V. Multigrid

The Full Approximation Storage (FAS) multigrid scheme (Ref. [1]) must be used since the set of equations are nonlinear. A development of the FAS scheme is given below. Consider the problem

$$L^h u^h = f^h \qquad (21)$$

where L^h is a nonlinear operator on a grid, G^h, with spacing h. The forcing function, f is known and U^h is the solution to the problem on the grid with spacing h. If we take u^h as an approximation to U^h with an error of

$$v^h = U^h - u^h,$$

Eq. (21) can be written as

$$L^h (u^h + v^h) = f^h \qquad (22)$$

$L^h u^h$ is subtracted from both sides of (18) give:

$$L^h (u^h + v^h) - L^h (u^h) = f^h - L^h (u^h) \qquad (23)$$

If the terms are smooth, they can be represented on a coarser grid, G^{2h}, with spacing 2h. The grid G^{2h} is formed by deleting every other point in

G^h. Therefore, G^{2h} G^h. Points are eliminated from G^{2h} to form G^{4h} and so forth to form G^{8h}, G^{16h}, etc. Written on the coarse grid G^{2h}, Eq. (23) becomes

$$L^{2h}\,(I_h^{2h}\,u^h + V^{2h}) - L^{2h}\,(I_h^{2h}\,u^h) = I_n^{2h}\,(f^h - L^h\,u^h) \tag{24}$$

or

$$L^{2h}\,(u^{2h}) = f^{2h} \tag{25}$$

where

$$f^{2h} = I_n^{2h}\,(f^h - L^h\,u^h) + L^{2h}\,(I_h^{2h}\,u^h)$$

and I_n^{2h} is the restriction operator.

Since Eq. (25) is on a coarser grid than Eq. (22), the numerical solution for u^{2h} is much cheaper to obtain because fewer points are involved. Note that the operator used on the coarse grid has the same form as the fine grid operator, the grid spacing (h and 2h) being the only difference. Once the values of u^{2h} are obtained, the fine grid iterative solution is updated using the following equation:

$$(u^h)_{New} = (u^h)_{Old} + I_{2h}^h\,[u^{2h} - I_n^{2h}\,(u^h)_{Old}] \tag{26}$$

where I_{2h}^h is the prolongation operator.

The restriction operator has two forms. One form is used to restrict the dependent variables, $L_n^{2h}\,(u^h)$; i.e. the flow quantities ρ, ρu, and ρv. For these, the volume weighted average of the values of the function at midcells of the four fine grid cells contained in a course grid cell are used to set the value of the coarse grid (See Fig. 1). The other form of the restriction operator is for the restriction of residuals, $I_h^{2h}\,[L^h\,(u^h)]$. A

simple summation of the residuals over the four fine grid cells composing the coarse grid cell is performed.

The restriction operations are performed for all interior points of the flow field. At the outer boundaries, only the values of the functions are restricted, with no residual restriction. These values are frozen to the fine grid values and are not updated on the coarse grids since a lift-correction scheme is used to set the outer-boundary values on the fine grid. The lift-correction scheme was found to be less accurate on the coarse grids and tended to slow the convergence. At the airfoil surface, the values are not frozen and the same boundary condition was used for all the grids. At the wake cut, flow values at ghost points were set equal to the flow values from the proper points across the wake on all the grids.

The prolongation operation used in the current work is a bilinear interpolation in the computation space of the corrections at the four coarse grid cells adjacent to the fine grid midcell (see Figure 2). Volume weighting is not used.

As a first cut, a fixed V-cycle with four grids was used; a fine grid with 209x33 cells and three coarse grids with 105x17, 53x9 and 27x5 cells, respectively. The program was constructed to allow the number of iterations on each grid between restrictions and prolongation to be controlled by input. Either first-order (k_s=0) or second-order (k_s=1) approximations can be used, in any combination, on each of the grids. Local time stepping was employed on each grid with a single reference CFL number controlling all grids.

Before discussing multigrid results, it is necessary to define work units. Conceptually, a work unit is the amount of work required to perform one fine grid (G^h) iteration. It follows that the work required to perform an

iteration on the grid G^{2h} is 1/4 for two-dimensional problems. For grid G^{4h} the work per iteration is 1/16 and for grid G^{8h} the work is 1/64. To be honest, the work required to restrict from one grid to the next must be included since a residual calculation is necessary on both the fine and course grids. On the conservative side, this can be estimated as the sum of the work to perform a fine grid iteration and a coarse grid iteraction, 1.25 for the restriction from G^h to G^{2h}. (The inclusion of the work required for the grid transfer is less important if the residual calculation is a small part of the update calculation, such as for highly implicit schemes. It is more important for explicit schemes.) The work writes for the present multigrid computations, expressed as multiples of the single (fine) grid iteration, are therefore obtained from

$$wu = n_h + 1.25 + n_{2h} \quad \frac{1}{4} + 0.3125 + n_{4h} \quad \frac{1}{16} + (\frac{1}{16} + \frac{1}{64}) + n_{8h} \quad \frac{1}{64}$$

where $n_h \ldots n_{8h}$ are the number of iterations on each grid. The work unit count was increased in the case of additional iterations during the prolongation process. A more precise accounting for the grid transfers would involve timing the transfer and then calculating the work required based on the time required. In the present study, the conservative method of adding the work required to perform fine and a coarse grid iteration is used. This produces work which is high by about 15%.

VI. Results

The present method was first tested in the single grid version on several transonic flow configurations that included internal as well as external cases. Only the more interesting test cases will be shown here.

Inviscid results - The performance of the inviscid (Euler) scheme can be

demonstrated on the several standard transonic airfoil test cases.

In the present work, the NACA 0012 airfoil at various Mach numbers and angles of attack was selected. The flow was predicted using a 201x31 C-grid that was generated using elliptic grid generation. There were 141 points on the airfoil. Following three flow conditions are frequently encountered in literature and will be therefore discussed here:

a) $M_\infty = 0.8$, α (angle of attack) = 0 - this supercritical case is particularly well suited to test the ability of a numerical method to perserve the symmetry of the flow. The lift coefficient C_l should be zero, while the drag coefficient C_d will be nonzero due to the shock. The present method predicted the C_l to be 1.41×10^{-8}, a value that is certainly acceptably close to zero. The drag correspond to $C_d = 0.0087$, which is in very good resulting Mach number and pressure contours are shown in Fig. 3. The shocks were captured within at most two zones and are very crisp; the Mach contours are very smooth indicating the absence of spurious oscillations. The convergence history is shown in Fig. 4 for the optimum CFL number of 27.5. At this CFL number, the spectral radius of the convergence ρ_{sp} was approximately 0.969, which is low for a single grid calculation. The residuals shown in Fig. 4 are the L-2 norm of the density residual (marked with crosses) and the maximum residual of the density. The correct number of supersonic points was reached after only 100 iterations (Fig 4).

b) $M_\infty = 0.8$, $\alpha = 1.25^0$ - another supercritical case; it is well suited for testing the performance of the boundary conditions, since the lift is very sensitive to their influence. The results obtained from the present method were $C_l = 0.3617$ and $C_d = 0.0233$. These results are in good agreement with data published in Reference [17]. The range of best results was given as C_l =

0.3632 - 0.3661 and C_d = 0.0229 - 0.0230, achieved on grids that extended up to 96 cords from the airfoil!! The comparison with results published by Anderson et al. was also favorable; they reported C_l = 0.363 and C_d = 0.0234. The corresponding Mach numbe and pressure contours are shown in Fig. 5. The shock on the upper surface was again very well captured. The convergence history for this case is presented in Fig. 6. Although not as good as in the $\alpha = 0$ case, the spectral radius was still a reasonable $\rho_{sp} = 0.975$ at the optimum CFL number of 21. The correct number of supersonic points was obtained after 189 iterations.

c) $M_\infty = 0.63$, $\alpha = 2^0$ - this case is subcritical. Here, the main difficulty lies, besides the correct lift prediction, in the drag calculation. In the absence of shocks, the C_d should be zero. The present scheme computed C_l = 0.3302 and C_d = 0.0006. Both values are in reasonable agreement with results reported by Anderson, Thomas and van Leer [8], given as C_l = 0.332 and C_d = 0.0006. The Mach number and pressure contours can be seen in Fig. 7. The residual history in Fig. 8 is somewhat surprising. Although the Mach number is lower than in the previous cases, the rate of convergence is by far the best. This is reflected in the value of the spectral radius $\rho_{sp} = 0.964$, obtained at an optimum CFL number of 18. The flow is subcritical with no supersonic points.

Viscous results - The present simplified viscous algorithm was tested on several configurations. The consistancy of the method was first investigated by grid refinement study done on compressible subsonic and supersonic boundary layer flow on a flat plate. The freestream Mach number was 0.5 and the reference Reynolds number was Re_o = 5,000. The computations were carried out on three grids: 51x51, 51x76 and 51x101 that were exponentially stretched in the direction normal to the plate. The grid was refined in the direction

normal to the plate. The results will be presented in the full version of the paper; they can be summarized by stating that the skin friction coefficients as well as the velocity profiles consistantly improved as the grid was refined and agreed closely with the Blasius solution.

The viscous method was subseqently tested on a supersonic diffusor configuration with inflow Mach number M=2.0 and upper wall compression corner that generated on oblique shock. The shock was reflected off the lower wall at a Reynolds number $Re_x = 3\ 10^5$ based on the length along the lower wall, and was strong enough to cause separation of the boundary layer. The configuration is similar to tht used by Thomas and Walters [20]. Computations using the present method were done on two grids: 51x51 (coarse) and 51x10 (finer). The resulting ratio of static pressure to total pressure as well as skin friction coefficient C_f at the lower are shown in Fig. 9 as compared with experimental data given by Hakkinen et al. [19]. It can be seen that the coarse grid had insufficient resolution to correctly predict the extend of the separated zone and the formation of the pressure plateau typical for lambda shock structure. Refining the grid resulted in much better prediction of the pressure and C_f; except for the reattachment region, the agreement with experimental results is quite good. This tendency is consistant with results given in Ref. [20]; minor differences in the pressure profile can be explained by the presence of the upper wall in present configuration. The rate of convergence, given in Fig. 10, was rather high. At an maximum local CFL number of 1450 (global time steps were used in this case) the maximum residual was reduced eight orders of magnitude within 500 iterations.

More recently, results were obtained for laminar flows over transonic airfoil NACA 0012, showin separated flow near the trailing edge. These will be included in the full version of this paper.

Multigrid - All the multigrid computations were carried out on a 32 bit computer, as compared to the single grid cases shows above, which were executed on a 60 bit machine. To show the acceleration of the multigrid scheme, results for the test case of a NACA0012 at $M_\infty = .80$, $\alpha = 1.25$ are presented. In Figure 11, is the convergence history of a baseline, no multigrid calculation at optimum CFL number. This calculation was performed on a 201x31 cell grid, somewhat coarser than the 209x33 cell grid used for the multigrid calculations. One hundred and sixty-five iterations were required to converge the lift coefficient to within 1% of the converged solution and over 250 iterations were required to reduce the maximum residual three orders of magnitude. (see case 1, Table 2). Figure 3 shows the logarithm of the maximum residual divided by the initial residual, the lift coefficient, and the number of supersonic points are plotted versus number of iterations.

The baseline calculation was then accelerated using multigrid with four grids in a V-cycle. As a first cut, two iterations were performed on each of the grids for 40 cycles. The convergence history of this case (case 2 of Table 10 is shown in Figure 12. The lift converged (within 1%) in 107 work units. In case 3, more iterations were performed on the coarser grids and the lift was obtained in 88 work units. The residual also converged more quickly (Figure 13).

Other researchers have successfully used multigrid without performing iteractions between the prolongation operations (Ref. 5, 8). This was tried in case 4 (Figure 14) and was found to reduce the time needed to achieve the lift coefficient (83 work units) but produced a slower reduction of the maximum residual.

Since the computer program used in the presen work allows a choice

between first and second-order differencing, a scheme with second-order differencing on the fine grid and first-order differencing on the coarse grids was tried. It was hoped the first-order differencing would provide better smoothing on the coarse grids. This was not the case. (See Figure 15). The lift required 105 work units to achieve a level within 1% of the converged answer.

It should be noted that same investigation (Ref. 17) define work units simply as cycles. Using this definition of convergence and acceleration, the present method converged in 9-30 cycles, starting from uniform flow. This corresponds to an acceleration ratio of 10-12.

To date, the optimum V-cycle FAS multigrid strategy found with this method of solving the 2-D isenthalpic Euler equations is to use second-order differencing or all the grids and to perform more iterations on the coarser grids than on the fine grid. To most quickly obtain the lift, no iterations should be performed between prolongations. It was found that the optimum CFL number with the multigrid acceleration was close to the optimum for the single grid calculation.

The present scheme was also used to predict flows about other AGARD airfoil configurations at following condition: $M_\infty = 0.85$, $\alpha = 1^0$; $M_\infty = 1.2$, $\alpha = 0^0$ and $M_\infty = 1.2$, $\alpha = 1^0$. The full version of this paper will also include a study of inviscid separation on a backward facing step, carried out with the present scheme.

The present algorithm is currently being extended to three dimensions. Some three-dimensional results will be also included in this paper.

Future Work

The acceleration obtained by increasing the number of iterations on the coarse grids suggested that a W-cycle may yield benefits over the V-cycle currently used. This will be investigated. To date, a simple reference CFL number from grid to grid should be explored.

Acknowledgement

This research was supported by NASA Langley Research Center Grant No. NAG-1-633. The authors would like to thank Mr. M. Salas for his help.

References

1. MacCormack, R. W., "Computational Efficiency Achieved by Time Splitting of Finite Difference Operators," AIAA Paper 72-154.

2. Briley, W. R. and McDonald, H., "Solution of the Three-Dimensional Compressible Navier-Stokes Equations by an Implicit Technique", Proceedings of the 4th International Conference on Numerical Methods in Fluid Dynamics, 1974.

3. Beam, R. M. and Warming, R. F., "An Implicit Factored Scheme for the Compressible Navier-Stokes Equations," AIAA Journal, Vol. 16, April 1978, pp. 393-402.

4. Jameson, A., Schmidt, W. and Turkel, E., "Numerical Solutions of the Euler Equations by Finite-Volume Methods Using Runge-Kutta Time-Stepping Schemes," AIAA Paper 81-1259, June 1981.

5. Jameson, A., "Solution of the Euler Equations for Two-Dimensional Transonic Flow by a Multigrid Method," Princeton University, MAE Report No. 1613, June 1983.

6. Chakravarthy, S. R., "Relaxation Methods for Unfactored Implicit Schemes," AIAA 84-0165, Jan. 1984.

7. Van Leer, B. and Mulder, W. A., "Relaxation Methods for Hyperbolic Equations," Report 84-20, Delft University of Technology, 1984.

8. Anderson, W. K., Thomas, J. L. and Van Leer, B., "A Comparison of Finite Volume Flux Vector Splittings for the Euler Equations," AIAA 85-0122, Jan. 1985.

9. von Lavante, E. and Trevino, J., "Numerical Predictions of Internal Flows Using a Diagonalized Flux Vector Splitting Algorithm," AIAA Paper 84-1246.

10. von Lavante, E. and Haertl, A., "Numerical Solutions of Euler Equations Using Simplified Flux Vector Splitting," AIAA paper 85-1333.

11. Jameson, A., Private Communication.

12. Swanson, R. C. and Turkel, E., "A Multistage Time-Stepping Scheme for the Navier-Stokes Equations," AIAA Paper 85-0035.

13. van Leer, B., "Flux-Vector Splitting for the Euler Equations," ICASE Report No. 82-30, September 1982.

14. van Leer, private communication.

15. Coakley, T. J., "Implicit Upwind Methods for the Compressible Navier-Stokes Equations," AIAA Paper 83-1958.

16. von Lavante, E. and Iyer, V. S. V., "Simplified Implicit Block-Bidiagonal Finite Difference Method for Solving the Navier-Stokes Equations," AIAA Journal, Vol. 23, No. 7, July 1985, pp. 1130-1132.

17. Jameson, A. and Yoon, S., "Multigrid Solution fo the Euler Equations Using Implicit Schemes," AIAA Paper 85-0293.

18. "Test Cases for Inviscid Flow Field Methods," AGARD Report AR-211, May 1985.

19. Hakkinen, R. J., Greber, I., Trilling, L. and Abarbanel, S. S., "The Interaction of an Oblique Shock Wave with a Laminar Boundary Layer," NASA Memo-2-18-59W, March 1959.

20. Thomas, J. L. and Walters, R. W., "Upwind Relaxation Algorithms for the Navier-Stokes Equations," AIAA Paper 85-1501.

Table 1 - Summary of Airfoil Results

Case	$M_\infty = 0.8, \alpha = 0$	$M_\infty = 0.8, \alpha = 1.25^0$	$M_\infty = 0.63, \alpha = 2^0$
C_ℓ	$1.41 \cdot 10^{-8}$	0.3617	0.3302
C_d	0.0087	0.0233	0.0006
Optimum CFL	27.5	21	18
ρ_{sp}	0.969	0.975	0.964

Table 2 - Summary of Results, Multigrid

Case	Convergence History in Figure	No. of Grids	Iterations on Each Grid	KS on Each Grid	Work to Reduce R_{max} 10^{-3}	Work to Get CL (1%)
1	11	1*	1	1	252	165
2	12	4	2-2-2-2-2-2-2	1,1,1,1	162	107
3	13	4	2-4-6-8-6-4-2	1,1,1,1	120	88
4	14	4	2-4-6-8-0-0-0	1,1,1,1	160	83
5	15	4	2-4-6-8-6-4-2	1,0,0,0	156	105

*201x31 cells

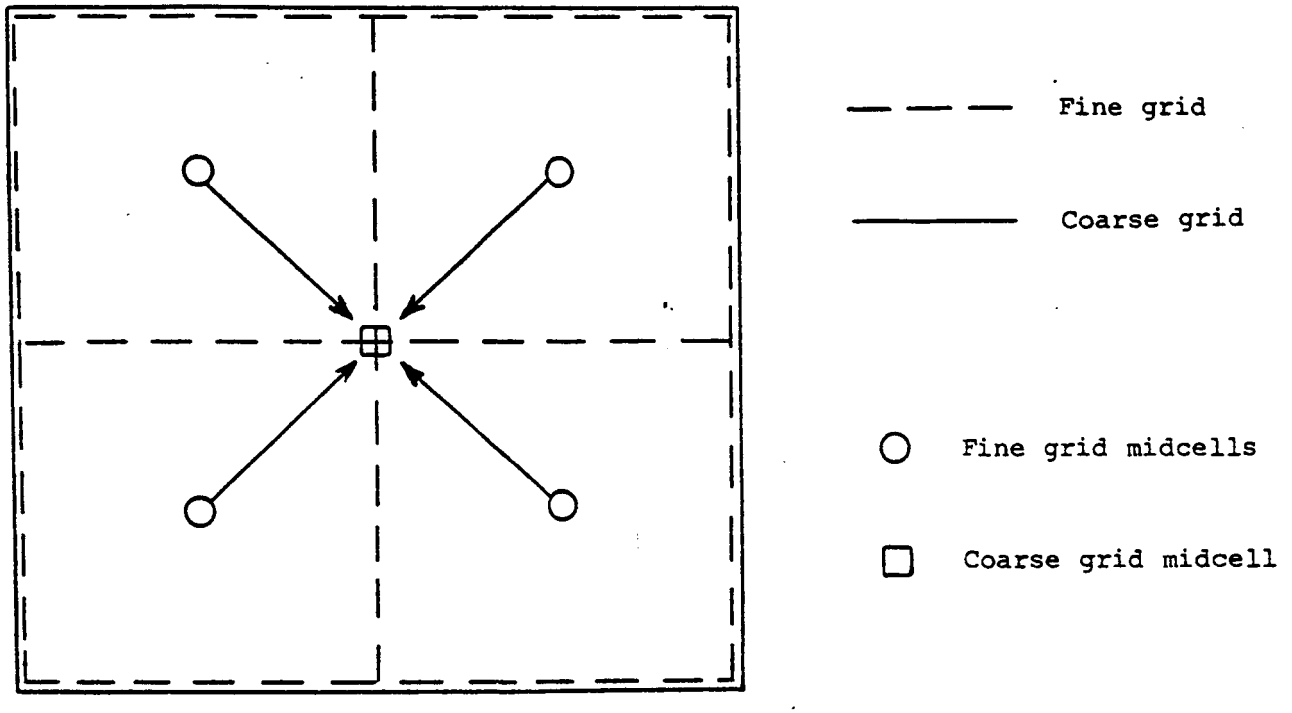


Figure 1.- Schematic of fine and coarse grid cells used for restriction.

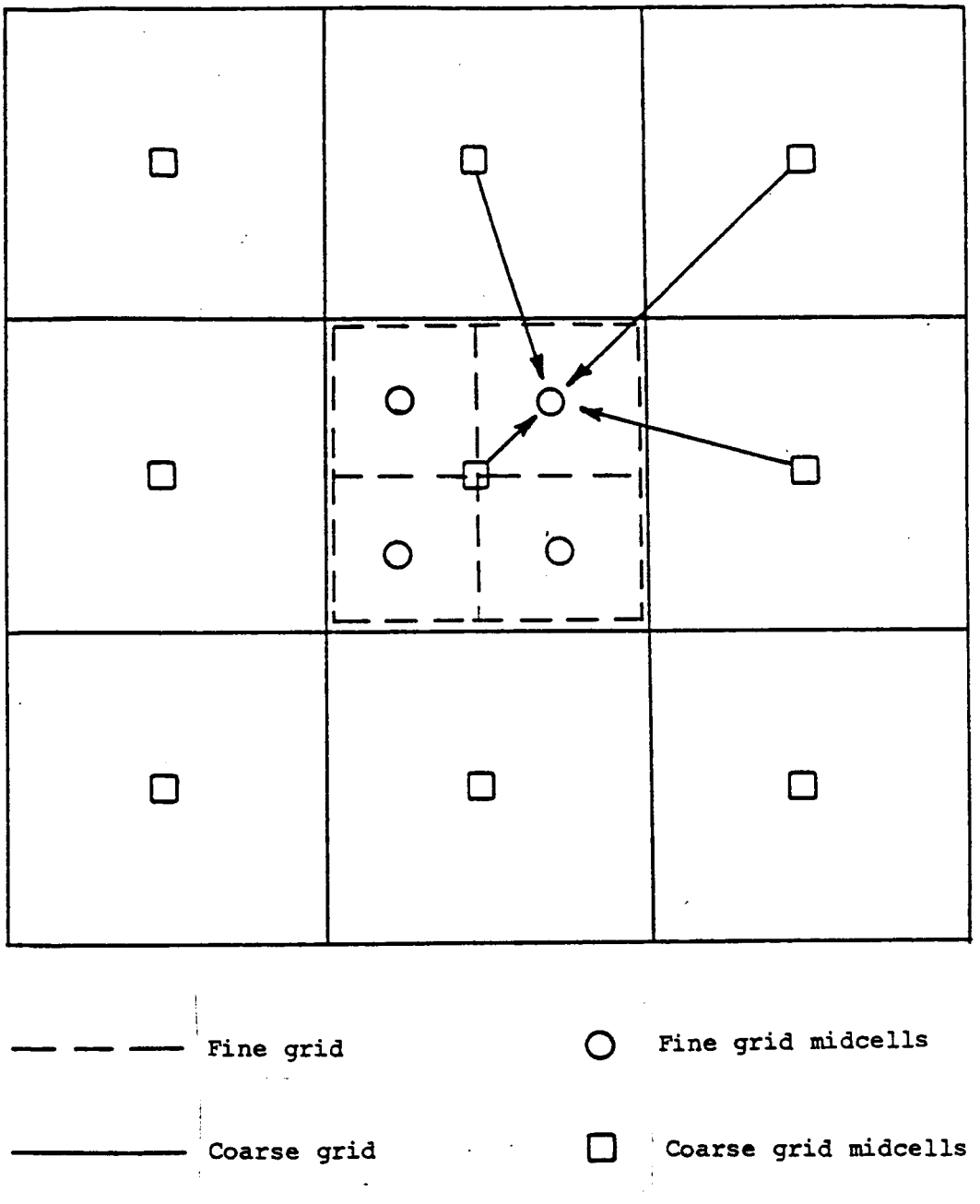
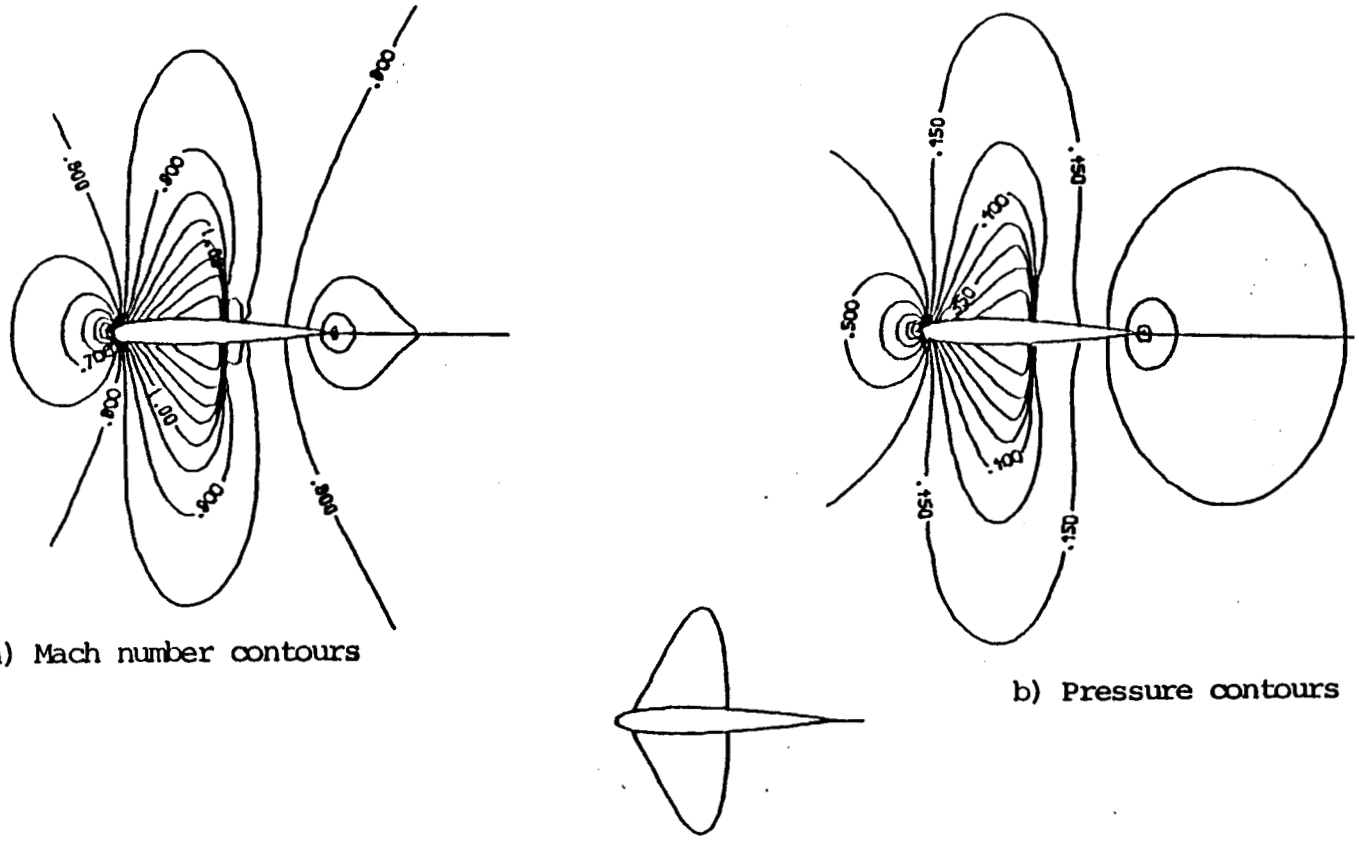


Figure 2.- Schematic of fine and coarse grid cells used for prolongation.



a) Mach number contours

b) Pressure contours

c) Sonic lines

Fig. 3 - Mach number contours, pressure contours and sonic lines for NACA 0012 airfoil at M_∞ =0.8, zero angle of attack.

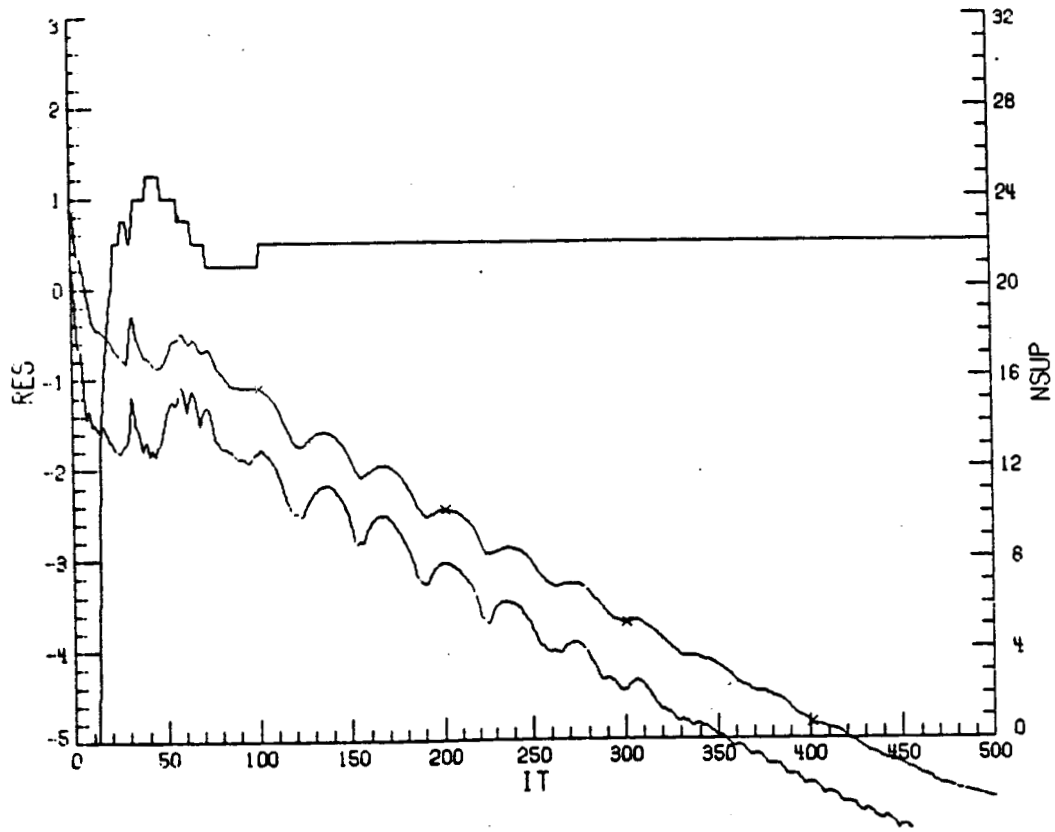
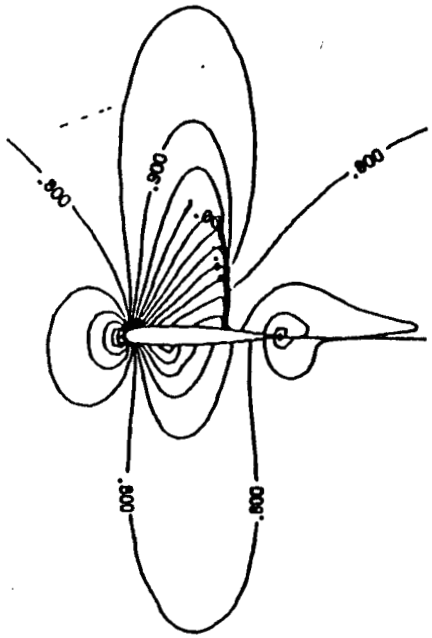
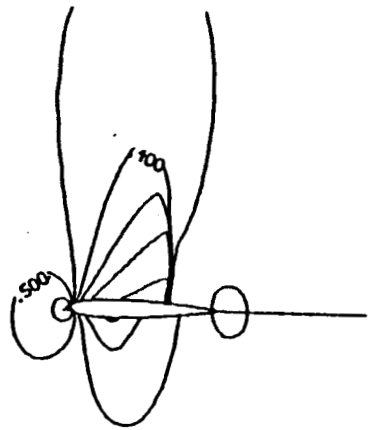


Fig. 4 – Convergence history and number of supersonic points for NACA 0012 airfoil at $M_\infty = 0.8$, zero angle of attack.



a) Mach contours



b) Pressure contours

c) Sonic lines

Fig. 5 - Mach number contours, pressure contours and sonic lines for NACA 0012 airfoil at $M_\infty = 0.8$, angle of attack = 1.25°.

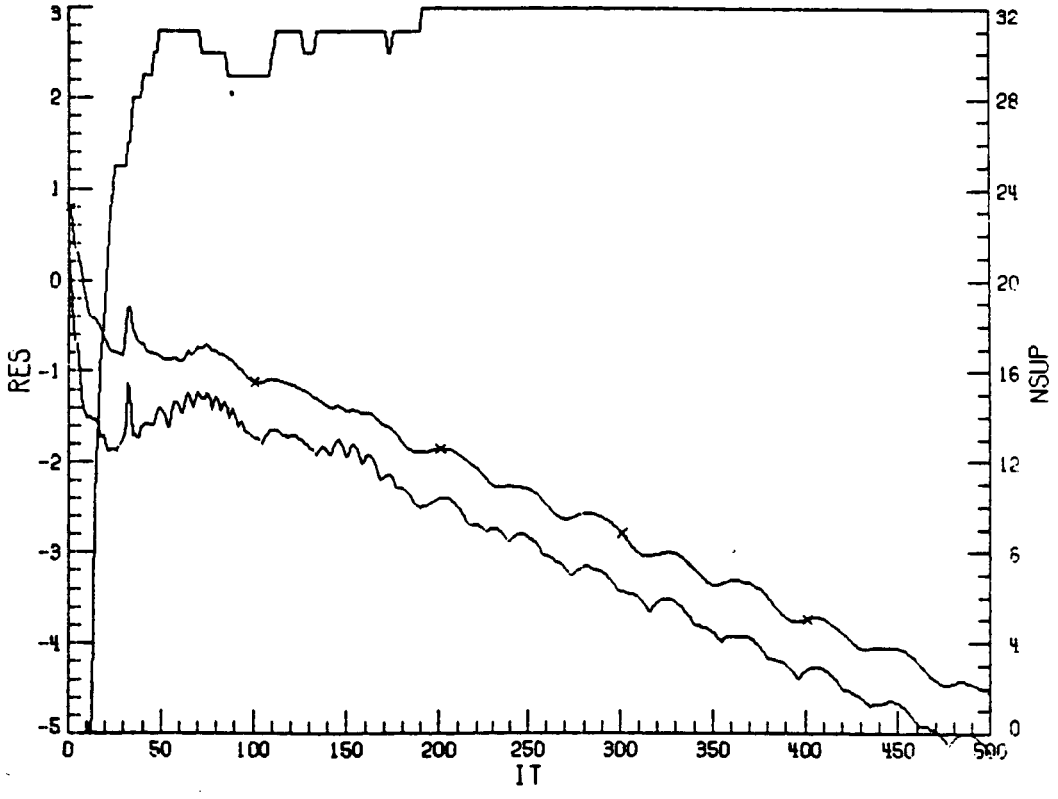
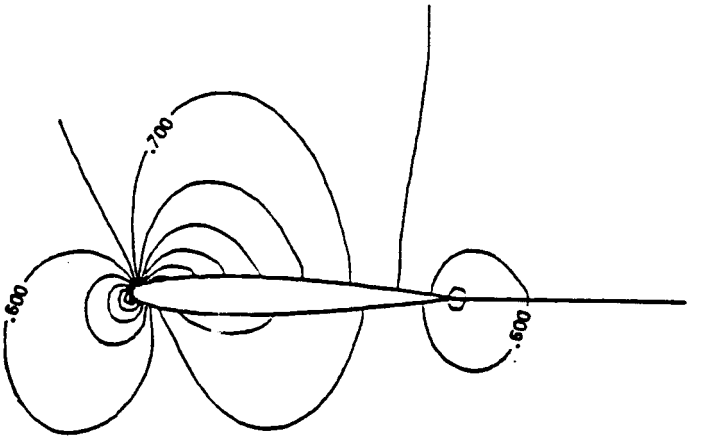
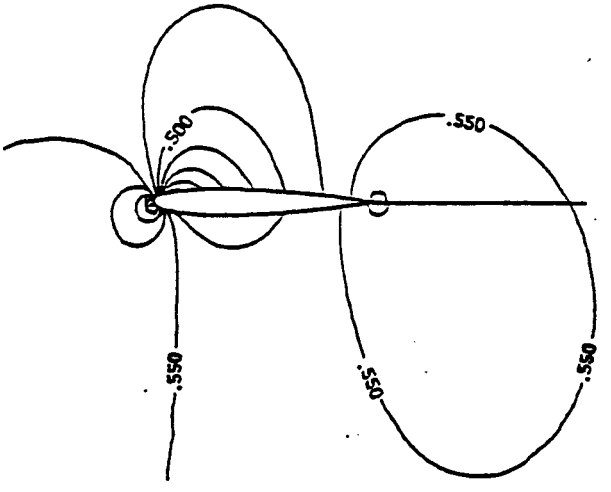


Fig. 6 - Convergence history and number of supersonic points for NACA 0012 airfoil at $M_\infty = 0.8$, angle of attack = 1.25°.



a) Mach number contours



b) Pressure contours

Fig. 7 - Mach number and pressure contours for NACA 0012 airfoil at $M_\infty = 0.63$, angle of attack = 2.0°.

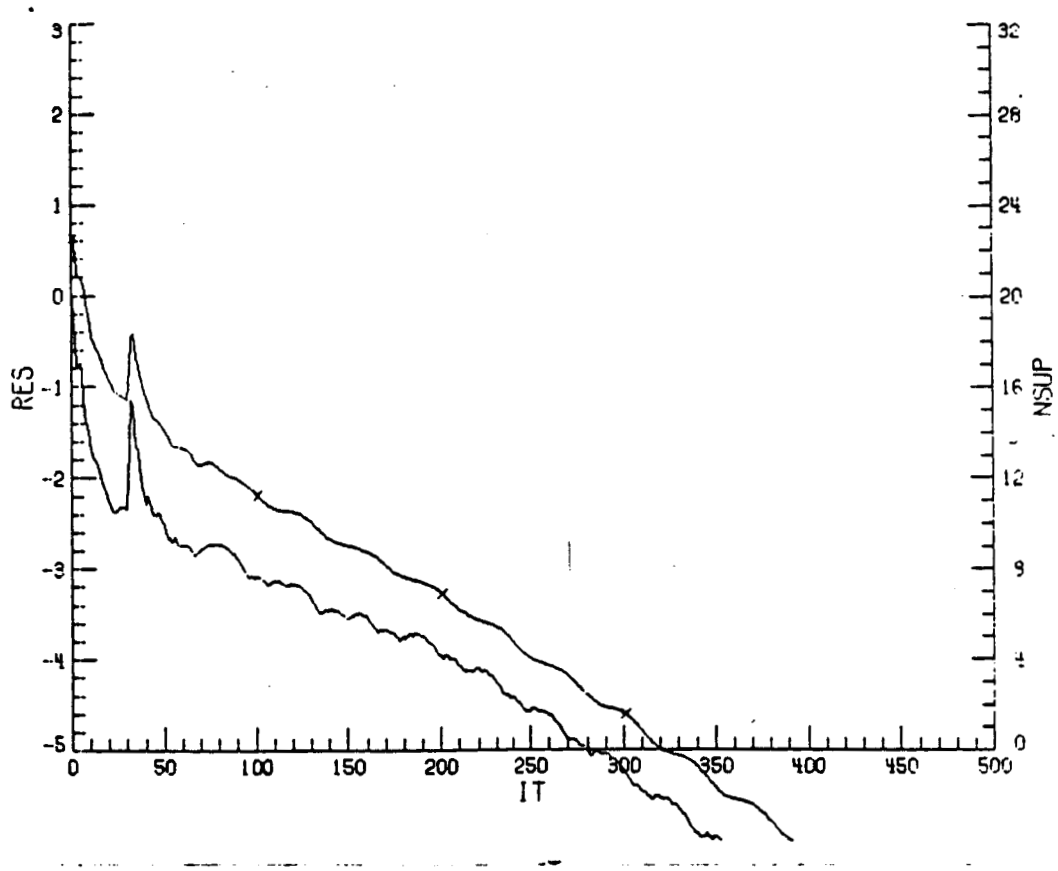
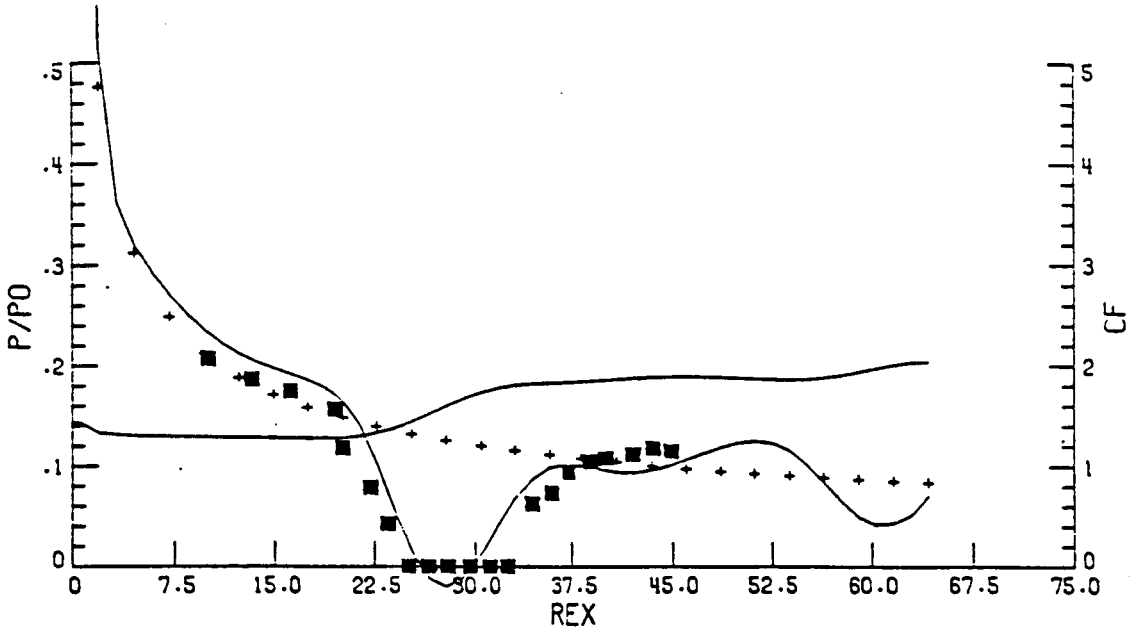
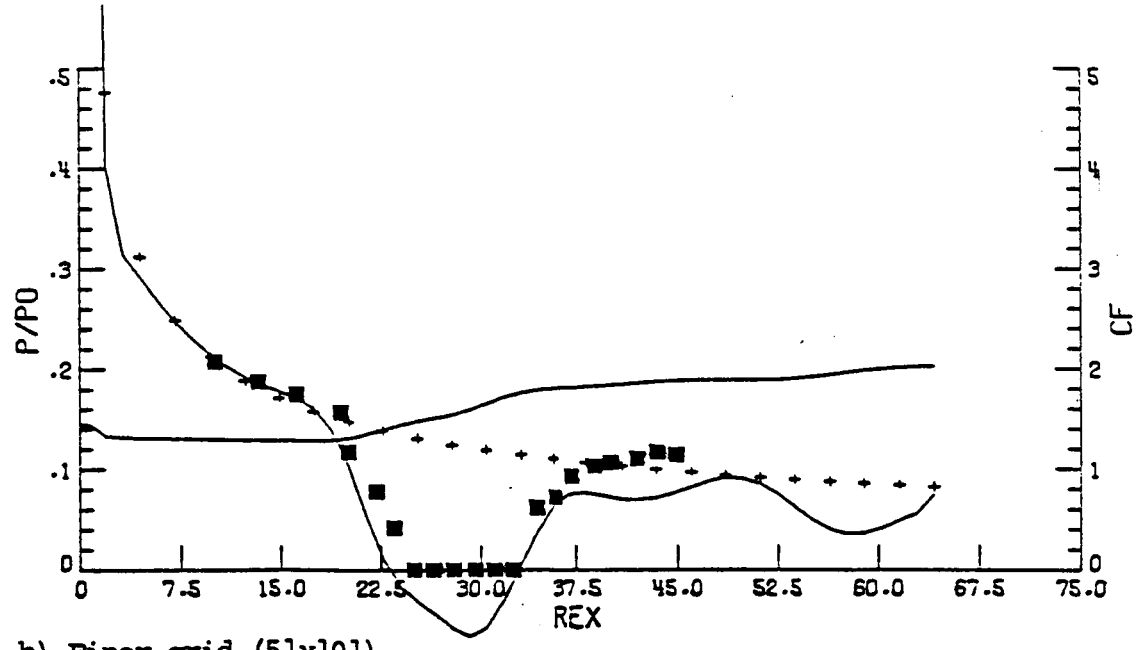


Fig. 8 - Convergence history for NACA 0012 airfoil at $M_\infty = 0.63$, angle of attack = 2°.



a) Coarse grid (51x51)



b) Finer grid (51x101)

■ experiment
\+ Blasius solution
— present results

Fig. 9 - Results for viscous diffusor calculations. Skin friction coefficient and pressure ratio as compared with experimental data.

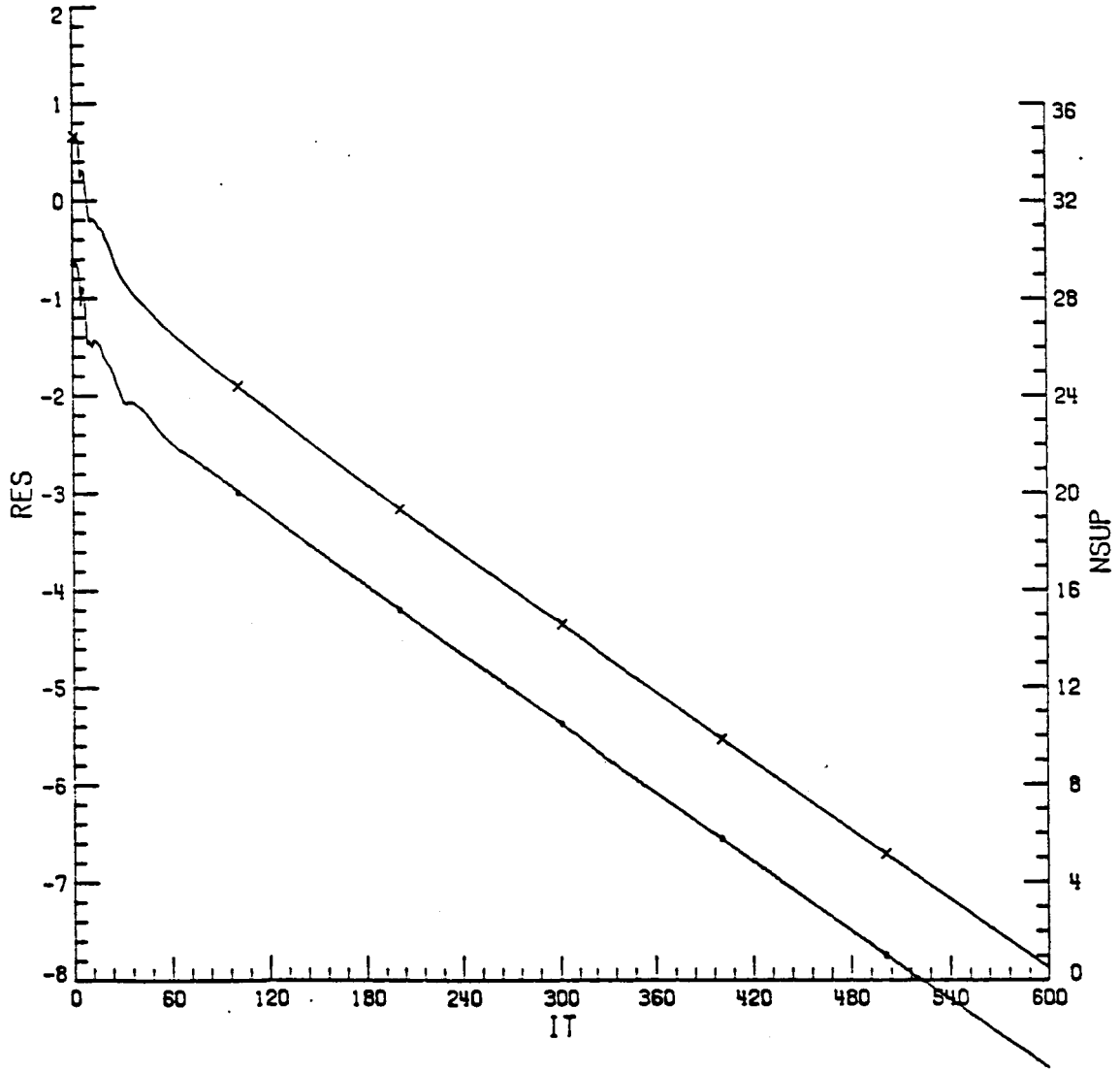


Fig. 10 - Convergence history for supersonic viscous diffusor using finer grid (51x101) at optimum maximum CFL number of 1450.

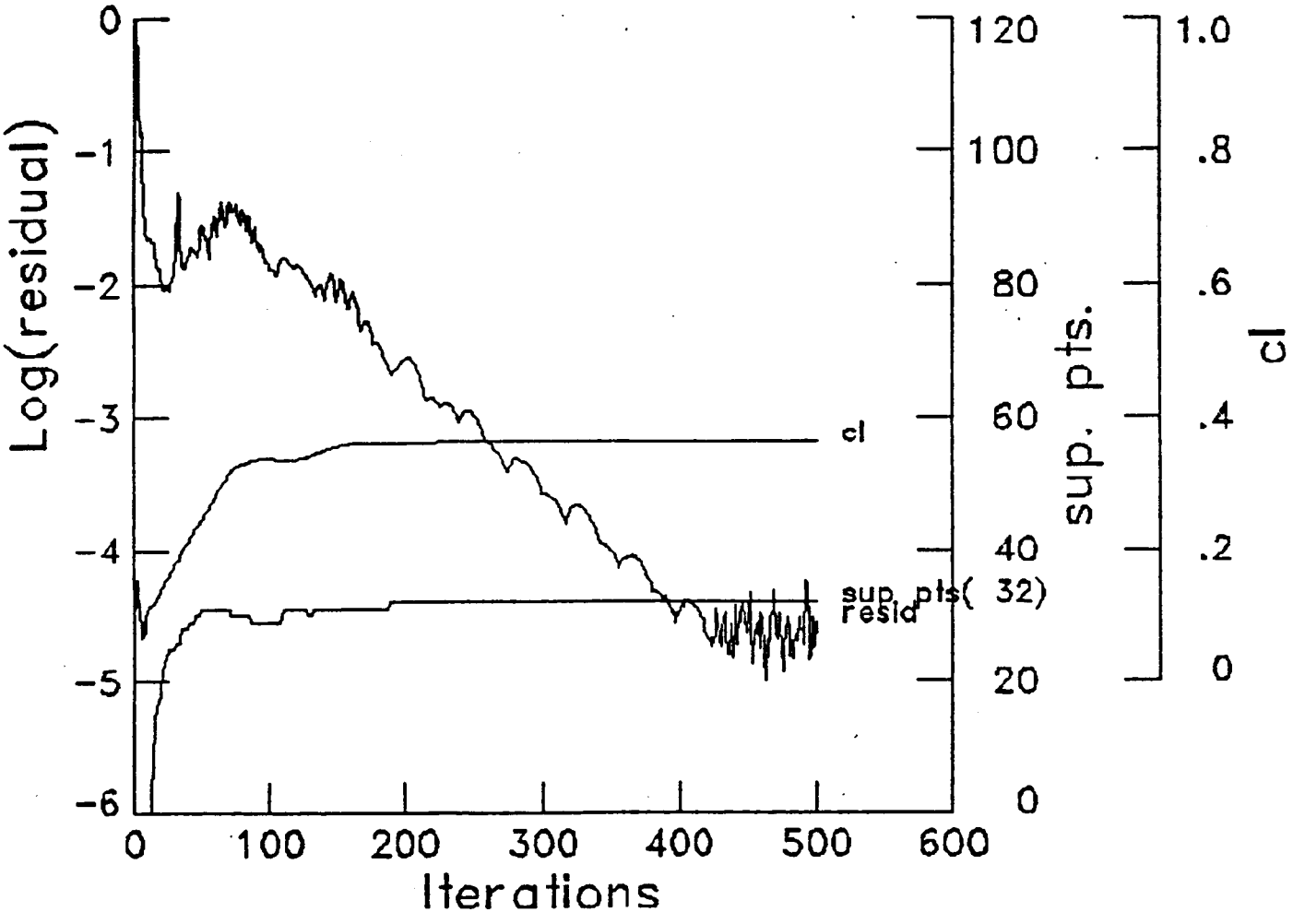


Figure 11.- Single grid iteration history at CFL = 21 for M_∞ = 0.8 and $\alpha = 1.25^\circ$ (Case 1).

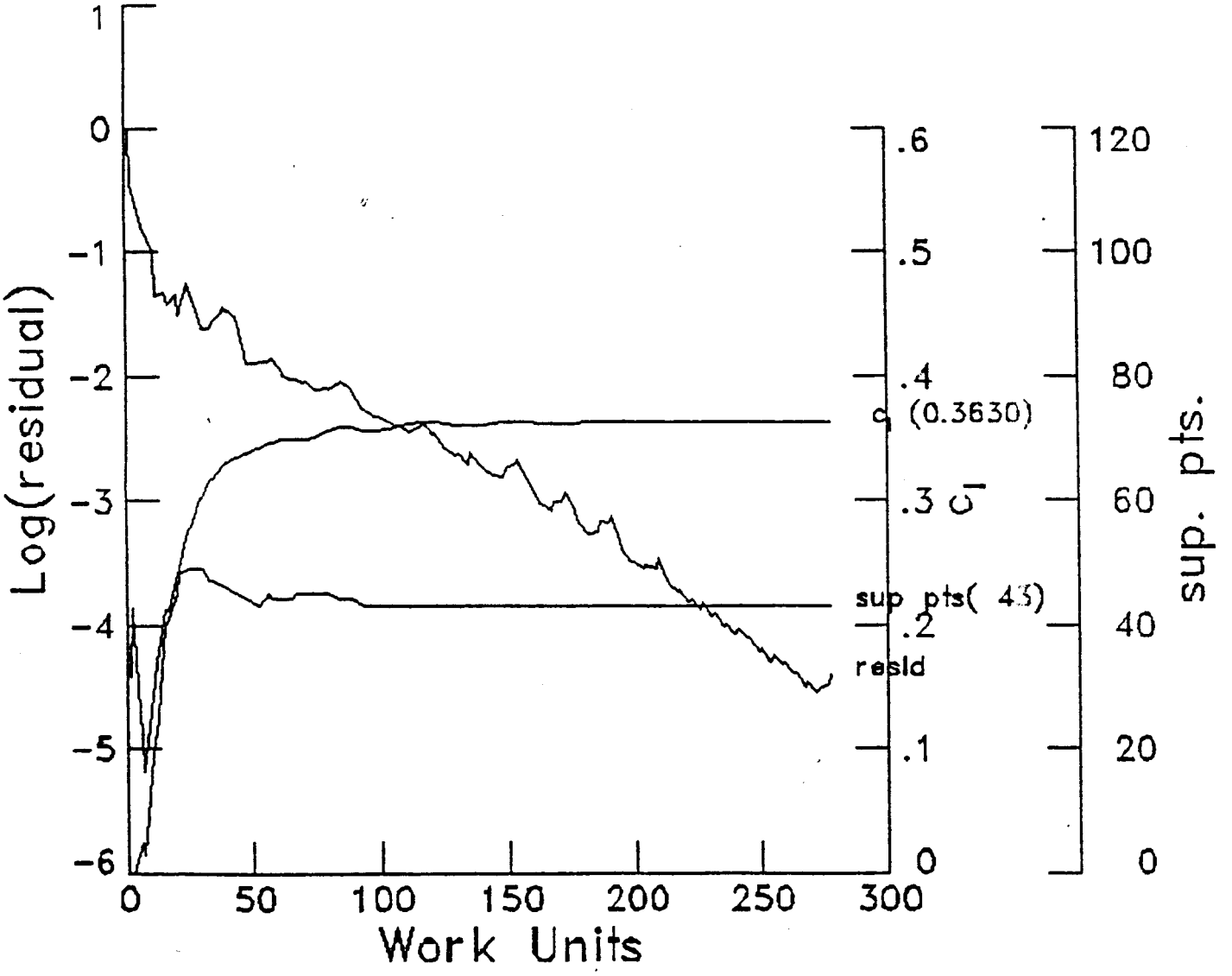


Figure 12.- Four grid multigrid iteration-history (60 cycles) at CFL = 21 for M_∞ = 0.8 and α = 1.25° (Case 2).

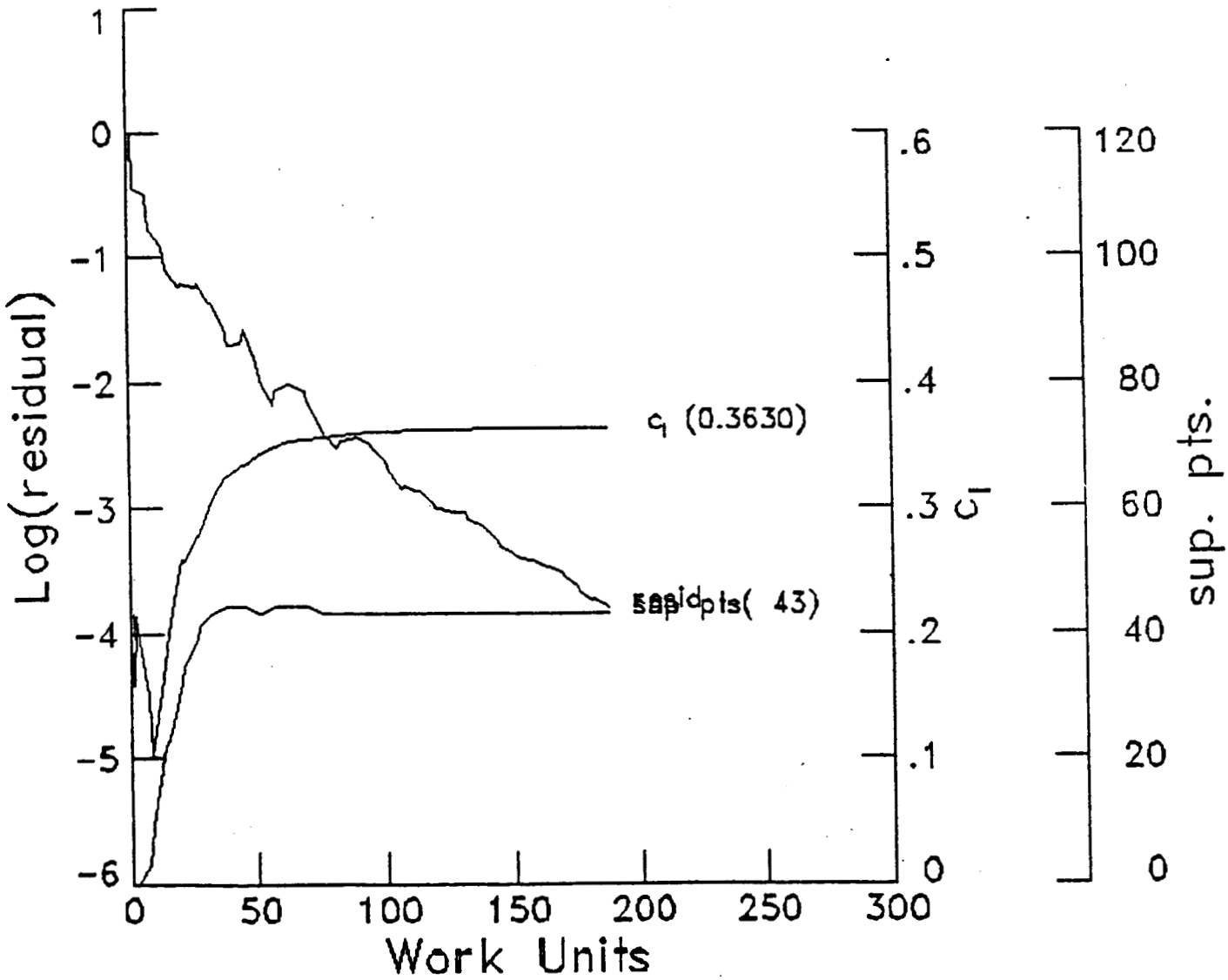


Figure 13.- Four grid multigrid iteration-history (30 cycles) at CFL = 21 for M_∞ = 0.8 and α = 1.25° (Case 3).

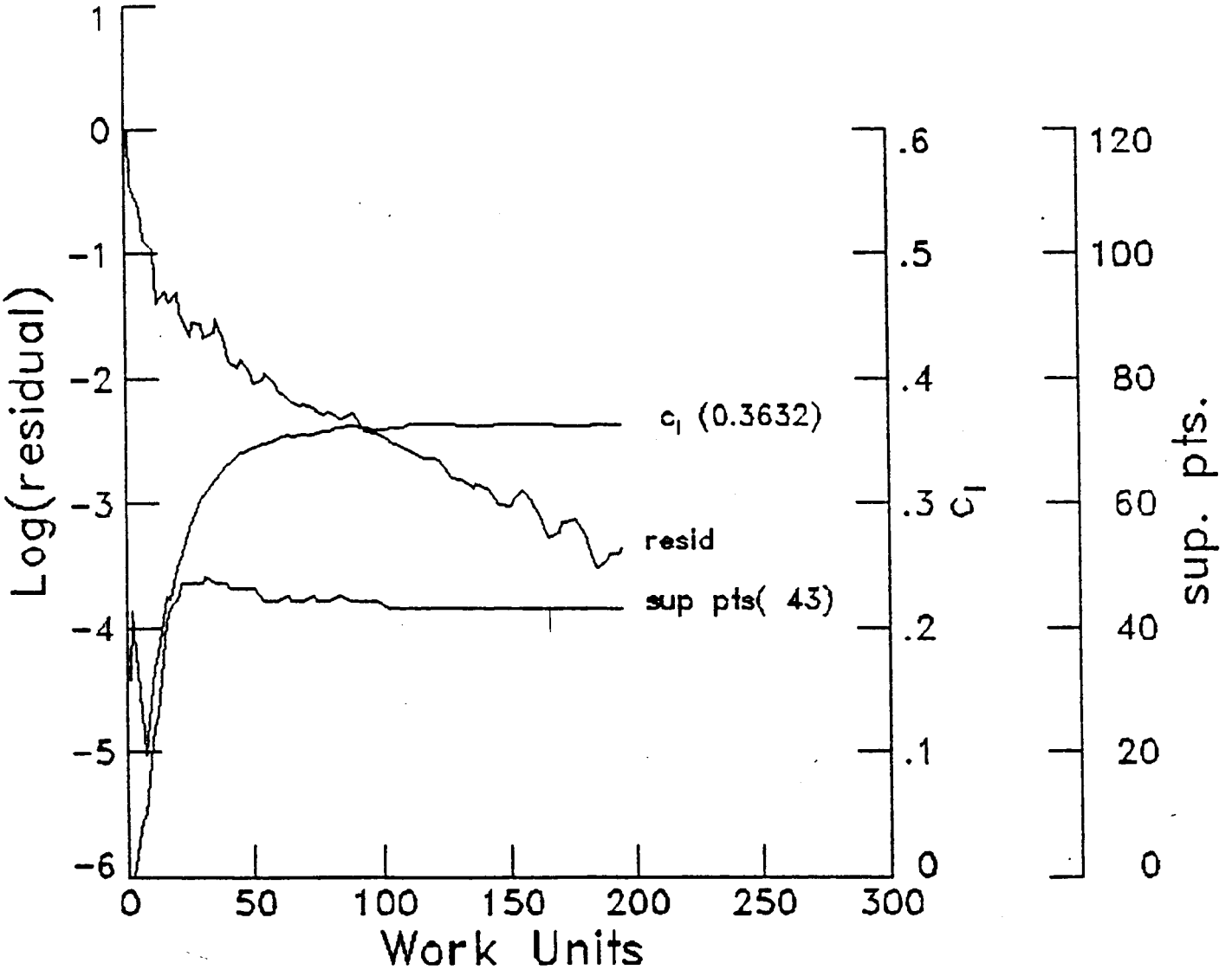


Figure 14.- Four grid multigrid iteration-history (40 cycles) at CFL = 21 for M_∞ = 0.8 and α = 1.25° (Case 4).

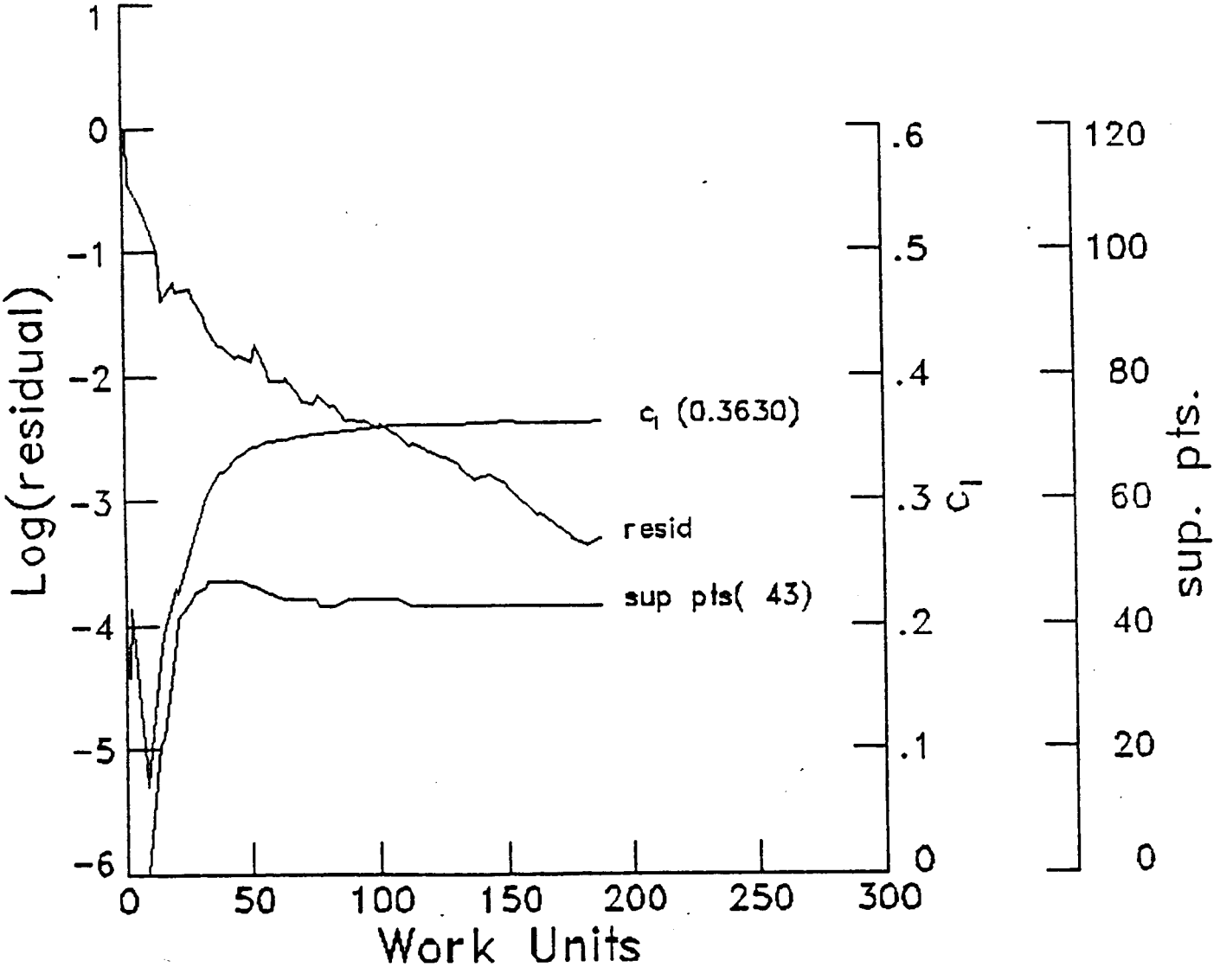


Figure 15.- Four grid multigrid iteration-history (30 cycles) at CFL = 21 for $M_\infty = 0.8$ and $\alpha = 1.25^\circ$ (Case 5).