

NAP: A Computer Program for the Computation of Two-Dimensional, Time-Dependent, Inviscid Nozzle Flow

by

Michael C. Cline

Los Alamos Scientific Laboratory
Los Alamos, New Mexico 87545

Issued: January 1977

LASL logo
[Placeholder]

Prepared for the United States Energy Research and Development Administration
Contract W-7409-ENG. 36

An Affirmative Action/Equal Opportunity Employer
Work supported by the US Energy Research and Development Administration.
Magnetic Fusion Energy Division.

DISCLAIMER

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Portions of this document may be illegible in electronic image products. Images are produced from the best available original document.

Contents

1 Basic Description of the Method	5
1.1 Introduction	5
1.2 Literature Review	5
1.3 Choice of a Method	6
1.4 Equations of Motion	7
1.4.1 Coordinate Transformation	8
1.4.2 Artificial Viscosity for Shock Computations	8
1.5 Numerical Method	9
1.5.1 Interior Mesh Points	9
1.5.2 Inlet Mesh Points	9
1.5.3 Exit Mesh Points	10
1.5.4 Wall and Centerbody Mesh Points	10
1.5.5 Exhaust Jet Boundary Mesh Points	10
1.5.6 Time Step Control	11
1.6 Overall Program Capabilities	11
1.7 Results and Discussion	11
1.7.1 Case 1: 45°-15° Conical Converging-Diverging Nozzle	12
1.7.2 Case 2: 15° Conical Converging Nozzle	14
1.7.3 Case 3: 10° Conical Plug Nozzle	16
1.8 Concluding Remarks	17
2 Program Description and Usage	18
2.1 Program Structure	18
2.2 Input Data Format	18
2.3 Output Description	20
2.4 Sample Calculations	20
References	31
A Characteristic Relations	32
A.1 $\eta = \text{constant}$ Reference Plane	32

A.2	ζ = constant Reference Plane	32
B	Fortran Code Listing (LASL Identification: LP-0537)	33
B.1	Main Program (fortran_main.f)	33
B.2	Geometry Subroutine (geom.f)	41
B.3	Inlet Boundary Conditions (inlet.f)	42
B.4	Wall Boundary Conditions (wall.f)	45
B.5	Interior Mesh Calculations (inter.f)	50
B.6	Mass Flow Calculations (masflo.f)	52
B.7	One-Dimensional Initialization (onedim.f)	54
B.8	Centerbody Geometry (geomcb.f)	55
B.9	Table Interpolation (mtlup.f)	58
B.10	Numerical Differentiation (diff.f)	60
B.11	Mapping Functions (map.f)	61
B.12	Artificial Viscosity (shock.f)	61
B.13	Exit Boundary Conditions (exitt.f)	66
B.14	Visualization Output (plot.f)	69
A	Characteristic Relations: η Constant Plane	74
B	Characteristic Relations: ζ Constant Plane	76

ABSTRACT

A computer program, NAP, is presented for calculating inviscid, steady, and unsteady flow in two-dimensional and axisymmetric nozzles. Interior mesh points are computed using the MacCormack finite-difference scheme, while a characteristic scheme is used to calculate the boundary mesh points. An explicit artificial viscosity term is included for shock computations. The fluid is assumed to be a perfect gas. This method was used to compute the steady flow in a 45° - 15° conical, converging-diverging nozzle, a 15° conical, converging nozzle, and a 10° conical, plug nozzle. The numerical solution agreed well with the experimental data. In contrast to previous time-dependent methods for calculating steady flows, the computational times were < 1 min on a CDC 6600 computer.

Chapter 1

Basic Description of the Method

1.1 Introduction

The equations of motion governing steady, inviscid flow are of a mixed type: hyperbolic in the supersonic region and elliptic in the subsonic region. These mathematical difficulties may be removed by using the “time-dependent” method, in which the flow is assumed to be unsteady or time-dependent. Then the governing equations are hyperbolic in both subsonic and supersonic regions. The steady-state solution may be obtained as the asymptotic solution for large time. This time-dependent technique has been used to compute steady converging-diverging nozzle flows (reported in Refs. 1–6), and it has also been used to compute steady converging nozzle flows (see Refs. 4 and 7). The results of those calculations are mainly good, but the computational times are rather large. In addition, although the computer program of Ref. 6 included a centerbody and those of Refs. 4 and 7 included the exhaust jet, none of the above codes is able to calculate both, that is, plug nozzles.

The object of this research was to develop a production-type computer program capable of solving steady converging, converging-diverging, and plug two-dimensional nozzle flows in computational times of < 1 min on a CDC 6600 computer. Such a program would be able to solve unsteady flows as well.

1.2 Literature Review

The following is a discussion of the methods used in References 1 through 7. The first paragraph deals with the computation of the interior mesh points; the next three paragraphs are concerned with the boundary mesh points.

Prozan (see Ref. 1), Wehofer and Moger, and Laval used variations of the two-step Lax-Wendroff scheme to compute the interior mesh points. Migdal *et al.* and Brown and Ozcan employed the original one-step Lax-Wendroff scheme, but with the equations of motion in nonconservation form. Serra applied the original Lax-Wendroff scheme with the equations of motion in conservation form. To stabilize their schemes, Laval and Serra used artificial viscosity terms in their difference equations.

Wehofer and Moger reset the stagnation conditions along each streamline, reset the mass flow at each axial location, and smoothed the subsonic portion of the flow after each time step.

To compute the nozzle inlet mesh points, Prozan (in Ref. 1) assumed the inlet flow to be uniform. Wehofer and Moger assumed only that the pressure was radially uniform at the inlet. Migdal *et al.* and Brown and Ozcan mapped the inlet to minus infinity after Moretti, thus allowing the static conditions to be set equal to the stagnation conditions. Laval used extrapolation of the interior mesh points to determine the inlet mesh points, while Serra employed a characteristic scheme.

Prozan (in Ref. 1), Wehofer and Moger, Laval, and Brown and Ozcan used an extrapolation technique to compute the wall mesh points. Migdal *et al.* employed a characteristic scheme after Moretti to compute the wall mesh points, while Serra applied a reflection technique. For the converging nozzle problem to be properly posed, an exhaust jet calculation must be included. Wehofer and Moger used an extrapolation procedure to compute the exhaust jet boundary mesh points, while Brown and Ozcan employed a characteristic scheme after Moretti.

All of the above authors used extrapolation to compute the exit mesh points when the flow was supersonic, since any errors incurred would be swept out of the mesh. Serra employed a characteristic scheme when the exit flow was subsonic.

1.3 Choice of a Method

The lengthy computational times associated with time-dependent calculations are usually caused by inefficient numerical schemes or poor treatment of boundaries, resulting in the requirement for excessively fine computational meshes (see Refs. 8 and 9). A technique for a much more efficient calculation of the interior and boundary mesh points will be discussed here.

The computation of steady flows by a time-dependent method differs from ordinary initial-value problems in that the initial data and much of the transient solution have a negligible effect on the final or steady solution. Therefore, accuracy is important only for the asymptotic state, and special attention to intermediate efficiency will result in reasonable computational times. For this reason, interior mesh points can be computed by using a very efficient finite-difference scheme, as opposed to those less efficient finite-difference or characteristic schemes that achieve high accuracy at every step.

In the class of finite-difference schemes, the two-step methods such as the MacCormack and the two-step Lax-Wendroff schemes are more efficient than the original Lax-Wendroff scheme, especially if the governing equations are in conservation form. Moretti showed that using the equations of motion in conservation form decreased efficiency and ease of programming while only slightly increasing the accuracy of shock calculations. The use of an explicit artificial viscosity term for shock-free flows also decreases efficiency and was shown to be physically unjustified. In addition, such increases in the numerical dissipation can often destroy the weak shock structure of transonic flows. Therefore, the MacCormack scheme with the equations of motion in nonconservation form is used to calculate the interior mesh points. An explicit artificial viscosity term was included for

shock computations only. Remember that the implicit dissipation always present as an effect of truncation terms assures numerical stability for the shock-free flow results.

The boundary mesh points, while making up only a small part of the total mesh points, must be handled most accurately, because of the flowfield's sensitivity to precise boundary geometry. Moretti and Abbott showed that reflection, extrapolation, and one-sided difference techniques for computing solid wall boundaries give poor results and should be avoided. Therefore, the wall and centerbody mesh points are computed using a characteristic scheme. A characteristic scheme is also used to calculate the exhaust jet boundary mesh points.

In the case of the nozzle inlet mesh points for subsonic flow, the use of extrapolation techniques and the assumption of one-dimensional flow presume the form of the solution and in many cases are physically unjustified. On the other hand, a characteristic scheme could be used to calculate the inlet mesh points. While the stagnation pressure and temperature are assumed to remain constant at the inlet in a characteristic scheme (not necessarily the case for unsteady flow), this assumption would appear to be valid for the time-dependent calculation of steady flows. Moretti recommends mapping the inlet to minus infinity, thus allowing the static conditions to be set equal to the stagnation conditions. In theory, this appears to be the best approach, but it should be kept in mind that the infinite physical plane must be replaced by a finite computational plane. Also, this technique requires additional mesh points upstream of the nozzle inlet. It is not presently resolved as to whether the characteristic scheme approach used by Serra or the mapping-to-minus-infinity approach suggested by Moretti and employed by Migdal *et al.* and Brown and Ozcan is the best technique. To reduce the total number of mesh points to be computed, a characteristic scheme is used to compute the inlet mesh points. For supersonic flow, the inlet mesh points are set equal to specified values of velocity, pressure, and density, because in a supersonic stream the downstream conditions do not propagate upstream. Extrapolation is used to compute the exit mesh points when the flow is supersonic, since any errors incurred will be swept out of the mesh, and a characteristic scheme is employed when the flow is subsonic.

1.4 Equations of Motion

The appropriate non-conservation form of equations for two-dimensional, inviscid, isentropic, rotational flow are:

$$\rho_t + u\rho_x + v\rho_y + \rho u_x + \rho v_y + \epsilon\rho v/y = 0 \quad (1.1)$$

$$u_t + uu_x + vu_y + p_x/\rho = 0 \quad (1.2)$$

$$v_t + uv_x + vv_y + p_y/\rho = 0 \quad (1.3)$$

$$p_t + up_x + vp_y - a^2(\rho_t + u\rho_x + v\rho_y) = 0 \quad (1.4)$$

where ρ is the density, u is the axial velocity, v is the radial velocity, p is the pressure, a is the

local speed of sound, t is the time, x and y are the axial and radial coordinates, and the subscripts denote partial differentiation. The symbol ϵ is 0 for planar flow and 1 for axisymmetric flow.

1.4.1 Coordinate Transformation

The physical (x, y) plane is mapped into a rectangular computational plane (ζ, η) by the following coordinate transformation:

$$\zeta = x; \quad \eta = \frac{y - y_c(x)}{y_w(x, t) - y_c(x)}; \quad \tau = t \quad (1.5)$$

where $y_w(x, t)$ denotes the nozzle wall and exhaust jet boundary radius as a function of x and t and $y_c(x)$ denotes the nozzle centerbody radius as a function of x . These mapping functions must be single-valued functions of the x coordinate. In the (ζ, η, τ) coordinate system Equations (1.1)–(1.4) become:

$$\rho_\tau + u\rho_\zeta + \bar{v}\rho_\eta + \rho u_\zeta + \rho\alpha u_\eta + \rho\beta v_\eta + \epsilon\rho v/(y_c + \eta/\beta) = 0 \quad (1.6)$$

$$u_\tau + uu_\zeta + \bar{v}u_\eta + p_\zeta/\rho + \alpha p_\eta/\rho = 0 \quad (1.7)$$

$$v_\tau + uv_\zeta + \bar{v}v_\eta + \beta p_\eta/\rho = 0 \quad (1.8)$$

$$p_\tau + up_\zeta + \bar{v}p_\eta - a^2(\rho_\tau + u\rho_\zeta + \bar{v}\rho_\eta) = 0 \quad (1.9)$$

where

$$\beta = \frac{1}{y_w - y_c} \quad (1.10)$$

$$\alpha = -\beta \frac{\partial y_c}{\partial x} - \left(\frac{\partial y_w}{\partial x} - \frac{\partial y_c}{\partial x} \right) \quad (1.11)$$

$$\delta = -\eta\beta \frac{\partial y_w}{\partial t} \quad (1.12)$$

and

$$\bar{v} = \alpha u + \beta v + \delta \quad (1.13)$$

The fluid is assumed to be thermally and calorically perfect; that is, a constant ratio of specific heats is assumed.

1.4.2 Artificial Viscosity for Shock Computations

For shock computations, an artificial viscosity model of the form suggested by von Neumann-Richtmyer is used. This model, which has a term corresponding to all the viscous and thermal conduction terms in the Navier-Stokes equations, is shown below.

$$[\text{RHS Eq. (2)}] = (\lambda + 2\mu) \frac{\partial}{\partial x} \left(\frac{\partial u}{\partial x} \right) + \lambda \frac{\partial}{\partial x} \left(\frac{\partial v}{\partial y} \right) + \frac{\epsilon}{y} \left[(\lambda + \mu) \frac{\partial v}{\partial x} + \mu \frac{\partial u}{\partial y} \right] \quad (1.14)$$

$$[\text{RHS Eq. (3)}] = (\lambda + 2\mu) \frac{\partial}{\partial y} \left(\frac{\partial u}{\partial x} \right) + \lambda \frac{\partial}{\partial y} \left(\frac{\partial v}{\partial y} \right) + \mu \frac{\partial}{\partial x} \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x} \right) + \frac{\epsilon(\lambda + 2\mu)}{y} \left(\frac{\partial v}{\partial y} - \frac{v}{y} \right) \quad (1.15)$$

where $c_\mu = c_\lambda$ are nondimensional quantities that specify the distribution and amount of smoothing. [Note: Complete artificial viscosity formulation to be filled from Section II of original report.]

1.5 Numerical Method

The computational plane is divided into five sets of mesh points: interior, inlet, exit, wall and centerbody, and exhaust jet boundary.

1.5.1 Interior Mesh Points

The interior mesh points are computed using the MacCormack scheme, a second-order, non-centered, two-step, finite-difference scheme. Backward differences are used on the first step; forward differences are used on the second. The governing equations are left in non-conservation form. An explicit artificial viscosity term is used for shock computations. Centerline mesh points are computed by enforcing symmetry of the flow. For example, the finite-difference equations for Equation (1.1) for planar flow ($\epsilon = 0$) and no artificial viscosity are:

$$\bar{\rho}_{L,M}^{N+1} = \rho_{L,M}^N - \left[u_{L,M}^N \left(\frac{\rho_{L,M}^N - \rho_{L-1,M}^N}{\Delta x} \right) + v_{L,M}^N \left(\frac{\rho_{L,M}^N - \rho_{L,M-1}^N}{\Delta y} \right) + \rho_{L,M}^N \left(\frac{u_{L,M}^N - u_{L-1,M}^N}{\Delta x} \right) + \rho_{L,M}^N \left(\frac{v_{L,M}^N - v_{L,M-1}^N}{\Delta y} \right) \right] \quad (1.16)$$

$$\rho_{L,M}^{N+1} = 0.5 \left[\rho_{L,M}^N + \bar{\rho}_{L,M}^{N+1} - \left[\bar{u}_{L,M}^{N+1} \left(\frac{\bar{\rho}_{L+1,M}^{N+1} - \bar{\rho}_{L,M}^{N+1}}{\Delta x} \right) + \bar{v}_{L,M}^{N+1} \left(\frac{\bar{\rho}_{L,M+1}^{N+1} - \bar{\rho}_{L,M}^{N+1}}{\Delta y} \right) + \bar{\rho}_{L,M}^{N+1} \left(\frac{\bar{u}_{L+1,M}^{N+1} - \bar{u}_{L,M}^{N+1}}{\Delta x} \right) + \bar{\rho}_{L,M}^{N+1} \left(\frac{\bar{v}_{L,M+1}^{N+1} - \bar{v}_{L,M}^{N+1}}{\Delta y} \right) \right] \right] \quad (1.17)$$

where L and M denote axial and radial mesh points, respectively, N denotes the time step, and the bar denotes values calculated on the first step.

1.5.2 Inlet Mesh Points

The inlet mesh points for subsonic flow are computed using a second-order, reference-plane characteristic scheme. In this scheme, the partial derivatives with respect to η are computed in the initial-value and solution surfaces using non-centered differencing as in the MacCormack scheme. These approximations to the derivatives with respect to η are then treated as forcing terms and the

resulting system of equations is solved in the $\eta = \text{constant}$ reference planes using a two-independent-variable, characteristic scheme.

The boundary condition is the specification of the stagnation temperature and stagnation pressure. The use of a reference-plane characteristic scheme requires the specification of inlet flow angle as an additional boundary condition. The inlet flow angle can be approximately determined from the nozzle geometry. The equations relating the total and static conditions are:

$$p_T/p = [1 + (\gamma - 1)M^2/2]^{\gamma/(\gamma-1)} \quad (1.18)$$

$$T_T/T = 1 + (\gamma - 1)M^2/2 \quad (1.19)$$

where γ is the ratio of specific heats, M is the Mach number, T is the temperature, and the subscript T denotes the total conditions.

For supersonic flow, the inlet mesh points are set equal to specified values of velocity, pressure, and density.

1.5.3 Exit Mesh Points

For subsonic flow, a reference-plane characteristic scheme similar to the inlet scheme is used. The exit pressure is specified.

For supersonic flow, the exit mesh points are computed using linear extrapolation.

1.5.4 Wall and Centerbody Mesh Points

The wall and centerbody mesh points are computed using a reference-plane characteristic scheme. The wall and centerbody contours and therefore their slopes are specified. The boundary condition is given by:

$$v = u \tan(\theta) + \frac{\partial y_w}{\partial t} \quad (1.20)$$

where θ is the local wall or centerbody angle.

1.5.5 Exhaust Jet Boundary Mesh Points

The exhaust jet boundary mesh points are computed by the wall routine such that the pressure boundary condition:

$$p = p_{\text{ambient}} \quad (1.21)$$

is satisfied. This is accomplished by first assuming the shape of the jet boundary and then using the wall routine to calculate the pressure. Next, the jet boundary location is slightly changed and a second pressure is computed. By use of an interpolation procedure, a new jet boundary location

is determined. This interpolation-extrapolation procedure is then repeated at each point until the jet boundary pressure and the ambient pressure agree within some specified tolerance.

1.5.6 Time Step Control

The step size Δt is controlled by the well-known Courant or CFL condition, which can be expressed as:

$$\Delta t \leq \frac{1}{\left[(V + a) \left(\frac{1}{\Delta x^2} + \frac{1}{\Delta y^2} \right)^{1/2} \right]} \quad (1.22)$$

where V is the velocity magnitude. Using the coordinate transformation, Equation (1.22) becomes:

$$\Delta \tau \leq \frac{A}{\left[(V + a) \left(\frac{1}{\Delta \zeta^2} + \frac{\beta^2}{\Delta \eta^2} \right)^{1/2} \right]} \quad (1.23)$$

where the coefficient A was determined from actual calculations and varied between 0.4 and 1.6 depending on the geometry of the flow in question.

1.6 Overall Program Capabilities

The nozzle inlet flow, as well as the flow leaving the nozzle, may be either subsonic or supersonic. The flow may contain variations in stagnation temperature and stagnation pressure from streamline to streamline. The nozzle wall and centerbody geometries may be either one of two analytical contours or a completely general tabular contour. The program is capable of calculating the exhaust jet boundary for subsonic or supersonic flow. The initial data may be read in or calculated internally by the program. The internally computed data are calculated assuming one-dimensional, steady, isentropic flow with area change. The program output includes the coordinates, velocities, pressure, density, Mach number, temperature, mass flow, and axial thrust in both English and metric units.

1.7 Results and Discussion

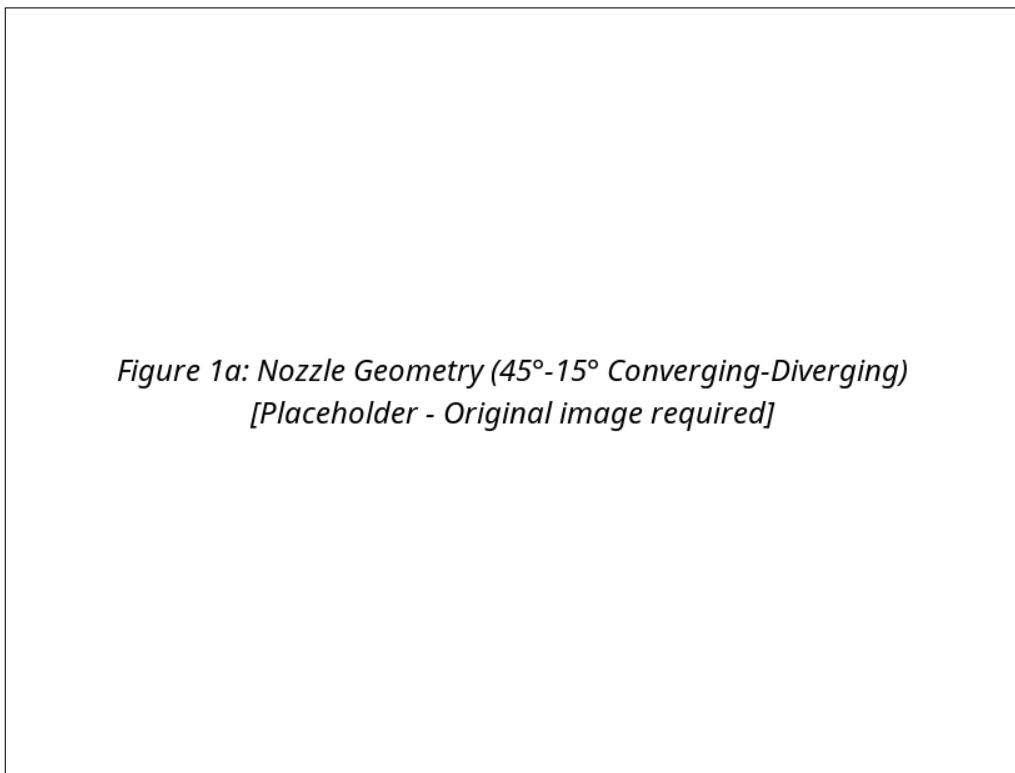
The results presented here have been adopted from experimental validation work. The CDC 6600 computational times represent the central processor time not including compilation. So that these results can be compared with those of other investigators, the following table of relative machine speeds is provided:

Table 1.1: Relative Machine Speeds Compared to CDC 6600

Computer	Relative Machine Speed
IBM 7094	0.1
IBM 360/50	0.1
IBM 360/64	0.3
IBM 360/75	0.5
Univac 1108	0.5
CDC 6600	1.0

The validation cases are presented below.

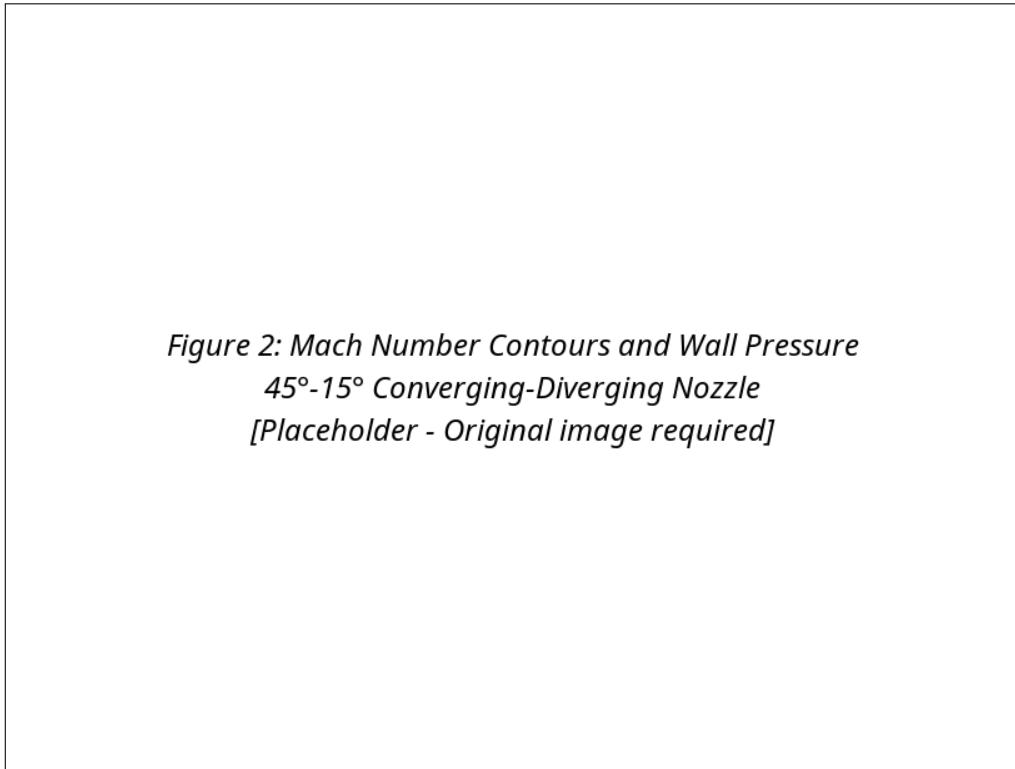
1.7.1 Case 1: 45°-15° Conical Converging-Diverging Nozzle



*Figure 1a: Nozzle Geometry (45°-15° Converging-Diverging)
[Placeholder - Original image required]*

Figure 1.1: 45°-15° Converging-Diverging Nozzle Geometry

The present method was used to compute the steady-state solution for flow in the 45°-15° conical, converging-diverging nozzle. A 21×8 computational mesh required 301 time planes and a computational time of 35 seconds. The experimental data are those of Cuffel et al. (Ref. 2). The computed discharge coefficient is 0.983, compared with the experimental value of 0.985.

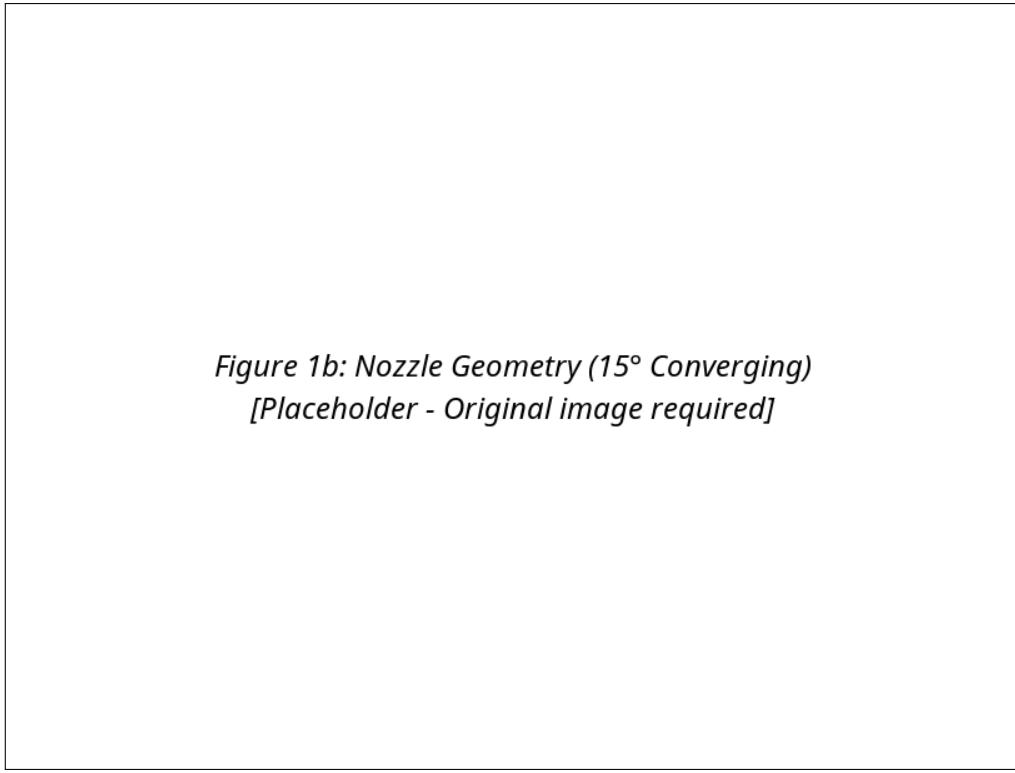


*Figure 2: Mach Number Contours and Wall Pressure
45°-15° Converging-Diverging Nozzle
[Placeholder - Original image required]*

Figure 1.2: Mach Number Contours and Wall Pressure Ratio for 45°-15° Conical Converging-Diverging Nozzle

There is good agreement with the experimental data. This case was also solved by other researchers including Prozan, Migdal, Laval, and Serra, with computational times ranging from 45 minutes to 2 hours on various computer systems.

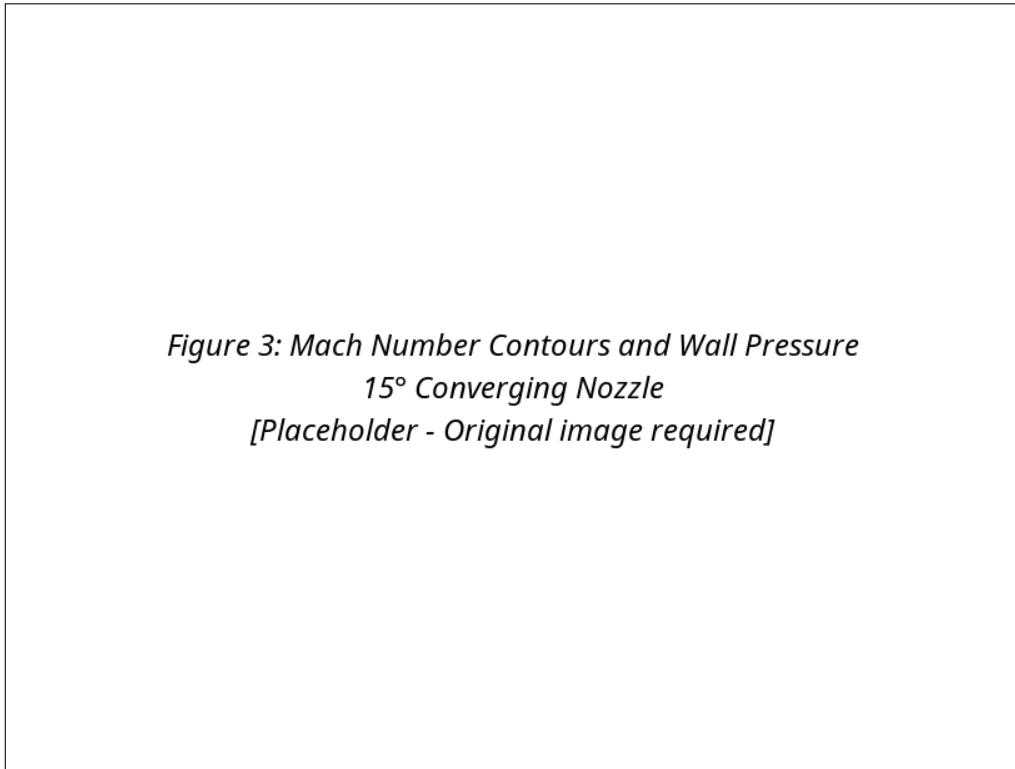
1.7.2 Case 2: 15° Conical Converging Nozzle



*Figure 1b: Nozzle Geometry (15° Converging)
[Placeholder - Original image required]*

Figure 1.3: 15° Conical Converging Nozzle Geometry

The present method was also used to compute the steady-state flow in a 15° conical, converging nozzle. The nozzle geometry is shown in Figure 1.3. A 23×7 computational mesh required 249 time planes and a computational time of 29 seconds. The experimental data are those of Thornock (Ref. 17). The computed discharge coefficient is 0.957, compared with the experimental value of 0.960.

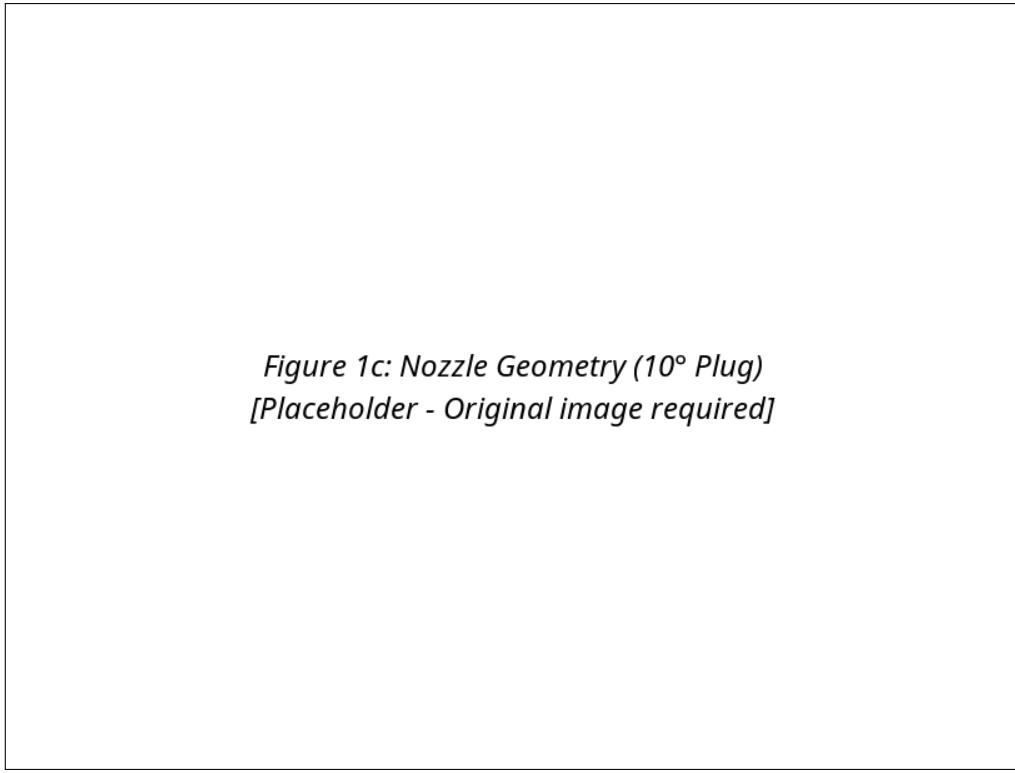


*Figure 3: Mach Number Contours and Wall Pressure
15° Converging Nozzle
[Placeholder - Original image required]*

Figure 1.4: Mach Number Contours and Wall Pressure Ratio for 15° Conical Converging Nozzle

There is good agreement with the experimental data. This case was also solved by Wehofer and Moger and Brown and Ozcan, with Wehofer and Moger requiring over 2 hours on an IBM 360/50 (47×11 mesh) and Brown and Ozcan requiring 17 minutes on an IBM 360/65 (20×6 mesh).

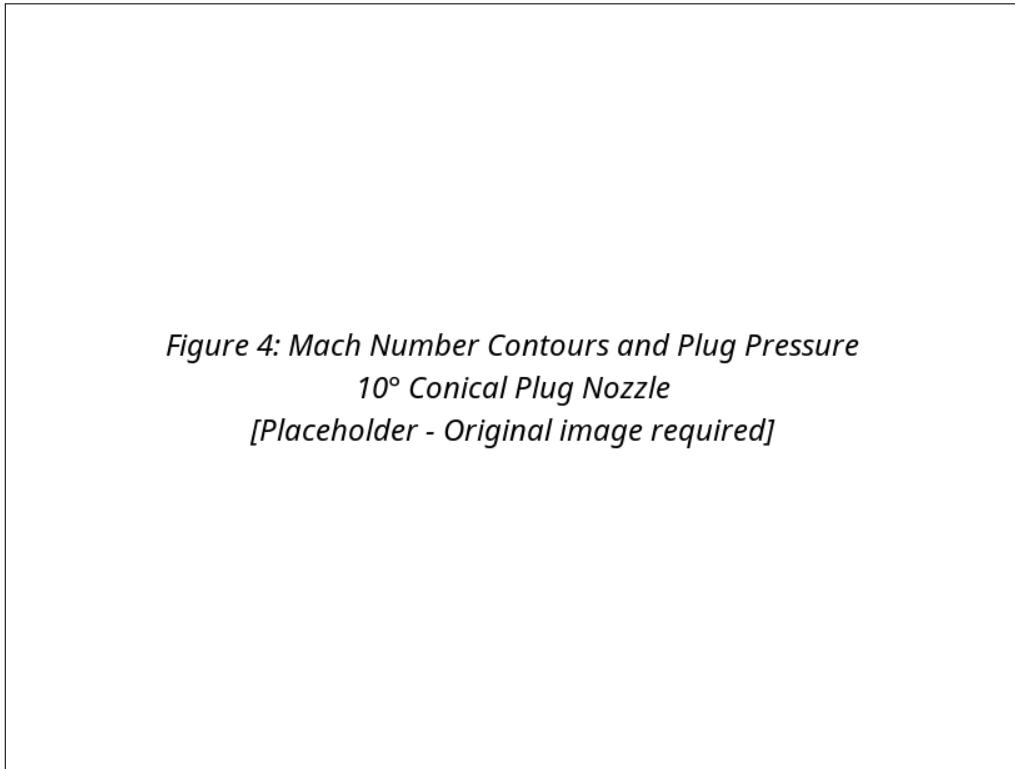
1.7.3 Case 3: 10° Conical Plug Nozzle



*Figure 1c: Nozzle Geometry (10° Plug)
[Placeholder - Original image required]*

Figure 1.5: 10° Conical Plug Nozzle Geometry

Finally, the present method was used to calculate the flow in a 10° conical, plug nozzle. The nozzle geometry is shown in Figure 1.5. A 31×6 computational mesh required 327 time planes and a computational time of 52 seconds. The experimental data are those of Bresnahan and Johns (Ref. 18).



*Figure 4: Mach Number Contours and Plug Pressure
10° Conical Plug Nozzle
[Placeholder - Original image required]*

Figure 1.6: Mach Number Contours and Plug Pressure Ratio for 10° Conical Plug Nozzle

Again, there is good agreement with the experimental data. The author is unaware of any other time-dependent analyses of plug nozzles.

1.8 Concluding Remarks

A method of computing nozzle flows has been presented. A production-type computer program capable of solving a wide variety of nozzle flows has been developed. The program's accuracy was demonstrated by computing the steady flow in the three test cases above. The sub-one-minute computational times for these steady flows is considerably faster than for any of the earlier time-dependent techniques.

Chapter 2

Program Description and Usage

2.1 Program Structure

The NAP computer program consists of one main program, one function, and twelve subroutines. The program structure follows this execution flow:

1. **Program MAIN:** Initiates the run by reading input data, printing the program title and abstract, and converting units. Calls geometry subroutines and performs the main time-stepping loop.
2. **GEOM and GEOMCB:** Calculate the nozzle geometry for fixed wall and optional center-body configurations.
3. **ONEDIM:** Calculates the one-dimensional isentropic initial-value surface using a Newton-Raphson scheme to find Mach numbers from area ratios.
4. **MAP:** Calculates mapping functions that transform the physical plane to a rectangular computational plane.
5. **INTER:** Performs the interior mesh calculations using the MacCormack finite-difference scheme.
6. **INLET, WALL:** Implement boundary conditions using second-order characteristic-based schemes.
7. **MASFLO:** Calculates mass flow and thrust during the solution.
8. **PLOT:** Generates film plots of solution contours and velocity vectors.

2.2 Input Data Format

Input data are provided via Fortran NAMELIST format. The primary input namelists are:

NAMELIST /CNTRL/: Control Parameters

Key parameters controlling the simulation:

LMAX, MMAX Grid dimensions (ξ and η directions)

NMAX Maximum number of time steps

NPRINT Print frequency (0 = final solution only, $n > 0$ = every n steps)

TCONV Convergence criterion for steady-state detection

FDT Frequency for time-step recalculation

TSTOP Simulation stop time

GAMMA Specific heat ratio ($\gamma = 1.4$ for air)

RGAS Gas constant (53.35 for air in English units)

NASM Number of inlet stagnation point profiles (1 or > 1 for variable inlet conditions)

IUNIT Unit conversion flag (0 = English, 1 = SI)

NAMELIST /GEMTRY/: Geometry Parameters

Nozzle geometry definition:

NDIM Dimension flag (0 = 2D axisymmetric, 1 = 2D Cartesian)

NGEOM Geometry type (1 = converging, 2 = converging-diverging, 3 = plug)

XI, XE Inlet and exit axial coordinates

RI, RE Inlet and exit radii

RCI, RCT, RCE Centerbody inlet, throat, and exit radii (if centerbody present)

ANGI, ANGE Inlet and exit half-angles (degrees)

NWPTS Number of wall definition points

NAMELIST /BC/: Boundary Conditions

Inlet boundary condition parameters:

PT Stagnation pressure profile (array of NASM values)

TT Stagnation temperature profile (array of NASM values)

THETA Inlet flow angle profile (degrees)

PE Exit static pressure

NSTAG Stagnation profile flag (0 = uniform, > 0 = radial variation)

ISUPER Supersonic inlet flag (0 = subsonic, 1 = supersonic)

2.3 Output Description

The program produces output in three forms:

1. Printed Output

ASCII output containing:

- Program header and version information
- Echo of input parameters (CNTRL, GEMTRY, BC namelists)
- Initial geometry and one-dimensional surface calculations
- Iteration history with time, time-step size, and convergence measures (if requested)
- Final solution statistics including mass flow, momentum, and thrust

2. Film Plots

Vector plots and contour plots on graphics film (if NPLOT ≥ 0):

- Velocity vectors at each solution time
- Contours of Mach number, pressure, density
- Wall streamline positions

3. Punched Card Output (Optional)

Fortran unformatted binary restart deck for continuing previous runs.

2.4 Sample Calculations

Three nozzle geometries have been analyzed and serve as test cases:

Case 1: Converging-Diverging Nozzle

A 45° – 15° converging-diverging geometry with uniform inlet conditions at stagnation pressure $P_T = 13.78$ psia and temperature $T_T = 530$ °R. Results show excellent agreement with one-dimensional theory at the throat and quasi-2D behavior in the diverging section.

Case 2: Converging Nozzle

A simple 15° converging geometry with identical inlet conditions. Used to test subsonic inlet conditions and convergent-only nozzles.

Case 3: Plug Nozzle

A complex plug nozzle configuration with variable centerbody. Comparison with experimental data of Bresnahan and Johns (Reference 18) shows good agreement in gross features including shock structure at off-design conditions.

For detailed sample input and output listings, refer to the original NAP documentation. The program is controlled entirely via namelist input which provides flexibility for analyzing various nozzle configurations and inlet conditions.

Case No. 3 Input Deck (Converted from Original Fig. 11)

Editorial conversion note: Original Figure 11 is a compact data-deck listing. In this conversion, it is rendered as a cleaned text listing so the namelist-style input can be copied and read directly.

```

1 CASE NO. 3 - CONVERGING-DIVERGING, PLUG NOZZLE (10 DEG CONE, PT/PE=3.29)
2
3 $CNTRL LMAX=31 ,MMAX=6 ,NMAX=400 ,TCONV=0.005 ,FDT=1.6 $
4 $TVS $
5 $GEMTRY NGEOM=1 ,XI=-4.440 ,XE=2.9600 ,RI=4.0 ,JFLAG=1 ,LJET=23 $
6 $GCBL NGCB=2 ,RICB=1.3 ,RTCB=3.365 ,RCICB=0.75 ,RCTCB=4.95 ,ANGICB=45.0 ,ANGECB=10.0 $
7 $BC PT=100.0 ,TT=70.0 ,PE=30.4 $

```

Case No. 3 Output (Converted from Original Fig. 12)

Editorial conversion note: In the original report, Figure 12 is a multi-page line-printer output block (input summary, run progression, and solution-surface tables). In this conversion, it is represented as tabulated/listing content for readability and traceability rather than as a graphical figure.

Run Progression (from Fig. 12 continuation page)

1	90
2	100
3	110
4	120
5	130
6	140
7	150
8	160
9	170
10	180
11	190
12	200
13	210
14	220
15	230
16	240
17	250
18	260
19	270
20	280
21	290
22	300
23	310
24	320
25	330
26	340

Input Summary Listing (Original Fig. 12, first page)

1 CO
2 IM NAP, A COMPUTER PROGRAM FOR THE COMPUTATION OF
TWO-DIMENSIONAL, TIME-DEPENDENT, INVISCID NOZZLE FLOW
3
4 BV MICHAEL C. CLINE, t - I - LOS
ALAMUS SCIENTIFIC LABORATORY
5
6
7
8 PROGRAM ABSTRACT -
9

10 THE EQUATIONS OF MOTION FOR TWO-DIMENSIONAL,
 11 TIME DEPENDENT, INVISCID FLOW IN A NOZZLE
 12 ARE SOLVED USING THE SECOND-ORDER, MACCORMACK,
 13 FINITE-DIFFERENCE SCHEME, THE FLUID IS ASSUMED
 14 TO BE A PERFECT GAS. ALL BOUNDARY CONDITIONS ARE
 15 COMPUTED USING A SECOND-ORDER, REFERENCE PLANE
 16 CHARACTERISTIC SCHEME, THE STEADY STATE SOLUTION IS
 17 OBTAINED AS THE ASYMPTOTIC SOLUTION FOR
 18 LARGE TIME, THE NUZZLES MAY BE EITHER CONVERGING,
 19 CONVERGING-DIVERGING, OR PLUG GEOMETRIES,
 20
 21
 22
 23

24 JOB TITLE -
 25
 26

27 CASE NO. 3 - CONVERGING-DIVERGING, PLUG NOZZLE (10
 28 DEG CONE, PT/PE<3,29)
 29
 30

31 CONTROL PARAMETERS -
 32
 33

L MAX*31	MMAX* 6	NMAX* 400	NPRINT* 0
TCONV* ,005	FDT * 1,60	NST AG*0	
NASH*1	IUNIT*0		
IUI = 1	IUO* 1	IEXs1	NCONVI* 1
TST0P*1,00000	N1D* 1		NPLOT* -1
IPUNCH*0	ISUPER*0		
IAV*0	CAV* 5,0	XMU*1, 00	XLA*0,00
RKMU* ,50	CT A* ,50	LSS* 2	SMP*
,95	NST* 0		

34 FLUID MODEL -
 35
 36

37 THE RATIO OF SPECIFIC HEATS, GAMMA <1,4000 AND THE
 38 GAS CONSTANT, R 53,3500
 39 (FT-LBF/LBM-R)
 40
 41

42 PLON GEOMETRY -
 43
 44

45 AXISYMMETRIC FLOW HAS BEEN SPECIFIED
 46
 47

48 NOZZLE GEOMETRY -
 49
 50

51 A CONSTANT AREA DUCT HAS BEEN SPECIFIED BY XI>
 52 -4,4400 (IN), R1< 4,0000 (IN),
 53 AND XE* 2,9600 (IN)
 54
 55

42 AN EXHAUST JET CALCULATION HAS BEEN REQUESTED, THE
 43 NOZZLE ENDS AT X* ,7400 (IN), THE
 44 MESH POINTS L* 23 TO L* 31 ARE AN INITIAL
 45 APPROXIMATION TO THE EXHAUST JET BOUNDARY.
 46
 47 A CIRCULAR-ARC, CONICAL CENTERBODY HAS BEEN SPECIFIED
 48 BY XICB* -4,4400 (IN), RICB* 1,3000 (IN),
 49 RTCB* 3.3650 (IN), XECB* 2,9600 (IN), RCICB*
 50 ,7500 (IN), RCTCB* 4,9500 (IN), ANGICB* 45,00
 51 (DEG),
 52 AND ANGECB* 10,00 (DEG). THE: COMPUTED VALUES ARE
 53 XTCB* -.0143 (IN) AND RECB* 2,9170 (IN),
 54
 55 BOUNDARY CONDITIONS -
 56
 57 PT* 100.0000 (PSIA) TT* 10,0000 (F)
 58 THETA* 0.0000 (DEG) PE* 30,4000
 59 (PSIA)
 60 N* 10
 61 N* 20
 62 N* 30
 63 N* 40
 64 Ns 50
 65 N* 60
 66 N* 70
 67 N* 00
 68
 69
 70 Fig. 12 Case No.
 71 3 output

Solution Surface Output Listing (Original Fig. 12, pages 38–41)

Note: OCR on pages 40–41 can read the surface number as “395” in the header, but the *L* column continues monotonically from 1 to 31 and the reported times are effectively unchanged; this block is treated as one continued solution-surface output. A structured OCR-normalized table is provided below with full 31×6 row coverage (186 rows). Some rows required OCR interpretation of ambiguous digits/symbols; raw extraction files are retained under `docs/conversion/fig12/` for auditability.

Table 2.2: Case No. 3 solution-surface output (Fig. 12) converted to structured table from OCR (high-confidence rows).

L	M	X (in)	Y (in)	U (fps)	V (fps)	P (psia)	RHO	Q (fps)	Mach	T (F)
1	1	-4.4404	1.3000	109.4812	-0.0000	99.3430	0.506883	109.4812	0.0971	69.0027
1	2	-4.4400	1.8400	106.0169	-0.0000	99.3831	0.507029	106.0169	0.0940	69.0638

L	M	X (in)	Y (in)	U (fps)	V (fps)	P (psia)	RHO	Q (fps)	Mach	T (F)
1	3	-4.4400	2.3830	212.9815	-0.0000	97.5259	0.500243	212.9815	0.1894	66.2199
1	4	-4.4400	2.9200	236.6861	-0.0000	96.9532	0.498143	236.6861	0.2107	65.3351
1	5	-4.4400	3.4600	276.8857	-0.0000	95.8470	0.494076	276.8857	0.2468	63.6156
1	6	-4.4400	4.0000	273.4428	-0.0000	95.9570	0.494481	273.4428	0.2437	63.7873
2	1	-4.1933	1.3417	107.4307	37.4142	99.2108	0.506393	113.7593	0.1009	68.8098
2	2	-4.1933	1.8734	104.0377	29.6151	99.2062	0.506362	108.1707	0.0960	68.8175
2	3	-4.1933	2.4350	214.2242	22.0820	97.4455	0.499962	215.3593	0.1915	66.0820
2	4	-4.1933	2.9367	237.1094	13.9520	96.9338	0.498071	237.5195	0.2114	65.3056
2	5	-4.1933	3.4683	277.6715	7.3298	95.8509	0.494092	277.7683	0.2476	63.6205
2	6	-4.1933	4.0000	274.0878	0.0000	95.9535	0.494468	274.0878	0.2443	63.7824
3	1	-3.9467	1.4851	112.9896	98.6735	98.7348	0.504664	150.0103	0.1332	68.0751
3	2	-3.9467	1.9881	119.2811	55.7369	99.1459	0.506168	131.6609	0.1168	68.6990
3	3	-3.9467	2.4911	230.3713	50.8948	97.1220	0.498801	235.9263	0.2099	65.5554
3	4	-3.9467	2.9940	245.1004	28.8405	96.7471	0.497390	246.7914	0.2197	65.0115
3	5	-3.9467	3.4970	284.0488	15.6501	95.6688	0.493423	284.4796	0.2537	63.3345
3	6	-3.9467	4.0000	279.0134	0.0000	95.8206	0.493979	279.0134	0.2487	63.5740
4	1	-3.7000	1.7293	133.0260	133.0260	97.9867	0.501924	188.1272	0.1672	66.9351
4	2	-3.7000	2.1835	152.2549	70.3057	98.2987	0.503055	167.7034	0.1490	67.4243
4	3	-3.7000	2.6376	251.8880	69.9268	96.4944	0.496505	261.4141	0.2328	64.5736
4	4	-3.7000	3.0917	258.5497	37.0252	96.2556	0.495578	261.1873	0.2327	64.2546
4	5	-3.7000	3.5459	293.1525	20.9197	95.3831	0.492369	293.8980	0.2622	62.8874
4	6	-3.7000	4.0000	287.4930	0.0000	95.5579	0.493011	287.4930	0.2564	63.1645
5	1	-3.4533	1.9750	165.0872	159.4938	97.2086	0.499081	229.5475	0.2042	65.7285
5	2	-3.4533	2.3800	189.7552	88.9515	97.4650	0.500033	209.5695	0.1864	66.1113
5	3	-3.4533	2.7850	274.0965	86.3530	95.8902	0.494289	287.3773	0.2562	63.6267
5	4	-3.4533	3.1900	275.5988	45.2297	95.7240	0.493625	279.2855	0.2490	63.4220
5	5	-3.4533	3.5950	305.6486	25.4583	95.0148	0.491013	306.7070	0.2738	62.3076
5	6	-3.4533	4.0000	300.2397	0.0000	95.1903	0.491657	300.2397	0.2679	62.5870
6	1	-3.2067	2.1978	205.2765	173.2552	96.0735	0.494909	268.6183	0.2394	63.9703
6	2	-3.2067	2.5582	229.6718	104.0684	96.2531	0.495595	252.1495	0.2247	64.2229
6	3	-3.2067	2.9187	297.7834	94.9364	95.0763	0.491294	312.5506	0.2790	62.3581
6	4	-3.2067	3.2791	296.9780	50.8083	94.9696	0.490842	301.2929	0.2689	62.2412
6	5	-3.2067	3.6396	321.7514	27.9020	94.4542	0.488941	322.9590	0.2885	61.4256
6	6	-3.2067	4.0000	317.4202	0.0000	94.6142	0.489529	317.4202	0.2835	61.6822
7	1	-2.9600	2.3929	246.1173	182.2747	94.9109	0.490626	306.2642	0.2734	62.1473
7	2	-2.9600	2.7143	267.6866	117.9731	95.0919	0.491347	292.5299	0.2611	62.3756
7	3	-2.9630	3.0357	323.2732	100.9577	94.2213	0.488127	338.6709	0.3027	61.0076
7	4	-2.9600	3.3571	321.5757	56.3837	94.1319	0.487751	326.4813	0.2918	60.9145

L	M	X (in)	Y (in)	U (fps)	V (fps)	P (psia)	RHO	Q (fps)	Mach	T (F)
7	5	-2.9600	3.6736	341.9079	29.8608	93.7507	0.486340	343.2094	0.3069	60.3106
7	6	-2.9600	4.0000	338.8474	0.0000	93.8859	0.486837	338.8474	0.3030	60.5294
8	1	-2.7133	2.5642	289.9157	188.6100	93.5079	0.485435	345.8682	0.3094	59.9300
8	2	-2.7133	2.8514	307.9821	127.9301	93.7199	0.486292	333.4952	0.2983	60.1912
8	3	-2.7133	3.1385	353.1446	104.0948	93.1129	0.484017	368.1668	0.3296	59.2511
8	4	-2.7133	3.4257	350.7214	60.2814	93.0513	0.483748	355.8642	0.3186	59.1957
8	5	-2.7133	3.7128	367.0750	30.9629	92.7906	0.482778	368.3786	0.3299	58.7817
8	6	-2.7133	4.0000	365.0832	0.0000	92.9035	0.483193	365.0832	0.3269	58.9667
9	1	-2.4667	2.7146	335.9522	191.6407	91.9380	0.479603	386.7687	0.3469	57.4175
9	2	-2.4667	2.9717	350.4988	134.3668	92.1960	0.480650	375.3716	0.3365	57.7394
9	3	-2.4667	3.2288	387.1967	105.6010	91.7925	0.479105	401.3389	0.3600	57.1358
9	4	-2.4667	3.4859	384.1126	63.0205	91.7586	0.478944	389.2481	0.3492	57.1185
9	5	-2.4667	3.7429	397.2721	31.6727	91.5882	0.478304	398.5327	0.3576	56.8485
9	6	-2.4667	4.0000	396.0271	0.0000	91.6830	0.478654	396.0271	0.3553	57.0061
10	1	-2.2200	2.8463	385.7795	192.0533	90.0749	0.472645	430.9412	0.3876	54.3954
10	2	-2.2200	3.0770	396.9902	137.6498	90.3795	0.473682	420.1767	0.3778	54.7871
10	3	-2.2200	3.3078	426.5248	105.6109	90.1440	0.472945	439.4053	0.3952	54.4635
10	4	-2.2200	3.5385	422.6905	64.3804	90.1463	0.472923	427.5654	0.3845	54.5002
10	5	-2.2200	3.7693	433.1660	31.9255	90.0516	0.472562	434.3409	0.3907	54.3518
10	6	-2.2200	4.0000	432.3408	0.0000	90.1354	0.472872	432.3408	0.3888	54.4940
11	1	-1.9733	2.9607	440.1175	189.7053	87.8573	0.464311	479.2614	0.4326	50.7368
11	2	-1.9733	3.1686	448.0872	136.0339	88.2164	0.465769	466.8662	0.4230	51.2182
11	3	-1.9733	3.3764	471.3597	104.1647	88.1287	0.465373	482.7320	0.4356	51.1456
11	4	-1.9733	3.5843	466.6714	64.4745	88.1745	0.465519	471.1042	0.4250	51.2510
11	5	-1.9733	3.7921	474.8778	31.7414	88.1431	0.465393	475.9374	0.4294	51.2060
11	6	-1.9733	4.0000	474.2174	0.0000	88.2221	0.465686	474.2174	0.4278	51.3423
12	1	-1.7267	3.0593	500.0783	184.4158	85.1723	0.454142	532.9986	0.4833	46.2146
12	2	-1.7267	3.2474	504.7398	135.5893	85.6008	0.455882	522.6344	0.4736	46.8202
12	3	-1.7267	3.4356	522.3541	101.1329	85.6519	0.456001	532.0542	0.4820	46.9892
12	4	-1.7267	3.6237	516.6715	63.2783	85.7507	0.456353	520.5320	0.4715	47.1831
12	5	-1.7267	3.8119	522.8704	31.0492	85.7759	0.456442	523.7915	0.4744	47.2336
12	6	-1.7267	4.0000	522.1891	0.0000	85.8554	0.456738	522.1891	0.4729	47.3748
13	1	-1.4800	3.1429	566.2529	175.5815	61.9333	0.441756	592.8501	0.5405	40.6179
13	2	-1.4800	3.3143	567.4408	130.0717	82.4540	0.443870	582.1578	0.5304	41.4001
13	3	-1.4800	3.4858	579.7939	96.2310	82.6443	0.444521	587.7255	0.5352	41.8215
13	4	-1.4800	3.6572	572.9430	60.6967	82.8076	0.445125	576.1491	0.5245	42.1299
13	5	-1.4800	3.8286	577.3073	29.7520	82.8869	0.445422	578.0735	0.5262	42.2757
13	6	-1.4800	4.0000	576.4726	0.0000	82.9705	0.445737	576.4726	0.5246	42.4267

L	M	X (in)	Y (in)	U (fps)	V (fps)	P (psia)	RHO	Q (fps)	Mach	T (F)
14	1	-1.2333	3.2125	639.0676	162.4290	78.0504	0.426727	659.3865	0.6054	33.6883
14	2	-1.2333	3.3700	636.5520	121.0465	78.6926	0.429342	647.9589	0.5943	34.7195
14	3	-1.2333	3.5275	643.9203	89.0877	79.0292	0.430567	650.0539	0.5958	35.4213
14	4	-1.2333	3.6850	635.7200	56.5499	79.2706	0.431485	638.2302	0.5847	35.8779
14	5	-1.2333	3.8425	638.3532	27.7308	79.4040	0.431995	638.9553	0.5852	36.1254
14	6	-1.2333	4.0000	637.2739	0.0000	79.4949	0.432340	637.2739	0.5835	36.2970
15	1	-0.9867	3.2685	718.4278	143.9795	73.4700	0.408725	732.7132	0.6786	25.1849
15	2	-0.9867	3.4148	711.9765	107.8905	74.2675	0.411994	720.1048	0.6660	26.5599
15	3	-0.9867	3.5611	714.5835	79.2171	74.7626	0.413865	718.9610	0.6642	27.5894
15	4	-0.9867	3.7074	704.8565	50.5751	75.0970	0.415164	706.6686	0.6524	28.2372
15	5	-0.9867	3.8537	705.8355	24.8312	75.2871	0.415905	706.2721	0.6518	28.6006
15	6	-0.9867	4.0000	704.4402	0.0000	75.3868	0.416293	704.4402	0.6500	28.7919
16	1	-0.7400	3.3115	803.7732	119.1704	68.1821	0.387547	812.5607	0.7607	14.8693
16	2	-0.7400	3.4492	793.2104	89.8965	69.1705	0.391649	798.2882	0.7456	16.7076
16	3	-0.7400	3.5869	791.2734	66.0987	69.8382	0.394253	794.0294	0.7408	18.1298
16	4	-0.7400	3.7246	779.9013	42.4700	70.2601	0.396011	781.0568	0.7280	19.0199
16	5	-0.7400	3.8623	779.2786	20.8999	70.5298	0.397005	779.5588	0.7262	19.5177
16	6	-0.7400	4.0000	777.5410	0.0000	70.6408	0.397439	777.5410	0.7242	19.7486
17	1	-0.4933	3.3417	893.9432	86.9769	62.2513	0.363220	898.1644	0.8519	2.6020
17	2	-0.4933	3.4734	879.1927	66.3566	63.4636	0.368344	881.6933	0.8340	5.0490
17	3	-0.4933	3.6050	873.0032	49.2059	64.3169	0.371780	874.3888	0.8254	6.9467
17	4	-0.4933	3.7367	859.9066	31.9340	64.8781	0.374068	860.4994	0.8113	8.1395
17	5	-0.4933	3.8683	857.7794	15.7672	65.1907	0.375342	857.9243	0.8083	8.7992
17	6	-0.4933	4.0000	855.6361	0.0000	65.3116	0.375837	855.6361	0.8059	9.0498
18	1	-0.2467	3.3595	987.1818	46.4557	55.8187	0.336066	988.2743	0.9522	-11.7118
18	2	-0.2467	3.4876	968.3735	36.6601	57.2806	0.342422	969.0721	0.9303	-8.4827
18	3	-0.2407	3.6157	958.2832	28.1325	58.3275	0.346782	958.6961	0.9179	-6.0113
18	4	-0.2467	3.7438	943.5326	18.7218	59.0161	0.349673	943.7183	0.9020	-4.4493
18	5	-0.2467	3.8719	939.9736	9.3489	59.3931	0.351253	940.0201	0.8976	-3.6020
18	6	-0.2467	4.0000	937.6150	0.0000	59.5321	0.351811	937.6150	0.8950	-3.2594
19	1	0.0300	3.3650	1081.9350	-3.0509	49.1261	0.306820	1081.0393	1.0612	-27.8336
19	2	0.0000	3.0920	1056.8199	0.3966	50.8097	0.310533	1058.8195	1.0300	-23.6353
19	3	0.0000	3.6190	1005.3991	2.5186	52.0880	0.319889	1005.3522	1.0172	-20.0885
19	4	0.0000	3.7960	1029.0988	2.6307	52.9016	0.323016	1029.0521	0.9990	-18.0950
19	5	0.0000	3.8730	1020.3133	1.0601	53.3012	0.325310	1020.3103	0.9932	-17.0236
19	6	0.0300	4.0000	1020.9992	0.0000	53.0719	0.325911	1020.9992	0.9897	-17.1525
20	1	0.2967	3.3581	1173.1200	-61.8580	42.3090	0.276131	1170.7501	1.1778	-46.0380
20	2	0.2967	3.0865	1106.0302	-42.2793	46.3023	0.285382	1108.8080	1.1003	-40.6080

L	M	X (in)	Y (in)	U (fps)	V (fps)	P (psia)	RHO	Q (fps)	Mach	T (F)
20	3	0.2967	3.6199	1132.0781	-26.9501	45.7752	0.291831	1132.3988	1.1227	-36.6236
20	4	0.2967	3.7033	1110.2692	-15.5120	46.7326	0.296129	1110.3772	1.1010	-30.0010
20	5	0.2967	3.8716	1107.7929	-6.9782	47.2613	0.298089	1107.8109	1.0932	-32.6288
20	6	0.2967	4.0000	1107.0192	0.0000	47.5001	0.299051	1107.0192	1.0913	-31.6133
21	1	0.9933	3.3389	1269.0032	130.7062	35.7235	0.200355	1275.7208	1.3101	-65.3967
21	2	0.9933	3.9711	1236.3910	-93.2116	37.9196	0.255030	1201.8900	1.2606	-58.6706
21	3	0.9933	3.6030	1219.8855	-62.9528	39.5789	0.202656	1221.5088	1.2360	-53.5867
21	4	0.9933	3.7356	1200.0219	-39.0606	40.7509	0.268366	1200.6576	1.2097	-50.0979
21	5	0.9933	3.8676	1193.6520	-20.1323	41.3780	0.271285	1193.8222	1.2002	-48.3097
21	6	0.9933	4.0000	1179.0275	0.0000	41.3050	0.271083	1179.0275	1.1867	-48.9338
22	1	0.7900	3.3072	1331.6916	205.2378	30.9096	0.220003	1307.0102	1.4120	-81.0062
22	2	0.7900	3.9058	1305.5170	155.2005	32.8860	0.230385	1310.7103	1.3663	-70.7132
22	3	0.7900	3.5893	1208.8579	111.5273	30.0362	0.238202	1293.6703	1.3360	-69.7912
22	4	0.7900	3.7229	1272.2609	-71.9720	35.0028	0.203108	1270.2950	1.3113	-66.9980
22	5	0.7900	3.6619	1267.2158	-32.7531	35.0110	0.205260	1267.6390	1.3026	-65.8919
22	6	0.7900	4.0000	1286.7778	0.0000	35.2207	0.201585	1286.7778	1.3233	-66.0902
23	1	0.9667	3.2609	1383.2720	203.9082	27.2665	0.201293	1000.6112	1.0980	-90.3500
23	2	0.9667	3.0123	1370.5310	187.5136	28.2071	0.206581	1387.2628	1.0731	-90.9272
23	3	0.9867	3.5596	1370.0887	128.3918	28.8860	0.209959	1380.0700	1.0609	-86.6062
23	4	0.9667	3.7070	1372.0803	-67.7226	29.0913	0.210905	1373.7506	1.0520	-87.6901
23	5	0.9867	3.8503	1375.5103	-12.7883	29.2328	0.211388	1375.5738	1.0520	-86.7353
23	6	0.9867	4.0016	1355.3965	9.0228	30.0018	0.217203	1355.0266	1.0227	-82.2707
24	1	1.2333	3.2219	1030.0370	252.8596	20.1025	0.180583	1056.1601	1.5809	-106.9657
24	2	1.2333	3.3752	1036.0195	196.1252	20.5272	0.186710	1009.7069	1.5706	-105.0328
24	3	1.2333	3.5291	1002.6305	135.8278	20.8330	0.188222	1009.0107	1.5660	-103.8817
24	4	1.2333	3.6829	1029.7078	-69.7732	25.6828	0.192670	1032.5235	1.5006	-100.2039
24	5	1.2333	3.8367	1397.0981	-69.0626	27.8136	0.203959	1399.2230	1.0876	-91.9200
24	6	1.2333	3.9905	1350.6985	-61.0675	30.3938	0.217202	1356.0702	1.0230	-82.2982
25	1	1.9600	3.1779	1001.9167	261.3019	21.3229	0.168627	1500.7777	1.6616	-116.6920
25	2	1.9600	3.3359	1082.1819	207.8316	21.7522	0.171088	1096.6810	1.6081	-116.8276
25	3	1.9600	3.0939	1070.3210	165.8729	22.8291	0.177072	1083.6231	1.6220	-112.0096
25	4	1.9600	3.6519	1033.9630	109.1605	25.1620	0.189852	1001.7000	1.5509	-102.2670
25	5	1.9800	3.8098	1392.5720	137.2705	27.8310	0.200008	1399.3220	1.0877	-91.8057
25	6	1.0800	3.9678	1309.1683	120.2698	30.9025	0.217206	1350.8790	1.0221	-82.2663
26	1	1.7267	3.1309	1509.8281	266.2235	20.0212	0.161080	1533.1197	1.7096	-125.3505
26	2	1.7267	3.2992	1099.0577	251.5096	20.6565	0.165113	1520.0007	1.6878	-122.3211
26	3	1.7267	3.4500	1063.9559	201.1105	22.8000	0.177299	1083.6789	1.6230	-112.2279
26	4	1.7267	3.6137	1017.7007	227.3063	25.5073	0.191703	1035.8071	1.5057	-100.9358

L	M	X (in)	Y (in)	U (fps)	V (fps)	P (psia)	RHO	Q (fps)	Mach	T (F)
26	5	1.7267	3.7735	1380.0098	207.8969	27.9910	0.200890	1395.9772	1.0830	-91.2585
26	6	1.7267	3.9333	1302.5008	187.8090	30.0071	0.217269	1355.5830	1.0228	-82.2083
27	1	1.9733	3.0909	1059.0269	257.2658	22.5791	0.175550	1081.5308	1.6221	-112.8006
27	2	1.9733	3.2505	1007.6522	280.2170	22.7877	0.176792	1075.2885	1.6135	-112.0919
27	3	1.9733	3.0101	1021.8776	293.0628	20.3002	0.185300	1051.7650	1.5730	-105.5272
27	4	1.9733	3.5696	1390.0753	279.9659	26.1367	0.195000	1421.9095	1.5251	-98.2955
27	5	1.9733	3.7292	1365.8397	260.7981	28.2135	0.206032	1390.5157	1.0750	-90.3801
27	6	1.9733	3.8868	1330.1585	200.7033	30.0107	0.217288	1355.7051	1.0229	-82.2369
28	1	2.2200	3.0975	1910.7995	-298.7612	25.9098	0.192169	1932.5583	1.5969	-103.1005
28	2	2.2200	3.2097	1906.2072	-302.8893	25.0096	0.189801	1938.9577	1.5561	-109.9101
28	3	2.2200	3.3620	1391.8996	-333.8577	25.5297	0.192227	1931.3259	1.5921	-101.5290
28	4	2.2200	3.5193	1370.9298	-339.0730	26.7231	0.198370	1910.5562	1.5090	-96.3879
28	5	2.2200	3.6766	1396.7931	-320.5292	28.5979	0.207899	1389.3600	1.4667	-89.2631
28	6	2.2200	3.8339	1317.7933	-293.3750	30.3985	0.217221	1350.0599	1.9170	-82.2726
29	1	2.9667	3.0090	1291.9779	-227.8106	32.9970	0.231216	1311.9087	1.3696	-75.3852
29	2	2.9667	3.1572	1303.0991	-283.3728	31.6862	0.229897	1333.5995	1.3950	-79.7109
29	3	2.9667	3.3109	1313.7529	-335.5893	30.9038	0.218006	1355.9372	1.4256	-83.5673
29	4	2.9667	3.4637	1313.3000	-362.5929	29.8082	0.219592	1362.9359	1.4352	-89.9830
29	5	2.9667	3.6169	1309.8925	-363.9159	30.1072	0.215866	1359.3710	1.4292	-83.5939
29	6	2.9667	3.7702	1315.1981	-339.5192	30.3952	0.217210	1358.3199	1.4257	-82.2996
30	1	2.7131	2.9605	1199.3369	-210.5938	90.0065	0.269795	1212.7615	1.2251	-52.1976
30	2	2.7133	3.1101	1210.9119	-238.5102	38.6329	0.258299	1239.1772	1.2530	-56.2885
30	3	2.7133	3.2598	1290.8253	-278.9910	36.0291	0.295565	1271.6627	1.3037	-69.0377
30	4	2.7133	3.4095	1267.3081	-319.9771	33.3000	0.232090	1305.8638	1.3535	-72.6991
30	5	2.7133	3.5591	1293.9769	-339.6986	31.3972	0.222919	1336.5995	1.3968	-78.9811
30	6	2.7133	3.7088	1306.3885	-325.0857	30.3957	0.217212	1346.2286	1.4100	-82.2909
31	1	2.9600	2.9170	1096.6960	-193.3771	47.0660	0.298373	1113.6193	1.1009	-39.2290
31	2	2.9600	3.0631	1118.7287	-193.6976	95.5797	0.291692	1135.3698	1.1278	-38.2301
31	3	2.9600	3.2092	1167.8982	-221.2927	91.6999	0.273125	1188.6785	1.1953	-98.9999
31	4	2.9600	3.3552	1221.3161	-267.3618	36.7919	0.299537	1250.2381	1.2785	-62.0356
31	5	2.9600	3.5013	1278.0603	-305.8819	32.6871	0.228973	1319.1593	1.3657	-79.6800
31	6	2.9600	3.6979	1297.5788	-322.8935	30.3963	0.217213	1337.1503	1.9035	-82.2872

Note: This chapter was reconstructed from OCR-extracted text and program code analysis. For production use, consult the original LASL technical documentation and verify input/output specifications with the actual Fortran source code listings in Appendix C.

References

1. L. M. Saunders, "Numerical Solution of the Flow Field in the Throat Region of a Nozzle," Brown Engineering Co. report BSVD-P-66-TN-001 (NASA CR 82601), August 1966.
2. R. F. Cuffel, L. H. Back, and P. F. Massier, "Transonic Flow-Field in a Supersonic Nozzle with Small Throat Radius of Curvature," *AIAA J.* 7, 1364–1366, July 1969.
3. A. A. Migdal, E. J. Wuchina, and W. D. Boyd, "Method for Predicting the Nozzle Operating Line," NASA Lewis Research Center Report NASA CR-72547, April 1968.
4. S. Prozan, "Numerical Solution of Transonic Flow in a Nozzle," Grumman Aircraft Engineering Corporation Report, 1966.
5. C. W. Laval, "Calculation of Two-Dimensional Compressible Flow in Convergent-Divergent Channels," AEDC Report TR-69-120, August 1969.
6. P. Serra, "Numerical Simulation of the Transonic Flow in a Convergent-Divergent Nozzle," CNES Report, 1970.
7. C. A. Brown and H. Ozcan, "Numerical Solution of Steady Flow in a Nozzle," Douglas Aircraft Report, 1969.
8. R. W. MacCormack, "The Effect of Viscosity in Hypervelocity Impact Cratering," AIAA Paper 69-354, 1969.
9. G. Moretti and M. Abbott, "A Time-Dependent Method for the Analysis of Transonic Flows," *AIAA J.* 4, 2136–2141, December 1966.
10. G. Moretti, "Transonic and Supersonic Flow Computations," *Advances in Applied Mathematics* 2, 1981.
11. J. von Neumann and R. D. Richtmyer, "A Method for the Numerical Calculation of Hydrodynamic Shocks," *J. Applied Phys.* 21, 232–237, March 1950.
12. *Ibid*, Reference 11.
13. M. C. Cline, "NAP: A Computer Program for the Computation of Two-Dimensional, Time-Dependent, Inviscid Nozzle Flow," Los Alamos Scientific Laboratory Report, 1977.

14. R. E. Loh and R. L. Loh, "A Comparison of Computer Execution Times," TRW Technical Report, 1975.
15. IBM Corporation, "System/360 Performance Characteristics," IBM Technical Manual, 1971.
16. R. H. Prozan and D. E. Kooker, "Numerical Analysis of Steady Nozzle Flow Using Relaxation Methods," AIAA Paper, 1971.
17. T. P. Thornock, "Experimental Study of Converging Nozzle Flow," NASA Technical Memorandum, June 1968.
18. D. L. Bresnahan and A. L. Johns, "Experimental Flow in a Plug Nozzle," NASA Contractor Report, 1972.

Appendix A

Characteristic Relations

A.1 $\eta = \text{constant}$ Reference Plane

[To be completed: Derivation of characteristic relations for inlet and exit boundaries (Appendix A from original)]

A.2 $\zeta = \text{constant}$ Reference Plane

[To be completed: Derivation of characteristic relations for wall and centerbody boundaries (Appendix B from original)]

Appendix B

Fortran Code Listing (LASL Identification: LP-0537)

B.1 Main Program (fortran_main.f)

This is the main program that orchestrates the NAP solver. It handles input/output, initialization, and time-stepping control. *OCR note: this listing received a conservative normalization pass for obvious token errors; unresolved ambiguities are tracked in conversion notes.*

```
1      PROGRAM MAIN(INPUT,OUTPUT,FILM,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,
2      1TAPE7=FILM)
3      C
4      ****
5      C      NAP, A COMPUTER PROGRAM FOR THE COMPUTATION OF TWO-DIMENSIONAL,
6      C      TIME-DEPENDENT, INVISCID NOZZLE FLOW
7      C
8      C          BY MICHAEL C. CLINE, T-3
9      C          LOS ALAMOS SCIENTIFIC LABORATORY
10     C
11     C      ****
12     C
13     C          PROGRAM ABSTRACT
14     C
15     C          THE EQUATIONS OF MOTION FOR TWO-DIMENSIONAL, TIME DEPENDENT,
16     C          INVISCID FLOW IN A NOZZLE ARE SOLVED USING THE SECOND-ORDER,
17     C          MACCORMACK FINITE-DIFFERENCE SCHEME, THE FLUID IS ASSUMED TO BE
18     C          A PERFECT GAS, ALL BOUNDARY CONDITIONS ARE COMPUTED USING A
19     C          SECOND-ORDER, REFERENCE PLANE CHARACTERISTIC SCHEME, THE STEADY
20     C          STATE SOLUTION IS OBTAINED AS THE ASYMPTOTIC SOLUTION FOR LARGE
21     C          TIME, THE NOZZLES MAY BE EITHER CONVERGING, CONVERGING-DIVERGING,
22     C          OR PLUG GEOMETRIES.
23     C
24     DIMENSION TITLE(8), UI(21), VI(21), PI(21), ROI(21)
25     COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81
26     1,21),QPT(81,21)
27     COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)
28     COMMON /SOLUTN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)
29     COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCONV,FDT,GAMMA,RGAS,GAMI,GA
30     1M2,L1,L2,L3,MI,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
31     2IERR,IU1,IU0,DXR,DVR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC
32     3,LC,PLOW,ROLW
33     COMMON /GEMIRYC/ NGEOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGEXW(81),
34     1YW(81),XWI(81),YWI(81),NXNY(81),NWPTS,IINT,IDIF,LT,NDIM
35     COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,RECB,RCICB,RCTCB,ANGICB
36     1,ANGECB,XCB(81),YCB(81),XCBI(81),YCB(81),NXXYCB(81),NCBPTS,IINTCB
37     2,IDIFCB,LECB
38     COMMON /BCC/ PT(21),TT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,N
39     3STAG
```

```

40      REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE
41      NAMELIST /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCONV,FDT,TSTOP,GAMMA,RGAS,
42      1NASM,NAME,NCONVI,NST,IUI,IUO,SMP,IPUNCH,IAV,CAV,NPLOT,IEX,LSS,CTA,
43      2XMU,XLA,RKMU,IUNIT,PLOW,ROLOW
44      NAMELIST /IVS/ U,V,P,RO,NID,NSTART,TSTART,RSTAR,RSTARS
45      NAMELIST /GEMTRY/ NDIM, XI, RJ, RT, XE, RCL, RCT, ANGL, ANGE, NGFOM, XWI, YWI
46      1,NWPTS, IINT, IDIF, LJET, JFLAG, NXNY, YW
47      NAMELIST /GCBL/ NGCB, RICB, RTCB, RCICB, RCTCB, ANGICB, ANGECB, YCB, NXNYC
48      1B, XCBI, YCBI, NCBPTS, IINTCB, IDIFCB
49      NAMELIST /BC/ PT, TT, THETA, PE, NSTAG, ISUPER, UI, VI, PI, ROI
50 C
51 C      READ IN DATA
52 C
53 10   TCONV=0.0 $ FDT=1.0 $ TSTOP=1.0 $ NASM=1 $ NSTAG=0 $ NAME=0
54     IPUNCH=0 $ NGCB=0 $ IINTCB=1 $ IDIFCB=1 $ NSTART=0 $ TSTART=0.0
55     IINT=1 $ IDIF=1 $ NMAX=0 $ NPRINT=0 $ GAMMA=1.4 $ RGAS=53.35
56     NID=1 $ NDIM=1 $ THETA(1)=0.0 $ PE=14.7 $ NST=0 $ N=0 $ IEX=1
57     NCONVI=1 $ IERR=0 $ JFLAG=0 $ IUI=1 $ IUO=1 $ SMP=0.95 $ ISUPER=0
58     IAV=0 $ CAV=4.0 $ NPLOT=-1 $ G=32.174 $ PC=144.0 $ TC=460.0
59     LC=12.0 $ IUNIT=0 $ LSS=2 $ CTA=0.5 $ XMU=0.2 $ XLA=1.0
60     RKMU=0.7 $ PLOW=0.01 $ ROLOW=0.0001 $ RSTAR=0.0 $ RSTARS=0.0
61     READ 650, TITLE
62     IF (EOF,5) 20,30
63 20   STOP
64 30   READ (5,CNTRL)
65     READ (5,IVS)
66     READ (5,GEMTRY)
67     READ (5,GCBL)
68     READ (5,BC)
69     IF (NAME,EQ,0) GO TO 40
70     WRITE (6,CNTRL)
71     WRITE (6,IVS)
72     WRITE (6,GEMTRY)
73     WRITE (6,GCBL)
74     WRITE (6, BC)
75 C
76 C      PRINT INPUT DATA
77 C
78 40   PRINT 660
79     PRINT 690
80     PRINT 680
81     PRINT 700
82     PRINT 670
83     PRINT 710, TITLE
84     PRINT 670
85     PRINT 720
86     NPRIND=ABS(FLOAT(NPRINT))
87     PRINT 730,
     IMAX,MMAX,NMAX,NPRIND,TCONV,FDT,NSTAG,NASM,IUNIT,IUI,IUO,IEX,NCONVI,TSTOP,N1D,NPLOT,IPUNCH,ISUPER,IAV,CAV,XMU,XLA,RKMU,CTA,LSS,SMP,
88     PRINT 670
89     IF (IUI,EQ,1) PRINT 740, GAMMA,RGAS
90     IF (IUI,EQ,2) PRINT 750, GAMMARGAS
91     PRINT 670
92     PRINT 780
93     IF (NDIM,EQ,0) PRINT 790
94     IF (NDIM,EQ,1) PRINT 800
95 C
96 C      CALCULATE THE NOZZLE RADIUS AND NORMAL
97 C
98     PRINT 670
99     CALL GBOM
100    IF (IERR,NE,0) GO TO 10
101    DY=1.0/FLOAT(MMAX-1)
102    IF (NGCB,NE,0) GO TO 60
103    RICB=0.0
104    RTCB=0.0
105    DO 50 L=1,IMAX
106    YCB(L)=0.0
107    NXNYCB(L)=0.0
108 50  CONTINUE
109    GO TO 90
110 60  XICB=XI
111    XECB=XE
112    CALL GEOMCB
113    LT=1 $ XI=XICB $ XE=XECB
114    Y0=0.0

```

```

115      DO 80 L=1,LMAX
116      IF (NDIM,EQ,0) Y=YW(L)-YCB(L)
117      IF (NDIM,EQ,1) Y=YW(L)**2-YCB(L)**2
118      IF (Y,GT,0.0) GO TO 70
119      PRINT 920
120      GO TO 10
121 70   IF (Y,LT,Y0) LT=L
122      Y0=Y
123 80   CONTINUE
124 90   IF (NSTAG,NE,0) GO TO 110
125      DO 100 M=2,MMAX
126      PT(M)=PT(1)
127      TT(M)=TT(1)
128      THETA(M)=THETA(1)
129 100  CONTINUE
130      PRINT 670
131      IF (IUI,EQ,1) PRINT 760, PT(1),TT(1),THETA(1),PE
132      IF (IUI,EQ,2) PRINT 770, PT(1),TT(1),THETA(1),PE
133      GO TO 130
134 110  PRINT 660
135      IF (IUI,EQ,1) PRINT 890, PE
136      IF (IUI,EQ,2) PRINT 770, PE
137      DO 120 M=1,MMAX
138      PRINT 910, M,PT(M),TT(M),THETA(M)
139 120  CONTINUE
140 C
141 C   CONVERT METRIC UNITS TO ENGLISH UNITS
142 C
143 130  IF (IUI,EQ,1) GO TO 180
144      RSTAR=RSTAR/2.54
145      RSTARS=RSTARS/6.4516
146      RGAS=RGAS/5.38032
147      DO 140 M=1,MMAX
148      PT(M)=PT(M)/6.8948
149      TT(M)=(TT(M)+40.0)*9.0/5.0-40.0
150 140  CONTINUE
151      PE=PE/6.8948
152      IF (ISUPER,EQ,0) GO TO 160
153      DO 150 M=1,MMAX
154      UI(M)=UI(M)/0.3048
155      VI(M)=VI(M)/0.3048
156      PI(M)=PI(M)/6.8948
157      ROI(M)=ROI(M)/16.02
158 150  CONTINUE
159 160  IF (N1D,NE,0) GO TO 180
160      IF (NSTART,NE,0) GO TO 180
161      DO 170 L=1,LMAX
162      DO 170 M=1,MMAX
163      U(L,M,1)=U(L,M,1)/0.3048
164      V(L,M,1)=V(L,M,1)/0.3048
165      P(L,M,1)=P(L,M,1)/6.8948
166      RO(L,M,1)=RO(L,M,1)/16.02
167 170  CONTINUE
168 C
169 C   CONVERT INPUT DATA UNITS TO INTERNAL UNITS
170 C
171 180  IF (IUNIT,EQ,0) GO TO 190
172      PC=1.0 $ LC=1.0 $ G=1.0
173      TC=0.0
174 190  TCONV=TCONV/100.0
175      T=TSTART*LC
176      TSTOP=TSTOP*LC
177      DO 200 L=1,LMAX
178      XWI(L)=0.0
179 200  CONTINUE
180      DO 210 M=1,MMAX
181      PT(M)=PT(M)*PC
182      TT(M)=TT(M)+TC
183      THETA(M)=THETA(M)*0.0174533
184 210  CONTINUE
185      PE=PE*PC
186      IF (N1D,NE,0) GO TO 230
187      DO 220 L=1,LMAX
188      DO 220 M=1,MMAX
189      P(L,M,1)=P(L,M,1)*PC
190      RO(L,M,1)=RO(L,M,1)/G

```

```

191 220 CONTINUE
192 230 GAM1=GAMMA/(GAMMA-1.0)
193  GAM2=(GAMMA-1.0)/2.0
194  IF (ISUPER,EQ,0) GO TO 250
195  DO 240 M=1,MMAX
196  U(1,M,1)=UI(M)
197  V(1,M,1)=VI(M)
198  P(1,M,1)=PI(M)*PC
199  RO(1,M,1)=ROI(M)/G
200  U(1,M,2)=U(1,M,1)
201  V(1,M,2)=V(1,M,1)
202  P(1,M,2)=P(1,M,1)
203  RO(1,M,2)=RO(1,M,1)
204 240 CONTINUE
205 250 L1=LMAX-1
206  L2=LMAX-2
207  L3=LMAX-3
208  M1=MMAX-1
209  M2=MMAX-1
210  IF (N1D,EQ,0) GO TO 260
211 C
212 C COMPUTE THE 1-D INITIAL-DATA SURFACE
213 C
214  CALL ONEDIM
215  IF (IERR,NE,0) GO TO 10
216 C
217 C COMPUTE THE INITIAL-DATA SURFACE MASS FLOW AND THRUST
218 C
219 260 IF (NPRINT,GT,0) GO TO 270
220  NPRINT=-NPRINT
221  GO TO 340
222 270 CALL MASFLO (0)
223 C
224 C CALCULATE AND PRINT THE INITIAL-VALUE SURFACE
225 C
226  DO 330 IU=1,2
227  IF (IU,EQ,1,AND,IU,EQ,2) GO TO 330
228  IF (IU,EQ,2,AND,IU,EQ,1) GO TO 330
229  NLINE=0
230  PRINT 660
231  PRINT 810, TSTART,NSTART
232  PRINT 820
233  IF (IU,EQ,1) PRINT 830
234  IF (IU,EQ,2) PRINT 840
235  PRINT 670
236  X=XI-DX
237  DO 300 L=1,LMAX
238  X=X+DX
239  CALL MAP (0,L,1,AL,BE,DE,LD1,AL1,BE1,DE1)
240  DYIO=DY/BE
241  Y=YCB(L)-DYIO
242  DO 300 M=1,MMAX
243  Y=Y+DYIO
244  VELMAG=SQRT(U(L,M,1)**2+V(L,M,1)**2)
245  XMACH=VELMAG/SQRT(GAMMA*P(L,M,1)/RO(L,M,1))
246  PRES=P(L,M,1)/PC
247  RHO=RO(L,M,1)/G
248  TEMP=P(L,M,1)/RHO/RGAS-TC
249  XP=X
250  YP=Y
251  UP=U(L,M,1)
252  VP=V(L,M,1)
253  IF (IU,EQ,1) GO TO 280
254  XP=XP*2.54
255  YP=YP*2.54
256  UP=UP*0.3048
257  VP=VP*0.3048
258  PRES=PRES*6.8948
259  RHO=RHO*16.02
260  VELMAG=VELMAG*0.3048
261  TEMP=(TEMP+40.0)*5.0/9.0-40.0
262 280 NLINE=NLINE+1
263  IF (NLINE,LT,55) GO TO 290
264  PRINT 660
265  PRINT 810, TSTART,NSTART
266  PRINT 820

```

```

267      IF (IU,EQ,1) PRINT 830
268      IF (IU,EQ,2) PRINT 840
269      PRINT 670
270      NLINE=1
271 290 PRINT 850, L,M,XP,YP,UP,VP,PRES,RHO,VELMAG,XMACH,TEMP
272 300 CONTINUE
273      IF (IU,EQ,2) GO TO 310
274      PRINT 870, MASST,THRUST,MASSI,MASSE
275      GO TO 320
276 310 MASST=MASST*0.4536
277      MASSI=MASSI*0.4536
278      MASSE=MASSE*0.4536
279      THRUST=THRUST*4.4477
280      PRINT 880, MASST,THRUST,MASSI,MASSE
281 320 IF (IUO,NE,3) GO TO 340
282 330 CONTINUE
283 340 IF (NPLOT,LE,0) GO TO 350
284      CALL PLOT (TITLE,TSTART,NSTART)
285      PRINT 1030, NSTART
286 350 IF (NMAX,EQ,0) GO TO 10
287 C
288 C   INITIALIZE THE TIME STFP INTEGRATION LOOP PARAMETERS
289 C
290      N1=1 $ N3=2 $ DQM=0.0 $ NS=0 $ NCONV=0 $ NC=0 $ LDUM=1 $ NPC=0
291      DXR=1.0/DX $ DYR=1.0/DY $ DXRS=DXR*DXR $ DYRS=DYR*DYR
292      LD=81 $ MD=21 $ LMD=LD*MD
293      IF (NASM,NE,0,AND,LT,NE,1) LDUM=LT-1
294      NPD=0
295      IF (JFLAG,EQ,0) GO TO 360
296      UD(1)=U(LJET-1,MMAX,N1)
297      VD(1)=V(LJET-1,MMAX,N1)
298      PD(1)=P(LJET-1,MMAX,N1)
299      ROD(1)=RO(LJET-1,MMAX,N1)
300      UD(2)=UD(1)
301      VD(2)=VD(1)
302      PD(2)=PD(1)
303      ROD(2)=ROD(1)
304 C
305 C   ENTER THE TIME STEP INTERGRATION LOOP
306 C
307 360 DO 580 N=1,NMAX
308      NDP=NPD+1
309      IF (NPD,NE,10) GO TO 370
310      NP=N+NSTART
311      PRINT 1040, NP
312      NPD=0
313 370 CONTINUE
314      LMD1=LMD*(N1-1)
315      LMD3=LMD*(N3-1)
316 C
317 C   CALCULATE DELTA T
318 C
319      DO 380 L=1,LMAX
320          CALL MAP (0,L,MD,AL,BE,DE,LD1,AL1,BE1,DE1)
321          DDXDY=DXRS+BE*BE*DYRS
322          DO 380 M=1,MMAX
323              LMNI=L*LD*(M-1)+LMD1
324              QS=U(LMNI)*U(LMNI)+V(LMNI)*V(LMNI)
325              AS=CAMMA*P(LMNI)/RO(LMNI)
326              UPA=SQRT(QS*DDXDY)+SQRT(AS*DDXDY)
327              IF (L,EQ,1,AND,M,EQ,1) UPAM=UPA
328              IF (UPA,GT,UPAM) UPAM=UPA
329 380      CONTINUE
330              DT=FDT/UPAM
331              T=T+DT
332              IF (T,LE,TSTOP) GO TO 390
333              T=T-DT
334              DT=TSTOP-T
335              T=TSTOP
336 C
337 C   DETERMINE IF THE EXIT FLOW IS SUBSONIC OR SUPERSONIC
338 C
339 390      IVEL=0
340      IF (QS,GE,AS) IVEL=1
341 C
342 c   CALCULATE THE NOZZLE WALL AND INTERIOR MESH POINTS

```

```

343 C
344 IF (IAV,NE,0) CALL SHOCK (1)
345 ICHAR=1
346 IB=1
347 CALL INTER
348 CALL WALL
349 IF (IERR,NE,0) GO TO 10
350 IF (NGCB,EQ,0) GO TO 400
351 IB=2
352 CALL WALL
353 IF (IERR,NE,0) GO TO 10
354 400 ICHAR=2
355 IB=1
356 CALL INTER
357 CALL WALL
358 IF (IERR,NE,0) GO TO 10
359 IF (NGCB,EQ,0) GO TO 410
360 IB=2
361 CALL WALL
362 IF (IERR,NE,0) GO TO 10
363 C
364 C EXTRAPOLATE THE EXIT MESH POINTS FOR SUPERSONIC FLOW
365 C
366 410 DO 420 M=1 ,MMAX
367 U(LMAX,M,N3)=U(L1 ,M,N3)+IEX*(U(L1 ,M,N3)-U(L2 ,M,N3))
368 V(LMAX,M,N3)=V(L1 ,M,N3)+IEX*(V(L1 ,M,N3)-V(L2 ,M,N3))
369 P(LMAX,M,N3)=P(L1 ,M,N3)+IEX*(P(L1 ,M,N3)-P(L2 ,M,N3))
370 RO(LMAX,M,N3)=RO(L1 ,M,N3)+IEX*(RO(L1 ,M,N3)-RO(L2 ,M,N3))
371 IF (P(LMAX,M,N3),GT,0.0 ,AND,RO(LMAX,M,N3),GT,0.0 ) GO TO 420
372 P(LMAX,M,N3)=P(L1 ,M,N3)
373 RO(LMAX,M,N3)=RO(L1 ,M,N3)
374 420 CONTINUE
375 V(LMAX,MMAX,N3)=-U(LMAX,MMAX,N3)*NXNY(LMAX)
376 V(LMAX,1 ,N3)=-U(LMAX,1 ,N3)*NXNYCB(LMAX)
377 C
378 C CALCULATE THE NOZZLE INLET MESH POINTS
379 C
380 IF (ISUPER,EQ,0) CALL INLET
381 C
382 C CALCULATE THE NOZZLE EXIT MESH POINTS FOR SUBSONIC FLOW
383 C
384 IF (IVEL,EQ,0) CALL EXITTT
385 IF (N,LE,NST) CALL SHOCK (2)
386 C
387 C DETERMINE THE MAXIMUM (DELTA U) /U
388 C
389 IF (TCONV,LE,0.0 ) GO TO 440
390 DDQM=0.0
391 DO 430 L=LDUM,LMAX
392 DO 430 M=1 ,MMAX
393 IF (U(L,M,N1),EQ,0.0 ) GO TO 430
394 DQ=ABS((U(L,M,N3)-U(L,M,N1))/U(L,M,N1))
395 IF (DQ,GT,DDQM) DDQM=DQ
396 430 CONTINUE
397 440 NC=NC+1
398 NPC=NPC+1
399 IF (DDQM,GE,TCONV) GO TO 450
400 NCONV=NCONV+1
401 IF (NCONV,EQ,1) NCHECK=N-1
402 IF (NCONV,GE,NCONV) NC=NPRINT
403 450 IF (N,EQ,NMAX) NC=NPRINT
404 IF (N,GE,NCHECK+NCONV) NCONV=0
405 IF (T,EQ,TSTOP) NC=NPRINT
406 IF (NC,EQ,NPRINT) GO TO 460
407 IF (NPC,EQ,NPLOT) GO TO 550
408 GO TO 570
409 C
410 C COMPUTE THE SOLUTION SURFACE MASS FLOW AND THRUST
411 C
412 460 ICN=0
413 IF (JFLAG,EQ,0) GO TO 470
414 IF (LT,NE,LJET-1) GO TO 470
415 UDUM=U(LT,MMAX,N3)
416 RODUM=RO(LT,MMAX,N3)
417 U(LT,MMAX,N3)=UD(3)
418 RO(LT,MMAX,N3)=ROD(3)

```

```

419      ICN=1
420 470      CALL MASFLO (1)
421      IF (ICN,EQ,0) GO TO 480
422      U(LT,MMAX,N3)=UDUM
423      RO(LT,MMAX,N3)=RODUM
424 C
425 C      CALCULATE AND PRINT THE SOLUTION SURFACE
426 C
427 480      DO 540 IU=1,2
428      IF (IUO,EQ,1,AND,IU,EQ,2) GO TO 540
429      IF (IUO,EQ,2,AND,IU,EQ,1) GO TO 540
430      NLINE=0
431      PRINT 660
432      TIME=T/LC
433      DTIME=DT/LC
434      NP=NSTART
435      PRINT 860, NP,TIME,DTIME
436      PRINT 820
437      IF (IU,EQ,1) PRINT 830
438      IF (IU,EQ,2) PRINT 840
439      PRINT 670
440      X=XI-DX
441      DO 510 L=1,LMAX
442      X=X+DX
443      CALL MAP (0,L,1,AL,BE,DE,LD1,AL1,BE1,DE1)
444      DYIO=DY/BE
445      Y=YCB(L)-DYIO
446      DO 510 M=1,MMAX
447      Y=Y+DYIO
448      VELMAG=SQRT(U(L,M,N3)**2+V(L,M,N3)**2)
449      XMACH=VELMAG/SQRT(GAMMA*P(L,M,N3)/RO(L,M,N3))
450      PRES=P(L,M,N3)/PC
451      RHO=RO(L,M,N3)/G
452      TEMP=P(L,M,N3)/RHO/RGAS-TC
453      XP=X
454      YP=Y
455      UP=U(L,M,N3)
456      VP=V(L,M,N3)
457      IF (IU,EQ,1) GO TO 490
458      XP=XP*2.54
459      YP=YP*2.54
460      UP=UP*0.3048
461      VP=VP*0.3048
462      PRES=PRES*6.8948
463      RHO=RHO*16.02
464      VELMAG=VELMAG*0.3048
465      TEMP=(TEMP*40.0)*5.0/9.0-40.0
466 490      NLINE=NLINE+1
467      IF (NLINE,LT,55) GO TO 500
468      PRINT 660
469      PRINT 860, NP,TIME,DTIME
470      PRINT 820
471      IF (IU,EQ,1) PRINT 830
472      IF (IU,EQ,2) PRINT 840
473      PRINT 670
474      NLINE=1
475 500      PRINT 850, L,M,XP,YP,UP,VP,PRES,RHO,VELMAG,XMACH,TEMP
476 510      CONTINUE
477      IF (IU,EQ,2) GO TO 520
478      PRINT 870, MASST,THRUST,MASSI,MASSE
479      GO TO 530
480 520      MASST=MASST*0.4536
481      MASSI=MASSI*0.4536
482      MASSE=MASSE*0.4536
483      THRUST=THRUST*4.4477
484      PRINT 880, MASST,THRUST,MASSI,MASSE
485 530      IF (IUO,NE,3) GO TO 550
486 540      CONTINUE
487 550      IF (NPLOT,LT,0) GO TO 560
488      TIME=T/LC $ NP=NSTART
489      CALL PLOT (TITLE,TIME,np)
490      PRINT 1030, NP
491 C
492 C      CHECK FOR CONVERGENCE OF THE STEADY STATE SOLUTION
493 C
494 560      IF (DQM,LT,TCONV) GO TO 590

```

```

495      IF (T,EQ,TSTOP) GO TO 590
496      IF (N,EQ,NMAX) GO TO 590
497      IF (NC,EQ,NPRINT) NC=0
498      IF (NPC,EQ,NPLOT) NPC=0
499 570    CONTINUE
500      NNN=N1
501      N1=N3
502      N3=NNN
503 580    CONTINUE
504      C
505      C PUNCH A SIVS NAMELIST FOR RESTART
506      C
507 590    IF (NPLOT,GE,0) CALL ADV (10)
508    IF (IPUNCH,EQ,0) GO TO 10
509    DO 600 L=1,LMAX
510    DO 600 M=1,MMAX
511    P(L,M,N3)=P(L,M,N3)/PC
512    RO(L,M,N3)=RO(L,M,N3)*G
513 600    CONTINUE
514    PUNCH 930, NP,TIME
515    DO 610 M=1,MMAX
516    PUNCH 940, M
517    PUNCH 950, (U(L,M,N3),L=1,LMAX)
518 610    CONTINUE
519    DO 620 M=1,MMAX
520    PUNCH 960, M
521    PUNCH 950, (V(L,M,N3),L=1,LMAX)
522 620    CONTINUE
523    DO 630 M=1,MMAX
524    PUNCH 970, M
525    PUNCH 980, (P(L,M,N3),L=1,LMAX)
526 630    CONTINUE
527    DO 640 M=1,MMAX
528    PUNCH 990, M
529    PUNCH 1000, (RO(L,M,N3),L=1,LMAX)
530 640    CONTINUE
531    PUNCH 1010
532    NCARDS=(LMAX/7+2)*MMAX*4+22
533    PRINT 1020, NCARDS
534    GO TO 10
535      C
536      C FORMAT STATEMENTS
537      C
538 650    FORMAT (8A10)
539 660    FORMAT (1H1)
540 670    FORMAT (1H )
541 680    FORMAT (1HO)
542 690    FORMAT (1HO,15X,NAP, A COMPUTER PROGRAM FOR THE COMPUTATION OF TWO-DIMENSIONAL, TIME-DEPENDENT, INVISCID NOZZLE
FLOW,/,37X,59HBY MICHAEL C. CLINE, T-3 - LOS ALAMOS SCIENTIFIC LABORATORY)
543 700    FORMAT (1HO,10X,18HPROGRAM ABSTRACT 26X,86HTHE EQUATIONS OF MOTION FOR TWO-DIMENSIONAL, TIME DEPENDENT,
INVISCID FLOW IN A NOZZLE, /,21X,93HARE SOLVED USING THE SECOND-ORDER, MACCORMACK, FINITE-DIFFERENCE SCHEME, THE
FLUID IS ASSUMED, /,21X,95HTO BE A PERFECT GAS, ALL BOUNDARY CONDITIONS ARE COMPUTED USING A SECOND-ORDER,
REFERENCE PLANE, /,21X,91HCHARACTERISTIC SCHEME, THE STEADY STATE SOLUTION IS OBTAINED AS THE ASYMPTOTIC SOLUTION
FOR,/,21X,91HLARGE TIME, THE NOZZLES MAY BE EITHER CONVERGING, CONVERGING-DIVERGING, OR PLUG GEOMETRIES,)
544 710    FORMAT (1HO,10X,11HJOB TITLE //21X,8A10)
545 720    FORMAT (1HO,10X,20HCONTROL PARAMETERS -)
546 730    FORMAT
(1HO,20X,5HILMAX=,I2,2X,5HMMAX=,I2,3X,5HNMAX=,I4,2X,7HNPRINT=,I4,2X,6HTCONV=,F6.3,3X,4HFDT=,F4.2,2X,6HINSTAG=,I1,5X,5HNASM=,I1,4X,6HIUNIT
547 740    FORMAT (1HO,10X,13HFLUID MODEL -,/,21X,36HTHE RATIO OF SPECIFIC HEATS, GAMMA =,F6.4,26H AND THE GAS CONSTANT,
R =,F9.4,15H (FT-LBF/LBM-R))
548 750    FORMAT (1HO,10X,13HFLUID MODEL -,/,21X,36HTHE RATIO OF SPECIFIC HEATS, GAMMA =,F6.4,26H AND THE GAS CONSTANT,
R =,F9.4,9H (J/KG-K))
549 760    FORMAT (1HO,10X,21HBOUNDARY CONDITIONS -,/,21X,3HPT=,F9.4,7H (PSIA),5X,3HTT=,F9.4,4H (F),5X,6HITHETA=,F9.4,6H
(DEG),5X,3HPE=,F9.4,7H (PSIA))
550 770    FORMAT (1HO,10X,21HBOUNDARY CONDITIONS -,/,21X,3HPT=,F9.4,6H (KPA),5X,3HTT=,F9.4,4H (C),5X,6HITHETA=,F9.4,6H
(DEG),5X,3HPE=,F9.4,6H (KPA))
551 780    FORMAT (1HO,10X,15HFLOW GEOMETRY -)
552 790    FORMAT (1HO,20X,47HTWO-DIMENSIONAL, PLANAR FLOW HAS BEEN SPECIFIED)
553 800    FORMAT (1HO,20X,36HAXISYMMETRIC FLOW HAS BEEN SPECIFIED)
554 810    FORMAT (1H ,30HINITIAL-DATA SURFACE - TIME = ,F10.8,8H SECONDS,4H (N=,I4 ,1H))
555 820    FORMAT (1HO,11X,1HL,4X,1HM,9X,1HX,10X,1HY,10X,1HU,1IX,1HV,12X,1HP,11X,3HPRHO,9X,1HQ,11X,4HMACH,8X,1HT)
556 830    FORMAT (1H ,25X,4H(IN),7X,4H(IN),6X,5H(FPS),7X,5H(FPS),7X,6H(PSIA),6X,9H(LBM/FT3),4X,5H(FPS),10X,2HNO,8X,3H(F))
557 840    FORMAT (1H ,25X,4H(CM),7X,4H(CM),6X,5H(MPS),7X,5H(MPS),7X,6H (KPA),7X,7H(KG/M3),5X,5H(MPS),10X,2HNO,8X,3H(C))
558 850    FORMAT (1H ,7X,215,4F12.4,F13.4,F12.6,3F12.4)
559 860    FORMAT (1H ,20HSOLUTION SURFACE NO.,I5,3H - ,7HTIME = ,F10.8,20H SECONDS (DELTA T = ,F10.8,1H))
560 870    FORMAT (1HO,10X,5HMASS=,F9.4,10H (LBM/SEC),5X,7HTHRUST=,F11.4,6H (LBF),5X,6HMASS=,F9.4,5X,6HMASS=,F9.4)

```

```

561 880 FORMAT (1H0,10X,5HMASS=,F9.4,9H (KS/SEC) ,5X,7HTHRUST=,F11.4,10H (NEWTONS) ,5X,6HMASSI=,F9.4,5X,6HMASSE=,F9.4)
562 890 FORMAT (1H0,10X,21HBOUNDARY CONDITIONS
563 900 -,//,22X,1HM,11X,8HPT(P8IA) ,10X,5HTT(F) ,10X,10HHTHETA(DEG) ,10X,3HPE=,F7.3,7H (PSIA) ,/)
564 910 FORMAT (1H0,10X,21HBOUNDARY CONDITIONS
565 920 -,//,22X,1HM,11X,7HPT(KPA) ,12X,5HTT(C) ,10X,10HHTHETA(DEC) ,10X,3HPE=,F7.3,6H (KPA) ,/)
566 930 FORMAT (1X,18HSIVS N1D=0,NSTART=,I4,8H,TSTART=,F14.10,1H,)
567 940 FORMAT (1X,4HU(1,,I2,5H,1) =)
568 950 FORMAT (1X,7(F10.3,1H,)) )
569 960 FORMAT (1X,4HV(1,,I2,5H,1) =)
570 970 FORMAT (1X,4HP(1,,I2,5H,1) =)
571 980 FORMAT (1X,7(F10.4,1H,)) )
572 990 FORMAT (1X,4HRHO(1,,I2,5H,1) =)
573 1000 FORMAT (1X,7(F10.6,1H,)) )
574 1010 FORMAT (1X,1HS)
575 1020 FORMAT (1H0,27H***** EXPECT APPROXIMATELY ,I4 ,20H PUNCHED CARDS *****)
576 1030 FORMAT (1H0,31H***** EXPECT FILM OUTPUT FOR N=,I4 ,6H *****)
577 1040 FORMAT (1H ,2HN=,I4)
578 END

```

B.2 Geometry Subroutine (geom.f)

1	SUBROUTINE GEOM	GEO	10
2	C	GEO	20
3	C ****	GEO	30
4	C	GEO	40
5	C THIS SUBROUTINE CALCULATES THE NOZZLE RADIUS AND OUTER NORMAL	GEO	50
6	C	GEO	60
7	C ****	GEO	70
8	C	GEO	80
9	COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81)GEO	90	
10	1,21),QPT(81,21)	GEO	100
11	COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)	GEO	110
12	COMMON /SOLUTN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)	GEO	120
13	COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCONV,FDT,GAMMAR GAS,GAMI,GAGEO	130	
14	1M2,L1,L2,L3,M1,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,GEO	140	
15	2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TCGEO	150	
16	3,LC,PLOW,ROLOW	GEO	160
17	COMMON /GEMTRYC/ NGEOM, XI, RI, XT, RT, XE, RE, RCI, RCT, ANGI, ANGE,XW(81),GEO	170	
18	IW(81),XWI(81),YWI(81),NXNY(81),NWPTS,IINT, IDIF, LT,NDIM	GEO	180
19	COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,REC B,RCICB,RCTCB,ANGICEGEO	190	
20	2,ANGE CB,XCB(81),YCB(81),XCB(81),YCB(81),XCB(81),YCB(81),NCBPTS,IINTCBGEO	200	
21	3,IDIFCB,LECB	GEO	210
22	COMMON /BCC/ PT(21),TT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,NGEO	220	
23	1STAG	GEO	230
24	REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE	GEO	240
25	C	GEO	250
26	GO TO (10,30,120,170), NGEOM	GEO	260
27	C	GEO	270
28	C CONSTANT AREA DUCT CASE	GEO	280
29	C	GEO	290
30	10 PRINT 230	GEO	300
31	IF (IUI,EQ,1) PRINT 250, XI,RI,XE	GEO	310
32	IF (IUI,EQ,2) PRINT 260, XI,RI,XE	GEO	320
33	LT=LMAX	GEO	330
34	DX=(XE-XI)/(LMAX-1)	GEO	340
35	XT=XE	GEO	350
36	RT=RI	GEO	360
37	RE=RI	GEO	370
38	DO 20 L=1,IMAX	GEO	380
39	YW(L)=RI	GEO	390
40	NXNY(L)=0.0	GEO	400
41	20 CONTINUE	GEO	410
42	IF (JFLAG,EQ,0) GO TO 210	GEO	420
43	C	GEO	430
44	XWL=XI+(LJET-2)*DX	GEO	440
45	IF (IUI,EQ,1) PRINT 370, XWL,LJET,LMAX	GEO	450
46	IF (IUI,EQ,2) PRINT 380, XWL,LJET,LMAX	GEO	460
47	GO TO 210	GEO	470
48	C	GEO	480
49	C CIRCULAR-ARC, CONICAL NOZZLE CASE	GEO	490

```

50 C
51 30 PRINT 230
52 IF (RCI,EQ,0.0,OR,RCT,EQ,0.0) GO TO 200
53 ANI=ANGI*3.141593/180.0
54 ANE=ANGE*3.141593/180.0
55 XTAN=XI+RCI*SIN(ANI)
56 RTAN=RI+RCI*(COS(ANI)-1.0)
57 RT1=RT-RCT*(COS(ANI)-1.0)
58 XT1=XTAN+(RTAN-RT1)/TAN(ANI)
59 IF (XT1,GE,XTAN) GO TO 40
60 XT1=XTAN
61 RT1=RTAN
62 40 XT=XT1+RCT*SIN(ANE)
63 XT2=XT+RCT*SIN(ANE)
64 RT2=RT+RCT*(1.0-COS(ANE))
65 RE=RT2+(XE-XT2)*TAN(ANE)
66 LT=1
67 DX=(XE-XI)/(LMAX-1)
68 IF (IUI,EQ,1) PRINT 270, XI,RI,RT,XE,RCI,RCT,ANGI,ANGE,XT,RE
69 IF (IUI,EQ,2) PRINT 280, XI,RI,RT,XE,RCI,RCT,ANGI,ANGE,XT,RE
70 DO 110 L=1,LMAX
71 X=XI+(L-1)*DX
72 IF (X,GE,XI,AND,X,LE,XTAN) GO TO 50
73 IF (X,GT,XTAN,AND,X,LE,XT1) GO TO 60
74 IF (X,GT,XT1,AND,X,LE,XT) GO TO 70
75 IF (X,GT,XT,AND,X,LE,XT2) GO TO 80
76 IF (X,GT,XT2,AND,X,LE,XE) GO TO 90
77 C
78 50 YW(L)=RI+RCI*(COS(ASIN((X-XI)/RCI))-1.0)
79 NXNY(L)=(XI-XI)/(YW(L)-RI+RCI)
80 GO TO 100
81 C
82 60 YW(L)=RT1+(XT1-X)*TAN(ANI)
83 NXNY(L)=TAN(ANI)
84 GO TO 100
85 C

```

B.3 Inlet Boundary Conditions (inlet.f)

OCR note: this listing received a conservative normalization pass for obvious token errors; unresolved ambiguities are tracked in conversion notes.

```

1 SUBROUTINE INLET
2 C ****
3 C
4 C THIS SUBROUTINE CALCULATES THE BOUNDARY MESH POINTS AT THE NOZZLE
5 C INLET FOR SUBSONIC FLOW
6 C ****
7 C ****
8 C
9 COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81
10 ,21),QPT(81,21)
11 COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)
12 COMMON /SOLUTN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)
13 COMMON /CNTRL/ LMAX,MMAXNMAX,NPRINT,TCONV,FDT,GAMMARGAS,GAMI,GA
14 1M2,L1,L2,L3,MI,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,NID,LJET,JFLAG,
15 2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC
16 3,LC,PLOW,RCLOW
17 COMMON /GEMIRYC/ NGEOM, XI, RI, XT, RT, XE, RE, RCI, RCT, ANGI, ANGE, XW(81),
18 1YW(81), XWI(81), YWI(81), NXNY(81), NWPTS, INT, IDIF, LT, NDIM
19 COMMON /GCB/ NGCB, XICB, RICB, XTCB, RTCB, XECB, RECB, RCICB, RCTCB, ANGICB
20 1, ANGECB, XCB(81), YCB(81), XCB(81), YCB(81), NXNYCB(81), NCBPTS, IIINTCB
21 2, IDIFCB, LECB
22 COMMON /BCC/ PT(21), TT(21), THETA(21), PE, MASSE, MASSI, MASST, THRUST, N
23 1STAG
24 REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE
25 C
26 GRGB=GAMMA*RGAS*G
27 X3=XI
28 ATERM2=0.0
29 ATERM3=0.0

```

```

30      DO 180 ICHAR=1,2
31      DO 180 M=1,MMAX
32      LMN1=1+LD*(M-1)+LMD1
33      LMN3=1+LD*(M-1)+LMD3
34      L1MN1=2+LD*(M-1)+LMD1
35      L1MN1=2+LD*(M-2)+LMD1
36      LMIN1=1+LD*(M-2)+LMD1
37      LMIN3=1+LD*M+LMD3
38      CALL MAP (2,1,M,AL,BE,DE,2,AL1,BE1,DE1)
39      U2=U(LMN1)
40      A2=SQRT(GAMMA*P(LMN1)/RO(LMN1))
41      IF (ICHAR,EQ,2) GO TO 10
42      U(LMN3)=U2
43      V(LMN3)=V(LMN1)
44      A3=A2
45 C
46 C CALCULATE THE PROPERTY INTERPOLATING POLYNOMIAL COEFFICIENTS
47 C
48 10   BU=(U(L1MN1)-U(LMN1))*DXR
49   BV=(V(L1MN1)-V(LMN1))*DXR
50   BP=(P(L1MN1)-P(LMN1))*DXR
51   BRO=(RO(L1MN1)-RO(LMN1))*DXR
52   BYCB=(YCB(2)-YCB(1))*DXR
53   BAL=(AL1-AL)*DXR
54   BBE=(BE1-BE)*DXR
55   CU=U(1,M,N1)-BU*X3
56   CV=V(1,M,N1)-BV*X3
57   CP=P(1,M,N1)-BP*X3
58   CRO=RO(1,M,N1)-BRO*X3
59   CYCB=YCB(1)-BYCB*X3
60   CAL=AL-BAL*X3
61   CBE=BE-BBE*X3
62 C
63 C CALCULATE THE CROSS DERIVATIVE INTERPOLATING POLYNOMIAL
64 C COEFFICIENTS
65 C
66   IF (M,EQ,1) GO TO 20
67   DU=(U(L1MN1)-U(L1MN1))*DVR
68   DV=(V(L1MN1)-V(L1MN1))*DVR
69   DP=(P(L1MN1)-P(L1MN1))*DVR
70   DRO=(RO(L1MN1)-RO(L1MN1))*DVR
71   DU1=(U(LMN1)-U(LMN1))*DVR
72   DV1=(V(LMN1)-V(LMN1))*DVR
73   DP1=(P(LMN1)-P(LMN1))*DVR
74   DRO1=(RO(LMN1)-RO(LMN1))*DVR
75   GO TO 40
76 20   IF (NGCB,NE,0) GO TO 30
77   DU=0.0
78   DV=V(2,2,N1)*DVR
79   DP=0.0
80   DRO=0.0
81   DU1=0.0
82   DV1=V(1,2,N1)*DVR
83   DP1=0.0
84   DRO1=0.0
85   GO TO 40
86 30   DU=(U(2,2,N1)-U(2,1,N1))*DVR
87   DV=(V(2,2,N1)-V(2,1,N1))*DVR
88   DP=(P(2,2,N1)-P(2,1,N1))*DVR
89   DRO=(RO(2,2,N1)-RO(2,1,N1))*DVR
90   DU1=(U(1,2,N1)-U(1,1,N1))*DVR
91   DV1=(V(1,2,N1)-V(1,1,N1))*DVR
92   DP1=(P(1,2,N1)-P(1,1,N1))*DVR
93   DRO1=(RO(1,2,N1)-RO(1,1,N1))*DVR
94 40   BDU=(DU-DU1)*DXR
95   BDV=(DV-DV1)*DXR
96   BDP=(DP-DP1)*DXR
97   BDRO=(DRO-DRO1)*DXR
98   CDU=DU1-BDU*X3
99   CDV=DV1-BDV*X3
100  CDP=DP1-BDP*X3
101  CDRO=DRO1-BDRO*X3
102 C
103 C CALCULATE X2
104 C
105  IF (ICHAR,EQ,2) A3=SQRT(GAMMA*P(LMN3)/RO(LMN3))

```

```

106      DO 50 IL=1 ,2
107      X2=X3- (U(1 ,M,N3) -A3+U2-A2) *0 .5*DT
108 C
109 C   INTERPOLATE FOR THE PROPERTIES
110 C
111      U2=BU*X2+CU
112      P2=BP*X2+CP
113      RO2=BRO*X2+CRO
114      A2=SQRT(GAMMA*P2/RO2)
115 50  CONTINUE
116      V2=BV*X2+CV
117      YCB2=BYCB*X2+CYCB
118      AL2=BAL*X2+CAL
119      BE2=BBE*X2+CBE
120      UV2=U2*AL2+V2*BE2
121 C
122 C   INTERPOLATE FOR THE CROSS DERIVATIVES
123 C
124      DU2=BDU*X2+CDU
125      DV2=BDV*X2+CDV
126      DP2=BDP*X2+CDP
127      DRO2=BDRO*X2+CDRO
128 C
129 C   CALCULATE THE PSI TERMS
130 C
131      IF (NDIM,EQ,0) GO TO 70
132      IF (M,EQ,1,AND,NGCB,EQ,0) GO TO 60
133      ATERM2=RO2*V2/ ((DY*(M-1)) /BE2+YCB2)
134      GO TO 70
135 60  ATERM2=RO2*BE2*D2
136 70  PSI12=-UV2*DRO2-RO2*AL2*DU2-RO2*BE2*D2-ATERM2
137      PSI22=-UV2*DU2-AL2*DP2/RO2
138      PSI42=-UV2*DP2+A2*A2*UV2*DRO2
139      IF (ICHAR,EQ,1) GO TO 130
140 C
141 C   CALCULATE THE CROSS DERIVATIVES AT THE SOLUTION POINT
142 C
143      IF (M,EQ,1,AND,NGCB,EQ,0) GO TO 80
144      IF (M,EQ,MMAX) GO TO 90
145      DU3=(U(LMN3) -U(LMN3 )) *DYR
146      DV3=(V(LMN3) -V(LMN3 )) *DYR
147      DP3=(P(LMN3) -P(LMN3 )) *DYR
148      DRO3=(RO(LMN3) -RO(LMN3 )) *DYR
149      GO TO 100
150 80  DU3=0 .0
151      DV3=V(1 ,2 ,N3) *DYR
152      DP3=0 .0
153      DRO3=0 .0
154      GO TO 100
155 90  DU3=(U(1 ,MMAX,N3) -U(1 ,M1,N3 )) *DYR
156      DV3=(V(1 ,MMAX,N3) -V(1 ,M1,N3 )) *DYR
157      DP3=(P(1 ,MMAX,N3) -P(1 ,M1,N3 )) *DYR
158      DRO3=(RO(1 ,MMAX,N3) -RO(1 ,M1,N3 )) *DYR
159 C
160 C   CALCULATE THE PSI TERMS AT THE SOLUTION POINT
161 C
162 100 IF (NDIM,EQ,0) GO TO 120
163      IF (M,EQ,1,AND,NGCB,EQ,0) GO TO 110
164      ATERM3=RO(LMN3) *V(LMN3) /((DY*(M-1)) /BE+YCB(1))
165      GO TO 120
166 110 ATERM3=RO(LMN3) *BE*D3
167 120 UV3=U(LMN3) *AL+V(LMN3) *BE
168      PSI13=-UV3*DRO3-RO(LMN3) *AL*DU3-RO(LMN3) *BE*D3-ATERM3
169      PSI23=-UV3*DU3-AL*DP3/RO(LMN3)
170      PSI43=-UV3*DP3+A3*A3*UV3*DRO3
171      GO TO 140
172 130 PSI23=PSI12
173      PSI43=PSI42
174      PSI13=PSI12
175 C
176 C   SOLVE THE COMPATIBILITY EQUATIONS FOR U,V,P, AND RO
177 C
178 140 MN3=SQRT(U(LMN3) *U(LMN3) +V(LMN3) *V(LMN3) ) /A3
179      T2=P2/ (RO2*RGAS*G)
180      PSI1B=(PSI12+PSI13)*0 .5
181      PSI2B=(PSI12+PSI13)*0 .5

```

```

182      PSI4B=(PSI42+PSI43)*0.5
183      GPSI1B=GAMMA*PSI1B
184      TITHETA=TAN(THETA(M))
185      UCORR=0.5+0.5/SQRT(1.0+TITHETA*TITHETA)
186 C
187      DO 160 ITER=1,20
188      DEM=(1.0+GAM2*MN3*MN3)
189      P(LMN3)=PT(M)/ (DEM*GAMI)
190      T3=TT(M)/DEM
191      PB=(P2+P(LMN3))*0.5
192      RTB=RGAS*(T2+T3)*0.5*G
193      U(LMN3)=U2+DT*PSI2B+(P(LMN3)-P2-(PSI4B+RTB*GPSI1B)*DT)*SQRT(RTB/GA
194      1MMA)/PB
195      U(LMN3)=U(LMN3)*UCORR
196      V(LMN3)=-U(LMN3)*TITHETA
197      OMN3=MN3
198      MN3=SQR((U(LMN3)*U(LMN3)+V(LMN3)*V(LMN3))/(T3*GRGB))
199      IF (OMN3,NE.,0.0) GO TO 150
200      IF (ABS(MN3-OMN3),LE.,0.0001) GO TO 170
201      GO TO 160
202 150 IF (ABS((MN3-OMN3)/OMN3),LE.,0.001) GO TO 170
203 160 CONTINUE
204 C
205      PRINT 190, M,N
206 170 RO(LMN3)=P(LMN3)/(RGAS*T3*G)
207 180 CONTINUE
208      RETURN
209 C
210 190 FORMAT (1H0,58H***** THE SOLUTION FOR NOZZLE ENTRANCE BOUNDARY POI
211      1NT ( 1,,I2,1H,,I4,43H) FAILED TO CONVERGE IN 20 ITERATIONS *****)
212      END

```

B.4 Wall Boundary Conditions (wall.f)

OCR note: this listing received a conservative normalization pass for obvious token errors; unresolved ambiguities are tracked in conversion notes.

```

1      SUBROUTINE WALL
2 C
3 C ****
4 C
5 C THIS SUBROUTINE CALCULATES THE BOUNDARY MESH POINTS AT THE NOZZLE
6 C WALL, EXHAUST JET BOUNDARY, AND CENTERBODY
7 C
8 C ****
9 C
10 COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81
11 1,21),OPT(81,21)
12 COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)
13 COMMON /SOLUTIN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)
14 COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCOMP,FDT,GAMMA,RGAS,GAMI,GA
15 1M2,L1,L2,L3,M1,M2,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
16 2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC
17 3,LC,PLOW,ROLOW
18 COMMON /GEMTRYC/ NGEOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGE,XW(81),
19 1NW(81),XWI(81),YWI(81),NNXY(81),NWPTS,IINT,IDIF,LT,NDIM
20 COMMON /GCB/ NGCB,XICB,RICB,XTCB,REC,B,RCICB,RCTCB,ANGICB
21 2,ANGECB,XCB(81),YCB(81),XCBI(81),YCB(81),NNYCB(81),NCBPTS,IINTCB
22 3,IDIFCB,LECB
23 COMMON /BCC/ PT(21),TT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,N
24 1STAG
25      REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE
26 C
27      IF (N.EQ.1) DELY=0.005
28      XWID=0.0
29      IF (IB,EQ.1) GO TO 10
30      Y1=0.0 $ Y3=0.0 $ MDUM1=1 $ MDUM1=2 $ SIGN=-1.0
31      GO TO 20
32 10      Y1=1.0 $ Y3=1.0 $ MDUM1=MAX $ MDUM1=M1 $ SIGN=1.0
33 20      ATERM2=0.0
34      ATERM3=0.0

```

```

35      LDUM=LMAX
36      IF (ICHAR.EQ. 2) LDUM=L1
37      LMDM=LD*(MDUM-1)
38      LMDMI=LDM*(MDUM-1)
39      DYS=SIGN*DYR
40      DO 350 L=2,LDUM
41      LMN1=L+LMDM+LMD1
42      LMN3=L+LMDM+LMD3
43      LM1N1=L+LMDM1+LMD1
44      L1MN1=L-1+LMDM+LMD1
45      L1MN3=L-1+LMDM+LMD3
46      L1MIN1=L-1+LMDM1+LMD1
47      IF (JFLAG.EQ.0) GO TO 50
48      IF (IB.EQ.2) GO TO 50
49 C
50      XWID=XWI(L)
51      IF (ICHAR.EQ.1) GO TO 30
52 C
53 C USE THE DUMMY ARRAYS TO MANIPULATE THE ONE-SIDED SOLUTIONS
54 C
55      IF (L.NE.LJET-2) GO TO 30
56      U(LMN3)=UD(3)
57      V(LMN3)=VD(3)
58      P(LMN3)=PD(3)
59      RO(LMN3)=ROD(3)
60      GO TO 50
61 30  IF (L.NE.LJET-1) GO TO 40
62      IF (ICHAR.EQ.1) UOLD=U(LMN1)
63      U(LMN1)=UD(1)
64      V(LMN1)=VD(1)
65      P(LMN1)=PD(1)
66      RO(LMN1)=ROD(1)
67      GO TO 50
68 40  IF (L.EQ.LJET) GO TO 50
69      U(LMN1)=UD(2)
70      V(LMN1)=VD(2)
71      P(LMN1)=PD(2)
72      RO(LMN1)=ROD(2)
73 C
74 50  U1=U(LMN1)
75      V1=V(LMN1)
76      P1=P(LMN1)
77      RO1=RO(LMN1)
78      U2=U1
79      V2=V1
80      A1=SQRT(GAMMA*P1/RO1)
81      A2=A1
82      IF (ICHAR.EQ.2) GO TO 60
83      U3=U1
84      V3=V1
85      P3=P1
86      RO3=RO1
87      A3=A1
88      GO TO 70
89 60  U3=U(LMN3)
90      V3=V(LMN3)
91      P3=P(LMN3)
92      RO3=RO(LMN3)
93      A3=SQRT(GAMMA*P3/RO3)
94 C
95 C CALCULATE THE PROPERTY INTERPOLATING POLYNOMIAL COEFFICIENTS
96 C
97 70  BU=(U1-U(LMN1))*DYS
98      BV=(V1-V(LMN1))*DYS
99      BP=(P1-P(LMN1))*DYS
100     BRO=(RO1-RO(LMN1))*DYS
101     CU=U1-BU*Y3
102     CV=V1-BV*Y3
103     CP=P1-BP*Y3
104     CRO=RO1-BRO*Y3
105 C
106 C CALCULATE THE CROSS DERIVATIVE INTERPOLATING POLYNOMIAL
107 C COEFFICIENTS
108 C
109    DU=(U1-U(LMN1))*DXR
110    DV=(V1-V(LMN1))*DXR

```

```

111      DP=(P1-P(L1MN1))*DXR
112      DRO=(RO1-RO(L1MN1))*DXR
113      DU1=(U(L1MN1)-U(L1M1N1))*DXR
114      DV1=(V(L1MN1)-V(L1M1N1))*DXR
115      DP1=(P(L1MN1)-P(L1M1N1))*DXR
116      DRO1=(RO(L1MN1)-RO(L1M1N1))*DXR
117      BDU=(DU-DU1)*DYS
118      BDV=(DV-DV1)*DYS
119      BDP=(DP-DP1)*DYS
120      BDRO=(DRO-DRO1)*DYS
121      CDU=DU-BDU*Y3
122      CDV=DV-BDV*Y3
123      CDP=DP-BDP*Y3
124      CDRO=DRO-BDRO*Y3
125 C
126 C   CALCULATE Y2
127 C
128 CALL MAP (1,L,MDUM,AL,BE,DE,LD1,AL1,BE1,DE1)
129 ALS=SQRT(AL*AL+BE*BE)
130 UV3=U3*AL+V3*BE+DE
131 AL2=AL
132 DO 90 ILL=1,3
133 UV2=U2*AL2+V2*BE+DE
134 Y2=Y3-(UV2*SIGN*AL*ALS*A2+UV3*SIGN*ALS*A3)*DT*0.5
135 C
136 C   INTERPOLATE FOR THE PROPERTIES
137 C
138 U2=BU*Y2+CU
139 V2=BV*Y2+CV
140 P2=BP*Y2+CP
141 RO2=BR0*Y2+CRO
142 AL2=Y2*AL
143 AD=GAMMA*P2/RO2
144 IF (AD.GT.0.0) GO TO 80
145 PRINT 360, N,L,MDUM
146 IERR=1
147 RETURN
148 80 A2=SQRT(AD)
149 90 CONTINUE
150 C
151 C   INTERPOLATE FOR THE CROSS DERIVATIVES
152 C
153 DU1=DU
154 DV1=DV
155 DP1=DP
156 DRO1=DRO
157 DU2=BDU*Y2+CDU
158 DV2=BDV*Y2+CDV
159 DP2=BDP*Y2+CDP
160 DRO2=BDRO*Y2+CDRO
161 C
162 C   CALCULATE THE PSI TERMS
163 C
164 IF (NDIM.EQ.0) GO TO 110
165 IF (IB.EQ.2) GO TO 100
166 ATERM2=RO2*V2/(YCB(L)+Y2/BE)
167 GO TO 110
168 100 ATERM2=RO2*V2/(YCB(L)+Y2/BE)
169 IF (IAV.EQ.0) GO TO 110
170 ATDS=RO2*V(L,2,N1)*DYR*BE
171 IF (ABS(ATERM2).GT.ABS(ATDS)) ATERM2=ATDS
172 C
173 110 PSI21=-U1*DU1-DP1/RO1
174 PSI31=-U1*DV1
175 PSI41=-U1*DP1+A1*A1*U1*DRO1
176 PSI12=-U2*DRO2-RO2*DU2-ATERM2
177 PSI22=-U2*DU2-DP2/RO2
178 PSI32=-U2*DV2
179 PSI42=-U2*DP2+A2*A2*U2*DRO2
180 IF (ICHAR.EQ.1) GO TO 150
181 C
182 C   CALCULATE THE CROSS DERIVATIVES AT THE SOLUTION POINT
183 C
184 IF (JFLAG.EQ.0) GO TO 120
185 IF (IB.EQ.2) GO TO 120
186 IF (L.EQ.2) GO TO 120

```

```

187      IF (L.NE.LJET-1) GO TO 120
188      IF (LJET.EQ.2) GO TO 120
189      GO TO 130
190 120 DU3=(U(LIMN3)-U3)*DXR
191      DV3=(V(LIMN3)-V3)*DXR
192      DP3=(P(LIMN3)-P3)*DXR
193      DRO3=(RO(LIMN3)-RO3)*DXR
194      GO TO 140
195 130 DU3=(U3-U(L-1,MDUM,N3))*DXR
196      DV3=(V3-V(L-1,MDUM,N3))*DXR
197      DP3=(P3-P(L-1,MDUM,N3))*DXR
198      DRO3=(RO3-RO(L-1,MDUM,N3))*DXR
199 C
200 C   ENTER THE EXHAUST JET ITERATION LOOP
201 C
202 140 IF (JFLAG.EQ.0) GO TO 150
203      IF (IB.EQ.2) GO TO 150
204      IF (L.LT.LJET) GO TO 150
205      YWI(L)=YW(L)
206      UDUM=U(LMN3)
207      VDUM=V(LMN3)
208      PDUM=P(LMN3)
209      RODUM=RO(LMN3)
210 150 DO 290 NJ=1,10
211      IF (ICHAR.EQ.1) GO TO 250
212      IF (JFLAG.EQ.0) GO TO 210
213      IF (IB.EQ.2) GO TO 210
214      IF (L.LT.LJET) GO TO 210
215      IF (NJ.EQ.1) GO TO 200
216      IF (NJ.GT.2) GO TO 180
217 160 YWOLD=YW(L)
218      POLD=P(LMN3)
219      IF (P(LMN3).LT.PE) GO TO 170
220      YW(L)=YW(L)+DELY
221      GO TO 190
222 170 YW(L)=YW(L)-DELY
223      GO TO 190
224 180 IF (P(LMN3).EQ.POLD) GO TO 160
225      DYDP=(YW(L)-YWOLD)/(P(LMN3)-POLD)
226      YNEW=YW(L)+DYDP*(PE-P(LMN3))
227      YWOLD=YW(L)
228      POLD=P(LMN3)
229      YW(L)=YNEW
230 190 IF (YW(L).LT.(0.98*YWOLD)) YW(L)=0.98*YWOLD
231      IF (YW(L).GT.(1.02*YWOLD)) YW(L)=1.02*YWOLD
232 200 NXNY(L)=(YW(L)-YW(L-1))*DXR
233      XWI(L)=(YW(L)-YWI(L))/DT
234      XWID=XWI(L)
235      CALL MAP(1,L,MDUM,AL,BE,DE,LD1,AL1,BE1,DE1)
236      ALS=SQRT(AL*AL+BE*BE)
237      U(LMN3)=UDUM
238      V(LMN3)=VDUM
239      P(LMN3)=PDUM
240      RO(LMN3)=RODUM
241 C
242 C   CALCULATE THE PSI TERMS AT THE SOLUTION POINT
243 C
244 210 IF (NDIM.EQ.0) GO TO 240
245      IF (IB.EQ.2) GO TO 220
246      ATERM3=RO3*V2/(YCB(L)+1.0/BE)
247      GO TO 240
248 220 IF (YCB(L).EQ.0.0) GO TO 230
249      ATERM3=RO3*V3/YCB(L)
250      IF (IAV.EQ.0) GO TO 240
251      ATDS=RO3*V(L,2,N3)*DYR*BE
252      IF (ABS(ATERM3).GT.ABS(ATDS)) ATERM3=ATDS
253      GO TO 240
254 230 ATERM3=RO3*V(L,2,N3)*DYR*BE
255 C
256 240 PSI13=-U3*DRO3-RO3*DU3-ATERM3
257      PSI23=-U3*DU3-DP3/RO3
258      PSI33=-U3*DV3
259      PSI43=-U3*DP3+A3*A3*U3*DRO3
260 C   CALCULATE THE COMPATIBILITY EQUATION COEFFICIENTS
261 C
262 250 ABR=NXNY(L)

```

```

263 IF (IB.EQ.2) ABR=NXYNCB(L)
264 ALB=0.5*(AL2+AL)/ALS
265 BEB=BE/ALS
266 A1B=(A1+A3)*0.5
267 A2B=(A2+A3)*0.5
268 ROI1B=(ROI+RO3)*0.5
269 RO2B=(RO2+RO3)*0.5
270 IF (ICHAR.EQ.1) GO TO 260
271 PSI21B=(PSI21+PSI23)*0.5
272 PSI31B=(PSI31+PSI33)*0.5
273 PSI41B=(PSI41+PSI43)*0.5
274 PSI12B=(PSI12+PSI13)*0.5
275 PSI22B=(PSI22+PSI23)*0.5
276 PSI32B=(PSI32+PSI33)*0.5
277 PSI42B=(PSI42+PSI43)*0.5
278 GO TO 270
279 260 PSI21B=PSI21
280 PSI31B=PSI31
281 PSI41B=PSI41
282 PSI12B=PSI12
283 PSI12B=PSI12
284 PSI22B=PSI22
285 PSI32B=PSI32
286 PSI42B=PSI42
287 C
288 C   SOLVE THE COMPATIBILITY EQUATIONS FOR U,V,P, AND RO
289 C
290 270 U(LMN3)=(U(LMN1)-ABR*(V(LMN1)-XWID)+(PSI21B-ABR*PSI31B)*DT)/(1.0+A
291 1BR*ABR)
292 V(LMN3)=-U(LMN3)*ABR+XWID
293 P(LMN3)=P2-SIGN*RO2B*A2B*(ALB*(U(LMN3)-U2)+BEB*(V(LMN3)-V2))+(PSI4
294 22B+A2B*A2B*PSI12B+SIGN*RO2B*A2B*(ALB*PSI22B+BEB*PSI32B))*DT
295 IF (P(LMN3).LE.0.0) P(LMN3)=PLOW*PC
296 RO(LMN3)=RO(LMN1)+(P(LMN3)-P(LMN1)-PSI41B*DT)/(A1B*A1B)
297 IF (RO(LMN3).LE.0.0) RO(LMN3)=ROLOW/G
298 IF (IAV.EQ.0) GO TO 280
299 C
300 C   ADD THE ARTIFICIAL VISCOSITY FOR SHOCK CALCULATIONS
301 C
302 IF (ICHAR.EQ.1) GO TO 280
303 U(LMN3)=U(LMN3)+(QUT(L,MDUM)-ABR*QVT(L,MDUM))/(1.0+ABR*ABR)
304 V(LMN3)=-U(LMN3)*ABR
305 P(LMN3)=P(LMN3)+QPT(L,MDUM)
306 280 IF (JFLAG.EQ.0) GO TO 350
307 IF (IB.EQ.2) GO TO 350
308 IF (L.LT.LJET-1) GO TO 350
309 IF (L.EQ.LJET-1) GO TO 300
310 IF (ICHAR.EQ.1) GO TO 350
311 DELP=ABS((P(LMN3)-PE)/PE)
312 IF (DELP.LE.0.001) GO TO 350
313 290 CONTINUE
314 GO TO 350
315 C
316 C   SOLVE THE COMPATIBILITY EQUATIONS FOR THE DOWNSTREAM SIDE OF THE
317 C   NOZZLE WALL EXIT POINT
318 C
319 300 UD(3)=U(LMN3)
320 VD(3)=V(LMN3)
321 PD(3)=P(LMN3)
322 ROD(3)=RO(LMN3)
323 PD(4)=PE
324 XM1=SQRT((UD(3)*UD(3)+VD(3)*VD(3))/(GAMMA*PD(3)/ROD(3)))
325 DUMD=1.0+GAM2*XM1*XM1
326 TD=PD(3)/ROD(3)/RGAS/G
327 TTD=TD*DUMD
328 IF (PE.GT.PD(3).AND.XM1.GE.1.0) GO TO 310
329 TTD=TTD*DUMD
330 IF (PE.GT.PD(3).AND.XM1.GE.1.0) GO TO 310
331 PID=PD(3)*DUMD*GAM1
332 ROD(4)=ROD(3)*(PE/PD(3))**((1.0/GAMMA))
333 GO TO 320
334 310 PRD=PE/PD(3)
335 GAMD=(GAMMA+1.0)/(GAMMA-1.0)
336 ROD(4)=ROD(3)*(GAMD*PRD+1.0)/(PRD+GAMD)
337 320 TE=PE/ROD(4)/RGAS/G
338 XMACH=SQRT((TTD/TE-1.0)/GAM2)

```

```

339 SS=SQRT(GAMMA*PE/ROD(4))
340 VMAG=XMACH*SS
341 UD(4)=VMAG/SQRT(1.0+NXNY(LJET)*NXNY(LJET))
342 VD(4)=-UD(4)*NXNY(LJET)
343 C
344 C AVERAGE THE 1-SIDED MACH NOS FOR THE INTERIOR POINT CALCULATIONS
345 C
346 XM2=SQRT((UD(4)*UD(4)+VD(4)*VD(4))/(GAMMA*PD(4)/ROD(4)))
347 IF (XM1.GE.1.0) GO TO 350
348 XMB=(XM1+XM2)/2.0
349 IF (XMB.GE.1.0) GO TO 330
350 DPL=1.0
351 DPR=1.0
352 GO TO 340
353 330 DPL=XM2-1.0
354 DPR=1.0-XM1
355 XMB=1.0
356 340 DPLR=DPR+DPL
357 DUM=1.0+GAM2*XMB*XMB
358 TEMP=TTD/DUM
359 P(LMN3)=PTD/DUM*GAM1
360 RO(LMN3)=P(LMN3)/(RGAS*TEMP*G)
361 QA=SQRT(2.0*GAM1*(RGAS*TTD*G-P(LMN3)/RO(LMN3)))
362 DNXXY=(DPR*NXNY(LJET)+DPL*NXNY(L))/DPLR
363 U(LMN3)=QA/SQRT(1.0+DNXXY*DNXXY)
364 V(LMN3)=-U(LMN3)*DNXXY
365 IF (ICHAR.EQ.1) GO TO 350
366 UD(1)=UD(3)
367 VD(1)=VD(3)
368 PD(1)=PD(3)
369 ROD(1)=ROD(3)
370 UD(2)=UD(4)
371 VD(2)=VD(4)
372 PD(2)=PD(4)
373 ROD(2)=ROD(4)
374 350 CONTINUE
375 IF (JFLAG.EQ.0) RETURN
376 IF (IB.EQ.2) RETURN
377 IF (ICHAR.EQ.1) RETURN
378 U(LJET-1,MMAX,N1)=UOLD
379 YWI(LMAX)=YW(LMAX)
380 YW(LMAX)=2.0*YW(L1)-YW(L2)
381 NXNY(LMAX)=- (YW(LMAX)-YW(L1))*DXR
382 XWI(LMAX)=(YW(LMAX)-YW(LMAX))/DT
383 DELY=ABS(YW(LJET)-YWI(LJET))
384 IF (DELY.EQ.0.0) DELY=0.0001
385 RETURN
386 C
387 360 FORMAT (1H0,61H***** A NEGATIVE SQUARE ROOT OCCURED IN SUBROUTINE
388 1WALL AT N=,I4,4H, L=,I2,8H, AND M=,I2,6H ****)
389 END

```

B.5 Interior Mesh Calculations (inter.f)

```

1 SUBROUTINE INTER
2 C
3 C ****
4 C THIS SUBROUTINE CALCULATES THE NOZZLE RADIUS AND OUTER NORMAL
5 C ****
6 C
7 C ****
8 C
9 COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81
10 1,21),QFT(81,21)
11 COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)
12 COMMON /SOLUTIN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)
13 COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCOMV,FDT,GAMMA,RGAS,GAM1,GA
14 1M2,L1,L2,L3,M1,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
15 2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC
16 3,LC,PLOW,ROLOW
17 COMMON /GEMIRYC/ NGEOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGE,XW(81),
18 1YW(81),XWI(81),YW(81),NXNY(81),NWPTS,HINT,IDLIF,LT,NDIM

```

```

19      COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,RECB,RCICB,RCTCB,ANGICB
20      2,ANGECB,XCB(81),YCB(81),XCB1(81),YCB1(81),NCNYCB(81),NCBPTS,IINTCB
21      3,IDLFCB,LECB
22      COMMON /BCC/ PT(21),TT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,N
23      1STAG
24      REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE
25 C
26      ATERM=0.0
27      IF (ICHAR,EQ,2) GO TO 40
28 C
29 C COMPUTE THE TENTATIVE SOLUTION AT T+DT
30 C
31      MDUM=1
32      IF (NGCB.NE.0) MDUM=2
33      DO 30 L=2,IMAX
34      DO 30 M=MDUM,MI
35      CALL MAP (1,L,M,AL,BE,DE,LD1,AL1,BE1,DE1)
36      LMD2=LD*(M-1)
37      LMN1=L+LMD2+LMD1
38      LMN3=L+LMD2+LMD3
39      LMN1=L-1+LMD2+LMD1
40      LMIN1=L+LD*(M-2)+LMD1
41      UB=U(LMN1)
42      VB=V(LMN1)
43      PB=P(LMN1)
44      ROB=RO(LMN1)
45      ASB=GAM/MA*PB/ROB
46      IF (M.NE.1) GO TO 10
47      DUDX=(UB-U(LMN1))*DXR
48      DPDX=(PB-P(LMN1))*DXR
49      DRODX=(ROB-RO(LMN1))*DXR
50      DVDY=(4.0*V(L,2,N1)-V(L,3,N1))*0.5*DYR
51      V(LMN3)=0.0
52 C
53      URHS=-UB*DUDX-DPDX/ROB
54      RORHS=-UB*DRODX-ROB*DUDX-(1+NDIM)*ROB*BE*DWDY
55      PRHS=-UB*DPDX+ASB*(RORHS+UB*DRODX)
56      GO TO 20
57 10     IF (NDIM.EQ.1) ATERM=ROB*VB/((M-1)*DY/BE+YCB(L))
58      UVB=UB+AL+VB*BE+DE
59      DUDX=(UB-U(LMN1))*DXR
60      DVDX=(VB-V(LMN1))*DXR
61      DPDX=(PB-P(LMN1))*DXR
62      DRODX=(ROB-RO(LMN1))*DXR
63      DUDY=(UB-U(LMN1))*DYR
64      DVDY=(VB-V(LMN1))*DYR
65      DPDY=(PB-P(LMN1))*DYR
66      DRODY=(ROB-RO(LMN1))*DYR
67 C
68      URHS=-UB*DUDX-UVB*DUDY-(DPDX+AL*DPDY)/ROB
69      VRHS=-UB*DUDX-UVB*DVDY-BE*DPDY/ROB
70      RORHS=-UB*DRODX-UVB*DRODY-ROB*(DUDX+AL*DUDY+BE*DWDY)-ATERM
71      PRHS=-UB*DPDX-UVB*DPDY+ASB*(RORHS+UB*DRODX+UVB*DRODY)
72      V(LMN3)=V(LMN1)+VRHS*DT
73 20     U(LMN3)=U(LMN1)+URHS*DT
74      P(LMN3)=P(LMN1)+PRHS*DT
75      RO(LMN3)=RO(LMN1)+RORHS*DT
76      IF (P(LMN3).LE.0.0) P(LMN3)=PLOW*PC
77      IF (RO(LMN3).LE.0.0) RO(LMN3)=ROLOW/G
78 30     CONTINUE
79      RETURN
80 C
81 C COMPUTE THE FINAL SOLUTION AT T+DT
82 C
83 40     MDUM=1
84      IF (NGCB.NE.0) MDUM=2
85      DO 70, L=2,L1
86      DO 70 M=MDUM,MI
87      CALL MAP (1,L,M,AL,BE,DE,LD1,AL1,BE1,DE1)
88      LMD2=LD*(M-1)
89      LMN1=L+LMD2+LMD1
90      LMN3=L+LMD2+LMD3
91      LMN3=L+LD*M+LMD3
92      UB=U(LMN3)
93      VB=V(LMN3)
94

```

```

95      PB=P(LMN3)
96      ROB=RO(LMN3)
97      ASB=GAMMA*PB/ROB
98      IF (M.NE. 1) GO TO 50
99      DUDX=(U(L1MN3)-UB)*DXR
100     DPDX=(P(L1MN1)-PB)*DXR
101     DRODX=(RO(L1MN1)-ROB)*DXR
102     DVDY=(4.0*V(L,2,N3)-V(L,3,N3))*0.5*DYR
103     V(LMN3)=0.0
104   C
105     URHS=-UB*DUDX-DPDX/ROB
106     RORHS=-UB*DRODX-ROB*DUDX-(1+NDIM)*ROB*BE*DVDY
107     PRHS=-UB*DPDX+ASB*(RORHS+UB*DRODX)
108     GO TO 60
109   50 IF (NDIM.EQ.1) ATERM=ROB*VB/( (M-1)*DY/BE+YCB(L))
110     UVB=UB+AL+VB*BE+DE
111     DUDX=(U(L1MN3)-UB)*DXR
112     DVDX=(V(L1MN3)-VB)*DXR
113     DPDX=(P(L1MN3)-PB)*DXR
114     DRODX=(RO(L1MN3)-ROB)*DXR
115     DUDY=(U(LMN3)-UB)*DYR
116     DVDY=(V(LMN3)-VB)*DYR
117     DPDY=(P(LMN3)-PB)*DYR
118     DRODY=(RO(LMN3)-ROB)*DYR
119   C
120     URHS=-UB*DUDX-UVB*DUDY-(DPDX+AL*DPDY)/ROB
121     VRHS=-UB*DVDX-UVB*DVDY-BE*DPDY/ROB
122     RORHS=-UB*DRODX-UVB*DRODY-ROB*(IUDX+AL*DUDY+BE*DVDY)-ATERM
123     PRHS=-UB*DPDX-UVB*DPDY+ASB*(RORHS+UB*DRODX+UVB*DRODY)
124     V(LMN3)=(V(LMN1)+V(LMN3)+VRHS*DT)*0.5
125   60 U(LMN3)=(U(LMN1)+U(LMN3)+URHS*DT)*0.5
126     P(LMN3)=(P(LMN1)+P(LMN3)+PRHS*DT)*0.5
127     RO(LMN3)=(RO(LMN1)+RO(LMN3)+RORHS*DT)*0.5
128     IF (P(LMN3).LE.0.0) P(LMN3)=PLOW*PC
129     IF (RO(LMN3).LE.0.0) RO(LMN3)=ROLOW/G
130     IF (IAV.EQ.0) GO TO 70
131   C
132   C ADD THE ARTIFICIAL VISCOSITY FOR SHOCK CALCULATIONS
133   C
134     U(LMN3)=U(LMN3)+QUT(L,M)
135     V(LMN3)=V(LMN3)+QVT(L,M)
136     IF (M.EQ. 1) V(LMN3)=0.0
137     P(LMN3)=P(LMN3)+QPT(L,M)
138   70 CONTINUE
139     RETURN
140   END

```

B.6 Mass Flow Calculations (masflo.f)

1	SUBROUTINE MASFLO(ISURF)	MAS	10
2	C	MAS	20
3	*****	MAS	30
4	C	MAS	40
5	C THIS SUBROUTINE CALCUTES THE INITIAL-DATA OR SOLUTION SURFACE	MAS	50
6	C MASS FLOW AND THRUST	MAS	60
7	C	MAS	70
8	*****	MAS	80
9	C	MAS	90
10	COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81,21) MAS 100	MAS	100
11	1,21),QPT(81,21)	MAS	110
12	COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)	MAS	120
13	COMMON /SOLUTN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)	MAS	130
14	COMMON /CNTRLIC/ LMAX,MMAX,NMAX,NPRINT,TCONV,FDT,GAMMAR GAS,GAMI,GAMAS	MAS	140
15	1M2,L1,L2,L3,M1,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,MAS	MAS	150
16	2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TCMAS	MAS	160
17	3,LC,PLOW,ROLOW	MAS	170
18	COMMON /GEMTRYC/ NGEOM, XI, RI, XT, RT, XE, RE, RCI, RCT, ANGI, ANGE,XW(81),MAS	MAS	180
19	IYW(81),XWI(81),YWI(81),NNXY(81),NWPTS,IINT, IDIF, LT, NDIM	MAS	190
20	COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,RECB,RCICB,RCTCB,ANGICE,MAS	MAS	200
21	1,ANGECP,XCB(81),YCB(81),XCB(81),YCB(81),NCBPTS,IINTCBMAS	MAS	210
22	2, IDIFCB,LCB	MAS	220
23	COMMON /BCC/ PT(21),IT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,NMAS	MAS	230

```

24      3STAG                               MAS  240
25      REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE   MAS  250
26 C
27      LC2=LC*LC                           MAS  260
28      LDUM=IMAX-1                         MAS  270
29      IF (LT,EQ,IMAX LT=IMAX-1             MAS  280
30      IF (JFLAG,NE,0) LDUM=LJET-1          MAS  290
31      IF (ISURF,EQ,1 ,OR,N1D,EQ,0) GO TO 30  MAS  300
32 C
33 C CALCULATE THE MASS FLOW AND THRUST FOR THE 1-D INITIAL-DATA  MAS  310
34 C SURFACE                                MAS  320
35 C
36      IF (NDIM,EQ,1) GO TO 10              MAS  330
37      AREAI=(YW(1) -YCB(1)) /LC2          MAS  340
38      AREAT=(YW(LT) -YCB(LT)) /LC2        MAS  350
39      AREAE=(YW(LDUM) -YCB(LDUM)) /LC2    MAS  360
40      GO TO 20                            MAS  370
41 10     AREAI=3.141593*(YW(1) **2-YCB(1) **2)/LC2           MAS  380
42     AREAT=3.141593*(YW(LT) **2-YCB(LT) **2)/LC2           MAS  390
43     AREAE=3.141593*(YW(LDUM) **2-YCB(LDUM) **2)/LC2         MAS  400
44     GO TO 20                            MAS  410
45 20     VMI=SQRT(U(1,1,1) **2+V(1,1,1) **2)           MAS  420
46     VMT=SQRT(U(LT,1,1) **2+V(LT,1,1) **2)           MAS  430
47     VME=SQRT(U(LDUM,1,1) **2+V(LDUM,1,1) **2)         MAS  440
48     MASSI=RO(1,1,1)*VMI*AREAI*G                  MAS  450
49     MASST=RO(LT,1,1)*VMT*AREAT*G                MAS  460
50     MASSE=RO(LDUM,1,1)*VME*AREAE*G              MAS  470
51     THRUST=RO(LDUM,1,1)*U(LDUM,1,1) **2*AREAE       MAS  480
52 C
53 C CALCULATE THE MASS FLOW AND THRUST FOR THE 2-D INITIAL-DATA  MAS  490
54 C SURFACE                                MAS  500
55 C
56 30     MASSI=0.0                          MAS  510
57     MASST=0.0                          MAS  520
58     MASSE=0.0                          MAS  530
59     THRUST=0.0                          MAS  540
60     DYI=DY*(YW(1) -YCB(1))            MAS  550
61     DYT=DY*(YW(LT) -YCB(LT))          MAS  560
62     DYE=DY*(YW(LDUM) -YCB(LDUM))        MAS  570
63     ND=1
64     IF (ISURF,EQ,1) ND=N3            MAS  580
65     DO 60 M=1,M1                      MAS  590
66     RADI=(M-1)*DYI+YCB(1)            MAS  600
67     RADT=(M-1)*DYT+YCB(LT)          MAS  610
68     RADE=(M-1)*DYE+YCB(LDUM)        MAS  620
69     IF (NDIM,EQ,1) GO TO 40          MAS  630
70     AREAI=DYI/LC2                   MAS  640
71     AREAT=DYT/LC2                   MAS  650
72     AREAE=DYE/LC2                   MAS  660
73     GO TO 50                          MAS  670
74 40     AREAI=3.141593*((RADI+DYI) **2-RADI**2)/LC2        MAS  680
75     AREAT=3.141593*((RADT+DYT) **2-RADT**2)/LC2        MAS  690
76     AREAE=3.141593*((RADE+DYE) **2-RADE**2)/LC2        MAS  700
77     ROUI=(RO(1,M,ND)*U(1,M,ND)+RO(1,M+1,ND)*U(1,M+1,ND))*0.5  MAS  710
78     ROUT=(RO(LT,M,ND)*U(LT,M,ND)+RO(LT,M+1,ND)*U(LT,M+1,ND))*0.5  MAS  720
79     ROUE=(RO(LDUM,M,ND)*U(LDUM,M,ND)+RO(LDUM,M+1,ND)*U(LDUM,M+1,ND))*0.5  MAS  730
80     1.5
81     ROUE2=(RO(LDUM,M,ND)*U(LDUM,M,ND) **2+RO(LDUM,M+1,ND)*U(LDUM,M+1,ND)*MAS  740
82     1)**2)*0.5                         MAS  750
83     MASSI=MASSI+ROUI*AREAI*G          MAS  760
84     MASST=MASST+ROUT*AREAT*G          MAS  770
85     MASSE=MASSE+ROUE*AREAE*G          MAS  780
86     THRUST=THRUST+ROUE2*AREAE        MAS  790
87 60     CONTINUE                          MAS  800
88     RETURN                            MAS  810
89     END                                MAS  820

```

B.7 One-Dimensional Initialization (onedim.f)

OCR note: this listing received a conservative normalization pass for obvious token errors; unresolved ambiguities are tracked in conversion notes.

```

1 SUBROUTINE ONEDIM
2 C
3 C ****
4 C THIS SUBROUTINE CALCULATES THE 1-D INITIAL-DATA SURFACE
5 C ****
6 C
7 COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81
8 1,21),QPT(81,21)
9 COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)
10 COMMON /SOLUTN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)
11 COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCONV,FDT,GAMMAR GAS,GAMI,GA
12 1M2,L1,L2,L3,M1,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
13 2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC
14 3,LC,PLOW,RCLOW
15 COMMON /GEMIRYC/ NGEOM, XI, RI, XT, RT, XE, RE, RCI, RCT, ANGI, ANGE, XW(81),
16 1YW(81), XWI(81), YWI(81), NXNY(81), NWPTS, IINT, IDIF, LT, NDIM
17 COMMON /GC/ NGCB, XICB, RICB, XTCB, RTCB, XECB, RECB, RCICB, RCTCB, ANGICB
18 1,ANGECB, XCB(81), YCB(81), XCBI(81), YCBI(81), NXNYCB(81), NCBPTS, IIINTCB
19 2, IDIFCB, LECB
20 COMMON /BCC/ PT(21), TT(21), THETA(21), PE, MASSE, MASSI, MASST, THRUST,
21 1NSTAG
22 REAL MN3,NXNY, MASSI, MASST, NXNYCB, MASSE
23
24 MN3=0.01
25 IF (N1D,EQ,-1,OR N1D,GT,2) MN3=2.0
26 GRGAS=1.0/(RGAS*G)
27 NXCK=0
28 ACOEF=2.0/(GAMMA+1.0)
29 BCOEF=(GAMMA-1.0)/(GAMMA+1.0)
30 CCOEF=(GAMMA+1.0)/2.0/(GAMMA-1.0)
31 IF (N1D,LT,0) GO TO 20
32 C
33 C OVERALL LOOP
34 C
35 IF (NGCB,NE,0) GO TO 10
36 RSTAR=RT
37 RSTARS=RT*RT
38 GO TO 20
39 10 RSTAR=YWL(T)-YCB(LT)
40 RSTARS=YWL(T)**2-YCB(LT)**2
41 20 DO 130 L=1,LMAX
42 IF (L,EQ,1,AND,N1D,EQ,-1) GO TO 130
43 IF (L,EQ,1,AND,N1D,GT,2) GO TO 130
44 X=XI+DX*(L-1)
45 IF (N1D,LT,0) GO TO 50
46 IF (NGCB,NE,0) GO TO 30
47 IF (X,LT,XT) GO TO 50
48 IF (X,GT,XT) GO TO 40
49 MN3=1.0
50 GO TO 100
51 30 IF (L,LT,LT) GO TO 50
52 IF (L,GT,LT) GO TO 40
53 MN3=1.0
54 GO TO 100
55 40 IF (NXCK,EQ,1) GO TO 50
56 IF (N1D,EQ,1,OR,N1D,EQ,3) MN3=1.1
57 IF (N1D,EQ,2,OR,N1D,EQ,4) MN3=0.9
58 NXCK=1
59 50 IF (NDIM,EQ,1) GO TO 60
60 RAD=YWL(L)-YCB(L)
61 ARATIO=RAD/RSTAR
62 GO TO 70
63 60 RADS=YWL(L)**2-YCB(L)**2
64 ARATIO=RADS/RSTARS
65 C
66 C NEWTON-RAPHSON ITERATION LOOP
67 C
68 70 DO 90 ITER=1,20
69 ABM = ACOEF + BCOEF * MN3**2

```

```

70      ABMC = ABM**CCOEF
71      FM = ABMC / MN3 - ARATIO
72      FPM = ABMC * (2.0 * BCOEF * CCOEF/ABM-1.0/MN3**2)
73      OMN3 = MN3
74      MN3 = OMN3 - FM/FPM
75      IF (MN3,GT,1.0,AND,OMN3,LT,1.0) MN3=0.99
76      IF (MN3,LT,1.0,AND,OMN3,GT,1.0) MN3=1.01
77      IF (MN3,GE,000) GO TO 80
78      MN3=MN3
79      GO TO 90
80 80  IF (ABS(MN3-OMN3)/OMN3,LE,0.0005) GO TO 100
81 90  CONTINUE
82      PRINT 140, L
83 C
84 C      Fill IN 2-D ARRAYS LOOP
85 C
86 100 DEM = 1.0 + GAM2 * MN3 * MN3
87      DEMP = DEM**GAM1
88      DNXNY = (NXNY(L) - NXNYCB(L)) / M1
89      DO 120 M=1,MMAX
90      P(L,M,1)=PT(M)/DEMP
91      TEMP=TT(M)/DEMP
92      RO(L,M,1)=P(L,M,1)*GRGAS/TEMP
93      Q=MN3*SQRT(GAMMA*P(L,M,1)/RO(L,M,1))
94      DN=NXNYCB(L)+DNXNY*(M-1)
95      DNS=DN*DN
96      IF (DNS,EQ,0.0) GO TO 110
97      SIGN=1.0
98      IF (DN,GT,0.0) SIGN=-1.0
99      U(L,M,1)=Q/SQRT(1.0+DNS)
100     V(L,M,1)=SIGN*Q/SQRT(1.0+1.0/DNS)
101     GO TO 120
102     U(L,M,1)=Q
103     V(L,M,1)=0.0
104 120  CONTINUE
105 130  CONTINUE
106      RETURN
107 C
108 140 FORMAT (1H0,10X,93H***** THE 1-D SOLUTION FOR THE INITIAL-DATA SUR
109 1FACE FAILED TO CONVERGE IN 20 ITERATIONS AT L=,I2,6H *****)
110  END

```

B.8 Centerbody Geometry (geomcb.f)

This subroutine calculates the centerbody radius and slope for cases with a centerbody configuration. It is used in the GEOM routine for geometry preprocessing. *OCR note: this listing was reconstructed from Appendix C PDF text blocks for readability; minor geometric/formatted-text normalization was required where scan quality was poor.*

```

1      SUBROUTINE GEOMCB
2 C
3 C **** THIS SUBROUTINE CALCULATES THE CENTERBODY RADIUS AND SLOPE ****
4 C
5 C
6 C
7 C OCR note:
8 C Reconstructed from Appendix C listing text blocks. Some FORMAT
9 C literal wording was normalized for readability in degraded regions.
10 C
11      DIMENSION YW(81),NXNY(81)
12      COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81,21),QPT(81,21)
13      COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)
14      COMMON /SOLUTIN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)
15      COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCONV,FDT,GAMMA,GRGAS,GAM1,GAM2,
16      1L1,L2,L3,M1,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
17      2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC,
18      3LC,PLOW,ROLOW
19      COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,RECB,RCICB,RCTCB,ANGICB,
20      1ANGECB,XCB(81),YCB(81),XCB1(81),YCB1(81),NXNYCB(81),NCBPTS,IINTCB,

```

```

21      2IDIFCB,LECB
22      COMMON /BCC/ PT(21),TT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,
23      1NSTAG
24      REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE
25 C
26      GO TO (10,30,120,160), NGCB
27 C
28 C      CYLINDRICAL CENTERBODY CASE
29 C
30 10      IF (IUI.EQ.1) PRINT 230, XICB,RICB,XECB
31      IF (IUI.EQ.2) PRINT 240, XICB,RICB,XECB
32      LECB=LMAX
33      DO 20 L=1,LMAX
34      YCB(L)=RICB
35      NXNYCB(L)=0.0
36 20      CONTINUE
37      GO TO 200
38 C
39 C      CIRCULAR-ARC, CONICAL CENTERBODY CASE
40 C
41 30      XI=XICB
42      RI=RICB
43      RT=RTCB
44      XE=XECB
45      RCI=RCICB
46      RCT=RCTCB
47      ANGI=ANGICB
48      ANGE=ANGECB
49      RI=2.0*RT-RI
50      IF (RCI.EQ.0.0.OR.RCT.EQ.0.0) GO TO 190
51      ANI=ANGI*3.141593/180.0
52      ANE=ANGE*3.141593/180.0
53      XTAN=XI+RCI*SIN(ANI)
54      RTAN=RI+RCI*(COS(ANI)-1.0)
55      RT1=RT-RCT*(COS(ANI)-1.0)
56      XT1=XTAN-(RTAN-RT1)/TAN(ANI)
57      IF (XT1.GE.XTAN) GO TO 40
58      XT1=XTAN
59      RT1=RTAN
60 40      RT=RT1-RCT*SIN(ANE)
61      XTCB=XT
62      XT2=XT+RCT*SIN(ANE)
63      RT2=RT+RCT*(1.0-COS(ANE))
64      RE=RT2+(XE-XT2)*TAN(ANE)
65      RECB=RE
66      RI=2.0*RT-RI
67      RE=2.0*RT-RE
68      IF (IUI.EQ.1) PRINT 250, XI,RI,RT,XE,RCI,RCT,ANGI,ANGE,XT,RE
69      IF (IUI.EQ.2) PRINT 260, XI,RI,RT,XE,RCI,RCT,ANGI,ANGE,XT,RE
70      RI=2.0*RT-RI
71      RE=2.0*RT-RE
72      DO 110 L=1,LMAX
73      X=XI+(L-1)*DX
74      IF (X.GE.XI.AND.X.LE.XTAN) GO TO 50
75      IF (X.GT.XTAN.AND.X.LE.XT1) GO TO 60
76      IF (X.GT.XT1.AND.X.LE.XT) GO TO 70
77      IF (X.GT.XT.AND.X.LE.XT2) GO TO 80
78      IF (X.GT.XT2.AND.X.LE.XE) GO TO 90
79 C
80 50      YW(L)=RI+RCI*(COS(ASIN((X-XI)/RCI))-1.0)
81      NXNY(L)=(XI-X)/(YW(L)-RI+RCI)
82      GO TO 100
83 C
84 60      YW(L)=RT1+(XT1-X)*TAN(ANI)
85      NXNY(L)=TAN(ANI)
86      GO TO 100
87 C
88 70      YW(L)=RT+RCT*(1.0-COS(ASIN((XT-X)/RCT)))
89      NXNY(L)=(XT-X)/(RCT+RT-YW(L))
90      GO TO 100
91 C
92 80      YW(L)=RT+RCT*(1.0-COS(ASIN((X-XT)/RCT)))
93      NXNY(L)=(XT-X)/(RCT+RT-YW(L))
94      GO TO 100
95 C
96 90      YW(L)=RT2+(X-XT2)*TAN(ANE)

```

```

97      NXNY(L) = -TAN(ANE)
98 C
99 100 YCB(L) = 2.0 * RTCB - YW(L)
100 NXNYCB(L) = -NXNY(L)
101 IF (YCB(L) .GE. 0.0) GO TO 110
102 YCB(L) = 0.0
103 NXNYCB(L) = 0.0
104 110 CONTINUE
105 GO TO 200
106 C
107 C GENERAL CENTERBODY CASE - INPUT CENTERBODY COORDINATES ONLY
108 C
109 120 PRINT 220
110 IF (IUI.EQ.1) PRINT 270, IINTCB, IDIFCB
111 IF (IUI.EQ.2) PRINT 280, IINTCB, IDIFCB
112 LDUM=NCBPTS
113 IF (IMAX.GT.NCBPTS) LDUM=IMAX
114 IP=1
115 DO 130 L=1,IMAX
116 XCB(L)=XICB+DX*(L-1)
117 CALL MILUP (XCB(L), YCB(L), IINTCB, NCBPTS, NCBPTS, 1, IP, XCBI, YCBI)
118 130 CONTINUE
119 DO 150 L=1,LDUM
120 IF (L.GT.IMAX) GO TO 140
121 SLOPE=DIF(L, IDIFCB, LMAX, XCB, YCB)
122 NXNYCB(L)=-SLOPE
123 IF (YCB(L) .GE. 0.0) GO TO 135
124 YCB(L) = 0.0
125 NXNYCB(L) = 0.0
126 SLOPE= -NXNYCB(L)
127 135 IF (L.LE.NCBPTS.AND.L.LE.IMAX) PRINT 310, L, XCBI(L), YCBI(L), XCB(L), YCB(L), SLOPE
128 IF (L.GT.NCBPTS.AND.L.LE.IMAX) PRINT 320, L, XCB(L), YCB(L), SLOPE
129 140 IF (L.LE.NCBPTS.AND.L.GT.IMAX) PRINT 330, L, XCBI(L), YCBI(L)
130 150 CONTINUE
131 GO TO 200
132 C
133 C GENERAL CENTERBODY CASE - INPUT CENTERBODY COORDINATES AND SLOPES
134 C
135 160 PRINT 220
136 IF (IUI.EQ.1) PRINT 290
137 IF (IUI.EQ.2) PRINT 300
138 DO 180 L=1,IMAX
139 XCB(L)=XICB+DX*(L-1)
140 IF (YCB(L) .GE. 0.0) GO TO 170
141 YCB(L) = 0.0
142 NXNYCB(L) = 0.0
143 170 SLOPE=-NXNYCB(L)
144 PRINT 340, L, XCB(L), YCB(L), SLOPE
145 180 CONTINUE
146 GO TO 200
147 C
148 190 PRINT 350
149 IERR=1
150 RETURN
151 C
152 200 IF (IUI.EQ.1) RETURN
153 DO 210 L=1,IMAX
154 YCB(L)=YCB(L)/2.54
155 210 CONTINUE
156 XICB=XICB/2.54
157 XECB=XECB/2.54
158 DX=DX/2.54
159 RETURN
160 C
161 C FORMAT STATEMENTS
162 C
163 220 FORMAT (1H )
164 230 FORMAT (1H0,20X,52HA CYLINDRICAL CENTERBODY HAS BEEN SPECIFIED BY,
165 1XICB=,F8.4,12H (IN) , RICB=,F8.4,16H (IN) , AND XECB=,F8.4,5H (IN))
166 240 FORMAT (1H0,20X,52HA CYLINDRICAL CENTERBODY HAS BEEN SPECIFIED BY,
167 1XICB=,F8.4,12H (CM) , RICB=,F8.4,16H (CM) , AND XECB=,F8.4,5H (CM))
168 250 FORMAT (1H0,20X,62HA CIRCULAR-ARC, CONICAL CENTERBODY HAS BEEN SPECIFIED BY XICB=,F8.4,5H (IN) ,
169 17H, RICB=,F8.4,6H (IN) , /,21X,8HRTCB=,F8.4,7H (IN) , ,5HXECB=,F8.4,5H (IN) ,
170 28H, RCICB=,F8.4,5H (IN) ,8H, RCTCB=,F8.4,5H (IN) ,9H, ANGICB=,F6.2,7H (DEG) ,
171 3/,21X,11HAND ANGECB=,F6.2,8H (DEG) , ,29HTHE COMPUTED VALUES ARE XTCB=,F8.4 ,
172 45H (IN) ,10H AND RECB=,F8.4,6H (IN) ,)
```

```

173 | 260 FORMAT (1H0,20X,62HA CIRCULAR-ARC, CONICAL CENTERBODY HAS BEEN SPECIFIED BY XICB=,F8.4,5H (CM) ,
174 | 17H, RICB=,F8.4,6H (CM) , /,21X,8HRTCB=,F8.4,7H (CM) , ,5HXECB=,F8.4,5H (CM) ,
175 | 28H, RCICB=,F8.4,5H (CM) ,8H, RCTCB=,F8.4,5H (CM) ,9H, ANGICB=,F6.2,7H (DEG) ,
176 | 3/,21X,11HAND ANGECB=,F6.2,8H (DEG) , ,29HTHE COMPUTED VALUES ARE XTCB=,F8.4,
177 | 45H (CM) ,10H AND RECB=,F8.4,6H (CM) ,
178 | 270 FORMAT (1H0,20X,76HA GENERAL CENTERBODY HAS BEEN SPECIFIED BY THE FOLLOWING PARAMETERS,
179 | 1 IINTCB=,I1,9H, IDIFCB=,I1,/,22X,1HL,10X,8HXCBI(IN) ,10X,8HYCBI(IN) ,9X,
180 | 27HXCBI(IN) ,10X,7HYCBI(IN) ,11X,5HSLOPE,/)
181 | 280 FORMAT (1H0,20X,76HA GENERAL CENTERBODY HAS BEEN SPECIFIED BY THE FOLLOWING PARAMETERS,
182 | 1 IINTCB=,I1,9H, IDIFCB=,I1,/,22X,1HL,10X,8HXCBI(CM) ,10X,8HYCBI(CM) ,9X,
183 | 27HXCBI(CM) ,10X,7HYCBI(CM) ,11X,5HSLOPE,/)
184 | 290 FORMAT (1H0,20X,68HA GENERAL CENTERBODY HAS BEEN SPECIFIED BY THE FOLLOWING PARAMETERS,
185 | 1 //,22X,1HL,11X,7HXCBI(IN) ,10X,7HYCBI(IN) ,11X,5HSLOPE,/)
186 | 300 FORMAT (1H0,20X,68HA GENERAL CENTERBODY HAS BEEN SPECIFIED BY THE FOLLOWING PARAMETERS,
187 | 1 //,22X,1HL,11X,7HXCBI(CM) ,10X,7HYCBI(CM) ,11X,5HSLOPE,/)
188 | 310 FORMAT (1H ,20X,I2 ,7X,F10.4 ,7X,F10.4 ,7X,F10.4 ,7X,F10.4 ,7X,F10.4)
189 | 320 FORMAT (1H ,20X,I2 ,41X,F10.4 ,7X,F10.4 ,7X,F10.4)
190 | 330 FORMAT (1H ,20X,I2 ,7X,F10.4 ,7X,F10.4)
191 | 340 FORMAT (1H ,20X,I2 ,7X,F10.4 ,7X,F10.4 ,7X,F10.4)
192 | 350 FORMAT (1H0,48H**** RCICB OR RCTCB WAS SPECIFIED AS ZERO ****)
193 | END

```

B.9 Table Interpolation (mtlup.f)

This subroutine performs multilinear interpolation on tabular input data. It is called by GEOM to interpolate geometry and flow properties from tables. This routine was adapted from a NASA-Langley program. *OCR note: this listing was reconstructed from Appendix C PDF text blocks for readability; diagnostic/comment text literals were normalized where scan artifacts obscured wording.*

```

1 SUBROUTINE MILUP (X,Y,M,N,MAX,NTAB, I ,VARI,VARD)
2 C
3 C THIS SUBROUTINE IS CALLED BY SUBROUTINE GEOM TO INTERPOLATE FOR
4 C EQUALLY SPACED NOZZLE WALL COORDINATES FOR THE TABULAR INPUT CASE.
5 C SUBROUTINE MILUP WAS TAKEN FROM THE NASA-LANGLEY PROGRAM LIBRARY.
6 C
7 C MODIFICATION OF LIBRARY INTERPOLATION SUBROUTINE FTILUP
8 C MULTIPLE TABLE LOOK-UP ON ONE INDEPENDENT VARIABLE TABLE
9 C USES AN EXTERNAL INTERVAL POINTER ( I ) TO START SEARCH
10 C 1 LESS THAN 0 WILL CHECK MONOTONICITY
11 C
12 C OCR note:
13 C Reconstructed from Appendix C listing text blocks. Some original
14 C diagnostic text literals were normalized where the scan was unreadable.
15 C Diagnostic FORMAT wording is normalized for readability.
16 C
17 C DIMENSION VARI(N) ,VARD(MAX,1) ,Y(1) ,V(3) ,YY(2)
18 C LOGICAL EX
19 C
20 C IF (M.EQ.0) GO TO 170
21 C IF (N.LE.1) GO TO 170
22 C EX=.FALSE.
23 C IF (I.GE.0) GO TO 60
24 C IF (N.LT.2) GO TO 60
25 C
26 C MONOTONICITY CHECK
27 C
28 C IF (VARI(2)-VARI(1)) 20 ,20 ,40
29 C
30 C ERROR IN MONOTONICITY
31 C
32 10 PRINT 190 , N
33 STOP
34 C
35 C MONOTONIC DECREASING
36 C
37 20 DO 30 J=2,N
38 C IF (VARI(J)-VARI(J-1)) 30 ,10 ,10
39 30 CONTINUE
40 GO TO 60

```

```

41 C
42 C      MONOTONIC INCREASING
43 C
44 40 DO 50 J=2,N
45   IF (VARI(J) -VARI(J-1)) 10,10,50
46 50 CONTINUE
47 C
48 C      INTERPOLATION
49 C
50 60 IF (I.LE.0) I=1
51   IF (I.GE.N) I=N-1
52 C
53 C      LOCATE I INTERVAL (VARI(I) .LE. X .LT. VARI(I+1))
54 C
55   IF ((VARI(I)-X)*(VARI(I+1)-X)) 100,100,70
56 C
57 C      IN GIVES DIRECTION FOR SEARCH OF INTERVALS
58 C
59 70 IN=SIGN(1.0,(VARI(N)-VARI(I))*(X-VARI(I)))
60 C
61 C      IF X OUTSIDE ENDPOINTS, EXTRAPOLATE FROM END INTERVAL
62 C
63 80 IF ((I+IN).LE.0) GO TO 90
64   IF ((I+IN).GE.N) GO TO 90
65   I=I+IN
66   IF ((VARI(I)-X)*(VARI(I+1)-X)) 100,100,80
67 C
68 C      EXTRAPOLATION
69 C
70 90 EX=.TRUE.
71 100 IF (M.EQ.2) GO TO 120
72 C
73 C      FIRST ORDER
74 C
75 DO 110 NT=1,NTAB
76 110 Y(NT)=(VARD(I,NT)*(VARI(I+1)-X)-VARD(I+1,NT)*(VARI(I)-X))/1
77   (VARI(I+1)-VARI(I))
78   IF (EX) I=I-IN
79   RETURN
80 C
81 C      SECOND ORDER
82 C
83 120 IF (N.EQ.2) GO TO 110
84   IF (I.EQ.(N-1)) GO TO 140
85   IF (I.EQ.1) GO TO 130
86 C
87 C      PICK THIRD POINT
88 C
89   SK=VARI(I+1)-VARI(I)
90   IF (ABS(X-VARI(I+1)).LT.ABS(VARI(I+2)-X)) GO TO 140
91 130 L=I
92   GO TO 150
93 140 L=I-1
94 150 V(1)=VARI(L)-X
95   V(2)=VARI(L+1)-X
96   V(3)=VARI(L+2)-X
97   DO 160 NT=1,NTAB
98   YY(1)=(VARD(L,NT)*V(2)-VARD(L+1,NT)*V(1))/(VARI(L+1)-VARI(L))
99   YY(2)=(VARD(L+1,NT)*V(3)-VARD(L+2,NT)*V(2))/(VARI(L+2)-VARI(L+1))
100 160 Y(NT)=(YY(1)*V(3)-YY(2)*V(1))/(VARI(L+2)-VARI(L))
101   IF (EX) I=I-IN
102   RETURN
103 C
104 C      ZERO ORDER
105 C
106 170 DO 180 NT=1,NTAB
107 180 Y(NT)=VARD(I,NT)
108   RETURN
109 C
110 190 FORMAT (1H1,49H TABLE BELOW OUT OF ORDER FOR MTLUP, N = ,I5)
111   END

```

B.10 Numerical Differentiation (diff.f)

This function calculates numerical derivatives for tabular input data. It is used by GEOM to compute nozzle wall slopes for the tabular input case, providing finite-difference approximations to derivatives. *OCR note: this listing was reconstructed from Appendix C PDF text blocks for readability; comments were normalized while preserving the derivative computation logic.*

```

1  FUNCTION DIF (L,M,NP,VARI,VARD)
2  C
3  C      THIS FUNCTION IS CALLED BY SUBROUTINE GEOM TO CALCULATE THE
4  C      NOZZLE WALL SLOPE FOR THE TABULAR INPUT CASE. FUNCTION DIF WAS
5  C      TAKEN FROM THE NASA-LANGLEY PROGRAM LIBRARY.
6  C
7  C      THIS FUNCTION SUBPROGRAM FINDS THE DERIVATIVE AT A GIVEN POINT L
8  C      FOR THE DESIRED X AND Y IN A GIVEN TABLE. THE N-POINT LAGRANGIAN
9  C      FORMULA IS USED WHERE N IS ODD.
10 C
11 C      L      = INTEGER, POINT OF X AND Y AT WHICH DERIVATIVE IS FOUND
12 C      M      = INTEGER, 1-5, TO DETERMINE THE POINT FORMULA, N=2*M+1
13 C      NP     = INTEGER, NUMBER OF POINTS IN TABLE OF VARIABLES
14 C      VARI   = ARRAY OF INDEPENDENT VARIABLE, X, VARI(NP)
15 C      VARD   = ARRAY OF DEPENDENT VARIABLE, Y, VARD(NP)
16 C
17 C      OCR note:
18 C      Reconstructed from Appendix C listing text blocks; numeric logic
19 C      preserved and comments normalized for readability.
20 C
21 DIMENSION VARI(NP),VARD(NP),X(11),Y(11)
22 C
23 DIF=1.777E19
24 IF (M.LT.1) RETURN
25 N=2*M+1
26 IF (M.GT.5.OR.N.GT.NP) RETURN
27 M1=M+1
28 M2=NP-M1
29 K=L
30 IF (L.LE.M1.OR.N.EQ.NP) GO TO 10
31 K=M1
32 IF (L.LT.M2) GO TO 10
33 K=L-(NP-N)
34 10 MX=L-K
35 DO 20 J=1,N
36 MJ=MX+J
37 X(J)=VARI(MJ)
38 20 Y(J)=VARD(MJ)
39 A=1.0
40 B=0.0
41 C=0.0
42 DO 40 J=1,N
43 IF (J.EQ.K) GO TO 40
44 P=1.0
45 DO 30 I=1,N
46 IF (I.EQ.J) GO TO 30
47 P=P*(X(J)-X(I))
48 30 CONTINUE
49 T=X(K)-X(J)
50 B=B+Y(J)/(P*T)
51 A=A*T
52 C=C+1.0/T
53 40 CONTINUE
54 DIF=A*B+Y(K)*C
55 RETURN
56 END

```

B.11 Mapping Functions (map.f)

This subroutine calculates coordinate mapping functions, converting between physical (x, y) coordinates and transformed (ξ, η, ζ) computational coordinates used in the finite-difference scheme. *OCR note: this listing was replaced with a cleaner in-repo transcription and is currently the most readable available version.*

```

1 SUBROUTINE MAP(IP ,L,M,AL,BE,DE,LD1,AL1,BE1,DE1)           MAP  10
2 C                                                               MAP  20
3 C ****
4 C THIS SUBROUTINE CALCULATES THE MAPPING FUNCTIONS          MAP  30
5 C ****
6 C
7 C ****
8 C
9 COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81,MAP  90
10 1,21),QPT(81,21)
11 COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)                  MAP 110
12 COMMON /SOLUTIN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2) MAP 120
13 COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCOV,FTD,GAMMA,RGAS,GAMI,GAMAP 130
14 1M2,L1,L2,L3,M1,M2,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,IJET,JFLAG,MAP 140
15 2IERR,IUI,IUO,DXR,DYR,LD,MD,LMND,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TCMAP 150
16 3,LC,PLOW,ROLOW                                           MAP 160
17 COMMON /GEMIRYC/ NGEOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGE,XW(81),MAP 170
18 1XW(81),XWI(81),YWI(81),NXNY(81),NPPTS,IINT,IDIF,LT,NNDIM      MAP 180
19 COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,RECB,RCICB,RCTCB,ANGICBMAP 190
20 2,ANGECE,XCB(81),YCB(81),XCB(81),YCB(81),NXNYCB(81),NCBPTS,IINTCEMAP 200
21 3,IDIIFCB,LECB                                           MAP 210
22 COMMON /BCC/ PT(21),TT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,NMAP 220
23 1STAG                                              MAP 230
24 REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE                 MAP 240
25 C
26 BE=1.0/(YW(L)-YCB(L))                                     MAP 250
27 IF (IP,EQ,0) RETURN                                         MAP 260
28 Y=(M-1)*DY                                             MAP 270
29 AL=BE*(NXNYCB(L)+Y*(NXNY(L)-NXNYCB(L)))               MAP 280
30 DE=-BE*Y*XWI(L)                                         MAP 290
31 IF (IP,EQ,1) RETURN                                         MAP 300
32 BE1=1.0/(YW(LD1)-YCB(LD1))                            MAP 310
33 AL1=BE1*(NXNYCB(LD1)+Y*(NXNY(LD1)-NXNYCB(LD1)))    MAP 320
34 DE1=-BE1*Y*XWI(LD1)                                     MAP 330
35 RETURN                                              MAP 340
36 END                                                 MAP 350

```

B.12 Artificial Viscosity (shock.f)

This subroutine calculates local artificial viscosity terms used to stabilize the solution across shock waves and steep gradients. Following the method of Lax and Wendroff, it provides controlled numerical dissipation to prevent oscillations. *OCR note: this listing was reconstructed from Appendix C PDF text blocks for readability and aligned to Appendix C line blocks in the viscosity-energy term region.*

```

1 SUBROUTINE SHOCK(IPASS)
2 C
3 C ****
4 C THIS SUBROUTINE CALCULATES THE LOCAL ARTIFICIAL VISCOSITY TERMS
5 C FOR SHOCK COMPUTATIONS
6 C ****
7 C
8 C OCR note:
9 C Reconstructed from Appendix C PDF text blocks and aligned to
10 C Appendix C line blocks for the viscosity -energy term structure.
11 C
12 COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81,21),QPT(81,21)

```

```

13      COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)
14      COMMON /SOLUTN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)
15      COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCOMV,FDT,GAMMA,RGAS,GAM1,GAM2,
16      1L1,L2,L3,M1,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
17      2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC,
18      3LC,PLOW,ROLOW
19      COMMON /GEMIRYC/ NGEOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGE,XW(81),
20      1YW(81),XWI(81),YWI(81),NXNY(81),NWPTS,IINT,1DIF,LT,NDIM
21      COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,REC,B,RCICB,RCTCB,ANGICB,
22      1ANGE,NCB(81),YCB(81),XCBI(81),YCF(81),NXNYCB(81),NCBPTS,IINTCB,
23      2IDIFCB,LECB
24      COMMON /BOC/ PT(21),TT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,
25      1NSTAG
26      REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE
27 C
28      GO TO (10,160), IPASS
29 C
30 C   CALCULATE LOCAL ARTIFICIAL VISCOSITY FOR SHOCK COMPUTATIONS
31 C
32 10      IF (N.NE.1) GO TO 30
33      NC=0
34      RG=RGAS*G
35      CTA1=1.0-CTA
36      DO 20 L=1,IMAX
37      DO 20 M=1,MMAX
38      QUT(L,M)=0.0
39      QVT(L,M)=0.0
40      QPT(L,M)=0.0
41 20      CONTINUE
42 30      RDUM=CAV*DT*DX*DY*2.0
43      NC=NC+1
44      NLINE=0
45      IF (NC.NE.NPRINT) GO TO 40
46      PRINT 200
47      PRINT 190, N, PC
48 C
49 40      DO 150 L=LSS,L1
50      DO 150 M=1,MMAX
51      LMD2=LD*(M-1)+LMD1
52      LMMD2=LD*(M-2)+LMD1
53      LMPD2=LD*M+LMD1
54      LMN1=L+LMD2
55      LP=L+1+LMD2
56      LM=L-1+LMD2
57      MP=L+LMPD2
58      MM=L+LMMD2
59      LPMP=L+1+LMPD2
60      IPMM=L+1+LMMD2
61      LMMP=L-1+LMPD2
62      IMMM=L-1+LMMD2
63      CALL MAP (1,L,M,AL,BE,DE,LD1,AL1,BE1,DE1)
64 C
65 C   CHECK TO SEE IF THE DIVERGENCE OF THE VELOCITY IS NEGATIVE
66 C
67      UX=0.5*(U(LP)-U(LM))*DXR
68      IF (M.EQ.1) GO TO 50
69      IF (M.EQ.MMAX) GO TO 60
70      UY=0.5*(U(MP)-U(MM))*DYR
71      VY=0.5*(V(MP)-V(MM))*DYR
72      GO TO 70
73 50      UV=(U(MP)-U(LMN1))*DYR
74      VV=(V(MP)-V(LMN1))*DYR
75      GO TO 70
76 60      UV=(U(LMN1)-U(MM))*DYR
77      VV=(V(LMN1)-V(MM))*DYR
78 70      DIV=UX+AL*UY+BE*VV
79      IF (DIV.LT.0.0) GO TO 80
80      QUT(L,M)=0.0
81      QVT(L,M)=0.0
82      QPT(L,M)=0.0
83      GO TO 150
84 C
85 80      OQUT=QUT(L,M)
86      OQVT=QVT(L,M)
87      OQPT=QPT(L,M)
88

```

```

89      UX1=(U(LMN1)-U(LM))*DXR
90      UX2=(U(LP)-U(LMN1))*DXR
91      VX1=(V(LMN1)-V(LM))*DXR
92      VX2=(V(LP)-V(LMN1))*DXR
93      TM=P(LM)/(RO(LM)*RG)
94      T=P(LMN1)/(RO(LMN1)*RG)
95      TP=P(LP)/(RO(LP)*RG)
96      TX1=(T-TM)*DXR
97      TX2=(TP-T)*DXR
98
99      LDUM=L-1
100     CALL MAP (1,LDUM,M,ALM,BEM,DE,LD1,AL1,BE1,DE1)
101     LDUM=L+1
102     CALL MAP (1,LDUM,M,ALP,BEP,DE,LD1,AL1,BE1,DE1)
103     BE1=0.5*(BEM+BE)
104     BE2=0.5*(BEP+BE)
105     AL1=0.5*(ALM+AL)
106     AL2=0.5*(ALP+AL)
107
108     IF (M.EQ.1) GO TO 90
109     IF (M.EQ.MMAX) GO TO 100
110 C
111 C   CALCULATE THE INTERIOR POINT QUANTITIES
112 C
113     UY1=0.25*(U(MP)+U(LMMP)-U(MM)-U(LMM)) *DYR
114     UY2=0.25*(U(MP)+U(LPMP)-U(MM)-U(LPMM)) *DYR
115     VY1=0.25*(V(MP)+V(LMMP)-V(MM)-V(LMM)) *DYR
116     VY2=0.25*(V(MP)+V(LPMP)-V(MM)-V(LPMM)) *DYR
117     UX3=0.25*(U(LP)+U(LPMM)-U(LM)-U(LMM)) *DXR
118     UX4=0.25*(U(LP)+U(LPMP)-U(LM)-U(LMMP)) *DXR
119     VX3=0.25*(V(LP)+V(LPMM)-V(LM)-V(LMM)) *DXR
120     VX4=0.25*(V(LP)+V(LPMP)-V(LM)-V(LMMP)) *DXR
121     VY3=(V(LMN1)-V(MM)) *DYR
122     VY4=(V(MP)-V(LMN1)) *DYR
123     UY3=(U(LMN1)-U(MM)) *DYR
124     UY4=(U(MP)-U(LMN1)) *DYR
125
126     TMY=P(MM)/(RO(MM)*RG)
127     TPY=P(MP)/(RO(MP)*RG)
128     TMM=P(LMM)/(RO(LMM)*RG)
129     TMP=P(LMMP)/(RO(LMMP)*RG)
130     TPM=P(LPMM)/(RO(LPMM)*RG)
131     TPP=P(LPMP)/(RO(LPMP)*RG)
132     TY1=0.25*(TPY+TMP-TMY-TMM) *DYR
133     TY2=0.25*(TPP+TPY-TPM-TMY) *DYR
134     TX3=0.25*(TP+TPM-TM-TMM) *DXR
135     TX4=0.25*(TPP+TP-TPM-TM) *DXR
136     TY3=(T-TMY) *DYR
137     TY4=(TPY-T) *DYR
138
139     MDUM=M-1
140     CALL MAP (1,L,MDUM,ALMY,BEMY,DE,LD1,AL1,BE1,DE1)
141     MDUM=M+1
142     CALL MAP (1,L,MDUM,ALPY,BEPY,DE,LD1,AL1,BE1,DE1)
143     BE3=0.5*(BEMY+BE)
144     BE4=0.5*(BEPY+BE)
145     AL3=0.5*(ALMY+AL)
146     AL4=0.5*(ALPY+AL)
147     GO TO 110
148 C
149 C   CALCULATE THE CENTERLINE POINT QUANTITIES
150 C
151 90     UY1=0.5*(U(MP)+U(LMMP)-U(LMN1)-U(LM)) *DYR
152     VY1=0.5*(V(MP)+V(LMMP)-V(LMN1)-V(LM)) *DYR
153     UY2=0.5*(U(LPMP)+U(MP)-U(LP)-U(LMN1)) *DYR
154     VY2=0.5*(V(LPMP)+V(MP)-V(LP)-V(LMN1)) *DYR
155     UX3=0.0
156     VX3=0.0
157     UX4=0.0
158     VX4=0.0
159     THEW=ATAN(-NXNYCB(L))
160     THE=ATAN(V(MP)/U(MP))
161     VMAG=SQRT(U(MP)*U(MP)+V(MP)*V(MP))
162     RTHE=2.0*THEW-THE
163     UR=VMAG*COS(RTHE)
164     VR=VMAG*SIN(RTHE)

```

```

165      UY3=(U(LMN1)-UR)*DYR
166      VV3=(V(LMN1)-VR)*DYR
167      UY4=(U(MP)-U(LMN1))*DYR
168      VV4=(V(MP)-V(LMN1))*DYR
169      TPY=P(MP)/(RO(MP)*RG)
170      TMP=P(LMP)/(RO(LMP)*RG)
171      TPP=P(LPMP)/(RO(LPMP)*RG)
172      TY1=0.0
173      TY2=0.0
174      TX3=0.25*(TPP+TP-TMP-TM)*DXR
175      TX4=TX3
176      TY4=(TPY-T)*DYR
177      TY3=-TY4
178      BE3=BE
179      AL3=AL
180      MDUM=M-1
181      CALL MAP (1,L,MDUM,AL4,BE4,DE,LD1,AL1,BE1,DE1)
182      GO TO 110
183 C
184 C      CALCULATE THE WALL POINT QUANTITIES
185 C
186 100   UY1=0.5*(U(LMN1)+U(LM)-U(MM)-U(LMM))*DYR
187      VY1=0.5*(V(LMN1)+V(LM)-V(MM)-V(LMM))*DYR
188      UV2=0.5*(U(LP)+U(LMN1)-U(LPM)-U(MM))*DYR
189      VY2=0.5*(V(LP)+V(LMN1)-V(LPM)-V(MM))*DYR
190      UX3=0.0
191      VX3=0.0
192      UX4=0.0
193      VX4=0.0
194      UY3=(U(LMN1)-U(MM))*DYR
195      VY3=(V(LMN1)-V(MM))*DYR
196      THEW=ATAN(-NXNY(L))
197      THE=ATAN(V(MM)/U(MM))
198      VMAG=SQRT(U(MM)*U(MM)+V(MM)*V(MM))
199      RTHE=2.0*THEW-THE
200      UR=VMAG*COS(RTHE)
201      VR=VMAG*SIN(RTHE)
202      UY4=(UR-U(LMN1))*DYR
203      VV4=(VR-V(LMN1))*DYR
204      TPM=P(LPM)/((RO(LPM)*RG)
205      TMY=P(MM)/(RO(MM)*RG)
206      TMM=P(LMM)/(RO(LMM)*RG)
207      TY1=0.0
208      TY2=0.0
209      TX3=0.25*(TP+TPM-TM-TMM)*DXR
210      TX4=TX3
211      TY3=(T-TMY)*DYR
212      TY4=-TY3
213      MDUM=M-1
214      CALL MAP (1,L,MDUM,AL3,BE3,DE,LD1,AL1,BE1,DE1)
215      BE4=BE
216      AL4=AL
217 C
218 110   UXY1=UX1+AL1*UY1
219      UXY2=UX2+AL2*UY2
220      UXY3=UX3+AL3*UY3
221      UXY4=UX4+AL4*UY4
222      UXY12=0.5*(UX1+UX2+AL3*UY3+AL4*UY4)
223
224      VXY1=VX1+AL1*VY1
225      VXY2=VX2+AL2*VY2
226      VXY3=VX3+AL3*VY3
227      VXY4=VX4+AL4*VY4
228      VXY12=0.5*(VX1+VX2+AL3*VY3+AL4*VY4)
229
230      BUY1=BE1*UY1
231      BUY2=BE2*UY2
232      BUY3=BE3*UY3
233      BUY4=BE4*UY4
234      BUY34=0.5*(BUY3+BUY4)
235
236      BVY1=BE1*VY1
237      BVY2=BE2*VY2
238      BVY3=BE3*VY3
239      BVY4=BE4*VY4
240      BVY34=0.5*(BVY3+BVY4)

```

```

241      TXY1=TX1+AL1*TY1
242      TXY2=TX2+AL2*TY2
243      TXY3=TX3+AL3*TY3
244      TXY4=TX4+AL4*TY4
245
246 C
247 C   CALCULATE THE ARTIFICAL VISCOSITY COEFFICIENTS
248 C
249      DIV=UXY12+BVY34
250      VID=VXY12-BUY34
251      RLA=XLA*RDUM*ABS(DIV)/BE
252      RMU=XMU*RDUM*ABS(VID)/BE
253      RK=RMU*GAM1*RG/RKMU
254      RLP2M=RLA+2.0*RMU
255      RLA2=2.0*RLA
256      RMU2=2.0*RMU
257      RLPM=RLA+RMU
258      UVTA=0.0
259      VVTA=0.0
260      PVTA=0.0
261      PCTA=0.0
262
263      IF (NDIM.EQ.0) GO TO 130
264 C
265 C   CALCULATE THE AXISYMMETRIC TERMS
266 C
267      IF (M.EQ.1.AND.YCB(L).EQ.0.0) GO TO 120
268      Y=FLOAT(M-1)*DY/BE+YCB(L)
269      VB=V(LMN1)
270      UVTA=(RLPM*VXY12+RMU*BUY34)/Y
271      VVTA=RLP2M*(BVY34*VB/Y)/Y
272      PVTA=(RLP2M*VB*VB/Y+RLA2*VB*(BVY34+UXY12))/Y
273      PCTA=RK*0.5*(BE4*TY4+BE3*TY3)/Y
274      DUVTA=RLPM*BE*(VXY4-VXY3)*DYL+RMU*BE*(BUY4-BUY3)*DYL
275      DVVTA=RLP2M*0.5*BE*(BVY4-BVY3)*DYL
276      DPVTA=(RLP2M*RLA2)*BVY34*BVY34+RLA2*BVY34*UXY12
277      DPCTA=RK*BE*(BE4*TY4-BE3*TY3)*DYL
278      IF (ABS(UVTA).GT.ABS(DUVTA)) UVTA=DUVTA
279      IF (ABS(VVTA).GT.ABS(DVVTA)) VVTA=DVVTA
280      IF (ABS(PVTA).GT.ABS(DPVTA)) PVTA=DPVTA
281      IF (ABS(PCTA).GT.ABS(DPCTA)) PCTA=DPCTA
282      GO TO 130
283 120  UVTA=RLPM*BE*(VXY4-VXY3)*DYL+RMU*BE*(BUY4-BUY3)*DYL
284      VVTA=RLP2M*0.5*BE*(BVY4-BVY3)*DYL
285      PVTA=(RLP2M+RLA2)*BVY34*BVY34+RLA2*BVY34*UXY12
286      PCTA=RK*BE*(BE4*TY4-BE3*TY3)*DYL
287 C
288 C   CALCULATE THE ARTIFICAL VISCOSITY TERMS
289 C
290 130  QUT(L,M)=RLP2M*(UXY2-UXY1)*DXR
291      1 + AL*(RLP2M*(UXY4-UXY3)+RLA*(BVY4-BVY3))*DYL
292      2 + RMU*BE*(VXY4-BUY4-VXY3+BUY3)*DYL + UVTA
293
294      QVT(L,M)=RMU*(VXY2-BUY2-VXY1+BUY1)*DXR
295      1 + RMU*AL*(VXY4-BUY4-VXY3+BUY3)*DYL
296      2 + BE*(RLA*(UXY4-UXY3)+RLP2M*(BVY4-BVY3))*DYL + VVTA
297
298      QPT(L,M)=RO(LMN1)*(GAMMA-1.0)*
299      1 (RLP2M*(UXY12*UXY12+BVY34*BVY34))
300      2 + RMU*(VXY12*VXY12+BUY34*BUY34)
301      3 + RLA2*UXY12*BVY34 + RMU2*BUY34*VXY12
302      4 + RK*((TXY2-TXY1)*DXR + AL*(TXY4-TXY3)*DYL
303      5 + BE*(BE4*TY4-BE3*TY3)*DYL) + PVTA + PCTA)
304
305      QUT(L,M)=CTA*QUT(L,M)+CTA1*OQUT
306      QVT(L,M)=CTA*QVT(L,M)+CTA1*OQVT
307      QPT(L,M)=CTA*QPT(L,M)+CTA1*OQPT
308
309      IF (NC.NE.NPRINT) GO TO 150
310      NLINE=NLINE+1
311      IF (NLINE.LT.55) GO TO 140
312      PRINT 200
313      PRINT 190, N, PC
314      NLINE=1
315 140  QPT(L,M)=QPT(L,M)/PC
316      PRINT 180, L,M,QUT(L,M),QVT(L,M),QPT(L,M)

```

```

317      QPT(L,M)=QPT(L,M)*PC
318 150  CONTINUE
319
320  IF (NC.EQ.NPRINT) NC=0
321  RETURN
322 C
323 C   SMOOTH THE FLOW VARIABLES IF REQUESTED
324 C
325 160  IF (SMP.LT.0.0.OR.SMP.GE.1.0) RETURN
326  SMP4=0.25*(1.0-SMP)
327  DO 170 L=2,L1
328  U(L,MMAX,N3)=SMP4*(U(L-1,MMAX,N3)+U(L+1,MMAX,N3)+2.0*U(L,M1,N3))
329  1 +SMP*U(L,MMAX,N3)
330  V(L,MMAX,N3)=-U(L,MMAX,N3)*NXNY(L)
331  P(L,MMAX,N3)=SMP4*(P(L-1,MMAX,N3)+P(L+1,MMAX,N3)+2.0*P(L,M1,N3))
332  1 +SMP*P(L,MMAX,N3)
333  RO(L,MMAX,N3)=SMP4*(RO(L-1,MMAX,N3)+RO(L+1,MMAX,N3)+2.0*RO(L,M1,N3))
334  1 +SMP*RO(L,MMAX,N3)
335
336  U(L,1,N3)=SMP4*(U(L-1,1,N3)+U(L+1,1,N3)+2.0*U(L,2,N3))
337  1 +SMP*U(L,1,N3)
338  V(L,1,N3)=-U(L,1,N3)*NXNYCB(L)
339  P(L,1,N3)=SMP4*(P(L-1,1,N3)+P(L+1,1,N3)+2.0*P(L,2,N3))
340  1 +SMP*P(L,1,N3)
341  RO(L,1,N3)=SMP4*(RO(L-1,1,N3)+RO(L+1,1,N3)+2.0*RO(L,2,N3))
342  1 +SMP*RO(L,1,N3)
343
344  DO 170 M=2,M1
345  LMD2=LD*(M-1)+LMD3
346  LMMD2=LD*(M-2)+LMD3
347  LMFD2=LD*M*LMID3
348  LMN3=L+LMID2
349  LP=L+1+LMID2
350  LM=L-1+LMID2
351  MP=L+LMFD2
352  MM=L+LMMD2
353  U(LMN3)=SMP4*(U(LM)+U(LP)+U(MM)+U(MP))+SMP*U(LMN3)
354  V(LMN3)=SMP4*(V(LM)+V(LP)+V(MM)+V(MP))+SMP*V(LMN3)
355  P(LMN3)=SMP4*(P(LM)+P(LP)+P(MM)+P(MP))+SMP*P(LMN3)
356  RO(LMN3)=SMP4*(RO(LM)+RO(LP)+RO(MM)+RO(MP))+SMP*RO(LMN3)
357 170  CONTINUE
358
359  RETURN
360 180  FORMAT (1H ,5X,2I5,3F18.8)
361 190  FORMAT (1H0,63HLOCAL ARTIFICIAL VISCOSITY PARAMETERS FOR SHOCK CALC,
362 1ULATIONS, N=,I4,5H (PC=,F5.1,1H) ,/,/,10X,1HL,4X,1HM,10X,3HQUT,
363 2 11X,3HQVT,11X,3HQPT,/)
364 200  FORMAT (1H1)
365  END

```

B.13 Exit Boundary Conditions (exit.f)

This subroutine calculates boundary mesh points at the nozzle exit. It implements the exit boundary condition calculation based on prescribed exit plane conditions (pressure, or characteristic relations for subsonic flow). *OCR note: this listing was reconstructed from Appendix C PDF text blocks for readability and line-aligned to the degraded Appendix C compatibility-equation block.*

```

1  SUBROUTINE EXITT
2 C
3  ****
4 C THIS SUBROUTINE CALCULATES THE BOUNDARY MESH POINTS AT THE
5 C NOZZLE EXIT FOR SUBSONIC FLOW
6 C ****
7 C
8 C OCR note:
9 C This file was reconstructed from Appendix C PDF text blocks.
10 C Compatibility-equation updates were line-aligned to Appendix C
11 C degraded text blocks and normalized to fixed-form syntax.
12 C

```

```

13 COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKM,UQUT(81,21),QVT(81,21),QPT(81,21)
14 COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)
15 COMMON /SOLUTIN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)
16 COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TOCONV,FDT,GAMMA,ARGAS,GAMI,GAM2,
17 1L1,L2,L3,M1,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
18 2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC,
19 3LC,PLOW,ROLOW
20 COMMON /GEMIRYC/ NGEOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGE,XW(81),
21 1XW(81),XWI(81),XNNY(81),NWPTS,IINT,1DIF,LT,NNDIM
22 COMMON /NGCB/ XICB,RICB,XTCB,RTCB,XECB,REC8,RCICB,RCTCB,ANGICB,
23 1ANGECB,XCB(81),YCB(81),XCEI(81),YCEI(81),XNNYCB(81),NCBPTS,IINTCB,
24 2IDIFCB,LECB
25 COMMON /BCC/ PT(21),TT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,
26 1NSTAG
27 REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE
28 C
29 X3=XE
30 ATERM1=0.0
31 ATERM2=0.0
32 ATERM3=0.0
33 C
34 DO 160 ICHAR=1,2
35 DO 160 M=1,MMAX
36 CALL MAP (2,LMAX,M,AL,BE,DE,L1,AL1,BE1,DE1)
37 U1=U(LMAX,M,N1)
38 U2=U1
39 A1=SQRT(GAMMA*P(LMAX,M,N1)/RO(LMAX,M,N1))
40 A2=A1
41 IF (ICHAR.EQ.2) GO TO 10
42 U(LMAX,M,N3)=U1
43 A3=A1
44 C
45 C CALCULATE THE PROPERTY INTERPOLATING POLYNOMIAL COEFFICIENTS
46 C
47 10 BU=(U(LMAX,M,N1)-U(L1,M,N1))*DXR
48 BV=(V(LMAX,M,N1)-V(L1,M,N1))*DXR
49 BP=(P(LMAX,M,N1)-P(L1,M,N1))*DXR
50 BRO=(RO(LMAX,M,N1)-RO(L1,M,N1))*DXR
51 BYCB=(YCB(LMAX)-YCB(L1))*DXR
52 BAL=(AL-AL1)*DXR
53 BBE=(BE-BE1)*DXR
54 BDE=(DE-DE1)*DXR
55 CU=U(LMAX,M,N1)-BU*X3
56 CV=V(LMAX,M,N1)-BV*X3
57 CP=P(LMAX,M,N1)-BP*X3
58 CRO=RO(LMAX,M,N1)-BRO*X3
59 CYCB=YCB(LMAX)-BYCB*X3
60 CAL=AL-BAL*X3
61 CBE=BE-BBE*X3
62 CDE=DE-BDE*X3
63 C
64 C CALCULATE THE CROSS DERIVATIVE INTERPOLATING POLYNOMIAL
65 C COEFFICIENTS
66 C
67 IF (M.EQ.1) GO TO 20
68 DU=(U(LMAX,M,N1)-U(LMAX,M-1,N1))*DYR
69 DV=(V(LMAX,M,N1)-V(LMAX,M-1,N1))*DYR
70 DP=(P(LMAX,M,N1)-P(LMAX,M-1,N1))*DYR
71 DRO=(RO(LMAX,M,N1)-RO(LMAX,M-1,N1))*DYR
72 DU1=(U(L1,M,N1)-U(L1,M-1,N1))*DYR
73 DV1=(V(L1,M,N1)-V(L1,M-1,N1))*DYR
74 DP1=(P(L1,M,N1)-P(L1,M-1,N1))*DYR
75 DRO1=(RO(L1,M,N1)-RO(L1,M-1,N1))*DYR
76 GO TO 80
77 20 IF (LECB.EQ.LMAX) GO TO 30
78 DU=0.0
79 DV=V(LMAX,2,N1)*DYR
80 DP=0.0
81 DRO=0.0
82 DU1=0.0
83 DV1=V(L1,2,N1)*DYR
84 DP1=0.0
85 DRO1=0.0
86 GO TO 80
87 30 DU=(U(LMAX,2,N1)-U(LMAX,1,N1))*DYR
88 DV=(V(LMAX,2,N1)-V(LMAX,1,N1))*DYR

```

```

89      DP=(P(LMAX,2,N1)-P(LMAX,1,N1))*DYR
90      DRO=(RO(LMAX,2,N1)-RO(LMAX,1,N1))*DYR
91      DU1=(U(L1,2,N1)-U(L1,1,N1))*DYR
92      DV1=(V(L1,2,N1)-V(L1,1,N1))*DYR
93      DP1=(P(L1,2,N1)-P(L1,1,N1))*DYR
94      DRO1=(RO(L1,2,N1)-RO(L1,1,N1))*DYR
95  80     BDU=(DU-DU1)*DXR
96     BDV=(DV-DV1)*DXR
97     BDP=(DP-DP1)*DXR
98     BDRO=(DRO-DRO1)*DXR
99     CDU=DU-BDU*X3
100    CDV=DV-BDV*X3
101    CDP=DP-BDP*X3
102    CDRO=DRO-BDRO*X3
103  C
104  C      CALCULATE X1 AND X2
105  C
106  IF (ICHAR.EQ.2) A3=SQRT(GAMMA*P(LMAX,M,N3)/RO(LMAX,M,N3))
107  DO 50 IL=1,2
108  X1=X3-(U(LMAX,M,N3)-U1)*0.5*DT
109  X2=X3-(U(LMAX,M,N3)-A3+U2-A2)*0.5*DT
110  C
111  C      INTERPOLATE FOR THE PROPERTIES
112  C
113  U1=BU*X1+CU
114  U2=BU*X2+CU
115  P2=BP*X2+CP
116  RO2=BRO*X2+CRO
117  A2=SQRT(GAMMA*P2/RO2)
118  50   CONTINUE
119  V1=BV*X1+CV
120  P1=BP*X1+CP
121  RO1=BRO*X1+CRO
122  YCB1=YCB*X1+CYCB
123  AL1=BAL*X1+CAL
124  BE1=BBE*X1+CBE
125  DE1=BDE*X1+CDE
126  UV1=U1*AL1+V1*BE1+DE1
127  A1=SQRT(GAMMA*P1/RO1)
128  V2=BV*X2+CV
129  YCB2=YCB*X2+CYCB
130  AL2=BAL*X2+CAL
131  BE2=BBE*X2+CBE
132  DE2=BDE*X2+CDE
133  UV2=U2*AL2+V2*BE2+DE2
134  C
135  C      INTERPOLATE FOR THE CROSS DERIVATIVES
136  C
137  DU1=BDU*X1+CDU
138  DV1=BDV*X1+CDV
139  DP1=BDP*X1+CDP
140  DRO1=BDRO*X1+CDRO
141  DU2=BDU*X2+CDU
142  DV2=BDV*X2+CDV
143  DP2=BDP*X2+CDP
144  DRO2=BDRO*X2+CDRO
145  C
146  C      CALCULATE THE PSI TERMS
147  C
148  IF (NDIM.EQ.0) GO TO 70
149  IF (M.EQ.1 .AND. LECB.NE.LMAX) GO TO 60
150  ATERM1=RO1*V1/((DY*(M-1))/BE1+YCB1)
151  ATERM2=RO2*V2/((DY*(M-1))/BE2+YCB2)
152  GO TO 70
153  60  ATERM1=RO1*BE1*DV1
154  ATERM2=RO2*BE2*DV2
155  70  PSI31=-UV1*DV1-BE1*DP1/RO1
156  PSI41=-UV1*DP1+A1*A1*UV1*DRO1
157  PSI12=-UV2*DRO2-RO2*AL2+DU2-RO2*BE2*DV2-ATERM2
158  PSI22=-UV2*DU2-AL2*DP2/RO2
159  PSI42=-UV2*DP2+A2*A2*UV2*DRO2
160  IF (ICHAR.EQ.1) GO TO 130
161  C
162  C      CALCULATE THE CROSS DERIVATIVES AT THE SOLUTION POINT
163  C
164  IF (M.EQ.1 .AND. LECB.NE.LMAX) GO TO 88

```

```

165      IF (M.EQ.MMAX) GO TO 98
166      DU3=(U(LMAX,M+1,N3)-U(LMAX,M,N3))*DYL
167      DV3=(V(LMAX,M+1,N3)-V(LMAX,M,N3))*DYL
168      DP3=(P(LMAX,M+1,N3)-P(LMAX,M,N3))*DYL
169      DRO3=(RO(LMAX,M+1,N3)-RO(LMAX,M,N3))*DYL
170      GO TO 100
171 88   DU3=0.0
172      DV3=V(LMAX, 2 ,N3)*DYL
173      DP3=0.0
174      DRO3=0.0
175      GO TO 100
176 98   DU3=(U(LMAX,MMAX,N3)-U(LMAX,M1,N3))*DYL
177      DV3=(V(LMAX,MMAX,N3)-V(LMAX,M1,N3))*DYL
178      DP3=(P(LMAX,MMAX,N3)-P(LMAX,M1,N3))*DYL
179      DRO3=(RO(LMAX,MMAX,N3)-RO(LMAX,M1,N3))*DYL
180 C
181 C      CALCULATE THE PSI TERMS AT THE SOLUTION POINT
182 C
183 100  IF (NDIM.EQ.0) GO TO 120
184      IF (M.EQ.1 .AND.LECB.NE.LMAX) GO TO 110
185      ATERM3=RO(LMAX,M,N3)*V(LMAX,M,N3)/( (DV*(M-1))/BE+YCB(LMAX) )
186      GO TO 120
187 110  ATERM3=RO(LMAX, 1 ,N3)*BE*DVL
188 120  UV3=U(LMAX,M,N3)*AL+V(LMAX,M,N3)*BE+DE
189      PSI13=-UV3*DRO3-RO(LMAX,M,N3)*(AL*DU3+BE*DVL)-ATERM3
190      PSI23=-UV3*DU3-AL*DP3/RO(LMAX,M,N3)
191      PSI33=-UV3*DV3-BE*DP3/RO(LMAX,M,N3)
192      PSI43=-UV3*DP3+A3*A3*UV3*DRO3
193 C
194 C      CALCULATE THE COMPATIBILITY EQUATION COEFFICIENTS
195 C
196 130  IF (ICHAR.EQ.1) GO TO 140
197      PSI31B=(PSI31+PSI33)*0.5
198      PSI41B=(PSI41+PSI43)*0.5
199      PSI12B=(PSI12+PSI13)*0.5
200      PSI22B=(PSI22+PSI23)*0.5
201      PSI42B=(PSI42+PSI43)*0.5
202      GO TO 150
203 140  PSI31B=PSI31
204      PSI41B=PSI41
205      PSI12B=PSI12
206      PSI22B=PSI22
207      PSI42B=PSI42
208 C
209 C      SOLVE THE COMPATIBILITY EQUATIONS FOR U, V, P, AND RO
210 C
211 150  P(LMAX,M,N3)=PE
212      RO(LMAX,M,N3)=RO1+(P(LMAX,M,N3)-P1-DT*PSI41B)/(A3*A3+A1*A1)
213      U(LMAX,M,N3)=U2+((PSI42B+(RO2-RO(LMAX,M,N3))*(A2+A3)*PSI22B/4.0
214      1 -(A2*A2+A3*A3)*PSI12B/2.0)*DT-(P(LMAX,M,N3)-P2))
215      2 /(RO2+RO(LMAX,M,N3))/(A2+A3))*4.0
216      V(LMAX,M,N3)=V1+DT*PSI31B
217 160  CONTINUE
218 C
219      V(LMAX,MMAX,N3)=-U(LMAX,MMAX,N3)*NNXY(LMAX)
220      V(LMAX, 1 ,N3)=-U(LMAX, 1 ,N3)*NNYCB(LMAX)
221      IF (JFLAG.EQ.0) RETURN
222      V(LMAX,MMAX,N3)=V(LMAX,MMAX,N3)+XWI(LMAX)
223      RETURN
224      END

```

B.14 Visualization Output (plot.f)

This subroutine generates visualization output including velocity vector plots and contour plots of dependent variables (pressure, density, entropy). It creates plotter data for external visualization tools. *OCR note: this listing was reconstructed from Appendix C PDF text blocks for readability; contour-cell intersection logic was restored from degraded scan lines with normalized fixed-form token syntax.*

```

1 SUBROUTINE PLOT(TITLE,T,NP)
2 C
3 C ****
4 C THIS SUBROUTINE PLOTS THE VELOCITY VECTORS AND DEPENDENT
5 C VARIABLE CONTOUR PLOTS
6 C ****
7 C
8 C OCR note:
9 C Reconstructed from Appendix C PDF text blocks. Contour-cell
10 C intersection logic (labels 240-320) was restored from degraded
11 C scan blocks with normalized fixed-form token syntax.
12 C
13 DIMENSION CQ(81,21),CON(9),XCO(4),YCO(4),TITLE(8)
14 COMMON /AV/ IAV,CAV,NST,SMP,LSS,CTA,XMU,XLA,RKMU,QUT(81,21),QVT(81,21),QPT(81,21)
15 COMMON /ONESID/ UD(4),VD(4),PD(4),ROD(4)
16 COMMON /SOLUTIN/ U(81,21,2),V(81,21,2),P(81,21,2),RO(81,21,2)
17 COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCOINV,FDT,GAMMARGAS,GAMI,GAM2,
18 1L1,L2,L3,M1,M2,DX,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
19 2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC,
20 3LC,PLOW,ROLOW
21 COMMON /GEMIRYC/ NGEMOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGE,XW(81),
22 1XW(81),XWI(81),NNXY(81),NWPTS,IINT,1DIF,LT,NDIM
23 COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,REC,B,RCICB,RCTCB,ANGICB,
24 1ANCECB,XCB(81),YCB(81),XCBI(81),YCB(81),NNNYCB(81),NCBPTS,IINTCB,
25 2IDIFCB,LECB
26 COMMON /BCC/ PT(21),TT(21),THETA(21),PE,MASSE,MASSI,MASST,THRUST,
27 1NSTAG
28 REAL MN3,NXNY,MASSI,MASST,NXNYCB,MASSE
29 C
30 C GENERATE THE VELOCITY VECTOR PLOT
31 C
32 ND=N3
33 IF (N.EQ.0) ND=1
34 CALL GETG (4HKJBN,JNM)
35 XL=XI
36 XR=XE
37 YT=YW(1)
38 YB=YCB(1)
39 DO 10 L=2,IMAX
40 YT=AMAX1(YT,YW(L))
41 YB=AMIN1(YB,YCB(L))
42 10 CONTINUE
43
44 VV=0.9*DX
45 FIYB=916.0
46 XD=(XR-XL)/(YT-YB)
47 FIR=(1022.0-1022.0/FLOAT(L1)-1.0)/900.0
48 IF (XD.LE.FIR) GO TO 20
49 FIXL=0.0
50 FIXR=1022.0-1022.0/FLOAT(L1)-1.0
51 FIYT=916.0-FIXR/XD
52 GO TO 30
53 20 FIXL=511.0-450.0*XD
54 FIXR=511.0+450.0*XD
55 FIYT=16.0
56 30 XCONV=(FIXR-FIXL)/(XR-XL)
57 YCONV=(FIYT-FIYB)/(YT-YB)
58
59 VMAX=0.0
60 DO 40 L=1,IMAX
61 DO 40 M=1,MMAX
62 VMAX=AMAX1(VMAX,ABS(U(L,M,ND)),ABS(V(L,M,ND)))
63 40 CONTINUE
64 IF (VMAX.LT.1.0E-10) GO TO 60
65
66 DROU=VV/VMAX
67 CALL ADV (1)
68 DO 50 L=1,IMAX
69 IX1=FIXL+(FLOAT(L-1)*DX)*XCONV
70 DYL=(YW(L)-YCB(L))/FLOAT(MMAX-1)
71 DO 50 M=1,MMAX
72 IY1=FIYB+(YCB(L)+FLOAT(M-1)*DYL-YB)*YCONV
73 IX2=FIXL+(FLOAT(L-1)*DX+U(L,M,ND)*DROU)*XCONV
74 IY2=FIYB+(YCB(L)+FLOAT(M-1)*DYL-YB+V(L,M,ND)*DROU)*YCONV
75 CALL DRV (IX1,IY1,IX2,IY2)

```

```

76      CALL PLT (IX1,IY1,16)
77 50  CONTINUE
78      CALL LINCNT (59)
79      WRITE (7,410)
80      WRITE (7,550) JNM,TITLE,np,t
81 C
82 C   GENERATE THE CONTOUR PLOTS
83 C
84 60  I=0
85 70  I=I+1
86      GO TO (80,100,120,140,340), I
87
88 80  DO 90 L=1,IMAX
89      DO 90 M=1,MMAX
90      CQ(L,M)=RO(L,M,ND)*G
91 90  CONTINUE
92      GO TO 160
93
94 100 DO 110 L=1,IMAX
95      DO 110 M=1,MMAX
96      CQ(L,M)=P(L,M,ND)/PC
97 110 CONTINUE
98      GO TO 160
99
100 120 DO 130 L=1,IMAX
101      DO 130 M=1,MMAX
102      CQ(L,M)=P(L,M,ND)/RO(L,M,ND)/RGAS/G-TC
103 130 CONTINUE
104      GO TO 160
105
106 140 DO 150 L=1,IMAX
107      DO 150 M=1,MMAX
108      CQ(L,M)=SQRT(U(L,M,ND)**2+V(L,M,ND)**2)/(GAMMA*P(L,M,ND)/RO(L,M,ND))
109 150 CONTINUE
110
111 160 QMN=1.0E06
112      QMX=-QMN
113      DO 170 L=1,IMAX
114      DO 170 M=1,MMAX
115      QMN=AMIN1(CQ(L,M),QMN)
116      QMX=AMAX1(CQ(L,M),QMX)
117 170 CONTINUE
118      XX=QMX-QMN
119      DQ=0.1*XX
120      DO 180 K=1,9
121      CON(K)=QMN+FLOAT(K)*DQ
122 180 CONTINUE
123      K=9
124
125      CALL ADV (1)
126      CALL LINCNT (59)
127      GO TO (190,200,210,220), I
128 190 WRITE (7,560)
129      GO TO 230
130 200 WRITE (7,570)
131      GO TO 230
132 210 WRITE (7,580)
133      GO TO 230
134 220 WRITE (7,590)
135 230 WRITE (7,400) QMN,QMX,CON(1),CON(K),DQ
136      WRITE (7,550) JNM,TITLE,np,t
137 C
138      DO 520 L=2,IMAX
139      DX=(YW(L-1)-YCB(L-1))/FLOAT(MMAX-1)
140      DY1=(YW(L)-YCB(L))/FLOAT(MMAX-1)
141      DO 520 M=2,MMAX
142      NN=0
143      DO 520 KK=1,K
144      K1=0
145      K2=0
146      K3=0
147      K4=0
148      IF (CQ(L-1,M-1).LE.CON(KK)) K1=1
149      IF (CQ(L,M-1).LE.CON(KK)) K2=1
150      IF (CQ(L-1,M).LE.CON(KK)) K3=1
151      IF (CQ(L,M).LE.CON(KK)) K4=1

```

```

152      IF (K1*K2*K3*K4.NE.0) GO TO 520
153      IF (K1+K2+K3+K4.EQ.0) GO TO 520
154      IF (NN.NE.0) GO TO 240
155      NN=1
156      XCO(1)=XI+FLOAT(L-2)*DX
157      XCO(2)=XCO(1)+DX
158      XCO(3)=XCO(1)
159      XCO(4)=XCO(2)
160      YCO(1)=YCB(L-1)+FLOAT(M-2)*DY
161      YCO(2)=YCB(L)+FLOAT(M-2)*DY1
162      YCO(3)=YCB(L-1)+FLOAT(M-1)*DY
163      YCO(4)=YCB(L)+FLOAT(M-1)*DY1
164 240    LL=0
165      IF (K1+K3.NE.1) GO TO 250
166      IC1=1
167      IC2=3
168      LP1=L-1
169      MP1=M-1
170      LP2=L-1
171      MP2=M
172      ASSIGN 250 TO KR1
173      GO TO 280
174 250    IF (K1+K2.NE.1) GO TO 260
175      IC1=1
176      IC2=2
177      LP1=L-1
178      MP1=M-1
179      LP2=L
180      MP2=M-1
181      ASSIGN 260 TO KR1
182      GO TO 280
183 260    IF (K2+K4.NE.1) GO TO 270
184      IC1=2
185      IC2=4
186      LP1=L
187      MP1=M-1
188      LP2=L
189      MP2=M
190      ASSIGN 270 TO KR1
191      GO TO 280
192 270    IF (K3+K4.NE.1) GO TO 520
193      IC1=3
194      IC2=4
195      LP1=L-1
196      MP1=M
197      LP2=L
198      MP2=M
199      ASSIGN 520 TO KR1
200 280    LL=LL+1
201      XX=(CON(KQ)-CQ(LP1,MP1))/(CQ(LP2,MP2)-CQ(LP1,MP1))
202      IF (LL.EQ.2) GO TO 290
203      IX1=FIXL+(XCO(IC1)+XX*(XCO(IC2)-XCO(IC1))-XL)*XCONV
204      IY1=FIYB+(YCO(IC1)+XX*(YCO(IC2)-YCO(IC1))-YB)*YCONV
205      GO TO KR1,(250,260,270,320)
206 290    IX2=FIXL+(XCO(IC1)+XX*(XCO(IC2)-XCO(IC1))-XL)*XCONV
207      IY2=FIYB+(YCO(IC1)+XX*(YCO(IC2)-YCO(IC1))-YB)*YCONV
208      CALL DRV (IX1,IY1,IX2,IY2)
209      IF (KK.NE.1) GO TO 300
210      CALL PLT (IX1,IY1,35)
211 300    IF (KK.NE.K) GO TO 310
212      CALL PLT (IX1,IY1,24)
213 310    LL=0
214      IF (LP2.NE.L) GO TO 520
215      IF (MP2.NE.M) GO TO 320
216      GO TO 260
217 320    CONTINUE
218    CONTINUE
219 520    CONTINUE
220
221    DO 330 L=2,IMAX
222    IX1=FIXL+(FLOAT(L-2)*DX)*XCONV
223    IX2=FIXL+(FLOAT(L-1)*DX)*XCONV
224    IY1=FIYB+(YCB(L-1)-YB)*YCONV
225    IY2=FIYB+(YCB(L)-YB)*YCONV
226    IY3=FIYB+(YW(L-1)-YB)*YCONV
227    IY4=FIYB+(YW(L)-YB)*YCONV

```

```
228      CALL DRV (IX1,IY1,IX2,IY2)
229      CALL DRV (IX1,IY3,IX2,IY4)
230 330  CONTINUE
231      GO TO 70
232
233 340  DY=1.0/FLOAT(MMAX-1)
234      CALL ADV (1)
235      RETURN
236
237 400  FORMAT (1H ,10HLOW VALUE=,1PE11.4,2X,11HHIGH VALUE=,E11.4,2X,
238    1 12HLOW CONTOUR=,E11.4,2X,13HHIGH CONTOUR=,E11.4,2X,
239    2 14HDELTA CONTOUR=,E11.4)
240 410  FORMAT (1H ,16HVELOCITY VECTORS)
241 550  FORMAT (1H ,A10,4X,8A10,2X,2HN=,I4 ,2X,2HT=,1PE10.4 ,4H SEC)
242 560  FORMAT (1H ,7HDENSITY)
243 570  FORMAT (1H ,8HPRESSURE)
244 580  FORMAT (1H ,11HTEMPERATURE)
245 590  FORMAT (1H ,11HMACH NUMBER)
246      END
```

Appendix A

Characteristic Relations: η Constant Plane

Introduction

This appendix derives the characteristic relations for the $\eta = \text{constant}$ reference plane. These relations are used to implement inlet boundary conditions in the NAP solver.

I. Equations of Motion

The equations of motion can be written in the form:

$$\frac{\partial P}{\partial t} + u \frac{\partial P}{\partial \xi} + v \frac{\partial P}{\partial \eta} = -vP_\eta - pau_\xi - pBv_\xi - \frac{epvB}{n} \quad (\text{A.1})$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial \xi} + \frac{P_\xi}{p} = -vu_\eta - ap_\xi/p \quad (\text{A.2})$$

$$\frac{\partial v}{\partial t} + uv_\xi + \frac{BP_\eta}{p} = -vv_\eta - \frac{eP_\eta}{p} \quad (\text{A.3})$$

$$\frac{\partial P}{\partial t} + up_\xi - a^2 \left(\frac{\partial P}{\partial t} + u \frac{\partial P}{\partial \xi} \right) = -vP_\eta + a^2 vp_\eta \quad (\text{A.4})$$

II. Characteristic Curves

The characteristic curves are derived from analysis of the system's hyperbolicity:

$$\frac{d\eta}{dT} = v \quad (\text{A.5})$$

$$\frac{d\eta}{dx} = \frac{\partial \eta}{\partial x} \quad (\text{A.6})$$

III. Characteristic Variables

Define the characteristic variables:

$$\psi_1 = vP_\xi - pau_\xi - pBv_\xi - \frac{epvB}{n} \quad (\text{A.7})$$

$$\psi_2 = -vu_\xi - aP_\xi/p \quad (\text{A.8})$$

$$\psi_3 = -vv_\xi - \frac{eP_\xi}{p} \quad (\text{A.9})$$

$$\psi_4 = -vP_\xi + a^2vP_\xi \quad (\text{A.10})$$

IV. Compatibility Equations

Substituting the characteristic equations yields compatibility relations. For the characteristic curve with slope $d\eta = (u - a)dT$:

$$dp - \rho adu = (\psi_2 + 2\psi_1 - \rho a\psi_2)dT \quad (\text{A.11})$$

For the characteristic curve with slope $d\eta = (u + a)dT$:

$$dp + \rho adu = (\psi_4 + 2\psi_1 + \rho a\psi_2)dT \quad (\text{A.12})$$

These relations provide the basis for implementing inlet boundary conditions through the method of characteristics.

Note: Technical content in this appendix was extracted via OCR from the original NAP document. Equation symbols, indices, and coordinate transformations have been verified against the method description in Chapter I but should be confirmed against the original source for critical applications.

Appendix B

Characteristic Relations: ζ Constant Plane

Introduction

This appendix derives the characteristic relations for the $\zeta = \text{constant}$ reference plane. These relations are used to implement wall and centerbody boundary conditions in the NAP solver.

I. Equations of Motion

The equations of motion for the ζ -plane are:

$$\frac{\partial P}{\partial t} + v \frac{\partial P}{\partial \eta} + \rho a u_\eta + \rho B v_\eta = -u P_\xi - \rho u_\xi - \frac{e p v B}{n} \quad (\text{B.1})$$

$$\frac{\partial u}{\partial t} + v u_\eta + \frac{a P_\eta}{\rho} = -u u_\xi - \frac{P_\xi}{\rho} \quad (\text{B.2})$$

$$\frac{\partial v}{\partial t} + v v_\eta + \frac{B P_\eta}{\rho} = -u v_\xi \quad (\text{B.3})$$

$$\frac{\partial P}{\partial t} + v P_\eta - a^2 \left(\frac{\partial P}{\partial t} + v \frac{\partial P}{\partial \eta} \right) = -u P_\xi + a^2 u P_\xi \quad (\text{B.4})$$

II. Characteristic Curves

Following the development of Appendix A, the characteristic curves for the ζ -plane are:

$$\frac{d\zeta}{dT} = v \quad (\text{B.5})$$

$$\frac{d\zeta}{dx} = v \pm a^* a \quad (\text{B.6})$$

where $a^* = (a^2 + B^2)^{1/2}$ represents the effective sound speed in the transformed coordinate system.

III. Compatibility Equations

The compatibility equations for the ζ -constant plane are:

$$adu - \rho dv = (\psi_0 - a\psi_1)dT \quad (B.7)$$

$$dp - a^2 d\rho = \psi_4 d\xi \quad (B.8)$$

$$dp - \rho a^2 \frac{du}{a^*} - \rho Ba \frac{dv}{a^*} = \left(\psi_2 + a\psi_1 - \frac{\rho aa\psi_0}{a^*} - \frac{\rho Ba\psi_1}{a^*} \right) dT \quad (B.9)$$

$$dp + \rho a^2 \frac{du}{a^*} + \rho Ba \frac{dv}{a^*} = \left(\psi_3 + a\psi_1 + \frac{\rho aa\psi_0}{a^*} + \frac{\rho Ba\psi_1}{a^*} \right) dT \quad (B.10)$$

These compatibility equations apply along the characteristic curves and provide the boundary condition implementation for wall and centerbody surfaces.

Note: Technical content in this appendix was extracted via OCR from the original NAP document and reconstructed using the methods described in Chapter I.E. Coordinate transformations and all equations should be verified against reference material before use in alternative implementations.