General Disclaimer

One or more of the Following Statements may affect this Document

- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.

- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational source and is the best copy available.

- This document may contain tone-on-tone or color graphs, charts and/or pictures, which have been reproduced in black and white.

- This document is paginated as submitted by the original source.

- Portions of this document are not fully legible due to the historical nature of some of the material. However, it is the best reproduction available from the original submission.

Produced by the NASA Center for Aerospace Information (CASI)
AN IMPROVED NUMERICAL
PROCEDURE FOR THE PARAMETRIC OPTIMIZATION OF
THREE DIMENSIONAL SCRAMJET NOZZLES (Advanced
Technology Labs., Inc., Westbury, N.Y.)
87 p HC $4.75

Advanced Technology Laboratories inc.
ATL TR 215
AN IMPROVED NUMERICAL PROCEDURE FOR THE
PARAMETRIC OPTIMIZATION OF THREE
DIMENSIONAL SCRAMJET NOZZLES

By
S. Dash and P. Del Guidice

PREPARED FOR
NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
LANGLEY RESEARCH CENTER
HAMPTON, VIRGINIA 23665

UNDER
NASA CONTRACT NO. NAS1-13303

BY
ADVANCED TECHNOLOGY LABORATORIES, INC.
Merrick and Stewart Avenues
Westbury, New York 11590
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I. INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>II. NUMERICAL PROCEDURES</td>
<td>3</td>
</tr>
<tr>
<td>A. IDEAL GAS GRID POINT CALCULATION</td>
<td>3</td>
</tr>
<tr>
<td>B. EQUILIBRIUM FLOW GRID POINT CALCULATION</td>
<td>7</td>
</tr>
<tr>
<td>III. DETERMINATION OF NOZZLE FLOW FIELDS</td>
<td>9</td>
</tr>
<tr>
<td>IV. THRUST, LIFT AND PITCHING MOMENT</td>
<td>16</td>
</tr>
<tr>
<td>V. VISCOUS EFFECTS</td>
<td>20</td>
</tr>
<tr>
<td>VI. CONCLUSIONS</td>
<td>22</td>
</tr>
<tr>
<td>APPENDIX I PROGRAM DESCRIPTION</td>
<td></td>
</tr>
<tr>
<td>APPENDIX II LISTING OF FROZEN FLOW PROGRAM</td>
<td></td>
</tr>
<tr>
<td>APPENDIX III LISTING OF EQUILIBRIUM FLOW PROGRAM</td>
<td></td>
</tr>
<tr>
<td>FIG. 1.</td>
<td>TYPICAL NOZZLE CONFIGURATION</td>
</tr>
<tr>
<td>FIG. 2.</td>
<td>GRID POINT CALCULATION</td>
</tr>
<tr>
<td>FIG. 3.</td>
<td>VARIATION OF VEHICLE EXPANSION WAVE STRENGTH</td>
</tr>
<tr>
<td>FIG. 4.</td>
<td>PARTIAL REFLECTION OF COWL EXPANSION/NO REFLECTION OF VEHICLE EXPANSION</td>
</tr>
<tr>
<td>FIG. 5.</td>
<td>PARTIAL REFLECTION OF COWL AND VEHICLE EXPANSIONS</td>
</tr>
<tr>
<td>FIG. 6.</td>
<td>COMPLETE REFLECTION OF COWL EXPANSION/ PARTIAL REFLECTION OF VEHICLE EXPANSION</td>
</tr>
<tr>
<td>FIG. 7.</td>
<td>PARTIAL REFLECTION VEHICLE EXPANSION/ COMPLETE REFLECTION OF COWL EXPANSION</td>
</tr>
<tr>
<td>FIG. 8.</td>
<td>COMPLETE REFLECTION OF COWL AND VEHICLE EXPANSIONS</td>
</tr>
<tr>
<td>FIG. 9.</td>
<td>INTERACTION OF EXTERNAL FLOW WITH VEHICLE UNDERSURFACE</td>
</tr>
<tr>
<td>FIG. 10.</td>
<td>THRUST VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION ANGLE</td>
</tr>
<tr>
<td>FIG. 11.</td>
<td>LIFT VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION</td>
</tr>
<tr>
<td>FIG. 12.</td>
<td>THRUST, LIFT, MOMENT</td>
</tr>
<tr>
<td>FIG. 13.</td>
<td>ELEMENTAL SURFACE AREA</td>
</tr>
<tr>
<td>FIG. 14.</td>
<td>UPPER OR LOWER SURFACE AREA ELEMENTS</td>
</tr>
<tr>
<td>FIG. 15.</td>
<td>SIDEWALL AREA ELEMENTS</td>
</tr>
</tbody>
</table>
FIG. 16  GEOMETRIC INPUTS

Page
A1-4
TR 215
SECTION I
INTRODUCTION

This report describes a parametric numerical procedure permitting the rapid determination of the performance of a class of scramjet nozzle configurations. The geometric complexity of these configurations rules out attempts to employ conventional nozzle design procedures, Reference (1), wherein properties at the nozzle exit plane are specified and wave cancellation techniques are then employed to design the wall surfaces. It is not feasible to stipulate exit conditions à priori and wave cancellation techniques employing three dimensional characteristics are beyond the current state of the art.

The current approach is an extension of work discussed in Reference (2) and employs a characteristic grid network with Riemann invariants as variables. Lateral expansion effects are incorporated via one dimensional approximations as suggested in Reference (3).

The numerical program developed permits the parametric variation of cowl length, turning angles on the cowl and vehicle undersurface and lateral expansion and is subject to fixed constraints such as the vehicle length and nozzle exit height. The program requires uniform initial conditions at the burner exit station and yields the location of all predominant wave zones, accounting for lateral expansion effects. In addition, the program yields the detailed pressure distribution on the cowl, vehicle undersurface and fences, if any, and calculates the nozzle thrust, lift and pitching moments. Viscous effects are included in the latter via the Spalding-Chi method described in Reference (4). Local heat transfer coefficients are computed from a modified Reynolds' analogy. Local vehicle external flow interaction and/or plume boundary effects are computed insofar as they affect vehicle undersurface pressure distributions.

Due to the differing techniques required for the calculation of ideal gas flows as compared to equilibrium flows, two separate numerical programs have been developed. The first program analyzes constant-γ ideal gas flow fields and a listing of this program is provided in Appendix II. The second
program analyzes equilibrium hydrogen-air flow fields via equilibrium curve fits and its listing is provided in Appendix III. A complete program description is provided in Appendix I.
Consider a typical nozzle configuration as depicted in Figure (1), where the lateral expansion distribution $Z(x)$ may result from a combination of several nozzles merging into a single nozzle. It is assumed in this preliminary analysis that the jets after merging are bounded by sidewalls which extend downstream of the merged section. The initial flow (at the burner exit) is represented as an average uniform flow. The assessment of nonuniformities at the entrance station may be obtained applying the numerical procedure described in Reference (5).

A. Ideal Gas Grid Point Calculation - Consider the calculational procedure required to determine the location and properties of a point 3 (as shown in Figure 2) where properties at 1 and 2 are known and 1-3 and 2-3 are characteristic surfaces. Along these surfaces the Riemann invariants are defined as

$$C_k = v - \theta$$

where $v$ is the Prandtl-Meyer function and $\theta$ the local flow deflections. Then at point 3

$$v_3 = \frac{1}{2} (v_1 + v_2) + \frac{1}{2} (\theta_1 - \theta_2)$$  \quad (2a)

$$\theta_3 = \frac{1}{2} (v_1 - v_2) + \frac{1}{2} (\theta_1 + \theta_2)$$  \quad (2b)

Employing the two dimensional value of expansion $\Delta v_3$ (from initial condition 1) the Mach number $M_3$ is obtained via the Prandtl-Meyer relation (where 1 denotes uniform initial flow properties at the burner exit)

$$\Delta v_3 = v_3 - v_1 = \sqrt{\frac{\gamma + 1}{\gamma - 1}} (\tan^{-1} \sqrt{\frac{\gamma - 1}{\gamma + 1}} (M_3^2 - 1) - \tan^{-1} \sqrt{\frac{\gamma - 1}{\gamma + 1}} (M_1^2 - 1)) - \tan^{-1} \sqrt{M_3^2 - 1} - \tan^{-1} \sqrt{M_1^2 - 1})$$

$$-3-$$
(Note that $\Delta \nu$ is the wave strength of an expansion fan, not the geometric angle between initial and final rays.)
FIGURE 2. GRID POINT CALCULATION

FIGURE 3. VARIATION OF VEHICLE EXPANSION WAVE STRENGTH
employing an iterative procedure to solve this transcendental equation for $M_3$. Then, with the Mach angle determined

$$M_3 = \sin^{-1} \left( \frac{1}{M_3} \right) \quad (4)$$

Equations (7a) and (7b) yield a tentative location for point 3, and the area ratio $(A/A)^*_3$ is calculated based on two dimensional considerations.

$$\frac{A}{A^*_3} = \frac{M_3 \left( 1 + \frac{\gamma-1}{2} \frac{M_3^2}{\gamma+1} \right)}{\left( \frac{\gamma+1}{2} \right)^{\frac{\gamma+1}{\gamma-1}}} \quad (5)$$

This ratio is corrected for lateral expansion by multiplying it by the ratio $Z_3/Z_1$, where the lateral expansion variable is expressed by a suitable polynomial curve fit

$$Z(x) = Ax^2 + Bx + C \quad (6)$$

where $Z_3 = Z(x_3)$ and $Z_1$ denotes the lateral extent of the nozzle at the initial station. The location $(x,y)_3$ is determined from

$$\frac{y_3 - y_{1,2}}{x_3 - x_{1,2}} = \frac{1}{2} \left[ \tan(\theta_{1,2} \pm \mu_{1,2}) + \tan(\theta_3 \pm \mu_3) \right] \quad (7a,b)$$

then

$$\frac{A}{A^*_3} = \frac{A}{A^*_3} \times \frac{Z_3}{Z_1} \quad (8)$$

The three dimensional corrected Mach number is obtained by replacing the two dimensional area ratio in Equation (5) by the three dimensional value given by Equation (8), and solving Equation (5) for $M_{3D}$ by an iterative process.
Equations (7a) and (7b) are resolved using the corrected Mach angle $\mu_{3D}$ and the entire procedure is repeated until two successive values of $x_3$ agree to within a prescribed tolerance.

A similar procedure is used to determine properties at grid points on boundaries with Equation (7a) or (7b) replaced with an equation describing the body geometry. Desired variables (P, T etc.) are then simply obtained by isentropic, constant $\gamma$ expansions from initial conditions.

B. Equilibrium Flow Grid Point Calculation - The geometric location of point 3 is obtained using Equations (7a) and (7b) and properties $(v, \theta)_3$ are obtained using Equations (2a,b) just as for the frozen calculation. The known two dimensional value of expansion $\Delta v_3 = v_3 - v_2$ is subdivided into a series of small $\Delta v_j$ increments. The initial value of isentropic exponent is obtained from

$$\Gamma = \Gamma (P, \phi, h) \quad (9)$$

where Equation (9) has been curve fit for equilibrium hydrogen-air as described in Reference (5).

The characteristic compatibility relation

$$\frac{d \ln P}{\Gamma} + \frac{d v}{\sinh \cos u} = 0 \quad (10)$$

applied across the interval $\Delta v_j$ yields the pressure, holding $u$ and $\Gamma$ equal to their value at the start of the increment. The density is obtained from the isentropic pressure-density relation

$$\frac{P}{\rho \Gamma} = \text{constant} \quad (11)$$
The velocity is obtained from the Bernoulli relation;

\[
\frac{dP}{\rho} + \frac{1}{2} dV^2 = 0 \tag{12}
\]

the enthalpy from the constancy of stagnation enthalpy;

\[ h + \frac{1}{2} V^2 = H = \text{constant} \tag{13} \]

and the Mach number from;

\[ M = \frac{V}{a}; \quad a = \left( \frac{\rho}{\rho} \right)^{\frac{1}{2}} \tag{14} \]

where \( \Gamma \) has been reevaluated employing Equation (9). This procedure is repeated in small steps \( \Delta v \) until the full wave \( \Delta v_3 \) has been integrated. Having the two dimensional value of \( M_3 \), point 3 can be tentatively located employing Equations (7a) and (7b). Then, the two dimensional area ratio can be computed from mass flow considerations

\[ \left( \frac{A_3}{A_1} \right)_{2D} = \frac{\rho_1 V_1}{\rho_3 V_3} \tag{15} \]

Since the effective area based on three dimensional considerations is

\[ \left( \frac{A_3}{A_1} \right)_{3D} = \left( \frac{A_3}{A_1} \right)_{2D} \times \frac{Z_3}{Z_1} \tag{16} \]

the product \( \rho_3 V_3 \) must be divided by \( Z_3/Z_1 \) to conserve mass flow

\[ \left( \frac{\rho_3 V_3}{Z_3} \right)_{3D} = \left( \frac{\rho_3 V_3}{Z_3} \right)_{2D} \times \frac{Z_1}{Z_3} \tag{17} \]

Then an iteration procedure is performed to determine the value of three dimensional expansion \( \Delta v \). The correct value being that which yields \( (\rho_3 V_3)_{3D} \) after application of the integration procedure of Equations (9) thru (14), and an update of the location of point 3 using Equations (7a) and (7b).
A nozzle calculation is performed subject to the following constraints:

1. The initial profile is uniform. For the frozen flow (constant $\gamma$) calculation this requires specification of the pressure $P_i$, flow deflection angle $\theta_i$, Mach number $M_i$, and specific heat ratio $\gamma$. For the equilibrium calculation one must specify $P_i$, $\theta_i$, $M_i$, the temperature $T_i$ and the fuel-air equivalence ratio $\phi_i$.

2. The initial turning at the vehicle undersurface ($\Delta\nu_v$) and cowl ($\Delta\nu_c$) occur via sharp corners as depicted in Figure 1.

3. The wall segments downstream of these sharp corners remain straight until the expansion waves emanating from the cowl and vehicle undersurface reach the walls (points $V_3$ and $C_3$ of Figure 1).

4. The nozzle exit height is specified ($y_{v_2} - y_{c_2}$).

5. The recompression on the vehicle undersurface (between $V_3$ and $V_2$) is parabolic while the cowl between is straight; (i.e., constant slope from $C_1$ to $C_3$).

6. The lateral expansion $Z(x)$ is specified via a geometric curve fit.

7. The cowl length and vehicle length are specified.

8. Local cowl external flow properties are specified.

The numerical logic employed in the parametric design procedure is to treat the cowl length ($x_{c_2} - x_{c_1}$) and the vehicle undersurface expansion $\Delta\nu_v$ as
parametric variables for fixed values of cowl turning $\Delta v_c$, nozzle exit height, lateral expansion and vehicle length. Initially, a short cowl length should be chosen such that the expansion waves from the vehicle expansion fan miss the cowl. For this cowl length, the value of the vehicle undersurface expansion wave is varied in small increments, the minimum amount of turning being that which introduces no recompression in the region $V_3$ to $V_2$ (i.e., the undersurface has no curvature) to a value for which the recompression produces zero deflection at the end of the vehicle. This is illustrated in Figure (3). Then the cowl length is increased in specified increments and the entire procedure is repeated.

For a given nozzle configuration, the calculational procedure is as follows: Point $V_2$ is located and vehicle expansion ($\Delta v_{vmin}$) is determined and segmented into small increments ($\Delta v_i^v$). The cowl expansion array is swept out by segmenting $\Delta v_c$ into small segments ($\Delta v_i^c$), and the interaction of each ray with the vehicle expansion is determined up to the vehicle surface (or exit plane) as discussed in Section II.

After completing the cowl expansion, it must be determined whether the first ray from the vehicle expansion intersects the cowl surface. If not, properties at $C_2$ are determined by inserting a data point $B$ (Figure 4) on the final cowl ray such that the characteristic $BC_2$ intersects point $C_2$. The reflected ray at $C_2$ is computed up to the vehicle surface or until the exit plane is crossed. If the first vehicle ray does intersect the cowl before $C_2$, the intersection is determined and the reflected ray computed up to the vehicle undersurface or up to the exit plane. The calculation proceeds to the next vehicle expansion ray and the above repeated until Point $C_2$ is reached. Note if the vehicle surface is chosen sufficiently long all cowl rays may be "captured" on the vehicle. In this case the local external flow and/or plume shape may affect the vehicle undersurface pressure distribution. The program automatically determines if such a calculation is to be performed. Figures (4) through (9) illustrate typical exit conditions.

Upon completion of the minimum vehicle turning configuration, the vehicle expansion is incremented by a specified amount and the above process repeated.
FIGURE 4. PARTIAL REFLECTION OF COWL EXPANSION/
NO REFLECTION OF VEHICLE EXPANSION

FIGURE 5. PARTIAL REFLECTION OF COWL AND VEHICLE EXPANSIONS

FIGURE 6. COMPLETE REFLECTION OF COWL EXPANSION/
PARTIAL REFLECTION OF VEHICLE EXPANSION
FIGURE 7. PARTIAL REFLECTION VEHICLE EXPANSION/COMPLETE REFLECTION OF COWL EXPANSION

FIGURE 8. COMPLETE REFLECTION OF COWL AND VEHICLE EXPANSIONS

FIGURE 9. INTERACTION OF EXTERNAL FLOW WITH VEHICLE UNDERSURFACE
until zero flow deflection results at point \( V_2 \). The cowl length is now incremented by a specified amount and the complete process repeated. In this manner a parametric map of lift, thrust and pitching moment is generated as depicted in Figures (10) and (11). The dotted lines are the present analysis for a cowl length of 5 and the solid lines were taken from the previous work described in Reference (2).
\[ T_f = 2000^\circ R \]
\[ P_i = 850 \text{ lb/ft}^2 \]
\[ M_i = 3 \]
\[ \gamma = 1.4 \]
\[ W = 28.96 \]
\[ P_e = 23.1 \text{ lb/ft}^2 \]
\[ \Delta y_{exit} = 6 \]
\[ x_{veh} = 15 \]
\[ \Delta y = 10^\circ \]
\[ z_L = 1 \].

Current Analysis

**FIGURE 10. THRUST VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION ANGLE**
FIGURE 11. LIFT VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION
The following definitions are used in this report for thrust, lift, pitching moment:

\[
T = \int_A (p-p_\infty) \hat{i}_x \cdot d\vec{A}_n + T_{vis}
\]  
(18)

\[
L = \int_A (p-p_\infty) \hat{i}_y \cdot d\vec{A}_n + L_{vis}
\]  
(19)

\[
M_y = + \int x \cdot dL + \int y \cdot dT
\]  
(20)

Figure 12 gives the orientation of the vectors with respect to the vehicle. Internally the integrals range over all the vehicle surface areas. Externally they range over the complete vehicle undersurface as defined by the bounding stream surface and/or flow fence.
As defined above the nozzle thrust, lift and pitching may include viscous forces. These effects will be discussed below. However, we define $T_{vis}$, $L_{vis}$ as

$$T_{vis} = -\int_{A} \left( \frac{\rho a^2}{2} \right)_{\text{local}} C_f \hat{x} \cdot d\mathbf{A}_s$$

$$L_{vis} = -\int_{A} \left( \frac{\rho a^2}{2} \right)_{\text{local}} C_f \hat{y} \cdot d\mathbf{A}_s$$

Surface Area Computation - Internally the nozzle surfaces are defined as the cowl internal surface, vehicle undersurface, and the nozzle sidewall. Externally the surfaces are defined as the nozzle undersurface and/or flow fences. The lateral extent of the cowl surface and nozzle undersurface is defined by the degree of lateral expansion desired.

It is assumed that the nozzle area can be approximated by a series of elemental quadrilaterals, as shown in Figure (13). From Reference (7) the unit normal for an elemental area may be obtained by defining two surface tangent vectors from the diagonals of the quadrilateral.

FIGURE 13. ELEMENTAL SURFACE AREA
That is

\[ \mathbf{T}_1 = T_{1x} \mathbf{i}_x + T_{1y} \mathbf{i}_y + T_{1z} \mathbf{i}_z \]  
\[ \mathbf{T}_2 = T_{2x} \mathbf{i}_x + T_{2y} \mathbf{i}_y + T_{2z} \mathbf{i}_z \]  

where

\[ T_{1x} = x_3 - y_2, \quad T_{1y} = y_3 - y_1, \quad T_{1z} = z_3 - z_1 \]
\[ T_{2x} = x_4 - y_2, \quad T_{2y} = y_4 - y_2, \quad T_{2z} = z_4 - z_2 \]

and the normal \( \mathbf{N} \) is defined as

\[ \mathbf{N} = \mathbf{T}_2 \times \mathbf{T}_1 \]

and the unit normal as

\[ \hat{n} = \frac{\mathbf{N}}{||\mathbf{N}||} \]  

A tangent plane is constructed using the normal vector and the two tangent vectors \( \mathbf{T}_2, \mathbf{T}_1 \). The corners of the surface element are projected onto this plane and the area and centroid of the quadrilateral are calculated as described in Reference (7).

Typical nozzle elemental areas are shown in Figures (14) and (15).
FIGURE 14. UPPER OR LOWER SURFACE AREA ELEMENTS

FIGURE 15. SIDEWALL AREA ELEMENTS
Local skin friction and heat transfer coefficients are computed via curve fit data supplied from Reference (7). These fits are based on the Spalding and Chi method of Reference (4). That is a suitably transformed skin-friction coefficient is given by incompressible formulas based on a suitably transformed Reynolds number, i.e.:

\[ C_{f_\delta} = \frac{C_{f_1}}{F_c} \]  
\[ C_{f_1} = f(R_{x_1}), \quad R_{x_1} = F_{Rx} \cdot Rx \]  

where

\[ C_f = \text{local skin friction coefficient} \]

\[ Rx = \text{Reynolds number} \]

\( (\cdot)_I \) = indicates incompressible

\( (\cdot)_\delta \) = indicates compressible

Now for \( R_{x_1} > 2540 \) the local skin friction is given from Reference (8) as

\[ C_{f_1} = 0.088 \left( \frac{\log R_{x_1} - 2.3686}{\log R_{x_1} - 1.5} \right)^3 \]  

and from Reference (7)

\[ F_c = \frac{A}{\left( \text{AR} \sin \left( \frac{A-B}{C} \right) + \text{AR} \sin \left( \frac{A+B}{C} \right) \right)^2} \]  
\[ A = \frac{H_{AW}}{H_\delta} - 1 \]  

\[ \approx 20 \]
The local properties external to the boundary layer are the local data computed by frozen (NOZD) or equilibrium (NOZDE) programs and are assumed to act through the centroid of the elemental area computed above. The computation requires that a boundary layer origin be specified since the nozzle is assumed to be an extension of the combustor. In addition, a recovery factor for an adiabatic wall calculation is required. However, as a user option wall temperature distributions may be specified.

Local heat transfer coefficients are computed from a modified Reynolds analogy for turbulent flow

\[ St = Sh \cdot C_f/2 \]  

The program requires "Sh" as an input item.
The numerical program developed should be a useful tool in rapidly assessing the affects of varying dominant parameters on scramjet exhaust nozzles. The program has the capability of analyzing a general class of scramjet nozzle configurations. Sophisticated force and moment calculations allow for the inclusion of local viscous affects and accurate computation of lateral forces. In addition, the effects of external flow conditions on nozzle performance may be rapidly assessed as part of the overall procedure. These features make the current program a valuable tool in designing scramjet nozzle configurations.
REFERENCES


APPENDIX I

PROGRAM DESCRIPTION

NOZD - Frozen Nozzle Design
NOZDE - Equilibrium Nozzle Design

A. INPUT

Card 1  (Format 8E10.0)

Column 1-10  PI  (initial pressure, lb/ft²)
            11-20  TI  (initial pressure, °R)
            21-30  wI  (initial molecular wt - frozen)
                    (initial fuel/air equivalence ratio
                     equilibrium)
            31-40  THI  (initial flow deflection angle, degrees)
            41-50  EMI  (initial Mach number)

#Frozen Deck
            51-60,  GAM1  (initial ratio of specific heats)
            61-70  PINF  (free stream pressure, lb/ft²)

#Equilibrium Deck
            51-60  PINF  (free stream pressure, lb/ft²)

Card 2  (Format 8E10.0)

Column 1-10  PF  (external pressure, lb/ft²)
            11-20  TF  (external temperature °R)
            21-30  wf  (external molecular weight)
            31-40  THF  (external flow deflection angle, degrees)
            41-50  EMF  (external Mach number)
            51-60  GF  (external ratio of specific heats)

Card 3  (Format 8E10.0)

Column 1-10  XV1  (axial location of throat on vehicle undersurface ft)
            11-20  YV1  (throat height on vehicle undersurface, ft)
            21-30  XV2  (axial location of vehicle end, ft)
TR 215

Column 31-40  XCl (axial location of throat on cowl surface, ft)
41-50  YCl (throat height on cowl surface, ft)
51-60  XC2 (length to end of cowl, ft)
61-70  DYV (vehicle exit height, \( Y_{v2} - Y_{c2} \), ft)
71-80  DNuc (\( \Delta v_c \) - cowl turning angle, degrees)

Card 4  (Format 8E10.0)
Column 1-10  XFI (axial location of vehicle fence on vehicle undersurface, ft)

Card 5  (Format 8E10.0)
Column 1-10  XTHX (total initial thrust, lbs)
11-20  XLFT (total initial lift, lbs)
21-30  XMOM (total initial pitching moment, lb-ft)
31-40  XVTHX (initial viscous thrust, lbs)
41-50  XVLFT (initial viscous lift, lbs)
51-60  XVMOM (initial viscous pitching moment, lb-ft)
61-70  XSHFT (x-moment axis, ft)
71-80  YSHFT (y-moment axis, ft)

Card 6  Format
Column 1-5  15  ICF - number of different cowl lengths to be executed
6-15  E10.0  DXC - increment to be added to original cowl length \( x_{c2} \)
16-20  15  JFT - number of vehicle turning angles to be run for each cowl length
21-30  E10.0  DTH - increment for vehicle turning angle in degrees
31-35  15  IVIS - option for viscous calculation - 0 if no calculation, 1 for calculation
36-40  15  IT - option for wall temperature calculation; 0-calculate \( T_{wall} \), 1-adiabatic wall

A1-2
(If fVIS is equal to 1 read this card)

**Card 6a**  (Format 8E10.0)

<table>
<thead>
<tr>
<th>Column</th>
<th>1-10</th>
<th>XSTR (virtual origin of boundary layer, ft)</th>
</tr>
</thead>
<tbody>
<tr>
<td>11-20</td>
<td>Rec</td>
<td>(recovery factor)</td>
</tr>
<tr>
<td>31-40</td>
<td>SH</td>
<td>(constant for Stanton number calculation</td>
</tr>
<tr>
<td></td>
<td></td>
<td>St = SH \cdot CF/2.)</td>
</tr>
<tr>
<td>41-50</td>
<td>RT</td>
<td>(throat height, ft)</td>
</tr>
</tbody>
</table>

(If IT is equal to 0 read this card)

**Card 6b**  (Format 8E10.0)

<table>
<thead>
<tr>
<th>Column</th>
<th>1-10</th>
<th>AH(1) coefficients in equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>11-20</td>
<td>BH(1)</td>
<td>( T_{wall} = AH \cdot X^2 + BH \cdot X + CH )</td>
</tr>
<tr>
<td>21-30</td>
<td>CH(1)</td>
<td></td>
</tr>
</tbody>
</table>

**Card 7**  (Format 8E10.0)

<table>
<thead>
<tr>
<th>Column</th>
<th>1-10</th>
<th>AZ coefficients in equation</th>
</tr>
</thead>
<tbody>
<tr>
<td>11-20</td>
<td>BZ</td>
<td>( Z = AZ \cdot X^2 + BZ \cdot X + CZ )</td>
</tr>
<tr>
<td>21-30</td>
<td>CZ</td>
<td></td>
</tr>
</tbody>
</table>

**B. OUTPUT**

Output variables are printed for each uprunning characteristics \( C_+ \) from cowl to vehicle surface or exit plane. In addition, values of thrust, lift and pitching moment are printed as well as the viscous contributions to these values. The value of ideal thrust printed is based on a one dimensional area considerations, assuming

\[
A_{exit} = (YV_2-YC_2) (ZC_2+ZV_2)/2.
\]
FIGURE 16. GEOMETRIC INPUTS
C. SUBROUTINES

1. CWALL - boundary calculation for cowl surface
2. ENDD - boundary point for vehicle end (XV2)
3. INT - interpolation subroutine
4. THM - computes ideal thrust
5. EM3D - computes Mach number correction for lateral expansion
   ideal gas only
6. SWITCH - resets initial calculational line
7. V_WALL - boundary calculation for vehicle surface
8. GEM - locations intersection of straight lines
9. FIX - computes general interior point properties as described in
   Section III
10. PM - calculates Prandtl-Meyer expansion for given $\Delta u$
11. GNURE - computes skin friction and heat transfer coefficients
12. VIS - computes viscosity
13. SNARF - computes elemental area and centroid
14. LTHM - computes lift, thrust and pitching moment for cowl, vehicle
   and fence surfaces
15. ERROR - Newton-Raphson method for finding roots
16. CNT - Commutes local plume shape
17. COWL - computes underexpansion interaction
18. PM1 - ideal gas Prandtl-Meyer expansion - equilibrium program
   only

D. FUNCTIONS

1. GETZ - computes z location via curve $z = A \cdot x^2 + B \cdot x + C$
   (The following are incorporated only into the equilibrium deck)
2. FH - computes static enthalpy of equilibrium mixture,
   \[ H = FH(P, \phi, T) \]

3. FT - computes static temperature from inversion of function, FH
   \[ T = FT(P, \phi, H) \]

4. FGAM - computes equilibrium isentropic component,
   \[ \Gamma = FGAM(T, P, \phi) \]

5. RHEQ - computes equilibrium mixture density,
   \[ \rho = RHEQ(H, P, \phi, T) \]
APPENDIX II

LISTING OF FROZEN FLOW PROGRAM
PROGRAM NOZZLE INPUT, OUTPUT, TAPE5=INPUT, TAPE6=OUTPUT)
COMMUN/IN/PI, I, XI, THI, EM1, GA1, Z1
COMMUN/SHFT/3SHFT, YSHFT
COMMUN/POLY, IPK1, IPK2
COMMUN/TH, 4AX/TH4AX
COMMUN/X(2, 250), Y(2, 250), Z(2, 250), TH(2, 250), XNU(2, 250),
1P(2, 250), T(2, 250), E(2, 250), XNU(2, 250), UT(2, 250)
COMMUN/C/NUV, DNUC, XC1, YC1, XCV, YCV, DNUL
COMMUN/C/L/LC2, YC2
COMMUN/P/PIVF, PIOT, T10T
COMMUN/DIS/23, 2H2, CZ
COMMUN/VAV, AV, CV, XV, XV2, YV2
COMMUN/H01/2H(3), 2H(3), 2H1, XS1R, REC, RH, SH, IT, IVIS
COMMUN/VISF/X, VT, XV, XVF, XVVF
COMMUN/C/CHCP1, G1, NGAS
COMMUN/F/PF, TF, NF, THF, EFM, GF
COMMUN/CNCTC/ITC, PVC, WC, GC, PC, TH, XNUC, ZC, EMC
DIMENSION H0L(11), HULE(6), DNUE(10)
DATA HOLE/2H00, 2H01, 2H02, 2H03, 2H04, 2H05/
DATA HOLE/2HV1, 2HC1, 2HA, 2HH, 2HC, 2HU, 2HV3, 2HC3, 2HF, 2HG, 2HV4/
97 FORMAT(15, 8E12.4)
98 FORMAT(/)
100 FORMAT(8E10.0)
D=0.
EPS=1. E=0.5
DO 23 L=1, 3
AH(L)=0.
BH(L)=0.
23 CH(L)=0.
LH1=1
READ(5, 100) PI, TI, XI, THI, EM1, GA1, PINF
READ(5, 100) PF, TF, NF, THF, EFM, GF
READ(5, 100) XV1, YV1, XV2, XCV, YCV, DYG, DNUC
READ(5, 100) XTF1
READ(5, 100) XTHX, YLFT, XMOM, XVTX, YVLF, XMOM, XSHFT, YSHFT
READ(5, 5921) IC, JXC, JTF, DTH, IVIS, IT
IF (IVIS, ER, 1) READ(5, 100) XSTR, REC, SH, RI
IF (IVIS, ER, 1) READ(5, 100) AM(L), BM(L), CM(L)
5921 FORMAT(15, E10.0, 15, E10.0, 315)
XTHX=THX
YLF1=YLFT
XMOM=XMOM
XVTX=XVTX
YVLF1=YVLF
XMOM=XMOM
DNUC=DNUC/57.3
THI=THI/57.3
DTH=DTH/57.3
READ(5, 100) AZ, BZ, CZ
ZC1=GETZ(XC1)
ZV1=GETZ(XV1)
ZC2=GETZ(XC2)
ZI=ZV1
GI=GA1
WRITE(6, 5922)
5922 FORMAT(1H177, 77)
WR11E(6, 1020)
1020 FORMAT(20X* ROZEN FLOW NOZZLE DESIGNA)
WRITE(6,5930) XV1,YV1,XV2,YC1,JC1,XC2,DYV
5930 FORMAT(/5*XYV1*10*XYV1*10*XYV2*10*XYC1*10*XYC2*10*XYDYV*)
17E13.5/)
1800 FORMAT(* THRUST= E13.5,7X*LIFT= E13.5,7X*MOM= E13.5)
WRITE(6,1800)XTHX,YLFT,XYMOM

1900 FORMAT(* VISCOUS THRUST= E13.5,7X*VISCOUS LIFT= E13.5,7X*)
WRITE(6,1900)XVTHX,YVLFT,XYVMOM

5923 FORMAT(9X*MEJ XI AXIS*/12XY**E13.5/12XY**E13.5/)
WRITE(6,1001)AZ,AY,CZ

1001 FORMAT(9X*LATERAL EXPANSION EQUATION*/9X,22HZ(X) = AZ**X+AY**X+CZ
1/12**AZ**E13.5/12**AY**E13.5/12**CZ**E13.5)

DUM=1.*((GJ-1.)/2.)**EM**X2
PTOT=PI**DUM**((GJ/((GJ-1.)))
TTOT=TI**DUM
RG=49800.
RGAS=RG/41

CPI=GAM1*RGAS/(GAM1=1.)
XC2=XC2
THC1=TH1-DWUC
DO 5000 IXC=1,ICF
XC2=XC2+FLOT((XC-1.)*DXC
CALL THM(E41,PI,GJ,PIF,YV1,JC1,DOV,XYV2,THMAX,XC2,ZI)
WRITE(6,5654)THRMAX

6364 FORMAT(1H131X*IDEAL THRUST = E12.4)
YV2=YC2+TA(YC1)*X(C2-XC1)
YV2=YC2+DYV
THC=TAN((YC2-YC1)/XC2-XC1))
DO 6000 JIT=1,JTF
XTHX=XTHXI
YLFH=YLFTI
XUM=XUM1I
XVTHX=XVTHXI
YVLFTI=YVLFTI
YVMOM=XVVMOM

KSP=50
ISTP=50
IENO=0
IEXT=0
IFLG=0
IND=0

THX=1.E+10
THJ1=FLDA1(JT=1)*DTH
DNUV=TAN((YV2-YV1)/(XV2-XV1))*TH1+THJ1
WRITE(6,1894)DNUV

1894 FORMAT(* VEHICLE EXPANSION = E13.5)
XV=XV1
AV=YV1
BV=TAN(TH1+DNUV)
CV=0.
YF1=AV+BV*X(F1-XV)
IDUM=DNUV+57,3
IF(IDUM,GT,11)IDUM=11
IF(IDUM,EQ,0)IDUM=1
DELNU=DNUV/FL0AT(IDUM)

IDUM=IDUM+1
IMAX=IDUM
IPMAX=IMAX+2
DO 10 IE=1,IMAX
X(IE,1)=E(V1)
Y(IE,1)=Y(V1)
Z(IE,1)=GETZ(XV1)
TH(1,1)=TH1+FLDA1(I-1)*DELNU

10 CONTINUE
```
10 CONTINUE
IDUM=IDUM*57.3
IF (IDUM,GT,11) IDUM=11
IF (IDUM,EQ,0) IDUM=1
DELNU=DNUC/FLAT(IDUM)
IDUM=IDUM+1
X(2,1)=XC1
Y(2,1)=YC1
Z(2,1)=GETZ(XC1)
ICWL=0
I=1
THII=THI
/ 20 CONTINUE
TH(2,1)=THII=FLAT(I-1)*DELNU
XNU(2,1)=FLOAT(1-1)*DELNU+XNU1
CALL PM(2,1)
22 CONTINUE
K=2
36 CONTINUE
IF (K,LT,KMAX) GO TO 41
IF (IEND,EQ,1) GO TO 53
CALL VNULL(I MAX,KMAX)
MM=K+1
L=K-1
IF (ICWL,EQ,1)L=K
IF (X(2,K).LT.XV2-EPS) GO TO 24
CALL ENDD(L,MM,KMAX)
X(2,K)=XV2
IF (KSTP.GE.KMAX) KSTP=KMAX
IND=1
GO TO 17
24 IF (I,GT,1) GO TO 30
A=V(Y(2,KMAX))
B=V(TAN(TH(2,KMAX))
C=V(YV2-AV-BV*(YV2-X(2,KMAX)))/(YV2-X(2,KMAX))**2
THV2=ATAN(3V+2.*CV*(YV2-X(2,KMAX))
XV3=X(2,KMAX)
IF (XF1,GT,XV) YF1=AV+BV*(XF1-XV)+CV*(XF1-XV)**2
SLE=(YF1-YC2)/(XF1-XC2)
IF (THV2,LT,0.) GO TO 462
DO 16 KT=2,KMAX
XT=XT2=XK3=XKU=0.25
M=K+1
J=KT-1
N=J
IF (K1,EQ,KMAX) J=KT-2
IF (K1,EQ,2.*K1,EQ,KMAX) N=J
IF (K1,EQ,2.*K1,EQ,KMAX) XK3=0.,
IF (K1,EQ,2.*K1,EQ,KMAX) XK1=XK2=XKU=333333
CALL LTH4(X(2,K),Y(2,K),Z(2,K),Y(1,K),Z(1,K),Y(1,J),Z(1,J),
1Y(1,J),Z(1,J),X(2,K),Y(2,K),Z(2,K),P(2,K),P(1,K),P(1,J),
2P(2,K),0(2,K),0(1,K),0(1,J),0(2,K),0(2,J),1(2,K),1(2,J),
ST(2,K),TH(2,K),TH(1,N),TH(1,J),TH(2,K),XK1,XK2,XK3,XKU=0.,
4XTHX,TLC1,X4NU,TCLF(M),ST(M),53)
16 CONTINUE
GO TO 53
11 CONTINUE
```
IF(ICAL.EQ.1.AND.K.EQ.2.AND.IEXT.EQ.1)CALL CNT
L=K
IF(ICAL.EQ.1)L=K+1
IF(I.EQ.1) L=K-1
L=M=1
LL=K+ICL-1
IF(L.GT.ISTP.AND.IEND.EQ.1)GO TO 30
MM=K-1
CALL FIX(2,K-1,1,2,K)
IF(I.EQ.1)GO TO 30
IF(K.GT.1)SLA=(Y(2,K)-Y(2,K))/X(2,K)*X(2,K)
CALL GEN(X(2,MM),Y(2,MM),SLA,XC2,YC2,SLF,XC,YC)
IF(XC.GE.XC1,LT,X(2,MM)=EPS)GO TO 30
KSTP=K
WRITE(6,88)KSTP,KMAX,IMAX
88 FORMAT(1X,4(5I1))
30 CONTINUE
L=M=1
IF(ICAL.EQ.1)GO TO 32
IF(IEXT.EQ.1)GO TO 32
IF(ICAL.EQ.0.OR.K.GT.2)GO TO 32
CALL LTM(X(1,1),Y(1,1),Z(1,1),X(2,1),Y(2,1),Z(2,1),X(2,1),
1Y(2,1),0.,X(1,1),Y(1,1),0.,P(1,1),P(2,1),P(2,1),P(1,1),Q(1,1),
2Q(2,1),Q(2,1),Q(1,1),T(2,1),T(2,1),T(2,1),T(1,1),T(1,1),
3TH(2,1),TH(1,1),25,25,25,25,25,25,1.,XTHX,YLFT,XYOM,CFL,STL,2)
CALL LTM(X(1,1),Y(1,1),Z(1,1),X(2,1),Y(1,2),Z(1,2),X(2,1),Y(2,1),
1Z(2,1),X(2,1),Y(2,1),Z(2,1),P(1,1),P(1,2),P(2,1),P(2,1),Q(1,1),
2Q(1,2),Q(2,1),Q(2,1),T(1,1),T(1,2),T(1,2),T(2,1),TH(1,2),
3TH(2,1),TH(2,1),33333.,33333.,33333.,0.,1.,XTHX,YLFT,XYOM,
4CF(1,1),ST(1,1),3)
32 CONTINUE
XR1=XR2=XR3=XR4=.25
IF(IEND.EQ.0.AND.K.EQ.KMAX)GO TO 17
AVG=4.
SLB=(Y(2,K)-Y(1,1))/X(2,K)-X(1,1)
CALL GEN(X(1,1),Y(1,1),SLB,XC2,YC2,SLF,XC,YC)
RAT=(XC-X(1,1))/X(2,K)-X(1,1)
IF(RAT.GT.99)GO TO 144
AVG=2.
GO TO 144
143 AVG=AVG-1.
144 CONTINUE
IF((X(2,MM)-X(1,LM)),EQ.0.)GOTO 148
SLB=(Y(2,MM)-Y(1,LM))/X(2,MM)-X(1,LM)
CALL GEN(X(1,1),Y(1,1),SLB,XC2,YC2,SLF,XC,YC)
RAT=(XC-X(1,LM))/X(2,MM)-X(1,LM)
IF(RAT.GT.99)GO TO 148
IF(RAT.GT.99)GO TO 145
AVG=AVG-2.
GO TO 148
145 AVG=AVG-1.
148 CONTINUE
CALL LTM(X(2,MM),Y(2,MM),Z(2,MM),X(1,LM),Y(1,LM),Z(1,LM),X(1,LM),
1Y(1,LM),Z(1,LM),X(2,K),Y(2,K),Z(2,K),P(2,MM),P(1,LM),P(1,LM),P(2,K),
2Q(2,MM),Q(1,LM),Q(1,LM),T(2,MM),T(1,LM),T(1,LM),T(1,LM),
3TH(2,MM),TH(1,LM),TH(1,LM),TH(2,R),TH(2,K),TH(2,K),XK1,XK2,XK3,XK4,AVG,XTHX,YLFT,
4XYOM,CFL,ST(1,1),3)
IF(IEND.EQ.1.OR.K.LT.KMAX)GO TO 33
17 CONTINUE
M=K-1
L=IN
IF(ICAL.EQ.1)L=K
CALL GEM(x(2,m),Y(2,m),SLA,XC2,YL2,SLF,XC,YC)
RAT=(X(x(2,m))/X(2,k)-X(2,k))
IF(RAT.GT.0.99)G0 TO 177
AVG=AVG-1.

177 SLB=(Y(2,m)-Y(1,l))/(X(2,m)-X(1,l))
CALL GEM(x(1,l),Y(1,l),SLB,XC2,YC2,SLF,XC,YC)
RAT=(X(x(1,l))/X(2,m)-X(1,l))
IF(RAT.GT.0.99)G0 TO 178
IF(RAT.GT.0.99)G0 TO 177
AVG=AVG-1.

178 CONTINUE
AVG=AVG/3.

CALL LTH(x(2,m),Y(2,m),Z(2,m),X(1,l),Y(1,l),Z(1,l),X(2,k),Y(2,k),
1Z(2,k),X(2,k),Y(2,k),Z(2,k),P(1,l),P(1,k),P(2,k),U(2,k),
2G(1,l),G(1,l),G(2,k),T(1,l),T(2,k),TH(2,k),TH(1,l),
3TH(2,k),TH(2,l),-33333,-33333,-33333,0.,AVG,XTH,YLFT,XMOD,
4CF(2,l),ST(1,l),3)

43 CALL LTH(x(1,l),Y(1,l),Z(1,l),X(1,l),Y(1,l),Z(1,l),X(2,k),Y(2,k),
1U,x(2,k),Y(2,k),Z(2,k),P(1,l),P(1,k),P(2,k),U(1,l),
2G(2,k),G(2,k),T(1,l),T(2,k),TH(1,l),TH(2,k),TH(2,k),
3TH(2,k),TH(2,l),-25,-25,-25,-25,1.,XTH,YLFT,XMOD,CFU,STU,1)

55 IF(1END,EQ,1)KMAX=ISTP
K=K+1
IF(K.LE.KMAX)G0 TO 36
KP=KMAX
IF(1END,EQ,1)KP=KMAX-1-ICAL
IF(1IND,EQ,1,1END=1

WRITE(6,6845)

6885 FORMAT(1*X*P1,X0*X*X11*X*Y*X*PRESSURE*S*X*ANGLE*S*X*MACH*S*X*TEMPERATU*
IRE=1*X*P1,FUCTI0N*S*X*VELOCITY*)
DO 50 L=1,K
WRITE(6,97) L,X(2,L),Y(2,L),P(2,L),T(2,L),X(2,L),T(2,L),XNU(2,L)
50 CONTINUE

WRITE(6,2091)CFU,STU,CFL,STL

2091 FORMAT(* VEH, FRIC, COFF=AE13.5,2*X*VEH, STANTON NUM-=*E13.5,2*X*
1COAL FRIC, COEF.=*E13.5,2*X*CFU,STU,STANTON NUM-=*E13.5)
WRITE(6,450)
NPR1=1MAX/S
DO 166 K=1,NPR1
K1=K$K=K1+PRTS@K3=K2+PR1,SK4=K3+NPR1,SK5=K4+NPR1
KDONE=K5
WRITE(6,500)CF(K1),ST(K1),CF(K2),ST(K2),CF(K3),ST(K3),
1CF(K4),ST(K4),CF(K5),ST(K5)

166 CONTINUE
IF(KDONE.EQ,1MAX)G0 TO 55
NPR2=5*NPR1+1
DO 414 K=NPR2,1MAX
WRITE(6,550)CF(K),ST(K)
414 CONTINUE
55 CONTINUE

450 FORMAT(1*X,5(6X,*CF *,9X,*ST *,3X))
500 FORMAT(1*X,10E13.5)
550 FORMAT(105X,2E13.5)
WRITE(6,98)
WRITE(6,1800)XTHX,YLFT,XMOD
WRITE(6,1900)XVTHX,YVLFT,XVMOD
IMAX=KMAX
I=I+1
IF(1.GT.100)NC=L=1
IF((NC,L,EQ,0)KMAX=KMAX+1
IF(IEXT.EQ.0.AND.X(1,1).GE.XC2-EPS)GO TO 5900
IF(IWL.EQ.1)GO TO 22
GO TO 20
462 CONTINUE
WRITE(6,992)
992 FORMAT(10H,2.0A)
GO TO 6000
5900 CONTINUE
IF(P(1,1).LE.PF)GO TO 5396
CALL COWL(JVUE)
IDOUM=6
I=2
XNU1=XNU(1,1)
TH1=TH(1,1)
ICWL=0
KMAX=KMAX+1
DELNU=ONUE/FLUAT(IDUM)
IDOUM=IDOUM+1
IEXT=1
GO TO 20
5396 CONTINUE
6000 CONTINUE
5000 CONTINUE
END
SUBROUTINE CWMALL(I, AX, KPA, XSTP, IS1P)
COMMON X/ X(1, 2, 50), Y(1, 2, 50), Z(1, 2, 50), TH(1, 2, 50), XNU(1, 2, 50),
T(1, 2, 50), T2(1, 2, 50), E(1, 2, 50), XNU(1, 2, 50), R(2, 50)
COMMON C/ ONIV, O, UC, JC1, YC1, YV1, DMUL
COMMON C/ CAL, XEC, YC2
D = 0.
1AD0 = 0.
LOOP = 0.
XTH1 = TH(1, 1) - XNU(1, 1)
Q1 = XNU(1, 1) - TH(1, 1)
CALL INT (1, 2), Y(1, 2), Z(1, 2), TH(1, 2), XNU(1, 2), P(1, 2), T(1, 2),
2EAM(1, 2), XNU(1, 2), D, D, D, D, D, D, D, D, D, D, D, XA, YA, ZA, THA, XNUA, PA, TA,
Q2 = XNU(1, 2) - TH(1, 2)
44 XTH2 = THA - XNUA
IF (LOOP.EQ. 0) XTHM3 = XTHM2
SLA = (TAN(XTHP2) + TAN(XTHM3))/2.
SLB = (YC2 - YC1)/(XC2 - XC1)
CALL GEM(XA, YA, SLA, XC2, YC2, SLB, XC, YC)
IF (XC .LE. XC) GO TO 43
1AD0 = 1
DX = X(1, 2) - X(1, 1)
SLU = (Y(1, 2) - Y(1, 1))/DX
IT = 1
XA = X(1, 2) + X(1, 1)/2.
24 RAX = (XA - X(1, 1))/DX
SLB = XTHM1 + RAX*(XTHM2 - XTHM1)
SLB = TAN(SLB)
IF (LOOP .EQ. 1) SLB = (SLB + TAN(XTHM2))/2.
CALL GEM(X(1, 1), Y(1, 1), SLU, XC2, YC2, SLB, XAT, YA)
ER = ABS((XA - XAT)/DX)
IF (ER .LT. 1. E-04) GO TO 63
IT = IT + 1
IF (IT .GT. 10) GO TO 68
XA = XAT
GO TO 24
68 WRITE(6, 38)
38 FORMAT(* TDO MANY ITER IN CHALL *)
WRITE(6, 39) LOOP, SLU, SLB, XA, XAT
39 FORMAT(1X, 15, 2E13.5)
STOP
63 CONTINUE
XC = XC2
YC = YC2
TM = IMAX + 1
X(1, IM) = XA
Y(1, IM) = YA
Z(1, IM) = GETZ(XA)
XNU(1, IM) = XNU(1, IM) + RAX*(XNU(1, 2) - XNU(1, 1))
DA = U + RAX*(Q2 - 31)
TH(1, IM) = XNU(1, IM) - GA
CALL PM (1, IM)
CALL INT (1, 2, X(1, IM), Y(1, IM), Z(1, IM), TH(1, IM), XNU(1, IM), P(1, IM),
1T(1, IM), EM(1, IM), XNU(1, IM), D, D, D, D, D, D, D, D, D, D, D, XA, YA, ZA, THA, XNUA,
2PA, TA, EMA, XNUA, 1)
43 RATHA = THA + XNUA
IF (ABS(XC2 - XC).LE. 0.001) XC = XC2
X(2, 1) = XC
Y(2, 1) = YC
CALL PM(2,1)
XTHM3=TH(2,1)-X4U(2,1)
IF(LUPEQ.1)GO TO 6
LOOP=1
GO TO 44
6 IF(IADD.EQ.0)RETURN
IMAX=IMAX+1
KMAX=KMAX+1
KSTOP=KSTOP+1
ISTP=ISTP+1
DO 46 JJ=3,IMAX
LJ=1MAX-JJ+3
LI=LJ-1
46 CALL SWITCH(1,L1,1,LJ)
WRITE(6,181)IADD,XA,X(1,1),X(1,2)
181 FORMAT(1X,15,3E13.5)
CALL INT(0,XA,YA,ZA,THA,XNUA,PA,TA,EMA,XNUA,D,D,D,D,D,D,D,D,
1X(1,2),Y(1,2),Z(1,2),TH(1,2),XNU(1,3),P(1,2),T(1,2),E(1,2),
2X4U(1,2),-1.)
RETURN
END
SUBROUTINE END0(L,KK,K6)
COMMON/VA,VB,VC,VX,VYV
COMMON/X(1,L),Y(1,L),Z(1,L),TH(1,L),XNU(1,L),
     T(1,L),EM(1,L),XUX(1,L),UX(1,L)
IT=1
B=0.
A=1.
X=TH(1,L)+XNU(1,L)
X=TH(2,KK)+XNU(2,KK)
X=0.
Q1=TH(1,L)+XNU(1,L)
Q2=TH(2,KK)+XNU(2,KK)
X=(X(1,L)+X(2,KK))/2.
DX=X(2,KK)-X(1,L)
SLM=(Y(2,KK)-Y(1,L))/DX
12 RAI=(XA-X(1,L))/DX
SLP=X(1)+RIAX(X(2)-X)
SLP=ASLP+3*X
CALL GE*(X(1,L),Y(1,L),SLM,XV2,YV2,SLP,XAT,YA)
EI=ABS((XA-XAT)/DX)
IF(EI.LE.1.E-03)GOTO 14
IT=IT+1
IF(IT.LE.10)GOTO 8
WRITE(6,33)
33 FORMAT(* TJO MANY ITER IN END *)
WRITE(6,34),K6,SLP,XAT,X(2,KK),X(1,L),SLM,XM3
STOP
6 XA=XAT
GOTO 12
14 QASQHAT*(Z2-U1)
TH(1,L+2)=TH(1,L)+RIAX(TH(2,KK)-TH(1,L))
XNU(1,L+2)=XNU(1,L+2)-RA*TH(1,L+2)
Z(1,L+2)=GETZ(XA)
CALL PM(1,L+2)
RA=XNU(1,L+2)-TH(1,L+2)
X(2,K6)=XV2
Y(2,K6)=YV2
Z(2,K6)=GETZ(XV2)
TH(2,K6)=ATA*(B+2.*CV*(XV2-XV))
XNU(2,K6)=RA+TH(2,K6)
CALL PM(2,K6)
IEND=T
IF(B.GT.0.)RETURN
IT=1
B=.5
A=+.5
X=TH(2,K6)+XNU(2,K6)
GOTO 12
END
SUBROUTINE INT(RAT, X1, Y1, Z1, TH1, XNU1, P1, T1, EM1, XMU1,
1X2, Y2, Z2, TH2, XNU2, P2, T2, EM2, XMU2, X3, Y3, Z3, TH3, XNU3, P3, T3,

2EM3, XNU3, OPT)
Q1=XNU1+OPT*TH1
Q2=XNU2+OPT*TH2
X3=X1+RAT*(X2-X1)
Y3=Y1+RAT*(Y2-Y1)
Q3=Q1+RAT*(Q2-Q1)
P3=P1+RAT*(P2-P1)
T3=T1+RAT*(T2-T1)
EM3=EM1+RAT*(EM2-EM1)
XNU3=XNU1+RAT*(XNU2-XNU1)
TH3=(US-XNU3)/OPT
XMU3=ASIN(1./EM3)
Z3=GETZ(X3)
RETURN
END
SUBROUTINE TH4(EM1,P1,G,PINF,YV1,YC1,DYV,XV2,THRMAX,XC2,Z1)
A1=YV1-YC1
A1=A1*Z1
Z1=GETZ(XV2)
Z2=GETZ(XC2)
A2=DYV*(Z1+Z2)/2.
FM1=1.+(G-1.)/2.*EM1**2
DUM=(2.*FM1/(G+1.))**((G+1.)/2.)/(G-1.)
ASTE=A1*EM1/DUM
P1UT=P1/FM1**((G/(1.-G)))
EM2=EM1*SGRT(A2/A1)
IF (EM1,GT,3.) LTR2=(A2/A1)**3*EM1
AF=A2/ASTE
IFM=0
10 CONTINUE
FM2=1.+(G-1.)/2.*EM2**2
DUM=(2.*FM2/(G+1.))**((G+1.)/2.)/(G-1.)
AF=AF/DUM/EM2
ERA=(AF-ATE)/AF
IF (ABS(ERA),LT,1.E-03) GO TO 20
CALL ERROR(20000,ITM,EM2,ERA,1.1,EM21,ERA1)
GO TO 10
20 CONTINUE
F1=P1*A1*(1.+G*EM1**2)
P2=F1**2*(G/(1.-G))*P1UT
F2=P2*A2*(1.+G*EM2**2)
THRMAX=F2=F1=P1*F**(A2=A1)
RETURN
END
SUBROUTINE EM3D(EM, EMX, X, G, Z, ZI)
GN=(G+1.)/((G-1.)/2.*)
Y=(1.+((G-1.)/2.*EM*EM)**GN
Y=Y/EM/((G+1.)/2.*GN
A=Y*Z/ZI
EMX=2.*SJRT(Z)
ITM=1
10 AT=(1.+((G-1.)/2.*EM*EM)**GN
AT=AT/EM/((G+1.)/2.*GN
ERM=(AT-A)/A
IF(ABS(ERM).LT.1.E-03) GO TO 20
CALL ERROR(10, ITM, EMX, ERM, 1, 1, EMX1, ERM1)
GO TO 10
20 CONTINUE
RETURN
END
SUBROUTINE SWITCH(12, K, 11, L)
COMMON/X/X(2,50), Y(2,50), Z(2,50), TH(2,50), XNU(2,50),
P(2,50), T(2,50), EM(2,50), XMU(2,50), Q(2,50),
X (11,L)=X (12,K)
Y (11,L)=Y (12,K)
Z (11,L)=Z (12,K)
TH (11,L)=TH (12,K)
XNU(11,L)=XNU(12,K)
P (11,L)=P (12,K)
T (11,L)=T (12,K)
EM (11,L)=EM (12,K)
XMU(11,L)=XMU(12,K)
Q(11,L)=Q(12,K)
RETURN
END
FUNCTION GETZ(x)
COMMON/DS/, AZ, BZ, CZ
GETZ=AZ**2+BZ*X+CZ
RETURN
END
SUBROUTINE WALL (IMAX, KMAX)

COMMON/V/X(X(2,50),Y(2,50),Z(2,50),TH(2,50),XNU(2,50),
1P(2,50),T(2,50),EW(2,50),XMU(2,50),W(2,50)
COMMON/VAV, BV, CV, XV, XV2, YV2
COMMON/C/ONUC, ONUC, XCV1, YCV1, XV1, YV1, DNUL

IEG=0
N=1
SL2=TH(1, IMAX)
SL1=TAN(TH(2, KMAX-1) + XMU(2, KMAX-1))
SL13=SL1

CALL GEM(X(2, KMAX-1), Y(2, KMAX-1), SL1, X(1, IMAX), Y(1, IMAX), SL2,
IXG, YG)

12 YN = AV + BV * (XG - XV) + CV * (XG - XV) ** 2

ERR = (YN - YG) / (YV1 - YE1)
IF(ABS(ERR), LT, 1.E-02) GO TO 10

CALL ERROR(1, IT, XG, ERR, 9, XCV1, ERM1)
YN = Y(2, KMAX - 1) + SL15 * (XG - X(2, KMAX - 1))
TH(2, KMAX) = ATAN(BV + 2. * CV * (XG - XV))
XNU(2, KMAX) = TH(2, KMAX) - TH(2, KMAX - 1) + XMU(2, KMAX - 1)
X(2, KMAX) = XG
Y(2, KMAX) = YG
Z(2, KMAX) = SETZ(XG)
CALL PM(2, KMAX)
IF(IEG, EN, 1) RETURN

IEG = 1
SL13 = 5 * (SL1 + TAN(TH(2, KMAX) + XMU(2, KMAX)))
GO TO 12

END
SUBROUTINE GEM(XA, YA, SLA, XB, YB, SLB, XC, YC)

XC = (YB - YA + SLA*XA - SLB*XB) / (SLA - SLB)
YC = YA + SLA*(XC - XA)
RETURN
END
SUBROUTINE ERROR (I,IT,X,ER,F,X1,ER1)
IT=IT+1
IF(IT.LT.15) GO TO 12
WRITE(6,13)
13 FORMAT(*ERROR TEST NUMBER *)
WRITE (6,20) 1
20 FORMAT(15)
STOP
12 IF(IT.GT.2) GO TO 14
ER1=ER
X1=X
X=X*F
IF(X.EQ.X1) X=X+.02
RETURN
14 XD=X1=ER1*(X-X1)/(ER-ER1)
ER1=ER
X1=X
X=Xd
RETURN
END
SUBROUTINE FIX(I1,K1,I2,K2,I3,K3)
COMMON/X/X(2,50),Y(2,50),Z(2,50),TH(2,50),XNU(2,50),
IP(2,50),T(2,50),E4(2,50),XNU(2,50),N(2,50)
DUM=0.
DUM4=0.
A=1.
B=0.
DUM1=TAN(TH(I1,K1)+XNU(I1,K1))
DUM2=TAN(TH(I2,K2)-XNU(I2,K2))
10 IF(B,GT,0.) DUM3=TAN(TH(I3,K3)+XNU(I3,K3))
IF(B,GT,0.) DUM4=TAN(TH(I3,K3)-XNU(I3,K3))
SL1=A*DUM1+B*DUM3
SL2=A*DUM2+B*DUM4
CALL G,4(X(I1,K1),Y(I1,K1),SL1,X(I2,K2),Y(I2,K2),SL2,X(I3,K3),
Y(I3,K3))
Z(I3,K3)=GE12(X(I3,K3))
XNU(I3,K3)=.5*(XNU(I1,K1)+XNU(I2,K2)+TH(I2,K2)-TH(I1,K1))
TH(I3,K3)=.5*(XNU(I2,K2)-XNU(I1,K1)+TH(I2,K2)+TH(I1,K1))
CALL PM(I3,K3)
IF(B,GT,0.) RETURN
A=.5
B=.5
GO TO 10
END
SUBROUTINE PM(IY, KX)
COMMON/X, PI,/T, T1, TH, EM, EM1, Z1, XNU(2, 30), Y(2, 50), Z(2, 50), TH(2, 30)
1P(2, 50), T(2, 50), EM(2, 50), XNU(2, 50), Q(2, 50)
COMMON/CP, CP1, G1, RGAS
DNU=XNU(IY, KX)
G=G1
EMI=EM1
GG=SIGN((G+1.)/(G-1.))
XM1=SQRT(EM1**2-1.)
XNU1=GG*ATAN(XM1/GG)-ATAN(XM1)
EM2=DNU/((1.5-XWU1)*(6.+EM1)+EM1)
I1=1
10 XM2=SIGN((EM2**2-1.)
DNU=GG*(ATAN(XM2/GG)-ATAN(XM1/GG)+ATAN(XM1)-ATAN(XM2)
ERROR=DNU-DNU
IF (ABS(ERROU), L1, 1.E-04) GO TO 20
CALL ERROR(I3, I75, EM2, ERROU, I1, IT1, EM21, ERROU1)
GO TO 10
20 CONTINUE
CALL EM30(EM2, EMX, X(IY, KX), G, Z (IY, KX), Z1)
P(IY, KX)=PTOT/(1.+((G-1.)/2.+EMX**2)**(G/(G+1.))
T(IY, KX)=TTOT/(1.+((G-1.)/2.+EMX**2)
EM(IY, KX)=EMX
XNU(IY, KX)=ASIN(1./EMX)
AX=SIGN(G*RGAS*T(IY, KX))
Q(IY, KX)=EMX*AX
RETURN
END
SUBROUTINE GNEUE(RH, GP, TR, XI, CF, ST, L)

COMMON/CP, CPI, GI, RGAS

COMMON/HOT/AH(3), BH(3), CH(3), XSTR, REC, RT, SH, IT, IB15

HDEL = R * G * A / 2.

HAw = 1. + REC * X * G / 2. / HDEL

IF (IT, EQ. 0) HA = AH(L) * (X - XI) ** 2 + BH(L) * (X - XI) + CH(L)

IF (IT, EQ. 1) GO TO 46

HA = CPI * T

HA = HA / HDEL

GO TO 48

46 HA = HA

48 A = HA - 1.

B = HA - 1.

C = SQRT((A + B) ** 2 + 4 * A)

FC = A / (ASIN((A - B) / C) + ASIN((A + B) / C)) ** 2

FX = HA ** (1.7 / 2) / (FC * (HA) ** (1.474))

CALL VIS(T, XM4U)

REX = RH * ((X + XSTR) * RT) / XM4U

REX = FX * REX

CF1 = 0.88 * ALOG10(RFX1) - 2.3686 / (ALOG10(REXI) - 1.5) ** 3

CF = CF1 / FC

ST = CF * SH / 2.

RETURN

END
SUBROUTINE VIS(T, XMUU)
XMUU=2.27*T**1.5*1.E-08/(T+198.6)
RETURN
END
SUBROUTINE SwARP(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,AVX,AVY,AVZ, 
1NX,1NY,1NZ,AS,X0,Y0,Z0,LO) 
DIMENSION XPA(4),YPA(4),ZPA(4),XI(4),ETA(4) 
XPA(1)=X1 
YPA(1)=Y1 
ZPA(1)=Z1 
XPA(2)=X2 
YPA(2)=Y2 
ZPA(2)=Z2 
XPA(3)=X3 
YPA(3)=Y3 
ZPA(3)=Z3 
XPA(4)=X4 
YPA(4)=Y4 
T1X=X3-X1 
T1Y=Y3-Y1 
T1Z=Z3-Z1 
T2X=X4-X2 
T2Y=Y4-Y2 
T2Z=Z4-Z2 
XNX=12Y*T1Z-T1Y*T2Z 
XNY=T1X*T2Z-T2X*T1Z 
XNZ=T1X*T1Y-T1X*T2Y 
VN=SQR((XNX**2+XNY**2+XNZ**2)) 
IF(VN.LE.1.E-15)GO TO 6 
XNX=XNX/VN 
XNY=XNY/VN 
XNZ=XNZ/VN 
D=XNX*(AVX-X1)+XNY*(AVY-Y1)+XNZ*(AVZ-Z1) 
PD=ABS(D) 
T=SQR(T1X*T1X+T1Y*T1Y+T1Z*T1Z) 
T1X=T1X/T 
T1Y=T1Y/T 
T1Z=T1Z/T 
T2X=XNY*T1Z-XNZ*T1Y 
T2Y=XNZ*T1X-XNY*T1X 
T2Z=XNX*T1Y-XNY*T1X 
DO 1000 J=1,4 
XPA(J)=XPA(J)+X*XD 
YPA(J)=YPA(J)+Y*YD 
ZPA(J)=ZPA(J)+Z*ZD 
D=D 
XDIF=XPA(J)-AVX 
YDIF=YPA(J)-AVY 
ZDIF=ZPA(J)-AVZ 
XI(J)=T1X*XDIF+T1Y*YDIF+T1Z*ZDIF 
1000 ETA(J)=T2X*XOIF+T2Y*YDIF+T2Z*ZDIF 
XIO=(XI(4)*(ETA(1)-ETA(2))+XI(2)*(ETA(4)-ETA(1)))/(ETA(2)-ETA(4)) 
I/3. 
ETA0=ETA(1)/3. 
DO 1020 J=1,4 
XI(J)=XI(J)-XI0 
1020 ETA(J)=ETA(J)-ETA0 
AS=ETA2(2)-ETA(4)))*(XI(5)-XI(1))/2.
SUBROUTINE GNT
COMM/1N/PI,T1,HI,THI,EMI,SA0I,Z1
COMM/VTC/TTC,PIC,VGC,PC,THC,NUC,ZC,EMC
COMM/P/PINFO,PTOT,TTUT
CC=AM/CPI,GI,KGAS
COMM/Y(X2,50),Z2,50),TH2,50),XP(Y2,50),
1P2,50),T2,50),EM2,50),XMY2,50),O2,50)
A=1$l. B=0.
PTOT=PTOT $ POTS=PTOT $ GIS=GI $ ZIS=ZIS $ EMI=EMI
IT=1
RD=XWU(1,2)+TH(1,2)
TH(2,1)=TH(1,1)
20 SLA=ATAN(TH(2,1))+ATAN(TH(1,1))
SLB=TH(1,2)-XWU(1,2)
SLB=ATAN(SLB)
IF(B.GT.O)SLA=SLA+B*ATAN(TH(2,1)-XMY(2,1))
CALL GEM(X(1,2),Y(1,2),SLB,X(1,1),Y(1,1),SLA,X(2,1),Y(2,1))
Z(2,1)=GETZ(X(2,1))
XNU(2,1)=R9-TH(2,1)
CALL PM(2,1)
TUTS=TUTS $ PTOT=PTOT $ GIS=GI $ ZIS=ZIS $ EMI=EMI
RC=NUC-THC
TH(2,2)=TH(2,1)
XNU(2,2)=R2+TH(2,2)
X(2,2)=X(2,1)
Y(2,2)=Y(2,1)
Z(2,2)=GETZ(X(2,2))
CALL PM(2,2)
ER3=(P(2,1)-P(2,2))/PC
IF(CABS(ER3)>1.0.E-04)GO TO 16
CALL ERROR(IER,IT,TH(2,1),ER3,1.01,TH23,ER23)
TUTS=TUTS $ PTOT=PTOT $ GIS=GI $ ZIS=ZIS $ EMI=EMI
GO TO 20
16 CONTINUE
TUTS=TUTS $ PTOT=POTS $ GIS=GI $ ZIS=ZIS $ EMI=EMI
IF(B.GT.O)RETURN
IT=1
A=.5 $ B=.5
GO TO 20
END
SUBROUTINE COSL(DNUE)

COMMON/X/X(2,50),Y(2,50),Z(2,50),TH(2,50),XNU(2,50),
1P(2,50),;I(2,50),EM(2,50),XNU(2,50),Q(2,50)
COMMON/CL/CL/CC2, YC2
COMMON/PC/PC,PT1,PT2,PT3
COMMON/P/PC,PT1,PT2,PT3
COMMON/EM/EM,EM,EM,EM,EM
COMMON/CT/CT,PTC,EMT,EMT,EMT,EMT
COMMON/EMT/EMT,EMT,EMT,EMT,EMT
COMMON/1T/1T,1T,1T,1T,1T

PCC=(PF+P(1,1))/2.
GM1=GF+1.
GP1=GF+1.
EMF2=EMF*EF
DUM=1.+GF*EMF2/2.
PTF=PF*DUM**2*(GF/GM1)

TTF=TF*DUM
IOTS=IIII1 $GIL=$GI
XNUF=ASIN(1./EF)

BET=(THF-XNUF+TH(1,1))*1.1-THF
20 DUM=(EMF*SIY(BET))*2
PC=(2.*GF)*DUM/2./GM1
TC=PC/(GM1*DUM+2.)/DUM/GP1
EMC=(EMF*PTC/(GM1*PC+GM1)-2.*PC/(PC+PC-1.))/(GM1*PC+GP1)/PC
EMC=EMC(SORT(EMC))
PC=PC+PF

THC=TAN(BET)*GM1*EMF2/(DUM-1.)/2.-1.)
THC=ASIN(1./THC+THF)
PTC=(2.*TC/GM1*EMF2+2.)*GM1/GM1
PTC=PTC/PTC

TTC=TTF
TC=TC*TF

RA=XNU(1,1)+TH(1,1)
X(2,1)=XC2 Y(2,1)=YC2 TH(2,1)=THC
Z(2,1)=GETZ(XC2)
XNU(2,1)=RA+TH(2,1)
CALL PM(2,1)
ER4=(PC+P(2,1))/PCC
IF(ABS(ER4) LT,1. E-04)GO TO 16
CALL ERROR(4,IT,BET,ER4,1.05,BET1,ER4)
GO TO 20

16 DNUE=AS(3*N(2,1)=XNUC(1,1))
IOTS=IOTS $GIL=$GI

RETURN

END
APPENDIX III

LISTING OF EQUILIBRIUM FLOW PROGRAM
PROGRAM NOZUE(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
COMMON/IV,PI,II,XI,THI,EMI,GI,ZI,HI,RHI,U1
COMMON/SHF/XSHFT,YSHFT
COMMON/THMAX/THM4AX
COMMON/XX(2,50),YY(2,50),Z(2,50),TH(2,50),XNU(2,50),
IP(2,50),IT(2,50),EM(2,50),XNU(2,50),O(2,50),LH(2,50),KH(2,50),
26(2,50)
COMMON/D/P411
COMMON/H01/AY(3),AH(3),CH(3),XSTR,REC,RH,SH,IT,IVIS
COMMON/C/NUV,DENUC,XC1,YC1,XV1,YV1,DMUL
COMMON/CL/XC2,YC2
COMMON/H/PINF,P10T,TT0T
COMMON/DS/AL,CL,CZ
COMMON/V/AV,BV,CV,XV,XV2,YV2
COMMON/V/IS/XTXH,YVLFT,XYMOM
COMMON/CP/CP1, RGAS
COMMON/F/IC/CF(50),ST(50)
COMMON/F/PF,TF,AF,THF,EMF,GF
COMMON/CNTIC/TTC,PIC,GC,PC,THC,XNUC,TC,EMC
DIMENSION H0L(11),HOLE(16),DNUE(10)
DATA H0L/2HE0,2HE1,2HE2,2HE3,2HE4,2HE5/
DATA HOL/2HV1,H0C1,2HA,2HB,2HC,2HD,2HV3,2HC3,2HF,2HG,2HV4/
97 FORMAT(15,EB12.4)
98 FORMAT(/)
50 FORMAT(E810.0)
D=0.
EPS=1.E-05
DU 23 L=1.5
AH(L)=0.
BH(L)=0.
23 CL(L)=0.
LH=1
READ(5,100) PI,II,THI,EMI,PINF
READ(5,100) PF,TF,AF,THF,EMF,GF
READ(5,100) XV1,YV1,XV2,XC1,YC1,XC2,DMY,DMUC
READ(5,100) XFI
READ(5,100) XTHX,YVLFT,XYMOM,XTXH,YVLFT,XYMOM,XSHFT,YSHFT
READ(5,5921) 1CF,DXC,J1F,UTH,IVIS,IT
IF(IVIS,EQ.1)READ(5,100)XSTR,REC,SH,RT
IF(IVIS,EQ.4,ANU,IT,EG,0)READ(5,100)AH(1),RH(1),CH(1)
5:21 FORMAT(15,E10.0,15,E10.0,315)
HI=FHI(PI,HI1,TI)
PHI1=AI
GI=FGAM(TI,PI,MI)
RHI=RHEG(THI,PI,MI,DUM)
AI=SQRT(G1*PI/RHI)
UIE=PI*AI
XTX1=XTXH
YFL1=YVLFT
XYMOM=XYMOM
XTX=XTXH
YVLF1=YVLFT
XYMOM=XYMOM
DNUC=DNUC/57.3
THI=THI/57.3
DTH=DTH/57.3
READ(5,100) A2,B2,CZ
ZC1=GETZ(XC1)
ZV1=GETZ(XV1)
ZC2=GETZ(XC2)
ZI=ZV1
ITEM 4
IF (K4) GO TO 54
SIM (R5)) END
166 CONTINUE
IF (KDUM.EQ.I) GO TO 55
NPR2 = 5 * NPR1 + 1
DO 414 K = N IPR2, I MAX
WRITE (6, 550) (CF(K), SI(K))
35 CONTINUE
414 CONTINUE
55 CONTINUE
450 FORMAT (1X, 5(6X, CF *, 9X, ST *, 3X))
500 FORMAT (1X, 10E13.5)
550 FORMAT (105X, 2E13.5)
WRITE (6, 998)
WRITE (6, 1800) XTHX, YLFT, XMOM
WRITE (6, 1900) XVTHX, YVLFT, XVMMOM
IMAX = KMAX
I = I + 1
IF (I.GT.IDU-M) ICWL = 1
IF (ICWL.EQ.0) KMAX = KMAX + 1
IF (IEND.EQ.1) I MTP = KSTP + ICWL
IF (IEND.EQ.1 .AND. KP.EQ.1) GO TO 6000
IF (IEND.EQ.1 .AND. IEXT.EQ.1) GO TO 6000
IF (IEXT.EQ.1 .AND. X(1,1)GE.XC2-EPS) GO TO 5900
IF (ICWL.EQ.1) GO TO 22
GO TO 20
462 CONTINUE
WRITE (6, 992)
992 FORMAT (A1H2*LT. 0*)
GO TO 6000
5900 CONTINUE
IF (P(1,1)LE.PF) GO TO 5396
CALL CUAL (3VUE)
IDUM = 6
IF (2)
XNU = XNU (1,1)
THI = TH (1,1)
ICWL = 0
KMAX = KMAX + 1
DELNU = DNUE/FLUAT (IDUM)
IDUM = IDUM + 1
IEXT = 1
GO TO 20
5396 CONTINUE
6000 CONTINUE
5000 CONTINUE
END
SUBROUTINE CALL (XAX, YAX, XSTP, YSTP)
COMMON /X/X(2,50), Y(2,50), Z(2,50), TH(2,50), XNU(2,50),
        IP(2,50), T(2,50), EM(2,50), XNU(2,50), O(2,50), H(2,50), RH(2,50),
        2G(2,50)
COMMON /C7/ DNU, DNU, XC1, YC1, XV1, YV1, DNUL
D=0,
IADD=0

LOOP=0
XTHM1=TH(1,1)-XNU(1,1)
QI=XNU(1,1)-TH(1,1)
CALL INT(0, X(1,2), Y(1,2), Z(1,2), TH(1,2), XNU(1,2), P(1,2), T(1,2),
        1H(1,2), EM(1,2), G(1,2), O(1,2), RH(1,2), XNU(1,2),
        2D, O, D, D, O, D, D, D,
        2X, Y, Z, THA, XNUA, PA, TA, HA, EMA, GA, UA, RHA, XNUA, -1,1)
Q2=XNU(1,2)-TH(1,2)

44 XTHM2=THA-XNUA
IF (LOOP.EQ.0) XTHM3=XTHM2
SLA=(TAN(XTHM2)+TA/(XTHM3))/2.
SLB=(YC2-YC1)/(XC2-XC1)
CALL GEM(XA-YA, SLA, XC2, YC2, SLB, XC, YC)
IF (XC.LE.XC2) GO TO 43
IADD=1
DX=X(1,2)-X(1,1)
SLU=(Y(1,2)-Y(1,1))/DX
IT=1

X4=(X(1,2)+X(1,1))/2.
24 RAT=(X4-X(1,1))/DX
SLB=XTHM1+RAT*(XTHM2-XTHM1)
SLB=TAN(SLB)
IF (LOOP.EQ.1) SLB=(SLB+TAN(XTHM3))/2.
CALL GEM(X(1,1), Y(1,1), SLU, XC2, YC2, SLB, XAT, YA)
ER=ABS((X4-XAT)/DX)
IF (ER.LT.1.E-04) GO TO 63
IT=IT+1
IF (IT.EQ.10) GO TO 68
X0=XAT
GO TO 24

68 WRITE(6,38)
38 FORMAT(* TOO MANY ITER IN CALL *).
WRITE(6,39) LOOP, SLU, SLB, XA, XAT
39 FORMAT(1X, IS, 5E13.5)
STOP

63 CONTINUE

CALL INT(RAT, X(1,1), Y(1,1), Z(1,1), TH(1,1), XNU(1,1), P(1,1), T(1,1),
        1H(1,1), EM(1,1), G(1,1), O(1,1), RH(1,1), XNU(1,1), X(1,2), Y(1,2),
        2Z(1,2), TH(1,2), XNU(1,2), P(1,2), T(1,2), H(1,2), EM(1,2), G(1,2),
        3O(1,2), RH(1,2), XNU(1,2), XA, YA, ZA, THA, XNUA, PA, TA, HA, EMA, GA, UA,
        RHA, XNUA, -1,1)
43 RA=THA+XNUA
IF (ABS(XC2-XC).LE.0.0001) XC=XC2
X(2,1)=XC
Y(2,1)=YC
Z(2,1)=GETZ(XC)
TH(2,1)=TH(1,1)
XNU(2,1)=RA-TH(2,1)
DNU=XNU(2,1)-XNUA
CALL PM(DNU, P(2,1), T(2,1), H(2,1), EM(2,1), G(2,1), O(2,1), RH(2,1),
        1PA, TA, HA, EMA, GA, UA, RHA)
RQ30=RH(2,1)*XNU(2,1)/Z(2,1)
DNU=RQ30
IT=1
10 CONTINUE
CALL PM(DNU3,P(2,1),T(2,1),M(2,1),EM(2,1),G(2,1),Q(2,1),RH(2,1),
PA,TA,HA,EMA,GA,UA,RHA)
RUT=RH(2,1)*N(2,1)
ERQ=(RQT-RJ30)/RQ30
IF(IIT.GE.10)IERT=.005
IF(ABS(ERQ).LT.ERT)GO TO 20
CALL ERRUR(IIT,DFU3,ERG,1.1,DNUZ1,ERQZ1)
GO TO 10
20 CONTINUE
XMU(2,1)=ASIN(1./EM(2,1))
XTHMS=TH(2,1)-XMU(2,1)
IF(LOOP,EQ.1)GO TO 6
LOOP=1
GO TO 44
6 IF(IAOD.EQ.0)RETURN
IMAX=IMAX+1
KMAX=KMAX+1
KSTP=KSTP+1
ISTP=ISTP+1
DO 46 JJ=J,IMAX
LJ=IMAX-JJ+1
46 CALL SWITCH(1,L1,1,LJ)
WRITE(6,181)IAOD,XA,X(1,1),X(1,2)
181 FORMAT(1X,15,3E15.5)
CALL INT0,XA,YA,IMA,XMUA,PA,TA,HA,EMA,GA,UA,RHA,XMUA,
1P,D,D,D,D,D,D,D,D,D,D,D,D,D,D,X(1,2),Y(1,2),Z(1,2),TH(1,2),XNU(1,2),
2P(1,2),T(1,2),M(1,2),EN(1,2),G(1,2),Q(1,2),RH(1,2),XMU(1,2),-1.)
RETURN
END
SUBROUTINE ENDU(L, KK, K6)

COMMON/V, AV, BV, CV, X, Y, Z, Y2, Y12

COMMON/X(2,50), Y(2,50), Z(2,50), TH(2,50), XNU(2,50),
  IP(2,50), IT(2,50), EM(2,50), XNU(2,50), G(2,50), H(2,50), RH(2,50),

2G(2,50)

IT=1

B=0.

A=1.

XM1=TH(1, L)+XNU(1, L)

XM2=TH(2, KK)+XNU(2, KK)

XM3=0.

Q1=TH(1, L)+XNU(1, L)

Q2=TH(2, KK)+XNU(2, KK)

XA=(X(1, L)+X(2, KK))/2.

DX=X(2, KK)=X(1, L)

SLM=(Y(2, KK)=Y(1, L))/DX

RAT=(XA-X(1, L))/DX

SLP=XM1*RAT*(XA-XM1)

SLP=AXSLP+3*X3

CALL GEM(X(1, L), Y(1, L), SLM, XNU, Y, Z, Y2, YV2, SLP, SLP, XAT, Y)

ER=ABS((XA-XAT)/DX)

IF (ER, LT, 1. E-05) GO TO 14

IT=IT+1

IF (IT, LE, 10) GO TO 8

WRITE(6, 33)

33 FORMAT(* TOO MANY ITER IN END *)

WRITE (6, 34), KK, K6, XA, XAT, X(2, KK), X(1, L), SLP, XV3

STOP

8 XA=XAT

GO TO 12

12 QA=Q1+RAT*(J2-Q1)

CALL INTRAT, X(1, L), Y(1, L), Z(1, L), TH(1, L), XNU(1, L), P(1, L),
  IT(1, L), H(1, L), EM(1, L), G(1, L), Q(1, L), RH(1, L), XNU(1, L), X(2, KK),
  2Y(2, KK), Z(2, KK), TH(2, KK), XNU(2, KK), P(2, KK), T(2, KK), H(2, KK),
  SEM(2, KK), G(2, KK), Q(2, KK), RH(2, KK), XNU(2, KK), XA, YA, ZA, THA, XNUA,

4PA, TA, HA, EMA, GA, UA, RHA, XNUA, 1.)

RA=XNUA-THA

X(2, K6)=XV2

Y(2, K6)=YV2

Z(2, K6)=GETZ(XV2)

TH(2, K6)=A1AN (BV*2, *CV*(XV2-XV))

XNU(2, K6)=RA+TH(2, K6)

DNU=NU(2, K6)-XNUA

CALL PM(DNJ, PT(2, K6), T(2, K6), H(2, K6), EM(2, K6), G(2, K6), G(2, K6),
  1RH(2, K6), PA, TA, HA, EMA, GA, UA, RHA)

RQD3=RH(2, K6)/Z(2, K6)

DNU3=DNU

IT=1

ERT=.001

10 CONTINUE

CALL PM(DNU3, P(2, K6), T(2, K6), H(2, K6), EM(2, K6), G(2, K6), G(2, K6),
  1RH(2, K6), PA, TA, HA, EMA, GA, UA, RHA)

RQT=RH(2, K6)*Q(2, K6)

ERG=(RQT-RQ3D)/RQ3D

IF (IT, GE, 10) ERT=.005

IF (ABS(ERG), LT, ERT) GO TO 16

CALL ERROR(2, IT, DNU3, ERG, 1, 1, DNUZ1, ERGZ1)

GO TO 10

16 XNU(2, K6)=ASIN(1./EM(2, K6))

IEND=1

IF (Q1, GT, 0.) RETURN
SUBROUTINE INT(RAT, X1, Y1, Z1, TH1, XNU1, P1, T1, H1, EM1, G1, U1, RH1, XMU1, X2, Y2, Z2, T42, XNU2, P2, T2, H2, EM2, G2, U2, RH2, XMU2, X3, Y3, Z3, TH3, XNU3, P3, T3, H3, EM3, G3, U3, RH3, XMU3, OPT)

HT1 = H1 + U1 * J1/2.
HT2 = H2 + U2 * J2/2.
Q1 = XNU1 + OPT * TH1
Q2 = XNU2 + OPT * TH2
X3 = X1 + RAT * (X2 - X1)
Y3 = Y1 + RAT * (Y2 - Y1)
G3 = G1 + RAT * (G2 - G1)
P3 = P1 + RAT * (P2 - P1)
T3 = T1 + RAT * (T2 - T1)
XNU3 = XNU1 + RAT * (XNU2 - XNU1)
TH3 = (TH1 - TH2) / OPT
HT3 = HT1 + RAT * (HT2 - HT1)
U3 = U1 + RAT * (U2 - U1)
G3 = G1 + RAT * (G2 - G1)
H3 = H1 + RAT * (H2 - H1)
RH3 = RH1 * (P3/P1) * (1/G3)
EM3 = U3 / SQRT(G3 * P3/RH3)
XMU3 = ASIN(1/EM3)
Z3 = GETZ(X3)
RETURN
END
SUBROUTINE TMH(RH1, UI, PI, HI, TI, EM1, GI, PIVF, YV1, YC1, DYV, X2, THRMAX,
DVU, DNUC, X2)
COMMON/3/AT, HZ, CZ
A1=YV1-YC1
F1=RHI*UI*A1
Z2=AZ*X2**2+HZ*X2+CZ
Z5=AZ*X2**2+HZ*X2+CZ
A2=(Z2+Z5)/2.*DYV
IT=1
DNU=(DVU+DNUC)*SQRT((Z2+Z5)/2.)
10 CALL PM(DVU, P2, T2, H2, EM2, CZ, UI, RH1)
F2=RH2*U2*A2
ERT=(F2-F1)/F1
IF(ABS(ERT).LT.1.E-04) GO TO 20
CALL ERROR(99, IT, DNU, ERT, 9, DNU1, ERT1)
GO TO 10
20 TH1=(PI-PIVF+RHI*UI*UI)*A1
TH2=(PI-PIVF+RHI*UI*UI)*A2
THRMAX=TH2-TH1
RETURN
END
SUBROUTINE EM3D(EM,EMX,X,G,Z,I)

GN=(G+1.)/(G-1.)/2.
Y=(1.+(G-1.)/2.*EM*EM)**GN
Y=Y/EM/((G+1.)/2.)*GN
A=Y*Z/ZI
EMX=EM*SORT(Z)

ITM=1

10 AT=(1.+(G-1.)/2.*EMX*EMX)**GN
AT=AT/EMX/((G+1.)/2.)*GN
ERM=(AT-A)/A

IF(ABS(ERM).LT.1.E-03) GO TO 20
CALL ERROR(100,ITM,EMX,ERM,1,1,EMX1,ERM1)
GO TO 10

20 CONTINUE
RETURN
END
SUBROUTINE SWITCH(I2,K,I1,L)

COMMON/X/X(2,50),Y(2,50),Z(2,50),TH(2,50),XNU(2,50),
   1P(2,50),T(2,50),E(2,50),XMU(2,50),U(2,50),H(2,50),RH(2,50),
   2G(2,50)

X(I1,L)=X(I2,K)
Y(I1,L)=Y(I2,K)
Z(I1,L)=Z(I2,K)
TH(I1,L)=TH(I2,K)
XNU(I1,L)=XNU(I2,K)
P(I1,L)=P(I2,K)
T(I1,L)=T(I2,K)
H(I1,L)=H(I2,K)
G(I1,L)=G(I2,K)
RH(I1,L)=RH(I2,K)
EM(I1,L)=EM(I2,K)
XMU(I1,L)=XMU(I2,K)
U(I1,L)=U(I2,K)
RETURN
END
FUNCTION GETZ(X)
COMMON/D3/ AZ, BZ, CZ
GETZ = AZ**2 + BZ*X + CZ
RETURN
END
SUBROUTINE VMAIL (IMAX,KMAX)
COMMUN/X/X(2,50),Y(2,50),Z(2,50),TH(2,50),XNU(2,50),
IP(2,50),T(2,50),EM(2,50),XNU(2,50),G(2,50),O(2,50),RH(2,50),
2G(2,50),
COMMUN/V/AV,sv,CV,XV,XV2,YV2,
COMMUN/C/DNUV,DNU,C,JC1,YC1,XV1,YV1,DNU
IBEG=0
IT=1
SL2=TH(1,1MAX)
SL1=TAN(TH(2,KMAX-1)+XNU(2,KMAX-1))
SL13=SL1
CALL GEM(X(2,KMAX-1),Y(2,KMAX-1),SL1,X(1,IMAX),Y(1,IMAX),SL2,
IXG,YG)
12 Y=R=AV+BV*(XG-XV)+CV*(XG-XV)*^2
ER=-(Y-XG)/(YV1=YC1)
IF(ABS(ER)^LT.1,E=02) GO TO 12
CALL ERROR(S,IT,XG,ER,9,XTG,ERW)
YG=Y(2,KMAX-1)+SL1*Y(XG-X(2,KMAX-1))
10 TH(2,KMAX)=ATAN(BV+CV*XG(XG-XV))
XNU(2,KMAX)=TH(2,KMAX)-TH(2,KMAX-1)+XNU(2,KMAX-1)
X(2,KMAX)=XG
Y(2,KMAX)=YG
Z(2,KMAX)=SETZ(XG)
K=KMAX
M=KMAX-1
DNU=XNU(2,K)-XNU(2,M)
CALL PM(DNU,P(2,K),T(2,K),H(2,K),EM(2,K),G(2,K),O(2,K),RH(2,K),
1P(2,M),T(2,M),H(2,M),EM(2,M),G(2,M),O(2,M),RH(2,M))
RQ3D=RH(2,K)*Q(2,K)/Z(2,K)
DNU=DNU
IT=1
ERT=.001
15 CONTINUE
CALL PM(DNU3,P(2,K),T(2,K),H(2,K),EM(2,K),G(2,K),O(2,K),RH(2,K),
1P(2,M),T(2,M),H(2,M),EM(2,M),G(2,M),O(2,M),RH(2,M))
RQT=RH(2,K)*U(2,K)
ERQ=TRUT=30/7/R30
IF(11,GE.10)ERT=.005
IF(ABST(ER)^LT.1,ERT)GO TO 10
CALL ERROR(4,IT,DNU3,ERQ,1,1,DNUZ1,ERQ1)
GO TO 15
10 CONTINUE
XNU(2,K)=%K1(1./EM(2,K))
IF(IBEG.EQ.1) RETURN
IBEG=1
SL1=SL1+TAN(TH(2,KMAX)+XNU(2,KMAX))
GO TO 12
END
SUBROUTINE GEN(XA,YA,SLA,XH,YH,SLB,XC,YC)
XC=(YB-YA+XA*X*A-SLB*XB)/(SLA-SLB)
YC=YA+SLA*(XC-XA)
RETURN
END
SUBROUTINE ERROR (I, IT, X, ER, E, X1, ER1)
    IT = IT + 1
    IF (IT .LE. 15) GO TO 12
    WRITE (6, 13)
    13 FORMAT (*ERROR TEST NUMBER *)
    WRITE (6, 20) I
    20 FORMAT (I5)
    STOP

12 IF (IT .GT. 2) GO TO 14
    ER1 = ER
    X1 = X
    X = X + F
    IF (X .EQ. X1) X = X + .02
    RETURN

14 X0 = X1 - ER1 * (X - X1) / (ER - ER1)
    ER1 = ER
    X1 = X
    X = X0
    RETURN
END
SUBROUTINE FIX(I,K1,K2,K3)
COMMUT/X/X(2,50),Y(2,50),Z(2,50),IM(2,50),XNU(2,50),
IP(2,50),TP(2,50),EM(2,50),XNU(2,50),G(2,50),H(2,50),RH(2,50),
28(2,50)
DUM3=0.
DUM4=0.
A=1.
B=0.
DUM1=TAN(T(I,K1)+XNU(I1,K1))
DUM2=TAN(T(I2,K2)+XNU(I2,K2))
XNU(I3,K3)=.5*(XNU(I1,K1)+XNU(I2,K2)+TH(I2,K2)-TH(I1,K1))
TH(I3,K3)=.5*(XNU(I2,K2)+XNU(I1,K1)+TH(I2,K2)+TH(I1,K1))
10 IF (B.GT.0.) DUM3=TAN(TH(I3,K3)+XNU(I3,K3))
IF (B.GT.0.) DUM4=TAN(TH(I3,K3)-XNU(I3,K3))
SL1=A*DUM1+B*DUM2
SL2=A*DUM2+B*DUM3
CALL GEN(X(I1,K1),Y(I1,K1),SL1,X(I2,K2),Y(I2,K2),SL2,X(I3,K3),
1Y(I3,K3))
Z(I3,K3)=GETZ(X(I3,K3))
DNU=XNU(I3,K3)-XNU(I1,K1)
CALL PM(DNU,P(I3,K3),T(I3,K3),H(I3,K3),EM(I3,K3),G(I3,K3),
1Q(I3,K3),R(I3,K3),P(I1,K1),T(I1,K1),H(I1,K1),EM(I1,K1),
2G(I1,K1),W(I1,K1),RH(I1,K1))
R3D=RH(I3,K3)*Q(I3,K3)/7(I3,K3)
ERT=.001
DNU3=DNU
IT=1
12 CONTINUE
CALL PM(DNU3,P(I3,K3),T(I3,K3),H(I3,K3),EM(I3,K3),G(I3,K3),
1Q(I3,K3),R(I3,K3),P(I1,K1),T(I1,K1),H(I1,K1),EM(I1,K1),
2G(I1,K1),W(I1,K1),RH(I1,K1))
ROT=RH(I3,K3)*Q(I3,K3)
ERQ=(ROT=R3D)/R03D
IF (IT.GE.10) ERT=.005
IF (ABS(ERQ).LT.ERT) GO TO 20
CALL ERROR(5,17,DNU3,ERQ,1.1,DNU3,ERQZ1)
GO TO 12
20 XNU(I3,K3)=ASIN(1./EM(I3,K3))
IF (B.GT.0.) RETURN
A=.5
B=.5
GO TO 10
END
SUBROUTINE PM(DNU,P2,T2,H2,E2M2,G2, U2, RH2, PI, T1, HI, EMI, GI, U1, RHI)
COMMON/V,J, PHI1

DTH=1./57.3
IFAN=ABS(DNU)/DTH+1
DUM=DNU
IF(IFAN.GT.1)DUM=DNU/FLGAT(IFAN=1)
P1=PI
T1=TI
EM1=EMI
HI=HI
G1=GI
RH1=RHI
A1=SQRT(G1*P1/RH1)
U1=Ul
U1=U1*U1
HT=HI+U1/2.
P1=ALOG(P1)
XNU=0.
XNU1=0.
IF(IFAN.NE.1)IFAN=IFAN-1
DO 10 I=1,1,FAN
XNU1=ASIN(1./EM1)
B1=G1*EM1/COS(XNU1)
P2=HI*DUM+P1
RH2=(P2-P1)/G1
RH2=RH1*EXP(RH2)
P2P=EXP(P2)
P1P=EXP(P1)
U2=U1*2.*G1/(G1-1.)*(P2P/RH2-P1P/RH1)
H2=HI*U2/2.
T2=FT(P2P,PHI1,H2)
G2=FGAM(T2,P2P,PHI1)
A2=G2*P2P/RH2
EM2=SQRT(U2/A2)
XNU=XNU1+DJM
XNU1=XNU
P1=P2
T1=T2
G1=G2
EM1=EM2
U1=U2
RH1=RH2
10 CONTINUE
P2=EXP(P2)
U2=SQRT(U2)
RETURN
END
END
SUBROUTINE PMI (IX, KX)

COMMON/IP,P1,IL,ILX,THI,PRM,rais,HI,RI,UL

COMMON/X/X(2,50),Y(2,50),Z(2,50),TH(2,50),XNU(2,50),

1P(2,50),T(2,50),EM(2,50),XNU(2,50),O(2,50),H(2,50),RN(2,50).

G(2,50)

COMMON/P/NF,P1JT,TTNT

COMMON/CP/CPI,

RGAS

DNUN=XNU(IX,KX)

G=GI

EM1=EMI

GG=SRT((G+1.)/(G-1.))

XM1=SRT(EM1**2-1.)

XNU1=GG*ATAN(XM1/GG)-ATAN(XM1)

EM2=DNUN/(1.5-XNU1)*(6.-EM1)+EM1

IT3=1

XM2=SRT((EM2**2-1.)

DNUN=GG*(ATAN(XM2/GG)-ATAN(XM1/GG))+ATAN(XM1)-ATAN(XM2)

ERNU=DNUN-DNUT

IF(ABS(ERNU).LT.1.E-04) GO TO 20

CALL ERROR(6,IT3,EM2,ERNU,1.11,EM21,ERNU1)

GO TO 10

20 CONTINUE

CALL E450(EM2,EMX,X(IX,KX),G,Z (IX,KX),Z1)

P(IX,KX)=PI(1.+G-1.)/2.*EMX**2)**(G/(G-1.))

TI(IX,KX)=TJ1/(1.+G-1.)/2.*EMX**2

EM(IX,KX)=EMX

XNU(IX,KX)=ASIN(1./EMX)

AX=SRT(G*RGAS*1(IX,KX))

Q(IX,KX)=EMX*AX

RETURN

END
SUBROUTINE GNURE(RH,Q,P,T,H, X,X1,CF,ST,L)
COMMON/CP/CPI, RGAS
COMMON/D/PHI
COMMON/HOT/AH(3),BH(3),CH(3),XSTR, REC, RT,S,H,IT,IVIS
HDEL=R@G*0.4/2.
HAH=1. +REC*G*Q/2. /HDEL
IF(IT .EQ.0)TW=AH(L)*(X-X1)**2+BH(L)*(X-X1)+CH(L)
IF(IT .EQ.1)GO TO 46
HW=FH(P,PH11,TA)
HW=HW/HDEL
GO TO 46
46 MW=HAH
A=HAW=1.
B=HA=1.
C=SQRT((A+B)**2+4.**A)
FC=A/(ASIN((A-H)/C)+ASIN((A+H)/C))**2
FRX=HW**2/(FC*(HW)**(1.474))
CALL VISIT(X,XMU)
REX=RH*Q*(X+XSTR)*RT /XMMU
REXI=FRX*REX
CFI=.08*(ALOG10(REXI)-2.3686)/(ALOG10(REXI)-1.5)**3
CF=CFI/FC
ST=CF*SH/2.
RETURN
END
SUBROUTINE VISIT(X,XMU)
XMU=2.27*X**1.5*1.2-08/(1+198.6)
RETURN
END
SUBROUTINE SUBST (XI, YI, ZI, XP, X2, Y2, Z2, X3, Y3, Z3, X4, Y4, Z4, AVX, AVY, AVZ,
1NNX, NNY, NZ0, AS, X0, Y0, Z0, L1)
D1=NSIG(XP(4), YP(4), ZP(4), XI(4), ETA(4))
XPA(1)=XI
YPAP(1)=YI
ZPA(1)=ZI
ZPA(2)=Z2
YPA(2)=Y2
XPA(2)=X2
XPA(3)=X3
YPA(3)=Y3
ZPA(3)=Z3
ZPA(4)=Z4
YP(4)=Y4
XPA(4)=X4
T1X=X3-X1
T1Y=Y3-Y1
T1Z=Z3-Z1
T2X=X4-X2
T2Y=Y4-Y2
T2Z=Z4-Z2
XNX=T2Y*T1Z-T1Y*T2Z
XNY=TI+X*T2Z-T2X*T1Z
XNZ=T2X*T1Y-T1X*T2Z
VNA=SURF (XNX*X+XY*X+XNZ+Z)
IF (VNA.LE.1.E-13) GO TO 6
XNX=XNX/V
XNY=XNY/V
XNZ=XNZ/V
D=XNX*(AVX-X1)+XNY*(AVY-Y1)+XNZ*(AVZ-Z1)
PD=ABS(D)
T=SQRT(T1X*T1X+T1Y*T1Y+T1Z*T1Z)
T1X=X1+T1Y/T
T1Y=Y1+T1Z/T
T1Z=Z1+T1Y/T
T2X=XNY*T1Z-XNZ*T1Y
T2Y=XNZ*T1X-XNY*T1Z
T2Z=XNX*T1Y-XNY*T1X
DO 1000 J=1, 4
XPA(J)=XPA(J)+XNX*D
YP(J)=YP(J)+XNY*D
ZPA(J)=ZPA(J)+XNZ*D
D=D+D
XDIFF=XPA(J)-AVX
YDIFF=YP(J)-AVY
ZDIFF=ZPA(J)-AVZ
XI(J)=T1X+XDIFF+T1Y*YDIFF+T1Z*ZDIFF
1000 ETA(J)=T2X*XDIFF+T2Y*YDIFF+T2Z*ZDIFF
XIO=(XI(4)*ETA(1)+ETA(2)+XI(2)*ETA(4)+ETA(1))/ETA(2)-ETA(4)
1/3.
ETA0=ETA(1)/3.
DO 1020 J=1, 4
XI(J)=XI(J)-XI0
1020 ETA(J)=ETA(J)-ETA0
XD=AVX*XI+T1X*XI0+T2X*ETA0
YD=AVY*XI+T1Y*XI0+T2Y*ETA0
ZD=AVZ*XI0+T1Z*XI0+T2Z*ETA0
AS=ETA(2)-ETA(4)*(XI(3)-XI(1))/2.
AS=ABS(AS)
RETURN
6 CONTINUE
X=AVX $ Y=AVY $ Z=AVZ
SUBROUTINE: TRAX(X1, Y1, Z1, X2, Y2, Z2, X3, Y3, Z3, X4, Y4, Z4, P1, P2, P3, P4, TH1, TH2, TH3, TH4, X1, X2, X3, X4, AVG)

COMMON/D/PHI1
COMMON/P/PIVF, PIVT, TOT
COMMON/CP/CP, RGAS
COMMON/SHIFT/XSHIFT, YSHIFT
COMMON/MOTION/AVH, BH, CH, XSTR, REC, RT, SH, IT, IVIS

COMMON/USF/XTXH, YLVFT, X40
P=XX11*1+XX2*2+XX3*3+XX4*4
Q=XX1*1+XX2*2+XX3*3+XX4*4
T=XX1*1+XX2*2+XX3*3+XX4*4
TH=XX1*1+XX2*2+XX3*3+XX4*4
H=FH(P, PHI1, T)

RH=RH(EH9, P, PHI1, DUL)
R=H+Q*Q/2.
R=CP1*TH+U*3/2.
AVX=XX1*1+XX2*2+XX3*3+XX4*4
AVY=XX1*1+XX2*2+XX3*3+XX4*4
AVZ=XX1*1+XX2*2+XX3*3+XX4*4

CALL SNAVF(x1, y1, z1, x2, y2, z2, x3, y3, z3, x4, y4, z4, AVX, AVY, AVZ, XNX, YNY, XNZ, ASS, XOR, Y0, Z1)

CFF=0.
XBP=0.

IF (IVIS.EQ.1) CALL GNUMR(RH, U, F, T, R, XO, XBP, CFF, ST, 1 )

RH=RH(U*Q/2.
P=AVF*PIVF
DXTX=PAV*XNY*ASS
DYLF=PAV*XNY*ASS
XNZ=1.

IF (LH.EQ.3) XNZ=2.

DXTXV=CFF *COS(TH)*ASS*RH

DYLFTV=CFF *SIN(TH)*ASS*RH

DYLFTV=DLYFTV*AVG

XMS=XU*XSHIFT

YMS=YU*YSHIFT

DMUV=MUV*DXTXV*XMS*DYLFTV

XVTH=XVTH+DXTXV

YVLFT=YLFT+DYLFTV

XVMON=XVMON+DMUV

DXTX=DXTX*AVG+DXTXV

DYLFT=DYLFT*AVG+DYLFTV

DMUV=MUV*DXTXV*XMS*DYLFT

RETURN

END
SUBROUTINE CONL(DNUE)
COMMON/X/X(2,50), Y(2,50), Z(2,50), TH(2,50), XNU(2,50),
J(2,50), T(2,50), EM(2,50), XM(2,50), O(2,50), H(2,50), RM(2,50),
2G(2,50)
COMMON/CM/XC2, YC2
COMMON/P/PVF, PDOT, I10T
COMMON/P/PF, PF, IF, AIF, THF, EMF, GF
COMMON/C/CP, C, RGAS
COMMON/CNT/TC, PTC, MC, GC, PC, THC, XNUC, ZC, EMC
PCC=(PF+P(1,1))/2.
GM1=GF-1.
GP1=GH+1.
EMF2=EMF*EMF
DUM=1.+G41*EMF2/2.
PTF=PF*DUM**GM1
ITF=TF*DUM
GIS=GI
XNU=ASIN(1./EMF)
IT=1
BET=(THF-X4UF+TH(1,1))*1.1=THF
20 DUM=(EMF*SIN(BET))**2
PC=(2.*GF*DUM-GM1)/GP1
TC=PC*(G41*DUM+2.)/DUM/GP1
EMC=(EMF2*(GP1*PC+GM1)-2.*(PC*PC-1.))/GM1*PC+GP1/PC
EMC=SWRT(WC)
PC=PC*PF
THC=TAN(BET)*(GP1*EMF2/(DUM-1.)/2.-1.)
THC=ATAN(1./THC)+THF
PTC=(2.*TC/(G41*EMF2+2.))*GM1
PTC=PC/PTC
TC=TF
ITC=TC*TF
RA=X4UF(1,1)+TH(1,1)
X(2,1)=XG2 SY(2,1)=YC2 $TH(2,1)=THC
Z(2,1)=GETZ(XG2)
XNU(2,1)=RA=TH(2,1)
DNU=XNU(2,1)-XNU(1,1)
CALL P4D0+uP(2,1), T(2,1), H(2,1), EM(2,1), G(2,1), 0(2,1), RM(2,1),
1P(1,1), I(1,1), TH(1,1), EM(1,1), GM1, 1, O(1,1), RM(1,1))
ER4=(PC-P(2,1))/PCC
IF(AABS(ER4).LT.1.E-03)GO TO 16
CALL ERROR(9, IT, BET, ER4, 1.05, BET1, ER41)
GO TO 20
16 DNUE=XNU(2,1)-XNU(1,1)
GI=GIS
MC=GF
GC=GF
ZC=Z(2,1)
XNUC=0.
RETURN
END
FUNCTION F(H(P1,F,11)
P=P*1*1.01325E+05/2116.
T=117.58
F2=F*F
IF(T.LT.0.) GO TO 400
IF(T.GT.2000.) GO TO 190
IF(F.GT.1.) GO TO 191
120 A=1.0-07*(1.042*F2 +.8242*F.+9.87)
B=0.001*(-.01167*F2 +.1503*F+.938)
C=-0.0284*F2 +.0731*F+.4293
GO TO 290
190 A=1.0-07*(1.787*F2 -5.48*F+5.4)
B=0.001*(-.1867*F2 +1.11*F+.176)
C=-0.0933*F2 +3.975*F=2.408
GO TO 290
190 IF(F.GT.1.) GO TO 192
A=0.000001*(1.792*F2+.3983*F+.31)
B=0.001*(-.905*F2 -.07917*F+.245)
C=10.86*F2 -.1183*F+.97
GO TO 290
190 A=0.000001*(4.51*F2 -13.9*F+11.59)
B=0.001*(-23.08*F2 +66.82*F-52.61)
C=27.05*F2 -73.73*F+58.59
290 H1=A*T*T+B*T+C
IF(T.LE.2000.) GO TO 370
A10=ALOG(P)/2.3*5.
29=A10*A10*10*10*10
H1=H1*1.0+f*(T/2000.-1.)*Z9
70 H1=H1*1.0+f*6
GO TO 340
400 T2=T1*1
T3=T2*1
T4=T3*1
T5=T4*1
H1=H1*8314./xMM1
50 CONTINUE
FH=H1*10.7639
RETURN
END
FUNCTION FT(P1, F, H5)
DATA I63/0/.
IF(LG=0)
P=P1*1.01325E+05/2116.
H=H5 /10.7639/1.E+06
F2=F*F
A10=ALOG(P)/2.3-5.
Z9=.125*A10*A10 -.275*A10
IT=1
IF(I63.EQ.1) GO TO 1000
I63=1
T=1500.
T0=1500.
IF(F,GE,0.) GO TO 120
T=600.
T0=T
000 CONTINUE
IF(F,GT,0.) GO TO 400
GO TO 120
50 E0=(H-H1)/H
IF(ABS(E0).LT.1.E-04) GO TO 340
500 T=TO*1.1
502 IT=2
IF(F,LT,0.) GO TO 400
GO TO 120
100 E1=(H-H1)/H
IF(ABS(E1).LT.1.E-04) GO TO 340
IT=IT+1
IF(11,LT,21) GO TO 10
IF(ABS(T-2000.).LT.10.) GO TO 830
WRITE(6,031) P1, H5, T
STOP
130 IF(I&AG,E,1) GO TO 504
IF: AG=1
T0=2000.
T=2000.
IF(F,LT,0.) GO TO 400
GO TO 120
04 WRITE(6,11) E1
11 FORMAT(* TEMPERATURE IN F1 SET TO 2000 = ERROR = *E13.5)  GO TO 340
10 T9=TO-F0*(1-T0)/(E1-E0)
05 E0=E1
TO=T
T=T9
IF(F,LT,0.) GO TO 400
20 A1=1.E-07*(-.1042*F2 +.8242*F+.987)
B=.001*(-.01157*F2 +.1503*F+.938)
C=-.0284*F2 +.6731*F+.4293
IF(F,LE,E,1.) GO TO 190
A=1.E-07*(.1787*F2 -.548*F+.64)
B=.001*(-.1807*F2 +.111*F+.176)
C=-.0933*F2 +3.975*F-.2608
10 IF(T,LE,2000.) GO TO 290
A=.000001*(-.792*F2 +.3983*F+.31)
B=.001*(-.905*F2 -.07917*F+.245)
C=10.86*F2 -1.183*F+.97
IF(F,LE,E,1.) GO TO 290
A=.000001*(-.481*F2 -.13.9*F+.11.59)
B=.001*(-.23.08*F2 +66.82*F-57.61)
C=.000001*(-.857*F2 -.13.9*F+.31)
IF (T.LT.2000.) GO TO 370
    H1=H1*(1.+F.*((T/2000.)-1.)*29)
  370 CONTINUE
  GO TO 350
  400 T2=1.*T
  T3=T2*T
  T4=T3*T
  T5=T4*T
  IF (F.LT.-1.5) GO TO 450
  XMM1=16.043
  A1=4.2497678
  A2=-9.9126562E-03
  A3=3.1602134E+05
  A4=-2.9215432E-08
  A5=9.510350E-12
  A6=-1.0166326E+04
  GO TO 460
  450 CONTINUE
  A1=1.120246
  A2=1.3905710E+02
  A3=2.0563746E-06
  A4=-1.1560272E-08
  A5=5.2385929E-12
  A6=5.3328896E+03
  XMM1=26.054
  H1=H1*8314.7*XMM171.E+06
  350 IF (T.EQ.1) GO TO 100
  100 T0=T
  F1=T1*1.8
  RETURN
  END
FUNCTION HREOG(H,P1,F,T)
T1=FT(P1,F,H)
T=T1/1.8
P=P*1.013256+05/2116.
IF(F.LT.0.) GO TO 2260
FNM=1.53*F+F-5.895*F+28.965
FNN=1.6*F+F-10.6*F+33.6
IF(T.GT.2000.) GO TO 2030
XM=FNM
IF(F.LT.1.) GO TO 2160
XM=FNN
GO TO 2160
2030 FF=F*F
A=-2.3*FF+4.01*F+1.736
B=8.81*FF-15.92*F+5.66
C=26.86*FF+33.21*F+14.58
XN=-.4375*FF+.0255*F+2.08
D=A+(ALOG(P)/2.3)**1.5+b*(ALOG(T)/2.3)+C
XM=FNM-D*((T-2000.)/1000.)*XN
IF(F.LT.1.) GO TO 2160
A=-.022*FF+2.363*F+1.905
B=2.76*FF-7.56*F+8.68
C=-5.6*FF+7.36*F+27.15
XN=-.47*FF+.025*F+.35
D=A+(ALOG(P)/2.3)**1.5+b*(ALOG(T)/2.3)+C
XM=FNN-D*((T-2000.)/1000.)*XN
GO TO 2160
2260 KF=F-.5
IF(KF.EQ.-1.) XM=16.043
IF(KF.EQ.-2.) XM=28.054
2160 HREOG=P*X**1/8314.3*6.2428E-02/32.174
T=1.*1.8
RETURN
END
FUNCTION FGAM(T1,P1,F)
T=T1/1.8
T2=T1*T
P=P1*1.01425E+05/2116.
XM=0.
IF(F.LT.0.) GO TO 550
IF(T.LE.1000.) GO TO 440
XM=-2.15E-08*T2 +.00091*T-.0695
440 XM=-0.09*T2 -.00002*T-.019
IF(F.LE.1.) GO TO 470
470 G=-1.833E-07*T2 +.000075*T+1.367
IF(T.LT.500.) GO TO 520
G=2.E-08*T2 -.000138*T+1.423
IF(T.LT.2000.) GO TO 520
G=7.267E-09*T2 -.000457*T+1.85
20 G=G+XM*(ALJG(P)/2.3-5.)+XM*(F-1.)
GO TO 530
50 T3=T2*T
T4=T3*T
CP=A1+A2*T+A3*T2+A4*T3+A5*T4
G=CP*(CP-1.)
30 CONTINUE
FGAM=G
RETURN
END