

General Disclaimer

One or more of the Following Statements may affect this Document

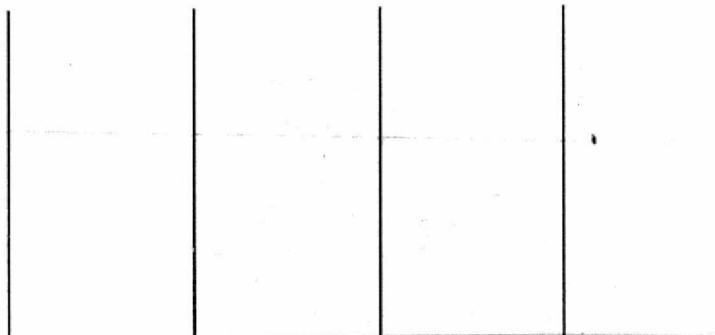
- This document has been reproduced from the best copy furnished by the organizational source. It is being released in the interest of making available as much information as possible.
- This document may contain data, which exceeds the sheet parameters. It was furnished in this condition by the organizational 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.

(NASA-CR-132695) 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

N75-29359

Unclas
32395

CSCI 20D G3/34



Advanced Technology Laboratories inc.

MAY 1975

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

INDEX

	<u>Page</u>
I. INTRODUCTION	1
II. NUMERICAL PROCEDURES	3
A. IDEAL GAS GRID POINT CALCULATION	3
B. EQUILIBRIUM FLOW GRID POINT CALCULATION	7
III. DETERMINATION OF NOZZLE FLOW FIELDS	9
IV. THRUST, LIFT AND PITCHING MOMENT	16
V. VISCOUS EFFECTS	20
VI. CONCLUSIONS	22
APPENDIX I PROGRAM DESCRIPTION	
APPENDIX II LISTING OF FROZEN FLOW PROGRAM	
APPENDIX III LISTING OF EQUILIBRIUM FLOW PROGRAM	

TR 215
LIST OF FIGURES

	<u>Page</u>
FIG. 1. TYPICAL NOZZLE CONFIGURATION	4
FIG. 2. GRID POINT CALCULATION	5
FIG. 3. VARIATION OF VEHICLE EXPANSION WAVE STRENGTH	5
FIG. 4. PARTIAL REFLECTION OF COWL EXPANSION/NO REFLECTION OF VEHICLE EXPANSION	11
FIG. 5. PARTIAL REFLECTION OF COWL AND VEHICLE EXPANSIONS	11
FIG. 6. COMPLETE REFLECTION OF COWL EXPANSION/PARTIAL REFLECTION OF VEHICLE EXPANSION	11
FIG. 7. PARTIAL REFLECTION VEHICLE EXPANSION/COMPLETE REFLECTION OF COWL EXPANSION	12
FIG. 8. COMPLETE REFLECTION OF COWL AND VEHICLE EXPANSIONS	12
FIG. 9. INTERACTION OF EXTERNAL FLOW WITH VEHICLE UNDERSURFACE	12
FIG. 10. THRUST VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION ANGLE	14
FIG. 11. LIFT VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION	15
FIG. 12. THRUST, LIFT, MOMENT	16
FIG. 13. ELEMENTAL SURFACE AREA	17
FIG. 14. UPPER OR LOWER SURFACE AREA ELEMENTS	19
FIG. 15. SIDEWALL AREA ELEMENTS	19

LIST OF FIGURES (Continued)

	<u>Page</u>
FIG. 16 GEOMETRIC INPUTS	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 *á 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 under surface 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.

TR 215
SECTION II
NUMERICAL PROCEDURES

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

$$C_{\pm} = v \mp \theta \quad (1)$$

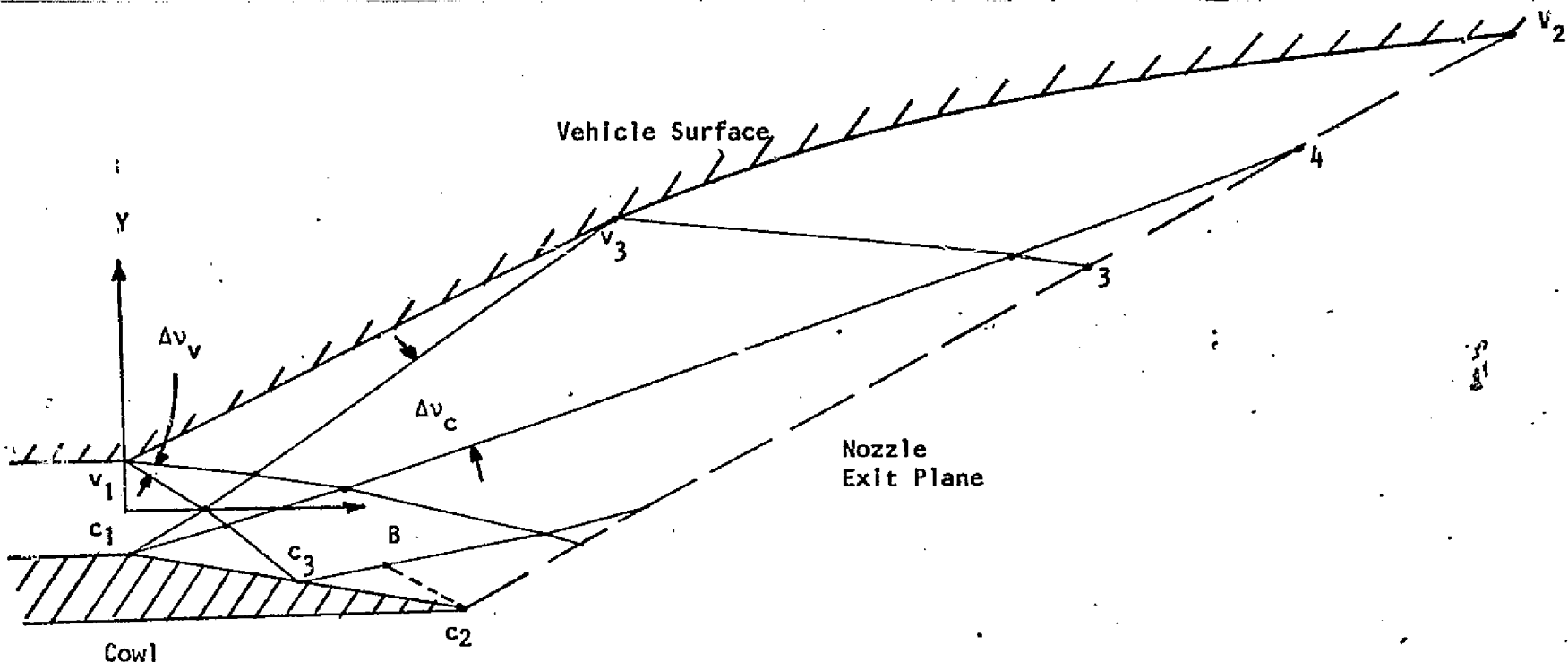
where v is the Prandtl-Meyer function and θ the local flow deflections. Then at point 3

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

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

Employing the two dimensional value of expansion Δv_3 (from initial condition i) the Mach number M_3 is obtained via the Prandtl-Meyer relation (where i denotes uniform initial flow properties at the burner exit)

$$\begin{aligned} \Delta v_3 = v_3 - v_i &= \sqrt{\frac{\gamma+1}{\gamma-1}} \left(\tan^{-1} \sqrt{\frac{\gamma-1}{\gamma+1}} (M_3^2 - 1) - \tan^{-1} \sqrt{\frac{\gamma-1}{\gamma+1}} (M_i^2 - 1) \right) \\ &- \left(\tan^{-1} \sqrt{M_3^2 - 1} - \tan^{-1} \sqrt{M_i^2 - 1} \right) \end{aligned} \quad (3)$$



*(Note that Δv is the wave strength of an expansion fan, not the geometric angle between initial and final rays.)

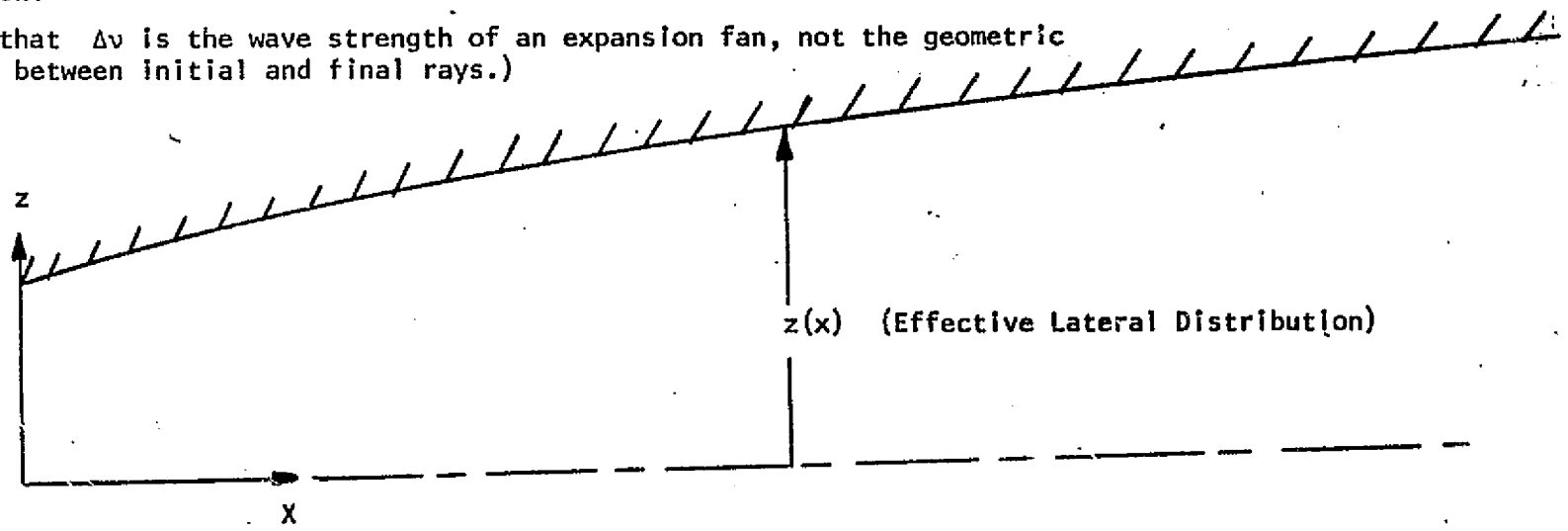


FIGURE 1. TYPICAL NOZZLE CONFIGURATION

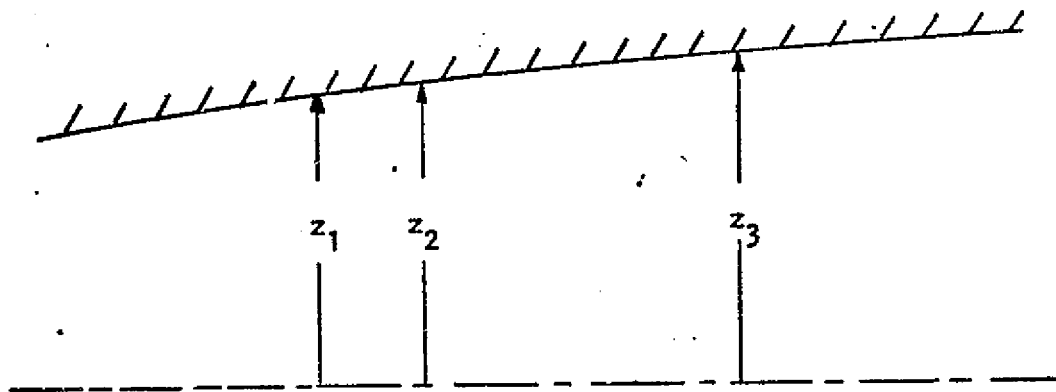
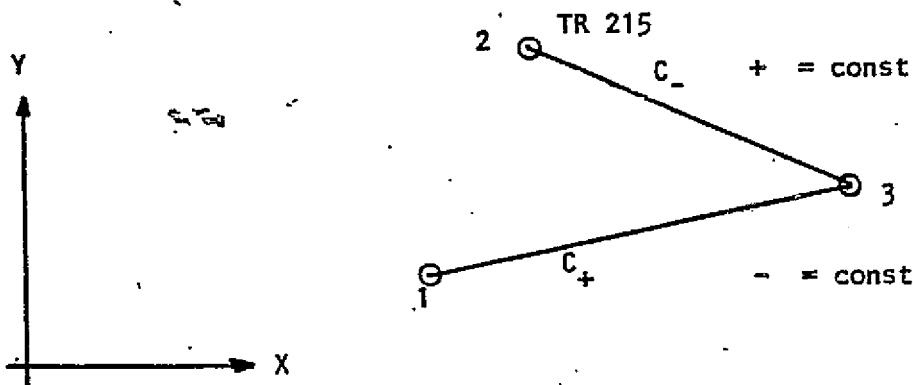


FIGURE 2. GRID POINT CALCULATION

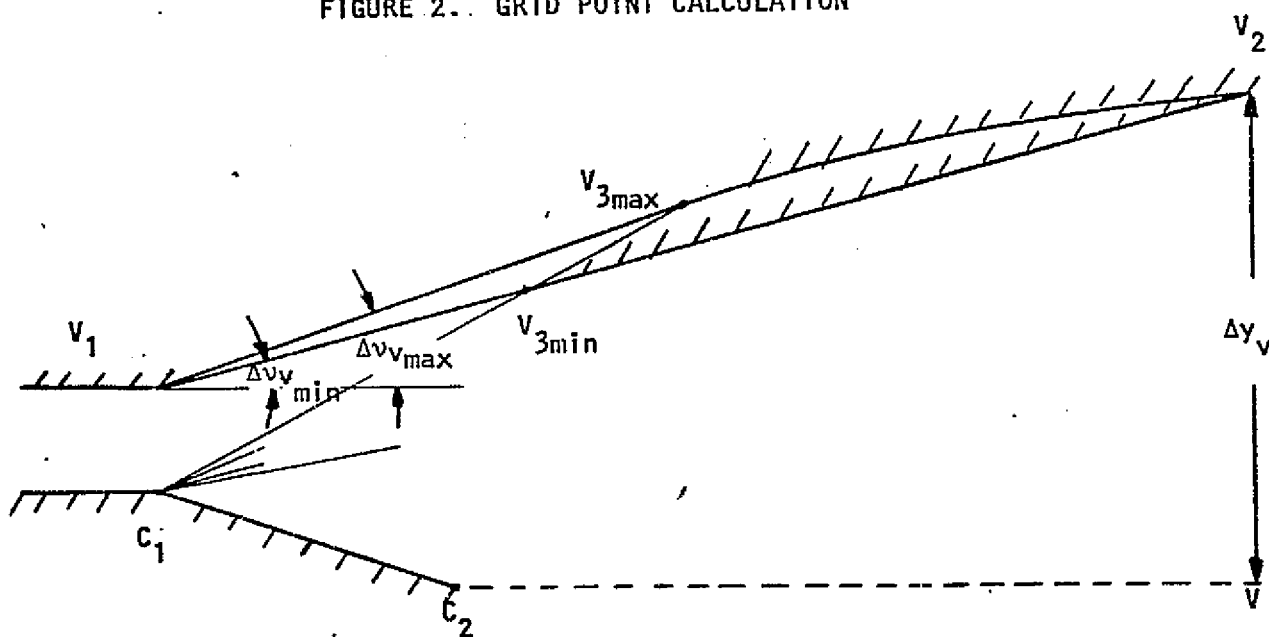


FIGURE 3. VARIATION OF VEHICLE EXPANSION WAVE STRENGTH

employing an iterative procedure to solve this transcendental equation for M_3 . Then, with the Mach angle determined

$$\mu_3 = \sin^{-1} \left(\frac{1}{M_3} \right) \quad (4)$$

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.

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

This ratio is corrected for lateral expansion by multiplying it by the ratio Z_3/Z_i 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_i denotes the lateral extent of the nozzle at the initial station. The location $(x,y)_3$ is determined from

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

then

$$\left(\frac{A}{A^*} \right)_3 = \left(\frac{A}{A^*} \right)_3 * Z_3/Z_i \quad (8)$$

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 μ_{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 γ expansions from initial conditions.

B. Equilibrium Flow Grid Point Calculation - The geometric location of point 3 is obtained employing 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 Δv_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

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

applied across the interval Δv_j yields the pressure, holding μ and Γ equal to their value at the start of the increment. The density is obtained from the isentropic pressure-density relation

$$P/\rho^\Gamma = \text{constant} \quad (11)$$

The velocity is obtained from the Bernoulli relation;

$$\frac{dP}{\rho} + \frac{1}{2} dV^2 = 0 \quad (12)$$

the enthalpy from the constancy of stagnation enthalpy;

$$h + \frac{1}{2} V^2 = H = \text{constant} \quad (13)$$

and the Mach number from;

$$M = V/a; \quad a = \left(\frac{\Gamma P}{\rho}\right)^{\frac{1}{2}} \quad (14)$$

where Γ has been reevaluated employing Equation (9). This procedure is repeated in small steps Δv_j until the full wave Δv_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

$$\left(\frac{A_3}{A_i}\right)_{2D} = \rho_i V_i / \rho_3 V_3 \quad (15)$$

Since the effective area based on three dimensional considerations is

$$\left(\frac{A_3}{A_i}\right)_{3D} = \left(\frac{A_3}{A_i}\right)_{2D} * \frac{Z_3}{Z_i} \quad (16)$$

the product $\rho_3 V_3$ must be divided by Z_3/Z_i to conserve mass flow

$$\left(\rho_3 V_3\right)_{3D} = \left(\rho_3 V_3\right)_{2D} * \frac{Z_i}{Z_3} \quad (17)$$

Then an iteration procedure is performed to determine the value of three dimensional expansion $(\Delta v_3)_{3D}$. 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)

TR 215
SECTION III
DETERMINATION OF NOZZLE FLOW FIELDS

A nozzle calculation is performed subject to the following constraints:

1. The initial profile is uniform. For the frozen flow (constant γ) calculation this requires specification of the pressure p_i , flow deflection angle θ_i , Mach number M_i , and specific heat ratio γ . For the equilibrium calculation one must specify P_i , θ_i , M_i , the temperature T_i and the fuel-air equivalence ratio ϕ_i .
2. The initial turning at the vehicle undersurface (Δv_v) and cowl (Δv_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 Δv_v as

parametric variables for fixed values of cowl turning Δv_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 $(\Delta v_v)_{\min}$ is determined and segmented into small increments $(\Delta v_i)_v$. The cowl expansion array is swept out by segmenting Δv_c into small segments $(\Delta v_i)_c$ and the interaction of each ray with the vehicle expansion is determined upto 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 upto 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 upto 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

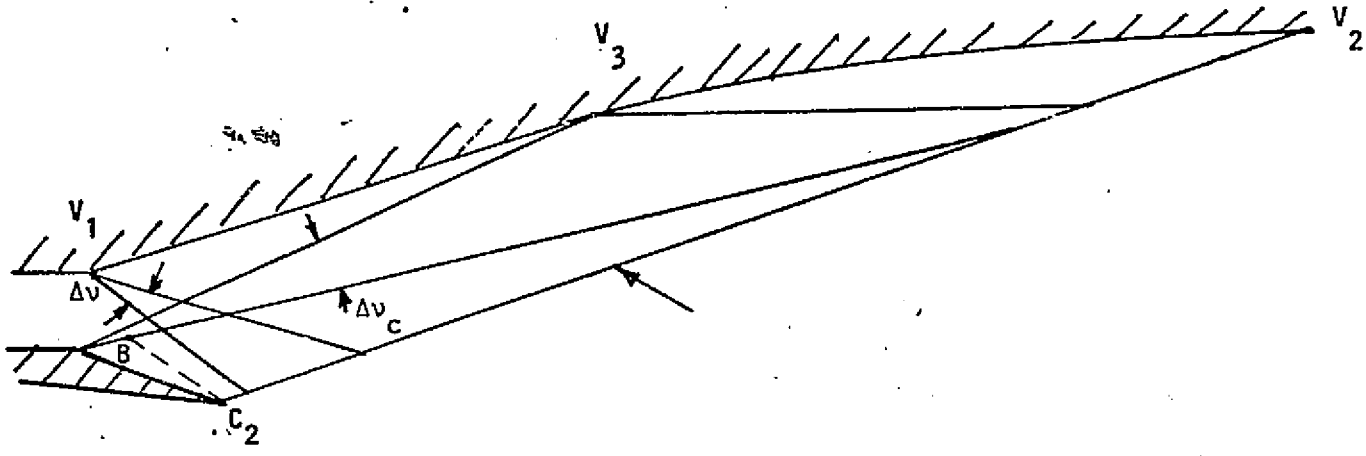


FIGURE 4. PARTIAL REFLECTION OF COWL EXPANSION/
NO REFLECTION OF VEHICLE EXPANSION

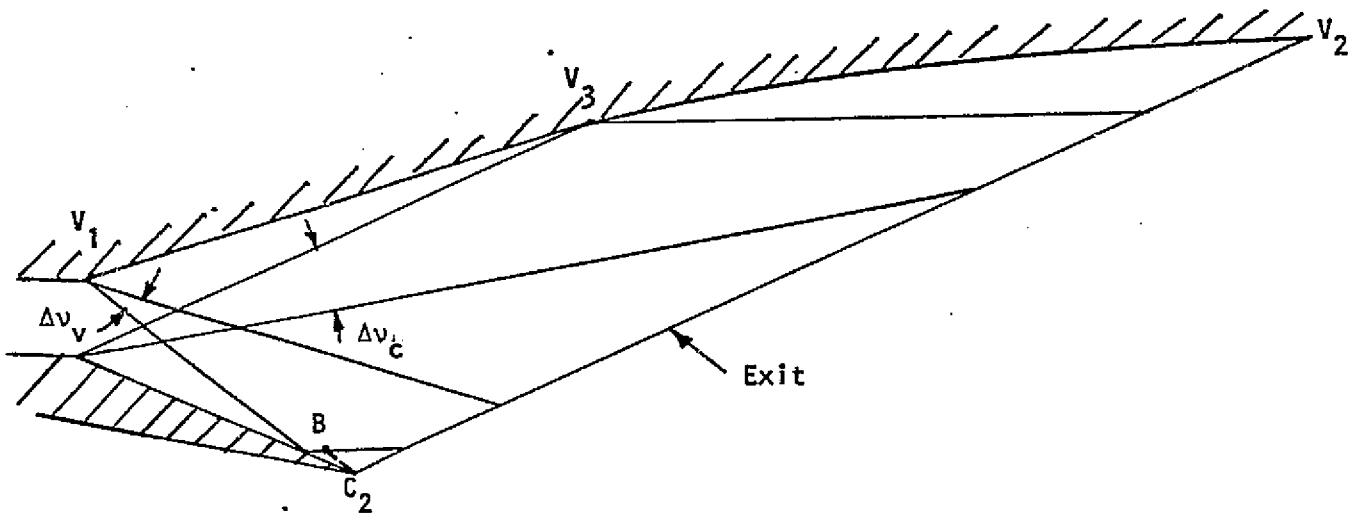


FIGURE 5. PARTIAL REFLECTION OF COWL AND VEHICLE EXPANSIONS

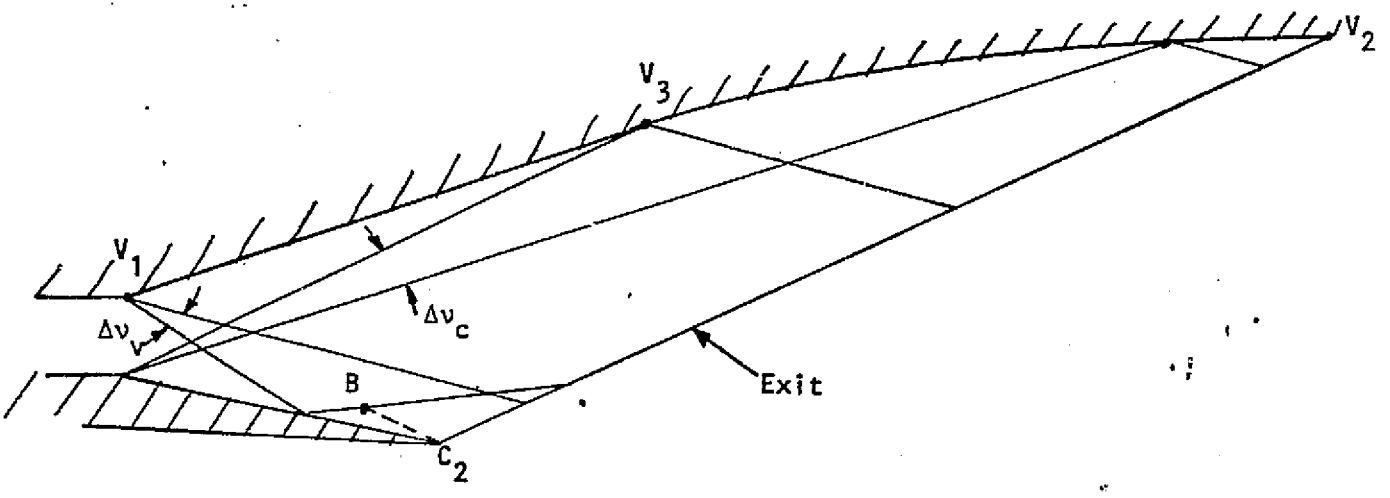


FIGURE 6. COMPLETE REFLECTION OF COWL EXPANSION/
PARTIAL REFLECTION OF VEHICLE EXPANSION

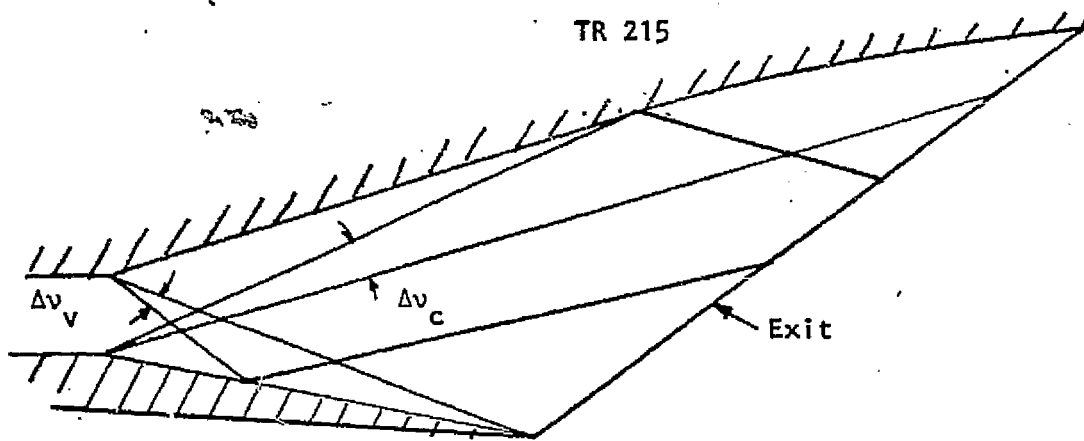


FIGURE 7. PARTIAL REFLECTION VEHICLE EXPANSION/
COMPLETE REFLECTION OF COWL EXPANSION

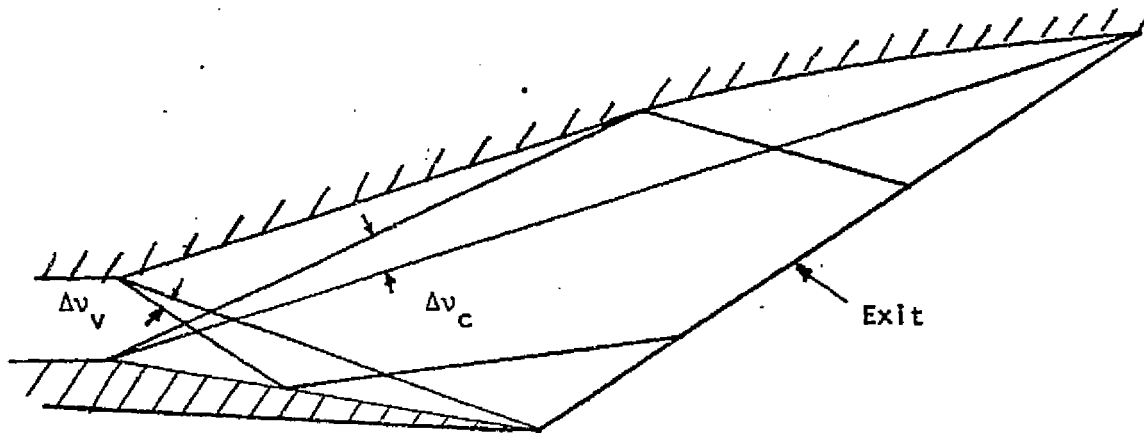


FIGURE 8. COMPLETE REFLECTION OF COWL AND VEHICLE EXPANSIONS

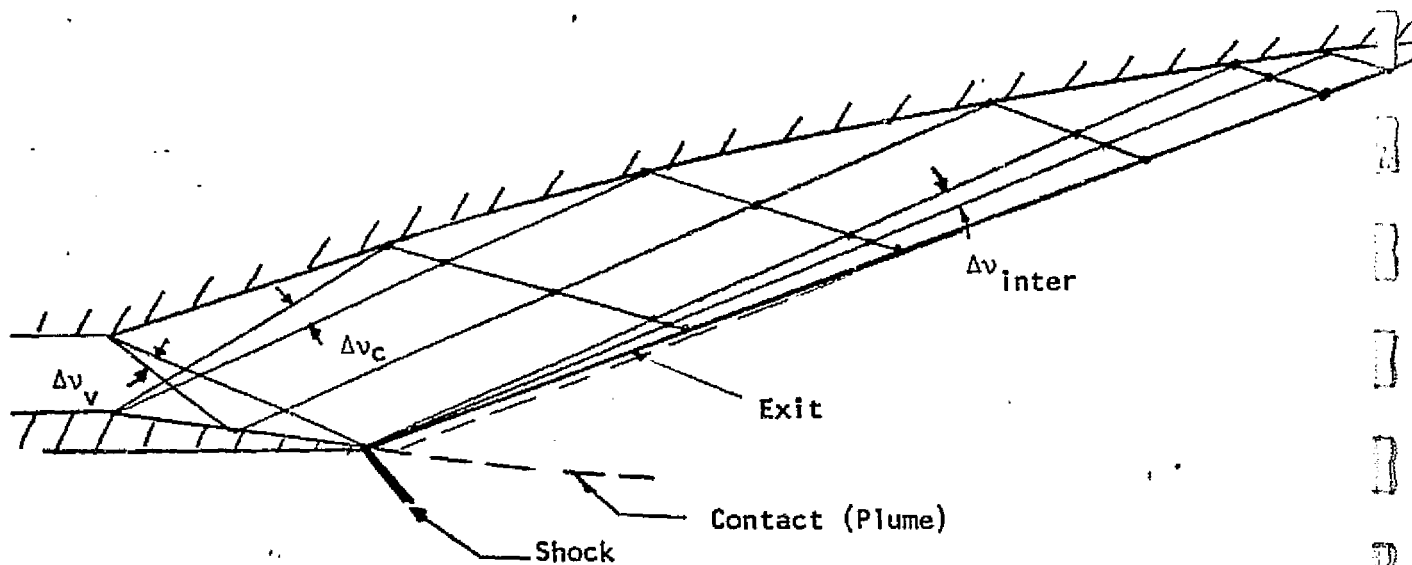


FIGURE 9. INTERACTION OF EXTERNAL FLOW WITH VEHICLE UNDERSURFACE

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).

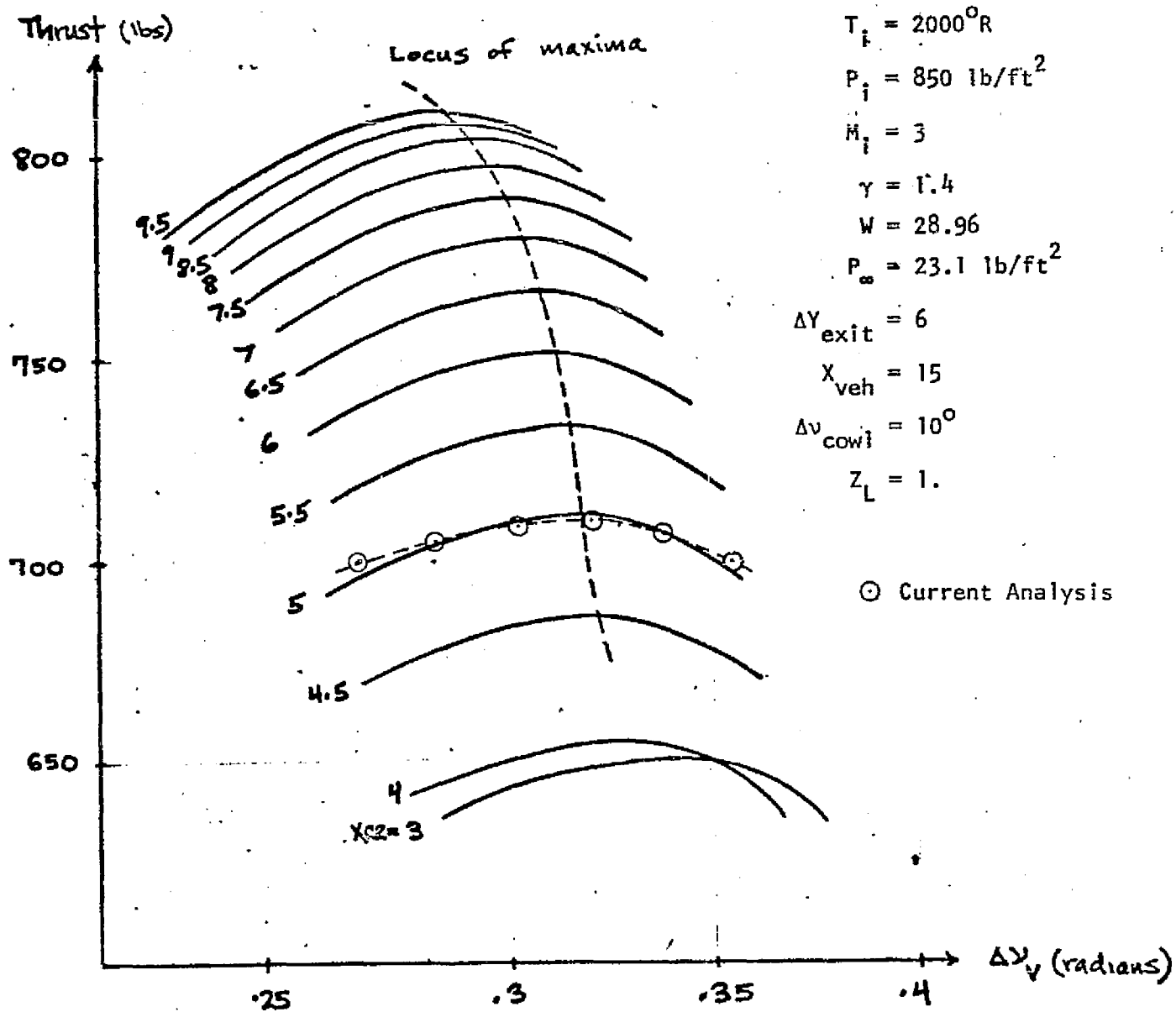


FIGURE 10. THRUST VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION ANGLE

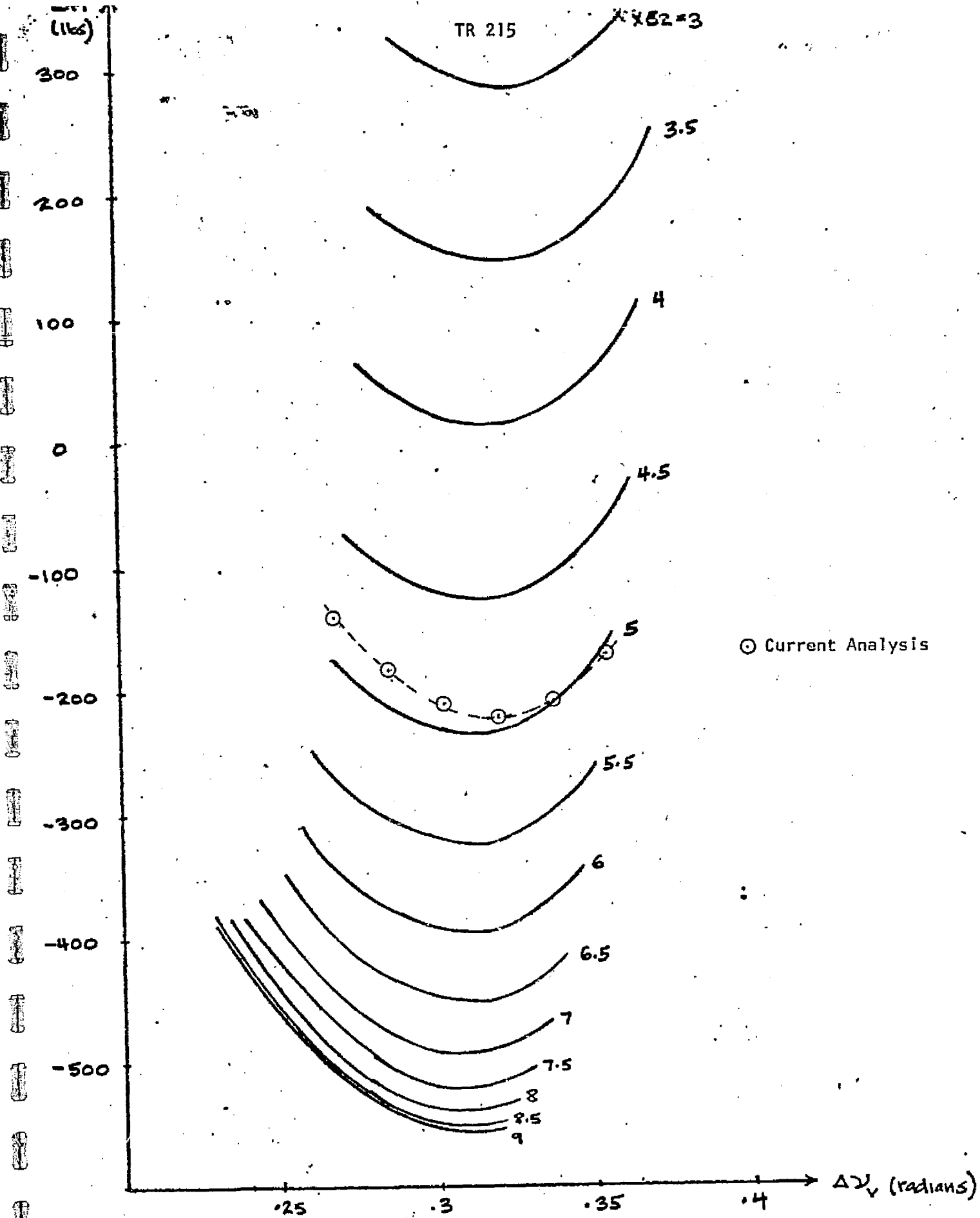


FIGURE 11 . LIFT VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION

THRUST, LIFT AND PITCHING MOMENT

The following definitions are used in this report for thrust, lift, pitching moment

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

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

$$M_y = + \int x \cdot dL + \int y \cdot dT \quad (20)$$

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.

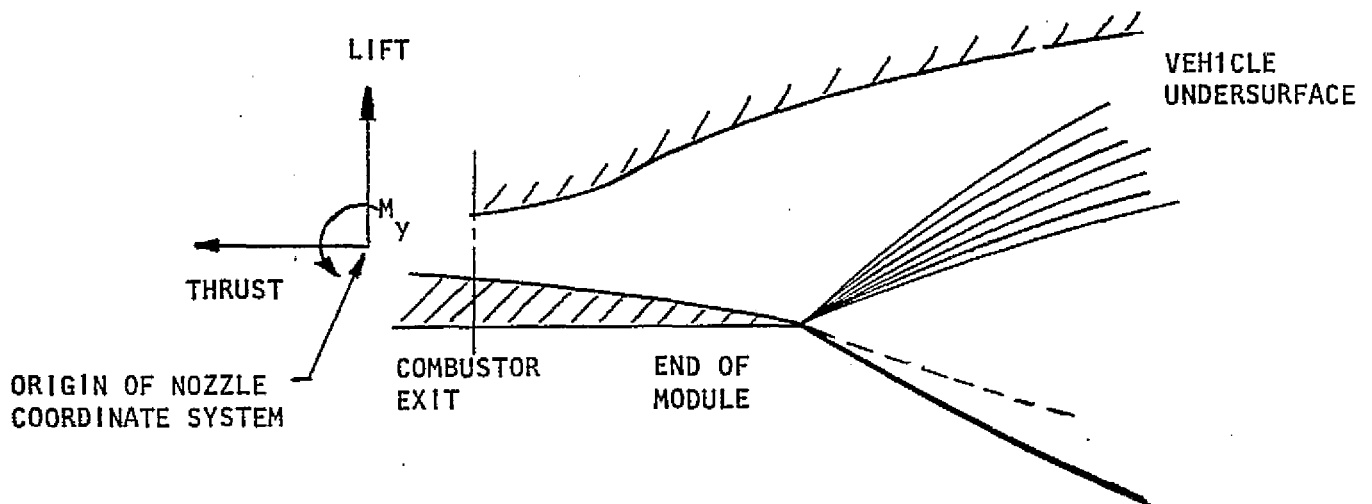


FIGURE 12. THRUST, LIFT, MOMENT

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

$$T_{vis} = - \int_A \left(\frac{\rho q^2}{2} \right)_{local} C_f \hat{i}_x \cdot d\vec{A}_s \quad (21)$$

$$L_{vis} = - \int_A \left(\frac{\rho q^2}{2} \right)_{local} C_f \hat{i}_y \cdot d\vec{A}_s \quad (22)$$

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.

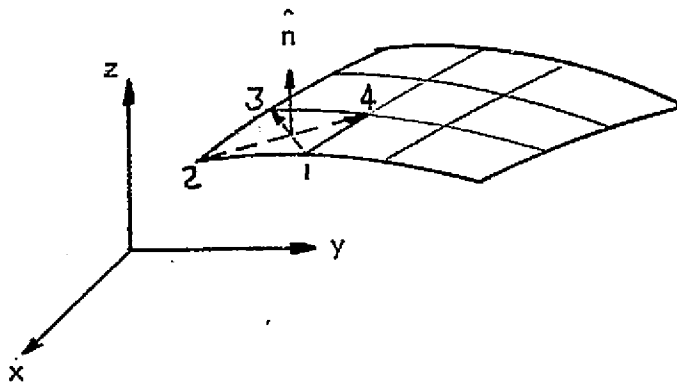


FIGURE 13. ELEMENTAL SURFACE AREA

That is

$$\vec{T}_1 = T_{1x} \hat{i}_x + T_{1y} \hat{i}_y + T_{1z} \hat{i}_z \quad (23a)$$

$$\vec{T}_2 = T_{2x} \hat{i}_x + T_{2y} \hat{i}_y + T_{2z} \hat{i}_z \quad (23b)$$

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 \vec{N} is defined as

$$\vec{N} = T_2 \times T_1$$

and the unit normal as

$$\hat{n} = \frac{\vec{N}}{|\vec{N}|} \quad (24)$$

A tangent plane is constructed using the normal vector and the two tangent vectors \vec{T}_2 , \vec{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)

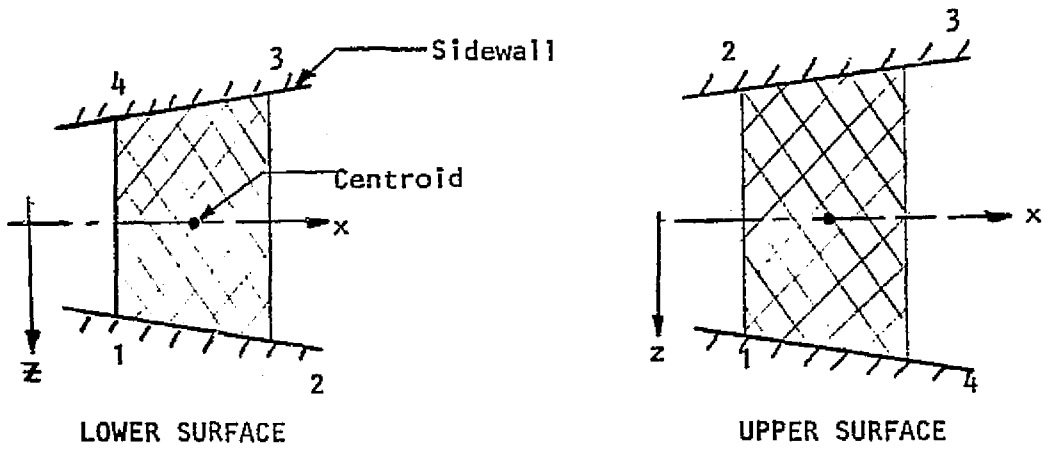


FIGURE 14. UPPER OR LOWER SURFACE AREA ELEMENTS

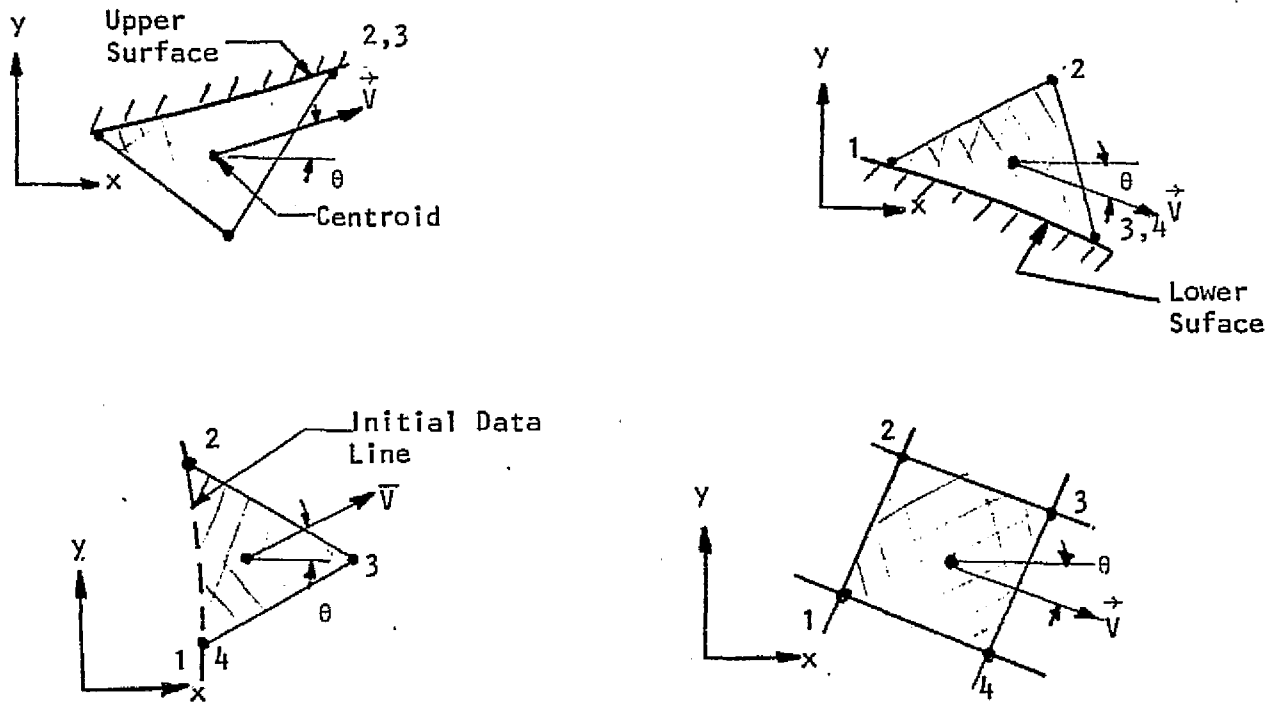


FIGURE 15. SIDEWALL AREA ELEMENTS

TR 215
SECTION V
VISCOUS EFFECTS

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.:

$$C_{f_{\delta}} = C_{f_i} / F_c \quad (25)$$

$$C_{f_i} = f(Rx_i), \quad Rx_i = F_{Rx} \cdot Rx \quad (26)$$

where

C_f = local skin friction coefficient

Rx = Reynolds number

$()_i$ = indicates incompressible

$()_{\delta}$ = indicates compressible

Now for $Rx_i > 2540$ the local skin friction is given from Reference (8) as

$$C_{f_i} = .088 (\log Rx_i - 2.3686) / (\log Rx_i - 1.5)^3 \quad (27)$$

and from Reference (7)

$$F_c = A / \left(\text{ARSIN} \left(\frac{A-B}{C} \right) + \text{ARSIN} \left(\frac{A+B}{C} \right) \right)^2 \quad (28)$$

$$A = \frac{H_{AW}}{H_{\delta}} - 1 \quad (29a)$$

$$B = \frac{H_W}{H_\delta} - 1 \quad (29b)$$

$$C = ((A+B)^2 + 4A)^{1/2} \quad (29c)$$

$$F_{Rx} = \left(\frac{H_{AW}}{H_\delta}\right)^q / (F_c \left(\frac{H_W}{H_\delta}\right)^{p+q}), \quad q = 0.772, \quad p = 0.702 \quad (30)$$

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

$$St = Sh \cdot C_f / 2 \quad (31)$$

The program requires "Sh" as an input item.

TR. 215,
SECTION VI
CONCLUSIONS

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

1. Dash, S., "The Determination of Nozzle Contours for Rotational, Non-Homentropic Gas Mixtures," ATL TR 148, March 1970.
2. Dash, S., Del Guidice, P., and Kalben, P., "A Numerical Procedure for the Parametric Optimization of Three Dimensional Scramjet Nozzles," ATL TR 188, NASA CR 132440, October 1973.
3. Ferri, A., Dash, S. and Del Guidice, P., "Methodology for Three Dimensional Nozzle Design," ATL TR 195, March 1974
4. Spalding, D.B. and Chi, S.W., "Drag of a Compressible Turbulent Boundary Layer on a Smooth Flat Plate With and Without Heat Transfer," J. Fluid Mechanics, Vol. 18, Pt.1, January 1964.
5. Del Guidice, P., Dash, S. and Kalben, P., "A Source Flow Characteristic Technique for the Analysis of Scramjet Exhaust Flow Fields," ATL TR 186, March 1974.
6. Dash, S. and Del Guidice, P., "Analysis and Design of Three Dimensional Supersonic Nozzles," ATL TR 166, October 1972.
7. Gentry, A.E. and Smyth, D.N., "Hypersonic Arbitrary-Body Aerodynamic Computer Program, Mark II Version," Vol. II, Douglas Report DAC 61552, April 1968.
8. Sivells, J.C. and Payne, R.G., "A Method of Calculating Turbulent Boundary Layer Growth at Hypersonic Mach Numbers," AEDC TR 59-3, March 1959.

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/ft ²)
	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	GAMI	(initial ratio of specific heats)
	61-70	PINF	(free stream pressure, lb/ft ²)

*Equilibrium Deck

	51-60	PINF	(free stream pressure, lb/ft ²)
--	-------	------	---

Card 2 (Format 8E10.0)

Column	1-10	PF	(external pressure, lb/ft ²)
	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 under- surface ft)
	11-20	YV1	(throat height on vehicle undersurface, ft)
	21-30	XV2	(axial location of vehicle end, ft)

Column	31-40	XCI	(axial location of throat on cowl surface, ft)
	41-50	YCI	(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	(Δv_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

(If IVIS is equal to 1 read this card)

Card 6a (Format 8E10.0)

Column	1-10	XSTR (virtual origin of boundary layer, ft)
	11-20	Rec (recovery factor)
	31-40	SH (constant for Stanton number calculation St = SH · CF/2.)
	41-50	RT (throat height, ft)

(If IT is equal to 0 read this card)

Card 6b (Format 8E10.0)

Column	1-10	AH(1) coefficients in equation
	11-20	BH(1) $T_{wall} = AH \cdot X^2 + BH \cdot X + CH$
	21-30	CH(1)

Card 7 (Format 8E10.0)

Column	1-10	AZ coefficients in equation
	11-20	BZ $Z = AZ \cdot X^2 + BZ \cdot X + CZ$
	21-30	CZ

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

$$A_{exit} = (YV_2 - YC_2) (ZC_2 + ZV_2) / 2.$$

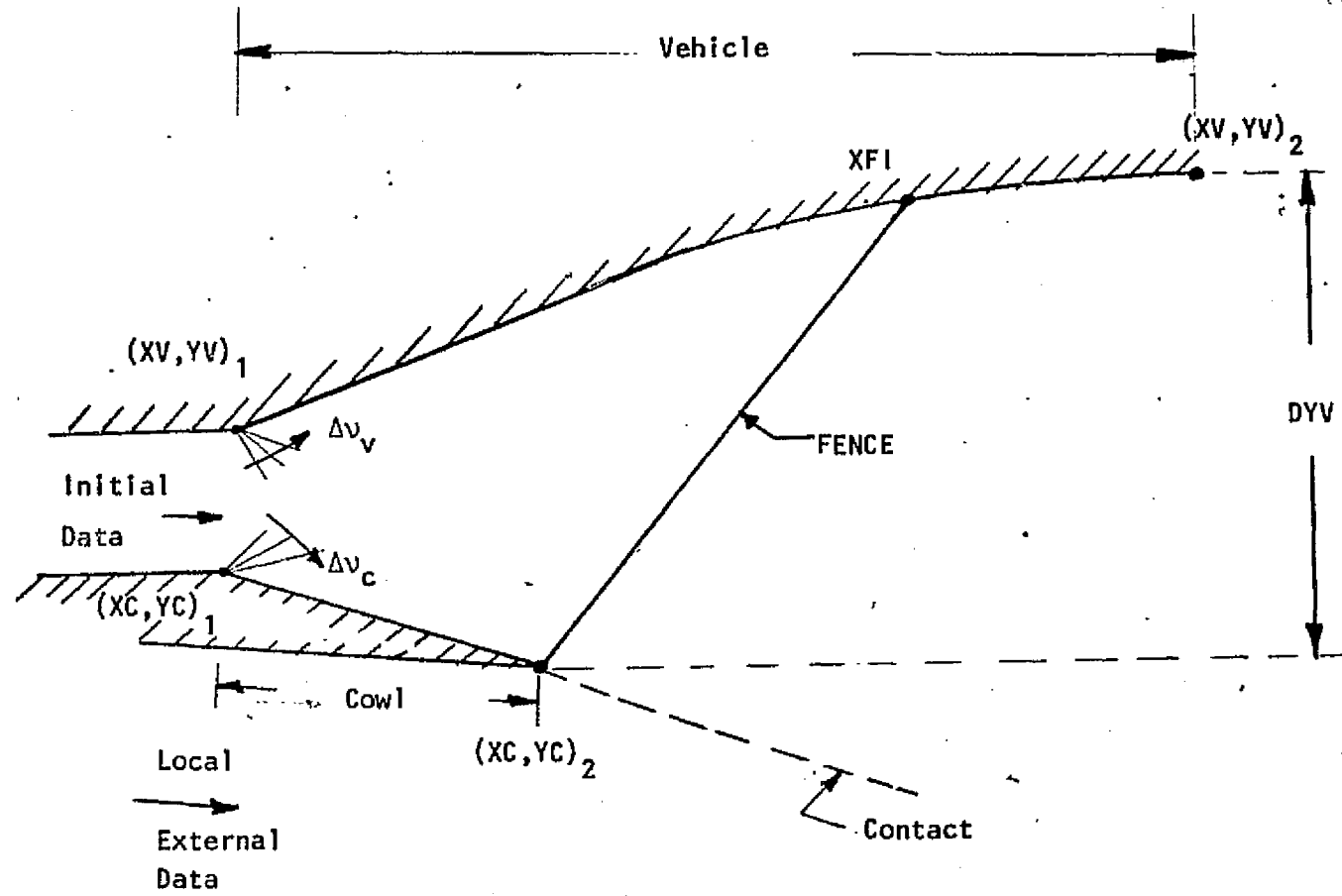


FIGURE 16. GEOMETRIC INPUTS

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 Δv
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 - Computes local plume shape
17. COWL - computes underexpansion interaction
18. PMI - 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, $T=FGAM(T,P,\phi)$
5. RHEQ- computes equilibrium mixture density,
 $\rho = RHEQ(H,P,\phi,T)$

APPENDIX II

LISTING OF FROZEN FLOW PROGRAM

```

PROGRAM NOZD(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
COMMON/IN/PI, TI, WI, THI, EMI, GAMI, ZI
COMMON/SHF/XSHFT, YSHFT
COMMON/PDL/ IPR, IPOLY
COMMON/THRMAX/THRMAX
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)
COMMON/C/DNUV, DNUC, XC1, YC1, XV1, YV1, DNUL
COMMON/CAL/XC2, YC2
COMMON/P/PINF, PTOT, T10T
COMMON/D3/ AZ, BZ, CZ
COMMON/V/AV, BV, CV, XV, XV2, YV2
COMMON/HOT/AH(3), BH(3), CH(3), XSTR, REC, RI, SH, IT, IVIS
COMMON/VISF/XVTHX, YVLFT, XVMOM
COMMON/CP/CP1, GI, RGAS
COMMON/FRIC/CF(50), ST(50)
COMMON/F/PF, TF, WF, THF, EMF, GF
COMMON/CNTC/TTC, PTC, WC, GC, PC, THC, XNUC, ZC, EMC
DIMENSION HOLE(11), HOLE(6), DNUE(10)
DATA HOLE/2HE0, 2HE1, 2HE2, 2HE3, 2HE4, 2HE5/
DATA HOLE/2HV1, 2HC1, 2HA, 2HB, 2HC, 2HD, 2HV3, 2HC3, 2HF, 2HG, 2HV4/
97 FORMAT(15,8E12.4)
98 FORMAT(/)
100 FORMAT(8E10.0)
D=0.
EPS=1.E-05
DO 23 L=1,3
AH(L)=0.
BH(L)=0.
23 CH(L)=0.
LH=1
READ(5,100) PI, TI, WI, THI, EMI, GAMI, PINF.
READ(5,100) PF, TF, WF, THF, EMF, GF
READ(5,100) XV1, YV1, XV2, XC1, YC1, XC2, DYV, DNUC
READ(5,100) XF1
READ(5,100) XVTHX, YLFT, XMMOM, XVTHX, YVLFT, XVMOM, XSHFT, YSHFT
READ(5,5921) ICF, OXC, JTF, DTH, IVIS, IT
IF(IVIS.EQ.1)READ(5,100)XSTR,REC, SH,RI
IF(IVIS.EQ.1.AND.IT.EQ.0)READ(5,100)AH(1),BH(1),CH(1)
5921 FORMAT(15,E10.0,15,E10.0,315)
XTHXI=XTHX
YLFTI=YLFT
XMMOI=XMMOM
XVTHXI=XVTHX
YVLFTI=YVLFT
XVMOMI=XVMOM
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=GAMI
WRITE(6,5922)
5922 FORMAT(1H177777)
WRITE(6,1020)
1020 FORMAT( 20X*FROZEN FLOW NOZZLE DESIGN*)

```

```

WRITE(6,5930) XV1,YV1,XV2,XC1,YC1,XC2,DYV
5930 FORMAT(///6X*XV1*10X*YV1*10X*XV2*10X*XC1*10X*YC1*10X*XC2*10X*DYV*/
17E13.5///)
1800 FORMAT(* THRUST= *E13.5,7X*LIFT = *E13.5,7X*MOMENT= *E13.5)
WRITE(6,1800)XTHX,YLFT,XMOM
1900 FORMAT(* VISCOUS THRUST= *E13.5,7X*VISCOUS LIFT= *E13.5,7X*
1VISCOUS MOMENT= *E13.5)
WRITE(6,1900)XVTHX,YVLF1,XVMOM
WRITE(6,5923)XSHF1,YSHF1
5923 FORMAT(9X*MOMENT AXIS*/12X*X =*E13.5/12X*Y =*E13.5///)
WRITE(6,1001) AZ,BZ,CZ
1001 FORMAT(9X*LATERAL EXPANSION EQUATION*/9X,22HZ(X) = AZ*X**2+BZ*X+CZ
1/12X*AZ =*E13.5/12X*BZ =*E13.5/12X*CZ =*E13.5)
DUM=1.+(GI-1.)/2.*EMI**2
PTOT=PI*DUM**(GI/(GI-1.))
TTOT=TI*DUM
RG=49800.
RGAS=RG/NI
CPI=GAMI*RGAS/(GAMI-1.)
XC22=XC2
THC1=THI-DNUC
DO 5000 IXC=1,ICF
XC2=XC22+FLOAT(IXC-1)*DXC
CALL THM(EMI,PI,GI,PI*F,YV1,YC1,DYV,XV2,THRMAX,XC2,ZI)
WRITE(6,6304) THRMAX
6364 FORMAT(1H131X*IDEAL THRUST =*E12.4)
YC2=YC1+TAN(THC1)*(XC2-XC1)
YV2=YC2+DYV
THC=ATAN((YC2-YC1)/(XC2-XC1))
DO 6000 JT=1,JTF
XTHX=XTHXI
YLFT=YLFTI
XMOM=XMOMI
XVTHX=XVTHXI
YVLF1=YVLF1I
XVMOM=XVMOMI
KSTP=50
1STP=50
IEND=0
IEXT=0
IFLG=0
IND=0
YHX=1.E+10
THJT=FLOAT(JT-1)*DTH
DNUV=ATAN((YV2-YV1)/(XV2-XV1))-THI+THJT
WRITE(6,1894)DNUV
1894 FORMAT(* VEHICLE EXPANSION = *E13.5)
XV=XV1
AV=YV1
BV=TAN(THI+DNUV)
CV=0.
YF1=AV+Bv*(XF1-XV)
IDUM=DNUV*57.3
IF(IDUM.GT.11) IDUM=11
IF(IDUM.EQ.0) IDUM=1
DELNU=DNUV/FLOAT(IDUM)
IDUM=IDUM+1
IMAX=IDUM
KMAX=IMAX+2
DO 10 I=1,IMAX
X(1,I)=XV1
Y(1,I)=YV1
Z(1,I)=GETZ(XV1)
TH(1,I)=THI+FLOAT(I-1)*DELNU

```

```

10 CONTINUE
IDUM=DNUC*57.3
IF(IDUM.GT.11) IDUM=11
IF(IDUM.EQ.0) IDUM=1
DELNU=DNUC/FLOAT(IDUM)
IDUM=IDUM+1
X(2,1)=XC1
Y(2,1)=YC1
Z(2,1)=GETZ(XC1)
ICWL=0
I=1
THII=THI
XNUI=0.
20 CONTINUE
TH(2,1)=THI-FLOAT(I-1)*DELNU
XNU(2,1)=FLOAT(I-1)*DELNU+XNUI
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 VWALL(IMAX,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
AV=Y(2,KMAX)
BV=TAN(TH(2,KMAX))
CV=(YV2-AV-BV*(XV2-X(2,KMAX)))/(XV2-X(2,KMAX))**2
THV2=ATAN(BV+2.*CV*(XV2-X(2,KMAX)))
XV=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
XK1=XK2=XK3=XK4=0.25
M=KT-1
J=KT-1
N=J
IF(KT.EQ.KMAX) J=KT-2
IF(K1.EQ.2.OR.KT.EQ.KMAX) N=J
IF(KT.EQ.2.OR.KT.EQ.KMAX) XK3=0.
IF(KT.EQ.2.OR.KT.EQ.KMAX) XK1=XK2=XK4=.333333
CALL LTHM(X(2,M),Y(2,M),Z(2,M),X(1,M),Y(1,M),Z(1,M),X(1,J),
1Y(1,J),Z(1,J),X(2,KT),Y(2,KT),Z(2,KT),P(2,M),P(1,M),P(1,J),
2P(2,KT),Q(2,M),Q(1,M),Q(1,J),Q(2,KT),T(2,M),T(1,M),T(1,J),
3T(2,KT),TH(2,M),TH(1,M),TH(1,J),TH(2,KT),XK1,XK2,XK3,XK4,1.,
4XTHX,YLFT,XMOM,CF(M),ST(M),3)
IF(K1.EQ.KMAX)
1CALL LTHM(X(1,J),Y(1,J),Z(1,J),X(1,J),Y(1,J),0.,X(2,KT),Y(2,KT),0.
2,X(2,KT),Y(2,KT),Z(2,KT),P(1,J),P(1,J),P(2,KT),P(2,KT),Q(1,J),
3Q(1,J),Q(2,KT),Q(2,KT),T(1,J),T(1,J),T(2,KT),T(2,KT),TH(1,J),
4TH(1,J),TH(2,KT),TH(2,KT),.25,.25,.25,.25,1.,XTHX,YLFT,XMOM,
SCFU,STU,1)
16 CONTINUE
GO TO 53
41 CONTINUE

```

```

IF(ICWL.EQ.1.AND.K.EQ.2.AND.IEXT.EQ.1)CALL CMT
L=K
IF(ICWL.EQ.1)L=K+1
IF(I.EQ.1) L =K-1
LM=L-1
LL=K+ICWL-1
IF(L .GT. ISTOP.AND.IEND.EQ.1)GO TO 30
MM=K-1
CALL FIX(2,K-1,1,L ,2,K)
IF(I.EQ.1)GO TO 30
IF(K.GT.1)SLA=(Y(2,K)-Y(2,MM))/(X(2,K)-X(2,MM))
CALL GEM(X(2,MM),Y(2,MM),SLA,XC2,YC2,SLE,XC,YC)
IF(XC.GE.X(2,K) .OR.XC.LT.X(2,MM)-EPS)GO TO 30
KSTP=K
WRITE(6,88)KSTP,ISTP,KMAX,IMAX
88 FORMAT(1X,4I5)
30 CONTINUE
LM=L-1
IF(I.EQ.1)GO TO 33
IF(IEXT.EQ.1)GO TO 32
IF(ICWL.EQ.0.OR.K.GT.2)GO TO 32
CALL LTHM(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(1,1),T(2,1),T(2,1),T(1,1),TH(1,1),TH(2,1),
3TH(2,1),TH(1,1),.25,.25,.25,.25,1.,XTHX, YLFT, XMOM,CFL,SIL,2)
CALL LTHM(X(1,1),Y(1,1),Z(1,1),X(1,2),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(2,1),T(2,1),TH(1,1),TH(1,2),
3TH(2,1),TH(2,1),.33333,.33333,.33333,0.,1.,XTHX, YLFT, XMOM,
4CF(1),ST(1),3)
32 CONTINUE
XK1=XK2=XK3=XK4=.25
IF(IEND.EQ.0.AND.K.EQ.KMAX)GO TO 17
AVG=4.
SLB=(Y(2,K)-Y(1,L))/(X(2,K)-X(1,L))
CALL GEM(X(1,L),Y(1,L),SLB,XC2,YC2,SLE,XC,YC)
RAT=(XC-X(1,L))/(X(2,K)-X(1,L))
IF(RAT.GT.0.99)GO TO 144
IF(RAT.GT.0.)GO TO 143
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 GEM(X(1,LM),Y(1,LM),SLB,XC2,YC2,SLE,XC,YC)
RAT=(XC-X(1,LM))/(X(2,MM)-X(1,LM))
IF(RAT.GT.0.99)GO TO 148
IF(RAT.GT.0.)GO TO 145
AVG=AVG-2.
GO TO 148
145 AVG=AVG-1.
148 CONTINUE
AVG=AVG/4.
CALL LTHM(X(2,MM),Y(2,MM),Z(2,MM),X(1,LM),Y(1,LM),Z(1,LM),X(1,L),
1Y(1,L),Z(1,L),X(2,K),Y(2,K),Z(2,K),P(2,MM),P(1,LM),P(1,L),P(2,K),
2Q(2,MM),Q(1,LM),Q(1,L),Q(2,K),T(2,MM),T(1,LM),T(1,L),T(2,K),
3TH(2,MM),TH(1,LM),TH(1,L),TH(2,K),XK1,XK2,XK3,XK4,AVG,XTHX, YLFT,
4XMOM,CF(LL),ST(LL),3)
IF(IEND.EQ.1.OR.K.LT.KMAX)GO TO 33
17 CONTINUE
M=K-1
L=M
IF(ICWL.EQ.1)L=K

```



```

CALL GEM(X(2,M),Y(2,M),SLA,XC2,YL2,SLE,XC,YC)
RAT=(XC-X(2,M))/(X(2,K)-X(2,M))
IF(RAT.GT.0.99)GO 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,SLE,XC,YC)
RATE=(XC-X(1,L))/(X(2,M)-X(1,L))
IF(RAT.GT.0.99)GO TO 178
IF(RAT.GT.0.)GO TO 173
AVG=AVG-2.
173 AVG=AVG-1.
178 CONTINUE
AVG=AVG/3.
CALL LTHM(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(2,M),P(1,L),P(2,K),P(2,K),Q(2,M),
2Q(1,L),Q(2,K),Q(2,K),T(2,M),T(1,L),T(2,K),T(2,K),TH(2,M),TH(1,L),
3TH(2,K),TH(2,K),.33333,.33333,.33333,0.,AVG,XTHX,YLEFT,XMOM,
4CF(LL),ST(LL),3)
45 CALL LTHM(X(1,L),Y(1,L),Z(1,L),X(1,L),Y(1,L),0.,X(2,K),Y(2,K),
10.,X(2,K),Y(2,K),Z(2,K),P(1,L),P(1,L),P(2,K),P(2,K),Q(1,L),Q(1,L),
2Q(2,K),Q(2,K),T(1,L),T(1,L),T(2,K),T(2,K),TH(1,L),TH(1,L),TH(2,K),
3TH(2,K),.25,.25,.25,.25,1.,XTHX,YLEFT,XMOM,CFU,STU,1)
33 IF(IEEND.EQ.1)KMAX=ISTP
K=K+1
IF(K.LE.KMAX)GO TO 36
53 KP=KMAX
IF(IEEND.EQ.1)KP=KMAX-1-ICWL
IF(IND.EQ.1,1END=1
IND=0
WRITE(6,6885)
6885 FORMAT(1X*PT.*6X*X*11X*Y*8X*PRESSURE*5X*ANGLE*8X*MACH*4X*TEMPERATI
1RE*1X*PM FUNCTION*3X*VELOCITY*)
DO 50 L=1,KP
WRITE(6,97) L,X(2,L),Y(2,L),P(2,L),TH(2,L),EW(2,L),T(2,L),XNU(2,
1L),Q(2,L)
50 CALL SWITCH(2,L,1,L)
WRITE(6,2001)CFU,STU,CFL,STL
2001 FORMAT(* VEH. FRIC. COEFF.= *E13.5,2X*VEH. STANTON NUM.= *E13.5,2X*
1COWL FRIC. COEF.= *E13.5,2X*COWL STANTON NUM.= *E13.5)
WRITE(6,450)
NPR1=1MAX/5
DO 166 K=1,NPR1
K1=K&K2=K1+NPR1&K3=K2+NPR1&K4=K3+NPR1&K5=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)GO TO 55
NPR2=5*NPR1+1
DO 414 K=NPR2,1MAX
WRITE(6,550)CF(K),ST(K)
414 CONTINUE
55 CONTINUE
450 FORMAT(1X,5(6X,*CF *,9X,* ST *,3X))
500 FORMAT(1X,10E13.5)
550 FORMAT(105X,2E13.5)
WRITE(6,98)
WRITE(6,1800)XTHX,YLEFT,XMOM
WRITE(6,1900)XVTHX,YVLEFT,XVMOM
1MAX=KMAX
I=I+1
IF(I.GT.10JM)ICWL=1
IF(ICWL.EQ.0)KMAX=KMAX+1

```

```
IF(IEXT.EQ.0.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(*THV2.LI.0*)
GO TO 6000
5900 CONTINUE
IF(P(1,1).LE.PF)GO TO 5396
CALL COWL(DNUE)
IDUM=6
I=2
XNCI=XNU(1,1)
THI1=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 CWALL(IMAX,KMAX,KSTP,IS1P)
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),XMU(2,50),Q(2,50)
COMMON/C/DNUV,DNUC,XC1,YC1,XV1,YV1,DMUL
COMMON/CAL/XC2,YC2
D=0.
IADD=0
LOOP=0
XTH1=TH(1,1)-XMU(1,1)
Q1=XNU(1,1)-TH(1,1)
CALL INT(0.,X(1,2),Y(1,2),Z(1,2),TH(1,2),XMU(1,2),P(1,2),T(1,2),
1EM(1,2),XMU(1,2),D,D,D,D,D,D,D,D,D,D,XA,YA,ZA,THA,XNUA,PA,TA,
2EMA,XNUA,-1.)
Q2=XNU(1,2)-TH(1,2)
44 XTH2=THA-XNUA
IF(LOOP.EQ.0)XTH3=XTH2
SLA=(TAN(XTH2)+TAN(XTH3))/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
XA=(X(1,2)+X(1,1))/2.
24 RAT=(XA-X(1,1))/DX
SLB=XTH1+RAT*(XTH2-XTH1)
SLB=TAN(SLB)
IF(LOOP.EQ.1)SLB=(SLB+TAN(XTH3))/2.
CALL GEM(X(1,1),Y(1,1),SLU,XC2,YC2,SLB,XAT,YA)
ER=ABS((XA-XAT)/DX)
IF(ER.LI.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(* TOO MANY ITER IN CWALL *)
WRITE(6,39)LOOP,SLU,SLB,XA,XAT
39 FORMAT(1X,I5,5E13.5)
STOP
63 CONTINUE
XC=XC2
YC=YC2
IM=IMAX+1
X(1,IM)=XA
Y(1,IM)=YA
Z(1,IM)=GETZ(XA)
XNU(1,IM)=XNU(1,1)+RAT*(XNU(1,2)-XNU(1,1))
QA=Q1+RAT*(Q2-Q1)
TH(1,IM)=XNU(1,IM)-QA
CALL PM(1,IM)
CALL INT(0.,X(1,IM),Y(1,IM),Z(1,IM),TH(1,IM),XMU(1,IM),P(1,IM),
1T(1,IM),EM(1,IM),XMU(1,IM),D,D,D,D,D,D,D,D,D,D,XA,YA,ZA,THA,XNUA,
2PA,TA,EMA,XNUA,1.)
45 RA=THA+XNUA
IF(ABS(XC2-XC).LE..0001)XC=XC2
X(2,1)=XC
Y(2,1)=YC

```

```
CALL PM(2,1)
XTHM3=TH(2,1)-XMU(2,1)
IF(LOOP.EQ.1)GO TO 6
LOOP=1
GO TO 44
6 IF(IADD.EQ.0)RETURN
IMAX=IMAX+1
KMAX=KMAX+1
KSTP=KSTP+1
ISTP=ISTP+1
DO 46 JJ=3,IMAX
LJ=IMAX-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,I5,3E13.5)
CALL INT(0.,XA,YA,ZA,THA,XNUA,PA,TA,EMA,XMUA,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),EM(1,2),
2XMU(1,2),-1.)
RETURN
END
```

```
SUBROUTINE ENDD(L, KK, K6)
COMMON/V/AV, BV, CV, XV, XV2, YV2
COMMON/X/X(2, 50), Y(2, 50), Z(2, 50), TH(2, 50), XNU(2, 50),
IP(2, 50), I(2, 50), EM(2, 50), XMU(2, 50), Q(2, 50)
```

```
IT=1
```

```
B=0.
```

```
A=1.
```

```
XM1=TH(1, L )+XMU(1, L )
```

```
XM2=TH(2, KK)+XMU(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
```

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

```
SLP=XM1+RAT*(XM2-XM1)
```

```
SLP=A*SLP+B*XM3
```

```
CALL GEM(X(1, L ), Y(1, L ), SLM, XV2, YV2, SLP, XAT, YA)
```

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

```
IF(ER.LT.1.E-03)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)L, KK, K6, XA, XAT, X(2, KK), X(1, L ), SLP, XM3
```

```
34 FORMAT(1X, 3I5, 6E13.5)
```

```
STOP
```

```
8 XA=XAT
```

```
GO TO 12
```

```
14 QA=Q1+RAT*(Q2-Q1)
```

```
TH(1, L +2)=TH(1, L )+RAT*(TH(2, KK)-TH(1, L ))
```

```
XNU(1, L +2)=QA-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)=ATAN (BV+2.*CV*(XV2-XV))
```

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

```
CALL PM(2, K6)
```

```
IEND=1
```

```
IF(B.GT.0.)RETURN
```

```
IT=1
```

```
B=.5
```

```
A=.5
```

```
XM3=TH(2, K6)+XMU(2, K6)
```

```
GO TO 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,XMU3,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=(Q3-XMU3)/OPT  
XMU3=ASIN(1./EM3)  
Z3=GETZ(X3)  
RETURN  
END
```

SUBROUTINE THM(EM1,P1,G,PINF,YV1,YC1,DYV,XV2,THRMAX,XC2,ZI)

A1=YV1-YC1

A1=A1*ZI

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.))

AST=A1*EM1/DUM

PIOT=P1/FM1*(G/(1.-G))

EM2=EM1*SQRT(A2/A1)

IF(EM1.GT.3.)EM2=(A2/A1)**.3*EM1

AF=A2/AST

ITM=0

10 CONTINUE

FM2=1.+(G-1.)/2.*EM2**2

DUM=(2.*FM2/(G+1.))*((G+1.)/2./(G-1.))

AFT=DUM/EM2

ERA=(AF1-AF)/AF

IF(ABS(ERA).LT.1.E-03) GO TO 20

CALL ERROR(2000,ITM,EM2,ERA,1.1,EM21,ERA1)

GO TO 10

20 CONTINUE

F1=P1*A1*(1.+G*EM1**2)

P2=FM2*(G/(1.-G))*PIOT

F2=P2*A2*(1.+G*EM2**2)

THRMAX=F2-F1-PINF*(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=EM*SQRT(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(10, 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),
IP(2,50),I(2,50),EM(2,50),XMU(2,50),Q(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)=I (I2,K)
EM (I1,L)=EM (I2,K)
XMU(I1,L)=XMU(I2,K)
Q(I1,L)=Q(I2,K)
RETURN
END
```

```
FUNCTION GETZ(X)
COMMON/D3/ AZ,BZ,CZ
GETZ=AZ*X**2+BZ*X+CZ
RETURN
END
```

```

SUBROUTINE VWALL (JMAX,KMAX)
COMMON/X/X(2,50),Y(2,50),Z(2,50),TH(2,50),XMU(2,50),
IP(2,50),T(2,50),EM(2,50),XMU(2,50),Q(2,50)
COMMON/V/AV,BV,CV,XV,XV2,YV2
COMMON/C/DNUV,DNUC,XC1,YC1,XV1,YV1,DNUL
IBEG=0
ITN=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 YW=AV+BV*(XG-XV)+CV*(XG-XV)**2
ERW=(YW-YG)/(YV1-YC1)
IF(ABS(ERW).LT.1.E-02) GO TO 10
CALL ERROR(1,ITN,XG,ERW,.9,XC1,ERW1)
YG=Y(2,KMAX-1)+SL13*(XG-X(2,KMAX-1))
10 TH(2,KMAX)=ATAN(BV+2.*CV*(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)=GETZ(XG)
CALL PM(2,KMAX)
IF(IBEG.EQ.1) RETURN
IBEG=1
SL13=.5*(SL1+TAN(TH(2,KMAX)+XMU(2,KMAX)))
GO TO 12
END

```

```
SUBROUTINE GEM(XA, YA, SLA, XH, 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) I
```

```
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),
1P(2,50),T(2,50),E4(2,50),XMU(2,50),W(2,50)
DUM3=0.
DUM4=0.
A=1.
B=0.
DUM1=TAN(TH(I1,K1)+XMU(I1,K1))
DUM2=TAN(TH(I2,K2)-XMU(I2,K2))
10 IF(B.GT.0.) DUM3=TAN(TH(I3,K3)+XMU(I3,K3))
IF(B.GT.0.) DUM4=TAN(TH(I3,K3)-XMU(I3,K3))
SL1=A*DUM1+B*DUM3
SL2=A*DUM2+B*DUM4
CALL G4M(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))
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(IX,KX)
COMMON/IV/PI, TI, VI, TH1, EMI, GAMI, ZI
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), XMU(2,50), Q(2,50)
COMMON/P/PI*F, PI*OT, TI*OT
COMMON/CP/CP1, GI, RGAS
DNU=XNU(IX,KX)
G=GI
EMI=EMI
GG=SQRT((G+1.)/(G-1.))
XM1=SQRT(EMI**2-1.)
XNU1=GG*ATAN(XM1/GG)-ATAN(XM1)
EM2=DNU/(1.5-XNU1)*(6.-EMI)+EMI
IT3=1
10 XM2=SQRT(EM2**2-1.)
DNUT=GG*(ATAN(XM2/GG)-ATAN(XM1/GG))+ATAN(XM1)-ATAN(XM2)
ERNU=DNU-DNUT
IF(ABS(ERNU).LT.1.E-04) GO TO 20
CALL ERROR(3,IT3,EM2,ERNU,1.,1,EM21,ERNU1)
GO TO 10
20 CONTINUE
CALL EM3D(EM2,EMX,X(IX,KX),G,Z(IX,KX),ZI)
P(IX,KX)=PI*OT/(1.+(G-1.)/2.*EMX**2)**(G/(G-1.))
T(IX,KX)=TI*OT/(1.+(G-1.)/2.*EMX**2)
EM(IX,KX)=EMX
XMU(IX,KX)=ASIN(1./EMX)
AX=SQRT(G*RGAS*T(IX,KX))
Q(IX,KX)=EMX*AX
RETURN
END

```

```

SUBROUTINE GNDRE(RH,G,P,T,R,          X,X1,CF,ST,L)
COMMON/CP/CPI,G1, RGAS
COMMON/HOT/AH(3),BH(3),CH(3),XSTR, REC,      RT,SH,IT,IVIS
HDEL=R-Q*Q/2.
HAW=1.+REC*Q*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=CPI*TW
HW=HA/HDEL
GO TO 48
46 HW=HAW
48 A=HAW-1.
B=HW-1.
C=SQRT((A+B)**2+4.*A)
FC=A/(ASIN((A-B)/C)+ASIN((A+B)/C))**2
FRX=HAW**(.7/2)/(FC*(HW)**(1.474))
CALL VIS(T,XMMU)
REX=RH*Q*(X+XSTR)*RT /XMMU
REXI=FRX*REX
CFI=.088*(ALOG10(REXI)-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 SWARF(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,AVX,AVY,AVZ,  
1XNX,XNY,XNZ,AS,X0,Y0,Z0,LH)
```

```
DIMENSION XPA(4),YPA(4),ZPA(4),XI(4),ETA(4)
```

```
XPA(1)=X1
```

```
YPA(1)=Y1
```

```
ZPA(1)=Z1
```

```
ZPA(2)=Z2
```

```
YPA(2)=Y2
```

```
XPA(2)=X2
```

```
XPA(3)=X3
```

```
YPA(3)=Y3
```

```
ZPA(3)=Z3
```

```
ZPA(4)=Z4
```

```
YPA(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=T1X*T2Z-T2X*T1Z
```

```
XNZ=T2X*T1Y-T1X*T2Y
```

```
VN=SQRT(XNX**2+XNY**2+XNZ**2)
```

```
IF(VN.LE.1.E-13)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=SQRT(T1X*T1X+T1Y*T1Y+T1Z*T1Z)
```

```
T1X=T1X/T
```

```
T1Y=T1Y/T
```

```
T1Z=T1Z/T
```

```
T2X=XNY*T1Z-XNZ*T1Y
```

```
T2Y=XNZ*T1X-XNX*T1Z
```

```
T2Z=XNX*T1Y-XNY*T1X
```

```
DO 1000 J=1,4
```

```
XPA(J)=XPA(J)+XNX*D
```

```
YPA(J)=YPA(J)+XNY*D
```

```
ZPA(J)=ZPA(J)+XNZ*D
```

```
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*XDIF+T2Y*YDIF+T2Z*ZDIF
```

```
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)-XIO
```

```
1020 ETA(J)=ETA(J)-ETA0
```

```
XU=AVX+T1X*XIO+T2X*ETA0
```

```
YU=AVY+T1Y*XIO+T2Y*ETA0
```

```
ZU=AVZ+T1Z*XIO+T2Z*ETA0
```

```
AS=(ETA(2)-ETA(4))*(XI(3)-XI(1))/2.
```

```
AS=ABS(AS)
```

AS=0.
RETURN
END

SUBROUTINE LTHM(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,P1,P2,P3,P4,
1Q1,Q2,Q3,Q4,T1,T2,T3,T4,TH1,TH2,TH3,TH4,XK1,XK2,XK3,XK4,AVG,
2XXTHX,XYLFT,XXMOM,CFF,ST,LH)
COMMON/P/PINF,PTOT,TTOT
COMMON/CP/CPI,GI,RGAS
COMMON/SHF/XSHFT,YSHFT
COMMON/HOT/AH(3),BH(3),CH(3),XSTR,REC,RT,SH,IT,IVIS
COMMON/VISF/XVTHX,YVLFT,XVMOM
P=XK1*P1+XK2*P2+XK3*P3+XK4*P4
Q=XK1*Q1+XK2*Q2+XK3*Q3+XK4*Q4
T=XK1*T1+XK2*T2+XK3*T3+XK4*T4
TH=XK1*TH1+XK2*TH2+XK3*TH3+XK4*TH4
RH=P/RGAS/T
R=CPI*T+Q*Q/2.
AVX=XK1*X1+XK2*X2+XK3*X3+XK4*X4
AVY=XK1*Y1+XK2*Y2+XK3*Y3+XK4*Y4
AVZ=XK1*Z1+XK2*Z2+XK3*Z3+XK4*Z4
CALL SNARF(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,AVX,AVY,AVZ,XNX,
1XNY,XNZ,ASS,XU,YU,ZU,LH)
CFF=0.
XBP=0.
IF(IVIS.EQ.1)CALL GNURE(RH,Q,P,T,R,XU,XBP,CFF,ST,1)
RHQ=RH*Q*Q/2.
PAV=P-PINF
DXTHX=-PAV*XNX*ASS
DYLFT=PAV*XNY*ASS
XNZZ=1.
IF(LH.EQ.3)XNZZ=XNZ
DXTHXV=-CFF *COS(TH)*ASS*RHQ
DYLFTV=CFF *SIN(TH)*ASS*RHQ
DXTHXV=DXTHXV*AVG
DYLFTV=DYLFTV*AVG
XMS=XU-XSHFT
YMS=YU-YSHFT
DMOMV=YMS*DXTHXV+XMS*DYLFTV
XVTHX=XVTHX+DXTHXV
YVLFT=YVLFT+DYLFTV
XVMOM=XVMOM+DMOMV
DXTHX=DXTHX*AVG+DXTHXV
DYLFT=DYLFT*AVG+DYLFTV
DMOM=YMS*DXTHX+XMS*DYLFT
XXTHX=XXTHX+DXTHX
XYLFT=XYLFT+DYLFT
XXMOM=XXMOM+DMOM
RETURN
END

```

SUBROUTINE CNT
COMMON/IN/PI, TI, NI, THI, EMI, GAMI, ZI
COMMON/CNTC/TTC, PTC, PC, GC, PC, THC, XNUC, ZC, EMC
COMMON/P/PINF, PTOT, TTOT
COMMON/CP/CPI, GI, RGAS
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), XMU(2,50), Q(2,50)
A=1. $ B=0.
TOTS=TTOT $ POTS=PTOT $ GIS=GI $ ZIS=ZIS $ EMIS=EMI
IT=1
RB=XNU(1,2)+TH(1,2)
TH(2,1)=TH(1,1)
20 SLA=A*TAN(TH(2,1))+B*TAN(TH(1,1))
SLB=TH(1,2)-XNU(1,2)
SLB=A*TAN(SLB)
IF(B.GT.0.)SLB=SLB+B*TAN(TH(2,1)-XNU(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)=RB-TH(2,1)
CALL PM(2,1)
TTOT=TTC $ PTOT=PTC $ GI=GC $ ZI=ZC $ SEMI=EMC
RC=XNUC-THC
TH(2,2)=TH(2,1)
XNU(2,2)=RC+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(ABS(ER3).LT.1.E-04)GO TO 16
CALL ERROR(IER,IT,TH(2,1),ER3,1.01,TH23,ER23)
TTOT=TOTS $ PTOT=POTS $ GI=GIS $ ZI=ZIS $ SEMI=EMIS
GO TO 20
16 CONTINUE
TTOT=TOTS $ PTOT=POTS $ GI=GIS $ ZI=ZIS $ SEMI=EMIS
IF(B.GT.0.)RETURN
IT=1
A=.5 $ B=.5
GO TO 20
END

```

```

SUBROUTINE COWL(DNUE)
COMMON/X/X(2,50),Y(2,50),Z(2,50),TH(2,50),XNU(2,50),
IP(2,50),I(2,50),EM(2,50),XMU(2,50),Q(2,50)
COMMON/CAL/XC2,YC2
COMMON/P/PINF,PIOT,TTOT
COMMON/F/PF,TF,WF,THF,EMF,GF
COMMON/CP/CPI,GI,KGAS
COMMON/CNTC/TTC,PTC,WC,GC,PC,THC,XNUC,ZC,EMC
PCC=(PF+P(1,1))/2.
GM1=GF-1.
GP1=GF+1.
EMF2=EMF*EMF
DUM=1.+GM1*EMF2/2.
PTF=PF*DUM** (GF/GM1)
TTF=TF*DUM
TOTS=TTOT $POTS=PIOT $GIS=GI
XMUF=ASIN(1./EMF)
IT=1
BET=(THF-XMUF+TH(1,1))*1.1-THF
20 DUM=(EMF*SI(BET))*2
PC=(2.*GF*DUM-GM1)/GP1
TC=PC*(GM1*DUM+2.)/DUM/GP1
EMC=(EMF2*(GP1*PC+GM1)-2.*(PC*PC-1.))/(GM1*PC+GP1)/PC
EMC=SQRT(EMC)
PC=PC*PF
THC=TAN(BET)*(GP1*EMF2/(DUM-1.)/2.-1.)
THC=ATAN(1./THC)+THF
PTC=(2.*TC/(GM1*EMF2+2.))* (GF/GM1)
PTC=PC/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,ER41)
GO TO 20
16 DNUE=ABS(XNU(2,1)-XMU(1,1))
TTOT=TOTS $PTOT=POTS $GI=GIS
WC=WF
GC=GF
ZC=Z(2,1)
XNUC=0.
RETURN
END

```

ORIGINAL PAGE IS
OF POOR QUALITY

#								
850.	2000.	28.96	0.	3.	1.4	23.1		
23.09	418.8	28.97	0.	10.	1.4			
0.	1.	15.	0.	0.	5.	6.	10.	
15.								
0.	0.	0.	0.	0.	0.	0.	.5	
	1	1.	1	1.	1	1		
0.	1.	1.	1.					
0.	0.	1.						

TR 215

APPENDIX III

LISTING OF EQUILIBRIUM FLOW PROGRAM

```

PROGRAM NOZDE(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
COMMON/IN/PI, TI, NI, THI, EM1, GI, ZI, HI, RHI, UI
COMMON/SHF/XSHFT, YSHFT
COMMON/IHRMAX/THRMAX
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), XMU(2,50), Q(2,50), H(2,50), KH(2,50),
2G(2,50)
COMMON/D/PHI1
COMMON/HOT/AH(3), BH(3), CH(3), XSTR, REC, RT, SH, IT, IVIS
COMMON/C/DNUV, DNUC, XC1, YC1, XV1, YV1, DNUL
COMMON/CWL/XC2, YC2
COMMON/P/PINF, PTOT, TTOT
COMMON/D3/AZ, BZ, CZ
COMMON/V/AV, BV, CV, XV, XV2, YV2
COMMON/VISF/XVTHX, YVLFT, XVMOM
COMMON/CP/CPI, RGAS
COMMON/FRIC/CF(50), ST(50)
COMMON/F/PF, TF, WF, THF, EMF, GF
COMMON/CNTC/TTC, PTC, WC, GC, PC, THC, XNUC, ZC, EMC
DIMENSION HOL(11), HOLE(6), DNUF(10)
DATA HOLE/2HE0, 2HE1, 2HE2, 2HE3, 2HE4, 2HE5/
DATA HOL/2HV1, 2HC1, 2HA, 2HB, 2HC, 2HD, 2HV3, 2HC3, 2HF, 2HG, 2HV4/
97 FORMAT(I5, B12.4)
98 FORMAT(/)
99 FORMAT(8E10.0)
D=0.
EPS=1.E-05
DO 23 L=1, 3
AH(L)=0.
BH(L)=0.
23 CH(L)=0.
LH=1
READ(5, 100) PI, TI, NI, THI, EM1, PINF
READ(5, 100) PF, TF, WF, THF, EMF, GF
READ(5, 100) XV1, YV1, XV2, XC1, YC1, XC2, DVV, DNUC
READ(5, 100) XF1
READ(5, 100) XTHX, YLFT, XMOM, XVTHX, YVLFT, XVMOM, XSHFT, YSHFT
READ(5, 5921) ICF, OXC, JTF, DTH, IVIS, IT
IF(IVIS.EQ.1) READ(5, 100) XSTR, REC, SH, RT
IF(IVIS.EQ.1.AND.IT.EQ.0) READ(5, 100) AH(1), BH(1), CH(1)
5921 FORMAT(I5, E10.0, I5, F10.0, 3I5)
HI=FH(PI, NI, TI)
PHI=NI
GI=FGAM(TI, PI, NI)
RHI=RHEG(HI, PI, NI, DUM)
AI=SQRT(GI*PI/RHI)
UI=EM1*AI
XTHXI=XTHX
YLFTHI=YLFT
XMOMI=XMOM
XVTHXI=XVTHX
YVLFTI=YVLFT
XVMOMI=XVMOM
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)
ZV2=ZV1

```



```

5922 FORMAT(1H177777)
WRITE(6,1020)
1020 FORMAT( 14X*E Q U I L I B R I U M F L O W N O Z Z L E D E
1 S I G N*)
WRITE(6,400) PI,THI,EMI,TI,GI,WI,PINF
400 FORMAT( //30X,*INITIAL PROFILE*//3X*PRESSURE*7X*THETA*8X*MACH*
16X*TEMPERATURE*5X*GAMMA*7X*PHI*8X*PINF*//7E13.5)
WRITE(6,5930) XV1,YV1,XV2,XC1,YC1,XC2,DYV
5930 FORMAT(//75X*XV1*10X*YV1*10X*XV2*10X*XC1*10X*YC1*10X*XC2*10X*DYV*//
17E13.5//)
1800 FORMAT(* THRUST= *E13.5,7X*LIFT = *E13.5,7X*MOMENT= *E13.5)
WRITE(6,1800)XTHX,YLFT,XMOM
1900 FORMAT(* VISCOUS THRUST= *E13.5,7X*VISCOUS LIFT= *E13.5,7X*
1VISCOUS MOMENT= *E13.5)
WRITE(6,1900)XVTHX,YVLFT,XVMOM
WRITE(6,5923)XSHFT,YSHFT
5923 FORMAT(9X*MOMENT AXIS*/12X*X ==E13.5/12X*Y ==E13.5//)
WRITE(6,1001) AZ,BZ,CZ
1001 FORMAT(9X*LATERAL EXPANSION EQUATION*/9X,22HZ(X) = AZ*X**2+BZ*X+CZ
.1/12X*AZ ==E13.5/12X*BZ ==E13.5/12X*CZ ==E13.5)
RG=49800.
RGAS=RG/WF
CPI=G F*RGAS/(G F-1.)
XC2=XC2
THC1=THI-DNUC
DO 5000 IXC=1,ICF
XC2=XC22+FLOAT(IXC-1)*DXC
YC2=YC1+TAN(THC1)*(XC2-XC1)
YV2=YC2+DYV
DNUV=ATAN((YV2-YV1)/(XV2-XV1))-THI
CALL THM(RHI,UI,PI,HI,TI,EMI,GI,PINF,YV1,YC1,DYV,XV2,THRMAX,DNUV,
1DNUC,XC2)
WRITE(6,6364) THRMAX
6364 FORMAT(1H131X*IDEAL THRUST ==E12.4)
YC2=YC1+TAN(THC1)*(XC2-XC1)
YV2=YC2+DYV
THC=ATAN((YC2-YC1)/(XC2-XC1))
DO 6000 JT=1,JTF
XTHX=XTHXI
YLFT=YLFTI
XMOM=XMOMI
XVTHX=XVTHXI
YVLFT=YVLFTI
XVMOM=XVMOMI
KSTP=50
ISTP=50
IEND=0
IEXT=0
IFLG=0
IND=0
THX=1.E+10
THJT=FLOAT(JT-1)*DTH
DNUV=ATAN((YV2-YV1)/(XV2-XV1))-THI+THJT
WRITE(6,1894)DNUV
1894 FORMAT(* VEHICLE EXPANSION = *E13.5)
XV=XV1
AV=YV1
BV=TAN(THI+DNUV)
CV=0.
YF1=AV+BV*(XF1-XV)
IDUM=DNUV*57.3
IF(IDUM.GT.11) IDUM=11
IF(IDUM.EQ.0) IDUM=1
DELNU=DNUV/FLOAT(IDUM)
IDUM=IDUM+1
TMAX=IDUM

```

```

DO 10 I=1, IMAX
X(1, I)=XV1
Y(1, I)=YV1
Z(1, I)=GETZ(XV1)
TH(1, I)=THI+FLCAT(I-1)*DELNU
M=I-1
XNU(1, I)=FLOAT(I-1)*DELMU
IF(I.NE.1)GO TO 8
P(1, I)=PIST(1, I)=TISH(1, I)=HISEM(1, I)=EMISG(1, I)=GISQ(1, I)=UIS
IRH(1, I)=RHI
GO TO 9
8 CALL PM(DELNU, P(1, I), T(1, I), H(1, I), EM(1, I), G(1, I), Q(1, I), RH(1, I),
1P(1, M), T(1, M), H(1, M), EM(1, M), G(1, M), Q(1, M), RH(1, M))
9 XMU(1, I)=ASIN(1./EM(1, I))
10 CONTINUE
IDUM=DNUC*57.3
IF(IDUM.GT.11) IDUM=11
IF(IDUM.EQ.0) IDUM=1
DELNU=DNUC/FLOAT(IDUM)
IDUM=IDUM+1
X(2, 1)=XC1
Y(2, 1)=YC1
Z(2, 1)=GETZ(XC1)
ICWL=0
I=1
THI=THI
XNUI=0.
20 CONTINUE
TH(2, 1)=THI-FLOAT(I-1)*DELNU
XNU(2, 1)=FLCAT(I-1)*DELNU+XNU1
IF(I.NE.1)GO TO 86
P(2, 1)=PIST(2, 1)=TISH(2, 1)=HISEM(2, 1)=EMISG(2, 1)=GISQ(2, 1)=UIS
IRH(2, 1)=RHI
GO TO 19
86 CALL PM(DELNU, P(2, 1), T(2, 1), H(2, 1), EM(2, 1), G(2, 1), Q(2, 1), RH(2, 1),
1P(1, 1), T(1, 1), H(1, 1), EM(1, 1), G(1, 1), Q(1, 1), RH(1, 1))
19 XMU(2, 1)=ASIN(1./EM(2, 1))
22 CONTINUE
K=2
35 CONTINUE
IF(K.LT.KMAX) GO TO 41
IF(IEND.EQ.1)GO TO 53
CALL VWALL(INAX, 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
AV=Y(2, KMAX)
BV=TAN(TH(2, KMAX))
CV=(YV2-AV-BV*(XV2-X(2, KMAX)))/(XV2-X(2, KMAX))**2
THV2=ATAN(BV+2.*CV*(XV2-X(2, KMAX)))
XV=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 KI=2, KMAX
XK1=XK2=XK3=XK4=0.25
M=KI-1
J=KI-1

```

```

IF(KT.EQ.KMAX)J=KT-2
IF(KT.EQ.2.OR.KT.EQ.KMAX)N=J
IF(KT.EQ.2.OR.KT.EQ.KMAX)XK3=0.
IF(KT.EQ.2.JR.KT.EQ.KMAX)XK1=XK2=XK4=.333333
CALL LTHM(X(2,M),Y(2,M),Z(2,M),X(1,N),Y(1,N),Z(1,N),X(1,J),
1Y(1,J),Z(1,J),X(2,KT),Y(2,KT),Z(2,KT),P(2,M),P(1,N),P(1,J),
2P(2,KT),Q(2,M),Q(1,N),Q(1,J),Q(2,KT),T(2,M),T(1,N),T(1,J),
3T(2,KT),TH(2,M),TH(1,N),TH(1,J),TH(2,KT),XK1,XK2,XK3,XK4,1.,
4XTHX,YLFT,XMOM,CF(M),ST(M),3)
IF(KT.EQ.KMAX)
1CALL LTHM(X(1,J),Y(1,J),Z(1,J),X(1,J),Y(1,J),0.,X(2,KT),Y(2,KT),0.
2,X(2,KT),Y(2,KT),Z(2,KT),P(1,J),P(1,J),P(2,KT),P(2,KT),Q(1,J),
3Q(1,J),Q(2,KT),Q(2,KT),T(1,J),T(1,J),T(2,KT),T(2,KT),TH(1,J),
4TH(1,J),TH(2,KT),TH(2,KT),.25,.25,.25,.25,1.,XTHX,YLFT,XMOM,
5CFU,STU,1)
16 CONTINUE
GO TO 53
41 CONTINUE
IF(ICWL.EQ.1.AND.K.EQ.2.AND.IEXT.EQ.0)CALL CWall(IMAX,KMAX,
1KSTP,ISTP)
IF(ICWL.EQ.1.AND.K.EQ.2.AND.IEXT.EQ.1)CALL CNT
L=K
IF(ICWL.EQ.1)L=K+1
IF(I.EQ.1) L =K-1
LL=K+ICWL-1
LM=L-1
IF(L .GT. ISTP.AND.IEND.EQ.1)GO TO 30
MM=K-1
CALL FIX(2,K-1,1,L ,2,K)
IF(I.EQ.1)GO TO 30
IF(K.GT.1)SLA=(Y(2,K)-Y(2,MM))/(X(2,K)-X(2,MM))
CALL GEM(X(2,MM),Y(2,MM),SLA,XC2,YC2,SLE,XC,YC)
IF(XC.GE.X(2,K) .OR.XC.LT.X(2,MM)-EFS)GO TO 30
KSTP=K
WRITE(6,88)KSTP,ISTP,KMAX,IMAX
88 FORMAT(1X,4I5)
30 CONTINUE
LM=L-1
IF(I.EQ.1)GO TO 33
IF(IEXT.EQ.1)GO TO 32
IF(ICWL.EQ.0.OR.K.GT.2)GO TO 32
CALL LTHM(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(1,1),T(2,1),T(2,1),T(1,1),TH(1,1),TH(2,1),
3TH(2,1),TH(1,1),.25,.25,.25,.25,1.,XTHX,YLFT,XMOM,CFL,STL,2)
CALL LTHM(X(1,1),Y(1,1),Z(1,1),X(1,2),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(2,1),T(2,1),TH(1,1),TH(1,2),
3TH(2,1),TH(2,1),.33333,.33333,.33333,0.,1.,XTHX,YLFT,XMOM,
4CF(1),ST(1),5)
32 CONTINUE
XK1=XK2=XK3=XK4=.25
IF(IEND.EQ.0.AND.K.EQ.KMAX)GO TO 17
AVG=4.
SLB=(Y(2,K)-Y(1,L))/(X(2,K)-X(1,L))
CALL GEM(X(1,L),Y(1,L),SLB,XC2,YC2,SLE,XC,YC)
RAT=(XC-X(1,L))/(X(2,K)-X(1,L))
IF(RAT.GT.0.99)GO TO 144
IF(RAT.GT.0.)GO TO 143
AVG=2.
GO TO 144
143 AVG=AVG-1.
144 CONTINUE
IF((X(2,MM)-X(1,LM)).EQ.0.)GOTO 148

```

```

RAT=(XC-X(1,LM))/(X(2,MM)-X(1,LM))
IF(RAT.GT.0.99)GO TO 148
IF(RAT.GT.0.)GO TO 145
AVG=AVG-2.
GO TO 146
145 AVG=AVG-1.
148 CONTINUE
AVG=AVG/4.
CALL LTHM(X(2,MM),Y(2,MM),Z(2,MM),X(1,LM),Y(1,LM),Z(1,LM),X(1,L),
1Y(1,L),Z(1,L),X(2,K),Y(2,K),Z(2,K),P(2,MM),P(1,LM),P(1,L),P(2,K),
2Q(2,MM),Q(1,LM),G(1,L),G(2,K),T(2,MM),T(1,LM),T(1,L),T(2,K),
3TH(2,MM),TH(1,LM),TH(1,L),TH(2,K),XK1,XK2,XK3,XK4,AVG,XTHX, YLFT,
4XMOM,CF(LL),ST(LL),3)
IF(IEND.EQ.1.OR.K.LT.KMAX)GO TO 33
17 CONTINUE
M=K-1
L=M
IF(ICWL.EQ.1)L=K
AVG=3.
SLA=(Y(2,K)-Y(2,M))/(X(2,K)-X(2,M))
CALL GEM(X(2,M),Y(2,M),SLA,XC2,YC2,SLE,XC,YC)
RAT=(XC-X(2,M))/(X(2,K)-X(2,M))
IF(RAT.GT.0.99)GO 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,SLE,XC,YC)
RAT=(XC-X(1,L))/(X(2,M)-X(1,L))
IF(RAT.GT.0.99)GO TO 178
IF(RAT.GT.0.)GO TO 173
AVG=AVG-2.
173 AVG=AVG-1.
178 CONTINUE
AVG=AVG/3.
CALL LTHM(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(2,M),P(1,L),P(2,K),P(2,K),Q(2,M),
2Q(1,L),Q(2,K),Q(2,K),T(2,M),T(1,L),T(2,K),T(2,K),TH(2,M),TH(1,L),
3TH(2,K),TH(2,K),.33333,.33333,.33333,0.,AVG,XTHX, YLFT, XMOM,
4CF(LL),ST(LL),3)
43 CALL LTHM(X(1,L),Y(1,L),Z(1,L),X(1,L),Y(1,L),0.,X(2,K),Y(2,K),
10.,X(2,K),Y(2,K),Z(2,K),P(1,L),P(1,L),P(2,K),P(2,K),Q(1,L),Q(1,L),
2Q(2,K),Q(2,K),T(1,L),T(1,L),T(2,K),T(2,K),TH(1,L),TH(1,L),TH(2,K),
3TH(2,K),.25,.25,.25,.25,1.,XTHX, YLFT, XMOM,CFU,STU,1)
33 IF(IEND.EQ.1)KMAX=ISTP
K=K+1
IF(K.LE.KMAX)GO TO 36
53 KP=KMAX
IF(IEND.EQ.1)KP=KMAX-1-ICWL
IF(IND.EQ.1)IEND=1
IND=0
WRITE(6,6885)
6885 FORMAT(1X*PT.*6X**11X*Y*8X*PRESSURE*5X*ANGLE*8X*MACH*4X*TEMPERATU
IRE*1X*PM FJUNCTION*3X*VELOCITY*)
DO 50 L=1,KP
WRITE(6,97) L,X(2,L),Y(2,L),P(2,L),TH(2,L),EM(2,L),T(2,L),XW(2,
1L),Q(2,L)
50 CALL SWITCH(2,L,1,L)
WRITE(6,2001)CFU,STU,CFL,STL
2001 FORMAT(* VEH. FRIC. COEF= *E13.5,2X*VEH. STANTON NUM.= *E13.5,2X*
1COWL FRIC. COEF.= *E13.5,2X*COWL STANTON NUM.= *E13.5)
WRITE(6,450)
NPR1=IMAX/5
DO 166 K=1,NPR1
K1=K$K2=K1+NPR1$K3=K2+NPR1$K4=K3+NPR1$K5=K4+NPR1
KDONE=K5

```

```
166 CONTINUE
IF(KDUNE.EQ.IMAX)GO TO 55
NPR2=5*NPR1+1
DO 414 K=NPR2,IMAX
WRITE(6,550)CF(K),ST(K)
414 CONTINUE
55 CONTINUE
450 FORMAT(1X,5(6X,*CF *,9X,* ST *,3X))
500 FORMAT(1X,10E13.5)
550 FORMAT(105X,2E13.5)
WRITE(6,98)
WRITE(6,1800)XTHX,YLFT,XMDM
WRITE(6,1900)XVTHX,YVLFT,XVMDM
IMAX=KMAX
I=I+1
IF(I.GT.IDUM)ICWL=1
IF(ICWL.EQ.0)KMAX=KMAX+1
IF(IEND.EQ.1)ISTP=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.0.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(*THV2.LT.0*)
GO TO 6000
5900 CONTINUE
IF(P(1,1).LE.PF)GO TO 5396
CALL COWL(DNUE)
IDUM=6
I=2
XNUI=XNU(1,1)
THII=TH(1,1)
ICWL=0
KMAX=KMAX+1
DELNU=DNUE/FLOAT(IDUM)
IDUM=IDUM+1
IEXT=1
GO TO 20
5396 CONTINUE
6000 CONTINUE
5000 CONTINUE
END
```

```

SUBROUTINE C.WALL(CMAX,RMAX,KSTP,ISTP)
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),XMU(2,50),Q(2,50),H(2,50),RH(2,50),
2G(2,50)
COMMON/C/DNUV,DNUC,XC1,YC1,XV1,YV1,DNUL
COMMON/CWL/XC2,YC2
D=0.
IADD=0
LOOP=0
XTHM1=TH(1,1)-XMU(1,1)
Q1=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),Q(1,2),RH(1,2),XMU(1,2),
2D,D,D,D,D,D,D,D,D,D,D,D,
2XA,YA,ZA,THA,XNUA,PA,TA,HA,EMA,GA,UA,RHA,XMUA,-1.)
Q2=XNU(1,2)-TH(1,2)
44 XTHM2=THA-XMUA
IF (LOOP.EQ.0)XTHM3=XTHM2
SLA=(TAN(XTHM2)+TAN(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
XA=(X(1,2)+X(1,1))/2.
24 RAT=(XA-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((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(* TOO MANY ITER IN CWALL *)
WRITE(6,39)LOOP,SLU,SLB,XA,XAT
39 FURNAT(1X,I5,5E13.5)
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),Q(1,1),RH(1,1),XMU(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),
3Q(1,2),RH(1,2),XMU(1,2),XA,YA,ZA,THA,XNUA,PA,TA,HA,EMA,GA,UA,
4RHA,XMUA,-1.)
43 RA=THA+XNUA
IF (ABS(XC2-XC).LE..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),Q(2,1),RH(2,1),
1PA,TA,HA,EMA,GA,UA,RHA)
RQ3D=RH(2,1)*W(2,1)/Z(2,1)
DNU3=DNU
IIT=1

```

```

ERT=.001
10 CONTINUE
CALL PM(DNU3,P(2,1),T(2,1),H(2,1),EM(2,1),G(2,1),Q(2,1),RH(2,1),
1PA,TA,HA,EMA,GA,UA,RHA)
RQT=RH(2,1)*Q(2,1)
ERQ=(RQT-RQ3D)/RQ3D
IF(ITT.GE.10)ERT=.005
IF(ABS(ERQ).LT.ERT)GO TO 20
CALL ERROR(1,ITT,DNU3,ERQ,1.1,DNUZ1,ERQZ1)
GO TO 10
20 CONTINUE
XMU(2,1)=ASIN(1./EM(2,1))
XTHM3=TH(2,1)-XMU(2,1)
IF(LOOP.EQ.1)GO TO 6
LOOP=1
GO TO 44
6 IF(IADD.EQ.0)RETURN
IMAX=IMAX+1
KMAX=KMAX+1
KSIP=KSIP+1
ISTP=ISTP+1
DO 46 JJ=3,IMAX
LJ=IMAX-JJ+3
L1=LJ-1
46 CALL SWITCH(1,L1,1,LJ)
WRITE(6,181)IADD,XA,X(1,1),X(1,2)
181 FORMAT(1X,I5,3E15.5)
CALL INT(0.,XA,YA,ZA,THA,XNUA,PA,TA,HA,EMA,GA,UA,RHA,XMUA,
ID,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),H(1,2),EM(1,2),G(1,2),Q(1,2),RH(1,2),XMU(1,2),-1.)
RETURN
END

```

```

SUBROUTINE ENDD(L, KK, K6)
COMMON/V/AV, BV, CV, XV, YV2, Y12
COMMON/X/X(2,50), Y(2,50), Z(2,50), TH(2,50), XMU(2,50),
1P(2,50), T(2,50), EM(2,50), XMU(2,50), Q(2,50), H(2,50), RH(2,50),
2G(2,50)
ITT=1
B=0.
A=1.
XM1=TH(1,L)+XMU(1,L)
XM2=TH(2, KK)+XMU(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
12 RAT=(XA-X(1,L))/DX
SLP=XM1+RAT*(XM2-XM1)
SLP=A*SLP+B*XM3
55 CALL GEM(X(1,L), Y(1,L), SLM, XV2, YV2, SLP, XAT, YA)
ER=ABS((XA-XAT)/DX)
IF(ER.LT.1.E-03)GO TO 14
ITT=ITT+1
IF(ITT.LE.10)GO TO 8
WRITE(3,33)
33 FORMAT(* TOO MANY ITER IN END *)
WRITE(6,34)L, KK, K6, XA, XAT, X(2, KK), X(1,L), SLP, XM3
34 FORMAT(1X, 3I5, 6E13.5)
STOP
8 XA=XAT
GO TO 12
14 QA=Q1+RAT*(Q2-Q1)
CALL INT(RAT, X(1,L), Y(1,L), Z(1,L), TH(1,L), XMU(1,L), P(1,L),
1T(1,L), H(1,L), EM(1,L), G(1,L), Q(1,L), RH(1,L), XMU(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),
3EM(2, KK), G(2, KK), Q(2, KK), RH(2, KK), XMU(2, KK), XA, YA, ZA, THA, XNUA,
4PA, TA, HA, EMA, GA, UA, RHA, XMUA, I.)
RA=XNUA-THA
X(2, K6)=XV2
Y(2, K6)=YV2
Z(2, K6)=GETZ(XV2)
TH(2, K6)=ATAN (BV+2.*CV*(XV2-XV))
XNU(2, K6)=RA+TH(2, K6)
DNU=XNU(2, K6)-XNUA
CALL PM(DNU, P(2, K6), T(2, K6), H(2, K6), EM(2, K6), G(2, K6), Q(2, K6),
1RH(2, K6), PA, TA, HA, EMA, GA, UA, RHA)
RQ3D=RH(2, K6)*J(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), Q(2, K6),
1RH(2, K6), PA, TA, HA, EMA, GA, UA, RHA)
RQT=RH(2, K6)*Q(2, K6)
ERQ=(RQT-RQ3D)/RQ3D
IF(IT.GE.10)ERT=.005
IF(ABS(ERQ).LT.ERT)GO TO 16
CALL ERROR(2, IT, DNU3, ERQ, 1.1, DNUZ1, ERQZ1)
GO TO 10
16 XMU(2, K6)=ASIN(1./EM(2, K6))
IEND=1
IF(B.GT.0.)RETURN

```


B=.5

A=.5

XM3=TH(2,K5)+XMU(2,K6)

GO TO 12

END

```
1 SUBROUTINE INT(RAT,X1,Y1,Z1,TH1,XNU1,P1,T1,H1,EM1,G1,U1,RH1,XMU1,
2 X2,Y2,Z2,TH2,XNU2,P2,T2,H2,EM2,G2,U2,RH2,XMU2,X3,Y3,Z3,TH3,
3 2XNU3,P3,T3,H3,EM3,G3,U3,RH3,XMU3,OPT)
4 HT1=H1+U1*J1/2.
5 HT2=H2+U2*J2/2.
6 Q1=XNU1+OPT*TH1
7 Q2=XNU2+OPT*TH2
8 X3=X1+RAT*(X2-X1)
9 Y3=Y1+RAT*(Y2-Y1)
10 Q3=Q1+RAT*(Q2-Q1)
11 P3=P1+RAT*(P2-P1)
12 T3=T1+RAT*(T2-T1)
13 XNU3=XNU1+RAT*(XNU2-XNU1)
14 TH3=(Q3-XNU3)/OPT
15 HT3=HT1+RAT*(HT2-HT1)
16 U3=U1+RAT*(U2-U1)
17 G3=G1+RAT*(G2-G1)
18 H3=H1+RAT*(H2-H1)
19 RH3=RH1*(P3/P1)**(1./G3)
20 EM3=U3/SQRT(G3*P3/RH3)
21 XMU3=ASIN(1./EM3)
22 Z3=GETZ(X3)
23 RETURN
24 END
```

```
SUBROUTINE THM(RHI,UI,PI,HI,TI,EMI,GI,PINF,YV1,YC1,DYV,XV2,THRMAX,  
1DNUV,DNUC,XC2)
```

```
COMMON/D3/AZ,BZ,CZ
```

```
A1=YV1-YC1
```

```
F1=RHI*UI*A1
```

```
Z2=AZ*XV2*XV2+BZ*XV2+CZ
```

```
Z5=AZ*XC2*XC2+BZ*XC2+CZ
```

```
A2=(Z2+Z5)/2.*DYV
```

```
IT=1
```

```
DNU=(DNUV+DNUC)*SQRT((Z2+Z5)/2.)
```

```
10 CALL PM(DNU,P2,T2,H2,EM2,G2,U2,RH2,PI,TI,HI,EMI,GI,UI,RHI)
```

```
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-PINF+RHI*UI*UI)*A1
```

```
TH2=(P2-PINF+RH2*U2*U2)*A2
```

```
THRMAX=TH2-TH1
```

```
RETURN
```

```
END
```

SUBROUTINE FM3D(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=EM*SQRT(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),
IP(2,50),T(2,50),EM(2,50),XMU(2,50),Q(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)
Q(I1,L)=Q(I2,K)
RETURN
END
```

```
FUNCTION GETZ(X)  
COMMON/D3/ AZ,BZ,CZ  
GETZ=AZ*X**2+BZ*X+CZ  
RETURN  
END
```

```

SUBROUTINE VWALL (IMAX,KMAX)
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),XMU(2,50),Q(2,50),H(2,50),RH(2,50),
2G(2,50)
COMMON/V/AV,BV,CV,XV,XV2,YV2
COMMON/C/DNUV,DNUC,XC1,YC1,XV1,YV1,DNUL
IBEG=0
ITW=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,
1XG,YG)
12 YW=AV+BV*(XG-XV)+CV*(XG-XV)**2
ERW=(YW-YG)/(YV1-YC1)
IF(ABS(ERW).LT.1.E-02) GO TO 10
CALL ERROR(3,ITW,XG,ERW,.9,XG1,ERW1)
YG=Y(2,KMAX-1)+SL13*(XG-X(2,KMAX-1))
10 TH(2,KMAX)=ATAN(BV+2.*CV*(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)=GETZ(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),Q(2,K),RH(2,K),
1P(2,M),T(2,M),H(2,M),EM(2,M),G(2,M),Q(2,M),RH(2,M))
RQ3D=RH(2,K)*Q(2,K)/Z(2,K)
DNU3=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),Q(2,K),RH(2,K),
1P(2,M),T(2,M),H(2,M),EM(2,M),G(2,M),Q(2,M),RH(2,M))
RQT=RH(2,K)*Q(2,K)
ERQ=(RQT-RQ3D)/RQ3D
IF(IT.GE.10)ERT=.005
IF(ABS(ERQ).LT.ERT)GO TO 16
CALL ERROR(4,IT,DNU3,ERQ,1.1,DNUZ1,ERQZ1)
GO TO 15
16 CONTINUE
XMU(2,K)=ASIN(1./EM(2,K))
IF(IBEG.EQ.1) RETURN
IBEG=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) 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 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),XMU(2,50),
1P(2,50),T(2,50),EM(2,50),XMU(2,50),Q(2,50),H(2,50),RH(2,50),
2G(2,50)
DUM3=0.
DUM4=0.
A=1.
B=0.
DUM1=TAN(TH(I1,K1)+XMU(I1,K1))
DUM2=TAN(TH(I2,K2)-XMU(I2,K2))
XNU(I3,K3)=.5*(XNU(I1,K1)+XMU(I2,K2)+TH(I2,K2)-TH(I1,K1))
TH(I3,K3)=.5*(XNU(I2,K2)-XMU(I1,K1)+TH(I2,K2)+TH(I1,K1))
10 IF(B.GT.0.) DUM3=TAN(TH(I3,K3)+XMU(I3,K3))
IF(B.GT.0.) DUM4=TAN(TH(I3,K3)-XMU(I3,K3))
SL1=A*DUM1+B*DUM3
SL2=A*DUM2+B*DUM4
CALL GEM(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),RH(I3,K3),P(I1,K1),T(I1,K1),H(I1,K1),EM(I1,K1),
2G(I1,K1),Q(I1,K1),RH(I1,K1))
RQ3D=RH(I3,K3)*Q(I3,K3)/Z(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),RH(I3,K3),P(I1,K1),T(I1,K1),H(I1,K1),EM(I1,K1),
2G(I1,K1),Q(I1,K1),RH(I1,K1))
RQT=RH(I3,K3)*Q(I3,K3)
ERQ=(RQT-RQ3D)/RQ3D
IF(IT.GE.10)ERT=.005
IF(ABS(ERQ).LT.ERT)GO TO 20
CALL ERROR(5,IT,DNU3,ERQ,1.1,DNUZ1,ERQZ1)
GO TO 12
20 XMU(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,EM2,G2,U2,RH2,P1,T1,HI,EMI,G1,JI,RHI)
COMMON/O/ PHI1
DTH=1./57.5
IFAN=ABS(DNU)/DTH+1
DUM=DNU
IF(IFAN.GT.1)DUM=DNU/FLGAT(IFAN-1)
P1=P1
T1=T1
EM1=EMI
H1=HI
G1=G1
RH1=RHI
A1=SQRT(G1*P1/RH1)
U1=UI
U1=U1*U1
HT=H1+U1/2.
P1=ALOG(P1)
XNU=0.
XNU1=0.
IF(IFAN.NE.1)IFAN=IFAN-1
DO 10 I=1,IFAN
XMU1=ASIN(1./EM1)
B1=G1*EM1/COS(XMU1)
P2=-B1*DUM+P1
RH2=(P2-P1)/G1
RH2=RH1*EXP(RH2)
P2P=EXP(P2)
P1P=EXP(P1)
J2=U1-2.*G1/(G1-1.)*(P2P/RH2-P1P/RH1)
H2=HT-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
J1=J2
RH1=RH2
10 CONTINUE
P2=EXP(P2)
U2=SQRT(J2)
RETURN
END
END

```

```

SUBROUTINE PMI(IX,KX)
COMMON/IV/PI, TI, AI, THI, FMI, GI, ZI, HI, RHI, UI
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), XMU(2,50), Q(2,50), H(2,50), Rh(2,50),
2G(2,50)
COMMON/P/PINF, PTOT, TTOT
COMMON/CP/CPI, RGAS
DNU=XNU(IX,KX)
G=GI
EM1=EMI
GG=SQRT((G+1.)/(G-1.))
XM1=SQRT(EM1**2-1.)
XNU1=GG*ATAN(XM1/GG)-ATAN(XM1)
EM2=DNU/(1.5-XNU1)*(6.-EM1)+EM1
IT3=1
10 XM2=SQRT(EM2**2-1.)
DNU1=GG*(ATAN(XM2/GG)-ATAN(XM1/GG))+ATAN(XM1)-ATAN(XM2)
ERNU=DNU-DNU1
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 EM3D(EM2, EMX, X(IX,KX), G, Z(IX,KX), ZI)
P(IX,KX)=PTOT/(1.+(G-1.)/2.*EMX**2)**(G/(G-1.))
T(IX,KX)=TTOT/(1.+(G-1.)/2.*EMX**2)
EM(IX,KX)=EMX
XMO(IX,KX)=ASIN(1./EMX)
AX=SQRT(G*RGAS*T(IX,KX))
Q(IX,KX)=EMX*AX
RETURN
END

```

```

SUBROUTINE GNURE(RH,Q,P,T,R,          X,X1,CF,ST,L)
COMMON/CP/CPI,          RGAS
COMMON/O/PHI1
COMMON/HOT/AH(3),BH(3),CH(3),XSTR,  REC,          RT,SH,IT,IVIS
HDEL=R-Q*Q/2.
HAW=1.+REC*Q*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,PHI1,TW)
HW=HW/HDEL
GO TO 48
46 HW=HAW
48 A=HAW-1.
B=HW-1.
C=SQRT((A+B)**2+4.*A)
FC=A/(ASIN((A-B)/C)+ASIN((A+B)/C))**2
FRX=HAW**(.772)/(FC*(HW)**(1.474))
CALL VIS(T,XMMU)
REX=RH*Q*(X+XSTR)*RT /XMMU
REXI=FRX*REX
CFI=.088*(ALOG10(REXI)-2.3686)/(ALOG10(REXI)-1.5)**3
CF=CFI/FC
ST=CF*SH/2.
RETURN
END
SUBROUTINE VIS(T,XMMU)
XMMU=2.27*T**1.5*1.E-08/(T+198.6)
RETURN
END

```

```
SUBROUTINE SWARF(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,AVX,AVY,AVZ,  
1XNX,XNY,XNZ,AS,X0,Y0,Z0,LH)
```

```
DIMENSION XPA(4),YPA(4),ZPA(4),XI(4),ETA(4)
```

```
XPA(1)=X1
```

```
YPA(1)=Y1
```

```
ZPA(1)=Z1
```

```
ZPA(2)=Z2
```

```
YPA(2)=Y2
```

```
XPA(2)=X2
```

```
XPA(3)=X3
```

```
YPA(3)=Y3
```

```
ZPA(3)=Z3
```

```
ZPA(4)=Z4
```

```
YPA(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=T1X*T2Z-T2X*T1Z
```

```
XNZ=T2X*T1Y-T1X*T2Y
```

```
VN=SQRT(XNX**2+XNY**2+XNZ**2)
```

```
IF(VN.LE.1.E-13)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=SQRT(T1X*T1X+T1Y*T1Y+T1Z*T1Z)
```

```
T1X=T1X/T
```

```
T1Y=T1Y/T
```

```
T1Z=T1Z/T
```

```
T2X=XNY*T1Z-XNZ*T1Y
```

```
T2Y=XNZ*T1X-XNX*T1Z
```

```
T2Z=XNX*T1Y-XNY*T1X
```

```
DO 1000 J=1,4
```

```
XPA(J)=XPA(J)+XNX*D
```

```
YPA(J)=YPA(J)+XNY*D
```

```
ZPA(J)=ZPA(J)+XNZ*D
```

```
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*XDIF+T2Y*YDIF+T2Z*ZDIF
```

```
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)-XIO
```

```
1020 ETA(J)=ETA(J)-ETA0
```

```
X0=AVX+T1X*XIO+T2X*ETA0
```

```
Y0=AVY+T1Y*XIO+T2Y*ETA0
```

```
Z0=AVZ+T1Z*XIO+T2Z*ETA0
```

```
AS=(ETA(2)-ETA(4))*(XI(3)-XI(1))/2.
```

```
AS=ABS(AS)
```

```
RETURN
```

```
6 CONTINUE
```

```
XU=AVX $ Y0=AVY $ Z0=AVZ
```

AS=0.
RETURN
END

```
SUBROUTINE TH4(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,P1,P2,P3,P4,  
I01,Q2,Q3,Q4,T1,T2,T3,T4,TH),TH2,TH3,TH4,XK1,XK2,XK3,XK4,AVG,  
2XTHX,XYLFT,XXMOM,CFF,ST,LH)  
COMMON/O/PHI1  
COMMON/P/PINF,PTOT,TTOT  
COMMON/CP/CPI, RGAS  
COMMON/SHF/XSHFT,YSHFT  
COMMON/HOT/AH(3),BH(3),CH(3),XSTR, REC, RT,SH,IT,IVIS  
COMMON/VISF/XVTHX,YVLFT,XVMOM  
P=XK1*P1+XK2*P2+XK3*P3+XK4*P4  
Q=XK1*Q1+XK2*Q2+XK3*Q3+XK4*Q4  
T=XK1*T1+XK2*T2+XK3*T3+XK4*T4  
TH=XK1*TH1+XK2*TH2+XK3*TH3+XK4*TH4  
H=FH(P,PHI1,T)  
RH=RHEQ(H,P,PHI1,DUM)  
R=H+Q*Q/2.  
R=CPI*T+Q*3/2.  
AVX=XK1*X1+XK2*X2+XK3*X3+XK4*X4  
AVY=XK1*Y1+XK2*Y2+XK3*Y3+XK4*Y4  
AVZ=XK1*Z1+XK2*Z2+XK3*Z3+XK4*Z4  
CALL SNARF(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,AVX,AVY,AVZ,XNX,  
1XNY,XNZ,ASS,XO,YO,7(LH))  
CFF=0.  
XBP=0.  
IF(IVIS.EQ.1)CALL GNURE(RH,Q,F,T,R, XO,XBP,CFF,ST,1 )  
RHQ=RH*Q*Q/2.  
PAV=P-PINF  
DXTHX=-PAV*XNX*ASS  
DYLFT= PAV*XNY*ASS  
XNZZ=1.  
IF(LH.EQ.3)XNZZ=XNZ  
DXTHXV=-CFF *COS(TH)*ASS*RHQ  
DYLFTV=CFF *SIN(TH)*ASS*RHQ  
DXTHXV=DXTHXV*AVG  
DYLFTV=DYLFTV*AVG  
XMS=XO-XSHFT  
YMS=Y0-YSHFT  
DMOMV=YMS*DXTHXV+XMS*DYLFTV  
XVTHX=XVTHX+DXTHXV  
YVLFT=YVLFT+DYLFTV  
XVMOM=XVMOM+DMOMV  
DXTHX=DXTHX*AVG+DXTHXV  
DYLFT=DYLFT*AVG+DYLFTV  
DMOM=YMS*DXTHX+XMS*DYLFT  
XXTHX=XXTHX+DXTHX  
XYLFT=XYLFT+DYLFT  
XXMOM=XXMOM+DMOM  
RETURN  
END
```

SUBROUTINE CNT

COMMON/IN/PI, TI, XI, TH1, EM1, GI, ZI, OI, RHI, OI

COMMON/CNTC/TTC, PTC, AC, GC, PC, THC, XNUC, ZC, EMC

COMMON/P/PIVF, PTOT, ITOT

COMMON/CP/CPI, RGAS

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), XMU(2,50), Q(2,50), H(2,50), RH(2,50),
2G(2,50)

A=1. & B=0.

GIS=GI \$ZIS=ZI\$EMIS=EM1

ITT=1

RB=XNU(1,2)+TH(1,2)

TH(2,1)=TH(1,1)

20 SLA=A*TAN(TH(2,1))+B*TAN(TH(1,1))

SLB=TH(1,2)-XMU(1,2)

SLB=A*TAN(SLB)

IF(B.GT.0.)SLB=SLB+B*TAN(TH(2,1)-XMU(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)=RB-TH(2,1)

DNU=XNU(2,1)-XNU(1,1)

CALL PM(DNU, P(2,1), T(2,1), H(2,1), EM(2,1), G(2,1), Q(2,1), RH(2,1),

1P(1,1), T(1,1), H(1,1), EM(1,1), G(1,1), Q(1,1), RH(1,1))

RQ3D=RH(2,1)*Q(2,1)/Z(2,1)

DNU3=DNU

ERT=.001

IT=1

10 CONTINUE

CALL PM(DNU3, P(2,1), T(2,1), H(2,1), EM(2,1), G(2,1), Q(2,1), RH(2,1),

1P(1,1), T(1,1), H(1,1), EM(1,1), G(1,1), Q(1,1), RH(1,1))

RQT=RH(2,1)*Q(2,1)

ERQ=(RQT-RQ3D)/RQ3D

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

IF(ABS(ERQ).LT.ERT)GO TO 14

CALL ERROR(7, IT, DNU3, ERQ, 1.1, DNUZ1, ERQZ1)

GO TO 10

4 XMU(2,1)=ASIN(1./EM(2,1))

ITOT=TTC & PTOT=PTC & GI=GC & ZI=ZC & EM1=EMC

RC=XNUC-THC

TH(2,2)=TH(2,1)

XNU(2,2)=RC+TH(2,2)

X(2,2)=X(2,1)

Y(2,2)=Y(2,1)

Z(2,2)=GETZ(X(2,2))

CALL PMI(2,2)

ER3=(P(2,1)-P(2,2))/PC

IF(ABS(ER3).LT.1.E-04)GO TO 16

CALL ERROR(8, ITT, TH(2,1), ER3, 1.01, TH23, ER23)

GI=GIS \$ZI=ZIS \$EM1=EMIS

GO TO 20

16 CONTINUE

GI=GIS \$ZI=ZIS \$EM1=EMIS

IF(B.GT.0.)RETURN

ITT=1

A=.5 & B=.5

GO TO 20

END


```

SUBROUTINE COWL(DNUE)
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),XMF(2,50),O(2,50),H(2,50),RH(2,50),
2G(2,50)
COMMON/CWL/XC2,YC2
COMMON/P/PI,VF,PTOT,ITOT
COMMON/F/PF,IF,WF,THF,EMF,GF
COMMON/CP/CP1,      RGAS
COMMON/CNTC/TTC,PTC,WC,GC,PC,THC,XNUC,ZC,EMC
PCC=(PF+P(1,1))/2.
GM1=GF-1.
GP1=GF+1.
EMF2=EMF*EMF
DUM=1.+GM1*EMF2/2.
PTF=PF*DUM**{(GF/GM1)}
ITF=IF*DUM
GIS=GI
XNUF=ASIN(1./EMF)
IT=1
BET=(THF-XNUF+TH(1,1))*1.1-THF
20 DUM=(EMF*SIN(BET))**2
PC=(2.*GF*DUM-GM1)/GP1
TC=PC*(GM1*DUM+2.)/DUM/GP1
EMC=(EMF2*(GP1*PC+GM1)-2.*(PC*PC-1.))/(GM1*PC+GP1)/PC
EMC=SQRT(EMC)
PC=PC*PF
THC=ATAN(BET)*(GP1*EMF2/(DUM-1.)/2.-1.)
THC=ATAN(1./THC)+THF
PTC=(2.*TC/(GM1*EMF2+2.))**{(GF/GM1)}
PTC=PC/PTC
TTC=ITF
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)
DNUE=XNU(2,1)-XNU(1,1)
CALL P4(DNU,P(2,1),T(2,1),H(2,1),EM(2,1),G(2,1),O(2,1),RH(2,1),
IP(1,1),T(1,1),H(1,1),EM(1,1),G(1,1),O(1,1),RH(1,1))
ER4=(PC-P(2,1))/PCC
IF(ABS(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
WC=WF
GC=GF
ZC=Z(2,1)
XNUC=0.
RETURN
END

```

```

FUNCTION FH(P1,F,T1)
P=P1*1.01325E+05/2116.
T=T1/1.8
F2=F*F
IF(F.LT.0.) GO TO 400
IF(T.GT.2000.) GO TO 190
IF(F.GT.1.) GO TO 191
120 A=1.E-07*(-.1042*F2 +.8242*F+.987)
B=.001*(.01167*F2 +.1503*F+.938)
C=-.0284*F2 +.6731*F+.4293
GO TO 290
191 A=1.E-07*(1.787*F2 -5.48*F+5.4)
B=.001*(-.1867*F2 +1.11*F+.176)
C=-.0933*F2 +3.975*F-2.808
GO TO 290
190 IF(F.GT.1.) GO TO 192
A=.000001*(1.792*F2 +.3983*F+.31)
B=.001*(-9.05*F2 -.07917*F+.245)
C=10.86*F2 -.1183*F+.97
GO TO 290
192 A=.000001*(4.81*F2 -13.9*F+11.59)
B=.001*(-23.08*F2 +66.82*F-52.61)
C=27.05*F2 -73.73*F+58.39
290 H1=A*T*T+B*T+C
IF(T.LE.2000.) GO TO 370
A10=ALOG(P)/2.3-5.
Z9=.125*A10*A10 -.275*A10
H1=H1*(1.+(1.+F)*(T/2000.-1.)*Z9)
370 H1=H1*1.E+06
GO TO 340
400 T2=T*T
T3=T2*T
T4=T3*T
T5=T4*T
H1=A1*T+A2*T2/2.+A3*T3/3.+A4*T4/4.+A5*T5+A6
H1=H1*8314./XMM1
340 CONTINUE
FH=H1*10.7639
RETURN
END

```

```

FUNCTION FT(P1,F,H5)
DATA I63/0/
IFLAG=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.LT.0.) GO TO 400
GO TO 120
50 E0=(H-H1)/H
IF(ABS(E0).LT.1.E-04) GO TO 340
500 T =T0*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(IT.LT.21) GO TO 10
IF(ABS(T-2000.).LT.10.) GO TO 830
WRITE(6,831) P1,H5,T
831 FORMAT(* ERROR IN FT*/* P1 = *E13.5,5X,*H1 = *E13.5,5X,*T = *E13.5
1)
STOP
130 IF(IFLAG.EQ.1) GO TO 504
IFLAG=1
T0=2000.
T=2000.
IF(F.LT.0.) GO TO 400
GO TO 120
04 WRITE(6,11) E1
11 FORMAT(* TEMPERATURE IN FT SET TO 2000 - ERROR = *E13.5)
GO TO 340
10 T9=T0-E0*(T-T0)/(E1-E0)
05 E0=E1
T0=T
T=T9
IF(F.LT.0.) GO TO 400
20 A=1.E-07*(-.1042*F2 +.8242*F+.987)
B=.001*(.01167*F2 +.1503*F+.936)
C=-.0284*F2 +.6731*F+.4293
IF(F.LE.1.) GO TO 190
A=1.E-07*(1.787*F2 -5.48*F+5.4)
B=.001*(-.1807*F2 +1.11*F+.176)
C=-.0933*F2 +3.975*F-2.808
10 IF(T.LE.2000.) GO TO 290
A=.000001*(1.792*F2 +.3983*F+.31)
B=.001*(-9.05*F2 -.07917*F+.245)
C=10.86*F2 -.1183*F+.97
IF(F.LE.1.) GO TO 290
A=.000001*(4.81*F2 -13.9*F+11.59)
B=.001*(-23.08*F2 +66.82*F-52.61)

```

IF(T.LE.2000.) GO TO 370
H1=H1*(1.+(1.+F)*(T/2000.-1.)*Z9)

370 CONTINUE
GO TO 350

400 T2=T*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=-6.9126562E-03
A3=3.1602134E-05
A4=-2.9715432E-08
A5=9.5103550E-12
A6=-1.0186632E+04

GO TO 460

450 CONTINUE

A1=1.1202436
A2=1.3905716E-02
A3=2.0568374E-06
A4=-1.1560272E-08
A5=5.2386929E-12
A6=5.3328896E+03

XMM1=28.054

470 H1=A1*T+A2*T2/2.+A3*T3/3.+A4*T4/4.+A5*T5+A6

H1=H1*8314./XMM1/1.E+06

350 IF(IT.EQ.1) GO TO 50

GO TO 100

50 T0=T

FT=T*1.8

RETURN

END

FUNCTION RHEQ(H,P1,F,T)

T1=FT(P1,F,H)

T=T1/1.8

P=P1*1.01325E+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=8.61*FF-15.42*F-6.66

C=-16.88*FF+33.21*F+14.58

XN=-.4375*FF+.0625*F+2.08

D=A*(ALOG(P)/2.3)**1.5+B*(ALOG(P)/2.3)+C

XM=FNM-D*((T-2000.)/1000.）**XN

IF(F.LT.1.) GO TO 2160

A=-.822*FF+2.363*F+1.905

B=2.76*FF-7.56*F-8.68

C=-3.6*FF+7.36*F+27.15

XN=-.47*FF+1.825*F+.35

D=A*(ALOG(P)/2.3)**1.5+B*(ALOG(P)/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 RHEQ=P*XM/T/8314.3*6.2428E-02/32.174

T=T*1.8

RETURN

END

FUNCTION FGAM(T1,P1,F)

T=T1/1.8

T2=T*T

P=P1*1.01325E+05/2116.

XM=0.

IF(F.LT.0.) GO TO 550

IF(T.LE.1000.) GO TO 440

XM=-2.15E-08*T2 +.000091*T-.0695

440 XN=4.E-09*T2 -.00002*T-.019

IF(F.LE.1.) GO TO 470

XN=.0339*SQRT(T)-.000391*T-.681

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-08*T2 -.000457*T+1.85

20 G=G+XM*(ALOG(P)/2.3-5.)+XN*(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

#							
926.3	3970.2	1.	0.	2,776	21,443		
52.90	488.8	28,965	0.	8,636	1.4		
0.	1.	18.54	1.11	0.	3.12	7,866	6.
6.							
5	.5	5	1.				
0.	0.	1.					