AEDC-TR-77-20

cy. 1

VISCOUS SHOCK LAYER SOLUTIONS FOR HYPERSONIC SPHERE-CONES

DEPARTMENT OF AEROSPACE ENGINEERING
UNIVERSITY OF CINCINNATI
CINCINNATI, OHIO 45221

January 1977

Final Report for Period March 1976 – October 1976

Approved for public release; distribution unlimited.

Prepared for

DIRECTORATE OF TECHNOLOGY (DYR)
ARNOLD ENGINEERING DEVELOPMENT CENTER
ARNOLD AIR FORCE STATION, TENNESSEE 37389

NOTICES

This final report was submitted by University of Cincinnati, Cincinnati, OH 45221 under contract F40600-74-C-0011, job order P00001, with the Arnold Engineering Development Center, Arnold Air Force Station, TN 37389. Mr. Elton R. Thompson, DYR, was the Air Force Project Engineer.

APPROVAL STATEMENT

This technical report has been reviewed and is approved for publication.

FOR THE COMMANDER

Elton R. Thompson Robert O Dietz

ELTON R. THOMPSON Research & Development Division Directorate of Technology	ROBERT O. DIETZ Director of Technology

UNCLASSIFIED

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: AEDC-TR-77-20	2. GOVT ACCESSION NO.
3. RECIPIENT'S CATALOG NUMBER	
4. TITLE (and Subtitle): VISCOUS SHOCK LAYER SOLUTIONS FOR HYPERSONIC SPHERE-CONES	5. TYPE OF REPORT & PERIOD COVERED: Final Report for Period - March - October 1976
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): B. N. Srivastava, M. J. Werle, and R. T. Davis	8. CONTRACT OR GRANT NUMBER(s): F40600-74-C-0011
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Department of Aerospace Engineering, University of Cincinnati, Cincinnati, Ohio 45221	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS: Program Element 65807F
11. CONTROLLING OFFICE NAME AND ADDRESS: Arnold Engineering Development Center (DYFS), Air Force Systems Command, Arnold Air Force Station, Tennessee 37389	12. REPORT DATE: January 1977
	13. NUMBER OF PAGES: 180
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE: N/A

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

1. Sphere-Cones.
2. Blunt cones
3. Cones -- Hypersonic flow.

18. SUPPLEMENTARY NOTES

Available in DDC

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

blunt bodies	layers
flow fields	shock(mechanics)
hypersonic flow	numerical analysis
equations	

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

An analysis and numerical solution of the fully viscous shock layer equations for hypersonic flow past spherically blunted cones is presented. Attention is drawn to the adverse effect in the numerical solution due to discontinuities in the flow derivatives caused by the discontinuity in the surface curvature at the sphere/cone tangency point, when the usual surface coordinate frame of reference is used. It is shown that a finite difference formulation that accounts for the imbedded gradient discontinuities resolves these nu-

20. ABSTRACT (Continued)

merical difficulties. This concept is demonstrated through a numerical scheme which utilizes a time dependent relaxation technique for the bow shock shape. A model problem analogous to the sphere/cone juncture problem is first formulated and finite difference schemes developed and demonstrated for this case. This approach is then extended to the solution of the viscous shock layer equations for hypersonic flow past spherically blunted cones with half cone angles varying from 30° to 0° at high Reynolds number. For these cases the present results are found to compare well with independent inviscid flow calculations. In addition, excellent comparisons with experimental pressure and heat transfer data on a 7.5° half angle cone are obtained.

AFSC
Arnold AFS Tenn

PREFACE

The results reported herein were obtained from the Arnold Engineering Development Center by the University of Cincinnati, Department of Aerospace Engineering under Contract F40600-74-C-0011. The authors of this report were B. N. Srivastava, M. J. Werle, and R. T. Davis. The Air Force Project Engineer for this contract was E. R. Thompson, AEDC/DYR. The Program Element Number was 65807F.

This report covers the work during the period March 1976 to October 1976. The reproducibles used in the reproduction of this report were supplied by the authors.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

Recent developments in aerodynamics and space flight have caused increasingly focused attention on the problem of theoretically predicting the blunt body flow field. Three current numerical approaches for treating this problem include solution of either the full Navier-Stokes equations, the second-order boundary-layer equations, or the viscous shock layer equations. Use of the full Navier-Stokes equations [1] has been quite successful in providing solutions for stagnation regions, but generally have been applied for only one nose radius downstream. This is because the elliptic nature of these equations increases the complexity of the solution procedure and restricts the application of these methods in the downstream direction. While there are several computational difficulties associated with the second-order boundary layer approach [2], many of the difficulties associated with computing viscous hypersonic flows over blunt bodies can be overcome through use of the viscous shock layer equations. In this approach the entire flow field, from the body to the shock, is treated in a unified manner. This is the approach taken here, wherein the basic method of Davis [2] is applied to nonanalytic bodies such as spherically blunted cones.

Blunted cones merit further study, both because some blunting of the tip of a sharp cone due to extreme heating appears unavoidable and because blunting has favorable effects on transition. Previous research [3, 4] has shown that a small amount of bluntness to a sharp cone is conducive to a delay in boundary layer transition. Unfortunately only limited work has been done in the past for such nonanalytic bodies, in part, due to the difficulties associated with the discontinuity occurring in the surface curvature at the sphere/cone juncture point whenever a surface coordinate system is used. Miner and Lewis [5] and Kang and Dunn [6] have been able to obtain some numerical solutions for such nose cone problems. The approach of Miner and Lewis [5] was to smoothen the effect of the curvature discontinuity at the sphere/cone tangency point by constructing an artificially continuous distribution of curvature. Kang and Dunn [6] approached the same problem by application of a Karman-Pohlhausen integral method to the Navier-Stokes equations under a thin shock layer assumption. They treated the nose region as well as the afterbody conical section of the spherically blunted cones in a similar manner. While these solutions yield acceptable numerical results, it appears that a more consistant formulation, which accounted for the flow behavior at the sphere cone tangency point, could improve these solutions and enhance their reliability.

The basic difficulty, however, is that only a very limited amount of information is available for the flow behavior at such a curvature discontinuity - information that could be utilized in a computational procedure to achieve more accurate solutions. It is known from inviscid considerations that a streamline curvature discontinuity produces a discontinuity in the flow gradients along that streamline. Viscous effects in such problems are expected to remove such surface gradient jumps in a manner shown by Messiter and Hu [7] for two dimensional flows. Their analysis for high Reynolds number flows shows that the jump in pressure gradient predicted by the inviscid flow theory is smoothed by viscous effects through a triple deck structure near the juncture region. However the region within which this smoothing takes place is found to be small for realistic finite Reynolds numbers and it would be anticipated that a viscous numerical calculation with a finite mesh size might not capture this physical phenomenon and would therefore still predict a gradient discontinuity. Thus, one is faced with the prospect of generating finite difference approximations for regions in which the flow gradients are virtually discontinuous.

An added complication arises because the viscous shock layer equations, when written in a surface coordinate system, contain explicit dependence on the surface curvature. The surface curvature undergoes a discontinuous change from a value of one on the sphere to a value of zero on the cone portion resulting in the appearance of gradient discontinuities in all flow variables at all points normal to the body surface across the shock layer thickness at the juncture point. These discontinuities are purely an artifice and occur only because of the choice of surface coordinates. None-the-less they must be accounted for in any numerical approximating procedure if reliable results are to be achieved.

An analytical assessment of the flow behavior at the sphere/cone juncture point and subsequently a proper numerical formulation of this problem is the purpose of the present study. It is believed that when a surface coordinate frame of reference is used a finite difference formulation of the governing equations must be such that the longitudinal derivatives be carefully evaluated at the sphere/cone juncture point in order to eliminate large numerical truncation errors. This can be done by ensuring that the finite difference form of the longitudinal derivatives avoid any differencing across the sphere/cone juncture discontinuity. This technique is demonstrated in the present approach where the numerical scheme utilizes a time dependent relaxation technique for the shock wave shape. A model problem analogous to the sphere/cone juncture problem is first formulated and finite difference schemes developed and demonstrated for this case. It is shown that the present method accurately captures the anticipated discontinuous behavior of the flow derivatives at the model juncture point. This concept is then extended for the solution of the viscous shock layer equations for hypersonic flow past spherically blunted cones with half cone angles varying from 30° to 0° under various free stream conditions. The results indicate good comparisons with inviscid solutions and experimental data.

II. GOVERNING EQUATIONS

The viscous shock layer concept has been presented in detail by Davis [2] and therefore is only summarized here. The compressible Navier-Stokes equations are written in a boundary layer like coordinate system (see Fig. 1) and nondimensionalized by variables which are of order one in the region near the body surface (boundary layer) for large Reynolds numbers. The same set of equations are then written in variables which are of order one in the essentially inviscid region outside the boundary layer. In the final set of equations terms are retained that are second-order in the inverse square root of a Reynolds number. A comparison of the two sets of equations is then made and one set of equations is found from them which is valid to second order in both the (inviscid) outer and inner (viscous) regions. A solution to this set of equations is thus uniformly valid to second order in the entire shock layer for arbitrary Reynolds number. The resulting equations (and notation) are the same as those presented by Davis [2] and are given as:

Continuity:

$$[(r+n\cos\phi)^j \rho u]_s + [(1+\kappa n)(r+n\cos\phi)^j \rho v]_n = 0 \qquad \text{(1a)}$$

Longitudinal Momentum:

$$\rho\{u\, u_s/(1+\kappa n) + v\, u_n + \kappa uv/(1+\kappa n)\} + p_s/(1+\kappa n)$$

$$= [\varepsilon^2/(1+\kappa n)^2(r+n\cos\phi)^j][(1+\kappa n)^2(r+n\cos\phi)^j\tau]_n \tag{1b}$$

where,

$$\tau = \mu[u_n - \kappa u/(1+\kappa n)] \tag{1c}$$

Normal Momentum:

$$\rho\{u\, v_s/(1+\kappa n) + v\, v_n - \kappa u^2/(1+\kappa n)\} + p_n = 0 \tag{1d}$$

which with the thin shock layer approximation becomes,

$$-\rho\, \kappa u^2/(1+\kappa n) + p_n = 0 \tag{1e}$$

Energy Equation:

$$\rho\{uT_s/(1+\kappa n) + vT_n\} - u\, p_s/(1+\kappa n) - v\, p_n = \varepsilon^2\tau^2/\mu$$

$$+[\varepsilon^2/(1+\kappa n)(r+n\cos\phi)^j][(1+\kappa n)(r+n\cos\phi)^j q]_n \tag{1f}$$

where

$$q = \mu\, T_n/\sigma \tag{1g}$$

Equation of State:

$$p = (\gamma-1)\rho T/\gamma \tag{1h}$$

Viscosity Law:

$$\mu = T^{3/2}(1+c')/(T+c') \tag{1i}$$

where

$$c' = c^*/M_\infty^2\ T_\infty^*\ (\gamma-1)$$

and c^* is taken to be 198.6°R for air.

The boundary conditions employed here are the no slip surface conditions,

$$u(s,o) = v(s,o) = 0 \tag{2a}$$

and

$$T(s,o) = T_w \tag{2b}$$

while at the shock location the oblique shock relations are used to relate the flow variables just aft of the shock to the free stream conditions through the local shock slope. These relations are given as

$$u_{sh} = \tilde{u}_{sh} \sin(\alpha+\beta) + \tilde{v}_{sh} \cos(\alpha+\beta) \tag{3a}$$

$$v_{sh} = - \tilde{u}_{sh} \cos(\alpha+\beta) + \tilde{v}_{sh} \sin(\alpha+\beta) \tag{3b}$$

where $\tilde{u}_{sh}$ and $\tilde{v}_{sh}$ are velocity components at the shock given as

$$\tilde{u}_{sh} = \cos\alpha \tag{3c}$$

$$\tilde{v}_{sh} = - \sin\alpha/\rho_{sh} \tag{3d}$$

and

$$\rho_{sh} = \gamma p_{sh}/(\gamma-1) T_{sh} \tag{3e}$$

with

$$p_{sh} = [2/(\gamma+1)]\sin^2\alpha - (\gamma-1)/\gamma(\gamma+1)M_\infty^2 \quad (3f)$$

$$T_{sh} = (\tilde{u}_{sh} - \cos\alpha)^2/2 + \{[4\gamma/(\gamma+1)^2]\sin^2\alpha + [2/(\gamma-1)$$
$$-4(\gamma-1)/(\gamma+1)^2]/M_\infty^2 - 4/(\gamma+1)^2 M_\infty^4 \sin^2\alpha\}/2 \quad (3g)$$

The shock angle, α, is related to the shock thickness, n_s, through the relation

$$\frac{dn_s}{ds}/(1+\kappa n_s) = \tan(\alpha-\phi) \quad (4a)$$

The value of n_s itself can be written from mass conservation considerations as

$$(r+n_s\cos\phi)^{1+j} = 2^j \int_0^{n_s} \rho u(r+n\cos\phi)^j \, dn \quad (4b)$$

For reasons explained in Reference [2], the above equations will be normalized according to the following scheme:

$$\eta = n/n_{sh} \qquad \xi = s \qquad \bar{u} = u/u_{sh} \qquad \bar{v} = v/v_{sh}$$

$$\bar{t} = T/T_{sh} \qquad \bar{p} = p/p_{sh} \qquad \bar{\rho} = \rho/\rho_{sh} \qquad \bar{\mu} = \mu/\mu_{sh}$$

(5a-h)

The differential relations needed to transform equations (1a) through (1i) are given by

$$\partial/\partial s = \partial/\partial\xi - \eta(n'_{sh}/n_{sh})\ \partial/\partial\eta$$

$$\partial/\partial n = (1/n_{sh})\ \partial/\partial\eta$$

$$\partial^2/\partial n^2 = (1/n_{sh}^2)\ \partial^2/\partial\eta^2 \tag{6}$$

where,

$$n'_{sh} = (dn_{sh}/d\xi)$$

The s-momentum and energy equations (1b) and (1f) written in the transformed ξ,η plane can be conveniently put in a standardized form for a parabolic equation as,

$$\partial^2 w/\partial\eta^2 + \alpha_1(\partial w/\partial\eta) + \alpha_2 w + \alpha_3 + \alpha_4(\partial w/\partial\xi) = 0 \tag{7}$$

where w represents $\bar{u}$ for the s-momentum equation, and $\bar{t}$ for the energy equation. For the momentum equation the coefficients $\alpha_1 \rightarrow \alpha_4$ are:

$$\alpha_1 = \frac{\rho_{sh}u_{sh}n'_{sh}}{\varepsilon^2\ \mu_{sh}}\ \frac{n_{sh}}{1+\kappa n_{sh}\eta}\ \frac{\bar{\rho}\bar{u}\eta}{\bar{\mu}} - \frac{\rho_{sh}v_{sh}n_{sh}}{\varepsilon^2\mu_{sh}}\ \frac{\bar{\rho}\bar{v}}{\bar{\mu}}$$

$$+ \bar{\mu}_\eta/\bar{\mu} + \frac{\kappa n_{sh}}{1+\kappa n_{sh}\eta} + \frac{\cos\phi n_{sh}}{\gamma+n_{sh}\eta\cos\phi} \tag{8a}$$

$$\alpha_2 = - \frac{\rho_{sh} u_{sh} n'_{sh}}{\varepsilon^2 \mu_{sh}} \frac{n_{sh}}{1+\kappa n_{sh}\eta} \frac{\bar{\rho}\bar{u}}{\bar{\mu}} - \frac{\rho_{sh} v_{sh} n_{sh}}{\varepsilon^2 \mu_{sh}} \frac{\kappa n_{sh}}{1+\kappa n_{sh}\eta} \bar{\rho} \frac{\bar{v}}{\bar{\mu}}$$

$$- \kappa \frac{n_{sh}}{1+\kappa n_{sh}\eta} \bar{\mu}_\eta / \bar{\mu} - \left(\frac{\kappa n_{sh}}{1+\kappa n_{sh}\eta} + \frac{\cos\phi\, n_{sh}}{r + n_{sh}\eta\cos\phi}\right) \times \left(\frac{\kappa n_{sh}}{1+\kappa n_{sh}\eta}\right) \tag{8b}$$

$$\alpha_3 = - \frac{p_{sh} n_{sh}}{\varepsilon^2 \mu_{sh}} \frac{n_{sh}}{1+\kappa n_{sh}} 1/\bar{\mu}\ 1/u_{sh} \left(\bar{p}_\xi - \frac{n'_{sh}}{n_{sh}} \eta \bar{p}_\eta + \frac{p'_{sh}}{p_{sh}} \bar{p}\right) \tag{8c}$$

$$\alpha_4 = - (\rho_{sh} u_{sh} n_{sh} / \varepsilon^2 \mu_{sh}) (n_{sh} / (1+\kappa n_{sh}\eta)) \frac{\bar{\rho}\bar{u}}{\bar{\mu}} \tag{8d}$$

For the energy equation the coefficients are

$$\alpha_1 = \frac{\rho_{sh} u_{sh} n'_{sh} \sigma}{\varepsilon^2 \mu_{sh}} \frac{n_{sh}}{1+\kappa n_{sh}\eta} \frac{\bar{\rho}\bar{u}\eta}{\bar{\mu}} - \frac{\rho_{sh} v_{sh} n_{sh} \sigma}{\varepsilon^2 \mu_{sh}} \times \frac{\bar{\rho}\bar{v}}{\bar{\mu}}$$

$$+ \bar{\mu}_\eta / \bar{\mu} + \frac{\kappa n_{sh}}{1+\kappa n_{sh}\eta} + \frac{\cos\phi\, n_{sh}}{r + n_{sh}\eta\cos\phi} \tag{9a}$$

$$\alpha_2 = -(\rho_{sh} u_{sh} T'_{sh} / \varepsilon^2 \mu_{sh} T_{sh}) (n^2_{sh} / (1+\kappa n_{sh}\eta)) \times \frac{\bar{\rho}\bar{u}}{\bar{\mu}} \tag{9b}$$

$$\alpha_3 = \frac{\rho_{sh} u_{sh} n_{sh} \sigma}{\varepsilon^2 \mu_{sh} T_{sh}} 1/\bar{\mu} \left[\frac{n_{sh}\bar{u}}{1+\kappa n_{sh}\eta} \left(\bar{p}_\xi - \frac{n'_{sh}}{n_{sh}} \eta \bar{p}_\eta + \frac{p'_{sh}}{p_{sh}} \bar{p}\right)\right.$$

$$\left. + \frac{v_{sh}}{u_{sh}} \bar{v}\bar{p}_\eta\right] + \frac{u^2_{sh}\sigma}{T_{sh}} \left(\bar{u}_\eta - \frac{\kappa n_{sh}}{1+\kappa n_{sh}\eta} \bar{u}\right)^2 \tag{9c}$$

$$\alpha_4 = -(\sigma\rho_{sh}u_{sh}n_{sh}/\varepsilon^2\mu_{sh})(n_{sh}'/(1+\kappa n_{sh}\eta))\frac{\bar{\rho}\bar{u}}{\bar{\mu}} \tag{9d}$$

The remaining differential equations are first order and are the continuity equation:

$$[n_{sh}(r+n_{sh}\eta\cos\phi)\ \rho_{sh}u_{sh}\ \bar{\rho}\bar{u}]_\xi + [(r+n_{sh}\eta\cos\phi)\times$$
$$\{(1+\kappa n_{sh}\eta)\rho_{sh}v_{sh}\ \bar{\rho}\bar{v} - n_{sh}'\ \rho_{sh}u_{sh}\ \bar{\rho}\bar{u}\eta\}]_\eta = 0 \tag{10}$$

and the n-momentum equation:

$$\frac{\bar{\rho}\bar{u}}{(1+\kappa n_{sh}\eta)}(\bar{v}_\xi - n_{sh}'/n_{sh}\ \eta\bar{v}_\eta + \frac{v_{sh}'}{v_{sh}}\bar{v}) + \frac{v_{sh}}{u_{sh}}\frac{\bar{\rho}\bar{v}}{n_{sh}}\bar{v}_\eta$$
$$- \frac{\kappa}{1+\kappa n_{sh}\eta}\ \frac{u_{sh}}{v_{sh}}\ \bar{\rho}\bar{u}^2 + \frac{p_{sh}}{\rho_{sh}u_{sh}v_{sh}n_{sh}}\bar{p}_\eta = 0 \tag{11}$$

where with the thin layer approximation this equation becomes,

$$\bar{p}_\eta = [\kappa/(1+\kappa n_{sh}\eta)](\rho_{sh}u_{sh}^2 n_{sh}/p_{sh})\ \bar{\rho}\bar{u}^2 \tag{12}$$

This leaves the equation of state,

$$\bar{p} = \bar{\rho}\bar{t} \tag{13}$$

and the viscosity law,

$$\bar{\mu} = [(T_{sh} + c')/(T_{sh}\bar{t} + c')]\ \bar{t}^{3/2} \tag{14}$$

At the shock location all variables are unity,

$$\bar{u} = \bar{v} = \bar{t} = \bar{p} = \bar{\rho} = \bar{\mu} = 1 \quad \text{at} \quad \eta = 1 \tag{15}$$

An equation of mass conservation can be obtained from equation (10) by integrating from $\eta = 0$ to $\eta = 1$ while holding ξ constant. This results in

$$\frac{dm}{d\xi} = (r+n_{sh}\cos\phi)\,[n'_{sh}\rho_{sh}u_{sh} - (1+\kappa n_{sh})\rho_{sh}v_{sh}] \tag{16a}$$

where

$$m = \int_0^1 n_{sh}(r+n_{sh}\eta\cos\phi)\rho_{sh}u_{sh}\,\bar{\rho}\bar{u}\,d\eta \quad . \tag{16b}$$

is proportional to the rate of mass flux between the body n and shock at a given position on the body surface.

Equations (7), (10)-(14) and (16), constitute the complete set of governing equations for the unknowns $\bar{u}$, $\bar{v}$, $\bar{t}$, $\bar{p}$, $\bar{\mu}$, $\bar{\rho}$ and n_{sh}. These equations are solved along with the surface boundary conditions given by equations (2a) and (2b) and the shock conditions given by equation (15). The mass conservation equation (16a) and (16b) is used to determine the shock stand off distance n_{sh}. The general procedure is to evaluate the rate of mass flux between the body and shock at a given position on the body surface from equation (16b) using the values of the physical quantities previously calculated and then evaluate n_{sh} from equation (16a).

III. JUNCTURE REGION

1. Inviscid Flow Analysis

The aim of the present section is to analyze the nature of the flow at a sphere/cone juncture point in the light of various forms of the gas dynamic equations. It is of interest to analyze the viscous as well as inviscid gas dynamic equations in order to understand the physical behavior at a point in the flow field where a discontinuity in curvature is encountered. In order to be fully consistent in the present analysis, it is desirable to first consider the flow behavior from an inviscid standpoint and then subsequently include viscous effects.

From an inviscid standpoint, it is known from Euler's equations that a streamline curvature discontinuity produces a discontinuity in the flow gradients only along that streamline [8]. For locally supersonic flows, the discontinuity at a juncture point (e.g. sphere/cone tangency point) in the flow gradients will be propagated along the characteristic lines inclined at the local Mach angle to the flow direction. Eventually, therefore, such discontinuities in the flow gradients would propagate everywhere within the flow field downstream of the juncture discontinuity due to successive reflections of these characteristic lines from the shock and body surface.

Note, however, that at the juncture point the flow gradients will be continuous across the shock layer except at the body surface where gradient discontinuities will be present.. Note must also be made here that the flow variables themselves are found to be continuous at the juncture point across the shock layer. However, an added complication arises due to the explicit appearance of the surface curvature in these equations when written in a surface coordinate system. Since the surface curvature itself is discontinuous at the sphere/cone juncture point, it is necessary to assess the influence of the discontinuity on the flow properties and their derivatives. Intuitively it is obvious that a mere coordinate transformation would not affect the physical behavior so that the flow variables themselves are continuous at the juncture point all across the shock layer. This can also be straightforwardly demonstrated through consideration of the integral form of the conservation laws in the surface coordinate system as shown in Appendix (A).

Note that the set of equations (A7 , 14, 20) in Appendix (A) provide for either the trivial case of $p_1 = p_2$ and $u_1 = u_2$, or a shock like discontinuity. Since there is no physical event that could cause a shock at the sphere/cone juncture point, it is fairly evident that the trivial solution is expected for this case indicating that the flow variables are continuous at the sphere/cone juncture point in the surface coordinate system. However the same conclusion does not apply to the flow derivatives with respect to the surface distance. This is evident from the differential form of the two-dimensional inviscid gas dynamic equations as recovered from the integral equations. These are given as,

<u>Continuity</u>

$$(\rho u)_s + (1+\kappa n)(\rho v)_n + \kappa \rho v = 0 \tag{17}$$

<u>s-Momentum</u>

$$p_s + \rho u u_s + (1+\kappa n)\ \rho v u_n + \kappa \rho u v = 0 \tag{18}$$

<u>n-Momentum</u>

$$\rho u v_s /(1+\kappa n) + \rho v v_n - \kappa \rho u^2/(1+\kappa n) + p_n = 0 \tag{19}$$

<u>Energy Equation</u>

$$\rho u T_s + \rho v T_n (1+\kappa n) - u p_s - v p_n (1+\kappa n) = 0 \tag{20}$$

<u>Equation of State</u>

$$p = \left(\frac{\gamma-1}{\gamma}\right) \rho T \tag{21}$$

Using the equation of state and defining $a^2 = (\gamma p/\rho)$, the energy equation can be rewritten as,

$$u p_s + v(1+\kappa n) p_n - u a^2 \rho_s - v(1+\kappa n) a^2 \rho_n = 0 \tag{22}$$

Equations (17) through (22) can now be used on the two sides of the sphere/cone juncture point, noting that the surface curvature, κ, takes a value of 1.0 on the spherical part and a value of 0 on the conical part. Using subscript 1 and 2 for the sphere and cone portions respectively and noting that the flow variables are continuous across the juncture point, the inviscid equations provide the following jump conditions at the juncture point,

$$\rho_{s_1} = (1+n)\ \rho_{s_2} \tag{23a}$$

$$p_{s_1} = (1+n)\ p_{s_2} \tag{23b}$$

$$u_{s_1} = (1+n)\ u_{s_2} - v \tag{23c}$$

$$v_{s_1} = (1+n)\ v_{s_2} + u \tag{23d}$$

In addition to the flow variables themselves, consideration must also be given to the bow shock shape. Note that for problems of this type the shock shape itself would be expected to be smooth through the sphere cone juncture point with the shock slope at any location given in terms of the axial coordinate system (Figure 1) as

$$\tan\alpha = \frac{dR}{dx} \tag{24}$$

Since the shock shape, R, itself is a smooth function through the sphere/cone tangency point, the shock slope, α, would be smooth at this point. However, the first derivative of the shock distance, R, with respect to the surface coordinate system is obtained from the geometrical relation;

$$\frac{dR}{ds} = (1+\kappa n_s) \frac{\sin\alpha}{\cos(\alpha-\phi)} \tag{25}$$

Since the shock angle, α, and the body angle, ϕ, are continuous functions of the surface distance, this relation yields a jump condition for dR/ds at the sphere/cone tangency point as,

$$\left(\frac{dR}{ds}\right)_{sphere} = (1+n_s)\left(\frac{dR}{ds}\right)_{cone} \tag{26}$$

Similar discontinuous behavior can be shown to appear in derivatives such as dn_s/ds and dx_s/ds as shown by the following expressions,

$$\frac{dn_s}{ds} = (1+\kappa n_s) \tan(\alpha-\phi) \tag{27}$$

$$\frac{dx_s}{ds} = (1+\kappa n_s) \frac{\cos\alpha}{\cos(\alpha-\phi)} \tag{28}$$

The corresponding jump conditions, therefore, can be written as,

$$\left(\frac{dn_s}{ds}\right)_{sphere} = (1+n_s)\left(\frac{dn_s}{ds}\right)_{cone} \tag{29}$$

$$\left(\frac{dx_s}{ds}\right)_{sphere} = (1+n_s)\left(\frac{dx_s}{ds}\right)_{cone} \tag{30}$$

It is of interest to note that the expression (24), can be rewritten using the surface-coordinate system as,

$$\frac{dx_s}{ds}\tan\alpha = \frac{dR}{ds}$$

indicating that the discontinuity associated with dx_s/ds and dR/ds at the juncture point are of the nature such that the shock slope, α, itself is continuous at this point.

It must be pointed out here that similar jump conditions can also be established for the second derivatives of the flow quantities mentioned above, whenever required. These derivatives are undefined at the juncture point in this coordinate system, however finite values exist for these quantitites immediately ahead and behind the juncture point. A typical case where higher derivatives are needed is at the shock location. The derivatives of the flow properties behind the shock are shown in Appendix (B). Note that these derivatives (B11, B15-17) undergo discontinuous changes at the sphere/cone juncture point and the magnitude of these discontinuities are related to the surface curvature, κ, and the discontinuity associated with the second derivative of the shock shape, d^2R/ds^2. However it is important to emphasize, here, that this situation is entirely due to the use of the surface coordinate system.

It is, thus, found that within the framework of Euler's equations the flow properties are continuous whereas the flow derivatives with respect to surface distance are discontinuous across the layer at the sphere/cone juncture point. The slope of the shock relative to the surface distance is also found to be discontinuous at this point.

2. Viscous Flow Analysis

(a) Classical Boundary Layer

From a viscous standpoint it seems more rational to first address the question of validity of the various forms of the viscous gas dynamic equations as applied to the spherically blunted cones with a discontinuous surface curvature at the juncture point. The boundary layer version of the Navier-Stokes equations cannot hold at the sphere/cone juncture point becuase the longitudinal derivative of the surface curvature, $\partial\kappa/\partial s >> 1$ [9, 10] and thus the gradient of the corresponding inviscid pressure is discontinuous there. Any such discontinuity would seem to be in violation of the boundary layer scaling laws wherein longitudinal derivatives are assumed to be much smaller than normal derivatives. This becomes more apparent when one considers inclusion of higher order terms in the boundary layer equations. The second order correction (in $Re^{-1/2}$) due to longitudinal curvature is driven by the rate of change of curvature - thus causing this higher order effect to rise up to first order level near a sphere/cone juncture point. It is, therefore, apparent that a new local solution needs to be developed near the juncture point in order to accommodate this anomoly. Such an analysis has been performed by Messiter and Hu [7] for two-dimensional flows.

(b) Triple Deck Analysis

A study of the juncture region has been completed by Messiter and Hu [7] for two-dimensional flow problems. They point out that, unlike the classical boundary layer case, an interaction with the external flow must be taken into account, this occurring through a small pressure change acting over a suitably small distance along the boundary layer. The details of the resulting local pressure distribution cannot be specified in advance, but must be found by studying changes in the boundary layer coupled with small perturbations on the external flow. Messiter's analysis shows that the discontinuity in the pressure gradient predicted by the inviscid flow theory can be removed by using a triple deck formulation and continuous expressions for the pressure gradient can be obtained which are presumed to be correct asymptotic representations as the viscosity coefficient approaches zero. This is achieved by noting that, locally, the most important changes in the profile shape occur in a thin sublayer [11, 12] close to the wall where the changes in the viscous, pressure, and inertia forces are all of the same order as the characteristic Reynolds number tends to infinity. The remainder of the boundary layer experiences primarily a displacement effect because of the small acceleration of the fluid in the sublayer, and the resulting small decrease in the flow deflection angle is nearly constant across most of the boundary layer. The interaction of the boundary layer with the external flow occurs in a streamwise distance, $X = 0(\mathrm{Re}^{-3/8})$, and the sublayer thickness is given by $Y = 0(\mathrm{Re}^{-5/8})$, while the pressure change is found to be of $0(\mathrm{Re}^{-3/8})$. The present sphere/cone problem is more complex due to the axisymmetric nature of the body and the fact that the approaching boundary layer at the juncture point is not that due to a flat plate as it was in Messiter and Hu's analysis. However, an approximate calculation can be performed using their analysis to determine the scale within which the viscous smoothing takes place at the sphere/cone juncture point. This can be done by using the local Reynolds number at the sphere/cone juncture point. Figure (2) shows the results of such an analysis where the surface pressure is shown against the distance in physical coordinates. The asymptotic smoothing of the inviscid pressure gradient at the juncture point is seen to be achieved for this case in a very small physical distance upstream and, comparatively larger, but yet small distance downstream.

(c) Viscous Shock Layer

These results imply two important points. First, near the sphere/cone juncture region the correct asymptotic solution can be obtained provided the viscous set of gas dynamic equations are such that they retain the boundary layer and inertia terms in the viscous region and allow for displacement interaction with the inviscid flow. One way to ensure this criteria is to use the full Navier-Stokes equations. However, the full viscous shock layer equations also seem to be sufficient since they contain all the viscous terms of the triple deck model plus the inertia terms that take into account interaction effects in the inviscid flow.

The second important point is that the interaction effects will be significant in only a very small region of the physical flow and will be difficult to detect for high Reynolds number cases.

However, note that the present choice of the coordinate system would introduce discontinuities in the longitudinal flow gradients at the sphere/cone juncture point, as observed for the inviscid flow. It can be shown also that when viscous effects, as included in the full shock layer equations, are accounted for, the flow variables themselves are continuous through the sphere/cone juncture point (see Appendix A). This can be done by considering the integral form of the viscous equations and by evaluating them for an infinitesimally small element in the surface coordinate system (Appendix A). Note that once again, the set of equations (A7 , 14, 20) give either a trivial solution yielding $p_1 = p_2$ and $u_1 = u_2$ or a shock like jump discontinuity. It is observed that since there is no physical event that could cause a shock at the sphere/cone juncture point, the trivial solution is the only possibility indicating that the flow variables are continuous at the juncture point for the full shock layer equations in the surface coordinate system. However the same conclusion does not apply to the flow derivatives with respect to the surface distance. This is evident from the differential form of the full shock layer equations. These full shock layer equations (1a-h) can be used to determine the jump conditions on the two sides of the sphere/cone juncture point, noting that the surface curvature, κ, takes a value of one on the spherical part and a value of zero on the conical part and also that the flow properties are continuous through this juncture point. This procedure is similar to that adopted for the inviscid set of equations. Note also that the jump conditions associated with the shock shape derivatives would remain the same as those for the inviscid case. Thus a proper physical behavior of the full viscous shock layer equations at the sphere/cone juncture point is summarized as follows:

1. The flow variables are continuous at the sphere/cone juncture point.
2. The use of a surface coordinate system introduces discontinuities in the flow gradients relative to surface distance everywhere across the shock layer at the juncture point.
3. Independent of the choice of the coordinate system, inviscid theory predicts discontinuities in flow gradients only at the surface at the sphere/cone juncture point. However the viscous

flow analysis of Messiter [7] indicates that in the limiting case of very high Reynolds number, this discontinuity would be smoothed out by the sublayer interaction effect within the inner scale length.

4. Within the viscous layer the gradient discontinuities due to the choice of the coordinate system would tend to drop out of the lead order viscous equations as the Reynolds number tends to infinity.

3. Thin Layer Analysis

Many studies [13, 14] in the past have used the thin layer version of the full shock layer equations to predict flow properties within the shock layer region for analytic bodies such as spheres, paraboloids and hyperboloids at high Mach number. The simplifying assumptions inherent in the thin shock layer approximations cause a change in the character of the governing equations and thus of the juncture point analysis presented above. In order to analyze this set of equations, the inviscid form of these equations are first considered here. Attention is first drawn to the characteristics of these equations. In the surface coordinate system the inviscid thin shock layer equations are given by equations (17, 18, 20, 21) and the normal momentum equation is given as

$$p_n(1+\kappa n) - \kappa\rho u^2 = 0 \tag{31}$$

These sets of equations can be shown to be parabolic in nature indicating that the characteristics of the flow are perpendicular to the surface of the body (see Appendix C). This suggests that for the thin shock layer equations information from the body surface is propagated along a line perpendicular to the body surface unlike the full shock layer equations where the characteristic lines are inclined at the local Mach angle of the flow. For this reason it is obvious that any discontinuity in the flow derivatives or otherwise discontinuity at the sphere/cone juncture would be felt all across the shock layer immediately at the juncture point. This physical behavior of the inviscid thin shock layer equations is significantly different from their full shock layer counterpart* and would be expected to manifest itself rather dramatically in the solutions obtained.

In order to further study the behavior of the thin shock layer equations at the sphere/cone juncture point

---

* It can also be shown from the analysis of the characteristics of the full shock layer equations that their characteristics tend to become perpendicular to the body surface in the limit as $\gamma \to 1$, i.e. the thin shock layer approximation is approached (see Appendix C).

attention is now directed to the normal momentum equation given above. This shows that the pressure will be a constant across the shock layer on the conical portion of the body whereas a pressure variation will be encountered on the spherical portion. This can only occur if a jump is allowed in the pressure at the sphere/cone tangency point. It is to be noted here that this need for a discontinuity in pressure is valid for inviscid as well as viscous flows since the normal momentum equation under the thin layer approximations remains unaltered. However it is of interest to note that since under the inviscid thin layer approximation the information is propagated normal to the body surface, the bow shock wave is expected to feel the presence of the sphere/cone juncture point and its attendant pressure discontinuity immediately above the juncture location.

It is, thus, important to the whole structure of the flow that the nature of the thin layer solutions be delineated. To determine the thin layer "jump condition", the approach taken here is to revert to the integral form of the full governing equations, and to assess the influence of the thin layer approximations on the generalized jump conditions so obtained. To do this one must first identify the "thin" layer terms in the general analysis, identify their contributions to the integral formation of the governing equation, and then make the thin layer assumption. In a manner similar to the full shock layer equations, an element of infinitisimally small size is considered in this coordinate system as shown in Appendix (A). The integral form of the momentum equation (A11) when evaluated for this element, is shown to be equation (A12) when no approximation is made. It is now necessary to neglect from this equation those terms which lead to thin shock layer approximations. However, at this point those terms which are neglected in making these assumptions are unknown. Therefore, one must extend this derivation to obtain the differential form of the governing equations and track back those terms which are neglected when thin shock layer assumptions are made. This is achieved in Appendix (A). The equations, so obtained, are given as:

$$p_1 + \rho_1 u_1^2 = p_2 + \rho_2 u_2^2 \qquad (32)$$

$$\rho_1 u_1 = \rho_2 u_2 \qquad (33)$$

$$h_1 + \frac{u_1^2}{2} = h_2 + \frac{u_2^2}{2} \qquad (34)$$

Note that these equations differ from the corresponding equations for the full shock layer form only in the one respect that the v component of velocity does not appear in the present form and thus is unrestricted in its jump behavior across the juncture point.

The equations (32-34) present the set of conditions that must be met at the sphere/cone juncture by all the flow variables in order to accommodate the pressure jump predicted by the normal momentum equation. These equations are quite similar to the "shock discontinuities" of a perfect gas except that they do not contain the v-component of velocity. The fact the v-component of velocity does not appear in these jump conditions is not surprising since a thin shock layer approximation tacitly assumes the v-component of velocity to be small compared to the longitudinal velocity and as such plays a secondary role in the conservation laws.

The admissibility of jump conditions in the flow variables at the juncture point renders the physics of the flow rather complex in this region. At the bow shock, such jump conditions would be expected to produce a discontinuous shock slope at the juncture point. It would appear that the juncture discontinuity propagates normal to the body surface through the local characteristics directly to the bow shock shape.

It is not really clear how the discontinuities in the flow variables themselves are accommodated by the thin shock layer equations. Surely discontinuities in the gradients are to be expected but more confusing is the anticipated behavior of the viscous flow regions. Here it is not clear whether discontinuous solutions can exist since preliminary analysis would seem to indicate that they would be of the subsonic "expansion shock" type. Further study of this point is warranted but shall be deferred from the current effort.

IV. NUMERICAL METHOD

1. General Considerations

Several methods have been presented for solving the "thin" shock layer version of the more general viscous shock layer equations [13, 14]. These approaches have two limitations. First they are based on the assumption that the pressure gradient normal to the body surface is established entirely by centrifugal effects, and second, that the shock wave lies parallel to the body surface. In an attempt to remove these limitations, methods have been developed [2, 15] for addressing the full shock layer equations through a relaxation process wherein the thin shock layer assumptions are removed by an iterative process. While, in general, such methods have been successful, they encounter difficulty whenever the shock layer thickness becomes large. This difficulty usually manifests itself as a divergent behavior in the iteration scheme. In an attempt to overcome this problem, a new relaxation scheme was developed in Reference [16] where an initial solution was relaxed in an artificial time like manner toward the sought after "steady state" solution. In this sense the approach is similar to the relaxation scheme presented by Davis [2], Davis and Nei [17] and Srivastava, Werle and Davis [18], the primary difference being the manner in which the "new guess" on the solution is defined after a given step in the relaxation process. The method so developed has been found to work well for analytic bodies such as paraboloids, hyperboloids and spheres.

Application of this approach to nonanalytic bodies such as spherically blunted cones encounters numerical difficulties at a sphere/cone juncture point where the longitudinal flow derivatives in a surface coordinate system undergo discontinuous changes. The method of solution presented here represents an adaptation of the earlier time like relaxation scheme [16] to problems with imbedded discontinuities in the flow derivatives.

In order to demonstrate the present approach, the s-momentum equation of the viscous shock layer equations is first rewritten in the form

$$\frac{\partial^2 \bar{u}}{\partial \eta^2} + \beta_1 \frac{\partial \bar{u}}{\partial \eta} + \beta_2 \frac{d^2 R}{ds^2} + \beta_3 \frac{dR}{ds} + \beta_4 + \beta_5 \frac{\partial \bar{u}}{\partial \xi} = 0 \qquad (35)$$

where β_1, β_2, β_3, β_4 and β_5 can be obtained from Reference [16] and are given in Appendix (D).

The present time like relaxation scheme utilizes a two step process in which the first step of the method is somewhat similar to an alternating direction implicit method and it yields the flow variables in the shock layer region while the second step is used to update the shock shape itself. This scheme can be demonstrated through the s-momentum equation (7) written in a two step time formulation (see Fig. 3) as,

First Sweep *

$$\frac{\partial^2 \bar{u}^*}{\partial \eta^2} + \beta_1^* \frac{\partial \bar{u}}{\partial \eta} + \beta_2^* \left[\frac{\partial^2 R^n}{\partial s^2} - \frac{\partial R^*}{\partial t}\right] + \beta_3^* \frac{\partial R^n}{\partial s} + \beta_4^* + \beta_5^* \frac{\partial \bar{u}^*}{\partial \xi} = 0 \tag{36a}$$

Second Sweep n+1

$$\beta_2^* \frac{\partial^2 R^{n+1}}{\partial s^2} - \beta_2^* \frac{\partial R^{n+1}}{\partial t} + \beta_3^* \frac{\partial R^{n+1}}{\partial s} + \left[\frac{\partial^2 \bar{u}}{\partial \eta^2} + \beta_1 \frac{\partial \bar{u}}{\partial \eta} + \beta_5 \frac{\partial \bar{u}}{\partial \xi} + \beta_4\right]^* = 0 \tag{36b}$$

Note that the "steady state" version of these equations are precisely the "full" shock layer equations.

The boundary conditions associated with star sweep equation (36a) are typical no slip conditions (2a, 2b) at the surface and Rankine-Hugoniot conditions (3a-3g) at the shock location. However, the boundary conditions associated with the final sweep are the same as those used in Reference [16] and are given as,

i) At $s = 0$ $\quad R = 0$ (37a)

ii) At $s = s_{max}$ $\quad R_{max}^{n+1} = R_{max}^*$ (37b)

There are two points of interest produced by the sphere/cone curvature discontinuity. First note that the coefficients in the s-momentum equation (7) contain the surface curvature, which itself undergoes a discontinuous change from a value of one on the spherical body to a value of zero on the conical body at the sphere/cone juncture point. This then causes the β coefficients of equation (36) to undergo discontinuous changes at the juncture point. As a second point of interest, it is also noted that the derivatives of the shock shape with respect to s, explicitly appear in the governing equations, such as in equation (36). These shock derivatives have been shown to undergo discontinuities at the juncture point in Section III. This jump condition on the first derivative of the shock shape is given by equation (26). A similar jump condition on the second derivative of the shock shape can be obtained through use of the governing differential equations.

As a result then it is seen that the governing equation (36) contains both flow coefficients and shock derivatives which undergo discontinuous change at the juncture point which necessarily produce solutions with discontinuous gradients (see Section III). Such results obviously require modifications in order to obtain numerical solutions of this set of governing equations to assure that finite differencing is not done across such discontinuous regions. Note that the numerical difficulties associated with equation (36a) can be overcome by structuring the finite difference grid system with a point at the juncture thereby avoiding differencing across the discontinuity. However difficulty is still encountered in the second step of the solution process due to the discontinuities that occur in the shock shape derivatives. The occurrence of these discontinuities requires the use of special difference relations at the juncture point. It will be shown here, through a model problem representing the second step of the present numerical scheme that this difficulty can be overcome if the difference form of the differentials are formulated such that they comprehend the juncture jump conditions.

2. A Model Problem

The governing equation for a model problem and its associated boundary and jump conditions are formulated here in order to demonstrate the concepts associated with the viscous shock layer solution for spherically blunted cones.

The governing equation for this model problem is taken to be analogous to the second step of the viscous shock layer scheme and is given as

$$\frac{d^2R}{ds^2} + \alpha_1 \frac{dR}{ds} + \alpha_2 R + \alpha_3 = 0 \tag{38}$$

Note that the coefficients $\alpha_1, \alpha_2, \alpha_3$ in the above equation are to be selected so that this equation models the second step of the viscous shock layer scheme. This would require that either some or all of these coefficients undergo a discontinuous change at the model juncture point in the solution region. For present purposes the coefficients are taken to be one set of constants in the region ahead of the juncture and a different set of constants aft of the juncture. A comparison with the second step of the viscous shock layer scheme shows that the coefficient α_2 does not encounter any jump whereas α_1 and α_3 do encounter jumps in their magnitudes at the sphere/cone tangency point. Thus, the present model problem is set up such that the coefficients $\alpha_1, \alpha_2, \alpha_3$ take constant values (corresponding to the spherical section) in one region and different constant values (corresponding to the conical section) aft of the juncture location.

The boundary conditions to be applied to this model problem are established to closely correspond to the second step of the viscous shock layer solution. These then are given as

$$R = 0 \quad \text{at} \quad s = 0 \tag{39a}$$

$$R = R_\ell \quad \text{at} \quad s = s_{max} \tag{39b}$$

In addition to these, jump conditions analogous to those discussed for the viscous shock layer equations must be established and applied at the juncture point.

At the juncture it is required that the function R, be continuous but that the first derivative, dR/ds, be represented by the relation

$$\left(\frac{dR}{ds}\right)_{-s_{jump}} = K_1 \left(\frac{dR}{ds}\right)_{+s_{jump}} \quad (40)$$

Note that the value of K_1 will be determined here to match the jump actually encountered by the viscous shock layer shock derivative at the sphere/cone tangency point.

From the governing equation (38), it is found that the second derivative, d^2R/ds^2, also undergoes a jump at the juncture point as given by the relation

$$\left(\frac{d^2R}{ds^2}\right)_{-s_{jump}} = \left(\frac{d^2R}{ds^2}\right)_{+s_{jump}} + K_2\left(\frac{dR}{ds}\right)_{+s_{jump}} + K_3 \quad (41a)$$

where

$$K_2 = (\alpha_1)_{+s_{jump}} - K_1(\alpha_1)_{-s_{jump}}$$

$$K_3 = (\alpha_3)_{-s_{jump}} - (\alpha_3)_{+s_{jump}} \quad (41b)$$

The exact solution for the above equations, associated boundary conditions and jump requirements can be found easily and is given as

$$R = A\,e^{ms} + B\,e^{-ns} - \alpha_3/\alpha_2$$

for either region, where

$$m = -\alpha_1/2 + \sqrt{\alpha_1^2/4 - \alpha_2} > 0 \qquad \text{for} \qquad \alpha_2 < 0$$

$$n = \alpha_1/2 + \sqrt{\alpha_1^2/4 - \alpha_2} > 0 \qquad \text{for} \qquad \alpha_2 < 0$$

This gives,

$$R = (\alpha_3/\alpha_2)\ e^{ms} + B[e^{-ns} - e^{ms}] - \alpha_3/\alpha_2 \tag{42}$$

when $\quad 0 \leq s \leq -s_{jump}$

Also,

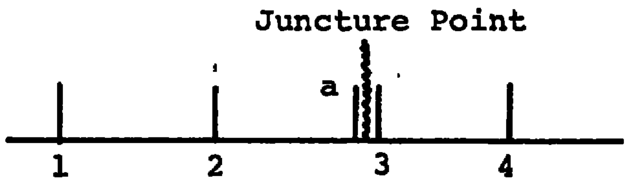
$$R = R_\ell\ e^{D_1(s-s_{max})} - B_1[e^{D_1(s-2s_{max})} - e^{-D_1 s}$$
$$+ \alpha_3/\alpha_2\ [e^{D_1(s-s_{max})} - 1] \tag{43}$$

for $\quad +s_{jump} \leq s \leq s_{max}$

where $D_1 = \sqrt{-\alpha_2}$ for $\alpha_1 = 0$ in this region. Equations (42, 43) contain two undetermined constants B and B_1 which can be determined by using the jump condition (40) and the condition that R be continuous at the juncture point.

3. Model Problem Finite Difference Formulation

The finite difference form of the differentials in equation (38) requires special attention at the juncture point in order to account for the jump conditions associated with the various derivatives. This formulation is shown through the figure below where typical mesh points are shown with a jump occurring immediately ahead of point 3.



Mesh point "a" is located immediately ahead of the juncture point. While formulating the difference form of the derivatives at point "3", a Taylor's series representation which avoids any series expansion across the discontinuity is utilized. For this case, then, a Taylor's series expansion is used from point 3 to 4 and from point "a" to 2. Points "a" and 3 across the discontinuity are related through the jump conditions (40, 41). This procedure yields after proper manipulation (see Appendix E) the following forms of the difference representations of the differentials in equation (38) at the juncture point.

$$\left(\frac{dR}{ds}\right)_3 = \frac{R_4 - R_2}{\Delta\left(1+K_1-\frac{\Delta}{2}K_2\right)} + \frac{\Delta K_3}{2\left(1+K_1-\frac{\Delta}{2}K_2\right)} + 0(\Delta^2) \qquad (44a)$$

$$\left(\frac{d^2R}{ds^2}\right)_3 = \frac{(R_2-R_3)+\left(K_1-\frac{\Delta}{2}K_2\right)(R_4-R_3)}{(\Delta^2/2)\left(1+K_1-\frac{\Delta}{2}K_2\right)} - \frac{2K_3}{\Delta^2\left(1+K_1-\frac{\Delta}{2}K_2\right)} + 0(\Delta) \qquad (44b)$$

Note is made here that these expressions contain the constants associated with the jump conditions (40, 41) and that they reduce to their proper central difference representation for zero strength jump (e.g., K_1=1, K_2=0 and K_3=0). A formally second order accurate representation can similarly be formulated for the second derivative by evaluating the error term of equation (44b) using the governing differential equation at the point where the jump is taking place in the solution regime. The error term in Equation (44b) is given as

$$E = \frac{\Delta}{3}\left[\frac{(K_1 - \frac{\Delta}{2} K_2)\ R_3''' - R_a'''}{(1+K_1 - \frac{\Delta}{2} K_2)}\right] + 0(\Delta^2)$$

For the model problem the first term (order Δ) of the error expression can be evaluated by first differentiating the governing differential equation and then evaluating it on the two sides of the point where the jump in derivatives occurs. This yeilds

$$R_a''' + \alpha_{1a} R_a'' + \alpha_{2a} R_a' = 0 \qquad \text{(45a)}$$

and

$$R_3''' + \alpha_{1_3} R_3'' + \alpha_{2_3} R_3' = 0 \qquad \text{(45b)}$$

Manipulation of equations (45a,b) results in

$$(K_1 - \frac{\Delta}{2} K_2) R_3''' - R_a''' = -(K_1 - \frac{\Delta}{2} K_2)\alpha_{1_3} R_3''$$

$$+ \alpha_{1a} R_a'' + \alpha_{2a} R_a' - (K_1 - \frac{\Delta}{2} K_2)\ \alpha_{2_3} R_3' \qquad \text{(46)}$$

A substitution of this expression in (44b) would result in a second order accurate second derivative at the juncture point. For further details see Appendix (E).

For the sake of identification in the rest of this text, the first of these above difference expressions (44) will be referred to as the "first order accurate scheme", even though only the second derivative at the juncture point experiences a formally first order error. The second of these where the second derivative is also formally second order accurate will be referred to, here, as the "second order accurate" scheme.

Before the numerical scheme outlined above is used in the viscous shock layer equations, a test of this scheme's ability to approximate the exact solution is essential.

4. Model Problem Numerical Results

Figure 4 shows the variable "R" as a function of distance for the model problem. The coefficients α_1, α_2, α_3 in the model equation were chosen to approximately represent the sphere/cone juncture point of a 40° half angle spherically blunted cone. Severe error is seen in the numerical solution when the jump effects are ignored. However, when proper jump effects are accounted for, the exact solution is virtually recovered by the present numerical scheme. Figure 5 shows the first derivative for the same problem. It is noted that the discontinuity predicted by the exact solution is virtually captured exactly by the numerical scheme if proper jump effects are accounted for in the numerical scheme. Figure 6 shows the second derivative, d^2R/ds^2, as a function of surface distance. It is seen also from this figure that large numerical errors are present when the effects of the jump are ignored. However, the exact solution is accurately recovered when proper jump effects are included.

For the sake of completeness, the model problem was also solved using the "second order accurate scheme" of equation (44). It was found that for this case the numerical difference could not be detected to the scale of plot shown in Figures 4 through 6. In order to clarify this point, further studies were undertaken. Figure 7 shows the model problem shock curvature, d^2R/ds^2, at the junction point as a function of the step size "Δs" for the "first order" and "second order" numerical schemes. Note here is made of the fact that the so-called "first order scheme" employs a formally first order accurate representation of the second derivative at only one point in the mesh system, i.e. at the juncture point. It is seen from Figure 7 that the "second order scheme" shows a parabolic behavior as the step size "Δs" is reduced, as one would expect. However, the "first order scheme" does not show a linear dependence on Δs but rather a parabolic behavior. To further detail these results, Figure 8 replots these curves against the square of the step size "Δs". This figure clearly indicates that both the "first" and "second" order schemes yield results that approach the exact solution as though they were second order accurate. The explanation of these results is given in detail in Appendix (F) where it is verified that it is basically a manifestation of the fact that a local truncation error of order Δs at a finite number of points in a finite difference mesh does not necessarily produce a first order global error.

It is, therefore, established that the so-called "first order accurate scheme" is essentially second order accurate. For this reason it was found unnecessary to use the "second order accurate scheme" for present purposes.

5. Application to the Full Viscous Shock Layer Equations

The finite difference expressions so developed can now be applied to the solution of the full viscous shock layer equations for hypersonic flow past spherically blunted cones. A detailed description of the method for evaluating the jump conditions associated with the shock wave derivatives at the sphere/cone juncture point for the full viscous shock layer equations is presented in Appendix (G). It is shown here, that the proper jump condition associated with the first derivative, dR/ds, at the juncture point can be estimated by purely geometrical considerations and the fact that the shock wave shape, R, itself is smooth at this point. However, the jump condition associated with the second derivative, d^2R/ds^2, must be obtained by using the differential equation itself on the two sides of the sphere/cone tangency point in a manner similar to that adopted for the model problem discussed earlier. This is shown further in Appendix (G). It is shown in this appendix that care must be used while evaluating the jump conditions in the viscous shock layer code. For present purposes, the jump conditions were evaluated at the first mesh point away from the wall.

6. Overall Method of Solution

The overall method of solution for the full viscous shock layer equations is as follows. An initial guess was first made on the shock shape. Based on this guess the first and second derivatives of the shock distance were computed using central differences at points away from the sphere/cone tangency point. However, since jump conditions on the shock derivatives are not known initially at the sphere/cone juncture point, a second order accurate three point backward difference schemes was used on the spherical part and a three point forward difference scheme was used on the conical part for the first derivative of the shock stand off distance in order to avoid any differencing across the juncture point. In a like manner, four point second order accurate schemes were used for the second derivative of the shock stand off distance at the juncture point. The star sweep equations were then solved by starting at the stagnation point, where both $\partial\bar{u}/\partial\xi$ and $\partial\bar{t}/\partial\xi$ vanish, thus reducing the governing equations to ordinary differential equations. The first equation solved was the energy equation so that thereafter all quantities such as viscosity related to temperature could be evaluated. Next, the s-momentum equation was integrated to determine a $\bar{u}$-velocity profile, and then the continuity equation was solved to determine first the shock stand-off distance from equation (10) and then the $\bar{v}$-component of the velocity from equation (16). Finally equation (11) was integrated to determine the local pressure level. The coefficients in the governing equations were then reevaluated using the new flow variables. Repetition of the above steps at a given station continued until the solution converged. The method then stepped along the body surface and iterated at each station to achieve converged solutions. To accelerate the convergence process, the previous station values of the profiles were used at each new step as a first guess. One difficulty encountered during this iteration scheme was the presence of an oscillatory behavior of the normal velocity component, $\bar{v}$, at some station in the s-direction [Ref. 16]. This oscillatory behavior of the physical quantities was overcome by an under-relaxation scheme as shown:

$$w = F_1 w_1 + (1-F_1)w_2$$

where w_1 is the most recently calculated physical quantity and w_2 is the value obtained from the previous calculated value of this quantity. It was observed that a value of F_1 of 0.2 to 0.4 produced convergence in most cases considered. In general it was also found that such an under-relaxation technique was needed only for the pressure and v-component of velocity.

Once the above method had passed over the entire mesh the second sweep equations were invoked. The final sweep equation (36b) was then solved using the two boundary conditions of equations (37a,b). No iteration of the final sweep equation was required since it is linear. However note that the final sweep equation requires the necessary jump conditions associated with the first derivative of the shock stand off distance, dR/ds and also that associated with the second derivative, d^2R/ds^2. These jump conditions were evaluated using the flow properties obtained in the star sweep calculations. The final sweep equation was required since it is linear. However note that the final sweep equation requires the necessary jump conditions associated with the first derivative of the shock stand off distance, dR/ds and also that associated with the second derivative, d^2R/ds^2. These jump conditions were evaluated using the flow properties obtained in the star sweep calculations. The shock shape obtained from the final sweep was used then to solve the next star sweep in time. The procedure continued in time until two alternate final sweeps converged to a desired degree of accuracy. Appendix (H) discusses further details of the computer program used to obtain the present numerical results.

7. Grid Sizes for Shock Layer Solution

The following normal step sizes distributions were used in the finite difference solution of the full viscous shock layer equations for the cases presented in the following section.

$Re_\infty = 1.515 \times 10^3$	
η_{range}	$\Delta\eta$
0.0 - 0.050	0.001
0.05 - 0.65	0.015
0.65 - 1.0	0.035

$Re_\infty = 3 \times 10^5$	
η_{range}	$\Delta\eta$
0.0 - 0.005	0.0001
0.005 - 0.50	0.005
0.50 - 1.0	0.0099

V. RESULTS AND DISCUSSION

The general analysis and the numerical techniques discussed earlier were used to obtain the solutions of the full viscous shock layer equations for hypersonic flow past various spherically blunted cones in order to test the reliability of this technique. Since the interest in the present study was centered on the sphere/cone tangency region, numerical solutions were generated only to about 2-3 nose radii downstream of the stagnation point for a range of large as well as small cone angles of the spherically blunted cones. Numerical solutions were obtained for a wide range of cone half angles from 30° to 0° at various test conditions corresponding to available data and other calculations.

Figure 9 shows the surface pressure distribution obtained here for a 30° half cone angle spherically blunted cone at a free stream Mach number, $M_\infty = 10$, free stream Reynolds number, $Re_\infty = 3 \times 10^5$ and a wall to stagnation temperature ratio, $T_w/T_o = 0.05$. These test conditions were chosen in order to compare the predicted numerical results with the inviscid solutions of Inouye et al. [8] for the same body. The present calculations for this case were made using a variable normal step size, $\Delta\eta$, which ensured at least 10-15 mesh points within the boundary layer regime while the longitudinal step size, Δs, was selected such that a mesh point of the numerical scheme coincided with the sphere/cone juncture point. The time step size, Δt, was taken to be 3.5.

Figure 9 shows that when the shock jump conditions are accounted for in the finite difference formulation as shown earlier, the predicted surface pressure compares well with the inviscid solution. It is of interest to note here that the discontinuity in the surface pressure gradient at the sphere/cone juncture point predicted by the inviscid theory is virtually reproduced by the present viscous model for this very high Reynolds number case. However, in the light of the analysis of Messiter and Hu [7] for a simple two-dimensional flow with a curvature discontinuity, one might have anticipated a viscous smoothing of the discontinuity in the surface pressure gradient at the juncture point. It is now clear from the discussion of Section III that the viscous smoothing for this problem is of a very mild nature and occurs over such a short distance that it is not seen to the scale of the present calculations.

Figure 9 also presents two other numerical results for the same test conditions. One of these is the case where a numerical solution of the full viscous shock layer equations was obtained by ignoring the relevant jump conditions associated with the shock derivatives at the juncture point and by adjusting the mesh of the numerical scheme such that the juncture point lies between two mesh points. This case then allows an assessment of the numerical errors that are introduced in the computational method when one ignores the relevant jump effects associated with the juncture point. Note that the numerical errors are large in the juncture region and tend to diminish away from the juncture region. The second case shown in Figure 9 is similar to the first except that the mesh system was aligned so that a mesh point coincided with the sphere/cone tangency point. This case allows an assessment of the importance of the shock jump conditions on the surface properties. Apparently these are of a dominant nature in this region of the flow.

Figure 10 shows the surface heat transfer distribution for this case under identical flow conditions. It is observed that the erratic behavior of the computational results persist when the jump effects are ignored whereas the inclusion of these effects tend to eliminate this erratic behavior completely. From equation (26) it is seen that the jump conditions associated with the shock derivatives tend to increase in magnitude as the cone angle for the spherically blunted cones is reduced. It is, therefore, pertinent to test this computational technique for lower cone angles in order to establish the generality of this scheme. Further numerical solutions were obtained for lower cone angles ranging from 20° to 0° in order to assess the influence of the jump effects on the surface properties. Difficulties were encountered here while attempting to reduce the cone angle mainly due to the choice of the initial shock shape for such bodies. This was overcome here by reducing the cone angle in increments of about 5° with the number of mesh points between the juncture and stagnation point kept fixed. This resulted in an increase in longitudinal step size, Δs, as the cone angle was reduced but this technique was found to work well for all cases that are presented here. It should also be noted that care was exercised to at least include 10-15 points within the boundary layer while selecting the normal step size, $\Delta\eta$. Figure 11 through 15 present the results for such a calculation for cone angles ranging from 20° to 0°. It is seen from Figures 11 and 13, which show the surface pressure distributions, that the results of such a calculation compare well with the inviscid solution, when the proper jump conditions associated with the shock derivatives are included in the solution scheme. These figures also show the case when such jump effects are ignored in the calculation, thereby causing rather erratic behavior. Figures 12 and 14 show similar results for the surface heat transfer rates. Note that Figure 15 for a 0° cone angle (spherical-cylinder body) does not contain the numerical results corresponding to the no jump case. This is because the errors were of such large magnitude at the juncture point that a properly converged solution could not be obtained.

Thus far it has been shown that the present computational approach yields good numerical solutions in comparison with the inviscid theory [8]. Further comparisons are now presented with the available experimental data for spherically blunted cones. Figure 16 presents surface pressure for 7.5° half angle spherically blunted cone at free stream Mach number, $M_\infty = 13.41$, free stream Reynolds number, $Re_\infty = 1515$, wall to stagnation temperature ratio, $T_w/T_o = 0.0741$ and a free stream temperature, $T_\infty = 200°R$ corresponding to the experimental data of Pappas and Lee [19]. Significant differences from that of the inviscid results are noticed at the sphere/cone juncture point for this case. The inviscid pressure distributions shown earlier through Figures 11-13 predict a discontinuity in the pressure gradient at the sphere/cone juncture point. However for the present low Reynolds number of 1515, viscous effects smooth the discontinuity completely. The result of the present calculations are seen to compare well with the data shown in Figure 16.

Note also that similar numerical calculations which do not include the proper jump effects at the juncture point are seen to yield seriously erroneous results. Figure 17 presents the ratio of the wall to the stagnation point heat transfer for the same test case. The comparison of the present calculations with experimental data when the jump effects are included is seen to be excellent. Again large errors are observed when these jump effects are ignored. Figures 16 and 17 also show the numerical results obtained by Miner and Lewis [5] for the same body (with an artificially smoothed juncture point) under identical test conditions. It is seen from these figures that the present results tend to show better agreement with the experimental data when shock jump effects are directly accounted for at the sphere/cone juncture point.

VI. CONCLUSIONS AND RECOMMENDATIONS

An analysis of the physical flow behavior at the sphere/cone tangency point has been made. This analysis indicated that, independent of the choice of the coordinate system, inviscid theory would predict a discontinuity in the flow gradients only at the surface at the sphere/cone juncture point. However, following the analysis of Messiter [7], it was found that in the limiting case of very high Reynolds number this discontinuity would be smoothed out by the sublayer interaction effect within the inner scale length. It has also been shown that the use of a surface coordinate system introduces discontinuities in the flow gradients relative to surface distances everywhere across the shock layer and at the body surface at the juncture point. Within the viscous layer this gradient discontinuity due to the coordinate system would tend to vanish to lead order as the Reynolds number tends to infinity. Analytical jump conditions were developed at the sphere/cone juncture point for these discontinuous flow gradients associated with the choice of surface coordinate system. Finite difference formulations were then developed that account for these embedded gradient discontinuities in order to eliminate numerical difficulties in the solution of the full viscous shock layer equations. Such solutions were obtained by a numerical scheme which utilizes a time dependent relaxation technique for the bow shock wave shape. Comparisons of the present results with inviscid solutions at high Reynolds numbers and experimental data at intermediate ones were found to be good.

While the present technique for treating the sphere/cone juncture region has been shown to yield good results, the present numerical scheme which is essentially a time dependent relaxation technique encountered certain difficulties worth noting. One difficulty that does arise is the oscillatory behavior observed in the iterative solution of the shock layer equation at some point downstream on the surface. While an under-relaxation scheme was found to effectively remove this problem for most flow conditions of interest, it is recommended that in future studies the continuity and normal momentum equations be coupled during the iteration process. Another difficulty that was found for this relaxation scheme was the initialization process used for the bow shock shape. While this technique enjoys a greater degree of flexibility as compared to previous techniques, non-the-less difficulties were encountered for the low cone angle cases of the spherically blunted cone studies. Future studies should consider use of the inviscid bow shock shape as the natural initial shape for this relaxation technique.

REFERENCES

1. Jain, A.C. and Adimurthy, V., "Hypersonic Merged Stagnation Shock Layers, Part II, Cold Wall Case," AIAA Journal, Vol. 12, No. 3, March 1974, pp. 348-354.
2. Davis, R.T., "Numerical Solution of the Hypersonic Viscous Shock Layer Equations," AIAA Journal, Vol. 8, No. 5, May 1970, pp. 843-851.
3. Moeckel, W.E., "Some Effects of Bluntness on Boundary Layer Transition and Heat Transfer at Supersonic Speeds," NACA Report 1312, 1957.
4. Rogers, Ruth H., "Boundary Layer Development in Supersonic Shear Flow," AGARD Report 269, 1960.
5. Miner, E.W. and Lewis, C.H., "Hypersonic Ionizing Air Viscous Shock Layer Flows Over Sphere Cones," AIAA Journal, January 1975, pp. 80-88.
6. Kang, S.W. and Dunn, M.G., "Hypersonic Viscous Shock Layer With Chemical Non-Equilibrium for Spherically Blunted Cones," AIAA Journal, Vol. 10, No. 10, October 1972, pp. 1361-1362.
7. Messiter, A.F. and Hu, J.J., "Laminar Boundary Layer at a Discontinuity in Wall Curvature," Qr. of Appl. Maths., July 1975, pp. 175-181.

8. Inouye, M., Rakich, J. and Lomax, H., "A Description of Numerical Methods and Computer Programs for Two-Dimensional and Axi-Symmetric Supersonic Flow Over Blunt Nosed and Flared Bodies," TND-2970, August 1965, NASA.

9. Neyland, V.Ya., and Sychev, V.V., "Asymptotic Solutions of Navier Stokes Equations in Regions with Large Local Perturbations," AN SSSR, IZVESTIYA, Meknanika Zhidkosti i Gaza, No. 4, 1966, pp. 43-49, English translation by Translation Division, Foreign Technology Division, WP-AFB, Ohio, No. FTD-HT-23-339-68.

10. Brailovskaya, I.Yu., "Flow of a Viscous Gas Near a Wall With a Corner," Soviet Physics - DOKLADY, Vol. 12, No. 9, March 1968, pp. 845-847.

11. Stewartson, K., "On the Boundary Layer Near Corners," Q. J. Mech. Appl. Math. 23, 1970, pp. 137-152. Corrections and an addition, 24, pp. 387-389, 1971.

12. Stewartson, K., "On the Flow Near the Trailing Edge of a Flat Plate," II, Mathematika 16, 106-121.

13. Dellinger, T.C., "Nonequilibrium Air Ionization in Hypersonic Fully Viscous Shock Layers," AIAA Paper No. 70-806, June 1970.

14. Cheng, H.K., "The Blunt-Body Problem in Hypersonic Flow at Low Reynolds Number," Report AF-1285-A-10, 1963, Cornell Aeronautical Laboratory.

15. Srivastava, B.N., Werle, M.J. and Davis, R.T., "Solution of the Hypersonic Viscous Shock Layer Equations for Flow Past a Paraboloid," Report No. AFL-74-4-10, Department of Aerospace Engineering, University of Cincinnati, April 1974. Also appeared in AIAA Journal, Vol. 14, No. 2, February 1976, pp. 257-259.

16. Werle, M.J., Srivastava, B.N. and Davis, R.T., "Numerical Solutions to the Full Viscous Shock Layer Equations Using an ADI Technique, " Report No. AFL 74-7-13, Department of Aerospace Engineering, University of Cincinnati, August 1974.

17. Davis, R.T. and Nei, Y.W., "Numerical Solution of the Viscous Shock Layer Equations for Flow Past Spheres and Paraboloids," Final Report, Sandia Corporation, Contract No. 48-9195.

18. Srivastava, B.N., Werle, M.J. and Davis, R.T., "Stagnation Region Solutions of the Full Viscous Shock Layer Equations," AIAA Journal, Vol. 14, No. 2, February 1976, pp. 274-276. Also Report No. AEDC-TR-76-53, August 1976, Arnold Engineering Development Center, AAFS, Tennessee 27289, Presented by B.N. Srivastava at Eleventh Southeastern Seminar on Thermal Sciences, University of Tennessee, April 28-29, 1975.

19. Pappas, C.C. and Lee, G., "Heat Transfer and Pressure on a Hypersonic Blunt Cone With Mass Addition," AIAA Journal, Vol. 8, No. 5, May 1970, pp. 954-956.

20. Davis, R.T. and Flügge-Lotz, I., "Second-Order Boundary-Layer Effects in Hypersonic Flow Past Axisymmetric Blunt Bodies," J. Fluid Mech., Vol. 20, Part 4, 1964, pp. 593-623.

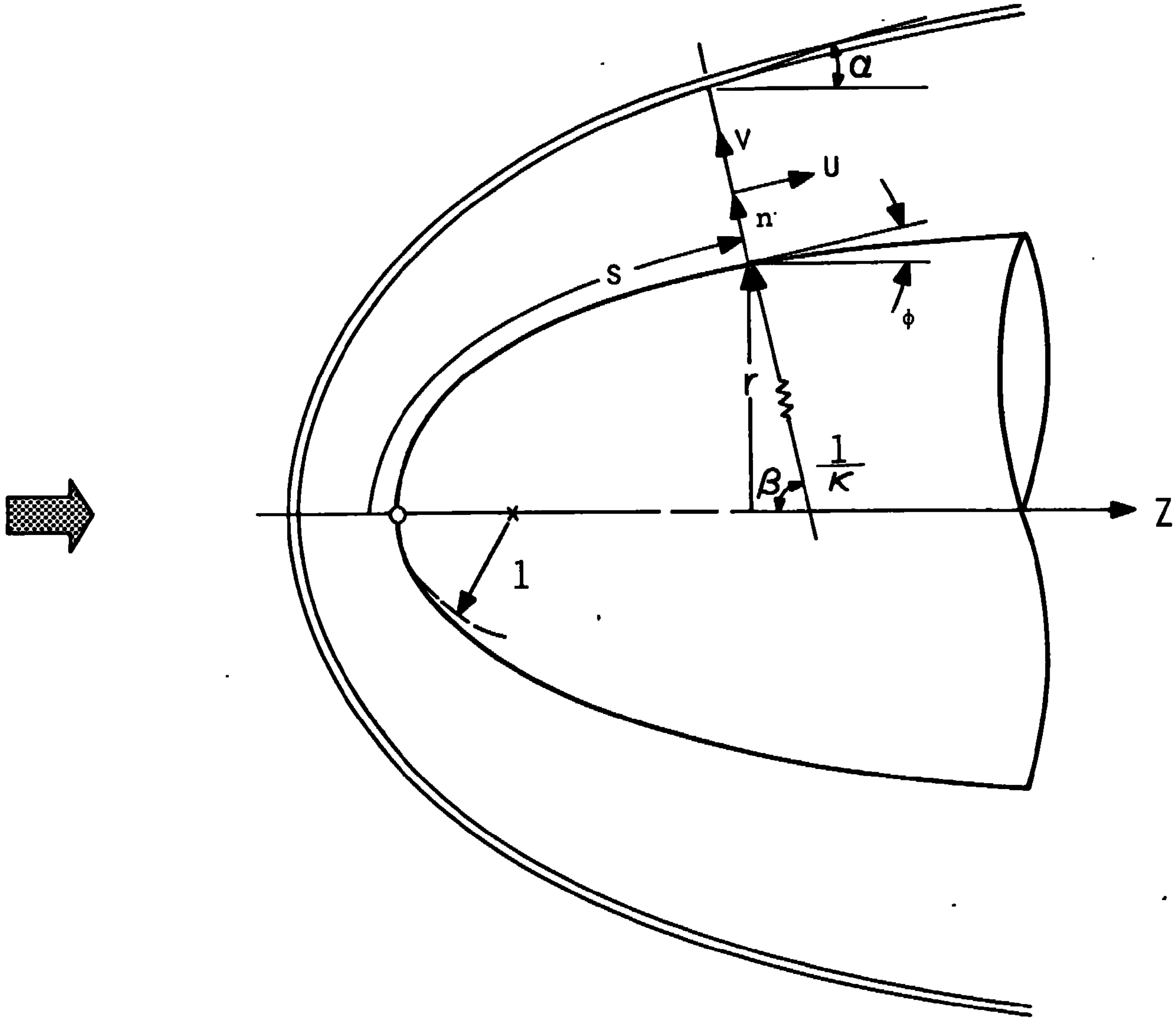


FIGURE 1. COORDINATE SYSTEM

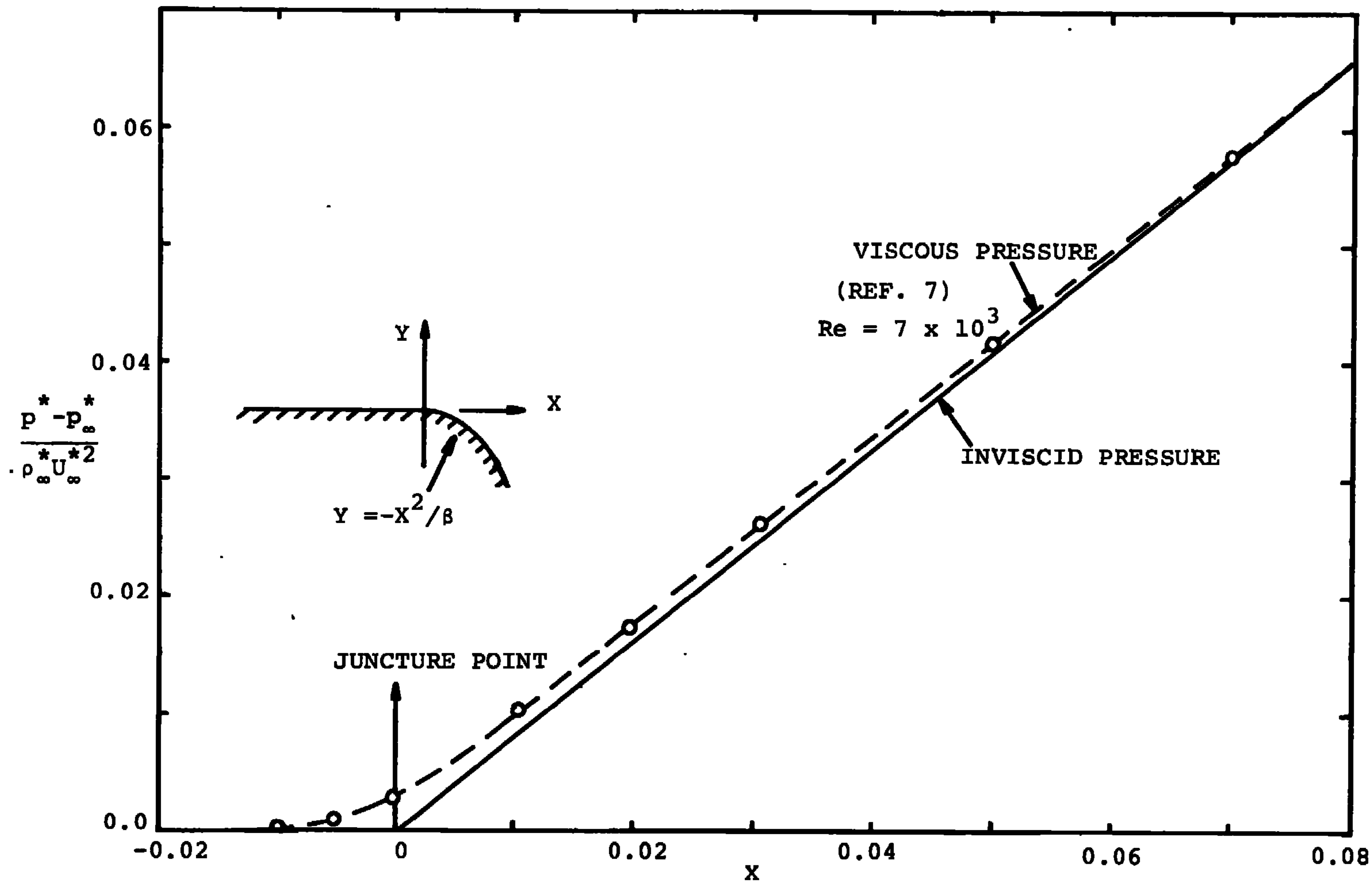


FIGURE 2 SUBLAYER INTERACTION PRESSURE DISTRIBUTION

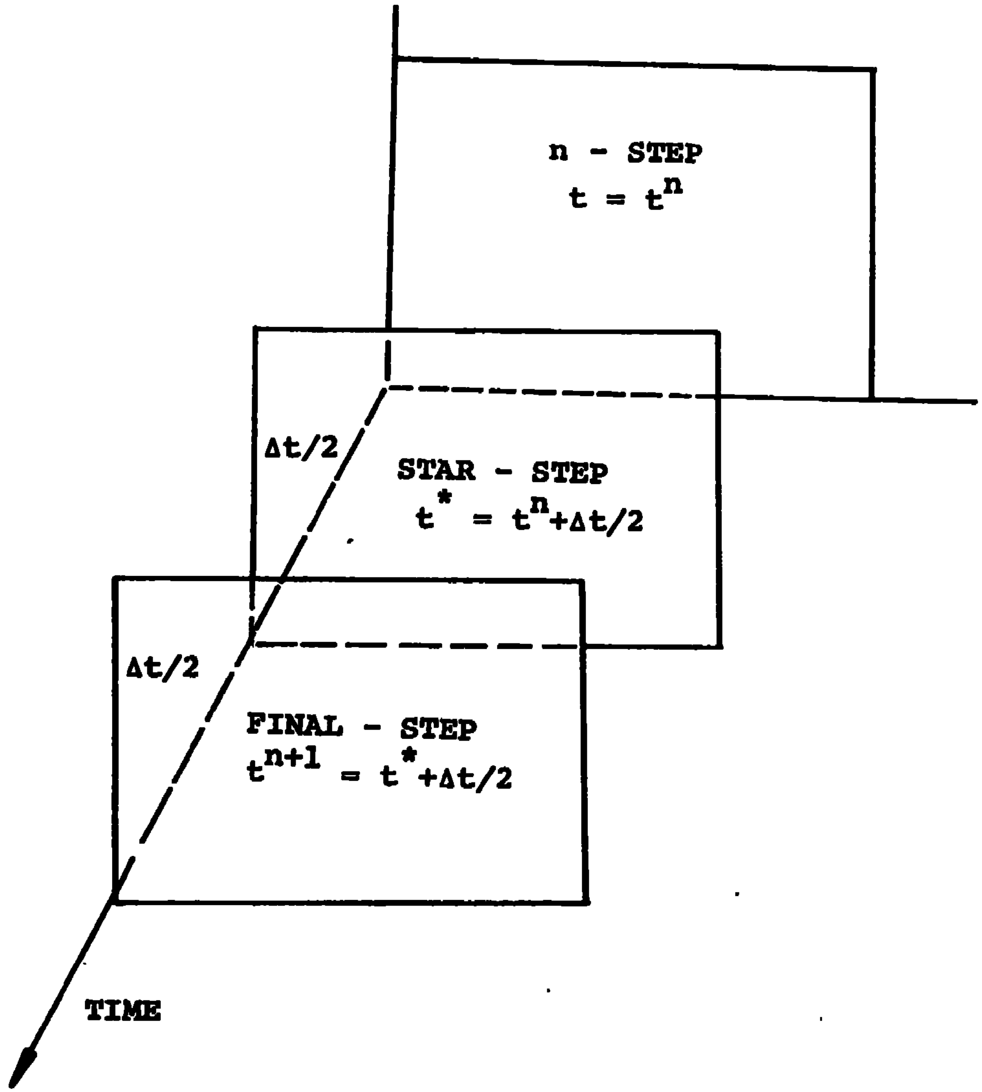


FIGURE 3. ADI NUMERICAL SCHEME

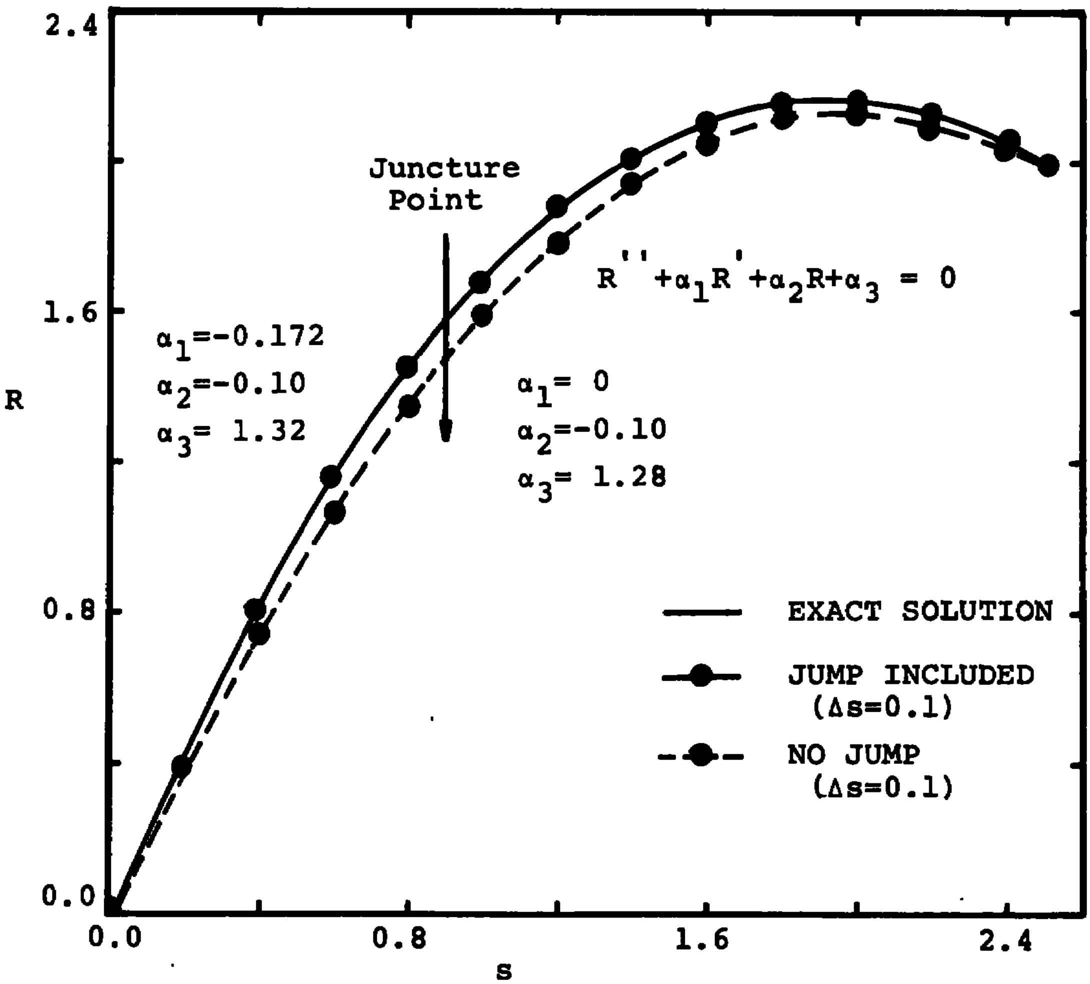


FIGURE 4 MODEL SHOCK SHAPE

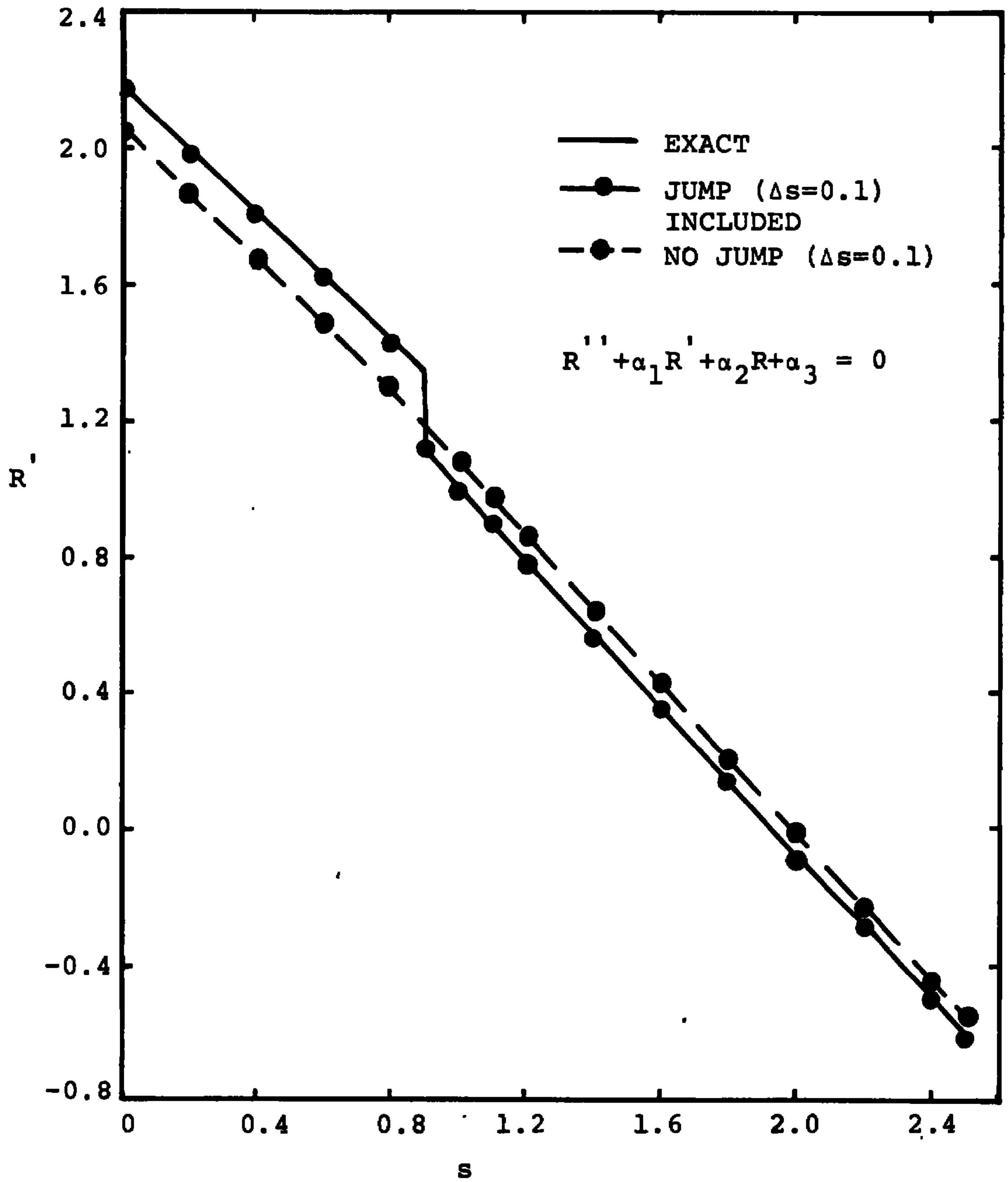


FIGURE 5 MODEL SHOCK SLOPE

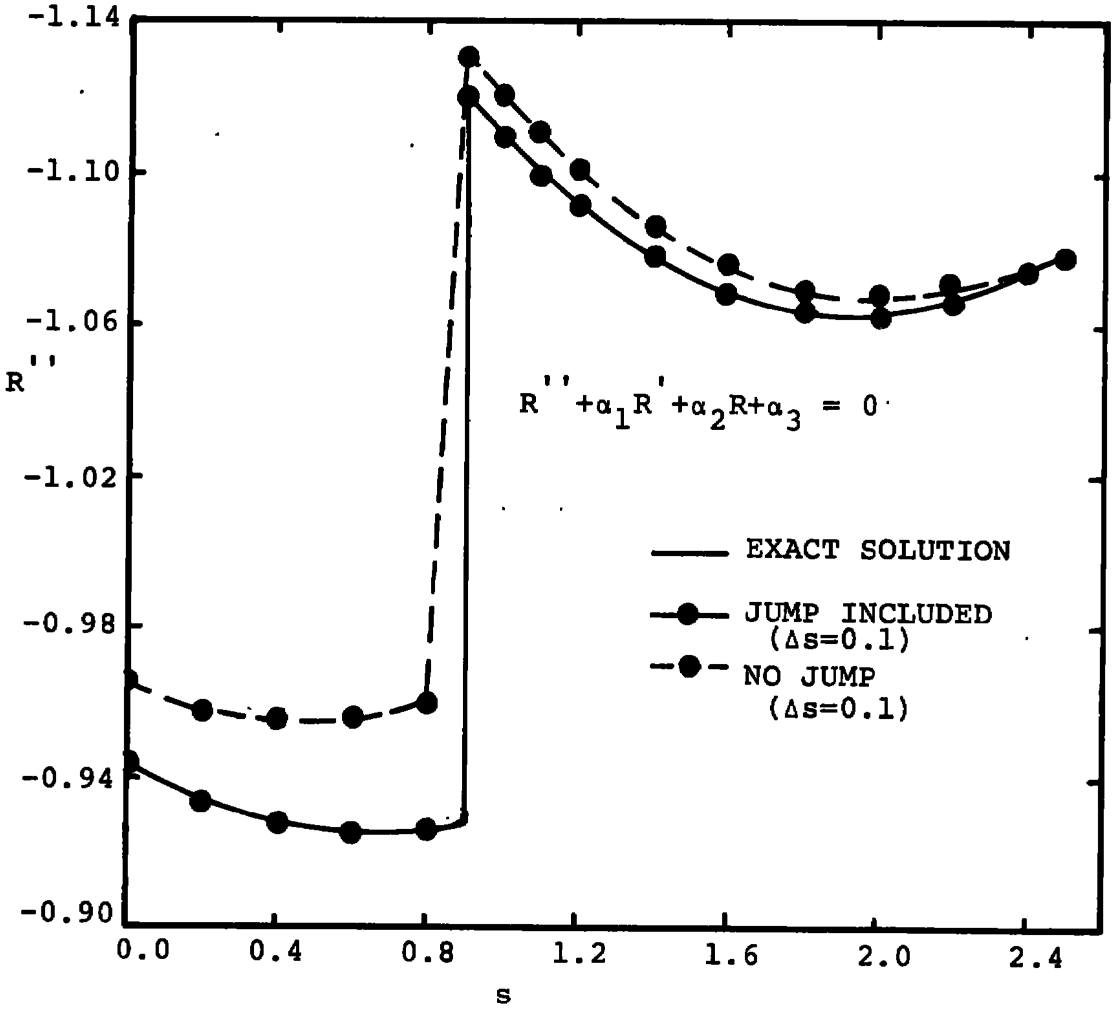


FIGURE 6 MODEL SHOCK CURVATURE

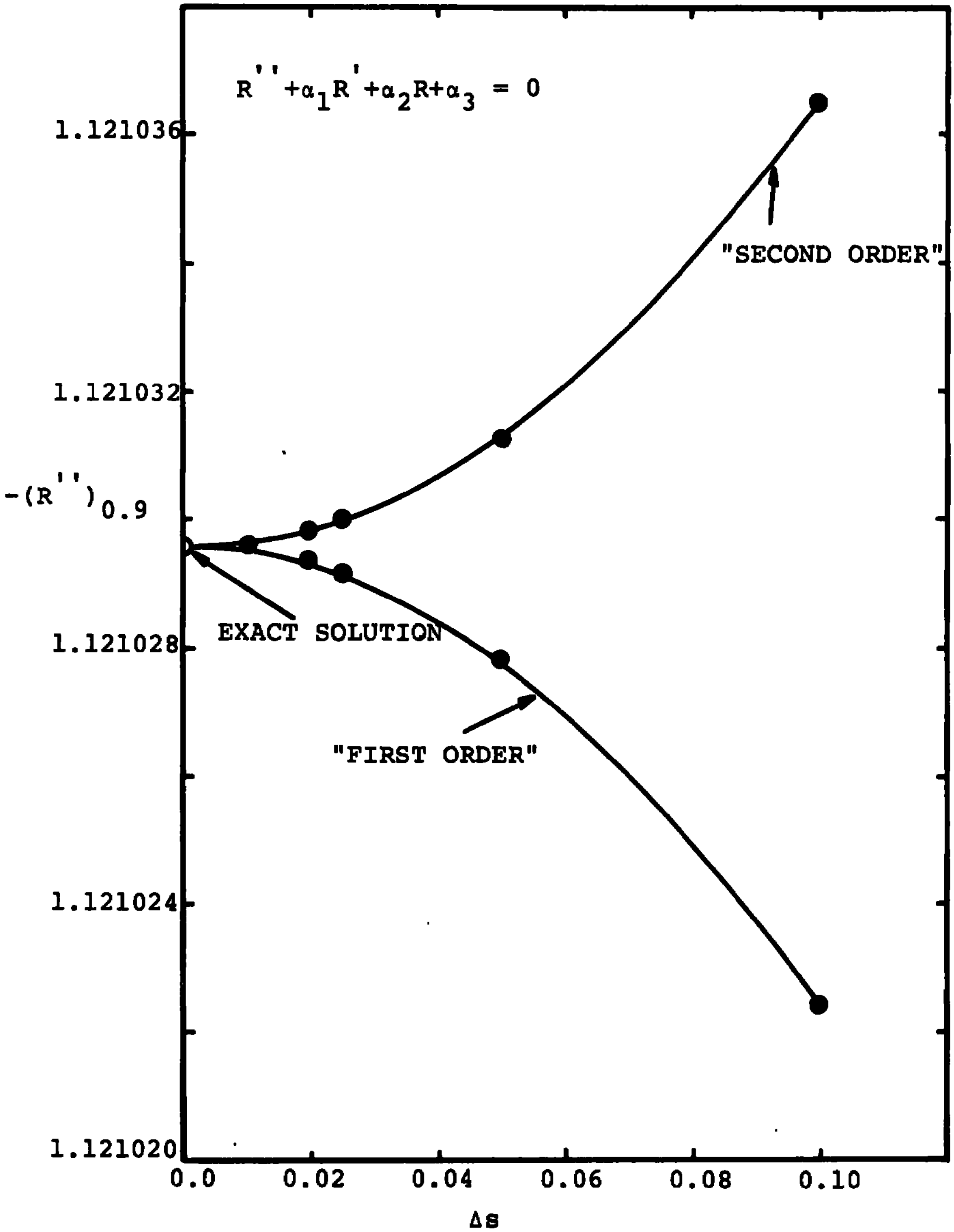


FIGURE 7 MODEL STEP SIZE STUDY

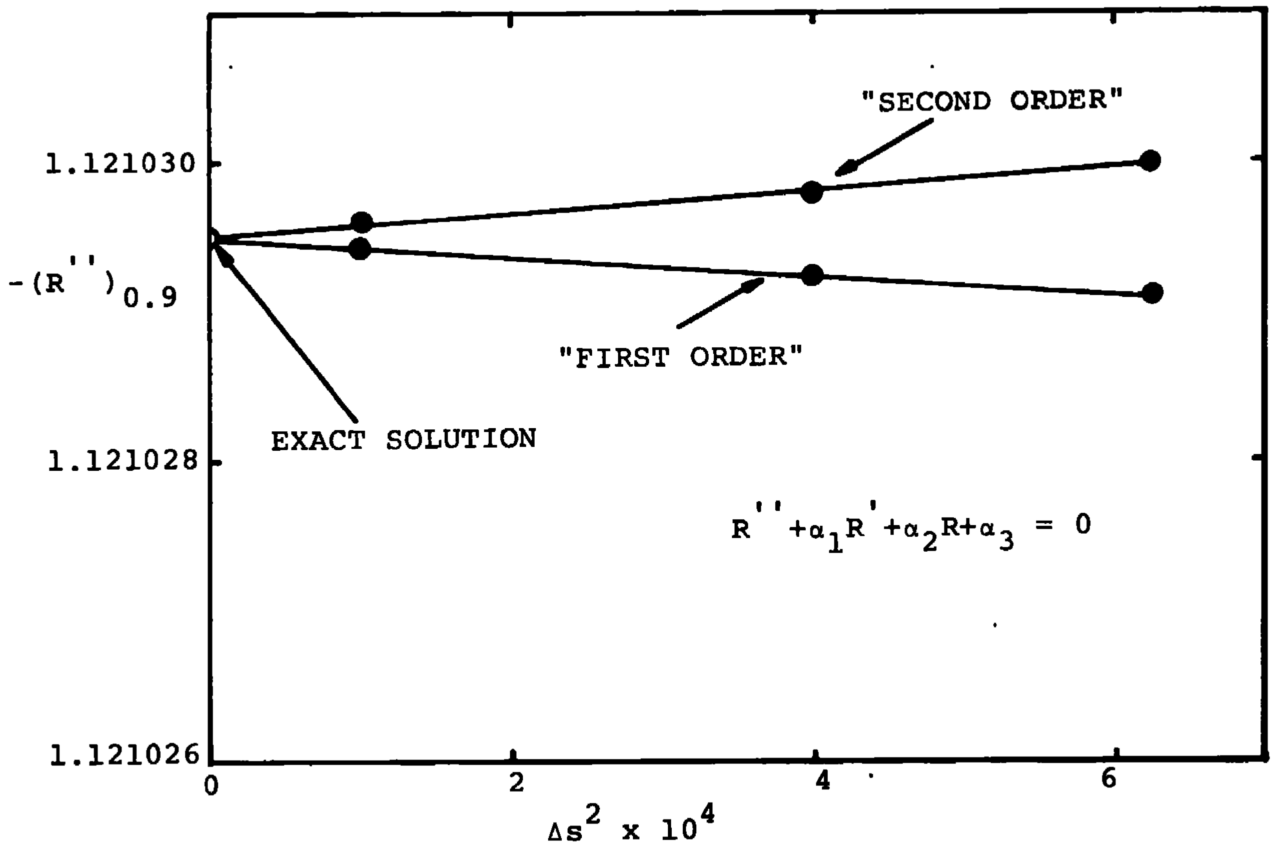


FIGURE 8 MODEL STEP SIZE STUDY

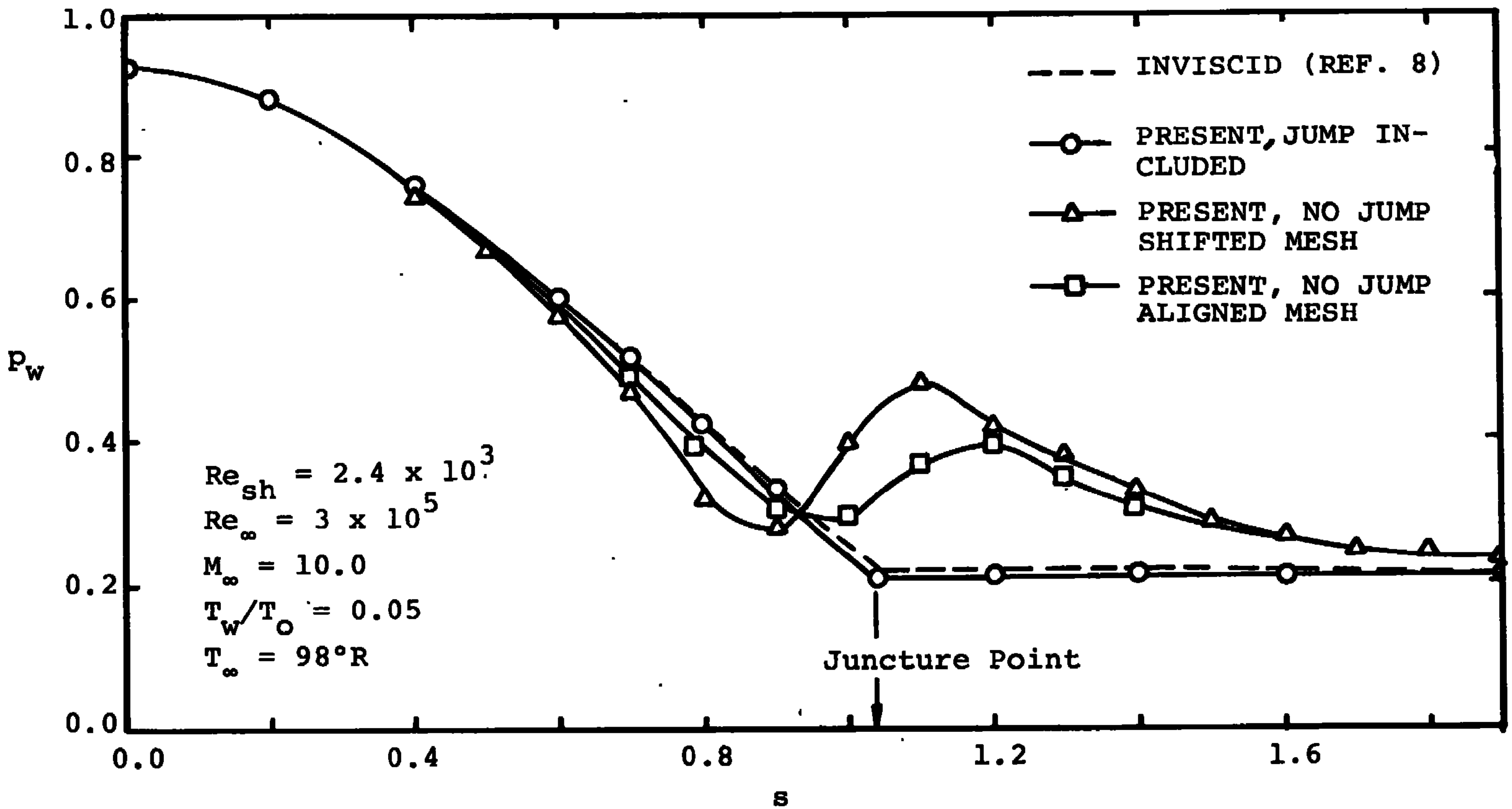


FIGURE 9 SURFACE PRESSURE DISTRIBUTION FOR A 30° HALF ANGLE SPHERE-CONE

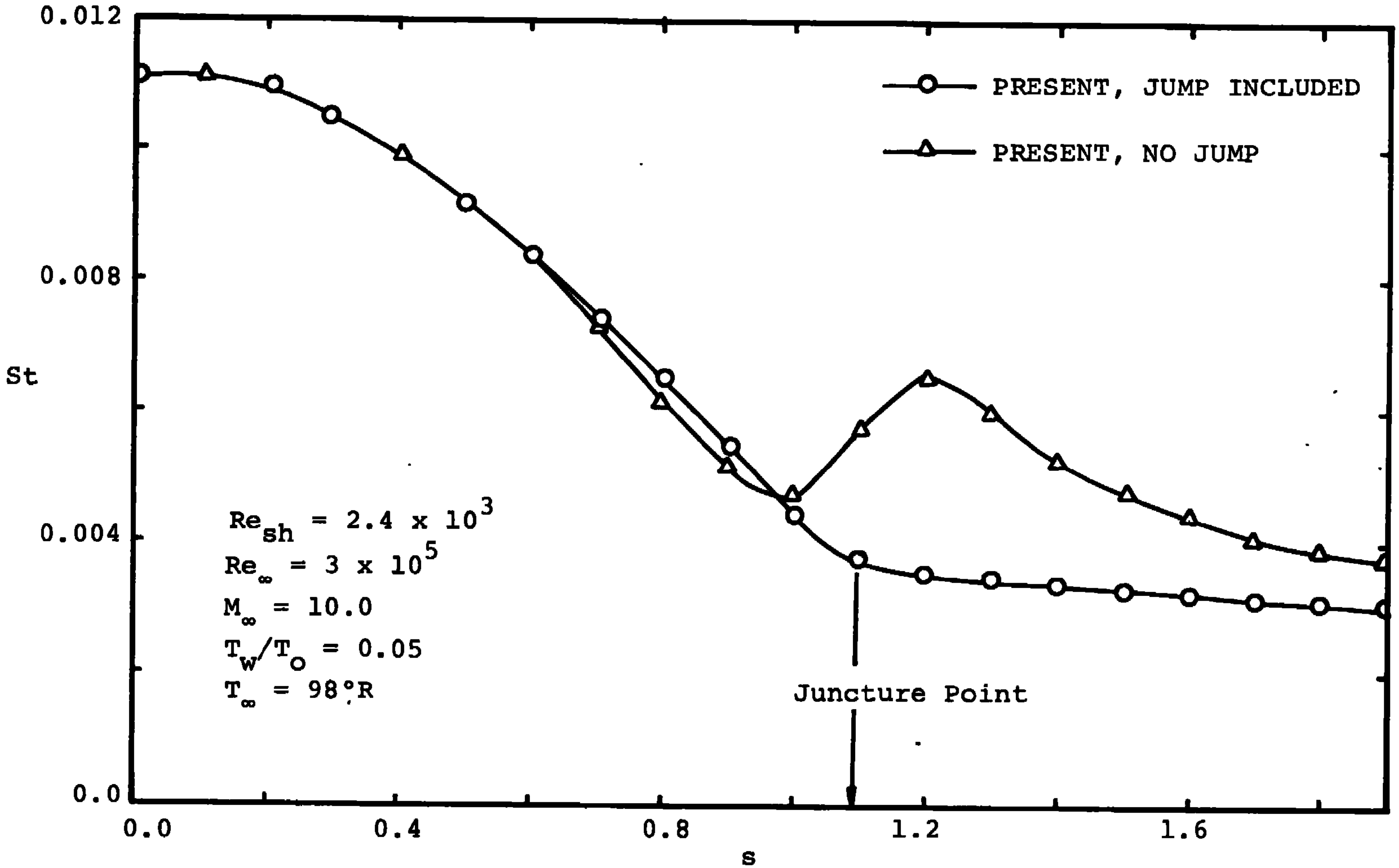


FIGURE 10 HEAT TRANSFER DISTRIBUTION FOR A 30° SPHERE-CONE

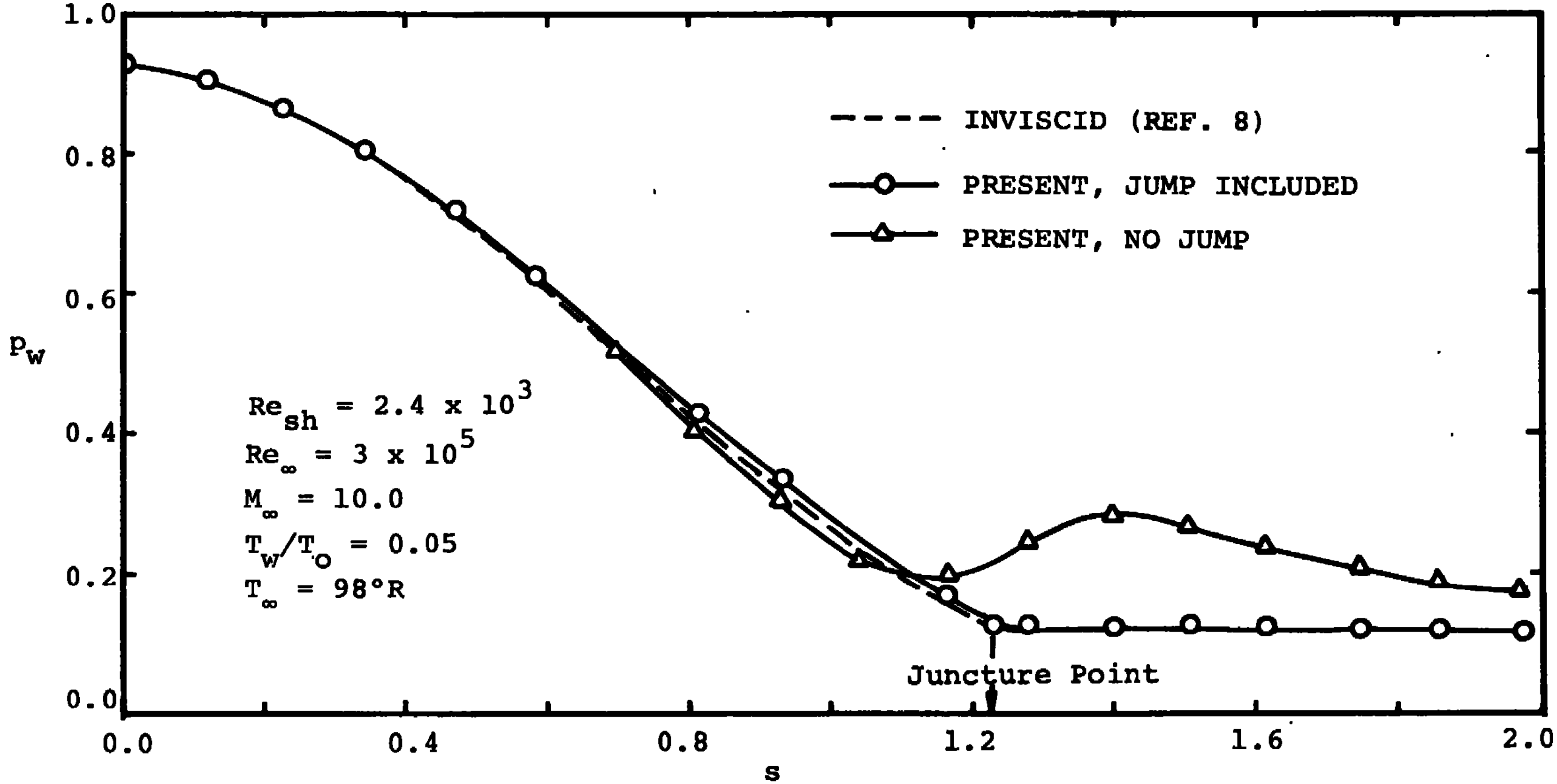


FIGURE 11 SURFACE PRESSURE DISTRIBUTION FOR A 20^{o} SPHERE-CONE

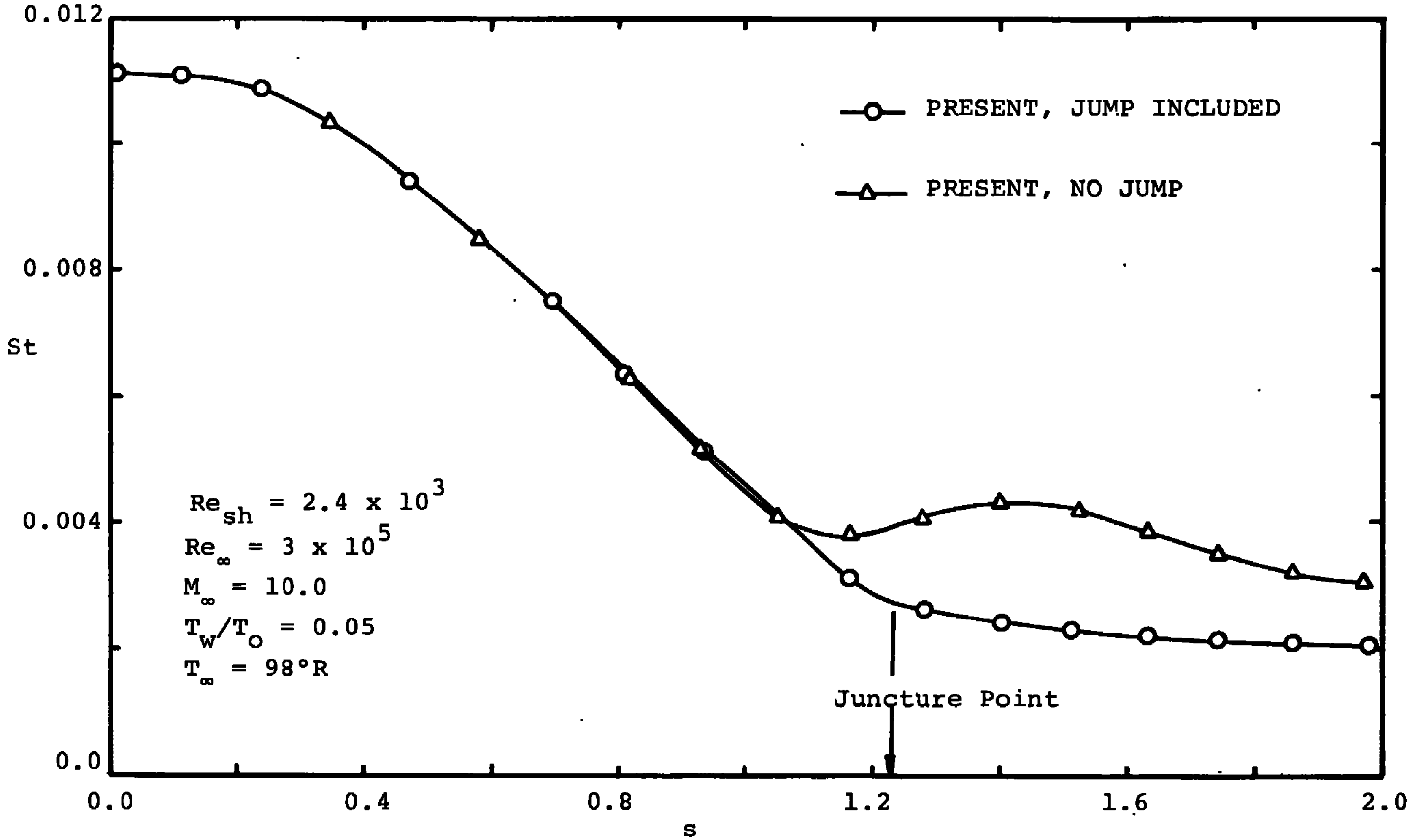


FIGURE 12 HEAT TRANSFER DISTRIBUTION FOR A 20^0 SPHERE-CONE

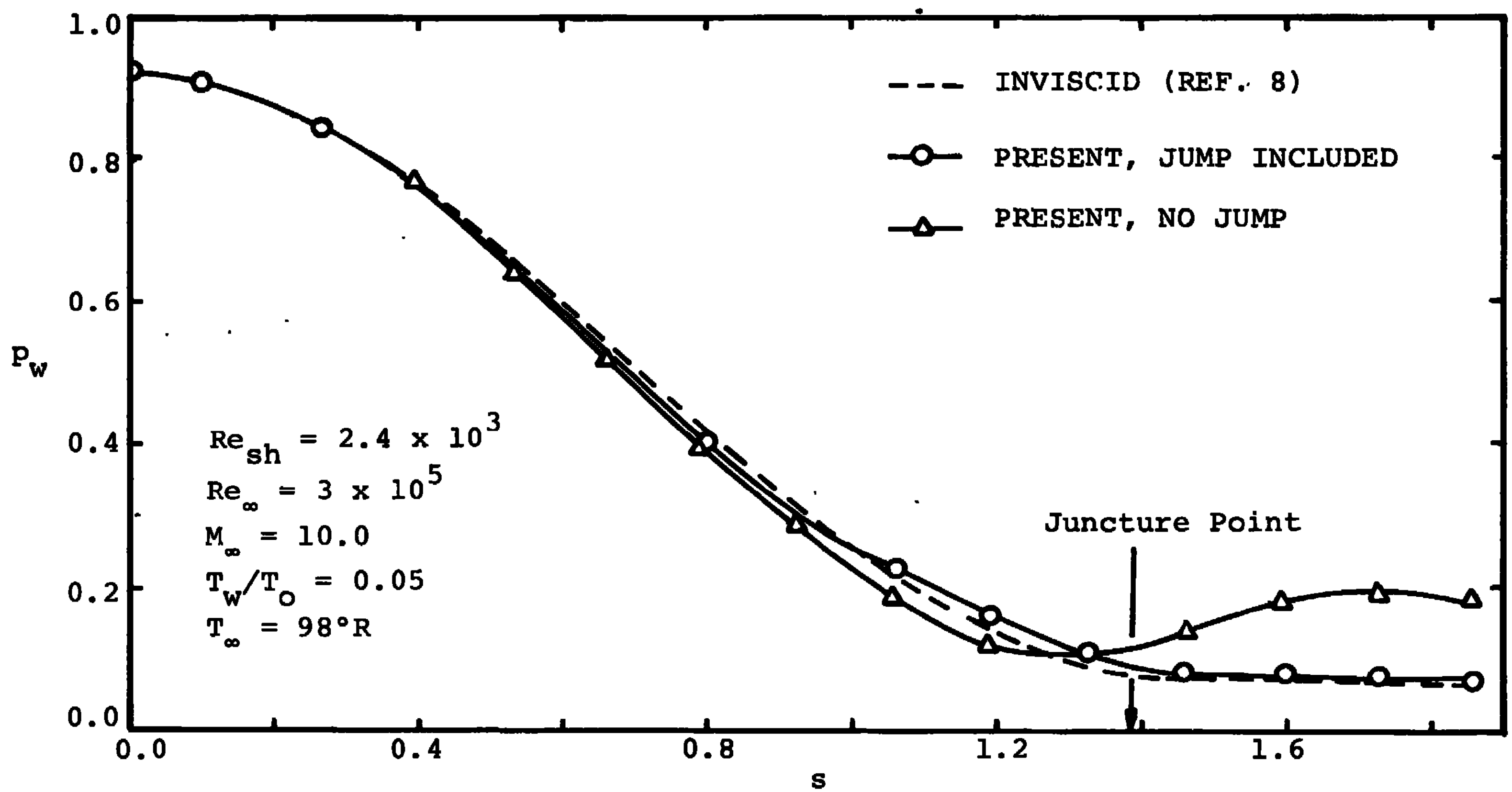


FIGURE 13 SURFACE PRESSURE DISTRIBUTION FOR A 10^o SPHERE-CONE

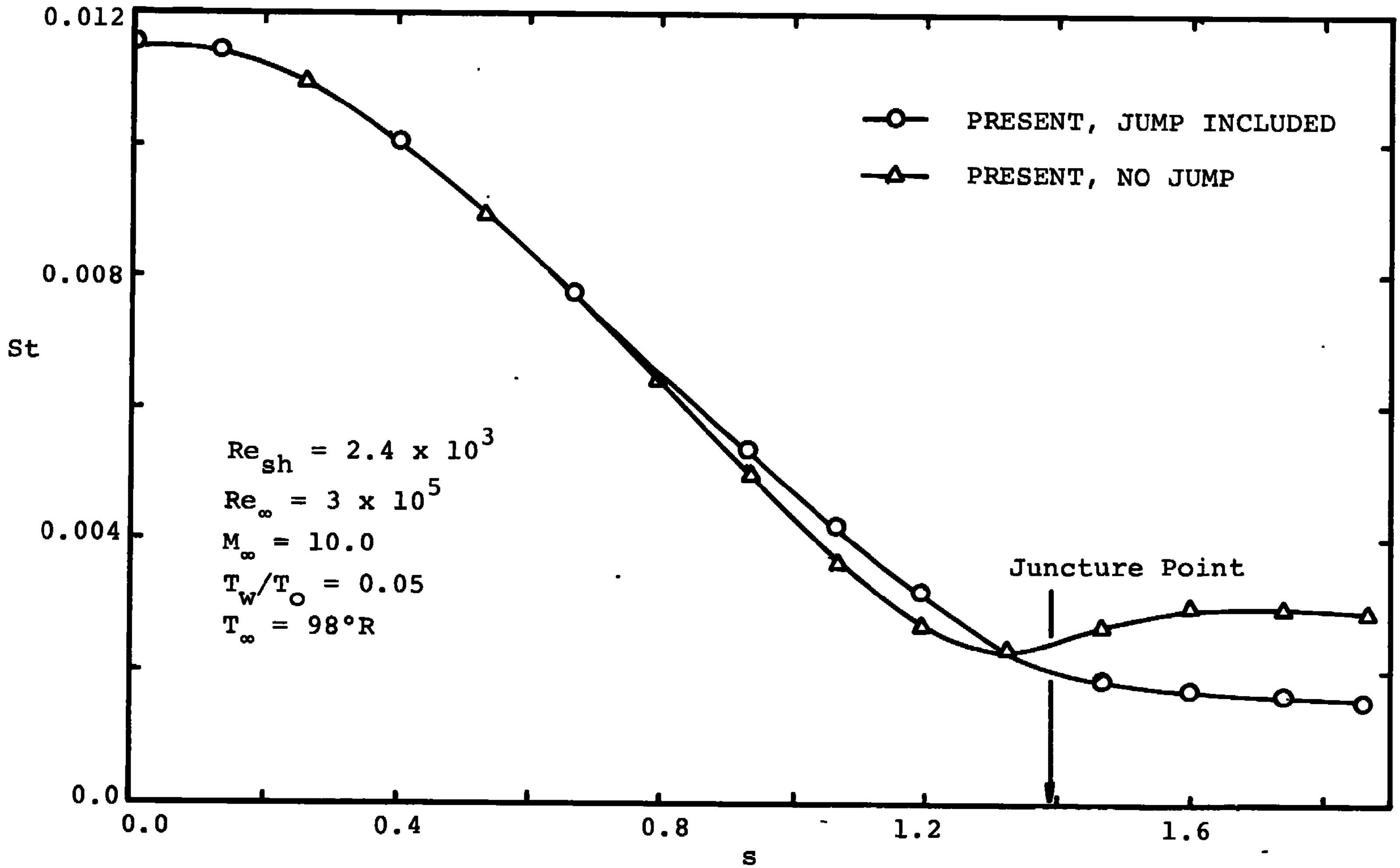


FIGURE 14 HEAT TRANSFER DISTRIBUTION FOR A 10^o SPHERE-CONE

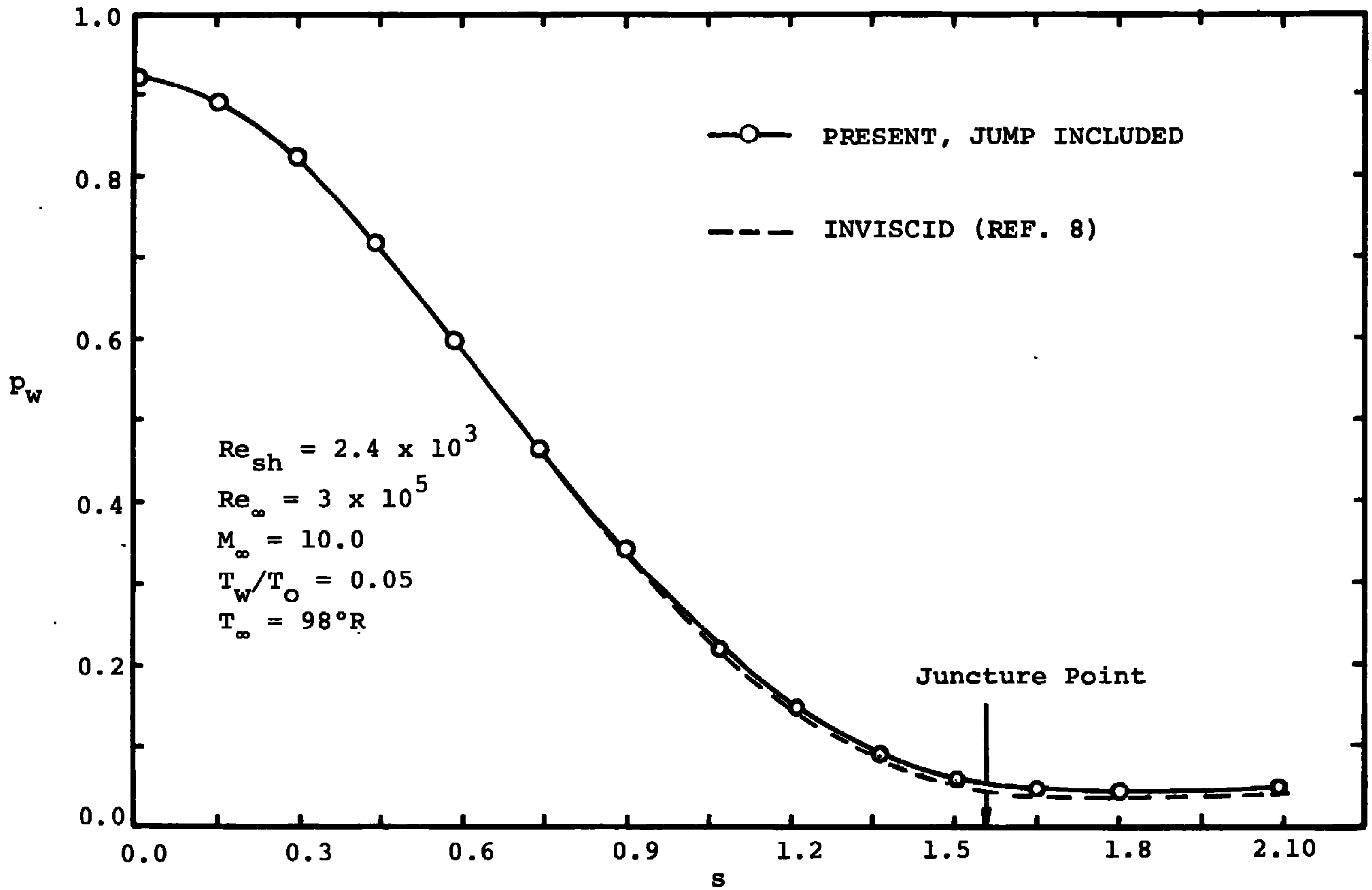


FIGURE 15 SURFACE PRESSURE DISTRIBUTION FOR A 0^0 SPHERE-CONE

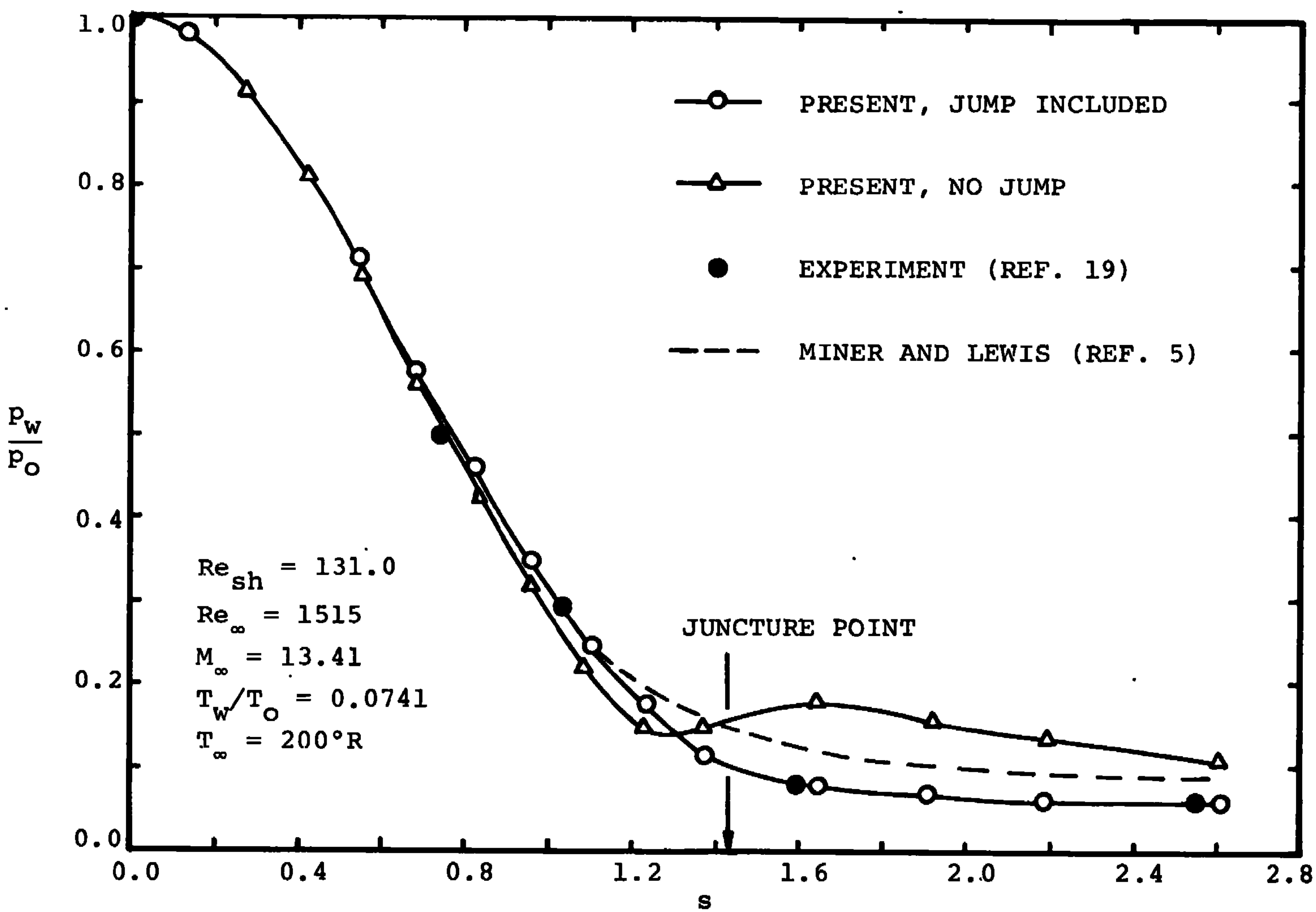


FIGURE 16 SURFACE PRESSURE DISTRIBUTION FOR A 7.5° SPHERE-CONE

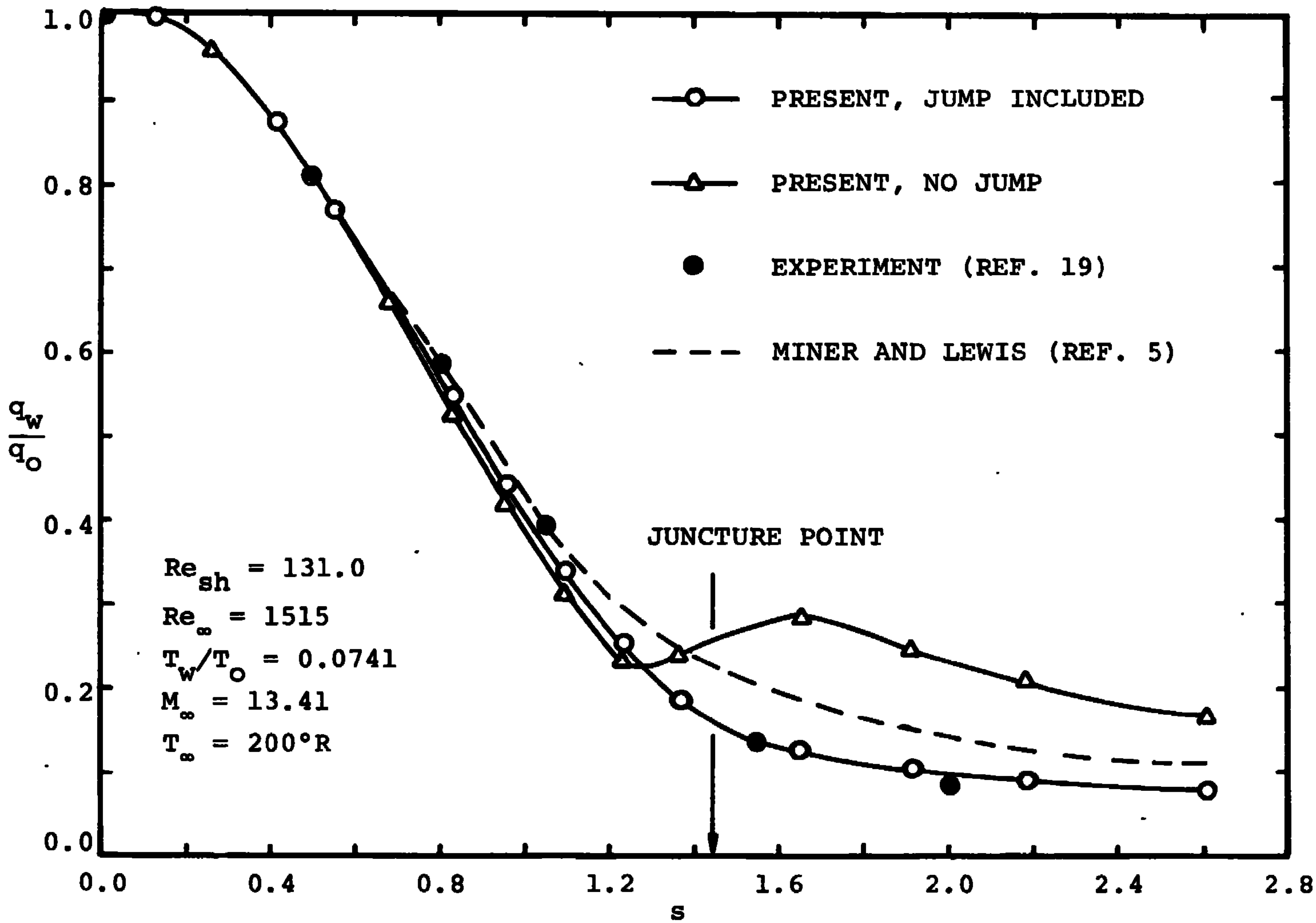


FIGURE 17 HEAT TRANSFER DISTRIBUTION FOR A 7.5^{0} SPHERE-CONE

APPENDIX A

ANALYSIS OF THE FLOW VARIABLES IN THE JUNCTURE REGION

The purpose of the present section is to study the analytical behavior of the flow field properties across a sphere/cone juncture point. In particular it is necessary to determine what, if any, restrictions the conservation laws place on the flow variables across a curvature discontinuity. For simplicity, only two dimensional flow is considered.

The analysis naturally begins with the integral form of the conservation laws since they alone are capable of accommodating discontinuities in the flow variables if they are called for. Since the viscous shock layer approach employs approximations to the differential form of the governing equations, it is first necessary to identify the equivalent approximations in the integral formulation of the problem. This is first performed for a fluid element located away from the juncture point with the results subsequently generalized to the juncture point.

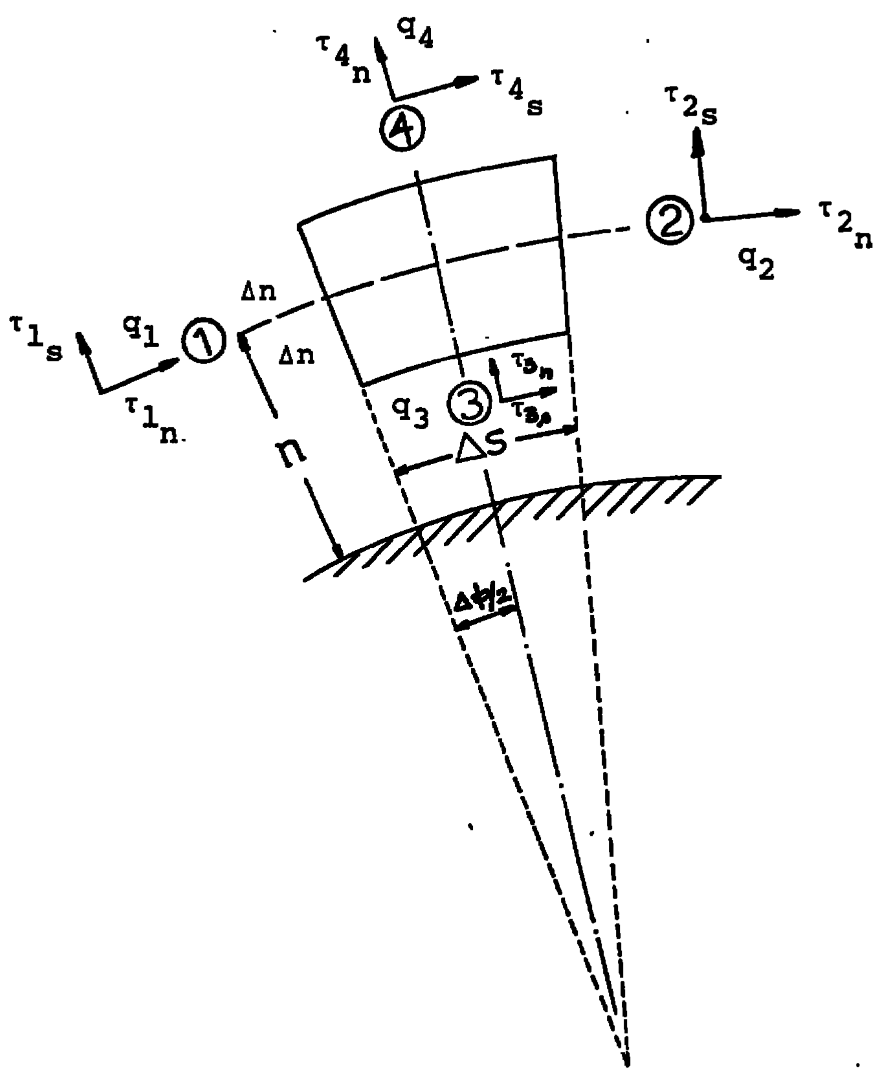
To do this, first consider the infinitessimal element shown in the sketch below as being located at some point s,n away from the point of surface curvature discontinuity. Note that from the geometry,

$$\Delta s_3 = (1 + \kappa n - \kappa \Delta n)\ \Delta s \tag{A1}$$

$$\Delta s_4 = (1 + \kappa n + \kappa \Delta n)\ \Delta s \tag{A2}$$

and the unit vectors required in the conservation equations are given as

$$
\begin{aligned}
\vec{e}_{s_1} &= \cos\frac{\Delta\phi}{2}\,\vec{e}_s + \sin\frac{\Delta\phi}{2}\,\vec{e}_n \\
\vec{e}_{s_2} &= \cos\frac{\Delta\phi}{2}\,\vec{e}_s - \sin\frac{\Delta\phi}{2}\,\vec{e}_n \\
\vec{e}_{n_1} &= \cos\frac{\Delta\phi}{2}\,\vec{e}_n - \sin\frac{\Delta\phi}{2}\,\vec{e}_s \\
\vec{e}_{n_2} &= \cos\frac{\Delta\phi}{2}\,\vec{e}_n + \sin\frac{\Delta\phi}{2}\,\vec{e}_s
\end{aligned}
\tag{A3}
$$



For small, $\frac{\Delta\phi}{2}$, these yield,

$$\vec{e}_{s_1} = \vec{e}_s + 1/2\ \kappa\ \vec{e}_n\ \Delta s + \ldots$$

$$\vec{e}_{s_2} = \vec{e}_s - 1/2\ \kappa\ \vec{e}_n\ \Delta s + \ldots$$

$$\vec{e}_{n_1} = \vec{e}_n - 1/2\ \kappa\ \vec{e}_s\ \Delta s + \ldots \qquad \text{(A4)}$$

$$\vec{e}_{n_2} = \vec{e}_n + 1/2\ \kappa\ \vec{e}_s\ \Delta s + \ldots$$

Also

$$\vec{e}_{s_3} = \vec{e}_{s_4} = \vec{e}_s$$

$$\vec{e}_{n_3} = \vec{e}_{n_4} = \vec{e}_n$$

The continuity equation in integral form is given as,

$$\iint (\rho\vec{v}.\vec{n})ds = 0 \qquad \text{(A5)}$$

which can be evaluated on the element shown to yield,

$$(\rho_2 u_2 - \rho_1 u_1)2\Delta n + [\rho_4 v_4 \Delta s_4 - \rho_3 v_3 \Delta s_3] = 0 \qquad \text{(A6a)}$$

so that with application of equations (A1) and (A2) this gives

$$(\rho_2 u_2 - \rho_1 u_1)2\Delta n + [\rho_4 v_4 (1 + \kappa n + \kappa\Delta n)$$
$$- \rho_3 v_3 (1+\kappa n-\kappa\Delta n)]\Delta s = 0 \qquad \text{(A6b)}$$

so that as $\Delta s \to 0$

$$\rho_2 u_2 = \rho_1 u_1 \tag{A7}$$

The differential form of this equation can be recovered using Taylor series expansion to write, for example, that

$$p_2 = p + \frac{\Delta s}{2} p_s + \ldots \tag{A8}$$

$$p_1 = p - \frac{\Delta s}{2} p_s + \ldots \tag{A9}$$

which, when used in equation (A6b) gives

$$(\rho u)_s + [(1 + \kappa n)\rho v]_n = 0 \tag{A10}$$

The most important point to note here is that for this equation no terms are neglected in the thin shock layer approach and hence, the equivalent thin layer version of the mass conservation law across a general line is given by equation (A7). This is not the case for the momentum equation as shown below. The momentum equation in integral form is given as:

$$\iint [(\rho\vec{v}\cdot\vec{n})\vec{v} + \rho\vec{n}]ds - \iint \vec{\tau}\, ds = 0 \tag{A11}$$

Evaluating the first (inviscid) integral on the four sides of the control volume shown in the sketch yields

$$\begin{aligned} I_1 = &-[(\rho_1 u_1^2 + p_1)\vec{e}_{s_1} + \rho_1 u_1 v_1 \vec{e}_{n_1}]2\Delta n + [(\rho_2 u_2^2 + p_2)\vec{e}_{s_2} \\ &+ \rho_2 u_2 v_2 \vec{e}_{n_2}]2\Delta n - [\rho_3 u_3 v_3 \vec{e}_{s_3} + (\rho_3 v_3^2 + p_3)\vec{e}_{n_3}]\Delta s_3 \\ &+ [\rho_4 v_4 u_4 \vec{e}_{s_4} + (\rho_4 v_4^2 + p_4)\vec{e}_{n_4}]\Delta s_4 \end{aligned} \tag{A12}$$

which is now rewritten using equations (A1)-(A4) to give

$$I_1 = \{2\Delta n \, [(p_2 + \rho_2 u_2{}^2) - (p_1 + \rho_1 u_1{}^2) + 1/2\kappa\Delta s(\rho_2 u_2 v_2 + \rho_1 u_1 v_1)] + \Delta s \, [\rho_4 u_4 v_4 (1 + \kappa n + \kappa\Delta n) - \rho_3 u_3 v_3 (1 + \kappa n - \kappa\Delta n)]\}\vec{e}_s$$

$$+ \{2\Delta n \, [\underline{\underline{\rho_2 u_2 v_2}} - \underline{\underline{\rho_1 u_1 v_1}} - 1/2 \, \kappa\Delta s(p_2 + \rho_2 u_2{}^2 + p_1 + \rho_1 u_1{}^2)]$$

$$+ \Delta s \, [(p_4 + \underline{\underline{\rho_4 v_4}}{}^2) \, (1 + \kappa n + \kappa\Delta n) - (p_3 + \underline{\underline{\rho_3 v_3}}{}^2) \, (1 + \kappa n - \kappa\Delta n)]\}\vec{e}_n$$

Similarly evaluating the viscous term yields,

$$\begin{aligned} I_2 = {} & (\underline{\tau_{1_n} \vec{e}_{n_1}} + \underline{\tau_{1_s} \vec{e}_{s_1}}) \, 2\Delta n + (\tau_{4_s} \vec{e}_{s_4} + \underline{\tau_{4_n} \vec{e}_{n_4}})\Delta s_4 \\ & + (\underline{\tau_{2_n} \vec{e}_{n_2}} + \underline{\tau_{2_s} \vec{e}_{s_2}}) \, 2\Delta n + (\underline{\tau_{3_n} \vec{e}_{n_3}} + \tau_{3_s} \vec{e}_{s_3}) \, \Delta s_4 \end{aligned} \qquad \text{(A13)}$$

In these relations those terms that would be dropped in the full shock layer approach have been identified with a single underscoring, while those additional terms whose contributions would be dropped in a thin layer approach have been given a double underscoring. These terms were identified here by first proceeding to the differential form of the governing equations, as was done for the continuity above, marking the shock layer approximations there and then tracking these backward to their source terms in equations (A12) and (A13).

The most important result to emerge from these equations is that across a general line (i.e., $\Delta s \to 0$), the flow variables in a shock layer are governed by the conditions that

$$p_2 + \rho_2 u_2^2 = p_1 + \rho_1 u_1^2 \tag{A14}$$

$$\rho_1 u_1 v_1 = \rho_2 u_2 v_2 \tag{A15}$$

Combining the second of these with the continuity equation, (A7) requires that in a full layer approach the normal velocity be continuous across a line, while in the thin layer model, no such restriction is encountered.

The same procedure can now be applied to the integral form of the energy equation in viscous flows which is given as

$$\iint \rho\left(h + \frac{v^2}{2}\right)(\vec{v}\cdot\vec{n})\,ds - \iint_s \vec{\tau}\cdot\vec{v}\,ds + \iint_s \vec{n}\cdot\vec{q}\,ds = 0 \tag{A16}$$

Evaluating the first of these terms on the four sides of the element, as before, yields,

$$\begin{aligned} I_3 = {} & 2\Delta n[\rho_2 u_2(h_2 + u_2^2/2 + V_2^2/2) - \rho_1 u_1(h_1 + u_1^2/2 + v_1^2/2)] \\ & + \Delta s[\rho_4 v_4(h_4 + u_4^2/2 + v_4^2/2)(1 + \kappa n + \kappa\Delta n) - \rho_3 v_3 \\ & (h_3 + u_3^2/2 + v_3^2/2)(1 + \kappa n - \kappa\Delta n)] \end{aligned} \tag{A17}$$

Similarly the viscous terms become

$$\iint_s \vec{\tau}\cdot\vec{v}\,ds = (\underline{\tau_{1_n} u_1} + \underline{\tau_{1_s} v_1})\,2\Delta n + (\tau_{4_s} u_4 + \underline{\tau_{4_n} v_4})\,\Delta s_4$$

$$+ (\underline{\tau_{2_n} u_2} + \underline{\tau_{2_s} v_2})\;2\Delta n + (\underline{\tau_{3_n} v_3} + \tau_{3_s} u_3)\,\Delta s_3 \qquad \text{(A18)}$$

and

$$\iint \vec{n}\cdot\vec{q}\;ds = \underline{-q_1 2\Delta n} + q_4\;\Delta s_4 + \underline{q_2\;2\Delta n} - q_3\;\Delta s_3 \qquad \text{(A19)}$$

where again those terms equivalent to the shock layer approximation have been identified with a single underscoring and those terms equivalent to the present thin shock layer concept have been given a double underscoring.

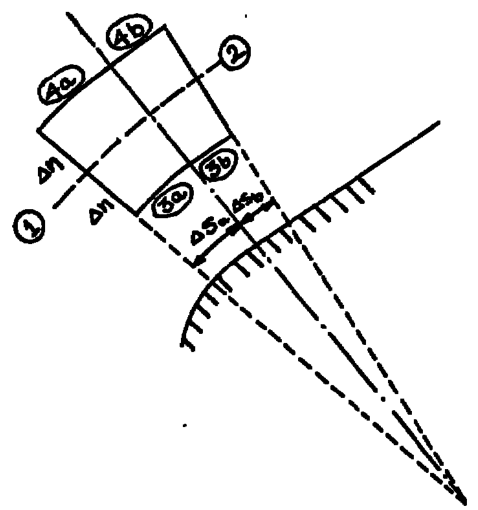
It is now possible to evaluate the resulting constraint on the flow variables across a line by setting $\Delta s \to o$ to obtain

$$h_2 + u_2^2/2 + \underline{\underline{v_2^2/2}} = h_1 + u_1^2/2 + \underline{\underline{v_1^2/2}} \qquad \text{(A20)}$$

Evaluation of this relation in combination with equation (A7), (A14), and (A15) verifies that whereas all variables ρ, u, v, and h must meet certain constraints across a line in the full shock layer approach, the present thin shock layer concept provides no constraint on the normal velocity v.

Now that the equivalent thin layer terms have been identified for a region of continuous surface curvature, one can proceed to the sphere/cone juncture point. For this study the element now straddles the juncture point and interest centers on the relation between the variables over faces 1 and 2 as Δs_a and $\Delta s_b \to o$. With this in mind it is clear that the only terms of concern in the conservation laws are those integrals over the end faces 1 and 2. Thus, for example, only the first term of the continuity equation (A6a) need be considered and thus one can write immediately that across the juncture point

$$\rho_2 u_2 = \rho_1 u_1 \tag{A21}$$



For momentum conservation, equations (A12) and (A13) represent the appropriate point of departure except that now one cannot employ equations (A1)-(A2) and (A4). None-the-less it is still clear that only those terms with Δn as a coefficient are of interest here and that all others can be ignored. With this in mind one can proceed to seek the limit form of the shock layer model as $\Delta s \to 0$ while keeping in mind the fact that on each side of the juncture line the shock layer approximations still hold. With this approach the results are identical to those presented in equations (A14) and (A15) for momentum and in equation (A20) for energy.

APPENDIX B

DERIVATION OF SHOCK DERIVATIVES

The shock derivatives du_{sh}/ds, dT_{sh}/ds, dp_{sh}/ds and dv_{sh}/ds are derived in this appendix for use in the viscous shock layer solution.

In the spatial coordinate system the shock angle, α, is written as, (Figure 1)

$$\alpha = \tan^{-1}\left(\frac{dR}{dx_s}\right) \tag{B1}$$

where $R = y_B + n_s\cos\phi$ and $x_s = x_B - n_s\sin\phi$ (B2)

Hence the derivative $d\alpha/ds$ is evaluated as,

$$\frac{d\alpha}{ds} = \frac{1}{\left[1+\left(\frac{dR}{dx_s}\right)^2\right]}\frac{d^2R}{dx_s^2}\frac{dx_s}{ds} \tag{B3}$$

Note that

$$\frac{dx_s}{ds} = \cos\phi(1+\kappa n_s) - n_s' \sin\phi \tag{B4}$$

and

$$\frac{dn_s}{ds} = (1+\kappa n_s)\tan(\alpha-\phi) \tag{B5}$$

combining (B4) and (B5) yields

$$\frac{dx_s}{ds} = (1+\kappa n_s)\frac{\cos\alpha}{\cos(\alpha-\phi)} \tag{B6}$$

substituting (B6) in (B3) and noting that $dR/dx_s = \tan\alpha$ yields after certain manipulations,

$$\frac{d\alpha}{ds} = (1+\kappa n_s)\,\frac{\cos^3\alpha}{\cos(\alpha-\phi)}\,\frac{d^2R}{dx_s^2} \tag{B7}$$

It is to be also noted that,

$$\tan\alpha = \frac{dR}{dx_s} = \frac{dR/ds}{dx_s/ds} \tag{B8}$$

Hence the second derivative, d^2R/dx_s^2, can be shown to be

$$\frac{d^2R}{dx_s^2} = \frac{d^2R/ds^2}{(dx_s/ds)^2} - \frac{d^2x_s/ds^2\ dR/ds}{(dx_s/ds)^3} \tag{B9}$$

Substituting for dx_s/ds from (B6) and then evaluating (B7) yields after proper manipulations,

$$\frac{d\alpha}{ds} = \frac{d^2R}{ds^2}\left[\frac{\cos^2(\alpha-\phi)}{(1+\kappa n_s)\cos\phi}\right] - \frac{dR}{ds}\left[\frac{\kappa\sin(2\alpha-2\phi)}{\cos\phi(1+\kappa n_s)}\right] \tag{B10}$$

Note now that from Reference [16] equation (A-6) gives

$$\frac{du_{sh}}{ds} = K_1\frac{d\alpha}{ds} - \kappa K_2 \tag{B11}$$

where

$$K_1 = \left(1 - \frac{\gamma-1}{\gamma}\frac{T_{sh}}{P_{sh}}\right)\cos(2\alpha+\beta) - \sin\alpha\cos(\alpha+\beta)\left(\frac{\gamma-1}{\gamma}\right)K_1' \tag{B12}$$

where K_1' is given by (A4) of Reference [16] and is of the form,

$$\frac{d}{ds}\left(\frac{T_{sh}}{P_{sh}}\right) = K_1' \frac{d\alpha}{ds} \tag{B13}$$

Also,

$$K_2 = \cos\alpha \cos(\alpha+\beta) + \frac{\gamma-1}{\gamma} \frac{T_{sh}}{P_{sh}} \sin\alpha \sin(\alpha+\beta) \tag{B14}$$

Likewise for other flow properties, [Ref. 16]

$$\frac{dp_{sh}}{ds} = K_3 \frac{d\alpha}{ds} \tag{B15}$$

where,

$$K_3 = \frac{2}{(\gamma+1)} \sin 2\alpha$$

$$\frac{dT_{sh}}{ds} = K_4 \frac{d\alpha}{ds} \tag{B16}$$

and

$$K_4 = \frac{2}{(\gamma+1)^2} \sin 2\alpha + \frac{4}{(\gamma+1)^2 M_\infty^4} \frac{\cos\alpha}{\sin^3\alpha}$$

$$\frac{dv_{sh}}{ds} = K_5 \frac{d\alpha}{ds} - K_6 \kappa \tag{B17}$$

where K_5 and K_6 are given by expression (A12) of Reference [16].

APPENDIX C

CHARACTERISTICS OF THE SHOCK LAYER EQUATIONS

The characteristics are obtained from the inviscid full shock layer equations (17-22) and the corresponding "strip conditions" given by

$$du = \frac{\partial u}{\partial s} ds + \frac{\partial u}{\partial n} dn \tag{C1}$$

$$dv = \frac{\partial v}{\partial s} ds + \frac{\partial v}{\partial n} dn \tag{C2}$$

$$d\rho = \frac{\partial \rho}{\partial s} ds + \frac{\partial \rho}{\partial n} dn \tag{C3}$$

$$dp = \frac{\partial p}{\partial s} ds + \frac{\partial p}{\partial n} dn \tag{C4}$$

The terms in the normal momentum equation (19) which are not included in the thin shock layer version of the full shock layer equations will be marked here by introduction of a multiplicative factor, α, which would be zero for the thin shock layer equations.

Using Cramer's rule to identify the derivatives gives

$$\frac{\partial u}{\partial n} = |A|/|B| \tag{C5}$$

where

$$|B| = \begin{vmatrix} dn & ds & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & dn & ds & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & dn & ds & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & dn & ds \\ 0 & \rho & (1+\kappa n)\rho & 0 & (1+\kappa n)v & u & 0 & 0 \\ (1+\kappa n)\rho v & \rho u & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & \alpha(1+\kappa n)\rho v & \alpha\rho u & 0 & 0 & (1+\kappa n) & 0 \\ 0 & 0 & 0 & 0 & -(1+\kappa n)va^2 & -ua^2 & (1+\kappa n)v & u \end{vmatrix}$$

$$|A| = \begin{vmatrix} du & ds & 0 & 0 & 0 & 0 & 0 & 0 \\ dv & 0 & dn & ds & 0 & 0 & 0 & 0 \\ d\rho & 0 & 0 & 0 & dn & ds & 0 & 0 \\ dp & 0 & 0 & 0 & 0 & 0 & dn & ds \\ -\kappa\rho v & \rho & (1+\kappa n)\rho & 0 & (1+\kappa n)v & u & 0 & 0 \\ -\kappa\rho uv & \rho u & 0 & 0 & 0 & 0 & 0 & 1 \\ \kappa\rho u^2 & 0 & \alpha(1+\kappa n)\rho v & \alpha\rho u & 0 & 0 & (1+\kappa n) & 0 \\ 0 & 0 & 0 & 0 & -(1+\kappa n)va^2 & -ua^2 & (1+\kappa n)v & u \end{vmatrix} \tag{C6}$$

Setting $|B| = 0$, expanding and simplifying yields

$$\begin{aligned}[]&[udn - (1+\kappa n)vds]\{\alpha dn^3 u(u^2-a^2) - \alpha dn^2 ds\, u^2(1+\kappa n)v \\ &- \alpha dn^2 ds\, v(1+\kappa n)(u^2-a^2) - dnds^2 u(1+\kappa n)^2(a^2-\alpha v^2) \\ &- dsdn^2\, v(1+\kappa n)u^2\alpha + ds^2 dn\, v^2(1+\kappa n)^2\, u\alpha \\ &+ \alpha\rho^2 v^2(1+\kappa n)^2\, ds^2 dn\, u - (1+\kappa n)^3 ds^3 v(-\alpha v^2+a^2)\} = 0\end{aligned} \tag{C7}$$

Note that

$$u dn - (1+\kappa n) v\, ds = 0$$

gives the equation of a streamline in this coordinate system. With further simplification, the final expressions for the characteristics can be written as

$$\left(\frac{dn}{ds}\right)_{1,2} = \frac{\dfrac{-uv(1+\kappa n)}{a^2} \mp \dfrac{(1+\kappa n)}{a} \sqrt{\dfrac{1}{\alpha}(u^2+\alpha v^2-a^2)}}{(1 - u^2/a^2)} \qquad \text{(C8)}$$

Note that when $\alpha=1$, which corresponds to the full shock layer equations, the characteristics are inclined at a Mach angle in the supersonic flow. However, as α approaches zero, corresponding to the thin shock layer version of these equations, the characteristics in the flow field tend to become perpendicular to the surface. This result can be verified by beginning with the thin layer equations and repeating the above derivation. For the thin layer case it can be shown that the characteristics are given as

$$- ds^2 \rho^2 a^2 (1+\kappa n)^2 \, [dnu - (1+\kappa n) vds]^2 = 0 \qquad \text{(C9)}$$

Note that this equation indicates that either

$$ds^2 = 0 \qquad \text{(C10)}$$

or

$$[udn - (1+\kappa n) vds]^2 = 0 \qquad \text{(C11)}$$

Since (C11) represents the equation of a streamline in this coordinate system, the equation for the characteristics are given by (C10). Along these characteristics, compatibility conditions must be satisfied, these being obtained from,

$$ds^2\rho(1+\kappa n)\ [-ua^2dn + (1+\kappa n)va^2ds]\,[-\rho\kappa u^2dn+\rho(1+\kappa n)udu$$
$$+ (1+\kappa n)dp + \rho(1+\kappa n)\kappa uvds] = 0 \qquad (C12)$$

Note that this equation (C12) is satisfied along a characteristic line $ds=0$ indicating that no other additional condition need be satisfied. It is, therefore, seen that the inviscid set of thin shock layer equations predict coincident characteristics normal to the surface of the body. Davis [20] discusses the characteristics and the nature of these equations (i.e. whether they are elliptic, parabolic or hyperbolic) when viscous effects are included.

APPENDIX D

ADI FORMULATION OF S-MOMENTUM EQUATION

The s-momentum equation is written in the following form in the surface coordinate system.

$$\frac{\rho u\ u_s}{(1+\kappa n)} + \rho v u_n + \frac{\kappa u v \rho}{(1+\kappa n)} + \frac{p_s}{(1+\kappa n)}$$

$$= [\varepsilon^2/(1+\kappa n)^2 (r+n\ \cos\phi)^j] [(1+\kappa n)^2 (r+n\ \cos\phi)^j \tau]_n \tag{D1}$$

where

$$\tau = \mu [u_n - \kappa u/(1+\kappa n)]$$

Using the transformation given in equations (5a-h) this becomes

$$\frac{\partial^2 \bar{u}}{\partial \eta^2} + \alpha_1 \frac{\partial \bar{u}}{\partial \eta} + \alpha_2\ \bar{u} + \alpha_3 + \alpha_4 \frac{\partial \bar{u}}{\partial \xi} = 0 \tag{D2}$$

where α_1, α_2, α_3 and α_4 are given by equations (8a-d). Note that u'_{sh} and p'_{sh} appear in the coefficients α_2 and α_3. From Appendix (B) u'_{sh} and p'_{sh} are written as,

$$u'_{sh} = K_1 \frac{d\alpha}{ds} - K_2 \tag{D3}$$

$$p'_{sh} = K_3 \frac{d\alpha}{ds} \tag{D4}$$

where $d\alpha/ds$ is given by (B10) of Appendix (B). Substituting $d\alpha/ds$ in (D3) and (D4) and then evaluating the coefficients α_2 and α_3 yields

$$\alpha_2 = \gamma_2 \frac{d^2R}{ds^2} + \gamma_3 \frac{dR}{ds} + \gamma_4 \tag{D5}$$

$$\alpha_3 = \gamma_5 \frac{d^2R}{ds^2} + \gamma_6 \frac{dR}{ds} + \gamma_7 \tag{D6}$$

where,

$$\gamma_4 = +A\kappa\ K_2 + B + C + D \tag{D7}$$

$$\gamma_3 = +A\ K_1 \frac{K\sin(2\alpha-2\phi)}{\cos\phi(1+\kappa n_s)} \tag{D8}$$

$$\gamma_2 = -A\ K_1 \frac{\cos^2(\alpha-\phi)}{(1+\kappa n_s)\cos\phi} \tag{D9}$$

and

$$A = + \frac{\rho_{sh}\ n_{sh}}{\varepsilon^2\ \mu_{sh}} \ \frac{\kappa n_{sh}}{1+\kappa n_{sh}\eta} \ \frac{\bar{\rho}\bar{u}}{\bar{\mu}}$$

$$B = - \frac{\rho_{sh} v_{sh} n_{sh}}{\varepsilon^2\ \mu_{sh}} \ \frac{n_{sh}}{1+\kappa n_{sh}\eta} \ \frac{\bar{\rho}\bar{v}}{\bar{\mu}}$$

$$C = - \frac{\kappa n_{sh}}{1+\kappa n_{sh}\eta} \ \frac{\bar{\mu}_n}{\bar{\mu}}$$

$$D = -\left(\frac{\kappa n_{sh}}{(1+\kappa n_{sh}\eta)} + \frac{\cos\phi\ n_{sh}}{r+n_{sh}\cos\phi\eta}\right) \frac{\kappa n_{sh}}{(1+\kappa n_{sh}\eta)}$$

In a similar manner,

$$\gamma_7 = A_1 \; \bar{p}_\xi - \frac{n_{sh}'}{n_{sh}} \; \eta \; \bar{p}_\eta) \tag{D10}$$

$$\gamma_6 = -A_1 \frac{\bar{p}}{p_{sh}} K_3 \frac{K \sin(2\alpha - 2\phi)}{\cos\phi(1+\kappa n_{sh})} \tag{D11}$$

$$\gamma_5 = A_1 \frac{\bar{p}}{p_{sh}} \frac{\cos^2(\alpha-\phi)}{(1+\kappa n_{sh})\cos\phi} \tag{D12}$$

and

$$A_1 = - \frac{p_{sh} \; n_{sh}}{\varepsilon^2 \; \mu_{sh}} \; \frac{n_{sh}}{(1+\kappa n_{sh}\eta)} \; \frac{1}{\bar{\mu} \; u_{sh}}$$

hence equation (D2) can now be written as

$$\frac{\partial^2 \bar{u}}{\partial \eta^2} + \alpha_1 \frac{\partial \bar{u}}{\partial \eta} + (\gamma_2 \frac{d^2 R}{ds^2} + \gamma_3 \frac{dR}{ds} + \gamma_4)\bar{u}$$

$$+ \; (\gamma_5 \frac{d^2 R}{ds^2} + \gamma_6 \frac{dR}{ds} + \gamma_7) + \alpha_4 \frac{\partial \bar{u}}{\partial \xi} \; = \; 0$$

Further rearrangement yields,

$$\frac{\partial^2 \bar{u}}{\partial \eta^2} + \alpha_1 \frac{\partial \bar{u}}{\partial \eta} + (\gamma_2 \bar{u} + \gamma_5) \frac{d^2 R}{ds^2} + (\gamma_3 \bar{u} + \gamma_6) \frac{dR}{ds}$$

$$+ \; (\gamma_4 \bar{u} + \gamma_7) + \alpha_4 \frac{\partial \bar{u}}{\partial \xi} \; = \; 0 \tag{D13}$$

For the first half time step of the present alternating direction implicit method, this last expression is written as

First Sweep: $t^* = t^n + \Delta t/2$

$$\frac{\partial^2 \bar{u}^*}{\partial \eta^2} + \beta_1^* \frac{\partial \bar{u}}{\partial \eta} + \beta_2^* \left[\frac{\partial^2 R^n}{\partial s^2} - \frac{\partial R^*}{\partial t}\right] + \beta_3^* \frac{\partial R^n}{\partial s}$$

$$+ \beta_4^* + \beta_5^* \frac{\partial \bar{u}^*}{\partial \xi} = 0 \qquad \text{(D14)}$$

For the second half time step equation (D13) is written as

Second Sweep: $t^{n+1} = t^* + \frac{\Delta t}{2}$

$$\beta_2^* \frac{\partial^2 R^{n+1}}{\partial s^2} - \beta_2^* \frac{\partial R^{n+1}}{\partial t} + \beta_3^* \frac{\partial R^{n+1}}{\partial s} + \left[\frac{\partial^2 \bar{u}}{\partial \eta^2} + \beta_1 \frac{\partial \bar{u}}{\partial \eta}\right.$$

$$\left. + \beta_5 \frac{\partial \bar{u}}{\partial \xi} + \beta_4\right]^* = 0 \qquad \text{(D15)}$$

where,

$$\beta_1 = \alpha_1$$

$$\beta_2 = \gamma_2 \bar{u} + \gamma_5$$

$$\beta_3 = \gamma_3 \bar{u} + \gamma_6$$

$$\beta_4 = \gamma_4 \bar{u} + \gamma_7$$

$$\beta_5 = \alpha_4 \qquad \text{(D16)}$$

However note that equation (D15) for R must be independent of η indicating that the coefficients of this equation must be independent of η.

It can be shown by proper substitution that,

$$\beta_3/\beta_2 = -2\kappa \tan(\alpha-\phi) \tag{D17}$$

whereas by using the first sweep equation

$$\left[\frac{\partial^2 \bar{u}}{\partial \eta^2} + \beta_1 \frac{\partial \bar{u}}{\partial \eta} + \beta_5 \frac{\partial \bar{u}}{\partial \xi} + \beta_4\right]/\beta_2$$

$$= -\frac{\partial^2 R^n}{\partial s^2} + \frac{2R^*}{\Delta t} - \frac{2R^n}{\Delta t} + 2\kappa \tan(\alpha-\phi) \frac{\partial R^n}{\partial s} \tag{D18}$$

Substituting all of this the final sweep equation is given as,

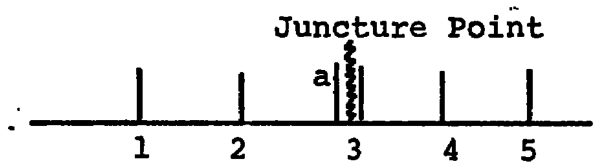
$$\frac{\partial^2 R^{n+1}}{\partial s^2} - 2\kappa \tan(\alpha-\phi) \frac{\partial R^{n+1}}{\partial s} - \frac{2}{\Delta t} R^{n+1} - \frac{\partial^2 R^n}{\partial s^2}$$

$$+ 2\kappa \tan(\alpha-\phi) \frac{\partial R^n}{\partial s} + \frac{2}{\Delta t} (2R^* - R^n) = 0 \tag{D19}$$

APPENDIX E

DERIVATION OF FINITE DIFFERENCE EXPRESSIONS AT A JUNCTURE POINT

Consider a typical mesh system where a juncture point occurs immediately ahead of point 3. Point 'a' is taken to be immediately ahead of the juncture point.



Jump conditions associated with the first and second derivatives at the juncture point are given in the form

$$\left(\frac{dR}{ds}\right)_a = K_1 \left(\frac{dR}{ds}\right)_3 \tag{E1}$$

$$\left(\frac{d^2R}{ds^2}\right)_a = \left(\frac{d^2R}{ds^2}\right)_3 + K_2\left(\frac{dR}{ds}\right)_3 + K_3 \tag{E2}$$

where K_1, K_2 and K_3 are known and are given by equations (41b) for the model problem. Using Taylor's series expansion,

$$R_2 = R_a - \Delta R_a' + \frac{\Delta^2}{2} R_a'' - \frac{\Delta^3}{6} R_a''' + \ldots \tag{E3}$$

$$R_4 = R_3 + \Delta R_3' + \frac{\Delta^2}{2} R_3'' + \frac{\Delta^3}{6} R_3'' + \ldots \tag{E4}$$

Rearranging this, (E3) can be rewritten using (E1) and (E2) as,

$$R_2 = R_3 - R_3'\left(K_1 - \frac{\Delta}{2} K_2\right)\Delta + \frac{\Delta^2}{2} K_3 + \frac{\Delta^2}{2} R_3''$$

$$- \frac{\Delta^3}{6} R_a''' + \ldots \tag{E5}$$

so that finally,

$$R_3' = \frac{(R_4 - R_2)}{\Delta\left(K_1 - \frac{\Delta}{2} K_2 + 1\right)} + \frac{\Delta K_3}{2\left(K_1 - \frac{\Delta}{2} K_2 + 1\right)} + 0(\Delta^2) \tag{E6}$$

Using (E4) and (E5) and simplifying yields,

$$R_3'' = \frac{(R_2 - R_3) + \left(K_1 - \frac{\Delta}{2} K_2\right)(R_4 - R_3)}{\frac{\Delta^2}{2}\left(K_1 - \frac{\Delta}{2} K_2 + 1\right)} - \frac{K_3}{\frac{\Delta^2}{2}\left(K_1 - \frac{\Delta}{2} K_2 + 1\right)}$$

$$- \frac{\Delta}{3}\left[\frac{\left(K_1 - \frac{\Delta}{2} K_2\right) R_3''' - R_a'''}{\left(K_1 - \frac{\Delta}{2} K_2 + 1\right)}\right] + 0(\Delta^2) \tag{E7}$$

Expression (E7) gives a formally first order accurate finite difference form of the second derivative immediately behind the juncture point. Note that the first order error in (E7) can be estimated for the simple model problem through differentiation of the differential equation on the two sides of the juncture point. From the model equation (38) differentiated and evaluating at "a" and "3", one obtains

$$R_a''' + \alpha_{1a} R_a'' + \alpha_2 R_a' = 0 \tag{E8}$$

$$R_3''' + \alpha_{1_3} R_3'' + \alpha_2 R_3' = 0 \tag{E9}$$

so that

$$R_a''' - (K_1 - \frac{\Delta}{2} K_2)R_3''' = [\alpha_{1_3}(K_1 - \frac{\Delta}{2} K_2) - \alpha_{1a}]R_3''$$
$$- R_3'[K_2 \alpha_{1a} - \alpha_2(K_1 - \frac{\Delta}{2} K_2) + \alpha_2 K_1] - K_3 \alpha_{1a} \tag{E10}$$

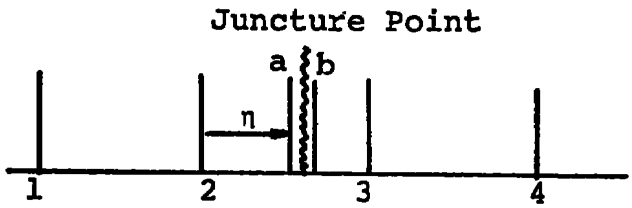
The error term in (E7) can now be estimated as,

$$E = \frac{\Delta[\alpha_{1_3}(K_1 - \frac{\Delta}{2} K_2) - \alpha_{1a}]}{3(K_1 - \frac{\Delta}{2} K_2 + 1)} R_3'' - \Big[R_3'[\{\alpha_{1a}K_2 - \alpha_2(K_1 - \frac{\Delta}{2} K_2)$$
$$+ \alpha_2 K_1\}\Delta] + K_3 \alpha_{1a} \Delta\Big] \Big/ 3(K_1 - \frac{\Delta}{2} K_2 + 1) \tag{E11}$$

Note that when (E11) is substituted in (E7) and formally the second order accurate finite difference form of the first derivative, dR/ds, is utilized, this would result in a formally second order accurate finite difference form of the second derivative, d^2R/ds^2.

The above difference formulations are valid only when a mesh point of the finite difference scheme coincides with the juncture point where discontinuities in derivatives are encountered. These formulations need to be rederived when the juncture point lies between two mesh points of the finite difference solution scheme. This is achieved in the following analysis.

Consider the mesh points as shown where the juncture point lies at a finite distance of away from point 2. Points a and b are immediately ahead and behind the juncture point.



Let η represent a fraction of the step size Δ. Thus,

$$\eta = \xi\Delta \qquad 1 \leq \xi \leq 0 \tag{E12}$$

Expressions are now sought for the shock derivatives at points "2" and "3" with an embedded jump occurring from "a" to "b". Using Taylor's series expansion,

$$R_1 = R_2 - \Delta R_2' + \frac{\Delta^2}{2} R_2'' - \frac{\Delta^3}{6} R_2''' + \ldots \tag{E13}$$

$$R_a = R_2 + \Delta\xi R_2' + \frac{\Delta^2}{2} \xi^2 R_2'' + \frac{\Delta^3}{6} \xi^3 R_2''' + \ldots \tag{E14}$$

$$R_3 = R_b + \Delta(1-\xi) R_b' + \frac{\Delta^2}{2} (1-\xi)^2 R_b'' + \frac{\Delta^3}{6}(1-\xi)^3 R_b''' + \ldots \tag{E15}$$

$$R_a' = R_2' + \Delta\xi R_2'' + \frac{\Delta^2\xi^2}{2} R_2''' + \ldots \tag{E16}$$

$$R_a'' = R_2'' + \Delta\xi R_2''' + \ldots \tag{E17}$$

After considerable manipulations these give,

$$R_2'' = \frac{(\xi+A)R_1 + R_3 + \frac{\Delta^2}{2}(1-\xi)^2 K_3 - R_2(\xi+A+1)}{\Delta^2[\frac{\xi+A}{2} + \frac{\xi^2}{2} + \xi A + \frac{(1-\xi)^2}{2}]} + 0(\Delta) \tag{E18}$$

and

$$R_2' = \{R_3 - 2R_1P_2 - R_2(1-2P_2) + \frac{\Delta^2}{2}(1-\xi^2)K_3\}/[\Delta(P_1+P_2)] + 0(\Delta^2) \tag{E19}$$

where

$$P_1 = \xi + A$$

$$P_2 = \frac{\xi^2}{2} + \xi A + (1-\xi)^2/2$$

In a similar fashion one obtains,

$$R_3'' = [R_2 + R_4P_1 - R_3(1+P_1) - \frac{\Delta^2}{2}\xi^2 K_3]/[(P_2+P_1/2)\Delta^2] + 0(\Delta) \tag{E20}$$

and

$$R_3' = [2R_4P_2 + R_3(1-2P_2) - R_2 + \frac{\Delta^2}{2}\xi^2 K_3]/[(P_1+2P_2)\Delta] + 0(\Delta^2) \tag{E21}$$

where

$$P_1 = [\xi(K_1 - \frac{\Delta}{2}\xi K_2) + (1-\xi)]$$

$$P_2 = [\xi(1-\xi)(K_1 - \frac{\Delta}{2}\xi K_2) + \frac{\xi^2}{2} + \frac{1}{2}(1-\xi)^2]$$

APPENDIX F

ERROR ANALYSIS OF THE "SHOCK JUMP" MODEL PROBLEM

For the simple "shock jump" model problem studied in Section IV the exact solution is known and thus a detailed study of the accuracy of the finite difference scheme can be undertaken.

First, in order to establish the truncation error of the finite difference expression (44b) at the juncture point, the exact values of the shock shape, R, were used in this expression to obtain a numerical estimate to the second derivative, d^2R/ds^2, at the point of discontinuity, i.e., at $S=0.9$. Figure 18 shows that this derivative linearly approaches its exact value in Δ, indicating that the finite difference form (44b) is indeed first order accurate at the point of discontinuity. Note also, from this figure that at any other point such as $S=0.5$ where no discontinuity of any kind is present the finite difference expression (44b) is seen to be second order accurate as anticipated since in such a case the jump constants K_1, K_2 and K_3 take values of 1, 0 and 0 respectively. These results clearly indicate that the local truncation errors are of first order at a jump and second order everywhere else. It is, therefore, evident that the second order type behavior observed earlier can only be explained through a study of the overall truncation error of the numerical scheme.

In order to better demonstrate this concept a simpler problem for which an analytical assessment of the error is possible, was considered. Thus we take the simple problem given as

$$\frac{d^2R}{ds^2} - b^2R = 0 \tag{F1}$$

Subject to the boundary conditions

$$R(0) = 0 \qquad \text{and} \qquad R(1) = 1 \tag{F2}$$

which has the exact solution as,

$$R = [\frac{e^{bs} - e^{-bs}}{e^{b} - e^{-b}}] \tag{F3}$$

Consideration is now given to three different schemes for numerically solving the same problem, the first two of which establish the method for assessing the overall truncation error and the last of which directly addresses the present problem.

Case (a) - Consider the case where d^2R/ds^2 is represented by a second order accurate expression in the entire solution region. In such a case it is possible to show straightforwardly that the difference solution approaches the exact solution (F3) quadratically in Δs. Since this is the basis for later studies, it is shown below in detail.

The difference version of equation (F1) is given as

$$R_{i+1} - R_i(2 + b^2\Delta^2) + R_{i-1} = 0 \tag{F4}$$

$$R_o = 0 \quad \text{and} \quad R_N = 1 \tag{F5}$$

which has a solution of the form

$$R_i = A\, s_1^i + B\, s_2^i \tag{F6}$$

where

$$s_{1,2} = 1 + \frac{b^2\Delta^2}{2} \pm \sqrt{\frac{b^4\Delta^4}{4} + b^2\Delta^2} \tag{F7}$$

Applying the boundary condition yields the final solution

$$R_i = \left[\frac{s_1^i - s_2^i}{s_1^N - s_2^N}\right] \tag{F8}$$

In the limit of zero step size, Δ, the difference solution (F8) should approach the exact solution (F3) quadratically in Δ. To verify this, consider now the limiting process, as $\Delta \to 0$. First write that

$$s_{1,2} = 1 \pm b\Delta + \frac{b^2\Delta^2}{2} \pm \frac{b^3\Delta^3}{8} + \ldots \tag{F9}$$

and note that

$$s_1^i = s_1^{s/\Delta} \tag{F10}$$

which is now rewritten as

$$s_1^i = e^{s/\Delta \log_e(1+b\Delta)} e^{s/\Delta \log_e(1+ \frac{b^2\Delta^2}{2})} \{1 - \frac{3b^3\Delta^3}{8} - \frac{25}{128} b^5\Delta^5 + \ldots. + \frac{3b^4\Delta^4}{8} + \frac{25b^6\Delta^6}{128} \ldots\}^{s/\Delta}$$

Expanding $\log_e(1+x)$ for small x and rewriting the resulting expression in terms of exponential functions we have,

$$s_1^{s/\Delta} = e^{bs} e^{-b^2\Delta s/2} e^{b^3\Delta^2 s/2} \ldots \times e^{b^2\Delta s/2} e^{-b^4\Delta^3 s/8} \times e^{b^6\Delta^5 s/24} \ldots e^{s/\Delta \log_e(1- \frac{3b^3\Delta^3}{8})} \{1 + \frac{3b^4\Delta^4}{8} (1- \frac{3b^3\Delta^3}{8})^{-1} \ldots\}^{s/\Delta}$$

Note that the first order contribution in Δ, is precisely cancelled out yielding,

$$s_1^{s/\Delta} = e^{bs} [(1+ \frac{5b^3\Delta^2 s}{24} + \ldots)(1+\ldots)\ldots]$$

It is seen, therefore, that $s_1^{s/\Delta}$ approaches its exact solution, e^{bs}, quadratically in Δ.

Due to symmetry, a similar analysis for the remaining terms results in the following forms.

$$s_1^i = e^{bs}[1 + A\Delta^2 + B\Delta^3 + ...]$$

$$s_2^i = \bar{e}^{bs}[1 + A_1\Delta^2 + B_1\Delta^3 + ...]$$

$$s_1^N = e^{b}[1 + a\Delta^2 + b\Delta^3 + ...]$$

$$s_2^N = \bar{e}^{b}[1 + a_1\Delta^2 + b_1\Delta^3 + ...]$$

Hence the difference solution, as $\Delta \to 0$, can be written as

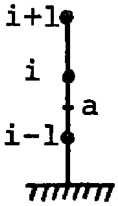
$$R_i = \frac{(e^{bs}-\bar{e}^{bs})+\Delta^2(Ae^{bs}-A_1\bar{e}^{bs})+\Delta^3(...)}{(e^{b}-\bar{e}^{b})+\Delta^2(ae^{b}-a_1\bar{e}^{b})+\Delta^3(...)} \tag{F11}$$

which can be manipulated to the form

$$R_i = \frac{e^{bs} - \bar{e}^{bs}}{e^{b} - \bar{e}^{b}} + 0(\Delta^2)$$

Figure 19 shows the computational result verifying this analytical derivation where the function "R" and its derivative, d^2R/ds^2, are seen to approach their exact values as a straight line in the square of the step size, Δ^2.

Case (b) - Consider now a case when equation (F1) is solved with an algorithm that is first order accurate in the entire solution region. To do this the source term R of difference equation is written at the midpoint between two mesh points, point a, using the average value of R_i and R_{i-1} to approximate R_a. A centered second order difference is used to represent R_i''. The resulting first order accurate difference scheme is given as



$$R_{i+1}(2-b^2\Delta^2) - R_i(4+b^2\Delta^2) + 2R_{i-1} = 0 \quad \text{(F12)}$$

Using the same boundary conditions as before,

$$R_o = 0 \;, \qquad R_N = 1 \quad \text{(F13)}$$

the difference equation solution is found to be

$$R_i = \frac{s_1^i - s_2^i}{s_1^N - s_2^N} \quad \text{(F14)}$$

where now

$$s_1 = 1 + b\Delta + \frac{3b^2\Delta^2}{4} + \frac{17b^3\Delta^3}{32} + \ldots.$$

$$s_2 = 1 - b\Delta + \frac{3b^2\Delta^2}{4} - \frac{17b^3\Delta^3}{32} + \ldots.$$

Following the same procedure of manipulation as before, it can be shown that

$$s_1^i = (e^{bs}\, e^{-b^2\Delta s/2}\, e^{b^3\Delta^2 s/3} \ldots..)$$
$$(e^{3/4\, b^2\Delta s}\; e^{-9/16\, b^4\Delta^3 s} \ldots)$$

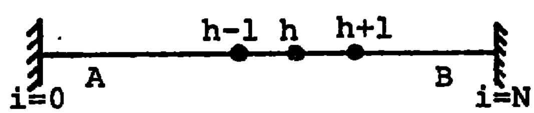
Note, here, that the first order term in Δ, does not cancel out as in the previous case and the final expression can be rewritten as

$$s_1^i = e^{bs}[(1+\frac{b^2\Delta s}{4} + \ldots)(1+\ldots)\ \ldots]$$

Figure 20 shows the computational result verifying that function "R" and its derivative d^2R/ds^2 approach their exact values linearly in Δ.

Case (c) - Consider now the case when one point in the finite difference scheme has a first order error while all other mesh points are formulated in a second order accurate sense. This case is then similar to the original model problem of Section IV where the local truncation error was second order at all points except one where a first order local error was encountered. Through study of the present problem one can more easily see how the introduction of a first order error at a single point does not cause the global truncation error to rise to a first order level.

For this study consider the following mesh system:



where in regions (A) and (B), the governing equation (F1), will be written using a second order accurate central difference expression while at the mesh point $i = h$, a first order accurate version of the governing equation (F1) will be used by again evaluating the source term R between two mesh points. Note that this procedure is analogous to the earlier model problem (Eq. 38) where at the model juncture point the difference equation was first order accurate due to the jump effect while at all other points it was second order accurate. The advantage in the present simpler problem is that following the procedures of cases (a) and (b) a compact analytical error analysis can be made.

The difference equations to be solved are given as

$$R_{i+1} - R_i(2+b^2\Delta^2) + R_{i-1} = 0 \qquad i = 1, \ldots h-1 \qquad \text{(F15)}$$

$$R_{i+1}(2-b^2\Delta^2) - R_i(4+b^2\Delta^2) + 2R_{i-1} = 0 \qquad i = h \qquad \text{(F16)}$$

and

$$R_{i+1} - (2+b^2\Delta^2)R_i + R_{i-1} = 0 \qquad i = h+1, \ldots N-1 \qquad \text{(F17)}$$

subject to the boundary conditions

$$\begin{aligned} i &= 0 \qquad R_o = 0 \\ i &= N \qquad R_N = 1 \end{aligned} \qquad \text{(F18)}$$

A finite difference solution of (F15) between mesh points i=1 to i=h-1 yields the following result

$$R_i = R_h \left[\frac{s_1^i - s_2^i}{s_1^h - s_2^h}\right] \qquad \text{(F19)}$$

whereas between mesh points i=h+1 to i=N-1 one obtains

$$R_i = A\, s_1^i + B\, s_2^i \qquad \text{(F20)}$$

where,

$$A = \frac{R_h}{s_1^h} - \frac{s_2^h}{s_1^h}\{(1-\frac{R_h\, s_1^N}{s_1^h})/(s_2^N - \frac{s_2^h\, s_1^N}{s_1^h})\}$$

$$B = (1 - \frac{R_h\, s_1^N}{s_1^h})/(s_2^N - \frac{s_2^h\, s_1^N}{s_1^h})$$

and

$$s_1 = 1 + b\Delta + \frac{b^2\Delta^2}{2} + \frac{b^3\Delta^3}{8} + \ldots.$$

$$s_2 = 1 - b\Delta + \frac{b^2\Delta^2}{2} - \frac{b^3\Delta^3}{8} + \ldots.$$

Note that both solutions (F19) and (F20) depend on R_h which is still unknown. The difference equation (F16) for the "h" point relates R_h to R_{h+1} and R_{h-1} through the expression

$$R_h = \frac{R_{h+1}(2-b^2\Delta^2)+2R_{h-1}}{(4+b^2\Delta^2)}$$

Since R_{h+1} and R_{h-1} are known in terms of R_h from equations (F19) and (F20), this equation would, after proper manipulation, yield an expression for R_h in terms of known quantities such as b and step size Δ. It is then possible to analytically assess the error term in this expression in the limit of zero step size. Note also, here, that this error would propagate to other mesh points through the solution (F19) ahead of this point and through the solution (F20) behind the mesh points. It is, therefore, necessary to first estimate the error in "R_h" before an attempt is made to analyze the errors at any other points ahead or behind it. R_h can be rewritten in the following form

$$R_h = \frac{1}{2}\left[\frac{R_{h+1}\left(1-\frac{b^2\Delta^2}{2}\right)+R_{h-1}}{(1+b^2\Delta^4/4)}\right]$$

Substituting R_{h+1} from (F20) and R_{h-1} from (F19) and after some manipulations one obtains the following expression:

$$R_h\left[1-\frac{b^2\Delta^2}{2}\ \frac{K_1}{K_2}\ \frac{s_1^h - s_2^h}{s_1^N - s_2^N}\right] = \left(\frac{s_1^h - s_2^h}{s_1^N - s_2^N}\right)\left(1-\frac{b^2\Delta^2}{2}\right)$$

where,

$$K_1 = (s_2^N\ s_1^h - s_2^h\ s_1^N - s_2^N\ s_1^{h+1} + s_1^N\ s_2^{h+1})$$

and

$$K_2 = (s_1^h\ s_2^{h+1} - s_2^h\ s_1^{h+1})$$

Consider now the term

$$I = \frac{b^2\Delta^2}{2}\ \frac{K_1}{K_2}\ \frac{s_1^h - s_2^h}{s_1^N - s_2^N}$$

which can be reformulated as,

$$I = \frac{b^2\Delta^2}{2}\left[\frac{s_2^N}{s_2^h}\left(\frac{1-s_1}{s_2-s_1}\right) - \frac{s_1^N}{s_1^h}\left(\frac{1-s_2}{s_2-s_1}\right)\right]\left(\frac{s_1^h - s_2^h}{s_1^N - s_2^N}\right)$$

Noting that,

$$\frac{1 - s_1}{s_2 - s_1} = +\frac{1}{2} + \frac{b\Delta}{4} - \frac{b^3\Delta^3}{32}$$

and

$$\frac{1 - s_2}{s_2 - s_1} = -\frac{1}{2} + \frac{b\Delta}{4} + \frac{b^3\Delta^3}{32} \quad . . .$$

$$R_h = \left(\frac{s_1^h - s_2^h}{s_1^N - s_2^N}\right)\left[1 - \frac{b^2\Delta^2}{2}\right]\left[1 + \frac{b^2\Delta^2}{2}\left\{\frac{s_2^N}{s_2^h}\left(\frac{1}{2} + \frac{b\Delta}{4}\right.\right.\right.$$

$$\left.\left.- \frac{b^3\Delta^3}{32} \ldots\right) - \frac{s_1^N}{s_1^h}\left(-\frac{1}{2} + \frac{b\Delta}{4} + \frac{b^3\Delta^3}{32} + \ldots\right)\right\}\left(\frac{s_1^h - s_2^h}{s_1^N - s_2^N}\right)$$

$$\left. + \ldots.\right]$$

Expanding this,

$$R_h = \frac{s_1^h - s_2^h}{s_1^N - s_2^N} + \frac{s_1^h - s_2^h}{s_1^N - s_2^N}\,\frac{b^2\Delta^2}{2}\left\{\frac{s_2^N}{s_2^h}\left(\frac{1}{2} + \frac{b\Delta}{4} - \frac{b^3\Delta^3}{32} \ldots\right)\right.$$

$$\left.- \frac{s_1^N}{s_1^h}\left(-\frac{1}{2} + \frac{b\Delta}{4} + \frac{b^3\Delta^3}{32} + \ldots\right)\right\}\left(\frac{s_1^h - s_2^h}{s_1^N - s_2^N}\right)$$

$$- \frac{s_1^h - s_2^h}{s_1^N - s_2^N}\;\frac{b^2\Delta^2}{2} - \frac{b^4\Delta^4}{4}\{\ldots.\} + \ldots.$$

The extraction of the error term from this expression as Δ approaches zero is a process involving further manipulations, however if it is first noted that

$$s_1^h = e^{bs_h} [1 + A\,\Delta^2 + B\,\Delta^3 + \ldots.]$$

Where A and B are known constants, and that a similar statement holds for the terms s_2^h , s_1^N and s_2^N , then it is possible to rewrite R_h as

$$R_h = [\frac{e^{bs_h} - e^{-bs_h}}{e^b - e^{-b}}] + 0(\Delta^2)$$

It is, thus, observed that the first order local truncation error at point "h" contributes to higher order global errors and the difference solution still approaches the exact solution quadratically in Δ. Note also that one can assess the error in R'' from the differential equation

$$(\frac{d^2R}{ds^2})_h = (b^2\ R)_h$$

so that on substituting for R_h, this finite difference solution gives

$$(\frac{d^2R}{ds^2})_h = b^2\ [\frac{e^{bs_h} - e^{-bs_h}}{e^b - e^{-b}}] + b^2 0(\Delta^2)$$

Hence

$$(\frac{d^2R}{ds^2})_h = (\frac{d^2R}{ds^2})_{exact} + b^2\ 0(\Delta^2)$$

indicating that the finite difference second derivative solution would also approach the exact solution quadratically in Δ. It is now possible to direct attention to the analytical difference solutions in regions (A) and (B), ahead and behind of the present mesh point "h". From equation (F19) it is directly seen in the light of previous derivation and the fact that R_h has a second order error associated with it, that the difference solution in region (A) would approach the exact solution as second order accurate in the limit of zero step size, Δ. The same result is also true for the region (B), since this region is written as second order accurate. Figure F-2 shows the computational result for this case where a first order accurate difference equation was used at a point $s = 0.5$ while all other mesh points were written in a second order accurate sense. The function R and its derivative d^2R/ds^2 are seen to approach their exact value as a straight line in the square of the step size, Δ, verifying the present analytical result. There remains now only the question as to when would this first order local truncation error in the numerical scheme produce an explicit first order global error. This point is addressed through Figures F-3 and F-4. It is seen from Figure F-4 that when a computational scheme utilizes the first order difference equation in a fixed region of the entire solution regime, i.e. between $0.3 \leq s \leq 0.7$, the function "R" and its derivative, d^2R/ds^2 approach the exact solution at a point $s=0.5$ linearly in Δ indicating that the overall truncation error is of first order. Figure F-3 shows a similar study where the first order difference equation was only used on a fixed number of points, i.e. $(N)_{0.5} - 3 \leq (N)_{0.5} \leq (N)_{0.5} + 3$, in the entire solution regime. This figure shows that in this case the function "R" and its derivative, d^2R/ds^2 approach the exact solution as second order accurate scheme in the step size indicating that in this case the overall error is of second order. It is, thus, clear that a local error of order Δ will not sum up to a global error of order Δ if it only occurs at a finite number of points as the mesh is refined. The global error will only rise up to first order level if an infinite number of points contribute a first order local error as the mesh is refined.

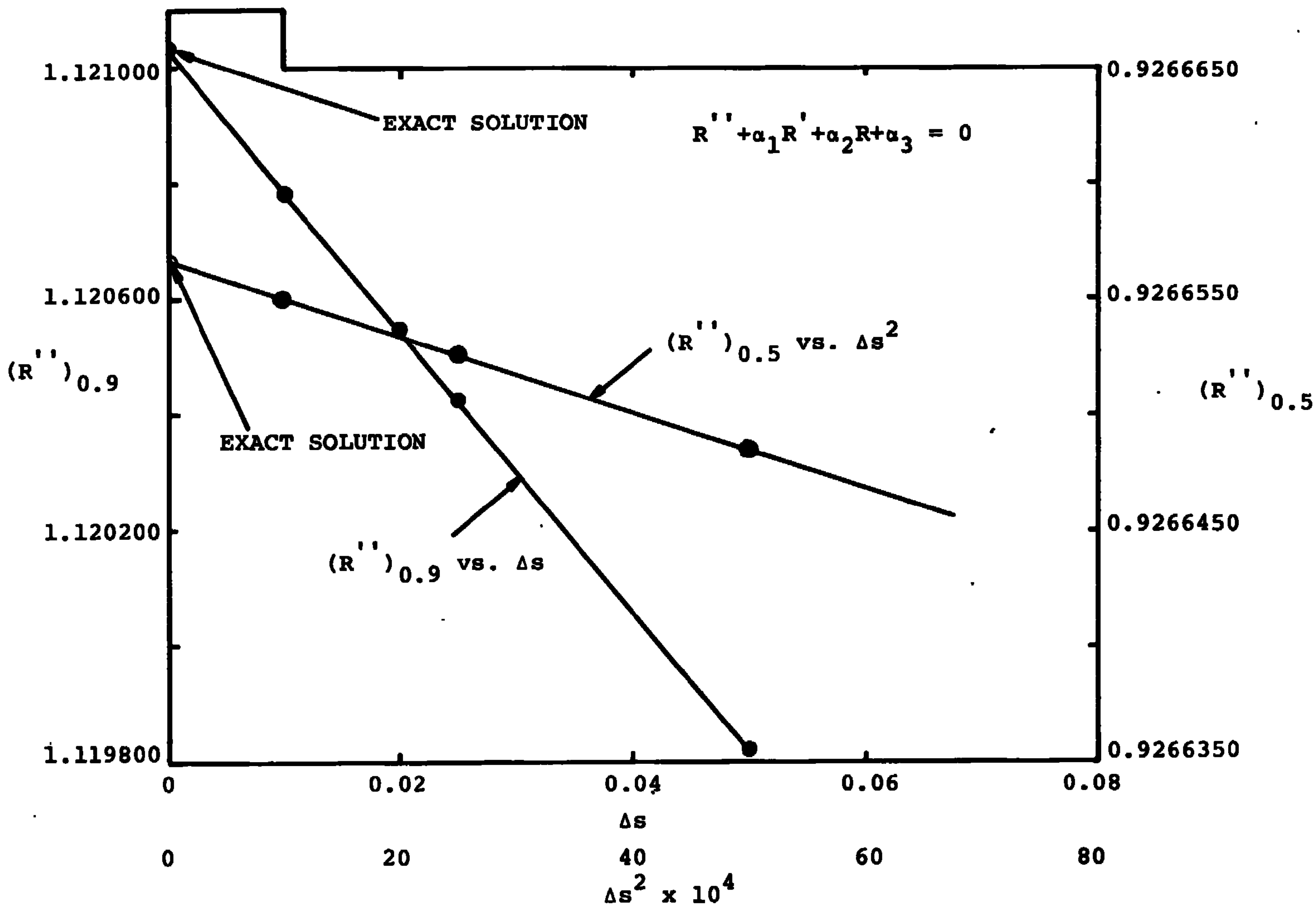


FIGURE F-1 MODEL STEP SIZE STUDY

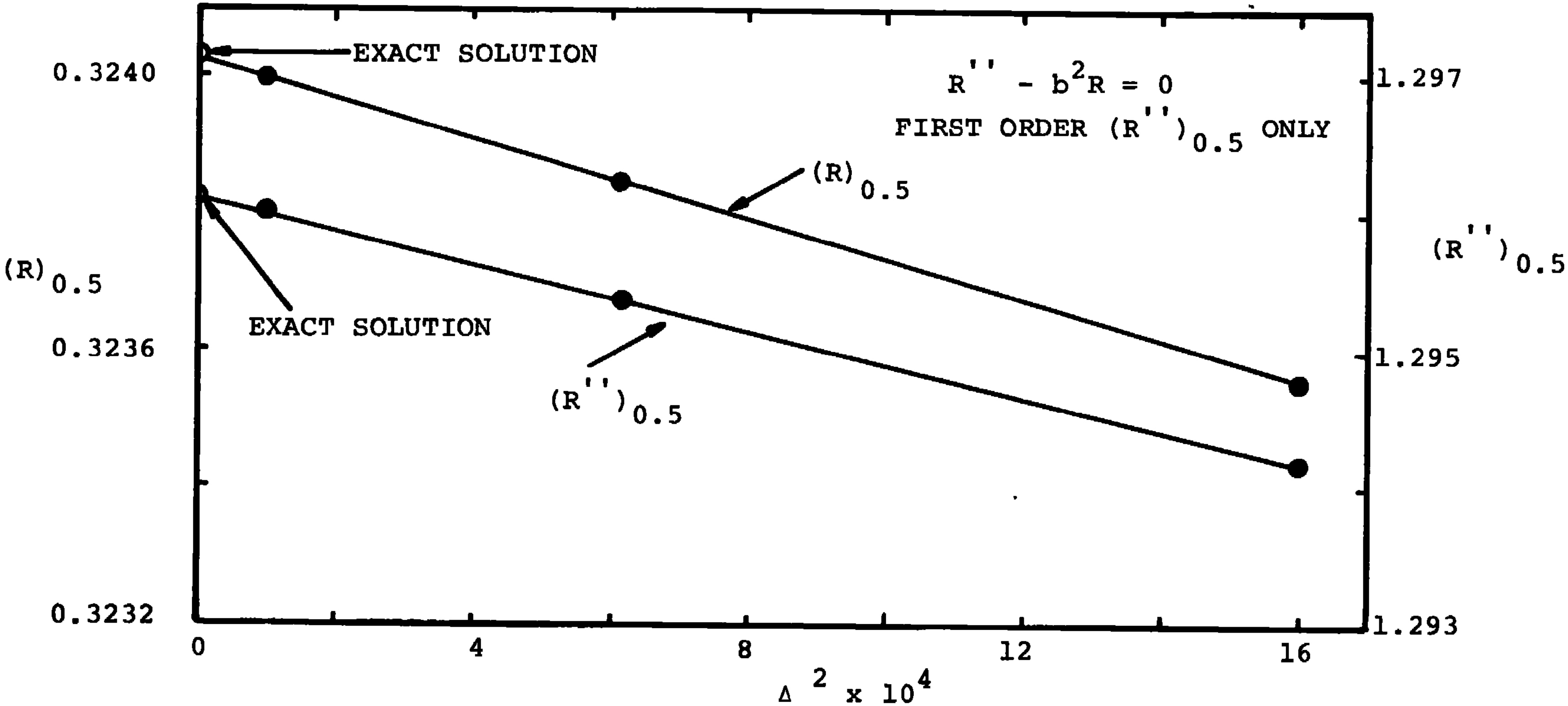


FIGURE F-2 MODEL PROBLEM ERROR STUDY

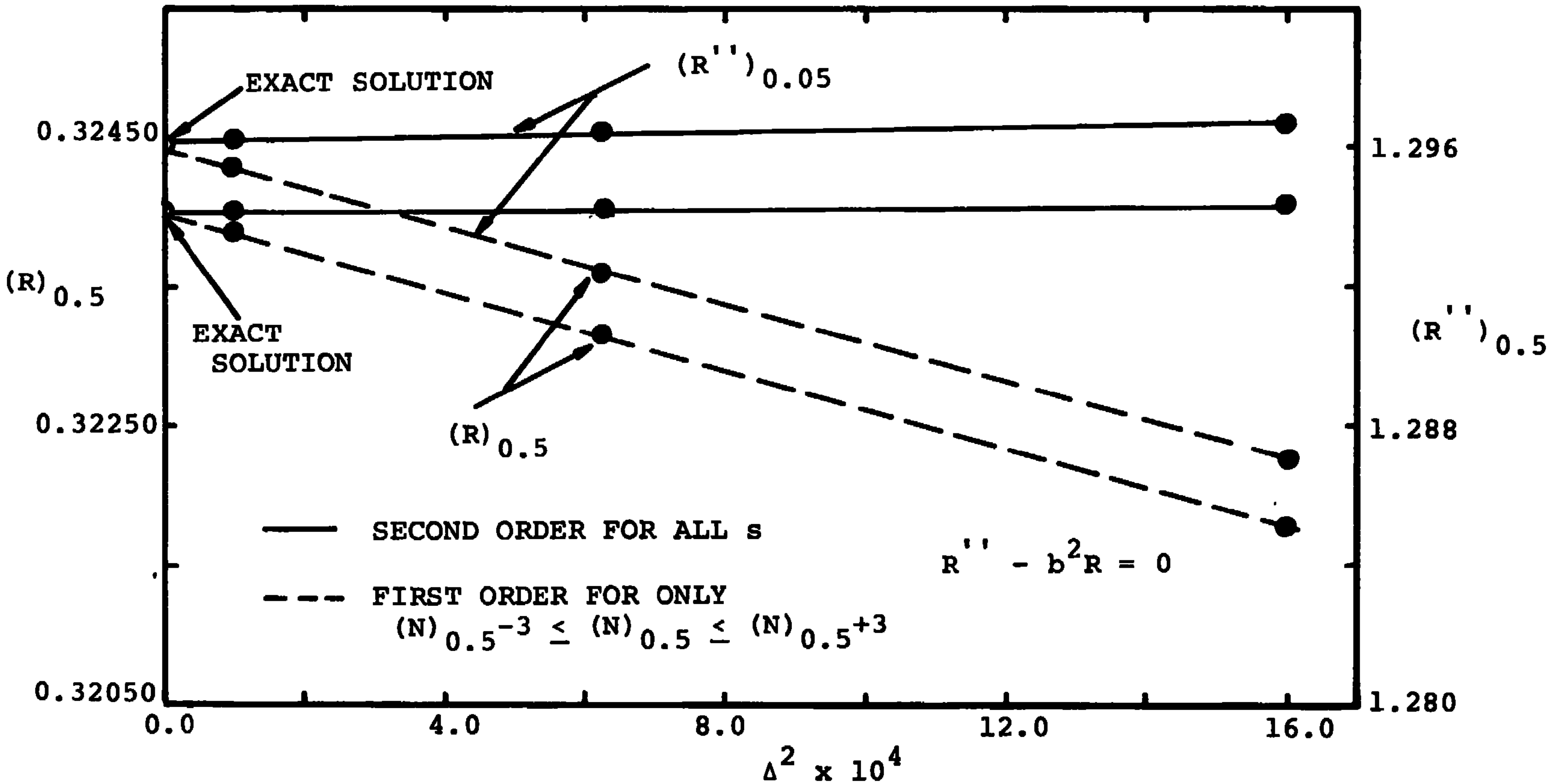


FIGURE F-3 MODEL PROBLEM ERROR STUDY

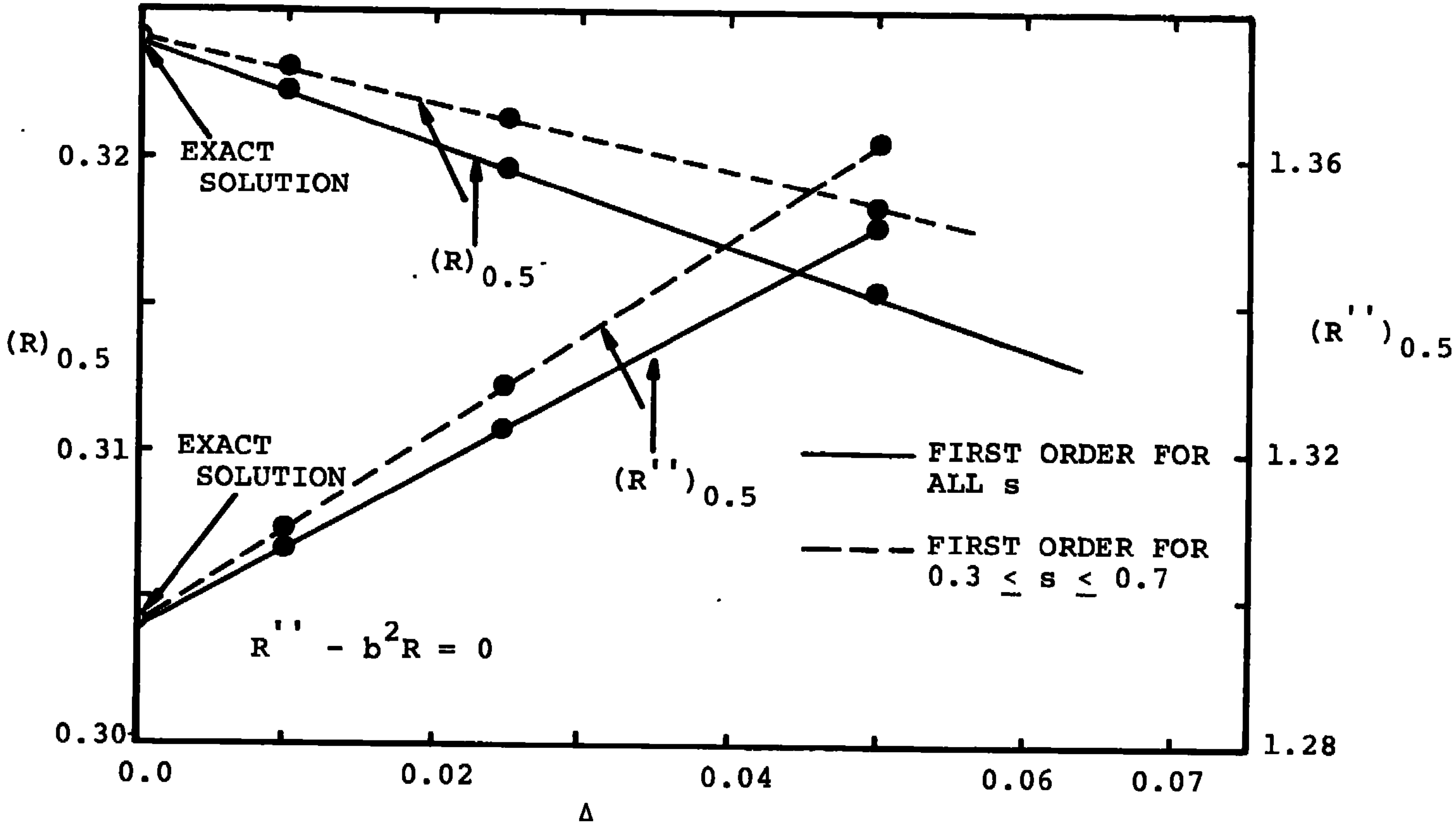


FIGURE F-4 MODEL PROBLEM ERROR STUDY

APPENDIX G

NUMERICAL EVALUATION OF THE JUNCTURE POINT JUMP CONDITIONS

The first and second time sweep of the numerical scheme for the viscous shock layer solution has been discussed in Appendix (D) as represented by equations (D14) and (D15). Both of the time sweep solutions require information about the jumps associated with the terms, $\partial^2 R/\partial s^2$ and $\partial R/\partial s$ at the juncture point. The jump condition associated with the first derivative, dR/ds, is obtained straightforwardly from geometric considerations as

$$\left(\frac{dR}{ds}\right)_{sphere} = (1+n_s)\left(\frac{dR}{ds}\right)_{cone} \tag{G1}$$

However the jump condition associated with, d^2R/ds^2 must be obtained from the momentum equation (D13), which may be rewritten in the form

$$\frac{d^2R}{ds^2} - 2\kappa\,\tan(\alpha-\phi)\,\frac{dR}{ds} + \left[\frac{\bar{u}_{\eta\eta} + \alpha_1\bar{u}_\eta + \gamma_4\bar{u} + \gamma_7 + \alpha_4\bar{u}_\xi}{(\gamma_2\,\bar{u} + \gamma_5)}\right]$$

$$= 0 \tag{G2}$$

Note that the last term in (G2) must be independent of η. Hence, evaluating (G2) on the two sides of the juncture point yields,

$$(R'')_{sphere} - 2\tan(\alpha-\phi)\ (R')_{sphere} + (Fe)_{sphere} = 0 \tag{G3}$$

$$(R'')_{cone} + (Fe)_{cone} = 0 \tag{G4}$$

where,

$$Fe = \left[\frac{\bar{u}_{\eta\eta}+\alpha_1\bar{u}_\eta + \gamma_4\bar{u} + \gamma_7 + \alpha_4\bar{u}_\xi}{(\gamma_2\bar{u} + \gamma_5)}\right] \tag{G5}$$

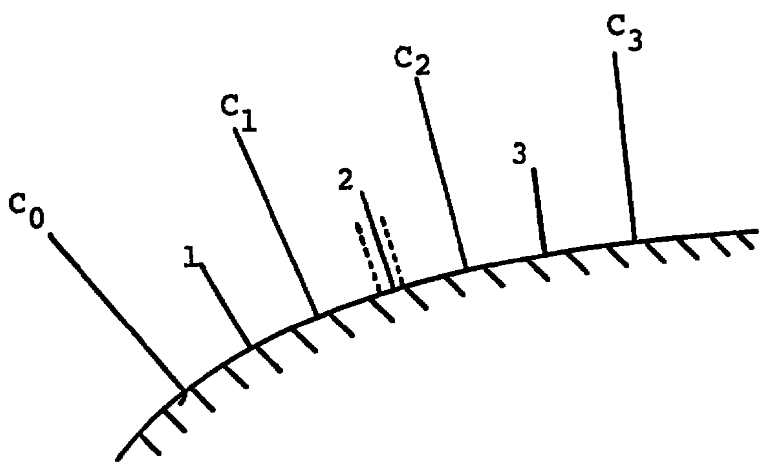
These two can be combined to give

$$(R'')_{sphere} = (R'')_{cone} + 2\tan(\alpha-\phi)(R')_{sphere}$$

$$+ (Fe)_{cone} - (Fe)_{sphere} = 0 \tag{G6}$$

The term "Fe" should be a constant across the shock layer since the associated equation (G2) is independent of the normal coordinate, η. This was verified numerically at every stage of the calculation procedure for the viscous shock layer code. It was also found that there usually was a point in the shock layer where the denominator $(\gamma_2\bar{u} + \gamma_5)$ in equation (G6) would pass through zero. It is, therefore, obvious that at such a point the term "Fe" would be in error due to numerical truncation process and care should be exercised to avoid any such region while evaluating these jump conditions. For the present viscous shock layer code the jump conditions were evaluated at the first grid point away from the wall.

Another difficulty that was encountered in the numerical evaluation of the shock jump conditions stemed from the manner in which the s-momentum equation (D2, Appendix D), was solved in the present form of the viscous shock layer code. The sketch below shows a typical finite difference mesh configuration for the present scheme.



Due to the nature of the ADI algorithm as applied here, it was necessary to solve the star time sweep equations for the flow properties $\bar{u}$, etc. at the numbered points "1", "2", "3" etc., while the final time sweep equations were solved for the shock shape at points C_0, C_1, C_2, C_3 etc. The jump conditions given by equations (G1) and (G6) were to be applied in the middle of the second sweep mesh from points "a" to "b" of the sketch. This then requires that the driving function, Fe, defined in equation (G5) be evaluated at points "a" and "b". To do this properly, the value of Fe at "a" was obtained by extrapolating its values at C_0 and C_1 to "a" while the value at "b" was obtained through extrapolation of Fe's values at points C_2 and C_3. The results thus achieved were found to be consistant throughout the calculations.

APPENDIX H

COMPUTER CODE FOR THE FULL VISCOUS SHOCK LAYER EQUATIONS FOR SPHERICALLY BLUNTED CONES

The following computer code, written in Fortran IV was used to obtain numerical solution of the full viscous shock layer equations for hypersonic flow past spherically blunted cones. The input quantities are:

Main Program

DT	Time step size.
IE	Number of mesh points in the η-direction.
IEND	Number of mesh points in the s-direction.
REYIN	Free stream Reynolds number, Re_∞.
RMAC	Free stream Mach number, M_∞.
BO	Wall to stagnation temperature ratio, T_w/T_o.
TEMP	Free stream temperature, T_∞ in degree Rankine.
GAM	Ratio of specific heats, γ.
SIGM	Prandtl number, σ.
XFACT	Convergence criterion for solving the governing equations by iteration.
THETA1	Sphere/cone angle for which solution is desired, θ.
THETA	Sphere/cone angle whose solution is used as an initial guess on the shock shape.

DTHETA Increment parameter which controls the increment in angle, $\Delta\theta$.

W Under relaxation parameter required during a profile iteration procedure.

WW Relaxation parameter used for two consecutive final sweeps.

J η-point across the shock layer, where shock jump conditions are evaluated, also convergence criteria put for profile iteration.

NJNC Number of mesh points between juncture and stagnation point.

DY Normal step size, $\Delta\eta$.

NITER Number of profile iterations.

NTIME Number of time cycles.

Input Parameters

THIN Positive when thin shock layer equations used.

THINI Negative when full shock layer equations used.

RUMP Positive when jump conditions are included.

SWFAC Positive when wall slip included.

SSFAC Positive when shock slip included.

AHALF Positive when an initial guess for half the longitudinal step size of input guess needed.

Output Quantities

MINF Free stream Mach number.

TW/TO Wall to stagnation temperature ratio, t_w/t_o.

EPS Defined as, $[\mu^*(u_\infty^{*2}/C_p^*)/\rho_\infty^* U_\infty^* a^*]^{1/2}$.

REY(INF) Free stream Reynolds number, Re_∞.

S ξ, surface distance.

X Axial distance measured from nose.

RSH Shock distance measured from axis.

NSH Shock stand off distance normal to body surface.

XSH Shock axial distance measured from body nose point.

R Normal distance to the body surface from the axis.

NSHP $dn_s/d\xi$.

USH u-component of velocity behind the shock.

VSH v-component of velocity behind the shock.

TSH Temperature behind the shock.

RSH Density behind the shock.

PSH Pressure behind the shock.

USP $du_{sh}/d\xi$.

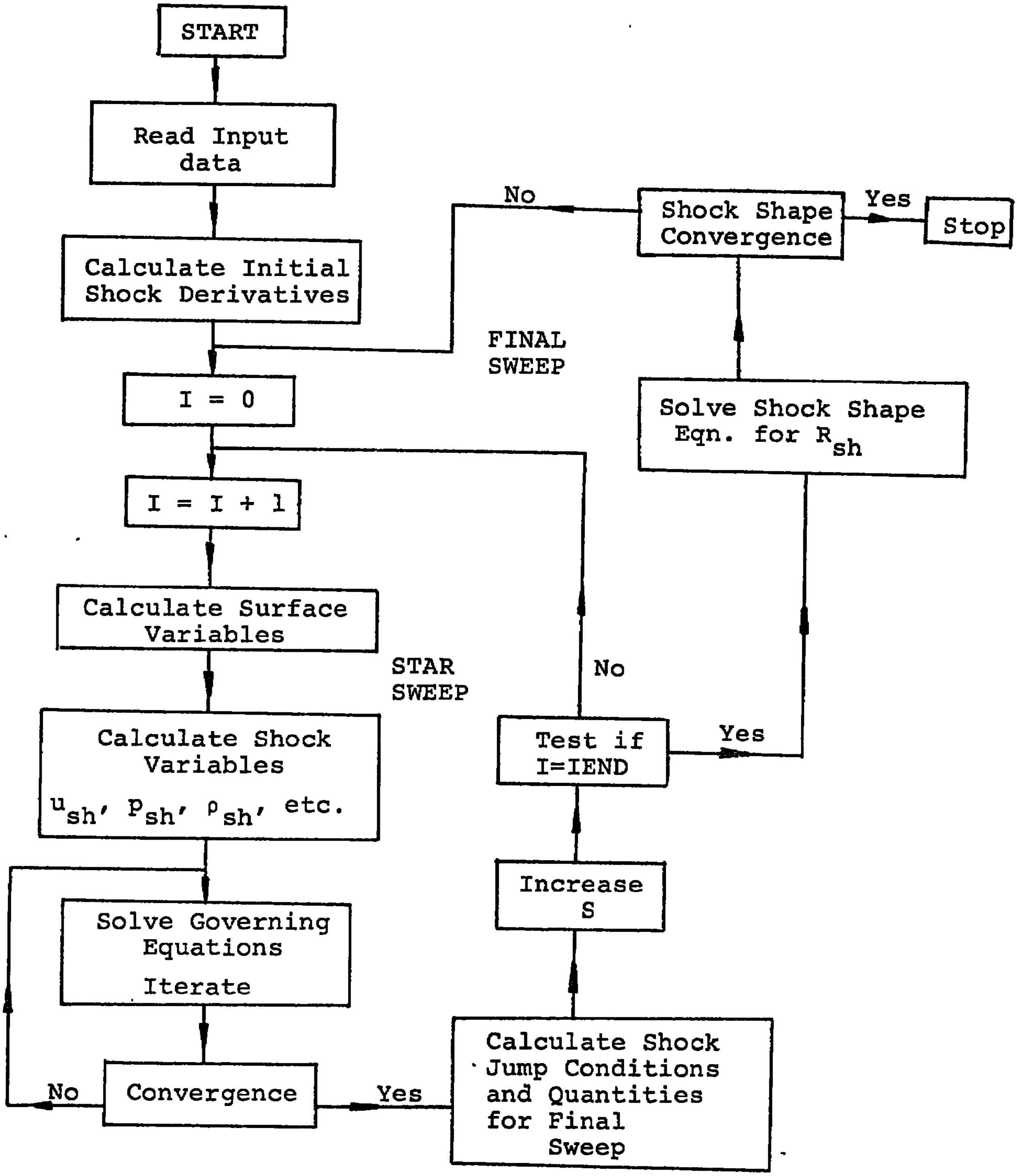
VSP $dv_{sh}/d\xi$.

TSP	$d\tau_{sh}/d\xi$.
RSP	$d\rho_{sh}/d\xi$.
PSP	$dp_{sh}/d\xi$.
PWALL	Pressure at the wall, $p^*/\rho_\infty^* U_\infty^{*2}$.
PW/PO	Pressure ratio at the wall
CF	Skin friction coefficient, $2\tau_w^*/\rho_\infty^* U_\infty^{*2}$.
STAN	Stanton number, $q_w/(H_o-H_w)$.
HEAT	Wall heat transfer, $q_w^*/(\rho_\infty^* U_\infty^{*3})$.

List of Subroutines

Name	Function
DERIV	Calculates the derivatives of the initial shock shape.
GEOM	Calculates body geometry for any given longitudinal location, s.
SHVALS	Calculates properties behind the shock.
MANISH	Utilizes shock jump conditions to evaluate new shock shape, final sweep.
PUSHPA	Evaluates further shock quantities using new shock shape.
PEQSO	Solves tridiagonal difference equation.
BOUND	Provides initial coefficient for derivative boundary conditions.
BOUND1	Provides initial coefficients for derivative boundary condition at s=0 on the shock for final sweep.

Flow Chart for Sphere Cone Program:



```
FORTRAN IV G LEVEL  21                MAIN                DATE = 76296          05/03/31

0001           IMPLICIT REAL*8 (A-H, O-Z)
0002              COMMON/MAIN1/
              1           T1(201),T2(201),TC(201),U1(201),U2(201),UC(201),
              1           V1(201),V2(201),VC(201),P1(201),P2(201),PC(201),
              2           T1N(201),T2N(201),TCN(201),U1N(201),U2N(201),UCN(201),
              3           T1NN(201),T2NN(201),U1NN(201),U2NN(201),PCN(201),
              4           AA(201),BB(201),RVISC(201),CON(201),VISC(201),
              5           RCON(201),CPST(201),RNSH(201),RCSF(201),
              6           R1(201),R2(201),RC(201),P2N(201),P0(201),P0N(201),
              7           PS(201),V2N(201),VS(201),CO1(201),CO2(201),PE(201),
              8           P1N(201),P21(201),P22N(201),P22(201),PFAC(201),P21N(201)
              9                ,VG(201),VGN(201),P33(201),P33N(201),VGS(201),
              1           V0(201),V0N(201),XM(201),PITO(201),UCNN(201)
0003           COMMON /PEQS/  DS,DN(201),IM,IE,A1(201),A2(201),A3(201),A4(201),
              1               XN(202)
0004           COMMON /INSH/ CONO   ,   GAM   ,   S    ,   UPSH   ,   XNS   ,
              1              EPS    ,   RMAC  ,   TPSH  ,   VISCO
0005            DIMENSION                 P2G(201)
0006            DIMENSION     P13N(202)
0007            DIMENSION     PFAM(210)
0008            DIMENSION     C12(210),C11(210)
0009            DIMENSION   CNS2P(40),CNS2PP(40)
0010            DIMENSION        V1G(210)
0011            DIMENSION        V2G(210)
0012            DIMENSION      CNS2(110)
0013               DIMENSION     VCDI(201,5)
0014               DIMENSION     VCD1(201,5)
0015            DIMENSION      YNSH(110),YNSP(110),YNSPP(110)
0016               COMMON/PUSHY/    DERIV1,THMAX
0017           COMMON/OUTSH/ PPS    ,   RRS    ,   TTS   ,   UUS1   ,   VVS   ,
              1              PPS1   ,   RRS1   ,   TTS1  ,   UUS2   ,   VVS1  ,
              2              PSP    ,   RRS2   ,   TSP   ,   USP    ,   VVS2  ,
              3              PPS2   ,   RSP    ,   TTS2  ,   UUS    ,   VSP
0018             COMMON/BASU/       XS3 ,        CONE
0019            COMMON/KINNI/    XNSH(110),XNSP(110),XNSPP(110)
0020            COMMON/MANIS/    AXSH(110),AXSP(110),AXSPP(110)
0021            COMMON/MANU/EE1,FF1,IEND,IEND1,AAA1(110),AAA2(110),AAA3(110)
              1,AAA4(110)
0022            COMMON/CON/    NJNC,NJ1,RUMP
0023            COMMON/MAIN2/     CNS,ALP,CONP ,AKK1,ALP3,PHI3,VVM,AKK2,AKK3,X1SP
              1,PHI,CSF2,RS2,TKK1,TKK2,TKK3
0024            COMMON/MN2/  RE1A,RE1B,RE2A,RE2B,RE3A,RE3B,YNPPJ,YNSPJ,YNSHJ,DT,
              1    RSH1,I,J,XNSPJ,AMM3(202),RSH
0025           DATA BLNK /' '/,BNO/'NO'/
           C COMPUTER CODE FOR SPHERICALLY BLUNTED CONE USING FULL SHOCK LAYER
           C EQUATIONS  (PROGRAMMED BY B.N.SRIVASTAVA)
0026            READ(5,30)  DT,THETA,THETA1,WW,XFACT
0027            READ(5,31)   J,NJNC,IE,IEND
0028            READ(5,32)   RMAC,B0,REYIN,TEMP,GAM,SIGM
0029        30      FORMAT(5F10.6)
0030         31     FORMAT(4I10)
0031         32     FORMAT(6F12.4)
0032            WRITE(6,33)  DT, THETA,THETA1,WW,XFACT,GAM,SIGM,J,NJNC
```

```
0033          33     FORMAT(1H1,45X,10HINPUT DATA,/
                   17F12.6,2I10)
0034                 DTHETA=3.50D0
0035                 NTIME1=0
0036                 NJ1=NJNC-1
0037                 NJ2=NJ1-1
0038                 NJNC1=NJNC+1
0039                 THMAX=THETA*3.14159225D0/180.0D0
0040                 SMAX=3.1415926535897932D0/2.0D0-THMAX
0041                  ANJ1=NJ1
0042                  DS=SMAX/(ANJ1-0.50D0)
0043                 RUMP=1.0D0
0044                 RSH1I=0.0D0
0045                 THSL=1.0000
0046                 THINI=-1.0D0
0047                 AFULL=-1.0D0
0048                 ALSL=1.0000D0
0049                 CONVER=-1.0D0
0050          36      CONTINUE
0051                  THIN=THINI
0052            35 CONTINUE
0053                 TIME=0.0D0
0054                 NTIME=0
0055                 SWFAC=-1.00D0
0056                 SSFAC=-1.00D0
0057                  CALL    DERIV(DS,IEND,IEND1,YNSH,YNSP,YNSPP)
0058                 THMAX=THETA*3.14159225D0/180.0D0
0059                 SMAX=3.1415926535897932D0/2.0D0-THMAX
0060                 ANJ1=NJ1
0061                 DS=SMAX/(ANJ1-0.50D0)
0062                 NJ2=NJ1-1
0063                 NJNC1=NJNC+1
0064          77     CONTINUE
0065                 IF(TIME.GT.35.0D0)     DT=100.0D0
0066                 THMAX=THETA*3.14159225D0/180.0D0
0067                 SMAX=3.1415926535897932D0/2.0D0-THMAX
0068                 DS=SMAX/(ANJ1-0.50D0)
0069          778    CONTINUE
0070                 NTIME1=NTIME1+1
0071                 DO  777     N=1,IEND1
0072                 CNS2(N)=YNSH(N)
0073                 CNS2P(N)=YNSP(N)
0074                 CNS2PP(N)=YNSPP(N)
0075          777    CONTINUE
0076                 DERIV1=-1.0D0
0077                 XNS0=XNSH(1)
0078                  NTIME=NTIME+1
0079                IM=IE-1
0080                XN(1)=0.D0
0081                DO 15 N=1,IE
0082                 IF(XN(N).LE.1.0D0)    DY=0.0350D0
0083                 IF(XN(N).LE.0.6499999D0)     DY=0.0150D0
0084                 IF(XN(N).LE.0.04999999D0)     DY=0.0010D0
```

```
0085                DN(N)=DY
0086             15 XN(N+1)=XN(N)+DN(N)
0087                NITER=0
0088                 RSH=0.0D0
0089                 RSH1=RSH1I
0090                UUS0=0.0
0091                URSH=0.0
0092                UPSH=0.0D0
0093                TPSH=0.0D0
0094                VISCO=0.0D0
0095                CONO=0.0D0
0096                ASL=1.2304*(2.0-THSL)/THSL
0097                BSL=1.1750*(2.0-THSL)/THSL
0098                 CSL=2.3071D0*(2.0D0-ALSL)/ALSL
0099                XNS=XNS0
0100                XNS1=XNS
0101                DS2=DS/2.0D0
0102                CK=1.0D0
0103                CSF=0.0D0
0104                SIF=1.0D0
0105                RS=0.0D0
0106                RS2=0.0D0
0107                XB=0.0
0108                CDF=0.0
0109                CDP=0.0
0110                CDP1=0.
0111                CDP2=0.
0112                CDF1=0.
0113                CDF2=0.
0114                CDPD=0.
0115                CDFD=0.
0116                CNS=(XNS1+XNS)/2.
0117                POIP=((GAM+1.)*RMAC*RMAC  /2.)**(GAM/(GAM-1.))/(GAM*RMAC*RMAC*
                   1     (2.*GAM*RMAC*RMAC/(GAM+1.)-(GAM-1.)/(GAM+1.))**(1./(GAM-1.)))
0118                 TW=B0*(1.0D0/((GAM-1.0D0)*RMAC*RMAC)+0.50D0)
0119                 TB=TW*((GAM-1.0D0)*RMAC*RMAC*TEMP)
0120                CONP=198.6/((GAM-1.)*RMAC*RMAC*TEMP)
0121                VISRA=(1.0+CONP)*(1.0/((GAM-1.)*RMAC*RMAC))**1.5/(1./((GAM-1.)*
                   1     RMAC*RMAC)+CONP)
0122                EPS = 1.0 / DSQRT(REYIN*VISRA)
0123                CALL SHVALS(1.0D0,0.0D0,1.0D0,0.0D0,TTS0,VVS0,UUS0,PPS0,1)
0124                TTS=TTS0
0125                DO 100 N=1,IE
0126                RNSH(N)=CNS/(1.+CK*CNS*XN(N))
0127                RCSF(N)=CNS/(1.+CK*CNS*XN(N))
0128                U1(N)=XN(N)
0129                U2(N)=XN(N)
0130                U1N(N)=1.0
0131                U2N(N)=1.0
0132                U1NN(N)=0.0
0133                UC(N)=XN(N)
0134                UCN(N)=1.0
0135                V1(N)=XN(N)
```

```
            V2(N)=XN(N)
            VC(N)=XN(N)
            T1(N)=1.0-(1.0-XN(N))*(1.0-TW/TTS0)
            T2(N)=T1(N)
            T1N(N)=1.0-TW/TTS0
            T2N(N)=T1N(N)
            T1NN(N)=0.0
            TC(N)=T1(N)
            TCN(N)=T1N(N)
            VISC(N)=(TTS+CONP)*TC(N)**1.5/(TTS*TC(N)+CONP)
            RVISC(N)=(TTS*TC(N)+3.0*CONP)/(2.0*TC(N)*(TTS*TC(N)+CONP))*TCN(N)
            CON(N)=VISC(N)
            RCON(N)=RVISC(N)
            P1(N)=1.0
            P2(N)=1.0
            PC(N)=1.0
            PS(N)=0.0
            PO(N)=1.0
            PON(N)=0.0
            R1(N)=P1(N)/T1(N)
            R2(N)=R1(N)
            RC(N)=R1(N)
            PFAC(N)=1.0
            PCN(N)=0.0
            P1N(N)=0.0
        100 P2N(N)=0.0
            AA(1)=0.0
            BB(1)=0.0
            VISCO=(1.0+CONP)*TTS**1.5/(TTS+CONP)
            CONO=VISCO/SIGM
            DO 5000 I=1,IEND
             CRNI=1.0D0
             IF(I.LT.60)         W=0.8
             IF(I.LT.9)        W=0.60D0
            YI=I
            S=(YI-1.0D0)*DS
            CALL GEOM(S,DS2,RS2,CK2,CSF2,SIF2,XB2)
            PHI = DARCOS(CSF2)
             PHI2=PHI
             IF(I.EQ.1)   PHI1= 3.141592653589793 2D0/2.0D0
              PHIS2=(PHI2-PHI1)/DS*2.0D0
              IF(I.EQ.1)         PHIS=PHIS2
             IF(CONE.LT.0.0D0)      PHIS2=-1.0D0
             IF(CONE.LT.0.0D0)        PHIS=-1.0D0
             IF(CONE.GT.0.0D0)        PHIS2=0.0D0
             IF(CONE.GT.0.0D0)          PHIS=0.0D0
             DXDS1=(AXSP(I)+AXSP(I+1))/2.0D0
             X1SP=XNSP(I)
             XNSPM=(XNSP(I)+XNSP(I+1))/2.0D0
             ALP=DATAN((YNSP(I)+YNSP(I+1))/(2.0D0*DXDS1))
             IF(RUMP.LT.0.0D0)    GO    TO     305
             IF(I.EQ.NJ1)XNSPM=(3.0D0*XNSP(NJ1)-XNSP(NJ2))/2.0D0
             IF(I.EQ.NJ1)  AXSPJ=(3.0D0*AXSP(NJ1)-AXSP(NJ2))/2.0D0
```

```
FORTRAN IV G LEVEL  21                  MAIN                DATE = 76296          05/03/31

            IF(I.EQ.NJ1)  YNPPJ=(3.0D0*YNSPP(NJ1)-YNSPP(NJ2))/2.0D0
            IF(I.EQ.NJ1)     YNSHJ=(3.0D0*YNSH(NJ1)-YNSH(NJ2))/2.0D0
            IF(I.EQ.NJ1)     XNSHJ=(3.0D0*XNSH(NJ1)-XNSH(NJ2))/2.0D0
            IF(I.EQ.NJ1) YNSPJ=(3.0D0*YNSP(NJ1)-YNSP(NJ2))/2.0D0
            IF(I.EQ.NJ1)       ALP=DATAN(YNSPJ/AXSPJ)
      305     CONTINUE
            IF(I.EQ.1)     ALP=(22.0D0/14.0D0+DATAN(YNSP(I+1)/AXSP(I+1)))/2.0
           SP = DSIN(ALP)
           CP = DCOS(ALP)
           SPB=SP*SIF2+CP*CSF2
           CPB=CP*SIF2-SP*CSF2
      2000 CONTINUE
           CALL SHVALS ( SP, CP, SPB, CPB, TTSH, VRSH, URSH, PPSH, 2)
C*******************************************
            IF(I.EQ.1)    ALP3= 3.14159265358979320D0/2.0D0
            IF(I.EQ.1)    PHI3= 3.14159265358979320D0/2.0D0
            BLP=ALP3
            PHP=PHI3
             AK=TTS/PPS
            AR=2.0D0*GAM*RMAC*RMAC*DSIN(BLP)**2.0D0-(GAM-1.0D0)
            AR1=2.0D0*GAM*GAM*RMAC*RMAC*DSIN(2.0D0*BLP)/(AR*(GAM+1.0D0)**1.0
          1D0)
           AR2=4.0D0*GAM*DCOS(BLP)/((GAM+1.0D0)*RMAC**2.0D0*AR*DSIN(BLP)**
          13.0D0)
            AR3=4.0D0*GAM**3.0D0*RMAC**4.0D0*DSIN(2.0D0*BLP)*DSIN(BLP)**2.0D0
          1/(AR**2.0D0*(GAM+1.0D0)**1.0D0)
            AR4=2.0D0*GAM*RMAC**2.0D0*DSIN(2.0D0*BLP)/(AR*AR)*(GAM*(GAM+1.0D0
          1)/(GAM-1.0D0)-2.0D0*GAM*(GAM-1.0D0)/(GAM+1.0D0))
           AR5=4.0D0*GAM*GAM*DSIN(2.0D0*BLP)/((GAM+1.0D0)*AR*AR*DSIN(BLP)**2.
          10D0)
           ARR1=AR1+AR2-AR3-AR4+AR5
           AKK1=-DSIN(2.0D0*BLP-PHP)*(1.0D0-(GAM-1.0D0)*AK/GAM)
          1+DSIN(BLP)*DSIN(BLP-PHP)*(GAM-1.0D0)/GAM*ARR1
           AK11=+DCOS(2.0D0*BLP-PHP)*(1.0D0-(GAM-1.0D0)*AK/GAM)
          1-DSIN(BLP)*DCOS(BLP-PHP)*(GAM-1.0D0)/GAM*ARR1
           AKK2=DCOS(BLP)*DSIN(BLP-PHP)-(GAM-1.0D0)/GAM*AK*DSIN(BLP)*DCOS(BLP
          1-PHP)
            AKKK2=-DCOS(BLP)*DCOS(BLP-PHP)-(GAM-1.0D0)/GAM*AK*DSIN(BLP)*DSIN
          1(BLP-PHP)
            AK6=2.0D0*GAM/((GAM+1.0D0)**2.0D0)*DSIN(2.0D0*BLP)
            AK7=4.0D0*DCOS(BLP)/((GAM+1.0D0)**2.0D0*RMAC**4.0D0*DSIN(BLP)**3.
          10D0)
            AKK6=AK6+AK7
            AKK3=2.0D0/(GAM+1.0D0)*DSIN(2.0D0*BLP)
            IF(I.EQ.1)      GO   TO     302
              DALDS=(DCOS(ALP3-PHI3)**2.0D0/((1.0D0+CK*CNS)*DCOS(PHI3)))
          1*(YNSPP(I)-((RSH-YNSH(I))/DT)*2.0D0)-YNSP(I)*CK*DSIN(ALP3*2.0D0
          2-2.0D0*PHI3)/((1.0D0+CK*CNS)*DCOS(PHI3))
      302     CONTINUE
            IF(I.EQ.1)  DALDS=((XNSPP(1)-((CNS-XNSH(1))/DT)*2.0D0)/(1.0D0+
          1CNS)-1.0D0)
             USP=AKK1*DALDS+AKK2*PHIS
             VSP=AK11*DALDS+AKKK2*PHIS
```

```
FORTRAN IV G LEVEL  21                  MAIN                  DATE = 76296            05/03/31
0228              PSP=AKK3*DALDS
0229              TSP=AKK6*DALDS
0230              RSP=(GAM/(GAM-1.0D0))*(PSP*TTS-TSP*PPS)/(TTS*TTS)
0231              IF(I.EQ.1)   PSP=0.0D0
0232              IF(I.EQ.1)   TSP=0.0D0
0233                 IF(I.EQ.1)          VSP=0.0D0
0234        37     CONTINUE
0235              TLP=ALP
0236              THP=PHI
0237              TK=TTS2/PPS2
0238              TR=2.0D0*GAM*RMAC*RMAC*DSIN(TLP)* DSIN(TLP)-(GAM-1.0D0)
0239              TR1=2.0D0*GAM*GAM*RMAC*RMAC*DSIN(2.0D0*TLP)/(TR*(GAM+1.0D0)**1.0
                1D0)
0240             TR2=4.0D0*GAM*DCOS(TLP)/((GAM+1.0D0)*RMAC**2.0D0*TR*DSIN(TLP)*
                1DSIN(TLP)*DSIN(TLP))
0241              TR3=4.0D0*GAM**3.0D0*RMAC**4.0D0*DSIN(2.0D0*TLP)*DSIN(TLP)**2.0D0
                1/(TR**2.0D0*(GAM+1.0D0)**1.0D0)
0242              TR4=2.0D0*GAM*RMAC**2.0D0*DSIN(2.0D0*TLP)/(TR*TR)*(GAM*(GAM+1.0D0
                1)/(GAM-1.0D0)-2.0D0*GAM*(GAM-1.0D0)/(GAM+1.0D0))
0243             TR5=4.0D0*GAM*GAM*DSIN(2.0D0*TLP)/((GAM+1.0D0)*TR*TR*DSIN(TLP)**2.
                10D0)
0244              TRR1=TR1+TR2-TR3-TR4+TR5
0245            TKK1=-DSIN(2.0D0*TLP-THP)*(1.0D0-(GAM-1.0D0)*TK/GAM)
                1+DSIN(TLP)*DSIN(TLP-THP)*(GAM-1.0D0)/GAM*TRR1
0246             TK11=+DCOS(2.0D0*TLP-THP)*(1.0D0-(GAM-1.0D0)*TK/GAM)
                1-DSIN(TLP)*DCOS(TLP-THP)*(GAM-1.0D0)/GAM*TRR1
0247             TKK2=DCOS(TLP)*DSIN(TLP-THP)-(GAM-1.0D0)/GAM*TK*DSIN(TLP)*DCOS(TLP
                1-THP)
0248              TKKK2=-DCOS(TLP)*DCOS(TLP-THP)-(GAM-1.0D0)/GAM*TK*DSIN(TLP)*DSIN
                1(TLP-THP)
0249              DALDS1=(DCOS(TLP-THP)**2.0D0/((1.0D0+CK2*XNS)*DCOS(THP)))
                1*((YNSPP(I)+YNSPP(I+1))/2.0D0-((RSH1-((YNSH(I)+YNSH(I+1))/2.0D0))
                2/DT)*2.0D0)-CK2*DSIN(TLP*2.0D0-THP*2.0D0)/((1.0D0+CK2*XNS)*
                2DCOS(THP))*((YNSP(I)+YNSP(I+1))/2.0D0)
0250              IF(RUMP.LT.0.0D0)     GO    TO     306
0251              IF(I.EQ.NJ1)
                1 DALDS1=(DCOS(TLP-THP)**2.0D0/((1.0D0+CK*XNS)*DCOS(THP)))
                2*(YNPPJ     -((RSH1-YNSHJ)
                3/DT)*2.0D0)-CK2*DSIN(TLP*2.0D0-THP*2.0D0)/((1.0D0+CK2*XNS)*
                4DCOS(THP))*YNSPJ
0252       306     CONTINUE
0253              VSP1=TK11*DALDS1+TKKK2*PHIS2
          C****************************************
0254              IF(I.EQ.1)     VVM=VVS
0255              IF(I.GT.1)        VVM=1.000
0256             VISCO=(1.0+CONP)*TTS**1.5/(TTS+CONP)
0257             CONO=VISCO/SIGM
0258             REFAC=RRS*VVM*CNS/(EPS*EPS*VISCO)
0259             VIS2=(TTS2+CONP)*T2(1)**1.5/(TTS2*T2(1)+CONP)
0260             XKSL = VIS2*RRS*VVM*DSQRT((GAM-1.0D0)*TTS2*T2(1)/GAM)/
                1               (PPS2*P2(1)*REFAC)
0261             DO 200 N=1,IE
0262             CPST(N)=1.0
```

```
0263              IF (S.GE.0.0001) GO TO 160
0264              PFAC(N)=4.0D0*(P2(N)+(PPS2/PPS0-2.0D0)*P0(N))/(UUS2*DS)
                 1     -XNSP(2)*XN(N)*PON(N)/(2.0D0*UUS2*CNS)
0265              GO TO 200
0266          160  CONTINUE
0267          200 CONTINUE
            C     SOLVE ENERGY EQUATION
0268              DO 500 N=1,IE
0269              A1(N)=REFAC*VISCO*CPST(N)*(UUS*XNSP(I)*RNSH(N)*RC(N)*UC(N)*XN(N)
                 1     /(VVM*CNS)-RC(N)*VC(N))/(CONO*CON(N))+RCON(N)+CK*RNSH(N)
                 2     +RCSF(N)
0270              A4(N)=-REFAC*VISCO*CPST(N)*UUS*RNSH(N)*RC(N)*UC(N)/(VVM*CONO*
                 1     CON(N))
0271              A2(N)=A4(N)*TSP/TTS
0272          500 A3(N)=REFAC*PPS*VISCO*(RNSH(N)*UUS*UUS*UC(N)*PFAC(N)+VVM*VC(N)*
                 1     PCN(N))/(TTS*RRS*VVM*CONO*CON(N))+UUS*UUS*VISC(N)*VISCO*
                 2     (UCN(N)-CK*RNSH(N)*UC(N))**2 /(TTS*CONO*CON(N))
0273              GAMP=GAM+1.0D0
0274              GAMM=GAM-1.0D0
0275              RMACQ=RMAC*RMAC
0276              EPSQ=EPS*EPS
0277              SPQ=SP*SP
0278              FOGQ=4.0D0/(GAMP*GAMP)
0279              DEN=RMACQ*RMACQ*SPQ
0280              CS1=SP*XNS/(EPSQ*CONO)
0281              CS2=-((URSH-CP)**2+FOGQ*GAM*SPQ+(2.0D0/GAMM-FOGQ*GAMM)/RMACQ
                 1  -FOGQ/DEN)*0.5D0*SP*XNS/(EPSQ*CONO*TTS2)
0282              IF (SWFAC) 501,501,502
0283          502 CB1=-1./(CSL*XKSL)
0284              CB2=TW/(TTS2*CSL*XKSL)
0285              CALL BOUND(T1NN,T1N,T1,CB1,CB2,E1,F1,CRNI)
0286              GO TO 503
0287          501 E1=0.0
0288              F1=TW/TTS2
0289          503 CALL PEQSO(T1NN,T1N,T1,T2NN,T2N,T2,E1,F1,CRNI,CS1,CS2,SSFAC,1,0,1)
0290              TTS2G=TTS2
0291              IF (SSFAC) 521,521,522
0292          522 TPSH=T2N(IE)
0293              TTS2=T2(IE)*TTS2G
0294              IF (S.GE.0.0001) GO TO 525
0295              TTS1=TTS2
0296          525 TTS=(TTS2+TTS1)/2.0
0297              DO 524 N=1,IE
0298              T2NN(N)=T2NN(N)*TTS2G/TTS2
0299              T2N(N)=T2N(N)*TTS2G/TTS2
0300          524 T2(N)=T2(N)*TTS2G/TTS2
0301              VISCO=(1.0+CONP)*TTS**1.5/(TTS+CONP)
0302              CONO=VISCO/SIGM
0303              REFAC=RRS*VVM*CNS/(EPS*EPS*VISCO)
0304              GO TO 523
0305          521 TPSH=0.
            C     SOLVE S MOMENTUM EQUATION
0306          523 XU25=U2(15)
```

FORTRAN IV G LEVEL 21 MAIN DATE = 76296 05/03/31

```
0307            XU2S=U2(J)
0308            VISZ=(TTS2+CONP)*T2(1)**1.5/(TTS2*T2(1)+CONP)
0309            XKSL = VIS2*RRS*VVM*DSQRT((GAM-1.0D0)*TTS2*T2(1)/GAM)/
               1                (PPS2*P2(1)*REFAC)
0310            DO 540 N=1,IE
0311            R2(N)=P2(N)/T2(N)
0312            IF (S.GE.0.0001) GO TO 541
0313            R1(N)=R2(N)
0314            T1(N)=T2(N)
0315            T1N(N)=T2N(N)
0316            T1NN(N)=T2NN(N)
0317      541    CONTINUE
0318            TC(N)=(T1(N)+T2(N))/2.
0319            TCN(N)=(T1N(N)+T2N(N))/2.
0320            VISC(N)=(TTS+CONP)*TC(N)**1.5/(TTS*TC(N)+CONP)
0321            RVISC(N)=(TTS*TC(N)+3.0*CONP)/(2.0*TC(N)*(TTS*TC(N)+CONP))*TCN(N)
0322            CON(N)=VISC(N)
0323            RCON(N)=RVISC(N)
0324        540 RC(N)=PC(N)/TC(N)
0325            DO 600 N=1,IE
0326            A1(N)=REFAC*(UUS*XNSP(I)*RNSH(N)*RC(N)*UC(N)*XN(N)/(VVM*CNS)
               1      -RC(N)*VC(N))/VISC(N)+RVISC(N)+CK*RNSH(N)+RCSF(N)
0327            A2(N)=-REFAC*(USP*RNSH(N)*RC(N)*UC(N)/VVM+CK*RNSH(N)*RC(N)
               1      *VC(N))/VISC(N)-CK*RNSH(N)*RVISC(N)-(CK*RNSH(N)+
               2      RCSF(N))*CK*RNSH(N)
0328            A3(N)=-REFAC*PPS*RNSH(N)*PFAC(N)/(VISC(N)*RRS*VVM)
0329        600 A4(N)=-REFAC*UUS*RNSH(N)*RC(N)*UC(N)/(VISC(N)*VVM)
0330            CS1=SP*SPB*UUS2*XNS/(EPS*EPS*VISCO*URSH)
               1   -CK2*XNS/(1.+CK2*XNS)
0331            CS2=-SP*XNS*(CP+VVS2*CPB)/(EPS*EPS*VISCO*URSH)
0332            IF (SWFAC) 601,601,602
0333        602 CB1=-1./(ASL*XKSL)-CK2*XNS
0334            CB2=0.
0335            CALL BOUND(U1NN,U1N,U1,CB1,CB2,E1,F1,CRNI)
0336            GO TO 603
0337        601 E1=0.0
0338            F1=0.0
0339        603 CALL PEQSO(U1NN,U1N,U1,U2NN,U2N,U2,E1,F1,CRNI,CS1,CS2,SSFAC,1.,1)
0340            UUS2G=UUS2
0341            IF (SSFAC) 621,621,622
0342        622 UPSH=U2N(IE)-CK2*XNS*U2(IE)/(1.+CK2*XNS)
0343            UUS2=U2(IE)*UUS2G
0344            IF (S.GE.0.0001) GO TO 625
0345            UUS1=-UUS2
0346        625 UUS=(UUS2+UUS1)/2.0
0347            DO 624 N=1,IE
0348            U2NN(N)=U2NN(N)*UUS2G/UUS2
0349            U2N(N)=U2N(N)*UUS2G/UUS2
0350        624 U2(N)=U2(N)*UUS2G/UUS2
0351            GO TO 623
0352        621 UPSH=0.
          C     SOLVE MASS CONSERVATION EQUATION
0353        623 CONTINUE
```

```
0354          DO 640 N=1,IE
0355          IF (S.GE.0.0001) GO TO 641
0356          U1(N)=U2(N)
0357          U1N(N)=U2N(N)
0358          U1NN(N)=U2NN(N)
0359      641 UC(N)=(U1(N)+U2(N))/2.
0360           UCNN(N)=(U1NN(N)+U2NN(N))/2.0D0
0361      640 UCN(N)=(U1N(N)+U2N(N))/2.
0362           DO   6055        N=1,IE
0363           P2G(N)=P2(N)
0364           V1G(N)=VC(N)
0365           V2G(N)=V2(N)
0366     6055    CONTINUE
0367           XV10=P2(J)
0368           XV50=T2(J)
0369          DO 700 N=2,IE
0370          AA(N)=AA(N-1)+DN(N-1)*(R2(N-1)*U2(N-1)+R2(N)*U2(N))/2.
0371      700 BB(N)=BB(N-1)+DN(N-1)*(R2(N-1)*U2(N-1)*XN(N-1)
             1      +R2(N)*U2(N)*XN(N))/2.
0372          IF (S.GE.0.0001) GO TO 705
0373          AIA=8.*BB(IE)*RRS2*UUS2*CSF2/DS-DS
0374          BIB=4.*AA(IE)*RRS2*UUS2*RS2/DS-DS
0375          CIC=-DS
0376          ROT=BIB*BIB-AIA*CIC
0377          XNS = (-BIB + DSQRT(ROT)) / AIA
0378           X1R=XNS
0379          DO 701 N=1,IE
0380           XNR=XNS
0381          C12(N)=C02(N)
0382          C02(N)=XNR*(RS2*AA(N)+XNR*CSF2*BB(N))*RRS2*UUS2
0383          VC(N)=8.*XNS*(AA(N)*RS2+XNS*BB(N)*CSF2)*RRS2*UUS2/(DS*DS*(1.+
             1      XNS*XN(N))**2 *RC(N))
0384           V2(N)=VC(N)
0385      701 CONTINUE
0386          GO TO 711
0387      705 AIA=BB(IE)*CSF2*RRS2*UUS2
0388          BIB=AA(IE)*RS2*RRS2*UUS2/2.
0389          CIC=-C01(IE)+(RS+CNS*CSF)*((1.+CK*CNS)*RRS*VVS-XNSP(I)*RRS*UUS)*DS
0390          ROT=BIB*BIB-AIA*CIC
0391          XNS = (-BIB + DSQRT(ROT)) / AIA
0392          CIC1=-C11(IE)+(RS2+X1R*CSF2)*((1.+CK2*X1R)*RRS2*VVS2-XNSPM*RRS2
             1*UUS2)*DS
0393           ROT1=BIB*BIB-AIA*CIC1
0394           XN1S=(-BIB+DSQRT(ROT1))/AIA
0395          DO 710 N=1,IE
0396           XNR=XNS
0397          C02(N)=XNR*(RS2*AA(N)+XNR*CSF2*BB(N))*RRS2*UUS2
0398          VI=(C02(N)-C01(N))/DS
0399          VC(N)=-VI/(RRS*VVM*RC(N)*(1.+CK*CNS*XN(N))*(RS+CNS*XN(N)*CSF))+
             1      XNSP(I)*XN(N)*UUS*UC(N)/(VVM*(1.+CK*CNS*XN(N)))
0400           X1R=XN1S
0401          C12(N)=X1R*(RS2*AA(N)+X1R*CSF2*BB(N))*RRS2*UUS2
0402          VM=(C12(N)-C11(N))/DS
```

```
0403              V2(N)=-VM/(RRS2*VVM*R2(N)*(1.+CK2*X1R*XN(N))*(RS2+X1R*XN(N)*CSF2)
                2)+XNSPM*XN(N)*UUS2*U2(N)/(VVM*(1.+CK2*X1R*XN(N)))
0404         145  CONTINUE
0405         710  CONTINUE
0406         711 IF (S.GE.0.0001) GO TO 715
0407              XNS1=XNS
0408         715 CNS=(XNS1+XNS)/2.
0409              RSH1=RS2+XNS*CSF2
0410              IF(I.EQ.1)        RSH1I=RSH1
0411              XSH=XB-CNS*SIF
0412              RSH=RS+CNS*CSF
0413              IF(THIN.GT.0.0D0)        GO   TO   9116
0414        6020    CONTINUE
0415              DO    9117    N=1,IM
0416              VC(N)=(W*V1G(N)+(1.-W)*VC(N))
0417              V2(N)=(W*V2G(N)+(1.-W)*V2(N))
0418        9117  CONTINUE
0419        9116   CONTINUE
0420              IF (THIN.GE.0.0) GO TO 716
0421              VVS2G=VVS
0422              VVS2G=VVS2
0423              VPG=VSP1
0424         716 CONTINUE
0425              DO 712 N=1,IE
0426              RNSH(N)=CNS/(1.+CK*CNS*XN(N))
0427              IF (S.GE.0.0001) GO TO 713
0428              V1(N)=VC(N)
0429              RCSF(N)=CNS/(1.+CK*CNS*XN(N))
0430              GO TO 714
0431         713 RCSF(N)=CSF*CNS/(RS+CNS*XN(N)*CSF)
0432         714   CONTINUE
0433              VS(N)=(V2(N)-V1(N))/DS
0434              IF (THIN.GE.0.0) GO TO 717
0435              VG(N)=V2(N)
0436              VGS(N)=VS(N)
0437              IF(NTIME.EQ.1)        GO   TO   718
0438              IF(I.LE.2)   VGS(N)=(VCDI(N,I+1)-VCDI(N,I))/DS
0439              IF(I.EQ.1)   VGS(N)=(VCD1(N,I+1)-VCD1(N,I))/DS
0440        718        CONTINUE
0441                IF(I.EQ.1)   V0(N)=VC(N)
0442         717 CONTINUE
0443        712    CONTINUE
0444              DO 720 N=2,IM
0445              IF (THIN.GE.0.0) GO TO 720
0446              V0N(N)=(DN(N-1)*V0(N+1)/DN(N)-DN(N)*V0(N-1)/DN(N-1))/(DN(N)+
               1          DN(N-1))+(DN(N)-DN(N-1))*V0(N)/(DN(N)*DN(N-1))
0447              VGN(N)=(DN(N-1)*VG(N+1)/DN(N)-DN(N)*VG(N-1)/DN(N-1))/(DN(N)+
               1          DN(N-1))+(DN(N)-DN(N-1))*VG(N)/(DN(N)*DN(N-1))
0448         720 V2N(N)=(DN(N-1)*V2(N+1)/DN(N)-DN(N)*V2(N-1)/DN(N-1))/(DN(N)+
               1          DN(N-1))+(DN(N)-DN(N-1))*V2(N)/(DN(N)*DN(N-1))
0449              IF (THIN.GE.0.0) GO TO 725
0450              V0N(IE)=V0(IE)*(DN(IM-1)+2.*DN(IM))/(DN(IM)*(DN(IM)+DN(IM-1)))
               1          -V0(IE-1)*(DN(IM-1)+DN(IM))/(DN(IM)*DN(IM-1))
```

```
                  2         +V0(IE-2)*DN(IM)/(DN(IM-1)*(DN(IM)+DN(IM-1)))
0451                 VGN(IE)=VG(IE)*(DN(IM-1)+2.*DN(IM))/(DN(IM)*(DN(IM)+DN(IM-1)))
                  1         -VG(IE-1)*(DN(IM-1)+DN(IM))/(DN(IM)*DN(IM-1))
                  2         +VG(IE-2)*DN(IM)/(DN(IM-1)*(DN(IM)+DN(IM-1)))
0452                 V0N(1)=-V0(1)*(DN(2)+2.*DN(1))/(DN(1)*(DN(1)+DN(2)))
                  1         +V0(2)*(DN(2)+DN(1))/(DN(1)*DN(2))
                  2         -V0(3)*DN(1)/(DN(2)*(DN(1)+DN(2)))
0453                 VGN(1)=-VG(1)*(DN(2)+2.*DN(1))/(DN(1)*(DN(1)+DN(2)))
                  1         +VG(2)*(DN(2)+DN(1))/(DN(1)*DN(2))
                  2         -VG(3)*DN(1)/(DN(2)*(DN(1)+DN(2)))
0454            725 CONTINUE
0455                 V2N(IE)=V2(IE)*(DN(IM-1)+2.*DN(IM))/(DN(IM)*(DN(IM)+DN(IM-1)))
                  1         -V2(IE-1)*(DN(IM-1)+DN(IM))/(DN(IM)*DN(IM-1))
                  2         +V2(IE-2)*DN(IM)/(DN(IM-1)*(DN(IM)+DN(IM-1)))
0456                 V2N(1)=-V2(1)*(DN(2)+2.*DN(1))/(DN(1)*(DN(1)+DN(2)))
                  1         +V2(2)*(DN(2)+DN(1))/(DN(1)*DN(2))
                  2         -V2(3)*DN(1)/(DN(2)*(DN(1)+DN(2)))
           C          SOLVE N MOMENTUM EQUATION
0457                 P21N(IE)=RRS2*UUS2**2 *CK2*XNS/(PPS2*(1.+CK2*XNS))
0458                 P21(IE)=1.0
0459                 P2N(IE)=P21N(IE)
0460                 P2(IE)=1.
0461                 RC(IE)=1.0
0462                 IF (THIN.GE.0.0) GO TO 750
0463                  IF(I.EQ.1)
                    1P33N(IE)=-RRS2*VVS2G*VVS2G*((1.-UUS2*XNSPM/(VVS2G*(1.+CK2*XNS)))*
                    1          VGN(IE)+UUS2*XNS*VPG/VVS2G/(VVS2G*(1.+CK2*XNS)))/PPS2
0464                  IF(I.GT.1)
                    1P33N(IE)=-RRS2*((VVS2G-UUS2*XNSPM/(1.0D0+CK2*XNS))*VGN(IE)
                    2+UUS2*XNS/(1.0D0+CK2*XNS)*VGS(IE))/PPS2
0465                 P33(IE)=0.0
0466                 P2N(IE)=P2N(IE)+P33N(IE)
0467            750 CONTINUE
0468                 PC(IE)=1.
0469                 PCN(IE)=(P1N(IE)+P2N(IE))/2.
0470                 IF (S.LE.0.0001) GO TO 755
0471                 PFAC(IE)=(-XNSP(I)*PCN(IE)/CNS+PSP/PPS)/UUS
0472            755 CONTINUE
0473                 PE(IE)=1.
0474                 PS(IE)=0.
0475                 R2(IE)=1.
0476                 IF (S.GE.0.0001) GO TO 800
0477                 CALL SHVALS(1.0D0,0.0D0,1.0D0,0.0D0,TTS0,VVS0,UUS0,PPS0,1)
0478                 P0N(IE)=0.0
0479                 IF (THIN.GE.0.0) GO TO 760
           C         P0N(IE)=VVSG(1)*V0N(IE)/PPS0
0480                 P0N(IE)= VVS   *V0N(IE)/PPS0
0481            760 CONTINUE
0482                 P0(IE)=1.0
0483                 P1(IE)=1.
0484                 P1N(IE)=P2N(IE)
0485                 R1(IE)=1.
0486                 V1(IE)=1.
```

```
FORTRAN IV G LEVEL  21                  MAIN                  DATE = 76296          05/03/31
 0487         800 KON=IM
 0488             DO 810 N=1,IM
 0489             P21N(KON)=RRS2*UUS2**2 *CK2*XNS*R2(KON)*U2(KON)**2 /(PPS2*(1.+CK2*
               1            XNS*XN(KON)))
 0490             P21(KON)=P21(KON+1)-DN(KON)*(P21N(KON+1)+P21N(KON))/2.
 0491             P2N(KON)=P21N(KON)
 0492              P22(KON)=0.0D0
 0493             P2(KON)=P21(KON)
 0494             IF (THIN.GE.0.0) GO TO 805
 0495              IF(I.EQ.1)
               1P33N(KON)=-RRS2*VVS2G*VVS2G*((R2(KON)*VG(KON)-R2(KON)*U2(KON)*UUS2
               1          *XNSPM*XN(KON)/(VVS2G*(1.+CK2*XNS*XN(KON))))*VGN(KON)
               2          +UUS2*XNS*R2(KON)*U2(KON)*(VGS(KON)+VPG*VG(KON)/VVS2G)
               3          /(VVS2G*(1.+CK2*XNS*XN(KON))))/PPS2
 0496              IF(I.GT.1)
               1P33N(KON)=-RRS2                  *((R2(KON)*VG(KON)-R2(KON)*U2(KON)*UUS2
               1          *XNSPM*XN(KON)/(1.0D0*(1.+CK2*XNS*XN(KON))))*VGN(KON)
               2          +UUS2*XNS*R2(KON)*U2(KON)*(VGS(KON))
               3          /(1.0D0*(1.+CK2*XNS*XN(KON))))/PPS2
 0497             P33(KON)=P33(KON+1)-DN(KON)*(P33N(KON+1)+P33N(KON))/2.
 0498             P2N(KON)=P2N(KON)+P33N(KON)
 0499             P2(KON)=P2(KON)+P33(KON)
 0500              IF(THIN.GT.0.0D0)    GO    TO    9222
 0501              P2(KON)=(W*P2G(KON)+(1.-W)*P2(KON))
 0502        9222 CONTINUE
 0503         805 CONTINUE
 0504             R2(KON)=P2(KON)/T2(KON)
 0505             IF (S.GE.0.0001) GO TO 801
 0506             PON(KON)=0.0
 0507             IF (THIN.GE.0.0) GO TO 807
 0508             PON(KON)= VVS    *P0(KON)*V0(KON)*V0N(KON)/(PPS0*T2(KON))
 0509         807 CONTINUE
 0510             P0(KON)=P0(KON+1)-DN(KON)*(PON(KON+1)+PON(KON))/2.
 0511             P1(KON)=P2(KON)
 0512             P1N(KON)=P2N(KON)
 0513             R1(KON)=R2(KON)
 0514             V1(KON)=V2(KON)
 0515         801 PE(KON)=P21(KON)+P22(KON)
 0516             PC(KON)=(P1(KON)+P2(KON))/2.
 0517             RC(KON)=PC(KON)/TC(KON)
 0518             PCN(KON)=(P1N(KON)+P2N(KON))/2.
 0519             PS(KON)=(P2(KON)-P1(KON))/DS
 0520             IF (S.LE.0.0001) GO TO 810
 0521             PFAC(KON)=(PS(KON)-XNSP(I)*XN(KON)*PCN(KON)/CNS+PSP*PC(KON)/PPS)/
               1            UUS
 0522         810 KON=KON-1
 0523             NITER=NITER+1
 0524              IF(NITER.GT.300)        GO    TO    6000
 0525              TFACT1=XV10-P2(J)
 0526              TFACT2=XV50-T2(J)
 0527              TFACT=XU25-U2(J)
 0528              IF(DABS(TFACT)-XFACT)     822,821,821
 0529         821 GO TO 2000
```

```
      822  CONTINUE
           IF(DABS(TFACT1)-XFACT)   824,821,821
     9777  CONTINUE
      824   CONTINUE
           IF(DABS(TFACT2)-XFACT)   820,821,821
      820  IF(CONVER.GT.0.0D0)    GO   TO   823
           CONVER=1.0D0
             GO   TO    2000
      823  CONTINUE
          IF (S.GE.0.0001) GO TO 830
          TST=(1.+TW/TTSO)/2.0
          VIS4=(TTSO+CONP)*TST**1.5/(TTSO*TST+CONP)
          VIS3=VISCO*VIS4
          REY=1.0/(EPS*EPS*VISCO)
          XKXK=(GAM-1.)/(GAM+1.)*(TTSO/TW+1.)*BO/(2.*EPS*EPS*VIS3)
            IF(AFULL.GT.0.0D0)      GO   TO     202
          GO TO 222
      202  CONTINUE
          CNT=BLNK
          CWS=BLNK
          CSS=BLNK
          IF(THIN .EQ. -1.) CNT=BNO
          IF(SWFAC.EQ. -1.) CWS=BNO
          IF(SSFAC.EQ. -1.) CSS=BNO
          WRITE (6,922) CNT, CWS, CSS, IE, IEND, DS
      922 FORMAT(1H0,3XA2,17H THIN SHOCK LAYER,3X,A2,10H WALL SLIP,3X,A2,
         111H SHOCK SLIP,5X,15HNO STEPS IN N =,I4,
         216H NO STEPS IN S =,I4,5H DS =,F5.3)
          WRITE (6,924) RMAC, BO, EPS, REYIN, REY
      924 FORMAT(1H0,5X4HMINF,7X5HTW/TO,7X3HEPS,7X8HREY(INF),6X6HREY(S)/
         13F12.4,2E13.3)
      222 CONTINUE
      830 REFAC=RRS*VVM*CNS/(EPS*EPS*VISCO)
          CFCH=2.*UUS*RRS*VVM*VISC(1)*(UCN(1)-CK*CNS*UC(1))/REFAC
          HEAT= TTS*RRS*VVM*(CONO*CON(1)*TCN(1)/VISCO+UUS*UUS*VISC(1)*UC(1)
         1     *UCN(1)/TTS)/REFAC
          STAN=HEAT/(0.5+1.0/((GAM-1.0)*RMAC*RMAC)-TW)
            XNSP(I)=(XNS-XNS1)/DS
          DO 840 N=1,IE
          XM(N)=DSQRT((UUS*UUS*UC(N)*UC(N)+VVM*VVM*VC(N)*VC(N))/
         1      ((GAM-1.)*TTS*TC(N)))
          PO2PO1=1.0
          IF (XM(N).LE.1.0) GO TO 845
          PO2PO1=((GAM+1.0)*XM(N)*XM(N)/(2.+(GAM-1.)*XM(N)*XM(N)))**
         1      (GAM/(GAM-1.))/(2.*GAM*XM(N)*XM(N)/(GAM+1.)-(GAM-1.)/(GAM
         2      +1.))**(1./(GAM-1.))
      845 PITO(N)=PO2PO1*PC(N)*PPS*(1.+(GAM-1.)*XM(N)*XM(N)/2.)**(GAM/
         1       (GAM-1.))/POIP
           IF(I.EQ.1)   VC(N)=VC(N)*VVS
           IF(I.EQ.1)    V2(N)=VC(N)
           IF(I.LE.5)        VCDI(N,I)=VC(N)
           IF(I.EQ.1)       VCD1(N,I)=VC(N)/VVS
           IF(I.EQ.2)       VCD1(N,I)=VC(N)/VVS
```

```
                U1(N)=U2(N)
                V1(N)=V2(N)
                T1(N)=T2(N)
                R1(N)=R2(N)
                P1(N)=P2(N)
                T1N(N)=T2N(N)
                T1NN(N)=T2NN(N)
                U1N(N)=U2N(N)
                U1NN(N)=U2NN(N)
                 C11(N)=C12(N)
            840 CO1(N)=CO2(N)
                PWALL=PPS*PC(1)
                IF (SWFAC) 843,843,844
            844 PWALL=PWALL-BSL*EPS**2*DSQRT((GAM-1.0D0)/(GAM*TTS*TC(1)))*VISCO
              1      *VISC(1)*TTS*TCN(1)/CNS
            843 CONTINUE
                IF (S.LE.0.0001) GO TO 841
                CDP2=4.*RS*SIF*PWALL
                CDF2=2.*RS*CSF*CFCH
                CDPD=CDPD+(CDP1+CDP2)*DS/2.
                CDFD=CDFD+(CDF1+CDF2)*DS/2.
                CDP=CDPD/(RS*RS)
                CDF=CDFD/(RS*RS)
            841 IF (S.GE.0.0001) GO TO 842
                PWALO=PWALL
                CDF=0.0
                CDP=2.0*PWALL
            842 CDTOT=CDF+CDP
                CDP1=CDP2
                CDF1=CDF2
                PWRAT=PWALL/PWALO
                XNS1=XNS
                UUS1=UUS2
                VVS1=VVS2
                TTS1=TTS2
                PPS1=PPS2
                RRS1=RRS2
                 IL=IEND-1
                 IF(I.EQ.IEND)     XNS21=RSH1
                 IF(I.EQ.IEND)      XNS20=RSH
                  IF(AFULL.GT.0.0D0)      GO   TO     203
                 GO   TO   223
          203   CONTINUE
                WRITE (6,926) S,X8,RS,CNS,XNSP(I),XSH,RSH,NITER
                WRITE (6,928) UUS, VVS, TTS, RRS, PPS
            928 FORMAT(1H0, 8X3HUSH,10X3HVSH,10X3HTSH,10X3HRSH,10X3HPSH/3X6F13.6)
                 WRITE(6,864)      USP,VSP,TSP,RSP,PSP
           864  FORMAT(1H0, 8X3HUSP,10X3HVSP,10X3HTSP,10X3HRSP,10X3HPSP/3X6F13.6)
                WRITE(6,927) CFCH,HEAT,STAN,CDF,CDP,CDTOT,PWALL,PWRAT
            927 FORMAT(1H0,5X,2HCF,10X4HHEAT,8X4HSTAN,8X3HCDF,9X3HCDP,9X5HCDTOT,7X
              15HPWALL,7X5HPW/P0/8F12.6)
            926 FORMAT(1H0,5X,1HS,11X1HX,11X1HR,11X3HNSH,9X4HNSHP,8X3HXSH,9X3HRSH,
              15X,7HNO ITER/7F12.6,I6)
```

```
  556 FORMAT(3X,3F10.6)
      GO   TO    223
  930 FORMAT(1H0, 8X5HU/USH,8X5HV/VSH,8X5HT/TSH,8X5HR/RSH,4X11HP/PSH(APP
     1R),6X5HP/PSH,8X5HN/NSH,8X4HMACH,9X4HPITO)
  932 FORMAT(3X,9F13.6)
  223 CONTINUE
      IF(I.EQ.NJ1)    AM=1.0D0+XNS
      AAK3=1.0D0
      IF(I.EQ.NJ1)      AAK5=2.0D0*AM*DTAN(ALP-PHI)
      IF(I.EQ.1)   GO   TO     999
      AA3=-YNSPP(I)+2.0D0*CK*DTAN(ALP3-PHI3)*YNSP(I)+2.0D0*(2.0D0*RSH-
     1 YNSH(I))/DT
        AAA1(I)=-2.0D0*CK*DTAN(ALP3-PHI3)
       AAA2(I)=-2.0D0/DT
      AAA3(I)=AA3
       AAA4(I)=0.0D0
      IF(I.EQ.NJ1)  ALP3A=(3.0D0*AAA3(NJ1)-AAA3(NJ1-1))/2.0D0
      IF(I.EQ.NJNC1)  ALP3B=(3.0D0*AAA3(NJNC)-AAA3(NJNC+1))/2.0D0
  999  CONTINUE
      RS=RS2
      S=S+DS2
      CALL GEOM(S,DS2,RS,CK,CSF,SIF,XB)
        PHI3=DARCOS(CSF)
       PHI1=PHI3
        IF(CONE.GT.0.0D0)          PHIS=0.0D0
       IF(CONE.GT.0.0D0)       PHIS2=0.0D0
       IF(CONE.LT.0.0D0)       PHIS=-1.0D0
       IF(CONE.LT.0.0D0)       PHIS2=-1.0D0
      ALP3=DATAN ( YNSP(I+1)/AXSP(I+1))
      SP3=DSIN(ALP3)
      CP3=DCOS(ALP3)
      SPB3=SP3*SIF+CP3*CSF
      CPB3=CP3*SIF-SP3*CSF
      PHIS=(PHI3-PHI)/DS*2.0D0
 5010   FORMAT(6F10.4)
      RS2=RS
      NITER=0
        CONVER=-1.0D0
       IF(I.EQ.1)        CNS0=CNS
 5000 CONTINUE
CN+1   TIME    SWEEP    STARTS
        ENDNSH=(XNS21-XNS20)+XNS21
       CALL   BOUND1(EE1,FF1)
       EE1=0.0D0
        FF1=0.0D0
       AAK4=ALP3B-ALP3A
       IEND1=IEND+1
       CALL   MANISH(YNSPP,YNSP,YNSH,ENDNSH,DS,AAK3,AAK4,AAK5,AM)
        CALL      PUSHPA(YNSH,DS,IEND,YNSP)
       TIME=TIME+DT
       IF(AFULL.GT.0.0D0.AND.THIN.GT.0.0D0)    GO  TO    6005
        IF(THETA.GT.THETA1)      GO TO 8010
       IF(AFULL.GT.0.0D0)    GO    TO    6005
```

```
            DO     8009      N=1,IEND1
            CONV=DABS(YNSH(N)-CNS2(N))
            IF(CONV.GT.0.001D0)     GO     TO     8010
 8009         CONTINUE
            CONV2=DABS(CNS0-XNSH(1))
            GO     TO     18
 8010       CONTINUE
            DO    78     N=1,IEND1
            YNSH(N)=WW*CNS2(N)+YNSH(N)*(1.0-WW)
            YNSP(N)=WW*CNS2P(N)+YNSP(N)*(1.0-WW)
            YNSPP(N)=WW*CNS2PP(N)+YNSPP(N)*(1.0-WW)
   78         CONTINUE
              CALL      PUSHPA(YNSH,DS,IEND,YNSP)
            IF(NTIME1.GT.2.AND.THETA.GT.THETA1)      THETA=THETA -DTHETA
            IF(NTIME1.GT.2.AND.THETA.GT.THETA1)      NTIME1=0
            GO    TO    77
   18       CONTINUE
            IF(AFULL.LT.0.0D0)        NTIME=0
            AFULL=1.0D0
            GO    TO    77
 6005       CONTINUE
            IF(THIN.EQ.-1.0D0)     GO    TO    19
            IEND=IEND-1
            THIN=THINI
            NTIME=0
            AFULL=-1.0D0
            GO    TO    77
 5050       CONTINUE
   19       CONTINUE
            DO  1200   N=1,IEND1
            WRITE(7,1201)     YNSH(N),YNSP(N),YNSPP(N),AXSP(N),XNSH(N),XNSP(N)
 1200       CONTINUE
 1201       FORMAT(6F12.8)
 6000       CONTINUE
            STOP
            END
```

```
FORTRAN IV G LEVEL  21                 PEQSO               DATE = 76296          05/03/31

0001             SUBROUTINE PEQSO(W1NN,W1N,W1,W2NN,W2N,W2,E1,F1,CRNI,CS1,CS2,SSFAC
                1,END,IBACK)
0002             IMPLICIT REAL*8 (A-H, O-Z)
0003             COMMON /PEQS/  DS,DN(201),IM,IE,A1(201),A2(201),A3(201),A4(201),
                1               XN(202)
0004             DIMENSION W1NN(201),W1N(201),W1(201)
0005             DIMENSION E(201),F(201),W2NN(201),W2N(201),W2(201)
0006             E(1)=E1
0007             F(1)=F1
0008             DO 10 N=2,IM
0009             A=(2.0D0-A1(N)*DN(N))/(DN(N-1)*(DN(N)+DN(N-1)))*CRNI
0010             B=((-2.D0+A1(N)*(DN(N)-DN(N-1)))/(DN(N)*DN(N-1))+A2(N))*CRNI
                1  +A4(N)/DS
0011             C=(2.0D0+A1(N)*DN(N-1))/(DN(N)*(DN(N)+DN(N-1)))*CRNI
0012             D=-(W1NN(N)+A1(N)*W1N(N)+A2(N)*W1(N))*(1.0D0-CRNI)
                1  -A3(N)+A4(N)*W1(N)/DS
0013             E(N)=-C/(B+A*E(N-1))
0014          10 F(N)=(D-A*F(N-1))/(B+A*E(N-1))
0015             IF (SSFAC) 11,11,12
0016          12 SK1=CS1+(DN(IM-1)+2.0D0*DN(IM))/(DN(IM)*(DN(IM)+DN(IM-1)))
                1   -(DN(IM-1)+DN(IM))*E(IM)/(DN(IM-1)*DN(IM))
                2   -DN(IM)*(B*E(IM)+C)/(A*DN(IM-1)*(DN(IM)+DN(IM-1)))
0017             SK2=-CS2+(DN(IM-1)+DN(IM))*F(IM)/(DN(IM-1)*DN(IM))
                1   +DN(IM)*(B*F(IM)-D)/(A*DN(IM-1)*(DN(IM-1)+DN(IM)))
0018             W2(IE)=SK2/SK1
0019             GO TO 13
0020         11   W2(IE)=END
0021          13 KON=IM
0022             DO 20 N=2,IE
0023             W2(KON)=E(KON)*W2(KON+1)+F(KON)
0024          20 KON=KON-1
0025              W2(1)=E1*W2(2)+F1
0026              GO   TO   (21,5) , IBACK
0027         21    CONTINUE
0028             DO 30 N=2,IM
0029             W2NN(N)=2.0D0*(W2(N+1)/DN(N)+W2(N-1)/DN(N-1))/(DN(N)+DN(N-1))
                1          -2.0D0*W2(N)/(DN(N)*DN(N-1))
0030          30 W2N(N)=(DN(N-1)*W2(N+1)/DN(N)-DN(N)*W2(N-1)/DN(N-1))/(DN(N)+
                1         DN(N-1))+(DN(N)-DN(N-1))*W2(N)/(DN(N)*DN(N-1))
0031             W2N(1)=-W2(1)*(DN(2)+2.0D0*DN(1))/(DN(1)*(DN(2)+DN(1)))
                1        +W2(2)*(DN(2)+DN(1))/(DN(2)*DN(1))
                2        -W2(3)*DN(1)/(DN(2)*(DN(1)+DN(2)))
0032             W2N(IE)=W2(IE)*(DN(IM-1)+2.0D0*DN(IM))/(DN(IM)*(DN(IM)+DN(IM-1)))
                1          -W2(IE-1)*(DN(IM-1)+DN(IM))/(DN(IM)*DN(IM-1))
                2          +W2(IE-2)*DN(IM)/(DN(IM-1)*(DN(IM)+DN(IM-1)))
0033             W2NN(1)=-W2N(1)*(DN(2)+2.D0*DN(1))/(DN(1)*(DN(2)+DN(1)))
                1          +W2N(2)*(DN(2)+DN(1))/(DN(2)*DN(1))
                2          -W2N(3)*DN(1)/(DN(2)*(DN(1)+DN(2)))
0034             W2NN(IE)=W2N(IE)*(DN(IM-1)+2.D0*DN(IM))/(DN(IM)*(DN(IM)+DN(IM-1)))
                2          +W2N(IE-2)*DN(IM)/(DN(IM-1)*(DN(IM)+DN(IM-1)))
                1          -W2N(IE-1)*(DN(IM-1)+DN(IM))/(DN(IM)*DN(IM-1))
0035              GO   TO    100
0036         5    CONTINUE
```

```
FORTRAN IV G LEVEL  21                    PEQSO               DATE = 76296            05/03/31

                    W2N(1)=0.0000
                    DO      101    N=2,IE
                  W2N(N)=(W2(N)-W2(N-1))/DN(N-1)
                  W2NN(N)=(W2N(N)-W2N(N-1))/DN(N-1)
            101     CONTINUE
                    W2NN(1)=(W2N(2)-W2N(1))/DN(1)
              100 CONTINUE
                  RETURN
                  END
```

```
FORTRAN IV G LEVEL  21                BOUND               DATE = 76296          05/03/31

0001            SUBROUTINE BOUND(W1NN,W1N,W1,CB1,CB2,E1,F1,CRNI)
0002            IMPLICIT REAL*8 (A-H, O-Z)
0003            COMMON /PEQS/  DS,DN(201),IM,IE,A1(201),A2(201),A3(201),A4(201),
               1               XN(202)
0004            DIMENSION W1NN(201),W1N(201),W1(201)
0005            A=(2.0D0-A1(2)*DN(2))/(DN(1)*(DN(2)+DN(1)))*CRNI
0006            B=((-2.D0+A1(2)*(DN(2)-DN(1)))/(DN(2)*DN(1))+A2(2))*CRNI+A4(2)/DS
0007            C=(2.0D0+A1(2)*DN(1))/(DN(2)*(DN(2)+DN(1)))*CRNI
0008            D=-(W1NN(2)+A1(2)*W1N(2)+A2(2)*W1(2))*(1.0D0-CRNI)
               1  -A3(2)+A4(2)*W1(2)/DS
0009            XK1=CB1-(DN(2)+2.0D0*DN(1))/(DN(1)*(DN(2)+DN(1)))+A*DN(1)/(C*DN(2)
               1    *(DN(2)+DN(1)))
0010            XK2=(DN(2)+DN(1))/(DN(2)*DN(1))+B*DN(1)/(C*DN(2)*(DN(2)+DN(1)))
0011            XK3=CB2-D*DN(1)/(C*DN(2)*(DN(2)+DN(1)))
0012            E1=-XK2/XK1
0013            F1=-XK3/XK1
0014            RETURN
0015            END
```

```
FORTRAN IV G LEVEL  21                    GEOM                  DATE = 76296              05/03/31

0001              SUBROUTINE GEOM(S,DS,RS,CK,CSF,SIF,XB)
            C     SPHERE-CONE
0002              IMPLICIT REAL*8 (A-H, O-Z)
0003                COMMON/BASU/      XS3 ,       CONE
0004               COMMON/PUSHY/    DERIV1,THMAX
0005                F=12.50D0
0006              XS=S+DS
0007                XS3=XS
0008               SMAX=3.141592653589793 2D0/2.0D0-THMAX
0009               IF(XS.GT.SMAX)      GO       TO       100
0010              RS=DSIN(XS)
0011               CK=1.0D0-1.0D0/(1.0D0+DEXP(-F*(XS-SMAX)))
0012              CK=1.D0
0013               CONE=-1.0D0
0014              CSF=RS
0015              SIF=DCOS(XS)
0016              XB=1.0D0-DCOS(XS)
0017               GO   TO     200
0018        100     CONTINUE
0019               XS1=XS-SMAX
0020               CSF=DCOS(THMAX)
0021                SIF=DSIN(THMAX)
0022               CK=1.0D0-1.0D0/(1.0D0+DEXP(-F*(XS-SMAX)))
0023               CK=0.0D0
0024               CONE=1.0D0
0025               RS=DSIN(SMAX)+XS1*DSIN(THMAX)
0026               XB=1.0D0-DCOS(SMAX)+XS1*DCOS(THMAX)
0027        300     FORMAT(4F10.5)
0028        200       CONTINUE
0029             RETURN
0030             END
```

```
FORTRAN IV G LEVEL  21                SHVALS              DATE = 76296          05/03/31

            SUBROUTINE SHVALS (SP, CP, SPB, CPB, TTSH, VRSH, URSH, PPSH, ID)
            IMPLICIT REAL*8 (A-H, O-Z)
            COMMON /INSH/ CONO  ,  GAM  ,  S    ,  UPSH  ,  XNS  ,
           1              EPS   ,  RMAC ,  TPSH ,  VISCO
            COMMON/OUTSH/ PPS   ,  RRS  ,  TTS  ,  UUS1  ,  VVS  ,
           1              PPS1  ,  RRS1 ,  TTS1 ,  UUS2  ,  VVS1 ,
           2              PSP   ,  RRS2 ,  TSP  ,  USP   ,  VVS2 ,
           3              PPS2  ,  RSP  ,  TTS2 ,  UUS   ,  VSP
            COMMON /PEQS/  DS,DN(201),IM,IE,A1(201),A2(201),A3(201),A4(201),
           1               XN(202)
            GAMP  = GAM + 1.0D0
            GAMM  = GAM - 1.0D0
            RMACQ = RMAC * RMAC
            FOGQ  = 4.0D0/(GAMP*GAMP)
            SPQ   = SP   * SP
            EPSQ  = EPS  * EPS
            DEN   = RMACQ * RMACQ * SPQ
            URSH = SP * CP / (SP + EPSQ * VISCO * UPSH/XNS)
            TTSH = ((URSH-CP)**2 + FOGQ*GAM*SPQ +(2.0D0/GAMM-FOGQ*GAMM)/RMACQ
           1                   - FOGQ/DEN)*0.5D0*SP/(SP+EPSQ*CONO*TPSH/XNS)
            PPSH =(2.0D0*SPQ - GAMM/(GAM*RMACQ)) / GAMP
            RRSH = GAM * PPSH / (GAMM * TTSH)
            VRSH = -SP / RRSH
            GO TO (20,5) , ID
          5 CONTINUE
            TTS2 = TTSH
            PPS2 = PPSH
            RRS2 = RRSH
            UUS2 =  URSH * SPB + VRSH * CPB
            VVS2 = -URSH * CPB + VRSH * SPB
            IF (S .GE. .0001) GO TO 10
            UUS1 = -UUS2
            VVS1 =  VVS2
            TTS1 =  TTS2
            PPS1 =  PPS2
            RRS1 =  RRS2
      10    CONTINUE
            UUS = (UUS1 + UUS2) / 2.0D0
            VVS = (VVS1 + VVS2) / 2.0D0
            TTS = (TTS1 + TTS2) / 2.0D0
            PPS = (PPS1 + PPS2) / 2.0D0
            RRS = (RRS1 + RRS2) / 2.0D0
            USP = (UUS2 - UUS1) / DS
            VSP = (VVS2 - VVS1) / DS
            TSP = (TTS2 - TTS1) / DS
            PSP = (PPS2 - PPS1) / DS
            RSP = (RRS2 - RRS1) / DS
         20 CONTINUE
            RETURN
            END
```

```
FORTRAN IV G LEVEL  21                  BOUND1                 DATE = 76296            05/03/31

                SUBROUTINE   BOUND1(EE1,FF1)
               IMPLICIT REAL*8 (A-H, O-Z)
               COMMON /PEQS/  DS,DN(201),IM,IE,A1(201),A2(201),A3(201),A4(201),
              1                XN(202)
                A=1.0D0
                B=-2.0D0+DS*DS*A2(2)
                C=1.0D0
                D=-DS*DS*A3(2)
                EE1=(B+4.0D0*C)/(3.0D0*C-A)
                FF1=-D/(3.0D0*C-A)
                RETURN
                END
```

```
            SUBROUTINE    PUSHPA(YNSH,DS,IEND,YNSP)
           IMPLICIT REAL*8 (A-H, O-Z)
            COMMON/KINNI/    XNSH(110),XNSP(110),XNSPP(110)
            COMMON/PUSHY/     DERIV1,THMAX
            COMMON/CON/   NJNC,NJ1,RUMP
            DIMENSION   YNSH(110),YNSP(110)
            DIMENSION      ACKP(110)   ,APH(110)
             COMMON/BASU/      XS3 ,       CONE
            COMMON/MANIS/  AXSH(110),AXSP(110),AXSPP(110)
            XS=0.0D0
            IEND1=IEND+1
            DO    500      I=2,IEND1
            CALL    GEOM (XS,DS,RS,CK,CSF,SIF,XB)
            XNSH(I)=(YNSH(I)-RS)/CSF
            AXSH(I)=XB-XNSH(I)*SIF
          ACKP(I)=(1.0D0+CK*XNSH(I))
            APH(I)=DARCOS(CSF)
             XS=XS3
      500     CONTINUE
            XNSH(1)=(4.0D0*XNSH(2)-XNSH(3))/3.0D0
            AXSH(1)=-XNSH(1)
            DO    12   N=2,IEND
            IF(RUMP.LT.0.0D0)     GO    TO      510
            IF(N.EQ.NJ1)     GO     TO    501
            IF(N.EQ.NJNC)      GO    TO    504
      510    CONTINUE
            AXSP(N)=(AXSH(N+1)-AXSH(N-1))/(2.0D0*DS)
            GO     TO      502
       501     CONTINUE
            AXSP(NJ1)=(3.0D0*AXSH(NJ1)-4.0D0*AXSH(NJ1-1)+AXSH(NJ1-2))/
          1(2.0D0*DS)
            GO    TO      503
      504    CONTINUE
             AXSP(NJNC)=(4.0D0*AXSH(NJNC+1)-AXSH(NJNC+2)-3.0D0*AXSH(NJNC))/
          1(2.0D0*DS)
      503    CONTINUE
       502   CONTINUE
      12    CONTINUE
            XNSP(1)=0.0D0
            AXSP(1)=0.0D0
            AXSP(IEND1)=(3.0D0*AXSH(IEND1)-4.0D0*AXSH(IEND1-1)+AXSH(IEND1-2))
          1/(2.0D0*DS)
             DO      860       I=1,IEND1
            IF(I.EQ.1)    GO    TO     81
            TALP=YNSP(I)/AXSP(I)
            XNSP(I)=ACKP(I)*((TALP-DTAN(APH(I)))/(1.0+TALP*DTAN(APH(I))))
             GO    TO       511
      512      CONTINUE
               IF(I.LT.IEND1)     XNSP(I)=(XNSH(I+1)-XNSH(I-1))/(DS*2.0D0)
              IF(I.EQ.IEND1)    XNSP(I)=(3.0D0*XNSH(IEND1)-4.0D0*XNSH(IEND1-1)
          1+XNSH(IEND1-2))/(2.0D0*DS)
      511      CONTINUE
      81    CONTINUE
```

```
FORTRAN IV G LEVEL  21                    PUSHPA              DATE = 76299          23/14/20

 0050         860    CONTINUE
 0051                XNSPP(1)=(2.0D0*XNSH(1)-5.0D0*XNSH(2)+4.0D0*XNSH(3)-XNSH(4))
                    1/(DS*DS)
 0052                RETURN
 0053                END
```

```
FORTRAN IV G LEVEL  21                DERIV              DATE = 76296          05/03/31

              SUBROUTINE     DERIV(DS,IEND,IEND1,AXNSH,AXNSP,AXNSPP)
              IMPLICIT REAL*8 (A-H, O-Z)
              DIMENSION       X1SP(110),A1SP(110),X1SH(110)
              DIMENSION     YNSH(110),YNSP(110),YNSPP(110)
              DIMENSION      ACKP(110)   ,APH(110)
              COMMON/CON/    NJNC,NJ1,RUMP
               COMMON/BASU/      XS3 ,       CONE
              COMMON/KINNI/     XNSH(110),XNSP(110),XNSPP(110)
              COMMON/MANIS/    AXSH(110),AXSP(110),AXSPP(110)
              COMMON/PUSHY/     DERIV1,THMAX
              DIMENSION   AXNSH(119),AXNSP(119),AXNSPP(119)
              DERIV1=1.0D0
              AHALF=1.0D0
              AHALF=-1.0D0
             IEND1=IEND+1
              READ(5,80)    (XNSH(I),I=1,8)
              READ(5,80)    (XNSH(I),I=9,16)
              READ(5,81)    (XNSH(I),I=17,22)
              READ(5,80)    (XNSH(I),I=23,30)
              READ(5,80)    (XNSH(I),I=31,38)
              READ(5,81)    (XNSH(I),I=39,44)
              READ(5,80)    (XNSH(I),I=45,52)
              READ(5,80)    (XNSH(I),I=53,60)
              READ(5,82)    XNSH(61)
          82   FORMAT(F10.6)
         80   FORMAT(8F10.6)
         81   FORMAT(6F10.6)
              XS=0.0D0
              DO     500      I=2,IEND1
              CALL     GEOM (XS,DS,RS,CK,CSF,SIF,XB)
              YNSH(I)=RS+XNSH(I)*CSF
              AXSH(I)=XB-XNSH(I)*SIF
             ACKP(I)=(1.0D0+CK*XNSH(I))
              APH(I)=DARCOS(CSF)
              IF(I.EQ.2)        YS1=RS
               XS=XS3
        500        CONTINUE
              YNSH(1)=0.0D0
              AXSH(1)=-XNSH(1)
              DO    12    N=2,IEND
              IF(RUMP.LT.0.0D0)      GO    TO      510
              IF(N.EQ.NJ1)      GO     TO    501
              IF(N.EQ.NJNC)      GO    TO     504
        510     CONTINUE
              YNSP(N)=(YNSH(N+1)-YNSH(N-1))/(2.0D0*DS)
              AXSP(N)=(AXSH(N+1)-AXSH(N-1))/(2.0D0*DS)
              GO    TO     502
         501     CONTINUE
              AXSP(NJ1)=(3.0D0*AXSH(NJ1)-4.0D0*AXSH(NJ1-1)+AXSH(NJ1-2))/
            1(2.0D0*DS)
              YNSP(NJ1)=(3.0D0*YNSH(NJ1)-4.0D0*YNSH(NJ1-1)+YNSH(NJ1-2))/
            1(2.0D0*DS)
              GO   TO    503
```

```
FORTRAN IV G LEVEL  21                    DERIV               DATE = 76296          05/03/31
 0052         504     CONTINUE
 0053                 AXSP(NJNC)=(4.0D0*AXSH(NJNC+1)-AXSH(NJNC+2)-3.0D0*AXSH(NJNC))/
                     1(2.0D0*DS)
 0054                 YNSP(NJNC)=(4.0D0*YNSH(NJNC+1)-YNSH(NJNC+2)-3.0D0*YNSH(NJNC))/
                     1(2.0D0*DS)
 0055         503     CONTINUE
 0056          502    CONTINUE
 0057         12     CONTINUE
 0058                 YNSP(1)=(4.0D0*YNSH(2)-YNSH(3)-3.0D0*YNSH(1))/(2.0D0*DS)
 0059                YNSP(1)=XNSH(1)+YS1/DS
 0060                YNSP(1)=XNSH(1)+1.0D0
 0061                XNSP(1)=0.0D0
 0062                AXSP(1)=0.0D0
 0063                AXSP(IEND1)=(3.0D0*AXSH(IEND1)-4.0D0*AXSH(IEND1-1)+AXSH(IEND1-2))
                    1/(2.0D0*DS)
 0064                YNSP(IEND1)=(3.0D0*YNSH(IEND1)-4.0D0*YNSH(IEND1-1)+YNSH(IEND1-2))
                    1/(2.0D0*DS)
 0065                DO    700     I=1,IEND1
 0066                IF(I.EQ.1)       GO   TO     600
 0067                TALP=YNSP(I)/AXSP(I)
            C         IF(RUMP.LT.0.0D0)               GO      TO       512
 0068                XNSP(I)=ACKP(I)*((TALP-DTAN(APH(I)))/(1.0+TALP*DTAN(APH(I))))
 0069                GO    TO      511
 0070         512     CONTINUE
 0071                  IF(I.LT.IEND1)     XNSP(I)=(XNSH(I+1)-XNSH(I))/(2.0D0*DS)
 0072                 IF(I.EQ.IEND1)     XNSP(I)=(3.0D0*XNSH(IEND1)-4.0D0*XNSH(IEND1-1)
                    1+XNSH(IEND1-2))/(2.0D0*DS)
 0073         511     CONTINUE
 0074         600    CONTINUE
 0075         700       CONTINUE
 0076                DO     28        N=2,IEND
            C        IF(RUMP.LT.0.0D0)          GO     TO       530
 0077                IF(N.EQ.NJ1)       GO      TO      531
 0078                IF(N.EQ.NJNC)      GO      TO       532
 0079         530     CONTINUE
 0080                YNSPP(N)=(YNSH(N+1)+YNSH(N-1)-2.0D0*YNSH(N))/(DS*DS)
 0081                GO      TO      533
 0082         531     CONTINUE
 0083                YNSPP(N)=(2.0D0*YNSH(N)+4.0D0*YNSH(N-2)-5.0D0*YNSH(N-1)-YNSH(N-3)
                    1)/(DS*DS)
 0084                GO    TO    533
 0085         532     CONTINUE
 0086                YNSPP(N)=(2.0D0*YNSH(N)-5.0D0*YNSH(N+1)+4.0D0*YNSH(N+2)-YNSH(N+3)
                    1)/(DS*DS)
 0087         533     CONTINUE
 0088         28      CONTINUE
 0089                YNSPP(1)=(4.0D0*YNSP(2)-YNSP(3)-3.0D0*YNSP(1))/(2.0D0*DS)
 0090                YNSPP(1)=0.0D0
 0091                YNSPP(IEND1)=(3.0D0*YNSP(IEND1)-4.0D0*YNSP(IEND1-1)+YNSP(IEND1-2)
                    1)/(2.0D0*DS)
 0092               YNSPP(IEND1)=(2.0D0*YNSH(IEND1)-5.0D0*YNSH(IEND1-1)+4.0D0*YNSH(
                    1IEND1-2)-YNSH(IEND1-3))/(DS*DS)
            C     ******************************************************
```

```
            IF(AHALF.LT.0.0D0)    GO   TO     6006
            APARA=-1.0D0
            APARA=1.0D0
      6019  CONTINUE
            DO   6009   I=1,IEND
            AXNSH(I+I)=(YNSH(I)+YNSH(I+1))/2.0D0
            AXNSP(I+I)=(YNSP(I)+YNSP(I+1))/2.0D0
            AXNSPP(I+I)=(YNSPP(I)+YNSPP(I+1))/2.0D0
            A1SP(I+I)=(AXSP(I)+AXSP(I+1))/2.0D0
            X1SH(I+I)=(XNSH(I)+XNSH(I+1))/2.0D0
            X1SP(I+I)=(XNSP(I)+XNSP(I+1))/2.0D0
      6009   CONTINUE
            NJJ1=NJ1+NJ1
            I2END=2*IEND+1
            II=1
            DO    6007     I=1,I2END,2
            AXNSH(I)=YNSH(II)
            AXNSP(I)=YNSP(II)
            AXNSPP(I)=YNSPP(II)
            X1SH(I)=XNSH(II)
            X1SP(I)=XNSP(II)
            A1SP(I)=AXSP(II)
            II=II+1
      6007   CONTINUE
            NJ1=NJJ1
             NJNC=NJJ1+1
            IEND1=I2END
            IEND=IEND1-1
            DO    6011     N=1,IEND1
             XNSH(N)=X1SH(N)
            AXSP(N)=A1SP(N)
            XNSP(N)=X1SP(N)
      6011  CONTINUE
           AXNSH(NJJ1)=(2.0D0*AXNSH(NJJ1-1)-AXNSH(NJJ1-2))
            AXNSP(NJJ1)=(2.0D0*AXNSP(NJJ1-1)-AXNSP(NJJ1-2))
            AXNSPP(NJJ1)=(2.0D0*AXNSPP(NJJ1-1)-AXNSPP(NJJ1-2))
            XNSH(NJJ1)=(2.0D0*XNSH(NJJ1-1)-XNSH(NJJ1-2))
            XNSP(NJJ1)=(2.0D0*XNSP(NJJ1-1)-XNSP(NJJ1-2))
            AXSP(NJJ1)=(2.0D0*AXSP(NJJ1-1)-AXSP(NJJ1-2))
              IF(APARA.GT.0.0D0)   GO   TO    6020
            DO    6021    I=1,IEND1
            YNSH(I)=AXNSH(I)
            YNSP(I)=AXNSP(I)
            YNSPP(I)=AXNSPP(I)
      6021  CONTINUE
            GO    TO    6019
      6020  CONTINUE.
      6006     CONTINUE
            IF(AHALF.GT.0.0D0)      GO       TO       6010
C     ***************************************************
            DO  1200    N=1,IEND1
              READ(5,1201)      YNSH(N),YNSP(N),YNSPP(N),AXSP(N),XNSH(N),XNSP(N)
      1200  CONTINUE
```

```
FORTRAN IV G LEVEL  21                  DERIV               DATE = 76296          05/03/31

 0145                XNSPP(1)=(2.0D0*XNSH(1)-5.00D0*XNSH(2)+4.0D0*XNSH(3)-XNSH(4))
                    1/(DS*DS)
 0146                DO   6008    I=1,IEND1
 0147                 AXNSH(I)=YNSH(I)
 0148                AXNSP(I)=YNSP(I)
 0149                AXNSPP(I)=YNSPP(I)
 0150          6008     CONTINUE
 0151          6010    CONTINUE
 0152          1201  FORMAT(6F12.8)
 0153                ANSH=YNSH(IEND1)
 0154                RETURN
 0155                  END
```

```
      SUBROUTINE  MANISH(ARXX,ARX,AR,RL,DX,AAK3,AAK4,AAK5,AM)
     IMPLICIT REAL*8 (A-H, O-Z)
      COMMON/MANU/EE1,FF1,IEND,IEND1,AAA1(110),AAA2(110),AAA3(110)
     1,AAA4(110)
      COMMON/CON/   NJNC,NJ1,RUMP
      DIMENSION   ARXX(110),ARX(110),AR(110)
      DIMENSION    E(120),F(120)
      E(1)=EE1
       F(1)=FF1
      XI=0.50D0
      R0=0.0D0
      IEND1=IEND+1
       IM=IEND1-1
      IF(RUMP.GT.0.0D0)    GO    TO    501
      AM=1.0D0
      AAK3=1.0D0
      AAK1=1.0D0
      AAK4=0.0D0
      AAK5=0.0D0
  501  CONTINUE
       DO     10      N=2,IM
      ALP1B=AAA1(N)
      ALP1A=AAA1(N)
      ALP2=AAA2(N)
      ALP3B=AAA3(N)
      ALP3A=AAA3(N)
      AK1=1.0D0
      AK3=1.0D0
      AK4=0.0D0
      AK5=0.0D0
      IF(N.LT.NJ1)     GO     TO     11
      IF(N.GT.NJNC)        GO     TO      11
      AK1=AM
      AK3= AAK3
      AK4=AAK4
      AK5=AAK5
      IF(N.EQ.NJ1)         GO.  TO    12
      AP1=XI*(AK1-DX*XI*AK5/2.0D0 )+(1.0D0-XI)
      AP2=XI*(1.0D0-XI)*(AK1-DX*XI*AK5/2.0D0)+XI*XI*AK3/2.0D0+
     1(1.0D0-XI)**2.0D0/2.0D0
      AP3=AP1+2.0D0*AP2
      A=2.0D0/(DX*DX*AP3)-ALP1B/(DX*AP3)
      B=-2.0D0*(1.0D0+AP1)/(DX*DX*AP3)+ALP1B*(1.0D0-2.0D0*AP2)/(DX*AP3)
     1+ALP2
      C=AP1*2.0D0/(DX*DX*AP3)+2.0D0*ALP1B*AP2/(DX*AP3)
      D=-ALP3B+XI*XI*AK4/AP3-DX*ALP1B*XI*XI*AK4/(2.0D0*AP3)
      GO   TO   13
   12  CONTINUE
      AS=(1.0D0-XI)*(1.0D0-DX*(1.0D0-XI)*AK5/(2.0D0*AK3))/AK1
      AP1=XI+AS
      AP2=XI*XI/2.0D0+XI*AS+(1.0D0-XI)**2.0D0/(2.0D0*AK3)
      AP3=AP1+2.0D0*AP2
      A=AP1*2.0D0/(DX*DX*AP3)-2.0D0*ALP1A*AP2/(DX*AP3)
```

```
FORTRAN IV G LEVEL  21                    MANISH                 DATE = 76296          05/03/31

0051             B=-2.0D0*(1.0D0+AP1)/(DX*DX*AP3)-ALP1A*(1.0D0-2.0D0*AP2)/(DX*AP3)
                1+ALP2
0052             C=2.0D0/(DX*DX*AP3)+ALP1A/(DX*AP3)
0053             A14=AK4/AK3
0054             D=-ALP3A-ALP1A*DX*((1.0D0-XI)**2.0D0)*A14/(2.0D0*AP3)-(1.0D0-XI)
                1**2.0D0*A14/AP3
0055             GO  TO  14
0056        11     CONTINUE
0057             A=1.0D0/(DX*DX)-AAA1(N)/(2.0D0*DX)
0058             B=-2.0D0/(DX*DX)+AAA2(N)
0059             C=1.0D0/(DX*DX)+AAA1(N)/(2.0D0*DX)
0060             D=-AAA3(N)
0061         14    CONTINUE
0062        13    CONTINUE
0063             E(N)=-C/(B+A*E(N-1))
0064             F(N)=(D-A*F(N-1))/(B+A*E(N-1))
0065        10   CONTINUE
0066             KON=IM
0067             AR(IEND1)=RL
0068             AR(1)=R0
0069            DO   20    N=2,IEND1
0070             AR(KON)=E(KON)*AR(KON+1)+F(KON)
0071       20     KON=KON-1
          C   CALCULATION    OF     DERIVATIVES
0072             DO    30    N=2,IM
0073             IF(N.EQ.NJ1)      GO   TO    31
0074             IF(N.EQ.NJNC)               GO    TO     40
0075             ARX(N)=(AR(N+1)-AR(N-1))/(2.0D0*DX)
0076             ARXX(N)=(AR(N+1)+AR(N-1)-2.0D0*AR(N))/(DX*DX)
0077             GO   TO    50
0078        31    CONTINUE
0079             AK1=AM
0080             AK3=AAK3
0081             AK4=AAK4
0082             AK5=AAK5
0083             AS=(1.0D0-XI)*(1.0D0-DX*(1.0D0-XI)*AK5/(2.0D0*AK3))/AK1
0084             AP1=XI+AS
0085             AP2=XI*XI/2.0D0+XI*AS+(1.0D0-XI)**2.0D0/(2.0D0*AK3)
0086             AP3=AP1+2.0D0*AP2
0087             A14=AK4/AK3
0088             ARX(N)=(AR(N+1)-2.0D0*AR(N-1)*AP2-AR(N)*(1.0D0-2.0D0*AP2)+DX*DX
                1*(1.0D0-XI)*(1.0D0-XI)*A14)/(DX*AP3)
0089             ARXX(N)=(AP1*AR(N-1)+AR(N+1)-AR(N)*(AP1+1.0D0)+DX*DX*(1.0D0-XI)*
                1(1.0D0-XI)*A14/2.0D0)/(DX*DX*AP3/2.0D0)
0090             GO   TO   50
0091        40   CONTINUE
0092             AK1=AM
0093             AK3=AAK3
0094             AK4=AAK4
0095             AK5=AAK5
0096             AP1=XI*(AK1-DX*XI*AK5/2.0D0)+(1.0D0-XI)
0097             AP2=XI*(1.0D0-XI)*(AK1-DX*XI*AK5/2.0D0)+ XI*XI*AK3/2.0D0+
                1(1.0D0-XI)**2.0D0/2.0D0
```

```
      AP3=AP1+2.0D0*AP2
      ARX(N)=(2.0D0*AP2*AR(N+1)+AR(N)*(1.0D0-2.0D0*AP2)-AR(N-1)+DX*DX
     1*XI*XI*AK4/2.0D0)/(DX*AP3)
      ARXX(N)=(AR(N-1)+AR(N+1)*AP1-AR(N)*(1.0D0+AP1)-DX*DX*XI*XI*AK4
     1/2.0D0)/(DX*DX*AP3/2.0D0)
   50    CONTINUE
   30  CONTINUE
      ARX(1)=(-3.0D0*AR(1)+4.0D0*AR(2)-AR(3))/(2.0D0*DX)
      ARX(IEND1)=(3.0D0*AR(IEND1)-4.0D0*AR(IEND1-1)+AR(IEND1-2))/
     1(2.0D0*DX)
      ARXX(1)=0.0D0
      ARXX(1)=(2.0D0*AR(1)-5.0D0*AR(2)+4.0D0*AR(3)-AR(4))/(DX*DX)
      ARXX(IEND1)=(2.0D0*AR(IEND1)-5.0D0*AR(IEND1-1)+4.0D0*AR(IEND1-2)
     1-AR(IEND1-3))/(DX*DX)
      RETURN
       END
```

INPUT DATA

```
3.500000   11.000000   7.500000   0.800000   0.000010   1.400000   0.700000   2   12

NO THIN SHOCK LAYER   NO WALL SLIP   NO SHOCK SLIP     NO STEPS IN N = 101 NO STEPS IN S =  20 DS =0.137
```

MINF	TW/TO	EPS	REY(INF)	REY(S)
13.4100	0.0741	0.1049	0.152D 04	0.131D 03

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
0.0	0.0	0.0	0.131292	0.0	-0.131292	0.0	22

USH	VSH	TSH	RSH	PSH
0.0	-0.170417	0.497582	5.837150	0.829846

USP	VSP	TSP	RSP	PSP
0.953739	0.0	0.0	-0.000000	0.0

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.0	0.086606	0.182013	0.0	1.842190	1.842190	0.921095	1.000000

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
0.137133	0.009388	0.136704	0.129787	-0.021954	-0.119180	0.154446	22

USH	VSH	TSH	RSH	PSH
0.118740	-0.166786	0.491211	5.835012	0.818925

USP	VSP	TSP	RSP	PSP
0.874223	0.050466	-0.096666	-0.032436	-0.165709

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.030719	0.087050	0.182947	0.004213	1.788548	1.792761	0.899922	0.977014

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
0.274266	0.037376	0.270840	0.131982	0.053970	-0.089673	0.306586	28

USH	VSH	TSH	RSH	PSH
0.232639	-0.155419	0.473214	5.828668	0.788073

USP	VSP	TSP	RSP	PSP
0.783952	0.116371	-0.165070	-0.059678	-0.282969

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.060506	0.083455	0.175392	0.010444	1.735561	1.746005	0.845565	0.918000

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
0.411399	0.083438	0.399892	0.140863	0.075557	-0.045672	0.456222	31

USH	VSH	TSH	RSH	PSH
0.339555	-0.135769	0.445626	5.817947	0.740780

USP	VSP	TSP	RSP	PSP
0.772023	0.169457	-0.239527	-0.097640	-0.410606

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.081402	0.075554	0.158787	0.019760	1.653118	1.672878	0.761814	0.827075

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
0.548532	0.146709	0.521435	0.153348	0.106525	0.015859	0.601396	33

USH	VSH	TSH	RSH	PSH
0.438613	-0.107593	0.410673	5.802299	0.680864

USP	VSP	TSP	RSP	PSP
0.691323	0.236448	-0.278643	-0.133720	-0.477657

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.097430	0.066936	0.140674	0.031548	1.547861	1.579408	0.654892	0.710993

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
0.685665	0.226002	0.633188	0.171189	0.153678	0.093502	0.741583	33

USH	VSH	TSH	RSH	PSH
0.526864	-0.070203	0.371583	5.781352	0.613854

USP	VSP	TSP	RSP	PSP
0.607613	0.303819	-0.299625	-0.175594	-0.513623

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.104987	0.057308	0.120440	0.044853	1.429514	1.474367	0.538326	0.584442

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
0.822798	0.319827	0.733052	0.196135	0.210140	0.186422	0.876829	32

USH	VSH	TSH	RSH	PSH
0.601265	-0.022700	0.331978	5.755171	0.545963

USP	VSP	TSP	RSP	PSP
0.490385	0.383401	-0.286747	-0.210476	-0.491541

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.104886	0.047424	0.099668	0.058590	1.309062	1.367651	0.423802	0.460106

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
0.959931	0.426424	0.819152	0.230085	0.285009	0.294452	1.007627	31

USH	VSH	TSH	RSH	PSH
0.660195	0.035136	0.294536	5.724023	0.481780

USP	VSP	TSP	RSP	PSP
0.371903	0.459203	-0.261860	-0.244099	-0.448872

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.098341	0.038033	0.079931	0.071925	1.196499	1.268424	0.321772	0.349336

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
1.097064	0.543789	0.889872	0.276469	0.391476	0.417661	1.135894	60

USH	VSH	TSH	RSH	PSH
0.704822	0.101709	0.259543	5.686685	0.421797

USP	VSP	TSP	RSP	PSP
0.263302	0.520587	-0.238522	-0.286226	-0.408860

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.088185	0.029434	0.061860	0.084468	1.099116	1.183583	0.233792	0.253820

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
1.234197	0.669721	0.943883	0.343699	0.589031	0.556204	1.268295	55

USH	VSH	TSH	RSH	PSH
0.738785	0.173704	0.225124	5.638285	0.362799

USP	VSP	TSP	RSP	PSP
0.226261	0.533528	-0.259566	-0.414421	-0.445605

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.075887	0.021742	0.045694	0.096233	1.021446	1.117679	0.159441	0.173100

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
1.371330	0.801854	0.980172	0.439152	0.803091	0.714838	1.410617	46

USH	VSH	TSH	RSH	PSH
0.761990	0.249529	0.191662	5.574810	0.305444

USP	VSP	TSP	RSP	PSP
0.187022	0.534116	-0.275811	-0.606137	-0.472741

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.062026	0.015841	0.033292	0.107395	0.967262	1.074657	0.105711	0.114766

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
1.508463	0.937454	1.000395	0.529161	0.509632	0.868384	1.525029	36

USH	VSH	TSH	RSH	PSH
0.783683	0.280815	0.167276	5.515229	0.263647

USP	VSP	TSP	RSP	PSP
0.189939	-0.110097	-0.118247	-0.340903	-0.202664

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.049193	0.012505	0.026280	0.117948	0.937070	1.055018	0.080785	0.087705

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
1.645596	1.073414	1.018294	0.596364	0.470475	0.995573	1.609556	34

USH	VSH	TSH	RSH	PSH
0.807850	0.265884	0.152159	5.467445	0.237738

USP	VSP	TSP	RSP	PSP
0.158379	-0.106792	-0.099747	-0.347341	-0.170948

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.040721	0.010799	0.022696	0.125728	0.909735	1.035462	0.071915	0.078075

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
1.782729	1.209373	1.036193	0.658261	0.432264	1.123453	1.688823	43

USH	VSH	TSH	RSH	PSH
0.827629	0.251828	0.139646	5.420094	0.216294

USP	VSP	TSP	RSP	PSP
0.127609	-0.097378	-0.081236	-0.335642	-0.139215

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.036198	0.009647	0.020275	0.131422	0.883284	1.014706	0.065597	0.071217

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
1.919862	1.345333	1.054093	0.715520	0.402819	1.251939	1.763491	42

USH	VSH	TSH	RSH	PSH
0.843984	0.238788	0.129191	5.373517	0.198377

USP	VSP	TSP	RSP	PSP
0.106790	-0.090286	-0.068657	-0.331244	-0.117652

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.033008	0.008852	0.018604	0.135843	0.857802	0.993646	0.060989	0.066214

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
2.056995	1.481293	1.071992	0.769285	0.381310	1.380881	1.834696	42

USH	VSH	TSH	RSH	PSH
0.857710	0.226726	0.120330	5.327713	0.183194

USP	VSP	TSP	RSP	PSP
0.091665	-0.084696	-0.059486	-0.330612	-0.101929

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.030649	0.008257	0.017353	0.139349	0.833314	0.972663	0.057375	0.062290

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
2.194128	1.617253	1.089892	0.819539	0.351615	1.510282	1.902419	43

USH	VSH	TSH	RSH	PSH
0.869244	0.215709	0.112817	5.283300	0.170320

USP	VSP	TSP	RSP	PSP
0.077149	-0.077060	-0.050508	-0.319154	-0.086540

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.028774	0.007799	0.016391	0.142159	0.809813	0.951972	0.054544	0.059216

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
2.331261	1.753213	1.107791	0.864317	0.301446	1.640397	1.964714	45

USH	VSH	TSH	RSH	PSH
0.878704	0.206006	0.106604	5.241906	0.159675

USP	VSP	TSP	RSP	PSP
0.059373	-0.063303	-0.039178	-0.277096	-0.067122

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.027154	0.007448	0.015654	0.144409	0.787287	0.931696	0.052554	0.057056

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
2.468394	1.889172	1.125691	0.904733	0.288002	1.771081	2.022684	42

USH	VSH	TSH	RSH	PSH
0.886844	0.197100	0.101215	5.201760	0.150442

USP	VSP	TSP	RSP	PSP
0.050353	-0.056688	-0.033461	-0.262395	-0.057324

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.025854	0.007137	0.014999	0.146203	0.765712	0.911916	0.050898	0.055258

S	X	R	NSH	NSHP	XSH	RSH	NO ITER
2.605527	2.025132	1.143590	0.950673	0.382001	1.901045	2.086130	48

USH	VSH	TSH	RSH	PSH
0.895073	0.187512	0.095721	5.155987	0.141031

USP	VSP	TSP	RSP	PSP
0.067366	-0.080597	-0.045135	-0.395463	-0.077315

CF	HEAT	STAN	CDF	CDP	CDTOT	PWALL	PW/PO
0.025167	0.006817	0.014328	0.147680	0.745016	0.892696	0.048489	0.052643

SYMBOLS

c^*	viscosity law constant, $c^* = 198.6°R$
C_f	skin friction coefficient, $2\tau^*/(\rho_\infty^* u_\infty^{*2})$
C_p^*	specific heat of constant pressure
H	nondimensional total enthalpy, H^*/u_∞^{*2}
k	thermal conductivity
M_∞	free stream Mach number
n	coordinate measured normal to the body, nondimensionalized by the body nose radius
n_s	shock stand off distance normal to the body surface
p	nondimensional pressure, $p^*/(\rho_\infty^* u_\infty^{2*})$
q	nondimensional heat transfer, $q^*/(\rho_\infty^* u_\infty^{*3})$
r	nondimensional axisymmetric radius
R	defined as $y_B + n_s \cos\phi$
s	nondimensional surface distance coordinate
st	Stanton number, $st = q_w/(H_o-H_w)$
T	nondimensional temperature, $T = T^*/(u_\infty^{*2}/C_p^*)$
T_∞^*	free stream temperature
u	nondimensional velocity component tangent to the body surface, u^*/u_∞^* .
u_∞^*	free stream velocity
$\tilde{u}$	nondimensional component of velocity aft and tangent to the shock interface
v	nondimensional velocity component normal to the body surface, v^*/u_∞^*
$\tilde{v}$	nondimensional component of velocity aft and normal to shock interface

x_B axial distance for body surface measured from stagnation point

x_s defined as $x_B - n_s \sin\phi$

y_B normal distance for body surface measured from axis

α shock angle, see Figure 1

β angle defined in Figure 1

γ ratio of specific heats

ε perturbation parameter, $\varepsilon = [\mu^*(u_\infty^{*2}/C_p^*)/\rho_\infty^* u_\infty^* a^*]^{1/2}$

κ nondimensional surface curvature

μ nondimensional coefficient of viscosity, $\mu = \mu^*/\mu^*(u_\infty^{*2}/C_p^*)$

ρ nondimensional density, $\rho = \rho^*/\rho_\infty^*$

ρ_∞^* free stream density

τ nondimensional shear stress, $\tau^*/(\rho_\infty^* u_\infty^{*2})$

ϕ body angle defined in Figure 1

σ Prandtl number, $\sigma = \mu C_p/K$

Subscripts

1 wall value

0 stagnation conditions

s used for longitudinal derivatives

sh conditions immediately behind the shock wave

∞ free stream conditions

Superscripts

-	physical quantities normalized by their shock values
*	dimensional quantities, also used for first sweep of ADI numerical scheme
J	0 for plane flow and 1 for axisymmetric flow
n+1	represents second sweep of ADI numerical scheme