[redacted]

(NASA-CR-132440) A NUMERICAL PROCEDURE FOR THE PARAMETRIC OPTIMIZATION OF THREE DIMENSIONAL SCRAMJET NOZZLES (Advanced Technology Labs., Inc., Westbury, N.Y.) ~~81~~ p HC $7.25 CSCL 20D

88

N74-22926

Unclas
G3/12 39395

ADVANCED TECHNOLOGY LABORATORIES, INC.

NASA CR-132440

N74 22926

OCTOBER 1973

ATL TR 188

A NUMERICAL PROCEDURE FOR THE PARAMETRIC OPTIMIZATION OF THREE DIMENSIONAL SCRAMJET NOZZLES

By

S. Dash, P. Del Guidice
and P. Kalben

PREPARED FOR

NATIONAL AERONAUTICS AND SPACE ADMINISTRATION
LANGLEY RESEARCH CENTER
HAMPTON, VIRGINIA 23365

BY

ADVANCED TECHNOLOGY LABORATORIES, INC.
Merrick and Stewart Avenues
Westbury, New York 11590

INDEX

LIST OF FIGURES

LIST OF TABLES

I. INTRODUCTION

This report describes a numerical procedure permitting the rapid determination of the internal performance of a class of scramjet nozzle configurations. The geometric complexity of these configurations rules out attempts to employ conventional nozzle design procedures[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 approach developed is based on the construction of quasi two dimensional simple wave networks, wherein lateral expansion effects are incorporated via one dimensional approximations, as first suggested by Dr. Antonio Ferri[2]. A numerical procedure following this approach has been developed and results obtained are highly comparable to those obtained employing a characteristic procedure.

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 and vehicle undersurface and calculates the nozzle thrust, lift and pitching moment.

Due to the differing techniques required for the calculation of frozen flows as compared to equilibrium flows, two separate numerical programs have been developed. The first program analyzes constant γ frozen 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. The frozen flow program computes a complete nozzle flow field in a fraction of a second while the equilibrium program has a running time of several seconds on a CDC 6600, as compared to running times of the order of a minute for a quasi two dimensional characteristic approach[3] and twenty minutes for a three dimensional numerical calculation[4].

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 procedures described in Reference (3).

For this configuration, the total amount of expansion from entrance conditions is known (based on two dimensional considerations) at the grid points labeled V_1, C_1, A, B, $\overline{C}$, D, V_3, C_3, E_2, E_3, F and G and can be readily obtained at points V_2, C_2, E_1 and E_4. The numerical procedure predicts the location of these grid points based on three dimensional flow considerations and effectively distributes the waves on the cowl and vehicle undersurface to assess the pressure distribution. It should be noted that a significant amount of the logical procedures in the numerical program are employed to distinguish the varying types of wave situations that may be encountered. In the configurations shown the expansion waves emanating from the cowl (at C_1) and the vehicle (at V_1) are both only partially captured on the undersurface and cowl respectively. τ_V denotes the portion of the cowl expansion wave $\Delta\nu_C$ captured on the vehicle undersurface while τ_C denotes that portion of the vehicle expansion wave $\Delta\nu_V$ captured on the cowl. In other situations, these waves may be totally captured or not captured at all, hence all these various situations must be distinguishable and treated discretely.

Before treating the overall logic entailed in a nozzle calculation, the calculational procedure for a typical grid point will be described:

A. Frozen Flow 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. The total amount of expansion ($\Delta\nu_3$) based on two dimensional considerations and the flow deflection (θ_3) is known at point 3. Along 1-3 or 2-3 we have

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

Employing the two dimensional value of expansion $\Delta\nu_3$ (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)

$$\Delta\nu_3 = \nu_3-\nu_i = \sqrt{\frac{\gamma+1}{\gamma-1}}\left(\tan^{-1}\sqrt{\left(\frac{\gamma-1}{\gamma+1}\right)(M_3^2-1)} - \tan^{-1}\sqrt{\left(\frac{\gamma-1}{\gamma+1}\right)(M_i^2-1)}\right) - \left(\tan^{-1}\sqrt{M_3^2-1} - \tan^{-1}\sqrt{M_i^2-1}\right) \tag{2}$$

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

$$\mu_3 = \sin^{-1}\left(\frac{1}{M_3}\right) \tag{3}$$

Equations (1a) and (1b) yield a tentative location for point 3, and the area ratio $(A/A^*)_3$ is calculated based on two dimensional considerations

$$\left(\frac{A}{A_*}\right)_3 = \frac{M_3\left(1 + \frac{\gamma-1}{2}M_3^2\right)^{\frac{\gamma+1}{2(\gamma-1)}}}{\left(\frac{\gamma+1}{2}\right)^{\frac{\gamma+1}{2(\gamma-1)}}} \tag{4}$$

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

$$Z(x) = Ax^2 + Bx + C \tag{5}$$

where $Z_3 = Z(x_3)$ and Z_i denotes the lateral extent of the nozzle at the initial station.

$$\left(\frac{A}{A_*}\right)_3' = \left(\frac{A}{A_*}\right)_3 * Z_3/Z_i \tag{6}$$

Then, the three dimensional corrected Mach number is obtained by replacing the two dimensional area ratio in Equation (4) by the three dimensional value given by Equation (6), and solving Equation (4) for $M_{3_{3D}}$ by an iterative process. Equations (1a) and (1b) are resolved using the corrected Mach angle $\mu_{3_{3D}}$ and the entire procedure is repeated until two successive values of x_3 are within a prescribed tolerance.

A similar procedure is used to determine properties at grid points on boundaries with Equation (1a) or (1b) 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 (1a) and (1b) just as for the frozen calculation. The known two dimensional value of expansion $\Delta\nu_3$ is subdivided into a series of small $\Delta\nu_j$ increments. The initial value of isentropic exponent Γ is obtained from

$$\Gamma = \Gamma\ (P, \phi, h) \tag{7}$$

where Equation (7) has been curve fit for equilibrium hydrogen-air as described in Reference (5).

The characteristic compatibility relation

$$\frac{d\ \ln P}{\Gamma} \pm \frac{d\nu}{\sin\mu\ \cos\mu} = 0 \tag{8}$$

applied across the interval $\Delta\nu_j$ yields the pressure, holding Γ and μ equal to their values at the start of the increment. The density is obtained from the isentropic pressure-density relation

$$P/\rho^{\Gamma} = \text{constant} \tag{9}$$

The velocity is obtained from the Bernoulli relation;

$$\frac{dP}{\rho} + \frac{1}{2}\, dV^2 = 0 \tag{10}$$

the enthalpy from the constancy of stagnation enthalpy;

$$h + \frac{1}{2} V^2 = H = \text{constant} \tag{11}$$

and the Mach number from;

$$M = V/a; \quad a = \left(\frac{\Gamma P}{\rho}\right)^{\frac{1}{2}} \tag{12}$$

where Γ has been reevaluated employing Equation (7). This procedure is repeated in small steps $\Delta\nu_j$ until the full wave $\Delta\nu_3$ has been integrated. Having the two dimensional value of M_3, point 3 can be tentatively located employing Equations (1a) and (1b). Then, the two dimensional area ratio can be computed from mass flow considerations

$$(A_3/A_i)_{2D} = \rho_i V_i/\rho_3 V_3 \tag{13}$$

Since the effective area based on three dimensional considerations is

$$(A_3/A_i)_{3D} = (A_3/A_i)_{2D} * Z_3/Z_i \tag{14}$$

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

$$(\rho_3\, V_3)_{3D} = (\rho_3\, V_3)_{2D} * \frac{Z_i}{Z_3} \tag{15}$$

Then an iteration procedure is performed to determine the value of three dimensional expansion $(\Delta\nu_3)_{3D}$ the correct value being that which yields $(\rho_3\, V_3)_{3D}$ after application of the integration procedure of Equations (7) thru (12), and an update of the location of point 3 using Equations (1a) and (1b).

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 ($\Delta\nu_v$) 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 C_3 and C_2 is straight.

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

7. The cowl length and vehicle length 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 $\Delta\nu_c$, 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 is flat) 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:

1. Grid points A, B, $\overline{C}$ and D are calculated following the procedure described in the previous section. The two dimensional values of expansion from initial conditions and flow deflection angles at these points are:

$$\Delta\nu_A = 0 \quad , \quad \theta_A = \theta_i$$

$$\Delta\nu_B = \Delta\nu_C, \quad \theta_B = \theta_i - \Delta\nu_C$$

$$\Delta\nu_{\overline{C}} = \Delta\nu_V, \quad \theta_{\overline{C}} = \theta_i + \Delta\nu_V$$

$$\Delta\nu_D = \Delta\nu_C + \Delta\nu_{V_1}, \quad \theta_D = \theta_i + \Delta\nu_V - \Delta\nu_C$$

2. The amount of the vehicle expansion wave captured on the cowl (τ_C) is determined by an iterative procedure. A value of τ_C is assumed which yields the location of grid points T and W as depicted in Figure (1). The correct value of τ_C is that for which the wave from W intersects the end of the cowl precisely. A linear error extrapolation routine is used to speed convergence of this iterative process. Having determined τ_C, the two dimensional expansion at the end of the cowl (assuming no recompressions between C_3 and C_2) is

$$\Delta\nu_{c_2} = \Delta\nu_c + 2\tau_c$$

since the portion of the wave captured is fully reflected.

3. The amount of the cowl expansion wave captured on the vehicle undersurface τ_v is determined by a similar iterative procedure and the two dimensional expansion at the end of the vehicle is given by

$$\Delta\nu_{v_2} = \Delta\nu_v + 2\tau_v + \Delta\nu_r$$

where the recompression $\Delta\nu_r$ is given by

$$\Delta\nu_r = \theta_{v_2} - \theta_{v_3}$$

Note that for given nozzle parameters, the equation for the parabolic recompression between V_3 and V_2 is determined in the course of the numerical calculations. Specification of the cowl length and expansion $\Delta\nu_c$ yields x_{c_2} and y_{c_2}; the nozzle length and exit height yields x_{v_2} and y_{v_2} and the vehicle turn $\Delta\nu_v$ yields x_{v_3} and y_{v_3}. Hence, the parabola is determined for each case satisfying the conditions

y_{v_3} and θ_{v_3} specified at x_3

y_{v_2} specified at x_2

(where $\theta_{v_3} = \theta_i + \Delta\nu_v$).

4. The determination of the exit profile (E_0-E_5) is highly dependent on the particular configuration being studied. For the case depicted in Figure (1), the cowl and vehicle expansion waves are partially captured resulting in a zone of uniform two dimensional wave strength between E_2 and E_3 ($\Delta\nu_{E_2} = \Delta\nu_{E_3} = \Delta\nu_v + \Delta\nu_c$). Points

E_1 and E_4 have this value of expansion and additionally receive portions of the reflected waves. Hence, the two dimensional value of expansion experienced at E_1 is given by

$$\Delta\nu_{E_1} = \Delta\nu_{v_1} + \Delta\nu_{c_1} + f^* \tau_c$$

where
$$f = 1 - \frac{(y_{E_1} - y_{c_2})}{(y_{E_2} - y_{c_2})}$$

while that at E_4 is given by

$$\Delta\nu_{E_4} = \Delta\nu_{v_1} + \Delta\nu_{c_1} + g^* (\tau_v + \Delta\nu_r)$$

where
$$g = 1 - \frac{(y_{E_4} - y_{v_2})}{(y_{E_3} - y_{v_2})}$$

The corresponding flow deflection angles are:

$$\theta_{E_1} = \theta_i + \Delta\nu_{v_1} - \Delta\nu_{c_1} - f^* \tau_c$$

$$\theta_{E_4} = \theta_i + \Delta\nu_{v_1} - \Delta\nu_{c_1} + g^* (\tau_v + \Delta\nu_r)$$

Now consider the wave configuration depicted in Figure (4) (where the F point and G point are downstream of the exit plane and hence not required in the calculational procedure). In this situation, the two dimensional value of expansion experienced at E_1 is given by

$$\Delta\nu_{E_1} = \Delta\nu_{c_1} + \tau_c + f_1^* (\Delta\nu_{v_1} - \tau_c)$$

where $f_1 = \dfrac{y_{E_1} - y_{c_2}}{y_{E_2} - y_{c_2}}$

while that at E_4 is given by

$$\Delta\nu_{E_4} = \Delta\nu_{v_1} + \tau_v + g_1 * (\Delta\nu_{c_1} - \tau_v)$$

where

$$g_1 = \frac{y_{E_4} - y_{v_2}}{y_{E_3} - y_{v_2}}$$

The corresponding flow deflection angles are:

$$\theta_{E_1} = \theta_i - \Delta\nu_{c_1} + \tau_c + f_1 * (\Delta\nu_v - \tau_c)$$

$$\theta_{E_4} = \theta_i + \Delta\nu_{v_1} - \tau_v - g_1 * (\Delta\nu_c - \tau_v).$$

The program developed analyzes both these wave configurations and combinations of the two. In addition, it can analyze cases where the full cowl expansion is captured on the vehicle undersurface as depicted in Figure (5), and cases where no region of uniform two dimensional wave strength on the exit plane exists as depicted in Figure (6). For these latter cases, not all points at the exit plane are computed by the program due to the logical complexities imposed by the wide variety of wave configurations that may occur.

IV. DETERMINATION OF THRUST, LIFT AND PITCHING MOMENT

The thrust, lift and pitching moment are determined by a combination of integrating the pressure distribution on the cowl and vehicle undersurface and by a momentum flux calculation. Referring to Figure (1), the pressure distribution on the vehicle undersurface between V_1 and V_3 and on the cowl between C_1 and C_3 are readily determined since the two dimensional values of expansion on these surfaces are constant ($\Delta\nu_{v_1}$ and $\Delta\nu_{c_1}$ respectively). Then, the pressure is determined by accounting for lateral expansion by the procedure described in Section II (i.e., by employing Equations (1) - (6) for frozen flows and Equations (7) - (15) for equilibrium flows with the value of two dimensional expansion being constant on each surface).

Since the lift and pitching moment are quite sensitive to the pressure distribution the expansion fans from the cowl and undersurface must be distributed properly in the region V_3 to V_2 on the undersurface and C_3 to C_2 on the cowl. For points on the vehicle undersurface (as depicted in Figure 7) the value of two dimensional expansion at a point x* is given by

$$\Delta\nu^* = \Delta\nu_v + \Delta\nu_r^* + 2\tau^*$$

where

$$\Delta\nu_r^* = \theta^* - \theta_{v_3}$$

and

$$\tau^* = \tan^{-1}\left(\left[\frac{(x_{v_3}-x_{c_1})}{(x^*-x_{c_1})} - 1.\right] \Big/ \left[\frac{(x_{v_3}-x_{c_1})}{(x_{v_2}-x_{c_1})} - 1.\right] * \tan(\tau_v)\right) \tag{16}$$

In the above relation τ^* varies from o to τ_v over the interval x_{v_3} to x_{v_2}. If the full wave is captured on the undersurface as depicted in Figure (5), then τ^* would vary from o to τ_v in the interval x_{v_3} to x_{v_4} which is affected in Equation (16) by replacing x_{v_2} by x_{v_4}. Then $\tau^*=\tau_v$ between x_{v_4} and x_{v_2}.

Similarly, for points on the cowl (Figure 8) the value of two dimensional expansion at a point $\tilde{x}$ is given by

$$\Delta\tilde{\nu} = \Delta\nu_c + 2\tilde{\tau}$$

where

$$\tilde{\tau} = \tan^{-1}\left(\left[\frac{(x_{c_3}-x_{v_1})}{(\tilde{x}-x_{v_1})} - 1.\right] \Big/ \left[\frac{(x_{c_3}-x_{v_1})}{(x_{c_2}-x_{v1})} - 1.\right] * \tan(\tau_c) \right) \tag{17}$$

In this relation $\tilde{\tau}$ varies from o to τ_c in the interval x_{c_3} to x_{c_2}. Knowing the two dimensional values of expansion, the pressure distribution accounting for lateral effects is obtained as previously described.

The lift is determined solely from the pressure distribution on the cowl and vehicle undersurface assuming that the internal walls and side flow fences provide no contribution. Then, the lift is given by the integration

$$L = \int_{x_{v_1}}^{x_{v_2}} (P-P_\infty)\ Z(x)\ dx - \int_{x_{c_1}}^{x_{c_2}} (P-P_\infty)\ Z(x)\ dx \tag{18}$$

The values of thrust and pitching moment contributed by the cowl and vehicle undersurface are given by

$$Th^* = \int_{y_{v_1}}^{y_{v_2}} (P-P_\infty)\ Z\ dy - \int_{y_{c_1}}^{y_{c_2}} (P-P_\infty)\ Z\ dy \tag{19}$$

$$M^* = -\int_{y_{v_1}}^{y_{v_2}} (P-P_\infty)\ Z\ (y-\bar{y})\ dy + \int_{x_{v_1}}^{x_{v_2}} (P-P_\infty)\ Z\ (x-\bar{x})\ dx$$

$$+ \int_{y_{c_1}}^{y_{c_2}} (P-P_\infty)\ Z\ (y-\bar{y})\ dy - \int_{x_{v_1}}^{x_{v_2}} (P-P_\infty)\ Z\ (x-\bar{x})\ dx \tag{20}$$

where $(\bar{x},\bar{y})$ is the origin for the pitching moment.

To assess the sidewall contributions to the thrust and pitching moment, momentum conservation integrals must be applied at the entrance and exit planes of the nozzle since the sidewall pressures are not explicitly calculated in the analysis. Before calculating these integrals, the exiting and entering mass flux must be compared since slight discrepancies in mass flow can yield substantial errors in momentum integrals. The mass flux is given by

$$\psi = \int_y \frac{\rho\ v \sin(\theta_s - \theta)}{\sin\theta_s}\ Z\ dy \tag{21}$$

where θ_s represents the inclination of the plane of integration with respect to the horizontal (i.e., for the exit plane depicted in Figure 1, $\theta_s = \tan^{-1}((y_{v_2}-y_{c_2})/(x_{v_2}-x_{c_2}))$. To perform an integration over the exit plane where properties are only known at the discrete points E_0-E_5, the two dimensional value of wave strength is linearly distributed between E points and properties are obtained at these interpolated points following the procedures outlined in Section II. The properties are corrected to yield the correct existing mass flow as follows for frozen flow.

Let $R = \psi_2/\psi_1$ represent the ratio of exiting to entering mass flux as determined by the integration of Equation (21). For frozen flows, the mass flux can be expressed solely in terms of the Mach number distribution

$$\psi = \int_y \frac{D(M)\sin(\theta_s - \theta)}{\sin\theta_s}\ Z\ dy \tag{22}$$

where

$$D(M) = M\left(1 + \frac{\gamma-1}{2} M^2\right)^{-(\gamma+1)/2(\gamma-1)} \tag{23}$$

Then, at all points on the exit profile D(M) is corrected by division by R

$$D'(M) = D(M)/R$$

and the corrected Mach number is found by solving the transcendental equation (Equation 23) for M' by an iterative procedure; then, desired properties are obtained by standard isentropic expansions

$$P' = P\left(\frac{1 + \frac{\gamma-1}{2} M^2}{1 + \frac{\gamma-1}{2} M'^2}\right)^{\frac{\gamma}{\gamma-1}} \tag{24}$$

$$T' = T\left(\frac{1 + \frac{\gamma-1}{2} M^2}{1 + \frac{\gamma-1}{2} M'^2}\right) \tag{25}$$

For equilibrium calculations, at all points on the exit plane, the product (ρu) is divided by R

$$(\rho u)' = \frac{\rho u}{R}$$

and the value of $\Delta\nu'$ is sought that produces this change. Hence, an iterative procedure is employed wherein the value of $\Delta\nu$ at all points on the exit plane is perturbed to produce the corrected product $(\rho u)'$ employing Equations (7) - (12).

Then, having corrected the mass flow by modification of properties at the exit plane, the momentum flux is given by

$$F_x = \int_y \left[\frac{\rho V^2 \sin(\theta_s-\theta)\cos\theta + (P-P_\infty)\sin\theta_s}{\sin\theta_s}\right] z\,dy \tag{26}$$

which for frozen flow, simplifies to

$$F_x = \int_y \left[\frac{\gamma P M^2 \sin(\theta_s-\theta)\cos\theta + (P-P_\infty)\sin\theta_s}{\sin\theta_s}\right] z\,dy \tag{27}$$

Then, the total thrust is given by

$$Th = F_{x_2} - F_{x_1} \tag{28}$$

where 1 and 2 represent integrations over the entrance and exit stations respectively. The thrust attributed to the sidewalls is then given by

$$Th_{sw} = Th - Th^{*} \tag{29}$$

The pitching moment is given by

$$M = PM_2 - PM_1 \tag{30}$$

where

$$PM = \int_Y [f_x(y-\bar{y}) - f_y(x-\bar{x})]\, z dy \tag{31}$$

and

$$f_x = \frac{\rho V^2 \sin(\theta_s - \theta)\cos\theta + (P-P_\infty)\sin\theta_s}{\sin\theta_s} \tag{32}$$

$$f_y = \frac{\rho V^2 \sin(\theta_s - \theta)\sin\theta + (P-P_\infty)\cos\theta_s}{\sin\theta_s} \tag{33}$$

V. SAMPLE CALCULATIONS

To assess the validity of the developed program, a comparison was made with the source flow characteristic program described in Reference (3). The nozzle configuration and resultant wave field (as determined from the parametric design program (NOZD)) is depicted in Figure (9). The data at various points in the flow field is given in Table I. A comparison of the pressure distributions downstream of the cowl and undersurface expansion fans is depicted in Figure (10). The source flow characteristic program (NOZBOD) had the source located at x = - 20 and had the sharp turns at the cowl and undersurface occur over a distance Δx = .15. The source flow program calculates frozen chemistry via appropriate curve fits while the wave program calculates constant γ frozen flows. It should be noted that the program NOZD required .4 seconds for this calculation as compared with 55 seconds for NOZBOD, indicating that the ratio of running times for frozen flow calculations is of the order of 140 to 1. The pressure distribution as calculated by NOZD shows excellent agreement with the second order source flow calculations of NOZBOD and the thrust values as calculated by the two programs agreed to within one percent. Additionally, depicted in Figure (10) are the corresponding pressure distributions on the cowl and undersurface for the equivalent nozzle with no lateral expansion. The data at various points in the flow field for this calculation are given in Table II.

To illustrate the parametric design process, a two dimensional calculation was performed for a nozzle subject to the following constraints:

Entrance Height	1.0
Exit Height	6.0
Vehicle Length	15.0
Cowl Expansion	10^{0}

For these constraints, the cowl length was systematically varied from 3 to 9 and for each cowl length the vehicle undersurface expansion was varied in one degree increments as depicted in Figure (3).

The initial conditions for this case are:

$$P = 850 \text{ lb/ft}^2$$
$$M = 3$$
$$\gamma = 1.4$$
$$\Phi = 0$$

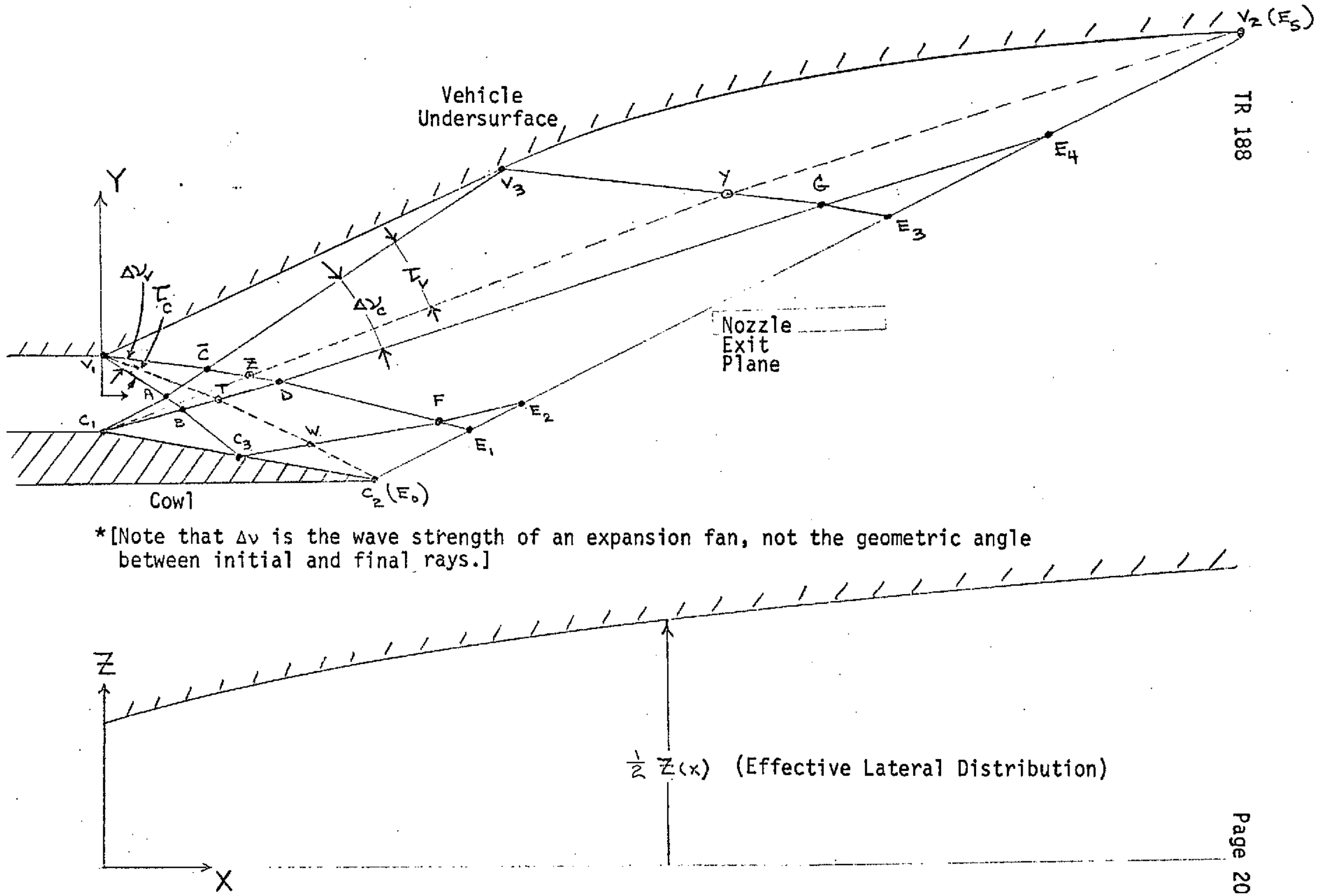
and

$$P_\infty = 23.1 \text{ lb/ft}^2$$

The thrust is plotted versus vehicle turning angle with the cowl length as a parameter in Figure (11), and the lift is plotted in Figure (12).

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 results obtained by this program compare favorably with more precise analytical methods with a considerable savings in overall running time. While the current program cannot analyze all possible configurations, it is felt that once a user gains familiarity with the program, changes to accommodate other configurations can readily be made.



*[Note that Δν is the wave strength of an expansion fan, not the geometric angle between initial and final rays.]

FIGURE 1. TYPICAL NOZZLE CONFIGURATION

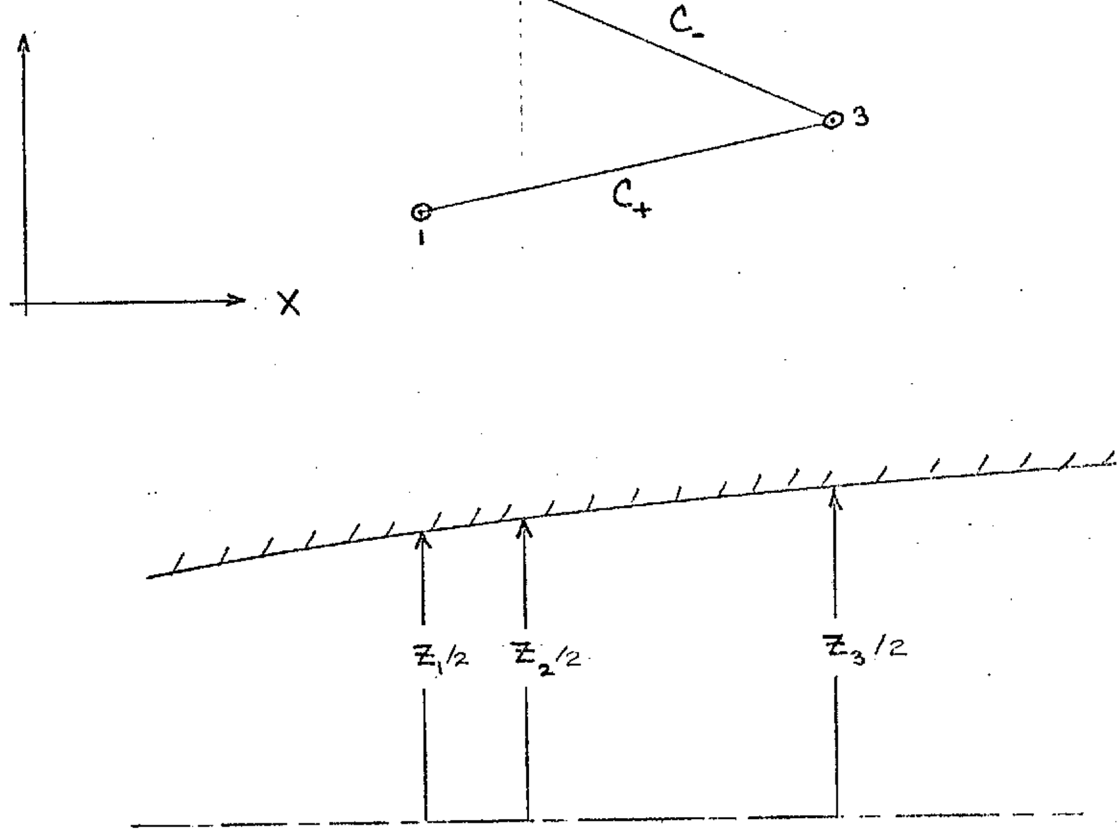


FIGURE 2. GRID POINT CALCULATION

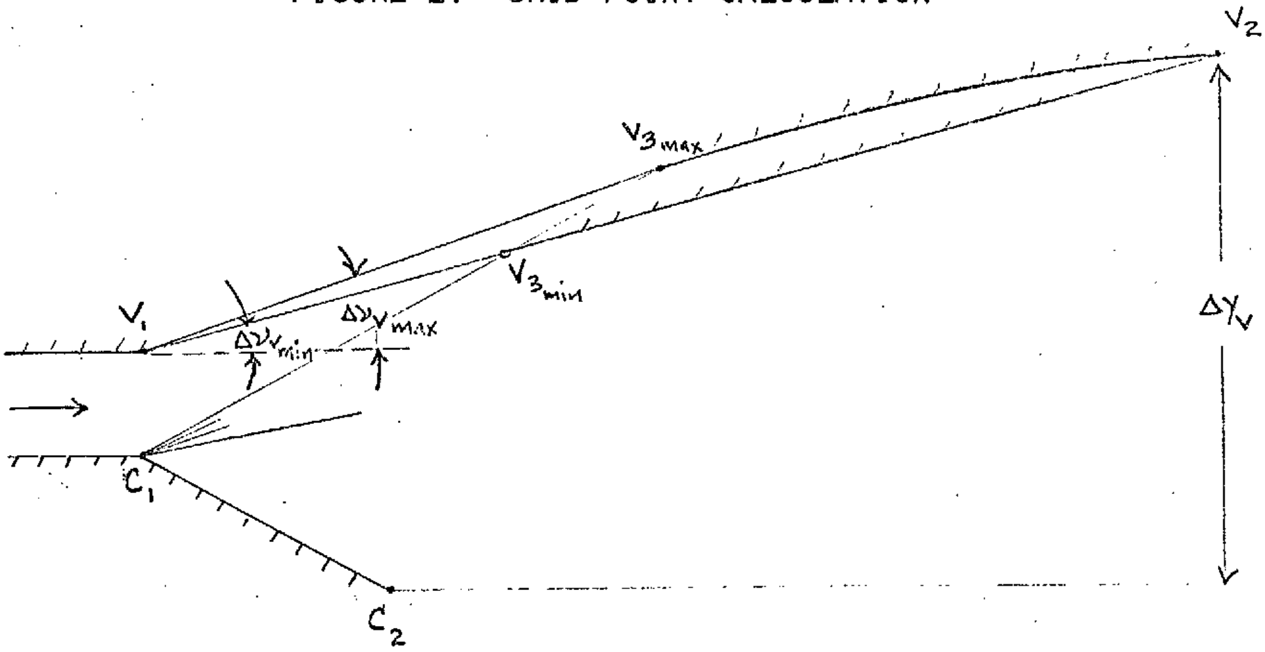


FIGURE 3. VARIATION OF VEHICLE EXPANSION WAVE STRENGTH

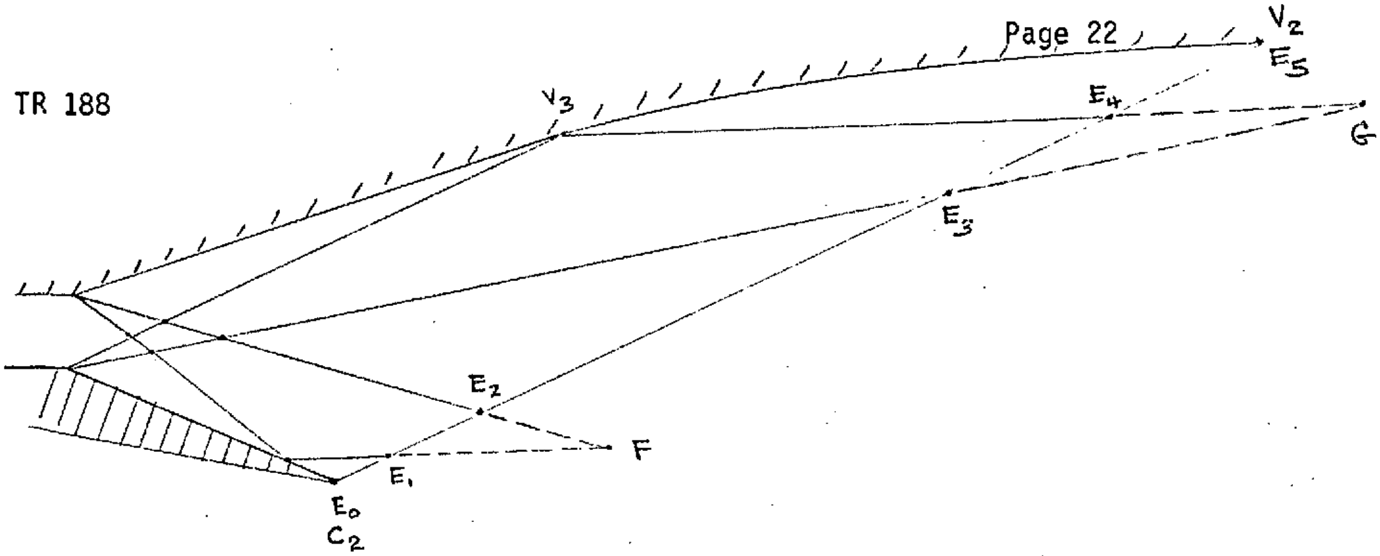


FIGURE 4. F AND G POINT DOWNSTREAM OF EXIT PLANE

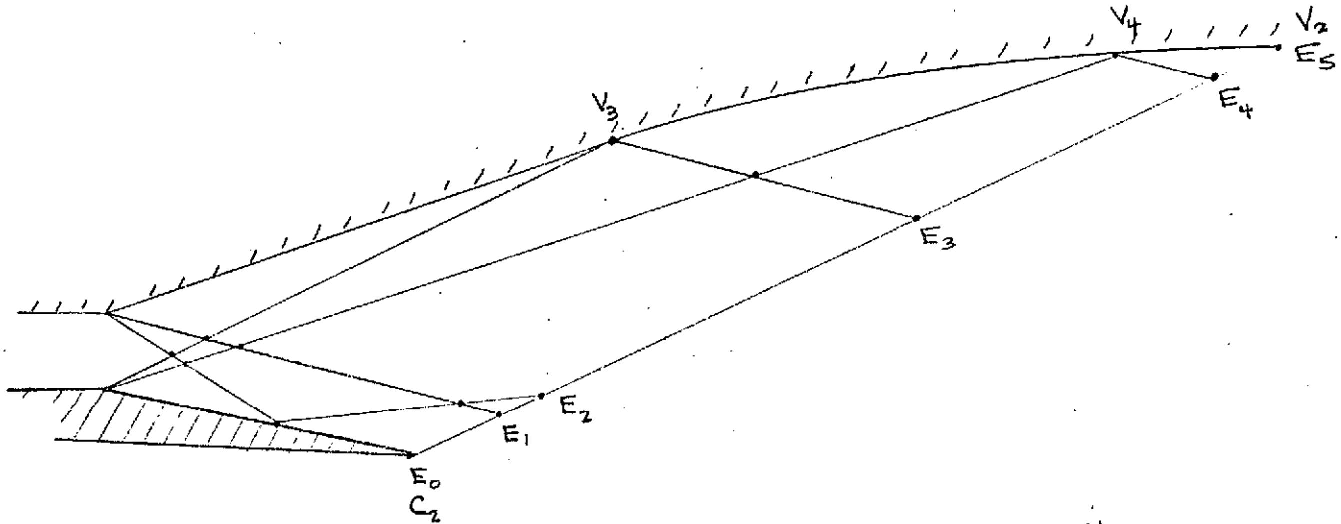


FIGURE 5. FULL COWL WAVE CAPTURED ON UNDERSURFACE

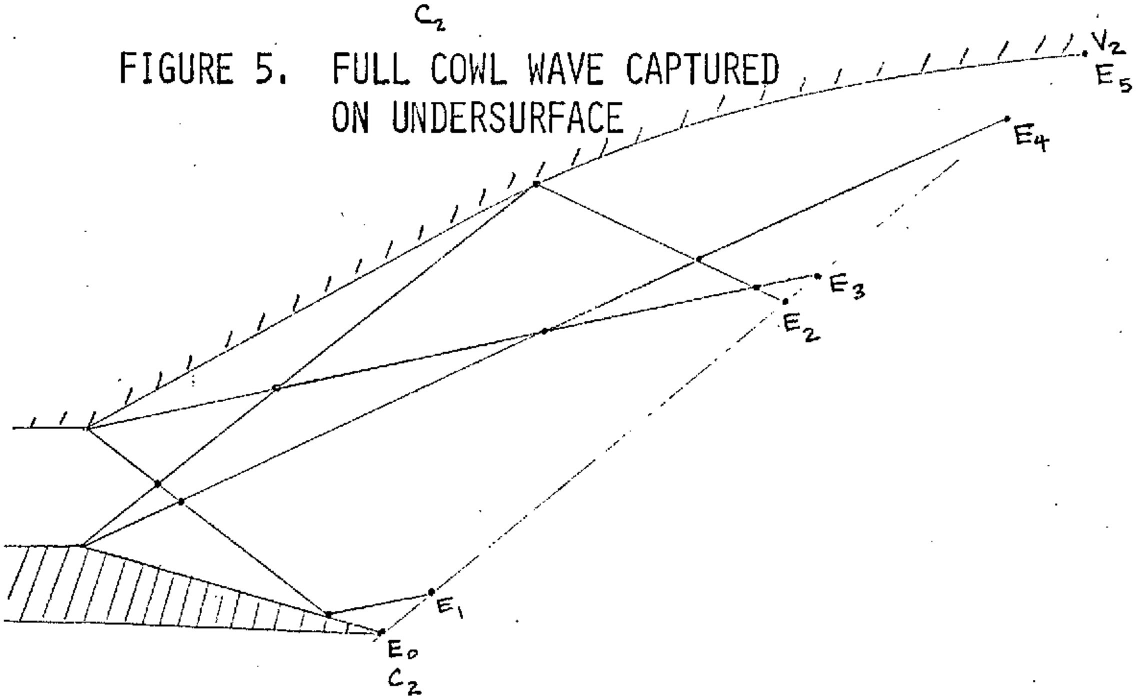


FIGURE 6. NO UNIFORM WAVE ZONE ON EXIT PLANE

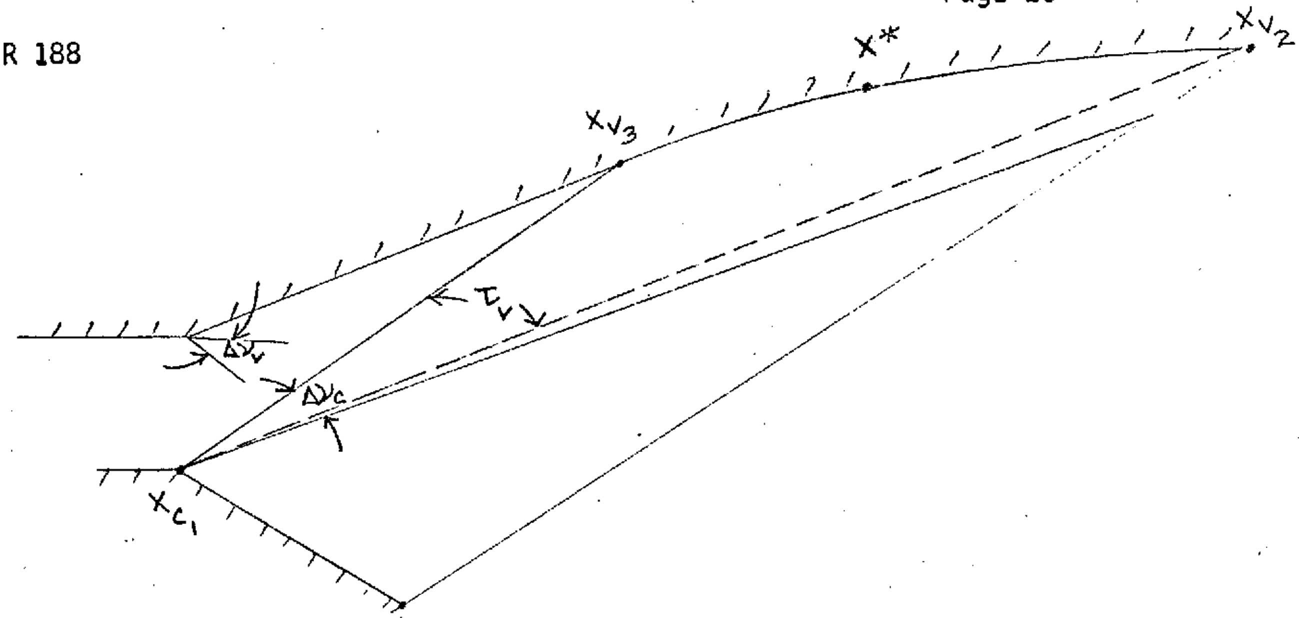


FIGURE 7. WAVE DISTRIBUTION ON UNDERSURFACE

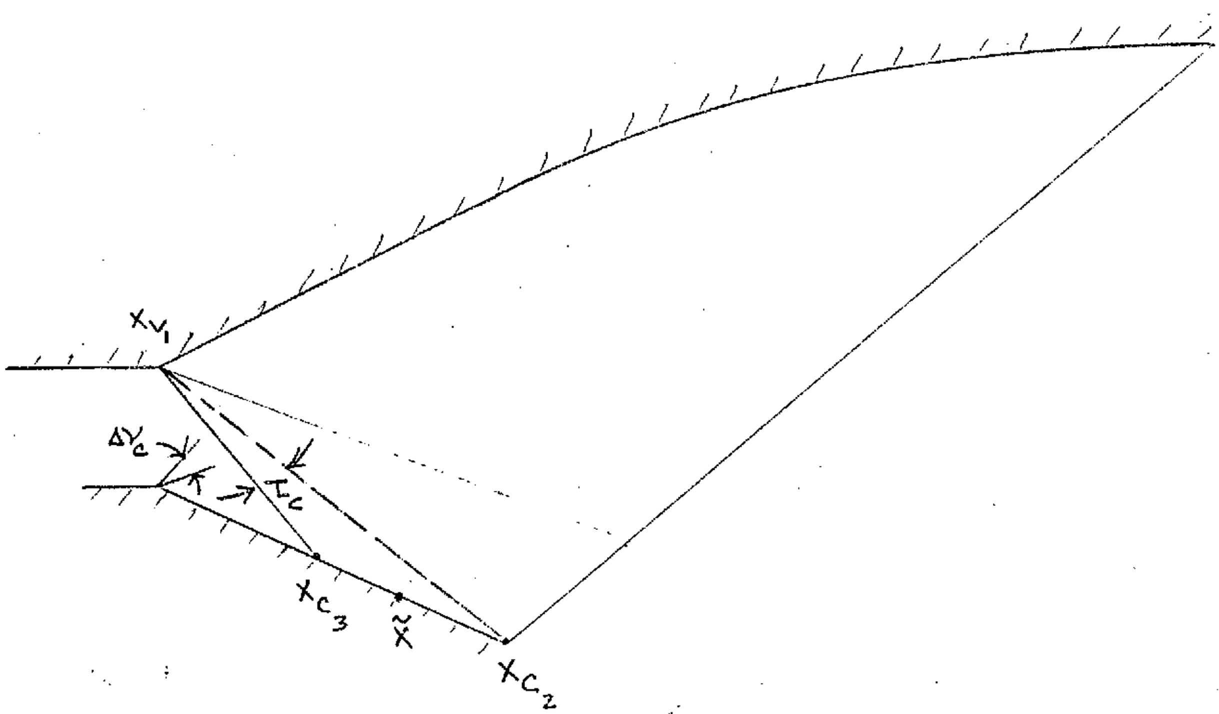


FIGURE 8. WAVE DISTRIBUTION ON COWL

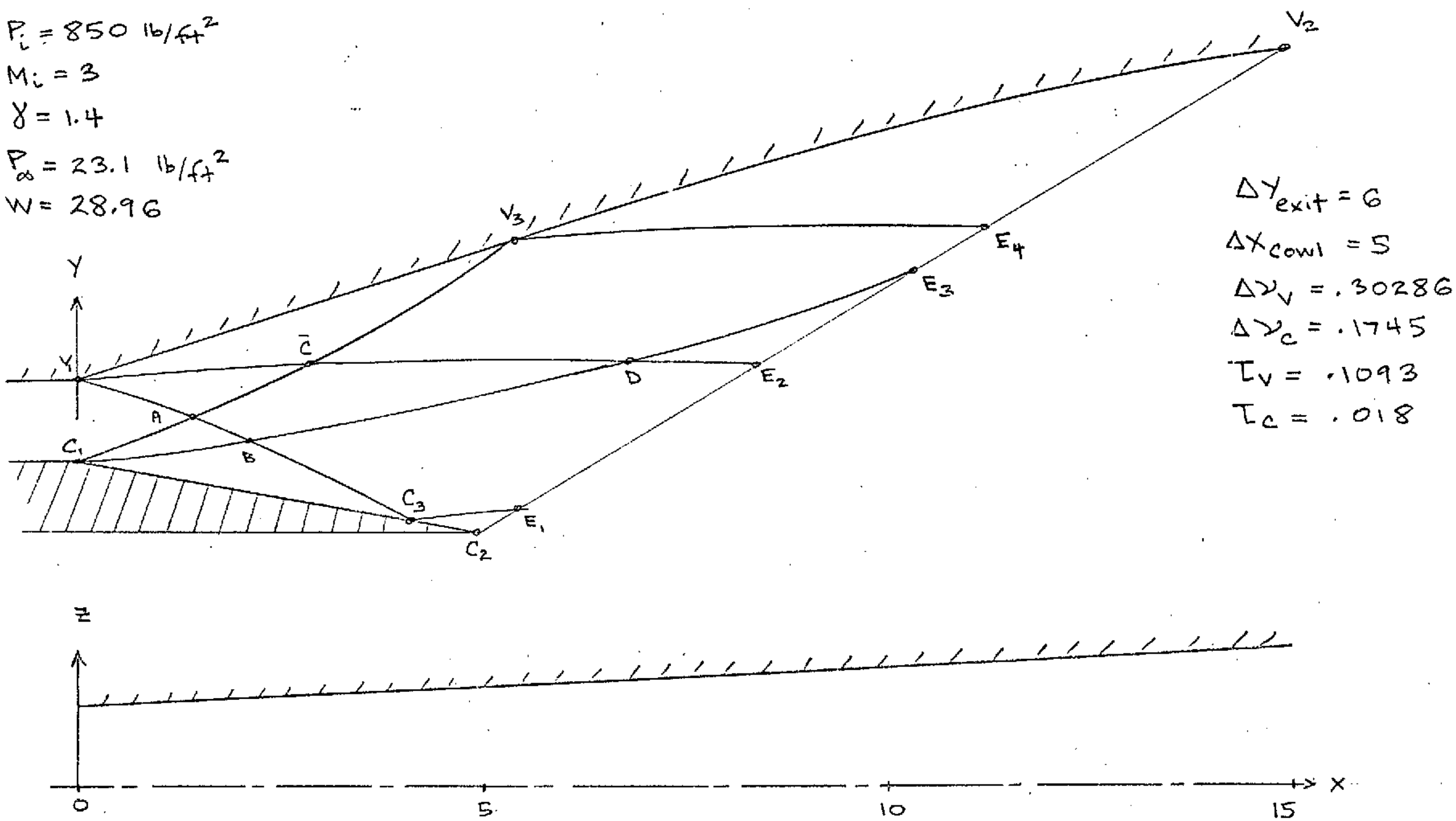


FIGURE 9. WAVE PATTERN IN THREE DIMENSIONAL COMPARISON CALCULATION

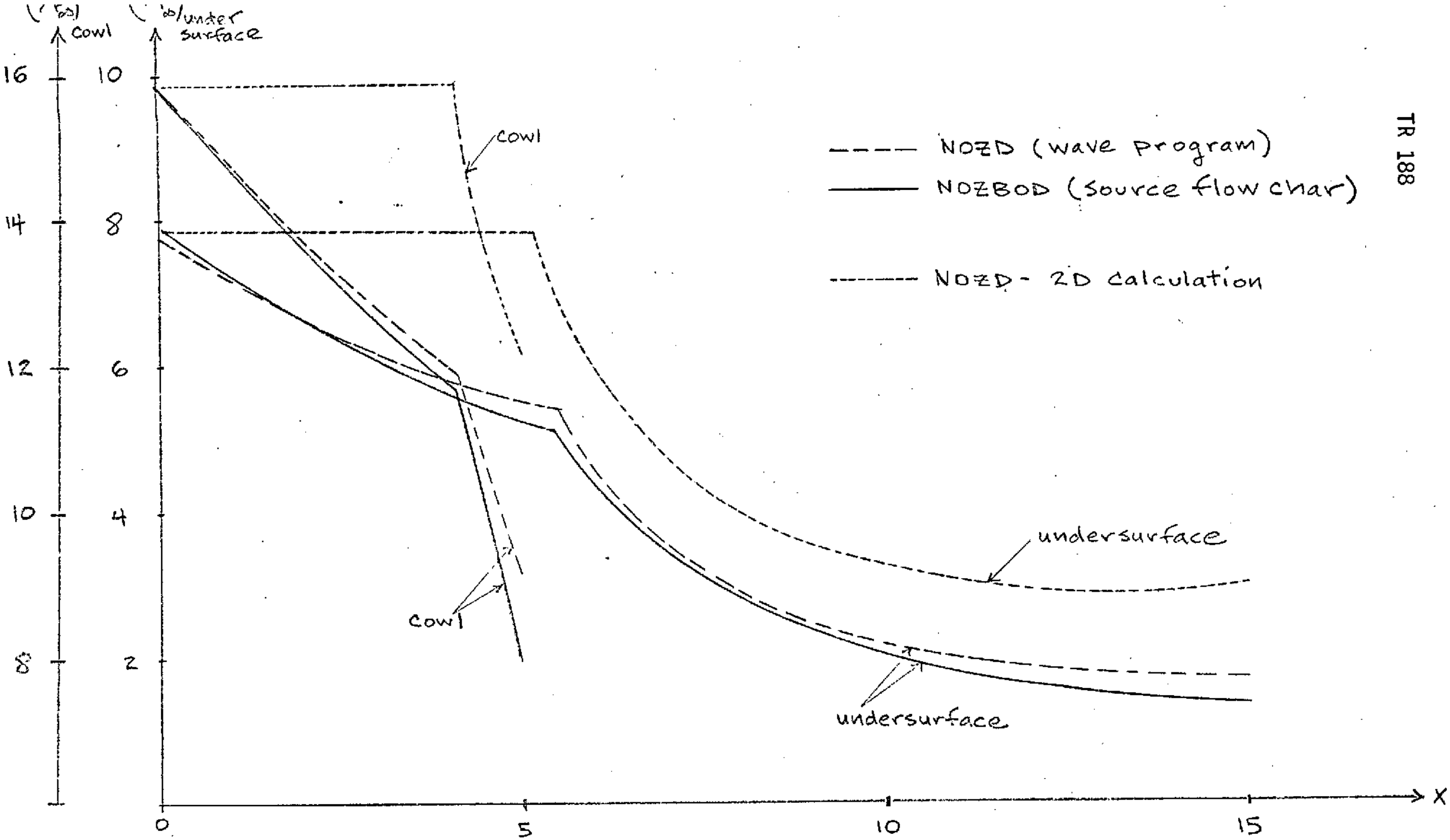


FIGURE 10. COMPARISON OF PRESSURE DISTRIBUTIONS WAVE PROGRAM VS. CHAR PROGRAM

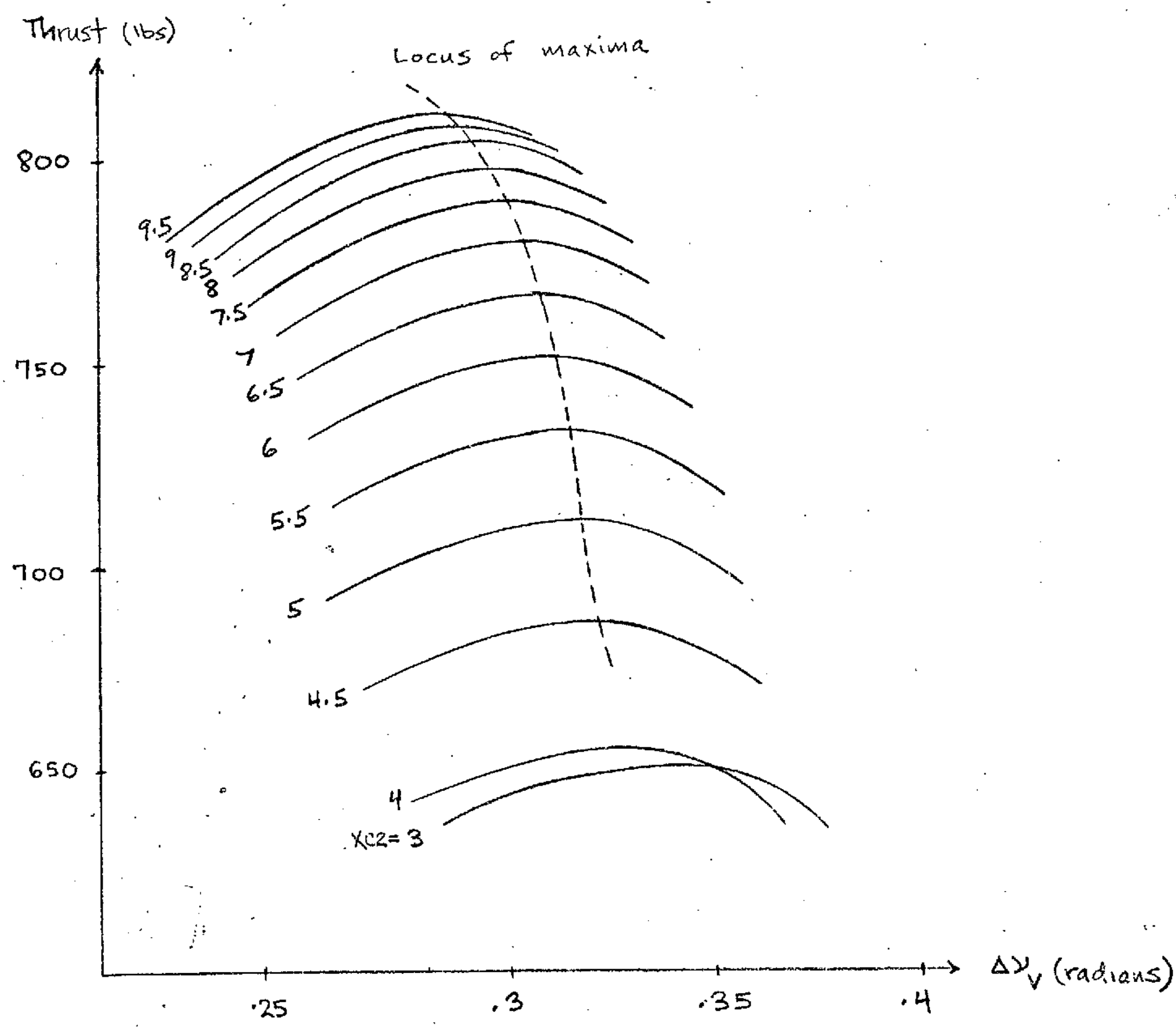


FIGURE 11. THRUST VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION ANGLE

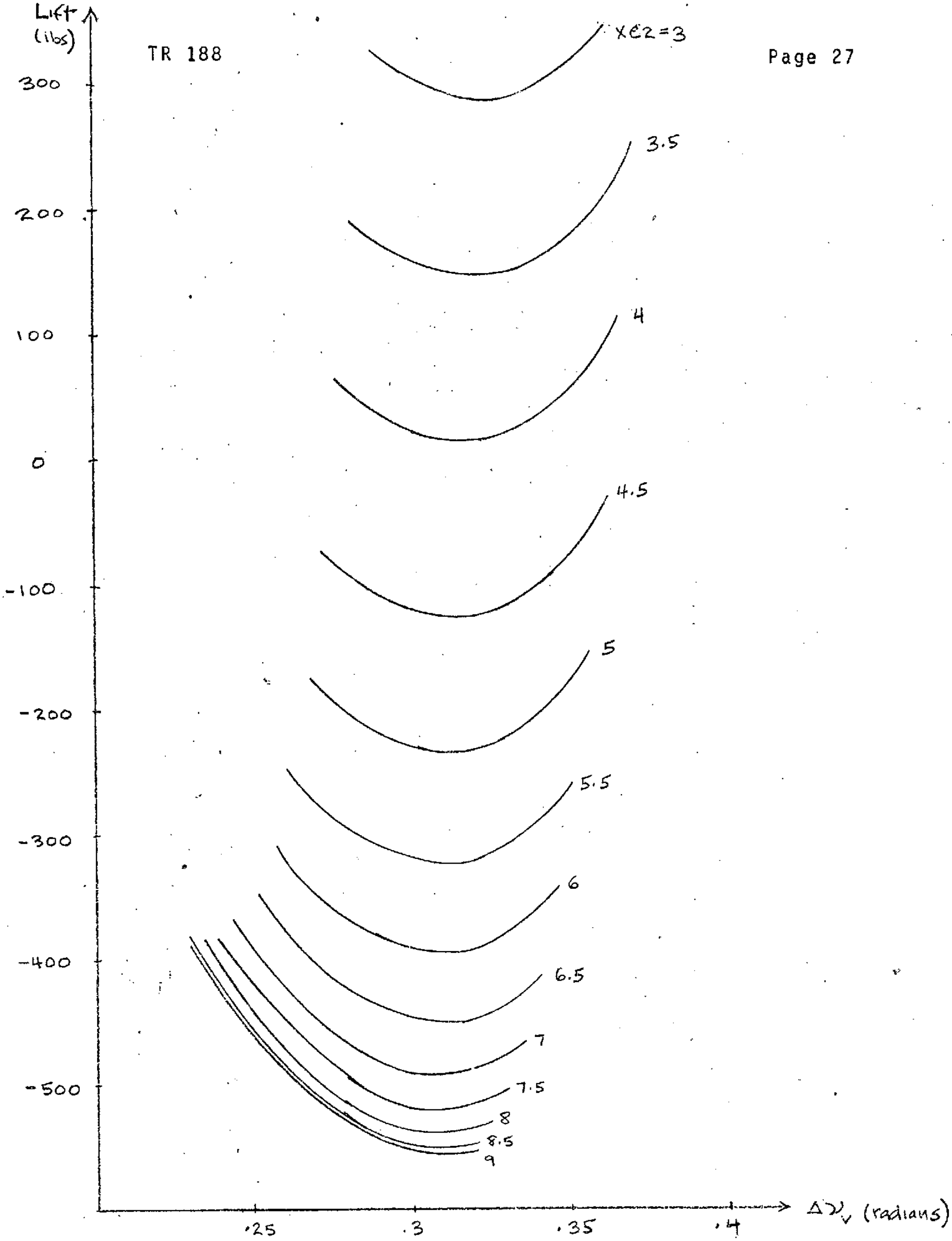


FIGURE 12. LIFT VARIATION WITH COWL LENGTH AND VEHICLE EXPANSION

XC2 = 5.0000E+00
DNUV = 3.0286E-01
DNUC = 1.7452E-01
TAU = 1.0927E-01
TAUC = 1.7788E-02
TAURV = -1.1235E-01
TAURC = 0.

PT.	X	Y	Z	THETA	MACH	DNU2D
V1	0.	1.0000E+00	1.0000E+00	3.0286E-01	4.1372E+00	3.0286E-01
C1	0.	0.	1.0000E+00	-1.7452E-01	3.6070E+00	1.7452E-01
A	1.4457E+00	5.0000E-01	1.0723E+00	0.	3.0971E+00	0.
B	2.1320E+00	2.1850E-01	1.1066E+00	-1.7452E-01	3.7171E+00	1.7452E-01
C	2.9043E+00	1.1839E+00	1.1452E+00	3.0286E-01	4.2930E+00	3.0286E-01
D	6.8416E+00	1.2088E+00	1.3421E+00	1.2834E-01	5.4797E+00	4.7738E-01
V3	5.4918E+00	2.7161E+00	1.2746E+00	3.0286E-01	4.4170E+00	3.0286E-01
C3	4.1220E+00	-7.2677E-01	1.2061E+00	-1.7452E-01	3.8115E+00	1.7452E-01

NO F POINT
NO G POINT

PT.	X	Y	Z	THETA	MACH	DNU2D	P	T
E0	5.0000E+00	-8.8157E-01	1.2500E+00	-1.7452E-01	3.9908E+00	2.1010E-01	2.1547E+02	6.7562E+02
E1	5.5189E+00	-5.7023E-01	1.2759E+00	-1.1295E-01	4.1213E+00	2.3609E-01	1.8130E+02	6.4310E+02
E2	8.3471E+00	1.1267E+00	1.4174E+00	1.2834E-01	5.5195E+00	4.7738E-01	3.4006E+01	3.9867E+02
E3	1.0361E+01	2.3349E+00	1.5180E+00	1.2834E-01	5.6112E+00	4.7738E-01	3.0791E+01	3.8751E+02
E4	1.1157E+01	2.8125E+00	1.5578E+00	1.3999E-01	5.5691E+00	4.6574E-01	3.2221E+01	3.9257E+02
E5	1.5000E+01	5.1184E+00	1.7500E+00	1.9051E-01	5.3691E+00	4.0906E-01	4.0124E+01	4.1796E+02

TABLE I DATA FOR THREE DIMENSIONAL SAMPLE CALCULATION

XC2 = 5.0000E+00
DNUV = 3.0286E-01
DNUC = 1.7452E-01
TAU = 1.2823E-01
TAUC = 2.3670E-02
TAURV = -1.0915E-01
TAURC = 0.

PT.	X	Y	Z	THETA	MACH	DNU2D
V1	0.	1.0000E+00	1.0000E+00	3.0286E-01	4.1372E+00	3.0286E-01
C1	0.	0.	1.0000E+00	-1.7452E-01	3.6070E+00	1.7452E-01
A	1.4264E+00	5.0000E-01	1.0000E+00	0.	3.0230E+00	0.
B	2.0862E+00	2.2290E-01	1.0000E+00	-1.7452E-01	3.6070E+00	1.7452E-01
C	2.8135E+00	1.1654E+00	1.0000E+00	3.0286E-01	4.1372E+00	3.0286E-01
D	6.2253E+00	1.1457E+00	1.0000E+00	1.2834E-01	5.0682E+00	4.7738E-01
V3	5.2206E+00	2.6313E+00	1.0000E+00	3.0286E-01	4.1372E+00	3.0286E-01
C3	3.9703E+00	-7.0002E-01	1.0000E+00	-1.7452E-01	3.6070E+00	1.7452E-01
		NO F POINT				
G	1.0516E+01	2.6006E+00	1.0000E+00	1.2834E-01	5.0682E+00	4.7738E-01

PT.	X	Y	Z	THETA	MACH	DNU2D	P	T
E0	5.0000E+00	-8.8157E-01	1.0000E+00	-1.7452E-01	3.7908E+00	2.2188E-01	2.8241E+02	7.2992E+02
E1	5.6980E+00	-4.6278E-01	1.0000E+00	-9.0052E-02	3.9441E+00	2.5899E-01	2.2938E+02	6.8781E+02
E2	8.1527E+00	1.0100E+00	1.0000E+00	1.2834E-01	5.0682E+00	4.7738E-01	5.6430E+01	4.6074E+02
E3	1.0773E+01	2.5825E+00	1.0000E+00	1.2834E-01	5.0682E+00	4.7738E-01	5.6430E+01	4.6074E+02
E4	1.1180E+01	2.8265E+00	1.0000E+00	1.3018E-01	5.0797E+00	4.7922E-01	5.5685E+01	4.5899E+02
E5	1.5000E+01	5.1184E+00	1.0000E+00	1.9371E-01	4.9023E+00	4.5016E-01	6.8504E+01	4.8698E+02

TABLE II DATA FOR TWO DIMENSIONAL SAMPLE CALCULATION

REFERENCES

1. Dash, S., "The Determination of Nozzle Contours for Rotational, Non-Homentropic Gas Mixtures," ATL TR 148, March 1970.

2. Ferri, A., Dash, S., and Del Guidice, P., "Methodology for Three Dimensional Nozzle Design," ATL TR 195, March 1974.

3. 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.

4. Dash, S. and Del Guidice, P., "Three Dimensional Nozzle-Exhaust Flow Field Analysis by a Reference Plane Technique," AIAA Paper No. 72-704, June 1972.

5. Dash, S. and Del Guidice, P., "Analysis and Design of Three Dimensional Supersonic Nozzles," ATL TR 166, October 1972.

APPENDIX I

PROGRAM DESCRIPTION

A. Input

Card 1 (Format 8E10.0)

Column 1-10 P (initial pressure, lb/ft^2)

11-20 Th (initial flow deflection angle, radians)

21-30 EM (initial Mach number - frozen for frozen deck, equilibrium for equilibrium deck)

31-40 T (initial temperature, degrees Rankine)

* For Frozen Deck

41-50 G (specific heat ratio γ - frozen)

51-60 WGAS (molecular weight)

61-70 PINF (external pressure P_∞, lb/ft^2)

* For Equilibrium Deck

41-50 PHI (fuel-air equivalence ratio Φ)

51-60 PINF (external pressure P_∞, lb/ft^2)

Card 2 (Format 8E10.0)

Column 1-10 XV1 (X_{V_1} of Figure 1, ft)

11-20 YV1 (Y_{V_1})

21-30 XV2 (X_{V_2})

31-40 XC1 (X_{C_1})

41-50 YC1 (Y_{C_1})

51-60 XC2 (X_{C_2})

61-70 DYV (vehicle exit height $Y_{V_2}-Y_{C_2}$)

71-80 DNUC ($\Delta\nu_C$-cowl turning angle-degrees)

Card 3

Column 1-5	(I5)	ICF-number of different cowl lengths to be executed
6-15	(E10.0)	DXC-increment to be added to original cowl length X_{C_2}
16-20	(I5)	JTF-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	(I5)	IPR-print option IPR=0 prints properties on undersurface and cowl IPR=1 suppresses above printout
36-40	(I5)	IPOLY-polynomial print option IPOLY=0 prints polynomial coefficients of wall surfaces in form suitable for running program NOZBOD (Reference 3) IPOLY=1 suppresses above printout

Card 4 (8E10.0)

Column 1-10	XSHF (X location of origin for moment calculation)
11-20	YSHF (Y location of moment origin)

Card 5 (8E10.0)

Column 1-10	AZ	coefficients of lateral expansion quadratic
11-20	BZ	$Z(X)=AZ*X^2+BZ*X+CZ$
21-30	CZ	(where Z(X) is ratio of lateral nozzle extent at X station to that at initial station)

C

B. Output

The output is readily understood by referring to the figures in the report. If the F point or G point falls downstream of the nozzle exit station E_0-E_5, this is indicated in the printout by "No F Point" or "No G Point" respectively. If there is no uniform two dimensional wave zone, the exit line is not computed. The percentage thrust loss is based on the calculation of the ideal thrust using one dimensional considerations based on the entrance and exit area projections onto planes X=constant (i.e., A_{exit}=(YV2-YC2)*(ZV2+ZC2)/2).

3. RHEQ-density fit in equilibrium deck ρ=RHEQ(P,T,Φ)

4. FGAM-isentropic fit in equilibrium deck Γ=FGAM(P,T,Φ)

5. FH-static enthalpy fit in equilibrium deck h-FH(P,T,Φ)

APPENDIX II

LISTING OF FROZEN FLOW PROGRAM

```
      PROGRAM NOZD(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      COMMON/A/P1(3),TH1(3),EM1(3),A1(3),G1(3)
      COMMON/SHF/ XSHF,YSHF
      COMMON/POL/ IPR,IPOLY
      COMMON/THRMAX/THRMAX
      DIMENSION T1(3)
      COMMON/B/XA,YA,XB,YB,XC,YC,XD,YD,
     1THA,THB,THC,THD,EMA,EMB,EMC,EMD,XMUA,XMUB,XMUC,XMUD
      COMMON/C/DNUV,DNUC,XC1,YC1,XV1,YV1,DNUL
      COMMON/D3/ AZ,BZ,CZ
      COMMON /V/ AV,BV,CV
      COMMON/P/ PINF
      DIMENSION P2(10),TH2(10),EM2(10),T2(10),G2(10),X2(10),Y2(10)
     1,Z2(10)
      DIMENSION HOL(11),HOLE(6),DNUE(10)
      DATA HOLE/2HE0,2HE1,2HE2,2HE3,2HE4,2HE5/
      DATA HOL/2HV1,2HC1,2HA ,2HB ,2HC ,2HD ,2HV3,2HC3,2HF ,2HG ,2HV4/
      GETZ(X5)=AZ*X5*X5+BZ*X5+CZ
   98 FORMAT(7E13.5)
   99 FORMAT(/)
  100 FORMAT(8E10.0)
      READ(5,100) P1(1),TH1(1),EM1(1),T1(1),G1(1),WGAS,PINF
      DO 106 I=2,3
      P1(I)=P1(1) $ TH1(I)=TH1(1) $ EM1(I)=EM1(1) $ G1(I)=G1(1)
      T1(I)=T1(1)
  106 CONTINUE
      DUM=1.+(G1(1)-1.)/2.*EM1(1)**2
      PTOT=P1(1)*DUM**(G1(1)/(G1(1)-1.))
      TTOT=T1(1)*DUM
      RG=49800.
      RGAS=RG/WGAS
      TAURC=0.
      READ(5,100) XV1,YV1,XV2,XC1,YC1,XC2,DYV,DNUC
      READ(5,5921) ICF,DXC,JTF,DTH,IPR,IPOLY
 5921 FORMAT(I5,E10.0,I5,E10.0,2I5)
      READ(5,100) XSHF,YSHF
      DNUC=DNUC/57.3
      READ(5,100) AZ,BZ,CZ
      ZC1=AZ*XC1*XC1+BZ*XC1+CZ
      ZV1=AZ*XV1*XV1+BZ*XV1+CZ
      ZC2=GETZ(XC2)
      WRITE(6,5922)
 5922 FORMAT(1H1/////)
      WRITE(6,1020)
 1020 FORMAT(     20X*F R O Z E N   F L O W   N O Z Z L E   D E S I G N*)
      WRITE(6,400)  P1(1),TH1(1),EM1(1),T1(1),G1(1),WGAS,PINF
  400 FORMAT(     ///30X,*INITIAL PROFILE*//9X*PRESSURE*7X*THETA*8X*MACH*
     16X*TEMPERATURE*5X*GAMMA*7X*MOL WT*8X*PINF*/7E13.5)
      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///)
      WRITE(6,5923) XSHF,YSHF
 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)
```

```
      XC22=XC2
      THC1=TH1(3)-DNUC
      DO 5000 IXC=1,ICF
      XC2=XC22+FLOAT(IXC-1)*DXC
      CALL THM(EM1(1),P1(1),G1(1),PINF,YV1,YC1,DYV,XV2,THRMAX,XC2)
      WRITE(6,6364) THRMAX
 6364 FORMAT(1H131X*IDEAL THRUST =*E12.4)
      YC2=YC1+TAN(THC1)*(XC2-XC1)
      YV2=YC2+DYV
      DO 6000 JT=1,JTF
      THX=1.E+10
      THJT=FLOAT(JT-1)*DTH
      DNUV=ATAN((YV2-YV1)/(XV2-XV1))-TH1(3)+THJT/57.3
      CALL PM(EM1(1),DNUC,EMC1,G1(1))
      CALL PM(EM1(1),DNUV,EMV1,G1(1))
      IT2=1
 1000 CONTINUE
      CALL CHAR
      DNUA=0.
      DNUD=DNUV+DNUC
      ZA=GETZ(XA)
      ZB=GETZ(XB)
      ZC=GETZ(XC)
      ZD=GETZ(XD)
      RAT=0.
      THC3=THC1
      CALL PM(EM1(1),DNUC,EMC3,G1(1))
      CALL FIX(XC1,YC1,THC1,0.,XB,YB,THB,XMUB,XC3,YC3,THC3,XMUC3,EMC3,
     1G1(1))
      ZC3=GETZ(XC3)
      IF(XC3.LT.XC2) GO TO 177
      TAUC=0.
      XC3=XC2
      DNUE0=DNUC
      THC2=THC1
      THE0=THC2
      CALL PM(EM1(1),DNUE0,EME0,G1(1))
      CALL EM3D(EME0,EME0N,XC2,G1(1))
      EME0=EME0N
      XMUE0=ASIN(1./EME0)
      GO TO 175
  177 CONTINUE
      IT9=1
      RAT=.1
  176 CONTINUE
      XMUBD=XMUB+RAT*(XMUD-XMUB)
      THBD=THB+RAT*(THD-THB)
      XT=XB+RAT*(XD-XB)
      YT=YB+RAT*(YD-YB)
      TAUC=RAT*DNUV
      IF(RAT.GT.1.0) GO TO 6000
      DNUW=DNUC+TAUC
      CALL PM(EM1(1),DNUW,EMW,G1(1))
      THW=THC3+TAUC
      CALL FIX(XC3,YC3,THC3,XMUC3,XT,YT,THBD,XMUBD,XW,YW,THW,XMUW,EMW,
     1G1(1))
```

```
      DNUE0=DNUC+2.*TAUC
      THC2=THC1
      THE0=THC2
      CALL PM(EM1(1),DNUE0,EME0,G1(1))
      CALL EM3D(EME0,EME0N,XC2,G1(1))
      EME0=EME0N
      XMUE0=ASIN(1./EME0)
      SLWC2=(TAN(THW-XMUW)+TAN(THE0-XMUE0))/2.
      YC2T=YW+SLWC2*(XC2-XW)
      ER9=YC2-YC2T
      IF(ABS(ER9).LT.1.E-03) GO TO 175
      CALL ERROR(4,IT9,RAT,ER9,1.11,RAT1,ER91)
      GO TO 176
  175 CONTINUE
      THV3=TH1(1)+DNUV
      CALL PM(EM1(1),DNUV,EMV3,G1(1))
      THV1=THV3
      CALL FIX(XC,YC,THC,XMUC,XV1,YV1,THV1,0.,XV3,YV3,THV3,XMUV3,EMV3,
     1G1(1))
      AV=YV3 $ BV=TAN(THV3)
      CV=(YV2-AV-BV*(XV2-XV3))/(XV2-XV3)**2
      ZV3=GETZ(XV3)
      THV2=ATAN(BV+2.*CV*(XV2-XV3))
      IF(THV2.LT.0.) GO TO 462
      TAURV=THV2-THV3
      ITZ=1
      ITAU=0
      RAT=1.
  545 CONTINUE
  530 CONTINUE
      XZ=XC+RAT*(XD-XC)
      YZ=YC+RAT*(YD-YC)
      TAU=RAT*DNUC
      THZ=THC+RAT*(THD-THC)
      EMZ=EMC+RAT*(EMD-EMC)
      XMUZ=ASIN(1./EMZ)
      THY=THV3-TAU
      DNUY=DNUV+TAU
      CALL PM(EM1(1),DNUY,EMY,G1(1))
      CALL FIX(XZ,YZ,THZ,XMUZ,XV3,YV3,THV3,XMUV3,XY,YY,THY,XMUY,EMY,G1(1
     1))
      DNUE5=DNUV+2.*TAU+TAURV
      THE5=THV2
      CALL PM(EM1(1),DNUE5,EME5,G1(1))
      CALL EM3D(EME5,EME5N,XV2,G1(1))
      EME5=EME5N
      XMUE5=ASIN(1./EME5)
      SL2T=(TAN(THY+XMUY)+TAN(THE5+XMUE5))/2.
      YV2T=YY+SL2T*(XV2-XY)
      IF(ITZ.EQ.1.AND.YV2T.GE.YV2) GO TO 542
      EZ=YV2T-YV2
      IF(ABS(EZ).LT.1.E-03)GO TO 540
      IF(DNUC.EQ.0.) GO TO 540
      CALL ERROR(5,ITZ,RAT,EZ,.1,RAT1,EZ1)
      GO TO 530
  542 ITAU=1
```

```
540 CONTINUE
    DNUE2=DNUV+DNUC
    CALL PM(EM1(1),DNUE2,EME2,G1(1))
    THE2=TH1(1)+DNUV-DNUC
    THF=THE2
    EMF=EME2
    CALL FIX(XC3,YC3,THC3,XMUC3,XD,YD,THD,XMUD,XF,YF,THF,XMUF,EMF,
   1G1(1))
    SLE=(YV2-YC2)/(XV2-XC2)
    THE=ATAN(SLE)
    EMFT=EME2
    CALL FIX(XC3,YC3,THC3,XMUC3,XC2,YC2,THE,0.,XFT,YFT,THF,XMUFT,EMFT,
   1G1(1))
    IF(XF.GT.XFT) GO TO 300
    CALL FIX(XF,YF,THF,XMUF,XC2,YC2,THE,0.,XE2,YE2,THE2,XMUE2,EME2,
   1G1(1))
    ZF=GETZ(XF)
    SLF=TAN(THF-XMUF)
    CALL GEM(XF,YF,SLF,XC2,YC2,SLE,XE1,YE1)
    ITE1=0
368 CONTINUE
    RAT=(YE1-YC2)/(YE2-YC2)
    DNUE1=DNUV+DNUC+(1.-RAT)*TAUC
    CALL PM(EM1(1),DNUE1,EME1,G1(1))
    THE1=TH1(1)+DNUV-DNUC-(1.-RAT)*TAUC
    CALL FIX(XC2,YC2,THE,0.,XF,YF,THF,XMUF,XE1,YE1,THE1,XMUE1,EME1,
   1G1(1))
    IF(ITE1.EQ.1) GO TO 369
    ITE1=1
    GO TO 368
369 CONTINUE
    GO TO 310
300 CONTINUE
    CALL FIX(XC2,YC2,THE,0.,XD,YD,THD,XMUD,XE2,YE2,THE2,XMUE2,EME2,
   1G1(1))
    SLCE=TAN(THC3+XMUC3)
    CALL GEM(XC3,YC3,SLCE,XC2,YC2,SLE,XE1,YE1)
    ITE1=0
381 CONTINUE
    RAT=(YE1-YC2)/(YE2-YC2)
    DNUE1=DNUC+TAUC+RAT*(DNUV-TAUC)
    CALL PM(EM1(1),DNUE1,EME1,G1(1))
    THE1=TH1(1)-DNUC+TAUC+RAT*(DNUV-TAUC)
    CALL FIX(XC3,YC3,THC3,XMUC3,XC2,YC2,THE,0.,XE1,YE1,THE1,XMUE1,EME1
   1,G1(1))
    IF(ITE1.EQ.1) GO TO 380
    ITE1=1
    GO TO 381
380 CONTINUE
310 CONTINUE
    THE3=THE2
    DNUE3=DNUE2
    CALL PM(EM1(1),DNUE3,EME3,G1(1))
    XMUE3=ASIN(1./EME3)
    IF(ITAU.NE.1) GO TO 311
    XG=XY $ YG=YY $ THG=THY $ EMG=EMY $ XMUG=XMUY
```

```
      XGT=XG
      GO TO 3588
  311 CONTINUE
      THG=THE3 $ EMG=EME3
      CALL FIX(XD,YD,THD,XMUD,XV3,YV3,THV3,XMUV3,XG,YG,THG,XMUG,EMG,
     1G1(1))
      EMGT=EME3
      CALL FIX(XC2,YC2,THE,0.,XV3,YV3,THV3,XMUV3,XGT,YGT,THG,XMUGT,EMGT,
     1G1(1))
      IF(XG.GT.XGT) GO TO 350
 3588 CONTINUE
      CALL FIX(XC2,YC2,THE,0.,XG,YG,THG,XMUG,XE3,YE3,THE3,XMUE3,EME3,
     1G1(1))
      ZG=GETZ(XG)
      IF(YE3.LE.YE2) GO TO 359
      SLE4=TAN(THG+XMUG)
      CALL GEM(XG,YG,SLE4,XC2,YC2,SLE,XE4,YE4)
      IT4=0
  358 CONTINUE
      RAT=(YE4-YV2)/(YE3-YV2)
      DNUE4=DNUV+DNUC+(1.-RAT)*(TAURV+TAU)
      CALL PM(FM1(1),DNUE4,EME4,G1(1))
      THE4=TH1(1)+DNUV-DNUC+(1.-RAT)*(TAURV+TAU)
      CALL FIX(XG,YG,THG,XMUG,XC2,YC2,THE ,0.,XE4,YE4,THE4,XMUE4,EME4,
     1G1(1))
      IF(IT4.EQ.1) GO TO 359
      IT4=1
      GO TO 358
  359 CONTINUE
      IF(ITAU.EQ.0) GO TO 360
      SLV32=(YV2-YV3)/(XV2-XV3)
      SLG=TAN(THG+XMUG)
      CALL GEM(XV3,YV3,SLV32,XG,YG,SLG,XV4,YV4)
      ITV4=1
  362 YV4=AV+BV*(XV4-XV3)+CV*(XV4-XV3)**2
      THV4=ATAN(BV+2.*CV*(XV4-XV3))
      TAURV4=THV4-THV3
      DNUV4=DNUV+5.*TAU+TAURV4
      CALL PM(FM1(1),DNUV4,EMV4,G1(1))
      CALL FM3D(EMV4,EMV4N,XV4,G1(1))
      EMV4=EMV4N
      XMUV4=ASIN(1./EMV4)
      YV4T=YG+.5*(TAN(THG+XMUG)+TAN(THV4+XMUV4))*(XV4-XG)
      ERV4=(YV4T-YV4)/YV2
      IF(ABS(ERV4).LT.1.E-03) GO TO 361
      CALL ERROR(361,ITV4,XV4,ERV4,1,1,XV41,ERV41)
      ZV4=GETZ(XV4)
      IF(ITV4.EQ.1) GO TO 361
      GO TO 362
  361 DNUE4=DNUV4
      IF(YE3.LE.YE2) GO TO 460
      CALL PM(FM1(1),DNUE4,EME4,G1(1))
      THE4=THV4
      CALL FIX(XC2,YC2,THE,0.,XV4,YV4,THV4,XMUV4,XE4,YE4,THE4,XMUE4,
     1EME4,G1(1))
      GO TO 360
```

```
  350 CONTINUE
      IT4=0
      CALL FIX(XD,YD,THD,XMUD,XC2,YC2,THE,0.,XE3,YE3,THE3,XMUE3,EME3,
     1G1(1))
      IF(YE3.LE.YE2) GO TO 460
      SLE4=TAN(THV3-XMUV3)
      CALL GEM(XV3,YV3,SLE4,XC2,YC2,SLE,XE4,YE4)
  371 CONTINUE
      RAT=(YE4-YV2)/(YE3-YV2)
      DNUE4=DNUV+TAU+RAT*(DNUC-TAU)
      CALL PM(EM1(1),DNUE4,EME4,G1(1))
      THE4=TH1(1)+DNUV-TAU-RAT*(DNUC-TAU)
      CALL FIX(XC2,YC2,THE,0.,XV3,YV3,THV3,XMUV3,XE4,YE4,THE4,XMUE4,EME4
     1,G1(1))
      IF(IT4.EQ.1)GO TO 370
      IT4=1
      GO TO 371
  370 CONTINUE
  360 CONTINUE
      IF(YE3.LE.YE2) GO TO 460
      X2(1)=XC2 $ X2(2)=XE1 $ X2(3)=XE2 $ X2(4)=XE3 $ X2(5)=XE4
      X2(6)=XV2
      Y2(1)=YC2 $ Y2(2)=YE1 $ Y2(3)=YE2 $ Y2(4)=YE3 $ Y2(5)=YE4
      Y2(6)=YV2 $ EM2(1)=EME0 $ EM2(2)=EME1 $ EM2(3)=EME2 $ EM2(4)=EME3
      EM2(5)=EME4 $ EM2(6)=EME5 $ TH2(1)=THE0 $ TH2(2)=THE1
      TH2(3)=THE2 $ TH2(4)=THE3 $ TH2(5)=THE4 $ TH2(6)=THE5
      DNUE(1)=DNUE0 $ DNUE(2)=DNUE1 $ DNUE(3)=DNUE2 $ DNUE(4)=DNUE3
      DNUE(5)=DNUE4 $ DNUE(6)=DNUE5
      INDEX=0
  180 CONTINUE
      DO 160 I=1,6
      DUM=1.+(G1(1)-1.)/2.*EM2(I)**2
      P2(I)=PTOT/DUM**(G1(1)/(G1(1)-1.))
      T2(I)=TTOT/DUM
      Z2(I)=AZ*X2(I)**2+BZ*X2(I)+CZ
  160 CONTINUE
      THS1=1.5707963
      IF(XV1.NE.XC1) THS1=ATAN((YV1-YC1)/(XV1-XC1))
      THS2=ATAN((YV2-YC2)/(XV2-XC2))
      ZI=.5*(GETZ(XC1)+GETZ(XV1))
      XMASS1=P1(1)*EM1(1)/SQRT(T1(1))*SIN(THS1-TH1(1))/SIN(THS1)*ZI*
     1(YV1-YC1)
      DUM=G1(1)*EM1(1)**2*SIN(THS1-TH1(1))/SIN(THS1)
      U=1.-PINF/P1(1)
      THX1=P1(1)*(U+DUM*COS(TH1(1)))*ZI*(YV1-YC1)
      CALL XTH(X2,Y2,TH2,DNUE,EM1,G1,PTOT,TTOT,THS2,THX1,XMASS1,THX)
      ISH=0
      GO TO 461
  460 CONTINUE
      ISH=1
      WRITE(6,991)
  991 FORMAT(* YE3.LE.YE2*)
  461 CONTINUE
      WRITE(6,1004) XC2,DNUV,DNUC,TAU,      TAUC,TAURV,TAURC
 1004 FORMAT(      32X*XC2    =*E12.4/32X*DNUV   =*E12.4/32X*DNUC   =*E12.4/
     132X*TAU    =*E12.4/ 32X*TAUC   =*E12.4/32X*TAURV =*E12.4/32X*TAURC =
```

```
     1*E12.4///)
      WRITE(6,1003)
 1003 FORMAT(1X*PT.*6X*X*11X*Y*11X*Z* 8X*THETA*7X*MACH*8X*DNU2D*/)
      WRITE(6,1002)HOL(1),XV1,YV1,ZV1,THV1,EMV1,DNUV,HOL(2),XC1,YC1,
     1ZC1,THC1,EMC1,DNUC,HOL(3),XA,YA,ZA,THA,EMA,DNUA,HOL(4),XB,YB,ZB,
     1THB,EMB,DNUC,HOL(5),XC,YC,ZC,THC,EMC,DNUV,HOL(6),XD,YD,ZD,THD,EMD,
     1DNUD,HOL(7),XV3,YV3,ZV3,THV3,EMV3,DNUV,HOL(8),XC3,YC3,ZC3,THC3,
     1EMC3,DNUC
 1002 FORMAT(1XA2,6E12.4)
      IF(XF.GT.XFT) WRITE(6,1005)
 1005 FORMAT(20X*NO F POINT*)
      IF(XF.LE.XFT) WRITE(6,1002)HOL(9),XF,YF,ZF,THF,EMF,DNUD
      IF(XG.GT.XGT) WRITE(6,1006)
 1006 FORMAT(20X*NO G POINT*)
      IF(XG.LE.XGT) WRITE(6,1002)HOL(10),XG,YG,ZG,THG,EMG,DNUD
      IF(ITAU.EQ.1)WRITE(6,1002)HOL(11),XV4,YV4,ZV4,THV4,EMV4,DNUV4
      WRITE(6,1013)
 1013 FORMAT(///)
      IF(ISH.EQ.1) GO TO 7384
      WRITE(6,1007)
 1007 FORMAT(1X*PT.*6X*X*11X*Y*11X*Z*8X*THETA*7X*MACH*8X*DNU2D*10X*P*11X
     1*T*/)
      WRITE(6,1010)(HOLE(I),X2(I),Y2(I),Z2(I),TH2(I),EM2(I),DNUE(I),
     1P2(I),T2(I),I=1,6)
 1010 FORMAT(1XA2,8E12.4)
 7384 CONTINUE
      XI=XV1
      THI=TH1(1)
      CALL POLY(XV1,YV1,THI,XV3,THV3,XV2,THV2,XC1,YC1,XC2,THC2,XI)
      CALL        THRUST(XV1,YV1,EMV1,XV3,YV3,THV3,XV2,XV4,TAU,DNUV,
     1XC1,YC1,EMC1,XC3,YC3,XC2,YC2,TAUC,DNUC,PTOT,ITAU,THX)
      WRITE(6,99)
      WRITE(6,99)
      GO TO 6000
  462 CONTINUE
      WRITE(6,992)
  992 FORMAT(*THV2.LT.0*)
  450 CONTINUE
 6000 CONTINUE
 5000 CONTINUE
      END
```

```
      SUBROUTINE CHAR
      COMMON/A/P1(3),TH1(3),EM1(3),A1(3),G1(3)
      COMMON/B/XA,YA,XB,YB,XC,YC,XD,YD,
     1THA,THB,THC,THD,EMA,EMB,EMC,EMD,XMUA,XMUB,XMUC,XMUD
      COMMON/C/DNUV,DNUC,XC1,YC1,XV1,YV1,DNUL
      G=G1(1)
      XMU1=ASIN(1./EM1(1))
      EMA=EM1(1)
      THA=TH1(1)
      CALL FIX(XC1,YC1,TH1(1),XMU1,XV1,YV1,TH1(1),XMU1,
     1XA,YA,THA,XMUA,EMA,G)
      THB=THA-DNUC
      CALL PM(EM1(1),DNUC,EMB,G)
      EMB1=EMB
      XMUB1=ASIN(1./EMB1)
      CALL FIX(XC1,YC1,THB,XMUB1,XA,YA,THA,XMUA,XB,YB,THB,XMUB,EMB,G)
      THC=TH1(1)+DNUV
      CALL PM(EM1(1),DNUV,EMC,G)
      EMC1=EMC
      XMUC1=ASIN(1./EMC1)
      CALL FIX(XA,YA,THA,XMUA,XV1,YV1,THC,XMUC1,XC,YC,THC,XMUC,EMC,G)
   60 DNUD=DNUV+DNUC
      CALL PM(EM1(1),DNUD,EMD,G)
      THD=TH1(1)+DNUV-DNUC
      CALL FIX(XB,YB,THB,XMUB,XC,YC,THC,XMUC,XD,YD,THD,XMUD,EMD,G)
      RETURN
      END
```

```
      SUBROUTINE THRUST(XV1,YV1,EMV1,XV3,YV3,THV3,XV2,XV4,TAU,DNUV,
     1XC1,YC1,EMC1,XC3,YC3,XC2,YC2,TAUC,DNUC,PTOT,ITAU,THX)
      COMMON/A/P1(3),TH1(3),EM1(3),A1(3),G1(3)
      COMMON/D3/ AZ,BZ,CZ
      COMMON /V/ AV,BV,CV
      COMMON/P/ PINF
      COMMON/SHF/ XSHF,YSHF
      COMMON/POL/ IPR,IPOLY
      COMMON/THRMAX/THRMAX
      COMMON/PMPM/ PMOM
  100 FORMAT(8E13.5)
      GETZ(X5)=AZ*X5*X5+BZ*X5+CZ
      PZ(EM5)=PTOT/(1.+(G1(1)-1.)/2.*EM5**2)**(G1(1)/(G1(1)-1.))-PINF
      XLIF=0.
      XTH=0.
      XMOM=0.
      DX=(XV2-XV1)/50.
      N=(XV3-XV1)/DX+1.
      DXN=(XV3-XV1)/FLOAT(N)
      N=N+1
      CALL PM(EM1(1),DNUV,EMX1,G1(1))
      XA=XV1 $ YA=YV1 $ ZA=GETZ(XA)
      CALL EM3D(EMX1,EMA,XA,G1(1))
      PA=PZ(EMA)
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 50
      WRITE(6,110)
  110 FORMAT(///35X*VEHICLE UNDERSURFACE*/7X*X*12X*Y*12X*Z*9X*P/PINF*
     18X*MACH*9X*LIFT*8X*THRUST*7X*MOMENT*)
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   50 CONTINUE
      DO 10 I=2,N
      XB=XA+DXN
      YB=YV1+(XB-XV1)/(XV3-XV1)*(YV3-YV1)
      ZB=GETZ(XB)
      CALL EM3D(EMX1,EMB,XB,G1(1))
      PB=PZ(EMB)
      DUM=(PA+PB)*(ZB+ZA)/4.
      DXLF= DUM*(XB-XA)
      DXTH=DUM*ABS(YB-YA)
      YBB=(YA+YB)/2.-YSHF
      XBB=-((XA+XB)/2.-XSHF)
      DXMOM=-YBB*DXTH+XBB*DXLF
      XLIF=XLIF+DXLF
      XTH=XTH+DXTH
      XMOM=XMOM+DXMOM
      XA=XB $ YA=YB $ ZA=ZB $ PA=PB
      EMA=EMB
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 51
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   51 CONTINUE
   10 CONTINUE
      N=(XV2-XV3)/DX+1.
```

```
      DXN=(XV2-XV3)/FLOAT(N)
      N=N+1
      DO 20 I=2,N
      XB=XA+DXN
      YB=AV+BV*(XB-XV3)+CV*(XB-XV3)**2
      ZB=GETZ(XB)
      THB=ATAN(BV+2.*CV*(XB-XV3))
      TAURB=THB-THV3
      RATB=(XB-XV3)/(XV2-XV3)
      IF(ITAU.EQ.1) RATB=(XB-XV3)/(XV4-XV3)
      IF(RATB.GT.1.) RATB=1.
      DNUB=DNUV+TAURB+2.*RATB*TAU
      X1=XV3 $ X2=XV2
      IF(ITAU.EQ.1) X2=XV4
      RATA=((X1-XC1)/(XB-XC1)-1.)/((X1-XC1)/(X2-XC1)-1.)
      IF(RATA.GE.1.) RATA=1.
      TAUX=ATAN(RATA*TAN(TAU))
      DNUB=DNUV+TAURB+2.*TAUX
      CALL PM(EM1(1),DNUB,EMX,G1(1))
      CALL FM3D(EMX ,EMB,XB,G1(1))
      PB=PZ(EMB)
      DUM=(PA+PB)*(ZB+ZA)/4.
      DXLF= DUM*(XB-XA)
      DXTH=DUM*ABS(YB-YA)
      YBB=(YA+YB)/2.-YSHF
      XBB=-((XA+XB)/2.-XSHF)
      DXMOM=-YBB*DXTH+XBB*DXLF
      XLIF=XLIF+DXLF
      XTH=XTH+DXTH
      XMOM=XMOM+DXMOM
      XA=XB $ YA=YB $ ZA=ZB $ PA=PB
      EMA=EMB
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 52
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   52 CONTINUE
   20 CONTINUE
      XLV=XLIF
      XTHV=XTH
      XMOMV=XMOM
      WRITE(6,101)
  101 FORMAT(/)
      XA=XC1 $ YA=YC1 $ ZA=GETZ(XA)
      XLIF=0.
      XTH=0.
      XMOM=0.
      N=(XC3-XC1)/DX+1.
      DXN=(XC3-XC1)/FLOAT(N)
      N=N+1
      CALL PM(EM1(1),DNUC,EMY1,G1(1))
      CALL FM3D(EMY1,EMA,XA,G1(1))
      PA=PZ(EMA)
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 53
```

```
      WRITE(6,111)
  111 FORMAT(///50X*COWL*                        /7X*X*12X*Y*12X*Z*9X*P/PINF*
     18X*MACH*9X*LIFT*8X*THRUST*7X*MOMENT*)
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   53 CONTINUE
      DO 30 I=2,N
      XB=XA+DXN
      YB=YC1+(XB-XC1)/(XC3-XC1)*(YC3-YC1)
      ZB=GETZ(XB)
      CALL EM3D(EMY1,EMB,XB,G1(1))
      PB=P7(EMB)
      DUM=(PA+PB)*(ZB+ZA)/4.
      DXLF=-DUM*(XB-XA)
      DXTH=DUM*ABS(YB-YA)
      YBB=(YA+YB)/2.-YSHF
      XBB=-((XA+XB)/2.-XSHF)
      DXMOM=-YBB*DXTH+XBB*DXLF
      XLIF=XLIF+DXLF
      XTH=XTH+DXTH
      XMOM=XMOM+DXMOM
      XA=XB $ YA=YB $ ZA=ZB $ PA=PB
      EMA=EMB
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 54
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   54 CONTINUE
   30 CONTINUE
      N=(XC2-XC3)/DX+1.
      IF(N.EQ.1) GO TO 60
      DXN=(XC2-XC3)/FLOAT(N)
      N=N+1
      DO 40 I=2,N
      XB=XA+DXN
      YB=YC3+(XB-XC3)/(XC2-XC3)*(YC2-YC3)
      ZB=GETZ(XB)
      RAT=(XB-XC3)/(XC2-XC3)
      DNUB=DNUC+RAT*2.*TAUC
      RATA=((XC3-XV1)/(XB-XV1)-1.)/((XC3-XV1)/(XC2-XV1)-1.)
      TAUY=ATAN(RATA*TAN(TAUC))
      DNUB=DNUC+2.*TAUY
      CALL PM(EM1(1),DNUB,EMX,G1(1))
      CALL EM3D(EMX ,EMB,XB,G1(1))
      PB=P7(EMB)
      DUM=(PA+PB)*(ZB+ZA)/4.
      DXLF=-DUM*(XB-XA)
      DXTH=DUM*ABS(YB-YA)
      YBB=(YA+YB)/2.-YSHF
      XBB=-((XA+XB)/2.-XSHF)
      DXMOM=-YBB*DXTH+XBB*DXLF
      XLIF=XLIF+DXLF
      XTH=XTH+DXTH
      XMOM=XMOM+DXMOM
      XA=XB $ YA=YB $ ZA=ZB $ PA=PB
      EMA=EMB
      PX=PA+PINF
```

```
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 55
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   55 CONTINUE
   40 CONTINUE
   60 CONTINUE
      XTH=XTH+XTHV
      XLIF=XLIF+XLV
      XMOM=XMOM+XMOMV
      WRITE(6,4203) XLIF
 4203 FORMAT(///20X*TOTAL LIFT =*E12.4)
      WRITE(6,1008) XTH
 1008 FORMAT(/20X*THRUST (FROM COWL AND VEHICLE UNDERSURFACE) =*E12.4)
      WRITE(6,4206) XMOM
 4206 FORMAT(/20X*MOMENT (FROM COWL AND VEHICLE UNDERSURFACE) =*E12.4)
      IF(THX.LT.1.E+09) GO TO 4201
      WRITE(6,4200)
 4200 FORMAT(/20X*EXIT PROFILE AND THRUST AND MOMENT (FROM SIDE WALLS) N
     1OT CALCULATED.*)
      GO TO 4212
 4201 IF(THX.GT.0.) GO TO 4205
      WRITE(6,4204) THX
 4204 FORMAT(/20X*THRUST (FROM SIDE WALLS) =*E12.4)
      PMOM1=0.
      WRITE(6,4211) PMOM1
 4211 FORMAT(/20X*MOMENT (FROM SIDE WALLS) =*E12.4)
      GO TO 4212
 4205 DTHQ=THX-XTH
      WRITE(6,4204) DTHQ
      XTH=THX
      PMOM1=PMOM-XMOM
      WRITE(6,4211) PMOM1
 4212 CONTINUE
      PTHL=(THRMAX-XTH)/THRMAX*100.
      WRITE(6,1563) PTHL
 1563 FORMAT(/40X*PERCENT THRUST LOSS =*F6.3)
      RETURN
      END
```

```
      SUBROUTINE THM(EM1,P1,G,PINF,YV1,YC1,DYV,XV2,THRMAX,XC2)
      COMMON/D3/AZ,BZ,CZ
      Z=AZ*XV2**2+BZ*XV2+CZ
      A1=YV1-YC1
      Z2=AZ*XC2*XC2+BZ*XC2+CZ
      A2=DYV*(Z+Z2)/2.
      FM1=1.+(G-1.)/2.*EM1**2
      DUM=(2.*FM1/(G+1.))**((G+1.)/2./(G-1.))
      AST=A1*EM1/DUM
      PTOT=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=(AFT-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))*PTOT
      F2=P2*A2*(1.+G*EM2**2)
      THRMAX=F2-F1-PINF*(A2-A1)
      RETURN
      END
```

```
      SUBROUTINE XTH(X2,Y2,TH2,DNUE,EM1,G1,PTOT,TTOT,THS2,
     1THX1,       XMASS1,THX)
      DIMENSION XNU(50),X(50),Y(50),Z(50),EM(50),P(50),TH(50),T(50)
      COMMON/P /PINF
      COMMON/D3/AZ,BZ,CZ
      COMMON/SHF/XSHF,YSHF
      COMMON/PMPM/ PMOM
      DIMENSION DNUE(10),X2(10),Y2(10),TH2(10),EM1(03),G1(03)
      GETZ(X5)=AZ*X5*X5+BZ*X5+CZ
      THX=0.
      IF(AZ+BZ.EQ.0.) RETURN
      INDEX=0
  180 CONTINUE
      DO 10 I=1,5
      J1=(I-1)*5+1
      J2=I*5
      IF(I.EQ.5) J2=26
      DO 20 J=J1,J2
      IF(INDEX.EQ.1) GO TO 86
      RAT=FLOAT(J-J1)/FLOAT(J2-J1)
      XNU(J)=DNUE(I)+RAT*(DNUE(I+1)-DNUE(I))
      X(J)=X2(I)+RAT*(X2(I+1)-X2(I))
      Y(J)=Y2(I)+RAT*(Y2(I+1)-Y2(I))
      TH(J)=TH2(I)+RAT*(TH2(I+1)-TH2(I))
      CALL PM(EM1(1),XNU(J),EM(J),G1(1))
      CALL FM3D(EM(J),EMX,X(J),G1(1))
      EM(J)=EMX
      Z(J)=GETZ(X(J))
   86 CONTINUE
      DUM=1.+(G1(1)-1.)/2.*EM(J)*EM(J)
      DG=G1(1)/(G1(1)-1.)
      P(J)=PTOT/DUM**DG
      T (J)=TTOT/DUM
   20 CONTINUE
   10 CONTINUE
      DUM=G1(1)*EM (1)**2*SIN(THS2-TH (1))/SIN(THS2)
      E1=P(1)*EM(1)/SQRT(T(1))*SIN(THS2-TH(1))/
     1SIN(THS2)*Z (1)
      U=1.-PINF/P (1)
      F1=P (1)*(U +DUM*COS(TH (1)))*Z (1)
      W1=P(1)*(U/TAN(THS2)+DUM*SIN(TH(1)))*Z(1)
      XMASS2=0.
      THX2=0.
      PMOM=0.
      DO 150 I=2,26
      E2=P(I)*EM(I)/SQRT(T(I))*
     1SIN(THS2-TH (I))/SIN(THS2)*Z (I)
      DUM=G1(1)*EM (I)**2*SIN(THS2-TH (I))/SIN(THS2)
      U=1.-PINF/P (I)
      F2=P (I)*(U +DUM*COS(TH (I)))*Z (I)
      W2=P(I)*(U/TAN(THS2)+DUM*SIN(TH(I)))*Z(I)
      THX2=THX2+.5*(F1+F2)*(Y (I)-Y (I-1))
      XMASS2=XMASS2+.5*(E1+E2)*(Y (I)-Y (I-1))
      DUM1=.5*(F1*(Y(I-1)-YSHF)+F2*(Y(I)-YSHF))
      DUM2=.5*(W1*(X(I-1)-XSHF)+W2*(X(I)-XSHF))
      PMOM=PMOM+(DUM1-DUM2)*(Y(I)-Y(I-1))
```

```
      E1=E2 $ F1=F2 $ W1=W2
  150 CONTINUE
  100 FORMAT(7E13.5)
      RATM=XMASS2/XMASS1
      EMS=RATM-1
      IF(ABS(EMS).LT.1.E-03)GO TO 4625
      INDEX=INDEX+1
      IF(INDEX.GT.1)GO TO 4625
      DO 170 I=1,26
      FM1=FM(EM (I),G1(1))
      FM2=FM1/RATM
      ITM=1
      EMT=RATM*EM (I)
  197 CONTINUE
      FMT=FM(EMT,G1(1))
      ERM=(FM2-FMT)/FM2
      IF(ABS(ERM).LT.1.E-03)GO TO 171
      DUMD=1./RATM
      CALL ERROR(6,ITM,EMT,ERM,DUMD,EMT1,ERM1)
      GO TO 197
  171 CONTINUE
      EM (I)=EMT
  170 CONTINUE
      GO TO 180
 4625 CONTINUE
      THX=THX2-THX1
      RETURN
      END
```

```
      SUBROUTINE EM3D(EM,EMX,X,G)
      COMMON/D3/ AZ,BZ,CZ
      Z=AZ*X*X+BZ*X+CZ
      GN=(G+1.)/(G-1.)/2.
      Y=(1.+(G-1.)/2.*EM*EM)**GN
      Y=Y/EM/((G+1.)/2.)**GN
      A=Y*Z
      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 FIX(X1,Y1,TH1,XMU1,X2,Y2,TH2,XMU2,X3,Y3,TH3,XMU3,EM3,G)
      SL1=TAN(TH1+XMU1)
      SL2=TAN(TH2-XMU2)
      CALL GEM(X1,Y1,SL1,X2,Y2,SL2,X3,Y3)
      ITM=0
   10 CONTINUE
      CALL EM3D(EM3,EM3N,X3,G)
      XMU3=ASIN(1./EM3N)
      SL13=SL1/2.+.5*TAN(TH3+XMU3)
      IF(XMU1.EQ.0.)SL13=SL1
      SL23=SL2/2.+.5*TAN(TH3-XMU3)
      IF(XMU2.EQ.0.)SL23=SL2
      CALL GEM(X1,Y1,SL13,X2,Y2,SL23,X3,Y3)
      IF(ITM.EQ.1) GO TO 20
      ITM=1
      GO TO 10
   20 EM3=EM3N
      RETURN
      END
```

```
      SUBROUTINE PM (EM1,DNU,EM2,G)
      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.)
      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,1,11,EM21,ERNU1)
      GO TO 10
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE POLY(XV1,YV1,THI,XV3,THV3,XV2,THV2,
     1XC1,YC1,XC2,THC2,XI)
      COMMON/Y/A11,A12,A13,A21,A22,A23,A31,A32,A33
      COMMON/POL/ IPR,IPOLY
      DIMENSION X(10),Y(10),TH(10)
      IF(IPOLY.EQ.1) GO TO 20
      IT=0
      X(1)=XV1
      X(2)=XV1+.01*(XV2-XV1)
      X(3)=XV3
      X(4)=XV2
      TH(1)=THI
      TH(2)=THV3
      TH(3)=THV3
      TH(4)=THV2
      TH(1)=-TH(1) $ TH(2)=-TH(2) $ TH(3)=-TH(3) $ TH(4)=-TH(4)
      Y(1)=YC1
      N=3
      WRITE(6,100)
  100 FORMAT(///30X*VEHICLE COORDINATES*/
     115X*X*4X*TO*4X*X*15X*COORDINATES*)
   30  DO 10 I=1,N
      X2=X(I+1)-XI
      X1=X(I)-XI
      A=(TAN(TH(I+1))-TAN(TH(I)))/2./(X2-X1)
      B=TAN(TH(I+1))-2.*A*X2
      C=Y(I)-B*X1-A*X1*X1
      Y(I+1)=A*X2*X2+B*X2+C
      WRITE(6,69) X(I),X(I+1),A,B,C
   69 FORMAT(10X5E11.3)
   10 CONTINUE
      IF(IT.EQ.1)GO TO 20
      IT=1
      X(1)=XC1 $ X(2)=X(1)+.01*(XV2-XV1)
      X(3)=XC2
      TH(2)=THC2
      TH(3)=TH(2)
      TH(2)=-TH(2) $ TH(3)=-TH(3)
      Y(1)=YV1
      N=2
      WRITE(6,101)
  101 FORMAT(/30X*COWL COORDINATES*/
     115X*X*4X*TO*4X*X*15X*COORDINATES*)
      GO TO 30
   20 CONTINUE
      RETURN
      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
```

APPENDIX III

LISTING OF EQUILIBRIUM FLOW PROGRAM

```
      PROGRAM NOZD(INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
      COMMON/A/ PI,THI,EMI,GI,TI,HI,PHI,UI
      COMMON/B/XA,YA,XB,YB,XC,YC,XD,YD,
     1THA,THB,THC,THD,EMA,EMB,EMC,EMD,XMUA,XMUB,XMUC,XMUD
      COMMON/C/DNUV,DNUC,XC1,YC1,XV1,YV1,DNUL
      COMMON/D/ PHI
      COMMON/BR/PA,PB,PC,PD,TA,TB,TC,TD,GA,GB,GC,GD,UA,UB,UC,UD,RHA,
     1RHB,RHC,RHD
      COMMON/D3/ AZ,BZ,CZ
      COMMON/V/AV,BV,CV
      COMMON/P/PINF
      COMMON/SHF/XSHF,YSHF
      COMMON/POL/IPR,IPOLY
      COMMON/THRMAX/THRMAX
      DIMENSION P2(10),TH2(10),EM2(10),T2(10),G2(10),X2(10),Y2(10)
     1,Z2(10),RH2(10),U2(10)
      DIMENSION HOL(11),HOLE(6),DNUE(10)
      DATA HOLE/2HE0,2HE1,2HE2,2HE3,2HE4,2HE5/
      DATA HOL/2HV1,2HC1,2HA ,2HB ,2HC ,2HD ,2HV3,2HC3,2HF ,2HG ,2HV4/
      GETZ(X5)=AZ*X5*X5+BZ*X5+CZ
   98 FORMAT(7E13.5)
   99 FORMAT(/)
  100 FORMAT(8F10.0)
      READ(5,100) PI,THI,EMI,TI,PHI,PINF
      HI=FH(PI,PHI,TI)
      GI=FGAM(TI,PI,PHI)
      RHI=RHEQ(HI,PI,PHI,TI)
      AI=SQRT(GI*PI/RHI)
      UI=EMI*AI
      TAURC=0.
      READ(5,100) XV1,YV1,XV2,XC1,YC1,XC2,DYV,DNUC
      DNUC=DNUC/57.3
      READ(5,5921) ICF,DXC,JTF,DTH,IPR,IPOLY
 5921 FORMAT(I5,E10.0,I5,E10.0,2I5)
      READ(5,100) XSHF,YSHF
      READ(5,100) AZ,BZ,CZ
      ZC1=GETZ(XC1)
      ZV1=GETZ(XV1)
      ZC2=GETZ(XC2)
      WRITE(6,5922)
 5922 FORMAT(1H1/////)
      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,PHI,PINF
  400 FORMAT(     ///30X,*INITIAL PROFILE*//3X*PRESSURE*6X*THETA*9X*MACH*
     16X*TEMPERATURE*6X*PHI*9X*PINF*/6E13.5)
      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///)
      WRITE(6,5923) XSHF,YSHF
 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)
      XC22=XC2
```

```
      THC1=THI    -DNUC
      YC2=YC1+TAN(THC1)*(XC2-XC1)
      YV2=YC2+DYV
      DNUV=ATAN((YV2-YV1)/(XV2-XV1))-THI
      DO 5000 IXC=1,ICF
      XC2=XC22+FLOAT(IXC-1)*DXC
      CALL THM(RHI,UT,PI,PINF,YV1,YC1,DYV,XV2,THRMAX,DNUV,DNUC,XC2)
      WRITE(6,6364) THRMAX
 6364 FORMAT(1H131X*IDEAL THRUST =*E12.4)
      YC2=YC1+TAN(THC1)*(XC2-XC1)
      YV2=YC2+DYV
      DO 6000 JT=1,JTF
      THX=1.E+10
      THJT=FLOAT(JT-1)*DTH
      DNUV=ATAN((YV2-YV1)/(XV2-XV1))-THI+THJT/57.3
      CALL PM(DNUC,PC1,TC1,EMC1,GC1,UC1,RHC1)
      CALL PM(DNUV,PV1,TV1,EMV1,GV1,UV1,RHV1)
      IT2=1
 1000 CONTINUE
      CALL CHAR
      DNUA=0.
      DNUD=DNUV+DNUC
      ZA=GETZ(XA)
      ZB=GETZ(XB)
      ZC=GETZ(XC)
      ZD=GETZ(XD)
      RAT=0.
      THC3=THC1
      CALL PM(DNUC,PC3,TC3,EMC3,GC3,UC3,RHC3)
      CALL FIX(XC1,YC1,THC1,0.,XB,YB,THB,XMUB,XC3,YC3,THC3,XMUC3,EMC3,
     1GC3,
     1PC3,TC3,RHC3,UC3,DNUC)
      ZC3=GETZ(XC3)
      IT176=0
      IF(XC3.LT.XC2) GO TO 177
      IT176=1
      TAUC=0.
      XC3=XC2
      DNUE0=DNUC
      GO TO 178
  177 CONTINUE
      IT9=1
      RAT=.1
  176 CONTINUE
      XMUBD=XMUB+RAT*(XMUD-XMUB)
      THBD=THB+RAT*(THD-THB)
      XT=XB+RAT*(XD-XB)
      YT=YB+RAT*(YD-YB)
      TAUC=RAT*DNUV
      DNUW=DNUC+TAUC
      CALL PM(DNUW,PW,TW,EMW,GW,UW,RHW)
      THW=THC3+TAUC
      CALL FIX(XC3,YC3,THC3,XMUC3,XT,YT,THBD,XMUBD,XW,YW,THW,XMUW,EMW,
     1GW,
     1PW,TW,RHW,UW,DNUW)
      DNUE0=DNUC+2.*TAUC
```

```
  178 CONTINUE
      THC2=THC1
      THE0=THC2
      CALL PM(DNUE0,PE0,TE0,EME0,GE0,UE0,RHE0)
      ZE0=GETZ(XC2)
      RQ3D=RHE0*UE0/ZE0
      DNU3E=DNUE0*SQRT(ZE0)
      ITE0=1
  200 CONTINUE
      CALL PM(DNU3E,PE0,TE0,EME0,GE0,UE0,RHE0)
      RQ3T=RHE0*UE0
      ERQE=(RQ3T-RQ3D)/RQ3D
      IF(ABS(ERQE).LT.1.E-03) GO TO 210
      CALL ERROR(200,ITE0,DNU3E,ERQE,1.1,DNU3E1,ERQE1)
      GO TO 200
  210 CONTINUE
      XMUE0=ASIN(1./EME0)
      IF(IT176.EQ.1) GO TO 175
      SLWC2=(TAN(THW-XMUW)+TAN(THE0-XMUE0))/2.
      YC2T=YW+SLWC2*(XC2-XW)
      ER9=YC2-YC2T
      IF(ABS(ER9).LT.1.E-03) GO TO 175
      CALL ERROR(4,IT9,RAT,ER9,1.11,RAT1,ER91)
      GO TO 176
  175 CONTINUE
      THV3=THI    +DNUV
      CALL PM(DNUV,PV3,TV3,EMV3,GV3,UV3,RHV3)
      THV1=THV3
      CALL FIX(XC,YC,THC,XMUC,XV1,YV1,THV1,0.,XV3,YV3,THV3,XMUV3,EMV3,
     1GV3,
     1PV3,TV3,RHV3,UV3,DNUV)
      AV=YV3 $ BV=TAN(THV3)
      CV=(YV2-AV-BV*(XV2-XV3))/(XV2-XV3)**2
      ZV3=GETZ(XV3)
      THV2=ATAN(BV+2.*CV*(XV2-XV3))
      IF(THV2.LT.0.) GO TO 462
      TAURV=THV2-THV3
      ITZ=1
      ITAU=0
      RAT=1.
  545 CONTINUE
  530 CONTINUE
      XZ=XC+RAT*(XD-XC)
      YZ=YC+RAT*(YD-YC)
      TAU=RAT*DNUC
      THZ=THC+RAT*(THD-THC)
      EMZ=EMC+RAT*(EMD-EMC)
      XMUZ=ASIN(1./EMZ)
      THY=THV3-TAU
      DNUY=DNUV+TAU
      CALL PM(DNUY,PY,TY,EMY,GY,UY,RHY)
      CALL FIX(XZ,YZ,THZ,XMUZ,XV3,YV3,THV3,XMUV3,XY,YY,THY,XMUY,EMY,GY,
     1PY,TY,RHY,UY,DNUY)
      DNUE5=DNUV+2.*TAU+TAURV
      THE5=THV2
      CALL PM(DNUE5,PE5,TE5,EME5,GE5,UE5,RHE5)
```

```
      ZE5=GETZ(XV2)
      RO3D=RHE5*UE5/ZE5
      DNU35=DNUE5*SQRT(ZE5)
      ITE5=1
  220 CONTINUE
      CALL PM(DNU35,PE5,TE5,EME5,GE5,UE5,RHE5)
      RO3T=RHE5*UE5
      ERO5=(RO3T-RO3D)/RO3D
      IF(ABS(ERO5).LT.1.E-03) GO TO 230
      CALL ERROR(220,ITE5,DNU35,ERO5,.1,1,DNU351,ERO51)
      GO TO 220
  230 CONTINUE
      XMUE5=ASIN(1./EME5)
      SL2T=(TAN(THY+XMUY)+TAN(THE5+XMUE5))/2.
      YV2T=YY+SL2T*(XV2-XY)
      IF(ITZ.EQ.1.AND.YV2T.GT.YV2) GO TO 542
      EZ=YV2T-YV2
      IF(ABS(EZ).LT.1.E-03)GO TO 540
      IF(DNUC.EQ.0.) GO TO 540
      CALL ERROR(5,ITZ,RAT,EZ,.1   ,RAT1,EZ1)
      GO TO 530
  542 ITAU=1
  540 CONTINUE
      DNUE2=DNUV+DNUC
      CALL PM(DNUE2,PE2,TE2,EME2,GE2,UE2,RHE2)
      THE2=THI    +DNUV-DNUC
      THF=THE2
      RHF=RHE2 $ UF=UE2
      CALL FIX(XC3,YC3,THC3,XMUC3,XD,YD,THD,XMUD,XF,YF,THF,XMUF,EMF,
     1GF,
     1PF,TF,RHF,UF,DNUE2)
      SLE=(YV2-YC2)/(XV2-XC2)
      THE=ATAN(SLE)
      UFT=UF2 $ RHFT=RHF2
      CALL FIX(XC3,YC3,THC3,XMUC3,XC2,YC2,THE,0.,XFT,YFT,THF,XMUFT,EMFT,
     1GFT,
     1PFT,TFT,RHFT,UFT,DNUE2)
      IF(XF.GT.XFT) GO TO 300
      CALL FIX(XF,YF,THF,XMUF,XC2,YC2,THE,0.,XE2,YE2,THE2,XMUE2,EME2,
     1GE2,
     1PE2,TE2,RHE2,UE2,DNUE2)
      ZF=GETZ(XF)
      SLF=TAN(THF-XMUF)
      CALL GEM(XF,YF,SLF,XC2,YC2,SLE,XE1,YE1)
      ITE1=0
  368 CONTINUE
      RAT=(YE1-YC2)/(YE2-YC2)
      DNUE1=DNUV+DNUC+(1.-RAT)*TAUC
      CALL PM(DNUE1,PE1,TE1,EME1,GE1,UE1,RHE1)
      THE1=THI    +DNUV-DNUC-(1.-RAT)*TAUC
      CALL FIX(XC2,YC2,THE,0.,XF,YF,THF,XMUF,XE1,YE1,THE1,XMUE1,EME1,
     1GE1,
     1PE1,TE1,RHE1,UE1,DNUE1)
      IF(ITE1.EQ.1) GO TO 369
      ITE1=1
      GO TO 368
```

```
  369 CONTINUE
      GO TO 310
  300 CONTINUE
      CALL FIX(XC2,YC2,THE,0.,XD,YD,THD,XMUD,XE2,YE2,THE2,XMUE2,EME2,
     1GE2,
     1PE2,TE2,RHE2,UE2,DNUE2)
      SLCE=TAN(THC3+XMUC3)
      CALL GEM(XC3,YC3,SLCE,XC2,YC2,SLE,XE1,YE1)
      ITE1=0
  381 CONTINUE
      RAT=(YE1-YC2)/(YE2-YC2)
      DNUE1=DNUC+TAUC+RAT*(DNUV-TAUC)
      CALL PM(DNUE1,PE1,TE1,EME1,GE1,UE1,RHE1)
      THE1=THI    -DNUC+TAUC+RAT*(DNUV-TAUC)
      CALL FIX(XC3,YC3,THC3,XMUC3,XC2,YC2,THE,0.,XE1,YE1,THE1,XMUE1,EME1
     1,GE1,
     1PE1,TE1,RHE1,UE1,DNUE1)
      IF(ITE1.EQ.1) GO TO 380
      ITE1=1
      GO TO 381
  380 CONTINUE
  310 CONTINUE
      THE3=THE2
      DNUE3=DNUE2
      CALL PM(DNUE3,PE3,TE3,EME3,GE3,UE3,RHE3)
      XMUE3=ASIN(1./EME3)
      IF(ITAU.NE.1) GO TO 311
      XG=XY $ YG=YY $ THG=THY $ RHG=RHY $ UG=UY $ EMG=EMY $ XMUG=XMUY
      PG=PY $ TG=TY
      XGT=XG
      GO TO 3588
  311 CONTINUE
      THG=THE3
      RHG=RHE3 $ UG=UE3
      CALL FIX(XD,YD,THD,XMUD,XV3,YV3,THV3,XMUV3,XG,YG,THG,XMUG,EMG,
     1GG,
     1PG,TG,RHG,UG,DNUE3)
      RHGT=RHE3 $ UGT=UE3
      CALL FIX(XC2,YC2,THE,0.,XV3,YV3,THV3,XMUV3,XGT,YGT,THG,XMUGT,EMGT,
     1GGT,
     1PGT,TGT,RHGT,UGT,DNUE3)
      IF(XG.GT.XGT) GO TO 350
 3588 CONTINUE
      CALL FIX(XC2,YC2,THE,0.,XG,YG,THG,XMUG,XE3,YE3,THE3,XMUE3,EME3,
     1GE3,
     1PE3,TE3,RHE3,UE3,DNUE3)
      ZG=GETZ(XG)
      IF(YE3.LE.YE2) GO TO 359
      SLE4=TAN(THG+XMUG)
      CALL GEM(XG,YG,SLE4,XC2,YC2,SLE,XE4,YE4)
      IT4=0
  358 CONTINUE
      RAT=(YE4-YV2)/(YE3-YV2)
      DNUE4=DNUV+DNUC+(1.-RAT)*(TAURV+TAU)
      CALL PM(DNUE4,PE4,TE4,EME4,GE4,UE4,RHE4)
      THE4=THI    +DNUV-DNUC+(1.-RAT)*(TAURV+TAU)
```

```
      CALL FIX(XG,YG,THG,XMUG,XC2,YC2,THE ,0.,XE4,YE4,THE4,XMUE4,EME4,
     1GE4,
     1PE4,TE4,RHE4,UE4,DNUE4)
      IF(IT4.EQ.1) GO TO 359
      IT4=1
      GO TO 358
  359 CONTINUE
      IF(ITAU.EQ.0) GO TO 360
      SLV32=(YV2-YV3)/(XV2-XV3)
      SLG=TAN(THG+XMUG)
      CALL GEM(XV3,YV3,SLV32,XG,YG,SLG,XV4,YV4)
      ITV4=1
  362 YV4=AV+BV*(XV4-XV3)+CV*(XV4-XV3)**2
      THV4=ATAN(BV+2.*CV*(XV4-XV3))
      TAURV4=THV4-THV3
      DNUV4=DNUV+2.*TAU+TAURV4
      CALL PM(DNUV4,PV4,TV4,EMV4,GV4,UV4,RHV4)
      CALL FIX(XV4,YV4,.7854,0.,XV4,YV4,0.,0.,XV4,YV4,THV4,XMUV4,EMV4,
     1GV4,PV4,TV4,RHV4,UV4,DNUV4)
      YV4T=YG+.5*(TAN(THG+XMUG)+TAN(THV4+XMUV4))*(XV4-XG)
      ERV4=(YV4T-YV4)/YV2
      IF(ABS(ERV4).LT.1.E-03) GO TO 361
      CALL ERROR(361,ITV4,XV4,ERV4,1.1,XV41,ERV41)
      ZV4=GETZ(XV4)
      IF(ITV4.EQ.1) GO TO 361
      GO TO 362
  361 DNUE4=DNUV4
      IF(YE3.LE.YE2) GO TO 460
      CALL PM(DNUE4,PE4,TE4,EME4,GE4,UE4,RHE4)
      THE4=THV4
      CALL FIX(XC2,YC2,THE,0.,XV4,YV4,THV4,XMUV4,XE4,YE4,THE4,XMUE4,
     1EME4,GE4,
     1PE4,TE4,RHE4,UE4,DNUE4)
      GO TO 360
  350 CONTINUE
      IT4=0
      CALL FIX(XD,YD,THD,XMUD,XC2,YC2,THE,0.,XE3,YE3,THE3,XMUE3,EME3,
     1GE3,
     1PE3,TE3,RHE3,UE3,DNUE3)
      IF(YE3.LE.YE2) GO TO 460
      SLE4=TAN(THV3-XMUV3)
      CALL GEM(XV3,YV3,SLE4,XC2,YC2,SLE,XE4,YE4)
  371 CONTINUE
      RAT=(YE4-YV2)/(YE3-YV2)
      DNUE4=DNUV+TAU+RAT*(DNUC-TAU)
      CALL PM(DNUE4,PE4,TE4,EME4,GE4,UE4,RHE4)
      THE4=THI    +DNUV-TAU-RAT*(DNUC-TAU)
      CALL FIX(XC2,YC2,THE,0.,XV3,YV3,THV3,XMUV3,XE4,YE4,THE4,XMUE4,EME4
     1,GE4,
     1PE4,TE4,RHE4,UE4,DNUE4)
      IF(IT4.EQ.1)GO TO 370
      IT4=1
      GO TO 371
  370 CONTINUE
  360 CONTINUE
      IF(YE3.LE.YE2) GO TO 460
```

```
      X2(1)=XC2 $ X2(2)=XE1 $ X2(3)=XE2 $ X2(4)=XE3 $ X2(5)=XE4
      X2(6)=XV2
      Y2(1)=YC2 $ Y2(2)=YE1 $ Y2(3)=YE2 $ Y2(4)=YE3 $ Y2(5)=YE4
      Y2(6)=YV2 $ EM2(1)=EME0 $ EM2(2)=EME1 $ EM2(3)=EME2 $ EM2(4)=EME3
      EM2(5)=EME4 $ EM2(6)=EME5 $ TH2(1)=THE0 $ TH2(2)=THE1
      TH2(3)=THE2 $ TH2(4)=THE3 $ TH2(5)=THE4 $ TH2(6)=THE5
      DNUE(1)=DNUE0 $ DNUE(2)=DNUE1 $ DNUE(3)=DNUE2 $ DNUE(4)=DNUE3
      DNUE(5)=DNUE4 $ DNUE(6)=DNUE5
      RH2(1)=RHE0 $ RH2(2)=RHE1 $ RH2(3)=RHE2 $ RH2(4)=RHE3
      RH2(5)=RHE4 $ RH2(6)=RHE5
      U2(1)=UE0 $ U2(2)=UE1 $ U2(3)=UE2 $ U2(4)=UE3 $ U2(5)=UE4
      U2(6)=UE5
      G2(1)=GE0 $ G2(2)=GE1 $ G2(3)=GE2 $ G2(4)=GE3 $ G2(5)=GE4
      G2(6)=GE5
      P2(1)=PE0 $ P2(2)=PE1 $ P2(3)=PE2 $ P2(4)=PE3 $ P2(5)=PE4
      P2(6)=PE5
      T2(1)=TE0 $ T2(2)=TE1 $ T2(3)=TE2 $ T2(4)=TE3 $ T2(5)=TE4
      T2(6)=TE5
      INDEX=0
  180 CONTINUE
      DO 160 I=1,6
      Z2(I)=AZ*X2(I)**2+BZ*X2(I)+CZ
  160 CONTINUE
      THS1=1.5707963
      IF(XV1.NE.XC1) THS1=ATAN((YV1-YC1)/(XV1-XC1))
      THS2=ATAN((YV2-YC2)/(XV2-XC2))
      ZIQ=.5*(GETZ(XC1)+GETZ(XV1))
      XMASS1=RHI*UI*SIN(THS1-THI)/SIN(THS1)*ZIQ*(YV1-YC1)
      THX1=(RHI*UI*UI*SIN(THS1-THI)*COS(THI)+(PI-PINF)*SIN(THS1))*ZIQ*(Y
     1V1-YC1)/SIN(THS1)
      CALL XTH(X2,Y2,TH2,DNUE,PINF,THS2,THX1,XMASS1,THX)
      ISH=0
      GO TO 461
  460 CONTINUE
      ISH=1
      WRITE(6,991)
  991 FORMAT(* YE3.LE.YE2*)
  461 CONTINUE
      WRITE(6,1004) XC2,DNUV,DNUC,TAU,     TAUC,TAURV,TAURC
 1004 FORMAT(      32X*XC2    =*E12.4/32X*DNUV   =*E12.4/32X*DNUC   =*E12.4/
     132X*TAU    =*E12.4/ 32X*TAUC   =*E12.4/32X*TAURV  =*E12.4/32X*TAURC =
     1*E12.4///)
      WRITE(6,1007)
      WRITE(6,1010)HOL(1),XV1,YV1,ZV1,THV1,EMV1,DNUV,PV1,TV1,HOL(2),XC1,
     1YC1,ZC1,THC1,EMC1,DNUC,PC1,TC1,HOL(3),XA,YA,ZA,THA,EMA,DNUA,PA,TA
     1,HOL(4),XB,YB,ZB,
     1THB,EMB,DNUC,PB,TB,HOL(5),XC,YC,ZC,THC,EMC,DNUV,PC,TC,HOL(6),XD,YD
     1,ZD,THD,EMD,
     1DNUD,PD,TD,HOL(7),XV3,YV3,ZV3,THV3,EMV3,DNUV,PV3,TV3,HOL(8),XC3,YC
     13,ZC3,THC3,
     1EMC3,DNUC,PC3,TC3
      IF(XF.GT.XFT) WRITE(6,1005)
 1005 FORMAT(20X*NO F POINT*)
      IF(XF.LE.XFT) WRITE(6,1010)HOL(9),XF,YF,ZF,THF,EMF,DNUD,PF,TF
      IF(XG.GT.XGT) WRITE(6,1006)
 1006 FORMAT(20X*NO G POINT*)
```

```
      IF(XG.LE.XGT) WRITE(6,1010)HOL(10),XG,YG,ZG,THG,EMG,DNUD,PG,TG
      IF(ITAU.EQ.1)WRITE(6,1010)HOL(11),XV4,YV4,ZV4,THV4,EMV4,DNUV4,PV4
     1,TV4
      WRITE(6,1013)
 1013 FORMAT(///)
      IF(ISH.EQ.1) GO TO 7384
      WRITE(6,1007)
 1007 FORMAT(1X*PT.*6X*X*11X*Y*11X*Z*8X*THETA*7X*MACH*8X*DNU2D*10X*P*11X
     1*T*/)
      WRITE(6,1010)(HOLE(I),X2(I),Y2(I),Z2(I),TH2(I),EM2(I),DNUE(I),
     1P2(I),T2(I),I=1,6)
 1010 FORMAT(1XA2,8E12.4)
 7384 CONTINUE
      XI=XV1
      CALL POLY(XV1,YV1,THI,XV3,THV3,XV2,THV2,XC1,YC1,XC2,THC2,XI)
      CALL THRUST(XV1,YV1,EMV1,XV3,YV3,THV3,XV2,XV4,TAU,DNUV,XC1,YC1,
     1EMC1,XC3,YC3,XC2,YC2,TAUC,DNUC,ITAU,THC1,THV1,THX)
      WRITE(6,99)
      WRITE(6,99)
      GO TO 6000
  462 CONTINUE
      WRITE(6,992)
  992 FORMAT(*THV2.LT.0*)
  450 CONTINUE
 6000 CONTINUE
 5000 CONTINUE
      END
```

```
      SUBROUTINE CHAP
      COMMON/A/ PI,THI,EMI,GI,TI,HI,RHI,UI
      COMMON/B/XA,YA,XB,YB,XC,YC,XD,YD,
     1THA,THB,THC,THD,EMA,EMB,EMC,EMD,XMUA,XMUB,XMUC,XMUD
      COMMON/C/DNUV,DNUC,XC1,YC1,XV1,YV1,DNUL
      COMMON/BB/PA,PB,PC,PD,TA,TB,TC,TD,GA,GB,GC,GD,UA,UB,UC,UD,RHA,
     1RHB,RHC,RHD
      COMMON/D/ PHI
      XMUI=ASIN(1./EMI)
      THA=THI $ RHA=RHI $ UA=UI
      DNUA=0.
      CALL FIX(XC1,YC1,THI    ,XMUI,XV1,YV1,THI    ,XMUI,
     1XA,YA,THA,XMUA,EMA,GA,
     1PA,TA,RHA,UA,DNUA)
      THB=THA-DNUC
      CALL PM(DNUC,PB,TB,EMB,GB,UB,RHB)
      EMB1=EMB
      XMUB1=ASIN(1./EMB1)
      CALL FIX(XC1,YC1,THB,XMUB1,XA,YA,THA,XMUA,XB,YB,THB,XMUB,EMB,GB,
     1PB,TB,RHB,UB,DNUC)
      THC=THI    +DNUV
      CALL PM(DNUV,PC,TC,EMC,GC,UC,RHC)
      EMC1=EMC
      XMUC1=ASIN(1./EMC1)
      CALL FIX(XA,YA,THA,XMUA,XV1,YV1,THC,XMUC1,XC,YC,THC,XMUC,EMC,GC,
     1PC,TC,RHC,UC,DNUV)
   60 DNUD=DNUV+DNUC
      CALL PM(DNUD,PD,TD,EMD,GD,UD,RHD)
      THD=THI    +DNUV-DNUC
      CALL FIX(XB,YB,THB,XMUB,XC,YC,THC,XMUC,XD,YD,THD,XMUD,EMD,GD,
     1PD,TD,RHD,UD,DNUD)
      RETURN
      END
```

```
      SUBROUTINE THRUST(XV1,YV1,EMV1,XV3,YV3,THV3,XV2,XV4,TAU,DNUV,
     1XC1,YC1,EMC1,XC3,YC3,XC2,YC2,TAUC,DNUC,      ITAU,THC1,THV1,THX)
      COMMON/D3/ AZ,BZ,CZ
      COMMON /V/ AV,BV,CV
      COMMON/P/ PINF
      COMMON/SHF/ XSHF,YSHF
      COMMON/POL/ IPR,IPOLY
      COMMON/THRMAX/THRMAX
      COMMON/PMPM/ PMOM
  100 FORMAT(8E13.5)
      GETZ(X5)=AZ*X5*X5+BZ*X5+CZ
      XLIF=0.
      XTH=0.
      XMOM=0.
      DX=(XV2-XV1)/50.
      N=(XV3-XV1)/DX+1.
      DXN=(XV3-XV1)/FLOAT(N)
      N=N+1
      CALL PM(DNUV,PA,TA,EMA,GA,UA,RHA)
      XA=XV1 $ YA=YV1 $ ZA=GETZ(XA)
      THA=THV1
      CALL FIX(XA,YA,.7854,0.,XA,YA,0.,0.,XA,YA,THA,XMUA,EMA,GA,PA,TA,
     1RHA,UA,DNUV)
      PA=PA-PINF
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 50
      WRITE(6,110)
  110 FORMAT(///35X*VEHICLE UNDERSURFACE*/7X*X*12X*Y*12X*Z*9X*P/PINF*
     18X*MACH*9X*LIFT*8X*THRUST*7X*MOMENT*)
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   50 CONTINUE
      DO 10 I=2,N
      XB=XA+DXN
      YB=YV1+(XB-XV1)/(XV3-XV1)*(YV3-YV1)
      ZB=GETZ(XB)
      THB=ATAN((YV3-YV1)/(XV3-XV1))
      CALL PM(DNUV,PB,TB,EMB,GB,UB,RHB)
      CALL FIX(XB,YB,.7854,0.,XB,YB,0.,0.,XB,YB,THB,XMUB,EMB,GB,PB,TB,
     1RHB,UB,DNUV)
      PB=PB-PINF
      DUM=(PA+PB)*(ZB+ZA)/4.
      DXLF= DUM*(XB-XA)
      DXTH=DUM*ABS(YB-YA)
      YBB=(YA+YB)/2.-YSHF
      XBB=-((XA+XB)/2.-XSHF)
      DXMOM=-YBB*DXTH+XBB*DXLF
      XLIF=XLIF+DXLF
      XTH=XTH+DXTH
      XMOM=XMOM+DXMOM
      XA=XB $ YA=YB $ ZA=ZB $ PA=PB
      EMA=EMB
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 51
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
```

```
   51 CONTINUE
   10 CONTINUE
      N=(XV2-XV3)/DX+1.
      DXN=(XV2-XV3)/FLOAT(N)
      N=N+1
      DO 20 I=2,N
      XB=XA+DXN
      YB=AV+BV*(XB-XV3)+CV*(XB-XV3)**2
      ZB=GETZ(XB)
      THB=ATAN(BV+2.*CV*(XB-XV3))
      TAURB=THB-THV3
      RATB=(XB-XV3)/(XV2-XV3)
      IF(ITAU.EQ.1) RATB=(XB-XV3)/(XV4-XV3)
      IF(RATB.GT.1.) RATB=1.
      DNUB=DNUV+TAURB+2.*RATB*TAU
      X1=XV3 $ X2=XV2
      IF(ITAU.EQ.1) X2=XV4
      RATA=((X1-XC1)/(XB-XC1)-1.)/((X1-XC1)/(X2-XC1)-1.)
      IF(RATA.GE.1.) RATA=1.
      TAUX=ATAN(RATA*TAN(TAU))
      DNUB=DNUV+TAURB+2.*TAUX
      CALL PM(DNUB,PB,TB,EMB,GB,UB,RHB)
      CALL FIX(XB,YB,.7854,0.,XB,YB,0.,0.,XB,YB,THB,XMUB,EMB,GB,PB,TB,
     1RHB,UB,DNUB)
      PB=PB-PINF
      DUM=(PA+PB)*(ZB+ZA)/4.
      DXLF= DUM*(XB-XA)
      DXTH=DUM*ABS(YB-YA)
      YBB=(YA+YB)/2.-YSHF
      XBB=-((XA+XB)/2.-XSHF)
      DXMOM=-YBB*DXTH+XBB*DXLF
      XLIF=XLIF+DXLF
      XTH=XTH+DXTH
      XMOM=XMOM+DXMOM
      XA=XB $ YA=YB $ ZA=ZB $ PA=PB
      EMA=EMB
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 52
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   52 CONTINUE
   20 CONTINUE
      XLV=XLIF
      XTHV=XTH
      XMOMV=XMOM
      WRITE(6,101)
  101 FORMAT(/)
      XA=XC1 $ YA=YC1 $ ZA=GETZ(XA)
      XLIF=0.
      XTH=0.
      XMOM=0.
      N=(XC3-XC1)/DX+1.
      DXN=(XC3-XC1)/FLOAT(N)
      N=N+1
      CALL PM(DNUC,PA,TA,EMA,GA,UA,RHA)
      THA=THC1
```

```
      CALL FIX(XA,YA,.7854,0.,XA,YA,0.,0.,XA,YA,THA,XMUA,EMA,GA,PA,TA,
     1RHA,UA,DNUC)
      PA=PA-PINF
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 53
      WRITE(6,111)
  111 FORMAT(///50X*COWL*                    /7X*X*12X*Y*12X*Z*9X*P/PINF*
     18X*MACH*9X*LIFT*8X*THRUST*7X*MOMENT*)
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   53 CONTINUE
      DO 30 I=2,N
      XB=XA+DXN
      YB=YC1+(XB-XC1)/(XC3-XC1)*(YC3-YC1)
      ZB=GETZ(XB)
      THB=ATAN((YC3-YC1)/(XC3-XC1))
      CALL PM(DNUC,PB,TB,EMB,GB,UB,RHB)
      CALL FIX(XB,YB,.7854,0.,XB,YB,0.,0.,XB,YB,THB,XMUB,EMB,GB,PB,TB,
     1RHB,UB,DNUC)
      PB=PB-PINF
      DUM=(PA+PB)*(ZB+ZA)/4.
      DXLF=-DUM*(XB-XA)
      DXTH=DUM*ABS(YB-YA)
      YBB=(YA+YB)/2.-YSHF
      XBB=-((XA+XB)/2.-XSHF)
      DXMOM=-YBB*DXTH+XBB*DXLF
      XLIF=XLIF+DXLF
      XTH=XTH+DXTH
      XMOM=XMOM+DXMOM
      XA=XB $ YA=YB $ ZA=ZB $ PA=PB
      EMA=EMB
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 54
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   54 CONTINUE
   30 CONTINUE
      N=(XC2-XC3)/DX+1.
      IF(XC2.EQ.XC3) GO TO 60
      DXN=(XC2-XC3)/FLOAT(N)
      N=N+1
      DO 40 I=2,N
      XB=XA+DXN
      YB=YC3+(XB-XC3)/(XC2-XC3)*(YC2-YC3)
      ZB=GETZ(XB)
      RAT=(XB-XC3)/(XC2-XC3)
      DNUB=DNUC+RAT*2.*TAUC
      RATA=((XC3-XV1)/(XB-XV1)-1.)/((XC3-XV1)/(XC2-XV1)-1.)
      TAUY=ATAN(RATA*TAN(TAUC))
      DNUB=DNUC+2.*TAUY
      CALL PM(DNUB,PB,TB,EMB,GB,UB,RHB)
      THB=ATAN((YC2-YC3)/(XC2-XC3))
      CALL FIX(XB,YB,.7854,0.,XB,YB,0.,0.,XB,YB,THB,XMUB,EMB,GB,PB,TB,
     1RHB,UB,DNUB)
      PB=PB-PINF
      DUM=(PA+PB)*(ZB+ZA)/4.
```

```
      DXLF=-DUM*(XB-XA)
      DXTH=DUM*ABS(YB-YA)
      YBB=(YA+YB)/2.-YSHF
      XBB=-((XA+XB)/2.-XSHF)
      DXMOM=-YBB*DXTH+XBB*DXLF
      XLIF=XLIF+DXLF
      XTH=XTH+DXTH
      XMOM=XMOM+DXMOM
      XA=XB $ YA=YB $ ZA=ZB $ PA=PB
      EMA=EMB
      PX=PA+PINF
      PX=PX/PINF
      IF(IPR.EQ.1) GO TO 55
      WRITE(6,100) XA,YA,ZA,PX,EMA,XLIF,XTH,XMOM
   55 CONTINUE
   40 CONTINUE
   60 CONTINUE
      XTH=XTH+XTHV
      XLIF=XLIF+XLV
      XMOM=XMOM+XMOMV
      WRITE(6,4203) XLIF
 4203 FORMAT(///20X*TOTAL LIFT =*E12.4)
      WRITE(6,1008) XTH
 1008 FORMAT(/20X*THRUST (FROM COWL AND VEHICLE UNDERSURFACE) =*E12.4)
      WRITE(6,4206) XMOM
 4206 FORMAT(/20X*MOMENT (FROM COWL AND VEHICLE UNDERSURFACE) =*E12.4)
      IF(THX.LT.1.E+09) GO TO 4201
      WRITE(6,4200)
 4200 FORMAT(/20X*EXIT PROFILE AND THRUST AND MOMENT (FROM SIDE WALLS) N
     1OT CALCULATED.*)
      GO TO 4212
 4201 IF(THX.GT.0.) GO TO 4205
      WRITE(6,4204) THX
 4204 FORMAT(/20X*THRUST (FROM SIDE WALLS) =*E12.4)
      PMOM1=0.
      WRITE(6,4211) PMOM1
 4211 FORMAT(/20X*MOMENT (FROM SIDE WALLS) =*E12.4)
      GO TO 4212
 4205 DTHQ=THX-XTH
      WRITE(6,4204) DTHQ
      XTH=THX
      PMOM1=PMOM-XMOM
      WRITE(6,4211) PMOM1
 4212 CONTINUE
      PTHL=(THRMAX-XTH)/THRMAX*100.
      WRITE(6,1563) PTHL
 1563 FORMAT(/40X*PERCENT THRUST LOSS =*F6.3)
      RETURN
      END
```

```
      SUBROUTINE XTH(X2,Y2,TH2,DNUE,PINF,THS2,
     1THX1,      XMASS1,THX)
      COMMON/D3/AZ,BZ,CZ
      COMMON/SHF/XSHF,YSHF
      COMMON/PMPM/ PMOM
      COMMON/DNU3D/DNU3D
      DIMENSION XNU(50),X(50),Y(50),Z(50),EM(50),P(50),TH(50),T(50)
      DIMENSION DNUE(10),X2(10),Y2(10),TH2(10),RHO(50),U(50),G(50)
      GETZ(X5)=AZ*X5*X5+BZ*X5+CZ
      THX=0.
      IF(AZ+BZ.EQ.0.) RETURN
      INDEX=0
      DO 10 I=1,5
      J1=(I-1)*5+1
      J2=I*5
      IF(I.EQ.5) J2=26
      DO 20 J=J1,J2
      RAT=FLOAT(J-J1)/FLOAT(J2-J1)
      XNU(J)=DNUE(I)+RAT*(DNUE(I+1)-DNUE(I))
      X(J)=X2(I)+RAT*(X2(I+1)-X2(I))
      Y(J)=Y2(I)+RAT*(Y2(I+1)-Y2(I))
      TH(J)=TH2(I)+RAT*(TH2(I+1)-TH2(I))
      CALL PM(XNU(J),P(J),T(J),EM(J),G(J),U(J),RHO(J))
      CALL FIX(X(J),Y(J),.7854,0.,X(J),Y(J),0.,0.,X(J),Y(J),TH(J),XMU,
     1EM(J),G(J),P(J),T(J),RHO(J),U(J),XNU(J))
      XNU(J)=DNU3D
      Z(J)=GETZ(X(J))
   20 CONTINUE
   10 CONTINUE
   86 CONTINUE
      I=1
      F1=RHO(I)*U(I)*SIN(THS2-TH(I))/SIN(THS2)*Z(I)
      F1=(RHO(I)*U(I)*U(I)*SIN(THS2-TH(I))*COS(TH(I))+(P(I)-PINF)*SIN(TH
     1S2))*Z(I)/SIN(THS2)
      W1=(RHO(I)*U(I)*U(I)*SIN(THS2-TH(I))*SIN(TH(I))+(P(I)-PINF)*COS(TH
     1S2))*Z(I)/SIN(THS2)
      XMASS2=0.
      THX2=0.
      PMOM=0.
      DO 150 I=2,26
      E2=RHO(I)*U(I)*SIN(THS2-TH(I))/SIN(THS2)*Z(I)
      F2=(RHO(I)*U(I)*U(I)*SIN(THS2-TH(I))*COS(TH(I))+(P(I)-PINF)*SIN(TH
     1S2))*Z(I)/SIN(THS2)
      W2=(RHO(I)*U(I)*U(I)*SIN(THS2-TH(I))*SIN(TH(I))+(P(I)-PINF)*COS(TH
     1S2))*Z(I)/SIN(THS2)
      THX2=THX2+.5*(F1+F2)*(Y  (I)-Y  (I-1))
      XMASS2=XMASS2+.5*(E1+E2)*(Y  (I)-Y  (I-1))
      DUM1=.5*(F1*(Y(I-1)-YSHF)+F2*(Y(I)-YSHF))
      DUM2=.5*(W1*(X(I-1)-XSHF)+W2*(X(I)-XSHF))
      PMOM=PMOM+(DUM1-DUM2)*(Y(I)-Y(I-1))
      E1=E2 $ F1=F2 $ W1=W2
  150 CONTINUE
      RATM=XMASS2/XMASS1
      EMS=RATM-1
      IF(ABS(EMS).LT.1.E-03)GO TO 4625
      INDEX=INDEX+1
```

```
      IF(INDEX.GT.1)GO TO 4625
      DO 170 I=1,26
      FM1=FM(EM(I),G(I))
      FM2=FM1/RATM
      ITM=1
      EMT=RATM*EM (I)
  197 CONTINUE
      FMT=FM(EMT,G(I))
      ERM=(FM2-FMT)/FM2
      IF(ABS(ERM).LT.1.E-03)GO TO 171
      DUMD=1./RATM
      CALL ERROR(6,ITM,EMT,ERM,DUMD,FMT1,ERM1)
      GO TO 197
  171 CONTINUE
      DUM=SQRT((G(I)+1.)/(G(I)-1.))
      ETSQ=SQRT(EMT*EMT-1.)
      ESQ=SQRT(EM(I)*EM(I)-1.)
      XNU2=DUM*ATAN(ETSQ/DUM)-ATAN(ETSQ)
      XNU1=DUM*ATAN(ESQ/DUM)-ATAN(ESQ)
      XNU(I)=XNU(I)+XNU2-XNU1
      CALL PM(XNU(I),P(I),T(I),EM(I),G(I),U(I),RHO(I))
  170 CONTINUE
      GO TO 86
 4625 CONTINUE
      THX=THX2/RATM-THX1
      RETURN
      END
```

```
      SUBROUTINE FIX(X1,Y1,TH1,XMU1,X2,Y2,TH2,XMU2,X3,Y3,TH3,XMU3,EM3,G3
     1,P3,T3,RH3,U3,DNU3)
      COMMON/D3/ AZ,BZ,CZ
      COMMON/DNU3D/DNU3D
      ITM=0
   99 CONTINUE
      SL1=TAN(TH1+XMU1)
      SL2=TAN(TH2-XMU2)
      CALL GEM(X1,Y1,SL1,X2,Y2,SL2,X3,Y3)
      IF(X3.LT.0.) WRITE(6,100)
  100 FORMAT(* NO X3 POINT IN FIX*)
      IF(X3.LT.0.) STOP
      RQ2D=RH3*U3
  200 CONTINUE
      IF(X3.LT.0.) GO TO 99
      Z=AZ*X3*X3+BZ*X3+CZ
      RQ3D=RQ2D/Z
      DNU3D=DNU3
      ITQ=1
      ERT=.001
   10 CONTINUE
      CALL PM(DNU3D,P3,T3,EM3,G3,U3,RH3)
      RQ3T=RH3*U3
      ERQ=(RQ3T-RQ3D)/RQ3D
      IF(ITQ.GE.10) ERT=.005
      IF(ABS(ERQ).LT.ERT) GO TO 20
      CALL ERROR(20,ITQ,DNU3D,ERQ,1.,1,DNU3D1,ERQ1)
      GO TO 10
   20 CONTINUE
      XMU3=ASIN(1./EM3)
      SL13=SL1/2.+.5*TAN(TH3+XMU3)
      IF(XMU1.EQ.0.)SL13=SL1
      SL23=SL2/2.+.5*TAN(TH3-XMU3)
      IF(XMU2.EQ.0.)SL 23=SL2
      CALL GEM(X1,Y1,SL13,X2,Y2,SL23,X3,Y3)
      IF(ITM.EQ.1) GO TO 300
      ITM=1
      GO TO 200
  300 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE THM(RHI,UI,PI,PINF,YV1,YC1,DYV,XV2,THRMAX,DNUV,DNUC,XC2
     1)
      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,EM2,G2,U2,RH2)
      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
```

```
      FUNCTION FM(XM,G)
      FM=XM/(1.+(G-1.)/2.*XM**2)**((G+1.)/2./(G-1.))
      RETURN
      END
```

```
SUBROUTINE GEM(XA,YA,SLA,XB,YB,SLB,XC,YC)
XC=(YB-YA+SLA*XA-SLB*XB)/(SLA-SLB)
YC=YA+SLA*(XC-YA)
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 POLY(XV1,YV1,THI,XV3,THV3,XV2,THV2,
     1XC1,YC1,XC2,THC2,XI)
      COMMON/X/A11,A12,A13,A21,A22,A23,A31,A32,A33
      COMMON/POL/ IPR,IPOLY
      DIMENSION X(10),Y(10),TH(10)
      IF(IPOLY.EQ.1) GO TO 20
      IT=0
      X(1)=XV1
      X(2)=XV1+.01*(XV2-XV1)
      X(3)=XV3
      X(4)=XV2
      TH(1)=THI
      TH(2)=THV3
      TH(3)=THV3
      TH(4)=THV2
      TH(1)=-TH(1) $ TH(2)=-TH(2) $ TH(3)=-TH(3) $ TH(4)=-TH(4)
      Y(1)=YC1
      N=3
      WRITE(6,100)
  100 FORMAT(///30X*VEHICLE COORDINATES*/
     115X*X*4X*TO*4X*X*15X*COORDINATES*)
   30 DO 10 I=1,N
      X2=X(I+1)-XI
      X1=X(I)-XI
      A=(TAN(TH(I+1))-TAN(TH(I)))/2./(X2-X1)
      B=TAN(TH(I+1))-2.*A*X2
      C=Y(I)-B*X1-A*X1*X1
      Y(I+1)=A*X2*X2+B*X2+C
      WRITE(6,69) X(I),X(I+1),A,B,C
   69 FORMAT(10X5E11.3)
   10 CONTINUE
      IF(IT.EQ.1)GO TO 20
      IT=1
      X(1)=XC1$X(2)=X(1)+.01*(XV2-XV1)
      X(3)=XC2
      TH(2)=THC2
      TH(3)=TH(2)
      TH(2)=-TH(2) $ TH(3)=-TH(3)
      Y(1)=YV1
      N=2
      WRITE(6,101)
  101 FORMAT(/30X*COWL COORDINATES*/
     115X*X*4X*TO*4X*X*15X*COORDINATES*)
      GO TO 30
   20 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PM(DNU,P2,T2,EM2,G2,U2,RH2)
      COMMON/A/ PI,THI,EMI,GI,TI,HI,RHI,UI
      COMMON/D/ PHI1
      DTH=2./57.3
      IFAN=DNU/DTH
      IF(DTH.GE.DNU)DTH=DNU
      IF(DTH.EQ.DNU)IFAN=1
      P1=PI
      T1=TI
      EM1=EMI
      H1=HI
      G1=GI
      RH1=RHI
      A1=SQRT(G1*P1/RH1)
      U1=UI
      U1=EM1*A1
      U1=U1*U1
      HT=H1+U1/2.
      P1=ALOG(P1)
      XNU=0.
      XNU1=0.
      DO 10 I=1,IFAN
      IF(I.EQ.IFAN)DTH=DNU-XNU
      XMU1=ASIN(1./EM1)
      B1=G1*EM1/COS(XMU1)
      P2=-B1*DTH+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=HT-U2/2.
      T2=FT(P2P,PHI1,H2)
      G2=FGAM(T2,P2P,PHI1)
      A2=G2*P2P/RH2
      EM2=SQRT(U2/A2)
      XNU=XNU1+DTH
      XNU1=XNU
      P1=P2
      T1=T2
      G1=G2
      EM1=EM2
      U1=U2
      RH1=RH2
   10 CONTINUE
      P2=EXP(P2)
      U2=SQRT(U2)
      RETURN
      END
```

```
      FUNCTION FI(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.
      79=.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
 1000 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 34
      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 FI*/* P1 = *F13.5,5X,*H1 = *E13.5,5X,*T = *E13.5
     1)
      STOP
  830 IF(IFLAG.EQ.1) GO TO 504
      IFLAG=1
      T0=2000.
      T=2000.
      IF(F.LT.0.) GO TO 400
      GO TO 120
  504 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)
  505 E0=E1
      T0=T
      T=T9
      IF(F.LT.0.) GO TO 400
  120 A=1.E-07*(-.1042*F2  +.8242*F+.987)
      B=.001*(.01167*F2 +.1503*F+.938)
      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*(-.1067*F2 +1.11*F+.176)
      C=-.0233*F2 +3.974*F-2.898
  190 IF(T.LE.2000.) GO TO 290
```

```
      A=.000001*(1.792*F2 +.3983*F+.31)
      B=.001*(-9.05*F2 -.07917*F+.245)
      C=10.80*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)
      C=27.05*F2 -73.73*F+58.39
  290 H1=A*T*T+B*T+C
      IF(T.LE.2000.) GO TO 370
      H1=H1*(1.+(1.+F)*(T/2000.-1.)**9)
  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=3.1682134E-05
      A4=-2.9715432E-08
      A5=9.5103580E-12
      A6=-1.0186032E+04
      GO TO 460
  450 CONTINUE
      A1=1.1202436
      A2=1.3905716E-02
      A3=2.6568374E-06
      A4=-1.1560272E-08
      A5=5.2386929E-12
      A6=5.3328896E+03
      XMM1=28.054
  460 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
  340 T0=T
      FT=T*1.0
      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=FNN-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-.0645
  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
  520 G=G+XM*(ALOG(P)/2.3-5.)+XN*(F-1.)
      GO TO 530
  550 T3=T2*T
      T4=T3*T
      CP=A1+A2*T+A3*T2+A4*T3+A5*T4
      G=CP*(CP-1.)
  530 CONTINUE
      FGAM=G
      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 -5.46*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*(1.702*F2 +.3963*F+.31)
      B=.001*(-9.05*F2 -.07917*F+.245)
      C=10.86*F2 -.1183*F+.97
      GO TO 290
  192 A=.000001*(4.01*F2 -13.9*F+11.09)
      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
```