AD734892

DA-1T061101A91A
AMCMS Code: 501A.11.84400
HDL Proj: 39839

HDL-TR-1567

ION AND ELECTRON DISTRIBUTIONS IN THE BOUNDARY LAYER OF HYPERSONIC VEHICLES FOR CHEMICAL NONEQUILIBRIUM FLOW

Part II
METHOD OF SOLUTION AND COMPUTER PROGRAM

by

Arthur Hausner

November 1971

H•D•L

U.S ARMY MATERIEL COMMAND

HARRY DIAMOND LABORATORIES

WASHINGTON, D.C. 20438

ABSTRACT

Various computational aspects have been investigated to numerically solve charge-conservation equations and Poisson's equation for the electric field yielding ion and electron distributions. These equations were derived and presented by the Harry Diamond Laboratories as part I of this study. The computational aspects, reported herein as part II of the study, include: (1) transformations to reduce the steep slopes of the input functions and to simplify the solutions of the equations; (2) linearization of the equations to permit use of matrix methods in their solution; (3) derivation of small-value, asymptotic solutions to provide starting conditions in the matrix solution; (4) a computer program listing, description, and sample output; and (5) descriptions of an independent check solution and other checks to confirm validity of the results. The computer program is written to accommodate any consistent set of boundary conditions. Although the equations are linearized, the nonlinear terms are approximated in a way to insure rapid convergence of solutions to the exact equations.

Preceding page blank

CONTENTS

APPENDICES

ILLUSTRATIONS

TABLES

Preceding page blank

A study has been conducted by the Harry Diamond Laboratories of ion-electron distributions for a hypersonic vehicle with a 10-degree semivertex, sharply pointed cone at Mach 8 and 10 for sea-level flight. ~~Detailed~~ calculations are reported by Pollin[1] as part I of this study. This report, ~~which represents part II of the study,~~ is concerned specifically with computational aspects of the ion-electron-distribution equations ~~detailed~~ in part I.[1] The equations to be solved are presented herein to describe the transformations required to simplify their numerical solution. Readers who are particularly interested in the derivation and precise symbol definitions should refer to part I.[1]

Basically, a parabolic system of five first-order, partial differential equations describe the phenomenon (listed as equations 18 and 20 in part I). Use of the compact "±" notation to condense two equations into one is shown below.

$$\frac{\overset{\pm}{J}_\eta}{en_e} = -\overset{\pm}{s}\left[\frac{v(x,y)}{P(\eta)}\right]_\eta - \frac{W(\eta,\overset{+}{s},\overset{-}{s})}{\rho_e c_e}\delta(x) + \frac{u(\eta)\delta(x)}{P(\eta)}\overset{\pm}{s}_x \tag{1}$$

$$\overset{\pm}{s}_\eta = \frac{\delta(x)P(\eta)\dfrac{\overset{\pm}{J}}{en_e} + \delta(x)v(x,y)\overset{\pm}{s} \mp \overset{\pm}{K}(\eta)\overset{\pm}{s}\phi_\eta}{\overset{\pm}{D}(\eta) + D_T(x,\eta)} \tag{2}$$

$$\phi_{\eta\eta} = -\frac{10^{14}}{8.85}\,en_e\frac{[\delta(x)]^2}{P(\eta)}(\overset{+}{s} - \overset{-}{s}) \tag{3}$$

Equations (1) and (2) are each two equations--read first with upper and then with lower signs whenever two signs appear. The dependent variables are $\overset{+}{J}$, $\overset{-}{J}$, $\overset{+}{s}$, $\overset{-}{s}$, and ϕ; the independent variables are $0 \le x \le \infty$ and $0 \le y \le \delta(x)$ with η defined by

$$\eta = \frac{y}{\delta(x)} \tag{4}$$

so that $0 \le \eta \le 1$ is used as a normalized independent variable. This is convenient also because many of the input functions are ultimately functions of η. The other symbols are either constants (ρ_e, c_e, e, n_e), or other known functions [$P(\eta)$, $v(x,y)$, $W(\eta,\overset{+}{s},\overset{-}{s})$, $\delta(x)$, $u(\eta)$, $\overset{\pm}{D}(\eta)$, $\overset{\pm}{D}_T(x,\eta)$, $\overset{\pm}{K}(\eta)$]; some of the constants and functions change with geometry, velocity, and altitude of the vehicle.

[1]Pollin, I., "Ion and Electron Distributions in the Boundary Layer of Hypersonic Vehicles for Chemical Nonequilibrium Flow--Part I: Aerodynamics and Numerical Results," HDL-TR-1565, August 1971.

Preceding page blank

Subscripts x and η indicate partial derivatives with respect to these variables. Note that ϕ does not appear except with subscript η; ϕ_η is considered as a basic dependent variable. The voltage ϕ is calculated once ϕ_η is obtained.

The known functions are sometimes derived from formulas and sometimes are calculated in point-data form. We have

$$\delta(x) = bx^{0.8}, \text{ b constant,} \tag{5}$$

$$u(\eta) = u_\delta \eta^{1/8}, \; u_\delta \text{ constant,} \tag{6}$$

$$W(\eta,\overset{+}{s},\overset{-}{s}) = KFNO(\eta) - B(\eta)\overset{+}{s}\overset{-}{s} \tag{7}$$

where KFNO is used for K_f<N><O> and $B(\eta)$ is used for $K_r[n_e/P(\eta)]^2$. Both K_f and K_r are found as functions of the known temperature function $T \equiv T(\eta)$:

$$K_f = \frac{5\times10^{-11}\exp(-32{,}500/T)}{T^{0.5}}$$

$$K_r = 3\times10^{-3}/T^{1.5} \tag{8}$$

Also,

$$\overset{\pm}{K}(\eta) = \frac{11{,}600}{T(\eta)}\overset{\pm}{D}(T) \tag{9}$$

$$D_T(x,\eta) = 0.02u_\delta \delta(x)\eta^{9/8} \tag{10}$$

and $\overset{+}{D}(T)$ and $\overset{-}{D}(T)$ are related by

$$\overset{-}{D}(T) = 234\overset{+}{D}(T) \tag{11}$$

Finally, $v(x,y)$ must satisfy

$$\left[u_\delta \frac{x}{P(\eta)}\eta^{1/8}\right]_x + x\left[\frac{v(x,y)}{P(\eta)}\right]_y = 0 \tag{12}$$

with $v(x,0) = 0$. A lengthy computation shows that in the case considered here,

$$v(x,y) = u_\delta \frac{\delta(x)}{x} f(\eta) \tag{13}$$

where $f(\eta)$ is given by

$$f(\eta) = 0.8\,\eta^{9/8} - P(\eta)\int_0^\eta \frac{1.8\,t^{1.8}}{P(t)}\,dt \tag{14}$$

System (1) to (3) is also subject to boundary conditions; if $\overset{\pm}{s}(x,\eta)$, $\overset{\pm}{J}(x,\eta)$, and $\phi_\eta(x,\eta)$ are solutions, we have

$$\overset{+}{s}(0,\eta) = \bar{s}(0,\eta) = 0 \tag{15}$$

$$\overset{+}{s}(x,0) = \bar{s}(x,0) = 0 \tag{16}$$

$$\overset{+}{s}(x,1) = \bar{s}(x,1) = \phi_\eta(x,1) = 0 \tag{17}$$

2. THE J' TRANSFORMATION

A convenient simplification to system (1) to (3) results from transformation of the current variables $\overset{\pm}{J}$ to $\overset{\pm}{J}'$ defined by

$$\overset{\pm}{J}' = \left[\frac{\overset{\pm}{J}}{en_e} + \frac{\overset{\pm}{s}v(x,y)}{P(\eta)}\right]\delta(x) \tag{18}$$

By taking a derivitive with respect to η,

$$(\overset{\pm}{J}')_\eta = \left[\left(\frac{\overset{\pm}{J}}{en_e}\right)_\eta + \overset{\pm}{s}_\eta\left(\frac{v}{P}\right) + \overset{\pm}{s}\left(\frac{v}{P}\right)_\eta\right]\delta(x) \tag{19}$$

and substituting into equations (1) and (2), we obtain a simpler system, here expressed in terms of the more primitive functions

$$(\overset{\pm}{J}')_\eta = u_\delta \frac{[\delta(x)]^2}{x}\frac{f(\eta)}{P(\eta)}\overset{\pm}{s}_\eta - \frac{[\delta(x)]^2}{\rho_e c_e}[KFNO(\eta) - B(\eta)\overset{+}{s}\bar{s}] + u_\delta \frac{[\delta(x)]^2}{P(\eta)}\eta^{1/8}\overset{\pm}{s}_x \tag{20}$$

$$\overset{\pm}{s}_\eta = \frac{\mp 11{,}600\,\dfrac{\overset{\pm}{D}(T)}{T(\eta)}\overset{\pm}{s}\phi_\eta + P(\eta)\overset{\pm}{J}'}{\overset{\pm}{D}(T) + D_T(x,\eta)} \tag{21}$$

$$\phi_{\eta\eta} = -\frac{10^{14}}{8.85}\,\epsilon n_e\,\frac{[\delta(x)]^2}{P(\eta)}(\overset{+}{s} - \bar{s}) \tag{22}$$

This transformation has two purposes: (1) to eliminate the need for derivatives of $f(\eta)/P(\eta)$ in the solution, and (2) to eliminate the singularity $\delta(x)/x$ at $x = 0$. This term now appears as $\delta^2/x = b^2x^{0.6}$.

3. SMALL-VALUE, ASYMPTOTIC SOLUTIONS

Because of the fractional powers of x, a Taylor series solution does not exist at $x = 0$. In an attempt to find small-value (of x) solutions, we may eliminate the fractional powers in x by the transformation

$$t = x^{0.2} \text{ or } t^5 = x \tag{23}$$

with

$$\frac{dx}{dt} = 5t^4 , \qquad \overset{\pm}{s}_x = \overset{\pm}{s}_t \frac{dt}{dx} \tag{24}$$

and

$$\delta(x) = bt^4 \tag{25}$$

Equations (20) to (22) transform to

$$(\overset{\pm}{J}')_\eta = u_\delta b^2 t^3 \frac{f(\eta)}{P(\eta)} \overset{\pm}{s}_\eta - \frac{b^2 t^8}{\rho_e c_e} [\mathrm{KFNO}(\eta) - B(\eta)\overset{+}{s}\overset{-}{s}] + \frac{u_\delta b^2 t^4 \eta^{1/8} \overset{\pm}{s}_t}{5P(\eta)} \tag{26}$$

$$\overset{\pm}{s}_\eta = \frac{\mp 11,600 \dfrac{\overset{\pm}{D}(T)}{T(\eta)} \overset{\pm}{s}\phi_\eta + P(\eta)\overset{\pm}{J}'}{\overset{\pm}{D}(T) + D_T(t^5,\eta)} \tag{27}$$

$$\phi_{\eta\eta} = - \frac{10^{14}}{8.85} en_e \frac{b^2 t^8}{P(\eta)} (\overset{+}{s} - \overset{-}{s}) \tag{28}$$

A power series solution now takes the form

$$\overset{\pm}{J}' = \sum_{i=0}^{\infty} \overset{\pm}{J}_i(\eta) t^i \tag{29}$$

$$\overset{\pm}{s} = \sum_{i=0}^{\infty} \overset{\pm}{s}_i(\eta) t^i \tag{30}$$

$$\phi_\eta = \sum_{i=0}^{\infty} (\phi_\eta)_i(\eta) t^i \tag{31}$$

By substituting equations (29) to (31) into equations (26) to (28) and equating coefficients of powers of t, ordinary differential equations are found for the coefficients. [This procedure first requires a power series representation for $1/(\overset{\pm}{D} + D_T)$.] The smallest set of nonvanishing coefficients occurs for subscript 8 in equations (29) and (30) and subscript 16 in equation (31). We have

$$(\overset{\pm}{J}_8)_\eta = - \frac{b^2}{\rho_e c_e} \mathrm{KFNO}(\eta) \tag{32}$$

$$(\overset{\pm}{s}_8)_\eta = \frac{P(\eta)}{\overset{\pm}{D}(T)} \overset{\pm}{J}_8 \tag{33}$$

$$[(\phi_\eta)_{16}]_\eta = - \frac{10^{14}}{8.85} en_e \frac{b^2}{P(\eta)} (\overset{+}{s}_8 - \overset{-}{s}_8) \tag{34}$$

subject to the boundary conditions of equations (16) and (17); that is, $\overset{+}{s}_8(0) = \overset{-}{s}_8(0) = \overset{+}{s}_8(1) = \overset{-}{s}_8(1) = (\phi_\eta)_{16}(1) = 0$. Thus, we observe that for small t corresponding to small x,

$$\overset{\pm}{J}'(x,\eta) \approx \overset{\pm}{J}_8(\eta)x^{1.6} \tag{35}$$

$$\overset{\pm}{s}(x,\eta) \approx \overset{\pm}{s}_8(\eta)x^{1.6} \tag{36}$$

$$\phi_\eta(x,\eta) \approx (\phi_\eta)_{16}(\eta)x^{3.2} \tag{37}$$

This analysis also suggests that ϕ_η is essentially decoupled from the equations for $\overset{+}{s}$ and $\overset{-}{s}$ for small x. The nonlinear term $\phi_\eta\overset{\pm}{s}$ is negligible. In addition, the "+" and "-" equations are decoupled, since the nonlinear term $\overset{+-}{ss}$ is negligible. These results are almost independent of the boundary conditions and must be considered when sets of boundary conditions are desired other than those given.

4. THE z,q TRANSFORMATION

The independent variables η and x are inconvenient variables to compute with. At x = 0, we have from equation (36) that $\overset{\pm}{s}_x = 0$, but that $\overset{\pm}{s}_{xx}$ and higher derivatives are infinite. The finite difference scheme used to approximate $\overset{\pm}{s}_x$ from a sequence of values of $\overset{\pm}{s}$ is subject to large error when $\overset{\pm}{s}_{xxx}$ is large. Therefore, it is advantageous to use smaller intervals near x = 0 than are otherwise used. This situation is reinforced by the expected behavior of $\overset{\pm}{s}_x \to 0$ as $x \to \infty$; the intervals in x can then be taken further apart. To accommodate both situations, we compute in a new variable z defined by

$$z = x^{0.8} \text{ or } x = z^{1.25} \tag{38}$$

mostly to simplify checking small-value results ($\overset{\pm}{s} \sim z^2$) and provide for $\overset{\pm}{s}_{zz}$ to be finite. The equations are changed by replacing x by $z^{1.25}$ and

$$\overset{\pm}{s}_x = \overset{\pm}{s}_z \frac{dz}{dx} = \frac{0.8\overset{\pm}{s}_z}{z^{0.25}} \tag{39}$$

A similar, but more severe situation exists for the η independent variable. The driving function KFNO(η) is monotonically decreasing and is almost an impulse function. For the Mach 8 data, KFNO diminishes more than three orders of magnitude in the range $0 \le \eta \le 0.01$. It is essential in any finite difference scheme approximating derivatives with respect to η to have many intervals for small η. A transformation that automatically increases the number of η intervals for small η is given by

$$q = \ln(\eta + a) \tag{40}$$

where $a > 0$ is chosen to satisfy any particular requirement. In this case, a was chosen so that $\ell n(a + 0.0001) - \ell n(a)$ was equal to $1/400th$ of the total range of q. Equivalently, this assured that when 400 equal intervals were taken in the q direction, the first value of q corresponded to $\eta = 0$, and the second to $\eta = 0.0001$. The procedure was motivated by the fact that many of the input functions, including KFNO(η), were constant for $0 \le \eta \le 0.0001$ and dropped of steeply thereafter. A special computer program computed $\ell n(a) = -4.7939598$.

From equation (40), we have

$$\eta = e^q - a \tag{41}$$

$$\frac{d\eta}{dq} = e^q \tag{42}$$

This transformation is made by replacing η by $e^q - a$ [eq (41)] and replacing any variable r_η by

$$r_\eta = r_q \frac{dq}{d\eta} = \frac{r_q}{e^q} \tag{43}$$

In practice, the η notation was retained, except for derivatives with respect to η. All functions were computed as functions of η as found from equation (41) for any q. The transformed equations [from (20) to (22)] now appear as

$$\frac{(\overset{\pm}{J}')_q}{e^q} = u_\delta \frac{\delta^2(z^{1.25}) f(\eta)}{e^q\ z^{1.25}\ P(\eta)} \overset{\pm}{s}_q - \frac{\delta^2(z^{1.25})}{\rho_e c_e} [\mathrm{KFNO}(\eta) - B(\eta)\overset{+-}{ss}] + 0.8 u_\delta \frac{\delta^2(z^{1.25})\eta^{1/8}}{z^{0.25}\ P(\eta)} \overset{\pm}{s}_z \tag{44}$$

$$\frac{\overset{\pm}{s}_q}{e^q} = \frac{\mp 11{,}600 \frac{\overset{\pm}{D}(T)}{T(\eta)} \overset{\pm}{s}\phi_\eta + P(\eta)\overset{\pm}{J}'}{\overset{\pm}{D}(T) + D_T(z^{1.25}, \eta)} \tag{45}$$

$$\frac{(\phi_\eta)_q}{e^q} = -\frac{10^{14}}{8.85} e n_e \frac{\delta^2(z^{1.25})}{P(\eta)} (\overset{+}{s} - \overset{-}{s}) \tag{46}$$

with $\delta(z^{1.25}) = bz$.

5. NUMERICAL SOLUTION METHOD--LINEARIZATION

A "marching" technique is used to solve system (44) to (46) numerically. Given solutions at $z = z_{j-1}$, i.e., $\overset{\pm}{s}(z_{j-1},q)$, $\overset{\pm}{J}'(z_{j-1},q)$, and $\phi_\eta(z_{j-1},q)$, ordinary differential equations in η are found at $z = z_j = z_{j-1} + \Delta z$ by an appropriate finite difference approximation for $\overset{\pm}{s}_z$ in terms of $\overset{\pm}{s}(z_j,q)$, $\overset{\pm}{s}(z_{j-1},q)$, etc. To simplify notation hereafter, we refer to a function at fixed z_j with subscript j; i.e., $\overset{\pm}{s}(z_j,q) \equiv \overset{\pm}{s}_j$, understanding that it is also a function of η.

The marching technique requires the solution of a two-point boundary-value problem with nonlinear differential equations. Although these can be solved directly by iterative processes, faster matrix algorithms can be used for linearized equations, which, at the same time, do not materially affect convergence characteristics. Accordingly, system (44) to (46) is linearized by deriving differential equations for <u>changes</u> in $\overset{\pm}{s}_{j-1}$, $\overset{\pm}{J}'_{j-1}$, and $(\phi_\eta)_{j-1}$. Defining

$$\overset{\pm}{s}_j = \overset{\pm}{s}_{j-1} + \Delta\overset{\pm}{s}_j \tag{47}$$

$$\overset{\pm}{J}'_j = \overset{\pm}{J}'_{j-1} + \Delta\overset{\pm}{J}_j \tag{48}$$

$$(\phi_\eta)_j = (\phi_\eta)_{j-1} + (\Delta\phi_\eta)_j \tag{49}$$

we note that system (44) to (46) holds at both values of z_{j-1} and z_j. We substitute equations (47) to (49) into equations (44) to (46) and subtract from the resulting equations a similar set of equations obtained when $\overset{\pm}{s}_{j-1}$, $\overset{\pm}{J}'_{j-1}$, and $(\phi_\eta)_{j-1}$ are substituted into equations (44) to (46). This yields differential equations for $\Delta\overset{\pm}{s}_j$, $\Delta\overset{\pm}{J}_j$, and $(\Delta\phi_\eta)_j$ with $\overset{\pm}{s}_{j-1}$, $\overset{\pm}{J}'_{j-1}$, and $(\phi_\eta)_{j-1}$ as additional input functions. The final equations, however, depend upon how th derivative $\overset{\pm}{s}_z$ is treated, as well as the nonlinear terms $\Delta\overset{\pm}{s}_j(\Delta\phi_\eta)_j$ and $\Delta\overset{+}{s}_j\Delta\overset{-}{s}_j$.

A backward difference approximation[2] is used for $\overset{\pm}{s}_z$ when substituting at $z = z_j$:

$$\overset{\pm}{s}_z = \frac{3\overset{\pm}{s}_j - 4\overset{\pm}{s}_{j-1} + \overset{\pm}{s}_{j-2}}{2\Delta z} + O(\Delta z^2)$$

---

[2] Kopal, Z., "Numerical Analysis," John Wiley & Sons, New York, 1955, pp. 515-516.

$$\dot{\bar{s}}_z^{\pm} = \frac{3\Delta\bar{s}_j^{\pm} - \Delta\bar{s}_{j-1}^{\pm}}{2\Delta z} + O(\Delta z^2) \tag{50}$$

where $\Delta\bar{s}_{j-1}^{\pm} = \bar{s}_{j-1}^{\pm} - \bar{s}_{j-2}^{\pm}$, a quantity that must be stored during the computation. The appendage $O(\Delta z^2)$ indicates the approximation is of second order in Δz.

Equation (50) is not valid when $z = \Delta z$, i.e., for the first value of z, because $\Delta\bar{s}_0^{\pm}$ is nonexistent. Here, we use the condition that $\bar{s}_z^{\pm} = 0$ at $z = 0$. We may then show that the approximation becomes

$$\bar{s}_z^{\pm} = \frac{2\Delta\bar{s}_1^{\pm}}{\Delta z} + O(\Delta z^2) \tag{51}$$

A central difference approximation is used for $\bar{s}_z^{\pm}$ when substituting at $z = z_{j-1}$

$$\bar{s}_z^{\pm} = \frac{\bar{s}_j^{\pm} - \bar{s}_{j-2}^{\pm}}{2\Delta z} + O(\Delta z^2)$$

$$= \frac{\Delta\bar{s}_j^{\pm} + \Delta\bar{s}_{j-1}^{\pm}}{2\Delta z} + O(\Delta z^2) \tag{52}$$

When $z = \Delta z$, we take $\bar{s}_z^{\pm} = 0$ at $z = 0$, since it is a boundary condition.

Similarly, it is shown in appendix A that

$$\Delta\bar{s}_j^{\pm}(\Delta\phi_n)_j = 0 + O(\Delta z^2) \tag{53}$$

$$\Delta\bar{s}_j^{+}\Delta\bar{s}_j^{-} = 0 + O(\Delta z^2) \tag{54}$$

The substitution of equations (50) to (54) yields second-order computations of $\Delta\bar{s}_j^{\pm}$, $\Delta\bar{J}_j^{\pm}$, and $(\Delta\phi_n)_j$ which, in turn, produce linear convergence in z for $\bar{s}^{\pm}$, $\bar{J}'^{\pm}$, and ϕ_n. (If Δz is halved, each $\Delta\bar{s}^{\pm}$, $\Delta\bar{J}^{\pm}$, and $\Delta\phi_n$ is produced with one fourth the error, but there are then twice as many intervals to sum to reach a given z. Hence, if z is halved, the final error is only halved.) However, it was found that the error introduced by equations (53) and (54) was greater than that of equations (50) to (52). Accordingly, equations (53) and (54) were changed to (see appendix A)

$$\Delta\bar{s}_j^{\pm}(\Delta\phi_n)_j = \Delta\bar{s}_{j-1}^{\pm}(\Delta\phi_n)_j + O(\Delta z^3) \tag{55}$$

$$\Delta\bar{s}_j^{+}\Delta\bar{s}_j^{-} = \Delta\bar{s}_{j-1}^{+}(\Delta\bar{s}^{-})_j + O(\Delta z^3) \tag{56}$$

or $\qquad = \Delta\bar{s}_{j-1}^{-}(\Delta\bar{s}^{+})_j + O(\Delta z^3)$

which still are linear terms in subscript j variables.

The final system of linear equations that must be solved for each j is given by (the subscript j is hereafter omitted to emphasize the dependent variables; $\Delta s \equiv \Delta s_j$, etc.):

$$e^{-q}\frac{d(\Delta\overset{\pm}{s})}{dq} = \frac{1}{\overset{\pm}{D}(\eta) + D_T(z_j^{1.25},\eta)}\left\{\mp\overset{\pm}{K}(\eta)\left[(\overset{\pm}{s}_{j-1} + \Delta\overset{\pm}{s}_{j-1})\Delta\phi_\eta + (\phi_\eta)_{j-1}\Delta\overset{\pm}{s} - \frac{\Delta D_T\overset{\pm}{s}_{j-1}(\phi_\eta)_{j-1}}{\overset{\pm}{D}(\eta) + D_T(z_{j-1}^{1.25},\eta)}\right] + P(\eta)\left[\Delta\overset{\pm}{J} - (\overset{\pm}{J}')_{j-1}\frac{\Delta D_T}{\overset{\pm}{D}(\eta) + D_T(z_{j-1}^{1.25},\eta)}\right]\right\} \tag{57}$$

$$e^{-q}\frac{d(\Delta\overset{\pm}{J})}{dq} = u_\delta\frac{f(\eta)}{P(\eta)}\left[\frac{\delta_j^2}{z_j^{1.25}}\frac{\Delta\overset{\pm}{s}_q}{e^q} + \Delta\left(\frac{\delta^2}{z^{1.25}}\right)\frac{(\overset{\pm}{s}_q)_{j-1}}{e^q}\right] - \frac{KFNO(\eta)\Delta\delta^2}{\rho_e c_e} + \frac{B(\eta)}{\rho_e c_e}\left\{\delta_j^2\left[(\overset{+}{s}_{j-1} + \Delta\overset{+}{s}_{j-1})\Delta\bar{s} + \bar{s}_{j-1}\Delta\overset{+}{s}\right] + \Delta\delta^2\overset{+}{s}_{j-1}\bar{s}_{j-1}\right\} + \frac{0.4u_\delta}{\Delta z}\frac{\eta^{1/8}}{P(\eta)}\left[\left(\frac{3\delta_j^2}{z_j^{0.25}} - \frac{\delta_{j-1}^2}{z_{j-1}^{0.25}}\right)\Delta\overset{\pm}{s} - \left(\frac{\delta_j^2}{z_j^{0.25}} + \frac{\delta_{j-1}^2}{z_{j-1}^{0.25}}\right)\Delta\overset{\pm}{s}_{j-1}\right] \tag{58}$$

$$e^{-q}\frac{d(\Delta\phi_\eta)}{dq} = -\frac{10^{14}}{8.85}\frac{en_e}{P(\eta)}\left[\delta_j^2(\Delta\overset{+}{s} - \Delta\bar{s}) + \Delta\delta^2(\overset{+}{s}_{j-1} - \bar{s}_{j-1})\right] \tag{59}$$

where

$$\Delta D_T = 0.02u_\delta\Delta z\,\eta^{1/8} \tag{60}$$

$$\Delta\left(\frac{\delta^2}{z^{1.25}}\right) = \frac{\delta_j^2}{z_j^{1.25}} - \frac{\delta_{j-1}^2}{z_{j-1}^{1.25}} \tag{61}$$

$$\Delta\delta^2 = \delta_j^2 - \delta_{j-1}^2 \tag{62}$$

If $\Delta\overset{\pm}{s}_0$ is initially set equal to 0, the correct formulas are obtained when $j = 1$ if the term $3\delta_j^2/z_j^{0.25}$ is changed to $4\delta_j^2/z_j^{0.25}$ in equation (58).

It should be emphasized that quadratic convergence in Δz can be obtained if all approximations are such that the $\Delta\overset{\pm}{s}$, $\Delta\overset{\pm}{J}$, and $\Delta\phi_\eta$ are computed with order Δz^3. Although this was not done, appendix B indicates the changes required.

5.1 Matrix Formulation

If a grid of N + 1 equally-spaced points is placed along q, starting at q_1 corresponding to $\eta = 0$ and ending at q_{N+1} corresponding to $\eta = 1$, we may correspond to these points 5(N + 1) functional values we seek. Thus, for every j, we seek for all q_i, $1 \le i \le N + 1$ the values $\Delta\overset{+}{s}_i$, $\Delta\overset{-}{s}_i$, $\Delta\overset{\pm}{J}_i$, and $(\Delta\phi_\eta)_i$. Approximate values may be found with second-order error in Δq by satisfying equations (57) to (62) at the half-interval stations by using the approximations

$$(r_q)_{i+1/2} = \frac{r_{i+1} - r_i}{\Delta q} + O(\Delta q^2) \quad 1 \le i \le N \tag{63}$$

$$r_{i+1/2} = \frac{r_{i+1} + r_i}{2} + O(\Delta q^2) \quad 1 \le i \le N \tag{64}$$

where r is any required variable or stored variable at j - 1. This results in 5N linear algebraic equations, which, coupled with the five boundary conditions of equations (16) and (17), may be solved for the 5N + 5 functional values. By incrementing z_j, storing the required $\Delta\overset{+}{s}$ at each i (to be the new set of Δs_{j-1}), and mechanizing equations (47) to (49), the procedure may be used to reach any desired value of z.

5.2 Computer Solution

The computer program is written to accommodate any set of feasible boundary conditions, since this was one uncertain aspect at the time of development. It is important to prearrange the equations so that their final matrix form has a matrix coefficient that is block-tridiagonal (each block a 5×5 square matrix), in order to use an efficient algorithm[3] for their solution. A difficulty in doing this is the uncertainty of where the boundary conditions will be imposed. There must be five conditions, but m of these will be at $\eta = 0$ and 5 - m at $\eta = 1$.

---

[3]Isaacson, E. and Keller, H.B., "Analysis of Numerical Methods," John Wiley & Sons, New York, 1966, pp. 58-61.

The method used is described with the aid of figure 1, where the scheme is drawn for the boundary conditions expressed by equations (16) and (17). An array IDX is read in with values 0 or 1. As shown in figure 1, IDX(1) corresponds to $\overset{+}{s}$, etc. If IDX = 0, then the finite difference equation obtained at station i + 1/2 (the vertical arrow) is used in the i-*th* group of five equations (the diagonal arrow). If IDX = 1, the finite difference equation obtained at station i - 1/2 is used in the i-*th* group of five equations. In the first group of five equations, there are no $\Delta\overset{+}{J}$ and $\Delta\overset{-}{J}$ finite difference equations; these are replaced by the boundary conditions (in this case, $\Delta\overset{+}{s} = 0$ and $\Delta\bar{s} = 0$). In the last group of five equations, there are no $\Delta\overset{+}{s}$, $\Delta\bar{s}$, and $\Delta\phi_\eta$ finite difference equations; these are also replaced by the boundary conditions $\Delta\overset{+}{s} = 0$, $\Delta\bar{s} = 0$, and $\Delta\phi_\eta = 0$. Note in this case that IDX(1) = 1 and IDX(3) = 0 could have been used because of the symmetry in the boundary conditions; similarly, IDX(2) = 1 and IDX(4) = 0 could have been used. The final coefficient matrix appears as

$$A = \begin{pmatrix} A_1 & C_1 & & & & & & & \\ B_2 & A_2 & C_2 & & & & & 0 & \\ & \cdot & \cdot & \cdot & & & & & \\ & & \cdot & \cdot & \cdot & & & & \\ & & & \cdot & \cdot & \cdot & & & \\ & & & & B_i & A_i & C_i & & \\ & & & & & \cdot & \cdot & \cdot & \\ & & & & & & \cdot & \cdot & \cdot \\ & 0 & & & & & & \cdot \; \cdot \; \cdot & \\ & & & & & & B_N & A_N & C_N \\ & & & & & & & B_{N+1} & A_{N+1} \end{pmatrix} \qquad (65)$$

where each nonzero entry is a 5×5 matrix. If IDX(L) = 0, nonzero elements are contributed to A_i and C_i on line L. If IDX(L) = 1, nonzero elements are contributed to B_i and A_i on line L. The procedure to solve the resulting equations[3] requires the storage of N 5×5 Γ matrices and 5(N + 1) $\vec{y}$ elements. Thus, the limit of N is about 400 on the HDL 7094 without external storage. A more complete description and listing of the code is given in appendix C.

---

[3]Isaacson, E. and Keller, H.B., "Analysis of Numerical Methods," John Wiley & Sons, New York, 1966, pp. 58-61.

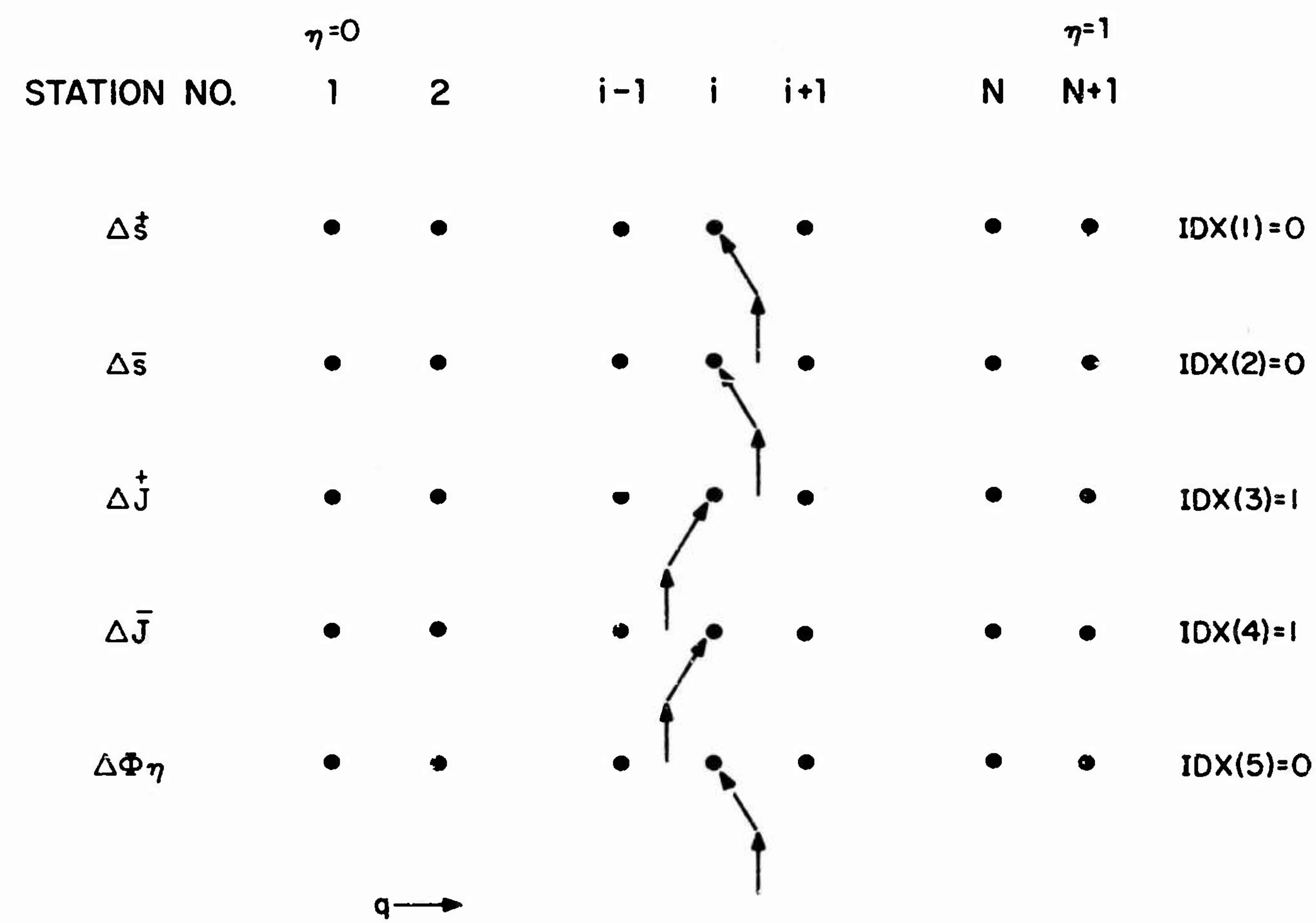

Figure 1. Scheme for matrix generation.

5.3 Solution Checks

A separate program, COMPF (not included here), computed $f(\eta)$ using equation (14) from the $P(\eta)$ data fed in, and punched the data out in the proper format. It utilized a standard integration routine; the final results were checked approximately by Simpson's rule for the Mach 8 data. Thereafter, the program was assumed correct for other data.

Another separate program, FTEST (also not listed here), checked out the FUNCT subroutine listed in appendix C. This was important to insure that all functions of η were properly generated and stored for use in the main body of the program. Included was a printout utilizing the stored functions. Points were spot checked in each column to insure reasonableness. The printout for the Mach 10 data is shown in table I.

In addition to the printout of all functions, graphs of the functions were plotted against η using HDL's CalComp plotting equipment by another specially written subprogram GRAPHS. In this way, gross errors or any disturbance from the quadratic interpolation of the input data could be easily detected. Examples of the graphs are shown in figure 2.

The major check on the validity of the results was a separate independent program, POLCHK, which solved system (57) to (59) directly with the aid of a differential-equation-solving subroutine FNOL2. Solutions of two-point boundary-value problems require no iteration with linear differential equations. Let the vector-matrix differential equation be

$$A(\eta)\,\frac{dx}{d\eta} = g(\eta) \tag{66}$$

with $x_1 = \Delta\overset{+}{s}$, $x_2 = \Delta\overset{-}{s}$, $x_3 = \Delta\overset{+}{J}$, $x_4 = \Delta\overset{-}{J}$, and $x_5 = \Delta\delta_\eta$. We start at $\eta = 1$ and run solutions with decreasing η, using the known initial (at $\eta = 1$) conditions $x_1(1) = 0$, $x_2(1) = 0$, and $x_5(1) = 0$. The other conditions vary in each of the three separate solutions:

$$A(\eta)\,\frac{dp}{d\eta} = g(\eta), \qquad p_3 = d_1,\ p_4 = 0 \tag{67}$$

$$A(\eta)\,\frac{dq}{d\eta} = g(\eta), \qquad q_3 = 0,\ q_4 = d_2 \tag{68}$$

$$A(\eta)\,\frac{dr}{d\eta} = g(\eta), \qquad r_3 = d_1,\ r_4 = d_2 \tag{69}$$

where $p_k(1) = q_k(1) = r_k(1) = x_k(1) = 0$ when $k = 1, 2$, or 5.

Because of the superposition theorem, it is easily shown that

$$x = ap + bq + cr, \tag{70}$$

if and only if

Table I. FTEST output for Mach 10 data.

V = 3.35280E 05   C - 0.0200   RHOECE = 5.630CCE 12   NE = 5.63000E 12   DT FACTOR = 5710.

ETA	P	T	F	KFNO	R	CLPLUS	KPLUS	CLMINUS	KMINUS	ETA**.125	ETA**1.125
4.9935E-05	1.0000E 00	3.6720E 03	-1.1560E-05	2.8045E 17	4.2735E 17	2.3594E 00	7.4535E 00	5.5210E 02	1.7441E 03	2.8997E-01	1.4450E-05
8.3951E-04	9.5053E-01	3.5410E 03	-2.8348E-04	1.0033E 17	4.7947E 17	2.2224E 00	7.2811E 00	5.2910E 02	1.7038E 03	4.1557E-01	3.0966E-04
1.8146E-03	9.3300E-01	3.4844E 03	-6.3365E-04	5.2619E 16	5.3110E 17	2.1557E 00	7.2099E 00	5.0679E 02	1.6871E 03	4.5428E-01	8.2399E-04
2.8315E-03	9.1637E-01	3.4329E 03	-1.0276E-03	3.2490E 16	5.5305E 17	2.1151E 00	7.1471E 00	4.9494E 02	1.6724E 03	4.8026E-01	1.3596E-03
3.9515E-03	9.0042E-01	3.3849E 03	-1.4787E-03	2.1812E 16	5.7531E 17	2.0682E 00	7.0878E 00	4.8395E 02	1.6585E 03	5.0072E-01	1.9785E-03
5.1842E-03	8.4737E-01	3.3476E 03	-1.9923E-03	1.6086E 16	6.2278E 17	2.0309E 00	7.0275E 00	4.7524E 02	1.6468E 03	5.1001E-01	2.68 6E-03
6.5425E-03	8.7370E-01	3.3113E 03	-2.5670E-03	1.1761E 16	6.5079E 17	1.9946E 00	6.9875E 00	4.6573E 02	1.6351E 03	5.3329E-01	3.4945E-03
8.0356E-03	8.6314E-01	3.2781E 03	-3.2238E-03	8.6539E 15	6.7695E 17	1.9614E 00	6.9407E 00	4.5897E 02	1.6241E 03	5.4718E-01	4.3370E-03
9.5915E-03	8.5470E-01	3.2451E 03	-3.9543E-03	6.2772E 15	7.0382E 17	1.9285E 00	6.8955E 00	4.5125E 02	1.6131E 03	5.6026E-01	5.4217E-03
1.1491E-02	8.4574E-01	3.2151E 03	-4.7720E-03	4.5374E 15	7.2926E 17	1.8954E 00	6.8474E 00	4.4472E 02	1.6028E 03	5.7229E-01	6.5731E-03
1.3488E-02	8.3642E-01	3.1842E 03	-5.6833E-03	3.2108E 15	7.5648E 17	1.8675E 00	6.8033E 00	4.3599E 02	1.5920E 03	5.8375E-01	7.8713E-03
1.5593E-02	8.2657E-01	3.1516E 03	-6.6977E-03	2.2342E 15	7.8646E 17	1.8349E 00	6.7538E 00	4.2739E 02	1.5804E 03	5.9496E-01	9.3261E-03
1.7997E-02	8.1654E-01	3.1176E 03	-7.8153E-03	1.5201E 15	8.1933E 17	1.8009E 00	6.7059E 00	4.2141E 02	1.5680E 03	6.0350E-01	1.0957E-02
2.0757E-02	8.0471E-01	3.0938E 03	-9.0614E-03	1.0334E 15	8.5327E 17	1.7671E 00	6.6472E 00	4.1350E 02	1.5554E 03	6.1607E-01	1.2784E-02
2.3588E-02	7.9774E-01	3.0523E 03	-1.0457E-02	7.1540E 14	8.8563E 17	1.7356E 00	6.5951E 00	4.0614E 02	1.5435E 03	6.2632E-01	1.4839E-02
2.7902E-02	7.8451E-01	3.0189E 03	-1.2010E-02	4.7384E 14	9.2198E 17	1.7021E 00	6.5405E 00	3.9429E 02	1.5305E 03	6.3618E-01	1.7119E-02
3.0458E-02	7.7871E-01	2.9851E 03	-1.3767E-02	3.0330E 14	9.6245E 17	1.6676E 00	6.4845E 00	3.9022E 02	1.5174E 03	6.4531E-01	1.9678E-02
3.4352E-02	7.6327E-01	2.9455E 03	-1.5572E-02	1.8731E 14	1.0078E 18	1.6332E 00	6.4316E 00	3.8216E 02	1.5050E 03	6.5613E-01	2.2538E-02
3.9250E-02	7.5732E-01	2.9059E 03	-1.7522E-02	1.1173E 14	1.0583E 18	1.5980E 00	6.3747E 00	3.7393E 02	1.4926E 03	6.6587E-01	2.5734E-02
4.3341E-02	7.4675E-01	2.8575E 03	-1.9960E-02	6.7381E 13	1.1105E 18	1.5645E 00	6.3294E 00	3.6613E 02	1.4811E 03	6.7555E-01	2.9304E-02
4.8590E-02	7.5539E-01	2.8279E 03	-2.2461E-02	3.9155E 13	1.1577E 18	1.5312E 00	6.2811E 00	3.5831E 02	1.4698E 03	6.8519E-01	3.3290E-02
5.4324E-02	7.2434E-01	2.7838E 03	-2.5152E-02	2.1447E 13	1.2326E 18	1.4959E 00	6.2269E 00	3.5004E 02	1.4576E 03	6.9481E-01	3.7741E-02
6.0535E-02	7.1214E-01	2.7413E 03	-2.8111E-02	1.1082E 13	1.3054E 18	1.4570E 00	6.1653E 00	3.4094E 02	1.4427E 03	7.0442E-01	4.2708E-02
6.7078E-02	6.9932E-01	2.6947E 03	-3.1309E-02	5.4065E 12	1.3900E 18	1.4167E 00	6.0954E 00	3.3139E 02	1.4266E 03	7.1404E-01	4.8251E-02
7.3221E-02	6.8634E-01	2.6479E 03	-3.4829E-02	2.5861E 12	1.4815E 18	1.3753E 00	6.0247E 00	3.2181E 02	1.4098E 03	7.2367E-01	5.4451E-02
8.3364E-02	6.7730E-01	2.5991E 02	-3.8729E-02	1.1533E 12	1.5854E 18	1.3324E 00	5.9474E 00	3.1183E 02	1.3917E 03	7.3335E-01	6.1335E-02
9.2002E-02	6.5851E-01	2.5473E 03	-4.2906E-02	4.6784E 11	1.7057E 18	1.2856E 00	5.8681E 00	3.0154E 02	1.3731E 03	7.4303E-01	6.9031E-02
1.0315E-01	6.4421E-01	2.4944E 03	-4.7466E-02	1.7416E 11	1.8392E 18	1.2443E 00	5.7855E 00	2.9117E 02	1.3540E 03	7.5277E-01	7.7614E-02
1.1453E-01	6.3074E-01	2.4426E 03	-5.2527E-02	6.0160E 10	1.9300E 18	1.2015E 00	5.7054E 00	2.8115E 02	1.3352E 03	7.6256E-01	8.7186E-02
1.2671E-01	6.1633E-01	2.3872E 03	-5.7909E-02	1.8812E 10	2.1462E 18	1.1563E 00	5.6187E 00	2.7057E 02	1.3148E 03	7.7240E-01	9.7859E-02
1.4323E-01	6.2049E-01	2.3281E 03	-6.3713E-02	5.3629E 09	2.3437E 18	1.1088E 00	5.5247E 00	2.5946E 02	1.2928E 03	7.8232E-01	1.0976E-01
1.5553E-01	5.8472E-01	2.2554E 03	-7.0014E-02	1.3474E 09	2.5794E 18	1.0597E 00	5.4257E 00	2.4786E 02	1.2691E 03	7.9230E-01	1.2303E-01
1.7162E-01	5.6755E-01	2.1973E 03	-7.6638E-02	3.0639E 08	2.8622E 18	1.0078E 00	5.3155E 00	2.3583E 02	1.2438E 03	8.0236E-01	1.3782E-01
1.8972E-01	5.4958E-01	2.1300E 03	-8.3646E-02	6.2172E 07	3.2025E 18	9.5486E-01	5.2003E 00	2.2344E 02	1.2169E 03	8.1249E-01	1.5431E-01
2.0976E-01	5.3142E-01	2.0599E 03	-9.1016E-02	1.1592E 07	3.5015E 18	9.0239E-01	5.0817E 00	2.1115E 02	1.1891E 03	8.2272E-01	1.7264E-01
2.3191E-01	5.1266E-01	1.9886E 03	-9.8693E-02	1.8684E 06	4.0753E 18	8.5014E-01	4.9592E 00	1.9993E 02	1.1604E 03	8.3303E-01	1.9317E-01
2.5616E-01	4.9359E-01	1.9136E 03	-1.0659E-01	2.6712E 05	4.6627E 18	7.9738E-01	4.8337E 00	1.8659E 02	1.1311E 03	8.4343E-01	2.1600E-01
2.8281E-01	4.7341E-01	1.8353E 03	-1.1493E-01	3.2988E 04	5.3964E 18	7.4418E-01	4.7036E 00	1.7414E 02	1.1007E 03	8.5394E-01	2.4145E-01
3.1218E-01	4.5237E-01	1.7554E 03	-1.2340E-01	3.5275E 03	6.3013E 18	6.9085E-01	4.5653E 00	1.6166E 02	1.0683E 03	8.6454E-01	2.6981E-01
3.4445E-01	4.3225E-01	1.6749E 03	-1.3191E-01	3.2240E 02	7.4311E 18	6.3675E-01	4.4125E 00	1.4900E 02	1.0325E 03	8.7525E-01	3.0142E-01
3.7998E-01	4.1354E-01	1.5994E 03	-1.4010E-01	2.4272E 01	8.8997E 18	5.8177E-01	4.2460E 00	1.3613E 02	9.9356E 02	8.8606E-01	3.3665E-01
4.1916E-01	3.8835E-01	1.5546E 03	-1.4828E-01	1.5241E 00	1.0776E 19	5.3099E-01	4.0938E 00	1.2425E 02	9.5796E 02	8.9699E-01	3.7590E-01
4.6221E-01	3.5533E-01	1.4174E 03	-1.5523E-01	7.5518E-02	1.3278E 19	4.8193E-01	3.9440E 00	1.1277E 02	9.2291E 02	9.0803E-01	4.1965E-01
5.0955E-01	3.2335E-01	1.3243E 03	-1.5894E-01	2.6549E-03	1.6965E 19	4.3149E-01	3.7795E 00	1.0097E 02	8.8441E 02	9.1918E-01	4.6839E-01
5.6181E-01	3.1502E-01	1.2253E 03	-1.5810E-01	6.0307E-05	2.2190E 19	3.8021E-01	3.5973E 00	8.3959E 01	8.4224E 02	9.3046E-01	5.2271E-01
6.1921E-01	2.3942E-01	1.1225E 03	-1.5445E-01	8.3335E-07	3.0172E 19	3.2949E-01	3.4050E 00	7.7099E 01	7.9677E 02	9.4185E-01	5.8324E-01
6.8251E-01	2.6231E-01	1.0155E 03	-1.4438E-01	5.9061E-09	4.2599E 19	2.7756E-01	3.1930E 00	6.5418E 01	7.4717E 02	9.5337E-01	6.5068E-01
7.5210E-01	2.3339E-01	9.0429E 02	-1.2640E-01	1. 094E-11	6.3920E 19	2.3057E-01	2.9574E 00	5.3955E 01	6.9212E 02	9.6502E-01	7.2583E-01
8.2886E-01	2.0416E-01	7.8996E 02	-9.5132E-02	1.1 73E-14	1.0275E 20	1.8359E-01	2.6958E 00	4.2959E 01	6.3082E 02	9.7683E-01	8.0957E-01
9.1328E-01	1.7192E-01	6.5537E 02	-3.9242E-02	7.2416E-19	1.8704E 20	1.3814E-01	2.4047E 00	3.2324E 01	5.6259E 02	9.8871E-01	9.0297E-01
9.9405E-01	1.4039E-01	5.4337E 02	4.2378E-02	1.8630E-24	3.8095E 20	9.8054E-02	2.0932E 00	2.2945E 01	4.8993E 02	9.9924E-01	9.9321E-01

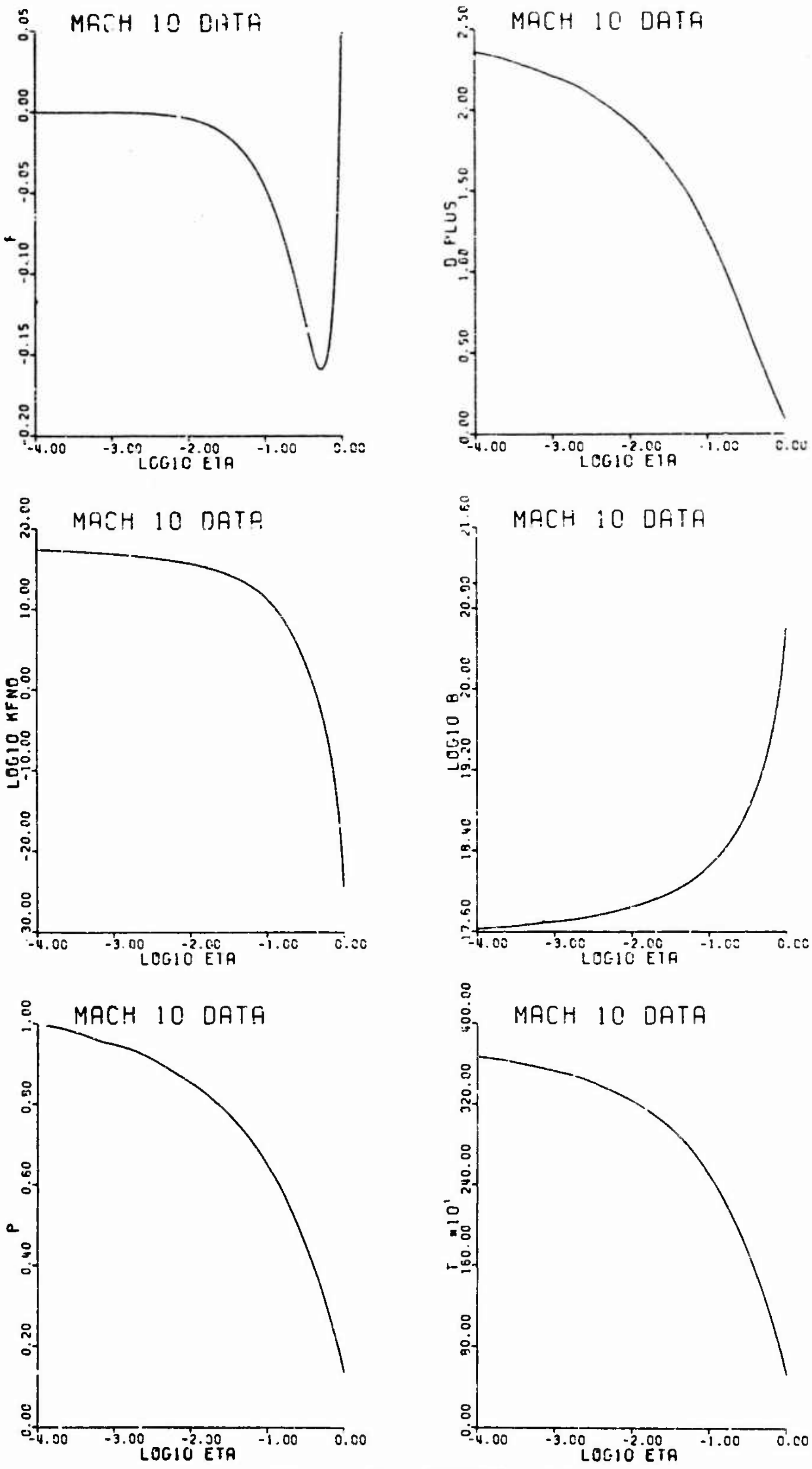

Figure 2. GRAPHS output.

$$a + b + c = 1, \tag{71}$$

and the boundary conditions are satisfied:

$$\Delta\overset{+}{s}(0) = x_1(0) = ap_1(0) + bq_1(0) + cr_1(0) = 0 \tag{72}$$

$$\Delta\overset{-}{s}(0) = x_2(0) = ap_2(0) + bq_2(0) + cr_2(0) = 0. \tag{73}$$

Equations (71) to (73) can be solved for a, b, and c. The initial conditions d_1 and d_2 can be arbitrarily nonzero, but in practice they are chosen so that $r(\eta)$ is close to $x(\eta)$; i.e., c will be very nearly equal to 1. This minimizes roundoff errors since the magnitudes of a, b, and c can otherwise be quite large but of opposite sign. The program was written so that d_1 and d_2 were iterated on successive runs h and h + 1 before z was stepped up:

$$(d_1)_{h+1} = (d_1)_h(a + c) \tag{74}$$

$$(d_2)_{h+1} = (d_2)_h(b + c) \tag{75}$$

The iteration was stopped when the magnitudes a and b were appropriately small enough. Theoretically, c at step 3 should equal c at step 2 exactly, but because of roundoff errors and the inability to exactly solve differential equations numerically, the iteration index occasionally went to 4.

A fourth-order Runge-Kutta integration scheme was used to compute the values of variables at fixed points. These points were sufficiently close, so that error in the η direction was essentially negligible. This check showed that the finite difference scheme of section 5.2 was working correctly, and that the error introduced in the η direction was less than 2 percent when using 400 intervals (for Mach 8 data).

By obtaining runs at a given Δz, $\Delta z/2$, and $\Delta z/4$, the linear nature of the convergence in the z direction was verified. In addition, the error using

$$\Delta z = \frac{100^{0.8}}{50} = \frac{z_{max}}{50}$$

was less than 5 percent (for Mach 10 data). Such a run corresponded to about 8 minutes of IBM 7094 CPU time.

Appendix A. <u>Order of Errors in Nonlinear Approximations</u>

Since

$$\overset{\pm}{s} = \overset{\pm}{s}_{j-1} + \left(\frac{\partial \overset{\pm}{s}}{\partial z}\right)_{j-1} (z - z_{j-1}) + \frac{1}{2}\left(\frac{\partial^2 \overset{\pm}{s}}{\partial z^2}\right)_{j-1} (z - z_{j-1})^2 + \ldots$$

we have

$$\overset{\pm}{s}_j - \overset{\pm}{s}_{j-1} = \Delta\overset{\pm}{s}_j = \left(\frac{\partial \overset{\pm}{s}}{\partial z}\right)_{j-1} \Delta z + \frac{1}{2}\left(\frac{\partial^2 \overset{\pm}{s}}{\partial z^2}\right)_{j-1} \Delta z^2 + \ldots$$

Similarly,

$$(\Delta\phi_\eta)_j = \left(\frac{\partial \phi_\eta}{\partial z}\right)_{j-1} \Delta z + \frac{1}{2}\left(\frac{\partial^2 \phi_\eta}{\partial z^2}\right)_{j-1} \Delta z^2 + \ldots$$

so that

$$\Delta\overset{\pm}{s}_j (\Delta\phi_\eta)_j = \left(\frac{\partial \overset{\pm}{s}}{\partial z}\right)_{j-1} \left(\frac{\partial \phi_\eta}{\partial z}\right)_{j-1} \Delta z^2 + \ldots$$

$$= 0 + \mathcal{O}(\Delta z^2)$$

$$\Delta\overset{+}{s}_j \Delta\overset{-}{s}_j = \left(\frac{\partial \overset{+}{s}}{\partial z}\right)_{j-1} \left(\frac{\partial \overset{-}{s}}{\partial z}\right)_{j-1} \Delta z^2 + \ldots$$

$$= 0 + \mathcal{O}(\Delta z^2)$$

This implies that ignoring the nonlinear terms simply introduces a second-order error. On the other hand, we also have

$$\Delta\overset{\pm}{s}_{j-1} = \left(\frac{\partial \overset{\pm}{s}}{\partial z}\right)_{j-2} \Delta z + \frac{1}{2}\left(\frac{\partial^2 \overset{\pm}{s}}{\partial z^2}\right)_{j-2} \Delta z^2 + \ldots$$

so that by subtracting

$$\Delta\overset{\pm}{s}_j - \Delta\overset{\pm}{s}_{j-1} = \left[\left(\frac{\partial \overset{\pm}{s}}{\partial z}\right)_{j-1} - \left(\frac{\partial \overset{\pm}{s}}{\partial z}\right)_{j-2}\right]\Delta z + \frac{1}{2}\left[\left(\frac{\partial^2 \overset{\pm}{s}}{\partial z^2}\right)_{j-1} - \left(\frac{\partial^2 \overset{\pm}{s}}{\partial z^2}\right)_{j-2}\right]\Delta z^2 + \ldots$$

But,

$$\frac{\partial \overset{\pm}{s}}{\partial z} = \left(\frac{\partial \overset{\pm}{s}}{\partial z}\right)_{j-1} + \left(\frac{\partial^2 \overset{\pm}{s}}{\partial z^2}\right)_{j-1} (z - z_{j-1}) + \ldots$$

so that

$$\left(\frac{\partial \overset{\pm}{s}}{\partial z}\right)_{j-1} - \left(\frac{\partial \overset{\pm}{s}}{\partial z}\right)_{j-2} = \left(\frac{\partial^2 \overset{\pm}{s}}{\partial z^2}\right)_{j-1} \Delta z + \ldots$$

This leads to

$$\Delta \bar{s}^{\pm}_{j} = \Delta \bar{s}^{\pm}_{j-1} + \mathcal{O}(\Delta z^2)$$

and

$$\Delta \bar{s}^{\pm}_{j} (\Delta \phi_{\eta})_{j} = \Delta \bar{s}^{\pm}_{j-1} (\Delta \phi_{\eta})_{j} + \mathcal{O}(\Delta z^3)$$

Equation (56) follows similarly. Since $\Delta \bar{s}^{\pm}_{j-1}$ must be stored for the derivative approximation equation (50) anyway, no additional storage is required.

Appendix B. Formulas for Quadratic Convergence in z

The first set of $\Delta\overset{\pm}{s}$, $\Delta\overset{\pm}{J}$, and $\Delta\phi_\eta$ can be computed with a second-order error, but each set thereafter should be computed with third-order to obtain quadratic convergence. No change, therefore, occurs for $j = 1$; equation (51) applies for the $\overset{\pm}{s}_z$ approximation at $z = \Delta z$.

For $j = 2$, i.e., $z = 2\Delta z$, a special computation must be made, taking into account the stored $\Delta\overset{\pm}{s}_1$ and the fact that $\overset{\pm}{s}_z = 0$ at $z = 0$. Note that we need to estimate $\overset{\pm}{s}_z$ at $z = \Delta z$ as well as $z = 2\Delta z$. If $z = z_1 (= \Delta z)$, we set up a Taylor expansion of $s(z)$ about z_1.

$$s = s_1 + s_1'(z - z_1) + \frac{1}{2} s_1''(z - z_1)^2 + \frac{1}{6} s_1'''(z - z_1)^3$$

$$+ \frac{1}{24} s_1''''(z - z_1)^4 + \ldots$$

$$s' = s_1' + s_1''(z - z_1) + \frac{1}{2} s_1'''(z - z_1)^2 + \frac{1}{6} s_1''''(z - z_1)^3 + \ldots$$

At $z = z_1 - \Delta z$ $(= z_0 = 0)$, $s = 0$ and $s' = 0$. Also, at $z = z_1 + \Delta z = z_2$, $s = s_2$. Thus,

$$0 = s_0 = s_1 - s_1'\Delta z + \frac{1}{2} s_1''\Delta z^2 - \frac{1}{6} s_1'''\Delta z^3 + \frac{1}{24} s_1''''\Delta z^4 + \ldots$$

$$0 = s_1' - s_1''\Delta z + \frac{1}{2} s_1'''\Delta z^2 - \frac{1}{6} s_1''''\Delta z^3 + \ldots$$

$$s_2 = s_1 + s_1'\Delta z + \frac{1}{2} s_1''\Delta z^2 + \frac{1}{6} s_1'''\Delta z^3 + \frac{1}{24} s_1''''\Delta z^4 + \ldots$$

Eliminating s_1'' and s_1''' from the equations yields

$$s_1' = \frac{4s_1 + s_2}{4\Delta z} - \frac{s_1''''}{24}\Delta z^3 + \ldots$$

or, in terms of Δs_2 and Δs_1,

$$s_1' = \frac{5\Delta s_1 + \Delta s_2}{4\Delta z} + O(\Delta z^3)$$

since $s_2 - s_1 = \Delta s_2$ and $s_1 = \Delta s_1$.

To find s_2' in terms of Δs_1 and Δs_2, a Taylor series about $z = z_2$ is constructed:

$$s = s_2 + s_2'(z - z_2) + \frac{1}{2} s_2''(z - z_2)^2 + \frac{1}{6} s_2'''(z - z_2)^3$$

$$+ \frac{1}{24} s_2''''(z - z_2)^4 + \ldots$$

with

$$s' = s_2' + s_2''(z - z_2) + \frac{1}{2} s_2'''(z - z_2)^2 + \frac{1}{6} s_2''''(z - z_2)^3 + \ldots$$

At $z = 0$ $(z - z_2 = -2\Delta z)$, $s = s' = 0$, and at $z = z_1$, $s = s_1$. Thus,

$$0 = s_2 - 2s_2'\Delta z + 2s_2''\Delta z^2 - \frac{8}{6} s_2'''\Delta z^3 + \frac{16}{24} s_2''''\Delta z^4 + \ldots$$

$$0 = s_2' - 2s_2''\Delta z + 2 s_2'''\Delta z^2 - \frac{8}{6} s_2''''\Delta z^3 + \ldots$$

$$s_1 = s_2 - s_2'\Delta z + \frac{1}{2}s_2''\Delta z^2 - \frac{1}{6} s_2'''\Delta z^3 + \frac{1}{24} s_2''''\Delta z^4 + \ldots$$

Eliminating s_2'' and s_2''' from the equations yields

$$s_2' = \frac{2s_2 - 4s_1}{\Delta z} + \frac{1}{6} s_2''''\Delta z^3 + \ldots$$

In terms of Δs_2 and Δs_1,

$$s_2' = \frac{2\Delta s_2 - 2\Delta s_1}{\Delta z} + O(\Delta z^3)$$

A projected Δs_j for use in the nonlinear approximations (55) and (56) can also be found by eliminating s_2' and s_2'' from the above equations. We have

$$s_2 = 4s_1 + \frac{2}{3} s_2'''\Delta z^3 + \ldots$$

or

$$\Delta s_2 = 3\Delta s_1 + O(\Delta z^3)$$

From appendix A,

$$\Delta\overset{\pm}{s}_2(\Delta\phi_\eta)_2 = 3\Delta\overset{\pm}{s}_1(\Delta\phi_\eta)_2 + O(\Delta z^4)$$

$$\Delta\overset{+}{s}_2\Delta\overset{-}{s}_2 = 3\Delta\overset{+}{s}_1\Delta\overset{-}{s}_2 + O(\Delta z^4)$$

or

$$= 3\Delta\overset{+}{s}_2\Delta\overset{-}{s}_1 + O(\Delta z^4)$$

For $j \geq 3$, we may use standard formulas[2]

$$s_{j-1}' = \frac{s_{j-3} - 6s_{j-2} + 3s_{j-1} + 2s_j}{6\Delta z} + O(\Delta z^3)$$

---

[2]Kopal, Z., "Numerical Analysis," John Wiley & Sons, New York, 1955, pp. 515-516.

or

$$s'_{j-1} = \frac{2\Delta s_j + 5\Delta s_{j-1} - \Delta s_{j-2}}{6\Delta z} + O(\Delta z^3)$$

and

$$s'_j = \frac{-2s_{j-3} + 9s_{j-2} - 18s_{j-1} + 11s_j}{6\Delta z} + O(\Delta z^3)$$

or

$$s'_j = \frac{11\Delta s_j - 7\Delta s_{j-1} + 2\Delta s_{j-2}}{6\Delta z} + O(\Delta z^3)$$

The use of these formulas naturally requires the additional storage for $\Delta \overset{\pm}{s}_{j-2}$.

A projected Δs_j may also be derived with error of order $(\Delta z)^3$ by constructing a Taylor series about $z = z_j$:

$$s = s_j + s'_j(z - z_j) + \frac{1}{2} s''_j(z - z_j)^2 + \frac{1}{6} s'''_j (z - z_j)^3 + \ldots$$

and substituting for previously found points:

$$s_{j-1} = s_j - s'_j\Delta z + \frac{1}{2} s''_j\Delta z^2 - \frac{1}{6} s'''_j \Delta z^3 + \ldots$$

$$s_{j-2} = s_j - 2s'_j\Delta z + \frac{4}{2} s''_j\Delta z^2 - \frac{8}{6} s'''_j \Delta z^3 + \ldots$$

$$s_{j-3} = s_j - 3s'_j\Delta z + \frac{9}{2} s''_j\Delta z^2 - \frac{27}{6} s'''_j \Delta z^3 + \ldots$$

By eliminating s'_j and s''_j, we get

$$3s_{j-1} - 3s_{j-2} + s_{j-3} = s_j - s'''_j \Delta z^3 + \ldots$$

or

$$\Delta s_j = 2\Delta s_{j-1} - \Delta s_{j-2} + O(\Delta z^3)$$

Appendix C. Program Description and Listing

In the listing of the computer program (presented on pages 30-40), the main program, POLMD2, utilizes the following subroutines to simplify the flow.

FUNCT--Reads in the main data, computes some needed constants and all needed functions of η at half-integer stations by quadratic interpolation.

TERP2N (not listed)--Called by FUNCT to perform quadratic interpolation.

INITLZ--Initializes all variables and computes other needed constants.

The subroutines FUNCT, TERP2N, and INITLZ were overlayed by subsequent subroutines, since they are needed only at the start of the run.

XSTEP--Steps z and computes all constants that are functions of z only.

EMPTY--Initializes the A_i, B_i, and C_i matrices, and the f_i vector to 0. Note that the A_i matrix is called X in the program, the B_i is called BB, and the C_i matrix is adjoined with the f_i vector to form a 5×6 matrix called D. (This permits the later computation of $A_i^{-1}C$ and $A_i^{-1}f_i$ in one operation.)

EMPTY starts the minor i loop, the purpose of which is to eliminate the i-*th* block of five equations

$$B_i \begin{pmatrix} \Delta \overset{+}{s}_j \\ \Delta \overset{-}{s}_j \\ \Delta \overset{+}{J}_j \\ \Delta \overset{-}{J}_j \\ (\Delta\phi_\eta)_j \end{pmatrix}_{i-1} + A_i \begin{pmatrix} \Delta \overset{+}{s}_j \\ \Delta \overset{-}{s}_j \\ \Delta \overset{+}{J}_j \\ \Delta \overset{-}{J}_j \\ (\Delta\phi_\eta)_j \end{pmatrix}_{i} + C_i \begin{pmatrix} \Delta \overset{+}{s}_j \\ \Delta \overset{-}{s}_j \\ \Delta \overset{+}{J}_j \\ \Delta \overset{-}{J}_j \\ (\Delta\phi_\eta)_j \end{pmatrix}_{i+1} = f_i$$

($B_i = 0$ for $i = 1$ and $C_i = 0$ for $i = N + 1$) where the subscript j indicates that $z = j\Delta z$.

FIRST, SECOND, THIRD, FOURTH, FIFTH--are entries into EMPTY that fill the X, BB, and D matrices line by line. Note that single subscript arithmetic is used to conserve computer time and core.

END0, END1--are called when i = 1 and N + 1 to take the boundary conditions into account.

BOUND0, BOUND1--define the boundary conditions at $\eta = 0$ and $\eta = 1$. This is the only subroutine the user must write if his boundary conditions differ from equations (16) and (17). [Actually, the supplied BOUND1 contained the condition that $\phi_\eta(x,1) = P_1\delta$; P_1 was read in to be 0 to satisfy equation (17).]

XDCORR--corrects the A_i and f_i arrays by the matrix multiplication:

$$A_i = A_i - B_i\Gamma_{i-1} \ , \ i = 2,3,\ldots,N+1$$

$$f_i = f_i - B_i y_{i-1} \ , \ i = 2,3,\ldots,N+1$$

as required by the algorithm.[3]

To speed the process, only the possible nonzero elements of B_i were used and every term was written out.

GELG (not listed)--computes $A_i^{-1}D$ by Gaussian elimination using complete pivoting.

The first five columns of $A_i^{-1}D$ are stored as the Γ_i array (in the computer Z array) and the last column of $A_i^{-1}D$ is stored as the y_i array (in the computer XX array).

REST--does the rest of the computation in the i loop:

1. Back multiplies to find the variables.
2. Updates all needed variable functions of η and some functions of x.
3. Determines if a printout is required for the value of z just completed.
4. If a printout is needed, computes ϕ_j by rectangular formulas, retransforms to $\overset{\pm}{J}$ from $\overset{\pm}{J}'$, and computes other functions as needed for the printout. A sample printout is shown in table C-I on page 41.

In the main program, the unlabeled COMMON is placed in equivalence with the Z array to save core. All input data and some variables needed in the FUNCT, INITLZ, and REST routines are not needed during the i loop and can be shared with the Z array. Liberal use of labeled COMMONS served to otherwise transfer data from subroutine to subroutine.

[3]Isaacson, E. and Keller, H.B., "Analysis of Numerical Methods," John Wiley & Sons, New York, 1966, pp. 58-61.

```
      DIMENSION Z(10000)
      COMMON ETAPT(100),PPT(300),ETALG(100),BOB(200),TPT2(100),DPT(100),
     1 O(2),F1(3),PHJ(401),NE2,XN,XZ,XEND,XPRINT,SNE,DY,YO,ETAI,XI,Y,
     2 EXPD,UPLS,A1,A2,A3,DELTAX,JOP,JOM,VSM,PHIY,DSYSP,DSYSM,SNM,
     3 TSYSP,TSYSM
      COMMON/CONST1/XJ,DELJX,ZJ,DZ25,DELJ
      COMMON/CONST2/PI,P2,P3,P4,P5,P6
      COMMON/CONST3/ DELZ,CDEL2,RHOECE,VDX,ENEE,XZ12,XJ12,DELJ12,DELJX2,
     1 DZ252,C(8),V,NE,CDEL,CDELE,DDELTA,DELTA
      COMMON/FUN1/DPLUS(400),DMINUS(400),EP(400),ETA98(400),T29(400),
     1 B(400),FPEY(400),KFNO(400),DYEY(400),P(400),T29M(400)
      COMMON/FUN2/ B0,B1,KFNOO,KFNO1
      COMMON/FUN3/ ETAQ(51),FQ(51),DTPLQ(51),FPO(51),PO(51),DPLUQ(51),
     1 DPMIQ(51),ETA9C(51)
      COMMON/INTLZ/PHIJ(401),SPJ(401),SMJ(401),JPJ(401),JMJ(401),
     1 DSPJ(401),DSMJ(401),SPJH(400),SMJH(400),PHIJH(400)
      COMMON/MAT1/X(25),D(30),BB(25)
      COMMON/MAT2/XX(2005)
      COMMON/IND1/ I,J,M,II,JJ,K
      COMMON/INDX/ IDX(6)
      COMMON/NUMB/ N,NH,NP1,NC,NCOUNT
      EQUIVALENCE (Z(1),ETAPT(1))
      REAL JOP,JOM,KFNO,JPJ,JMJ,NE,NE2,KFNOO,KFNO1
C  COMPUTE FUNCTIONS
      CALL FUNCT
C  INITIALIZE ALL VARIABLES
      CALL INITLZ
C  ROUTINES FUNCT, INITLZ, AND TEPP2N ARE NO LONGER NEEDED.
C  MAJOR Z LOOP
      DO 999 J = 1,M
C       INITIALIZE X FUNCTIONS.
      CALL XSTEP
C  MINOR ETA LOOP
      DO 900 I = 1,NP1
C  EMPTY AND FILL X, D, AND BB MATRICES WITH A, C AND F, AND B.
      CALL EMPTY
      IF(I.EQ.1) GO TO 10
      IF(I.EQ.NP1) GO TO 20
      CALL FIRST
      CALL SECOND
      CALL THIRD
      CALL FOURTH
      CALL FIFTH
      GO TO 3
   10 CALL END0
      GO TO 150
   20 CALL END1
C  CORRECT FOR B WHEN I NOT EQUAL TO 1
    3 CALL XDCORR(Z)
C     READY TO INVERT
  150 CALL GELG(D,X,5,6)
C  TRANSFER GAMMA MATRICES AND Y VECTOR TO STORAGE.
      IF(I.EQ.NPI) GO TO 200
      JJ = 25*(I-1)
      DO 160 II = 1,25
```

NOT REPRODUCIBLE

```
      K = II + JJ
  160 Z(K) = D(II)
  200 JJ = 5*(I-1)
      DO 210 II = 1,5
      K = JJ + II
  210 XX(K) = D(II+75)
C  END OF ETA LOOP.
  900 CONTINUE
C  UPDATE AND WRITE IN REST ROUTINE.
      CALL REST(Z)
C  END OF Z LOOP.
  999 CONTINUE
      STOP
      END
```

FUNCT. EFN SOURCE STATEMENT IFN(S)

```
      SUBROUTINE FUNCT
      COMMON ETAPT(100),PPT(300),ETALG(100),BOB(200),TPT2(100),DPT(100),
     1 Q(2),F1(3),PHJ(401),NE2,XN,XZ,XEND,XPRINT,ENE,DY,YO,ETAI,XI,Y,
     2 EXPO,DPLS,A1,A2,A3,DELTAX,JOP,JOM,VSM,PHIY,DSYSP,DSYSM,SNM,
     3 TSYSP,TSYSM
      COMMON/CONST3/ DELZ,CDELZ,RHOECE,VDX,ENEE,XZ12,XJ12,DELJ12,DELJX2,
     1 DZZ52,C(8),V,NE,CDEL,CDELS,DDELTA,DELTA
      COMMON/FUN1/DPLUS(400),DMINUS(400),EP(400),ETA98(400),T29(400),
     1 B(400),FPEY(400),KFNO(400),DYEY(400),P(400),T29M(400)
      COMMON/FUN2/ B0,B1,KFNO0,KFNO1
      COMMON/FUN3/ ETAQ(51),FQ(51),DTPLQ(51),FPQ(51),PQ(51),DPLUQ(51),
     1 DPMIQ(51),ETA90(51)
      COMMON/IND1/ I,J,M,II,JJ,K
      COMMON/NUMB/ N,NW,NP1,NC,NCOUNT
      REAL NE,NE2,JOP,JOM,KFNO,KFNO0,KFNO1
      READ(5,1000) V,CDEL,RHOECE,NE,CDEL2
 1000 FORMAT(6E13.5)
      READ(5,30) II,JJ,K
   30 FORMAT(3I5)
      READ(5,1000)( ETAPT(I),I=1,II)
      READ(5,1000)(   PPT(I),I=1,II)
      J = II + 1
      N = II + II
      READ(5,1000)(   PPT(I),I=J,N)
      J = N + 1
      N = N + II
      READ(5,1000)(   PPT(I),I=J,N)
      READ(5,1000)(   BOB(I),I=1,JJ)
      J = JJ + 1
      N = JJ + JJ
      READ(5,1000)(   BOB(I),I=J,N)
      READ(5,1000)(  TPT2(I),I=1,K)
      READ(5,1000)(   DPT(I),I=1,K)
      DO 5 I = 1,JJ
      ETALG(I) = ALOG(ETAPT(I))
      N = JJ + I
      BOB(N) = ALOG(BOB(N))
    5 BOB(I) = ALOG(BOB(I))
      READ(5,1000) XN,   XZ,XEND,XPRINT,YO,DY
      N = XN
      NW = N/50
      IF(NW.EQ.0) NW = 1
      NE2= NE*NE*.00375
      VSM = DY - YO
      DY = VSM/XN
      B1 = EXP(YO)
      NP1 = N + 1
      DO 10 I = 1,N
      XI = FLOAT(I) - .5
      Y = YO + XI/XN   *VSM
      EXPO = EXP(Y)
      DYEY(I)   = 1./EXPO/DY
      ETAI      = EXPO - B1
      B0 = ETAI**.125
      ETA98(I) = B0*ETAI
```

NOT REPRODUCIBLE

```
      IF(ETAI    .GT..0001) GO TO 7
      F1(1) = PPT(1)
      F1(2) = PPT(II + 1)
      F1(3) = -.8*ETA98(I)
      KFNO(I) = BOB(1) + BOB(JJ+1)
      GO TO 8
    7 CALL TERP2N(II,3,ETAI    ,ETAPT,PPT,F1)
      XI = ALOG(ETAI    )
      CALL TERP2N(JJ,2, XI,ETALG,BOB,Q)
      KFNO(I) = Q(1) + Q(2)
    8 CALL TERP2N( K,1,F1(2),TPT2,DPT,DPLS)
      P(I) = F1(1)
      T29(I) = 5800./F1(2)*DPLS
      EP(I) = B0/P(I)
      DPLUS(I) = DPLS*2.
      DMINUS(I) = 234.*DPLUS(I)
      ETA98(I) = ETA98(I)*COEL2*2.
      T29M(I) = T29(I)*234.
      FPEY(I) = F1(3)/P(I)*DYEY(I)
      XI    = F1(2)**.5
      KFNO(I) = EXP(KFNO(I) - 32500./F1(2) - 23.718998)/XI
   10 B(I) = NE2/F1(2)/XI    /P(I)/P(I)
      DO 20 I = 1,51
      XI = I-1
      Y = Y0 + XI/50. *VSM
      EXPO = EXP(Y)
      ETAQ(I) = EXPO - B1
      IF(I.EQ.1) ETAQ(1) = 0.
      ETA9Q(I) = ETAQ(I)**1.125
      IF(ETAQ(I).GT..0001) GO TO 27
      F1(1) = PPT(1)
      F1(2) = PPT(II + 1)
      F1(3) = -.8*ETA9Q(I)
      GO TO 28
   27 CALL TERP2N(II,3,ETAQ(I),ETAPT,PPT,F1)
   28 CALL TERP2N( K,1,F1(2),TPT2,DPT,DPLS)
      PQ(I) = F1(1)
      FQ(I) = F1(3)
      FPQ(I)= F1(3)/F1(1)
      DPLUQ(I) = DPLS
      DPMIQ(I) = DPLS*234.
      ETA9Q(I) = ETA9Q(I)*COEL2
   20 DTPLQ(I) = 11600./F1(2)*DPLS
      I = II + 1
      XI = PPT(I)**.5
      B0 = NE2/  PPT(I)/XI/PQ( 1)/PQ( 1) *4.
      KFNO0 = BOB(JJ+1) + BOB(1)
      KFNO0 = EXP(KFNO0 - 32500./  PPT(I) - 23.718998)/XI
      I = II + II
      XI = PPT(I)**.5
      B1 = NE2/  PPT(I)/XI/PQ(51)/PQ(51) *4.
      CALL TERP2N(JJ,2,0.,ETALG,BOB,Q)
      KFNO1 = Q(1) + Q(2)
      KFNO1 = EXP(KFNO1 - 32500./  PPT(I) - 23.718998)/XI
      RETURN
      END
```

INIT. EFN SOURCE STATEMENT - IFN(S) -

```
      SUBROUTINE INITLZ
      COMMON ETAPT(100),PPT(300),ETALG(100),BOB(200),TPT2(100),DPT(100),
     1 D(2),F1(3),PHJ(401),NE2,XN,XZ,XEND,XPRINT,ENE,DY,YO,ETAI,XI,Y,
     2 EXPO,DPLS,A1,A2,A3,DELTAX,JOP,JOM,VSM,PHIY,DSYSP,DSYSM,SNM,
     3 TSYSP,TSYSM
      COMMON/CONST1/XJ,DELJX,ZJ,DZ25,DELJ
      COMMON/CONST2/P1,P2,P3,P4,P5,P6
      COMMON/CONST3/ DELZ,CDEL2,RHOECE,VDX,ENEE,XZ12,XJ12,DELJ12,DELJX2,
     1 DZ252,C(8),V,NE,CDEL,CDELE,DDELTA,DELTA
      COMMON/INTLZ/PHIJ(401),SPJ(401),SMJ(401),JPJ(401),JMJ(401),
     1 DSPJ(401),DSMJ(401),SPJH(400),SMJH(400),PHIJH(400)
      COMMON/INDI/ I,J,M,II,JJ,K
      COMMON/INDX/ IDX(6)
      COMMON/NUMB/ N,NW,NP1,NC,NCOUNT
      REAL NE,NE2,JOP,JOM,JPJ,JMJ
      READ(5,10) PHIJ(1),P1,P2,P3,P4,P5,P6
   10 FORMAT(7E10.5)
      READ(5,11) IDX(1),IDX(2),IDX(3),IDX(4),IDX(5),IDX(6)
   11 FORMAT(6I5)
      XEND = XEND**.3
      CDEL2 = CDEL*CDEL
      ENE = 1.6E-19*NE
      ENEE = ENE/8.85E-14*.5
      CDELE= CDEL/ENE
      DO 21 I = 1,N
      PHIJ(I) = PHIJ(1)
      PHIJH(I) = PHIJ(1) + PHIJ(1)
      SPJH(I) = 0.
      SMJH(I) = 0.
       SPJ(I) = 0.
       SMJ(I) = 0.
       JPJ(I) = 0.
       JMJ(I) = 0.
      DSPJ(I) = 0.
   21 DSMJ(I) = 0.
      PHIJ(NP1) = PHIJ(1)
       SPJ(NP1) = 0.
       SMJ(NP1) = 0.
       JPJ(NP1) = 0.
       JMJ(NP1) = 0.
      DSPJ(NP1) = 0.
      DSMJ(NP1) = 0.
      XJ = 0.
      DELJX = 0.
      DELJ = 0.
      ZJ = 0.
      DZ25 = 0.
      NC = 0
      NCOUNT = XZ/XPRINT
      M = XZ
      DELZ = XEND/XZ
      VDX = .2*V/DELZ
      DDELTA = CDEL*DELZ
      RETURN
      END
```

NOT REPRODUCIBLE

STEP. - EFN SOURCE STATEMENT - IFN(S) -

```
      SUBROUTINE XSTEP
      COMMON/CONST1/XJ,DELJX,ZJ,DZ25,DELJ
      COMMON/CONST3/ DELZ,CDEL2,RHOECE,VDX,ENEE,XZ12,XJ12,DELJ12,DELJX2,
     1 DZ252,C(8),V,NE,CDEL,CDELE,DDELTA,DELTA
      COMMON/IND1/ I,J,M,II,JJ,K
      REAL NE
      XZ12 = ZJ + DELZ
      DELTA = CDEL*XZ12
      DELJ12 = DELTA*DELTA
      XJ12 = XZ12**1.25
      DELJX2 = DELJ12/XJ12
      DZ252 = DELJ12/(XZ12**.25)
      C( 8) = DELJ12 - DELJ
      C( 1) = DELJ12/RHOECE
      C( 2) = VDX*(3.*DZ252 - DZ25)
      IF(J.EQ.1) C(2) =      C(2)/.75
      C( 3) = V    *DELJX2
      C( 4) = ENEE*DELJ12
      C( 5) = V    *(DELJX2 - DELJX)
      C( 6) = C( 8)/RHOECE
      C( 7) = VDX*(DZ252 + DZ25)
      C( 8) = ENEE*C( 8)
      RETURN
      END
```

BOUND. - EFN SOURCE STATEMENT - IFN(S) -

```
      SUBROUTINE BOUND0
      COMMON/FUN3/ ETAQ(51),FQ(51),DTPLQ(51),FPQ(51),PQ(51),DPLUQ(51),
     1 DPMIQ(51),ETA9Q(51)
      COMMON/INTLZ/PHIJ(401),SPJ(401),SMJ(401),JPJ(401),JMJ(401),
     1 DSPJ(401),DSMJ(401),SPJH(400),SMJH(400),PHIJH(400)
      COMMON/MAT1/X(25),D(30),BB(25)
      COMMON/CONST3/ DELZ,CDEL2,RHOECE,VDX,ENEE,XZ12,XJ12,DELJ12,DELJX2,
     1 DZ252,C(8),V,NE,CDEL,CDELE,DDELTA,DELTA
      COMMON/CONST2/P1,P2,P3,P4,P5,P6
      COMMON/CONST1/XJ,DELJX,ZJ,DZ25,DELJ
      REAL JPJ,JMJ,NE
      X(3) = 10.E10
      X(9) = 10.E10
      D(28) = 0.
      D(29) = 0.
      RETURN
      ENTRY BOUND1
      X( 1) = 10.E10
      X( 7) = 10.E10
      X(25) = 10.E10
      D(26) = 0.
      D(27) = 0.
      D(30) = P1*DDELTA*X(1)
      RETURN
      END
```

```
      SUBROUTINE EMPTY
      COMMON/CONST3/ DELZ,CDEL2,RHOECE,VDX,ENEE,XI12,XJ12,DELJ12,DELJX2,
     1 DZ252,C(8),V,NE,CDEL,CDELE,DDELTA,DELTA
      COMMON/FUN1/DPLUS(400),DMINUS(400),EP(400),ETA98(400),T29(400),
     1 B(400),FPEY(400),KFNO(400),DYEY(400),P(400),T29M(400)
      COMMON/INTLZ/PHIJ(401),SPJ(401),SMJ(401),JPJ(401),JMJ(401),
     1 DSPJ(401),DSMJ(401),SPJH(400),SMJH(400),PHIJH(400)
      COMMON/MAT1/X(25),D(30),BB(25)
      COMMON/IND1/ I,J,M,II,JJ,K
      COMMON/INDX/ IDX(6)
      REAL NE,             JPJ,JMJ,KFNO
      DO 50 II = 1,25
      BB(II) = 0.
       X(II) = 0.
 50    D(II) = 0.
      RETURN
      ENTRY FIRST
      K = I-IDX(1)
      D(26) = DDELTA*ETA98(K)
      X(11) = DPLUS(K) + DELTA*ETA98(K)
      D(26) = D(26)/(X(11) - D(26))
      X(1) = T29(K)/X(11)
      X(21) = X(1)*(SPJH(K) + DSPJ(K+1) + DSPJ(K))
      X(1) = X(1)*PHIJH(K)
      X(11) = -P(K)/X(11)
      D(26) = (X(1)*SPJH(K) + X(11)*(JPJ(K+1)+JPJ(K)))*D(26)
      IF(IDX(1).EQ.1) GO TO 80
      D(1) = X(1) + DYEY(K)
      X(1) = X(1) - DYEY(K)
      D(11) = X(11)
      D(21) = X(21)
      RETURN
 80   BB(1) = X(1) - DYEY(K)
       X(1) = X(1) + DYEY(K)
      BB(11) = X(11)
      BB(21) = X(21)
      RETURN
      ENTRY SECOND
      K = I - IDX(2)
      D(27) = DDELTA*ETA98(K)
      X(17) = DMINUS(K) + DELTA*ETA98(K)
      D(27) = D(27)/(X(17)-D(27))
      X(7) = -T29M(K)/X(17)
      X(22) = X(7)*(SMJH(K) + DSMJ(K+1) + DSMJ(K))
      X(7) = X(7)*PHIJH(K)
      X(17) = -P(K)/X(17)
      D(27) = (X(7)*SMJH(K) + X(17)*(JMJ(K+1)+JMJ(K)))*D(27)
      IF(IDX(2).EQ.1) GO TO 10
       D(7) = X(7) + DYEY(K)
       X(7) = X(7) - DYEY(K)
       D(17) = X(17)
       D(22) = X(22)
      RETURN
 10   BB(7) = X(7) - DYEY(K)
       X(7) = X(7) + DYEY(K)
```

```
      BB(17) = X(17)
      BB(22) = X(22)
      RETURN
      ENTRY THIRD
      K = I - IDX(3)
      X(8) = B(K)*C(1)
      X(3) = -X(8)*SMJH(K) - C(2)*EP(K)
      X(8) = -X(8)*(SPJH(K) + DSPJ(K+1) + DSPJ(K))
      X(13) = C(3)*FPEY(K)
      D(23) = C(5)*FPEY(K)*(SPJ(K+1) - SPJ(K))   +C(6)*(-KFND(K) + B(K)*
     1 SPJH(K)*SMJH(K))   -C(7)*EP(K)*(DSPJ(K+1) + DSPJ(K))
      IF(IDX(3).EQ.1) GO TO 20
       D(3) = X(3) - X(13)
      X(3) = X(3) + X(13)
       D(8) = X(8)
       D(13) = DYEY(K)
       X(13) =-DYEY(K)
      RETURN
   20 BB(3) = X(3) + X(13)
      X(3) = X(3) - X(13)
      BB(8) = X(8)
      BB(13) =-DYEY(K)
       X(13) = DYEY(K)
      RETURN
      ENTRY FOURTH
      K = I - IDX(4)
      X(4) = B(K)*C(1)
      X(9) = -X(4)*SPJH(K) - C(2)*EP(K)
      X(4) = -X(4)*(SMJH(K) + DSMJ(K+1) + DSMJ(K))
      X(19) = C(3)*FPEY(K)
      D(29) = C(5)*FPEY(K)*(SMJ(K+1) - SMJ(K))   +C(6)*(-KFND(K) + B(K)*
     1 SPJH(K)*SMJH(K))   -C(7)*EP(K)*(DSMJ(K+1) + DSMJ(K))
      IF(IDX(4).EQ.1) GO TO 30
       D(4) = X(4)
       D(9) = X(9) - X(19)
       X(9) = X(9) + X(19)
       D(19) = DYEY(K)
       X(19) =-DYEY(K)
      RETURN
   30 BB(4) = X(4)
      BB(9) = X(9)  + X(19)
       X(9) = X(9) - X(19)
      BB(19) =-DYEY(K)
       X(19) = DYEY(K)
      RETURN
      ENTRY FIFTH
      K = I - IDX(5)
      X(5) = C(4)/P(K)
      X(10)= -X(5)
      D(30) = -C(8)/P(K)*(SPJH(K)-SMJH(K))
      IF(IDX(5).EQ.1) GO TO 40
       D(5) = X(5)
       D(10) = X(10)
       D(25) = DYEY(K)
       X(25) =-DYEY(K)
      RETURN
   40 BB(5) = X(5)
      BB(10) = X(10)
      BB(25) =-DYEY(K)
       X(25) = DYEY(K)
  100 RETURN
      END
```

```
      SUBROUTINE REST(Z)
      DIMENSION Z(10000)
      COMMON ETAPT(100),PPT(300),ETALG(100),BOB(200),TPT2(100),DPT(100),
     1 Q(2),F1(3),PHJ(401),NE2,XN,XZ,XEND,XPRINT,ENE,DY,YO,ETAI,XI,Y,
     2 EXPO,DPLS,A1,A2,A3,DELTAX,JOP,JOM,VSM,PHIY,DSYSP,DSYSM,SNM,
     3 TSYSP,TSYSM
      COMMON/CONST1/XJ,DELJX,ZJ,DZ25,DELJ
      COMMON/CONST2/P1,P2,P3,P4,P5,P6
      COMMON/CONST3/ DELZ,CDEL2,RHOECE,VDX,ENEE,XZ12,XJ12,DELJ12,DELJX2,
     1 DZ252,C(8),V,NE,CDEL,CDELE,DDELTA,DELTA
      COMMON/FUN1/DPLUS(400),DMINUS(400),EP(400),ETA98(400),T29(400),
     1 B(400),FPEY(400),KFNO(400),DYEY(400),P(400),T29M(400)
      COMMON/FUN3/ ETAO(51),FO(51),DTPLQ(51),FPO(51),PQ(51),DPLUQ(51),
     1 DPMIQ(51),ETA9C(51)
      COMMON/INTLZ/PHIJ(401),SPJ(401),SMJ(401),JPJ(401),JMJ(401),
     1 DSPJ(401),DSMJ(401),SPJH(400),SMJH(400),PHIJH(400)
      COMMON/MAT2/XX(2005)
      COMMON/IND1/ I,J,M,II,JJ,K
      COMMON/INDX/ IDX(6)
      COMMON/NUMB/ N,NW,NP1,NC,NCOUNT
      REAL JOP,JOM,KFNO,JPJ,JMJ,NE,NE2
C     READY FOR BACK MULTIPLICATION
      JJ = 5*N
      II = 25*N
      DO 250 I = 1,N
      II = II - 25
      JJ5 = JJ
      JJ = JJ-5
      DO 250 K = 1,5
      L1 = JJ + K
      L2 = II + K
      DO 250 L = 1,5
      L3 = JJ5 + L
      XX(L1) = XX(L1) - Z(L2)*XX(L3)
  250 L2 = L2 + 5
C     ANSWERS ARE IN XX ARRAY. WRITE AND UPDATE BEFORE STEPPING X.
      K = -4
      DO 260 I = 1,NP1
      K = K +5
      DSPJ(I) = XX(K)
      DSMJ(I) = XX(K+1)
      SPJ(I) = SPJ(I) + DSPJ(I)
      SMJ(I) = SMJ(I) + DSMJ(I)
      JPJ(I) = JPJ(I) + XX(K+2)
      JMJ(I) = JMJ(I) + XX(K+3)
      PHIJ(I) = PHIJ(I) + XX(K+4)
      IF(I.EQ.1) GO TO 260
      II = I-1
      SPJH(II) = SPJ(I) + SPJ(II)
      SMJH(II) = SMJ(I) + SMJ(II)
      PHIJH(II) = PHIJ(I) + PHIJ(II)
  260 CONTINUE
      XJ = XJ12
      DELJ = DELJ12
      DELJX = DELJX2
```

```
      ZJ = XZ12
      DZ25 = DZ252
      NC = NC + 1
      IF(NC.LT.NCOUNT) RETURN
      NC = 0
C     WRITE HEADING
      WRITE(5,1001) XJ,DELZ,N
 1001 FORMAT(5H1X = ,F7.2,5X,10HDELTA Z = ,F9.4,5X,16HETA INTERVALS = ,
     1I3,/88X,5HKPLUS,4X,6HD PLUS,9H  D MINUS,3X,6HDTPLUS,9H  DTMINUS /
     23X,3HETA,5X,6HNMINUS,4X,5HSPLUS,4X,6HSMINUS,5X,5HJPLUS,4X,6HJMINUS
     3,4X,6HPHIETA,5X,3HPHI,7X,2H-V,5X,5H*PHIY,4X,5H*SY/S,4X,5H*SY/S,
     44X,5H*SY/S,4X,5H*SY/S //)
C  COMPUTE PHI ( = PHJ)
      IF(IDX(5).EQ.1) GO TO 251
      PHJ(NP1) = 0.
      DO 252 I = 1,N
      K = NP1 - I
  252 PHJ(K) = PHJ(K+1) - .5*PHIJH(K)/DYEY(K)
      GO TO 275
  251 PHJ(1) = 0.
      DO 253 I = 1,N
  253 PHJ(I+1) = PHJ(I) + .5*PHIJH(I)/DYEY(I)
  275 A1 = V*DELJX
      A2 =CDELE*ZJ
      DELTAX = -DELTA/XJ*V
      II = 0
      DO 300 I = 1,NP1,NW
      II = II + 1
      A3 = A1*FPQ(II)
      JOP =(JPJ(I) - A3*SPJ(I))/A2
      JOM =(JMJ(I) - A3*SMJ(I))/A2
      A3 = PQ(II)/DELTA
      VSM = DELTAX*FQ(II)
      SNM = NE*SNJ(I)/PQ(II)
      PHIY = DTPLQ(II )*PHIJ(I)/DELTA
      DSYSP = -PHIY + A3*JPJ(I)/SPJ(I)
      DSYSM =  PHIY*234. + A3*JMJ(I)/SMJ(I)
      A3 = ETA9Q(II)*DELTA
      TSYSP = DPLUQ(II) + A3
      TSYSM = DPMIQ(II) + A3
      DSYSP = DSYSP*DPLUQ(II)/TSYSP
      DSYSM = DSYSM*DPMIQ(II)/TSYSM
      TSYSP = DSYSP*A3/DPLUQ(II)
      TSYSM = DSYSM*A3/DPMIQ(II)
      IF(SPJ(I).NE.0.) GO TO 280
      DSYSP = 1.E38
      TSYSP = 1.E38
  280 IF(SMJ(I).NE.0.) GO TO 300
      DSYSM = 1.E38
      TSYSM = 1.E38
  300 WRITE(5,1003) ETAQ(II),SNM,SPJ(I),SMJ(I),JOP,JOM,PHIJ(I),PHJ(I),
     1 VSM,PHIY,DSYSP,DSYSM,TSYSP,TSYSM
 1003 FORMAT(1P2E9.2,1P5E10.3,1P7E9.2)
      IF(XJ.GT.P6) STOP
      RETURN
      END
```

XDCOR. EFN SOURCE STATEMENT IFN(S)

```
      SUBROUTINE XDCORR(Z)
      DIMENSION Z(10000)
      COMMON/MAT1/X(25),D(30),BB(25)
      COMMON/MAT2/XX(2005)
      COMMON/IND1/ I,J,M,II,JJ,K
      COMMON/INDX/ IDX(6)
      JJ = 25*(I-2)
      JJ0= 5*I - 9
      JJ1 = JJ0+ 1
      JJ2 = JJ0+ 2
      JJ3 = JJ0+ 3
      JJ4 = JJ0+ 4
      IF(IDX(1).EQ.0) GO TO 7
      DO  5 II = 1,21,5
      K = JJ + II
    5 X(II) = X(II)-BB(1)*Z(K   )-BB(11)*Z(K+2)-BB(21)*Z(K+4)
      D(26)=D(26)-BB(1)*XX(JJ0)-BB(11)*XX(JJ2)-BB(21)*XX(JJ4)
    7 IF(IDX(2).EQ.0) GO TO 12
      DO 10 II = 2,22,5
      K = JJ + II
   10 X(II) = X(II)-BB(7)*Z(K   )-BB(17)*Z(K+2)-BB(22)*Z(K+3)
      D(27)=D(27)-BB(7)*XX(JJ1)-BB(17)*XX(JJ3)-BB(22)*XX(JJ4)
   12 IF(IDX(3).EQ.0) GO TO 17
      DO 15 II = 3,23,5
      K = JJ + II
   15 X(II) = X(II)-BB(3)*Z(K-2)-BB( 8)*Z(K-1)-BB(13)*Z(K)
      D(28)=D(28)-BB(3)*XX(JJ0)-BB( 8)*XX(JJ1)-BB(13)*XX(JJ2)
   17 IF(IDX(4).EQ.0) GO TO 22
      DO 20 II = 4,24,5
      K = JJ + II
   20 X(II) = X(II)-BB(4)*Z(K-3)-BB( 9)*Z(K-2)-BB(19)*Z(K)
      D(29)=D(29)-BB(4)*XX(JJ0)-BB( 9)*XX(JJ1)-BB(19)*XX(JJ3)
   22 IF(IDX(5).EQ.0) RETURN
      DO 25 II = 5,25,5
      K = JJ + II
   25 X(II) = X(II)-BB(5)*Z(K-4)-BB(10)*Z(K-3)-BB(25)*Z(K)
      D(30)=D(30)-BB(5)*XX(JJ0)-BB(10)*XX(JJ1)-BB(25)*XX(JJ4)
      RETURN
      END
```

ENDO. EFN SOURCE STATEMENT IFN(S)

```
      SUBROUTINE ENDO
      COMMON/INDX/ IDX(6)
      IF(IDX(1).EQ.0) CALL FIRST
      IF(IDX(2).EQ.0) CALL SECOND
      IF(IDX(3).EQ.0) CALL THIRD
      IF(IDX(4).EQ.0) CALL FOURTH
      IF(IDX(5).EQ.0) CALL FIFTH
      CALL BOUND0
      RETURN
      ENTRY END1
      IF(IDX(1).EQ.1) CALL FIRST
      IF(IDX(2).EQ.1) CALL SECOND
      IF(IDX(3).EQ.1) CALL THIRD
      IF(IDX(4).EQ.1) CALL FOURTH
      IF(IDX(5).EQ.1) CALL FIFTH
      CALL BOUND1
      RETURN
      END
```

Table C-I. Sample output.

X = 13.37 DELTA Z = 0.7962 ETA INTERVALS = 400

ETA	NMINUS	SPLUS	SMINUS	JPLUS	JMINUS	PHIETA	PHI	-V	KPLUS *PHIY	D PLUS *SY/S	D MINUS *SY/S	DTPLUS *SY/S	UTMINUS *SY/S
0.00E-39	0.00E-39	0.000E-39	0.000E-39	8.883E-06	8.919E-06	1.835E 01	-1.58E 00	0.00E-39	8.59E 02	1.00E 38	1.00E 38	1.00E 38	1.00E 38
8.35E-04	1.09E 07	2.814E-04	1.537E-06	5.119E-06	5.154E-06	1.830E 01	-1.56E 00	1.05E 00	8.37E 02	1.15E 04	3.15E 06	1.89E 02	2.23E 03
1.75E-03	1.94E 07	5.865E-04	3.221E-06	3.356E-06	3.389E-06	1.818E 01	-1.55E 00	2.43E 00	9.23E 02	3.66E 03	1.28E 06	1.43E 02	2.14E 03
2.76E-03	2.69E 07	7.037E-04	4.372E-06	2.264E-06	2.294E-06	1.799E 01	-1.53E 00	4.01E 00	8.08E 02	1.49E 03	7.21E 05	9.67E 02	2.06E 03
3.88E-03	3.36E 07	7.533E-04	5.382E-06	1.494E-06	1.511E-06	1.776E 01	-1.51E 00	5.78E 00	7.91E 02	5.81E 02	4.66E 05	5.61E 02	1.97E 03
5.10E-03	3.97E 07	7.890E-04	6.294E-06	9.015E-07	9.244E-07	1.749E 01	-1.49E 00	7.82E 00	7.73E 02	1.45E 02	3.24E 05	2.01E 02	1.92E 03
6.45E-03	4.53E 07	7.903E-04	7.125E-06	4.230E-07	4.413E-07	1.718E 01	-1.46E 00	1.01E 01	7.54E 02	-8.59E 01	2.35E 05	-1.54E 02	1.85E 03
7.94E-03	5.13E 07	7.764E-04	7.892E-06	3.894E-08	5.196E-08	1.684E 01	-1.44E 00	1.27E 01	7.34E 02	-2.09E 02	1.76E 05	-4.91E 02	1.78E 03
9.57E-03	5.65E 07	7.517E-04	8.606E-06	-2.695E-07	-2.622E-07	1.647E 01	-1.41E 00	1.55E 01	7.13E 02	-2.70E 02	1.36E 05	-7.91E 02	1.72E 03
1.14E-02	6.17E 07	7.205E-04	9.281E-06	-5.134E-07	-5.172E-07	1.607E 01	-1.38E 00	1.87E 01	6.91E 02	-2.96E 02	1.09E 05	-1.00E 03	1.59E 03
1.34E-02	6.65E 07	6.957E-04	9.927E-06	-7.046E-07	-7.099E-07	1.565E 01	-1.35E 00	2.25E 01	6.69E 02	-3.02E 02	8.84E 04	-1.24E 03	1.65E 03
1.55E-02	7.19E 07	6.480E-04	1.057E-05	-8.506E-07	-8.629E-07	1.520E 01	-1.32E 00	2.65E 01	6.45E 02	-2.95E 02	7.42E 04	-1.54E 03	1.70E 03
1.79E-02	7.72E 07	6.102E-04	1.121E-05	-9.591E-07	-9.787E-07	1.474E 01	-1.28E 00	3.09E 01	6.20E 02	-2.80E 02	6.42E 04	-1.66E 03	1.76E 03
2.06E-02	8.24E 07	5.726E-04	1.187E-05	-1.037E-06	-1.065E-06	1.425E 01	-1.24E 00	3.58E 01	5.95E 02	-2.61E 02	5.70E 04	-1.97E 03	1.85E 03
2.35E-02	8.85E 07	5.360E-04	1.257E-05	-1.095E-06	-1.130E-06	1.374E 01	-1.20E 00	4.14E 01	5.70E 02	-2.41E 02	5.15E 04	-2.18E 03	1.99E 03
2.67E-02	9.55E 07	5.066E-04	1.332E-05	-1.135E-06	-1.179E-06	1.322E 01	-1.16E 00	4.75E 01	5.43E 02	-2.21E 02	4.74E 04	-2.40E 03	2.15E 03
3.02E-02	1.02E 08	4.666E-04	1.415E-05	-1.151E-06	-1.214E-06	1.267E 01	-1.11E 00	5.43E 01	5.16E 02	-2.02E 02	4.42E 04	-2.52E 03	2.36E 03
3.41E-02	1.10E 08	4.343E-04	1.507E-05	-1.176E-06	-1.238E-06	1.211E 01	-1.07E 00	6.17E 01	4.89E 02	-1.83E 02	4.17E 04	-2.68E 03	2.50E 03
3.84E-02	1.20E 08	4.036E-04	1.611E-05	-1.182E-06	-1.253E-06	1.152E 01	-1.02E 00	6.98E 01	4.62E 02	-1.66E 02	3.97E 04	-2.83E 03	2.87E 03
4.31E-02	1.30E 08	3.745E-04	1.731E-05	-1.183E-06	-1.263E-06	1.093E 01	-9.63E-01	7.88E 01	4.35E 02	-1.50E 02	3.79E 04	-2.96E 03	3.21E 03
4.82E-02	1.43E 08	3.471E-04	1.869E-05	-1.179E-06	-1.269E-06	1.031E 01	-9.08E-01	8.88E 01	4.07E 02	-1.36E 02	3.61E 04	-3.14E 03	3.55E 03
5.39E-02	1.54E 08	3.213E-04	2.029E-05	-1.172E-06	-1.273E-06	9.679E 00	-8.51E-01	9.96E 01	3.79E 02	-1.23E 02	3.42E 04	-3.29E 03	3.90E 03
6.02E-02	1.75E 08	2.969E-04	2.215E-05	-1.162E-06	-1.273E-06	9.032E 00	-7.92E-01	1.11E 02	3.50E 02	-1.11E 02	3.20E 04	-3.45E 03	4.25E 03
6.71E-02	1.93E 08	2.739E-04	2.429E-05	-1.151E-06	-1.273E-06	8.371E 00	-7.32E-01	1.24E 02	3.21E 02	-1.00E 02	2.96E 04	-3.61E 03	4.57E 03
7.47E-02	2.19E 08	2.522E-04	2.675E-05	-1.134E-06	-1.271E-06	7.700E 00	-6.71E-01	1.38E 02	2.92E 02	-9.03E 01	2.70E 04	-3.76E 03	4.83E 03
8.31E-02	2.47E 08	2.319E-04	2.954E-05	-1.126E-06	-1.268E-06	7.022E 00	-6.07E-01	1.54E 02	2.62E 02	-8.14E 01	2.41E 04	-3.97E 03	5.03E 03
9.23E-02	2.79E 08	2.128E-04	3.266E-05	-1.117E-06	-1.263E-06	6.341E 00	-5.48E-01	1.70E 02	2.34E 02	-7.33E 01	2.11E 04	-4.16E 03	5.12E 03
1.02E-01	3.15E 08	1.948E-04	3.608E-05	-1.098E-06	-1.258E-06	5.662E 00	-4.87E-01	1.86E 02	2.06E 02	-6.61E 01	1.80E 04	-4.36E 03	5.07E 03
1.14E-01	3.54E 08	1.778E-04	3.971E-05	-1.082E-06	-1.250E-06	4.993E 00	-4.28E-01	2.08E 02	1.79E 02	-5.99E 01	1.48E 04	-4.60E 03	4.35E 03
1.26E-01	3.93E 08	1.613E-04	4.343E-05	-1.067E-06	-1.241E-06	4.343E 00	-3.70E-01	2.30E 02	1.53E 02	-5.41E 01	1.16E 04	-4.85E 03	4.45E 03
1.39E-01	4.40E 08	1.468E-04	4.705E-05	-1.051E-06	-1.229E-06	3.720E 00	-3.16E-01	2.53E 02	1.27E 02	-4.89E 01	8.60E 03	-5.17E 03	3.97E 03
1.54E-01	4.84E 08	1.327E-04	5.037E-05	-1.034E-06	-1.214E-06	3.133E 00	-2.65E-01	2.78E 02	1.07E 02	-4.40E 01	5.92E 03	-5.41E 03	3.11E 03
1.71E-01	5.24E 08	1.195E-04	5.298E-05	-1.017E-06	-1.196E-06	2.589E 00	-2.18E-01	3.04E 02	8.66E 01	-3.96E 01	3.52E 03	-5.72E 03	2.18E 03
1.89E-01	5.50E 08	1.071E-04	5.476E-05	-9.989E-07	-1.175E-06	2.098E 00	-1.76E-01	3.32E 02	6.86E 01	-3.55E 01	1.50E 03	-6.07E 03	1.10E 03
2.09E-01	5.81E 08	9.552E-05	5.544E-05	-9.809E-07	-1.151E-06	1.663E 00	-1.39E-01	3.61E 02	5.31E 01	-3.19E 01	-1.16E 02	-6.45E 03	-9.54E 01
2.30E-01	6.01E 08	8.476E-05	5.490E-05	-9.626E-07	-1.125E-06	1.289E 00	-1.07E-01	3.92E 02	4.02E 01	-2.87E 01	-1.32E 03	-6.69E 03	-1.35E 03
2.55E-01	6.05E 08	7.473E-05	5.314E-05	-9.441E-07	-1.096E-06	9.744E-01	-7.99E-02	4.24E 02	2.96E 01	-2.57E 01	-2.17E 03	-7.37E 03	-2.55E 03
2.81E-01	5.95E 08	6.544E-05	5.028E-05	-9.253E-07	-1.066E-06	7.185E-01	-5.76E-02	4.57E 02	2.13E 01	-2.31E 01	-2.70E 03	-7.91E 03	-3.75E 03
3.10E-01	5.77E 08	5.686E-05	4.651E-05	-9.066E-07	-1.035E-06	5.145E-01	-3.98E-02	4.90E 02	1.48E 01	-2.07E 01	-2.99E 03	-8.62E 03	-5.26E 03
3.42E-01	5.47E 08	4.894E-05	4.209E-05	-8.879E-07	-1.004E-06	3.565E-01	-2.60E-02	5.24E 02	9.90E 00	-1.85E 01	-3.09E 03	-9.25E 03	-6.59E 03
3.78E-01	5.09E 08	4.168E-05	3.727E-05	-8.695E-07	-9.730E-07	2.373E-01	-1.56E-02	5.57E 02	6.34E 00	-1.65E 01	-3.05E 03	-1.01E 04	-7.95E 03
4.17E-01	4.66E 08	3.504E-05	3.229E-05	-8.515E-07	-9.434E-07	1.495E-01	-8.21E-03	5.90E 02	3.85E 00	-1.49E 01	-2.94E 03	-1.11E 04	-9.40E 03
4.59E-01	4.19E 08	2.900E-05	2.735E-05	-8.343E-07	-9.155E-07	8.698E-02	-3.24E-03	6.18E 02	2.16E 00	-1.35E 01	-2.80E 03	-1.24E 04	-1.10E 04
5.06E-01	3.70E 08	2.353E-05	2.261E-05	-8.183E-07	-8.898E-07	4.427E-02	-2.32E-04	6.34E 02	1.05E 00	-1.21E 01	-2.62E 03	-1.39E 04	-1.28E 04
5.58E-01	3.22E 08	1.864E-05	1.816E-05	-8.039E-07	-8.665E-07	1.677E-02	1.28E-03	6.33E 02	3.80E-01	-1.09E 01	-2.42E 03	-1.56E 04	-1.50E 04
6.15E-01	2.77E 08	1.430E-05	1.400E-05	-7.909E-07	-8.455E-07	7.573E-04	1.73E-03	6.18E 02	1.63E-02	-9.93E 00	-2.24E 03	-1.93E 04	-1.79E 04
6.78E-01	2.22E 08	1.049E-05	1.042E-05	-7.798E-07	-8.273E-07	-6.725E-03	1.50E-03	5.82E 02	-1.35E-01	-9.16E 00	-2.09E 03	-2.24E 04	-2.13E 04
7.43E-01	1.71E 08	7.157E-06	7.169E-06	-7.712E-07	-8.125E-07	-8.179E-03	9.50E-04	5.11E 02	-1.53E-01	-8.70E 00	-2.00E 03	-2.67E 04	-2.82E 04
8.24E-01	1.19E 08	4.312E-06	4.345E-06	-7.652E-07	-8.016E-07	-5.878E-03	4.07E-04	3.89E 02	-1.00E-01	-8.49E 00	-2.06E 03	-4.16E 04	-4.09E 04
9.08E-01	6.31E 07	1.723E-06	1.950E-06	-7.624E-07	-7.953E-07	-2.163E-03	7.18E-05	1.75E 02	-3.29E-02	-1.13E 01	-2.63E 03	-7.68E 04	-7.63E 04
1.00E 00	0.00E-39	0.000E-39	0.000E-39	-7.621E-07	-7.937E-07	0.000E-39	0.00E-39	-1.98E 02	0.00E-39	1.00E 38	1.00E 38	1.00E 38	1.00E 38

NOT REPRODUCIBLE

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1. ORIGINATING ACTIVITY (Corporate author) Harry Diamond Laboratories Washington, D.C. 20438	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3. REPORT TITLE

Ion and Electron Distributions in the Boundary Layer of Hypersonic Vehicles for Chemical Nonequilibrium Flow--Part II: Method of Solution and Computer Program

4. DESCRIPTIVE NOTES (Type of report and inclusive dates)

5. AUTHOR(S) (First name, middle initial, last name)

Arthur Hausner

6. REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
November 1971	48	3

8a. CONTRACT OR GRANT NO.	9a. ORIGINATOR'S REPORT NUMBER(S)
b. PROJECT NO. DA-1T061101A91A	HDL-TR-1567
c. AMCMS Code: 501A.11.84400	9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)
d. HDL Proj: 39839	

10. DISTRIBUTION STATEMENT

Approved for public release; distribution unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
	USAMC

13. ABSTRACT

Various computational aspects have been investigated to numerically solve charge-conservation equations and Poisson's equation for the electric field yielding ion and electron distributions. These equations were derived and presented by the Harry Diamond Laboratories as part I of this study. The computational aspects, reported herein as part II of the study, includes: (1) transformations to reduce the steep slopes of the input functions and to simplify the solutions of the equations; (2) linearization of the equations to permit use of matrix methods in their solution; (3) derivation of small-value, asymptotic solutions to provide starting conditions in the matrix solution; (4) a computer program listing, description, and sample output; and (5) descriptions of an independent check solution and other checks to confirm validity of the results. The computer program is written to accommodate any consistent set of boundary conditions. Although the equations are linearized, the nonlinear terms are approximated in a way to insure rapid convergence of solutions to the exact equations.

DD FORM 1 NOV 65 1473 REPLACES DD FORM 1473, 1 JAN 64, WHICH IS OBSOLETE FOR ARMY USE.

Unclassified
Security Classification

14. KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Ionized flow	8	3				
Plasma	8	3				
Hypersonic vehicles	8	3				
Electron density distribution	8	3				
Nonequilibrium chemical flow	8	3				
Partial differential equations	8	3				