NASA CR 159679

Department of Aerospace Engineering
and Applied Mechanics
University of Cincinnati

FLOW IN NONROTATING PASSAGES OF RADIAL INFLOW TURBINES

BY

E. BASKHARONE, A. HAMED AND W. TABAKOFF

Supported by:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Lewis Research Center

NASA Grant No. NSG 3066

September 1979

NASA CR-159679

FLOW IN NONROTATING PASSAGES OF RADIAL INFLOW TURBINES

by

E. Baskharone, A. Hamed and W. Tabakoff

September 1979

Supported by:

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION

Lewis Research Center

NASA Grant No. NSG 3066

1. Report No. NASA CR-159679	2. Government Accession No.	3. Recipient's Catalog No.
4. Title and Subtitle FLOW IN NONROTATING PASSAGES OF RADIAL INFLOW TURBINES		5. Report Date September 1979 6. Performing Organization Code
7. Author(s) E. Baskharone, A. Hamed and W. Tabakoff		8. Performing Organization Report No. 10. Work Unit No.
9. Performing Organization Name and Address Dept. of Aerospace Engineering & Applied Mechanics, University of Cincinnati, Cincinnati, Ohio 45221		11. Contract or Grant No. NASA NSG 3066 13. Type of Report and Period Covered: Contractor Report
12. Sponsoring Agency Name and Address National Aeronautics and Space Administration Washington, D.C. 20546		14. Sponsoring Agency Code

15. Supplementary Notes
Project Manager, Dr. Arthur J. Glassman, Turbine Branch, NASA Lewis Research Center, Cleveland, Ohio 44135.

16. Abstract

The analysis of irrotational incompressible flow field in the stator unit of a radial inflow turbine is presented. The solution in the combined scroll-nozzle assembly is complicated by the domain geometry and by its multi-connectivity. This model is necessary however in order to provide a better understanding of the mutual interaction effects of these two components on the flow field. The finite element method is used in the solution which is limited to the two dimensional case. A substructuring technique is adopted in the computational procedure and results in considerable savings in both computer time and core storage requirements. The results are presented for the flow velocity magnitude and direction in the scroll and through the various nozzles, for two nozzle blade geometries. In addition, the mass flow rates in the different nozzles are computed and their deviations from the mean value determined.

17. Key Words (Suggested by Author(s))		18. Distribution Statement	
Radial Turbine Scroll		Unclassified - unlimited	
19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	99	

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

N79-32212#

TABLE OF CONTENTS

	Page
SUMMARY	1
INTRODUCTION	1
ANALYSIS	2
Problem Formulation	3
Governing Equations	4
The Boundary Conditions	4
The Finite Element Method	5
The Substructuring Technique	9
Illustrative Example	9
Application of the Technique to the Flow Field in the Stator Unit	11
RESULTS AND DISCUSSION	14
CONCLUSIONS	17
REFERENCES	18
NOMENCLATURE	21
TABLE AND FIGURES	22
APPENDIX	31

LIST OF TABLES AND FIGURES

Table		Page
1	Effect of Using the Substructuring Technique on Computer Time and Storage	22

Figure		
1	Stator Unite of a Radial Turbine	23
2	The Finite Element Discretization Model	24
3	Solution Domain of the Example Problem	25
4	Nondimensional Velocity Potential Distribution for Cambered Nozzle Blades	26
5	Nondimensional Velocity Distribution for Cambered Nozzle Blades	27
6	Nondimensional Velocity Distribution for Symmetric Nozzle Blades	28
7	Nondimensional Velocity Distribution for Symmetric Nozzle Blades	29
8	Nondimensional Mass Flow Distribution Through the Different Nozzle Channels	30

SUMMARY

The analysis of irrotational incompressible flow field in the stator unit of a radial inflow turbine is presented. The solution in the combined scroll-nozzle assembly is complicated by the domain geometry and by its multi-connectivity. This model is necessary however in order to provide a better understanding of the mutual interaction effects of these two components on the flow field. The finite element method is used in the solution which is limited to the two dimensional case. A substructuring technique is adopted in the compuational procedure and results in considerable savings in both computer time and core storage requirements. The results are presented for the flow velocity magnitude and direction in the scroll and through the various nozzles, for two nozzle blade geometries. In addition, the mass flow rates in the different nozzles are computed and their deviations from the mean value determined.

INTRODUCTION

In many industrial applications where low power output is required, radial gas turbines can offer many advantages over the axial flow turbines. The large clearances at the rotor tip, compared to the blade size, have a smaller effect on their performance, and they are also simpler to fabricate. A survey of the literature reveals that in recent years, several studies [1-5] have been concerned with improving the performance of small radial inflow turbines. Most of the investigations of radial inflow turbine performance, have been based on simplified one dimensional flow analyses in the scroll, and empirical relations for loss prediction [6-8]. Some studies [9] assumed free vortex flow in the scroll and vaneless nozzle, to evaluate the radial turbine performance parameters. The authors have obtained a solution to the three dimensional inviscid flow field in a radial inflow turbine scroll using the finite element method [10]. To our knowledge, this is the only three dimensional flow solution in this turbine component at present. Because of the complexity of this flow field, very few experimental measurements of the flow in the scroll, such as those reported in references [11] and [12] can be found in the literature. Reference [13] reports a detailed experimental and analytical study of the losses in both the vaned and vaneless regions of a radial inflow turbine nozzle. It was found in that study that in addition to the boundary layer parameters, the losses are influenced by the flow angle at the vaneless nozzle exit and the pressure distribution over the nozzle blades. These flow parameters can be determined from a nonviscous flow analysis in the turbine scroll nozzle assembly such as the present one.

It is important to have a thorough understanding of the flow behavior in the different turbine components for further performance improvements. For that purpose, and in order to account for the mutual effects of the scroll and nozzle vane geometry on the flow behavior, a detailed study of the flow in a radial inflow turbine scroll nozzle assembly is presented here. Through this analysis, numerical results were computed to show the effect of the nozzle vane shapes and setting on the magnitude and direction of the flow velocity at exit from the nozzle. The actual mass flow between the different vaned nozzle channels was also determined to check its variation from the uniform ideal distribution.

ANALYSIS

The flow in the turbine scroll and nozzles is actually three dimensional, as confirmed by the experimental flow measurements at inlet to and exit from the vaned nozzle region in reference [11]. Even if the viscous effects are not taken into consideration, the continuous mass discharge to the turbine nozzles, will result in a three dimensional flow field in the scroll [10]. In a different but parallel investigation [14], the three dimensional scroll flow was analyzed to determine the resulting velocity components in the scroll cross sectional planes, which are associated with various through flow velocity profiles. If the mutual effects of both scroll and nozzle blade geometries on the flow are to be investigated, it is necessary to consider the flow field in the whole scroll nozzle assembly, which is shown schematically in Fig. 1. Because of the difficulties that would be involved in satisfying the boundary conditions in such multiply connected domain, the analytical solution of such flow field would not have been possible before the development of the finite element methods in the field of solid mechanics and its later applications to flow problems. Therefore, it is natural, that careful problem formulation is necessary for the effective handling of the complicated boundary conditions in the above described flow field.

Problem Formulation

The potential function was chosen as the primary dependent variable in the following formulation, since it is more suitable than the stream function, for handling the boundary conditions involved in the present problem. The difficulty in using the stream function as the primary dependent variable, in flow fields with several immersed bodies, lies in not knowing the mass flow between every two of them a priori. Some investigators [15] suggested the superposition of several stream function solutions to satisfy the tangency condition at the different boundaries, in the finite element solutions of such problems. That approach is not suitable for our problem since its complexity increases with the increased number of blades. In selecting the appropriate formulation to be used in our problem, careful consideration should be given to handling the boundary conditions effectively and with no extra complications. In the potential function formulation, natural type boundary conditions, i.e. zero normal potential gradients, are specified at the boundaries of these bodies, which greatly simplifies the solution procedure. It has also been found in flow problems that solutions obtained using the finite element method, with straight sided elements approximating generally curved boundaries, are more accurate when Neumann rather than Dirichlet type boundary conditions are involved [16].

Governing Equations

The irrotational flow field can be described by the velocity potential function ϕ, with the velocity vector $\bar{V}$, expressed in terms of ϕ as follows:

$$\bar{V} = \nabla\phi \qquad (1)$$

The equation of conservation of mass, in terms of the velocity potential, for steady state is given by:

$$\nabla\cdot(\rho\nabla\phi) = 0 \qquad (2)$$

In this case, the isentropic relations can be used to determine the compressible flow density ρ.

The Boundary Conditions

The flow velocities normal to the flow field boundaries are zero at the solid surfaces and are related to the mass flow across the nonsolid domain boundaries. It is well known that the Joukowski condition must be satisfied for potential flows around airfoils. Blade-to-blade turbomachine flow computations, using both finite element and finite difference methods have reportedly been using the stream function formulation [17, 18]. In these cases, the far downstream flow angle is modified during the iterative solution, until the flow velocities at the airfoil trailing edge are equal on both suction and pressure sides. When the velocity potential formulation is used, the boundary conditions involve instead the velocity component in the direction normal to the exit station, which is solely related to the mass flow in the turbine. This situation is not peculiar however, since the potential function is actually multi-valued in the flow field, if any tangential forces are exerted by the blades. The lift or blade loading itself is related to the change of momentum in the direction of the cascade front.

Permiable boundary surfaces, which are separated by infinitesimal width, were therefore introduced in the vaneless space of the solution domain at the trailing edges of all the nozzle vanes as shown by the dotted lines in Fig. 1. The magnitudes of the jumps in the value of the velocity potential across these splitting surfaces were introduced as new variables in the solution. By guaranteeing the continuity in flow velocities across the corresponding boundaries of each two elements on the opposite sides, these variables were determined in the finite element solution.

Referring to Fig. 1, the boundary conditions can therefore be expressed in the following general form:

$$\frac{\partial \phi}{\partial n} = f \tag{3}$$

where $f = 0$ on the solid boundaries;

$f = V_1$ at inlet to the scroll;

$f = V_{n_2}$ at exit from the turbine nozzle;

and f is continuous but unknown across the splitting boundaries extending from the nozzle blades.

The Finite Element Method

The three basic steps involved in the finite element method are the domain discretization, the integral statement of a problem and the local approximation of the function in each element. Several approaches such as variational principles, weighted residuals, and least square methods which are described in references [19-22] can be followed to arrive at the integral

statement of the problem. Although a classical variational statement exists for the potential flow problem, the element equations are derived using Galerkin's method.
This approach is followed here, since it makes it easier to view the finite element discretization procedure as a simple means of finding finite approximate solutions for the governing differential equations, and boundary conditions of the problem.

The unknown velocity potential solution can be approximated in each element by:

$$\phi^e = \sum N_i \phi_i = [N]\{\phi\}^e \tag{4}$$

where $\{\phi\}^e$ is the column vector of the potential function values, ϕ_i, at the nodes of the element e, and [N] is the row vector of the interpolation functions N_i. The value of N_i is one at the node i and zero at the other nodes.

The integral statement of the problem is obtained by requiring the weighted average of the error resulting from the substitution of equation (4) into equation (2) to vanish over the domain D. The weighted residual techniques differ in their choice of the weighting functions. In the Galerkin's method, the weighting functions are chosen to be the same as N_i, leading to the following equation:

$$\int_D N_i \; \nabla \cdot (\rho \nabla \phi) \; dv = 0 \tag{5}$$

The boundary conditions are naturally introduced through integration by parts, which leads to the following relation:

$$\int_D \nabla N_i \cdot (\rho \nabla \phi) = \int_{\partial D} N_i \, \rho \, V_n \; dS \tag{6}$$

where V_n is the flow velocity component perpendicular to the boundary, ∂D, of the domain D. The right-hand side of equation (6) can be recognized as the weighted average of the mass flow across the boundary ∂D. This equation holds for the entire solution domain as well as for any arbitrary subdomain, or element, of the whole region. This last choice results in the desired element equation:

$$\int_{D_e} \nabla N_i \cdot (\rho \nabla [N] \{\phi\}^e) dv = \int_{\partial D_e} N_i \rho V_n \, dS \tag{7}$$

where the potential function representation ϕ^e within the element, as given by equation (4) has been used in the left-hand side of the above equation. For incompressible flow, only the interpolation functions, N_i, are dependent on the space coordinates, and the following expression is obtained for the element's equation:

$$\int_{D_e} \left[\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y} + \frac{\partial N_i}{\partial z} \frac{\partial N_j}{\partial z}\right] \phi_j \, dv$$

$$= \int_{\partial D_e} N_i V_n \, dS \tag{8}$$

Once the element shape, and the set of approximation functions are chosen, these equations can be written explicitly for each element. The resulting equations are then assembled for all the elements in the flow region and their solution results in the desired values of the nodal potential functions.

With the large number of nodal points that are required in the flow field discretization, it has been necessary to restrict the solution to the two dimensional case at the present time. The element's equation reduces to the following form in this case:

$$\iint_{D_e} \left[\frac{\partial N_i}{\partial x} \frac{\partial N_j}{\partial x} + \frac{\partial N_i}{\partial y} \frac{\partial N_j}{\partial y}\right] \phi_j \, dxdy = \int_{\partial D_e} N_i V_n \, d\Sigma \tag{9}$$

where $d\Sigma$ is the differential arc length along the boundary.

It is well known in finite element applications that better accuracy can be achieved at the cost of problem categories to be solved for a given computer capacity [23]. Since no similar solution of flow problems in terms of the degree of domain connectability is known by the authors to exist in the literature, a discretization procedure using the simplest linear straight sided elements was followed. The domain discretization is shown schematically in Fig. 2, where small size elements are used in the nozzle channels, with the size of the elements increasing gradually toward the outer portions in the scroll, to reflect the expected variations in the flow field [14].

The solution domain is divided into a number of continuous nonoverlapping triangular finite elements as shown in Fig. 2. After assembling all the elements equations, the integrals of the normal velocities over the elements sides will cancel, with the exception of those that coincide with the permeable boundaries, where the normal velocities are specified. The resulting set of algebraic equations can be written in the following form:

$$[K]\{\phi\} = \{F\} \qquad (10)$$

where [K] is the stiffness matrix of coefficients, whose elements depend solely on the elements shape for a given interpolation function, and $\{\phi\}$ is the column vector of unknown potential functions. The vector {F} depends on the specified flow velocity component, perpendicular to the boundaries at inlet to the scroll and exit from the nozzle. Although the stiffness matrix in finite element solutions is usually symmetric and banded, the introduction of the splitting lines in the flow field downstream of the nozzle blades result in a nonsymmetric, nonbanded sparse stiffness matrix.

The Substructuring Technique

In a general potential flow problem involving large numbers of unknown nodal values, the classical method of assembling the finite element equations can be lengthy and costly in terms of computer time. This problem can be alleviated through using a technique in which the solution domain is divided into a number of subdomains called substructures. The basic idea behind the substructuring technique lies in eliminating the unknown field variables associated with nodes inside each substructure (termed interior nodes). The number of unknowns is hence reduced to the number of nodes existing on the boundary and the interfaces between the substructures which are referred to as the exterior nodes.

Illustrative Example

In order to illustrate and explain the substructuring techniques, let us apply it to a simple potential flow problem with a simply connected domain shown schematically in Fig. 3a. For this purpose, the global equation (10) is rewritten in the following form:

$$[A]\ \{x\} = \{B\} \qquad (11)$$

where, [A] is the global stiffness matrix,

{x} is the column vector of the unknown nodal values of the velocity potential, and

{B} is the overall load vector.

Let the solution domain be divided into three substructures as shown in Fig. 3b. The stiffness matrix [A] in equation (9) is now partitioned in conjunction with the substructured domain giving rise to the following set of equations:

$$
\begin{aligned}
&[S_{11}]\{\bar{X}\} + [S_{12}]\{X_1\} + [S_{13}]\{X_2\} + [S_{14}]\{X_3\} = \{R\} \\
&[S_{21}]\{\bar{X}\} + [S_{22}]\{X_1\} = 0 \\
&[S_{31}]\{\bar{X}\} + [S_{33}]\{X_2\} = 0 \\
&[S_{41}]\{\bar{X}\} + [S_{44}]\{X_3\} = 0
\end{aligned}
\qquad (12)
$$

where, $\{\bar{X}\}$ stands for the column vector of the potential discrete values at the exterior nodes of all three substructures,

$\{X_i\}$ is the array of the potential values at the interior nodes of the ith substructure $(1 \leq i \leq 3)$; and

$[S_{ij}]$ refers to the different partitions of the stiffness matrix.

It can be seen from equation (12) that only the exterior nodes are associated with nonzero partitions of the load vector. Next, the unknown vectors $\{x_i\}$ $(1 \leq i \leq 3)$ are eliminated from the system of equations (12) resulting in the following 'reduced' set of equations,

$$[\bar{A}]\{\bar{X}\} = \{R\} \qquad (13)$$

where $[\bar{A}]$ is the reduced stiffness matrix which can be expressed as follows:

$$[\bar{A}] = [S_{11}] - \sum_{i=1}^{3} [S_{1,i+1}][S_{i+1,i+1}]^{-1}[S_{i+1,1}] \qquad (14)$$

This expression can be further simplified for a symmetric global stiffness matrix [A] as follows:

$$[\bar{A}] = S_{11} - \sum_{i=1}^{3} [S_{1,i+1}][S_{i+1,i+1}]^{-1}[S_{1,i+1}]^{t} \qquad (15)$$

The effectiveness of the substructuring technique, in computer storage requirements, is dependent on the relative dimension of the matrix $[\bar{A}]$ compared to $[A]$. The improvement becomes more pronounced as the ratio of interior to exterior nodes in each substructure becomes substantially higher than one.

Upon the solution of the reduced system of equations represented by equation (13), the unknown vectors $\{X_i\}$ at the interior nodes of the various substructures can be evaluated through a back substitution process as follows:

$$\{X_i\} = - [S_{i+1,i+1}]^{-1} [S_{1,i+1}]^t \{\bar{X}\} \;, \; 1 \leq i \leq 3 \tag{16}$$

The extension of the foregoing procedure to a total of 'N' substructures is straight forward. In this case, the reduced stiffness matrix takes the following form:

$$[\bar{A}] = [S_{11}] - \sum_{i=1}^{N} [S_{1,i+1}][S_{i+1,i+1}]^{-1}[S_{1,i+1}]^t \tag{17}$$

The nodal values of the potential at the interior nodes of the ith substructure can similarly be computed as follows:

$$\{X_i\} = - [S_{i+1,i+1}]^{-1} [S_{1,i+1}]^t \{\bar{X}\}, \quad 1 \leq i \leq N \tag{18}$$

<u>Application of the Technique to the Flow Field in the Stator Unit</u>:

Figure 2 shows the substructuring and the discretization model used in the present analysis. Each substructure consists of a blade-to-blade nozzle channel and its corresponding downstream vaneless nozzle and upstream scroll passage sections. The first substructure includes only the scroll inlet portion. The nodes, which are on the splitting lines, represent in this case exterior nodes for the various substructures. When applying the substructuring technique to our problem, special handling of the multiply connected region is required. As explained in the analysis, the pairs of permeable boundaries that extend from the trailing edges of the nozzle guide vanes to the flow exit station, reduce the solution region into a simply connected domain. These are treated as fictitious boundaries across which the potential is allowed to be discontinuous while the continuity of the velocity vector is introduced as a detailed boundary condition [24]. The discrete values of the velocity components along the splitting lines were taken as unknowns. The same velocity components were also used to evaluate the boundary term associated with the finite elements on both sides of the splitting boundaries. The basic difference from the simple example discussed earlier is in the assymmetry of the stiffness matrix introduced by the splitting boundaries. With the exception of the set of equations representing the compatability conditions at these boundaries, the rest of the matrix of coefficients is symmetric.

The same principles discussed in the preceding chapter are applied to this latter matrix giving the following equations:

$$
\begin{aligned}
&[S_{11}]\ \{\bar{\phi}\} + \sum_{i=1}^{N} [S_{1,i+1}]\ \{\phi_i\} + [G]\ \{V\} = \{R\} \\
&[S_{21}]\ \{\bar{\phi}\} + [S_{22}]\ \{\phi_1\} = 0 \\
&\quad\vdots \\
&[S_{N+1,1}]\ \{\bar{\phi}\} + [S_{N+1,N+1}]\ \{\phi_N\} = 0 \\
&[H]\ \{\bar{\phi}\} + [T]\ \{V\} = 0
\end{aligned}
\qquad (19)
$$

where, N is the total number of substructures,

$\{\bar{\phi}\}$ is the column vector of the nodal values of the function at the exterior nodes of all substructures,

$\{\phi_i\}$ $1 \leq i \leq N$, is the column vector of the potential values at the interior nodes of the ith substructure.

$\{V\}$ is the column vector of the discrete values of the velocity component normal to the splitting lines and potential discontinuities; and

$[G]$ is the rectangular matrix of coefficients corresponding to the normal velocities in $\{V\}$.

In this case, the reduced stiffness matrix can be written as follows:

$$[\bar{A}] = [S_{11}] - \{\sum_{j=2}^{N+1} [S_{1j}][S_{jj}]^{-1}[S_{1j}]^{t}\} - [G][T]^{-1}[H] \qquad (20)$$

RESULTS AND DISCUSSION

A computer program was developed which discretizes the solution domain, generates the stiffness matrix, and solves numerically the resulting set of equations. The input to the program includes geometric data such as the inlet scroll width, the inner and outer diameters of the vaned region, the number of blades and their configuration. A particular operating condition, i.e. volume flow rate, is specified by the given inlet velocity, from which the radial velocity component at exit is calculated in accordance with the principle of mass conservation. Different exit station locations were tried in the numerical solution, and the one chosen was just far enough from the blades, that any further inward radial displacement did not influence the rest of the flow field. A listing of the program, a glossary of the main program and subroutine variables, and a description of the input and output are given in the appendix.

The results are presented in nondimensional form, with the normalized potential, ϕ^*, defined as follows:

$$\phi^* = \frac{\phi}{V_1 L_1}$$

where V_1 is the uniform inlet flow velocity, and L_1 is a characteristic dimension at the scroll inlet. The resulting velocity potential distribution is then used to determine the flow velocity, q, which is also presented in the following dimensionless form:

$$q^* = \frac{q}{\sqrt{\gamma R T_t}}$$

where γ is the gas specific heat ratio, R is the gas constant and T_t is the total temperature.

The incompressible two dimensional flow field computations were carried out for an assembly of linearly varying area scroll with two different nozzle blade configurations. The inner and outer radii of the vaned nozzle region, the number of nozzle vanes, as well as the scroll geometry were kept the same, while two different nozzle blades were investigated. The chord length was equal to 0.048 meters for both cambered and symmetric blade geometries, with thirteen blades in the vaned nozzle region between the radii of 0.0725 and 0.0885 meters. In both cases it was found sufficient to place the exit station at three quarter the radial chord downstream of the vanes trailing edges.

All the geometrical data, with the exception of the uncambered blades was taken similar to reference [13]. The actual scroll area variation was converted into an equivalent two dimensional passage to be consistent with our two dimensional analysis. As a result of the linear variation in the scroll width from the inlet value of 0.1875 meters the scrolls outer boundary made an angle of 80 degrees with the radial direction.

The normalized velocity potential and flow velocity distributions are shown in Figs. 4 and 5 for the middle three channels between the cambered nozzle blades. The exit flow angle from the radial direction station was found to be equal to 74 degrees in this case. It must be noted that the flow velocities are inversely proportional to the sine of this angle in the case of incompressible flow. The results, obtained for the case of symmetric nozzle vanes are shown in Figs. 6 and 7. Both symmetric and unsymmetric nozzle vanes had the same chord length and setting angle. Comparing Figs. 4 and 5 with Figs. 6 and 7, it can be seen that higher flow velocities are generally observed in the vaned nozzle region, and at the exit station in the case of symmetric nozzle vanes. The flow direction is also different in the two cases, and the flow exit angle of 78 degrees from the radial direction resulted in the case of symmetric vanes.

In the numerical solution, the flow velocities, as well as the velocity potential were calculated throughout the whole flow field. The mass flow through each nozzle channel between two consecutive blades was also determined, to find out the deviation from the desired uniform distribution. The mass flow distribution between the thirteen nozzle channels is shown in Fig. 8. The mass flow in each channel of that figure is normalized with respect to the value that would pass in each channel, if the total mass flow was equally divided between all the nozzle channels. From the computed results shown in this figure, it can be seen that the largest deviations are observed in the first and last nozzle channels. More mass flow passes in these two channels than in the rest. The least deviation from the average value is observed at the middle flow channels. Practically the same mass flow distribution was also observed in the different channels between the symmetric nozzle blades. This leads us to believe that the deviations are mostly affected by the geometry of the scroll, rather than that of the nozzle vanes.

The effect of using the substructuring technique on reducing the computer time and storage requirements was investigated. This effect is demonstrated by comparing these computer requirements when the substructuring technique is used to those when it is not. In both cases, partially packed arrays were used to store the non-zero elements of the matrix of coefficients [25]. The results of this comparison are presented in Table 1, from which it is clear that a drastic reduction in both the core size and CPU time is achieved when the substructuring technique is used. In the present study we have not tried to maximize this reduction, which is generally dependent on the ratio of the number of interior to exterior nodes in the substructures. This technique becomes more effective as the ratio is increased. The choice of the substructures in our problem was strongly dictated by the domain geometry and its multiconnectivety, which in turn is determined by the number of nozzle blades.

CONCLUSIONS

The results presented here were obtained from two dimensional incompressible flow computations. The analysis can be extended to the compressible flow case without a drastic increase in computer core storage requirements. The generalization to the three dimensional flow will require careful considerations in order to maintain the core requirement at a reasonable level. The two dimensional flow results presented here, can provide a fair description of the nozzle flow at the vanes midspan plane. The computed velocity distribution over the nozzle vane surfaces, and the flow exit angle can therefore be used to evaluate the performance of the different vane shapes.

REFERENCES

1. Hamed, A., Sheoran, Y. and Tabakoff, W., "Cooling Considerations for Design of a Radial Inflow Turbine," ASME Paper No. 77-GT-82, March 1977.
2. Rodgers, C., "Advanced Radial Inflow Turbine Rotor Program Design and Dynamic Testing," NASA CR-135080, September 1976.
3. Calvert, G.S. and Okapuu, U., "Design and Evaluation of a High Temperature Radial Turbine," Phase I - Final Report, USAAVLABS Technical Report 68-69.
4. Watanabe, I., Ariga, I. and Mashimo, T., "Effect of Dimensional Parameters of Impellers on Performance Characteristics of a Radial Inflow Turbine," ASME Paper No. 70-GT-90.
5. Watanabe, I. and Ando, T., "Experimental Study on Radial Turbine With Special Reference to the Influence of the Number of Impeller Blades on Performance Characteristics," Bulletin of JSME, Vol. 2, No. 7, 1959, pp. 457-462.
6. Kastner, L.J. and Bhinder, F.S., "A Method for Predicting the Performance of a Centripetal Gas Turbine Fitted with a Nozzleless Volute Casing," ASME Paper No. 75-GT-65.
7. Wallace, F.J., Cave, P.R. and Miles, J., "Performance of Inward Radial Flow Turbines Under Steady Flow Conditions With Special Reference to High Pressure Ratios and Partial Admission," Proc. Instn. Mech. Engrs., Vol. 184, Part 1, No. 56, 1969-1970.
8. Futral, S.M. and Wasserbauer, C.A., "Off-Design Performance Prediction with Experimental Verification for a Radial Inflow Turbine," NASA TND 2621, 1965.
9. Lagneau, J.P., "Contribution to the Study of Advanced Small Radial Gas Turbines," VKI IN38, March 1970.
10. Hamed, A. and Baskharone, E., "Analysis of the Three Dimensional Flow in a Turbine Scroll," ASME Paper, accepted for presentation to the Symposium on "Flow in Primary, Non-Rotating Passages in Turbomachines," in the ASME Winter Annual Meeting.

11. Knoernschild, E.M., "The Radial Turbine for Low Specific Speeds and Low Velocity Factors," Journal of Engineering for Power, Trans. ASME, Series A, Vol. 83, No. 1, January 1961, pp. 1-8.

12. Bhinder, F.S., "Investigation of the Flow in the Nozzleless Spiral Casing of a Radial Inward-Flow Gas Turbine," Proc. Inst. Mech. Engrs., Vol. 184, Pt. 36 (11), pp. 66-71.

13. Khalil, I.M., Tabakoff, W. and Hamed, A., "Losses in Radial Inflow Turbines," Journal of Fluids Engineering, Trans. ASME, Series I, Vol. 98, Sept. 1976, pp. 364-373.

14. Hamed, A., Abdallah, S. and Tabakoff, W., "Flow Study in the Cross Sectional Plane of a Turbine Scroll," AIAA Paper No. 77-714, presented at AIAA 10th Fluid and Plasma Dynamics Conference, Albuquerque, NM, June 27-29, 1977.

15. de Vries, G. and Norrie, D.H., "The Application of the Finite Element Technique to Potential Flow Problems," Journal of Applied Mechanics, Vol. 38, 1971, pp. 798-802.

16. Thompson, D.S., "Flow Through a Cascade of Aerofoils," International Symposium of Finite Element Methods in Flow Problems, University of Wales, Swansea, Wales, 1974.

17. Prince, T.C., "Prediction of Transonic Inviscid Steady Flow in Cascades by Finite Element Methods," General Electric Report No. R76AEG192.

18. Katsanis, T., "Fortran Program for Calculating Transonic Velocities on a Blade-to-Blade Stream Surface of a Turbomachine," NASA TND-5427, September 1969.

19. Huebner, K., The Finite Element Method for Engineers, 1975.

20. Norrie, D.H. and de Vries, G., "The Finite Element Method Fundamentals and Applications," 1973.

21. Zienkiewicz, O.C., "The Finite Element Method in Engineering Science," 1971.

22. Conner, J.J. and Brebbia, C.A., "Finite Element Techniques for Fluid Flow," 1976.

23. Shen, S., "Finite-Element Methods in Fluid Mechanics," Annual Review of Fluid Mechanics, Vol. 9, 1977, pp. 421-445.

24. Schechter, R.S., "The Variational Method in Engineering," McGraw Hill, 1967.

25. Gupta, S.K. and Tanji, K.K., "Computer Program for Solution of Large, Sparse, Unsymmetric Systems of Linear Equations," Int. J. Num. Meth. Engrg., Vol. 11, No. 8, 1977, pp. 1251-1259.

NOMENCLATURE

D	the solution domain
f	the value of the flow velocity component normal to the boundary
{F}	the load vector
[K]	the stiffness matrix
L	inlet width of the two dimensional scroll
n	the direction normal to the boundary
{N}	the column vector of the elements interpolation functions
$\bar{V}$	flow velocity vector
q	flow velocity
ρ	flow density
ϕ	the flow velocity potential

Subscripts

1	refer to conditions at inlet to the scroll
2	refer to conditions at exit from the nozzle
e	refer to a finite element
i	refer to variables at the node i
n	refer to the direction perpendicular to the solution domain boundary

Superscripts

e	associated with the finite element
*	nondimensional parameters

Method of Assembling F.E. Equations	From Contributions of Elements	From Contributions of Substructures
CORE USED	1,052 K	588 K
CPU SECONDS	507	234
KILOBYTE-MINUTES	18,605	6,246

TABLE 1. EFFECT OF USING THE SUBSTRUCTURING TECHNIQUE ON COMPUTER TIME AND STORAGE.

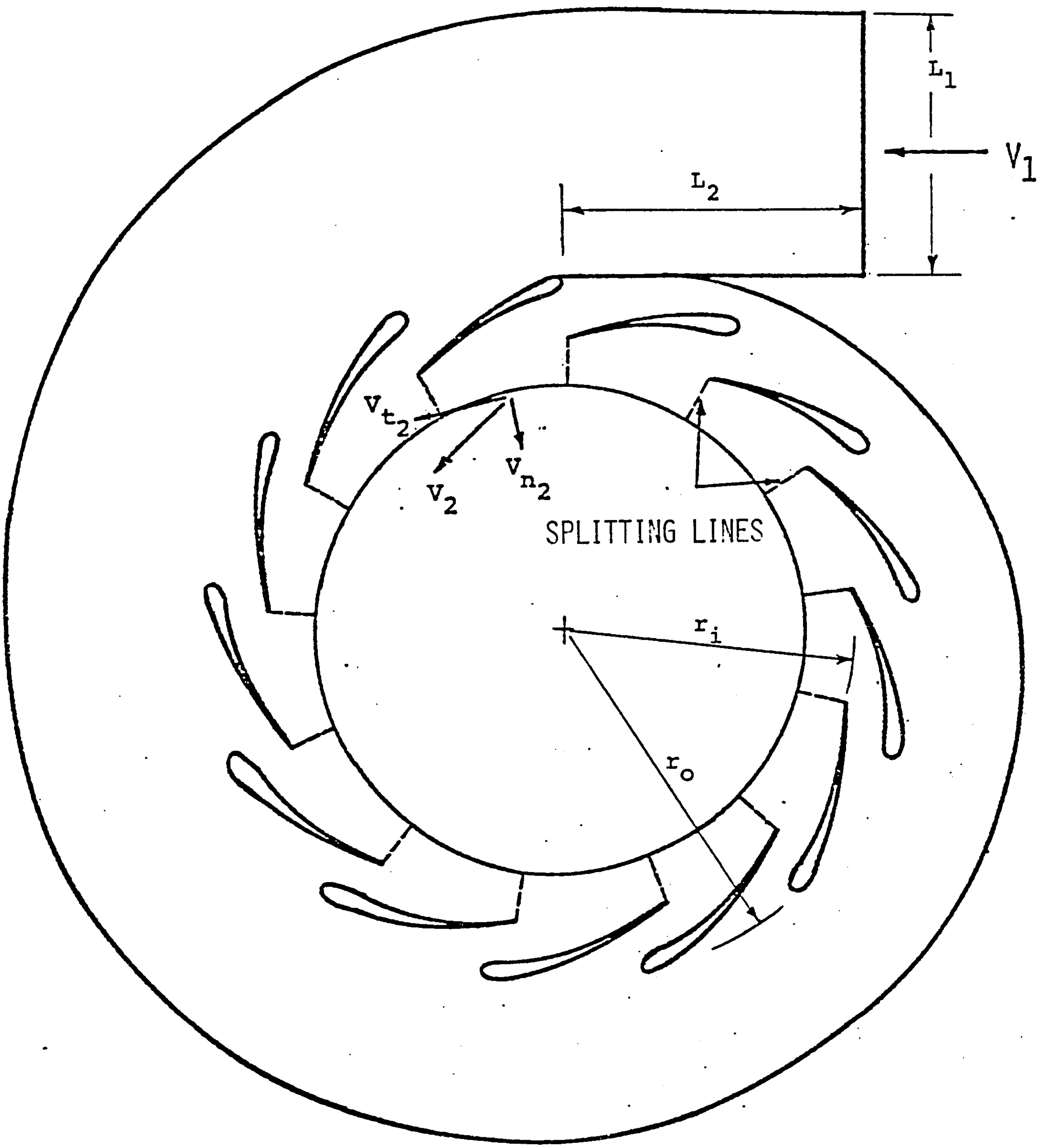


FIG. 1. STATOR UNIT OF A RADIAL TURBINE.

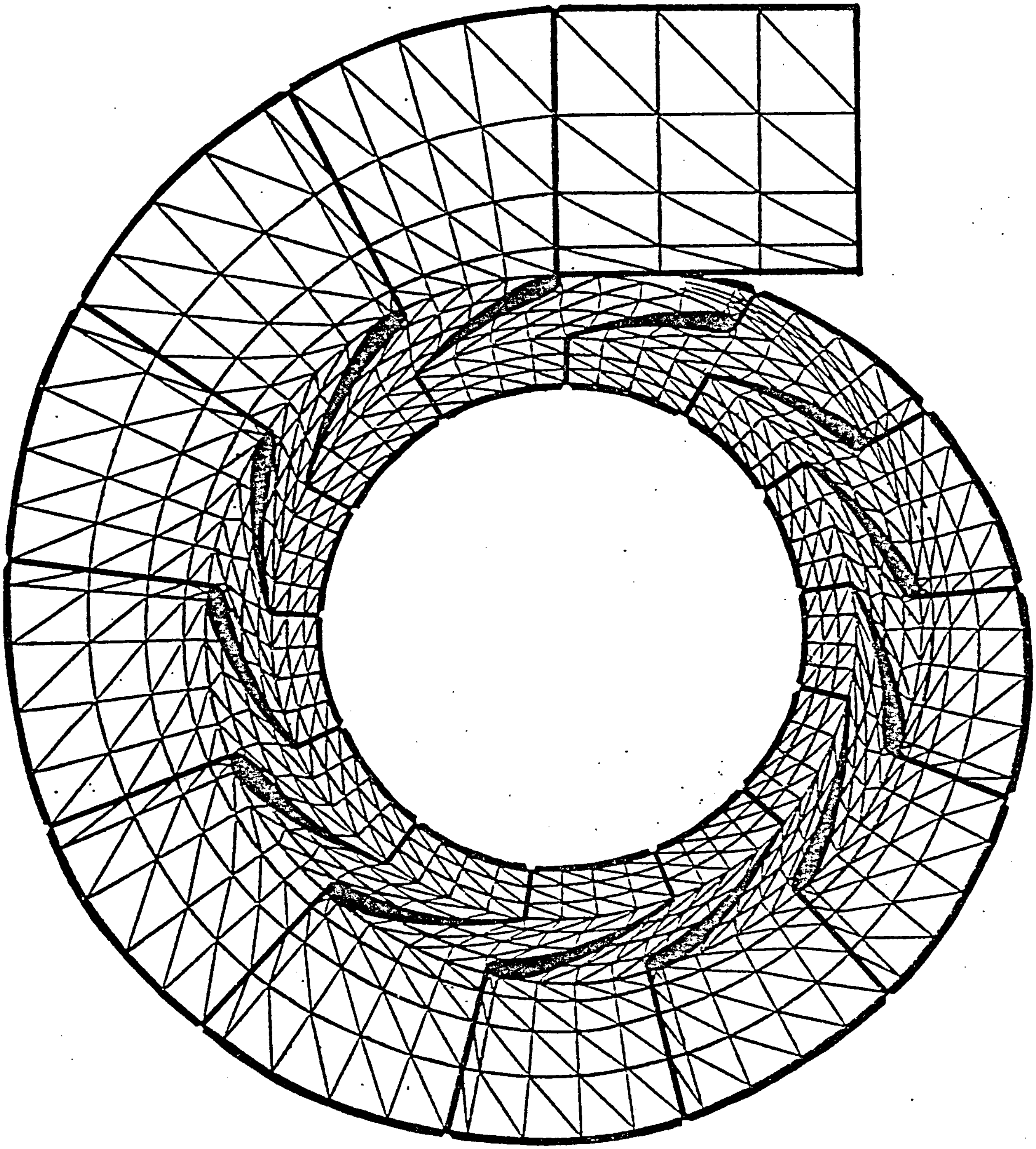
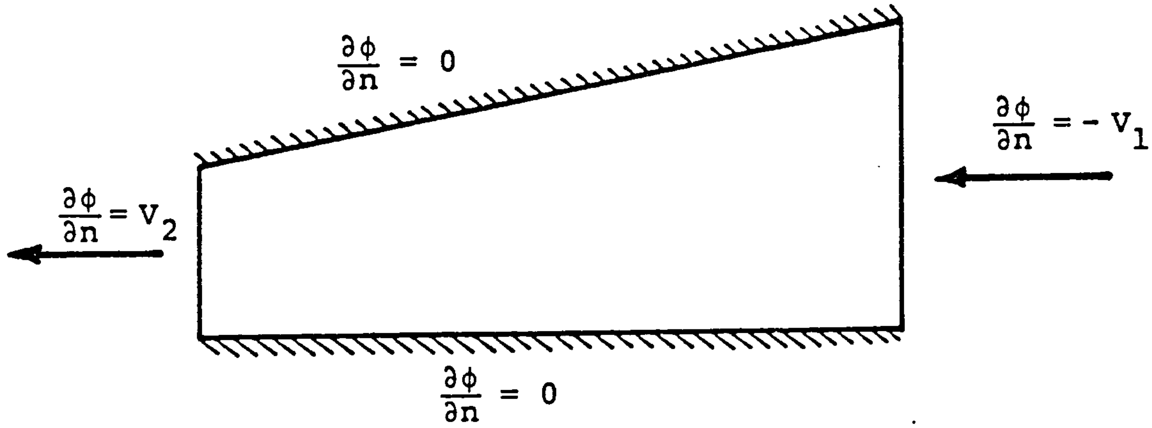
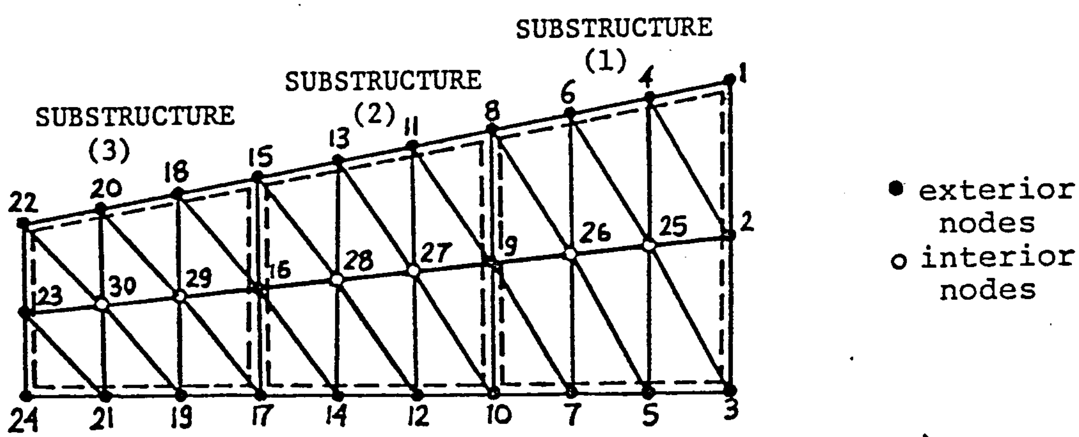


FIG. 2. THE FINITE ELEMENT DISCRETIZATION MODEL.



a) FLOW REGION AND BOUNDARY CONDITIONS



b) FINITE ELEMENT MODEL

FIG. 3. SOLUTION DOMAIN OF THE EXAMPLE PROBLEM.

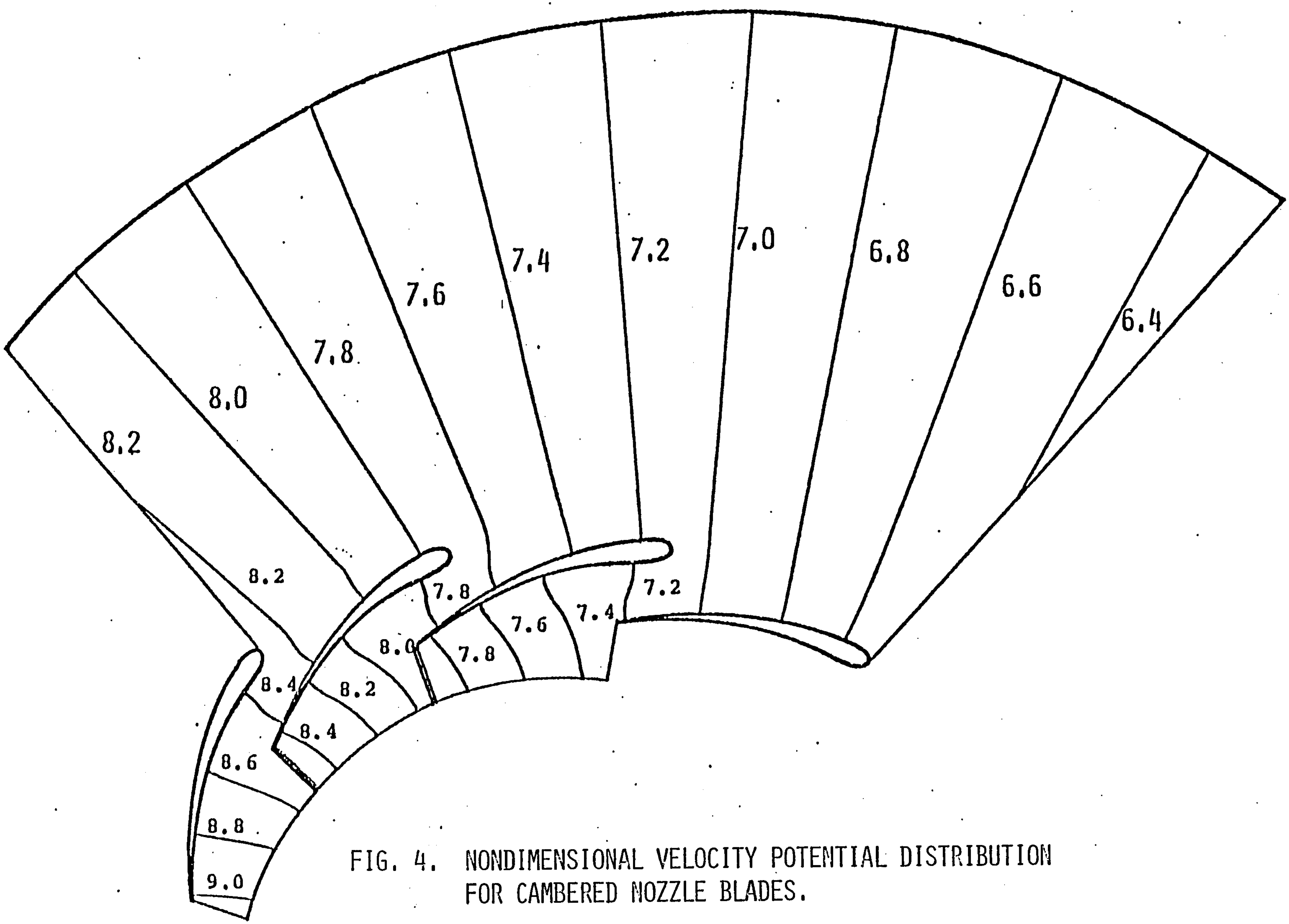


FIG. 4. NONDIMENSIONAL VELOCITY POTENTIAL DISTRIBUTION FOR CAMBERED NOZZLE BLADES.

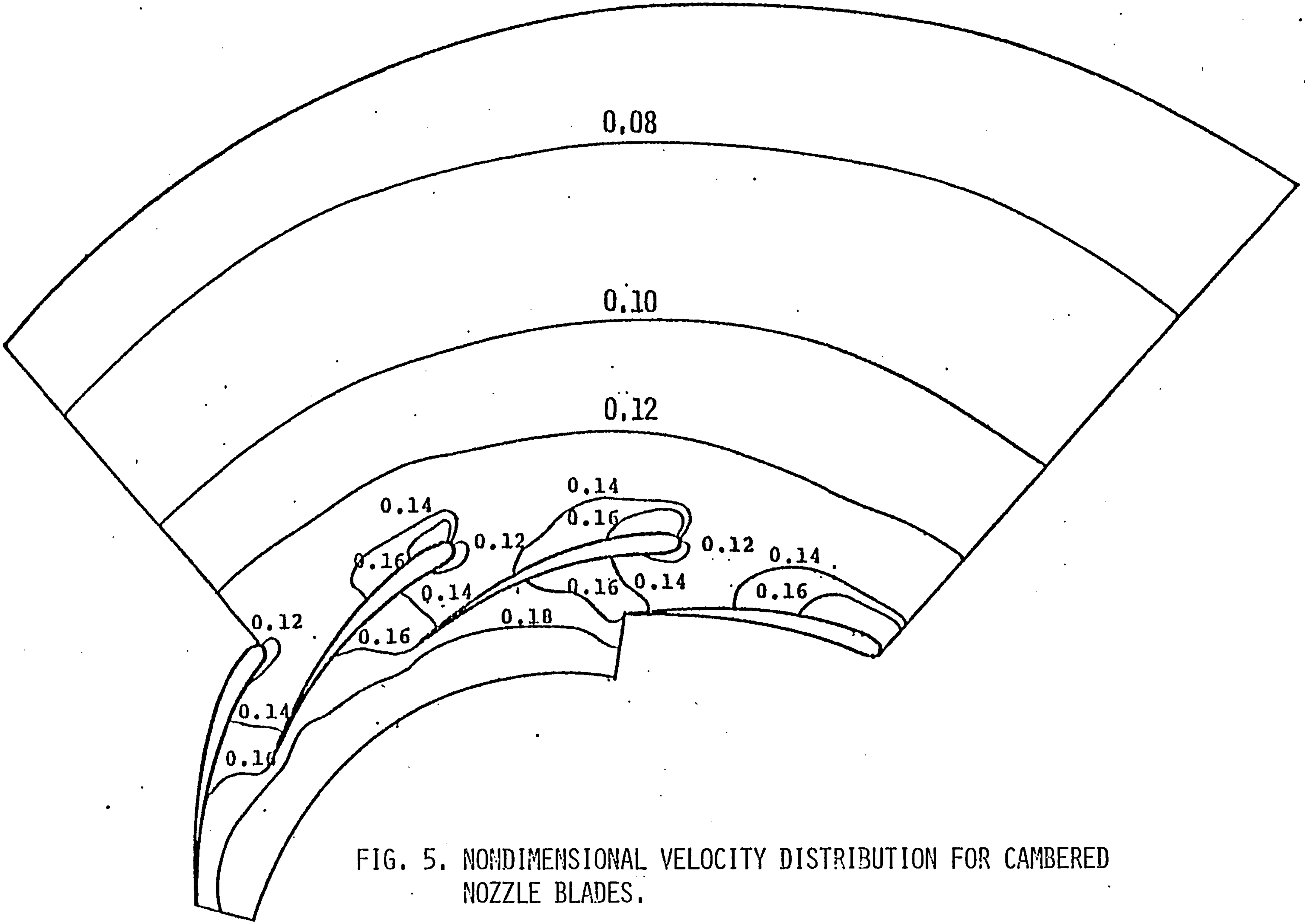


FIG. 5. NONDIMENSIONAL VELOCITY DISTRIBUTION FOR CAMBERED NOZZLE BLADES.

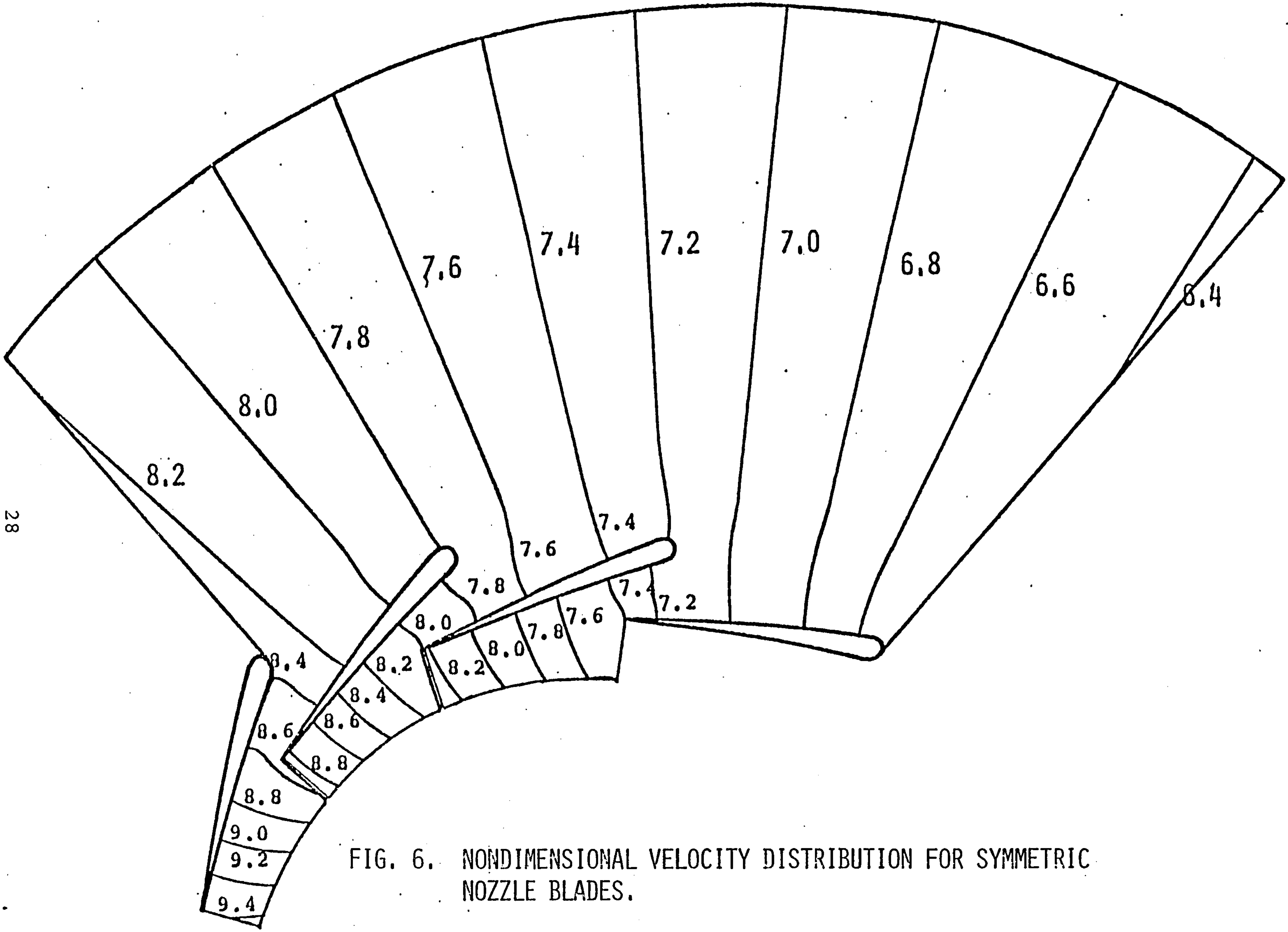


FIG. 6. NONDIMENSIONAL VELOCITY DISTRIBUTION FOR SYMMETRIC NOZZLE BLADES.

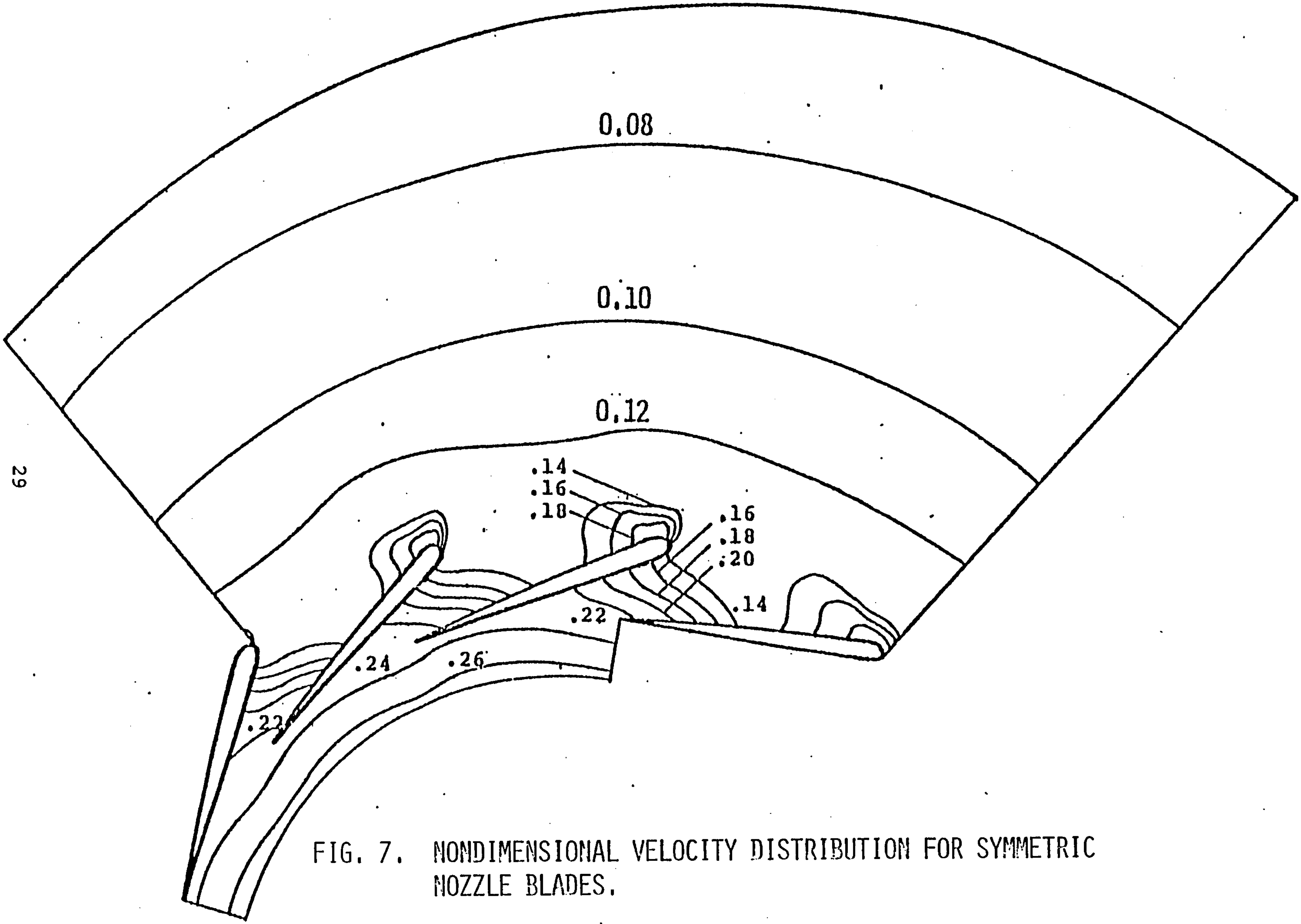


FIG. 7. NONDIMENSIONAL VELOCITY DISTRIBUTION FOR SYMMETRIC NOZZLE BLADES.

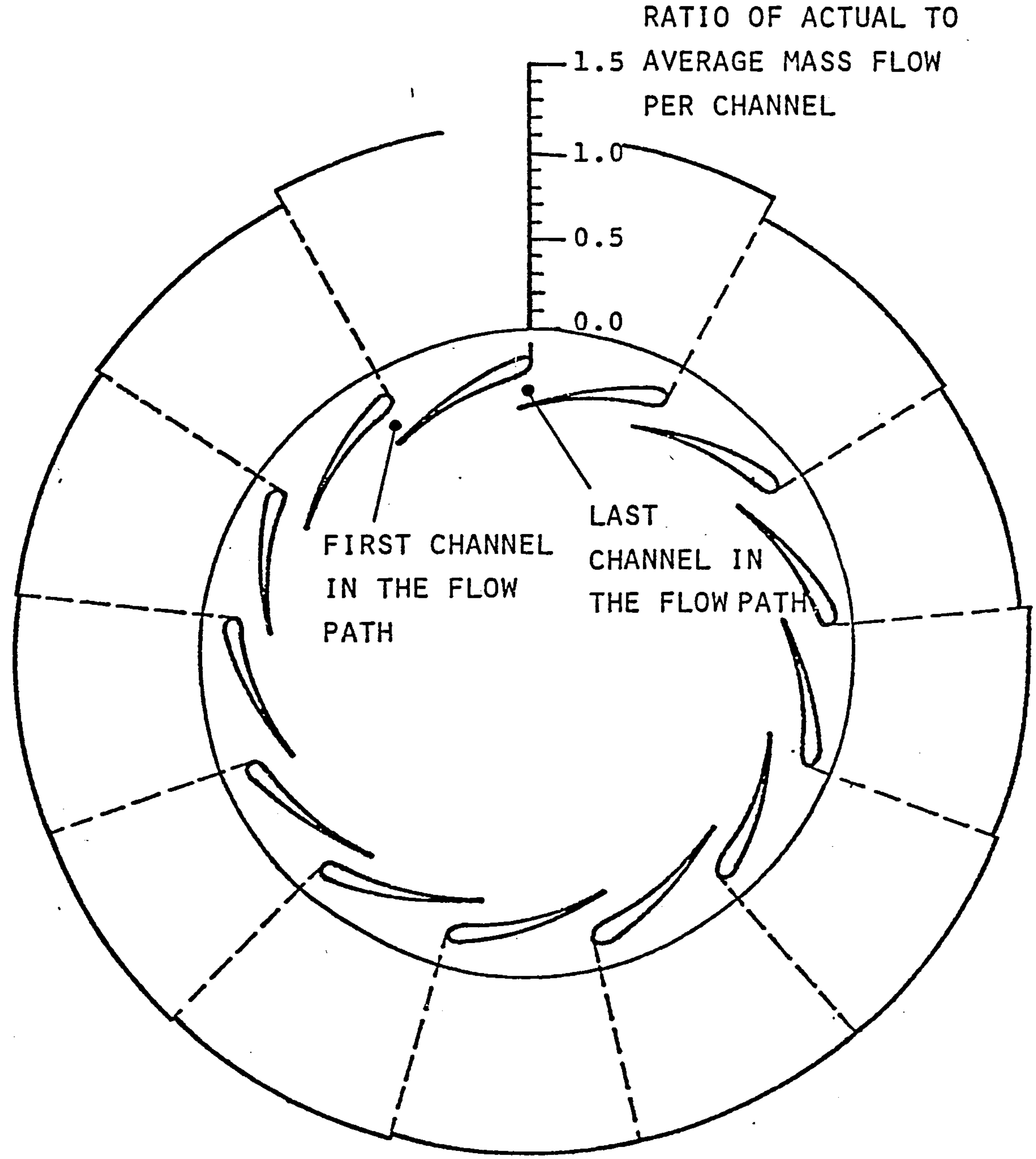


FIG. 8. NONDIMENSIONAL MASS FLOW DISTRIBUTION THROUGH THE DIFFERENT NOZZLE CHANNELS.

APPENDIX

A FORTRAN COMPUTER PROGRAM FOR CALCULATING THE FLOW FIELD IN THE SCROLL-NOZZLE ASSEMBLY OF A RADIAL TURBINE

The computer program discretizes the solution domain and provides the finite element solution and the flow properties. The user can conveniently specify adequate mesh size in the different portions of the flow region through the discretization input parameters in the input data. A schematic of the finite element discretization model is shown in Fig. 2, with the substructures consisting of a nozzle channel and the corresponding scroll and vaneless portions. This model is used in all but the first substructure, which consists of the scroll inlet portion.

The program consists of several subroutines which are called by the main program. First the subroutine 'DATA' reads all the input data including the stator unit basic dimensions, the finite element discretization parameters and the operating conditions. The spatial distribution of the nodal points and the system topology are computed by the two subroutines 'CORDNT' and 'DISCRT'. In subroutine 'MAP', all the nodes are renumbered, with the degrees of freedom corresponding to the exterior nodes going first, followed by the interior variables. Considering one substructure at a time, the subroutine 'SUBSTR' computes the corresponding stiffness matrix. Subroutine 'SUBSTR' also develops the functional relations which assure a locally compatible velocity vector across the splitting lines downstream of the nozzle vanes. This is accomplished by calling the three subroutines 'CONTRB', 'PLACE' and 'INTRNL'. The 'load vector' is computed is subroutine 'DERFAI' by assembling all the mass flux contributions of all the designated boundary segments. Subroutine 'MODIFY' stores the resulting finite element equations in partially packed arrays. These equations are solved for the potential values at the exterior nodes, in subroutine 'EQSOLV'. The potential values at all interior nodes are evaluated in subroutine 'SOLVE' by a back substitution process. The flow velocity and thermodynamic properties in the entire domain are computed in subroutines 'PHYS' and the mass flux in the various nozzle channels are determined in subroutine 'MASFLW'.

Glossary of Program Variables:

The following section contains the definition of the principal variables in the main program and subroutines, and includes all the common variables. The names of all dimensional arrays are followed by the variables that determines what their dimensions should be. An asterisk is used to denote the variables in the program input data.

A(NPT,NNN)	The partially packed array containing the nonzero elements of the reduced stiffness matrix [matrix $[\bar{A}]$ in eq. (20)].
AAA (NNN)	The array where the nonzero elements of the pivotal row are temporarily stored in the process of solving the set of finite element equations.
B (IMAXI,NRED)	The matrix S^t_{1j} in eq. (20).
BB (3,3)	The finite element stiffness matrix.
C (NRED,IMAXI)	The matrix S_{1j} in eq. (20).
DENS	The stagnation density (kg/mt^3).
FAI (MTØT)	The velocity potential at the nodal points (mt^2/sec).
GAMA*	Ratio of specific heats.
GAMM* (IDN)	The angular spacing between the blade convex and concave surfaces (γ_1, γ_2,... etc. in Fig. A2), (radians).

INTERN (KTØT,IMAXI)	The nodal numbers of the exterior nodes in a substructure.
IMG (MTØT)	The array used for storing the nodal numbers in the substructured solution domain.
IK*	The number of blade-to-blade scroll circumferential stations to be used in the discretization (N_5 in Fig. A1).
IBANDW (NPT)	The array containing the number of nonzero elements in the rows of the array [A].
ICOL (NPT,NNN)	The partially packed array of the column indices corresponding to the entries of the array [A].
IDN*	The number of radial stations in the nozzle vaned portion, not including that at the nozzle blades trailing edges [N_3 in Fig. A1].
IMAXB	The maximum number of exterior nodes in any substructure.
IMAXI	The maximum number of interior nodes in any substructure.
ISTEP*	The number of nodes existing on a radial line in the scroll passage (N_1 in Fig. A1).
ISTRT*	The number of cross sectional stations in the scroll inlet portion (N_2 in Fig. A1).
ILG (NPT)	A temporary storage array for the column indices of the nonzero elements in any row of the [A] array.
KTØT	The number of substructures = NBLD + 1.
KT	The number of radial stations in the scroll curved portion.
L,M and N (NTØT)	The numbers assigned to the nodal points of a finite element.

LBC and MBC (LTOT) — The numbers assigned to the nodal points of a boundary segment where the normal derivative of the potential function is specified and not equal to zero.

LQQ and MQQ (IMAXI) — Temporary integer arrays used in inverting of the matrix $[S_{jj}]$ in eq. (20).

LTØT — The number of boundary segments where the normal derivative of the velocity potential is specified.

MAXWID — An output parameter referring to the column dimension of the arrays (A) and (ICOL) actually used in the equation solver.

MTØT — The total number of nodes in the finite element model.

MINV — An IBM library program used for inverting a square matrix.

NBLD* — The number of nozzle vanes.

NNCØL (NPT) — The array of column indices of the entries in any row of the array [A].

NNN* — The dimension for the columns [A] and [ICOL] arrays which should be greater than the maximum number of nonzero elements in any row of these arrays.

NPT — The dimension for the rows of the [A] and [ICOL] arrays.

NPIV (NPT) — The array of column indices associated with the entries of the [AAA] array.

NRED — The total number of exterior nodes for all substructures.

NSUBB (KTOT)	The number of exterior nodes in each substructure.
NSUBI (KTØT)	The number of interior nodes in a substructure.
NTØT	The total number of finite elements in the solution domain.
NUNV*	The number of radial stations downstream of the nozzle vanes (N_4 in Fig. 7).
P (NPT)	The load vector in the system of finite element equations.
PROP (MTOT, 7)	Temporary storage for the computed flow thermodynamic properties averaged at each node.
PSTG*	Stagnation pressure (Newtons/mt^2)
R*	The gas constant (Joules/kg °K)
RI*	The inner radius of the vaned nozzle (mt)
RO*	The outer radius of the vaned nozzle (mt).
S (IMAXI,IMAXI)	The matrix $[S_{jj}]$ in equation (20).
STOR (IMAXI)	The nodal values of the potential function at the interior nodes of a substructure.
STRTL*	The length of the scroll entry portion (L_2 in Fig. 1) (mt).
STWDTH*	The width of the scroll passage at the inlet station (L_1 in Fig. 1) (mt).
TEM (NPT)	Temporary storage for the nonzero elements in any row of the matrix [A].
THET* (IDN+1)	Angular coordinates of the nodes on the convex surface of the first nozzle blade (θ_1, θ_2, ...etc. in Fig. A2) (radians).

TSTG*	Stagnation temperature (°K)
U1 (NTØT)	The x-velocity component in the finite elements (mt/sec).
V1 (NTØT)	The y-velocity component in the finite elements (mt/sec).
VN1*	The inlet flow velocity (V_1 in Fig. 1) (mt/sec)
VN (LTOT)	The nonzero normal derivatives of the potential function at the boundary segments (mt/sec).
WDTH (KT)	The scroll passage width at the designated stations (mt).
X (MTOT)	The x-coordinates of all the nodal points (mt).
XY (MTOT)	Temporary storage for both [X] and [Y] arrays.
Y (MTOT)	The y-coordinate of all the nodal points (mt).
ZTEST	The minimum allowable value for any of the entries of the array [AAA] before it is considered a zero element.

PROGRAM LISTING

```
C
C
C
C        A   F O R T R A N   C O M P U T E R   P R O G R A M
C
C             FOR CALCULATING THE FLOW FIELD PARAMETERS
C     IN THE SUBSTRUCTURED DOMAIN OF A RADIAL TURBINE STATOR UNIT
C
C
C
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/MNL/LMT(78),LNT(78)
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTOKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/ADL/NU
      COMMON/DEL/IDUM,NR,NC
      COMMON/SHO/LZ,MZ,NZ
      COMMON/MAX/PIVROW,PIVCOL,OPROW,IY
      COMMON/GAW/ICTT
      COMMON/EEW/IQD
      COMMON/MAR/TEM(457),ILG(457)
      COMMON/RTH/AAA(457),IBANDW(457),NPIV(457),NNCOL(457),NPT,NNN
      COMMON/NAL/QUANT,IRR,ILL
      COMMON/ZFF/DIS(84)
      COMMON/EZZ/U1(1812),V1(1812)
      COMMON/FLT/IRAD
      COMMON/ACM/PRCP(817,7)
      CALL UNDFLW
      CALL DATA
      CALL CORDNT
      CALL DISCRT
      CALL MAP
      CALL DERFAI
      CALL SUBSTR
      CALL EQSOLV
      CALL SOLVE
      CALL PHYS
      CALL MASFLW
      STOP
      END
```

```
      SUBROUTINE DATA
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/MNL/LMT(78),LNT(78)
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTCT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTCT1,MTCT2,MRM,IMM,MAA
      COMMON/GAM/MTOKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/ADL/NU
      COMMON/DEL/IDUM,NR,NC
      COMMON/SHO/LZ,MZ,NZ
      COMMON/MAX/PIVROW,PIVCOL,OPROW,IY
      COMMON/GAW/ICTT
      COMMON/EEW/IQD
      COMMON/MAR/TEM(457),ILG(457)
      COMMON/RTH/AAA(457),IBANDW(457),NPIV(457),NNCOL(457),NPT,NNN
      COMMON/NAL/QUANT,IRR,ILL
      COMMON/ZFF/DIS(84)
      COMMON/DOM/CONS
      COMMON/FLT/IRAD
C
C
C   READ & PRINT THE INPUT DATA
C
C
      WRITE(6,200)
      READ(5,100)R,GAMA,PSTG,TSTG,VN1
      WRITE(6,210)
      WRITE(6,220)R,GAMA,PSTG,TSTG,VN1
      READ(5,110)STWDTH,STRTL,RO,RI
      WRITE(6,230)
      WRITE(6,240)STWDTH,STRTL,RO,RI
      READ(5,120)NBLD,ISTRT,ISTEP,IDN,NUNV,IK
      WRITE(6,250)
      WRITE(6,260)NBLD,ISTRT,ISTEP,IDN,NUNV,IK
      READ(5,130)(THET(I),I=1,IDN)
      WRITE(6,270)(THET(I),I=1,IDN)
      IDM=IDN-1
```

```
      READ(5,130)(GAMM(I),I=1,IDM)
      WRITE(6,280)(GAMM(I),I=1,IDM)
      READ(5,140)NNN
      WRITE(6,290)NNN
C
C
C   CALCULATE THE PROGRAM CONSTANTS
C
C
      IK=IK+1
      IRAD=IK
      ISTRT=ISTRT-1
      NSTPS=IDN+NUNV-1
      ISTEP=ISTEP+1
      IDN=IDN-1
      IDNM=IDN+1
      NUNV=NUNV-1
      KTOT=NBLD+1
      NRED=((NBLD+2)*(ISTEP-1))+((NSTPS-1)*2*NBLD)
     X+((IK-3)*NBLD*2)+((NBLD-1)*3)+((ISTRT-1)*2)
      KT=NBLD*(IK-1)
      IMAXB=((ISTEP+NSTPS-2)*2)+((IK-3)*2)+3
      IMAXI=((ISTEP+NSTPS-4)*(IK-3))+ISTEP-3
      NCONT=((NSTPS-IDN-1)*2)*NBLD
      MC=NNN
      DRN=(RO-RI)/IDN
      IDNL=IDN+2
      IL=IK-1
      DO 10 I=IDNL,NSTPS
   10 THET(I)=THET(IDNM)
      DO 20 I=1,IDN
   20 BET(I)=(6.283185-(NBLD*GAMM(I)))/(NBLD*(IL-1))
      DO 30 I=IDNM,NSTPS
   30 BET(I)=6.283185/(NBLD*(IL-1))
      DANG=BET(1)
      DTHET=GAMM(1)
      IL3=IK-1
      JSTRT=ISTEP-1
      IST=ISTRT*JSTRT
      KMM=NSTPS-IDN
      MTOT=((ISTEP-1)*ISTRT)+((((ISTEP-2)+NSTPS)*(IK-1)*NBLD)-((NSTPS-
     XIDN)*(NBLD-1))+NBLD-1+(KMM*(NBLD-1))
      NTOT=((((ISTEP-2)*(ISTRT+KT-1))+((IK-2)*(NSTPS-1)*NBLD))*2)+
     XNBLD-1
      LTOT=((IK-2)*NBLD)+ISTRT-1
      NW=MTOT-NSTPS+1
      IU1=ISTEP-2
      IST2=((ISTRT+1)*JSTRT)+1
      IEL=JSTRT+NSTPS-2
      ITA=IEL-IU1
      IIN=(IU1*ISTRT*2)+1
      LIN=IST+1
      IN1=IEL+1
```

```
      NBW=IN1+2
      IN2=IK-2
      IU2=IU1+1
      IM1=IST2+IDN-1
      DENS=PSTG/(R*TSTG)
      IA=KMM+1
      IM2=MTOT-KMM+1
      IB=IA+KMM-1
      NBW1=NBW+1
      NBWP=NBW+2
      MATCH=KMM+1
      MTOT3=MTOT+KMM-1
      MATCH1=MATCH-1
      MTOTP=MTOT+(NBLD*(KMM-1)*2)
      MTOTM=MTOTP-1
      IMA=IM2+1
      IMB=MTOT-1
      MTOT1=MTOT+1
      MTOT2=MTOT+2
      IMM=IM1+KMM-1
      MAA=NBW1+KMM
      MTOTKM=MTOT-KMM+1
      NU=MTOTP
      MR=(2*NBW)+2
      MRM=MC-1
      ZTEST=10.0**(-40.0)
      ILL=IK-2
      ISM=IST+ISTEP-2+NSTPS
      CONS=STWDTH*VN1*DENS
      NPT=NRED+(NBLD*(KMM-1)*2)
      NRED1=NRED+1
      NINT=MTOT-NRED
C
C
C  CALCULATE THE SCROLL PASSAGE WIDTH AT THE APPROPRIATE
C  TANGENTIAL LOCATIONS
C
C
      I=1
      ATHET=0.0
   25 JK=1
   27 WDTH(I)=STWDTH*(1.0-(ATHET/6.283185))
      I=I+1
      IF(I.GT.KT) GO TO 29
      JK=JK+1
      IF(JK.GT.IL3) GO TO 28
      ATHET=ATHET+DANG
      GO TO 27
   28 ATHET=ATHET+DTHET
      GO TO 25
   29 CONTINUE
      WDTH(1)=STWDTH
      WDTH(KT)=0.1*(RO-RI)
```

```
C
C
C   CALCULATE THE NODAL NUMBERS OF THE END POINTS OF THE INLET
C   AND EXIT BOUNDARY SEGMENTS
C
C
      I=1
      LBC(I)=1
   63 MBC(I)=LBC(I)+1
      I=I+1
      IF(I.GT.IU1) GO TO 64
      LBC(I)=MBC(I-1)
      GO TO 63
   64 I=IU1+1
      JW=1
      ICH=1
      LBC(I)=ISM
   61 MBC(I)=LBC(I)+ISTEP-2+NSTPS
      I=I+1
      JW=JW+1
      IF(JW.GT.ILL) GO TO 62
      LBC(I)=MBC(I-1)
      GO TO 61
   62 ICH=ICH+1
      JW=1
      IF(ICH.GT.NBLD) GO TO 69
      LBC(I)=MBC(I-1)+IN1+1
      GO TO 61
   69 VN2=(VN1*STWDTH)/(6.283185*(RI-((KMM-1)*DRN)))
      DO 70 I=1,IU1
   70 VN(I)=-1.0*VN1
      DO 80 I=IU2,LTOT
   80 VN(I)=VN2
      WRITE(6,350)
      WRITE(6,360)MTOT,NTOT,LTOT,KTOT,NRED,NPT,IMAXB,IMAXI
      WRITE(6,370)KT
      DO 83 I=1,KT
   83 WRITE(6,380)I,WDTH(I)
   88 DO 90 I=1,NTOT
   90 RHO(I)=DENS
C
C
C   ASSIGN ZERO VALUES FOR ALL ELEMENTS OF THE OTHER ARRAYS
C
C
      DO 92 I=1,NPT
      DO 91 J=1,NNN
      A(I,J)=0.0
      ICOL(I,J)=0
   91 CONTINUE
   92 CONTINUE
      DO 93 I=1,NPT
   93 P(I)=0.0
```

```
      DO 95 I=1,IMAXI
      DO 94 J=1,IMAXI
      S(I,J)=0.0
   94 CONTINUE
   95 CONTINUE
      DO 97 I=1,IMAXI
      DO 96 J=1,NRED
      B(I,J)=0.0
   96 CONTINUE
   97 CONTINUE
      DO 99 I=1,NRED
      DO 98 J=1,IMAXI
      C(I,J)=0.0
   98 CONTINUE
   99 CONTINUE
      DO 192 I=1,IMAXI
      LQQ(I)=0
      MQQ(I)=0
  192 CONTINUE
      DO 8202 I=1,NPT
      AAA(I)=0.0
      IBANDW(I)=0
      NPIV(I)=0
      NNCOL(I)=0
 8202 CONTINUE
C
C
C  FORMAT STATEMENTS
C
C
  100 FORMAT(5F10.0)
  110 FORMAT(4F10.0)
  120 FORMAT(6I5)
  130 FORMAT(8F10.0)
  140 FORMAT(I5)
  200 FORMAT(1H1/10X,'INPUT DATA'/10X,10('*')/)
  210 FORMAT(21X,'R',11X,'GAMA',11X,'PSTG',11X,'TSTG',12X,'VN1')
  220 FORMAT(7X,5(3X,E12.5)/)
  230 FORMAT(16X,'STWDTH',10X,'STRTL',13X,'RO',13X,'RI')
  240 FORMAT(7X,4(3X,E12.5)/)
  250 FORMAT(11X,'NBLD',3X,'ISTRT',3X,'ISTEP',5X,'IDN',4X,'NUNV'
     X,6X,'IK')
  260 FORMAT(7X,6(3X,I5)//)
  270 FORMAT(10X,'ARRAY THET:',7(1X,E12.5))
  280 FORMAT(10X,'ARRAY GAMM:',7(1X,E12.5))
  290 FORMAT(//12X,'NNN'/10X,I5////10X,
     X'E N D   O F   I N P U T   D A T A')
  350 FORMAT(1H1/10X,'PROGRAM PARAMETERS AS CALCULATED FROM THE INPUT DA
     XTA'/10X,52('*')/)
  360 FORMAT(10X,'MTOT=',I5/10X,'NTOT=',I5/10X,'LTOT=',I5/10X,'KTOT=',I5
     X/10X,'NRED=',I5/10X,'NPT=',I5/10X,'IMAXB=',I5/10X,'IMAXI=',I5////
     X5X,'NOTE:'/5X,5('*')/10X,'AT THIS STEP , IT IS IMPORTANT TO COMPAR
     XE THE PRECEEDING VALUES WITH THE DIMENSIONS OF THE ARRAYS IN THE C
     XOMMON STATEMENTS'/10X,'THE LATTER SHOULD BE OF GREATER OR , AT LEA
     XST , EQUAL MAGNITUDES')
  370 FORMAT(1H1/40X,'ARRAY OF THE SCROLL PASSAGE WIDTH'
     X/40X,33('*')/48X,'AT',I5,2X,'STATIONS'/48X,17('*')//45X
     X,'STATION',12X,'WIDTH'/)
  380 FORMAT(47X,I5,5X,E12.5)
      RETURN
      END
```

```
      SUBROUTINE CORDNT
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/MNL/LMT(78),LNT(78)
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTCKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/ADL/NU
      COMMON/DEL/IDUM,NR,NC
      COMMON/SHO/LZ,MZ,NZ
      COMMON/MAX/PIVROW,PIVCOL,OPROW,IY
      COMMON/GAW/ICTT
      COMMON/EEW/IQD
      COMMON/MAR/TEM(457),ILG(457)
      COMMON/RTH/AAA(457),IBANDW(457),NPIV(457),NNCOL(457),NPT,NNN
      COMMON/NAL/QUANT,IRR,ILL
      COMMON/ZFF/DIS(84)
      COMMON/FLT/IRAD
C
C
C  CALCULATE THE NODAL COORDINATES OF THE SCROLL CURVED PORTION
C
C
      IZT=0
      IZF=ISTEP-2
      DO 388 IJ=1,IZF
  388 IZT=IZT+IJ
      I=1
      JW=0
      JK=0
      GO TO 101
   99 JK=JK+1
  101 IF(JK.EQ.JSTRT) GO TO 102
      IZD=IZF-JK
      IF(JK.EQ.IZF) GO TO 91
      IPLC=0
      DO 89 IJ=1,IZD
```

```
 89 IPLC=IPLC+IJ
    Y(I)=RO+((IPLC*STWDTH)/IZT)
    GO TO 92
 91 Y(I)=RO
 92 CONTINUE
    X(I)=STRTL-((JW*STRTL)/JSTRT)
    I=I+1
    GO TO 99
102 JW=JW+1
    IF(JW.EQ.ISTRT) GO TO 103
    JK=0
    GO TO 101
103 IST1=IST+1
    IL1=IK-2
    KM1=0
    J=1
    ANGL=1.570796
  2 IN=0
  6 ICT=0
  9 RA=RO+WDTH(J)
  7 IF(I.GT.NW) GO TO 14
    X(I)=RA*COS(ANGL)
    Y(I)=RA*SIN(ANGL)
    ICT=ICT+1
    IF(ICT.GT.IU1) GO TO 8
    I=I+1
    IZD=IZF-ICT
    IPLC=0
    IF(IZD.EQ.0) GO TO 593
    DO 93 IJ=1,IZD
 93 IPLC=IPLC+IJ
593 RA=RO+((IPLC*WDTH(J))/IZT)
    GO TO 7
  8 IF(KM1.NE.1) GO TO 208
    I=I+1
    ANGL=ANGL-(DTHET/2.0)
    RA=RO
    X(I)=RA*COS(ANGL)
    Y(I)=RA*SIN(ANGL)
208 IN=IN+1
    IF(IN.GT.IL1) GO TO 10
    J=J+1
    I=I+NSTPS
    ANGL=ANGL+DANG
    IF(KM1.EQ.1) ANGL=ANGL+(DTHET/2.0)
    KM1=0
    GO TO 6
 10 ANGL=ANGL+DTHET
    J=J+1
    I=I+NSTPS
    KM1=1
    GO TO 2
 14 CONTINUE
```

```
C
C
C  CALCULATE THE NOZZLE VANES NODAL COORDINATES
C
C
      IDNN=IDN+1
   21 IK1=IK-1
      KM2=0
      I=IST2
      IND=0
      ALFA=THET(2)
   15 IM=0
   18 J=1
   16 IF(J.GE.1.AND.J.LT.IDN) ALFA=THET(J+1)+(IM*BET(J+1))+(IND*(((IK
     X-2)*BET(J+1))+GAMM(J+1)))
      IF(J.GE.IDN.AND.J.LE.ITA) ALFA=THET(J+1)+(IM*BET(J+1))+(IND*(((IK
     X-2)*BET(J+1))))
      IF(J.GT.ITA) GO TO 17
      IF(J.GT.IDN) GO TO 9998
      RN=RO-(J*DRN)
      GO TO 8889
 9998 RN=RO-(IDN*DRN)-((J-IDN)*DRN)
 8889 IF(I.GT.MTOT) GO TO 221
      X(I)=RN*COS(ALFA)
      Y(I)=RN*SIN(ALFA)
      J=J+1
      I=I+1
      IF(KM2.EQ.1.AND.J.EQ.IDN) GO TO 37
      GO TO 16
   17 IM=IM+1
      IF(IM.EQ.IK1) GO TO 19
      I=I+ISTEP-1
      KM2=0
      GO TO 18
   19 IND=IND+1
      I=I+ISTEP
      KM2=1
      GO TO 15
   37 IBX=I-IN1-1
      IBY=IBX+KMM-1
      ICX=I
      ICY=ICX+KMM-1
      DO 38 JK=ICX,ICY
      I=JK
      JL=JK-ICX+IBX
      X(JK)=X(JL)
      Y(JK)=Y(JL)
   38 CONTINUE
      I=I+ISTEP
      KM2=0
      IM=IM+1
      GO TO 18
  221 CONTINUE
      WRITE(6,9977)
      DO 9900 I=1,MTOT
 9900 WRITE(6,9988)I,X(I),Y(I)
 9977 FORMAT(1H1//41X,'NODAL COORDINATES'/41X,17('*')///32X,'NODE',9X
     X,'X-COORD',7X,'Y-COORD'/)
 9988 FORMAT(31X,I5,4X,E12.5,2X,E12.5)
      RETURN
      END
```

```
      SUBROUTINE DISCRT
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,CRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTCKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/FLT/IRAD
C
C
C   COMPUTE THE ORDERS OF EACH TRINGULAR ELEMENT THREE NODES
C
C
      ISTEN=ISTEP-1
      KAB=0
      IU3=IU1-1
      I=1
      L1=1
    2 ICOT=0
    3 L(I)=L1
      M(I)=L1+JSTRT
      N(I)=L1+1
      J=I+1
      L(J)=N(I)
      M(J)=M(I)
      N(J)=M(I)+1
      ICOT=ICOT+1
      IF(ICOT.EQ.IU1) GO TO 4
      I=J+1
      L1=L1+1
      GO TO 3
    4 I=J+1
      IF(I.EQ.IIN) GO TO 5
      L1=L1+2
      GO TO 2
    5 NSTP=IDN+KMM-1
      KM3=0
      I=IIN
```

```
      KX=IDN+IU1+1
      KY=IDN+IU1
      L1=LIN
 7727 IQ=0
   27 ICOT=1
      GO TO 38
   28 ICOT=ICOT+1
   38 L(I)=L1
      M(I)=L1+IN1
      IF(KAB.EQ.1.AND.ICOT.LE.ISTEN) M(I)=M(I)+1
      N(I)=L1+1
      IF(KAB.EQ.1.AND.ICOT.EQ.ISTEN) N(I)=N(I)+1
      J=I+1
      L(J)=N(I)
      M(J)=M(I)
      N(J)=M(I)+1
  225 I=J+1
      IF(I.GT.NTOT) GO TO 232
      IF(ICOT-IEL) 20,21,21
   20 L1=L1+1
      IF(KAB.EQ.1.AND.ICOT.EQ.ISTEN) L1=L1+1
      GO TO 28
   21 IQ=IQ+1
      IF(IQ.EQ.IN2) GO TO 22
      KAB=0
      L1=L1+2
      GO TO 27
   22 IU=0
      L1=L1+2
   26 L(I)=L1
      M(I)=L1+IN1
      N(I)=L1+1
      J=I+1
      L(J)=N(I)
      M(J)=M(I)
      N(J)=M(I)+1
      I=J+1
      IU=IU+1
      IF(IU.EQ.IU3) GO TO 23
      L1=L1+1
      GO TO 26
   23 L1=L1+1
      L(I)=L1
      M(I)=L1+IN1+2
      N(I)=L1+1
      J=I+1
      L(J)=L1
      M(J)=L1+IN1
      N(J)=L1+IN1+2
      K=J+1
      L(K)=L1+IN1+2
      M(K)=L1+IN1
      N(K)=L1+IN1+1
      I=K+1
      L1=L1+NSTPS+1
      KAB=1
      GO TO 7727
  232 RETURN
      END
```

```
      SUBROUTINE MAP
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTCT,NTOT,LTCT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTOKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHC(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/FLT/IRAD
C
C
C  COMPUTE THE ENTRIES OF ARRAY (NSUBB)
C
C
C
      NSUBB(1)=((ISTEP-1)*2)+((ISTRT-1)*2)
      NSUBB(2)=((ISTEP+NSTPS-2)*2)+((IK-3)*2)
      DO 7001 IJ=3,KTOT
 7001 NSUBB(IJ)=((ISTEP+NSTPS-2)*2)+((IK-2)*2)+1
C
C
C  COMPUTE THE ENTRIES OF ARRAY (NSUBI)
C
C
      NSUBI(1)=((ISTEP-3)*(ISTRT-1))
      NSUBI(2)=((ISTEP+NSTPS-4)*(IK-3))
      DO 7002 IJ=3,KTOT
 7002 NSUBI(IJ)=(ISTEP-3)+NSUBI(2)
C
C
C  COMPUTE THE ENTRIES OF ARRAY (IMG)
C
C
      ITG=IK-3
      ISTRTT=ISTRT-1
      ISE=ISTEP-3
      ISTEP1=ISTEP-1
      DO 7003 I=1,ISTEP1
 7003 IMG(I)=I
```

```
      I=ISTEP1+1
      ITANG=1
 7004 II1=ISTEP+((ITANG-1)*2)
      JJ1=NRED+((ITANG-1)*(ISTEP-3))+1
      IMG(I)=II1
      I=I+1
      I1=I
      I2=I1+ISTEP-4
      DO 7005 I=I1,I2
 7005 IMG(I)=JJ1+I-I1
      I=I2+1
      IMG(I)=II1+1
      ITANG=ITANG+1
      IF(ITANG.EQ.ISTRT) GO TO 7006
      I=I+1
      GO TO 7004
 7006 I=I+1
      II1=II1+2
      ILEN=ISTEP+NSTPS-2
      I1=I
      I2=I1+ILEN-1
      DO 7007 I=I1,I2
 7007 IMG(I)=II1+I-I1
      I=I2+1
      ITANG=1
 7008 CONTINUE
      II1=IMG(I-1)+1
      JJ1=JJ1+ISTEP-3
      IF(ITANG.GT.1) JJ1=JJ1+ILEN-ISTEP+1
      IMG(I)=II1
      I=I+1
      I1=I
      I2=I1+ILEN-3
      DO 7009 I=I1.I2
 7009 IMG(I)=JJ1+I-I1
      I=I2+1
      IMG(I)=II1+1
      ITANG=ITANG+1
      IF(ITANG.GT.ITG) GO TO 7010
      I=I+1
      GO TO 7008
 7010 I=I+1
      ICHNL=1
 7110 CONTINUE
      II1=IMG(I-1)+1
      I1=I
      I2=I1+ILEN-1
      DO 7011 I=I1,I2
 7011 IMG(I)=II1+I-I1
      IF(ICHNL.EQ.NBLD) GO TO 7019
      I=I2+1
 7012 II1=IMG(I-1)+NSTPS+2
      JJ1=JJ1+ILEN-2
```

```
      IMG(I)=II1
      I=I+1
      I1=I
      I2=I1+ISE-1
      DO 7013 I=I1,I2
 7013 IMG(I)=JJ1+I-I1
      I=I2+1
      KK1=II1-NSTPS
      IMG(I)=KK1
      I=I+1
      IMG(I)=KK1-1
      I=I+1
      II1=KK1+1
      I1=I
      I2=I1+NSTPS-2
      DO 7014 I=I1,I2
 7014 IMG(I)=II1+I-I1
      I=I2+1
      II1=IMG(I-1)+2
      JJ1=JJ1+ISE
      IMG(I)=II1
      I=I+1
      I1=I
      I2=I1+ILEN-3
      DO 7015 I=I1,I2
 7015 IMG(I)=JJ1+I-I1
      I=I2+1
      IMG(I)=II1+1
      I=I+1
      ITANG=1
 7016 II1=II1+2
      JJ1=JJ1+ILEN-2
      IMG(I)=II1
      I=I+1
      I1=I
      I2=I1+ILEN-3
      DO 7017 I=I1,I2
 7017 IMG(I)=JJ1+I-I1
      I=I2+1
      IMG(I)=II1+1
      ITANG=ITANG+1
      IF(ITANG.EQ.ITG) GO TO 7018
      I=I+1
      GO TO 7016
 7018 ICHNL=ICHNL+1
      I=I+1
      GO TO 7110
 7019 CONTINUE
C
C
C  CALCULATE THE ENTRIES OF ARRAY (IBOUND)
C
C
```

```
      I=1
      NS=NSUBB(I)
      IBB=1
      DO 7020 J=1,NS
 7020 IBOUND(I,J)=IBB+J-1
      I=I+1
      NG=NS
      NS=NSUBB(I)
      IBB=IBOUND(I-1,NG)-ISTEP+2
      DO 7021 J=1,NS
 7021 IBOUND(I,J)=IBB+J-1
      I=I+1
 7022 NG=NS
      NS=NSUBB(I)
      IBB=IBOUND(I-1,NG)-ILEN+1
      DO 7023 J=1,ISTEP1
 7023 IBOUND(I,J)=IBB+J-1
      J=ISTEP1
      J1=ISTEP1+1
      J2=NS
      IBB=IBOUND(I,J)+NSTPS
      DO 7024 J=J1,J2
 7024 IBOUND(I,J)=IBB+J-J1
      I=I+1
      IF(I.GT.KTOT) GO TO 7025
      GO TO 7022
 7025 CONTINUE
C
C
C  CALCULATE THE ENTRIES OF ARRAY (INTERN)
C
C
      I=1
      I1=NRED+1
 7026 I2=I1+NSUBI(I)-1
      NSUBII=NSUBI(I)
      DO 7027 J=1,NSUBII
 7027 INTERN(I,J)=I1+J-1
      I=I+1
      IF(I.GT.KTOT) GO TO 7028
      I1=I2+1
      GO TO 7026
 7028 CONTINUE
C
C
C  CHANGE THE SYSTEM TOPOLOGY IN ACCORDANCE WITH
C  THE SUBSTRUCTURED FINITE ELEMENT MODEL
C
C
C
C
C  MAP THE SYSTEM TOPOLOGY INTO THE DESIRED F.E. MODEL
C
```

```
C
   10 DO 13 I=1,MTOT
      IMGG=IMG(I)
      XY(IMGG)=X(I)
   13 CONTINUE
      DO 14 I=1,MTOT
   14 X(I)=XY(I)
      DO 15 I=1,MTOT
      IMGG=IMG(I)
      XY(IMGG)=Y(I)
   15 CONTINUE
      DO 16 I=1,MTOT
   16 Y(I)=XY(I)
      DO 17 J=1,NTOT
      LQ=L(J)
      MQ=M(J)
      NQ=N(J)
      L(J)=IMG(LQ)
      M(J)=IMG(MQ)
      N(J)=IMG(NQ)
   17 CONTINUE
      DO 18 K=1,LTOT
      LBCC=LBC(K)
      MBCC=MBC(K)
      LBC(K)=IMG(LBCC)
      MBC(K)=IMG(MBCC)
   18 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CONTRB(J)
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICOMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTOKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/SHO/LZ,MZ,NZ
      COMMON/GAW/ICTT
      COMMON/EEW/IQD
      COMMON/DOM/CONS
      COMMON/FLT/IRAD
      LZ=L(J)
      MZ=M(J)
      NZ=N(J)
C
C
C   COMPUTE THE CONTRIBUTION MATRIX OF THE TRIANGULAR ELEMENT UNDER
C   CONSIDERATION
C
C
   10 AR=DABS((X(MZ)*Y(NZ))-(X(NZ)*Y(MZ))+(X(NZ)*Y(LZ))-(X(LZ)*Y(NZ))
     X+(X(LZ)*Y(MZ))-(X(MZ)*Y(LZ)))*0.5
      B1=Y(MZ)-Y(NZ)
      B2=Y(NZ)-Y(LZ)
      B3=Y(LZ)-Y(MZ)
      C1=X(NZ)-X(MZ)
      C2=X(LZ)-X(NZ)
      C3=X(MZ)-X(LZ)
      GO TO(20,30,40),ID
   20 BB(1,1)=(((B1*B1)+(C1*C1))*CONS)/(4.0*AR)
      BB(1,2)=(((B1*B2)+(C1*C2))*CONS)/(4.0*AR)
      BB(1,3)=(((B1*B3)+(C1*C3))*CONS)/(4.0*AR)
      GO TO 50
   30 BB(2,1)=(((B2*B1)+(C2*C1))*CONS)/(4.0*AR)
      BB(2,2)=(((B2*B2)+(C2*C2))*CONS)/(4.0*AR)
      BB(2,3)=(((B2*B3)+(C2*C3))*CONS)/(4.0*AR)
      GO TO 50
   40 BB(3,1)=(((B3*B1)+(C3*C1))*CONS)/(4.0*AR)
      BB(3,2)=(((B3*B2)+(C3*C2))*CONS)/(4.0*AR)
      BB(3,3)=(((B3*B3)+(C3*C3))*CONS)/(4.0*AR)
   50 RETURN
      END
```

```
      SUBROUTINE DERFAI
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      CCMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTCKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      CCMMON/FLT/IRAD
      DO 50 I=1,LTOT
C
C
C   CALCULATE THE LOCAL LOAD VECTOR CORRESPONDING TO A BOUNDARY
C   SEGMENT WHERE THE VALUE OF THE MASS FLUX IS PRESCRIBED
C
C
      LB=LBC(I)
      MB=MBC(I)
      DX=X(MB)-X(LB)
      DY=Y(MB)-Y(LB)
      DX1=ABS(DX)
      DY1=ABS(DY)
      DIST=((DX1**2.0)+(DY1**2.0))**0.5
   30 CONST=0.5*VN(I)*DIST*DENS
   40 P(LB)=P(LB)+CCNST
      P(MB)=P(MB)+CCNST
   50 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SUBSTR
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/MNL/LMT(78),LNT(78)
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICOMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTOKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/ADL/NU
      COMMON/GAW/ICTT
      COMMON/EEW/IQD
      COMMON/MAR/TEM(457),ILG(457)
      COMMON/RTH/AAA(457),IBANDW(457),NPIV(457),NNCOL(457),NPT,NNN
      COMMON/NAL/QUANT,IRR,ILL
      COMMON/FLT/IRAD
      DIMENSION RB(33,33)
C
C
C   CONSTRUCT THE REDUCED STIFFNESS MATRIX
C
C
      NUNV=NSTPS-IDN-1
      NCRD=IDN
      DO 520 K=1,KTCT
      KSUB=NSUBB(K)
      DO 120  II=1,KSUB
      IW=0
      IZ=0
      I=IBOUND(K,II)
      IF(K.EQ.1) GO TO 215
      IMTT=0
      KKJJ=K-1
      DO 201  IJ=1,KKJJ
      IISUB=NSUBB(IJ)
      DO 200  IK=1,IISUB
      IF(I.NE.IBOUND(IJ,IK)) GO TO 200
      IMTT=1
  200 CONTINUE
```

```
201 CONTINUE
    IF(IMTT.EQ.1) GO TO 120
215 DO 110 J=1,NTOT
    IW=0
    IZ=0
    IF(I.NE.L(J)) GO TO 50
    ID=1
    DO 40 NN=1,NRED
    IF(IW.EQ.1.AND.IZ.EQ.1) GO TO 110
    IF(M(J).NE.NN) GO TO 30
    IW=1
    ID=1
    CALL CONTRB(J)
    IF(IZ.EQ.1) GO TO 29
    IRR=I
    ILL=I
    QUANT=BB(1,1)
    CALL PLACE
 29 IRR=I
    ILL=M(J)
    QUANT=BB(1,2)
    CALL PLACE
    GO TO 40
 30 IF(N(J).NE.NN) GO TO 40
    IZ=1
    ID=1
    CALL CONTRB(J)
    IF(IW.EQ.1) GO TO 31
    IRR=I
    ILL=I
    QUANT=BB(1,1)
    CALL PLACE
 31 IRR=I
    ILL=N(J)
    QUANT=BB(1,3)
    CALL PLACE
 40 CONTINUE
    IF(IW.NE.0.OR.IZ.NE.0) GO TO 110
    CALL CONTRB(J)
    IRR=I
    ILL=I
    QUANT=BB(1,1)
    CALL PLACE
    GO TO 110
 50 IF(I.NE.M(J)) GO TO 80
    ID=2
    DO 70 NN=1,NRED
    IF(IW.EQ.2.AND.IZ.EQ.2) GO TO 110
    IF(L(J).NE.NN) GO TO 60
    ID=2
    IW=2
    CALL CONTRB(J)
    IRR=I
```

```
      ILL=L(J)
      QUANT=BB(2,1)
      CALL PLACE
      IF(IZ.EQ.2) GO TO 70
      IRR=I
      ILL=I
      QUANT=BB(2,2)
      CALL PLACE
   60 IF(N(J).NE.NN) GO TO 70
      ID=2
      IZ=2
      CALL CONTRB(J)
      IF(IW.EQ.2) GO TO 61
      IRR=I
      ILL=I
      QUANT=BB(2,2)
      CALL PLACE
   61 IRR=I
      ILL=N(J)
      QUANT=BB(2,3)
      CALL PLACE
   70 CONTINUE
      IF(IW.NE.0.OR.IZ.NE.0) GO TO 110
      CALL CONTRB(J)
      IRR=I
      ILL=I
      QUANT=BB(2,2)
      CALL PLACE
      GO TO 110
   80 IF(I.NE.N(J)) GO TO 110
      ID=3
      DO 100 NN=1,NRED
      IF(IW.EQ.3.AND.IZ.EQ.3) GO TO 110
      IF(L(J).NE.NN) GO TO 90
      ID=3
      IW=3
      CALL CONTRB(J)
      IRR=I
      ILL=L(J)
      QUANT=BB(3,1)
      CALL PLACE
      IF(IZ.EQ.3) GC TO 100
      IRR=I
      ILL=I
      QUANT=BB(3,3)
      CALL PLACE
   90 IF(M(J).NE.NN) GO TO 100
      ID=3
      IZ=3
      CALL CONTRB(J)
      IRR=I
      ILL=M(J)
      QUANT=BB(3,2)
```

```
      CALL PLACE
      IF(IW.EQ.3) GO TO 100
      IRR=I
      ILL=I
      QUANT=BB(3,3)
      CALL PLACE
  100 CONTINUE
      IF(IW.NE.0.OR.IZ.NE.0) GO TO 110
      CALL CONTRB(J)
      IRR=I
      ILL=I
      QUANT=BB(3,3)
      CALL PLACE
      GO TO 110
  110 CONTINUE
  120 CONTINUE
  520 CONTINUE
      CALL MODIFY
      DO 620 K=1,KTCT
      DO 20 IJ=1,IMAXI
      DO 10 IK=1,IMAXI
      S(IJ,IK)=0.0
   10 CONTINUE
   20 CONTINUE
      DO 24 IJ=1,IMAXI
      DO 22 IK=1,NRED
      B(IJ,IK)=0.0
   22 CONTINUE
   24 CONTINUE
      DO 3 IJ=1,NRED
      DO 2 IK=1,IMAXI
      C(IJ,IK)=0.0
    2 CONTINUE
    3 CONTINUE
      DO 8304 IJ=1,IMAXI
      LQQ(IJ)=0
      MQQ(IJ)=0
 8304 CONTINUE
      CALL INTRNL(K)
      NSUBI1=NSUBI(K)
      CALL RECIP(RB,NSUBI1)
 8200 DO 140 IJ=1,NRED
      DO 135 IK=1,NSUBI1
      C(IJ,IK)=B(IK,IJ)
  135 CONTINUE
  140 CONTINUE
      DO 170 IJ=1,NRED
      DO 150 IQ=1,IMAXI
      STOR(IQ)=C(IJ,IQ)
  150 C(IJ,IQ)=0.0
      DO 160 IK=1,NSUBI1
      SUM=0.0
      DO 155 IL=1,NSUBI1
```

```
      SUM=SUM+(STOR(IL)*S(IL,IK))
  155 CONTINUE
      C(IJ,IK)=-1.0*SUM
  160 CONTINUE
  170 CONTINUE
      DO 7200 IJ=1,NRED
      DO 190 IK=1,NRED
      SUM=0.0
      DO 180 IL=1,NSUBI1
      SUM=SUM+(C(IJ,IL)*B(IL,IK))
  180 CONTINUE
      IRR=IJ
      ILL=IK
      QUANT=SUM
      CALL PLACE
  190 CONTINUE
 7200 CONTINUE
  620 CONTINUE
      CALL MODIFY
      DO 210 I=1,NRED
      DO 205 J=1,MC
      IF(ICOL(I,J).EQ.0) GO TO 207
  205 CONTINUE
      WRITE(6,206)
  206 FORMAT(1H1//40X,'THE COLUMN DIMENSION OF THE MATRICES (A),(ICOL) I
     XS TOO SMALL'/40X,'INCREASE THE VARIABLE  NNN')
      STOP
  207 IBANDW(I)=J-1
  210 CONTINUE
 4003 CONST=-0.5*DRN*DENS
      NUNVP=NUNV+1
      DO 1620 I=1,NBLD
      IF(I.GT.1) GO TO 570
      NSUBB1=NSUBB(KTOT)
      J1=IBOUND(KTOT,NSUBB1)-NUNV
      J2=J1+NUNV
      DO 540 J=J1,J2
      INDX1=IBANDW(J)+1
      INDX2=IBANDW(J)+2
      IF(J.EQ.J1.OR.J.EQ.J2) GO TO 530
      A(J,INDX1)=CONST
      ICOL(J,INDX1)=NRED+J-J1
      A(J,INDX2)=CONST
      ICOL(J,INDX2)=ICOL(J,INDX1)+1
      GO TO 540
  530 IF(J.EQ.J2) GO TO 535
      A(J,INDX1)=CONST
      ICOL(J,INDX1)=NRED+1
      GO TO 540
  535 A(J,INDX1)=CONST
      ICOL(J,INDX1)=NRED+NUNV
  540 CONTINUE
      K1=IBOUND(2,1)+ISTEP+NCRD-2
```

```
      K2=K1+NUNV
      DO 560 K=K1,K2
      INDX1=IBANDW(K)+1
      INDX2=IBANDW(K)+2
      IF(K.EQ.K1.OR.K.EQ.K2) GO TO 550
      A(K,INDX1)=-1.0*CONST
      ICOL(K,INDX1)=NRED+K-K1
      A(K,INDX2)=-1.0*CONST
      ICOL(K,INDX2)=ICOL(K,INDX1)+1
      GO TO 560
  550 IF(K.EQ.K2) GO TO 555
      A(K,INDX1)=-1.0*CONST
      ICOL(K,INDX1)=NRED+1
      GO TO 560
  555 A(K,INDX1)=-1.0*CONST
      ICOL(K,INDX1)=NRED+NUNV
  560 CONTINUE
      GO TO 1620
  570 NSUBB1=NSUBB(I)
      J1=IBOUND(I,NSUBB1)-NUNV
      J2=J1+NUNV
      DO 590 J=J1,J2
      DO 572 IJ=1,MC
      IF(ICOL(J,IJ).EQ.0) GO TO 573
  572 CONTINUE
      WRITE(6,206)
      STOP
  573 INDX1=IJ
      INDX2=IJ+1
      IF(J.EQ.J1.OR.J.EQ.J2) GO TO 580
      A(J,INDX1)=CONST
      ICOL(J,INDX1)=NRED+((I-1)*NUNV*2)+J-J1
      A(J,INDX2)=CONST
      ICOL(J,INDX2)=ICOL(J,INDX1)+1
      GO TO 590
  580 IF(J.EQ.J2) GO TO 585
      A(J,INDX1)=CONST
      ICOL(J,INDX1)=NRED+((I-1)*NUNV*2)+1
      GO TO 590
  585 A(J,INDX1)=CONST
      ICOL(J,INDX1)=NRED+((I-1)*NUNV*2)+NUNV
  590 CONTINUE
      K1=J1+NCRD+NUNV+2
      K2=K1+NUNV
      DO 610 K=K1,K2
      DO 592 IJ=1,MC
      IF(ICOL(K,IJ).EQ.0) GO TO 595
  592 CONTINUE
      WRITE(6,206)
      STOP
  595 INDX1=IJ
      INDX2=IJ+1
      IF(K.EQ.K1.OR.K.EQ.K2) GO TO 600
```

```
      A(K,INDX1)=-1.0*CONST
      ICOL(K,INDX1)=NRED+((I-1)*NUNV*2)+K-K1
      A(K,INDX2)=-1.0*CONST
      ICOL(K,INDX2)=ICOL(K,INDX1)+1
      GO TO 610
  600 IF(K.EQ.K2) GO TO 605
      A(K,INDX1)=-1.0*CONST
      ICOL(K,INDX1)=NRED+((I-1)*NUNV*2)+1
      GO TO 610
  605 A(K,INDX1)=-1.0*CONST
      ICOL(K,INDX1)=NRED+((I-1)*NUNV*2)+NUNV
  610 CONTINUE
 1620 CONTINUE
      DO 660 I=1,NBLD
      IF(I.GT.1) GO TO 640
      NSUBB1=NSUBB(KTOT)
      J1=IBOUND(KTOT,NSUBB1)-NUNV
      K1=IBOUND(2,1)+ISTEP+NCRD-2
      DO 630 J=1,NUNV
      IFR1=J1+J-1
      IFR2=IFR1+1
      IRER1=K1+J-1
      IRER2=IRER1+1
      IRO1=NRED+J
      IRO2=IRO1+NUNV
      A(IRO1,1)=1.0
      ICOL(IRO1,1)=IFR1
      A(IRO1,2)=-1.0
      ICOL(IRO1,2)=IFR2
      A(IRO1,3)=1.0
      ICOL(IRO1,3)=NRED+NUNV+J
      A(IRO2,1)=1.0
      ICOL(IRO2,1)=IRER1
      A(IRO2,2)=-1.0
      ICOL(IRO2,2)=IRER2
      A(IRO2,3)=1.0
      ICOL(IRO2,3)=ICOL(IRO1,3)
  630 CONTINUE
      GO TO 660
  640 NSUBB1=NSUBB(I)
      J1=IBOUND(I,NSUBB1)+NCRD+2
      K1=IBOUND(I,NSUBB1)-NUNV
      DO 650 J=1,NUNV
      IFR1=J1+J-1
      IFR2=IFR1+1
      IRER1=K1+J-1
      IRER2=IRER1+1
      IRO1=NRED+((I-1)*NUNV*2)+J
      IRO2=IRO1+NUNV
      A(IRO1,1)=1.0
      ICOL(IRO1,1)=IFR1
      A(IRO1,2)=-1.0
      ICOL(IRO1,2)=IFR2
```

```
      A(IRO1,3)=1.0
      ICOL(IRO1,3)=NRED+((I-1)*NUNV*2)+NUNV+J
      A(IRO2,1)=1.0
      ICOL(IRO2,1)=IRER1
      A(IRO2,2)=-1.0
      ICOL(IRO2,2)=IRER2
      A(IRO2,3)=1.0
      ICOL(IRO2,3)=ICOL(IRO1,3)
  650 CONTINUE
  660 CONTINUE
 9000 RETURN
      END
```

```
      SUBROUTINE INTRNL(K)
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LGQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICOMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTCKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/ADL/NU
      COMMON/GAW/ICTT
      COMMON/EEW/IQD
      COMMON/NAL/QUANT,IRR,ILL
      COMMON/FLT/IRAD
C
C
C  EVALUATE THE CONTRIBUTION SUBMATRICES CORRESPONDING TO THE
C  INTERIOR NODES OF THE K(TH) SUBSTRUCTURE
C
C
      JSUB=NSUBI(K)
      LSUB=NSUBB(K)
      DO 430 JJ=1,JSUB
      I=INTERN(K,JJ)
      DO 421 J=1,NTOT
      ILD1=0
      ILD2=0
      IF(I.NE.L(J)) GO TO 220
      ID=1
      DO 210 MM=1,LSUB
      IF(M(J).NE.IBOUND(K,MM)) GO TO 130
      ILD1=1
  130 IF(N(J).NE.IBOUND(K,MM)) GO TO 210
      ILD2=1
  210 CONTINUE
  140 IF(ILD1.EQ.0.AND.ILD2.EQ.0) GO TO 150
      GO TO 160
  150 ID=1
      ICO1=I-INTERN(K,1)+1
      ICO2=M(J)-INTERN(K,1)+1
```

```
      ICO3=N(J)-INTERN(K,1)+1
      IRO=ICO1
      CALL CONTRB(J)
      S(IRO,ICO1)=S(IRO,ICO1)+BB(1,1)
      S(IRO,ICO2)=S(IRO,ICO2)+BB(1,2)
      S(IRO,ICO3)=S(IRO,ICO3)+BB(1,3)
      GO TO 420
  160 IF(ILD1.EQ.1.AND.ILD2.EQ.1) GO TO 170
      GO TO 180
  170 ID=1
      ICO1=I-INTERN(K,1)+1
      ICO2=M(J)
      ICO3=N(J)
      IRO=ICO1
      CALL CONTRB(J)
      S(IRO,ICO1)=S(IRO,ICO1)+BB(1,1)
      B(IRO,ICO2)=B(IRO,ICO2)+BB(1,2)
      B(IRO,ICO3)=B(IRO,ICO3)+BB(1,3)
      GO TO 420
  180 IF(ILD2.EQ.1) GO TO 190
      GO TO 200
  190 ID=1
      ICO1=I-INTERN(K,1)+1
      ICO2=M(J)-INTERN(K,1)+1
      ICO3=N(J)
      IRO=ICO1
      CALL CONTRB(J)
      S(IRO,ICO1)=S(IRO,ICO1)+BB(1,1)
      S(IRO,ICO2)=S(IRO,ICO2)+BB(1,2)
      B(IRO,ICO3)=B(IRO,ICO3)+BB(1,3)
      GO TO 420
  200 ID=1
      ICO1=I-INTERN(K,1)+1
      ICO2=M(J)
      ICO3=N(J)-INTERN(K,1)+1
      IRO=ICO1
      CALL CONTRB(J)
      S(IRO,ICO1)=S(IRO,ICO1)+BB(1,1)
      S(IRO,ICO3)=S(IRO,ICO3)+BB(1,3)
      B(IRO,ICO2)=B(IRO,ICO2)+BB(1,2)
      GO TO 420
  220 IF(I.NE.M(J)) GO TO 320
      ID=2
      DO 310 MM=1,LSUB
      IF(L(J).NE.IBOUND(K,MM)) GO TO 230
      ILD1=1
  230 IF(N(J).NE.IBOUND(K,MM)) GO TO 310
      ILD2=1
  310 CONTINUE
  240 IF(ILD1.EQ.0.AND.ILD2.EQ.0) GO TO 250
      GO TO 260
  250 ID=2
      ICO1=L(J)-INTERN(K,1)+1
```

```
      ICO2=I-INTERN(K,1)+1
      ICO3=N(J)-INTERN(K,1)+1
      IRO=ICO2
      CALL CONTRB(J)
      S(IRO,ICO1)=S(IRO,ICO1)+BB(2,1)
      S(IRO,ICO2)=S(IRO,ICO2)+BB(2,2)
      S(IRO,ICO3)=S(IRO,ICO3)+BB(2,3)
      GO TO 420
  260 IF(ILD1.EQ.1.AND.ILD2.EQ.1) GO TO 270
      GO TO 280
  270 ID=2
      ICO1=L(J)
      ICO2=I-INTERN(K,1)+1
      ICO3=N(J)
      IRO=ICO2
      CALL CONTRB(J)
      S(IRO,ICO2)=S(IRO,ICO2)+BB(2,2)
      B(IRO,ICO1)=B(IRO,ICO1)+BB(2,1)
      B(IRO,ICO3)=B(IRO,ICO3)+BB(2,3)
      GO TO 420
  280 IF(ILD2.EQ.1) GO TO 290
      GO TO 300
  290 ID=2
      ICO1=L(J)-INTERN(K,1)+1
      ICO2=I-INTERN(K,1)+1
      ICO3=N(J)
      IRO=ICO2
      CALL CONTRB(J)
      S(IRO,ICO1)=S(IRO,ICO1)+BB(2,1)
      S(IRO,ICO2)=S(IRO,ICO2)+BB(2,2)
      B(IRO,ICO3)=B(IRO,ICO3)+BB(2,3)
      GO TO 420
  300 ID=2
      ICO1=L(J)
      ICO2=I-INTERN(K,1)+1
      ICO3=N(J)-INTERN(K,1)+1
      IRO=ICO2
      CALL CONTRB(J)
      S(IRO,ICO2)=S(IRO,ICO2)+BB(2,2)
      S(IRO,ICO3)=S(IRO,ICO3)+BB(2,3)
      B(IRO,ICO1)=B(IRO,ICO1)+BB(2,1)
      GO TO 420
  320 IF(I.NE.N(J)) GO TO 421
      ID=3
      DO 410 MM=1,LSUB
      IF(L(J).NE.IBOUND(K,MM)) GO TO 330
      ILD1=1
  330 IF(M(J).NE.IBOUND(K,MM)) GO TO 410
      ILD2=1
  410 CONTINUE
  340 IF(ILD1.EQ.0.AND.ILD2.EQ.0) GO TO 350
      GO TO 360
  350 ID=3
```

```
      ICO1=L(J)-INTERN(K,1)+1
      ICO2=M(J)-INTERN(K,1)+1
      ICO3=I-INTERN(K,1)+1
      IRO=ICO3
      CALL CONTRB(J)
      S(IRO,ICO1)=S(IRO,ICO1)+BB(3,1)
      S(IRO,ICO2)=S(IRO,ICO2)+BB(3,2)
      S(IRO,ICO3)=S(IRO,ICO3)+BB(3,3)
      GO TO 420
  360 IF(ILD1.EQ.1.AND.ILD2.EQ.1) GO TO 370
      GO TO 380
  370 ID=3
      ICO1=L(J)
      ICO2=M(J)
      ICO3=I-INTERN(K,1)+1
      IRO=ICO3
      CALL CONTRB(J)
      S(IRO,ICO3)=S(IRO,ICO3)+BB(3,3)
      B(IRO,ICO1)=B(IRO,ICO1)+BB(3,1)
      B(IRO,ICO2)=B(IRO,ICO2)+BB(3,2)
      GO TO 420
  380 IF(ILD2.EQ.1) GO TO 390
      GO TO 400
  390 ID=3
      ICO1=L(J)-INTERN(K,1)+1
      ICO2=M(J)
      ICO3=I-INTERN(K,1)+1
      IRO=ICO3
      CALL CONTRB(J)
      S(IRO,ICO1)=S(IRO,ICO1)+BB(3,1)
      S(IRO,ICO3)=S(IRO,ICO3)+BB(3,3)
      B(IRO,ICO2)=B(IRO,ICO2)+BB(3,2)
      GO TO 420
  400 ID=3
      ICO1=L(J)
      ICO2=M(J)-INTERN(K,1)+1
      ICO3=I-INTERN(K,1)+1
      IRO=ICO3
      CALL CONTRB(J)
      S(IRO,ICO2)=S(IRO,ICO2)+BB(3,2)
      S(IRO,ICO3)=S(IRO,ICO3)+BB(3,3)
      B(IRO,ICO1)=B(IRO,ICO1)+BB(3,1)
  420 CONTINUE
  421 CONTINUE
  430 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE RECIP(RB,NSUBI1)
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      DIMENSION RB(NSUBI1,NSUBI1)
C
C
C   INVERT THE MATRIX S(I,J)
C
C
      DO 200 IJ=1,NSUBI1
      DO 100 IK=1,NSUBI1
      RB(IJ,IK)=S(IJ,IK)
  100 CONTINUE
  200 CONTINUE
      CALL MINV(RB,NSUBI1,AF,LQQ,MQQ)
      DO 400 IJ=1,NSUBI1
      DO 300 IK=1,NSUBI1
      S(IJ,IK)=RB(IJ,IK)
  300 CONTINUE
  400 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PLACE
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/MNL/LMT(78),LNT(78)
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,CRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTCKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHC(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/ADL/NU
      COMMON/DEL/IDUM,NR,NC
      COMMON/SHO/LZ,MZ,NZ
      COMMON/MAX/PIVROW,PIVCOL,OPROW,IY
      COMMON/GAW/ICTT
      COMMON/EEW/IQD
      COMMON/MAR/TEM(457),ILG(457)
      COMMON/RTH/AAA(457),IBANDW(457),NPIV(457),NNCOL(457),NPT,NNN
      COMMON/NAL/QUANT,IRR,ILL
      COMMON/FLT/IRAD
C
C
C   PLACE THE ELEMENTS OF A LOCAL STIFFNESS MATRIX IN THE PROPER PLACES
C   IN THE PARTIALLY PACKED ARRAYS IN WHICH THE GLOBAL REDUCED STIFFNESS
C   MATRIX IS TO BE STORED
C
C
      IF(QUANT.EQ.0.0) GO TO 90
      DO 10 J=1,MC
      IF(ICOL(IRR,J).EQ.ILL) GO TO 20
      IF(ICOL(IRR,J).EQ.0) GO TO 30
   10 CONTINUE
      WRITE(6,1000) IRR,ILL,QUANT
 1000 FORMAT(//10X,'IRR=',I5,2X,'ILL=',I5,2X,'QUANT=',E12.5)
      WRITE(6,100)
  100 FORMAT(1H1//40X,'MC IS TOO SMALL')
      STOP
   20 A(IRR,J)=A(IRR,J)+QUANT
      GO TO 90
   30 A(IRR,J)=QUANT
      ICOL(IRR,J)=ILL
   90 RETURN
      END
```

```
      SUBROUTINE MODIFY
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/MNL/LMT(78),LNT(78)
      COMMON/SMA/XY(817),P(457),STOR(33),ICCL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTCT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTCTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTCKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/ADL/NU
      COMMON/DEL/IDUM,NR,NC
      COMMON/SHO/LZ,MZ,NZ
      COMMON/MAX/PIVROW,PIVCOL,OPROW,IY
      COMMON/GAW/ICTT
      COMMON/EEW/IQD
      COMMON/MAR/TEM(457),ILG(457)
      COMMON/RTH/AAA(457),IBANDW(457),NPIV(457),NNCOL(457),NPT,NNN
      COMMON/NAL/QUANT,IRR,ILL
      COMMON/FLT/IRAD
C
C
C   REARRANGE THE CONSTRUCTION OF THE PARTIALLY PACKED ARRAYS
C   IN SUCH A WAY THAT THE NONZERO ELEMENTS IN EACH ROW COME FIRST
C
C
      DO 100 I=1,NRED
      DO 40 IJ=1,MC
      TEM(IJ)=0.0
      ILG(IJ)=0
   40 CONTINUE
      ICD=0
      DO 80 J=1,NRED
      DO 60 K=1,MC
      IF(ICOL(I,K).EQ.0) GC TC 60
      IF(ICOL(I,K).NE.J) GO TO 60
      ICD=ICD+1
      TEM(ICD)=A(I,K)
      ILG(ICD)=ICOL(I,K)
   60 CONTINUE
   80 CONTINUE
   70 DO 75 IJ=1,MC
      A(I,IJ)=TEM(IJ)
      ICOL(I,IJ)=ILG(IJ)
   75 CONTINUE
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE EQSOLV
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/MNL/LMT(78),LNT(78)
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTCTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTOKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/BOL/AA(3,3),BB(3,3),RHC(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/ADL/NU
      COMMON/DEL/IDUM,NR,NC
      COMMON/SHO/LZ,MZ,NZ
      COMMON/MAX/PIVROW,PIVCOL,OPROW,IY
      COMMON/GAW/ICTT
      COMMON/EEW/IQD
      COMMON/MAR/TEM(457),ILG(457)
      COMMON/RTH/AAA(457),IBANDW(457),NPIV(457),NNCOL(457),NPT,NNN
      COMMON/FLT/IRAD
C
C
C   THIS SUBROUTINE FOLLOWS THE SAME COMPUTATIONAL PROCEDURE
C   AS IN REFERENCE (16)
C
C
C
C
C   S O L V E   T H E   R E D U C E D   S Y S T E M   O F   E Q U A T I O N S
C
C
C
C
C   INITIALIZE THE BAND WIDTH COUNTER
C
C
      IF(ABS(ZTEST).GT.0.0001) ZTEST=0.0
      NSTOP=0
      MAXWID=0
      DO 1 I=1,NPT
      IBANDW(I)=0
    1 CONTINUE
```

```
      DO 5 I=1,NPT
      DO 2 J=1,NNN
      NC=ICOL(I,J)
      IF(NC.EQ.0) GC TO 3
      IBANDW(I)=J
    2 CONTINUE
    3 IF(MAXWID.LT.J) MAXWID=J
      IF(J.NE.1) GO TO 5
      WRITE(6,4)I
    4 FORMAT('ALL ELEMENTS IN ROW',I5,'ARE=0.0,PROGRAM IS STOPPED')
      NSTOP=1
      GO TO 9988
    5 CONTINUE
 9988 IF(NSTOP.EQ.1) STOP
      NPT1=NPT-1
      DO 23 LL=1,NPT1
C
C
C  FINDING THE ROW WITH MINIMUM BANDWIDTH
C
C
      KK=100000
      DO 6 I=LL,NPT
      IC=IBANDW(I)
      IF(IC.LE.0) GC TO 6
      IF(IC.GE.KK) GO TO 6
      MINROW=I
      KK=IC
    6 CONTINUE
C
C
C  INTERCHANGE ROWS -- MINROW WITH LL
C
C
      LM=IBANDW(LL)
      M=MINROW
      DO 7 I=1,LM
      NNCOL(I)=ICOL(M,I)
      ICOL(M,I)=ICOL(LL,I)
      AAA(I)=A(M,I)
      A(M,I)=A(LL,I)
    7 CONTINUE
      SAVE=P(LL)
      P(LL)=P(M)
      P(M)=SAVE
      IBANDW(LL)=IBANDW(M)
      IBANDW(M)=LM
C
C
C  FINDING BIG A IN MINROW
C
C
      NC=IBANDW(LL)
```

```
      NNCOL(NC+1)=0
      MINROW=LL
      A1=0.0
      I=MINROW
      DO 8 J=1,NC
      AAB=ABS(AAA(J))
      IF(AAB.LT.A1) GO TO 8
      A1=AAB
      IY=J
    8 CONTINUE
      MAXCOL=NNCOL(IY)
      NPIV(LL)=MAXCOL
C
C
C   NORMALIZE THE MINRCW
C
C
      XX=AAA(IY)
      IF(ABS(XX).LT.0.000001) WRITE(6,8701)LL,MINROW,NC,IY,XX,NNCOL(IY)
 8701 FORMAT(2X,'AT ELEM. STEP NO.',I5,'ROW NO.',I5,'HAVING MIN ENTRIES
     XOF',I5/2X,'THE ELEM NO.',I5,' IN THIS ROW IS',E12.5,'AND IT IS IN C
     XOL NO.',I5)
      DO 9 J=1,NC
      AAA(J)=AAA(J)/XX
      ICOL(MINROW,J)=NNCOL(J)
    9 A(MINROW,J)=AAA(J)
      P(MINROW)=P(MINROW)/XX
      A(MINROW,IY)=1.0
      AAA(IY)=1.0
C
C
C   FINDING THE ROWS WHICH CONTAIN MAXCOL
C
C
      LL1=LL+1
      DO 22 I=LL1,NPT
      IF(IBANDW(I).EQ.0) GO TO 22
      NC=IBANDW(I)
      DO 21 J=1,NC
      IF(ICOL(I,J)-MAXCOL)21,10,22
C
C
C   IF ICOL(I,J) IS EQUAL TO MAXCOL THEN ROW CONTAINS THE VARIABLE
C   NOPROW=THE ROW BEING OPERATED ON
C
C
   10 NOPROW=I
      JKOP=1
      JKPI=1
      CW=-A(NOPROW,J)
      P(NOPROW)=P(MINROW)*CW+P(NOPROW)
   11 CONTINUE
      IF(NNCOL(JKPI).EQ.0) GO TO 22
```

```
      IF(ICOL(NOPROW,JKOP).EQ.0) GO TO 12
      IF(NNCOL(JKPI)-ICOL(NOPROW,JKOP))12,14,20
   12 IBANDW(I)=IBANDW(I)+1
      IF(MAXWID.LT.IBANDW(I)) MAXWID=IBANDW(I)
      IF(MAXWID.GT.NNN) GO TO 31
      II=IBANDW(I)
      JKL=JKOP+1
   13 IX=II-1
      A(NOPROW,II)=A(NOPROW,IX)
      ICOL(NOPROW,II)=ICOL(NOPROW,IX)
      II=IX
      IF(IX.GE.JKL) GO TO 13
      A(NOPROW,JKOP)=AAA(JKPI)*CW
      ICOL(NOPROW,JKOP)=NNCOL(JKPI)
      IX=ICOL(NOPROW,JKOP)
      GO TO 19
C
C
C  MINROW AND THE ROW BEING CONSIDERED CONTAIN THIS ELEMENT.
C  SHIFTING OF BOTH ROWS IS DONE AND NOPROW IS OPERATED
C
C
   14 IX=ICOL(NOPROW,JKOP)
      IF(IX.EQ.MAXCOL) GO TO 15
      XX=AAA(JKPI)*CW+A(NOPROW,JKOP)
      A(NOPROW,JKOP)=XX
C
C
C  TESTING TO SEE IF ANY OTHER ELEMENTS WERE ELEMINATED OTHER THAN
C  MAXCOL IN THE NOPROW
C
C
      ATEST=ABS(XX)-ZTEST
      IF(ATEST.GT.0.0) GO TO 19
   15 IBANDW(NOPROW)=IBANDW(NOPROW)-1
      IF(IBANDW(NOPROW))16,16,17
   16 WRITE(6,29)MINROW,MAXCOL,NOPROW
      STOP
   17 IX=IBANDW(NOPROW)
      DO 18 NK=JKOP,IX
      A(I,NK)=A(I,NK+1)
   18 ICOL(I,NK)=ICOL(I,NK+1)
      IX=IX+1
      ICOL(I,IX)=0
      A(I,IX)=0.0
      JKPI=JKPI+1
      GO TO 11
   19 JKPI=JKPI+1
C
C
C  MINROW DOES NOT CONTAIN THIS ELEMENT SHIFT NOPROW AND CONTINUE
C
C
```

```
   20 JKOP=JKOP+1
      GO TO 11
   21 CONTINUE
   22 CONTINUE
   23 CONTINUE
C
C
C  ELEMINATE MINROW AND MAXCOL FROM BEING CONSIDERED AGAIN
C
C
      NPIV(NPT)=ICOL(NPT,1)
      WRITE(6,35)MAXWID
C
C
C  BACK SUBSTITUTION
C
C
      DO 24 I=1,NPT
   24 AAA(I)=0.0
      DO 27 I=1,NPT
      II=NPT-I+1
      LM=IBANDW(II)
      NP=NPIV(II)
      IF(NP.EQ.0) GO TO 27
      DO 26 J=1,LM
      NN=ICOL(II,J)
      IF(NN.EQ.NP) GO TO 25
      P(II)=P(II)-AAA(NN)*A(II,J)
      GO TO 26
   25 IJ=J
   26 CONTINUE
      AAA(NP)=P(II)/A(II,IJ)
   27 CONTINUE
C
C
C  STORE THE SOLUTION IN THE FIRST COLUMN OF THE MATRIX (A)
C
C
      DO 28 I=1,NPT
      A(I,1)=AAA(I)
   28 CONTINUE
      GO TO 33
   29 FORMAT(//'MATRIX IS SINGULAR'/'MINROW=',I5,'MAXCOL=',I5,'ROW OPERA
     XTED',I5//)
   31 WRITE(6,32)I,LL
      STOP
   32 FORMAT('COLUMN DIMENSIONS OF NCOL AND A EXCEEDED IN ROW',I5
     X,'WHILE THE NUMBER OF ROWS OPERATED WERE',I5
     X///10X,'INCREASE THE VALUE OF THE VARIABLE (NNN) IN THE INPUT DATA
     X'/10X,'AND ACCORDINGLY THE COLUMN DIMENSION OF THE MATRICES (A) &
     X(ICOL)')
   33 RETURN
   35 FORMAT(1H1////10X,'NOTE'/10X,4('*')//14X,'THE VALUE OF THE INPUT V
     XARIABLE (NNN) IS SUFFICIENT FOR THE SOLUTION PROCESS'
     X//14X,'THE ACTUAL VALUE NEEDED IS',I5)
      END
```

```
      SUBROUTINE SOLVE
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/MNL/LMT(78),LNT(78)
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTCTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTCKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHO(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/SHO/LZ,MZ,NZ
      COMMON/MAR/TEM(457),ILG(457)
      COMMON/RTH/AAA(457),IBANDW(457),NPIV(457),NNCOL(457),NPT,NNN
      COMMON/NAL/QUANT,IRR,ILL
      COMMON/FLT/IRAD
      DIMENSION RB(33,33)
C
C
C   KNOWING THE VELOCITY POTENTIAL VALUES AT THE EXTERNAL NODES,
C   CALCULATE THE VALUES ASSOCIATED WITH THE INTERNAL NODES INSIDE
C   EACH SUBDOMAIN
C
C
      DO 19 I=1,NRED
   19 P(I)=A(I,1)
      DO 80 I=1,KTOT
      DO 20 IJ=1,IMAXI
      DO 10 IK=1,IMAXI
      S(IJ,IK)=0.0
   10 CONTINUE
   20 CONTINUE
      DO 40 IJ=1,IMAXI
      DO 30 IK=1,NRED
      B(IJ,IK)=0.0
   30 CONTINUE
   40 CONTINUE
      K=I
      CALL INTRNL(K)
      NSUBI1=NSUBI(K)
      NUM=NSUBI1
```

```
      CALL RECIP(RB,NSUBI1)
 1500 DO 41  IJ=1,NUM
      DO 241  IQ=1,NRED
      TEM(IQ)=B(IJ,IQ)
  241 CONTINUE
      SUM=0.0
      DO 242  IL=1,NRED
      SUM=SUM+(TEM(IL)*P(IL))
  242 CONTINUE
      B(IJ,1)=SUM
   41 CONTINUE
      DO 50  IJ=1,IMAXI
   50 STOR(IJ)=0.0
      DO 51  IJ=1,NUM
      SUM=0.0
      DO 53  IL=1,NUM
      SUM=SUM+(S(IJ,IL)*B(IL,1))
   53 CONTINUE
      STOR(IJ)=SUM
   51 CONTINUE
      DO 60  IJ=1,NUM
   60 STOR(IJ)=-1.0*STOR(IJ)
C   PLACE THE NODAL VALUES OFTHE VELOCITY POTENTIAL  INSIDE THIS PARTICULAR
C   SUBDOMAIN IN THE PROPER PLACES OF THE GLOBAL COLUMN VECTOR FAI(I)
C
C
      DO 70  IJ=1,NUM
      IJJ=INTERN(K,IJ)
   70 FAI(IJJ)=STOR(IJ)
   80 CONTINUE
C
C
C   PLACE THE NODAL VALUES OF THE VELOCITY POTENTIAL ALONG THE DIFFERENT
C   BOUNDARIES IN THEIR CORRESPONDING LOCATIONS IN THE GLOBAL VECTOR
C
C
      DO 90  IJ=1,NRED
   90 FAI(IJ)=P(IJ)
      RETURN
      END
```

```
      SUBROUTINE PHYS
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LGQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTCT,NRED,IMAXB,IMAXI,NREQ
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICOMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTOT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTCKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHC(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/EZZ/U1(1812),V1(1812)
      COMMON/FLT/IRAD
      COMMON/ACM/PRCP(817,7)
C
C
C   THIS SUBROUTINE COMPUTES THE FLOW PHYSICAL AND THERMODYNAMIC
C   PROPERTIES ,NAMELY THE VELOCITY,TEMPERATURE,MACH NUMBER,PRESSURE
C   AND DENSITY
C
C
C
C
C   TRANSFORM THE SYSTEM TOPOLOGY BACK TO THE UNSUBSTRUCTURED MODEL
C
C
      DO 1003 I=1,MTOT
 1003 XY(I)=0.0
      DO 1004 I=1,MTOT
      IMGG=IMG(I)
 1004 XY(I)=X(IMGG)
      DO 1005 I=1,MTOT
 1005 X(I)=XY(I)
      DO 1006 I=1,MTOT
 1006 XY(I)=0.0
      DO 1007 I=1,MTOT
      IMGG=IMG(I)
 1007 XY(I)=Y(IMGG)
      DO 1008 I=1,MTOT
 1008 Y(I)=XY(I)
      DO 1009 I=1,MTOT
 1009 XY(I)=0.0
```

```
      DO 1010 I=1,MTOT
      IMGG=IMG(I)
 1010 XY(I)=FAI(IMGG)
      DO 1011 I=1,MTOT
 1011 FAI(I)=XY(I)
      DO 11 I=1,NTOT
      L1=L(I)
      M1=M(I)
      N1=N(I)
      DO 4 J=1,MTOT
      IF(L1.EQ.IMG(J)) GO TO 5
    4 CONTINUE
    5 L(I)=J
      DO 6 J=1,MTOT
      IF(M1.EQ.IMG(J)) GO TO 7
    6 CONTINUE
    7 M(I)=J
      DO 8 J=1,MTOT
      IF(N1.EQ.IMG(J)) GO TO 9
    8 CONTINUE
    9 N(I)=J
   11 CONTINUE
C
C
C  CALCULATE THE AVERAGED FLOW PROPERTIES AT EACH NODE
C
C
      IK=IRAD
      DO 10 I=1,MTOT
   10 FAI(I)=FAI(I)*STWDTH*VN1
      WRITE(6,600)
      DO 437 I=1,MTCT
  437 WRITE(6,610)I,FAI(I)
      DO 13 I=1,MTOT
      DO 12 J=1,7
      PROP(I,J)=0.0
   12 CONTINUE
   13 CONTINUE
      WRITE(6,601)
      DO 30 I=1,NTOT
      LA=L(I)
      MA=M(I)
      NA=N(I)
      B1=Y(MA)-Y(NA)
      B2=Y(NA)-Y(LA)
      B3=Y(LA)-Y(MA)
      C1=X(NA)-X(MA)
      C2=X(LA)-X(NA)
      C3=X(MA)-X(LA)
      AR=0.5*((X(MA)*Y(NA))-(X(NA)*Y(MA))+(X(NA)*Y(LA))-(X(LA)*Y(NA))+
     X(X(LA)*Y(MA))-(X(MA)*Y(LA)))
      AR=ABS(AR)
      U1(I)=((B1*FAI(LA))+(B2*FAI(MA))+(B3*FAI(NA)))/(2.0*AR)
```

```
      V1(I)=((C1*FAI(LA))+(C2*FAI(MA))+(C3*FAI(NA)))/(2.0*AR)
      U2=ABS(U1(I))
      V2=ABS(V1(I))
      V=((U2**2.0)+(V2**2.0))**0.5
      IF(U1(I).GT.0.000.AND.V1(I).GT.0.000) GO TO 40
      IF(U1(I).LT.0.000.AND.V1(I).GT.0.000) GO TO 50
      IF(U1(I).LT.0.000.AND.V1(I).LT.0.000) GO TO 60
      IF(U1(I).GT.0.000.AND.V1(I).LT.0.000) GO TO 70
      IF(U1(I).EQ.0.000.AND.V1(I).GT.0.000) GO TO 80
      IF(U1(I).EQ.0.000.AND.V1(I).LT.0.000) GO TO 90
      IF(U1(I).GT.0.000.AND.V1(I).EQ.0.000) GO TO 100
      IF(U1(I).LT.0.000.AND.V1(I).EQ.0.000) GO TO 110
      IF(U1(I).EQ.0.000.AND.V1(I).EQ.0.000) GO TO 120
   40 ANGLE=(ATAN(V2/U2))*57.296
      GO TO 130
   50 ANGLE=(3.142-(ATAN(V2/U2)))*57.296
      GO TO 130
   60 ANGLE=(3.142+(ATAN(V2/U2)))*57.296
      GO TO 130
   70 ANGLE=(6.283-(ATAN(V2/U2)))*57.296
      GO TO 130
   80 ANGLE=90.0
      GO TO 130
   90 ANGLE=270.0
      GO TO 130
  100 ANGLE=0.0
      GO TO 130
  110 ANGLE=180.0
      GO TO 130
  120 ANGLE=0.0
  130 TLOC=TSTG-(((V**2.0)*(GAMA-1.0))/(2.0*GAMA*R))
      AMAK=V/((GAMA*R*TLOC)**0.5)
      POWER=1.0/(GAMA-1.0)
      RHO(I)=DENS/(1.000+(((GAMA-1.000)/2.000)*(AMAK**2.000)))**POWER
      QLOC=RHO(I)*R*TLOC
      TLOC=TLOC/TSTG
      DENST=RHO(I)/DENS
      QLOC=QLOC/PSTG
      AMAK=V/ABS(VN1)
      PROP(LA,1)=PROP(LA,1)+U1(I)
      PROP(MA,1)=PROP(MA,1)+U1(I)
      PROP(NA,1)=PROP(NA,1)+U1(I)
      PROP(LA,2)=PROP(LA,2)+AMAK
      PROP(MA,2)=PROP(MA,2)+AMAK
      PROP(NA,2)=PROP(NA,2)+AMAK
      PROP(LA,3)=PROP(LA,3)+V1(I)
      PROP(MA,3)=PROP(MA,3)+V1(I)
      PROP(NA,3)=PROP(NA,3)+V1(I)
      PROP(LA,4)=PROP(LA,4)+TLOC
      PROP(MA,4)=PROP(MA,4)+TLOC
      PROP(NA,4)=PROP(NA,4)+TLOC
      PROP(LA,5)=PROP(LA,5)+QLOC
      PROP(MA,5)=PROP(MA,5)+QLOC
```

```
      PROP(NA,5)=PROP(NA,5)+QLOC
      PROP(LA,6)=PROP(LA,6)+DENST
      PROP(MA,6)=PROP(MA,6)+DENST
      PROP(NA,6)=PROP(NA,6)+DENST
      PROP(LA,7)=PROP(LA,7)+1.0
      PROP(MA,7)=PROP(MA,7)+1.0
      PROP(NA,7)=PROP(NA,7)+1.0
      WRITE(6,8202)I,L(I),M(I),N(I),V,AMAK,ANGLE,TLOC,QLOC,DENST
   30 CONTINUE
      WRITE(6,630)
      DO 350 I=1,MTOT
      DO 340 J=1,6
      PROP(I,J)=PROP(I,J)/PROP(I,7)
  340 CONTINUE
      U1(I)=PROP(I,1)
      V1(I)=PROP(I,2)
      U2=ABS(U1(I))
      V2=ABS(V1(I))
      IF(U1(I).GT.0.000.AND.V1(I).GT.0.000) GO TO 8040
      IF(U1(I).LT.0.000.AND.V1(I).GT.0.000) GO TO 8050
      IF(U1(I).LT.0.000.AND.V1(I).LT.0.000) GO TO 8060
      IF(U1(I).GT.0.000.AND.V1(I).LT.0.000) GO TO 8070
      IF(U1(I).EQ.0.000.AND.V1(I).GT.0.000) GO TO 8080
      IF(U1(I).EQ.0.000.AND.V1(I).LT.0.000) GO TO 8090
      IF(U1(I).GT.0.000.AND.V1(I).EQ.0.000) GO TO 8100
      IF(U1(I).LT.0.000.AND.V1(I).EQ.0.000) GO TO 8110
      IF(U1(I).EQ.0.000.AND.V1(I).EQ.0.000) GO TO 8120
 8040 ANGLE=(ATAN(V2/U2))*57.296
      GO TO 8130
 8050 ANGLE=(3.142-(ATAN(V2/U2)))*57.296
      GO TO 8130
 8060 ANGLE=(3.142+(ATAN(V2/U2)))*57.296
      GO TO 8130
 8070 ANGLE=(6.283-(ATAN(V2/U2)))*57.296
      GO TO 8130
 8080 ANGLE=90.0
      GO TO 8130
 8090 ANGLE=270.0
      GO TO 8130
 8100 ANGLE=0.0
      GO TO 8130
 8110 ANGLE=180.0
      GO TO 8130
 8120 ANGLE=0.0
 8130 CONTINUE
      UU1=U1(I)
      VV1=V1(I)
      PROP(I,1)=((UU1*UU1)+(VV1*VV1))**0.5
      PROP(I,3)=ANGLE
      WRITE(6,620)I,(PROP(I,K),K=1,6)
  350 CONTINUE
  600 FORMAT(1H1//45X,'VELOCITY POTENTIAL DISTRIBUTION'/
     X45X,31('*')//46X,'NODE',5X,'POTENTIAL'/)
```

```
 610 FORMAT(45X,I5,4X,E12.5)
 601 FORMAT(1H1//39X,'FLOW PHYSICAL AND THERMODYNAMIC PROPERTIES'
    X/39X,42('*')/53X,'INSIDE ELEMENTS'/53X,15('*')///
    X3X,'ELEM. NO.',13X,'NODAL NUMBERS',10X,'V MAGNITUDE',2X,'V/INLET V
    XEL.',4X,'VEL. ANGLE',3X,'TEMPERATURE',7X,'DENSITY',6X,'PRESSURE'
    X/22X,'OF ELEMENT VERTICES'//)
8202 FORMAT(6X,I5,8X,3(1X,I5),8X,6(2X,E12.5))
 630 FORMAT(1H1//39X,'FLOW PHYSICAL AND THERMODYNAMIC PROPERTIES'/39X
    X,42('*')/50X,'AVERAGED AT EACH NODE'/50X,21('*')///24X,'NODE'
    X,3X,'V MAGNITUDE',2X,'V/INLET VEL.',4X,'VEL. ANGLE',3X,'TEMPERATUR
    XE',7X,'DENSITY',6X,'PRESSURE'//)
 620 FORMAT(23X,I5,6(2X,E12.5))
 140 RETURN
     END
```

```
      SUBROUTINE MASFLW
      REAL*8 X,Y,AA,BB
      COMMON/WAG/A(457,80),S(33,33),B(33,379),C(379,33),INTERN(14,33)
     X,IBOUND(14,33),LQQ(33),MQQ(33),IMG(817),ID
      COMMON/PAR/KTOT,NRED,IMAXB,IMAXI,NREQ
      COMMON/SMA/XY(817),P(457),STOR(33),ICOL(457,80)
      COMMON/MIL/R,GAMA,PSTG,TSTG,STWDTH,STRTL,VN1,VN2
      COMMON/ESM/WDTH(100),RI,RO,DRN,DANG,DTHET,DENS
      COMMON/NWL/ICCMP,IFORM,NITR,NPRNT,IPR,NBLD,KT
      COMMON/WAD/IK,ISTEP,ISTRT,JSTRT,NSTPS,IDN
      COMMON/ZAK/THET(9),BET(9),GAMM(5)
      COMMON/TRZ/MTCT,NTOT,LTOT,NBW,NBW1,NBWP
      COMMON/SPC/X(817),Y(817)
      COMMON/LOC/IST,NW,IL,KMM,IU1,IST2
      COMMON/ALT/IEL,ITA,IIN,LIN,IN1,IN2,IU2
      COMMON/HAP/IDN1,IM1,IM2,IA,IB,MATCH,MATCH1
      COMMON/ALF/MTOTP,MTOTM,MR,MC,JELE,JELIM,JCOL
      COMMON/BAT/ZTEST,IMA,IMB,MTOT1,MTOT2,MRM,IMM,MAA
      COMMON/GAM/MTOKM,ITER,MTM,LBC(84),MBC(84)
      COMMON/TAG/L(1812),M(1812),N(1812)
      COMMON/BOL/AA(3,3),BB(3,3),RHC(1812),VN(84),FAI(1246),EPS
      COMMON/SIA/NSUBB(14),NSUBI(14)
      COMMON/EZZ/U1(1812),V1(1812)
      COMMON/FLT/IRAD
      COMMON/ACM/PROP(817,7)
C
C
C  CALCULATE THE FLOW DISCHARGE THROUGH EACH OF THE NOZZLE CHANNELS
C
C
      WRITE(6,170)
      JDN=IDN+1
      JDN=1
      IN3=IN1+2
      IK=IRAD
      IL1=IK-2
      IST3=IST+IU1+1
      KI=IST3
      RK=RO
      DO 10 I=1,NBLD
      FLW=0.0
      NDSCT=0
      IF(I.NE.1) KI=MK2+IN1
      DO 20 J=1,IL1
      K1=KI+((J-1)*IN1)
      IF(I.NE.1.AND.J.NE.1) K1=KI+((J-1)*IN1)+1
      K2=K1+IN1
      IF(I.NE.1.AND.J.EQ.1) K2=K2+1
      MK2=K2
      DO 30 K=1,NTOT
      IF(L(K).EQ.KI.AND.N(K).EQ.K2) GO TO 40
   30 CONTINUE
   40 KJ=K
```

```
      XC=(X(K1)+X(K2))/2.0
      YC=(Y(K1)+Y(K2))/2.0
      DDX=X(K1)-X(K2)
      DDY=Y(K1)-Y(K2)
      DDL=((DDX*DDX)+(DDY*DDY))**0.5
      XC=XC/RO
      YC=YC/RO
      DISG=-1.0*DDL*((XC*U1(KJ))+(YC*V1(KJ)))
      FLW=FLW+DISG
   20 CONTINUE
      FAI(I)=FLW
   10 CONTINUE
      SUM=0.0
      DO 1009 I=1,NBLD
      SUM=SUM+FAI(I)
 1009 CONTINUE
      DO 1010 I=1,NBLD
      FAI(I)=FAI(I)/SUM
      FAI(I)=NBLD*FAI(I)
      WRITE(6,175)I,FAI(I)
 1010 CONTINUE
  170 FORMAT(1H1//35X,'RATIO OF CALCULATED TO THEORETICAL FLOW DISCHARGE
     X'/35X,49('*')/43X,'THROUGH EACH OF THE NOZZLE CHANNELS'/43X,35('*'
     X)//43X,'NOZZLE CHANNEL',3X,'FLOW DISCHARGE'/)
  175 FORMAT(54X,I5,3X,E12.5)
      RETURN
      END
```

Input Data

It is relatively easy to prepare the program input which consists of only a few cards. The input data is classified and described in Table A1, with the corresponding formats in the correct order. Figures A1 and A2, illustrate the various input parameters, and Table A2 provides one set of sample input data in the appropriate units.

Classification	Type of Input Data	Input Variables	Format	Number of Cards
1	The stator unit operating conditions	R, GAMA, PSTG, TSTG, VN1	(5F10.0)	1
2	The flow domain major dimensions	STWDTH, STRTL, RO, RI	(4F10.0)	1
3	Finite element model discretization parameters	NBLD, ISTRT, ISTEP, IDN, NUNV, IK	(6I5)	1
4	Nozzle vane geometry	[THET(I), I=1, IDN+1]	(8F10.0)	[IDN+1)/8]
5	Nozzle vane geometry	[GAMM(I), I=1, IDN]	(8F10.0)	[IDN/8]
6	Column dimensions of the partially packed arrays [A] and [ICOL]	NNN	(I5)	1

TABLE A1. DESCRIPTION OF THE INPUT DATA.

Card						
1	R: 287.0	GAMA: 1.40	PSTG: 150000.0	TSTG: 289.0	VN1: 32.3	
2	STWDTH: 0.1875	STRTL: 0.104	RØ: 0.0885	RI: 0.0725		
3	NBLD: 13	ISTRT: 6	ISTEP: 5	IDN: 5	NUNV: 4	IK: 5
4	THET(1): 1.5708	THET(2): 1.7191	THET(3): 1.8326	THET(4): 1.9687	THET(5): 2.1328	
5	GAMM(1): 0.034907	GAMM(2): 0.10646	GAMM(3): 0.087266	GAMM(4): 0.045379		
6	NNN: 80					

TABLE A2. SAMPLE INPUT DATA.

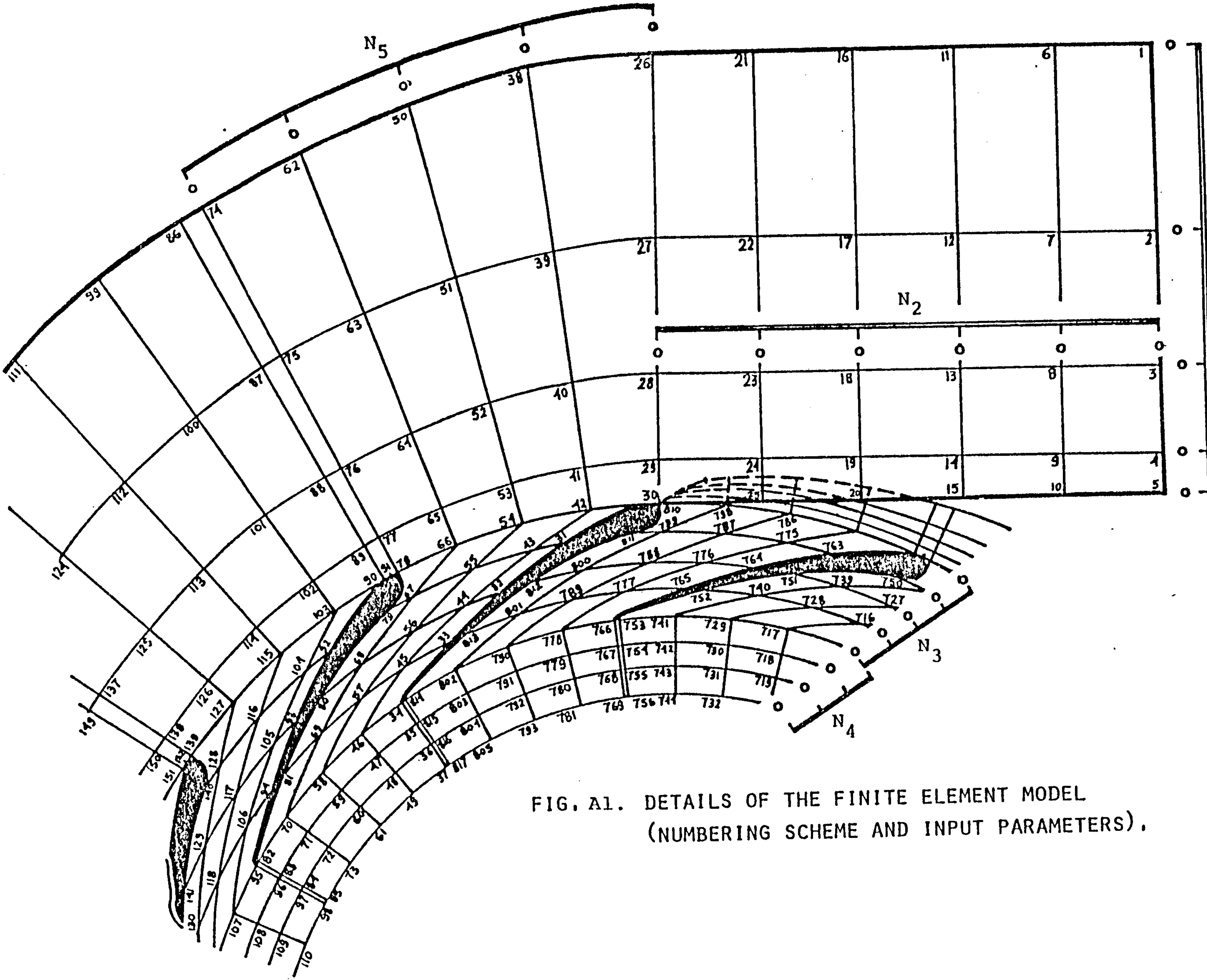


FIG. A1. DETAILS OF THE FINITE ELEMENT MODEL (NUMBERING SCHEME AND INPUT PARAMETERS).

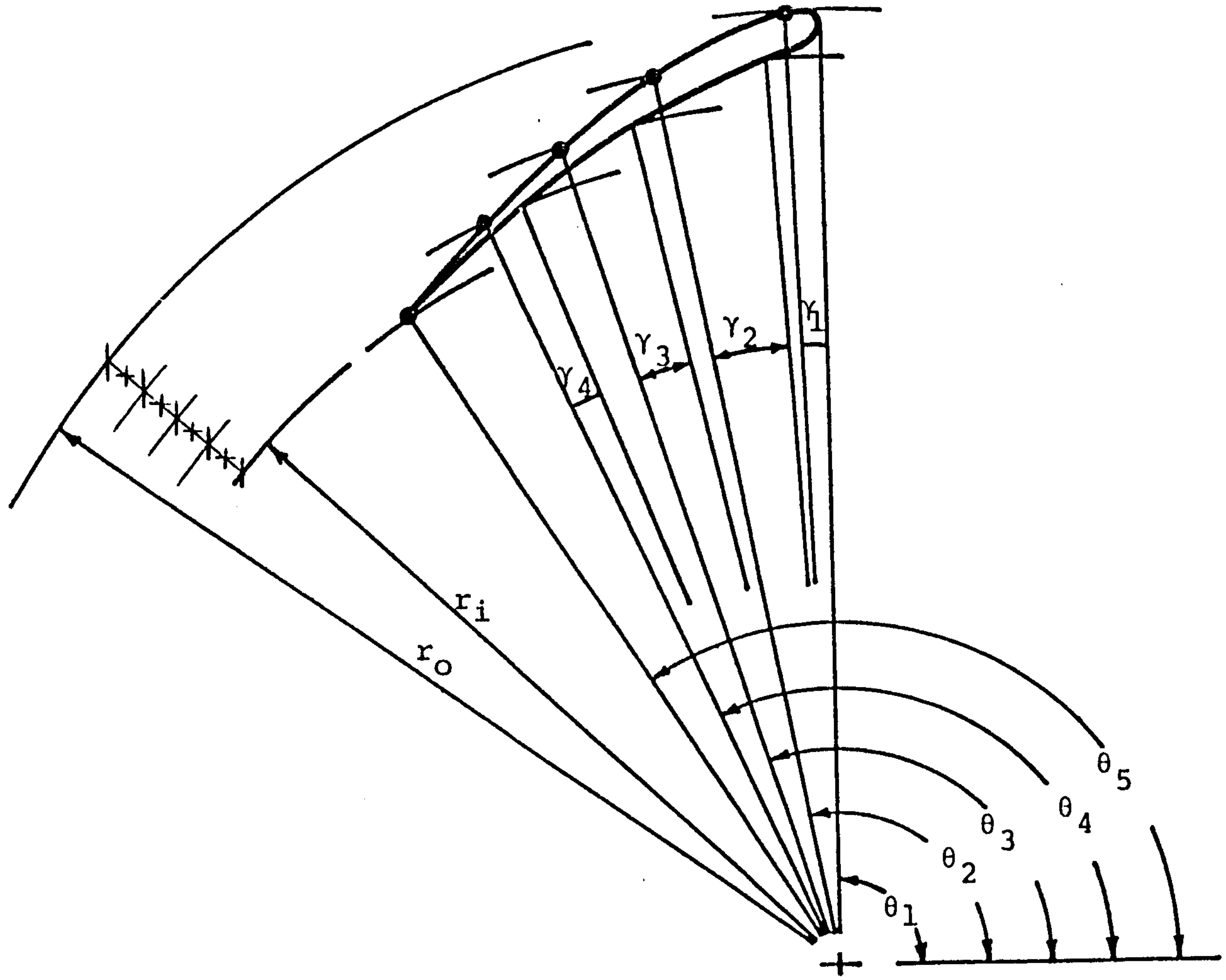


FIG. A2. THE VANED NOZZLE REGION INPUT PARAMETER.

The Program Output:

A sample output is given which corresponds to the input data of Table A2. Only the first few lines of the various output sections are reproduced, since the complete output is very long. These sections have been numbered to correspond to the following description:

(1) The first section is a list of the input data as it appears in Table A2.

(2) Calculated program constants which include the variables to be used in discretizing the solution domain.

(3) A listing of the array WDTH which is calculated in subroutine DATA.

(4) The arrays X and Y containing the coordinates of the nodal points (only parts of these arrays are shown).

(5) This note gives the value of the variable NNN which was actually needed to solve the final set of equations.

(6) A listing of the potential discrete values (only part of the array is shown).

(7) A listing of the flow physical and thermodynamic properties inside the finite elements which includes the velocity magnitude (mt/sec), the nondimensional velocity (normalized with respect to the stator inlet velocity), the velocity angle in degrees (the inlet velocity V_1 in Fig. 1 has an angle of 180°), followed by the thermodynamic properties (temperature, density and pressure) all normalized with regard to the stagnation values.

(8) The same flow properties as in (7) averaged at the nodes.

(9) A listing of the mass flow rate through each blade-to-blade channel (nondimensionalized with regard to the mean value).

1

```
INPUT DATA
**********

           R           GAMA           PSTG           TSTG            VNI
 0.28700E+03    0.14000E+01    0.15000E+06    0.28900E+03    0.32300E+02

      STWDTH          STRTL             RC             RI
 0.18750E+00    0.10400E+00    0.88500E-01    0.72500E-01

 NBLD   ISTRT   ISTEP    IDN    NJNV    IK
   13       6       5      5       4     5


ARRAY THET:  0.15708E+01  0.17191E+01  0.18326E+01  0.19687E+01  0.21328E+01
ARRAY GAMM:  0.34907E-01  0.10646E+00  0.87266E-01  0.45379E-01


  NNN
   80


E N D   O F   I N P U T   D A T A
```

2

```
      PROGRAM PARAMETERS AS CALCULATED FROM THE INPUT DATA
      ****************************************************

      MTCT=  817
      NTOT= 1292
      LTOT=   56
      KTOT=   14
      NRED=  379
      NPT=  457
      IMAXB=   33
      IMAXI=   33


NOTE:
*****.
      AT THIS STEP , IT IS IMPORTANT TO COMPARE THE PRECEEDING VALUES WITH THE DIMENSIONS OF THE ARRAYS IN THE COMMON STATEMENTS
      THE LATTER SHOULD BE OF GREATER OR , AT LEAST , EQUAL MAGNITUDES
```

ARRAY OF THE SCROLL PASSAGE WIDTH
****
AT 65 STATIONS
****

3

STATION	WIDTH
1	0.18750E+00
2	0.18415E+00
3	0.18081E+00
4	0.17746E+00
5	0.17412E+00
6	0.17308E+00
7	0.16973E+00
8	0.16639E+00
9	0.16304E+00
10	0.15970E+00
11	0.15865E+00
12	0.15531E+00
13	0.15196E+00
14	0.14862E+00
15	0.14527E+00
16	0.14423E+00
17	0.14089E+00
18	0.13754E+00
19	0.13419E+00
20	0.13085E+00
21	0.12981E+00
22	0.12646E+00
23	0.12312E+00
24	0.11977E+00
25	0.11643E+00
26	0.11538E+00
27	0.11204E+00
28	0.10869E+00
29	0.10535E+00
30	0.10200E+00
31	0.10096E+00
32	0.97616E-01
33	0.94271E-01
34	0.90926E-01
35	0.87580E-01
36	0.86539E-01
37	0.83193E-01
38	0.79848E-01
39	0.76503E-01
40	0.73157E-01
41	0.72116E-01
42	0.68770E-01
43	0.65425E-01
44	0.62080E-01
45	0.58734E-01
46	0.57693E-01
47	0.54347E-01
48	0.51002E-01
49	0.47657E-01
50	0.44311E-01
51	0.43270E-01
52	0.39924E-01
53	0.36579E-01
54	0.33234E-01
55	0.29888E-01
56	0.28847E-01
57	0.25501E-01
58	0.22156E-01
59	0.18811E-01
60	0.15465E-01
61	0.14424E-01
62	0.11078E-01
63	0.77329E-02
64	0.43875E-02
65	0.16000E-02

NODAL COORDINATES
****

4

NODE	X-COORD	Y-COORD
1	0.10400D+00	0.27600D+00
2	0.10400D+00	0.20100D+00
3	0.10400D+00	0.14475D+00
4	0.10400D+00	0.10725D+00
5	0.10400D+00	0.88500D-01
6	0.83200D-01	0.27600D+00
7	0.83200D-01	0.20100D+00
8	0.83200D-01	0.14475D+00
9	0.83200D-01	0.10725D+00
10	0.83200D-01	0.88500D-01
11	0.62400D-01	0.27600D+00
12	0.62400D-01	0.20100D+00
13	0.62400D-01	0.14475D+00
14	0.62400D-01	0.10725D+00
15	0.62400D-01	0.88500D-01
16	0.41600D-01	0.27600D+00
17	0.41600D-01	0.20100D+00
18	0.41600D-01	0.14475D+00
19	0.41600D-01	0.10725D+00
20	0.41600D-01	0.88500D-01
21	0.20800D-01	0.27600D+00
22	0.20800D-01	0.20100D+00
23	0.20800D-01	0.14475D+00
24	0.20800D-01	0.10725D+00
25	0.20800D-01	0.88500D-01
26	0.34986D-06	0.27600D+00
27	0.25479D-06	0.20100D+00
28	0.18349D-06	0.14475D+00
29	0.13595D-06	0.10725D+00
30	0.11218D-06	0.88500D-01
31	-0.12486D-01	0.93572D-01
32	-0.20835D-01	0.77757D-01
33	-0.29643D-01	0.70523D-01
34	-0.38634D-01	0.61349D-01
35	-0.36502D-01	0.57964D-01
36	-0.34371D-01	0.54579D-01
37	-0.32239D-01	0.51194D-01
38	-0.30501D-01	0.27094D+00
39	-0.22261D-01	0.19774D+00
40	-0.16081D-01	0.14284D+00
41	-0.11960D-01	0.10624D+00
42	-0.99003D-02	0.87945D-01
43	-0.20293D-01	0.82027D-01
44	-0.28420D-01	0.75316D-01
45	-0.37171D-01	0.66862D-01
46	-0.45747D-01	0.56245D-01
47	-0.43223D-01	0.53141D-01
48	-0.40699D-01	0.50038D-01
49	-0.38175D-01	0.46935D-01
50	-0.59876D-01	0.26257D+00
51	-0.43796D-01	0.19206D+00
52	-0.31736D-01	0.13917D+00
53	-0.23696D-01	0.10391D+00
54	-0.19676D-01	0.86285D-01
55	-0.27919D-01	0.79754D-01
56	-0.35726D-01	0.72138D-01
57	-0.44254D-01	0.62400D-01
58	-0.52193D-01	0.50320D-01
59	-0.49313D-01	0.47544D-01
60	-0.46434D-01	0.44768D-01

5

```
NOTE
****

    THE VALUE OF THE INPUT VARIABLE (NNN) IS SUFFICIENT FOR THE SOLUTION PROCESS

    THE ACTUAL VALUE NEEDED IS   76
```

6

VELOCITY POTENTIAL DISTRIBUTION
****

NODE	POTENTIAL
1	-0.16148E+02
2	-0.15974E+02
3	-0.15751E+02
4	-0.15654E+02
5	-0.15641E+02
6	-0.15489E+02
7	-0.15308E+02
8	-0.15072E+02
9	-0.14960E+02
10	-0.14954E+02
11	-0.14859E+02
12	-0.14653E+02
13	-0.14381E+02
14	-0.14250E+02
15	-0.14231E+02
16	-0.14260E+02
17	-0.14012E+02
18	-0.13678E+02
19	-0.13493E+02
20	-0.13463E+02
21	-0.13699E+02
22	-0.13388E+02
23	-0.12965E+02
24	-0.12686E+02
25	-0.12621E+02
26	-0.13187E+02
27	-0.12786E+02
28	-0.12251E+02
29	-0.11818E+02
30	-0.11617E+02
31	-0.10825E+02
32	-0.10246E+02
33	-0.96655E+01
34	-0.90357E+01
35	-0.89924E+01
36	-0.89400E+01
37	-0.88801E+01
38	-0.12473E+02
39	-0.12144E+02
40	-0.11689E+02
41	-0.11298E+02
42	-0.11067E+02
43	-0.10404E+02
44	-0.98636E+01
45	-0.92850E+01
46	-0.85643E+01
47	-0.85060E+01
48	-0.84465E+01
49	-0.83845E+01
50	-0.11802E+02
51	-0.11520E+02
52	-0.11126E+02
53	-0.10773E+02
54	-0.10552E+02
55	-0.10013E+02
56	-0.95059E+01
57	-0.88832E+01
58	-0.80696E+01
59	-0.80095E+01
60	-0.79486E+01

FLOW PHYSICAL AND THERMODYNAMIC PROPERTIES
****
INSIDE ELEMENTS
****

ELEM. NO.	NODAL NUMBERS OF ELEMENT VERTICES			V MAGNITUDE	V/INLET VEL.	VEL. ANGLE	TEMPERATURE	DENSITY	PRESSURE
1	1	6	2	0.31742E+02	0.98274E+00	0.18421E+03	0.99826E+00	0.99394E+00	0.99567E+00
2	2	6	7	0.32131E+02	0.99476E+00	0.18435E+03	0.99822E+00	0.99379E+00	0.99556E+00
3	2	7	3	0.32284E+02	0.99952E+00	0.18709E+03	0.99820E+00	0.99373E+00	0.99552E+00
4	3	7	8	0.32877E+02	0.10179E+01	0.18733E+03	0.99814E+00	0.99350E+00	0.99535E+00
5	3	8	4	0.32712E+02	0.10127E+01	0.18454E+03	0.99816E+00	0.99356E+00	0.99540E+00
6	4	8	9	0.33117E+02	0.10253E+01	0.18486E+03	0.99811E+00	0.99340E+00	0.99529E+00
7	4	9	5	0.33006E+02	0.10219E+01	0.18121E+03	0.99812E+00	0.99345E+00	0.99532E+00
8	5	9	10	0.33066E+02	0.10237E+01	0.18132E+03	0.99812E+00	0.99342E+00	0.99530E+00
9	6	11	7	0.30411E+02	0.94151E+00	0.18459E+03	0.99841E+00	0.99444E+00	0.99602E+00
10	7	11	12	0.31601E+02	0.97835E+00	0.18501E+03	0.99828E+00	0.99399E+00	0.99571E+00
11	7	12	8	0.31757E+02	0.98320E+00	0.18759E+03	0.99826E+00	0.99393E+00	0.99566E+00
12	8	12	13	0.33576E+02	0.10395E+01	0.18829E+03	0.99806E+00	0.99322E+00	0.99515E+00
13	8	13	9	0.33344E+02	0.10323E+01	0.18482E+03	0.99808E+00	0.99331E+00	0.99522E+00
14	9	13	14	0.34686E+02	0.10739E+01	0.18582E+03	0.99793E+00	0.99277E+00	0.99483E+00
15	9	14	10	0.34517E+02	0.10686E+01	0.18127E+03	0.99795E+00	0.99284E+00	0.99488E+00
16	10	14	15	0.34739E+02	0.10755E+01	0.18165E+03	0.99792E+00	0.99274E+00	0.99481E+00
17	11	16	12	0.28921E+02	0.89538E+00	0.18547E+03	0.99856E+00	0.99497E+00	0.99640E+00
18	12	16	17	0.30997E+02	0.95967E+00	0.18615E+03	0.99835E+00	0.99422E+00	0.99587E+00
19	12	17	13	0.31196E+02	0.96582E+00	0.18893E+03	0.99832E+00	0.99414E+00	0.99582E+00
20	13	17	18	0.34335E+02	0.10630E+01	0.18998E+03	0.99797E+00	0.99291E+00	0.99493E+00
21	13	18	14	0.33999E+02	0.10526E+01	0.18594E+03	0.99801E+00	0.99305E+00	0.99503E+00
22	14	18	19	0.36700E+02	0.11362E+01	0.18772E+03	0.99768E+00	0.99190E+00	0.99421E+00
23	14	19	15	0.36382E+02	0.11264E+01	0.18158E+03	0.99772E+00	0.99204E+00	0.99431E+00
24	15	19	20	0.36964E+02	0.11444E+01	0.18252E+03	0.99765E+00	0.99179E+00	0.99413E+00
25	16	21	17	0.27158E+02	0.84080E+00	0.18702E+03	0.99873E+00	0.99556E+00	0.99683E+00
26	17	21	22	0.30277E+02	0.93736E+00	0.18791E+03	0.99842E+00	0.99448E+00	0.99606E+00
27	17	22	18	0.30572E+02	0.94651E+00	0.19122E+03	0.99839E+00	0.99438E+00	0.99598E+00
28	18	22	23	0.35085E+02	0.10862E+01	0.19240E+03	0.99788E+00	0.99260E+00	0.99471E+00
29	18	23	19	0.34621E+02	0.10719E+01	0.18819E+03	0.99794E+00	0.99279E+00	0.99485E+00
30	19	23	24	0.39524E+02	0.12237E+01	0.19087E+03	0.99731E+00	0.99062E+00	0.99329E+00
31	19	24	20	0.38851E+02	0.12028E+01	0.18240E+03	0.99740E+00	0.99093E+00	0.99351E+00
32	20	24	25	0.40657E+02	0.12587E+01	0.18494E+03	0.99715E+00	0.99007E+00	0.99290E+00
33	21	26	22	0.25001E+02	0.77402E+00	0.18958E+03	0.99892E+00	0.99624E+00	0.99731E+00
34	22	26	27	0.29410E+02	0.91053E+00	0.19048E+03	0.99851E+00	0.99480E+00	0.99628E+00
35	22	27	23	0.29884E+02	0.92520E+00	0.19460E+03	0.99846E+00	0.99463E+00	0.99616E+00
36	23	27	28	0.35599E+02	0.11021E+01	0.19552E+03	0.99782E+00	0.99238E+00	0.99455E+00
37	23	28	24	0.35103E+02	0.10868E+01	0.19226E+03	0.99788E+00	0.99259E+00	0.99470E+00
38	24	28	29	0.43316E+02	0.13411E+01	0.19551E+03	0.99677E+00	0.98873E+00	0.99194E+00
39	24	29	25	0.41889E+02	0.12969E+01	0.18479E+03	0.99698E+00	0.98946E+00	0.99246E+00
40	25	29	30	0.49416E+02	0.15299E+01	0.19252E+03	0.99579E+00	0.98536E+00	0.98952E+00
41	26	38	27	0.23147E+02	0.71664E+00	0.19335E+03	0.99908E+00	0.99677E+00	0.99769E+00
42	27	38	39	0.28751E+02	0.89014E+00	0.19536E+03	0.99858E+00	0.99502E+00	0.99644E+00
43	27	39	28	0.29049E+02	0.89936E+00	0.19913E+03	0.99855E+00	0.99492E+00	0.99637E+00
44	28	39	40	0.35625E+02	0.11029E+01	0.19981E+03	0.99781E+00	0.99237E+00	0.99454E+00
45	28	40	29	0.35516E+02	0.10996E+01	0.19903E+03	0.99783E+00	0.99242E+00	0.99458E+00
46	29	40	41	0.44876E+02	0.13893E+01	0.20014E+03	0.99653E+00	0.98791E+00	0.99135E+00
47	29	41	30	0.43861E+02	0.13579E+01	0.19414E+03	0.99669E+00	0.98845E+00	0.99174E+00
48	30	41	42	0.57615E+02	0.17837E+01	0.19901E+03	0.99428E+00	0.98013E+00	0.98577E+00
49	30	42	31	0.59021E+02	0.18273E+01	0.20328E+03	0.99400E+00	0.97916E+00	0.98507E+00
50	31	42	43	0.55533E+02	0.17193E+01	0.20855E+03	0.99469E+00	0.98153E+00	0.98677E+00
51	31	43	32	0.56991E+02	0.17644E+01	0.21276E+03	0.99441E+00	0.98056E+00	0.98608E+00
52	32	43	44	0.51236E+02	0.15862E+01	0.21839E+03	0.99548E+00	0.98426E+00	0.98874E+00
53	32	44	33	0.50953E+02	0.15775E+01	0.21752E+03	0.99553E+00	0.98444E+00	0.98886E+00
54	33	44	45	0.47558E+02	0.14724E+01	0.22307E+03	0.99610E+00	0.98643E+00	0.99029E+00
55	33	45	34	0.49077E+02	0.15194E+01	0.22812E+03	0.99585E+00	0.98555E+00	0.98966E+00
56	34	45	46	0.53944E+02	0.16701E+01	0.21929E+03	0.99499E+00	0.98257E+00	0.98752E+00
57	34	46	35	0.54366E+02	0.16831E+01	0.22368E+03	0.99491E+00	0.98230E+00	0.98732E+00
58	35	46	47	0.61555E+02	0.19057E+01	0.23285E+03	0.99347E+00	0.97735E+00	0.98377E+00
59	35	47	36	0.59589E+02	0.18448E+01	0.22495E+03	0.99388E+00	0.97876E+00	0.98478E+00

7

FLOW PHYSICAL AND THERMODYNAMIC PROPERTIES
****
AVERAGED AT EACH NODE
****

8

NODE	V MAGNITUDE	V/INLET VEL.	VEL. ANGLE	TEMPERATURE	DENSITY	PRESSURE
1	0.31742E+02	0.98274E+00	0.18421E+03	0.99826E+00	0.99394E+00	0.99567E+00
2	0.32044E+02	0.99234E+00	0.18522E+03	0.99823E+00	0.99382E+00	0.99558E+00
3	0.32616E+02	0.10100E+01	0.18632E+03	0.99817E+00	0.99360E+00	0.99542E+00
4	0.32931E+02	0.10200E+01	0.18353E+03	0.99813E+00	0.99347E+00	0.99533E+00
5	0.33036E+02	0.10228E+01	0.18127E+03	0.99812E+00	0.99344E+00	0.99531E+00
6	0.31428E+02	0.97300E+00	0.18438E+03	0.99830E+00	0.99406E+00	0.99575E+00
7	0.31834E+02	0.98587E+00	0.18601E+03	0.99825E+00	0.99390E+00	0.99564E+00
8	0.32885E+02	0.10185E+01	0.18624E+03	0.99813E+00	0.99349E+00	0.99535E+00
9	0.33603E+02	0.10410E+01	0.18323E+03	0.99805E+00	0.99320E+00	0.99514E+00
10	0.34107E+02	0.10560E+01	0.18142E+03	0.99799E+00	0.99300E+00	0.99500E+00
11	0.30310E+02	0.93841E+00	0.18502E+03	0.99841E+00	0.99447E+00	0.99604E+00
12	0.31331E+02	0.97032E+00	0.18694E+03	0.99830E+00	0.99408E+00	0.99577E+00
13	0.33505E+02	0.10378E+01	0.18728E+03	0.99806E+00	0.99323E+00	0.99516E+00
14	0.35135E+02	0.10889E+01	0.18401E+03	0.99787E+00	0.99256E+00	0.99468E+00
15	0.36028E+02	0.11154E+01	0.18193E+03	0.99776E+00	0.99219E+00	0.99442E+00
16	0.29024E+02	0.89862E+00	0.18620E+03	0.99854E+00	0.99492E+00	0.99637E+00
17	0.30742E+02	0.95219E+00	0.18859E+03	0.99836E+00	0.99428E+00	0.99591E+00
18	0.34194E+02	0.10594E+01	0.18920E+03	0.99798E+00	0.99294E+00	0.99495E+00
19	0.37102E+02	0.11509E+01	0.18556E+03	0.99761E+00	0.99168E+00	0.99405E+00
20	0.38816E+02	0.12020E+01	0.18332E+03	0.99740E+00	0.99093E+00	0.99351E+00
21	0.27474E+02	0.85072E+00	0.18812E+03	0.99869E+00	0.99543E+00	0.99673E+00
22	0.30018E+02	0.92997E+00	0.19110E+03	0.99843E+00	0.99452E+00	0.99608E+00
23	0.34939E+02	0.10826E+01	0.19224E+03	0.99788E+00	0.99260E+00	0.99471E+00
24	0.39754E+02	0.12350E+01	0.18847E+03	0.99725E+00	0.99040E+00	0.99313E+00
25	0.43895E+02	0.13618E+01	0.18773E+03	0.99664E+00	0.98830E+00	0.99163E+00
26	0.25943E+02	0.80040E+00	0.19105E+03	0.99884E+00	0.99593E+00	0.99710E+00
27	0.29277E+02	0.90734E+00	0.19480E+03	0.99850E+00	0.99475E+00	0.99625E+00
28	0.35663E+02	0.11053E+01	0.19677E+03	0.99777E+00	0.99224E+00	0.99445E+00
29	0.42985E+02	0.13358E+01	0.19426E+03	0.99676E+00	0.98872E+00	0.99193E+00
30	0.52333E+02	0.16247E+01	0.19767E+03	0.99519E+00	0.98327E+00	0.98802E+00
31	0.57049E+02	0.17703E+01	0.20813E+03	0.99436E+00	0.98042E+00	0.98597E+00
32	0.53009E+02	0.16427E+01	0.21610E+03	0.99514E+00	0.98309E+00	0.98789E+00
33	0.49053E+02	0.15231E+01	0.22283E+03	0.99583E+00	0.98547E+00	0.98960E+00
34	0.52361E+02	0.16242E+01	0.22356E+03	0.99525E+00	0.98347E+00	0.98817E+00
35	0.58354E+02	0.18112E+01	0.22733E+03	0.99409E+00	0.97947E+00	0.98529E+00
36	0.63228E+02	0.19607E+01	0.22770E+03	0.99308E+00	0.97599E+00	0.98279E+00
37	0.67391E+02	0.20891E+01	0.22890E+03	0.99214E+00	0.97276E+00	0.98047E+00
38	0.24608E+02	0.76220E+00	0.19556E+03	0.99894E+00	0.99630E+00	0.99735E+00
39	0.28620E+02	0.88709E+00	0.19949E+03	0.99856E+00	0.99497E+00	0.99640E+00
40	0.35926E+02	0.11131E+01	0.20166E+03	0.99773E+00	0.99209E+00	0.99435E+00
41	0.45239E+02	0.14034E+01	0.20070E+03	0.99639E+00	0.98743E+00	0.99100E+00
42	0.54318E+02	0.16854E+01	0.20496E+03	0.99486E+00	0.98213E+00	0.98720E+00
43	0.53598E+02	0.16619E+01	0.21293E+03	0.99503E+00	0.98270E+00	0.98762E+00
44	0.49172E+02	0.15245E+01	0.21918E+03	0.99582E+00	0.98544E+00	0.98958E+00
45	0.49674E+02	0.15472E+01	0.22541E+03	0.99569E+00	0.98498E+00	0.98925E+00
46	0.55917E+02	0.17412E+01	0.22945E+03	0.99454E+00	0.98101E+00	0.98640E+00
47	0.61280E+02	0.19027E+01	0.23287E+03	0.99348E+00	0.97738E+00	0.98379E+00
48	0.65477E+02	0.20320E+01	0.23254E+03	0.99257E+00	0.97423E+00	0.98152E+00
49	0.68784E+02	0.21326E+01	0.23458E+03	0.99181E+00	0.97164E+00	0.97966E+00
50	0.23770E+02	0.73641E+00	0.20079E+03	0.99901E+00	0.99653E+00	0.99752E+00
51	0.28146E+02	0.87211E+00	0.20451E+03	0.99860E+00	0.99512E+00	0.99651E+00
52	0.35946E+02	0.11137E+01	0.20660E+03	0.99773E+00	0.99207E+00	0.99433E+00
53	0.44960E+02	0.13934E+01	0.20692E+03	0.99646E+00	0.98767E+00	0.99118E+00
54	0.51272E+02	0.15893E+01	0.21055E+03	0.99544E+00	0.98414E+00	0.98865E+00
55	0.49186E+02	0.15237E+01	0.21516E+03	0.99592E+00	0.98545E+00	0.98958E+00
56	0.46633E+02	0.14563E+01	0.22241E+03	0.99618E+00	0.98671E+00	0.99049E+00
57	0.52233E+02	0.16304E+01	0.23255E+03	0.99520E+00	0.98332E+00	0.98806E+00
58	0.58517E+02	0.18176E+01	0.23706E+03	0.99405E+00	0.97935E+00	0.98520E+00
59	0.62661E+02	0.19439E+01	0.23588E+03	0.99320E+00	0.97641E+00	0.98309E+00
60	0.65343E+02	0.20581E+01	0.23944E+03	0.99238E+00	0.97357E+00	0.98105E+00
61	0.69557E+02	0.21563E+01	0.24138E+03	0.99163E+00	0.97102E+00	0.97922E+00
62	0.23204E+02	0.71891E+00	0.20641E+03	0.99905E+00	0.99669E+00	0.99764E+00
63	0.27790E+02	0.86107E+00	0.20989E+03	0.99863E+00	0.99523E+00	0.99659E+00
64	0.35914E+02	0.11128E+01	0.21179E+03	0.99773E+00	0.99208E+00	0.99433E+00
65	0.44545E+02	0.13803E+01	0.21200E+03	0.99654E+00	0.98795E+00	0.99138E+00
66	0.48670E+02	0.15073E+01	0.21261E+03	0.99591E+00	0.98576E+00	0.98981E+00
67	0.43949E+02	0.13819E+01	0.21667E+03	0.99655E+00	0.98798E+00	0.99140E+00
68	0.47683E+02	0.14893E+01	0.23145E+03	0.99593E+00	0.98606E+00	0.99003E+00
69	0.55151E+02	0.17193E+01	0.23814E+03	0.99467E+00	0.98149E+00	0.98674E+00
70	0.60372E+02	0.18737E+01	0.24329E+03	0.99369E+00	0.97807E+00	0.98429E+00
71	0.63666E+02	0.19734E+01	0.24567E+03	0.99300E+00	0.97570E+00	0.98258E+00
72	0.66955E+02	0.20763E+01	0.24580E+03	0.99224E+00	0.97311E+00	0.98072E+00
73	0.69962E+02	0.21681E+01	0.24757E+03	0.99154E+00	0.97071E+00	0.97899E+00
74	0.22773E+02	0.70562E+00	0.21231E+03	0.99939E+00	0.99681E+00	0.99772E+00

9

RATIO OF CALCULATED TO THEORETICAL FLOW DISCHARGE
THROUGH EACH OF THE NOZZLE CHANNELS

NOZZLE CHANNEL	FLOW DISCHARGE
1	0.10779E+01
2	0.10134E+01
3	0.99030E+00
4	0.98297E+00
5	0.98001E+00
6	0.97786E+00
7	0.97623E+00
8	0.97469E+00
9	0.97298E+00
10	0.97311E+00
11	0.97926E+00
12	0.10028E+01
13	0.10984E+01