

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	23
A Characteristic Relations	24
A.1 $\eta = \text{constant}$ Reference Plane	24

A.2	$\zeta = \text{constant}$ Reference Plane	24
B	Fortran Code Listing (LASL Identification: LP-0537)	25
B.1	Main Program (fortran_main.f)	25
B.2	Geometry Subroutine (geom.f)	33
B.3	Inlet Boundary Conditions (inlet.f)	34
B.4	Wall Boundary Conditions (wall.f)	37
B.5	Interior Mesh Calculations (inter.f)	42
B.6	Mass Flow Calculations (masflo.f)	44
B.7	One-Dimensional Initialization (onedim.f)	45
A	Characteristic Relations: η Constant Plane	48
B	Characteristic Relations: ζ Constant Plane	50

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

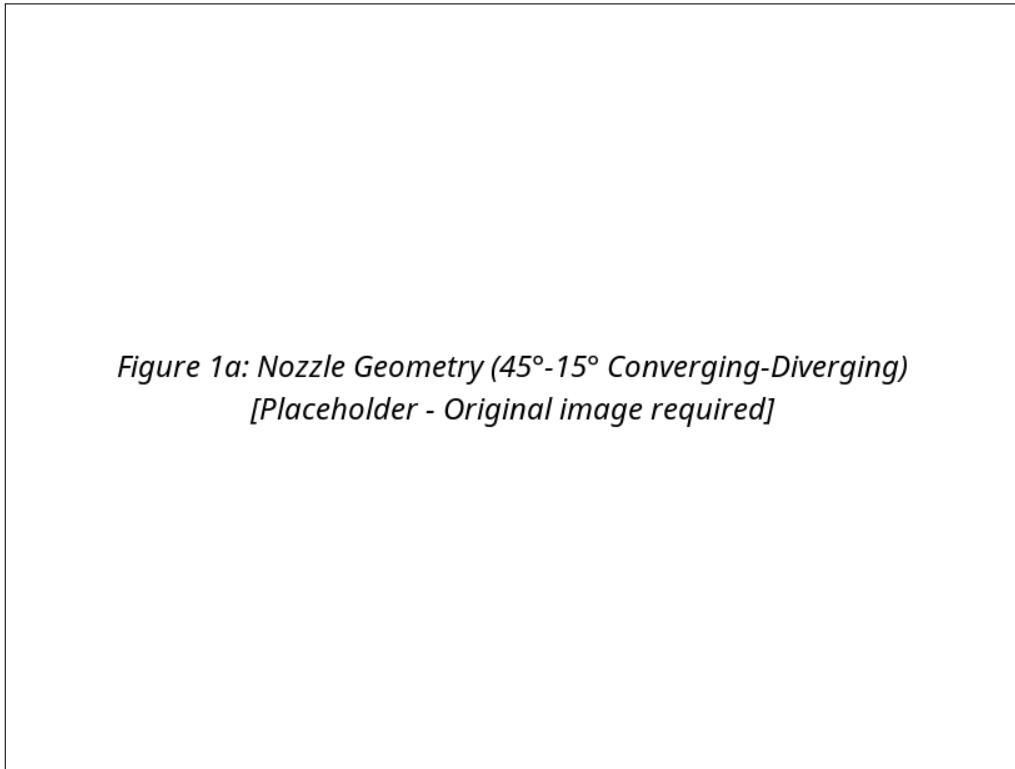
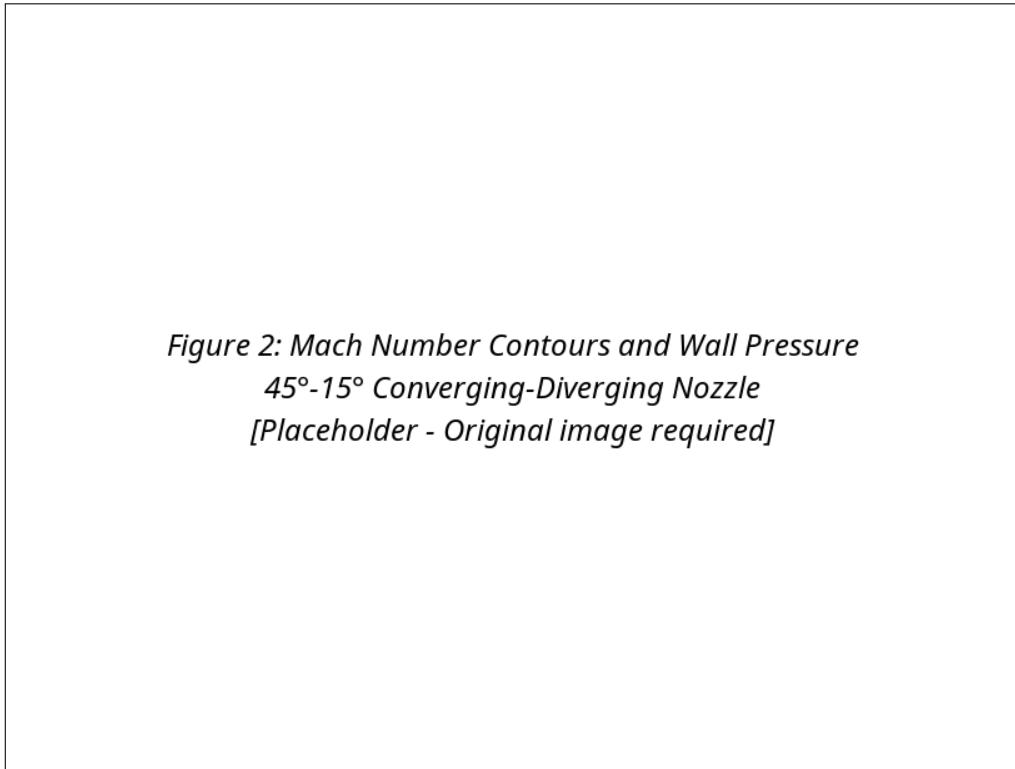


Figure 1a: Nozzle Geometry (45°-15° Converging-Diverging)
[Placeholder - Original image required]

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.

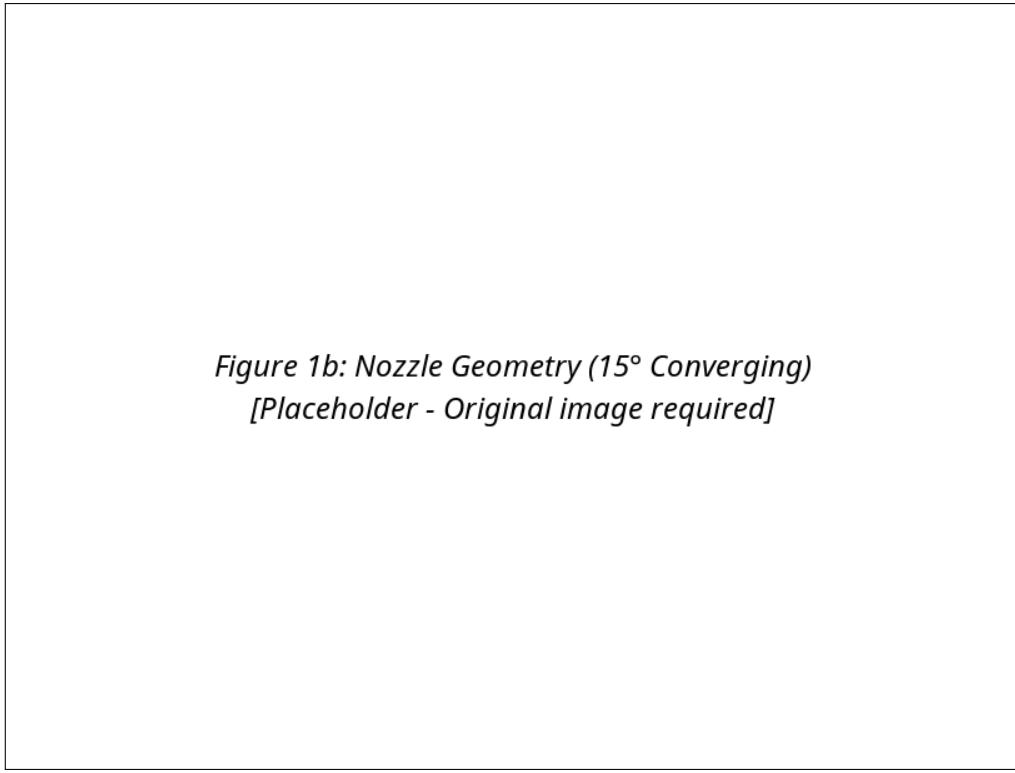


*Figure 2: Mach Number Contours and Wall Pressure
45°-15° Converging-Diverging Nozzle
[Placeholder - Original image required]*

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.

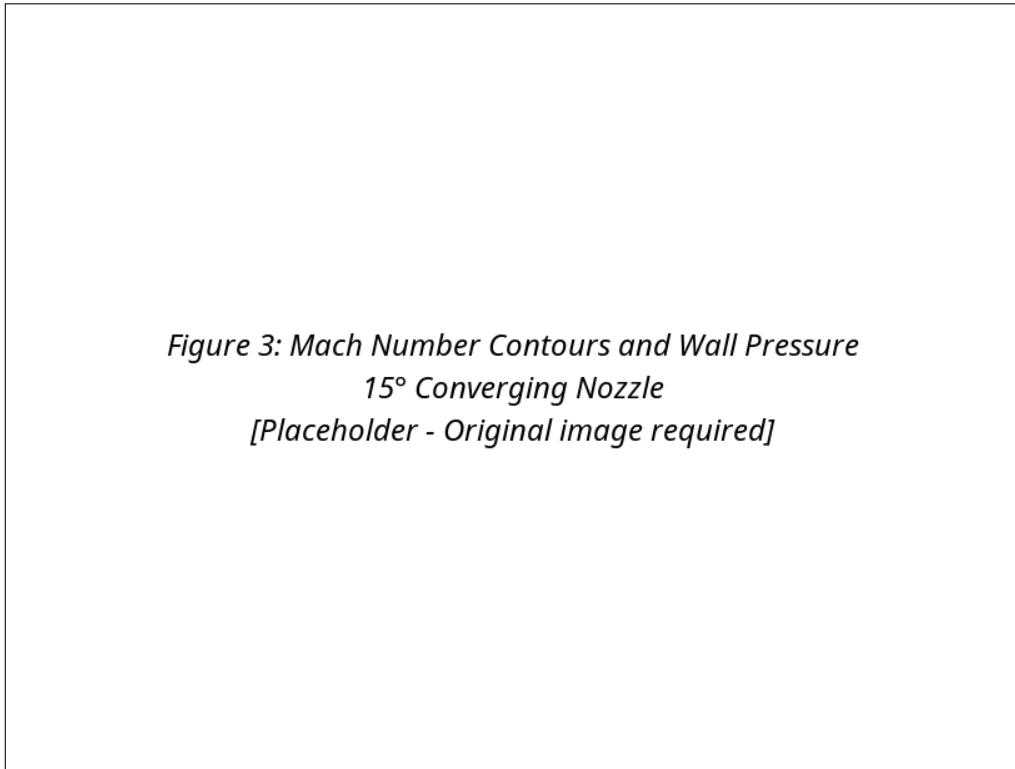
1.7.2 Case 2: 15° Conical Converging Nozzle



*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.

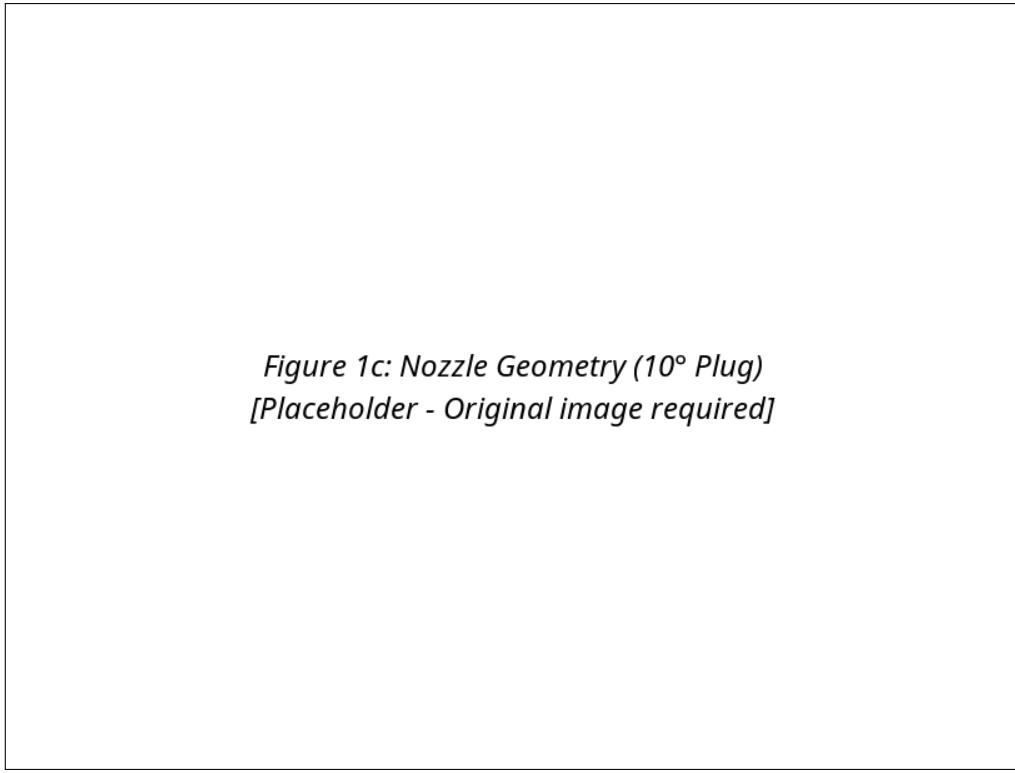


*Figure 3: Mach Number Contours and Wall Pressure
15° Converging Nozzle
[Placeholder - Original image required]*

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).

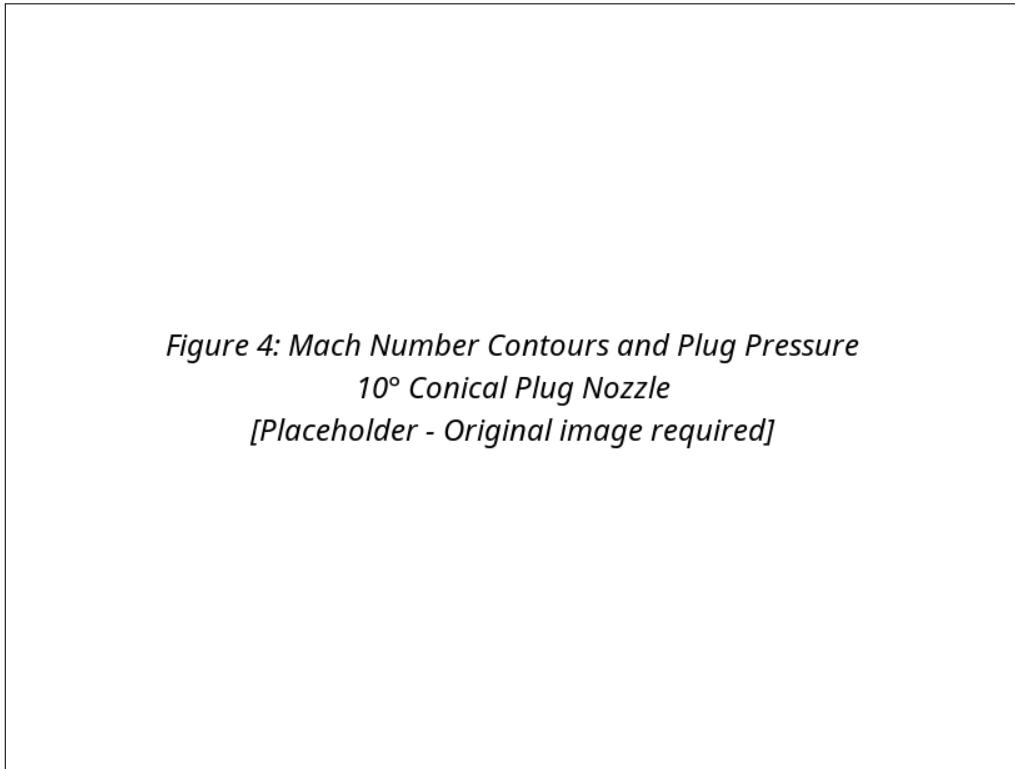
1.7.3 Case 3: 10° Conical Plug Nozzle



*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.

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.

```
1 PROGRAM MAIN(INPUT,OUTPUT,FILM,PUNCH,TAPE5=INPUT,TAPE6=OUTPUT,
2 1TAPE7=FILM)
3 C
4 C ****
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),R0(81,21,2)
29 COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCONV,FDT,GAMMA,RGAS,GAM1,GA
30 1M2,L1,L2,L3,M1,M2,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
31 2IERR,IUI,IUO,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC
32 3,LC,PLOW,ROLOW
33 COMMON /GEMTRYC/ NGEOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGE,XW(81),
34 1YW(81),XWI(81),YWI(81),NXNY(81),NWPTS,IINT,IDL,LT,NDIM
35 COMMON /GCB/ NGCB,XICB,RICB,XTCB,XECB,RECB,RCICB,RCTCB,ANGICB
36 1,ANGEBC,XCB(81),YCB(81),XCBI(81),YCFI(81),NXNYCB(81),NCBPTS,IINTCB
37 2,IDLFCB,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      1 NASM,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,N1D,NSTART,TSTART,RSTAR,RSTARS
45      NAMELIST /GEMTRY/ NDIM,XI,RI,RT,XE,RCI,RCT,ANGI,ANCE,NGEOM,XWI,YWI
46      1,NWPTS ,IINT ,IDIF ,LJET ,JFLAG ,NXNY ,YW
47      NAMELIST /GCB/ NGCB ,RICB ,RCTB ,RCICB ,RCTCB ,ANGICB ,ANGECB ,YCB ,NXNYC
48      1B ,XCB ,YCB ,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      N1D=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,GCB)
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,GCB)
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,
     LMAX ,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, GAMMA ,RGAS
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 GEOM
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,LMAX
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,E0,0) Y=YW(L)-YCB(L)

```

```

117      IF (NOIM,E0,1) 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,E0,2) 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=LC=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,EG,0) GO TO 250
195      DO 240 M=1,MMAX
196      U(1,M,1)=UI(M)
197      V(1,M,I)=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 MASFL0 (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=YCBL(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,XHACH,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 (IU0,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      PO(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          DXDY=DXRS+BE*BE*DYRS
322          DO 380 M=1,MMAX
323              LMN1=L+LD*(M-1)+LMD1
324              QS=U(LMN1)*U(LMN1)+V(LMN1)*V(LMN1)
325              AS=GAMMA*P(LMN1)/RO(LMN1)
326              UPA=SQRT(QS*DXDY)+SQRT(AS*DXDY)
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(U(L1,M,N3)-RO(L2,M,N3))
371      IF (P(LMAX,M,N1),GT,0.0,AND,RO(LMAX,M,N3),GT,0.0) GO TO 420 'MrNI^CT.B.B) 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)*NNNY(LMAX)
376      V(LMAX,1,3)=-U(LMAX,1,N3)*NNNYCB(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 EXITT
385      IF (N,LE,NST) CALL SHOCK (2)
386      C
387      C      DETERMINE THE MAXIMUM (DELTA U)/U
388      C
389      IF (TCOV,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 43
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=NCP+1
399      IF (DDQM,GE,TCOV) GO TO 450
400      NCONV=NCONV+1
401      IF (NCONV,EQ,1) NCHECK=N-1
402      IF (NCONV,GE,NCONVI) NC=NPRINT
403 450      IF (N,EQ,NMAX) NC=NPRINT
404      IF (N,GE,NCHECK+NCONVI) 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 (JFIAG,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=N|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,IJMAX
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,H,N3)**2+V(L,MN3)**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,H,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=RRHO*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 85, 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*0.4535
481      MASSI=MASSI*0.4535
482      MASSE=MASSE*0.4535
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      TIM=T/LC $ NP=N|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,8HTHE EQUATIONS OF MOTION FOR TWO-DIMENSIONAL, TIME DEPENDENT,
INVISCID FLOW IN A NOZZLE,/,21X,9HARE SOLVED USING THE SECOND-ORDER, MACCORMACK, FINITE-DIFFERENCE SCHEME, THE
FLUID IS ASSUMED,/,21X,9HTO BE A PERFECT GAS, ALL BOUNDARY CONDITIONS ARE COMPUTED USING A SECOND-ORDER,
REFERENCE PLANE,/,21X,9HCHARACTERISTIC SCHEME, THE STEADY STATE SOLUTION IS OBTAINED AS THE ASYMPTOTIC SOLUTION
FOR,/,21X,9HARGE 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,5HIMAX=,I2 ,2X,5HMMAX=,I2 ,3X,5HNMAX=,I4 ,2X,7HNPRINT=,I4 ,2X,6HICONV=,F6. 3 ,3X,4HFDT=,F4. 2 ,2X,6HNSTAG=,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,6HHTETA=,F9.4 ,6H
(DEG),5X,3HPE=,F9.4 ,7H (PSIA))
550 770  FORMAT (1HO,10X,21HBOUN0ARY CONDITIONS --,/,21X,3HPT=,F9.4 ,6H (KPA),5X,3HTT=,F9.4 ,4H (C),5X,6HHTETA=,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,3HFRHO,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,6HMASSI=,F9.4 ,5X,6HMASSE=,F9.4 )
561 880  FORMAT (1HO,10X,5HMASS=,F9.4 ,9H (KS/SEC) ,5X,7HTHRUST=,F11.4 ,10H (NEWTONS) ,5X,6HMASSI=,F9.4 ,5X,6HMASSE=,F9.4 )
562 890  FORMAT (1HO,10X,21HBOUNDARY CONDITIONS

```

```

563   900  -,//,22X,1HM 11X,8HPT(P8IA),10X,5HTT(F),10X,10HTHETA(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,10HTHETA(DEC),10X,3HPE=,F7.3,6H (KPA),/
566   930  FORMAT (1H0,78H***** THE RADIUS OF THE CENERBOOY IS LARGER THAN THE NOZZLE WALL RADIUS *****)*
567   940  FORMAT (1X,18HSIVS NID=0,NSTART=,I4,8H,TSTART=,F14.10,1H,)
568   950  FORMAT (1X,4H(1,,I2,5H,1) =)
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=,14)
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(81GEO	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),R0(81,21,2)	GEO	120
13	COMMON /CNTRL/ LMAX,MMAX,NMAX,NPRINT,TCQNV,FDT,GAMMA,RGAS,GAM1,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	1YW(81),XWI(81),YWI(81),NXNY(81),NWPTS,IINT,IDL,LT,NDIM	GEO	180
19	COMMON /GCB/ NGCB,XICB,RICB,XTCB,XECB,RECB,RCICB,RCTCB,ANGICBGE0	190	
20	2,ANGECB,XCB(81),YCB(81),XCBI(81),YCBI(81),NXNYCB(81),NCBPTS,IINTCBGE0	200	
21	3,IDLFCB,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	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,LMAX	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	GEO	500
51	30 PRINT 230	GEO	510

```

52      IF (RCI, EQ, 0.0, OR, RCT, EQ, 0.0) GO TO 200           GEO 520
53      ANI=ANGI*3.141593/180.0                                GEO 530
54      ANE=ANGE*3.141593/180.0                                GEO 540
55      XTAN=XI+RCI*SIN(ANI)                                 GEO 550
56      RTAN=RI+RCI*(COS(ANI)-1.0)                           GEO 560
57      RT1=RT-RCT*(COS(ANI)-1.0)                           GEO 570
58      XT1=XTAN+(RTAN-RT1)/TAN(ANI)                         GEO 580
59      IF (XT1, GE, XTAN) GO TO 40                           GEO 590
60      XT1=XTAN                                         GEO 600
61      RT1=RTAN                                         GEO 610
62 40     XT=XT1+RCT*SIN(ANI)                                GEO 620
63     XT2=XT+RCT*SIN(ANE)                                GEO 630
64     RT2=RT+RCT*(1.0-COS(ANE))                          GEO 640
65     RE=RT2+(XE-XT2)*TAN(ANE)                           GEO 650
66     LT=1                                              GEO 660
67     DX=(XE-XI)/(LMAX-1)                                GEO 670
68     IF (IUI, EQ, 1) PRINT 270, XI, RI, RT, XE, RCI, RCT, ANGI, ANGE, XT, RE   GEO 680
69     IF (IUI, EQ, 2) PRINT 280, XI, RI, RT, XE, RCI, RCT, ANGI, ANGE, XT, RE   GEO 690
70     DO 110 L=1,LMAX                                     GEO 700
71     XI=XI+(L-1)*DX                                    GEO 710
72     IF (X, GE, XI, AND, X, LE, XTAN) GO TO 50          GEO 720
73     IF (X, GT, XTAN, AND, X, LE, XT1) GO TO 60          GEO 730
74     IF (X, GT, XT1, AND, X, LE, XT) GO TO 70            GEO 740
75     IF (X, GT, XT, AND, X, LE, XT2) GO TO 80            GEO 750
76     IF (X, GT, XT2, AND, X, LE, XE) GO TO 90            GEO 760
77 C
78 50     YW(L)=RI+RC*(COS(ASIN((X-XI)/RCI))-1.0)        GEO 780
79     NXNY(L)=(XI-XI)/(YW(L)-RI+RCI)                      GEO 790
80     GO TO 100                                         GEO 800
81 C
82 60     YW(L)=RT1+(XT1-X)*TAN(ANI)                      GEO 810
83     NXNY(L)=TAN(ANI)                                    GEO 830
84     GO TO 100                                         GEO 840
85 C

```

B.3 Inlet Boundary Conditions (inlet.f)

```

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 1,21), 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, MMAX, NMAX, NPRINT, TCONV, 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, DYL, LD, MD, LMD1, LMD3, IB, RSTAR, RSTARS, NPLOT, G, PC, TC
16 3, LC, PLLOW, ROLOW
17 COMMON /GEMTRYC/ NGEOM, XI, RI, XT, RT, XE, RE, RCI, RCT, ANGI, ANGE, XW(81),
18 1YW(81), XWI(81), YWI(81), NXNY(81), NWPTS, IINT, IDIF, LT, NDIM
19 COMMON /GCB/ NGCB, XICB, RICB, XTCB, RTCB, XECB, RECB, RCICB, RCTCB, ANGICB
20 1, ANGECB, XCB(81), YCB(81), XCBI(81), YCBI(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 LM1N1=2+LD*(M-2)+LMD1

```

```

36      LM1N1=1+LD*(M-2)+LMD1
37      LM1N3=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
46      C
47      C   CALCULATE THE PROPERTY INTERPOLATING POLYNOMIAL COEFFICIENTS
48      C
49      10    BU=(U(L1MN1)-U(LMN1))*DXR
50      BV=(V(L1MN1)-V(LMN1))*DXR
51      BU=(P(L1MN1)-P(LMN1))*DXR
52      BRO=(RO(L1MN1)-(LMN1))*DXR
53      BYCB=(YCB(2)-YCB(1))*DXR
54      BAL=(AL1-AL)*DXR
55      BBE=(BE1-BE)*DXR
56      CU=U(1,M,N1)-BU*X3
57      CV=V(1,M,N1)-BV*X3
58      CP=P(1,M,N1)-B*X3
59      CRO=RO(1,M,N1)-BRO*X3
60      CYCB=YCB(1)-BYCB*X3
61      CAL=AL-BAL*X3
62      CBE=BE-BBE*X3
63
64      C   CALCULATE THE CROSS DERIVATIVE INTERPOLATING POLYNOMIAL
65      COEFFICIENTS
66
67      IF (M,EQ,1) GO TO 20
68      DU=(U(L1MN1)-U(L1M1N1))*DYR
69      DV=(V(L1MN1)-V(L1M1N1))*DYR
70      DP=(P(L1MN1)-P(L1M1N1))*DYR
71      DRO=(RO(L1MN1)-RO(L1M1N1))*DYR
72      DU1=(U(LMN1)-U(LM1N1))*DYR
73      DV1=(V(LMN1)-V(LM1N1))*DYR
74      DP1=(P(LMN1)-P(LM1N1))*DYR
75      DR01=(RO(LMN1)-RO(LM1N1))*DYR
76      GO TO 40
77      20    IF (NGCB,NE,0) GO TO 30
78      DU=0.0
79      DV=V(2,2,N1)*DYR
80      DP=0.0
81      DRO=0.0
82      DU1=0.0
83      DV1=V(1,2,N1)*DYR
84      DP1=0.0
85      DR01=0.0
86      GO TO 40
87      30    DU=(U(2,2,N1)-U(2,1,N1))*DYR
88      DV=(V(2,2,N1)-V(2,1,N1))*DYR
89      DP=(P(2,2,N1)-P(2,1,N1))*DYR
90      DRO=(RO(2,2,N1)-RO(2,1,N1))*DYR
91      DU1=(U(1,2,N1)-U(1,1,N1))*DYR
92      DV1=(V(1,2,N1)-V(1,1,N1))*DYR
93      DP1=(P(1,2,N1)-P(1,1,N1))*DYR
94      DR01=(RO(1,2,N1)-RO(1,1,N1))*DYR
95      40    BDU=(DU-DU1)*DXR
96      BDV=(DV-DV1)*DXR
97      BDP=(DP-DP1)*DXR
98      BDR0=(DRO-DR01)*DXR
99      CDU=DU1-BDU*X3
100     CDV=DV1-BDV*X3
101     CDP=DP1-BDP*X3
102     CDR0=DR01-BD*X3
103
104     C   CALCULATE X2
105
106     IF (ICHAR,EQ,2) A3=SQRT(GAMMA*P(LMN3)/RO(LMN3))
107     DO 50 IL=1,2
108     X2=X3-(U(1,M,N3)-A3+U2-A2)*0.5*DT
109
110     C   INTERPOLATE FOR THE PROPERTIES
111     U2=BU*X2+CU

```

```

112      P2=BP*X2+CP
113      R02=BR0*X2+CRO
114      A2=SQRT(GAMMA*P2/R02)
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      DR02=BDR0*X2+CDR0
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=R02*V2/(DY*(M-1)/(BE2+YCB2))
134      GO TO 70
135 60  ATERM2=R02*BE2*DVA
136 70  PSI12=-UV2*DR02-R02*AL2*DU2-R02*BE2*DVA-ATERM2
137      PSI22=-UV2*DU2-AL2*DP2/R02
138      PSI42=-UV2*DP2+A2*A2*UV2*DR02
139      IF (CHAR,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))*DVR
146      DV3=(V(LMN3)-V(LMN3))*DVR
147      DP3=(P(LMN3)-P(LMN3))*DVR
148      DR03=(R0(LMN3)-R0(LMN3))*DVR
149      GO TO 100
150 80  DU3=0.0
151      DV3=V(1,2,N3)*DVR
152      DP3=0.0
153      DR03=0.0
154      GO TO 100
155 90  DU3=(U(1,MMAX,N3)-U(1,M1,N3))*DVR
156      DV3=(V(1,MMAX,N3)-V(1,M1,N3))*DVR
157      DP3=(P(1,MMAX,N3)-P(1,M1,N3))*DVR
158      DR03=(R0(1,MMAX,N3)-(1,M1,N3))*DVR
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=R0(LMN3)*V(LMN3)/(DY*M-1)/BE+YCB(1)
165      GO TO 120
166 110 ATERM3=R0(LMN3)*BE*DVA
167 120 UV3=U(LMN3)*AL+V(LMN3)*BE
168      PSI13=-UV3*DR03-R0(LMN3)*AL*DU3-R0(LMN3)*BE*DVA-ATERM3
169      PSI23=-UV3*DU3-AL*DP3/R0(LMN3)
170      PSI43=-UV3*DP3+A3*A3*UV3*DR03
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 R0
177 C
178 140 MN3=SQRT(U(LMN3)*U(LMN3)+V(LMN3)*V(LMN3))/A3
179      T2=P2/(R02*RGAS*G)
180      PSI1B=(PSI12+PSI13)*0.5
181      PSI2B=(PSI22+PSI23)*0.5
182      PSI4B=(PSI42+PSI43)*0.5
183      GPS1B=GAMMA*PSI1B
184      TTHETA=TAN(THETA(M))
185      UCORR=0.5+0.5/SQRT(1.0+TTHETA*TTHETA)
186 C
187      DO 160 ITER=1,20

```

```

188      DEM=(1.0+GAM2*MN3*MN3)
189      P(LMN3)=PT(M)/(DEM**GAM1)
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)*TTHETA
197      OMN3=MN3
198      MN3=SQRT((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 R0(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)

```

1      SUBROUTINE WALL
2      C
3      ****
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      ****
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 /SOLUTN/ U(81,21,2),V(81,21,2),P(81,21,2),R0(81,21,2)
14     COMMON /CNTRL/ LMAX,MMAX,IMAX,NPRINT,TCONV,FDT,GAMMA,RGAS,GAM1,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     1YW(81),XWI(81),YWI(81),NXNY(81),NWPTS,IINT,IDIF,LT,NDIM
20     COMMON /GCB/ NGCB,XICB,RICB,XTCB,XECB,RECB,RCICB,RCTCB,ANGICB
21     2,ANGECB,XCB(81),YCB(81),XCFI(81),YCFI(81),NXNYCB(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 $ MDUM=MMAX $ NDUM1=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     LMDM1=LD*(MDUM1-1)
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      L1MN1=L-1+LMDM1+LMD1
47      IF (JFLAG.EQ.0) GO TO 50
48      IF (IB.EQ.2) GO TO 50
49      C
50      XWID=WXI(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(L1MN3)=UD(3)
57      V(L1MN3)=VD(3)
58      P(L1MN3)=PD(3)
59      RO(L1MN3)=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(L1MN1)=UD(2)
70      V(L1MN1)=VD(2)
71      P(L1MN1)=PD(2)
72      RO(L1MN1)=ROD(2)
73      C
74      50 U1=U(LMN1)
75      V1=V(LMN1)
76      P1=P(LMN1)
77      R01=RO(LMN1)
78      U2=U1
79      V2=V1
80      A1=SQRT(GAMMA*P1/R01)
81      A2=A1
82      IF (ICHAR.EQ.2) GO TO 60
83      U3=U1
84      V3=V1
85      P3=P1
86      R03=R01
87      A3=A1
88      GO TO 70
89      60 U3=U(LMN3)
90      V3=V(LMN3)
91      P3=P(LMN3)
92      R03=RO(LMN3)
93      A3=SQRT(GAMMA*P3/R03)
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=(R01-R0(LMN1))*DYS
101     CU=U1-BU*Y3
102     CV=V1-BV*Y3
103     CP=P1-BP*Y3
104     CR0=R01-BRO*Y3
105     C
106    C CALCULATE THE CROSS DERIVATIVE INTERPOLATING POLYNOMIAL
107    C COEFFICIENTS
108    C
109    DU=(U1-U(L1MN1))*DXR
110    DV=(V1-V(L1MN1))*DXR
111    DP=(P1-P(L1MN1))*DXR
112    DR0=(R01-R0(L1MN1))*DXR
113    DUI=(U(LMN1)-U(L1MN1))*DXR
114    DV1=(V(LMN1)-V(L1MN1))*DXR
115    DP1=(P(LMN1)-P(L1MN1))*DXR
116    DR01=(R0(LMN1)-R0(L1MN1))*DXR
117    BDU=(DU-DUI)*DYS
118    BDV=(DV-DV1)*DYS
119    BDP=(DP-DP1)*DYS
120    DR0=(DR0-DR01)*DYS

```

```

121      CDU=DU-BDU*Y3
122      CDV=DV-BDV*Y3
123      CDP=DP-BDP*Y3
124      CDRO=DR0-BDRO*Y3
125      C
126      C   CALCULATE Y2
127      C
128      CALL MAP (1,L,MDUM,AL,BEW,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      R02=BR0*Y2+CR0
142      AL2=Y2*AL
143      AD=GAMMA*P2/R02
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      DR01=DR0
157      DU2=BDU*Y2+CDU
158      DV2=BDV*Y2+CDV
159      DP2=BDP*Y2+CDP
160      DR02=BDRO*Y2+CDRO
161      C
162      C   CALCULATE THE PSI TERMS
163      C
164      IF (NDIM.EQ.0) TO TO 110
165      IF (IB.EQ.2) GO TO 100
166      ATERM2=R02*V2/(YCB(L)+Y2/BE)
167      GO TO 110
168 100  ATERM2=R02*V2/(YCB(L)+Y2/BE)
169      IF (IAV.EQ.0) GO TO 110
170      ATDS=R02*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*DVI
175      PSI41=-U1*DP1+A1*A1*U1*DR01
176      PSI12=-U2*DR02-R02*DU2-ATERM2
177      PSI22=-U2*DU2-DP2/RO2
178      PSI32=-U2/DV2
179      PSI42=-U2*DP2+A2*A2*U2*DR02
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 (ILJET.EQ.2) GO TO 120
189      GO TO 130
190 120  DU3=(U(L1MN3)-U3)*DXR
191      DV3=(V(L1MN3)-V3)*DXR
192      DP3=(P(L1MN3)-P3)*DXR
193      DR03=(RO(L1MN3)-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      DR03=(R03-R0(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 (ICAR.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      YWNEW=YW(L)+DYDP*(PE-P(LMN3))
227      YWOLD=YW(L)
228      POLD=P(LMN3)
229      YW(L)=YWNEW
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,BEW,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=R03*V2/(YCB(L)+1.0/BE)
247      GO TO 240
248 220 IF (YCB(L).EQ.0.0) GO TO 230
249      ATERM3=R03*V3/YCB(L)
250      IF (IAV.EQ.0) GO TO 240
251      ATDS=R03*V(L,2,N3)*DYR*BE
252      IF (ABS(ATERM3).GT.ABS(ATDS)) ATERMS=ATDS
253      GO TO 240
254 230 ATERMS=R03*V(L,2,N3)*DYR*BE
255      C
256 240 PSI13=-U3*DR03-R03*DU3-ATERM3
257      PSI23=-U3*DU3-DP3/RP3
258      PSI33=-U3*DVS
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=NXNYCB(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      R01B=(R01+R03)*0.5
269      R02B=(R02+R03)*0.5
270      IF (ICAR.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 R0
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*R02B*A2B*(ALB*(U(LMN3)-U2)+BEB*(V(LMN3)-V2))+(PSI4
294 22B+A2B*A2B+PSI12B+SIGN*R02B*A2B*(ALB*PSI22B+BEB*PSI32B))*DT
295      IF (P(LMN3).LE.0.0) P(LMN3)=PLOW*PC
296      RO(LMN3)=R0(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=TD*DUMD
330      IF (PE.GT.PD(3).AND.XM1.GE.1.0) GO TO 310
331      PTD=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     DNXNY=(DPR+NXNY(LJET)+DPL*NXNY(L))/DPLR
363     U(LMN3)=QA/SQRT(1.0+DNXNY*DNXNY)
364     V(LMN3)=-U(LMN3)*DNXNY
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)-YWI(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 QUARE 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),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,MMAX,IMAX,NPRINT,TCONV,FDT,GAMMA,RGAS,GAM1,GA
14    1M2,L1,L2,L3,M1,M2,DY,DT,N,N1,N3,NASM,IVEL,ICHAR,N1D,LJET,JFLAG,
15    2IERR,IUI,IUD,DXR,DYR,LD,MD,LMD1,LMD3,IB,RSTAR,RSTARS,NPLOT,G,PC,TC
16    3,LC,PLOW,ROLOW
17 COMMON /GEMTRYC/ NGEOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGE,XW(81),
18    1YW(81),XWI(81),YWI(81),NXNY(81),NWPTS,IINT,IDIF,LT,NDIM
19 COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,RECB,RCICB,RCTCB,ANGICB
20    2,ANGECB,XCB(81),YCB(81),XCBI(81),YCBI(81),NXNYCB(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,LMAX
34 DO 30 M=MDUM,M1
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 L1MN1=L-1+LMD2+LMD1
40 LM1N1=L+LD*(M-2)+LMD1
41 UB=U(LMN1)
42 VB=V(LMN1)
43 PB=P(LMN1)
44 ROB=RO(LMN1)
45 ASB=GAMMA*PB/ROB
46 IF (M.NE.1) GO TO 10
47 DUDX=(UB-U(L1MN1))*DXR
48 DPDX=(PB-P(L1MN1))*DXR
49 DRDX=(ROB-RO(L1MN1))*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*DRDX-ROB*DUDX-(1+NDIM)*ROB*BE*DVDY
55 PRHS=-UB*DPDX+ASB*(RORHS+UB*DRDX)
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(L1MN1))*DXR
60 DVDX=(VB-V(L1MN1))*DXR
61 DPDX=(PB-P(L1MN1))*DXR
62 DRDX=(ROB-RO(L1MN1))*DXR
63 DUDY=(UB-U(LM1N1))*DYR
64 DVDY=(VB-V(LM1N1))*DYR
65 DPDY=(PB-P(LM1N1))*DYR
66 DRDY=(ROB-RO(LM1N1))*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*DRDX-UVB*DRDY-ROB*(DUDX+AL*DUDY+BE*DVDY)-ATERM
71 PRHS=-UB*DPDX-UVB*DPDY+ASB*(RORHS+UB*DRDX+UVB*DRDY)
72 V(LMN3)=V(LMN1)+VRHS*dt
73 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,M1
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 L1MN3=L+1+LMD2+LMD3
92 LM1N3=L+LD*M+LMD3
93 UB=U(LMN3)
94 VB=V(LMN3)
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 DRDX=(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)*DYS
116      DVDY=(V(LMN3)-VB)*DYS
117      DPDY=(P(LMN3)-PB)*DYS
118      DRODY=(RO(LMN3)-ROB)*DYS
119 C
120      URHS=-UB*DUDX-UVB*DUDY-(DPDX+AL*DPDY)/ROB
121      VRHS=-UB*DUDX-UVB*DUDY-BE*DPDY/ROB
122      RORHS=-UB*DRODX-UVB*DRODY-ROB*(DUDX+AL*DUDY+BE*DUDY)-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
3 C ****
4 C THIS SUBROUTINE CALCUTES THE INITIAL -DATA OR SOLUTION SURFACE   MAS 30
5 C MASS FLOW AND THRUST                                              MAS 40
6 C
7 C ****
8 C ****
9 C
10 COMMON /AV/ IAV ,CAV ,NST ,SMP ,LSS ,CTA ,XMU ,XLA ,RKMU ,QUT(81,21) ,QVT(81MAS 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 /CNTRL/ LMAX ,MMAX ,NMAX ,NPRINT ,TCONV ,FDT ,GAMMA ,RGAS ,GAM1 ,GAMAS 140
15 1M2 ,L1 ,L2 ,L3 ,M1 ,M2 ,DX ,DY ,DT ,N ,N1 ,N3 ,NASM ,IVEL ,ICHAR ,N1D ,LJET ,JFLAG ,MAS 150
16 2IERR ,IUI ,IUO ,DXR ,DYS ,LD ,MD ,LMD1 ,LMD3 ,IB ,RSTAR ,RSTARS ,NPLOT ,G ,PC ,TCMAS 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 180
19 1YW(81) ,XWI(81) ,YWI(81) ,NXNY(81) ,NWPTS ,IINT ,IDIF ,LT ,NDIM   MAS 190
20 COMMON /GCB/ NGCB ,XICB ,RICB ,XTCB ,RTCGB ,RECB ,RCICB ,RCTCB ,ANGICBMAS 200
21 1 ,ANGEGB ,XCB(81) ,YCB(81) ,XCBI(81) ,YCBI(81) ,NXNYCB(81) ,NCBPTS ,IINTCBMAS 210
22 2 ,IDIFCB ,LECB                                         MAS 220
23 COMMON /BCC/ PT(21) ,TT(21) ,THETA(21) ,PE ,MASSE ,MASSI ,MASST ,THRUST ,NMAS 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=LMAX -1                                         MAS 270
29 IF (LT,EQ,LMAX) LT=LMAX -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
35 C
36 C
37 C
38 C
39 C
40 C
41 C
42 C
43 C
44 C
45 C
46 C
47 C
48 C
49 C
50 C
51 C
52 C
53 C
54 C
55 C
56 C
57 C
58 C
59 C
60 C
61 C
62 C
63 C
64 C
65 C
66 C
67 C
68 C
69 C
70 C
71 C
72 C
73 C
74 C
75 C
76 C
77 C
78 C
79 C
80 C
81 C
82 C
83 C
84 C
85 C
86 C
87 C
88 C
89 C
90 C
91 C
92 C
93 C
94 C
95 C
96 C
97 C
98 C
99 C
100 C
101 C
102 C
103 C
104 C
105 C
106 C
107 C
108 C
109 C
110 C
111 C
112 C
113 C
114 C
115 C
116 C
117 C
118 C
119 C
120 C
121 C
122 C
123 C
124 C
125 C
126 C
127 C
128 C
129 C
130 C
131 C
132 C
133 C
134 C
135 C
136 C
137 C
138 C
139 C
140 C

```

```

34 C     SURFACE                               MAS  340
35 C                                         MAS  350
36 IF (NDIM,EQ,1) GO TO 10                  MAS  360
37 AREA1=(YW(1)-YCB(1))/LC2                MAS  370
38 AREAT=(YW(LT)-YCB(LT))/LC2              MAS  380
39 AREAE=(YW(LDUM)-YCB(LDUM))/LC2          MAS  390
40 GO TO 20                                MAS  400
41 10  AREA1=3.141593*(YW(1)**2-YCB(1)**2)/LC2   MAS  410
42  AREAT=3.141593*(YW(LT)**2-YCB(LT)**2)/LC2   MAS  420
43  AREAE=3.141593*(YW(LDUM)**2-YCB(LDUM)**2)/LC2  MAS  430
44 GO TO 20                                MAS  440
45 20  VMI=$QRT(U(1,1,1)**2+V(1,1,1)**2)      MAS  450
46  VMT=$QRT(U(LT,1,1)**2+V(LT,1,1)**2)       MAS  460
47  VME=$QRT(U(LDUM,1,1)**2+V(LDUM,1,1)**2)    MAS  470
48  MASSI=RO(1,1,1)*VMI*AREAI*G               MAS  480
49  MASST=RO(LT,1,1)*VMT*AREAT*G             MAS  490
50  MASSE=RO(LDUM,1,1)*VME*AREAE*G           MAS  500
51  THRUST=RO(LDUM,1,1)*U(LDUM,1,1)**2*AREAE  MAS  510
52 C                                         MAS  520
53 C     CALCULATE THE MASS FLOW AND THRUST FOR THE 2-D INITIAL -DATA  MAS  530
54 C     SURFACE                               MAS  540
55 C                                         MAS  550
56 30  MASSI=0.0                                MAS  560
57  MASST=0.0                                  MAS  570
58  MASSE=0.0                                  MAS  580
59  THRUST=0.0                                  MAS  590
60  DYI=DY*(YW(1)-YCB(1))                     MAS  600
61  DYT=DY*(YW(LT)-YCB(LT))                   MAS  610
62  DYE=DY*(YW(LDUM)-YCB(LDUM))              MAS  620
63  ND=1                                      MAS  630
64 IF (ISURF,EQ,1) ND=N3                    MAS  640
65 DO 60 M=1,M1                            MAS  650
66  RADI=(M-1)*DYI+YCB(1)                   MAS  660
67  RADT=(M-1)*DYT+YCB(LT)                 MAS  670
68  RADE=(M-1)*DYE+YCB(LDUM)              MAS  680
69 IF (NDIM,EQ,1) GO TO 40                  MAS  690
70  AREA1=DYI/LC2                           MAS  700
71  AREAT=DYT/LC2                           MAS  710
72  AREAE=DYE/LC2                           MAS  720
73 GO TO 50                                MAS  730
74 40  AREA1=3.141593*((RADI+DYI)**2-RADI**2)/LC2  MAS  740
75  AREAT=3.141593*((RADT+DYT)**2-RADT**2)/LC2  MAS  750
76  AREAE=3.141593*((RADE+DYE)**2-RADE**2)/LC2  MAS  760
77  ROUI=(RO(1,M,ND)*U(1,M,ND)+RO(1,M+1,ND)*U(1,M+1,ND))*0.5  MAS  770
78  ROUT=(RO(LT,M,ND)*U(LT,M,ND)+RO(LT,M+1,ND)*U(LT,M+1,ND))*0.5  MAS  780
79  ROUE=(RO(LDUM,M,ND)*U(LDUM,M,ND)+RO(LDUM,M+1,ND)*U(LDUM,M+1,ND))*0MAS  790
80  1.5                                     MAS  800
81  ROUE2=(RO(LDUM,M,ND)*U(LDUM,M,ND)**2+RO(LDUM,M+1,ND)*U(LDUM,M+1,ND)*MAS  810
82  1)**2)*0.5                             MAS  820
83  MASSI=MASSI+ROUI*AREAI*G               MAS  830
84  MASST=MASST+ROUT*AREAT*G             MAS  840
85  MASSE=MASSE+ROUE*AREAE*G            MAS  850
86  THRUST=THRUST+ROUE2*AREAE          MAS  860
87 60  CONTINUE                                MAS  870
88  RETURN                                   MAS  880
89  END                                     MAS  890

```

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

```

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,GAMMA,RGAS,GAM1,GA
12 1M2,L1,L2,L3,M1,M2,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,ROLOW
15      COMMON /GEMTRYC/ NGEOM,XI,RI,XT,RT,XE,RE,RCI,RCT,ANGI,ANGE,XW(81),
16      1YW(81),XWI(81),YWI(81),NXNY(81),NWPTS,IINT,IDL,LT,NDIM
17      COMMON /GCB/ NGCB,XICB,RICB,XTCB,RTCB,XECB,RECB,RCICB,RCTCB,ANGICB
18      1,ANGECB,XCB(81),YCB(81),XCBI(81),YCB(81),NXNYCB(81),NCBPTS,IINTCB
19      2,IDLFCB,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 (ND,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=YW(LT)-YCB(LT)
40      RSTARS=YW(LT)**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=YW(L)-YCB(L)
61      ARATIO=RAD/RSTAR
62      GO TO 70
63      60 RADS=YW(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,0.00) 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)/DEM
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
```

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.