N O T I C E

THIS DOCUMENT HAS BEEN REPRODUCED FROM MICROFICHE. ALTHOUGH IT IS RECOGNIZED THAT CERTAIN PORTIONS ARE ILLEGIBLE, IT IS BEING RELEASED IN THE INTEREST OF MAKING AVAILABLE AS MUCH INFORMATION AS POSSIBLE

OLD DOMINION UNIVERSITY RESEARCH FOUNDATION

DEPARTMENT OF MATHEMATICAL SCIENCES
SCHOOL OF SCIENCES AND HEALTH PROFESSIONS
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA

A MODIFIED DODGE ALGORITHM FOR THE PARABOLIZED NAVIER-STOKES EQUATIONS AND COMPRESSIBLE DUCT FLOWS

(NASA-CR-164435) A MODIFIED DODGE ALGORITHM FOR THE PARABOLIZED NAVIER-STOKES EQUATIONS AND COMPRESSIBLE DUCT FLOWS Progress Report, 1 Dec. 1980 - 31 May 1981 (Old Dominion Univ., Norfolk, Va.) 22 p N81-25334
G3/34 Unclas 26560

By

C. H. Cooke, Principal Investigator

Progress Report
For the period December 1, 1980 - May 31, 1981

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

JUN 1981
RECEIVED
NASA STI FACILITY
ACCESS DEPT.

Under
Research Grant NSG 1517
Douglas L. Dwoyer, Technical Monitor
Subsonic-Transonic Aerodynamics Division

June 1981

DEPARTMENT OF MATHEMATICAL SCIENCES
SCHOOL OF SCIENCES AND HEALTH PROFESSIONS
OLD DOMINION UNIVERSITY
NORFOLK, VIRGINIA

A MODIFIED DODGE ALGORITHM FOR THE PARABOLIZED NAVIER-STOKES EQUATIONS AND COMPRESSIBLE DUCT FLOWS

By

C. H. Cooke, Principal Investigator

Progress Report
For the period December 1, 1980 - May 31, 1981

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

Under
Research Grant NSG 1517
Douglas L. Dwoyer, Technical Monitor
Subsonic-Transonic Aerodynamics Division

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

June 1981

A MODIFIED DODGE ALGORITHM FOR THE PARABOLIZED NAVIER-STOKES EQUATIONS AND COMPRESSIBLE DUCT FLOWS

By

C. H. Cooke*

SUMMARY

A revised version of Dodge's split-velocity method for numerical calculation of compressible duct flow has been developed. The revision incorporates balancing of mass flow rates on each marching step in order to maintain front-to-back continuity during the calculation. The (checkerboard) zebra algorithm is applied to solution of the three-dimensional continuity equation in conservative form. A second-order A-stable linear multistep method is employed in effecting a marching solution of the parabolized momentum equations. A checkerboard SOR iteration is used to solve the resulting implicit nonlinear systems of finite-difference equations which govern stepwise transition.

INTRODUCTION

It has been said that the full Navier-Stokes equations represent the ultimate mathematical model upon which to base numerical algorithms for predicting flows of practical significance. However, even with the advent of the so-called vector computers with vast virtual memory and quadrupled processing speeds, extant numerical and computational difficulties are sufficient to merit a search for simpler mathematical models and less complicated numerical methods which can still provide useful solutions to problems of interest. Thus, considerable analysis and numerical experiment has been devoted to the exploitation of parabolized marching methods for flow prediction. References 1 to 7 represent a perhaps typical but by no means exhaustive sampling of the available literature on this subject.

---

*Professor, Department of Mathematical Sciences, Old Dominion University, Norfolk, Virginia 23508.

The parabolized marching methods are somewhat more general in application than the classical boundary-layer approach, since transverse pressure gradients are not disregarded and, in some cases, upstream influences can be transmitted through the pressure field. However, the basic assumption that streamwise viscous diffusion can be neglected restricts application to flows with a primary flow direction, limited upstream influence, and which may exhibit at worst crossplane recirculation. Unfortunately, in subsonic and transonic wind-tunnel flows, the elliptic upstream influence can be a significant factor in the flow dynamics; hence, interest arises in simpler mathematical models which permit this interaction. A case in point has been the development of Dodge's velocity splitting method, which allows global propagation of influence through the pressure field, and which has met with successes in both unconfined compressible and confined incompressible flows (ref. 7-10). However, as yet the method is by no means fully proven.

In this paper we shall be concerned with the application of a compressible formulation of Dodge's split velocity technique (ref. 9) to the calculation of developing (nonentry region) flow in a square duct. The original method has been revised to effect constant mass flow rate on each transverse plane while marching down the channel. Parabolized momentum equations are employed. However, a fully elliptic pressure field is allowed by the iterative manner in which the solution of the continuity equation is coupled into the calculation procedure. Application of the presently developed computer algorithm is restricted to subsonic flow. It could readily be altered to allow transonic calculations through modification or replacement of the algorithm used to solve the conservative continuity equation.

Computational simplicity as well as numerical stability is achieved in marching the momentum equations with an A-stable (ref. 11) implicit linear multistep method, the equations of which are iteratively solved at each step by employing checkerboard successive overrelaxation. While this solution procedure may be considered expensive, the presence of quadratic as well as higher order nonlinearities in the parabolized momentum equations requires that some iteration be employed to improve accuracy. As an extra benefit, the wide-ranging stability of the

resulting marching equations appears well worth the cost.

Finally, the peak efficiency of the methods developed is undoubtably best realized on the computer system for which it has been designed, namely, the Cyber 203. For such machines, a numerical algorithm must effectively exploit the array processing capabilities; otherwise, methods which are not highly vectorizable misuse the available computing potential and can result in quite ordinary processing speeds. The explicit nature of the checkerboard algorithm yields a highly vectorizable method ideally suited for the array processor.

In certain parabolized marching schemes for confined flow (ref. 1) it has been the practice to decouple streamwise and transverse pressure gradients. Some argue (ref. 12) that this is necessary in order to obtain meaningful physical solutions with parabolized equations. While results are still inconclusive, computational experience gained in the current research appears to support this belief. Weak, but not total, uncoupling of the streamwise pressure gradient has appeared necessary, although this may stem from the manner in which local continuity of mass flow is enforced.

LIST OF SYMBOLS

C_p, C_v	specific heats
P	static pressure
ρ	density
$\overline{w}$	3-D velocity vector
μ	viscosity
Re	Reynolds number, $Re = \frac{P_o a_o D}{\mu_o}$
γ	C_p/C_v
T	temperature
$T_o, P_o, \rho_o, a_o, \mu_o$	reservoir values for temperature, pressure, density, speed of sound, and viscosity
ϕ	scalar potential
α	relaxation parameter
D	channel half-width
M	Mach number

ORIGINAL PAGE IS
OF POOR QUALITY

PARABOLIZED GOVERNING EQUATIONS

The nondimensionalized Navier-Stokes equations for compressible steady flow with which we shall be concerned are now written.

Continuity:

$$\nabla \cdot \rho\bar{w} = 0 \tag{1}$$

Momentum:

$$\rho\left(\bar{w} \cdot \nabla\right)\bar{w} = -\nabla P - \nabla x\left(\frac{\mu}{Re}\nabla x\bar{w}\right) + \nabla\left(\frac{4}{3}\frac{\mu}{Re}\nabla \cdot \bar{w}\right) \tag{2}$$

Energy:

$$T = T_o - \frac{1}{2}\bar{w}^2 \tag{3}$$

Here, for flow in ducts with nonconducting walls, the usual energy equation has been replaced by the algebraic constant total temperature, relation (3). The constitutive relations are

$$P = \frac{\gamma-1}{\gamma}\rho T \tag{4}$$

and the viscosity approximation

$$\mu = (\gamma - 1)T. \tag{5}$$

For subsonic flow the governing equations are elliptic. However, a common approximation used to parabolize these equations (refs. 1,2) is obtained by neglecting streamwise diffusion terms in equation (2). With the exception of the entry region, the approximation is considered valid for flow in channels whose lengths are large compared to half-width (ref. 2). Perhaps it should be remarked that, when Dodge's method is applied in obtaining numerical solutions of these equations, the approximation is only a partial parabolization since the pressure field is obtained from an elliptic boundary value problem. This, of course, allows global propagation of disturbances, through the pressure field and the iteration process.

DODGE'S METHOD

Introduction

In Dodge's method, the total velocity vector $\bar{w}$ is arbitrarily decomposed as a sum of rotational and irrotational parts:

$$\bar{w} = \nabla\phi + \bar{u} \tag{6}$$

where ϕ is a scalar potential. Pressure is hypothesized to depend solely upon the irrotational velocity acc .ding to the isentropic relationship

$$P = P_o(1 - \nabla\phi^2/2T_o)^{\gamma/\gamma-1} \tag{7}$$

However, density is decomposed as the sum of a viscous contribution ρ_v and an isentropic contribution ρ^*:

$$\rho = \rho_v + \rho^* \tag{8}$$

where

$$\rho^* = \rho_o(1 - \nabla\phi^2/2T_o)^{1/\gamma-1} \tag{9}$$

Substituting equations (6), (7), and (9) in equations (1) and (2) leads to the so-called split equations

$$\nabla \cdot \rho\nabla\phi = -\nabla \cdot \rho\bar{u} \tag{10}$$

and

$$\rho(\bar{w} \cdot \nabla)\bar{w} - \rho^*(\nabla\phi \cdot \nabla)\nabla\phi + \nabla x\left(\frac{\mu}{Re}\nabla x\bar{w}\right) - \nabla\left(\frac{4}{3}\frac{\mu}{Re}\nabla \cdot \bar{w}\right) = 0 \tag{11}$$

Equations (10) and (11) are to be iteratively solved: equation (10) with a 3-D relaxation method for elliptic equations, following which the parabolized version of equation (11) is marched downstream by employing a checkerboard iteration to solve an implicit system of equations at each step. A synopsis of the iteration procedure is now presented.

Overview of Iteration Procedure

(1) Determine a suitable initial pressure distribution P_o by estimating a global ϕ distribution. In this investigation, pressure on the first pass is assumed to be a function of only streamwise displacement, and a mass-balancing operation establishes the initial pressure field.

(2) Employing the current pressure field, march a parabolized version of equation (11) down the duct, simultaneously storing the right-hand side of equation (10). [See also eq. (17)].

(3) Solve equation (10) [or eq. (17)] to obtain an updated pressure field.

(4) Repeat the computational pass consisting of steps (2) and (3) until sufficient passes and a converged pressure field are obtained.

Dodge's Method Revised

Dodge (ref. 9) reports problems arising from adjustment of front-to-back continuity requirements with an iteration which is similar to that previously outlined. It is expected that this slow convergence stems from incomplete satisfaction of the continuity equation which could, for example, be solved after the momentum march terminates in some form such as

$$\nabla \cdot \rho_n \nabla \phi_n = -\nabla \cdot (\rho \bar{u})_{n-1} \tag{12}$$

This is in contrast to the usual parabolized marching methods for which both mass and velocity variables are updated at each marching step.

Physically, in order to maintain continuity in a channel flow, the mass flow rate

$$\omega = \iint \rho \left(\frac{\partial \phi}{\partial X} + u \right) dydz = \iint \rho \bar{w}_{normal} dydz \tag{13}$$

must remain constant at each transverse plane. However, in Dodge's (unrevised) method this provision is only weakly incorporated through equation (10), which is solved globally upon termination of a marching pass. Thus, poor satisfaction of mass balancing during the momentum marching process is only to be expected, as numerical experimentation indicates.

Consequently, we have chosen to revise the Dodge technique in a manner which alleviates this difficulty. This was at first attempted by employing

$$\phi = \int_0^X g(\xi) d\xi + \tilde{\phi}(x,y,z) \tag{14}$$

to write equation (12) in the form

$$\nabla \cdot \left(\rho \nabla \tilde{\phi} \right)_n = -\nabla \cdot \left(\rho \bar{u} \right)_{n-1} - \frac{\partial}{\partial X} (\rho g)_{n-1} \tag{15}$$

The function g is determined by iterating the numerical counterpart of the parabolized equation (11) at each fixed marching step until numerical balance of mass flow rate is achieved. This is accomplished through gradual changes in streamwise velocity, pressure, and density effected by the equation

$$g^{k+1} = g^k - \alpha\left[\omega - \iint\left(\rho w_{normal}\right)^k dydz\right] + \left[\iint \rho^k dydz\right] \tag{16}$$

with α a relaxation parameter. Aside from the benefit of an instanteous balance in mass flow rate, another merit of this device is that fewer global iterations are required in the relaxation solution of equation (15), as it is now more nearly satisfied at the outset.

However, this approach was found defective, in theory as well as in fact. The solving of equation (15) in the form indicated yields nonphysical results, as it provides a quasi-full potential transonic flow equation whose elliptic-hyperbolic transition point can differ markedly from Mach 1. This difficulty can be largely alleviated, although not totally circumvented, by replacing equation (15) with the equation

$$\nabla \cdot (\rho\nabla\phi)_n = -\nabla \cdot (\rho\bar{u})_{n-1} \tag{17}$$

whose point of transition more closely approximates the physics of the flow.

Pressure gradients in Dodge's unrevised method would be computed on pass n from the equation

$$\frac{\partial p^n}{\partial x_i} = -\left[\rho^*(\nabla\phi \cdot \nabla)\nabla\phi\right]_i^n \tag{18}$$

In the revised version, pressure gradients are allowed to develop during the mass-balancing iteration according to the equations

$$\frac{\partial p^{n,k}}{\partial x} = -\rho^*_{k,n}\left[g^k\, g_x^k + \phi_y^n\, \phi_{xy}^n + \phi_z^n\, \phi_{xz}^n\right] \tag{19}$$

$$\frac{\partial p^{n,k}}{\partial y} = -\rho^*_{k,n}\left[g^k\, \phi_{xy}^n + \phi_y^n\, \phi_{yy}^n + \phi_z^n\, \phi_{yz}^n\right] \tag{20}$$

$$\frac{\partial p^{n,k}}{\partial z} = -\rho^*_{k,n}\left[g^k\, \phi_{xz}^n + \phi_y^n\, \phi_{yy}^n + \phi_z^n\, \phi_{zz}^n\right] \tag{21}$$

The quantity g is determined through equation (16), and g_x by second-order backward differencing. This precedure represents a weak decoupling of the streamwise pressure gradient, since the g terms are the dominant contributions, and since these contributions are determined from local plane-to-plane continuity considerations, somewhat independently of the output from the global continuity equation [eq. (17)] on the previous pass.

NUMERICAL ANALYSIS

The algorithm deemed most efficient for numerically solving equation (17) on the array-processing computer is the Zebra algorithm of South et al. (ref. 13). This 3-D relaxation technique is in some respects similar to the hopscotch method of Gourlay (ref. 14). In equation (17) central differences are applied to all derivative terms. Variables in plane i are updated in checkerboard fashion, plane by plane in a downstream sweep, using already updated values at plane $i - 1$ and old iteration values in plane $i + 1$. Iterative repetition of downstream sweeps is used to converge the field, with a relaxation parameter employed to speed convergence.

A second-order accurate, implicit linear multistep method is used on equation (11) to march in the stepwise direction. The implicit equations are iteratively solved using a checkerboard successive overrelaxation scheme, with mass balancing built in as previously described. Streamwise derivatives are backward differenced second-order accurate, while derivatives in the (transverse) cross-plane are approximated second-order using central differences. A prediction of form

$$f_i = 2f_{i-1} - f_{i-2} \tag{22}$$

is used to initially estimate a velocity variable in plane i. The checkerboard method is then employed on the differenced counterpart of equation (11) to update variables in plane i in two cycles, with values updated on cycle 1 fed into the succeeding cycle. This two-cycle update process is iterated, employing equations (16) and (19) to (21) to alter the flow speed and pressure gradients until a balance in mass flow is achieved.

DEVELOPING FLOW IN A STRAIGHT DUCT

The revised method of Dodge has been employed to develop a finite-difference numerical model for three-dimensional viscous flows in confined regions. For boundary-layer resolution, the capability to allow individual coordinate stretching in each coordinate direction has been incorporated.

ORIGINAL PAGE IS
OF POOR QUALITY

The method so developed has been programmed using the SL1 vector language for the Cyber 203 array processor, and appears debugged. The 32-bit half-word option of SL1 has been employed in programming the Zebra relaxation algorithm for solving equation (17), while 64-bit full-word arithmetic is used in programming the checkerboard marching algorithm. The program has been tested by application to the problem of computing the steady developing flow in a straight duct (see fig. 1). Boundary conditions for the problem are now given.

Boundary Conditions

Inflow: $T = T_o - \frac{1}{2} u_i^2$

specified velocity profiles, $\overline{W}_a = \overline{H}(y,z)$,

$\rho_i = R(y,z)$, P_i = constant

$\phi_x(0,y,z) = g(0) = g_o$

Duct walls: velocity no slip, $\phi_n = 0$, $T = T_w$, $\rho = \rho_w$

Outflow: ρ_v extrapolated, $\phi = \phi_m$, g_m extrapolated

Artificial barriers: The computational domain is taken to be one quarter of the total duct cross section, and symmetry conditions are applied at the two resulting (nonwall) artificial barriers. Here the normal velocity component vanishes together with normal derivatives of ϕ and the other velocity variables. The variables P and T, of course, depend on ϕ and velocity at these boundaries. However, for constant total temperature, vanishing normal derivative in T, ρ is the natural boundary condition.

On the first pass, the value ϕ_m is allowed to develop in the calculation from mass flow rate balancing down the duct. Thereafter, it is held fixed.

COMPUTATIONAL RESULTS

The method developed and programmed has been exercised by application to the duct flow calculation with Reynolds number = 100 and Mach number ranging from 0.1 to 0.3 down the duct. An initial pressure distribution

$$\phi = \int_0^x g(x)dx \tag{23}$$

is determined by mass balancing on the first pass. On successive

marching passes, this distribution is corrected according to equation (17) plus whatever g corrections are necessary in order to balance mass at each crossplane of the calculation. The above process is iteratively repeated until the maximum change in the pressure field becomes sufficiently small.

Figures 2 to 4 exhibit computational results after 32 passes. However, it can be inferred from the numerical performance indicators shown by figures 5 to 7* that the iteration has sufficiently converged to produce essentially the same results in around 20 iterations. This figure corresponds roughly to that given by Dodge (ref. 9) as the number necessary to converge a similar problem. However, no real comparison of iteration counts to convergence can be drawn, as tolerance levels for convergence of various iterative processes could be expected to affect the number of passes needed.

Since the flow was not started at the channel entrance, but with a velocity profile supplied at some point farther downstream, not much can be said in terms of quantitative assessment of the numerical results, which for the most part appear qualitatively excellent. The direction of the crossflow and other features in figure 2 appear reasonable and agree with that of a computational experiment by Baker (ref. 15) for laminar corner flow. The approximately linear variation in centerline pressure exhibited by the graph in figure 3 is certainly reasonable for nearly incompressible flow, as also exhibited away from the channel entry region in a computational experiment of Briley (ref. 1). Perhaps the most questionable feature of the results is the tail-off in centerline velocity of figure 4. This could be caused by a calculation not yet completely converged near the outflow. However, it is perhaps more likely to be the result of the outflow boundary condition treatment. For example, the condition ϕ = constant in the outflow plane forces ϕ_y and ϕ_z to vanish, which may not be physically reasonable throughout the outflow plane, particularly since this does not happen upstream.

*Ordinate values in figures 4 to 7 have been magnified by a factor of 10.

SUMMARY AND CONCLUSIONS

A revised version of Dodge's split-velocity method for numerical solution of compressible confined flow has been developed. Preliminary results for low Mach number flow appear encouraging. However, the method in general is by no means fully understood or confidently tested. A curious feature of the present approach is exhibited by the need for weak decoupling of streamwise pressure gradients in order to achieve a convergent numerical process. However, Spalding (ref. 12) alleges that, in order to achieve physical solutions, a full decoupling is necessary with parabolized equations, and Briley (ref. 1) reports successful and meaningful calculation obtained using an algorithm incorporating this practice. Other questions which bear investigation concern the performance of the revised algorithm for higher Reynold's and Mach number flows. To gain further confidence in the method, detailed comparisons with independent computational results need to be initiated.

The revision of Dodge's method reported herein is new to the method, although classical in physical origins and certainly used previously with other computational methods. This investigation has proven the checkerboard iteration to be a convenient method for solving implicit finite-difference models of the Navier-Stokes equations on the vector computer. Further evidence of the computational utility of the zebra algorithm for solving the full-potential equation (with a forcing term added) in three dimensions has also been gained.

In conclusion, it is expected that forthcoming investigation will be directed:

(a) to providing details of computation times for the revised Dodge method,

(b) to obtaining comparisons with independent numerical results, and

(c) to calculating higher Reynolds and Mach number flow.

REFERENCES

1. Briley, W.R.: Numerical Method for Predicting Three-Dimensional Steady Viscous Flow in Ducts. J. Comput. Phys., Vol. 14, No. 1, Jan. 1974.

2. Carlson, G.A.; and Hornbeck, R.W.: A Numerical Solution for Laminar Entrance Flow in a Square Duct. J. Appl. Mech., Mar. 1973, pp. 25-30.

3. Chilukuri, R.; and Pletcher, R.H.: Numerical Solutions to the Partially Parabolized Navier-Stokes Equations for Developing Flow in a Channel. Numerical Heat Transfer, Vol. 3, 1980, pp. 169-188.

4. Rubin, S.G.; Khosla, P K.; and Saari, S.: Laminar Flow in Rectangular Channels. Computers and Fluids, Vol. 5, 1977, pp. 151-173.

5. Hirsh, Richard S.: Numerical Solution of Supersonic Three-Dimensional Free Mixing Flows Using the Parabolic-Elliptic Navier-Stokes Equations. NASA TN D-8195, Nov. 1976.

6. Patanker, S.V.; and Spalding, D.B.: A Calculation Procedure for Heat, Mass, and Momentum Transfer in Three-Dimensional Parabolic Flows. Int. J. Heat and Mass Transfer, Vol. 15, 1972, pp. 1787-1806.

7. Cosner, R.R.: Fast Navier-Stokes Solution of Transonic Flowfield about Axisymmetric Afterbodies. AIAA-80-0193, AIAA 18th Aerospace Sciences Meeting (Pasadena, Calif.), Jan. 14-16, 1980.

8. Dodge, P.R.: A Numerical Method for 2-D and 3-D Viscous Flows. AIAA Paper No. 76-425, 9th Fluid and Plasma Dynamics Conference (San Diego, CA), July 14-16, 1976.

9. Dodge, P.R.: 3-D Heat Transfer Analysis Program. AFAPL-TR-77, 21-2541(7), Sept. 1977.

10. Dwoyer, D.L.: Velocity Split Navier-Stokes Solution Procedure for Incompressible High Reynolds Number External Flows. NASA TN-1655, Apr. 1980.

11. Dahlquist, G.: A Special Stability Problem for Linear Multistep Methods. BIT, Vol. 3, 1963, pp. 27-43.

12. Spalding, D.B.: Numerical Computation of Steady Boundary Layers. Proceedings, Conf. on Computational Methods and Problems in Aeronautical Fluid Dynamics. Univ. of Manchester, Sept. 1974.

13. South, J.C.; Keller, J.D.; and Hafez, M.M.: Vector Processor Algorithms for Transonic Flow Calculations. Proc. of AIAA Fourth Computational Fluid Dynamics Conference (Williamsburg, VA), July 23-24, 1979.

ORIGINAL PAGE IS
OF POOR QUALITY

14. Gourlay, A.R.: Hopscotch: A Fast Second-Order Partial Derivative Equation Solver. J. Inst. Math. Appl., 1970, Vol. 6, pp. 375-390.

15. Baker, A.J.; and Orzechowski, J.A.: A Continuity-Constraint Finite Element Algorithm for Three-Dimensional Parabolic Flow. Proceedings, Joint ASME-AIAA Symposium on Computers in Flow Prediction and Flow Dynamics Measurements (Washington, DC), Nov. 15-20, 1980 (to appear).

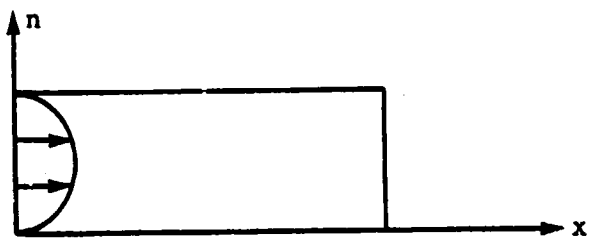

Figure 1. Duct profile.

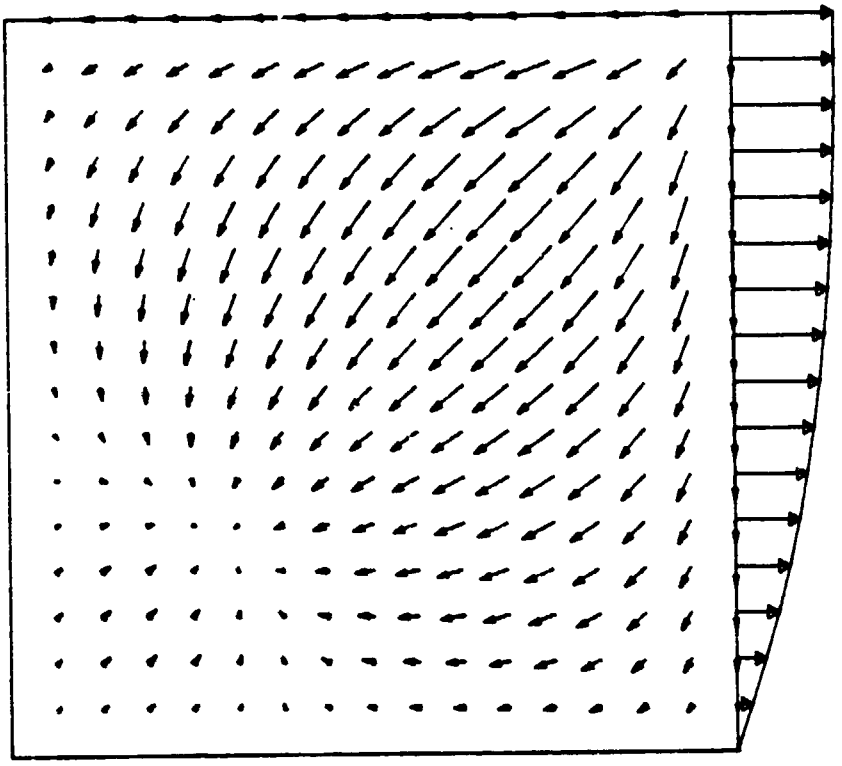

Figure 2. Crossflow and streamwise velocity component at channel exit.

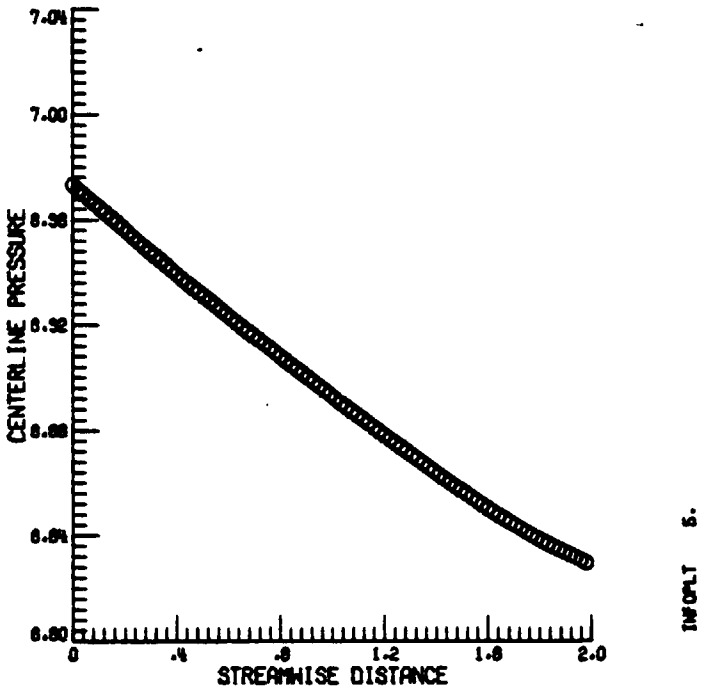

Figure 3. Centerline pressure variation down channel.

ORIGINAL PAGE IS
OF POOR QUALITY

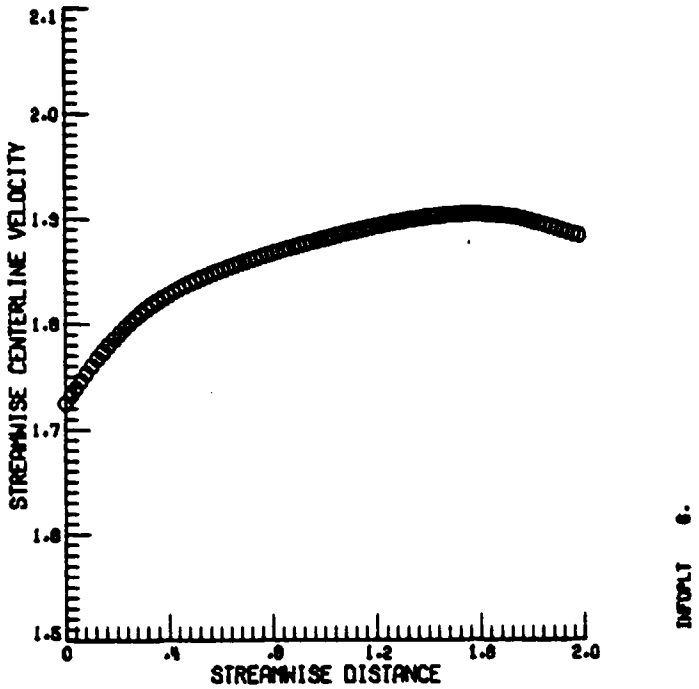

Figure 4. Variation in centerline streamwise velocity component.

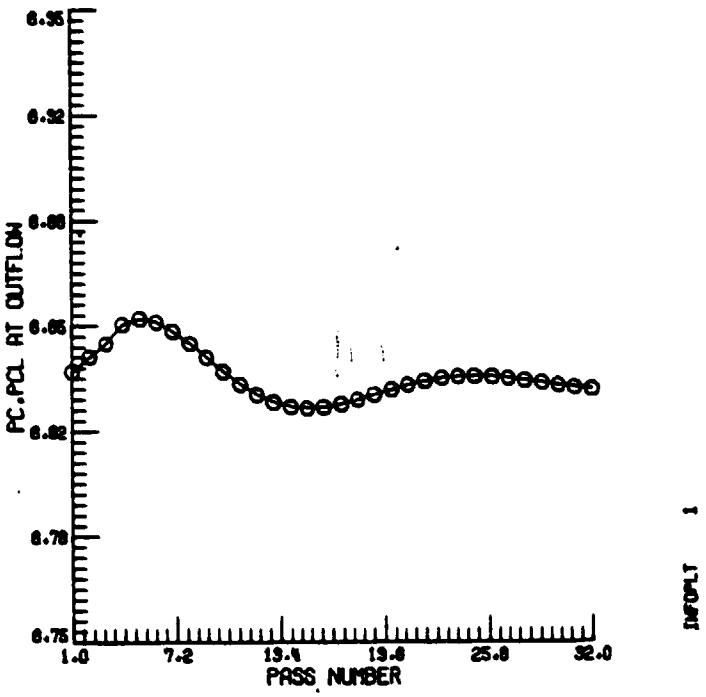

Figure 5. Outflow duct centerline pressure vs. iteration count.

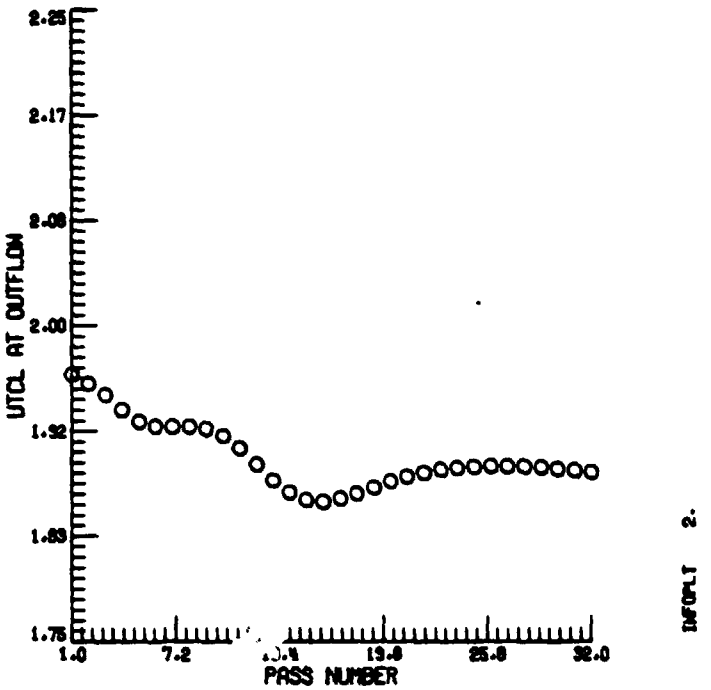

Figure 6. Outflow duct centerline streamwise velocity vs. iteration count.

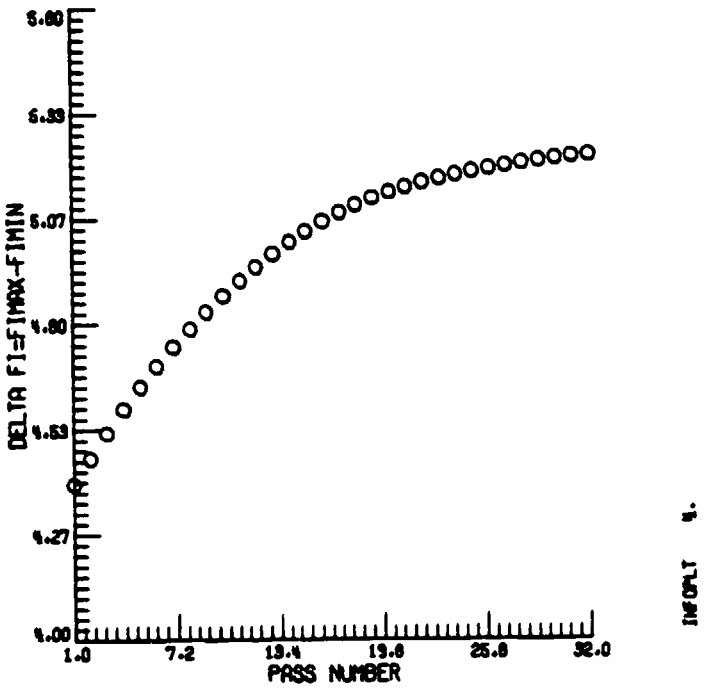

Figure 7. Channelwise potential increment $\Delta\phi$ vs. iteration count.

ORIGINAL PAGE IS
OF POOR QUALITY