NASA-CR-3267 19800015748

NASA Contractor Report 3267

Application of Advanced Computational Procedures for Modeling Solar-Wind Interactions With Venus - Theory and Computer Code

Stephen S. Stahara, Daniel Klenke,
Barbara C. Trudinger, and John R. Spreiter

CONTRACT NASW-3182
MAY 1980

NASA

NASA Contractor Report 3267

Application of Advanced Computational Procedures for Modeling Solar-Wind Interactions With Venus - Theory and Computer Code

Stephen S. Stahara, Daniel Klenke,
Barbara C. Trudinger, and John R. Spreiter
Nielsen Engineering & Research, Inc.
Mountain View, California

Prepared for
NASA Headquarters
under Contract NASW-3182

National Aeronautics
and Space Administration

Scientific and Technical
Information Office

1980

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

LIST OF ILLUSTRATIONS (Concluded)

APPLICATION OF ADVANCED COMPUTATIONAL PROCEDURES FOR MODELING SOLAR-WIND INTERACTIONS WITH VENUS - THEORY AND COMPUTER CODE

by

Stephen S. Stahara, Daniel Klenke,
Barbara C. Trudinger, and John R. Spreiter

SUMMARY

Advanced computational procedures are developed and applied to the prediction of solar-wind interaction with nonmagnetic terrestrial-planet atmospheres, with particular emphasis to Venus. The theoretical method is based on a single-fluid, steady, dissipationless, magnetohydrodynamic continuum model, and is appropriate for the calculation of axisymmetric, supersonic, super-Alfvénic solar-wind flow past terrestrial planets. The procedures, which consist of finite-difference codes to determine the gasdynamic properties and a variety of special-purpose codes to determine the frozen magnetic field, streamlines, contours, plots, etc. of the flow, are organized into one computational program which has been extensively documented and is presented in a general user's manual included as part of this report.

Theoretical results based upon these procedures are reported for a wide variety of solar-wind conditions and ionopause obstacle shapes. Plasma and magnetic-field comparisons in the ionosheath are also provided with actual spacecraft data obtained by the Pioneer-Venus Orbiter. These results have verified the appropriateness of the basic theoretical model, and have indicated the importance of accounting for the variable oncoming direction of the interplanetary solar wind.

INTRODUCTION

The magnetohydrodynamic models (refs. 1-9) of solar-wind interaction with planetary magneto/ionospheres and their associated calculations of the detailed flow and magnetic-field properties provide the basis of the theoretical understanding and interpretation of phenomena occurring in space around terrestrial planets from the viewpoint of a fluid rather than particle description of the flow. The general value and usefulness of results based on these models are now well established, and have advanced to the point where theoretical calculations can be used to predict important planetary and magnetic-field characteristics.

Prior to the previous work reported in reference 9, the utility of calculations based on these models was severely restricted due both to the fact that the original solution techniques employed bordered on what was barely possible at the time, as well as that considerable hand computation and intervention was required. Moreover, reported results were carried out for only a limited set of solar-wind conditions such as obstacle shape, oncoming Mach number, interplanetary magnetic field, etc., and were presented in archival publications only in the form of plots from which results for other conditions had to be determined by interpolation. The importance of the preliminary work of reference 9 was that advanced computational methods, based on current state-of-the-art algorithms, were introduced to this problem to provide the basic gas-dynamic solutions. The frozen-in magnetic-field was then solved for on the high-resolution flow-field grid, and the entire computational procedure was assembled into a user-oriented program providing the detailed flow-field and magnetic-field properties in a convenient output format.

In the current work reported here, those basic procedures have been extended and generalized in several important directions. These include the capability for treating very low oncoming interplanetary gasdynamic Mach numbers ($M_\infty \simeq 2.0$), as well as quite general ionopause shapes. A new family of ionopause shapes has been developed which accounts for the effect of gravitational variation in scale height. Additionally, the capability for determining the plasma gasdynamic and magnetic-field properties along an arbitrary spacecraft trajectory, simultaneously accounting for an arbitrary oncoming direction of the solar wind, has been developed. Moreover, a large number of sample calculations have been performed for typical solar-wind conditions and, using the output contour-plot capability, a catalog of these cases were established and are archived here for convenient quick-look use. Finally, a number of successful comparisons were made by the present computational model with actual spacecraft observations obtained from initial orbits of the Pioneer-Venus Orbiter. These comparisons have both provided a verification of the basic theoretical model as well as demonstrated its value as a convenient research tool capable of routinely providing details of the solar-wind/planetary atmosphere interaction process not previously attainable--at modest computational cost and in a format directly compatible with observational data.

LIST OF SYMBOLS

a	speed of sound, $(\gamma p/\rho)^{1/2}$
A	Alfvén speed, $(B^2/4\pi\rho)^{1/2}$
$\bar{A}$	Jacobian matrix associated with IMP code, equal to $\partial\hat{E}/\partial\hat{U}$
$\underset{\sim}{B}$	magnetic field vector
$\bar{B}$	Jacobian matrix associated with IMP code, equal to $\partial\hat{F}/\partial\hat{U}$
C_p	specific heat at constant pressure
C_v	specific heat at constant volume
D	distance defined by eq. (59)
e	internal energy, eq. (3)
e_t	total energy, eq. (44)
E	column matrix defined by eq. (42)
$\hat{E}$	column matrix associated with IMP code, equal to $(\xi_T U + \xi_X E + \xi_R F)/J$
F	column matrix defined by eq. (42)
$\hat{F}$	column matrix associated with IMP code, equal to $(\eta_T U + \eta_X E + \eta_R F)/J$
g	acceleration due to gravity
g_k	gravitational component, eq. (5)
G	column matrix defined by eq. (42)
h	enthalpy, eq. (47)
h_t	total enthalpy, eq. (47)
H	local scale height of atmosphere, $\bar{R}T/\bar{M}g$
$\bar{H}$	local scale height with gravitational variation, $H(R_R/R_S)^2$
J	Jacobian matrix, eq. (43)
K	constant defined by eq. (34)
$\Delta\underset{\sim}{\ell}$	vector length of elemental magnetic flux tube
M	local Mach number, $\lvert\underset{\sim}{v}\rvert/a$
$\bar{M}$	nondimensional mean molecular mass, equal to 1/2 for ionized atomic hydrogen

M_A	local Alfven Mach number, $\lvert \underset{\sim}{v} \rvert / A$
p	pressure
q	shock velocity
Q	dummy parameter
r	spherical radial distance
R	cylindrical radial distance
$\bar{R}$	gas constant, 8.315×10^7 ergs/gm°K
R_i	spherical radius of ionopause, eq. (39)
R_o	spherical distance from center of planet to ionopause nose
S_k	Poynting vector component
ΔS	incremental distance along streamline
t, T	time
(u,v,w)	velocity components associated with the (X,Y,Z) coordinate directions, respectively
U	column matrix defined by eq. (42)
$\hat{U}$	column matrix associated with IMP code, equal to U/J
$\underset{\sim}{v}$	velocity vector
(x,y,z) or (x_w,y_w,z_w)	solar-wind oriented Cartesian coordinates with origin at planetary center, x positive upstream and z positive northward
(x_s,y_s,z_s)	sun-planet oriented Cartesian coordinates with origin at planetary center, x_s positive toward sun, y_s positive opposite to planetary orbital motion, and z_s positive northward
(x',y',z')	solar-wind oriented Cartesian coordinates defined by an azimuthal rotation given by eq. (70)
(X,Y,Z)	solar-wind oriented Cartesian coordinates with origin at planetary center, X positive downstream and Z positive northward
α_p	interplanetary magnetic-field angle between perpendicular and parallel components, eq. (62)
α_n	interplanetary magnetic-field angle between normal and in-plane components, eq. (63)

β	spherical polar angle, measured with origin at planet center, from subsolar point away from undisturbed solar wind direction; varies from 0 in upstream direction to π in downstream direction; eq. (39)
γ	ratio of plasma specific heats
δ	angle defined by eq. (59)
δ_{ik}	Kronecker delta
δ_S	local angle of bow shock wave
$(\delta_\xi, \delta_\eta)$	second-order difference operators in (,) direction
ε	smoothing coefficient in IMP code
η	transformation variable, eqs. (40), (48)
θ	azimuthal rotation angle in solar-wind (X,Y,Z) system, eq. (69); also shock tangency angle, eq. (59)
Λ	quantity defined by eq. (36)
ξ	transformation variable, eqs. (40), (48)
ρ	density
σ	conductivity
τ	transformed time, eq. (40)
Φ	gravitational potential, eq. (5)
ϕ_p	solar-wind polar angle
ψ	angle between outward normal to magneto/ionosphere boundary and oncoming undisturbed solar wind, eq. (32); also, angle of magnetic component $(\underset{\sim}{B}/B_\infty)_\perp$, eq. (58)

Subscripts

b	obstacle body
i	ionopause
n	normal direction
P	arbitrary point
R	reference quantity
s	planetary surface; also streamline
S	shock surface

LIST OF SYMBOLS (Concluded)

s_t	stagnation conditions
t	tangential direction
o	reference quantity at subsolar point
1	conditions upstream of a discontinuity
2	conditions downstream of a discontinuity
∞	interplanetary undisturbed quantity
$(\parallel, \perp, n)$	parallel, perpendicular, and normal magnetic-field components as defined in eq. (56)

Superscripts

^	unit vector
*	relative to shock

ANALYSIS

The Mathematical Model - Formulation of the Fluid Representation

The fundamental assumption underlying the present work and that reported in all of the references cited above is that the average bulk properties of solar-wind flow around a planetary magneto/ionosphere can be adequately described by the continuum equations of magnetohydrodynamics for a single-component perfect gas having infinite electrical conductivity and zero viscosity and thermal conductivity. Theoretical justification of this point has not yet been established, and proof remains essentially qualitative at present. The primary justification for use of the continuum fluid model is the outstanding agreement of the qualitative results predicted on this basis with those actually measured in space. It appears that the continuum model is capable of accounting both for many of the details as well as the broad features of the observations.

Governing equations.- The equations which express the conservation of the average bulk mass, momentum, energy, and magnetic field of the solar-wind plasma are given by the following expressions:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_k} (\rho v_k) = 0 \tag{1}$$

$$\frac{\partial}{\partial t} (\rho v_i) + \frac{\partial}{\partial x_k} \left(\rho v_i v_k + p\delta_{ik} - \frac{B_i B_k}{4\pi} + \frac{B^2}{8\pi} \delta_{ik} + \frac{g_i g_k}{4\pi G} - \frac{g^2}{8\pi G} \delta_{ik} \right) = 0 \tag{2}$$

$$\frac{\partial}{\partial t} \left(\frac{\rho v^2}{2} + \rho e + \rho\Phi + \frac{B^2}{8\pi} \right) + \frac{\partial}{\partial x_k} \left[\rho v_k \left(\frac{v^2}{2} + e + \frac{p}{\rho} + \Phi \right) + S_k \right] = 0 \tag{3}$$

$$\frac{\partial B_i}{\partial t} = \frac{\partial}{\partial x_k} (v_i B_k - v_k B_i) \; , \; \frac{\partial B_i}{\partial x_i} = 0 \tag{4}$$

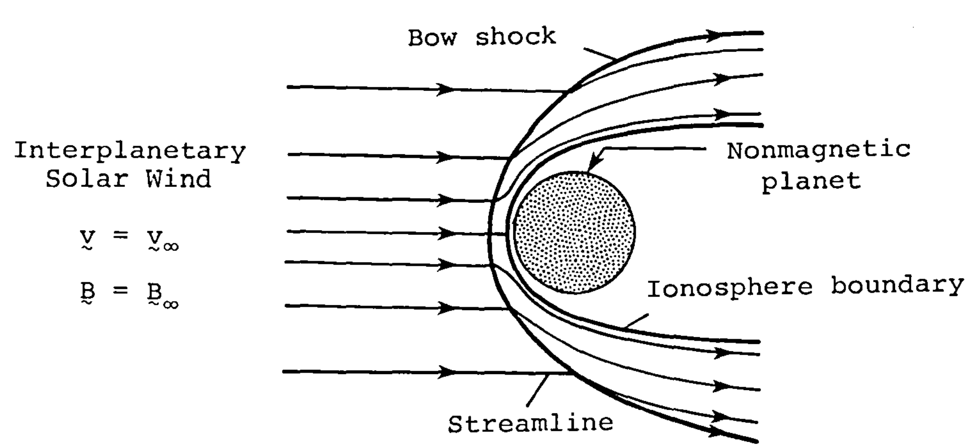
where

$$g_i = -\frac{\partial \Phi}{\partial x_i} \quad , \quad S_k = \frac{1}{4\pi}\left(v_k B^2 - B_k v_i B_i\right) \tag{5}$$

and the equation of state of a perfect gas is given by

$$p = \frac{\rho \bar{R} T}{\bar{M}} \tag{6}$$

In these equations and those to follow, the symbols ρ, p, $\underset{\sim}{v}$, T, $e = C_v T$, and $h = C_p T$ refer to the density, pressure, velocity, temperature, internal energy and enthalpy, and C_v and C_p refer to the specific heats at constant volume and pressure. We define the symbol $\bar{R} = (C_p - C_v)\ \bar{M} = 8.31 \times 10^7$ ergs/gm°K as the universal gas constant, and $\bar{M}$ as the mean molecular weight nondimensionalized so that $\bar{M} = 16$ for atomic oxygen. For fully ionized hydrogen, $\bar{M}$ is thus 1/2. The magnetic field $\underset{\sim}{B}$ and the Poynting vector $\underset{\sim}{S}$ for the flux of electromagnetic energy are expressed in terms of gaussian units. The gravitational potential Φ and acceleration $\underset{\sim}{g}$ are assumed to be due to massive fixed bodies so that their time derivatives are zero. These equations apply in the region exterior to the ionosphere boundary, as shown in the sketch below, and also in a degenerate sense in the ionosphere.



Conditions at discontinuities.- Because of the omission of dissipative terms in these equations, surfaces of discontinuity may develop in the solution, across which the fluid and magnetic properties change abruptly, but in such a way that mass, momentum, magnetic flux, and energy are conserved. These are approximations to comparatively thin surfaces across which similar but continuous changes in the fluid and magnetic properties occur in the corresponding theory of a dissipative gas, and correspond physically to the bow wave, ionosphere boundary, and possibly other thin regions of rapidly changing properties. Across these surfaces, continuous solutions of the dissipationless differential equations cease to exist. The flow is no longer governed solely by the differential equations (1) to (4), but must be supplemented by additional considerations. The conservation of mass, momentum, magnetic flux, and energy lead to the following conditions which relate quantities on the two sides of any such discontinuity:

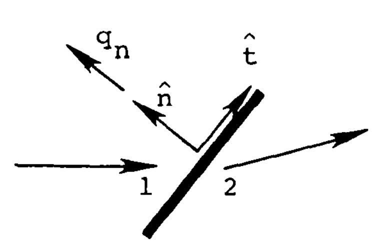
$$\left[\rho v_n^*\right] = 0 \tag{7}$$

$$\left[\rho \underset{\sim}{v} \cdot v_n^* + (p + B^2/8\pi)\hat{n} - B_n \underset{\sim}{B}_t/4\pi\right] = 0 \tag{8}$$

$$\left[\underset{\sim}{B}_t \cdot v_n^* - B_n \cdot \underset{\sim}{v}_t\right] = 0 \tag{9}$$

$$\left[v_n^* \left(\frac{1}{2}\rho v^2 + \rho e + p + \frac{B^2}{4\pi}\right) + q_n \cdot \left(p + \frac{B^2}{8\pi}\right) - \frac{B_n(\underset{\sim}{v} \cdot \underset{\sim}{B})}{4\pi}\right] = 0 \tag{10}$$

Here, $(\hat{n},\hat{t})$ denote unit vectors normal and tangential to the discontinuity surface, as sketched below,



where q_n represents the local normal velocity of the discontinuity surface, and $v_n^* = v_n - q_n$ is the fluid normal velocity component relative to the normal velocity q_n of the discontinuity surface. The square brackets are used to indicate the difference between the enclosed quantities on the two sides of the discontinuity, as in $[Q] = Q_2 - Q_1$ where subscripts 1 and 2 refer to conditions on the upstream and downstream sides, respectively, of the discontinuity.

Five classes of discontinuities are described by Eqs. (7-10). Those with $v_n^* = 0$ are called tangential discontinuities or contact discontinuities according to whether or not B_n vanishes. Discontinuities across which there is flow ($v_n^* \neq 0$) are divided into three categories called rotational discontinuities, and fast and slow shock waves. Some properties which distinguish the various discontinuities are indicated by the following relationships:

Tangential:

$$v_n^* = B_n = 0, \ [\underset{\sim}{v}_t] \neq 0, \ [\underset{\sim}{B}_t] \neq 0, \ [\rho] \neq 0, \ [p + B^2/8\pi] = 0 \tag{11}$$

Contact:

$$v_n^* = 0, \ B_n \neq 0, \ [\underset{\sim}{v}] = [\underset{\sim}{B}] = [p] = 0, \ [\rho] \neq 0 \tag{12}$$

Rotational:

$$\begin{gathered} v_n^* = \pm B_n/\sqrt{4\pi\rho}\ , \ [\underset{\sim}{v}_t] = \pm[\underset{\sim}{B}_t]/\sqrt{4\pi\rho} \\ [\rho] = [p] = [v_n] = [v^2] = [B^2] = [B_n] = 0 \end{gathered} \tag{13}$$

Fast and Slow Shock Waves:

$$v_n^* \neq 0, \ [\rho] > 0, \ [p] > 0, \ [B_n] = 0$$

$$\left(\rho v_n^*\right)_{fast} \geq \left(\rho v_n^*\right)_{rot.} \geq \left(\rho v_n^*\right)_{slow} \tag{14}$$

$$\underset{\sim}{B}_t \text{ and } B^2 \begin{Bmatrix}\text{increase}\\ \text{decrease}\end{Bmatrix} \text{ through } \begin{Bmatrix}\text{fast}\\ \text{slow}\end{Bmatrix} \text{ shock waves}$$

Of the five classes of discontinuities possible, two of these, the fast shock wave and the tangential discontinuity, are of concern in the present applications. The first relates conditions on the two sides of the bow shock wave, and any other shock waves present, while the latter has properties required to describe a boundary surface (ionopause) that separates the flowing solar wind and the planetary ionosphere. More detailed consideration of the tangential discontinuity condition leads to a determination of the ionopause shape, as described in the following sections.

With regard to conditions at the bow wave, for solar-wind flows past Venus, as well as Mars and the Earth, that discontinuity can only be represented by a fast shock wave since the mass flux through each of the other possible choices is too small. With regard to conditions at the ionopause, of the various possibilities, only the tangential discontinuity has properties compatible with those required to describe a boundary surface that separates the externally flowing solar wind and the planetary atmosphere; that is, the condition $v_n^* = 0$ prohibits flow across the boundary, while the condition $B_n = 0$ must hold since by assumption no magnetic field exists interior to the ionopause and the solenoidal jump condition $[B_n] = 0$ always holds.

Frozen-field approximation.- Two important parameters characterize the solar-wind flow at any field point as described by eqs. (1-5). These are the Mach number $M = v/a$ and the Alfvén Mach number $M_A = v/A$. The former is the ratio of the flow velocity to the speed of sound $a = (\gamma p/\rho)^{\frac{1}{2}}$, while the latter is the ratio of the flow velocity to the speed A =

$(B^2/4\pi\rho)^{\frac{1}{2}}$ of a rotational or Alfvén wave propagating along the direction of the magnetic field.

For typical solar-wind conditions (refs. 5,6), both the oncoming Mach number and the Alfvén Mach number are high ($M_\infty \simeq M_A \simeq 0(10)$). In this instance, an important simplification of the magnetohydrodynamic equations occurs. This is so because the order of magnitude of the inertia term in differential equation (2) for the momentum is related to that of the magnetic terms by the square of the Alfvén Mach number. When the latter is large, therefore, the magnetic terms in eqs. (2),(3),(8), and (10) decouple from the gasdynamic portions of those equations. Furthermore, for Earth, Venus, or Mars, the strong interactive nature of the flow permits the terms involving $\underset{\sim}{g}$ and Φ to be disregarded because of the relative smallness of their effect on the fluid motion (ref. 5). The equations for the fluid motion thereby reduce to those of gasdynamics, while the magnetic field $\underset{\sim}{B}$ can be determined subsequently by solving the remaining equations using the values for $\underset{\sim}{v}$ already determined. The magnetic field, determined in this fashion, is usually interpreted as being "frozen-in" or moving with the fluid (ref. 5).

This then results in the following differential and conservation equations; for the flow field

$$\frac{\partial\rho}{\partial t} + \frac{\partial}{\partial x_k}(\rho v_k) = 0 \qquad (15)$$

$$\frac{\partial}{\partial t}(\rho v_i) + \frac{\partial}{\partial x_k}(\rho v_i v_k + p\delta_{ik}) = 0 \qquad (16)$$

$$\frac{\partial}{\partial t}\left(\frac{\rho v^2}{2} + \rho e\right) + \frac{\partial}{\partial x_k}\left[\rho v_k\left(\frac{v^2}{2} + e + p/\rho\right)\right] = 0 \qquad (17)$$

$$\left[\rho v_n^*\right] = 0 \qquad (18)$$

$$\left[\rho \underset{\sim}{v} \cdot v_n^* + p\right] = 0 \qquad (19)$$

$$\left[v_n^* \cdot \left(\frac{1}{2} \rho v^2 + \rho e + p \right) \right] = 0 \tag{20}$$

and for the magnetic field

$$\frac{\partial B_i}{\partial t} + \frac{\partial}{\partial x_k} (v_k B_i - v_i B_k) = 0 \tag{21}$$

$$\frac{\partial B_i}{\partial x_i} = 0 \tag{22}$$

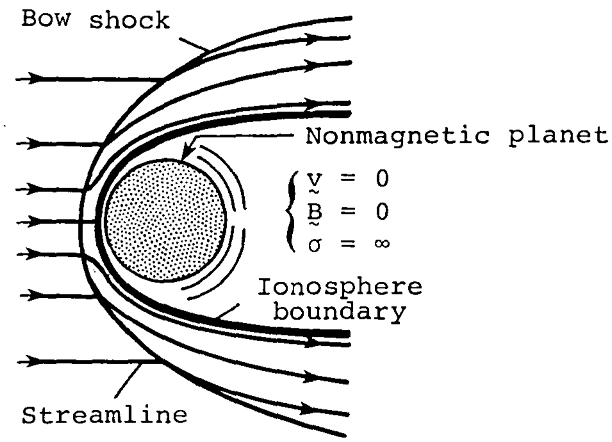
$$\left[B_n \right] = 0 \tag{23}$$

$$\left[B_n \cdot \underset{\sim}{v}_t - \underset{\sim}{B}_t \cdot v_n^* \right] = 0 \tag{24}$$

Equations (15) to (24) provide the governing equations which form the basis of the mathematical representation of the solar wind-magneto/ionosphere interaction problem considered here. For all of the results as well as the computer codes presented herein, we are interested exclusively in the steady-state solution to these equations which are obtained by setting $\partial/\partial t = 0$ and $v_n^* = v_n$, i.e. $q_n = 0$. We have presented the unsteady equations, however, since one of the computational methods used to determine the gasdynamic solution employs an unsteady procedure, integrating in time until the steady-state solution is asymptotically obtained.

Determination of the Ionosphere Boundary

The determination of the ionosphere boundary initiates from the assumptions that the ionosphere, or at least the outer part of it that participates in the interaction with the solar wind, is idealized as a spherically-symmetric and hydrostatically-supported plasma having infinite electrical conductivity, effectively bound to the planet and incapable of mixing with the solar wind, as indicated in the sketch below:



This interior plasma is separated from the flowing solar plasma by a tangential discontinuity across which the relations

$$v_n = B_n = [p + B^2/8\pi] = 0$$
$$[\underset{\sim}{v}_t] \neq 0; \quad [\underset{\sim}{B}_t] \neq 0; \quad [\rho] \neq 0 \tag{25}$$

given previously (eq. (11)) must hold. The basis for important simplifying approximations to these conditions, which can be assumed to apply at the Venusian ionosphere boundary and possibly for that at Mars as well, is that the gas pressure p is much larger than the magnetic pressure $B^2/8\pi$ on both sides of the ionopause. Therefore, the discontinuity pressure balance relation $[p + B^2/8\pi] = 0$ of eq. (25) reduces to a simple equality between the ionosphere pressure and the static pressure of the flowing solar plasma adjacent to the ionopause, i.e.

$$(p)_{atm.} = (p)_{flow} \tag{26}$$

Determination of the ionospheric pressure in the vicinity of the ionopause for the ionosphere models chosen in this study proceeds from the assumption of hydrostatic support, which implies a quiescent ionosphere where the bulk motions of the gas with respect to the planet are sufficiently small ($\underset{\sim}{v} = 0$) that equilibrium exists between the pressure gradient and gravity, viz.

$$dp/dr = -\rho g \tag{27}$$

where p and ρ are the gas pressure and density, r is the radial distance measured from the center of the planet, and g is the acceleration due to gravity. The variation of g is inversely proportional to r_s, so that $g = g_s(r_s/r)^2$ where the subscript s denotes values at the surface of the planet. Since the density ρ is related to the pressure according to the perfect gas law eq. (6), eq. (27) can be integrated to yield

$$p = p_R \exp\left(-\int_{R_R}^{r} \frac{dr}{H}\right) \tag{28}$$

where p_R is the pressure at some reference radius R_R and H is the local scale height of the atmosphere given by $H = \bar{R}T/\bar{M}g$.

If H is regarded as constant; that is, if variations of g and T with r are neglected, eq. (27) can be integrated directly to yield

$$p = p_R \exp\left(-\frac{r - R_R}{H}\right) \tag{29}$$

In view of uncertainties associated with measurements of the atmospheric properties of Venus and Mars, the variation of p with r as given by eq. (29) was adopted in the initial solar wind/ionosphere applications (ref. 6) and was also used in the previous study (ref. 9) involving the initial application of advanced computational methods to this problem. With preliminary ionospheric data now available from the Pioneer-Venus spacecraft (refs. 10 and 11), some of these uncertainties for Venus have been removed. It has been found that the assumption of an isothermal (T = constant) atmosphere at typical ionopause heights is quite reasonable. Consequently, there is no need to neglect the variation of gravity in the scale height in eq. (28). Including this effect leads to the following result for the pressure

$$p = p_R \exp\left[-\frac{R_R \cdot (r - R_R)}{\bar{H} \cdot r}\right] \tag{30}$$

where

$$\bar{H} = H_s \cdot (R_R/R_s)^2 \tag{31}$$

and R_s is the planetary radius and $H_s = \overline{R}T/\overline{M}g_s$. Equations (29) and (30) provide the two models employed in this study for the ionosphere pressure variation which is required in eq. (26) for the pressure balance condition at the ionopause.

For the a priori determination of the static pressure of the flowing solar-wind plasma on the exterior boundary of the ionosphere - $(p)_{flow}$ in eq. (26) - we use, as in all previous applications, the Newtonian approximation

$$p = p_{st} \cos^2 \psi \tag{32}$$

where ψ is the angle between the outward normal to the magnetosphere boundary and the flow direction of the oncoming undisturbed solar wind, and p_{st} is the stagnation or ram pressure exerted on the nose of the ionopause and is given by

$$p_{st} = K\rho_\infty v_\infty^2 \tag{33}$$

In this relation, K is a constant usually taken as one, but whose actual value is

$$K = \frac{1}{\gamma}\left[\frac{\left[(\gamma + 1)/2\right]^{(\gamma + 1)}}{\gamma - (\gamma - 1)/2M_\infty^2}\right]^{\frac{1}{\gamma - 1}} \tag{34}$$

For the high Mach number flows typical of solar-wind conditions, K approaches 0.844 for $\gamma = 2$ and 0.881 for $\gamma = 5/3$. Modification of the product $K\rho_\infty$ in eq. (33) to account for the presence of minor constituents such as ionized helium in the solar wind, as well as a discussion of the differences in that product between a fluid and collisionless representative, is provided in reference 8. The important implication associated with the introduction of the Newtonian approximation is that the calculation of the shape of the ionosphere boundary decouples from the calculation of the external flow. We then arrive at the following equation for the pressure balance at the ionopause locations R_i:

$$K\rho_\infty v_\infty^2 \cos^2 \psi = p_R \, \Lambda(R_i) \tag{35}$$

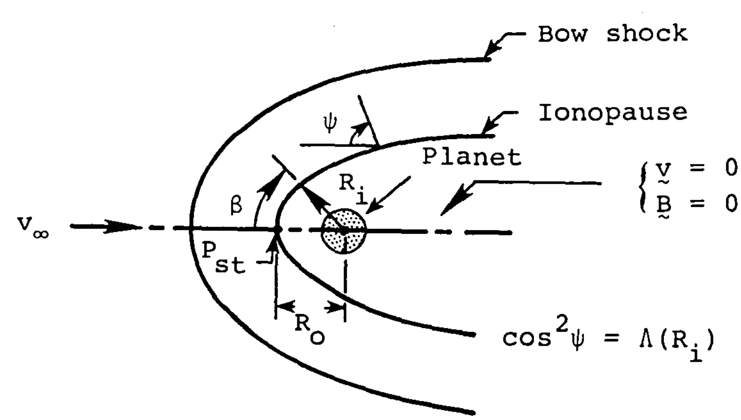
where

$$\Lambda(R_i) = \begin{cases} \exp\left[-\left(\dfrac{R_i - R_R}{H}\right)\right] & g, T = \text{Const.} \quad (36a) \\ \exp\left[-R_R \left(\dfrac{R_i - R_R}{H \cdot R_i}\right)\right] & g = g_s\left(\dfrac{r_s}{r}\right)^2, T = \text{Const.} \quad (36b) \end{cases}$$

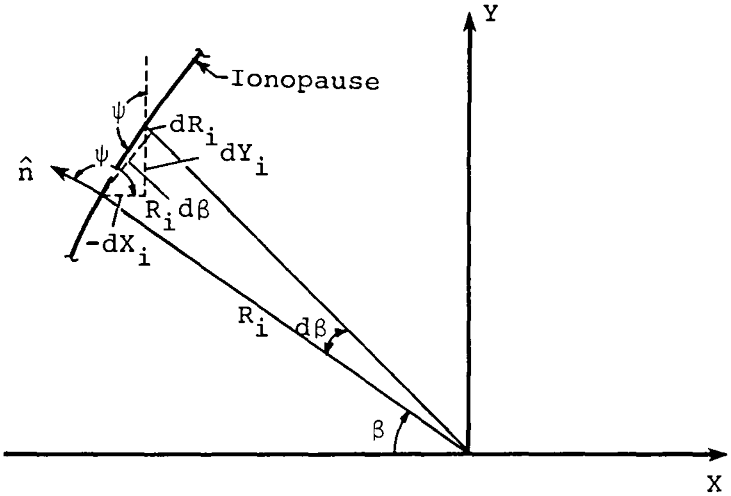
depending upon whether the gravitational variation is included in scale height or not. It is convenient to choose as the reference radius and location the stagnation point on the ionopause; that is, $R_R = R_o$ where R_o is the distance from the center of the planet to the nose of the ionopause. This implies that $p_R = p_o = K\rho_\infty v_\infty^2$ and that at all points along the ionosphere boundary

$$\cos^2 \psi = \Lambda(R_i) \tag{37}$$

The final mathematical statement of the free-boundary problem for determining the shape of the ionosphere boundary then is summarized in the sketch below:



In order to proceed to a final determination of the ionopause shape, it is necessary to relate the local angle ψ to the local coordinates (R_i, β) of the boundary. This is accomplished with the help of the following sketch



from which we find

$$\cos^2 \psi = \left(\frac{dY_i}{dS}\right)^2 = \frac{(R_i d\beta \cos \beta + dR_i \sin \beta)^2}{dR_i^2 + (R_i d\beta)^2} \tag{38}$$

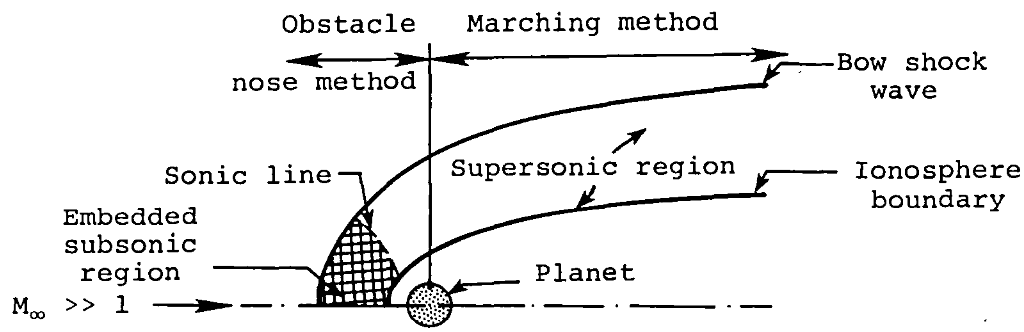
This results in the following ordinary differential equation for the ordinates of the ionosphere boundary

$$\frac{dR_i}{d\beta} = R_i \left[\frac{\sin 2\beta - 2\sqrt{\Lambda - \Lambda^2}}{2(\Lambda - \sin^2 \beta)} \right] \quad 0 \leq \beta \leq \pi \tag{39}$$

where Λ is defined by eqs. (36a,b) and β is the angle measured from the subsolar point as indicated above. Results for various ionopause shapes obtained by integrating eq. (39) for different values of H/R_o using the constant scale-height model eq. (36a) were provided in ref. 9. Similar results using the isothermal model, eq. (36b) for different values of $\overline{H}/R_o$ in the range $0.01 \leq \overline{H}/R_o \leq 0.5$ are provided in figure 1, where for comparison purposes the constant scale-height shapes for corresponding H/R_o values are also illustrated. We note that the range of interest for planetary applications to Venus and Mars appears to be $0.01 \lesssim \overline{H}/R_o \lesssim 0.10$. Tabulated ordinates of Y_i/R_o vs. X/R_o are provided in Table 1 for $\overline{H}/R_o = 0.01, 0.05, 0.10, 0.20$, and 0.25, where $Y_i = R_i \sin \beta$ is the cylindrical radial coordinate of the ionopause profile.

Calculation of the Gasdynamic Flow Properties

Determination of the gasdynamic flow properties is, both conceptually and computationally, the most difficult and time-consuming portion of the total calculation of the solar-wind/terrestrial-planet interaction; and represents the heart of the present modeling effort insofar as the application of advanced computational procedures is concerned. The calculation consists of determining solutions to the differential equations and discontinuity conservation equations given by eqs. (15-20). Since in solar-wind/terrestrial-planet interactions, both the downstream tail region (far field) as well as the region in the vicinity of the obstacle nose (near field) are generally of interest, the computational methods selected must be capable of efficiently determining this entire flow field. In view of the need to carry the flow calculation to an arbitrary downstream distance, the most computationally-expedient procedure is to subdivide the flow field into two regions, as indicated in the sketch below:



This sketch illustrates the essential features of the high-supersonic Mach number flow typical of solar-wind flows past terrestrial planets. Of particular note is the embedded subsonic pocket located at the nose of the ionopause. The presence of this subsonic pocket necessitates use of a computational method capable of treating mixed subsonic/supersonic flows. Downstream of this region, the flow becomes supersonic and remains so for the convex shapes typical of solar-wind/ionosphere boundaries. In that region, a more computationally-economical procedure than that required near the nose can be employed. Such a subdivision of both flow field and solution procedures is common practice for calculating such flows and was employed in the previous solar-wind applications as well as in a related application to space shuttle reentry flows (ref. 12). The precise surface on which the solutions are joined is relatively arbitrary; in our procedure it was convenient to place it along a plane through the planet center and normal to the free-stream direction of the solar wind, i.e. the dawn-dusk terminator. As illustrated in fig. 2, this position is further downstream than used in the former work in which an inverse iteration method was used for the nose region and the method of characteristics was used for the remaining supersonic region. In light of recent advances, both of the techniques used in the former procedures, particularly the inverse method, are now considered obsolete and much inferior to more current methods. In the new code, those two methods have been superceded by: (1) a new axisymmetric implicit unsteady Euler-equation solver (IMP) specifically developed for the present application, which determines the steady-state solution in the nose region by an asymptotic time-marching procedure, and (2) a shock-capturing marching solution (SCT) which spatially advances the solution downstream as far as required by solving the steady Euler equations.

Nose region solution - implicit unsteady Euler equation method.- The partial differential equations employed in the implicit (IMP) code are the unsteady gasdynamic Euler eqs. (15-20) for axisymmetric flow. These equations may be written in conservation-law form under the generalized independent variable transformation

$$\tau = T, \quad \xi = \xi(T,X,R), \quad \eta = \eta(T,X,R) \tag{40}$$

as follows

$$(U/J)_\tau + \left[(\xi_T U + \xi_X E + \xi_R F)/J\right]_\xi + \left[(\eta_T U + \eta_X E + \eta_R F)/J\right]_\eta + G = 0 \tag{41}$$

where

$$U = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho e_t \end{bmatrix} \qquad E = \begin{bmatrix} \rho u \\ p + \rho u^2 \\ \rho u v \\ (\rho e_t + p)u \end{bmatrix}$$

$$F = \begin{bmatrix} \rho v \\ \rho u v \\ p + \rho v^2 \\ (\rho e_t + p)v \end{bmatrix} \qquad G = \frac{1}{RJ}\begin{bmatrix} \rho v \\ \rho u v \\ \rho v^2 \\ (\rho e_t + p)v \end{bmatrix} \tag{42}$$

and the Jacobian

$$J = \xi_X \eta_R - \xi_R \eta_X \tag{43}$$

In eqs. (40) through (43), T denotes time, X is the axial downstream coordinate, and R the cylindrical radial distance; u and v the velocity components in the X and R directions; and e_t is the total energy per unit mass, which for a perfect gas is related to the other quantities by

$$e_t = p/[\rho(\gamma - 1)] + (u^2 + v^2)/2 \tag{44}$$

The subscripts in eqs. (41) and (43) denote partial derivatives with respect to the indicated variable.

The analysis commences by introducing a computational mesh in polar (r,β) coordinates such that one family of coordinates consists of rays from the planetary center spaced at equal increments of β measured from the obstacle nose, and the other of curved lines intersecting each ray so as to divide the portion of it between the ionopause and the shock wave into a fixed number of equal segments. The coordinate transformation eq. (40) is then used to map the portion of the X,R,T physical space bounded by (1) the bow wave, (2) the downstream outflow boundary at $\beta = \pi/2$, (3) the obstacle surface, and (4) the stagnation streamline at $\beta = 0$ into a rectangle in the ξ,η,τ computational space as illustrated in fig. 3. Generally, the transformation metrics at each time step are not known beforehand, and must be determined numerically as part of the solution. Integration step size is established by using the eigenvalues of the Jacobian matrices $\overline{A}$ and $\overline{B}$, where $\overline{A} = \partial\hat{E}/\partial\hat{U}$, $\overline{B} = \partial\hat{F}/\partial\hat{U}$, and $\hat{U} = U/J$, $\hat{E} = (\xi_T U + \xi_X E + \xi_R F)/J$, and $\hat{F} = (\eta_T U + \eta_X E + \eta_R F)/J$.

Boundary conditions necessary for the specification of a properly-posed mathematical problem are that the flow (a) satisfy the axisymmetric Rankine-Hugoniot shock relations derivable from eq. (41) along (1), (b) be entirely supersonic along (2), (c) be parallel to boundaries (3) and (4), and (d) be symmetric about boundary (4). Initial flow-field conditions are determined by use of an approximating formula for the coordinates of the bow shock wave which is dependent on γ, M_∞ and the shape of the obstacle, and by prescribing a Newtonian pressure distribution on the obstacle. Since the maximum entropy streamline wets the obstacle surface, that fact plus the known flow direction on the obstacle serve to determine the remainder of the flow properties on that surface. A linear variation for the flow properties between the bow shock and the obstacle is then prescribed. This provides the initial flow field which is then integrated in a time-asymptotic fashion until the steady-state solution is obtained.

The basic numerical algorithm used in the IMP code was developed by Beam and Warming (ref. 13) and is second-order accurate, noniterative, and spatially factored. In particular, the "delta form" with Euler time differencing is employed. When applied to eq. (41), the algorithm assumes the form

$$(I + \Delta\tau\delta_\xi\bar{A}^n)(I + \Delta\tau\delta_\eta\bar{B}^n)(\hat{U}^{n+1} - \hat{U}^n) = -\Delta\tau(\delta_\xi\hat{E}^n + \delta_\eta\hat{F}^n + G) \tag{45}$$

where $\bar{A}$ and $\bar{B}$ are the Jacobian matrices, I is the identity matrix, δ_ξ and δ_η are second-order, central-difference operators, $\hat{U}^{n+1} = \hat{U}(n\Delta\tau)$ and $\Delta\tau$ is the integration step size.

Equation (45) is solved at the interior points only. It requires two 4x4 block tridiagonal inversions at each time step of the integration. The solution proceeds as follows:

1. Define $\Delta\hat{U} = \hat{U}^{n+1} - \hat{U}^n$

2. Form the right-hand side of eq. (45) and store results in the $\hat{U}^{n+1}$ array.

3. Apply smoothing $\hat{U}^{n+1} = \hat{U}^{n+1} - (\varepsilon/8)S/J$.

4. Define $\bar{U} = (I + \Delta\tau\delta_\eta\bar{B}^n)\Delta\hat{U}$ and solve the matrix equation $(I + \Delta\tau\delta_\xi\bar{A}^n)\bar{U} = \hat{U}^{n+1}$ for $\bar{U}$ storing the result in the $\hat{U}^{n+1}$ array.

5. Solve the matrix equation $(I + \Delta\tau\delta_\eta\bar{B}^n)\Delta\hat{U} = \hat{U}^{n+1}$ for $\Delta\hat{U}$.

6. Obtain the values of $\hat{U}^{n+1}$ from the relation $\hat{U}^{n+1} = \Delta\hat{U} + \hat{U}^n$.

7. Transfer contents of $\hat{U}^{n+1}$ to $\hat{U}^{n}$ and repeat all steps until satisfactory convergence is attained.

In step 3 a fourth-order smoothing term S is used to eliminate nonlinear instabilities that may arise since the use of central differences in the spatial directions results in a neutrally stable algorithm. This smoothing term is given by

$$S_{jk} = (\hat{U}J)^{n+1}_{j+2,k} - 4\left[(\hat{U}J)^{n+1}_{j+1,k} + (\hat{U}J)^{n+1}_{j-1,k}\right] + 12(\hat{U}J)^{n+1}_{j,k} + (\hat{U}J)^{n+1}_{j-2,k}$$

$$+ (\hat{U}J)^{n+1}_{j,k+2} - 4\left[(\hat{U}J)^{n+1}_{j,k+1} + (\hat{U}J)^{n+1}_{j,k-1}\right] + (\hat{U}J)^{n+1}_{j,k-2} \qquad (46)$$

and ε, the smoothing coefficient, chosen from the range $0 \leq \varepsilon \leq 0.4$ depending upon the size of the time step. The j and k indices correspond to the ξ and η directions, respectively. At the points adjacent to the boundaries a special form of the smoothing term is used.

At the boundaries, modification of the differencing algorithm to account for the particular conditions described above is accomplished as follows. The obstacle-surface flow-tangency condition is incorporated through the use of Kentzer's scheme (ref. 14), while at the symmetry plane, the variables are reflected according to whether they are odd or even. At the outflow boundary where the flow is entirely supersonic, the dependent variables are determined by extrapolation from the adjacent interior points. For the upstream boundary formed by the bow shock wave, the sharp discontinuity approach of reference 15 is used. The interior flow field bounded by these various boundaries is treated in shock-capturing fashion and, therefore, allows for the correct formation of secondary internal shocks.

In the initial development of the nose-region solution procedure (ref. 9), it was found that for certain ionopause obstacle shapes which have a significant amount of lateral flaring at the dawn-dusk terminator, for example, constant scale-height shapes for $H/R_o \geq 0.5$, and/or cases involving low free-stream Mach numbers $M_\infty \leq 3$, the axial component of velocity at some points on the terminator plane $\beta = \pi/2$ may become subsonic. Although this has no effect whatsoever on the nose-region solver, for these cases the downstream solution cannot be obtained since the marching-region solver which determines the solution downstream of this starting plane, and which is described in detail in the following section, requires supersonic axial velocities in order to proceed. Under the work reported here, this limitation has been removed by developing the capability for adding an additional portion of the flow field, located downstream of the terminator, to the blunt-body solution as illustrated in figure 4. This effectively generalizes the capability of the present procedures to treat a wide variety of ionopause shapes - including all of the shapes of interest described by the constant scale-height and scale-height with gravitational variation atmospheric models found from eqs. (36a,b) - as well as to treat relatively low free-stream Mach numbers, $M_\infty \simeq 2.0$. Details of this capability are provided in the Computer Program Users Manual, Section A.2.1.1 of this report.

Downstream region solution - shock capturing marching method.- Since the shock-capturing technique employed has been described previously in references 16-18, only an outline of the salient features is provided here. The analysis is based on the conservation-law form of the gasdynamic Euler equations for steady axisymmetric flow, which can be readily obtained from eqs. (40) through (44) by setting the τ derivatives to zero. The fourth of this set of equations representing conservation of energy ρe_t can be integrated for steady flow to yield the following relation for the total enthalpy

$$h_t = h + (u^2 + v^2)/2 = \text{constant} \tag{47}$$

where $h = e + p/\rho = C_p T$ is the enthalpy per unit mass.

The computational mesh is defined by lines of constant X and $(R-R_b)/(R_s-R_b)$, where R_s and R_b are functions of X that describe the radial cylindrical coordinates of the ionopause and bow shock wave at the same X as the field point (X,R). The three remaining partial differential equations for conservation of mass and of axial and radial momentum are then transformed to a rectangular computational space by the transformation

$$\xi = X, \quad \eta = \frac{R - R_b}{(R_s - R_b)} \tag{48}$$

to obtain

$$\partial\tilde{E}/\partial\xi + \partial\tilde{F}/\partial\eta + \tilde{G} = 0 \tag{49}$$

$$\tilde{E} = E, \quad \tilde{F} = \left\{F - \left[\frac{\partial}{\partial\xi} R_b + \eta \frac{\partial}{\partial\xi} (R_s - R_b)\right]\right\}/(R_s - R_b)$$

$$\tilde{G} = G + \frac{E}{R_s - R_b} \frac{\partial}{\partial\xi} (R_s - R_b) \tag{50}$$

The finite-difference counterpart of eq. (49) is integrated with respect to the hyperbolic coordinate ξ to yield values of the conservative varible E. Subsequent to each integration step, the physical flow variables p, ρ, u, and v must be decoded from the components e_i of E. This necessitates the solution of four simultaneous, nonlinear equations consisting of eq. (47) together with the three elements e_i. This can be done readily by using the relations $v = e_3/e_1$, $p = e_2 - e_1u$, and $\rho = e_1/u$ together with the expression $h = \gamma/(\gamma-1)(p/\rho)$ for a perfect gas to determine the following quadratic equation for u

$$\frac{u^2}{2} + \frac{\gamma}{\gamma - 1} \left(\frac{e_2 - e_1u}{e_1}\right) u - h_t + \left(\frac{e_3}{e_1}\right)^2$$

$$= - \frac{\gamma + 1}{2(\gamma - 1)} u^2 + \left(\frac{\gamma}{\gamma - 1}\right) \frac{e_2}{e_1} u - h_t + \left(\frac{e_3}{e_1}\right)^2 = 0 \tag{51}$$

Two roots exist; one corresponds to subsonic flow and is discarded since u is always supersonic in the present application, while the other corresponds to supersonic flow and gives the desired solution.

Since only the bow shock wave is treated as a sharp discontinuity and any others that may be present are "captured" by the difference algorithm, selection of the appropriate finite difference scheme to advance the calculation in the ξ direction is of prime importance. Following the analysis of refs. 16-18, the numerical integration of eq. (49) is accomplished using the finite-difference predictor-corrector scheme of MacCormack (ref. 19), the most efficient second-order algorithm for shock-capturing calculations. General descriptions of the method can be found in the references cited.

Calculation of the streamlines.- The streamlines are determined by integrating fluid particle trajectories through the known velocity field since this procedure was found to be more accurate than the alternative mass-flow calculation. The calculation of a particular streamline is initiated at the point where the streamline crosses the bow shock, as illustrated in figure 5. At that point, exact values of the streamline slope dR_S/dX are known in terms of the local shock angle δ_S and free-stream quantities according to the relation

$$\frac{dR_S}{dX} = \frac{(2\cot \delta_S)(M_\infty^2 \sin^2 \delta_S - 1)}{2 + M_\infty^2(\gamma + 1 - 2\sin^2 \delta_S)} \tag{52}$$

which is contained implicitly in both the blunt-body (IMP) and marching (SCT) code solutions. At other points in the flow field, the local streamline slope is given by the ratio of radial to downstream velocity, i.e.,

$$dR_S/dX = v/u \tag{53}$$

and the streamline determination is made by stepwise integration in X using a modified third-order Euler predictor-corrector method. Bivariate linear interpolation from the flow-field grid points is employed to obtained the velocity components (u,v) required at the stepwise points along the streamline trajectory. Separate streamline calculations are made for the nose region (IMP results) and downstream region (SCT results) because of the different coordinate systems employed in those two regions.

Calculation of the Magnetic Field

With the flow properties known from the gasdynamic calculations, determination of the steady magnetic field $\underset{\sim}{B}$ proceeds by integrating the remaining magnetohydrodynamic equations not employed in the gasdynamic analysis, that is eqs. (21-24) with $\partial/\partial t = 0$:

$$\begin{aligned} &\operatorname{curl} (\tilde{B} \times \tilde{v}) = 0, \quad \operatorname{div} \tilde{B} = 0 \\ &[B_n \tilde{v}_t - \tilde{B}_t v_n] = 0, \quad [B_n] = 0 \end{aligned} \tag{54}$$

These equations are commonly interpreted as indicating the field lines move with the fluid. The analysis associated with eqs. (54) leads to a straightforward calculation in which the vector distance from each point on an arbitrarily-selected field line to its corresponding point on an adjacent field line in the downstream direction is determined by numerically integrating $\int \tilde{v}\, dt$ over a fixed time interval Δt. Once the coordinates of the field lines are determined, the magnetic field at any point may be calculated from the relation

$$\frac{\tilde{B}}{|\tilde{B}_\infty|} = \frac{\rho}{\rho_\infty} \frac{\Delta \tilde{\ell}}{|\Delta \tilde{\ell}_\infty|} \tag{55}$$

where $\Delta \tilde{\ell}$ is the vector length of a small element of a flux tube. Figure 6 clarifies these quantities for the plane of magnetic symmetry defined by the plane containing the axis of symmetry of the obstacle and the magnetic-field lines upstream of the bow wave for the special case when the latter is perpendicular to the flow. In that figure the open symbol ○ denotes locations of points on the streamlines corresponding to the fixed-time interval $\Delta t = \Delta S_\infty / v_\infty$.

Such a procedure is valid generally, but its use in the present calculations is confined to only the component of the magnetic field $(\tilde{B})_\perp$ just described. The remainder of the magnetic-field calculation makes use of a decomposition due to Alksne and Webster (ref. 20) in which the axisymmetric properties of the gasdynamic solution and the linearity of the magnetic-field eqs. (54) are employed to derive the following relationship for the magnetic field $\tilde{B}_P$ at any point P:

$$\tilde{B}_P = \left(\frac{\tilde{B}_P}{B_\infty}\right)_{\parallel} B_{\infty_\parallel} + \left(\frac{\tilde{B}_P}{B_\infty}\right)_{\perp} B_{\infty_\perp} + \hat{e}_n \left(\frac{B_P}{B_\infty}\right)_n B_{\infty_n} \tag{56}$$

As illustrated in figure 7, subscripts ", $\perp$, and n refer to contributions associated with the components $B_{\infty_{\parallel}}$ of $\underset{\sim}{B}_\infty$ parallel to $\underset{\sim}{v}_\infty$; the component $B_{\infty_\perp}$ perpendicular to $\underset{\sim}{v}_\infty$ in the plane that contains the point P, the center of the planet, and the vector $\underset{\sim}{v}_\infty$; and the component B_{∞_n} normal to the latter plane, and $\hat{e}_n$ is a unit vector in the latter direction. The unit ratios $(\underset{\sim}{B}_P/B_\infty)_{\parallel}$ and $(B_P/B_\infty)_n$ can be calculated directly from the gas-dynamic solution by the expressions

$$\left(\frac{\underset{\sim}{B}_P}{B_\infty}\right)_{\parallel} = \frac{\rho_P \underset{\sim}{v}_P}{\rho_\infty |\underset{\sim}{v}_\infty|}, \quad \left(\frac{B_P}{B_\infty}\right)_n = \frac{R_P \rho_P}{R_{P_\infty} \rho_\infty} \tag{57}$$

where R_P is the radial cylindrical coordinate of the streamline through P, as indicated in figure 7.

In carrying out the determination of $(\underset{\sim}{B}_P/B_\infty)_\perp$ using eq. (55), values for $\Delta\underset{\sim}{\ell}/|\Delta\underset{\sim}{\ell}_\infty|$ are determined initially at the points where the streamlines and perpendicular-component field lines intersect. A generalized quadrilateral interpolation scheme followed by a fifth-order smoothing is then employed to determine the corresponding values at the computational grid points where values for ρ/ρ_∞ are available for calculation of $(\underset{\sim}{B}_P/B_\infty)_\perp$. At the bow shock, an exact formula is used

$$\begin{aligned} (|\Delta\underset{\sim}{\ell}|/|\Delta\underset{\sim}{\ell}_\infty|)^2 &= 1 + \cot^2\theta(1+D^2) - 2D\times\csc\theta\times\cot\theta\times\cos(\theta-\delta) \\ \psi &= \theta + \sin^{-1}\left\{[D\times\cot\theta\times\sin(\theta-\delta)]/(|\Delta\underset{\sim}{\ell}|/|\Delta\underset{\sim}{\ell}_\infty|)\right\} \end{aligned} \tag{58}$$

where

$$\begin{aligned} D^2 &= 1 - 4(M_\infty^2\sin^2\theta - 1)(\gamma M_\infty^2\sin^2\theta + 1)/[(\gamma+1)^2 M_\infty^4\sin^2\theta] \\ \cot\delta &= \tan\theta\times\left\{(\gamma+1)M_\infty^2/[2(M_\infty^2\sin^2\theta - 1)] - 1\right\} \\ \theta &= \tan^{-1}\left(\frac{dR_S}{dX}\right) \end{aligned} \tag{59}$$

Finally, the resultant magnetic field can then be expressed in components relative to any orthogonal (X,Y,Z) coordinate system. For convenience of illustration, we have chosen the point P to lie in the (X,Y) plane. Relative to this reference frame, the magnetic components are

$$B_X/B_\infty = \left[(|\underset{\sim}{B}|/B_\infty)_\parallel \cos\phi \cos\alpha_p + (|\underset{\sim}{B}|/B_\infty)_\perp \cos\psi \sin\alpha_p)\right]\cos\alpha_n$$

$$B_Y/B_\infty = \left[(|\underset{\sim}{B}|/B_\infty)_\parallel \sin\phi \cos\alpha_p + (|\underset{\sim}{B}|/B_\infty)_\perp \sin\psi \sin\alpha_p)\right]\cos\alpha_n$$

$$B_Z/B_\infty = (B/B_\infty)_n \sin\alpha_n \tag{60}$$

where ϕ is the local flow angle given by

$$\phi = \tan^{-1}\left(\frac{v}{u}\right) \tag{61}$$

and the interplanetary magnetic field angles α_p and α_n indicated in figure 7 are defined by

$$\alpha_p = \tan^{-1}\left[\frac{B_{\infty_\perp}}{B_{\infty_\parallel}}\right] = \tan^{-1}\left[\frac{B_{Y_\infty}}{B_{X_\infty}}\right] \tag{62}$$

$$\alpha_n = \tan^{-1}\left[\frac{B_{\infty_n}}{\sqrt{(B_{\infty_\parallel})^2 + (B_{\infty_\perp})^2}}\right] = \tan^{-1}\left[\frac{B_{Z_\infty}}{\sqrt{(B_{X_\infty})^2 + (B_{Y_\infty})^2}}\right] \tag{63}$$

The generalizations of these results when the point P is at some arbitrary (Y,Z) location, i.e. not in the (X,Y) plane, are provided below in the spacecraft trajectory section.

Calculation of the Contour Lines

Contours are calculated for nondimensionalized velocity $|\underset{\sim}{v}|/v_\infty$, density ρ/ρ_∞, magnetic field components $(|\underset{\sim}{B}|/B_\infty)_\parallel$, $(|\underset{\sim}{B}|/B_\infty)_\perp$, and $(B/B_\infty)_n$ by application of a modified version of a contour procedure developed at NASA/Ames Research Center. After specifying a value for the contour line, the boundary is searched for intervals which bracket the selected value. After locating one such point by interpolation, the remainder of the contour is determined by 'walking' around the contour, searching at each step for the interval and then interpolating to find the point through which the contour line next passes. This is repeated until a boundary point is reached. Then closed contours are found in a similar manner. Linear interpolation is used throughout the process. Since the temperature is a function of $|\underset{\sim}{v}|/v_\infty$ only for a specified M_∞ and γ,

$$T/T_\infty = 1 + \left[\left(\frac{\gamma - 1}{2}\right) M_\infty^2\right]\left[1 - \left(\frac{|\underset{\sim}{v}|}{v_\infty}\right)^2\right] \qquad (64)$$

velocity contours may also be considered as temperature contours with only a relabeling required. The coordinates of the contour lines are output either or both as listings and pen plots.

Solar-Ecliptic/Solar-Wind Coordinate Transformations

In order to facilitate comparison of results from the current theoretical model with actual observational data obtained by a spacecraft, it is necessary to consider the appropriate transformations between the spacecraft and solar-wind coordinate systems. Part of the data required as input to the theoretical model consists of oncoming interplanetary values of solar-wind temperature, density, and velocity and magnetic-field vector components. These are naturally obtained in the spacecraft coordinate system, and are usually reported in a sun-planet or solar-ecliptic reference frame. The key coordinate system for the theoretical model is one which aligns the axial direction with the oncoming solar wind, since the gasdynamic calculation is assumed to be axisymmetric about this direction. Thus, the interplanetary input data must be transformed to the solar-wind system to initiate the theoretical determination. Once the gasdynamic and magnetic-field calculations in the solar-wind system are complete, those results must then be transformed back to the sun-planet system to allow direct comparison with spacecraft data obtained at arbitrary locations in the solar-wind/ionosphere interaction region. Consequently, direct and inverse transformations for both spatial coordinates as well as vector quantities between these reference frames are required.

For the measurements of the oncoming interplanetary solar-wind velocity we have assumed that the velocity is obtained with reference to a sun-planet (x_s, y_s, z_s) system with origin at planetary center and in which the x_s-axis points to the sun, the y_s-axis is opposite to the planetary orbital motion, and the z_s-axis points northward. The direction of the oncoming solar wind is such that the total abberation or azimuthal angle, including planetary orbital motion, of the solar-wind velocity vector in the plane of the ecliptic is Ω and the out-of-ecliptic plane or polar angle is ϕ_p. The positive sense of the azimuthal angle is for east-to-west flow and for the polar angle for north-to-south flow, as indicated in figure 8. In that figure we have also indicated the solar-wind (x, y, z) coordinate system so defined by (Ω, ϕ_p). For the gas-dynamic calculation, the (x, y, z) system is somewhat inconvenient since the direction of solar-wind flow is in the negative x-direction. Hence, the internal gasdynamic and magnetic-field calculations are performed in an (X, Y, Z) system as shown in figure 9.

The coordinate and vector transformations from the ecliptic sun-planet (x_s, y_s, z_s) system to the (X, Y, Z) solar-wind system are given by

$$\begin{pmatrix} Q_X \\ Q_Y \\ Q_Z \end{pmatrix} = \begin{pmatrix} -\cos\Omega\cos\phi_p & -\sin\Omega\cos\phi_p & \sin\phi_p \\ \sin\Omega & -\cos\Omega & 0 \\ -\cos\Omega\sin\phi_p & \sin\Omega\sin\phi_p & \cos\phi_p \end{pmatrix} \begin{pmatrix} Q_{x_s} \\ Q_{y_s} \\ Q_{z_s} \end{pmatrix} \tag{65}$$

where (Q_X, Q_Y, Q_Z) represents the Cartesian components of any vector referred to the solar-wind (X, Y, Z) coordinate system, and $(Q_{x_s}, Q_{y_s}, Q_{z_s})$ represents the corresponding vector in the sun-planet ecliptic (x_s, y_s, z_s) system. Thus, for a transformation of coordinates

$$\begin{aligned} (Q_x, Q_y, Q_z) &= (X, Y, Z) \\ (Q_{x_s}, Q_{y_s}, Q_{z_s}) &= (x_s, y_s, z_s) \end{aligned} \tag{66}$$

while for a vector transformation of, say, the magnetic field

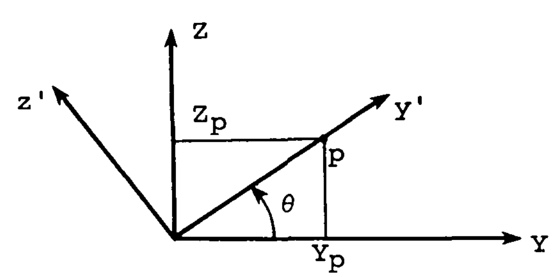
$$\begin{aligned} (Q_x, Q_y, Q_z) &= (B_x, B_y, B_z) \\ (Q_{x_s}, Q_{y_s}, Q_{z_s}) &= (B_{x_s}, B_{y_s}, B_{z_s}) \end{aligned} \tag{67}$$

The inverse transformation from the solar-wind to the sun-ecliptic system is given by

$$\begin{Bmatrix} Q_{X_S} \\ Q_{Y_S} \\ Q_{Z_S} \end{Bmatrix} = \begin{bmatrix} -\cos\Omega\cos\phi_p & \sin\Omega & -\cos\Omega\sin\phi_p \\ \sin\Omega\cos\phi_p & -\cos\Omega & \sin\Omega\sin\phi_p \\ \sin\phi_p & 0 & \cos\phi_p \end{bmatrix} \begin{Bmatrix} Q_X \\ Q_Y \\ Q_Z \end{Bmatrix} \tag{68}$$

Properties Along a Spacecraft Trajectory

One of the primary aims of the present effort has been the development of the capability to determine plasma and magnetic-field properties, as predicted by the present theoretical model, at locations specified along an arbitrary spacecraft trajectory, and in such a form as to enable comparisons to be made directly with actual spacecraft data. To this end, the following procedure has been developed and implemented in the associated computer code. First, from the known oncoming interplanetary conditions provided in a sun-planet reference frame, the azimuthal and polar solar-wind angles (Ω,ϕ_p) are employed to establish both the location of the trajectory point in the solar-wind (X,Y,Z) frame as well as the interplanetary magnetic-field components $(B_{X_\infty},B_{Y_\infty},B_{Z_\infty})$ using the transformation eq. (65). Next, the axisymmetric gasdynamic and unit magnetic-field calculations are carried out. Because the gasdynamic flow is axisymmetric in the (X,Y,Z) system, the internal coordinate system in which the trajectory calculations are actually performed may be rotated about the X axis into the most convenient orientation. If we consider a point P located at (X_p,Y_p,Z_p), then the rotation most appropriate for the present application is indicated in the sketch below



where the angle θ is given by

$$\theta = \tan^{-1}\left[\frac{Z_p}{Y_p}\right] \tag{69}$$

This rotation defines a new coordinate system (x',y',z') where

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix} \tag{70}$$

in which

$$\begin{aligned} x' &= X_p \\ y' &= \sqrt{Y_p^2 + Z_p^2} \\ z' &= 0 \end{aligned} \tag{71}$$

Thus, the (x',y') plane which contains the X axis and the arbitrary point P corresponds directly to the plane $(X,R) = (X_p, \sqrt{Y_p^2 + Z_p^2})$ in which the axisymmetric gasdynamic flow properties are calculated. In particular, the velocity magnitude v, density ρ, and flow angle ϕ at the point P are found by bilinear interpolation through the (X,R) flow-field grid. The vector velocity in the (X,Y,Z) system is then given by the transformation

$$\begin{pmatrix} v_X \\ v_Y \\ v_Z \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} v\cos\phi \\ v\sin\phi \\ 0 \end{pmatrix} \tag{72}$$

and then in the sun-ecliptic system by the transformation given in eq. (68)

$$\begin{pmatrix} v_{x_s} \\ v_{y_s} \\ v_{z_s} \end{pmatrix} = \begin{pmatrix} -\cos\Omega\cos\phi_p & \sin\Omega & -\cos\Omega\sin\phi_p \\ \sin\Omega\cos\phi_p & -\cos\Omega & \sin\Omega\sin\phi_p \\ \sin\phi_p & 0 & \cos\phi_p \end{pmatrix} \begin{pmatrix} v_X \\ v_Y \\ v_Z \end{pmatrix} \tag{73}$$

Calculation of the magnetic field at an arbitrary point is somewhat more complicated since these components are dependent upon the orientation of the incident interplanetary magnetic field. With the known (B_{X_∞}, B_{Y_∞}, B_{Z_∞}) components, the corresponding components (B'_{X_∞}, B'_{Y_∞}, B'_{Z_∞}) in the rotated (x',y',z') system are given by

$$\begin{pmatrix} B'_{X_\infty} \\ B'_{Y_\infty} \\ B'_{Z_\infty} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} B_{X_\infty} \\ B_{Y_\infty} \\ B_{Z_\infty} \end{pmatrix} \tag{74}$$

In this reference frame, the perpendicular, parallel, and normal interplanetary components are identified as

$$\begin{aligned} B_{\infty_\parallel} &= B'_{X_\infty} \\ B_{\infty_\perp} &= B'_{Y_\infty} \\ B_{\infty_n} &= B'_{Z_\infty} \end{aligned} \tag{75}$$

Then, the magnetic field angles α'_p and α'_n in the rotated system are given by

$$\alpha'_p = \tan^{-1}\left[\frac{B_{\infty_\perp}}{B_{\infty_\parallel}}\right] = \tan^{-1}\left[\frac{B'_{Y_\infty}}{B'_{X_\infty}}\right] \tag{76}$$

$$\alpha'_n = \tan^{-1}\left[\frac{B_{\infty_n}}{\sqrt{\left(B_{\infty_\parallel}\right)^2 + \left(B_{\infty_\perp}\right)^2}}\right] = \tan^{-1}\left[\frac{B'_{Z_\infty}}{\sqrt{\left(B'_{X_\infty}\right) + \left(B'_{Y_\infty}\right)^2}}\right] \tag{77}$$

The magnetic angle ψ associated with the incident perpendicular component and the unit magnetic-field ratios $(|\tilde{B}|/B_\infty)_\parallel$, $(|\tilde{B}|/B_\infty)_\perp$, $(B/B_\infty)_n$ in the rotated system are next determined by bilinear interpolation through the flow-field grid. Then, the magnetic-field components (B'_x,B'_y,B'_z) in the rotated system are calculated from

$$B'_x = \cos\alpha'_n \left(\cos\phi \cdot \cos\alpha'_p \cdot \left|\frac{\tilde{B}}{B_\infty}\right|_\parallel + \cos\psi \cdot \sin\alpha'_p \cdot \left|\frac{\tilde{B}}{B_\infty}\right|_\perp\right) \cdot B_\infty \tag{78}$$

$$B'_y = \cos \alpha'_n \left\{ \sin \phi \cdot \cos \alpha'_p \cdot \left|\frac{\underset{\sim}{B}}{B_\infty}\right|_{\parallel} + \sin \psi \cdot \sin \alpha'_p \cdot \left|\frac{\underset{\sim}{B}}{B_\infty}\right|_{\perp} \right\} \cdot B_\infty \tag{79}$$

$$B'_z = \sin \alpha'_n \cdot \left(\frac{B}{B_\infty}\right)_n \cdot B_\infty \tag{80}$$

The magnetic-field components in the solar-wind (X,Y,Z) system are then determined from the rotational transformation.

$$\begin{pmatrix} B_X \\ B_Y \\ B_Z \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta & -\sin \theta \\ 0 & \sin \theta & \cos \theta \end{pmatrix} \begin{pmatrix} B'_x \\ B'_y \\ B'_z \end{pmatrix} \tag{81}$$

and finally in the sun-planet system from

$$\begin{pmatrix} B_{x_s} \\ B_{y_s} \\ B_{z_s} \end{pmatrix} = \begin{pmatrix} -\cos \Omega \cos \phi_p & \sin \Omega & -\cos \Omega \sin \phi_p \\ \sin \Omega \cos \phi_p & -\cos \Omega & \sin \Omega \sin \phi_p \\ \sin \phi_p & 0 & \cos \phi_p \end{pmatrix} \begin{pmatrix} B_X \\ B_Y \\ B_Z \end{pmatrix} \tag{82}$$

RESULTS

Using the computational procedures developed under the current modeling effort, a large variety and number of different solar-wind/planetary-ionosphere interraction results were systematically obtained. These results were directed toward the following specific objectives: (1) verification of the correctness of the procedures, (2) demonstration of their flexibility and generality for a variety of cases covering ranges typical of solar-wind conditions, (3) establishment of a catalog of flow and magnetic-field results for a large number of solar-wind flows, and (4) comparisons of theoretical predictions with data obtained from spacecraft measurements. The results obtained associated with these objectives are discussed below.

Verification of the correctness of the procedures developed under the current effort primarily involved testing the computational extensions developed regarding both the gasdynamic and magnetic-field calculation methods reported in ref. 9. For the gasdynamic solver, this consists of demonstrating the extended blunt-body capability. As discussed previously in the section describing the nose-region solution and also in section A.2.1.1 of the computer manual, that extension involves the addition to the nose-region flow field of a region downstream of the dawn-dusk terminator - which is the usual plane terminating the nose-region solution. This added capability effectively removes any restriction with regard to obstacle shape and interplanetary gasdynamic Mach number of the previous procedures (ref. 9); and permits the calculation of ionopause shapes which have significant flaring in the radial direction at the dawn-dusk terminator, as well as flows at very low ($M_\infty \simeq 2.0$) free-stream Mach numbers. In figure 10, we present results for the bow shock locations for $M_\infty = 8.0$, $\gamma = 5/3$ flow past constant scale-height ionopause shapes (see eq. (36a) with $H/R_O = 0.5$ and 1.0. The downstream solutions for neither of these shapes could be determined with the previous procedures (ref. 9), whereas with the present method they present no problem. The downstream locations to which the nose-region solutions were extended were $X/R_O = (0.54, 0.67)$, respectively, for the $H/R_O = (0.5, 1.0)$ ionopause shapes - indicating that the addition of an extensive downstream region to the nose solution for such flows is unnecessary. This is important, as the nose-region solver requires significantly more computational time for a given flow-field region than the marching solver. Hence, minimization of the nose-region flow field is essential in minimizing the total computational time.

In figure 11, we display additional results using the extended nose-region grid capability to demonstrate the ability of the current method for calculating very low interplanetary gasdynamic Mach number flows. Bow shock locations are shown for $M_\infty = 2.0$ and 3.0, $\gamma = 5/3$ flows past an ionopause obstacle shape with gravitational variation included in scale-height having $\overline{H}/R_O = 0.25$ (see eq. (36b)). This particular obstacle is a relatively blunt shape, as can be observed from the ionopause profiles presented previously in figure 1, and, computationally, presents a more difficult flow to determine than flows for shapes having less flaring. For applications to terrestrial planets, such as Mars and Venus, typical ionopause shapes occurring in nature appear to lie in the range $0.01 \lesssim \overline{H}/R_O \lesssim 0.10$. Consequently, demonstration of the ability of the current procedures to treat successfully such flows as shown in figures 10 and 11 - which lie at the limits of interest as far as applications to nonmagnetic terrestrial planets, indicates that these procedures will not be restricted insofar as ionopause geometry and interplanetary solar-wind conditions are concerned.

Corresponding verification of the extensions to the procedures for the magnetic-field calculation has involved demonstration of the correctness of the magnetic-field prediction at any arbitrary point in the solar-wind flow field. This was accomplished by consideration of a variety of special test cases in which the location in the flow field and the incident interplanetary magnetic-field orientation were systematically changed so as to produce both symmetric and antisymmetric changes in the resultant ionosheath magnetic field, as well as to reverse the roles of the perpendicular and normal components. All of these various permutations of the magnetic-field calculation procedure were successfully verified.

One of the primary objectives of the present work was to demonstrate the flexibility and generality of the present procedures by exercising them over a wide range of ionopause geometries and solar wind oncoming conditions so as to cover, insofar as possible, the entire range of practical interest of these parameters. These calculations were to be summarized in a convenient format and then archived so as to provide at-a-glance information regarding the variation of the flow-field and unit magnetic-field quantities. The output format selected was the automatic pen-plot output option of the program involving plots of the flow-field streamlines, and contours of the velocity magnitude $|\underset{\sim}{v}|/v_\infty$, temperature T/T_∞, density ρ/ρ_∞, and the field-line locations and contours of the unit magnetic-field ratios $(|\underset{\sim}{B}|/B_\infty)_\parallel$ and $(|\underset{\sim}{B}|/B_\infty)_\perp$.

The test cases selected for this catalog involved a ratio of specific heats $\gamma = 5/3$ and the following matrix of free-stream Mach numbers M_∞ and ionopause shapes:

$$M_\infty = \{2.0, 3.0, 5.0, 8.0, 12.0, 25.0\}$$
$$H/R_o = \{0.01, 0.10, 0.25\}$$
$$\overline{H}/R_o = \{0.10, 0.20, 0.25\}$$

Thus, a total of 36 separate cases were calculated. The plot output for these cases is provided in Appendix B, which also presents a convenient page index to the individual results. These archived results provide an very convenient means of determining the overall dependence of flow-field and magnetic-field quantities with M_∞ and obstacle shape, in particular the variation of bow shock location and flow-field contour changes. We note that the range of free-stream Mach numbers selected easily spans the entire range of solar-wind conditions usually encountered, while the different obstacle shapes provide a wide variation as well, as can be observed from figure 1.

The final and ultimate check of the current procedures lies in the comparison of the results predicted by the present model with data actually measured by a spacecraft. To that end, we have made a number of preliminary comparisons with data obtained from orbits 3 and 6 of the Pioneer-Venus Orbiter spacecraft.

The overall features of the spacecraft trajectory crossings of the solar-wind/Venus-ionosphere interaction region are provided in the sketch given in figure 12. In that figure, which is referred to the sun-Venus solar-ecliptic coordinate system, we note in particular the highly elliptic spacecraft orbit (periapsis $\simeq$ 200 Km, apoapsis $\simeq$ 66,000 Km) and the crossings of the bow shock and ionopause surfaces. The oncoming solar-wind direction, with arbitrary azimuthal (aberration) and polar angles (Ω, ϕ_p) is as indicated, with the ionopause and bow shock surfaces symmetric about that direction. The oncoming arbitrary interplanetary magnetic field $\underset{\sim}{B}$ is also as indicated.

The procedural outline employed for the theoretical comparisons is as follows:

I. Orbital data selection

Select data from an orbit when solar-wind conditions are relatively steady.

II. Theoretical calculations

Input:

Ionospheric ρ and T versus altitude from orbiter retarding potential analyzer (ORPA)

Solar wind $\underset{\sim}{v}_{\infty}$, ρ_{∞}, T_{∞} from orbiter plasma analyzer (OPA)

Solar wind $\underset{\sim}{B}_{\infty}$ from orbiter magnetometer (OMAG)

Trajectory coordinates

Output: (Contours and/or time histories along orbital trajectory)

Ionosheath ρ, T, $\underset{\sim}{v}$, $\underset{\sim}{B}$ and their scalar components in solar ecliptic coordinates

III. Comparisons with Spacecraft data

Observational ionosheath data for ρ, $|\underset{\sim}{v}|$, T from OPA and for $\underset{\sim}{B}$ from OMAG with two sets of theoretical predictions based on $\left\{\begin{matrix}\text{last}\\ \text{first}\end{matrix}\right\}$ interplanetary solar-wind properties (v_{∞}, T_{∞}, ρ_{∞}, $\underset{\sim}{B}_{\infty}$) measured $\left\{\begin{matrix}\text{before}\\ \text{after}\end{matrix}\right\}$ bow shock $\left\{\begin{matrix}\text{inbound}\\ \text{outbound}\end{matrix}\right\}$ crossings.

First, the selection of the particular orbit for which theoretical calculations and data comparisons will be carried out must be made. This choice is based on spacecraft observations of the oncoming interplanetary solar wind, and for the cases reported here, the selections were made when conditions appeared relatively steady. In particular, the interplanetary conditions regarding solar-wind velocity, density, temperature and magnetic field based on the orbiter solar-wind plasma analyzer (OPA) and fluxgate magnetometer (OMAG) measurements just prior to inbound bow shock crossing and immediately after outbound bow shock crossing were analyzed by the Pioneer-Venus investigators responsible for these instruments for a number of the initial orbits of the Pioneer-Venus spacecraft, and on this basis the selection of Orbits 3 and 6 were made.*

To initiate the theoretical calculations, information regarding both the ionospheric obstacle shape and the oncoming interplanetary conditions are required. The determination of the obstacle shape is based on measurements of atmospheric density and temperature as a function of altitude made by the orbiter retarding potential analyzer at (ORPA) locations interior to the ionopause boundary.** These measurements yield the variation of atmospheric pressure with altitude in the vicinity of ionopause altitudes. From this information, the value of the scale-height parameter from the atmospheric pressure model given by either eq. (29) or (30) can be determined. For Venus, it appears that the ionosphere/solar-wind interaction is such that the ionopause wraps tightly about the planet (ref. 10). Our calculations based on ORPA data for Orbits 3 and 6 indicate scale heights of approximately 200 Km, which yield a corresponding range of values for H and $\overline{H}$ of $0.02 \lesssim H/R_O, \overline{H}/R_O \lesssim 0.05$. We note that for such small values of scale height, the two ionospheric pressure models eqs. (29) and (30) yield essentially the same obstacle shape, as can be seen from figure 1. For the comparisons reported here for both Orbits 3 and 6, we have selected a value of $\overline{H}/R_O = 0.03$. With regard to oncoming interplanetary conditions, we require as input the solar-wind bulk velocity $\underset{\sim}{v}_\infty$, density ρ_∞, temperature T_∞, and magnetic field $\underset{\sim}{B}_\infty$. The first three are provided by the orbiter plasma analyzer (OPA), while the magnetic field is given by the orbiter fluxgate magnetometer (OMAG). We note that the OPA provides either ion density and temperature or electron density and temperature, but not both simultaneously. For orbits 3 and

---

*Special thanks are due to J. H. Wolfe and J. P. Mihalov who provided information regarding the solar-wind plasma from OPA measurements (refs. 21,22) and to C. T. Russell, R. C. Elphic, and J. A. Slavin for magnetic-field information from OMAG measurements (refs. 23,24).

**Special thanks are due to W. C. Knudsen and K. Spenner for providing the ionospheric plasma information from ORPA measurements (refs. 10,11).

6, ion measurements were available and have been employed. Information regarding the oncoming direction of the solar wind, as given by the angles (Ω, ϕ_p), defines the requisite coordinate rotations required to align the gasdynamic calculation in the oncoming solar-wind direction; while information of solar-wind speed, density, and temperature serve to define the oncoming gasdynamic Mach number required to initiate the gasdynamic calculations.

With this information, the detailed gasdynamic and unit magnetic-field calculations in the ionosheath region can be carried out. In order to provide an idea of the detail obtained by the present computational procedures in these calculations, we have displayed in figure 13 the flow-field grid for one of the gasdynamic flow solutions used in the data comparisons discussed below. The result shown is for $M_\infty = 3.0$, $\gamma = 5/3$ flow past an ionopause obstacle shape with $\overline{H}/R_o = 0.03$, and is shown carried to a downstream location of $X/R_o = 3.0$. The flow field properties $[\underset{\sim}{v}/v_\infty, \rho/\rho_\infty, T/T_\infty]$ and the unit frozen magnetic-field ratios $[(\underset{\sim}{B}/B_\infty)_\parallel, (\underset{\sim}{B}/B_\infty)_\perp, (B/B_\infty)_n]$ are determined at each intersection of the grid lines, including the bow shock, stagnation streamline, and ionopause boundaries. The final output of the calculaton consists of detailed flow-field and magnetic-field properties in the ionosheath region, both in terms of tabular output and plotted contours and time histories along the orbital trajectory of the velocity magnitude and components, density, temperature, and magnetic-field magnitude and components. Complete details are provided in section A.4 of the Computer User's Manual.

For the comparisons with spacecraft data, the most convenient portion of the output format are the time-history predictions along the spacecraft orbit. The observational data used for comparisons with the theoretical predictions in the ionosheath region include plasma density, velocity, and temperature from OPA measurements and magnetic field from OMAG measurements. For the theoretical predictions, two sets of results are usually generated based on $\begin{Bmatrix}\text{last}\\\text{first}\end{Bmatrix}$ interplanetary solar-wind properties $(\underset{\sim}{v}_\infty, T_\infty, \rho_\infty, \underset{\sim}{B}_\infty)$ measured $\begin{Bmatrix}\text{before}\\\text{after}\end{Bmatrix}$ bow shock $\begin{Bmatrix}\text{inbound}\\\text{outbound}\end{Bmatrix}$ crossing.

In figure 14, we have displayed some overall flow-field results for Orbit 6. Indicated in that figure are bow shock locations for the three combinations of free-stream Mach number M_∞ and plasma specific heat ratio γ, i.e. $(M_\infty,\gamma) = (13.3,5/3), (13.3,2), (3.0,5/3)$ for flow about an ionopause with $\bar{H}/R_o = 0.03$. Also indicated are two sets of points (-⊙-, -⊡-) representing the spacecraft trajectory for orbit 6 as viewed in two solar-wind oriented coordinate systems. The trajectory indicated by the solid lines and circles (-⊙-) is that based on the last measured direction $(\Omega,\phi_p) = (6.5°,-1.4°)$ of the interplanetary solar wind just prior to crossing the bow shock on the inbound leg, while the dashed line and squares (--⊡--) denotes the trajectory based on the first measured direction $(\Omega,\phi_p) = (4.9°,7.6°)$ of the solar wind immediately after crossing the bow shock on the outboard leg. We note that the spatial location of the spacecraft trajectory in solar-wind coordinates depends only on the direction (Ω,ϕ_p) of the oncoming solar wind, but not on its magnitude. With regard to the results indicated in figure 14 for the spacecraft trajectory, we observe the extremely large dependence of spatial position of a trajectory point, as viewed in solar-wind coordinates, on solar-wind direction. For the particular inbound and outbound solar-wind angles indicated, the shift in X-coordinate of a trajectory point can be as high as a quarter of the Venusian planetary radius, which obviously results in substantial differences in predicted flow and magnetic-field properties. In previous work, the influence of the angular shift in the solar wind was generally considered to be small and negligible. The current results, however, indicate that this purely geometrical effect can be surprisingly large, even for directional shifts of less than 5°, and must be accounted for in any realistic theoretical comparison with data. See reference 7 for another example of the importance of this effect.

Finally, with regard to the three sets of bow shock results displayed in figure 14, these calculations represent an attempt to resolve the uncertainty in the oncoming free-stream Mach number and ratio of specific heats of the plasma. Because only solar-wind ion temperatures from the OPA were available for Orbit 6, the initial calculation of the free-stream Mach number was based on the assumption that $T_e = T_i$, which leads to $M_\infty = 13.3$. A ratio of specific heats $\gamma = 5/3$ was assumed, and these interplanetary values result in the bow shock indicated by the dot-dash line. That shock location is in poor agreement with the observational shock crossings, indicated as occurring between the pairs of solid circles and squares. A separate uncertainty arises from the possibility that the magnetic field may act to align the plasma particle motion in its direction, thus effectively reducing the number of degrees of translational freedom from 3 to 2 and thereby increasing the ratio of specific heats from 5/3 to 2. To investigate this possibility, we have repeated the $M_\infty = 13.3$ calculation using $\gamma = 2$. That result is indicated by the dashed line, and is in better but still not completely satisfactory agreement with the observed shock locations. Finally, if it is assumed that the oncoming interplanetary electron temperature is not equal to the ion temperature, but is substantially higher, we are lead to low Mach numbers of the order of $M_\infty \simeq 3$-5. We have displayed bow shock results of a $M_\infty = 3.0$, $\gamma = 5/3$ calculation in figure 14 as the solid line, and observe that based on this Mach number and the inbound solar-wind direction, the observational shock crossings display very good agreement with the theoretical results.

Figure 15 displays the time-history comparisons of the theoretically-predicted bulk plasma density, speed and temperature in the ionosheath region with OPA measurements of these quantities. These theoretical results were based on a gasdynamic flow solution with $M_\infty = 13.3$, $\gamma = 2.0$. In these results, the solid lines with circles correspond to results based on inbound interplanetary conditions, while the dashed lines with squares correspond to outbound conditions. We note that while the few data points available are in general agreement with the theoretical calculations, the lack of more detailed plasma measurements in the ionosheath prevents a definitive conclusion. The OPA instrument requires approximately 9 minutes to acquire sufficient data to enable predictions of the bulk plasma quantities. While this time lag presents no problem when the spacecraft is in the interplanetary solar wind, the large resolution time effectively averages the plasma quantities in the ionosheath over such a large spatial range that only overall comparisons of the bulk plasma properties are possible.

The situation is quite different for the magnetic field, as the OMAG instrument provides essentially instantaneous magnetic-field measurements. Comparisons of the frozen magnetic-field predictions with data are displayed in figures 16(a,b). These comparisons employ the gas-dynamic solution $M_\infty = 13.3$, $\gamma = 2$ for which plasma properties were given in figure 15. In figure 16a, we display two sets of theoretical calculations for the magnitude of the magnetic field, based on the inbound and outbound interplanetary magnetic field conditions as indicated on the figure. In these comparisons, we observe very good agreement with both sets of predictions. In particular, on the inbound leg, the theoretical predictions based on the inbound interplanetary conditions are in very good agreement with the data, while the outbound-condition predictions are clearly not as favorable. On the other hand, as we proceed in time along the outbound leg, the opposite is true. Here, the outbound-condition predictions are in very good agreement with the data, while the inbound-condition predictions are notably inferior, particularly with regard to shock crossing. Corresponding results for the magnetic-field components are provided in figure 16b, and display a similar behavior. The agreement of the theoretical results with data for the individual components is remarkable, confirming the accuracy of the frozen-field model, as well as the shift of the ionosheath magnetic field from a solution related to inbound interplanetary conditions to one related to outbound conditions.

For Orbit 3, similar comparisons as those shown in figures 14-16 for Orbit 6 are given in figures 17 to 19. In figure 17, we have provided the bow shock locations for five different combinations of M_∞ and γ as indicated. The Mach numbers $M_\infty = 7.38$, 5,96 correspond, respectively, to the inbound and outbound interplanetary conditions for $|\underset{\sim}{v}_\infty|$, ρ_∞, T_∞ as measured by the OPA, while the two values of $\gamma = 5/3, 2$ used in the calculations represent our uncertainty of the ratio of plasma specific heats. We have also indicated for reference the bow shock location for $M_\infty = 3.0$, $\gamma = 5/3$ as given previously in figure 14 for Orbit 6. Note that the observational shock crossings are again closest to the $M_\infty = 3.0$, $\gamma = 5/3$ shock. Also provided in figure 17 are the orbital trajectories as viewed in solar-wind coordinates for the inbound $(\Omega,\phi_p) = (3.3°,0.15°)$ and outbound $(\Omega,\phi_p) = (3.7°,4.9°)$ solar-wind directions.

The comparisons for the bulk plasma properties for Orbit 3 are provided in figure 18. Again we note an overall agreement for bulk plasma speed and density, but note an observable discrepancy in the temperature.

This is thought to be indicative of the manner in which the bulk properties from the theoretical model are being interpreted in relation to the observational measurements; i.e. the theoretical values correspond to those for a single-component plasma, while the measurements are in terms of a multi-component plasma. Whether the theoretical plasma properties require rescaling or reformulation, or whether their present formulation is appropriate for comparison with the multi-component data, appears to be a necessary and important subject for future study.

Results for the magnetic-field comparisons are displayed in figures 19(a,b), which provide time-histories of both the magnitude and the individual magnetic-field components based on both inbound and outbound interplanetary conditions. We note again, although the shock crossing comparisons are somewhat in disagreement since the gasdynamic flow fields used in these results were for $M_\infty = 7.56$, 5.96 and $\gamma = 2$, the reasonable comparisons are obtained for the ionosheath magnetic field. In particular, we observe the drift with time along the trajectory of the trajectory of the agreement of theory with data from the predictions based on inbound interplanetary conditions on the inbound leg, to those based on outbound conditions on the outbound leg.

In order to demonstrate the improvement obtained in magnetic-field results when a gasdynamic flow-field solution is employed which more closely agrees with the observational bow shock location, we have displayed in figure 20(a,b) the analogous time-history magnetic-field comparisons when using a $M_\infty = 3.0$, $\gamma = 5/3$ gasdynamic result. In this case, results were computed for only the inbound direction $(\Omega,\phi_p) = (3.3°, 0.15°)$ of the solar wind. As can be seen, there is a marked improvement in the agreement near the bow shock, and quite good agreement throughout the ionosheath as well as, for both the magnitude and the individual magnetic-field components. We note that the general agreement of theory and observation of the individual components demonstrates both the accuracy of the calculation and, in particular, the need for accounting in the theoretical results of the variable direction of the interplanetary solar wind.

CONCLUDING REMARKS

The application of advanced computational procedures was undertaken for the purpose of modeling the interaction of the solar wind with non-magnetic planets, with particular emphasis on Venus. Based on the successful theoretical model employed previously (ref. 9), i.e., the steady, dissipationless, magnetohydrodynamic model for axisymmetric, supersonic, super-Alfvénic solar-wind flow, a number of important theoretical extensions have been developed and included in the computational procedures. These include the capability for treating very low oncoming interplanetary gasdynamic Mach numbers ($M_\infty \simeq 2.0$), as well as quite general ionopause shapes. A new family of ionopause shapes has been developed which includes the effect of gravitational variation in scale height, and has been incorporated in the computational program. Additionally, the capability for determining the plasma gasdynamic and magnetic-field properties along any arbitrary spacecraft trajectory, accounting for an arbitrary oncoming direction of the solar wind, has been developed. All of these developments have been incorporated into an assemblage of computer codes to enable detailed calculations of the solar-wind interaction with planetary atmospheres. The computer codes have been extensively documented and are described in a computer user's manual included as part of this report.

Comparisons are reported which verify the correctness of these new procedures, and which demonstrate their capability for computing a wide range of flows encompassing those typical of solar-wind conditions about terrestrial planetary atmospheres. A catalog of sample solar-wind flows covering a large number of flow conditions and ionopause geometries was established, and reported in summary format in the forms of contour plots of important flow-field and magnetic-field properties. Finally, successful comparisons of results from the theoretical model were made with actual spacecraft data obtained from initial orbits of the Pioneer-Venus Orbiter. These results have indicated the importance, heretofor largely neglected, of the directional variability of the oncoming solar wind. All of these results, taken in toto, serve to verify the basic theoretical model which underlies the present procedures. Furthermore, it demonstrates the value of the present computational procedures as a research tool capable of routinely providing - at small computation cost and in a format directly compatible with experimental observations - details of the solar-wind/planetary atmosphere interaction process not previously attainable.

With regard to future uses as well as improvements of the present model, the obvious need for a detailed study involving comparisons between theory and observations for a large number of orbits of the Pioneer-Venus Orbiter is clear. Based on the preliminary comparisons for orbits 3 and 6, the frozen magnetic-field model appears to be remarkably accurate for relatively quiet-time conditions. Similar comparisons of the plasma properties indicate a need for an improved interpretation of the results from the single-fluid theory in terms of multi-component measurements. Questions regarding the possible suppression by the interplanetary magnetic field of the number of degrees of freedom of the plasma require further study and could be clarified through systematic comparisons with data. Additionally, observations from the Pioneer-Venus Orbiter of the nightside ionosphere of Venus have indicated a more complex and dynamic structure than suspected. These observations point, in particular, toward the need for improvement of the simple model used in the present method for the determination of the ionosphere boundary. This improved determination would involve an iterative procedure in which a balance of the sum of the solar-wind gasdynamic plus magnetic pressure along the ionopause surface would be maintained against the ionospheric pressure. The present method, which balances the Newtonian pressure distribution against the ionospheric pressure, represents the first step in this iteration.

ACKNOWLEDGEMENTS

Support for the research reported in this investigation was provided by National Aeronautics and Space Administration, Headquarters under Contract NASW-3182 with Robert Murphy as Technical Monitor. Special thanks are given to J. H. Wolfe and J. D. Mihalov for generously providing solar-wind plasma information from Pioneer-Venus Orbiter plasma analyer measurements, to C. T. Russell and J. A. Slavin for magnetic field information from Pioneer-Venus Orbiter fluxgate magnetometer measurements, and W. C. Knudsen for ionosphere plasma information from Pioneer-Venus Orbiter retarding potential analyzer measurements.

APPENDIX A

COMPUTER PROGRAM USER'S MANUAL

APPENDIX A

COMPUTER PROGRAM USER'S MANUAL

TABLE OF CONTENTS

This Page Intentionally Left Blank

APPENDIX A

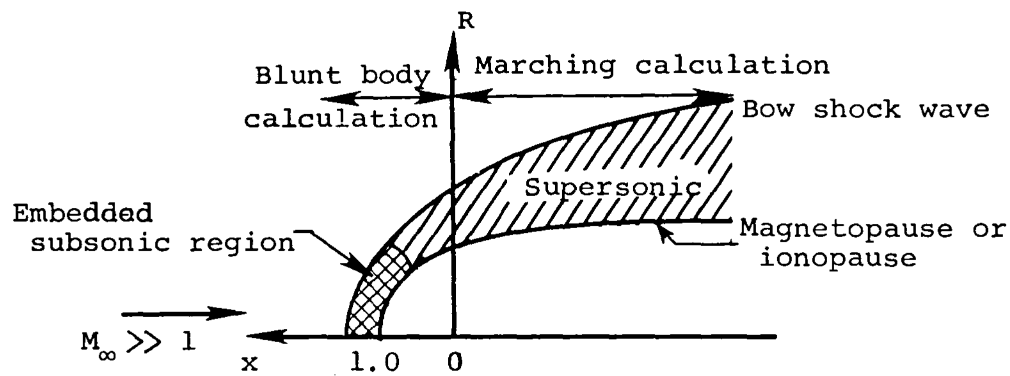
A.1 INTRODUCTION

The purpose of this appendix is to describe the operation of the assemblage of computer codes which were developed in conjunction with the theoretical work presented in this report and organized into one program, and to provide sufficient detail to permit understanding and use of the program. The program computes the flow field of the solar wind about a terrestrial planet, using a procedure for the calculation of supersonic/hypersonic flow about an axisymmetric blunt body. The corresponding frozen-in magnetic field is calculated from the previously-determined velocity and density fields. Streamlines and contour lines of various flow-field properties and magnetic-field components are also determined. Next, these flow-field and magnetic-field values are calculated for points along a user-specified trajectory.

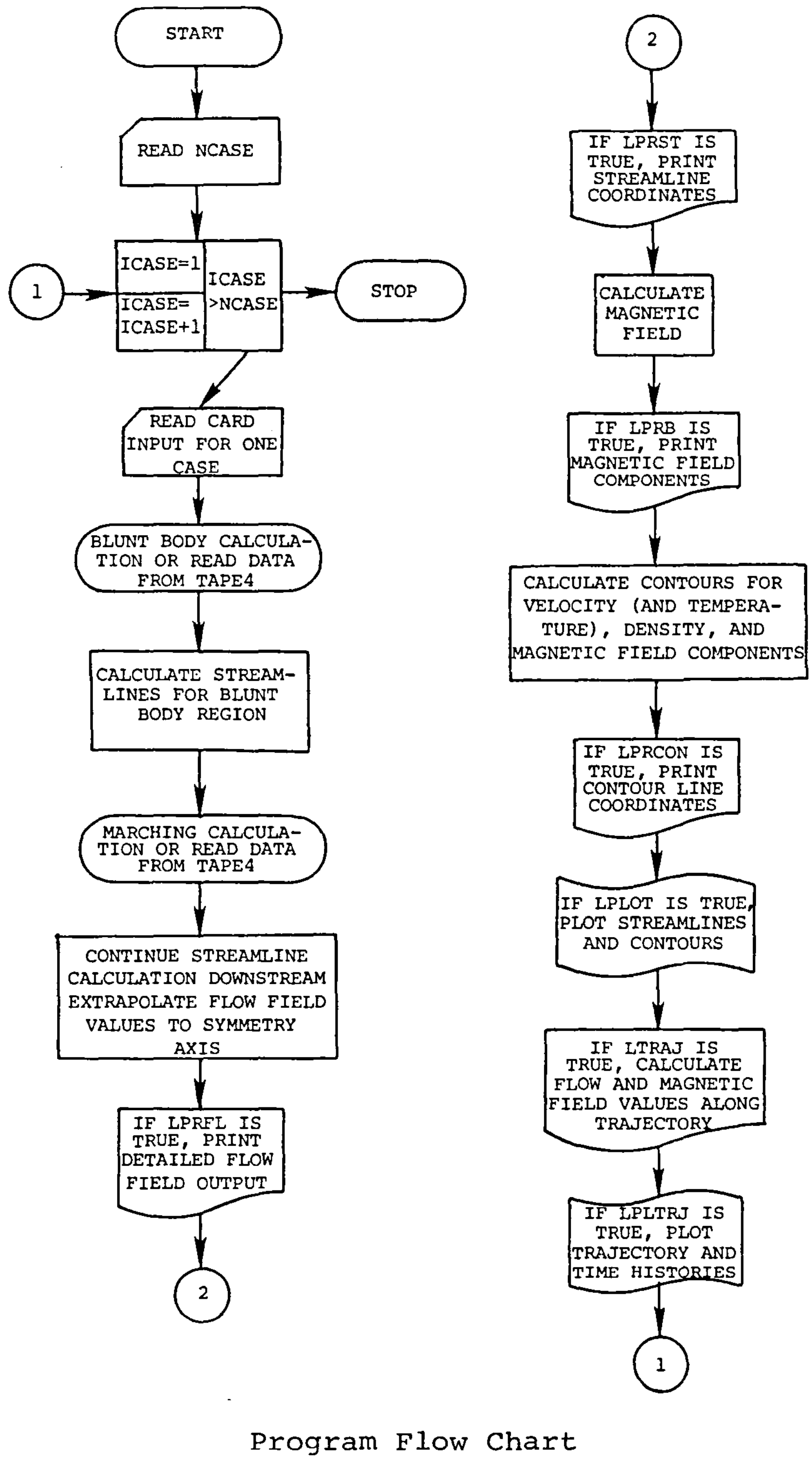
A description of the general operating procedure of the program is given, with descriptions of input and output. The program is written in FORTRAN IV and has been developed on a CDC 7600 computer. University Computing Company (UCC) Standard Plotting Software and Functional Software packages are used to produce automated plots. Files used, besides TAPE5 for INPUT and TAPE6 for OUTPUT, are TAPE1 for the plot file (system default), TAPE4 for input file for rerun option, and TAPE9 for storing data for rerun. Typical run times for cases using the default parameters are 110 to 120 seconds, using the OPT=2 compiler. For a case using the rerun option, which employs a previously-calculated flow field, the run time is approximately 15 seconds. The storage requirements are $146K_8$ for small core memory and $273K_8$ for large core memory.

A.2 PROGRAM DESCRIPTION

For computational purposes, the flow is subdivided into two regions, as indicated in the sketch below, with the center of the planet as origin.



The region near the nose of the magnetopause/ionopause includes all of the imbedded subsonic flow and part of the supersonic flow. An axisymmetric implicit unsteady Euler equation solver is used to calculate this part of the flow field. Using the solution plane at x = 0.0 to provide starting conditions, the flow field in the purely supersonic downstream region is determined by integrating the steady Euler equations using a spatial-marching procedure. Streamlines, the magnetic field, and contours are calculated using the entire flow field, distinguishing between the two regions as required by the different forms of the computational grids. A rerun capability is provided, where flow-field data is read from a file written on a previous run, rather than repeating the blunt-body and marching calculations. The computations proceed as shown in the sketch below, which provides an overall flow chart of the complete program. The program provides for several cases to be run consecutively.



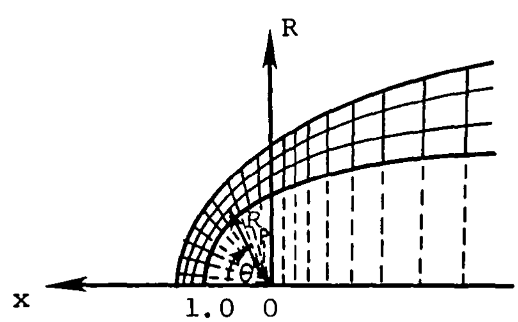
Program Flow Chart

A.2.1 Calculation Procedure

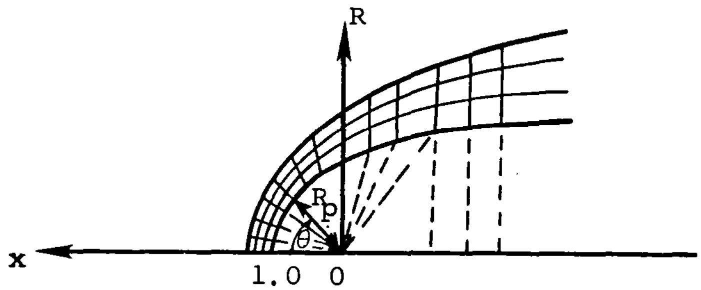
After reading in the number of cases in the run, each case is calculated independently. Subroutine INPUT reads in all card input required for one case, viz. a title, flow conditions, obstacle geometry, calculation and print control parameters, and desired contour values. The user may supply the obstacle geometry in the form of a shape table for an axisymmetric body, or use one of the default shapes which are calculated internally by the program. These default shapes are the magnetopause equatorial trace, constant scale-height ionopauses, and ionopauses having gravitational variation in scale-height. The input is printed as the first item of output.

A.2.1.1 Blunt-body calculation

A computational mesh in polar (R_p, θ) coordinates is established for the blunt-body calculation; then, for the marching calculation, this is extended into a cylindrical (x,R) system, as indicated below:



This method has proven effective except for certain obstacle shapes which have a significant amount of flaring at the terminator and/or cases involving low free-stream Mach numbers $M_\infty \leq 3$. Under such conditions, the axial component of velocity may become subsonic at the starting plane of a marching calculation (terminator) and the calculation cannot proceed. In this case the blunt-body grid must be extended past $\theta = 90^o$ as shown below:



The number of rays added to the blunt-body mesh is controlled by the input variable NXADD, and are limited by the requirement, NBLUNT + NXADD $\leq$ 39.

All lengths, x, R, R_p, are normalized so that the nose of the obstacle is at x = 1.0. For the default shapes, rays at equal angular increments of $\Delta\theta$ are used, starting at $-\Delta\theta/2$, up to $90^\circ + \Delta\theta$, where $\Delta\theta = 90^\circ/(\text{NBLUNT}-1.5)$, and NBLUNT is an input parameter describing the number of angular mesh points to be used in the blunt-body calculation. Program default value is NBLUNT = 24, so that for the default mesh, $\Delta\theta = 4^\circ$. The obstacle shape is determined by integrating the appropriate differential equation by a trapezoidal predictor-corrector method. For a user-supplied shape, the θ grid is determine by rays from the origin through the first NBLUNT points, and the reflection of the first ray about the x-axis. Values for R_p are determined by dividing the line segments between the body and bow shock wave into NR-1 equal intervals. Thus, including the obstacle and bow shock wave, the grid forms NR arcs around the obstacle. A starting solution for the blunt-body calculation is obtained by guessing a bow shock shape and by prescribing a Newtonian pressure distribution on the body. Noting that the maximum entropy streamline wets the body, other flow properties on the body surface can then be calculated. An initial flow field is then established by linear interpolation between the obstacle and the guessed bow shock, where the Rankine-Hugoniot relations hold. The integration proceeds in time for ITER steps. The initial bow shock shape used for the magnetopause equatorial trace and for an ionopause with $H/R_o \geq 0.1$ is a correlation shape depending on (M_∞, γ, H/R_o) and given by the parabola $R_p = \delta_1\sqrt{\delta_o - x}/\sqrt{\delta_o}$ where

$$\delta_o = 1.0 + 1.1 \{[(\gamma-1)M_\infty^2 + 2]/(\gamma+1)M_\infty^2\} \times (0.9 + 0.5\ H/R_o)$$

$$\delta_1 = \Delta_o \{(1.273 + 0.009\ M_\infty^2)(0.904 + 0.655\ H/R_o) \times [3.95 - 5.3\ H/R_o + 3.85\ (H/R_o)^2]\} + (R_{body})_{x=0.0}$$

$$\Delta_o = [(\gamma-1)M_\infty^2 + 2]/[(\gamma+1)M_\infty^2] \times 0.78$$

For a user-supplied obstacle shape and for an ionopause with $H/R_o < 0.1$, the initial shock shape used is the curve $R_p = \sqrt{[1 + \Delta_o(1 + 0.68\ \theta^2 + 0.16\ \theta^4)]}$. Information on convergence, the final sonic line locations, and the body and final bow shock shape are printed from this calculation.

The flow chart for the blunt-body code is shown in Figure A.1(a).

A.2.1.2 Marching calculation

The results at the $\theta = 90°$ plane of the blunt-body calculation are used as starting conditions for the marching calculation, after proper variable normalization for the internal marching calculation. For default geometries, the obstacle shape is determined by integration of the appropriate differential equation proceeding from the nose downstream at equal θ increments to form a body-shape table. The stepsize along the x-axis is recalculated at every ICONST(49) with ICONST(49) being set to 10. At each x-location, R_{body} is determined by linear interpolation. The computational mesh is extended by adding the line perpendicular to the x-axis at each step, divided in the same manner as for the blunt nose. The calculation marches downstream with a maximum stepsize of 1.0 until the terminal location specified by the user has been passed. However, the number of steps is limited to 75, after which the calculation will end regardless of the x-location. The coordinates of the obstacle and bow shock are printed at each step.

The grid coordinates and flow-field values are written to a file, TAPE9, which may be saved to use as input for a later run. This rerun option, which replaces construction of the computational mesh and performance of the blunt-body and marching calculations with the reading of the rerun input file TAPE4, is described in section A.2.2. The flow chart for the marching calculation is provided in figure A.1(b).

A.2.1.3 Streamline calculation

The streamlines are calculated in two sections, following each of the flow-field calculations. Using the results of the blunt-body calculation, i.e. the (x,R) grid coordinates, (R_p,θ) grid coordinates, density ρ/ρ_∞, and velocity components v_X/v_∞ and v_R/v_∞, the velocity magnitude $|\underset{\sim}{v}|/v_\infty$ and flow angle ϕ are calculated. Density ρ/ρ_∞ and velocity magnitude $|\underset{\sim}{v}|/v_\infty$ are first smoothed along the rays of constant-θ, using a third-degree least-squares fit with respect to R_p. Streamlines are then calculated downstream to x = 0.0, using the trajectory method and integrating through the velocity field by means of a third-order modified Euler integration procedure with the grid locations on the bow shock used as starting positions. The flow angle $\phi = \tan^{-1}(v_R/v_X)$ at each point is determined using bivariate linear interpolation first in θ, then R_p. Points for which $\theta < 0^\circ$ or $\theta > 90^\circ$ are discarded in the interpolation.

The marching calculation provides (x,R) grid coordinates, and values of density ρ/ρ_t, and velocity components v_X/v_t and v_R/v_t, where t denotes free-stream stagnation conditions. For compatibility with the blunt-body solution, the flow-field values are converted to ρ/ρ_∞, v_X/v_∞, v_R/v_∞ before calculating the resultant velocity magnitude

$|\underset{\sim}{v}|/v_\infty$ and flow angle ϕ. The streamline calculation is continued downstream, employing the same method as in the nose region. Starting positions on the shock wave for the streamline calculation in the marching zone are set at equal R-increments, with a maximum of 50 streamlines calculated. The flow angle is determined using bivariate linear interpolation first in x, then in R.

Along the symmetry axis, values of x, ρ/ρ_∞, and $|\underset{\sim}{v}|/v_\infty$ are determined by extrapolation, using a third-order Lagrangian polynomial in θ on each arc of the computational grid. Exact values for the stagnation streamline are used where possible, viz. at the bow shock

$$\rho/\rho_\infty = (\gamma+1)M_\infty^2/[(\gamma-1)M_\infty^2 + 2]$$

$$|\underset{\sim}{v}|/v_\infty = 1/(\rho/\rho_\infty)$$

at the body surface

$$\rho/\rho_\infty = (\rho/\rho_\infty)_{\text{shock}} \cdot \left\{\left[\left((\gamma+1)M_\infty\right)^2 / \left(4\gamma M_\infty^2 - 2(\gamma-1)\right)\right]^{1/(\gamma-1)}\right\}$$

$$|\underset{\sim}{v}|/v_\infty = 0.0$$

$$x = 1.0$$

Detailed flow-field output may now be printed by subroutine FLOUT, with LPRFL as print control variable. In addition to grid coordinates, density, velocities and flow angle, values of temperature T/T_∞ and pressure P/P_∞ are output, where

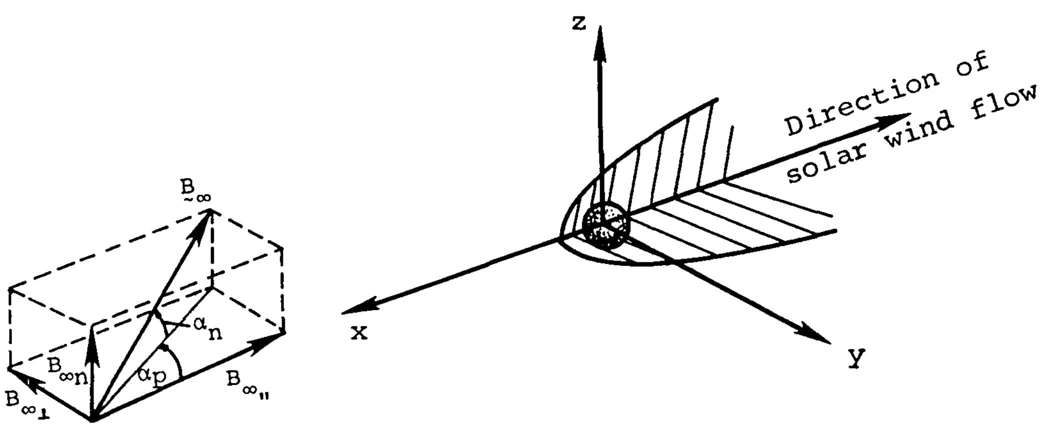
$$T/T_\infty = 1 + [(\gamma-1)/2]\cdot M_\infty^2\cdot[1 - (|\underset{\sim}{v}|/v_\infty)^2]$$

$$P/P_\infty = (\rho/\rho_\infty)(T/T_\infty)$$

Streamline coordinates may also be printed by subroutine STOUT, with LPRST as print control variable. A plot of the streamlines is generated if the variable LPLOT is true. A flow chart of the streamline calculation is shown in figure A.1(c).

A.2.1.4 Magnetic-field calculation

The magnetic field is determined by separately calculating the unit components whose directions are parallel, perpendicular, and normal to the flow, in the undisturbed solar wind. These components are then added vectorially, the resultant being expressed in orthogonal (x,y,z) components. The angles in the free stream α_p and α_n between the magnetic field and the flow, as shown in the sketch below, are either input or, in the case of a trajectory calculation, are calculated internally from the input interplanetary magnetic field.



The magnetic-field components are calculated using the following formulae in which $\underset{\sim}{e}$ signifies a vector of magnitude e in the direction of the component field line, and $\hat{n}$ the unit normal vector.

$$\left(\frac{\underset{\sim}{B}}{B_\infty}\right)_{\shortparallel} = \left(\frac{\underset{\sim}{V}}{V_\infty}\right)\left(\frac{\rho}{\rho_\infty}\right); \quad \left(\frac{\underset{\sim}{B}}{B_\infty}\right)_{\perp} = \left(\frac{\Delta\underset{\sim}{\ell}}{\Delta\ell_\infty}\right)\left(\frac{\rho}{\rho_\infty}\right); \left(\frac{B}{B_\infty}\right)_n = \left(\frac{R}{R_\infty}\right)\left(\frac{\rho}{\rho_\infty}\right)$$

$$\left(\frac{\underset{\sim}{B}}{B_\infty}\right) = \left(\frac{\underset{\sim}{B}}{B_\infty}\right)_{\shortparallel}\left(\frac{B_{\infty_{\shortparallel}}}{B_\infty}\right) + \left(\frac{\underset{\sim}{B}}{B_\infty}\right)_{\perp}\left(\frac{B_{\infty_{\perp}}}{B_\infty}\right) + \hat{n}\left(\frac{B}{B_\infty}\right)_n\left(\frac{B_{\infty_n}}{B_\infty}\right)$$

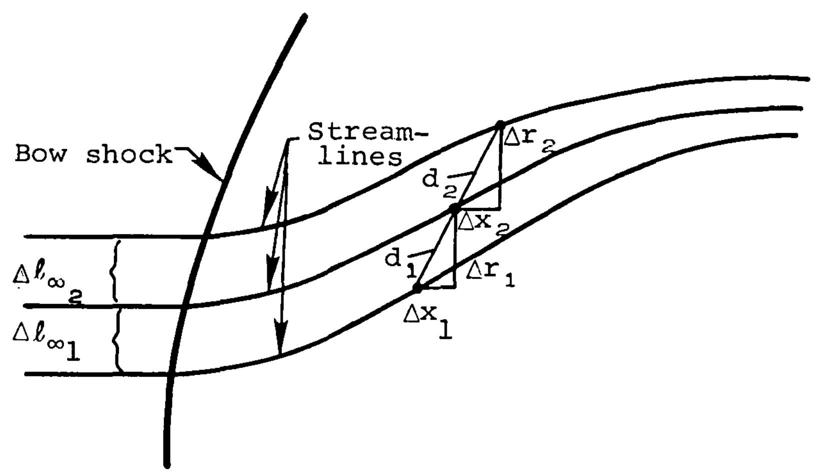
The magnetic-field line vector component $\underset{\sim}{B}_{\shortparallel}$ which results from the interplanetary component $B_{\infty_{\shortparallel}}$ that is parallel to the undisturbed solar flow has local magnitude given by $(|\underset{\sim}{v}|/v_\infty)(\rho/\rho_\infty)$, and the same local direction ϕ as the fluid flow. Determination of the normal magnetic-field component B_n requires calculation of R/R_∞, where R_∞ is the free-stream cylindrical R-ordinate of the streamline through the point under consideration. This is calculated by linearly interpolating in the local radial cylindrical coordinate R between the streamlines, with $R/R_\infty = 1.0$ along the x-axis. The magnetic-field vector component $\underset{\sim}{B}_{\perp}$ resulting from the interplanetary component $B_{\infty\perp}$ which is perpendicular to the undisturbed solar-wind flow requires the distance vector $\Delta\underset{\sim}{\ell}/\Delta\ell_\infty$, whose magnitude is $|\Delta\underset{\sim}{\ell}|/\Delta\ell_\infty$ and direction is ψ, where $|\Delta\underset{\sim}{\ell}|/\Delta\ell_\infty$ is the stretching factor of the perpendicular field at the point, and ψ is the direction of the field line through the point. The magnitude and direction of $\Delta\underset{\sim}{\ell}/\Delta\ell_\infty$ are calculated according to

$$\frac{|\Delta\underset{\sim}{\ell}|}{\Delta\ell_\infty} = \frac{d_1 \cdot d_2}{d_1 + d_2} \cdot \frac{1}{\Delta\ell_{\infty_1} + \Delta\ell_{\infty_2}}$$

and

$$\psi = \frac{\tan^{-1}\left(\frac{\Delta r_1}{\Delta x_1}\right) \cdot d_2 + \tan^{-1}\left(\frac{\Delta r_2}{\Delta x_2}\right) \cdot d_1}{(d_1 + d_2)}$$

where the quantities d_1, d_2, Δx_1, Δx_2, Δr_1, Δr_2, $\Delta\ell_{\infty 1}$, and $\Delta\ell_{\infty 2}$ are described by the sketch below. The points marked (•) on the streamlines represent equal-time intervals in the flow.



The perpendicular field lines are determined by integrating $\int \underset{\sim}{v} dt$ along each streamline, using trapezoidal integration to locate points along the streamline at regular increments in time, Δt, starting at a perpendicular field line ahead of the bow shock. Values for $|\Delta\underset{\sim}{\ell}|/\Delta\ell_\infty$ and ψ are calculated at the points where the perpendicular field lines and streamlines intersect, interpolating only along the field lines. A generalized quadrilateral interpolation scheme is then employed to determine $|\Delta\underset{\sim}{\ell}|/\Delta\ell_\infty$ and ψ at the computational grid points, using the quadrilateral containing the point formed by the intersection of pairs of adjacent streamlines and perpendicular field lines. At the bow shock, an exact formula is used, viz.

$$(|\Delta\underset{\sim}{\ell}|/\Delta\ell_\infty)^2 = 1 + \cot^2\theta(1+D^2) - 2D \times \csc\theta \times \cot\theta \times \cos(\theta-\delta)$$

$$\psi = \theta + \sin^{-1}[D \times \cot\theta \times \sin(\theta-\delta)/(|\Delta\underset{\sim}{\ell}|/\Delta\ell_\infty)]$$

where

$$D^2 = 1 - 4(M_\infty^2 \sin^2\theta - 1)(\gamma M_\infty^2 \sin^2\theta + 1)/[(\gamma+1)^2 M_\infty^4 \sin^2\theta]$$

$$\cot\delta = \tan\theta \times \{(\gamma+1)M_\infty^2/[2(M_\infty^2 \sin^2\theta - 1)] - 1\}$$

$$\theta = \tan^{-1}\left(\frac{dR_{shock}}{dx}\right)$$

The values of $|\Delta \underset{\sim}{\ell}|/\Delta\ell_\infty$ at the grid points are smoothed using fifth-order least-squares fit with respect to arc length along the arcs of th grid. The resultant magnetic field can then be expressed in orthogonal (x,y,z) components. The code determines these components for the case when the field point is located in the (x,y) plane, i.e., z = 0. These components are given by

$$B_x/B_\infty = \cos\alpha_n \times [\cos\phi \times \cos\alpha_p \times (|\underset{\sim}{B}|/B_\infty)_{\parallel} + \cos\psi \times \sin\alpha_p \times (|\underset{\sim}{B}|/B_\infty)_{\perp}]$$

$$B_y/B_\infty = \cos\alpha_n \times [\sin\phi \times \cos\alpha_p \times (|\underset{\sim}{B}|/B_\infty)_{\parallel} + \sin\psi \times \sin\alpha_p \times (|\underset{\sim}{B}|/B_\infty)_{\perp}]$$

$$B_z/B_\infty = \sin\alpha_n \times (B/B_\infty)_n$$

Magnetic-field components may now be printed by subroutine BOUT, with LPRB as print control parameter. The magnetic field is not calculated when LPRB = .FALSE. and KBCON=0. A flow chart of the magnetic-field calculation is shown in figure A.1(d).

A.2.1.5 Contour calculation and plot generation

Contours are calculated for velocity $|\underset{\sim}{v}|/v_\infty$, density ρ/ρ_∞, and magnetic components $(|\underset{\sim}{B}|/B_\infty)_{\parallel}$, $(|\underset{\sim}{B}|/B_\infty)_{\perp}$, and $(|\underset{\sim}{B}|/B_\infty)_n$. The method used is a modified version of a procedure developed by R. Sorenson of NASA/Ames Research Center. The boundary is searched for intervals which bracket a contour point. Having found one point, the remainder of the contour is determined by 'walking' around the contour, searching at each step for the interval through which the contour line next passes, until a boundary point is reached. Then closed contours are found in a similar manner. Linear interpolation is used throughout the process. Note that since T/T_∞ is a function of $|\underset{\sim}{v}|/v_\infty$ only, velocity contours may also be considered as temperature contours. Temperature and velocity are related by the following function.

$$T/T_\infty = 1 + \frac{\gamma-1}{2} M_\infty^2 \left[1 - \left(\frac{|\underset{\sim}{v}|}{v_\infty}\right)^2\right]$$

The coordinates of the contour lines can be printed by subroutine CONOUT, with LPRCON as print control parameter.

The program segment which controls the generation of contour plots is accessed only when LPLOT = .TRUE. The UCC Plot Routines used to produce these plots are AXIS, CHAR, DASH, DOTLN, ENPLT, GREEK, MATH, NUMPLT, PLOT, PLTLN, POLAR, RESET, SCALF, and VECTOR. A flow chart of the contour calculation and plot generation in figure A.1(e).

A.2.1.6 Trajectory calculation

This segment of the program provides theoretical plasma and magnetic-field properties in an output form that is useful for direct comparison with actual spacecraft data. Given a sequence of coordinates describing the spacecraft trajectory, the program calculates the density, temperature, and velocity and magnetic-field components at each point. Generation of trajectory plots is controlled by the logical variable LPLTRJ. The trajectory calculation proceeds as follows.

Input to this calculation includes interplanetary values of temperature, density, velocity, and magnetic field together with the trajectory coordinates. The trajectory input is required as a function of time and normalized by planetary radius. If the logical variable LSUN is TRUE, then it is assumed that the trajectory coordinates and vector quantities are expressed in terms of a sun-planet (ecliptic) coordinate system. In this case, these quantities are converted by the program into a solar-wind coordinate system by the transformation

$$\begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} = \begin{pmatrix} \cos \Omega \cos \phi_p & -\sin \Omega \cos \phi_p & \sin \phi_p \\ \sin \Omega & \cos \Omega & 0 \\ -\cos \Omega \sin \phi_p & \sin \Omega \sin \phi_p & \cos \phi_p \end{pmatrix} \begin{pmatrix} x_s \\ y_s \\ z_s \end{pmatrix}$$

where (x_w, y_w, z_w) are coordinates in the solar-wind system and (x_s, y_s, z_s) are coordinates in the sun-planet system. The angles Ω and ϕ_p are the azimuthal (total aberration) and polar angles, respectively. The azimuthal angle, Ω, is the angle in the plane of the ecliptic between the sun-planet line and the oncoming solar-wind, i.e., the x_s-axis and the x_w-axis as shown in figure A.2. The angle ϕ_p, positive for southward solar-wind flow, measures the deviation of the solar-wind from the plane of the ecliptic. Figure A.2 illustrates the transformation from sun-planet ecliptic coordinates to solar-wind coordinates. In this case the azimuthal and polar angles indicated are both positive.

If LSUN is FALSE, it is assumed that all input data are referenced to the solar-wind coordinate system and this transformation is not performed.

In order to conform with the internal flow-field and magnetic-field calculations, the signs of the x and y components of the trajectory and vector quantities are reversed. This is, in effect, another coordinate transformation which is defined by

$$\begin{pmatrix} x_c \\ y_c \\ z_c \end{pmatrix} = \begin{pmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 1 & 0 & 1 \end{pmatrix} \begin{pmatrix} x_w \\ y_w \\ z_w \end{pmatrix} = \begin{pmatrix} -\cos\Omega\cos\phi_p & -\sin\Omega\cos\phi_p & \sin\phi_p \\ \sin\Omega & -\cos\Omega & 0 \\ -\cos\Omega\sin\phi_p & \sin\Omega\sin\phi_p & \cos\phi_p \end{pmatrix} \begin{pmatrix} x_s \\ y_s \\ z_s \end{pmatrix}$$

where (x_c, y_c, z_c) are coordinates referenced to the internal calculation system. This transformation is illustrated in figure A.3. Also shown in figure A.3 is the relationship of the interplanetary parallel, perpendicular, and normal magnetic field components to the internal calculation system. Specifically,

$$B_{\infty_{\parallel}} = B_{x_c}, \; B_{\infty_{\perp}} = B_{y_c}, \text{ and } B_{\infty_n} = B_{z_c}$$

The angles α_p and α_n are now calculated from the relationships

$$\alpha_p = \tan^{-1}\left(\frac{B_{y_c}}{B_{x_c}}\right)$$

and

$$\alpha_n = \tan^{-1}\left(\frac{B_{y_c}}{\sqrt{(B_{x_c})^2 + (B_{y_c})^2}}\right)$$

At this point, all data is in a form compatible with the internal calculations and the program can interpolate for flow and magnetic-field values along the trajectory. The following procedure is repeated at each trajectory point. Noting that the flow is axisymmetric, the coordinate system may be rotated to the most convenient orientation for the calculation. The present (x_c, y_c, z_c) coordinates are converted to (x_c, R) coordinates by a rotation in the (y_c, z_c) plane about the x_c-axis through the angle $\theta = \tan^{-1}[z_c/y_c]$. This rotation defines a new coordinate system (x', y', z') in which $z' = 0$. Subroutine IJRAJ now locates the point with reference to the computational flow-field grid. The point is either within the ionopause, in the grid region, or beyond the bow shock. If the point is within the ionopause, all values are set to zero. If the point lies beyond the bow shock, all quantities assume their free-stream values. For points within the grid, the velocity magnitude, density, and flow angle ϕ are found by interpolation using function FTRAJ. From the flow angle ϕ and the rotation angle θ, velocity components in the (x_c, y_c, z_c) system can be calculated according to

$$v_{x_c} = v\cos\phi$$

$$v_{y_c} = v\sin\phi\cos\theta$$

$$v_{z_c} = v\sin\phi\sin\theta$$

Calculation of the magnetic field is complicated somewhat because the components are dependent on the incident magnetic field. Using α_p and α_n, B'_{x_∞}, B'_{y_∞}, and B'_{z_∞} are calculated in the rotated (x', y', z') system by

$$B'_{x_\infty} = B_{x_{c_\infty}}$$

$$B'_{y_\infty} = B_{y_{c_\infty}}\cos\theta + B_{z_{c_\infty}}\sin\theta$$

$$B'_{z_\infty} = -B_{y_{c_\infty}}\sin\theta + B_{z_{c_\infty}}\cos\theta$$

Then α'_p and α'_n are defined by

$$\alpha'_p = \tan^{-1}\left(\frac{B'_{y_\infty}}{B'_{x_\infty}}\right) \text{ and } \alpha'_n = \tan^{-1}\left[\frac{B'_{z_\infty}}{\sqrt{(B'_{x_\infty})^2 + (B'_{y_\infty})^2}}\right]$$

Interpolation is then carried out to determine the magnetic angle ψ and the ratios $\left|\frac{\tilde{B}}{B_\infty}\right|_{\parallel}$, $\left|\frac{\tilde{B}}{B_\infty}\right|_{\perp}$, and $\left(\frac{B}{B_\infty}\right)_n$ in the rotated system again using the function FTRAJ. Next, the magnetic-field components B_x', B_y', B_z' in the rotated system are calculated from

$$B_x' = \cos\alpha_n' \left(\cos\phi \cdot \cos\alpha_p' \cdot \left|\frac{\tilde{B}}{B_\infty}\right|_{\parallel} + \cos\psi \cdot \sin\alpha_p' \cdot \left|\frac{\tilde{B}}{B_\infty}\right|_{\perp}\right) \cdot B_\infty$$

$$B_y' = \cos\alpha_n' \left(\sin\phi \cdot \cos\alpha_p' \cdot \left|\frac{\tilde{B}}{B_\infty}\right|_{\parallel} + \sin\psi \cdot \sin\alpha_p' \cdot \left|\frac{\tilde{B}}{B_\infty}\right|_{\perp}\right) \cdot B_\infty$$

$$B_z' = \sin\alpha_n' \cdot \left(\frac{B}{B_\infty}\right)_n \cdot B_\infty$$

Finally, these magnetic-field components are rotated back through the angle θ to yield magnetic-field components referenced to the internal calculation system (x_c, y_c, z_c) by

$$\begin{pmatrix} B_{x_c} \\ B_{y_c} \\ B_{z_c} \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & -\sin\theta \\ 0 & \sin\theta & \cos\theta \end{pmatrix} \begin{pmatrix} B_x' \\ B_y' \\ B_z' \end{pmatrix}$$

Subroutine TROUT now prints the trajectory output in both the solar-wind (x_c, y_c, z_c) and the sun-planet (x_s, y_s, z_s) coordinate systems using the transformation below to obtain sun-planet magnetic-field vector components from solar-wind magnetic-field vector components.

$$\begin{pmatrix} B_{x_s} \\ B_{y_s} \\ B_{z_s} \end{pmatrix} = \begin{pmatrix} -\cos\Omega\cos\phi_p & \sin\Omega & -\cos\Omega\sin\phi_p \\ \sin\Omega\cos\phi_p & -\cos\Omega & \sin\Omega\sin\phi_p \\ \sin\phi_p & 0 & \cos\phi_p \end{pmatrix} \begin{pmatrix} B_{x_c} \\ B_{y_c} \\ B_{z_c} \end{pmatrix}$$

The transformation of the solar-wind velocity components (v_{x_C}, v_{y_C}, v_{z_C}) into sun-planet components (v_{x_S},v_{y_S},v_{z_S}) is also done using the same transformation.

Finally, if LPLTRJ is true, a file of trajectory plots is created. A flow chart for this program segment is shown in figure A.1(f).

A.2.2 Rerun Option

The rerun option is used when LRERUN = .TRUE. The blunt-body and marching calculations are replaced with the reading of grid coordinates and flow-field values from the rerun file, TAPE4, which contains data written to TAPE9, then saved, on a previous run. Different values for any parameter not used in the flow-field calculations may be specified, viz. contour values, plot length, magnetic-field angles, and output options. Values of AMACH, GAMMA, and HRO are required input, to ensure that the input rerun file does contain the case desired for rerun. If the geometry is user-supplied, the body-shape table will be read from TAPE4, and should not be input from cards.

After reading the card input, MACH, GAMMA, and HRO are tested against values from TAPE4. The grid coordinates and flow-field values from the blunt-body calculation are read in, then smoothed, and streamlines calculated for this region, as previously described. The results of the marching calculation are then read, and the streamline calculation continued downstream. The calculations then proceed as described in section A.2.1.

A run must not contain more than one case which uses the rerun option.

A.2.3 Program Limitations and Precautions

The program makes some assumptions about the geometry of the obstacle shape around which flow is to be calculated, and about the flow field. The obstacle shape is assumed to be monotonically increasing in cylindrical radius R, going downstream. The nose of the obstacle is at $x = 1.0$. The origin of the (x,R) coordinate system is the center of the planet. Obstacle shapes with sharp corners should be avoided. In the magnetic-field calculation, the first streamline is assumed to be inside the arc described by the grid points immediately off the body, downstream of $x = 0.0$. To reduce computational costs, a grid using NR = 10 may be used, in which case a lower value of CN may be required. This would reduce the running time by approximately 40 percent. A free-stream Mach number less than 2.0 is not advised.

A.2.4 Convergence Criteria for Blunt-Body Calculation

The output provides two measures of the convergence of the blunt-body calculation. The RMS of shock speed and maximum shock speed are printed at each iteration. These quantities should both tend to zero as the iterations proceed. A value for q_{RMS}, RMS of shock speed, of

$$q_{RMS} < \sqrt{\gamma} \times M_\infty \times 10^{-3}$$

where γ is the specific heat ratio, and M_∞ is the free-stream Mach number, usually indicates a converged solution. The RMS of error in enthalpy, HT, should be less than 1 percent, with the maximum enthalpy error also of that order.

The Courant number, CN, determines the time step size used by the calculation. A value not greater than the default of 3.0 should be used. For low Mach numbers or a coarser mesh than the default grid, a lower value may be preferable. If the default value does not generate a converged solution, or if the error message from subroutine SHOCK is printed, try lowering CN in increments of 0.5 to find a better value of CN. User-supplied bodies may also require a lower Courant number.

A.3 DESCRIPTION OF INPUT

This section describes the card input for the program. An alphabetized dictionary of input variables is provided, defining the varibles, listing default values and limitations. A discussion of the preparation of the card input is then presented, followed by a description of the input card format.

A.3.1 Dictionary of Input Variables

AMACH free-stream Mach number; $3.0 \leq \text{AMACH} \leq 25.0$ is recommended

ANGN the angle, in degrees, measuring the deviation of the free-stream magnetic field from the plane in which $\underset{\sim}{B}_{\infty_{\parallel}}$ and $\underset{\sim}{B}_{\infty_{\perp}}$ lie; equal to $\tan^{-1}\left(B_{\infty_n} / \sqrt{|B^2_{\infty_{\parallel}}| + |B^2_{\infty_{\perp}}|}\right)$; see figure A.3, measured in the (x_c, y_c, z_c) coordinate system; only specified when interplanetary magnetic-field components not specified.

ANGP the angle, in degrees, measuring the deviation of the in-plane magnetic component $(\underset{\sim}{B}_{\infty_{\parallel}} + \underset{\sim}{B}_{\infty_{\perp}})$ from the direction of flow; equal to $\tan^{-1}(B_{\infty_{\parallel}}/B_{\infty_{\perp}})$; see figure A.3, measured in the (x_c, y_c, z_c) coordinate system; only specified when interplanetary magnetic-field components not specified.

AZANG angle in the ecliptic plane between the sun-planet line and the direction of solar-wind flow. See figure A.2 for positive direction.

BCON(I) KBCON-dimensional array specifying values to be used for magnetic field strength contours

BINF magnetic field strength free-stream value; set to 1.0 if plots desired in nondimensionalized units.

BX1 x_s-component of interplanetary magnetic field; referred to sun-planet coordinates

BY1 y_s-component of interplanetary magnetic field; referred to sun-planet coordinates

BZ1 z_s-component of interplanetary magnetic field; referred to sun-planet coordinates

CN Courant number used for blunt-body calculation; program default value is 3.0

GAMMA ratio of plasma specific heats

HRO obstacle geometry indicator:

HRO > 0. - ionopause with H/R_o = HRO
HRO = 0. - magnetopause equatorial trace
HRO < 0. - geometry is user-supplied

ITER integer, number of iterations for blunt-body calculation; program default value is 300

KBCON integer, number of values specified for magnetic-field contours; $0 \leq$ KBCON ≤ 20

KRCON integer, number of values specified for density contours; $0 \leq$ KRCON ≤ 20

KVCON integer, number of values specified for velocity magnitude contours; $0 \leq$ KVCON ≤ 20

LGRAV logical variable indicating whether default ionopause is calculated with gravitational variation in scale height

FALSE - no
TRUE - yes

LPLOT logical variable indicating whether to create plots or plot file

FALSE - no

TRUE - yes

LPLTRJ logical variable indicating whether to create trajectory and time history plots

FALSE - no

TRUE - yes

LPRB logical variable indicating whether to print magnetic field output

FALSE - no

TRUE - yes

LPRCON logical variable indicating whether to print coordinates of contours lines

FALSE - no

TRUE - yes

LPRFL logical variable indicating whether to print detailed flow-field output

FALSE - no

TRUE - yes

LPRST logical variable indicating whether to print coordinates of streamlines

FALSE - no

TRUE - yes

LRERUN logical variable indicating whether this case uses rerun option

FALSE - perform blunt-body and marching calculations

TRUE - read results of a previous calculation from TAPE4

LRSTRT logical variable indicating whether to use previous shock shape as initial guess for blunt body

TRUE - use shock shape from previous solution. (Must have a full solution as an earlier run in same job.)

FALSE - use default initial guess for shock shape

LSUN logical variable indicating whether trajectory input is referenced to sun-planet coordinate system

FALSE - trajectory input in solar-wind coordinates

TRUE - trajectory input in sun-planet coordinates

LTRAJ logical variable indicating whether to perform a trajectory calculation

TRUE - trajectory calculation, data provided

FALSE - no trajectory calculation

MARKT(I) NMARKT - dimensional array specifying points to be marked for cross reference. If K = NMARKT(I), the Kth point of the trajectory is to be marked.

NBLUNT integer, number of angular mesh points for blunt-body calculation; for user-supplied geometry, XX(NBLUNT-1)=0.0; program default value, and maximum, is 24

NBOD integer, number of points in body-shape table when geometry is user-supplied; $1 \leq$ NBOD ≤ 100

NCASE integer, number of cases to be run consecutively; NCASE > 1

NMARKT integer, numbered values specified for cross reference points; $0 \leq$ NMARKT ≤ 12.

NR integer, number of radial mesh points; program default value, and maximum, is 19

NTRAJ integer, number of points specified in trajectory table

NXADD integer, number of points to be added to blunt-body grid past $\theta = 90°$, default value is 0.

POLANG angle, measured in degrees, between the plane of the ecliptic and direction of solar-wind flow; positive for southward flow; see figure A.2

RCON(I) KRCON - dimensional array specifying values to be used for density contours

RHOINF density-free stream value; set to 1.0 if plots desired in nondimensional units

RPLNT radius of planet in units of nose radius, R_{PLNT}/R_o

RR(I) NBOD - dimensional array representing the R-locations, in cylindrical (x,R) coordinates, of the user-supplied body shape; in units of nose radius

TITLE descriptive heading of the case, to be printed on the first page of output; may contain up to 80 characters, including blanks

TMPINF free-stream temperature; set to 1.0 if plots desired in nondimensional units

TTRAJ(I) NTRAJ - dimensioned array specifying time locations of trajectory points

VCON(I) KVCON - dimensional array specifying values to be used for velocity contours

VINF free-stream velocity; set to 1.0 if plots desired in non-dimensional units

XCALC terminal downstream x-location for marching calculation of flow field; XCALC < 0.0; program default value is -1.0

XPLOT terminal downstream x-location for calculation of streamlines, magnetic field, and contours; XCALC $\leq$ XPLOT $\leq$ 0.0; program default value is -1.0

XTRAJ(I) NTRAJ - dimensioned array specifying x_s-locations of trajectory points; in units of planetary radius; when (ANGP,ANGN) are specified, XTRAJ(I) is referred to solar-wind x_c-locations; see figures A.2 and A.3

XX(I) NBOD - dimensional array representing the x-locations, in cylindrical (x,R) coordinates, of the user-supplied body shape; in units of nose radius. See figures A.2 and A.3

YTRAJ(I) NTRAJ - dimensioned array specifying y_s-locations of trajectory points; in units of planetary radius; when (ANGP,ANGN) are specified, YTRAJ(I) is referred to solar-wind y_c-locations; see figures A.2 and A.3

ZTRAJ(I) NTRAJ - dimensioned array specifying z_s-locations of trajectory points; in units of planetary radius; when (ANGP,ANGN) are specified, ZTRAJ(I) is referred to solar-wind z_c-locations; see figures A.2 and A.3

A.3.2 Preparation of Input Data

The card input for a run consists of one card containing the number of cases to be run consecutively, Item 0, followed by a set of input for each case, Item 1 through Item 7, and Item 8 if required. Where a default value is to be used, the input field should be left blank.

For each case, all required variables which do not assume their default values should be specified. The input format for all cards is described in section A.3.3.

Item 0 - This item consists of one card, containing the number of cases in this run, NCASE.

Item 1 - This card provides identification of the case, TITLE, which is printed on the first page of the output for this case.

Item 2 - This card contains information on the flow conditions and body geometry, and parameters required for the blunt-body and marching calculations. AMACH, GAMMA, and HRO must be specified for each case. For the rerun option, the values are tested against the values from the rerun file. The parameters XCALC, NR, NBLUNT, CN, ITER are used only when the flow field is to be calculated. These variables each assume a default value if the input field is blank.

Item 3 - This item consists of one card containing the rerun indicator, LRERUN, the output control variables LPRFL, LPRST, LPRCON, LPRB, and LPLOT, the trajectory indicator LTRAJ, and the restart indicator LRSTRT.

Item 4 - This card contains the variables XPLOT, ANGP, ANGN, NXADD, and LGRAV. The value for XPLOT is changed by the program to be the x-location of the marching calculation immediately upstream of the input value for XPLOT. The angles describing the deviation of the magnetic field from the flow, ANGP and ANGN, are not required when LPRB = .FALSE; KBCON = 0, and LTRAJ = .FALSE. since the magnetic field is not calculated under these conditions. ANGP is the angle between the vectors $(\underset{\sim}{B}_{\infty_{\parallel}} + \underset{\sim}{B}_{\infty_{\perp}})$ and $\underset{\sim}{v}_{\infty}$, while ANGN is the angle between $\underset{\sim}{B}_{\infty}$ and $(\underset{\sim}{B}_{\infty_{\parallel}} + \underset{\sim}{B}_{\infty_{\perp}})$, where $\underset{\sim}{B}_{\infty_{\parallel}}$, $\underset{\sim}{B}_{\infty_{\perp}}$, $\underset{\sim}{B}_{\infty_{n}}$ are the components of the free-stream magnetic field, $\underset{\sim}{B}_{\infty}$, which are parallel, perpendicular, and normal to $\underset{\sim}{v}_{\infty}$, and are as indicated in figure A.3. The two angles ANGP and ANGN fully determine the half plane for which the magnetic field is to be calculated. The magnetic field for the other half of the plane may be calculated by rerunning with the sign of ANGP reversed. When $(\underset{\sim}{B}_{\infty_{\parallel}} + \underset{\sim}{B}_{\infty_{\perp}}) = 0$, ANGN = $\pm 90^{\circ}$, ANGP = 0°; and, when $\underset{\sim}{B}_{\infty_n} = 0$, ANGN = 0°. Note that ANGP and ANGN are referenced to the (x_c, y_c, z_c) system and are specified only when the interplanetary magnetic-field components are not specified.

If both LTRAJ = .TRUE. and LSUN = .TRUE., then ANGP and ANGN are calculated internally from the interplanetary magnetic-field components BX1, BY1, and BZ1.

Item 5 - This item contains the values for the velocity contours. The first card contains KVCON, the number of values specified for VCON. If KVCON > 0, the contour values are then read. Up to three cards may be required to accommodate the values, eight per card, maximum of 20. The contour values should be monotonically increasing, with at least one value within the range of the magnitude of the velocity in the region for which contours are to be calculated.

Item 6 - This item contains the values for the density contours. The description is similar to that for Item 5, with KRCON being the number of values specified, and RCON the array of values.

Item 7 - This item contains the values for the magnetic-field contours. The description is similar to that for Item 5, with KBCON being the number of values specified, and BCON the array of values. Note that the same contour values are used for the parallel and perpendicular components.

Item 8 - This optional item is required when HRO < 0.0 and LRERUN = .FALSE., and contains the body-shape table for the user-supplied geometry. The first card contains NBOD, the number of points in the shape table. The next NBOD cards contain the cylindrical (x,R) coordinates of these points, [XX(I), RR(I)], one point per card. The points supplied by the user determine the θ-spacing of the mesh used for the blunt-body calculation. The first point should be near, but not on, the x-axis. A suggested location is such that the θ-spacing between the first point and the x-axis is half the θ-spacing between the first two points. The blunt-body calculation adds a point which is the reflection about the x-axis of the first point in the body-shape table. The $(\mathrm{NBLUNT}-1)^{th}$ point should be at x = 0.0. The BLUNT^{th} point is also used to create the grid for the blunt-body calculation. The coordinates must be normalized so that the planet center is at (0.,0.) and the nose of the body at (1.,0.).

Item 9 - This optional item is read only when LTRAJ is TRUE. The first card contains NTRAJ, the number of points in the trajectory. Then follows NTRAJ cards, each containing the time T, and location (x_s, y_s,z_s) of one point. The time values should be monotonically increasing. At present, NTRAJ $\leq$ 100 is required. Note that when ANGP and ANGN are specified, the trajectory is specified in (x_c,y_c,z_c) coordinates.

Item 10 - This item is ready only when LTRAJ is TRUE. The variable LPLTRJ indicates whether plots are to be produced of the trajectory and time histories. The relative size of the planet to the ionopause is given by RPLNT, which may be 0.0, in which case, a value of 1.0 is assumed in the calculations, but the planet is not drawn on the plots. Next are the four free-stream values v_∞, T_∞, ρ_∞, B_∞. If the plots are desired to be in nondimensional units, any or all of these values may be input as 1.0. Each quantity must have a value, zero is not permissible.

Item 11 - This item is read only when LTRAJ is TRUE. The first card contains NMARKT, the number of values specified for MARKT, (presently maximum of 12). If NMARKT = 0, only this card is required. If NMARKT > 0, the values of MARKT are read, 8 per card.

Item 12 - This item, which includes the variables LSUN, AZANG, POLANG, BX1, BY1, and BZ1, is read only when LTRAJ is true.

A.3.3 Format of Input Data

Four format types are used for the input data. For real numbers (F-format), a decimal point is required. Integers (I-format) should be right-adjusted in the field. For logical variables (L-format), the first non-blank character in the field, which should be 'T' or 'F', determines the value. Note that a blank input field is interpreted as 'FALSE'. The title, which is in A-format, may contain any valid character.

A description of the card format of the input data follows, with item numbers corresponding to those in section A.3.2:

Item No. 0: 1 card

Variable	NCASE
Card Column	10
Format type	I

Item No. 1: 1 card

Variable	Title
Card Column	80
Format type	A

Item No. 2: 1 card

Variable	AMACH	GAMMA	HRO	XCALC	NR	NBLUNT	CN	ITER
Card column	10	20	30	40	50	60	70	80
Format type	F	F	F	F	F	F	F	F

Item No. 3: 1 card

Variable	LRERUN	LPRFL	LPRST	LPRCON	LPRB	LPLOT	LTRAJ	LRSTRT
Card column	10	20	30	40	50	60	70	80
Format type	L	L	L	L	L	L	L	L

Item No. 4: 1 card

Variable	XPLOT	ANGP	ANGN	NXADD	LGRAV
Card column	10	20	30	40	50
Format type	F	F	F	N	L

Item No. 5: a) 1 card

Variable	KVCON
Card column	10
Format type	I

b) 0 to 3 cards as needed for up to 20 values, 8 per card

Variable	VCON(1)	VCON(2)			VCON(KVCON)			
Card column	10	20	30	40	50	60	70	80
Format type	F	F	F	F	F	F	F	F

Item No. 6: a) 1 card

Variable	KRCON
Card column	10
Format type	I

b) 0 to 3 cards

Variable	RCON(1)	RCON(2)			RCON(KRCON)			
Card column	10	20	30	40	50	60	70	80
Format type	F	F	F	F	F	F	F	F

Item No. 7 a) 1 card

Variable	KBCON
Card column	10
Format type	I

b) 0 to 3 cards

Variable	BCON(1)	BCON(2)			BCON(KBCON)			
Card column	10	20	30	40	50	60	70	80
Format type	F	F	F	F	F	F	F	F

Item No. 8 a) 1 card (this item required only when HRO < 0.0 and LRERUN = .FLASE.)

Variable	NBOD
Card column	10
Format type	I

b) NBOD cards

XX(I)	RR(I)
10	20

Item No. 9: a) 1 card (this item read only when LTRAJ is TRUE)

Variable	NTRAJ
Card column	10
Format type	I

b) NTRAJ cards

Variable	TTRAJ(I)	XTRAJ(I)	YTRAJ(I)	ZTRAJ(I)
Card column	10	20	30	40
Format type	F	F	F	F

Item No. 10: 1 card (this item read only when LTRAJ is TRUE)

Variable	LPLTRJ	RPLNT	VINF	RHOINF	TMPINF	BINF
Card column	10	20	30	40	50	60
Format type	L	F	F	F	F	F

Item No. 11: a) 1 card (this item read only when LTRAJ is TRUE)

Variable	NMARKT
Card column	10
Format type	I

b) 0-2 cards

Variable	MARKT(1)	MARKT(2)			MARKT(NMARKT)			
Card column	10	20	30	40	50	60	70	80
Format type	I	I	I	I	I	I	I	I

Item No. 12: 1 card (this item read only when LTRAJ is TRUE)

Variable	LSUN	AZANG	POLANG	BX1	BY1	BZ1
Card column	10	20	30	40	50	60
Format type	L	F	F	F	F	F

A.4 DESCRIPTION OF OUTPUT

This section describes the output of the computer program. The contents of each output item are specified and discussed. The printed output consists of seven items, five of which are optional and are controlled with input parameters. Plotted output is also optional.

The first output item consists of a banner page and the input data. The input is presented in two forms: first, as images of the input cards, and then with identification of each variable. Default values are printed as if they were input. Parameters CN, NR, NBLUNT, ITER for the blunt-body calculation and XCALC, the terminal location for the marching calculation, are printed only when the flow field is to be calculated. When the obstacle geometry is user-supplied, the input body-shape table is printed. For a default geometry, the body shape is indicated by the description "default ionopause shape for constant scale height with H/RO = ", or "default ionopause shape with gravitational variation in scale height, H/RO = ". Trajectory input is printed only when LTRAJ is true.

The second output item is not printed when LRERUN = .TRUE. From the blunt-body calculation, the shock speed at each iteration, the final enthalpy error, final sonic-line location, and body and final bow-shock shape are printed. For the marching calculation, the downstream x-location and body and shock ordinates are output. There is no control variable allowing the user to suppress this item of output when the flow field is calculated.

Detailed flow-field output is the third item, and is printed only when LPRFL = .TRUE. Coordinates are labeled as X/D, R/D, RP/D, or X/RO, R/RO, RP/RO, to emphasize that distances are normalized by the distance from the center of the planet to the nose of the body, D for the magnetopause, RO for an ionopause. Along the symmetry axis, the values printed are velocity magnitude V/VINF, density RHO/RHOINF, temperature T/TINF, and pressure P/PINF. Over the rest of the flow field, values are also given for velocity components VX/VINF, VR/VINF, and flow angle ϕ. Note that the flow angle is the deviation of the flow about the obstacle, and so $0^{\circ} \leq \phi \leq 90^{\circ}$.

The next output item is the (x,R) coordinates of the streamlines. For blunt-body region, the (R_p,θ) coordinates of the starting position on the bow shock wave are also given. This item is printed only when LPRST = .TRUE.

The magnetic-field components are then printed, if LPPRB = .TRUE. The location of each point is defined in (R_p,θ) coordinates for the blunt-body region, and (x,R) coordinates for the downstream marching region. The components along field lines parallel, perpendicular, and normal to the flow in the free stream are printed as B/BINF(PARALLEL), B/BINF(PERP), B/BINF(NORMAL). The orthogonal (x_c,y_c,z_c) components of the resultant are printed as BX/BINF(RESULTANT), BY/BINF (RESULTANT), BZ/BINF(RESULTANT). The magnetic field in the symmetry (x_c,y_c) plane, defined by the vector sum $[(\underset{\sim}{B}/B_{\infty})_{\parallel} + (\underset{\sim}{B}/B_{\infty})_{\perp}]$, is also printed, and is given by the magnitude B/BINF(IN-PLANE) and direction B-ANGLE(IN-PLANE) of the vector. We note, as pointed out in the text, that the orthogonal magnetic-field components printed here correspond to those in the (x_c,y_c) plane, i.e., $z_c = 0$.

The next item printed is the (x_c,R) coordinates of the contours, for which LPRCON is the logical control variable. Noting that temperature and velocity contours coincide, the corresponding value of T/TINF is printed along with V/VINF for the velocity contours. There are three nonfatal error messages which may occur - see section A.5.

Trajectory output is the last item to be printed. This output is presented first in terms of the solar-wind coordinate system (x_c, y_c, z_c), and then in terms of sun-planet coordinates (x_s,y_s,z_s).

The trajectory coordinates are printed as a function of time and are shown normalized by both RO and the planetary radius. Next, flow and magnetic-field componets are printed for each trajectory point. This output is presented in both nondimensional and dimensionalized forms and includes $|\underset{\sim}{v}|$, v_x, v_y, v_z, density, temperature, $|\underset{\sim}{B}|$, B_x, B_y, and B_z.

The program also has the capability to produce two sets of plotted output using UCC plot routines AXIS, CHAR, DASH, DOTLN, ENPLT, GREEK, MATH, NUMPLT, PLOT, PLTLN, POLAR, SCALF, and VECTOR. The first set of plots is generated when LPLOT = .TRUE. and provides a pictorial representation of the streamlines and contours with a maximum of seven frames produced. The first frame is a plot of the streamlines followed by contour plots of velocity magnitude, temperature, and density. The next three frames are contour plots of the unit parallel, perpendicular, and normal magnetic-field components. These plots are referred to the solar-wind (x,R) coordinate system.

The second set of plots is produced according to the value of the logical variable LPLTRJ. This set consists of twelve plots. The first frame is a projection of the trajectory rotated onto the x-R plane. The second frame is a plot of the trajectory projected onto the y_c-z_c plane. The remaining frames are time-history plots of density, temperature, velocity, and magnetic field. The velocity plots include magnitude and three components as do the magnetic field plots. The vector components are referred to the sun-planet ecliptic (x_s, y_s, z_s) coordinates.

A.5 PROGRAM ERROR MESSAGES

This section lists the messages printed by the program, and indicates what action should be taken by the user.

(1) ***** EXECUTION TERMINATED *****
RERUN DATA ON TAPE4 DOES NOT AGREE
WITH CASE SPECIFIED ON CARD INPUT:

	MACH NO.	GAMMA	H/RO
FROM CARDS			
FROM TAPE4			

The first three parameters of item 2 of the input for a case using the rerun option should agree with those used when creating the file. The tolerance used in comparing the values is 10^{-5}. For a user-supplied geometry, it is sufficient for both values of H/R_o to be negative.

(2) ***** EXECUTION TERMINATED *****
ARRAY OF CONTOUR VALUES IMPROPERLY SPECIFIED

When specified, the contour values should be monotonically increasing with at least one value in the range of the velocity, density, or magnetic-field strength for the region under consideration. This error does not inhibit generation of the rerun file.

(3) CONTOUR SEARCH ABORTED - TABLE OVERFLOW IN NAD

The program allows for 29 contour lines to be found, storing the starting address of each contour line in array NAD. This message indicates that at least one more contour line could be found. If the user requires all the contours of the levels specified, the case should be rerun in two parts. Otherwise, reduce the number of contour levels specified.

(4) CONTOUR SEARCH ABORTED - TABLE OVERFLOW IN (X,Y)

The contour lines may be described by up to 1000 points, stored in arrays X and Y. This message indicates that more points would be required for the contour lines requested. The last contour line found will be incomplete. As with (3), either reduce the number of contour levels or run as two cases.

(5) NEGATIVE PRESSURE DETECTED BY SHOCK AT J=
PN= PO= PTAU=

This message is printed by the blunt-body code when a negative pressure has been calculated at the shock on this iteration, at radial locations J. The quantities printed are: PN, the pressure calculated on this step; PO, the pressure from the previous step; and PTAU, the partial derivative of pressure with respect to time. This condition indicates that the shock wave motion is too extreme. Lowering the value of CN, and thus reducing the time step, may remove the problem.

The following messages (6)-(10) usually result from using an obstacle geometry which is in some way too severe for the program to handle in its present form. The obstacle slope may be sufficiently high at x = 0.0 that the axial Mach number becomes subsonic in the starting solution for the marching calculation, or there may be a sharp corner in the profile. Check input, particularly free-stream Mach number and body geometry.

(6) NEGATIVE PRESSURE ON BODY DETECTED BY BNDRY, PB= AT J=

This message indicates that a negative pressure on the body, PB, has been calculated at radial location J.

(7) NEGATIVE PRESSURE OR DENSITY ON BODY DETECTED BY BNDRYM AT X=
PB= RHOB= VXB= VRB=

The program makes internal corrections when this condition occurs, resulting pressure PB, density RHOB, and velocity components VXB and VRB.

(8) NEGATIVE SIGMA-BAR-1 IN EIGENM INDICATES SUBSONIC FLOW AT I=

(9) NEGATIVE SIGMA-BAR-2 IN EIGENM INDICATES SUBSONIC FLOW AT I=

These messages are printed when subsonic flow is detected by the marching calculation. The computed stepsize for this region will be quite small.

(10) ------------BODY TURN STOPPED AT M2=100------------

This message indicates that the body has a sharp corner, which has been limited to 100° when being transformed.

A.6 SAMPLE CASE

The sample case presented in this section is based on actual interplanetary conditions as measured by the solar-wind plasma analyzer, the fluxgate magnetometer, and retarding potential plasma analyzer on the Pioneer-Venus Orbiter for orbit 3.

The sample case is run alone and is set up to produce all possible output. The gasdynamic solution is to be calculated about a default ionopause shape with $H/R_o = 0.03$, $M_\infty = 3.0$, and $\gamma = 5/3$. The value of H/R_o is based on measurements of ionospheric density and temperature by the retarding potential plasma analyzer. Streamlines, magnetic-field components, and contours are desired to a downstream location of $-5.5\ x/R_o$. Contour values are specified for all quantities. Interplanetary values for velocity magnitude and direction, density, and temperature were provided by the solar-wind plasma analyzer and for the magnetic field by the fluxgate magnetometer.

The input data is tabulated in figure A.4, with item numbers corresponding to those in sections A.3.2 and A.3.3. The first card, item 0, indicates that there is one case to run. The remaining fifty-five cards provide the data for this case. Item 1 contains the identifying title. On the next card, item 2, values are specified for AMACH, GAMMA, HRO, and XCALC. The other data fields are left blank to indicate that the default values will be used. The values of the logical variables of item 3 specify that the flow field is to be calculated and that full printed and plotted output is to be produced. Item 4 defines the plot length to be -5.5 x/R_o. The fields for ANGP and ANGN are left blank as they are to be calculated internally by the program. Items 5, 6, and 7 specify the contour levels to be used - 14 for velocity and temperature, 11 for density, and 13 for magnetic-field strength. Item 8 is omitted because the obstacle geometry is one of the default shapes for which the coordinates are calculated internally. The next 37 cards, item 9, are the trajectory coordinates, indicating time (in minutes from periapsis and the three spacial coordinates normalized by planetary radius). Item 10 indicates that trajectory plots are to be generated. This item also specifies free-stream values of velocity, density, temperature, and magnetic-field strength. The next two cards, item 11, indicates that the fourth, ninth, eleventh, and nineteenth trajectory points are to be marked on the plots for cross- reference. The last input card, item 12 indicates that the given trajectory coordinates are expressed in sun-planet coordinates. The azimuthal and polar angles, Ω and ϕ_p, are also specified by this item as are the free-stream magnetic-field components.

Figure A.5 presents portions of the printed output from this sample case. The full printed output is approximately 6,000 lines. Figure A.6 shows the 19 plots which are produced by the program for this case.

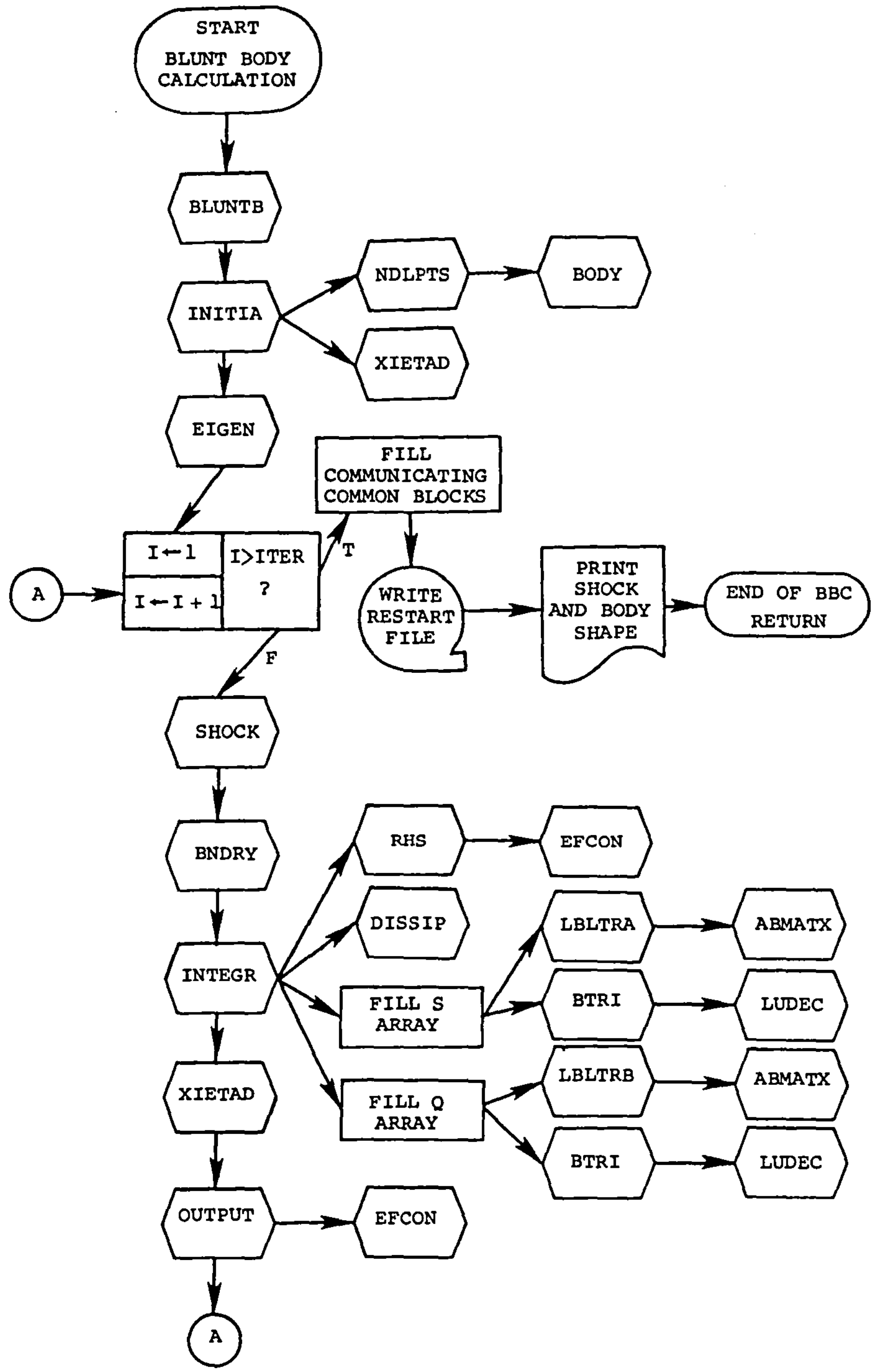


Figure A.1(a).- Flow chart for blunt-body calculation.

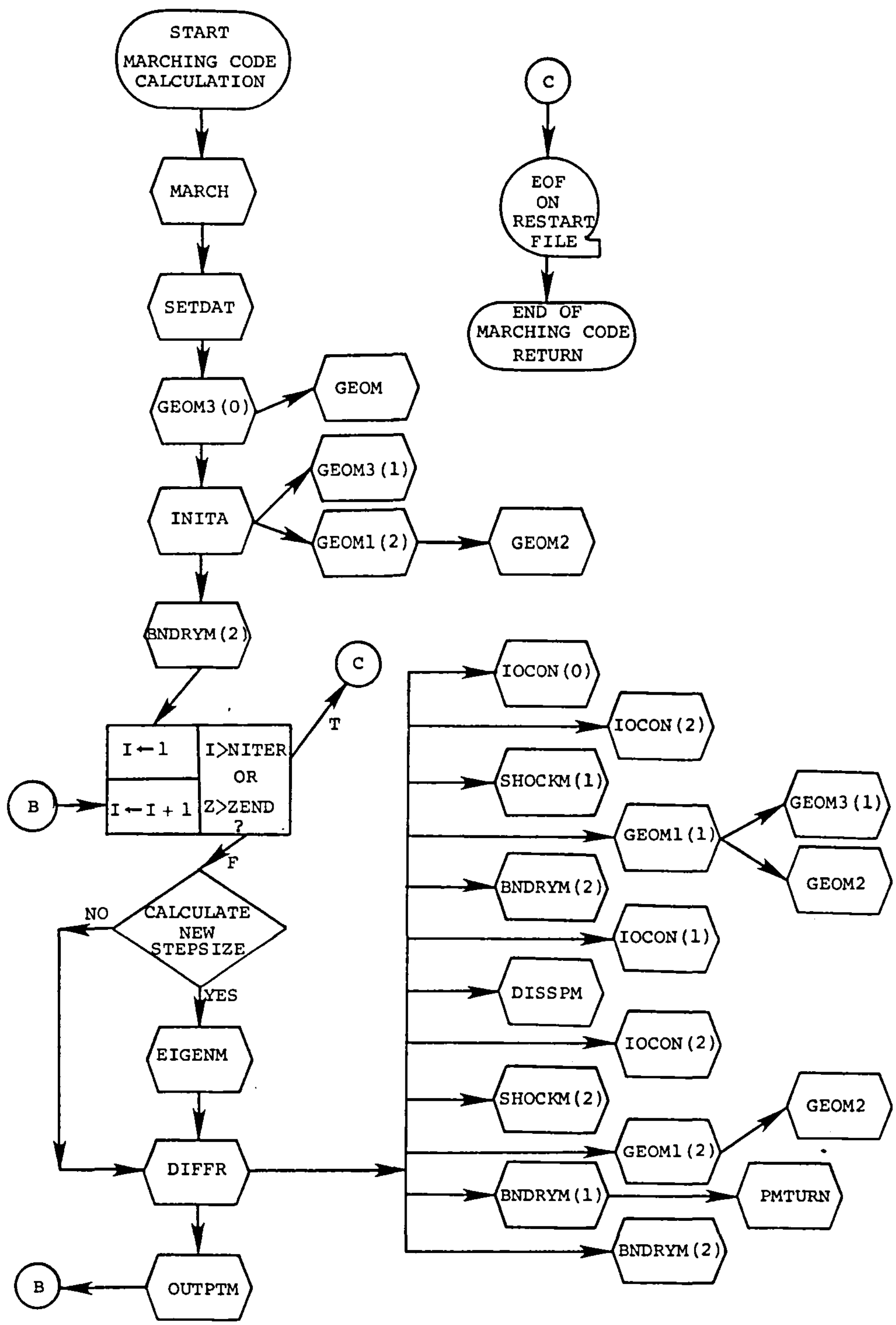


Figure A.1(b).- Flow chart for marching calculation.

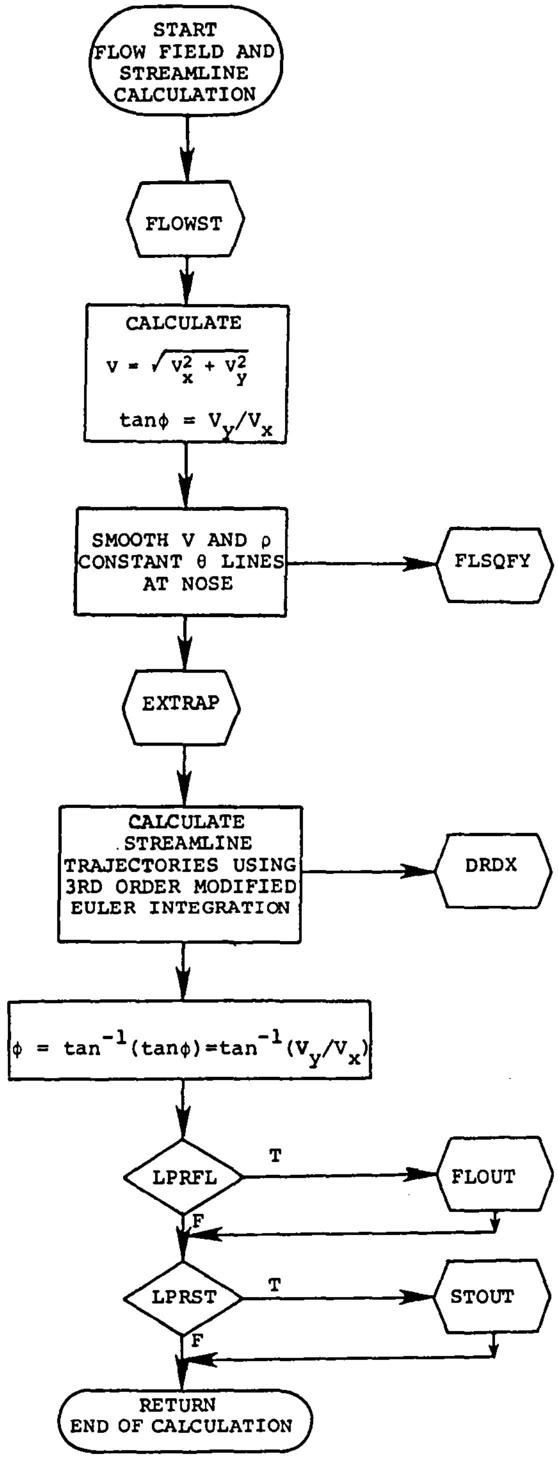


Figure A.1(c).- Flow chart of streamline calculation.

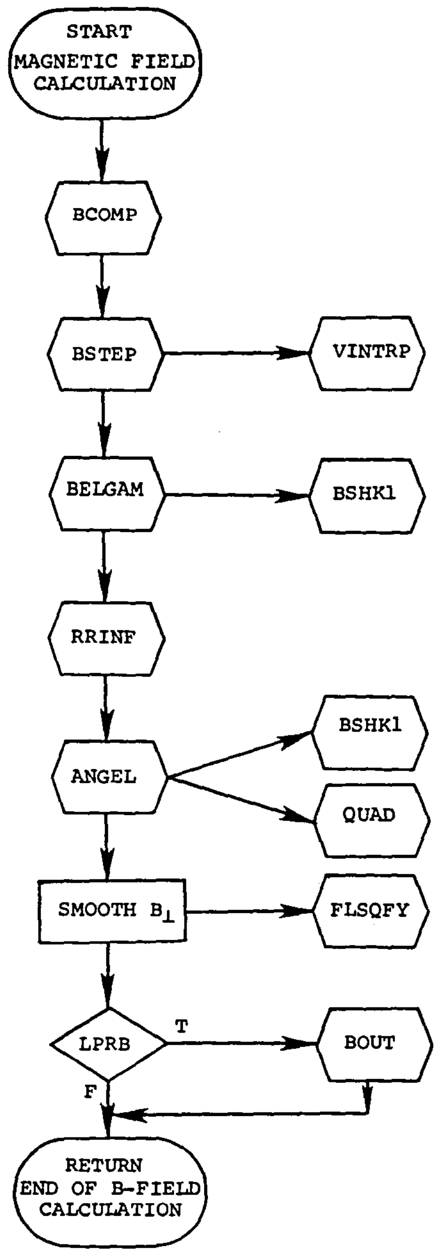


Figure A.1(d).- Flow chart of magnetic-field calculation.

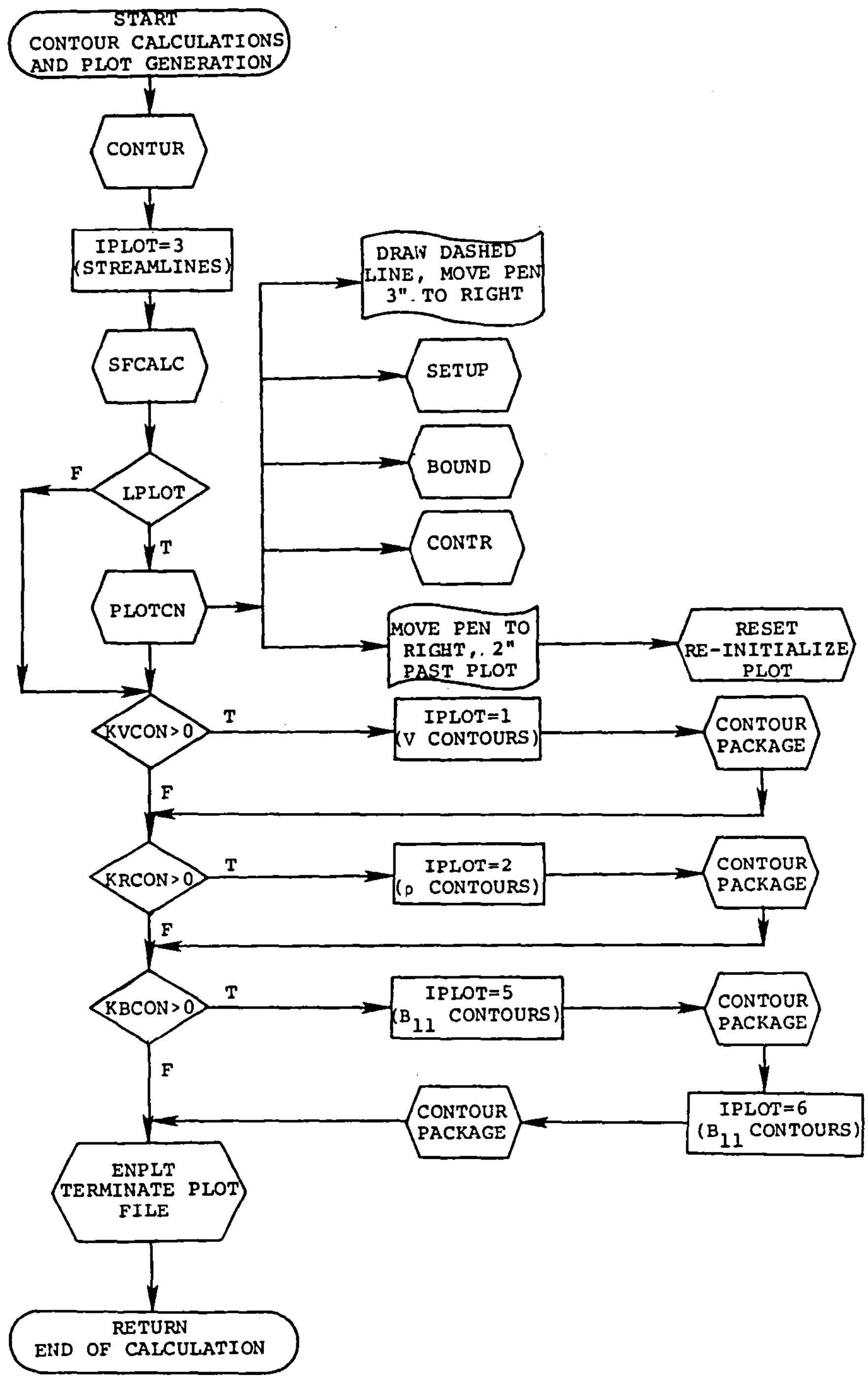


Figure A.1(e).- Flow chart of contour and plot generation calculation.

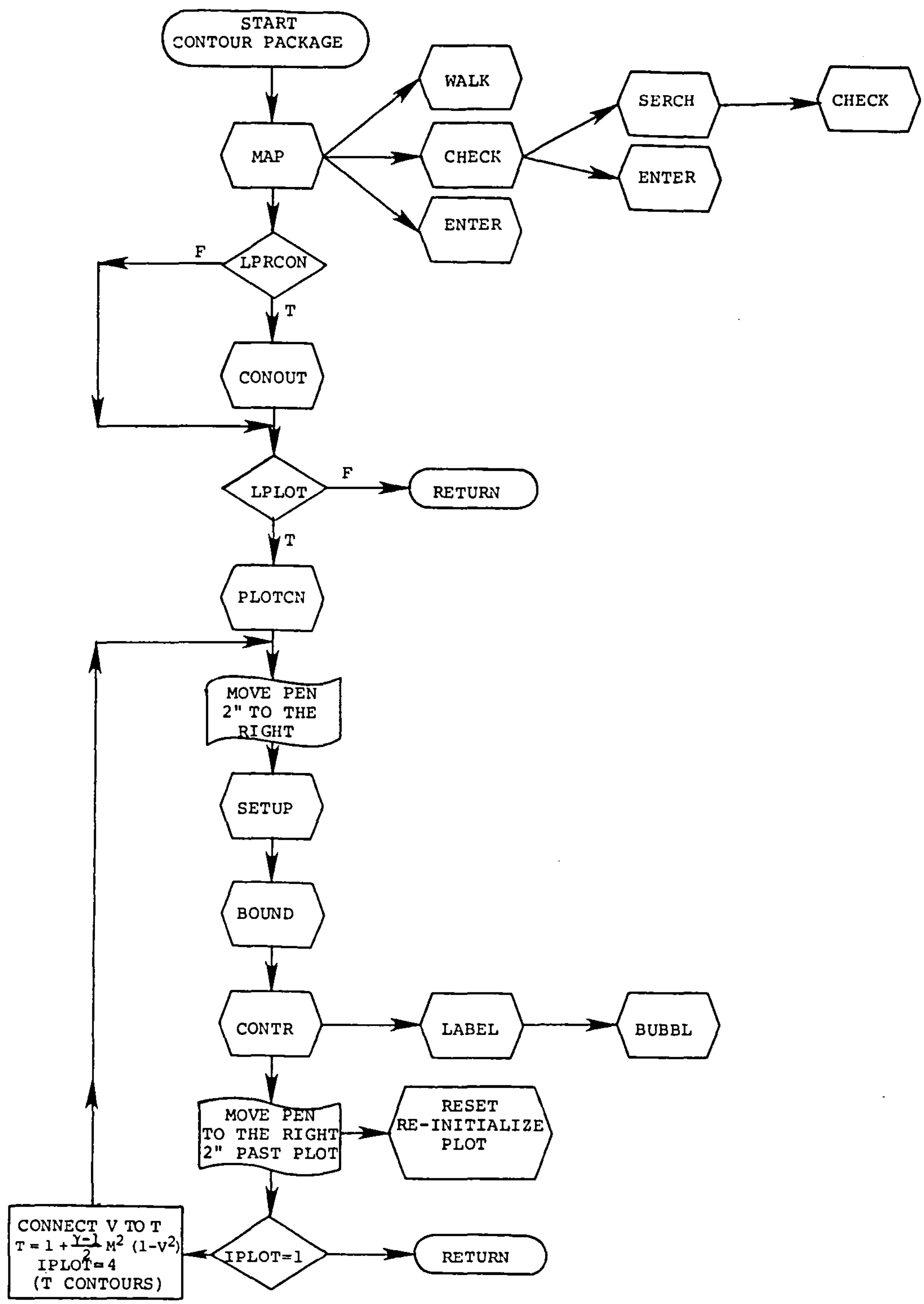


Figure A.1(e).- Concluded

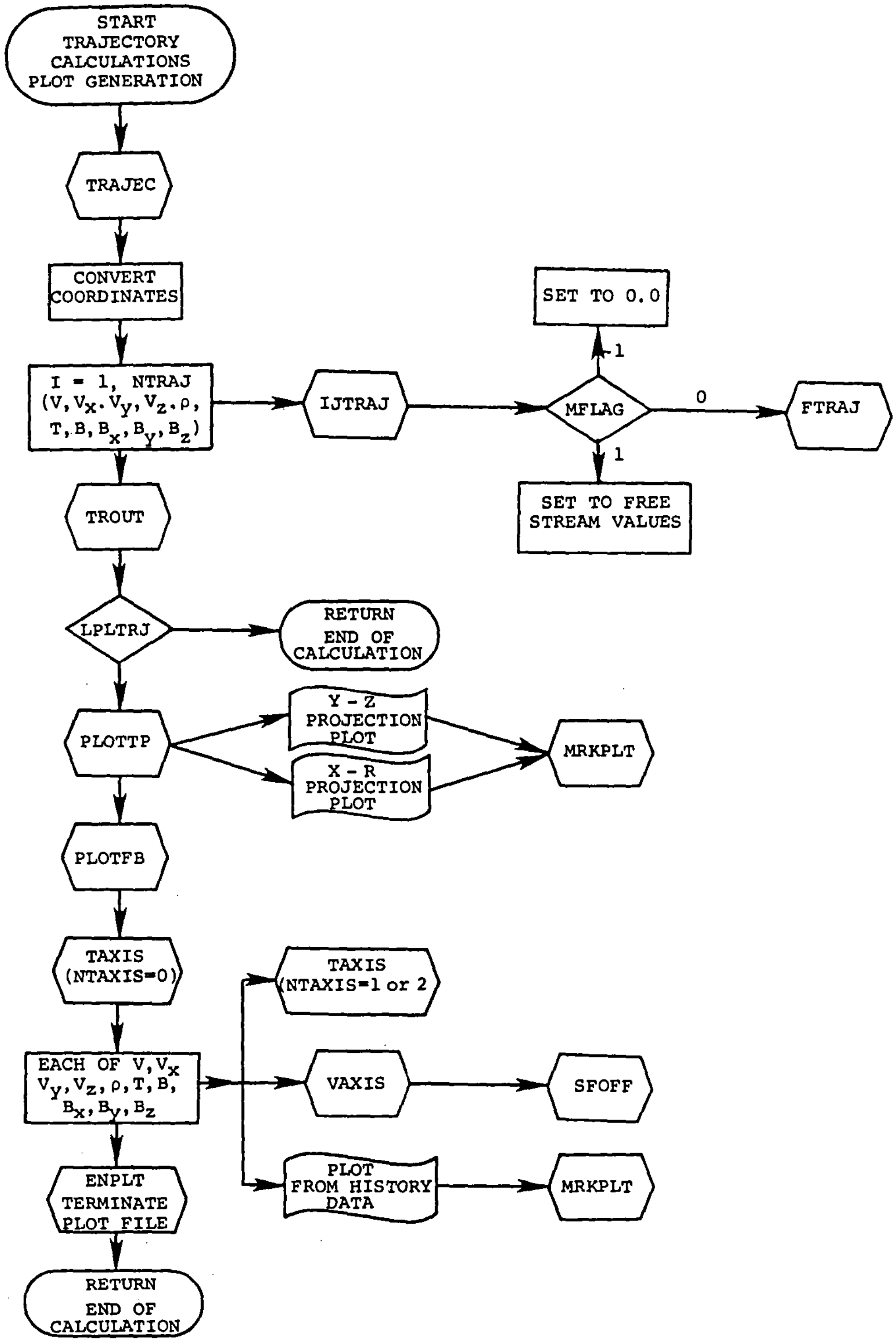


Figure A.1(f).- Flow chart of trajectory calculation.

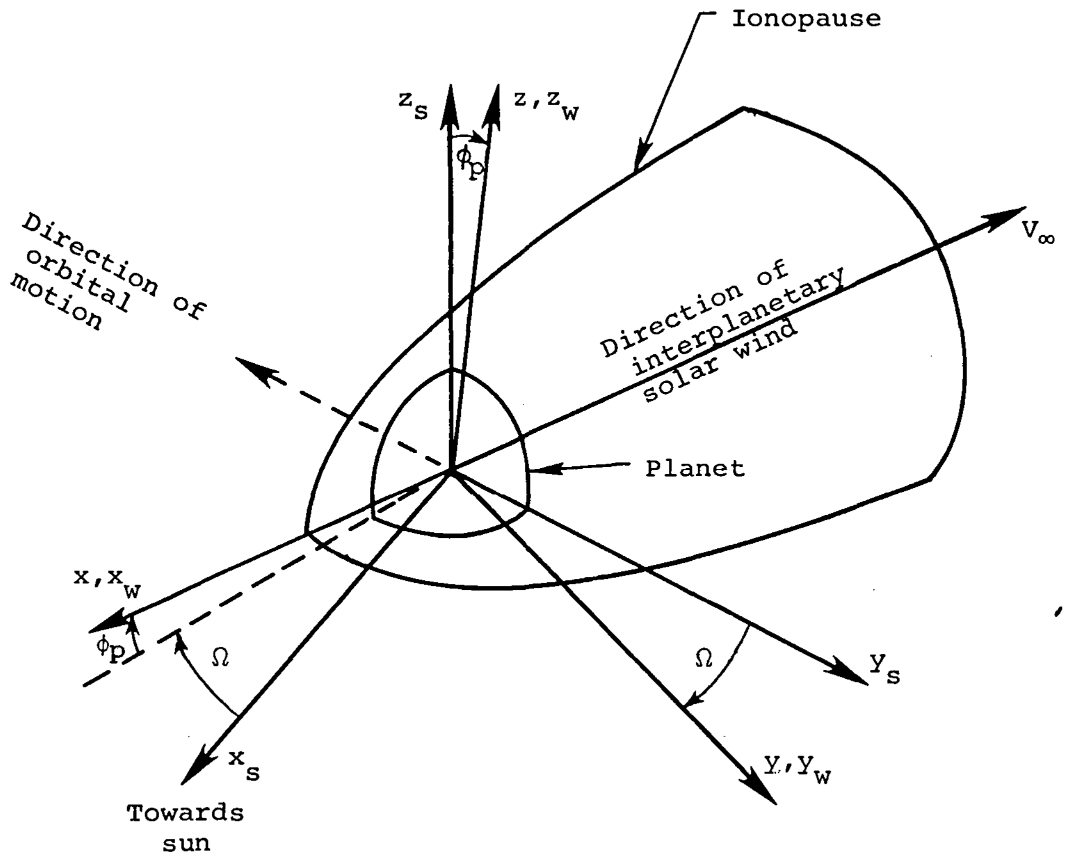


Figure A.2.- Illustration of the azimuthal (Ω) and polar (ϕ_p) solar-wind angles, both shown in a positive sense.

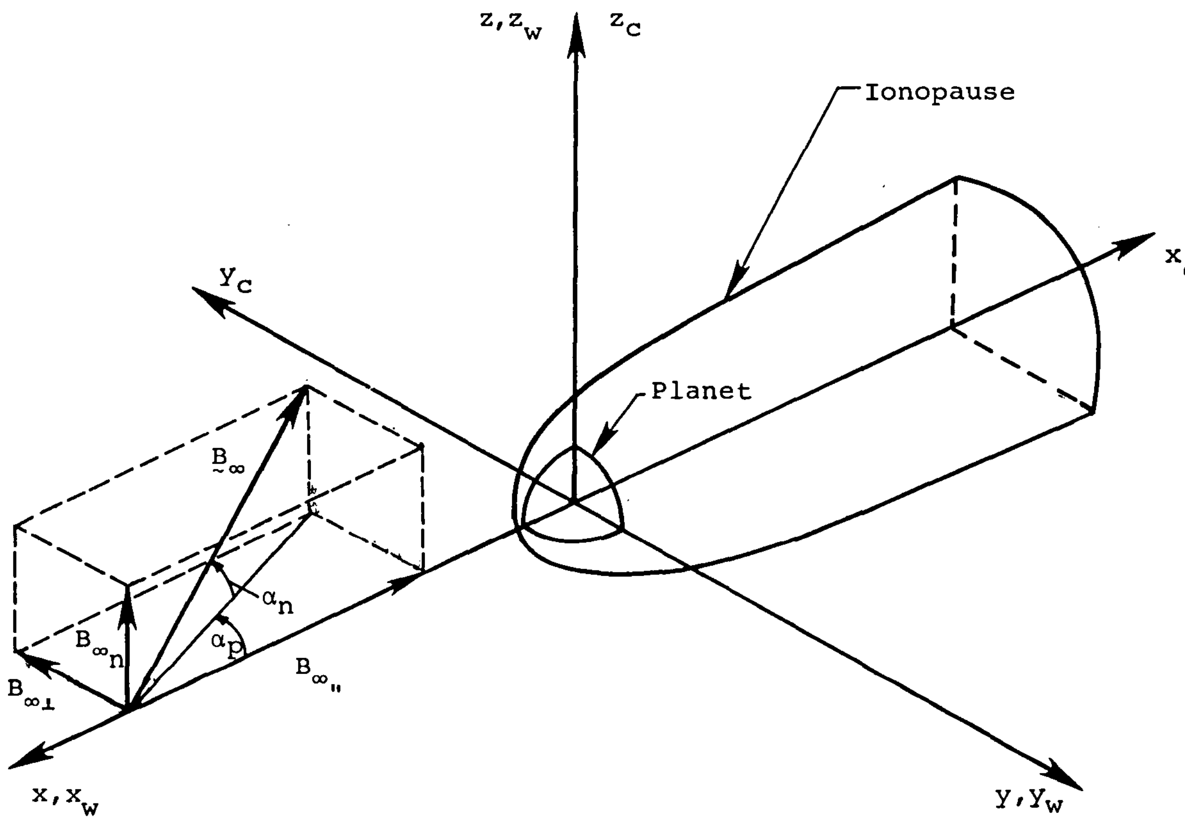


Figure A.3.- Illustration of the interplanetary magnetic field and magnetic-field angles (α_p, α_n) in the solar-wind aligned coordinate systems (x,y,z), (x_w,y_w,z_w), and (x_c,y_c,z_c).

Item No.	Column No. 10	20	30	40	50	60	70	80
0	1							
1	SAMPLE CASE	(DEFAULT IONOPAUSE SHAPE WITH H/RO = 0.03)						
2	3.0	1.6666667	0.03	-10.0				
3	F	T	T	T	T	T	T	F
4	-5.5							
5	14							
	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.75
	0.8	0.93	0.85	0.9	0.92	0.94		
6	11							
	0.5	0.8	1.2	1.6	2.0	2.5	3.0	3.5
	3.8	4.0	4.2					
7	13							
	0.45	0.6	0.8	1.0	1.25	1.5	2.0	2.5
	3.0	3.5	4.0	5.0	6.0			
9	37							
	-90.870	-0.843	3.787	2.027				
	-85.537	-0.765	3.600	2.045				
	-80.203	-0.684	3.393	2.062				
	-74.870	-0.600	3.177	2.073				
	-70.603	-0.533	2.999	2.078				
	-64.203	-0.430	2.721	2.079				
	-57.802	-0.324	2.427	2.067				
	-51.402	-0.217	2.118	2.043				
	-40.735	-0.034	1.562	1.961				
	-39.668	-0.017	1.509	1.951				
	-38.602	-0.010	1.422	1.932				
	-35.935	0.049	1.281	1.902				
	-34.867	0.67	1.232	1.886				
	-32.735	0.158	1.108	1.852				
	-24.202	0.2456	0.5826	1.6658				
	-21.002	0.2954	0.3749	1.5693				
	-16.735	0.356	0.090	1.409				
	-14.602	0.382	-0.052	1.315				
	-11.402	0.415	-0.205	1.150				
	-8.102	0.437	-0.478	0.945				
	-4.900	0.442	-0.703	0.660				
	6.298	0.309	-1.042	-0.266				
	9.498	0.237	-1.055	-0.546				
	12.593	0.160	-1.038	-0.801				
	17.393	0.034	-0.966	-1.162				
	29.127	-0.275	-0.673	-1.887				
	34.460	-0.407	-0.512	-2.155				
	43.860	-0.561	-0.312	-2.444				
	47.260	-0.705	-0.108	-2.670				
	53.660	-0.844	0.0982	-2.925				
	56.862	-0.91	0.201	-3.031				
	58.993	-0.954	0.269	-3.097				
	60.060	-0.975	0.303	-3.131				
	62.193	-1.018	0.371	-3.195				
	67.528	-1.122	0.541	-3.347				
	73.928	-1.241	0.739	-3.514				
	80.328	-1.357	0.839	-3.671				
10	T	0.968	392.0	20.2	102000.	8.8225		
11	4							
	4	9	11	19				
12	T	3.3	0.15	-1.74	-8.64	0.4		

Figure A.4.- Card input for sample case.

```
**************************************************************
**************************************************************
**                      PROGRAM SOLAR                       **
**                                                          **
**                        DETERMINES                        **
**                                                          **
**   SOLAR WIND FLOW PAST PLANETARY MAGNETO/IONOSPHERES     **
**                                                          **
**                          USING                           **
**                                                          **
**     FULLY CONSERVATIVE FINITE DIFFERENCE ALGORITHMS      **
**                                                          **
**                                                          **
**                       WRITTEN BY:                        **
**                                                          **
**       STEPHEN S. STAHARA, BARBARA C. TRUDINGER,          **
**                                                          **
**                  AND DANIEL J. KLENKE                    **
**                                                          **
**                                                          **
**          NIELSEN ENGINEERING AND RESEARCH, INC.          **
**                                                          **
**               MOUNTAIN VIEW, CALIFORNIA                  **
**                                                          **
**************************************************************
**************************************************************
```

```
                                LISTING OF INPUT CARDS FOR THIS RUN
                                ***********************************

        1
  SAMPLE CASE   (DEFAULT IONOPAUSE SHAPE WITH H/RO = 0.03)
3.0       1.6666667 0.03      -10.0
F         T         T         T         T         T         T         F
-5.5
        14
   0.1       0.2       0.3       0.4       0.5       0.6       0.7       0.75
   0.8       0.85      0.88      0.90      0.92      0.94
        11
   0.5       0.8       1.2       1.6       2.0       2.5       3.0       3.5
   3.8       4.0       4.2
        13
   0.45      0.6       0.8       1.0       1.25      1.5       2.0       2.5
   3.0       3.5       4.0       5.0       6.0
        37
-90.970   -0.843    3.787     2.027
-85.537   -0.765    3.600     2.045
-80.203   -0.694    3.393     2.062
-74.870   -0.600    3.177     2.073
-70.533   -0.533    2.999     2.078
-64.203   -0.430    2.721     2.079
-57.802   -0.324    2.427     2.067
-51.402   -0.217    2.118     2.043
-40.735   -0.034    1.562     1.961
-39.668   -0.017    1.509     1.951
-38.602   -0.010    1.422     1.932
-35.935   0.049     1.291     1.902
-34.867   0.067     1.232     1.886
-32.735   0.158     1.108     1.852
-24.202   0.2456    0.5826    1.5658
-21.002   0.2954    0.3749    1.5693
-16.735   0.356     0.099     1.409
-14.602   0.382     -0.052    1.315
-11.402   0.415     -0.205    1.153
-5.102    0.437     -0.478    0.945
-4.900    0.442     -0.703    0.660
6.298     0.309     -1.042    -0.266
9.498     0.237     -1.055    -0.545
12.593    0.163     -1.038    -0.801
17.393    0.034     -0.966    -1.162
29.127    -0.275    -0.673    -1.887
34.453    -0.407    -0.512    -2.155
40.859    -0.562    -0.312    -2.444
47.260    -0.705    -0.169    -2.670
53.650    -0.844    0.0982    -2.925
56.967    -0.91     0.201     -3.031
58.993    -0.954    0.269     -3.097
60.060    -0.975    0.303     -3.131
62.193    -1.018    0.371     -3.195
67.528    -1.122    0.541     -3.347
73.928    -1.241    0.739     -3.514
83.328    -1.357    0.939     -3.671
T         0.968     392.0     20.2      10230...  6.8215
        4
        4         9         11        19
T         3.3       0.15      -1.74     -8.64     0.4
```

Figure A.5.- Abbreviated print output for sample case.

```
SAMPLE CASE   (DEFAULT IONOPAUSE SHAPE WITH H/RC = 0.03)

                                   INPUT VARIABLES
                                   ***************

INTERPLANETARY MACH NO. =  3.00

SPECIFIC HEAT RATIO = 1.667

OBSTACLE GEOMETRY: DEFAULT IONOPAUSE SHAPE
                   FOR CONSTANT SCALE HEIGHT WITH H/RO= .03

PARAMETERS FOR BLUNT BODY CALCULATION

     NO. OF RADIAL MESH POINTS  =  15
     NO. OF ANGULAR MESH POINTS =  24
     NO. OF ADDITIONAL POINTS IN
             BLUNT BODY MESH  =   0
     COURANT NUMBER            = 3.00
     NO. OF ITERATIONS         = 300

TERMINAL DOWNSTREAM LOCATION FOR MARCHING CALCULATION,X/RC=-10.00

TERMINAL DOWNSTREAM LOCATION FOR PLOTTING,X/RC= -5.51

LREBIN = F

LPRFL  = T

LPRST  = T

LPRCON = T

LPRR   = T

LPLOT  = T

LTRAJ  = T

LRSTRT = F

LPLTRJ = T
```

Figure A.5.- Continued.

INPUT FOR TRAJECTORY CALCULATION
****

R/RPLANET = 1.0331

INTERPLANETARY VELOCITY = 3.920E+02

INTERPLANETARY DENSITY = [illegible]

INTERPLANETARY TEMPERATURE = 1.020E+05

INTERPLANETARY MAGNETIC FIELD
MAGNITUDE = .882E+01
X-COMPONENT = -.174E+01
Y-COMPONENT = -.656E+01
Z-COMPONENT = [illegible]

AZIMUTHAL ANGLE = [illegible]

POLAR ANGLE = [illegible]

NTRAJ = 37 NMARKT = 4
(* = POINT TO BE MARKED FOR CROSS REFERENCE)

N		TTRAJ	XTRAJ	YTRAJ	ZTRAJ
		(SUN-PLANET COORDINATE SYSTEM)			
1		[illegible]	-.9431	3.7872	2.0270
2		-85.5377	-.7530	3.6000	2.0450
3		[illegible]	[illegible]	3.3930	2.0520
4	*	-74.8760	-.6300	3.1770	2.0730
5		-70.5030	-.5330	2.9990	2.0790
6		[illegible]	-.4330	2.7220	2.0790
7		-57.8020	-.3240	2.4270	2.0570
8		[illegible]	-.2177	[illegible]	2.0490
9	*	-45.7350	[illegible]	[illegible]	1.9530
10		-39.6590	[illegible]	[illegible]	1.9510
11	*	[illegible]	[illegible]	[illegible]	1.9320
12		-35.9250	.0490	1.2910	1.9020
13		[illegible]	[illegible]	1.2320	1.8380
14		-32.735[illegible]	[illegible]	1.108[illegible]	1.692[illegible]
15		[illegible]	.2435	.5926	1.6650
16		[illegible]	.2954	.3740	1.5593
17		-15.7350	.3551	.0900	1.4090
18		[illegible]	.332[illegible]	-.0520	1.3150
19	*	[illegible]	.415[illegible]	[illegible]	[illegible]
20		[illegible]	.437[illegible]	-.479[illegible]	.9450
21		[illegible]	.4420	-.7330	[illegible]
22		[illegible]	[illegible]	-1.0470	-.2660
23		9.698[illegible]	.237[illegible]	-1.0850	-.5460
24		12.593[illegible]	[illegible]	[illegible]	[illegible]
25		17.3930	[illegible]	-.966[illegible]	-1.3520
26		[illegible]	-.272[illegible]	-.6730	-1.8870
27		34.4500	-.4370	-.5120	-2.1550
28		[illegible]	[illegible]	-.3120	-2.4440
29		47.267[illegible]	[illegible]	[illegible]	[illegible]
30		[illegible]	-.844[illegible]	.1982	-2.9150
31		56.3620	-.91[illegible]	.2010	-3.0310
32		[illegible]	-.954[illegible]	.2690	-3.0970
33		[illegible]	-.973[illegible]	.3030	-3.1310
34		62.[illegible]	[illegible]	.37[illegible]	-3.195[illegible]
35		67.129[illegible]	-1.022[illegible]	.54[illegible]	-3.3470
36		73.923	-1.24[illegible]	.739[illegible]	-3.514[illegible]
37		80.3280	-1.3570	.339[illegible]	-3.671[illegible]

VALUES SPECIFIED FOR CONTOUR CALCULATION
----

14 CONTOUR LEVELS FOR VELOCITY:

.100	.200	.300	.400	.500	.600
.700	.750	.800	.850	.8[illegible]	.9[illegible]
.920	.940				

11 CONTOUR LEVELS FOR DENSITY:

.500	.800	1.200	1.600	2.0[illegible]	2.500
3.000	3.500	3.800	4.000	4.230	

13 CONTOUR LEVELS FOR MAGNETIC FIELD STRENGTH:

.450	.600	.8[illegible]	1.000	1.230	1.500
2.000	2.500	3.0[illegible]	3.5[illegible]	4.000	5.000
6.000					

Figure A.5.- Continued.

BLUNT BODY CALCULATION
****

```
ITERATION   1    RMS OF SHOCK SPEED= 1.5177E-02   MAXIMUM SHOCK SPEED= 5.9856E-02 AT J=25
ITERATION   2    RMS OF SHOCK SPEED= 7.5422E-03   MAXIMUM SHOCK SPEED= 1.9263E-02 AT J=23
ITERATION   3    RMS OF SHOCK SPEED= 1.0212E-02   MAXIMUM SHOCK SPEED= 3.0756E-02 AT J=23
ITERATION   4    RMS OF SHOCK SPEED= 1.5033E-02   MAXIMUM SHOCK SPEED= 4.1855E-02 AT J=23
ITERATION   5    RMS OF SHOCK SPEED= [illegible]  MAXIMUM SHOCK SPEED= 5.2467E-02 AT J=23


ITERATION 296    RMS OF SHOCK SPEED= [illegible]  MAXIMUM SHOCK SPEED= 1.2973E-03 AT J=16
ITERATION 297    RMS OF SHOCK SPEED= 5.3517E-04   MAXIMUM SHOCK SPEED= 1.2751E-03 AT J=16
ITERATION 298    RMS OF SHOCK SPEED= 5.237[illegible]E-04   MAXIMUM SHOCK SPEED= 1.2411E-03 AT J=13
ITERATION 299    RMS OF SHOCK SPEED= [illegible]  MAXIMUM SHOCK SPEED= 1.2853E-03 AT J=13
ITERATION 300    RMS OF SHOCK SPEED= [illegible]  MAXIMUM SHOCK SPEED= 1.2779E-03 AT J=13

FINAL SONIC LINE LOCATION

XSL=  .5622    RSL=  .7745
XSL=  .6734    RSL=  .8031
XSL=  .6839    RSL=  .8292
XSL=  .6941    RSL=  .8539
XSL=  .7067    RSL=  .8777
XSL=  .7211    RSL=  .8998
XSL=  .7373    RSL=  .9230
XSL=  .7553    RSL=  .9394
XSL=  .7743    RSL=  .9540
XSL=  .7940    RSL=  .9638
XSL=  .8160    RSL=  .9820
XSL=  .84[illegible]  RSL=  .9923
XSL=  .864     RSL= [illegible]
XSL=  .8892    RSL= 1.0067
XSL=  .9155    RSL= [illegible]
XSL=  .943[illegible]  RSL= [illegible]
XSL=  .9719    RSL= [illegible]
XSL= [illegible]  RSL= [illegible]
XSL= 1.0321    RSL=  .9986

<<<< % ERROR IN HT =   .4559E+00   RMS OF % ERROR IN HT =   .7480E-01 >>>>

BODY AND FINAL SHOCK SHAPE
```

J	X(BODY)	Y(BODY)	X(SHOCK)	Y(SHOCK)
2	.9994	.0349	1.3043	.0455
3	.9948	.1046	1.3001	.1356
4	.9856	.1738	1.2904	.2275
5	.9719	.2423	1.2780	.3186
6	.9536	.3098	1.2589	.4091
7	.9314	.3761	1.2374	.4993
8	.9039	.4409	1.2094	.5893
9	.8726	.5039	1.1769	.6807
10	.8372	.5647	1.1419	.7702
11	.7979	.6234	1.1022	.8511
12	.7547	.6795	1.0561	.9532
13	.7076	.7329	1.0067	1.0423
14	.6574	.7834	.9516	1.1329
15	.6036	.8308	.8910	1.2264
16	.5468	.8750	.8241	1.3193
17	.4869	.9158	.7525	1.4152
18	.4244	.9531	.6729	1.5114
19	.3592	.9868	.5676	1.6128
20	.2916	1.0170	.4917	1.7143
21	.2219	1.0434	.3877	1.8239
22	.1499	1.0663	.2719	1.9343
23	.0759	1.0857	.1437	2.0555
24	[illegible]	1.1017	[illegible]	2.1803

Figure A.5.- Continued.

MARCHING CALCULATION
••••••••••••••••••••

STEP NO.	DOWNSTREAM LOCATION	BODY ORDINATE	SHOCK ORDINATE
1	-.0468	1.1096	2.2210
2	-.0935	1.1167	2.2605
3	-.1403	1.1225	2.2993
4	-.1871	1.1273	2.3377
5	-.2425	1.1319	2.3823
6	-.2979	1.1354	2.4267
7	-.3533	1.1380	2.4702
8	-.4087	1.1400	2.5131
9	-.4641	1.1414	2.5555
10	-.5329	1.1425	2.6073
11	-.6016	1.1432	2.6583
12	-.6703	1.1436	2.7087
13	-.7390	1.1439	2.7583
14	-.8077	1.1440	2.8073
15	-.8922	1.1441	2.8674
16	-.9766	1.1441	2.9267
17	-1.0546	1.1441	2.9851
18	-1.1494	1.1441	3.0428
19	-1.2349	1.1441	3.0997
20	-1.3404	1.1441	3.1692
21	-1.4460	1.1441	3.2377
22	-1.5515	1.1441	3.3053
23	-1.6571	1.1441	3.3723
24	-1.7626	1.1441	3.4381
25	-1.8920	1.1441	3.5179
26	-2.0213	1.1441	3.5967
27	-2.1506	1.1441	3.6745
28	-2.2799	1.1441	3.7515
29	-2.4093	1.1441	3.8276
30	-2.5663	1.1441	3.9190
31	-2.7234	1.1441	4.0093
32	-2.8805	1.1441	4.0986
33	-3.0375	1.1441	4.1870
34	-3.1946	1.1441	4.2744
35	-3.3837	1.1441	4.3786
36	-3.5728	1.1441	4.4817
37	-3.7619	1.1441	4.5837
38	-3.9511	1.1441	4.6847
39	-4.1401	1.1441	4.7845
40	-4.3659	1.1441	4.9032
41	-4.5917	1.1441	5.0204
42	-4.8175	1.1441	5.1365
43	-5.0433	1.1441	5.2517
44	-5.2691	1.1441	5.3659
45	-5.5367	1.1441	5.5001
46	-5.8043	1.1441	5.6333
47	-6.0719	1.1441	5.7650
48	-6.3395	1.1441	5.8959
49	-6.6072	1.1441	6.0258
50	-6.9222	1.1441	6.1776
51	-7.2373	1.1441	6.3282
52	-7.5523	1.1441	6.4777
53	-7.8673	1.1441	6.6262
54	-8.1824	1.1441	6.7737
55	-8.5511	1.1441	6.9451
56	-8.9198	1.1441	7.1154
57	-9.2885	1.1441	7.2845
58	-9.6571	1.1441	7.4526
59	-10.0258	1.1441	7.6198

DETAILED FLOW FIELD OUTPUT
••••••••••••••••••••••••••

FLOW FIELD VALUES EXTRAPOLATED TO SYMMETRY AXIS, THETA = 0.00 DEGREES

I	R/RB	V/VINF	RHO/RHOINF	T/TINF	P/PINF
1	1.0000	0.0000	3.4163	4.0000	13.6730
2	1.0169	.0240	3.4094	3.9983	13.6278
3	1.0338	.0447	3.4026	3.9946	13.5920
4	1.0507	.0551	3.3941	3.9873	13.5331
5	1.0676	.0853	3.3929	3.9782	13.4578
6	1.0845	.1053	3.3592	3.9667	13.3549
7	1.1014	.1250	3.3531	3.9531	13.2554
8	1.1183	.1444	3.3347	3.9375	13.1303
9	1.1352	.1634	3.3147	3.9199	12.9905
10	1.1521	.1921	3.2911	3.9005	12.8378
11	1.1690	.2004	3.2662	3.8795	12.6711
12	1.1859	.2184	3.2393	3.8570	12.4937
13	1.2025	.2359	3.2104	3.8232	12.3361
14	1.2197	.2528	3.1798	3.8082	12.1394
15	1.2366	.2694	3.1475	3.7823	11.9047
16	1.2535	.2854	3.1135	3.7556	11.6932
17	1.2704	.3009	3.0780	3.7284	11.4760
18	1.2873	.3159	3.0410	3.7008	11.2543
19	1.3042	.3303	3.0010	3.6667	11.0309

Figure A.5.- Continued.

FLOW FIELD VALUES FROM BLUNT BODY CALCULATION

ANGULAR LOCATION NO. 2, AT THETA = 2[illegible] DEGREES

I	RR/R.	R/R.	X/R.	VR/VINF	VX/VINF	FLOW ANGLE	V/VINF	RHO/RHOINF	T/TINF	P/PINF
1	1.0[illegible]	.0349	.9994	.0159	.0025	55.1063	.0435	3.4261	3.9994	13.6624
2	1.0170	.0355	1.0164	.0230	.0233	44.5965	.0347	3.4125	3.9965	13.6340
3	1.0339	.0361	1.0333	.0247	.0459	28.3110	.0540	3.4044	3.9913	13.5877
4	1.0509	.0367	1.0502	.0236	.0693	18.5328	.0725	3.3949	3.9837	13.5243
5	1.0679	.0373	1.0672	.0221	.0900	13.7795	.0929	3.3931	3.9741	13.4445
6	1.0848	.0379	1.0841	.0213	.1130	10.9514	.1134	3.3689	3.9625	13.3492
7	1.1017	.0384	1.1011	.0207	.1292	9.0979	.1304	3.3525	3.9491	13.2390
8	1.1187	.0390	1.1180	.0202	.1492	7.7517	.1487	3.3339	3.9337	13.1147
9	1.1356	.0396	1.1349	.0196	.1661	6.7313	.1666	3.3132	3.9167	12.9769
10	1.1525	.0402	1.1518	.0191	.1839	5.9443	.1843	3.2904	3.8981	12.8264
11	1.1695	.0409	1.1689	.0187	.2021	5.3144	.2017	3.2657	3.8779	12.6639
12	1.1864	.0414	1.1857	.0183	.2192	4.7999	.2189	3.2390	3.8562	12.4899
13	1.2034	.0420	1.2026	.01[illegible]	.2349	4.3695	.2356	3.2102	3.8331	12.3052
14	1.2203	.0426	1.2196	.0176	.2515	4.0037	.2525	3.1797	3.8087	12.1103
15	1.2373	.0432	1.2365	.0172	.2591	3.6847	.2690	3.1473	3.7829	11.9059
16	1.2542	.0438	1.2534	.0169	.2844	3.3995	.2853	3.1133	3.7557	11.6926
17	1.2712	.0444	1.2704	.0165	.3005	3.1221	.3015	3.0775	3.7274	11.4769
18	1.2881	.0450	1.2873	.0161	.3164	2.8742	.3174	3.0401	3.6977	11.2414
19	1.3050	.0455	1.3043	.0151	.3337	2.5934	.3332	3.0011	3.6669	11.0046

ANGULAR LOCATION NO. 3, AT THETA = 5[illegible] DEGREES

I	RR/R.	R/R.	X/R	VR/VINF	VX/VINF	FLOW ANGLE	V/VINF	RHO/RHOINF	T/TINF	P/PINF
1	1.0003	.1046	.9948	.0474	.0047	84.3214	.0534	3.4025	3.9914	13.5810
2	1.0174	.1063	1.0120	.0694	.0274	68.4426	.0696	3.3975	3.9856	13.5411
3	1.0344	.1081	1.0287	.0731	.0521	54.5202	.0857	3.3903	3.9782	13.4873
4	1.0515	.1099	1.0457	.0699	.0737	43.4788	.1016	3.3809	3.9693	13.4200
5	1.0685	.1117	1.0627	.0659	.0945	34.8732	.1169	3.3694	3.9590	13.3393
6	1.0856	.1135	1.0795	.0637	.1142	29.1369	.1327	3.3557	3.9471	13.2454
7	1.1026	.1153	1.0966	.0619	.1334	24.8826	.1485	3.3399	3.9338	13.1385
8	1.1197	.1170	1.1135	.0602	.1519	21.6279	.1643	3.3221	3.9196	13.0190
9	1.1367	.1188	1.1305	.0587	.1690	19.0432	.1812	3.3022	3.9026	12.8871
10	1.1538	.1206	1.1475	.0573	.1875	16.9878	.1960	3.2862	3.8848	12.7429
11	1.1708	.1224	1.1644	.0560	.2045	15.3039	.2117	3.2563	3.8653	12.5868
12	1.1879	.1242	1.1814	.0549	.2217	13.9075	.2273	3.2305	3.8443	12.4190
13	1.2050	.1260	1.1984	.0539	.2384	12.7296	.2435	3.2027	3.8217	12.2397
14	1.2220	.1277	1.2153	.0529	.2549	11.7267	.2596	3.1729	3.7975	12.0493
15	1.2391	.1295	1.2323	.0521	.2713	10.8040	.2759	3.1414	3.7716	11.8479
16	1.2561	.1313	1.2492	.0513	.2877	10.1209	.2921	3.1079	3.7440	11.6359
17	1.2732	.1331	1.2662	.0506	.3041	9.4773	.3084	3.0726	3.7146	11.4136
18	1.2902	.1349	1.2832	.0504	.3213	8.9454	.3249	3.0356	3.6834	11.1813
19	1.3073	.1366	1.3001	.0514	.3376	8.4842	.3416	2.9965	3.6503	10.9392

ANGULAR LOCATION NO.23, AT THETA = 8[illegible] DEGREES

I	RR/R.	R/R.	X/R.	VR/VINF	VX/VINF	FLOW ANGLE	V/VINF	RHO/RHOINF	T/TINF	P/PINF
1	1.0884	1.0857	.0759	.1990	.8433	12.2791	.8565	.9897	1.7562	1.7381
2	1.1424	1.1396	.0797	.2222	.8182	15.1913	.8481	1.0972	1.8422	2.0218
3	1.1964	1.1935	.0835	.2339	.7997	16.2997	.8341	1.1992	1.9130	2.2941
4	1.2504	1.2473	.0872	.2424	.7868	17.0651	.8223	1.2959	1.9704	2.5535
5	1.3044	1.3012	.0910	.2490	.7731	17.8554	.8132	1.3876	2.0161	2.7974
6	1.3584	1.3551	.0948	.2541	.7645	18.3889	.8067	1.4745	2.0513	3.0247
7	1.4124	1.4090	.0985	.2582	.7579	18.8133	.8025	1.5570	2.0776	3.2348
8	1.4664	1.4629	.1023	.2616	.7532	19.1555	.7967	1.6353	2.0960	3.4276
9	1.5205	1.5168	.1061	.2646	.7497	19.4333	.7942	1.7097	2.1079	3.6038
10	1.5745	1.5706	.1099	.2671	.7476	19.6621	.7929	1.7804	2.1143	3.7642
11	1.6285	1.6245	.1136	.2694	.7454	19.8493	.7924	1.8477	2.1162	3.9103
12	1.6825	1.6784	.1174	.2714	.7483	20.0154	.7927	1.9120	2.1149	4.0436
13	1.7365	1.7323	.1211	.2731	.7457	20.1348	.7935	1.9734	2.1112	4.1663
14	1.7905	1.7862	.1249	.2747	.7449	20.2433	.7945	2.0323	2.1063	4.2806
15	1.8445	1.8401	.1287	.2761	.7451	20.3361	.7956	2.0889	2.1012	4.3892
16	1.8986	1.8939	.1324	.2773	.7454	20.4062	.7964	2.1434	2.0970	4.4948
17	1.9526	1.9479	.1362	.2780	.7453	20.4365	.7969	2.1963	2.0948	4.6007
18	2.0066	2.0017	.1400	.2787	.7466	20.4719	.7969	2.2477	2.0955	4.7100
19	2.0606	2.0556	.1437	.2810	.7459	20.6477	.7957	2.2978	2.1064	4.8263

ANGULAR LOCATION NO.24, AT THETA = 9[illegible] DEGREES

I	RR/R.	R/R.	X/R.	VR/VINF	VX/VINF	FLOW ANGLE	V/VINF	RHO/RHOINF	T/TINF	P/PINF
1	1.1017	1.1017	-.0000	.1533	.8722	10.6163	.8849	.8977	1.6566	1.4818
2	1.1616	1.1616	-.0000	.1933	.8467	12.8880	.8672	1.0083	1.7437	1.7576
3	1.2216	1.2216	-.0000	.2097	.8247	14.2631	.8524	1.1127	1.8205	2.0257
4	1.2816	1.2816	-.0000	.2214	.8399	15.3207	.8404	1.2123	1.8828	2.2826
5	1.3415	1.3415	-.0000	.2335	.7953	16.1522	.8302	1.3169	1.9325	2.5257
6	1.4015	1.4015	-.0000	.2379	.7867	16.8154	.8224	1.3968	1.9711	2.7533
7	1.4614	1.4614	-.0000	.2435	.7795	17.3513	.8165	1.4922	2.0001	2.9647
8	1.5214	1.5214	-.0000	.2495	.7742	17.7921	.8122	1.5534	2.0208	3.1599
9	1.5813	1.5813	-.0000	.2526	.7703	18.1583	.8094	1.6407	2.0345	3.3380
10	1.6413	1.6413	-.0000	.2563	.7674	18.4659	.8078	1.7142	2.0423	3.5011
11	1.7012	1.7012	-.0000	.2598	.7654	18.7266	.8071	1.7843	2.0455	3.6499
12	1.7612	1.7612	-.0000	.2623	.7641	18.9491	.8072	1.8512	2.0452	3.7861
13	1.8211	1.8211	-.0000	.2649	.7632	19.1395	.8078	1.9151	2.0426	3.9117
14	1.8811	1.8811	-.0000	.2671	.7625	19.3041	.8086	1.9763	2.0387	4.0293
15	1.9411	1.9411	-.0000	.2691	.7621	19.4481	.8094	2.0351	2.0347	4.1406
16	2.0010	2.0010	-.0000	.2707	.7618	19.5644	.8103	2.0916	2.0317	4.2495
17	2.0610	2.0610	-.0000	.2719	.7619	19.6466	.8102	2.1462	2.0309	4.3587
18	2.1209	2.1209	-.0000	.2731	.7622	19.7114	.8097	2.1991	2.0334	4.4716
19	2.1809	2.1809	-.0000	.2749	.7520	19.9410	.8082	2.2505	2.0464	4.5919

Figure A.5.- Continued.

FLOW FIELD VALUES FROM MARCHING CALCULATION

ADDITIONAL AXIAL LOCATION NO. 1, AT X/R. = -.[illegible]

I	R/R.	VR/VINF	VX/VINF	FLOW ANGLE	V/VINF	RHO/RHOINF	T/TINF	P/PINF
1	1.1096	.1421	.3491	9.[illegible]	[illegible]	[illegible]	1.5679	1.3151
2	1.17[illegible]	.1752	.6149	11.1141	.4776	.3522	1.6356	1.6387
3	1.2339	.1945	.5397	13.4472	.4519	1.0672	1.7712	1.6962
4	1.29[illegible]	.2066	.6223	14.2234	.8459	1.1734	1.9384	2.1513
5	1.3567	.2[illegible]	.[illegible]92	15.1791	.8466	1.2651	1.9922	2.3925
6	1.4[illegible]	.2252	.7987	15.9462	.8377	1.3544	1.9297	2.6136
7	1.4[illegible]	.2319	.7329	16.5672	.8251	1.4395	1.9574	2.9177
8	1.5419	.2417	.7467	17.1799	.82[illegible]	1.5213	1.9775	3.0193
9	1.6037	.2462	.7413	17.[illegible]	.8149	1.5[illegible]2	1.9976	3.1971
10	1.6054	.2[illegible]	.7759	17.[illegible]	.8163	1.6764	2.0012	3.3549
11	1.7271	.2444	.7743	16.[illegible]62	.8[illegible]	1.7499	2.0077	3.5124
12	1.7[illegible]	.2578	.7724	18.4562	.8143	1.6[illegible]	2.0116	3.6605
13	1.8[illegible]	.2518	.77[illegible]	18.6953	.8139	1.5883	2.0125	3.8001
14	1.9123	.2624	.7706	18.[illegible]	.8134	1.9527	2.0134	3.9319
15	1.9741	.2657	.75[illegible]	19.0621	.8137	2.0145	2.0139	4.0569
16	2.0359	.2577	.7582	19.2111	.8125	2.0729	2.0144	4.1755
17	2.0472	.2652	.7578	19.3222	.8137	2.1269	2.0139	4.2932
18	2.1592	.27[illegible]	.75[illegible]	19.41[illegible]	.8144	2.1774	2.0102	4.3770
19	2.2210	.2722	.7584	19.[illegible]40	.81[illegible]	2.2297	2.0[illegible]61	4.4734

ADDITIONAL AXIAL LOCATION NO. 2, AT X/R. = -.0935

I	R/RC	VR/VINF	VX/VINF	FLOW ANGLE	V/VINF	RHO/RHOINF	T/TINF	P/PINF
1	1.0[illegible]67	.1271	.9033	7.7024	.8115	.7913	1.5176	1.2925
2	1.1[illegible]2	.1572	.4723	10.1398	.8470	.9065	1.6347	1.4791
3	1.2431	.1794	.5527	11.6824	.8726	1.0224	1.7219	1.7605
4	1.3073	.1987	.5374	12.1676	.8540	1.1263	1.7925	2.0179
5	1.3709	.2[illegible]	.[illegible]21[illegible]	14.[illegible]47[illegible]	.8471	1.2221	1.8470	2.2577
6	1.4244	.2187	.8102	15.1109	.8397	1.3[illegible]32	1.8883	2.4798
7	1.4979	.227[illegible]	.8[illegible]4	15.8141	.8339	1.4[illegible]4	1.9183	2.6967
8	1.5619	.2339	.7964	16.39[illegible]	.8238	1.4642	1.9407	2.8804
9	1.6[illegible]	.2397	.7907	16.[illegible]950	.8253	1.5652	1.9567	3.0626
10	1.6676	.2449	.78[illegible]	17.2149	.823[illegible]	1.6434	1.9579	3.2341
11	1.7571	.2489	.7827	17.6651	.82[illegible]	1.7[illegible]	1.9755	3.3957
12	1.8157	.2532	.7464	17.576[illegible]	.82[illegible]	1.7915	1.9865	3.5481
13	1.8792	.2567	.77[illegible]	18.2452	.8198	1.8611	1.9838	3.6921
14	1.9427	.2597	.7771	18.4792	.8193	1.9275	1.9860	3.8280
15	2.0063	.2623	.77[illegible]	18.6424	.8180	1.9907	1.9883	3.9575
16	2.0698	.2646	.7746	18.8597	.8186	2.0519	1.9899	4.0809
17	2.1334	.2656	.7735	19.[illegible]	.8185	2.1065	1.9902	4.1920
18	2.1969	.2583	.7733	19.1214	.8191	2.1599	1.9873	4.2923
19	2.2605	.2497	.7743	19.2122	.8197	2.2121	1.9844	4.3896

ADDITIONAL AXIAL LOCATION NO.[illegible], AT X/R. = -9.5571

I	R/R	VR/VINF	VX/VINF	FLOW ANGLE	V/VINF	RHO/RHOINF	T/TINF	P/PINF
1	1.1441	-.[illegible]	.9[illegible]	-.[illegible]	.9612	.6645	1.3423	.8910
2	1.4946	-.[illegible]12	.9545	-.0747	.9545	.7144	1.2665	.8921
3	1.6451	-.[illegible]33	.9671	-.1335	.9576	.7455	1.1945	.8919
4	2.1956	-.[illegible]30	.9775	-.1761	.9776	.7676	1.1329	.8923
5	2.5441	-.[illegible]42	.996[illegible]	-.2440	.9961	.8227	1.0926	.8907
6	2.8965	-.[illegible]	.9923	-.2277	.9923	.8509	1.0462	.8901
7	3.2471	-.[illegible]43	.9962	-.2498	.9962	.8763	1.0229	.8961
8	3.5974	-.[illegible]74	.9999	-.[illegible]	.9999	.8859	1.0014	.8962
9	3.9479	-.[illegible]23	1.[illegible]37	-.7[illegible]29	1.[illegible]36	.9323	.9764	.8639
10	4.2984	-.[illegible]105	1.0042	-.[illegible]24	1.0042	.8956	.9745	.8726
11	4.6489	-.[illegible]	[illegible]15	-.[illegible]34	1.0014	.931[illegible]	.9913	.9229
12	4.9993	.[illegible]124	.9972	.74[illegible]	.9973	.9792	1.0162	.9951
13	5.3498	.[illegible]27	.99[illegible]	1.5197	.9929	1.0318	1.0463	1.0793
14	5.7003	.[illegible]4	.99[illegible]	2.3756	.9959	1.[illegible]67	1.0779	1.1712
15	6.0507	.0542	.9861	3.1578	.9816	1.1414	1.1096	1.2665
16	6.4012	.0[illegible]7	.9735	3.5347	.9759	1.[illegible]980	1.1427	1.3693
17	6.7517	.0769	.9675	4.6275	.9736	1.2814	1.1737	1.4687
18	7.1022	.0903	.9605	5.3724	.9644	1.3095	1.2077	1.5915
19	7.4526	.[illegible]993	.9[illegible]49	5.9165	.9600	1.3579	1.2352	1.6771

ADDITIONAL AXIAL LOCATION NO.[illegible], AT X/R. = -1[illegible].02[illegible]

I	R/RC	VR/VINF	VX/VINF	FLOW ANGLE	V/VINF	RHO/RHOINF	T/TINF	P/PINF
1	1.1441	-.[illegible]	.9469	-.[illegible]	.9619	.6657	1.3439	.8946
2	1.5[illegible]39	-.[illegible]13	.9547	-.0759	.9547	.7[illegible]	1.2659	.8947
3	1.8637	-.0[illegible]23	.9674	-.1363	.9674	.7561	1.1922	.8963
4	2.2234	-.[illegible]31	.9781	-.1797	.9781	.7920	1.1297	.8948
5	2.5832	-.[illegible]42	.9867	-.2432	.9867	.8276	1.0795	.8934
6	2.9429	-.[illegible]52	.9929	-.2974	.9928	.8554	1.0433	.8919
7	3.3027	-.[illegible]45	.9965	-.2572	.9966	.8[illegible]	1.0213	.8979
8	3.6624	-.[illegible]67	[illegible]	-.3[illegible]	[illegible]	.9[illegible]9	.9999	.8907
9	4.0222	-.0120	1.0036	-.6940	1.0036	.9471	.9752	.8678
10	4.3820	-.[illegible]12	1.0045	-.6975	1.0046	.8965	.9725	.8719
11	4.7417	-.[illegible]2[illegible]	[illegible]22	-.[illegible]	1.0022	.9295	.9873	.9174
12	5.1015	.[illegible]2	.9979	.6416	.9980	.9761	1.0122	.9880
13	5.4612	.0291	.9967	1.4491	.9970	1.0270	1.0426	1.0703
14	5.8210	.0390	.997[illegible]	2.2621	.9977	1.0815	1.0721	1.1606
15	6.1807	.0520	.9913	3.0359	.9924	1.1355	1.1047	1.2543
16	6.54[illegible]	.0649	.9746	3.6088	.9769	1.1513	1.1376	1.3552
17	6.9003	.0762	.9685	4.4969	.9725	1.2441	1.1683	1.4535
18	7.26[illegible]	.[illegible]82	.9627	5.2421	.9657	1.3015	1.2022	1.5648
19	7.6[illegible]98	.[illegible]271	.958[illegible]	[illegible]	.9619	1.3494	1.2295	1.6590

Figure A.5.- Continued.

STREAMLINE TRAJECTORY CALCULATION
****

4[illegible] STREAMLINES CALCULATED

STREAMLINE NO. [illegible], STARTING AT Y/R. = [illegible]3[illegible]43, R/R = .[illegible]455
(CORRESPONDS TO THETA = 2[illegible] DEGREES, RR/R. = [illegible]3[illegible])

Y/R	R/R.
1.3[illegible]43	.[illegible]445
1.2782	.0469
1.2521	.[illegible]424
1.226[illegible]	.[illegible]3[illegible]3
1.[illegible]993	.[illegible]526
1.1736	.[illegible]57[illegible]
1.1477	.[illegible]59[illegible]
1.1217	.[illegible]537
1.0956	.[illegible]7[illegible]3
1.0[illegible]9[illegible]	.[illegible]6[illegible]2
1.0434	.[illegible]976
1.0[illegible]73	.[illegible]340
.9912	.2[illegible]
.9651	.2975
.9391	.37[illegible]5
.9130	.4322
.8869	.4953
.8608	.5323
.8347	.57[illegible]
.8[illegible]86	.6137
.7826	.6491
.7565	.68[illegible]
.7304	.7123
.7043	.7416
.6782	.7571
.6521	.7939
.6260	.8153
.6[illegible]	.8373
.5739	.8[illegible]
.5478	.8779
.5217	.8973
.4956	.9134
.4695	.9269
.4434	.9454
.4174	.96[illegible]1
.3913	.9730
.3652	.9872
.3391	.9996
.3130	1.0111
.2869	1.0220
.2609	1.[illegible]334
.2348	1.0421
.2087	1.0[illegible]
.1826	1.0[illegible]
.1565	1.0578
.1304	1.0753
.1043	1.0823
.0783	1.0[illegible]
.0[illegible]22	1.0964
.0261	1.[illegible]
-.[illegible]	1.1[illegible]6
-.1[illegible]56	1.1[illegible]
-.2157	1.1291
-.[illegible]6[illegible]	1.1[illegible]95
-.2[illegible]	1.1349
-.2765	1.1389
-.3322	1.1423
-.3875	1.1446
-.4429	1.1461
-.4983	1.1473
-.5537	1.1481
-.6091	1.1487
-.6644	1.1491
-.7198	1.1493
-.7752	1.1494
-.8305	1.1495
-.8859	1.1495
-.9413	1.1496
-.9967	1.1495
-1.0520	1.1495
-1.1074	1.1495
-1.1627	1.1495
-1.2181	1.1495
-1.2735	1.1495
-1.3288	1.1495
-1.3842	1.1495
-1.4396	1.1494
-1.4949	1.1494
-1.5503	1.1494
-1.6057	1.1494
-1.6610	1.1494
-1.7164	1.1494
-1.7718	1.1494
-1.8272	1.1494
-1.8825	1.1494
-1.9379	1.1494
-1.9933	1.1493
-2.0486	1.1493
-2.1040	1.1493
-2.1594	1.1493
-2.2147	1.1493
-2.2701	1.1493
-2.3255	1.1493
-2.3808	1.1493
-2.4362	1.1493

Figure A.5.- Continued.

-2.4915	1.1493
-2.5469	1.1493
-2.6023	1.1493
-2.6577	1.1493
-2.7131	1.1493
-2.7684	1.1493
-2.8238	1.1493
-2.8792	1.1493
-2.9346	1.1493
-2.9900	1.1493
-3.0453	1.1493
-3.1007	1.1493
-3.1560	1.1493
-3.2114	1.1493
-3.2667	1.1493
-3.3221	1.1493
-3.3775	1.1493
-3.4328	1.1493
-3.4882	1.1493
-3.5435	1.1493
-3.5989	1.1493
-3.6543	1.1493
-3.7097	1.1493
-3.7651	1.1493
-3.8204	1.1493
-3.8758	1.1493
-3.9311	1.1493
-3.9865	1.1493
-4.0419	1.1493
-4.0972	1.1493
-4.1526	1.1493
-4.2080	1.1493
-4.2634	1.1492
-4.3187	1.1492
-4.3741	1.1492
-4.4295	1.1492
-4.4848	1.1492
-4.5402	1.1492
-4.5956	1.1492
-4.6509	1.1492
-4.7063	1.1492
-4.7617	1.1492
-4.8170	1.1492
-4.8724	1.1492
-4.9278	1.1492
-4.9831	1.1492
-5.0385	1.1492
-5.0939	1.1492
-5.1492	1.1492
-5.2046	1.1492
-5.2600	1.1492
-5.3153	1.1492
-5.3707	1.1492
-5.4261	1.1492
-5.4814	1.1492
-5.5367	1.1492

STREAMLINE NO.48, STARTING AT X/R0 = -5.1643, R/R0 = 5.3129

X/R0	R/R0
-5.1643	5.3129
-5.2197	5.3211
-5.2751	5.3291
-5.3304	5.3371
-5.3858	5.3448
-5.4412	5.3526
-5.4965	5.3602
-5.5367	5.3656

STREAMLINE NO.49, STARTING AT X/R0 = -5.4133, R/R0 = 5.4382

X/R0	R/R0
-5.4133	5.4382
-5.4686	5.4461
-5.5240	5.4539
-5.5367	5.4557

Figure A.5.- Continued.

MAGNETIC FIELD COMPONENTS

ANGULAR LOCATION NO. 1, AT THETA = 0.0000 DEGREES

I	RR/R	B/BINF (PARALLEL)	B/BINF (PERP)	B/BINF (IN-PLANE)	B-ANGLE (IN-PLANE)	B/BINF (NORMAL)	BX/BINF (RESULTANT)	BY/BINF (RESULTANT)	BZ/BINF (RESULTANT)
1	1.0000	[illegible]							
2	1.0169	.1819	12.0043	11.8869	89.9445	3.4086	.0115	11.9724	.1558
3	1.0339	.1524	9.7572	8.6703	89.3588	3.4026	.0213	9.6013	.1555
4	1.0507	.2725	7.1915	7.1192	89.7508	3.3941	.0325	7.1116	.1551
5	1.0676	.2967	6.2600	6.2178	89.5251	3.3829	.0405	6.2112	.1546
6	1.0845	.3549	5.6043	5.6[illegible]9	89.4916	3.3592	.0499	5.6329	.1540
7	1.1014	.4201	5.1904	5.1392	89.3433	3.3531	.0589	5.1339	.1533
8	1.1183	.4814	4.9735	4.8257	[illegible]	3.3367	.0676	4.8202	.1524
9	1.1352	.5416	4.5573	4.5127	89.1335	3.3140	.0763	4.5174	.1515
10	1.1521	.5994	4.3350	4.2926	89.4755	3.2911	.0842	4.2975	.1504
11	1.1690	.6547	4.1194	4.0700	88.7077	3.2662	.0919	4.0746	.1493
12	1.1859	.7073	3.9295	3.8817	88.5362	3.2393	.0993	3.8964	.1481
13	1.2028	.7571	3.7333	3.7478	88.3729	3.2104	.1063	3.7424	.1467
14	1.2197	[illegible]	3.6351	3.6018	88.2216	3.1798	.1129	3.5953	.1453
15	1.2366	.8479	3.4934	3.4606	88.1267	3.1475	.1191	3.4551	.1439
16	1.2535	.8896	3.3339	3.3526	87.3551	3.1135	.1248	3.3469	.1423
17	1.2704	.9261	3.2761	3.2462	87.7320	3.0790	.1300	3.2492	.1407
18	1.2873	.9613	3.1651	3.1396	87.5360	3.0410	.1344	3.1334	.1390
19	1.3042	[illegible]	[illegible]	2.9735	87.7908	3.0000	.1404	2.9671	.1371

ANGULAR LOCATION NO. 2, AT THETA = 2.[illegible] DEGREES

I	RR/R	B/BINF (PARALLEL)	B/BINF (PERP)	B/BINF (IN-PLANE)	B-ANGLE (IN-PLANE)	B/BINF (NORMAL)	BX/BINF (RESULTANT)	BY/BINF (RESULTANT)	BZ/BINF (RESULTANT)
1	1.0000	.3466							
2	1.0170	.2161	11.8564	11.7504	88.1212	5.1915	.3852	11.7418	.2372
3	1.0339	.1928	9.7063	9.6276	88.1159	5.0053	.2534	9.6133	.2288
4	1.0509	.2699	7.1605	7.1018	88.1429	4.7899	.2209	7.0907	.2189
5	1.0679	.3142	6.2394	6.2147	88.1349	4.6098	.2019	6.1989	.2167
6	1.0848	.3766	5.6450	5.6015	88.0696	4.4608	.1885	5.5925	.2039
7	1.1017	.4371	5.1740	5.1341	87.9758	4.3127	.1811	5.1256	.1971
8	1.1187	.4986	4.8536	4.8171	87.8727	4.1792	.1755	4.8087	.1910
9	1.1356	.5591	4.5402	4.5066	87.7509	4.0560	.1759	4.4985	.1854
10	1.1525	.6165	4.3168	4.2854	87.6474	3.9327	.1757	4.2773	.1797
11	1.1695	.6588	4.1032	4.0738	87.5317	3.8181	.1753	4.0655	.1745
12	1.1864	.7190	3.9133	3.8865	87.3763	3.7174	.1757	3.8785	.1695
13	1.2034	.7573	3.7359	3.7412	87.2930	3.5969	.1772	3.7331	.1644
14	1.2203	.8030	3.6[illegible]9	3.5968	17.1354	3.4924	.1762	3.5966	.1596
15	1.2373	.8448	3.4777	3.4549	87.3528	3.3897	.1775	3.4468	.1549
16	1.2542	.8893	3.3565	3.3446	87.1061	3.2873	.1687	3.3370	.1502
17	1.2712	.9278	3.2564	3.2371	87.1714	3.1906	.1595	3.2296	.1458
18	1.2881	.9459	3.1513	3.1254	87.2493	3.0952	.1561	3.1225	.1415
19	1.3051	[illegible]	[illegible]	2.9825	86.4379	3.0011	.1951	2.9795	.1372

ANGULAR LOCATION NO. 3, AT THETA = 5.[illegible] DEGREES

I	RR/R	B/BINF (PARALLEL)	B/BINF (PERP)	B/BINF (IN-PLANE)	B-ANGLE (IN-PLANE)	B/BINF (NORMAL)	BX/BINF (RESULTANT)	BY/BINF (RESULTANT)	BZ/BINF (RESULTANT)
1	1.0003	.1317							
2	1.0174	.2356	11.5576	11.4749	84.4879	5.9365	1.1021	11.4099	.4685
3	1.0344	.2490	9.7510	9.6112	84.6493	5.3471	.7319	9.4553	.3815
4	1.0513	.3416	7.0552	7.0330	84.7598	6.8859	.6161	6.9995	.3147
5	1.0683	.3941	6.1883	6.1521	85.1166	5.0142	.5240	6.1333	.2744
6	1.0856	.4454	5.5332	5.5625	85.1349	5.4257	.4773	5.5367	.2480
7	1.1026	.4961	5.1094	5.0826	85.1275	4.9977	.4330	5.0791	.2284
8	1.1197	.5459	4.7957	4.7815	85.0926	4.8835	.4076	4.7594	.2141
9	1.1367	.5969	4.6395	4.4762	85.3739	4.4229	.3941	4.4579	.2022
10	1.1538	.6429	4.2657	4.2764	85.0250	4.1955	.3687	4.2359	.1918
11	1.1708	.6969	4.0563	4.0467	84.9972	4.0069	.3524	4.0295	.1831
12	1.1879	.7350	3.9715	3.8544	84.9394	3.8389	.3405	3.8453	.1755
13	1.2050	.7907	3.7227	3.7179	84.1762	3.6849	.3317	3.6992	.1684
14	1.2220	.8244	3.5774	3.5729	84.8153	3.5496	.3225	3.5555	.1622
15	1.2391	.8668	3.4333	3.4359	84.7555	3.4241	.3131	3.4180	.1565
16	1.2561	.9170	3.3261	3.3247	84.7053	3.3055	.3069	3.3971	.1511
17	1.2732	.9477	3.2150	3.2176	84.6259	3.1984	.3019	3.2201	.1462
18	1.2902	.9861	[illegible]	3.1118	84.5340	3.0957	.2961	3.0944	.1415
19	1.3073	1.0231	2.9991	2.9971	84.4344	2.9968	.2923	2.9738	.1376

Figure A.5.- Continued.

ANGULAR LOCATION NO.23, AT THETA = 36.[illegible] DEGREES

I	RR/R	R/RINE (PARALLEL)	R/RINE (PERP)	R/RINE (IN-PLANE)	R-ANGLE (IN-PLANE)	R/RINE (NORMAL)	RX/RINE (RESULTANT)	RY/RINE (RESULTANT)	RZ/RINE (RESULTANT)
1	1.0[illegible]	.9559							
2	1.0424	.9306	3.5484	3.7422	21.0446	4.0759	3.4883	1.3424	.1263
3	1.1064	1.0072	2.6347	2.7374	27.1395	3.0973	2.4340	1.2455	.1411
4	1.2514	1.0659	2.1387	2.3211	32.3881	2.7332	1.9587	1.2409	.1249
5	1.3644	1.1294	1.9746	2.1101	36.9390	2.5833	1.6916	1.2657	.1172
6	1.3584	1.1584	1.8744	1.9801	41.1527	2.4599	1.4957	1.3379	.1126
7	1.4274	1.2464	1.7453	1.9240	44.3750	2.4025	1.3734	1.3462	.1098
8	1.4064	1.3124	1.7465	1.8435	47.3451	2.3438	1.2746	1.3340	.1080
9	1.5205	1.3575	1.7157	1.8816	49.5324	2.3364	1.1996	1.4197	.1068
10	1.5745	1.4111	1.6973	1.8451	51.9353	2.3211	1.1349	1.4491	.1060
11	1.6258	1.4442	1.5660	1.8264	53.7759	2.3169	1.0999	1.4797	.1054
12	1.6679	1.5197	1.6795	1.8321	55.3379	2.3[illegible]	1.0613	1.5154	.1052
13	1.7355	1.5659	1.6773	1.8374	56.7370	2.2935	1.0041	1.5305	.1048
14	1.79[illegible]	1.5967	1.5787	1.8357	57.7451	2.2913	.9721	1.5543	.1047
15	1.8445	1.5619	1.6344	1.8636	59.1324	2.2691	.9453	1.5810	.1046
16	1.8985	1.7171	1.6924	1.8535	50.[illegible]	2.2891	.9227	1.5056	.1046
17	1.9528	1.7603	1.7[illegible]	1.8651	60.9435	2.29[illegible]	[illegible]	1.5286	.1047
18	2.0066	1.7938	1.7137	1.8762	51.1431	2.2959	.8929	1.5502	.1048
19	2.0[illegible]	1.8245	1.7[illegible]	1.8777	52.3624	2.2978	.8784	1.6572	.1056

ANGULAR LOCATION NO.24, AT THETA = 90.0000 DEGREES

I	RR/R	R/RINE (PARALLEL)	R/RINE (PERP)	R/RINE (IN-PLANE)	R-ANGLE (IN-PLANE)	R/RINE (NORMAL)	RX/RINE (RESULTANT)	RY/RINE (RESULTANT)	RZ/RINE (RESULTANT)
1	1.0117	.7744							
2	1.1616	.8761	3.3958	3.4545	19.[illegible]412	3.7219	3.2614	1.1319	.1762
3	1.2216	.9494	2.4144	2.5249	25.4046	2.8490	2.2765	1.0901	.1303
4	1.2816	1.0144	2.1296	2.3457	31.5556	2.5491	1.8279	1.1229	.1165
5	1.3425	1.0449	1.9[illegible]	2.1549	35.5[illegible]	2.4353	1.5661	1.1565	.1099
6	1.4015	1.1487	1.7254	1.8548	41.7471	2.3271	1.4337	1.2296	.1064
7	1.4624	1.2107	1.66[illegible]	1.8[illegible]	44.3[illegible]	2.2795	1.2424	1.2614	.1042
8	1.5214	1.2690	1.6343	1.7708	47.2529	2.2549	1.2[illegible]7	1.2991	.1031
9	1.5[illegible]3	1.3[illegible]	1.6152	1.7548	49.9335	2.2379	1.1384	1.3416	.1023
10	1.6423	1.3848	1.6[illegible]	1.7455	52.[illegible]63	2.2299	1.0722	1.3791	.1019
11	1.7012	1.4472	1.5966	1.7415	53.4352	2.2252	1.0256	1.4057	.1017
12	1.7612	1.4943	1.5[illegible]45	1.7424	55.[illegible]	2.2237	.9859	1.4350	.1016
13	1.8211	1.5469	1.5967	1.7465	56.[illegible]	2.2245	.9507	1.4533	.1017
14	1.8811	1.5996	1.5[illegible]	1.754[illegible]	53.2338	2.2263	.9223	1.4898	.1018
15	1.9412	1.6472	1.5219	1.7655	59.3552	2.2295	.8993	1.5174	.1019
16	2.0010	1.6947	1.5291	1.7709	60.2969	2.2337	.87[illegible]5	1.5435	.1021
17	2.0[illegible]	1.7398	1.5346	1.7935	51.0450	2.2385	.8674	1.5577	.1023
18	2.1209	1.79[illegible]	1.5486	1.8157	51.5166	2.2432	.8594	1.5905	.1026
19	2.1539	1.8185	1.5558	1.8197	52.[illegible]492	2.2535	.8527	1.6058	.1029

ADDITIONAL AXIAL LOCATION NO. 1, AT X/R. = -.0468

I	R/R	R/RINE (PARALLEL)	R/RINE (PERP)	R/RINE (IN-PLANE)	R-ANGLE (IN-PLANE)	R/RINE (NORMAL)	RX/RINE (RESULTANT)	RY/RINE (RESULTANT)	RZ/RINE (RESULTANT)
1	1.1090	.7552							
2	1.1725	.8967	3.[illegible]65	3.27[illegible]	1[illegible].[illegible]69	3.5517	3.[illegible]	1.1103	.1623
3	1.2339	.9906	2.3099	2.4133	24.8756	2.7529	2.1971	1.0142	.1258
4	1.295	1.0338	1.9457	2.0598	30.9459	2.4755	1.7644	1.0581	.1131
5	1.3567	1.0616	1.7881	1.8795	36.1459	2.3425	1.5153	1.1072	.1070
6	1.4155	1.1291	1.6595	1.7557	40.3315	2.2672	1.3588	1.1557	.1036
7	1.4612	1.1876	1.6069	1.7369	44.2412	2.2395	1.2490	1.2105	.1016
8	1.5419	1.2491	1.5777	1.7[illegible]	47.1563	2.2124	1.1626	1.2535	.1006
9	1.6037	1.3093	1.5534	1.7[illegible]	49.7439	2.1953	1.0977	1.2964	.1001
10	1.6654	1.3484	1.5562	1.6962	5[illegible].7915	2.1863	1.0436	1.3349	.0999
11	1.7271	1.4262	1.5345	1.6975	53.4691	2.1876	.9959	1.3696	.1000
12	1.7888	1.4925	1.5573	1.7090	55.5965	2.1907	.9634	1.4021	.1001
13	1.8506	1.5377	1.5534	1.72[illegible]	56.7933	2.1972	.9319	1.4341	.1004
14	1.9123	1.5992	1.5724	1.7229	58.2430	2.2022	.9049	1.4641	.1007
15	1.9740	1.6391	1.5944	1.7375	59.4375	2.2095	.8825	1.4945	.1010
16	2.0358	1.6463	1.5982	1.7531	60.3645	2.2145	.8659	1.5221	.1012
17	2.0975	1.7305	1.6110	1.7683	61.0556	2.2198	.8549	1.5458	.1015
18	2.1592	1.7733	1.6238	1.7838	61.7140	2.2231	.8459	1.5662	.1016
19	2.2210	1.8177	1.6333	1.7960	62.0527	2.2297	.8409	1.5949	.1019

ADDITIONAL AXIAL LOCATION NO. 2, AT X/R3 = -.0955

I	R/R.	R/RINE (PARALLEL)	R/RINE (PERP)	R/RINE (IN-PLANE)	R-ANGLE (IN-PLANE)	R/RINE (NORMAL)	RX/RINE (RESULTANT)	RY/RINE (RESULTANT)	RZ/RINE (RESULTANT)
1	1.1447	.7219							
2	1.1902	.9134	3.0334	3.1155	16.3126	3.4165	2.9829	.8946	.1557
3	1.2436	.9909	2.2335	2.3340	23.3501	2.6573	2.1349	.9310	.1225
4	1.3172	.9564	1.9597	1.9707	30.7492	2.3979	1.7207	.9926	.1096
5	1.3709	1.0353	1.8354	1.8036	35.5363	2.2737	1.4557	1.0468	.1039
6	1.4344	1.1138	1.5962	1.7194	40.1333	2.2092	1.3132	1.1371	.1010
7	1.4979	1.1565	1.55[illegible]	1.6793	43.9325	2.1727	1.2074	1.1531	.0993
8	1.5615	1.2297	1.5277	1.6597	47.0371	2.1561	1.1293	1.2126	.0985
9	1.6210	1.2917	1.5156	1.6524	49.6367	2.1492	1.0693	1.2577	.0982
10	1.6844	1.3528	1.5147	1.6523	51.9964	2.1489	1.0185	1.2988	.0982
11	1.7521	1.4120	1.5165	1.6570	53.9526	2.1542	.9764	1.3366	.0985
12	1.8157	1.4499	1.5222	1.6656	55.5619	2.1597	.9409	1.3722	.0987
13	1.8792	1.5258	1.5315	1.6776	56.9954	2.1695	.9128	1.4054	.0992
14	1.9427	1.5793	1.5424	1.6905	58.3149	2.1761	.8872	1.4374	.0995
15	2.0063	1.6313	1.5572	1.7077	59.5[illegible]	2.1860	.8659	1.4690	.0999
16	2.0698	1.6797	1.5725	1.7256	60.4159	2.1926	.8513	1.4991	.1002
17	2.1334	1.7244	1.5875	1.7432	61.3454	2.2004	.8421	1.5243	.1006
18	2.1969	1.7592	1.6029	1.7521	61.3974	2.2054	.8427	1.5454	.1008
19	2.2605	1.8132	1.6142	1.7778	62.1791	2.2121	.8309	1.5672	.1011

ADDITIONAL AXIAL LOCATION NO.44, AT X/R3 = -5.2651

I	R/R	R/RINE (PARALLEL)	R/RINE (PERP)	R/RINE (IN-PLANE)	R-ANGLE (IN-PLANE)	R/RINE (NORMAL)	RX/RINE (RESULTANT)	RY/RINE (RESULTANT)	RZ/RINE (RESULTANT)
1	1.1442	.6548							
2	1.3767	.6440	1.1971	1.2607	19.0739	1.5489	1.1931	.4332	.0708
3	1.6132	.6758	.9649	1.0377	54.1552	1.2495	.5894	.8160	.0571
4	1.8478	.7177	.9322	.9579	66.4459	1.1462	.3825	.8772	.0524
5	2.0823	.7424	.8929	.9131	76.5026	1.1051	.3045	.8599	.0505
6	2.3168	.7484	.8475	.8626	77.5504	1.0754	.2583	.8221	.0492

Figure A.5.- Continued.

7	2.5544	.7987	.7913	[illegible]	73.6105	1.0441	.2283	.7764	.0477
8	2.7815	.8131	.7647	.7788	72.5559	1.0269	.2317	.7426	.0474
9	3.0201	.8457	.7599	.7779	73.2241	1.0641	.2343	.7440	.0486
10	3.2500	.9126	.7775	.7917	72.5330	1.0219	.2344	.7525	.0508
11	3.4194	.9449	.7979	.8225	73.3032	1.1637	.2361	.7869	.0532
12	3.7241	1.0275	.8195	.8585	73.1631	1.2160	.2483	.8208	.0556
13	3.9556	1.0790	.8507	.8939	72.8128	1.2652	.2815	.8464	.0579
14	4.1932	1.1822	.8987	.9418	72.1471	1.3143	.3041	.8903	.0601
15	4.4277	1.2102	.9469	.9951	72.9853	1.3892	.3279	.9398	.0621
16	4.6623	1.2481	.9947	1.0451	72.6290	1.4127	.3432	.9991	.0641
17	4.8967	1.3227	1.0356	1.0972	72.6651	1.4459	.3629	1.0341	.0659
18	5.1313	1.3765	1.0842	1.1479	72.4919	1.4814	.3829	1.0808	.0677
19	5.3659	1.4726	1.1224	1.1916	70.3123	1.5112	.4009	1.1212	.0694

ADDITIONAL AXIAL LOCATION NO.45, AT Y/RO = -0.5367

I	R/RO	B/BINF (PARALLEL)	B/BINF (PERP)	B/BINF (IN-PLANE)	B-ANGLE (IN-PLANE)	B/BINF (NORMAL)	BX/BINF (RESULTANT)	BY/BINF (RESULTANT)	BZ/BINF (RESULTANT)
1	1.0442	.6755							
2	1.3562	.6457	1.0204	1.0852	19.957	1.5337	1.0187	.3712	.0701
3	1.6282	.6788	.9219	.9655	56.3893	1.2410	.5341	.9036	.0567
4	1.8701	.7116	.9360	.9588	66.2543	1.1413	.3547	.8897	.0522
5	2.1171	.7470	.9165	.9341	72.0706	1.1015	.2573	.8878	.0503
6	2.3514	.7756	.8807	.8950	74.0330	1.0759	.2451	.8599	.0492
7	2.5962	.7896	.8352	.8478	74.8517	1.0462	.2213	.8175	.0478
8	2.8381	.8178	.7931	.8074	73.9732	1.0340	.2227	.7752	.0473
9	3.0801	.8464	.7981	.8132	74.4496	1.0565	.2151	.7729	.0483
10	3.3221	.9009	.7992	.8167	74.7046	1.0809	.2152	.7869	.0503
11	3.5641	.9513	.8109	.8419	74.2067	1.1522	.2289	.8092	.0527
12	3.8060	1.0229	.8513	.8784	74.1780	1.2035	.2466	.8439	.0550
13	4.0481	1.0839	.8728	.9172	72.0543	1.2593	.2791	.8622	.0573
14	4.2901	1.1441	.9097	.9601	71.8379	1.3011	.2990	.9113	.0595
15	4.5321	1.2029	.9687	1.0136	71.5313	1.3458	.3191	.9609	.0615
16	4.7741	1.2593	1.0121	1.0629	71.3940	1.3891	.3388	1.0063	.0635
17	5.0161	1.3123	1.0528	1.1067	71.0551	1.4272	.3579	1.0482	.0652
18	5.2581	1.3668	1.0965	1.1579	70.9259	1.4674	.3779	1.0933	.0671
19	5.5001	1.4124	1.1171	1.1836	70.3851	1.4977	.3933	1.1152	.0685

VELOCITY AND TEMPERATURE CONTOURS
****

14 VELOCITY (TEMPERATURE) CONTOUR LINES FOUND

8 POINTS IN CONTOUR LINE OF V/VINF = .1000, T/TINF = 3.970

Y/RO	R/RO
.9882	.1342
1.0067	.1450
1.0259	.1344
1.0448	.1138
1.0459	.1270
1.0654	.0593
1.0735	.0375
1.0840	[illegible]

15 POINTS IN CONTOUR LINE OF V/VINF = .2000, T/TINF = 3.880

Y/RO	R/RO
.9605	.2944
.9787	.2846
.9976	.2824
1.0171	.2775
1.0373	.2693
1.0513	.2621
1.0579	.2578
1.0786	.2407
1.1061	.2183
1.1171	.1974
1.1224	.1569
1.1457	.1356
1.1517	.1211
1.1672	.1463
1.1666	[illegible]

Figure A.5.- Continued.

11 MAGNETIC FIELD CONTOUR LINES FOUND
(FOR COMPONENT ALONG FIELD LINES PARALLEL TO FLOW IN FREESTREAM)

11 POINTS IN CONTOUR LINE OF R/RINF (PARALLEL) = .450

X/RO	R/RO
.9821	.1963
.9999	.1904
1.0192	.1813
1.0212	.1801
[illegible]	[illegible]
1.0582	.1473
1.0792	.1169
1.0812	.1136
[illegible]	.0953
[illegible]	.1385
[illegible]	[illegible]

15 POINTS IN CONTOUR LINE OF R/RINF (PARALLEL) = .5[illegible]

X/RO	R/RO
.9667	.2515
.9853	.2598
1.0147	.2554
[illegible]	.2526
[illegible]	.2481
1.0439	.2373
[illegible]	.2221
[illegible]	[illegible]
[illegible]	.1926
[illegible]	.1763
1.1298	.1245
[illegible]	[illegible]
1.1493	[illegible]
[illegible]	[illegible]
1.1523	[illegible]

12 MAGNETIC FIELD CONTOUR LINES FOUND
(FOR COMPONENT ALONG FIELD LINES PERPENDICULAR TO FLOW IN FREESTREAM)

[illegible] POINTS IN CONTOUR LINE OF R/RINF (PERPENDICULAR) = 5.[illegible]

X/RO	R/RO
[illegible]	.9282
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]
.7361	.7543
[illegible]	[illegible]
.7864	[illegible]
.9278	[illegible]
[illegible]	[illegible]
.9217	.5322
[illegible]	[illegible]
.9577	.4571
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	.2590
[illegible]	.2293
[illegible]	[illegible]
1.0679	[illegible]
1.0743	[illegible]
[illegible]	[illegible]

25 POINTS IN CONTOUR LINE OF R/RINF (PERPENDICULAR) = 5.10[illegible]

X/RO	R/RO
[illegible]	1.1242
.3721	[illegible]
.4427	.9932
[illegible]	.9585
[illegible]	.9146
.6352	.8763
[illegible]	[illegible]
[illegible]	[illegible]
.7576	.7794
.8023	.7279
[illegible]	[illegible]
[illegible]	.6094
[illegible]	[illegible]
[illegible]	[illegible]
.9463	[illegible]
.9645	[illegible]
[illegible]	.4321
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	.2671
[illegible]	.2178
[illegible]	[illegible]
[illegible]	[illegible]
[illegible]	[illegible]

Figure A.5.- Continued.

DENSITY CONTOURS
****

7 DENSITY CONTOUR LINES FOUND

25 POINTS IN CONTOUR LINE OF RHO/RHOINF = 2.[illegible]

X/R0	R/R0
.943[illegible]	.4512
.9[illegible]55	.4782
.9201	.4949
.9339	.5111
.9431	.5266
.9629	.5413
.9731	.5549
.9940	.5672
1.0[illegible]07	.5775
1.0243	.5839
1.0470	.5925
1.0669	.5933
1.0[illegible]94	.5910
1.1[illegible]3[illegible]	.5875
1.1342	.5659
1.1463	.5591
1.1684	.5346
1.2[illegible]27	.4659
1.2[illegible]42	.4929
1.2394	.4327
1.2466	.3815
1.2676	.3161
1.2862	.2763
1.2987	.1365
1.3032	.0655

26 POINTS IN CONTOUR LINE OF RHO/RHOINF = 2.[illegible]

X/R0	R/R0
.7322	.6523
.7566	.7121
.75[illegible]5	.7434
.7642	.776[illegible]
.7661	.7933
.7571	.8[illegible]90
.7649	.8643
.77[illegible]	.8827
.770[illegible]	.9183
.77[illegible]4	.92[illegible]5
.7592	.9611
.757[illegible]	1.0[illegible]42
.7635	1.05[illegible]2
.7633	1.[illegible]5[illegible]9
.7558	1.[illegible]969
.7480	1.1313
.7414	1.1804
.7366	1.2[illegible]57
.7296	1.2725
.7[illegible]46	1.3252
.6980	1.3435
.668[illegible]	1.4232
.6517	1.4527
.6285	1.5[illegible]56
.5953	1.6[illegible]8[illegible]
.5757	1.6233

Figure A.5.- Continued.

13 MAGNETIC FIELD CONTOUR LINES FOUND
(FOR COMPONENT ALONG FIELD LINES NORMAL TO FLOW IN FREESTREAM)

26 POINTS IN CONTOUR LINE OF B/BINF (NORMAL) = 5.[illegible]

X/RC	R/RC
1.0142	.0687
1.0319	.0575
1.[illegible]475	.[illegible]79[illegible]
1.[illegible]627	.[illegible]
1.[illegible]626	.11[illegible]7
1.0625	.1170
1.[illegible]521	.1955
1.0427	.2549
1.0400	.2593
1.0394	.2611
1.[illegible]190	.3311
.9945	.[illegible]
.9643	.4703
.9291	.5364
.8899	.5996
.8822	.[illegible]
.8454	.[illegible]5
.7981	.7185
.7459	.7724
.6996	.8[illegible]19
.6395	.8217
.6226	.87[illegible]7
.5720	.9[illegible]54
.5[illegible]79	.9[illegible]33
.44[illegible]6	.9[illegible]
.3976	[illegible]72

31 POINTS IN CONTOUR LINE OF B/BINF (NORMAL) = 5.[illegible]

X/RC	R/RC
.2[illegible]	1.[illegible]744
.3053	1.[illegible]649
.3792	1.[illegible]391
.44[illegible]	[illegible]72
.5[illegible]56	.9697
.5352	.9[illegible]
.5622	.93[illegible]
.6441	.[illegible]92
.7[illegible]59	.[illegible]2
.7134	.[illegible]37[illegible]
.7647	.7918
.8[illegible]85	.737[illegible]
.8455	.[illegible]036
.[illegible]	.[illegible]
.9153	.5[illegible]74
.95[illegible]	.5523
.9651	.536[illegible]
.9937	.4645
1.[illegible]25[illegible]	.4[illegible]42
1.052[illegible]	.34[illegible]
1.[illegible]721	.2573
1.[illegible]9	.22[illegible]
1.[illegible]595	.[illegible]921
1.0961	.[illegible]93
1.0965	.[illegible]5[illegible]
1.0316	.[illegible]8[illegible]
1.0659	.[illegible]91
1.0498	.[illegible]4[illegible]
1.0337	.0361
1.0333	.[illegible]36[illegible]
1.0164	.[illegible]317

Figure A.5.- Continued.

TRAJECTORY CALCULATION

(SOLAR WIND COORDINATE SYSTEM)

TRAJECTORY COORDINATES

R /RPLANET = [illegible] 331

N	TIME	X/RP	Y/RP	Z/RP	R/RP	X/RPLANET	Y/RPLANET	Z/RPLANET	R/RPLANET
1	[illegible]	[illegible]	-3.6125	1.9640	4.2125	1.0543	-3.7322	2.0298	4.2494
2	-88.5370	.9347	-3.4364	1.9421	3.9571	.9555	-3.5530	2.0475	4.0982
3	-84.2130	.8449	-3.2409	1.9712	3.8174	.8728	-3.3496	2.0643	3.9392
4	-74.8700	.7515	-3.1360	2.0156	3.6417	.7765	-3.1372	2.0750	3.7614
5	-72.6030	.6749	-2.9695	2.0133	3.5143	.6993	-2.9633	2.0799	3.6304
6	-64.2030	.5519	-2.6056	2.1139	3.2932	.5895	-2.6947	2.1815	3.4121
7	-57.8020	.4431	-2.3274	2.0070	3.0708	.4278	-2.4743	2.1692	3.1715
8	-51.4020	.3228	-2.0347	1.9745	2.8385	.3332	-2.1121	2.1433	2.9319
9	-49.7350	.1149	-1.8870	1.8888	2.4243	.1187	-1.5575	1.9613	2.5945
10	-39.6080	.0956	-1.4973	1.8838	2.3457	.0987	-1.5055	1.9519	2.4646
11	-39.6070	.0546	-1.3737	1.8704	2.3216	.0568	-1.4491	1.9322	2.3973
12	-38.9350	.0192	-1.2407	1.8417	2.2212	.0198	-1.2917	1.9921	2.2936
13	-34.9670	-.0105	-1.1943	1.8247	2.1914	[illegible]	-1.2338	1.8956	2.2537
14	-32.7350	-.0954	-1.0725	1.7925	2.1025	-.0985	-1.1153	1.8517	2.1517
15	-24.2020	-.2091	-.3757	1.6120	1.7122	-.2160	-.3956	1.6652	1.7696
16	-21.6020	-.2686	-.3795	1.5184	1.5649	-.2774	-.3923	1.5686	1.6165
17	-16.7350	-.3425	-.1153	1.3930	1.3672	-.3539	-.1193	1.4391	1.4124
18	-14.6020	-.3754	.0291	1.2719	1.2723	-.3878	.0239	1.3141	1.3143
19	-11.4020	-.4154	.1751	1.1121	1.1258	-.4291	.1444	1.1489	1.1630
20	-8.0120	-.4523	.4375	.9136	1.0130	-.4663	.4521	.9435	1.0445
21	-4.9000	-.4696	.5547	.5277	.9139	-.4935	.5764	.5597	.9442
22	6.2940	-.3567	.9898	-.2164	1.0229	-.3678	1.0221	-.2570	1.0568
23	9.4980	-.2956	1.0163	-.5293	1.1371	-.2959	1.0396	-.5459	1.1746
24	12.5930	-.2104	.9942	-.7759	[illegible]	-.2174	1.0271	-.8016	1.3928
25	17.3930	-.0837	.9316	-1.1296	1.4657	-.0965	.9624	-1.1622	1.5196
26	29.1270	.2331	.6557	-1.6260	1.9436	.2407	.6377	-1.8864	2.0078
27	34.4600	.3707	.5175	-2.0851	2.1493	.3825	.5346	-2.1543	2.2194
28	40.8600	.5310	.3328	-2.3444	2.3477	.5455	.3438	-2.4426	2.4566
29	47.2600	.6874	.1437	-2.5628	2.5968	.7146	.1494	-2.6692	2.6723
30	53.6600	.8285	-.0479	-2.8292	2.8295	.8557	-.0495	-2.9229	2.9232
31	56.8620	.8983	-.1435	-2.9317	2.9352	.9262	-.1492	-3.0284	3.0322
32	59.9930	.9448	-.2064	-2.9924	3.0126	.9767	-.2136	-3.0945	3.1018
33	60.0600	.9671	-.2385	-3.0259	3.0377	.9995	-.2464	-3.1284	3.1384
34	62.1530	1.0125	-.3015	-3.0901	3.1369	1.0463	-.3118	-3.1923	3.2075
35	67.5280	1.1229	-.4613	-3.2371	3.2695	1.1604	-.4755	-3.3440	3.3776
36	73.9280	1.2494	-.6450	-3.3983	3.4495	1.2907	-.6663	-3.5106	3.5733
37	80.3280	1.3730	-.8346	-3.5499	3.6461	1.4184	-.8593	-3.6673	3.7566

FLOW FIELD AND MAGNETIC FIELD COMPONENTS ALONG TRAJECTORY

(SOLAR WIND COORDINATE SYSTEM)

(NON-DIMENSIONALIZED BY INTERPLANETARY VALUES)

N	TIME	V/VINF	VX/VINF	VY/VINF	VZ/VINF	RHO/RHOINF	TEMP/TEMPINF	B/BINF	BX/BINF	BY/BINF	BZ/BINF
1	-90.9700	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
2	-85.5370	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
3	-80.2030	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
4	-74.9700	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
5	-70.6030	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
6	-64.2030	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
7	-57.9020	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
8	-51.4020	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
9	-45.7350	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
10	-39.6680	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
11	-38.5020	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
12	-35.9350	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
13	-34.9670	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
14	-32.7350	.8012	.7503	-.1435	.2336	2.2747	2.0790	2.0935	-.0612	2.0363	.4687
15	-24.2020	.7818	.7294	-.0944	.2651	2.0697	2.1642	2.2495	-.0223	2.2119	.4032
16	-21.6020	.7568	.7115	-.0692	.2775	2.0072	2.2389	2.3575	.0394	2.3341	.3513
17	-16.7350	.7445	.6934	-.0231	.2957	1.9225	2.3373	2.5864	.1469	2.5720	.2303
18	-14.6020	.7338	.6667	.0064	.3035	1.8318	2.3846	2.7636	.2707	2.7472	.1311
19	-11.4020	.7209	.6451	.0530	.3179	1.7959	2.4410	2.2865	.5542	3.2390	-.0556
20	-9.1020	.7092	.6195	.1492	.3214	1.7512	2.4901	4.5650	1.6565	4.2664	-.6335
21	-4.9000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
22	6.2980	.7564	.6509	.3159	-.0875	1.5244	2.2854	3.9879	3.1313	2.3617	.7219
23	9.4980	.7711	.7214	.2632	-.1385	1.5133	2.2164	3.1617	2.1819	2.1275	.8425
24	12.5930	.7337	.7329	.2259	-.1716	1.5379	2.1575	2.5095	1.3736	1.9715	.7097
25	17.3930	.8304	.7576	.1647	-.1999	1.6035	2.0781	2.2091	.8413	1.9704	.5383
26	29.1270	.8347	.7997	.0634	-.2247	1.7579	1.9035	2.0584	.3976	2.0022	.2647
27	34.4600	.8405	.8115	.0580	-.2338	1.8313	1.8505	2.0405	.3221	2.0069	.1790
28	40.8600	.8579	.8209	.0333	-.2365	1.9021	1.9027	2.0259	.2692	2.0056	.0974
29	47.2500	.8526	.8297	.0131	-.2358	1.9410	1.7675	2.0042	.2307	1.9966	.0342
30	53.6600	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
31	56.8620	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
32	59.9930	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
33	60.0600	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
34	62.1530	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
35	67.5280	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
36	73.9280	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457
37	80.3280	1.0000	1.0000	0.0000	0.0000	1.0000	1.0000	1.0000	.1404	.9890	.0457

Figure A.5.- Continued.

FLOW FIELD AND MAGNETIC FIELD COMPONENTS ALONG TRAJECTORY

(SOLAR WIND COORDINATE SYSTEM)

(DIMENSIONAL, USING INPUT INTERPLANETARY VALUES)

INTERPLANETARY MACH NUMBER = 3.30
RATIO OF SPECIFIC HEATS = 1.6667
INTERPLANETARY VELOCITY = 3.920E+02
INTERPLANETARY DENSITY = 2.020E+01
INTERPLANETARY TEMPERATURE = 1.020E+05

INTERPLANETARY MAGNETIC FIELD
MAGNITUDE = .882E+01
X-COMPONENT = 1.239E+00
Y-COMPONENT = 8.726E+00
Z-COMPONENT = 4.032E-01

N	TIME	/V/	VX	VY	VZ	RHO	TEMP	/B/	BX	BY	BZ
1	-90.8700	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
2	-85.5370	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
3	-80.2030	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
4	-74.8700	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
5	-70.6030	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
6	-64.2030	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
7	-57.8020	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
8	-51.4020	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
9	-40.7350	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
10	-39.6680	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
11	-38.6620	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
12	-35.9350	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
13	-34.8670	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
14	-32.7350	3.137E+02	2.941E+02	-5.627E+01	9.343E+01	4.595E+01	2.121E+05	1.844E+01	-5.400E-01	1.797E+01	4.135E+00
15	-24.2020	3.065E+02	2.859E+02	-3.717E+01	1.039E+02	4.179E+01	2.210E+05	1.954E+01	-1.970E-01	1.951E+01	3.557E+00
16	-21.0620	3.006E+02	2.789E+02	-2.744E+01	1.088E+02	4.055E+01	2.281E+05	2.080E+01	1.729E-01	2.057E+01	3.099E+00
17	-16.7350	2.918E+02	2.678E+02	-9.063E+00	1.157E+02	3.883E+01	2.384E+05	2.292E+01	1.295E+00	2.269E+01	2.032E+00
18	-14.6020	2.877E+02	2.619E+02	2.710E+00	1.190E+02	3.799E+01	2.432E+05	2.438E+01	2.388E+00	2.424E+01	1.157E+00
19	-11.4020	2.826E+02	2.529E+02	1.961E+01	1.246E+02	3.628E+01	2.490E+05	2.930E+01	4.890E+00	2.856E+01	-4.856E-01
20	-8.1020	2.786E+02	2.428E+02	5.847E+01	1.221E+02	3.537E+01	2.541E+05	4.027E+01	1.461E+01	3.711E+01	-5.589E+00
21	-4.9000	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
22	6.2980	2.964E+02	2.673E+02	1.238E+02	-3.232E+01	3.039E+01	2.331E+05	3.518E+01	2.763E+01	2.604E+01	6.369E+00
23	9.4980	3.023E+02	2.789E+02	1.032E+02	-5.427E+01	3.050E+01	2.261E+05	2.789E+01	1.925E+01	1.877E+01	7.433E+00
24	12.5930	3.072E+02	2.973E+02	8.576E+01	-6.693E+01	3.107E+01	2.201E+05	2.210E+01	1.212E+01	1.739E+01	6.262E+00
25	17.3930	3.138E+02	2.970E+02	6.457E+01	-7.797E+01	3.239E+01	2.120E+05	1.949E+01	7.422E+00	1.736E+01	4.749E+00
26	29.1270	3.277E+02	3.135E+02	3.258E+01	-8.965E+01	3.571E+01	1.942E+05	1.816E+01	3.508E+00	1.766E+01	2.335E+00
27	34.4600	3.318E+02	3.181E+02	2.274E+01	-9.164E+01	3.699E+01	1.888E+05	1.800E+01	2.842E+00	1.771E+01	1.579E+00
28	40.8600	3.355E+02	3.222E+02	1.305E+01	-9.271E+01	3.942E+01	1.839E+05	1.787E+01	2.375E+00	1.769E+01	8.597E-01
29	47.2600	3.382E+02	3.252E+02	5.142E+00	-9.245E+01	3.921E+01	1.803E+05	1.768E+01	2.035E+00	1.756E+01	3.615E-01
30	53.6600	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
31	56.8620	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
32	58.9930	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
33	60.0600	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
34	62.1930	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
35	67.5280	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
36	73.9280	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01
37	80.3280	3.920E+02	3.920E+02	0.	0.	2.020E+01	1.020E+05	8.823E+00	1.239E+00	8.726E+00	4.032E-01

TRAJECTORY CALCULATION

(SUN-PLANET COORDINATE SYSTEM)

TRAJECTORY COORDINATES

RG/RPLANET = 1.0331

N	TIME	X/RG	Y/RG	Z/RG	R/RG	X/RPLANET	Y/RPLANET	Z/RPLANET	R/RPLANET
1	-90.8700	-.8160	3.6658	1.9621	4.1579	-.8430	3.7870	2.0270	4.2954
2	-85.5370	-.7405	3.4848	1.9796	4.0078	-.7650	3.6000	2.0450	4.1403
3	-80.2030	-.6621	3.2844	1.9960	3.8434	-.6840	3.3930	2.0620	3.9704
4	-74.8700	-.5808	3.0753	2.0067	3.6721	-.6000	3.1770	2.0730	3.7935
5	-70.6030	-.5159	2.9030	2.0115	3.5318	-.5330	2.9990	2.0780	3.6486
6	-64.2030	-.4162	2.6339	2.0125	3.3148	-.4300	2.7210	2.0790	3.4243
7	-57.8020	-.3136	2.3493	2.0009	3.0859	-.3240	2.4270	2.0670	3.1879
8	-51.4020	-.2101	2.0502	1.9776	2.8496	-.2170	2.1180	2.0430	2.9427
9	-41.7350	-.0329	1.5120	1.8982	2.4268	-.0340	1.5620	1.9610	2.5071
10	-39.6680	-.0165	1.4607	1.8886	2.3875	-.0170	1.5090	1.9510	2.4665
11	-38.6620	-.0097	1.3765	1.8762	2.3221	-.0100	1.4220	1.9320	2.3989
12	-35.9350	.0474	1.2400	1.8411	2.2198	.0490	1.2810	1.9020	2.2932
13	-34.8670	.0649	1.1926	1.8256	2.1806	.0670	1.2320	1.8860	2.2527
14	-32.7350	.1529	1.0725	1.7927	2.0891	.1580	1.1080	1.8520	2.1581
15	-24.2020	.2377	.5640	1.6125	1.7083	.2456	.5826	1.6659	1.7647
16	-21.0020	.2859	.3629	1.5191	1.5618	.2954	.3749	1.5693	1.6135
17	-16.7350	.3446	.0871	1.3639	1.3667	.3560	.0900	1.4090	1.4119
18	-14.6020	.3698	-.0503	1.2729	1.2739	.3820	-.0520	1.3150	1.3160
19	-11.4020	.4017	-.1984	1.1132	1.1307	.4150	-.2050	1.1500	1.1681
20	-8.1020	.4230	-.4627	.9148	1.0251	.4370	-.4780	.9450	1.0590
21	-4.9000	.4279	-.6805	.6399	.9334	.4420	-.7030	.6610	.9643
22	6.2980	.2991	-1.0087	-.2575	1.0410	.3090	-1.0420	-.2660	1.0754
23	9.4980	.2294	-1.0212	-.5285	1.1499	.2370	-1.0550	-.5460	1.1879
24	12.5930	.1549	-1.0048	-.7794	1.2692	.1600	-1.0380	-.8010	1.3111
25	17.3930	.0329	-.9351	-1.1248	1.4627	.0340	-.9660	-1.1620	1.5111
26	29.1270	-.2662	-.6515	-1.8266	1.9393	-.2750	-.6730	-1.8870	2.0034
27	34.4600	-.3940	-.4956	-2.0860	2.1441	-.4070	-.5120	-2.1550	2.2150
28	40.8600	-.5430	-.3020	-2.3658	2.3850	-.5610	-.3120	-2.4440	2.4638
29	47.2600	-.6824	-.1045	-2.5846	2.5867	-.7050	-.1080	-2.6700	2.6722
30	53.6600	-.8170	.0951	-2.8314	2.8330	-.8440	.0982	-2.9250	2.9266
31	56.8620	-.8809	.1946	-2.9340	2.9405	-.9100	.2010	-3.0310	3.0377
32	58.9930	-.9235	.2604	-2.9979	3.0092	-.9540	.2690	-3.0970	3.1097
33	60.0600	-.9438	.2933	-3.0308	3.0450	-.9750	.3030	-3.1310	3.1456
34	62.1930	-.9854	.3591	-3.0928	3.1135	-1.0180	.3710	-3.1950	3.2165
35	67.5280	-1.0861	.5237	-3.2399	3.2819	-1.1220	.5410	-3.3470	3.3904
36	73.9280	-1.2013	.7154	-3.4016	3.4760	-1.2410	.7390	-3.5140	3.5909
37	80.3280	-1.3136	.9090	-3.5535	3.6679	-1.3570	.9390	-3.6710	3.7892

Figure A.5.- Continued.

FLOW FIELD AND MAGNETIC FIELD COMPONENTS ALONG TRAJECTORY

(SUN-PLANET COORDINATE SYSTEM)

(NON-DIMENSIONALIZED BY INTERPLANETARY VALUES)

N	TIME	V/VINF	VX/VINF	VY/VINF	VZ/VINF	RHO/RHOINF	TEMP/TEMPINF	B/BINF	BX/BINF	BY/BINF	BZ/BINF
1	-99.8700	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
2	-95.5370	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
3	-80.2030	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
4	-74.9700	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
5	-70.6130	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
6	-64.2030	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
7	-57.9120	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
8	-51.4120	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
9	-40.7350	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
10	-39.6660	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
11	-38.6020	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
12	-35.9350	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
13	-34.8570	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
14	-32.7350	.8307	-.7414	.1605	.2364	2.2747	2.1790	2.0905	-.0573	-2.0364	.4688
15	-24.2020	.7418	-.7146	.1367	.2681	2.0497	2.1682	2.2495	-.1061	-2.2095	.4032
16	-21.0620	.7648	-.7170	[illegible]	.2757	2.0072	2.2959	2.3578	-.1547	-2.3261	.3512
17	-16.7350	.7648	-.6814	.1624	.2932	1.9225	2.3373	2.5864	-.2952	-2.5592	.2300
18	-14.6020	.7339	-.6681	.1315	.3015	1.8835	2.3746	2.7636	-.4287	-2.7270	.1304
19	-13.4020	.7230	-.6477	-.0127	.3162	1.7959	2.4416	3.2865	-.7396	-3.2617	-.0565
20	-8.1020	.7097	-.6274	-.1132	.3096	1.7512	2.4921	4.5652	-1.8942	-4.1041	-.6378
21	-4.9600	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000
22	6.2980	.7567	-.6987	-.2761	-.0842	1.5044	2.2854	3.9873	-3.2639	-2.1774	.7137
23	9.4980	.7721	-.7230	-.2213	-.1403	1.5120	2.2164	3.1617	-2.3029	-1.9982	.8368
24	12.5930	.7837	-.7428	-.1763	-.1727	1.5379	2.1575	2.5055	-1.4857	-1.6891	.7061
25	17.3930	.8034	-.7653	-.1209	-.2019	1.6135	2.0781	2.2091	-.9547	-1.9186	.5361
26	29.1270	.8636	-.8026	-.0372	-.2336	1.7679	1.9035	2.0594	-.5129	-1.9759	.2636
27	34.4600	.8455	-.8124	-.0112	-.2359	1.8313	1.8505	2.0479	-.4376	-1.9850	.1782
28	40.8600	.8559	-.8218	.0140	-.2337	1.9021	1.8027	2.0259	-.3844	-1.9668	.0967
29	47.2600	.8926	-.8284	.0345	-.2384	1.9412	1.7675	2.0047	-.3459	-1.9740	.0336
30	53.6600	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
31	56.8620	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
32	58.9930	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
33	60.0600	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
34	62.1930	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
35	67.5280	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
36	73.9280	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453
37	97.3280	1.0000	-.9983	.0576	-.0026	1.0000	1.0000	1.0000	-.1972	-.9793	.0453

FLOW FIELD AND MAGNETIC FIELD COMPONENTS ALONG TRAJECTORY

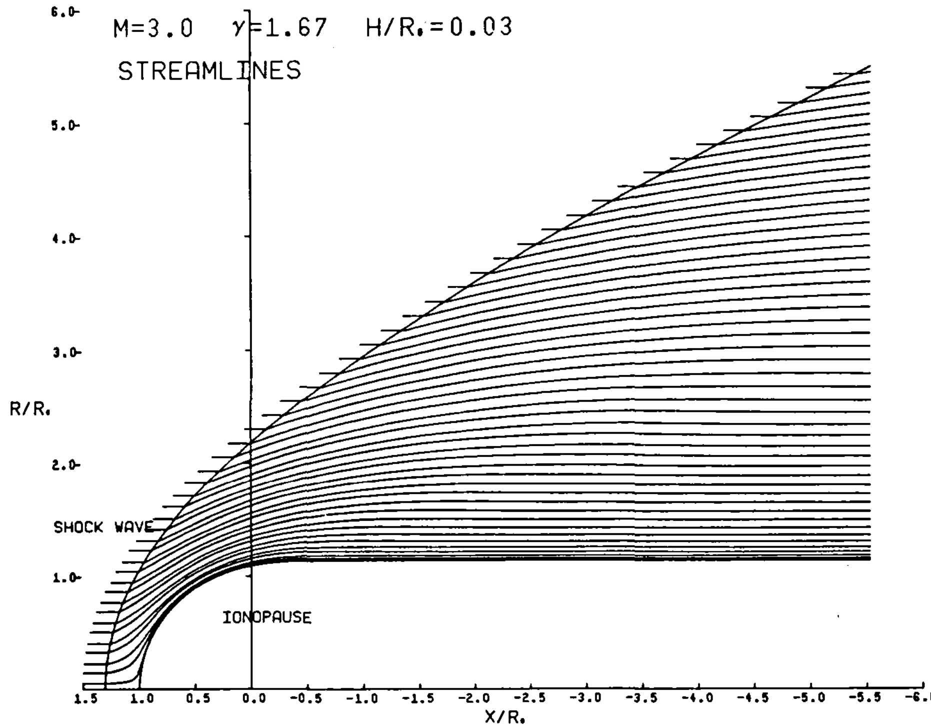
(SUN-PLANET COORDINATE SYSTEM)

(DIMENSIONAL, USING INPUT INTERPLANETARY VALUES)

INTERPLANETARY MACH NUMBER	= [illegible]	INTERPLANETARY MAGNETIC FIELD		
RATIO OF SPECIFIC HEATS	= 1.6667	MAGNITUDE	=	8.823E+00
INTERPLANETARY VELOCITY	= 3.920E+02	X-COMPONENT	=	-1.740E+00
INTERPLANETARY DENSITY	= 2.020E+01	Y-COMPONENT	=	-8.640E+00
INTERPLANETARY TEMPERATURE	= 1.020E+05	Z-COMPONENT	=	4.000E-01
AZIMUTHAL ANGLE	= [illegible]			
POLAR ANGLE	= [illegible]			

N	TIME	|V|	VX	VY	VZ	RHO	TEMP	|B|	BX	BY	BZ
1	-99.8700	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
2	-95.5370	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
3	-80.2030	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
4	-74.9700	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
5	-70.6130	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
6	-64.2030	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
7	-57.9120	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
8	-51.4120	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
9	-40.7350	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
10	-39.6660	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
11	-38.6020	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
12	-35.9350	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
13	-34.8570	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
14	-32.7350	3.257E+02	-2.906E+02	7.312E+00	9.266E+01	4.595E+01	2.121E+05	1.844E+01	-5.059E-01	-1.797E+01	4.136E+00
15	-24.2020	[illegible]	-2.836E+02	5.359E+01	1.325E+02	4.179E+01	2.212E+05	1.984E+01	-9.363E-01	-1.949E+01	3.557E+00
16	-21.0620	[illegible]	-2.772E+02	4.316E+01	1.081E+02	4.055E+01	2.282E+05	2.080E+01	-1.365E+00	-2.052E+01	3.099E+00
17	-16.7350	2.919E+02	-2.671E+02	2.444E+01	1.149E+02	3.883E+01	2.384E+05	2.282E+01	-2.604E+00	-2.258E+01	2.029E+00
18	-14.6020	2.877E+02	-2.619E+02	1.239E+01	1.182E+02	3.799E+01	2.432E+05	2.438E+01	-3.782E+00	-2.406E+01	1.150E+00
19	-13.4020	2.826E+02	-2.539E+02	-4.999E+00	1.239E+02	3.628E+01	2.490E+05	2.900E+01	-6.525E+00	-2.625E+01	-4.984E-01
20	-8.1020	2.780E+02	-2.461E+02	-4.438E+01	1.214E+02	3.537E+01	2.541E+05	4.027E+01	-1.671E+01	-3.621E+01	-5.627E+00
21	-4.9600	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
22	6.2980	2.966E+02	-2.739E+02	-1.082E+02	-3.302E+01	3.039E+01	2.331E+05	3.518E+01	-2.880E+01	-1.921E+01	6.297E+00
23	9.4980	3.023E+02	-2.842E+02	-8.598E+01	-5.500E+01	3.054E+01	2.261E+05	2.789E+01	-2.032E+01	-1.763E+01	7.383E+00
24	12.5930	3.072E+02	-2.916E+02	-6.909E+01	-6.769E+01	3.107E+01	2.201E+05	2.210E+01	-1.312E+01	-1.667E+01	6.230E+00
25	17.3930	3.138E+02	-3.000E+02	-4.739E+01	-7.975E+01	3.259E+01	2.120E+05	1.949E+01	-8.423E+00	-1.693E+01	4.730E+00
26	29.1270	3.277E+02	-3.146E+02	-1.460E+01	-9.147E+01	3.571E+01	1.942E+05	1.816E+01	-4.525E+00	-1.743E+01	2.326E+00
27	34.4600	3.314E+02	-3.186E+02	-4.439E+00	-9.247E+01	3.699E+01	1.888E+05	1.800E+01	-3.861E+00	-1.751E+01	1.572E+00
28	40.8600	3.355E+02	-3.221E+02	5.514E+00	-9.155E+01	3.842E+01	1.839E+05	1.737E+01	-3.392E+00	-1.753E+01	8.535E-01
29	47.2600	3.382E+02	-3.247E+02	1.357E+01	-9.330E+01	3.921E+01	1.803E+05	1.769E+01	-3.744E+00	-1.742E+01	2.962E-01
30	53.6600	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
31	56.8620	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
32	58.9930	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
33	60.0600	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
34	62.1930	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
35	67.5280	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
36	73.9280	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01
37	97.3280	3.920E+02	-3.913E+02	2.257E+01	-1.026E+00	2.020E+01	1.020E+05	8.823E+00	-1.740E+00	-8.640E+00	4.000E-01

Figure A.5.- Concluded.



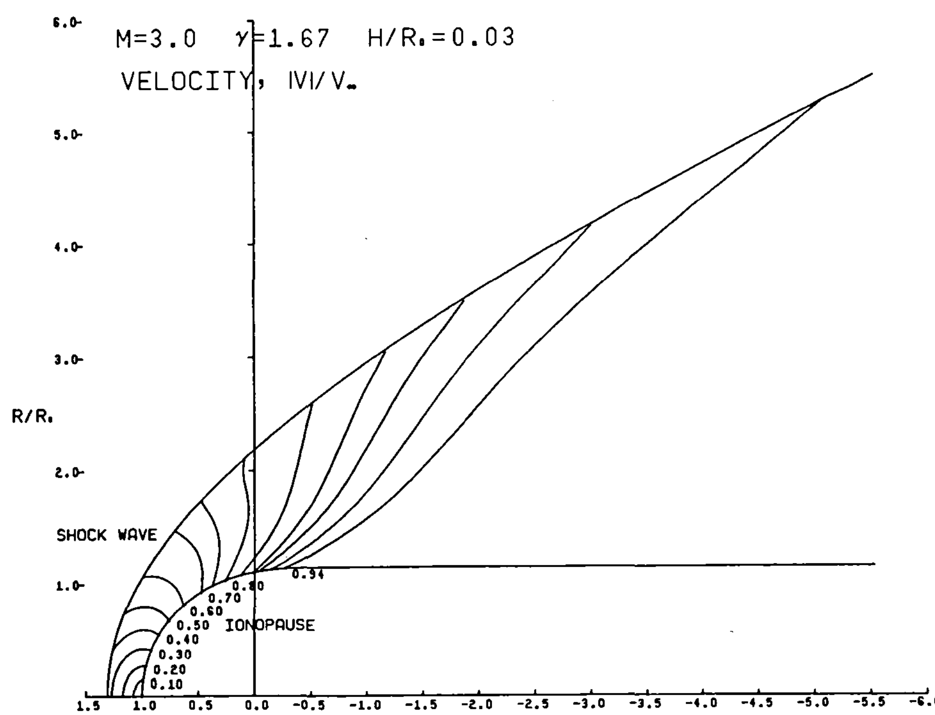
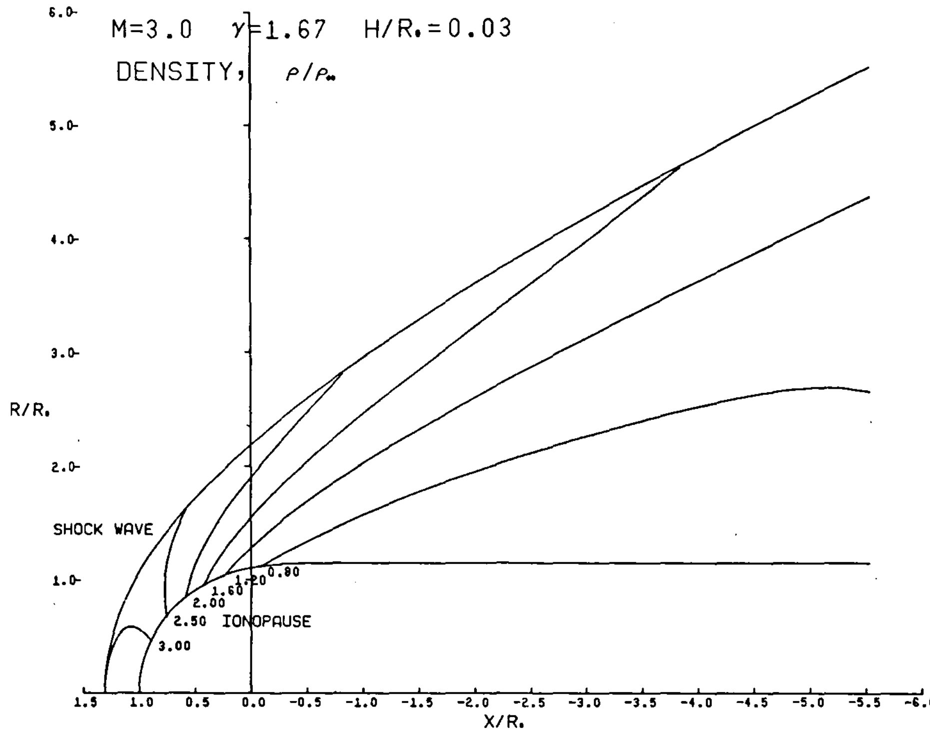


Figure A.6.- Plot output for sample case.



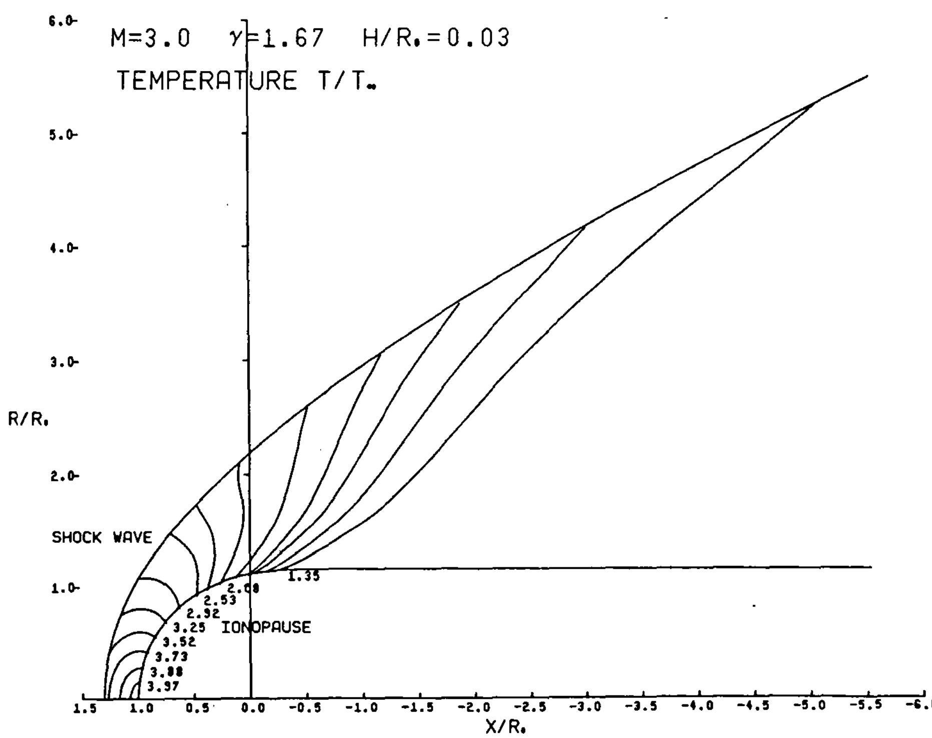
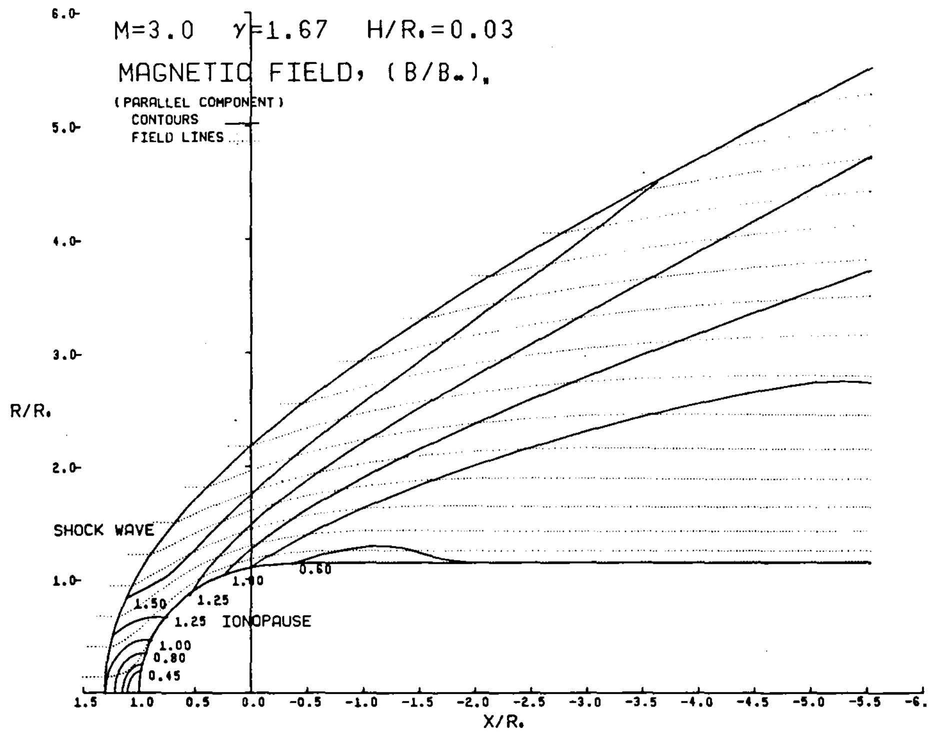


Figure A.6.- Continued



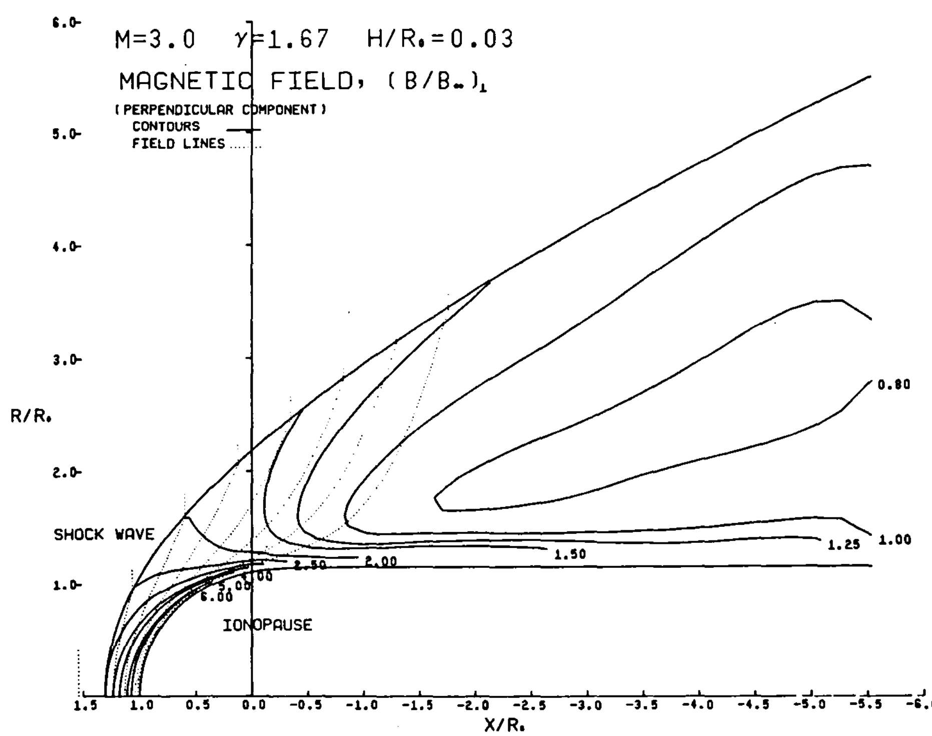


Figure A.6.- Continued

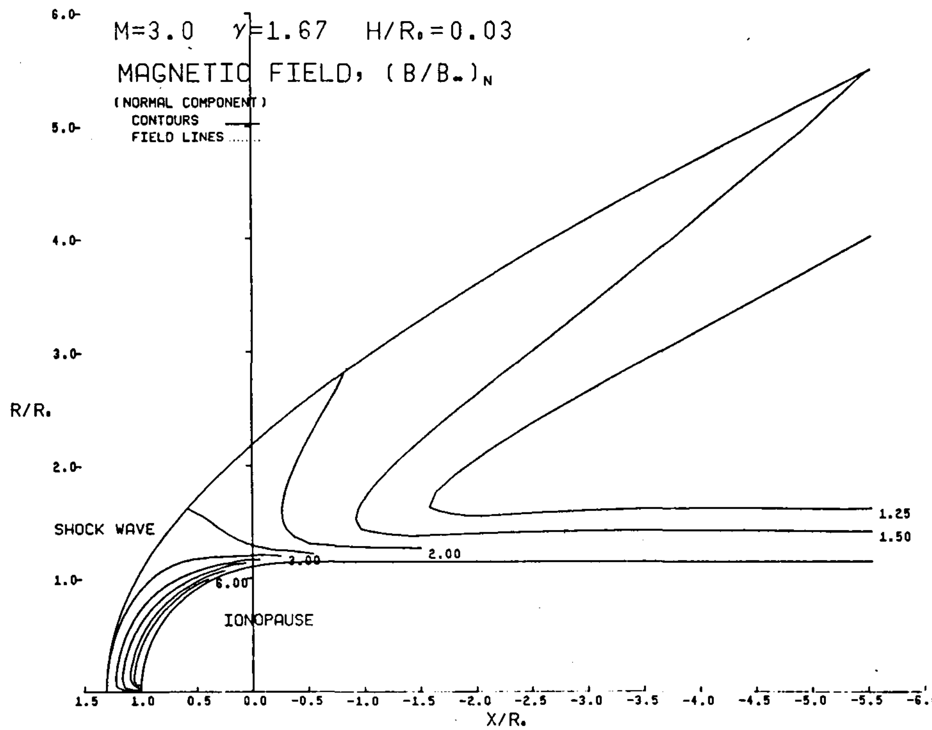


Figure A.6.- Continued

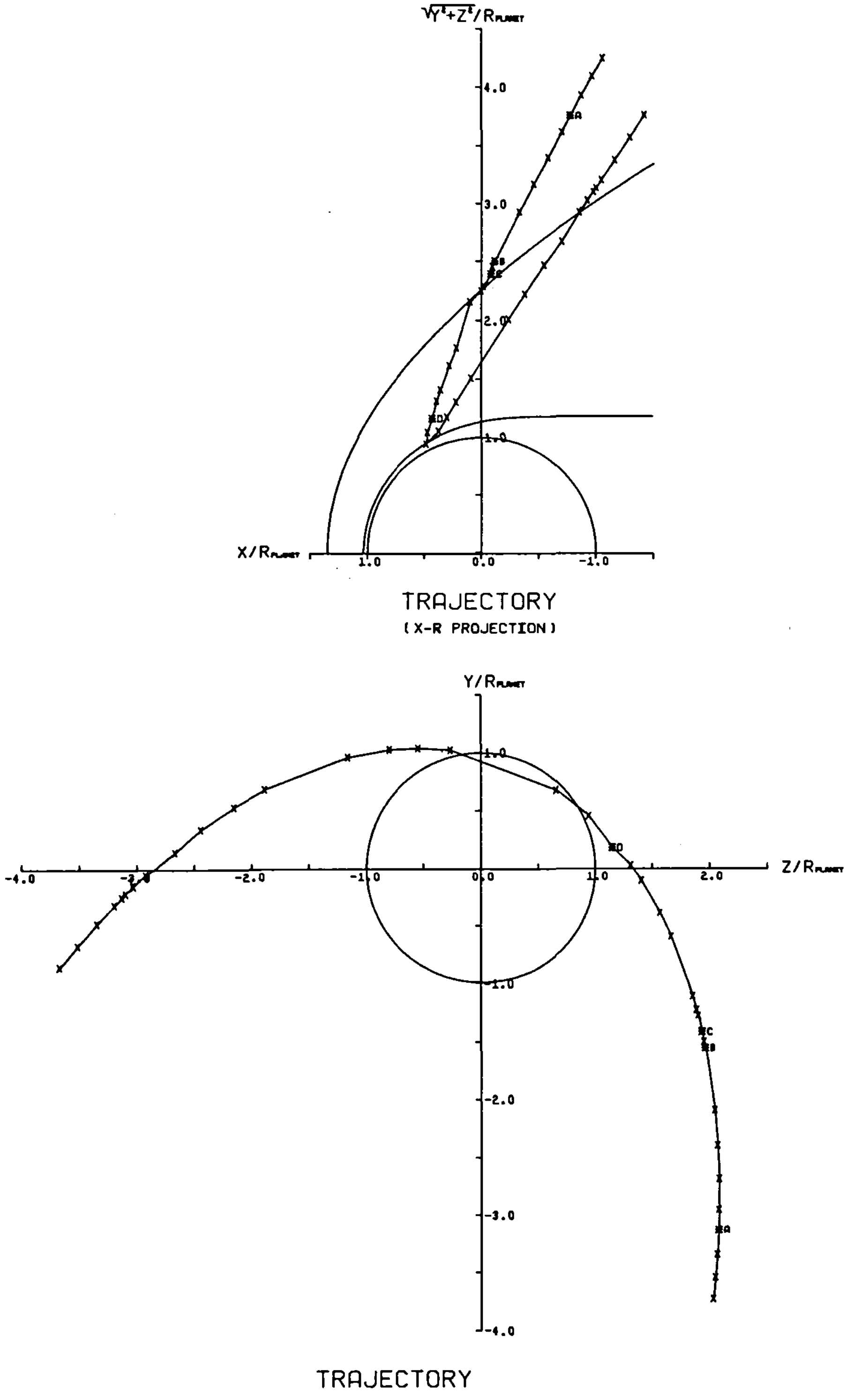
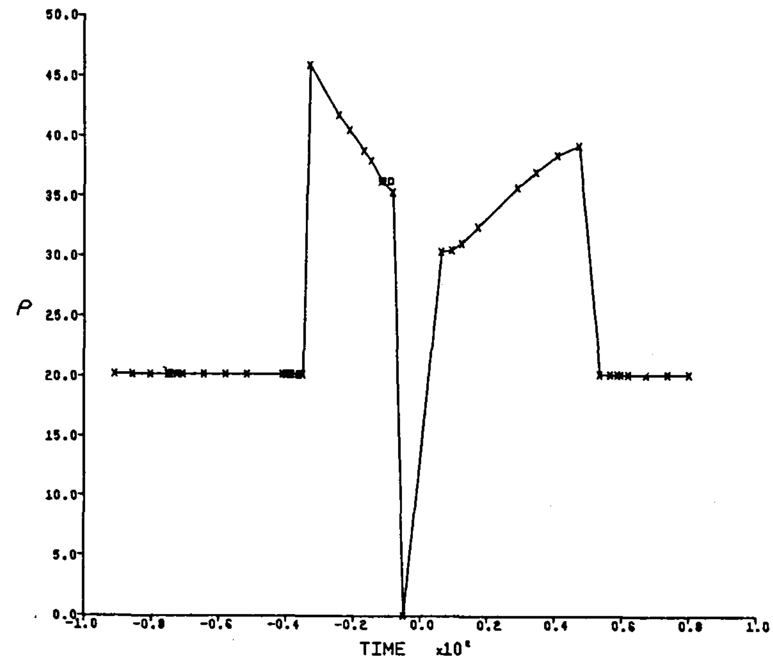


Figure A.6.- Continued

DENSITY vs TIME



TEMPERATURE vs TIME

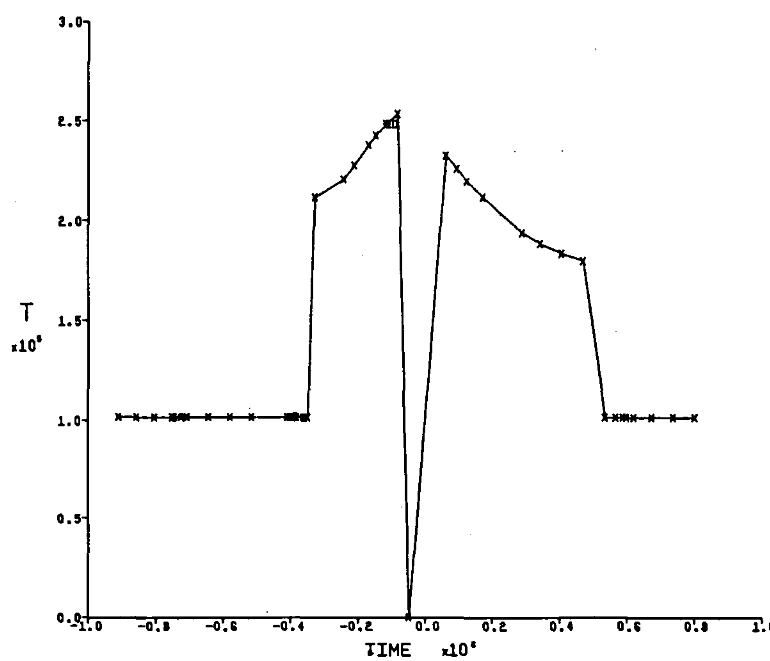
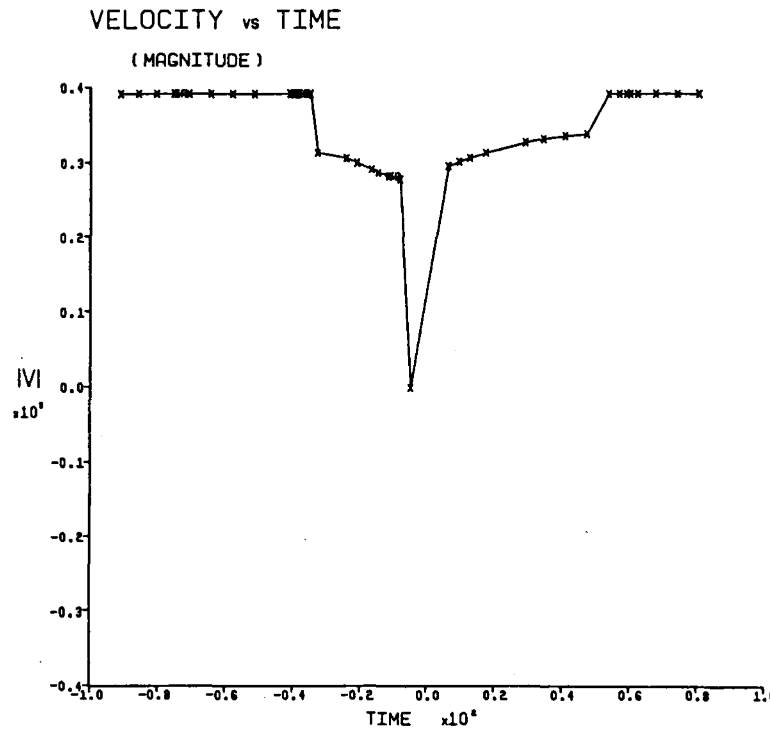


Figure A.6.- Continued.



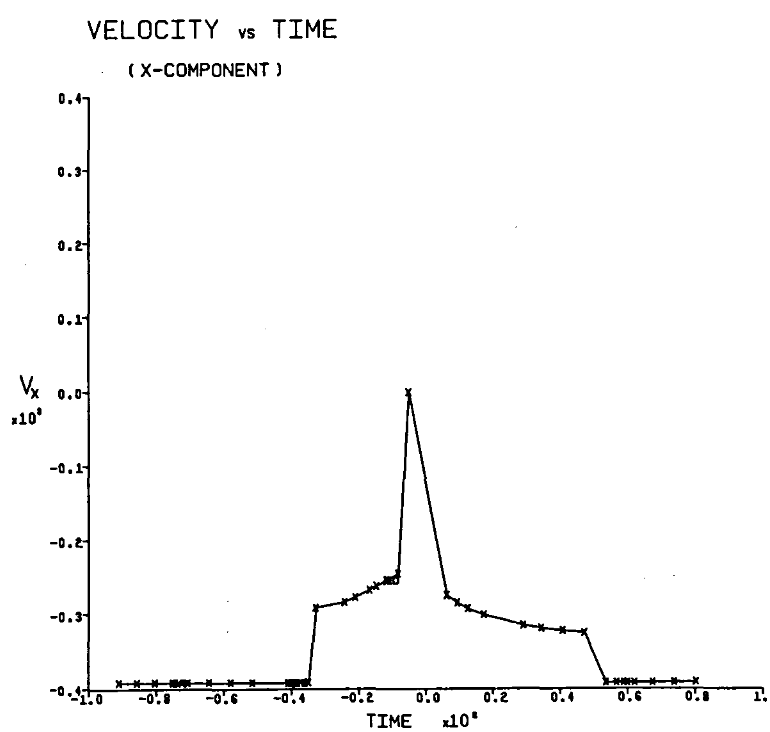
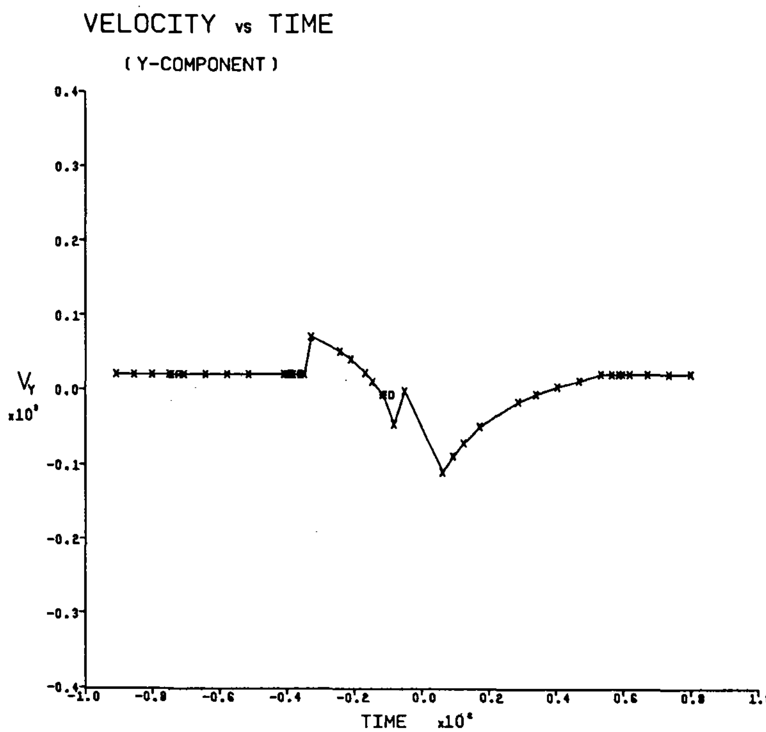


Figure A.6.- Continued



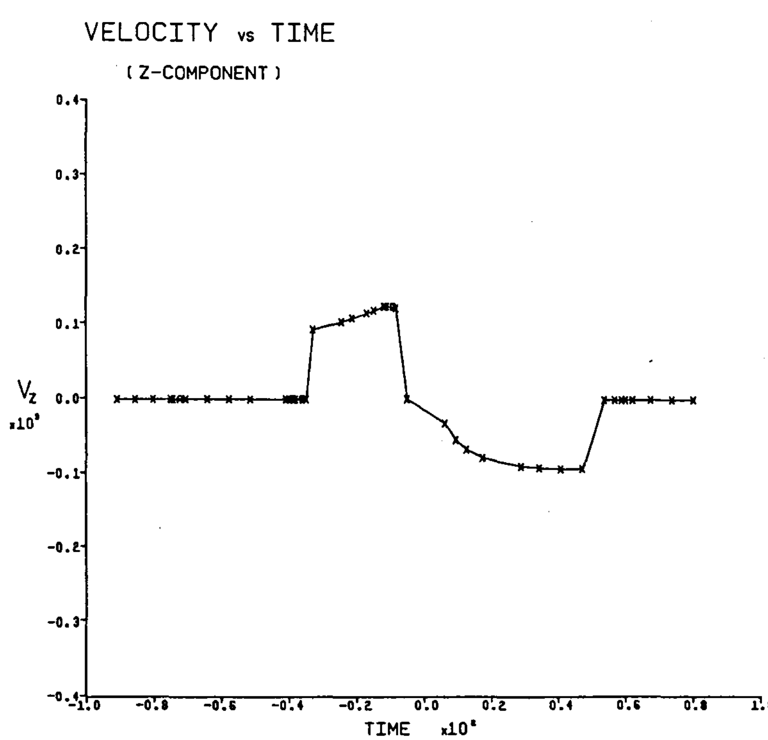
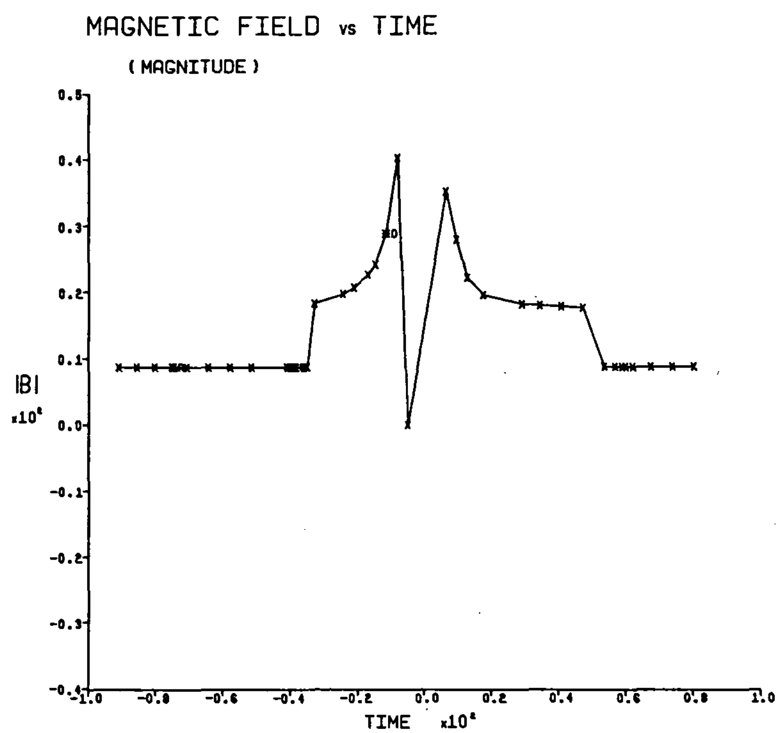


Figure A.6.- Continued.



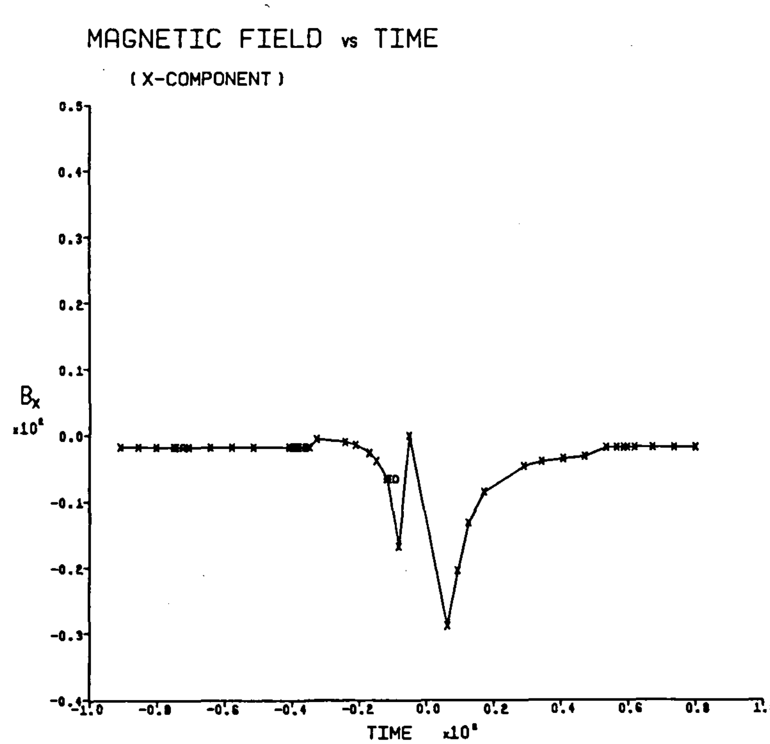


Figure A.6.- Continued.

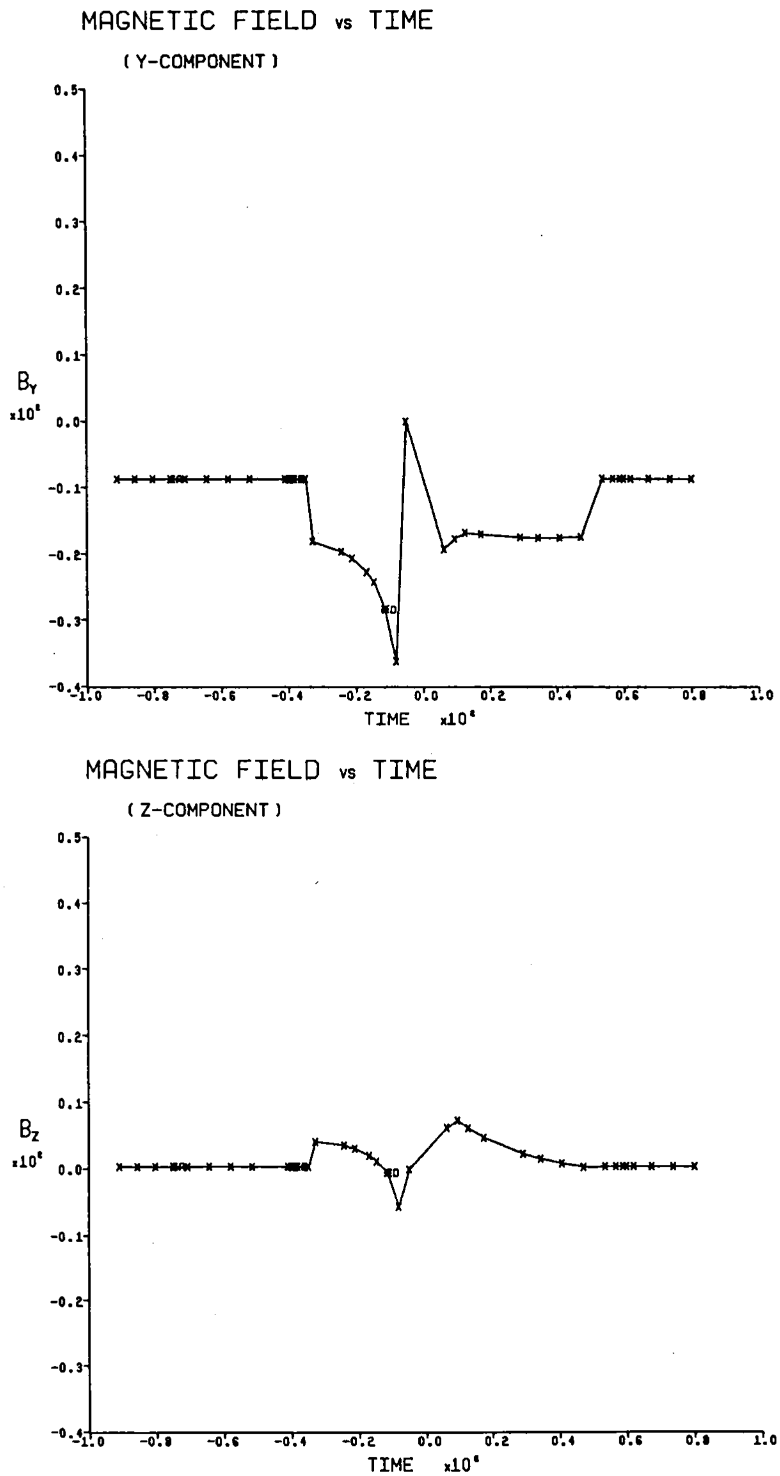


Figure A.6.- Concluded.

APPENDIX B

LISTING OF COMPUTER PROGRAM

```
      PROGRAM MAIN (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,
     *              TAPE1,TAPE4,TAPE9)
      LOGICAL LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      COMMON /PROPT/ LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
C
      WRITE(6,200)
      CALL ECHINP
      READ(5,100) NCASE
      DO 20 ICASE=1,NCASE
      CALL INPUT
      IF (LRERUN) GO TO 10
      CALL BLUNTB
      CALL MARCH
      GO TO 15
   10 CALL RERUN
   15 CONTINUE
      CALL FLOWST
      IF (LPRFL) CALL FLOUT
      IF (LPRST) CALL STOUT
      CALL SCOMP
      IF (LPRB) CALL BOUT
      CALL CONTUR
      IF (LTRAJ) CALL TRAJEC
   20 CONTINUE
      STOP
C
  100 FORMAT(I10)
  200 FORMAT(1H1,//////////,33X,62(1H*)/33X,62(1H*)/33X,2H**,23X,
     *       13HPROGRAM SOLAR,22X,2H**,/33X,2H**,58X,2H**/,33X,2H**,24X,
     *       10HDETERMINES,24X,2H**,/,33X,2H**,58X,2H**/,33X,2H**,4X,
     *       50HSOLAR WIND FLOW PAST PLANETARY MAGNETO/IONOSPHERES,4X,
     *       2H**,/,33X,2H**,58X,2H**,/,33X,2H**,27X,5HUSING,26X,2H**,/,
     *       33X,2H**,58X,2H**,/33X,2H**,6X,47HFULLY CONSERVATIVE FINITE
     * DIFFERENCE ALGORITHMS,5X,2H**,/,33X,2H**,58X,2H**,/,33X,2H**,58X,
     *       2H**,/,33X,2H**,58X,2H**,/,33X,2H**,24X,10HWRITTEN BY,24X,
     *2H**,/,33X,2H**,58X,2H**,/,33X,2H**,9X,41HSTEPHEN S. STAHARA, BARB
     *ARA C. TRUDINGER,,5X,2H**,/,33X,2H**,58X,2H**,/,33X,2H**,20X,
     *       20HAND DANIEL J. KLENKE,18X,2H**,/,33X,2H**,58X,2H**,/,33X,
     *       2H**,58X,2H**,/,33X,2H**,58X,2H**,/,33X,2H**,10X,38HNIELSEN
     * ENGINEERING AND RESEARCH, INC.,10X,2H**,/,33X,2H**,58X,2H**,/,33X
     *       ,2H**,17X,25HMOUNTAIN VIEW, CALIFORNIA,16X,2H**,/,33X,2H**,
     *       58X,2H**,/,33X,2H**,58X,2H**,/,33X,62(1H*),/,33X,62(1H*))
      END
```

```
      SUBROUTINE ANGEL(ELLINF,GAMEL,I,J)
C
C  THIS SUBROUTINE INTERPOLATES FOR EL/ELINF AND GAM AT THE (I,J) GRID
C         POINT FROM THE ARRAYS ELBF AND GAMBF
C
      LEVEL 2,NB,NBF,XBF,RBF,ELBF,GAMBF
      COMMON /BVAL/ NB,NBF(51),XBF(51,100),RBF(51,100),ELBF(51,100),
     *  GAMBF(51,100)
      COMMON /SHOCKS/ DRSDX(100),DST(50)
      LEVEL 2, XST,YST,NUMST,NST
      COMMON /STREAM/ XST(50,152),YST(50,152),NUMST(50),NST
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NXPLOT
      DIMENSION XO(4),RO(4),ELO(4),GAMO(4)
C
C     IF POINT IS ON SHOCK, USE FORMULAE AND SAVE RESULTS
C
      IF (I .LT. NRMAX) GO TO 100
      IF (J .GT. 1) GO TO 10
      EL=ELBF(1,2)
      ANG=GAMBF(1,2)
      GO TO 40
   10 THET=ATAN(DRSDX(J))
      S2=SIN(THET)**2
      EM2=AMACH**2
      GAM2=(GAMMA+1.0)*0.5*EM2
      DD2=1.0-(EM2*S2-1.0)*(GAMMA*EM2*S2+1.0)/(GAM2**2*S2)
      DD=SQRT(DD2)
      COTH=1.0/DRSDX(J)
      TD=GAM2/(EM2*S2-1.0)-1.0
      DELT=ATAN(COTH/TD)
      EL=SQRT(1.0+COTH*COTH*(1.0+DD2)-2.0*DD*(TD+1.0)*COTH*SIN(DELT))
      ANG=THET+ASIN(DD*COTH*SIN(THET-DELT)/EL)
   40 ELLINF=EL
      GAMEL=ANG
      RETURN
C
C     IF POINT IS ON AXIS USE LINEAR INTERPOLATION
C
  100 X=XC(I,J)
      Y=YC(I,J)
      IF (J .GT. 1) GO TO 200
      N=NBF(1)
      DO 110 K=1,N
      IF (X .LT. XBF(1,K)) GO TO 120
  110 CONTINUE
  120 ELLINF=ELBF(1,K-1)+(X-XBF(1,K-1))*(ELBF(1,K)-ELBF(1,K-1))
     *          /(XBF(1,K)-XBF(1,K-1))
      GAMEL=ANG
      RETURN
C
C     IF POINT IS AT X=ZPLOT, USE LINEAR INTERPOLATION
C
  200 IF (J .LT. NXMAX) GO TO 300
      NIM=NBF(3)+1
      DO 210 IM=3,NST
      IN=IM+1
      N=NBF(IN)+1
      IF (Y .LT. RBF(IN,N)) GO TO 220
      NIM=N
  210 CONTINUE
      ELLINF=ELBF(IN,N)+(Y-RBF(IN,N))*(EL-ELBF(IN,N))
     *          /(YC(NRMAX,NXMAX)-RBF(IN,N))
      GAMEL=GAMBF(IN,N)+(Y-RBF(IN,N))*(ANG-GAMBF(IN,N))
     *          /(YC(NRMAX,NXMAX)-RBF(IN,N))
      RETURN
  220 ELLINF=ELBF(IM,NIM)+(Y-RBF(IM,NIM))*(ELBF(IN,N)-ELBF(IM,NIM))
     *          /(RBF(IN,N)-RBF(IM,NIM))
      GAMEL=GAMBF(IM,NIM)+(Y-RBF(IM,NIM))*(GAMBF(IN,N)-GAMBF(IM,NIM))
     *         /(RBF(IN,N)-RBF(IM,NIM))
      RETURN
C
C     INTERIOR POINT - USE QUADRILATERAL INTERPOLATION
C
  300 N1=NBF(1)
      NMAX=NBF(3)+1
      DO 600 IST=1,NST
      N2=NBF(IST+1)+1
      IF (N2 .GT. NMAX) N2=NMAX
      N=MINO(N1,N2)
      DO 310 JJ=1,N
      IF (X .LT. XBF(IST+1,JJ)) GO TO 340
  310 CONTINUE
      GO TO 560
C
C     FIND QUADRILATERAL WHICH CONTAINS POINT
C
  340 IF (Y .GT. RBF(IST+1,JJ)) GO TO 580
      IF (XBF(IST+1,JJ) .LT. XST(IST,1)) GO TO 520
      SLOPE=(RBF(IST+1,JJ)-Y)/(XBF(IST+1,JJ)-X)
      SLOPE1=(RBF(IST+1,JJ)-RBF(IST+1,JJ-1))
     *          /(XBF(IST+1,JJ)-XBF(IST+1,JJ-1))
      IF (SLOPE1 .GT. SLOPE) GO TO 580
      IF (JJ .EQ. N) GO TO 350
      SLOPE2=(RBF(IST+1,JJ)-RBF(IST,JJ))/(XBF(IST+1,JJ)-XBF(IST,JJ))
      IF (SLOPE2 .LT. SLOPE) GO TO 390
  350 CONTINUE
      XO(1)=XBF(IST+1,JJ)
      RO(1)=RBF(IST+1,JJ)
      ELO(1)=ELBF(IST+1,JJ)
      GAMO(1)=GAMBF(IST+1,JJ)
      IF (XBF(IST+1,JJ-1) .LT. XST(IST,1)) GO TO 540
      XO(4)=XBF(IST+1,JJ-1)
      RO(4)=RBF(IST+1,JJ-1)
      ELO(4)=ELBF(IST+1,JJ-1)
      GAMO(4)=GAMBF(IST+1,JJ-1)
      IF (JJ .EQ. N) GO TO 430
  360 IF (IST .EQ. 1) GO TO 370
      IF (XBF(IST,JJ-1) .LT. XST(IST-1,1)) GO TO 560
  370 XO(3)=XBF(IST,JJ-1)
      RO(3)=RBF(IST,JJ-1)
      ELO(3)=ELBF(IST,JJ-1)
      GAMO(3)=GAMBF(IST,JJ-1)
  380 XO(2)=XBF(IST,JJ)
      RO(2)=RBF(IST,JJ)
      ELO(2)=ELBF(IST,JJ)
      GAMO(2)=GAMBF(IST,JJ)
```

```
      ELLINF=QUAD(XO,RO,ELO,X,Y)
      GAMEL=QUAD(XO,RO,GAMO,X,Y)
      RETURN
  390 CONTINUE
      IF (SLOPE2 .LT. 0.0) GO TO 350
      IF (XBF(IST,JJ-1) .GT. X) GO TO 500
      JJ=JJ+1
      IF (JJ .LE. N) GO TO 340
C
C     DOWNSTREAM BOUNDARY CUTS QUADRILATERAL CONTAINING POINT
C
      IF (XBF(IST+1,JJ-1) .LT. XST(IST,1)) GO TO 460
      XO(4)=XBF(IST+1,JJ-1)
      RO(4)=RBF(IST+1,JJ-1)
      ELO(4)=ELBF(IST+1,JJ-1)
      GAMO(4)=GAMBF(IST+1,JJ-1)
  410 IF (IST .GT. NST) GO TO 470
      XO(1)=XBF(IST+1,JJ)
      RO(1)=RBF(IST+1,JJ)
      ELO(1)=ELBF(IST+1,JJ)
      GAMO(1)=GAMBF(IST+1,JJ)
  430 IF (XBF(IST,JJ-1) .LT. XST(IST-1,1)) GO TO 490
      XO(3)=XBF(IST,JJ-1)
      RO(3)=RBF(IST,JJ-1)
      ELO(3)=ELBF(IST,JJ-1)
      GAMO(3)=GAMBF(IST,JJ-1)
  450 NS=NBF(IST)+1
      XO(2)=XBF(IST,NS)
      RO(2)=RBF(IST,NS)
      ELO(2)=ELBF(IST,NS)
      GAMO(2)=GAMBF(IST,NS)
      ELLINF=QUAD(XO,RO,ELO,X,Y)
      GAMEL=QUAD(XO,RO,GAMO,X,Y)
      RETURN
  460 XO(4)=XSHK(J)
      RO(4)=YSHK(J)
      ELO(4)=EL
      GAMO(4)=ANG
      GO TO 410
  470 XO(1)=ZPLOT
      CALL BSHK1(XO(1),RO(1),GAMO(1),ELO(1))
      GO TO 430
  490 XO(3)=XST(IST-1,1)
      CALL BSHK1(XO(3),RO(3),GAMO(3),ELO(3))
      GO TO 450
C
C     QUADRILATERAL CONTAINING POINT IS REFLEXIVE
C
  500 JJ=JJ-1
      SLOPE=(Y-RBF(IST,JJ-1))/(X-XBF(IST,JJ-1))
      SLOPE1=(RBF(IST+1,JJ-1)-RBF(IST,JJ-1))
     *        /(XBF(IST+1,JJ-1)-XBF(IST,JJ-1))
      IF (SLOPE1 .LT. SLOPE) GO TO 500
      SLOPE2=(RBF(IST,JJ)-RBF(IST,JJ-1))/(XBF(IST,JJ)-XBF(IST,JJ-1))
      IF (SLOPE2 .GT. SLOPE) GO TO 500
      GO TO 350
C
C     POINT IS CLOSE TO SHOCK - NEED VALUES ON SHOCK
C
  520 XO(1)=XBF(IST+1,JJ)
      CALL BSHK1(XO(1),RO(1),GAMO(1),ELO(1))
      SLOPE=(RO(1)-Y)/(XO(1)-X)
      SLOPE2=(RO(1)-RBF(IST,JJ))/(XO(1)-XBF(IST,JJ))
      IF (SLOPE2 .LT. SLOPE) GO TO 390
  540 XO(4)=XSHK(J)
      RO(4)=YSHK(J)
      ELO(4)=EL
      GAMO(4)=ANG
      GO TO 360
  560 XO(3)=XST(IST-1,1)
      CALL BSHK1(XO(3),RO(3),GAMO(3),ELO(3))
      GO TO 390
  590 N1=N2
  600 CONTINUE
C
C           PROGRAM SHOULD NEVER REACH THIS CONDITION
C
      WRITE(6,1000)
      STOP
 1000 FORMAT(1H1,10X,14HERROR IN ANGEL//5X,
     * 35HPROBABLE CAUSE - XPLOT IS TOO LARGE)
      END
```

```
      SUBROUTINE BCOMP
C
C     THIS SUBROUTINE CALCULATES THE COMPONENTS OF THE MAGNETIC FIELD
C     PARALLEL, PERPENDICULAR AND NORMAL TO THE FLOW.
C
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      LEVEL 2, BPARA,BPERP,BNORM,BMAG,BANG
      COMMON /BCOMPS/ BPARA(20,100),BPERP(20,100),BNORM(20,100),
     *  BMAG(20,100),BANG(20,100)
      LOGICAL LPERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      COMMON /PROPT/ LPERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      COMMON /BIN/ ANGP,ANGN,KBCON,BCON(20)
      DIMENSION S(100,6),W(100),A(6),XLSQ(100),YLSQ(100)
      DATA W/100*1.0/
C
C     CALCULATE PERPENDICULAR FIELD LINES
C
      IF (KBCON.EQ.0 .AND. .NOT.LPRB .AND. .NOT.LTRAJ) RETURN
      CALL BSTEP
      CALL BELGAM
C
C     CALCULATE R/RINF AND EL/ELINF AT EACH GRID POINT, THEN SMOOTH
C     ALONG CONSTANT-I LINES, USING FIFTH ORDER LEAST SQUARES FIT
C
      NRM=NRMAX-1
      NXM=NXMAX-1
      DO 10 J=1,NXMAX
      BNORM(NRMAX,J)=RRINF(NRMAX,J)
      CALL ANGEL(ELLINF,SLOPEL,NRMAX,J)
      BPERP(NRMAX,J)=ELLINF
      BANG(NRMAX,J)=SLOPEL
      DO 10 I=2,NRM
      BNORM(I,J)=RRINF(I,J)
      CALL ANGEL(ELLINF,SLOPEL,I,J)
      BPERP(I,J)=ELLINF
      BANG(I,J)=SLOPEL
   10 CONTINUE
      XLSQ(1)=0.0
      DO 60 I=2,NRM
      DO 20 J=1,NXM
      XLSQ(J+1)=XLSQ(J)+SQRT((YC(I,J+1)-YC(I,J))**2
     *              +(XC(I,J+1)-XC(I,J))**2)
   20 CONTINUE
      DO 40 J=1,NXMAX
      YLSQ(J)=BPERP(I,J)
   40 CONTINUE
      CALL FLSQFY(NXMAX,5,XLSQ,YLSQ,W,100,S,A,IER)
      DO 50 J=1,NXMAX
      XA=XLSQ(J)
      BPERP(I,J)=((((A(6)*XA+A(5))*XA+A(4))*XA+A(3))*XA+A(2))*XA+A(1)
   50 CONTINUE
   60 CONTINUE
C
C     CALCULATE COMPONENTS OF MAGNETIC FIELD - PARALLEL, PERPENDICULAR,
C     AND NORMAL TO DIRECTION FLOW
C
      DO 70 J=1,NXMAX
      BPARA(1,J)=VF(1,J)*RHOF(1,J)
      DO 70 I=2,NRMAX
      BPARA(I,J)=VF(I,J)*RHOF(I,J)
      BPERP(I,J)=BPERP(I,J)*RHOF(I,J)
      BNORM(I,J)=BNORM(I,J)*RHOF(I,J)
   70 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE BELGAM
      LEVEL 2,NB,NBF,XBF,RBF,ELBF,GAMBF
      COMMON /BVAL/ NB,NBF(51),XBF(51,100),RBF(51,100),ELBF(51,100),
     *  GAMBF(51,100)
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NXPLOT
      LEVEL 2, XST,YST,NUMST,NST
      COMMON /STREAM/ XST(50,152),YST(50,152),NUMST(50),NST
      COMMON /SHOCKS/ DRSDX(100),DST(50)
      DATA PION2 /1.5707961327/
C
```

```
C      THIS SUBROUTINE CALCULATES THE MAGNITUDE AND DIRECTION OF
C      EL/ELINE AT THE POINTS WHERE THE STREAMLINES INTERSECT THE
C      MAGNETIC FIELD LINES WHICH ARE PERPENDICULAR TO THE FLOW
C      IN FREESTREAM
C
       DST(1)=0.5/YST(1,1)
       DO 10 I=2,NST
       DST(I)=1.0/(YST(I,1)-YST(I-1,1))
    10 CONTINUE
C
C      SET ARRAYS TO FREE STREAM VALUES
C
       NSTP1=NST+1
       JBFMAX=NBF(1)
       DO 30 I=1,NSTP1
       IF (NBF(I) .LT. JBFMAX) GO TO 30
       JBFMAX=NBF(I)
    30 CONTINUE
       DO 20 J=1,JBFMAX
       DO 20 I=1,NSTP1
       ELBF(I,J)=1.0
       GAMBF(I,J)=PION2
    20 CONTINUE
C
C       VALUES ALONG FIELD LINES WHICH CROSS SHOCK
C
       DO 130 J=3,NB
       IF (J .GT. NBF(1)) GO TO 100
       D1=SQRT((XBF(2,J)-XBF(1,J))**2+RBF(2,J)**2)*2.0
       ELBF(2,J)=D1*DST(1)
       GAM1=PION2
       DX2=XBF(3,J)-XBF(2,J)
       DR2=RBF(3,J)-RBF(2,J)
       D2=SQRT(DX2*DX2+DR2*DR2)
       GAM2=ATAN(DR2/DX2)
       ELBF(2,J)=D1*D2/(D1+D2)*(DST(2)+DST(1))
       GAMBF(2,J)=(GAM1*D2+GAM2*D1)/(D1+D2)
       GO TO 140
   100 CONTINUE
       D1=SQRT((XBF(2,J)-XBF(2,J-1))**2+(RBF(2,J)-RBF(2,J-1))**2)*2.0
       DX2=XBF(3,J)-XBF(2,J)
       DR2=RBF(3,J)-RBF(2,J)
       D2=SQRT(DX2*DX2+DR2*DR2)
       GAM2=ATAN(DR2/DX2)
       ELBF(2,J)=(ELBF(2,J-1)+D1*DST(2))*D2/(D1+D2)
       GAMBF(2,J)=(D1*GAM2+D2*GAMBF(2,J-1))/(D1+D2)
   140 CONTINUE
       DO 110 I=3,NST
       D1=D2
       GAM1=GAM2
       IF (XBF(I+1,J) .LT. XST(I,1)) GO TO 120
       DX2=XBF(I+1,J)-XBF(I,J)
       DR2=RBF(I+1,J)-RBF(I,J)
       D2=SQRT(DX2*DX2+DR2*DR2)
       ELBF(I,J)=D1*D2/(D1+D2)*(DST(I-1)+DST(I))
       GAM2=ATAN(DR2/DX2)
       GAMBF(I,J)=(GAM1*D2+GAM2*D1)/(D1+D2)
   110 CONTINUE
   120 XSH=XBF(I+1,J)
       CALL RSHK1(XSH,RSH,GAMSH,ELSH)
       DX2=XSH-XBF(I,J)
       DR2=RSH-RBF(I,J)
       D2=SQRT(DX2*DX2+DR2*DR2)*2.0
       ELBF(I,J)=D1*(DST(I-1)*D2+ELSH)/(D1+D2)
       GAMBF(I,J)=(D2*GAM1+D1*GAMSH)/(D1+D2)
   130 CONTINUE
       JBFMAX=NBF(3)
C
C      VALUES ALONG FIELD LINES WHICH END AT DOWNSTREAM BOUNDARY
C
       NB1=NB+1
       DO 190 J=NB1,JBFMAX
       IF (J .GT. NBF(1)) GO TO 150
       D1=SQRT((XBF(2,J)-XBF(1,J))**2+RBF(2,J)**2)*2.0
       ELBF(1,J)=D1*DST(1)
       GAM1=PION2
       DX2=XBF(3,J)-XBF(2,J)
       DR2=RBF(3,J)-RBF(2,J)
       D2=SQRT(DX2*DX2+DR2*DR2)
       GAM2=ATAN(DR2/DX2)
       ELBF(2,J)=D1*D2/(D1+D2)*(DST(2)+DST(1))
       GAMBF(2,J)=(GAM1*D2+GAM2*D1)/(D1+D2)
       GO TO 160
   150 D1=SQRT((XBF(2,J)-XBF(2,J-1))**2+(RBF(2,J)-RBF(2,J-1))**2)*2.0
```

```
       DX2=XBF(3,J)-XBF(2,J)
       DR2=RBF(3,J)-RBF(2,J)
       D2=SQRT(DX2*DX2+DR2*DR2)
       GAM2=ATAN(DR2/DX2)
       ELBF(2,J)=(ELBF(2,J-1)+D1*DST(2))*D2/(D1+D2)
       GAMBF(2,J)=(D1*GAM2+D2*GAMBF(2,J-1))/(D1+D2)
   160 CONTINUE
       DO 170 I=3,NST
       IF (NBF(I+1) .LT. J) GO TO 180
       D1=D2
       GAM1=GAM2
       DX2=XBF(I+1,J)-XBF(I,J)
       DR2=RBF(I+1,J)-RBF(I,J)
       D2=SQRT(DX2*DX2+DR2*DR2)
       ELBF(I,J)=D1*D2/(D1+D2)*(DST(I-1)+DST(I))
       GAM2=ATAN(DR2/DX2)
       GAMBF(I,J)=(GAM1*D2+GAM2*D1)/(D1+D2)
   170 CONTINUE
   180 CONTINUE
       EL1=D1*DST(I-2)
       EL2=D2*DST(I-1)
       D1=D1+2.0*D2
       D2=-D2
       ELBF(I,J)=(EL1*D2+EL2*D1)/(D1+D2)
       GAMBF(I,J)=(GAM1*D2+GAM2*D1)/(D1+D2)
       IF (GAMBF(I,J) .LT. 0.0) GAMBF(I,J)=0.0
   190 CONTINUE
C
C      EXTRAPOLATE ALONG STREAMLINES TO LAST GRID LINE
C
       CALL RSHK1(ZPLOT,Y1,GAM1,EL1)
       DO 200 I=3,NSTP1
       N=NBF(I)
       NUMM=NUMST(I-1)
       IF (XBF(I,N-1) .LT. XST(I-1,1)) GO TO 220
       FAC=(ZPLOT-XBF(I,N-1))/(XBF(I,N)-XBF(I,N-1))
       XBF(I,N+1)=XST(I-1,NUMM)
       RBF(I,N+1)=YST(I-1,NUMM)
       ELBF(I,N+1)=FAC*ELBF(I,N)+(1.0-FAC)*ELBF(I,N-1)
       GAMBF(I,N+1)=FAC*GAMBF(I,N)+(1.0-FAC)*GAMBF(I,N-1)
       IF (GAMBF(I,N+1) .LT. 0.0) GAMBF(I,N+1)=0.0
       N1=N
       I1=I
       GO TO 200
   220 RBF(I,N+1)=YST(I-1,NUMM)
       XBF(I,N+1)=XST(I-1,NUMM)
       FAC=(Y1-RBF(I,N+1))/(Y1-RBF(I1,N1+1))
       ELBF(I,N+1)=ELBF(I1,N1+1)*FAC+(1.0-FAC)*EL1
       GAMBF(I,N+1)=GAMBF(I1,N1+1)*FAC+(1.0-FAC)*GAM1
       IF (GAMBF(I,N+1) .LT. 0.0) GAMBF(I,N+1)=0.0
   200 CONTINUE
       RETURN
       END



       SUBROUTINE BOUND
C
C      THIS ROUTINE DRAWS AND LABELS SHOCK WAVE AND
C           MAGNETOSPHERE OR IONOSPHERE BOUNDARY.
C           UCC PLOT SUBROUTINES USED ARE
C                VECTOR,CHAR,POLAR.
C
       COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
      *  NBMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
       COMMON /SCALE/ XSF,YSF,XMAX,YMAX,XLNGTH,YLNGTH
       DIMENSION R(21),A(21)
       DIMENSION LABMG(2)
       DATA LABMG(1),LABMG(2),LABIO/10HMAGNETOPAU,2HSE,9HIONOPAUSE/
       DATA LABSH/10HSHOCK WAVE/
       DATA A/0.,.15708,.31416,.47124,.62832,.78540,
      1       .94248,1.09956,1.25664,1.41372,1.57096,
      2       1.72788,1.88496,2.04204,2.19911,2.35619,
      3       2.51327,2.67035,2.82743,2.98451,3.14159/
       DATA R/21*0.1/
       DATA NP/21/
C
C      DRAW AND LABEL BODY BOUNDARY.
C
       CALL VECTOR(XBOD,YBOD,NXMAX,1,0,1H )
       XLABL=-1.0+1.0/XSF
       YLABL=0.6
       IF (NHINDX .EQ.1) GO TO 5
```

```
      CALL CHAR(XLABL,YLABL,0.0,.12,LABMG,12)
      GO TO 10
C
C     IONOPAUSE - ADD HRO TO LABEL
C
    5 CALL CHAR(XLABL,YLABL,0.0,.12,LABIO,9)
      XCH=-1.5+3.4/XSF
      YCH=YMAX-0.3/YSF
      CALL CHAR(XCH,YCH,0.0,.2,3HH/R,3)
      CALL CHAR(XCH+.6/XSF,YCH,0.0,.075,1H0,1)
      CALL CHAR(XCH+.75/XSF,YCH,0.0,.2,1H=,1)
      CALL NUMPLT(XCH+1./XSF,YCH,0.0,.2,HRO,2)
C
C     DRAW AND LABEL SHOCK WAVE.
C
   10 CONTINUE
      CALL VECTOR(XSHK,YSHK,NXMAX,1,0,1H )
      YLABL=-1.0-1.0/YSF
      XLABL=1.4
      CALL CHAR(XLABL,YLABL,0.0,.12,LABSH,10)
C
C     DRAW AND LABEL PLANET FOR MAGNETOSPHERE
C
      IF (NHINDX .EQ. 1) RETURN
      RMAX=.1*XSF
      CALL POLAR(R,A,NP,1,0,1H ,-RMAX,1.)
      CALL CHAR(.1,.08,0.0,.12,6HPLANET,6)
C
      RETURN
      END
```

```
      SUBROUTINE BOUT
C
C     THIS SUBROUTINE PRINTS OUT THE MAGNETIC FIELD
C
      COMMON /BIN/ ANGP,ANGN,KBCON,BCON(20)
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NXPLOT
      LEVEL 2, ANG,DXTH,DEG
      COMMON /DRD/ ANG(20,100),DXTH(20,20),DEG
      LEVEL 2, BPARA,BPERP,BNORM,BMAG,BANG
      COMMON /BCOMPS/ BPARA(20,100),BPERP(20,100),BNORM(20,100),
     *  BMAG(20,100),BANG(20,100)
      COMMON /BLUNT/ THETA(25),RR(20,25),NBLUNT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
C
      SANGP=SIN(ANGP)
      CANGP=COS(ANGP)
      SANGN=SIN(ANGN)
      CANGN=COS(ANGN)
      WRITE(6,100)
C
C     PRINT MAGNETIC FIELD FOR NOSE REGION
C
      DO 10 J=1,NBLUNT
      IF (NHINDX .EQ. 1) GO TO 3
      WRITE(6,110) J,THETA(J)
      GO TO 5
    3 WRITE(6,115) J,THETA(J)
    5 I=1
      WRITE(6,120) I,RR(I,J),BPARA(I,J)
      DO 10 I=2,NRMAX
      B1=SANGP*BPERP(I,J)
      B2=CANGP*BPARA(I,J)
      F1=COS(ANG(I,J))*B2+COS(BANG(I,J))*B1
      F2=SIN(ANG(I,J))*B2+SIN(BANG(I,J))*B1
      BMAG(I,J)=SQRT(F1*F1+F2*F2)
      BANGP=ATAN2(F2,F1)*DEG
      BX=F1*CANGN
      BY=F2*CANGN
      BZ=BNORM(I,J)*SANGN
      WRITE (6,120) I,RR(I,J),BPARA(I,J),BPERP(I,J),BMAG(I,J),BANGP,
     *  BNORM(I,J),BX,BY,BZ
   10 CONTINUE
C
C     PRINT MAGNETIC FIELD DOWNSTREAM
C
      NX1=NBLUNT+1
      DO 20 J=NX1,NXMAX
      JZ=J-NBLUNT
      Z=-XC(1,J)
      IF (NHINDX .EQ. 1) GO TO 13
      WRITE(6,130) JZ,Z
      GO TO 15
   13 WRITE(6,135) JZ,Z
   15 I=1
      WRITE(6,120) I,YC(I,J),BPARA(I,J)
      DO 20 I=2,NRMAX
      B1=SANGP*BPERP(I,J)
      B2=CANGP*BPARA(I,J)
      F1=COS(ANG(I,J))*B2+COS(BANG(I,J))*B1
      F2=SIN(ANG(I,J))*B2+SIN(BANG(I,J))*B1
      BMAG(I,J)=SQRT(F1*F1+F2*F2)
      BANGP=ATAN2(F2,F1)*DEG
      BX=F1*CANGN
      BY=F2*CANGN
      BZ=BNORM(I,J)*SANGN
      WRITE (6,120) I,YC(I,J),BPARA(I,J),BPERP(I,J),BMAG(I,J),BANGP,
     *  BNORM(I,J),BX,BY,BZ
   20 CONTINUE
      RETURN
C
  100 FORMAT(1H1//52X,25HMAGNETIC FIELD COMPONENTS/52X,25(1H*)/)
  110 FORMAT(//21H ANGULAR LOCATION NO.,I2,12H, AT THETA =,F8.4,
     *  9H DEGREES//
     *  4X,1HI,5X,4HRR/D,8X,2(6HB/BINF,6X),6HB/BINF,5X,7HB-ANGLE,5X,
     *  7H B/BINF,5X,7HBX/BINF,5X,7HBY/BINF,5X,7HBZ/BINF/
     *  21X,10H(PARALLEL),4X,6H(PERP),4X,2(10H(IN-PLANE),2X),1X,
     *  8H(NORMAL),1X,3(1X,11H(RESULTANT)))
  115 FORMAT(//21H ANGULAR LOCATION NO.,I2,12H, AT THETA =,F8.4,
     *  9H DEGREES//
     *  4X,1HI,6X,5HRR/RC,7X,2(6HB/BINF,6X),6HB/BINF,5X,7HB-ANGLE,5X,
     *  7H B/BINF,5X,7HBX/BINF,5X,7HBY/BINF,5X,7HBZ/BINF/
     *  21X,10H(PARALLEL),4X,6H(PERP),4X,2(10H(IN-PLANE),2X),1X,
     *  8H(NORMAL),1X,3(1X,11H(RESULTANT)))
  120 FORMAT(I5,9(2X,F11.4))
  130 FORMAT(//30H ADDITIONAL AXIAL LOCATION NO.,I2,10H, AT X/D =,F8.4//
     *  4X,1HI,7X,3HR/D,9X,2(6HB/BINF,6X),6HB/BINF,5X,7HB-ANGLE,5X,
     *  7H B/BINF,5X,7HBX/BINF,5X,7HBY/BINF,5X,7HBZ/BINF/
     *  21X,10H(PARALLEL),4X,6H(PERP),4X,2(10H(IN-PLANE),2X),1X,
     *  8H(NORMAL),1X,3(1X,11H(RESULTANT)))
  135 FORMAT(//30H ADDITIONAL AXIAL LOCATION NO.,I2,11H, AT X/RC =,F8.4/
     *  4X,1HI,7X,4HR/RC,7X,2(6HB/BINF,6X),6HB/BINF,5X,7HB-ANGLE,5X,
     *  7H B/BINF,5X,7HBX/BINF,5X,7HBY/BINF,5X,7HBZ/BINF/
     *  21X,10H(PARALLEL),4X,6H(PERP),4X,2(10H(IN-PLANE),2X),1X,
     *  8H(NORMAL),1X,3(1X,11H(RESULTANT)))
      END
```

```
      SUBROUTINE BSHK1 (X,R,ANG,EL)
C
C     THIS SUBROUTINE CALCULATES THE MAGNITUDE AND DIRECTION OF
C     EL/ELINF AT THE SHOCK, AND THE R-LOCATION, GIVEN THE
C     X-LOCATION OF THE POINT
C
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      COMMON /SHOCKS/ DRSDX(100),DST(50)
      DO 10 J=1,NXMAX
      IF (X .LT. XSHK(J)) GO TO 20
   10 CONTINUE
      J=NXMAX
   20 R=YSHK(J-1)+(X-XSHK(J-1))*(YSHK(J)-YSHK(J-1))/(XSHK(J)-XSHK(J-1))
      THET1=ATAN(DRSDX(J-1))
      THET2=ATAN(DRSDX(J))
      THET=THET1+(X-XSHK(J-1))*(THET2-THET1)/(XSHK(J)-XSHK(J-1))
C
C     CALCULATE MAGNITUDE OF VERTICAL FIELD COMPONENT OF SHOCK
C
      S2=SIN(THET)**2
      EM2=AMACH**2
      GAM2=(GAMMA+1.0)*0.5*EM2
      DD2=1.0-(EM2*S2-1.0)*(GAMMA*EM2*S2+1.0)/(GAM2**2*S2)
      DD=SQRT(DD2)
      COTH=1.0/TAN(THET)
      TD=GAM2/(EM2*S2-1.0)-1.0
      DELT=ATAN(COTH/TD)
      EL=SQRT(1.0+COTH*COTH*(1.0+DD2)-2.0*DD*(TD+1.0)*COTH*SIN(DELT))
C
C     CALCULATE ANGLE OF VERTICAL FIELD COMPONENT
C
      ALPHA=ASIN(DD*COTH*SIN(THET-DELT)/EL)
      ANG=THET+ALPHA
```

```
      RETURN
      END


      SUBROUTINE BSTEP
C
C     THIS SUBROUTINE CALCULATES THE VERTICAL FIELD LINES,
C        BY INTEGRATING ALONG EACH STREAMLINE TO LOCATE POSITIONS
C        AT EQUAL TIME INTERVALS
C
      COMMON /BLUNT/ THETA(25),RP(20,25),NBLUNT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,HHINDX
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      COMMON /DNSTRM/ ZPLOT,NZERO,NZADD,NXPLOT
      LEVEL 2,NB,NBF,XBF,RBF,ELBF,GAMBF
      COMMON /BVAL/ NB,NBF(51),XBF(51,100),RBF(51,100),ELBF(51,100),
     *  GAMBF(51,100)
      LEVEL 2, XST,YST,NUMST,NST
      COMMON /STREAM/ XST(50,152),YST(50,152),NUMST(50),NST
C
C     SECOND FIELD LINE IS TANGENT TO SHOCK NOSE
C
      JBFMAX=99
      NB=MIN1(ZPLOT*10.0+14.0,30.0)
      XSHKZ=XC(NRMAX,1)
      DSINF=(XST(NST,1)-XSHKZ)/(FLOAT(NB)-1.5)
      DELT=DSINF
C
C     CALCULATE WHERE VERTICAL FIELD LINES CROSS SYMMETRY AXIS
C
      DO 600 J=1,JBFMAX
      RBF(1,J)=0.0
  600 CONTINUE
      XBF(1,1)=XSHKZ-DSINF
      XBF(1,2)=XSHKZ
C
C     ASSUME CONSTANT DECELERATION BETWEEN GRID POINTS
C
      J=2
      TOLD=0.0
      IR=NRMAX
      V2=VF(IR,1)
  610 V1=V2
      IR=IR-1
      IF (IR .LE. 0) GO TO 640
      V2=VF(IR,1)
      DT=2.0*(XC(IR,1)-XC(IR+1,1))/(V2+V1)
      A=(V2-V1)/DT
      TNEW=TOLD+DT
      IF (TNEW .GE. DELT) GO TO 620
      TOLD=TNEW
      GO TO 610
  620 DT=DELT-TOLD
      DELS=V1*DT+0.5*A*DT*DT
      J=J+1
      XBF(1,J)=XC(IR+1,1)+DELS
      TOLD=TNEW-DELT
  630 IF (J .GE. JBFMAX) GO TO 640
      IF (TOLD .LT. DELT) GO TO 610
      V1=V1+A*DT
      DELS=V1*DELT+0.5*A*DELT**2
      TOLD=TOLD-DELT
      DT=DELT
      J=J+1
      XBF(1,J)=XBF(1,J-1)+DELS
      GO TO 630
  640 CONTINUE
      NBF(1)=J
      I=1
C
C     CALCULATE WHERE VERTICAL FIELD LINES CROSS STREAMLINES
C
      DO 700 IM=1,NST
      I=IM+1
      XBF(I,1)=XSHKZ-DSINF
      RBF(I,1)=YST(IM,1)
      XBF(I,2)=XSHKZ
      RBF(I,2)=YST(IM,1)
C
C     LOCATE POINTS BEFORE SHOCK WAVE
C
      DO 710 J=3,NB
      IF (XBF(I,J-1)+DSINF .GT. XST(IM,1)) GO TO 720
      XBF(I,J)=XBF(I,J-1)+DSINF
      RBF(I,J)=YST(IM,1)
  710 CONTINUE
C
C     LOCATE POINTS WITHIN THE MAGNETO/IONSPHERE
C
  720 TOLD=(XST(IM,1)-XBF(I,J-1))
      J=J-1
      KST=1
      IF (IM .GE. NBLUNT) GO TO 810
      V2=VF(NRMAX,I)
      GO TO 730
  810 DO 820 JJ=NBLUNT,NXMAX
      IF (XST(IM,1) .GT. XC(NRMAX,JJ)) GO TO 820
      V2=VF(NRMAX,JJ-1)+(VF(NRMAX,JJ)-VF(NRMAX,JJ-1))
     *     *(XST(IM,1)-XC(NRMAX,JJ-1))/(XC(NRMAX,JJ)-XC(NRMAX,JJ-1))
      GO TO 730
  820 CONTINUE
  730 V1=V2
      X1=XST(IM,KST+1)
      Y1=YST(IM,KST+1)
      V2=VINTRP(X1,Y1)
      DST=SQRT((XST(IM,KST+1)-XST(IM,KST))**2
     *     +(YST(IM,KST+1)-YST(IM,KST))**2)
      DT=2.0*DST/(V1+V2)
      A=(V2-V1)/DT
      TNEW=TOLD+DT
      IF(TNEW.GE.DELT) GO TO 740
      KST=KST+1
      TOLD=TNEW
      IF (KST .GE. NUMST(IM)) GO TO 760
      GO TO 730
  740 J=J+1
      DT=DELT-TOLD
      DELS=V1*DT+0.5*A*DT*DT
      XBF(I,J)=XST(IM,KST)+DELS*(XST(IM,KST+1)-XST(IM,KST))/DST
      RBF(I,J)=YST(IM,KST)+DELS*(YST(IM,KST+1)-YST(IM,KST))/DST
      TOLD=TNEW-DELT
      KST=KST+1
      IF (KST .GE. NUMST(IM)) GO TO 760
  750 IF (J .GE. JBFMAX) GO TO 760
      IF (TOLD .LT. DELT) GO TO 730
      J=J+1
      V1=V1+A*DT
      DELS=V1*DELT+0.5*A*DELT*DELT
      XBF(I,J)=XBF(I,J-1)+DELS*(XST(IM,KST)-XST(IM,KST-1))/DST
      RBF(I,J)=RBF(I,J-1)+DELS*(YST(IM,KST)-YST(IM,KST-1))/DST
      DT=DELT
      TOLD=TOLD-DELT
      GO TO 750
  760 CONTINUE
      NBF(I)=J
      IF (J .GT. NBF(I-1)) NBF(I)=NBF(I-1)
  700 CONTINUE
      RETURN
      END


      SUBROUTINE BUBBL(A,S,B,K,TLB,NLAB)
C
C     USING A BUBBLE TECHNIQUE, THIS ROUTINE SORTS THE REAL
C          ARRAY S INTO ASCENDING ORDER AND CHANGES
C          THE ORDER OF ARRAYS A AND B IN A CORRESPONDING
C          MANNER.  K IS THE NUMBER OF DATA POINTS TO
C          BE SORTED.
C
      DIMENSION A(1),S(1),B(1),TLB(1)
C
      IF(K.EQ.1) RETURN
      K1=K-1
      DO 100 I=1,K1
      L=I+1
      DO 100 J=L,K
      IF(S(J).GT.S(I)) GO TO 100
C
C     INTERCHANGE ARRAYS
C
      TEMP=S(I)
      S(I)=S(J)
```

```
      B(J)=TEMP
      TEMP=A(I)
      A(I)=A(J)
      A(J)=TEMP
      TEMP=B(I)
      B(I)=B(J)
      B(J)=TEMP
  100 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE CHECK(ICHK,NXY,KOD2,J,K,NVAL,KOD9)
C
C        CONTOUR PROGRAMS MAP, WALK, SERCH, ENTER, AND CHECK
C        WRITTEN BY REESE SORENSON, NASA-AMES RES. CTR., AUG., 1974.
C          (MODIFIED VERSION)
C
C        GIVEN THAT A LINE PASSES THROUGH AN INTERVAL UNDER
C        INVESTIGATION.  QUESTION: IS IT A NEW LINE, OR IS IT
C        PART OF A LINE ALREADY RECORDED?
C
C        KOD9     =1        OK, NEW POINT.
C        KOD9     =2        BAD, OLD POINT.
C
      DIMENSION ICHK(4,1)
C
      DO 1 L=1,NXY
      IF(KOD2.NE.ICHK(1,L))  GO TO 1
      IF(J.NE.ICHK(2,L))  GO TO 1
      IF(K.NE.ICHK(3,L))  GO TO 1
      IF(NVAL.EQ.ICHK(4,L))  GO TO 2
1     CONTINUE
      KOD9=1
      GO TO 3
2     KOD9=2
3     RETURN
      END
```

```
      SUBROUTINE CONOUT (ACONT,FACT,NHINDX)
C
C           THIS ROUTINE WRITES OUT THE CONTOUR LINES FOUND BY
C           SUBROUTINE MAP
C
      COMMON /ICHECK/ ICHK(4,1300)
      COMMON /PLOTC/ CONTX(1000),CONTY(1000),CVAL(30),NAD(30),IPLOT
      DIMENSION ACONT(1)
C
C     WRITE HEADING FOR THIS SET OF CONTOURS
C
      NMAX=NAD(1)
      IF(IPLOT.EQ.2) GO TO 10
      IF (IPLOT .GT. 4) GO TO 15
      WRITE(6,400) NMAX
      GO TO 20
   10 WRITE(6,500) NMAX
      GO TO 20
   15 IF (IPLOT .EQ. 5) WRITE(6,600) NMAX
      IF (IPLOT .EQ. 6) WRITE(6,610) NMAX
      IF (IPLOT.EQ.7) WRITE (6,640) NMAX
   20 CONTINUE
C
C     REVERSE SIGN OF X FOR OUTPUT
C
      JMAX=NAD(NMAX+1)-1
      DO 2 J=1,JMAX
    2 CONTX(J)=-CONTX(J)
C
C     PRINT CONTOUR LINE FOR EACH VALUE
C
      NAD(1)=1
      DO 1 N=1,NMAX
      NP=NAD(N+1)-NAD(N)
      NCONT=NAD(N)
      L=ICHK(4,NCONT)
      CVAL(N)=ACONT(L)
      M1=NAD(N)
      M2=NAD(N+1)-1
      IF(IPLOT.EQ.2) GO TO 30
      IF (IPLOT .GT. 4) GO TO 35
      TVAL=1.0+FACT*(1.0-CVAL(N)**2)
      WRITE(6,410) NP,CVAL(N),TVAL
      GO TO 40
   30 WRITE(6,510) NP,CVAL(N)
      GO TO 40
   35 IF (IPLOT .EQ. 5) WRITE(6,620) NP,CVAL(N)
      IF (IPLOT .EQ. 6) WRITE(6,630) NP,CVAL(N)
      IF (IPLOT.EQ.7) WRITE (6,650) NP,CVAL(N)
   40 IF(NHINDX.EQ.1) GO TO 45
      WRITE(6,420)
      GO TO 50
   45 WRITE(6,425)
   50 CONTINUE
      WRITE(6,430) (CONTX(I),CONTY(I),I=M1,M2)
    1 CONTINUE
      NAD(1)=NMAX
C
C     RESTORE SIGN OF X
C
      DO 3 J=1,JMAX
    3 CONTX(J)=-CONTX(J)
      RETURN
C
  400 FORMAT(1H1//49X,33HVELOCITY AND TEMPERATURE CONTOURS/49X,33(1H*)//
     *        10X,I3,43H VELOCITY (TEMPERATURE) CONTOUR LINES FOUND)
  410 FORMAT(///5X,I3,35H POINTS IN CONTOUR LINE OF V/VINF =,F7.3,
     *  11H,  T/TINF =,F7.3//)
  420 FORMAT(14X,3HX/D,17X,3HR/D/)
  425 FORMAT(14X,4HX/RO,16X,4HR/RO/)
  430 FORMAT(9X,F10.4,10X,F10.4)
  500 FORMAT(1H1//57X,16HDENSITY CONTOURS/57X,16(1H*)//
     *        10X,I3,28H DENSITY CONTOUR LINES FOUND)
  510 FORMAT(///5X,I3,39H POINTS IN CONTOUR LINE OF RHO/RHOINF =,F7.3//)
  600 FORMAT(1H1//53X,23HMAGNETIC FIELD CONTOURS/53X,23(1H*)//
     *        10X,I3,35H MAGNETIC FIELD CONTOUR LINES FOUND/
     *         14X,33H(FOR COMPONENT ALONG FIELD LINES ,
     *          31HPARALLEL TO FLOW IN FREESTREAM))
  610 FORMAT(1H1//10X,I3,35H MAGNETIC FIELD CONTOUR LINES FOUND/
     *         14X,33H(FOR COMPONENT ALONG FIELD LINES ,
     *         36HPERPENDICULAR TO FLOW IN FREESTREAM))
  620 FORMAT(///5X,I3,34H POINTS IN CONTOUR LINE OF B/BINF ,
     1       12H(PARALLEL) =,F7.3//)
  630 FORMAT(///5X,I3,34H POINTS IN CONTOUR LINE OF B/BINF ,
     *       17H(PERPENDICULAR) =,F7.3//)
  640 FORMAT(1H1//10X,I3,35H MAGNETIC FIELD CONTOUR LINES FOUND/
     *       14X,33H(FOR COMPONENT ALONG FIELD LINES ,
     *       29HNORMAL TO FLOW IN FREESTREAM))
  650 FORMAT(///5X,I3,34H POINTS IN CONTOUR LINE OF B/BINF ,
     *       10H(NORMAL) =,F7.3//)
      END
```

```
      SUBROUTINE CONTR
C
C     THIS SUBROUTINE PLOTS AND LABELS CONTOUR LINES.
C     ALSO DRAWS FIELD LINES FOR MAGNETIC FIELD STRENGTH PLOTS.
C          UCC PLOT SUBROUTINES USED ARE
C               DOTLN,NUMPLT,VECTOR.
C
      DIMENSION XS(153),YS(153)
      COMMON /LABLS/ XLAB(30),YLAB(30),CV(30),NCL,ILB(30),NLAB
      COMMON /SCALE/ XSF,YSF,XMAX,YMAX,XLNGTH,YLNGTH
      LEVEL 2,NR,NRF,XRF,RRF,ELRF,GAMRF
      COMMON /RVAL/ NR,NRF(51),XRF(51,100),RRF(51,100),ELRF(51,100),
     *  GAMRF(51,100)
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NXPLOT
      COMMON /BOUNDS/ XBOD(130),YBOD(130),XSHK(150),YSHK(150),
     *  NBMAX,NXMAX,AMACH,GAMMA,HPO,NHINDX
      LEVEL 2, XST,YST,NUMST,NST
      COMMON /STREAM/ XST(50,152),YST(50,152),NUMST(50),NST
      COMMON /PLOTC/ CONTX(1000),CONTY(1000),CVAL(30),NAD(30),IPLOT
C
      IF(IPLOT.EQ.3) GO TO 70
C
C     DRAW CONTOUR LINES.
C
      IR=1
```

```
      NCL=NAD(1)
      DO 50 N=1,NCL
      IE=NAD(N+1)
      NPTS=IE-IB
      CALL VECTOR(CONTX(IB),CONTY(IB),NPTS,1,3,1H )
C
C     LABEL CONTOURS.
C
C     DETERMINE WHICH END OF CONTOUR LINE IS CLOSER
C          TO BOUNDARY.
C          LABEL CONTOUR AT THAT END.
C
C
C     FIND MINIMUM DISTANCE FROM CONTOUR END POINTS
C          TO BOUNDARY.
C
      K=1
      XC=CONTX(IB)
      YC=CONTY(IB)
    5 DMIN=999.
      DO 10 I=1,NXMAX
      D=SQRT((XBOD(I)-XC)**2+(YBOD(I)-YC)**2)
      IF(D.GE.DMIN) GO TO 10
      DMIN=D
   10 CONTINUE
      IF(K.EQ.2) GO TO 15
      DMIN1=DMIN
      XC=CONTX(IE-1)
      YC=CONTY(IE-1)
      K=2
      GO TO 5
   15 CONTINUE
      DMIN2=DMIN
C
C     DETERMINE WHICH END OF CONTOUR LINE IS
C          CLOSER TO BOUNDARY.
C
      IF(DMIN1.LT.DMIN2) GO TO 20
      XLAB(N)=CONTX(IE-1)+.1/XSF
      YLAB(N)=CONTY(IE-1)-.1/XSF
      GO TO 30
   20 XLAB(N)=CONTX(IB)+.1/XSF
      YLAB(N)=CONTY(IB)-.1/XSF
   30 CONTINUE
      CV(N)=CVAL(N)
      IB=NAD(N+1)
   50 CONTINUE
C
C     SUBROUTINE LABEL CHECKS THE ARRAYS OF LABEL
C          COORDINATES FOR OVERLAP.  IF A LABEL OVERLAPS,
C          IT IS NOT SAVED UNLESS IT DEFINES A MINIMUM
C          OR MAXIMUM CONTOUR VALUE.
C          ARRAY ILB CONTAINS INDICES OF LABELS TO BE
C          PLOTTED.
C
      CALL LABEL
C
C     LABEL CONTOUR LINES
C
      DO 60 I=1,NLAB
      N=ILB(I)
      CALL NUMPLT(XLAB(N),YLAB(N),0.0,.1,CV(N),2)
   60 CONTINUE
      IF (IPLOT .EQ. 5) GO TO 100
      IF (IPLOT .EQ. 6) GO TO 150
      RETURN
C
C     DRAW STREAMLINES
C
   70 CONTINUE
      DO 75 K=1,NST
      NN=NUMST(K)
      XS(1)=XST(K,1)-0.2
      YS(1)=YST(K,1)
      DO 80 N=1,NN
      XS(N+1)=XST(K,N)
      YS(N+1)=YST(K,N)
   80 CONTINUE
      CALL VECTOR(XS,YS,NN+1,1,0,1H )
   75 CONTINUE
      RETURN
C
C     DRAW STREAMLINES FOR MAGNETIC FIELD PLOT
C
  100 DO 110 K=2,NST,3
      NN=NUMST(K)
      XS(1)=XST(K,1)-0.2
      YS(1)=YST(K,1)
      DO 120 N=1,NN
      XS(N+1)=XST(K,N)
      YS(N+1)=YST(K,N)
  120 CONTINUE
      DO 130 N=1,NN
      CALL DOTLN(XS(N),YS(N),XS(N+1),YS(N+1),20,0)
  130 CONTINUE
  110 CONTINUE
      RETURN
C
C     DRAW PERPENDICULAR FIELD LINES FOR MAGNETIC FIELD PLOT
C
  150 CONTINUE
      XS(1)=XBF(1,1)
      XS(2)=XBF(1,1)
      YS(1)=0.0
      YS(2)=0.4
      CALL DOTLN(XS(1),YS(1),XS(2),YS(2),20,0)
      NS=NBF(1)-3
      DO 160 K=3,NB,2
      DO 170 J=1,NXMAX
      IF (XSHK(J) .LT. XBF(NST+1,K)) GO TO 170
      GO TO 180
  170 CONTINUE
      J=NXMAX
  180 CONTINUE
      RSH=YSHK(J-1)+(XBF(NST,K)-XSHK(J-1))*(YSHK(J)-YSHK(J-1))
     *           /(XSHK(J)-XSHK(J-1))
      DO 190 J=1,NST
      IF (RBF(J,K) .GT. RSH) GO TO 200
      XS(J)=XBF(J,K)
      YS(J)=RBF(J,K)
  190 CONTINUE
  200 XS(J)=XBF(NST+1,K)
      YS(J)=RSH
      XS(J+1)=XS(J)
      YS(J+1)=AMIN1(RSH+0.2,YMAX)
      NN=J
      N1=1
C
C   OMIT PART OF LINE IF IT MIGHT BE DRAWN THROUGH BODY
C
      IF (YS(NN-1) .LT. 0.0) GO TO 207
      N=2
      DO 206 J=1,NXMAX
  202 IF (N .GE. NN) GO TO 206
      IF (XS(N) .GT. XBOD(J)) GO TO 204
      N=N+1
      GO TO 202
  204 YSXBOD=YS(N-1)+(YS(N)-YS(N-1))*(XBOD(J)-XS(N-1))/(XS(N)-XS(N-1))
      IF (YSXBOD .LT. YBOD(J)) N1=N
  206 CONTINUE
  207 CONTINUE
      DO 210 N=N1,NN
      CALL DOTLN(XS(N),YS(N),XS(N+1),YS(N+1),20,0)
  210 CONTINUE
      IF (NS .LT. K) RETURN
  160 CONTINUE
C
C   DRAW FIELD LINES WHICH DO NOT CROSS SHOCK
C
      NS=NB+1+MOD(NB,2)
      NF=MIN0(NB+10,NBF(1))
      DO 220 K=NS,NF,2
      DO 230 J=1,NST
      XS(J)=XBF(J,K)
      YS(J)=RBF(J,K)
      IF (NBF(J+1) .LT. K) GO TO 240
  230 CONTINUE
  240 NN=J
      XS(NN+1)=ZPLOT
      YS(NN+1)=YS(NN)+(XS(NN+1)-XS(NN))*(YS(NN)-YS(NN-1))
     *           /(XS(NN)-XS(NN-1))
      N=2
      N1=1
      DO 246 J=1,NXMAX
  242 IF(N .GE. NN) GO TO 246
      IF (XS(N) .GT. XBOD(J)) GO TO 244
      N=N+1
      GO TO 242
```

```
  244 YSXBOD=YS(N-1)+(YS(N)-YS(N-1))*(XBOD(J)-XS(N-1))/(XS(N)-XS(N-1))
      IF (YSXBOD .LT. YBOD(J)) N1=N
  245 CONTINUE
      DO 250 N=N1,NN
      CALL DOTLN(XS(N),YS(N),XS(N+1),YS(N+1),20,3)
  250 CONTINUE
  220 CONTINUE
      RETURN
C
      END
```

```
      SUBROUTINE CONTUR
C
C     SUBROUTINE CONTUR CONTROLS CALCULATING AND PRINTING THE CONTOURS
C                   AND CREATING THE PLOTS
C
      COMMON /RIN/ ANGP,ANGN,KBCON,BCON(20)
      LEVEL 2, BPARA,BPERP,BNORM,BMAG,BANG
      COMMON /BCOMPS/ BPARA(20,100),BPERP(20,100),BNORM(20,100),
     *  BMAG(20,100),BANG(20,100)
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      COMMON /CONT/ KVCON,VCON(20),KRCON,RCON(20)
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NXPLOT
      COMMON /PLOTC/ CONTX(1000),CONTY(1000),CVAL(30),NAD(30),IPLOT
      LOGICAL LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      COMMON /PROPT/ LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      LEVEL 2, XST,YST,NUMST,NST
      COMMON /STREAM/ XST(50,152),YST(50,152),NUMST(50),NST
      COMMON /TCHECK/ TCHK(4,1000)
      DIMENSION BSCM(20,100)
C
C     PLOT STREAMLINES
C
      IF (LPLOT) CALL SECALC(XSHK,YSHK,NXMAX)
      IPLOT=3
      IF (LPLOT) CALL PLOTCN
C
C
C     CALCULATE VELOCITY CONTOUR LINES
C            (GRID IS NRMAX BY NXMAX)
C
      IF (KVCON .LE. 0) GO TO 10
      CALL MAP(VF,20,CONTX,CONTY,KVCON,2,VCON,NAD,1000,30,TCHK,
     *     1,1,NRMAX,NXMAX)
C
C     PLOT VELOCITY AND TEMPERATURE CONTOURS
C
      IPLOT=1
      FACT=0.5*(GAMMA-1.0)*AMACH*AMACH
      IF (LPRCON) CALL CONOUT(VCON,FACT,NHINDX)
      IF (LPLOT) CALL PLOTCN
C
C     CALCULATE DENSITY CONTOUR LINES
C
   10 IF (KRCON .LE. 0) GO TO 20
      CALL MAP(RHOF,20,CONTX,CONTY,KRCON,2,RCON,NAD,1000,30,TCHK,
     *     1,1,NRMAX,NXMAX)
C
C     PLOT DENSITY CONTOURS
C
      IPLOT=2
      IF (LPRCON) CALL CONOUT(RCON,FACT,NHINDX)
      IF (LPLOT) CALL PLOTCN
C
C     CALCULATE, PRINT, AND PLOT CONTOUR LINES FOR PARALLEL
C     AND PERPENDICULAR MAGNETIC FIELD COMPONENTS
C
   20 IF (KBCON .LE. 0) GO TO 200
      DO 100 J=1,NXMAX
      DO 100 I=1,NRMAX
      BSCM(I,J)=BPARA(I,J)
  100 CONTINUE
      CALL MAP(BSCM,20,CONTX,CONTY,KBCON,2,BCON,NAD,1000,30,ICHK,
     *     1,1,NRMAX,NXMAX)
      IPLOT=5
      IF (LPRCON) CALL CONOUT(BCON,FACT,NHINDX)
      IF (LPLOT) CALL PLOTCN
      DO 110 J=1,NXMAX
      DO 110 I=2,NRMAX
      BSCM(I,J)=BPERP(I,J)
  110 CONTINUE
      CALL MAP(BSCM,20,CONTX,CONTY,KBCON,2,BCON,NAD,1000,30,ICHK,
     *      2,1,NRMAX,NXMAX)
      IPLOT=6
      IF (LPRCON) CALL CONOUT(BCON,FACT,NHINDX)
      IF (LPLOT) CALL PLOTCN
      DO 120  J=1,NXMAX
      DO 120  I=2,NRMAX
      BSCM(I,J)=BNORM(I,J)
  120 CONTINUE
      CALL MAP(BSCM,20,CONTX,CONTY,KBCON,2,BCON,NAD,1000,30,ICHK,
     *       2,1,NRMAX,NXMAX)
      IPLOT=7
      IF (LPRCON) CALL CONOUT(BCON,FACT,NHINDX)
      IF (LPLOT) CALL PLOTCN
C
  200 CONTINUE
      IF (LPLOT) CALL ENPLT(3.0,0.0)
      RETURN
      END
```

```
      FUNCTION DRDX(X,Y)
C
C          THIS FUNCTION DETERMINES THE SLOPE OF THE STREAMLINE
C          AT THE POINT (X,Y)
C
      COMMON /BLUNT/ THETA(25),RP(20,25),NBLUNT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      LEVEL 2, ANG,DXTH,DEG
      COMMON /DRD/ ANG(20,100),DXTH(100),DEG
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
C
C          LOCATING POINT IN GRID
C
      IF (Y .GE. 0.0) GO TO 10
      THTA=ATAN2(Y,-X)*DEG
      R=SQRT(Y**2+X**2)
      DO 3 J=1,NBLUNT
      IF (THETA(J).GT.THTA) GO TO 5
    3 CONTINUE
      J=NBLUNT
    5 JR=J-1
      IF (JR.LT.1) JR=1
      SLOPE=(THTA-THETA(JR))/DXTH(JR)
      R2=RP(1,JR)+(RP(1,JR+1)-RP(1,JR))*SLOPE
      DO 7 I=2,NRMAX
      R1=R2
      R2=RP(I,JR)+(RP(I,JR+1)-RP(I,JR))*SLOPE
      IF (R2 .GT. R) GO TO 8
    7 CONTINUE
      I=NRMAX
    8 GO TO 20
C
   10 CONTINUE
      R=Y
      DO 13 J=NBLUNT,NXMAX
      IF (XC(1,J) .GT.X) GO TO 15
   13 CONTINUE
      J=NXMAX
   15 JR=J-1
      IF (JR.LT.NBLUNT) JR=NBLUNT
      SLOPE=(X-XC(1,JR))/DXTH(JR)
      R2=YC(1,JR)+(YC(1,JR+1)-YC(1,JR))*SLOPE
      DO 17 I=2,NRMAX
      R1=R2
      R2=YC(I,JR)+(YC(I,JR+1)-YC(I,JR))*SLOPE
      IF (R2 .GT. R) GO TO 18
   17 CONTINUE
      I=NRMAX
   18 CONTINUE
C
C          BIVARIATE LINEAR INTERPOLATION
C
   20 CONTINUE
      DR1=ANG(I-1,JR)+(ANG(I-1,JR+1)-ANG(I-1,JR))*SLOPE
      DR2=ANG(I,JR)+(ANG(I,JR+1)-ANG(I,JR))*SLOPE
      DRDX=DR1+(DR2-DR1)*(R-R1)/(R2-R1)
```

```
      RETURN
      END
```

```
      SUBROUTINE ECHINP
C
C                    PRINTS INPUT CARDS USED FOR RUN
C
      DIMENSION CRD(8)
      WRITE (6,110)
   10 CONTINUE
      READ (5,100) CRD
C                    YES, NO
      IF (EOF(5)) 30,20
   20 CONTINUE
      WRITE (6,101) CRD
      GO TO 10
   30 CONTINUE
      REWIND 5
      RETURN
  100 FORMAT(8A10)
  101 FORMAT(1X,8A10)
  110 FORMAT(1H1,49X,35HLISTING OF INPUT CARDS FOR THIS RUN/49X,35(1H*)
     1  ///)
      END
```

```
      SUBROUTINE ENTER(KOD2,J,K,NVAL,A1,A2,JMIN,KMIN,ICHK,KOD4,X,Y,NXY,
     *  ACONT,ISIZ1)
C
C         CONTOUR PROGRAMS MAP, WALK, SERCH, ENTER, AND CHECK
C         WRITTEN BY REESE SORENSON, NASA-AMES RES. CTR., AUG., 1974.
C           (MODIFIED VERSION)
C
C         ASSUMING THAT A POINT ON A CONTOUR LINE HAS BEEN FOUND,
C         THIS SUBROUTINE RECORDS THAT POINT IN THE BOOKKEEPING ARRAYS.
C
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      DIMENSION ICHK(4,1),X(1),Y(1),ACONT(1)
C
      NXY=NXY+1
      IF(NXY.GT.ISIZ1)  GO TO 1
C
      ICHK(1,NXY)=KOD2
      ICHK(2,NXY)=J
      ICHK(3,NXY)=K
      ICHK(4,NXY)=NVAL
C
C     IF ENDPOINTS ARE EQUAL, ENTER MIDPOINT
C
      IF (( A2-A1).EQ.0.0) GO TO 6
      DIF=(ACONT(NVAL)-A1)/(A2-A1)
      GO TO (2,3),KOD2
6     DIF=0.5
      GO TO (2,3),KOD2
C
C           INTERPOLATE FOR CONTOUR POSITION
C
    2 X2=XC(J,K+1)
      Y2=YC(J,K+1)
      GO TO 4
    3 X2=XC(J+1,K)
      Y2=YC(J+1,K)
    4 X1=XC(J,K)
      Y1=YC(J,K)
      X(NXY)=X1+DIF*(X2-X1)
      Y(NXY)=Y1+DIF*(Y2-Y1)
      KOD4=1
      GO TO 5
C
1     WRITE(6,101)
  101 FORMAT(49H1CONTOUR SEARCH ABORTED - TABLE OVERFLOW IN (X,Y))
      KOD4=2
C
5     CONTINUE
      RETURN
      END
```

```
      SUBROUTINE EXTRAP
C
C     THIS ROUTINE CALCULATES EXTRAPOLATED VALUES OF
C          RHO AND V/V AT POINTS ALONG THE BOUNDARY THETA=0 USING
C          A LAGRANGIAN INTERPOLATING POLYNOMIAL OVER
C          THREE UNEQUALLY SPACED POINTS ON EACH RADIAL CURVE.
C
      COMMON /BLUNT/ THETA(25),RP(20,25),NBLUNT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,NRO,NHINDX
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      COMMON /SHOCKS/ DRSDX(100),DST(50)
C
C     CALCULATE LAGRANGIAN COEFFICIENTS
C
      TH23=THETA(2)-THETA(3)
      TH24=THETA(2)-THETA(4)
      TH34=THETA(3)-THETA(4)
      E2=THETA(3)*THETA(4)/(TH23*TH24)
      E3=THETA(2)*THETA(4)/(-TH23*TH34)
      E4=THETA(2)*THETA(3)/(TH24*TH34)
C
C     CALCULATE XC, RHO,AND V AT THETA=0.
C
      AM2=AMACH*AMACH
      DO 10 I=2,NRMAX
      YC(I,1)=0.0
      XC(I,1)=-(E2*RP(I,2)+E3*RP(I,3)+E4*RP(I,4))
      RP(I,1)=-XC(I,1)
      IF (I.EQ.NRMAX) GO TO 10
      RHOF(I,1)=E2*RHOF(I,2)+E3*RHOF(I,3)+E4*RHOF(I,4)
      VF(I,1)=E2*VF(I,2)+E3*VF(I,3)+E4*VF(I,4)
      IF (VF(I,1) .LT. 0.) VF(I,1)=0.
   10 CONTINUE
C
C     CALCULATE EXACT VALUES AT SHOCK WAVE
C
      RHOF(NRMAX,1)=(GAMMA+1.)*AM2/((GAMMA-1.)*AM2+2.)
      VF(NRMAX,1)=1.0/RHOF(NRMAX,1)
C
C     EXACT VALUES AT BODY
C
      YC(1,1)=0.0
      XC(1,1)=-1.0
      RP(1,1)=1.0
      VF(1,1)=0.0
      RHOF(1,1)=RHOF(NRMAX,1)*((GAMMA+1.0)**2*AM2*0.5/(2.0*GAMMA*AM2-
     *      (GAMMA-1.0)))**(1.0/(GAMMA-1.0))
      DRSDX(1)=9999.0
C
C     DEFINE BOUNDARY ARRAYS FOR IONOPAUSE/MAGNETOPAUSE AND SHOCK
C
      DO 20 J=1,NXMAX
      XBOD(J)=XC(1,J)
      YBOD(J)=YC(1,J)
      XSHK(J)=XC(NRMAX,J)
      YSHK(J)=YC(NRMAX,J)
   20 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE FLOUT
C
C     THIS ROUTINE PRINTS THE FLOW FIELD VALUES WHICH WILL BE USED
C          TO CALCULATE THE STREAMLINES AND CONTOURS
C
      COMMON /BLUNT/ THETA(25),RP(20,25),NBLUNT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,NRO,NHINDX
      LEVEL 2, ANG,DXTH,DEG
      COMMON /DRD/ ANG(20,100),DXTH(100),DEG
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      LEVEL 2, VX,VY
      COMMON/VCOMP/ VX(20,100),VY(20,100)
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NXPLOT
C
C     REVERSE SIGN OF XC FOR OUTPUT
C
```

```
      JMAX=NZEND+NZADD
      DO 2 J=1,JMAX
      DO 2 I=1,NRMAX
    2 XC(I,J)=-XC(I,J)
C
C     PRINT VALUES ALONG SYMMETRY AXIS
C
      WRITE(6,210)
      FACT=0.5*(GAMMA-1.0)*AMACH*AMACH
      IF(NHINDX.EQ.1) GO TO 10
      WRITE(6,220)
      GO TO 20
   10 WRITE(6,230)
   20 CONTINUE
      DO 30 I=1,NRMAX
      T=1.0+FACT*(1.0-VF(I,1)**2)
      P=RHOF(I,1)*T
      WRITE(6,200) I,XC(I,1),VF(I,1),RHOF(I,1),T,P
   30 CONTINUE
C
C     PRINT VALUES OVER BLUNT BODY
C
      WRITE(6,240)
      DO 40 J=2,NBLUNT
      WRITE(6,250) J,THETA(J)
      IF(NHINDX.EQ.1) GO TO 50
      WRITE(6,260)
      GO TO 60
   50 WRITE(6,270)
   60 CONTINUE
      DO 40 I=1,NRMAX
      T=1.0+FACT*(1.0-VF(I,J)**2)
      P=RHOF(I,J)*T
      ALPHA=DEG*ANG(I,J)
      WRITE(6,290) I,RP(I,J),YC(I,J),XC(I,J),VY(I,J),VX(I,J),ALPHA,
     * VF(I,J),RHOF(I,J),T,P
   40 CONTINUE
C
C     PRINT VALUES FOR MARCHING CODE REGION
C
      WRITE(6,280)
      NX1=NBLUNT+1
      DO 70 J=NX1,JMAX
      JZ=J-NBLUNT
      IF(NHINDX.EQ.1) GO TO 80
      WRITE(6,290) JZ,XC(1,J)
      WRITE(6,300)
      GO TO 90
   80 WRITE(6,310) JZ,XC(1,J)
      WRITE(6,320)
   90 CONTINUE
      DO 70 I=1,NRMAX
      T=1.0+FACT*(1.0-VF(I,J)**2)
      P=RHOF(I,J)*T
      ALPHA=DEG*ANG(I,J)
      WRITE(6,290) I,YC(I,J),VY(I,J),VX(I,J),ALPHA,VF(I,J),RHOF(I,J),T,P
   70 CONTINUE
C
C     RESTORE SIGN OF XC
C
      DO 3 J=1,JMAX
      DO 3 I=1,NRMAX
    3 XC(I,J)=-XC(I,J)
      RETURN
C
  200 FORMAT(2X,I3,10(2X,F10.4))
  210 FORMAT(1H1//51X,26HDETAILED FLOW FIELD OUTPUT/51X,26(1H*)/////
     *       50H FLOW FIELD VALUES EXTRAPOLATED TO SYMMETRY AXIS, ,
     *       20HTHETA = 0.00 DEGREES/)
  220 FORMAT(4X,1HI,7X,3HX/D,9X,5HV/VINF,3X,10HRHO/RHOINF,5X,6HT/TINF,
     * 6X,6HP/PINF)
  230 FORMAT(4X,1HI,7X,4HR/R0,7X,6HV/VINF,3X,10HRHO/RHOINF,5X,6HT/TINF,
     * 6X,6HP/PINF)
  240 FORMAT(1H1,41X,45HFLOW FIELD VALUES FROM BLUNT BODY CALCULATION)
  250 FORMAT(//21H ANGULAR LOCATION NO.,I2,12H, AT THETA =,F8.4,
     * 8H DEGREES/)
  260 FORMAT(4X,1HI,7X,4HRP/D,8X,3HR/D,9X,3HX/D,7X,7HVR/VINF,5X,
     * 7HVX/VINF,3X,10HFLOW ANGLE,5X,6HV/VINF,3X,10HRHO/RHOINF,5X,
     * 6HT/TINF,6X,6HP/PINF)
  270 FORMAT(4X,1HI,6X,5HRP/R0,8X,4HR/R0,8X,4HX/R0,6X,7HVR/VINF,5X,
     * 7HVX/VINF,3X,10HFLOW ANGLE,5X,6HV/VINF,3X,10HRHO/RHOINF,5X,
     * 6HT/TINF,6X,6HP/PINF)
  280 FORMAT(1H1,41X,43HFLOW FIELD VALUES FROM MARCHING CALCULATION)
  290 FORMAT(//30H ADDITIONAL AXIAL LOCATION NO.,I2,10H, AT X/D =,F8.4/)
  300 FORMAT(4X,1HI,7X,3HR/D,7X,7HVR/VINF,5X,7HVX/VINF,3X,
     * 10HFLOW ANGLE,5X,6HV/VINF,3X,10HRHO/RHOINF,5X,6HT/TINF,6X,
     * 6HP/PINF)
  310 FORMAT(//30H ADDITIONAL AXIAL LOCATION NO.,I2,11H, AT X/R0 =,
     * F8.4/)
  320 FORMAT(4X,1HI,7X,4HR/R0,6X,7HVR/VINF,5X,7HVX/VINF,3X,
     * 10HFLOW ANGLE,5X,6HV/VINF,3X,10HRHO/RHOINF,5X,6HT/TINF,6X,
     * 6HP/PINF)
      END
```

```
      SUBROUTINE FLOWST
C
C          THIS ROUTINE CALCULATES THE MAGNITUDE AND DIRECTION OF
C          THE VELOCITY, THEN CALCULATES THE TRAJECTORY STREAMLINES
C
      COMMON /BLUNT/ THETA(25),RP(20,25),NBLUNT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     * NRMAX,NXMAX,AMACH,GAMMA,HRD,NHINDX
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NXPLOT
      LEVEL 2, ANG,DXTH,DEG
      COMMON /DRD/ ANG(20,100),DXTH(100),DEG
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      LEVEL 2, XST,YST,NUMST,NST
      COMMON /STREAM/ XST(50,152),YST(50,152),NUMST(50),NST
      LEVEL 2, VX,VY
      COMMON/VCOMP/ VX(20,100),VY(20,100)
      DIMENSION W(20),S(25,6),AR(5),AV(5)
      DATA W/20*1.0/
C
C          CALCULATE VELOCITY AND FLOW ANGLE FROM VELOCITY COMPONENTS
C
C                  POLAR COORDINATE REGION
C
      DEG=57.29577951
      THETA(1)=0.0
      NXMAX=NXMAX+NZADD
      DO 10 J=2,NBLUNT
      THETA(J)=THETA(J)*DEG
      DXTH(J-1)=THETA(J)-THETA(J-1)
      DO 10 I=1,NRMAX
      VF(I,J)=SQRT(VX(I,J)**2+VY(I,J)**2)
      IF (ABS(VX(I,J)) .LT. 1.0E-4) GO TO 5
      ANG(I,J)=VY(I,J)/VX(I,J)
      GO TO 10
    5 ANG(I,J)=1.0E-4
   10 CONTINUE
C
C                 CYLINDRICAL COORDINATE REGION
C
      NX1=NBLUNT+1
      DO 20 J=NX1,NXMAX
      DXTH(J-1)=XC(1,J)-XC(1,J-1)
      DO 20 I=1,NRMAX
      VF(I,J)=SQRT(VX(I,J)**2+VY(I,J)**2)
      IF (ABS(VX(I,J)) .LT. 1.0E-4) GO TO 15
      ANG(I,J)=VY(I,J)/VX(I,J)
      GO TO 20
   15 ANG(I,J)=1.0E4
   20 CONTINUE
C
C          SMOOTH RHOF AND VF ALONG CONSTANT-THETA LINES AT NOSE,
C          USING THIRD DEGREE LEAST SQUARES FIT
C
      DO 25 J=2,NBLUNT
      CALL FLSQFY(NRMAX,3,RP(1,J),RHOF(1,J),W,25,S,AR,IER)
      CALL FLSQFY(NRMAX,3,RP(1,J),VF(1,J),W,25,S,AV,IER)
      DO 25 I=1,NRMAX
      XRP=RP(I,J)
      VF(I,J)=((AV(4)*XRP+AV(3))*XRP+AV(2))*XRP+AV(1)
      RHOF(I,J)=((AR(4)*XRP+AR(3))*XRP+AR(2))*XRP+AR(1)
   25 CONTINUE
C
C          EXTRAPOLATE FOR VALUES ALONG AXIS OF SYMMETRY
C
      CALL EXTRAP
      DO 30 I=1,NRMAX
      ANG(I,1)=0.0
   30 CONTINUE
C
C
```

```
C         CALCULATE STARTING POINTS FOR STREAMLINE CALCULATION
C                   USE GRID POINTS ON SHOCK IN NOSE REGION
C                   EQUAL Y-SPACING ON SHOCK DOWNSTREAM
C
      DO 32 J=2,NBLUNT
      XST(J-1,1)=XC(NRMAX,J)
      YST(J-1,1)=YC(NRMAX,J)
   32 CONTINUE
      NST=NBLUNT-1
      ADDST=51-NST
      DO 34 K=NBLUNT,NXMAX
      IF (YC(1,K) .GT. ZPLOT) GO TO 35
   34 CONTINUE
      K=NXMAX
   35 YSTF=YC(NRMAX,K)
      NXPLOT=K
      YSTN=YST(NST,1)
      DRST=AMAX1(YSTN-YST(NST-1,1),(YSTF-YSTN)/ADDST)
      KX=NBLUNT
      DO 39 K=NBLUNT,50
      YSTN=YSTN+DRST
      IF (YSTN .GE.YSTF) GO TO 40
      NST=NST+1
      DO 37 JX=KX,NXMAX
      IF (YSTN .GT. YC(NRMAX,JX)) GO TO 37
      YST(NST,1)=YSTN
      XST(NST,1)=XC(NRMAX,JX-1)+(XC(NRMAX,JX)-XC(NRMAX,JX-1))
     $           *(YSTN-YC(NRMAX,JX-1))/(YC(NRMAX,JX)-YC(NRMAX,JX-1))
      GO TO 39
   37 CONTINUE
   39 KX=JX
   40 CONTINUE
      ZPLOT=XC(1,NXPLOT)
C
C         CALCULATE STREAMLINE TRAJECTORIES
C         USING THIRD ORDER MODIFIED EULER INTEGRATION PROCEDURE
C
      DWRX1=-XC(NRMAX,2)*.02
      DWRX2=AMAX1(DWRX1,ZPLOT*.01+1.0E-6)
      DO 50 JST=1,NST
      XS=XST(JST,1)
      YS=YST(JST,1)
      DWRX=DWRX1
      IF (XS .GT. 0.0) DWRX=DWRX2
      DO 45 N=2,152
      IF (XS+DWRX .GE.ZPLOT) GO TO 47
      DX=DWRX*.2
      DO 43 K=1,5
      DRDX1=DRDX(XS,YS)
      YS1=YS+DRDX1*DX
      XS1=XS
      XS=XS1+DX
      YS=YS+0.5*(DRDX1+DRDX(XS,YS1))*DX
      IF (YS1.LE.0.0 .AND. XS.GT.0.0) DWRX=DWRX2
   43 CONTINUE
      XST(JST,N)=XS
      YST(JST,N)=YS
   45 CONTINUE
      N=152
   47 DX=(ZPLOT-XS)*.2
      DO 48 K=1,5
      DRDX1=DRDX(XS,YS)
      YS1=YS+DRDX1*DX
      XS=XS+DX
      YS=YS+0.5*(DRDX1+DRDX(XS,YS1))*DX
   48 CONTINUE
      XST(JST,N)=XS
      YST(JST,N)=YS
      NUMST(JST)=N
   50 CONTINUE
C
C             CONVERT ANG ARRAY FROM FLOW FIELD SLOPE TO FLOW ANGLE
C
      DO 60 J=2,NXMAX
      DO 60 I=1,NRMAX
      ANG(I,J)=ATAN(ANG(I,J))
   6  CONTINUE
      NXMAX=NXPLOT
      RETURN
      END
```

```
      SUBROUTINE FLSOFY (M,N,X,Y,W,MN1,S1,A,IER)
C******
C        PURPOSE
C           FLSOFY CONSTRUCTS A LEAST SQUARES POLYNOMIAL APPROXIMATION
C           OF SPECIFIED DEGREE TO A GIVEN SET OF DATA POINTS WITH
C           GIVEN WEIGHTS USING ORTHOGONAL POLYNOMIALS.
C        USAGE
C           DIMENSION X(M),Y(M),W(M),S1(MN1,6),A(N+1)
C           CALL FLSOFY (M,N,X,Y,W,MN1,S1,A,IER)
C        INPUT PARAMETERS
C           M    - NUMBER OF DATA POINTS
C           N    - DEGREE OF POLYNOMIAL DESIRED, N.LT.M
C           X    - ARRAY OF INDEPENDENT VARIABLE
C           Y    - ARRAY OF DEPENDENT VARIABLE
C           W    - ARRAY OF POSITIVE WEIGHTS
C           MN1  - ROW DIMENSION OF SCRATCH ARRAY S1, MN1.GE.M
C           S1   - SCRATCH ARRAY
C                  IN THE COLUMN DEFINITIONS BELOW,
C                        I    REFERS TO POLYNOMIAL ORDER
C                        J    REFERS TO ROW INDEXING WITHIN A COLUMN
C                  COL 1    POLYNOMIAL P(I-1), VALUE AT EACH X(J) THRU 240
C                           COEFFICIENT OF EACH X**(J-1) TERM AFTER 240
C                  COL 2    POLYNOMIAL P(I), SAME FORM AS COL 1
C                  COL 3    ALPHA(I+1), WHERE I+1=J
C                  COL 4    BETA(I), WHERE I=J-1
C                  COL 5    S(I), WHERE I=J-1
C                  COL 6    SIGMA**2, WHERE I=J-1
C        OUTPUT PARAMETERS
C           A    - ARRAY OF COMPUTED COEFFICIENTS
C                     A(1) THRU A(N+1) CONTAIN COMPUTED POLYNOMIAL
C                     COEFFICIENTS, IN ORDER OF INCREASING DEGREE.
C           IER  - ERROR INDICATOR
C                     .EQ.0    SUCCESS
C                     .EQ.1    (N.GE.M).OR.(N.LE.0)
C                     .EQ.2    MN1.LT.M
C                     .EQ.3    M.LE.1
C                     .EQ.4    W(I).LE.0
C        REFERENCE
C           FORSYTHE, G. E., GENERATION AND USE OF ORTHOGONAL
C           POLYNOMIALS FOR DATA-FITTING WITH A DIGITAL COMPUTER,
C           J. SIAM, VOL. 5, NO. 2, (JUNE 1957) PP. 74-88.
C           NOTE - MOST NOTATION AND MOST LOCAL VARIABLE NAMES ARE
C           BASED ON THE REFERENCE. IE WPP IS W(I,I)=(P(I),P(I)),
C           BUT W, WT AND W(I) REFER TO THE WEIGHTS ARRAY.
C******
      DIMENSION X(1),Y(1),W(1),S1(MN1,1),A(1)
C
C  ***  INITIAL SET-UP  ***
C          TEST FOR USER ERRORS
  100 IER = 4
      IF ((N.GE.M).OR.(N.LE.0)) IER=1
      IF (MN1.LT.M) IER=2
      IF (M.LE.1) IER=3
      IF (IER.NE.4) GO TO 990
C          BETA(0)=0, P(0)=1, P(-1)=0, W(0,0)=SUM(W(I))
C          DELTA**2(-1)=(WF,F)
      N1 = N+1
      S1(1,4)=0.
      DSQ=0.
      WPP=0.
      DO 110 J=1,M
      S1(J,2)=1.
      S1(J,1)=0.
      WT = W(J)
      IF (WT.LE.0.) GO TO 990
      WPP = WPP+WT
  110 DSQ = DSQ+WT*Y(J)*Y(J)
C
C  ***  GENERATE ORTHOGONAL POLYNOMIALS - THRU 240  ***
      DO 240 I=1,N1
      WYP=0.
      WXPP=0.
C          COMPUTE (WP(I),XP(I)), AND OMEGA(I)=(F,P(I))
      DO 200 J=1,M
      TEMP=W(J)*S1(J,2)
      IF (I.LT.N1) WXPP=WXPP+TEMP*X(J)*S1(J,2)
  200 WYP=WYP+TEMP*Y(J)
C          S(I)=OMEGA(I)/W(I,I)
      S1(I,5)=WYP/WPP
C          DELTA**2(I)=DELTA**2(I-1)-S(I)**2W(I,I)
      DSQ=DSQ-S1(I,5)*S1(I,5)*WPP
C          SIGMA**2(I)=DELTA**2(I)/(DEGREES OF FREEDOM)
C          SIGMA**2(M)=0
      NR = M-I
```

```
      IF (BR.NE.0.) BR = DSO/BR
      S1(I,6) = BR
C          THRU 240 - CALCULATIONS FOR NEXT I, SKIP WHEN I=N+1
C          ALPHA(I+1)=(WP(I),XP(I))/W(I,I)
      IF (I.GE.N1) GO TO 240
      S1(I,3)=WXPP/WPP
      WPPO=WPP
      WPP=0.
      BT=S1(I,4)
      AL=S1(I,3)
C          P(I+1)=(X-ALPHA(I+1))P(I)-BETA(I)P(I-1)
C          W(I+1,I+1)=(WP(I+1),P(I+1)), B(I+1)=W(I+1,I+1)/W(I,I)
      DO 230 J=1,M
      TEMP=(X(J)-AL)*S1(J,2)-BT*S1(J,1)
      WPP=WPP+W(J)*TEMP**2
      S1(J,1)=S1(J,2)
  230 S1(J,2)=TEMP
      S1(I+1,4)=WPP/WPPO
  240 CONTINUE
C
C  ***  COMPUTE COEFFICIENTS OF LEAST SQUARES POLYNOMIAL
C        A = S(0)P(0)+...+S(N)P(N)   ***
C          P(0)=1, P(-1)=0, A(INITIAL)=S(0)*(0)=S(0)
      DO 300 IR=1,N1
      A(IR) = 0.
      S1(IR,1)=0.
  300 S1(IR,2)=0.
      S1(1,2)=1.
      A(1)=S1(1,5)
C          SUM LOOP THRU 310, I = ORDER OF POLYNOMIAL FORMED
      DO 310 I=1,N
      AL=S1(I,3)
      BT=S1(I,4)
      T2=0.
      I1 = I+1
C          FORM P(I+1)=XP(I)-ALPHA(I+1)P(I)-BETA(I)P(I-1)
C          AND ADD TO POLYNOMIAL SUM IN A
      DO 310 IR=1,I1
      T1=T2-AL*S1(IR,2)-BT*S1(IR,1)
      T2=S1(IR,2)
      S1(IR,1)=S1(IR,2)
      S1(IR,2)=T1
  310 A(IR)=A(IR)+T1*S1(I+1,5)
      IER = 0
  950 RETURN
      END
```

```
      SUBROUTINE IJTRAJ(XP,YP,IT,JT,OI,OJ,MFLAG)
C
C          THIS SUBROUTINE LOCATES A TRAJECTORY POINT IN
C          THE COMPUTATIONAL GRID
C
C        IT,JT IS LOWER LEFT CORNER OF GRID SECTION CONTAINING (XP,YP)
C
      COMMON /BLUNT/ THETA(25),RP(20,25),NBLUNT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      LEVEL 2, ANG,DXTH,DEG
      COMMON /DRD/ ANG(20,100),DXTH(100),DEG
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
C
C          POINT IS IN POLAR COORDINATE REGION
C
      IF (YP .GE. 0.0) GO TO 50
      THTA=ATAN2(YP,-XP)*DEG
      R=SQRT(YP**2+XP**2)
      DO 10 J=2,NBLUNT
      IF (THETA(J) .GT. THTA) GO TO 20
   10 CONTINUE
      J=NBLUNT
   20 JT=J-1
      OJ=(THETA(J)-THTA)/DXTH(JT)
      R2=RP(1,JT)*OJ+RP(1,JT+1)*(1.0-OJ)
      IF (R .LT.R2) GO TO 100
      DO 30 I=2,NRMAX
      R1=R2
      R2=RP(I,JT)*OJ+RP(I,JT+1)*(1.0-OJ)
      IF (R2 .GT. R) GO TO 40
   30 CONTINUE
      GO TO 150
   40 IT=I-1
      OI=(R2-R)/(R2-R1)
      MFLAG=0
      RETURN
C
C          POINT IS IN RECTANGULAR COORDINATE REGION
C
   50 CONTINUE
      NX1=NBLUNT+1
      DO 60 J=NX1,NXMAX
      IF (XC(1,J) .GT.XP) GO TO 70
   60 CONTINUE
      J=NXMAX
   70 JT=J-1
      OJ=(XC(1,JT+1)-XP)/DXTH(JT)
      Y2=YC(1,JT)*OJ+YC(1,JT+1)*(1.0-OJ)
      IF (YP .LT. Y2) GO TO 100
      DO 80 I=2,NRMAX
      Y1=Y2
      Y2=YC(I,JT)*OJ+YC(I,JT+1)*(1.0-OJ)
      IF (Y2 .GT. YP) GO TO 90
   80 CONTINUE
      GO TO 150
   90 IT=I-1
      OI=(Y2-YP)/(Y2-Y1)
      MFLAG=0
      RETURN
C
C          POINT IS INSIDE IONOPAUSE
C
  100 CONTINUE
      MFLAG=-1
      RETURN
C
C          POINT IS BEYOND BOW SHOCK
C
  150 CONTINUE
      MFLAG=1
      RETURN
C
      END
```

```
      SUBROUTINE INPUT
C
C     THIS ROUTINE READS ALL DATA REQUIRED FOR ONE CASE,
C          EXCEPT FLOW FIELD DATA FOR RERUN
C
      COMMON /RTN/ ANGP,ANGN,KRCON,RCON(20)
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,FNT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      COMMON /CONT/ KVCON,VCON(20),KRCON,RCON(20)
      COMMON /BLUNT/ THETA(25),RP(20,25),NBLUNT
      COMMON /ONSTRM/ ZPLOT,NZEND,NZADD,NXPLOT
      COMMON/JOE/ZL1,CF1,CF2,ZLF,ZTRAN,DZTRAN
      LOGICAL LGRAV
      COMMON/NUBOD/XX(100),YY(100),NBOD,LGRAV
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      LOGICAL LRERUN,LPRFL,LPRST,LPRCON,LPRR,LPLOT,LTRAJ,LRSTRT
      COMMON /PROPT/ LRERUN,LPRFL,LPRST,LPRCON,LPRR,LPLOT,LTRAJ,LRSTRT
      COMMON /TRJDAT/ NTRAJ,TTRAJ(100),XTRAJ(100),YTRAJ(100),ZTRAJ(100),
     *  VTRAJ(100),VXTRAJ(100),VYTRAJ(100),VZTRAJ(100),TMPTRJ(100),
     *  BTRAJ(100),BXTRAJ(100),BYTRAJ(100),BZTRAJ(100),ROTRAJ(100),
     *  RTRAJ(100)
      COMMON /MKPLOT/ NMARKT,MARKT(12)
      LOGICAL LPLTRJ
      COMMON /TROPT/ LPLTRJ,RPLNT,VINF,RHOINF,TMPINF,BINF
      LOGICAL LSUN
      COMMON /SUNDAT/ LSUN,XTRAJS(100),YTRAJS(100),ZTRAJS(100),
     *              RTRAJS(100),AZANG,POLANG,RX1,RY1,RZ1
      DIMENSION TITLE(8)
      DIMENSION NK(100)
      DIMENSION RR(100)
      EQUIVALENCE (RR(1),YY(1))
      DATA MBLANK/1H /,MSTAR/1H*/
C
C     READ CARD INPUT FOR ONE COMPLETE CALCULATION
C
      READ(5,100) TITLE
```

```
      READ(5,110) AMACH,GAMMA,HRO,XCALC,NR,NBLUNT,CN,ITER
      HORN=HRO
      READ(5,120) LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      READ(5,130) XPLOT,ANGP,ANGN,NXADD,LGRAV
      IF (LPLOT) LPRCON=.TRUE.
      IF (XPLOT .GE. 0.0) XPLOT=-1.0
      ZPLOT=-XPLOT
      READ(5,140) KVCON
      IF (KVCON .GT. 0)  READ(5,160) (VCON(I),I=1,KVCON)
      READ(5,140) KRCON
      IF (KRCON .GT. 0)  READ(5,160) (RCON(I),I=1,KRCON)
      READ(5,140) KBCON
      IF (KBCON .GT. 0)  READ(5,160) (BCON(I),I=1,KBCON)
      IF (.NOT. LTRAJ) GO TO 5
      READ(5,170) NTRAJ,(TTRAJ(I),XTRAJ(I),YTRAJ(I),ZTRAJ(I),I=1,NTRAJ)
      READ(5,190) LPLTRJ,RPLNT,VINF,RHOINF,TMPINF,BINF
      READ(5,140) NMARKT
      IF (NMARKT .GT. 0) READ(5,180) (MARKT(I),I=1,NMARKT)
      READ(5,190) LSUN,AZANG,POLANG,BX1,BY1,BZ1
      IF (LSUN) CALL ROTAT(1)
    5 CONTINUE
      NHINDX=0
      IF(HORN.GT.0.0) NHINDX=1
      IF (LRERUN) GO TO 10
      IF (HORN .LT. 0.0) READ(5,150) NBOD,(XX(I),RR(I),I=1,NBOD)
C
C      INITIALIZE DEFAULT PARAMETERS
C
      IF (XCALC .GE. 0.0) XCALC=-1.0
      ZLF=-XCALC
      IF (NR .LE. 0) NR=19
      IF (NBLUNT .LE. 0) NBLUNT=24
      IF (CN .LE. 0.0) CN=3.0
      IF(ITER.LE.0) ITER=300
      IF (NXADD .LE. 0) NXADD=9
      JMAX=NBLUNT+NXADD+1
      NXMAX=JMAX-1
      NRMAX=NR
      KMAX=NR
      NZEND=JMAX-1
      NZADD=0
      XMACH=AMACH
      GAM=GAMMA
C
C     WRITE OUT INPUT AND DEFAULT VALUES TO BE USED
C
   10 CONTINUE
      WRITE(6,200) TITLE
      IF (LRERUN) RETURN
      WRITE(6,210) AMACH,GAMMA
      WRITE(9) NBLUNT,NR,AMACH,GAMMA,HORN
      IF (HORN) 14,16,18
   14 WRITE(6,214) NBOD,(XX(I),RR(I),I=1,NBOD)
      WRITE(9) NBOD,(XX(I),RR(I),I=1,NBOD)
      GO TO 20
   16 WRITE(6,216)
      GO TO 20
   18 IF (LGRAV) GO TO 19
      WRITE (6,218) HORN
      GO TO 20
   19 WRITE (6,219) HORN
   20 CONTINUE
      WRITE(6,220) NR,NBLUNT,NXADD,CN,ITER,XCALC
      WRITE (6,230) XPLOT
      IF (LSUN) GO TO 17
      WRITE (6,231) ANGP,ANGN
   17 CONTINUE
      WRITE(6,235) LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      IF (.NOT. LTRAJ) GO TO 25
      RCRP=1.0/RPLNT
      WRITE(6,260) LPLTRJ,RCRP,VINF,RHOINF,TMPINF
      IF (LSUN) WRITE (6,269) BINF,BX1,BY1,BZ1,AZANG,POLANG
      WRITE(6,263) NTRAJ,NMARKT
      DO 21 N=1,NTRAJ
   21 NK(N)=NBLANK
      IF (NMARKT .LE. 0) GO TO 24
      DO 22 I=1,NMARKT
      K=MARKT(I)
   22 NK(K)=NSTAR
   24 CONTINUE
      WRITE (6,268)
      WRITE (6,267) (N,NK(N),TTRAJ(N),XTRAJS(N),YTRAJS(N),ZTRAJS(N),
     *              N=1,NTRAJ)
   25 CONTINUE
      WRITE(6,240)
      WRITE(6,242) KVCON
      IF (KVCON .GT. 0)  WRITE(6,250) (VCON(I),I=1,KVCON)
      WRITE(6,244) KRCON
      IF (KRCON .GT. 0)  WRITE(6,250) (RCON(I),I=1,KRCON)
      WRITE(6,246) KBCON
      IF (KBCON .GT. 0)  WRITE(6,250) (BCON(I),I=1,KBCON)
      DTOR=.01745329252
      ANGP=ANGP*DTOR
      ANGN=ANGN*DTOR
      RETURN
C
  100 FORMAT(8A10)
  110 FORMAT(4F10.0,2I10,F10.0,I10)
  120 FORMAT(8L10)
  130 FORMAT (3F10.0,I10,L10)
  140 FORMAT(I10)
  150 FORMAT(I10/(2F10.0))
  160 FORMAT(8F10.0)
  170 FORMAT(I10/(4F10.0))
  180 FORMAT(8I10)
  190 FORMAT(L10,5F10.0)
C
  200 FORMAT(1H1//25X,8A10///55X,15HINPUT VARIABLES/55X,15(1H*))
  210 FORMAT(//20X,25HINTERPLANETARY MACH NO. =,F6.2
     *       ///20X,21HSPECIFIC HEAT RATIO =,F6.3)
  214 FORMAT(//20X,46HOBSTACLE GEOMETRY: USER-SUPPLIED COORDINATES -,
     *        I4,7H POINTS//33X,3HX/D,11X,3HR/D/
     *       (30X,F9.4,6X,F8.4))
  216 FORMAT(//20X,69HOBSTACLE GEOMETRY: DEFAULT MAGNETOPAUSE COORDINAT,
     *        21HES - EQUATORIAL TRACE)
  218 FORMAT(//20X,42HOBSTACLE GEOMETRY: DEFAULT IONOPAUSE SHAPE,
     *        /,39X,36HFOR CONSTANT SCALE HEIGHT WITH H/RO=,F4.2)
  219 FORMAT(//20X,42HOBSTACLE GEOMETRY: DEFAULT IONOPAUSE SHAPE,
     *        /,39X,31HWITH GRAVITATIONAL VARIATION IN,
     *        /,39X,19HSCALE HEIGHT, H/RO=,F4.2)
  220 FORMAT(//20X,37HPARAMETERS FOR BLUNT BODY CALCULATION//
     *       25X,28HNO. OF RADIAL MESH POINTS  =,I4/
     *        25X,28HNO. OF ANGULAR MESH POINTS =,I4/
     *        25X,27HNO. OF ADDITIONAL POINTS IN/
     *        35X,18HBLUNT BODY MESH  =,I4/
     *        25X,14HCOURANT NUMBER,13X,1H=,F5.2/
     *        25X,17HNO. OF ITERATIONS,10X,1H=,I4/
     *       //20X,42HTERMINAL DOWNSTREAM LOCATION FOR MARCHING ,
     *        ,7HCALCULATION,X/RD=F6.2)
  230 FORMAT(//20X,47HTERMINAL DOWNSTREAM LOCATION FOR PLOTTING,X/RD=,
     *        F6.2)
  231 FORMAT(//20X,53HUSER SPECIFIED DEVIATION IN SOLAR-WIND COORDINATES
     *        /20X,47HOF MAGNETIC FIELD SYMMETRY-PLANE COMPONENT FROM,
     * /20X,17H FLOW DIRECTION =,F7.2,8H DEGREES
     *        //20X,50HUSER SPECIFIED DEVIATION IN SOLAR-WIND COORDINATES
     *        /20X,58HOF MAGNETIC FIELD FROM MAGNETIC SYMMETRY PLANE =,
     *        F7.2,9H DEGREES)
  235 FORMAT(//20X,8HLRERUN =,L2//20X,8HLPRFL  =,L2//20X,8HLPRST  =,L2,
     *        //20X,8HLPRCON =,L2//20X,8HLPRB   =,L2//20X,8HLPLOT  =,L2
     *        //20X,8HLTRAJ  =,L2//20X,8HLRSTRT =,L2)
  240 FORMAT(1H1//45X,40HVALUES SPECIFIED FOR CONTOUR CALCULATION/
     *        45X,40(1H-))
  242 FORMAT(////20X,I2,29H CONTOUR LEVELS FOR VELOCITY:)
  244 FORMAT(////20X,I2,29H CONTOUR LEVELS FOR DENSITY:)
  246 FORMAT(////20X,I2,44H CONTOUR LEVELS FOR MAGNETIC FIELD STRENGTH:)
  250 FORMAT(/(10X,6F10.3))
  260 FORMAT(/20X,9HLPLTRJ =,L2/
     *  1H1//50X,32HINPUT FOR TRAJECTORY CALCULATION/
     *  50X,32(1H*)////
     *  /20X,34HRO/RPLANET                        =,F9.4/
     *  /20X,34HINTERPLANETARY VELOCITY           =,1PE11.2/
     *  /20X,34HINTERPLANETARY DENSITY            =,E11.3/
     *  /20X,34HINTERPLANETARY TEMPERATURE        =,E11.3/
     *  /20X,34HINTERPLANETARY MAGNETIC FIELD      )
  263 FORMAT(////25X,7HNTRAJ =,I4,15X,9HNMARKT =,I3/
     *  30X,45H( *  = POINT TO BE MARKED FOR CROSS REFERENCE)///
     *  28X,1HN,9X,5HTTRAJ,8X,5HXTRAJ,7X,5HYTRAJ,7X,5HZTRAJ/)
  267 FORMAT((25X,I4,1X,A1,3X,F10.4,5X,F8.4,2(4X,F8.4)))
  268 FORMAT(44X,32H(SUN-PLANET COORDINATE SYSTEM))
  269 FORMAT(35X,16H  MAGNITUDE  =,E11.3
     *       /35X,14HX-COMPONENT  =,E11.3
     *       /35X,14HY-COMPONENT  =,E11.3
     *       /35X,14HZ-COMPONENT  =,E11.3
     *      //20X,29HAZIMUTHAL ANGLE                  =,E11.3/
     *       /20X,29HPOLAR ANGLE                      =,E11.3)
      END
```

```
      SUBROUTINE LABEL
C
C     THIS ROUTINE ELIMINATES OVERLAP IN PLOTTING A
C          SET OF CONTOUR LABELS.
C
      COMMON /LABLS/ XLAB(30),YLAB(30),CV(30),NCL,ILB(30),NLAB
      COMMON /SCALE/ XSF,YSF,XMAX,YMAX,XLNGTH,YLNGTH
C
C     SORT THE ARRAY YLAB INTO ASCENDING ORDER.  CHANGE
C          THE ORDER OF XLAB AND CV TO CORRESPOND.
C
      CALL SUBBL(XLAB,YLAB,CV,NCL,ILB,NLAB)
C
C
C     FIND FIRST LABEL FROM SORTED ARRAY THAT IS ABOVE X-AXIS.
C
      DO 5 K=1,NCL
      IF(YLAB(K).LT.0.) GO TO 5
      KMIN=K
      GO TO 10
    5 CONTINUE
C
C     NO LABELS FOUND ABOVE X-AXIS.  BAD SET OF CONTOUR VALUES
C          SPECIFIED.  STOP PROGRAM.
C
      WRITE(6,200)
  200 FORMAT( 1H1,10H**********,1)X,48HUNABLE TO LABEL CONTOUR LINES BEC
     1AUSE OF OVERLAP,10X,10H**********)
      STOP
C
   10 ILB(1)=KMIN
      NLAB=1
      IF(NCL.EQ.1) RETURN
C
C     EXAMINE ALL LABELS BEYOND LABEL KMIN FOR OVERLAP.
C
      K=2
      KMINP=KMIN+1
      DO 15 I=KMINP,NCL
      IF(YLAB(I).LT.YLAB(KMIN)+.1/YSF.AND.XLAB(I).LT.XLAB(KMIN)
     1   +.4/XSF) GO TO 15
C
C     SAVE ARRAY INDEX FOR LABEL TO BE PLOTTED.
C
      KMIN=I
      ILB(K)=I
      NLAB=K
      K=K+1
   15 CONTINUE
C
C     ENSURE THAT THE LAST LABEL IS PLOTTED.
C
      ILB(NLAB)=NCL
C
C
      RETURN
      END
```

```
      SUBROUTINE MAP  (A,JDIM,X,Y,NLIN,KOD,ACONT,NAD,ISIZ1,ISIZ2,ICHK,
     *                 JMIN,KMIN,JMAX,KMAX)
C
C         CONTOUR PROGRAMS MAP, WALK, SERCH, ENTER, AND CHECK
C         WRITTEN BY REESE SORENSON, NASA-AMES RES. CTR., AUG., 1974.
C           (MODIFIED VERSION)
C
C         THIS SUBROUTINE AND THOSE WHICH IT CALLS PREPARE DATA
C         FOR CONTOUR PLOTTING.
C
C           CALLING PARAMETERS:
C
C         A        TWO-DIMENSIONAL ARRAY TO BE CONTOUR PLOTTED.
C         JDIM     FIRST-POSITION DIMENSION-SIZE OF A.
C         X,Y      ARRAYS TO CONTAIN CONTOUR LINE DATA UPON RETURN.
C                  EACH PAIR OF X(I) AND Y(I) REPRESENTS A POINT ON A
C                  CONTOUR LINE.
C         NLIN     IS THE NUMBER OF CONSTANT-A CONTOUR LEVELS.
C         KOD      =1 IF CONTOUR LEVELS ARE TO BE COMPUTED INTERNALLY
C                  BY LINEARLY INTERPOLATING BETWEEN THE MAXIMUM AND
C                  MINIMUM VALUES OF A.
C                  =2 IF CONTOUR LEVELS ARE TO BE GIVEN BY THE USER.
C         ACONT    ARRAY OF CONTOUR LEVELS.  SCRATCH IF KOD=1.
C                  IF KOD=2, ACONT MUST BE FILLED WITH MONOTONICALLY
C                  INCREASING CONTOUR LEVELS.  SHOULD BE DIMENSIONED
C                  AT LEAST NLIN.
C         NAD      IT SHOULD BE RECOGNIZED THAT MANY CONTOUR LINES
C                  MIGHT RESULT FROM EACH CONTOUR LEVEL.
C                  UPON RETURN, NAD(1) IS THE TOTAL NUMBER OF CONTOUR
C                  LINES.  ALL THE LINES ARE RUN-TOGETHER, HEAD TO TAIL
C                  IN ARRAYS X AND Y.  THE FIRST LINE STARTS AT X(1)
C                  AND Y(1) AND ENDS AT X(I) AND Y(I) FOR I=NAD(2)-1.
C                  THE N-TH CONTOUR LINE STARTS AT X(I) AND Y(I) FOR
C                  I=NAD(N) AND ENDS AT X(I) AND Y(I) FOR I=NAD(N+1)-1.
C                  NAD(N) FOR N.GE.2 IS THE STARTING ADDRESS IN X AND Y
C                  OF THE N-TH CONTOUR LINE.
C         ISIZ1    DIMENSION SIZE OF X AND Y.
C         ISIZ2    DIMENSION SIZE OF NAD.  IT IS DIFFICULT TO SAY
C                  HOW LARGE X, Y, AND NAD SHOULD BE DIMENSIONED.
C                  THIS DEPENDS ON THE DIMENSION SIZE OF ARRAY A,
C                  THE VALUE OF NLIN, AND THE DEGREE OF ECCENTRICITY
C                  OF THE SURFACE REPRESENTED BY ARRAY A.  ERROR MESSAGES
C                  WILL RESULT IF THESE ARRAYS ARE TOO SMALL.
C         ICHK     AN ARRAY TO BE USED AS SCRATCH BY THE CONTOUR PROGRAMS.
C                  IT SHOULD HAVE AT LEAST 4*ISIZ1 CELLS.
C         JMIN,KMIN,JMAX,KMAX      IT IS RECOGNIZED THAT ONE MIGHT
C                  WISH TO CONTOUR PLOT AN ARRAY A FOR SUBSCRIPTS
C                  STARTING AT SOME VALUE LARGER THAN 1.  THIS SUBROUTINE
C                  WILL PROCESS ARRAY A(J,K) FOR JMIN.LE.J.LE.JMAX,
C                  AND KMIN.LE.K.LE.KMAX.  JMIN, KMIN, JMAX, AND KMAX
C                  ARE THE LIMITS ON THE SUBSCRIPTS.  JMIN AND KMIN MAY
C                  BE 1.
C
C
      DIMENSION A(1),X(1),Y(1),ACONT(1),NAD(1),ICHK(4,1)
C
      IDEX(J,K)=J+(K-1)*JDIM
C
C        FIND AMAX AND AMIN
      I=IDEX(JMIN,KMIN)
      AMAX=A(I)
      AMIN=AMAX
      DO 1 K=KMIN,KMAX
      DO 1 J=JMIN,JMAX
      I=IDEX(J,K)
      VAL=A(I)
      IF(VAL-AMAX)2,1,3
3     AMAX=VAL
      GO TO 1
2     IF(VAL-AMIN)4,1,1
4     AMIN=VAL
1     CONTINUE
C
C        FIND ACONT BY LINEARLY INTERPOLATING
      GO TO (5,6),KOD
5     DIF=(AMAX-AMIN)/FLOAT(NLIN+1)
      DO 7 N=1,NLIN
7     ACONT(N)=AMIN+FLOAT(N)*DIF
      NMIN=1
      NMAX=NLIN
      GO TO 8
C
C        CHECK ACONT IF GIVEN BY USER
6     NMIN=0
10    NMIN=NMIN+1
      IF(NMIN.GT.NLIN)  GO TO 9
      IF(AMIN.GT.ACONT(NMIN))  GO TO 10
C
      NMAX=NLIN+1
11    NMAX=NMAX-1
      IF(NMAX.LT.1)  GO TO 9
      IF(AMAX.LT.ACONT(NMAX))  GO TO 11
      IF(NMIN.GT.NMAX)  GO TO 9
C
      IF (NMIN.EQ.NMAX) GO TO 12
      NST=NMIN+1
      DO 13 N=NST,NMAX
      IF(ACONT(N).LE.ACONT(N-1))  GO TO 9
13    CONTINUE
   12 CONTINUE
C
C        PART 1:  GO AROUND THE BOUNDARY LOOKING FOR LINES WHICH
C        INTERSECT THE BOUNDARY.
8     NAD(1)=0
      J=JMIN
      K=KMIN
```

```
      NINT=2*(JMAX-JMIN)+2*(KMAX-KMIN)
      KOD7=2
      KOD2=1
      I=IDEX(JMIN,KMIN)
      AA=A(I)
      NXY=0
C
      DO 20 N=1,NINT
      GO TO (21,22,23,24),KOD7
C
C        ORIENTATION 1: UPWARDS
21    J=J-1
      I=IDEX(J,K)
      IF(J.GE.JMIN)  GO TO 25
      J=JMIN
      I=IDEX(J,K+1)
      KOD7=2
      KOD2=1
      GO TO 25
C
C        ORIENTATION 2: TO THE RIGHT
22    K=K+1
      I=IDEX(J,K+1)
      IF(K.LT.KMAX)  GO TO 25
      I=IDEX(J+1,K)
      KOD7=3
      KOD2=2
      GO TO 25
C
C        ORIENTATION 3:  DOWNWARD
23    J=J+1
      I=IDEX(J+1,K)
      IF(J.LT.JMAX)  GO TO 25
      K=KMAX-1
      I=IDEX(J,K)
      KOD7=4
      KOD2=1
      GO TO 25
C
C        ORIENTATION 4:  TO THE LEFT
24    K=K-1
      I=IDEX(J,K)
      IF(K.GE.KMIN) GO TO 25
      K=KMIN
      J=JMAX-1
      I=IDEX(J,K)
      KOD7=1
      KOD2=2
C
C        FIND AH AND AL
25    AB=AA
      AA=A(I)
      IF(AB-AA)26,26,27
26    AH=AA
      AL=AB
      GO TO 28
27    AH=AB
      AL=AA
C
C        CHECK TO SEE IF A CONTOUR LINE PASSES THROUGH THE INTERVAL
C        UNDER CONSIDERATION.
28    KOD8=1
      DO 29 NVAL=NMIN,NMAX
      VAL=ACONT(NVAL)
      GO TO (37,30),KOD8
37    IF(VAL.LT.AL)  GO TO 29
      KOD8=2
30    IF(VAL.GT.AH)  GO TO 20
C
C        CHECK TO SEE IF THE CONTOUR LINE POINT JUST FOUND IS A
C        NEW ONE, OR THE TAIL END OF ONE ALREADY FOUND.
      CALL CHECK(ICHK,NXY,KOD2,J,K,NVAL,KOD9)
      IF(KOD9.EQ.2)  GO TO 29
C
C        DETERMINE A1 AND A2
      IF(KOD7.EQ.1.OR.KOD7.EQ.4)  GO TO 32
      A2=AA
      A1=AB
      GO TO 33
32    A2=AB
      A1=AA
C
C        ENTER IN THE TABLES THE POINT JUST FOUND.
33    NDUM=NAD(1)+1
      IF(NDUM.GT.ISIZ2)  GO TO 35
      NAD(1)=NDUM
      CALL ENTER(KOD2,J,K,NVAL,A1,A2,JMIN,KMIN,ICHK,KOD4,X,Y,NXY,ACONT,
     *           ISIZ1)
      IF(KOD4.EQ.2)  GO TO 70
      NAD(NDUM)=NXY
C
C        NOW FOLLOW UP THIS LEAD,  WALK THE LENGTH OF THE LINE.
      KOD5=KOD7
      KOD6=2
      JS=J
      KS=K
36    CALL WALK(KOD5,JS,KS,A,JDIM,ICHK,JMIN,JMAX,KMIN,KMAX,KOD4,NXY,
     *           ACONT,KOD6,X,Y,NVAL,ISIZ1)
      IF(KOD4.EQ.2)  GO TO 70
      IF(KOD5.GT.0)  GO TO 36
C
29    CONTINUE
20    CONTINUE
C
C
C        PART 2: SCAN THE WHOLE FIELD FOR LINES THAT DON'T INTERSECT
C        THE BOUNDARY
      JE=JMAX-1
      KE=KMAX-1
      DO 40 K=KMIN,KE
      I=IDEX(JMIN,K)
      A2=A(I)
      DO 40 J=JMIN,JE
      A1=A2
      DO 44 KOD2=1,2
      GO TO (41,42),KOD2
41    IF(J.EQ.JMIN)  GO TO 44
      I=IDEX(J,K+1)
      A2=A(I)
      GO TO 43
42    I=IDEX(J+1,K)
      A2=A(I)
      IF(K.EQ.KMIN)  GO TO 44
43    CONTINUE
C
C        FIND AH AND AL
      IF(A1-A2)45,45,46
45    AH=A2
      AL=A1
      GO TO 47
46    AH=A1
      AL=A2
C
C        CHECK TO SEE IF A CONTOUR LINE PASSES THROUGH THE INTERVAL
C        UNDER CONSIDERATION.
47    KOD8=1
      DO 48 NVAL=NMIN,NMAX
      VAL=ACONT(NVAL)
      GO TO (49,50),KOD8
49    IF(VAL.LT.AL)  GO TO 48
      KOD8=2
50    IF(VAL.GT.AH)  GO TO 44
C
C        WE HAVE FOUND A POINT ON A CONTOUR LINE.  CHECK TO SEE IF
C        IT IS A NEW ONE, OR IF WE HAVE ALREADY DONE THIS LINE.
      CALL CHECK(ICHK,NXY,KOD2,J,K,NVAL,KOD9)
      IF(KOD9.EQ.2)  GO TO 48
C
C        WE HAVE A POINT.  ENTER IT IN THE TABLES.
      NDUM=NAD(1)+1
      IF(NDUM.GT.ISIZ2)  GO TO 35
      NAD(1)=NDUM
      CALL ENTER(KOD2,J,K,NVAL,A1,A2,JMIN,KMIN,ICHK,KOD4,X,Y,NXY,ACONT,
     *           ISIZ1)
      IF(KOD4.EQ.2)  GO TO 70
      NAD(NDUM)=NXY
C
C        NOW WALK THE LENGTH OF THE LINE.
      KOD5=KOD2
      KOD6=2
      NXYST=NXY
      JS=J
      KS=K
56    CALL WALK(KOD5,JS,KS,A,JDIM,ICHK,JMIN,JMAX,KMIN,KMAX,KOD4,NXY,
     *           ACONT,KOD6,X,Y,NVAL,ISIZ1)
      IF(KOD4.EQ.2)  GO TO 70
      IF(KOD5.GT.0)  GO TO 56
C
```

```
C        IF THIS LINE IS A CLOSED CURVE, CLOSE IT BY ENTERING FIRST
C        POINT AGAIN.
      DIST=SQRT((X(NXYST)-X(NXY))**2+(Y(NXYST)-Y(NXY))**2)
      IF(DIST.GT.1.42)  GO TO 40
      NXY=NXY+1
      IF(NXY.GT.ISIZ1)  GO TO 60
      X(NXY)=X(NXYST)
      Y(NXY)=Y(NXYST)
      ICHK(1,NXY)=ICHK(1,NXYST)
      ICHK(2,NXY)=ICHK(2,NXYST)
      ICHK(3,NXY)=ICHK(3,NXYST)
      ICHK(4,NXY)=ICHK(4,NXYST)
C
48    CONTINUE
44    CONTINUE
40    CONTINUE
C
C        ENTER END OF LAST LINE IN BOOKKEEPING ARRAYS.
      NXY=NXY+1
70    NDUM=NAD(1)+1
      IF(NDUM.GT.ISIZ2)  GO TO 35
      NAD(NDUM)=NXY
      GO TO 34
C
C        WRITE ERROR MESSAGES.
35    WRITE(6,102)
  102 FORMAT(47H1CONTOUR SEARCH ABORTED - TABLE OVERFLOW IN NAD)
      NAD(1)=ISIZ2-1
      GO TO 34
C
9     WRITE(6,103)
  103 FORMAT(1H1,15X,5(1H*),2X,20HEXECUTION TERMINATED,2X,5(1H*)//
     *   11X,44HARRAY OF CONTOUR VALUES IMPROPERLY SPECIFIED)
      STOP
C
60    WRITE(6,104)
      GO TO 70
  104 FORMAT(49H1CONTOUR SEARCH ABORTED - TABLE OVERFLOW IN (X,Y))
C
34    RETURN
      END
```

```
      SUBROUTINE MRKPLT(X,Y,XSF,YSF)
C
C        THIS SUBROUTINE MARKS WITH A SYMBOL AND A LETTER, A-H AND J-M,
C        THOSE POINTS WHICH ARE TO BE FLAGGED TO FACILITATE
C        CROSS-REFERNCING BETWEEN THE VARIOUS PLOTS FOR THE TRAJECTORY
C
      COMMON /MKPLOT/ NMARKT,MARKT(12)
      DIMENSION X(1),Y(1)
      DIMENSION MARK(12)
C
      DATA MARK/1HA,1HB,1HC,1HD,1HE,1HF,1HG,1HH,1HJ,1HK,1HL,1HM/
C
      IF (NMARKT.EQ.0) RETURN
      DO 100 I=1,NMARKT
      N=MARKT(I)
      XP=X(N)
      YP=Y(N)
      CALL GRAF(XP,YP,.03,3)
      XCH=XP+0.07/XSF
      YCH=YP-.04/YSF
      CALL CHAR(XCH,YCH,0.0,0.8,MARK(I),1)
  100 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE PLOTCN
C
C     THIS ROUTINE CONTROLS THE DRAWING OF THE PLOTS
C             UCC PLOT SUBROUTINES CALLED ARE
C                DASH,ENPLT,PLOT.
C
C     IPLOT  = 1      FOR VELOCITY CONTOUR PLOT
C     IPLOT  = 2      FOR DENSITY CONTOUR PLOT
C     IPLOT  = 3      FOR STREAMLINE PLOT
C     IPLOT  = 4      FOR TEMPERATURE PLOT
C     IPLOT  = 5      FOR PARALLEL B-FIELD COMPONENT PLOT
C     IPLOT  = 6      FOR PERPENDICULAR B-FIELD COMPONENT PLOT
C     IPLOT  = 7      FOR NORMAL B-FIELD COMPONENT PLOT
C
      COMMON /PLOTC/ CONTX(1000),CONTY(1000),CVAL(30),NAD(30),IPLOT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NBMAX,NXMAX,AMACH,GAMMA,HPD,NHINDX
      COMMON /SCALE/ XSF,YSF,XMAX,YMAX,XLNGTH,YLNGTH
C
C
C     PLOT STREAMLINES
C          THE FIRST CALL TO PLOT INITIALIZES THE PLOT.
C          SCALE FACTORS ARE EQUAL TO 1.
C     SUBROUTINE SETUP ESTABLISHES PERMANENT PLOT ORIGIN,
C          DRAWS AXES, LABELS, AND TITLE.
C     SUBROUTINE BOUND DRAWS AND LABELS SHOCK WAVE, PLANET,
C          AND MAGNETOSPHERE (IBODY=0) OR IONOSPHERE
C          (IBODY=1) BOUNDARY.
C
      IF(IPLOT.NE.3) GO TO 10
      CALL DASH(0.0,0.0,0.0,0.0,1)
      CALL PLOT(3.0,-12.0,-3)
      CALL SETUP(IPLOT,AMACH,GAMMA,NHINDX)
      CALL BOUND
      CALL CONTR
      CALL PLOT(XMAX,0.0,-3)
      CALL RESET
      CALL PLOT(2.0,0.0,-3)
      RETURN
C
C
C     DRAW CONTOUR PLOTS
C
   10 CONTINUE
      CALL PLOT(2.0,-12.0,-3)
      CALL SETUP(IPLOT,AMACH,GAMMA,NHINDX)
      CALL BOUND
      CALL CONTR
      CALL PLOT(XMAX,0.0,-3)
      CALL RESET
      CALL PLOT(2.0,0.0,-3)
      IF(IPLOT.EQ.1) GO TO 20
      RETURN
C
C     CHANGE VELOCITY CONTOUR VALUES TO TEMPERATURE
C
   20 IPLOT=4
      FACT=0.5*(GAMMA-1.0)*AMACH*AMACH
      NVAL=NAD(1)
      DO 25 I=1,NVAL
      CVAL(I)=1.0+FACT*(1.0-CVAL(I)**2)
   25 CONTINUE
      GO TO 10
      END
```

```
      SUBROUTINE PLOTER
C
C        THIS SUBROUTINE CONTROLS THE DRAWING OF THE TIME HISTORY PLOTS
C        OF VELOCITY, TEMPERATURE, DENSITY, AND MAGNETIC FIELD
C
      COMMON /TRJDAT/ NTRAJ,TTRAJ(100),XTRAJ(100),YTRAJ(100),ZTRAJ(100),
     *  VTRAJ(100),VXTRAJ(100),VYTRAJ(100),VZTRAJ(100),TMPTRJ(100),
     *  BTRAJ(100),BXTRAJ(100),BYTRAJ(100),BZTRAJ(100),RDTRAJ(100),
     *  RTRAJ(100)
      LOGICAL LPLTRJ
      COMMON /TROPT/ LPLTRJ,RPLNT,VINF,RHOINF,TMPINF,BINF
      DATA PLTSZE/7.0/, TLNGTH/8.0/
      DATA NSYM/6HX     /, ISYM/1/
C
C        CALCULATE SETUP PARAMETERS FOR TIME AXIS
C
      NTAXIS=0
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
C
C        PLOT V VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,VINF,PLTSZE,10)
      CALL SCALE(TIMESF,VSF,1)
```

```
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,VTRAJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,VTRAJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
C
C         PLOT VX VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,VINF,PLTSZE,11)
      CALL SCALF(TIMESF,VSF,1)
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,VXTRAJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,VXTRAJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
C
C         PLOT VY VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,VINF,PLTSZE,12)
      CALL SCALF(TIMESF,VSF,1)
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,VYTRAJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,VYTRAJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
C
C         PLOT VZ VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,VINF,PLTSZE,13)
      CALL SCALF(TIMESF,VSF,1)
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,VZTRAJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,VZTRAJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
C
C         PLOT TEMPERATURE VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,TMPINF,PLTSZE,20)
      CALL SCALF(TIMESF,VSF,1)
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,TMPTRJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,TMPTRJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
C
C         PLOT DENSITY VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,RHOINF,PLTSZE,30)
      CALL SCALF(TIMESF,VSF,1)
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,ROTRAJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,ROTRAJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
C
C         PLOT B VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,BINF,PLTSZE,40)
      CALL SCALF(TIMESF,VSF,1)
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,BTRAJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,BTRAJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
C
C         PLOT BX VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,BINF,PLTSZE,41)
      CALL SCALF(TIMESF,VSF,1)
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,BXTRAJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,BXTRAJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
C
C         PLOT BY VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,BINF,PLTSZE,42)
      CALL SCALF(TIMESF,VSF,1)
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,BYTRAJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,BYTRAJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
C
C         PLOT BZ VS TIME
C
      CALL TAXIS(TIMESF,TOFFST,TLNGTH,NTAXIS)
      CALL VAXIS(VSF,VOFFST,BINF,PLTSZE,43)
      CALL SCALF(TIMESF,VSF,1)
      CALL OFFST(TOFFST,VOFFST,1)
      CALL VECTOR(TTRAJ,BZTRAJ,NTRAJ,1,ISYM,MSYM)
      CALL MRKPLT(TTRAJ,BZTRAJ,TIMESF,VSF)
      CALL RESET
      CALL PLOT(PLTSZE,0.0,-3)
      CALL ENPLT(3.0,0.0)
C
      RETURN
      END
```

```
      SUBROUTINE PLOTTP
C
C
C        THIS SUBROUTINE PLOTS (X,R) AND (Y,Z) PROJECTIONS OF THE
C        TRAJECTORY, WHERE R=SQRT(Y*Y+Z*Z), USING THE SAME SCALE FACTOR
C        FOR BOTH PLOTS.
C        CERTAIN POINTS ARE FLAGGED TO PERMIT CROSS-REFERENCING
C        WITH THE PLOTS OF FLOW FIELD AND MAGNETIC FIELD DATA
C
      COMMON /PLTTP/ XTMAX,XTMIN,YTMAX,YTMIN,ZTMAX,ZTMIN,RTMAX
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NBMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      COMMON /TRJDAT/ NTRAJ,TTRAJ(100),XTRAJ(100),YTRAJ(100),ZTRAJ(100),
     *  VTRAJ(100),VXTRAJ(100),VYTRAJ(100),VZTRAJ(100),TMPTRJ(100),
     *  BTRAJ(100),BXTRAJ(100),BYTRAJ(100),BZTRAJ(100),ROTRAJ(100),
     *  RTRAJ(100)
      LOGICAL LPLTRJ
      COMMON /TROPT/ LPLTRJ,RPLNT,VINF,RHOINF,TMPINF,BINF
C
      DIMENSION TITLXR(2),TITLYZ(2)
      DATA PIOV2/1.570796327/
      DATA PLTSZE/6.0/
      DATA TITLEI/10HTRAJECTORY/
      DATA TITLYZ/10H(Y-Z PROJE,6HCTION)/
      DATA TITLXR/10H(X-R PROJE,6HCTION)/
C
C        ESTABLISH PLOT BOUNDS
C
      RNOSE=1.0/RPLNT
      XTMAX=0.0
      XTMIN=0.0
      YTMAX=0.0
      YTMIN=0.0
      ZTMAX=0.0
      ZTMIN=0.0
      RTMAX=0.0
      DO 70 N=1,NTRAJ
      IF (XTRAJ(N).GT.XTMAX) XTMAX=XTRAJ(N)
      IF (XTRAJ(N).LT.XTMIN) XTMIN=XTRAJ(N)
      IF (YTRAJ(N).GT.YTMAX) YTMAX=YTRAJ(N)
      IF (YTRAJ(N).LT.YTMIN) YTMIN=YTRAJ(N)
      IF (ZTRAJ(N).GT.ZTMAX) ZTMAX=ZTRAJ(N)
      IF (ZTRAJ(N).LT.ZTMIN) ZTMIN=ZTRAJ(N)
      IF (RTRAJ(N).GT.RTMAX) RTMAX=RTRAJ(N)
   70 CONTINUE
      XTMAX=AMAX1(XTMAX*RNOSE,1.5)
      XTMIN=AMIN1(XTMIN*RNOSE,-1.5)
      YTMAX=AMAX1(YTMAX*RNOSE,1.5)
      YTMIN=AMIN1(YTMIN*RNOSE,-1.5)
      ZTMAX=AMAX1(ZTMAX*RNOSE,1.5)
      ZTMIN=AMIN1(ZTMIN*RNOSE,-1.5)
      RTMAX=AMAX1(RTMAX*RNOSE,2.0)
      XTMAX=0.5*AINT(2.0*XTMAX+.999)
      XTMIN=0.5*AINT(2.0*XTMIN-.999)
      YTMAX=0.5*AINT(2.0*YTMAX+.999)
      YTMIN=0.5*AINT(2.0*YTMIN-.999)
```

```
      ZTMAX=0.5*AINT(2.0*ZTMAX+.999)
      ZTMIN=0.5*AINT(2.0*ZTMIN-.999)
      RTMAX=0.5*AINT(2.0*RTMAX+.999)
C
C       COMPUTE SCALE FACTOR
C
      YTSIZE=YTMAX-YTMIN
      TSF=PLTSZE/AMAX1(YTSIZE,RTMAX)
      TSFRO=TSF*RNOSE
      CALL PLOT(4.0,-12.0,-3)
C
C                 (Y,Z) PROJECTION
C
C         DRAW AXES USING SUBROUTINE AXIS
C
      CALL PLOT(0.0,1.5-YTMIN*TSF,-3)
      NTC=INT(2.*(ZTMAX-ZTMIN)+1.0)
      CALL AXIS(0.0,0.0,1H ,1,(ZTMAX-ZTMIN)*TSF,-NTC,1,0.0)
      CALL PLOT(-ZTMIN*TSF,YTMIN*TSF,-3)
      NTC=INT(2.0*YTSIZE+1.0)
      CALL AXIS(0.0,0.0,1H ,0,YTSIZE*TSF,-NTC,2,PION2)
      CALL PLOT(0.0,-YTMIN*TSF,-3)
C
C         LABEL AND ANNOTATE Z-AXIS (HORIZONTAL)
C
      CALL SCALF(TSF,TSF,1)
      YCH=-.15/TSF
      ZCH=AINT(ZTMIN)
      DZCH2=1.0
  100 CONTINUE
      CALL NUMPLT(ZCH,YCH,0.0,-0.1,ZCH,1)
      ZCH=ZCH+DZCH2
      IF (ZCH.LE.ZTMAX) GO TO 100
      ZCH=ZTMAX+.2/TSF
      YCH=-.05/TSF
      CALL CHAR(ZCH,YCH,0.0,.14,3HZ/R,3)
      ZCH=ZCH+.42/TSF
      CALL CHAR(ZCH,YCH,0.0,.06,6HPLANET,6)
C
C         LABEL AND ANNOTATE Y-AXIS (VERTICAL)
C
      YCH=AINT(YTMIN)
      DYCH2=1.0
      ZCH=.05/TSF
  110 CONTINUE
      IF (YCH.NE.0.0) CALL NUMPLT(ZCH,YCH-.05/TSF,0.0,0.1,YCH,1)
      YCH=YCH+DYCH2
      IF (YCH.LE.YTMAX) GO TO 110
      ZCH=-.2/TSF
      YCH=YTMAX+.1/TSF
      CALL CHAR(ZCH,YCH,0.0,.14,3HY/R,3)
      IF (RPLNT .LE. 0.0) GO TO 120
      ZCH=ZCH+.42/TSF
      CALL CHAR(ZCH,YCH,0.0,.06,6HPLANET,6)
C
C         DRAW PLANET AND LABEL PLOT
C
      CALL ELIPS(1.0,0.0,1.0,1.0,0.0,0.0,0.0,3)
  120 CONTINUE
      ZCH=0.5*(ZTMAX+ZTMIN)-1.0/TSF
      YCH=YTMIN-0.7/TSF
      CALL CHAR(ZCH,YCH,0.0,0.2,TITLE1,10)
      ZCH=ZCH+.04/TSF
      YCH=YCH-0.3/TSF
      CALL CHAR(ZCH,YCH,0.0,0.12,TITLYZ,16)
C
C         DRAW (Y,Z) PROJECTION
C
      ZTMAX=ZTMAX*RPLNT
      YTMIN=YTMIN*RPLNT
      CALL PLOT(0.0,0.0,3)
      CALL SCALF(TSFRO,TSFRO,1)
      CALL VECTOR(ZTRAJ,YTRAJ,NTRAJ,1,1,1HX)
      CALL MRKPLT(ZTRAJ,YTRAJ,TSF,TSF)
      CALL PLOT(ZTMAX,YTMIN,-3)
      CALL RESET
C
C                     (X,R) PROJECTION
C
C
C          DRAW, LABEL, AND ANNOTATE X-AXIS (HORIZONTAL)
C
      NTC=INT(2.0*(XTMAX-XTMIN)+1.0)
      YCH=0.0
      IF (YTSIZE-RTMAX.GE.1.0) YCH=0.5
      CALL PLOT(4.0,YCH,-3)
      CALL AXIS(0.0,0.0,1H ,-1,(XTMAX-XTMIN)*TSF,-NTC,1,0.0)
      CALL PLOT(-XTMIN*TSF,0.0,-3)
      CALL SCALF(TSF,TSF,1)
      RCH=-.15/TSF
      XCH=AINT(XTMIN)
      DXCH2=1.0
  200 CONTINUE
      CALL NUMPLT(XCH,RCH,0.0,-0.1,-XCH,1)
      XCH=XCH+DXCH2
      IF (XCH.LE.XTMAX) GO TO 200
      XCH=XTMIN-0.9/TSF
      RCH=-.05/TSF
      CALL CHAR(XCH,RCH,0.0,.14,3HX/R,3)
      XCH=XCH+.42/TSF
      CALL CHAR(XCH,RCH,0.0,.06,6HPLANET,6)
C
C         DRAW, LABEL, AND ANNOTATE R-AXIS (VERTICAL)
C
      NTC=INT(2.*RTMAX+1.0)
      CALL AXIS(0.0,0.0,1H ,0,RTMAX*TSF,-NTC,2,PION2)
      RCH=1.0
      DRCH2=1.0
      XCH=.05/TSF
  210 CONTINUE
      CALL NUMPLT(XCH,RCH-.05/TSF,0.0,0.1,RCH,1)
      RCH=RCH+DRCH2
      IF (RCH.LE.RTMAX) GO TO 210
      XCH=-0.8/TSF
      RCH=RTMAX+0.1/TSF
      CALL MATH(XCH,RCH,.24/TSF,0.0,21)
      XCH=XCH+0.16/TSF
      CALL PLOT(XCH+.55/TSF,RCH+.20/TSF,2)
      CALL CHAR(XCH,RCH,0.0,.14,1HY,1)
      CALL CHAR(XCH+.14/TSF,RCH+.11/TSF,0.0,.06,1H2,1)
      XCH=XCH+0.2/TSF
      CALL CHAR(XCH,RCH,0.0,.14,2H+Z,2)
      CALL CHAR(XCH+.28/TSF,RCH+.11/TSF,0.0,.06,1H2,1)
      XCH=XCH+.35/TSF
      CALL CHAR(XCH,RCH,0.0,.14,2H/R,2)
      IF (RPLNT .LE. 0.0) GO TO 215
      XCH=XCH+.25/TSF
      CALL CHAR(XCH,RCH,0.0,.06,6HPLANET,6)
C
C         DRAW PLANET AND IONOPAUSE, AND LABEL PLOT
C
      CALL ELIPS(1.0,0.0,1.0,1.0,0.0,0.0,180.0,3)
  215 CONTINUE
      XTMIN=XTMIN*RPLNT
      XTMAX=XTMAX*RPLNT
      RTMAX=RTMAX*RPLNT
      CALL PLOT(0.0,0.0,3)
      CALL SCALF(TSFRO,TSFRO,1)
      DO 220 I=1,NXMAX
      IF (XBOD(I).GT.XTMAX) GO TO 230
      IF (YBOD(I).GT.RTMAX) GO TO 230
      NBODY=I
  220 CONTINUE
  230 CONTINUE
      DO 240 I=1,NXMAX
      IF (XSHK(I).GT.XTMAX) GO TO 250
      IF (YSHK(I).GT.RTMAX) GO TO 250
      NSHOK=I
  240 CONTINUE
  250 CONTINUE
      CALL VECTOR(XBOD,YBOD,NBODY,1,0,1H )
      CALL VECTOR(XSHK,YSHK,NSHOK,1,0,1H )
      XCH=0.5*(XTMIN+XTMAX)-1.0/TSFRO
      RCH=-0.7/TSFRO
      CALL CHAR(XCH,RCH,0.0,0.2,TITLE1,10)
      XCH=XCH+0.04/TSFRO
      RCH=RCH-0.3/TSFRO
      CALL CHAR(XCH,RCH,0.0,0.12,TITLXR,16)
C
C         DRAW (X,R) PROJECTION
C
      CALL VECTOR(XTRAJ,RTRAJ,NTRAJ,1,1,1HX)
      CALL MRKPLT(XTRAJ,RTRAJ,TSF,TSF)
      CALL PLOT(XTMAX,0.0,-3)
      CALL RESET
      RETURN
      END
```

```
      FUNCTION QUAD (X,Y,F,XP,YP)
C
C     THIS FUNCTION PERFORMS A GENERALIZED QUADRILATERAL INTERPOLATION,
C     GIVEN THE COORDINATES OF THE VERTICES, THE VALUES OF THE FUNCTION
C     AT THOSE POINTS, AND THE COORDINATES OF THE POINT AT WHICH A
C     VALUE OF THE FUNCTION IS DESIRED, (XP,YP)
C    X AND Y ARE THE COORDINATE ARRAYS AND F IS THE FUNCTION ARRAY
C
      DIMENSION X(4),Y(4),F(4)
C
      S1=4.0*XP-(X(1)+X(2)+X(3)+X(4))
      S2=X(1)+X(2)-X(3)-X(4)
      S3=X(1)-X(2)+X(3)-X(4)
      S4=-X(1)+X(2)+X(3)-X(4)
      T1=4.0*YP-(Y(1)+Y(2)+Y(3)+Y(4))
      T2=Y(1)+Y(2)-Y(3)-Y(4)
      T3=Y(1)-Y(2)+Y(3)-Y(4)
      T4=-Y(1)+Y(2)+Y(3)-Y(4)
      A=S2*T3-S3*T2
      B=S1*T3+S2*T4-S3*T1-S4*T2
      C=S1*T4-S4*T1
      D=SQRT(B*B-4.0*A*C)
      ETA=-(B+D)/(A*2.0)
      IF (ABS(ETA) .GT. 1.0) ETA=ETA+D/A
      XI=(S1+ETA*S2)/(S4+ETA*S3)
      QUAD=0.25*(F(1)*(1.0-ETA)*(1.0-XI)+F(2)*(1.0-ETA)*(1.0+XI)
     *          +F(3)*(1.0+ETA)*(1.0+XI)+F(4)*(1.0+ETA)*(1.0-XI))
      RETURN
      END


      SUBROUTINE RERUN
      COMMON /CONT/ KVCON,VCON(20),KRCON,RCON(20)
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      COMMON /BLUNT/ THETA(25),RR(20,25),NBLUNT
      LEVEL 2, VX,VY
      COMMON/VCOMP/ VX(20,100),VY(20,100)
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      COMMON /ONSTRM/ ZPLOT,NZEND,NZADD,NXPLOT
      LOGICAL LGRAV
      COMMON/NUBOD/XX(100),YY(100),NBOD,LGRAV
      COMMON /BIN/ ANGP,ANGN,KBCON,BCON(20)
      LOGICAL LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      COMMON /PROPT/ LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      COMMON /SHOCKS/ DRSDX(100),DST(50)
      COMMON /MKPLOT/ NMARKT,MARKT(12)
      LOGICAL LPLTRJ
      COMMON /TROPT/ LPLTRJ,RPLNT,VINF,RHOINF,TMPINF,BINF
      COMMON /TRJDAT/ NTRAJ,TTRAJ(100),XTRAJ(100),YTRAJ(100),ZTRAJ(100),
     *  VTRAJ(100),VXTRAJ(100),VYTRAJ(100),VZTRAJ(100),TMPTRJ(100),
     *  RTRAJ(100),BXTRAJ(100),BYTRAJ(100),BZTRAJ(100),ROTRAJ(100),
     *  BTRAJ(100)
      LOGICAL LSUN
      COMMON /SUNDAT/ LSUN,XTRAJS(100),YTRAJS(100),ZTRAJS(100),
     *                RTRAJS(100),AZANG,POLANG,BX1,BY1,BZ1
C
      DIMENSION PR(100)
      EQUIVALENCE (PR(1),YY(1))
      DIMENSION NK(100)
      DATA NBLANK/1H /,NSTAR/1H*/
C
C     THIS SUBROUTINE READS DATA FROM TAPE4 (WRITTEN TO TAPE9 ON
C     A PREVIOUS RUN) TO ALLOW RESTARTING OF THE PROGRAM USING
C     DIFFERENT CONTOUR VALUES, PLOT SIZE, AND/OR MAGNETIC FIELD
C     ANGLES, WITHOUT REPEATING THE CALCULATION OF THE VELOCITY
C     AND DENSITY FIELDS.
C
      READ(4) NBLUNT,NRMAX,AMACH4,GAM4,HRO4
      GAM=GAMMA
      IF (ABS(AMACH-AMACH4) .GT. 1.0E-5) GO TO 100
      IF (ABS(GAM-GAM4) .GT. 1.0E-5) GO TO 100
      IF (HRO .LT. 0.0  .AND.  HRO4 .LT. 0.0) GO TO 2
      IF (ABS(HRO-HRO4) .GT. 1.0E-5) GO TO 100
      GO TO 3
    2 READ(4) NBOD,(XX(I),RR(I),I=1,NBOD)
    3 CONTINUE
      NXMAX=NBLUNT
C
C     PRINT INPUT DATA
C
      WRITE(6,213) AMACH,GAM
      IF (HRO) 4,6,8
    4 WRITE(6,214) NBOD,(XX(I),RR(I),I=1,NBOD)
      GO TO 9
    6 WRITE(6,216)
      GO TO 9
    8 IF (LGRAV) GO TO 11
      WRITE(6,218) HRO
      GO TO 9
   11 WRITE(6,219) HRO
    9 CONTINUE
      XPLOT=-ZPLOT
      IF (LSUN) GO TO 17
      WRITE (6,231) ANGP,ANGN
   17 CONTINUE
      WRITE(6,235) LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ
      IF (.NOT. LTRAJ) GO TO 25
      RORP=1.0/RPLNT
      WRITE(6,260) LPLTRJ,RORP,VINF,RHOINF,TMPINF
      IF (LSUN) WRITE (6,269) BINF,BX1,BY1,BZ1,AZANG,POLANG
      WRITE(6,263) NTRAJ,NMARKT
      DO 21 N=1,NTRAJ
   21 NK(N)=NBLANK
      IF (NMARKT .LE. 0) GO TO 24
      DO 22 I=1,NMARKT
      K=MARKT(I)
   22 NK(K)=NSTAR
   24 CONTINUE
      WRITE (6,267) (N,NK(N),TTRAJ(N),XTRAJS(N),YTRAJS(N),ZTRAJS(N),
     *                N=1,NTRAJ)
   25 CONTINUE
      WRITE(6,240)
      WRITE(6,242) KVCON
      IF (KVCON .GT. 0)  WRITE(6,250) (VCON(I),I=1,KVCON)
      WRITE(6,244) KRCON
      IF (KRCON .GT. 0)  WRITE(6,250) (RCON(I),I=1,KRCON)
      WRITE(6,246) KBCON
      DTOR=.01745329252
      ANGP=ANGP*DTOR
      ANGN=ANGN*DTOR
      IF (KBCON .GT. 0)  WRITE(6,250) (BCON(I),I=1,KBCON)
C
  213 FORMAT(//20X,25HINTERPLANETARY MACH NO. =,F5.2
     *        ///20X,21HSPECIFIC HEAT RATIO =,F6.3)
  214 FORMAT(//20X,46HOBSTACLE GEOMETRY: USER-SUPPLIED COORDINATES -,
     *        I4,7H POINTS//33X,3HX/D,11X,3HR/D/
     *        (30X,F8.4,6X,F8.4))
  216 FORMAT(//20X,49HOBSTACLE GEOMETRY: DEFAULT MAGNETOPAUSE COORDINAT,
     *        21HES - EQUATORIAL TRACE)
  218 FORMAT(//20X,48HOBSTACLE GEOMETRY: DEFAULT IONOPAUSE COORDINATES,
     *        /,39X,36HFOR CONSTANT SCALE HEIGHT WITH H/RO=,F4.2)
  219 FORMAT (//20X,75HOBSTACLE GEOMETRY: DEFAULT IONOPAUSE WITH GRAVITA
     *TIONAL VARIATION FOR H/RO=,F4.2)
  230 FORMAT(//20X,47HTERMINAL DOWNSTREAM LOCATION FOR PLOTTING,X/RL=,
     *        F6.2)
  231 FORMAT(//20X,50HUSER SPECIFIED DEVIATION IN SOLAR WIND COORDINATES
     *        /20X,47HOF MAGNETIC FIELD SYMMETRY-PLANE COMPONENT FROM,
     *        17H FLOW DIRECTION =,F7.2,8H DEGREES
     *        //20X,50HUSER SPECIFIED DEVIATION IN SOLAR WIND COORDINATES
     *        /20X,50HOF MAGNETIC FIELD FROM MAGNETIC SYMMETRY PLANE =,
     *        F7.2,9H DEGREES)
  235 FORMAT(//20X,8HLRERUN =,L2//20X,8HLPRFL  =,L2//20X,8HLPRST  =,L2,
     *        //20X,8HLPRCON =,L2//20X,8HLPRB   =,L2//20X,8HLPLOT  =,L2,
     *        //20X,8HLTRAJ  =,L2)
  240 FORMAT(1H1//45X,43HVALUES SPECIFIED FOR CONTOUR CALCULATION/
     *        45X,40(1H-))
  242 FORMAT(////20X,I2,29H CONTOUR LEVELS FOR VELOCITY:)
  244 FORMAT(////20X,I2,28H CONTOUR LEVELS FOR DENSITY:)
  246 FORMAT(////20X,I2,44H CONTOUR LEVELS FOR MAGNETIC FIELD STRENGTH:)
  250 FORMAT(/(10X,6F10.3))
  260 FORMAT(/20X,8HLPLTRJ =,L2/
     *  1H1//50X,32HINPUT FOR TRAJECTORY CALCULATION/
     *   50X,32(1H*)////
     *  /20X,34HRO/RPLANET                        =,F8.4/
     *  /20X,34HINTERPLANETARY VELOCITY           =1PE11.3/
     *  /20X,34HINTERPLANETARY DENSITY            =,E11.3/
     *  /20X,34HINTERPLANETARY TEMPERATURE        =,E11.3/
     *  /20X,34HINTERPLANETARY MAGNETIC FIELD     )
  263 FORMAT(////25X,7HNTRAJ =,I4,15X,8HNMARKT =,I3/
     *  30X,45H( *  = POINT TO BE MARKED FOR CROSS REFERENCE)///
     *  28X,1HN,9X,5HTTRAJ,8X,5HXTRAJ,7X,5HYTRAJ,7X,5HZTRAJ/)
  267 FORMAT((25X,I4,1X,A1,3X,F10.4,5X,F8.4,2(4X,F8.4)))
  268 FORMAT(45X,30H(SUN-PLANET COORDINATE SYSTEM))
  269 FORMAT(35X,14H  MAGNITUDE  =,E11.3
     *        /35X,14HX-COMPONENT  =,E11.3
```

```
     *        /35X,14HY-COMPONENT  =,E11.3
     *        /35X,14HZ-COMPONENT  =,E11.3
     *       //20X,29HAZIMUTHAL ANGLE                  =,E11.3/
     *        /20X,29HPOLAR ANGLE                      =,E11.3)
C
C
C     READ DATA GENERATED BY BLUNT BODY CODE
C
      READ(4) (J1,THETA(J),DRSDX(J),(I1,RR(I,J),I=1,NRMAX),J=2,NBLUNT)
      DO 10 J=2,NBLUNT
      READ(4) K,(I1,XC(I,J),YC(I,J),VX(I,J),VY(I,J),RHOF(I,J),I=1,NRMAX)
   10 CONTINUE
C
C     READ DATA GENERATED BY MARCHING CODE
C
      NZADD=0
      NZEND=NBLUNT
      NZI=NBLUNT+1
      DO 20 J=NZI,100
      READ(4) J1,Z,DRSDX(J)
      IF (INT(EOF(5)) .NE. 0) GO TO 30
      IF (XC(1,J-1) .GT. ZPLOT) GO TO 30
      NZADD=NZADD+1
      DO 20 I=1,NRMAX
      READ(4) I1,YC(I,J),VX(I,J),VY(I,J),RHOF(I,J)
      XC(I,J)=Z
   20 CONTINUE
   30 CONTINUE
      RETURN
C
C     PRINT ERROR MESSAGE IF TAPE4 IS NOT THE SAME CASE AS
C     SPECIFIED IN CARD INPUT  -  PROGRAM IS STOPPED
C
  100 WRITE(6,1000) AMACH,GAM,HRO,AMACH4,GAM4,HRO4
 1000 FORMAT(1H1,19X,5(1H*),2X,20HEXECUTION TERMINATED 2X,5(1H*)//20X,
     *        34HRERUN DATA ON TAPE4 DOES NOT AGREE/28X,10HWITH CASE ,
     *        24HSPECIFIED ON CARD INPUT://25X,8HMACH NO.,6X,5HGAMMA,9X,
     *        4HH/RC//12X,10HFROM CARDS,3(4X,F10.4)/12X,10HFROM TAPE4,
     *        3(4X,F10.4))
      STOP
      END


      SUBROUTINE ROTAT(IND)
      COMMON /BIN/ ANGP,ANGN,KBCON,BCON(20)
      COMMON /TRJDAT/ NTRAJ,TTRAJ(100),XTRAJ(100),YTRAJ(100),ZTRAJ(100),
     *  VTRAJ(100),VXTRAJ(100),VYTRAJ(100),VZTRAJ(100),TMPTRJ(100),
     *  BTRAJ(100),BXTRAJ(100),BYTRAJ(100),BZTRAJ(100),ROTRAJ(100),
     *  RTRAJ(100)
      LOGICAL LSUN
      COMMON /SUNDAT/ LSUN,XTRAJS(100),YTRAJS(100),ZTRAJS(100),
     *                  RTRAJS(100),AZANG,POLANG,BX1,BY1,BZ1
C
C
C
      IF (IND.LT.0) GO TO 100
C
C          TRANSFORM TRAJECTORY AND FREESTREAM B-FIELD COORDINATES
C          TO SOLAR WIND COORDINATE SYSTEM
C
      TEMPAZ=AZANG*.01745329
      TEMPOL=POLANG*.01745329
      SAZ=SIN(TEMPAZ)
      CAZ=COS(TEMPAZ)
      SPOL=SIN(TEMPOL)
      CPOL=COS(TEMPOL)
      DO 1  I=1,NTRAJ
      XTRAJS(I)=XTRAJ(I)
      YTRAJS(I)=YTRAJ(I)
      ZTRAJS(I)=ZTRAJ(I)
    1 CONTINUE
      DO 2  I=1,NTRAJ
      XTRAJ(I)=(XTRAJS(I)*CAZ-YTRAJS(I)*SAZ)*CPOL+ZTRAJS(I)*SPOL
      YTRAJ(I)=XTRAJS(I)*SAZ+YTRAJS(I)*CAZ
      ZTRAJ(I)=-(XTRAJS(I)*CAZ-YTRAJS(I)*SAZ)*SPOL+ZTRAJS(I)*CPOL
      XTRAJ(I)=-XTRAJ(I)
      YTRAJ(I)=-YTRAJ(I)
    2 CONTINUE
C
C          CALCULATE ANGN AND ANGP
C
      BX=(BX1*CAZ-BY1*SAZ)*CPOL+BZ1*SPOL
      BY=BX1*SAZ+BY1*CAZ
      BZ=-(BX1*CAZ-BY1*SAZ)*SPOL+BZ1*CPOL
      BX=-BX
      BY=-BY
      ANGP=ATAN2(BY,BX)
      ANGN=ATAN2(BZ,SQRT(BX*BX+BY*BY))
      RTOD=57.29578122
      ANGP=ANGP*RTOD
      ANGN=ANGN*RTOD
      RETURN
C
C
C
  100 CONTINUE
      RETURN
      END


      FUNCTION RRINF(I,J)
C
C     THIS ROUTINE CALCULATES R/RINF AT THE (I,J) GRID POINT
C
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      LEVEL 2, XST,YST,NUMST,NST
      COMMON /STREAM/ XST(50,152),YST(50,152),NUMST(50),NST
C
C     IF POINT IS ON SHOCK BOUNDARY R/RINF=1.0
C
      IF (I .LT. NRMAX  .AND.  J .GT. 1) GO TO 10
      RRINF=1.0
      RETURN
C
C     BRACKET POINT BY TWO STREAMLINES
C
   10 CONTINUE
      X=XC(I,J)
      Y=YC(I,J)
      R2=0.0
      DO 20 JST=1,NST
      NN=NUMST(JST)
      DO 30 K=1,NN
      IF (XST(JST,K) .GT. X) GO TO 40
   30 CONTINUE
      K=NN
   40 CONTINUE
      J2=JST
      J1=JST-1
      R1=R2
      IF (K .EQ. 1) R2=YST(J2,1)
      IF (K .GT. 1) R2=YST(J2,K-1)+(X-XST(J2,K-1))
     *                 *(YST(J2,K)-YST(J2,K-1))/(XST(J2,K)-XST(J2,K-1))
      IF (R2 .GT. Y) GO TO 60
   20 CONTINUE
   60 IF (J2 .EQ. 1) GO TO 100
C
C     INTERPOLATE FOR R/RINF
C
      RINF1=YST(J1,1)
   70 RINF2=YST(J2,1)
      RRINF=Y/(RINF1+(RINF2-RINF1)*(Y-R1)/(R2-R1))
      RETURN
C
C     USE SYMMETRY AXIS AND BODY FOR ZERO-TH STREAMLINE
C
  100 CONTINUE
      IF (X .GT. -1.0) GO TO 110
      RRINF=1.0+Y/R2*(R2/YST(1,1)-1.0)
      RETURN
  110 DO 120 K=2,NXMAX
      IF (X .LE. XC(1,K)) GO TO 130
  120 CONTINUE
  130 R1=YC(1,K-1)+(X-XC(1,K-1))*(YC(1,K)-YC(1,K-1))/(XC(1,K)-XC(1,K-1))
      RINF1=0.0
      GO TO 70
      END
```

```
      SUBROUTINE SERCH(J,K,KOD2,A,JDIM,ICHK,NKY,KOD3,NVAL,A1,A2,ACONT)
C
C        CONTOUR PROGRAMS MAP, WALK, SERCH, ENTER, AND CHECK
C        WRITTEN BY REESE SORENSON, NASA-AMES RES. CTR., AUG., 1974.
C          (MODIFIED VERSION)
C
C        THIS SUBROUTINE CHECKS WHETHER A CONTOUR LINE AT LEVEL NVAL
C        PASSES THROUGH AN INTERVAL HAVING INDICES J,K AT IT'S
C        LEFT(KOD2=2) OR BOTTOM(KOD2=1) POINT.
C
      DIMENSION A(1),ACONT(1),ICHK(4,1)
C
      IDEX(J,K)=J+(K-1)*JDIM
C
      GO TO (1,2),KOD2
1     I=IDEX(J,K+1)
      GO TO 3
2     I=IDEX(J+1,K)
3     A2=A(I)
      I=IDEX(J,K)
      A1=A(I)
C
      IF(A1-A2)4,5,5
4     AH=A2
      AL=A1
      GO TO 6
5     AH=A1
      AL=A2
C
6     VAL=ACONT(NVAL)
      IF(VAL.LT.AL)  GO TO 12
      IF(VAL.GT.AH)  GO TO 12
C
      CALL CHECK(ICHK,NKY,KOD2,J,K,NVAL,KOD9)
      IF(KOD9.EQ.2)  GO TO 12
C
      KOD3=1
      GO TO 20
12    KOD3=0
20    RETURN
      END


      SUBROUTINE SETUP(IPLOT,FMACH,GAM,IBODY)
C
C     THIS ROUTINE ESTABLISHES PLOT ORIGIN, DRAWS
C          AND LABELS AXES, AND WRITES TITLE.
C          UCC PLOT SUBROUTINES USED ARE
C               PLOT,AXIS,NUMPLT,CHAR,SCALE,
C               GREEK,MATH,PLTLN,DOTLN.
C
      DIMENSION TITLE3(2),TITLE4(2)
      DIMENSION TITLE5(2),TITLE6(3),TITLE7(2),TITL56(2)
      DIMENSION TITLEF(2)
      COMMON /SCALE/ XSF,YSF,XMAX,YMAX,XLNGTH,YLNGTH
      DATA PI2/1.57079633/
      DATA TITLE1,TITLE2/9HVELOCITY,,8HDENSITY,/
      DATA TITLE3(1),TITLE3(2)/10HSTREAMLINE,1HS/
      DATA TITLE4(1),TITLE4(2)/10HTEMPERATUR,1HE/
      DATA TITLE5(1),TITLE5(2)/10H(PARALLEL ,10HCOMPONENT)/
      DATA TITLE6(1),TITLE6(2),TITLE6(3)/10H(PERPENDIC,10HULAR COMPO,
     1      5HNENT)/
      DATA TITLE7(1),TITLE7(2)/10H(NORMAL CO,10HMPONENT)   /
      DATA TITL56(1),TITL56(2)/10HMAGNETIC F,5HIELD,/
      DATA TITLEC/9HCONTOURS/,TITLEF(1),TITLEF(2)/10HFIELD LINE,1HS/
C
C
C     SET ORIGIN AT LEFT END OF X-AXIS
C
      CALL PLOT(0.0,1.0,-3)
      XS=XSF
      YS=YSF
      CALL SCALE(XS,YS,1)
C
C     DRAW AND LABEL X-AXIS.  NOTE THAT SUBROUTINE AXIS
C          REQUIRES PARAMETERS IN INCHES, NOT USER UNITS.
C
      NTC=INT(2.0*(XMAX+1.5)+1.0)
      CALL AXIS(0.0,0.0,1H ,0,XLNGTH,-NTC,1,0.0)
C
C     SET PERMANENT ORIGIN AT X=0
      CALL PLOT(1.5,0.0,-3)
      XS=XSF
      YS=YSF
      CALL SCALE(XS,YS,1)
C
C     DRAW Y-AXIS
C
      NTC=INT(YMAX)+1
      CALL AXIS(0.0,0.0,1H ,0,YLNGTH,-NTC,2,PI2)
C
C     LABEL X-AXIS
C
      YCH=-.4/YSF
      XCH=0.5*XMAX-.75-.21/XSF
      IF (IBODY.EQ.1) GO TO 2
      CALL CHAR(XCH,YCH,0.0,.14,3HX/D,3)
      GO TO 3
    2 CALL CHAR(XCH,YCH,0.0,.14,3HX/R,3)
      XCH=XCH+.42/XSF
      CALL CHAR(XCH,YCH,0.0,.05,1HO,1)
    3 CONTINUE
C
C     ANNOTATE X-AXIS - NOTE ANNOTATION IS POSITIVE LEFT.
C
      YY=-.15/YSF
      IX=0
      XX=-1.5
    5 DELX=FLOAT(IX)*0.5
      XDX=XX+DELX
      CALL NUMPLT(XDX,YY,0.0,-0.1,-XDX,1)
      IX=IX+1
      IF(XDX.LT.XMAX) GO TO 5
C
C     ANNOTATE Y-AXIS AT SIDE EDGE OF PLOT
C
      YY=0.0
   10 YY=YY+1.0
      XCH=-1.5-.25/XSF
      CALL NUMPLT(XCH,YY-.05/YSF,0.0,-0.1,YY,1)
      XTIC=XCH+0.16/XSF
      CALL CHAR(XTIC,YY-.05/YSF,0.0,.09,1H-,1)
      IF(YY.LT.YMAX) GO TO 10
C
C     LABEL Y-AXIS.
C
      XCH=-1.5-0.85/XSF
      YCH=AINT((YMAX-1.0)*0.5)+0.5-.07/YSF
      IF (IBODY.EQ.1) GO TO 12
      CALL CHAR(XCH,YCH,0.0,.14,3HR/D,3)
      GO TO 13
   12 CALL CHAR(XCH,YCH,0.0,.14,3HR/R,3)
      XCH=XCH+.42/XSF
      CALL CHAR(XCH,YCH,0.0,.05,1HO,1)
   13 CONTINUE
C
C     DRAW TITLE.  IPLOT DETERMINES WHICH TITLE
C          SHOULD BE DRAWN.
C
      YCH=YMAX-0.8/YSF
      XCH=-1.5+0.45/XSF
      IF(IPLOT.EQ.2) GO TO 15
      IF(IPLOT.EQ.3) GO TO 20
      IF (IPLOT.EQ.4) GO TO 14
      IF (IPLOT.GT.4) GO TO 21
C
C     VELOCITY PLOT
C
      CALL CHAR(XCH,YCH,0.0,.20,TITLE1,9)
      XCH=XCH+2./XSF
      CALL PLTLN(XCH,YCH,XCH,YCH+.2/YSF)
      CALL CHAR(XCH+.05/XSF,YCH,0.0,.2,1HV,1)
      XCH=XCH+.25/XSF
      CALL PLTLN(XCH,YCH,XCH,YCH+.2/YSF)
      CALL CHAR(XCH+.05/XSF,YCH,0.0,.2,2H/V,2)
      CALL MATH(XCH+0.4/XSF,YCH-.05/YSF,.15/XSF,0.0,15)
      GO TO 25
C
C     TEMPERATURE PLOT
C
   14 CONTINUE
      CALL CHAR(XCH,YCH,0.0,.20,TITLE4,11)
      XCH=XCH+2.4/XSF
      CALL CHAR(XCH,YCH,0.0,.2,3HT/T,3)
      CALL MATH(XCH+.55/XSF,YCH-.05/YSF,.15/XSF,0.0,15)
```

```
      GO TO 25
C
C     DENSITY PLOT
C
   15 CONTINUE
      CALL CHAR(XCH,YCH,0.0,.2,TITLE2,9)
      XCH=XCH+2./XSF
      CALL GREEK(XCH,YCH,.2,0.0,17)
      CALL CHAR(XCH+.2/XSF,YCH,0.0,.2,1H/,1)
      CALL GREEK(XCH+.35/XSF,YCH,.2,0.0,17)
      CALL MATH(XCH+.45/XSF,YCH-.05/YSF,.15/XSF,0.0,15)
      GO TO 25
C
C     STREAMLINES PLOT
C
   2. CONTINUE
      CALL CHAR(XCH,YCH,0.0,.20,TITLE3,11)
      GO TO 25
C
C     MAGNETIC FIELD PLOTS
C
   2. CONTINUE
      CALL CHAR(XCH,YCH,0.0,0.2,TITL56,15)
      XCH=XCH+3.2/XSF
      CALL CHAR(XCH,YCH,0.0,0.2,4H(B/B,4)
      CALL MATH(XCH+.75/XSF,YCH-.05/YSF,.15/XSF,0.0,15)
      CALL CHAR(XCH+0.9/XSF,YCH,0.0,0.2,1H),1)
      IF (IPLOT .EQ. 6) GO TO 22
      IF (IPLOT.EQ.7) GO TO 24
      CALL CHAR(XCH+1.2/XSF,YCH-.05/YSF,1.5708,0.1,1H=,1)
      XCH=-1.5+0.4/XSF
      YCH=YMAX-1.1/YSF
      CALL CHAR(XCH,YCH,0.0,0.10,TITLE5,23)
      GO TO 23
   22 CALL CHAR(XCH+1.15/XSF,YCH+0..5/YSF,3.1416,0.1,1HT,1)
      XCH=-1.5+.4/XSF
      YCH=YMAX-1.1/YSF
      CALL CHAR(XCH,YCH,0.0,0.10,TITL56,25)
      GO TO 23
   24 CALL CHAR(XCH+1.15/XSF,YCH-0.05/YSF,3.0,0.1,1HN,1)
      XCH=-1.5+0.4/XSF
      YCH=YMAX-1.1/YSF
      CALL CHAR(XCH,YCH,0.0,0.10,TITLE7,23)
   23 YCH=YMAX-1.3/YSF
      XCH=-1.5+0.6/XSF
      CALL CHAR(XCH,YCH,0.0,0.10,TITLEC,8)
      CALL PLTLN(XCH+1.1/XSF,YCH,XCH+1.5/XSF,YCH)
      YCH=YMAX-1.5/YSF
      CALL CHAR(XCH,YCH,0.0,0.10,TITLEF,11)
      CALL DOTLN(XCH+1.15/XSF,YCH,XCH+1.5/XSF,YCH,20,0)
C
C     DRAW MACH NUMBER (M) AND RATIO OF SPECIFIC
C          HEATS (GAMMA).
C
   25 CONTINUE
      YCH=YMAX-.3/YSF
      XCH=-1.5+.4/XSF
      CALL CHAR(XCH,YCH,0.0,.2,2HM=,2)
      CALL NUMPLT(XCH+.4/XSF,YCH,0.0,.2,TMACH,2)
      YCH=YCH+1.4/XSF
      CALL GREEK(XCH,YCH,.2,0.0,3)
      CALL CHAR(XCH+.2/XSF,YCH,0.0,.2,1H=,1)
      CALL NUMPLT(XCH+.4/XSF,YCH,0.0,.2,GAM,2)
C
      RETURN
      END


      SUBROUTINE SFCALC(XSHK,YSHK,NTMAX)
C
C     THIS ROUTINE DETERMINES SCALE FACTORS AND SIZE OF PLOT.
C
      COMMON /SCALE/ XSF,YSF,XMAX,YMAX,XLNGTH,YLNGTH
      DIMENSION XSHK(1),YSHK(1)
      DATA YSIZE/8.0/
C
C     CALCULATE XMAX AND YMAX BASED ON LAST POINT OF SHOCK WAVE.
C
      XMAX=AINT(XSHK(NTMAX))+0.5
      IF(XMAX.LT.XSHK(NTMAX)) XMAX=XMAX+.5
      YMAX=AINT(YSHK(NTMAX))+1.0
      IF(YMAX.LT.3.0)YMAX=3.0
      IF(XMAX.LT.0.5) XMAX=0.5
C
C     FIX PLOT SIZE IN Y-DIRECTION
C     ADJUST SCALE FACTORS FOR EQUAL X AND Y SCALE FACTORS
C
      YLNGTH=YSIZE
      YSF=YLNGTH/YMAX
      XSF=YSF
      XLNGTH=XSF*(XMAX+1.5)
      RETURN
      END


      SUBROUTINE SFOFF(LF,FINF,PLTSZE,FSF,FOFFST,NA,NTICK,AMIN,ADEL,
     *        FDEL)
C
C    THIS SUBROUTINE CALCULATES THE PARAMETERS REQUIRED TO DRAW THE
C         VERTICAL AXIS OF THE TRAJECTORY DATA PLOTS
C
C        LF = 1  VELOCITY
C           = 2  TEMPERATURE
C           = 3  DENSITY
C           = 4  MAGNETIC FIELD
C
      COMMON /TRJDAT/ NTRAJ,TTRAJ(100),XTRAJ(100),YTRAJ(100),ZTRAJ(100),
     *  VTRAJ(100),VXTRAJ(100),VYTRAJ(100),VZTRAJ(100),TMPTRJ(100),
     *  BTRAJ(100),BXTRAJ(100),BYTRAJ(100),BZTRAJ(100),ROTRAJ(100),
     *  RTRAJ(100)
C
      GO TO (100,200,300,400),LF
C
C          VELOCITY
C
  100 FMIN=0.0
      DO 110 I=1,NTRAJ
      IF (VXTRAJ(I) .LT. FMIN) FMIN=VXTRAJ(I)
  110 CONTINUE
      DO 120 I=1,NTRAJ
      IF (VYTRAJ(I) .LT. FMIN) FMIN=VYTRAJ(I)
  120 CONTINUE
      DO 130 I=1,NTRAJ
      IF (VZTRAJ(I) .LT. FMIN) FMIN=VZTRAJ(I)
  130 CONTINUE
      FMIN=FMIN*FINF
      FMAX=FINF
      GO TO 700
C
C          TEMPERATURE
C
  200 FMIN=0.0
      FMAX=0.0
      DO 210 I=1,NTRAJ
      IF (TMPTRJ(I) .GT. FMAX) FMAX=TMPTRJ(I)
  210 CONTINUE
      FMAX=FMAX*FINF
      GO TO 700
C
C          DENSITY
C
  300 FMIN=0.0
      FMAX=0.0
      DO 310 I=1,NTRAJ
      IF (ROTRAJ(I) .GT. FMAX) FMAX=ROTRAJ(I)
  310 CONTINUE
      FMAX=FMAX*FINF
      GO TO 700
C
C          MAGNETIC FIELD STRENGTH
C
  400 FMIN=0.0
      FMAX=0.0
      DO 410 I=1,NTRAJ
      IF (BXTRAJ(I) .LT. FMIN) FMIN=BXTRAJ(I)
  410 CONTINUE
      DO 420 I=1,NTRAJ
      IF (BYTRAJ(I) .LT. FMIN) FMIN=BYTRAJ(I)
  420 CONTINUE
      DO 430 I=1,NTRAJ
      IF (BZTRAJ(I) .LT. FMIN) FMIN=BZTRAJ(I)
  430 CONTINUE
      DO 440 I=1,NTRAJ
```

```
      IF (BTRAJ(I) .GT. FMAX) FMAX=BTRAJ(I)
  440 CONTINUE
      FMAX=FMAX*FINF
      FMIN=FMIN*FINF
C
C   FIND ORDER OF MAGNITUDE
C
  700 CONTINUE
      FDIFF=FMAX-FMIN
      FLOG=ALOG10(FDIFF)
      NLOG=INT(FLOG)
      IF (FLOG .LT. 0.0) NLOG=NLOG-1
C
C   FIND INCREMENT AS POWER OF 10 TIMES .1, .2, OR .5
C
      ADIFF=FDIFF*10.**(-NLOG)
      IF (ADIFF .GT. 2.0) GO TO 720
      ADEL=0.2
      NA=NLOG
      GO TO 740
  720 CONTINUE
      IF (ADIFF .GT. 5.0) GO TO 730
      ADEL=0.5
      NA=NLOG
      GO TO 740
  730 CONTINUE
      ADEL=0.1
      NA=NLOG+1
  740 A10N=10.**NA
      IF (NA .NE. 1) GO TO 750
      ADEL=ADEL*10.
      NA=0
      A10N=1.0
  750 CONTINUE
C
C   CALCULATE SCALE AND OFFSET
C
      BMIN=0.0
      IF (FMIN .LT. 0.0) BMIN=1.0-1.0E-6
      MMIN=INT(FMIN/(ADEL*A10N)-BMIN)
      BMAX=1.0-1.0E-6
      MMAX=INT(FMAX/(ADEL*A10N)+BMAX)
      NTICK=MMAX-MMIN+1
      AMIN=FLOAT(MMIN)*ADEL
      AMAX=FLOAT(MMAX)*ADEL
      FOFFST=AMIN*A10N/FINF
      FSF=PLTSZE/((AMAX-AMIN)*A10N)
      FDEL=ADEL*A10N*FSF
      FSF=FSF*FINF
      RETURN
      END
```

```
      SUBROUTINE STOUT
C
C     THIS SUBROUTINE PRINTS THE STREAMLINES CALCULATED
C          IN SUBROUTINE FLOWST
C
      COMMON /PLUNT/ THETA(25),RP(20,25),NPLUNT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NKMAX,AMACH,GAMMA,HRO,NMINDX
      LEVEL 2, XST,YST,NUMST,NST
      COMMON /STREAM/ XST(50,152),YST(50,152),NUMST(50),NST
C
C     REVERSE SIGN OF XST FOR OUTPUT
C
      DO 2 K=1,NST
      NN=NUMST(K)
      DO 2 J=1,NN
    2 XST(K,J)=-XST(K,J)
C
C     STREAMLINES FOR MAGNETOSPHERE
C
      WRITE(6,600) NST
      IF(NMINDX.EQ.1) GO TO 1
      DO 10 K=1,NST
      WRITE(6,610) K,XST(K,1),YST(K,1)
      IF(XST(K,1).GE.0.0) WRITE(6,620) THETA(K+1),RP(NPMAX,K+1)
      NN=NUMST(K)
      WRITE(6,630) (XST(K,J),YST(K,J),J=1,NN)
   10 CONTINUE
```

```
      GO TO 4
C
C     STREAMLINES FOR IONOPAUSE SHAPE
C
    1 DO 11 K=1,NST
      WRITE(6,611) K,XST(K,1),YST(K,1)
      IF (XST(K,1) .GE. 0.0) WRITE(6,621) THETA(K+1),RP(NPMAX,K+1)
      NN=NUMST(K)
      WRITE(6,631) (XST(K,J),YST(K,J),J=1,NN)
   11 CONTINUE
C
C     RESTORE SIGN OF XST
C
    4 CONTINUE
      DO 3 K=1,NST
      NN=NUMST(K)
      DO 3 J=1,NN
    3 XST(K,J)=-XST(K,J)
      RETURN
C
  600 FORMAT(1H1//49X,33HSTREAMLINE TRAJECTORY CALCULATION/49X,33(1H*)/
     *  //10X,I3,23H STREAMLINES CALCULATED)
  610 FORMAT(/////15H STREAMLINE NO.,I2,19H, STARTING AT X/D =,F8.4,
     *  8H,  R/D =,F8.4)
  611 FORMAT(/////15H STREAMLINE NO.,I2,20H, STARTING AT X/R0 =,F8.4,
     *  9H,  R/R0 =,F8.4)
  620 FORMAT(13X,23H(CORRESPONDS TO THETA =,F6.2,9H DEGREES,,2X,
     *        6HRP/D =,F8.4,1H))
  621 FORMAT(13X,23H(CORRESPONDS TO THETA =,F6.2,9H DEGREES,,2X,
     *        7HRP/R0 =,F9.4,1H))
  630 FORMAT(/15X,3HX/D,12X,3HR/D/(9X,F10.4,5X,F10.4))
  631 FORMAT(/15X,4HX/R0,11X,4HR/R0/(9X,F10.4,5X,F10.4))
      END
```

```
      SUBROUTINE TAXIS(TIMESF,TOFFST,PLTSZE,NAXIS)
C
C        SUBROUTINE TAXIS CALCULATES THE SCALE FACTOR AND OFFSET FOR
C        THE TIME AXIS, AND PARAMETERS REQUIRED TO DRAW THE AXIS.
C       WHEN NAXIS=1,2  THE TIME AXIS IS PLOTTED
C
      COMMON /TRJDAT/ NTRAJ,TTRAJ(100),XTRAJ(100),YTRAJ(100),ZTRAJ(100),
     *  VTRAJ(100),VXTRAJ(100),VYTRAJ(100),VZTRAJ(100),TMPTRJ(100),
     *  BTRAJ(100),BXTRAJ(100),BYTRAJ(100),BZTRAJ(100),ROTRAJ(100),
     *  PTRAJ(100)
C
C        FIND THE RANGE AND INCREMENT REQUIRED
C        INCREMENT IS POWER OF 10 TIMES .1,.2, OR .5
C
      IF (NAXIS .GT. 0) GO TO 150
      NAXIS=1
      TMIN=TTRAJ(1)
      TMAX=TTRAJ(NTRAJ)
      TDIFF=TMAX-TMIN
      TLOG=ALOG10(TDIFF)
      NTLOG=INT(TLOG)
      IF (TLOG.LT.0.0) NTLOG=NTLOG-1
      ADIFF=TDIFF*10.**(-NTLOG)
      IF (ADIFF.GT.2.0) GO TO 20
      ADEL=2.0
      NA=NTLOG-1
      GO TO 60
   20 CONTINUE
      IF (ADIFF.GT.5.0) GO TO 40
      ADEL=5.0
      NA=NTLOG-1
      GO TO 60
   40 CONTINUE
      ADEL=1.0
      NA=NTLOG
   60 CONTINUE
      NA=NA+1
      ADEL=ADEL*0.1
      A10N=10.**NA
      IF (NA.LT.0 .OR. NA.GT.1) GO TO 100
      ADEL=ADEL*A10N
      A10N=1.0
      NAXIS=2
C
C        CALCULATE SCALE AND OFFSET
C
```

```
  100 CONTINUE
      BMIN=0.0
      BMAX=0.0
      IF (TMIN.LT.0.0) BMIN=1.0-1.0E-6
      IF (TMAX.GT.0.0) BMAX=1.0-1.0E-6
      AFIRST=AINT(TMIN/(ADEL*AION)-BMIN)*ADEL
      ALAST=AINT(TMAX/(ADEL*AION)+BMAX)*ADEL
      NTICK=INT((ALAST-AFIRST)/ADEL+1.0E-6)+1
      TOFFST=AFIRST*AION
      TIMESF=PLTSZE/((ALAST-AFIRST)*AION)
      TDEL=ADEL*AION*TIMESF
      RETURN
C
C         DRAW AXIS ONE INCH UP FROM PLOT BOUNDARY
C
  150 CONTINUE
      CALL PLOT(4.0,-12.0,-3)
      CALL PLOT(0.0,1.0,-3)
      CALL AXIS(0.0,0.0,1H ,-1,PLTSZE,-NTICK,1.0,0.0)
C
C         ANNOTATE AND LABEL AXIS
C
      YCH=-.15
      TCH=0.0
      ACH=AFIRST
      DO 200 N=1,NTICK
      CALL NUMPLT(TCH,YCH,[illegible],ACH,1)
      TCH=TCH+TDEL
      ACH=ACH+ADEL
  200 CONTINUE
      YCH=-.45
      IF (NAXIS.EQ.2) GO TO 250
      TCH=0.5*(PLTSZE-1.4)
      CALL CHAR(TCH,YCH,0.0,.14,9HTIME    1,0,9)
      CALL CHAR(TCH+.91,YCH,0.0,.07,1H-,1)
      YNA=FLOAT(NA)
      CALL NUMPLT(TCH+1.26,YCH+.11,0.0,.06,YNA,-1)
      RETURN
  250 CONTINUE
      TCH=0.5*(PLTSZE-.56)
      CALL CHAR(TCH,YCH,0.0,.14,4HTIME,4)
      RETURN
      END



      SUBROUTINE TRAJEC
C
C         THIS SUBROUTINE CALCULATES THE VALUES OF THE FLOW FIELD
C         AND THE MAGNETIC FIELD COMPONENTS ALONG A SPECIFIED
C         TRAJECTORY
C
      LEVEL 2, BPARA,BPERP,BNORM,BMAG,BANG
      COMMON /BCOMPS/ BPARA(20,100),BPERP(20,100),BNORM(20,100),
     *  BMAG(20,100),BANG(20,100)
      COMMON /BIN/ ANGP,ANGN,KBCON,BCON(20)
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NXMAX,NYMAX,AMACH,GAMMA,HRD,NHINDX
      LEVEL 2, ANG,DXTH,DEG
      COMMON /DRD/ ANG(20,100),DYTH(100),DEG
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      COMMON /TRJDAT/ NTRAJ,TTRAJ(100),XTRAJ(100),YTRAJ(100),ZTRAJ(100),
     *  VTRAJ(100),VXTRAJ(100),VYTRAJ(100),VZTRAJ(100),TMPTRJ(100),
     *  BTRAJ(100),BXTRAJ(100),BYTRAJ(100),BZTRAJ(100),RDTRAJ(100),
     *  RTRAJ(100)
      LOGICAL LPLTRJ
      COMMON /TROPT/ LPLTRJ,RPLNT,VINF,RHOINF,TMPINF,BINF
C
      FTRAJ(I,J,VAL)=(QI*VAL(I,J)+(1.0-QI)*VAL(I+1,J))*QJ
     *               +(QI*VAL(I,J+1)+(1.0-QI)*VAL(I+1,J+1))*(1.0-QJ)
C
      CANGN=COS(ANGN)
      SANGN=SIN(ANGN)
      CANGP=COS(ANGP)
      SANGP=SIN(ANGP)
C
C         CONVERT X,Y,Z FOR TRAJECTORY FROM UNITS OF PLANET RADII
C                   TO UNITS OF IONOPAUSE NOSE RADII
C
      IF (RPLNT .EQ. 0.0) GO TO 20
      DO 10 N=1,NTRAJ
      XTRAJ(N)=XTRAJ(N)*RPLNT
      YTRAJ(N)=YTRAJ(N)*RPLNT
      ZTRAJ(N)=ZTRAJ(N)*RPLNT
      RTRAJ(N)=SQRT(ZTRAJ(N)**2+YTRAJ(N)**2)
   10 CONTINUE
   20 CONTINUE
C
C      ROTATE ABOUT X-AXIS UNTIL Z=0.0
C
      DO 100 N=1,NTRAJ
      THET=ATAN2(ZTRAJ(N),YTRAJ(N))
      CTHET=COS(THET)
      STHET=SIN(THET)
      XP=XTRAJ(N)
      YP=YTRAJ(N)*CTHET+ZTRAJ(N)*STHET
      ZP=0.0
C
C         INTERPOLATE FOR FLOW FIELD
C
      CALL IJTRAJ(XP,YP,I,J,QI,QJ,MFLAG)
      IF (MFLAG) 80,40,90
   40 CONTINUE
      PHI=FTRAJ(I,J,ANG)
      VTRAJ(N)=FTRAJ(I,J,VF)
      RDTRAJ(N)=FTRAJ(I,J,RHOF)
      TMPTRJ(N)=1.0+((GAMMA-1.0)*0.5*AMACH**2)*(1.0-VTRAJ(N)**2)
      CPHI=COS(PHI)
      SPHI=SIN(PHI)
      VXTRAJ(N)=VTRAJ(N)*CPHI
      VYTRAJ(N)=VTRAJ(N)*SPHI*CTHET
      VZTRAJ(N)=VTRAJ(N)*SPHI*STHET
C
C         INTERPOLATE FOR MAGNETIC FIELD
C
      BPARAT=FTRAJ(I,J,BPARA)
      IF (I .EQ. 1) I=2
      BPERPT=FTRAJ(I,J,BPERP)
      BNORMT=FTRAJ(I,J,BNORM)
      PSI=FTRAJ(I,J,BANG)
      SPSI=SIN(PSI)
      CPSI=COS(PSI)
      BXINFP=CANGN*CANGP
      BYINFP=CANGN*SANGP*CTHET+SANGN*STHET
      BZINFP=SANGN*CTHET-CANGN*SANGP*STHET
      ANGPP=ATAN2(BYINFP,BXINFP)
      ANGNP=ATAN2(BZINFP,SQRT(BXINFP**2+BYINFP**2))
      CANGPP=COS(ANGPP)
      SANGPP=SIN(ANGPP)
      CANGNP=COS(ANGNP)
      SANGNP=SIN(ANGNP)
      BXP=CANGNP*(CPHI*CANGPP*BPARAT+CPSI*SANGPP*BPERPT)
      BYP=CANGNP*(SPHI*CANGPP*BPARAT+SPSI*SANGPP*BPERPT)
      BZP=SANGNP*BNORMT
      BXTRAJ(N)=BXP
      BYTRAJ(N)=BYP*CTHET-BZP*STHET
      BZTRAJ(N)=BZP*CTHET+BYP*STHET
      BTRAJ(N)=SQRT(BXP**2+BYP**2+BZP**2)
      GO TO 100
C
C         POINT IS INSIDE IONOPAUSE
C
   80 CONTINUE
      VTRAJ(N)=0.0
      RDTRAJ(N)=0.0
      TMPTRJ(N)=0.0
      VXTRAJ(N)=0.0
      VYTRAJ(N)=0.0
      VZTRAJ(N)=0.0
      BTRAJ(N)=0.0
      BXTRAJ(N)=0.0
      BYTRAJ(N)=0.0
      BZTRAJ(N)=0.0
      GO TO 100
C
C         POINT IS BEYOND BOW SHOCK
C
   90 CONTINUE
      VTRAJ(N)=1.0
      RDTRAJ(N)=1.0
      TMPTRJ(N)=1.0
      VXTRAJ(N)=1.0
      VYTRAJ(N)=0.0
      VZTRAJ(N)=0.0
      BTRAJ(N)=1.0
      BXTRAJ(N)=CANGN*CANGP
      BYTRAJ(N)=CANGN*SANGP
```

```
      BZTRAJ(N)=SANGN
  100 CONTINUE
C
C        OUTPUT TRAJECTORY INFORMATION, AND CREATE PLOTS IF REQUIRED
C
      CALL TROUT
      IF (.NOT.LPLTRJ) GO TO 200
      CALL PLOTTP
      CALL PLOTFS
  200 CONTINUE
      RETURN
      END


      SUBROUTINE TROUT
C
C         THIS SUBROUTINE PRINTS THE TRAJECTORY COORDINATES AND THE
C        FLOW FIELD AND MAGNETIC FIELD COMPONENTS ALONG THE TRAJECTORY
C        AS TABLES WITH TIME AS THE REFERENCE QUANTITY
C
      COMMON /BIN/ ANGP,ANGN,KBCON,BCON(20)
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NBMAX,NKMAX,AMACH,GAMMA,HRD,NHINDX
      LOGICAL LPERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      COMMON /PROPT/ LRERUN,LPRFL,LPRST,LPRCON,LPRB,LPLOT,LTRAJ,LRSTRT
      COMMON /TRJDAT/ NTRAJ,TTRAJ(100),XTRAJ(100),YTRAJ(100),ZTRAJ(100),
     *  VTRAJ(100),VXTRAJ(100),VYTRAJ(100),VZTRAJ(100),TMPTRJ(100),
     *  BTRAJ(100),BXTRAJ(100),BYTRAJ(100),BZTRAJ(100),ROTRAJ(100),
     *  RTRAJ(100)
      LOGICAL LPLTRJ
      COMMON /TROPT/ LPLTRJ,RPLNT,VINF,RHOINF,TMPINF,BINF
      LOGICAL LSUN
      COMMON /SUNDAT/ LSUN,XTRAJS(100),YTRAJS(100),ZTRAJS(100),
     *                RTRAJS(100),AZANG,POLANG,BX1,BY1,BZ1
      DIMENSION VXTRAJS(100),VYTRAJS(100),VZTRAJS(100),BXTRAJS(100),
     *          BYTRAJS(100),BZTRAJS(100)
C
C        WRITE COORDINATES (X,Y,Z,R) AS A FUNCTION OF TIME,
C          IN UNITS OF BOTH RO AND RPLANET
C
      RNOSE=1.0
      IF (RPLNT .GT. 0.0) RNOSE=1.0/RPLNT
      WRITE(6,1000)
      WRITE (6,1001)
      WRITE(6,1100) RNOSE
      WRITE(6,1200)
      DO 20 N=1,NTRAJ
      XRPLNT=XTRAJ(N)*RNOSE
      YRPLNT=YTRAJ(N)*RNOSE
      ZRPLNT=ZTRAJ(N)*RNOSE
      RRPLNT=RTRAJ(N)*RNOSE
      XT=XTRAJ(N)
      WRITE(6,2000) N,TTRAJ(N),XT,YTRAJ(N),ZTRAJ(N),RTRAJ(N),XRPLNT,
     *  YRPLNT,ZRPLNT,RRPLNT
   20 CONTINUE
C
C        WRITE OUT FLOW FIELD AND MAGNETIC FIELD COMPONENTS
C          NON-DIMENSIONAL WITH RESPECT TO FREE STREAM
C
      WRITE(6,1300)
      WRITE (6,1001)
      WRITE(6,1400)
      WRITE(6,1500)
      DO 40 N=1,NTRAJ
      WRITE(6,2100) N,TTRAJ(N),VTRAJ(N),VXTRAJ(N),VYTRAJ(N),VZTRAJ(N),
     *  ROTRAJ(N),TMPTRJ(N),BTRAJ(N),BXTRAJ(N),BYTRAJ(N),BZTRAJ(N)
   40 CONTINUE
C
C        WRITE OUT FLOW FIELD AND MAGNETIC FIELD COMPONENTS
C        DIMENSIONALIZE BY INPUT FREE STREAM VALUES
C
      SANGN=SIN(ANGN)
      CANGN=COS(ANGN)
      SANGP=SIN(ANGP)
      CANGP=COS(ANGP)
      XO=RINF*CANGN*CANGP
      YO=RINF*CANGN*SANGP
      ZO=RINF*SANGN
      WRITE(6,1300)
      WRITE (6,1001)
      WRITE(6,1600)
      WRITE(6,1700) AMACH,GAMMA,RINF,VINF,XO,RHOINF,YO,TMPINF,ZO
      WRITE(6,1800)
      DO 60 N=1,NTRAJ
      VDIM=VTRAJ(N)*VINF
      VXDIM=VXTRAJ(N)*VINF
      VYDIM=VYTRAJ(N)*VINF
      VZDIM=VZTRAJ(N)*VINF
      RHODIM=ROTRAJ(N)*RHOINF
      TMPDIM=TMPTRJ(N)*TMPINF
      BDIM=BTRAJ(N)*BINF
      BXDIM=BXTRAJ(N)*BINF
      BYDIM=BYTRAJ(N)*BINF
      BZDIM=BZTRAJ(N)*BINF
      WRITE(6,2200) N,TTRAJ(N),VDIM,VXDIM,VYDIM,VZDIM,RHODIM,TMPDIM,
     *  BDIM,BXDIM,BYDIM,BZDIM
   60 CONTINUE
      IF (LSUN) GO TO 70
      RETURN
C
C          TRANSFORM OUTPUT QUANTITIES TO SUN-PLANET COORDINATE SYSTEM
C
   70 CONTINUE
      DO 75  N=1,NTRAJ
      XTRAJS(N)=XTRAJS(N)*RPLNT
      YTRAJS(N)=YTRAJS(N)*RPLNT
      ZTRAJS(N)=ZTRAJS(N)*RPLNT
   75 CONTINUE
      TEMPAZ=AZANG*.01745329
      TEMPOL=POLANG*.01745329
      SAZ=SIN(TEMPAZ)
      CAZ=COS(TEMPAZ)
      SPOL=SIN(TEMPOL)
      CPOL=COS(TEMPOL)
      DO 80  N=1,NTRAJ
      RTRAJS(N)=SQRT(YTRAJS(N)*YTRAJS(N)+ZTRAJS(N)*ZTRAJS(N))
      VXTRAJS(N)=-CAZ*CPOL*VXTRAJ(N)-SAZ*VYTRAJ(N)-CAZ*SPOL*VZTRAJ(N)
      VYTRAJS(N)=SAZ*CPOL*VXTRAJ(N)-CAZ*VYTRAJ(N)+SAZ*SPOL*VZTRAJ(N)
      VZTRAJS(N)=-SPOL*VXTRAJ(N)+CPOL*VZTRAJ(N)
      BXTRAJS(N)=-CAZ*CPOL*BXTRAJ(N)-SAZ*BYTRAJ(N)-CAZ*SPOL*BZTRAJ(N)
      BYTRAJS(N)=SAZ*CPOL*BXTRAJ(N)-CAZ*BYTRAJ(N)+SAZ*SPOL*BZTRAJ(N)
      BZTRAJS(N)=-SPOL*BXTRAJ(N)+CPOL*BZTRAJ(N)
   80 CONTINUE
      XOS=-CAZ*CPOL*XO-SAZ*YO-CAZ*SPOL*ZO
      YOS=SAZ*CPOL*XO-CAZ*YO+SAZ*SPOL*ZO
      ZOS=-SPOL*XO+CPOL*ZO
C
C          WRITE COORDINATES (X,Y,Z,R) AS A FUNCTION OF TIME
C          IN UNITS OF BOTH RO AND RPLANET
C
      WRITE (6,1000)
      WRITE (6,1002)
      WRITE (6,1100) RNOSE
      WRITE (6,1200)
      DO 90  N=1,NTRAJ
      XRPLTS=XTRAJS(N)*RNOSE
      YRPLTS=YTRAJS(N)*RNOSE
      ZRPLTS=ZTRAJS(N)*RNOSE
      RRPLTS=RTRAJS(N)*RNOSE
      XTS=XTRAJS(N)
      WRITE (6,2000) N,TTRAJ(N),XTS,YTRAJS(N),ZTRAJS(N),RTRAJS(N),XRPLTS
     *,                YRPLTS,ZRPLTS,RRPLTS
   90 CONTINUE
C
C          WRITE FLOW FIELD AND MAGNETIC FIELD COMPONENTS
C          NON-DIMENSIONALIZED WITH RESPECT TO FREESTREAM
C
      WRITE (6,1300)
      WRITE (6,1002)
      WRITE (6,1400)
      WRITE (6,1500)
      DO 95  N=1,NTRAJ
      WRITE (6,2100) N,TTRAJ(N),VTRAJ(N),VXTRAJS(N),VYTRAJS(N),
     *                VZTRAJS(N),ROTRAJ(N),TMPTRJ(N),BTRAJ(N),BXTRAJS(N),
     *                BYTRAJS(N),BZTRAJS(N)
   95 CONTINUE
C
C          WRITE FLOW FIELD AND MAGNETIC FIELD COMPONENTS
C          DIMENSIONALIZED BY INPUT FREESTREAM VALUES
C
      WRITE (6,1300)
      WRITE (6,1002)
      WRITE (6,1600)
      WRITE (6,1700) AMACH,GAMMA,RINF,VINF,XOS,RHOINF,YOS,TMPINF,ZOS
      WRITE (6,1701) AZANG,POLANG
```

```
      WRITE (6,1000)
      DO 100  N=1,NTRAJ
      VDIM=VTRAJ(N)*VINF
      VXDIM=VXTRAJS(N)*VINF
      VYDIM=VYTRAJS(N)*VINF
      VZDIM=VZTRAJS(N)*VINF
      RHODIM=RDTRAJ(N)*RHOINF
      TMPDIM=TMPTRJ(N)*TMPINF
      BDIM=BTRAJ(N)*BINF
      BXDIM=BXTRAJS(N)*BINF
      BYDIM=BYTRAJS(N)*BINF
      BZDIM=BZTRAJS(N)*BINF
      WRITE (6,2200) N,TTRAJ(N),VDIM,VXDIM,VYDIM,VZDIM,RHODIM,TMPDIM,
     *               BDIM,BXDIM,BYDIM,BZDIM
  100 CONTINUE
      IF (.NOT. LPLTRJ) GO TO 200
C
C        TRANSFORM COMPONENT DATA TO SUN-PLANET COORDINATES
C
      DO 150  N=1,NTRAJ
      VXTRAJ(N)=VXTRAJS(N)
      VYTRAJ(N)=VYTRAJS(N)
      VZTRAJ(N)=VZTRAJS(N)
      BXTRAJ(N)=BXTRAJS(N)
      BYTRAJ(N)=BYTRAJS(N)
      BZTRAJ(N)=BZTRAJS(N)
  150 CONTINUE
  200 CONTINUE
      RETURN
C
 1000 FORMAT(1H1,52X,22HTRAJECTORY CALCULATION/53X,22(1H*))
 1001 FORMAT (53X,30H(SOLAR WIND COORDINATE SYSTEM))
 1002 FORMAT (53X,30H(SUN-PLANET COORDINATE SYSTEM))
 1100 FORMAT(///53X,22HTRAJECTORY COORDINATES/53X,22(1H-)
     *  //20X,17HRO/RPLANET =,F8.4)
 1200 FORMAT(//4X,1HN,7X,4HTIME,9X,4HX/R0,7X,4HY/R0,7X,4HZ/R0,7X,4HR/R0,
     *  8X,9HX/RPLANET,2X,9HY/RPLANET,2X,9HZ/RPLANET,2X,9HR/RPLANET/)
 1300 FORMAT(////36X,46HFLOW FIELD AND MAGNETIC FIELD COMPONENTS ALONG,
     *  11H TRAJECTORY/36X,57(1H-))
 1400 FORMAT(/42X,46H(NON-DIMENSIONALIZED BY INTERPLANETARY VALUES))
 1500 FORMAT(//4X,1HN,5X,4HTIME,7X,6HV/VINF,5X,7HVX/VINF,4X,7HVY/VINF,
     *  4X,7HVZ/VINF,2X,10HRHO/RHOINF,1X,11HTEMP/TMPINF,2X,6HB/BINF,5X,
     *  7HBX/BINF,4X,7HBY/BINF,4X,7HBZ/BINF/)
 1500 FORMAT(/41X,48H(DIMENSIONAL, USING INPUT INTERPLANETARY VALUES))
 1700 FORMAT(/21X,30HINTERPLANETARY MACH NUMBER    =,F7.2,8X,
     *        29HINTERPLANETARY MAGNETIC FIELD/2.X,
     *        30HRATIO OF SPECIFIC HEATS       =,F8.4,9X,13HMAGNITUDE   =,
     *        E10.3/20X,30HINTERPLANETARY VELOCITY      =,1PE10.3,7X,
     *        13HX-COMPONENT =,E10.3/20X,
     *        30HINTERPLANETARY DENSITY       =,E10.3,7X,
     *        13HY-COMPONENT =,E10.3/20X,
     *        30HINTERPLANETARY TEMPERATURE   =,E10.3,7X,
     *        13HZ-COMPONENT =,E10.3)
 1700 FORMAT(20X,30HAZIMUTHAL ANGLE               =,F1 .3
     *        /20X,30HPOLAR ANGLE                   =,F10.3)
 1800 FORMAT(//4X,1HN,5X,4HTIME,8X,3HV/V,9X,2HVX,9X,2HVY,9X,2HVZ,9X,
     *  3HRHO,7X,4HTEMP,8X,3HB/B,9X,2HBX,9X,2HBY,9X,2HBZ/)
 2000 FORMAT(1X,I4,3X,F10.4,2(5X,F8.4,3(3X,F9.4)))
 2100 FORMAT(1X,I4,1X,F10.4,10(3X,F9.4))
 2200 FORMAT(1X,I4,1X,F10.4,1X,1P10E11.3)
      END



      SUBROUTINE VAXIS(VSF,VOFFST,VINF,PLTSIZE,LV)
C
C   SUBROUTINE VAXIS CALCULATES THE PARAMETERS REQUIRED FOR THE
C        VERTICAL AXIS FOR THE TRAJECTORY PLOTS, THEN DRAWS AND
C        LABELS THE AXIS.  LV INDICATES FOR WHICH COMPONENT THE
C        AXIS IS REQUIRED.  IT IS ASSUMED THAT THE X,Y,Z COMPONENTS
C        IMMEDIATELY FOLLOW THE MAGNITUDE PLOTS, AND THE PARAMETERS
C        ARE NOT RE-CALCULATED.
C
      DATA PION2/1.5707963/
C
C   CALCULATE PARAMETERS, THEN DRAW AXIS WITH TICK MARKS
C
      LF=LV/10
      LCOMP=LV-10*LF+1
      IF (LCOMP .GT. 1) GO TO 10
      CALL SEOFF(LF,VINF,PLTSZE,VSF,VOFFST,NA,NTICK,AMIN,ADEL,VDEL)
   10 CONTINUE
      CALL AXIS(0.0,0.0,1H ,1,PLTSZE,-NTICK,2,PION2)
C
C        ANNOTATE AXIS, TO THE LEFT
C
      TCH=-.25
      VCH=-.05
      ACH=AMIN
      DO 30 N=1,NTICK
      CALL NUMPLT(TCH,VCH,0.0,-0.1,ACH,1)
      VCH=VCH+VDEL
      ACH=ACH+ADEL
   30 CONTINUE
C
C        LABEL AXIS, AND WRITE TITLE ON PLOT
C
      VCH2=0.5*PLTSZE
      TCH2=-0.8
      VCH=PLTSZE+0.7
      TCH=0.0
      GO TO (110,120,130,140),LF
  110 CONTINUE
      CALL CHAR(TCH2,VCH2,0.0,0.2,1HV,1)
      CALL CHAR(TCH,VCH,0.0,0.2,8HVELOCITY,8)
      TCH=TCH+1.8
      GO TO 150
  120 CONTINUE
      CALL CHAR(TCH2,VCH2,0.0,0.2,1HT,1)
      CALL CHAR(TCH,VCH,0.0,0.2,11HTEMPERATURE,11)
      TCH=TCH+2.4
      GO TO 150
  130 CONTINUE
      CALL GREEK(TCH2,VCH2,0.25,0.0,17)
      CALL CHAR(TCH,VCH,0.0,0.2,7HDENSITY,7)
      TCH=TCH+1.6
      GO TO 150
  140 CONTINUE
      CALL CHAR(TCH2,VCH2,0.0,0.2,1HB,1)
      CALL CHAR(TCH,VCH,0.0,0.2,14HMAGNETIC FIELD,14)
      TCH=TCH+3.0
  150 CONTINUE
      CALL CHAR(TCH,VCH,0.0,0.1,2HVS,2)
      TCH=TCH+.6
      CALL CHAR(TCH,VCH,0.0,0.2,4HTIME,4)
      IF (LF.EQ.2 .OR. LF.EQ.3) GO TO 300
      TCH1=0.5
      VCH1=PLTSZE+0.25
      GO TO (210,220,230,240),LCOMP
C
  210 CONTINUE
      CALL CHAR(TCH1,VCH1,0.0,.14,11H(MAGNITUDE),11)
      CALL PLTLN(TCH2-0.05,VCH2,TCH2-0.05,VCH2+.2)
      CALL PLTLN(TCH2+0.19,VCH2,TCH2+0.19,VCH2+.2)
      GO TO 300
C
  220 CONTINUE
      CALL CHAR(TCH1,VCH1,0.0,.14,13H(X-COMPONENT),13)
      CALL CHAR(TCH2+0.13,VCH2-0.05,0.0,0.1,1HX,1)
      GO TO 300
C
  230 CONTINUE
      CALL CHAR(TCH1,VCH1,0.0,.14,13H(Y-COMPONENT),13)
      CALL CHAR(TCH2+0.13,VCH2-0.05,0.0,0.1,1HY,1)
      GO TO 300
C
  240 CONTINUE
      CALL CHAR(TCH1,VCH1,0.0,.14,13H(Z-COMPONENT),13)
      CALL CHAR(TCH2+0.13,VCH2-0.05,0.0,0.1,1HZ,1)
C
C        ADD 10**N SCALING FACTOR IF REQUIRED
C
  300 CONTINUE
      IF (NA .EQ. 0) GO TO 350
      TCH=-0.90
      VCH=VCH2-0.35
      CALL CHAR(TCH,VCH,0.0,0.16,1HX,1)
      CALL CHAR(TCH+0.09,VCH,0.0,0.12,2H10,2)
      TCH=TCH+0.30
      XNA=FLOAT(NA)
      CALL NUMPLT(TCH,VCH+.11,0.0,.06,XNA,-1)
  350 CONTINUE
      RETURN
      END
```

```
      FUNCTION VINTRP(X,Y)
C
C     THIS FUNCTION INTERPOLATES FOR V AT (X,Y) FROM THE GRID VALUES
C
      COMMON /BLUNT/ THETA(25),RP(20,25),NBLUNT
      COMMON /BOUNDS/ XBOD(100),YBOD(100),XSHK(100),YSHK(100),
     *  NRMAX,NXMAX,AMACH,GAMMA,HRO,NHINDX
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)
      DATA DEG/57.29578/
      IF(X.GT.0.0) GO TO 100
      THET=ATAN(-Y/X)*DEG
      DO 10 NT=2,NBLUNT
      IF(THET.LT.THETA(NT)) GO TO 20
   10 CONTINUE
   20 R=SQRT(X*X+Y*Y)
      DTHET=(THETA(NT)-THET)/(THETA(NT)-THETA(NT-1))
      RP1=DTHET*RP(1,NT-1)+(1.0-DTHET)*RP(1,NT)
      DO 30 NR=2,NRMAX
      RP2=DTHET*RP(NR,NT-1)+(1.0-DTHET)*RP(NR,NT)
      IF(R.LT.RP2) GO TO 40
      RP1=RP2
   30 CONTINUE
   40 DR=(RP2-R)/(RP2-RP1)
      VINTRP=(DTHET*VF(NR-1,NT-1)+(1.0-DTHET)*VF(NR-1,NT))*DR
     *  +(DTHET*VF(NR,NT-1)+(1.0-DTHET)*VF(NR,NT))*(1.0-DR)
      RETURN
  100 R=Y
      DO 110 NT=NBLUNT,NXMAX
      IF(X.LT.XC(1,NT)) GO TO 120
  110 CONTINUE
      NT=NXMAX
  120 DTHET=(XC(1,NT)-X)/(XC(1,NT)-XC(1,NT-1))
      RP1=DTHET*YC(1,NT-1)+(1.0-DTHET)*YC(1,NT)
      DO 130 NR=2,NRMAX
      RP2=DTHET*YC(NR,NT-1)+(1.0-DTHET)*YC(NR,NT)
      IF(R.LT.RP2) GO TO 40
      RP1=RP2
  130 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE WALK(KOD5,J,K,A,JDIM,ICHK,JMIN,JMAX,KMIN,KMAX,KOD6,
     *                NXY,ACONT,KODS,X,Y,NVAL,ISIZ)
C
C        CONTOUR PROGRAMS MAP, WALK, SERCH, ENTER, AND CHECK
C        WRITTEN BY REESE SORENSON, NASA-AMES RES. CTR., AUG., 1974.
C          (MODIFIED VERSION)
C
C        GIVEN ONE POINT ON A LINE, THIS SUBROUTINE WALKS AROUND
C        THE REST OF THE LINE, RECORDING THE REST OF THE LINE.
C
      DIMENSION A(1),ICHK(4,1),ACONT(1)
      DIMENSION K3(3),A1(3),A2(3),JJ(3),KK(3),K2(3)
C
      GO TO (10,11,12,13),KOD5
C
C        ORIENTATION 1: UPWARD
10    JJ(1)=J
      JJ(2)=J
      JJ(3)=J+1
      KK(1)=K
      KK(2)=K+1
      KK(3)=K
      GO TO 15
C
C        ORIENTATION 2: TO THE RIGHT
11    JJ(1)=J
      JJ(2)=J+1
      JJ(3)=J
      KK(1)=K+1
      KK(2)=K
      KK(3)=K
      GO TO 16
C
C        ORIENTATION 3: DOWNWARD
12    JJ(1)=J+1
      JJ(2)=J
      JJ(3)=J
      KK(1)=K-1
      KK(2)=K-1
      KK(3)=K-1
15    K2(1)=1
      K2(2)=2
      K2(3)=1
      GO TO 14
C
C        ORIENTATION 4:  TO THE LEFT
13    JJ(1)=J-1
      JJ(2)=J-1
      JJ(3)=J-1
      KK(1)=K
      KK(2)=K
      KK(3)=K+1
16    K2(1)=2
      K2(2)=1
      K2(3)=2
C
C        SEARCH THE 3 POSSIBLE DIRECTIONS.
14    DO 17 N=1,3
17    CALL SERCH(JJ(N),KK(N),K2(N),A,JDIM,ICHK,NXY,K3(N),NVAL,A1(N),
     *           A2(N),ACONT)
C
      K3S=K3(1)+K3(2)+K3(3)
      IF(K3S-1)1,2,18
C
C        BRANCH POINT
18    IF(K3(KOD5).NE.1)  GO TO 9
      K6=KOD6
      GO TO 4
C
C        ONE IN: ONE OUT.  FIND WHERE OUT.
2     DO 3 K6=1,3
      IF(K3(K6).EQ.1)  GO TO 4
3     CONTINUE
C
C        RECORD THE POINT
4     CALL ENTER(K2(K6),JJ(K6),KK(K6),NVAL,A1(K6),A2(K6),JMIN,KMIN,ICHK,
     *           KOD6,X,Y,NXY,ACONT,ISIZ)
C
C        RESET J AND K
      J=JJ(K6)
      K=KK(K6)
C
C        SEE IF WE'RE AT A BOUNDARY.  IF SO, QUIT.
      IF(J.GT.JMIN)  GO TO 23
      GO TO (26,19,24,23),KOD5
23    IF(J.LT.JMAX)  GO TO 19
      GO TO (24,25,26,19),KOD5
24    IF(K6.EQ.3)  GO TO 1
      GO TO 19
25    IF(K6.EQ.2)  GO TO 1
      GO TO 19
26    IF(K6.EQ.1)  GO TO 1
C
19    IF(K.GT.KMIN)  GO TO 27
      GO TO (9,30,29,28),KOD5
27    IF(K.LT.KMAX)  GO TO 8
      GO TO (29,28,9,30),KOD5
28    IF(K6.EQ.1)  GO TO 1
      GO TO 8
29    IF(K6.EQ.2)  GO TO 1
      GO TO 8
30    IF(K6.EQ.3)  GO TO 1
C
C        PREPARE FOR NEXT STEP
8     KOD6=4-K6
      KOD5=KOD5+K6-2
      IF(KOD5.GT.4)  KOD5=KOD5-4
      IF(KOD5.LT.1)  KOD5=KOD5+4
      GO TO 31
C
    9 CONTINUE
1     KOD5=0
32    RETURN
      END
```

```
      SUBROUTINE ARMATX(J,K,I)
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     * CHORD,NCA,NCB,NCC,AA,H,OMEGA,MU,NL,IT,TAU,ITER,ENT,PTORT,PINF,
```

```
     *RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH            COM1       4
      LEVEL 2,X,Y,XET,XEX,XEY,D                                        COM2       2
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),XEX(40,20,2),       COM2       3
     *  XEY(40,20,2),D(40,20)                                          COM2       4
      LEVEL 2,Q,EF,S,G,AB                                              COM3       2
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)        COM3       3
C...FORM JACOBIAN MATRICES AT A GIVEN J-K NODE POINT.  A MATRIX IF I=1,ABMATX     6
C   B MATRIX IF I=2.                                                   ABMATX     7
      XX=XET(J,K,I)                                                    ABMATX     8
      YY=XEX(J,K,I)                                                    ABMATX     9
      ZZ=XEY(J,K,I)                                                    ABMATX    10
      RI=1.0/Q(J,K,1)                                                  ABMATX    11
      U=Q(J,K,2)*RI                                                    ABMATX    12
      V=Q(J,K,3)*RI                                                    ABMATX    13
      SS=GAMM1*0.5*(U*U+V*V)                                           ABMATX    14
      T=YY*U+ZZ*V                                                      ABMATX    15
      W=GAM*Q(J,K,4)*RI                                                ABMATX    16
      AB(1,1)=XX                                                       ABMATX    17
      AB(1,2)=YY                                                       ABMATX    18
      AB(1,3)=ZZ                                                       ABMATX    19
      AB(1,4)=0.0                                                      ABMATX    20
      AB(2,1)=YY*SS-U*T                                                ABMATX    21
      AB(2,2)=XX-YY*(GAM-2.0)*U+T                                      ABMATX    22
      AB(2,3)=-YY*GAMM1*V+ZZ*U                                         ABMATX    23
      AB(2,4)=YY*GAMM1                                                 ABMATX    24
      AB(3,1)=ZZ*SS-V*T                                                ABMATX    25
      AB(3,2)=YY*V-ZZ*GAMM1*U                                          ABMATX    26
      AB(3,3)=XX-ZZ*(GAM-2.0)*V+T                                      ABMATX    27
      AB(3,4)=ZZ*GAMM1                                                 ABMATX    28
      AB(4,1)=T*(2.0*SS-W)                                             ABMATX    29
      AB(4,2)=(W-SS)*YY-GAMM1*T*U                                      ABMATX    30
      AB(4,3)=(W-SS)*ZZ-GAMM1*T*V                                      ABMATX    31
      AB(4,4)=XX+GAM*T                                                 ABMATX    32
      RETURN                                                           ABMATX    33
      END                                                              ABMATX    34


      SUBROUTINE BLUNTB                                                BLUNTB     2
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,COM1       2
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,  COM1       3
     *RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH            COM1       4
      LEVEL 2,X,Y,XET,XEX,XEY,D                                        COM2       2
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),XEX(40,20,2),       COM2       3
     *  XEY(40,20,2),D(40,20)                                          COM2       4
      LEVEL 2,Q,EF,S,G,AB                                              COM3       2
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)        COM3       3
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NYPLOT                          DNSTRM     2
      COMMON /BLUNT/ THETA(20),RP(20,25),NBLUNT                        BLUNT      2
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)      FLOW       2
      LEVEL 2, VX,VY                                                   VCOMP      2
      COMMON/VCOMP/ VX(20,100),VY(20,100)                              VCOMP      3
      COMMON /TRANSF/ AMACH,GAMF,KM2,PIINF,RIINF,VIINF,PQQ(20,3),      TRANSF     2
     * RHOQ(20,3),UQQ(20,3),VQQ(20,3),WQQ(20,3),RSQQ(3),               TRANSF     3
     * RSTQQ(3),RSPHQ(3),HRQ,RQQ,XINIT                                 TRANSF     4
      COMMON /SHOCKS/ DRSDX(100),DST(50)                               SHOCKS     2
C...INITIALIZE                                                         BLUNTB    12
      WRITE(6,2)                                                       BLUNTB    13
    2 FORMAT(1H1///53X,22HBLUNT BODY CALCULATION/53X,22(1H*)///)       BLUNTB    14
      CALL INITIA                                                      BLUNTB    15
C...DETERMINE STEPSIZE                                                 BLUNTB    16
      CALL EIGEN                                                       BLUNTB    17
C...INTEGRATE EQUATIONS                                                BLUNTB    18
      DO 1 I=1,ITER                                                    BLUNTB    19
      IT=IT+1                                                          BLUNTB    20
      CALL SHOCK                                                       BLUNTB    21
      CALL BNDRY                                                       BLUNTB    22
      CALL INTEGR                                                      BLUNTB    23
      CALL XYETAD(1)                                                   BLUNTB    24
    1 CONTINUE                                                         BLUNTB    25
C...PRINT DATA                                                         BLUNTB    26
      CALL OUTPUT(1)                                                   BLUNTB    27
C                                                                      BLUNTB    28
C......DATA FOR CONTOURS, ALSO WRITTEN TO TAPE9 FOR RERUN              BLUNTB    29
C                                                                      BLUNTB    30
      FACV=1.0/(XMACH*SQRT(GAM))                                       BLUNTB    31
      DO 10 J=2,JMAX                                                   BLUNTB    32
      DRSDX(J)=-XEX(J,KMAX,2)/XEY(J,KMAX,2)                            BLUNTB    33
      DO 10 K=1,KMAX                                                   BLUNTB    34
      XC(K,J)=X(J,K)-1.0                                               BLUNTB    35
      YC(K,J)=Y(J,K)                                                   BLUNTB    36
      VX(K,J)=Q(J,K,2)/Q(J,K,1)*FACV                                   BLUNTB    37
      VY(K,J)=Q(J,K,3)/Q(J,K,1)*FACV                                   BLUNTB    38
      RHOF(K,J)=Q(J,K,1)*D(J,K)                                        BLUNTB    39
   10 CONTINUE                                                         BLUNTB    40
      DO 20 J=2,NBLUNT                                                 BLUNTB    41
      THETA(J)=ATAN2(YC(1,J),-XC(1,J))                                 BLUNTB    42
      DO 20 K=1,KMAX                                                   BLUNTB    43
      RP(K,J)=SQRT(XC(K,J)**2+YC(K,J)**2)                              BLUNTB    44
   20 CONTINUE                                                         BLUNTB    45
      WRITE(9) (J,THETA(J),DRSDX(J),(K,RP(K,J),K=1,KMAX),J=2,NBLUNT)   BLUNTB    46
      DO 30 J=2,NBLUNT                                                 BLUNTB    47
      WRITE(9) J,(K,XC(K,J),YC(K,J),VX(K,J),VY(K,J),RHOF(K,J),K=1,KMAX)BLUNTB    48
   30 CONTINUE                                                         BLUNTB    49
      NX1=NBLUNT+1                                                     BLUNTB    50
      IF (JMAX .EQ. NX1) GO TO 60                                      BLUNTB    51
      DO 40 J=NX1,JM                                                   BLUNTB    52
      WRITE(9) J,XC(1,J),DRSDX(J)                                      BLUNTB    53
      DO 40 K=1,KMAX                                                   BLUNTB    54
      WRITE(9) K,YC(K,J),VX(K,J),VY(K,J),RHOF(K,J)                     BLUNTB    55
   40 CONTINUE                                                         BLUNTB    56
   60 CONTINUE                                                         BLUNTB    57
C                                                                      BLUNTB    58
C     PRINT BODY AND SHOCK LOCATIONS                                   BLUNTB    59
C                                                                      BLUNTB    60
      WRITE(6,100)                                                     BLUNTB    61
      DO 80 J=2,JM                                                     BLUNTB    62
      XBOD=-XC(1,J)                                                    BLUNTB    63
      XSHK=-XC(KMAX,J)                                                 BLUNTB    64
      WRITE(6,110) J,XBOD,YC(1,J),XSHK,YC(KMAX,J)                      BLUNTB    65
   80 CONTINUE                                                         BLUNTB    66
  100 FORMAT(//27H BODY AND FINAL SHOCK SHAPE//                        BLUNTB    67
     *    4X,1HJ,6X,7HX(BODY),5X,7HY(BODY),8X,8HX(SHOCK),6X,8HY(SHOCK))BLUNTB    68
  110 FORMAT(2X,I3,5X,F7.4,5X,F7.4,8X,F7.4,5X,F7.4)                    BLUNTB    69
C                                                                      BLUNTB    70
C......STARTING PLANE FOR MARCHING CODE AT THETA=90 DEGREES            BLUNTB    71
C                                                                      BLUNTB    72
      XINIT=XC(1,JMAX-1)                                               BLUNTB    73
      RQQ=YC(1,NBLUNT)                                                 BLUNTB    74
      AMACH=XMACH                                                      BLUNTB    75
      KM2=KMAX-2                                                       BLUNTB    76
      HRQ=HORN                                                         BLUNTB    77
      GAMF=GAM                                                         BLUNTB    78
      GAM1=GAMM1                                                       BLUNTB    79
      PK1=AMACH**2*0.5*GAM1                                            BLUNTB    80
      PK2=1.0+PK1                                                      BLUNTB    81
      PKPINF=PK2**(-GAM/GAM1)                                          BLUNTB    82
      PKRINF=PK2**(-1./GAM1)                                           BLUNTB    83
      PKQINF=SQRT(PK1/PK2)                                             BLUNTB    84
      DO 300 K=1,KMAX                                                  BLUNTB    85
      RHO=Q(JM,K,1)*D(JM,K)                                            BLUNTB    86
      U=Q(JM,K,2)/Q(JM,K,1)                                            BLUNTB    87
      V=Q(JM,K,3)/Q(JM,K,1)                                            BLUNTB    88
      EN=Q(JM,K,4)*D(JM,K)                                             BLUNTB    89
      P=GAM1*(EN-RHO*0.5*(U*U+V*V))                                    BLUNTB    90
      DO 300 I=1,3                                                     BLUNTB    91
      PQQ(K,I)=P*PKPINF/PINF                                           BLUNTB    92
      RHOQ (K,I)=RHO*PKRINF/RINF                                       BLUNTB    93
      UQQ(K,I)=U*PKQINF/QINF                                           BLUNTB    94
      VQQ(K,I)=V*PKQINF/QINF                                           BLUNTB    95
      WQQ(K,I)=0.0                                                     BLUNTB    96
  300 CONTINUE                                                         BLUNTB    97
      DO 310 KK=1,3                                                    BLUNTB    98
      RSQQ(KK)=Y(JM,KMAX)                                              BLUNTB    99
      RSTQQ(KK)=-XEX(JM,KMAX,2)/XEY(JM,KMAX,2)                         BLUNTB   100
      RSPHQ(KK)=0.0                                                    BLUNTB   101
  310 CONTINUE                                                         BLUNTB   102
      RETURN                                                           BLUNTB   103
      END                                                              BLUNTB   104


      SUBROUTINE BNDRY                                                 BNDRY      2
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,COM1       2
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,  COM1       3
     *RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH            COM1       4
      LEVEL 2,X,Y,XET,XEX,XEY,D                                        COM2       2
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),XEX(40,20,2),       COM2       3
     *  XEY(40,20,2),D(40,20)                                          COM2       4
      LEVEL 2,Q,EF,S,G,AB                                              COM3       2
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)        COM3       3
      COMMON /PUV/ P(40,3),PXI(40),PETA(40),U(40,3),UXI(40),UETA(40),  PUV        2
     *  V(40,3),VXI(40),VETA(40),R(40,3)                               PUV        3
C...USE REFLECTION TO SIMULATE PLANE OF SYMMETRY AT J=2                BNDRY      7
```

```
      DO 12 K=1,KMAX
      DD=D(2,K)/D(1,K)
      Q(1,K,1)=Q(2,K,1)*DD
      Q(1,K,2)=Q(2,K,2)*DD
      Q(1,K,3)=-Q(2,K,3)*DD
   12 Q(1,K,4)=Q(2,K,4)*DD
C...USE FIRST ORDER EXTRAPOLATION TO SIMULATE SUPERSONIC OUTFLOW
C...BOUNDARY CONDITION AT JMAX
      DO 1 N=1,4
      DO 1 K=2,KM
    1 Q(JMAX,K,N)=(2.*Q(JM,K,N)*D(JM,K)-Q(JM-1,K,N)*D(JM-1,K))/
     > D(JMAX,K)
C...SATISFY TANGENCY CONDITION USING CHARACTERISTIC EQUATION
      DO 3 K=1,3
      DO 3 J=1,JMAX
      Z=1./Q(J,K,1)
      R(J,K)=Q(J,K,1)*D(J,K)
      U(J,K)=Q(J,K,2)*Z
      V(J,K)=Q(J,K,3)*Z
      E2=Q(J,K,4)*D(J,K)
    3 P(J,K)=(E2-0.5*R(J,K)*(U(J,K)**2+V(J,K)**2))*GAMM1
C...COMPUTE P-XI, U-XI,V-XI, P-ETA,U-ETA, AND V-ETA DERIVATIVES
      DO 4 J=2,JM
      PXI(J)=(P(J+1,1)-P(J-1,1))*0.5
      UXI(J)=(U(J+1,1)-U(J-1,1))*0.5
    4 VXI(J)=(V(J+1,1)-V(J-1,1))*0.5
      PXI(1)=-PXI(2)
      UXI(1)=-UXI(2)
      VXI(1)=VXI(2)
      PXI(JMAX)=(3.*P(JMAX,1)-4.*P(JM,1)+P(JM-1,1))*0.5
      UXI(JMAX)=(3.*U(JMAX,1)-4.*U(JM,1)+U(JM-1,1))*0.5
      VXI(JMAX)=(3.*V(JMAX,1)-4.*V(JM,1)+V(JM-1,1))*0.5
      DO 5 J=1,JMAX
      PETA(J)=(-3.*P(J,1)+4.*P(J,2)-P(J,3))*0.5
      UETA(J)=(-3.*U(J,1)+4.*U(J,2)-U(J,3))*0.5
      VETA(J)=(-3.*V(J,1)+4.*V(J,2)-V(J,3))*0.5
      IF(J.EQ.1.OR.J.EQ.JMAX) GO TO 5
      P(J,2)=P(J+1,1)+P(J-1,1)-2.*P(J,1)
    5 CONTINUE
      P(1,2)=P(2,2)
      P(JMAX,2)=0.
      K=1
      DO 2 J=1,JMAX
      CRR=SQRT(GAM*P(J,1)/R(J,1))
      Z=1.0/SQRT(XEY(J,1,2)**2+YEY(J,1,2)**2)
      UBAR=XET(J,1,1)*U(J,1)+XEY(J,1,1)+V(J,1)*YEY(J,1,1)
      VBAR=XET(J,1,2)*U(J,1)+XEY(J,1,2)+V(J,1)*YEY(J,1,2)
      EE=UBAR*PXI(J)+R(J,1)*CRR*Z*(XEY(J,1,1)*UXI(J)+YEY(J,1,1)*VXI(J))
     > -CRR*Z*(YEY(J,1,2)*(XEY(J,1,1)*PXI(J)+R(J,1)*UBAR*UXI(J))+
     > YEY(J,1,2)*(XEY(J,1,1)*PXI(J)+R(J,1)*UBAR*VXI(J)))
     > +(V(J,1)/Y(J,1))*FLOAT(JCS)
      PTAU=CRR/Z*PETA(J)-R(J,1)*CRR**2*(XEY(J,1,2)*UETA(J)+YEY(J,1,2)*
     > VETA(J))-EE
      P1=P(J,1)+PTAU*DT*0.2+0.10*P(J,2)
      IF (P1 .LT. 0.0) GO TO 9
   10 CONTINUE
      IF(J.LE.2) P1=PT+(P(3,1)-PT)*0.11111111
      R1=(P1/ENT)**(1.0/GAM)
      Q1=SQRT(2.*GAM/GAMM1*ABS(PTOT-P1/R1))
      IF(ABS(YEY(J,1,2))-0.0001) 6,6,7
    6 THETA=1.570796327
      GO TO 8
    7 THETA=ATAN(-XEY(J,1,2)/YEY(J,1,2))
    8 CONTINUE
      U1=Q1*COS(THETA)
      V1=Q1*SIN(THETA)
      DI=1./D(J,1)
      Q(J,1,1)=R1*DI
      Q(J,1,2)=R1*U1*DI
      Q(J,1,3)=R1*V1*DI
      Q(J,1,4)=(P1/GAMM1+0.5*R1*Q1**2)*DI
    2 CONTINUE
      RETURN
C
C     PRINT MESSAGE FOR NEGATIVE PRESSURE
C
    9 WRITE(6,100) P1,J
      P1=ABS(P1)
      GO TO 10
  100 FORMAT(5X,44HNEGATIVE PRESSURE ON BODY DETECTED BY BNDRY,,2X,
     *  3HP1=,1PE10.3,6H AT J=,I2)
      END
```

```
      SUBROUTINE BODY(Y,X,RNOSE,ANG,H)
      LOGICAL LGRAV
      COMMON/NUBOD/XX(100),YY(100),NBOD,LGRAV
      E(A)=EXP(-ABS(A-RNOSE)/H)
      F(A,B,K)=.5*A*((SIN(2.*B)+(-1.)**K*2.*SQRT(E(A)-E(A)**2))
     >      /(E(A)-SIN(B)**2))
      G(A,B)=ABS(A*((A**6*SIN(B)*COS(B)+SQRT(A**6-1.))/(A**6*COS(B)**2-1
     *      .)))
      F2(A,B,C,D)=A*(B*C-SQRT(D*(1.-B)))/(D-B*B)
      F3(A,B)=EXP(-ABS(1.-1./A)/B)
C
C...... H .NE. 0.0 : IONOPAUSE FOR A NONMAGNETIC PLANET
C...... H .EQ. 0.0 : EQUATORIAL PLANE FOR A MAGNETIC PLANET
C
      IF(H .EQ. 0.0)GO TO 10
      IF (H.LT.0.0) GO TO 35
C
C......THIS DETERMINES THE BODY SHAPE OF A NONMAGNETIC PLANET
C
C
      IMAX=301
      IMAXM1=IMAX-1
      IF(ANG .EQ. 0.)IMAXM1=1
      PI=3.1415926535898
      RAD=180./PI
      DELTAT=ANG/FLOAT(IMAXM1)
      THETA=0.
      RI=RNOSE
      ISTART=1
      IF (H.LE.0.1) GO TO 20
C
C......PERFORM INTEGRATION FROM 0 TO <ANG> DEGREES
C
 2    CONTINUE
      IF (LGRAV) GO TO 45
      DO 5 I=ISTART,IMAXM1
C......PREDICTOR
      K=1
      THETA1=THETA
      N=1
      R=RI+DELTAT*F(RI,THETA1,N)
      THETA=THETA+DELTAT
C......CORRECTOR
 1    CONTINUE
      R=RI+.5*DELTAT*(F(RI,THETA1,N)+F(R,THETA,N))
      K=K+1
      IF(K .LT. 5)GO TO 1
      X=1.-R*COS(THETA)
      Y=R*SIN(THETA)
      RI=R
 5    CONTINUE
      RETURN
 20   CONTINUE
      IF (H .LT. 0.1) GO TO 30
      DO 25 I=1,IMAXM1
      R=RNOSE+H*SIN(THETA)**2
      X=1.0-R*COS(THETA)
      Y=R*SIN(THETA)
      ISTART=I
      RI=R
      IF (THETA .GE. 5.0/RAD) GO TO 2
      THETA=THETA+DELTAT
   25 CONTINUE
C
C       USE SINUSOIDAL NOSE SHAPE FOR H/R0 .LT. 0.1
C
   30 R=RNOSE+H*SIN(ANG)**2
      X=1.0-R*COS(ANG)
      Y=R*SIN(ANG)
      RETURN
C
C......THIS DETERMINES THE BODY SHAPE OF A MAGNETIC PLANET
C
 10   CONTINUE
      IMAX=301
      IMAXM1=IMAX-1
      IF(ANG .EQ. 0.0)IMAXM1=1
      PI=3.1415926535898
      RAD=180./PI
      DELTAT=ANG/FLOAT(IMAXM1)
      THETA=PI/2.
      RI=RNOSE
```

```
C
C.....PERFORM INTEGRATION FROM 0 TO ANG DEGREES
C
      DO 15 I=1,IMAXM1
C.....PREDICTOR
      THETA1=THETA
      R=R1+DELTAT*G(R1,THETA)
      THETA=THETA+DELTAT
C.....CORRECTOR
      R=R1+0.5*DELTAT*(G(R1,THETA1)+G(R,THETA))
      R1=R
 15   CONTINUE
      X=1.0-R*COS(ANG)
      Y=R*SIN(ANG)
      RETURN
C
C......INTERPOLATE FROM USER-SUPPLIED BODY SHAPE
C
   35 CONTINUE
      DO 39 I=2,NBOD
      IF (ATAN2(YY(I),XX(I)) .GT. ANG) GO TO 40
   39 CONTINUE
   40 R1=SQRT(YY(I)**2+XX(I)**2)
      R2=SQRT(YY(I-1)**2+XX(I-1)**2)
      ANG1=ATAN2(YY(I),XX(I))
      ANG2=ATAN2(YY(I-1),XX(I-1))
      R=R1+(ANG-ANG1)/(ANG2-ANG1)*(R2-R1)
      X=1.0-R*COS(ANG)
      Y=SIN(ANG)*R
      RETURN
C
C.....DETERMINES BODY SHAPE WITH GRAVITATIONAL VARIATION IN H
C
 45   CONTINUE
      IF (H.LT.0.1) GO TO 20
C
C.....PERFORM INTEGRATION FROM 0 TO ANG DEGREES
C
      DO 55  I=ISTART,IMAXM1
      AS=SIN(THETA)
      AC=COS(THETA)
      THETA1=THETA
C.....PREDICTOR
      A1=F3(R1,H)
      K=1
      DRX=F2(R1,AS,AC,A1)
      IF (DRX.LT.0.0) DRX=2.*H*AS*AC
      R=R1+DRX*DELTAT
      THETA=THETA+DELTAT
      AS1=SIN(THETA)
      AC1=COS(THETA)
C......CORRECTOR
 50   CONTINUE
      A11=F3(R,H)
      DRXX=F2(R,AS1,AC1,A11)
      IF (DRXX.LT.0.0) DRXX=2.*H*AS1*AC1
      SLOPE=.5*(DRX+DRXX)
      R=R1+SLOPE*DELTAT
      K=K+1
      IF (K.LT.5) GO TO 50
      X=1.0-R*AC1
      Y=R*AS1
      R1=R
 55   CONTINUE
      RETURN
      END
```

```
      SUBROUTINE BTRI(IL,IU)
      LEVEL 2,Q,EF,S,G,AB
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)
      LEVEL 2, A,B,C,HD
      COMMON /COM4/ A(40,4,4),B(40,4,4),C(40,4,4),HD(40,4,4)
      REAL L11,L21,L22,L31,L32,L33,L41,L42,L43,L44
      COMMON /LUD/ L11,L21,L22,L31,L32,L33,L41,L42,L43,L44,V1,V2,V3,V4,
     *   U12,U13,U14,U23,U24,U34
      DIMENSION H(4,4)
C  INVERSION OF BLOCK TRIDIAGONAL... A,B,C ARE 4*4 BLOCKS
C  EF IS FORCING FUNCTION AND SOLUTION IS OUTPUT IN EF, B IS OVERLOADED
C  BLOCK INVERSIONS USE NONPIVOTED LU DECOMPOSITION
C  IL AND IU ARE STARTING AND FINISHING INDICES
      IS = IL +1
      I = IL
      DO 11 N=1,4
      DO 11 M=1,4
   11 H(N,M)=B(I,N,M)
      CALL LUDEC(H)
      D1 = V1*EF(I,1)
      D2 = V2*(EF(I,2) - L21*D1)
      D3 = V3*(EF(I,3) - L31*D1 - L32*D2)
      D4 = V4*(EF(I,4) - L41*D1 - L42*D2 - L43*D3)
      EF(I,4) = D4
      EF(I,3) = D3 - U34*D4
      EF(I,2) = D2 - U24*D4 - U23*EF(I,3)
      EF(I,1) = D1 - U14*D4 - U13*EF(I,3) - U12*EF(I,2)
      DO 12 M=1,4
      D1 = V1*C(I,1,M)
      D2 = V2*( C(I,2,M) - L21*D1)
      D3 = V3*( C(I,3,M) - L31*D1 - L32*D2)
      D4 = V4*( C(I,4,M) -L41*D1 - L42*D2 - L43*D3)
      B(I,4,M) = D4
      B(I,3,M) = D3 - U34*D4
      B(I,2,M) = D2 - U24*D4 - U23*B(I,3,M)
      B(I,1,M) = D1 - U14*D4 - U13*B(I,3,M) - U12*B(I,2,M)
C...FORWARD SWEEP
   12 CONTINUE
      DO 13 I=IS,IU
      IR = I -1
      DO 14 N=1,4
      EF(I,N) = EF(I,N) -A(I,N,1)*EF(IR,1) -A(I,N,2)*EF(IR,2)
     *       -A(I,N,3)*EF(IR,3) -A(I,N,4)*EF(IR,4)
      DO 14 M=1,4
      H(N,M) = B(I,N,M) -A(I,N,1)*B(IR,1,M) -A(I,N,2)*B(IR,2,M)
     1     -A(I,N,3)*B(IR,3,M) - A(I,N,4)*B(IR,4,M)
   14 CONTINUE
      CALL LUDEC(H)
      D1 = V1*EF(I,1)
      D2 = V2*(EF(I,2) - L21*D1)
      D3 = V3*(EF(I,3) - L31*D1 - L32*D2)
      D4 = V4*(EF(I,4) - L41*D1 - L42*D2 - L43*D3)
      EF(I,4) = D4
      EF(I,3) = D3 - U34*D4
      EF(I,2) = D2 - U24*D4 - U23*EF(I,3)
      EF(I,1) = D1 - U14*D4 - U13*EF(I,3) - U12*EF(I,2)
      IF( I - IU)16,13,13
   16 CONTINUE
      DO 15 M=1,4
      D1 = V1*C(I,1,M)
      D2 = V2*( C(I,2,M) - L21*D1)
      D3 = V3*( C(I,3,M) - L31*D1 - L32*D2)
      D4 = V4*( C(I,4,M) -L41*D1 - L42*D2 - L43*D3)
      B(I,4,M) = D4
      B(I,3,M) = D3 - U34*D4
      B(I,2,M) = D2 - U24*D4 - U23*B(I,3,M)
      B(I,1,M) = D1 - U14*D4 - U13*B(I,3,M) - U12*B(I,2,M)
   15 CONTINUE
   13 CONTINUE
C...BACK SUBSTITUTION
      IT = IL + IU
      DO 21 II = IS,IU
      I = IT - II
      IP = I+1
      DO 22 N=1,4
      EF(I,N) = EF(I,N) -B(I,N,1)*EF(IP,1) -B(I,N,2)*EF(IP,2)
     *       -B(I,N,3)*EF(IP,3) -B(I,N,4)*EF(IP,4)
   22 CONTINUE
   21 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE DISSIP
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,X,Y,XET,XEX,XEY,D
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),XEX(40,20,2),
     *  XEY(40,20,2),D(40,20)
      LEVEL 2,Q,EF,S,G,AB
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)
C...SMOOTH IN THE X AND Y DIRECTIONS AND ADD SMOOTHING TERM TO S ARRAY
      KMM=KM-1
```

```
      JMM=JM-1
C...SMOOTHING IN XI DIRECTION
      DO 1 N=1,4
      DO 2 K=2,KM
      S(2,K,N)=S(2,K,N)-SMU*0.125*(-2.0*Q(1,K,N)*D(1,K)+5.0*Q(2,K,N)
     > *D(2,K)-4.0*Q(3,K,N)*D(3,K)+Q(4,K,N)*D(4,K))/D(2,K)
      S(JM,K,N)=S(JM,K,N)-SMU*0.125*(-2.0*Q(JMAX,K,N)*D(JMAX,K)+
     > 5.0*Q(JM,K,N)*D(JM,K)-4.0*Q(JMM,K,N)*D(JMM,K)+Q(JM-2,K,N)
     > *D(JM-2,K))/D(JM,K)
      DO 2 J=3,JMM
    2 S(J,K,N)=S(J,K,N)-SMU*0.125*(Q(J-2,K,N)*D(J-2,K)-4.0*Q(J-1,K,N)
     > *D(J-1,K)+6.0*Q(J,K,N)*D(J,K)-4.0*Q(J+1,K,N)*D(J+1,K)+Q(J+2,K,N)
     > *D(J+2,K))/D(J,K)
C...SMOOTHING IN ETA DIRECTION
      DO 1 J=2,JM
      S(J,2,N)=S(J,2,N)-SMU*0.125*(-2.0*Q(J,1,N)*D(J,1)+5.0*Q(J,2,N)
     > *D(J,2)-4.0*Q(J,3,N)*D(J,3)+Q(J,4,N)*D(J,4))/D(J,2)
      S(J,KM,N)=S(J,KM,N)-SMU*0.125*(-2.0*Q(J,KMAX,N)*D(J,KMAX)+
     > 5.0*Q(J,KM,N)*D(J,KM)-4.0*Q(J,KMM,N)*D(J,KMM)+Q(J,KM-2,N)
     > *D(J,KM-2))/D(J,KM)
      DO 1 K=3,KMM
    1 S(J,K,N)=S(J,K,N)-SMU*0.125*(Q(J,K-2,N)*D(J,K-2)-4.0*Q(J,K-1,N)
     > *D(J,K-1)+6.0*Q(J,K,N)*D(J,K)-4.0*Q(J,K+1,N)*D(J,K+1)+Q(J,K+2,N)
     > *D(J,K+2))/D(J,K)
      RETURN
      END

      SUBROUTINE EFCON(J,K,I)
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITEP,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,X,Y,XET,XEX,XEY,D
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),YEX(40,20,2),
     *  XEY(40,20,2),D(40,20)
      LEVEL 2,Q,EE,S,G,AB
      COMMON /COM3/ Q(40,20,4),EE(40,4),S(40,20,4),G(4),AB(4,4)
C...FORM E CONSERVATIVE VARIABLES (I=1) OR F CONSERVATIVE VARIABLES
C...(I=2) AT A GIVEN NODE POINT
      W=Q(J,K,1)
      RI=1./W
      U=Q(J,K,2)*RI
      V=Q(J,K,3)*RI
      PDJ=GAMM1*(Q(J,K,4)-W*.5*(U*U+V*V))
      XX=XET(J,K,I)
      YY=YEX(J,K,I)
      Z=XEY(J,K,I)
      CAPUV=XX+YY*U+Z*V
      G(1)=W*CAPUV
      G(2)=Q(J,K,2)*CAPUV+YY*PDJ
      G(3)=Q(J,K,3)*CAPUV+Z*PDJ
      G(4)=Q(J,K,4)*CAPUV+(CAPUV-XX)*PDJ
C...SOURCE TERM IN ETA-MOM. EQN. FOR AXISYMMETRIC FLOW
      IF(JCS.EQ.0.OR.I.EQ.1.OR.K.EQ.1.OR.K.EQ.KMAX) RETURN
      YI=DT/Y(J,K)
      S(J,K,1)=S(J,K,1)+Q(J,K,3)*YI
      S(J,K,2)=S(J,K,2)+Q(J,K,3)*YI*U
      S(J,K,3)=S(J,K,3)+Q(J,K,3)*YI*V
      S(J,K,4)=S(J,K,4)+(Q(J,K,4)+PDJ)*V*YI
      RETURN
      END

      SUBROUTINE EIGEN
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITEP,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,X,Y,XET,XEX,XEY,D
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),YEX(40,20,2),
     *  XEY(40,20,2),D(40,20)
      LEVEL 2,Q,EE,S,G,AB
      COMMON /COM3/ Q(40,20,4),EE(40,4),S(40,20,4),G(4),AB(4,4)
C...COMPUTE STEPSIZE GIVEN COURANT NUMBER
      IF(IPRT.GT.0) WRITE(6,103)
      SIGMAX=0.0
      DO 1 K=1,KMAX
      DO 1 J=1,JMAX
      RI=1.0/Q(J,K,1)
      U=Q(J,K,2)*RI
      V=Q(J,K,3)*RI
      SPSND=SQRT(GAM*GAMM1*(Q(J,K,4)*RI-0.5*(U*U+V*V)))
      XIX=XEX(J,K,1)
      XIY=XEY(J,K,1)
      ETAX=XEX(J,K,2)
      ETAY=XEY(J,K,2)
      SIGA=ABS(XET(J,K,1)+U*XIX+V*XIY)+SPSND*SQRT(XIX**2+XIY**2)
      SIGB=ABS(XET(J,K,2)+U*ETAX+V*ETAY)+SPSND*SQRT(ETAX**2+ETAY**2)
      SIGAB=AMAX1(SIGA,SIGB)
      IF(SIGAB.GT.SIGMAX) SIGMAX=SIGAB
      IF(IPRT.EQ.0) GO TO 1
      WRITE(6,101) J,K,SIGA,SIGB
    1 CONTINUE
      DT=CN/ABS(SIGMAX)
      H=0.5*DT
  103 FORMAT(1H0,3X,1HJ,4X,1HK,7X,4HSIGA,8X,4HSIGB)
  101 FORMAT(2I5,2F12.6)
  102 FORMAT(//22H DETERMINING STEPSIZE:,3X,7HSIGMAX=,1PE11.4,3X,3HCN=,
     *  E9.2,3X,3HDT=,E11.4//)
      RETURN
      END

      SUBROUTINE INITIA
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITEP,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,X,Y,XET,XEX,XEY,D
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),YEX(40,20,2),
     *  XEY(40,20,2),D(40,20)
      LEVEL 2,Q,EE,S,G,AB
      COMMON /COM3/ Q(40,20,4),EE(40,4),S(40,20,4),G(4),AB(4,4)
      LEVEL 2, A,B,C,HD
      COMMON /COM4/ A(40,4,4),B(40,4,4),C(40,4,4),HD(40,4,4)
      PI=3.141592654
      RADI=57.29577951
      RINF=1.0
      PINF=1.
      IPRT=0
      ALPHA=0.0
      JCS=1
      RNOSE=1.0
      SMU=.4
      CINF=SQRT(PINF*GAM/RINF)
      ALPH=ALPHA/RADI
      QINF=XMACH*CINF
      UINF=QINF*COS(ALPH)
      VINF=QINF*SIN(ALPH)
      JM=JMAX-1
      KM=KMAX-1
      GAMM1=GAM-1.0
      GAMIT=1.0/GAMM1
      TAU=0.0
      IT=0
C...LOCATE X AND Y NODAL POINTS OF MESH
      CALL NDLPTS
C...COMPUTE METRICS
      CALL XIETAD(3)
C...INITIALIZE Q VECTOR TO FREE STREAM VALUES
      DO 1 K=1,KMAX
      DO 1 J=1,JMAX
      DI=1.0/D(J,K)
      Q(J,K,1)=RINF*DI
      Q(J,K,2)=RINF*UINF*DI
      Q(J,K,3)=RINF*VINF*DI
      Q(J,K,4)=(PINF*GAMIT+RINF*QINF**2*0.5)*DI
C...SET S ARRAY TO 0 EVERYWHERE
      DO 1 N=1,4
    1 S(J,K,N)=0.0
C...INITIALIZE FLOW FIELD FOR BLUNT BODY PROBLEM
      GAMP1=GAM+1.0
      DO 2 J=2,JMAX
      IF(ABS(XEX(J,1,2))-0.000001) 6,6,7
    6 THET=90./RADI
      GO TO 8
    7 THET=ATAN(-XEY(J,1,2)/XEX(J,1,2))
    8 CONTINUE
      K=KMAX
      SANG=0.5*PI-ATAN(-XEY(J,K,2)/XEX(J,K,2))
      XX=XMACH**2*SIN(SANG)**2
```

```
      PS=(2.0*GAM*XX-GAMM1)/GAMP1*PINF
      RS=GAMP1*XX/(GAMM1*XX+2.0)*RINF
      US=(1.0-2.0*(XX-1.0)/GAMP1/XMACH**2)*QINF
      VS=2.0*(XX-1.0)*COS(SANG)/(GAMP1*XMACH**2*SIN(SANG))*QINF
      IF(J.GT.2) GO TO 3
      X1=(2.0*GAM*XMACH**2-GAMM1)/(GAM+1.0)
      X2=(GAM+1.0)*XMACH**2/(GAMM1*XMACH**2+2.0)
      P1=X1*PINF
      R1=X2*RINF
      ENT=P1/R1**GAM
      PT=(1.0/X1)**(1.0/GAMM1)*(0.5*(GAM+1.)*XMACH**2)**(GAM/GAMM1)*PINF
      XX=1.0+0.5*GAMM1*XMACH**2
      PTORT=XX*PINF/RINF
    3 CONTINUE
      PB=PINF*((PT/PINF-1.0)*(1.0-1.02*SIN(THET)**2+0.12*SIN(THET)**4)+
     > 1.0)
      RB=(PB/ENT)**(1.0/GAM)
      QB=SQRT(2.0*GAM/GAMM1*ABS(PTORT-PB/RB))
      YY=PI*0.5-THET
      UB=QB*COS(YY)
      VB=QB*SIN(YY)
      ZKM=KMAX-1
      DO 2 K=1,KMAX
      YY=FLOAT(K-1)/ZKM
      PRESS=PB+YY*(PS-PB)
      RHO=RB+YY*(RS-RB)
      UVEL=UB+YY*(US-UB)
      VVEL=VB+YY*(VS-VB)
      DI=1.0/D(J,K)
      Q(J,K,1)=RHO*DI
      Q(J,K,2)=RHO*UVEL*DI
      Q(J,K,3)=RHO*VVEL*DI
    2 Q(J,K,4)=(PRESS*GAM1I+RHO*(UVEL**2+VVEL**2)*0.5)*DI
C...REFLECT METRICS AND DEPENDENT VARIABLES ABOUT PLANE OF SYMMETRY
      DO 4 K=1,KMAX
      DD=(-1.0)**JCS
      DD=(+1.0)**JCS
      D(1,K)=D(2,K)*DD
      XEX(1,K,1)=-XEX(2,K,1)
      XEY(1,K,1)=XEY(2,K,1)
      XEX(1,K,2)=XEX(2,K,2)
      XEY(1,K,2)=-XEY(2,K,2)
      DO 5 N=1,4
    5 Q(1,K,N)=Q(2,K,N)*DD
    4 Q(1,K,3)=-Q(2,K,3)*DD
      RETURN
      END
```

```
      SUBROUTINE INTEGR
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TH,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,Q,EF,S,G,AB
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)
C...COMPUTE FORCING FUNCTION AND STORE TEMPORARILY IN S ARRAY
      CALL RHS
C...ADD FOURTH ORDER DISSIPATION TO SMOOTH SOLUTION
      CALL DISSIP
C...SOLVE FOR Q-BAR-BAR
      DO 1 K=2,KM
C...FILL ELEMENTS OF I+H*DX A FOR BLOCK TRIDIAGONAL INVERSION AT EACH
C...K TH LEVEL
      CALL LBLTRA(K)
C...INVERT BLOCK TRIDIAGONAL MATRIX AT K TH LEVEL AND STORE SOLUTION IN
C...S ARRAY
      CALL BTRI(2,JM)
      DO 1 L=1,4
      DO 1 J=2,JM
    1 S(J,K,L)=EF(J,L)
C...SOLVE FOR Q-BAR
      DO 2 J=2,JM
C...FILL ELEMENTS OF I+H*DY B FOR BLOCK TRIDIAGONAL INVERSION AT EACH
C...J TH LEVEL
      CALL LBLTRB(J)
C...INVERT BLOCK TRIDIAGONAL MATRIX AT J TH LEVEL AND STORE SOLUTION IN
C...Q ARRAY
      CALL BTRI(2,KM)
      DO 2 L=1,4
      DO 2 K=2,KM
    2 Q(J,K,L)=EF(K,L)+Q(J,K,L)
      TAU=TAU+DT
      RETURN
      END
```

```
      SUBROUTINE LBLTRA(K)
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TH,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,Q,EF,S,G,AB
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)
      LEVEL 2, A,B,C,HD
      COMMON /COM4/ A(40,4,4),B(40,4,4),C(40,4,4),HD(40,4,4)
      DO 1 J=1,JMAX
C...LOAD BLOCK A MATRIX EVALUATED AT N TH LEVEL FOR ALL J INTO HD ARRAY
      CALL ABMATX(J,K,1)
      DO 1 M=1,4
      DO 1 L=1,4
    1 HD(J,L,M)=AB(L,M)
C...FILL OFF-DIAGONAL AND DIAGONAL ELEMENTS BASED ON A 2-ND ORDER
C...CENTRAL DIFFERENCE
      DO 2 J=2,JM
      DO 2 M=1,4
      DO 3 L=1,4
      A(J,L,M)=-HD(J-1,L,M)*H
      B(J,L,M)=0.0
    3 C(J,L,M)=HD(J+1,L,M)*H
C...FILL FORCING FUNCTION FROM S ARRAY
      EF(J,M)=S(J,K,M)
C...SET B ON THE DIAGONAL REPRESENTING THE IDENTITY MATRIX TO ONE
    2 B(J,M,M)=1.0
      RETURN
      END
```

```
      SUBROUTINE LBLTRB(J)
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TH,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,Q,EF,S,G,AB
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)
      LEVEL 2, A,B,C,HD
      COMMON /COM4/ A(40,4,4),B(40,4,4),C(40,4,4),HD(40,4,4)
      DO 1 K=1,KMAX
C...LOAD BLOCK B MATRIX EVALUATED AT N TH LEVEL FOR ALL K INTO HD ARRAY
      CALL ABMATX(J,K,2)
      DO 1 M=1,4
      DO 1 L=1,4
    1 HD(K,L,M)=AB(L,M)
C...FILL OFF-DIAGONAL AND DIAGONAL ELEMENTS BASED ON A 2-ND ORDER
C...CENTRAL DIFFERENCE
      DO 2 K=2,KM
      DO 2 M=1,4
      DO 3 L=1,4
      A(K,L,M)=-HD(K-1,L,M)*H
      B(K,L,M)=0.0
    3 C(K,L,M)=HD(K+1,L,M)*H
C...FILL FORCING FUNCTION FROM S ARRAY
      EF(K,M)=S(J,K,M)
C...SET B ON THE DIAGONAL REPRESENTING THE IDENTITY MATRIX TO ONE
    2 B(K,M,M)=1.0
      RETURN
      END
```

```
      SUBROUTINE LUDEC(A)
      DIMENSION A(4,4)
      REAL L11,L21,L22,L31,L32,L33,L41,L42,L43,L44
      COMMON /LUD/ L11,L21,L22,L31,L32,L33,L41,L42,L43,L44,V1,V2,V3,V4,
     *   U12,U13,U14,U23,U24,U34
C   SUBROUTINE COMPUTES L-U DECOMPOSITION ELEMENTS
      L11 = A(1,1)
       V1 = 1./L11
```

```
      U12 =  V1*A(1,2)
      U13 =  V1*A(1,3)
      U14 =  V1*A(1,4)
      L21 = A(2,1)
      L22 = A(2,2) - L21*U12
       V2 = 1./L22
      U23 = ( A(2,3) -L21*U13)* V2
      U24 = ( A(2,4) - L21*U14)* V2
      L31 = A(3,1)
      L32 = A(3,2) - L31*U12
      L33 = A(3,3) - L31*U13 -L32*U23
       V3 = 1./L33
      U34 = ( A(3,4) - L31*U14 - L32*U24)* V3
      L41 = A(4,1)
      L42 = A(4,2) - L41*U12
      L43 = A(4,3) - L41*U13 - L42*U23
      L44 = A(4,4) - L41*U14 -L42*U24 -L43*U34
       V4 = 1./L44
      RETURN
      END
```

```
      SUBROUTINE NOLPTS
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,X,Y,XET,XEX,XEY,D
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),XEX(40,20,2),
     *  XEY(40,20,2),D(40,20)
      LOGICAL LRERUN,LPRFL,LPRST,LPRCON,LPRR,LPLOT,LTRAJ,LRSTRT
      COMMON /PROPT/ LRERUN,LPRFL,LPRST,LPRCON,LPRR,LPLOT,LTRAJ,LRSTRT
      COMMON /BLUNT/ THETA(25),RP(20,25),NBLUNT
C...THIS SUBROUTINE DETERMINES THE X AND Y LOCATIONS OF THE NODE POINTS
      DTHET=1.57079633/(FLOAT(NBLUNT)-1.5)
C...J=0 FOR CYLINDER, J=1 FOR SPHERE.
      DELTO=(GAMM1*XMACH**2+2.0)/((GAM+1.0)*XMACH**2)*0.78*3.0**(1-JCS)
      IF(XMACH.LT.3.) DELTO=(2.02476*(XMACH-1.25)**2-5.65390*5*(XMACH-
     > 1.25)+4.4)/4.0**JCS
      IF (HORN .LT. 0.0) GO TO 10
      IF(HORN .LT. 0.1 .AND. HORN .GT. 0.0)GO TO 10
      ANG=90./57.2957795
      CALL BODY(Y90,X90,RNOSE,ANG,HORN)
      DELTO=DELTO*(1.2790+0.0090*XMACH)*(0.904+0.655*HORN)
      DELT1=DELTO*(3.95-5.3*HORN+3.85*HORN**2)*Y90
      DELTO=1.1*((GAMM1*XMACH**2+2.)/((GAM+1.)*XMACH**2))*(0.96+.5*HORN)
   10 CONTINUE
      DO 40 J=2,JMAX
C........BODY SHAPE
      THET=(FLOAT(J)-1.5)*DTHET
      IF (LRSTRT) THET=ATAN2(Y(J,1),1.0-X(J,1))
      CALL BODY(YB,XB,RNOSE,THET,HORN)
      XA=XB
C........SHOCK SHAPE
      IF (LRSTRT) GO TO 20
      IF (HORN .LT. 0.0) GO TO 15
      IF (HORN.LT.0.1 .AND. HORN.GT.0.0) GO TO 15
      IF (J .GT. NBLUNT) GO TO 12
      BBB=DELT1**2*(1.-SIN(THET)**2)/SIN(THET)**2/(2.+DELTO)
      FOURAC=4.*BBB*(1.+DELTO)
      YA=1.+(BBB-SQRT(BBB**2+FOURAC))*0.5
      THET90=1.570796326679
      IF((THET-THET90) .GE. 0.0)XA=1.+(BBB+SQRT(BBB**2+FOURAC))*0.5
   12 YA=DELT1*SQRT((1.+DELTO)-(1.-XA))/SQRT(1.+DELTO)
      GO TO 30
   15 CONTINUE
      IF (J .GT. NBLUNT) THET=ATAN2(YA,1.-XA)
      DELT=(1.0+0.68*THET**2+0.16*THET**4)*DELTO
      IF (J .GT. NBLUNT) GO TO 17
      XA=1.0-(1.0+DELT)*COS(THET)
   17 YA=(1.0+DELT)*SIN(THET)
      GO TO 30
   20 CONTINUE
      XA=X(J,KMAX)
      YA=Y(J,KMAX)
C........CALCULATE NODAL POINTS
   30 CONTINUE
      ZKM=KMAX-1
      DX=(XA-XB)/ZKM
      DY=(YA-YB)/ZKM
      DO 40 K=1,KMAX
      ZK=K-1
      X(J,K)=XB+ZK*DX
      Y(J,K)=YB+ZK*DY
   40 CONTINUE
      DO 50 K=1,KMAX
      X(1,K)=X(2,K)
      Y(1,K)=-Y(2,K)
   50 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE OUTPUT(L)
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,X,Y,XET,XEX,XEY,D
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),XEX(40,20,2),
     *  XEY(40,20,2),D(40,20)
      LEVEL 2,Q,EF,S,G,AB
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)
      DIMENSION SL(40),CON(8)
      GO TO (1,2,3,4,5),L
    1 CONTINUE
C...OUTPUT FLOWFIELD DATA
      RMS=0.0
      PERRMX=0.0
      HTINF=GAM/GAMM1*PINF/RINF+QINF**2*0.5
      WRITE(6,111)
      DO 10 K=1,KMAX
      DO 6 J=1,JMAX
      EN=Q(J,K,4)*D(J,K)
      RHO=Q(J,K,1)*D(J,K)
      U=Q(J,K,2)/Q(J,K,1)
      V=Q(J,K,3)/Q(J,K,1)
      P=GAMM1*(EN-RHO*0.5*(U*U+V*V))
      HT=GAM/GAMM1*P/RHO+0.5*(U*U+V*V)
      SS=SQRT(GAM*P/RHO)
      PERR=ABS(HT-HTINF)*100.0/HTINF
      IF(PERR.GT.PERRMX) PERRMX=PERR
      RMS=RMS+PERR**2
      SL(J)=SQRT(U*U+V*V)/SS
    6 CONTINUE
      DO 11 J=3,JMAX
      IF((SL(J).LE.1.0.AND.SL(J-1).GE.1.0).OR.(SL(J).GE.1.0.AND.SL(J-1)
     <.LE.1.0))GO TO 12
      GO TO 11
   12 JSL=J
      JSLM=JSL-1
      COEF=(1.0-SL(JSLM))/(SL(JSL)-SL(JSLM))
      XSL=X(JSLM,K)+COEF*(X(JSL,K)-X(JSLM,K))
      YSL=Y(JSLM,K)+COEF*(Y(JSL,K)-Y(JSLM,K))
      XSL=1.0-XSL
      WRITE(6,110) XSL,YSL
   11 CONTINUE
   10 CONTINUE
      RMS=SQRT(RMS/FLOAT(JMAX*KMAX))
      WRITE(6,107) PERRMX,RMS
      RETURN
    2 CONTINUE
C...OUTPUT E AND F CONSERVATIVE VARIABLES
      WRITE(6,133)
      DO 7 K=1,KMAX
      WRITE(6,134) K
      DO 7 J=1,JMAX
      CALL EFCON(J,K,1)
      DO 8 N=1,4
    8 CON(N)=G(N)
      CALL EFCON(J,K,2)
      DO 9 N=1,4
      NN=N+4
    9 CON(NN)=G(N)
    7 WRITE(6,135) J,(CON(N),N=1,8)
      RETURN
    3 CONTINUE
      RETURN
    4 CONTINUE
      WRITE(6,139)
      WRITE(6,109) ((J,K,X(J,K),Y(J,K),(XET(J,K,I),XEX(J,K,I),XEY(J,K,I)
     > ,I=1,2),D(J,K),K=1,KMAX),J=1,JMAX)
      RETURN
    5 CONTINUE
```

```
      RETURN
  103 FORMAT(1H0,37X,32H<<<< CONSERVATIVE VARIABLES >>>>)
  104 FORMAT(3H0K=,I2//4X,1HJ,6X,2HE1,10X,2HE2,10X,2HE3,10X,2HE4,10X,
     > 2HF1,10X,2HF2,10X,2HF3,10X,2HF4/)
  105 FORMAT(I5,8E12.4)
  107 FORMAT(21H0<<<< % ERROR IN HT =,E12.4,3X,22HRMS OF % ERROR IN HT =
     > ,E12.4,5H >>>>)
  108 FORMAT(1H1,3X,1HJ,4X,1HK,8X,1HX,11X,1HY,10X,4HXI-T,8X,4HXI-X,8X,
     > 4HXI-Y,7X,5HETA-T,7X,5HETA-X,7X,5HETA-Y,8X,1HJ/)
  109 FORMAT(2I5,9F12.6)
  110 FORMAT(5H,YSL=,F7.4,3X,4HRSL=,F7.4)
  111 FORMAT(//26H FINAL SONIC LINE LOCATION/)
      END
```

```
      SUBROUTINE RHS
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,Q,EF,S,G,AB
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)
C...THIS SUBROUTINE COMPUTES THE RIGHT HAND SIDE OF THE DELTA FORM
C...EQUATION
C...FORM E CONSERVATIVE VARIABLES AND DIFFERENCE. STORE IN THE S ARRAY
      DO 1 K=2,KM
      DO 2 J=1,JMAX
      CALL EFCON(J,K,1)
      DO 2 N=1,4
    2 EF(J,N)=G(N)
C...CENTRAL DIFFERENCE E CONSERVATIVE VARIABLE
      DO 1 N=1,4
      DO 1 J=2,JM
    1 S(J,K,N)=(EF(J+1,N)-EF(J-1,N))*H
C...FORM F CONSERVATIVE VARIABLES AND DIFFERENCE. ADD TO PREVIOUS S
C...ARRAY
      DO 3 J=2,JM
      DO 4 K=1,KMAX
      CALL EFCON(J,K,2)
      DO 4 N=1,4
    4 EF(K,N)=G(N)
C...CENTRAL DIFFERENCE F CONSERVATIVE VARIABLE
      DO 3 N=1,4
      DO 3 K=2,KM
      S(J,K,N)=-S(J,K,N)-(EF(K+1,N)-EF(K-1,N))*H
    3 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SHOCK
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT,
     > CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,
     <RINF,QINF,CINF,JCS,TM,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH
      LEVEL 2,X,Y,XET,XEX,XEY,D
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),XEX(40,20,2),
     *  XEY(40,20,2),D(40,20)
      LEVEL 2,Q,EF,S,G,AB
      COMMON /COM3/ Q(40,20,4),EF(40,4),S(40,20,4),G(4),AB(4,4)
      COMMON /PUV/ P(40,3),PXI(40),PETA(40),U(40,3),UXI(40),UETA(40),
     *  V(40,3),VXI(40),VETA(40),R(40,3)
C...COMPUTE THE FLOW VARIABLES ONE MESH INTERVAL BELOW SHOCK
      RMS=0.0
      QSEM=0.0
      JMM=JMAX-2
      ZKM=KMAX-1
      DO 3 K=1,3
      DO 3 J=1,JMAX
      KK=KMAX-3+K
      Z=1.0/Q(J,KK,1)
      R(J,K)=Q(J,KK,1)*D(J,KK)
      U(J,K)=Q(J,KK,2)*Z
      V(J,K)=Q(J,KK,3)*Z
      E2=Q(J,KK,4)*D(J,KK)
    3 P(J,K)=(E2-0.5*R(J,K)*(U(J,K)**2+V(J,K)**2))*GAMM1
C...COMPUTE P-XI, U-XI, P-ETA, U-ETA, AND V-ETA DERIVATIVES
      DO 4 J=2,JM
      PXI(J)=(P(J+1,3)-P(J-1,3))*0.5
      UXI(J)=(U(J+1,3)-U(J-1,3))*0.5
    4 VXI(J)=(V(J+1,3)-V(J-1,3))*0.5
      PXI(1)=-PXI(2)
      UXI(1)=-UXI(2)
      VXI(1)=VXI(2)
      PXI(JMAX)=(3.0*P(JMAX,3)-4.0*P(JM,3)+P(JMM,3))*0.5
      UXI(JMAX)=(3.0*U(JMAX,3)-4.0*U(JM,3)+U(JMM,3))*0.5
      VXI(JMAX)=(3.0*V(JMAX,3)-4.0*V(JM,3)+V(JMM,3))*0.5
      DO 5 J=1,JMAX
      PETA(J)=(3.0*P(J,3)-4.0*P(J,2)+P(J,1))*0.5
      UETA(J)=(3.0*U(J,3)-4.0*U(J,2)+U(J,1))*0.5
      VETA(J)=(3.0*V(J,3)-4.0*V(J,2)+V(J,1))*0.5
      IF(J.EQ.1.OR.J.EQ.JMAX) GO TO 5
      P(J,2)=P(J+1,3)+P(J-1,3)-2.0*P(J,3)
    5 CONTINUE
      P(1,2)=P(2,2)
      P(JMAX,2)=0.0
      DO 1 J=1,JMAX
      K=KMAX
C...DETERMINE SHOCK ANGLE DELTA=ARCTAN(-ETAY/ETAX) J,KMAX
      DELTA=ATAN(-XEY(J,K,2)/XEX(J,K,2))
      SD=SIN(DELTA)
      CD=COS(DELTA)
      U1T=QINF*CD
      UBAR=XET(J,K,1)+U(J,3)*XEX(J,K,1)+V(J,3)*XEY(J,K,1)
      VBAR=XET(J,K,2)+U(J,3)*XEX(J,K,2)+V(J,3)*XEY(J,K,2)
      RCS=GAM*P(J,3)
      PTAU=-UBAR*PXI(J)-VBAR*PETA(J)-RCS*(UXI(J)*XEX(J,K,1)+
     > VXI(J)*XEY(J,K,1)+UETA(J)*XEX(J,K,2)+VETA(J)*XEY(J,K,2)+
     > (V(J,3)/Y(J,K))*FLOAT(JCS))
      P2=P(J,3)+PTAU*DT*0.2+0.10*P(J,2)
      IF(J .EQ. JMAX)P2=2.*P(JM,3)-P(JM-1,3)
      IF(P2.LE.0.0) GO TO 6
      Z=GAM+1.0
      XMX=SQRT(0.5/GAM*(P2/PINF*Z+GAMM1))
      QS=CINF*XMX-U1T
      PA=P(J,3)
      RA=R(J,3)
      UA=U(J,3)
      VA=V(J,3)
      EA=PA/GAMM1+0.5*RA*(UA**2+VA**2)
      U2T=2.0*(1.0-XMX**2)*CINF/((GAM+1.0)*XMX)+U1T
      R2=RINF*(P2/PINF+GAMM1/Z)/(1.0+GAMM1/Z*P2/PINF)
      U2=QINF*SD**2+U2T*CD
      V2=QINF*SD*CD-U2T*SD
      E2=P2/GAMM1+0.5*R2*(U2**2+V2**2)
C...COMPUTE CONSERVATIVE VARIABLES AT SHOCK
      K=KMAX
      DI=1.0/D(J,K)
      Q(J,K,1)=R2*DI
      Q(J,K,2)=R2*U2*DI
      Q(J,K,3)=R2*V2*DI
      Q(J,K,4)=E2*DI
C...DETERMINE ANGLE OF XI=CONST LINE WITH X-AXIS
      K=KMAX
      IF(ABS(XEY(J,K,1))-0.000001) 7,7,8
    7 THETA=1.5707963
      GO TO 9
    8 CONTINUE
      THETA=ATAN(XEX(J,K,1)/XEY(J,K,1))
    9 CONTINUE
C...COMPUTE SHOCK SPEED IN X AND Y DIRECTIONS
      BETA=THETA-DELTA
      QSE=QS/COS(BETA)
      IF(ABS(QSE) .GE. ABS(QSEM))JQS=J
      IF(ABS(QSE) .GE. ABS(QSEM))QSEM=QSE
      RMS=RMS+QSE**2
      XST=-QSE*COS(THETA)
      YST=QSE*SIN(THETA)
      THETA=THETA*57.29578
      DELTA=DELTA*57.29578
      BETA=BETA*57.29578
C...PROPAGATE SHOCK
      X(J,K)=X(J,K)+XST*DT
      Y(J,K)=Y(J,K)+YST*DT
C...ADJUST OTHER GRID POINTS
      DO 2 K=2,KM
      ZKFAC=FLOAT(K-1)/ZKM
      X(J,K)=(X(J,KMAX)-X(J,1))*ZKFAC+X(J,1)
      Y(J,K)=(Y(J,KMAX)-Y(J,1))*ZKFAC+Y(J,1)
    2 CONTINUE
    1 CONTINUE
      RMS=SQRT(RMS/FLOAT(JMAX))
      WRITE(6,102) IT,RMS,QSEM,JQS
      RETURN
```

```
    5 CONTINUE                                                          SHOCK    109
      WRITE(6,103) J,P2,P(J,3),PTAU                                     SHOCK    110
      STOP                                                              SHOCK    111
  102 FORMAT(10H ITERATION,I4,4X,19HRMS OF SHOCK SPEED=,1PE11.4,3X,     SHOCK    112
     * 20HMAXIMUM SHOCK SPEED=,E11.4,6H AT J=,I2)                       SHOCK    113
  103 FORMAT(1H0,41HNEGATIVE PRESSURE DETECTED BY SHOCK AT J=,I2/       SHOCK    114
     *   3X,3HP=,1PE10.3,3X,3HPD=,3X,E10.3,3X,5HPTAU=,E10.3)            SHOCK    115
      END                                                               SHOCK    116


      SUBROUTINE XIETAD (IFLAG)                                         XIETAD     2
      COMMON/COM1/JMAX,KMAX,JM,KM,XMACH,ALPHA,GAM,GAMM1,CN,DT,SMU,IPRT, COM1       2
     * CHORD,NCA,NCB,NCC,AA,H,OMEGA,NU,NL,IT,TAU,ITER,ENT,PTORT,PINF,   COM1       3
     *RINF,QINF,CINF,JCS,TH,CLUS,PT,HORN,RNOSE,NCASE,NPUNCH             COM1       4
      LEVEL 2,X,Y,XET,XEX,XEY,D                                         COM2       2
      COMMON /COM2/ X(40,20),Y(40,20),XET(40,20,2),XEX(40,20,2),        COM2       3
     *  XEY(40,20,2),D(40,20)                                           COM2       4
      LEVEL 2,Q,EF,S,G,AB                                               COM3       2
      COMMON /COM3/ Q(40,20,4),EF(4,41),S(40,20,4),G(4),AB(4,4)         COM3       3
      DO 11 K=1,KMAX                                                    XIETAD     6
      DO 11 J=1,JMAX                                                    XIETAD     7
      XET(J,K,1)=0.0                                                    XIETAD     8
      XET(J,K,2)=0.0                                                    XIETAD     9
   11 CONTINUE                                                          XIETAD    10
      JMM=JM-1                                                          XIETAD    11
      KMM=KM-1                                                          XIETAD    12
C...COMPUTE X-XI AND Y-XI; DXI AND DETA = 1                              XIETAD    13
      DO 1 K=1,KMAX                                                     XIETAD    14
      DO 2 J=2,JM                                                       XIETAD    15
      XEY(J,K,2)=(X(J+1,K)-X(J-1,K))*0.5                                XIETAD    16
    2 XEX(J,K,2)=(Y(J+1,K)-Y(J-1,K))*0.5                                XIETAD    17
      XEY(1,K,2)=(-3.0*X(1,K)+4.0*X(2,K)-X(3,K))*0.5                    XIETAD    18
      XEY(JMAX,K,2)=(3.0*X(JMAX,K)-4.0*X(JM,K)+X(JMM,K))*0.5            XIETAD    19
      XEX(1,K,2)=(-3.0*Y(1,K)+4.0*Y(2,K)-Y(3,K))*0.5                    XIETAD    20
    1 XEX(JMAX,K,2)=(3.0*Y(JMAX,K)-4.0*Y(JM,K)+Y(JMM,K))*0.5            XIETAD    21
C...COMPUTE X-ETA AND Y-ETA                                              XIETAD    22
      DO 3 J=1,JMAX                                                     XIETAD    23
      DO 4 K=2,KM                                                       XIETAD    24
      XEY(J,K,1)=(X(J,K+1)-X(J,K-1))*0.5                                XIETAD    25
    4 XEX(J,K,1)=(Y(J,K+1)-Y(J,K-1))*0.5                                XIETAD    26
      XEY(J,1,1)=(-3.0*X(J,1)+4.0*X(J,2)-X(J,3))*0.5                    XIETAD    27
      XEY(J,KMAX,1)=(3.0*X(J,KMAX)-4.0*X(J,KM)+X(J,KMM))*0.5            XIETAD    28
      XEX(J,1,1)=(-3.0*Y(J,1)+4.0*Y(J,2)-Y(J,3))*0.5                    XIETAD    29
    3 XEX(J,KMAX,1)=(3.0*Y(J,KMAX)-4.0*Y(J,KM)+Y(J,KMM))*0.5            XIETAD    30
C...COMPUTE XI-X, XI-Y, ETA-X, AND ETA-Y                                 XIETAD    31
      DO 5 K=1,KMAX                                                     XIETAD    32
      DO 5 J=1,JMAX                                                     XIETAD    33
      DT=1.0/(XEX(J,K,1)*XEY(J,K,2)-XEY(J,K,1)*XEX(J,K,2))              XIETAD    34
      DTI=DT                                                            XIETAD    35
      IF(IFLAG.EQ.0) GO TO 7                                            XIETAD    36
C...ADJUST CONSERVATIVE VARIABLES BASED ON NEW MESH                      XIETAD    37
      DO 6 N=1,4                                                        XIETAD    38
    6 Q(J,K,N)=Q(J,K,N)*D(J,K)/DTI                                      XIETAD    39
    7 CONTINUE                                                          XIETAD    40
C...THE GEOMETRIC JACOBIAN IS DEFINED HERE AND STORED IN THE D ARRAY     XIETAD    41
      D(J,K)=DTI                                                        XIETAD    42
      XEY(J,K,1)=XEX(J,K,1)*DT                                          XIETAD    43
      XEY(J,K,1)=-XEY(J,K,1)*DT                                         XIETAD    44
      XEX(J,K,2)=-XEX(J,K,2)*DT                                         XIETAD    45
    5 XEY(J,K,2)=XEY(J,K,2)*DT                                          XIETAD    46
C...REFLECT METRICS AND DEPENDENT VARIABLES ABOUT PLANE OF SYMMETRY      XIETAD    47
      IF(IFLAG.EQ.0) GO TO 8                                            XIETAD    48
      DO 9 K=1,KMAX                                                     XIETAD    49
      DD=(+1.0)**JCS                                                    XIETAD    50
      D(1,K)=D(2,K)*DD                                                  XIETAD    51
      XEX(1,K,1)=-XEX(2,K,1)                                            XIETAD    52
      XEY(1,K,1)=XEY(2,K,1)                                             XIETAD    53
      XEX(1,K,2)=XEX(2,K,2)                                             XIETAD    54
      XEY(1,K,2)=-XEY(2,K,2)                                            XIETAD    55
      DO 10 N=1,4                                                       XIETAD    56
   10 Q(1,K,N)=Q(2,K,N)*DD                                              XIETAD    57
    9 Q(1,K,3)=-Q(2,K,3)*DD                                             XIETAD    58
    8 CONTINUE                                                          XIETAD    59
      RETURN                                                            XIETAD    60
      END                                                               XIETAD    61


      SUBROUTINE MARCH                                                  MARCH      2
      COMMON /IDVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24) IDVARB     2
      COMMON/JOE/ZL1,CF1,CF2,ZLF,ZTRAN,DZTRAN                           JOE        2
      LEVEL 2, RHO,P,U,V,W,ROR,ROBZ,VINF,WINF,RORPH,RB,RBZ,RBPH,DTDPH,  PVARB      2
     *  RCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT, PVARB     3
     *  RSPHIT                                                          PVARB      4
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41), PVARB     5
     *        ROR(41)     , ROBZ(41) , VINF(41) , WINF(41) ,            PVARB      6
     *        RORPH(41)   ,              RB(41)    , RBZ(41)   , RBPH(41) , PVARB  7
     *       DTDPH(24,41), RCT(41)  ,DTDZ(24,41),DTDR(41) , ACT(41)  ,  PVARB      8
     *        ICONST(50) , GAM(20)  , CONST(50) ,NREGON    , RS(41)  ,  PVARB      9
     *        RSZ(41)     , RSPHI(41), RST(41)    , RSZT(41), RSPHIT(41) PVARB    10
      COMMON/SVARB/T,Z      , PHI    , DT     , DZ     , DPHI   , ZINT   , SVARB  2
     *        ZEND   , PT     , ALPHA  , GAMMA  , SIGMA  , XMACH  , TAPE1  , SVARB  3
     *        TAPE2  , DISK1  , ALPH   , DISK2  , SIGM   , NPRNT  , DZDT   , SVARB  4
     *        DZDPH  , ZM     , TMWD   , TMLD   , TMW    , TML    , TTMW   , SVARB  5
     *        TTML   , RZ     , PZ     , NIPHI  , NIT    , KPHI   , NITER  , SVARB  6
     *        NPHI   , NPHI1  , NPHI2  , NPHI3  , NPHM1  , NPHM2  , NPHM3  , SVARB  7
     *        NT     , NT1    , NT2    , NT3    , PHIFD  , NCONE  , RADI   , SVARB  8
     *        PHIF   , METHOD, LAG     , NBC    , PINF   , RHOIN  , UINF   , SVARB  9
     *QINF,GASCON,NREAL,NPUNCH                                          SVARB     10
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NXPLOT                          DNSTRM     2
C                                                                       MARCH      8
C                                                                       MARCH      9
      CALL SETDAT                                                       MARCH     10
      CALL GEOM3(O,PHIP,NPHI,Z,RB,RBZ,RBPH,IPRNT)                       MARCH     11
      SIGMA=ATAN(CF2)*57.29578                                          MARCH     12
      ICONST(48)=0                                                      MARCH     13
      CALL INITA                                                        MARCH     14
      IF (Z .GT. ZEND) GO TO 19                                         MARCH     15
      NITER=100-NZEND                                                   MARCH     16
      WRITE(6,610)                                                      MARCH     17
      STPSZE=NT-1                                                       MARCH     18
      CALL BNDRYM(2)                                                    MARCH     19
      DO 4 JUDI=1,NITER                                                 MARCH     20
      ICONST(5)=JUDI                                                    MARCH     21
C.........COMPUTE AUTOMATIC STEPSIZE                                    MARCH     22
      IF (JUDI .EQ. 1) GO TO 3                                          MARCH     23
      IF (MOD(JUDI,ICONST(49)).NE.0) GO TO 5                            MARCH     24
    3 CALL EIGENM                                                       MARCH     25
      IF (DZ .LT. 1.0) GO TO 5                                          MARCH     26
      DZDT=STPSZE                                                       MARCH     27
      DZ=DZDT*DT                                                        MARCH     28
      DZDPH=DZ/DETA                                                     MARCH     29
5     CONTINUE                                                          MARCH     30
      CALL DIFFR                                                        MARCH     31
C.........WRITE DATA FOR RERUN ON TAPE9                                 MARCH     32
C         AND STORE IN ARRAYS USED BY CONTOUR ROUTINES                  MARCH     33
      CALL OUTPTM                                                       MARCH     34
      IF (Z .GT. ZEND) GO TO 19                                         MARCH     35
4     CONTINUE                                                          MARCH     36
   19 CONTINUE                                                          MARCH     37
C.........ALL DATA REQUIRED FOR RERUN OF THIS CASE IS NOW ON TAPE9      MARCH     38
      END FILE 9                                                        MARCH     39
      RETURN                                                            MARCH     40
C                                                                       MARCH     41
  610 FORMAT(1H1//54X,20HMARCHING CALCULATION/54X,20(1H*)///            MARCH     42
     *     2X,8HSTEP NO.,4X,19HDOWNSTREAM LOCATION,4X,13HBODY ORDINATE, MARCH     43
     *     6X,14HSHOCK ORDINATE/)                                       MARCH     44
      END                                                               MARCH     45


      SUBROUTINE BNDRYM(K2)                                             BNDRYM     2
      COMMON/ENTRO/S(41),ZBS,ZFLD,ITPRTR,ITPRTF,NCASE,NTDSDS            ENTRO      2
      LEVEL 2, RHO,P,U,V,W,ROR,ROBZ,VINF,WINF,RORPH,RB,RBZ,RBPH,DTDPH,  PVARB      2
     *  RCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT, PVARB     3
     *  RSPHIT                                                          PVARB      4
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41), PVARB     5
     *        ROR(41)     , ROBZ(41) , VINF(41) , WINF(41) ,            PVARB      6
     *        RORPH(41)   ,              RB(41)    , RBZ(41)   , RBPH(41) , PVARB  7
     *       DTDPH(24,41), RCT(41)  ,DTDZ(24,41),DTDR(41) , ACT(41)  ,  PVARB      8
     *        ICONST(50) , GAM(20)  , CONST(50) ,NREGON    , RS(41)  ,  PVARB      9
     *        RSZ(41)     , RSPHI(41), RST(41)    , RSZT(41), RSPHIT(41) PVARB    10
      COMMON/SVARB/T,Z      , PHI    , DT     , DZ     , DPHI   , ZINT   , SVARB  2
     *        ZEND   , PT     , ALPHA  , GAMMA  , SIGMA  , XMACH  , TAPE1  , SVARB  3
     *        TAPE2  , DISK1  , ALPH   , DISK2  , SIGM   , NPRNT  , DZDT   , SVARB  4
     *        DZDPH  , ZM     , TMWD   , TMLD   , TMW    , TML    , TTMW   , SVARB  5
     *        TTML   , RZ     , PZ     , NIPHI  , NIT    , KPHI   , NITER  , SVARB  6
     *        NPHI   , NPHI1  , NPHI2  , NPHI3  , NPHM1  , NPHM2  , NPHM3  , SVARB  7
     *        NT     , NT1    , NT2    , NT3    , PHIFD  , NCONE  , RADI   , SVARB  8
     *        PHIF   , METHOD, LAG     , NBC    , PINF   , RHOIN  , UINF   , SVARB  9
     *QINF,GASCON,NREAL,NPUNCH                                          SVARB     10
```

```
      DIMENSION PK13(41),PK14(41),PK21(41),PK22(41),PK23(41)
      LOGICAL IT2ND
      ARSIN(A)=ASIN(A)
      GO TO (10,18,11),KI
10    CONTINUE
C..WEAK OR SMALL ANGLE CORRECTIONS (USES PRANDTL-MEYER RELATIONS)
      DO 9 K=3,NPHI
      PK4=1.0/SQRT(RBZ(K)**2+1.0+(RBPH(K)/RB(K))**2)
      PK1=-RBZ(K)*PK4
      PK2=PK4
      PK3=-RBPH(K)/RB(K)*PK4
      IT2ND=.FALSE.
      QSQ=U(3,K)**2+V(3,K)**2+W(3,K)**2
      IF(P(3,K).GE.0.0) GO TO 4
C..NEGATIVE SURFACE PRESSURE
      ICHECK=1
      X=1.0-7
      IF (K .EQ. 3) WRITE(6,100) X,P(3,K),RHO(3,K),U(3,K),V(3,K)
      P(3,K)=PINF*(1.-0.5*GAMMA)
      RHO(3,K)=(P(3,K)/S(K))**(1.0/GAMMA)
      Q3K=SQRT(1.0-P(3,K)/RHO(3,K))
      U(3,K)=U(3,K)*Q3K/QSQ**0.5
      V(3,K)=V(3,K)*Q3K/QSQ**0.5
      W(3,K)=W(3,K)*Q3K/QSQ**0.5
      QSQ=Q3K**2
4     CONTINUE
      IF(RHO(3,K) .GE. 0.0)GO TO 5
      ICHECK=2
      X=1.0-7
      IF (K .EQ. 3) WRITE(6,100) X,P(3,K),RHO(3,K),U(3,K),V(3,K)
      RHO(3,K)=(P(3,K)/S(K))**(1.0/GAMMA)
      Q3K=SQRT(1.0-P(3,K)/RHO(3,K))
      U(3,K)=U(3,K)*Q3K/QSQ**0.5
      V(3,K)=V(3,K)*Q3K/QSQ**0.5
      W(3,K)=W(3,K)*Q3K/QSQ**0.5
5     CONTINUE
      PK5=SQRT(QSQ)
      PK6=(PK1*U(3,K)+PK2*V(3,K)+PK3*W(3,K))/PK5
      PK7=ARSIN(PK6)
      PK8=GAM(1)*P(3,K)/RHO(3,K)
      PK9=PK5**2/PK8
      PK10=PK9-1.0
      IF(PK10 .GT. 0.0)GO TO 6
      ICHECK=3
      X=1.0-7
      IF (K .EQ. 3) WRITE(6,100) X,P(3,K),RHO(3,K),U(3,K),V(3,K)
      PK10=0.5
      PK9=1.5
      PK8=PK5**2/PK9
      RHO(3,K)=GAM(1)*P(3,K)/PK8
      Q3K=SQRT(1.0-P(3,K)/RHO(3,K))
      U(3,K)=U(3,K)*Q3K/QSQ**0.5
      V(3,K)=V(3,K)*Q3K/QSQ**0.5
      W(3,K)=W(3,K)*Q3K/QSQ**0.5
6     CONTINUE
      PK11=GAMMA*PK9/SQRT(PK10)
      PK12=GAMMA*PK9*((GAMMA+1.0)*PK9**2-4.0*PK10)/(4.0*PK10**2)
      PK13(K)=P(3,K)*(1.0-PK11*PK7+PK12*PK7**2)
      FACTOR=0.5*GAMMA*PK9/(PK10**3.5)
      TERM1=(GAMMA+1.0)*PK9**4/6.0
      TERM2=-(5.0+7.0*GAMMA-2.0*GAMMA**2)*PK9**3/6.0
      TERM3= 5.0*(GAMMA+1.0)*PK9**2/3.0
      TERM4=4.0/3.0-2.0*PK9
      COEFF3=FACTOR*(TERM1+TERM2+TERM3+TERM4)
      PTEST=PK13(K)-P(3,K)*COEFF3*PK7**3
      IF (ABS(PK7).LT.ABS(0.01))GO TO 123
      XM1 = SQRT(PK9)
      CALL PMTURN(XM1,PK7,P2P1,NITS,GAMMA)
      PTRUE = P(3,K)*P2P1
      PTEST = PTRUE
  123 CONTINUE
      PK13(K)=PTEST
      PK14(K)=(PK13(K)/S(K))**(1.0/GAMMA)
      PK15=SQRT(1.0-PK13(K)/PK14(K))
      PK16=PK6*PK5*PK4
      PK17=U(3,K)+PK16*RBZ(K)
      PK18=V(3,K)-PK16
      PK19=W(3,K)+PK16*RBPH(K)/RB(K)
      PK20=SQRT(PK17**2+PK18**2+PK19**2)
      PK24=PK15/PK20
      PK21(K)=PK24*PK17
      PK22(K)=PK24*PK18
      PK23(K)=PK24*PK19
9     CONTINUE
11    CONTINUE
C..RESET BODY VARIABLES TO THOSE CALCULATED BY ABBETTS SCHEME
      DO 12 K=3,NPHI
      P(3,K)=PK13(K)
      RHO(3,K)=PK14(K)
      U(3,K)=PK21(K)
      V(3,K)=PK22(K)
      W(3,K)=PK23(K)
12    CONTINUE
      GO TO 21
18    CONTINUE
C
C..APPLY REFLECTION PRINCIPLE AT PLANES OF SYMMETRY
C
      DO 1 K=1,2
      M=6-K
      L=NPHI+K
      N=NPHI-K
      DO 1 J=3,NT2
      RHO(J,K)=RHO(J,M)
      RHO(J,L)=RHO(J,N)
      P(J,K)=P(J,M)
      P(J,L)=P(J,N)
      U(J,K)=U(J,M)
      U(J,L)=U(J,N)
      V(J,K)=V(J,M)
      V(J,L)=V(J,N)
      W(J,K)=-W(J,M)
      W(J,L)=-W(J,N)
      W(J,3)=0.0
      W(J,NPHI)=0.0
1     CONTINUE
21    CONTINUE
  100 FORMAT(1H0,49HNEGATIVE PRESSURE OR DENSITY ON BODY DETECTED BY ,
     *   11HBNDRY AT X=,F7.3/3X,3HP=,1PE10.3,3X,5HRHO=,E10.3,3X,
     *   4HVYB=,E10.3,3X,4HVRB=,E10.3)
      RETURN
      END
```

```
      SUBROUTINE DIFFR
      LEVEL 2,ETEMP,E0,F0,G0,H0
      COMMON /CVARS/ ETEMP(4,24,41), E0(4,24,41),
     *       F0(4,24,41)    , G0(4,24,41) , H0(4,24,41)
      LEVEL 2, RHO,P,U,V,W,ROB,ROBZ,VINF,WINF,ROBPH,RB,RBZ,RBPH,DTDPH,
     *  RCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT,
     *  RSPHIT
      COMMON /PVARS/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),
     *       ROB(41)     , ROBZ(41) , VINF(41) , WINF(41) ,
     *       ROBPH(41)  ,             RB(41)    , RBZ(41)   , RBPH(41) ,
     *      DTDPH(24,41), RCT(41)   ,DTDZ(24,41),DTDR(41) , ACT(41)   ,
     *       ICONST(50) , GAM(25)   , CONST(50) ,NREGON    , RS(41)    ,
     *       RSZ(41)    , RSPHI(41), RST(41)    , RSZT(41), RSPHIT(41)
      COMMON/SVARS/T,Z       , PHI    , DT     , DZ     , DPHI   , ZINT   ,
     *       ZEND   , PI     , ALPHA  , GAMMA  , SIGMA  , XMACH  , TAPE1  ,
     *       TAPE2  , DISK1  , ALPH   , DISK2  , SIGM   , NPRNT  , DZDT   ,
     *       DZDPH  , ZM     , THWD   , THLD   , THW    , THL    , TTHW   ,
     *       TTHL   , RZ     , RZ     , NIPHI  , NIT    , KPHI   , NITER  ,
     *       NPHI   , NPHI1  , NPHI2  , NPHI3  , NPHM1  , NPHM2  , NPHM3  ,
     *       NT     , NT1    , NT2    , NT3    , PHIFD  , NCONE  , RADI   ,
     *       PHIF   , METHOD, LAG    , NRC    , PINF   , RHOIN  , UINF   ,
     *QINF,GASCON,NREAL,NPUNCH
C..FORM CONSERVATIVE VARIABLES AT ALL POINTS
      CALL TOCON(0)
      DO 1 K=3,NPHI
      DO 1 J=3,NT2
      DO 1 N=1,4
      IF(J.EQ.NT2) GO TO 2
C..PREDICTOR STEP AT BODY AND IN FIELD
      ETEMP(N,J,K)=E0(N,J,K)-(DZDT*(F0(N,J+1,K)-F0(N,J,K))
     *                       +DZDPH*(G0(N,J,K+1)-G0(N,J,K))+DZ*H0(N,J,K))
      GO TO 1
2     CONTINUE
C..PREDICTOR STEP AT SHOCK
      ETEMP(N,J,K)=E0(N,J,K)-(DZDT*(F0(N,J,K)-F0(N,J-1,K))
     *                       +DZDPH*(G0(N,J,K+1)-G0(N,J,K))+DZ*H0(N,J,K))
      GO TO 1
1     CONTINUE
      Z=Z+DZ
C..DECODE CONSERVATIVE VARIABLES
      CALL TOCON(2)
C..CALCULATE PREDICTED SHOCK VALUES
      CALL SHOCKM(1)
```

```
C..CALCULATES GEOMETRIC FACTORS BASED ON NEW BODY AND SHOCK GEOMETRY
      CALL GEOM1(1)
C..APPLIES PLANE OF SYMMETRY BOUNDARY CONDITIONS
      CALL BNDRYM(2)
C..FORM INTERMEDIATE CONSERVATIVE VARIABLES AT ALL POINTS
      CALL IOCON(1)
      DO 3 KPHI=3,NPHI
      DO 3 J=3,NT2
      DO 3 N=1,4
      K=KPHI
C..DISSIPATION FUNCTION
      DISS=0.0
      IF (CONST(4).NE.0.0 .OR. CONST(5).NE.0.0) CALL DISSPM(N,J,K,DISS)
      IF(J.EQ.3) GO TO 9
      IF(J.EQ.NT2) GO TO 5
C..CORRECTOR IN FIELD
      ETEMP(N,J,K)=0.5*(EO(N,J,K)+ETEMP(N,J,K)-(DZDT*(FO(N,J,K)
     *-FO(N,J-1,K))+DZDPH*(GO(N,J,K)-GO(N,J,K-1))+DZ*HO(N,J,K))+DISS)
      GO TO 3
5     CONTINUE
C..CORRECTOR AT SHOCK
      ETEMP(N,J,K)=0.5*(ETEMP(N,J,K)+EO(N,J,K)-(DZDT*(FO(N,J,K)
     *-FO(N,J-1,K))+DZDPH*(GO(N,J,K)-GO(N,J,K-1))+DZ*HO(N,J,K))+DISS)
      GO TO 3
9     CONTINUE
C..CORRECTOR AT BODY
      ETEMP(N,J,K)=0.5*(ETEMP(N,J,K)+EO(N,J,K)-(DZDT*(FO(N,4,K)-FO(N,3,K
     *)
     *)+DZDPH*(GO(N,3,K)-GO(N,3,K-1))+DZ*HO(N,3,K))+DISS)
3     CONTINUE
C..DECODE CONSERVATIVE VARIABLES
      CALL IOCON(2)
C..CALCULATE CORRECTED SHOCK VALUES
      CALL SHOCKM(2)
C..CALCULATES GEOMETRIC FACTORS BASED ON OLD BODY AND NEW SHOCK GEOMETRY
      CALL GEOM1(2)
C..RESETS BODY VARIABLES
      CALL BNDRYM(1)
C..APPLIES PLANE OF SYMMETRY BOUNDARY CONDITIONS
      CALL BNDRYM(2)
      RETURN
      END
```

```
      SUBROUTINE DISSPM(N,J,K,DISS)
      LEVEL 2,ETEMP,EO,FO,GO,HO
      COMMON /CVARB/ ETEMP(4,24,41), EO(4,24,41),
     *        FO(4,24,41)     , GO(4,24,41) , HO(4,24,41)
      COMMON /IOVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24)
      LEVEL 2, RHO,P,U,V,W,ROR,ROBZ,VINF,WINF,RORPH,RB,RBZ,RBPH,DTDPH,
     *  BCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT,
     *  RSPHIT
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),
     *        ROR(41)     , ROBZ(41) , VINF(41) , WINF(41) ,
     *        RORPH(41)  ,            RB(41)     , RBZ(41)    , RBPH(41) ,
     *       DTDPH(24,41), BCT(41)   ,DTDZ(24,41),DTDR(41) , ACT(41)   ,
     *        ICONST(50) , GAM(20)   , CONST(50) ,NREGON    , RS(41)    ,
     *        RSZ(41)    , RSPHI(41), RST(41)    , RSZT(41), RSPHIT(41)
      COMMON/SVARB/T,Z      , PHI   , DT    , DZ    , DPHI  , ZINT  ,
     *        ZEND  , PI    , ALPHA , GAMMA , SIGMA , XMACH , TAPE1 ,
     *        TAPE2 , DISK1 , ALPH  , DISK2 , SIGM  , NPRNT , DZDT  ,
     *        DZDPH , ZM    , TMWD  , TMLD  , TMW   , TML   , TTMW  ,
     *        TTML  , RZ    , RZ    , NIPHI , NIT   , KPHI  , NITER ,
     *        NPHI  , NPHI1 , NPHI2 , NPHI3 , NPHM1 , NPHM2 , NPHM3 ,
     *        NT    , NT1   , NT2   , NT3   , PHIFD , NCONE , RADI  ,
     *        PHIF  , METHOD, LAG   , NBC   , PINF  , RHOIN , UINF  ,
     *OINF,GASCON,NREAL,NPUNCH
C
C
C......   CONST(4)<0 , LAX DAMPING
C......   CONST(4)=0 , NO DAMPING
C......   CONST(4)>0 , 4TH ORDER DAMPING
C
C
      IF(CONST(4))21,1,20
C...DISSIPATION TERM IN THE RADIAL DIRECTION
 20   IF(J .GE. 5 .AND. J .LE. NT)GO TO 5
      IF(J.LT.5) GO TO 7
      JD=NT
      GO TO 6
    7 JD=5
      GO TO 6
    5 JD=J
    6 DISSP=-CONST(4)*0.01*(EO(N,JD+2,K)+EO(N,JD-2,K)-4.0*(EO(N,JD+1,K)
     * +EO(N,JD-1,K))+6.0*EO(N,JD,K))
      GO TO 2
 21   CONTINUE
      IF(J .GE. 4 .AND. J .LE. NT1)GO TO 50
      IF(J.LT. 4)GO TO 70
      JD=NT
      GO TO 60
 70   JD=4
      GO TO 60
 50   JD=J
 60   DISSP=-CONST(4)*(.125*(EO(N,JD+1,K)+EO(N,JD-1,K))-0.25*EO(N,JD,K))
      GO TO 2
    1 DISSP=0.0
C
C
C......   CONST(5)<0 , LAX DAMPING
C......   CONST(5)=0 , NO DAMPING
C......   CONST(5)>0 , 4TH ORDER DAMPING
C
C
    2 IF(CONST(5))31,3,11
C...DISSIPATION TERM IN THE MERIDIANOL DIRECTION
 31   CONTINUE
      IF(K .GE. 4 .AND. K .LE. NPHM1)GO TO 80
      IF(K .LT. 4)GO TO 100
      KD=NPHM1
      GO TO 90
 100  KD=4
      GO TO 90
 80   KD=K
 90   DISSP=-CONST(5)*(.125*(EO(N,J,KD+1)+EO(N,J,KD-1))-0.25*EO(N,J,KD))
      GO TO 4
   11 CONTINUE
      IF(K.EQ.3) P(J,1)=P(J,5)
      IF(K.EQ.NPHI) P(J,NPHI2)=P(J,NPHM2)
      PF1=ABS(P(J,K+2)-2.0*P(J,K+1)+P(J,K))/(P(J,K+2)+2.0*P(J,K+1)+P(J,K
     *))
      PF2=ABS(P(J,K+1)-2.0*P(J,K)+P(J,K-1))/(P(J,K+1)+2.0*P(J,K)+P(J,K-1
     *))
      PF3=ABS(P(J,K)-2.0*P(J,K-1)+P(J,K-2))/(P(J,K)+2.0*P(J,K-1)+P(J,K-2
     *))
      DISSP=0.5/DETA*((PF1+PF2)*(EO(N,J,K+1)-EO(N,J,K))-
     * (PF2+PF3)*(EO(N,J,K)-EO(N,J,K-1)))*CONST(5)
      GO TO 4
    3 DISSP=0.0
    4 DISS=DISS+DISSP
      RETURN
      END
```

```
      SUBROUTINE EIGENM
      COMMON/CLUSTR/RJ,XI(24),TXI(24),TXIT(24)
      COMMON /IOVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24)
      LEVEL 2, RHO,P,U,V,W,ROR,ROBZ,VINF,WINF,RORPH,RB,RBZ,RBPH,DTDPH,
     *  BCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT,
     *  RSPHIT
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),
     *        ROR(41)     , ROBZ(41) , VINF(41) , WINF(41) ,
     *        RORPH(41)  ,            RB(41)     , RBZ(41)    , RBPH(41) ,
     *       DTDPH(24,41), BCT(41)   ,DTDZ(24,41),DTDR(41) , ACT(41)   ,
     *        ICONST(50) , GAM(20)   , CONST(50) ,NREGON    , RS(41)    ,
     *        RSZ(41)    , RSPHI(41), RST(41)    , RSZT(41), RSPHIT(41)
      COMMON/SVARB/T,Z      , PHI   , DT    , DZ    , DPHI  , ZINT  ,
     *        ZEND  , PI    , ALPHA , GAMMA , SIGMA , XMACH , TAPE1 ,
     *        TAPE2 , DISK1 , ALPH  , DISK2 , SIGM  , NPRNT , DZDT  ,
     *        DZDPH , ZM    , TMWD  , TMLD  , TMW   , TML   , TTMW  ,
     *        TTML  , RZ    , RZ    , NIPHI , NIT   , KPHI  , NITER ,
     *        NPHI  , NPHI1 , NPHI2 , NPHI3 , NPHM1 , NPHM2 , NPHM3 ,
     *        NT    , NT1   , NT2   , NT3   , PHIFD , NCONE , RADI  ,
     *        PHIF  , METHOD, LAG   , NBC   , PINF  , RHOIN , UINF  ,
     *OINF,GASCON,NREAL,NPUNCH
C.........K4=1 IMPLYS EIGENVALUES AND STEPSIZE
      IPRNT=ICONST(4)
      SIG12M=0.0
      SIG34M=0.0
      DO 1 K=3,NPHI
      DO 1 J=3,NT2
      T=XI(J)
      R=T*(ROR(K)-RB(K))+RB(K)
```

```
      C2=GAM(1)*P(J,K)/RHO(J,K)
      IF(C2) 17,17,18
17    CONTINUE
      C2=-C2
18    CONTINUE
      C=SQRT(C2)
      BP=DTD*H(J,K)*(ROB(K)-RP(K))/R
      Q1=(V(J,K)+W(J,K)*BP)*U(J,K)
      GQQ1=U(J,K)**2*(1.0+BP**2)+(V(J,K)+W(J,K)*BP)**2-C2*(1.0+BP**2)
      IF(GQQ1) 19,19,20
19    CONTINUE
      GQQ1=-GQQ1
      I=J-2
      IF (K .EQ. 3) WRITE(6,103) I
20    CONTINUE
      Q2=C*SQRT(GQQ1)
      Q3=U(J,K)**2-C2
      SIGB1=(Q1+Q2)/Q3
      SIGB2=(Q1-Q2)/Q3
      GQQ2=U(J,K)**2+W(J,K)**2-C2
      IF(GQQ2) 21,21,22
21    CONTINUE
      GQQ2=-GQQ2
      I=J-2
      IF (K .EQ. 3) WRITE(6,104) I
22    CONTINUE
      Q4= U(J,K)*W(J,K)
      Q5=C*SQRT(GQQ2)
      SIGB3=(Q4+Q5)/Q3/R*DTIL(K)
      SIGB4=(Q4-Q5)/Q3/R*DTIL(K)
C........COMPUTE LOCAL T AND PHI EIGENVALUES
      SIG1=ABS((DTDZ(J,K)+SIGB1*DTDR(K))*TXI(J))
      SIG2=ABS((DTDZ(J,K)+SIGB2*DTDR(K))*TXI(J))
      SIG12=AMAX1(SIG1,SIG2)
      SIG3=ABS(SIGB3)
      SIG4=ABS(SIGB4)
      SIG34=AMAX1(SIG3,SIG4)
      IF(SIG12.LE.SIG12M) GO TO 2
C........LOCATE MAXIMUM U-V EIGENVALUE
      JMAX1=J
      KMAX1=K
      SIG12M=SIG12
      ICONST(11)=JMAX1
      ICONST(12)=KMAX1
2     CONTINUE
      IF(SIG34.LE.SIG34M) GO TO 3
C........LOCATE MAXIMUM U-W EIGENVALUE
      JMAX2=J
      KMAX2=K
      ICONST(13)=JMAX2
      ICONST(14)=KMAX2
      SIG34M=SIG34
3     CONTINUE
1     CONTINUE
C........COMPUTE STEPSIZE BASED ON MAXIMUM EIGENVALUE
      DZ12=DT*CONST(9)/SIG12M
      DZ34=DETA*CONST(9)/SIG34M
      IF(DZ12.GT.DZ34) GO TO 4
      DZDT=CONST(9)/SIG12M
      DZ=DZDT*DT
      DZDPH=DZ/DETA
      ICONST(13)=100*ICONST(13)
      ICONST(14)=100*ICONST(14)
      GO TO 6
4     CONTINUE
      DZDPH=CONST(9)/SIG34M
      DZ=DZDPH*DETA
      DZDT=DZ/DT
      ICONST(11)=100*ICONST(11)
      ICONST(12)=100*ICONST(12)
6     CONTINUE
      RETURN
  103 FORMAT(1H0,41HNEGATIVE SIGMA-BAR-1 IN EIGENM INDICATES ,
     *    19HSUBSONIC FLOW AT I=,I2)
  104 FORMAT(1H0,41HNEGATIVE SIGMA-BAR-2 IN EIGENM INDICATES ,
     *    19HSUBSONIC FLOW AT I=,I2)
      END


      SUBROUTINE GEOM(RNOSE,ANG,RZ,DRDZ,ZSTA,H,R90,NMAX)
      DIMENSION ZSTA(200),DRDZ(200),RZ(200)
      LOGICAL LGRAV
      COMMON/NUBOD/XX(100),YY(100),NBOD,LGRAV
C
C......FUNCTION DEFINITIONS
C
      E(A)=EXP(-ABS(A-RNOSE)/H)
      F(A,B)=.5*A*((SIN(2.*B)-2.*SQRT(E(A)-E(A)**2))/(E(A)-SIN(B)**2))
      G(A,B)=ABS(A*((A**6*SIN(B)*COS(B)+SQRT(A**6-1.))/(A**6*COS(B)**2-
     *           1.)))
      F2(A,B,C,D)=A*(B*C-SQRT(D*(1.-D)))/(D-B*B)
      F3(A,B)=EXP(-ABS(1.-1./A)/B)
C
C......H .NE. 0.0 ; IONOPAUSE FOR ANONMAGNETIC PLANET
C......H .EQ. 0.0 ; EQUATORIAL PLANE FOR A MAGNETIC PLANET
C
      IF(H .EQ. 0.0)GO TO 10
      IF(H.LT.0.0) GO TO 20
      IF (H .LT. 0.01) GO TO 30
C
C......THIS DETERMINES THE BODY SHAPE OF A NONMAGNETIC PLANET
C
C
      IMAX=200
      PI=3.1415926535898
      RADI=180./PI
      DELTAT=ANG/RADI/FLOAT(IMAX)
      THETA=PI/2.
      R1=R90
      IF (LGRAV) GO TO 40
C
C......PERFORM AN INTEGRATION FROM 0 TO 170 DEGREES
C
      ZSTA(1)=0.0
      RZ(1)=R90
      THETA1=THETA
      DRDZ(1)=F(R1,THETA1)/R1
      J=2
      IMAXM1=IMAX-1
      DO 5 I=1,IMAXM1
C......PREDICTOR
      R=R1+DELTAT*F(R1,THETA1)
      THETA=THETA+DELTAT
C......CORRECTOR
      R=R1+0.5*DELTAT*(F(R1,THETA1)+F(R,THETA))
      DRDTH=F(R,THETA)
      R1=R
      THETA1=THETA
      ZSTA(J)=-R*COS(THETA)
      RZ(J)=R*SIN(THETA)
      DRDZ(J)=(DRDTH*SIN(THETA)+R*COS(THETA))/
     *        (-DRDTH*COS(THETA)+R*SIN(THETA))
      J=J+1
 5    CONTINUE
      NMAX=J-1
      RETURN
C
C       USE CYLINDRICAL BODY FOR H/RO .LT. 0.01
C
   30 ZSTA(1)=1.0
      ZSTA(2)=100.0
      RZ(1)=R90
      RZ(2)=R90
      DRDZ(1)=0.0
      DRDZ(2)=0.0
      NMAX=2
      RETURN
10    CONTINUE
C
C......THIS DETERMINES THE BODY SHAPE OF A MAGNETIC PLANET
C
C
      J=1
      ANG=ANG+90.
      IMAX=350
      PI=3.1415926535898
      RADI=180./PI
      DELTAT=ANG/RADI/FLOAT(IMAX+1)
      THETA=PI/2.
      R1=RNOSE
C
C......PERFORM AN INTEGRATION FROM 90 TO 260 DEGREES
C
      DO 15 I=1,IMAX
C......PREDICTOR
```

```
      THETA1=THETA
      R=R1+DELTAT*G(R1,THETA1)
      THETA=THETA+DELTAT
C......CORRECTOR
      R=R1+0.5*DELTAT*(G(R1,THETA1)+G(R,THETA))
      DRDTH=G(R,THETA)
      R1=R
      IF (THETA .LT. PI) GO TO 15
      ZSTA(J)=-R*SIN(THETA)
      RZ(J)=-R*COS(THETA)
      DRDZ(J)=(-DRDTH*COS(THETA)+R*SIN(THETA))/
     *        (-DRDTH*SIN(THETA)-R*COS(THETA))
      J=J+1
 15   CONTINUE
      NMAX=J-1
      RETURN
C
C......BODY SHAPE TABLE SUPPLIED BY USER
C
   20 CONTINUE
      J=1
      DX2=0.0
      DR2=0.0
      NBODM=NBOD-1
      DO 25 I=1,NBODM
      DX1=DX2
      DR1=DR2
      DX2=XX(I+1)-XX(I)
      DR2=YY(I+1)-YY(I)
      IF (XX(I) .GT.0.0) GO TO 25
      ZSTA(J)=-XX(I)
      RZ(J)=YY(I)
      D1=SQRT(DR1*DR1+DX1*DX1)
      D2=SQRT(DR2*DR2+DX2*DX2)
      DRDZ(J)=-(DR1*D2/DX1+DR2*D1/DX2)/(D1+D2)
      J=J+1
   25 CONTINUE
      ZSTA(J)=-XX(NBOD)
      RZ(J)=YY(NBOD)
      DRDZ(J)=-(DR2*(2.0*D2+D1)/DX2-DR1*D2/DX1)/(D1+D2)
      NMAX=J
      RETURN
 40   CONTINUE
      ZSTA(1)=0.0
      RZ(1)=R90
      THETA1=THETA
      A1=F3(R1,H)
      DRDZ(1)=F2(R1,1.0,0.0,A1)/R9,
      J=2
      IMAXM1=IMAX-1
      DO 50  I=1,IMAXM1
      AS=SIN(THETA)
      AC=COS(THETA)
C......PREDICTOR
      A1=F3(R1,H)
      K=0
      DRX=F2(R1,AS,AC,A1)
      IF (DRX.LT.0.0) DRX=2.*H*AS*AC
      R=R1+DRX*DELTAT
      THETA=THETA+DELTAT
      AS1=SIN(THETA)
      AC1=COS(THETA)
C......CORRECTOR
 45   CONTINUE
      A11=F3(R,H)
      DRXX=F2(R,AS1,AC1,A11)
      IF (DRXX.LT.0.0) DRXX=2.*H*AS1*AC1
      SLOPE=0.5*(DRX+DRXX)
      R=R1+SLOPE*DELTAT
      K=K+1
      IF (K.LT.5) GO TO 45
      ZSTA(J)=-R*AC1
      RZ(J)=R*AS1
      DRDZ(J)=(R*AC1+AS1*SLOPE)/(R*AS1-AC1*SLOPE)
      R1=R
      J=J+1
 50   CONTINUE
      NMAX=J-1
      RETURN
      END
```

```
      SUBROUTINE GEOM1(K5)
      COMMON/CLUSTR/RJ,XI(24),TXI(24),TXIT(24)
      COMMON /IDVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24)
      LEVEL 2, RHO,P,U,V,W,ROB,ROBZ,VINF,WINF,ROBPH,RB,RBZ,RBPH,DTDPH,
     *  BCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT,
     *  RSPHIT
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),
     *        ROB(41)     , ROBZ(41) , VINF(41) , WINF(41) ,
     *        ROBPH(41)  ,             RB(41)    , RBZ(41)   , RBPH(41) ,
     *       DTDPH(24,41), BCT(41)  ,DTDZ(24,41),DTDR(41) , ACT(41)  ,
     *        ICONST(50) , GAM(20)   , CONST(50) ,NREGON    , RS(41)    ,
     *        RSZ(41)     , RSPHI(41), RST(41)    , RSZT(41), RSPHIT(41)
      COMMON/SVARB/T,Z       , PHI    , DT     , DZ     , DPHI   , ZINT   ,
     *        ZEND   , PI     , ALPHA  , GAMMA  , SIGMA  , XMACH  , TAPE1  ,
     *        TAPE2  , DISK1  , ALPH   , DISK2  , SIGM   , NPRNT  , DZDT   ,
     *        DZDPH  , ZM     , TMWD   , TMLD   , TMW    , TML    , TTMW   ,
     *        TTML   , RZ     , BZ     , NIPHI  , NIT    , KPHI   , NITER  ,
     *        NPHI   , NPHI1  , NPHI2  , NPHI3  , NPHM1  , NPHM2  , NPHM3  ,
     *        NT     , NT1    , NT2    , NT3    , PHIFD  , NCONE  , RADI   ,
     *        PHIF   , METHOD, LAG    , NBC    , PINF   , RHOIN  , UINF   ,
     *OINF,GASCON,NREAL,NPUNCH
      IPRNT=ICONST(4)
      IF(K5.EQ.2) GO TO 12
      CALL GEOM3(1,PHIP,NPHI,Z,RB,RBZ,RBPH,IPRNT)
12    CONTINUE
      CALL GEOM2(K5)
      DO 1 J=3,NT2
      T=XI(J)
      DO 2 K=2,NPHI1
      PHI=PHIP(K)
      A=-RBZ(K)-T*(ROBZ(K)-RBZ(K))
      B=-RBPH(K)-T*(ROBPH(K)-RBPH(K))
      C=ROB(K)-RB(K)
      D=-(ROBZ(K)-RBZ(K))
      E=-(ROBPH(K)-RBPH(K))
      DTDZ(J,K)=A/C
      DTDPH(J,K)=B/C
      DTDR(K)=1.0/C
      ACT(K)=D/C
      BCT(K)=E/C
      R=C*T+RB(K)
      X=R*SIN(PHI)
      Y=-R*COS(PHI)
2     CONTINUE
1     CONTINUE
      RETURN
      END
```

```
      SUBROUTINE GEOM2(K3)
      COMMON /IDVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24)
      LEVEL 2, RHO,P,U,V,W,ROB,ROBZ,VINF,WINF,ROBPH,RB,RBZ,RBPH,DTDPH,
     *  BCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT,
     *  RSPHIT
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),
     *        ROB(41)     , ROBZ(41) , VINF(41) , WINF(41) ,
     *        ROBPH(41)  ,             RB(41)    , RBZ(41)   , RBPH(41) ,
     *       DTDPH(24,41), BCT(41)  ,DTDZ(24,41),DTDR(41) , ACT(41)  ,
     *        ICONST(50) , GAM(20)   , CONST(50) ,NREGON    , RS(41)    ,
     *        RSZ(41)     , RSPHI(41), RST(41)    , RSZT(41), RSPHIT(41)
      COMMON/SVARB/T,Z       , PHI    , DT     , DZ     , DPHI   , ZINT   ,
     *        ZEND   , PI     , ALPHA  , GAMMA  , SIGMA  , XMACH  , TAPE1  ,
     *        TAPE2  , DISK1  , ALPH   , DISK2  , SIGM   , NPRNT  , DZDT   ,
     *        DZDPH  , ZM     , TMWD   , TMLD   , TMW    , TML    , TTMW   ,
     *        TTML   , RZ     , BZ     , NIPHI  , NIT    , KPHI   , NITER  ,
     *        NPHI   , NPHI1  , NPHI2  , NPHI3  , NPHM1  , NPHM2  , NPHM3  ,
     *        NT     , NT1    , NT2    , NT3    , PHIFD  , NCONE  , RADI   ,
     *        PHIF   , METHOD, LAG    , NBC    , PINF   , RHOIN  , UINF   ,
     *OINF,GASCON,NREAL,NPUNCH
      DO 1 K=1,NPHI2
      PHI=PHIP(K)
      GO TO (3,2),K3
2     CONTINUE
      ROB(K)=RS(K)
      ROBZ(K)=RSZ(K)
      ROBPH(K)=RSPHI(K)
      GO TO 4
3     CONTINUE
      ROB(K)=RST(K)
      ROBZ(K)=RSZT(K)
      ROBPH(K)=RSPHIT(K)
```

```
4     CONTINUE
1     CONTINUE
      RETURN
      END


      SUBROUTINE GEOM3(K7,PHIP,NPHI,Z,RB,RBZ,RBPH,IPRNT)
      COMMON/JOE/ZL1,CF1,CF2,ZLF,ZTRAN,DZTRAN
      COMMON /TRANSF/ AMACH,GAMF,KM2,P1INF,R1INF,V1INF,P90(20,3),
     * RHO90(20,3),U90(20,3),V90(20,3),W90(20,3),RS90(3),
     * RSZ90(3),RSPH90(3),HRO,R90,XINIT
      LEVEL 2, RB,RBPH,RBZ
      DIMENSION RB(41),RBPH(41),RBZ(41),PHIP(41)
      DIMENSION ZSTA(200),DRDZ(200),RZ(200)
C
C......CONSTANTS
C
      IF(K7.NE.1) GO TO 1
      CF2=1.0
      ANG=59.5
      NST=1
      H=HRO
      RNOSE=1.
      CALL GEOM(RNOSE,ANG,RZ,DRDZ,ZSTA,H,R90,NMAX)
      NMAX1=NMAX-1
      RETURN
C
C......FIND CORRECT Z INTERVAL
C
    1 CONTINUE
      N=NST
      IF (Z.LT.ZSTA(NST)) GO TO 22
      DO 19 N=NST,NMAX1
      IF (Z.LT.ZSTA(N+1)) GO TO 20
   19 CONTINUE
      N=NMAX1
 20   CONTINUE
      NST=N
   22 CONTINUE
      RBODY=RZ(N)+(RZ(N+1)-RZ(N))/(ZSTA(N+1)-ZSTA(N))*(Z-ZSTA(N))
      RBODYZ=DRDZ(N)+(DRDZ(N+1)-DRDZ(N))/(ZSTA(N+1)-ZSTA(N))*(Z-ZSTA(N))
      RBODPH=0.0
      DO 30 K=3,NPHI
      RB(K)=RBODY
      RBPH(K)=RBODPH
      RBZ(K)=RBODYZ
   30 CONTINUE
      DO 36 K=1,2
      M=6-K
      I=NPHI+K
      N=NPHI-K
      RB(K)=RB(M)
      RB(I)=RB(N)
      RBZ(K)=RBZ(M)
      RBZ(I)=RBZ(N)
      RBPH(K)=-RBPH(M)
      RBPH(I)=-RBPH(N)
36    CONTINUE
      RETURN
      END


      SUBROUTINE INITA
      COMMON/CLUSTR/RJ,XI(24),TXI(24),TXIT(24)
      COMMON/ENTRO/S(41),ZBS,ZFLD,ITPRTB,ITPRTF,NCASE,NTOSOS
      COMMON /IDVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24)
      COMMON/JOE/ZL1,CF1,CF2,ZLF,ZTRAN,DZTRAN
      LEVEL 2, RHO,P,U,V,W,ROB,ROBZ,VINF,WINF,ROBPH,RB,RBZ,RBPH,DTDPH,
     *  BCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT,
     *  RSPHIT
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),
     *        ROB(41)     , ROBZ(41) , VINF(41) , WINF(41) ,
     *        ROBPH(41)   ,            RB(41)    , RBZ(41)   , RBPH(41) ,
     *      DTDPH(24,41), BCT(41)   ,DTDZ(24,41),DTDR(41) , ACT(41)  ,
     *        ICONST(50) , GAM(20)   , CONST(50) ,NREGON    , RS(41)   ,
     *        RSZ(41)     , RSPHI(41), RST(41)    , RSZT(41), RSPHIT(41)
      COMMON/SVARB/T,Z      , PHI    , DT     , DZ     , DPHI   , ZINT   ,
     *        ZEND   , PI     , ALPHA  , GAMMA  , SIGMA  , XMACH  , TAPE1  ,
     *        TAPE2  , DISK1  , ALPH   , DISK2  , SIGM   , NPRNT  , DZDT   ,
     *        DZDPH  , ZM     , TMWD   , TMLD   , TMW    , TML    , TTMW   ,
     *        TTML   , RZ     , BZ     , NIPHI  , NIT    , KPHI   , NITER  ,
     *        NPHI   , NPHI1  , NPHI2  , NPHI3  , NPHM1  , NPHM2  , NPHM3  ,
     *        NT     , NT1    , NT2    , NT3    , PHIFD  , NCONE  , RADI   ,
     *        PHIF   , METHOD, LAG    , NBC    , PINF   , RHOIN  , UINF   ,
     *QINF,GASCON,NREAL,NPUNCH
      COMMON /TRANSF/ AMACH,GAMF,KM2,P1INF,R1INF,V1INF,P90(20,3),
     * RHO90(20,3),U90(20,3),V90(20,3),W90(20,3),RS90(3),
     * RSZ90(3),RSPH90(3),HRO,R90,XINIT
      RADI=57.29578
      PI=3.14159265
      ZINT=XINIT
      Z=ZINT
      ZEND=ZLF
      ALPHA=0.0
      NIPHI=2
      NIT=KM2
      XMACH=AMACH
      GAMMA=GAMF
      ALPH=ALPHA/RADI
      SIGM=SIGMA/RADI
      PHIF=180.0/RADI
      NPHI=NIPHI+3
      NPHI1=NPHI+1
      NPHI2=NPHI+2
      NPHM1=NPHI-1
      NPHM2=NPHI-2
      NT=NIT+2
      NT1=NT+1
      NT2=NT+2
      ICONST(1)=1
      ICONST(10)=1
      IPRNT=ICONST(4)
      IF (ICONST(48).NE.1) ICONST(5)=0
      LAG=1
      DETA=PHIF/FLOAT(NIPHI)
      DPHI=DETA
      DT=1.0/FLOAT(NT-1)
      DZ=DZDT*DT
      DZDPH=DZ/DETA
      GAM(1)=(GAMMA-1.0)*0.5
      GAM(2)=GAM(1)/GAMMA
C..MERIDIONAL CLUSTERING
      PHIFD=0.0
      RK=0.0
      RJ=0.0
      IF(RK.EQ.0.0) GO TO 15
      YO=0.5/RK*ALOG((1.0+(EXP(RK)-1.0)*PHIFD/180.0)/
     * (1.0-(1.0-EXP(-RK))*PHIFD/180.0))
      YO1=SINH(RK*YO)
      YO2=YO1/(RK*PHIFD/RADI)
   15 CONTINUE
      DO 35 II=2,NPHI1
      ZI=II-3
      ETA(II)=ZI*DETA
      IF(RK.GT.0.0) GO TO 40
      PHIP(II)=ETA(II)
      DTIL(II)=1.0
      DTILE(II)=0.0
      GO TO 35
40    CONTINUE
      YO3=RK*(ETA(II)/PI-YO)
      SHETA=SINH(YO3)
      CHETA=COSH(YO3)
      PHIP(II)=PHIFD/RADI*(1.0+SHETA/YO1)
      DTIL(II)=YO2*PI/CHETA
      DTILE(II)=-YO2*RK*SHETA/CHETA**2
35    CONTINUE
      PHIP(2)=-PHIP(4)
      PHIP(NPHI1)=-PHIP(NPHM1)
      DTIL(2)=DTIL(4)
      DTIL(NPHI1)=DTIL(NPHM1)
      DTILE(2)=-DTILE(4)
      DTILE(NPHI1)=-DTILE(NPHM1)
      DTILE(3)=0.0
      DTILE(NPHI)=0.0
C..RADIAL CLUSTERING
      SINHRJ=SINH(RJ)
      DO 36 II=1,NT2
      ZI=II-3
      TP(II)=ZI*DT
      IF(RJ.EQ.0.0) GO TO 41
      DT=RJ*TP(II)
```

```
      SOT=SINH(OT)
      COT=COSH(OT)
      XI(II)=SOT/SINHRJ
      TXI(II)=SINHRJ/(RJ*COT)
      TXIT(II)=-SINHRJ*SOT/COT**2
      GO TO 42
41    XI(II)=TP(II)
      TXI(II)=1.0
      TXIT(II)=0.0
42    CONTINUE
36    CONTINUE
      NREAL=0
      NCONE=2
      NTDROS=0
      ZRS=9999.0
      ZFLD=9999.0
      ITPRTR=9999
      ITPRTF=9999
      CONST(9)=.9
      ICONST(49)=5
C
C..CALCULATE FREE STREAM QUANTITIES
C
      GAM(3)=GAMMA/(GAMMA-1.0)
      GAM(4)=1.0/(GAMMA-1.0)
      AA=1.0+GAM(1)*XMACH**2
      BB=AA-1.0
      PINF=1.0/AA**GAM(3)
      RHOIN=1.0/AA**GAM(4)
      QINF=SQRT(BB/AA)
      CONST(1)=SIN(ALPH)
      CONST(2)=COS(ALPH)
      UINF=QINF*CONST(2)
      DO 3 K=2,NPHI1
      PHI=PHIP(K)
      VINF(K)=-QINF*CONST(1)*COS(PHI)
      WINF(K)=QINF*CONST(1)*SIN(PHI)
3     CONTINUE
      DO 28 J=3,NT2
      DO 28 K=3,NPHI
      P(J,K)=P90(J-2,K-2)
      RHO(J,K)=RHO90(J-2,K-2)
      U(J,K)=U90(J-2,K-2)
      V(J,K)=V90(J-2,K-2)
      W(J,K)=W90(J-2,K-2)
   28 CONTINUE
      DO 29 K=3,NPHI
      RS(K)=RS90(K-2)
      RSZ(K)=RSZ90(K-2)
      RSPHI(K)=RSPH90(K-2)
   29 CONTINUE
      DO 25 K=1,2
      M=6-K
      I=NPHI+K
      N=NPHI-K
      RS(K)=RS(M)
      RS(I)=RS(N)
      RSZ(K)=RSZ(M)
      RSZ(I)=RSZ(N)
      RSPHI(K)=-RSPHI(M)
      RSPHI(I)=-RSPHI(N)
25    CONTINUE
      ZM=P(3,3)/ABS(RHO(3,3))**GAMMA
      ZM=ABS(ZM)
      DO 38 K=3,NPHI
      S(K)=P(3,K)/ABS(RHO(3,K))**GAMMA
      S(K)=ABS(S(K))
38    CONTINUE
      S(2)=S(4)
      S(NPHI1)=S(NPHM1)
      CALL GEOM3(1,PHIP,NPHI,Z,RB,RBZ,RBPH,IPRNT)
      CALL GEOM1(2)
      RETURN
      END

      SUBROUTINE IOCON(K1)
      COMMON/CLUSTR/RJ,XI(24),TXI(24),TXIT(24)
      LEVEL 2,ETEMP,EO,FO,GO,HO
      COMMON /CVARR/ ETEMP(4,24,41), EO(4,24,41),
     *       FO(4,24,41)     , GO(4,24,41) , HO(4,24,41)
      COMMON /IOVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24)
      LEVEL 2, RHO,P,U,V,W,ROB,ROBZ,VINF,WINF,RORPH,RB,RBZ,RBPH,DTDPH,
     *  BCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT,
     *  RSPHIT
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),
     *       ROB(41)     , ROBZ(41) , VINF(41) , WINF(41) ,
     *       RORPH(41)   ,            RB(41)    , RBZ(41)    , RBPH(41) ,
     *      DTDPH(24,41), BCT(41)  ,DTDZ(24,41),DTDR(41) , ACT(41)  ,
     *       ICONST(50) , GAM(20)   , CONST(50) ,NREGON     , RS(41)    ,
     *       RSZ(41)     , RSPHI(41), RST(41)     , RSZT(41), RSPHIT(41)
      COMMON/SVARB/T,Z          , PHI    , DT     , DZ     , DPHI   , ZINT   ,
     *       ZEND   , PT     , ALPHA  , GAMMA  , SIGMA  , XMACH  , TAPE1  ,
     *       TAPE2  , DISK1  , ALPH   , DISK2  , SIGM   , NPRNT  , DZDT   ,
     *       DZDPH  , ZM     , TMWD   , TMLD   , TMW    , TML    , TTMW   ,
     *       TTML   , RZ     , BZ     , NIPHI  , NIT    , KPHI   , NITER  ,
     *       NPHI   , NPHI1  , NPHI2  , NPHI3  , NPHM1  , NPHM2  , NPHM3  ,
     *       NT     , NT1    , NT2    , NT3    , PHIFD  , NCONE  , RADI   ,
     *       PHIF   , METHOD, LAG    , NBC    , PINF   , RHOIN  , UINF   ,
     *QINF,GASCON,NREAL,NPUNCH
      DIMENSION E(4),F(4),G(4),H(4)
      II=K1+1
      GO TO (5,5,2),II
      CONTINUE
      DO 3 K=2,NPHI1
      DO 3 J=3,NT2
      T=XI(J)
C..CODE CONSERVATIVE VARIABLES
      R=T*(ROB(K)-RB(K))+RB(K)
      RR=RHO(J,K)
      PP=P(J,K)
      UU=U(J,K)
      VV=V(J,K)
      WW=W(J,K)
      A=DTDZ(J,K)
      B=DTDR(K)
      C=DTDPH(J,K)
      D=ROB(K)-RB(K)
      IF(J .EQ. 3)GO TO 4875
      RJM1=XI(J-1)*D+RB(K)
      VSTAR1=DTDZ(J-1,K)*U(J-1,K)+B*V(J-1,K)+DTDPH(J-1,K)*W(J-1,K)/RJM1
      VSTAR1=VSTAR1*UU/U(J-1,K)
 4875 VSTAR2=A*UU+B*VV+C*WW/R
      IF(J .EQ. NT2)GO TO 4775
      RJP1=XI(J+1)*D+RB(K)
      VSTAR3=DTDZ(J+1,K)*U(J+1,K)+B*V(J+1,K)+DTDPH(J+1,K)*W(J+1,K)/RJP1
      VSTAR3=VSTAR3*UU/U(J+1,K)
 4775 CONTINUE
      IF(II-2)4,4,6
4     VBAR=VSTAR2
      IF(J .LE. 4)GO TO 14
      IF(J .GE. (NT2-1))GO TO 14
      IF(VSTAR2 .GT. VSTAR1)VBAR=0.5*(VSTAR2+VSTAR1)
14    WBAR=WW
      IF(WW .GT. W(J,K-1))WBAR=0.5*(WW+W(J,K-1))
      GO TO 10
6     VBAR=VSTAR2
      IF(J .LE. 4)GO TO 15
      IF(J .GE. (NT2-1))GO TO 15
      IF(VSTAR3 .GT. VSTAR2)VBAR=0.5*(VSTAR3+VSTAR2)
15    WBAR=WW
      IF(W(J,K+1) .GT. WW)WBAR=0.5*(W(J,K+1)+WW)
10    CONTINUE
      E(1)=RR*UU
      E(2)=GAM(2)*PP+E(1)*UU
      E(3)=E(1)*VV
      E(4)=E(1)*WW
      D=GAM(2)*PP
      F(1)=RR*VBAR
      F(2)=A*D+F(1)*UU
      F(3)=B*D+F(1)*VV
      F(4)=C*D/R+F(1)*WW
      G(1)=RR*WBAR/R
      G(2)=G(1)*UU
      G(3)=G(1)*VV
      G(4)=D/R+G(1)*WW
      H(1)=RR*VV/R-ACT(K)*E(1)-(BCT(K)+DTILE(K))*RR*WW/R-TXIT(J)*RR*VV
      HH=RR*VV*UU/R
      H(2)=HH       -ACT(K)*E(2)-(BCT(K)+DTILE(K))*RR*WW*UU/R-TXIT(J)*RR*V
     $V*UU
      H(3)=RR*(VV**2-WW**2)/R-ACT(K)*E(3)-(BCT(K)+DTILE(K))*RR*WW*VV/R-
     $TXIT(J)*(GAM(2)*PP+RR*VV**2)
      H(4)=2.0*RR*VV*WW/R-ACT(K)*E(4)-(BCT(K)+DTILE(K))*(GAM(2)*PP+RR*WW
     $**2)/R-TXIT(J)*RR*VV*WW
```

```
      DO 16 N=1,4                                                     IOCON      73
      H(N)=H(N)                                                       IOCON      74
      F(N)=F(N)*TXI(J)                                                IOCON      75
      G(N)=G(N)*DTIL(K)                                               IOCON      76
16    CONTINUE                                                        IOCON      77
      GO TO (17,18),II                                                IOCON      78
17    CONTINUE                                                        IOCON      79
C..SET CONSERVATIVE VARIABLES AT N                                    IOCON      80
      DO 20 N=1,4                                                     IOCON      81
      ED(N,J,K)=E(N)                                                  IOCON      82
      FD(N,J,K)=F(N)                                                  IOCON      83
      GO(N,J,K)=G(N)                                                  IOCON      84
      HO(N,J,K)=H(N)                                                  IOCON      85
20    CONTINUE                                                        IOCON      86
      GO TO 3                                                         IOCON      87
18    CONTINUE                                                        IOCON      88
C..SET CONSERVATIVE VARIABLES AT N+1 TILDE                            IOCON      89
      DO 9 N=1,4                                                      IOCON      90
      IF(J.EQ.NT2) GO TO 36                                           IOCON      91
      GO TO 35                                                        IOCON      92
C..RESET ETEMP AT BODY AND SHOCK                                      IOCON      93
36    ETEMP(N,J,K)=E(N)                                               IOCON      94
35    CONTINUE                                                        IOCON      95
      FO(N,J,K)=F(N)                                                  IOCON      96
      GO(N,J,K)=G(N)                                                  IOCON      97
      HO(N,J,K)=H(N)                                                  IOCON      98
9     CONTINUE                                                        IOCON      99
3     CONTINUE                                                        IOCON     100
      RETURN                                                          IOCON     101
2     CONTINUE                                                        IOCON     102
C..DECODE CONSERVATIVE VARIABLES--PERFECT GAS.                        IOCON     103
      AA=1.0-GAM(2)                                                   IOCON     104
      DO 1 K=3,NPHI                                                   IOCON     105
      DO 1 J=3,NT2                                                    IOCON     106
      A=ETEMP(1,J,K)                                                  IOCON     107
      B=ETEMP(2,J,K)                                                  IOCON     108
      C=ETEMP(3,J,K)                                                  IOCON     109
      D=ETEMP(4,J,K)                                                  IOCON     110
      BB=-B/A                                                         IOCON     111
      V(J,K)=C/A                                                      IOCON     112
      W(J,K)=D/A                                                      IOCON     113
      CC=GAM(2)*(1.0-V(J,K)**2-W(J,K)**2)                             IOCON     114
      DD=BB**2-4.0*AA*CC                                              IOCON     115
      IF(DD) 7,8,8                                                    IOCON     116
7     CONTINUE                                                        IOCON     117
      DD=0.0                                                          IOCON     118
8     CONTINUE                                                        IOCON     119
      U(J,K)=(-BB+SQRT(DD))/(2.0*AA)                                  IOCON     120
      RHO(J,K)=A/U(J,K)                                               IOCON     121
      P(J,K)=RHO(J,K)*(1.0-U(J,K)**2-V(J,K)**2-W(J,K)**2)             IOCON     122
1     CONTINUE                                                        IOCON     123
      RETURN                                                          IOCON     124
      END                                                             IOCON     125
```

```
      SUBROUTINE OUTPTM                                               OUTPTM      2
      COMMON/CLUSTR/RJ,XI(24),TXI(24),TXIT(24)                        CLUSTR      2
      COMMON /DNSTRM/ ZPLOT,NZEND,NZADD,NXPLOT                        DNSTRM      2
      COMMON /IDVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24)  IDVARB   2
      LEVEL 2, RHO,P,U,V,W,ROR,RORZ,VINF,WINF,RORPH,RR,RRZ,RRPH,DTDPH,   PVARB    2
     *  RCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSTT,  PVARB    3
     *  RSPHIT                                                        PVARB       4
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),  PVARB   5
     *        ROR(41)     , RORZ(41) , VINF(41) , WINF(41) ,          PVARB       6
     *        RORPH(41)  ,              RR(41)    , RRZ(41)   , RRPH(41) ,  PVARB  7
     *       DTDPH(24,41), RCT(41)  ,DTDZ(24,41),DTDR(41) , ACT(41)  ,  PVARB      8
     *        ICONST(50) , GAM(20)  , CONST(50) ,NREGON    , RS(41)   ,  PVARB     9
     *        RST(41)     , RSPHI(41), RST(41)    , RSTT(41), RSPHIT(41)  PVARB   10
      COMMON /SHOCKS/ DRSDX(100),DST(50)                              SHOCKS      2
      COMMON/SVARB/T,Z     , PHI    , DT     , DZ     , DPHI   , ZINT   ,  SVARB  2
     *        ZEND  , PI     , ALPHA , GAMMA , SIGMA , XMACH , TAPE1 ,  SVARB     3
     *        TAPE2 , DISK1 , ALPH  , DISK2 , SIGM   , NPRNT , DZDT   ,  SVARB    4
     *        DZDPH , ZM     , THWD  , THLD  , THW    , THL    , TTHW   ,  SVARB  5
     *        TTHL  , RZ     , RZ     , NIPHI , NIT    , KPHI   , NITER ,  SVARB  6
     *        NPHI  , NPHI1 , NPHI2 , NPHI3 , NPHM1 , NPHM2 , NPHM3 ,  SVARB      7
     *        NT     , NT1    , NT2    , NT3    , PHIFD , NCONE , RADT   ,  SVARB  8
     *        PHIF  , METHOD, LAG    , NBC    , PINF   , RHOIN , UINF   ,  SVARB  9
     *OINF,GASCON,NREAL,NPUNCH                                         SVARB      10
      COMMON /FLOW/ XC(20,100),YC(20,100),VF(20,100),RHOF(20,100)     FLOW        2
      LEVEL 2, VX,VY                                                  VCOMP       2
      COMMON/VCOMP/ VX(20,100),VY(20,100)                             VCOMP       3
C                                                                     OUTPTM     11
      FACT=0.5*XMACH*XMACH*(GAMMA-1.0)                                OUTPTM     12
      FACD=(FACT+1.0)**(1.0/(GAMMA-1.0))                              OUTPTM     13
      FACV=SQRT((FACT+1.0)/FACT)                                      OUTPTM     14
      NZADD=NZADD+1                                                   OUTPTM     15
      KZ=NZEND+NZADD                                                  OUTPTM     16
      ZC=-Z                                                           OUTPTM     17
      WRITE(6,600) NZADD,ZC,RR(3),ROR(3)                              OUTPTM     18
      DRSDX(KZ)=RORZ(3)                                               OUTPTM     19
      WRITE(9) NZADD,Z,DRSDX(KZ)                                      OUTPTM     20
      DO 20 J=3,NT2                                                   OUTPTM     21
      T=XI(J)                                                         OUTPTM     22
      Y=T*(ROR(3)-RB(3))+RB(3)                                        OUTPTM     23
      JZ=J-2                                                          OUTPTM     24
      XC(JZ,KZ)=Z                                                     OUTPTM     25
      YC(JZ,KZ)=Y                                                     OUTPTM     26
      VX(JZ,KZ)=U(J,3)*FACV                                           OUTPTM     27
      VY(JZ,KZ)=V(J,3)*FACV                                           OUTPTM     28
      RHOF(JZ,KZ)=RHO(J,3)*FACD                                       OUTPTM     29
      WRITE(9) J,Y,VX(JZ,KZ),VY(JZ,KZ),RHOF(JZ,KZ)                    OUTPTM     30
   20 CONTINUE                                                        OUTPTM     31
      RETURN                                                          OUTPTM     32
C                                                                     OUTPTM     33
  600 FORMAT(5X,I3,3(10X,F10.4))                                      OUTPTM     34
      END                                                             OUTPTM     35
```

```
      SUBROUTINE PHTURN(M1,DNU,P2P1,NITS,GAMMA)                       PHTURN      2
      REAL M1,M2,M                                                    PHTURN      3
               XNU(M) = ATAN(SQRT(C*(M*M-1.)))/SQRTC-ATAN(SQRT(M*M-1.))  PHTURN   4
               XNUP(M) = SQRT(M*M-1.)/(1.+(GM-1.)/2.*M*M)/M           PHTURN      5
      GM=GAMMA                                                        PHTURN      6
      C = (GM-1.)/(GM+1.)                                             PHTURN      7
      SQRTC = SQRT(C)                                                 PHTURN      8
      EPS = 0.1E-6                                                    PHTURN      9
      XNU1 = XNU(M1)                                                  PHTURN     10
      XNU2 = XNU1+DNU                                                 PHTURN     11
      XM = M1                                                         PHTURN     12
      DO 10 I=1,20                                                    PHTURN     13
      NITS = I                                                        PHTURN     14
      M2 = XM - (XNU(XM)-XNU2)/XNUP(XM)                               PHTURN     15
      IF (M2 .GT. 100.0) GO TO 20                                     PHTURN     16
      IF (ABS((M2-XM)/XM) .LT. EPS) GO TO 30                          PHTURN     17
   10 XM = M2                                                         PHTURN     18
   20 CONTINUE                                                        PHTURN     19
      M2 = 100.0                                                      PHTURN     20
      WRITE(6,1)                                                      PHTURN     21
   30 CONTINUE                                                        PHTURN     22
      P2P1 = ((1.+(GM-1.)/2.*M1*M1)/(1.+(GM-1.)/2.*M2*M2))**(GM/(GM-1.))  PHTURN  23
      RETURN                                                          PHTURN     24
    1 FORMAT(16X,47H- - - BODY TURN STOPPED AT M2 = 100.0 - - - -)    PHTURN     25
      END                                                             PHTURN     26
```

```
      SUBROUTINE SHOCKN(K4)                                           SHOCKN      2
      LEVEL 2,ETEMP,EO,FO,GO,HO                                       CVARB       2
      COMMON /CVARB/ ETEMP(4,24,41), EO(4,24,41),                     CVARB       3
     *        FO(4,24,41)     , GO(4,24,41) , HO(4,24,41)             CVARB       4
      COMMON /IDVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24)  IDVARB   2
      LEVEL 2, RHO,P,U,V,W,ROR,RORZ,VINF,WINF,RORPH,RR,RRZ,RRPH,DTDPH,   PVARB    2
     *  RCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSTT,  PVARB    3
     *  RSPHIT                                                        PVARB       4
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),  PVARB   5
     *        ROR(41)     , RORZ(41) , VINF(41) , WINF(41) ,          PVARB       6
     *        RORPH(41)  ,              RR(41)    , RRZ(41)   , RRPH(41) ,  PVARB  7
     *       DTDPH(24,41), RCT(41)  ,DTDZ(24,41),DTDR(41) , ACT(41)  ,  PVARB      8
     *        ICONST(50) , GAM(20)  , CONST(50) ,NREGON    , RS(41)   ,  PVARB     9
     *        RST(41)     , RSPHI(41), RST(41)    , RSTT(41), RSPHIT(41)  PVARB   10
      COMMON/SVARB/T,Z     , PHI    , DT     , DZ     , DPHI   , ZINT   ,  SVARB  2
     *        ZEND  , PI     , ALPHA , GAMMA , SIGMA , XMACH , TAPE1 ,  SVARB     3
     *        TAPE2 , DISK1 , ALPH  , DISK2 , SIGM   , NPRNT , DZDT   ,  SVARB    4
     *        DZDPH , ZM     , THWD  , THLD  , THW    , THL    , TTHW   ,  SVARB  5
     *        TTHL  , RZ     , RZ     , NIPHI , NIT    , KPHI   , NITER ,  SVARB  6
     *        NPHI  , NPHI1 , NPHI2 , NPHI3 , NPHM1 , NPHM2 , NPHM3 ,  SVARB      7
     *        NT     , NT1    , NT2    , NT3    , PHIFD , NCONE , RADT   ,  SVARB  8
     *        PHIF  , METHOD, LAG    , NBC    , PINF   , RHOIN , UINF   ,  SVARB  9
     *OINF,GASCON,NREAL,NPUNCH                                         SVARB      10
C..RANKINE-HUGONIOT FUNCTIONS                                         SHOCKN      7
      UITILD(A)=SQRT(C1+C2*(A-1.))                                    SHOCKN      8
```

```
      DRSDZ(A,B,C,D)=(UINF*C+A*SQRT(D*(1.+B*B)+C*C))/D
      RHOS(A)=(A+GRATIO)/(1.+GRATIO*A)
      ABAR(A,B,D,E,F)=(F-1.)*ABS(-UINF*D+A-B*E)/((D*D+1.+E*E)*F)
      US(A,B,C)=A-C*B
      VS(A,B)=A+B
      WS(A,B,C)=A-B*C
      GAMP1=GAMMA+1.
      GAMM1=GAMMA-1.
      GRATIO=GAMM1/GAMP1
      GAMP1G=GAMP1/GAMMA
      C1=0.5*GAMM1*PINF/RHOIN
      C2=0.5*GAMP1G*C1
      GO TO (1,2),K4
1     CONTINUE
C..SHOCK PREDICTOR
      DO 3 K=3,NPHI
3     RST(K)=RS(K)+DZ*RSZ(K)
      DO 4 K=1,2
      M=6-K
      I=NPHI+K
      N=NPHI-K
      RST(K)=RST(M)
      RST(I)=RST(N)
4     CONTINUE
      DO 5 K=3,NPHI
      RSPHIT(K)=(RST(K+1)-RST(K-1))/(2.0*DETA)*DTIL(K)
      PS=P(NT2,K)
      PSRAT=PS/PINF
      U1T=U1TILD(PSRAT)
      RHRAT=RHOS(PSRAT)
      RS1=RST(K)
      RSPH=RSPHIT(K)
      RSPHR=RSPH/RS1
      FACT1=VINF(K)-WINF(K)*RSPHR
      FACT2=UINF*UINF-U1T*U1T
      IF(FACT2 .LT. 0.)U1T=-U1T
      RSZT1=DRSDZ(U1T,RSPHR,FACT1,FACT2)
      RSZT(K)=RSZT1
      ABART=ABAR(VINF(K),WINF(K),RSZT1,RSPHR,RHRAT)
      UST=US(UINF,ABART,RSZT1)
      VST=VS(VINF(K),ABART)
      WST=WS(WINF(K),ABART,RSPHR)
      IF (NREAL.EQ.0) ROST=RHRAT*RHOIN
      RHO(NT2,K)=ROST
      U(NT2,K)=UST
      V(NT2,K)=VST
      W(NT2,K)=WST
5     CONTINUE
      DO 6 K=1,2
      M=6-K
      I=NPHI+K
      N=NPHI-K
      RSZT(K)=RSZT(M)
      RSZT(I)=RSZT(N)
      RSPHIT(K)=-RSPHIT(M)
      RSPHIT(I)=-RSPHIT(N)
6     CONTINUE
      RETURN
2     CONTINUE
C..SHOCK CORRECTOR
      DO 8 K=3,NPHI
8     RS(K)=RS(K)+0.5*(RSZ(K)+RSZT(K))*DZ
      DO 9 K=1,2
      M=6-K
      I=NPHI+K
      N=NPHI-K
      RS(K)=RS(M)
      RS(I)=RS(N)
9     CONTINUE
      DO 10 K=3,NPHI
      RSPHI(K)=(RS(K+1)-RS(K-1))/(2.0*DETA)*DTIL(K)
      PS=P(NT2,K)
      PSRAT=PS/PINF
      U1T=U1TILD(PSRAT)
      RHRAT=RHOS(PSRAT)
      RS1=RS(K)
      RSPH=RSPHI(K)
      RSPHR=RSPH/RS1
      FACT1=VINF(K)-WINF(K)*RSPHR
      FACT2=UINF*UINF-U1T*U1T
      IF(FACT2 .LT. 0.)U1T=-U1T
      RSZ1 =DRSDZ(U1T,RSPHR,FACT1,FACT2)
      RSZ(K)=RSZ1
      ABART=ABAR(VINF(K),WINF(K),RSZ1 ,RSPHR,RHRAT)
      USF=US(UINF,ABART,RSZ1)
      VSF=VS(VINF(K),ABART)
      WSF=WS(WINF(K),ABART,RSPHR)
      IF (NREAL.EQ.0) ROSF=RHRAT*RHOIN
      RHO(NT2,K)=ROSF
      U(NT2,K)=USF
      V(NT2,K)=VSF
      W(NT2,K)=WSF
10    CONTINUE
      DO 11 K=1,2
      M=6-K
      I=NPHI+K
      N=NPHI-K
      RSZ(K)=RSZ(M)
      RSZ(I)=RSZ(N)
      RSPHI(K)=-RSPHI(M)
      RSPHI(I)=-RSPHI(N)
11    CONTINUE
      RETURN
      END
```

```
      SUBROUTINE SETDAT
      COMMON/CLUSTR/RJ,XI(24),TXI(24),TXIT(24)
      LEVEL 2,ETEMP,EO,FO,GO,HO
      COMMON /CVARB/ ETEMP(4,24,41), EO(4,24,41),
     *       FO(4,24,41)     , GO(4,24,41) , HO(4,24,41)
      COMMON/ENTPO/S(41),ZBS,ZFLD,ITPRTB,ITPRTF,NCASE,NTDSOS
      COMMON /IDVARB/RK,ETA(41),PHIP(41),DTIL(41),DTILE(41),DETA,TP(24)
      COMMON/JOE/ZL1,CF1,CF2,ZLF,ZTRAN,DZTRAN
      LEVEL 2, RHO,P,U,V,W,ROB,ROBZ,VINF,WINF,ROBPH,RB,RBZ,RBPH,DTDPH,
     *  BCT,DTDZ,DTDR,ACT,ICONST,GAM,CONST,NREGON,RS,RSZ,RSPHI,RST,RSZT,
     *  RSPHIT
      COMMON /PVARB/ RHO(24,41), P(24,41), U(24,41), V(24,41), W(24,41),
     *       ROB(41)      , ROBZ(41) , VINF(41) , WINF(41) ,
     *       ROBPH(41)    ,            RB(41)     , RBZ(41)   , RBPH(41) ,
     *      DTDPH(24,41), BCT(41)  ,DTDZ(24,41),DTDR(41) , ACT(41)  ,
     *       ICONST(50) , GAM(20)   , CONST(50) ,NREGON    , RS(41)    ,
     *       RSZ(41)      , RSPHI(41), RST(41)    , RSZT(41), RSPHIT(41)
      COMMON/SVARB/T,Z       , PHI    , DT     , DZ     , DPHI   , ZINT   ,
     *       ZEND   , PI     , ALPHA  , GAMMA  , SIGMA  , XMACH  , TAPE1  ,
     *       TAPE2  , DISK1  , ALPH   , DISK2  , SIGM   , NPRNT  , DZDT   ,
     *       DZDPH  , ZM     , TMWD   , TMLD   , TMW    , TML    , TTMW   ,
     *       TTML   , RZ     , BZ     , NIPHI  , NIT    , KPHI   , NITER  ,
     *       NPHI   , NPHI1  , NPHI2  , NPHI3  , NPHM1  , NPHM2  , NPHM3  ,
     *       NT     , NT1    , NT2    , NT3    , PHIFD  , NCONE  , RADI   ,
     *       PHIF   , METHOD, LAG    , NBC    , PINF   , RHOIN  , UINF   ,
     *QINF,GASCON,NREAL,NPUNCH
C
C     INITIALISE ARRAYS
C
      ICONST(1)=0
      ICONST(2)=2
      ICONST(3)=0
      ICONST(4)=1
      DO 1 I=5,50
      ICONST(I)=0
    1 CONTINUE
      NBC=1
      METHOD=2
      NPRNT=0
      DZDT=0.0
    6 CONTINUE
      DO 8 I=1,20
      GAM(I)=0.0
    8 CONTINUE
      DO 9 I=1,50
      CONST(I)=0.0
    9 CONTINUE
      DO 10 I=1,24
      TP(I)=0.0
   10 CONTINUE
      DO 11 I=1,41
      ROB(I)=0.0
      ROBZ(I)=0.0
      VINF(I)=0.0
      WINF(I)=0.0
      ROBPH(I)=0.0
      RB(I)=0.0
      RBZ(I)=0.0
      RBPH(I)=0.0
```

```
      RCT(I)=0.0                                                       SETDAT    43
      DTDR(I)=0.0                                                      SETDAT    44
      ACT(I)=0.0                                                       SETDAT    45
      RS(I)=0.0                                                        SETDAT    46
      RSZ(I)=0.0                                                       SETDAT    47
      RSPHI(I)=0.0                                                     SETDAT    48
      RST(I)=0.0                                                       SETDAT    49
      RSZT(I)=0.0                                                      SETDAT    50
      RSPHIT(I)=0.0                                                    SETDAT    51
      ETA(I)=0.0                                                       SETDAT    52
      PHIP(I)=0.0                                                      SETDAT    53
      DTIL(I)=0.0                                                      SETDAT    54
      DTILE(I)=0.0                                                     SETDAT    55
      DO 11 J=1,24                                                     SETDAT    56
      RHO(J,I)=0.0                                                     SETDAT    57
      P(J,I)=0.0                                                       SETDAT    58
      U(J,I)=0.0                                                       SETDAT    59
      V(J,I)=0.0                                                       SETDAT    60
      W(J,I)=0.0                                                       SETDAT    61
      DTDPH(J,I)=0.0                                                   SETDAT    62
      DTDZ(J,I)=0.0                                                    SETDAT    63
   11 CONTINUE                                                         SETDAT    64
      RETURN                                                           SETDAT    65
      END                                                              SETDAT    66
```

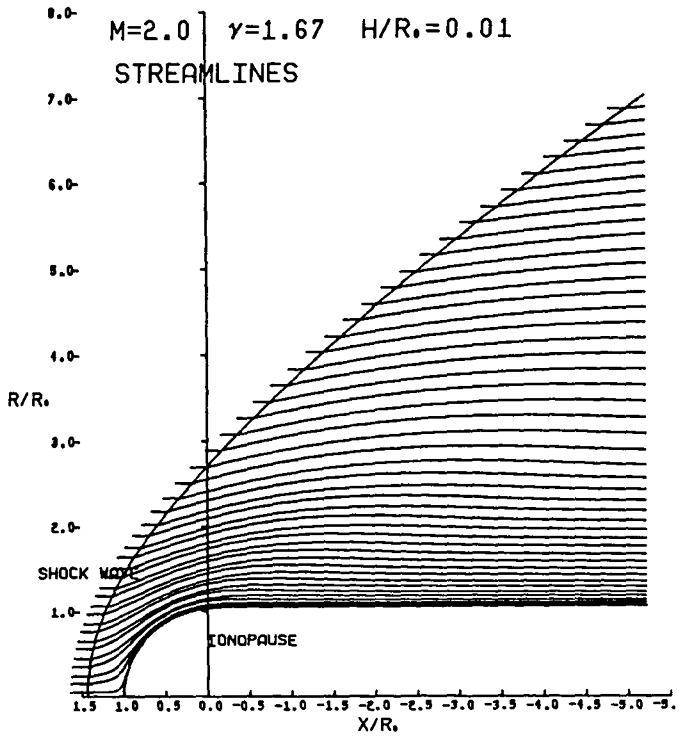
This Page Intentionally Left Blank

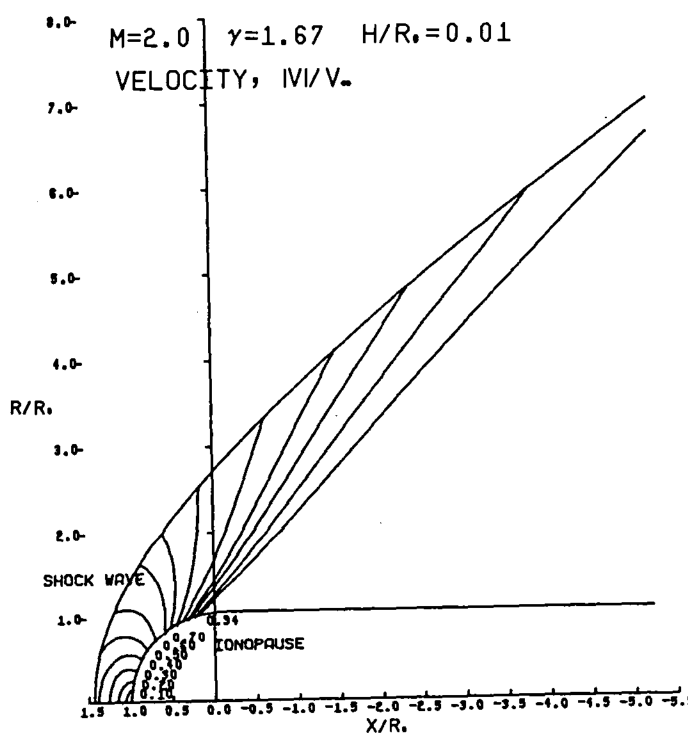
APPENDIX C

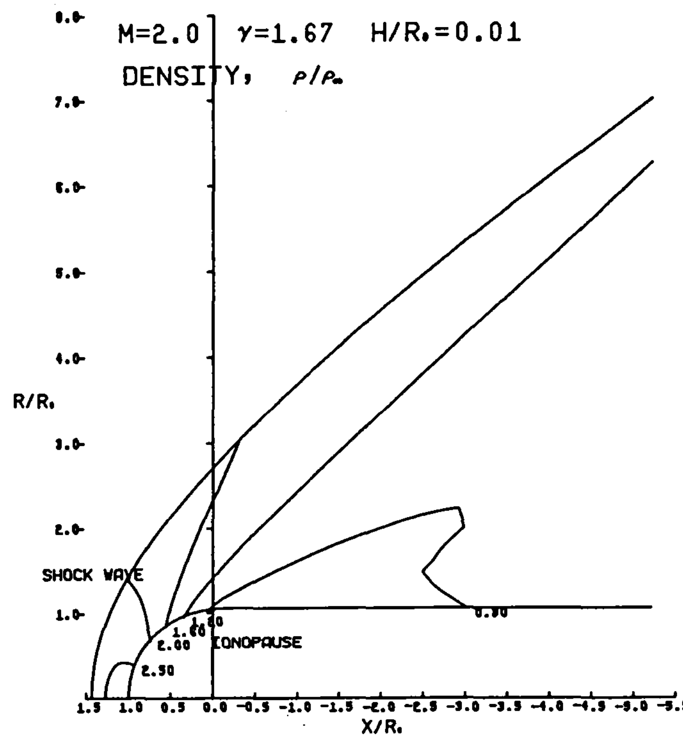
CATALOG OF TEST CASES

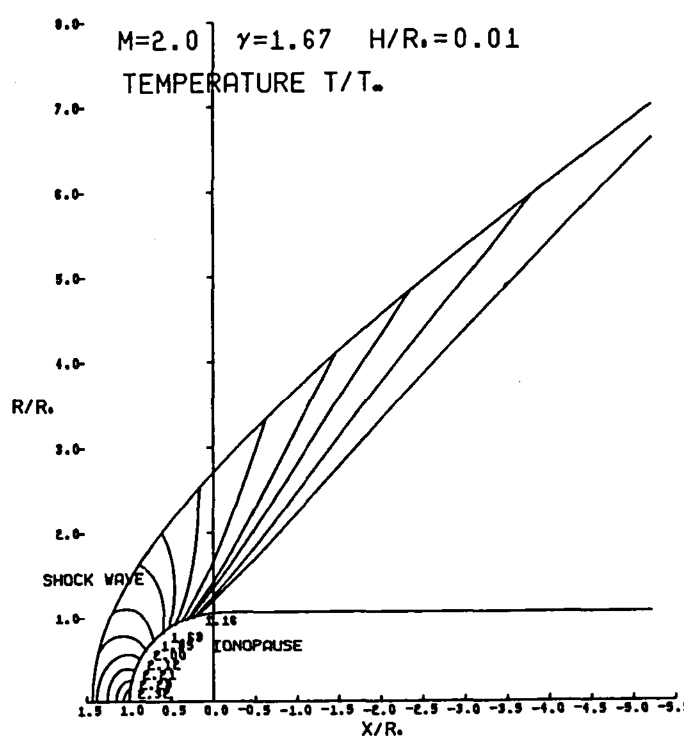
Catalog page number index for plasma streamline, velocity magnitude, density, temperature, and unit magnetic-field maps for various solar-wind flows past planetary ionopauses

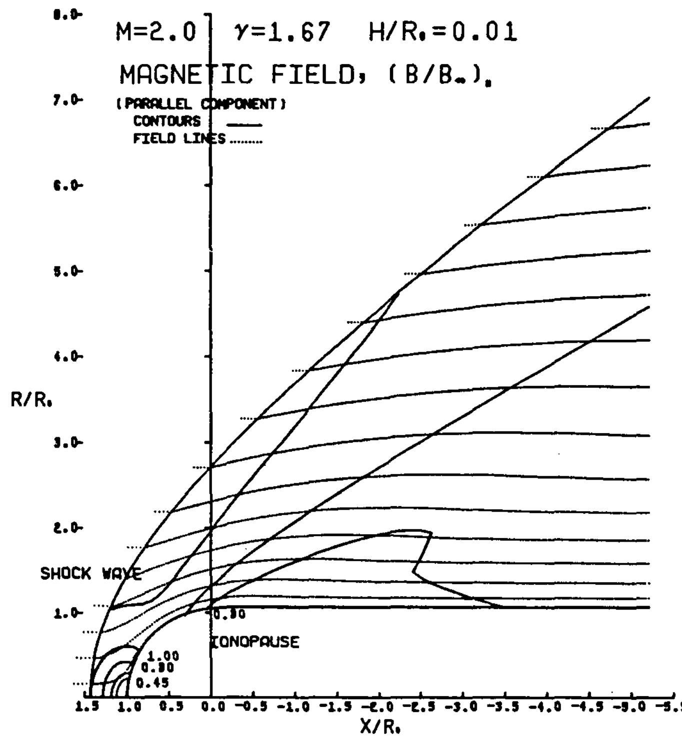
H/R_o	M_∞	γ	(Page No.) Streamlines	(Page No.) Velocity magnitude $\lvert \underset{\sim}{v} \rvert / v_\infty$	(Page No.) Density ρ/ρ_∞	(Page No.) Temperature T/T_∞	(Page No.) Parallel magnetic field $(\lvert \underset{\sim}{B} \rvert / B_\infty)_\parallel$	(Page No.) Perpendicular magnetic field $(\lvert \underset{\sim}{B} \rvert / B_\infty)_\perp$
.01	2.0	5/3	179	179	180	180	181	181
	3.0		182	182	183	183	184	184
	5.0		185	185	186	186	187	187
	8.0		188	188	189	189	190	190
	12.0		191	191	192	192	193	193
	25.0		194	194	195	195	196	196
.10	2.0		197	197	198	198	199	199
	3.0		200	200	201	201	202	202
	5.0		203	203	204	204	205	205
	8.0		206	206	207	207	208	208
	12.0		209	209	210	210	211	211
	25.0		212	212	213	213	214	214
.25	2.0		215	215	216	216	217	217
	3.0		218	218	219	219	220	220
	5.0		221	221	222	222	223	223
	8.0		224	224	225	225	226	226
	12.0		227	227	228	228	229	229
	25.0		230	230	231	231	232	232
$\bar{H}/R_o$	M_∞	γ	Streamlines	$\lvert \underset{\sim}{v} \rvert / v_\infty$	ρ/ρ_∞	T/T_∞	$(\lvert B \rvert / B_\infty)_\parallel$	$(\lvert B \rvert / B_\infty)_\perp$
.10	2.0	5/3	233	233	234	234	235	235
	3.0		236	236	237	237	238	238
	5.0		239	239	240	240	241	241
	8.0		242	242	243	243	244	244
	12.0		245	245	246	246	247	247
	25.0		248	248	249	249	250	250
.20	2.0		251	251	252	252	253	253
	3.0		254	254	255	255	256	256
	5.0		257	257	258	258	259	259
	8.0		260	260	261	261	262	262
	12.0		263	263	264	264	265	265
	25.0		266	266	267	267	268	268
.25	2.0		269	269	270	270	271	271
	3.0		272	272	273	273	274	274
	5.0		275	275	276	276	277	277
	8.0		278	278	279	279	280	280
	12.0		281	281	282	282	283	283
	25.0		284	284	285	285	286	286

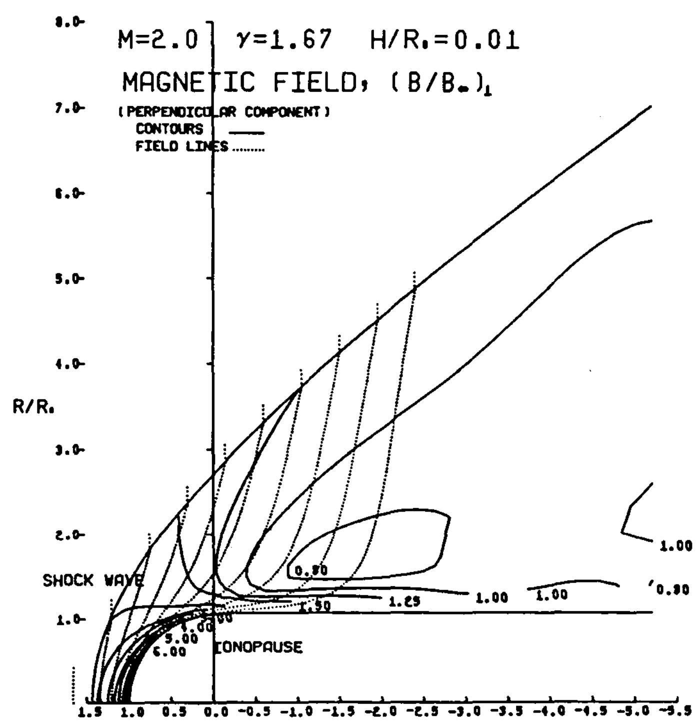

M=2.0 γ=1.67 H/R₀=0.01
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

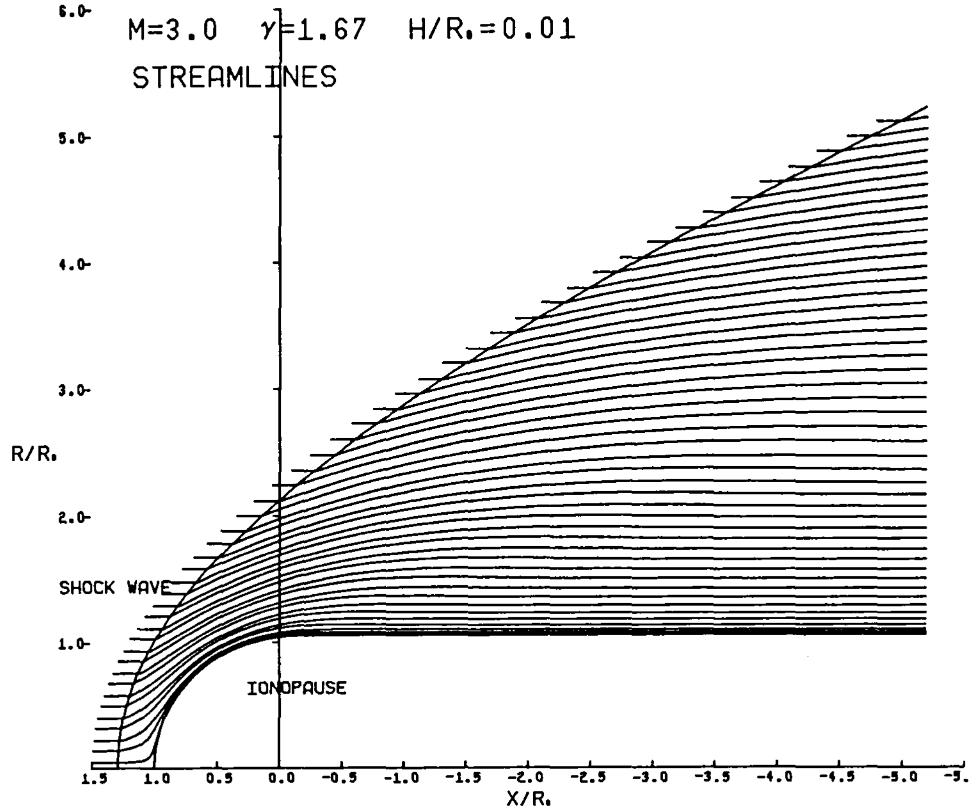

M=2.0 γ=1.67 H/R₀=0.01
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

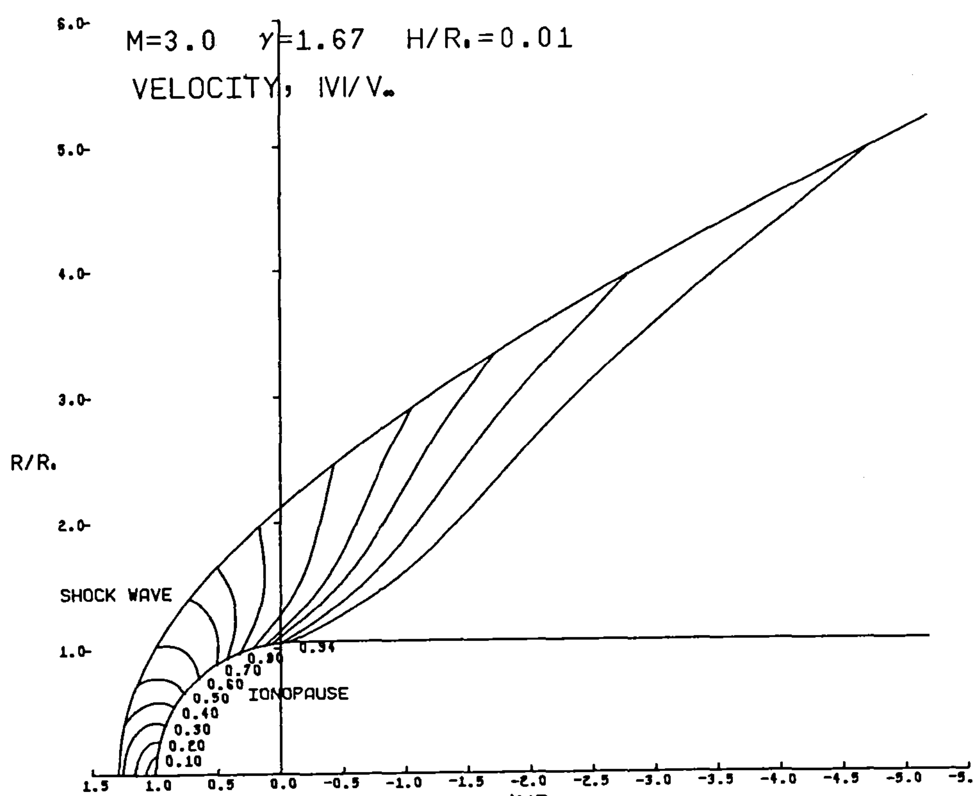

M=2.0 γ=1.67 H/R₀=0.01
DENSITY, ρ/ρ∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

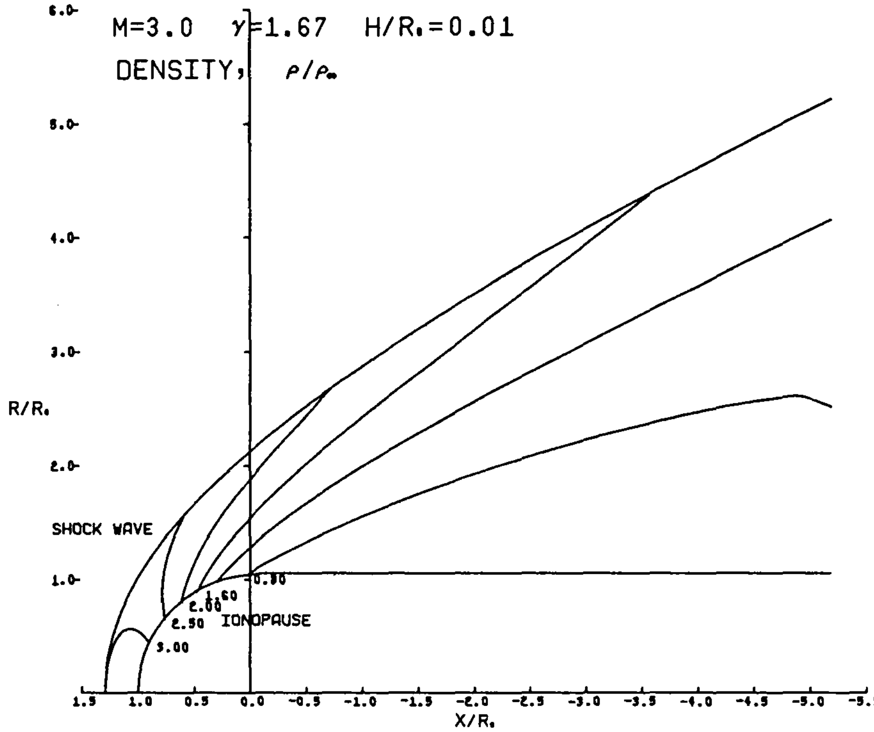

M=2.0 γ=1.67 H/R₀=0.01
TEMPERATURE T/T∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

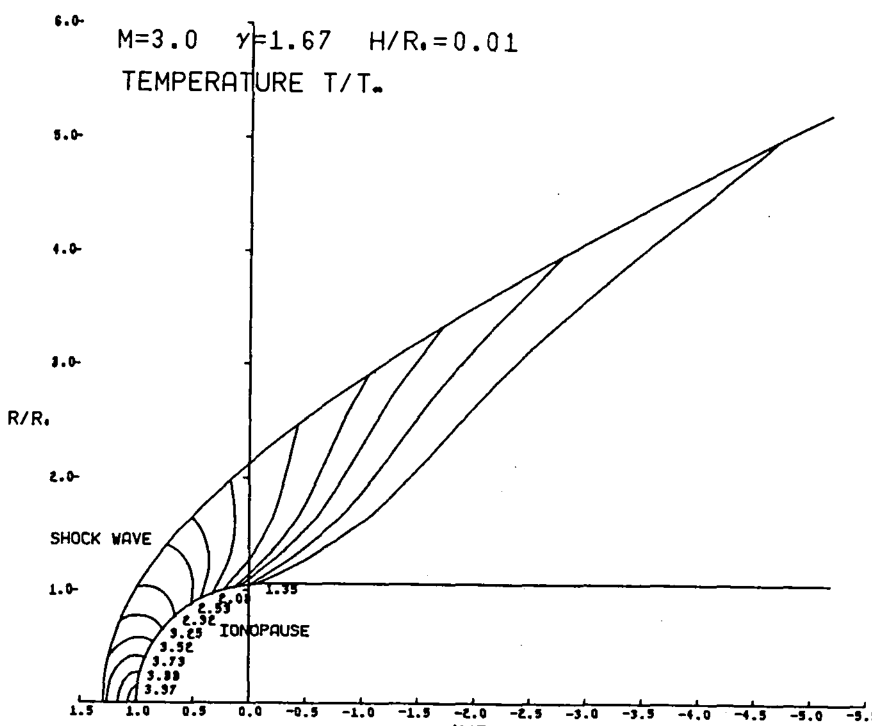

M=2.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

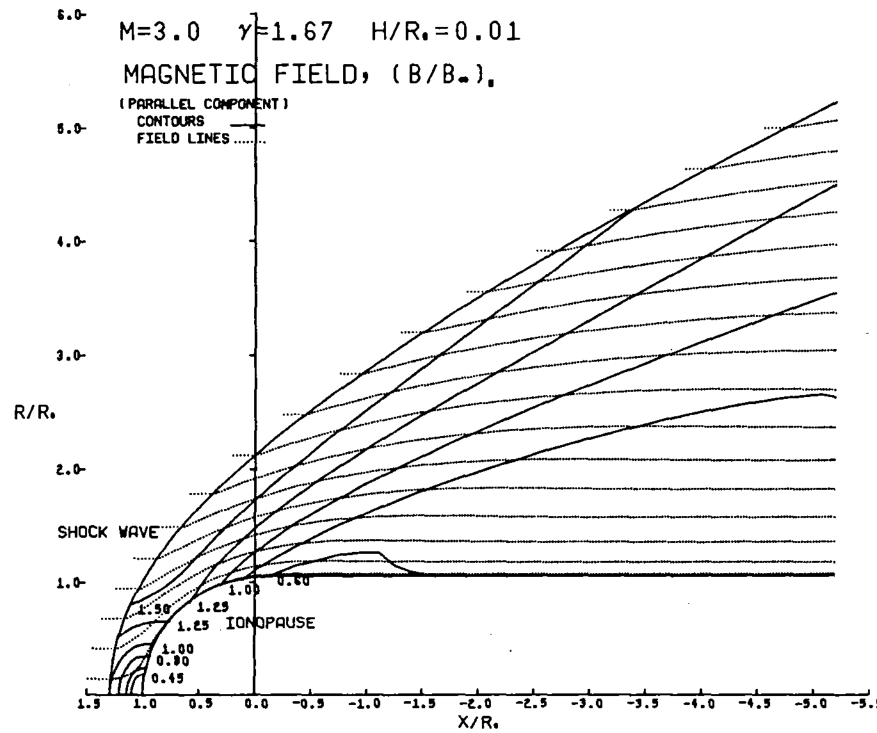

M=2.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

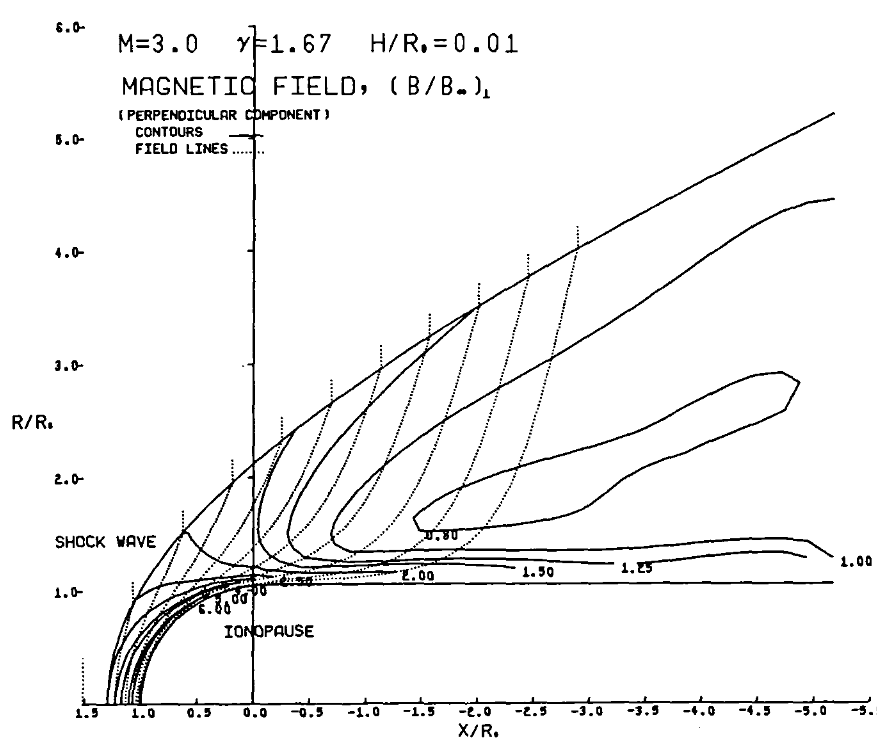

M=3.0 γ=1.67 H/R₀=0.01
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀
6.0
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

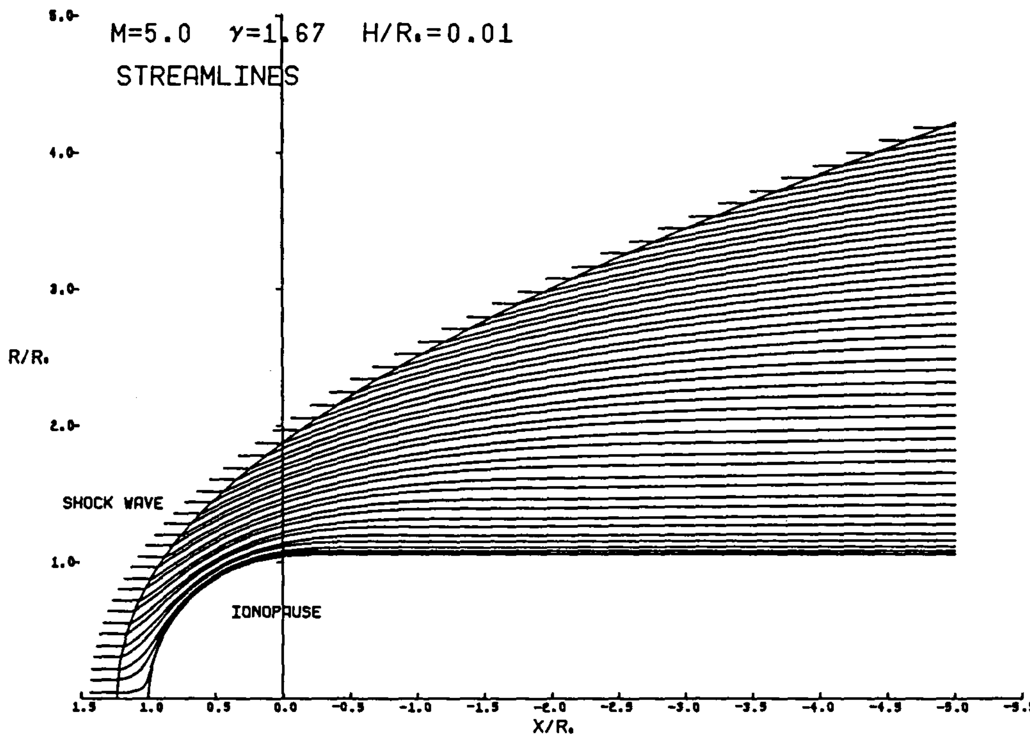

M=3.0 γ=1.67 H/R₀=0.01
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
0.94
0.80
0.70
0.60
0.50
0.40
0.30
0.20
0.10
X/R₀
6.0
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

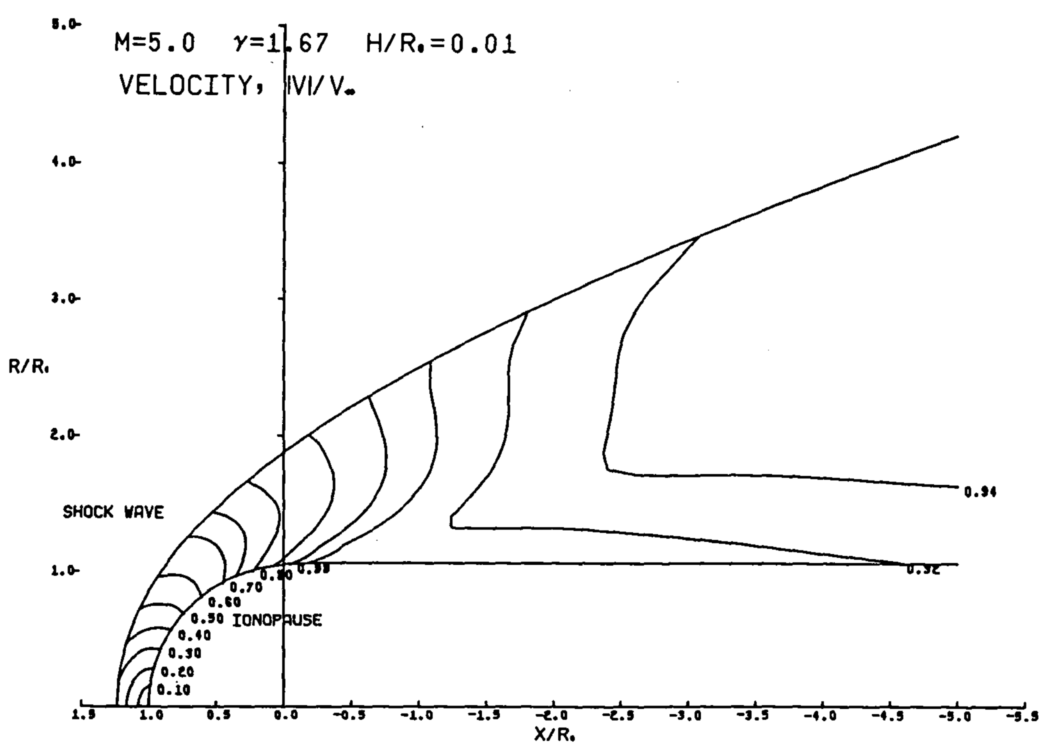

M=3.0 γ=1.67 H/R₀=0.01
DENSITY, ρ/ρ∞
R/R₀
SHOCK WAVE
IONOPAUSE
0.90
1.60
2.00
2.50
3.00
X/R₀

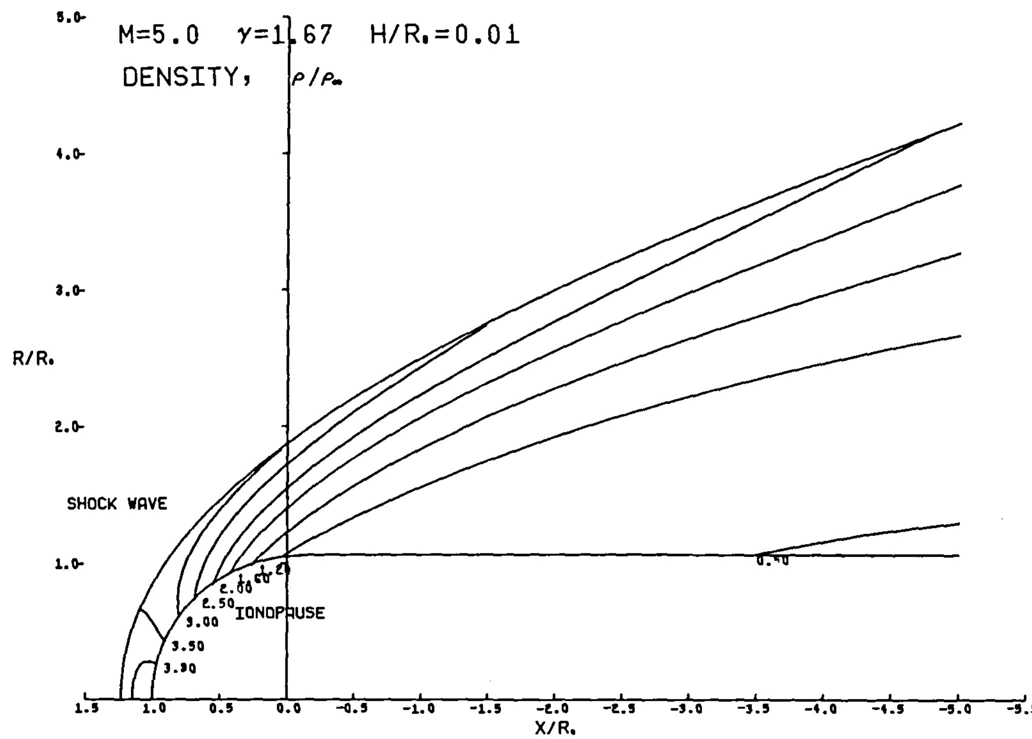

M=3.0 γ=1.67 H/R₀=0.01
TEMPERATURE T/T∞
R/R₀
SHOCK WAVE
IONOPAUSE
1.35
2.02
2.53
2.32
3.25
3.52
3.73
3.89
3.97
X/R₀

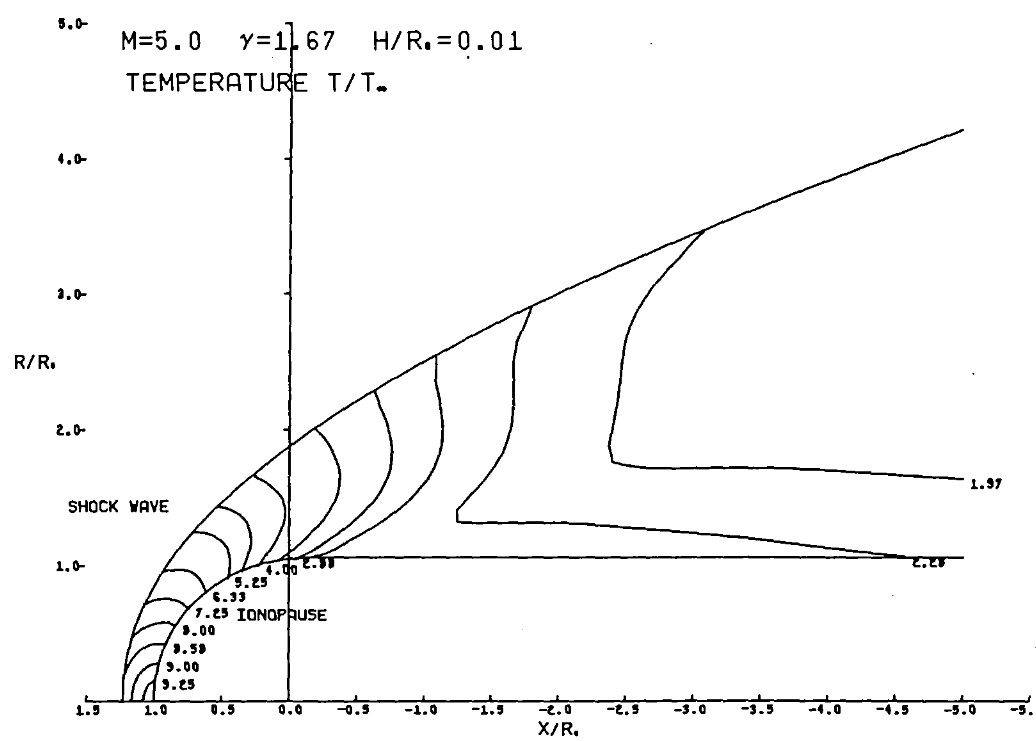

M=3.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.50
1.25
1.25
1.00
1.00
0.80
0.45
0.60

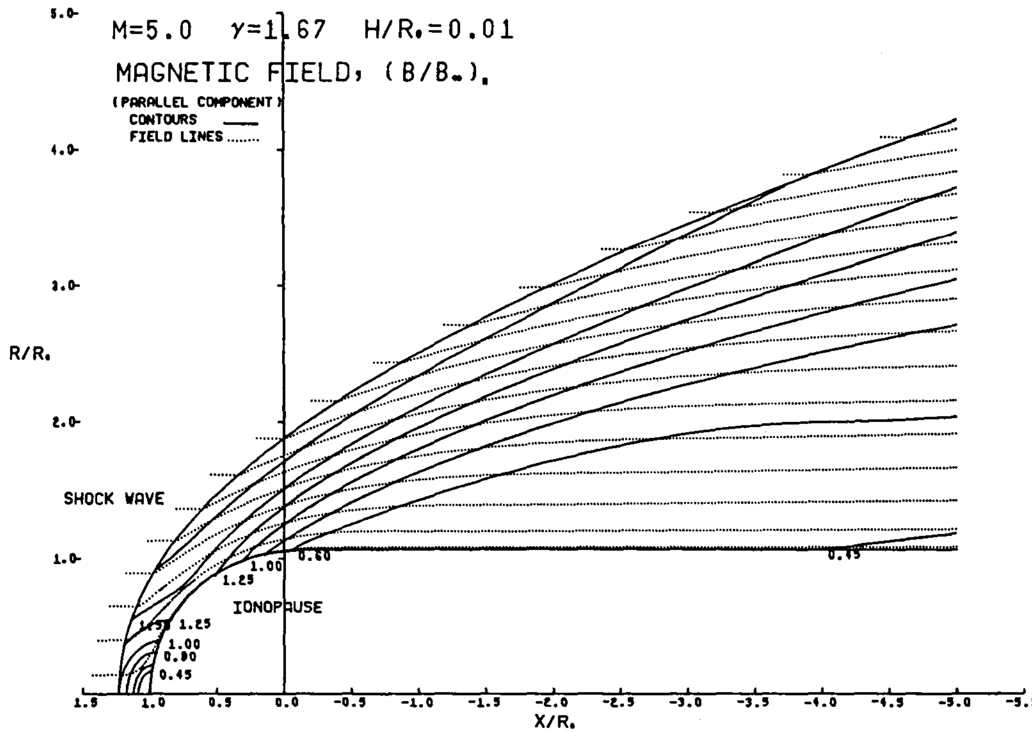

M=3.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
2.00
1.50
1.25
1.00
6.00

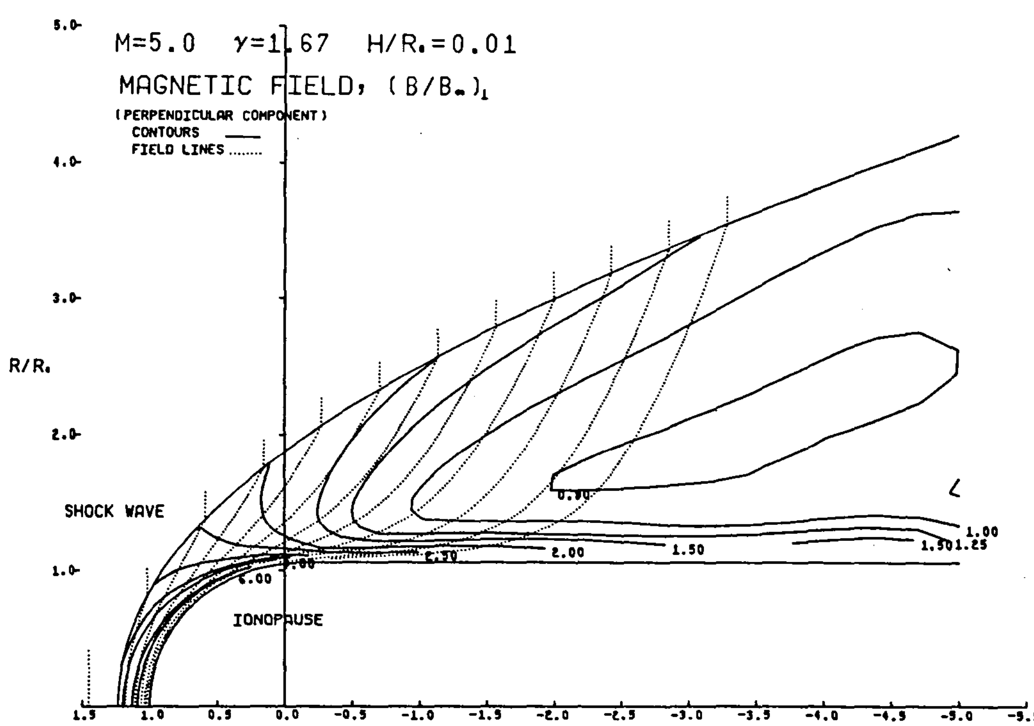

M=5.0 γ=1.67 H/R₀=0.01
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

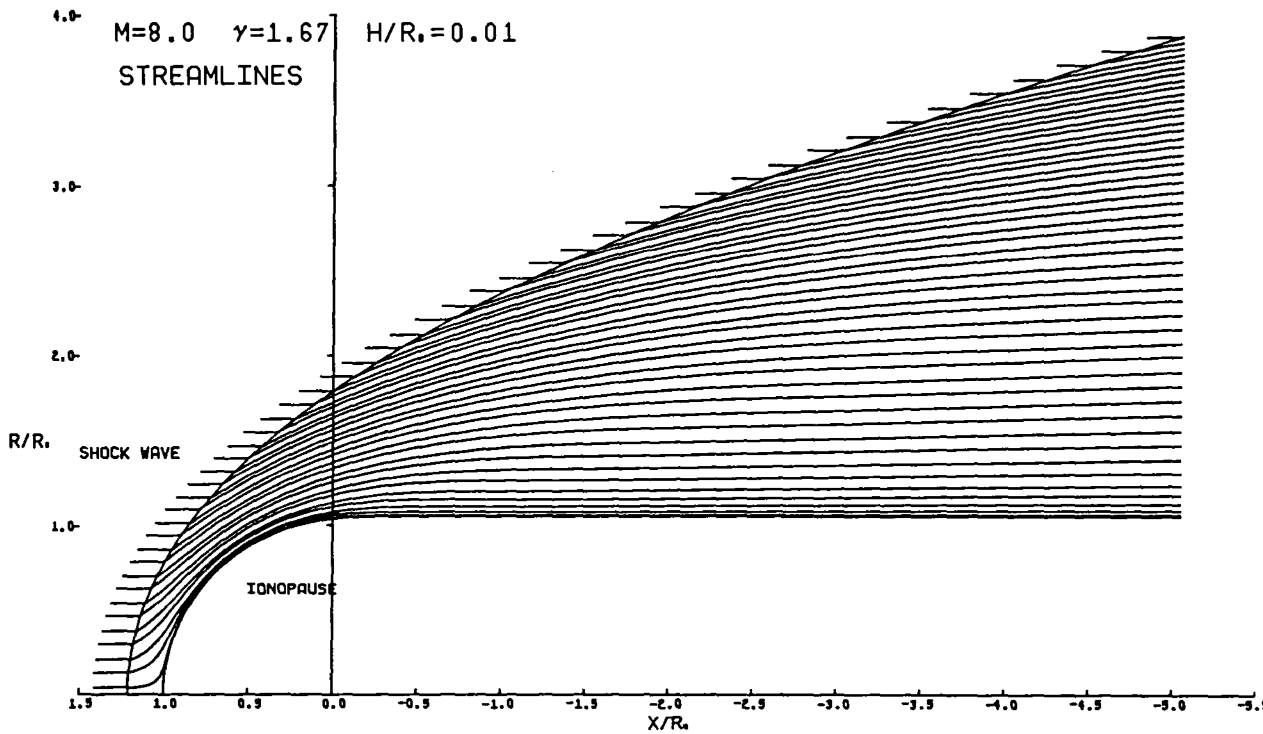

M=5.0 γ=1.67 H/R₀=0.01
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
0.94
0.92
0.80
0.70
0.60
0.50
0.40
0.30
0.20
0.10
R/R₀
X/R₀

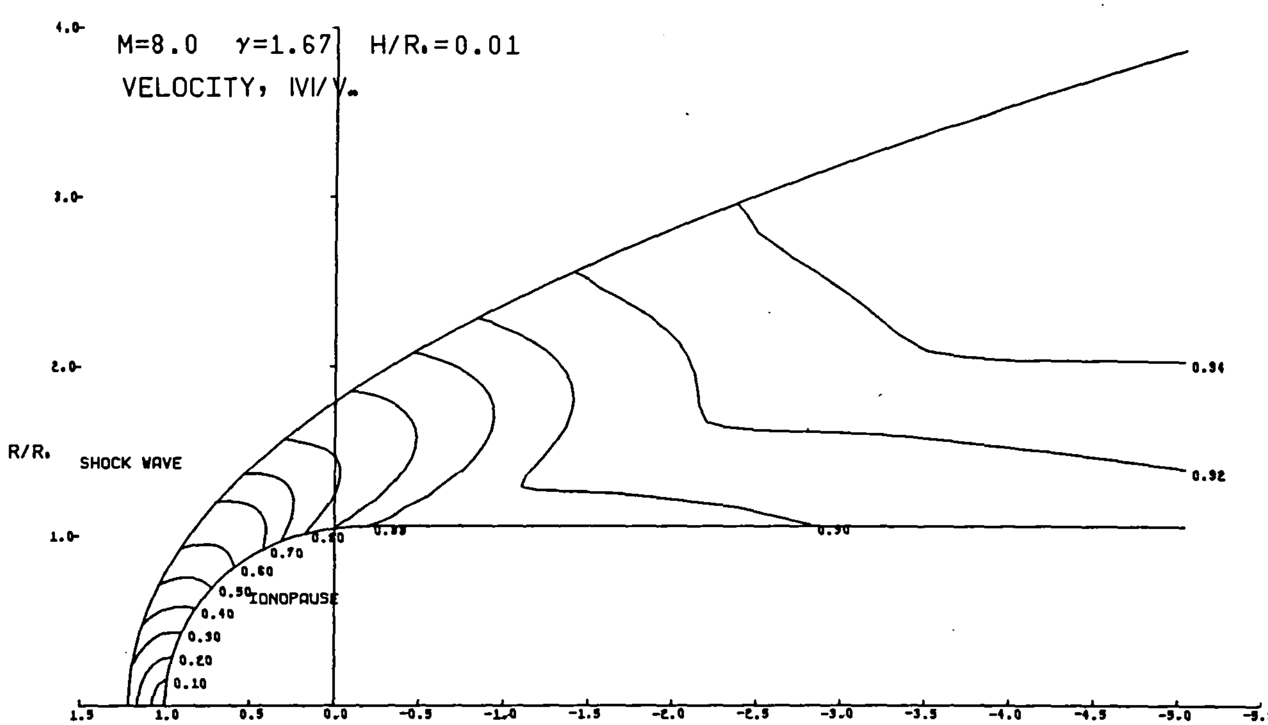

M=5.0 γ=1.67 H/R₀=0.01
DENSITY, ρ/ρ∞
R/R₀
SHOCK WAVE
IONOPAUSE
2.00
2.50
3.00
3.50
3.90
X/R₀

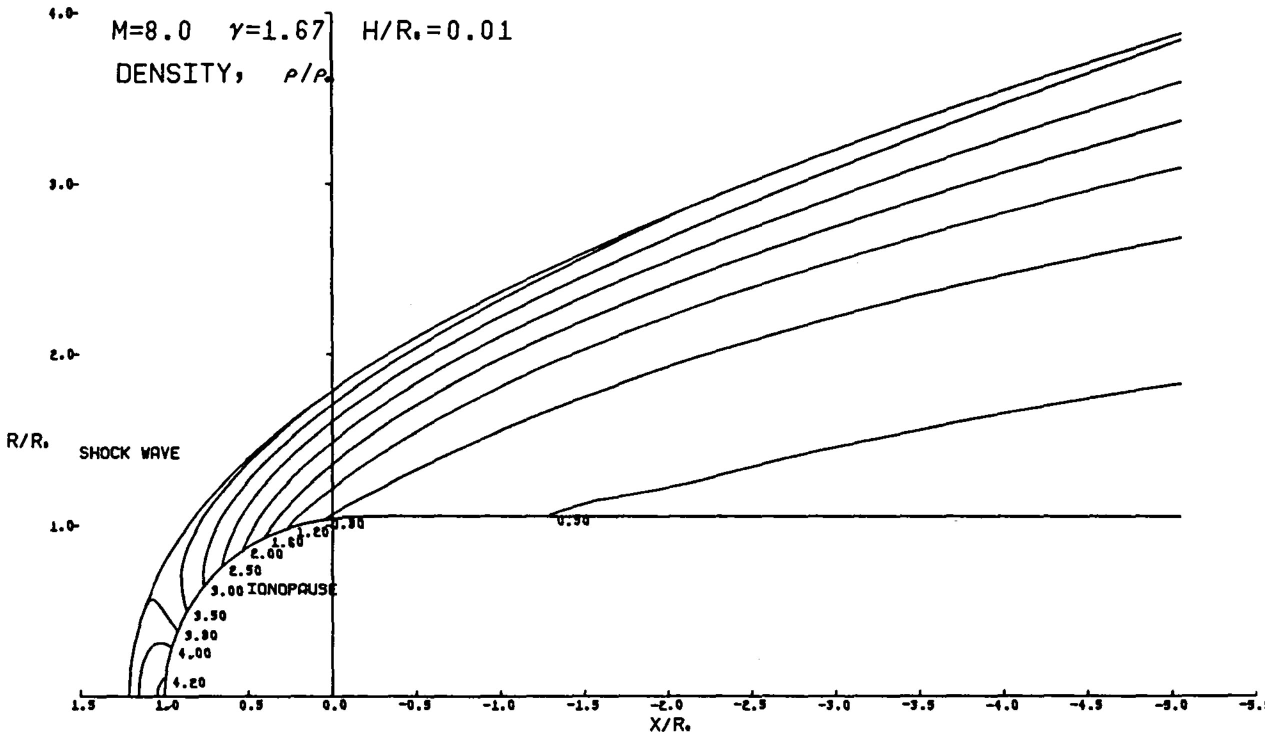

M=5.0 γ=1.67 H/R₀=0.01
TEMPERATURE T/T∞
R/R₀
SHOCK WAVE
IONOPAUSE
1.97
4.00
5.25
6.33
7.25
X/R₀

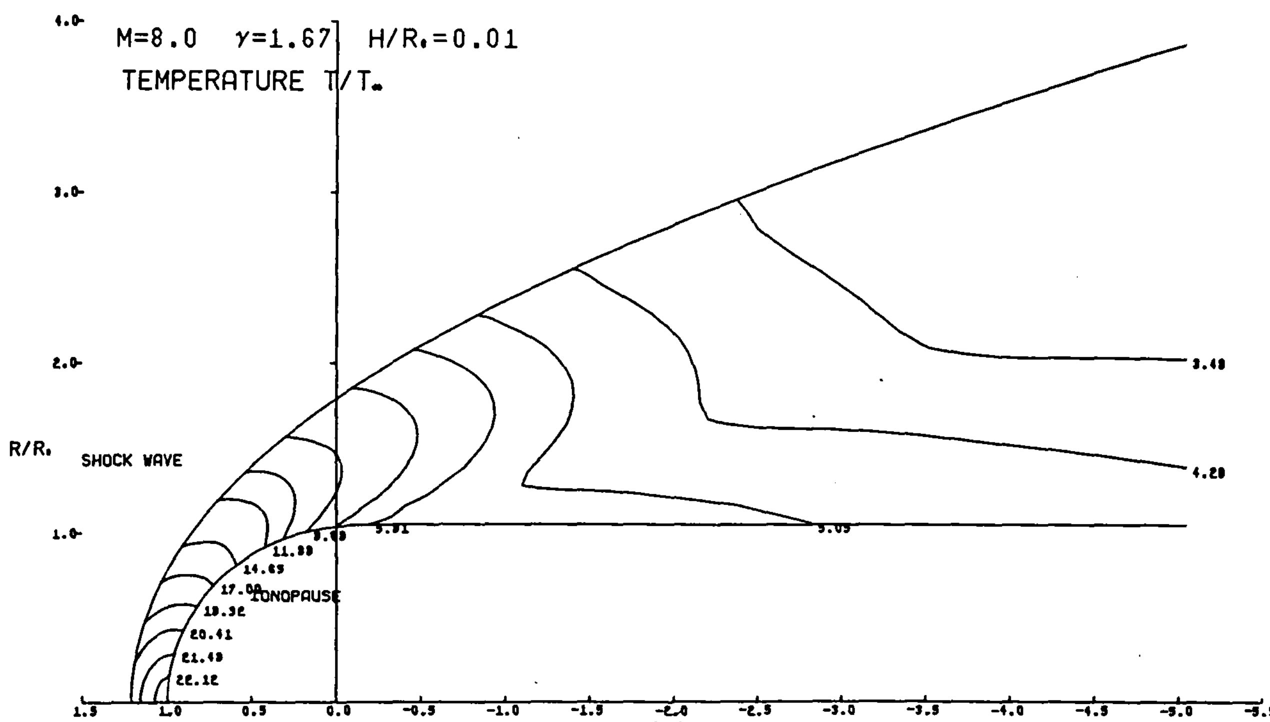

M=5.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
IONOPAUSE
1.00
1.25
0.60
0.45
1.25
1.00
0.80
0.45
X/R₀
5.0
4.0
3.0
2.0
1.0
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
-4.5
-5.0
-5.5

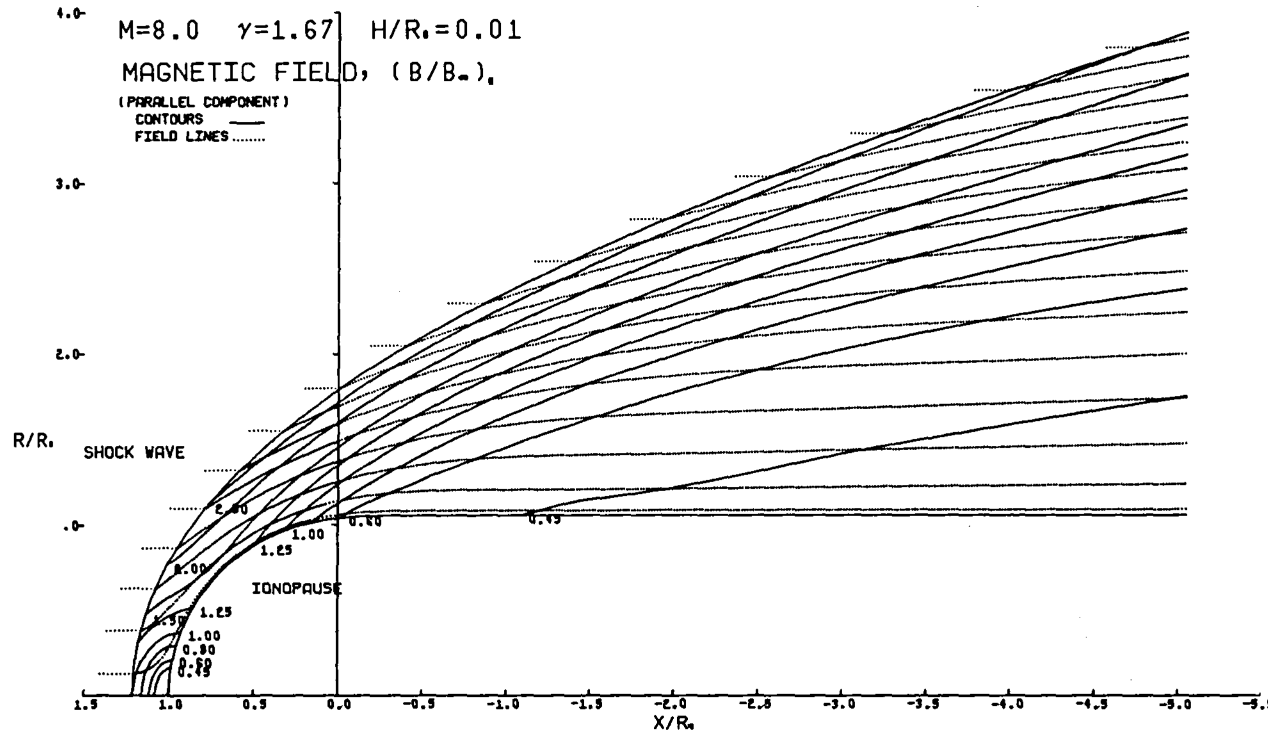

M=5.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
IONOPAUSE
2.50
2.00
1.50
1.00
1.50
1.25
6.00
0.90
X/R₀
5.0
4.0
3.0
2.0
1.0
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
-4.5
-5.0
-5.5

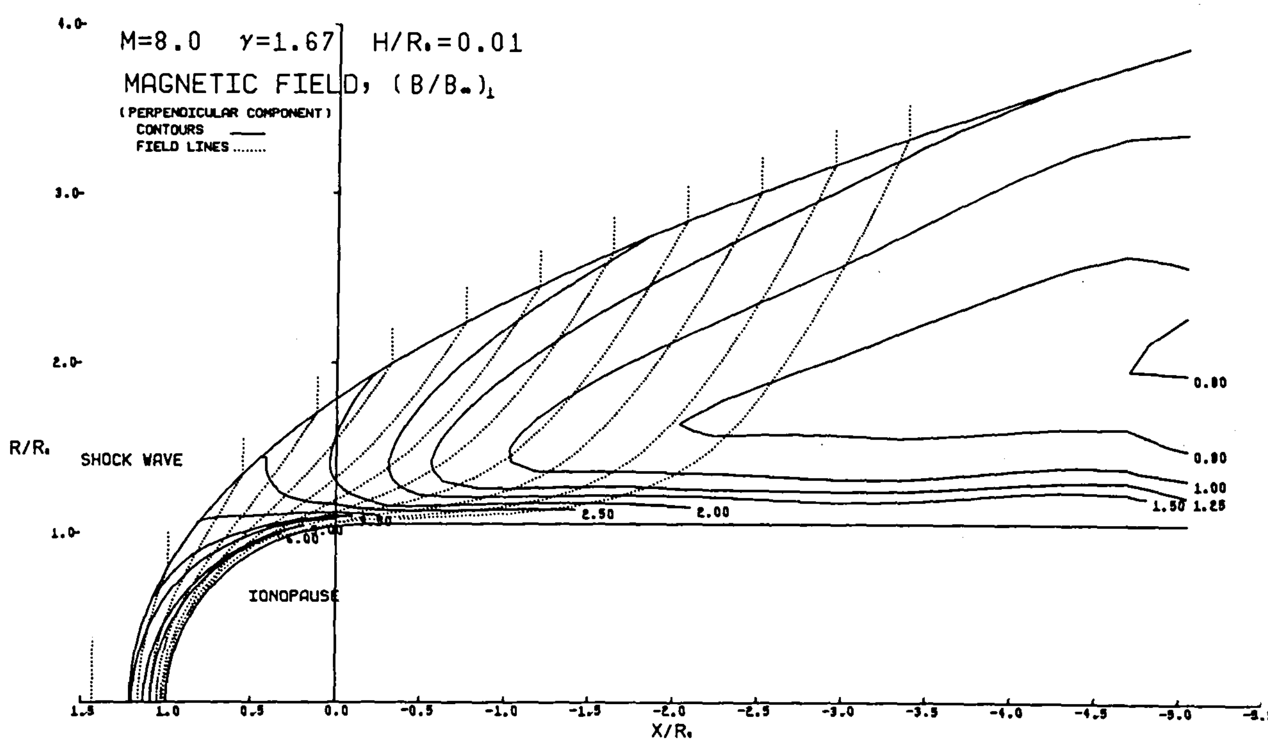

M=8.0 γ=1.67 H/R₀=0.01
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
4.0
3.0
2.0
1.0
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
-4.5
-5.0
-5.5

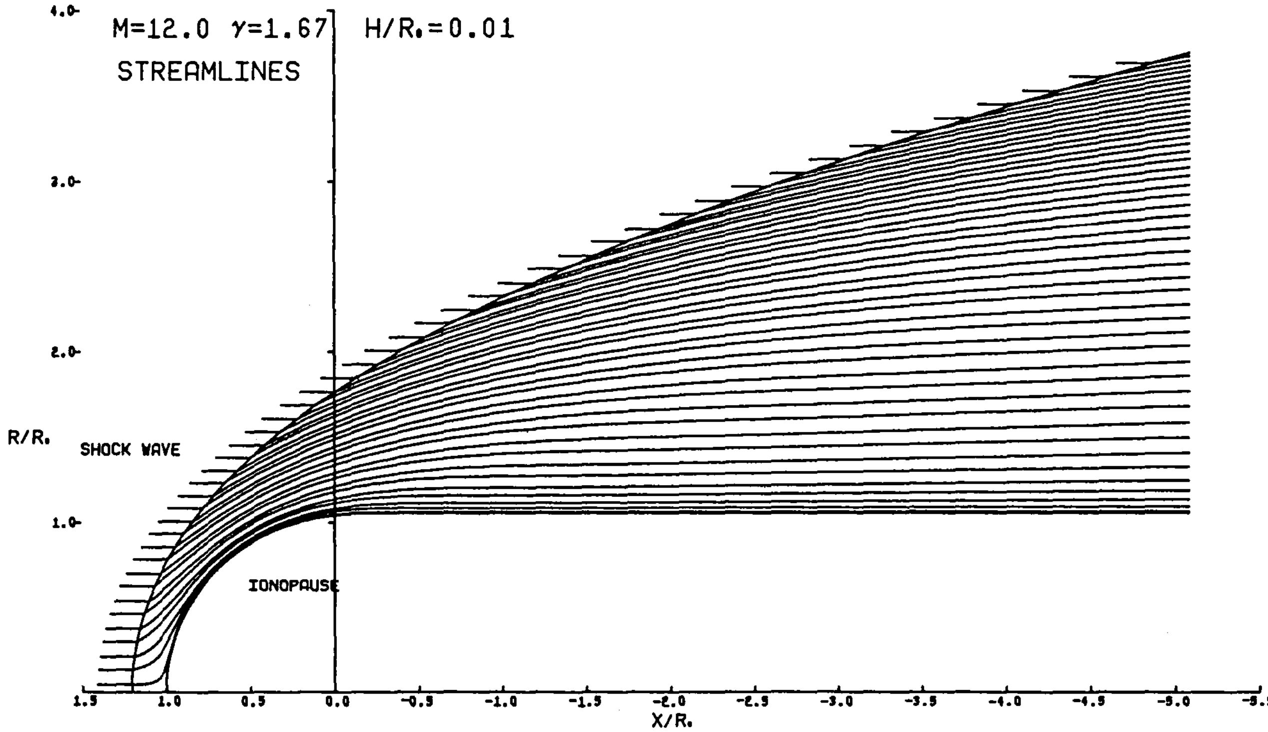

M=8.0 γ=1.67 H/R₀=0.01
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.94
0.92
0.90
0.80
0.70
0.60
0.50
0.40
0.30
0.20
0.10
4.0
3.0
2.0
1.0
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
-4.5
-5.0
-5.5

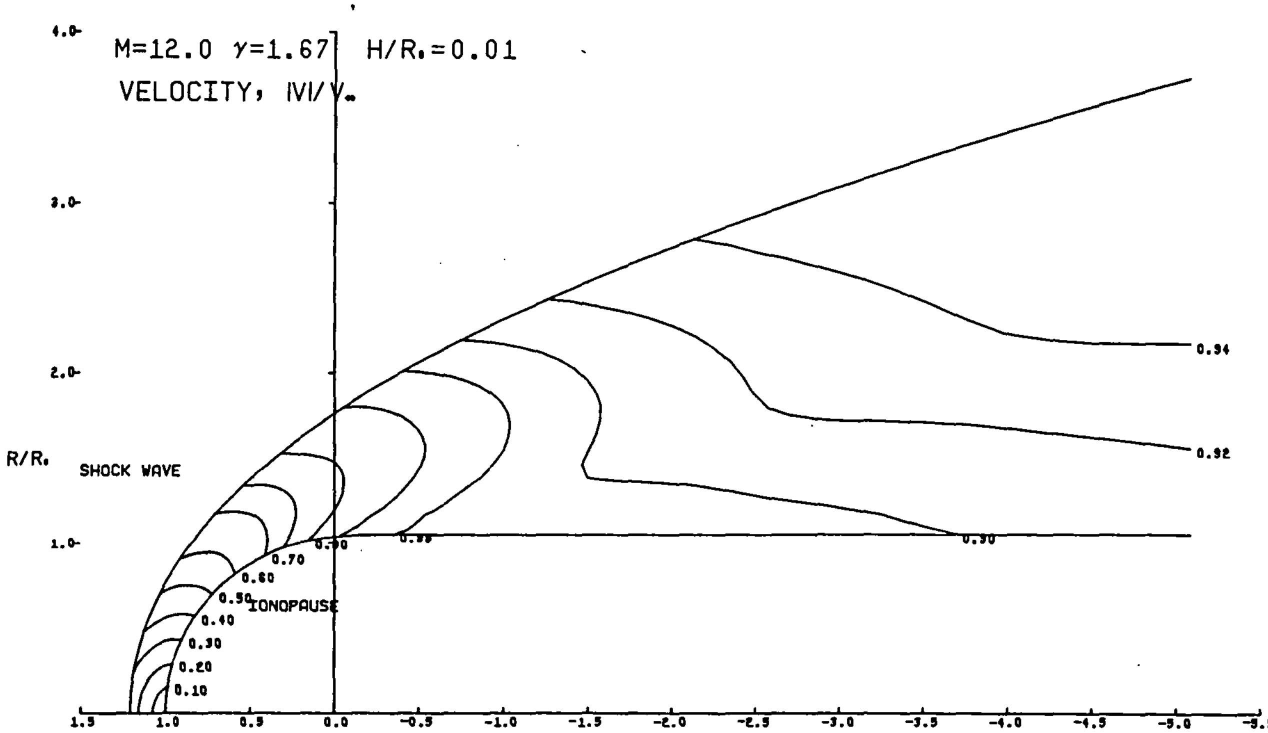

M=8.0 γ=1.67 H/R₀=0.01
DENSITY, ρ/ρ∞
R/R₀
SHOCK WAVE
IONOPAUSE
1.20
1.60
2.00
2.50
3.00
3.50
3.90
4.00
4.20
X/R₀

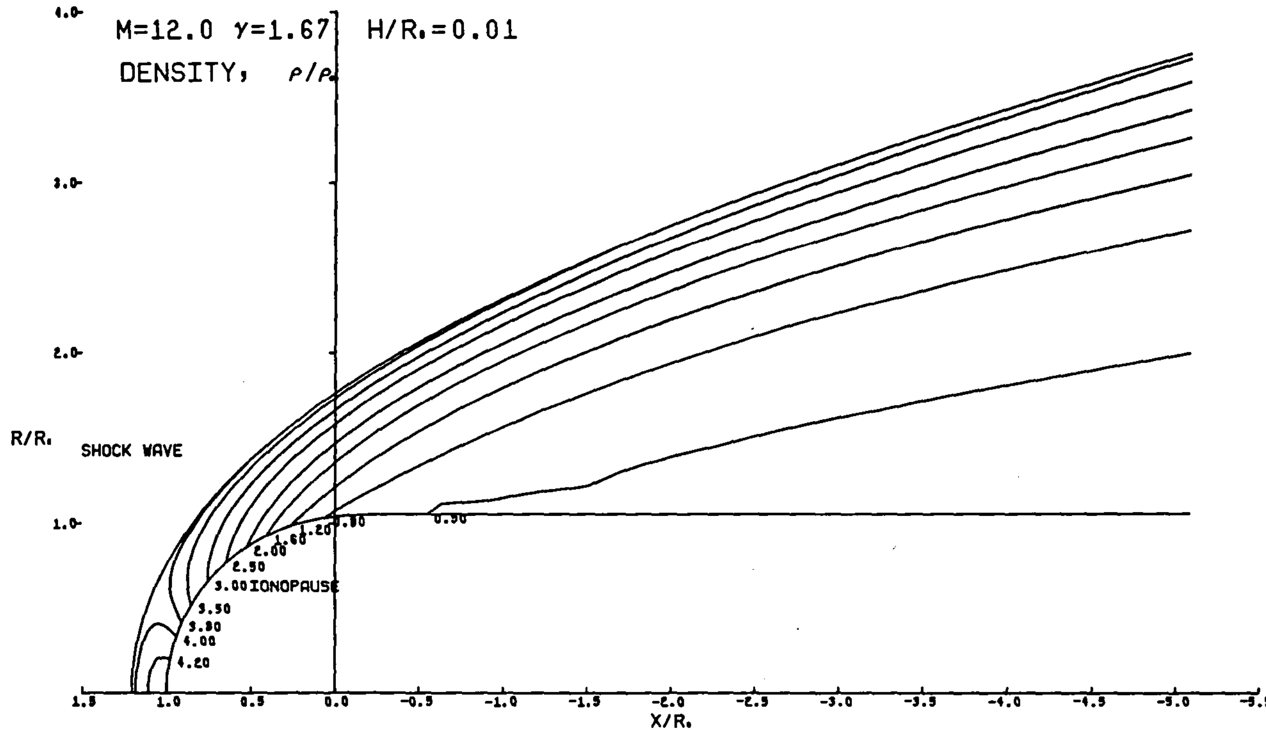

M=8.0 γ=1.67 H/R₀=0.01
TEMPERATURE T/T∞
R/R₀
SHOCK WAVE
IONOPAUSE
11.39
14.65
17.00
19.32
20.41
21.49
22.12
4.20
X/R₀


M=8.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

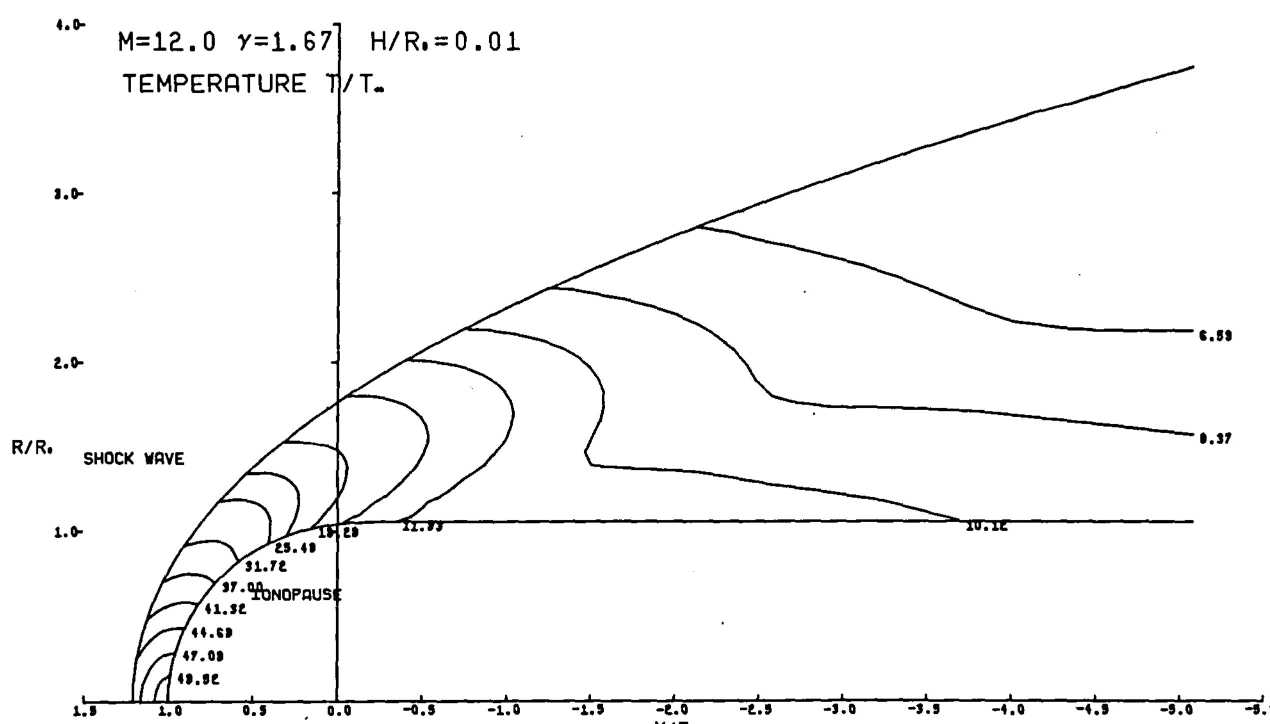

M=8.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀


M=12.0 γ=1.67 H/R₀=0.01
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

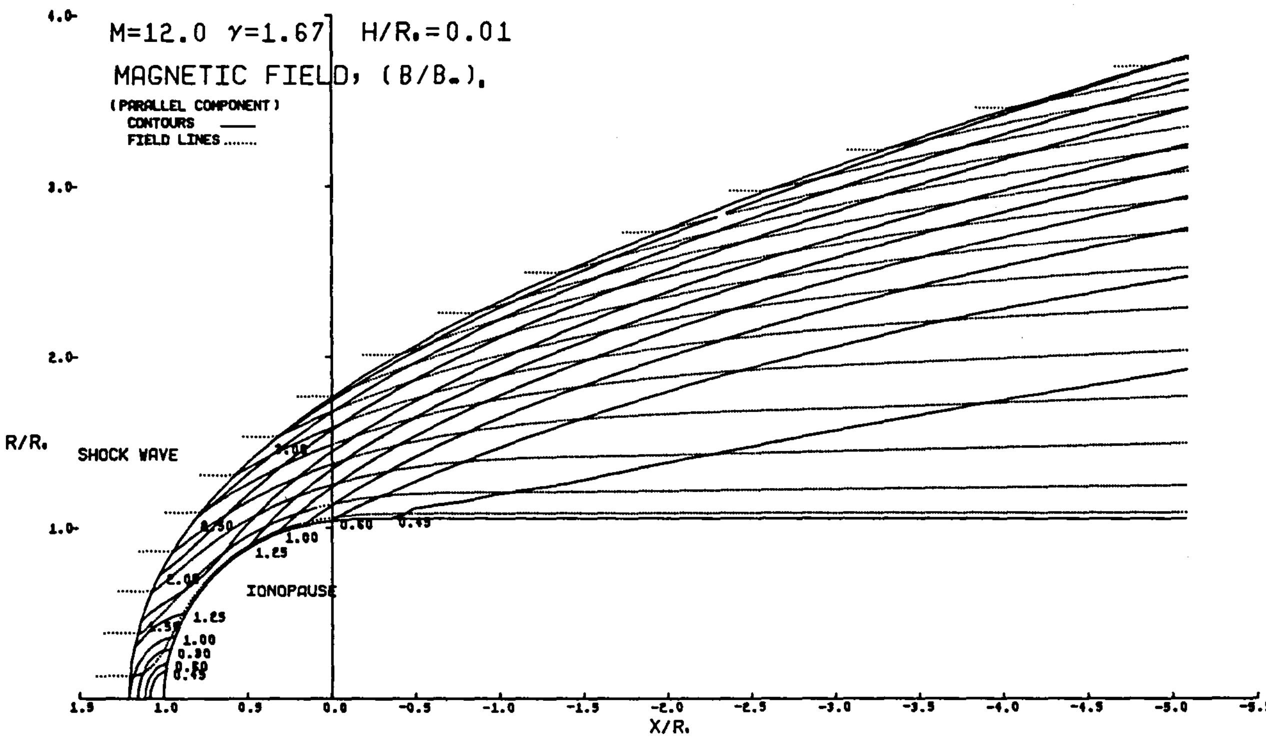

M=12.0 γ=1.67 H/R₀=0.01
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.94
0.92
0.90
0.70
0.60
0.50
0.40
0.30
0.20
0.10

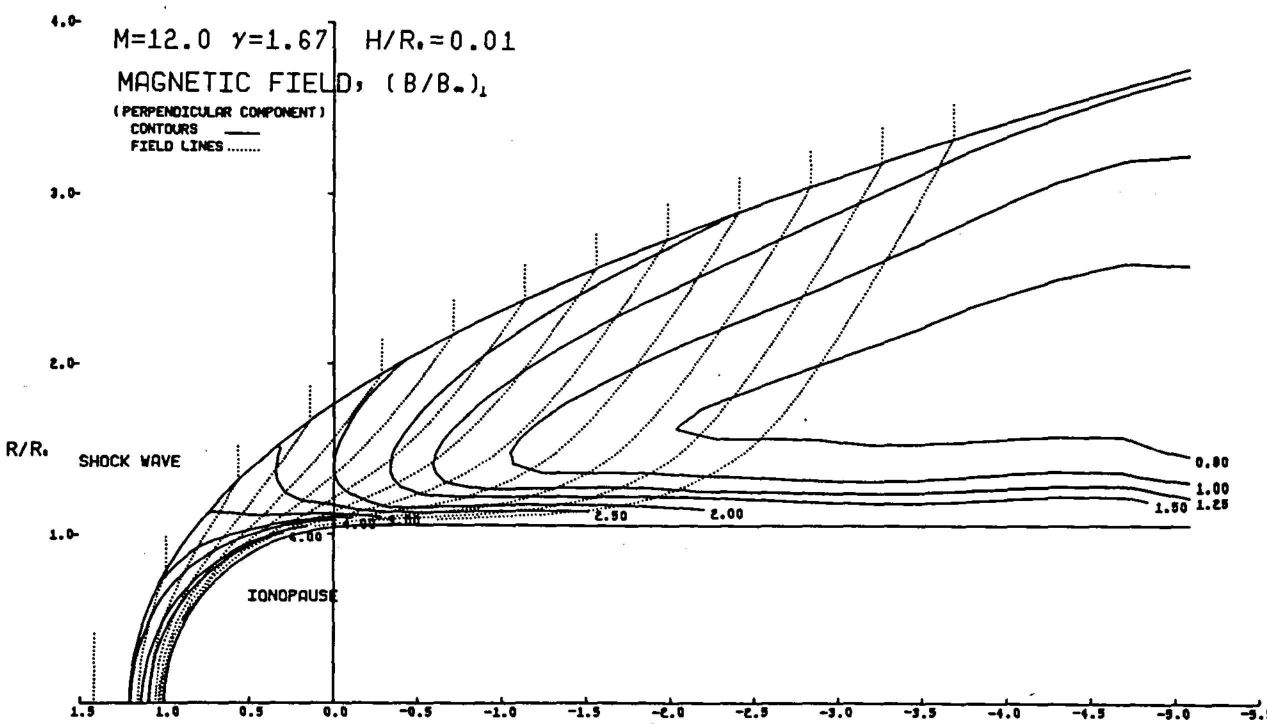


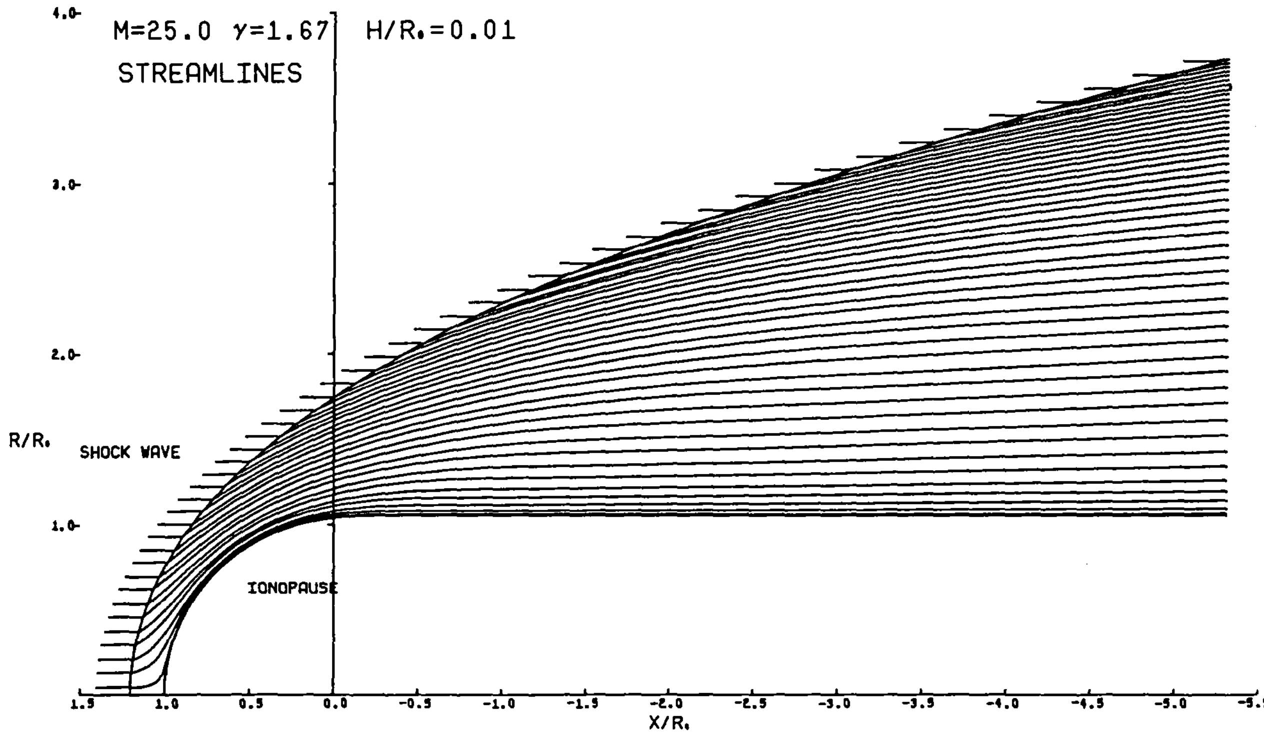
M=12.0 γ=1.67 H/R₀=0.01
DENSITY, ρ/ρ∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

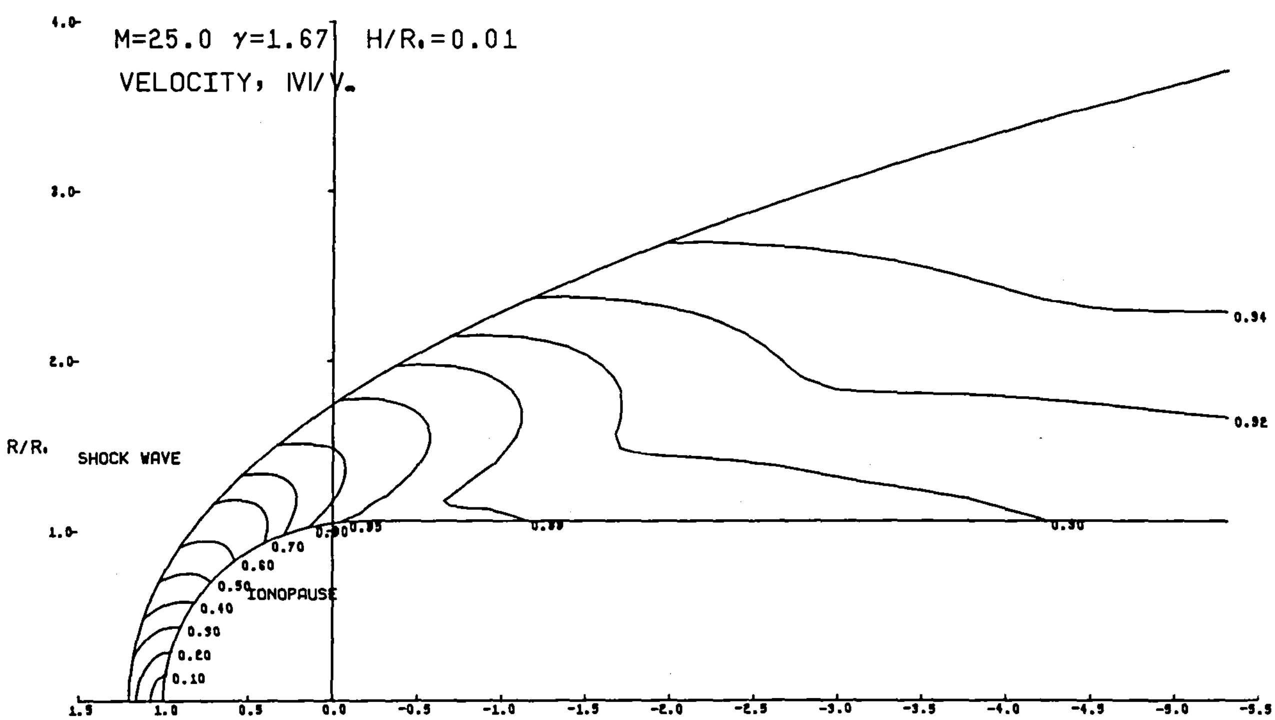


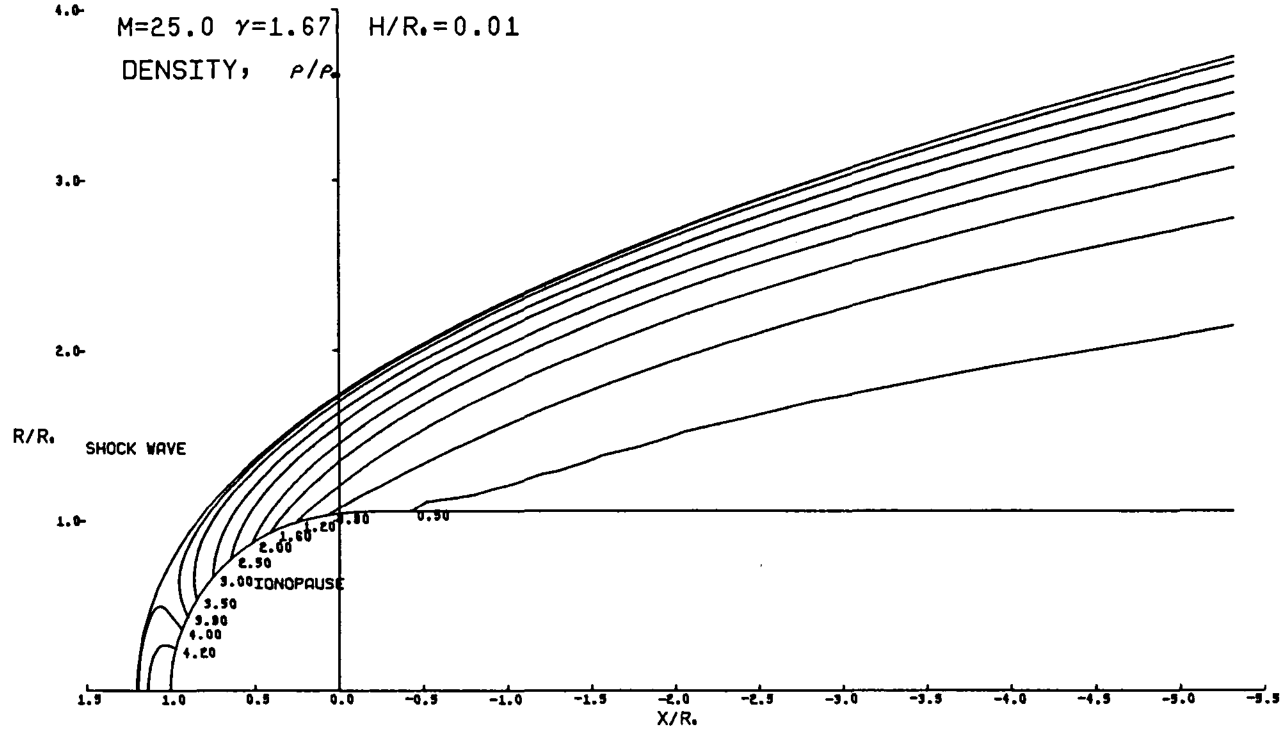
M=12.0 γ=1.67 H/R₀=0.01
TEMPERATURE T/T∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

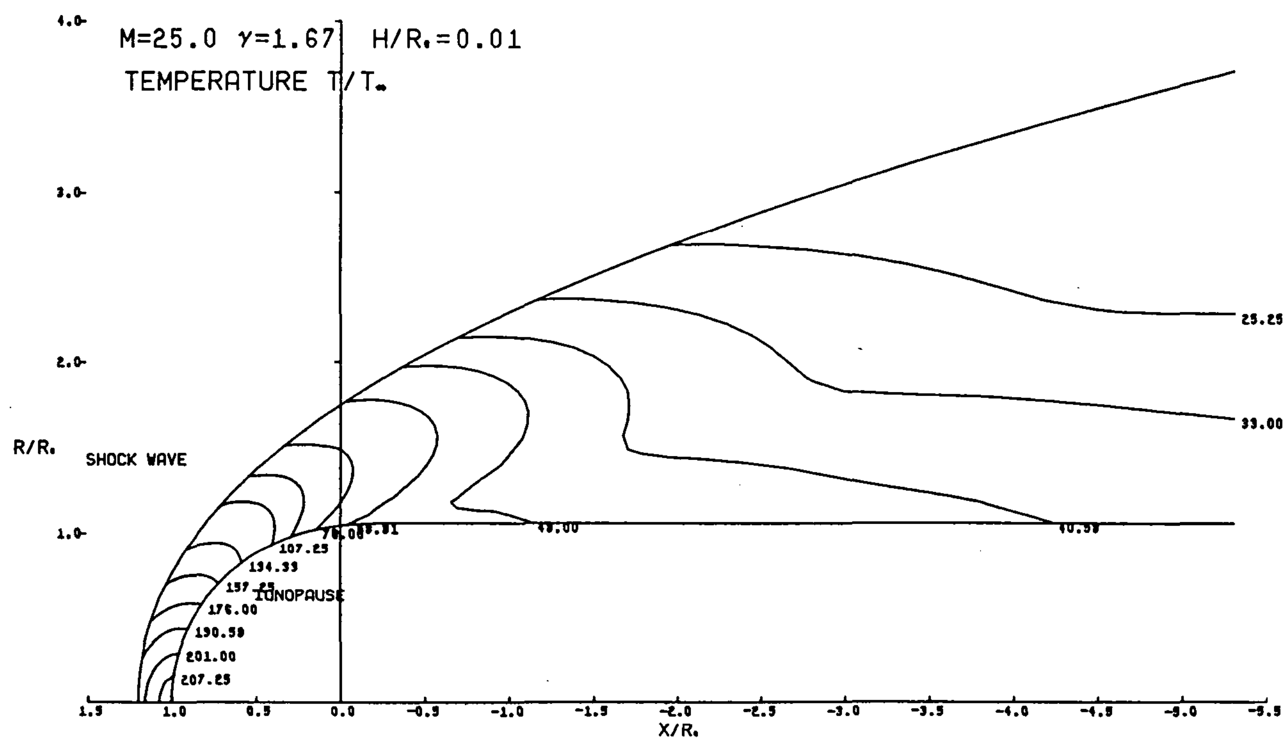

M=12.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

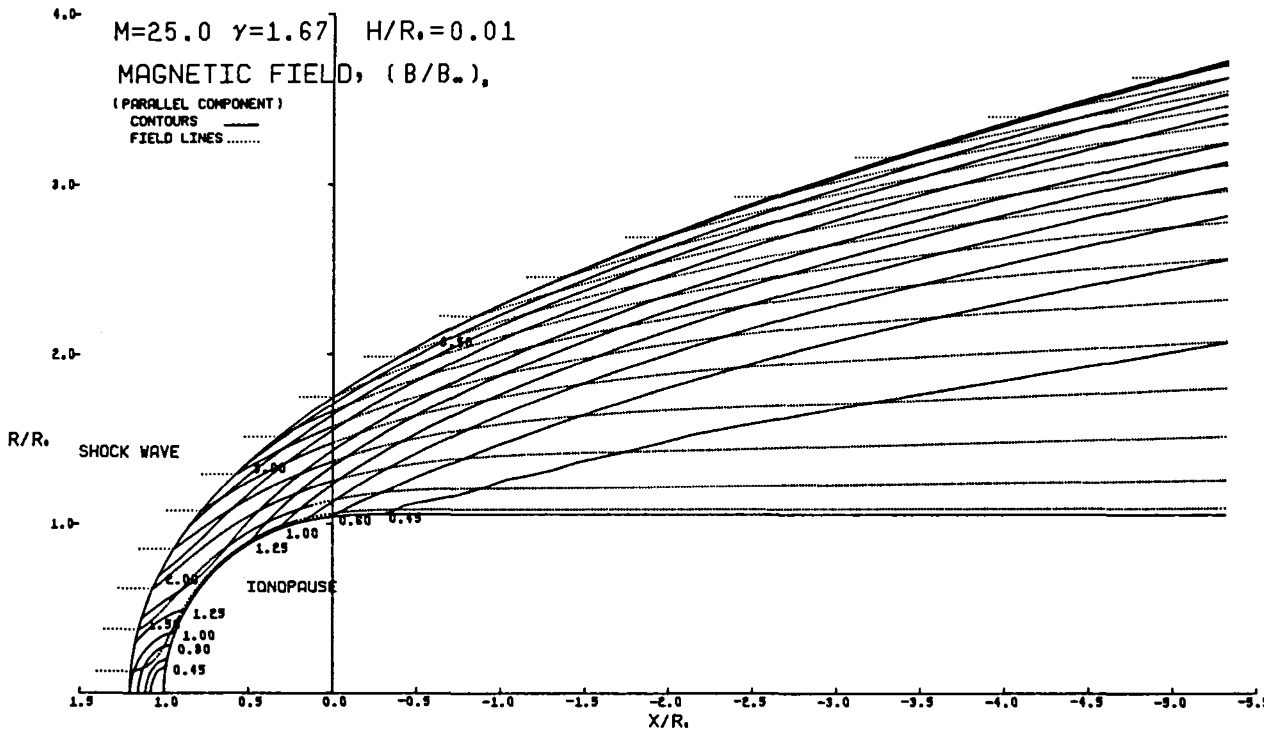

M=12.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

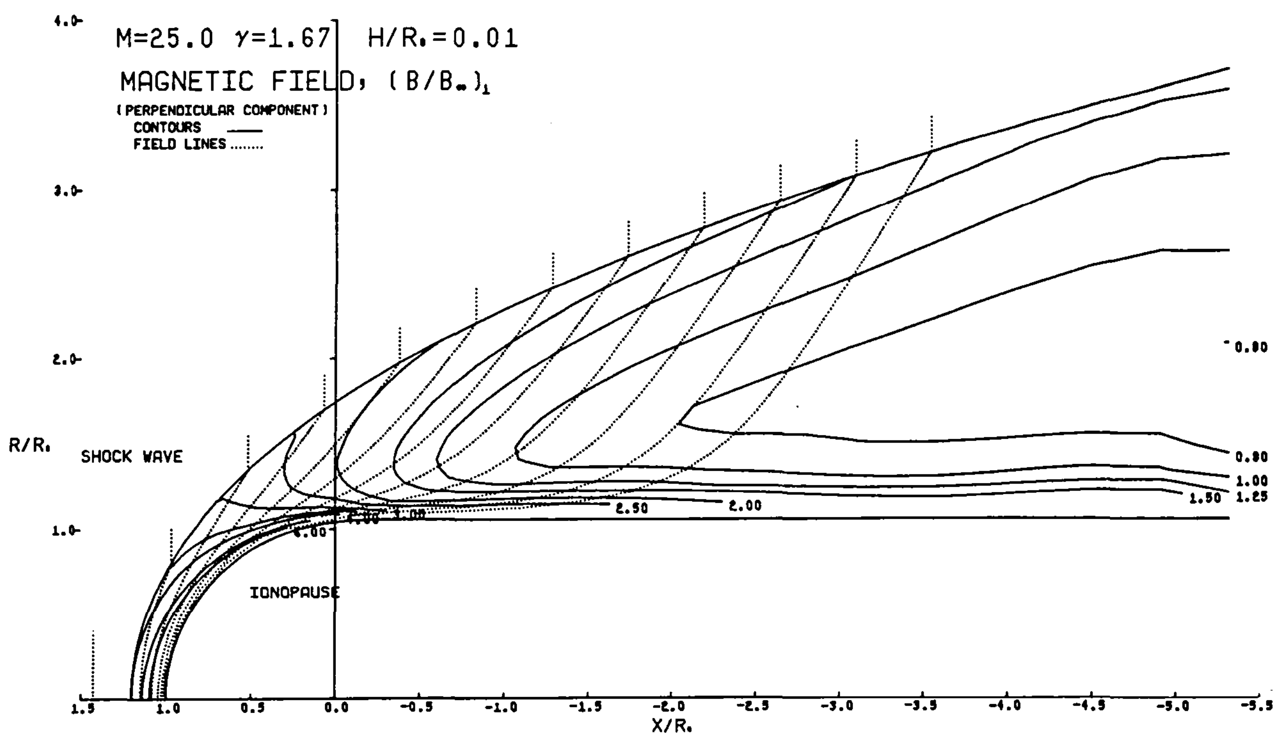

M=25.0 γ=1.67 H/R₀=0.01
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
4.0
3.0
2.0
1.0
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
-4.5
-5.0
-5.5

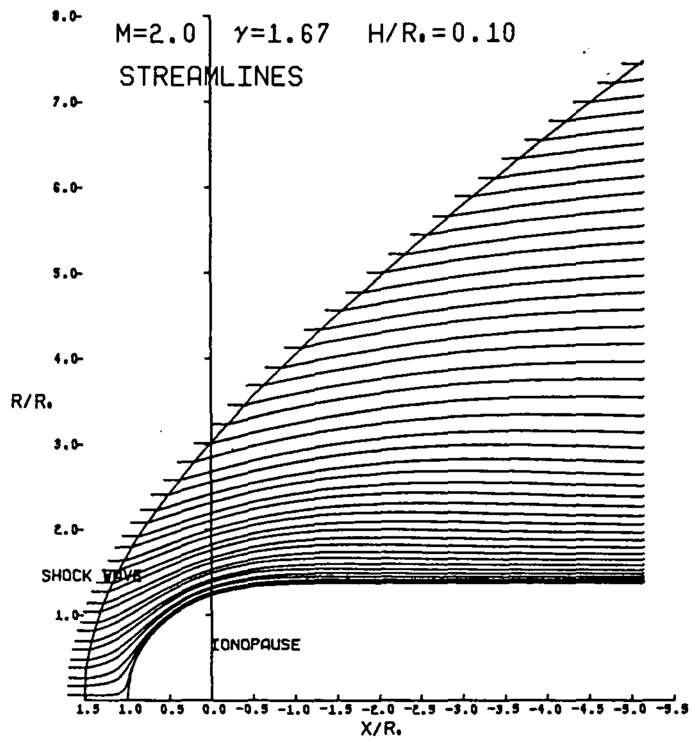

M=25.0 γ=1.67 H/R₀=0.01
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.94
0.92
0.90
0.70
0.60
0.50
0.40
0.30
0.20
0.10
4.0
3.0
2.0
1.0
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
-4.5
-5.0
-5.5

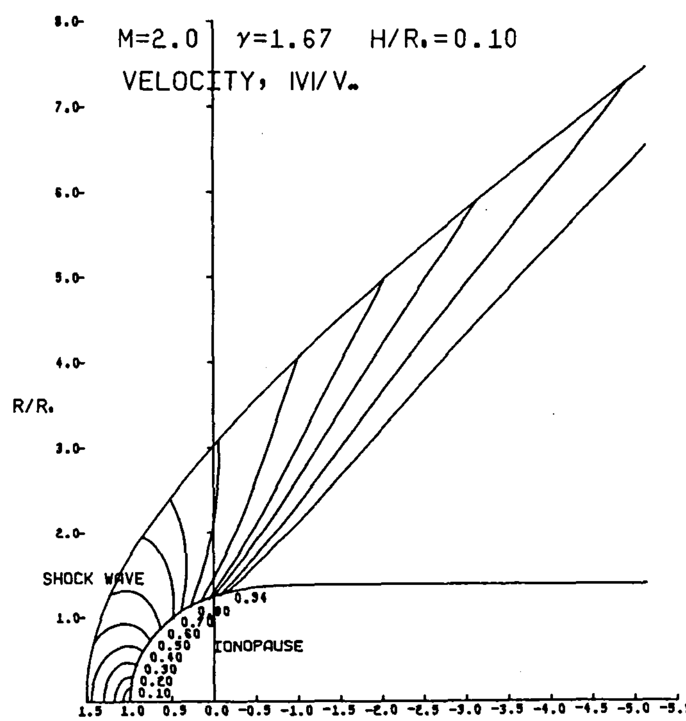

M=25.0 γ=1.67 H/R₀=0.01
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

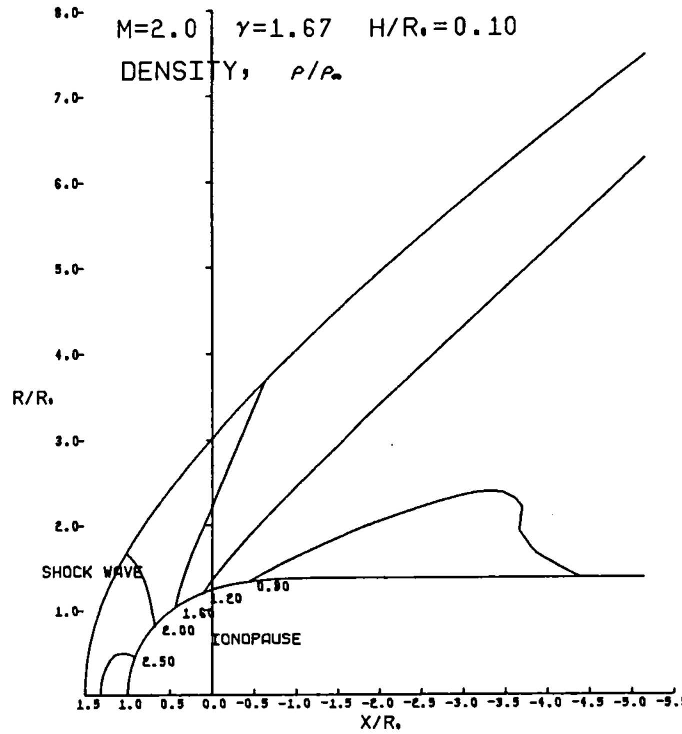

M=25.0 γ=1.67 H/R₀=0.01
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
25.25
33.00
R/R₀
X/R₀


M=25.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.50
3.00
1.00
1.25
0.80
0.45
2.00
1.25
1.50
1.00
0.80
0.45

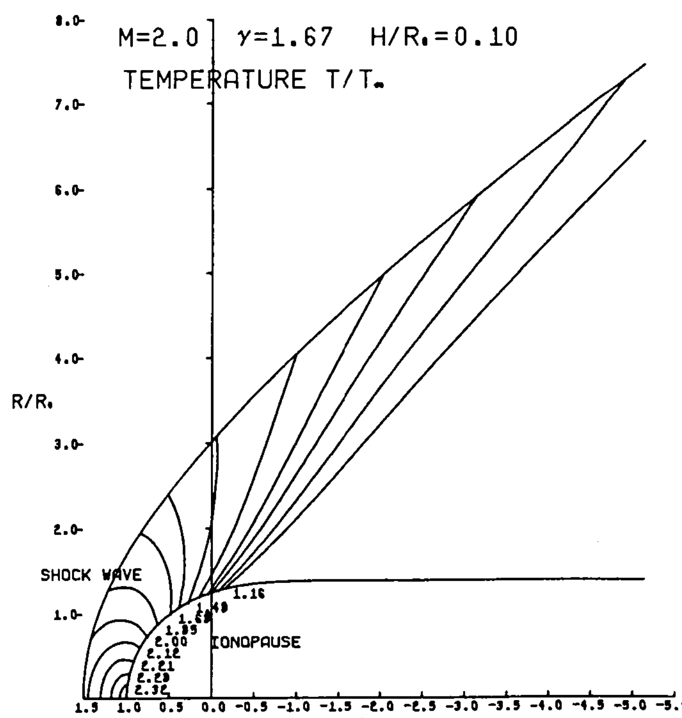

M=25.0 γ=1.67 H/R₀=0.01
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
0.80
1.00
1.50
1.25
2.50
2.00
4.00


M=2.0 γ=1.67 H/R₀=0.10
STREAMLINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE

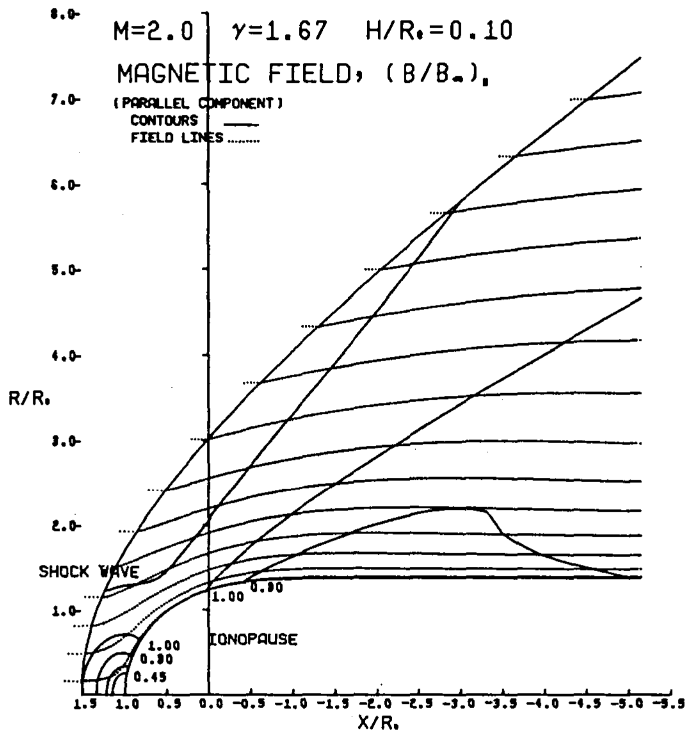

M=2.0 γ=1.67 H/R₀=0.10
VELOCITY, |V|/V∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.94
0.90
0.70
0.60
0.50
0.40
0.30
0.20
0.10

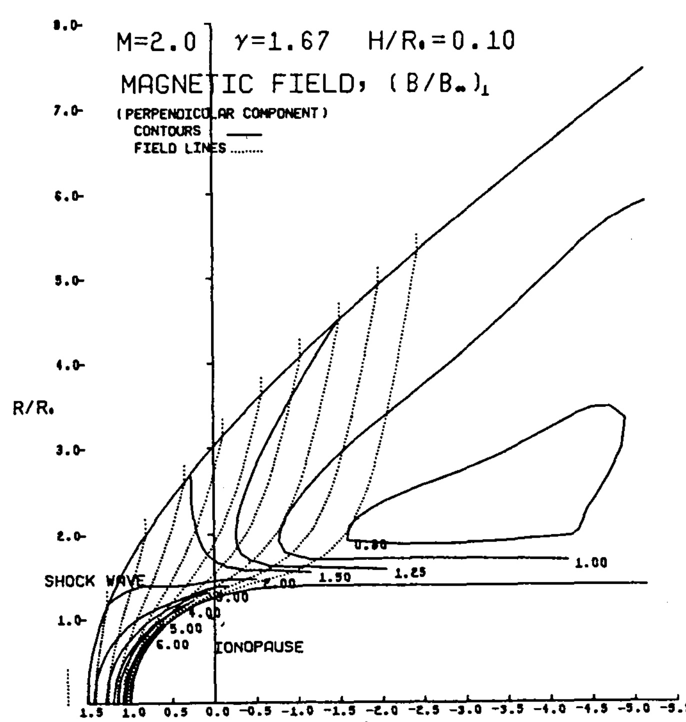


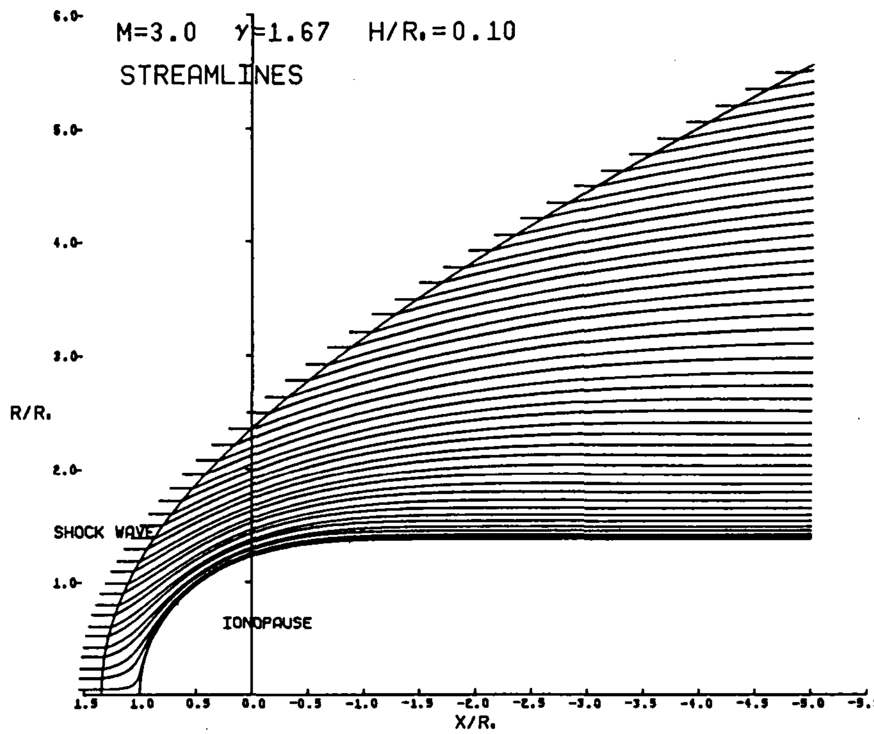
M=2.0 γ=1.67 H/R₀=0.10
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
2.50
2.00
1.60
1.20
0.90

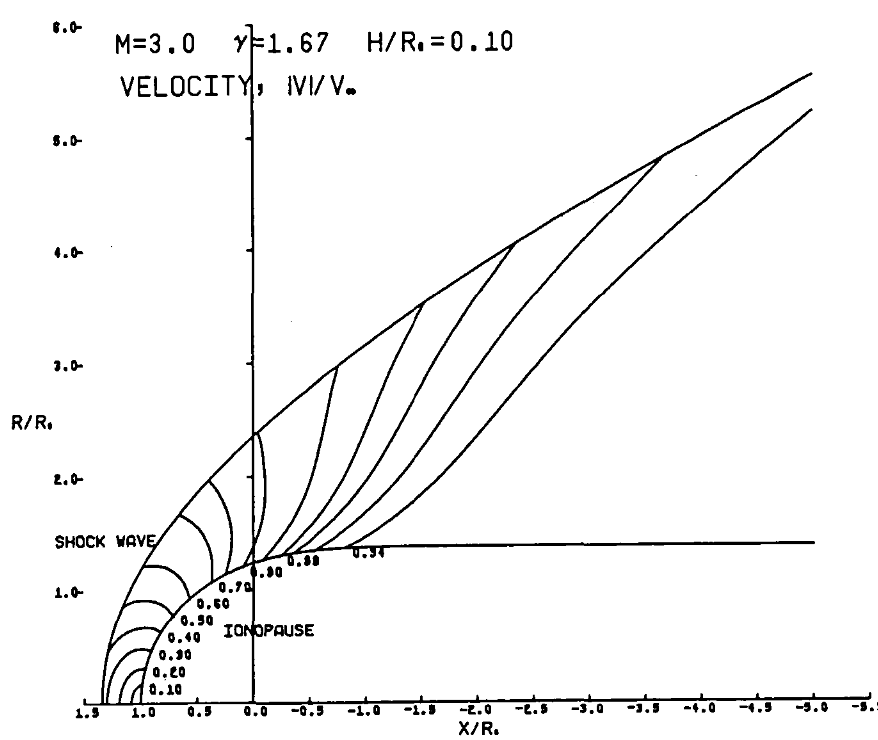


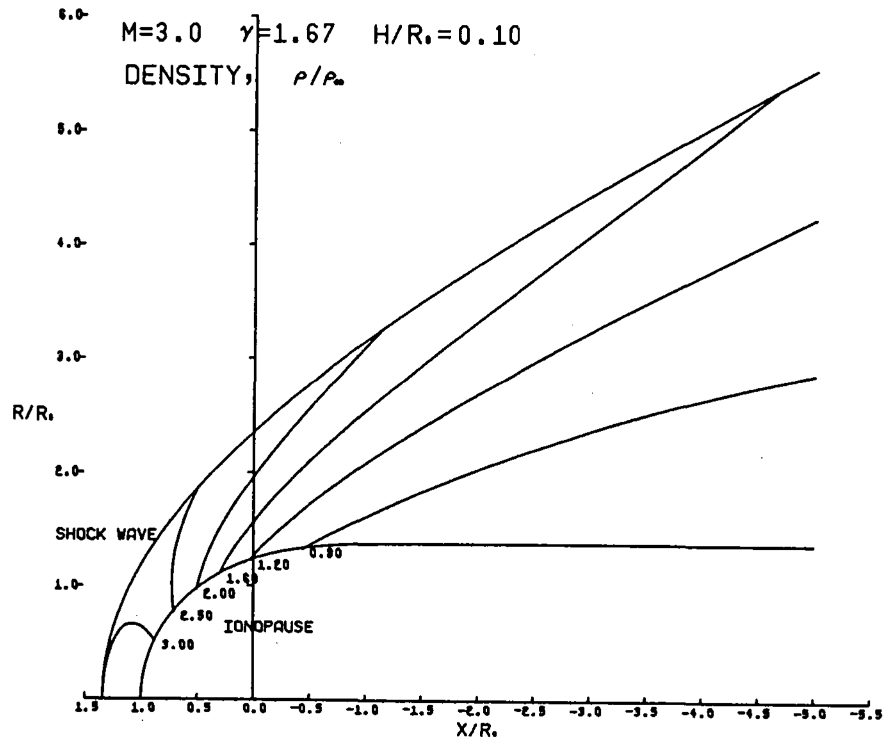
M=2.0 γ=1.67 H/R₀=0.10
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.16
1.49
1.69
1.85
2.00
2.12
2.21
2.29
2.32

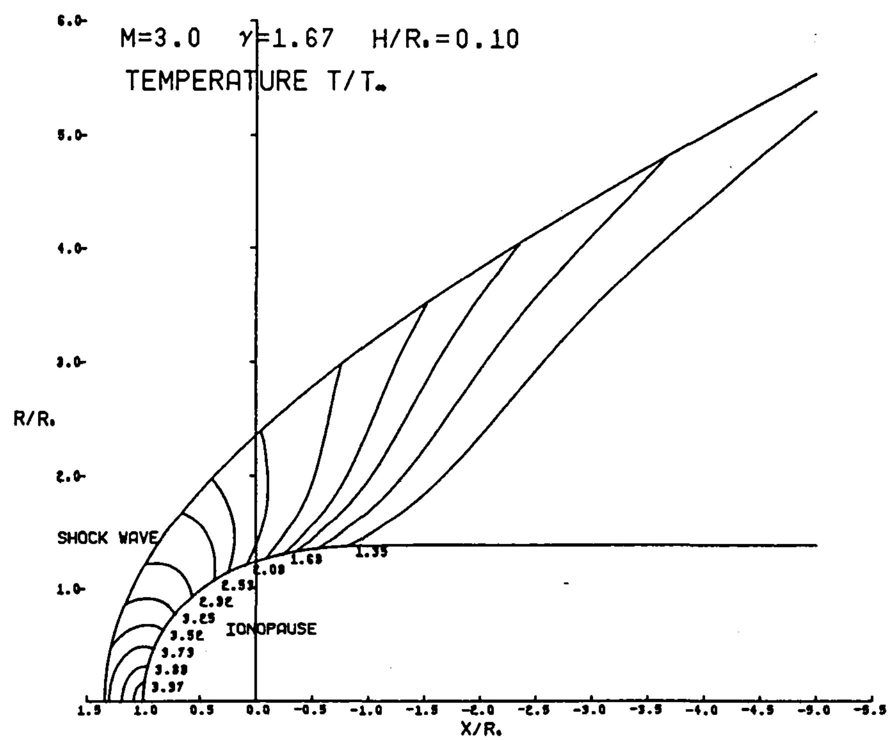

M=2.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.00
0.90
0.45

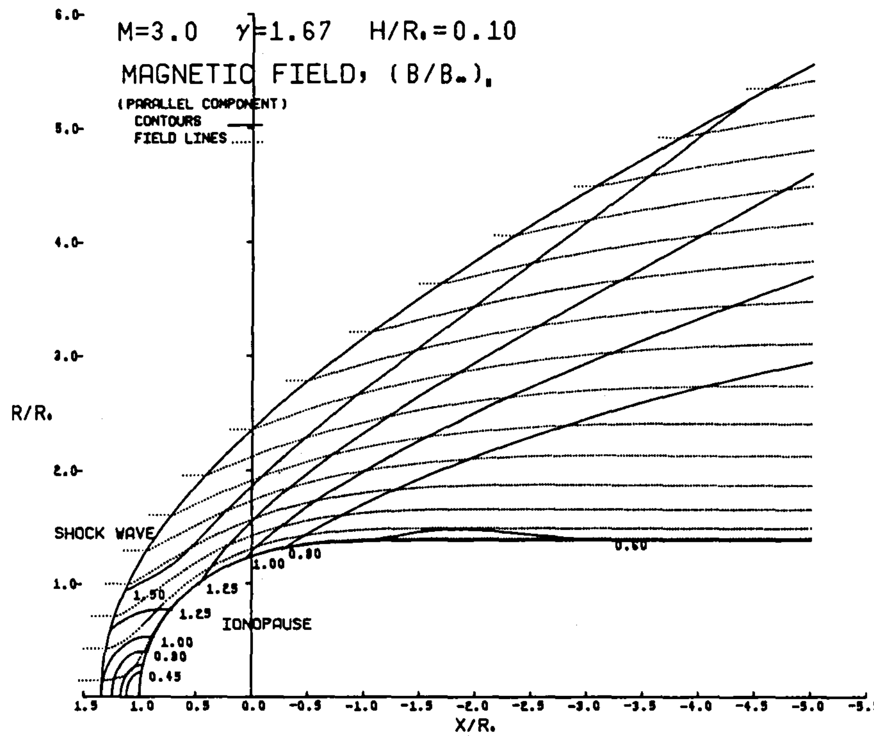

M=2.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.90
1.00
1.25
1.50
2.00
3.00
4.00
5.00
6.00

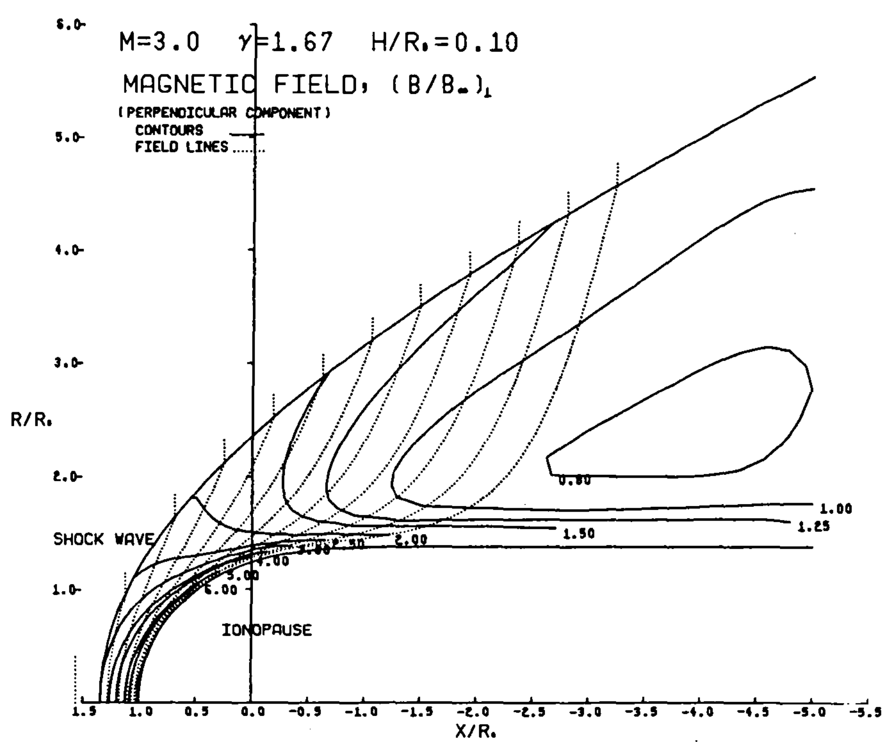

M=3.0 γ=1.67 H/R_0=0.10
STREAMLINES
R/R_0
X/R_0
SHOCK WAVE
IONOPAUSE
6.0
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

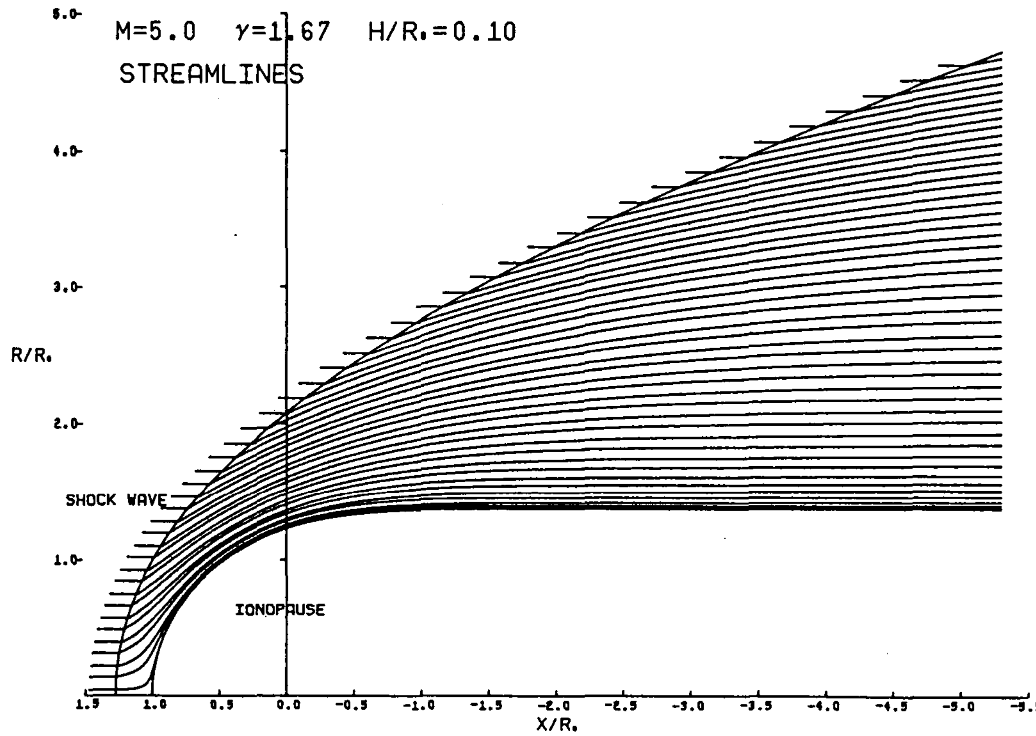

M=3.0 γ=1.67 H/R_0=0.10
VELOCITY, |V|/V_∞
R/R_0
X/R_0
SHOCK WAVE
IONOPAUSE
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.80
0.88
0.94
6.0
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

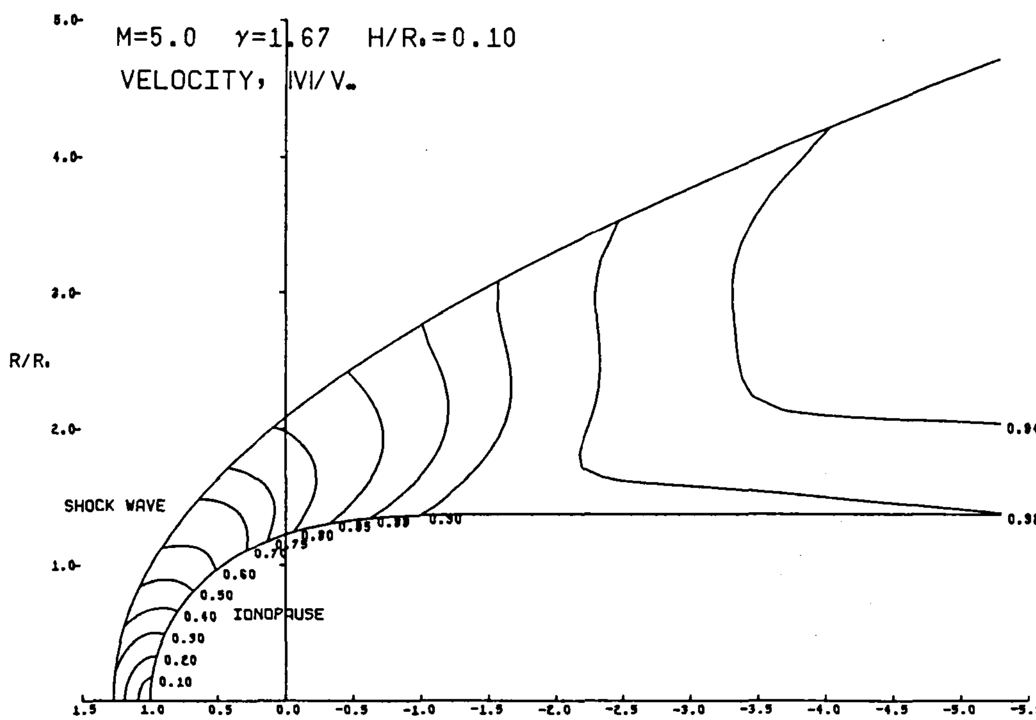

M=3.0 γ=1.67 H/R₀=0.10
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.20
1.60
2.00
2.50
3.00

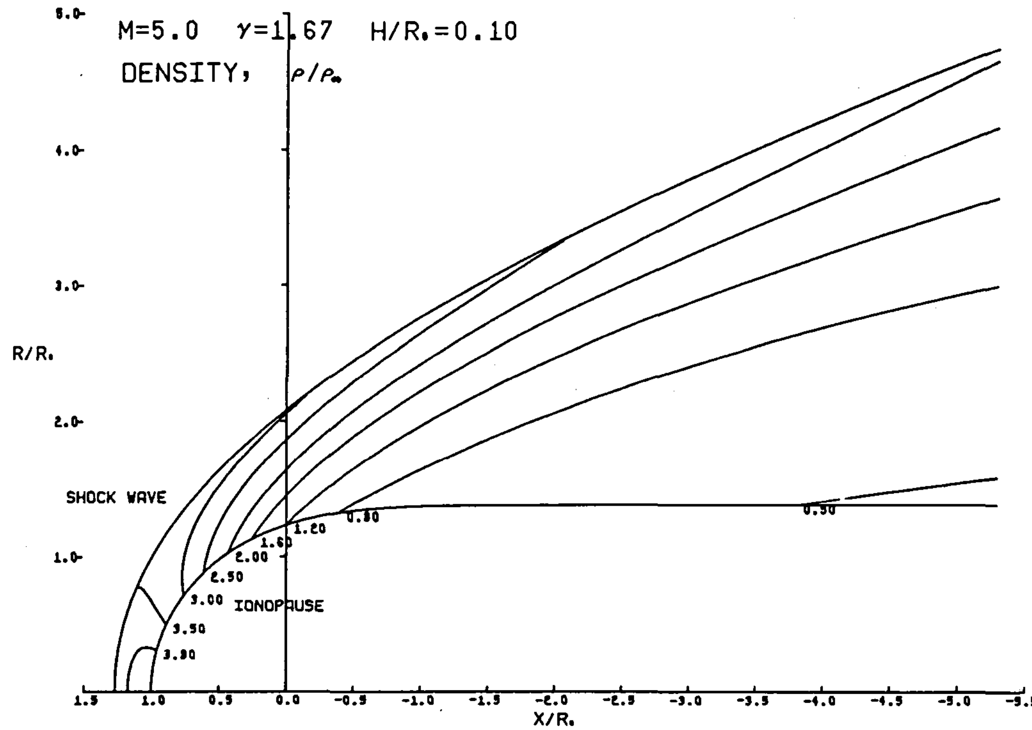

M=3.0 γ=1.67 H/R₀=0.10
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.35
1.63
2.08
2.53
2.92
3.25
3.52
3.73
3.88
3.97

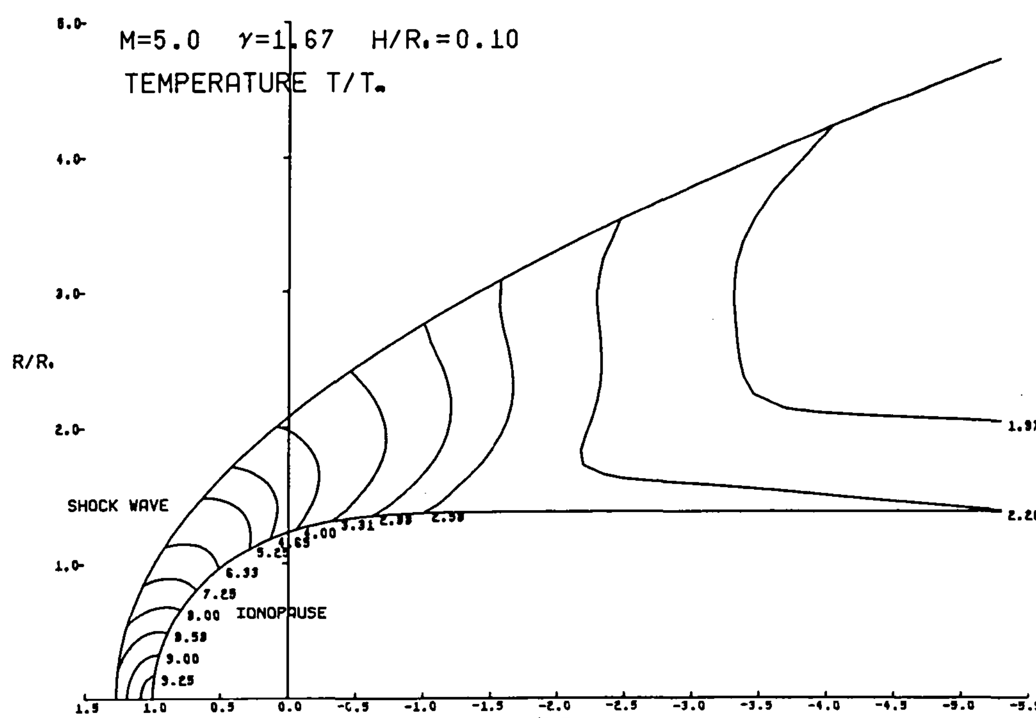

M=3.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
1.50
1.25
1.00
0.80
0.45
0.60

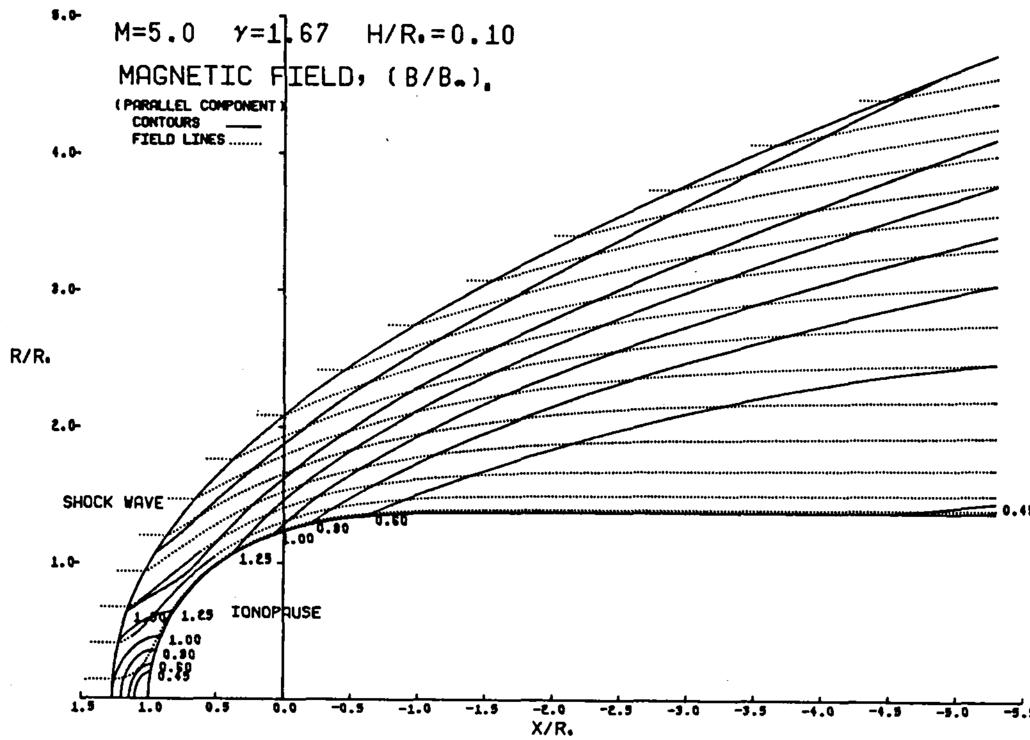

M=3.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.00
1.25
1.50
2.00
4.00
5.00
6.00

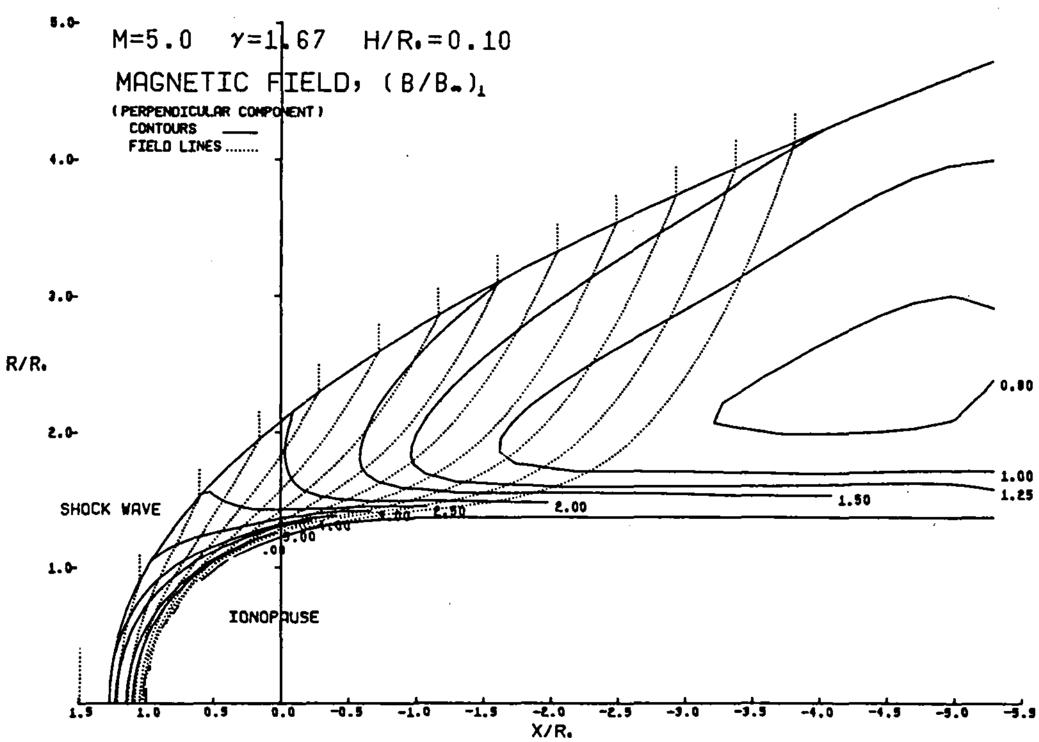

M=5.0 γ=1.67 H/R₀=0.10
STREAMLINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

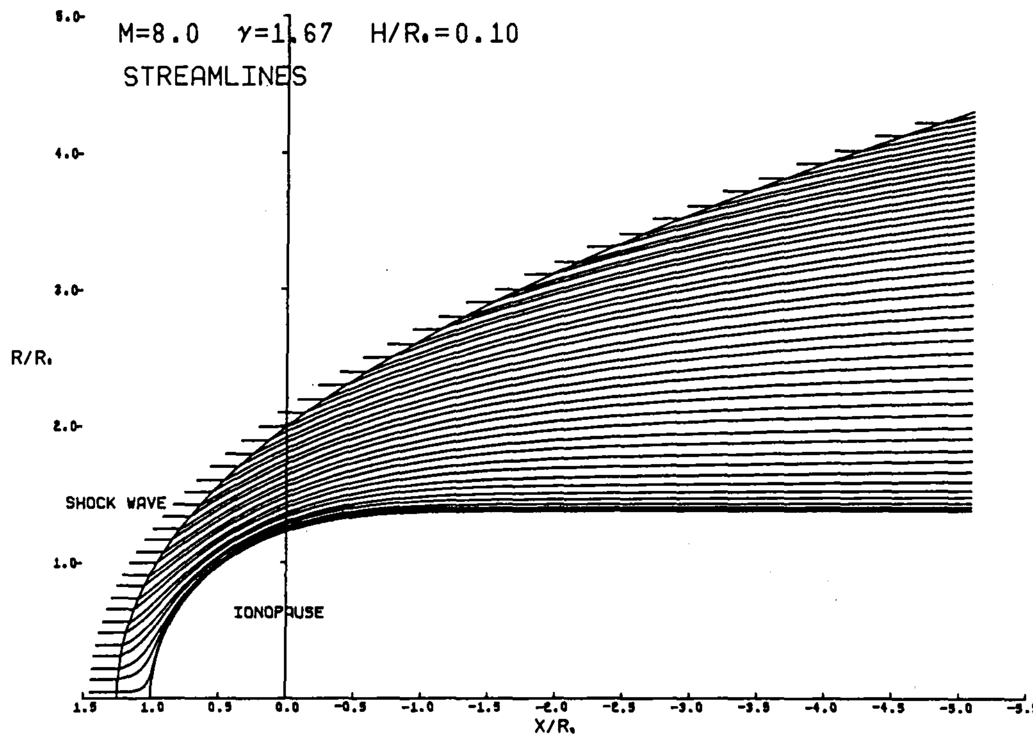

M=5.0 γ=1.67 H/R₀=0.10
VELOCITY, |V|/V∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.75
0.80
0.85
0.90
0.94
0.92
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

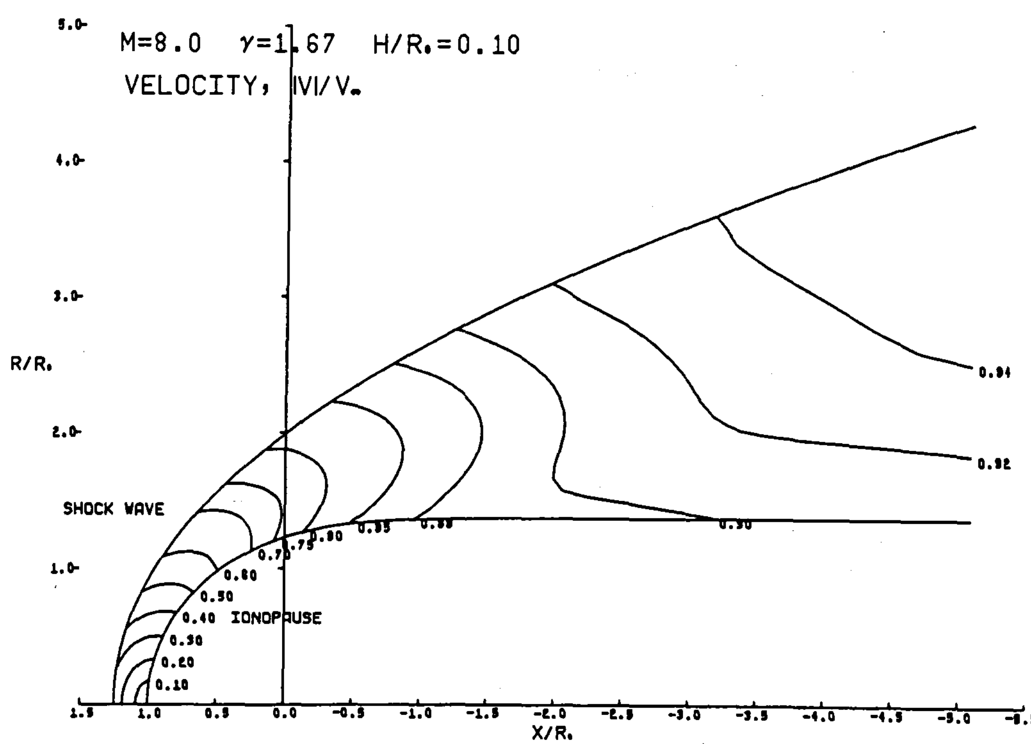

M=5.0 γ=1.67 H/R₀=0.10
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80
0.50

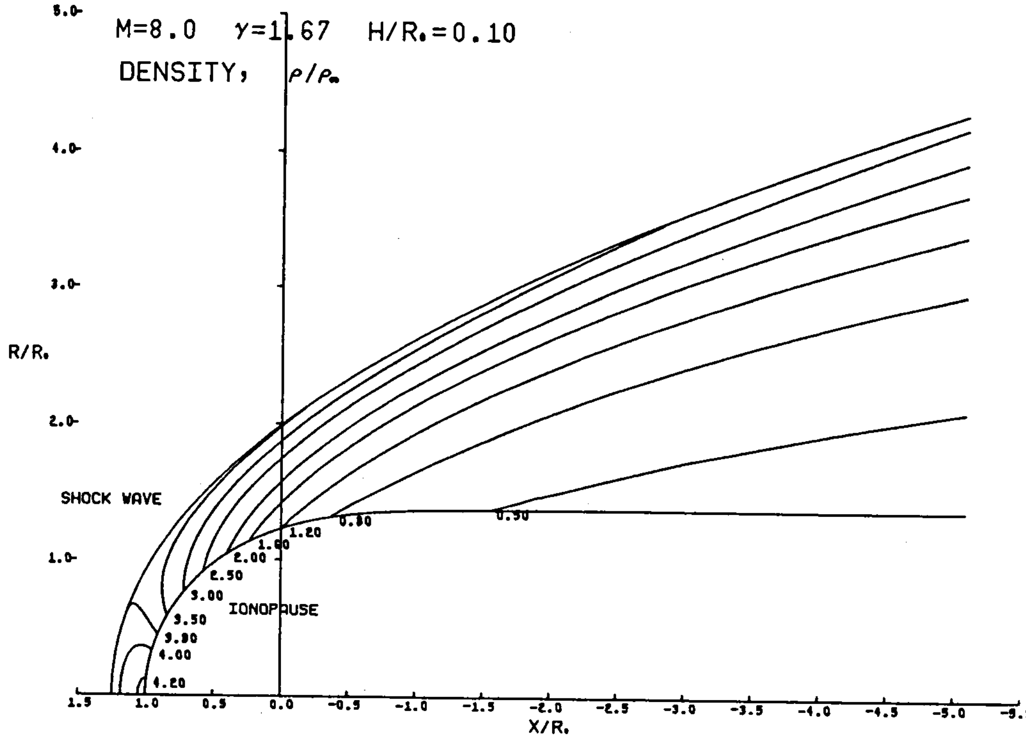

M=5.0 γ=1.67 H/R₀=0.10
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.97
3.31
4.00
4.65
5.25
6.33
7.25
8.00
9.00
9.25


M=5.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.45
0.80
0.60
1.00
1.25

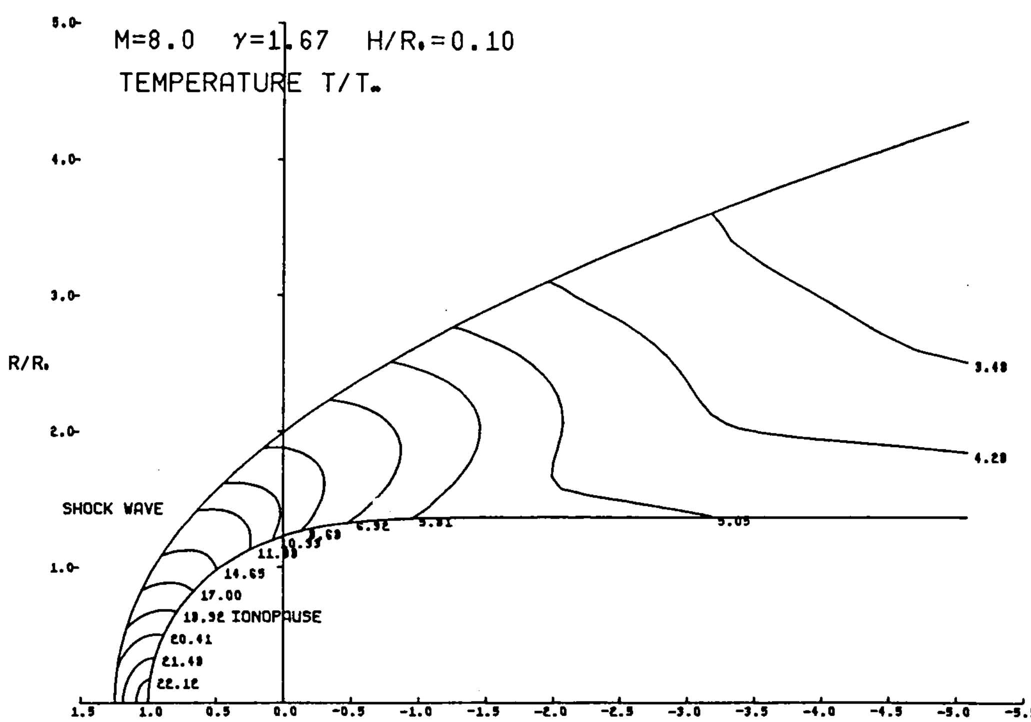

M=5.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.00
1.25
1.50
2.00
2.50


M=8.0 γ=1.67 H/R₀=0.10
STREAMLINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE

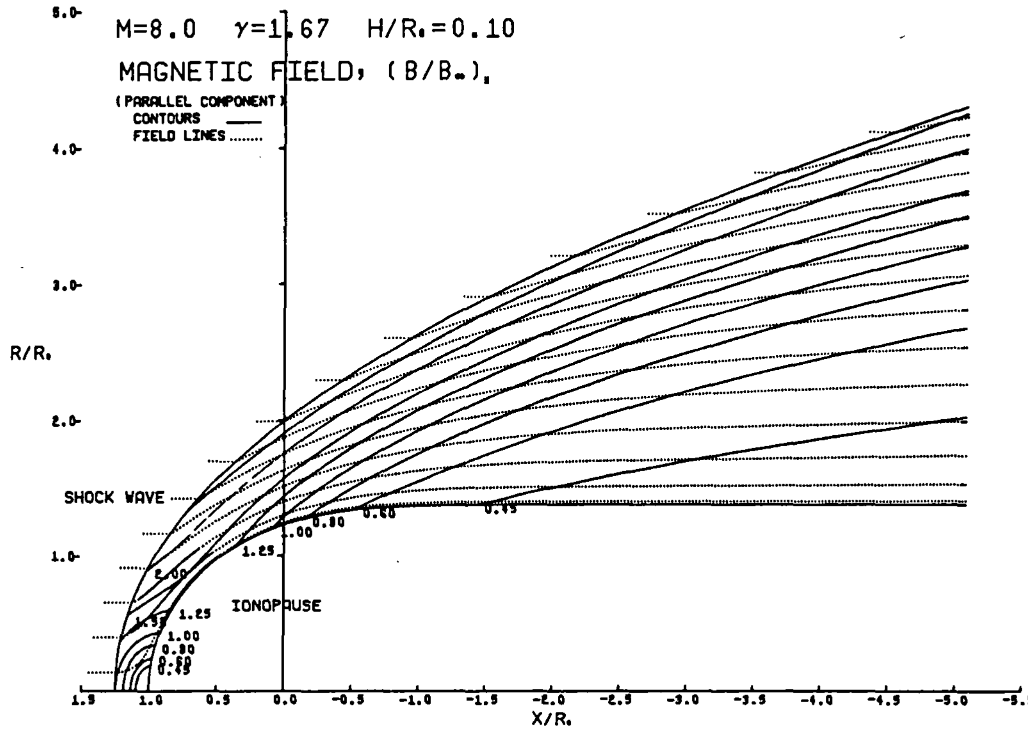

M=8.0 γ=1.67 H/R₀=0.10
VELOCITY, |V|/V∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.94
0.92
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10

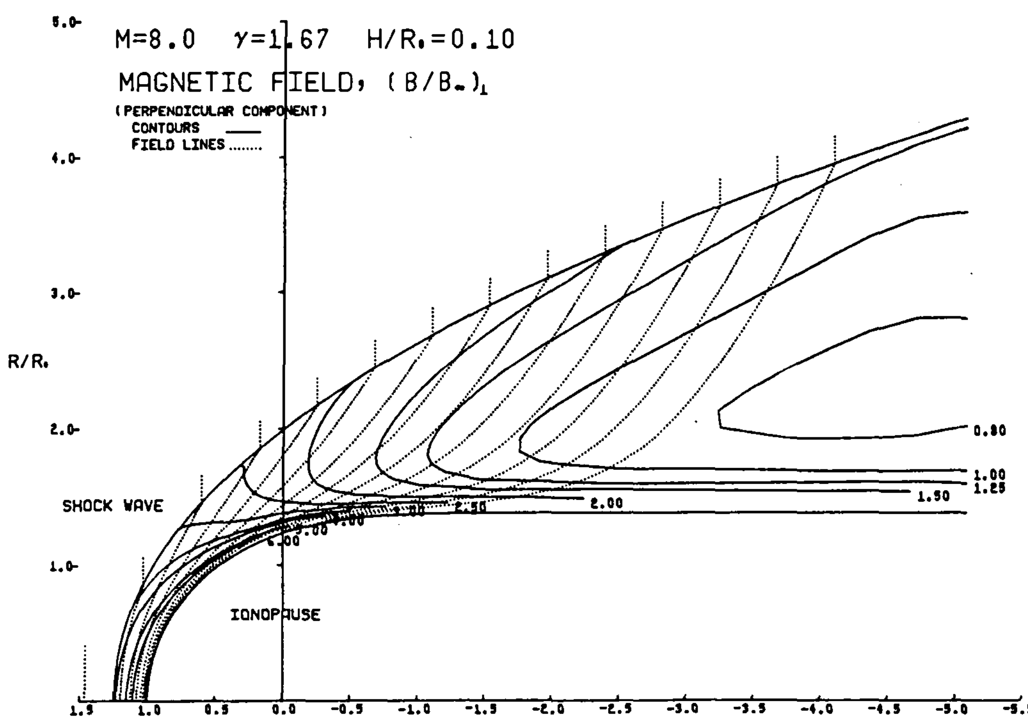


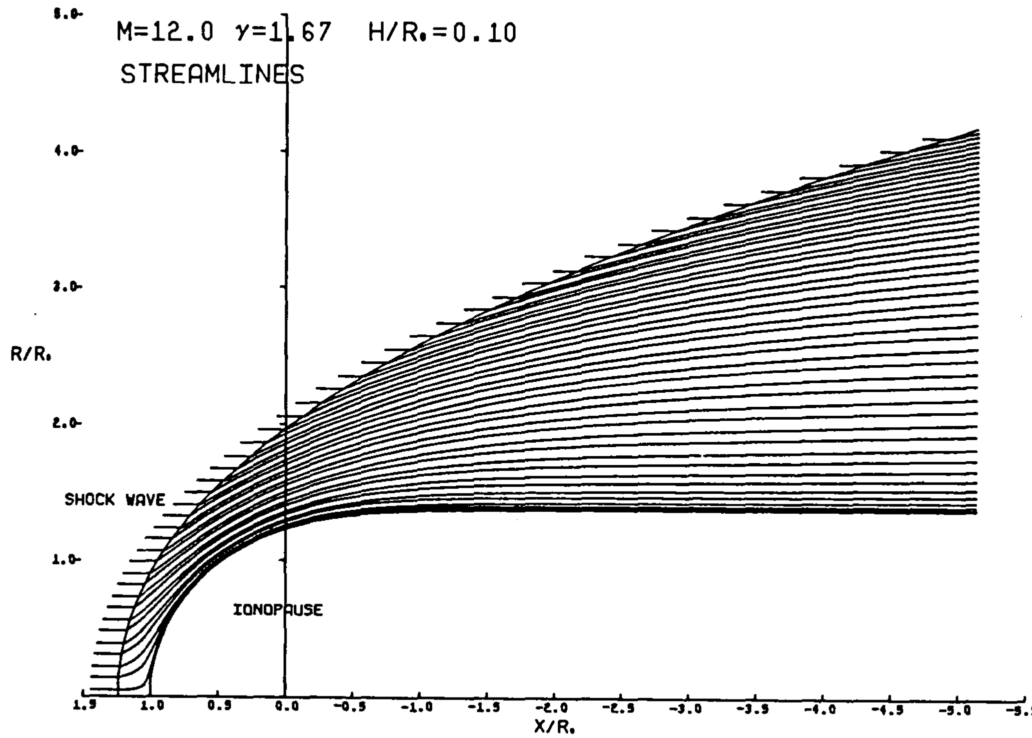
M=8.0 γ=1.67 H/R₀=0.10
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
0.90
1.20
1.60
2.00
2.50
3.00
3.50
3.90
4.00
4.20

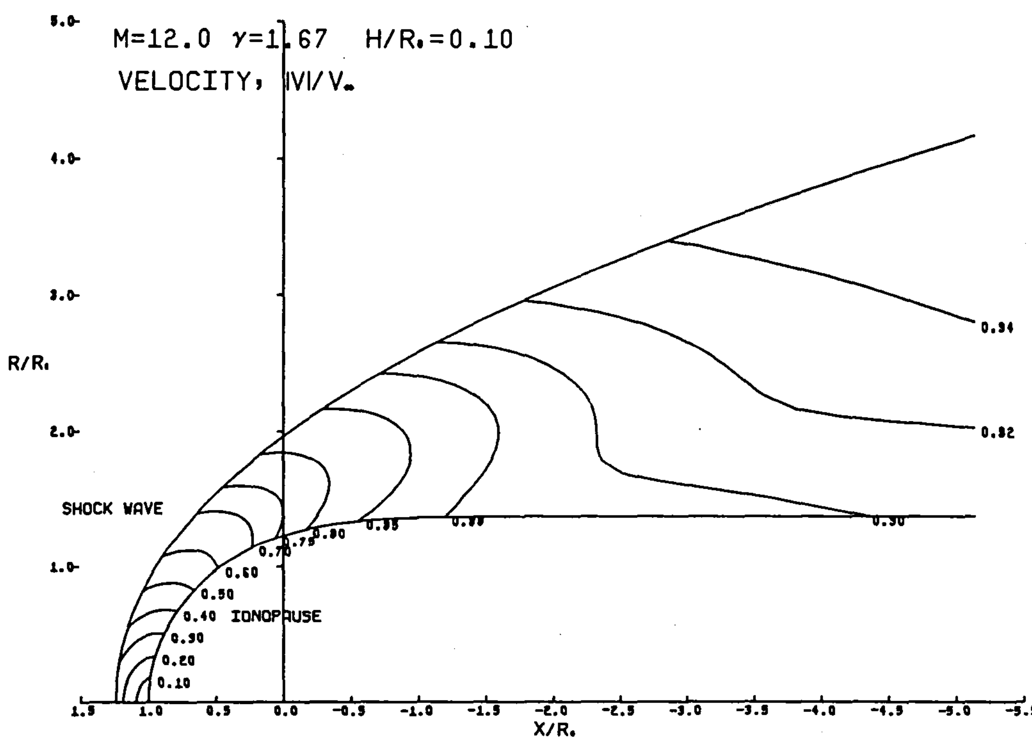


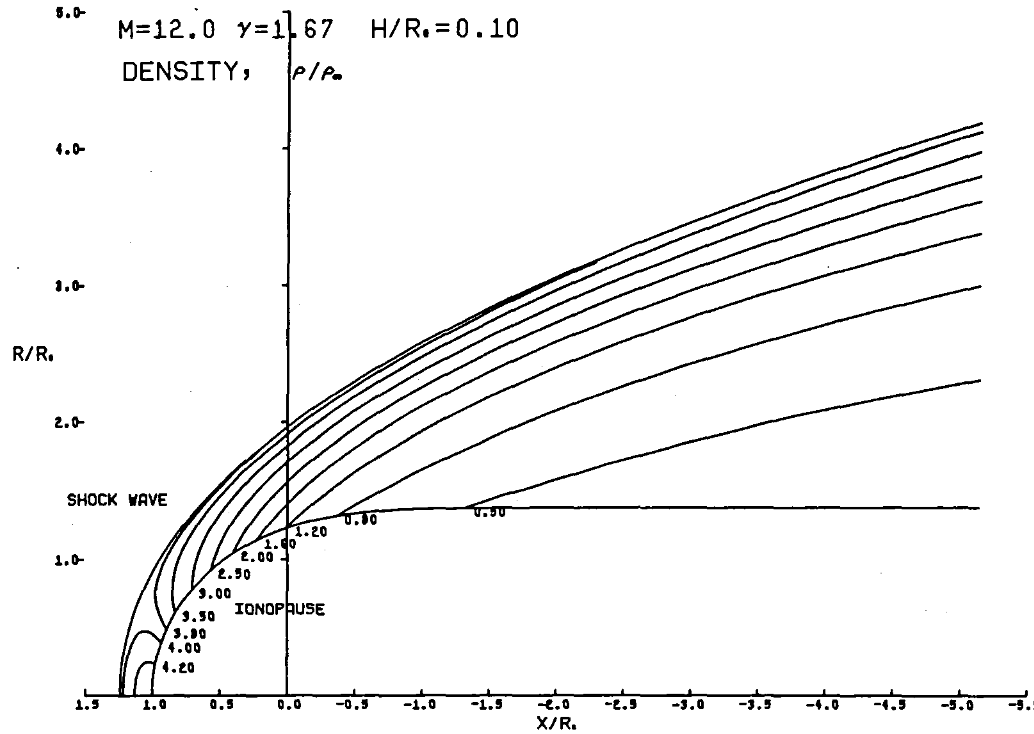
M=8.0 γ=1.67 H/R₀=0.10
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
3.48
4.28
14.65
17.00
20.41
21.49
22.12

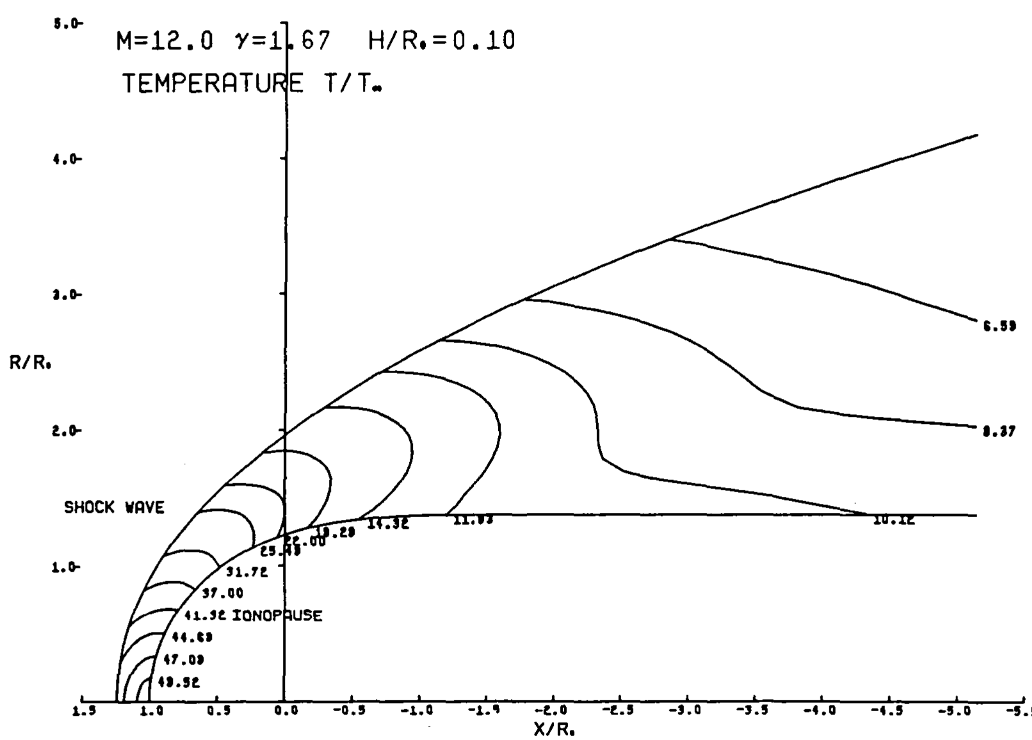

M=8.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

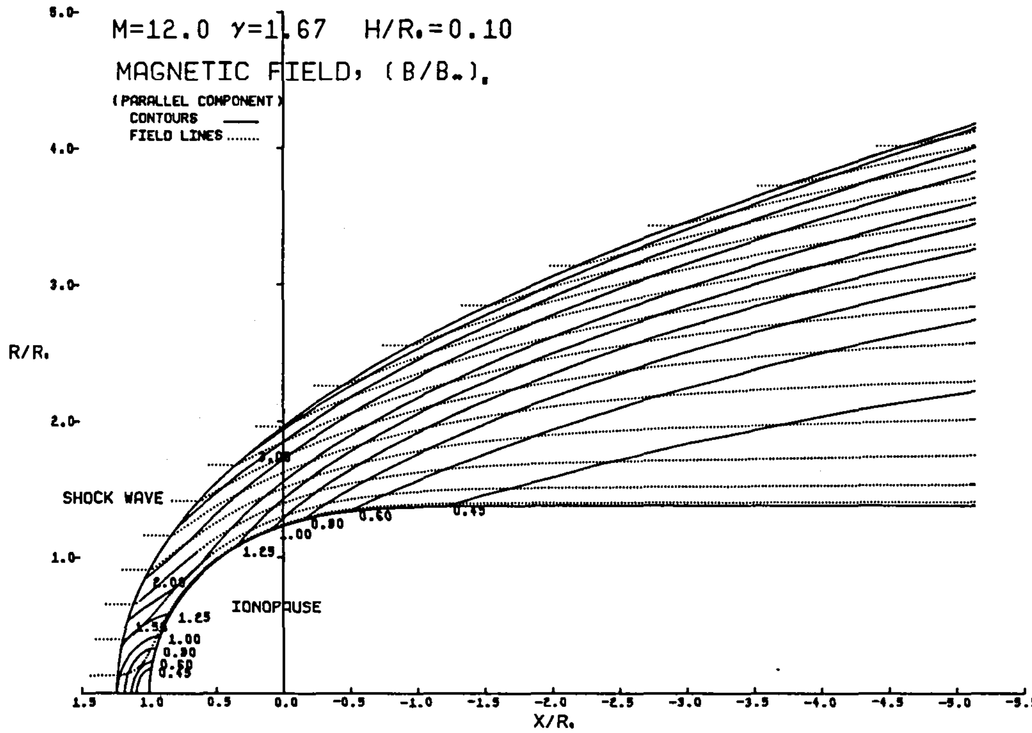

M=8.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀


M=12.0 γ=1.67 H/R₀=0.10
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

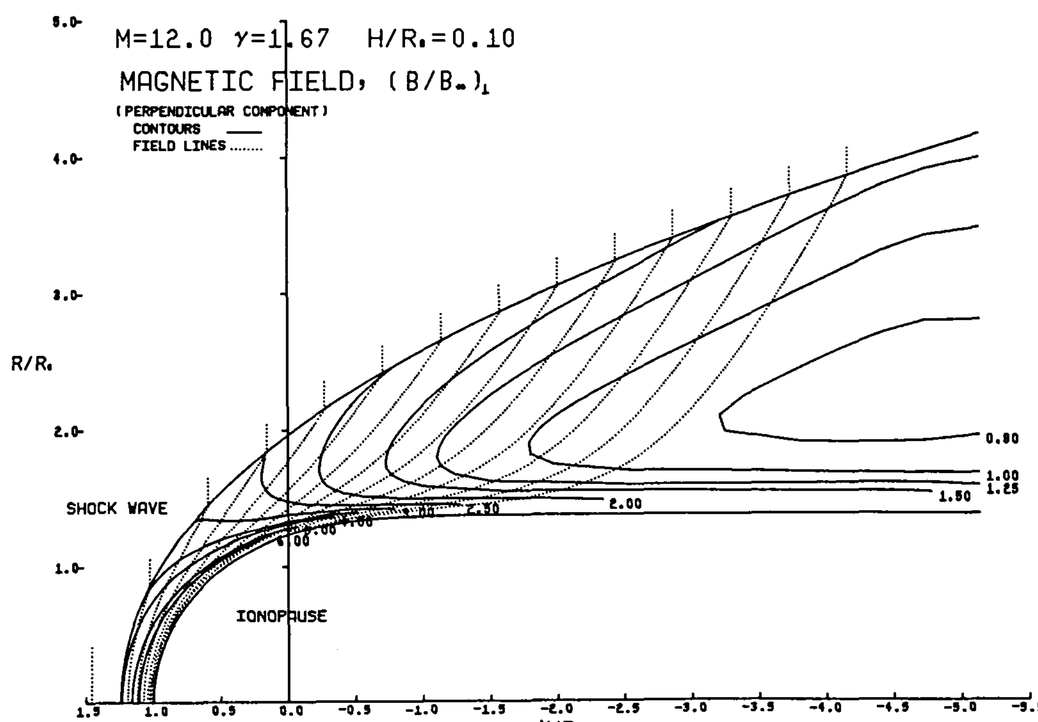

M=12.0 γ=1.67 H/R₀=0.10
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.75
0.80
0.85
0.89
0.90
0.92
0.94
X/R₀


M=12.0 γ=1.67 H/R₀=0.10
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE

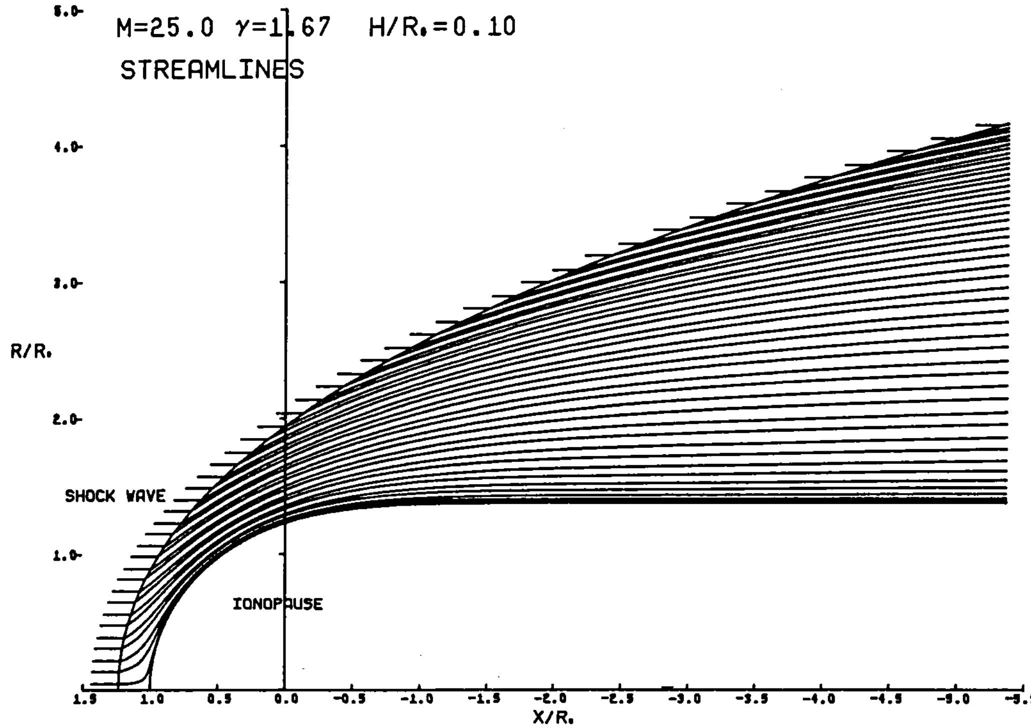

M=12.0 γ=1.67 H/R₀=0.10
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE

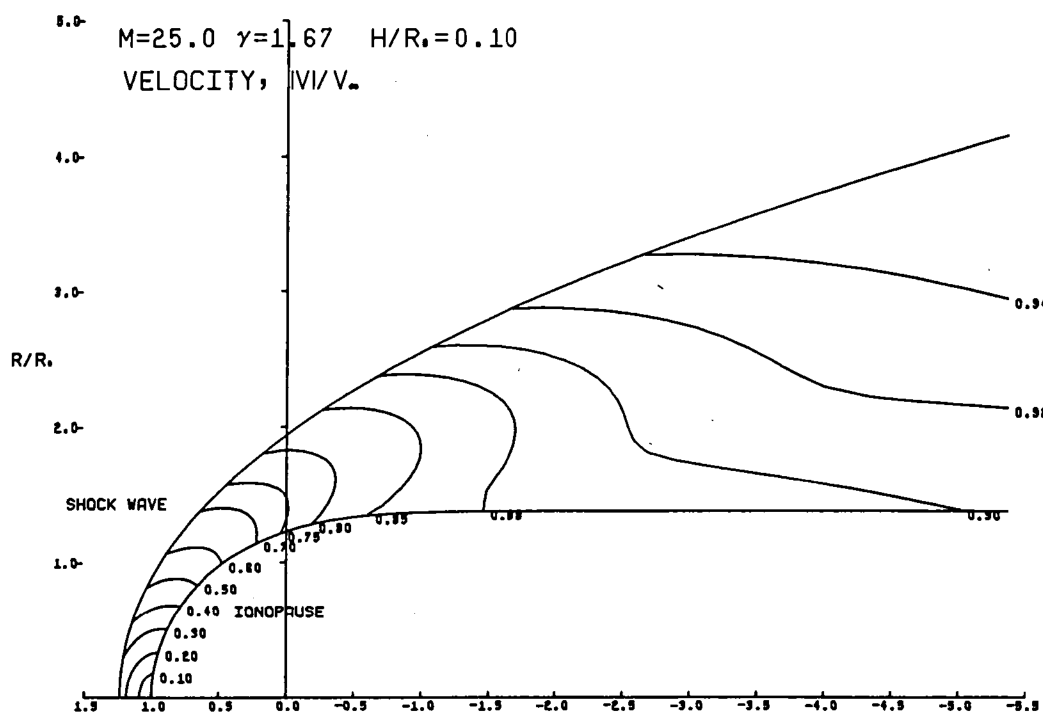


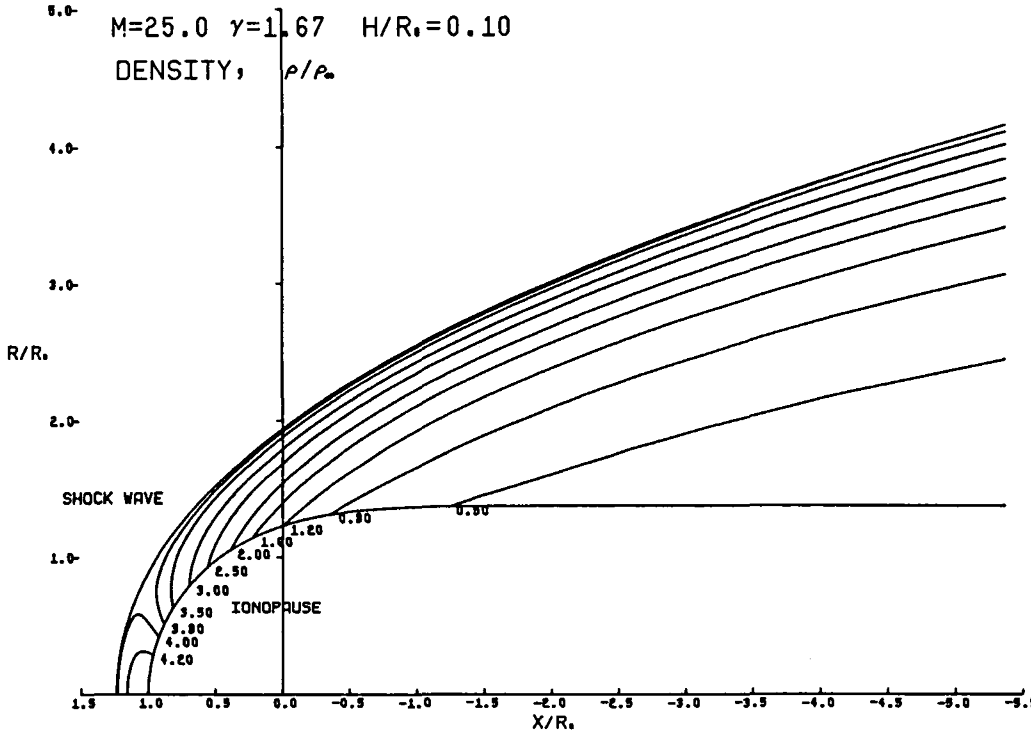
M=12.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
2.00
1.25
1.00
0.90
0.60
0.45
1.50

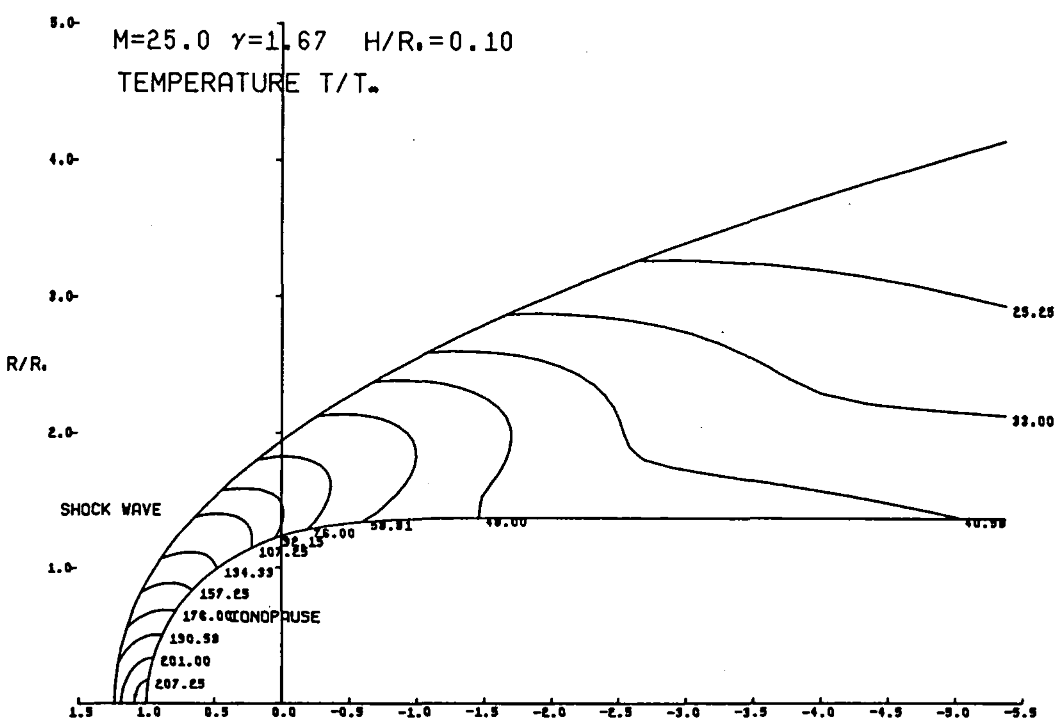


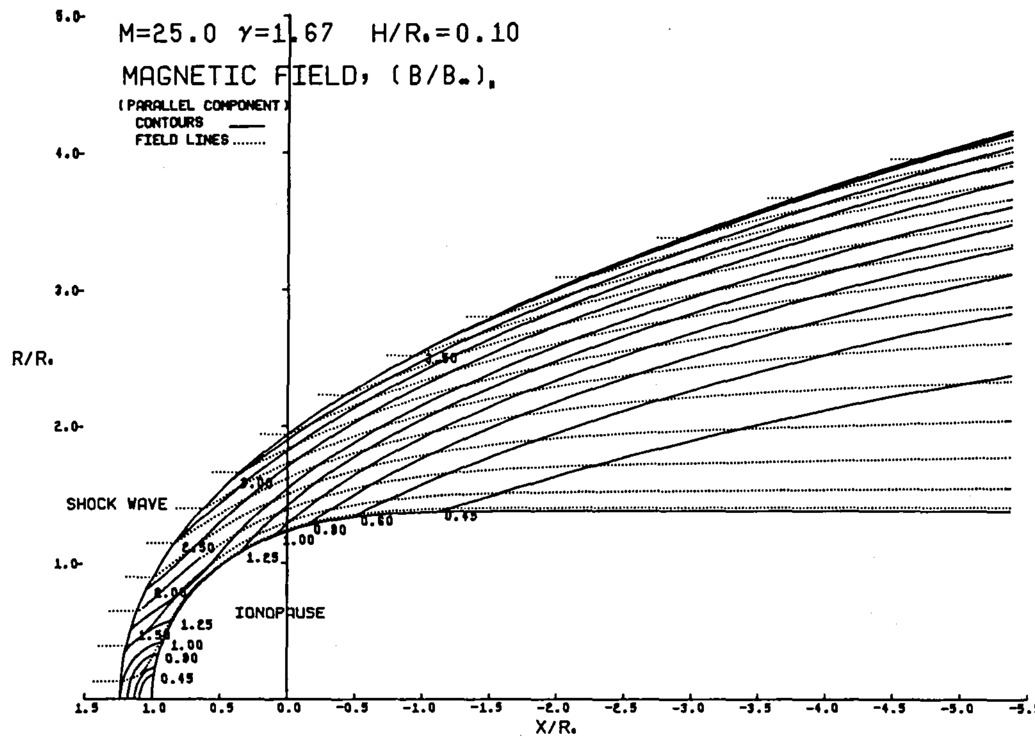
M=12.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.00
1.25
1.50
2.00
2.50
3.00
4.00
5.00
6.00

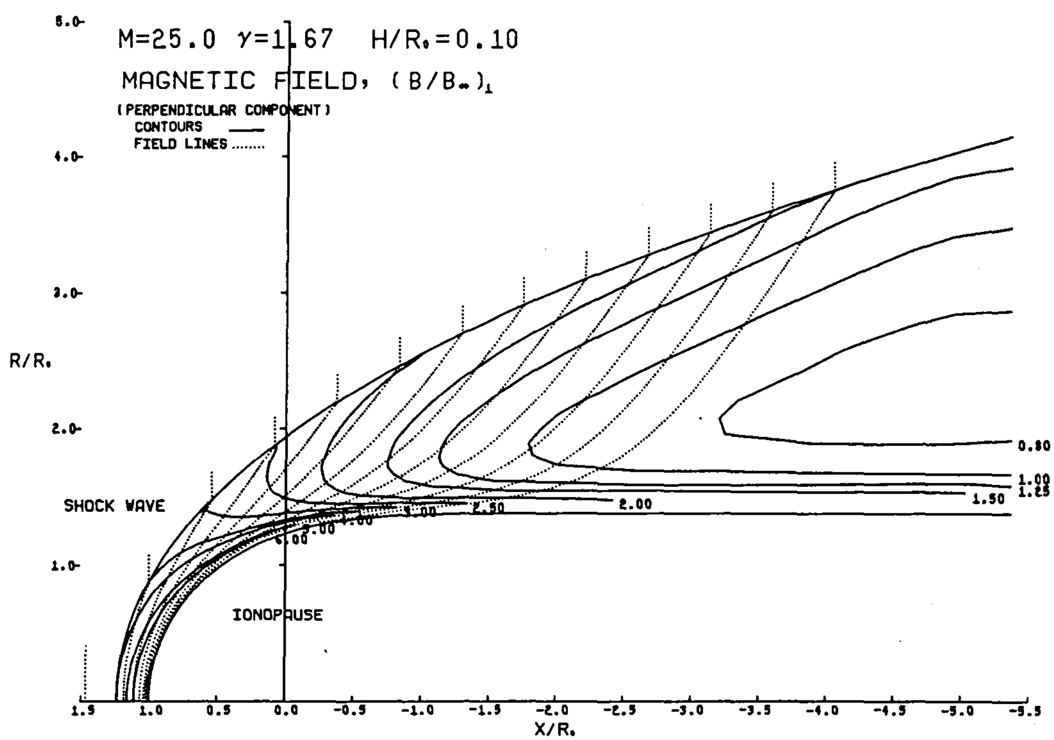

M=25.0 γ=1.67 H/R₀=0.10
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

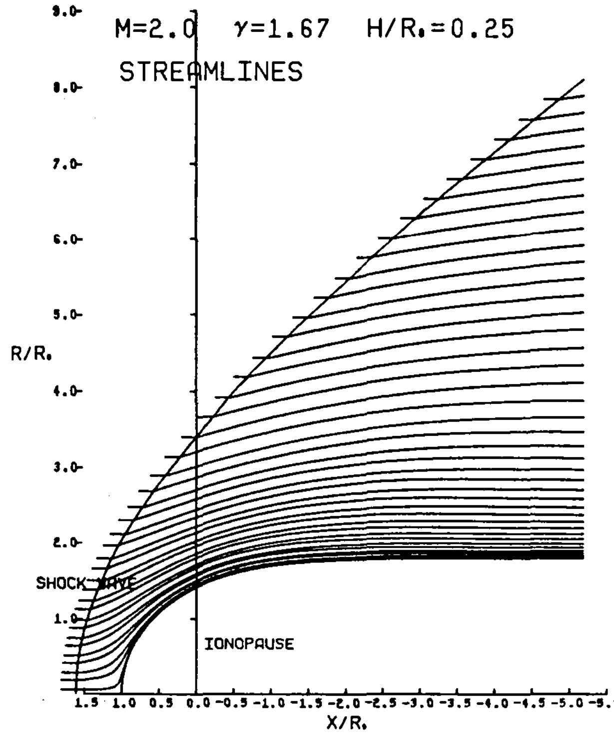

M=25.0 γ=1.67 H/R₀=0.10
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
0.94
0.92
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.75
0.80
X/R₀


M=25.0 γ=1.67 H/R₀=0.10
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.20
1.00
2.00
2.50
3.00
3.50
3.80
4.00
4.20

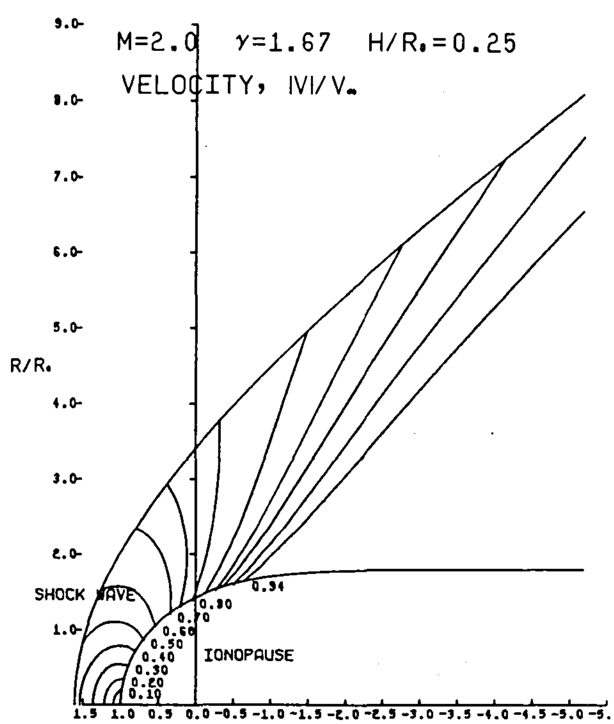

M=25.0 γ=1.67 H/R₀=0.10
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
25.25
33.00
76.00
107.25
134.33
157.25
176.00
190.58
201.00
207.25


M=25.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
3.50
3.00
2.50
2.00
1.50
1.25
1.00
0.80
0.60
0.45

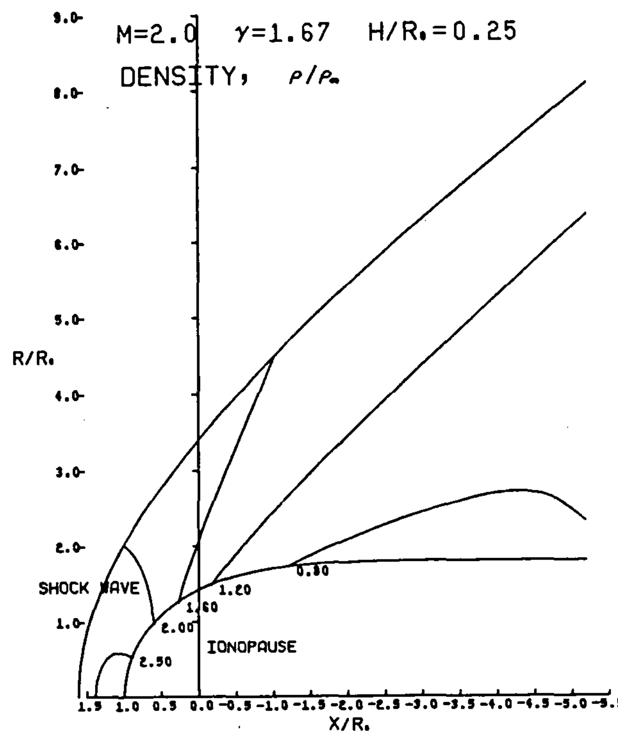

M=25.0 γ=1.67 H/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.00
1.25
1.50
2.00
2.50
3.00
4.00
6.00

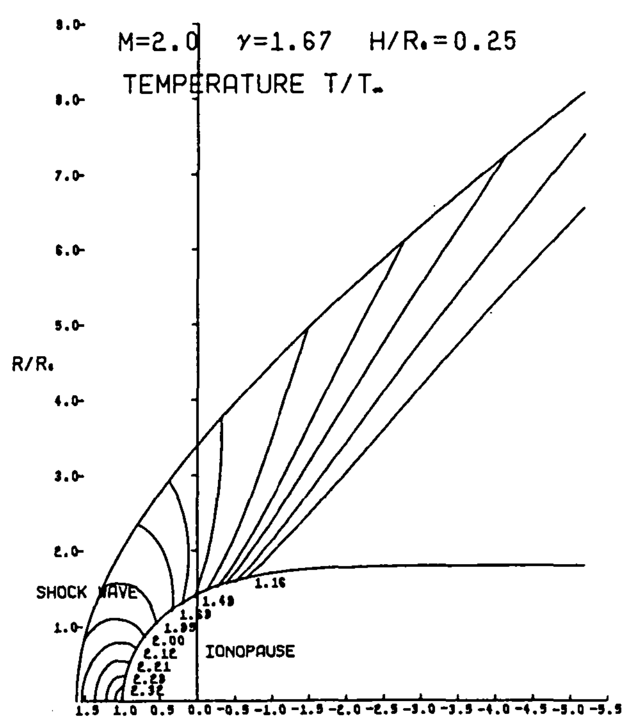


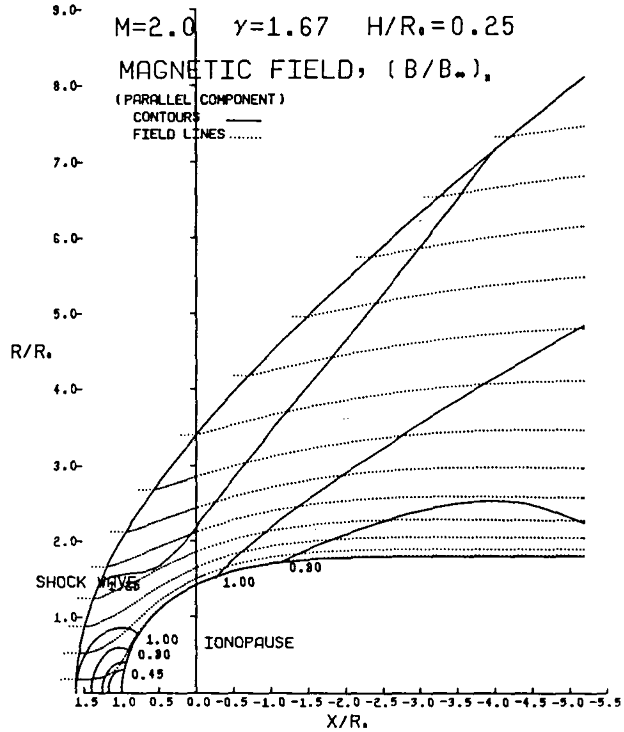
M=2.0 γ=1.67 H/R₀=0.25
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀



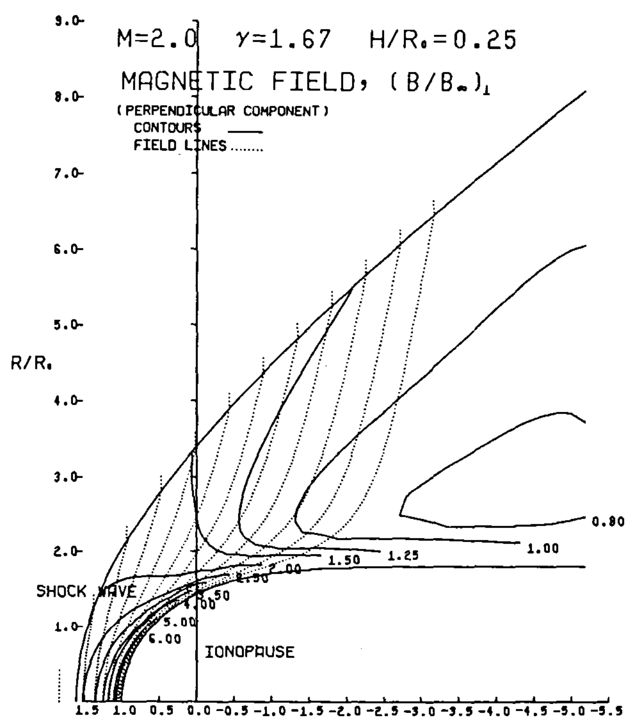
M=2.0 γ=1.67 H/R₀=0.25
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
0.94
0.90
0.70
0.60
0.50
0.40
0.30
0.20
0.10
X/R₀


M=2.0 γ=1.67 H/R₀=0.25
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
2.50
2.00
1.60
1.20
0.80

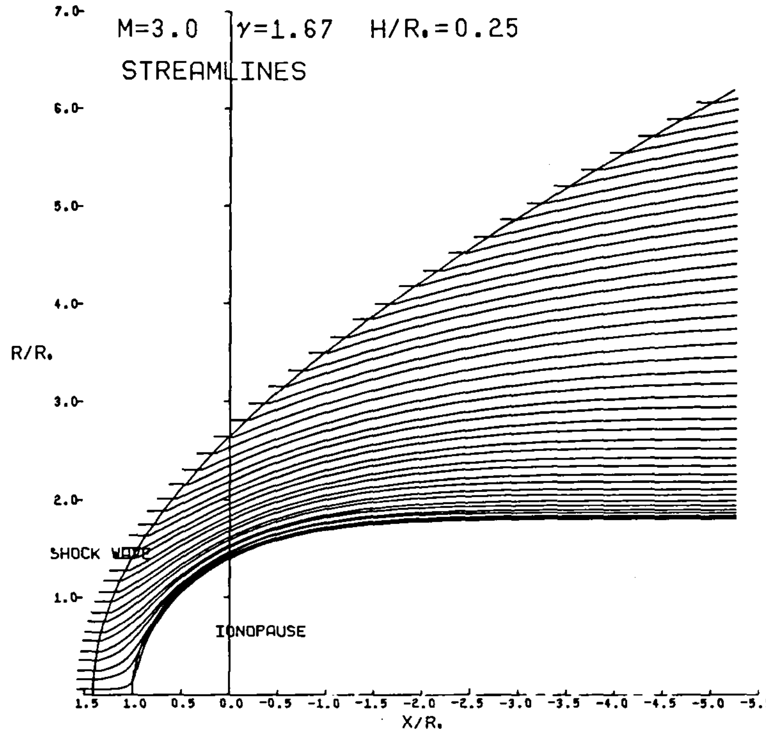

M=2.0 γ=1.67 H/R₀=0.25
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.16
1.49
2.00
2.12
2.21

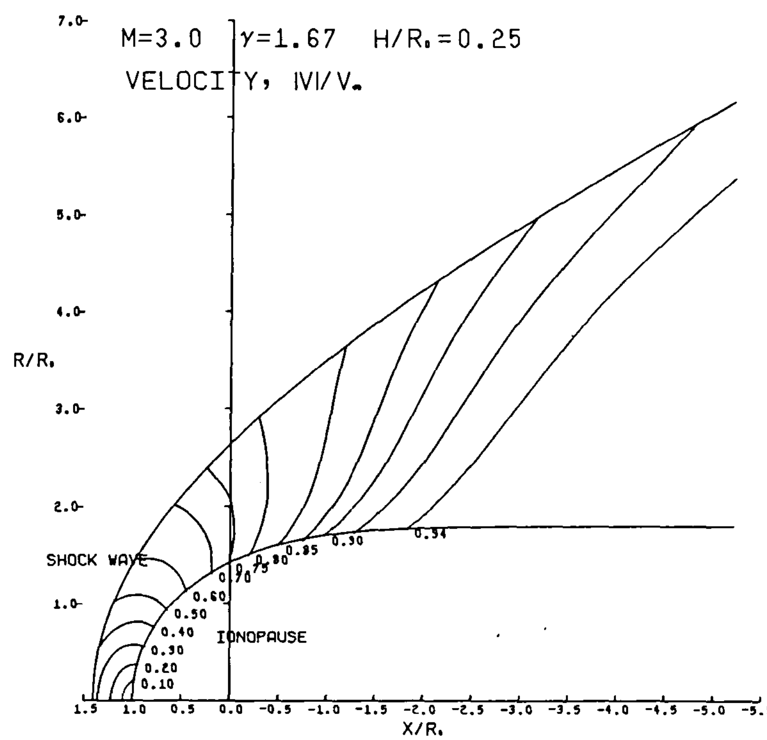


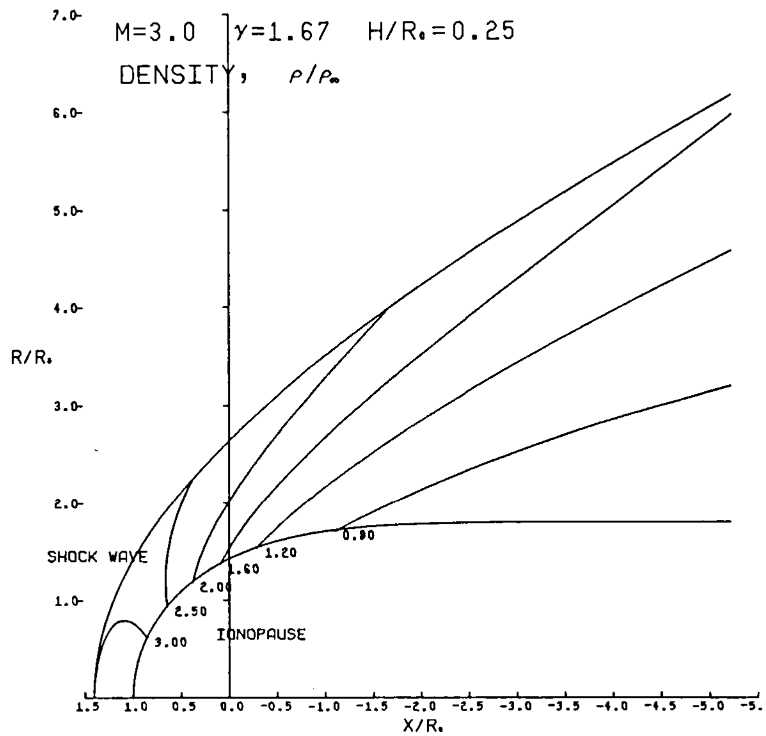
M=2.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.00
0.80
0.45

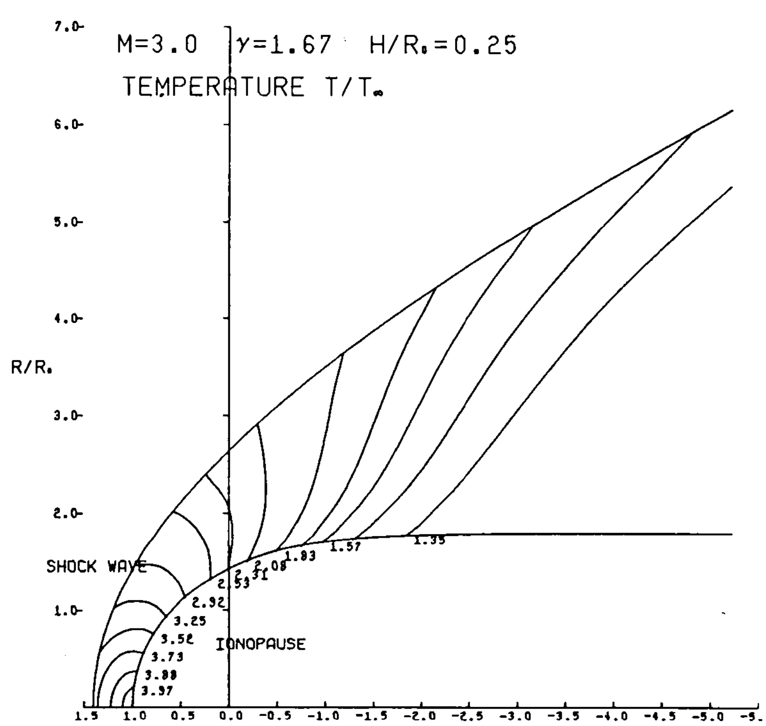


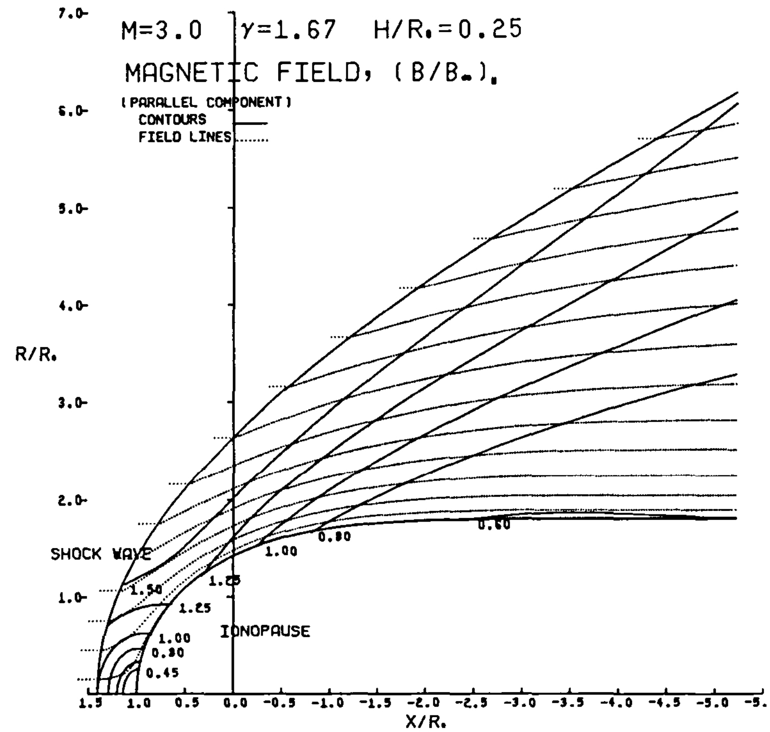
M=2.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.00
1.25
1.50
2.00
5.00
6.00

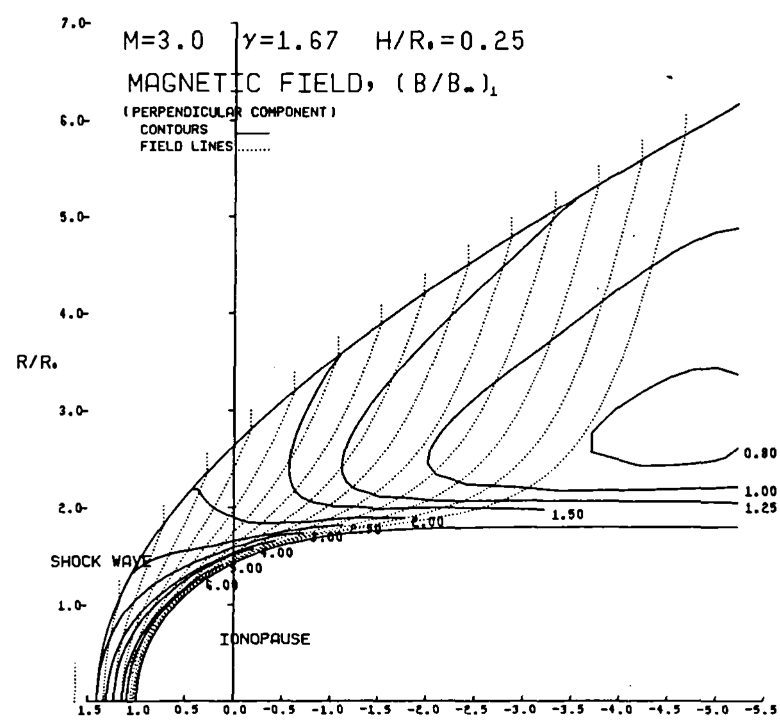

M=3.0 γ=1.67 H/R₀=0.25
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

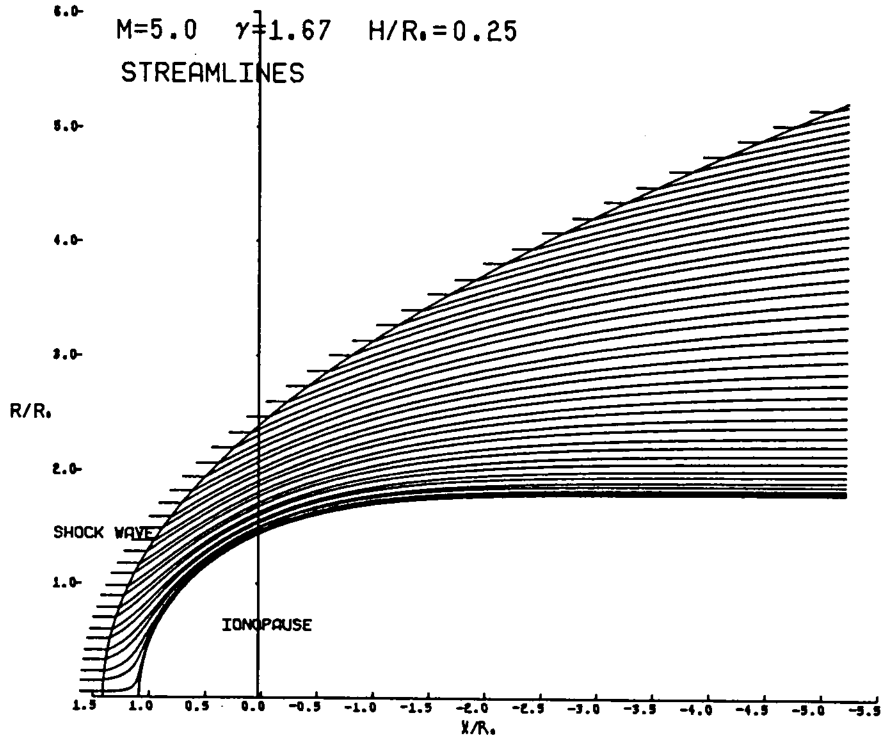

M=3.0 γ=1.67 H/R₀=0.25
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀


M=3.0 γ=1.67 H/R₀=0.25
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
3.00
2.50
2.00
1.60
1.20
0.90

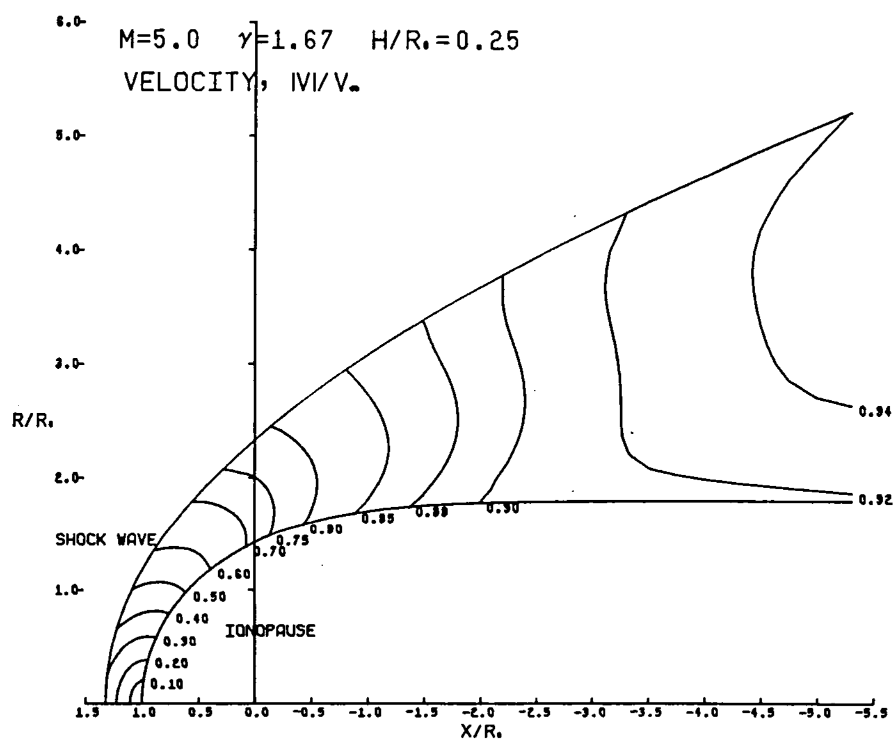

M=3.0 γ=1.67 H/R₀=0.25
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
3.97
3.88
3.73
3.52
3.25
2.92
2.53
2.31
2.08
1.83
1.57
1.35


M=3.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.50
1.25
1.00
0.80
0.45
0.60

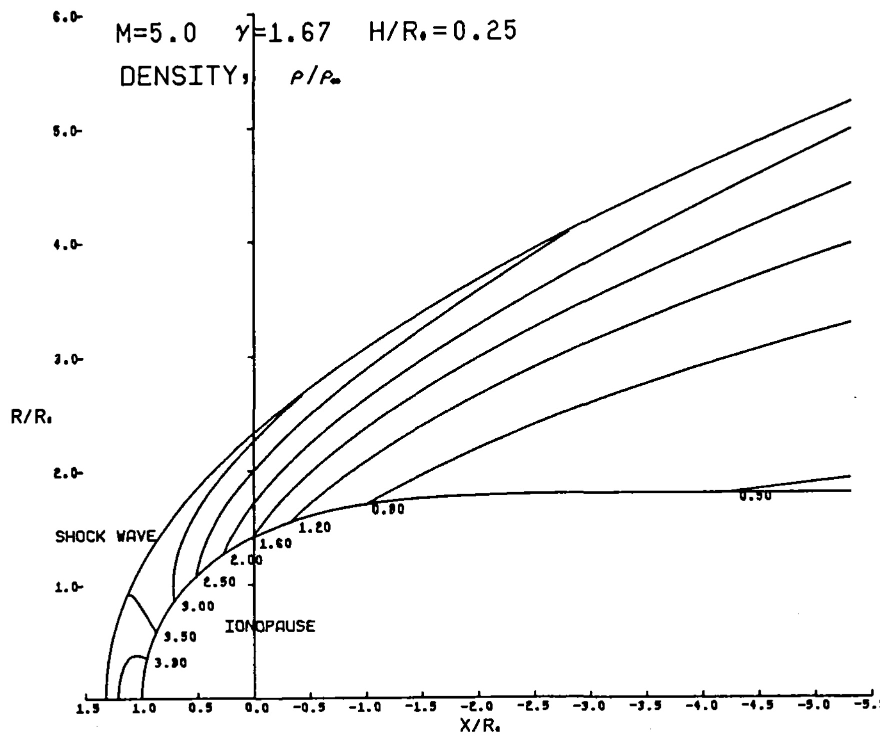

M=3.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.00
1.25
1.50
2.00

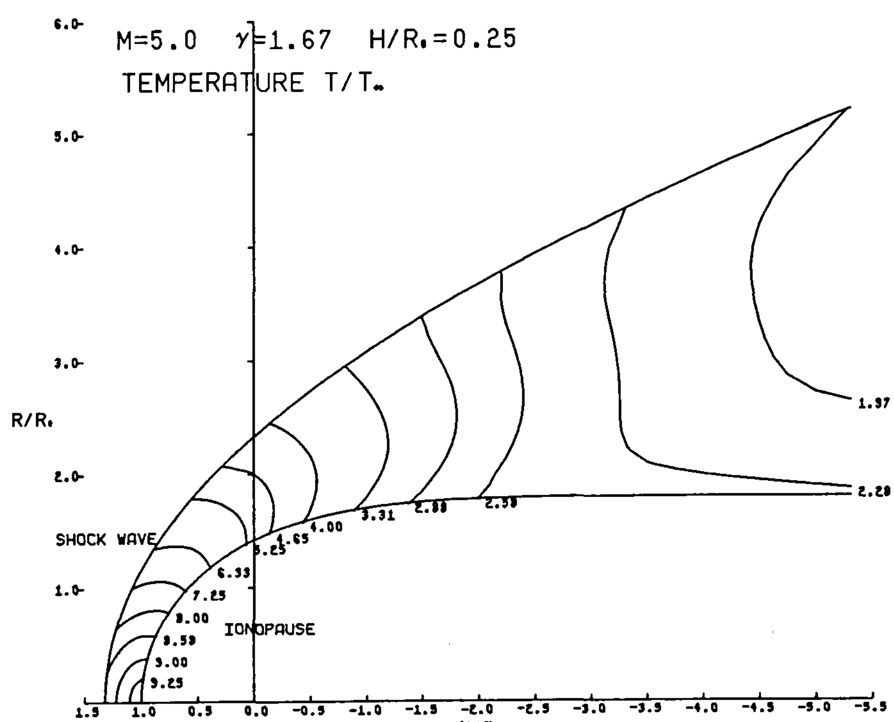


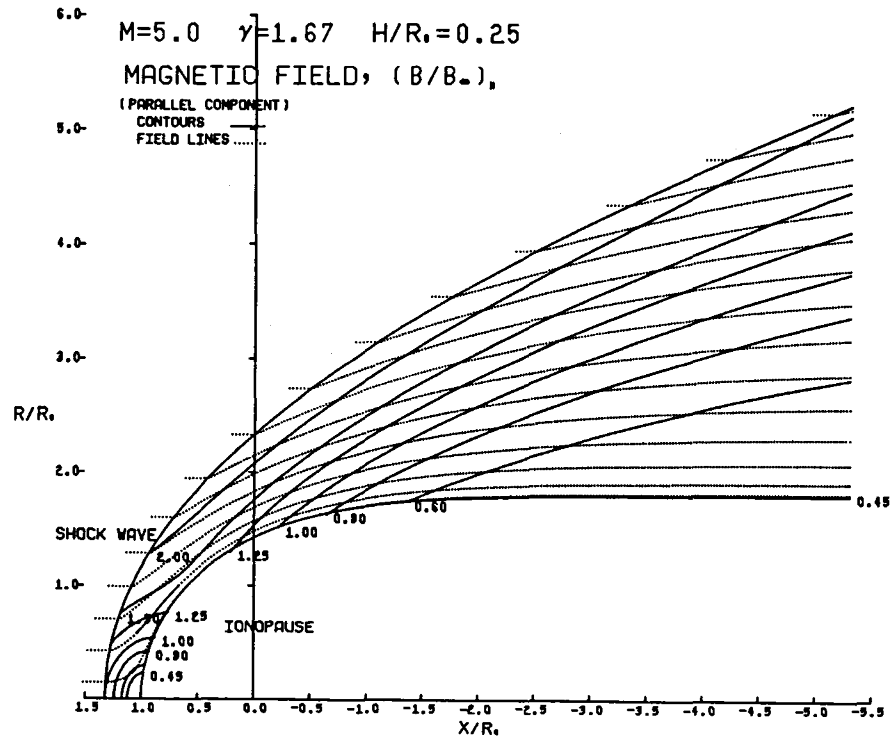
M=5.0 γ=1.67 H/R₀=0.25
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

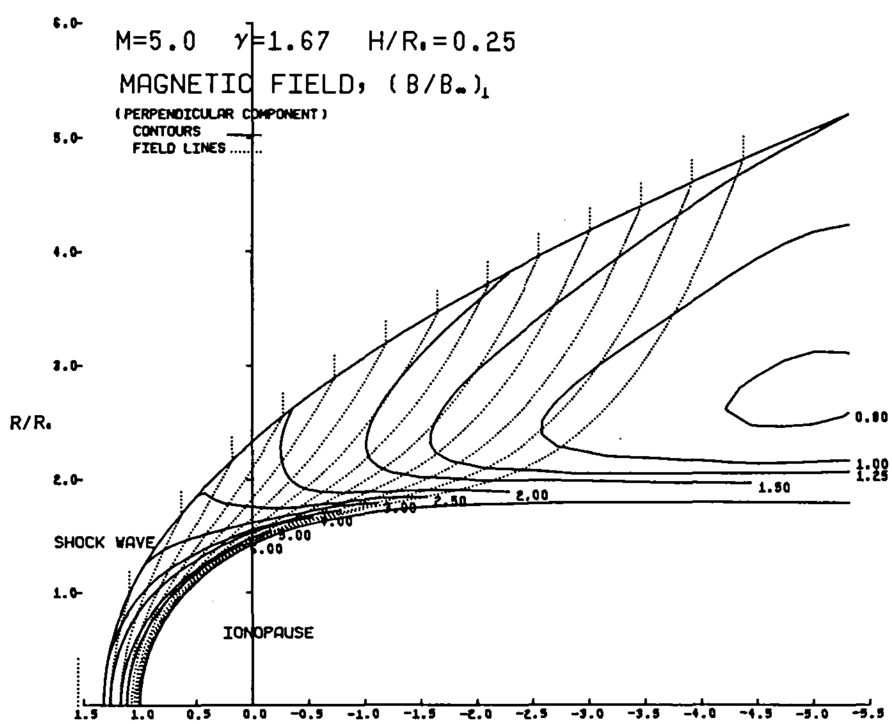


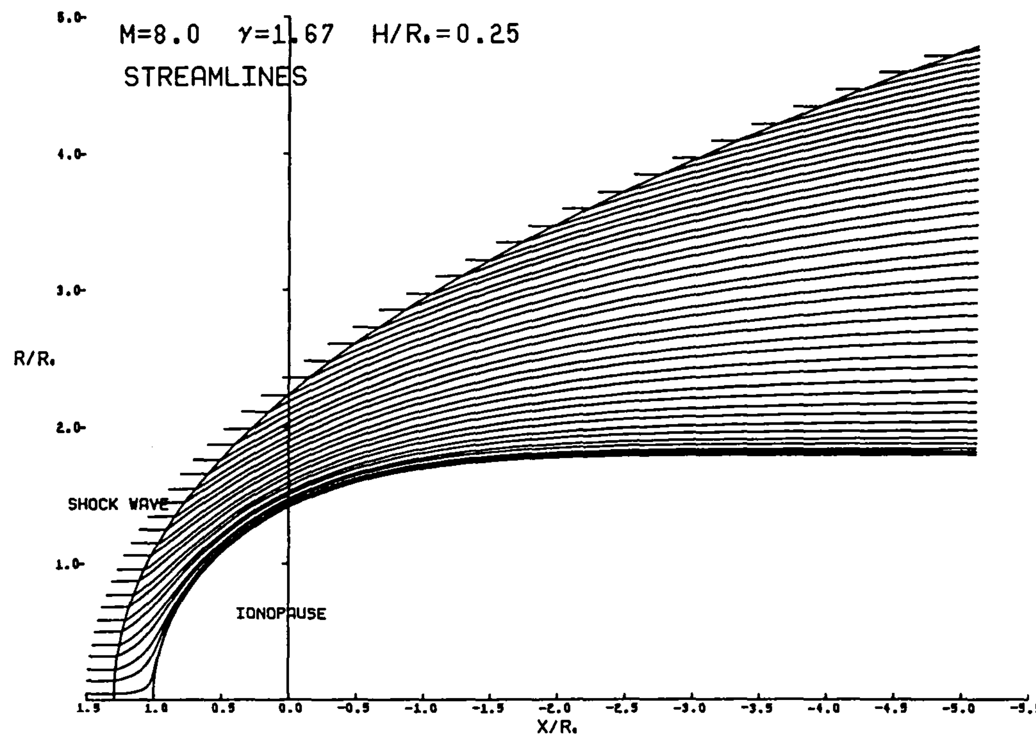
M=5.0 γ=1.67 H/R₀=0.25
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
0.94
0.92
0.90
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10
R/R₀
X/R₀

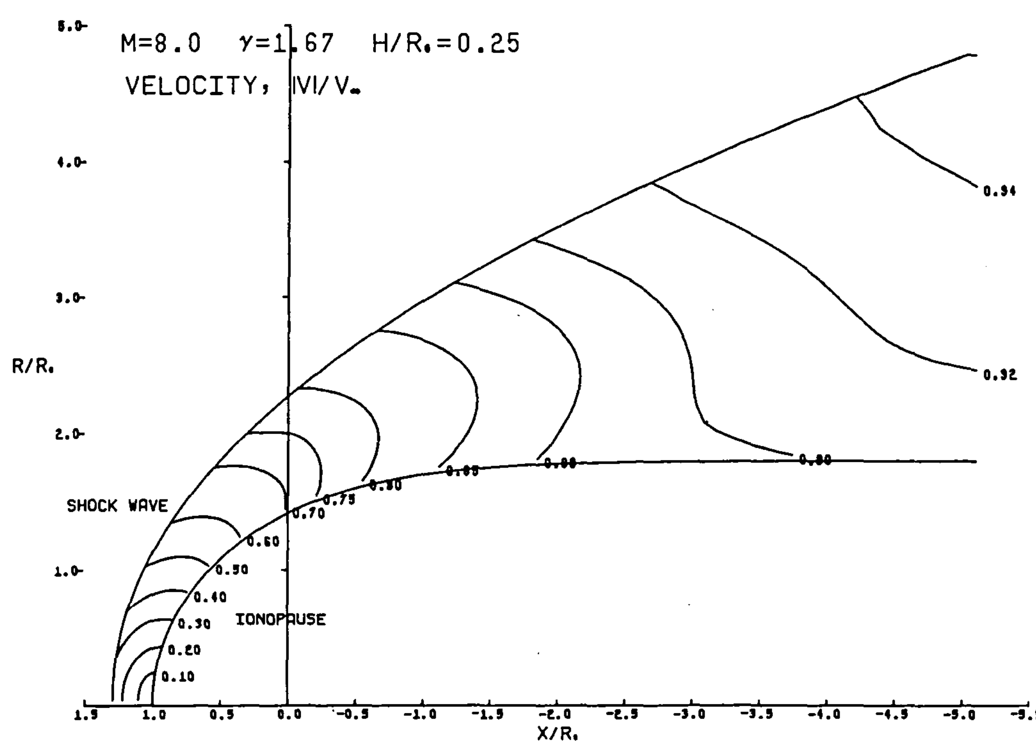

M=5.0 γ=1.67 H/R₀=0.25
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.90
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.90

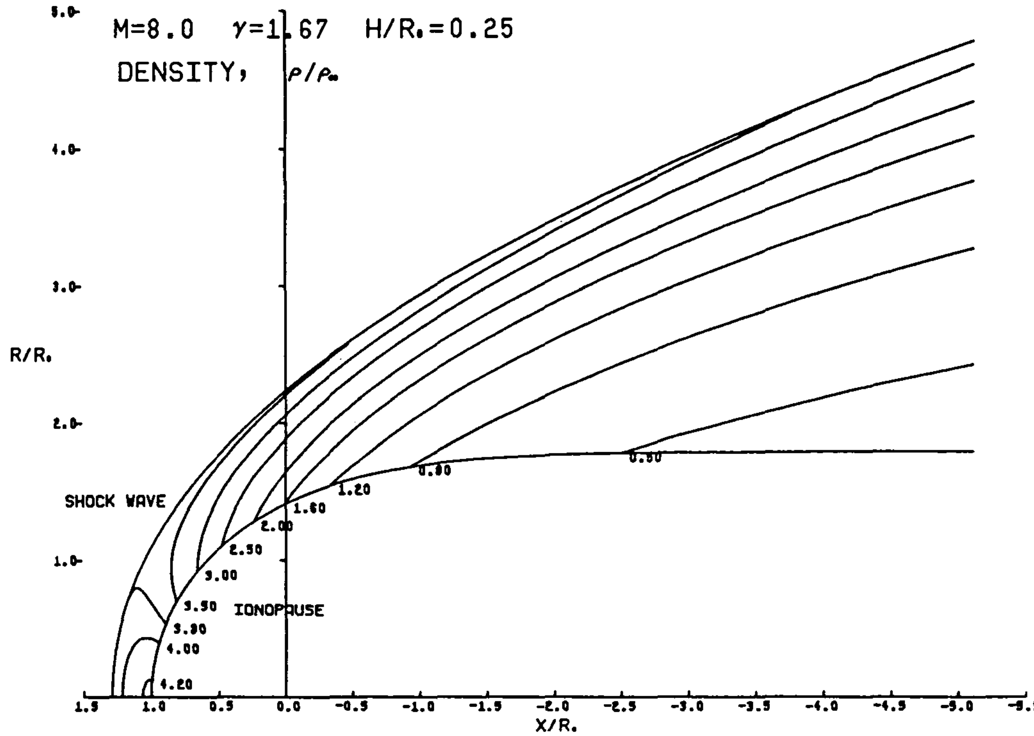

M=5.0 γ=1.67 H/R₀=0.25
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
1.97
2.28
2.59
2.88
3.31
4.00
4.65
5.25
6.33
7.25
8.00
8.59
9.00
9.25

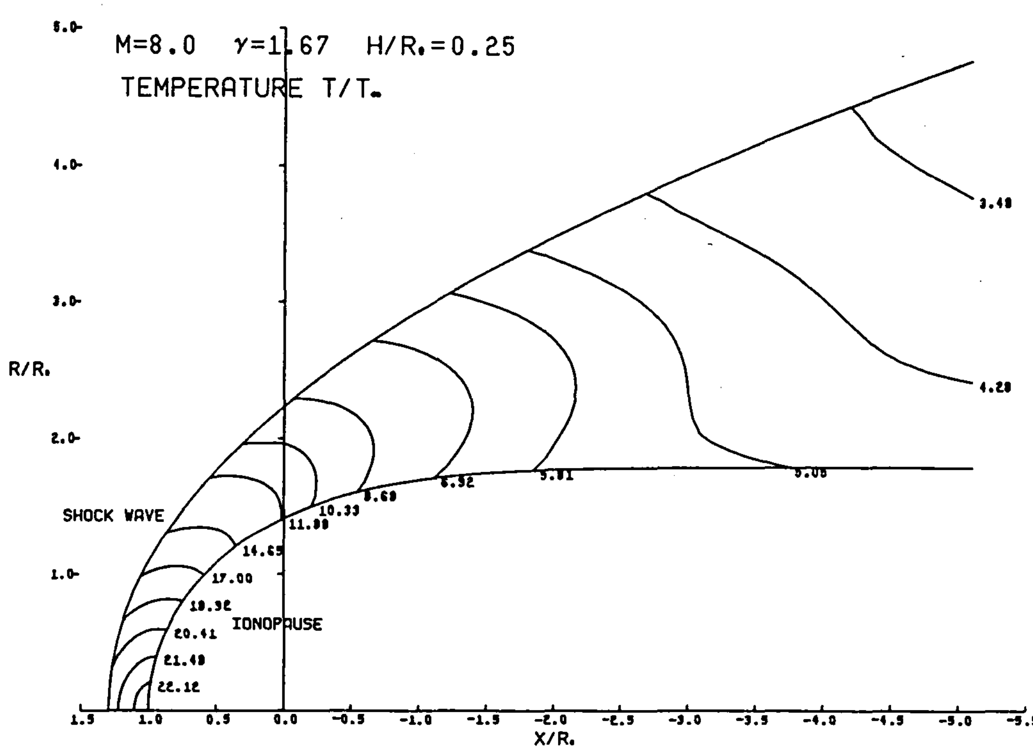

M=5.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.45
0.60
0.80
1.00
1.25
2.00
1.50

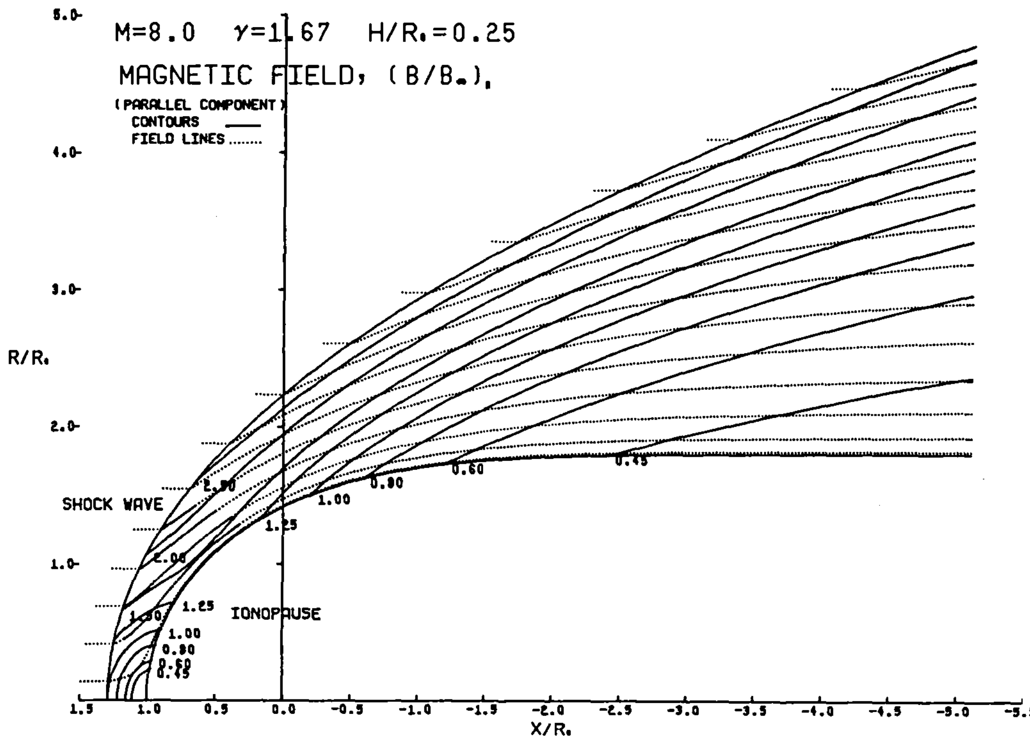

M=5.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.00
1.25
1.50
2.00
2.50
3.00
4.00
5.00

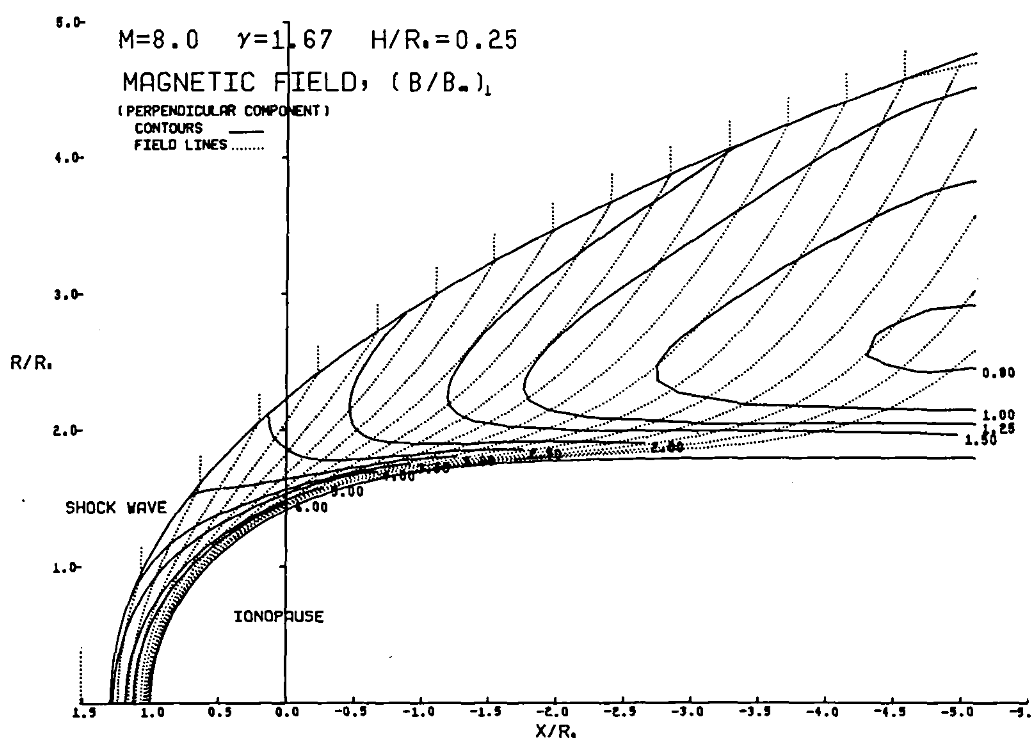

M=8.0 γ=1.67 H/R₀=0.25
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

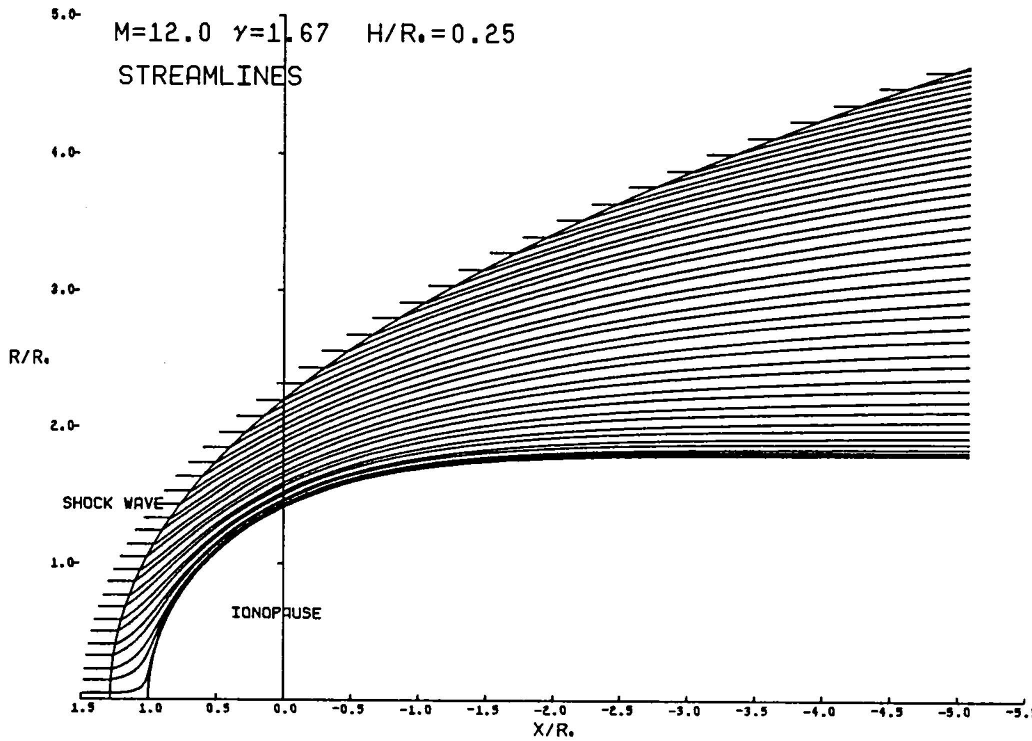

M=8.0 γ=1.67 H/R₀=0.25
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀
0.94
0.92
0.90
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10

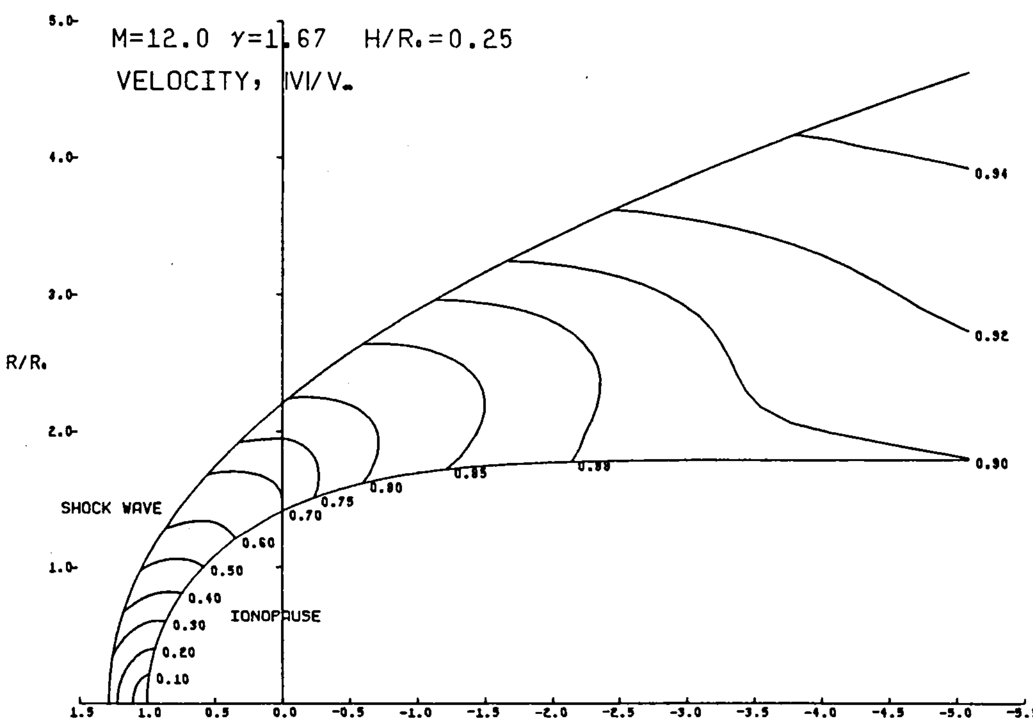

M=8.0 γ=1.67 H/R₀=0.25
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.50
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.90
4.00
4.20

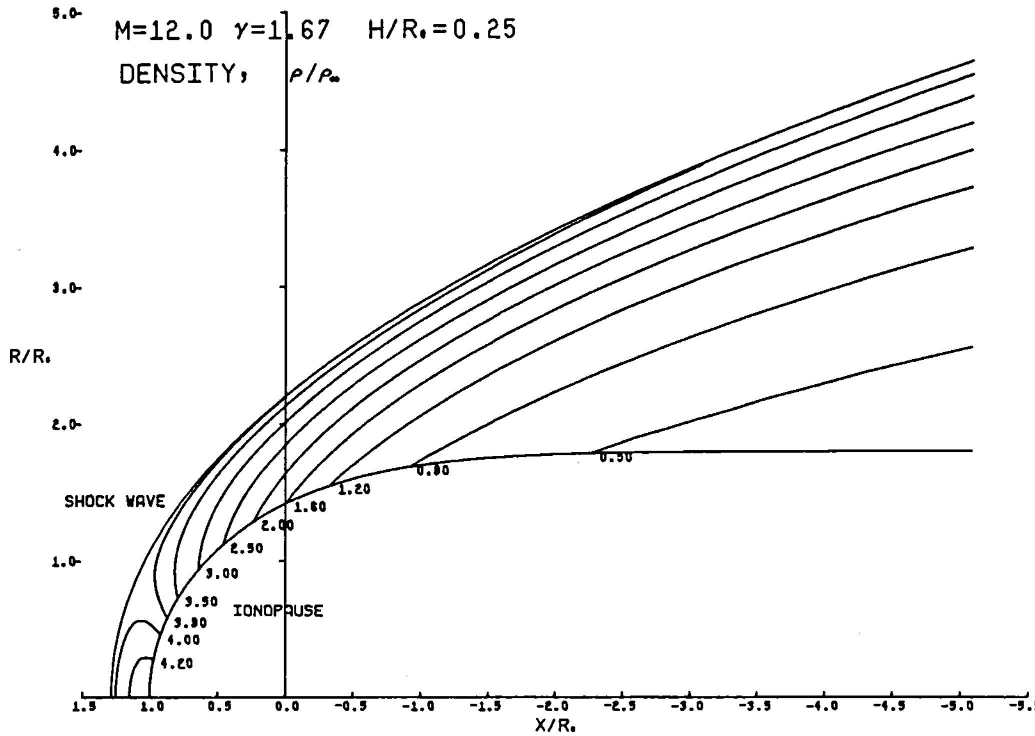

M=8.0 γ=1.67 H/R₀=0.25
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
3.48
4.28
5.08
5.81
6.32
8.69
10.33
11.88
14.65
17.00
19.32
20.41
21.49
22.12

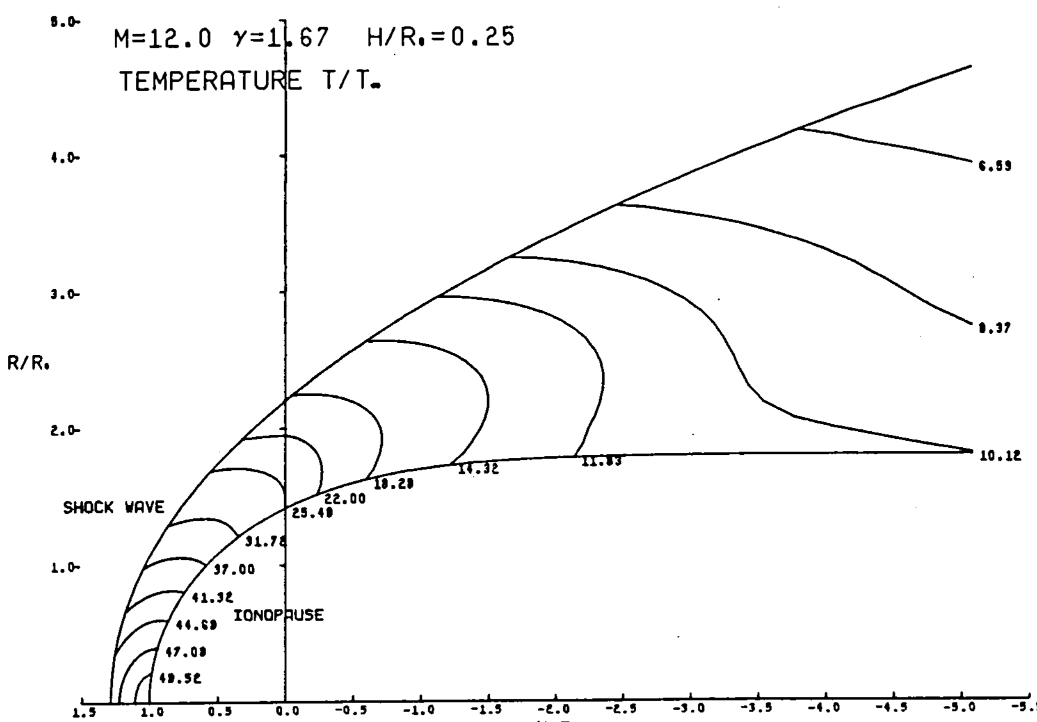

M=8.0 γ=1.67 H/Ro=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/Ro
X/Ro

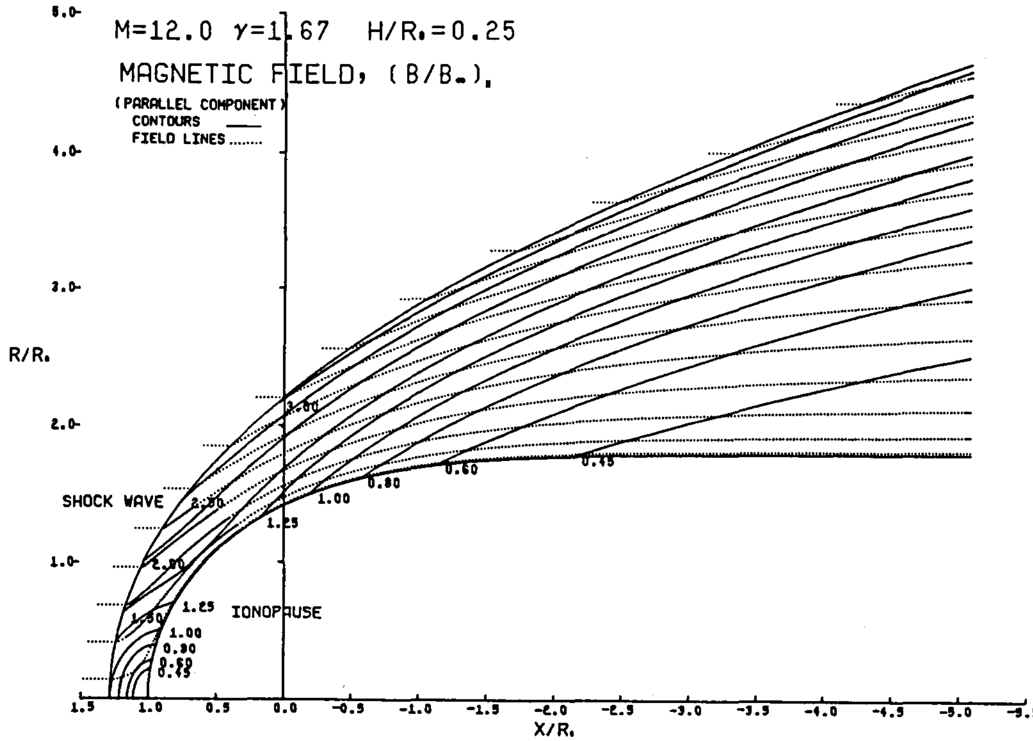

M=8.0 γ=1.67 H/Ro=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/Ro
X/Ro


M=12.0 γ=1.67 H/R₀=0.25
STREAMLINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE

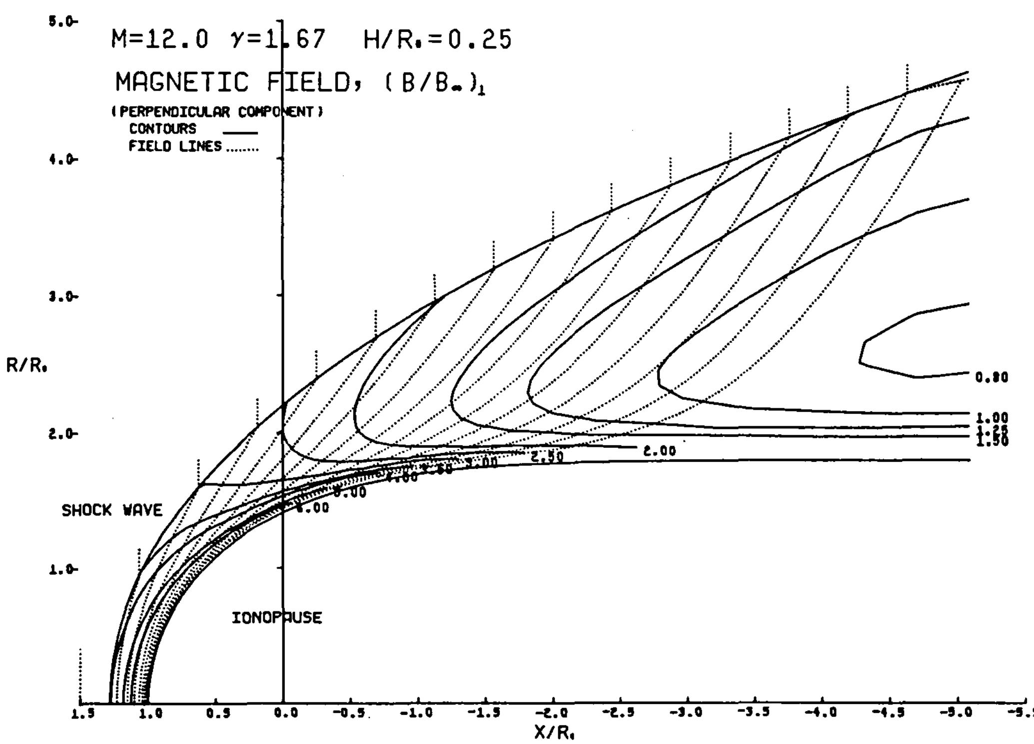

M=12.0 γ=1.67 H/R₀=0.25
VELOCITY, |V|/V∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.94
0.92
0.90
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10


M=12.0 γ=1.67 H/R₀=0.25
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.50
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80
4.00
4.20

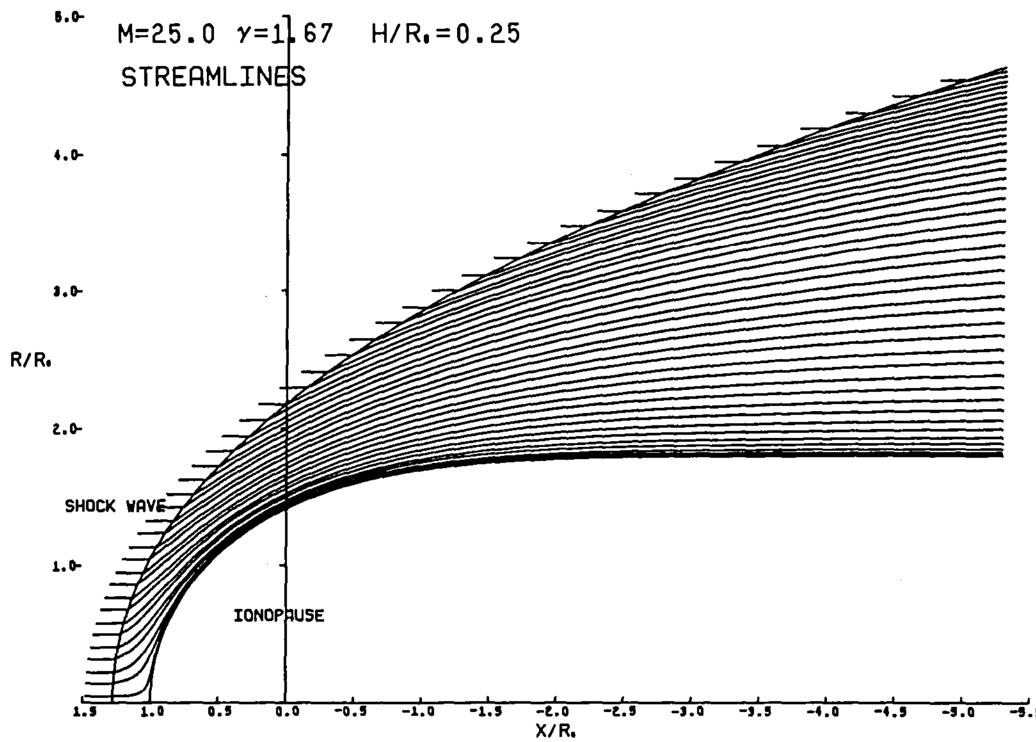

M=12.0 γ=1.67 H/R₀=0.25
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
6.53
8.37
10.12
11.83
14.32
18.29
22.00
25.49
31.78
37.00
41.32
44.69
47.08
49.52

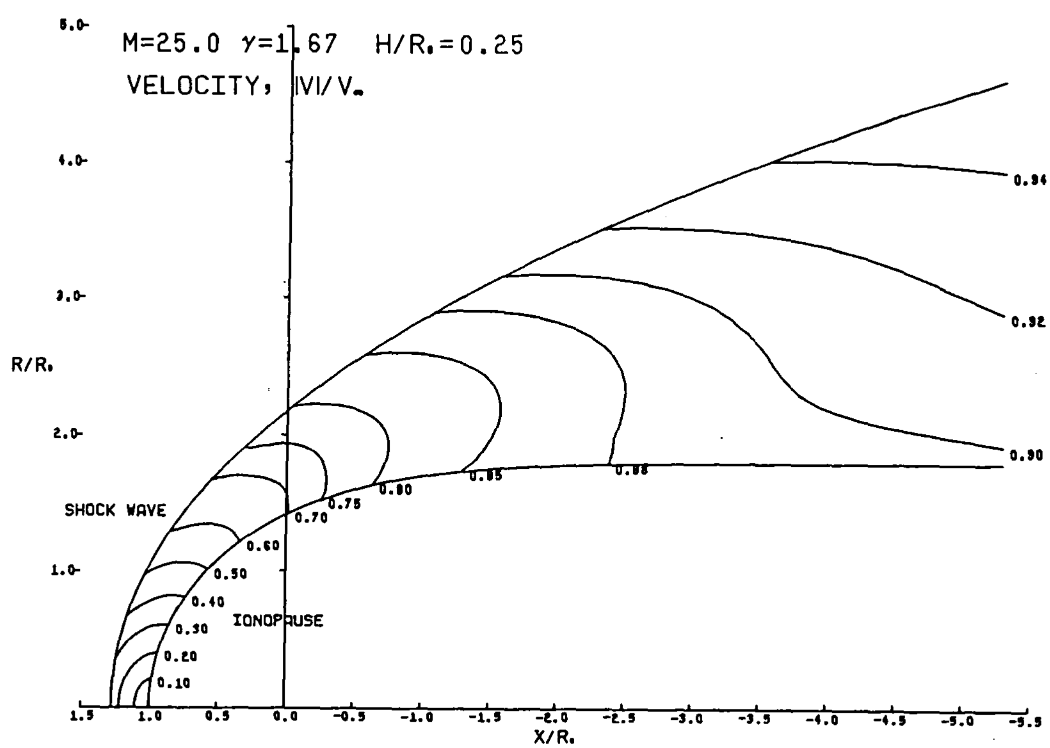


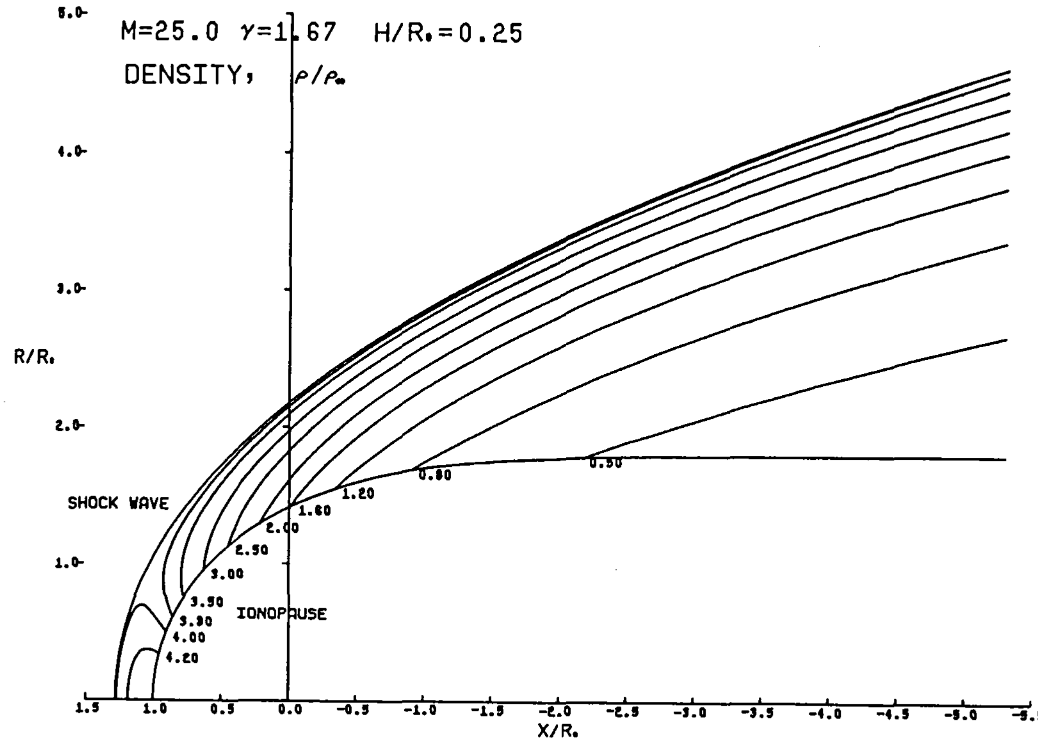
M=12.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

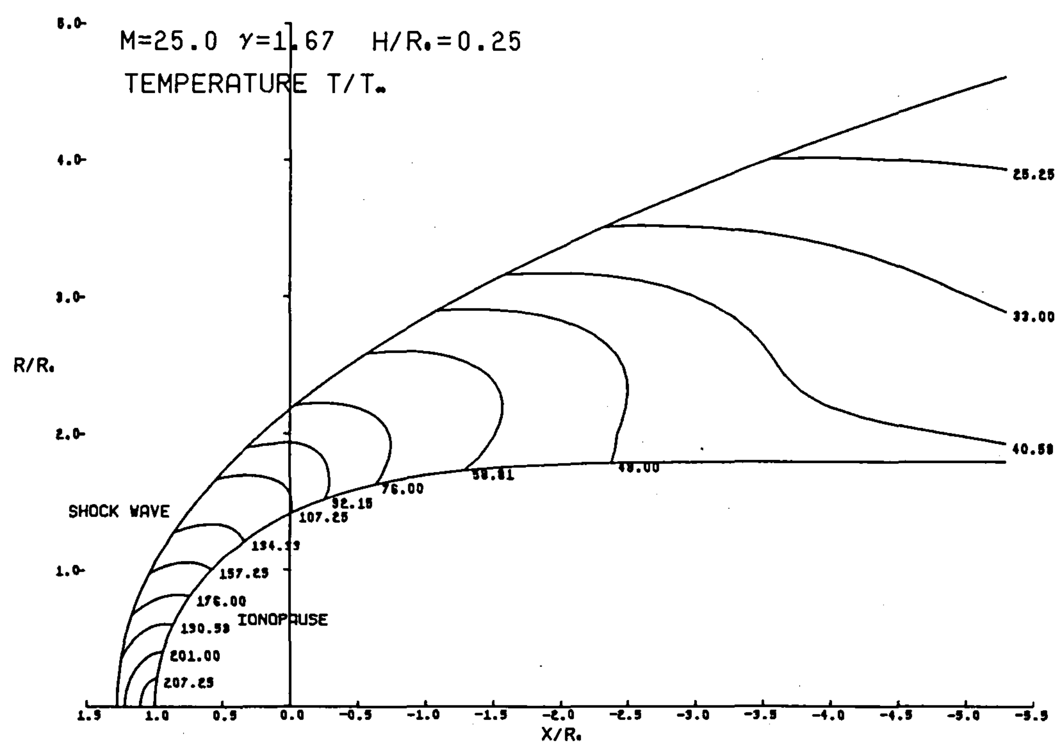


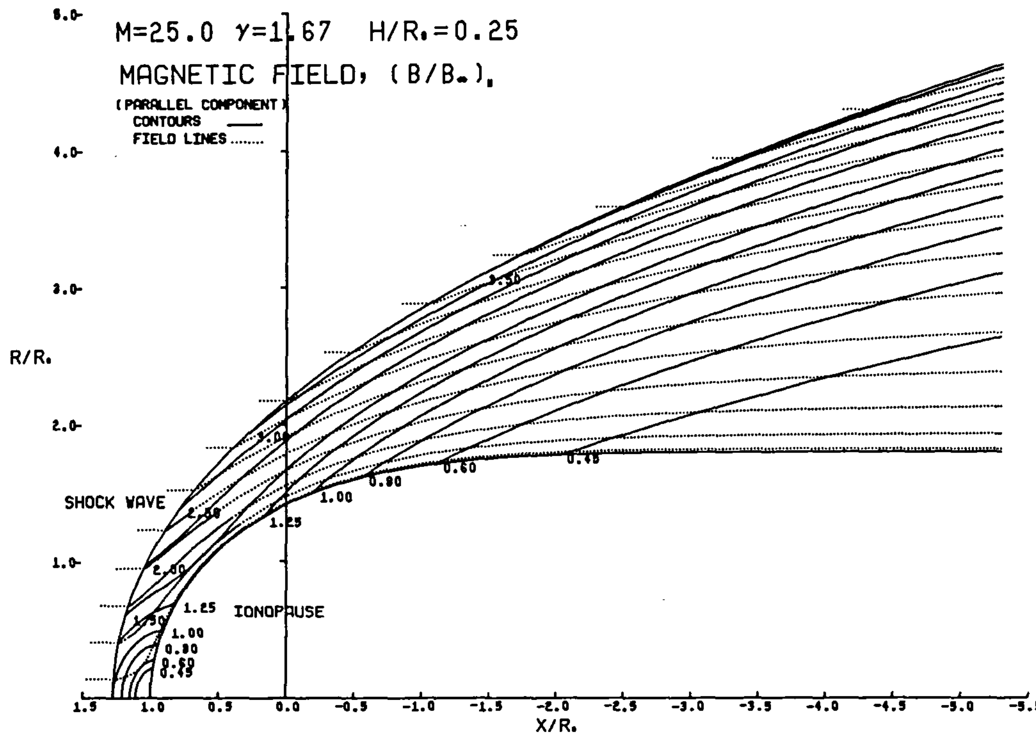
M=12.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

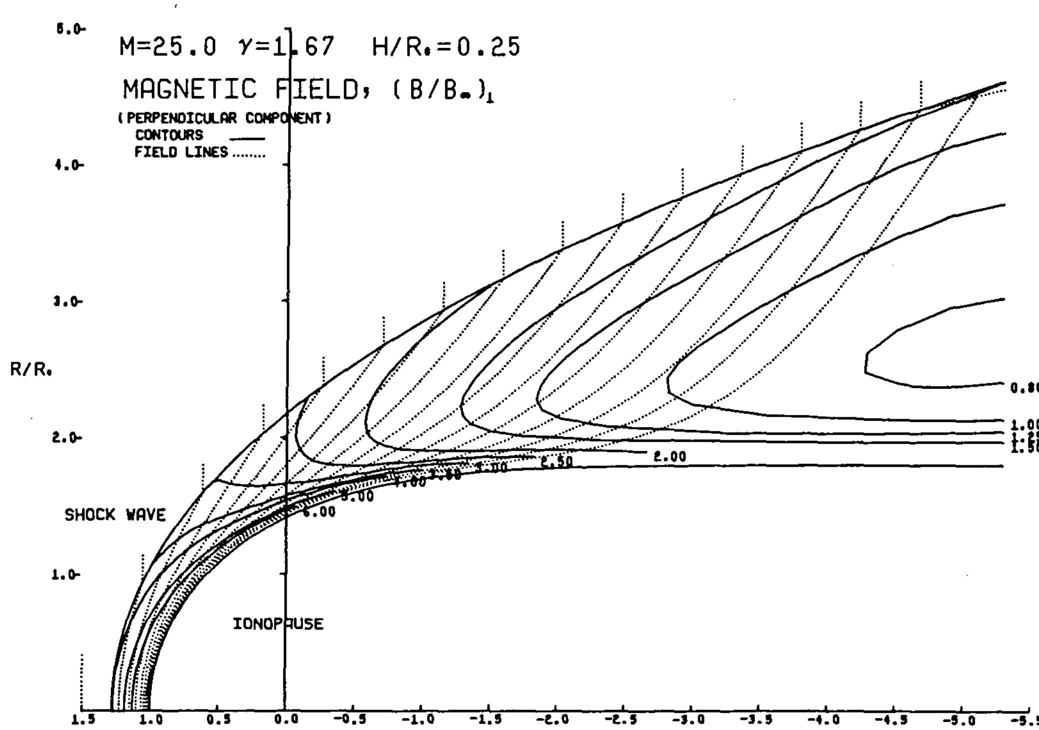

M=25.0 γ=1.67 H/R₀=0.25
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀
5.0
4.0
3.0
2.0
1.0
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
-4.5
-5.0
-5.5

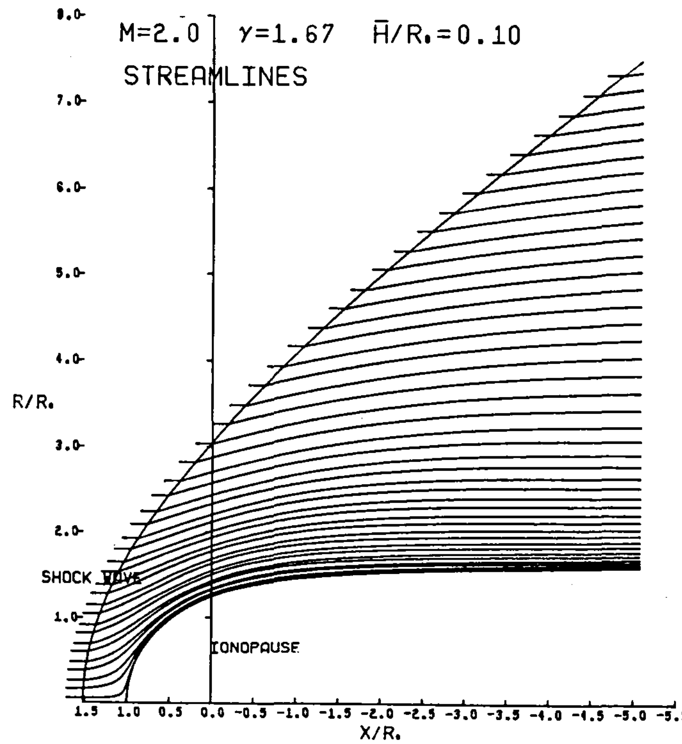

M=25.0 γ=1.67 H/R₀=0.25
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀
0.94
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10


M=25.0 γ=1.67 H/R₀=0.25
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.50
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.90
4.00
4.20

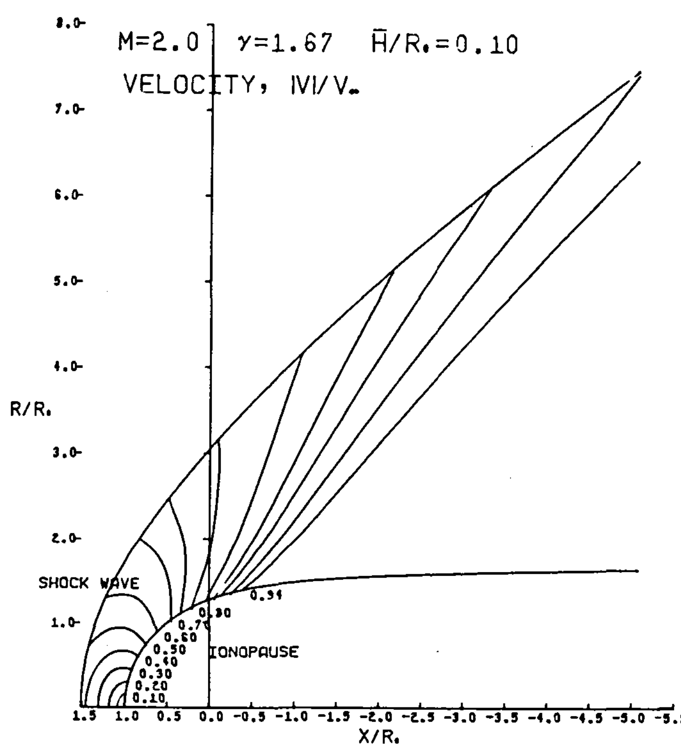

M=25.0 γ=1.67 H/R₀=0.25
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
25.25
33.00
40.50
76.00
92.15
107.25
157.25
176.00
201.00
207.25


M=25.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

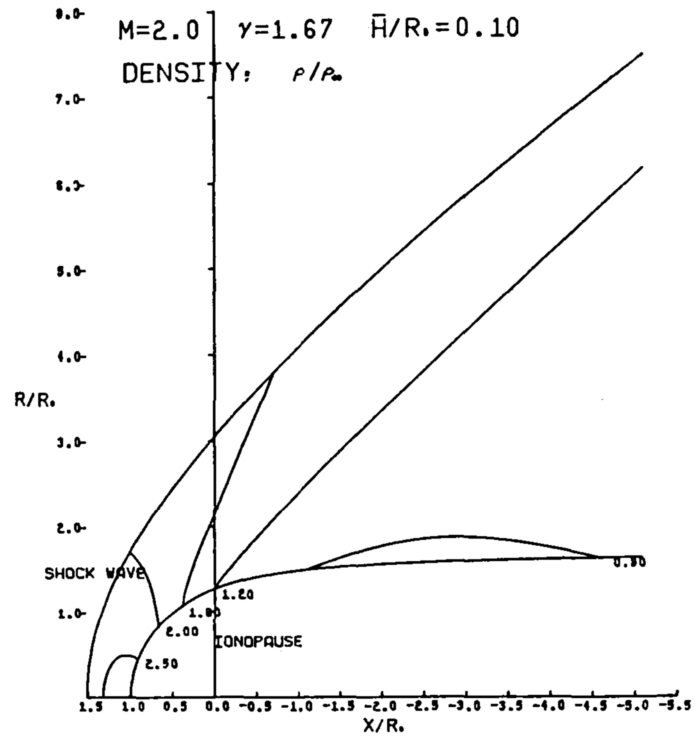

M=25.0 γ=1.67 H/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

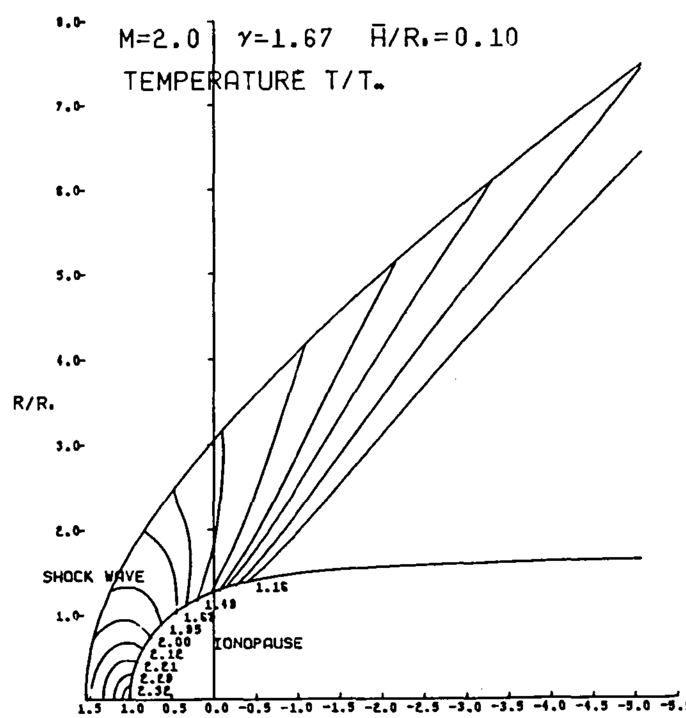


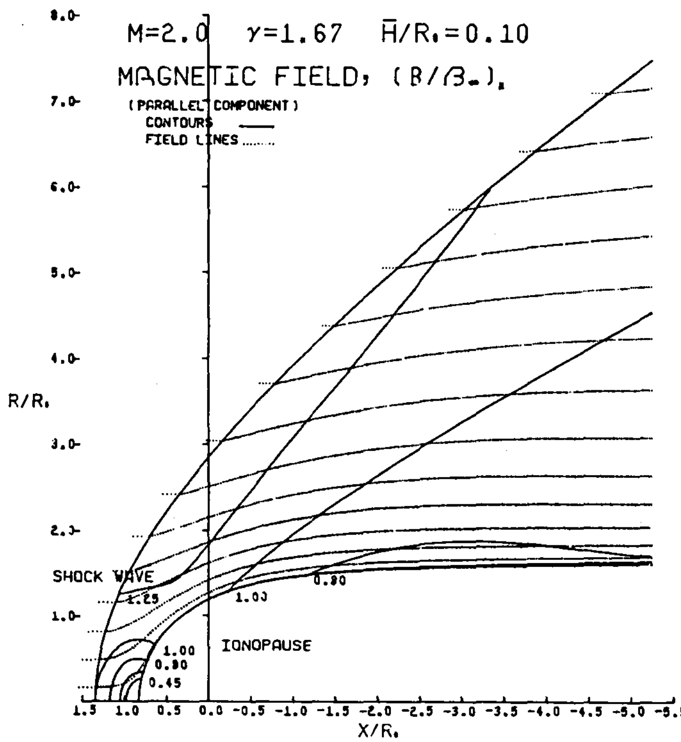
M=2.0 γ=1.67 H̄/R₀=0.10
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

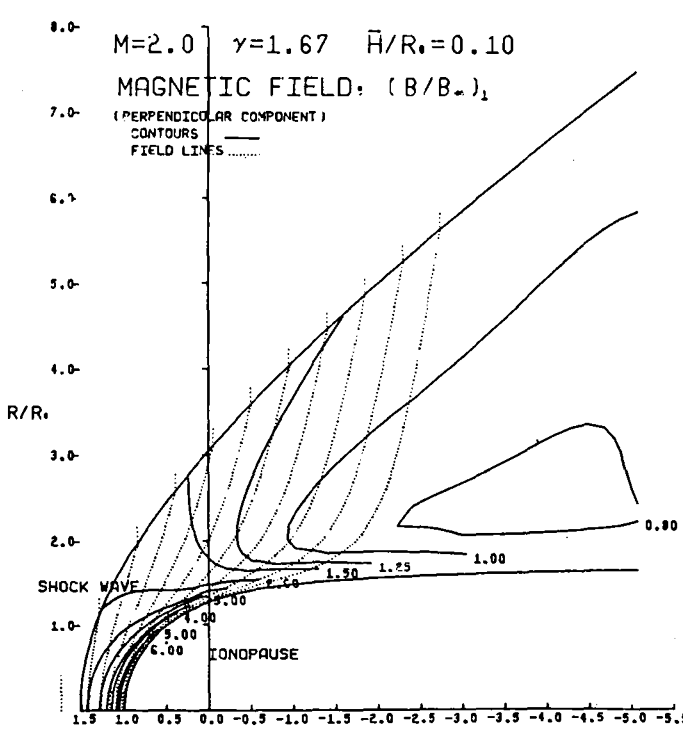


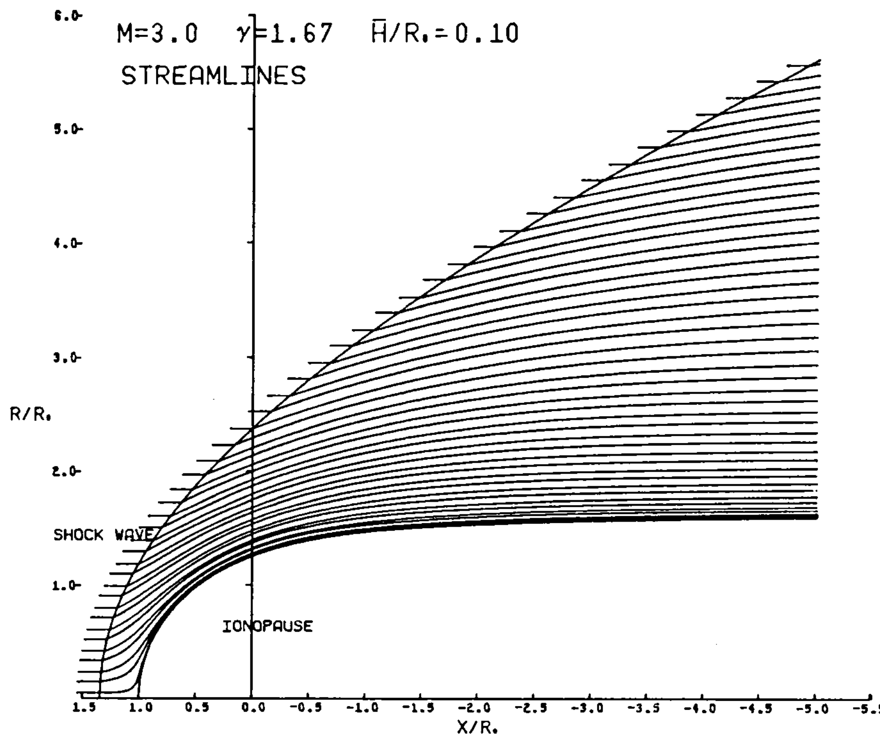
M=2.0 γ=1.67 H̄/R₀=0.10
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

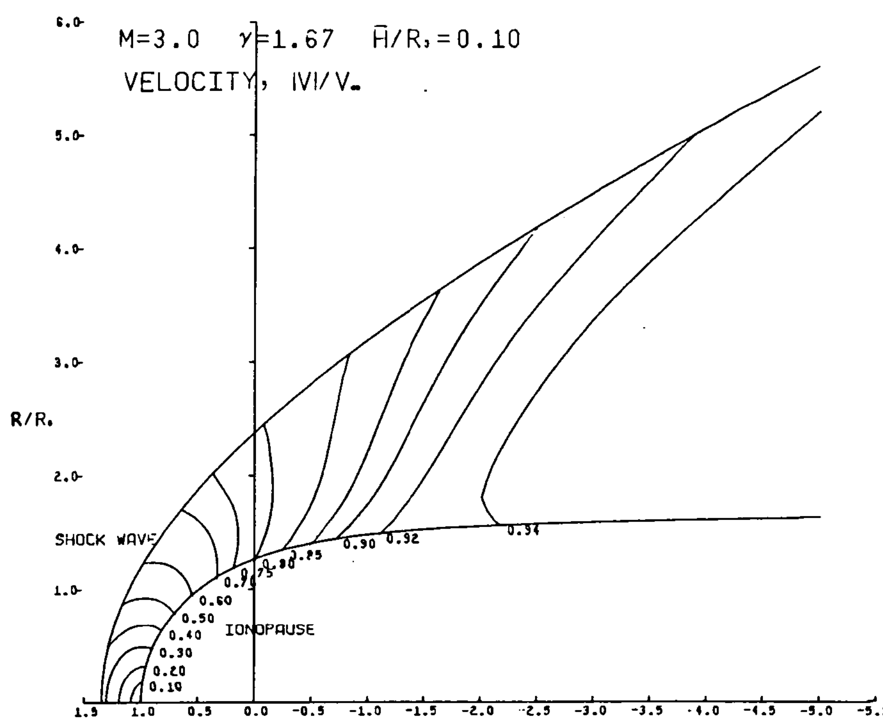

M=2.0 γ=1.67 H̄/R₀=0.10
DENSITY: ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.20
1.80
2.00
2.50
0.80

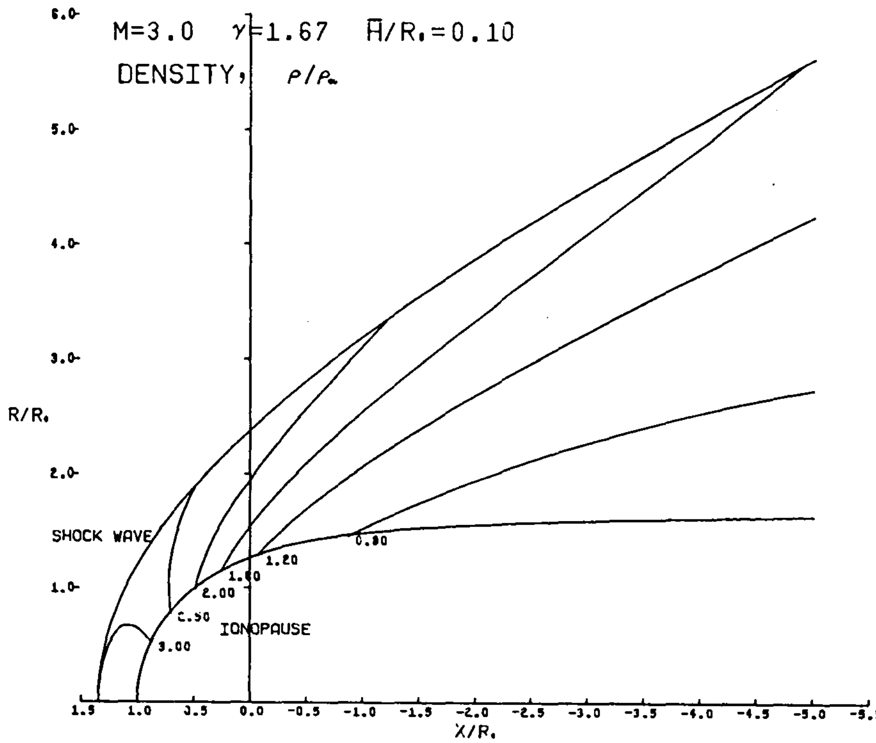

M=2.0 γ=1.67 H̄/R₀=0.10
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.16
1.49
1.67
1.85
2.00
2.12
2.21
2.28
2.32


M=2.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
1.25
1.00
0.80
1.00
0.80
0.45
IONOPAUSE
X/R₀

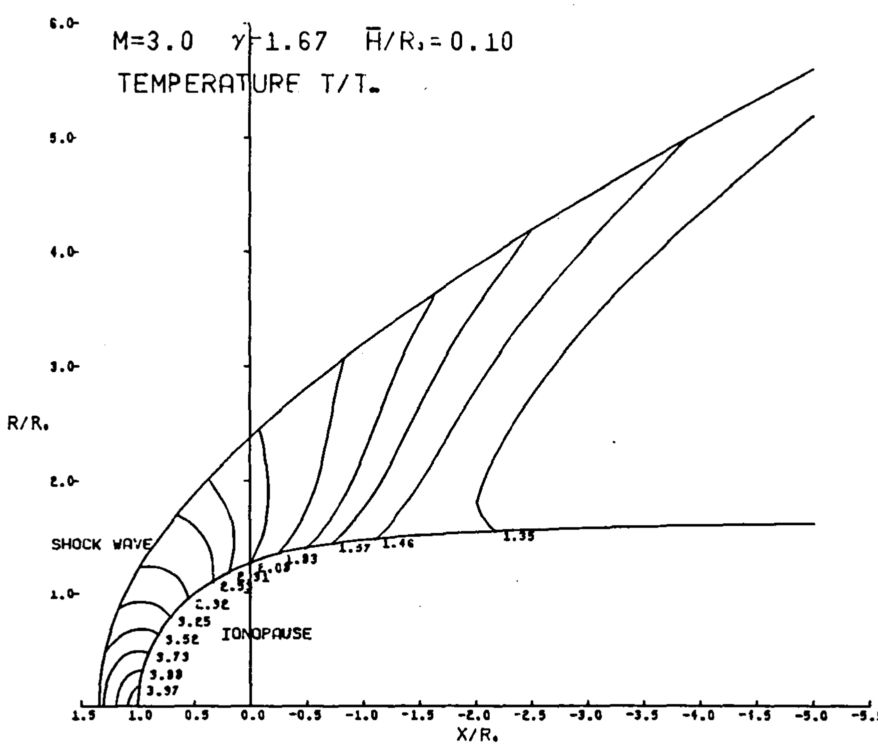

M=2.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD: (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
0.80
1.00
1.25
1.50
2.00
3.00
4.00
5.00
6.00
IONOPAUSE
X/R₀


M=3.0 γ=1.67 H̄/R₁=0.10
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₁
X/R₁
6.0
5.0
4.0
3.0
2.0
1.0
1.5
1.0
0.5
0.0
-0.5
-1.0
-1.5
-2.0
-2.5
-3.0
-3.5
-4.0
-4.5
-5.0
-5.5

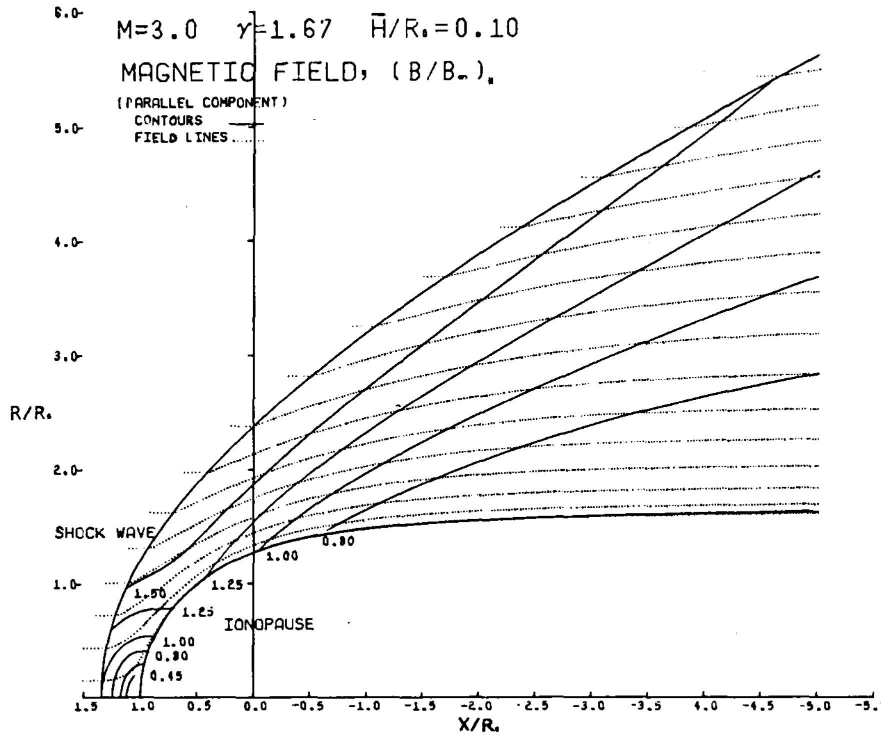

M=3.0 γ=1.67 H̄/R₁=0.10
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
R/R₁
X/R₁
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.75
0.80
0.85
0.90
0.92
0.94

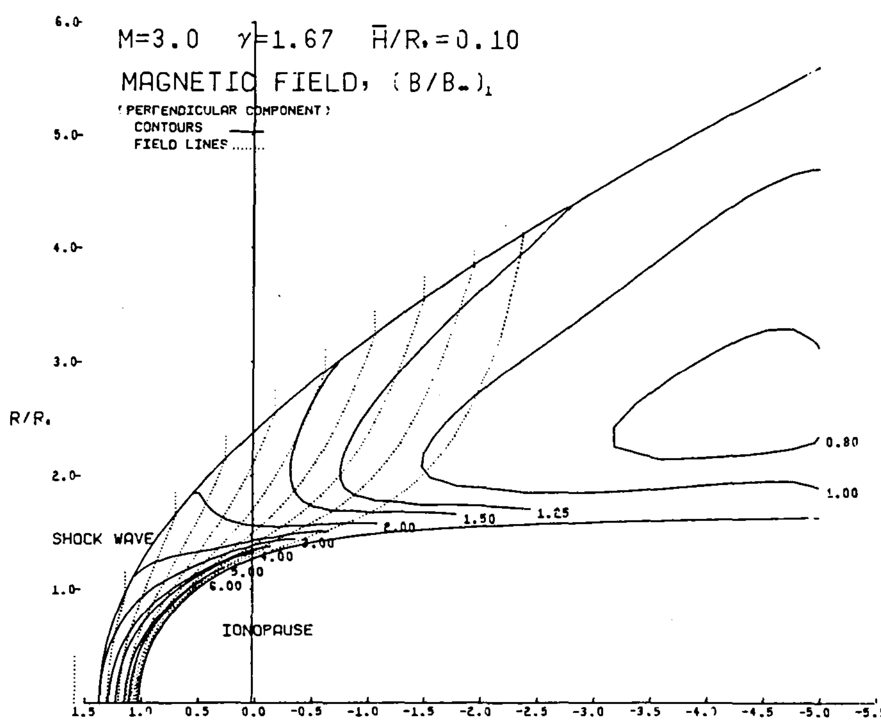


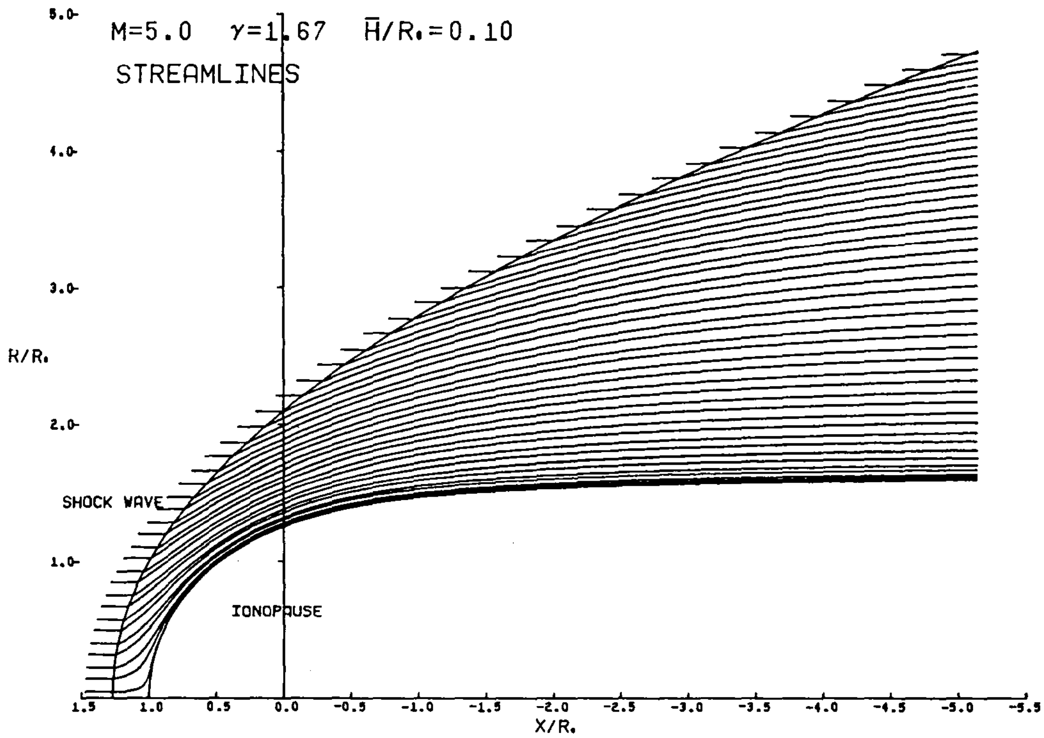
M=3.0 γ=1.67 H̄/R₀=0.10
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
3.00
2.50
2.00
1.60
1.20
0.80

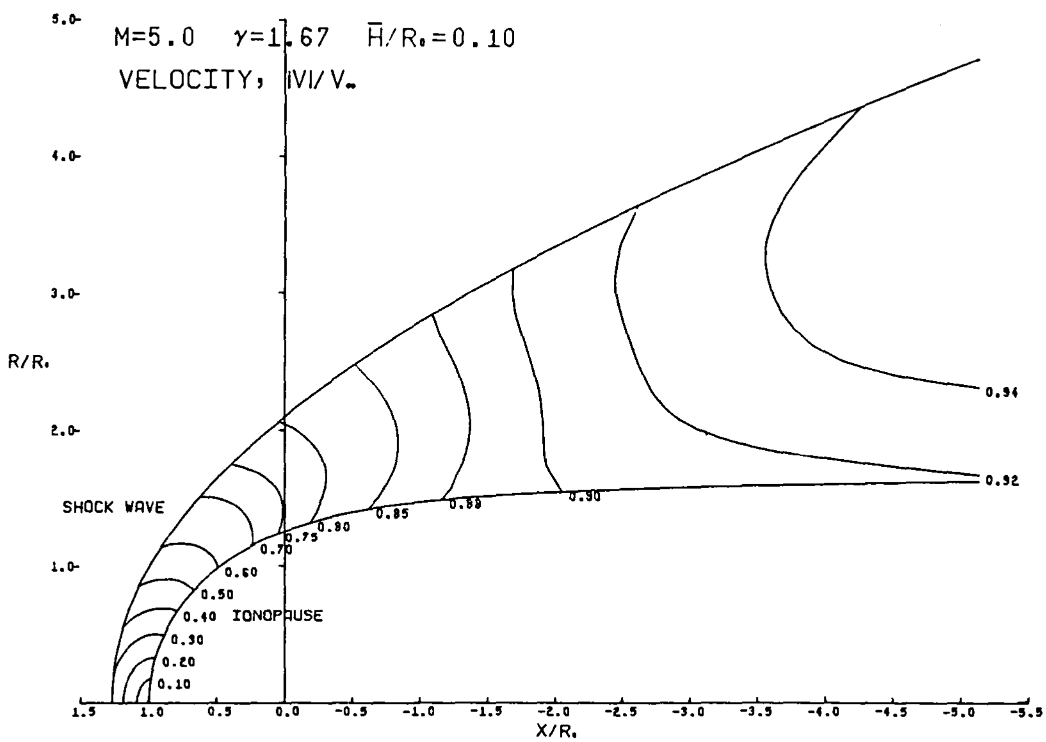


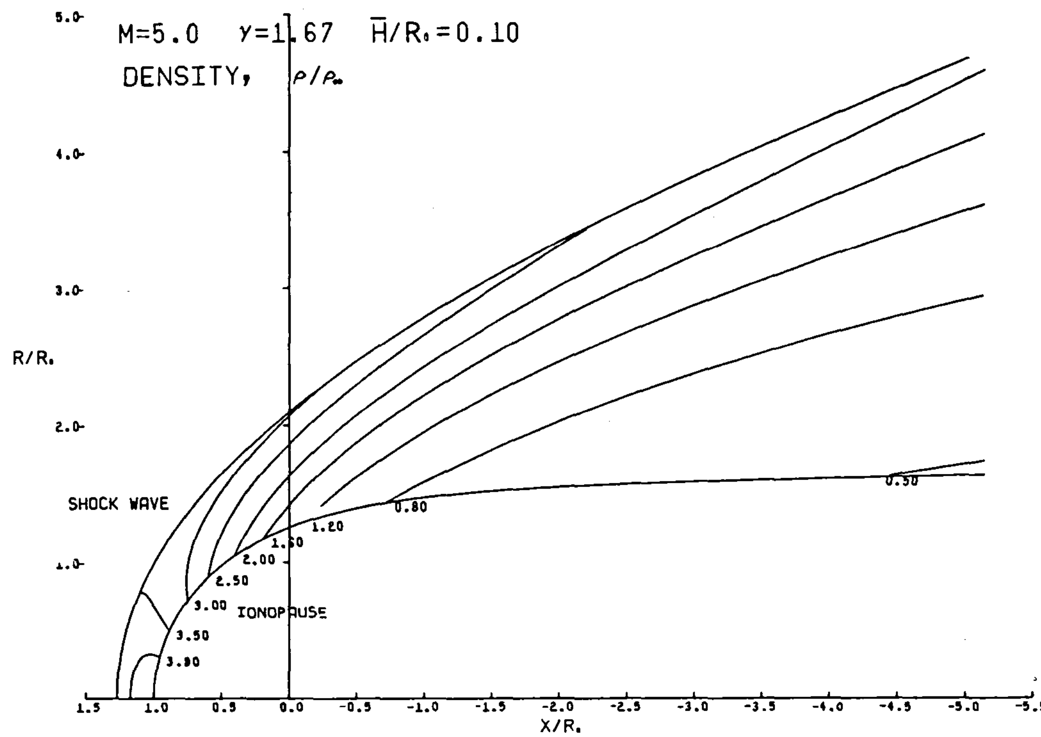
M=3.0 γ=1.67 H̄/R₀=0.10
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
3.97
3.89
3.73
3.52
3.25
2.32
1.83
1.57
1.46
1.35

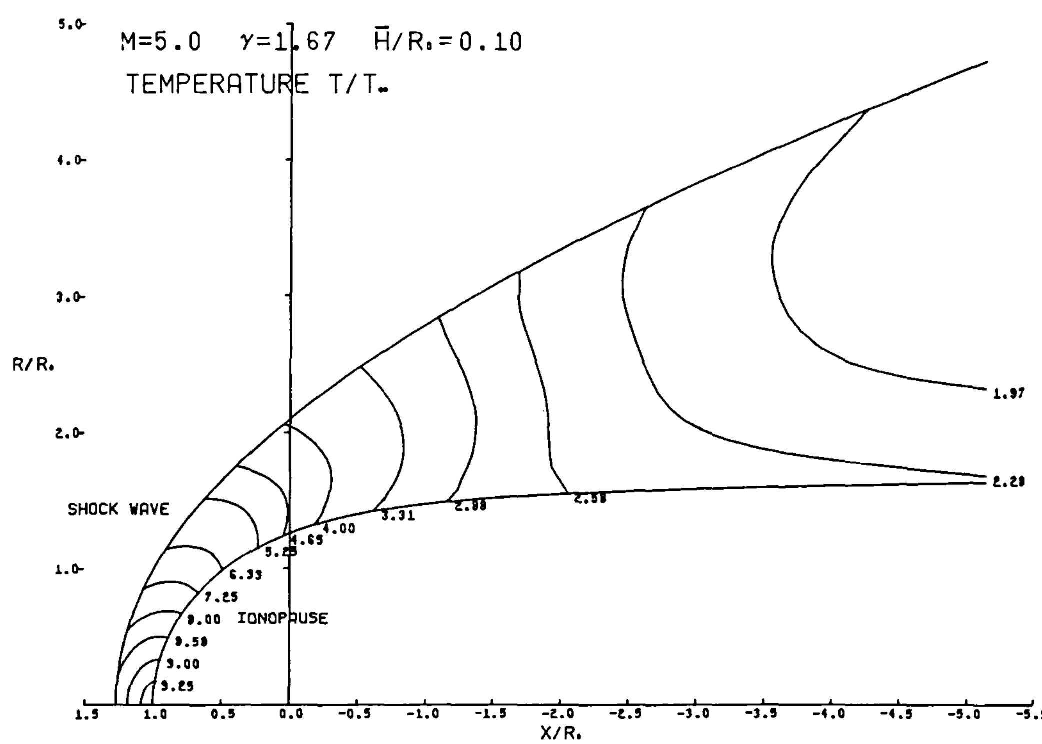

M=3.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
1.50
1.25
1.00
0.80
0.45
0.80
1.00
1.25

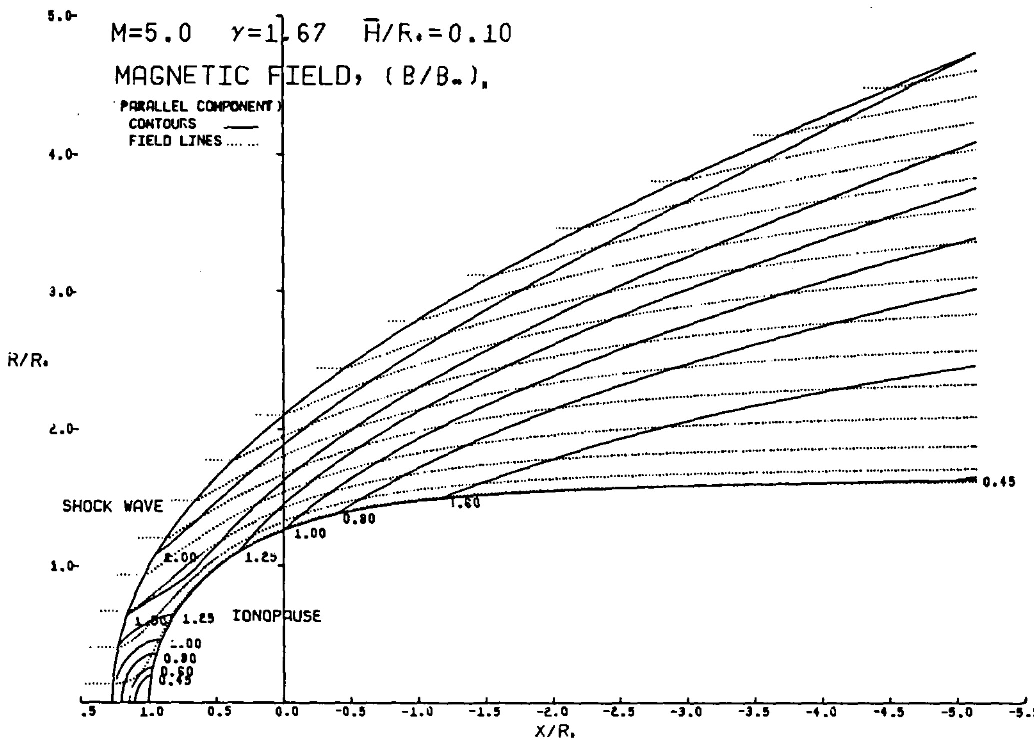

M=3.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.00
1.25
1.50
2.00
3.00
4.00
5.00
6.00

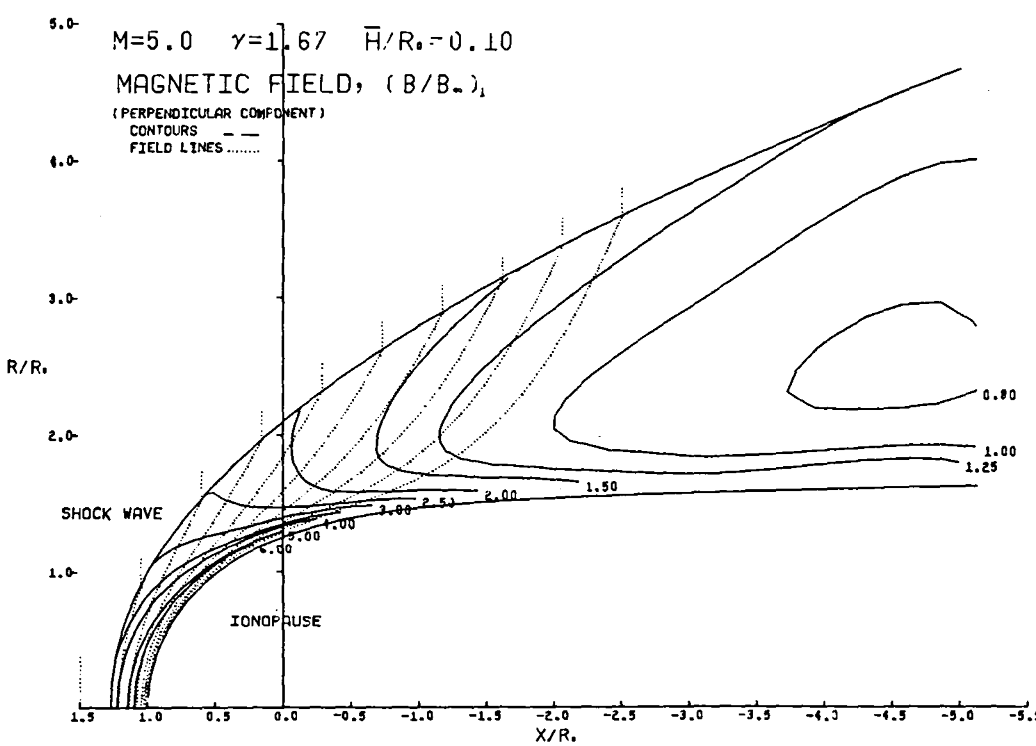

M=5.0 γ=1.67 H̄/R₀=0.10
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

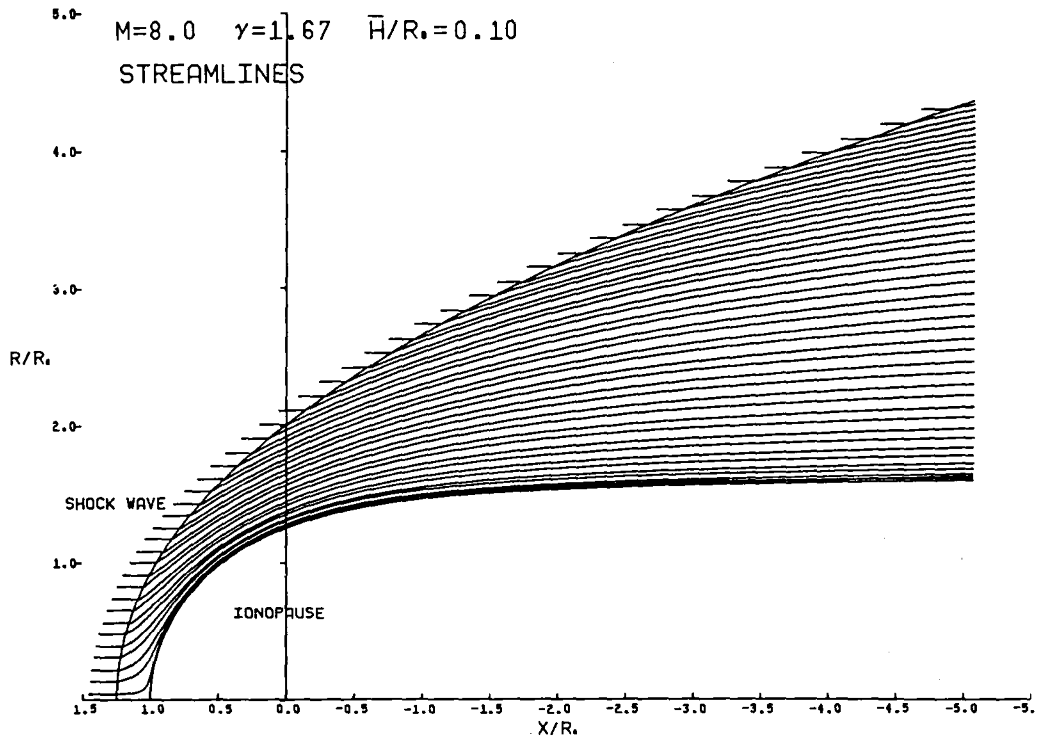

M=5.0 γ=1.67 H̄/R₀=0.10
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
0.94
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10
R/R₀
X/R₀
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

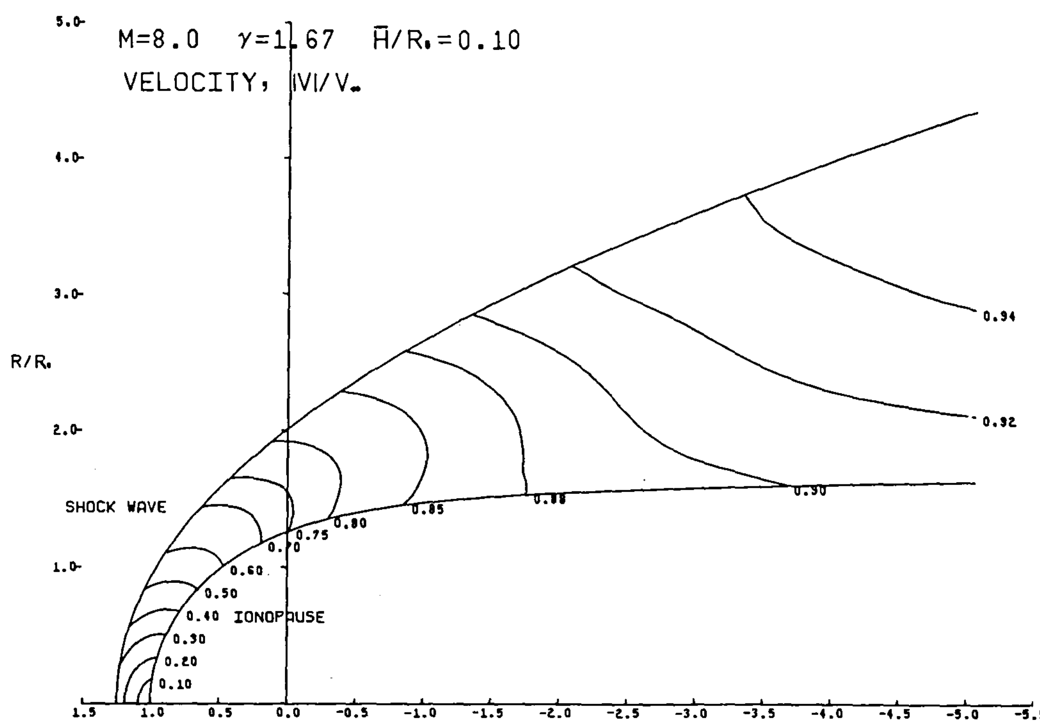

M=5.0 γ=1.67 H̄/R₀=0.10
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.50
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.90

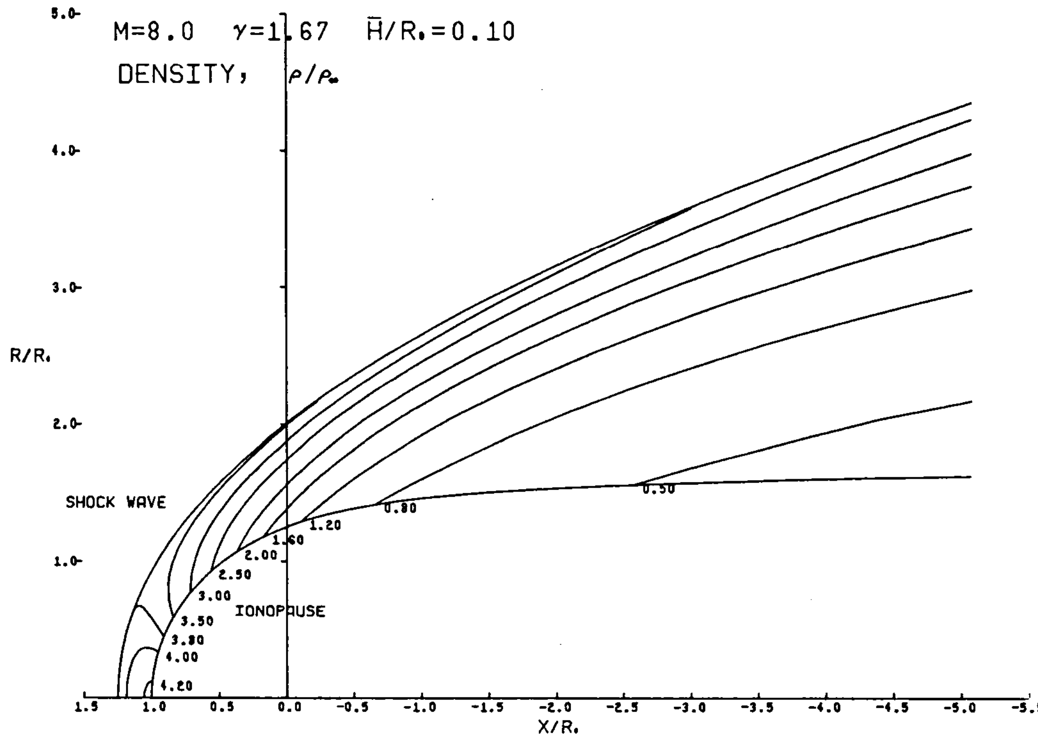

M=5.0 γ=1.67 H̄/R₀=0.10
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
1.97
2.29
2.58
2.88
3.31
4.00
4.65
5.25
6.33
7.25
9.00
9.59
9.00
9.25


M=5.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
PARALLEL COMPONENT
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.45
1.60
0.80
1.00
1.25
2.00

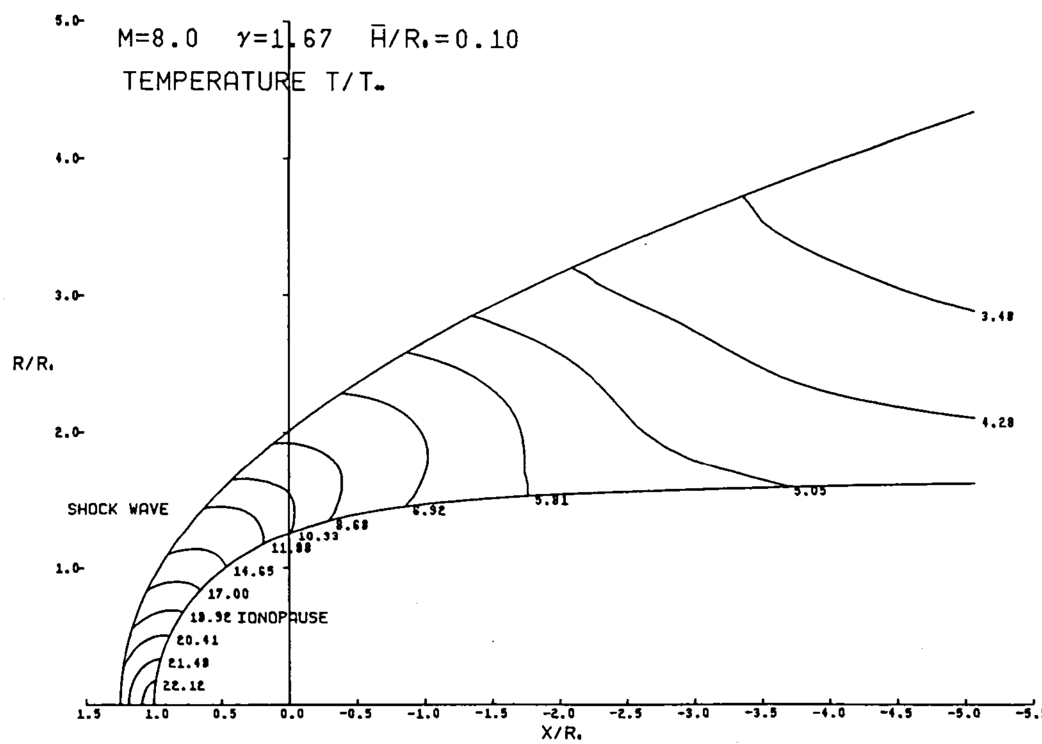

M=5.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.00
1.25
1.50
2.00
3.00
4.00
5.00
6.00


M=8.0 γ=1.67 H̄/R₀=0.10
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

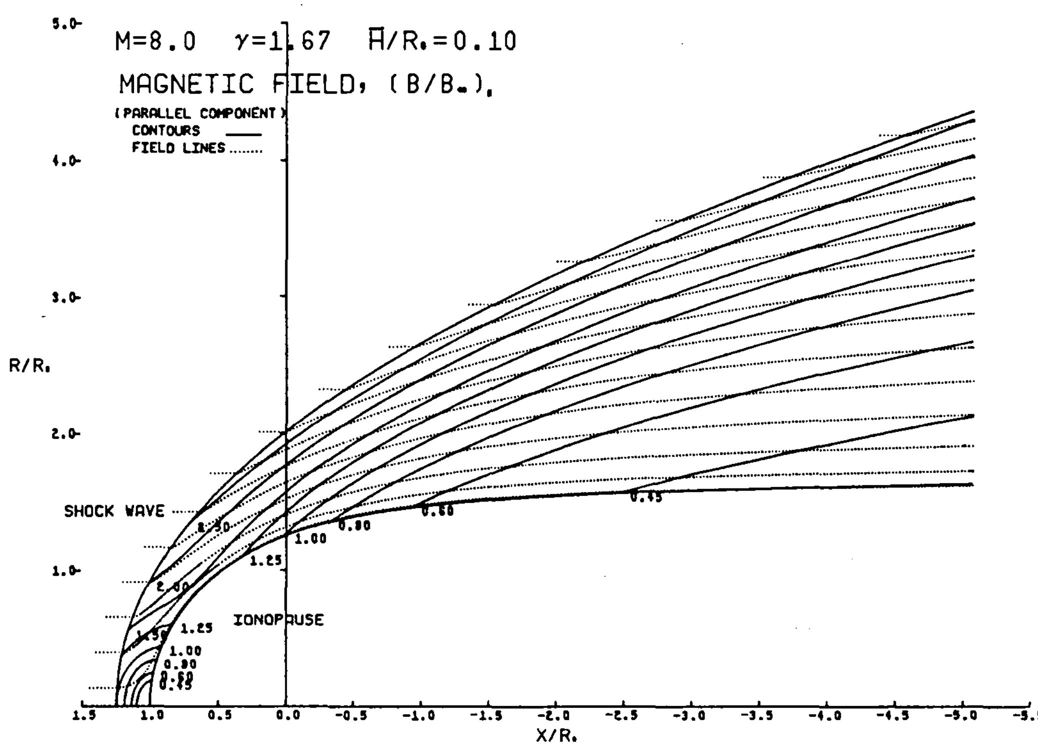

M=8.0 γ=1.67 H̄/R₀=0.10
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
0.94
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10
R/R₀
X/R₀



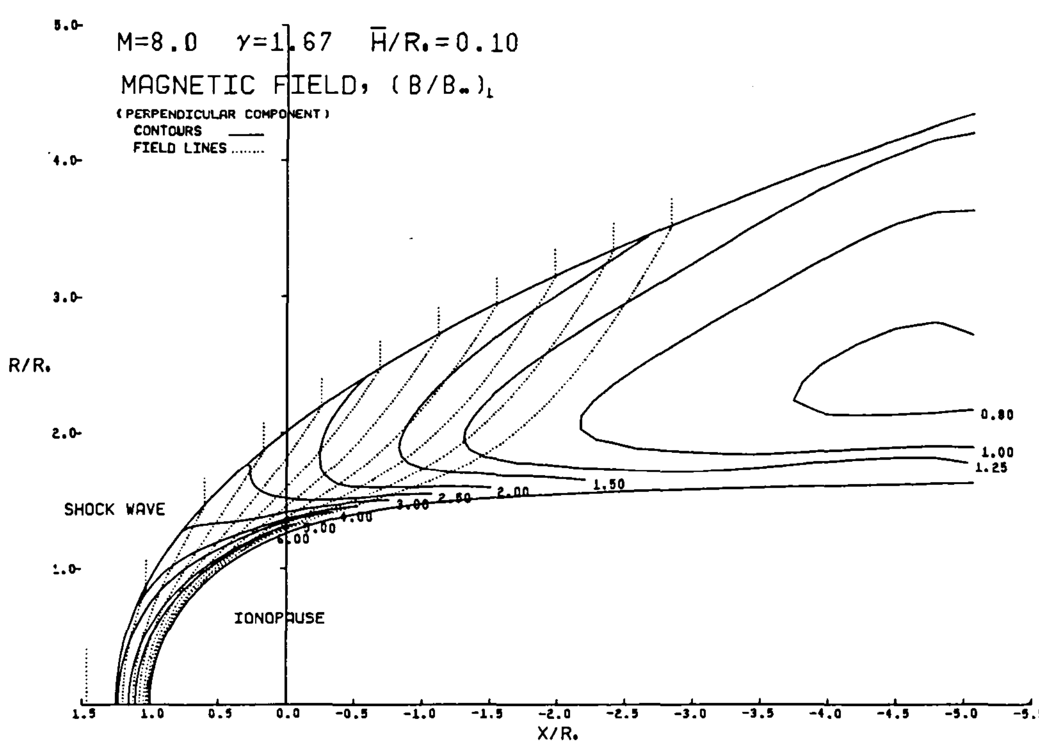
M=8.0 γ=1.67 H̄/R₀=0.10
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.50
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80
4.00
4.20



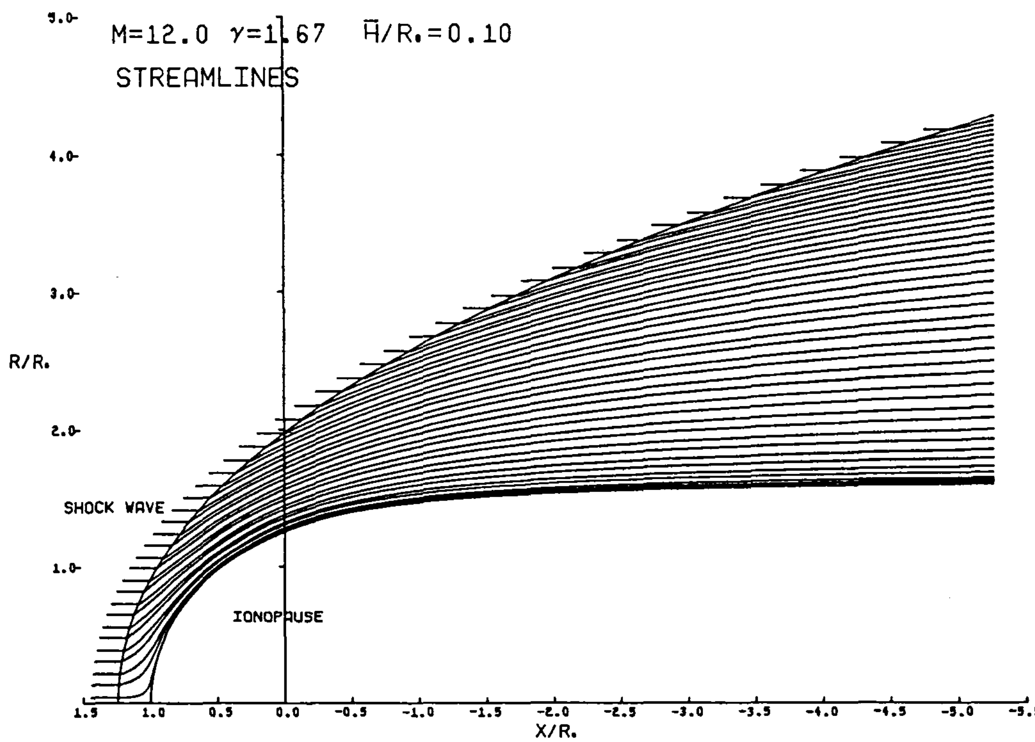
M=8.0 γ=1.67 H̄/R₀=0.10
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
3.48
4.28
5.05
5.81
6.92
8.68
10.33
11.88
14.65
17.00
18.92
20.41
21.48
22.12



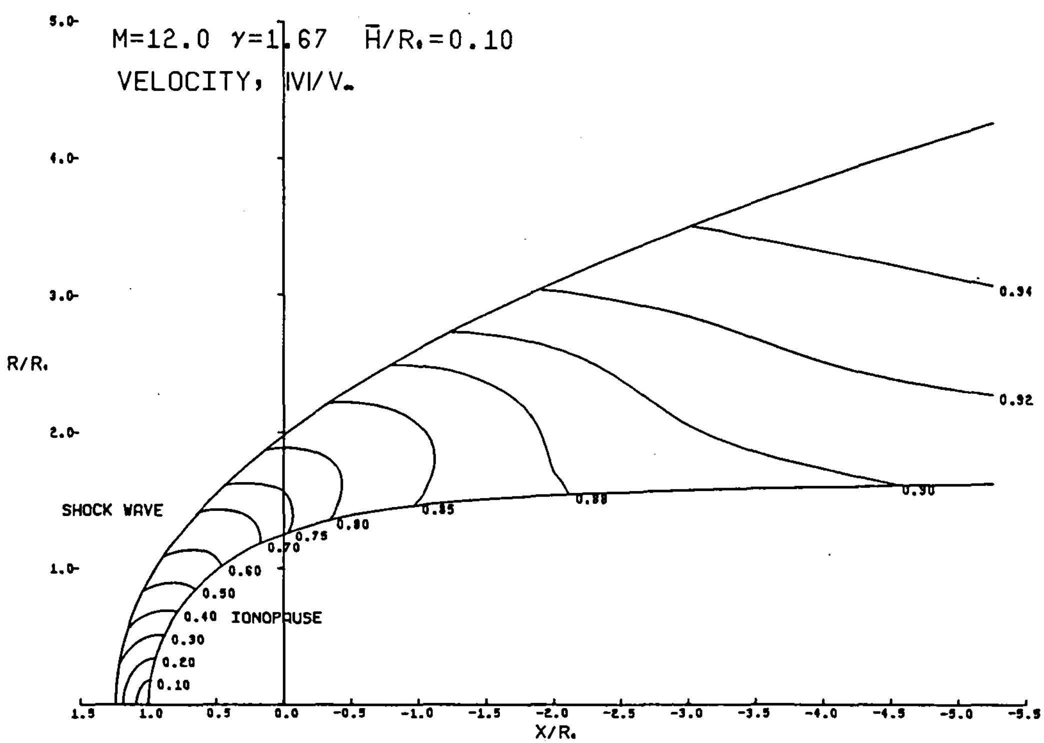
M=8.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀



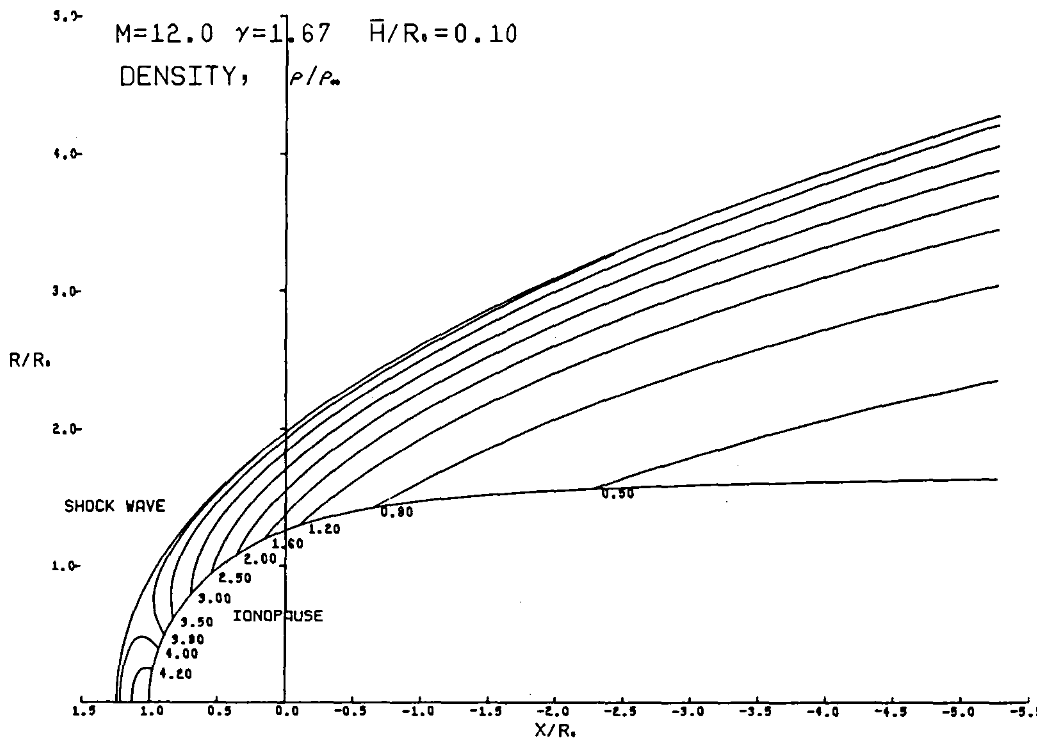
M=8.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

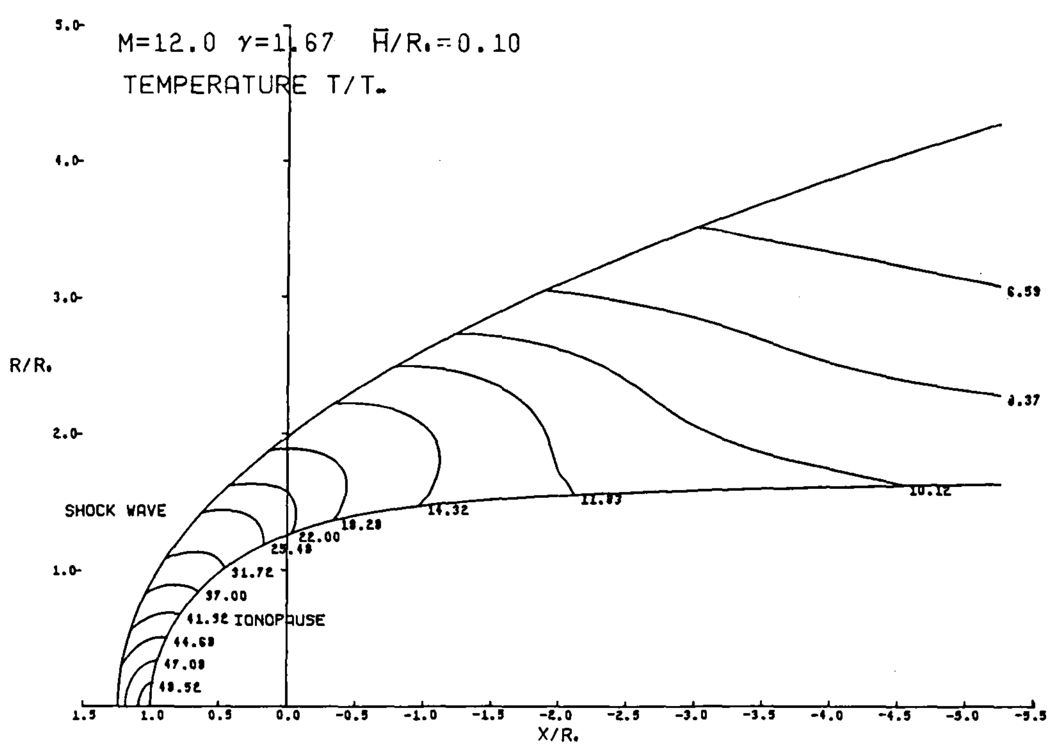


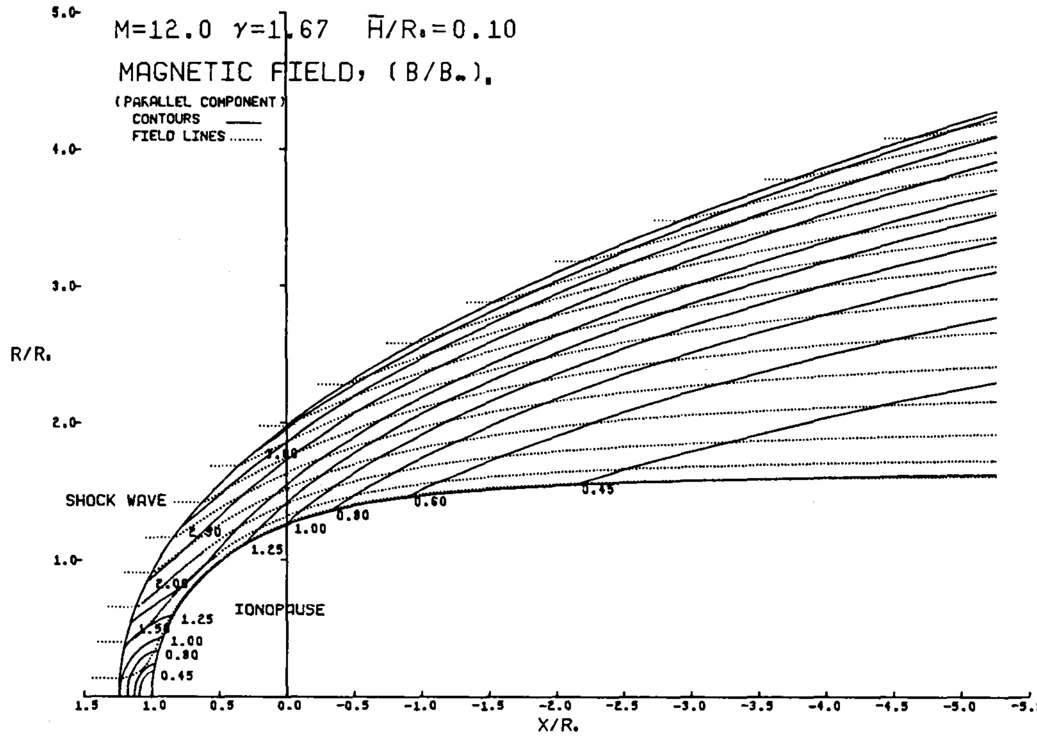
M=12.0 γ=1.67 H̄/R₀=0.10
STREAMLINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE

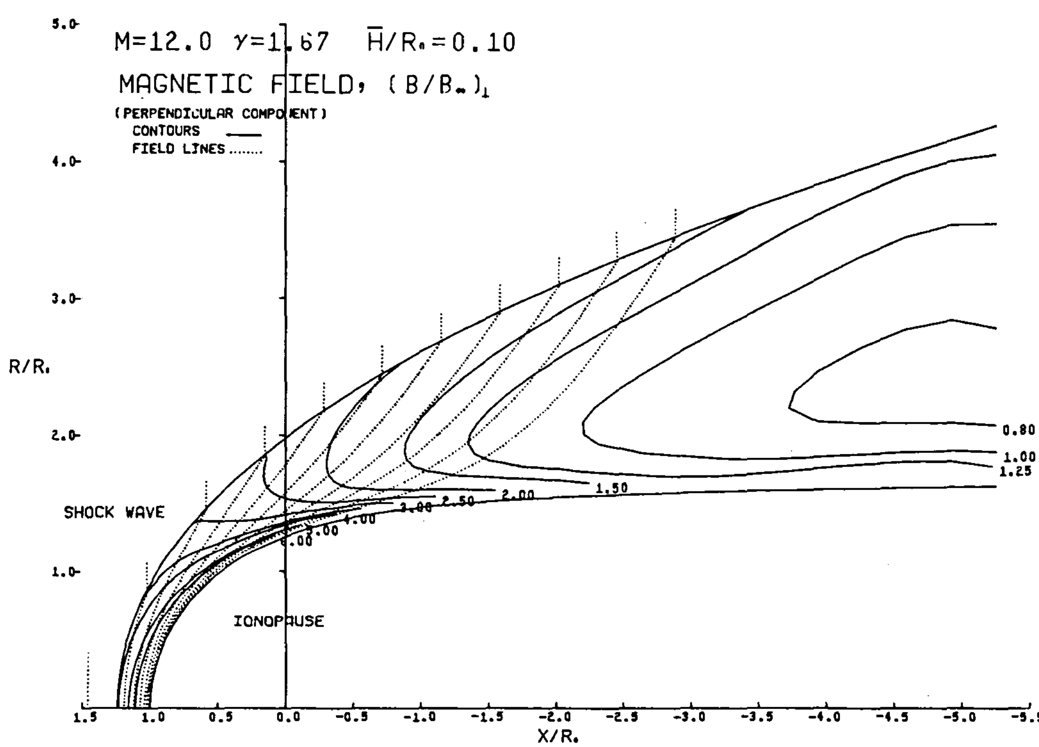


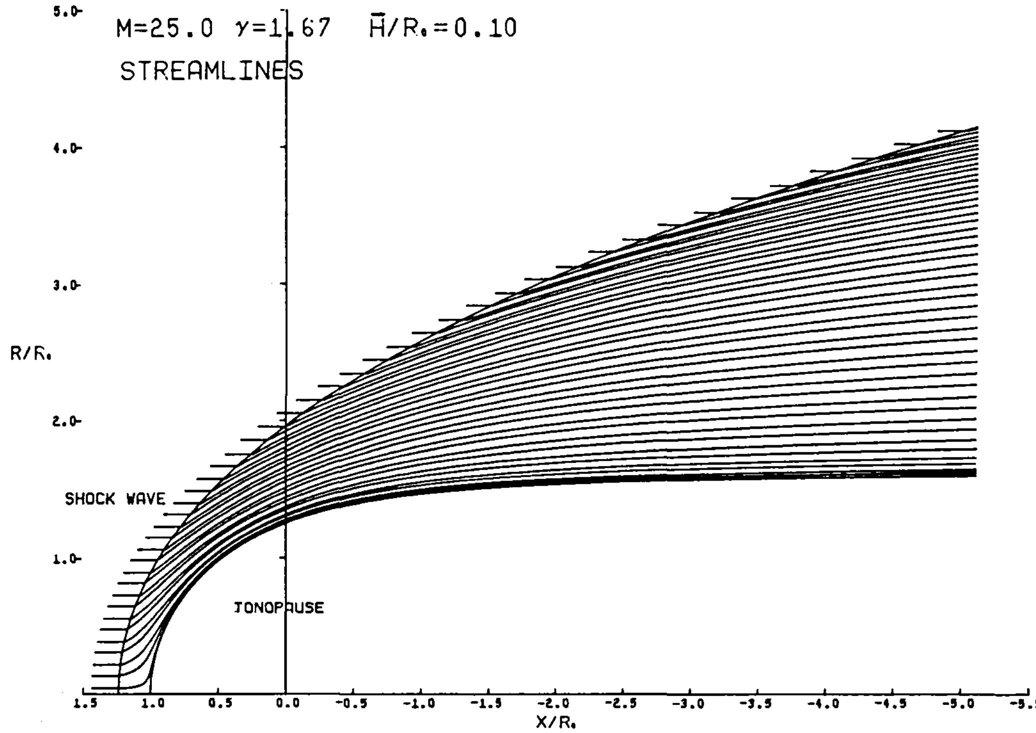
M=12.0 γ=1.67 H̄/R₀=0.10
VELOCITY, |V|/V∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.75
0.80
0.85
0.88
0.90
0.92
0.94


M=12.0 γ=1.67 H̄/R₀=0.10
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.50
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80
4.00
4.20

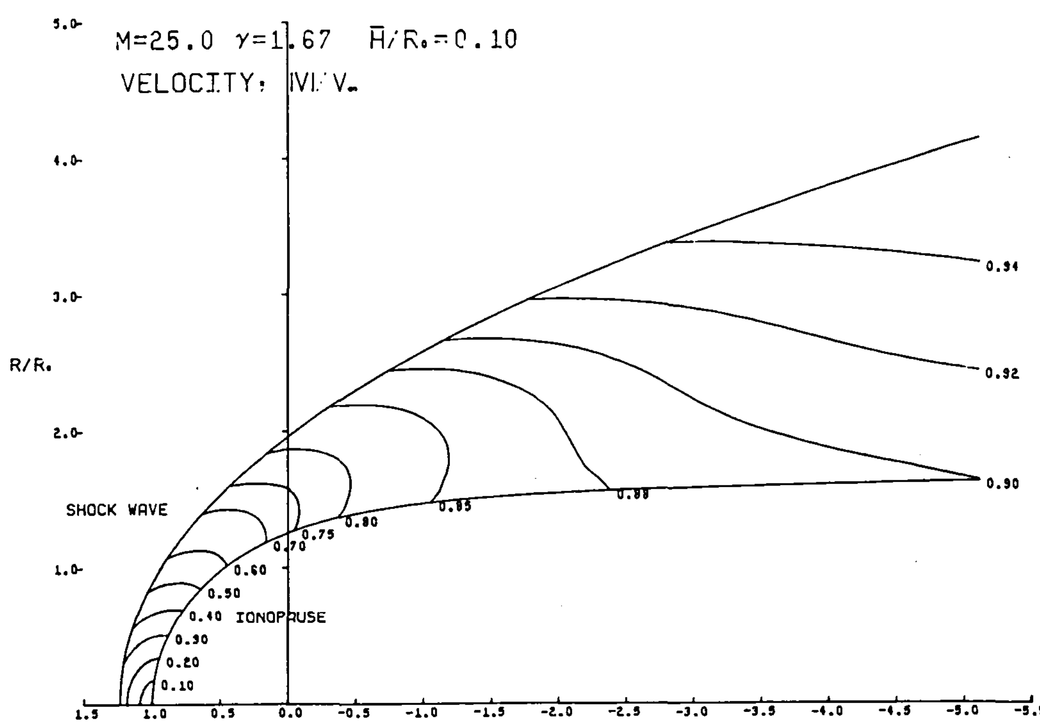

M=12.0 γ=1.67 H̄/R₀=0.10
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
6.59
8.37
10.12
11.83
14.32
18.28
22.00
25.48
31.72
37.00
41.32
44.68
47.08
48.52


M=12.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.45
0.60
0.80
1.00
1.25
1.50
2.00
2.50

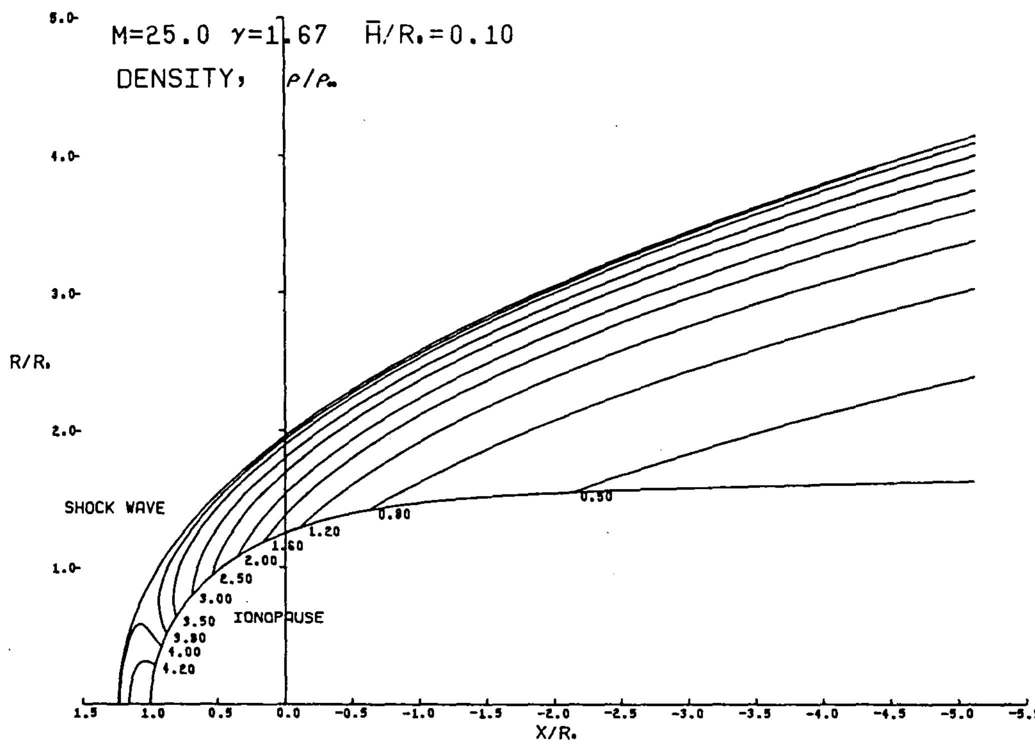

M=12.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.00
1.25
1.50
2.00
2.50
3.00
4.00



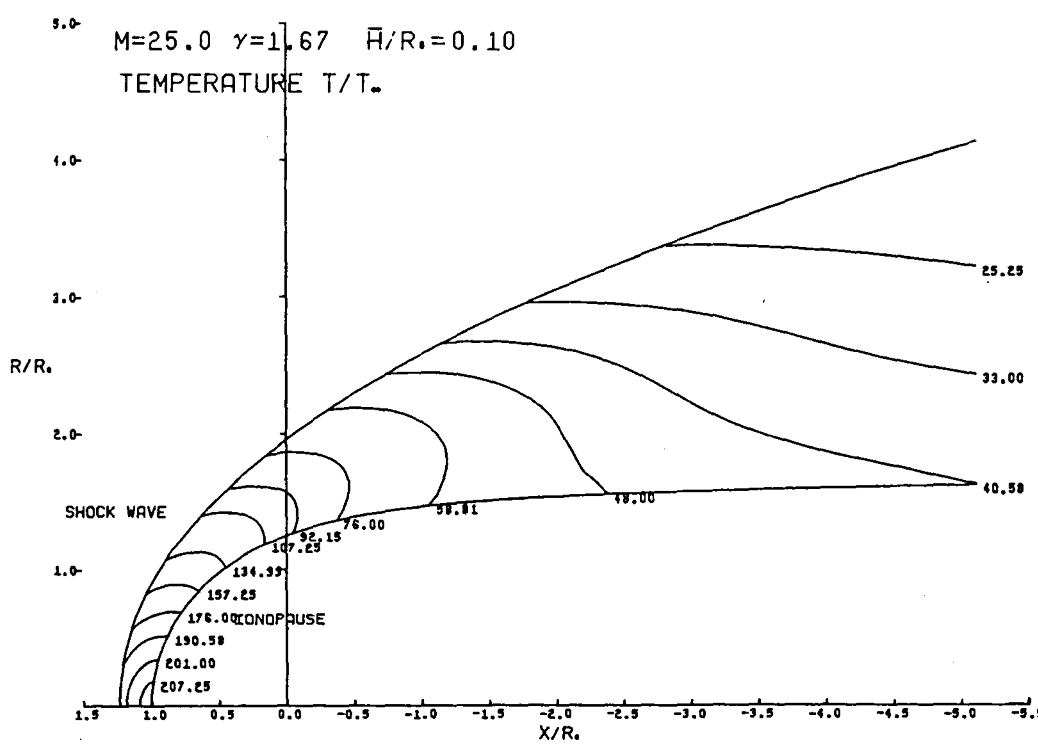
M=25.0 γ=1.67 H̄/R₀=0.10
STREAMLINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE



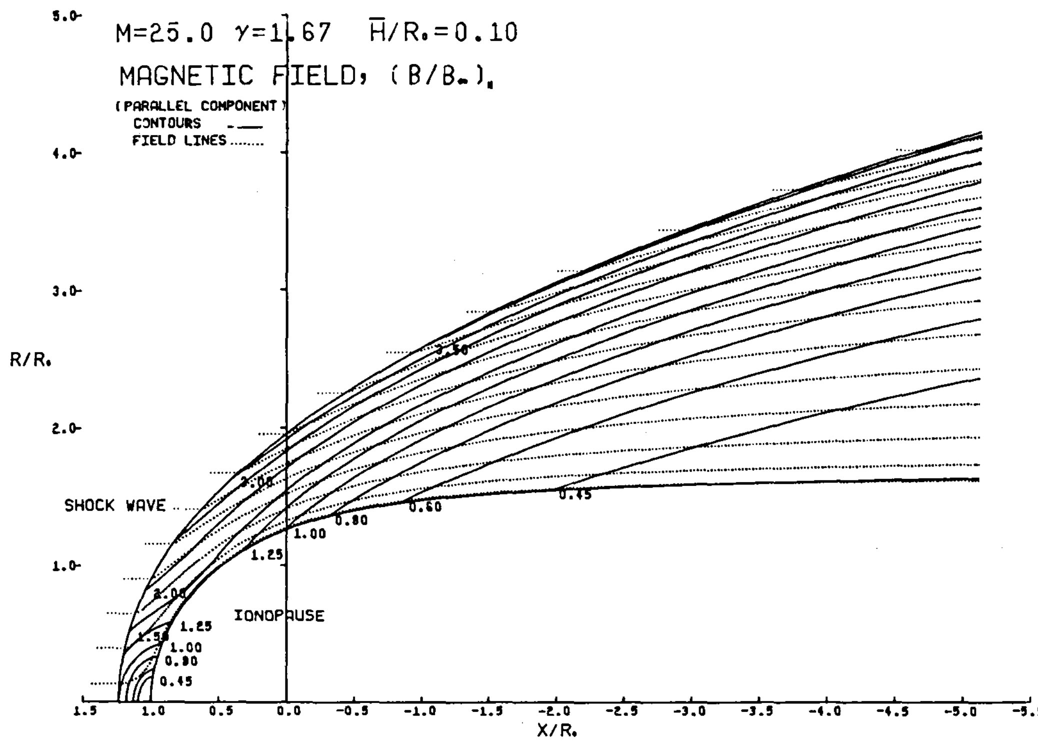
M=25.0 γ=1.67 H̄/R₀=0.10
VELOCITY: |V|/V∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.94
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10

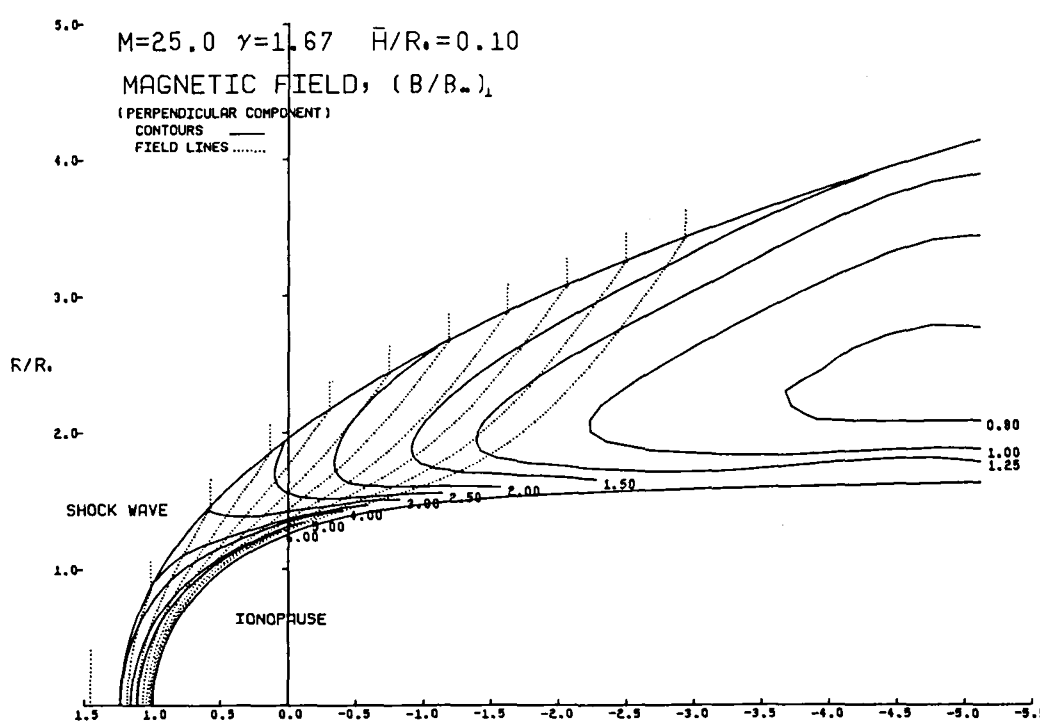


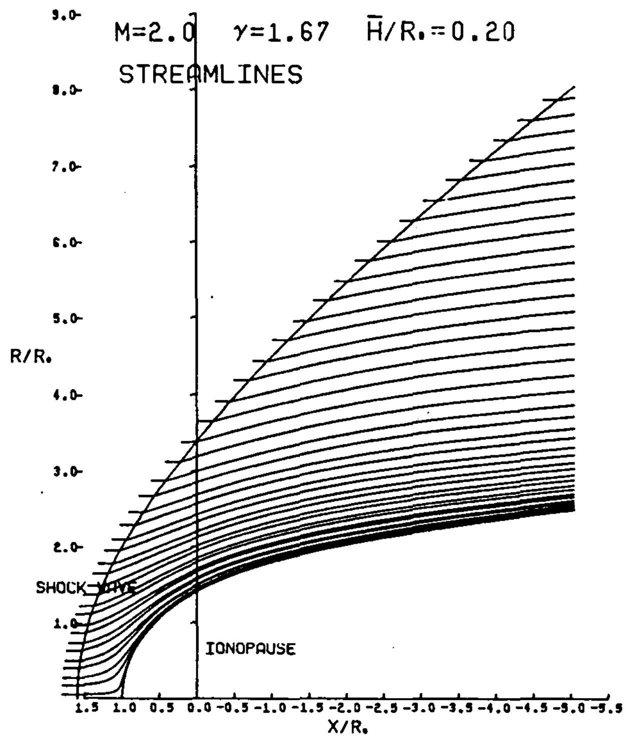
M=25.0 γ=1.67 H̄/R₀=0.10
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.50
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80
4.00
4.20

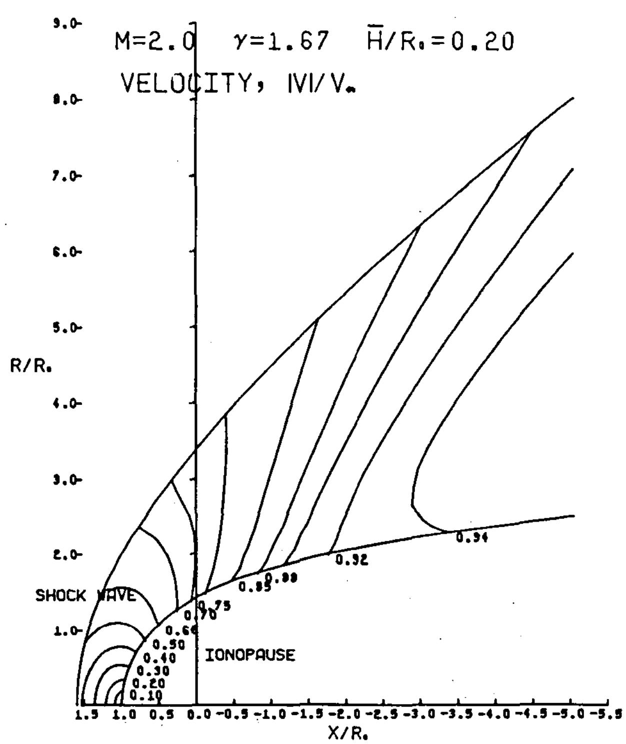


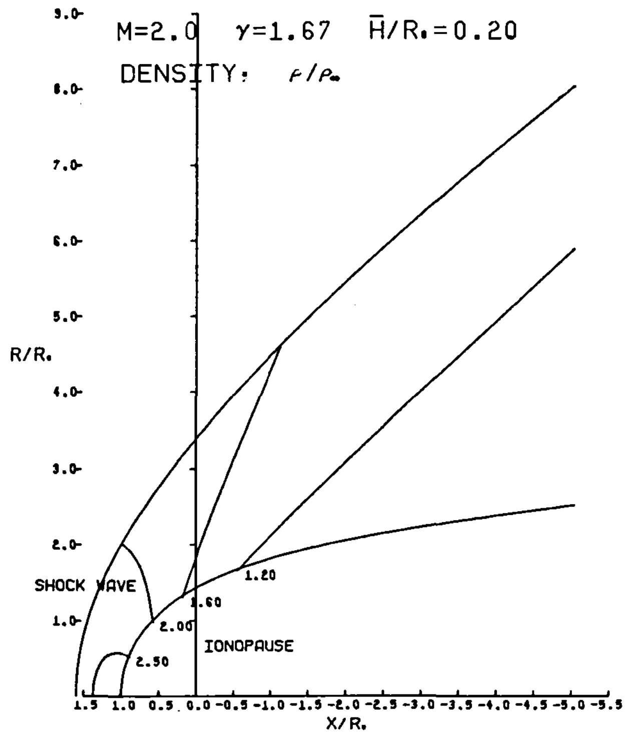
M=25.0 γ=1.67 H̄/R₀=0.10
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
25.25
33.00
40.58
48.00
58.81
76.00
92.15
107.25
134.33
157.25
176.00
190.58
201.00
207.25


M=25.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
2.50
3.00
2.00
1.50
1.25
1.00
0.80
0.60
0.45

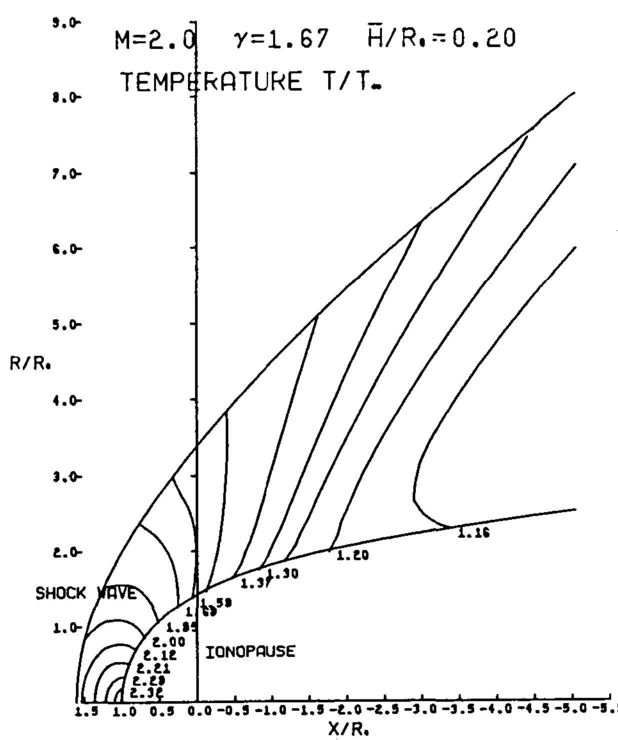

M=25.0 γ=1.67 H̄/R₀=0.10
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.00
1.25
1.50
2.00
2.50
3.00
4.00


M=2.0 γ=1.67 H̄/R₀=0.20
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

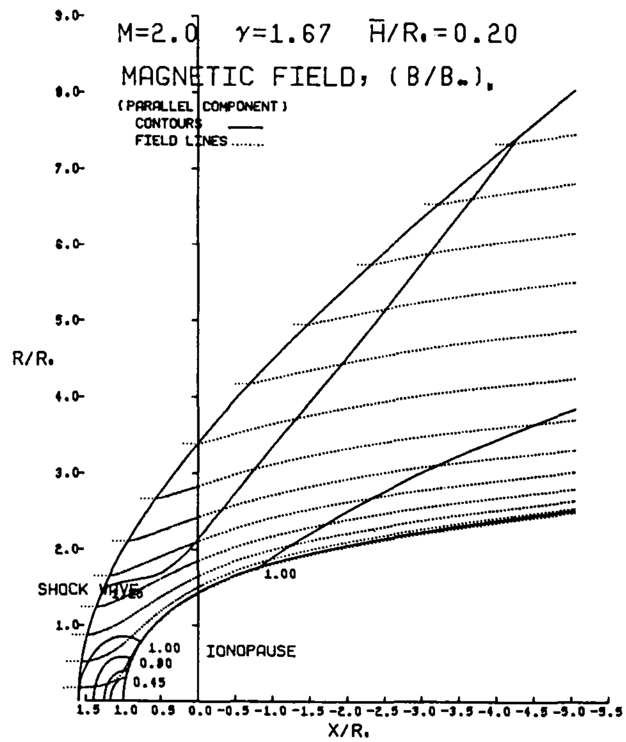

M=2.0 γ=1.67 H̄/R₀=0.20
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀
0.94
0.92
0.90
0.85
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10



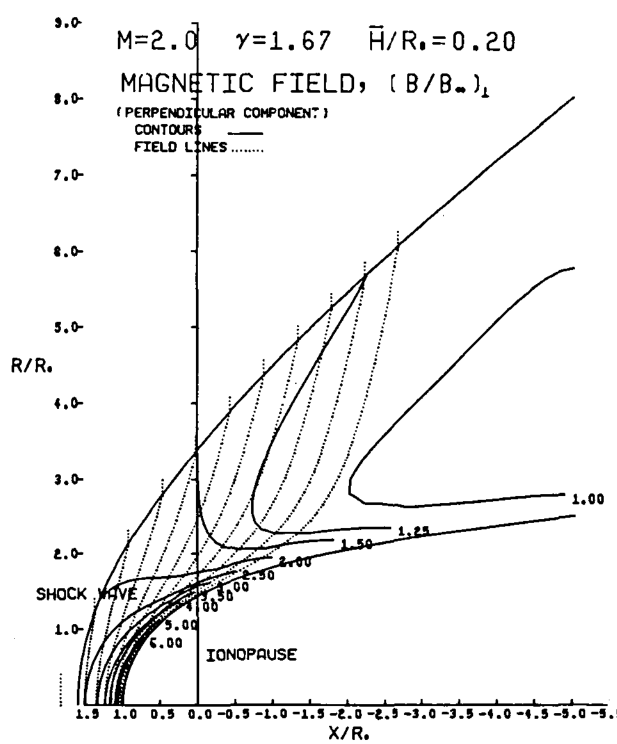
M=2.0 γ=1.67 H̄/R₀=0.20
DENSITY: ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.20
1.60
2.00
2.50



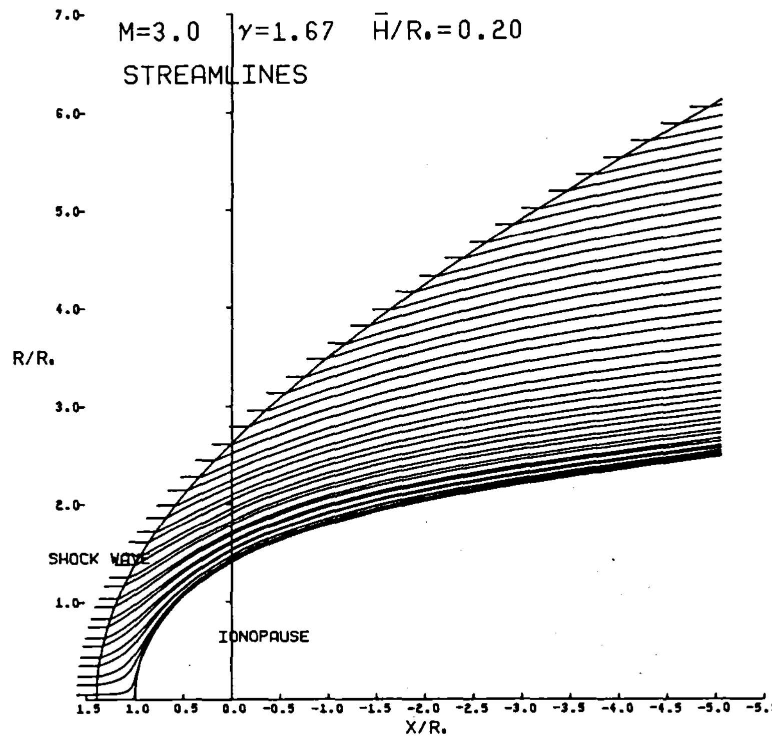
M=2.0 γ=1.67 H̄/R₀=0.20
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.16
1.20
1.30
1.37
1.59
1.85
2.00
2.12
2.21
2.29
2.36

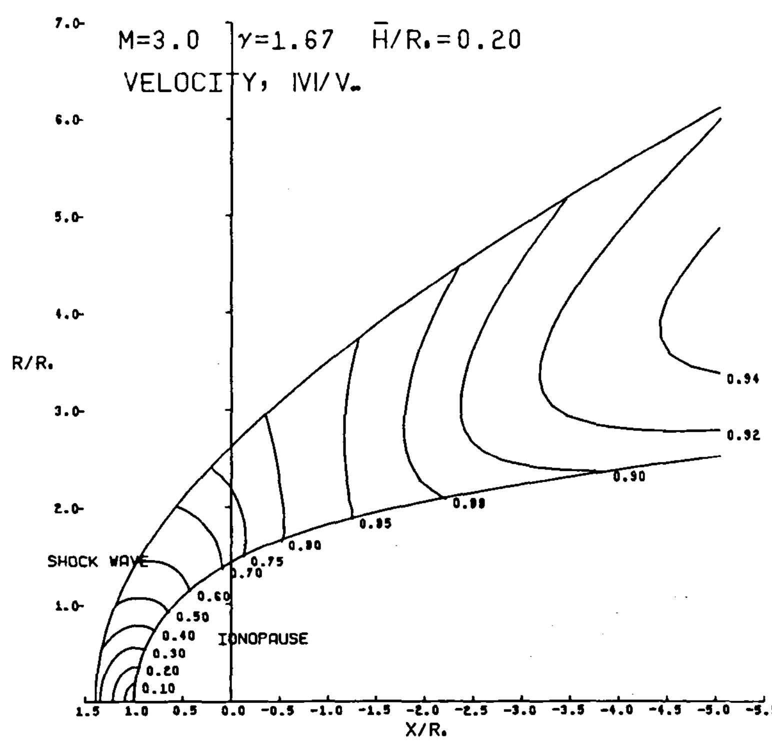


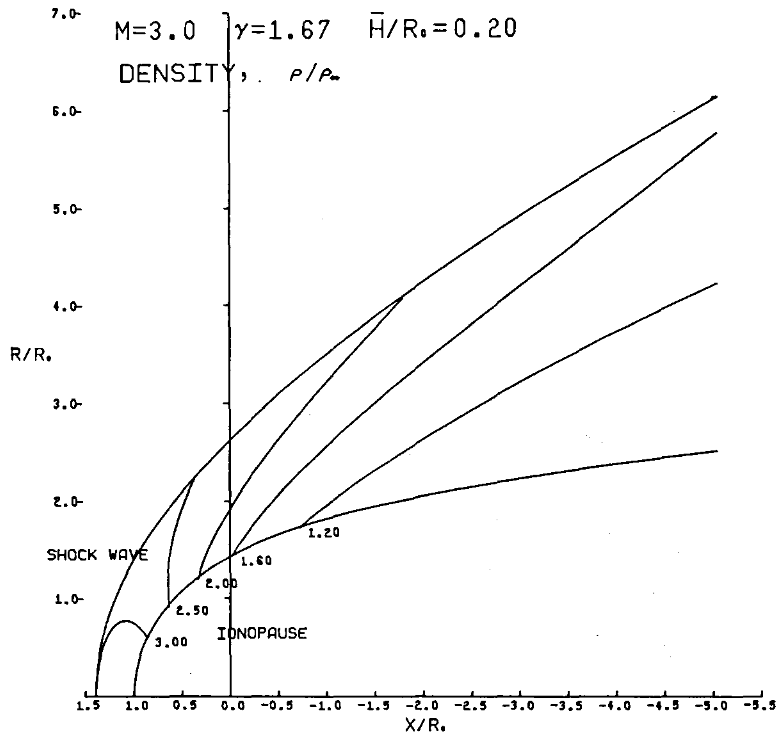
M=2.0 γ=1.67 H̄/R₀=0.20
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.00
0.80
0.45

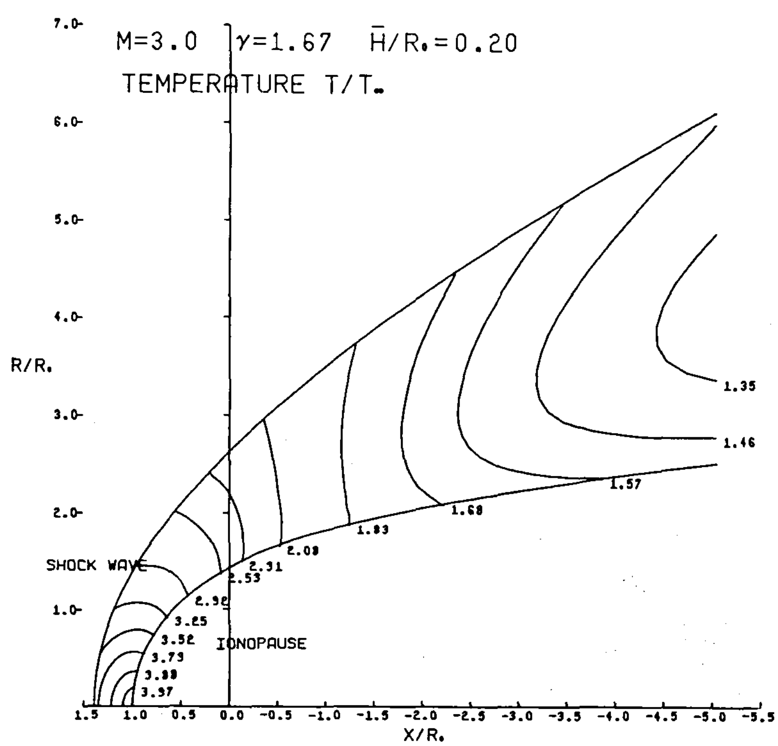


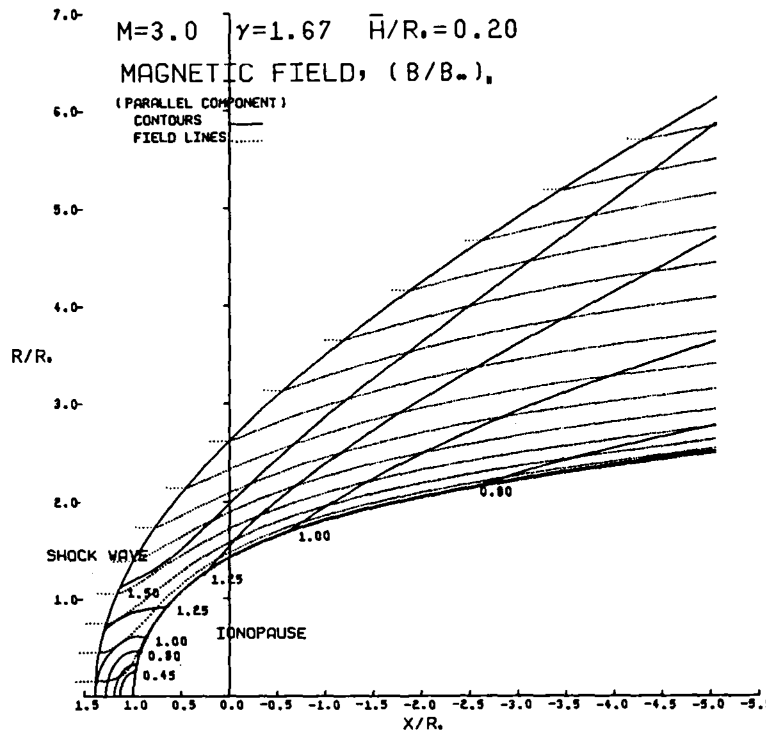
M=2.0 γ=1.67 H̄/R₀=0.20
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.00
1.25
1.50
2.00
2.50
5.00
6.00

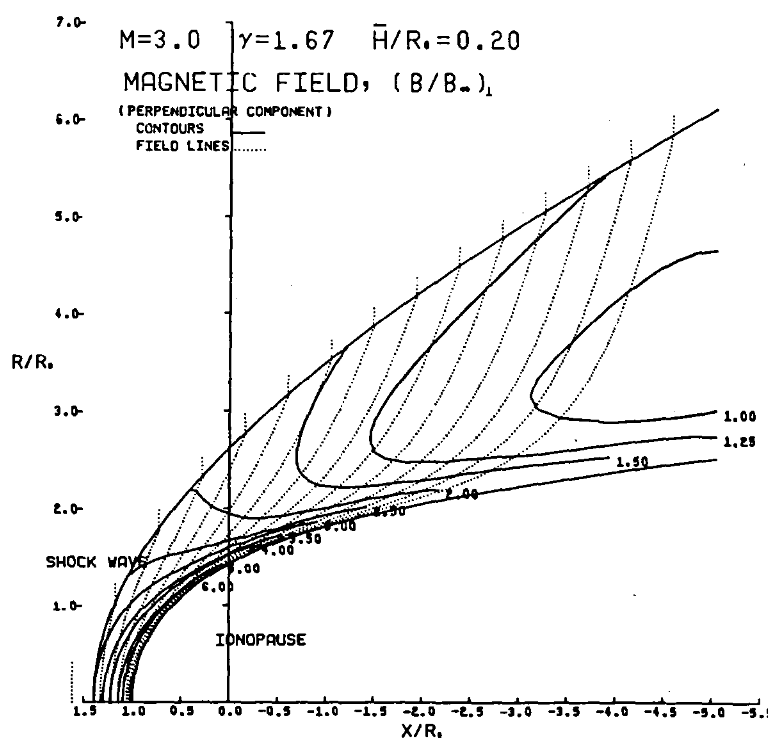

M=3.0 γ=1.67 H̄/R. =0.20
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R.
X/R.

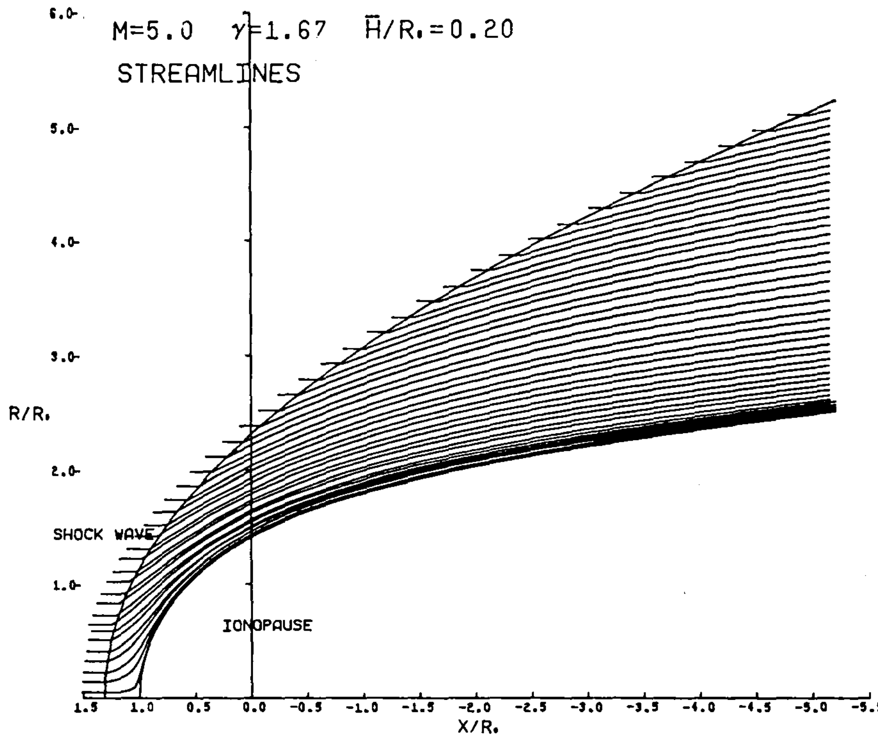

M=3.0 γ=1.67 H̄/R. =0.20
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
R/R.
X/R.
0.94
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10

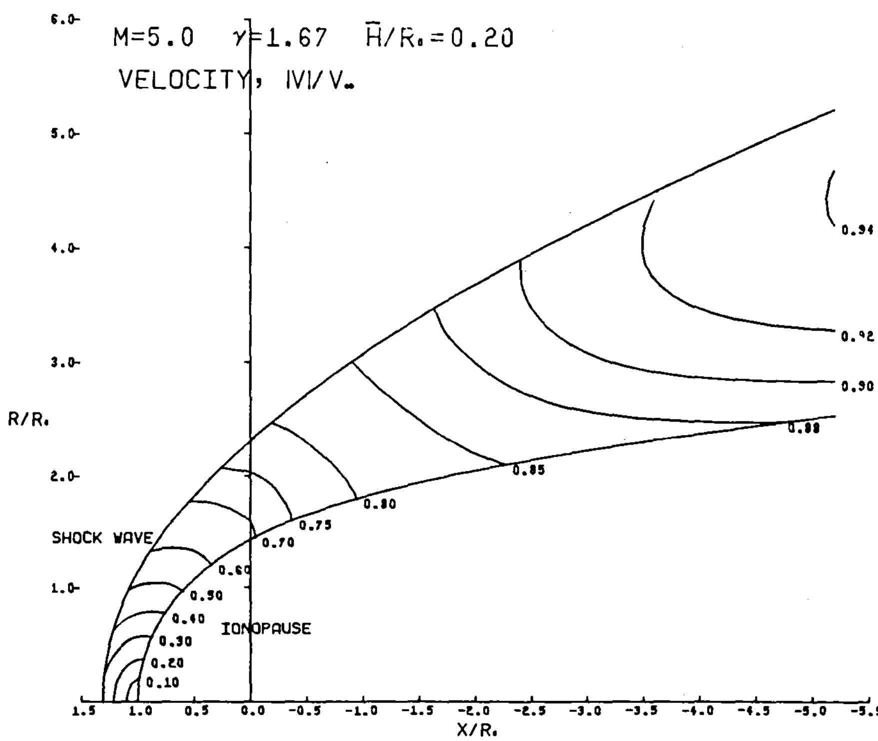

M=3.0 γ=1.67 H̄/R₀=0.20
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
3.00
2.50
2.00
1.60
1.20

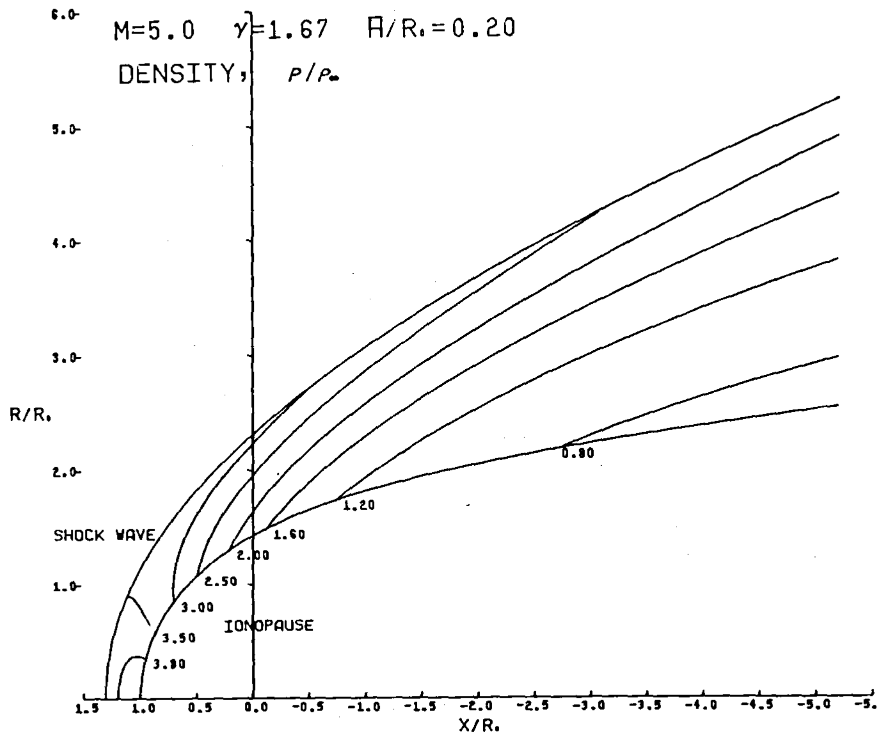

M=3.0 γ=1.67 H̄/R₀=0.20
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.35
1.46
1.57
1.68
1.83
2.09
2.31
2.53
2.92
3.25
3.52
3.73
3.88
3.97


M=3.0 γ=1.67 H̄/R₀=0.20
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.00
1.25
1.50
1.25
1.00
0.80
0.45
7.0
6.0
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

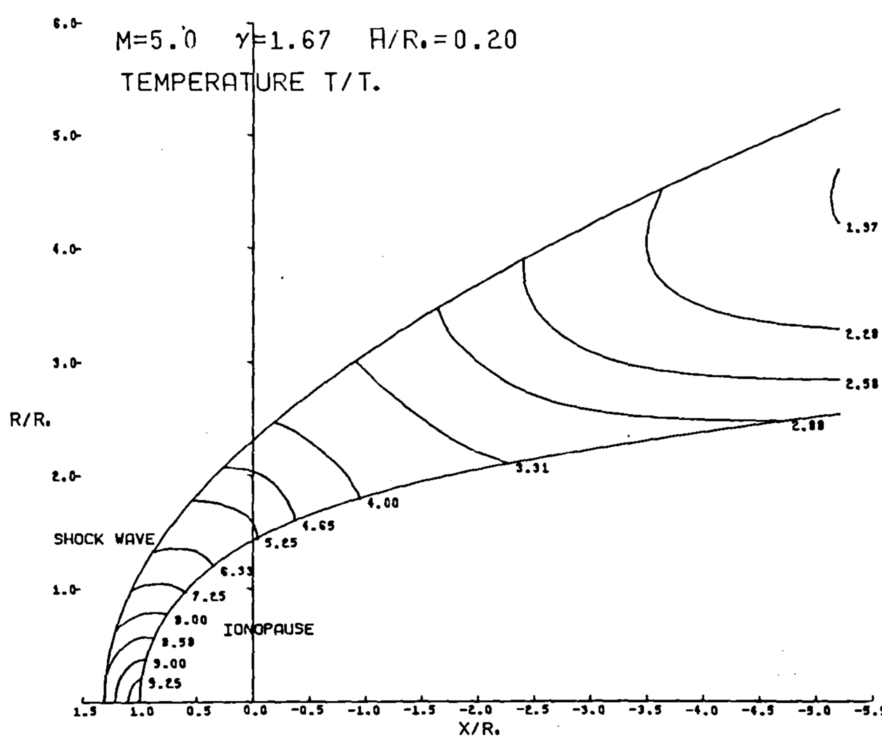

M=3.0 γ=1.67 H̄/R₀=0.20
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.00
1.25
1.50
2.00
6.00
7.0
6.0
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5


M=5.0 γ=1.67 H̄/R₀=0.20
STREAMLINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE

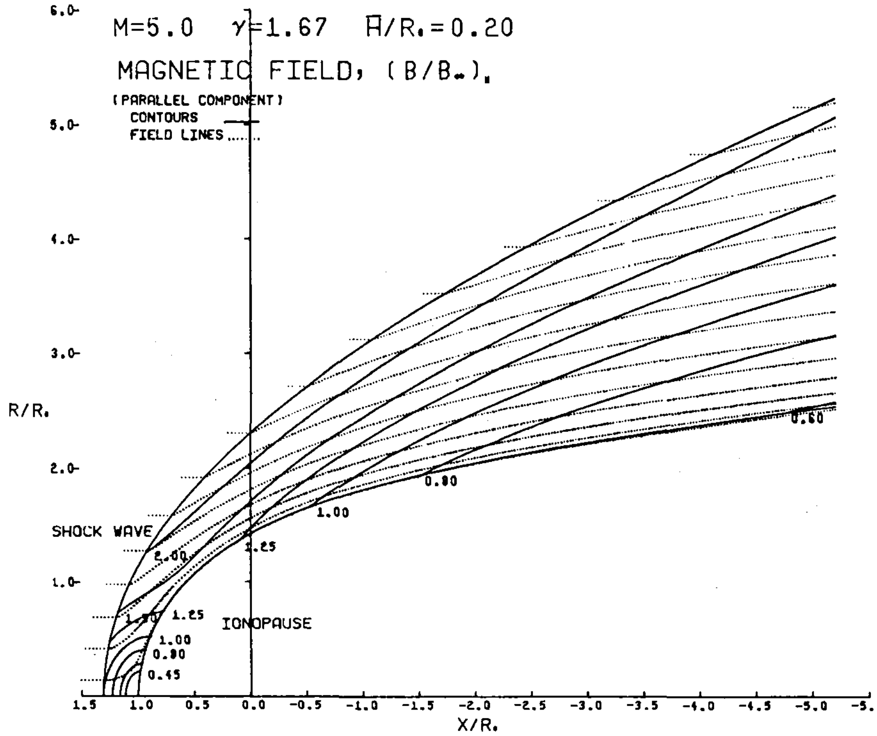

M=5.0 γ=1.67 H̄/R₀=0.20
VELOCITY, |V|/V∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.10
0.20
0.30
0.40
0.50
0.60
0.70
0.75
0.80
0.85
0.88
0.90
0.92
0.94

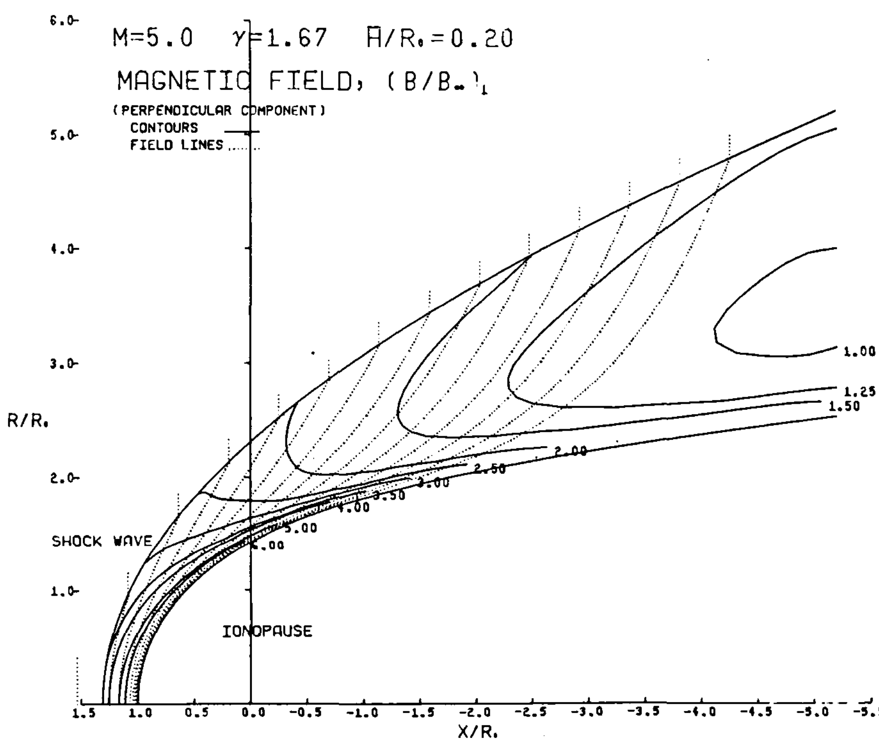


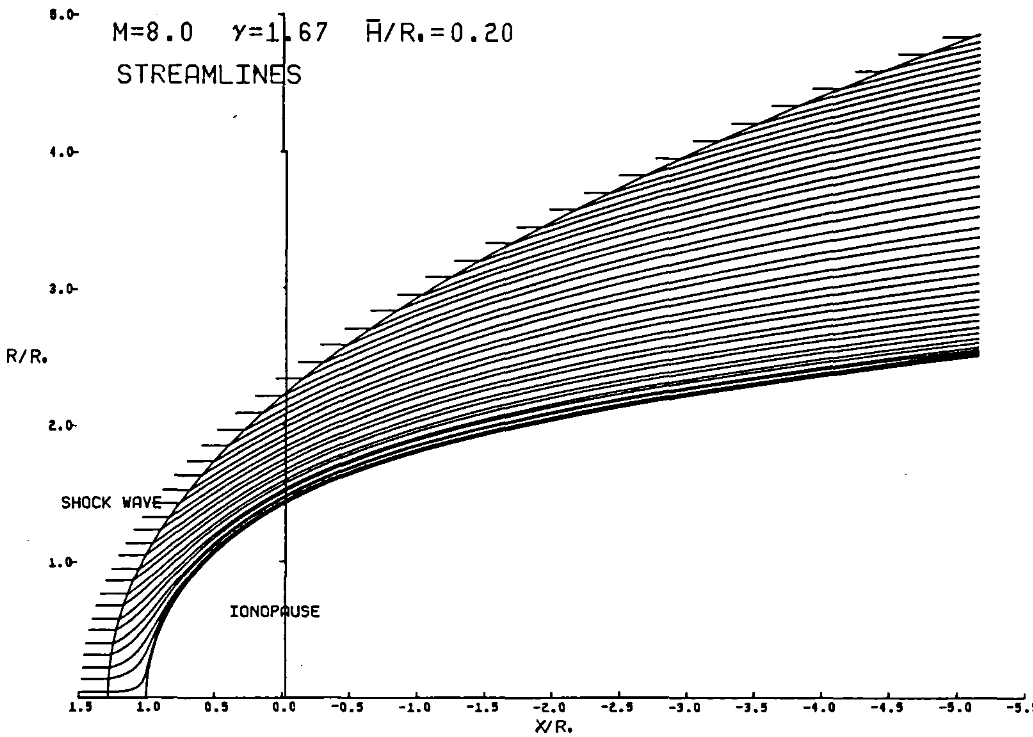
M=5.0 γ=1.67 A/R₀=0.20
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80



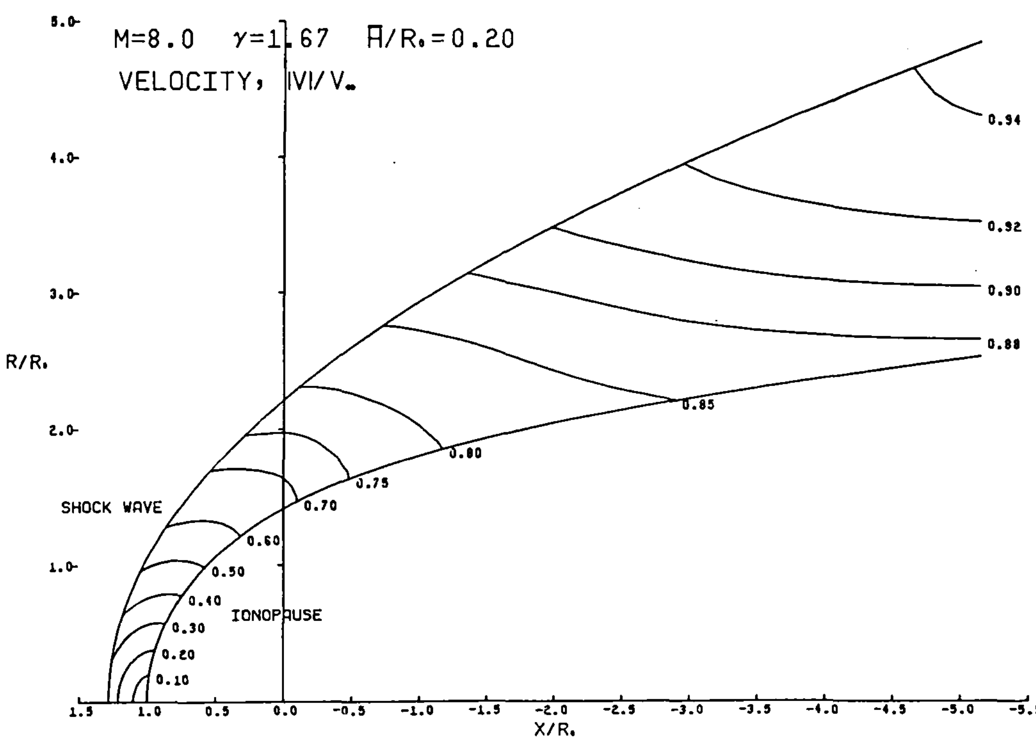
M=5.0 γ=1.67 A/R₀=0.20
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
1.37
2.28
2.58
2.88
3.31
4.00
4.65
5.25
6.33
7.25
8.00
8.58
9.00
9.25


M=5.0 γ=1.67 Ā/R₀=0.20
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
2.00
1.25
1.00
0.80
0.60
1.50
0.45

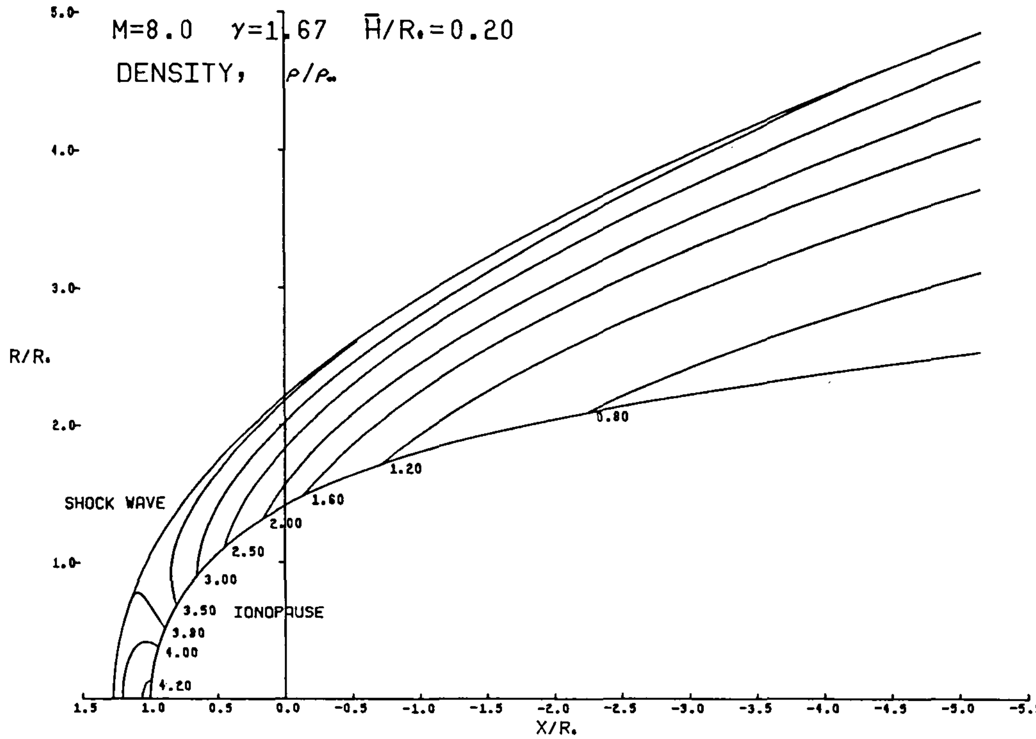

M=5.0 γ=1.67 Ā/R₀=0.20
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.00
1.25
1.50
2.00
2.50
3.00
3.50
4.00
5.00
6.00

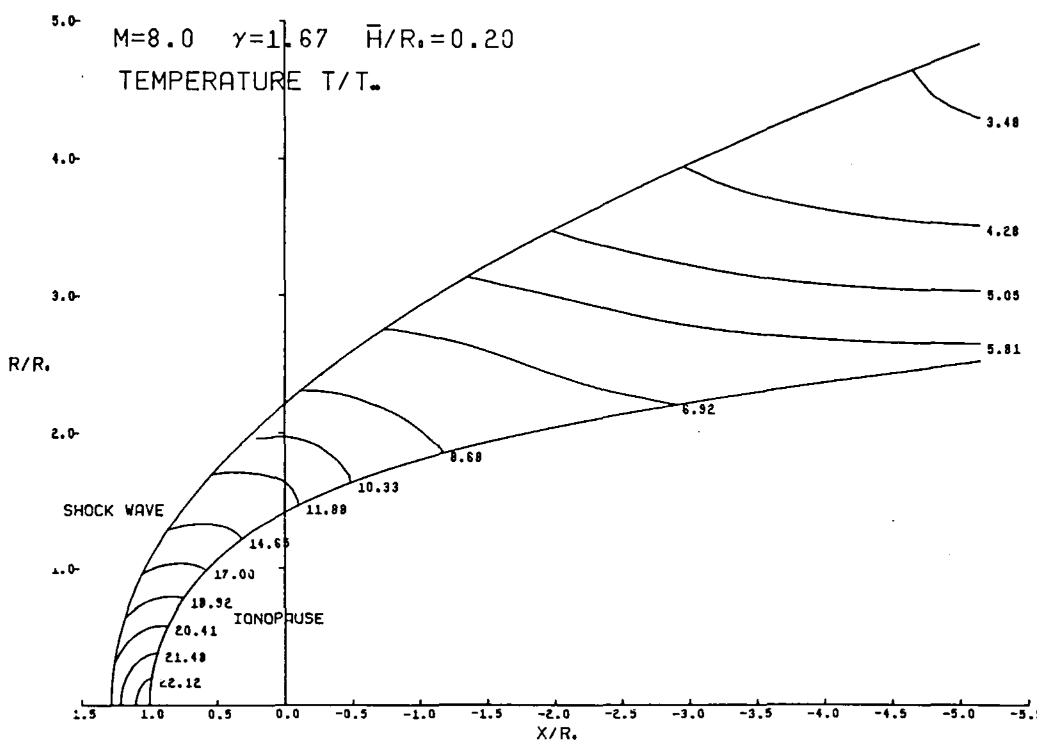


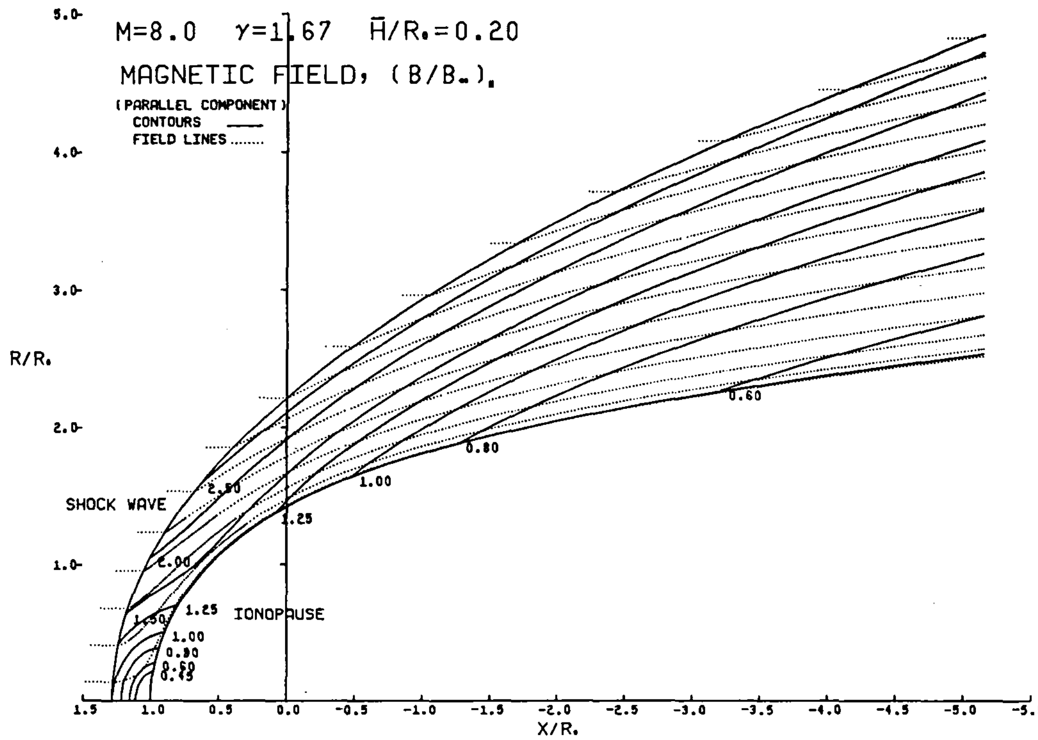
M=8.0 γ=1.67 Ā/R₀=0.20
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

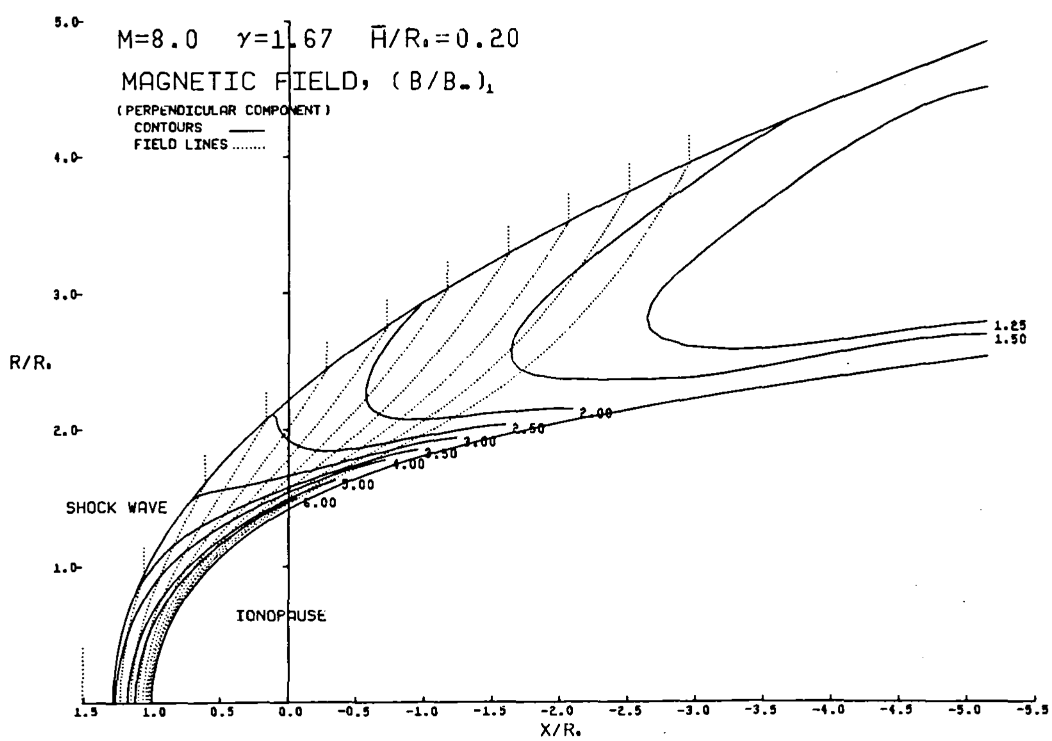


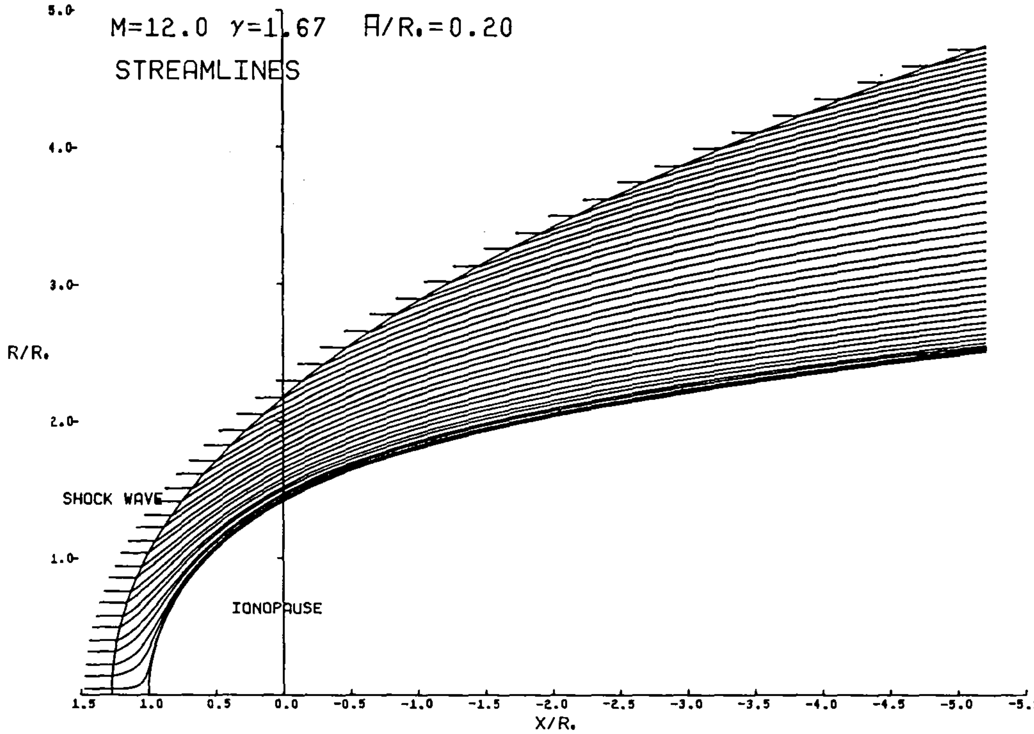
M=8.0 γ=1.67 Ā/R₀=0.20
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
0.94
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10
R/R₀
X/R₀

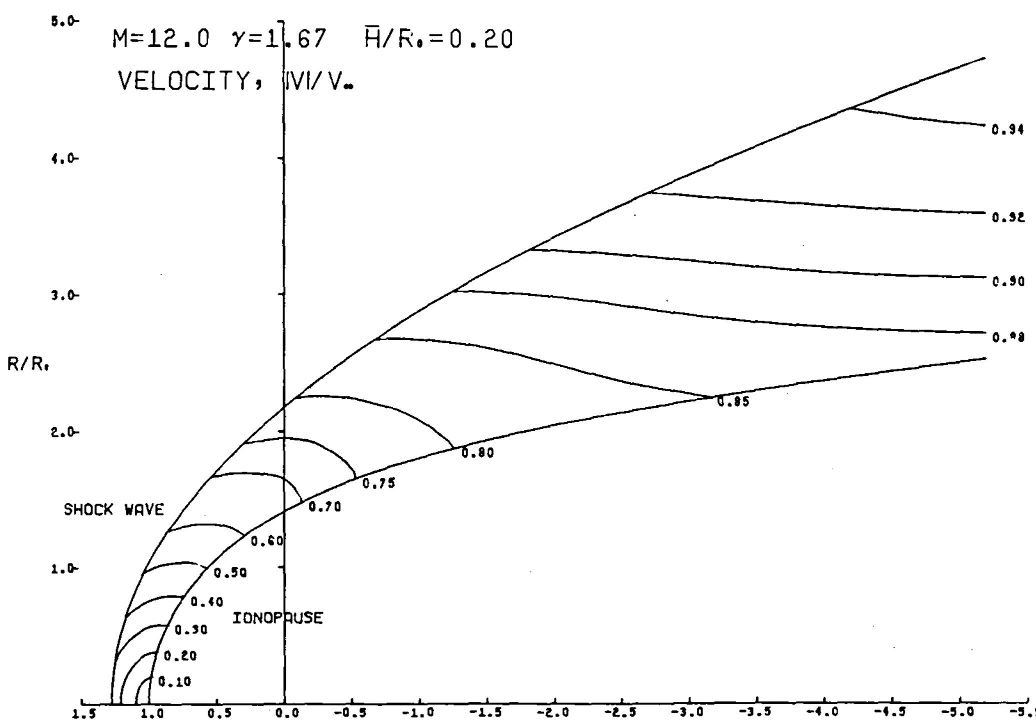

M=8.0 γ=1.67 H̄/R₀=0.20
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80
4.00
4.20

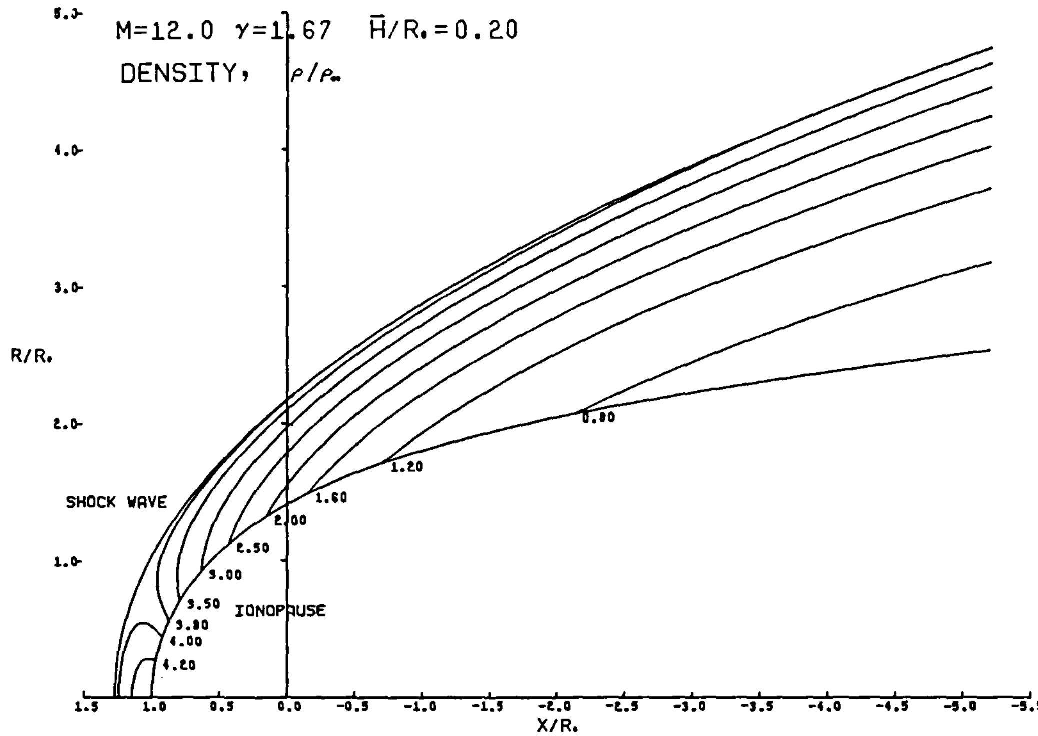

M=8.0 γ=1.67 H̄/R₀=0.20
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
3.48
4.28
5.05
5.81
6.92
8.68
10.33
11.88
14.65
17.00
19.92
20.41
21.48
22.12

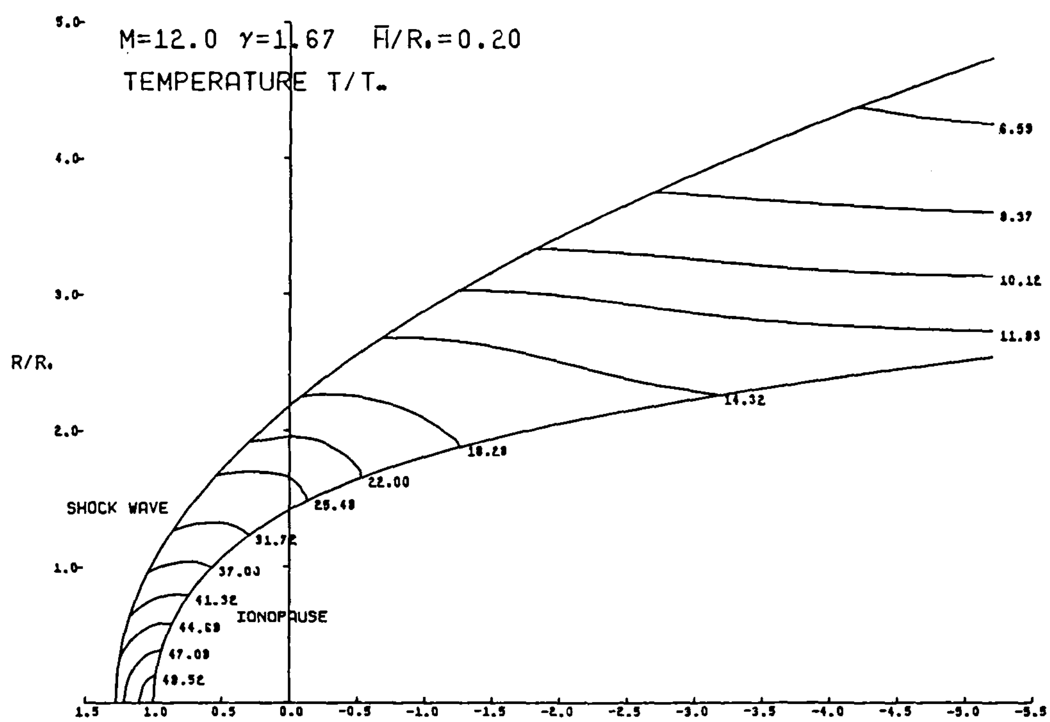

M=8.0 γ=1.67 H̄/R₀=0.20
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
2.50
2.00
1.50
1.25
1.00
0.90
0.60
0.45
0.80

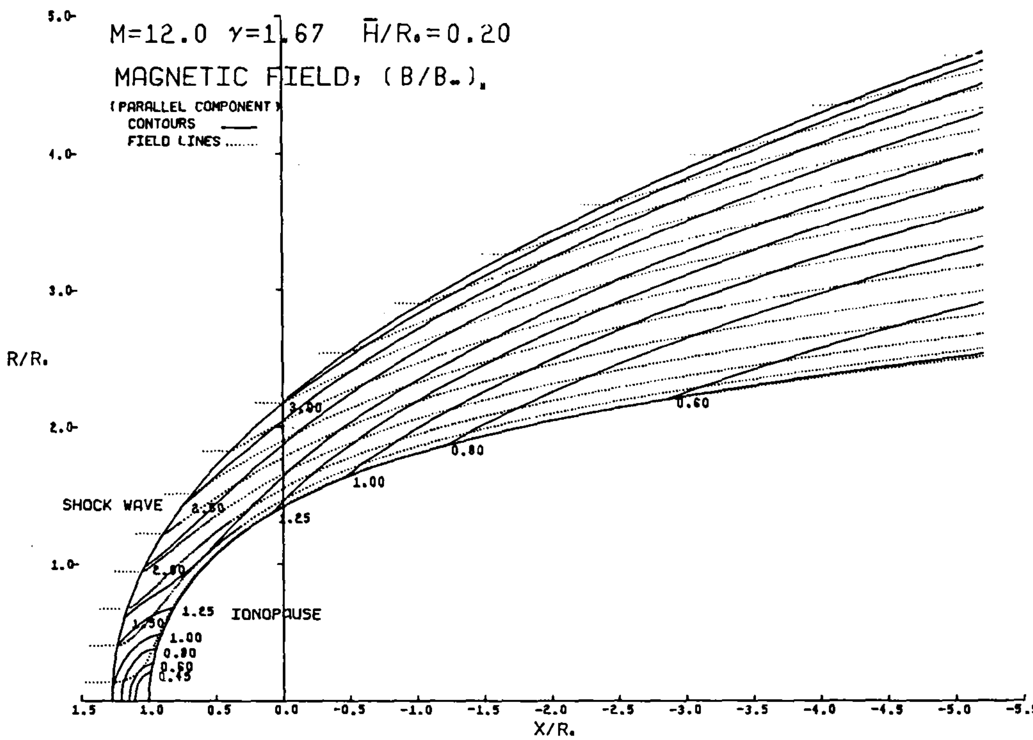

M=8.0 γ=1.67 H̄/R₀=0.20
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
1.25
1.50
2.00
2.50
3.00
3.50
4.00
5.00
6.00

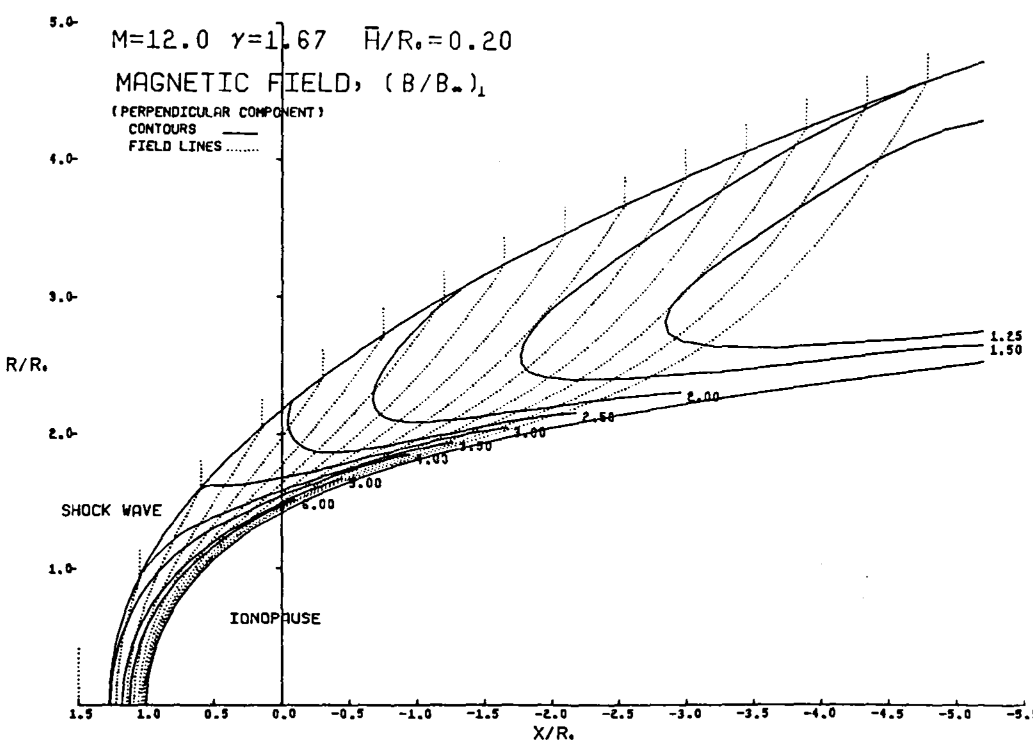

M=12.0 γ=1.67 Ā/R₀=0.20
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

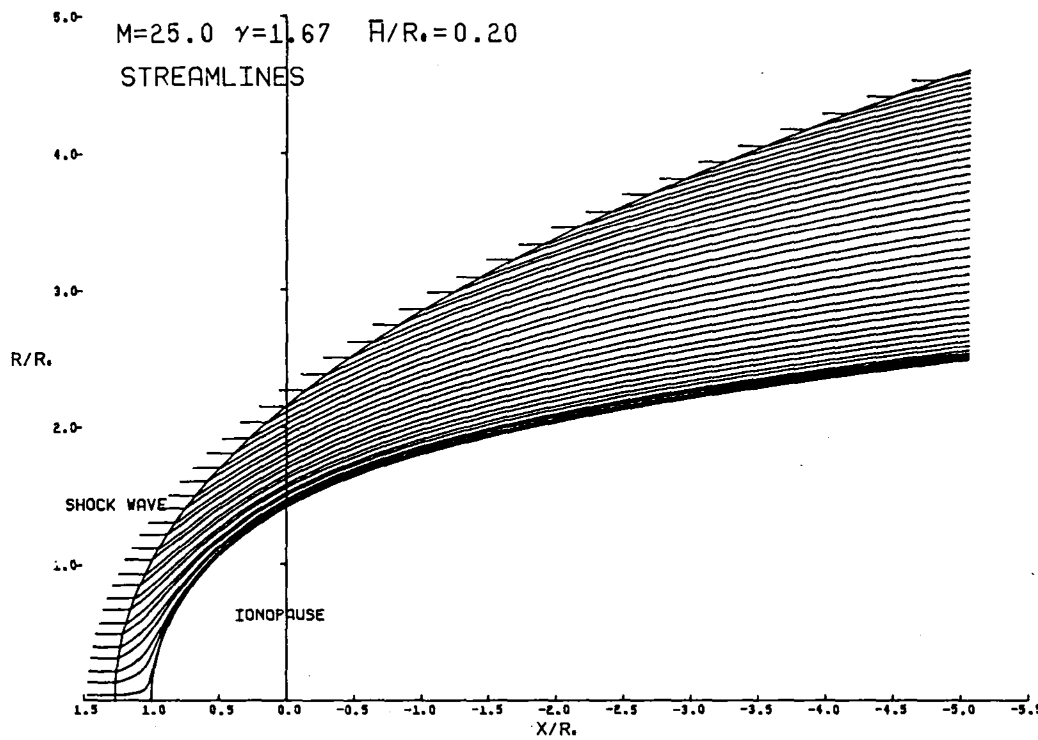

M=12.0 γ=1.67 Ā/R₀=0.20
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.94
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10

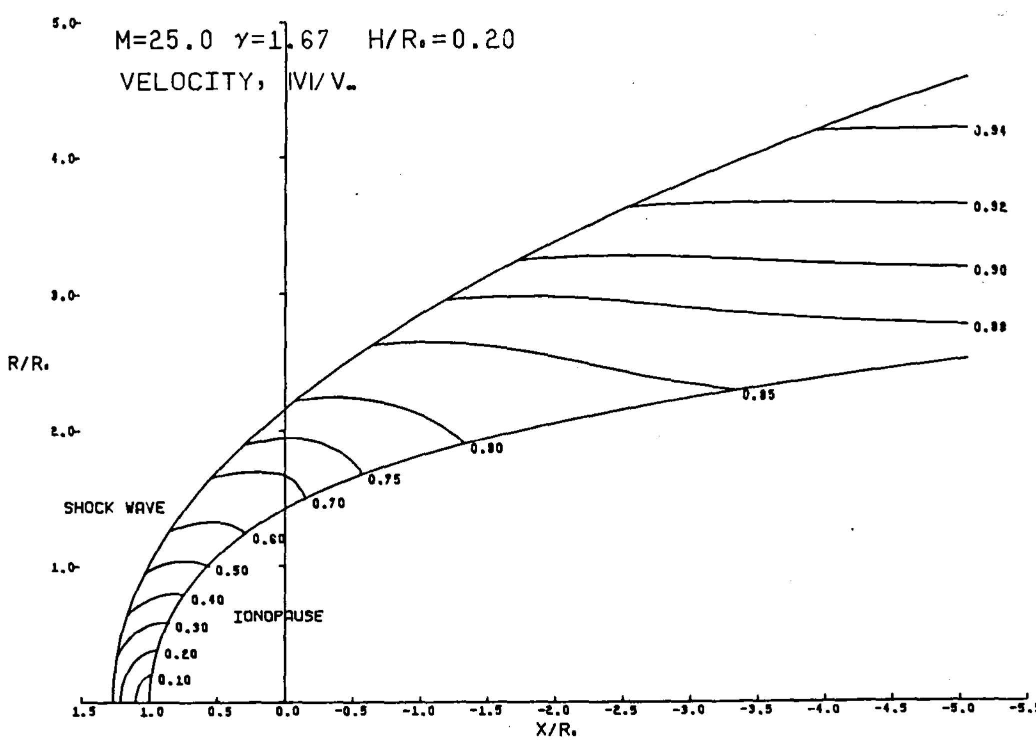

M=12.0 γ=1.67 H̄/R₀=0.20
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80
4.00
4.20

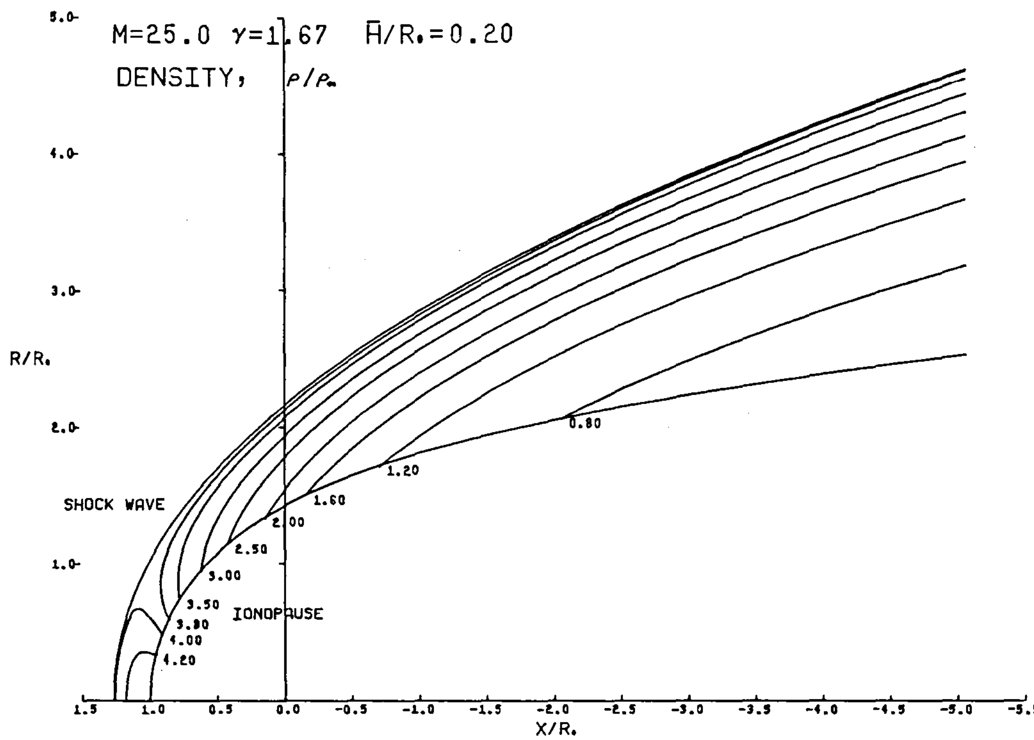

M=12.0 γ=1.67 H̄/R₀=0.20
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
6.59
8.37
10.12
11.83
14.32
18.28
22.00
25.48
31.72
37.03
41.32
44.69
47.09
49.52

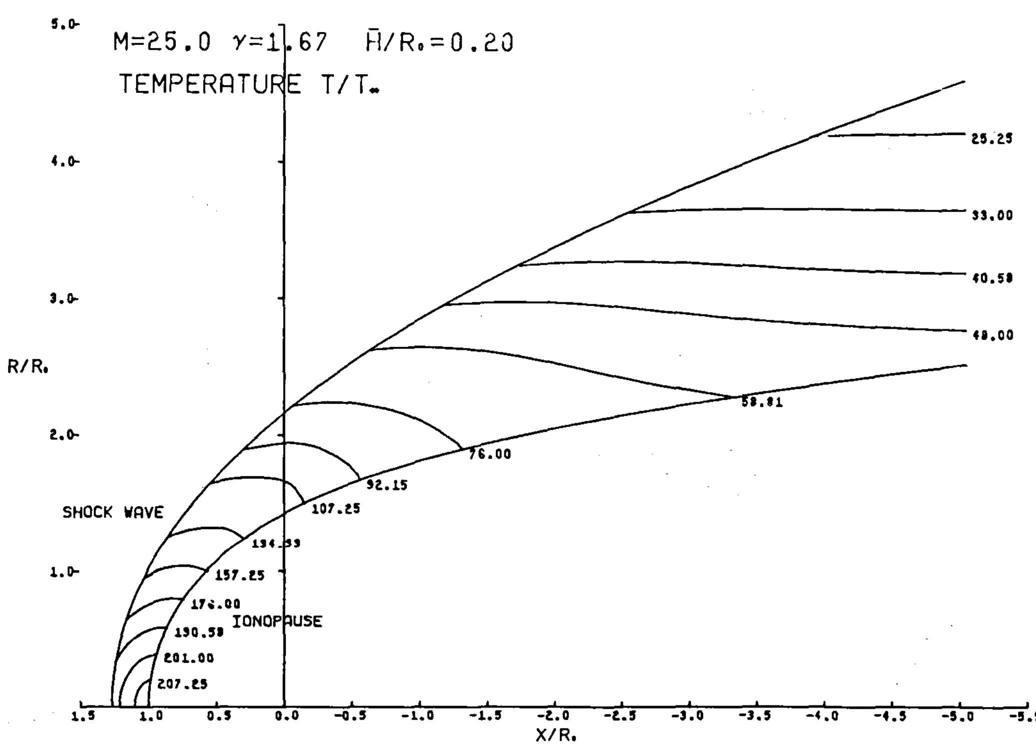

M=12.0 γ=1.67 H̄/R₀=0.20
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
3.00
2.50
2.00
1.50
1.25
1.00
0.80
0.60
0.45

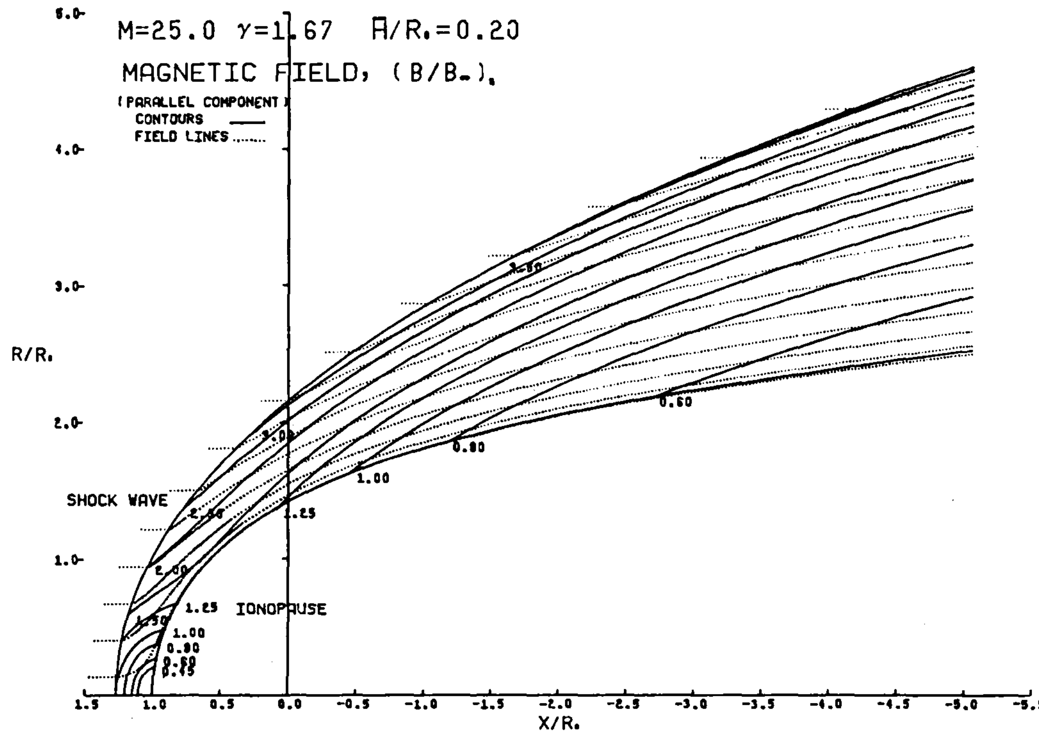

M=12.0 γ=1.67 H̄/R₀=0.20
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
1.25
1.50
2.00
2.50
3.00
6.00

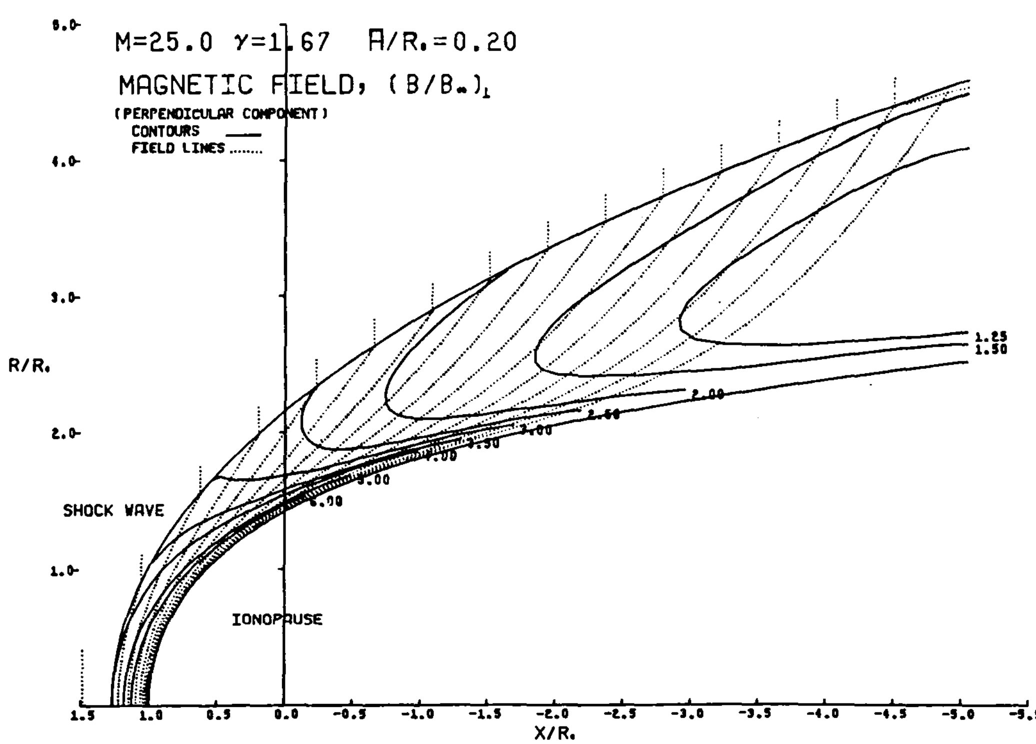

M=25.0 γ=1.67 H/R₀=0.20
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

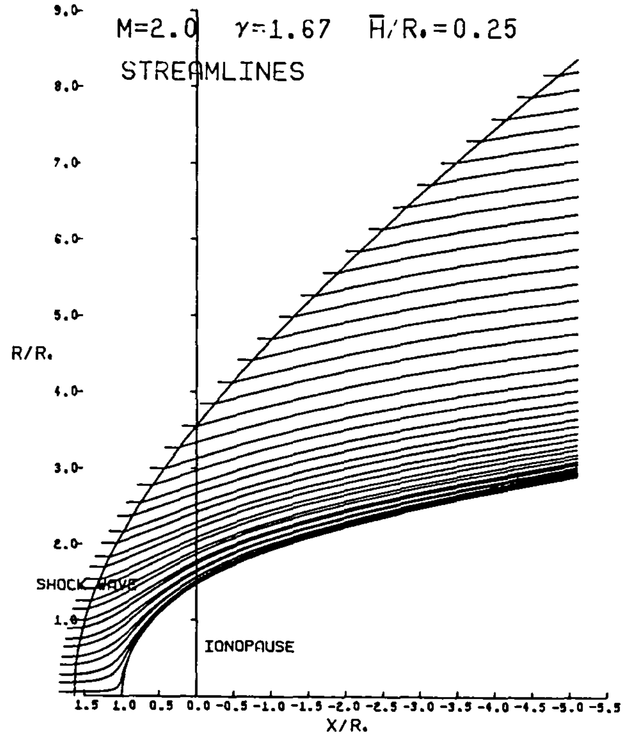

M=25.0 γ=1.67 H/R₀=0.20
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.94
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10

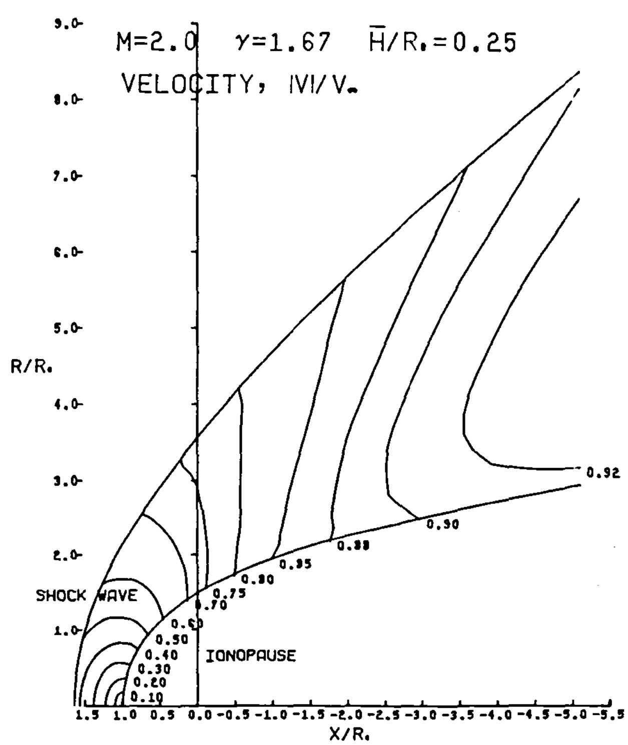

M=25.0 γ=1.67 Ā/R₀=0.20
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.90
4.00
4.20

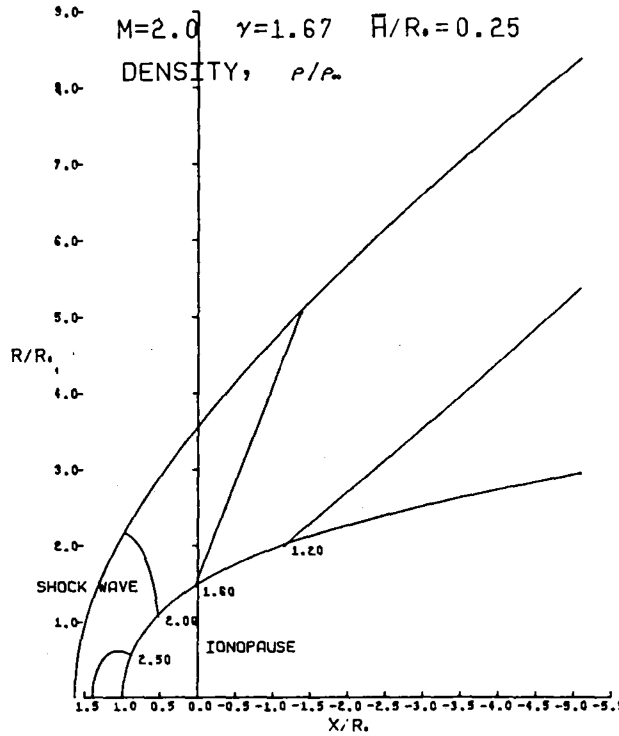

M=25.0 γ=1.67 Ā/R₀=0.20
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
25.25
33.00
40.58
49.00
58.81
76.00
92.15
107.25
134.33
157.25
176.00
190.58
201.00
207.25

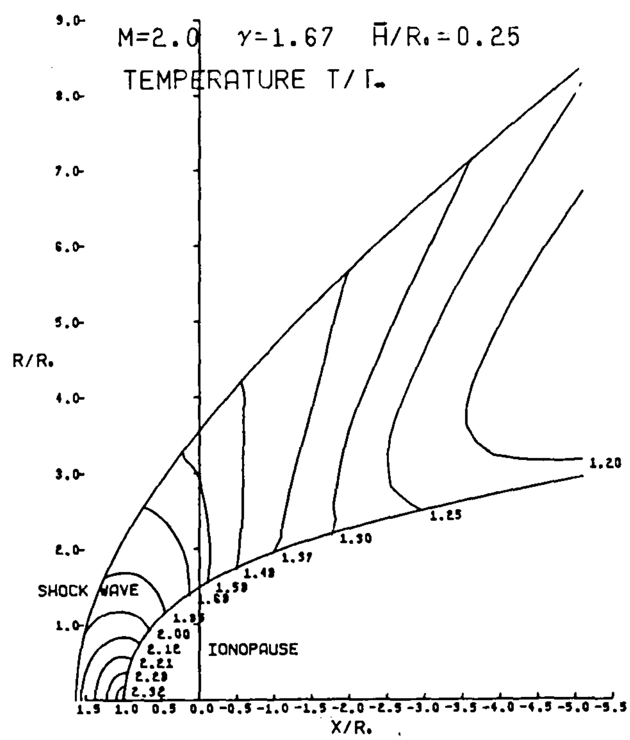

M=25.0 γ=1.67 A/R₀=0.20
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.00
0.80
0.60
1.25

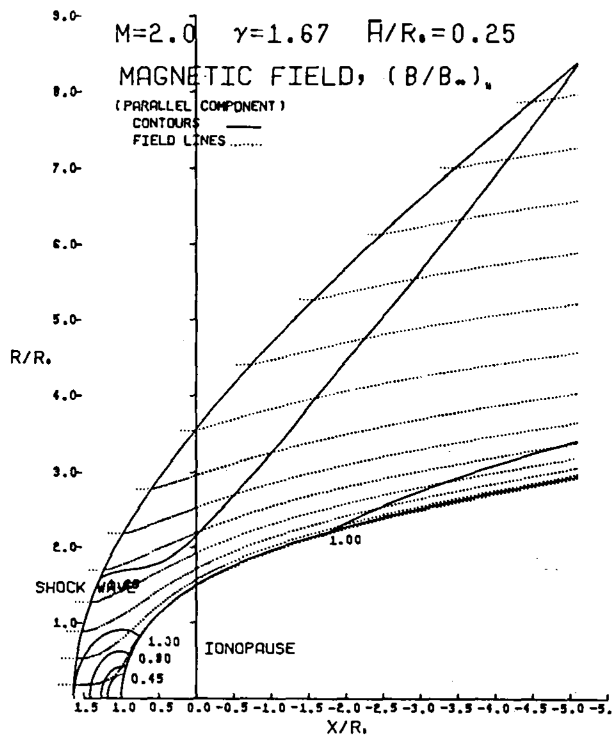

M=25.0 γ=1.67 A/R₀=0.20
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.25
1.50
2.00

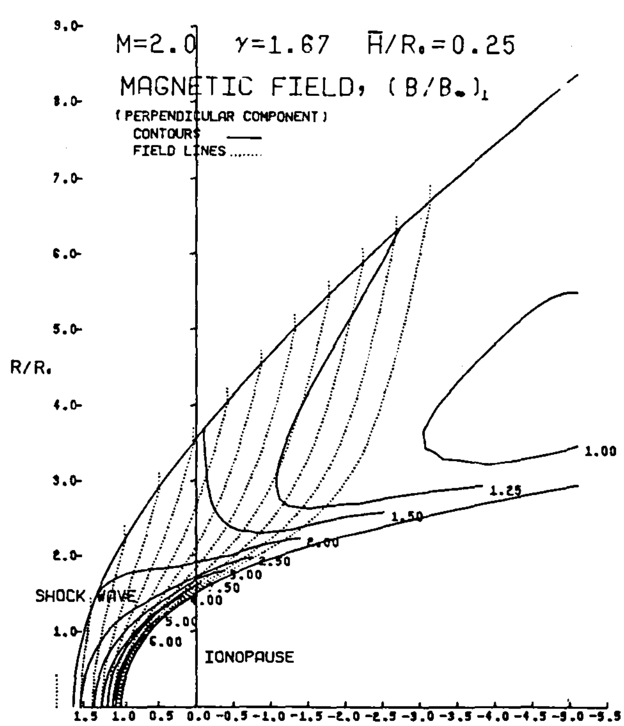

M=2.0 γ=1.67 H̄/R₀=0.25
STREAMLINES
R/R₀
SHOCK WAVE
IONOPAUSE
X/R₀

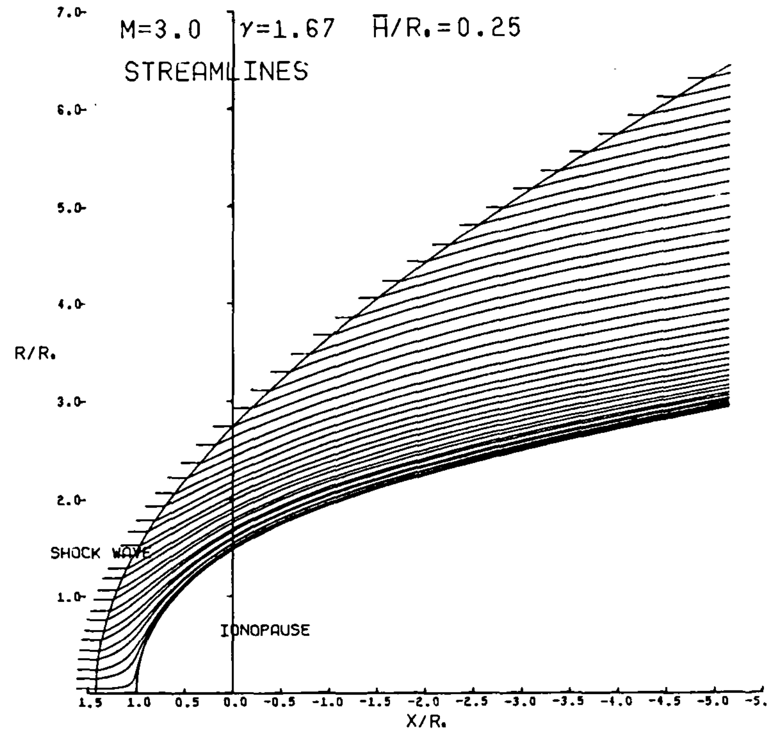

M=2.0 γ=1.67 H̄/R₀=0.25
VELOCITY, |V|/V∞
R/R₀
SHOCK WAVE
IONOPAUSE
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10
X/R₀

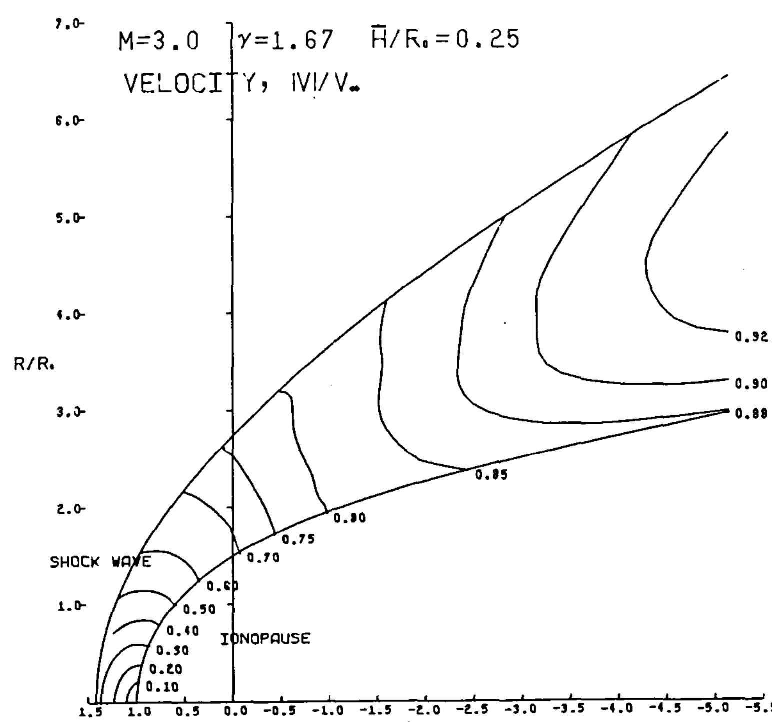

M=2.0 γ=1.67 H̄/R₀=0.25
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.20
1.60
2.00
2.50

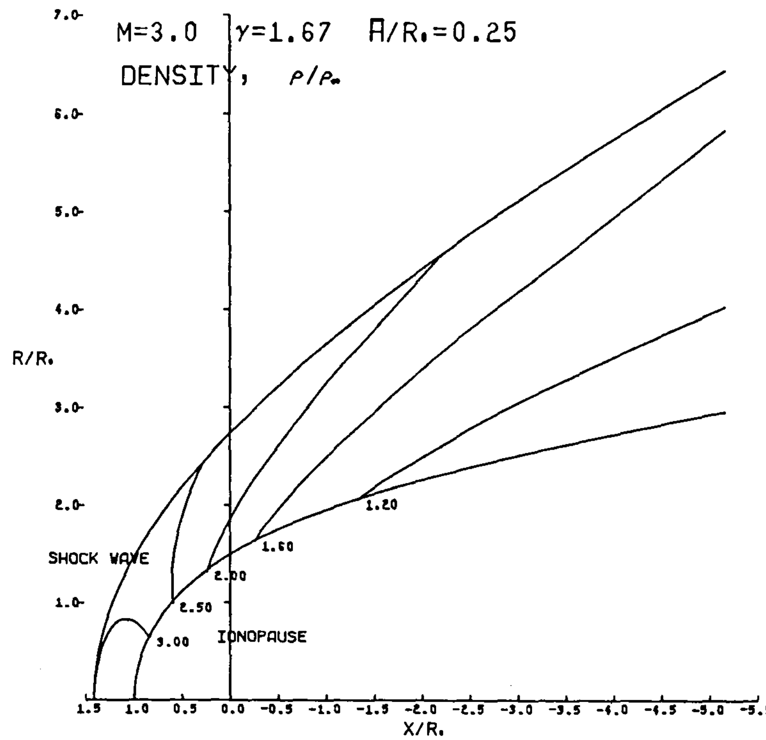

M=2.0 γ=1.67 H̄/R₀=0.25
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.20
1.25
1.30
1.37
1.43
1.50
1.60
2.00

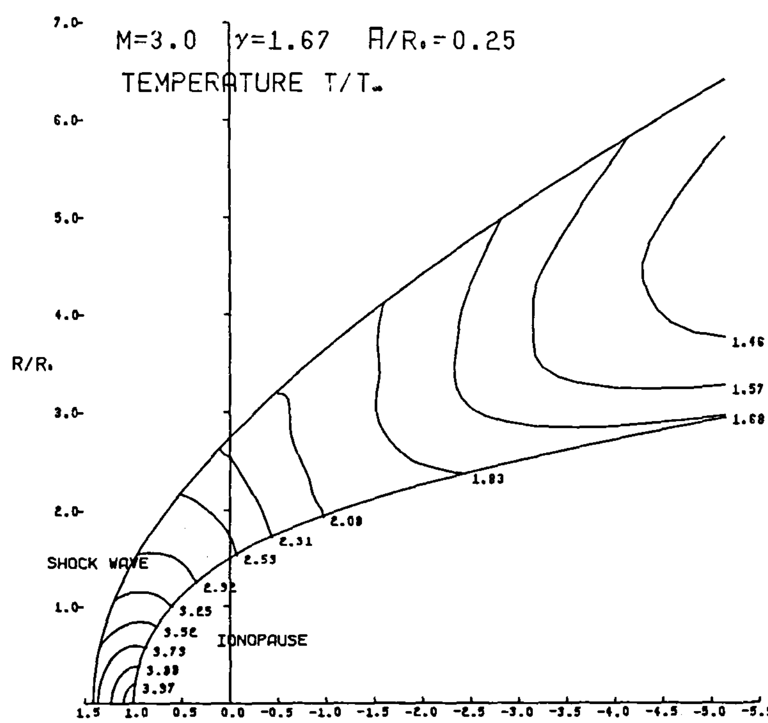

M=2.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
IONOPAUSE
1.00
0.80
0.45
X/R₀

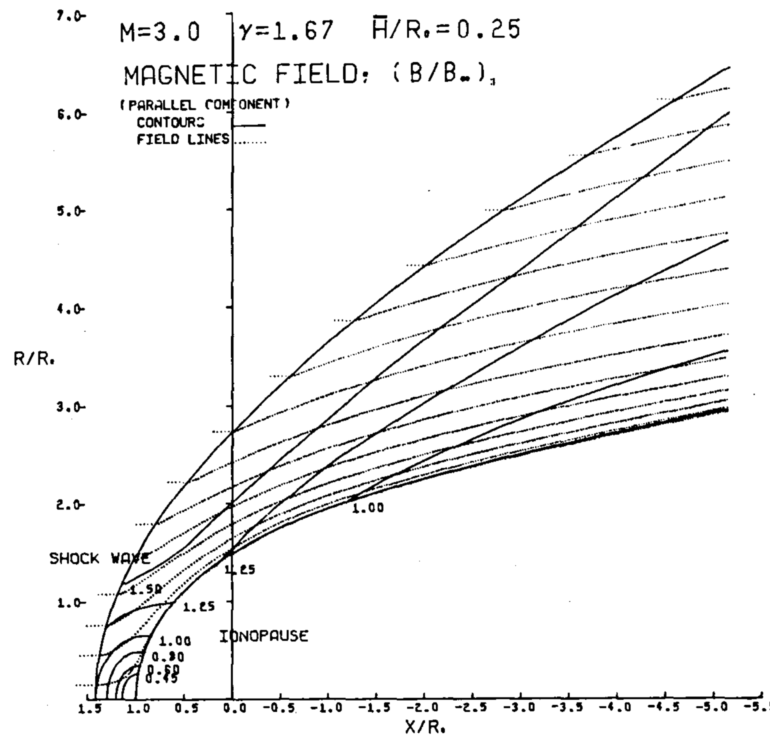

M=2.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
SHOCK WAVE
IONOPAUSE
1.00
1.25
1.50
2.00
2.50
3.00
5.00
6.00
X/R₀


M=3.0 γ=1.67 H̄/R₀=0.25
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
7.0
6.0
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5

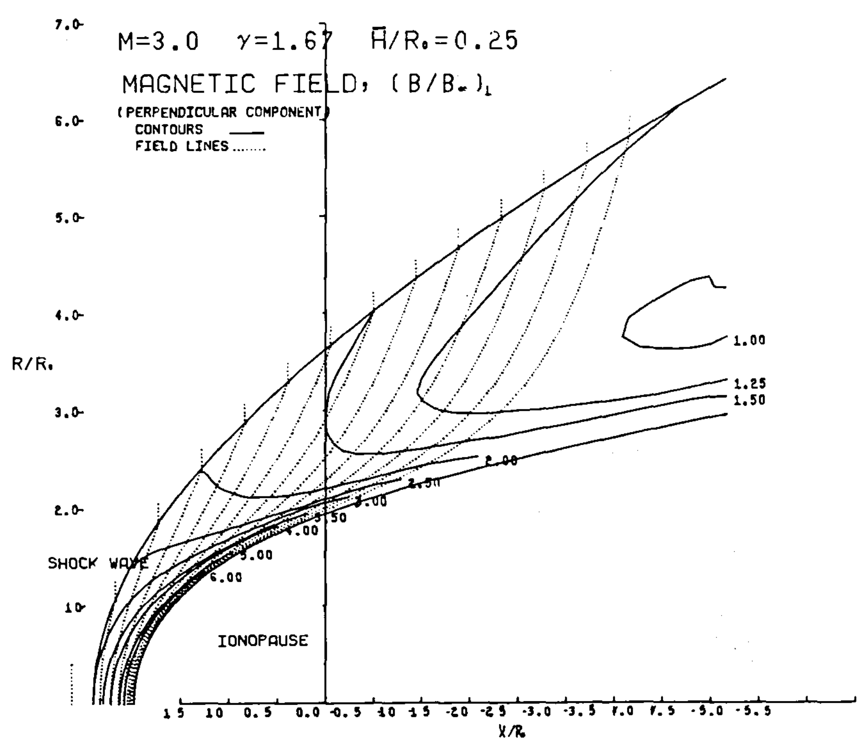

M=3.0 γ=1.67 H̄/R₀=0.25
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10
R/R₀
X/R₀
7.0
6.0
5.0
4.0
3.0
2.0
1.0
1.5 1.0 0.5 0.0 -0.5 -1.0 -1.5 -2.0 -2.5 -3.0 -3.5 -4.0 -4.5 -5.0 -5.5


M=3.0 γ=1.67 A/R₀=0.25
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
3.00
2.50
2.00
1.60
1.20

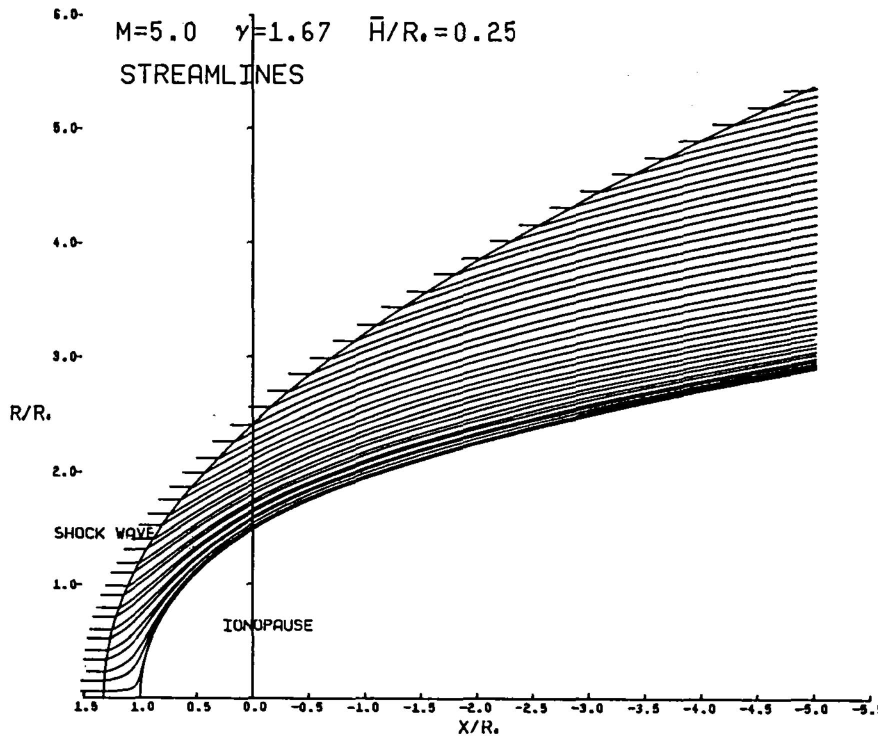

M=3.0 γ=1.67 A/R₀=0.25
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.46
1.57
1.68
1.83
2.09
2.31
2.53
2.92
3.25
3.52
3.73
3.88
3.97

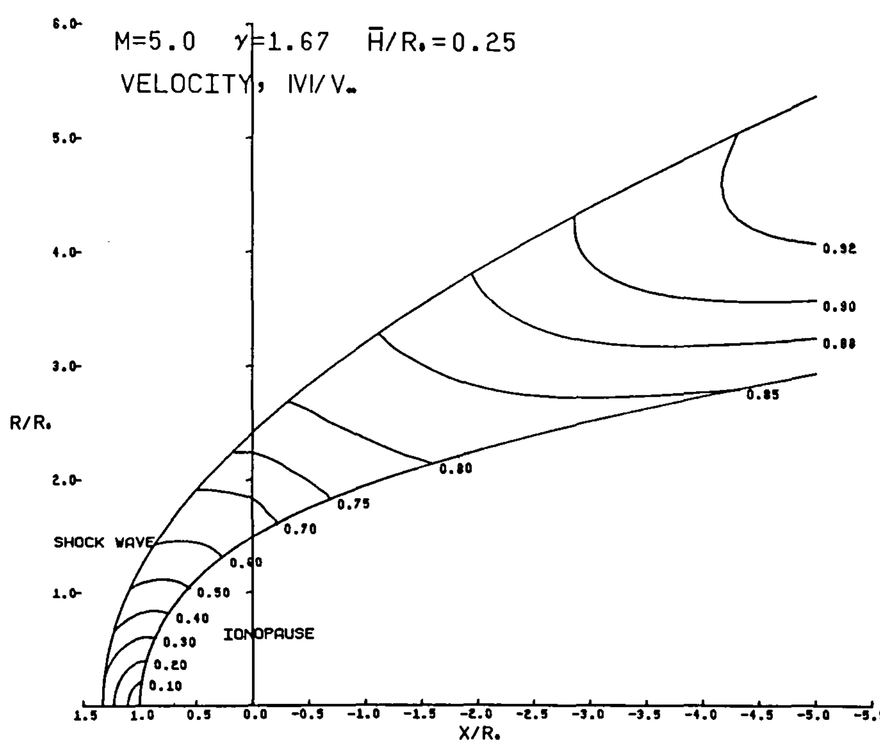


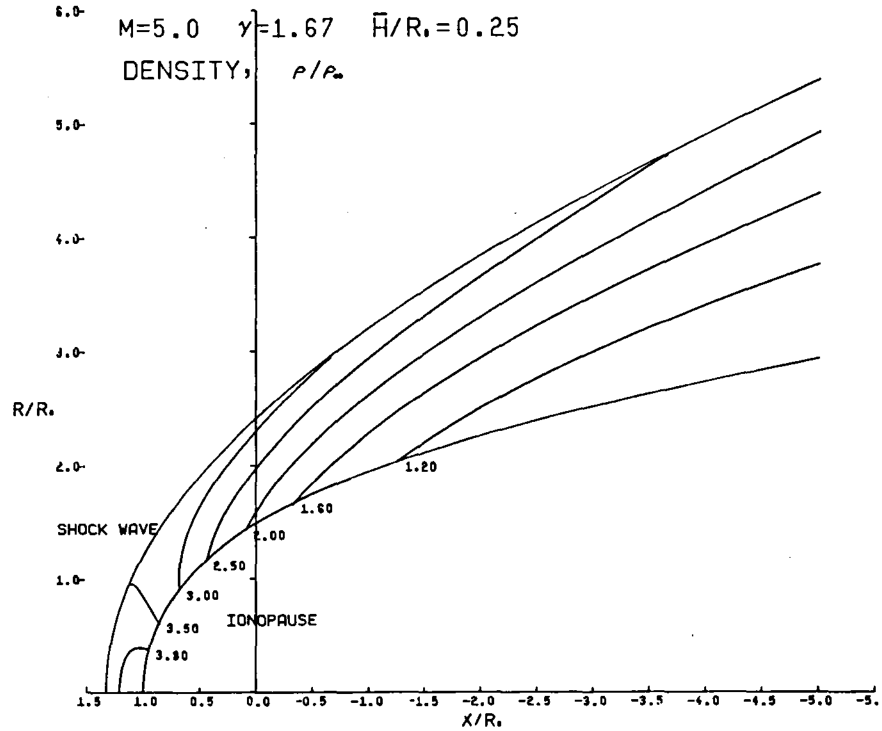
M=3.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD: (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.50
1.25
1.00



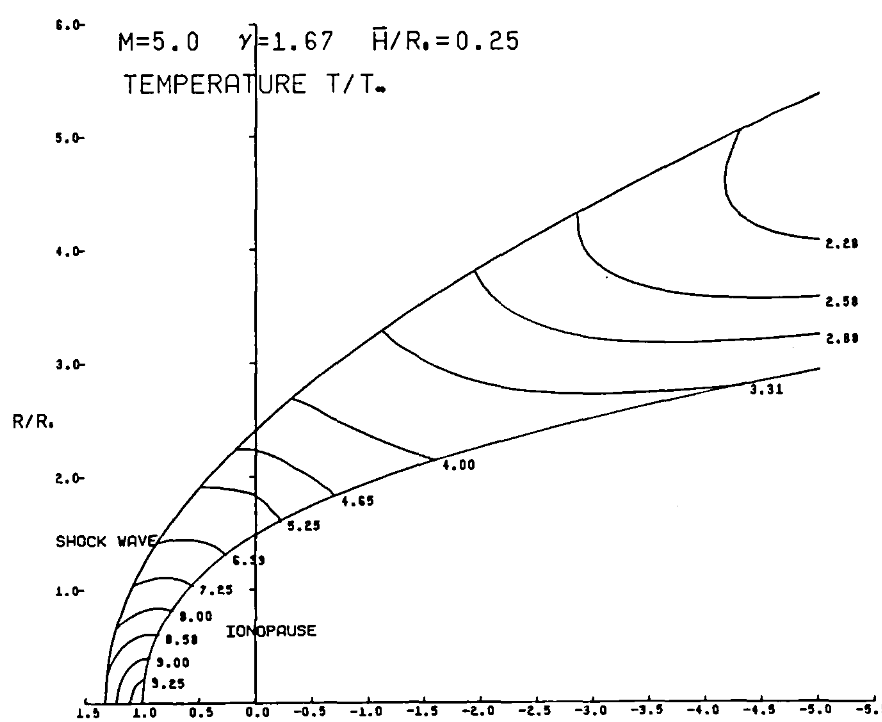
M=3.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD: (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.00
1.25
1.50
2.00
2.50
3.00
3.50
4.00
5.00
6.00


M=5.0 γ=1.67 H̄/R₀=0.25
STREAMLINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE

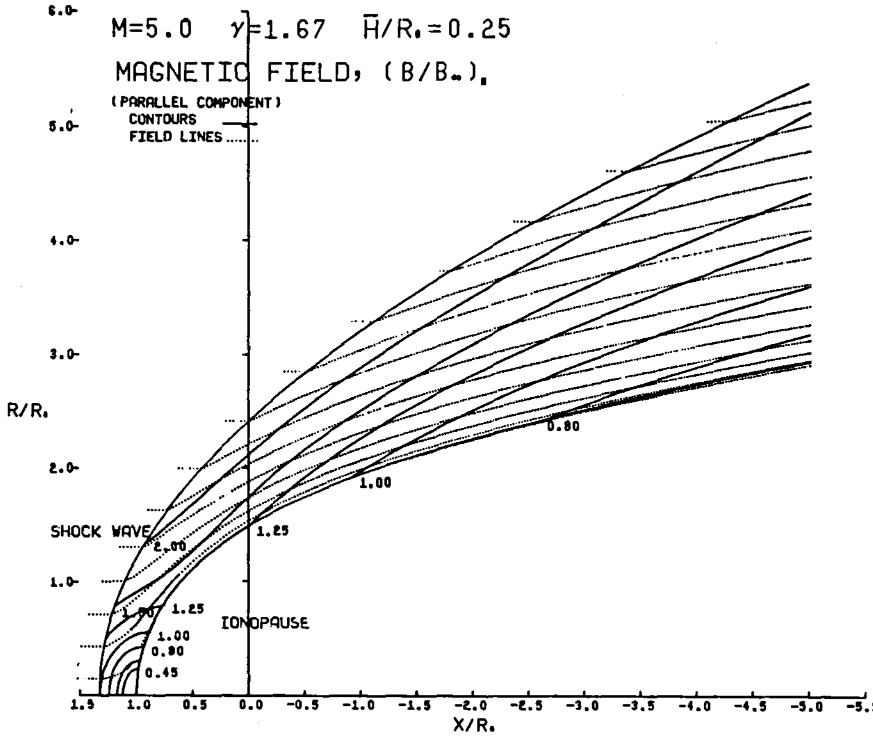

M=5.0 γ=1.67 H̄/R₀=0.25
VELOCITY, |V|/V∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10

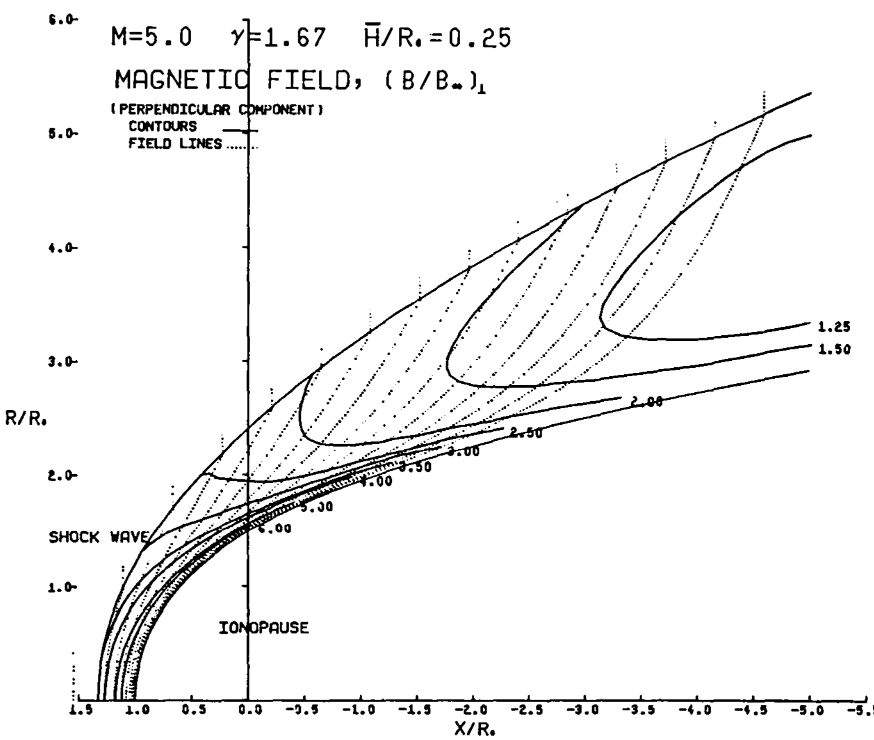


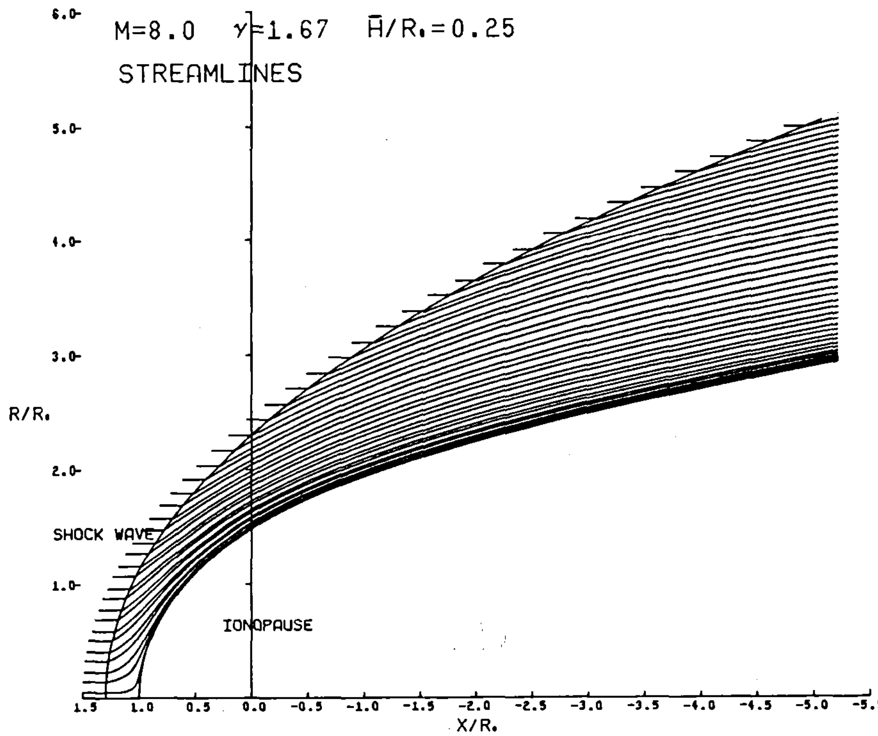
M=5.0 γ=1.67 $\bar{H}/R_0=0.25$
DENSITY, ρ/ρ_∞
SHOCK WAVE
IONOPAUSE
1.20
1.60
2.00
2.50
3.00
3.50
3.80
R/R₀
X/R₀

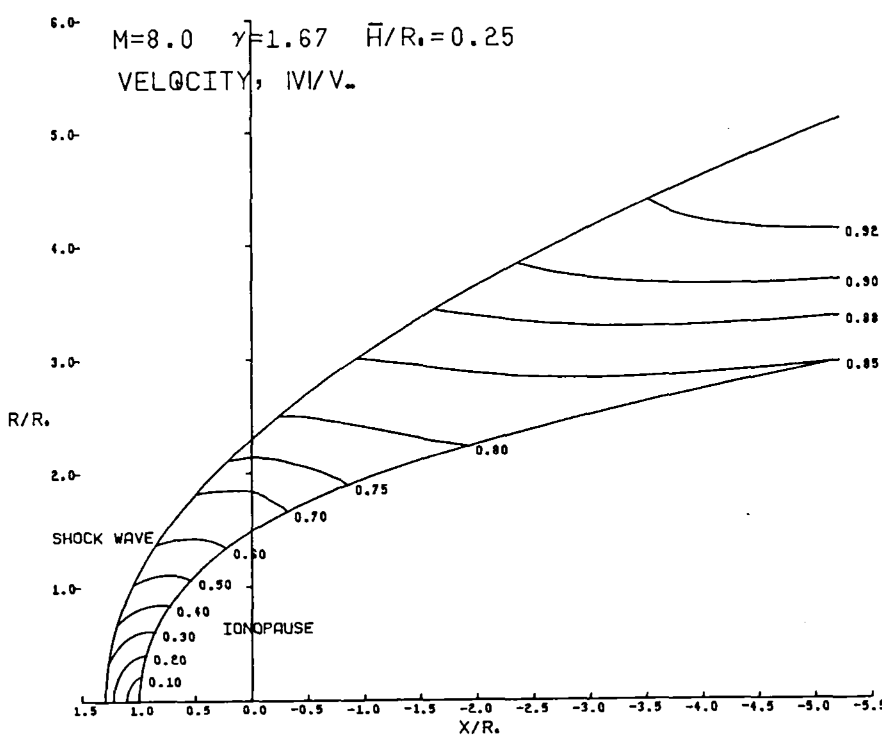


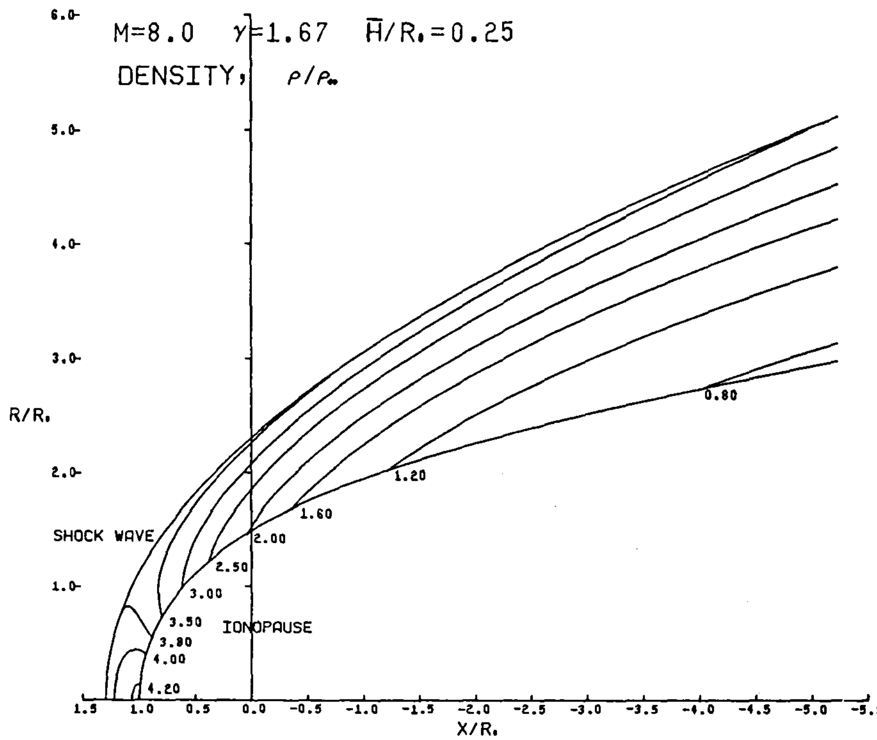
M=5.0 γ=1.67 $\bar{H}/R_0=0.25$
TEMPERATURE T/T_∞
SHOCK WAVE
IONOPAUSE
2.50
2.80
3.31
4.00
4.65
5.25
7.25
8.00
8.50
9.00
9.25
R/R₀
X/R₀


M=5.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
2.00
1.25
1.00
0.80
0.45

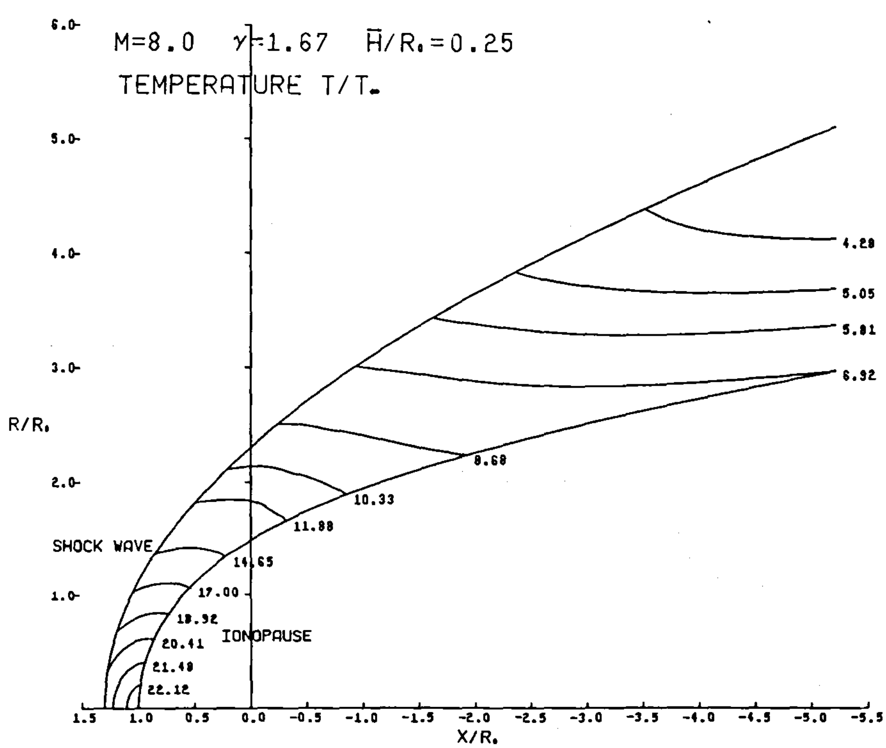

M=5.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.25
1.50
2.00
2.50
3.00
3.50
4.00
5.30
6.00


M=8.0 γ=1.67 H̄/R₀=0.25
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

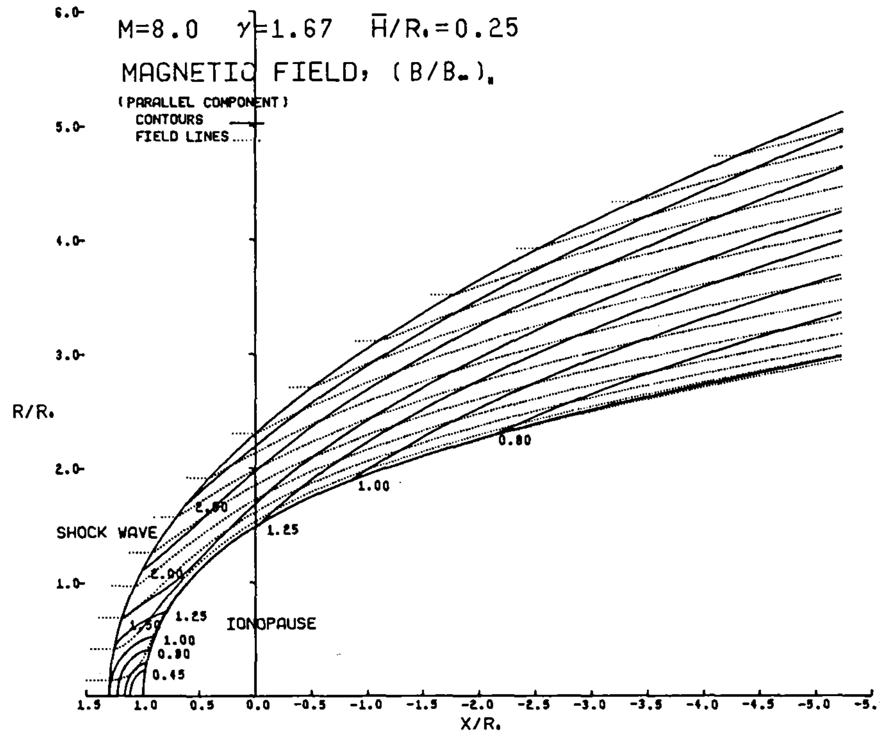

M=8.0 γ=1.67 H̄/R₀=0.25
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10
R/R₀
X/R₀



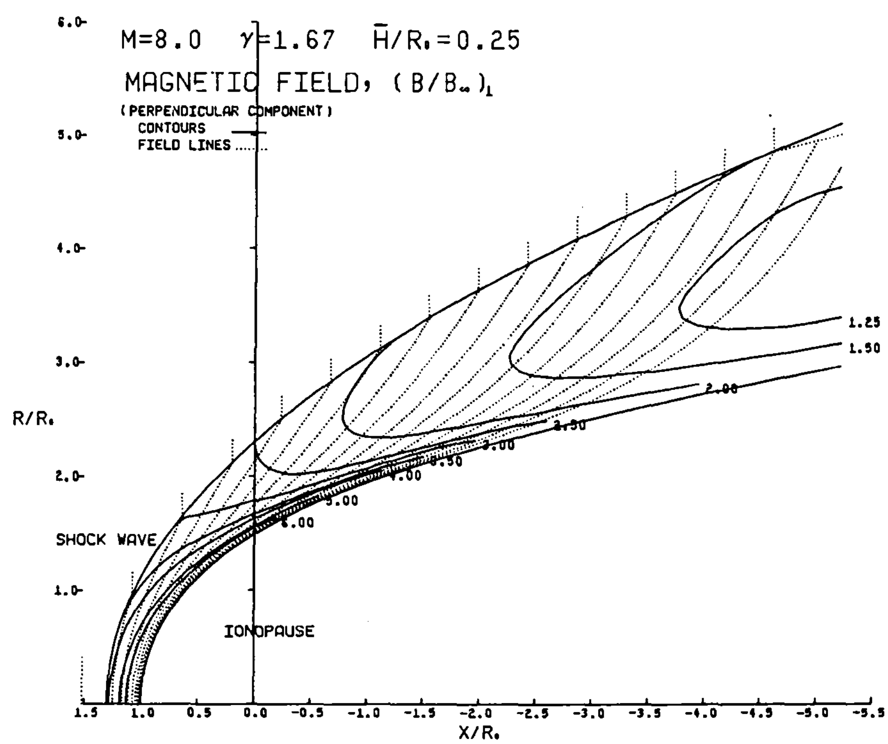
M=8.0 γ=1.67 H̄/R₀=0.25
DENSITY, ρ/ρ∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80
4.00
4.20



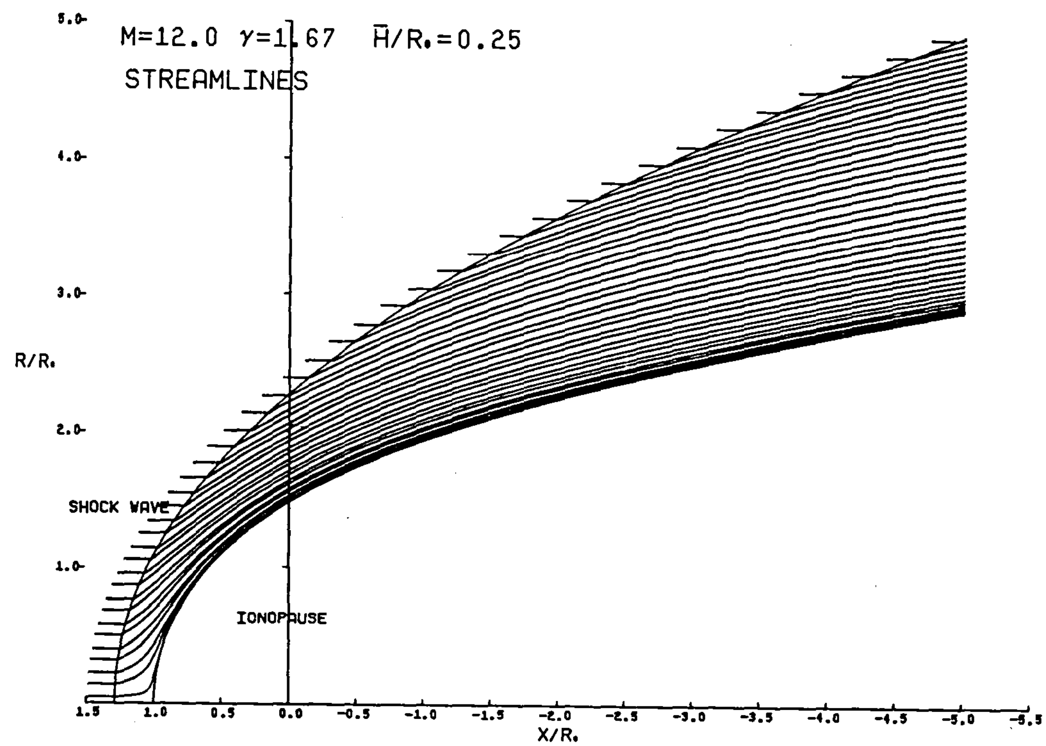
M=8.0 γ=1.67 H̄/R₀=0.25
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
4.28
5.05
5.81
6.92
8.68
10.33
11.88
14.65
17.00
18.92
20.41
21.49
22.12

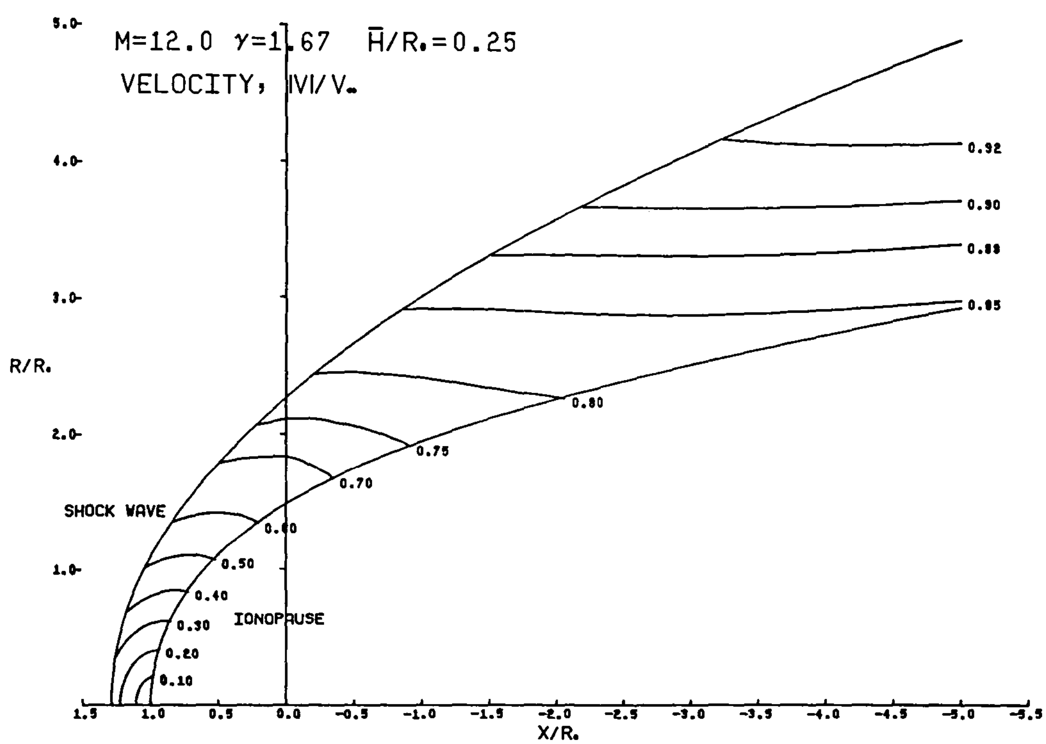


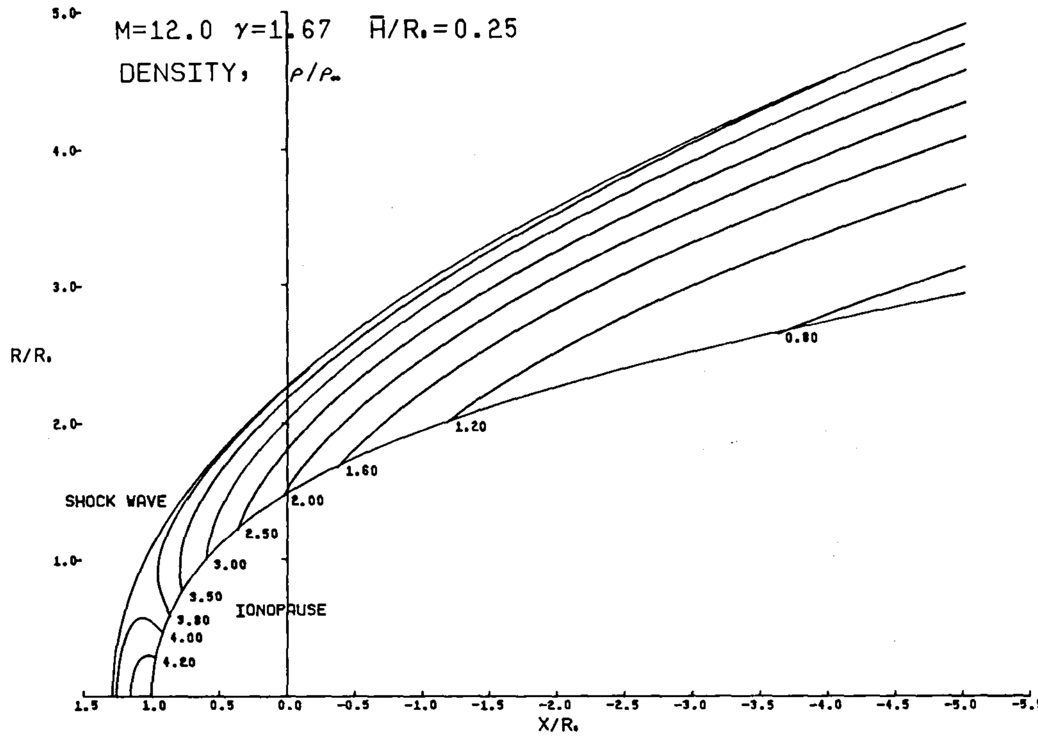
M=8.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
2.50
2.00
1.50
1.25
1.00
0.80
0.45

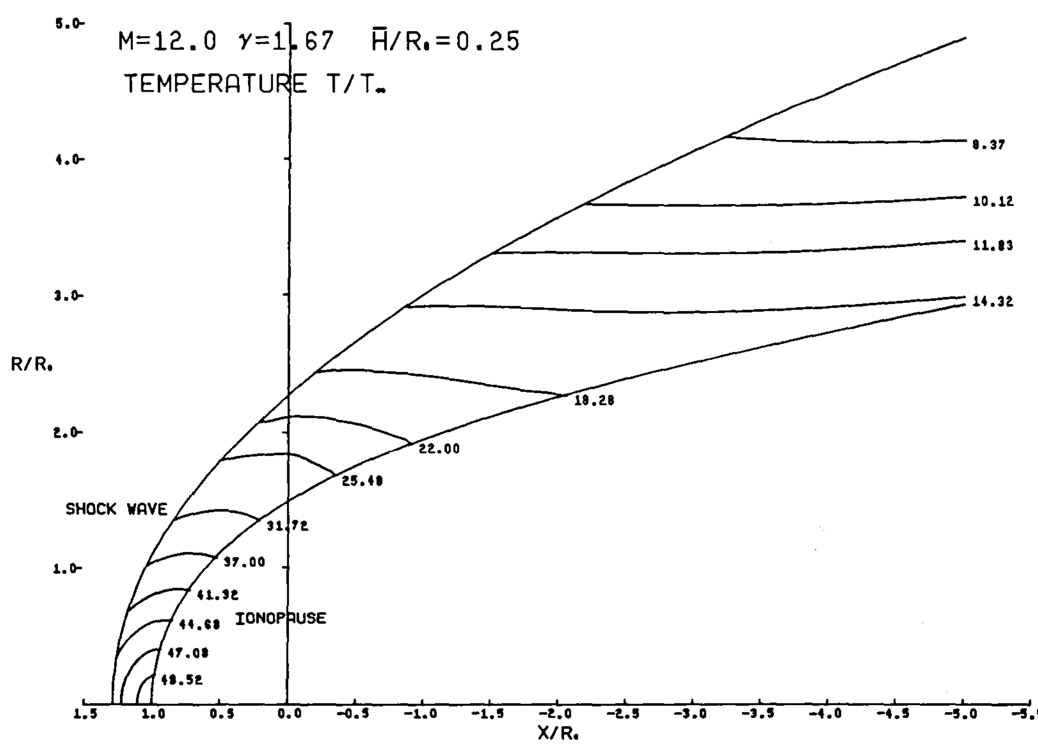


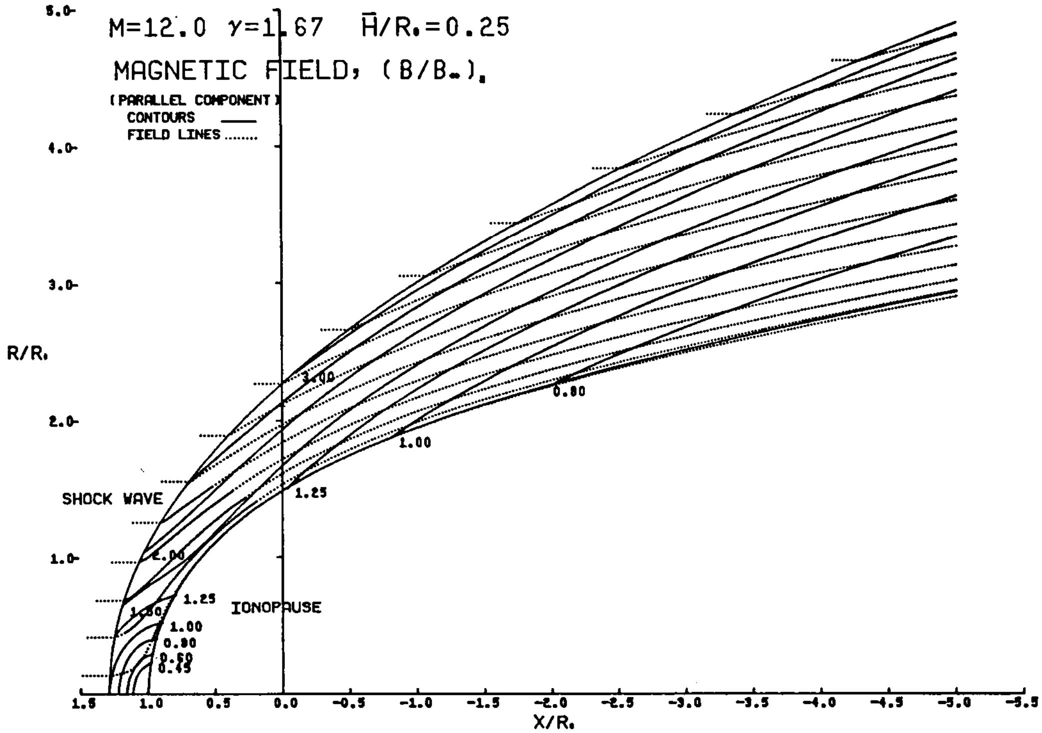
M=8.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.25
1.50
2.00
2.50
3.00
3.50
4.00
5.00
6.00

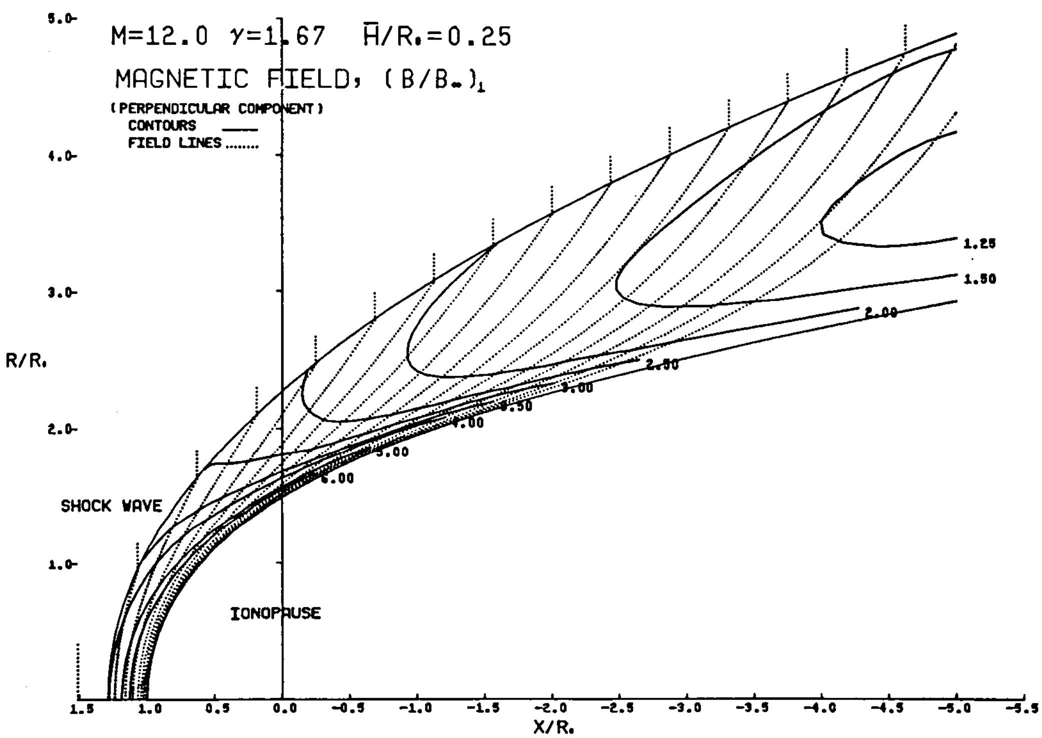

M=12.0 γ=1.67 H̄/R₀=0.25
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

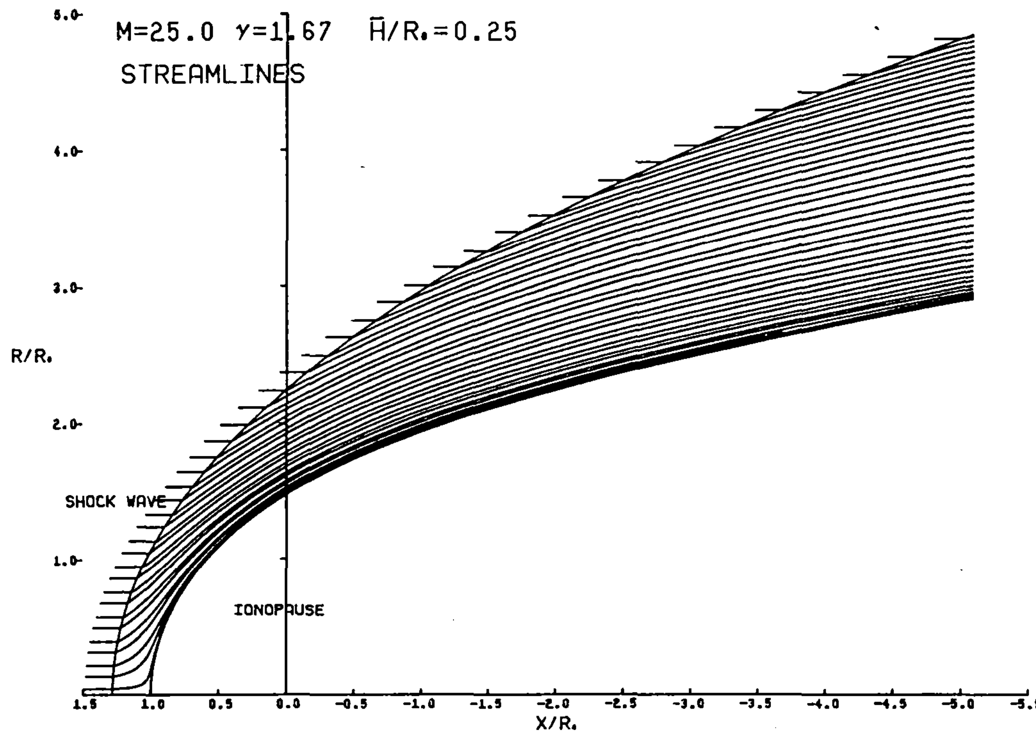

M=12.0 γ=1.67 H̄/R₀=0.25
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10

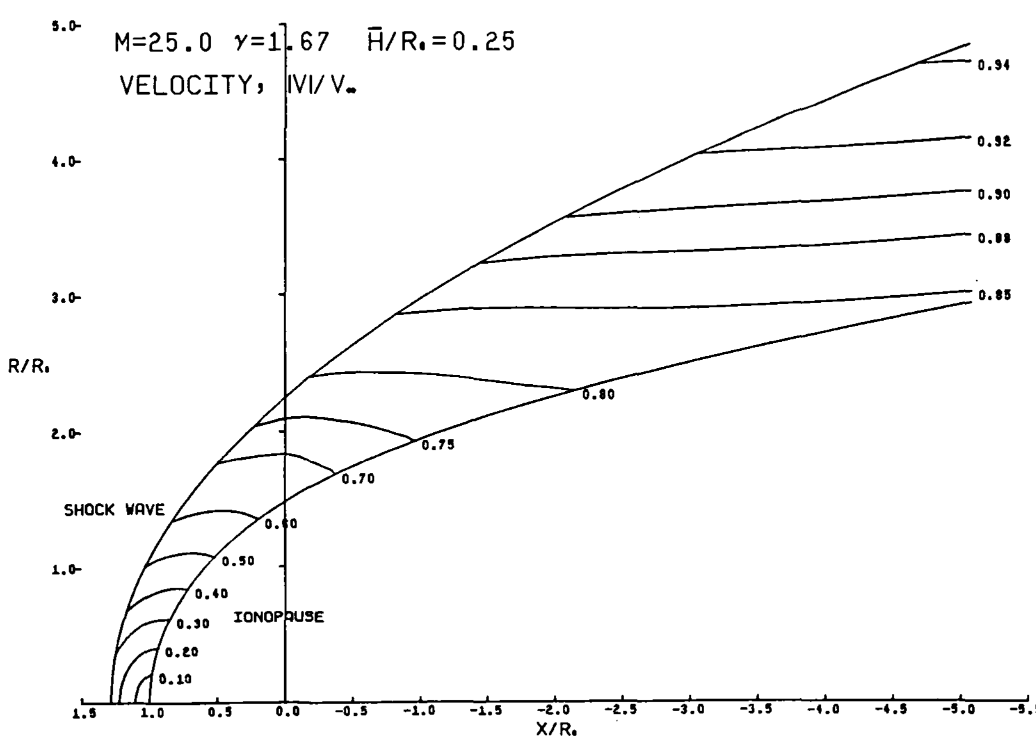

M=12.0 γ=1.67 H̄/R₀=0.25
DENSITY; ρ/ρ∞
SHOCK WAVE
IONOPAUSE
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.80
4.00
4.20
R/R₀
X/R₀

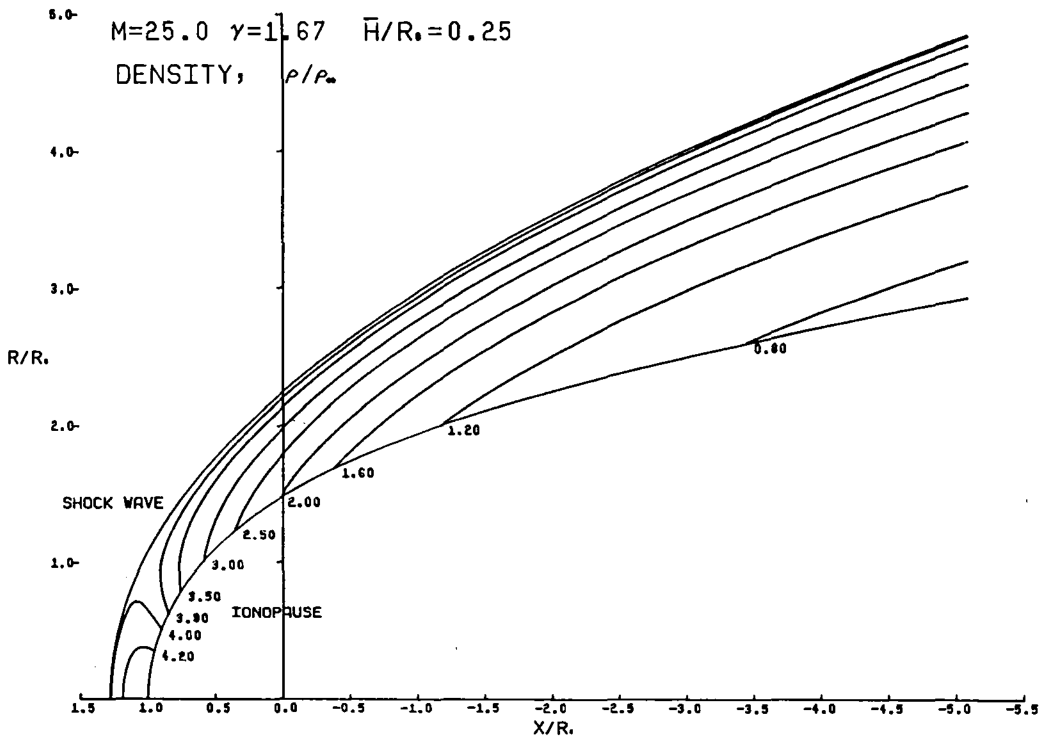

M=12.0 γ=1.67 H̄/R₀=0.25
TEMPERATURE T/T∞
SHOCK WAVE
IONOPAUSE
8.37
10.12
11.83
14.32
18.28
22.00
25.49
31.72
37.00
41.32
44.68
47.08
49.52
R/R₀
X/R₀


M=12.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
3.00
0.80
1.00
1.25
2.00
1.50
0.60
0.45

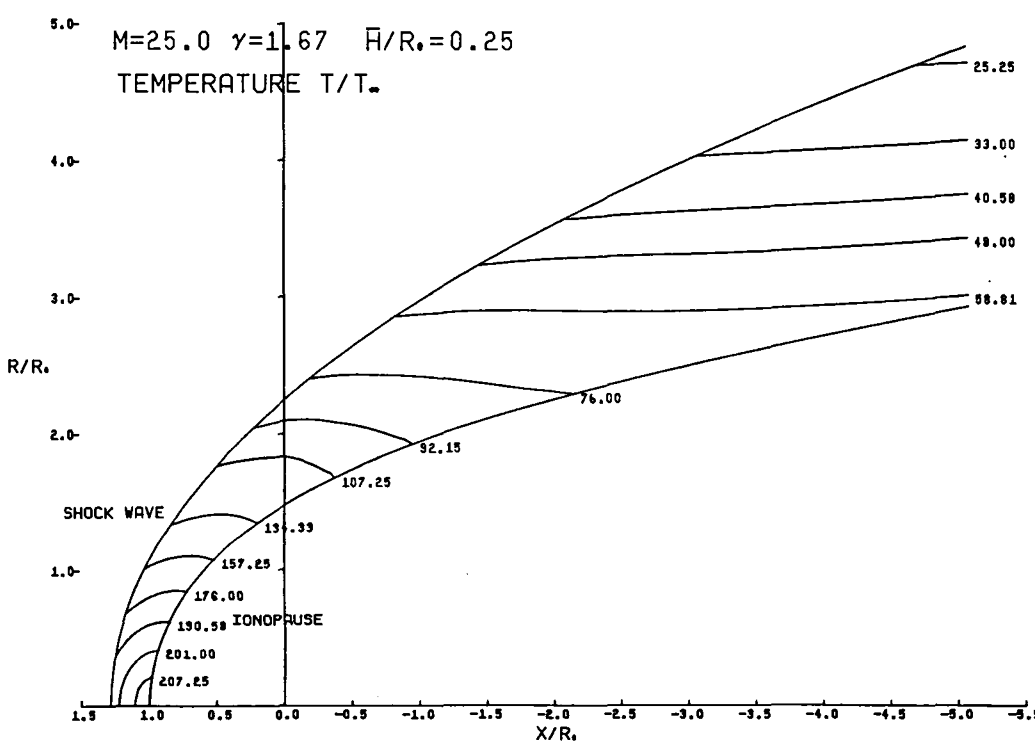

M=12.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
1.25
1.50
2.00
2.50
3.00
3.50
4.00
5.00
6.00


M=25.0 γ=1.67 H̄/R₀=0.25
STREAMLINES
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀

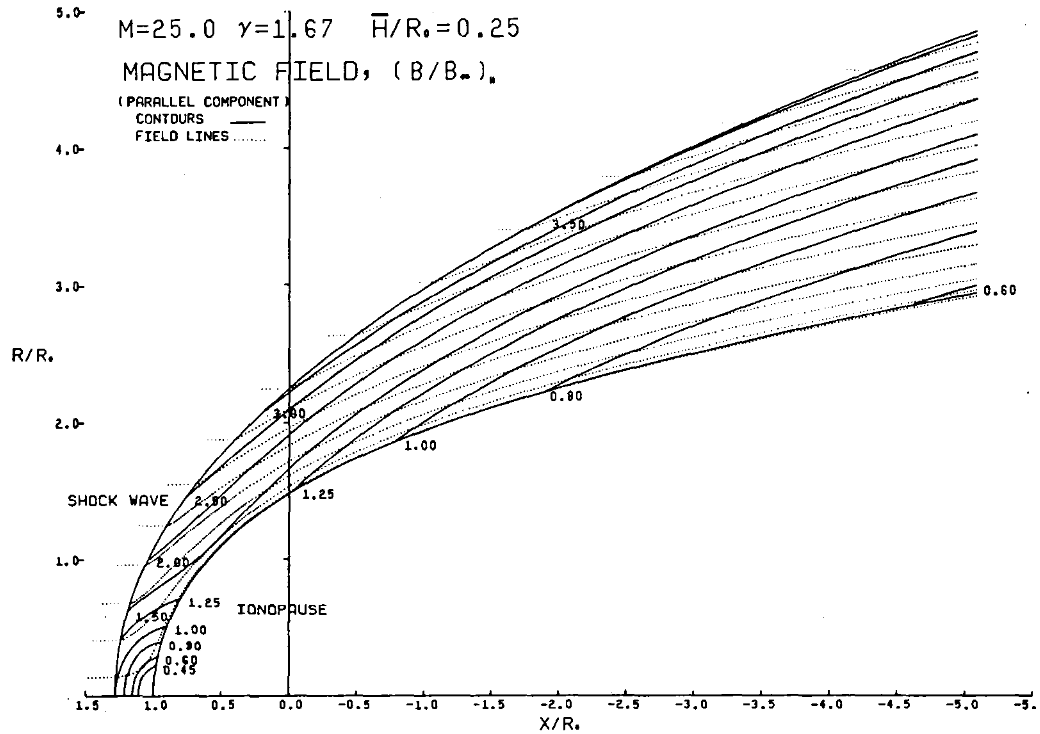

M=25.0 γ=1.67 H̄/R₀=0.25
VELOCITY, |V|/V∞
SHOCK WAVE
IONOPAUSE
R/R₀
X/R₀
0.94
0.92
0.90
0.88
0.85
0.80
0.75
0.70
0.60
0.50
0.40
0.30
0.20
0.10

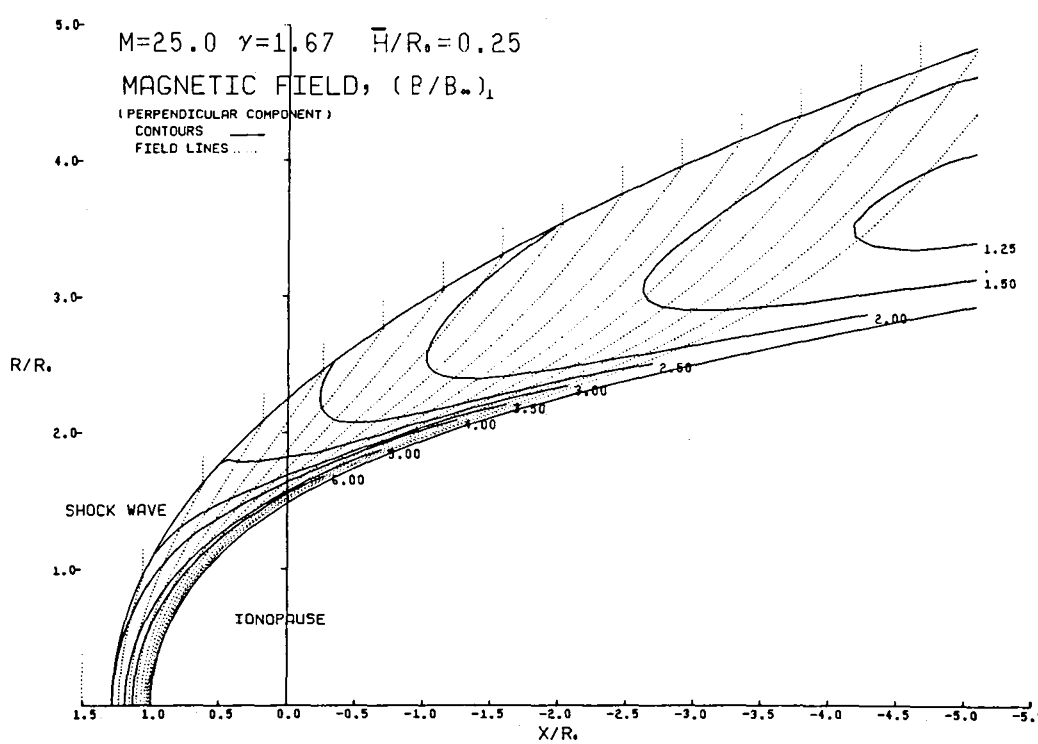


M=25.0 γ=1.67 H̄/R₀=0.25
DENSITY, ρ/ρ∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
0.80
1.20
1.60
2.00
2.50
3.00
3.50
3.90
4.00
4.20



M=25.0 γ=1.67 H̄/R₀=0.25
TEMPERATURE T/T∞
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
25.25
33.00
40.58
48.00
58.81
76.00
92.15
107.25
157.25
176.00
190.58
201.00
207.25


M=25.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)∥
(PARALLEL COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
3.50
3.00
2.50
2.00
1.50
1.25
1.00
0.90
0.60
0.45
0.80


M=25.0 γ=1.67 H̄/R₀=0.25
MAGNETIC FIELD, (B/B∞)⊥
(PERPENDICULAR COMPONENT)
CONTOURS
FIELD LINES
R/R₀
X/R₀
SHOCK WAVE
IONOPAUSE
1.25
1.50
2.00
2.50
3.00
3.50
4.00
5.00
6.00

REFERENCES

1. Spreiter, J. R. and Jones, W. P.: On the Effect of a Weak Interplanetary Magnetic Field on the Interaction Between the Solar Wind and the Geomagnetic Field. J. Geophy. Res., Vol. 68, 1963, pp. 3555-3564.

2. Spreiter, J. R., Alksne, A. Y., and Summers, A. L.: Hydromagnetic Flow Around the Magnetopause. Plan. & Space Sci., Vol. 14, 1966, pp. 223-253.

3. Dryer, M. and Faye-Petersen, R.: Magnetogasdynamic Boundary Condition for a Self-Consistent Solution to the Closed Magnetopause. AIAA Journal, Vol. 4, 1966, pp. 246-254.

4. Dryer, M. and Heckman, G. R.: Application of the Hypersonic Analogy to the Standing Shock of Mars. Solar Phys., Vol. 2, 1967, pp. 112-120.

5. Spreiter, J. R., Alksne, A. Y., and Summers, A. L.: External Aerodynamics of the Magnetosphere. Physics of the Magnetosphere (Ed. R. L. Carovillano, J. F. McClay, and H. R. Radoski), D. Reidel Pub. Co., 1968, pp. 304-378 (also NASA TN 4482, 1968).

6. Spreiter, J. R., Summers, A. L., and Rizzi, A. W.: Solar Wind Flow Past Nonmagnetic Planets - Venus and Mars. Plan. & Space Sci., Vol. 18, 1970, pp. 1281-1289.

7. Spreiter, J. R. and Rizzi, A. W.: Aligned Magnetohydrodynamics Solution for Solar Wind Flow Past the Earth's Magnetosphere. Acta Astronautica, Vol. 1, 1974, pp. 15-35.

8. Spreiter, J. R.: Magnetohydrodynamic and Gasdynamic Aspects of Solar-Wind Flow Around Terrestrial Planets: A Critical Review. NASA SP-397, 1975, pp. 135-150.

9. Stahara, S. S., Chaussee, D. S., Trudinger, B. C., and Spreiter, J. R.: Computational Techniques for Solar Wind Flows Past Terrestrial Planets - Theory and Computer Programs. NASA CR-2924, Nov. 1977.

10. Knudsen, W. C., Spenner, K., Spreiter, J. R., Miller, K. L., and Novak, V.: Thermal Structure and Major Ion Composition of the Venusian Ionosphere: First RPA Results from Venus Orbiter. Science, Vol. 203, No. 4382, Feb. 1979, pp. 757-763.

11. Knudsen, W. C., Spenner, K., Whitter, R. C., Spreiter, J. R., Miller, K. L., and Novak, V.: Thermal Structure and Energy Influx to the Day- and Nightside Venus Ionsphere. Science, Vol. 205, No. 4401, July 1979, pp. 105-107.

12. Kuhn, G. D., Goodwin, F. K., and Perkins, S. C., Jr.: User's Manual for Space-Shuttle Computer Programs. NEAR TR 110, Apr. 1976.

REFERENCES (Concluded)

13. Beam, R. M. and Warming, R. F.: An Implicit Finite-Difference Algorithm for Hyperbolic Systems in Conservation-Law Form. J. Comp. Phys., Vol. 22, No. 1, Sept. 1976.

14. Kentzer, C. P.: Discretization of Boundary Conditions on Moving Discontinuities. Proceedings of the Second International Conference on Numerical Methods in Fluid Dynamics, Lecture Notes in Physics, Vol. 8, M. Holt, ed., Berkeley, CA, 1970, pp. 108-113.

15. Thomas, P. D., Vinokur, M., Bastionon, R., and Conti, R. J.: Numerical Solution for the Three-Dimensional Hypersonic Flow Field of a Blunt Delta Body. AIAA J., Vol. 10, July 1972, pp. 887-894.

16. Kutler, P., Reinhardt, W. A., and Warming, R. F.: Numerical Computations of Multi-Shocked Three-Dimensional Supersonic Flow Fields with Real Gas Effects. AIAA Paper No. 72-702, June 1972.

17. Kutler, P., Reinhardt, W. A., and Warming, R. F.: Multi-Shocked, Three-Dimensional Supersonic Flow Fields with Real Gas Effects. AIAA J., Vol. 11, May 1973, pp. 657-664.

18. Chaussee, D. S., Holtz, T., and Kutler, P.: Inviscid Supersonic/ Hypersonic Body Flow Fields and Aerodynamics from Shock-Capturing Technique Calculations. AIAA Paper No. 75-837, June 1975.

19. MacCormak, R. W.: The Effect of Viscosity in Hypervelocity Impact Cratering. AIAA Paper No. 69-354, 1969.

20. Alksne, A. Y. and Webster, D. L.: Magnetic and Electric Fields in the Magnetosheath. Plan. & Space Sci., Vol. 18, 1970, pp. 1203-1212.

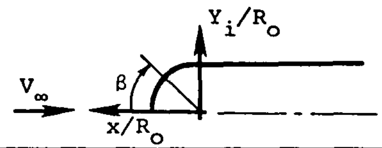
21. Wolfe, J., Intriligator, D. S., Mihalov, J., Collard, H., McKibbin, D., Whitten, R., and Barnes, A.: Initial Observations of the Pioneer Venus Orbiter Solar Wind Plasma Experiment. Science, Vol. 203, No. 4382, Feb. 1979, pp. 750-752.

22. Intriligator, D. S., Collard, H. R., Mihalov, J. P., Whitten, R. C., and Wolfe, J. H.: Electron Observation and Ion Flows from the Pioneer Venus Orbiter Plasma Analyzer Experiment. Science, Vol. 205, No. 4401, July 1979, pp. 116-119.

23. Russell, C. T., Elphic, R. C., and Slavin, J. A.: Initial Pioneer Venus Magnetic Field Results: Dayside Observations. Science, Vol. 203, No. 4382, Feb. 1979, pp. 745-748.

24. Russell, C. T., Elphic, R. C., and Slavin, J. A.: Initial Pioneer Venus Magnetic Field Results: Nightside Observations. Science, Vol. 205, No. 4401, July 1979, pp. 114-116.

Table 1.- Ordinates of Various Ionopause Shapes



β	IONOPAUSE $\bar{H}/R_o = 0.01$		IONOPAUSE $\bar{H}/R_o = 0.05$		IONOPAUSE $\bar{H}/R_o = 0.10$		IONOPAUSE $\bar{H}/R_o = 0.20$		IONOPAUSE $\bar{H}/R_o = 0.25$	
	x/R_o	Y_i/R_o	x/R_o	Y_i/R_o	x/R_o	Y_i/R_o	x/R_o	Y_i/R_o	x/R_o	Y_i/R_o
0°	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000	1.0000	0.0000
2°	0.9994	0.0349	0.9995	0.0349	0.9995	0.0349	0.9995	0.0349	0.9996	0.0349
6°	0.9946	0.1045	0.9950	0.1046	0.9953	0.1046	0.9958	0.1047	0.9960	0.1047
10°	0.9851	0.1737	0.9861	0.1739	0.9870	0.1740	0.9883	0.1743	0.9888	0.1744
14°	0.9709	0.2421	0.9727	0.2425	0.9746	0.2430	0.9771	0.2436	0.9781	0.2439
18°	0.9520	0.3093	0.9550	0.3103	0.9580	0.3113	0.9622	0.3126	0.9638	0.3132
22°	0.9285	0.3751	0.9330	0.3770	0.9374	0.3787	0.9435	0.3812	0.9459	0.3822
26°	0.9006	0.4393	0.9068	0.4423	0.9127	0.4451	0.9211	0.4492	0.9243	0.4508
30°	0.8684	0.5014	0.8764	0.5060	0.8840	0.5104	0.8949	0.5167	0.8991	0.5191
34°	0.8320	0.5612	0.8419	0.5679	0.8514	0.5743	0.8649	0.5834	0.8701	0.5869
38°	0.7916	0.6185	0.8035	0.6278	0.8148	0.6366	0.8312	0.6494	0.8374	0.6543
42°	0.7474	0.6729	0.7613	0.6854	0.7745	0.6973	0.7935	0.7145	0.8009	0.7211
46°	0.6995	0.7243	0.7153	0.7407	0.7303	0.7563	0.7520	0.7787	0.7604	0.7874
50°	0.6482	0.7725	0.6658	0.7934	0.6824	0.8133	0.7066	0.8421	0.7159	0.8532
54°	0.5937	0.8172	0.6128	0.8435	0.6309	0.8683	0.6571	0.9044	0.6673	0.9184
58°	0.5363	0.8582	0.5565	0.8906	0.5756	0.9212	0.6035	0.9657	0.6143	0.9831
62°	0.4761	0.8954	0.4971	0.9349	0.5168	0.9719	0.5456	1.0261	0.5569	1.0473
66°	0.4135	0.9287	0.4346	0.9761	0.4543	1.0203	0.4504	1.1147	0.4947	1.1744
70°	0.3487	0.9581	0.3691	1.0142	0.3882	1.0665	0.4163	1.1437	0.4274	1.1744
74°	0.2820	0.9833	0.3009	1.0492	0.3184	1.1103	0.3444	1.2010	0.3548	1.2374
78°	0.2135	1.0046	0.2298	1.0811	0.2448	1.1517	0.2673	1.2574	0.2764	1.3001
82°	0.1436	1.0219	0.1560	1.1098	0.1674	1.1908	0.1845	1.3130	0.1915	1.3628
86°	0.0724	1.0354	0.0794	1.1355	0.0858	1.2276	0.0956	1.3677	0.0997	1.4254
90°	0.0000	1.0454	0.0000	1.1583	0.0000	1.2620	0.0000	1.4218	0.0000	1.4883
94°	-0.0736	1.0523	-0.0824	1.1782	-0.0905	1.2943	-0.1032	1.4753	-0.1085	1.5516
98°	-0.1485	1.0566	-0.1680	1.1955	-0.1861	1.3244	-0.2148	1.5284	-0.2271	1.6156
102°	-0.2251	1.0591	-0.2572	1.2102	-0.2875	1.3524	-0.3361	1.5813	-0.3572	1.6807
106°	-0.3040	1.0603	-0.3506	1.2226	-0.3953	1.3785	-0.4686	1.6343	-0.5010	1.7472
110°	-0.3861	1.0607	-0.4488	1.2330	-0.5106	1.4027	-0.6142	1.6875	-0.6608	1.8156
114°	-0.4723	1.0609	-0.5527	1.2415	-0.6346	1.4253	-0.7753	1.7414	-0.8400	1.8866

Table 1.- Concluded.

β	IONOPAUSE $\bar{H}/R_o = 0.01$		IONOPAUSE $\bar{H}/R_o = 0.05$		IONOPAUSE $\bar{H}/R_o = 0.10$		IONOPAUSE $\bar{H}/R_o = 0.20$		IONOPAUSE $\bar{H}/R_o = 0.25$	
	x/R_o	Y_i/R_o	x/R_o	Y_i/R_o	x/R_o	Y_i/R_o	x/R_o	Y_i/R_o	x/R_o	Y_i/R_o
118°	-0.5641	1.0610	-0.6638	1.2484	-0.7690	1.4462	-0.9551	1.7963	-1.0427	1.9610
122°	-0.6630	1.0610	-0.7835	1.2539	-0.9159	1.4657	-1.1578	1.8529	-1.2746	2.0397
126°	-0.7708	1.0610	-0.9142	1.2583	-1.0782	1.4840	-1.3890	1.9118	-1.5434	2.1243
130°	-0.8903	1.0610	-1.0587	1.2617	-1.2597	1.5012	-1.6562	1.9738	-1.8598	2.2165
134°	-1.0246	1.0610	-1.2209	1.2643	-1.4654	1.5175	-1.9703	2.0403	-2.2393	2.3189
138°	-1.1783	1.0610	-1.4064	1.2664	-1.7027	1.5331	-2.3465	2.1128	-2.7047	2.4353
142°	-1.3580	1.0610	-1.6229	1.2679	-1.9817	1.5482	-2.8081	2.1939	-3.2913	2.5715
146°	-1.5730	1.0610	-1.8816	1.2692	-2.3176	1.5632	-3.3909	2.2872	-4.0570	2.7365
150°	-1.8377	1.0610	-2.1999	1.2701	-2.7338	1.5784	-4.1545	2.3986	-5.1025	2.9460
154°	-2.1754	1.0610	-2.6057	1.2709	-3.2686	1.5942	-5.2045	2.5384	-6.6208	3.2292
158°	-2.6262	1.0610	-3.1471	1.2715	-3.9884	1.6114	-6.7470	2.7260	-9.0300	3.6484
162°	-3.3654	1.0610	-3.9152	1.2721	-5.0210	1.6314	-9.2448	3.0038	-13.4322	4.3644
166°	-4.2564	1.0610	-5.1047	1.2727	-6.6450	1.6568	-13.9882	3.4877	-23.9161	5.9630
170°	-6.0204	1.0610	-7.2230	1.2736	-10.1609	1.7004	-26.3596	4.6480		
174°	-10.1111	1.0610	-12.1370	1.2758	-16.8192	1.7678				

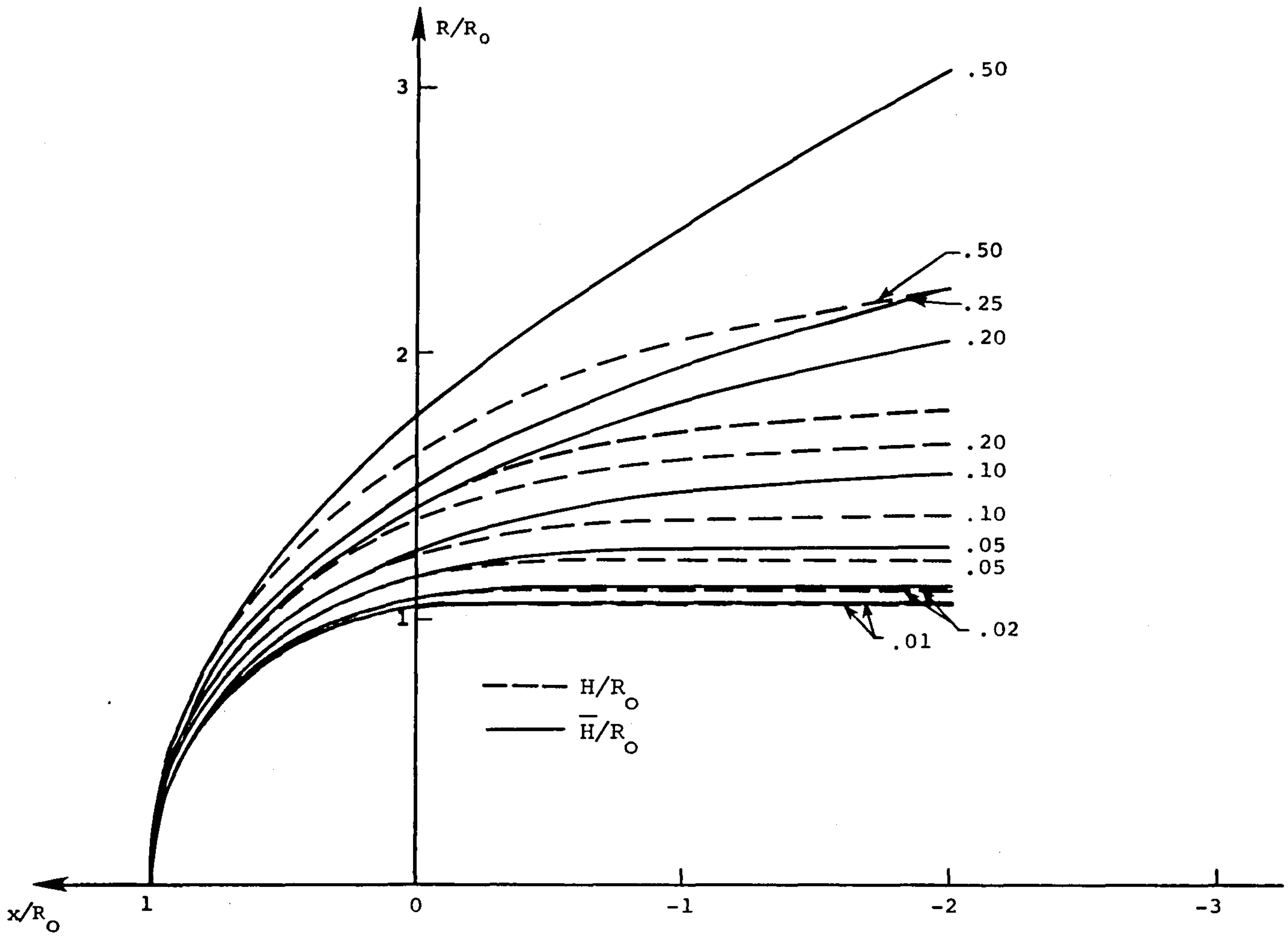
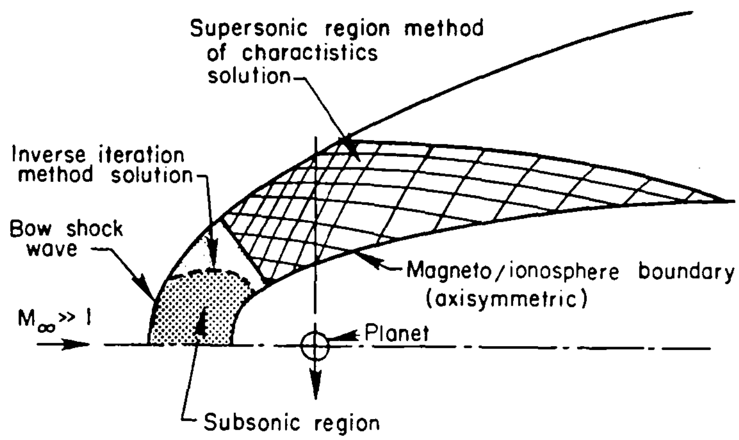
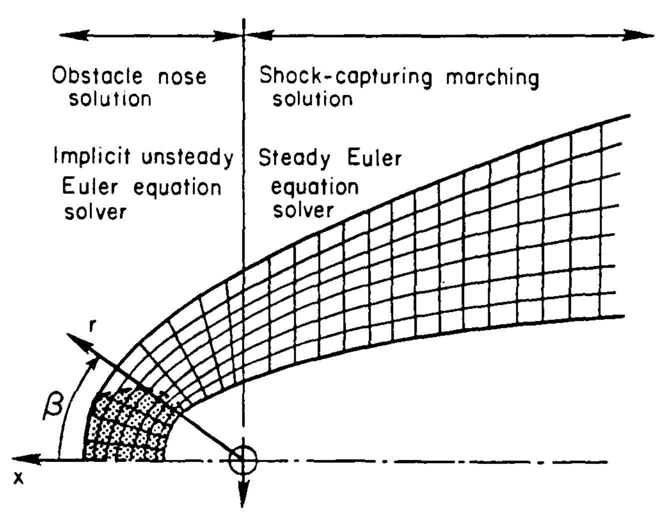


Figure 1.- Illustration of ionopause shapes for atmospheres with various (i) constant scale heights H/R_o and (ii) gravitational variation included in the scale height $\bar{H}/R_o$.



(a) Former method.



(b) Present method.

Figure 2.- Comparison of former and present computational procedures for determining the gasdynamic flow properties of solar wind-magneto/ionopause interactions.

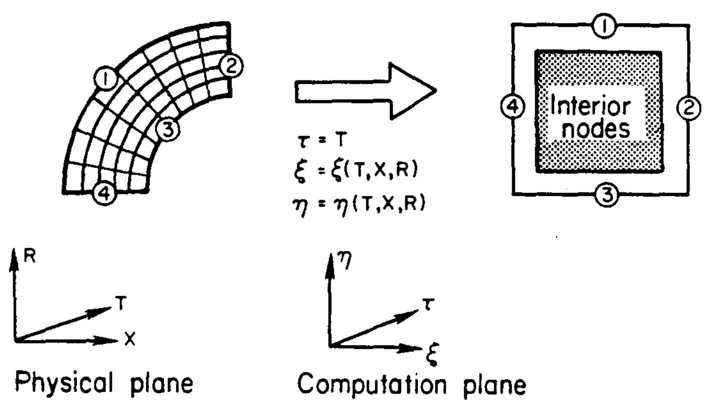


Figure 3.- Transformation from physical domain to rectangular computational domain.

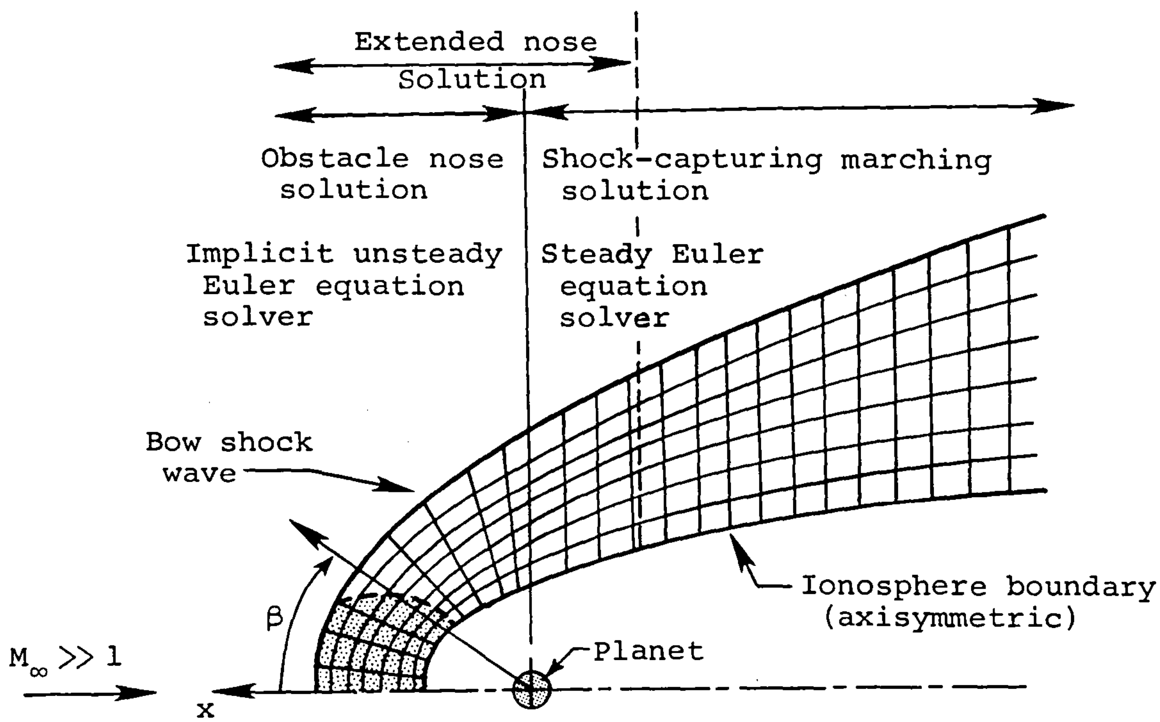


Figure 4.- Illustration of capability for providing an additional flow-field segment to the obstacle nose solution in the computational procedure for determining the gasdynamic flow properties of solar wind-ionopause interactions.

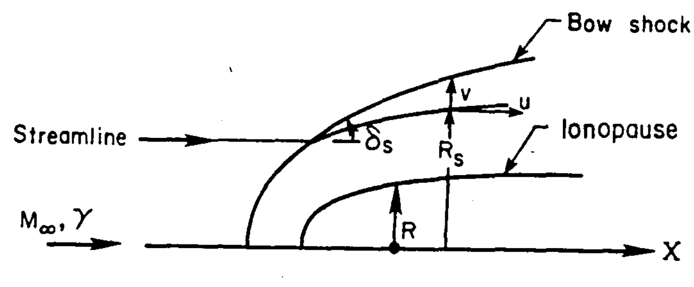


Figure 5.- Illustration of quantities used for streamline calculation.

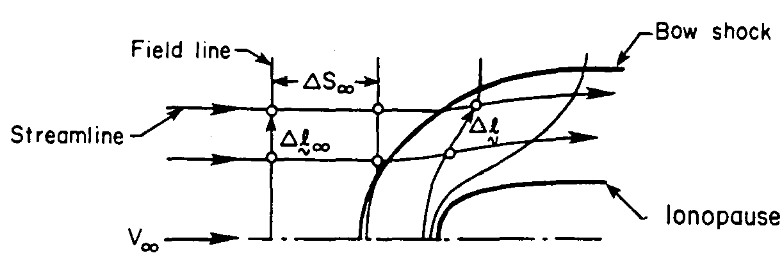


Figure 6.- Illustration of quantities used for magnetic field-line calculation in the plane of magnetic symmetry.

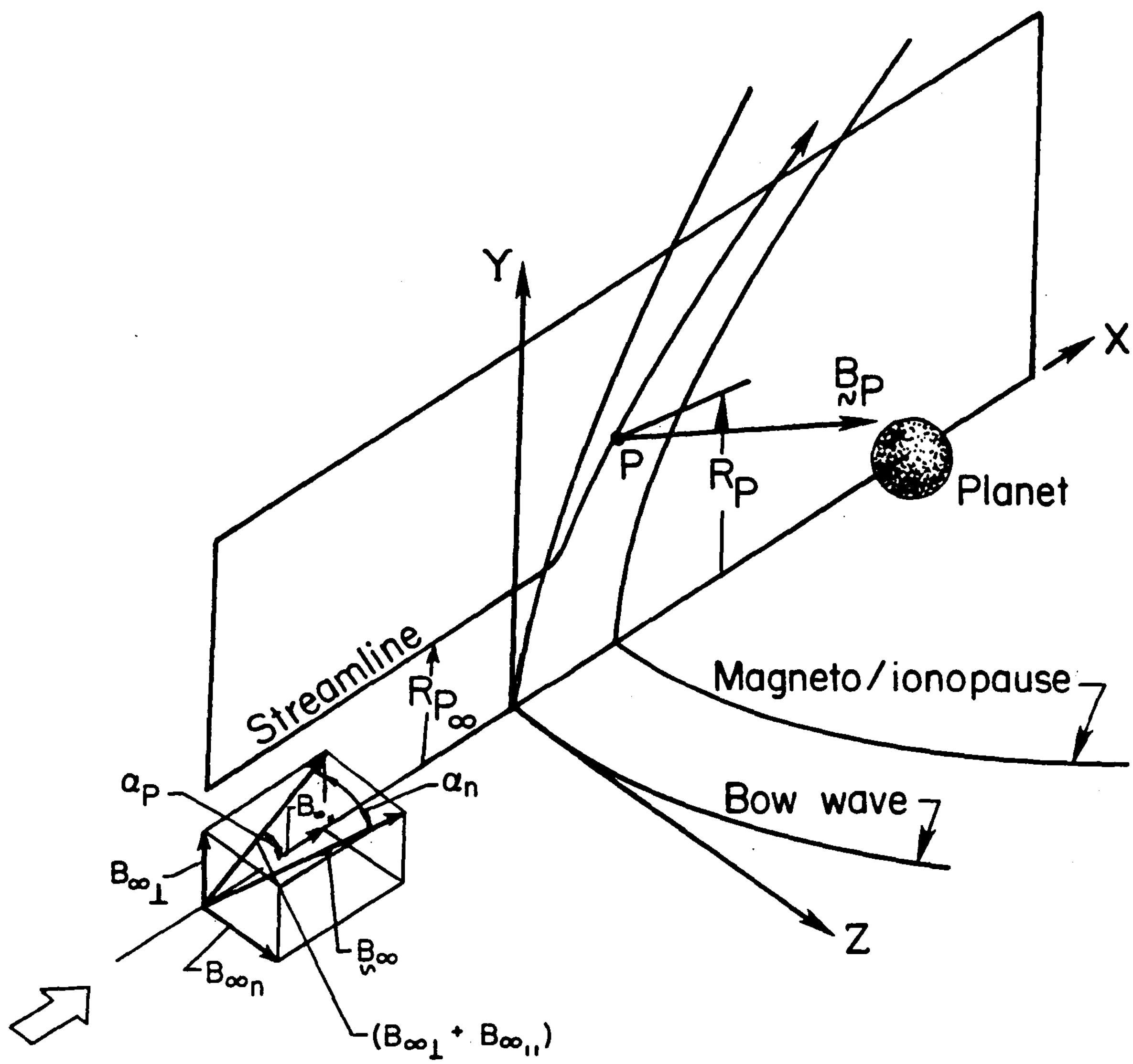


Figure 7.- Illustration of the components of the three-dimensional magnetic field.

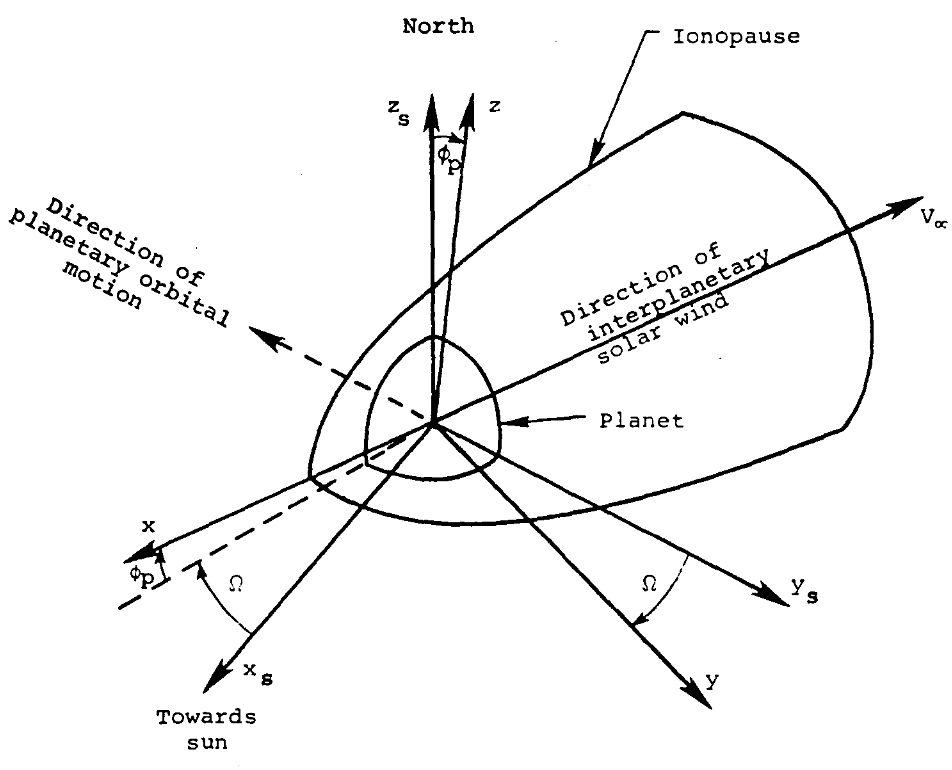


Figure 8.- Illustration of sun-planet (x_s, y_s, z_s) and solar wind (x,y,z) coordinate systems and the azimuthal (Ω) and polar (ϕ_p) solar-wind angles, both shown in a positive sense.

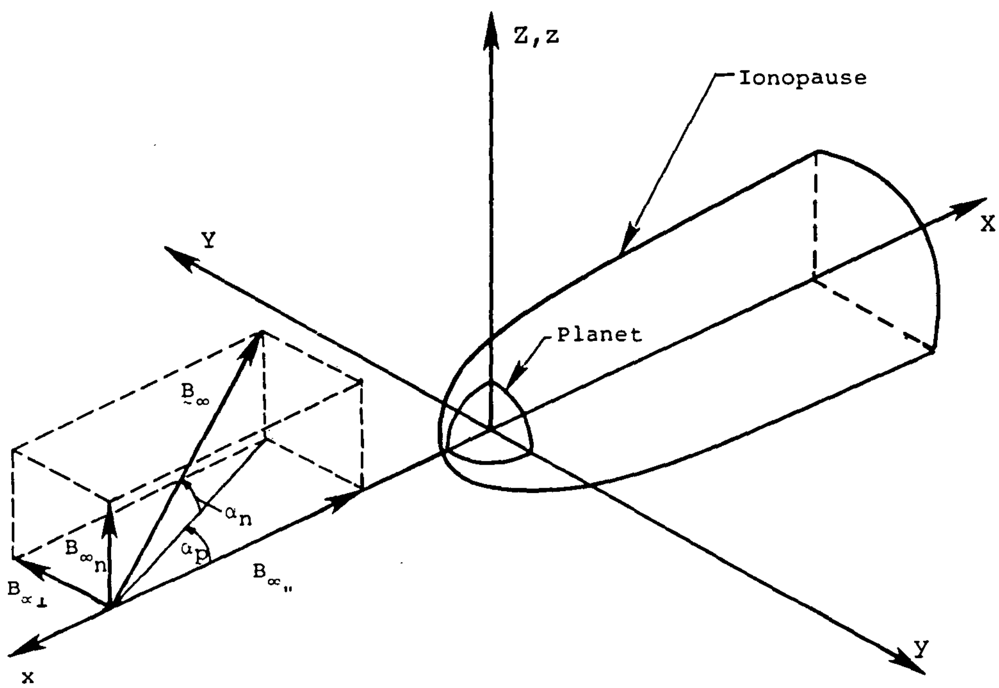


Figure 9.- Illustration of solar-wind (x,y,z) and (X,Y,Z) coordinate systems and the interplanetary magnetic field and magnetic-field angles (α_p,α_n).

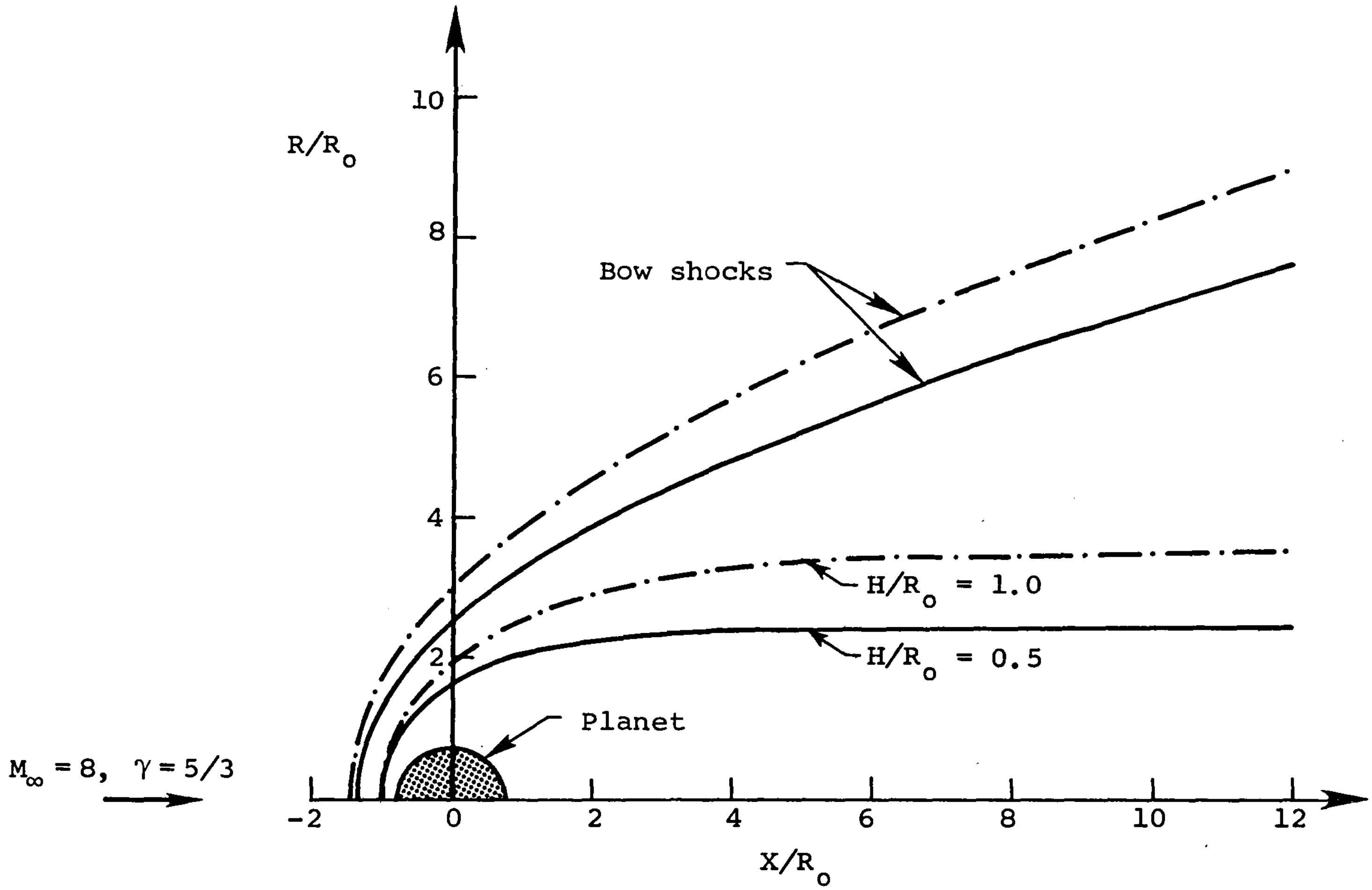


Figure 10.- Bow shock locations for $M_\infty = 8.0$, $\gamma = 5/3$ flow past constant scale-height ionopause shapes with $H/R_o = 0.5$ and 1.0.

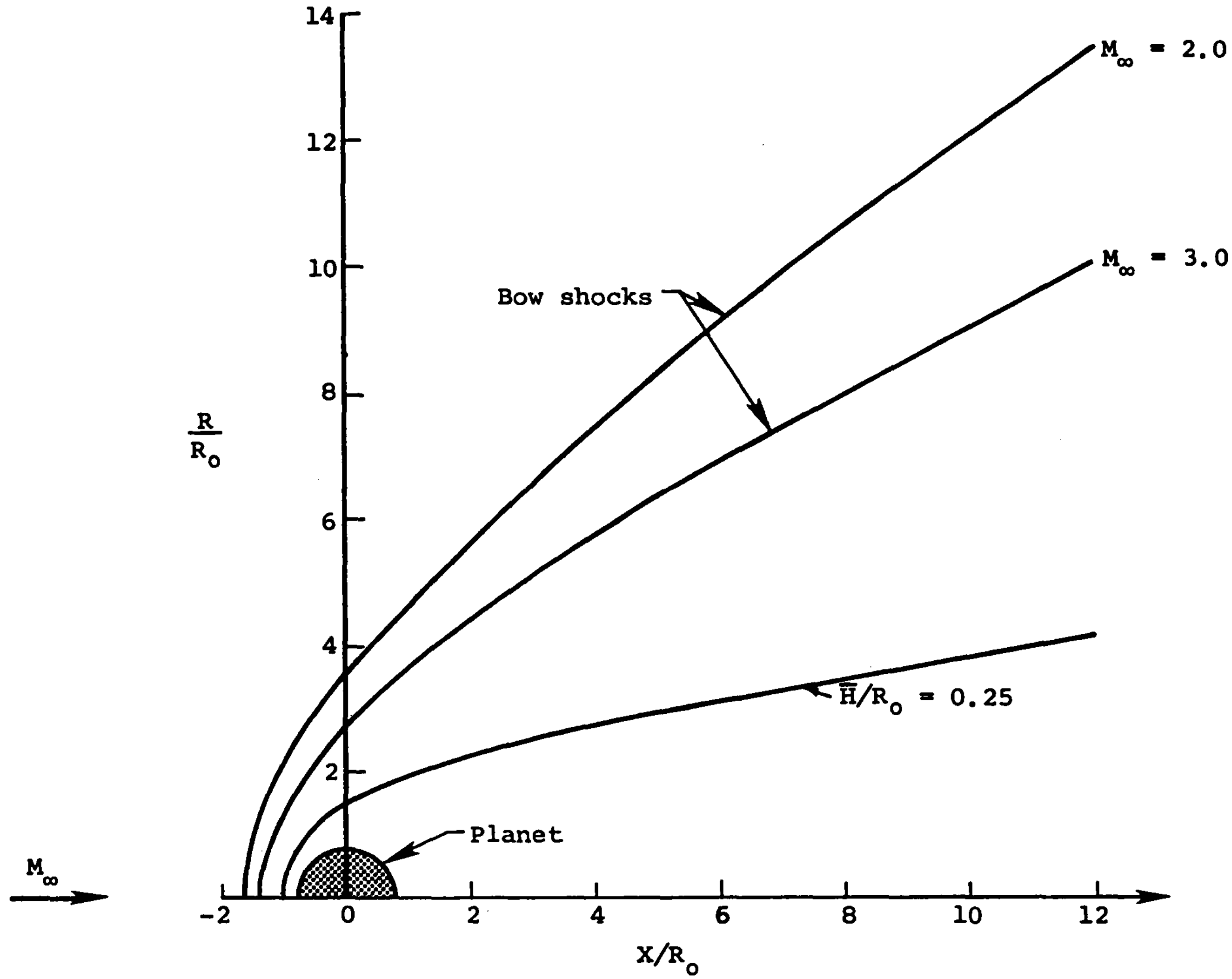


Figure 11.- Bow shock shapes for flow past an ionopause shape with gravitational variation included in scale height with $\bar{H}/R_o = 0.25$, $\gamma = 5/3$ and $M_\infty = 2.0$ and 3.0.

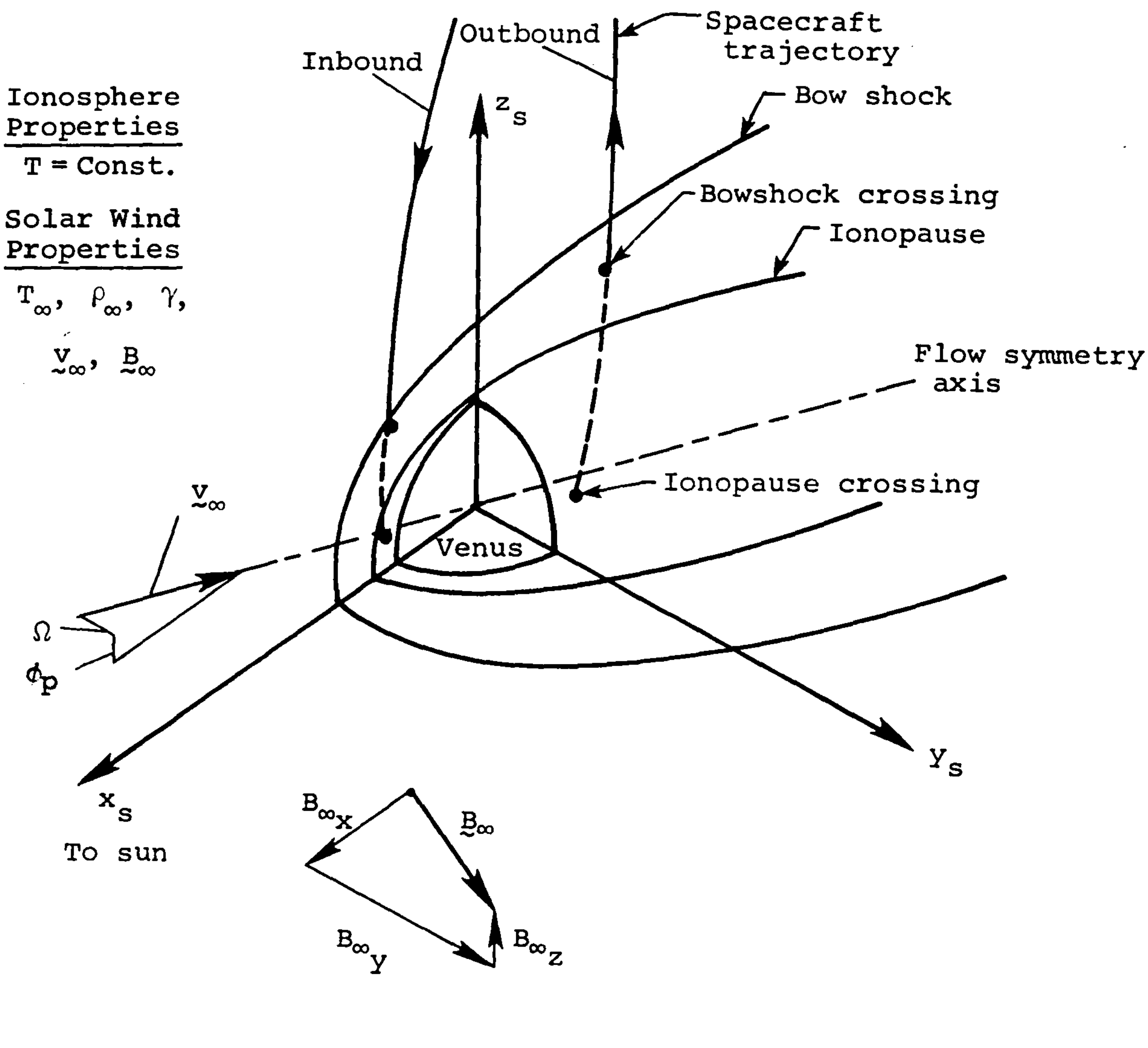


Figure 12.- Overall features of Pioneer-Venus orbiter trajectory crossings of solar-wind/Venus-ionosphere interaction region.

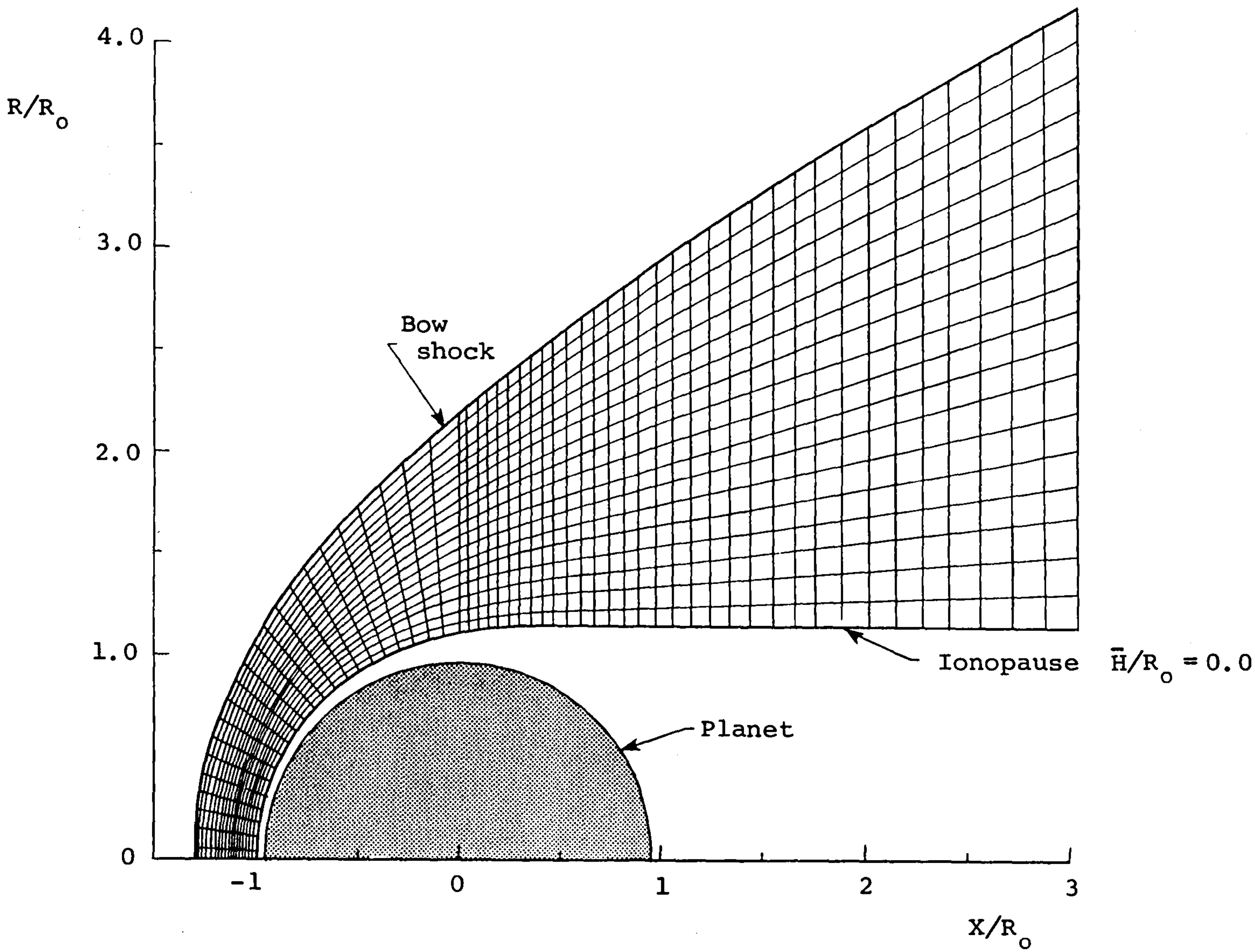


Figure 13.- Illustration of typical flow-field grid density for gasdynamic solution; $M_\infty = 3.0$, $\gamma = 5/3$.

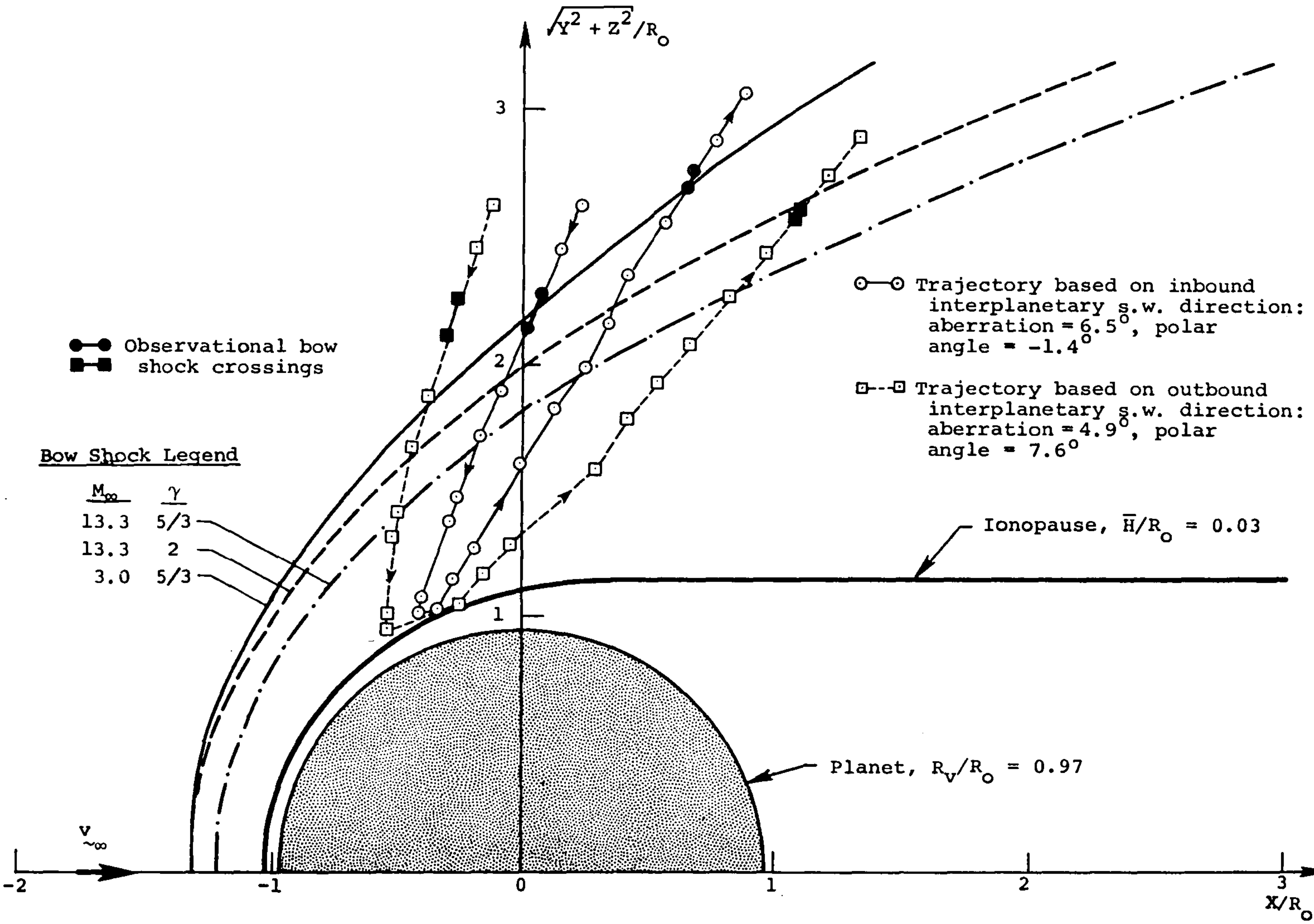


Figure 14.- P-V Orbit 6 trajectories and observational bow shock crossings as viewed in solar-wind coordinates based on inbound and outbound interplanetary solar-wind directions; also, various bow shock shapes for different interplanetary solar-wind conditions.

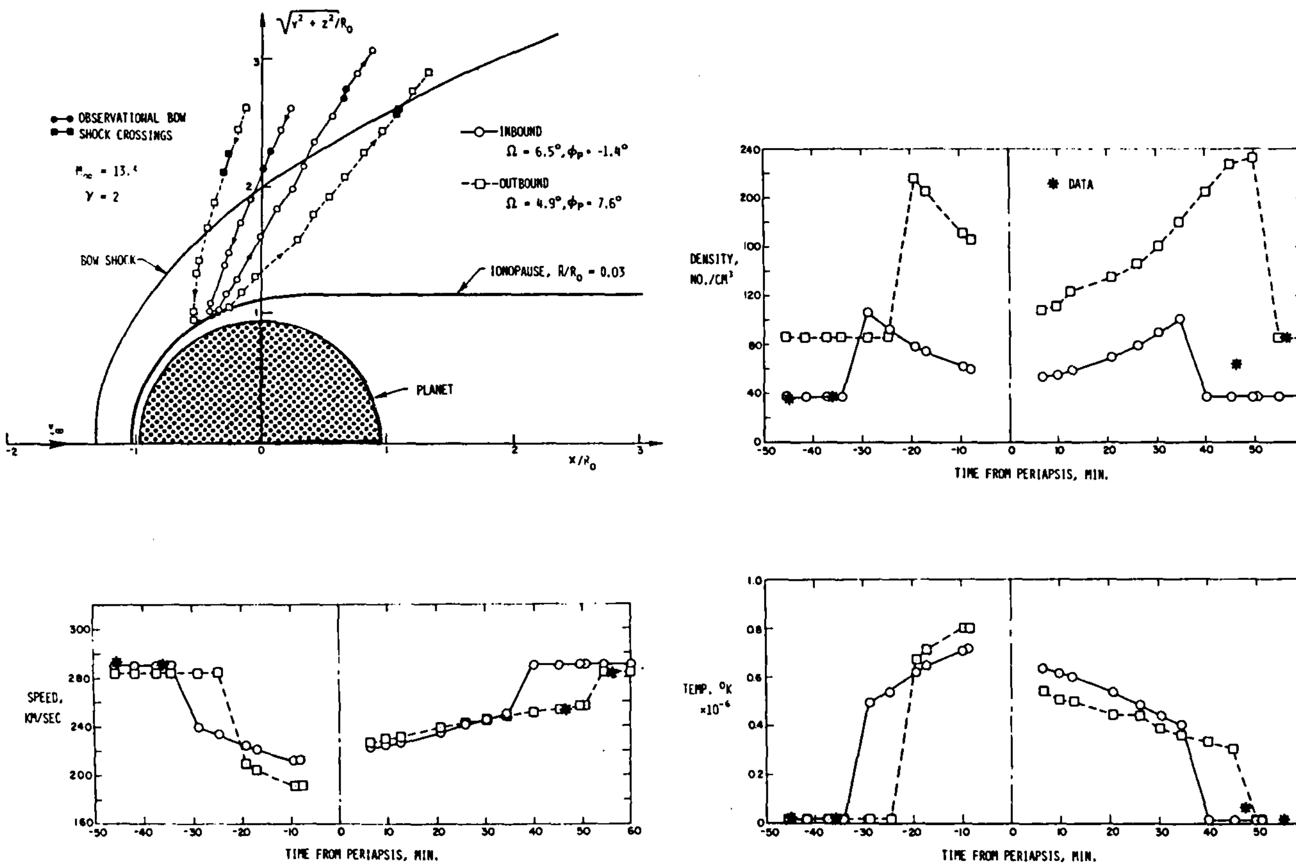
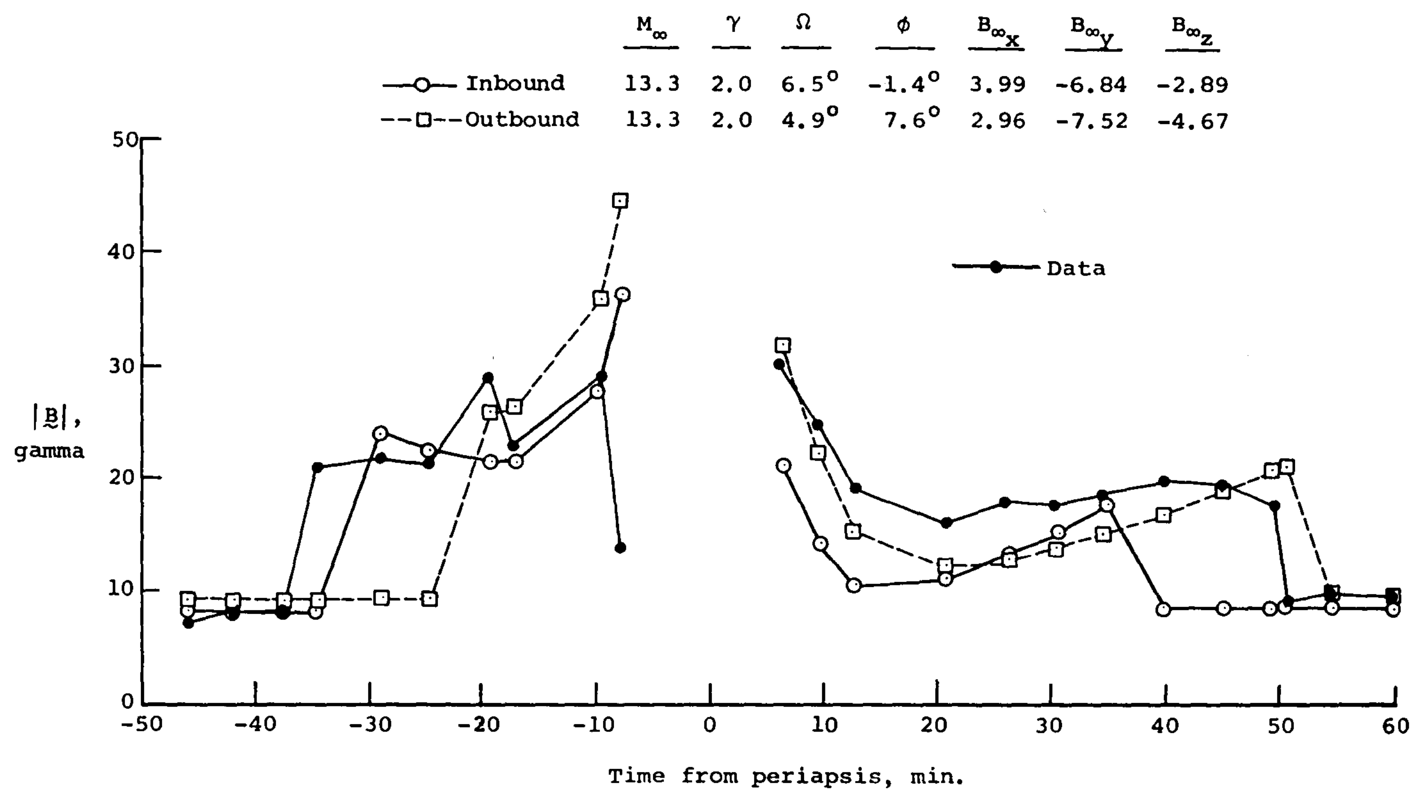
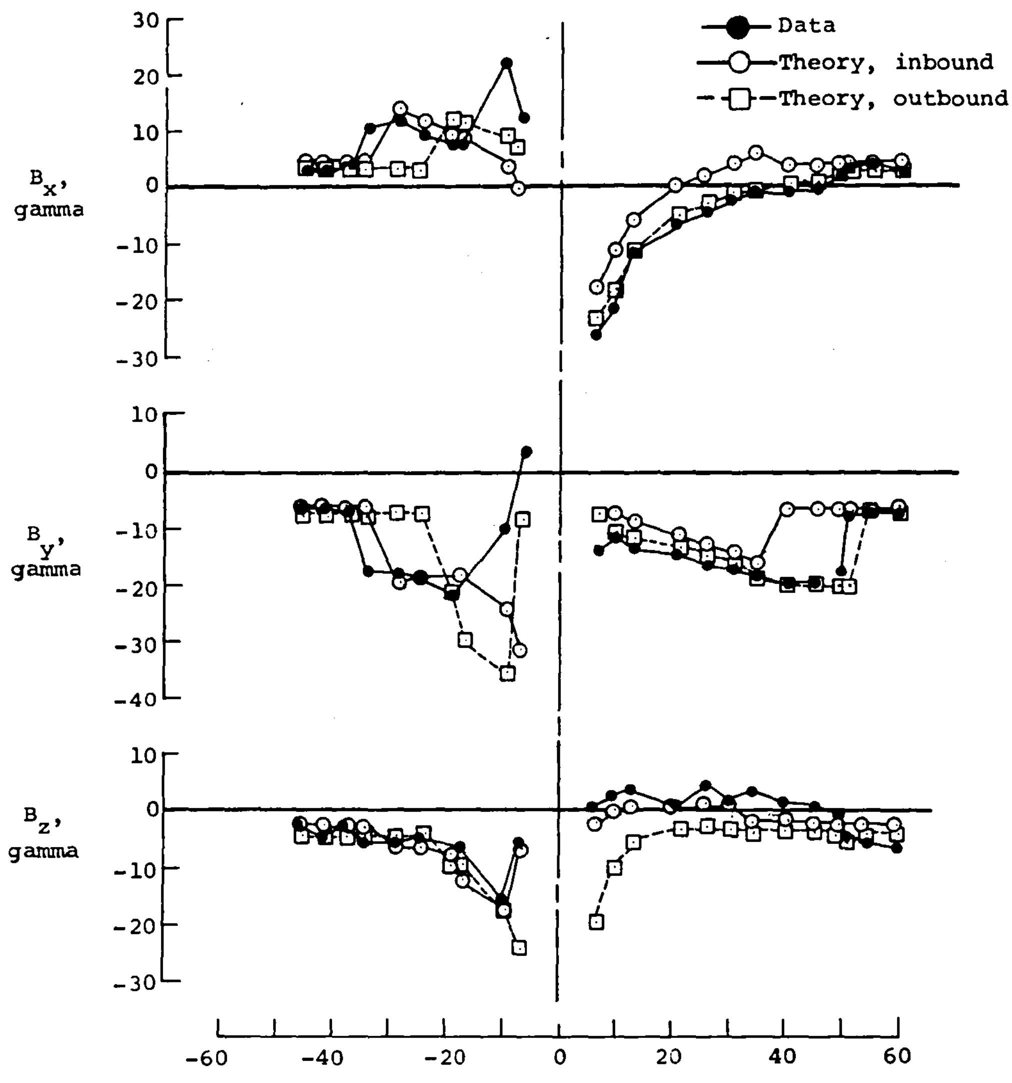


Figure 15.- Comparison of observed (OPA) and theoretical time histories of ionosheath plasma properties for P-V Orbit 6 based on inbound and outbound interplanetary solar-wind conditions using a gasdynamic solution for $M_\infty = 13.3$, $\gamma = 2.0$.



(a) Magnetic-field magnitude.

Figure 16.- Comparison of observed (OMAG) and theoretical time histories for the magnitude of the magnetic field for P-V Orbit 6 based on inbound and outbound interplanetary conditions using gasdynamic solution for $M_\infty = 13.3$, $\gamma = 2$.



Time from periapsis, min.

(b) Magnetic-field components.

Figure 16.- Concluded.

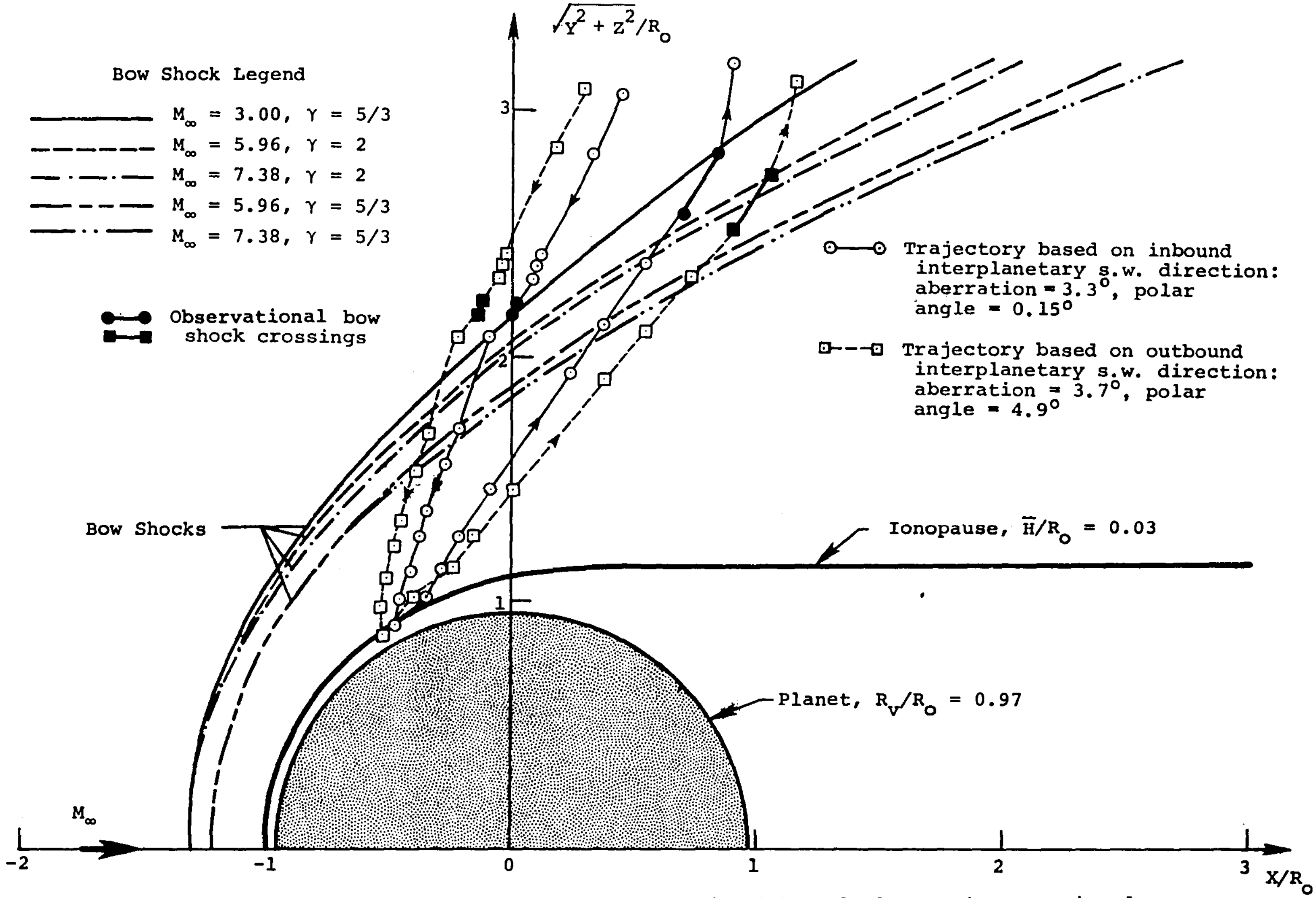


Figure 17.- P-V Orbit 3 trajectories and observational bow shock crossings as viewed in solar-wind coordinates based on inbound and outbound interplanetary solar-wind directions; also, various bow shock shapes for different interplanetary solar wind conditions.

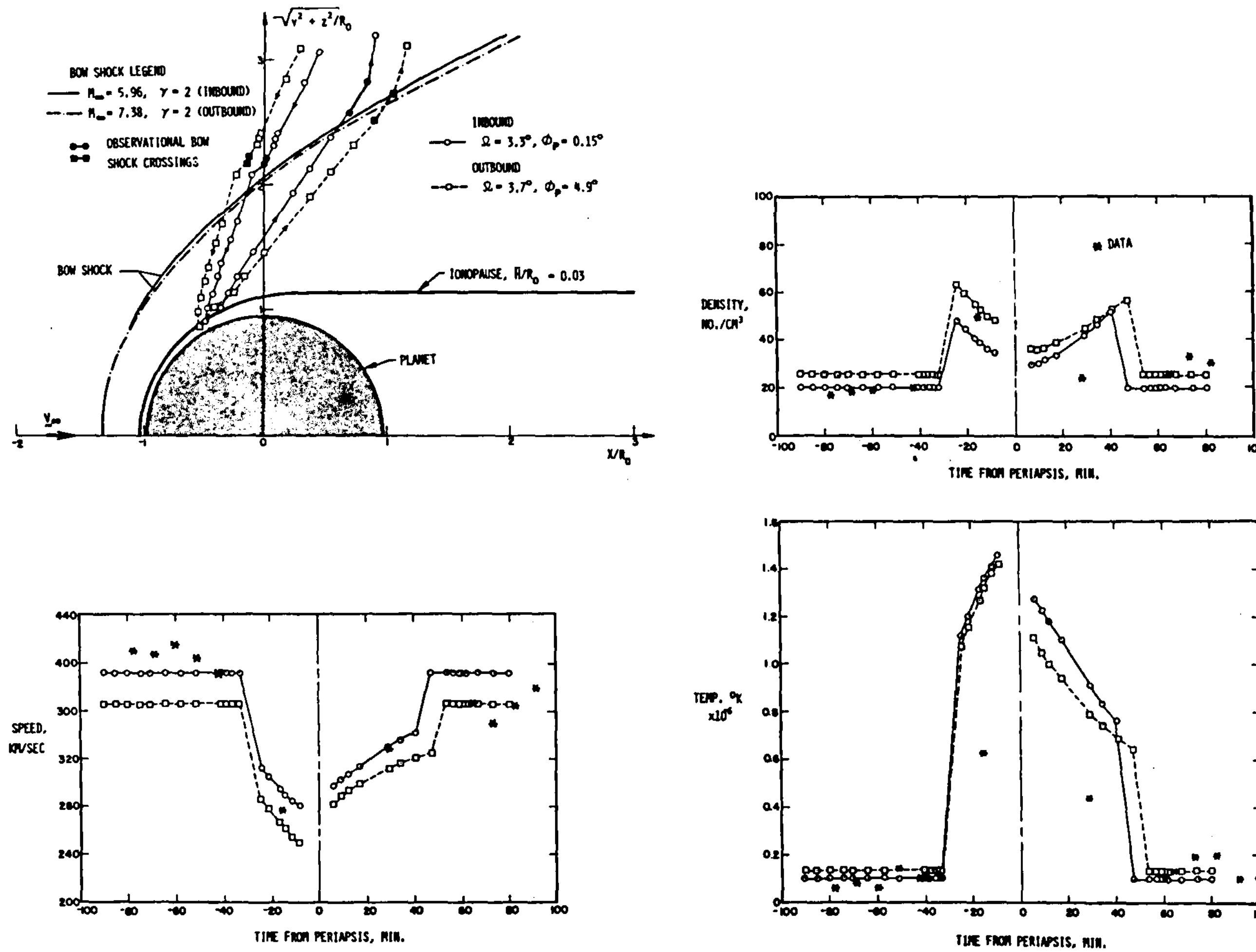
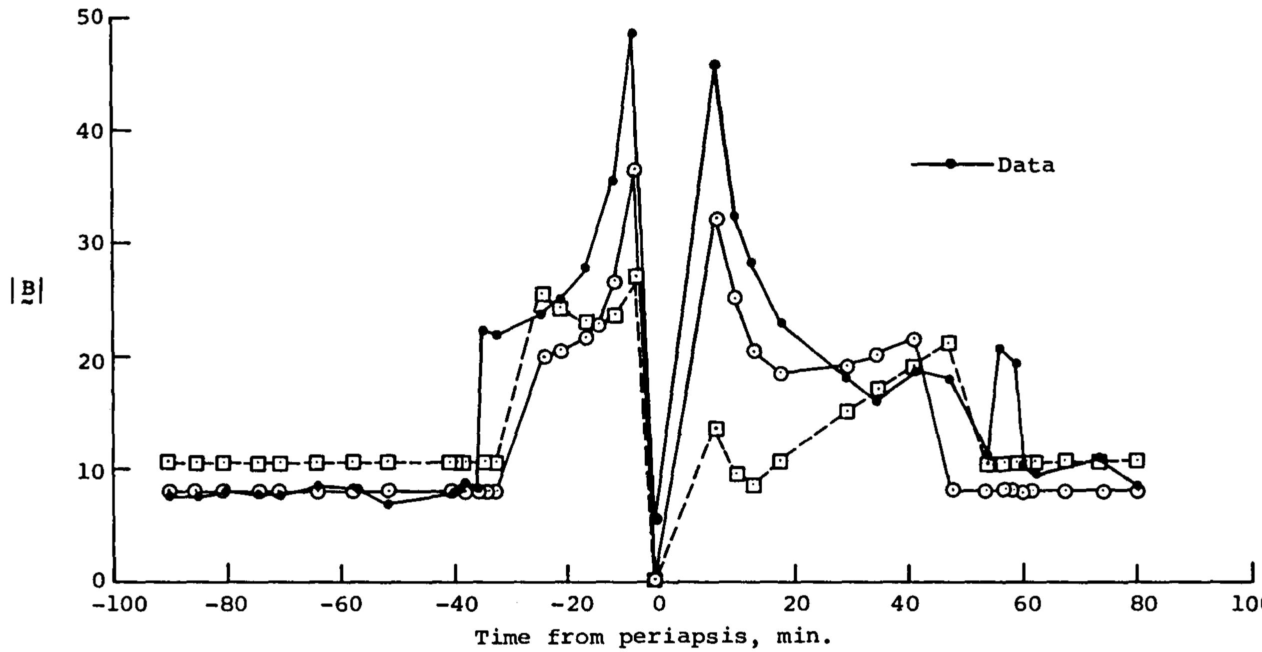


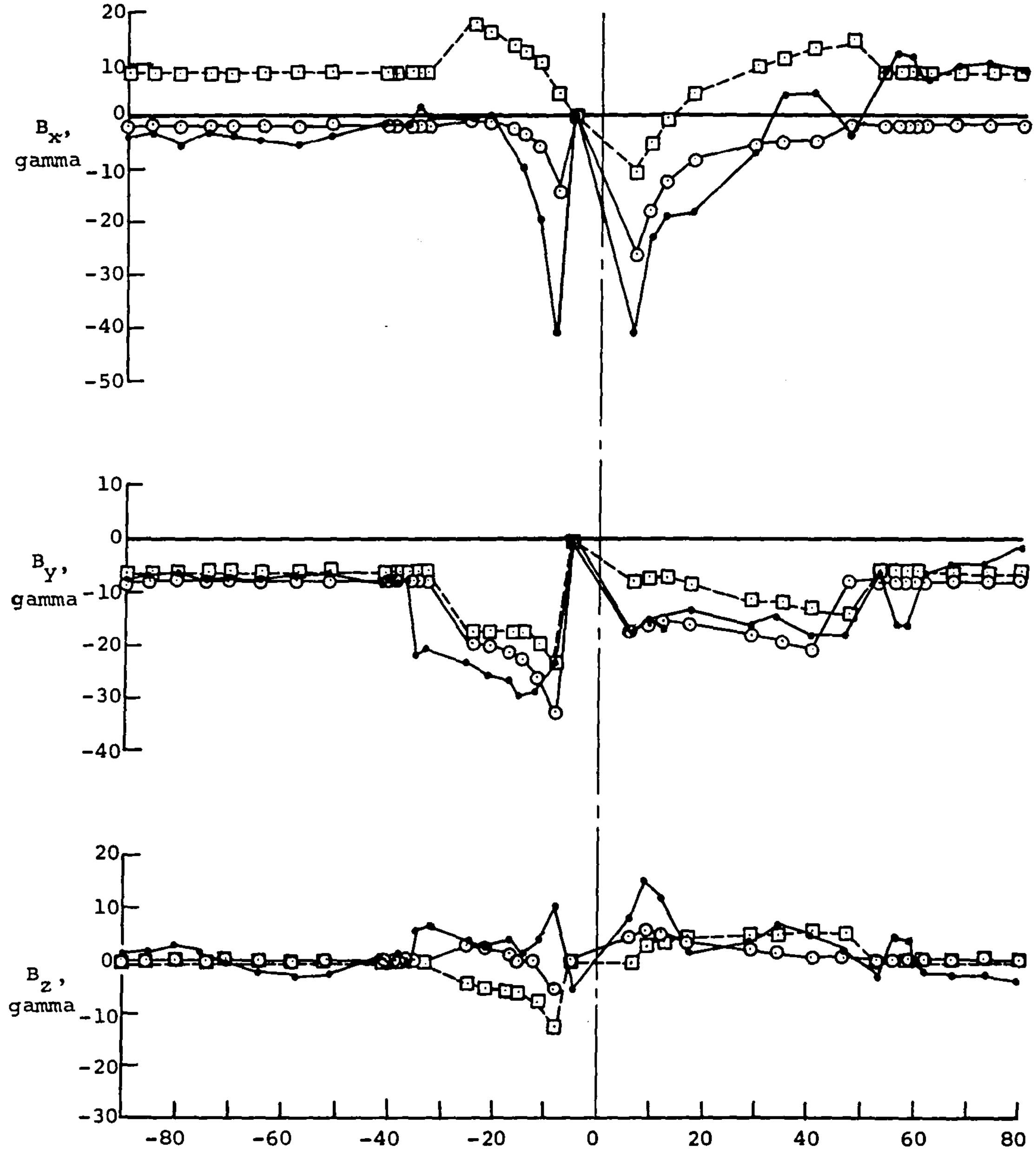
Figure 18.- Comparison of observed and theoretical time histories of ionosheath plasma properties for P-V Orbit 3 based on inbound and outbound interplanetary solar-wind conditions.

Theory	M_∞	γ	Ω	ϕ_p	B_{x_∞}	B_{y_∞}	B_{z_∞}
—○— Inbound	7.38	2.0	3.3°	0.15°	-1.78	-7.86	0.08
—□— Outbound	5.96	2.0	3.7°	4.90°	-8.38	-6.45	-0.31



(a) Magnetic-field magnitudes.

Figure 19.- Comparison of observed (OMAG) and theoretical time histories for the magnetic field for P-V Orbit 3 based on inbound and outbound interplanetary solar-wind conditions using gasdynamic solutions $M_\infty = 7.38$, $\gamma = 2.0$ for inbound and $M_\infty = 5.96$, $\gamma = 2.0$ for outbound calculations.

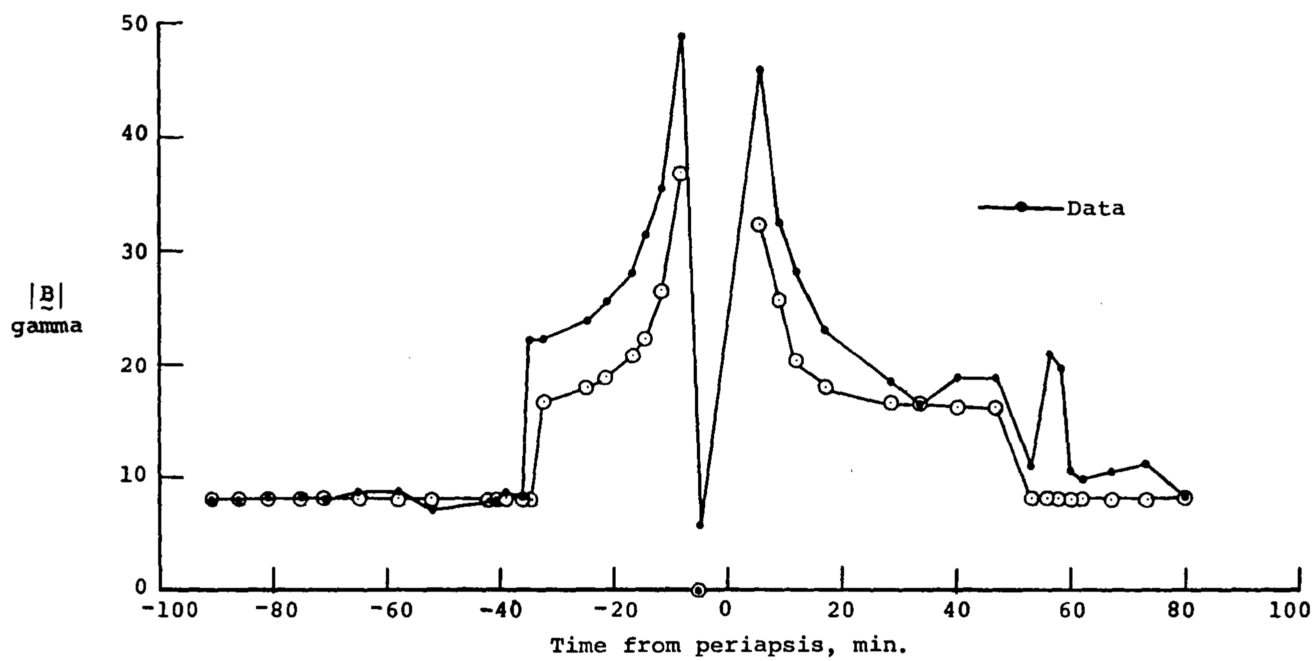


(b) Magnetic-field components.

Figure 19.- Concluded.

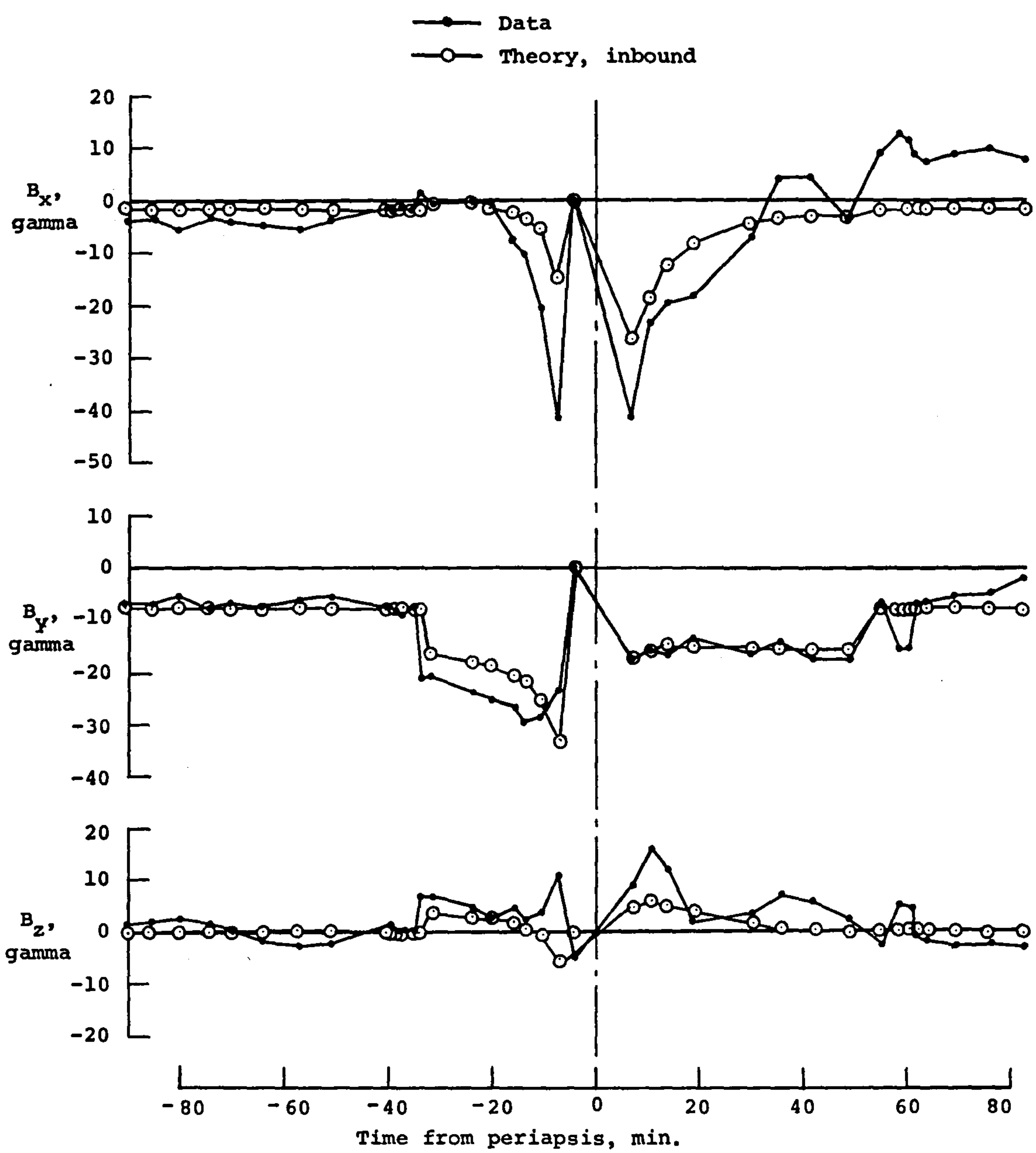
Interplanetary Conditions

Theory		M_∞	γ	Ω	ϕ_p	B_{x_∞}	B_{y_∞}	B_{z_∞}
—O—	Inbound	3.0	5/3	3.3°	0.15°	-1.78	-7.86	0.08



(a) Magnetic-field magnitude.

Figure 20.- Comparison of observed (OMAG) and theoretical time histories of the magnetic field for P-V Orbit 3 based on inbound solar wind interplanetary conditions using a gasdynamic solution for $M_\infty = 3.0$, $\gamma = 5/3$.



(b) Magnetic-field components.

Figure 20.- Concluded.

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
NASA CR-3267		

4. Title and Subtitle	5. Report Date
APPLICATION OF ADVANCED COMPUTATIONAL PROCEDURES FOR MODELING SOLAR-WIND INTERACTIONS WITH VENUS - THEORY AND COMPUTER CODE	May 1980
	6. Performing Organization Code: 498/C
7. Author(s): Stephen S. Stahara, Daniel Klenke, Barbara C. Trudinger, and John R. Spreiter	8. Performing Organization Report No.: NEAR TR 202
9. Performing Organization Name and Address: Nielsen Engineering & Research, Inc. 510 Clyde Avenue Mountain View, CA 94043	10. Work Unit No.
	11. Contract or Grant No.: NASW-3182
12. Sponsoring Agency Name and Address: National Aeronautics and Space Administration Washington, DC 20546	13. Type of Report and Period Covered: Contractor Report 7/78 - 9/79
	14. Sponsoring Agency Code

15. Supplementary Notes

NASA Headquarters Technical Monitor: Robert E. Murphy

16. Abstract

Advanced computational procedures are developed and applied to the prediction of solar-wind interaction with nonmagnetic terrestrial-planet atmospheres, with particular emphasis to Venus. The theoretical method is based on a single-fluid, steady, dissipationless, magnetohydrodynamic continuum model, and is appropriate for the calculation of axisymmetric, supersonic, super-Alfvénic solar-wind flow past terrestrial planets. The procedures, which consist of finite-difference codes to determine the gasdynamic properties and a variety of special-purpose codes to determine the frozen magnetic field, streamlines, contours, plots, etc. of the flow, are organized into one computational program which has been extensively documented and is presented in a general user's manual included as part of this report.

Theoretical results based upon these procedures are reported for a wide variety of solar-wind conditions and ionopause obstacle shapes. Plasma and magnetic-field comparisons in the ionosheath are also provided with actual spacecraft data obtained by the Pioneer-Venus Orbiter. These results have verified the appropriateness of the basic theoretical model, and have indicated the importance of accounting for the variable oncoming direction of the interplanetary solar wind.

17. Key Words (Suggested by Author(s))	18. Distribution Statement
Solar-Wind/Ionosphere Interaction Finite-Difference Methods Steady Flow Frozen Magnetic Field	Unclassified - Unlimited Subject Category 92

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price*
Unclassified	Unclassified	316	$11.75

*For sale by the National Technical Information Service, Springfield, Virginia 22161

NASA-Langley, 1980

End of Document