To the Graduate Council:

I am submitting herewith a thesis written by Peter E. Raad entitled "A General Grid Mapping and Finite Difference Investigation of Noise Propagation in Variable Area Ducts." I recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Mechanical Engineering.

James W. White

James W. White, Major Professor

We have read this thesis
and recommend its acceptance:

R J Krone

James A. Eulz

Ralph V. Grimilli

Accepted for the Council:

L Evans Peck

Vice Chancellor
Graduate Studies and Research

A GENERAL GRID MAPPING AND FINITE DIFFERENCE INVESTIGATION OF NOISE PROPAGATION IN VARIABLE AREA DUCTS

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Peter E. Raad

December 1981

3055551

DEDICATION

This thesis is dedicated to:

Virgin Mary, the mother of my Lord Jesus Christ

Heleny, my late grandmother

Muntaha, my mother

Jocelyne, my fiancee

ACKNOWLEDGMENTS

The author wishes to acknowledge his advisor, Dr. James W. White without whom this work would not have gone very far. His constant willingness to help during the difficult periods as well as his friendly attitude made this work experience a very pleasant one.

Thanks are also due to the members of the committee: Dr. R.V. Arimilli, Dr. J.A. Euler, and Dr. R.J. Krane for their guidance and support.

Appreciation is also expressed to NASA Lewis Research Center for their financial support of this work and to Dr. K.J. Baumeister, the NASA Technical Officer for the grant, for supplying related research papers and experimental results.

The author acknowledges the use of the University of Tennessee Computing Center, Knoxville, Tennessee, in obtaining the results shown here.

Recognition is given also to Carolyn Dexter, who assisted in the typing of this thesis.

Special thanks to all the special friends who made me feel at home away from home.

My love and appreciation to my relatives and friends back home for their love and their spiritual and financial support. They made it all possible.

My deepest love and admiration to my fiancee, Jocelyne Dahdah, for her faithfulness, spiritual support and endless love for the last five years. She makes all of this worthwhile.

Thank you Mom for being there from the minute I was born until this minute. If I owe anyone anything, I owe you everything. I will always love you and I will never be able to repay you. Thanks for your excellent cooking and your help in typing parts of this thesis.

I thank the Almighty God for all His blessings.

ABSTRACT

The primary objectives of this work were to develop grid mapping procedures and numerical techniques required for the simulation of noise propagation in variable area ducts. The techniques developed are capable of handling soft wall boundaries as well as satisfying far field radiation conditions.

In order to simulate the noise propagation in variable area ducts, the effort was divided into two main parts. First, irregular geometries in the physical (X-Y) plane were transformed into a square grid in the ($\xi - \eta$) plane using a general numerical mapping procedure. Next, the noise equations were solved in the transformed plane using an explicit finite difference technique.

The work resulted in a general FORTRAN program that was tested for cases containing different boundary conditions including soft and hard wall top and bottom conditions, and plane wave and far field exit conditions. The results were compared to available numerical and experimental data.

The noise simulation was found to be successful when a plane wave exit condition was used, whereas problems were encountered when a far field exit was used. The problems were determined to be the failure of the explicit method to eliminate numerically generated erroneous "steady state reflections" in the noise solution, and the

fact that the pressure wave did not expand exactly cylindrically in the vicinity of the far field exit boundary.

Recommendations for follow-up work include the use of an implicit scheme in place of the less sophisticated explicit methods, and the use of a free surface method at the far field exit boundary to insure correspondence between the physical shape of the exit boundary and the leaving pressure wave.

TABLE OF CONTENTS

CHAPTER		PAGE
I.	INTRODUCTION	1
II.	MAPPING OF THE FAR FIELD	11
III.	AN ALTERNATE MAPPING PROCEDURE	15
IV.	INVESTIGATION OF EXPLICIT AND IMPLICIT FINITE DIFFERENCE TECHNIQUES	17
V.	NUMERICAL SOLUTION OF THE PRESSURE FIELD	26
VI.	CONCLUSIONS AND RECOMMENDATIONS	37
LIST OF REFERENCES		43
APPENDIXES		49
APPENDIX A.	GENERAL MAPPING EQUATIONS AND PROCEDURE	50
APPENDIX B.	EXPLICIT AND FACTORED IMPLICIT FINITE DIFFERENCE EQUATIONS FOR A SQUARE DUCT	55
APPENDIX C.	DERIVATION OF THE EXPLICIT FINITE DIFFERENCE EQUATIONS FOR VARIABLE AREA DUCT ACOUSTICS	62
APPENDIX D.	USER'S GUIDE FOR THE PROGRAM NOISE	68
APPENDIX E.	SOURCE LISTING OF THE FORTRAN PROGRAM NOISE	70
APPENDIX F.	FIGURES	115
VITA		148

LIST OF FIGURES

FIGURE PAGE

1. Variable area duct in the physical (X-Y) plane 116
2. Computational grid in the transformed (ξ-η) plane 117
3. Variable area duct and far field: Regions and segments definitions 118
4. Variable area duct and far field: Initial condition. 11 grid points in the transverse direction 119
5. Variable area duct and far field: Final solution. 11 grid points in the transverse direction 120
6. Variable area duct and far field: Initial condition. 21 grid points in the transverse direction 121
7. Variable area duct and far field: Final solution. 21 grid points in the transverse direction 122
8. Variable area duct and far field: Initial condition. Non-uniformly spaced grid points in the transverse direction 123
9. Final solution of the initial condition in Figure (8) 124

10. Variable area duct and far field: Initial condition. Non-uniformaly spaced grid distribution in the transverse direction 125

11. Final solution of the initial condition in Figure (10) 126

12. Alternate mapping solution using an accelerated relaxation procedure 127

13. Initial condition and final solution of a rectangular duct 128

14. Absolute value of the maximum deviation from theoretical results versus time for the explicit solution of noise propagation in a uniform duct 129

15. Absolute value of the maximum deviation from theoretical results versus time for the implicit solution of noise propagation in a uniform duct 130

16. Absolute value of the maximum error in computed results versus time for the explicit solution of noise propagation in a uniform duct 131

17. Absolute value of the maximum error in computed results versus time for the implicit solution of noise propagation in a uniform duct 132

18. Variable area duct with far field: Variables' definitions and dimensions 133
19. Numerical and experimental axial pressure profiles for a variable area duct with a plane wave exit condition 134
20. Numerical and experimental transverse pressure profiles for the same duct in Figure (19) 135
21. Variable area duct with inlet and exit straight sections: Initial condition 136
22. Variable area duct with inlet and exit straight sections: Final solution 137
23. Numerical and experimental pressure profiles for the computational grid shown in Figure (22) 138
24. Square duct with one soft wall: Initial condition and final grid solution 139
25. Pressure magnitude along the duct axis for incident plane wave sound propagation in a soft wall duct 140
26. Pressure magnitude along flat section of variable area duct of Figure (1) after 3000 time steps of solution 141
27. Uniform area duct and far field: Initial condition 142

28. Uniform area duct and far field:
Final grid solution 143
29. Pressure magnitude along bottom wall of
the duct in Figure (28) 144
30. Pressure magnitude along flat wall of
variable area duct (Figure (5)) 145
31. Jet engine configuration and far field:
General shape 146
32 Computational square grid in (ξ-η) plane 147

NOMENCLATURE

C_0^*	ambient speed of sound, m/s
CPU	Central Processing Unit
f^*	frequency, Hz
i	grid location in ξ direction, also = $\sqrt{-1}$
j	grid increment in η direction
J	Jacobian, (see Appendix (A), equations (A-4))
k	time level
H^*	characteristic duct dimension in transverse direction, m
L^*	duct length, m
n	normal direction
P	dimensionless acoustic pressure, $=P^*/(\rho^* C_0^{*2})$, also grid control function
Q	grid control function
t	dimensionless time, $=t^* f^*$
t^*	time, s
u	dimensionless acoustic velocity in x direction, $=u^*/C_0^*$
u^*	acoustic velocity in x direction, m/s
v	dimensionless acoustic velocity in transverse direction, $=v^*/C_0^*$
v^*	acoustic velocity in transverse direction, m/s
X	dimensionless axial coordinate, $=X^*/H^*$
X^*	axial coordinate, m
Y	dimensionless transverse coordinate, $=Y^*/H^*$

Y^*	transverse coordinate, m
Z	maximum depth of variable area duct
α, β, γ	see Appendix (A), equations (A-6)
$\Delta\xi$, $\Delta\eta$	increments in transformed plane
∇	gradient operator
$\delta_{\xi\xi}$, $\delta_{\eta\eta}$	second order finite difference operators
ζ	specific acoustic impedance
ξ , η	coordinates in transformed plane
ρ^*	ambient fluid density, kg/m^3
ω	dimensionless frequency, $=H^*f^*/C_o^*$

CHAPTER I

INTRODUCTION

Since strict aircraft noise regulations were introduced in the late 1960's, the aerospace industry has become greatly involved in the efforts to reduce noise propagation in jet engines by treating the inner walls with acoustical absorbers. To accomplish this task while minimizing the weight of the absorbers, sophisticated and general purpose analytical techniques were needed to simulate sound propagation in variable area ducts with specified wall liner impedances. In a similar effort, the environmental industry has concerned itself with reducing the noise propagation from ventilation systems, and mufflers. From these efforts and some parallel academic studies resulted a number of finite element and finite difference techniques appropriate for the calculation of noise propagation. Even though many studies were directed more toward acoustic propagation in enclosed cavities such as rooms and automobile interiors [1-5], still many theories concentrated on applications with inlet and outlet flow, such as in jet engine ducts and mufflers. The numerical analyses which have provisions for inlet and outlet flows divide into several areas.

The steady state finite difference theory benefits from the fact that the acoustic properties are simple harmonic functions of time, resulting in governing equations which are independent of time. This theory was used by Alfredson [6] and Baumeister and Bittner [7] in their presentation of a finite difference theory for noise propagation in both hard, and soft wall ducts in the absence of flow. Baumeister and Rice [8] went on to include a mean flow in the previous analysis. Because of the nature of the chosen complex noise source, and the softwall boundary conditions, when present, Baumeister [8-9] showed that a large number of grid points was required in both the axial, and the transverse directions. This requirement was shown to be even more restrictive at higher frequencies. Still another problem faced the theory. The matrix resulting from the system of algebraic equations was found not to be diagonally dominant. As a result, iteration methods could not be used, but rather Gauss elimination had to be employed. Not only does the elimination technique generally require the storage of all the matrix elements, but in the two dimensional plane, the complex nature of the problem would quadruple the storage needed. For high frequencies, this would represent a great burden on even the largest of computers. Baumeister [9] and Quinn [10] have modified these large matrices to allow the use of iteration techniques but their procedures

yield a convergence which is too slow to be of much practical value in many situations. In addition, some tedious bookkeeping is required [11] when the finite difference techniques are extended to irregular ducts. In an attempt to resolve this problem, Quinn [12-13] applied the method of conformal mapping in conjunction with finite differences to analyze sound propagation in a hyperbolic horn, as well as a conical horn, and found the numerical results to be in excellent agreement with known analytical solutions.

Due to the problems previously discussed, the finite element theory [14-17] is often used in the analysis of ducts with a variable area. However, the steady state finite element theory requires very large matrices when many elements are required. As pointed out by Quinn [10], the advantage of finite elements versus finite differences is not so great when a mapping function is used to compute the mean flow. The mapping idea will be discussed in detail later. However, it can be noted here that with the use of a mapping procedure, the finite difference theory has a clear advantage over the finite element theory.

Special transformation techniques, such as the wave envelope technique [18-19], the marching techniques [20-21], and the multi-partitions techniques [22], have been developed and are being used to simulate noise propagation in turbojet engines, and in long sections.

However, special transformations are restricted, in general, to high frequencies and small reflections. They also tend to be specialized to rather particular situations.

In the past few years, in an attempt to eliminate matrix storage requirements, Baumeister [23-25] demonstrated the transient finite difference theory as an alternative to finite elements. Such an approach begins with a noise source radiating from the inlet boundary into a quiescent duct. An explicit iteration method then moves the solution in real time to obtain both the transient, and the steady state solutions. Baumeister demonstrated this time-dependent analysis to be considerably faster than the corresponding steady state methods. However, the computational times to steady state were generally comparable with the finite element methods for a moderate number of grid points. The major advantage of the transient analysis method remains to be the elimination of the matrix storage requirement which should make it useful in analyses requiring many grid points. This method still has problems in handling geometrically irregular boundary conditions, as well as a stability problem when the soft wall acoustic boundary conditions are included [23-25].

At this point, it can be seen that a good approach to the simulation of noise propagation in variable area ducts might be the use of a mapping procedure in conjunction

with a transient numerical technique. Such an approach should require less matrix storage than finite elements, and it should be computationally faster than a steady state technique. When this approach is used, special attention should be paid to accurately satisfying the area variations in the duct, as well as to the stability of the solution when soft wall conditions are applied to the existing boundaries. Chen and Caughey [26] used a conformal mapping technique to generate the required computational mesh to calculate the transonic and subsonic inlet flow fields in the DC-8 engine. However, the conformal mapping technique is limited to two dimensions, and allows only boundaries that can be generated by complex transformations. In an effort to expand their theory to include more general problems, Chen and Caughey [27] used a general grid-generation method to provide the computational mesh in a follow-up paper.

In acoustics as well as in other fields concerned with the numerical solutions of partial differential equations, there arises the need for accurate numerical representation of boundary conditions. This can be quite difficult when the boundary or boundary segments are geometrically irregular. Generally, one of two approaches are used when dealing with irregular boundaries. One is to approximately locate the boundary with a uniform grid. The other is to locate it exactly by the use of a variable

spacial grid. Neither of these approaches is desirable since the first one introduces error in the boundary location and the second results in less accurate finite difference approximations. The error resulting from the first approach is naturally intensified with sharper irregularities. The second approach requires "tailoring" which would greatly abate its generality. In mapping techniques, the most accurate way to represent boundaries is to make them coincide with some coordinate lines; for then, points of the finite difference grid are naturally located on the boundaries. In most partial differential equations, the boundary conditions constitute the most important factors to be satisfied. If the grid points on which the system of equations is to be solved do not coincide with the boundaries, inaccurate boundary conditions are introduced in the region of the problem where the solution is most highly sought. A general method of generating coordinate systems that fit the boundaries is to make the curvilinear coordinates solutions of a system of elliptic partial differential equations in the physical plane, with Dirichlet boundary conditions on all boundaries. By specifying a coordinate to be constant on each of the boundaries, a coordinate line is made coincident with every boundary. The resulting transformed coordinate system is not necessarily orthogonal. Orthogonality, however, is not required

unless scale factors and differential operators are used in the transformations [28, pg. 3].

The idea of a general grid-generation method was used as early as 1966 to map "irregular" physical planes into "regular" computational meshes. Winslow [29] and Chu [30] made the physical cartesian coordinates solutions of a quasi-linear elliptic system in the transformed plane by taking the transformed coordinates to be solutions of Laplace's equation in the physical plane. Barfield [31] and Amsden and Hirt [32] reversed the previous procedure and made the transformed coordinates solutions of a quasi-linear elliptic system in the physical plane by taking the physical cartesian coordinates to be solutions of a modified Laplace's equation in the transformed plane. Thompson et al followed the former idea and developed a method for the construction of body-fitted curvilinear coordinate systems applied to multi-connected regions in aerodynamic flows [28, 33-36]. The method achieves the solution to the mapping equations by successive over relaxation. Thompson's technique is quite general, extendable to three dimensions, and can even be applied to time dependent boundaries. However, solution of the mapping equations is dependent on the initial conditions in the physical plane, and requires the use of control parameters which might not be needed for simply connected domains.

In an effort to minimize the influence of the initial conditions in the physical plane on the stability and convergence of the solution, White [37] added a time derivative term to the quasi-linear elliptic equations originally used by Thompson. Since the mapping equations represent only geometric relationships and are independent of time, the time derivative terms added will only serve to smooth and dampen the solution. As time approaches "infinity", the solution of the equations will reach "steady state", and hence become independent of time. White used a factored implicit finite difference scheme to solve the quasi-linear partial differential equations. The results from White's theory proved to be very promising. White went on to solve the pressure equations over the transformed computational mesh of a variable area duct in the absence of flow. The results agreed well with experimental results, as well as with a parallel work in finite elements [38]. However, the noise pressure exit condition used by White gave rise to a moderate variation from the other results at the exit. By replacing the plane wave exit condition used by White with a far field, an improvement might be possible.

The idea of a far field exit condition is not new, and is usually used when solving conservation equations in infinite domains, such as in jet acoustics, external aerodynamics, external ballistics, seismology, and laser

fusion. The main advantage behind the use of far fields lies in their ability to simulate infinite domains, and thus eliminate the need for extensive computations over larger regions. A far field exit condition also has some very attractive computational features. It reduces substantially the "numerical" reflections when compared to other artificial boundaries, and it is a local boundary condition by nature [39]. In 1961, Karp [40] proved convergence of a far field expansion for two dimensional radiation functions. The expansion he developed was based on $1/r$, where r is the distance from a fixed point. In general, one requires that when the far field boundary condition is approximated by (one-sided) finite differences, the accuracy to these derivatives be no less than one order of accuracy less than the accuracy of the interior points [41]. The problem with the previous requirement is that it tends to introduce stability and accuracy difficulties since many mesh points normal to the boundary are involved. Bayliss and Turkel [42] found that for irregular regions, a first order accurate far field exit condition was greatly inferior to a second order condition. They also indicated that the one-dimensional characteristic exit condition was not useful, and tended to give rise to false numerical reflections when used with a far field geometric shape.

This work sought to develop grid mapping procedures and numerical techniques required for the simulation of noise propagation in variable area ducts. The techniques developed are also capable of handling soft wall boundary conditions as well as satisfying far field radiation conditions. In order to simulate the noise propagation in variable area ducts, two major steps were undertaken. First, the irregular geometry in the physical (X-Y) plane, shown in figure (1) (all figures are in Appendix (F)), was transformed into a square grid in the (ξ-η) plane, shown in figure (2), using the general numerical mapping procedure already developed by White [37]. Second, the noise equations in the transformed plane were solved using an explicit finite difference technique. In the following chapters, the mapping and acoustic numerical procedures will be outlined.

CHAPTER II

MAPPING OF THE FAR FIELD

In order to take full advantage of the numerical characteristics of the far field, a general mapping was performed. The additional curvatures that the far field introduced presented a new test case for White's general grid mapping procedure [37], and successfully handling the variable area duct with the far field attachement would greatly enhance its generality. Since the mapping equations contain time derivatives of order one (see Appendix (A), equations (A-8)), an initial condition for the variable area duct with the far field had to be computed. Due to the complexity of the far field shape, it was divided into three regions (see figure (3)), where special attention was used to join the boundary segments in adjacent regions with equal grid spacings. In doing so, the continuity of the grid increment dimensions along the upper and bottom surfaces was preserved. The difficulty in matching grid spacings when joining two regions of different lengths and number of grid points was overcome with the use of "one-sided" and "two-sided" geometric progressions. Region (II) was calculated first, and then with known starting grid divisions, boundary segments "d" and "h" (see figure (3)) were constructed using a one-sided progression. Since region (I) is

surrounded by the variable area duct on one side and region (II) on the other side, segments "b" and "f" each had to satisfy, in general, two different grid spacings, hence the need for a two-sided geometric progression. A detailed and comprehensive presentation of the above mentioned progressions can be found in Appendix (E). Figure (4) shows the initial condition of a variable area duct with a far field attachement. In the transverse direction, 11 grid points are equally spaced over the inlet boundary and around the far field circular boundary. In the axial direction, the variable area duct and the three regions forming the far field contain 11 grid points each. The initial computational domain described above was then solved using the procedure outlined in Appendix (A), and is shown in figure (5). It can be seen that the coordinate lines are smooth and no sharp discontinuities are present.

In the case where more grid points are needed next to areas of steep gradients, the number of grid points could be increased in the axial and/or transverse directions. Figures (6) and (7) show the initial grid and its final solution, respectively, where the number of grid points was doubled in the transverse direction. However, the previous solution was twice as expensive as the one in figure (5). A less expensive technique would be to keep the same number of grid points in the transverse direction

and replace the uniformly spaced grids on the "inlet" and "exit" boundaries (segments "k" and "l" in figure (3)) with a one-sided geometrically progressed string of grid points, where the smallest division is placed next to the upper boundary which contains the curved segments. The above described computational grid was successfully mapped, and figures (8) and (9) show the initial grid distribution and its final solution, respectively.

The ease with which the grid resolution can be enhanced is clearly one of the main advantages of this general grid mapping technique over the other schemes [31,32,36] previously described in the introduction. The weighting parameters used in the first two references can be very impractical to work with and the control terms in the third reference result in an unnecessary added complexity. Another main advantage of this general mapping method is the relatively unrestricted initial condition required in the solution of the mapping equations (equations (A-8), Appendix (A)). However, since every method must have limitations, this procedure was found to loose accuracy when a very small grid spacing was used in conjunction with a small number of grid points in the transverse direction, which resulted in a large progression ratio number (where the progression ratio is defined by $r=\Delta X(i+1)/\Delta X(i)$). Figures (10) and (11) show the initial condition and the final solution,

respectively, of such a case.

However general, clear, and simple to use the presented general mapping procedure might seem, the analytical and numerical mathematics behind it are not. Although the general mapping procedure is not too difficult to describe in a qualitative way, the governing analytical and numerical framework on which it is based is quite complex. The complexity in the derivation of the final equations lies in the fact that the coordinates $(\xi-\eta)$ of the transformed plane are made solutions of a system of modified Laplace equations in the physical plane. The transformation into the new plane makes the solution to the physical coordinates that of a system of coupled quasi-linear elliptic equations. Factored implicit difference algorithms are well suited for the solution of such systems, but the coupling makes it imperative to solve the complete grid in both directions for every time step until steady state condition is achieved.

In the next chapter, the feasibility of using a simpler mapping procedure than the one presented by White [37] will be investigated.

CHAPTER III

AN ALTERNATE MAPPING PROCEDURE

In an effort to uncouple the grid mapping equations, a different approach than the one presented in the previous chapter was undertaken. The following approach is similar in concept to those of Barfield [31] and Amsden and Hirt [32]. The method takes the physical coordinates to be solutions of Laplace equations in the transformed plane. The grid mapping equations then appear as

$$\begin{aligned} X_{\xi\xi} + X_{\eta\eta} &= 0 \\ Y_{\xi\xi} + Y_{\eta\eta} &= 0 \end{aligned} \qquad (1)$$

Since the above equations are uncoupled, an accelerated relaxation procedure was used to solve the mapping of the physical grid shown in figure (1). The above procedure yielded a convergence time that was shorter than that of the factored implicit scheme described in the previous chapter. However, the solution had a crossover of coordinate lines, as it can be seen in figure (12). The "spill over" on the top boundary can be attributed to the fact that the Y-coordinate is not a constant, whereas in the case of the reverse procedure described in the previous chapter, each boundary constitutes a constant coordinate line (see figure (2)) in the transformed $(\xi - \eta)$ plane.

The crossover problem can be cured by slightly modifying Laplace equations (1) to appear as

$$X_{\xi\xi} + aX_{\eta\eta} = 0$$
$$Y_{\xi\xi} + bY_{\eta\eta} = 0 \qquad (2)$$

where a and b are weighting parameters that allow some control over the coordinate line spacings. However, as indicated in the introduction, the presence of the weighting parameters in the equations hurts the generality of the method as the search for the "right" weighting factors can be a tedious and impractical task. It can be further argued that any crossover behavior can be expected to worsen when more complex physical computational grids are involved as in the case of the variable area with a far field attachement (see figure (4)). It can be then concluded that the weighting-average iteration method cannot be used as a basis for a general grid mapping procedure. It should nevertheless be pointed out that had the method been applicable, it would have been easy to implement and relatively inexpensive to run on a digital computer. It would have also provided two advantages that other more complicated schemes lack: guaranteed quadratic convergence and unrestricted initial conditions.

CHAPTER IV

INVESTIGATION OF EXPLICIT AND IMPLICIT FINITE DIFFERENCE TECHNIQUES

The second step of this two-part scheme for the simulation of noise propagation in variable area ducts was to solve the pressure equations over the transformed computational grid (see figure (5)). The latter $(\xi - \eta)$ grid was generated using the general grid mapping procedure outlined in Chapter (II). It should be emphasized at this point that the transformation of the "irregular" physical shape into a square grid yields a computational grid in which the coordinate lines are uniformly spaced and thus ideally suited for finite difference methods.

Many finite difference techniques have been developed through the years, and they all vary in accuracy, ease of implementation and stability characteristics. It is very important to choose an appropriate method in order to insure that the technique used is the least expensive and easiest to implement for the particular problem at hand. However, in other cases where accuracy is a priority over financial considerations, a more sophisticated method might be best suited for the solution of the problem. In this work, it was imperative to investigate more than one technique to allow a firsthand search for the most

economical and least numerically cumbersome scheme possible. Such a task is not always easy and a fair amount of compromise is necessary. It is clearly one of the laws of nature that accuracy and sophistication are not usually cheap. Two methods were investigated in this work: explicit finite differences and the factored implicit finite difference scheme. The former technique is by far the least stable and usually less accurate. However, the explicit method is the most widely used when ease of numerical implementation is called for. It is most recognized for doing the job quickly, cheaply, and with moderate accuracy. However, it is often the case in engineering that the size and the technical level of the problems encountered require more sophisticated numerical schemes. The factored implicit finite difference technique certainly fits in the latter group of schemes. It possesses the advantage of joining under one scheme the unrestricted stability of the pure implicit difference method with the accuracy of the Crank-Nicolson difference technique. It also allows the uncoupling of the mechanical solver in such a way that the axial and the transverse grid scannings (in two-dimensional problems) are done sequentially and independently of each other for every time step. Also, the method blends extremely well with marching and transient difference techniques.

In comparing the two finite difference schemes mentioned above, the simple geometry of a hard wall, rectangular duct (see figure (13)) in the absence of flow and with a plane wave exit condition was studied. In this case, since all the boundary segments were orthogonal, the grid mapping scheme reached the grid solution in the transformed plane in only one iteration. The initial condition in the physical plane and the mapped solution in the tansformed plane were identical since the grid under investigation was a square (see figure (13)). The noise source was placed at the inlet of the duct ($\xi=0$), and allowed to radiate into the initially quiescent duct as time progressed. The exit boundary condition at $\xi=1$ was that of a plane wave, and the top and bottom ($\eta=0$ and $\eta=1$) boundary conditions were those of a hard wall.

The dimensionless continuity and momentum acoustic equations for the two-dimensional zero mean flow case are given respectively by

$$\begin{aligned} \omega P_t &= -u_x - v_y \\ \omega u_t &= -P_x \\ \omega v_t &= -P_y \end{aligned} \qquad (3)$$

Equations (3) are combined into a single wave equation which appears as

$$\omega^2 P_{tt} = P_{xx} + P_{yy} \qquad (4)$$

The boundary conditions for the two dimensional constant

area duct shown in figure (13) are given below.

Inlet (X=0): $P=\exp(i2\pi t)$ (5)

This is the noise source radiating into the initially quiescent duct.

Bottom (Y=0): $P_y=0$ (6)

Top surface: $P_n=0$ (7)

Here, n is the normal to the upper surface in the outer direction.

Exit ($X=L^*/H^*$): $\omega P_t+P_x=0$ (8)

L^* and H^* are the dimensional length and height of the physical duct, respectively. Equation (8) is an approximate plane wave exit condition that permits plane waves to leave the duct without reflections.

Using the geometric transformation equations (A-3) through (A-5) in Appendix (A), equations (4) through (8) are transformed into the $(\xi-\eta)$ plane.

Wave equation: $\omega^2 P_{tt}= (\alpha P_{\xi\xi}-2\beta P_{\xi\eta}+\gamma P_{\eta\eta})/J2$ (9)

Inlet ($\xi=0$): $P=\exp(i2\pi t)$ (10)

Bottom ($\eta=0$): $(\gamma P_\eta-\beta P_\xi)/J\sqrt{\gamma}=0$

But since $\beta=0$ on the bottom surface, the above equation becomes

$$(\sqrt{\gamma}/J)P_\eta=0 \qquad (11)$$

Top (η=1): $(\gamma P_\eta - \beta P_\xi)/J\sqrt{\gamma}=0$ (12)

Plane exit (ξ=1): $\omega P_t + (Y_\eta P_\xi - Y_\xi P_\eta)/J=0$ (13)

Equation (9) along with the appropriate boundary conditions were then put into second-order time-accurate finite difference form, and solved using the explicit and factored implicit methods described earlier. A complete derivation of the two methods as applied to the above system of equations can be found in Appendix (B).

The pressure in the duct was initialized to a complex value of (1.0,0.0) and the noise source was then "turned on". It was expected that the value of the absolute pressure would change throughout the duct, and later come back to 1.0 (the exact solution) when steady state conditions were reached. Certainly, it was also expected that due to the numerical approximations involved, the exact value of 1.0 would not be obtained, but rather an "approximate" 1.0 would be achieved at steady state. The results were nothing of a surprise, but the comparison of the two methods as applied to the same problem were greatly instructive. In order to best picture the behavior of the explicit and the factored implicit methods, the pressure at all the grid points in the duct was scanned at every time step, and the greatest deviator from the exact solution was recorded. It was then possible to plot the maximum deviation from 1.0 versus non-dimensional time. Two pairs of plots (figures

(14)-(17)) were obtained for two different sets of duct characteristics.

In figures (14) and (15), the time step is 0.004 and the number of grid points in the axial direction is 21. Both explicit (figure (14)) and factored implicit (figure (15)) solutions were run for 1000 time steps, which allows a wave leaving the source to travel twice the length of the duct. In order to progress from one time level to the next, the factored implicit scheme requires tridiagonal matrix inverses that take around six times the CPU time required for an explicit method on the same grid. In spite of this, and not to mention the added difficulties in derivation and implementation of the difference equations, it can be clearly seen from the plots in figures (14) and (15) how much of an advantage the factored implicit method holds over the explicit method in the area of stability and accuracy. Since implicit methods used for the solution of linear partial differential equations are generally unconditionally stable, the time step can be made much larger, resulting in a more competitive CPU time between the two methods. Perhaps the most interesting and instructive piece of information that can be acquired from figures (14) and (15) is the difference in the manner with which the two methods handle the exit boundary condition. The factored implicit method smoothly integrates the exit boundary

condition into the solution of the pressure equation whereas the reaction of the explicit method can be described as "violent" and greatly unstable at first. However, as time progresses, the explicit method regains "control" and the exit boundary condition is finally assimilated in the numerical process. It can also be noticed that as the factored implicit scheme steadily reaches a maximum deviation position and maintains it, the explicit scheme is able to readjust itself to another maximum deviation position. Nevertheless, it should be pointed out that the ability of the factored implicit method to smoothly integrate the exit boundary condition should not be looked at as an inability to react to changes in the duct conditions. Further light can be shed on this beneficial characteristic by taking a closer look at the numerical nature of the two different schemes. When the explicit method is used to calculate the new pressure value at a grid location (i,j) at the new time level ($t+\Delta t$), the pressure value of the same grid (i,j) at the current time level (t) is used, as well as, all its neighbors (i+1,j ; i-1,j ; i,j+1 ; i,j-1) at time level (t). On the other hand, in the case of the factored implicit method, the neighbors of a grid point at the new time level ($t+\Delta t$) are involved. The latter fact not only provides the scheme with numerical damping but most importantly enables the method to "forsee" upcoming

changes in conditions and adapt to them accordingly. It should further be stated that when the exit boundary condition was not handled smoothly by the explicit method, some reflections were created off the approximate plane wave exit condition showing up as unsteadiness. Such behavior, already apparent in the case of a simple hard wall rectangular geometry can be expected to worsen when more complicated geometric and/or acoustic conditions are introduced. It should also be pointed out that both schemes have the same formal second-order time accuracy, and it can be seen from figures (14) and (15), that the "steady state" error for the explicit method is of the same order of magnitude as that of the factored implicit scheme, but somewhat larger. In order to better compare the two finite difference methods as applied to the zero flow hard wall straight duct case, the number of grid points in the X-direction was doubled while the time step was cut in half. As expected, the accuracy of both schemes improved by a slight amount (see figures (16) and (17) which present a plot of maximum "local" deviation versus time, similarily to figures (14) and (15)). Figures (16) and (17) do not contain any new information except in supporting the previous discussion pertaining to figures (14) and (15). However, they do emphasize the fact that although the accuracy improved in both cases, the explicit method would require twice the number of the

time steps to reach "steady state" in order to satisfy numerical stability restrictions, whereas the factored implicit scheme need not be run any longer. It can be further argued that since the pressure solution in the duct bearly changes after the wave reaches the exit, the scheme could be run for only one characteristic length and still maintain good accuracy. The characteristic time of a duct is defined in this work as being the non-dimensional time it takes the acoustic wave to travel the distance from the inlet noise source to the exit of the duct.

In spite of the strong case built for the use of factored implicit schemes, it should be pointed out that when the relative ease of development and implementation is taken into consideration, the explicit method remains a close and tough competitor to the more sophisticated factored implicit schemes. It is rightfully considered "good practice" in scientific research to use the most sophisticated and more expensive techniques only when the easier and less expensive schemes have failed or proven inadequate. The research work presented in this thesis is part of a three-year scientific effort in the simulation of noise propagation in variable area ducts, and so it was decided to initially use the explicit methods, hoping they would prove adequate for the problem at hand.

CHAPTER V

NUMERICAL SOLUTION OF THE PRESSURE FIELD

In order to achieve a numerical simulation of noise propagation in variable area ducts, it was first necessary to choose the best available grid mapping procedure and use it to map the irregularily shaped physical domain into a convenient square grid. It was then necessary to choose the most appropriate finite difference method in order to solve the acoustic field on the computed computational grid. As pointed out in the previous chapter, it was decided to initially use the explicit finite difference method in the last stage of this simulation. This chapter will be devoted to developing explicit numerical algorithms in the transformed plane ($\xi-\eta$) appropriate for the solution of noise propagation in variable area ducts, with or without acoustic liners. Both plane wave and far field exit conditions will be treated and compared.

The same wave equation (equation 4) introduced in the previous chapter will be solved for more complex physical domains and boundary conditions. The following are the wave equation and the boundary conditions for the variable area duct geometry in the physical (X-Y) plane.

Wave equation: $$\omega^2 P_{tt} = P_{xx}+P_{yy} \qquad (14)$$

Inlet (X=0): $$P=\exp(i2\pi t) \qquad (15)$$

This is the noise source radiating into the initially

quiescent duct.

Bottom (Y=0): $(\omega/\zeta)P_t-P_y=0$ (16)

The bottom wall can be a line of symmetry, a hard wall, or a soft wall. In both the symmetry and hard wall cases, we have $\zeta=\infty$.

Top surface: $(\omega/\zeta)P_t-P_n=0$ (17)

Here, n is the normal to the upper surface in the outer direction. In the case of a hard top wall, $\zeta=\infty$.

Exit ($X=L^*/H^*$): $\omega P_t+P_x=0$ (18)

L^* and H^* are the dimensional length and height of the physical duct, respectively. Equation (18) is an approximate exit condition that permits plane waves to leave the duct without reflections. In theory, the amount of reflections can be decreased by adding straight sections between the variable area duct and the exit condition (18). In the case where a far field is added to the exit of the variable area duct, as in figure (5), the exit condition [40] becomes

(at r=R) $\omega P_t+P_r+P/2r=0$ (19)

where r and R are defined in figure (18).

Using equations (A-3) through (A-5), as well as the following transformation (20), equations (14) through (19) are transformed into the (ξ-η) plane.

$$f_r = [(X-X_o)(Y_\eta f_\xi - Y_\xi f_\eta) + Y(X_\xi f_\eta - X_\eta f_\xi)]/rJ \qquad (20)$$

X_o in (20) is defined in figure (18), and all the other terms have been previously introduced.

Wave equation: $\omega^2 P_{tt} = (\alpha P_{\xi\xi} - 2\beta P_{\xi\eta} + \gamma P_{\eta\eta})/J^2$ (21)

Inlet (ξ=0): $P = \exp(i2\pi t)$ (22)

Bottom (η=0): $(\omega/\zeta)P_t - (\gamma P_\eta - \beta P_\xi)/J\sqrt{\gamma} = 0$ (23)

Top (η=1): $(\omega/\zeta)P_t + (\gamma P_\eta - \beta P_\xi)/J\sqrt{\gamma} = 0$ (24)

Plane exit (ξ=1): $P_t + (Y_\eta P_\xi - Y_\xi P_\eta)/J = 0$ (25)

Far field (ξ=1): $\omega P_t + P_\xi[(X-X_o)Y_\eta - YX_\eta]/RJ + P_\eta[YX_\xi - (X-X_o)Y_\xi]/RJ + P/2R = 0$ (26)

Equation (21) along with the appropriate boundary conditions were then put into second-order time-accurate finite difference form. The complete derivation of the explicit finite difference equations is outlined in Appendix (C).

The first case to be studied was that of a variable area duct with hard top and bottom walls and a plane wave exit condition. The results were well in agreement with the data obtained from the parallel experimental work being conducted by Dr. K.J. Baumeister at NASA Lewis Research Center (hereafter referred to as Baumeister's

results), as well as the finite element work of W. Eversman and R.J. Astley (hereafter referred to as Eversman-Astley's results). Figure (19) shows the solution of the above problem as a plot of the absolute pressure at the bottom wall versus the non-dimensional axial length of the duct. As it can be seen from the plot, the maximum deviation from Baumeister and Eversman-Astley's results is at the exit of the duct, which can be attributed to the reflecting waves unable to numerically "leave" the duct through the plane wave approximate condition. As stated earlier, a plane wave boundary condition is expected to introduce wave reflections back into the duct when "skewed" waves hit the exit. Use of the plane wave exit condition is thus appropriate only in cases where the plane wave exit is physically realistic. This is not to imply, however, that the pressure profile at each axial location must be plane.

It should be pointed out at this time that in all the cases studied in this work, the dimensionless frequency of the waves originating from the noise source were chosen such that the wave mode inside the duct was set to a plane one. All the frequencies were chosen below the "cutoff" frequency so that the wave originating at the source retained its strength throughout the duct. Waves with a frequency above the cutoff frequency would change shape, diffuse, and decay inside the duct. However, due to the

numerical approximations in the finite difference acoustic wave equation and boundary conditions, reflections can be, and usually are, numerical in nature whether the wave in the vicinity of the duct exit is plane in nature or not. It is therefore impossible to completely avoid reflections except by going to higher order difference equations which naturally are more tedious to develop and more expensive to solve.

Figure (20) shows a plot of the percent deviation between the pressure levels on the bottom wall and all the other grid points in the transverse direction at two fixed axial positions (X=0.3 and X=0.7). When the results of figure (20) are compared to Baumeister's experimental data and Eversman-Astley's finite element results, they are found to be in agreement. Such agreement in results from three fundamentally different approaches contributes to building a greater confidence in the numerical approach and its program implementation, so that later, more complex geometries and boundary conditions can be studied with the same computer code without having to run parallel tedious and expensive experiments.

It is often the case in jet engines and mufflers that the variable area duct is not a stand-alone component but rather a part of a more complex system that may very well be schematized as a straight inlet section followed by the variable area duct which in turn is followed by a straight

exit section. Earlier, the presence of a straight exit section was said to help reduce the presence of reflections in the duct. Straight section additions, however, increase the computational time drastically. The physical domain whose initial condition and final grid solution are shown in figures (21) and (22), respectively, was formed of a variable area duct with an inlet straight piece three times its length, and an exit straight section twice its length. The top and bottom walls were hard and the noise source was placed at the inlet of the straight inlet section. The problem was then run until "steady state" was reached throughout the whole duct system. The pressure results were plotted in figure (23) versus the axial position along the bottom surface of the duct. Baumeister's experimental data and Eversman-Astley's numerical results were plotted on the same figure for the purpose of comparison. The plot clearly shows all methods to be in excellent agreement, except in the straight exit duct (which the finite element solution did not cover) where the experimental results show a greater waiveness not present in the numerical finite difference solution. The test section blockage is seen to produce standing waves in the upstream inlet section. It is believed that some physical reflections were present in the experimental duct apparatus, and in all cases, the discrepency is too small to have any important influence on the total

performance of any of the three methods of analysis. Figure (23) was published in reference [38].

The next study case concerned itself with uniform area duct with a plane wave exit condition. Such a problem would provide the simulation presented in this work with a test of its capabilities to bring soft wall boundary conditions into the general solution. A specific acoustic impedance of value $\zeta = 0.16 - i(0.34)$ was imposed on the upper wall (due to symmetry, this problem is identical to a bottom soft wall case.) and a dimensionless frequency value $\eta = 0.3$ (see nomenclature for equation). The initial condition and final solution of the computational grid are shown in figure (24). The above problem was run until "steady state" was reached. The resulting absolute pressure values were plotted versus axial position along the bottom wall, as shown in figure (25). The other data plotted in figure (25) was obtained from figure (6) in reference [24], and used for comparison. The results of the current finite difference simulation method clearly match the predicted solution from the two other techniques.

Next, a variable area duct with a plane wave exit condition and a soft wall impedance imposed on the bottom boundary was studied ($\zeta = 0.085 - i(0.16)$ and $\eta = 0.125$). The solution was plotted as before on an "absolute pressure" versus distance "X" graph (see figure (26)).The results

appear to be reasonable but no other data was available for comparison. No additional information can be obtained from the above plot until future parallel experimental and/or numerical work is conducted in this area. All the previous test cases had a plane wave exit condition which was found to give rise to some reflections back into the duct.

In real life situations, it is desirable to be able to measure the pressure not only inside the duct but outside of it, as in the case of noise levels around jet engines and car mufflers. The addition of a far field computational grid at the exit of a general variable area duct would be expected to not only allow the measurement of noise behind and around the variable area but olso provide a theoretically better boundary condition, as it was pointed out in Chapter (I).

The first duct to be run with a far field exit condition was a straight duct test section with hard, top and bottom wall and no inlet or exit straight section. The physical grid was first initialized (figure (27)) and then solved to obtain the final computational grid solution (figure (28)). The non-dimensional parameters were chosen such that the far field's size was moderately larger than the duct's characteristic height. The length of the test section was 2.0, and its height was equal to 1.0. The far field had an exit radius of 3.0, a knee

radius of 0.25 and a total length of 4.25 from the end of the test section to the exit of the far field. The above problem was then run until "steady state" was reached. The resulting plot from the obtained solution is shown in figure (29). The numerical results inside the test section did not agree with those of the exact analytical solution even though the solution on the bottom surface appeared qualitatively correct in the far field region. One possible explanation for the observed discrepency could be the presence of reflections due to the fact that the pressure at the far field boundary does not represent a purely cylindrical expansion which is assumed by the far field boundary condition. Also, erroneous pressure perturbations generated in the time dependent solution due to the type of finite difference method used tend not to disappear as the periodic steady state is approached. Instead, these perturbations persist as "steady state reflections" which can markedly alter the noise level distribution. Another source of problems was felt to be those mesh points matching unequal grid spacings. It was already observed in the previous chapter how the explicit method tended to react "violently" to sudden changes in the state of the computational grid. Furthur investigation allowed the author to observe a tendency for the solution to improve when the computational grid was made "smoother". Such behavior, however, cannot be

allowed to take precedence over the generality of the problem. In oher words, the simulation should not be permitted to become a strong function of minor variations in the shape of the coordinate lines forming the computational grid. It was decided to run another test case with a far field exit condition hoping to get a better insight of the problem at hand and shed a stronger light on the behavior of the genaral grid mapping and explicit finite difference simulation in the presence of complexly shaped far fields. The run made was identical to the previous one in all perspectives except that the straight top wall of the test section was exchanged with a curved line described, as in all previous variable area ducts, by the following fourth degree polynomial

$$Y = Z(16(X/L)^2 - 32(X/L)^3 + 16(X/L)^4) \qquad (27)$$

with a maximum depth equal to half the height of the duct. The initial condition and the final solution of the computational grid have already been intrduced in figures (4) and (5), respectively. The acoustic solution was plotted (see Figure (30)), and the results for the variable area section were found to disagree with the results of the same variable area duct with a plane wave exit condition. The results in the far field section, however, looked qualitatively reasonable and the pressure level certainly decreased in the direction of the exit as

expected. The pressure level around the far field circular exit was found to decrease steadily for larger angles which were measured counterclockwise from the positive X-axis. Such behavior was expected and is correct and encouraging.

From the two previous cases, it became obvious that the general grid mapping in conjunction with the explicit difference method were not best suited for the solution of variable area ducts with complexly shaped far fields and nonuniformly spaced physical grids. It was felt at that stage of the research that if the simulation technique was to be kept general and independent of grid spacings, slope changes, and especially grid mapping results, a more sophisticated and stable transient finite difference technique, such as the factored implicit method, should be used. Such general grid mapping-factored implicit scheme combination would be expected to provide an overall better simulation performance in the presence of far fields and more complex variable area ducts. An example of that would be the simulation of a real jet engine which would have a non circular, more realistic knee curvature, as well as a center body piece at the inlet (see figure (31)).

CHAPTER VI

CONCLUSIONS AND RECOMMENDATIONS

The FORTRAN code (called NOISE) was constructed in this work for the simulation of noise propagation in variable area ducts, including the special case of an uniform area duct, with hard and/or soft wall conditions, and a choice of two exit conditions. NOISE has wide capabilities and the possible combinations that can be studied are literally infinite. However, the aim of this work was to develop the code, debug it, and test it for a few cases with preferably other experimental and/or numerical results available. The test cases studied in this work certainly do not cover all the possible combinations of the soft wall impedances, acoustic properties, and geometric dimensions.

Once sufficient confidence in this computer program is obtained, its capabilities can then be fully used with minimum restrictions. Some problem areas were discovered as described in the previous chapter but it is concluded that not only were the objectives satisfied for the second year of this three-year research effort but also enough research and testing was done to lay out the proper route to take in next year's work. The proposal for the third year of research is presently being reviewed by NASA and word of the funding is expected soon. At this point, it

would probably be useful to summarize the steps discussed in this work.

First, the acoustic far field was included in the numerical mapping procedure already developed by White [37], thus enabling the technique to apply an alternate more general exit condition and provide acoustic radiation patterns far from the exit of the duct. The mapping with the new far field addition was tested and found successful.

Second, the feasibility of using a possibly more efficient mapping technique than the one presented in this work was investigated. It was found that the complexity of the geometric shapes encountered in this work prohibited the use of a less sophisticated mapping technique. No attempt to cure the problem of crossover of coordinate lines in the alternate mapping procedure was undertaken since it was felt that any possible remedy would abate the generality of the mapping and introduce restrictions and difficulties.

Third, the explicit and factored implicit finite difference methods were studied and their behaviors were compared in the solution of the wave equation for the simple geometry of a square duct. Their performances were found to differ widely and even though the factored implicit scheme proved to be more stable and somewhat more accurate than the explicit method, the latter was chosen

for its simplicity. However, the factored implicit scheme should be useful in "follow-up work" such as noise propagation in ducts with flow. In that case, the acoustics are governed by a system of coupled partial differential equations.

Fourth, explicit numerical algorithms in the transformed plane were developed for the solution of noise propagation in variable area ducts, with and without acoustic liners. Two exit conditions were treated and compared to each other and to available experimental and/or numerical results. It was found that the plane wave exit condition gave very good results in most cases and the cylindrical far field exit condition gave erroneous results. Three possible problem areas were identified as being the failure of the explicit method to eliminate numerically generated erroneous "steady state reflections", the circular shape of the far field, and the grid spacing variation when matching subregions in the duct. The apparent sensitivity of the explicit finite difference method to the shape of the coordinate lines in the physical plane made the scheme unsuitable for the calculation of the pressure field in complex geometries. The second problem arose from the fact that the pressure wave did not expand exactly cylindrically in two dimensions. Because of this, numerical reflections occured at the far field exit boundary.

Last, several test cases were run for the mapping procedure and the explicit pressure algorithm. The numerical results obtained were then compared with known noise propagation analytical solutions and experimental data. This included hard wall and soft wall boundary conditions in two dimensional variable area geometries. The work summarized above resulted in a general computer program, NOISE, whose listing is in Appendix (E).

The central importance of this work was that it presented an accurate and efficient method of satisfying the acoustic boundary conditions on surfaces with strong curvature and/or slope discontinuity, without large matrix storage requirements. The accuracy and the applicability to a wider range of applications would have been further improved if the inclusion of the far field boundary condition had been fully successful on the first attempt.

Recommendations for next year's research include the further investigation of the far field, so that full advantage could be taken from its previously stated characteristics, and the use of the implicit methods in place of the less sophisticated explicit techniques. The incorporation of the implicit algorithm into the present procedure should provide the solution with improved accuracy and stability and a better handling of grid spacing variations in the physical domain. In the calculation of time dependent acoustics, it is very

important to minimize the generation of erroneous numerical perturbations, because these perturbations tend to remain in the solution as "steady state reflections", as evident in the results of figures (29) and (30).

Since it was felt that the pressure wave was unable to take the exact physical shape of the far field exit boundary, a good approach to resolve this problem could be to let the boundary itself take the shape of the pressure wave. In what follows is a basic and qualitative description of the general steps that would be followed to achieve a geometric correspondence between the physical exit boundary and the pressure wave. This method was described recently by White [45].

The proposed procedure is a free surface method that would look for a constant pressure boundary contour satisfying the following condition:

$$\omega P_t + P_n = 0 \qquad (28)$$

where $\vec{n} = -(\nabla P/|\nabla P|)$ and is perpendicular to the boundary. The search for the constant pressure boundary will be iterative in nature due to the nonlinear relationship between the boundary shape and the pressure wave contour at the exit of the far field.

The iterative scheme will first use an initial far field boundary shape to obtain the mapping of the initial computational grid. The steady state pressure profile

would then be computed. The unit vector $\vec{n}$ would then be obtained at each grid point location on the far boundary.

$$\vec{n}_i = -(\nabla P_i \; / \; |\nabla P|_i \;) \tag{29}$$

This will enable the technique to compute the shape of the new exit boundary. The X-coordinate of each of the mesh points on the far field boundary will be held constant while the Y-coordinate of each mesh point is located by numerically integrating

$$Y_{i+1} = Y_i - \int_{X_i}^{X_{i+1}} \tan(\theta) \; dX \tag{30}$$

where θ is the angle that the contour makes with the horizontal at a local mesh point. This same procedure will be repeated until the far boundary is coincident on successive calculations. Since most of the computational grid will remain unchanged from one calculation to the next, the grid mapping at each stage of the far boundary relocation is expected to be short and inexpensive. When successfully implemented, this technique should accurately account for the fact that in the vicinity of the duct exit, the pressure expansion is not exactly cylindrical.

In summary, the incorporation of the implicit method and the free surface technique is expected to introduce a greater stability into the pressure simulation, and a better accuracy at the exit of the far field.

LIST OF REFERENCES

LIST OF REFERENCES

1. Petyt, M., Lea, J., and Koopmann, G.H., "A Finite Element Method for Determining the Acoustic Modes of Irregular Shaped Cavities," Journal of Sound and Vibration, 45, 495-502 (1976).

2. Joppa, P.D., and Fyfe, I.M., "A Finite Element Analysis of the Impedance Properties of Irregular Shaped Cavities With Absorptive Boundaries," Journal of Sound and Vibration, 56, 61-69 (1978).

3. Nefske, D.J., and Howell, L.J., "Automobile Interior Noise Reduction Using Finite Element Methods," Paper No. 780365, presented at SAE Congress and Exposition, Detroit, Michigan, (February 1978).

4. Richards, T.L., and Jha, S.K., "A Simplified Finite Element Method for Studying Acoustic Characteristics Inside a Car Cavity," Journal of Sound and Vibration, 63, 61-72 (1979).

5. Unruh, J.F., "A Finite Element Subvolume Technique for Structural-Borne Interior Noise Prediction," AIAA Paper No. 79-0585, presented at AIAA 5th Aeroacoustics Conference, Seattle, Washington, (March 12-14, 1979).

6. Alfredson, R.J., "A Note on the Use of the Finite Difference Method for Predicting Steady State Sound Fields," Acoustica, 28 (5), 296-301 (May 1973).

7. Baumeister, K.J., and Bittner, E.C., "Numerical Simulation of Noise Propagation in Jet Engine Ducts," NASA TN D-7339 (1973).

8. Baumeister, K.J., and Rice, E.J., "A Difference Theory for Noise Propagation in an Acoustically Lined Duct With Mean Flow," in Aeroacoustics: Jet and Combustion Noise; Duct Acoustics, edited by Nagamatsu, H.T., O'Keefe, J.V., and Schwartz, I.R., Progress in Astronautics and Aeronautics Series, Vol. 37 (AIAA, New York, 1975), pp. 435-453.

9. Baumeister, K.J., "Time Dependent Difference Theory for Noise Propagation In Jet Engine Ducts," AIAA Paper No. 80-0098, presented at AIAA 18th Aerospace Sciences Meeting, Pasadena, California, (January 14-16, 1980).

10. Quinn, D.W., "A Finite Element Method for Computing Sound Propagation in Ducts Containing Flow," AIAA Paper No. 79-0661 (March 1979).

11. Gerald, C.F., Applied Numerical Analysis, (Addison-Wesley, Reading, MA., 1978), 2nd edition.

12. Quinn, D.W., "A Finite Difference Method for Computing Sound Propagation in Nonuniform Ducts," AIAA Paper No. 74-130 (January 1975).

13. Quinn, D.W., "Attenuation of Sound Associated With a Plane Wave in a Multi-Sectional Duct," in Aeroacoustics: Duct Acoustics, Fan Noise and Control Rotor Noise, edited by Schwartz, I.R., Nagamatsu, H.T., and Strahle, W., Progress in Astronautics and Aeronautics Series, Vol. 44 (AIAA, New York, 1976), pp. 331-345.

14. Craggs, A., "A Finite Element Model for Rigid Porous Absorbing Materials," Journal of Sound and Vibration, 61 (1), 101-111 (1978).

15. Craggs, A., "Coupling of Finite Element Acoustic Absorption Models," Journal of Sound and Vibration, 66 (4), 605-613 (1979).

16. Tag, I.A., and Lumsdaine, E., "A Finite Element Approach to the Problem of Noise Propagation and Attenuation in Industrial Duct," Paper presented at the ASME Annual Meeting, San Francisco, California, (December 10-15, 1978).

17. Tag, I.A., and Akin, J.E., "Finite Element Solution of Sound Propagation in Variable Area Duct," AIAA Paper No. 79-0663 (March 1979).

18. Baumeister, K.J., "Analysis of Sound Propagation in Ducts Using the Wave Envelope Concept," NASA TN D-7719 (1974).

19. Baumeister, K.J., "Wave Envelope Analysis of Sound Propagation in Ducts With Variable Axial Impedance," in Aeroacoustics: Duct Acoustics, Fan Noise and Control Rotor Noise, edited by Schwartz, I.R., Nagamatsu, H.T., and Strahle, W., Progress in Astronautics and Aeronautics Series, Vol. 44 (AIAA, New York, 1976), pp. 451-474.

20. Baumeister, K.J., "Numerical Spatial Marching Techniques for Estimating Duct Attenuation and Source Pressure Profiles," in 95th Meeting of the Acoustical Society of America, Providence, R.I., (May 16-19,

1978), also published as NASA TM-78857 (1978).

21. Baumeister, K.J., "Numerical Spatial Marching Techniques in Duct Acoustics," Journal Acoust. Soc. Am., 65, 297-306 (1979).

22. Baumeister, K.J., and Majjigi, R.K., "Applications of the Velocity Potential Function to Acoustic Duct Propagation and Radiation From Inlets Using Finite Element Theory," AIAA Paper No. 79-0680 (March 1979).

23. Baumeister, K.J., "Time dependent Difference Theory for Noise Propagation in Jet Engine Ducts," AIAA Paper No. 80-0098 (January 1980).

24. Baumeister, K.J., "A Time Dependent Difference Theory for Sound Propagation in Ducts With Flow," in 98th Meeting of the Acoustical Society of America, Salt Lake City, 1979 (also NASA TM 79302).

25. Baumeister, K.J., "Time Dependent Difference Theory for Sound Propagation in Axisymmetric Ducts With Flow," AIAA Paper No. 80-1017 (June 1980).

26. Chen, L.T., and Caughey, D.A., "Calculation of Transonic Inlet Flowfields Using Generalized Coordinates," Journal of Aircraft, 17, 167-174 (March 1980).

27. Chen, L.T., and Caughey, D.A., "Transonic Inlet Flow Calculations Using a General Grid-Generation Scheme," Journal of Fluids Engineering, 102, 309-315 (September 1980).

28. Thompson, J.F., Thames, F.C., and Mastin, C.W., "Boundary-Fitted Curvilinear Coordinate Systems for Solution of Partial Differential Equations on Fields Containing Any Number of Arbitrary Two-Dimensional Bodies," NASA CR-2729 (1977).

29. Winslow, A.J., "Numerical Solution of the Quasi-Linear Poison Equation in a Non-Uniform Triangular Mesh," Journal of Computational Physics, 2, 149 (1966).

30. Chu, W.H., "Development of a General Finite Difference Approximation for a General Domain, Part I: Machine Transformation," Journal of Computational Physics, 8, 392 (1971).

31. Barfield, W.D., "An Optimal Mesh Generator for Lagrangian Hydrodynamic Calculations in Two Space Dimensions," Journal of Computational Physics, 6, 417

(1970).

32. Amsden, A.A., and Hirt, C.W., "A Simple Scheme for Generating General Curvilinear Grids," Journal of Computational Physics, 11, 348 (1973).

33. Thompson, J.F., Thames, F.C., and Mastin, C.W., "Automatic Numerical Generation of Body-Fitted Curvilinear Coordinate System for Fields Containing Any Number of Arbitrary Two-Dimensional Bodies," Journal of Computational Physics, 15, 299 (1974).

34. Thames, F.C., Thompson, J.F., and Mastin, C.W., "Numerical Solution of the Navier-Stokes Equations for Arbitrary Two-Dimensional Airfoils," Proceedings of NASA Conference on Aerodynamic Analyses Requiring Advanced Computers, Langley Research Center, NASA SP-347 (1975).

35. Thompson, J.F., Thames, F.C., Mastin, C.W., and Shanks, S.P., "Numerical Solutions of the Unsteady Navier-Stokes Equations for Arbitrary Bodies Using Boundary-Fitted Curvilinear Coordinates," Proceedings of Arizona/AFOSR Symposium on Unsteady Aerodynamics, University of Arizona, Tucson, Arizona (1975).

36. Thompson, J.F., Thames, F.C., and Mastin, C.W., "TOMCAT- A Code for Numerical Generation of Boundary-Fitted Curvilinear Coordinate Systems on Fields Containing Any Number of Arbitrary Two-Dimensional Bodies," Journal of Computational Physics, 24, 274 (1977).

37. White, J.W., "A General Mapping Procedure for Variable Area Duct Acoustics," AIAA Paper No. 81-0094, presented at AIAA 19th Aerospace Sciences Meeting, St. Louis, Missouri, (January 12-15, 1981).

38. Baumeister, K.J., Eversman, W., Astley, R.J., and White, J.W., "Application of 'Steady' State Finite Element and Transient Finite Difference Theory to Sound Propagation in a Variable Area Duct: A Comparison With Experiment," Proposed for presentation at AIAA 7th Aeroacoustics Conference, Palo Alto, California, (October 5-7, 1981).

39. Engquist, B., and Majda, A., "Radiation Boundary Conditions for Acoustic and Elastic Wave Calculations," Communications on Pure and Applied Mathematics, 32, 313-357 (1979).

40. Karp, S.N., "A Convergent 'Farfield' Expansion for Two-Dimensional Radiation Functions," Communications on Pure and Applied Mathematics, 14, 427-434 (1961).

41. Gustafson, B., "The Convergence Rate for Difference Approximations to Mixed Initial Boundary Value Problems, Math. Comput., 29, 396-406 (1975).

42. Bayliss, A., and Turkel, E., "Radiation Boundary Conditions for Wave-Like Equations," Communications on Pure and Applied Mathematics, 33, 707-725 (1980).

43. Candel, S.M., "Acoustic Radiation From the End of a Two-Dimensional Duct, Effects of Uniform Flow and Duct Lining," Journal of Sound and Vibration, 28, 1-13 (1973).

44. Koch, W., "Radiation of Sound From a Two-Dimensional Acoustically Lined Duct," Journal of Sound and Vibration, 55, 255-274 (1977).

45. White, J.W., "A Coordinate Mapping Procedure For The Computation of Variable Area Duct Acoustics - Part Three," Proposal for the continuation of NASA grant NAG 3-18, October 1981.

APPENDIXES

APPENDIX A

GENERAL MAPPING EQUATIONS AND PROCEDURE

While the boundary grid points may be placed arbitrarily in the physical plane, the mapping locates them equally spaced on the corresponding segments in the transformed plane. This important feature of the mapping can allow a greater number of points around regions in the physical plane where the governing equations may give rise to steep gradients. The computations were performed over the uniformly spaced grid in the transformed plane where the finite difference approximations can be used with confidence. Since in the physical plane the interior points are not fixed, the mapping equations were solved to locate them. The system of elliptic partial differential equations defining the mapping is the following

$$\nabla^2 \xi = P(\xi, \eta)$$
$$\nabla^2 \eta = Q(\xi, \eta) \qquad \text{(A-1)}$$

The grid control parameters, P and Q, were neglected in the present work since they are mainly used in aerodynamic calculations in which the analysis is over multiply connected domains where some contour lines do not touch the boundary. Since the variable area ducts of interest in this study are simply connected domains, the control terms can be replaced by higher concentrations of grid points in

the steep gradient regions, when necessary. Equations (A-1) then become

$$\nabla^2\xi = 0$$
$$\nabla^2\eta = 0 \qquad \text{(A-2)}$$

The variables in equations (A-2) were then interchanged using

$$f_\xi = f_x X_\xi + f_y Y_\xi$$
$$f_\eta = f_x X_\eta + f_y Y_\eta \qquad \text{(A-3)}$$

Equations (A-3) are then solved for f_x and f_y to give

$$f_x = (f_\xi Y_\eta - f_\eta Y_\xi)/J$$
$$f_y = (f_\eta X_\xi - f_\xi X_\eta)/J \qquad \text{(A-4)}$$

where, $J = X_\xi Y_\eta - X_\eta Y_\xi$

Then the coordinates of the transformed plane can be differentiated using (A-4) to give

$$\xi_x = Y_\eta/J \qquad \xi_y = -X_\eta/J$$
$$\eta_x = -Y_\xi/J \qquad \eta_y = X_\xi/J \qquad \text{(A-5)}$$

The Laplacian operators would then appear as

$$J^3\nabla^2\xi = X_\eta[\alpha Y_{\xi\xi} - 2\beta Y_{\xi\eta} + \gamma Y_{\eta\eta}] - Y_\eta[\alpha X_{\xi\xi} - 2\beta X_{\xi\eta} + \gamma X_{\eta\eta}]$$

$$J^3\nabla^2\eta = -X_\xi[\alpha Y_{\xi\xi} - 2\beta Y_{\xi\eta} + \gamma Y_{\eta\eta}] + Y_\xi[\alpha X_{\xi\xi} - 2\beta X_{\xi\eta} + \gamma X_{\eta\eta}] \qquad \text{(A-6)}$$

where,
$$\alpha = X_\eta^2 + Y_\eta^2$$
$$\beta = X_\xi X_\eta + Y_\xi Y_\eta$$
$$\gamma = X_\xi^2 + Y_\xi^2$$

Combining equations (A-6) and (A-1) produces a system of coupled quasi-linear elliptic equations which is given by

$$\begin{aligned} \alpha X_{\xi\xi} - 2\beta X_{\xi\eta} + \gamma X_{\eta\eta} &= 0 \\ \alpha Y_{\xi\xi} - 2\beta Y_{\xi\eta} + \gamma Y_{\eta\eta} &= 0 \end{aligned} \qquad \text{(A-7)}$$

A damping time derivative is then added to equations (A-7) resulting in

$$\begin{aligned} \alpha X_{\xi\xi} - 2\beta X_{\xi\eta} + \gamma X_{\eta\eta} &= X_t \\ \alpha Y_{\xi\xi} - 2\beta Y_{\xi\eta} + \gamma Y_{\eta\eta} &= Y_t \end{aligned} \qquad \text{(A-8)}$$

The coupled system above is then solved in the transformed plane for the physical coordinates, X and Y, using a factored implicit finite difference scheme. The main steps of the numerical solution of equations (A-8) will now be outlined.

The factored scheme used is an alternating direction implicit scheme which exhibits spacial factoring and reduces to a series of tridiagonal matrix inversions which have to be performed at each time step as time progresses. Using Crank-Nicolson averaging, the finite difference form of equations (A-8) becomes

$$X^{k+1} - X^k = (\Delta t/2)[(\alpha X_{\xi\xi} - 2\beta X_{\xi\eta} + \gamma X_{\eta\eta})^{k+1}$$

$$+(\alpha X_{\xi\xi}-2\beta X_{\xi\eta}+\gamma X_{\eta\eta})^k]$$

$$Y^{k+1}-Y^k=(\Delta t/2)[(\alpha Y_{\xi\xi}-2\beta Y_{\xi\eta}+\gamma Y_{\eta\eta})^{k+1}$$
$$+(\alpha Y_{\xi\xi}-2\beta Y_{\xi\eta}+\gamma Y_{\eta\eta})^k] \qquad \text{(A-9)}$$

After rearranging equations (A-9) and neglecting fourth order derivatives equations (A-9) become

$$(1-\alpha(\Delta t/2)\delta_{\xi\xi})(1-\gamma(\Delta t/2)\delta_{\eta\eta})(X^{k+1}-X^k)=$$
$$[\alpha X^k_{\xi\xi}-2\beta X^k_{\xi\eta}+\gamma X^k_{\eta\eta}]\Delta t-\beta[X^k_{\xi\eta}-X^{k-1}_{\xi\eta}]\Delta t$$
$$(1-\alpha(\Delta t/2)\delta_{\xi\xi})(1-\gamma(\Delta t/2)\delta_{\eta\eta})(Y^{k+1}-Y^k)=$$
$$[\alpha Y^k_{\xi\xi}-2\beta Y^k_{\xi\eta}+\gamma Y^k_{\eta\eta}]\Delta t-\beta[Y^k_{\xi\eta}-Y^{k-1}_{\xi\eta}]\Delta t \qquad \text{(A-10)}$$

The coefficients (α,β,γ) in equations (A-10) are non-linear in nature and are treated explicitly in time to avoid added difficulties. The right-hand sides in equations (A-10) are made up of known quantities; say ϕ_1 and ϕ_2. Equations (A-10) then appear as

$$(1-L_\xi)(1-L_\eta)\Delta X^k=\phi_1$$
$$(1-L_\xi)(1-L_\eta)\Delta Y^k=\phi_2 \qquad \text{(A-11)}$$

where L_ξ and L_η are linear difference operators in the ξ and η directions, respectively, and

$$\Delta X^k=X^{k+1}-X^k$$
$$\Delta Y^k=Y^{k+1}-Y^k$$

When the following notations are introduced

$$(1-L_\eta)\Delta X^k=\Delta X^*$$

$$(1-L_{\eta})\Delta Y^{k}=\Delta Y^{*} \qquad \text{(A-12)}$$

where the (*) denotes an intermediate time level between (k) and (k+1), equations (A-11) become a system of one-demensional equations in the ξ-direction.

$$\begin{aligned}(1-L_{\xi})\Delta X^{*}&=\phi_1\\(1-L_{\xi})\Delta Y^{*}&=\phi_2\end{aligned} \qquad \text{(A-13)}$$

Each equation requires a tridiagonal matrix inversion to obtain the solution for ΔX^{*} and ΔY^{*}. Knowing these values, equations (A-12) can then be solved to obtain the values for X and Y at the new time step (k+1). This procedure is repeated to advance in time until the residual of motion of all the grid points is less than a maximum allowable value (ε). When the value (ε) is reached, the solution is said to have reached "steady state".

APPENDIX B

EXPLICIT AND FACTORED IMPLICIT FINITE DIFFERENCE EQUATIONS FOR A SQUARE DUCT

The finite difference equations for the various cells shown in figure (32) can be obtained by integrating the wave equation over the cell area. The wave equation being

$$\omega^2 P_{tt}=(\alpha P_{\xi\xi}-2\beta P_{\xi\eta}+\gamma P_{\eta\eta})/J^2 \qquad \text{(B-1)}$$

But since it is a square duct

$$J = \alpha = \gamma = 1 \qquad \text{and} \qquad \beta = 0$$

which reduces the wave equation to the following

$$\omega^2 P_{tt}=P_{\xi\xi}+P_{\eta\eta} \qquad \text{(B-2)}$$

Equation (B-2) is identical to the wave equation in the physical plane.

$$\omega^2 P_{tt}=P_{xx}+P_{yy} \qquad \text{(B-3)}$$

I. DERIVATION OF THE EXPLICIT DIFFERENCE EQUATIONS

$$\int_{-}^{+}\int_{-}^{+}(\omega^2 P_{tt}-P_{xx}-P_{yy})\,dx\,dy=0 \qquad \text{(B-4)}$$

where the plus sign (+) appearing in the upper limit of integration indicates the evaluation of the integrated parameters along either the upper or the right-hand

boundary of the integration cell. (All the considered integration cells are shown in figure (32) by dashed lines.) On the other hand, the negative sign (-) means to evaluate the parameters along the lower or the left-hand boundary of the cell. In both the plus (+) and minus (-) cases, the choice of the integration cell wall depends on whether x or y is considered.

$$\int_{-}^{+}\int_{-}^{+} \omega^2 P_{tt}\, dxdy = \int_{-}^{+}\int_{-}^{+} P_{xx}\, dxdy + \int_{-}^{+}\int_{-}^{+} P_{yy}\, dxdy \qquad \text{(B-5)}$$

Central Cell 1

$$(\omega^2/\Delta t^2)(P^{k+1}-2P^k+P^{k-1})\Delta x\Delta y=$$
$$\Delta y(P_x|^+-P_x|_-)+\Delta x(P_y|^+-P_y|_-) \qquad \text{(B-6)}$$

$$P^{k+1}-2P^k+P^{k-1}=(\Delta t/\omega)^2$$
$$[(1/\Delta x)((P_{i+1}-P_i)/\Delta x - (P_i-P_{i-1})/\Delta x)$$
$$+(1/\Delta y)((P_{j+1}-P_j)/\Delta y - (P_j-P_{j-1})/\Delta y)]^k \qquad \text{(B-7)}$$

$$P_{i,j}^{k+1}=2P_{i,j}^{k}-P_{i,j}^{k-1}+(\Delta t/\omega)^2$$
$$[(P_{i+1,j}-2P_{i,j}+P_{i-1,j})/\Delta x^2$$
$$+(P_{i,j+1}-2P_{i,j}+P_{i,j-1})/\Delta y^2]^k \qquad \text{(B-8)}$$

Upper Cell 2

Reconsidering equation (B-6), with $\int_{-}^{+} dy=\Delta y/2$ and $P_y|^+=0$ at the upper surface, it becomes

$$P_{i,j}^{k+1}=2P_{i,j}^{k}-P_{i,j}^{k-1}+(\Delta t/\omega)^2$$
$$[(P_{i+1,j}-2P_{i,j}+P_{i-1,j})/\Delta x^2$$
$$+2(P_{i,j-1}-P_{i,j})/\Delta y^2]^k \qquad \text{(B-9)}$$

Bottom Cell 3

Applying $\int_{-}^{+} dy=\Delta y/2$ and $P_y|_{-}=0$ at the wall, equation (B-6) becomes

$$P_{i,j}^{k+1}=2P_{i,j}^{k}-P_{i,j}^{k-1}+(\Delta t/\omega)^2 \left[(P_{i+1,j}-2P_{i,j}+P_{i-1,j})/\Delta x^2 +2(P_{i,j+1}-P_{i,j})/\Delta y^2\right]^k \qquad \text{(B-10)}$$

Exit Cell 4

Applying $\int_{-}^{+} dx=\Delta x/2$ and the plane wave exit condition $\omega P_t+P_x=0$ ($P_x|^{+}=-\omega P_t$), equation (B-6) becomes

$$(1+(\Delta t/\omega\Delta x))P^{k+1}=2P^{k}+((\Delta t/\omega\Delta x)-1)P^{k-1} +(\Delta t/\omega)^2 \left[2(P_{i-1,j}-P_{i,j})/\Delta x^2 +(P_{i,j+1}-2P_{i,j}+P_{i,j-1})/\Delta y^2\right]^k \qquad \text{(B-11)}$$

Upper Corner Cell 5

Applying $\int_{-}^{+} dx=\Delta x/2$, $\int_{-}^{+} dy=\Delta y/2$, $P_y|^{+}=0$, and $P_x|^{+}=-\omega P_t$ at the exit upper corner cell, equation (B-6) becomes

$$(1+(\Delta t/\omega\Delta x))P^{k+1}=2P^{k}+((\Delta t/\omega\Delta x)-1)P^{k-1} +2(\Delta t/\omega)^2 \left[(P_{i-1,j}-P_{i,j})/\Delta x^2 +(P_{i,j-1}-P_{i,j})/\Delta y^2\right]^k \qquad \text{(B-12)}$$

Lower Corner Cell 6

Applying $\int_{-}^{+} dx=\Delta x/2$, $\int_{-}^{+} dy=\Delta y/2$, $P_y|^{-}=0$, and $P_x|^{+}=-\omega P_t$ at the exit lower corner cell, equation (B-6) becomes

$$(1+(\Delta t/\omega\Delta x))P^{k+1}=2P^{k}+((\Delta t/\omega\Delta x)-1)P^{k-1} +2(\Delta t/\omega)^2 \left[(P_{i-1,j}-P_{i,j})/\Delta x^2\right.$$

$$+(P_{i,j+1}-P_{i,j})/\Delta y^2]^k \qquad \text{(B-13)}$$

II. DERIVATION OF THE FACTORED IMPLICIT EQUATIONS

Using Crank-Nicolson averaging, the wave equation (B-1) becomes

$$\omega^2 P_{tt}=(1/2)(\delta_{xx}P^{k-1}_{i,j}+\delta_{xx}P^{k+1}_{i,j}) +(1/2)(\delta_{yy}P^{k-1}_{i,j}+\delta_{yy}P^{k+1}_{i,j}) \qquad \text{(B-14)}$$

where δ_{xx} and δ_{yy} are second order finite difference operators. Letting $C=\Delta t^2/2\omega^2$, and rearranging equation (B-14)

$$(1-C\delta_{xx}-C\delta_{yy})P^{k+1}=2P^k-(1-C\delta_{xx}-C\delta_{yy})P^{k-1} \qquad \text{(B-15)}$$

Subtracting $(1-C\delta_{xx}-C\delta_{yy})P^k$ from both sides of equation (B-15) and neglecting fourth order mixed operators

$$(1-C\delta_{xx})(1-C\delta_{yy})\ P^k=2P^k -(1-C\delta_{xx}-C\delta_{yy})(P^{k-1}+P^k) \qquad \text{(B-16)}$$

where $\Delta P^k=(P^{k+1}-P^k)$.

Letting $\Delta P^*=(1-C\delta_{yy})\Delta P^k$ (where (*) indicates an intermediate time level between (k) and k+1)), there results

$$(1-C\delta_{xx})\Delta P^*=2P^k-(1-C\delta_{xx}-C\delta_{yy})(P^{k-1}+P^k) \qquad \text{(B-17)}$$

Equation (B-17) is then solved for ΔP^* using a tridiagonal matrix inversion. P^{k+1} is then computed from

$$(1-C\delta_{yy})\Delta P^k = \Delta P^* \tag{B-18}$$

Central Cell 1

For all the central cells,

$$\delta_{yy}S=(S_{i,j+1}-2S_{i,j}+S_{i,j-1})/\Delta y^2 \tag{B-19}$$

and

$$\delta_{xx}S=(S_{i+1,j}-2S_{i,j}+S_{i-1,j})/\Delta x^2 \tag{B-20}$$

where S is a general parameter.

Upper Cell 2

For all upper cells, $P_{i,j+1}=P_{i,j-1}$. δ_{xx} remains as in equation (B-20), while δ_{yy} becomes

$$\delta_{yy}S=2(S_{i,j-1}-S_{i,j})/\Delta y^2 \tag{B-21}$$

Bottom Cell 3

As in the case of upper cell 2, $P_{i,j-1}=P_{i,j+1}$. δ_{xx} remains unchanged and δ_{yy} becomes

$$\delta_{yy}S=2(S_{i,j+1}-S_{i,j})/\Delta y^2 \tag{B-22}$$

Central Exit Cell 4

Using Crank-Nicolson averaging, or integrating over the cell area, the wave equation (B-1) becomes

$$\omega^2 P^k_{tt}=(1/2)\delta_{yy}(P^{k-1}+P^{k-1})-(2\omega/\Delta x)P^k_t + (P^{k+1}_{i-1}-P^{k+1}_i)/\Delta x^2+(P^{k-1}_{i-1}-P^{k-1}_i)/\Delta x^2 \tag{B-23}$$

Letting $CA=\Delta t/\omega\Delta x$, $CB=\Delta t^2/2\omega^2$, and $CC=\Delta t^2/\omega^2\Delta x$, equation

(B-23) becomes

$$[1+CA-CB\delta_{yy}+CC\delta_x](P^{k+1}-P^k)=(P^k-P^{k-1})(1-CA)$$
$$+CB\delta_{yy}(P^{k-1}+P^k)-CC\delta_x(P^{k-1}+P^k) \qquad \text{(B-24)}$$

It is clear that the factorization of equation (B-24) can be almost impossible if left as it is. The key here is to divide the whole equation by (1+CA). Dividing by CD=1+CA, rearranging equation (B-24), and neglecting third order difference operators, there results

$$[1-(CB/CD)\delta_{yy}][1+(CC/CD)\delta_x]\ P^k=$$
$$(P^k-P^{k-1})[(1-CA)/CD]+(\Delta t^2/\omega^2 CD)$$
$$[\ (1/2)\delta_{yy}(P^k+P^{k-1})-(1/\Delta x)\delta_x(P^k+P^{k-1})\ ] \qquad \text{(B-25)}$$

In this case

$$\delta_{yy}S=(S_{i,j+1}-2S_{i,j}+S_{i,j-1})/\Delta y^2$$
$$\delta_x S=(S_{i,j}-S_{i-1,j})/\Delta x$$

Upper Corner Cell 5

Similarily to the previous case, the wave equation becomes

$$[1+(CC/CD)\delta_x][1+(CC/CD)\delta_y]\Delta P^k=$$
$$(P^k-P^{k-1})[(1-CA)/CD]-(CC/CD)[\delta_x P^k+\delta_x P^{k-1}]$$
$$-(\Delta t^2/\omega^2\Delta y CD)[\delta_y P^k+\delta_y P^{k-1}] \qquad \text{(B-26)}$$

where $\delta_x S=(S_{i,j}-S_{i-1,j})/\Delta x$

$\delta_y S=(S_{i,j}-S_{i,j-1})/\Delta y$

Bottom Corner Cell 6

The pressure wave equation for this case is

$$[1+(CC/CD)\delta_x][1-(CC/CD)\delta_y]\Delta P^k=$$
$$(P^k-P^{k-1})[(1-CA)/CD]-(CC/CD)[\delta_x P^k+\delta_x P^{k-1}]$$
$$+(\Delta t^2/\omega^2\Delta y CD)[\delta_y P^k+\delta_y P^{k-1}] \qquad (B-27)$$

where $\delta_x S=(S_{i,j}-S_{i-1,j})/\Delta x$

$\delta_y S=(S_{i,j+1}-S_{i,j})/\Delta y$

APPENDIX C

DERIVATION OF THE EXPLICIT FINITE DIFFERENCE EQUATIONS FOR VARIABLE AREA DUCT ACOUSTICS

The wave equation and all the boundary conditions in the transformed plane (ξ-η) for the general variable area duct acoustics are rewritten here from Chapter (V).

Wave equation: $$\omega^2 P_{tt} = (\alpha P_{\xi\xi} - 2\beta P_{\xi\eta} + \gamma P_{\eta\eta})/J^2 \qquad (C\text{-}1)$$

Inlet ($\xi=0$): $$P = \exp(i2\pi t) \qquad (C\text{-}2)$$

Bottom ($\eta=0$): $$(\omega/\zeta)P_t - (\gamma P_\eta - \beta P_\xi)/J\sqrt{\gamma} = 0 \qquad (C\text{-}3)$$

Top ($\eta=1$): $$(\omega/\zeta)P_t + (\gamma P_\eta - \beta P_\xi)/J\sqrt{\gamma} = 0 \qquad (C\text{-}4)$$

Plane exit ($\xi=1$): $$\omega P_t + (Y_\eta P_\xi - Y_\xi P_\eta)/J = 0 \qquad (C\text{-}5)$$

Far field ($\xi=1$): $$\omega P_t + P_\xi[(X-X_o)Y_\eta - YX_\eta]/RJ + P_\eta[YX_\xi - (X-X_o)Y_\xi]/RJ + P/2R = 0 \qquad (C\text{-}6)$$

Similarily to the previous appendix, the finite difference equations for the various cells shown in figure (1) are obtained by integrating the wave equation and the appropriate boundary conditions over the cell area.

$$\int_-^+\int_-^+ [\omega^2 P_{tt} = (\alpha P_{\xi\xi} - 2\beta P_{\xi\eta} + \gamma P_{\eta\eta})/J^2]\, d\xi\, d\eta \qquad (C\text{-}7)$$

$$\omega^2 \int_-^+\int_-^+ P_{tt}\, d\xi\, d\eta = (\alpha/J^2)\int_-^+\int_-^+ P_{\xi\xi}\, d\xi\, d\eta - (2\beta/J^2)\int_-^+\int_-^+ P_{\xi\eta}\, d\xi\, d\eta + (\gamma/J^2)\int_-^+\int_-^+ P_{\eta\eta}\, d\xi\, d\eta \qquad (C\text{-}8)$$

where the plus (+) and negative (-) signs are defined as before and applied here to the ξ and η directions.

Central Cell 1

$$(\omega^2/\Delta t^2)(P^{k+1}_{i,j}-2P^{k}_{i,j}+P^{k-1}_{i,j})\Delta\xi\Delta\eta=$$
$$(\Delta\eta/J^2)(P_\xi|^+-P_\xi|_-)+(\Delta\xi/J^2)(P_\eta|^+-P_\eta|_-) \qquad (C\text{-}9)$$

$$P^{k+1}_{i,j}=2P^{k}_{i,j}-P^{k-1}_{i,j}+(\Delta t/\omega J)^2$$
$$[(P_{i+1,j}-2P_{i,j}+P_{i-1,j})/\Delta\xi^2$$
$$+(P_{i,j+1}-2P_{i,j}+P_{i,j-1})/\Delta\eta^2]^k \qquad (C\text{-}10)$$

Upper Cell 2

With $\int_-^+ d\eta=\Delta\eta/2$ and the top boundary condition

$$P_\eta|^+=(\beta/\gamma)P_\xi-(\omega J/\zeta\sqrt{\gamma})P_t \qquad (C\text{-}11)$$

equation (C-8) becomes

$$\omega^2 P_{tt}\Delta\xi\Delta\eta/2=(\alpha\Delta\eta/2J^2)(P_\xi|^+-P_\xi|_-)$$
$$-(2\beta/J^2)(P|^+_+-P|^-_+-P|^+_-+P|^-_-)$$
$$+(\gamma\Delta\xi/J^2)(P_\eta|^+-P_\eta|_-) \qquad (C\text{-}12)$$

$$(1+CF)P^{k+1}=2P^k+(CF-1)P^{k-1}$$
$$+(\Delta t/\omega J)^2\,[\,\alpha P_{\xi\xi}-4\beta P_{\xi\eta}+(2/\Delta\eta)(\beta P_\xi-\gamma P_\eta)\,]^k \qquad (C\text{-}13)$$

where $CF=\Delta t\sqrt{\gamma}/\omega J\zeta\Delta\eta$

$$P_{\xi\xi}=(P_{i+1,j}-2P_{i,j}+P_{i-1,j})/\Delta\xi^2$$
$$P_\xi=(P_{i+1,j}-P_{i-1,j})/2_{\Delta\xi}$$
$$P_\eta=(P_{i,j}-P_{i,j-1})/\Delta\eta$$
$$P_{\xi\eta}=(P_{i+1,j}-P_{i-1,j}-P_{i+1,j-1}+P_{i-1,j-1})/4\Delta\xi\Delta\eta$$

Bottom Cell 3

The equation for the botton mesh points is identical to equation (C-13) except for the following conditions:

$$P_\eta|_- = (\beta/\gamma)P_\xi + (\omega J/\zeta\sqrt{\gamma})P_t$$

$$P_\xi = (P_{i-1,j} - P_{i+1,j})/2\Delta\xi$$

$$P_\eta = (P_{i,j} - P_{i,j+1})/\Delta\eta$$

$$P_{\xi\eta} = (P_{i+1,j+1} - P_{i-1,j+1} + P_{i-1,j} - P_{i+1,j})/4\Delta\xi\Delta\eta$$

Central Exit Cell 4

The equation for the exit mesh points in the case of a plane wave exit condition is

$$\omega^2 P_{tt}(\Delta\xi\Delta\eta)/2 = (\alpha\Delta\eta/J^2)(P_\xi|^+ - P_\xi|_-) - (2\beta/J^2)[P|^+_+ - P|^-_+ - P|^+_- + P|^-_-] + (\gamma\Delta\xi/2J^2)(P_\eta|^+ - P_\eta|_-) \qquad (C\text{-}14)$$

where $P_\xi|^+ = (Y_\xi/Y_\eta)P_\eta - (J\omega/Y_\eta)P_t$

$\int_-^+ d\xi = \Delta\xi/2$

Equation (C-14) then becomes

$$(1+CG)P^{k+1} = 2P^k + (CG-1)P^{k-1} + (\Delta t/\omega J)^2 [2\alpha(Y_\xi P_\eta/Y_\eta\Delta\xi - P_\xi/\Delta\xi) - 4\beta P_{\xi\eta} + \gamma P_{\eta\eta}]^k \qquad (C\text{-}15)$$

where $CG = \alpha\Delta t/\omega Y_\eta J\Delta\xi$

$$P_\xi = (P_{i,j} - P_{i-1,j})/\Delta\xi$$

$$P_\eta = (P_{i,j+1} - P_{i,j-1})/2\Delta\eta$$

$$P_{\xi\eta} = (P_{i,j+1} - P_{i,j-1} + P_{i-1,j-1} - P_{i-1,j+1})/4\Delta\xi\Delta\eta$$

In the case of a far field exit condition (C-6),

$$P_\xi|^+=[RJ/((X-X_o)Y_\eta-YX_\eta)]\ [-P/2R$$
$$-P_\eta(YX_\xi-(X-X_o)Y_\xi)/RJ-\omega P_t] \qquad (C\text{-}16)$$

the exit boundary pressure equation then becomes

$$(1+CH)P^{k+1}=2P^k+(CH-1)P^{k+1}$$
$$+(\Delta t/\omega J)^2\ [\ -(2\alpha/\Delta\xi CI)(JP/2+(CJ)P_\eta)$$
$$-(2\alpha/\Delta\xi)P_\xi-4\beta P_{\xi\eta}+\gamma P_{\eta\eta}]^k \qquad (C\text{-}17)$$

where $CI=(X-X_o)Y_\eta-YX_\eta$

$CH=\alpha R\Delta t/J\omega\Delta\xi(CI)$

$CJ=YX_\xi-(X-X_o)Y_\xi$

$P_{\eta\eta}=(P_{i,j+1}-2P_{i,j}+P_{i,j-1})/\Delta\eta^2$

$P_\eta=(P_{i,j+1}-P_{i,j-1})/2\Delta\eta$

$P_\xi=(P_{i,j}-P_{i-1,j})/\Delta\xi$

$P_{\xi\eta}=(P_{i,j+1}-P_{i,j-1}-P_{i-1,j+1}+P_{i-1,j-1})/4\Delta\xi\Delta\eta$

$P=P^k_{i,j}$

Upper Corner Cell 5

For the case of a plane wave exit condition, $\int_-^+dy=\Delta y/2$, $\int_-^+dx=\Delta x/2$. $P_\xi|^+$ and $P_\eta|^+$ are defined as before from the top wall and the plane wave exit conditions. The pressure equation becomes

$$(1+CK+CL)P^{k+1}=2P^k+(CK+CL-1)P^{k-1}$$
$$+(\Delta t/\omega J)^2\ [\ (2\alpha/\Delta\xi)(Y_\xi P_\eta/Y_\eta-P_\xi)$$
$$-8\beta P_{\xi\eta}+(2\gamma/\Delta\eta)(\beta P_\xi/\gamma-P_\eta)]^k \qquad (C\text{-}18)$$

where $CK=\alpha\Delta t/\omega J\Delta\xi Y$

$CL=\Delta t\sqrt{\gamma}/\omega J\zeta\Delta\eta$

$P_\xi=(P_{i,j}-P_{i-1,j})/\Delta\xi$

$P_\eta=(P_{i,j}-P_{i,j-1})/\Delta\eta$

$P_{\xi\eta}=(P_{i,j}-P_{i,j-1}+P_{i-1,j-1}-P_{i-1,j})/4\Delta\xi\Delta\eta$

In the case of a far field exit, all the previous conditions remain the same except for the exit boundary condition (C-6). The pressure equation then appears as

$$(1+CM+CN)P^{k+1}=2P^k+(CM+CN-1)P^{k-1}$$
$$+(\Delta t/\omega J)^2\,[-(2\alpha/\Delta\xi CI)(JP/2+(CJ)P_\eta$$
$$-(2\alpha/\Delta\xi)P_\xi-8\beta P_{\xi\eta}+(2\gamma/\Delta\eta)(\beta P_\xi/\gamma-P_\eta)]^k \qquad (C\text{-}19)$$

where $CI=(X-X_o)Y_\eta-YX_\eta$

$CJ=YX_\xi-(X-X_o)Y_\xi$

$CM=\alpha R\Delta t/\omega J\Delta\xi(CI)$

$CN=\sqrt{\gamma}\Delta t/\omega J\zeta\Delta\eta$

$P_\xi=(P_{i,j}-P_{i-1,j})/\Delta\xi$

$P_\eta=(P_{i,j}-P_{i,j-1})/\Delta\eta$

$P_{\xi\eta}=(P_{i,j}-P_{i-1,j}+P_{i-1,j-1}-P_{i,j-1})/4\Delta\xi\Delta\eta$

Bottom Corner Cell 6

Here, the plane wave exit condition is used, as well as, $\int_-^+ dy=\Delta y/2$ and $\int_-^+ dx=\Delta x/2$. The pressure equation for this case appears as

$$(1+CK+CL)P^{k+1}=2P^k+(CK+CL-1)P^{k-1}$$
$$+(\Delta t/\omega J)^2\,[(2\alpha/\Delta\xi)(Y_\xi P_\eta/Y_\eta-P_\xi)$$

$$-8\beta P_{\xi\eta}+(2\gamma/\Delta\eta)(P_{\eta}-\beta P_{\xi}/\gamma)]^{k} \qquad (C\text{-}20)$$

where CK and CL have been previously defined and

$$P_{\xi}=(P_{i,j}-P_{i-1,j})/\Delta\xi$$
$$P_{\eta}=(P_{i,j+1}-P_{i,j})/\Delta\eta$$
$$P_{\xi\eta}=(P_{i,j+1}-P_{i-1,j+1}+P_{i-1,j}-P_{i,j})/4\Delta\xi\Delta\eta$$

For the case of a far field exit, the pressure equation becomes

$$(1+CM+CN)P^{k+1}=2P^{k}+(CM+CN-1)P^{k-1}$$
$$+(\Delta t/\omega J)^{2}\ [-(2\alpha/\Delta\xi CI)(JP/2+(CJ)P_{\eta})$$
$$-(2\alpha/\Delta\xi)P_{\xi}-8\beta P_{\xi\eta}+(2\gamma/\Delta\eta)(P_{\eta}-\beta P_{\xi}/\gamma)]^{k} \qquad (C\text{-}21)$$

where CI, CJ, CM, and CN have been previously defined and

$$P_{\xi}=(P_{i,j}-P_{i-1,j})/\Delta\xi$$
$$P_{\eta}=(P_{i,j+1}-P_{i,j})/\Delta\eta$$
$$P_{\xi\eta}=(P_{i,j+1}-P_{i-1,j+1}+P_{i-1,j}-P_{i,j})/4\Delta\xi\Delta\eta$$

APPENDIX D

USER'S GUIDE FOR THE PROGRAM NOISE

NOISE was written in DEC-10 FORTRAN. It can, however, be easily adapted to any other FORTRAN processor with very few changes. The program will accept input data either interactively from the screen, or from a stored data file. In both cases, however, a minimum of 9, and a maximum of 15 program requests will have to be answered interactively at the beginning of each run.

The program is capable of storing a final grid formation in a data file for later use. Consequently, the user can supply the program with the file number of the grid data file to be read-in. Considerable CPU time savings can be achieved by storing the computational grid and reusing it later when appropriate. The program is also capable of storing the last two noise pressure time levels as well as the corresponding time step value in a data file for future initialization. In transient solutions, a great amount of computational time is often spent on bringing the solution to a certain point in time that might not be the desired steady state the user would like to achieve. Being able to "restart" the solution of the problem from that point rather than from the beginning can save the user a lot of frustration, time and money.

Since NOISE is interactive in nature, the user need only respond to the displayed questions without an in depth knowledge of the programming details. The displayed requests are very explicit and self explanatory, thus relieving the user from the trouble of studying a step-by-step user's guide as well as acquiring a good knowledge of the specifics of the program.

When possible, NOISE checks the answers for errors and repeats its request for input if in interactive mode, or displays a specific error message if reading data from a stored data file. Since the program has no means of knowing if the physical data being input contains any errors, it displays parts of the acquired data at two stages in the run and awaits permission to proceed if all systems are clear, or to exit if an error is dicovered.

The author has documented every request for input data in the program. This resulted in explicitly defined questions that can be easily understood and answered. The user need only follow the listing of the program in Appendix (E) and read the format statements of the data requests in sequence.

Explanatory comment statements can be found throughout the program. Each subroutine is preceded by a description of its function and its input and output parameters.

APPENDIX E

SOURCE LISTING OF THE FORTRAN PROGRAM NOISE

```
      REAL*4    X(90,30),Y(90,30),XI(90,30),ETA(90,30)
      REAL*4    BETAX(90),BETAY(90),DXS(90,30),DYS(90,30)
      REAL*4    ALPHAX(90),ALPHAY(90),XIETAX(30),C20,C30
      REAL*4    GAMA2(90,30),XJ1(90,30),VEC(30)
      REAL*4    ALPHA(90,30),BETA(90,30),CONST(90,30)
      REAL*4    XOLD(30),YOLD(30),DX(50),XANEW(30)
      REAL*4    PART2(30),PART5(30),CEXIT(30),DELTAT(90)
      REAL*4    LENOLD,LENEW,LENG,LENTOT
      COMPLEX   P(90,30),PO(90,30),PN(90,30)
      COMPLEX   P1,P2,P3,P4,P5,P6,P7,P8,P9,PZ
      COMPLEX   PXI2,PETA2,PXI,PETA,PETAXI
      COMPLEX   CUP(90),CDOWN(90),CUP1,CDOWN1
      COMPLEX   C21,C22,C31,C32,IMPED,IMPED5,IMPED9
      COMPLEX   ZUP1,ZUP2,ZUP3,ZDOWN1,ZDOWN2,ZDOWN3
      INTEGER   DVOUT,FILIN,XYDATA,SAVEXY
      INTEGER   PLTFIL,PRDATA,SAVPPO
      DIMENSION COVER(7),STARS(35),SCAN(25)
      COMMON/DUBS1/NX,NY
      COMMON/DUBS2/DX,YESNO6,DVOUT
      COMMON/DUBS3/RPN
C
C
      WRITE(5,10)
10    FORMAT(1H ,/' ENTER THE OUTPUT DEVICE NUMBER.')
      READ(5,*)DVOUT
C
      WRITE(5,20)
20    FORMAT(1H ,/' ENTER THE INPUT DATA FILE NUMBER.')
      READ(5,*)FILIN
C
30    WRITE(5,40)
40    FORMAT(1H ,/' ENTER "NEW" FOR A NEW GRID FORMATION,'/
     1 ' OR "OLD" TO READ THE FINAL GRID FROM A DATA FILE.')
      READ(5,50)OLDNEW
50    FORMAT( A3)
C
      IF(OLDNEW.NE.'OLD'.AND.OLDNEW.NE.'NEW')GO TO 30
      IF(OLDNEW.EQ.'NEW')GO TO 70
C
      WRITE(5,60)
60    FORMAT(1H ,/' ENTER THE DATA FILE NUMBER '/
     1 ' CONTAINING THE GRID''S FINAL FORMATION.')
      READ(5,*)XYDATA
C
      GO TO 110
```

```
C
70      WRITE(5,80)
80      FORMAT(1H ,/' INPUT Y TO SAVE THE GRID DATA ,'/
        1          ' INPUT N OTHERWISE.')
        READ(5,90)YESNO3
90      FORMAT( A1)
C
        IF(YESNO3.NE.'Y'.AND.YESNO3.NE.'N')GO TO 70
        IF(YESNO3.EQ.'N')GO TO 110
C
        WRITE(5,100)
100     FORMAT(1H ,/' ENTER THE DATA FILE DEVICE NUMBER IN '/
        1   ' WHICH TO SAVE THE GRID''S FINAL SOLUTION.')
        READ(5,*)SAVEXY
C
110     CONTINUE
C
120     WRITE(5,130)
130     FORMAT(1H ,/' ENTER "ABS1" IF P, AND PO ARE TO BE'/
        1   ' INITIALIZED TO (1.0,0.0). ENTER "INIT" TO'/
        2   ' READ P, AND PO INITIAL FROM A DATA FILE.')
        READ(5,140)ABSINI
140     FORMAT( A4)
C
        IF(ABSINI.NE.'ABS1'.AND.ABSINI.NE.'INIT')GO TO 120
        IF(ABSINI.EQ.'ABS1')GO TO 160
C
        WRITE(5,150)
150     FORMAT(1H ,/' ENTER THE DATA FILE NUMBER CONTAINING'
        1    /' THE INITIAL P, AND PO VALUES. ')
        READ(5,*)PRDATA
C
160     WRITE(5,170)
170     FORMAT(1H ,/' ENTER Y TO SAVE THE LAST TWO PRESSURE'
        1    /' LEVELS FOR FUTURE INITIALIZATION. '
        2    /' ENTER N OTHERWISE. ')
        READ(5,180)YESNO4
180     FORMAT( A1)
C
        IF(YESNO4.NE.'Y'.AND.YESNO4.NE.'N')GO TO 160
        IF(YESNO4.EQ.'N')GO TO 200
C
        WRITE(5,190)
190     FORMAT(1H ,/' ENTER THE DATA FILE NUMBER IN '/
        1 ' WHICH TO SAVE THE PRESSURE''S LAST TWO STEPS.')
        READ(5,*)SAVPPO
C
200     CONTINUE
C
        IF(OLDNEW.EQ.'OLD')GO TO 240
C
```

```
210     WRITE(5,220)
220     FORMAT(1H ,/' INPUT Y FOR A GRAPH OF THE INITIAL GRID.'
        1       /' INPUT N IF A GRAPH IS NOT WANTED.')
        READ(5,230)YESNO1
230     FORMAT(  A1)
C
        IF(YESNO1.NE.'Y'.AND.YESNO1.NE.'N')GO TO 210
C
240     WRITE(5,250)
250     FORMAT(1H ,/' INPUT Y FOR A GRAPH OF THE FINAL GRID.'
        1       /' INPUT N IF A GRAPH IS NOT WANTED.')
        READ(5,260)YESNO2
260     FORMAT(  A1)
C
        IF(YESNO2.NE.'Y'.AND.YESNO2.NE.'N')GO TO 240
        IF(YESNO2.NE.'Y'.AND.YESNO1.NE.'Y')GO TO 280
C
        WRITE(5,270)
270     FORMAT(1H ,/' INPUT AND SEPARATE BY COMMAS THE VIRTUAL'
        1     /' WINDOW COORDINATES:XMIN,XMAX,YMIN,YMAX.')
        READ(5,*)XMIN,XMAX,YMIN,YMAX
280     CONTINUE
C
290     WRITE(5,300)
300     FORMAT(1H ,/' ENTER Y FOR A PLOT OF PRESSURE',
        1   ' PRESSURE VERSUS DISTANCE "X"'/' ON THE BOTTOM'
        2   ,' SURFACE OF THE VARIABLE AREA DUCT.'/
        3   ' ENTER N IF PLOT IS NOT WANTED.')
        READ (5,310)YESNO9
310     FORMAT( A1)
C
        IF(YESNO9.NE.'Y'.AND.YESNO9.NE.'N')GO TO 290
        IF(YESNO9.EQ.'N')GO TO 330
C
        WRITE(5,320)
320     FORMAT(1H ,/' ENTER THE FILE DEVICE NUMBER IN'/
        1   ' WHICH TO STORE THE RESULTING P-X PLOT.')
        READ(5,*)PLTFIL
C
330     CONTINUE
C
340     WRITE(5,350)
350     FORMAT(1HO,'ENTER "UNI" FOR A UNIFORMLY',
        1   ' SPACED GRID POINT DISTRIBUTION'/' ON',
        2   ' THE INLET BOUNDARY. ENTER "VAR" FOR',
        3   ' A VARIABLE DISTRIBUTION.')
        READ(5,360)SPACE
360     FORMAT( A3)
C
        IF(SPACE.NE.'UNI'.AND.SPACE.NE.'VAR')GO TO 340
C
```

```
      IF(FILIN.NE.5)GO TO 390
370   WRITE(5,380)
380   FORMAT(1H ,/' ENTER "TWO", OR "THREE" FOR A 2, OR'/
     1 ' A 3-POINT BACKWARD TAYLOR SERIES EXPANSION'/
     2 ' IN THE EXIT BOUNDARY CONDITIONS. ')
C
390   CONTINUE
      READ(FILIN,400)SERIES
400   FORMAT( A5)
C
      IF(SERIES.EQ.'TWO'.OR.SERIES.EQ.'THREE')GO TO 410
      IF(FILIN.EQ.5)GO TO 370
C
      STOP ' STOP...DATA ERROR...READING: "SERIES". '
C
410   CONTINUE
C
      IF(FILIN.NE.5)GO TO 440
420   WRITE(5,430)
430   FORMAT(1H ,/' ENTER Y FOR A COMPLETE PRINT OUT',
     1  ' OF THE PRESSURE COEFFICIENTS'/' AND',
     2  ' CONSTANTS. ENTER N IF NONE IS WANTED.')
C
440   CONTINUE
      READ(FILIN,450)ALLOUT
450   FORMAT( A1)
C
      IF(ALLOUT.EQ.'Y'.OR.ALLOUT.EQ.'N')GO TO 460
      IF(FILIN.EQ.5)GO TO 420
C
      STOP ' STOP...DATA ERROR...READING: "ALLOUT". '
C
460   CONTINUE
C
      IF(FILIN.NE.5)GO TO 490
470   WRITE(5,480)
480   FORMAT(1H ,/' ENTER N TO PRINT THE ABSOLUTE ',
     1  'PRESSURES ONLY.'/' ENTER Y TO PRINT THE REAL',
     2  ', IMAGINARY, AND ABSOLUTE PRESSURES.')
C
490   CONTINUE
      READ(FILIN,500)YESNO7
500   FORMAT( A1)
C
      IF(YESNO7.EQ.'Y'.OR.YESNO7.EQ.'N')GO TO 510
      IF(FILIN.EQ.5)GO TO 470
C
      STOP ' STOP...DATA ERROR...READING: "YESNO7". '
C
510   CONTINUE
C
```

```
      IF(FILIN.NE.5)GO TO 540
520   WRITE(5,530)
530   FORMAT(1H ,/' ENTER Y FOR A STABILITY CRITERIA',
     1  ' CALCULATION ON THE GRID.'/' ENTER N IF ',
     2  'NONE IS WANTED.')
C
540   CONTINUE
      READ(FILIN,550)YESNO8
550   FORMAT( A1)
C
      IF(YESNO8.EQ.'Y'.OR.YESNO8.EQ.'N')GO TO 560
      IF(FILIN.EQ.5)GO TO 520
C
      STOP ' STOP...DATA ERROR...READING: "YESNO8". '
C
560   CONTINUE
C
      IF(FILIN.NE.5)GO TO 590
570   WRITE(5,580)
580   FORMAT(1H ,/' ENTER AND SEPERATE BY COMMAS '/
     1  'THE FOLLOWING PARAMETERS'/' FOR THE '
     2  'PRESSURE SOLUTION:'/
     3  ' THE TIME STEP VALUE "DT2",'
     4 /' THE NON-DIMENSIONAL FREQUENCY '
     5  '"ETA2",'/' THE TOTAL NUMBER OF TIME STEPS '
     6  '"ITMAX2", AND'/' THE PRINTING INCREMENT FOR'
     7  ' RESULTS OUTPUT "JPRNT2".')
C
590   CONTINUE
      READ(FILIN,*)DT2,ETA2,ITMAX2,JPRNT2
C
      IF(FILIN.NE.5)GO TO 620
600   WRITE(5,610)
610   FORMAT(1H ,/' INPUT IF "UP","DOWN","BOTH",'/
     1          ' OR "NONE" WALLS ARE SOFT. ')
C
620   CONTINUE
      READ(FILIN,630)SOFT
630   FORMAT( A4)
C
      IF(SOFT.EQ.'UP'.OR.SOFT.EQ.'DOWN'.OR.
     1  SOFT.EQ.'BOTH'.OR.SOFT.EQ.'NONE')GO TO 640
      IF(FILIN.EQ.5)GO TO 600
C
      STOP ' STOP...DATA ERROR...READING: "SOFT". '
C
640   CONTINUE
C
      IMPED9=CMPLX(0.33333E+33,0.33333E+33)
      IF(SOFT.EQ.'NONE')GO TO 790
C
```

```
      IF(FILIN.NE.5)GO TO 670
650   WRITE(5,660)
660   FORMAT(1H ,/' ENTER "MIX" FOR MIXED IMPEDANCES,'
     1   /' OR "SAME" IF SAME IMPEDANCE ALL OVER.')
C
670   CONTINUE
      READ(FILIN,680)MIXSAM
680   FORMAT( A4)
C
      IF(MIXSAM.EQ.'MIX'.OR.MIXSAM.EQ.'SAME')GO TO 690
      IF(FILIN.EQ.5)GO TO 650
C
      STOP ' STOP...DATA ERROR...READING: "MIXSAM". '
C
690   CONTINUE
C
      IF(FILIN.NE.5)GO TO 720
700   WRITE(5,710)
710   FORMAT(1H ,/' ENTER THE CHOICE FOR SOFT, OR HARD'/
     1 ' WALLS STARTING WITH THE INLET, MIDDLE, AND'/
     2 ' EXIT UPPER SECTIONS; THEN THE INLET, MIDDLE'/
     3 ' AND EXIT LOWER SECTIONS RESPECTIVELY. '//
     4 ' ENTER AND SEPERATE BY A SPACE: AN "S" FOR  '/
     5 ' SOFT, AND AN "H" FOR HARD FOR ALL SECTIONS'/
     6 ' IN THE ABOVE ORDER.')
C
720   CONTINUE
      READ(FILIN,730)(COVER(I),I=1,6)
730   FORMAT( 6A2)
C
      WARNG1='NO'
      DO 760 I=1,6
      IF(COVER(I).EQ.'S'.OR.COVER(I).EQ.'H')GO TO 760
      IF(FILIN.EQ.5)GO TO 750
C
      WRITE(5,740)I
740   FORMAT(1H ,/' DATA ERROR...READING: "COVER(',I1,').')
      STOP
C
750   WARNG1='YES'
760   CONTINUE
C
      IF(WARNG1.EQ.'YES')GO TO 700
C
      IF(MIXSAM.EQ.'MIX')GO TO 820
C
      IF(FILIN.NE.5)GO TO 780
      WRITE(5,770)
770   FORMAT(1H ,/' ENTER INSIDE A PAIR OF PARENTHESES,'/
     1 ' THE COMPLEX IMPEDANCE OF THE SOFT WALL(S).'/
     2 ' SEPERATE REAL AND IMAGINARY BY A COMMA. ')
```

```
C
780     CONTINUE
        READ(FILIN,*)IMPED
C
        GO TO 800
790     IMPED=IMPED9
800     CONTINUE
C
        DO 810 I=1,6
        IMPED5=IMPED
C
        IF(COVER(I).EQ.'H')     IMPED5=IMPED9
        IF(I.EQ.1)ZUP1=IMPED5
        IF(I.EQ.2)ZUP2=IMPED5
        IF(I.EQ.3)ZUP3=IMPED5
        IF(I.EQ.4)ZDOWN1=IMPED5
        IF(I.EQ.5)ZDOWN2=IMPED5
        IF(I.EQ.6)ZDOWN3=IMPED5
810     CONTINUE
C
        GO TO 930
C
820     CONTINUE
C
        IF(FILIN.NE.5)GO TO 840
        WRITE(5,830)
830     FORMAT(1H ,/' ENTER THE IMPEDANCES OF ""ONLY"" '/
        1 ' THE SOFT WALL(S) IN THE ORDER INDICATED '/
        2 ' BEFORE. EACH PAIR OF REAL AND IMAGINARY '/
        3 ' NUMBERS SHOULD BE PUT INSIDE RARENTHESES,'/
        4 ' AND SEPERATED BY COMMAS. '/
        5 ' HIT RETURN AFTER EACH PAIR. ')
C
840     CONTINUE
C
        DO 930 I=1,6
        IF(COVER(I).EQ.'H')GO TO 850
        READ(FILIN,*)IMPED
C
        GO TO 860
850     IMPED=IMPED9
860     CONTINUE
C
        GO TO (870,880,890,900,910,920),I
870     ZUP1=IMPED
        GO TO 930
880     ZUP2=IMPED
        GO TO 930
890     ZUP3=IMPED
        GO TO 930
900     ZDOWN1=IMPED
```

```
      GO TO 930
910   ZDOWN2=IMPED
      GO TO 930
920   ZDOWN3=IMPED
C
930   CONTINUE
C
      IF(OLDNEW.EQ.'OLD')GO TO 1850
C
      IF(FILIN.NE.5)GO TO 960
940   WRITE(5,950)
950   FORMAT(1H ,/
     1  ' DO YOU WANT A FAR FIELD EXIT CONDITION ?'/
     2  ' ENTER Y FOR YES, N FOR NO.')
C
960   CONTINUE
      READ(FILIN,970)YESNO5
970   FORMAT(  A1)
C
      IF(YESNO5.EQ.'Y'.OR.YESNO5.EQ.'N')GO TO 980
      IF(FILIN.EQ.5)GO TO 940
C
      STOP ' STOP...DATA ERROR...READING: "YESNO5". '
C
980   CONTINUE
C
      IF(FILIN.NE.5)GO TO 1000
      WRITE(5,990)
990   FORMAT(1H ,/
     1  ' ENTER AND SEPARATE BY COMMAS THE FOLLOWING VARI-'/
     2 ' ABLES: THE NUMBER (NB.) OF GRID POINTS (G.P.) IN'/
     3 ' THE Y-DIRECTION "NY", THE NB. OF G.P. IN THE X-'/
     4 ' DIRECTION FOR THE VARIABLE AREA DUCT (V.A.D.)'/
     5 ' "NX2"  WHICH SHOULD BE AN ODD NB.  ,THE NON-DIM-'/
     6 ' ENSIONAL (NON-D.) LENGTH OF THE V.A.D. "ALENG2" '/
     7 ' E.G. 1.0,2.2,...ETC., THE NON-D. HEIGHT OF THE '/
     8 ' V.A.D. "Z", AND THE MAXIMUM DEPTH OF THE THROAT'/
     9 ' OF THE V.A.D. "H". ')
C
1000  CONTINUE
      READ(FILIN,*)NY,NX2,ALENG2,Z,H
C
      IF(FILIN.NE.5)GO TO 1020
      WRITE(5,1010)
1010  FORMAT(1H ,/
     1  ' ENTER AND SEPARATE BY COMMAS THE NON-D. LENGTHS'/
     2 ' OF THE INLET "ALENG1" AND EXIT "ALENG3" EXTEN- '/
     3 ' SION STRAIGHT DUCTS  IN THAT ORDER. E.G. 1.0, '/
     4 ' 2.5,3.3,...ETC., AND 0.0 FOR NO EXTENSION DUCT'/
     5 ' ADDITION. ')
C
```

```
1020    CONTINUE
        READ(FILIN,*)ALENG1,ALENG3
C
        IF(FILIN.NE.5)GO TO 1040
        WRITE(5,1030)
1030    FORMAT(1H ,/' ENTER AND SEPERATE BY COMMAS '/
       1  'THE FOLLOWING'/' GRID SOLUTION PARAMETERS:'
       2 /' THE MAXIMUM NUMBER "ITMAX1" OF ALLOWABLE'
       3  ' ITERATIONS,'/' THE PRINTING INCREMENT '
       4  '"JPRNT1" FOR RESULTS OUTPUT,'/' AND THE'
       5  ' MAXIMUM PERMISSIBLE RESIDUAL "EPS"'
       6 /' OF TOTAL GRID VARIATION FOR CONVERGENCE.')
C
1040    CONTINUE
        READ(FILIN,*)ITMAX1,JPRNT1,EPS
C
        IF(YESNO5.EQ.'N')GO TO 1160
C
        IF(FILIN.NE.5)GO TO 1070
1050    WRITE(5,1060)
1060    FORMAT(1H ,/' DO YOU WANT TO PRINT OUT '/
       1     ' THE GEOMETRIC PROGRESSION PARAMETERS ?'/
       2     ' ENTER Y FOR YES, N FOR NO. ')
C
1070    CONTINUE
        READ(FILIN,1080)YESNO6
1080    FORMAT( A1)
C
        IF(YESNO6.EQ.'Y'.OR.YESNO6.EQ.'N')GO TO 1090
        IF(FILIN.EQ.5)GO TO 1050
C
        STOP ' STOP...DATA ERROR...READING: "YESNO6". '
C
1090    CONTINUE
C
        IF(FILIN.NE.5)GO TO 1110
        WRITE(5,1100)
1100    FORMAT(1H ,/' ENTER AND SEPERATE BY COMMAS '/
       1  ' THE FOLLOWING FAR FIELD PARAMETERS :'//
       2  ' THE NUMBER OF INLET GRID POINTS "LP1",'/
       3  ' THE NUMBER OF GRID POINTS AROUND THE HALF'/
       4  ' CIRCLE (KNEE) "LP2", AND THE NUMBER OF'/
       5  ' EXIT GRID POINTS "LP3".')
C
1110    CONTINUE
        READ(FILIN,*)LP1,LP2,LP3
C
        IF(FILIN.NE.5)GO TO 1130
        WRITE(5,1120)
1120    FORMAT(1H ,/' THE CENTER COORDINATES "XSCOOR",'/
       1  ' "YSCOOR", AND THE RADIUS "ROUND" OF THE HALF'/
```

```
      2  ' CIRCLE KNEE.')
C
1130  CONTINUE
      READ(FILIN,*)XSCOOR,YSCOOR,ROUND
C
      IF(FILIN.NE.5)GO TO 1150
      WRITE(5,1140)
1140  FORMAT(1H ,/' THE CENTER COORDINATES "XLCOOR",'/
      1  ' "YLCOOR", AND THE RADIUS "FIELD" OF THE'/
      2  ' FAR FIELD CIRCULAR EXIT.')
C
1150  CONTINUE
      READ(FILIN,*)XLCOOR,YLCOOR,FIELD
1160  CONTINUE
C
      WRITE(5,1170)NX2,NY
1170  FORMAT(1H ,///' THE NUMBER OF GRID POINTS IN ',
      1  'THE VARIABLE AREA SECTION (NX2) :',2X,I3//
      2  ' THE NUMBER OF GRID POINTS IN THE Y',
      3  ' DIRECTION (NY) :',2X,I3)
C
      WRITE(5,1180)ALENG1,ALENG2,ALENG3
1180  FORMAT(1H ,/' THE NON-DIMENSIONAL LENGTHS OF THE',
      1  ' ... INLET (ALENG1) :',T60,F4.1/T41,'V.A. ',
      2  '(ALENG2)  :',T60,F4.1/T41,'EXIT (ALENG3)  :',
      3  T60,F4.1)
C
      WRITE(5,1190)Z
1190  FORMAT(1H ,/' THE NON-DIMENSIONAL HEIGHT '
      1 ,'(Z) :',T35,F4.1)
C
      WRITE(5,1200)H
1200  FORMAT(1H ,/' THE NON-DIMENSIONAL DEPTH OF THE',
      1 ' THROAT (H) :',T48,F4.1)
C
      WRITE(5,1210)YESNO5
1210  FORMAT(1H ,//' IS THE EXIT CONDITION A FAR',
      1 ' FIELD ONE (YESNO5) ? (Y/N) .... : ',T66,A3)
C
      WRITE(5,1220)LP1,LP2,LP3
1220  FORMAT(1H ,/' THE NUMBER OF GRID POINTS IN ... ',
      1 'THE INLET PORTION (LP1)  :'T63,I3/T35,'THE '
      2 ,'"KNEE" PORTION (LP2) :',T63,I3/T35,'THE EXIT '
      3 ,'PORTION  (LP3)  :',T63,I3)
C
      WRITE(5,1230)XSCOOR,YSCOOR,ROUND
1230  FORMAT(1H ,//' ALL OF THE THE FOLLOWING (',
      1 'NON-DIMENSIONAL) DIMENSIONS'/' ARE MEASURED'
      2 ,' FROM THE EXIT OF THE NOZZLE ...'//
      3 ' THE "KNEE" CENTER COORDINATES (XSCOOR,YSCOOR) :'
      4 ,'  (',T53,F5.2,' , ',F5.2,'  )'/
```

```
     5 ' THE "KNEE" RADIUS (ROUND) :',T33,F5.2)
C
      WRITE(5,1240)XLCOOR,YLCOOR,FIELD
1240  FORMAT(1H ,/' THE FAR FIELD CENTER COORDINATES',
     1 ' (XLCOOR,YLCOOR) :  (',T56,F5.2,' , ',F5.2,
     2 '  )'/' THE FAR FIELD RADIUS (FIELD) :'
     3 ,T33,F5.2,//)
C
C
      PAUSE 'PAUSE HERE TO CHECK THE GRID DATA FOR ERRORS'
C
C
C.....START THE VARIABLE AREA DUCT GRID FORMATION.
C
      NYM1=NY-1
      NYM2=NY-2
      PI=3.1415927
      TWOPI=2.*PI
C
      JX1=ALENG1*(NX2-1.)/ALENG2
      NX1=JX1+1
C
      JX3=ALENG3*(NX2-1.)/ALENG2
      NX3=JX3+1
C
      NX7=NX2+JX1+JX3
      NX7P1=NX7+1
      NX7M1=NX7-1
      NX5=JX1+NX2
      ALENG5=ALENG1+ALENG2+ALENG3
C
      DO 1250 J=1,NY
      DO 1250 I=1,NX7
      X(I,J)=ALENG5*(I-1.)/(NX7-1.)
1250  CONTINUE
C
      DO 1260 I=1,NX7
      Y(I,NY)=Z
1260  CONTINUE
C
      CC=16.*H/(ALENG2**2)
      CD=-32.*H/(ALENG2**3)
      CE=16.*H/(ALENG2**4)
C
      DO 1270 I=NX1,NX5
      X6=X(I,NY)-ALENG1
      YP=CC*X6**2+CD*X6**3+CE*X6**4
      Y(I,NY)=Z-YP
1270  CONTINUE
C
      IF(SPACE.EQ.'UNI')GO TO 1300
```

```
C
      WRITE(5,1280)
1280  FORMAT(1HO,'ENTER THE UPPER GRID SPACING',
     1   ' ON THE INLET BOUNDARY.')
      READ(5,*)XAY
C
      DO 1290 I=1,NX7
      Y1=Y(I,NY)
      Y2=Y(I,1)
      LOCY=1
C
      CALL GOPRO1(Y1,Y2,NYM1,XAY,LOCY)
C
      DO 1290 JJ=1,NYM1
      J=NYM1+1-JJ
      Y(I,J)=Y(I,J+1)-DX(JJ)
1290  CONTINUE
      GO TO 1310
C
1300  CONTINUE
      DO 1310 I=1,NX7
      DO 1310 J=1,NY
      Y(I,J)=Y(I,NY)*(J-1.)/(NY-1.)
1310  CONTINUE
C
C.....START THE FAR FIELD GRID FORMATION.
C
      IF(YESNO5.EQ.'N')GO TO 1560
C
      XLFAR=XLCOOR+X(NX7,1)
      YLFAR=YLCOOR+Y(NX7,1)
      XSMALL=XSCOOR+X(NX7,1)
      YSMALL=YSCOOR+Y(NX7,1)
C
      NP1=LP1-1
      NP2=LP2-1
      NP3=LP3-1
      NP3M1=NP3-1
      NX10=NP1+NP2+NP3+NX7
      LIM1=NX7+NP1
      LIM1P1=LIM1+1
      LIM1M1=LIM1-1
      LIM2=NX7+NP1+NP2
      LIM2P1=LIM2+1
      LIM2P2=LIM2+2
      LIM2M1=LIM2-1
C
      Y(NX10,NY)=Z+2.*ROUND
      X(NX10,NY)=-SQRT(FIELD**2-(Y(NX10,NY)-YLFAR)**2)
     1        +XLFAR
      ANGLE1=PI/2.+ACOS(Y(NX10,NY)/FIELD)
```

```
C
      DO 1320 J=1,NY
      SLICE=ANGLE1*Y(NX7,J)/Z
      X(NX10,J)=FIELD*COS(SLICE)+XLFAR
      Y(NX10,J)=FIELD*SIN(SLICE)+YLFAR
1320  CONTINUE
C
      X1=X(NX7,NY)
      X2=XSMALL
      PERIM=PI*ROUND
      XA=PERIM/NP2
      LOC=2
C
      CALL GOPRO1(X1,X2,NP1,XA,LOC)
C
      DO 1330 J=1,NY
      DO 1330 I=NX7P1,LIM1
      X(I,J)=X(I-1,J)+DX(I-NX7)
      Y(I,J)=Y(1,J)
1330  CONTINUE
C
      ANGLE2=PI/NP2
      DO 1340 I=LIM1P1,LIM2
      SLICE2=-PI/2.+(I-LIM1)*ANGLE2
      X(I,NY)=ROUND*COS(SLICE2)+XSMALL
      Y(I,NY)=ROUND*SIN(SLICE2)+YSMALL
1340  CONTINUE
C
      X1=X(LIM2,NY)
      X2=X(NX10,NY)
      LOC=1
C
      CALL GOPRO1(X1,X2,NP3,XA,LOC)
C
      DO 1350 I=LIM2P1,NX10
      KLM=I-LIM2P1+1
      X(I,NY)=X(I-1,NY)-DX(KLM)
      Y(I,NY)=Y(NX10,NY)
1350  CONTINUE
C
      DO 1390 J=1,NY
      ARC=YSMALL-Y(LIM1,J)
      DETER1=ARC**2*(X(NX10,J)-XSMALL)
      DETER2=SQRT((Y(NX10,J)-YSMALL)**4*ARC**2
     1     -ARC**4*(Y(NX10,J)-YSMALL)**2
     2     +ARC**2*(Y(NX10,J)-YSMALL)**2*(X(NX10,J)-XSMALL)**2)
      DETER3=(Y(NX10,J)-YSMALL)**2+(X(NX10,J)-XSMALL)**2
      XPRIME=(DETER1+DETER2)/DETER3+XSMALL
      XSECON=(DETER1-DETER2)/DETER3+XSMALL
      YPRIME=(ARC**2-(XPRIME-XSMALL)*(X(NX10,J)-XSMALL))
     1     /(Y(NX10,J)-YSMALL)+YSMALL
```

```
      YSECON=(ARC**2-(XSECON-XSMALL)*(X(NX10,J)-XSMALL))
     1      /(Y(NX10,J)-YSMALL)+YSMALL
      TREX1=XPRIME*Y(NX10,J)-YPRIME*X(NX10,J)
     1     +XSMALL*(YPRIME-Y(NX10,J))
     2     +YSMALL*(X(NX10,J)-XPRIME)
      TREX2=XSECON*Y(NX10,J)-YSECON*X(NX10,J)
     1     +XSMALL*(YSECON-Y(NX10,J))
     2     +YSMALL*(X(NX10,J)-XSECON)
      TREX3=(XPRIME-XSMALL)**2+(YPRIME-YSMALL)**2-ARC**2
      TREX4=(XSECON-XSMALL)**2+(YSECON-YSMALL)**2-ARC**2
      IF(TREX1.GE.0..AND.ABS(TREX3).LE.0.0001)GO TO 1370
      IF(TREX2.GE.0..AND.ABS(TREX4).LE.0.0001)GO TO 1380
      WRITE(DVOUT,1360)
1360  FORMAT(' NO ROOTS IN THE POSITIVE Z-DIRECTION')
      STOP
C
1370  X(LIM2,J)=XPRIME
      Y(LIM2,J)=YPRIME
      GO TO 1390
1380  X(LIM2,J)=XSECON
      Y(LIM2,J)=YSECON
1390  CONTINUE
C
      DO 1400 J=1,NY
      XOLD(J)=X(LIM2,J)
      YOLD(J)=Y(LIM2,J)
1400  CONTINUE
C
      XBIG=X(LIM2,1)
      NUMB=1
      DO 1410 J=2,NY
      IF(XBIG.GT.X(LIM2,J))GO TO 1410
      XBIG=X(LIM2,J)
      NUMB=J
1410  CONTINUE
C
      SLOPE1=(Y(LIM2,NUMB+1)-Y(LIM2,NUMB))
     1      /(X(LIM2,NUMB+1)-X(LIM2,NUMB))
      BEE1=Y(LIM2,NUMB)-X(LIM2,NUMB)*SLOPE1
      XINTR1=-BEE1/SLOPE1
      X(LIM2,1)=XINTR1
      Y(LIM2,1)=0.0
      NUMBM1=NUMB-1
      DO 1420 J=2,NUMBM1
      SLOPE2=(Y(NX10,J)-Y(LIM2,J))/(X(NX10,J)-X(LIM2,J))
      BEE2=Y(NX10,J)-X(NX10,J)*SLOPE2
      X(LIM2,J)=(BEE2-BEE1)/(SLOPE1-SLOPE2)
      Y(LIM2,J)=(SLOPE2*BEE1-SLOPE1*BEE2)/(SLOPE2-SLOPE1)
1420  CONTINUE
C
      KFLAG=0
```

```
      DO 1480 JJ=1,NYM1
      J=NY-JJ
      IF(XOLD(J).NE.X(LIM2,J).AND.KFLAG.NE.1)GO TO 1440
      IF(XOLD(J).NE.X(LIM2,J).AND.KFLAG.EQ.1)GO TO 1450
      RNEW=YSMALL-Y(LIM1,J)
      ANGLE6=(PI/2.+ASIN((Y(LIM2,J)-YSMALL)/RNEW))/NP2
      DO 1430 I=LIM1P1,LIM2
      SLICE6=-PI/2.+(I-LIM1)*ANGLE6
      X(I,J)=RNEW*COS(SLICE6)+XSMALL
      Y(I,J)=RNEW*SIN(SLICE6)+YSMALL
1430  CONTINUE
      GO TO 1480
1440  CONTINUE
      KFLAG=1
      PEROLD=RNEW*ANGLE6*NP2
      DIVOLD=PEROLD/NP2
1450  CONTINUE
      RNEW=YSMALL-Y(LIM1,J)
      ANGLE6=PI/2.+ASIN((YOLD(J)-YSMALL)/RNEW)
      PERNEW=RNEW*ANGLE6
      LENG=SQRT((YOLD(J)-Y(LIM2,J))**2
     1   +(XOLD(J)-X(LIM2,J))**2)
      LENTOT=PERNEW+LENG
      DIVNEW=DIVOLD*LENTOT/PEROLD
      MANY=PERNEW/DIVNEW
      LEFT=NP2-MANY
      LPMANY=LIM1+MANY
C
      IF(MANY.LT.1)GO TO 1460
      DO 1460 I=LIM1P1,LPMANY
      ANGLE7=DIVNEW/RNEW*(I-LIM1)
      X(I,J)=RNEW*SIN(ANGLE7)+X(LIM1,J)
      Y(I,J)=RNEW*(1.-COS(ANGLE7))+Y(LIM1,J)
1460  CONTINUE
C
      DO 1470 I=1,LEFT
      II=LIM2-I
      ANGLE8=ATAN((Y(LIM2,J)-YOLD(J))
     1     /(X(LIM2,J)-XOLD(J)))
      HYPO=DIVNEW*I
      X(II,J)=X(LIM2,J)-HYPO*COS(ANGLE8)
      Y(II,J)=Y(LIM2,J)-HYPO*SIN(ANGLE8)
1470  CONTINUE
      PEROLD=LENTOT
      DIVOLD=DIVNEW
1480  CONTINUE
C
      XAOLD=XA
      LENOLD=ABS(X(NX10,NY)-X(LIM2,NY))
      DO 1510 JJ=1,NYM2
      J=NY-JJ
```

```
      TERM1=Y(NX10,J)-Y(LIM2,J)
      TERM2=X(NX10,J)-X(LIM2,J)
      ANGLE5=ATAN(TERM1/TERM2)
      LENEW=SQRT(TERM1**2+TERM2**2)
      XANEW(1)=XAOLD*LENEW/LENOLD
      X(LIM2P1,J)=X(LIM2,J)+XANEW(1)*COS(ANGLE5)
     1           *TERM2/ABS(TERM2)
      Y(LIM2P1,J)=Y(LIM2,J)+XANEW(1)*SIN(ANGLE5)
     1           *TERM2/ABS(TERM2)
      DO 1490 I=1,NP3M1
      XANEW(I+1)=XANEW(I)*RPN
1490  CONTINUE
C
      DO 1500 I=LIM2P2,NX10
      KLM=I-LIM2P1+1
      X(I,J)=X(I-1,J)+XANEW(KLM)*COS(ANGLE5)*TERM2/ABS(TERM2)
      Y(I,J)=Y(I-1,J)+XANEW(KLM)*SIN(ANGLE5)*TERM2/ABS(TERM2)
1500  CONTINUE
C
      XAOLD=XANEW(1)
      LENOLD=LENEW
C
1510  CONTINUE
C
      X1=X(LIM2,1)
      X2=X(NX10,1)
      XA=ABS(X(LIM2,1)-X(LIM2M1,1))
      LOC=1
C
      CALL GOPRO1(X1,X2,NP3,XA,LOC)
C
      DO 1520 I=LIM2P1,NX10
      KLM=I-LIM2P1+1
      X(I,1)=X(I-1,1)+DX(KLM)
      Y(I,1)=Y(NX10,1)
1520  CONTINUE
C
      X1=X(NX7,1)
      X2=X(LIM1,1)
      XA1=ABS(X(NX7,1)-X(NX7M1,1))
      XA2=ABS(X(LIM1P1,1)-X(LIM1,1))
      WRITE(5,1530)
1530  FORMAT(1H ,'ENTER JNP1,JNP2 FOR THE BOTTOM SURFACE')
      READ(5,*)JNP1,JNP2
C
      CALL GOPRO2(X1,X2,XA1,XA2,JNP1,JNP2)
C
      DO 1540 I=NX7P1,LIM1
      X(I,1)=X(I-1,1)+DX(I-NX7)
      Y(I,1)=Y(1,1)
1540  CONTINUE
```

```
C-
        X1=X(NX7,NY)
        X2=X(LIM1,NY)
        XA1=ABS(X(NX7,NY)-X(NX7M1,NY))
        XA2=ABS(X(LIM1,NY)-X(LIM1P1,NY))
        WRITE(5,1550)
1550    FORMAT(1H ,'ENTER JNP1,JNP2 FOR THE TOP SURFACE')
        READ(5,*)JNP1,JNP2
C
        CALL GOPRO2(X1,X2,XA1,XA2,JNP1,JNP2)
C
        DO 1560 I=NX7P1,LIM1
        X(I,NY)=X(I-1,NY)+DX(I-NX7)
        Y(I,NY)=Y(1,NY)
1560    CONTINUE
C
        IDT=0
        IPRNT=0
C
        NX=NX7+NP1+NP2+NP3
        NXM1=NX-1
C
C.....CONSTANTS FOR THE FACTORED GRID SCHEME.
C
        DETA=1./(NY-1.)
        DXI=1./(NX-1.)
        DT1=DXI**2
        IF(DETA .LT. DXI)DT1=DETA**2
C
        C1=DT1
        C2=2.*DXI**2
        C3=2.*DETA**2
        C4=2.*DETA
        C5=2.*DXI
        C6=DETA**2
        C7=DXI**2
        C8=4.*DETA*DXI
C
        IF(YESNO1.EQ.'Y')CALL PAINT1(X,Y,XMIN,XMAX,YMIN,YMAX)
C
        GO TO 2000
C
C.....THE FACTORED INTEGRATION SCHEME.
C
1570    DO 1610 J=2,NYM1
        DO 1590 I=2,NXM1
        X2=(X(I,J+1)-X(I,J-1))/C4
        X1=(X(I+1,J)-X(I-1,J))/C5
        Y2=(Y(I,J+1)-Y(I,J-1))/C4
        Y1=(Y(I+1,J)-Y(I-1,J))/C5
        X22=(X(I,J+1)-2*X(I,J)+X(I,J-1))/C6
```

```
      X11=(X(I+1,J)-2*X(I,J)+X(I-1,J))/C7
      X12=(X(I+1,J+1)+X(I-1,J-1)-X(I+1,J-1)-X(I-1,J+1))/C8
      Y22=(Y(I,J+1)-2*Y(I,J)+Y(I,J-1))/C6
      Y11=(Y(I+1,J)-2*Y(I,J)+Y(I-1,J))/C7
      Y12=(Y(I+1,J+1)+Y(I-1,J-1)-Y(I+1,J-1)-Y(I-1,J+1))/C8
      AL=X2**2+Y2**2
      BTA=X1*X2+Y1*Y2
      GAMA=X1**2+Y1**2
      RHSY=(AL*Y11+GAMA*Y22-2*BTA*Y12)*C1
      RHSX=(AL*X11+GAMA*X22-2*BTA*X12)*C1
      D1=-AL*DT1/C2
      D2=1.-2*D1
C
      IF(I .EQ. 2)GO TO 1580
C
      D5=D2+D1*BETAX(I-1)
      ALPHAX(I)=(RHSX-D1*ALPHAX(I-1))/D5
      BETAX(I)=-D1/D5
      D5=D2+D1*BETAY(I-1)
      ALPHAY(I)=(RHSY-D1*ALPHAY(I-1))/D5
      BETAY(I)=-D1/D5
      GO TO 1590
C
1580  ALPHAX(2)=RHSX/D2
      BETAX(2)=-D1/D2
      ALPHAY(2)=RHSY/D2
      BETAY(2)=-D1/D2
1590  CONTINUE
C
      DXS(NXM1,J)=ALPHAX(NXM1)
      DYS(NXM1,J)=ALPHAY(NXM1)
C
      DO 1600 II=3,NXM1
      I=NX+1-II
      DXS(I,J)=ALPHAX(I)+BETAX(I)*DXS(I+1,J)
1600  DYS(I,J)=ALPHAY(I)+BETAY(I)*DYS(I+1,J)
C
1610  CONTINUE
C
      RESIDX=0.
      RESIDY=0.
C
      DO 1650 I=2,NXM1
      DO 1630 J=2,NYM1
      X1=(X(I+1,J)-X(I-1,J))/C5
      Y1=(Y(I+1,J)-Y(I-1,J))/C5
      GAMA=X1**2+Y1**2
      D1=-GAMA*DT1/C3
      D2=1.-2*D1
      RHSX=DXS(I,J)
      RHSY=DYS(I,J)
```

```
      IF(J .EQ. 2)GO TO 1620
      D5=D2+D1*BETAX(J-1)
      ALPHAX(J)=(RHSX-D1*ALPHAX(J-1))/D5
      BETAX(J)=-D1/D5
      D5=D2+D1*BETAY(J-1)
      ALPHAY(J)=(RHSY-D1*ALPHAY(J-1))/D5
      BETAY(J)=-D1/D5
      GO TO 1630
C
1620  ALPHAX(2)=RHSX/D2
      BETAX(2)=-D1/D2
      ALPHAY(2)=RHSY/D2
      BETAY(2)=-D1/D2
1630  CONTINUE
C
      DXX1=ALPHAX(NYM1)
      X(I,NY-1)=X(I,NY-1)+DXX1
      RESIDX=ABS(DXX1)+RESIDX
      DYY1=ALPHAY(NYM1)
      Y(I,NY-1)=Y(I,NY-1)+DYY1
      RESIDY=ABS(DYY1)+RESIDY
C
      DO 1640 JJ=3,NYM1
      J=NY+1-JJ
      DXX1=ALPHAX(J)+BETAX(J)*DXX1
      DYY1=ALPHAY(J)+BETAY(J)*DYY1
      Y(I,J)=Y(I,J)+DYY1
      X(I,J)=X(I,J)+DXX1
      RESIDX=RESIDX+ABS(DXX1)
1640  RESIDY=RESIDY+ABS(DYY1)
1650  CONTINUE
C
      IDT=IDT+1
      IPRNT=IPRNT+1
C
      WRITE(DVOUT,1660)IDT,RESIDX,RESIDY
1660  FORMAT(1H ,/' IDT=',I4,2X,'RESIDX=',E12.5,2X,
     1       'RESIDY=',E12.5)
      RESUM=RESIDX+RESIDY
      WRITE(DVOUT,1670)RESUM,EPS,IDT,ITMAX1,IPRNT,JPRNT1
1670  FORMAT(1H ,'CHECK',2E12.4,4I5)
C
      IF((RESIDX+RESIDY) .LT. EPS)GO TO 1680
      IF(IDT .EQ. ITMAX1)GO TO 1800
      IF(IPRNT .EQ. JPRNT1)GO TO 1820
      GO TO 1570
C
1680  WRITE(DVOUT,1690)
1690  FORMAT(1H ,/' CONVERGENCE OF GRID')
C
      GO TO 1720
```

```
1700    WRITE(DVOUT,1710)
1710    FORMAT(1H ,//20X,'INITIAL VALUES')
C
1720    WRITE(DVOUT,1730)
1730    FORMAT(1H ,/20X,'X VALUES')
C
        DO 1740 JJ=1,NY
        J=NY+1-JJ
1740    WRITE(DVOUT,1750)(X(I,J),I=1,NX)
1750    FORMAT(1H ,13F10.5)
C
        WRITE(DVOUT,1760)
1760    FORMAT(1H ,/20X,'Y VALUES')
C
        DO 1770 JJ=1,NY
        J=NY+1-JJ
1770    WRITE(DVOUT,1780)(Y(I,J),I=1,NX)
1780    FORMAT(1H ,13F10.5)
C
        IF(IDT .EQ. 0)GO TO 1570
        IF((RESIDX+RESIDY) .LT. EPS)GO TO 1850
        IF(IDT .EQ. ITMAX1)GO TO 1840
        IF(IPRNT .EQ. 0)GO TO 1570
C
        WRITE(DVOUT,1790)IPRNT
1790    FORMAT(1H ,'IPRNT1=',I3)
1800    WRITE(DVOUT,1810)
1810    FORMAT(1H ,/' MAXIMUM ITERATIONS EXCEEDED.')
        GO TO 1720
C
1820    IPRNT=0
        WRITE(DVOUT,1830)IPRNT
1830    FORMAT(1H ,'IPRNT=',I3)
C
        GO TO 1720
C
1840    STOP ' NON-CONVERGENCE OF GRID. '
C
1850    CONTINUE
C
        IF(OLDNEW.EQ.'NEW')GO TO 1990
C
        DO 1860 K=1,58
        READ(XYDATA,1870)(SCAN(I),I=1,21)
1860    WRITE(DVOUT,1870)(SCAN(I),I=1,21)
1870    FORMAT(1H ,21A5)
C
        REWIND XYDATA
C
        READ(XYDATA,1880)NX1,NX2,NX3
1880    FORMAT(1H ,9(/),T95,I3/T95,I3/T95,I3)
```

```
      READ(XYDATA,1890)ALENG1,ALENG2,ALENG3
1890  FORMAT(1H ,/T91,F4.1/T91,F4.1/T91,F4.1)
      READ(XYDATA,1900)Z,H
1900  FORMAT(1H ,/T62,F4.1//T75,F4.1)
      READ(XYDATA,1910)YESNO5,LP1,LP2,LP3
1910  FORMAT(1H ,12(/),T92,A3//T88,I3/T88,I3/T88,I3)
      READ(XYDATA,1920)XSCOOR,YSCOOR,ROUND
1920  FORMAT(1H ,5(/),T80,F5.2,3X,F5.2/T60,F5.2)
      READ(XYDATA,1930)XLCOOR,YLCOOR,FIELD
1930  FORMAT(1H ,/T83,F5.2,3X,F5.2/T60,F5.2)
      READ(XYDATA,1940)NX,NY
1940  FORMAT(1H ,6(/),T90,I3/T90,I3////)
C
      WRITE(DVOUT,1950)XYDATA
1950  FORMAT(1H1,T25,'THE X AND Y ARRAYS OF THE ',
     1  'COMPLETE GRID AS READ FROM DATA FILE : FOR',
     2  I2,'.DAT'//T50,'X VALUES')
C
      DO 1960 JJ=1,NY
      J=NY-JJ+1
      READ(XYDATA,1750)(X(I,J),I=1,NX)
1960  WRITE(DVOUT,1750)(X(I,J),I=1,NX)
C
      WRITE(DVOUT,1970)
1970  FORMAT(1H1,T50,'Y VALUES')
C
      DO 1980 JJ=1,NY
      J=NY-JJ+1
      READ(XYDATA,1780)(Y(I,J),I=1,NX)
1980  WRITE(DVOUT,1780)(Y(I,J),I=1,NX)
C
      NX5=NX1+NX2-1
      NX7=NX1+NX2+NX3-2
      NXM1=NX-1
      NYM1=NY-1
      PI=3.1415927
      TWOPI=2.*PI
C
      DETA=1./(NY-1.)
      DXI=1./(NX-1.)
      DT1=DXI**2
      IF(DETA .LT. DXI)DT1=DETA**2
C
      C1=DT1
      C2=2.*DXI**2
      C3=2.*DETA**2
      C4=2.*DETA
      C5=2.*DXI
      C6=DETA**2
      C7=DXI**2
      C8=4.*DETA*DXI
```

```
C
      XLFAR=XLCOOR+X(NX7,1)
      YLFAR=YLCOOR+Y(NX7,1)
      XSMALL=XSCOOR+X(NX7,1)
      YSMALL=YSCOOR+Y(NX7,1)
C
      GO TO 2210
C
1990  CONTINUE
C
      IF(YESNO3.EQ.'N')GO TO 2210
C
      GO TO 2010
2000  KFLAG1=0
      KFILE9=DVOUT
2010  CONTINUE
C
      IF(KFLAG1.EQ.1)GO TO 2020
      DO 2020 I=1,32
      STARS(I)='***'
2020  CONTINUE
C
      WRITE(KFILE9,2030)(STARS(I),I=1,31)
2030  FORMAT(1H1,T25,25A3/T25,A3,T97,A3/T25,A3,T31,
     1 'THE FOLLOWING IS INFORMATION REGARDING THE ',
     2 'GRID GEOMETRY OF THE',T97,A3/T25,A3,T97,A3)
C
      WRITE(KFILE9,2040)(STARS(I),I=1,8)
2040  FORMAT(1H ,T25,A3,T38,3A3,T50,'VARIABLE AREA ',
     1 'DUCT SECTION',T79,3A3,T97,A3)
C
      WRITE(KFILE9,2050)(STARS(I),I=1,27)
2050  FORMAT(1H ,T25,A3,T97,A3/T25,25A3)
C
      WRITE(KFILE9,2060)NX1,NX2,NX3
2060  FORMAT(1H ,//T28,'THE NUMBER OF GRID POINTS IN ...'
     1 ,'THE INLET STRAIGHT SECTION (NX1) :',T95,I3/T60,
     2 'THE VARIABLE AREA  SECTION (NX2) :',T95,I3/T60,
     3 'THE EXIT STRAIGHT  SECTION (NX3) :',T95,I3)
C
      WRITE(KFILE9,2070)ALENG1,ALENG2,ALENG3
2070  FORMAT(1H ,/T28,'THEIR RESPECTIVE NON-DIMENSIONAL',
     1 ' LENGTHS ... INLET (ALENG1) :',T91,F4.1/T73,
     2 'V.A. (ALENG2)  :',T91,F4.1/T73,'EXIT (ALENG3)  '
     3 ,':',T91,F4.1)
C
      WRITE(KFILE9,2080)Z
2080  FORMAT(1H ,/T28,'THE NON-DIMENSIONAL HEIGHT '
     1 ,'(Z) :',T62,F4.1)
C
      WRITE(KFILE9,2090)H
```

```
2090    FORMAT(1H ,/T28,'THE NON-DIMENSIONAL DEPTH OF THE',
        1 ' THROAT (H) :',T75,F4.1,///)
C
        WRITE(KFILE9,2100)(STARS(I),I=1,31)
2100    FORMAT(1H ,T25,25A3/T25,A3,T97,A3/T25,A3,T31,
        1 'THE FOLLOWING IS INFORMATION REGARDING THE ',
        2 'GRID GEOMETRY OF THE',T97,A3/T25,A3,T97,A3)
C
        WRITE(KFILE9,2110)(STARS(I),I=1,8)
2110    FORMAT(1H ,T25,A3,T38,3A3,T51,'FAR FIELD  EXIT ',
        1 'SECTION',T78,3A3,T97,A3)
C
        WRITE(KFILE9,2050)(STARS(I),I=1,27)
C
        WRITE(KFILE9,2120)YESNO5
2120    FORMAT(1H ,//T28,'IS THE EXIT CONDITION A FAR',
        1 ' FIELD ONE (YESNO5) ? (Y/N) .... : ',T92,A3)
C
        WRITE(KFILE9,2130)LP1,LP2,LP3
2130    FORMAT(1H ,/T28,'THE NUMBER OF GRID POINTS IN ... ',
        1 'THE INLET PORTION (LP1)  :'T88,I3/T61,'THE '
        2 ,'"KNEE" PORTION (LP2) :',T88,I3/T61,'THE EXIT '
        3 ,'PORTION  (LP3)  :',T88,I3)
C
        WRITE(KFILE9,2140)XSCOOR,YSCOOR,ROUND
2140    FORMAT(1H ,//T28,'ALL OF THE THE FOLLOWING (',
        1 'NON-DIMENSIONAL) DIMENSIONS'/T28,'ARE MEASURED'
        2 ,' FROM THE EXIT OF THE NOZZLE ...'//T28,
        3 'THE "KNEE" CENTER COORDINATES (XSCOOR,YSCOOR) :'
        4 ,'  (',T80,F5.2,' , ',F5.2,'  )'/T28,
        5 'THE "KNEE" RADIUS (ROUND) :',T60,F5.2)
C
        WRITE(KFILE9,2150)XLCOOR,YLCOOR,FIELD
2150    FORMAT(1H ,/T28,'THE FAR FIELD CENTER COORDINATES',
        1 ' (XLCOOR,YLCOOR) :  (',T83,F5.2,' , ',F5.2,
        2 '  )'/T28,'THE FAR FIELD RADIUS (FIELD) :'
        3 ,T60,F5.2,///)
C
        WRITE(KFILE9,2160)(STARS(I),I=1,25)
2160    FORMAT(1H ,T25,25A3)
C
        WRITE(KFILE9,2170)NX,NY
2170    FORMAT(1H ,//T28,'THE TOTAL NUMBER OF GRID',
        1 ' POINTS IN ... THE X-DIRECTION (NX) :',T90,I3/
        2 T67,'THE Y-DIRECTION (NY) :',T90,I3,//)
C
        WRITE(KFILE9,2180)(STARS(I),I=1,25)
2180    FORMAT(1H ,T25,25A3/)
C
        KFILE9=SAVEXY
        KFLAG1=KFLAG1+1
```

```
      IF(KFLAG1.EQ.1)GO TO 1700
C
      DO 2190 JJ=1,NY
      J=NY-JJ+1
2190  WRITE(KFILE9,1750)(X(I,J),I=1,NX)
C
      DO 2200 JJ=1,NY
      J=NY-JJ+1
2200  WRITE(KFILE9,1780)(Y(I,J),I=1,NX)
C
2210  CONTINUE
C
      IF(YESNO2.EQ.'Y')CALL PAINT1(X,Y,XMIN,XMAX,YMIN,YMAX)
C
      IF(DVOUT.EQ.5)GO TO 2230
      WRITE(5,2220)ZUP1,ZUP2,ZUP3
2220  FORMAT(1H ,//' ZUP1=',2E13.5,T40,
     1          'ZUP2=',2E13.5,T80,'ZUP3=',2E13.5)
      WRITE(5,2250)ZDOWN1,ZDOWN2,ZDOWN3
      WRITE(5,2260)DT2,ETA2,ITMAX2,JPRNT2,SERIES
2230  CONTINUE
C
C
      PAUSE 'PAUSE HERE TO CHECK PRESSURE DATA FOR ERRORS'
C
C
C.....COMPUTE THE PRESSURE COEFFICIENTS.
C
      IDT=0
      IPRNT=0
C
      WRITE(DVOUT,2240)ZUP1,ZUP2,ZUP3
2240  FORMAT(1H1,'ZUP1=',2E13.5,T40,
     1          'ZUP2=',2E13.5,T80,'ZUP3=',2E13.5)
      WRITE(DVOUT,2250)ZDOWN1,ZDOWN2,ZDOWN3
2250  FORMAT(1H ,'ZDOWN1=',2E13.5,T40,
     1          'ZDOWN2=',2E13.5,T80,'ZDOWN3=',2E13.5)
      WRITE(DVOUT,2260)DT2,ETA2,ITMAX2,JPRNT2,SERIES
2260  FORMAT(1H ,/' DT2=',F9.5,5X,'ETA2=',F9.5,10X,
     1       'ITMAX2=',I5,7X,'JPRNT2=',I5,15X,
     2      A5,1X,'- PNT. BKWD. DIFF.'//)
C
C.....INTERIOR POINT COEFFICIENTS.
C
2270  DO 2280 J=2,NYM1
      DO 2280 I=2,NXM1
      XXI=(X(I+1,J)-X(I-1,J))/C5
      XETA=(X(I,J+1)-X(I,J-1))/C4
      YXI=(Y(I+1,J)-Y(I-1,J))/C5
      YETA=(Y(I,J+1)-Y(I,J-1))/C4
      XJ1(I,J)=XXI*YETA-XETA*YXI
```

```
      ALPHA(I,J)=XETA**2+YETA**2
      BETA(I,J)=XXI*XETA+YXI*YETA
      GAMA2(I,J)=XXI**2+YXI**2
2280  CONTINUE
C
C.....BOTTOM POINT COEFFICIENTS.
C
      J=1
      DO 2290 I=2,NXM1
      XXI=(X(I+1,J)-X(I-1,J))/C5
      YXI=(Y(I+1,J)-Y(I-1,J))/C5
      GAMA2(I,J)=XXI**2+YXI**2
      XETA=(-X(I,J+2)+4.*X(I,J+1)-3.*X(I,J))/C4
      YETA=(-Y(I,J+2)+4.*Y(I,J+1)-3.*Y(I,J))/C4
      XJ1(I,J)=XXI*YETA-XETA*YXI
      ALPHA(I,J)=XETA**2+YETA**2
      BETA(I,J)=XXI*XETA+YXI*YETA
2290  CONTINUE
C
C.....EXIT POINT COEFFICIENTS.
C
      I=NX
      DO 2310 J=2,NYM1
      YETA=(Y(I,J+1)-Y(I,J-1))/C4
      XXI=(3.*X(I,J)-4.*X(I-1,J)+X(I-2,J))/C5
      YXI=(3.*Y(I,J)-4.*Y(I-1,J)+Y(I-2,J))/C5
      XETA=(X(I,J+1)-X(I,J-1))/C4
      ALPHA(I,J)=XETA**2+YETA**2
      BETA(I,J)=XXI*XETA+YXI*YETA
      GAMA2(I,J)=XXI**2+YXI**2
      XJ1(I,J)=XXI*YETA-XETA*YXI
C
      IF(YESNO5.EQ.'Y')GO TO 2300
C
      CEXIT(J)=ALPHA(I,J)*DT2/(XJ1(I,J)*ETA2*DXI*YETA)
      XIETAX(J)=YXI/YETA
      GO TO 2310
C
2300  CONTINUE
      PART2(J)=(X(I,J)-XLFAR)*YETA-Y(I,J)*XETA
      CEXIT(J)=ALPHA(I,J)*DT2*FIELD/(XJ1(I,J)*ETA2*DXI)
      PART5(J)=Y(I,J)*XXI-(X(I,J)-XLFAR)*YXI
2310  CONTINUE
C
C.....BOTTOM RIGHT CORNER COEFFICIENTS.
C
      I=NX
      J=1
      XXI=(3.*X(I,J)-4.*X(I-1,J)+X(I-2,J))/C5
      YXI=(3.*Y(I,J)-4.*Y(I-1,J)+Y(I-2,J))/C5
      YETA=(-Y(I,J+2)+4.*Y(I,J+1)-3.*Y(I,J))/C4
```

```
      XETA=(-X(I,J+2)+4.*X(I,J+1)-3.*X(I,J))/C4
      ALPHA(I,J)=XETA**2+YETA**2
      BETA(I,J)=XETA*XXI+YXI*YETA
      GAMA2(I,J)=XXI**2+YXI**2
      XJ1(I,J)=XXI*YETA-XETA*YXI
C
      IF(YESNO5.EQ.'Y')GO TO 2320
C
      CEXIT(J)=ALPHA(I,J)*DT2/(XJ1(I,J)*ETA2*DXI*YETA)
      XIETAX(J)=YXI/YETA
      GO TO 2330
C
2320  CONTINUE
      PART2(J)=(X(I,J)-XLFAR)*YETA-Y(I,J)*XETA
      CEXIT(J)=ALPHA(I,J)*FIELD*DT2/(XJ1(I,J)*ETA2*DXI)
      PART5(J)=Y(I,J)*XXI-(X(I,J)-XLFAR)*YXI
2330  CONTINUE
C
C.....TOP RIGHT CORNER COEFFICIENTS.
C
      I=NX
      J=NY
      XXI=(3.*X(I,J)-4.*X(I-1,J)+X(I-2,J))/C5
      YXI=(3.*Y(I,J)-4.*Y(I-1,J)+Y(I-2,J))/C5
      YETA=(3.*Y(I,J)-4.*Y(I,J-1)+Y(I,J-2))/C4
      XETA=(3.*X(I,J)-4.*X(I,J-1)+X(I,J-2))/C4
      ALPHA(I,J)=XETA**2+YETA**2
      BETA(I,J)=YETA*YXI+XXI*XETA
      GAMA2(I,J)=XXI**2+YXI**2
      XJ1(I,J)=XXI*YETA-XETA*YXI
C
      IF(YESNO5.EQ.'Y')GO TO 2340
C
      CEXIT(J)=ALPHA(I,J)*DT2/(XJ1(I,J)*ETA2*DXI*YETA)
      XIETAX(J)=YXI/YETA
      GO TO 2350
C
2340  CONTINUE
      PART2(J)=(X(I,J)-XLFAR)*YETA-Y(I,J)*XETA
      CEXIT(J)=ALPHA(I,J)*DT2*FIELD/(XJ1(I,J)*ETA2*DXI)
      PART5(J)=Y(I,J)*XXI-(X(I,J)-XLFAR)*YXI
2350  CONTINUE
C
C.....TOP WALL POINTS COEFFICIENTS.
C
      J=NY
      DO 2360 I=2,NXM1
      XXI=(X(I+1,J)-X(I-1,J))/C5
      YXI=(Y(I+1,J)-Y(I-1,J))/C5
      XETA=(3.*X(I,J)-4.*X(I,J-1)+X(I,J-2))/C4
      YETA=(3.*Y(I,J)-4.*Y(I,J-1)+Y(I,J-2))/C4
```

```
        ALPHA(I,J)=XETA**2+YETA**2
        GAMA2(I,J)=XXI**2+YXI**2
        BETA(I,J)=XXI*XETA+YXI*YETA
        XJ1(I,J)=XXI*YETA-XETA*YXI
2360    CONTINUE
C
C.....EXCESSIVELY USED CONSTANT: CONST(I,J).
C
        DO 2370 J=1,NY
        DO 2370 I=2,NX
        CONST(I,J)=(DT2/(ETA2*XJ1(I,J)))**2
2370    CONTINUE
C
C.....COEFFICIENTS FOR SOFT WALL CALCULATIONS.
C
        DO 2380 I=1,NX7
        CDOWN(I)=CMPLX(0.,0.)
        CUP(I)=CMPLX(0.,0.)
2380    CONTINUE
C
        IF(SOFT.EQ.'NONE')GO TO 2440
        IF(SOFT.EQ.'DOWN')GO TO 2410
        IF(COVER(1).EQ.'H'.OR.ALENG1.EQ.0.0)GO TO 2390
C
        J=NY
        DO 2390 I=2,NX1
        CUP(I)=DT2*SQRT(GAMA2(I,J))
        1      /(XJ1(I,J)*ETA2*ZUP1*DETA)
2390    CONTINUE
C
        IF(COVER(2).EQ.'H')GO TO 2400
        IF(ALENG1.EQ.0.0)NX1=2
C
        J=NY
        DO 2400 I=NX1,NX5
        CUP(I)=DT2*SQRT(GAMA2(I,J))
        1      /(XJ1(I,J)*ETA2*ZUP2*DETA)
2400    CONTINUE
        IF(COVER(3).EQ.'H'.OR.ALENG3.EQ.0.0)GO TO 2410
C
        J=NY
        DO 2410 I=NX5,NX7
        CUP(I)=DT2*SQRT(GAMA2(I,J))
        1      /(XJ1(I,J)*ETA2*ZUP3*DETA)
2410    CONTINUE
C
        IF(SOFT.EQ.'UP')GO TO 2440
        IF(COVER(4).EQ.'H'.OR.ALENG1.EQ.0.0)GO TO 2420
C
        J=1
        DO 2420 I=2,NX1
```

```
      CDOWN(I)=DT2*SQRT(GAMA2(I,J))
     1        /(XJ1(I,J)*ETA2*ZDOWN1*DETA)
2420  CONTINUE
C
      IF(COVER(5).EQ.'H')GO TO 2430
      IF(ALENG1.EQ.0.0)NX1=2
C
      J=1
      DO 2430 I=NX1,NX5
      CDOWN(I)=DT2*SQRT(GAMA2(I,J))
     1        /(XJ1(I,J)*ETA2*ZDOWN2*DETA)
2430  CONTINUE
C
      IF(COVER(6).EQ.'H'.OR.ALENG3.EQ.0.0)GO TO 2440
C
      J=1
      DO 2440 I=NX5,NX7
      CDOWN(I)=DT2*SQRT(GAMA2(I,J))
     1        /(XJ1(I,J)*ETA2*ZDOWN3*DETA)
2440  CONTINUE
C
      IF(YESNO5.EQ.'N')GO TO 2450
      C21=CEXIT(1)+CDOWN(NX)*PART2(1)
      PART22=ALPHA(NX,1)*DT2**2/(ETA2**2*XJ1(NX,1)*DXI)
C     C22=C21+PART2(1)+PART22
      C22=C21+PART2(1)
      C31=CEXIT(NY)+CUP(NX)*PART2(NY)
      PART33=ALPHA(NX,NY)*DT2**2/(ETA2**2*XJ1(NX,NY)*DXI)
C     C32=C31+PART2(NY)+PART33
      C32=C31+PART2(NY)
      GO TO 2460
C
2450  CONTINUE
      C21=CEXIT(1)+CDOWN(NX)
      C22=C21+1.0
      C31=CEXIT(NY)+CUP(NX)
      C32=C31+1.0
C
2460  CONTINUE
C
C.....STABILITY CRITERIA CALCULATIONS FOR THE WHOLE
C.....COMPUTATIONAL GRID EXCLUDING THE I=1 POINTS.
C
      IF(YESNO8.EQ.'N')GO TO 2630
C
      WRITE(DVOUT,2470)
2470  FORMAT(1H ,/' STABILITY CRITERIA :'///' TIME STEP',
     1  ' VALUES FOR INTERIOR POINTS ( INCREASING I ; ',
     2  'DECREASING J ).'/)
C
      DO 2500 JJ=2,NYM1
```

```
      J=NYM1-JJ+2
C
      DO 2480 I=2,NXM1
      DELTAT(I)=ETA2*XJ1(I,J)*DXI*DETA/SQRT(ALPHA(I,J)
     1         *DETA**2+GAMA2(I,J)*DXI**2)
2480  CONTINUE
C
      WRITE(DVOUT,2490)(DELTAT(I),I=2,NXM1)
2490  FORMAT(1H ,10E13.4)
2500  CONTINUE
C
      WRITE(DVOUT,2510)
2510  FORMAT(1H ,/' TIME STEP VALUES FOR THE GRID ',
     1  'POINTS ON THE BOTTOM SURFACE.'/)
C
      J=1
      DO 2520 I=2,NXM1
      DELTAT(I)=ETA2*XJ1(I,J)*DXI*DETA/SQRT(ALPHA(I,J)
     1         *DETA**2+GAMA2(I,J)*DXI**2)
2520  CONTINUE
C
      WRITE(DVOUT,2490)(DELTAT(I),I=2,NXM1)
C
      WRITE(DVOUT,2530)
2530  FORMAT(1H ,/' TIME STEP VALUES FOR THE GRID ',
     1  'POINTS ON THE TOP SURFACE.'/)
C
      J=NY
      DO 2540 I=2,NXM1
      DELTAT(I)=ETA2*XJ1(I,J)*DXI*DETA/SQRT(ALPHA(I,J)
     1         *DETA**2+GAMA2(I,J)*DXI**2)
2540  CONTINUE
C
      WRITE(DVOUT,2490)(DELTAT(I),I=2,NXM1)
C
      WRITE(DVOUT,2550)
2550  FORMAT(1H ,/' TIME STEP VALUES FOR THE EXIT',
     1  ' POINTS ( I=NX ; INCREASING J )'/)
C
      I=NX
      DO 2560 J=2,NYM1
      DELTAT(J)=ETA2*XJ1(I,J)*DXI*DETA/SQRT(ALPHA(I,J)
     1         *DETA**2+GAMA2(I,J)*DXI**2)
2560  CONTINUE
C
      WRITE(DVOUT,2490)(DELTAT(J),J=2,NYM1)
C
      IF(YESNO5.EQ.'Y')GO TO 2590
C
      WRITE(DVOUT,2570)
2570  FORMAT(1H ,/' TIME STEP VALUES FOR THE PLANE',
```

```
     1   ' EXIT WALL TOP RIGHT CORNER.'/)
C
      I=NX
      J=NY
      DELTAT(J)=ETA2*XJ1(I,J)/SQRT(ALPHA(I,J)/DXI**2
     1         -ALPHA(I,J)*XIETAX(J)/(DXI*DETA)
     2         +GAMA2(I,J)/DETA**2)
C
      WRITE(DVOUT,2490)DELTAT(NY)
C
      WRITE(DVOUT,2580)
2580  FORMAT(1H ,/' TIME STEP VALUE FOR THE PLANE',
     1   ' WALL EXIT BOTTOM RIGHT CORNER'/)
C
      I=NX
      J=1
      DELTAT(J)=ETA2*XJ1(I,J)/SQRT(ALPHA(I,J)/DXI**2
     1         +ALPHA(I,J)*XIETAX(J)/(DXI*DETA)
     2         +GAMA2(I,J)/DETA**2)
C
      WRITE(DVOUT,2620)DELTAT(1)
C
      GO TO 2630
2590  CONTINUE
C
      WRITE(DVOUT,2600)
2600  FORMAT(1H ,/' TIME STEP VALUE FOR THE FAR FIELD',
     1   ' EXIT TOP RIGHT CORNER'/)
C
      I=NX
      J=NY
      DELTAT(J)=ETA2*XJ1(I,J)/SQRT(ALPHA(I,J)/DXI**2
     1         +ALPHA(I,J)*PART5(J)/(DXI*DETA
     2         *PART2(J))+GAMA2(I,J)/DETA**2)
C
      WRITE(DVOUT,2490)DELTAT(NY)
C
      WRITE(DVOUT,2610)
2610  FORMAT(1H ,/' TIME STEP VALUE FOR THE FAR FIELD',
     1   ' EXIT BOTTOM RIGHT CORNER'/)
C
      I=NX
      J=1
      DELTAT(J)=ETA2*XJ1(I,J)/SQRT(ALPHA(I,J)/DXI**2
     1         -ALPHA(I,J)*PART5(J)/(DXI*DETA
     2         *PART2(J))+GAMA2(I,J)/DETA**2)
C
      WRITE(DVOUT,2620)DELTAT(1)
2620  FORMAT(1H ,E13.4,/'1')
2630  CONTINUE
C
```

```
C.....COMPLETE PRINT OUT OF ALL PRESSURE
C.....COEFFICIENTS AND IMPORTANT CONSTANTS.
C
      IF(ALLOUT.EQ.'N')GO TO 2870
C
      WRITE(DVOUT,2640)
2640  FORMAT(1H ,'COMPLETE PRINTOUT OF ALL THE IMPORTANT'
     1  ,' PRESSURE COEFFICIENTS AND CONSTANTS.')
      WRITE(DVOUT,2650)
2650  FORMAT(1H ,//' ALPHA(I,J) ; I AND J INCREASING')
      DO 2660 J=1,NY
2660  WRITE(DVOUT,2670)(ALPHA(I,J),I=1,NX)
2670  FORMAT(1H ,12E11.3)
      WRITE(DVOUT,2680)
2680  FORMAT(1H ,/' BETA(I,J) ; I AND J INCREASING')
      DO 2690 J=1,NY
2690  WRITE(DVOUT,2670)(BETA(I,J),I=1,NX)
      WRITE(DVOUT,2700)
2700  FORMAT(1H ,/' GAMA2(I,J) ; I & J INCREASING')
      DO 2710 J=1,NY
2710  WRITE(DVOUT,2670)(GAMA2(I,J),I=1,NX)
      WRITE(DVOUT,2720)
2720  FORMAT(1H ,/' XJ1(I,J) ; I AND J INCREASING')
      DO 2730 J=1,NY
2730  WRITE(DVOUT,2670)(XJ1(I,J),I=1,NX)
      WRITE(DVOUT,2740)
2740  FORMAT(1H ,/' CONST(I,J) ; I $ J INCRESSING')
      DO 2750 J=1,NY
2750  WRITE(DVOUT,2670)(CONST(I,J),I=1,NX)
      WRITE(DVOUT,2760)
2760  FORMAT(1H ,/' XIETAX(J) ; J INCREAS.')
      WRITE(DVOUT,2670)(XIETAX(J),J=1,NY)
      WRITE(DVOUT,2770)
2770  FORMAT(1H ,/' PART2(J) ; J INCREAS.')
      WRITE(DVOUT,2670)(PART2(J),J=1,NY)
      WRITE(DVOUT,2780)
2780  FORMAT(1H ,/' PART5(J)')
      WRITE(DVOUT,2670)(PART5(J),J=1,NY)
      WRITE(DVOUT,2790)
2790  FORMAT(1H ,/' CEXIT(J)')
      WRITE(DVOUT,2670)(CEXIT(J),J=1,NY)
      WRITE(DVOUT,2800)
2800  FORMAT(1H ,/' CDOWN(I) ; I INCREAS.')
      WRITE(DVOUT,2670)(CDOWN(I),I=1,NX)
      WRITE(DVOUT,2810)
2810  FORMAT(1H ,/' CUP(I)')
      WRITE(DVOUT,2670)(CUP(I),I=1,NX)
      WRITE(DVOUT,2820)SOFT
2820  FORMAT(1H ,/' SOFT: ',A5)
      WRITE(DVOUT,2830)C21,C22,C31,C32
2830  FORMAT(1H ,/' C21=',2E11.4,4X,'C22=',2E11.4
```

```
     1     ,4X,'C31=',2E11.4,4X,'C32=',2E11.4)
      WRITE(DVOUT,2840)ETA2,DETA,DXI
2840  FORMAT(1H ,/' ETA2=',E11.4,4X,'DETA=',E11.4,
     1   4X,'DXI=',E11.4)
      WRITE(DVOUT,2850)NX,NXM1,NY,NYM1,NX5,NX7,NX1,DT2
2850  FORMAT(1H ,/' NX',I4,4X,'NXM1',I4,4X,'NY',I4,
     1   4X,'NYM1',I4,4X,'NX5',I4,4X,'NX7',I4,4X,
     2   'NX1',I4,8X,'DT2',F8.4)
      WRITE(DVOUT,2860)C1,C2,C3,C4,C5,C6,C7,C8
2860  FORMAT(1H ,/' C1',F8.4,4X,'C2',F8.4,4X,'C3',F8.4
     1   ,4X,'C4',F8.4,4X,'C5',F8.4,4X,'C6',F8.4,4X,
     2   'C7',F8.4,4X,'C8',F8.4,//)
2870  CONTINUE
C
C.....INITIALIZE TO LAST TWO PRESSURE LEVELS
C.....SAVED FROM A PREVIOUS RUN FOR CONTINUATION.
C
      IF(ABSINI.EQ.'ABS1')GO TO 2900
C
      DO 2880 JJ=1,NY
      J=NY-JJ+1
2880  READ(PRDATA,3330)(P(I,J),I=1,NX)
C
      DO 2890 JJ=1,NY
      J=NY-JJ+1
2890  READ(PRDATA,3330)(PO(I,J),I=1,NX)
C
      READ(PRDATA,3340)IDT
C
      GO TO 2920
C
2900  CONTINUE
C
C.....THE INITIAL CONDITIONS.
C
      DO 2910 I=1,NX
      DO 2910 J=1,NY
      P(I,J)=CMPLX(1.,0.)
2910  PO(I,J)=CMPLX(1.,0.)
C
C.....START THE PRESSURE SOLUTION.
C
2920  CONTINUE
C
C.....THE INLET BOUNDARY CONDITION.
C
      TIME=IDT*DT2
      PZ2=TWOPI*TIME
      PZ=CMPLX(0.,PZ2)
      DO 2930 J=1,NY
2930  P(1,J)=CEXP(PZ)
```

```
C
C.....INTERIOR POINTS.
C
      DO 2940 I=2,NXM1
      DO 2940 J=2,NYM1
      P1=P(I-1,J)
      P2=P(I,J+1)
      P3=P(I+1,J)
      P4=P(I,J-1)
      P5=P(I,J)
      P6=P(I-1,J+1)
      P7=P(I+1,J+1)
      P8=P(I+1,J-1)
      P9=P(I-1,J-1)
      PXI2=(P3-2.*P5+P1)/C7
      PETA2=(P2-2.*P5+P4)/C6
      PETAXI=(P7+P9-P6-P8)/C8
      PN(I,J)=2.*P5-PO(I,J)+CONST(I,J)*(ALPHA(I,J)*PXI2
     1       -2.*BETA(I,J)*PETAXI+GAMA2(I,J)*PETA2)
2940  CONTINUE
C
C.....BOTTOM SURFACE.
C
      J=1
      DO 2950 I=2,NXM1
      CDOWN1=CDOWN(I)+1.
      P1=P(I-1,J)
      P2=P(I,J+1)
      P3=P(I+1,J)
      P5=P(I,J)
      P6=P(I-1,J+1)
      P7=P(I+1,J+1)
      PXI=(P3-P1)/C5
      PXI2=(P3-2.*P5+P1)/C7
      PETAXI=(P7+P1-P6-P3)/C8
      PETA=(P2-P5)/DETA
      PN(I,J)=(2.*P5+PO(I,J)*(CDOWN(I)-1.)+CONST(I,J)
     1       *(ALPHA(I,J)*PXI2-4.*BETA(I,J)*PETAXI
     2       -2.*BETA(I,J)*PXI/DETA
     3       +2.*GAMA2(I,J)*PETA/DETA))/CDOWN1
2950  CONTINUE
C
C.....BOTTOM RIGHT CORNER.
C
      I=NX
      J=1
      P1=P(I-1,J)
      P2=P(I,J+1)
      P5=P(I,J)
      P6=P(I-1,J+1)
      PETAXI=(P2-P6+P1-P5)/C8
```

```
C
      IF(SERIES.EQ.'THREE')GO TO 2960
      PETA=(P2-P5)/DETA
      PXI=(P5-P1)/DXI
      GO TO 2970
C
2960  CONTINUE
      PETA=(-P(I,J+2)+4.*P(I,J+1)-3.*P(I,J))/C4
      PXI=(3.*P(I,J)-4.*P(I-1,J)+P(I-2,J))/C5
2970  CONTINUE
C
      IF(YESNO5.EQ.'N')GO TO 2980
      PARTP=PETA*PART5(J)
      PN(I,J)=(PART2(J)*(2.*P5-PO(I,J))+C21*PO(I,J)
     1      +CONST(I,J)*(-2.*ALPHA(I,J)/DXI*(PARTP
     2      +PART2(J)*PXI)+PART2(J)*(-8.*BETA(I,J)
     3      *PETAXI+2.*GAMA2(I,J)/DETA*(PETA-BETA(I,J)
     4      *PXI/GAMA2(I,J)))))/C22
      GO TO 2990
C
2980  CONTINUE
      PN(I,J)=(2.*P5+PO(I,J)*(C21-1.)+CONST(I,J)
     1      *(2.*ALPHA(I,J)/DXI*(XIETAX(J)*PETA-PXI)
     2      -8.*BETA(I,J)*PETAXI+2.*GAMA2(I,J)
     3      /DETA*(PETA-BETA(I,J)*PXI/GAMA2(I,J))))/C22
2990  CONTINUE
C
C.....TOP SURFACE.
C
      J=NY
      DO 3000 I=2,NXM1
      CUP1=CUP(I)+1.
      P1=P(I-1,J)
      P3=P(I+1,J)
      P4=P(I,J-1)
      P5=P(I,J)
      P8=P(I+1,J-1)
      P9=P(I-1,J-1)
      PXI2=(P3-2.*P5+P1)/C7
      PXI=(P3-P1)/C5
      PETA=(P5-P4)/DETA
      PETAXI=(P3+P9-P1-P8)/C8
      PN(I,J)=(2.*P5+PO(I,J)*(CUP(I)-1.)+CONST(I,J)
     1      *(ALPHA(I,J)*PXI2-4.*BETA(I,J)*PETAXI
     2      +2.*BETA(I,J)*PXI/DETA
     3      -2.*GAMA2(I,J)*PETA/DETA))/CUP1
3000  CONTINUE
C
C.....TOP RIGHT CORNER.
C
      J=NY
```

```
          I=NX
          P1=P(I-1,J)
          P4=P(I,J-1)
          P5=P(I,J)
          P9=P(I-1,J-1)
          PETAXI=(P5-P4+P9-P1)/C8
C
          IF(SERIES.EQ.'THREE')GO TO 3010
          PETA=(P5-P4)/DETA
          PXI=(P5-P1)/DXI
          GO TO 3020
C
3010      CONTINUE
          PETA=(3.*P(I,J)-4.*P(I,J-1)+P(I,J-2))/C4
          PXI=(3.*P(I,J)-4.*P(I-1,J)+P(I-2,J))/C5
3020      CONTINUE
C
          IF(YESNO5.EQ.'N')GO TO 3030
          PARTP=PETA*PART5(J)
          PN(I,J)=(PART2(J)*(2.*P5-PO(I,J))+C31*PO(I,J)
          1       +CONST(I,J)*(-2.*ALPHA(I,J)/DXI*(PARTP
          2       +PXI*PART2(J))+PART2(J)*(-8.*BETA(I,J)
          3       *PETAXI+2.*GAMA2(I,J)/DETA*(BETA(I,J)*PXI
          4       /GAMA2(I,J)-PETA))))/C32
          GO TO 3040
C
3030      CONTINUE
C
          PN(I,J)=(2.*P5+PO(I,J)*(C31-1.)+CONST(I,J)
          1       *(2.*ALPHA(I,J)/DXI*(XIETAX(J)*PETA-PXI)
          2       -8.*BETA(I,J)*PETAXI+2.*GAMA2(I,J)
          3       /DETA*(BETA(I,J)*PXI/GAMA2(I,J)-PETA)))/C32
3040      CONTINUE
C
C.....EXIT PLANE.
C
          I=NX
          DO 3080 J=2,NYM1
          P1=P(I-1,J)
          P2=P(I,J+1)
          P4=P(I,J-1)
          P5=P(I,J)
          P6=P(I-1,J+1)
          P9=P(I-1,J-1)
C
          IF(SERIES.EQ.'THREE')GO TO 3050
          PXI=(P5-P1)/DXI
          GO TO 3060
C
3050      CONTINUE
          PXI=(3.*P(I,J)-4.*P(I-1,J)+P(I-2,J))/C5
```

```
3060     CONTINUE
C
         PETA=(P2-P4)/C4
         PETA2=(P2-2.*P5+P4)/C6
         PETAXI=(P2-P4+P9-P6)/C8
C
         IF(YESNO5.EQ.'N')GO TO 3070
         PART9=ALPHA(I,J)*DT2**2/(ETA2**2*XJ1(I,J)*DXI)
C        C41=CEXIT(J)+PART2(J)+PART9
         C41=CEXIT(J)+PART2(J)
         PARTP=PETA*PART5(J)
         PN(I,J)=(PART2(J)*(2.*P5-PO(I,J))+CEXIT(J)
         1       *PO(I,J)+CONST(I,J)*(-2.*ALPHA(I,J)/DXI
         2       *(PARTP+PXI*PART2(J))+PART2(J)*(-4.
         3       *BETA(I,J)*PETAXI+GAMA2(I,J)*PETA2)))/C41
         GO TO 3080
C
3070     CONTINUE
C
         C41=CEXIT(J)+1.
         PN(I,J)=(2.*P5+PO(I,J)*(CEXIT(J)-1.)+CONST(I,J)
         1       *(2.*ALPHA(I,J)/DXI*(XIETAX(J)*PETA-PXI)
         2       -4.*BETA(I,J)*PETAXI+GAMA2(I,J)*PETA2))/C41
3080     CONTINUE
C
C.....INCREMENT THE TIME STEP (IDT),
C.....UPDATE THE PRESENT PRESSURES (P),
C.....AND THE OLD PRESSURES (PO),
C.....CHECK FOR DIVERGENCE, AND
C.....RERUN THE PRESSURE SOLUTION.
C
         IPRNT=IPRNT+1
         IDT=IDT+1
C
         DO 3100 I=2,NX
         DO 3100 J=1,NY
C
         IF(CABS(P(I,J)).LT.50.)GO TO 3090
         STOP ' STOP...DIVERGENT SOLUTION...TERMINATE. '
3090     CONTINUE
C
         PO(I,J)=P(I,J)
         P(I,J)=PN(I,J)
3100     CONTINUE
C
C.....PRINT THE ABSOLUTE PRESSURES OF THE
C.....BOTTOM ROW (J=1) AT EVERY TIME STEP.
C
         J=1
         DO 3110 I=1,NX2
         II=JX1+I
```

```
3110    VEC(I)=CABS(P(II,J))
        WRITE(DVOUT,3120)IDT,(VEC(I),I=1,NX2)
3120    FORMAT(1H ,I6,6X,2(12F10.5))
C
C.....PRINT THE REAL, IMAGINARY, AND ABSOLUTE
C.....PRESSURES FOR THE WHOLE GRID EVERY
C....."JPRNT2" TIME STEPS, AND AFTER THE
C.....LAST TIME STEP (ITMAX2).
C
        IF(IDT .EQ. ITMAX2)GO TO 3130
        IF(IPRNT .EQ. JPRNT2)GO TO 3150
        GO TO 2920
C
3130    WRITE(DVOUT,3140)
3140    FORMAT(1H ,/' MAXIMUM TIME STEPS COMPLETED.')
C
3150    IPRNT=0
C
        WRITE(DVOUT,3160)IDT
3160    FORMAT(1H ,/' IDT=',I5,20X,'PRESSURE FUNCTIONS')
C
        IF(YESNO7.EQ.'N')GO TO 3240
C
        DO 3170 I=1,NX
        DO 3170 J=1,NY
        DXS(I,J)=REAL(P(I,J))
3170    DYS(I,J)=AIMAG(P(I,J))
C
        WRITE(DVOUT,3180)
3180    FORMAT(1H ,/' REAL PRESSURES...')
C
        DO 3190 JJ=1,NY
        J=NY+1-JJ
3190    WRITE(DVOUT,3200)(DXS(I,J),I=1,NX)
3200    FORMAT(1H ,13F10.5)
C
        WRITE(DVOUT,3210)
3210    FORMAT(1H ,/' IMAGINARY PRESSURES...')
C
        DO 3220 JJ=1,NY
        J=NY+1-JJ
3220    WRITE(DVOUT,3230)(DYS(I,J),I=1,NX)
3230    FORMAT(1H ,13F10.5)
C
3240    DO 3250 I=1,NX
        DO 3250 J=1,NY
3250    DXS(I,J)=CABS(P(I,J))
        WRITE(DVOUT,3260)
3260    FORMAT(1H ,/' ABSOLUTE PRESSURES...')
C
        DO 3270 JJ=1,NY
```

```
        J=NY+1-JJ
3270    WRITE(DVOUT,3200)(DXS(I,J),I=1,NX)
C
        WRITE(DVOUT,3280)
3280    FORMAT(1H ,/)
C
        IF(IDT .LT. ITMAX2)GO TO 2920
C
C.....IF YESNO9 EQUALS 'Y' (YES), CALL
C.....SUBROUTINE "PLOT2" TO PLOT THE ABSOLUTE
C.....PRESSURE VERSUS DISTANCE "X" ALONG THE
C.....BOTTOM WALL OF THE VARIABLE AREA DUCT.
C
        IF(YESNO9.EQ.'N')GO TO 3300
C
        WRITE(5,3290)PLTFIL
        WRITE(DVOUT,3290)PLTFIL
3290    FORMAT(1H ,///' A PLOT OF THE ABSOLUTE ',
        1  'PRESSURE VERSUS DISTANCE "X" ON THE '/
        2  ' BOTTOM WALL OF THE VARIABLE AREA DUCT',
        3  ' IS IN FILE FOR',I2,'.PLT .')
C
C
        CALL PLOT2(VEC,NX2,PLTFIL)
C
C
3300    CONTINUE
C
C.....IF YESNO4 EQUALS 'Y' (YES), SAVE THE LAST
C.....TWO PRESSURE LEVELS, AND THE LAST TIME STEP
C.....FOR FUTURE CONTINUATION.
C
        IF(YESNO4.EQ.'N')GO TO 3350
C
        DO 3310 JJ=1,NY
        J=NY-JJ+1
3310    WRITE(SAVPP0,3330)(P(I,J),I=1,NX)
C
        DO 3320 JJ=1,NY
        J=NY-JJ+1
3320    WRITE(SAVPP0,3330)(P0(I,J),I=1,NX)
3330    FORMAT(1H ,14F9.5)
C
        WRITE(SAVPP0,3340)IDT
3340    FORMAT(1H ,/I5)
C
C.....END OF MAIN PROGRAM.
C
3350    STOP
        END
C
```

```
C#########################################################
C#########################################################
C##                                                     ##
C##   SUBROUTINE PAINT1 USES THE X AND Y VALUES OF THE  ##
C##   GRID POINTS TO DRAW THE INITIAL CONDITION AND/OR  ##
C##   THE FINAL SOLUTION OF THE COMPUTATIONAL GRID OF   ##
C##   THE GEOMETRIC DOMAIN UNDER INVESTIGATION. PAINT1  ##
C##   DRAWS THE CONSTANT J COORDINATE LINES FIRST, AND  ##
C##   THEN THE CONSTANT I COORDINATE LINES.             ##
C##   THE GRAPHICS PACKAGE USED IS THE TCS (TEKTRONIX   ##
C##   CONTROL SYSTEM), AND THE OUTPUT SHOULD BE PLOTTED ##
C##   ON A FLATBED PLOTTER WHILE THE PROGRAM IS RUNNING.##
C##   THE DESCRIPTION OF THE INPUT PARAMETERS FOLLOWS.  ##
C##                                                     ##
C##      X:  ARRAY CONTAINING X-VALUES OF GRID POINTS   ##
C##      Y:  ARRAY CONTAINING Y-VALUES OF GRID POINTS   ##
C##   XMIN:  MINIMUM HORIZONTAL USER COORDINATE         ##
C##   XMAX:  MAXIMUM HORIZONTAL USER COORDINATE         ##
C##   YMIN:  MINIMUM VERTICAL USER COORDINATE           ##
C##   YMAX:  MAXIMUM VERTICAL USER COORDINATE           ##
C##     NX:  TOTAL NUMBER OF GRID POINTS IN X-DIRECTION ##
C##     NY:  TOTAL NUMBER OF GRID POINTS IN Y-DIRECTION ##
C##                                                     ##
C#########################################################
C
      SUBROUTINE PAINT1(X,Y,XMIN,XMAX,YMIN,YMAX)
      DIMENSION X(90,30),Y(90,30)
      COMMON/DUBS1/NX,NY
C
      CALL INITT(120)
C
      CALL ANMODE
      WRITE(5,10)
10    FORMAT(1HO,'GET THE PLOTTER READY WHEN USING',
     1          ' ONE. PRESS "RETURN" WHEN READY.')
      CALL TINPUT(LLD)
      CALL HOME
      CALL NEWPAG
C
      LENY=980.*ABS((YMAX-YMIN)/(XMAX-XMIN))
C
      CALL DWINDO(XMIN,XMAX,YMIN,YMAX)
      CALL SWINDO(20,980,20,LENY)
C
      DO 20 J=1,NY
      CALL MOVEA(X(1,J),Y(1,J))
      DO 20 I=2,NX
20    CALL DRAWA(X(I,J),Y(I,J))
C
      CALL HOME
      CALL BELL
```

```
        CALL TINPUT(LLD)
C
        DO 30 I=1,NX
        CALL MOVEA(X(I,1),Y(I,1))
        DO 30 J=2,NY
30      CALL DRAWA(X(I,J),Y(I,J))
C
        CALL BELL
        CALL BELL
C
        CALL TINPUT(LLD)
        RETURN
        END
C
C#########################################################
C##                                                     ##
C##  SUBROUTINE PLOT2 PLOTS THE ABSOLUTE PRESSURE VALUE  ##
C##  VERSUS THE NON-DIMENSIONAL AXIAL POSITION ON THE    ##
C##  BOTTOM WALL OF THE DUCT UNDER INVESTIGATION.        ##
C##  THE GRAPHICS PACKAGE USED IS THE CALCOMP, AND THE   ##
C##  RESULTING GRAPH IS OUTPUT ON A DISK FILE. IT CAN    ##
C##  BE PREVIEWED ON THE SCREEN, OR PLOTTED ON THE       ##
C##  FLATBED PLOTTER BY RUNNING PRECAL.                  ##
C##  THE DESCRIPTION OF THE INPUT PARAMETERS FOLLOWS.    ##
C##                                                      ##
C##    PAXIS:  ARRAY CONTAINING ABS. PRESSURE VALUES     ##
C##      NX2:  NUMBER OF GRID POINTS IN TEST SECTION     ##
C##   PLTFIL:  OUTPUT FILE DEVICE NUMBER                 ##
C##                                                      ##
C#########################################################
C
        SUBROUTINE PLOT2(PAXIS,NX2,PLTFIL)
        DIMENSION PAXIS(25),XAXIS(25)
        INTEGER PLTFIL
C
        DO 10 I=1,NX2
        XAXIS(I)=(I-1.)/(NX2-1.)
10      CONTINUE
C
        XAXIS(NX2+1)=0.0
        XAXIS(NX2+2)=0.15
C
        CALL PLOTS(0,0,PLTFIL)
        CALL NEWPEN(2)
C
        CALL PLOT(1.,1.,-3)
C
        CALL SCALE(PAXIS,5.,NX2,1)
C
        CALL IAXIS(0.,0.,' ',-1001,7.,0.,XAXIS(NX2+1),
        1       XAXIS(NX2+2),2)
```

```
      CALL IAXIS(0.,0.,' ',1001,5.,90.,PAXIS(NX2+1),
     1        PAXIS(NX2+2),2)
C
      CALL FLINE(XAXIS,PAXIS,-NX2,1,1,2)
C
      CALL PLOT(10.,10.,999)
C
      RETURN
      END
C
C#########################################################
C##                                                     ##
C##  SUBROUTINE GOPRO1 PERFORMS A ONE-SIDED GEOMETRIC   ##
C##  PROGRESSION ON A SPECIFIED SEGMENT. THE FOLLOWING  ##
C##  IS A DESCRIPTION OF THE INPUT PARAMETERS.          ##
C##                                                     ##
C##       X1:  THE POSITION OF ONE EXTREME OF SEGMENT   ##
C##       X2:  THE POSITION OF THE OTHER SEGMENT END    ##
C##       NP:  NUMBER OF DIVISIONS ON THE SEGMENT       ##
C##       XA:  FIRST SPACING INCREMENT                  ##
C##      LOC:  IF =1 INDICATES THAT SMALLEST INCREMENT  ##
C##             IS NEXT TO X1 POSITION. IF =2 THEN NEXT ##
C##              TO X2 POSITION                         ##
C##       DX:  OUTPUT ARRAY CONTAINING THE GRID         ##
C##             INCREMENT DISTRIBUTION                  ##
C##      RPN:  GEOMETRIC PROGRESSION RATIO NUMBER       ##
C##    DVOUT:  OUTPUT FILE DEVICE NUMBER                ##
C##   YESNO6:  IF YES THEN PRINT OUT DX AND RPN         ##
C##                                                     ##
C#########################################################
C
      SUBROUTINE GOPRO1(X1,X2,NP,XA,LOC)
      DIMENSION  DX(50)
      INTEGER    DVOUT
      COMMON/DUBS2/DX,YESNO6,DVOUT
      COMMON/DUBS3/RPN
C
      IF(LOC.GT.2)STOP
      NPM1=NP-1
C
10    CONTINUE
      XS=ABS(X2-X1)
      C9=XS/(XA*NP)
      IKT=0
      IF(ABS(C9-1.).LT.1.E-6)GO TO 30
      RP=C9**(1./(NP-1.))+1.
      IF(RP.LT.2.)RP=RP-2.
20    F=RP**NP-1.-XS/XA*(RP-1.)
      FP=NP*RP**(NP-1)-XS/XA
      RPN=RP-F/FP
      IKT=IKT+1
```

```
      IF(IKT.GT.500)GO TO 110
      IF(ABS(RPN-RP).LT.1.E-6)GO TO 40
      RP=RPN
      GO TO 20
30    RPN=1.
C
40    CONTINUE
      IF(LOC.EQ.2)GO TO 50
      DX(1)=XA
      GO TO 60
50    DX(1)=XA*RPN**(NP-1.)
      RPN=1./RPN
C
60    CONTINUE
C
      DO 70 II=1,NPM1
      DX(II+1)=DX(II)*RPN
70    K=II+1
C
      IF(YESNO6.EQ.'N')GO TO 130
C
      WRITE(DVOUT,80)RPN,IKT
80    FORMAT(1H ,//' RATIO OF TYPE 1 GEOMETRIC ',
     1     'PROGRESSION=',F12.6,4X,' ITERATIONS='I4)
      WRITE(DVOUT,90)
90    FORMAT(1H ,/' GRID INCREMENT DISTRIBUTION')
      WRITE(DVOUT,100)(DX(K1),K1=1,K)
100   FORMAT(1H ,11F11.7)
C
      GO TO 130
110   WRITE(5,120)
120   FORMAT(1H ,/' NON-CONVERGENCE OF THE GEOMETRIC'
     1     ,' PROGRESSION OF TYPE: 1 ...STOP.')
      STOP
C
130   RETURN
      END
C
C#############################################################
C##                                                         ##
C##  SUBROUTINE GOPRO2 PERFORMS A TWO-SIDED GEOMETRIC       ##
C##  PROGRESSION ON A GIVEN SEGMENT. THE FOLLOWING IS       ##
C##  A DESCRIPTION OF THE INPUT PARAMETERS.                 ##
C##                                                         ##
C##       X1:  POSITION OF ONE SEGMENT END                  ##
C##       X2:  POSITION OF THE OTHER END                    ##
C##      XA1:  FIRST INCREMENT ON X1 SIDE                   ##
C##      XA2:  FIRST INCREMENT ON X2 SIDE                   ##
C##     JNP1:  NUMBER OF INCREMENTS FROM X1 SIDE            ##
C##     JNP2:  NUMBER OF INCREMENTS FROM X2 SIDE            ##
C##       DX:  OUTPUT ARRAY CONTAINING GRID                 ##
```

```
C##              INCREMENT DISTRIBUTION                          ##
C##     DVOUT:   OUTPUT FILE DEVICE NUMBER                       ##
C##    YESNO6:   IF YES THEN OUTPUT DX AND RPN'S                 ##
C##                                                              ##
C##############################################################
C
      SUBROUTINE GOPRO2(X1,X2,XA1,XA2,JNP1,JNP2)
      DIMENSION  DX(50)
      INTEGER    DVOUT
      COMMON/DUBS2/DX,YESNO6,DVOUT
C
      IF(YESNO6.EQ.'N')GO TO 20
      WRITE(DVOUT,10)X1,X2,JNP1,JNP2,XA1,XA2
10    FORMAT(1H ,//' X1=',F11.7,2X,'X2=',F11.7,2X,'JNP1=',
     1 I3,2X,'JNP2=',I3,2X,'XA1=',F11.7,2X,'XA2=',F11.7)
C
20    CONTINUE
C
      IF(JNP1 .EQ. 1 .OR. JNP2 .EQ. 1)GO TO 160
      JNP1M1=JNP1-1
      JNP2M1=JNP2-1
      I=0
      IKT=0
      XT=ABS(X2-X1)
      XC1=XT/XA2
      XC2=XA1/XA2
      XC3=(XA1/XA2)**(1./(JNP2-1.))
      XC4=(JNP1-1.)/(JNP2-1.)
      XC5=(XA1/XA2)**(JNP2/(JNP2-1.))
      XC6=JNP2*(JNP1-1.)/(JNP2-1.)
      NFLIPS=0
C
      IF((X2-X1)*.5/XA1 .GT. JNP1)RP=3.
      IF((X2-X1)*.5/XA1 .LE. JNP1)RP=0.5
C
30    XC7=XC2*(RP**JNP1-1.)*(XC3*RP**XC4-1.)
      XC8=(XC5*RP**XC6-1.)*(RP-1.)
      XC9=-XC1*(RP-1.)*(XC3*RP**XC4-1.)
      F=XC7+XC8+XC9
      XC7=XC2*(RP**JNP1-1.)*XC3*XC4*RP**(XC4-1.)
      XC8=XC2*JNP1*RP**(JNP1-1.)*(XC3*RP**XC4-1.)
      XC9=XC5*RP**XC6-1.+XC5*XC6*RP**(XC6-1.)
      XC10=-XC1*(RP-1.)*XC3*XC4*RP**(XC4-1.)
      XC11=-XC1*(XC3*RP**XC4-1.)
      FP=XC7+XC8+XC9+XC10+XC11
      RPN=RP-F/FP
      IF(RPN .LT. 0.)RPN=1.E-5
      IKT=IKT+1
C
      IF(ABS(RPN-RP) .LT. 1.E-6)GO TO 40
      IF(IKT .GT. 500)GO TO 130
```

```
      IF(RPN .LT. 1.E-5 .OR. RPN .GT. 100.)GO TO 130
C
      RP=RPN
      GO TO 30
C
40    CONTINUE
      IF(YESNO6.EQ.'N')GO TO 60
      WRITE(DVOUT,50)RPN,IKT
50    FORMAT(1H ,'RPN1=',F12.6,8X,'IKT1=',I5)
C
60    CONTINUE
      DX(I+1)=XA1
      K1=I+1
      R1=RPN
      R2=R1**((JNP1-1.)/(JNP2-1.))*(XA1/XA2)**(1./(JNP2-1.))
C
      IF(YESNO6.EQ.'N')GO TO 80
      WRITE(DVOUT,70)R2,IKT
70    FORMAT(1H ,'RPN2=',F12.6,8X,'IKT2=',I5)
C
80    CONTINUE
C
      DO 90 II=1,JNP1M1
      DX(I+II+1)=DX(I+II)*R1
90    K=I+II+1
C
      DX(K+1)=XA2*R2**(JNP2-1.)
      I=K
      R2=1./R2
C
      DO 100 II=1,JNP2M1
      DX(I+II+1)=DX(I+II)*R2
100   K=I+II+1
C
      K2=K
      I=K
      IF(YESNO6.EQ.'N')GO TO 180
      WRITE(DVOUT,110)
110   FORMAT(1H ,/' GRID INCREMENT DISTRIBUTION')
      WRITE(DVOUT,120)(DX(K3),K3=K1,K2)
120   FORMAT(1H ,11F11.7)
      GO TO 180
C
130   CONTINUE
      NFLIPS=NFLIPS+1
      IF(NFLIPS .EQ. 1)GO TO 150
C
      WRITE(5,140)
140   FORMAT(1H ,/' NON-CONVERGENCE OF THE GEOMETRIC'
     1      ,' PROGRESSION OF TYPE: 2 ...STOP.')
      STOP
```

```
C
150     IF(RPN .LT. 1.)RP=3.
        IF(RPN .GE. 1.)RP=0.5
C
        I=0
        IKT=0
        GO TO 30
C
160     WRITE(DVOUT,170)
170     FORMAT(1H ,/' NP=1,ITYPE=2')
        STOP
C
180     RETURN
        END
```

APPENDIX F

FIGURES

This appendix contains all the figures referred to in this text. The majority of the plots and graphs were produced on a TEKTRONIX flatbed plotter.

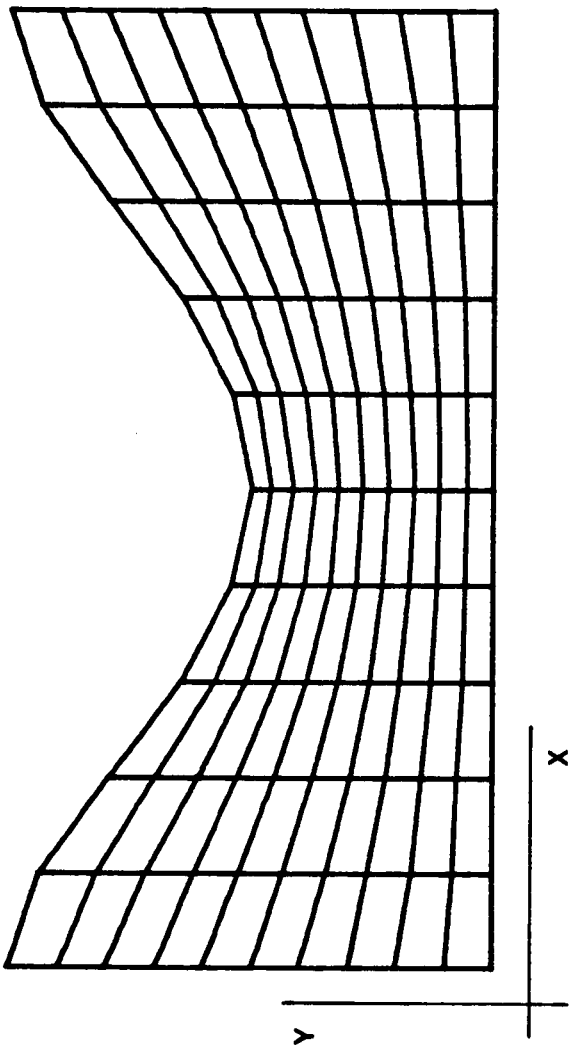


Figure 1. Variable area duct in the physical (X-Y) plane.

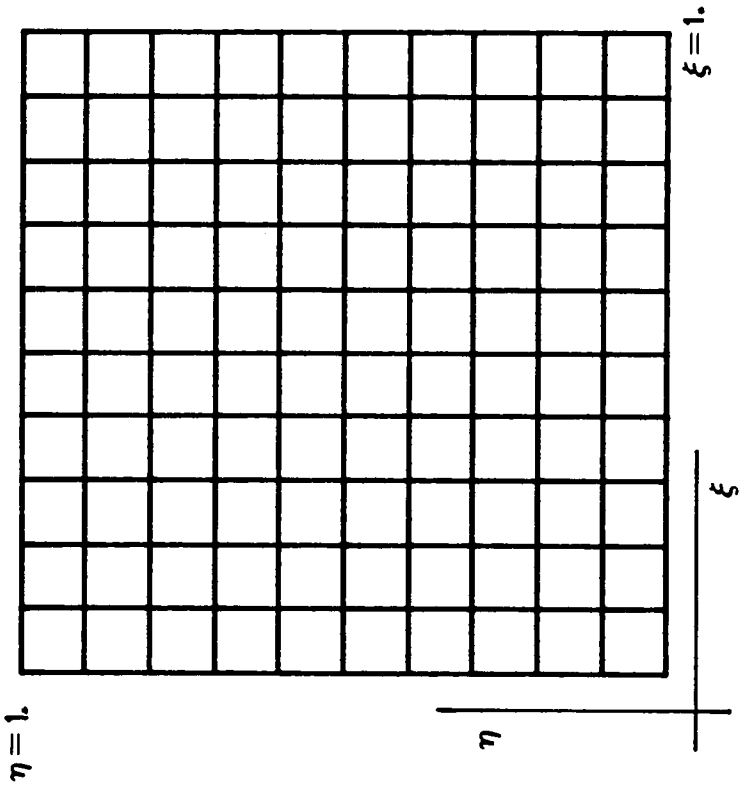


Figure 2. Computational grid in the transformed (ξ-η) plane.

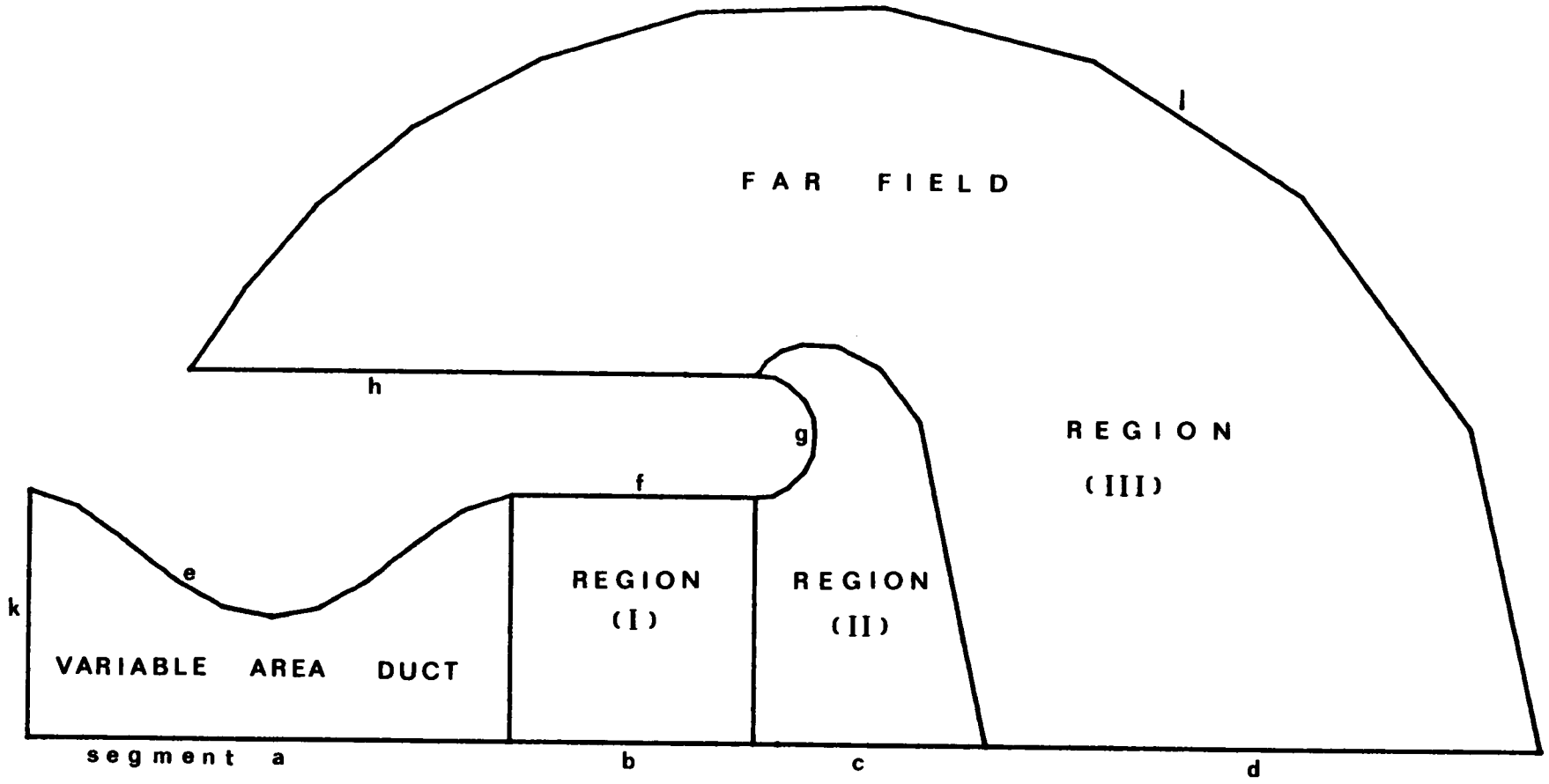


Figure 3. Variable area duct and far field: Regions and segments definitions.

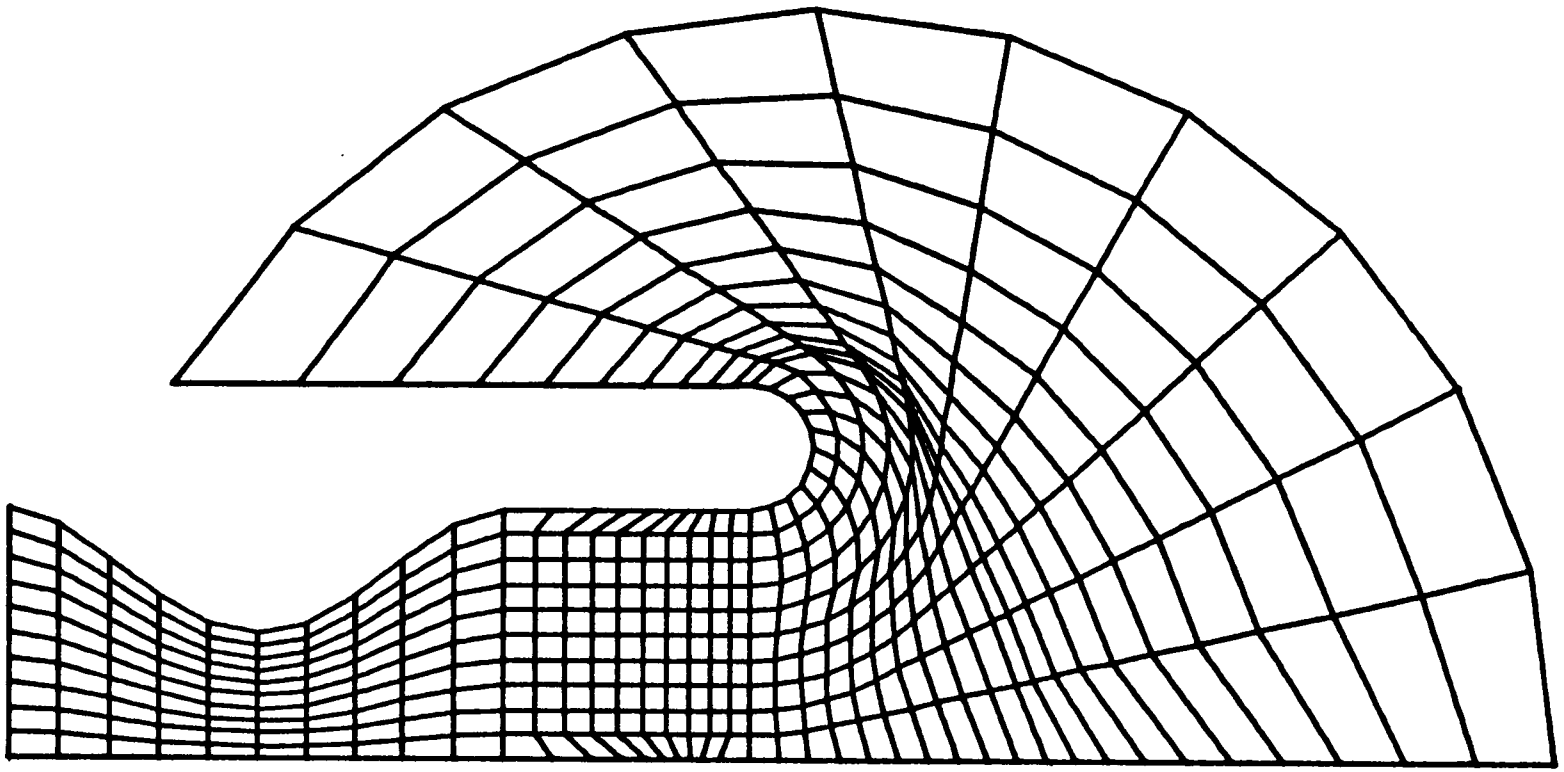


Figure 4. Variable area duct and far field: Initial condition. 11 grid points in the transverse direction.

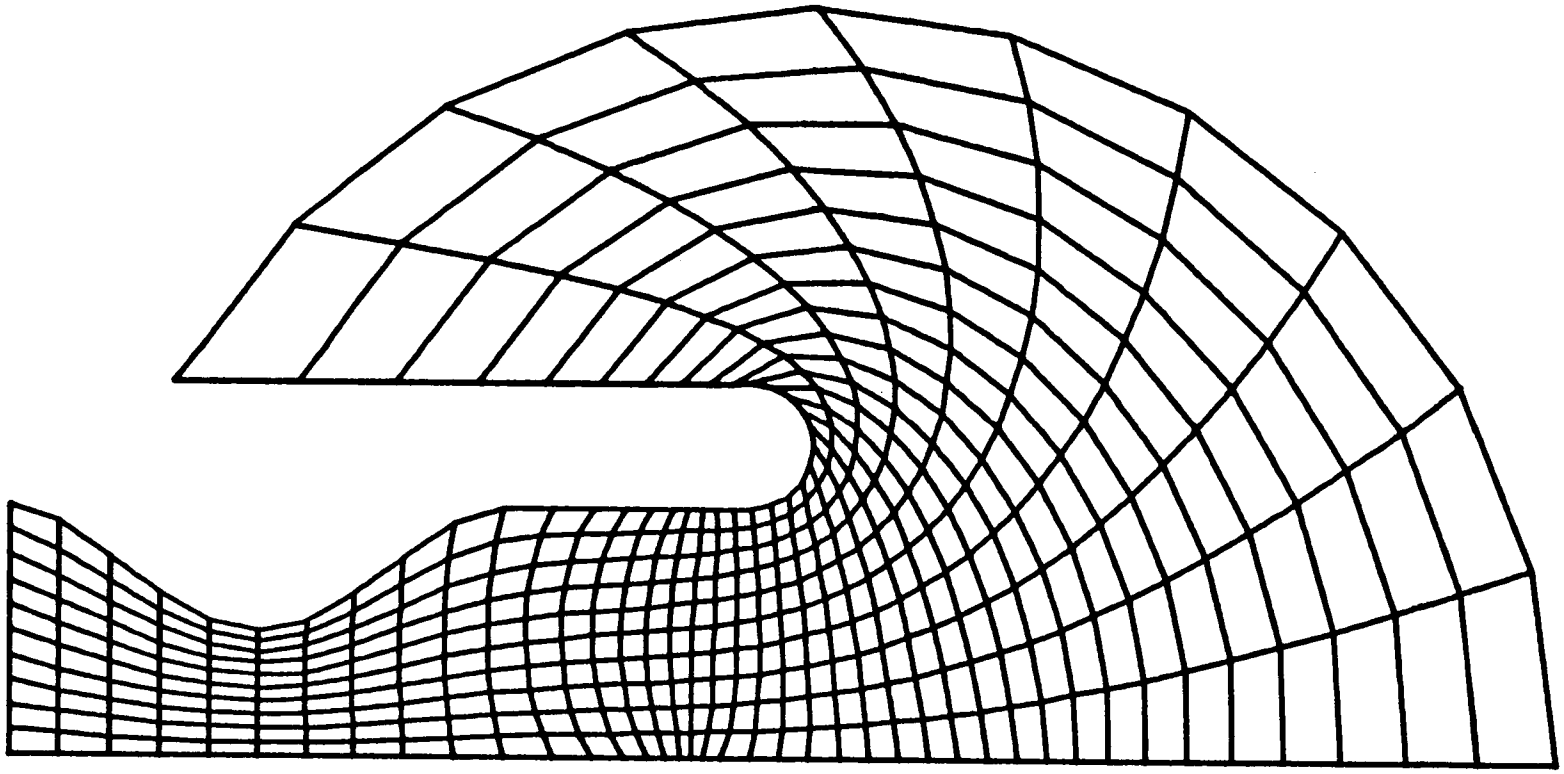


Figure 5. Variable area duct and far field: Final solution. 11 grid points in the transverse direction.

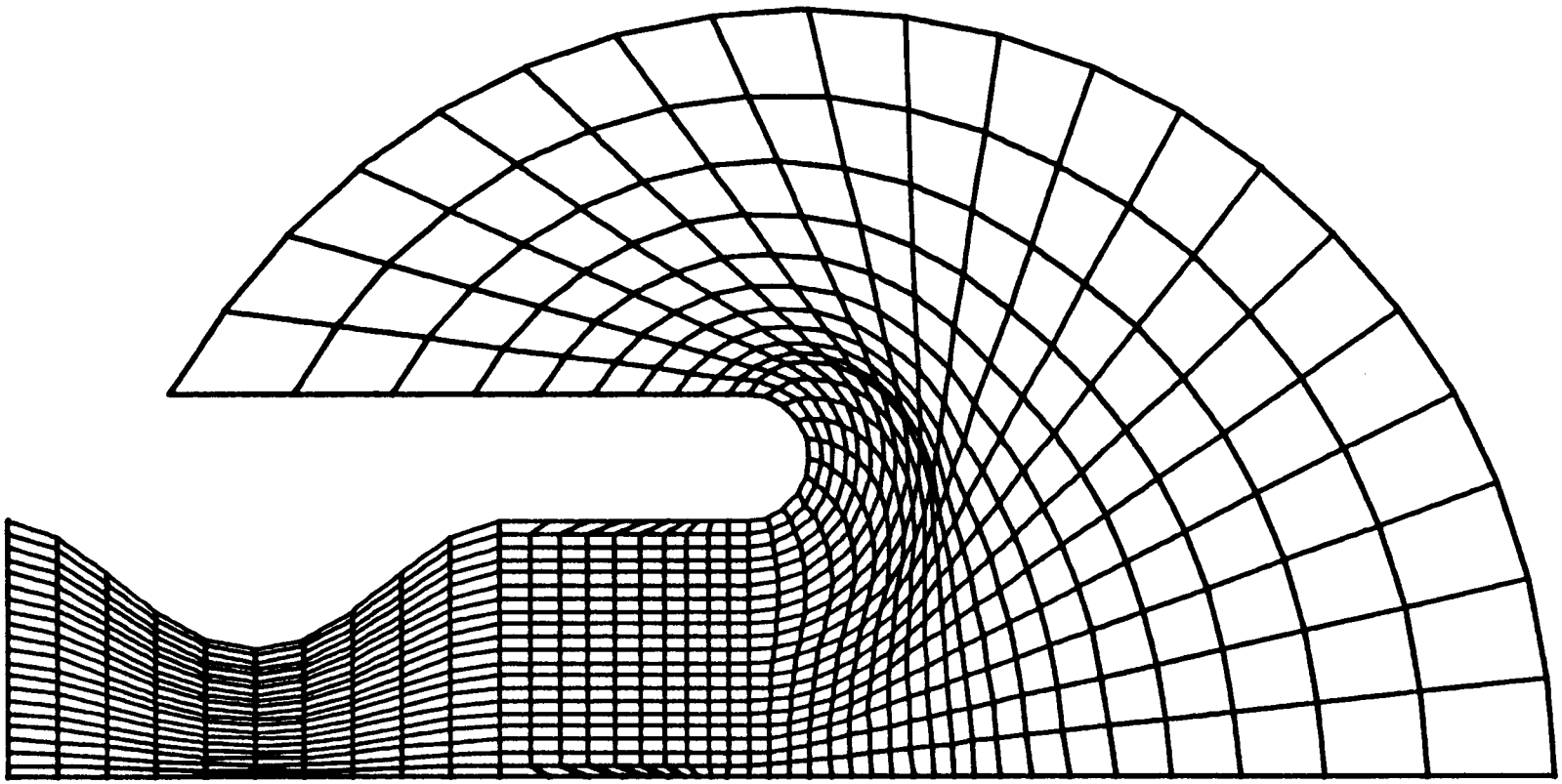


Figure 6. Variable area duct and far field: Initial condition. 21 grid points in the transverse direction.

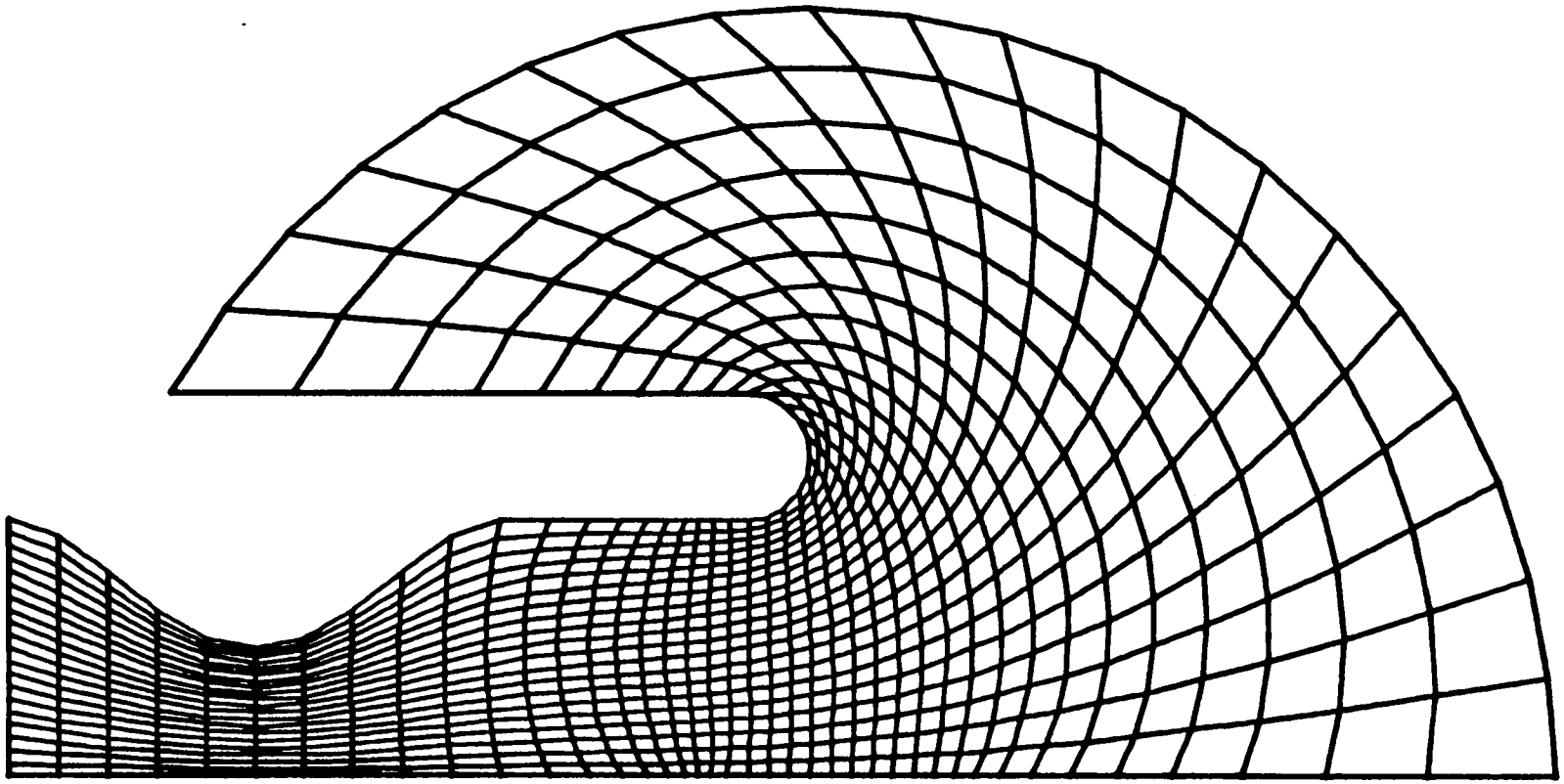


Figure 7. Variable area duct and far field: Final solution. 21 grid points in the transverse direction.

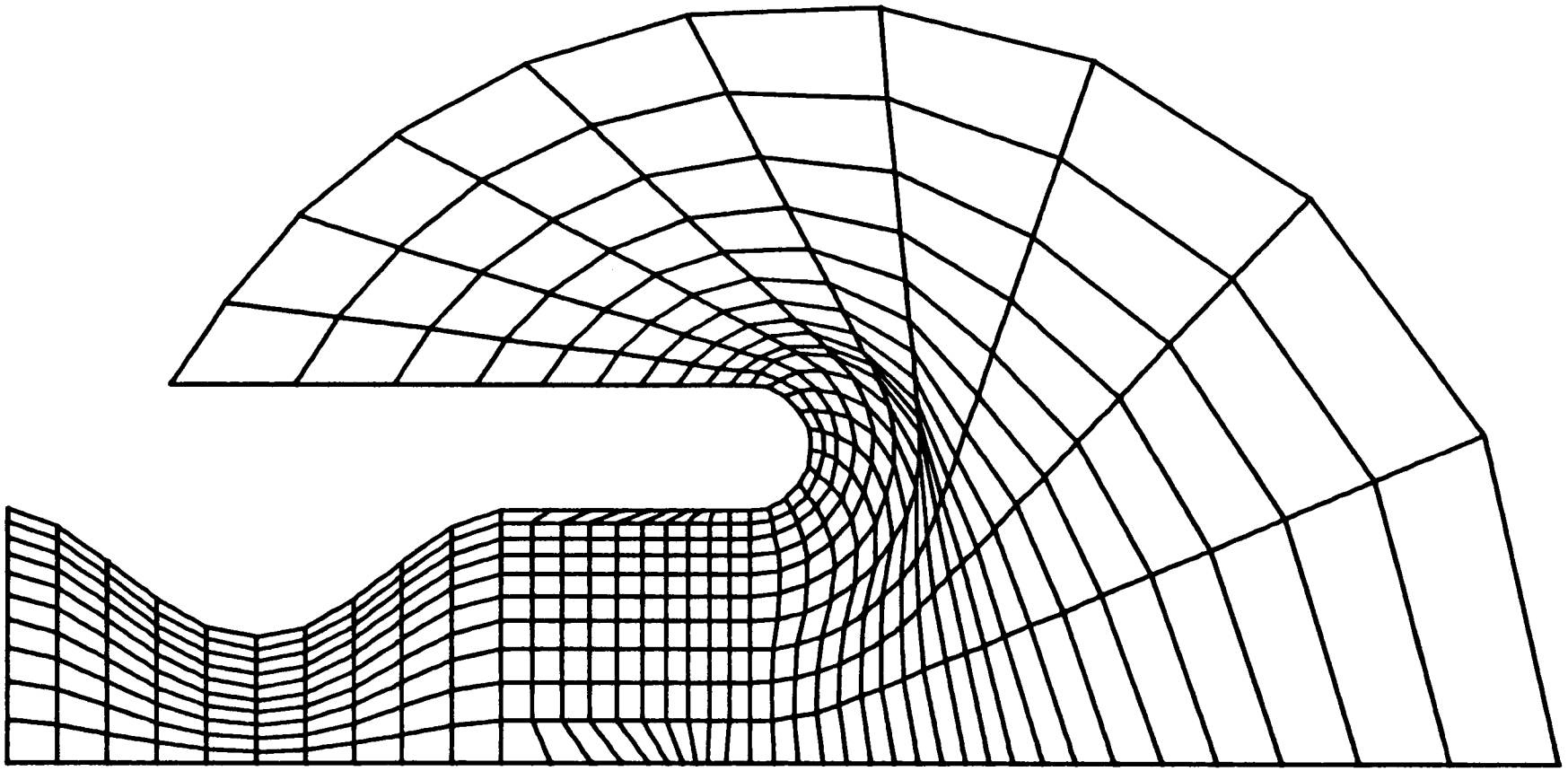


Figure 8. Variable area duct and far field: Initial condition.
Non-uniformly spaced grid points in the transverse direction.
Total height: 1.0; first grid spacing on top: 0.05.

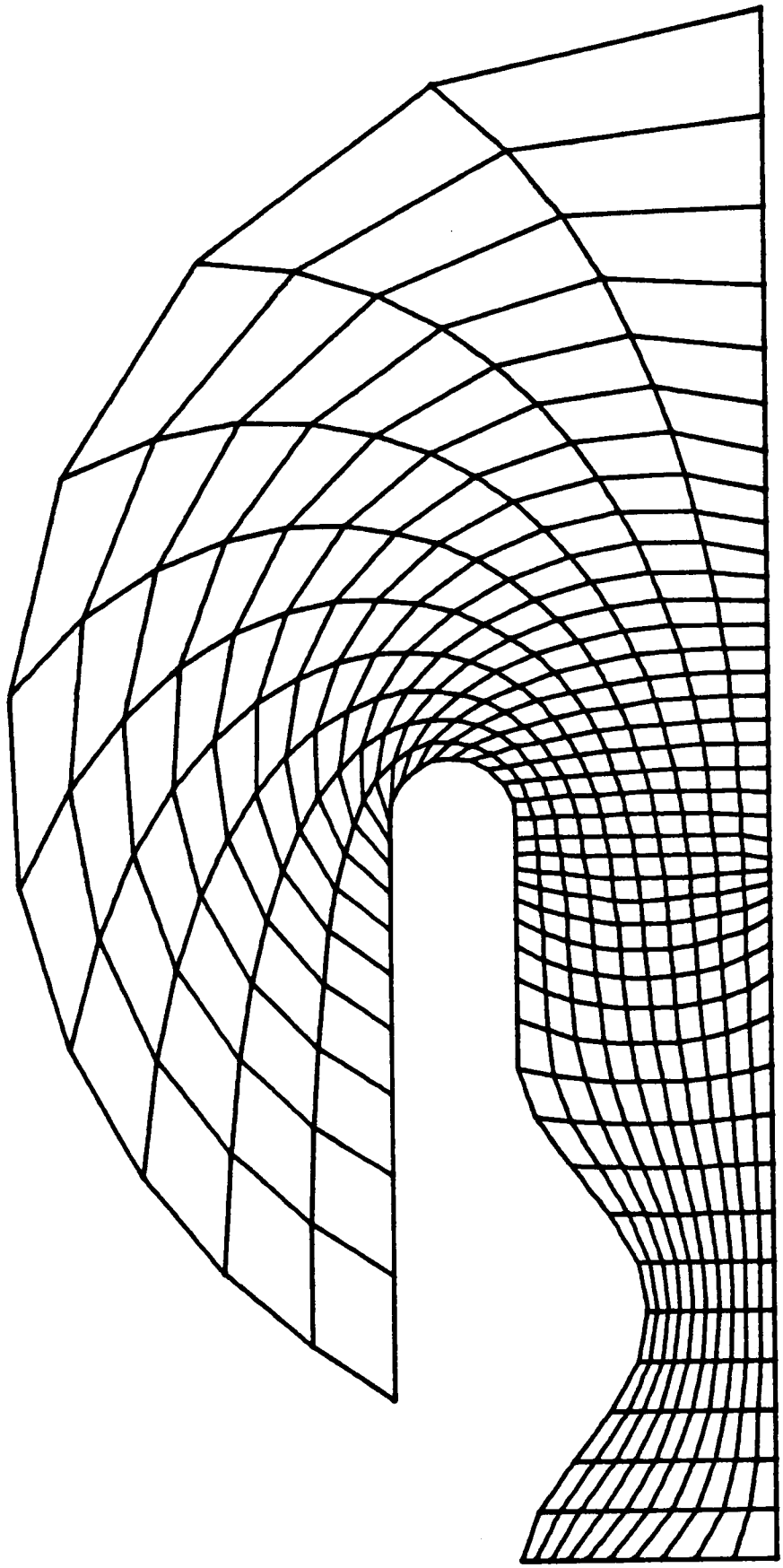


Figure 9. Final solution of the initial condition in Figure (8).

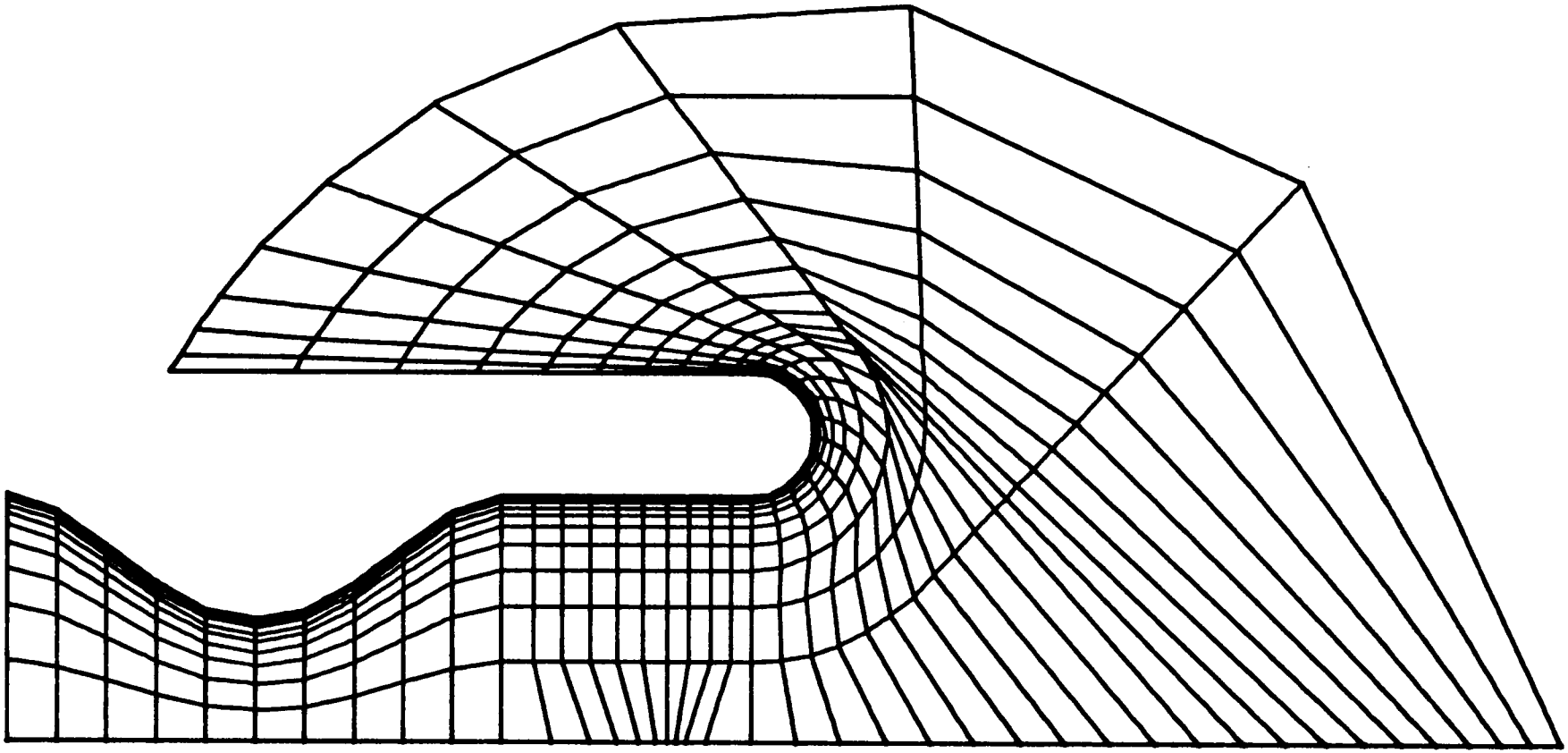


Figure 10. Variable area duct and far field: Initial condition. Non-uniformaly spaced grid distribution in the transverse direction. Total height: 1.0; upper grid spacing: 0.01.

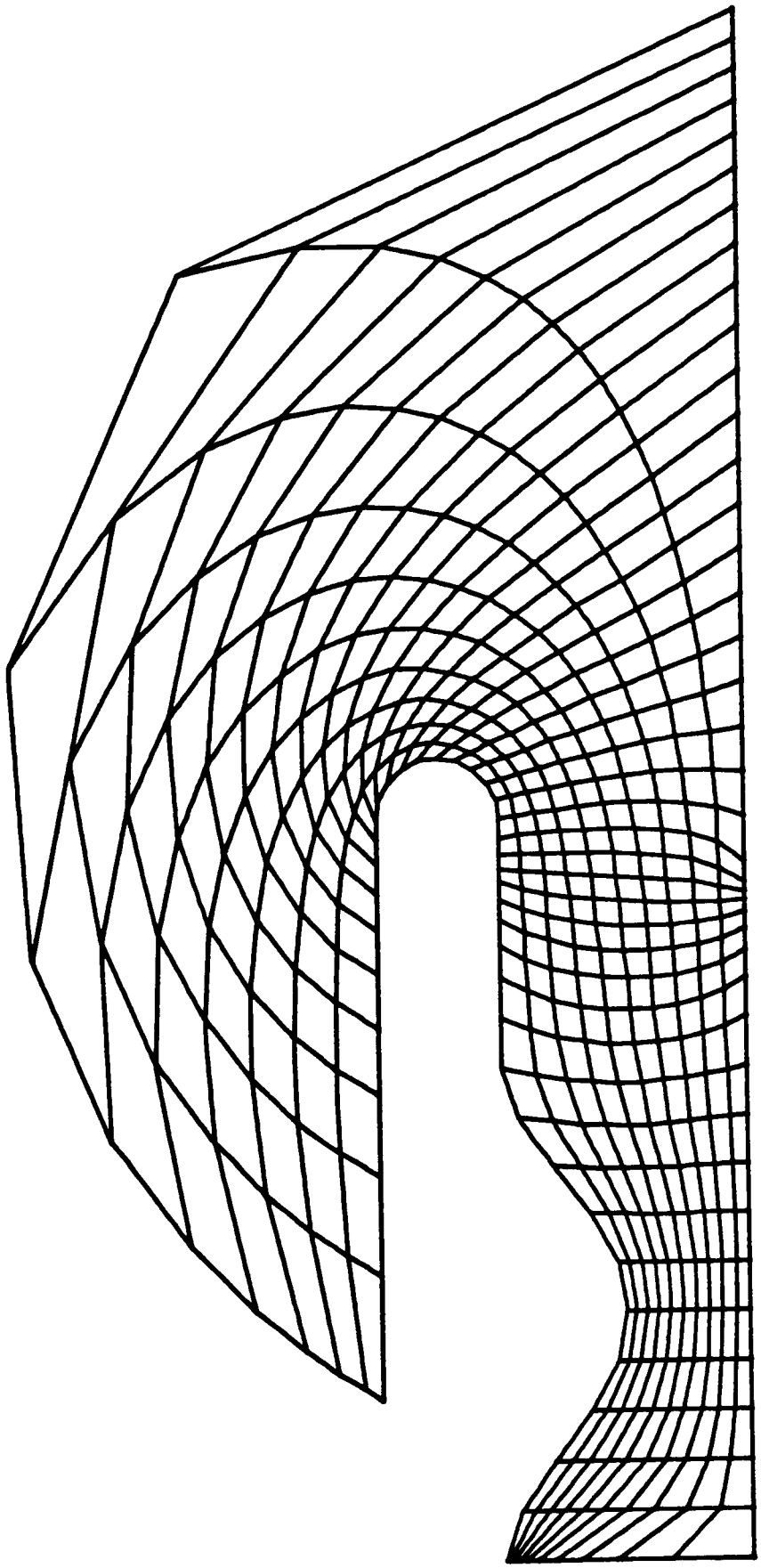


Figure 11. Final solution of the initial condition in Figure (10).

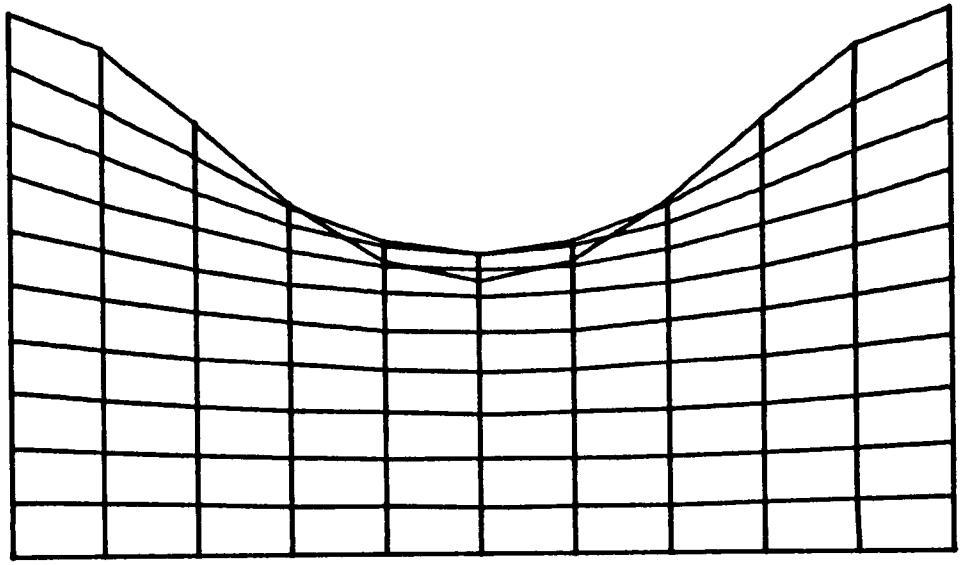


Figure 12. Alternate mapping solution using an accelerated relaxation procedure. Crossing of coordinate lines on top boundary.

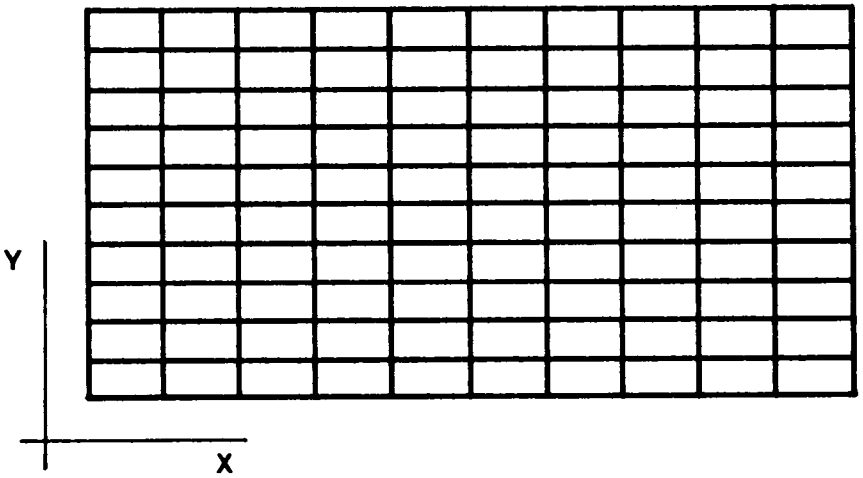


Figure 13. Initial condition and final solution of a rectangular duct. Hard wall condition on top and bottom walls.

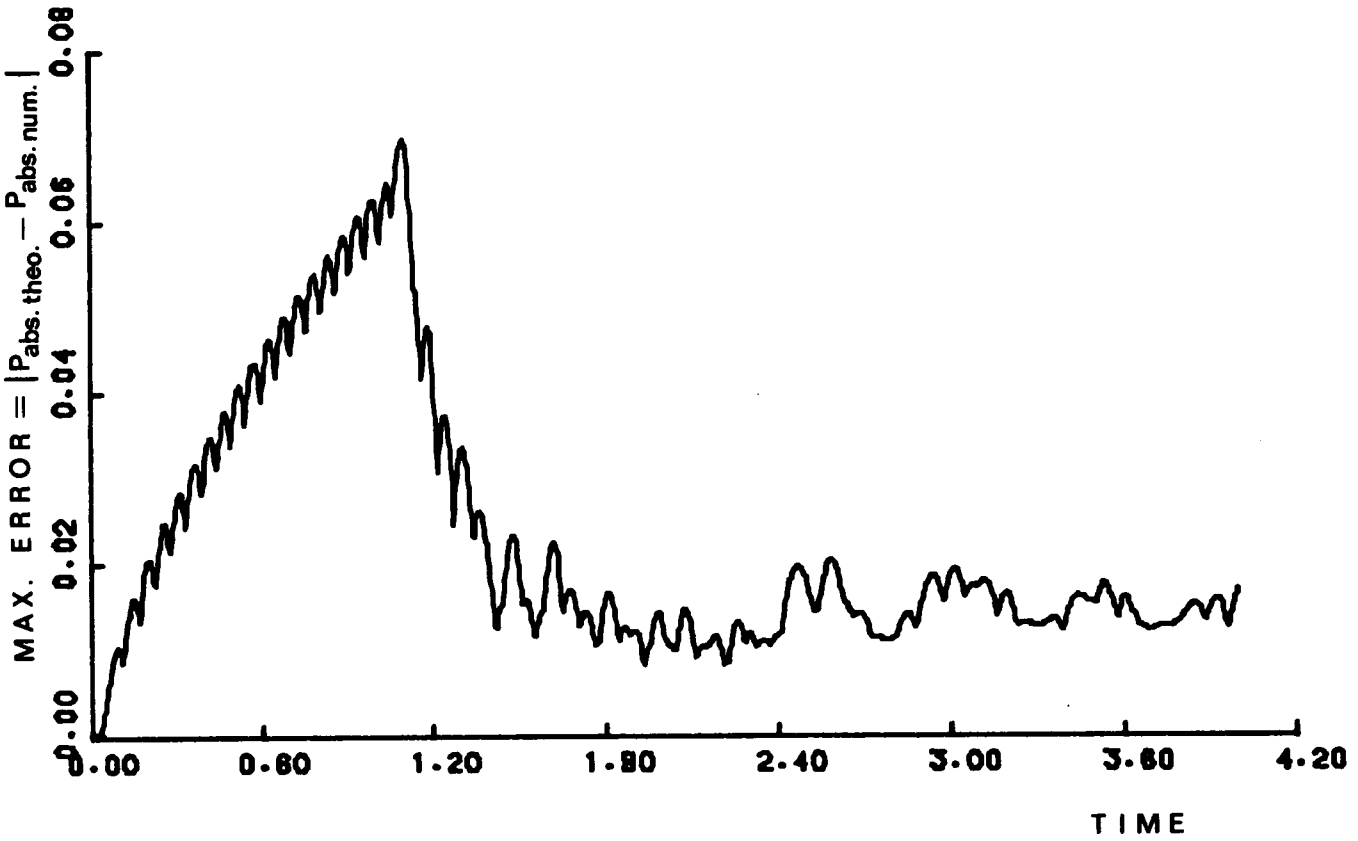


Figure 14. Absolute value of the maximum deviation from theoretical results versus time for the explicit solution of noise propagation in a uniform duct. $\Delta t = 0.004$; NX = 21.

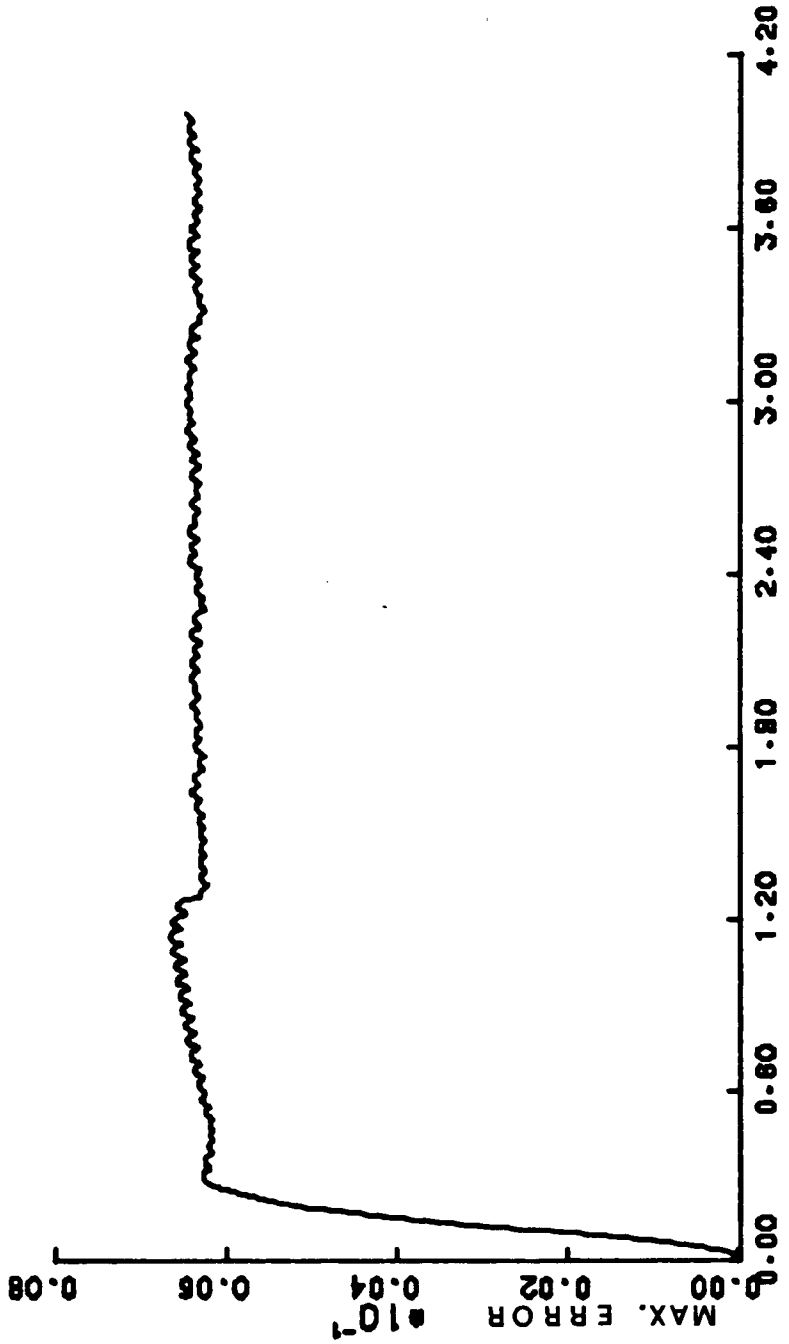


Figure 15. Absolute value of the maximum deviation from theoretical results versus time for the implicit solution of noise propagation in a uniform duct. $\Delta t = 0.004$; NX = 21.

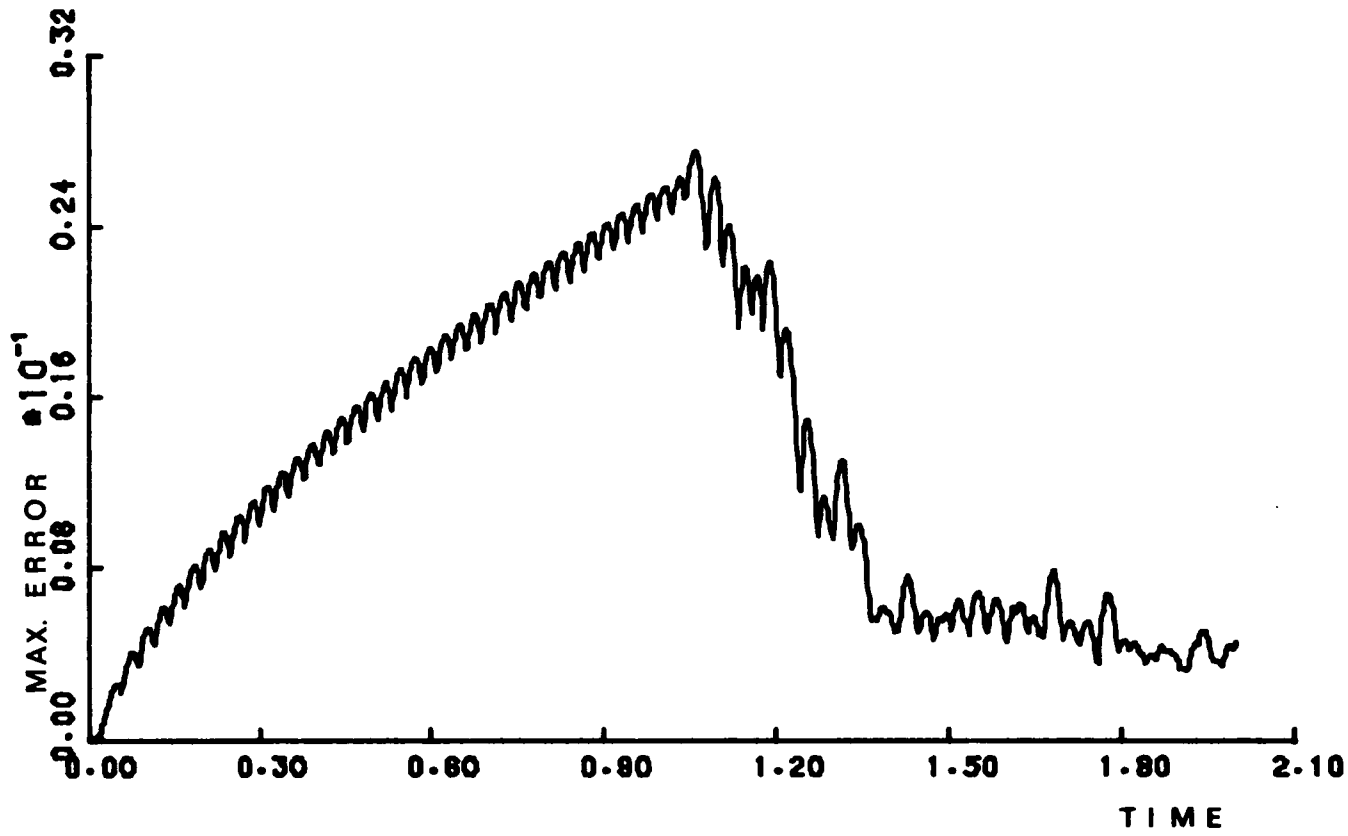


Figure 16. Absolute value of the maximum error in computed results versus time for the explicit solution of noise propagation in a uniform duct. Δt=0.002; NX=41.

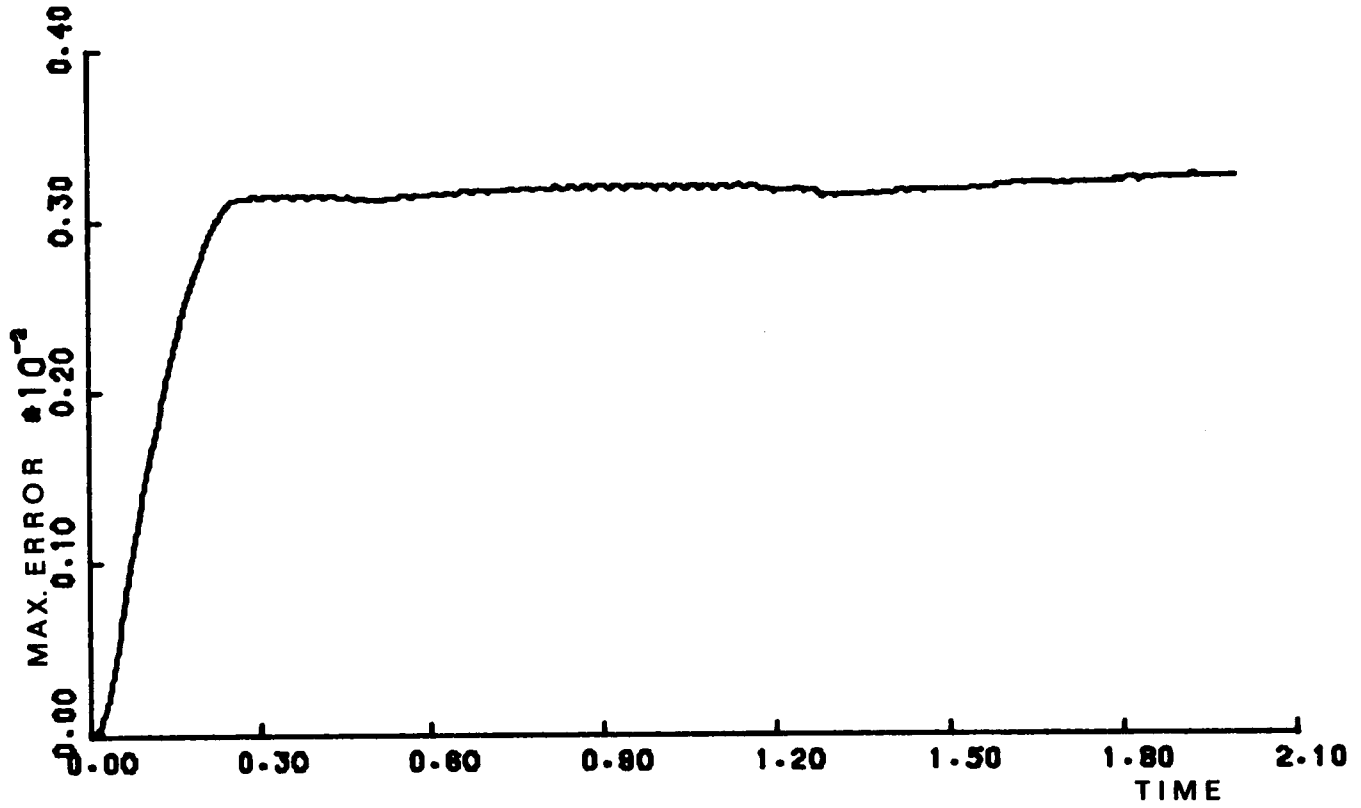


Figure 17. Absolute value of the maximum error in computed results versus time for the implicit solution of noise propagation in a uniform duct. $\Delta t=0.002$; NX=41.

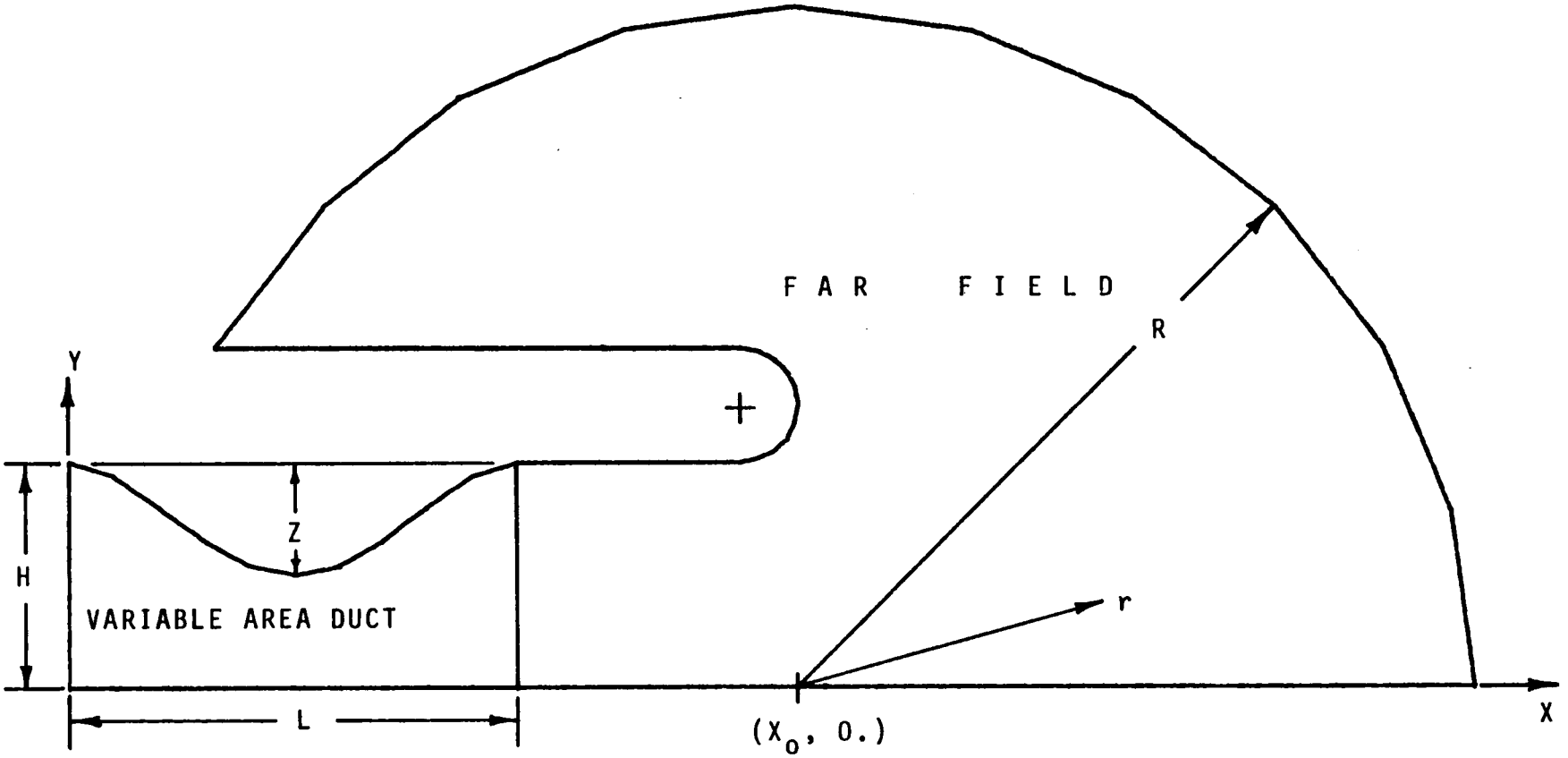


Figure 18. Variable area duct with far field: Variables' definitions and dimensions.

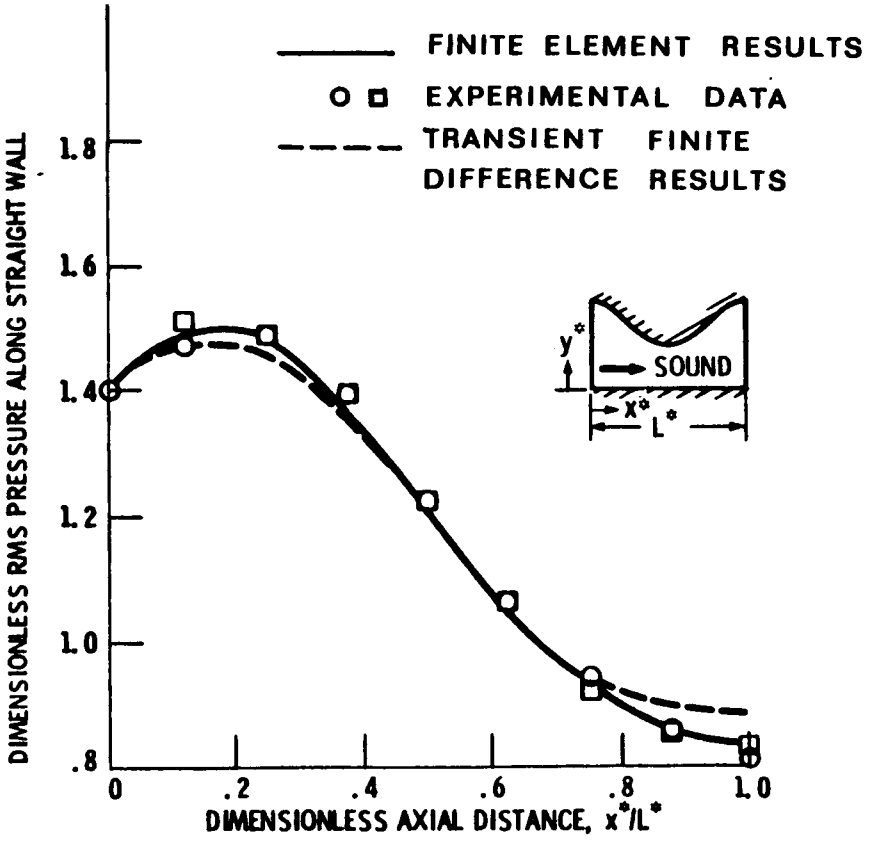


Figure 19. Numerical and experimental axial pressure profiles for a variable area duct with a plane wave exit condition. ω=0.172.

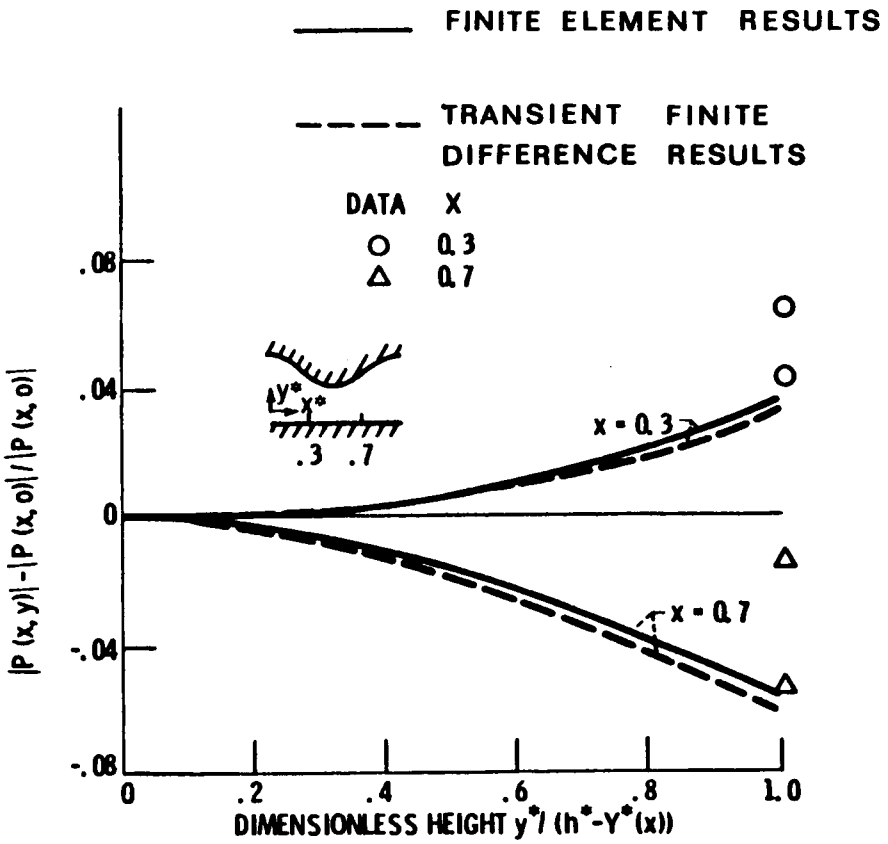


Figure 20. Numerical and experimental transverse pressure profiles for the same duct in Figure (19).

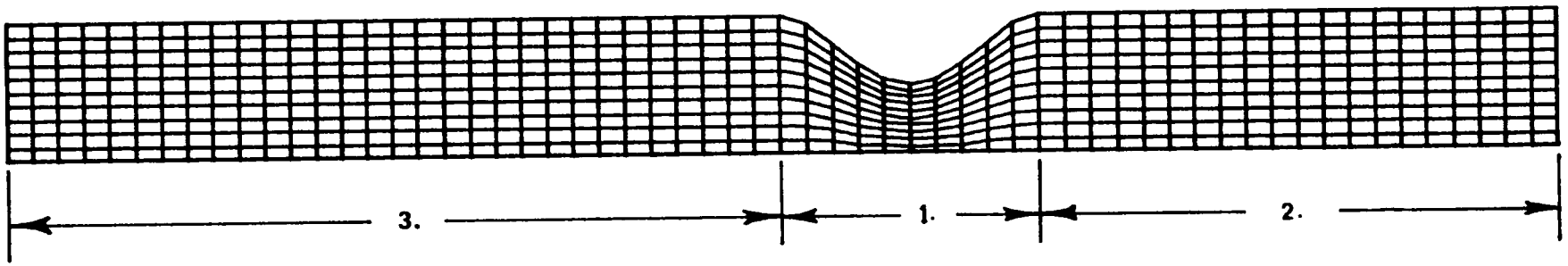


Figure 21. Variable area duct with inlet and exit straight sections: Initial condition.

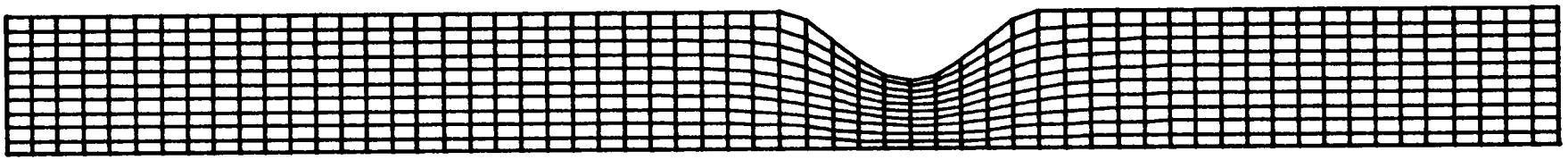


Figure 22. Variable area duct with inlet and exit straight sections: Final solution.

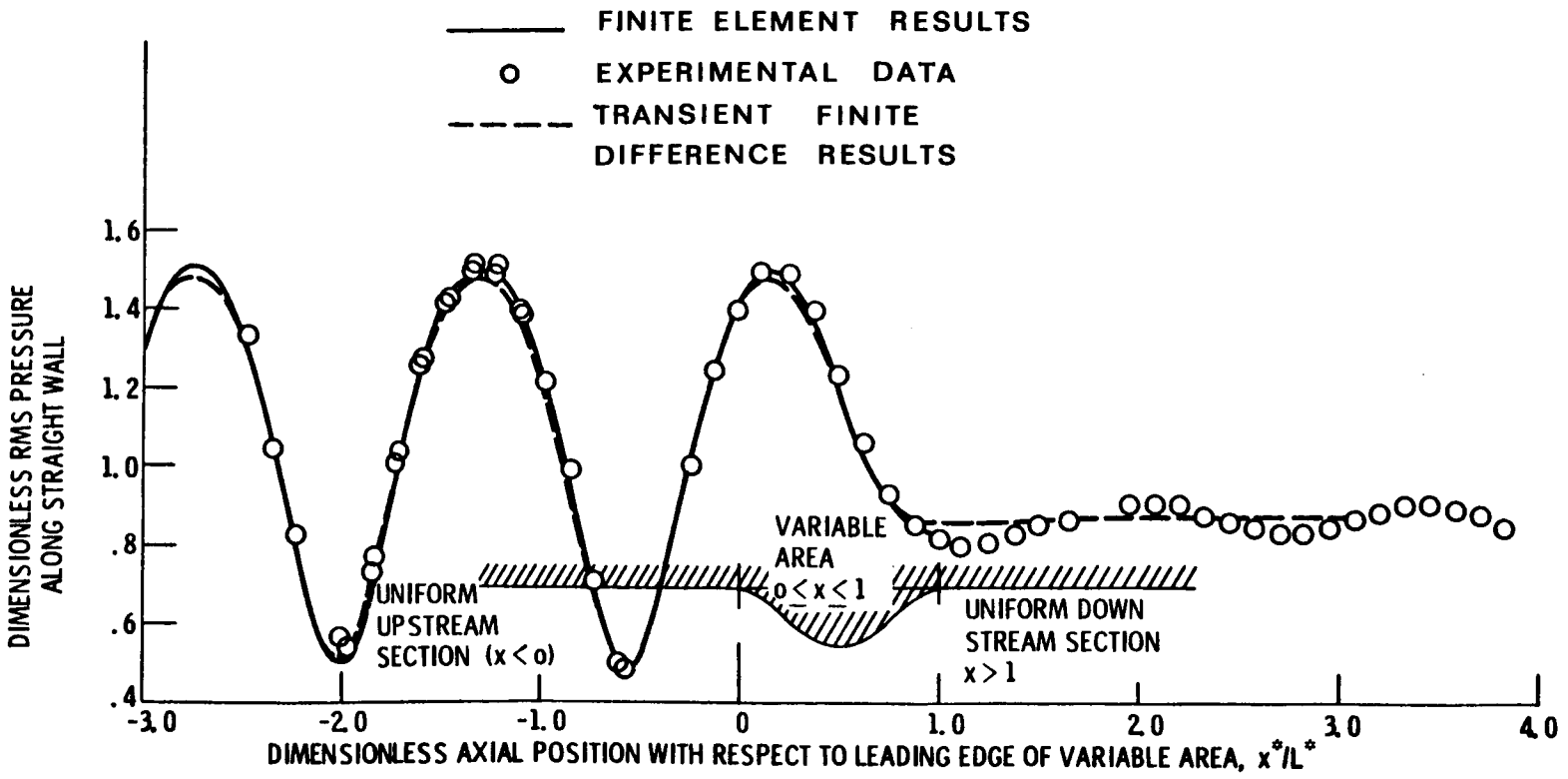


Figure 23. Numerical and experimental pressure profiles for the computational grid shown in Figure (22).

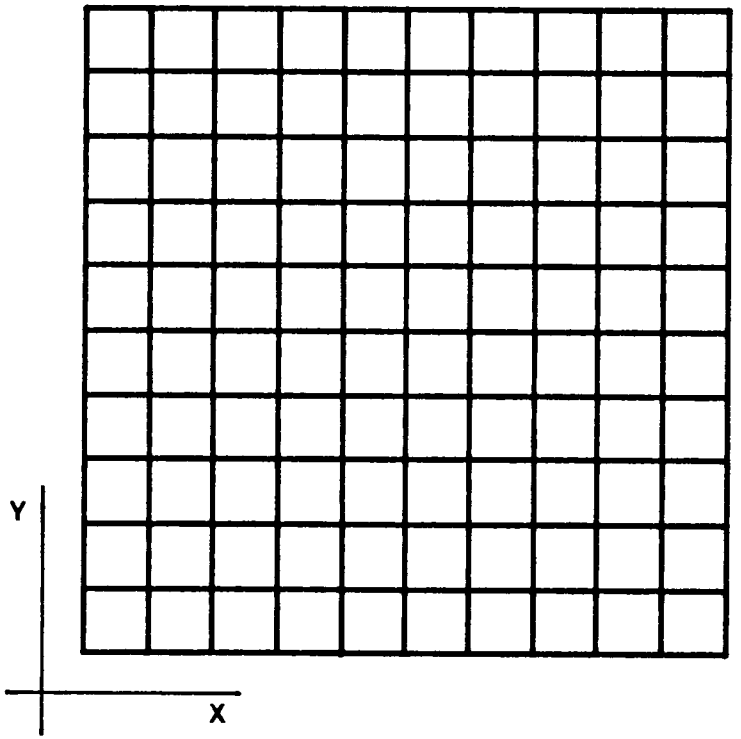


Figure 24. Square duct with one soft wall: Initial condition and final grid solution.

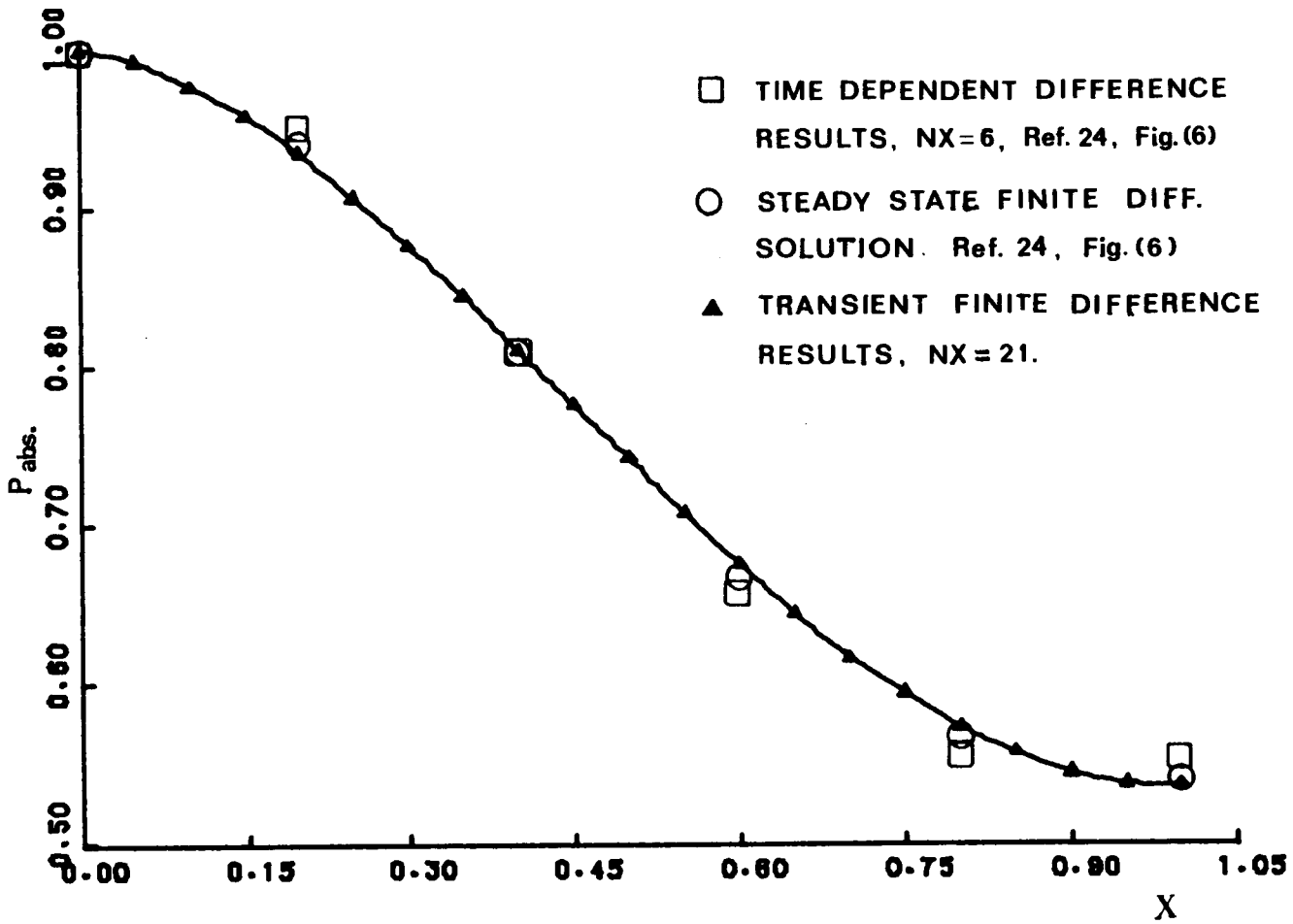


Figure 25. Pressure magnitude along the duct axis for incident plane wave sound propagation in a soft wall duct. η=0.3; ζ=0.16-i(0.34); L^*/H^*=1.0.

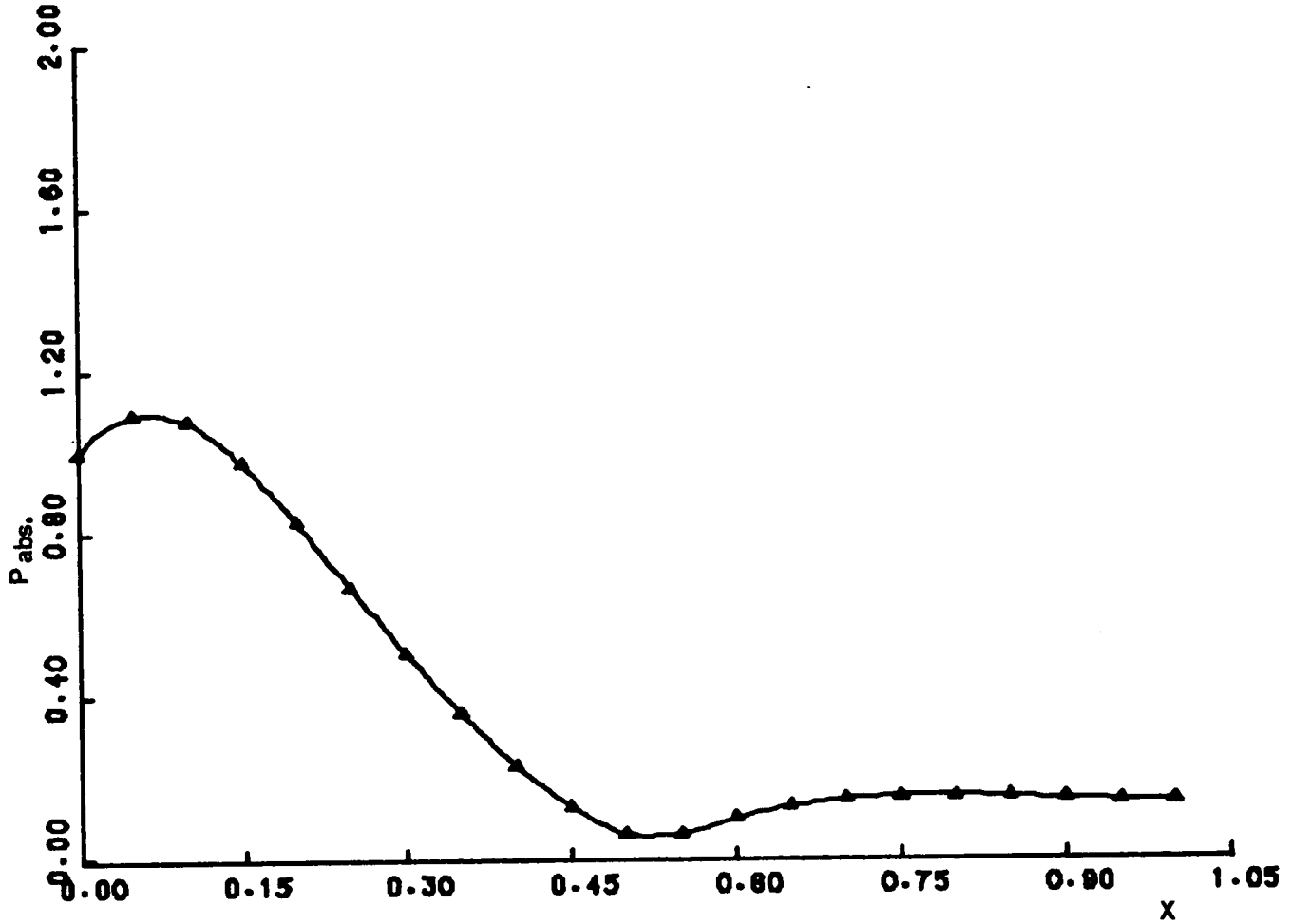


Figure 26. Pressure magnitude along flat section of variable area duct of Figure (1) after 3000 time steps of solution. $L^*/H^*=2.0$; $\eta=0.125$; $\zeta=0.085-i(0.16)$; $\Delta t=0.004$.

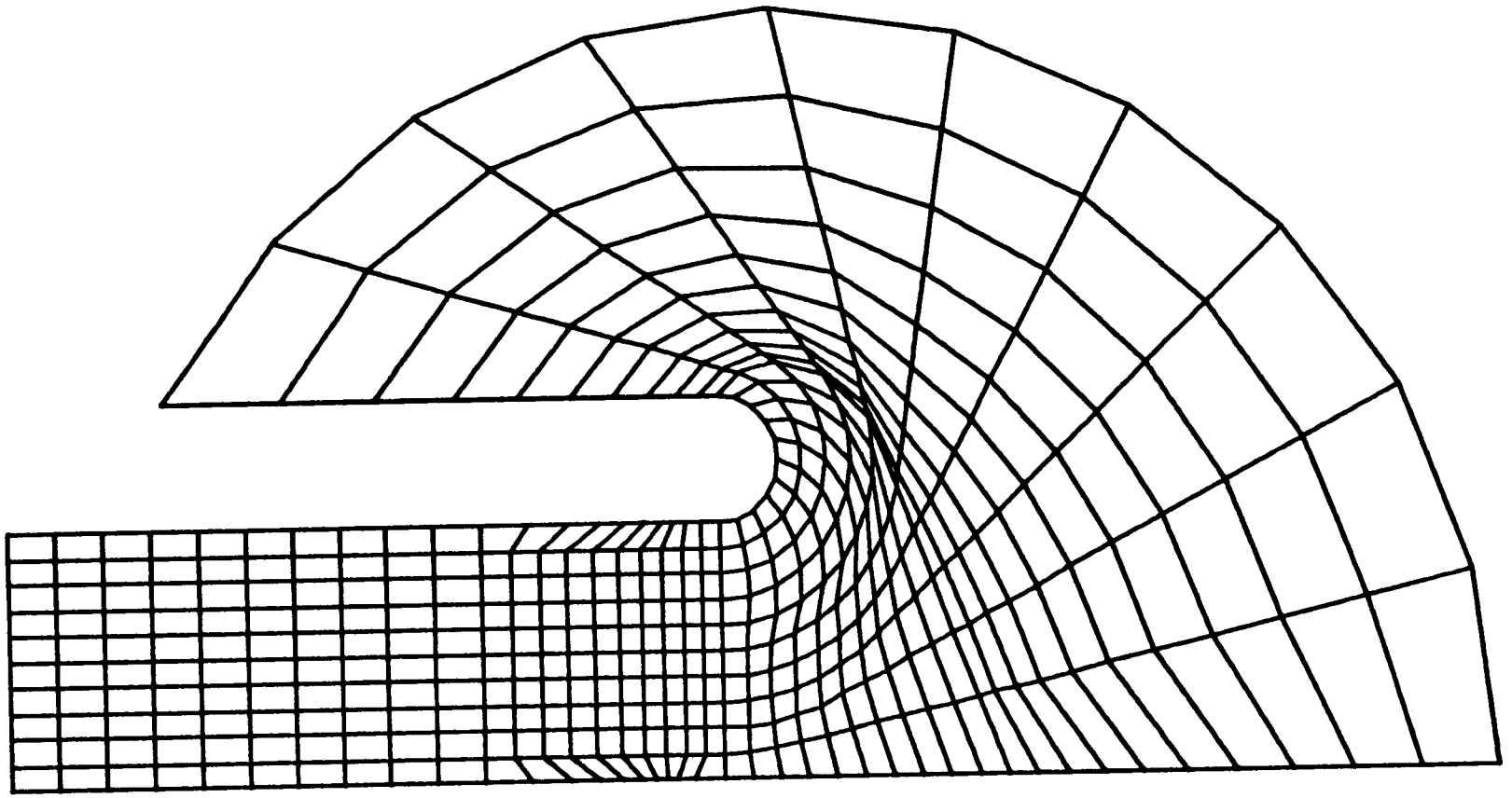


Figure 27. Uniform area duct and far field: Initial condition.

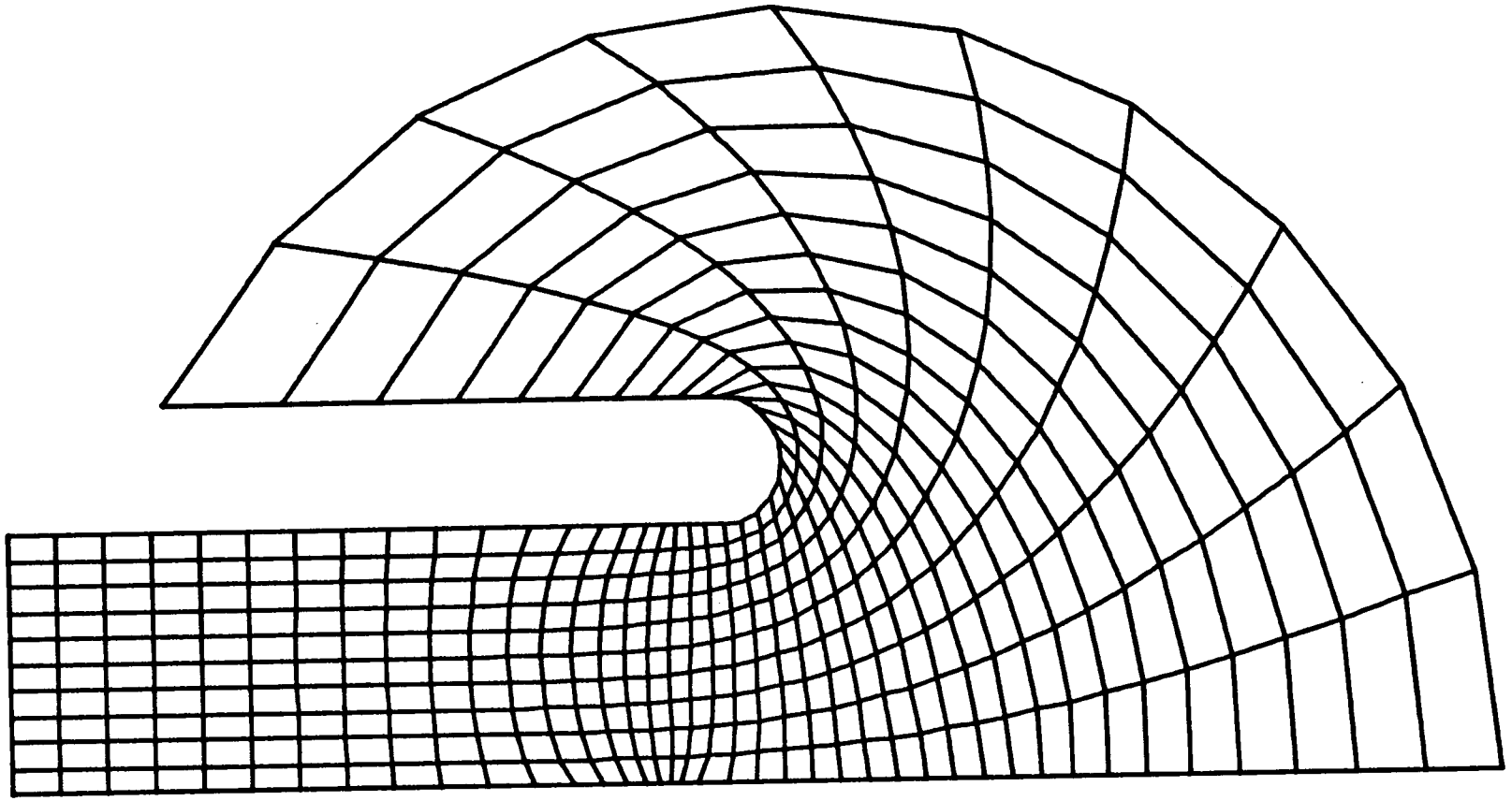


Figure 28. Uniform area duct and far field: Final grid solution.

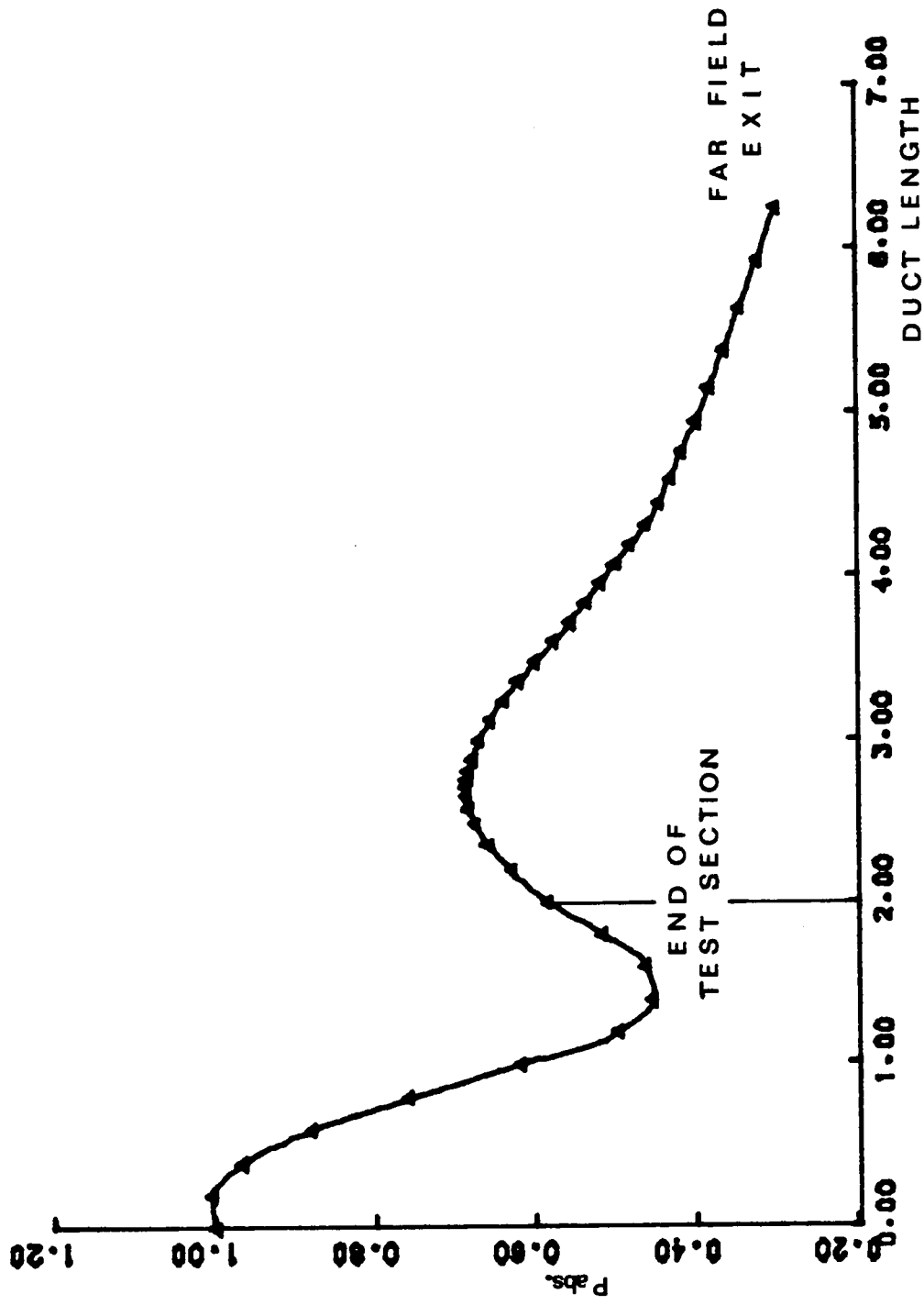


Figure 29. Pressure magnitude along bottom wall of the duct in Figure (28). Radius of the far field is three times the duct height.

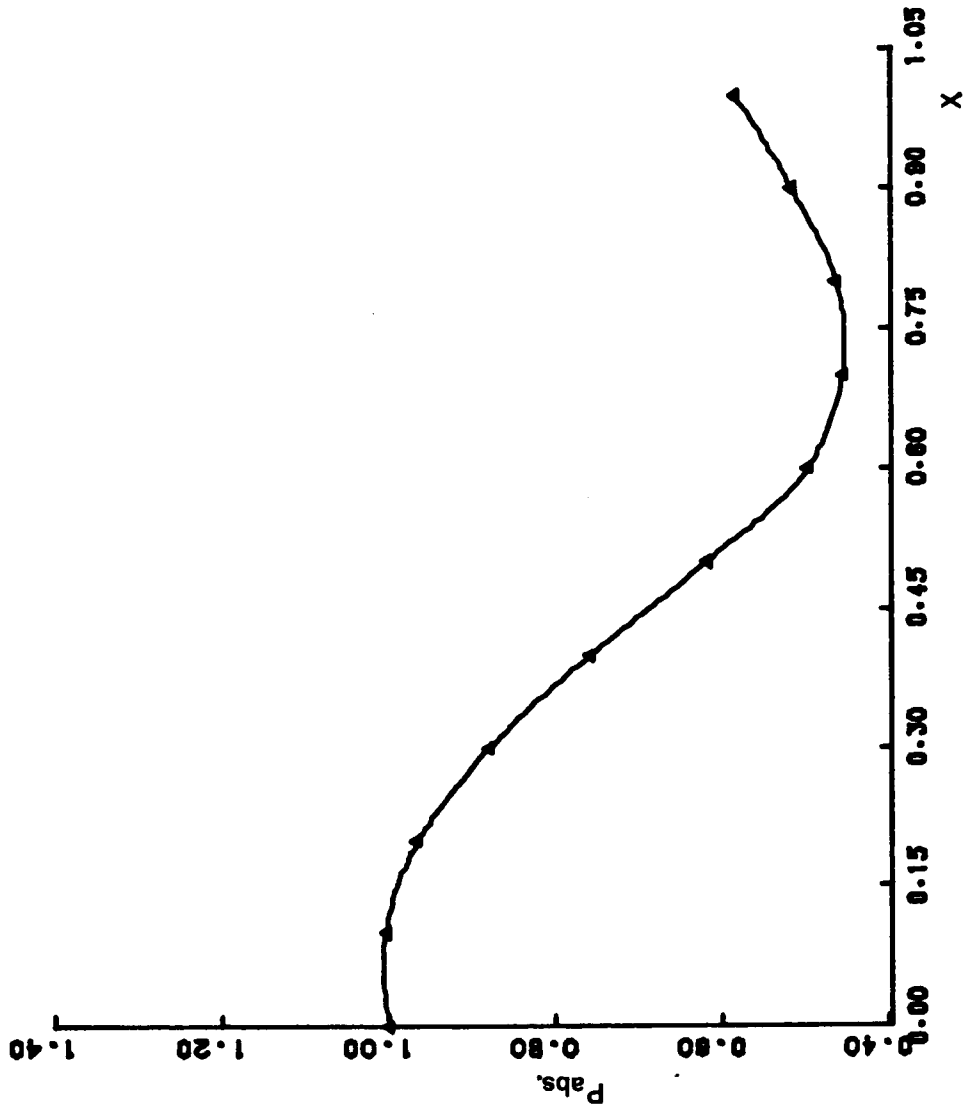


Figure 30. Pressure magnitude along flat wall of variable area duct (Figure (5)). Radius of the far field is three times the duct height.

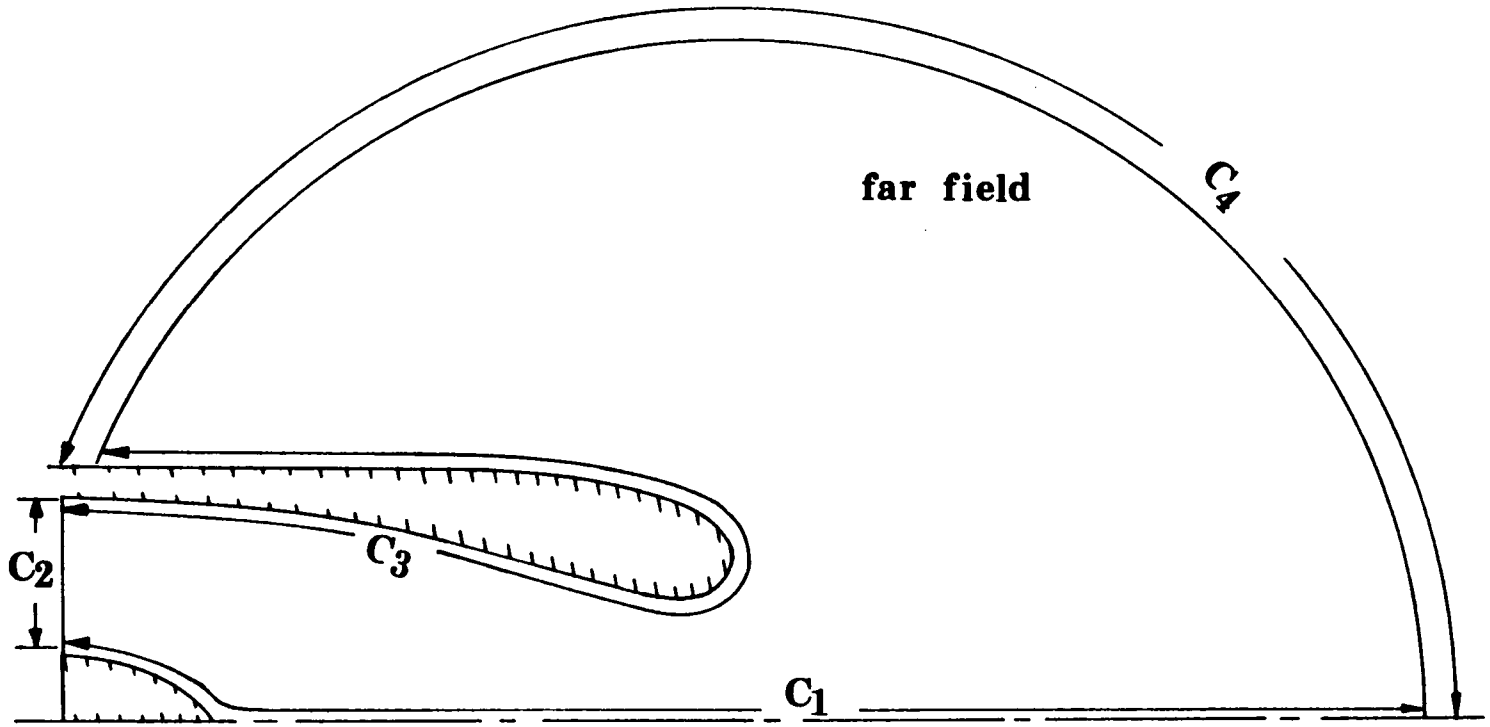


Figure 31. Jet engine configuration and far field: General shape.

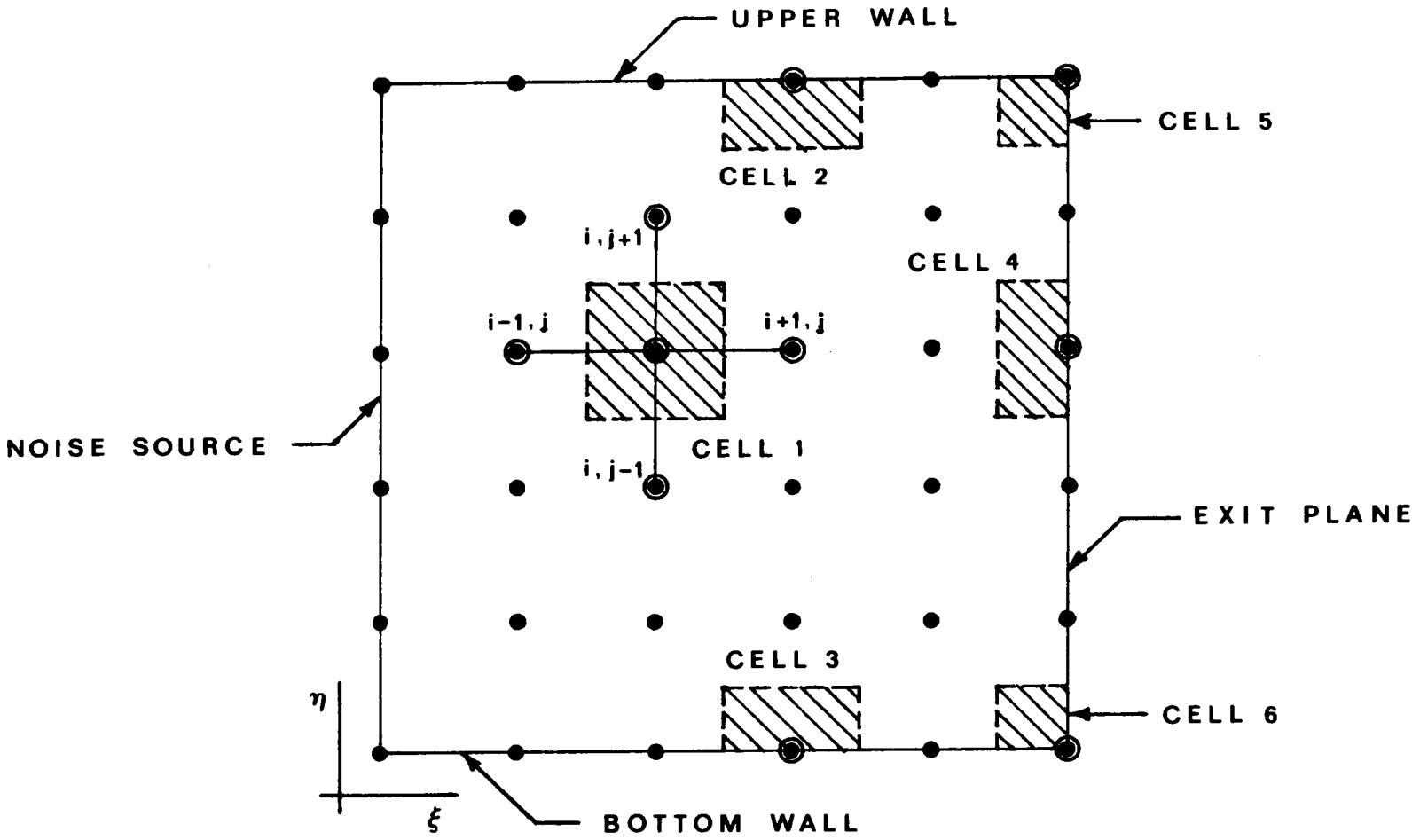


Figure 32. Computational square grid in (ξ-η) plane. Representative cells used in the finite difference formulation of the wave equation and its appropriate boundary conditions.

VITA

Peter Emile Raad was born in Beirut, Lebanon on July 22, 1959. He attended elementary school at the Collège De La Sagesse, Ashrafieh, Lebanon. He then attended the International College, Beirut, Lebanon, through the 13th grade and graduated from the class of Mathématiques Elémentaires in July 7, 1977. Later that summer, he took and passed the Lebanese Baccalaureat Part II (Section Mathématique) governmental exam.

In September 1977, he entered the University of Tennessee, Knoxville. In August 25, 1980, he received a Bachelor of Science degree in Mechanical Engineering.

In the fall of 1980 he accepted a research assistantship at the University of Tennessee, Knoxville, and began study toward a Master's degree. This degree was awarded in December 1981.

The author is a member of Tau Beta Pi, American Institute of Aeronautics and Astronautics and the American Society of Mechanical Engineers.