TANNEHILL, John Charles, 1943-
NUMERICAL COMPUTATION OF INTERMEDIATE ALTITUDE ROCKET EXHAUST PLUMES, INCLUDING NONEQUILIBRIUM CHEMICAL REACTIONS AND DIFFUSION.

Iowa State University, Ph.D., 1969
Engineering, aeronautical

University Microfilms, Inc., Ann Arbor, Michigan

NUMERICAL COMPUTATION OF INTERMEDIATE ALTITUDE ROCKET
EXHAUST PLUMES, INCLUDING NONEQUILIBRIUM
CHEMICAL REACTIONS AND DIFFUSION

by

John Charles Tannehill

A Dissertation Submitted to the
Graduate Faculty in Partial Fulfillment of
The Requirements for the Degree of
DOCTOR OF PHILOSOPHY

Major Subjects: Aerospace Engineering
Mechanical Engineering

Approved:

Signature was redacted for privacy.

In Charge of Major Work

Signature was redacted for privacy.

Heads of Major Departments

Signature was redacted for privacy.

Dean of Graduate College

Iowa State University
Ames, Iowa

1969

TABLE OF CONTENTS

LIST OF SYMBOLS

Variables

a_{ij}	=	Enthalpy polynomial constants
A	=	Area of streamtube
A_j	=	Chemical symbol used for species j
b_1	=	Defined by Equation 68
b_2	=	Defined by Equation 68
b_3	=	Defined by Equation 68
b_4	=	Defined by Equation 68
b_5	=	Defined by Equation 68
b_6	=	Defined by Equation 68
c	=	Constant in eddy viscosity expression
c_1	=	Constant in reaction rate equation
c_2	=	Constant in reaction rate equation
c_3	=	Constant in reaction rate equation
c_i	=	Mass fraction of species i
c_{p_f}	=	Frozen specific heat
c_{p_i}	=	Specific heat of species i
D_{ij}	=	Binary diffusion coefficient
h_i	=	Enthalpy of species i
h_i°	=	Heat of formation of species i
H	=	Total enthalpy
j	=	Metric coefficient
$\bar{J}_i$	=	Diffusion mass flux vector of species i
J_i	=	Defined by Equation 56

k	=	Mixture thermal conductivity
k_{b_m}	=	Backward reaction rate
k_{f_m}	=	Forward reaction rate
L	=	Characteristic length
Le	=	Lewis number $\left(\frac{\rho D_{ij} c_{p_f}}{k}\right)$
Le_t	=	Turbulent Lewis number
$\dot{m}$	=	Mass flow rate
M	=	Mach number
M'	=	Number of chemical reactions
M_i	=	Molecular weight of species i
n	=	Coordinate in normal direction
N	=	Number of species
N'	=	Number of species plus number of catalysts
p	=	Pressure
Pr	=	Prandtl number $\left(\frac{c_{p_f} \mu}{k}\right)$
Pr_t	=	Turbulent Prandtl number
$\overline{Q}$	=	Heat conduction vector
Q	=	Defined by Equation 55
r	=	Radius
$\overline{R}$	=	Universal gas constant
Re	=	Reynolds number
s	=	Coordinate in streamwise direction
T	=	Temperature
u	=	Velocity in streamwise direction

$\overline{V}$	=	Velocity vector
$\dot{W}_i$	=	Production rate of species i
[X]	=	Coefficient matrix in Equation 77
[Y]	=	Matrix of unknown molar concentrations in Equation 77
Y_i	=	Molar concentration of species i
Y_c	=	Molar concentration of a catalyst
[Z]	=	Constant matrix in Equation 77
β	=	Defined by Equation 28
β'	=	Defined by Equation 25
γ	=	Ratio of specific heats
δ	=	Small quantity
δ'	=	Shock deflection angle
δ_b	=	Boundary layer thickness
δ_v	=	Viscous layer thickness
$\overline{\overline{\delta}}$	=	Unit tensor
δn	=	Width of streamtube
δs	=	Step in streamwise direction
ΔY	=	Width of mixing layer
$\overline{\overline{\varepsilon}}$	=	Deformation tensor
ε_v	=	Eddy viscosity
ζ	=	Defined by Equation 82
θ	=	Flow angle
θ_n	=	$\partial\theta/\partial n$
θ_s		$\partial\theta/\partial s$
μ	=	Coefficient of viscosity

$\nu'_{j,m}$ = Stoichiometric coefficient of jth species in reaction m in forward direction

$\nu''_{j,m}$ = Stoichiometric coefficient of jth species in reaction m in backward direction

π = 3.14159

ρ = Density

σ = Shock angle

$\bar{\bar{\tau}}$ = Shear stress tensor

τ_{nn} = Shearing stress component given by Equation 22

τ_{ns} = Shearing stress component given by Equation 22

τ_{sn} = Shearing stress component given by Equation 22

τ_{ss} = Shearing stress component given by Equation 22

$\tau_{\omega\omega}$ = Shearing stress component given by Equation 22

Subscripts

b = Boundary condition

k = Condition in streamtube k

k+1 = Condition in streamtube k+1

ℓ = Condition at old surface

ℓ+1 = Condition at new surface

o = Reference value used for nondimensionalizing

s = Condition behind shock

∞ = Freestream condition

Other

$\frac{\partial}{\partial(\ )}$ = Partial differentiation with respect to ( )

O( ) = Of the order ( )

$\Delta_k[\ ]$	=	Takes difference of [ ] across kth streamtube
$(\ )^T$	=	Transpose ( )
Σ	=	Summation
Π	=	Product
$(\bar{\ })$	=	Either nondimensionalized ( ) or the average value of ( ) during step downstream
$(\ )\cdot(\ )$	=	Dot product
$\nabla(\ )$	=	Gradient of ( )
$\nabla\cdot(\ )$	=	Divergence of ( )

INTRODUCTION

Underexpanded Rocket Exhaust Plumes

During the past decade a considerable amount of effort has been expended in attempting to describe analytically the complex flowfield resulting from the interaction of the exhaust gases from a rocket nozzle with the surrounding airstream. In particular, much of this study has been devoted to the situation in which the exit pressure of the nozzle is much higher than the surrounding air pressure. The result of this pressure difference is the characteristic large, billowing plume. The underexpansion occurs because conventional bell-shaped nozzles cannot compensate for the decreasing ambient pressure as the altitude increases. Consequently, nozzles are usually designed to attain an optimum expansion ratio at an altitude which will give the maximum overall performance for the range of altitudes flown. This optimum point frequently occurs at relatively low altitudes. The German V-2 attained an optimum expansion ratio at approximately one mile in altitude (1), while the Thor missile became underexpanded at slightly over two miles in altitude (2).

A typical underexpanded exhaust plume from a single-engined missile is shown in Figure 1. In this case, the external air flow is supersonic and the resulting plume flow is everywhere supersonic except for a small region behind the Mach disc. In the region between the jet shock and the air shock lies a

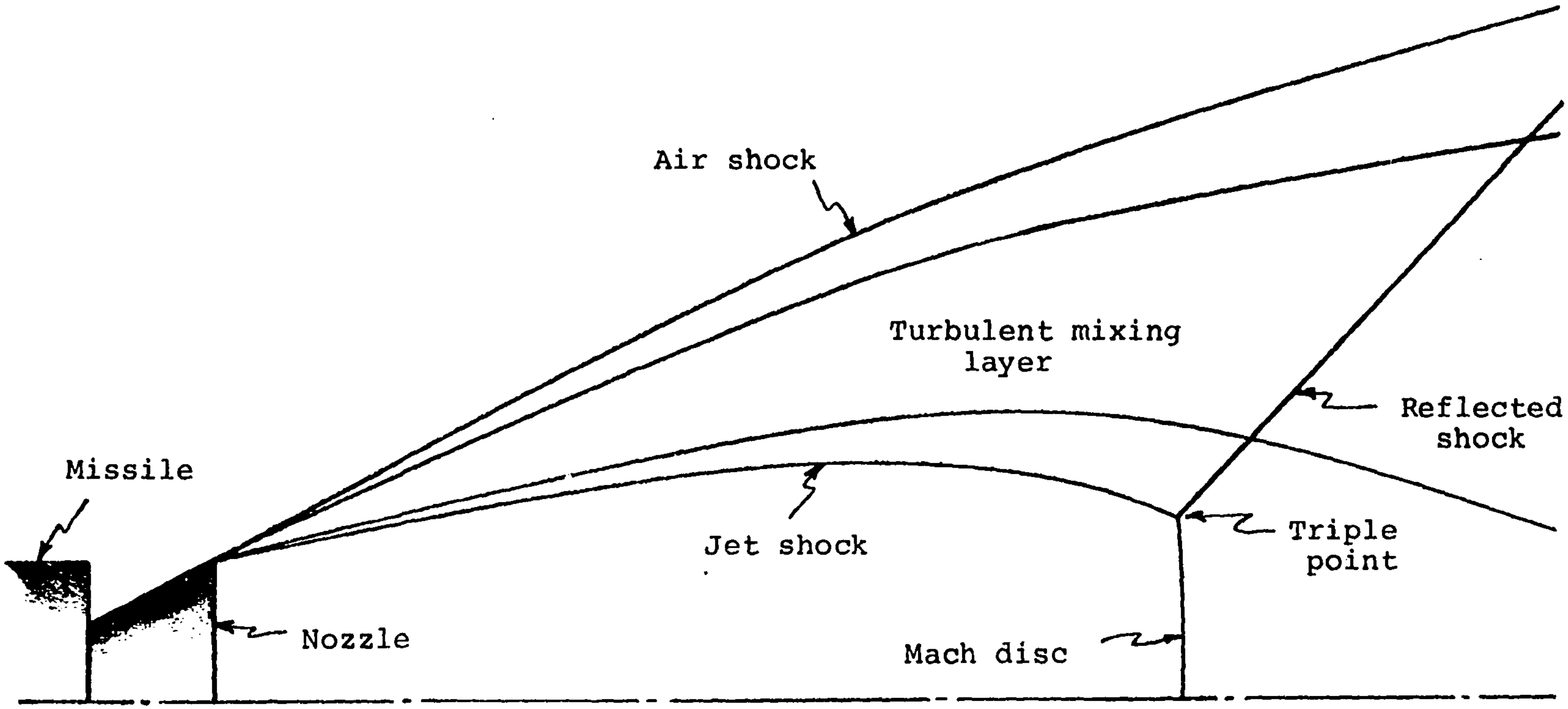

Figure 1. Underexpanded rocket exhaust plume

turbulent mixing layer in which the fuel-rich exhaust products mix with the air and burn. The air shock is the outer limit of the region in which the exhaust flow influences the air flow while the jet shock is at the inner limit. Consequently, the flow in the region bounded by the jet shock and Mach disc is identical to the flow which would result if the nozzle exhaust were expanded to a vacuum (3). Also in this region, the exhaust flow is usually considered to be inviscid and chemically frozen.

The flowfield pictured in Figure 1 is somewhat simplified because the air flow over the missile is usually separated by the time it reaches the rear of the missile. This results in a subsonic region which will affect the initial portion of the plume. Also, shocks which originate from the inside of many contoured nozzles are not pictured nor is the nozzle boundary layer which merges with the turbulent mixing layer in the plume. The thickness of this nozzle boundary layer is frequently increased by turbine exhaust injection.

It should also be mentioned that the plume expansion is not a steady state phenomenon since the missile is accelerating and the atmosphere is not uniform. However, effects due to these last two conditions are usually minor and are thus considered negligible.

At low altitudes where the exhaust flow is only slightly underexpanded and the air flow is still subsonic, the jet shock will intersect the nozzle axis and no Mach disc will be formed.

The jet shock will then be reflected alternately from the nozzle axis and the plume boundary resulting in the familiar shock diamond pattern. At these low altitudes, the assumption of chemical equilibrium can be used to describe the behavior of the major species in the turbulent mixing layer.

As the altitude increases to the intermediate altitude region, the Mach disc appears and increases in size. Likewise, as the external flow becomes supersonic, the air shock appears and remains attached to the edge of the nozzle until the plume expands to such a size as to cause detachment. When this occurs, a subsonic region is formed at the bow of the plume. Also, when the external flow becomes supersonic, the reflected shock from the Mach disc will pass through the mixing layer and will be only partially reflected back toward the nozzle axis. At these intermediate altitudes, the chemical reaction times are of the same order of magnitude as particle travel times so that nonequilibrium chemical reactions must be considered in the turbulent mixing layer.

Above approximately 300,000 feet, in the high altitude region, the assumption of frozen flow can be made everywhere in the plume flowfield. At such high altitudes, the transition to turbulence is delayed so that only laminar mixing need be considered (4).

Problems Concerned With Rocket Exhaust Plumes

The recent interest in rocket exhaust plumes has been stimulated by the many design problems which arise when the exhaust gases leave the nozzle. For example, the effects of the plume impinging on other vehicles , ground surfaces, and even on its own vehicle at very high altitudes has received considerable attention. Included in this area of study is the backflow resulting from the intersection of plumes from a multi-engined booster.

Another problem of great concern is the severe attenuation of communication signals between a space vehicle and ground control caused by the presence of free electrons in the rocket exhaust plume. Also, when the exhaust is highly underexpanded and the resulting plume is large, the air flow over the missile afterbody has been observed to separate causing possible unstable aerodynamic characteristics if control surfaces are located at the rear of the missile. And finally, the effects of radiation from the afterburning in the exhaust plume has received attention not only because of base heating of the missile but also because the infrared components of the radiation contribute to the ease of detecting the missile. Understanding of all of these areas requires a knowledge of the exhaust plume flowfield. Furthermore, it has been shown for the radar attenuation problem that the electron producing reactions are of little importance in determining the macroscopic flow properties in the exhaust plume (5). The same holds true for the infrared radiation studies since the radiation emitted will not effect, to any extent, the flow properties in the plume. This allows one to uncouple the flowfield calculation from the electron density predictions and infrared radiation calculations.

Review of Previous Work

With the advent of the high speed computer in the late 1950's, the numerical computation of the complex flowfield in a rocket exhaust plume became feasible. Since that time a wealth of information has been generated with the aid of the computer. No attempt will be made here to list all reports that have been written on the subject and only the more important ones will be discussed. Three papers in particular give excellent surveys of the literature. First of all, Adamson (4) has a fairly extensive bibliography of reports and in his paper he discusses some of the work done prior to 1959. Vick *et al*. (6) include a survey of the work done prior to 1964 while Farmer *et al*. (7) include reports up to 1966.

One of the initial assumptions that can be made to make the solution of exhaust plume flowfields tractable is to allow no mixing between the exhaust gases and the air. Then, in lieu of the mixing region, a dividing streamline is used to separate the two flows. The pressure on either side of this dividing streamline is usually set equal to either the ambient air pressure for quiescent external conditions, or the Newtonian shock layer pressure for hypersonic external air flows. Then, conventional computation methods such as the method of characteristics or the Lagrangian finite-difference technique can be applied to all supersonic portions of the flowfield.

In the method of characteristics solution, the flow at the lip of the nozzle is expanded through a Prandtl-Meyer expansion fan to the pressure calculated at the dividing streamline. After the exhaust flow passes through the expansion fan it is turned by the dividing streamline causing characteristics of the same family, reflected from the dividing streamline, to converge and form the jet shock. Obviously, the jet shock is of infinitesimal strength at the nozzle lip and increases in strength as it moves downstream where its diameter reaches a maximum and thereafter decreases. The method of characteristics technique will not predict the appearance of a Mach disc so the jet shock will decrease in diameter until it intersects the nozzle axis. Consequently, for highly underexpanded plumes, where the Mach disc is present, the method of characteristics will give erroneous results because in actuality there is a subsonic region formed behind the Mach disc and a shock is reflected toward the plume boundary from the point where the jet shock intersects the Mach disc. This point is called the triple point.

Three empirical techniques have been proposed to locate the position of the Mach disc. Adamson and Nicholls (8) have predicted the location of the Mach disc by specifying the pressure behind it. They have shown that this pressure varies from 1.0

to 1.3 times the ambient pressure for underexpanded jets exhausting to quiescent conditions. The second method, proposed by Eastman and Radtke (9), locates the Mach disc at the point where the pressure downstream of the jet shock reaches a minimum. And finally, the Mach disc has been located by specifying the angle of the Mach disc at the triple point. For example, Bowyer et al. (10, 11) assume that the Mach disc is normal to the incident flow at the triple point.

Once the Mach disc has been located, the computation can be restarted using a slip line, emanating from the triple point, to separate the subsonic flow from the supersonic flow. In the region between the slip line and the nozzle axis where the flow is at first subsonic, conventional one-dimensional flow equations can be used in an iteration fashion to locate the position of the slip line by equating the pressure on either side.

One of the classical reports applying the method of characteristics technique specifically to the calculation of rocket exhaust plumes is by Love et al. (12). However, in this report the so-called "foldback" method is employed in which the characteristic net, which folds back upon itself downstream of a shock point, is deleted leaving only one network. In addition, no shock relations are employed so that the resulting calculations are valid only in a region near the nozzle exit. These simplifying techniques have been eliminated by Vick et al. (13) and the resulting plume flowfields generated by their improved calculations have been experimentally verified (14).

Others using the method of characteristics are Moe and Troesch (15) and Eastman and Radtke (16). In addition to calculating the flow between the nozzle axis and the dividing streamline Moe and Troesch also calculate the flowfield between the air shock and the dividing streamline. On the other hand, Eastman and Radtke coalesce the air shock and the dividing streamline by assuming a Newtonian shock layer.

More recently, Prozan (17) has developed a method of characteristics computer program that calculates supersonic, inviscid flowfields assuming ideal, frozen, or equilibrium reacting gas mixtures. This program has been developed to a high degree of sophistication and is directly applicable to nozzle flows and plume flows. The user's manual for this program is by Butler (18) and a comparison between experimental data and results from this program has been made by Ratliff (19).

The method of characteristics has been applied to the flowfield created by two nozzles by several investigators including Whitehurst and Mourer (20).

In addition to the method of characteristics technique for computing inviscid plume flowfields, Boynton and Thomson (21) have recently devised a Lagrangian finite-difference technique which can be applied to the solution of inviscid as well as viscous plume flowfields. This method will be discussed in detail later.

Both the method of characteristics and the Lagrangian finite-difference technique require a considerable amount of effort and computer time. This has prompted several investigators to develop approximate methods which can be quickly applied to the problem of determining the gross plume structure of underexpanded exhaust flows. In particular, much of this effort has been devoted to determining the geometry of the dividing streamline between the air and exhaust flow, thus giving an approximation of the plume size. Some of the approximate methods are by Albini (22), Hubbard (23), and Luce and Jarvinen (24). Boynton has compared the approximate methods of Albini and Hubbard with the exact Lagrangian finite-difference technique in Reference (25).

The various inviscid methods just described are excellent for predicting the plume shape, jet shock location, and internal pressure distribution. However, since the air is not allowed to mix with the exhaust flow, the afterburning effects are totally ignored. This has led to several investigations which have used standard turbulent, reacting boundary layer equations to represent the turbulent mixing layer in the plume. In these equations the laminar transport coefficients are neglected in comparison with the turbulent transport coefficients. For edge conditions along the mixing layer boundary, most investigators have used information from the method of characteristics technique previously described, usually ignoring all shocks. However, when using the boundary layer equations to represent the turbulent mixing layer several problems immediately become apparent. First of all, this representation is valid only in the region near the nozzle exit, where the mixing layer is very thin, or far from the nozzle exit where the mixing layer is fully developed. In the intermediate region, where the mixing layer thickness is of the same order of magnitude as the internal inviscid flow thickness, the boundary layer equations are not valid. The second drawback in using the boundary layer equations is that a zero radial pressure gradient is assumed, and this turns out to be a poor approximation in the turbulent mixing layer, except near the nozzle exit. Since the diffusion process is dependent to some extent on the radial pressure gradient this creates further difficulties. And finally, and probably most important, it should be noted that when using the boundary layer equations to represent the turbulent mixing layer, a very difficult iteration is required when simultaneously solving the complete disturbed flowfield.

Nevertheless, several investigators have applied the theory developed by Libby (26) for the turbulent mixing of reactive gases between two coaxial, compressible streams to the turbulent mixing layer in a rocket exhaust plume. Libby assumed that both the turbulent Prandtl and Lewis numbers were equal to one and set the axial pressure gradient equal to zero. With these assumptions, Libby was able to find a closed form solution to the turbulent boundary layer equations with the aid of the von Mises transformation and an assumed eddy viscosity model. If this theory is applied to the problem of solving the turbulent mixing layer in an exhaust plume the applicability is very limited since a zero axial pressure gradient implies that the pressure at the exit of the nozzle is equal to the freestream pressure resulting in a balanced jet condition.

In any case, this theory has been applied by Feigenbutz (27) to the turbulent mixing layer in the exhaust plume of a single-engined missile using a flame-front chemical analysis. Later, Rozsa (28), (29), and (30) also made a complete investigation using Libby's theory. Both of the above authors stated that later appearing reports of their work would relax the assumption of a zero axial pressure gradient, but since no such reports have appeared it seems that, as yet, they have been unsuccessful in their attempts. Libby's theory has been extended by Audeh (31) and (32) to the turbulent mixing of three concentric reacting jets to account for the turbine exhaust injection along the nozzle wall.

In order to account for an axial pressure gradient using the turbulent boundary layer equations, it is apparent that other numerical techniques must be employed. Methods for attempting to do this range from conventional integral methods (33) to elaborate finite-difference solutions of the full turbulent boundary layer equations. Vasiliu (34) was the first to apply the latter method to the turbulent mixing layer in a rocket exhaust plume. In his solution, non-unity turbulent Lewis and Prandtl numbers are used and chemical nonequilibrium is assumed. Unfortunately, Vasiliu had numerical difficulties because he was limited to 18 internal grid points by the storage capacity of his computer. Since then, Edelman and Fortune (35) and (36) with a newer computer and an improved chemical nonequilibrium analysis were able to circumvent the problems which Vasiliu encountered.

Still, the major drawback in using conventional turbulent boundary layer equations to represent the turbulent mixing layer in the plume lies in the problem of determining the coupling effects between the two surrounding inviscid flows and the mixing layer. Iteration techniques must be employed which are based on the boundary layer displacement thickness. This makes the computation very difficult.

A way around this dilemma comes from the work done by Thomson (3) and Boynton (37), solving the problem of a high altitude rocket exhaust plume, and by Moretti et al. (38) solving the somewhat related problem of the reacting flowfield in a supersonic combustion ramjet (SCRAMJET). In these reports, new equations are used which were derived from the Navier-Stokes equations using an order of magnitude analysis in which the gradients of pressure, temperature, velocity, and concentrations are assumed much smaller in the direction of flow than in a direction normal to the flow. The resulting equations are somewhat similar to the boundary layer equations but include a normal momentum equation, thus allowing for a radial pressure gradient. In addition, when the transport terms are dropped from these equations, the usual inviscid flow equations are obtained.

Two techniques have been proposed to solve this set of mixed hyperbolic-parabolic equations. The first technique, (39) and (40), uses a modified method of characteristics procedure in which viscous forcing terms are used. This technique has been applied to supersonic combustion ramjet flowfields and to plume flowfields with no internal shock structure. The second technique is a Lagrangian finite-difference procedure which has been applied by Thomson and Boynton (21) principally to the solution of high altitude rocket exhaust plumes. This technique divides the flowfield into a grid consisting of streamtubes and the surfaces orthogonal to them. For high Mach number flows, such as in highly underexpanded plume flows, this procedure is more efficient than the previously mentioned modified method of characteristic procedure because in it the characteristic mesh becomes very compressed at high Mach numbers.

Boynton (37) assumes frozen flow and laminar transport in the viscous mixing layer which he extends from the internal plume shock to the external plume shock. In order to calculate a high altitude plume flowfield using his computer program one must first determine the undisturbed, internal inviscid flow by expanding the exhaust gases to a vacuum. Next, the initial location of the jet and air shocks must be determined. In attempting this, difficulties arise if the Lagrangian finite-difference technique is used. This is in contrast to the method of characteristics where shocks are relatively easy to locate. In any case, with the initial shock locations determined, the viscous shock layer with multicomponent diffusion in the region between the air and jet shocks may be solved by letting the two shocks propagate into the previously calculated undisturbed flows. Streamtubes are added to account for the mass flow that is added as the shocks propagate into the undisturbed flows. Thus, the entire disturbed region can be solved without iteration using this method.

Purpose of This Study

In the present study, the analysis made by Boynton (37) for high altitude rocket exhaust plumes is extended to intermediate altitude plumes by incorporating chemical nonequilibrium and turbulent transport capabilities into the computer program.

Chemical nonequilibrium is handled using either an explicit or an implicit technique. The implicit technique utilizes Moretti's linearization procedure (41) to reduce the species continuity equations to a set of algebraic equations in terms of the unknown molar concentrations at each new surface. The turbulent diffusion is described in terms of a single binary diffusion coefficient by applying Fick's law, thereby eliminating diffusion due to pressure or temperature gradients. The eddy viscosity model chosen is a Prandtl-like model which is constant across each orthogonal surface. In addition, the turbulent Prandtl and Lewis numbers are assumed constant throughout the mixing layer.

ANALYTICAL DEVELOPMENT

Derivation of Governing Equations

In this section the equations necessary for calculating the supersonic flowfield in an intermediate altitude rocket exhaust plume will be derived. These equations are applicable to both viscous and nonviscous regions if the proper terms are retained. In the viscous region between the air shock and jet shock all three transport processes occur resulting in viscosity, heat transfer, and diffusion. These transport processes are generally turbulent in the intermediate altitude region. Also, as the turbulent mixing layer increases in size the lateral pressure gradient can no longer be ignored. And finally, chemical nonequilibrium must be considered in this altitude region but translational, rotational, vibrational, and ionizational nonequilibrium are as usual considered negligible when calculating macroscopic flow properties in reacting plume flows.

In the following analysis, laminar transport is assumed initially in order to simplify the derivation. After the final forms of the laminar equations are obtained, they are transformed to their turbulent counterparts by an intuitive argument which uses information gained from conventional turbulent boundary layer theory. It is assumed further that the mean flow is in a steady state and is either axisymmetric or two-dimensional.

In accordance with the derivation of Edelman and Weilerstein (39), the general equations for the steady flow of a reacting mixture of perfect gases are:

Global continuity:

$$\nabla \cdot \rho\overline{V} = 0 \tag{1}$$

Species continuity:

$$\nabla \cdot \rho c_i\overline{V} = \rho\dot{W}_i - \nabla \cdot \overline{J}_i \tag{2}$$

Momentum:

$$\nabla \cdot [(\rho\overline{V})\overline{V}] = -\nabla p + \nabla \cdot \overline{\overline{\tau}} \tag{3}$$

Energy:

$$\nabla \cdot \rho\overline{V}H = \nabla \cdot (\overline{\overline{\tau}} \cdot \overline{V}) - \nabla \cdot \overline{Q} - \nabla \cdot \sum_{i=1}^{N} h_i\overline{J}_i \tag{4}$$

State:

$$p = \rho\overline{R}T \sum_{i=1}^{N} \frac{c_i}{M_i} \tag{5}$$

where the total enthalpy, H is defined by

$$H = \sum_{i=1}^{N} c_i h_i + \frac{\overline{V} \cdot \overline{V}}{2} \tag{6}$$

and where the subscript i represents the ith species from a total of N.

For a Newtonian fluid the viscous shearing stress is given by

$$\overline{\overline{\tau}} = \mu\overline{\overline{\varepsilon}} - \frac{2}{3}(\nabla \cdot \overline{V})\overline{\overline{\delta}}\mu \tag{7}$$

where

$$\bar{\bar{\varepsilon}} = [\nabla\bar{V} + (\nabla\bar{V})^T] \tag{8}$$

and $\bar{\bar{\delta}}$ is the unit tensor.

The heat transfer term can be evaluated using the Fourier law given by

$$\bar{Q} = -k\nabla T = -\frac{c_{p_f}\mu\nabla T}{Pr} \tag{9}$$

where the frozen specific heat, c_{p_f}, is defined as

$$c_{p_f} = \sum_{i=1}^{N} c_i c_{p_i} \tag{10}$$

The diffusional mass flux term can be simplified by assuming that the gas mixture is composed of light and heavy particles so that one binary diffusion coefficient applies for all species. In addition, if the pressure and thermal diffusion are neglected in comparison with the concentration diffusion (ordinary diffusion), then Fick's law (42) applies and the diffusional mass flux becomes

$$\bar{J}_i = -\rho D_{ij}\nabla c_i = -\mu\frac{Le}{Pr}\nabla c_i \tag{11}$$

The species production term, $\dot{W}_i$, represents the production rate of the ith species from the system of chemical reactions given by

$$\sum_{j=1}^{N'} \nu'_{j,m} A_j \underset{k_{b_m}}{\overset{k_{f_m}}{\rightleftharpoons}} \sum_{j=1}^{N'} \nu''_{j,m} A_j \quad ; \quad m = 1, M' \tag{12}$$

where N'species are involved in M' chemical reactions and the ν_j's are the stoichiometric coefficients associated with the A_j species. If the classical Arrhenius law (43) is assumed then the species production term is given by

$$\dot{W}_i = \frac{M_i}{\rho} \sum_{m=1}^{M'} (\nu''_{i,m} - \nu'_{i,m}) \left[k_{f_m} \prod_{j=1}^{N'} Y_j^{\nu'_{j,m}} - k_{b_m} \prod_{j=1}^{N'} Y_j^{\nu''_{j,m}} \right] \tag{13}$$

where the reaction rates are given by

$$k_{f_m} \text{ or } k_{b_m} = c_1 T^{c_2} e^{(-c_3/T)} \tag{14}$$

and where, Y_j, the molar concentration, is defined as

$$Y_j = \frac{\rho c_j}{M_j} \tag{15}$$

The above conservation equations are now written in the natural coordinate system, where s and n are the coordinates along and normal to a streamline.

Global continuity:

$$\frac{\partial \rho u}{\partial s} + \frac{j \rho u}{r} \sin \theta + \rho u \theta_n = 0 \tag{16}$$

or

$$\frac{\partial}{\partial s} (\rho A u) = 0 \tag{17}$$

Species continuity:

$$\rho u \frac{\partial c_i}{\partial s} = \rho \dot{W}_i + \mu \frac{Le}{Pr} \theta_n \frac{\partial c_i}{\partial s} - \mu \frac{Le}{Pr} \theta_s \frac{\partial c_i}{\partial n} + \frac{1}{r^j} \frac{\partial}{\partial s} (r^j \mu \frac{Le}{Pr} \frac{\partial c_i}{\partial s})$$
$$+ \frac{1}{r^j} \frac{\partial}{\partial n} (r^j \mu \frac{Le}{Pr} \frac{\partial c_i}{\partial n}) \tag{18}$$

Streamwise momentum:

$$\rho u \frac{\partial u}{\partial s} = - \frac{\partial p}{\partial s} + \frac{1}{r^j} \frac{\partial}{\partial n} (r^j \tau_{sn}) + \frac{1}{r^j} \frac{\partial}{\partial s} (r^j \tau_{ss}) + (\tau_{ss} - \tau_{nn}) \theta_n$$
$$- 2\tau_{sn}\theta_s - \tau_{\omega\omega} \frac{j}{r} \sin \theta \tag{19}$$

Normal momentum:

$$\rho u^2 \theta_s = - \frac{\partial p}{\partial n} + \frac{1}{r^j} \frac{\partial}{\partial n} (r^j \tau_{nn}) + \frac{1}{r^j} \frac{\partial}{\partial s} (r^j \tau_{sn}) + 2\theta_n \tau_{sn}$$
$$- \tau_{\omega\omega} \frac{j}{r} \cos \theta \tag{20}$$

Energy:

$$\rho u \frac{\partial H}{\partial s} = \tau_{ss} u \partial_n + \frac{1}{r^j} \frac{\partial}{\partial s} (r^j \tau_{ss} u) - \tau_{sn} u \theta_s + \frac{1}{r^j} \frac{\partial}{\partial n} (r^j \tau_{ns} u)$$
$$+ \frac{\mu}{Pr} (\theta_n \frac{\partial H}{\partial s} - \theta_s \frac{\partial H}{\partial n}) + \frac{1}{r^j} \frac{\partial}{\partial s} (r^j \frac{\mu}{Pr} \frac{\partial H}{\partial s}) + \frac{1}{r^j} \frac{\partial}{\partial n} (r^j \frac{\mu}{Pr} \frac{\partial H}{\partial n})$$
$$+ \frac{\mu}{Pr} \theta_n (Le-1) \sum_{i=1}^{N} h_i \frac{\partial c_i}{\partial s} - \frac{\mu}{Pr} \theta_s (Le-1) \sum_{i=1}^{N} h_i \frac{\partial c_i}{\partial n}$$
$$+ \frac{1}{r^j} \sum_{i=1}^{N} \frac{\partial}{\partial s} [r^j \frac{\mu}{Pr} (Le-1) h_i \frac{\partial c_i}{\partial s}] + \frac{1}{r^j} \sum_{i=1}^{N} \frac{\partial}{\partial n} [r^j \frac{\mu}{Pr} (Le-1) h_i \frac{\partial c_i}{\partial n}]$$
$$+ \frac{\mu}{Pr} [\theta_s \frac{\partial}{\partial s} (u^2/2) - \theta_n \frac{\partial}{\partial n} (u^2/2)] - \frac{1}{r^j} \frac{\partial}{\partial s} [r^j \frac{\mu}{Pr} \frac{\partial}{\partial s} (u^2/2)]$$
$$- \frac{1}{r^j} \frac{\partial}{\partial n} [r^j \frac{\mu}{Pr} \frac{\partial}{\partial n} (u^2/2)] \tag{21}$$

where the shearing stress components are given by

$$\tau_{ss} = 2\mu \frac{\partial u}{\partial s} - \frac{2}{3} \mu \left[\frac{\partial u}{\partial s} + \frac{ju}{r} \sin\theta + u\theta_n \right]$$

$$\tau_{nn} = 2\mu u\theta_n - \frac{2}{3} \mu \left[\frac{\partial u}{\partial s} + \frac{ju}{r} \sin\theta + u\theta_n \right] \tag{22}$$

$$\tau_{\omega\omega} = 2\mu \frac{j}{r} u \sin\theta - \frac{2}{3} \mu \left[\frac{\partial u}{\partial s} + \frac{ju}{r} \sin\theta + u\theta_n \right]$$

$$\tau_{ns} = \tau_{sn} = \mu \left[\frac{\partial u}{\partial n} + u\theta_s \right]$$

and where j, the metric coefficient, is equal to 0 for two-dimensional flow and is equal to 1 for axisymmetric flow.

In order to simplify the above equations, an order of magnitude analysis is usually employed. In the classical boundary layer approach, where the boundary layer thickness, δ_b, is assumed much smaller than a characteristic length, L, and the Reynolds number is very large, the following relationship holds:

$$\left(\frac{L}{\delta_b}\right)^2 \frac{\mu_o}{\rho_o u_o L} = O(1) \tag{23}$$

When this order of magnitude analysis is applied to the normal momentum equation it reduces to

$$\frac{\partial p}{\partial n} = 0 \tag{24}$$

If the lateral pressure gradient is to be retained, then Equation 23 must be replaced by a more general expression.

Next, consider a viscous flowfield whose region of influence is characterized by a lateral dimension δ_v which can range in size from the boundary layer thickness δ_b to the characteristic length L. For this situation,

$$\left(\frac{L}{\delta_v}\right)^2 \frac{\mu_o}{\rho_o u_o L} = O(\beta') \tag{25}$$

where β' ranges from 0 for inviscid flow to 1 for boundary layer flow. An example of this type of flowfield is the viscous mixing layer in a rocket exhaust plume where the initial thickness is very small at the nozzle exit and increases in size until it completely dominates the entire field. In this viscous mixing layer it can be assumed that gradients of pressure, temperature, velocity, and species concentrations are substantially smaller in the direction of flow than in a direction normal to the flow so that

$$\frac{\partial}{\partial s} = O\left(\frac{1}{L}\right) \tag{26}$$

and

$$\frac{\partial}{\partial n} = O\left(\frac{1}{\delta}\right) \tag{27}$$

In addition, since the present flow is assumed supersonic, it will disturb the entire field only over a lateral extent which is assumed to be of the order δ while the viscous effects are limited to a lateral extent δ_v. Let

$$\beta = \left(\frac{\delta_v}{\delta}\right)^2 < 1 \tag{28}$$

and consider the condition

$$\left(\frac{L}{\delta}\right)^2 \left(\frac{\mu_o}{\rho_o u_o L}\right) = O(\beta) \tag{29}$$

In general, β and $\frac{\delta}{L}$ are closer in magnitude to zero than one but cannot be neglected when compared to terms of the order 1. However, terms of the order $(\frac{\delta}{L})\beta$ or higher are assumed negligible. The present order of magnitude analysis is slightly different than the one given by Edelman and Weilerstein (39) since the flow angle θ is not always assumed small and of the order δ. In a rocket exhaust plume, the flow angle can be quite large near the nozzle exit, but decreases to zero in the far field where the mixing layer is fully developed. In the present analysis, θ_n and θ_s are assumed of the order 1/L.

Upon nondimensionalizing the variables and using Equation 29 , the previous set of equations become:

Global continuity:

$$\frac{\partial}{\partial \bar{s}} (\overline{\rho u}) + \frac{j}{\bar{r}} \bar{\rho}\, \bar{u} \sin\theta + \bar{\rho}\, \bar{u}\, \bar{\theta}_n = 0 \tag{30}$$

Species continuity:

$$\overline{\rho u} \frac{\partial c_i}{\partial \bar{s}} = \frac{L}{u_o} (\bar{\rho}\, \dot{W}_i) + (\tfrac{\delta}{L})^2 \beta \bar{\mu} \frac{Le}{Pr} \bar{\theta}_n \frac{\partial c_i}{\partial \bar{s}} - (\tfrac{\delta}{L}) \beta \bar{\mu} \frac{Le}{Pr} \bar{\theta}_s \frac{\partial c_i}{\partial \bar{n}}$$

$$+ (\tfrac{\delta}{L})^2 \beta \frac{1}{\bar{r}^j} \frac{\partial}{\partial \bar{s}} \left(\bar{r}^j\, \bar{\mu} \frac{Le}{Pr} \frac{\partial c_i}{\partial \bar{s}} \right) + \beta \frac{1}{\bar{r}^j} \frac{\partial}{\partial \bar{n}} \left(\bar{r}^j\, \bar{\mu} \frac{Le}{Pr} \frac{\partial c_i}{\partial \bar{n}} \right) \tag{31}$$

Streamwise momentum:

$$\bar{\rho}\,\bar{u}\,\frac{\partial \bar{u}}{\partial \bar{s}} = -\frac{p_o}{\rho_o u_o^2}\frac{\partial \bar{p}}{\partial \bar{s}} + (\beta)\frac{1}{\bar{r}^j}\frac{\partial}{\partial \bar{n}}(\bar{r}^j\,\bar{\tau}_{sn}) + \left(\frac{\delta}{L}\right)^2\beta\,\frac{1}{\bar{r}^j}\frac{\partial}{\partial \bar{s}}(\bar{r}^j\,\bar{\tau}_{ss})$$

$$+ \left(\frac{\delta}{L}\right)^2\beta(\bar{\tau}_{ss} - \bar{\tau}_{nn})\bar{\theta}_n - \left(\frac{\delta}{L}\right)\beta 2\,\bar{\tau}_{sn}\bar{\theta}_s - \left(\frac{\delta}{L}\right)^2\beta\,\bar{\tau}_{\omega\omega}\frac{j\sin\theta}{\bar{r}} \qquad (32)$$

Normal momentum:

$$\left(\frac{\delta}{L}\right)\bar{\rho}\,\bar{u}^2\bar{\theta}_s = -\frac{p_o}{\rho_o u_o^2}\frac{\partial \bar{p}}{\partial \bar{n}} + \left(\frac{\delta}{L}\right)^2\beta\left[\frac{1}{\bar{r}^j}\frac{\partial}{\partial \bar{n}}(\bar{r}^j\bar{\tau}_{nn}) + \frac{1}{\bar{r}^j}\frac{\partial}{\partial \bar{s}}(\bar{r}^j\bar{\tau}_{sn})\right]$$

$$+ \left(\frac{\delta}{L}\right)^2\beta 2\bar{\theta}_n\bar{\tau}_{sn} - \left(\frac{\delta}{L}\right)^3\beta\bar{\tau}_{\omega\omega}\,\frac{j}{\bar{r}}\cos\theta \qquad (33)$$

Energy:

$$\bar{\rho}\,\bar{u}\,\frac{\partial \bar{H}}{\partial \bar{s}} = \left(\frac{\delta}{L}\right)^2\beta\bar{\tau}_{ss}\bar{u}\,\bar{\theta}_n + \left(\frac{\delta}{L}\right)^2\beta\,\frac{1}{\bar{r}^j}\frac{\partial}{\partial \bar{s}}(\bar{r}^j\bar{\tau}_{ss}\bar{u}) - \left(\frac{\delta}{L}\right)\beta\,\bar{\tau}_{sn}\bar{u}\,\bar{\theta}_s$$

$$+ \beta\frac{1}{\bar{r}^j}\frac{\partial}{\partial \bar{n}}(\bar{r}^j\bar{\tau}_{ns}\bar{u}) + \left(\frac{\delta}{L}\right)^2\beta\frac{\bar{\mu}}{Pr}\,\bar{\theta}_n\,\frac{\partial \bar{H}}{\partial \bar{s}} - \left(\frac{\delta}{L}\right)\beta\frac{\bar{\mu}}{Pr}\,\bar{\theta}_s\,\frac{\partial \bar{H}}{\partial \bar{n}}$$

$$+ \left(\frac{\delta}{L}\right)^2\beta\frac{1}{\bar{r}^j}\frac{\partial}{\partial \bar{s}}\left(\bar{r}^j\,\frac{\bar{\mu}}{Pr}\,\frac{\partial \bar{H}}{\partial \bar{s}}\right) + \beta\frac{1}{\bar{r}^j}\frac{\partial}{\partial \bar{n}}\left(\bar{r}^j\,\frac{\bar{\mu}}{Pr}\,\frac{\partial \bar{H}}{\partial \bar{n}}\right) \qquad (34)$$

$$+ \left(\frac{\delta}{L}\right)^2\beta\frac{\bar{\mu}}{Pr}\,\bar{\theta}_n(Le-1)\sum_{i=1}^{N}\bar{h}_i\,\frac{\partial c_i}{\partial \bar{s}} - \left(\frac{\delta}{L}\right)\beta\,\frac{\bar{\mu}}{Pr}\,\bar{\theta}_s(Le-1)\sum_{i=1}^{N}\bar{h}_i\,\frac{\partial c_i}{\partial \bar{n}}$$

$$+ \left(\frac{\delta}{L}\right)^2\beta\frac{1}{\bar{r}^j}\sum_{i=1}^{N}\frac{\partial}{\partial \bar{s}}\left[\bar{r}^j\,\frac{\bar{\mu}}{Pr}(Le-1)\bar{h}_i\,\frac{\partial c_i}{\partial \bar{s}}\right]$$

$$+ \beta\frac{1}{\bar{r}^j}\sum_{i=1}^{N}\frac{\partial}{\partial \bar{n}}\left[\bar{r}^j\,\frac{\bar{\mu}}{Pr}(Le-1)\bar{h}_i\,\frac{\partial c_i}{\partial \bar{n}}\right] + \left(\frac{\delta}{L}\right)^2\beta\frac{\bar{\mu}}{Pr}\,\bar{\theta}_s\,\frac{\partial \bar{u}^2/2}{\partial \bar{s}}$$

$$- \left(\frac{\delta}{L}\right)\beta\,\bar{\theta}_n\frac{\bar{\mu}}{Pr}\,\frac{\partial \bar{u}^2/2}{\partial \bar{n}} - \left(\frac{\delta}{L}\right)^2\beta\frac{1}{\bar{r}^j}\,\frac{\partial}{\partial \bar{s}}\left(\bar{r}^j\frac{\bar{\mu}}{Pr}\,\frac{\partial \bar{u}^2/2}{\partial \bar{s}}\right)$$

$$- \beta\,\frac{1}{\bar{r}^j}\,\frac{\partial}{\partial \bar{n}}\left(\bar{r}^j\frac{\bar{\mu}}{Pr}\,\frac{\partial \bar{u}^2/2}{\partial \bar{n}}\right)$$

In these equations a bar over a variable indicates that it has been nondimensionalized.

After neglecting higher order terms and redimensionalizing the remaining variables, the final form of the laminar equations become:

Global continuity:

$$\frac{\partial}{\partial s}(\rho A u) = 0 \tag{35}$$

Species continuity:

$$\rho u \frac{\partial c_i}{\partial s} = \rho \dot{W}_i + \frac{1}{r^j}\frac{\partial}{\partial n}\left\{r^j \mu \frac{Le}{Pr}\frac{\partial c_i}{\partial n}\right\} \tag{36}$$

Streamwise momentum:

$$\rho u \frac{\partial u}{\partial s} + \frac{\partial p}{\partial s} = \frac{1}{r^j}\frac{\partial}{\partial n}\left\{r^j \mu \frac{\partial u}{\partial n}\right\} \tag{37}$$

Normal momentum:

$$\rho u^2 \theta_s + \frac{\partial p}{\partial n} = 0 \tag{38}$$

Energy:

$$\rho u \frac{\partial H}{\partial s} = \frac{1}{r^j}\frac{\partial}{\partial n}\left\{r^j \frac{\mu}{Pr}\frac{\partial H}{\partial n}\right\} + \frac{1}{r^j}\frac{\partial}{\partial n}\left[r^j\left(1 - \frac{1}{Pr}\right)\mu u\frac{\partial u}{\partial n}\right]$$

$$+ \frac{1}{r^j}\sum_{i=1}^{N}\frac{\partial}{\partial n}\left[r^j (Le-1)\frac{\mu}{Pr} h_i \frac{\partial c_i}{\partial n}\right] \tag{39}$$

State:

$$p = \rho \bar{R} T \sum_{i=1}^{N}\frac{c_i}{M_i} \tag{40}$$

If the transport terms are set equal to zero in the above equations, the usual inviscid equations are obtained; whereas if the normal pressure gradient term is set equal to zero, the boundary layer equations are obtained. This has led Edelman and Weilerstein (40) to appropriately name these equations the inviscid-viscid equations.

These equations are identical to similar equations derived by Boynton (37) if the species production term $\rho\dot{W}$ is set equal to zero and the energy equation is rearranged.

The above equations were derived for laminar transport processes. At this point it is assumed that the equations for turbulent transport are identical in form if the laminar transport coefficients are replaced by their turbulent counterparts. The appropriateness of this assumption stems from previous work in the area of turbulent mixing layers. Various authors (26), (44), and (45) have shown that the turbulent boundary layer equations can be made identical in form to the laminar boundary layer equations if applied to a turbulent mixing layer. In order to accomplish this, the expressions for the turbulent shear stress, turbulent heat flux, and turbulent diffusion are replaced by expressions suggested by T. V. Boussinesq (46). Then the dynamic viscosity, the coefficient of thermal conductivity, and the binary diffusion coefficient are neglected in comparison with their turbulent counterparts the eddy viscosity, the eddy thermal conductivity, and the eddy diffusivity. The resulting turbulent boundary layer equations, in terms of time-averaged flow variables, are identical in form to the previous laminar boundary layer equations.

The present equations (35, 36, 37, 38, 39, and 40) are the usual laminar boundary layer equations written in natural coordinates except for Equation 38, the normal momentum equation. However, this equation will not change in form if the flow is turbulent. Consequently, the final form of the turbulent equations, using the previous arguments, will become after some rearrangement;

Global continuity:

$$\frac{\partial}{\partial s}(\rho u A) = 0 \tag{41}$$

Species continuity:

$$\rho u \frac{\partial c_i}{\partial s} = \rho \dot{W} + \frac{1}{r^j} \frac{\partial}{\partial n} \left(r^j \varepsilon_v \frac{Le_t}{Pr_t} \frac{\partial c_i}{\partial n} \right) \tag{42}$$

Streamwise momentum:

$$\rho u \frac{\partial u}{\partial s} + \frac{\partial p}{\partial s} = \frac{1}{r^j} \frac{\partial}{\partial n} \left(r^j \varepsilon_v \frac{\partial u}{\partial n} \right) \tag{43}$$

Normal momentum:

$$\rho u^2 \frac{\partial \theta}{\partial s} + \frac{\partial p}{\partial n} = 0 \tag{44}$$

Energy:

$$\rho u \frac{\partial}{\partial s}(h + \tfrac{1}{2}u^2) = \frac{1}{r^j} \frac{\partial}{\partial n} \left[r^j \left(\frac{c_{p_f}}{Pr_t} \varepsilon_v \frac{\partial T}{\partial n} + \varepsilon_v u \frac{\partial u}{\partial n} + \sum_{i=1}^{N} \varepsilon_v \frac{Le_t}{Pr_t} h_i \frac{\partial c_i}{\partial n} \right) \right] \tag{45}$$

State:

$$p = \rho \bar{R} T \sum_{i=1}^{N} \frac{c_i}{M_i} \tag{46}$$

where ε_v, Pr_t, and Le_t are the eddy viscosity, turbulent Prandtl number, and the turbulent Lewis number, respectively. In addition, note that all variables are now time-averaged quantities.

Numerical Solution of Equations

The equations just derived contain both hyperbolic and parabolic types if the flow is everywhere supersonic. Two methods have been used to solve sets of equations of this type. Originally, Moretti et al. (38) proposed a modified method of characteristics technique which has been further developed by Edelman and Weilerstein (39, 40). In this technique the viscous contributions are treated as forcing terms in the compatability equations along the inviscid characteristics. The compatability equations can then be solved for the flow direction and pressure at the new grid points. With the pressure gradient thus determined, the usual boundary layer equations are solved at the new grid points for their viscous terms. Since these viscous terms appear in the compatability equations, a mechanism for an iterative solution is established. Thus, the average values of the viscous terms over the characteristic step are substituted into the compatability equations and the new flow angle and pressure are determined. The above cycle of calculations is repeated until the pressure change between successive iterations is less than some predetermined value.

The second method is the Lagrangian finite-difference technique which has been applied by Boynton and Thomson (21) to the nonreacting, laminar equations which were derived earlier. This method differs from Lagrangian finite-difference schemes previously reported in the literature (47) in that fluid elements are followed through space rather than time. The flowfield is divided into a grid which is composed of streamlines and the surfaces orthogonal to them. See Figure 2. In the convention adopted for this study, the kth streamtube is bounded on the left by the kth streamline for an observer facing downstream, and the orthogonal surfaces are designated by the index ℓ. In each streamtube the properties are constant in the orthogonal direction but vary between orthogonal surfaces.

To start the calculation, data along an initial orthogonal surface are required. These data include the flow properties in the streamtubes and the positions and flow angles of the dividing streamlines. In addition, the boundary conditions on either side of the flow must be specified. The curvature of the kth streamline at the initial ℓth surface is evaluated using the normal momentum equation. This curvature is used to extend the kth streamline to the orthogonal surface $\ell + 1$. This procedure is repeated for each streamline so that all streamtube areas at the new surface can be calculated. If the flow is

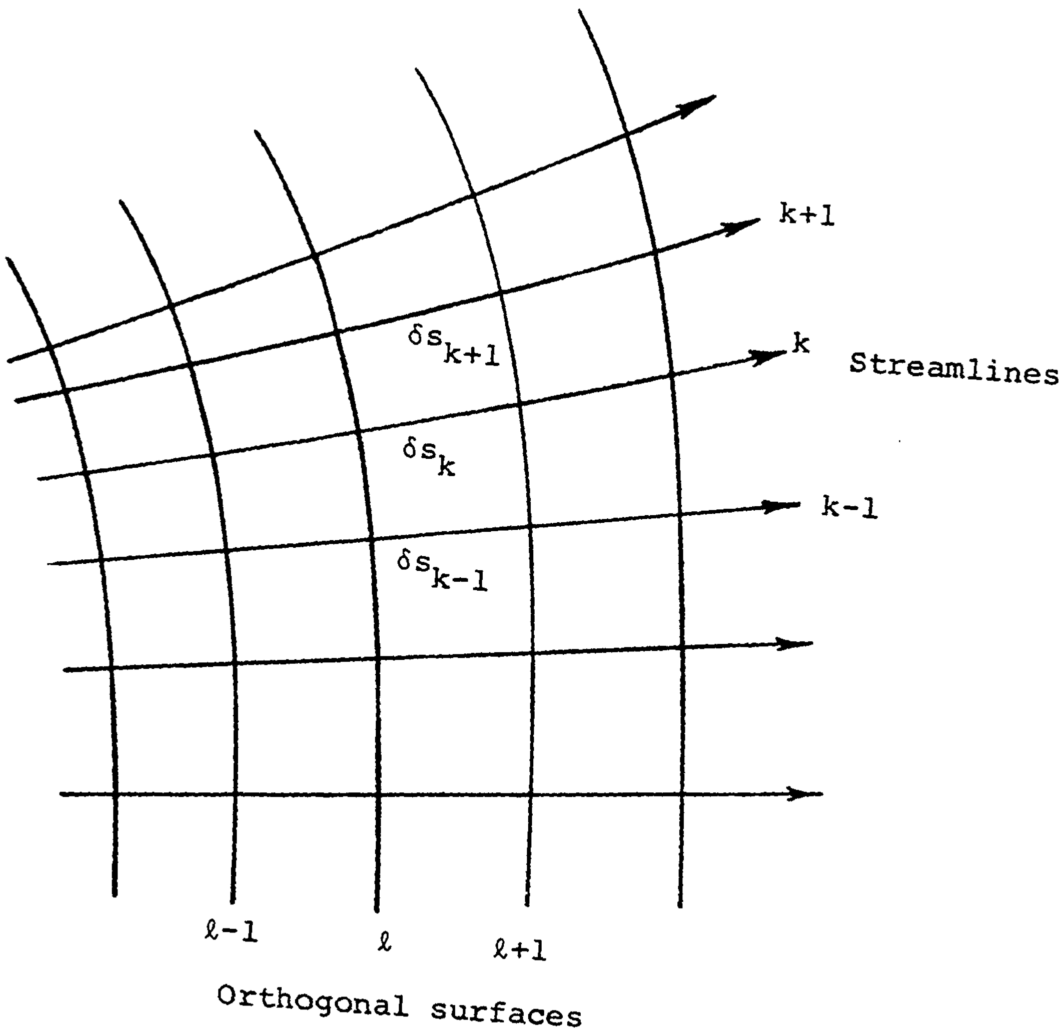

Figure 2. Lagrangian finite-difference mesh

inviscid, the remaining conservation equations can then be integrated directly along the streamtubes in the usual manner. However, if the flow is viscous, the net transfer of mass, momentum, and energy between streamtubes during the step downstream must be calculated using the flux terms in the conservation equations. Consequently, these conservation equations cannot be integrated directly but must be solved in finite-difference form. The calculation is arranged so that all mass, momentum, and energy lost from one streamtube reappears in an adjacent streamtube. Using the above technique, the flow properties at the $\ell + 1$ surface can be calculated. Each new surface is calculated twice. For the second pass, average values between the old and new surfaces are used. The computation marches downstream in this manner until the calculation is terminated.

This scheme is explicit and is therefore subject to instabilities unless the step size is limited. In addition, the streamline curvature must be computed using a weighted mean of the curvatures at the old and new surfaces to insure stability. A discussion of the stability of this technique appears in reference (21) where the stable step size is given as

$$\delta s \leq \frac{1}{2}\,\delta n \left[\frac{1}{Re} + (M^2 - 1)^{-1/2}\right]^{-1} \tag{47}$$

In the present study, the Lagrangian finite-difference technique of Boynton and Thomson is applied to the final set of equations which were derived. These equations can be applied to chemically reacting flowfields with turbulent transport. In finite-difference form, these equations become:

Global continuity:

$$\dot{m}_k = \rho_{k,\ell} u_{k,\ell} A_{k,\ell} = \rho_{k,\ell+1} u_{k,\ell+1} A_{k,\ell+1} \tag{48}$$

Species continuity:

$$\dot{m}_k (c_{i,k,\ell+1} - c_{i,k,\ell}) = \Delta_k \left[\overline{r_k^j \, J_{i,k} \, \delta s_k} \right] (2\pi)^j + \frac{\overline{\dot{W}_i} \, \overline{\delta s_k} \, \dot{m}_k}{\bar{u}_k} \tag{49}$$

Streamwise momentum:

$$\dot{m}_k (u_{k,\ell+1} - u_{k,\ell}) + \bar{A}_k (p_{k,\ell+1} - p_{k,\ell}) = \Delta_k \left[\overline{r_k^j \, \tau_k \, \delta s_k} \right] (2\pi)^j \tag{50}$$

Normal momentum:

$$-\left(\frac{\partial \theta}{\partial s}\right)_{k,\ell} = \left[\frac{4(2\pi r_{k,\ell})^j}{u_{k,\ell} + u_{k+1,\ell}} \right] \left[\frac{p_{k+1,\ell} - p_{k,\ell}}{\dot{m}_k + \dot{m}_{k+1}} \right] \tag{51}$$

Energy:

$$\dot{m}_k (h_{k,\ell+1} - h_{k,\ell} + \tfrac{1}{2} u_{k,\ell+1}^2 - \tfrac{1}{2} u_{k,\ell}^2) = \Delta_k \left[\overline{r_k^j (Q_k + [\tfrac{1}{2}(u_k^2 + u_{k+1}^2)]^{1/2} \tau_k + \sum_{i=1}^{N} h_{i,k} J_{i,k}) \delta s_k} \right] (2\pi)^j \tag{52}$$

State:

$$p_{k,\ell+1} = \rho_{k,\ell+1} \bar{R} \, T_{k,\ell+1} \sum_{i=1}^{N} \frac{c_{i,k,\ell+1}}{M_i} \tag{53}$$

where

$$\bar{\tau}_k = \overline{2(2\pi r)^j u_k \rho_k \varepsilon_v} \; \overline{\frac{(u_{k+1} - u_k)}{(\dot{m}_{k+1} + \dot{m}_k)}} \tag{54}$$

$$\bar{Q}_k = \overline{2(2\pi r)^j u_k \rho_k \frac{c_{p_f} \varepsilon_v}{Pr_t}} \; \overline{\frac{(T_{k+1} - T_k)}{(\dot{m}_{k+1} + \dot{m}_k)}} \tag{55}$$

$$\bar{J}_{i,k} = \overline{2(2\pi r)^j u_k \rho_k \frac{Le_t \varepsilon_v}{Pr_t}} \; \overline{\frac{(c_{i,k+1} - c_{i,k})}{\dot{m}_{k+1} + \dot{m}_k}} \tag{56}$$

and where the operator Δ_k takes the difference in the bracketed quantity across the kth streamtube. In the above equations, a bar over a variable indicates that the average value of the variable during the step downstream is to be used. In Equations 51, 54, 55, and 56, a transformation similar to that used by von Mises in boundary layer studies has been applied to minimize the effects of large tube-to-tube variations of flow properties at hypersonic speeds in which velocity and streamline curvature vary slowly across the flow (37). This transformation is

$$\dot{m} = (2\pi)^j \int_{n_o}^{n} \rho u r^j dn \tag{57}$$

Also, in the above equations, the enthalpy is given by

$$h = \sum_{i=1}^{N} c_i h_i \tag{58}$$

where

$$h_i = \int_o^T c_{p_i} dT + h_i^\circ \qquad (59)$$

If the species specific heat is assumed to be a polynomial function of temperature,

$$c_{p_i} = a_{i2} + 2a_{i3}T + 3a_{i4}T^2 + \ldots + (j-1)a_{ij}T^{j-2} \qquad (60)$$

then

$$h = \sum_{i=1}^{N} c_i (\sum_j a_{ij}T^{j-1}) \qquad (61)$$

The selection of a suitable eddy viscosity, ε_v, is of utmost importance in all turbulent flow studies including the present one. Unfortunately, a wide variety of expressions have been suggested by various authors. For the special case of compressible turbulent mixing, most authors have modified the incompressible eddy viscosity expressions of Prandtl (48) either by performing a transformation which relates the compressible flow to an incompressible flow or by including a representative density (49). In the present study, the latter form of modification is used in an expression proposed by Edelman and Fortune (35). This expression, which is constant across the width of the mixing layer, is written as

$$\varepsilon_v = \frac{\Delta Y}{c} [(\rho u)_{max} - (\rho u)_{min}] + 0.0485 \quad [\frac{gm}{cm\ sec}] \qquad (62)$$

where ΔY is the width of the mixing layer, c is a constant, and

$(\rho u)_{max}$ and $(\rho u)_{min}$ are the maximum and minimum values, respectively, of ρu across the mixing layer.

In addition, the turbulent Lewis and Prandtl numbers are assumed constant throughout the viscous flowfield.

With the above assumptions, the 6 + N Equations 48, 49, 50, 51, 52, 53 and 61 contain 6 + N unknowns $p_{k,\ell+1}$, $T_{k,\ell+1}$, $\rho_{k,\ell+1}$, $u_{k,\ell+1}$, $h_{k,\ell+1}$, $A_{k,\ell+1}$, and $c_{i,k,\ell+1}$. Equation 51 can be solved directly for the streamline curvatures at the initial surface; and the location of the grid points at the new orthogonal surface can be obtained by stepping downstream a stable stepping distance δs_k using a circular arc or a straight line to approximate the streamline curvature. With the areas, $A_{k,\ell+1}$, thus determined, the remaining equations may be written:

Global continuity:

$$T_{k,\ell+1} = b_3\, p_{k,\ell+1}\, u_{k,\ell+1} \tag{63}$$

Species continuity:

$$c_{i,k,\ell+1} = b_5 + b_6\, \bar{\dot{W}} \tag{64}$$

Streamwise momentum:

$$p_{k,\ell+1} = b_1 - b_2\, u_{k,\ell+1} \tag{65}$$

Energy:

$$2h_{k,\ell+1} + u^2_{k,\ell+1} = b_4 \tag{66}$$

Enthalpy:

$$h_{k,\ell+1} = \sum_{i=1}^{N} c_{i,k,\ell+1}\left[\sum_j a_{ij}\, T_{k,\ell+1}^{j-1}\right] \tag{67}$$

where

$$b_1 = \left(\left[\Delta_k \overline{[r_k^j\ \tau_k\ \delta s_k]}\,(2\pi)^j + \dot{m}_k u_{k,\ell}\right] / \overline{A}_k\right) + p_{k,\ell}$$

$$b_2 = \dot{m}_k/\overline{A}_k$$

$$b_3 = A_{k,\ell+1}/\ \overline{R}\left(\sum_{i=1}^{N} \frac{c_{i,k,\ell+1}}{M_i}\right)\dot{m}_k \tag{68}$$

$$b_4 = 2\Delta_k\ \overline{\left[r_k^j(Q_k + [\tfrac{1}{2}(u_k^2 + u_{k+1}^2)]^{1/2}\ \tau_k + \sum_{i=1}^{N} h_{i,k}J_{i,k})\delta s_k\right]}\ \frac{(2\pi)^j}{\dot{m}_k}$$

$$+ 2h_{k,\ell} + u_{k,\ell}^2$$

$$b_5 = \Delta_k\overline{\left[r_k^j\ J_{i,k}\ \delta s_k\right]}\ \frac{(2\pi)^j}{\dot{m}_k} + c_{i,k,\ell}$$

$$b_6 = \frac{\overline{\delta s}_k}{\overline{u}_k}$$

The terms just defined (b_1, b_2, b_3, b_4, b_5, and b_6) are functions of known quantities except for b_3 which contains the unknowns $c_{i,k,\ell+1}$. After combining Equations 63, 65, 66, and 67, a single equation in terms of the unknowns $u_{k,\ell+1}$ and $c_{i,k,\ell+1}$ is obtained:

$$2\sum_{i=1}^{N} c_{i,k,\ell+1}\left(\sum_j a_{ij}\left[b_3(b_1 - b_2\ u_{k,\ell+1})u_{k,\ell+1}\right]^{j-1}\right) + u_{k,\ell+1}^2 - b_4 = 0 \tag{69}$$

For nonreacting flows, Equations 64 can be solved immediately for the unknown mass fractions, $c_{i,k,\ell+1}$, at the new surface. However, if the flow is in chemical nonequilibrium, the computation becomes much more difficult.

During the last seven years, considerable progress has been made in coupling nonequilibrium chemical reactions with one-dimensional fluid flows. The nonlinear ordinary differential equations which result have been solved by a variety of different techniques. Initial studies made by Emanuel and Vincenti (50), Emanuel (51), Pergament (52), and others (53, 54) used conventional integration methods such as the Euler, Runge-Kutta, or Predictor-Corrector techniques. Unfortunately, the maximum step size allowed when using these techniques is severely limited by stability criteria. This is in spite of the fact that the resulting curves of mass fractions versus time are very smooth when properly computed. Because of these small step sizes, the computer times required are enormous, particularly for near-equilibrium flow. As a result, this type of calculation gained a certain amount of notoriety for being very costly. Then in 1965, Moretti (41) developed a technique in which he linearized the species continuity equations and uncoupled them from the other flow equations. The resulting linear, non-homogeneous system of ordinary differential equations were then solved by finding the complex eigenvalues and eigenvectors of an Nth order real matrix. As a result of this linearization, Moretti was able to use step sizes which were several orders of magnitude larger than had previously been used. Yet he obtained identical results. Since that time, DeGroat and Abbett (55) and Magnus and Schechter (56) have removed the difficulties involved in finding the complex eigenvalues and eigenvectors in Moretti's scheme. They have used a truncated power series for the unknown molar concentrations in conjunction with either a collocation or subdomain method to evaluate the coefficients in the power series.

In the present study, the small step sizes required for chemical nonequilibrium are not necessarily critical since the flow is always supersonic. Consequently, a small step in time may still result in a sizeable step in space. Furthermore, the step size is already limited by the stability requirements for the Lagrangian finite-difference technique.

For these reasons, the methods that have been developed in the present study to calculate the unknown mass fractions when the flow is in chemical nonequilibrium are not as elegant as those appearing in Reference (56). The primary goal here was to find simple methods which required little computer time and yet produced accurate results when using the maximum step size allowed by the Lagrangian finite-difference technique.

Two methods have been used to solve Equations 64 for the unknown mass fractions at the new surface. The first method, called Technique 1, is an explicit computation. In this method, Equation 64 is solved by assuming that all terms on the right hand side are known, thus permitting an immediate solution. On the first pass, the terms on the right hand side are evaluated using known quantities from the previously calculated surface while on the second pass, average values between the old and new surfaces are used.

The second method, called Technique 2, is an implicit procedure which allows larger step sizes than Technique 1 but requires more computation per step. The average species production term $\bar{\dot{W}}_{i,k}$, in Equation 64 is written as

$$\bar{\dot{W}}_{i,k} = \frac{M_i}{\bar{\rho}_k} \sum_{m=1}^{M'} (\nu''_{i,m} - \nu'_{i,m}) \overline{\left[k_{f_m} \prod_{j=1}^{N'} Y_{j,k}^{\nu'_{j,m}} - k_{b_m} \prod_{j=1}^{N'} Y_{j,k}^{\nu''_{j,m}} \right]} \tag{70}$$

where the bar over a quantity indicates that the average value during the step is to be used. A typical average molar concentration expression arising from the two product terms in Equation 70 is written as

$$\overline{Y_{i,k}Y_{j,k}} = \left(\frac{Y_{i,k,\ell} + Y_{i,k,\ell+1}}{2} \right) \left(\frac{Y_{j,k,\ell} + Y_{j,k,\ell+1}}{2} \right)$$

$$= \frac{1}{4}(Y_{i,k,\ell}Y_{j,k,\ell} + Y_{i,k,\ell}Y_{j,k,\ell+1} + Y_{i,k,\ell+1}Y_{j,k,\ell}$$

$$+ Y_{i,k,\ell+1}Y_{j,k,\ell+1}) \tag{71}$$

The nonlinear terms, $Y_{i,k,\ell+1}Y_{j,k,\ell+1}$, are linearized using Moretti's linearization technique:

$$Y_{i,k,\ell+1}Y_{j,k,\ell+1} = -Y_{i,k,\ell}Y_{j,k,\ell} + Y_{i,k,\ell+1}Y_{j,k,\ell} + Y_{i,k,\ell}Y_{j,k,\ell+1} \tag{72}$$

so that Equation 71 can be rewritten as

$$\overline{Y_{i,k}Y_{j,k}} = \frac{1}{2}(Y_{i,k,\ell}Y_{j,k,\ell+1} + Y_{i,k,\ell+1}Y_{j,k,\ell}) \tag{73}$$

Other expressions that arise from the two product terms in Equation 70 are linearized in a similar manner. For example,

$$\overline{Y_{i,k}Y_{i,k}} = Y_{i,k,\ell}Y_{i,k,\ell+1} \tag{74}$$

and

$$\overline{Y_{i,k}Y_{j,k}Y_{c,k}} = \frac{Y_{c,k,\ell}}{2}(Y_{i,k,\ell}Y_{j,k,\ell+1} + Y_{i,k,\ell+1}Y_{j,k,\ell}) \tag{75}$$

where $Y_{c,k,\ell}$ is the molar concentration of a catalyst which is assumed constant during the step.

After linearizing the species production term, the set of species continuity equations become

$$\frac{\bar{\rho}_k Y_{i,k,\ell+1}}{\rho_{k,\ell+1}} = \frac{\bar{\rho}_k}{\rho_{k,\ell}} Y_{i,k,\ell} + \frac{\bar{\rho}_k}{M_i} \Delta_k \left[\overline{r_k^j \; J_{i,k} \; \delta s_k} \right] \frac{(2\pi)^j}{\dot{m}_k} + b_6 \left[f(Y_{j,k,\ell}, Y_{j,k,\ell+1}, \bar{T}_k) \right] \tag{76}$$

On the first pass the temperature and density at the previously calculated surface are used for $\bar{T}_k$ and $\bar{\rho}_k$, while on the second pass the average values between the old and new surface are used. Likewise, the density at the old surface is used for $\rho_{k,\ell+1}$ on the first pass while the value at the new surface is used on the second pass. These assumptions eliminate all unknowns from Equations 76 except the molar concentrations terms $Y_{i,k,\ell+1}$, so that this set of equations can be formally written as

$$[X][Y] = [Z] \tag{77}$$

where [Y] is the column matrix of unknown molar concentrations at the new surface. This set of linear algebraic equations can be readily solved using standard techniques such as the Gauss-Jordan method (57). The mass fractions at the new surface are then calculated from

$$c_{i,k,\ell+1} = \frac{M_i Y_{i,k,\ell+1}}{\rho_{k,\ell+1}} \tag{78}$$

where $\rho_{k,\ell+1}$ is set equal to $\rho_{k,\ell}$ on the first pass and is set equal to $\rho_{k,\ell+1}$ on the second pass.

With the mass fractions at the new surface thus determined, Equation 69 can be solved for the unknown velocity at the new surface $u_{k,\ell+1}$ using the Newton-Raphson method (57). If, however, the specific heats are temperature independent, then Equation 69 reduces to

$$\left[2\left(\sum_{i=1}^{N} c_{i,k,\ell+1}\ a_{i2}\right) b_2 b_3 - 1\right] u_{k,\ell+1}^2$$

$$- \left[2\left(\sum_{i=1}^{N} c_{i,k,\ell+1}\ a_{i2}\right) b_1 b_3\right] u_{k,\ell+1} \tag{79}$$

$$- \left[2\left(\sum_{i=1}^{N} c_{i,k,\ell+1}\ a_{i1}\right) - b_4\right] = 0$$

which can be solved directly for $u_{k,\ell+1}$ using the guadratic formula. With the velocity determined, the remaining unknowns pressure, temperature, and density can be computed using Equations 65, 63, and 53, respectively.

The numerical technique just described is often applied to flowfields in which the mesh size is relatively large so that the replacement of derivatives with differences is somewhat questionable. However, since the increases in mass, momentum, and energy in one tube are exactly balanced by the losses in adjacent tubes, the integrated relations are satisfied over the entire flow.

Boundary Conditions

The natural coordinate system employed in the present analysis allows several different types of boundary conditions to be easily incorporated. First of all, since both a symmetry axis and a solid wall are streamlines, they can be used directly for either the outer or inner boundaries. Likewise, dividing streamlines can be used directly if the outside ambient pressure or Newtonian impact pressure, is specified:

$$p_b = p_\infty + \rho_\infty u_\infty^2 \sin^2 (\theta_b - \theta_\infty) \tag{80}$$

When shocks are present in the flowfield, it is much easier to compute the flowfield sequentially. That is, the undisturbed region is calculated initially and then the shocked region is computed using the shock as a boundary discontinuity which propagates into the previously calculated undisturbed region. The shock angle and conditions downstream of the shock can be obtained by requiring that the shock turn the flow parallel to the nearest streamline being carried in the calculation. For temperature independent specific heat the shock angle σ can be determined from (58)

$$\zeta^3 + b\zeta^2 + c\zeta + d = 0 \tag{81}$$

where

$$\begin{aligned}
\zeta &= \sin^2 \sigma \\
b &= -\frac{(M_\infty^2 + 2)}{M_\infty^2} - \gamma \sin^2 \delta' \\
c &= \frac{2 M_\infty^2 + 1}{M_\infty^4} + \left[\frac{(\gamma+1)^2}{4} + \frac{\gamma-1}{M_\infty^2}\right] \sin^2 \delta' \\
d &= -\frac{\cos^2 \delta'}{M_\infty^4}
\end{aligned} \tag{82}$$

using the Newton-Raphson method. The pressure and temperature downstream of the shock are then obtained from

$$p_s = p_\infty \left[\frac{2\gamma M_\infty^2 \zeta - (\gamma-1)}{\gamma+1}\right] \tag{83}$$

and

$$T_s = T_\infty \left(\frac{p_s}{p_\infty}\right) \frac{(\gamma-1) M_\infty^2 \zeta + 2}{(\gamma+1) M_\infty^2 \zeta} \tag{84}$$

If the specific heat is temperature dependent, then the following conservation equations across the shock must be solved iteratively for the unknown temperature, pressure, and shock angle;

$$p_s - p_\infty - \rho_\infty u_\infty^2 \sin^2 \sigma \left[1 - \frac{\tan(\sigma-\delta')}{\tan \sigma}\right] = 0 \tag{85}$$

$$h(T_s) = h(T_\infty) + \frac{1}{2} u_\infty^2 \left[1 - \frac{\cos^2 \sigma}{\cos^2 (\sigma-\delta')}\right] \tag{86}$$

$$p_\infty T_\infty \operatorname{ctn} \sigma + T_\infty \rho_\infty u_\infty^2 \sin \sigma \cos \sigma \left[1 - \frac{\tan(\sigma-\delta')}{\tan \sigma}\right] \tag{87}$$

$$- p_\infty T_s \operatorname{ctn} (\sigma-\delta') = 0$$

The additional mass that is added to the flowfield as the shock propagates upstream is accounted for by adding new streamtubes. A streamtube is added only when its mass flow is comparable to the mass flow in the previous set of streamtubes. This prevents the step size from becoming increasingly smaller as the size of the added streamtubes decreases. In order to reduce the magnitude of the calculation, streamtubes are combined if desired by a process which conserves mass, momentum, and energy. This is particularly useful in the case of shocked flows where streamtubes are continually being added.

COMPUTER PROGRAM

The Multitube computer program written by Boynton (37) was adapted in its original form for use on the Iowa State University Computation Center IBM 360-65 computer. This highly versatile program, written in Fortran IV language, applies the Lagrangian finite-difference technique just described to axisymmetric or two-dimensional, steady, nonreacting supersonic flowfields. The transport processes are laminar with the diffusion treated in full multicomponent formalism including pressure and thermal diffusion if desired. The program, which consists of a main program and 14 subroutines, was originally written to calculate high altitude rocket exhaust plumes but can be applied as well to nozzle flows and flows around bodies. In order to accomplish this, a number of options have been provided which include viscid or inviscid flow, constant or temperature dependent specific heat, and rotational or irrotational flow. In addition, the inner boundary can be a wall (axis), a free boundary, or a shock while the outer boundary can be any one of these or a Newtonian pressure boundary. The shocks are allowed to propagate into either a uniform or a nonuniform flowfield. However, there can be at most one external nonuniform flowfield. The program can accomodate 60 streamtubes, 10 species, and a fourth order enthalpy fit for each of the species.

In the present study, the original Multitube program has been modified, using the theory previously discussed, to enable it to calculate nonequilibrium, chemically reacting flowfields with turbulent transport. These additions allow the program to be used to calculate intermediate altitude rocket exhaust plumes.

The conversion of the original program required one additional subroutine, two additional named commons, and extensive revisions to several of the existing subroutines. However, since the original program was written in building-block form many of the original subroutines did not require any alterations. The inputs to the new program, given in Appendix B, have been arranged so that the inputs to the old program need not be altered. This is done by setting an input parameter called IBURN equal to zero to indicate frozen flow, whereas a nonequilibrium case is indicated by setting IBURN equal to one. If IBURN equals one, the program will read in the additional information required such as stoichiometric coefficients, reaction rates, etc. At present the program can handle 20 chemical reactions, 10 species, and 5 catalysts. The 5 catalysts are assumed to have molar concentrations which are linear combinations of the molar concentrations of the 10 species.

In a similar manner, an input parameter called ITURB is used to indicate turbulent transport so that additional information can be inputed such as the turbulent Lewis and Prandtl numbers.

In order to calculate chemical nonequilibrium flows, two new subroutines have been written called CHEM. However, only one such subroutine can be used at any given time. The first subroutine uses Technique 1 while the second uses Technique 2.

The CHEM subroutine using Technique 1 employs a method first used by Emanuel and Vincenti (50) to reduce numerical round-off errors when computing the contribution of the species production term. In addition, this technique can be applied to any set of 20 chemical reactions without modifying the subroutine. On the other hand, the subroutine using Technique 2 must be partially rewritten for each different set of chemical reactions used. This rewriting primarily consists of changing the elements in the matrices. While this CHEM subroutine is not as versatile as the CHEM subroutine employing Technique 1, it will allow a much larger stable step size.

The turbulent transport subroutine, TRANSP, calculates the eddy viscosity using Equation 62. The value calculated is assumed constant across the mixing layer between successive orthogonal surfaces. The constant in the eddy viscosity expression currently used is optional and is inputed into the program. An entirely different expression for the eddy viscosity could easily be added to the program if desired.

The original subroutine STABLE, which computes the maximum frozen flow stable step size, has not been altered to calculate the stable step size for chemical nonequilibrium flows. However, smaller step sizes can be specified by merely changing an input parameter called ALPHAH. This parameter indicates what fraction of the maximum frozen flow stable step size is actually to be taken.

A simplified flow chart for the new Multitube program appears in Appendix B along with a listing of the program. Included in this listing are both CHEM subroutines for Techniques 1 and 2. However, only one may be included for each computer run. The CHEM subroutine listed for Technique 2 was written for the chemical reactions that occur in the intermediate altitude rocket exhaust plume sample case which is discussed later. In addition, both TRANSP and FLUX subroutines for laminar and turbulent transport are also included, but only one pair can be used at any given time.

The above discussion has been limited primarily to the modifications performed on the original Multitube program. Further information concerning the original program can be found in Reference (37). Both the original and the modified versions of the Multitube program are on file in the Department of Aerospace Engineering, Iowa State University, Ames, Iowa.

RESULTS AND CONCLUSIONS

Nonequilibrium Chemical Reactions in Streamtube Held at Constant Pressure

The logic that was introduced into the Multitube program in order to calculate chemical nonequilibrium flows was initially checked by letting the modified program correlate the results that Magnus and Schechter (56) obtained in their investigation. One of the sample cases appearing in Reference (56) is an inviscid, constant pressure, constant enthalpy, chemical nonequilibrium streamtube analysis of a hydrogen-air chemistry system. The application of the modified program to this sample case is discussed in the present section.

The species considered in the present chemistry system are H, O, H_2O, OH, O_2, H_2, and N_2 which are numbered 1, 2, 3, 4, 5, 6, and 7, respectively. The first 6 species are assumed to react while the seventh species, nitrogen, is considered inert. The following reactions are assumed significant:

$$
\begin{aligned}
H + O_2 &\rightleftharpoons OH + O \\
O + H_2 &\rightleftharpoons OH + H \\
H_2 + OH &\rightleftharpoons H + H_2O \\
2OH &\rightleftharpoons O + H_2O \\
H_2 + X &\rightleftharpoons 2H + X \\
H_2O + X &\rightleftharpoons OH + H + X \\
OH + X &\rightleftharpoons O + H + X \\
O_2 + X &\rightleftharpoons 2O + X
\end{aligned}
\tag{88}
$$

where X is a catalyst whose molar concentration is assumed equal to the sum of the molar concentrations of the seven species. The corresponding reaction rates for the above reactions are given in Appendix A. The linear enthalpy relationships used in Reference (56) for each of the seven species are

$$
\begin{aligned}
h_1 &= 50{,}621 + 4.968\ T \\
h_2 &= 3633.64 + 0.31113\ T \\
h_3 &= -3602.92 + 0.67856\ T \\
h_4 &= 329.7 + 0.48735\ T \\
h_5 &= -122.16 + 0.28216\ T \\
h_6 &= -1869.5 + 4.0975\ T \\
h_7 &= -134.4 + 0.3072\ T
\end{aligned}
\tag{89}
$$

in units of cal./gm.

The initial streamtube conditions for the present sample case are

$$
\begin{aligned}
T &= 1100°K \\
p &= 0.135 \text{ atm.}
\end{aligned}
\tag{90}
$$

with initial mass fractions:

$$
\begin{aligned}
c_1 &= 0.1277 \times 10^{-8} \\
c_2 &= 0.2106 \times 10^{-8} \\
c_3 &= 0.0 \\
c_4 &= 0.0 \\
c_5 &= 0.2291069 \\
c_6 &= 0.014319 \\
c_7 &= 0.756574
\end{aligned}
\tag{91}
$$

In order to solve this constant pressure, nonequilibrium streamtube test case using the modified Multitube program, the STEP subroutine had to be partially rewritten. This subroutine normally calculates the flow properties at the new orthogonal surface using the theory previously discussed. However, since the pressure and enthalpy are held constant in the present problem, a different analysis must be employed.

In the new analysis, the mass fractions at the next step downstream are calculated by calling subroutine CHEM as before. With the new mass fractions thus calculated, the new temperature is determined in subroutine STEP using

$$T_{k,\ell+1} = \frac{h - \sum_{i=1}^{N} c_{i,k,\ell+1}\, a_{i1}}{\sum_{i=1}^{N} c_{i,k,\ell+1}\, a_{i2}} \tag{92}$$

since the enthalpy remains constant. The density is next determined using the equation of state:

$$\rho_{k,\ell+1} = \frac{p}{\sum_{i=1}^{N} \frac{c_{i,k,\ell+1}}{M_i} \bar{R} T_{k,\ell+1}} \tag{93}$$

In order to set up the calculation using the modified Multitube program, the inner boundary of the streamtube is taken as a symmetry axis while a dividing streamline is used for the outer boundary. The pressure acting on the outside of the dividing streamline is then set equal to the constant pressure in the streamtube. Consequently, the dividing streamline is always a straight line so that the computation will be stable for any step size used in the Lagrangian finite-difference computation. However, the step size is limited by the stability requirements of the nonequilibrium chemical analysis.

As mentioned previously, the coefficients in the matrices for subroutine CHEM (Technique 2) must be redetermined for each new chemical system. The coefficients in the matrices for the present chemical system appear in Appendix A. Since no diffusion is assumed to take place in this problem, the coefficient matrix can be reduced to a 4 by 4 matrix using the following equations which express the conservation of the number of atoms of oxygen and hydrogen:

$$Y_{5,k,\ell+1} = -\frac{1}{2}(Y_{2,k,\ell} + Y_{3,k,\ell} + Y_{4,k,\ell}) + \frac{\rho_{k,\ell+1}}{\rho_{k,\ell}}\Big[Y_{5,k,\ell} + \frac{1}{2}(Y_{2,k,\ell} + Y_{3,k,\ell} + Y_{4,k,\ell})\Big] \tag{94}$$

$$Y_{6,k,\ell+1} = -\frac{1}{2}(Y_{1,k,\ell} + Y_{4,k,\ell} + 2Y_{3,k,\ell}) + \frac{\rho_{k,\ell+1}}{\rho_{k,\ell}}\Big[Y_{6,k,\ell} + Y_{3,k,\ell} + \frac{1}{2}(Y_{1,k,\ell} + Y_{4,k,\ell})\Big] \tag{95}$$

A comparison between results using the modified Multitube program and Reference (56) is shown in Figures 3, 4, 5, and 6. In Figures 3, 4, and 5, the mass fractions of H, H_2, and OH are

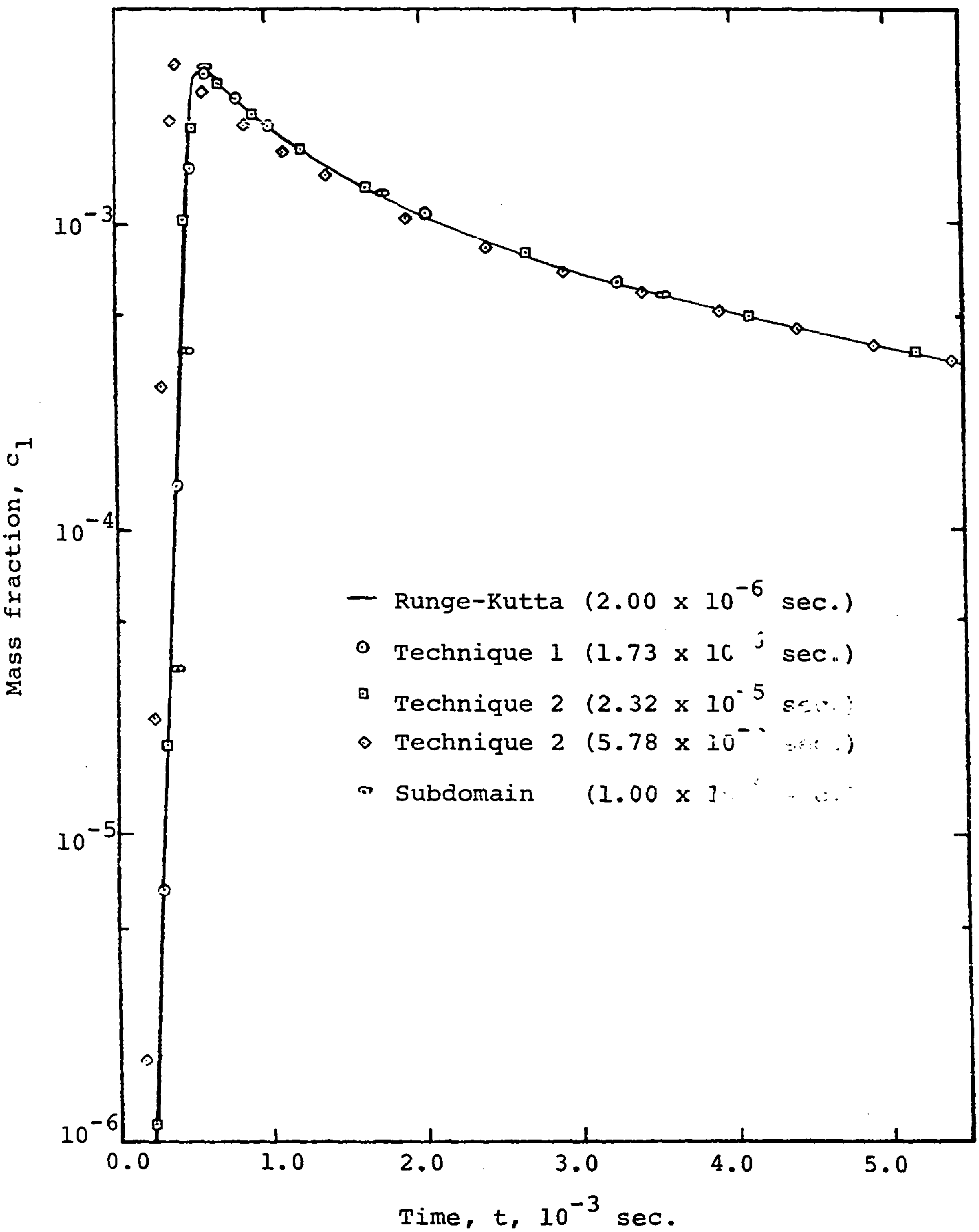

Figure 3. Mass fraction of H versus time

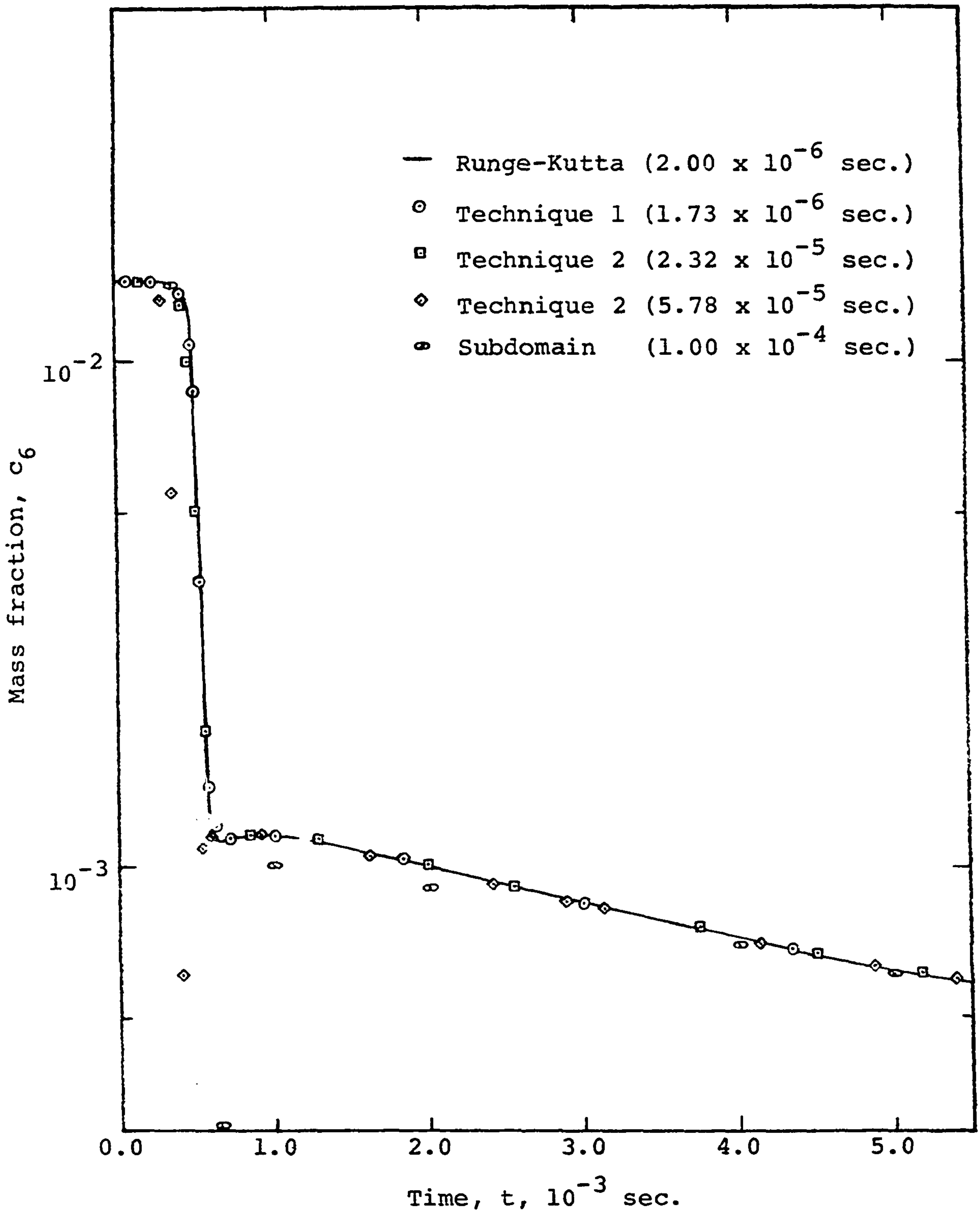

Figure 4. Mass fraction of H_2 versus time

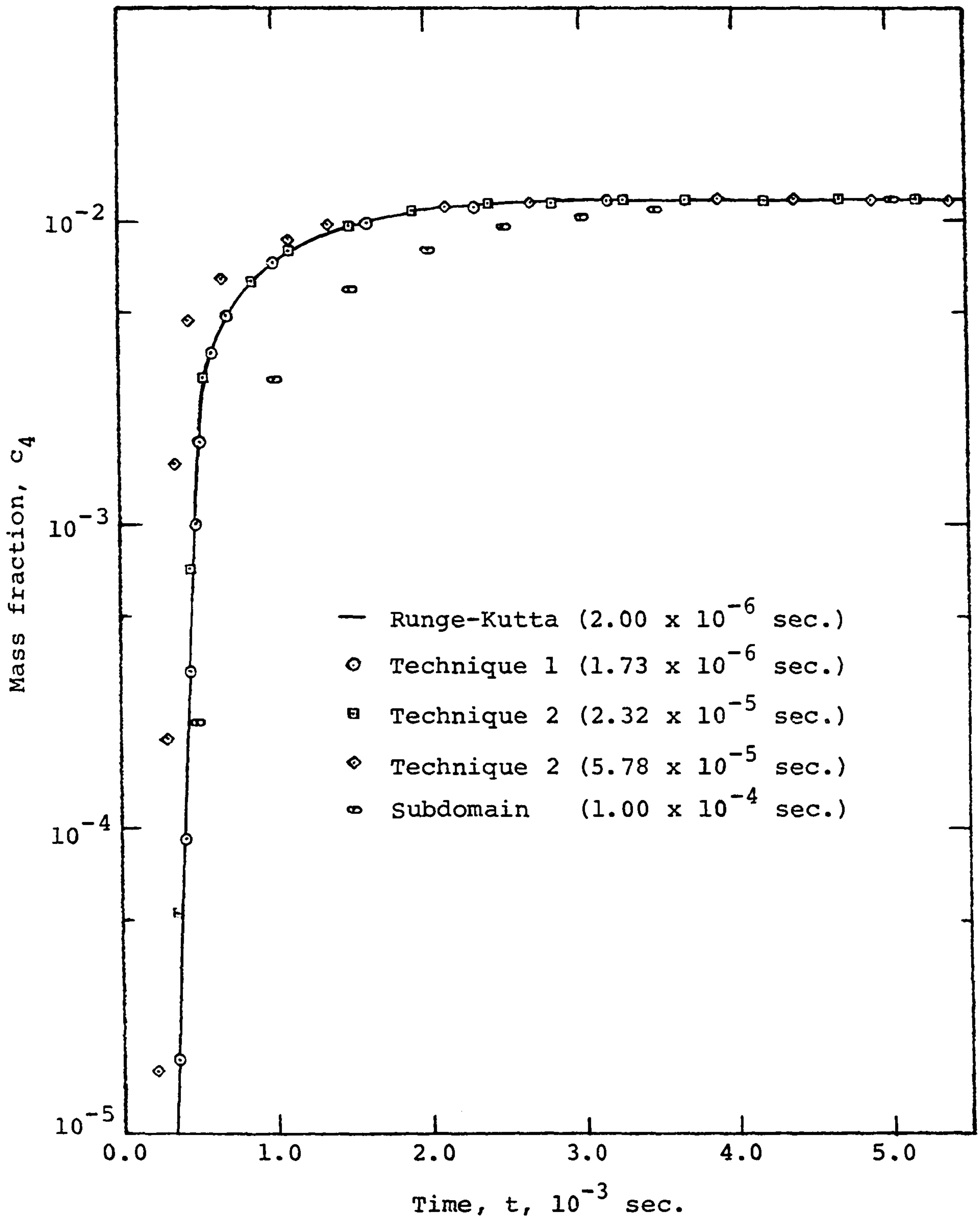

Figure 5. Mass fraction of OH versus time

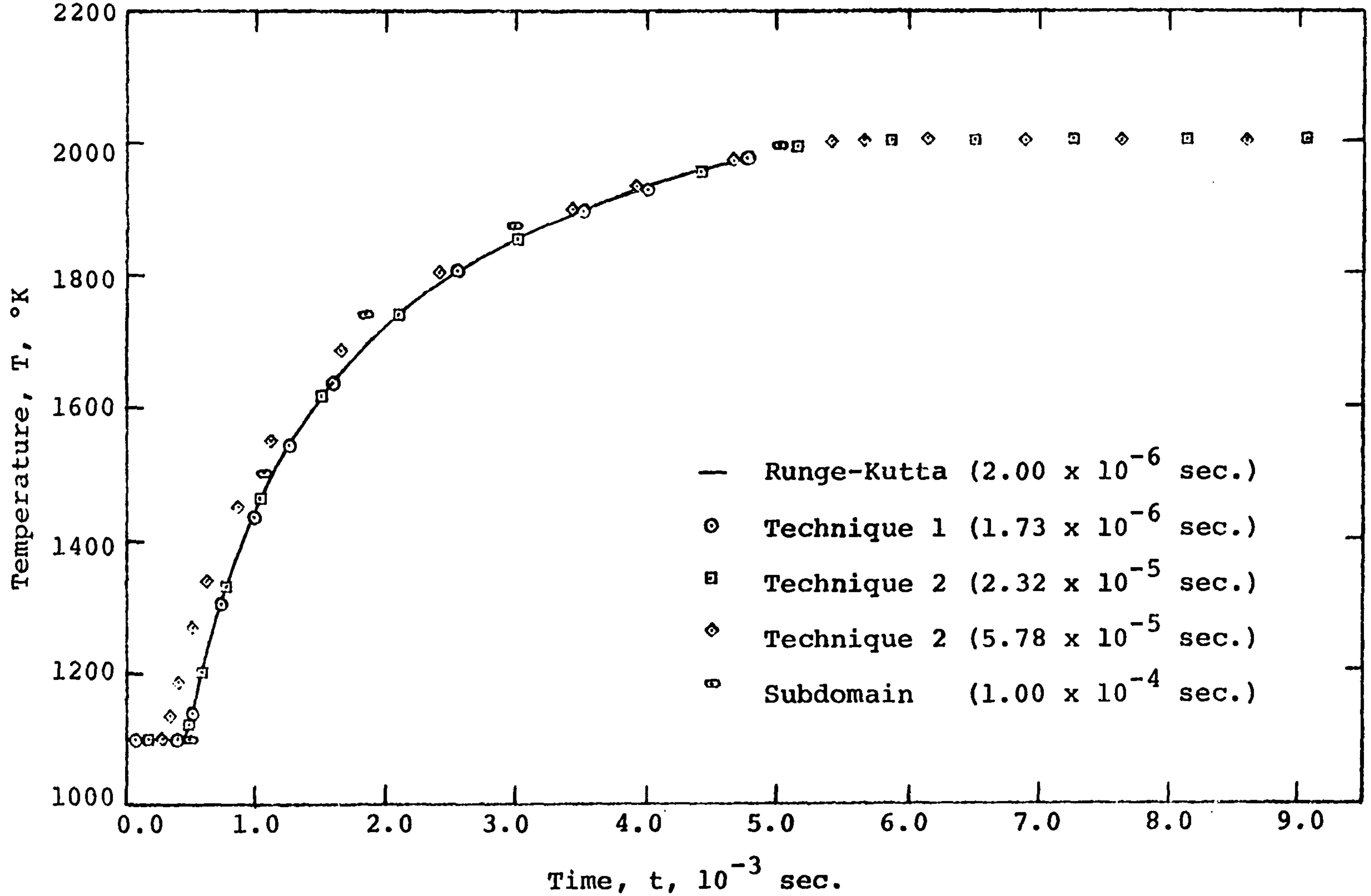

Figure 6. Temperature versus time

plotted as functions of time. The Runge-Kutta solution from Reference (56) is shown as a solid line. This method is unstable for step sizes slightly larger than 2×10^{-6} sec. while Technique 1 was found to be unstable for initial step sizes larger than 1.86×10^{-6} sec. However, excellent results were obtained, as shown, using a step size of 1.73×10^{-6} sec. in Technique 1. This corresponds to an initial ALPHAH (fraction of the maximum frozen flow step size allowed) equal to 0.3 if a representative velocity of 120,000 cm./sec. is assumed.

On the other hand, Technique 2 appears to remain stable for any step size but deviates from the correct solution as the step size is increased. However, even for a very large step size, the correct solution is attained as equilibrium is approached. Appearing in Figures 3, 4, and 5 are the results of two different calculations using Technique 2. The first calculation uses a step size of 2.32×10^{-5} sec. while the second uses a step size of 5.78×10^{-5} sec. The corresponding ALPHAH's for these two step sizes are 4 and 10, respectively, if a velocity of 120,000 cm./sec. is again assumed. For an initial step size as large as 5.78×10^{-5} sec., Technique 2 gives fairly accurate results. This is approximately 29 times larger than the maximum step size permitted using the Runge-Kutta method. Also shown in Figures 3, 4, and 5 are the values computed in Reference (56) using the subdomain method with a step size of 50 times the maximum allowable Runge-Kutta step size.

In Figure 6, the temperature is plotted as a function of time. Here again excellent results were obtained using either Techniques 1 or 2. The temperature appears to approach an equilibrium value after about 6 x 10^{-3} sec.

It should also be mentioned, that the numerical results obtained using Techniques 1 and 2 were identical to three significant figures if the same step size was specified. The computer time required to calculate a specific number of steps using Technique 1 was only slightly less than that required by Technique 2. However, since Technique 2 allows a much larger step size, the computer time required for a typical reaction is substantially less.

Intermediate Altitude Rocket Exhaust Plumes Assuming Inviscid, Nonreacting Flow

In this section, a comparison is made between the results obtained using the Multitube program and the Lockheed method of characteristics program (17) discussed earlier. The flowfield example used in the comparison is that of an intermediate altitude rocket exhaust plume calculated by assuming inviscid, nonreacting flow. In addition, the exhaust gases are not allowed to mix with the air and the position of the dividing streamline is determined by assuming that the pressure acting on the outside of the dividing streamline is the Newtonian shock layer pressure.

The initial conditions assumed for both calculations are the representative Thor nozzle exit conditions from Reference (2). These one-dimensional exit conditions are:

$$\begin{aligned} M_e &= 2.986 \\ p_e &= 0.617 \text{ atm.} \\ \bar{V}_e &= 270{,}000 \text{ cm./sec.} \\ T_e &= 2060°\text{K} \\ \gamma &= 1.210 \\ c_{p_f} &= 0.453 \text{ cal./gm.°K} \\ \theta_e &= 0° \\ r_e &= 57.91 \text{ cm.} \end{aligned} \tag{96}$$

The species present in the nozzle flow are H, H_2, H_2O, OH, O, O_2, CO, and CO_2 which are numbered 1, 2, 3, 4, 5, 6, 7, and 8, respectively. In addition, the mass fractions at the nozzle exit are

$$\begin{aligned} c_1 &= 0.000239 \\ c_2 &= 0.00713 \\ c_3 &= 0.2850 \\ c_4 &= 0.00195 \\ c_5 &= 0.000086 \\ c_6 &= 0.00035 \\ c_7 &= 0.2950 \\ c_8 &= 0.4099 \end{aligned} \tag{97}$$

The rocket exhaust plume is calculated at the intermediate altitude of 150,000 feet for a Thor missile flying a typical trajectory. The corresponding external freestream conditions at this altitude are

$$
\begin{aligned}
M_\infty &= 5.117 \\
p_\infty &= 1.3425 \times 10^{-3} \text{ atm.} \qquad (98) \\
\overline{V}_\infty &= 167{,}500 \text{ cm./sec.} \\
T_\infty &= 266.15°K
\end{aligned}
$$

In addition, the composition of the air is assumed to consist of 23.1% oxygen and 76.9% nitrogen by weight.

The position of the jet shock and dividing streamline calculated by the Lockheed method of characteristics program for the conditions given above is shown in Figure 7. For this calculation, 40 initial data points were used with many of these points located near the nozzle lip. Also shown in Figure 7 is the location of the triple point found by using the empirical technique suggested by Eastman and Radtke (9).

In order to calculate the present inviscid nozzle exhaust plume flowfield using the Multitube program, the exhaust flow must be first expanded to a vacuum. This calculation is necessary in order to describe the region bounded by the jet shock and the Mach disc. In this type of calculation, the flow at the nozzle lip expands very rapidly to very low pressures causing the streamtube areas to increase greatly in size. Consequently, a large number of very small streamtubes must be

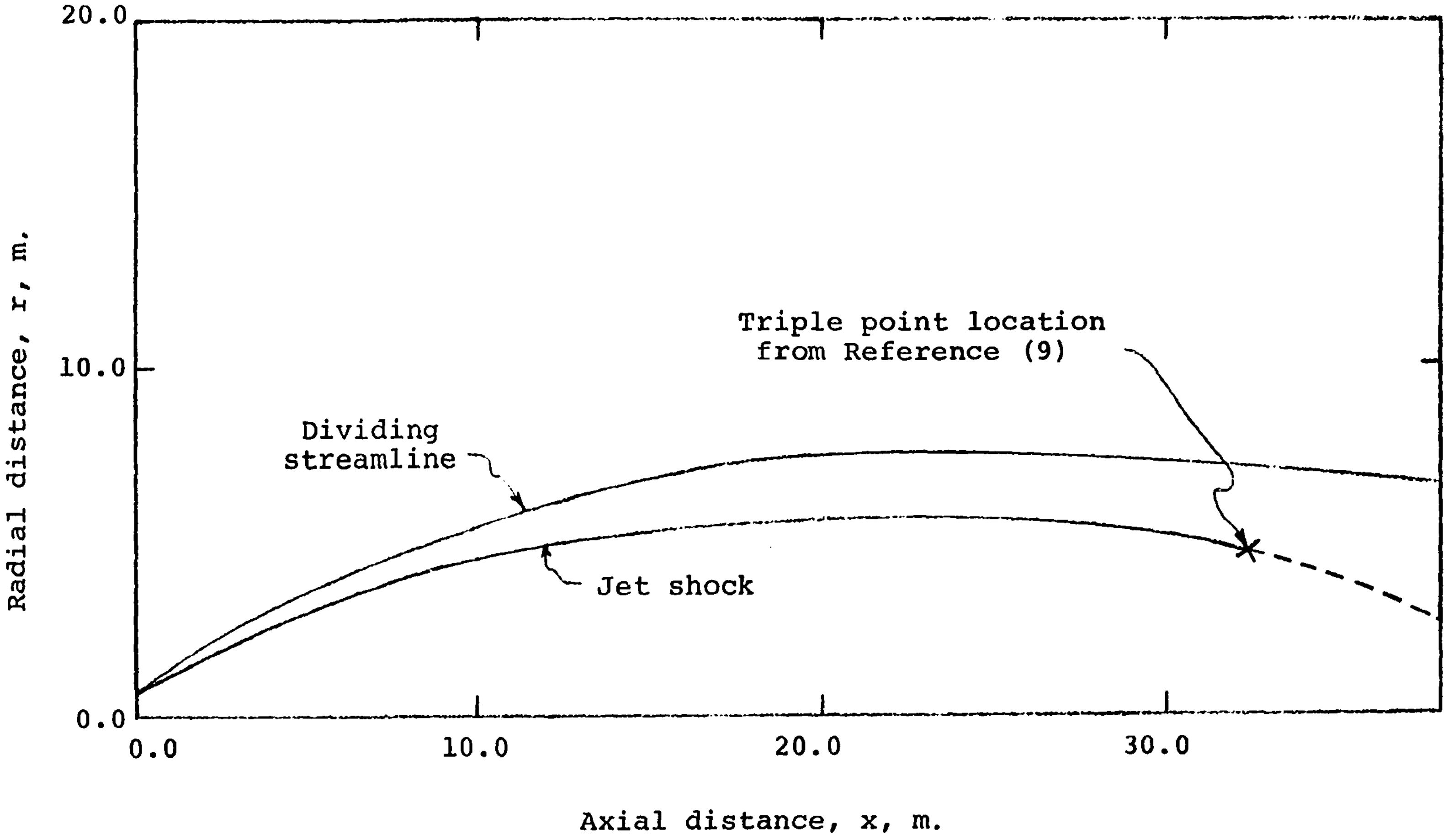

Figure 7. Thor plume at 150,000 feet (Lockheed method of characteristics program)

placed near the nozzle lip to initiate the calculation. For the present exhaust expansion, 43 initial streamtubes were selected with many clustered near the nozzle lip. The output of this calculation using the Multitube program is punched on computer cards for later use in computing the jet shock layer.

Some of the streamlines and orthogonal surfaces for this expansion are shown in Figure 8. Note that as the distance from the nozzle increases, the orthogonal surfaces become nearly circular as the streamlines straighten. This confirms the often used assumption that at large distances from a nozzle operating in a vacuum, the exhaust flow can be assumed to emanate from a source.

The next step in calculating the inviscid plume flowfield using the Multitube program is to locate the initial positions of the jet shock and dividing streamline. Boynton describes a technique in Reference (37) which locates the jet shock using the Multitube program. In order to apply this technique, the nozzle exhaust gases are expanded to the Newtonian shock layer pressure. At a point near the nozzle exit, the outermost streamline will almost intersect the adjacent streamline. Boynton states that the jet shock would pass through this point if it were present. With the jet shock approximately located in this manner, the flow angle is determined by matching the jet and freestream Newtonian pressures.

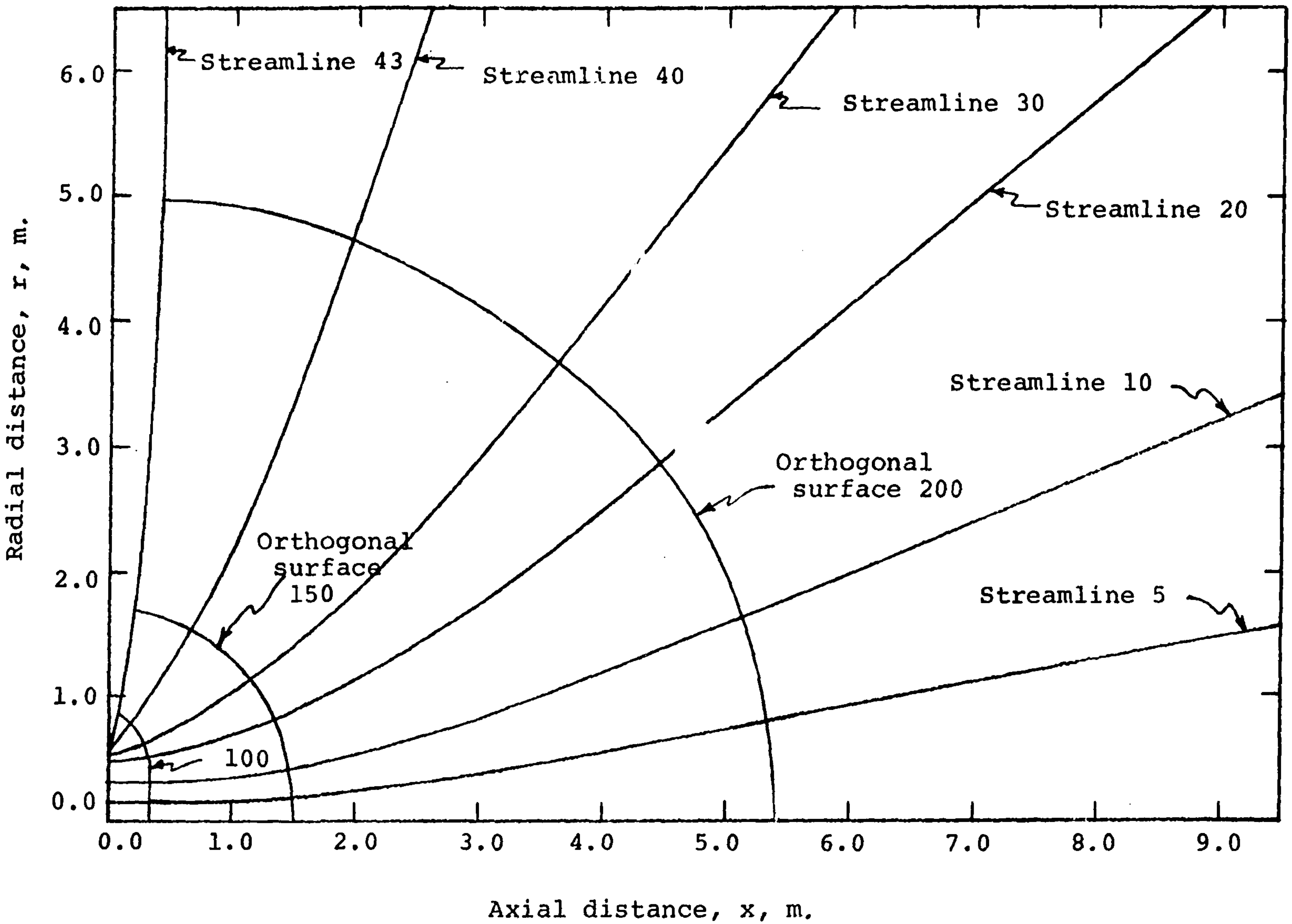

Figure 8. Thor exhaust expanded to vacuum

In the present calculations, the initial jet shock and dividing streamline locations, required for the Multitube calculation, were obtained using the Lockheed method of characteristics computer program. This is a very convenient and accurate method for locating the initial jet shock and dividing streamline positions.

Once the jet shock is located, the jet shock layer can be calculated letting the jet shock propagate into the previously calculated undisturbed inviscid flow. For the present test case, one streamtube was used to represent the shock layer at the initial orthogonal surface. This was possible because the initial mass flow in the shock layer near the nozzle exit is very small in comparison to the mass flow in the jet shock layer further downstream. As a result, it was found that the flow properties in the jet shock layer far downstream were not very dependent on the initial conditions.

For the present jet shock layer computations, the input parameter FSTEP was varied between 5 and 15. Essentially, this parameter indicates the number of streamtubes to be carried in the calculation if they are of similar $\dot{m}/r$. Since streamtubes are continually being added to the calculation as the shock propagates upstream, other streamtubes must be combined to satisfy the limitation imposed by FSTEP. It was found, that a large value of FSTEP gives a better resolution of the flowfield, but does not appreciably change the location of either the jet shock or dividing streamline.

The location of the dividing streamline and jet shock determined by the Multitube program for FSTEP equal to 5 is shown in Figure 9, along with the results obtained using the Lockheed method of characteristics program. The results from the two computer programs compare quite favorably.

In Figure 10, the pressures calculated by the two computer programs along orthogonal surface A in Figure 9 have been compared. The two results agree remarkably well.

The Lockheed method of characteristics program is much better suited for the particular inviscid flowfield computation considered in the present section. However, for higher altitude plumes, where the Mach number becomes quite large and the characteristic mesh becomes very compressed, the Multitube program becomes the more efficient method for the present type of problem.

Intermediate Altitude Rocket Exhaust Plumes Assuming Chemical Nonequilibrium Reactions and Turbulent Transport

The turbulent mixing of the air and exhaust gases which was ignored in the previous intermediate altitude rocket exhaust plume study is now considered. The mixing layer is assumed to extend from the jet shock to the air shock. In this region, turbulent diffusive transport in conjunction with nonequilibrium chemical reactions is assumed.

In order to start the calculation, an initial data line (orthogonal surface) is required. This data line must extend

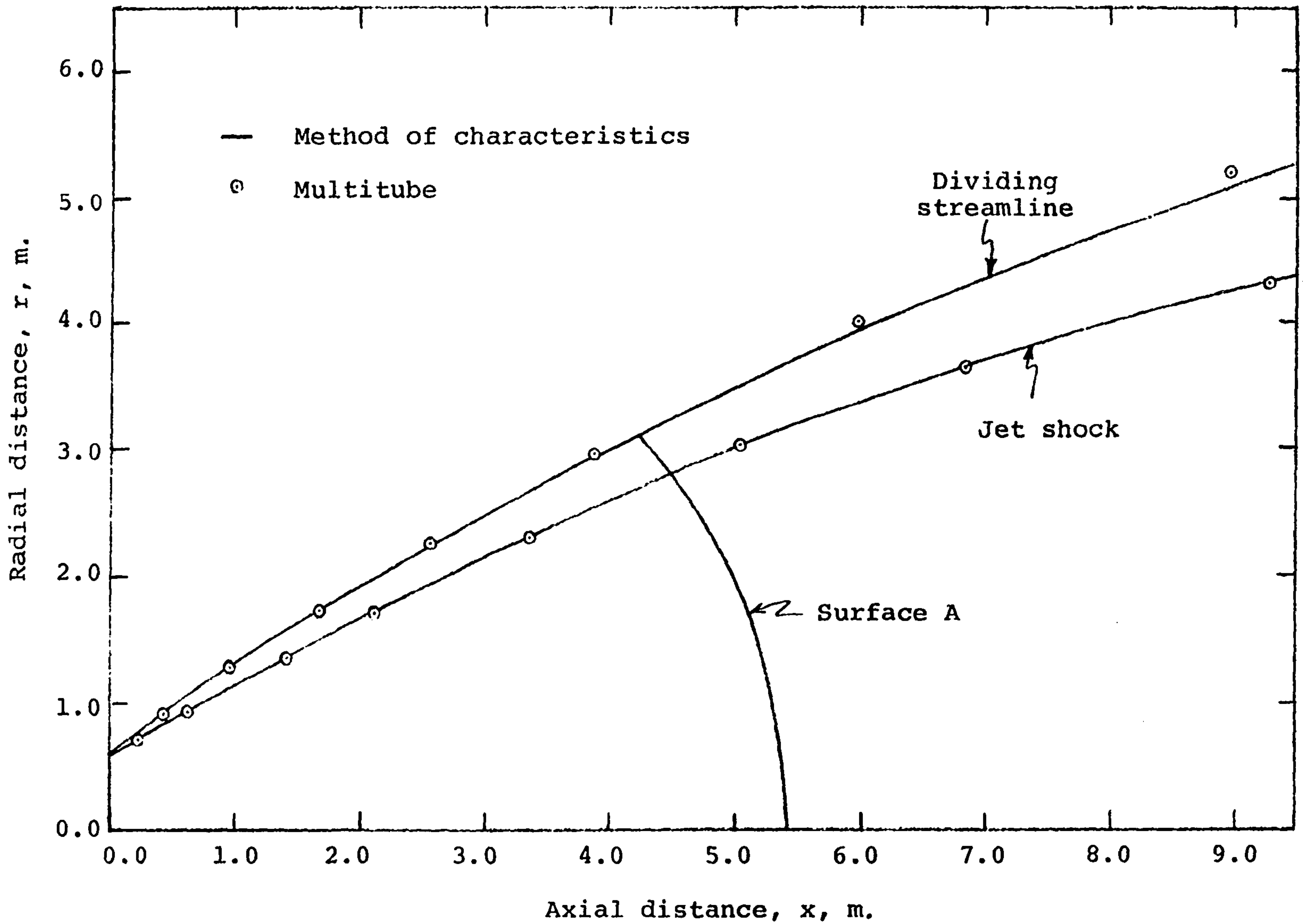

Figure 9. Thor plume at 150,000 feet

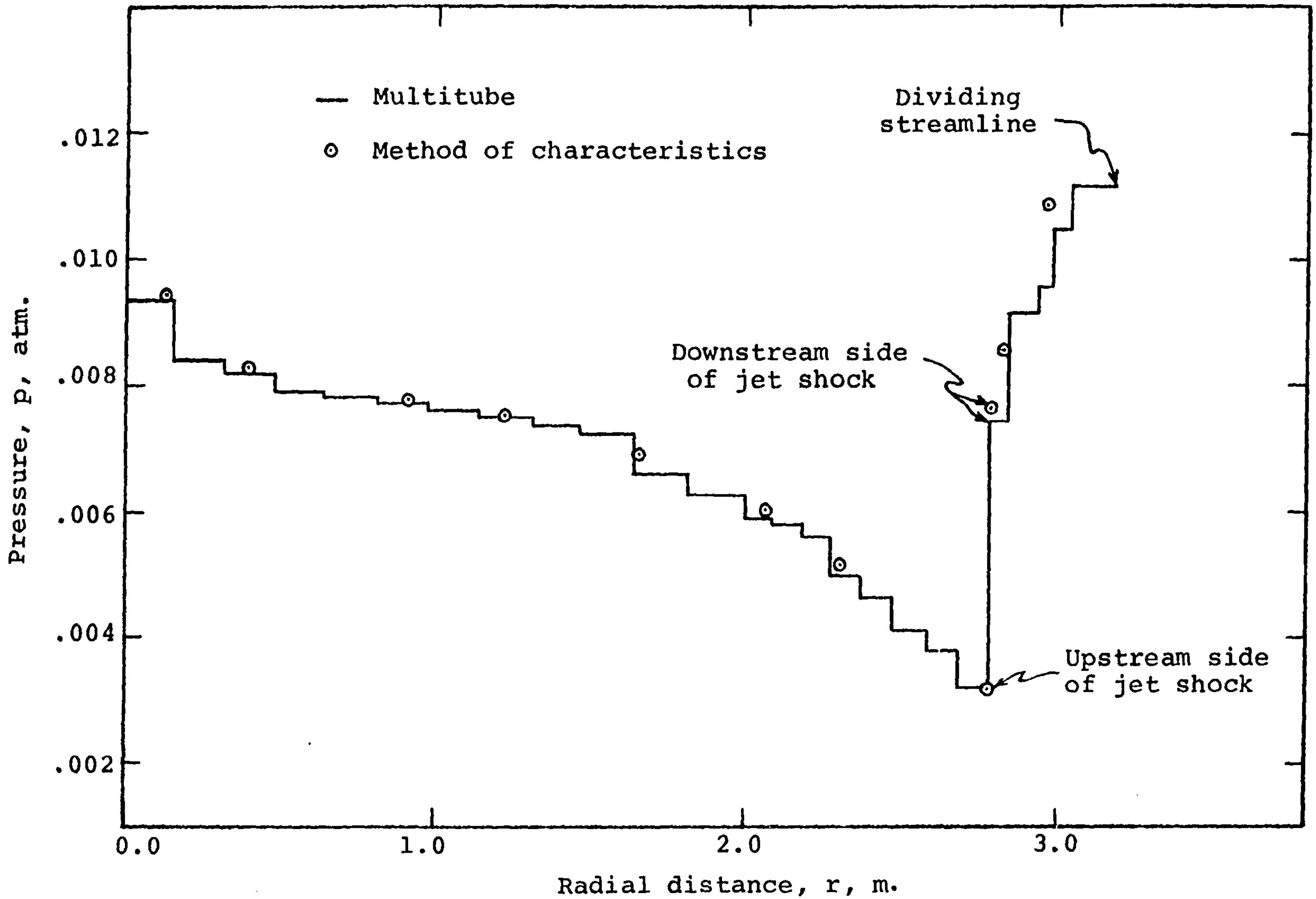

Figure 10. Pressure along Surface A

from the jet shock to the air shock. The initial data line used in the previous calculation, which extended from the initial jet shock position to the dividing streamline, may be used for part of the initial data line required in the present calculation. However, the flow conditions along the remaining segment of the initial data line, which lies in the air shock layer, must be computed. In this study these initial flow conditions have been determined by calculating the flow in the air shock layer from the nozzle lip to the initial data line. In this calculation, the flow is assumed inviscid and nonreacting and the dividing streamline found previously by the Lockheed method of characteristics program is used as a fixed inner boundary. The position of this dividing streamline is determined in the method of characteristics program by assuming that the air shock layer is a Newtonian shock layer. Consequently, the calculation of the flow in the air shock layer is only an approximation since the position of this dividing streamline is held fixed. However, since only the region in the air shock layer near the nozzle exit is computed, where the mass flow is quite small in comparison with the mass flow further downstream, it was found that the initial conditions determined by this method are quite adequate.

If the initial angle of the dividing streamline at the nozzle lip is less than the shock detachment wedge angle, then the initial slope of the air shock and the flow properties downstream of the shock can be readily determined. Since the flow is then everywhere supersonic, the air shock layer can be computed by the Multitube program letting the air shock propagate into the external uniform airstream. On the other hand, if the air shock detaches from the nozzle lip, then the subsequent calculation is complicated by the fact that a subsonic region exists and methods other than the Lagrangian finite-difference technique must be employed for the subsonic region. For the present Thor exit conditions, it was found that the air shock remains attached to the nozzle lip at the 150,000 ft. altitude currently considered, but detaches from the nozzle lip before an altitude of 200,000 ft. is reached.

In Figure 11, the initial portion of the air shock layer is shown. It has been calculated in the manner discussed above. For this calculation, one initial streamtube was used and the input parameter FSTEP was set equal to 4. Also shown in Figure 11 is the location of the initial data line (orthogonal surface) which is used in subsequent calculations.

In Figure 12, the air and jet shock layers have been computed using the initial data line just determined. This inviscid, nonreacting computation uses a dividing streamline to separate the two shock layers. The jet shock propagates into the previously calculated nonuniform internal inviscid flow while the air shock propagates into the uniform external flow. The species present in the jet shock layer are H, H_2, H_2O, OH,

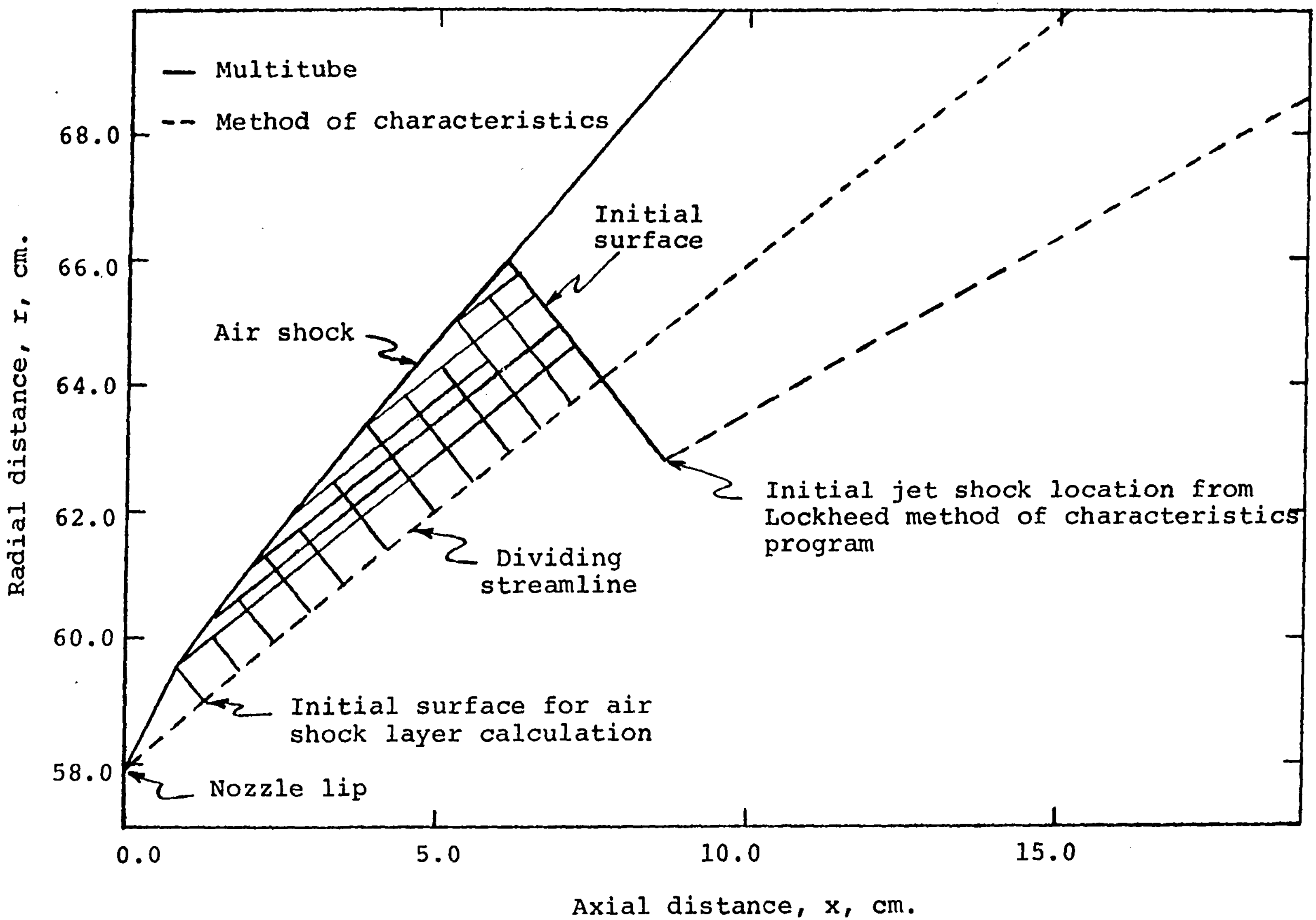

Figure 11. Air shock layer

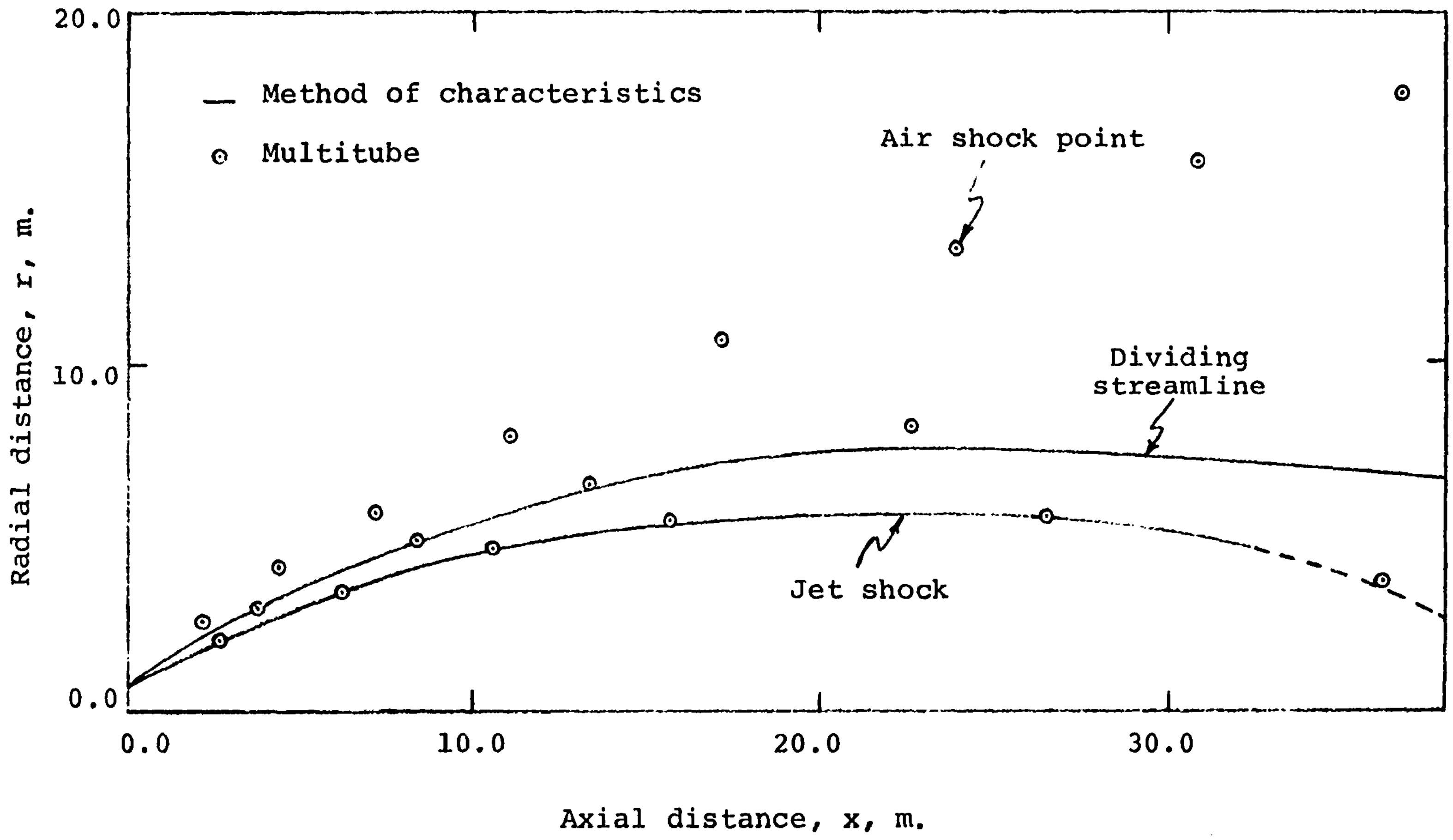

Figure 12. Thor plume at 150,000 feet

O, O_2, CO, and CO_2 which are numbered 1, 2, 3, 4, 5, 6, 7, and 8, as before. The species assumed present in the air shock layer are O_2 and N_2. The additional species N_2 is given the identification number 9. The linear enthalpy relationships given by Equations 89 were used for the corresponding species in the present calculation. The linear enthalpy relationships used for the two additional species, CO and CO_2, are

$$\begin{aligned} h_7 &= -1077 + 0.309\ T \\ h_8 &= -2290 + 0.324\ T \end{aligned} \tag{99}$$

in units of cal./gm. These linear enthalpy relationships were used in order to reduce computer time. In addition, the input parameter FSTEP was set equal to 15.

Also shown in Figure 12 is the dividing streamline and jet shock location determined by the Lockheed method of characteristics computer program. The closeness of the results indicates that the Newtonian pressure boundary assumption which is often used in plume studies is quite reasonable.

The above inviscid calculation was repeated assuming chemical nonequilibrium flow. As would be expected, the chemical reactions taking place were virtually frozen. This occurs because the nozzle flow is usually frozen by the time it reaches the nozzle exit and of course the air will not dissociate to any measurable degree for the temperatures encountered in the inviscid air shock layer.

The reactions considered are those suggested by Edelman and Fortune (35) for a rocket exhaust plume without turbine exhaust present.

$$
\begin{aligned}
CO + OH &\rightleftharpoons CO_2 + H \\
OH + H_2 &\rightleftharpoons H + H_2O \\
2OH &\rightleftharpoons H_2O + O \\
O + H_2 &\rightleftharpoons OH + H \\
H + O_2 &\rightleftharpoons OH + O \\
O + H + X &\rightleftharpoons OH + X \\
2O + X &\rightleftharpoons O_2 + X \\
2H + X &\rightleftharpoons H_2 + X \\
H + OH + X &\rightleftharpoons H_2O + X
\end{aligned}
\tag{100}
$$

In these reactions, X is a catalyst whose molar concentration is assumed equal to the sum of the molar concentrations of the 9 species. The reaction rates from Reference (35) are given in Appendix A along with the elements in the matrices for the CHEM (Technique 2) subroutine.

In the next calculation performed, the dividing streamline was removed and the air and exhaust gases were allowed to mix turbulently but not react. The turbulent Lewis and Prandtl numbers chosen were 1.20 and 0.80, respectively. In addition, the constant in the eddy viscosity expression was set equal to 900. In order to satisfy the requirement of a binary mixture, it can be assumed that H and H_2 are the light particles and the remaining species are the heavy particles. The results of this viscous calculation and the preceding inviscid calculation are shown in Figure 13. The air and jet shocks are displaced outward only slightly from their inviscid locations.

The above viscous calculation was repeated assuming chemical nonequilibrium flow. Both CHEM subroutines utilizing Techniques 1 and 2 were used in the calculation. The step size specified in both calculations was 0.9 times the maximum frozen flow step size computed in subroutine STABLE. The results obtained using the two versions of CHEM were practically indistinguishable. In addition, virtually no difference in results occurred if a smaller step size was specified.

The locations of the air and jet shocks computed by the present viscous, reacting calculations were identical to the locations determined in the previous viscous, nonreacting calculations. The species mass fractions along Surface B in Figure 13 are shown in Figures 14 to 22. Also included in these figures are the mass fractions along Surface B for the inviscid, nonreacting, and viscous, nonreacting flow cases previously discussed. It is apparent from these figures that the reactions are nearly frozen in the Thor plume at 150,000 ft. However, a few of the species such as atomic oxygen (O) varied considerably between the nonreacting and nonequilibrium reacting, viscous calculations.

The temperatures along surface B for the three different calculations are shown in Figure 23. It is apparent, that the

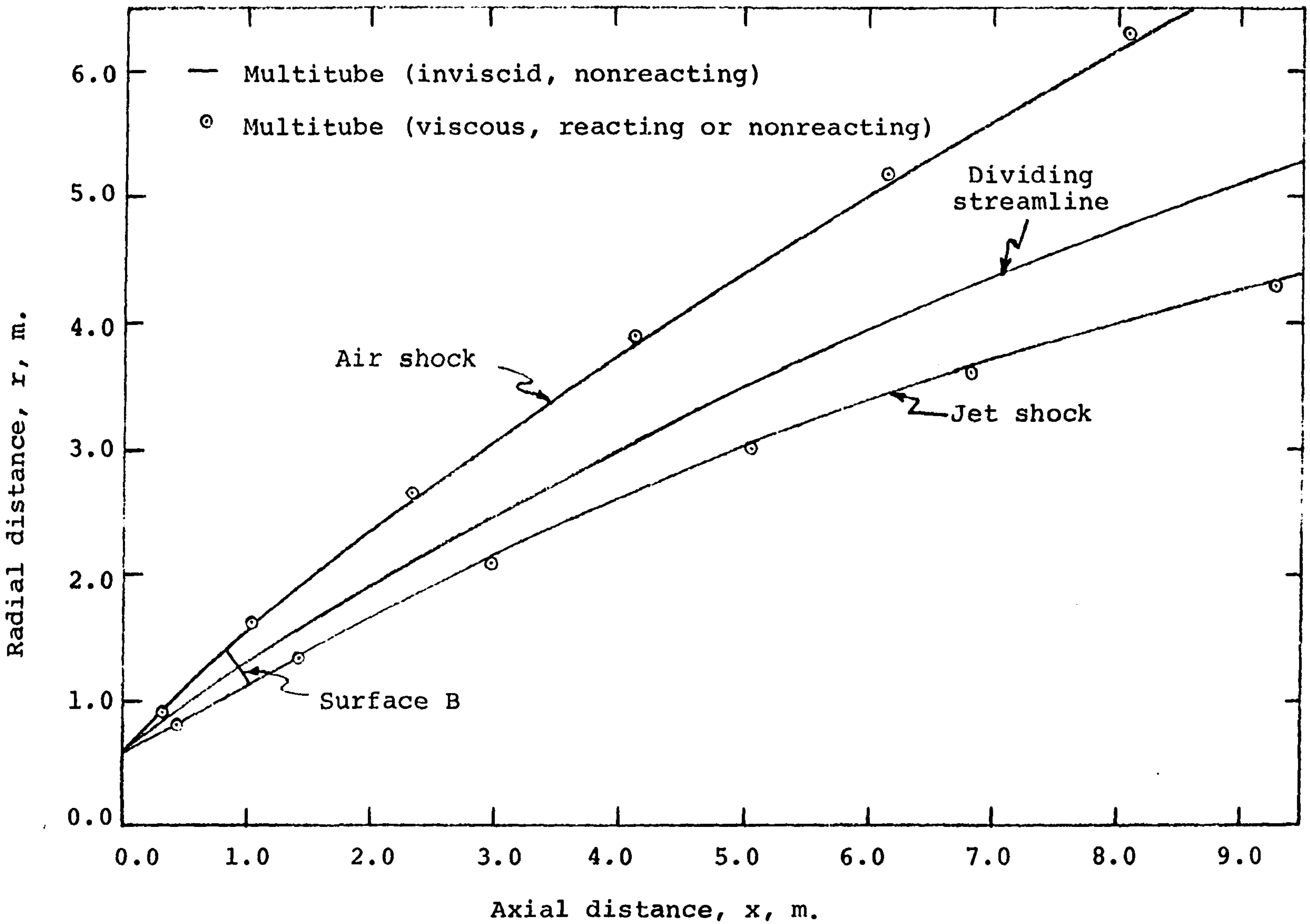

Figure 13. Thor air-jet shock layers at 150,000 feet

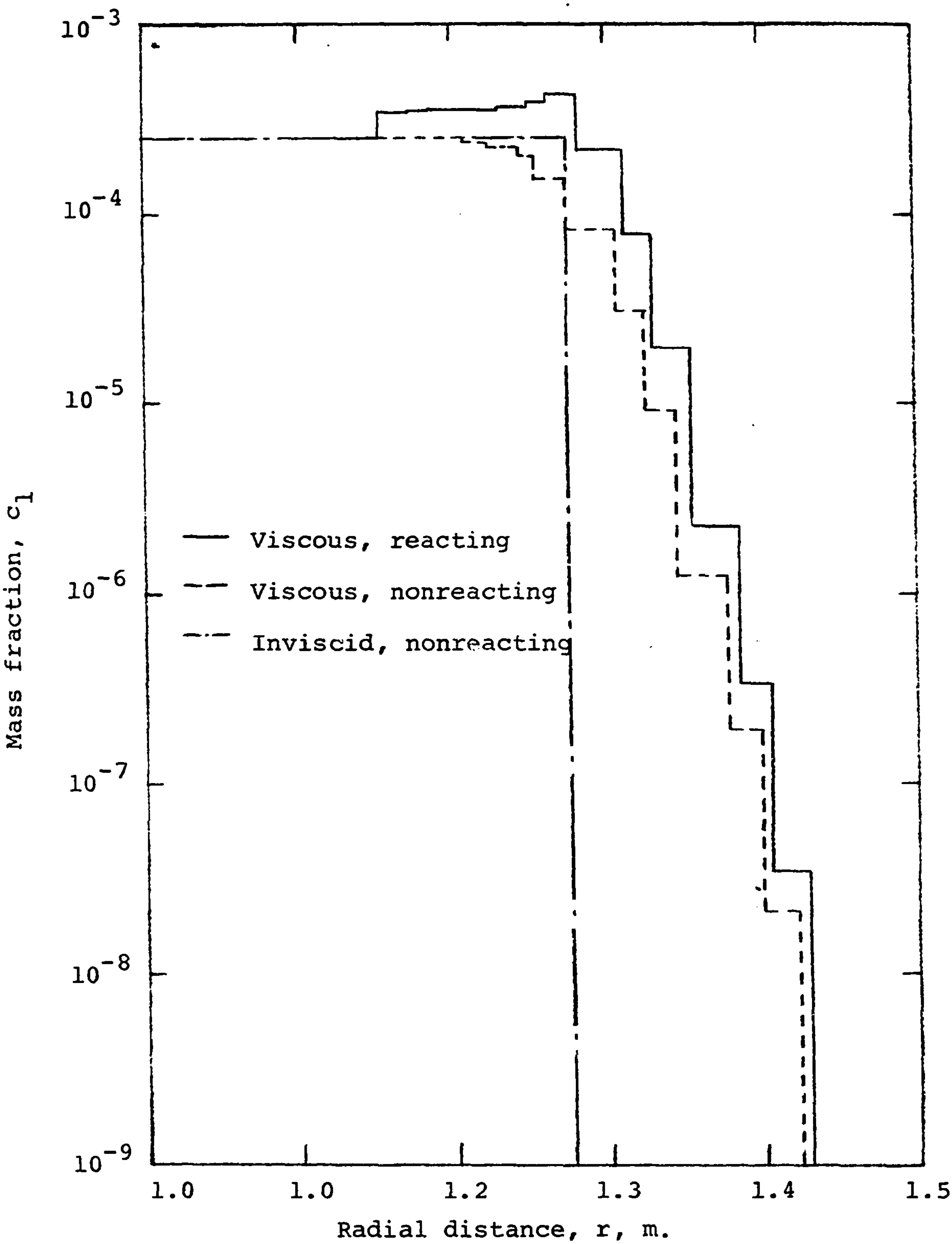

Figure 14. Mass fraction of H along Surface B

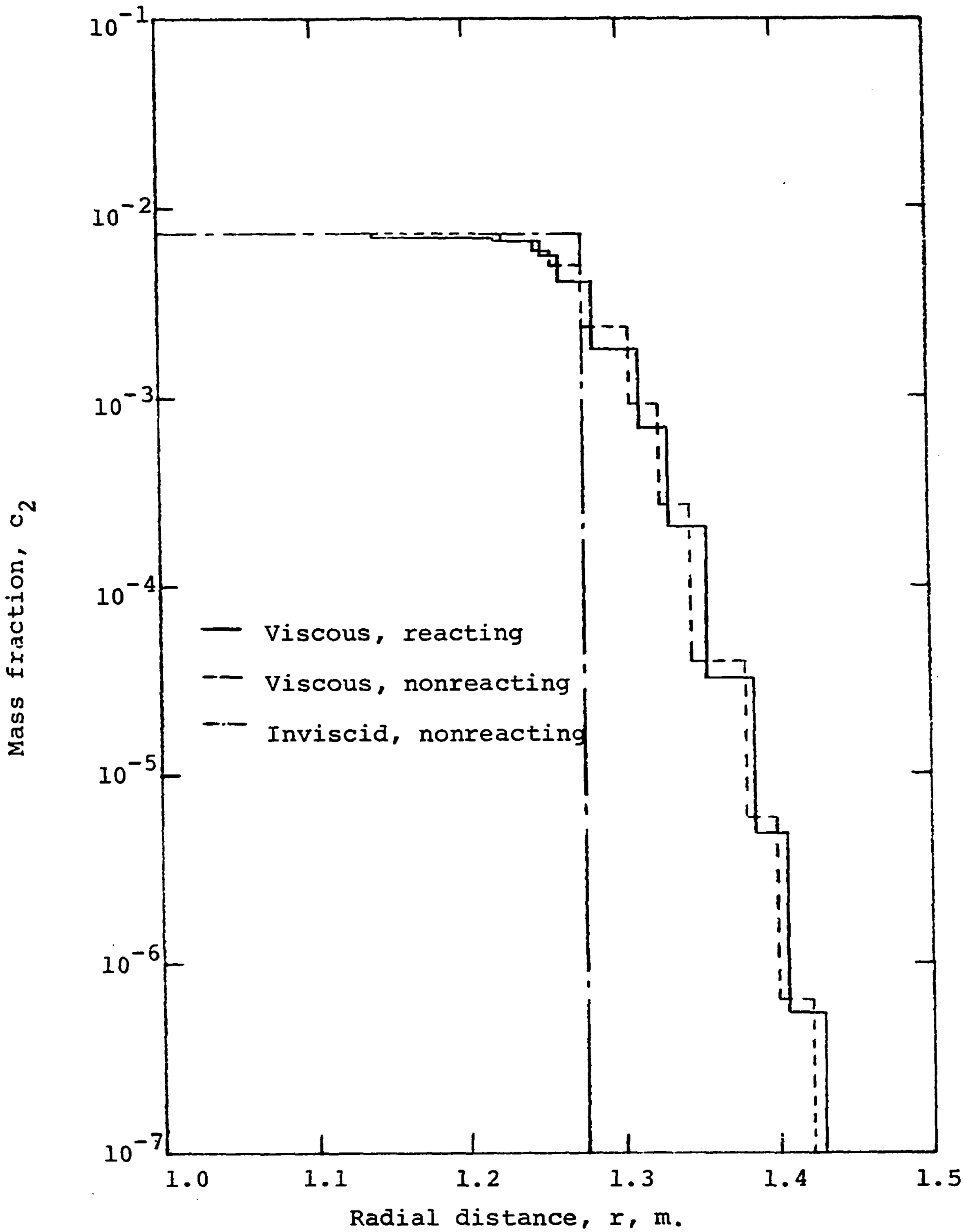

Figure 15. Mass fraction of H_2 along Surface B

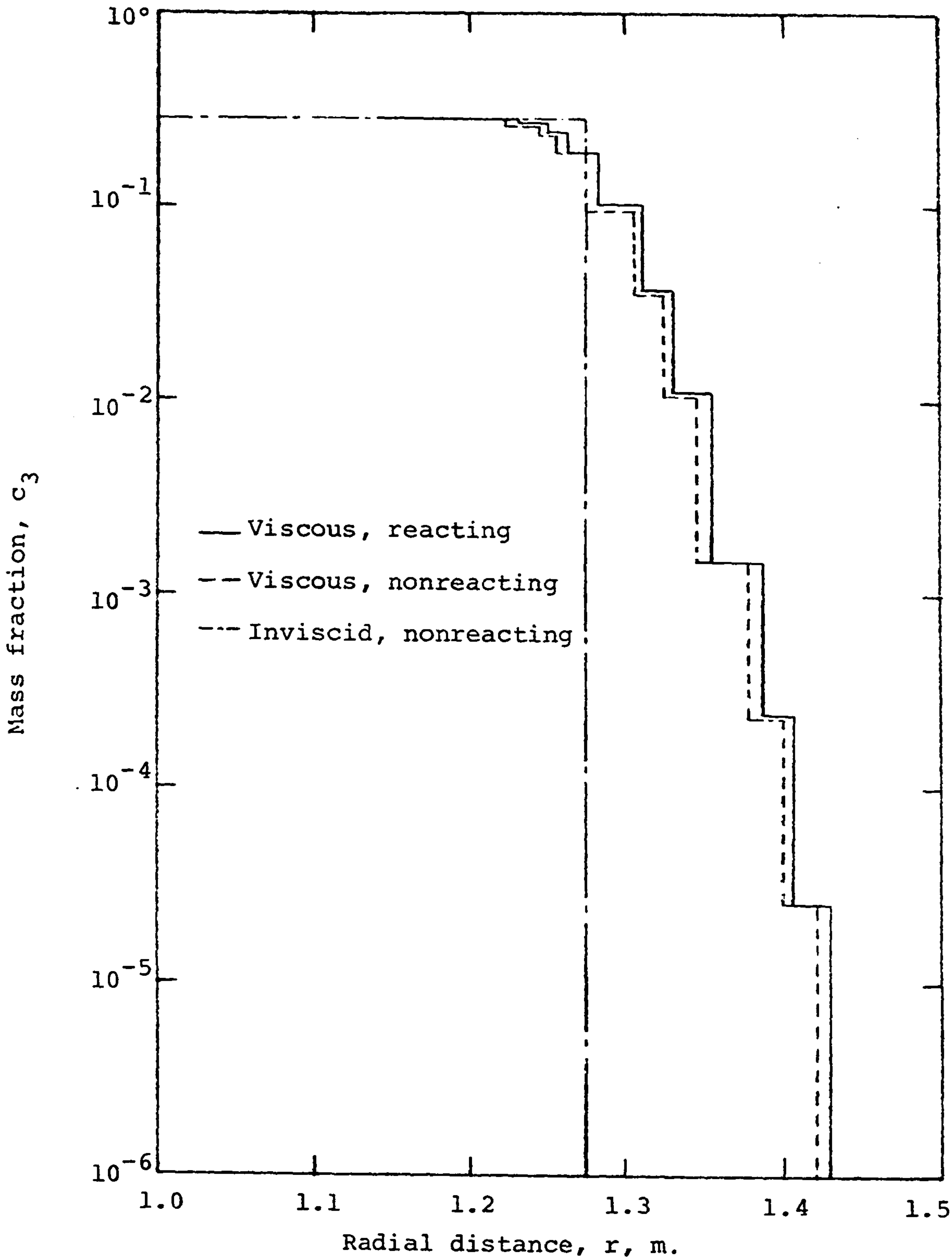

Figure 16. Mass fraction of H_2O along Surface B

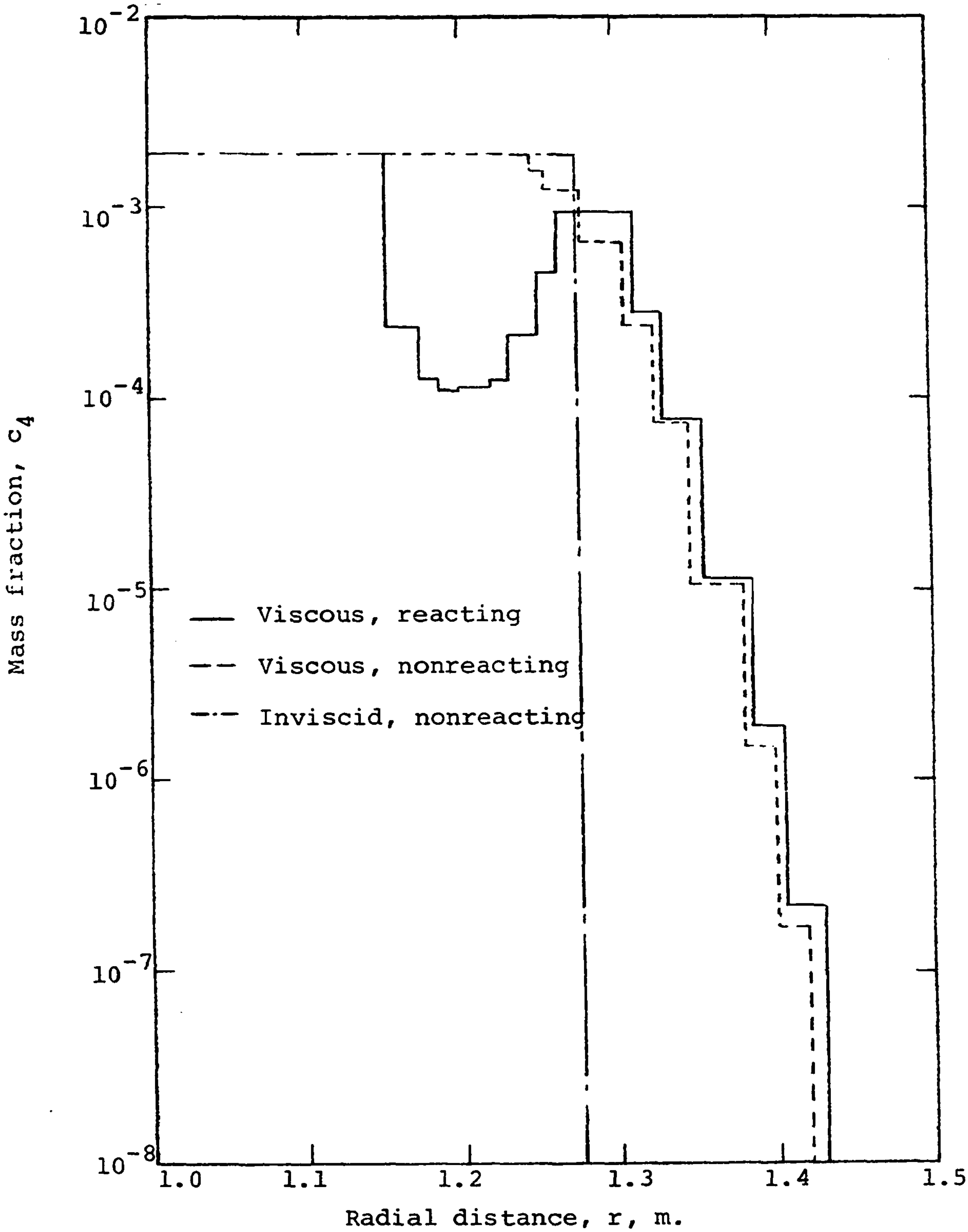

Figure 17. Mass fraction of OH along Surface B

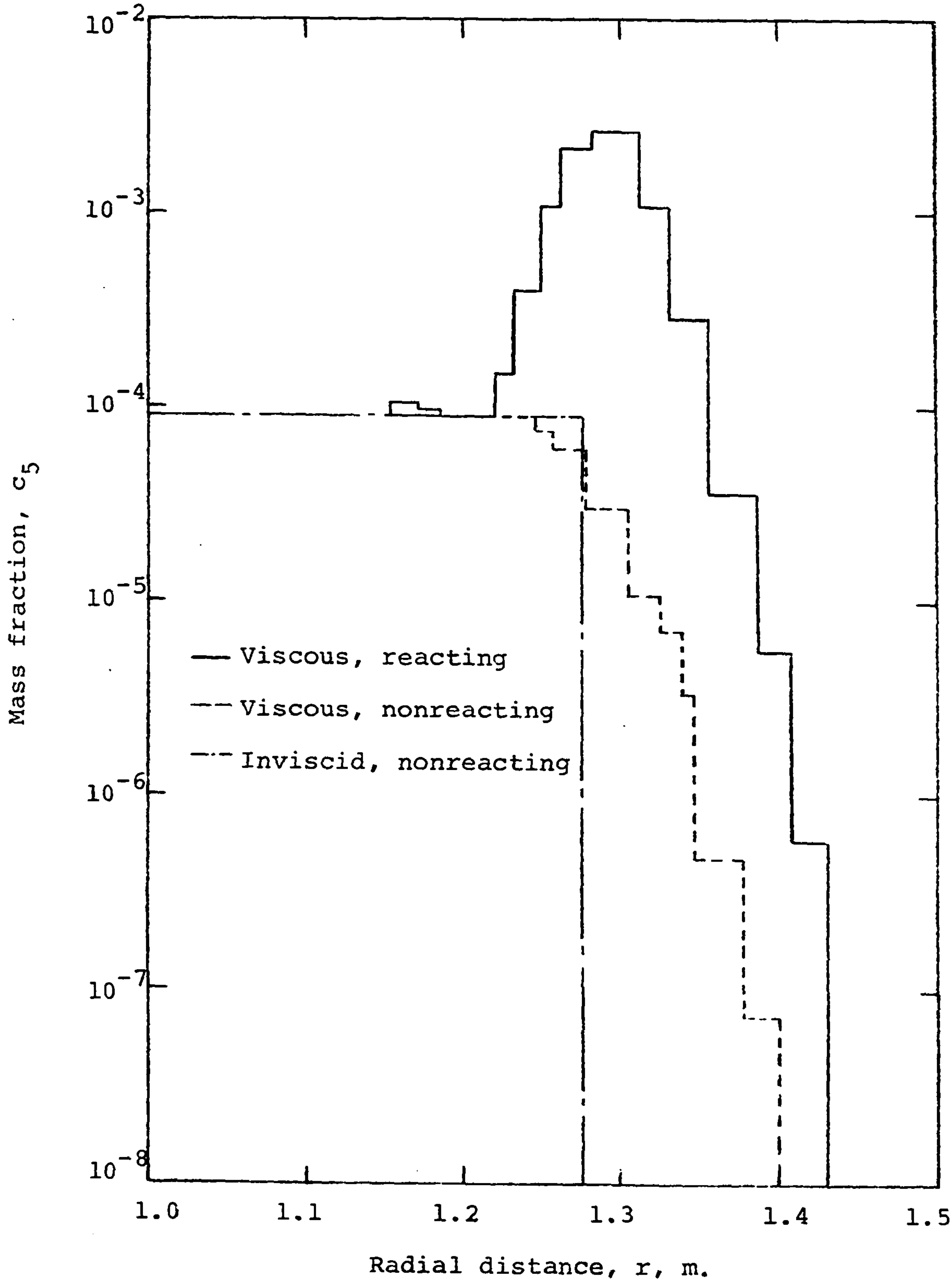

Figure 18. Mass fraction of O along surface B

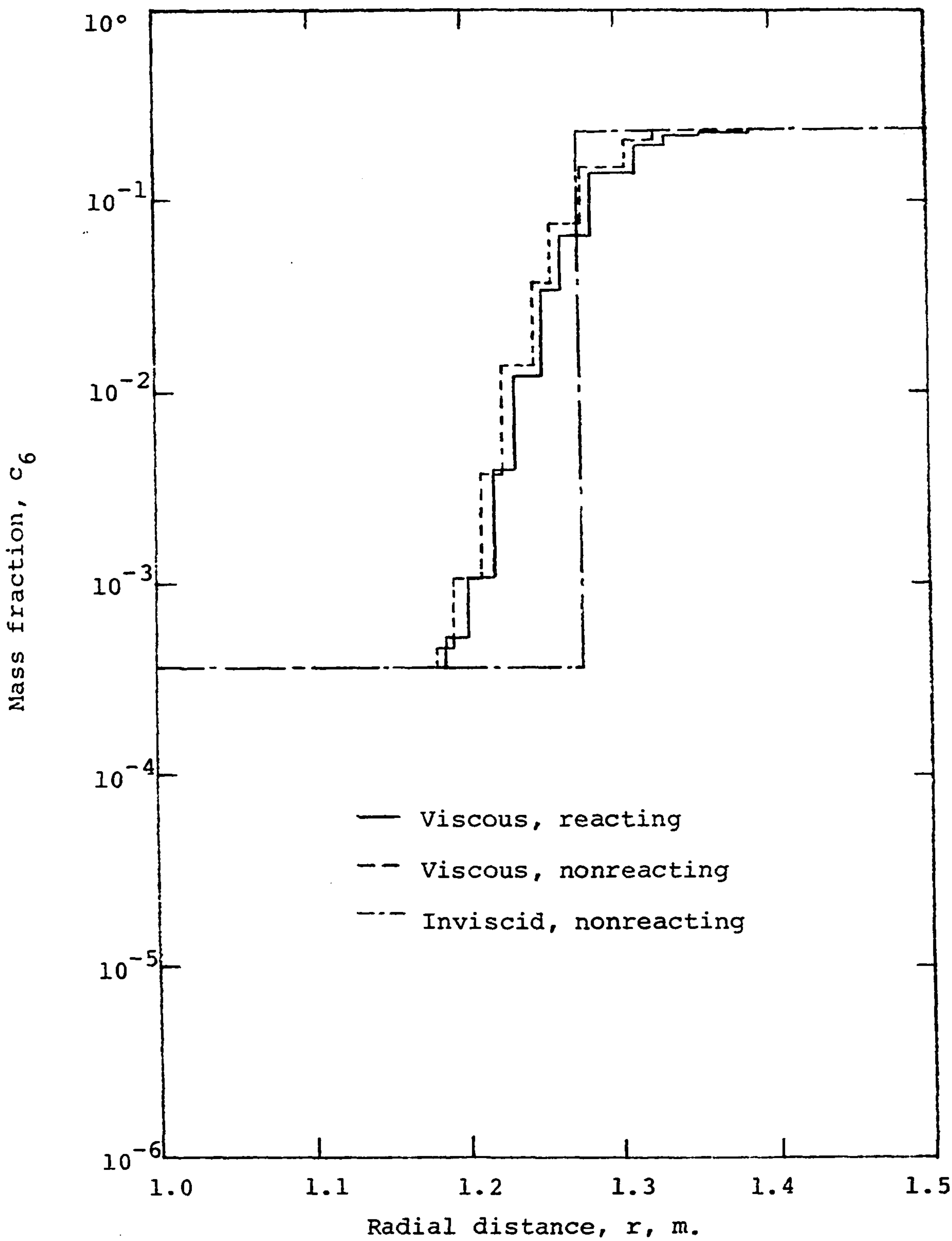

Figure 19. Mass fraction of O_2 along surface B

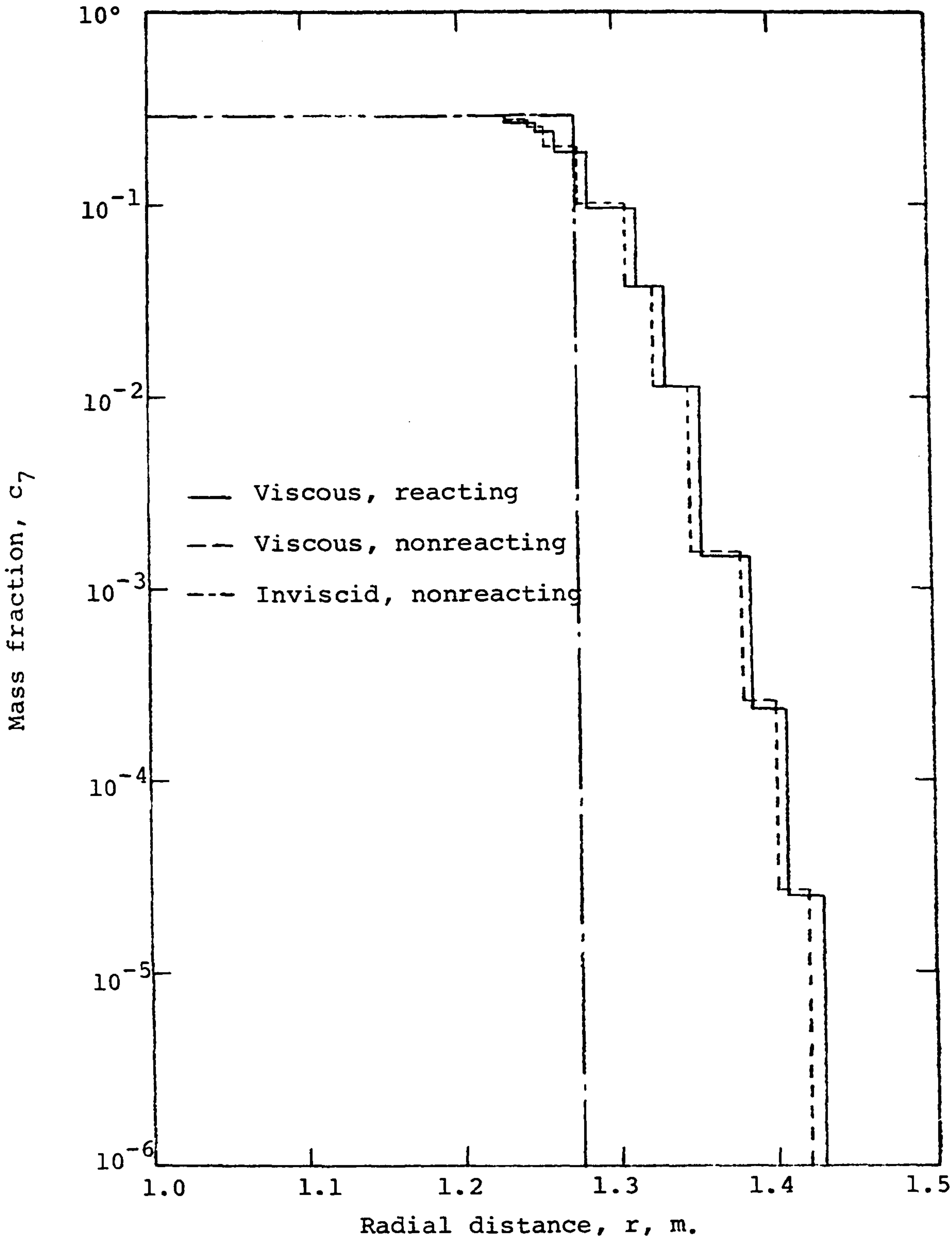

Figure 20. Mass fraction of CO along surface B

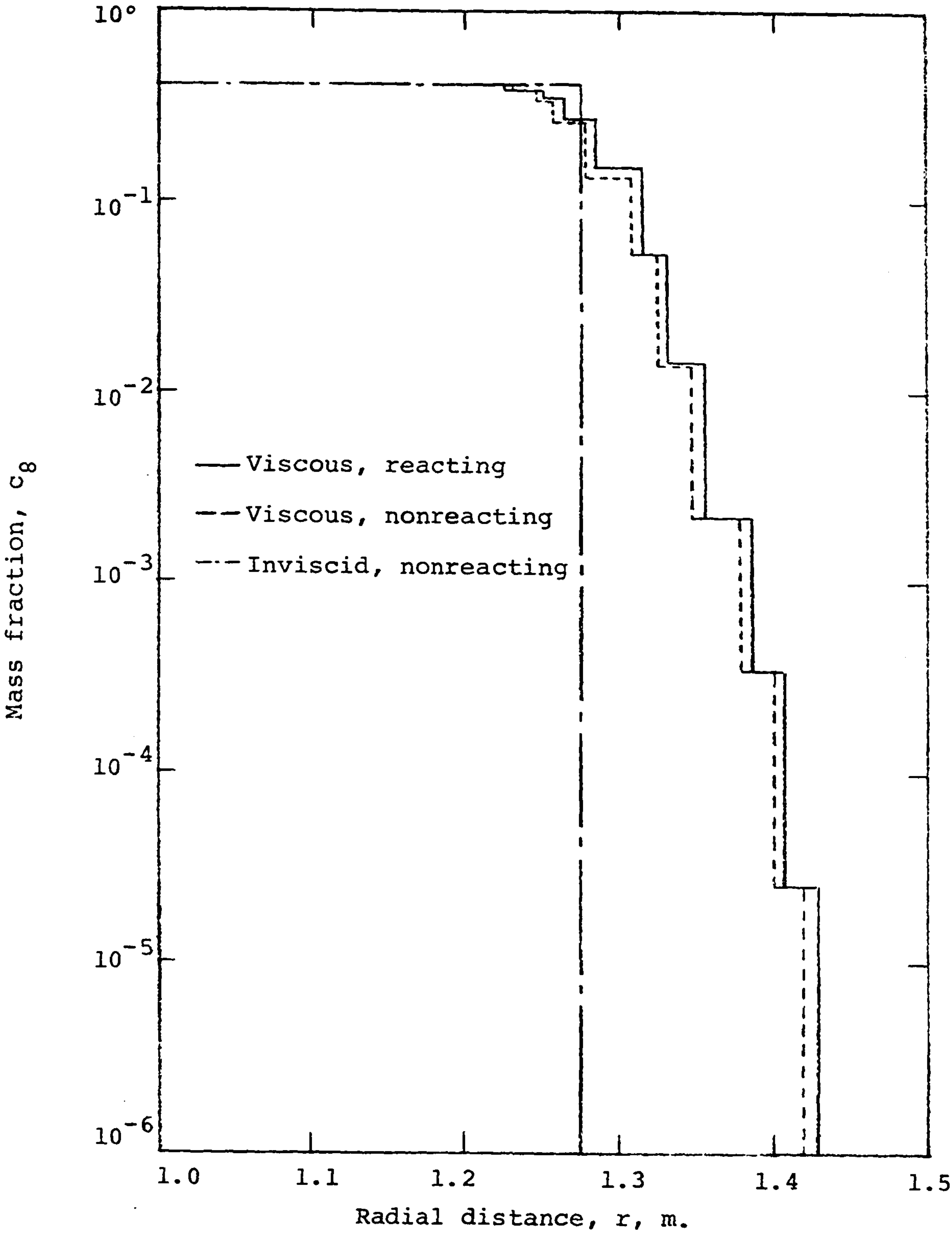

Figure 21. Mass fraction of CO_2 along surface B

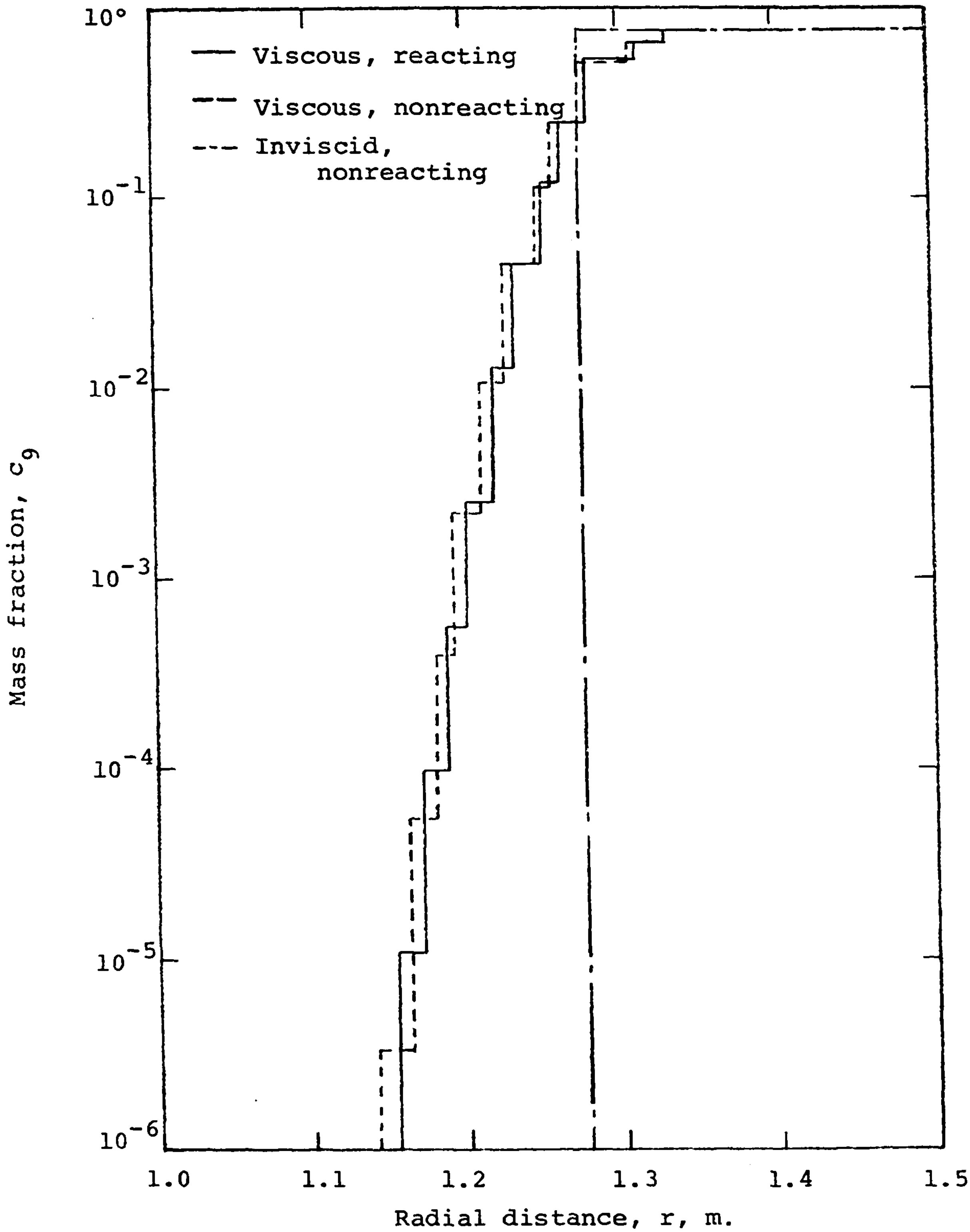

Figure 22. Mass fraction of N_2 along surface B

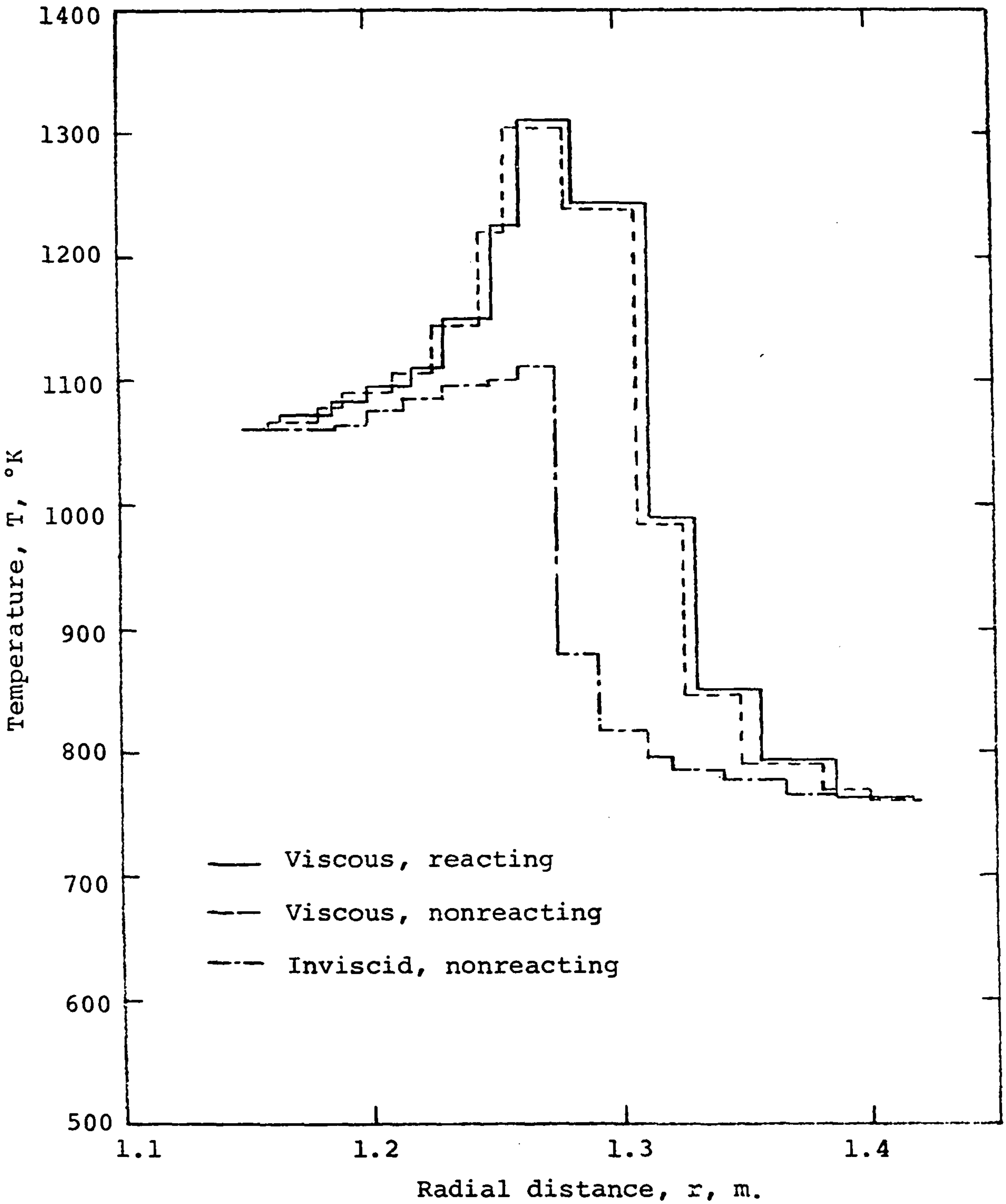

Figure 23. Temperature along surface B

increase in the Thor plume temperature above that computed by assuming inviscid, nonreacting flow is primarily due to the viscous mixing rather than the chemical reactions at the present altitude.

The computer execution times required to calculate the combined shock layers to the Mach disc using the IBM 360-65 computer for the inviscid, nonreacting; viscous, nonreacting; and viscous, reacting test cases were 1 min. 26 sec., 4 min. 12 sec., and 7 min. 35 sec., respectively.

Unfortunately, there is no experimental data to the knowledge of the author with which the Thor intermediate altitude rocket exhaust plumes calculated in the present study can be compared. As a matter of fact, there is very little experimental data available which is concerned directly with any intermediate altitude rocket exhaust plumes. This is due primarily to the difficulties involved in setting up the required test apparatus which would include a very large partial vacuum tank, a rocket nozzle through which reacting gases flow, and an external airstream. However, the analysis which has been applied in the present study is believed to give reasonable results with a minimum of computer time.

RECOMMENDATIONS FOR FURTHER STUDY

The extensions which can be made to the present study are numerous. First of all, since only one set of nozzle exit conditions was used to calculate the exhaust plume at one particular altitude in the present study, a useful extension would be to calculate exhaust plumes at various altitudes for different missiles. Included in this extension could be a study of the effects of using different eddy viscosity models. Hopefully, some of the resulting plume calculations could be compared with experimental data.

In the present analysis, various simplifying assumptions have been made. For example, the nozzle exit conditions were assumed one-dimensional and the nozzle boundary layer was ignored entirely. In addition, the external air flow was assumed uniform so that the irregularities caused by the missile's boundary layer, separated regions, and shocks were totally ignored. Any or all of these simplifying assumptions could be removed in future studies.

In addition, the plume flow downstream of the Mach disc should be calculated. In the present study, the plume flow was calculated up to but not beyond the first Mach disc. The difficulties encountered in attempting to calculate the plume flow further downstream are directly related to the subsonic region behind the Mach disc and the reflected shock from the triple point which passes through the turbulent mixing layer at intermediate altitudes because the external airstream is supersonic.

And finally, a useful extension of the present study would be to incorporate the necessary logic into the Multitube program to enable it to calculate chemical equilibrium flows. With this addition, the Multitube program could be used to calculate the supersonic flow in a plume at any altitude.

LITERATURE CITED

1. Sutton, G. P. Rocket propulsion elements. 2nd ed. New York, New York, John Wiley and Sons, Inc. 1958.

2. Tannehill, J. C. Afterburning of rocket exhausts. Aerospace Corporation Technical Memorandum 69(4129)-38. 1968.

3. Thomson, J. A. L. High altitude rocket plume structure. General Dynamics Convair Report GD/C-DBE65-023. 1965.

4. Adamson, T. C., Jr. The structure of the rocket exhaust plume without reaction at various altitudes. Ballistic Missile Radiation Analysis Center Report 4613-45-T. 1963.

5. Prozan, R. J. and Audeh, B. J. Prediction of electron density in rocket exhaust plumes. Lockheed Missiles and Space Company Report LMSC/HREC A784870. 1967.

6. Vick, A. R., Cubbage, J. M., and Andrews, E. H., Jr. Rocket exhaust plume problems and some recent related research. Unpublished mimeographed paper presented at a Specialists' Meeting on the Fluid Dynamic Aspects of Space Flight, Marseille, France, April 1964. Ames, Iowa, Department of Aerospace Engineering, Iowa State University of Science and Technology. ca. 1969.

7. Farmer, R. C., Prozan, R. J., McGimsey, L. R., and Ratliff, A. W. Verification of a mathematical model which represents large, liquid rocket-engine exhaust plumes. American Institute of Aeronautics and Astronautics Paper 66-650. 1966.

8. Adamson, T. C., Jr. and Nicholls, J. A. On the structure of jets from highly underexpanded nozzles into still air. Journal of the Aerospace Sciences 26:16-24. 1959.

9. Eastman, D. W. and Radtke, L. P. Location of the normal shock wave in the exhaust plume of a jet. American Institute of Aeronautics and Astronautics Journal 4:918-919. 1963.

10. Bowyer, J., D'Attorre, L., and Yoshihara, H. Transonic aspects of hypervelocity rocket plumes. In Olfe, D. B., and Zakkay, V., ed. Supersonic flow, chemical processes and radiation transfer. Pages 201-209. New York, New York, Pergamon Press. 1964.

11. D'Attorre, L. Experimental and theoretical studies of underexpanded jets near the Mach disc. General Dynamics Convair Report GDA-DBE 64-008. 1964.

12. Love, E. S., Grigsby, C. E., Lee, L. P., and Woodling, M. T. Experimental and theoretical studies of axisymmetric free jets. United States National Aeronautics and Space Administration Technical Report R-6. 1959.

13. Andrews, E. H., Jr., Vick, A. R., and Craidon, C. B. Theoretical boundaries and internal characteristics of exhaust plumes from three different supersonic nozzles. United States National Aeronautics and Space Administration Technical Note D-2650. 1965.

14. Vick, A. R., Andrews, E. H., Jr., Dennard, J. S., and Craidon, C. B. Comparisons of experimental free-jet boundaries with theoretical results obtained with the method of characteristics. United States National Aeronautics and Space Administration Technical Note D-2327. 1964.

15. Moe, M. M. and Troesch, B. A. Jet flows with shocks. American Rocket Society Journal 30:487-489. 1960.

16. Eastman, D. W. and Radtke, L. P. Two-dimensional or axially symmetric real gas flows by the method of characteristics. Boeing Company Document D2-10599. 1962.

17. Prozan, R. J. Development of a method of characteristics solution for supersonic flow of an ideal, frozen, or equilibrium reacting gas mixture. Lockheed Missiles and Space Company Report LMSC/HREC A782535. 1966.

18. Butler, H. W. Users manual for description of a digital computer program for nozzle and plume analysis by the method of characteristics. Lockheed Missiles and Space Company Report LMSC/HREC A783573. 1966.

19. Ratliff, A. W. Comparisons of experimental supersonic flow fields with results obtained by using a method of characteristics solution. Lockheed Missiles and Space Company Report LMSC/HREC A782592. 1966.

20. Whitehurst, C. A. and Mourer, J. Jet-shock interactions. The Journal of the Astronautical Sciences 15:71-79. 1968.

21. Boynton, F. P. and Thomson, A. Numerical computation of steady, supersonic two-dimensional gas flow in natural coordinates. Journal of Computational Physics 3:379-398. 1969.

22. Albini, F. A. Approximate computation of underexpanded jet structure. American Institute of Aeronautics and Astronautics Journal 3:1535-1537. 1965.

23. Hubbard, E. W. Approximate calculation of highly underexpanded jets. American Institute of Aeronautics and Astronautics Journal 4:1877-1879. 1966.

24. Luce, R. W. and Jarvinen, P. O. An approximate method for predicting plume sizes for nozzle flow into still air. American Institute of Aeronautics and Astronautics Journal 6:182-183. 1968.

25. Boynton, F. P. Highly underexpanded jet structure: exact and approximate calculations. American Institute of Aeronautics and Astronautics Journal 5:1703-1704. 1967.

26. Libby, P. A. Theoretical analysis of turbulent mixing of reactive gases with application to supersonic combustion of hydrogen. American Rocket Society Journal 32: 388-396. 1962.

27. Feigenbutz, L. V. Turbulent mixing and combustion of a rocket exhaust jet with the surrounding atmosphere. General Dynamics Convair Report AE63-0024. 1962.

28. Rozsa, R. B. Preliminary review of turbulent jet-mixing and afterburning in rocket exhaust plumes. Chrysler Corporation Space Division Technical Bulletin AE-64-79. 1964.

29. Rozsa, R. B. Constant pressure analysis of turbulent jet mixing with afterburning in rocket exhaust plumes. Chrysler Corporation Space Division Technical Bulletin AE-65-271. 1965.

30. Rozsa, R. B. Preliminary results of the study of turbulent jet mixing under chemical equilibrium conditions. Chrysler Corporation Space Division Technical Bulletin AE-65-309. 1965.

31. Audeh, B. J. Turbulent mixing of three concentric reacting jets. Heat Technology Laboratory Technical Report 29. 1965.

32. Audeh, B. J. Turbulent mixing of three concentric reacting jets. Heat Technology Laboratory Technical Report 41. 1967.

33. Peters, C. E., Peters, T., and Billings, R. B. Mixing and burning of bounded coaxial streams. Arnold Engineering Development Center Technical Report 65-4. 1965.

34. Vasiliu, J. Turbulent mixing of a rocket exhaust jet with a supersonic stream including chemical reactions. Journal of the Aerospace Sciences 29:19-28. 1962.

35. Edelman, R. and Fortune, O. Mixing and combustion in the exhaust plumes of rocket engines burning RPI and liquid oxygen. General Applied Science Laboratories, Inc. Technical Report 631. 1966.

36. Edelman, R. and Fortune, O. The effect of mixing, radiation, and finite-rate combustion upon the flow field and surroundings of the exhaust and liquid oxygen. General Applied Science Laboratories, Inc. Technical Report 681. 1967.

37. Boynton, F. P. The multitube supersonic flow computer code. General Dynamics Convair Report GDC-DBE67-003. 1967.

38. Ferri, A., Moretti, G., and Slutsky, S. Mixing processes in supersonic combustion. Journal Soc. Indust. Appl. Math. 13:229-258. 1965.

39. Edelman, R. and Weilerstein, G. Mixing and combustion in supersonic flow with lateral pressure gradient effects. General Applied Science Laboratories, Inc. Technical Report 636. 1968.

40. Edelman, R. and Weilerstein, G. A solution of the inviscid-viscid equations with applications to bounded and unbounded multicomponent reacting flows. American Institute of Aeronautics and Astronautics Paper 69-83. 1969.

41. Moretti, G. A new technique for the numerical analysis of nonequilibrium flows. American Institute of Aeronautics and Astronautics Journal 3:223-229. 1965.

42. Dorrance, W. H. Viscous hypersonic flow. New York, New York, McGraw-Hill Book Company, Inc. 1962.

43. Vincenti, W. G. and Kruger, C. H., Jr. Introduction to Physical Gas Dynamics. New York, New York, John Wiley and Sons, Inc. 1965.

44. Ferri, A., Libby, P. A., and Zakkay, V. Theoretical and experimental investigation of supersonic combustion. Aeronautical Research Laboratories Report ARL 62-467. 1962.

45. Kleinstein, G. On the mixing of laminar and turbulent axially symmetric compressible flows. Aeronautical Research Laboratories Report ARL 63-108. 1963.

46. Eckert, E. R. G. and Drake, R. M. Heat and mass transfer. 2nd ed. New York, New York, McGraw-Hill Book Company, Inc. 1959.

47. Richtmeyer, R. D. Difference method for initial-value problems. New York, New York, John Wiley and Sons, Inc. 1959.

48. Schlichting, H. Boundary layer theory. 4th ed. New York, New York, McGraw-Hill Book Company, Inc. 1960.

49. Eggers, J. M. Velocity profiles and eddy viscosity distributions downstream of a Mach 2.22 nozzle exhausting to quiescent air. United States National Aeronautics and Space Administration Technical Note D-3601. 1966.

50. Emanuel, G. and Vincenti, W. G. Method for calculation of the one-dimensional nonequilibrium flow of a general gas mixture through a hypersonic nozzle. Arnold Engineering Development Center Technical Documentary Report 62-131. 1962.

51. Emanuel, G. Problems underlying the numerical integration of the chemical and vibrational rate equations in a near-equilibrium flow. Arnold Engineering Development Center Technical Documentary Report 63-82. 1963.

52. Pergament, H. S. A theoretical analysis of non-equilibrium hydrogen-air reactions in flow systems. American Institute of Aeronautics and Astronautics Paper 63-113. 1963.

53. Eschenroeder, A. Q., Boyer, D. W. and Hall, J. G. Non-equilibrium expansions of air with coupled chemical reactions. Physics of Fluids 5:615-624. 1962.

54. Tyson, T. J. An implicit integration method for chemical kinetics. TRW Space Technology Laboratories Report 9840-6002-RU000. 1964.

55. DeGroat, J. J. and Abbett, M. J. A computation of one-dimensional combustion of methane. American Institute of Aeronautics and Astronautics Journal 3:381-383. 1965.

56. Magnus, D. E. and Schechter, H. S. Analysis of error growth and stability for the numerical integration of the equations of chemical kinetics. General Applied Science Laboratories, Inc. Technical Report 607. 1966.

57. McCalla, T. R. Introduction to numerical methods and fortran programming. New York, New York, John Wiley and Sons, Inc. 1967.

58. Ames Aeronautical Laboratory. Ames Research Staff. Equations, tables, and charts for compressible flow. United States National Advisory Committee for Aeronautics Report 1135. 1953.

ACKNOWLEDGEMENTS

The author wishes to express his sincere graditude to Dr. E. W. Anderson for his guidance and assistance during this study.

The author would also like to thank Mr. F. P. Boynton (General Dynamics Convair Division) who so graciously furnished the Multitube computer program and made constructive comments concerning the present research.

The author would also like to thank Mrs. Beverly Audeh (Lockheed Missiles and Space Company) and Dr. Lawrence G. Burch (The Boeing Company) who were instrumental in obtaining for the author the IBM 360-65 version of the Lockheed method of characteristics computer program.

APPENDIX A

Reaction Rates

The forward and backward reaction rates for the chemical reactions appearing in Equations 88 are given below. These rates are listed in the same order that the chemical reactions appeared in Equations 88.

$$k_{f_1} = 2.4 \times 10^{14}\, e^{(-8429.8/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{f_2} = 3.3 \times 10^{12}\, e^{(-3593.4/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{f_3} = 6.3 \times 10^{13}\, e^{(-2969.3/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{f_4} = 7.6 \times 10^{12}\, e^{(-503.3/T)} \quad [\text{cm.}^3/\text{mole-sec.}] \quad (101)$$

$$k_{f_5} = 2.4 \times 10^{19}\, T^{-0.86}\, e^{(-51957.7/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{f_6} = 1.2 \times 10^{23}\, T^{-1.34}\, e^{(-59399.6/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{f_7} = 7.5 \times 10^{14}\, T^{0.06}\, e^{(-50976.4/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{f_8} = 2.5 \times 10^{16}\, T^{-0.5}\, e^{(-59335.7/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{b_1} = 3.2 \times 10^{11}\, T^{0.47}\, e^{(-50.327/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{b_2} = 1.4 \times 10^{12}\, e^{(-2611.98/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{b_3} = 2.4 \times 10^{14}\, e^{(-10412.7/T)} \quad [\text{cm.}^3/\text{mole-sec.}]$$

$$k_{b_4} = 6.9 \times 10^{13}\, e^{(-8928.03/T)} \quad [\text{cm.}^3/\text{mole-sec.}] \quad (102)$$

$$k_{b_5} = 2.0 \times 10^{18}\, T^{-1.0} \qquad [\text{cm.}^6/\text{mole}^2\text{-sec.}] \qquad (102)$$

$$k_{b_6} = 2.3 \times 10^{21}\, T^{-1.5} \qquad [\text{cm.}^6/\text{mole}^2\text{-sec.}]$$

$$k_{b_7} = 3.0 \times 10^{14} \qquad [\text{cm.}^6/\text{mole}^2\text{-sec.}]$$

$$k_{b_8} = 2.2 \times 10^{13} \qquad [\text{cm.}^6/\text{mole}^2\text{-sec.}]$$

The chemical reactions in Equations 100 are identical to the reactions in Equations 88 except that one additional reaction is included. The additional reaction is

$$CO + OH \rightleftharpoons CO_2 + H \qquad (103)$$

The other reactions are the same if some of the forward and backward directions are reversed. Consequently, the reaction rates given by Equations 101 and 102 have been used for the corresponding reactions in Equations 100. These rates are identical to those given in Reference (35). The reaction rates for the chemical reaction given by Equation 103 are

$$k_{f_1} = 3.2 \times 10^{12}\, e^{(-3170.6/T)} \qquad [\text{cm.}^3/\text{mole-sec.}] \qquad (104)$$

$$k_{b_1} = 2.7 \times 10^{17}\, T^{-0.79}\, e^{(-15.45/T)} \qquad [\text{cm.}^3/\text{mole-sec.}]$$

where the subscript 1 is used because this is the first reaction in Equations 100.

Elements in Matrices for Constant Pressure Streamtube Calculation

The elements in the coefficient matrix [X] and the constant matrix [Z] required for subroutine CHEM (Technique 2) are given in the present section for the chemical nonequilibrium, constant pressure, streamtube test case. In the following equations for the elements of the matrices, the molar concentration terms are always evaluated at the old surface. That is, Y_i is used to represent $Y_{i,k,\ell}$. In addition, the reaction rates are to be calculated using an average value of temperature. The elements in the coefficient matrix [X] are:

$$x_{11} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{f_1} Y_5 + k_{b_2} Y_4 + k_{b_3} Y_3 + 4k_{b_5} Y_1 Y_c + k_{b_6} Y_4 Y_c + k_{b_7} Y_2 Y_c$$

$$+ (k_{f_2} Y_2 + k_{f_3} Y_4 + 2k_{f_5} Y_c)/2] + \bar{\rho}_k/\rho_{k,\ell+1}$$

$$x_{12} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_7} Y_1 Y_c - k_{b_1} Y_4 - k_{f_2} Y_6 - k_{f_1} Y_1/2]$$

$$x_{13} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_3} Y_1 - k_{b_6} Y_c - k_{f_1} Y_1/2 + k_{f_2} Y_2 + k_{f_3} Y_4 + 2k_{f_5} Y_c]$$

$$x_{14} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_2} Y_1 - k_{b_1} Y_2 - k_{f_3} Y_6 + k_{b_6} Y_1 Y_c - k_{f_7} Y_c - k_{f_1} Y_1/2$$

$$+ (k_{f_2} Y_2 + k_{f_3} Y_4 + 2k_{f_5} Y_c)/2]$$

$$x_{21} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_7} Y_2 Y_c - k_{f_1} Y_5 - k_{b_2} Y_4 - k_{f_2} Y_2/2]$$

$$x_{22} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_1}Y_4 + k_{f_2}Y_6 + k_{b_4}Y_3 + k_{b_7}Y_1Y_8 + 4k_{b_8}Y_2Y_c$$

$$+ (k_{f_1}Y_1 + 2k_{f_8}Y_c)/2 + \bar{\rho}_k/\rho_{k,\ell+1}$$

$$x_{23} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_4}Y_2 + (k_{f_1}Y_1 + 2k_{f_8}Y_c)/2 - k_{f_2}Y_2]$$

$$x_{24} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_1}Y_2 - k_{b_2}Y_1 - 2k_{f_4}Y_4 - k_{f_7}Y_c + (k_{f_1}Y_1 + 2k_{f_8}Y_c)/2$$

$$- k_{f_2}Y_2/2] \tag{105}$$

$$x_{31} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_3}Y_3 - k_{b_6}Y_4Y_c + k_{f_3}Y_4/2]$$

$$x_{32} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_4}Y_3]$$

$$x_{33} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_3}Y_1 + k_{b_4}Y_2 + k_{f_6}Y_c + k_{f_3}Y_4] + \bar{\rho}_k/\rho_{k,\ell+1}$$

$$x_{34} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{f_3}Y_6 - 2k_{f_4}Y_4 - k_{b_6}Y_1Y_c + k_{f_3}Y_4/2]$$

$$x_{41} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[-k_{f_1}Y_5 + k_{b_2}Y_4 - k_{b_3}Y_3 + k_{b_6}Y_4Y_c - k_{b_7}Y_2Y_c$$

$$+ (k_{f_2}Y_2 - k_{f_3}Y_4)/2]$$

$$x_{42} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[k_{b_1}Y_4 - k_{f_2}Y_6 - 2k_{b_4}Y_3 - k_{b_7}Y_1Y_c + k_{f_1}Y_1/2]$$

$$x_{43} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[-k_{b_3}\,Y_1 - 2k_{b_4}Y_2 - k_{f_6}Y_c + k_{f_1}Y_1/2 + k_{f_2}Y_2 - k_{f_3}Y_4]$$

$$x_{44} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[k_{b_1}Y_2 + k_{b_2}Y_1 + k_{f_3}Y_6 + 4k_{f_4}Y_4 + k_{b_6}Y_1Y_c + k_{f_7}Y_c$$

$$+ k_{f_1}Y_1/2 + (k_{f_2}Y_2 - k_{f_3}Y_4)/2] + \bar{\rho}_k/\rho_{k,\ell+1}$$

The elements in the constant matrix [Z] are:

$$z_1 = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[2k_{f_5}Y_6Y_c + k_{f_6}Y_3Y_c + k_{f_7}Y_4Y_c$$

$$- (k_{f_1}Y_1)\left(\frac{\rho_{k,\ell+1}}{2\rho_{k,\ell}}\right)(2Y_5 + Y_2 + Y_3 + Y_4) + (k_{f_2}Y_2 - k_{f_3}Y_4$$

$$- 2k_{f_5}Y_c)\left(\frac{\rho_{k,\ell+1}}{\rho_{k,\ell}}\right)(Y_6 + Y_3 + Y_1/2 + Y_4/2)] + (\bar{\rho}_k/\rho_{k,\ell})Y_1$$

$$z_2 = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{f_7} Y_4 Y_c + 2k_{f_8} Y_5 Y_c + (k_{f_1} Y_1 + 2k_{f_8} Y_c) \left(\frac{\rho_{k,\ell+1}}{2\rho_{k,\ell}} \right) (2Y_5$$

$$+ Y_2 + Y_3 + Y_4) - (k_{f_2} Y_2) \left(\frac{\rho_{k,\ell+1}}{\rho_{k,\ell}} \right) (Y_6 + Y_3 + Y_1/2 + Y_4/2)]$$

$$+ (\bar{\rho}_k / \rho_{k,\ell}) Y_2 \tag{106}$$

$$z_3 = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{f_6} Y_3 Y_c + (k_{f_3} Y_4) \left(\frac{\rho_{k,\ell+1}}{\rho_{k,\ell}} \right) (Y_6 + Y_3 + Y_1/2 + Y_4/2)]$$

$$+ (\bar{\rho}_k / \rho_{k,\ell}) Y_3$$

$$z_4 = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{f_6} Y_3 Y_c - k_{f_7} Y_4 Y_c + (k_{f_1} Y_1) \left(\frac{\rho_{k,\ell+1}}{2\rho_{k,\ell}} \right) (2Y_5 + Y_2 + Y_3$$

$$+ Y_4) + (k_{f_2} Y_2 - k_{f_3} Y_4) \left(\frac{\rho_{k,\ell+1}}{\rho_{k,\ell}} \right) (Y_6 + Y_3 + Y_1/2 + Y_4/2)]$$

$$+ (\bar{\rho}_k / \rho_{k,\ell}) Y_4$$

The molar concentrations of the first four species at the new surface can be determined using Equation 77 with the matrix elements given above. The molar concentrations of the two remaining active species are found by using Equations 94 and 95. The molar concentration of the seventh inert species (N_2) remains constant so that

$$Y_{7,k,\ell+1} = Y_{7,k,\ell} \tag{107}$$

In addition, the molar concentration of the catalyst at the new surface is determined using

$$Y_c = Y_1 + Y_2 + Y_3 + Y_4 + Y_5 + Y_6 + Y_7 \tag{108}$$

Elements in Matrices for Rocket Exhaust Plume Calculations

The equations necessary for calculating the elements in the coefficient matrix [X] and constant matrix [Z] are given in the present section for the chemistry system used in the exhaust plume calculations. In these equations, the subscripts on the reaction rates correspond to the chemical reactions given in Equations 100. The equations for the elements in the coefficient matrix [X] are:

$$x_{11} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[k_{b_1}Y_8 + k_{b_2}Y_3 + k_{b_4}Y_4 + k_{f_5}Y_6 + k_{f_6}Y_5Y_c + 4k_{f_8}Y_1Y_c + k_{f_9}Y_4Y_c] + \bar{\rho}_k/\rho_{k,\ell+1}$$

$$x_{12} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[-k_{f_2}Y_4 - k_{f_4}Y_5 - 2k_{b_8}Y_c]$$

$$x_{13} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[k_{b_2}Y_1 - k_{b_9}Y_c]$$

$$x_{14} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[-k_{f_1}Y_7 - k_{f_2}Y_2 + k_{b_4}Y_1 - k_{b_5}Y_5 - k_{b_6}Y_c + k_{f_9}Y_1Y_c]$$

$$x_{15} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[-k_{f_4}Y_2 - k_{b_5}Y_4 + k_{f_6}Y_1Y_c]$$

$$x_{16} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[k_{f_5}Y_1]$$

$$x_{17} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[-k_{f_1}Y_4]$$

$$x_{18} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[k_{b_1}Y_1]$$

$$x_{21} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[-k_{b_2}Y_3 - k_{b_4}Y_4 - 2k_{f_8}Y_1Y_c]$$

$$x_{22} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[k_{f_2}Y_4 + k_{f_4}Y_5 + k_{b_8}Y_c] + \bar{\rho}_k/\rho_{k,\ell+1}$$

$$x_{23} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[-k_{b_2}Y_1]$$

$$x_{24} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[k_{f_2}Y_2 - k_{b_4}Y_1]$$

$$x_{25} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\,[k_{f_4}Y_2]$$

$$x_{26} = 0.0$$

$$x_{27} = 0.0$$

$$x_{28} = 0.0$$

$$x_{31} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_2} Y_3 - k_{f_9} Y_4 Y_c]$$

$$x_{32} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{f_2} Y_4]$$

$$x_{33} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_2} Y_1 + k_{b_3} Y_5 + k_{b_9} Y_c] + \bar{\rho}_k / \rho_{k,\ell+1}$$

$$x_{34} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{f_2} Y_2 - 2k_{f_3} Y_4 - k_{f_9} Y_1 Y_c]$$

$$x_{35} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_3} Y_3]$$

$$x_{36} = 0.0$$

$$x_{37} = 0.0$$

$$x_{38} = 0.0$$

$$x_{41} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{b_1} Y_8 - k_{b_2} Y_3 + k_{b_4} Y_4 - k_{f_5} Y_6 - k_{f_6} Y_5 Y_c + k_{f_9} Y_4 Y_c]$$

$$x_{42} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\left[k_{f_2}Y_4 - k_{f_4}Y_5\right]$$

$$x_{43} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\left[-k_{b_2}Y_1 - 2k_{b_3}Y_5 - k_{b_9}Y_c\right]$$

$$x_{44} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\left[k_{f_1}Y_7 + k_{f_2}Y_2 + 4k_{f_3}Y_4 + k_{b_4}Y_1 + k_{b_5}Y_5 + k_{b_6}Y_c\right.$$

$$\left. + k_{f_9}Y_1Y_c\right] + \bar{\rho}_k/\rho_{k,\ell+1}$$

$$x_{45} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\left[-2k_{b_3}Y_3 - k_{f_4}Y_2 + k_{b_5}Y_4 - k_{f_6}Y_1Y_c\right]$$

$$x_{46} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\left[-k_{f_5}Y_1\right]$$

$$x_{47} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\left[k_{f_1}Y_4\right]$$

$$x_{48} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\left[-k_{b_1}Y_1\right] \tag{109}$$

$$x_{51} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\left[-k_{b_4}Y_4 - k_{f_5}Y_6 + k_{f_6}Y_5Y_c\right]$$

$$x_{52} = \frac{\overline{\delta s}_k}{2\bar{u}_k}\left[k_{f_4}Y_5\right]$$

$$x_{53} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_3} Y_5]$$

$$x_{54} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-2k_{f_3} Y_4 - k_{b_4} Y_1 + k_{b_5} Y_5 - k_{b_6} Y_c]$$

$$x_{55} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{b_3} Y_3 + k_{f_4} Y_2 + k_{b_5} Y_4 + k_{f_6} Y_1 Y_c + 4k_{f_7} Y_5 Y_c]$$

$$+ \bar{\rho}_k/\rho_{k,\ell+1}$$

$$x_{56} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{f_5} Y_1 - 2k_{b_7} Y_c]$$

$$x_{57} = 0.0$$

$$x_{58} = 0.0$$

$$x_{61} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{f_5} Y_6]$$

$$x_{62} = 0.0$$

$$x_{63} = 0.0$$

$$x_{64} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{b_5} Y_5]$$

$$x_{65} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{b_5} Y_4 - 2k_{f_7} Y_5 Y_c]$$

$$x_{66} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{f_5} Y_1 + k_{b_6} Y_c] + \bar{\rho}_k/\rho_{k,\ell+1}$$

$$x_{67} = 0.0$$

$$x_{68} = 0.0$$

$$x_{71} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{b_1} Y_8]$$

$$x_{72} = 0.0$$

$$x_{73} = 0.0$$

$$x_{74} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{f_1} Y_7]$$

$$x_{75} = 0.0$$

$$x_{76} = 0.0$$

$$x_{77} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [k_{f_1} Y_4] + \bar{\rho}_k/\rho_{k,\ell+1}$$

$$x_{78} = \frac{\overline{\delta s}_k}{2\bar{u}_k} [-k_{b_1} Y_1]$$

$$x_{81} = \frac{\overline{\delta s_k}}{2\bar{u}_k} [k_{b_1} Y_8]$$

$$x_{82} = 0.0$$

$$x_{83} = 0.0$$

$$x_{84} = \frac{\overline{\delta s_k}}{2\bar{u}_k} [-k_{f_1} Y_7]$$

$$x_{85} = 0.0$$

$$x_{86} = 0.0$$

$$x_{87} = \frac{\overline{\delta s_k}}{2\bar{u}_k} [-k_{f_1} Y_4]$$

$$x_{88} = \frac{\overline{\delta s_k}}{2\bar{u}_k} [k_{b_1} Y_1] + \bar{\rho}_k/\rho_{k,\ell+1}$$

The elements in the constant matrix [Z] are:

$$z_1 = \frac{\overline{\delta s_k}}{2\bar{u}_k} [k_{b_6} Y_4 Y_c + 2k_{b_8} Y_2 Y_c + k_{b_9} Y_3 Y_c] + (\bar{\rho}_k/\rho_{k,\ell}) Y_1 + \frac{\bar{\rho}_k}{\dot{m}_k M_1}$$

$$z_2 = \frac{\overline{\delta s_k}}{2\bar{u}_k} [-k_{b_8} Y_2 Y_c] + (\bar{\rho}_k/\rho_{k,\ell}) Y_2 + \frac{\bar{\rho}_k}{\dot{m}_k M_2} \Delta_k \overline{[r_k^j \; J_{i,k} \; \delta s_k]} (2\pi)^j$$

$$z_3 = \frac{\overline{\delta s_k}}{2\bar{u}_k} [-k_{b_9} Y_3 Y_c] + (\bar{\rho}_k/\rho_{k,\ell}) Y_3 + \frac{\bar{\rho}_k}{\dot{m}_k M_3} \Delta_k \overline{[r_k^j \; J_{i,k} \; \delta s_k]} (2\pi)^j$$

$$z_4 = \frac{\overline{\delta s_k}}{2\bar{u}_k}\,[-k_{b_6}Y_4Y_c + k_{b_9}Y_3Y_c] + (\bar{\rho}_k/\rho_{k,\ell})Y_4$$

$$+ \frac{\bar{\rho}_k}{\dot{m}_k M_4}\,\Delta_k\,\overline{[r_k^j\,J_{i,k}\,\delta s_k]}\,(2\pi)^j \qquad (110)$$

$$z_5 = \frac{\overline{\delta s_k}}{2\bar{u}_k}\,[k_{b_6}Y_4Y_c + 2k_{b_7}Y_6Y_c] + (\bar{\rho}_k/\rho_{k,\ell})Y_5$$

$$+ \frac{\bar{\rho}_k}{\dot{m}_k M_5}\,\Delta_k\,\overline{[r_k^j\,J_{i,k}\,\delta s_k]}\,(2\pi)^j$$

$$z_6 = \frac{\overline{\delta s_k}}{2\bar{u}_k}\,[-k_{b_7}Y_6Y_c] + (\bar{\rho}_k/\rho_{k,\ell})Y_6 + \frac{\bar{\rho}_k}{\dot{m}_k M_6}\,\Delta_k\,\overline{[r_k^j\,J_{i,k}\,\delta s_k]}\,(2\pi)^j$$

$$z_7 = (\bar{\rho}_k/\rho_{k,\ell})Y_7 + \frac{\bar{\rho}_k}{\dot{m}_k M_7}\,\Delta_k\,\overline{[r_k^j\,J_{i,k}\,\delta s_k]}\,(2\pi)^j$$

$$z_8 = (\bar{\rho}_k/\rho_{k,\ell})Y_8 + \frac{\bar{\rho}_k}{\dot{m}_k M_8}\,\Delta_k\,\overline{[r_k^j\,J_{i,k}\,\delta s_k]}\,(2\pi)^j$$

The molar concentrations of the eight active species at the new surface can be determined using Equation 77, as before. The molar concentration of the inert ninth species (N_2) remains constant and the molar concentration of the catalyst at the new surface is found using the equation

$$Y_c = Y_1 + Y_2 + Y_3 + Y_4 + Y_5 + Y_6 + Y_7 + Y_8 + Y_9 \qquad (111)$$

APPENDIX B

Inputs for Modified Multitube Program

A detailed description of the computer inputs required for the modified Multitube program is given in this section. This description is identical to the one appearing in Reference (37) except for the additional inputs and changes required for the modified program. The numbers in parentheses denote the maximum value of arrays.

Card # 1 (Format 23I3): Integral parameters

KMAX(60)	Initial number of streamtubes
NN(5)	Number of terms in enthalpy polynomial (must be 2 for constant-γ flow)
KP	Output control (prints every KP steps)
IP	Dummy variable
ITYPE	Inner boundary condition flag 1 - wall or axis 2 - free boundary 3 - shock boundary
IKIND	Outer boundary condition flag 1 - wall 2 - free boundary 3 - Newtonian pressure boundary 4 - shock
MMAX(50)	Number of inner boundary points
NMAX(50)	Number of outer boundary points

NDS(10)	Number of species in flow
IPD	Pressure diffusion switch (not applicable if ITURB = 1) 0 - pressure diffusion omitted not 0 - pressure diffusion included
NITER	Number of iterations for iterative solution
IDIFF	Dummy variable
LPLANE	Number of steps to be taken
IPTUC	Selects variable to be tested when combining streamtubes 1 - pressure 2 - temperature 3 - velocity 4 - mass fraction of species # 1
IPCH	Punch control (writes tape and/or punches every IPCH steps)
ISHOCK	Shock indicator 0 - no shocks 1 - shock at outer boundary, external stream uniform 2 - shock at outer boundary, external stream non-uniform 3 - shocks at both boundaries, external streams uniform 4 - shocks at one or both boundaries; inner stream non-uniform, outer stream uniform

5 - shocks at one or both boundaries; inner stream uniform, outer stream non-uniform

IBUGSH	Debug printout indicator
	0 - no debug printout
	not 0 - prints out certain details of calculation
ITD	Thermal diffusion indicator (not applicable if ITURB = 1)
	0 - no thermal diffusion
	not 0 - thermal diffusion included
ICSH	Indicates form of external flow
	0 - tabular input (punched cards)
	not 0 - analytical coefficients
IBURN	Chemical reaction indicator
	0 - nonreacting
	1 - reacting (chemical nonequilibrium)
NCR(20)	Number of chemical reactions if IBURN = 1
NCAT(5)	Number of catalysts if IBURN = 1
ITURB	Turbulent transport indicator
	0 - inviscid or laminar transport
	1 - turbulent transport

Card #2 (Format 6E12.6): Physical constants - these quantities allow a variety of units to be used

BETAP	Converts pressure units to dynamic units (e.g., 1.013 x 10^6 dyne cm^{-2} atm^{-1})

HJ — Mechanical equivalent of heat (e.g., 4.182 x 10^7 erg cal^{-1})

RCON — Universal gas constant in caloric units (e.g., 1.9871 cal deg^{-1}gm $mole^{-1}$)

Card #3 (Format 8E9.4, 1E8.4): Non-integral parameters

ALPHAH — Fraction of maximum stable stepping distance to be actually taken

EPSLON — Amount by which tube Mach numbers must exceed one for the calculation to continue

TOL — Convergence tolerance for iterative solutions

DELTA — Metric exponent

0 - dimensional flow

1.0 - axially symmetric flow

ATOL — Allowed fractional change in streamtube area per step

FSTEP — Tube combination multiplier (essentially the number of streamtubes to be carried if tubes are similar $\dot{m}/r$; the program combines tubes whenever $\dot{m}_k/r_k$ *FSTEP < $\sum_k \dot{m}_k/r$ unless a test variable varies too rapidly)

GRAD — Tube combination test variable multiplier (the program will combine tubes when quantity A, as indicated by IPTUC, varies by less than GRAD*$|A|$ between the tubes in question. Exception: when IPTUC = 4, the program never combines tubes of different composition in an inviscid calculation; slip lines are thereby preserved if different compositions are assigned to the two sides.)

FRAC Diffusion growth multiplier (fluxes reach their full values at x = FRAC* (initial width of flow))

FRACTN FSTEP growth multiplier (FSTEP varies linearly with number of steps taken from input value of KMAX at 0 steps to input value of FSTEP at LPLANE steps. This feature is sometimes useful in performing efficient calculations of shock layers.)

Card Group #3A (Format 10E8.5): Thermal diffusion constants. This card group is only present if ITD $\neq$ 0. Two cards:

1. ALFTD(I), I = 1, NDS (max. 10) - Thermal diffusion factor of species pair I, 1 at reference temperature
2. DEXP(I), I = 1, NDS (max. 10) - Temperature exponential constant of thermal diffusion factor ALFTD(I)

Card Group #4 (Format 5E12.6): Thermal properties of species. NDS cards, one for each species.

A(I,J), J = 1, NDS (max. 5 per card) - Enthalpy polynomial constants (give enthalpy per unit <u>mass</u>). The constant term must include the heat of formation if flow is reacting.

Card Group #5 (Format A4, 8X, 5E12.4, 1E8.2): Transport properties of species, mostly. NDS cards, one for each species.

IDENT(I)	Formula of species (e.g., N2, AIR, etc.)
MUO(I)	Viscosity of species at reference temperature; if the flow is inviscid, the viscosity of the first species is set equal to zero
TO(I)	Reference temperature for the species
OMEGA(I)	Temperature exponent of viscosity
PR(I)	Reciprocal of species Prandtl number
SC(I)	Reciprocal of species Schmidt number
MW(I)	Species molecular weight

Card #5A (Format 4E12.5): Turbulent transport properties. This [illegible] is present only if ITURB = 1.

TLE	Turbulent Lewis number
TPR	Turbulent Prandtl number
EDDYK	Constant in eddy viscosity expression
DELMIX	Initial value of $\Delta Y[(\rho u)_{max} - (\rho u)_{min}]$

Card Group #5B (Format 30I2) Stoichiometric coefficients in chemical reactions. These cards are present if IBURN = 1. There are NCR cards, one for each reaction. The stoichiometric coefficients in the forward direction are listed first and then the coefficients in the backward direction are given. The coefficients for the species are placed in the same order as the species appeared in Card Group #5, and then the coefficients for the catalysts are listed in their order of appearance in

Card Group #5D.

NU1(M1,J), NU2(M1,J) (where J = 1, NDS + NCAT)

Card Group #5C (Format 6E12.5) Constants in reaction rates. These cards are present if IBURN = 1. There are NCR cards, one for each reaction. The constants in the reaction rates are first given in the forward direction and then the backward direction.

RF1(M1), RF2(M1), RF3(M1), RB1(M1), RB2(M1), RB3(M1)

Card Group #5D (Format 10I2) Constants to determine molar concentrations of catalysts. These cards are present if NCAT is not equal to zero. There are NCAT cards, one for each catalyst. The molar concentration of each catalyst is assumed to be a linear combination of the molar concentrations of the species present. The coefficients for the linear combination are listed in the same order that the species appeared in Card Group #5.

ICAT(NCAT, I) (where I = 1, NDS)

Card Group #6 (Formats 6#12.6 and 10E8.5): Data along initial surface. The streamtube index should increase in the direction of generally increasing r (e.g., an axis should always be taken as the inner boundary, etc.).

(2*KMAX+1) cards, as follows:

X(K=0)	Axial coordinate of inner boundary point
R(K=0)	Radial coordinate of inner boundary point
PHI(K=0)	Flow angle at inner boundary point

Second card, and every other card to end of group:

X(K)	Axial coordinate of streamline at outside of tube K
R(K)	Radial coordinate of streamline at outside of tube K
PHI(K)	Flow angle of streamline at outside of tube K
P(K)	Pressure in tube K
T(K)	Temperature in tube K
U(K)	Velocity in tube K

Third card, and every other card to end of group:

C(I,K), I = 1, NDS	Species mass fractions in tube K

Card Group #7 (Format 5E12.6): Inner boundary data. MMAX cards, one for each point along the boundary. If the boundary is not a wall, at least two cards must be present.

XW(M)	Axial coordinate at point M
RW(M)	Radial coordinate at point M
PHIW(M)	Boundary angle at point M (will be calculated by the program if a zero value is input)
PW(M)	Pressure acting on inner boundary at point M (applies only to a free-boundary calculation)
SW(M)	Distance along boundary from first input point to point M (will be calculated by the program if a zero value is input)

Card Group #8 (Format 5E12.6): Outer boundary data. NMAX cards, one for each point along the boundary. If the boundary is not a wall, at least two cards must be present. Definitions similar to inner boundary.

XB(N)

RB(N)

PHIB(N)

PB(N)

SB(N)

Card Group #9 (Formats 6E12.6, 10E8.5): Data pertaining to shock and external stream at outer boundary. These cards must be present if the outer boundary is a shock or a Newtonian boundary (IKIND $\geq$ 3); if IKIND = 3, the first two cards may be left blank.

First card (E12):

XSH	Axial coordinate of shock point	
RSH	Radial coordinate of shock point	
PSI	Shock angle	
PSH	Pressure behind shock	Estimates only
TSH	Temperature behind shock	Estimates only
USH	Velocity behind shock	Estimates only

Second card (E8):

CSH(I), I = 1, NDS Species mass fractions at shock

Third card (E12):

XSTREM	Dummy variables - leave blank	
RSTREM		
PHSTRM	Flow angle in outer flow	Estimates only if outer flow is not uniform
PSTREM	Pressure in outer flow	
TSTREM	Temperature in outer flow	
USTREM	Velocity in outer flow	

Fourth card (E8):

CSTREM(I), I = 1, NDS Species mass fractions in outer flow

Card Group #10 (Formats 6E12.6, 10E8.5): Data pertaining to shock and external stream at inner boundary. Cards must be present if inner boundary is a shock (ITYPE = 3). Definitions similar to outer boundary.

First card (E12):

XSHW

RSHW

PSTW

PSHW

TSHW

USHW

Second card (E8):

CSHW(I), I = 1 NDS

Third card (E12):

XOO

ROO

PHOO

POO

TOO

UOO

Fourth card (E8):

COO(I), I = 1, NDS

Card Group #11: Description of the non-uniform external flow. Two options are provided: a) The flow is described in tabular form along successive surfaces orthogonal to the streamlines. Generally these cards are obtained as punched output from a previous calculation. b) The flow is described by an analytical expression applicable to vacuum jet plumes.

Option "a", tabular input:

First card, and every other card thereafter to a maximum of 60 per surface (Format 6E12.6, 2I4):

FX(K)	Axial coordinate of streamline	
FR(K)	Radial coordinate of streamline	
FPHI(K)	Flow angle of streamline	
FP(K)	Streamtube pressure	These may be left
FT(K)	Streamtube temperature	blank for the
FU(K)	Streamtube velocity	0^{th} streamtube

NHDL(K)	"Upstream index" of streamline	must be 0 for the 0th streamline
JNDR(K)	"Downstream index" of streamline	

Second card, and every other card thereafter to a maximum of 60 per surface (Format 10E8.0)

FC(I,K),I = 1, NDS Species mass fractions in streamtube

Insert two blank cards at end of each surface, and four blank cards at end of last surface.

Option "b", analytical coefficients for vacuum plume:

First card (Format 5E12.6, 6I3):

THETOO	Limiting angle of expansion
RHOE	Density at edge of nozzle exit
POE	Total pressure at edge of nozzle exit
TOE	Total temperature
REX	Nozzle exit radius
NEND	Number of "N" terms in density expression (6 max.)
NEMD	Number of "M" terms in density expression
NENT	Number of "N" terms in angle expression
NEMT	Number of "M" terms in angle expression
NENP	Number of "N" terms in total-pressure expression
NEMP	Number of "M" terms in total-pressure expression

Second and successive cards (NEND total. Format 6E12.6):

ARHO(M,N), M = 1, NEMD Density coefficients

(NEND+)th and successive cards (NENT total. Format 6E12.6):

ATHET(M,N), M = 1, NEMT Angle coefficients

(NEND+NENT+1)th and successive cards (NENP total. Format 6E12.6):

AP(M,N), M = 1, NEMP Total-pressure coefficients

Flow Chart for Modified Multitube Program

A simplified flow chart for the modified Multitube program is shown in Figure 24. Further details of the operation of the Multitube program can be found in Reference (37).

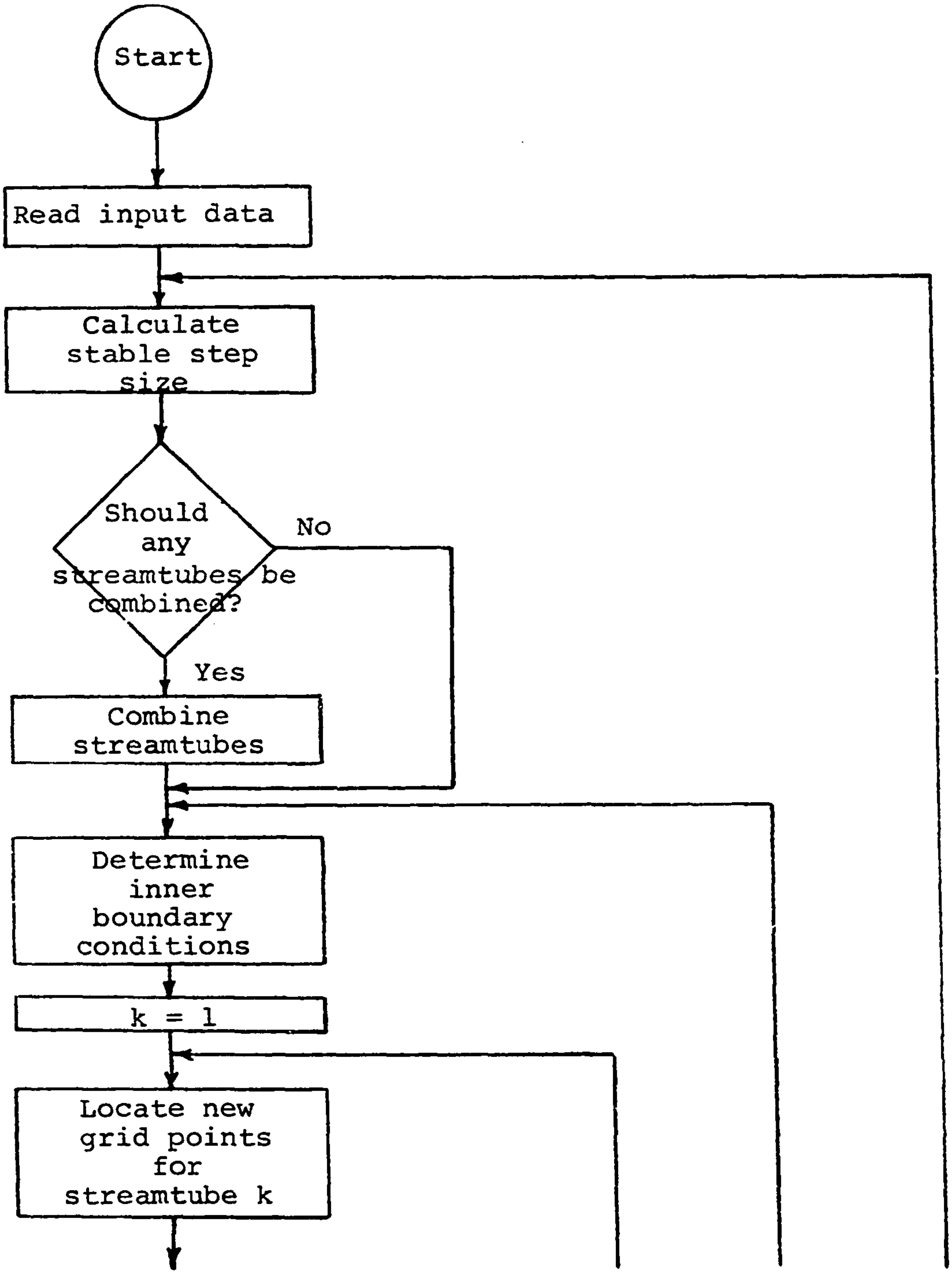

Figure 24. Modified Multitube program flow chart

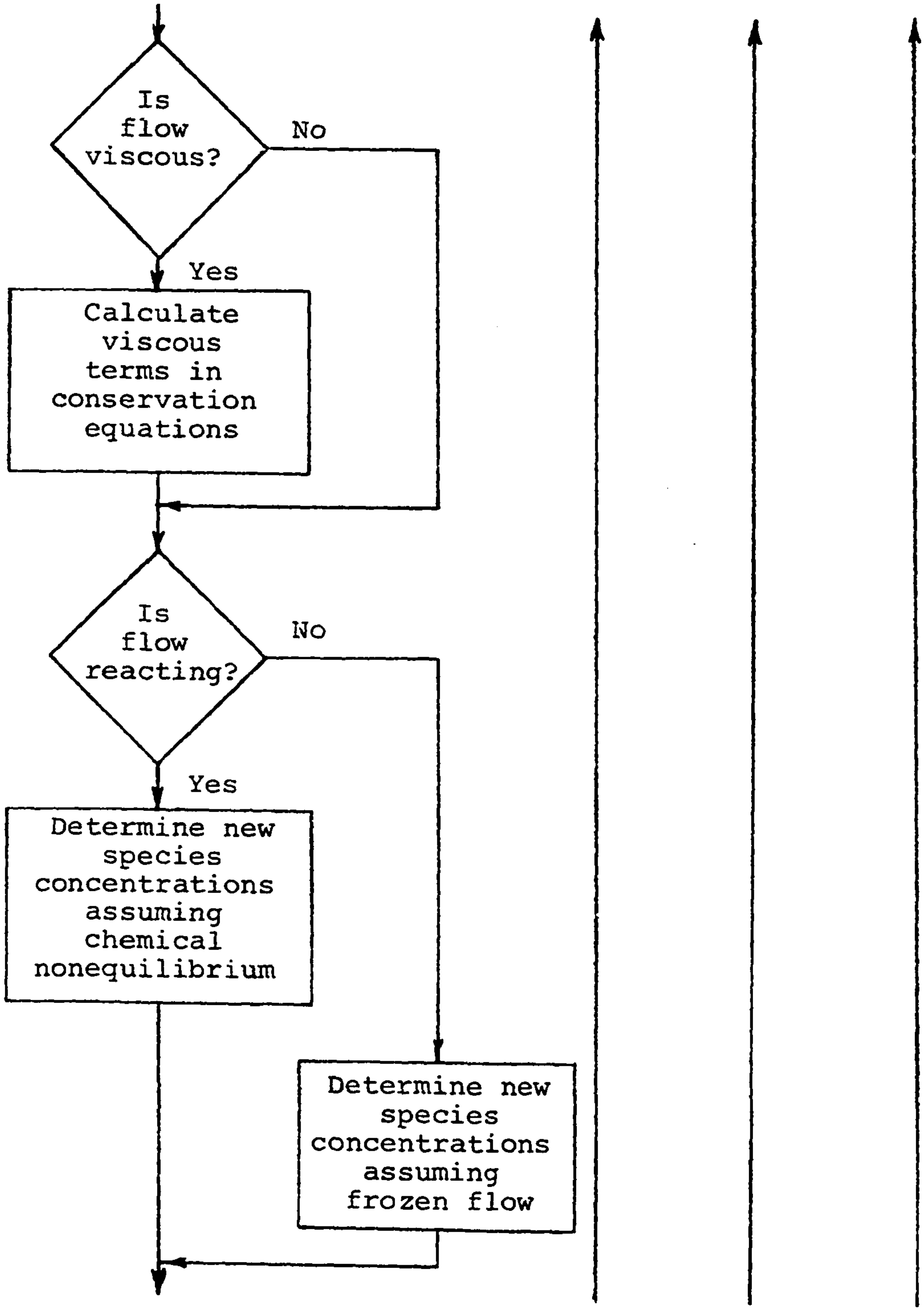

Figure 24 (Continued)

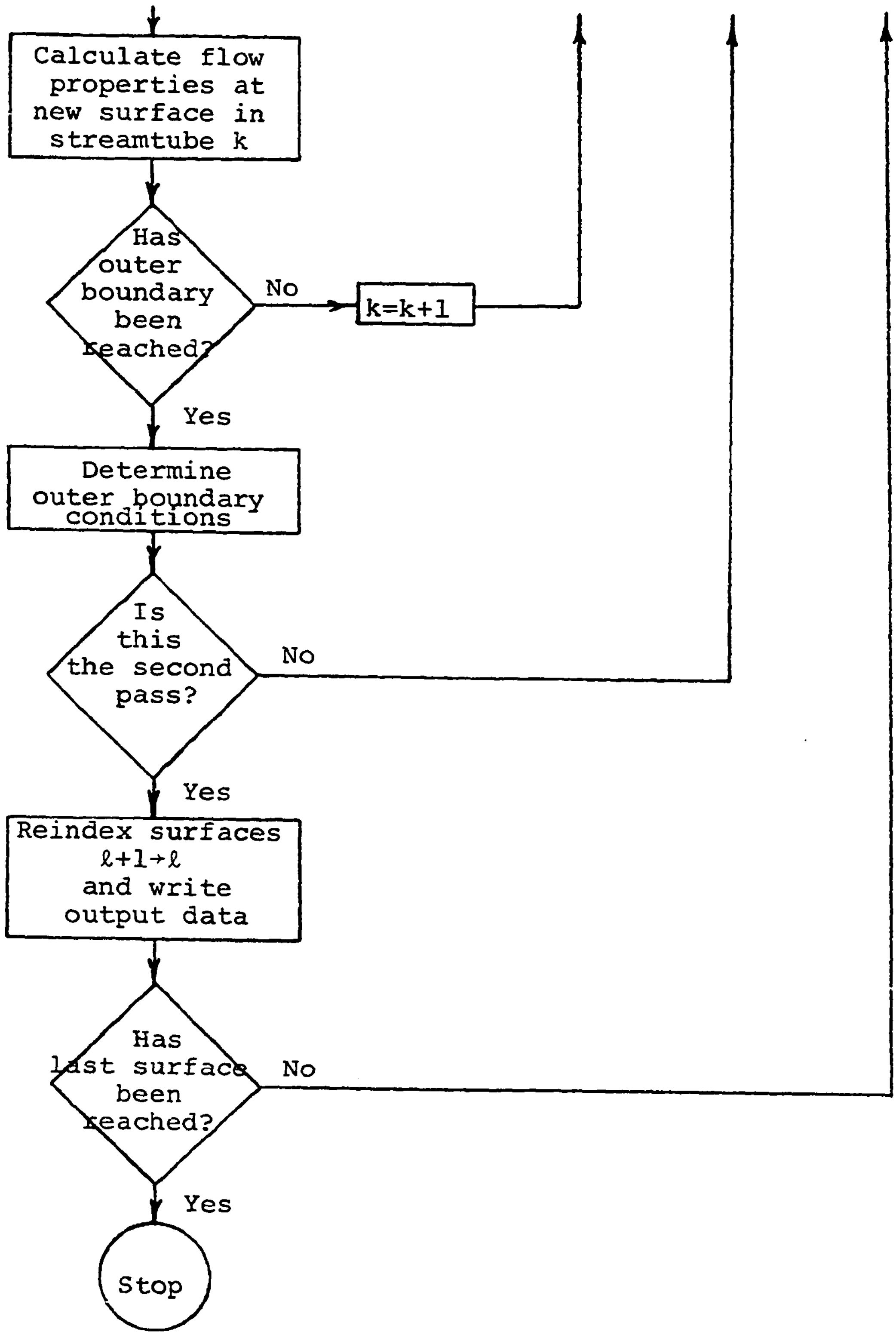

Figure 24 (Continued)

Listing of Modified Multitube Program

```
C
C     MULTITUBE SUPERSONIC FLOW COMPUTER PROGRAM INCLUDING LOGIC FOR
C     CHEMICAL NONEQUILIBRIUM AND TURBULENT TRANSPORT
C
      COMMON A(10,5)      ,AA(60)      ,ALFA(10,10),ALPHAH      ,ALPHAP
     1       ,ATOL        ,BETAP       ,BMIX       ,C11(10)     ,C(10,60)
     2       ,C12(10)     ,C2(10,60)   ,CABAR(10)  ,CBBAR(10)   ,CP
     3       ,CPS(10)     ,CPSH        ,CSH(10)    ,CSH1(10)    ,CSTREM(10)
     4       ,MXSTRM      ,D2IH(10,10) ,DEFF(10)   ,D2EFF(10)   ,DELTA
     5       ,D11         ,DELSS(60)   ,DELS       ,DELSO       ,DLS(60)
     6       ,DIH(10,10)  ,D12         ,DELY(60)   ,D13         ,DPDY(60)
     7       ,D14         ,DPHIDS(60)  ,EPCON
     8       ,EPSLON      ,EXTRA(50)   ,FSTEP      ,FMAX        ,GRAD
     9       ,H11         ,H(60)       ,HH         ,HJ          ,HPM(10)
     *       ,H2PM(10)    ,H3PM(10)    ,ICONST     ,ICOUNT      ,IDENT(10)
     1       ,IERROR      ,IEXTRA(50)  ,IFLAG      ,IKIND
     2       ,IPTUO       ,ISHOCK      ,ITYPE      ,IPD
     3       ,IDIFF       ,K           ,KAY        ,KAYS        ,KAY2
     4       ,KLO         ,KMAX
     5       ,KUP         ,KW          ,LL         ,LPLANE      ,MA
     6       ,MASH        ,MDOT        ,MMAX       ,MUO         ,MU
     7       ,MU2         ,MUS         ,MW         ,MW2         ,MWSH
     8       ,NBOUND      ,NDS         ,NITER      ,NMAX        ,NN
      COMMON NUCASE       ,OMEGA(10)   ,P11        ,P(60)       ,P12
     1       ,P2(60)                   ,PB(50)     ,PABAR       ,PBBAR
     2       ,PBS         ,P15         ,PHI(60)                 ,PHIB(50)
     3       ,PHISH1      ,PHSTRM                  ,PHIW(50)    ,P16
     4       ,PHI2(60)    ,PHIBS       ,PI         ,PR(10)      ,PSH
     5       ,PSH1        ,PSI         ,PSTREM                  ,PW(50)
     6       ,Q11         ,Q(60)       ,QXTR1      ,QXTR2
     7       ,QW(50)      ,R11         ,R(60)                   ,RB(50)
     8       ,RBS         ,R13         ,RBAR(60)   ,R14         ,R2(60)
     9       ,RCON        ,R15         ,RE(60)     ,RESH        ,R16
     *       ,RHO(60)     ,R17         ,RHO2(60)   ,RHS(10)     ,RHSEN
     1       ,RHSMOM      ,RHABAR      ,RHBBAR     ,RU          ,RSH
     2       ,RSTREM      ,RV          ,R19        ,RW(100)     ,S
     3                    ,SB(50)      ,SC(10)                  ,SW(50)
```

```
     4        ,S13          ,SX(60)       ,T11          ,T(60)        ,T12
     5        ,T2(60)       ,TABAR        ,TBBAR        ,TC(10)       ,T14
     6        ,TAW(60)      ,TXTR1        ,TXTR2        ,TOL          ,TSH
     7        ,TSH1         ,TSTREM                     ,TW(50)       ,TWS
     8        ,TS           ,U11          ,U(60)        ,U12          ,U2(60)
      COMMON UXTR1          ,UXTR2        ,U14          ,UBAR(60)     ,USH
     1        ,USH1         ,USTREM                     ,UW(50)       ,UWS
     2        ,X11          ,X(60)        ,X12          ,X2(60)
     3        ,XB(50)       ,XBS          ,XABAR(10)    ,XBBAR(10)    ,X15
     4        ,XS(10,60)    ,XSH          ,XSTREM                     ,XW(50)
     5                                   ,Y11          ,Y(60)        ,Y12
     6        ,Y2(60)       ,YABAR        ,YBBAR        ,ZA(10)       ,Z11(10)
     7        ,ZJ(10,60)    ,ZMW          ,S2X          ,R2SH         ,X2SH
     8        ,FX(2,60)     ,FR(2,60)     ,FPHI(2,60)   ,FP(2,60)     ,FT(2,60)
     9        ,FU(2,60)     ,INDL(2,60)   ,INDR(2,60)   ,FC(2,60,10),CARD1
     *        ,M            ,N
      REAL    KAY           ,KAYS(10)     ,KAY2         ,KW           ,MDOT(60)
     *        ,MA(60)       ,MU           ,MUS(10)      ,MUC(10)      ,MU2
     *        ,MW(10)       ,MW2          ,MASH         ,MASH1        ,MWSH
      COMMON/MAINSH/ KOUNT,KINT
      COMMON/IBUGS1/ IBUGSH
      DOUBLE PRECISION ERROR(3)
      EQUIVALENCE (MOVE,IEXTRA(15))
      EQUIVALENCE (KALARM, IEXTRA(20))
      DATA  ERROR             / 4HMAIN,5HBNDRY,4HSTEP /
C
C     SETTING UP THE PROBLEM
C
      DO 5 I=1,7971
    5 A(I,1)=0.0
      KALARM=0
      KOUNT=0
      KINT=0
      LL=-1
  100 CALL PUTIN
      ICONT=0
      INTGER=ALPHAP
```

```
      NUCASE=0
      IERROR=0
      LL=0
      S=0.0
C
C     ESTABLISHING STABLE STEP SIZE
C
  200 CALL STABLE
      NCOUNT = 0
      ICOUNT=0
      MOVE=0
      IF (LL-LPLANE   ) 250,250,240
  240 CALL PDUMP(A(1,1),N,5)
      CALL EXIT
  250 IFLAG=0
      L=0
      IF (ICOUNT    ) 260,270,260
  260 S=S-DELSO
      DELSO=0.5*DELSO
      GO TO 295
  270 KSM=IEXTRA(7)
      FCARE=COS(PHI(KSM)-PHI(L))
      IF (FCARE-0.25) 280,280,290
  280 DELSO=DELS*0.25
      GO TO 295
  290 DELSO=DELS*FCARE
C
C     INNER BOUDARY OR WALL CONDITIONS
C
  295 S=S+DELSO
  300 NBOUND=0
      L=0
      CALL BNDRY
      EXTRA(1) = Y(L)
      EXTRA(2)=R(L)
      IF (K-KMAX) 350,1000,1000
  350 IF(NUCASE) 9960,500,9960
```

```
C
C     CALCULATION OF STREAMLINE CURVATURE
C
  500 IF (IFLAG)     700,600,700
  600 DPDY(K)=2.0*(P(K+1)-P(K))/(Y(K+1)-Y(K-1))
      DPHIDS(K)=-2.0*(P(K+1)-P(K))/(RHO(K+1)*U(K+1)**2*DELY(K+1)+RHO(K)*
     *U(K)**2*DELY(K))*BETAP
      GO TO 800
  700 DPDY(K)=2.0*(P2(K+1)-P2(K))/(Y2(K+1)-Y2(K-1))
      DPHODS   =-2.0*(P2(K+1)-P2(K))/(RHO2(K+1)*U2(K+1)**2*DELY(K+1)+R
     *HO2(K)*U2(K)**2*DELY(K))*BETAP
      DPHIDS(K)=.45*DPHIDS(K)+.55*DPHODS
C
C     LOCATING NEW GRID POINTS
C
  800 PHI1=PHI(K)+DPHIDS(K)*D LS(K-1)
      THETA=PI/2.-.5*(PHI2(K-1)+PHI1)
      IF (ABS(THETA-PI/2.)-(10.**(-4))) 810,815,815
  810 X2(K)=X2(K-1)
      R2(K)=R(K)+(X2(K)-X(K))*TAN(.5*(PHI(K)+PHI1))
      GO TO 820
  815 X2(K)=(R2(K-1)-R(K)+X2(K-1)*TAN(THETA)+X(K)*TAN(.5*(PHI(K)+PHI1)))
     */(TAN(.5*(PHI(K)+PHI1))+TAN(THETA))
      R2(K)=R2(K-1)-(X2(K)-X2(K-1))*TAN(THETA)
  820 D=SQRT((X2(K)-X(K))**2+(R2(K)-R(K))**2)
      IF (ABS(PHI1-PHI(K))-(10.**(-6))) 821,822,822
  821 DLS(K)=D
      GO TO 824
  822 D LS(K)=D*(PHI1-PHI(K))/(2.0*SIN(.5*(PHI1-PHI(K))))
  824 PHI2(K)=PHI(K)+D LS(K)*DPHIDS(K)
C
C     CALCULATION OF STREAMTUBE PROPERTIES
C
      B=SQRT((X2(K)-X2(K-1))**2+(R2(K)-R2(K-1))**2)
      IF (ABS(PHI2(K)-PHI2(K-1))-(10.**(-4)))  825,825,850
  825 DELY(K)=B
      GO TO 875
```

```
  850 DELY(K)=(PHI2(K)-PHI2(K-1))*B/(2.0*SIN(.5*(PHI2(K)-PHI2(K-1))))
  875 AA(K)=PI*(R2(K)+R2(K-1))**DELTA*DELY(K)
      Y2(K)=Y2(K-1)+DELY(K)
      CALL STEP
      IF (KALARM) 885,885,877
  877 KALARM=0
      IF (ICOUNT-3) 250,9910,9910
  885 IF (NUCASE) 9960,886,9960
  886 IF (MOVE) 1000,890,1000
  890 IF (IBUGSH) 960,970,960
  960 CALL PUTOUT(1)
  970 K=K+1
      IF (K-KMAX)  500,1000,1000
C
C     OUTER BOUNDARY CONDITIONS
C
 1000 NBOUND=1
      CALL BNDRY
      IF (MOVE) 800,1025,800
 1025 MOVE=0
      IF (KALARM) 1030,1030,877
 1030 IF (NUCASE) 9960,1050,9960
 1050 Y2(K)=Y2(K-1)+DELY(K)
      IF (IBUGSH) 1060,1070,1060
 1060 CALL PUTOUT(1)
 1070 IF (IFLAG) 1080,1150,1080
 1080 MAX=KMAX+1
      DO 1100 L=1,MAX
      J=L-1
      R(J)=R2(J)
      X(J)=X2(J)
      PHI(J)=PHI2(J)
      IF (J) 1100,1100,1082
 1082 U(J)=U2(J)
      P(J)=P2(J)
      T(J)=T2(J)
      RHO(J)=RHO2(J)
```

```
      ZMW=0.0
      DO 1084 I=1,NDS
 1084 ZMW=ZMW+C2(I,J)/MW(I)
      ZMW=1.0/ZMW
      DO 1086 I=1,NDS
      C(I,J)=C2(I,J)
 1086 XS(I,J)=C(I,J)*ZMW/MW(I)
      H(J)=0.0
      DO 1088 I=1,NDS
      DO 1088 JJ=1,NN
 1088 H(J)=H(J)+A(I,JJ)*T(J)**(JJ-1)*C(I,J)
      Y(J)=Y(J-1)+DELY(J)
      SX(J)=SX(J)+DLS(J)
 1100 CONTINUE
      IF (IKIND-3) 1120,1101,1110
 1101 IF (ITYPE-1) 1102,1102,1120
 1102 NEXT=KMAX-1
      IF (P(KMAX)-2.0*P(NEXT)) 1120,1104,1104
 1104 WRITE (6,1103)
 1103 FORMAT (28H1JET SHOCK BEGINNING TO FORM)
      CALL PUTOUT(1)
      CALL EXIT
 1110 CALL SHOCKE
 1120 ICONT=ICONT+1
      LL=LL+1
      ICOUNT=0
      IF (MOD(ICONT,INTGER)) 200,1130,200
 1130 K=KMAX
      CALL PUTOUT(1)
      GO TO 200
 1150 IF (IKIND-4) 1170,1160,1160
 1160 CALL SHOCKE
 1170 IFLAG=1
      GO TO 300
C
C     ERROR MESSAGES
C
```

```
9910 NUCASE=676
9955 IERROR=1
9960 WRITE (6,9965) ERROR(IERROR),NUCASE
9965 FORMAT(27H1ERROR ORIGIN IN SUBROUTINE,1X,A6,19H,  STATEMENT NUMBER
    *,I6,1H.)
     CALL PDUMP(A(1,1),N,5)
     GO TO 100
     END
```

```
      SUBROUTINE BNDRY
C
C     THIS SUBROUTINE HANDLES MOST BOUNDARY CONDITIONS
C
      COMMON A(10,5)      ,AA(60)       ,ALFA(10,10),ALPHAH     ,ALPHAP
     1       ,ATOL        ,BETAP        ,BMIX        ,C11(10)    ,C(10,60)
     2       ,C12(10)     ,C2(10,60)    ,CABAR(10)   ,CBBAR(10)  ,CP
     3       ,CPS(10)     ,CPSH         ,CSH(10)     ,CSH1(10)   ,CSTREM(10)
     4       ,MXSTRM      ,D2IH(10,10),DEFF(10)      ,D2EFF(10)  ,DELTA
     5       ,D11         ,DELSS(60)    ,DELS        ,DELSC      ,DLS(60)
     6       ,DIH(10,10)  ,D12          ,DELY(60)    ,D13        ,DPDY(60)
     7       ,D14         ,DPHIDS(60)   ,EPCON
     8       ,EPSLON      ,EXTRA(50)    ,FSTEP       ,FMAX       ,GRAD
     9       ,H11         ,H(60)        ,HH          ,HJ         ,HPM(10)
     *       ,H2PM(10)    ,H3PM(10)     ,ICONST      ,ICOUNT     ,IDENT(10)
     1       ,IERROR      ,IEXTRA(50)   ,IFLAG       ,IKIND
     2       ,IPTUC       ,ISHOCK       ,ITYPE       ,IPD
     3       ,IDIFF       ,K            ,KAY         ,KAYS       ,KAY2
     4       ,KLO         ,KMAX
     5       ,KUP         ,KW           ,LL          ,LPLANE     ,MA
     6       ,MASH        ,MOCT         ,MMAX        ,MUC        ,MU
     7       ,MU2         ,MUS          ,MW          ,MW2        ,MWSH
     8       ,NBOUND      ,NDS          ,NITER       ,NMAX       ,NN
```

```
      COMMON NOCASE       ,OMEGA(10)  ,P11        ,P(60)      ,P12
     1      ,P2(60)                   ,P6(50)     ,PABAR      ,PBBAR
     2      ,PBS          ,P15        ,PHI(60)                ,PHIB(50)
     3      ,PHISH1       ,PHSTRM                 ,PHIW(50)   ,P18
     4      ,PHI2(60)     ,PHIBS      ,PI         ,PR(10)     ,PSH
     5      ,PSH1         ,PSI        ,PSTREM                 ,PW(50)
     6      ,Q11          ,Q(60)      ,QXTR1      ,QXTR2
     7      ,QW(50)       ,R11        ,R(60)                  ,RB(50)
     8      ,RBS          ,R13        ,RBAR(60)   ,R14        ,R2(60)
     9      ,RCON         ,R15        ,RE(60)     ,RESH       ,R16
     *      ,RHC(60)      ,R17        ,RHO2(60)   ,RHS(10)    ,RHSEN
     1      ,RHSMOM       ,RHABAR     ,RHBBAR     ,RU         ,RSH
     2      ,RSTREM       ,RV         ,R19        ,RW(100)    ,S
     3                    ,SB(50)     ,SC(10)                 ,SW(50)
     4      ,S13          ,SX(60)     ,T11        ,T(60)      ,T12
     5      ,T2(60)       ,TABAR      ,TBBAR      ,TC(10)     ,T14
     6      ,TAW(60)      ,TXTR1      ,TXTR2      ,TOL        ,TSH
     7      ,TSH1         ,TSTREM                 ,TW(50)     ,TWS
     8      ,TS           ,U11        ,U(60)      ,U12        ,U2(60)
      COMMON UXTR1        ,UXTR2      ,U14        ,UBAR(60)   ,USH
     1      ,USH1         ,USTREM                 ,UW(50)     ,UWS
     2      ,X11          ,X(60)      ,X12        ,X2(60)
     3      ,XB(50)       ,XBS        ,XABAR(10)  ,XBBAR(10)  ,X15
     4      ,XS(10,60)    ,XSH        ,XSTREM                 ,XW(50)
     5                                ,Y11        ,Y(60)      ,Y12
     6      ,Y2(60)       ,YABAR      ,YBBAR      ,ZA(10)     ,Z11(10)
     7      ,ZJ(10,60)    ,ZMW        ,S2X        ,R2SH       ,X2SH
     8      ,FX(2,60)     ,FR(2,60)   ,FPHI(2,60) ,FP(2,60)   ,FT(2,60)
     9      ,FU(2,60)     ,INDL(2,60) ,INDR(2,60) ,FC(2,60,10),CARD1
     *      ,M            ,N
      REAL   KAY          ,KAYS(10)   ,KAY2       ,KW         ,MDOT(60)
     *      ,MA(60)       ,MU         ,MUS(10)    ,MUC(10)    ,MU2
     *      ,MW(10)       ,MW2        ,MASH       ,MASH1      ,MWSH
      EQUIVALENCE (MOVE,IEXTRA(15))
      EQUIVALENCE (PWSH,EXTRA(11))
      IF (MOVE) 4950,50,4950
   50 IF (NBOUND) 3100,100,3100
```

```
C
C      WALL CONDITIONS
C
  100 K=0
      IF (ITYPE-2) 200,110,101
  110 PWSH=PW(1)+(PW(2)-PW(1))*S/SW(2)
      GO TO 101
C
C      SHOCK BOUNDARY
C
  101 IF (IFLAG) 103,102,103
  102 DPDY(K)=2.0*(P(K+1)-PWSH)/(Y(K+1)-Y(K))
      DPHIDS(K)=-DPDY(K)*BETAP/(RHO(K+1)*U(K+1)**2)
      GO TO 104
  103 DPDY(K)=2.0*(P2(K+1)-PWSH)/(Y2(K+1)-Y(K))
      DPHODS    =-DPDY(K)*BETAP/(RHO2(K+1)*U2(K+1)**2)
      DPHIDS(K)=.45*DPHIDS(K)+.55*DPHODS
  104 PHI2(K)=PHI(K)+DPHIDS(K)*DLS(K)
      IF (ABS(PHI2(K)-PHI(K))-10.0**(-4)) 105,105,106
  105 X2(K)=X(K)+DLS(K)*COS(PHI(K))
      R2(K)=R(K)+DLS(K)*SIN(PHI(K))
      GO TO 107
  106 RADW=DLS(K)/(PHI(K)-PHI2(K))
      XO=X(K)+RADW*SIN(PHI(K))
      RO=R(K)-RADW*COS(PHI(K))
      X2(K)=XO-RADW*SIN(PHI2(K))
      R2(K)=RO+RADW*COS(PHI2(K))
  107 Y2(K)=0.0
      Y(K)=0.0
      MAX=KMAX
      IF (ITYPE-3) 575,1075,1075
 1075 CALL SHOCKE
      IF (KMAX-MAX) 108,108,109
  108 K=0
      GO TO 575
  109 K=1
      GO TO 575
```

```
  200 M=1
      MAX=KMAX
C
C     FIXED WALL
C
  300 IF (S-SW(M+1)) 500,500,400
  400 M=M+1
      IF (M-MMAX) 300,300,9030
  500 K=0
      Y(K)=0.0
      Y2(K)=0.0
      PHI2(K)=PHIW(M)+(PHIW(M+1)-PHIW(M))*(S-SW(M))/(SW(M+1)-SW(M))
      IF (ABS(PHIW(M+1)-PHIW(M))-10.**(-4)) 525,525,550
  525 X2(K)=XW(M)+(XW(M+1)-XW(M))*(S-SW(M))/(SW(M+1)-SW(M))
      R2(K)=RW(M)+(RW(M+1)-RW(M))*( S-SW(M))/(SW(M+1)-SW(M))
      GO TO 575
  550 RADW=DELSO/(PHI(K)-PHI2(K))
      XO=X(K)+RADW*SIN(PHI(K))
      RO=R(K)-RADW*COS(PHI(K))
      X2(K)=XO-RADW*SIN(PHI2(K))
      R2(K)=RADW*COS(PHI2(K))+RO
  575 IF (IFLAG) 600,900,600
  600 RBAR(K)=.5*(R(K)+R2(K))
      RBAR(K+1)=.5*(R(K+1)+R2(K+1))
      YABAR=.5*(Y(K+1)+Y2(K+1))
      YBBAR=0.0
      UBAR(K+1)=SQRT(.5*(U(K+1)**2+U2(K+1)**2))
      RHABAR=.5*(RHO(K+1)+RHO2(K+1))
      TABAR=.5*(T(K+1)+T2(K+1))
      PABAR=.5*(P(K+1)+P2(K+1))
      MW2=0.0
      DO 700 I=1,NDS
      CABAR(I)=.5*(C(I,K+1)+C2(I,K+1))
  700 MW2=MW2+CABAR(I)/MW(I)
      MW2=1.0/MW2
      DO 800 I=1,NDS
      XABAR(I)=CABAR(I)*MW2/MW(I)
```

```
      H2PM(I)=0.0
      HPM(I)=0.0
      DO 800 J=1,NN
  800 HPM(I)=HPM(I)+A(I,J)*TABAR**(J-1)
      GO TO 1050
  900 RBAR(K)=R(K)
      RBAR(K+1)=R(K+1)
      YABAR=Y(K+1)
      YBBAR=0.0
      UBAR(K+1)=U(K+1)
      RHABAR=RHO(K+1)
      TABAR=T(K+1)
      PABAR=P(K+1)
      DO 1000 I=1,NDS
      CABAR(I)=C(I,K+1)
      XABAR(I)=XS(I,K+1)
      H2PM(I)=0.0
      HPM(I)=0.0
      DO 1000 J=1,NN
 1000 HPM(I)=HPM(I)+A(I,J)*T(K+1)**(J-1)
 1050 IF (IEXTRA(1)) 1100,1200,1100
 1100 MULT=0
      IBIG=1
      CALL TRANSP (TABAR,PABAR,K,MULT,IBIG)
 1200 TAW(K)=0.0
      Q(K)=0.0
      DO 1300 I=1,NDS
 1300 ZJ(I,K)=0.0
 1500 IF (KMAX-MAX) 1510,1510,1520
 1510 K=1
      GO TO 10000
 1520 K=2
      GO TO 10000
C
C     OUTER BOUNDARY CONDITIONS
C
 3100 N=1
```

```
      IF (LL) 3110,3110,3190
 3110 D=SQRT((X(K)-XB(N))**2+(R(K)-RB(N))**2)
      IF (ABS(PHI(K)-PHIB(N))-1.E-04) 3120,3120,3130
 3120 SX(K)=0
      GO TO 3190
 3130 SX(K)=D*(PHI(K)-PHIB(N))/(2.*SIN(0.5*(PHI(K)-PHIB(N))))
 3190 GO TO (3200,5000,5100,5700), IKIND
C
C     FIXED BOUNDARY
C
 3200 N=1
      SSX=SX(K)+DLS(K-1)
 3300 IF (SSX-SB(N+1)) 3500,3500,3400
 3400 N=N+1
      IF (N-NMAX) 3300,9050,9050
 3500 IF (ABS(PHIB(N+1)-PHIB(N))-1.E-04) 3600,3600,3700
 3600 PHIBAR=0.5*(PHI2(K-1)+PHIB(N))
      IF (ABS(PHIBAR)-1.E-04) 3620,3620,3640
 3620 X2(K)=X2(K-1)
      GO TO 3660
 3640 X2(K)=(R2(K-1)-R(K)+TAN(PHIB(N))*X(K)+X2(K-1)/TAN(PHIBAR))/(TAN(PH
     *IB(N))+1./TAN(PHIBAR))
 3660 R2(K)=R(K)+TAN(PHIB(N))*(X2(K)-X(K))
      PHI2(K)=PHIB(N)
      GO TO 4200
 3700 RADB=(SB(N+1)-SB(N))/(PHIB(N)-PHIB(N+1))
      ELL1=SQRT((XB(N+1)-XB(N))**2+(RB(N+1)-RB(N))**2)
      ELL2=SQRT(RADB**2-ELL1**2/4.)
      FIOTA=ATAN((RB(N+1)-RB(N))/(XB(N+1)-XB(N)))
      IF (RADB) 3740,3740,3720
 3720 XX=XB(N)+.5*(XB(N+1)-XB(N))+ELL2*SIN(FIOTA)
      RR=RB(N)+.5*(RB(N+1)-RB(N))-ELL2*COS(FIOTA)
      GO TO 3800
 3740 XX=XB(N)+.5*(XB(N+1)-XB(N))-ELL2*SIN(FIOTA)
      RR=RB(N)+.5*(RB(N+1)-RB(N))+ELL2*COS(FIOTA)
 3800 L=0
      EKS=X(K)+DLS(K-1)*COS(PHI(K))
```

```
      IF (RADB) 3820,3840,3840
 3820 AHR=RR-SQRT(RADB**2-(EKS-XX)**2)
      GO TO 3900
 3840 AHR=RR+SQRT(RADB**2-(EKS-XX)**2)
 3900 PHIBAR=.5*(PHI2(K-1)+ATAN((EKS-XX)/(RR-AHR)))
      GEE=(R2(K-1)-RR-(EKS-X2(K-1))/TAN(PHIBAR))**2-RADB**2+(EKS-XX)**2
      DRDX=(EKS-XX)/SQRT(RADB**2-(EKS-XX)**2)
      IF (RADB) 3910,3910,3905
 3905 DRDX=-DRDX
 3910 DPHDX=-.5*((AHR-RR)-(EKS-XX)*DRDX)/RADB**2
      DTPH=DPHDX/(COS(PHIBAR))**2
      DGEEDX=2.*((EKS-XX)-(R2(K-1)-RR-(EKS-X2(K-1))/TAN(PHIBAR))*(1./TAN
     *(PHIBAR)-(EKS-X2(K-1))*DTPH/(TAN(PHIBAR))**2))
      DX=-GEE/DGEEDX
      L=L+1
      EKS=EKS+DX
      IF (RADB) 3920,3940,3940
 3920 AHR=RR-SQRT(RADB**2-(EKS-XX)**2)
      GO TO 4000
 3940 AHR=RR+SQRT(RADB**2-(EKS-XX)**2)
 4000 IF (L-3) 3900,4100,4100
 4100 X2(K)=EKS
      R2(K)=AHR
      PHI2(K)=-ATAN((EKS-XX)/(AHR-RR))
 4200 D=SQRT((X2(K)-X(K))**2+(R2(K)-R(K))**2)
      IF (ABS(PHI2(K)-PHI(K))-1.E-04) 4300,4300,4400
 4300 DLS(K)=D
      GO TO 4500
 4400 DLS(K)=D*(PHI2(K)-PHI(K))/(2.*SIN(0.5*(PHI2(K)-PHI(K))))
 4500 B=SQRT((X2(K)-X2(K-1))**2+(R2(K)-R2(K-1))**2)
      IF (ABS(PHI2(K)-PHI2(K-1))-1.E-04) 4600,4600,4700
 4600 DELY(K)=B
      GO TO 4800
 4700 DELY(K)=B*(PHI2(K)-PHI2(K-1))/(2.0*SIN(.5*(PHI2(K)-PHI2(K-1))))
 4800 AA(K)=PI*(R2(K)+R2(K-1))**DELTA*DELY(K)
      Y2(K)=Y2(K-1)+DELY(K)
 4900 CALL STEP
```

```
 4950 MOVE=0
      GO TO 10000
C
C     FREE BOUNDARY
C
 5000 N=1
      PBS=PB(N)+(PB(N+1)-PB(N))*(SX(K)-SB(N))/(SB(N+1)-SB(N))
      GO TO 5700
 5100 IF (LL) 5400,5200,5400
C
C     NEWTONIAN BOUNDARY
C
 5200 FMSTRM=0.0
      DO 5300 I=1,NDS
 5300 FMSTRM=FMSTRM+CSTREM(I)/MW(I)
      FMSTRM=1.0/FMSTRM
      QSTRM=PSTREM*FMSTRM*(USTREM**2)/(RV*TSTREM*BETAP)
 5400 IF (IFLAG) 5525,5500,5525
 5500 SIGMA=PHI(K)-PHSTRM
      GO TO 5550
 5525 SIGMA=.5*(PHI(K)+PHI2(K))-PHSTRM
 5550 IF (SIGMA) 5575,5575,5600
 5575 PBS=PSTREM
      GO TO 5700
 5600 IF (SIGMA-0.5*PI) 5650,5650,5625
 5625 PBS=PSTREM+QSTRM
      GO TO 5700
 5650 PBS=PSTREM+QSTRM *(SIN(SIGMA))**2
 5700 IF (IFLAG) 5900,5800,5900
 5800 DPDY(K)=2.0*(PBS-P(K))/(Y(K)-Y(K-1))
      DPHIDS(K)=-DPDY(K)*BETAP/(RHO(K)*U(K)**2)
      GO TO 6000
 5900 DPDY(K)=2.0*(PBS-P2(K))/(Y2(K)-Y2(K-1))
      DPHODS=-DPDY(K)*BETAP/(RHO2(K)*U2(K)**2)
      DPHIDS(K)=.45*DPHIDS(K)+.55*DPHODS
 6000 MOVE=1
      RETURN
```

```
C
C     ERROR MESSAGES
C
 9000 NUCASE=100
      GO TO 9900
 9030 NUCASE=400
      GO TO 9900
 9050 NUCASE=3200
 9900 IERROR=2
10000 RETURN
      END
```

```
      SUBROUTINE CHEM
C
C     THIS SUBROUTINE CALCULATES THE NEW MASS FRACTIONS AT EACH
C     STEP WHEN THE FLOW IS IN CHEMICAL NONEQUILIBRIUM.   (TECHNIQUE 1)
C
      COMMON A(10,5)      ,AA(60)       ,ALFA(10,10),ALPHAH      ,ALPHAP          C02
     1       ,ATOL        ,BETAP        ,BMIX       ,C11(10)     ,C(10,60)        003
     2       ,C12(10)     ,C2(10,60)    ,CABAR(10)  ,CBBAR(10)   ,CP              C04
     3       ,CPS(10)     ,CPSH         ,CSH(10)    ,CSH1(10)    ,CSTREM(10)      C05
     4       ,MXSTRM      ,D2IH(10,10),DEFF(10)     ,D2EFF(10)   ,DELTA           006
     5       ,D11         ,DELSS(60)    ,DELS       ,DELSC       ,DLS(60)         007
     6       ,DIH(10,10)  ,D12          ,DELY(60)   ,D13         ,DPDY(60)        008
     7       ,D14         ,DPHIDS(60)   ,EPCON                                    C09
     8       ,EPSLON      ,EXTRA(50)    ,FSTEP      ,FMAX        ,GRAD            C10
     9       ,H11         ,H(60)        ,HH         ,HJ          ,HPM(10)         C11
     *       ,H2PM(10)    ,H3PM(10)     ,ICONST     ,ICOUNT      ,IDENT(10)       C12
     1       ,IERROR      ,IEXTRA(50)   ,IFLAG      ,IKIND                        013
     2       ,IPTUC       ,ISHOCK       ,ITYPE      ,IPD                          014
     3       ,IDIFF       ,K            ,KAY        ,KAYS        ,KAY2            C15
     4       ,KLO         ,KMAX                                                   016
```

```
5          ,KUP          ,KW           ,LL           ,LPLANE       ,MA
6          ,MASH         ,MDOT         ,MMAX         ,MUC          ,MU
7          ,MU2          ,MUS          ,MW           ,MW2          ,MWSH
8          ,NBOUND       ,NDS          ,NITER        ,NMAX         ,NN
 COMMON NUCASE           ,OMEGA(10)    ,P11          ,P(60)        ,P12
1          ,P2(60)                     ,PB(50)       ,PABAR        ,PBBAR
2          ,PBS          ,P15          ,PHI(60)                    ,PHIB(50)
3          ,PHISH1       ,PHSTRM                     ,PHIW(50)     ,P18
4          ,PHI2(60)     ,PHIBS        ,PI           ,PR(10)       ,PSH
5          ,PSH1         ,PSI          ,PSTREM                     ,PW(50)
6          ,Q11          ,Q(60)        ,QXTR1        ,QXTR2
7          ,QW(50)       ,R11          ,R(60)                      ,RB(50)
8          ,RBS          ,R13          ,RBAR(60)     ,R14          ,R2(60)
9          ,RCON         ,R15          ,RE(60)       ,RESH         ,R16
*          ,RHO(60)      ,R17          ,RHO2(60)     ,RHS(10)      ,RHSEN
1          ,RHSMOM       ,RHABAR       ,RHBBAR       ,RU           ,RSH
2          ,RSTREM       ,RV           ,R19          ,RW(100)      ,S
3                        ,SB(50)       ,SC(10)                     ,SW(50)
4          ,S13          ,SX(60)       ,T11          ,T(60)        ,T12
5          ,T2(60)       ,TABAR        ,TBBAR        ,TC(10)       ,T14
6          ,TAW(60)      ,TXTR1        ,TXTR2        ,TOL          ,TSH
7          ,TSH1         ,TSTREM                     ,TW(50)       ,TWS
8          ,TS           ,U11          ,U(60)        ,U12          ,U2(60)
 COMMON UXTR1            ,UXTR2        ,U14          ,UBAR(60)     ,USH
1          ,USH1         ,USTREM                     ,UW(50)       ,UWS
2          ,X11          ,X(60)        ,X12          ,X2(60)
3          ,XB(50)       ,XBS          ,XABAR(10)    ,XBBAR(10)    ,X15
4          ,XS(10,60)    ,XSH          ,XSTREM                     ,XW(50)
5                                      ,Y11          ,Y(60)        ,Y12
6          ,Y2(60)       ,YABAR        ,YBBAR        ,ZA(10)       ,Z11(10)
7          ,ZJ(10,60)    ,ZMW          ,S2X          ,R2SH         ,X2SH
8          ,FX(2,60)     ,FR(2,60)     ,FPHI(2,60)   ,FP(2,60)     ,FT(2,60)
9          ,FU(2,60)     ,INDL(2,60)   ,INDR(2,60)   ,FC(2,60,10)  ,CARD1
*          ,M            ,N
 REAL       KAY          ,KAYS(10)     ,KAY2         ,KW           ,MDOT(60)
*          ,MA(60)       ,MU           ,MUS(10)      ,MUO(10)      ,MU2
*          ,MW(10)       ,MW2          ,MASH         ,MASH1        ,MWSH
```

```
      COMMON/CHEMI/NU1(20,15),NU2(20,15),RF1(20),RF2(20),RF3(20),
     *RB1(20),RB2(20),RB3(20),NCAT,IBURN,NCR,SPROD(10),ICAT(5,10)
      DIMENSION  YC(15),RFOR(20),RBAC(20),RM1(20),RM2(20)
      IF (IFLAG.EQ.1) GO TO 500
      DO 100 I=1,NDS
      YC(I)=RHO(K)*C(I,K)/MW(I)
  100 CONTINUE
      IF (NCAT.EQ.0) GO TO 300
      DO 250 J=1,NCAT
      YC(NDS+J)=0.0
      DO 200 I=1,NDS
  200 YC(NDS+J)=YC(NDS+J)+ICAT(J,I)*YC(I)
  250 CONTINUE
  300 DO 400 M1=1,NCR
      RFOR(M1)=RF1(M1)*(T(K)**RF2(M1))*EXP(-RF3(M1)/T(K))
      RBAC(M1)=RB1(M1)*(T(K)**RB2(M1))*EXP(-RB3(M1)/T(K))
  400 CONTINUE
      GO TO 1000
  500 T3=(T(K)+T2(K))/2.0
      RHO3=(RHO(K)+RHO2(K))/2.0
      U3=SQRT(.5*(U(K)**2+U2(K)**2))
      DO 600 I=1,NDS
      YC(I)=RHO3*(C(I,K)+C2(I,K))/(2.0*MW(I))
  600 CONTINUE
      IF (NCAT.EQ.0) GO TO 800
      DO 750 J=1,NCAT
      YC(NDS+J)=0.0
      DO 700 I=1,NDS
  700 YC(NDS+J)=YC(NDS+J)+ICAT(J,I)*YC(I)
  750 CONTINUE
  800 DO 900 M1=1,NCR
      RFOR(M1)=RF1(M1)*(T3**RF2(M1))*EXP(-RF3(M1)/T3)
      RBAC(M1)=RB1(M1)*(T3**RB2(M1))*EXP(-RB3(M1)/T3)
  900 CONTINUE
 1000 L=NDS+NCAT
      DO 1105 M1=1,NCR
      PROD1=1.0
```

```
      PROD2=1.0
      DO 1100 J=1,L
      IF (NU1(M1,J).EQ.0) GO TO 1050
      PROD1=(PROD1)*(YC(J)**NU1(M1,J))
 1050 IF (NU2(M1,J).EQ.0) GO TO 1100
      PROD2=(PROD2)*(YC(J)**NU2(M1,J))
 1100 CONTINUE
      RM1(M1)=RFOR(M1)*PROD1
 1105 RM2(M1)=RBAC(M1)*PROD2
      DO 1400 I=1,NDS
      SPRO1=0.0
      SPRO2=0.0
      DO 1200 M1=1,NCR
      IF (NU2(M1,I)-NU1(M1,I)) 1110,1120,1130
 1110 LAM1=0
      LAM2=-(NU2(M1,I)-NU1(M1,I))
      GO TO 1140
 1120 LAM1=0
      LAM2=0
      GO TO 1140
 1130 LAM1=NU2(M1,I)-NU1(M1,I)
      LAM2=0
 1140 SPRO1=SPRO1+LAM1*RM1(M1)+LAM2*RM2(M1)
 1200 SPRO2=SPRO2+LAM1*RM2(M1)+LAM2*RM1(M1)
      SPROD(I)=SPRO1-SPRO2
      IF(IFLAG.EQ.1) GO TO 1300
      SPROD(I)=MW(I)*SPROD(I)*(DLS(K-1)+DLS(K))/(RHO(K)*U(K)*2.0)
      GO TO 1400
 1300 SPROD(I)=MW(I)*SPROD(I)*(DLS(K-1)+DLS(K))/(RHO3*U3*2.0)
 1400 CONTINUE
      DO 1450 I=1,NDS
 1450 C2(I,K)=C(I,K)+RHS(I)/MDOT(K)+SPROD(I)
      RETURN
      END
```

```
      SUBROUTINE CHEM
C
C     THIS SUBROUTINE CALCULATES THE NEW MASS FRACTIONS AT EACH
C     STEP WHEN THE FLOW IS IN CHEMICAL NONEQUILIBRIUM.   (TECHNIQUE 2)
C
      COMMON A(10,5)      ,AA(60)       ,ALFA(10,10),ALPHAH      ,ALPHAP
     1      ,ATOL         ,BETAP        ,BMIX        ,C11(10)     ,C(10,60)
     2      ,C12(10)      ,C2(10,60)    ,CABAR(10)   ,CBBAR(10)   ,CP
     3      ,CPS(10)      ,CPSH         ,CSH(10)     ,CSH1(10)    ,CSTREM(10)
     4      ,MXSTRM       ,D2IH(10,10),DEFF(10)      ,D2EFF(10)   ,DELTA
     5      ,D11          ,DELSS(60)    ,DELS        ,DELSC       ,DLS(60)
     6      ,DIH(10,10)   ,D12          ,DELY(60)    ,D13         ,DPDY(60)
     7      ,D14          ,DPHIDS(60)   ,EPCON
     8      ,EPSLCN       ,EXTRA(50)    ,FSTEP       ,FMAX        ,GRAD
     9      ,H11          ,H(60)        ,HH          ,HJ          ,HPM(10)
     *      ,H2PM(10)     ,H3PM(10)     ,ICONST      ,ICOUNT      ,IDENT(10)
     1      ,IERROR       ,IEXTRA(50)   ,IFLAG       ,IKIND
     2      ,IPTUC        ,ISHOCK       ,ITYPE       ,IPD
     3      ,IDIFF        ,K            ,KAY         ,KAYS        ,KAY2
     4      ,KLO          ,KMAX
     5      ,KUP          ,KW           ,LL          ,LPLANE      ,MA
     6      ,MASH         ,MDOT         ,MMAX        ,MUC         ,MU
     7      ,MU2          ,MUS          ,MW          ,MW2         ,MWSH
     8      ,NBOUND       ,NDS          ,NITER       ,NMAX        ,NN
      COMMON NUCASE       ,OMEGA(10)    ,P11         ,P(60)       ,P12
     1      ,P2(60)                     ,PB(50)      ,PABAR       ,PBBAR
     2      ,PBS          ,P15          ,PHI(60)                  ,PHIB(50)
     3      ,PHISH1       ,PHSTRM                    ,PHIW(50)    ,P18
     4      ,PHI2(60)     ,PHIBS        ,PI          ,PR(10)      ,PSH
     5      ,PSH1         ,PSI          ,PSTREM                   ,PW(50)
     6      ,Q11          ,Q(60)        ,QXTR1       ,QXTR2
     7      ,QW(50)       ,R11          ,R(60)                    ,RB(50)
     8      ,RBS          ,R13          ,RBAR(60)    ,R14         ,R2(60)
     9      ,RCON         ,R15          ,RE(60)      ,RESH        ,R16
     *      ,RHO(60)      ,R17          ,RHO2(60)    ,RHS(10)     ,RHSEN
     1      ,RHSMOM       ,RHABAR       ,RHBBAR      ,RU          ,RSH
     2      ,RSTREM       ,RV           ,R19         ,RW(100)     ,S
```

```
 3                      ,SB(50)     ,SC(10)                 ,SW(50)
 4      ,S13            ,SX(60)     ,T11        ,T(60)      ,T12
 5      ,T2(60)         ,TABAR      ,TBBAR      ,TC(10)     ,T14
 6      ,TAW(60)        ,TXTR1      ,TXTR2      ,TCL        ,TSH
 7      ,TSH1           ,TSTREM                 ,TW(50)     ,TWS
 8      ,TS             ,U11        ,U(60)      ,U12        ,U2(60)
  COMMON UXTR1          ,UXTR2      ,U14        ,UBAR(60)   ,USH
 1      ,USH1           ,USTREM                 ,UW(50)     ,UWS
 2      ,X11            ,X(60)      ,X12        ,X2(60)
 3      ,XB(50)         ,XBS        ,XABAR(10)  ,XBBAR(10)  ,X15
 4      ,XS(10,60)      ,XSH        ,XSTREM                 ,XW(50)
 5                                  ,Y11        ,Y(60)      ,Y12
 6      ,Y2(60)         ,YABAR      ,YBBAR      ,ZA(10)     ,Z11(10)
 7      ,ZJ(10,50)      ,ZMW        ,S2X        ,R2SH       ,X2SH
 8      ,FX(2,60)       ,FR(2,60)   ,FPHI(2,60) ,FP(2,60)   ,FT(2,60)
 9      ,FU(2,60)       ,INDL(2,60) ,INDR(2,60) ,FC(2,60,10),CARD1
 *      ,M              ,N
  REAL   KAY            ,KAYS(10)   ,KAY2       ,KW         ,MDOT(60)
 *      ,MA(50)         ,MU         ,MUS(10)    ,MUC(10)    ,MU2
 *      ,MW(10)         ,MW2        ,MASH       ,MASH1      ,MWSH
  COMMON/CHEMI/NU1(20,15),NU2(20,15),RF1(20),RF2(20),RF3(20),
 *RB1(20),RB2(20),RB3(20),NCAT,IBURN,NCR,SPRCD(10),ICAT(5,10)
  DIMENSION  YC(15),RFOR(20),RBAC(20),COEFF(8,8),CONST(8,1),
 *ZZJJ(15)
  IF (IFLAG.EQ.1) GO TO 500
  RHO3=RHO(K)
  RHO4=RHO(K)
  DO 100 I=1,NDS
  YC(I)=RHO(K)*C(I,K)/MW(I)
100 CONTINUE
  IF (NCAT.EQ.0) GO TO 300
  DO 250 J=1,NCAT
  YC(NDS+J)=0.0
  DO 200 I=1,NDS
200 YC(NDS+J)=YC(NDS+J)+ICAT(J,I)*YC(I)
250 CONTINUE
300 DO 400 M1=1,NCR
```

```
      RFOR(M1)=RF1(M1)*(T(K)**RF2(M1))*EXP(-RF3(M1)/T(K))
      RBAC(M1)=RB1(M1)*(T(K)**RB2(M1))*EXP(-RB3(M1)/T(K))
  400 CONTINUE
      TIME=(DLS(K-1)+DLS(K))/(2.0*U(K))
      GO TO 1000
  500 T3=(T(K)+T2(K))/2.0
      RHO3=(RHO(K)+RHO2(K))/2.0
      RHO4=RHO2(K)
      U3=SQRT(.5*(U(K)**2+U2(K)**2))
      DO 600 I=1,NDS
      YC(I)=RHO(K)*C(I,K)/MW(I)
  600 CONTINUE
      IF (NCAT.EQ.0) GO TO 800
      DO 750 J=1,NCAT
      YC(NDS+J)=0.0
      DO 700 I=1,NDS
  700 YC(NDS+J)=YC(NDS+J)+ICAT(J,I)*YC(I)
  750 CONTINUE
  800 DO 900 M1=1,NCR
      RFOR(M1)=RF1(M1)*(T3**RF2(M1))*EXP(-RF3(M1)/T3)
      RBAC(M1)=RB1(M1)*(T3**RB2(M1))*EXP(-RB3(M1)/T3)
  900 CONTINUE
      TIME=(DLS(K-1)+DLS(K))/(2.0*U3)
 1000 YYY=0.0
      COEFF(1,1)=RHO3/RHO4+(TIME/2)*(RBAC(1)*YC(8)+RBAC(2)*YC(3)+RBAC(4)
     1*YC(4)+RFOR(5)*YC(6)+RFOR(6)*YC(5)*YC(10)+4*RFOR(8)*YC(1)*YC(10)+
     2RFOR(9)*YC(4)*YC(10))
      COEFF(1,2)=(TIME/2)*(-RFOR(2)*YC(4)-RFOR(4)*YC(5)-RBAC(8)*YC(10)*2
     1)
      COEFF(1,3)=(TIME/2)*(RBAC(2)*YC(1)-RBAC(9)*YC(10))
      COEFF(1,4)=(TIME/2)*(-RFOR(1)*YC(7)-RFOR(2)*YC(2)+RBAC(4)*YC(1)-
     1RBAC(5)*YC(5)-RBAC(6)*YC(10)+RFOR(9)*YC(1)*YC(10))
      COEFF(1,5)=(TIME/2)*(-RFOR(4)*YC(2)-RBAC(5)*YC(4)+RFOR(6)*YC(1)*
     1YC(10))
      COEFF(1,6)=(TIME/2)*(RFOR(5)*YC(1))
      COEFF(1,7)=(TIME/2)*(-RFOR(1)*YC(4))
      COEFF(1,8)=(TIME/2)*(RBAC(1)*YC(1))
```

```
      CONST(1,1)=(RHO3/RHO(K))*YC(1)+(TIME/2)*(RBAC(6)*YC(4)*YC(10)+2*
     1RBAC(8)*YC(2)*YC(10)+RBAC(9)*YC(3)*YC(10))+RHS(1)*RHO3/(MDOT(K)*
     2MW(1))
      COEFF(2,1)=(TIME/2)*(-RBAC(2)*YC(3)-RBAC(4)*YC(4)-2*RFOR(8)*YC(1)*
     1YC(10))
      COEFF(2,2)=RHO3/RHO4+(TIME/2)*(RFOR(2)*YC(4)+RFOR(4)*YC(5)+RBAC(8)
     1*YC(10))
      COEFF(2,3)=(TIME/2)*(-RBAC(2)*YC(1))
      COEFF(2,4)=(TIME/2)*(RFOR(2)*YC(2)-RBAC(4)*YC(1))
      COEFF(2,5)=(TIME/2)*(RFOR(4)*YC(2))
      COEFF(2,6)=0.0
      COEFF(2,7)=0.0
      COEFF(2,8)=0.0
      CONST(2,1)=(RHO3/RHO(K))*YC(2)+(TIME/2)*(-RBAC(8)*YC(2)*YC(10))+
     1RHS(2)*RHO3/(MDOT(K)*MW(2))
      COEFF(3,1)=(TIME/2)*(RBAC(2)*YC(3)-RFOR(9)*YC(4)*YC(10))
      COEFF(3,2)=(TIME/2)*(-RFOR(2)*YC(4))
      COEFF(3,3)=RHO3/RHO4+(TIME/2)*(RBAC(2)*YC(1)+RBAC(3)*YC(5)+RBAC(9)
     1*YC(10))
      COEFF(3,4)=(TIME/2)*(-RFOR(2)*YC(2)-2*RFOR(3)*YC(4)-RFOR(9)*YC(1)*
     1YC(10))
      COEFF(3,5)=(TIME/2)*(RBAC(3)*YC(3))
      COEFF(3,6)=0.0
      COEFF(3,7)=0.0
      COEFF(3,8)=0.0
      CONST(3,1)=(RHO3/RHO(K))*YC(3)+(TIME/2)*(-RBAC(9)*YC(3)*YC(10))+
     1RHS(3)*RHO3/(MDOT(K)*MW(3))
      COEFF(4,1)=(TIME/2)*(-RBAC(1)*YC(8)-RBAC(2)*YC(3)+RBAC(4)*YC(4)-
     1RFOR(5)*YC(6)-RFOR(6)*YC(5)*YC(10)+RFOR(9)*YC(4)*YC(10))
      COEFF(4,2)=(TIME/2)*(RFOR(2)*YC(4)-RFOR(4)*YC(5))
      COEFF(4,3)=(TIME/2)*(-RBAC(2)*YC(1)-2*RBAC(3)*YC(5)-RBAC(9)*YC(10)
     1)
      COEFF(4,4)=RHO3/RHO4+(TIME/2)*(RFOR(1)*YC(7)+RFOR(2)*YC(2)+4*RFOR
     1(3)*YC(4)+RBAC(4)*YC(1)+RBAC(5)*YC(5)+RBAC(6)*YC(10)+RFOR(9)*YC(1)
     2*YC(10))
      COEFF(4,5)=(TIME/2)*(-2*RBAC(3)*YC(3)-RFOR(4)*YC(2)+RBAC(5)*YC(4)-
     1RFOR(6)*YC(1)*YC(10))
```

```
      COEFF(4,6)=(TIME/2)*(-RFOR(5)*YC(1))
      COEFF(4,7)=(TIME/2)*(RFOR(1)*YC(4))
      COEFF(4,8)=(TIME/2)*(-RBAC(1)*YC(1))
      CONST(4,1)=(RHO3/RHO(K))*YC(4)+(TIME/2)*(-RBAC(6)*YC(4)*YC(10)+
     1RBAC(9)*YC(3)*YC(10))+RHS(4)*RHO3/(MDOT(K)*MW(4))
      COEFF(5,1)=(TIME/2)*(-RBAC(4)*YC(4)-RFOR(5)*YC(6)+RFOR(6)*YC(5)*
     1YC(10))
      COEFF(5,2)=(TIME/2)*(RFOR(4)*YC(5))
      COEFF(5,3)=(TIME/2)*(RBAC(3)*YC(5))
      COEFF(5,4)=(TIME/2)*(-2*RFOR(3)*YC(4)-RBAC(4)*YC(1)+RBAC(5)*YC(5)-
     1RBAC(6)*YC(10))
      COEFF(5,5)=RHO3/RHO4+(TIME/2)*(RBAC(3)*YC(3)+RFOR(4)*YC(2)+RBAC(5)
     1*YC(4)+RFOR(6)*YC(1)*YC(10)+4*RFOR(7)*YC(5)*YC(10))
      COEFF(5,6)=(TIME/2)*(-RFOR(5)*YC(1)-2*RBAC(7)*YC(10))
      COEFF(5,7)=0.0
      COEFF(5,8)=0.0
      CONST(5,1)=(RHO3/RHO(K))*YC(5)+(TIME/2)*(RBAC(6)*YC(4)*YC(10)+2*
     1RBAC(7)*YC(6)*YC(10))+RHS(5)*RHO3/(MDOT(K)*MW(5))
      COEFF(6,1)=(TIME/2)*(RFOR(5)*YC(6))
      COEFF(6,2)=0.0
      COEFF(6,3)=0.0
      COEFF(6,4)=(TIME/2)*(-RBAC(5)*YC(5))
      COEFF(6,5)=(TIME/2)*(-RBAC(5)*YC(4)-2*RFOR(7)*YC(5)*YC(10))
      COEFF(6,6)=RHO3/RHO4+(TIME/2)*(RFOR(5)*YC(1)+RBAC(7)*YC(10))
      COEFF(6,7)=0.0
      COEFF(6,8)=0.0
      CONST(6,1)=(RHO3/RHO(K))*YC(6)+(TIME/2)*(-RBAC(7)*YC(6)*YC(10))+
     1RHS(6)*RHO3/(MDOT(K)*MW(6))
      COEFF(7,1)=(TIME/2)*(-RBAC(1)*YC(8))
      COEFF(7,2)=0.0
      COEFF(7,3)=0.0
      COEFF(7,4)=(TIME/2)*(RFOR(1)*YC(7))
      COEFF(7,5)=0.0
      COEFF(7,6)=0.0
      COEFF(7,7)=RHO3/RHO4+(TIME/2)*(RFOR(1)*YC(4))
      COEFF(7,8)=(TIME/2)*(-RBAC(1)*YC(1))
      CONST(7,1)=(RHO3/RHO(K))*YC(7)+RHS(7)*RHO3/(MDOT(K)*MW(7))
```

```
      COEFF(8,1)=(TIME/2)*(RBAC(1)*YC(8))
      COEFF(8,2)=0.0
      COEFF(8,3)=0.0
      COEFF(8,4)=(TIME/2)*(-RFOR(1)*YC(7))
      COEFF(8,5)=0.0
      COEFF(8,6)=0.0
      COEFF(8,7)=(TIME/2)*(-RFOR(1)*YC(4))
      COEFF(8,8)=RHO3/RHO4+(TIME/2)*(RBAC(1)*YC(1))
      CONST(8,1)=(RHO3/RHO(K))*YC(8)+RHS(8)*RHO3/(MDOT(K)*MW(8))
 1600 CALL MATE7 (8,8,1,COEFF,CONST,YYY,ZZJJ,J)
      GO TO (1700,1655,1675),J
 1655 WRITE (6,1660)
 1660 FORMAT(40H1DETERMINANT OVERFLOW IN SUBROUTINE CHEM)
      GO TO 1690
 1675 WRITE (6,1680)
 1680 FORMAT (40H1SINGULAR DETERMINANT IN SUBROUTINE CHEM)
 1690 CALL DUMP(A(1,1),N,5)
 1700 CONTINUE
      DO 1800 I=1,8
      C2(I,K)=COEFF(I,1)*MW(I)/RHO4
 1800 CONTINUE
      C2(9,K)=C(9,K)+RHS(9)/MDOT(K)
      RETURN
      END


      SUBROUTINE COMBO(L)
C
C     THIS SUBROUTINE COMBINES TWO STREAMTUBES, CONSERVING MASS,
C     MOMENTIM, AND ENERGY, WHENEVER THE ALLOWED STEPPING DISTANCE
C     BECOMES TOO LOW.
C
      COMMON A(10,5)    ,AA(60)     ,ALFA(10,10),ALPHAH     ,ALPHAP
```

```
1     ,ATOL       ,BETAP      ,BMIX       ,C11(10)    ,C(10,60)
2     ,C12(10)    ,C2(10,60)  ,CABAR(10)  ,CBBAR(10)  ,CP
3     ,CPS(10)    ,CPSH       ,CSH(10)    ,CSH1(10)   ,CSTREM(10)
4     ,MXSTRM     ,D2IH(10,10),DEFF(10)   ,D2EFF(10)  ,DELTA
5     ,D11        ,DELSS(60)  ,DELS       ,DELSO      ,DLS(60)
6     ,DIH(10,10) ,D12        ,DELY(60)   ,D13        ,DPDY(60)
7     ,D14        ,DPHIDS(60) ,EPCON
8     ,EPSLON     ,EXTRA(50)  ,FSTEP      ,FMAX       ,GRAD
9     ,H11        ,H(60)      ,HH         ,HJ         ,HPM(10)
*     ,H2PM(10)   ,H3PM(10)   ,ICONST     ,ICOUNT     ,IDENT(10)
1     ,IERROR     ,IEXTRA(50) ,IFLAG      ,IKIND
2     ,IPTUC      ,ISHOCK     ,ITYPE      ,IPD
3     ,IDIFF      ,K          ,KAY        ,KAYS       ,KAY2
4     ,KLO        ,KMAX
5     ,KUP        ,KW         ,LL         ,LPLANE     ,MA
6     ,MASH       ,MDOT       ,MMAX       ,MU0        ,MU
7     ,MU2        ,MUS        ,MW         ,MW2        ,MWSH
8     ,NBOUND     ,NDS        ,NITER      ,NMAX       ,NN
 COMMON NUCASE    ,OMEGA(10)  ,P11        ,P(60)      ,P12
1     ,P2(60)                 ,PB(50)     ,PABAR      ,PBBAR
2     ,PBS        ,P15        ,PHI(60)                ,PHIB(50)
3     ,PHISH1     ,PHSTRM                 ,PHIW(50)   ,P18
4     ,PHI2(60)   ,PHIBS      ,PI         ,PR(10)     ,PSH
5     ,PSH1       ,PSI        ,PSTREM                 ,PW(50)
6     ,Q11        ,Q(60)      ,QXTR1      ,QXTR2
7     ,QW(50)     ,R11        ,R(60)                  ,RB(50)
8     ,RBS        ,R13        ,RBAR(60)   ,R14        ,R2(60)
9     ,RCON       ,R15        ,RE(60)     ,RESH       ,R16
*     ,RHO(60)    ,R17        ,RHO2(60)   ,RHS(10)    ,RHSEN
1     ,RHSMOM     ,RHABAR     ,RHBBAR     ,RU         ,RSH
2     ,RSTREM     ,RV         ,R19        ,RW(100)    ,S
3                 ,SB(50)     ,SC(10)                 ,SW(50)
4     ,S13        ,SX(60)     ,T11        ,T(60)      ,T12
5     ,T2(60)     ,TABAR      ,TBBAR      ,TC(10)     ,T14
6     ,TAW(60)    ,TXTR1      ,TXTR2      ,TOL        ,TSH
7     ,TSH1       ,TSTREM                 ,TW(50)     ,TWS
8     ,TS         ,U11        ,U(60)      ,U12        ,U2(60)
```

```
      COMMON UXTR1         ,UXTR2         ,U14          ,UBAR(60)     ,USH
     1      ,USH1          ,USTREM                      ,UW(50)       ,UWS
     2      ,X11           ,X(60)         ,X12          ,X2(60)
     3      ,XB(50)        ,XBS           ,XABAR(10)    ,XBBAR(10)    ,X15
     4      ,XS(10,60)     ,XSH           ,XSTREM                     ,XW(50)
     5                                    ,Y11          ,Y(50)        ,Y12
     6      ,Y2(60)        ,YABAR         ,YBBAR        ,ZA(10)       ,Z11(10)
     7      ,ZJ(10,60)     ,ZMW           ,S2X          ,R2SH         ,X2SH
     8      ,FX(2,60)      ,FR(2,60)      ,FPHI(2,60)   ,FP(2,60)     ,FT(2,60)
     9      ,FU(2,60)      ,INDL(2,60)    ,INDR(2,60)   ,FC(2,60,10)  ,CARD1
     *      ,M             ,N
      REAL   KAY           ,KAYS(10)      ,KAY2         ,KW           ,MDOT(60)
     *      ,MA(60)        ,MU            ,MUS(10)      ,MUC(10)      ,MU2
     *      ,MW(10)        ,MW2           ,MASH         ,MASH1        ,MWSH
      COMMON/IBUGS1/ IBUGSH
      DIMENSION FTEST(3), NTEST(3)
      SUMDOT=0.0
      DO 50 KK=1,KMAX
   50 SUMDOT=SUMDOT+MDOT(KK)
      IF (MDOT(L)/(.5*(R(L)+R(L-1)))-SUMDOT/(.75*(R(1)+R(KMAX))*FSTEP))
     * 100,10000,10000
C
C     SETUP OF TESTS TO DETERMINE IF GRADIENTS ARE TOO HIGH TO ALLOW
C     COMBINING TUBES
C
  100 KI=L-1
      KF=L+1
      DO 150 L2=1,3
  150 NTEST(L2)=0
      NTRY=0
      IF (IBUGSH) 180,190,180
  180 WRITE (6,185)
  185 FORMAT (32H1   STREAMTUBES BEFORE COMBINING)
      CALL PUTOUT(1)
  190 GO TO (200,300,400,500),IPTUC
  200 DO 250 KJ=KI,KF
      IARG=KJ-KI+1
```

```
 250 FTEST( IARG  )=P(KJ)
     GO TO 600
 300 DO 350 KJ=KI,KF
     IARG=KJ-KI+1
 350 FTEST( IARG  )=T(KJ)
     GO TO 600
 400 DO 450 KJ=KI,KF
     IARG=KJ-KI+1
 450 FTEST( IARG  )=U(KJ)
     GO TO 600
 500 DO 550 KJ=KI,KF
     IARG=KJ-KI+1
 550 FTEST( IARG  )=C(1,KJ)
     IF (ABS(FTEST(1)-FTEST(2))-1.E-05) 570,570,560
 560 IF (IEXTRA(1)) 570,565,570
 565 NTEST(1)=1
 570 IF (ABS(FTEST(3)-FTEST(2))-1.E-05) 600,600,580
 580 IF (IEXTRA(1)) 600,585,600
 585 NTEST(3)=1
 600 IF (L-KMAX) 700,900,900
 700 IF (L-1) 1000,1000,800
 800 IF (MDOT(L+1)-MDOT(L-1)) 1000,1000,900
 900 IF (L-1) 10000,10000,950
 950 K2=KI
     L2=1
     RUP=R(L)
     RLO=R(KI-1)
     XUP=X(L)
     XLO=X(KI-1)
     PHIUP=PHI(L)
     PHILO=PHI(KI-1)
     GO TO 1100
1000 IF (L-KMAX) 1050,10000,10000
1050 K2=KF
     L2=3
     RUP=R(KF)
     RLO=R(KI)
```

```
      XUP=X(KF)
      XLO=X(KI)
      PHIUP=PHI(KF)
      PHILO=PHI(KI)
 1100 IF (FTEST(2)) 1120,1105,1120
 1105 IF (FTEST(L2)) 1110,1115,1110
 1110 IF (ABS(FTEST(2)-FTEST(L2))/FTEST(L2)-GRAD) 1150,1160,1160
 1115 IF (GRAD) 1160,1150,1160
 1120 IF (ABS(FTEST(2)-FTEST(L2))/FTEST(2)-GRAD) 1150,1160,1160
 1150 IF (NTEST(L2)) 1160,1200,1160
 1160 IF (NTRY) 10000,1170,10000
 1170 NTRY=1
      IF (L2-2) 1000,10000,900
C
C     COMBINING TUBES
C
 1200 D=SQRT((XUP-XLO)**2+(RUP-RLO)**2)
      IF (ABS(PHIUP-PHILO)-(10.**(-4))) 1130,1130,1125
 1125 D=D*(PHIUP-PHILO)/(2.0*SIN(0.5*(PHIUP-PHILO)))
 1130 DELY(L)=D
      FMOM=MDOT(L)*U(L)+MDOT(K2)*U(K2)+(AA(L)*P(L)+AA(K2)*P(K2))*BETAP
      AA(L)=PI*(RUP+RLO)**DELTA*DELY(L)
      FMDOT=MDOT(L)
      MDOT(L)=FMDOT+MDOT(K2)
      FH=(FMDOT*H(L)+MDOT(K2)*H(K2)+(FMDOT*U(L)**2+MDOT(K2)*U(K2)**2)/(2
     *.0*HJ))/MDOT(L)
 2100 SZMWC=0.0
      DO 2200 I=1,NDS
      C(I,L)=(FMDOT*C(I,L)+MDOT(K2)*C(I,K2))/MDOT(L)
      ZMWC=C(I,L)/MW(I)
 2200 SZMWC=SZMWC+ZMWC
      ZMW=1.0/SZMWC
      ITER=0
      CP=0.0
      CP1=0.0
      CP2=0.0
      TS=0.5*(T(L)+T(K2))
```

```
      US=0.5*(U(L)+U(K2))
      DO 2300 I=1,NDS
      XS(I,L)=C(I,L)*ZMW/MW(I)
      DO 2300 J=1,NN
      CP1=CP1+A(I,J)*FLOAT(J-1)*C(I,L)*T(K2)**(J-2)
 2250 CP2=CP2+A(I,J)*FLOAT(J-1)*C(I,L)*T(K )**(J-2)
 2300 CP=CP+A(I,J)*FLOAT(J-1)*C(I,L)*T(L)**(J-2)
      B1=FMOM/(BETAP*AA(L))
      B2=MDOT(L)/(BETAP*AA(L))
      B3=ZMW*AA(L)/(RV*MDOT(L))
      IF (ICONST) 2305,2310,2305
 2305 IF (0.5*(MA(L)+MA(K2))-20.) 2600,2310,2310
C
C      CONSTANT HEAT CAPACITY
C
 2310 B4=2.0*HJ*(CP1*T(K2)*MDOT(K2)+CP2*T(L)*FMDOT)/MDOT(L)+(MDOT(K2)*U
     *(K2)**2+FMDOT*U(L)**2)/MDOT(L)
      B5=HJ*CP*B3*B1
      B6=2.0*HJ*CP*B3*B2-1.0
      DISCR=B5**2-B6*B4
      IF (DISCR) 2400,2500,2500
 2400 WRITE (6,2410)
 2410 FORMAT (31H1NEGATIVE DISCRIMINANT IN COMBO)
 2415 CALL PDUMP (A(1,1),N,5)
      CALL EXIT
 2500 U(L)=(B5+SQRT(DISCR))/B6
      GO TO 2900
C
C      VARIABLE HEAT CAPACITY
C
 2600 ITER=ITER+1
      B4=2.0*HJ*FH
      DUM1=0.0
      DUM2=0.0
      DO 2650 I=1,NDS
      DO 2650 J=1,NN
      DUM1=DUM1+A(I,J)*C(I,L)*(B3*(B1-B2*US)*US)**(J-1)
```

```
2650 DUM2=DUM2+A(I,J)*C(I,L)*FLOAT(J-1)*(B1*US-B2*US**2)**(J-2)*B3**(J-
    *1)
     G=2.0*HJ*DUM1+US**2-B4
     DGDU=2.0*US+2.0*HJ*DUM2*(B1-2.0*B2*US)
     DELTAU=-G/DGDU
     US=US+DELTAU
     IF (ABS(DELTAU/US)-TOL) 2800,2700,2700
2700 IF (ITER-NITER) 2600,2600,2750
2750 WRITE (6,2755)
2755 FORMAT (27H1ITERATIONS FAILED IN COMBO)
     GO TO 2415
2800 U(L)=US
2900 P(L)=B1-B2*U(L)
     T(L)=B3*P(L)*U(L)
     H(L)=0.0
     DO 2910 I=1,NDS
     DO 2910 J=1,NN
2910 H(L)=H(L)+A(I,J)*C(I,L)*T(L)**(J-1)
     RHO(L)=ZMW*P(L)/(RV*T(L))
     FMAX=1.0
     IF (IEXTRA(1)) 3100,3000,3100
3000 RE(L)=1.E+20
     GO TO 3600
3100 IBIG=1
     MULT=0
     CALL TRANSP (T(L),P(L),L,MULT,IBIG)
     IF (KAY-MU*CP) 3300,3300,3200
3200 FMAX=KAY/(CP*MU)
3300 DO 3500 I=1,NDS
     MAX=I
     DO 3500 J=1,MAX
     IF (DIH(I,J)*RHO(L)/MU-FMAX) 3500,3500,3400
3400 FMAX=DIH(I,J)*RHO(L)/MU
3500 CONTINUE
     DELY(L)=DELY(L)+DELY(K2)
     RE(L)=RHO(L)*U(L)*DELY(L)/(MU*FMAX)
3600 MA(L)=U(L)*SQRT((CP*ZMW-RCON)/(CP*T(L)*RU))
```

```
C
C     REINDEXING STREAMTUBE PROPERTIES
C
      IF (K2-K) 3800,3800,3900
 3800 L=L-1
      GO TO 4000
 3900 PHI(L)=PHI(L+1)
      R(L)=R(L+1)
      SX(L)=SX(L+1)
      X(L)=X(L+1)
      Y(L)=Y(L+1)
      L=L+1
 4000 KMAX=KMAX-1
      IF (L-KMAX) 4100,4100,9000
 4100 DO 4300 L2=L,KMAX
      AA(L2)=AA(L2+1)
      DO 4200 I=1,NDS
      C(I,L2)=C(I,L2+1)
 4200 XS(I,L2)=XS(I,L2+1)
      DELY(L2)=DELY(L2+1)
      H(L2)=H(L2+1)
      MDCT(L2)=MDCT(L2+1)
      P(L2)=P(L2+1)
      PHI(L2)=PHI(L2+1)
      PHI2(L2)=PHI2(L2+1)
      R(L2)=R(L2+1)
      RE(L2)=RE(L2+1)
      RHO(L2)=RHO(L2+1)
      SX(L2)=SX(L2+1)
      T(L2)=T(L2+1)
      U(L2)=U(L2+1)
      X(L2)=X(L2+1)
      Y(L2)=Y(L2+1)
 4300 MA(L2)=MA(L2+1)
 9000 IF (IBUGSH) 9100,10000,9100
 9100 WRITE (6,9200)
 9200 FORMAT (31H1   STREAMTUBES AFTER COMBINING)
```

```
      CALL PUTOUT (1)
10000 RETURN
      END
```

```
      SUBROUTINE FLUX                                                   001
C
C     THIS SUBROUTINE CALCULATES THE FLUXES OF MASS, MOMENTUM, AND HEAT
C     ASSUMING LAMINAR TRANSPORT
C
      COMMON A(10,5)     ,AA(60)       ,ALFA(10,10) ,ALPHAH      ,ALPHAP
     1        ,ATOL        ,BETAP        ,BMIX        ,C11(10)     ,C(10,60)
     2        ,C12(10)     ,C2(10,60)    ,CABAR(10)   ,CBBAR(10)   ,CP
     3        ,CPS(10)     ,CPSH         ,CSH(10)     ,CSH1(10)    ,CSTREM(10)
     4        ,MXSTRM      ,D2IH(10,10) ,DEFF(10)    ,D2EFF(10)   ,DELTA
     5        ,D11         ,DELSS(60)    ,DELS        ,DELSC       ,DLS(60)
     6        ,DIH(10,10)  ,D12          ,DELY(60)    ,D13         ,DPDY(60)
     7        ,D14         ,DPHIDS(60)   ,EPCON
     8        ,EPSLON      ,EXTRA(50)    ,FSTEP       ,FMAX        ,GRAD
     9        ,H11         ,H(60)        ,HH          ,HJ          ,HPM(10)
     *        ,H2PM(10)    ,H3PM(10)     ,ICONST      ,ICOUNT      ,IDENT(10)
     1        ,IERROR      ,IEXTRA(50)   ,IFLAG       ,IKIND
     2        ,IPTUC       ,ISHOCK       ,ITYPE       ,IPD
     3        ,IDIFF       ,K            ,KAY         ,KAYS        ,KAY2
     4        ,KLO         ,KMAX
     5        ,KUP         ,KW           ,LL          ,LPLANE      ,MA
     6        ,MASH        ,MDOT         ,MMAX        ,MUC         ,MU
     7        ,MU2         ,MUS          ,MW          ,MW2         ,MWSH
     8        ,NBOUND      ,NDS          ,NITER       ,NMAX        ,NN
      COMMON NUCASE      ,OMEGA(10)    ,P11         ,P(60)       ,P12
     1        ,P2(60)                    ,PB(50)      ,PABAR       ,PBBAR
     2        ,PBS         ,P15          ,PHI(60)                  ,PHIB(50)
     3        ,PHISH1      ,PHSTRM                    ,PHIW(50)    ,P18
```

```
     4       ,PHI2(60)    ,PHIBS       ,PI          ,PR(10)      ,PSH
     5       ,PSH1        ,PSI         ,PSTREM                   ,PW(50)
     6       ,Q11         ,Q(60)       ,QXTR1       ,QXTR2
     7       ,QW(50)      ,R11         ,R(50)                    ,RB(50)
     8       ,RBS         ,R13         ,RBAR(60)    ,R14         ,R2(60)
     9       ,RCON        ,R15         ,RE(60)      ,RESH        ,R16
     *       ,RHO(60)     ,R17         ,RHO2(60)    ,RHS(10)     ,RHSEN
     1       ,RHSMOM      ,RHABAR      ,RHBBAR      ,RU          ,RSH
     2       ,RSTREM      ,RV          ,R19         ,RW(100)     ,S
     3                    ,SB(50)      ,SC(10)                   ,SW(50)
     4       ,S13         ,SX(60)      ,T11         ,T(60)       ,T12
     5       ,T2(60)      ,TABAR       ,TBBAR       ,TO(10)      ,T14
     6       ,TAW(60)     ,TXTR1       ,TXTR2       ,TCL         ,TSH
     7       ,TSH1        ,TSTREM                   ,TW(50)      ,TWS
     8       ,TS          ,U11         ,U(60)       ,U12         ,U2(60)
      COMMON UXTR1        ,UXTR2       ,U14         ,UBAR(60)    ,USH
     1       ,USH1        ,USTREM                   ,UW(50)      ,UWS
     2       ,X11         ,X(60)       ,X12         ,X2(60)
     3       ,XB(50)      ,XBS         ,XABAR(10)   ,XBBAR(10)   ,X15
     4       ,XS(10,60)   ,XSH         ,XSTREM                   ,XW(50)
     5                                 ,Y11         ,Y(60)       ,Y12
     6       ,Y2(60)      ,YABAR       ,YBBAR       ,ZA(10)      ,Z11(10)
     7       ,ZJ(10,60)   ,ZMW         ,S2X         ,R2SH        ,X2SH
     8       ,FX(2,60)    ,FR(2,60)    ,FPHI(2,60)  ,FP(2,60)    ,FT(2,60)
     9       ,FU(2,60)    ,INDL(2,60)  ,INDR(2,60)  ,FC(2,60,10),CARD1
     *       ,M           ,N
      REAL    KAY         ,KAYS(10)    ,KAY2        ,KW          ,MDOT(60)
     *       ,MA(60)      ,MU          ,MUS(10)     ,MUO(10)     ,MU2
     *       ,MW(10)      ,MW2         ,MASH        ,MASH1       ,MWSH
      COMMON/INPUX/ ITD,X0,Y0,FRAC
      COMMON/TRANUX/DT(10),D2T(10)
      DIMENSION CDENOM(10), PDENOM(10), TDENOM(10)
      DIMENSION DENOM (10,10)
      DIMENSION BETA(10)
      DIMENSION IDROP(10),ZZJJ(10)
      IF (NBOUND) 10,30,10
   10 Q(K)=0.0
```

```
      TAW(K)=0.0
      DO 20 I=1,NDS
   20 ZJ(I,K)=0.0
      GO TO 10000
C
C     SETUP OF AVERAGED FLOW QUANTITIES NEEDED
C
   30 YCBAR=YBBAR
      YBBAR=YABAR
      TBBAR=TABAR
      RHBBAR=RHABAR
      DO 50 I=1,NDS
      CBBAR(I)=CABAR(I)
   50 XBBAR(I)=XABAR(I)
      PBBAR=PABAR
      NOVER=0
      IF (IFLAG) 100,400,100
  100 YABAR=.5*(Y(K+1)+Y2(K+1))
      TABAR=.5*(T(K+1)+T2(K+1))
      KU=K+1
      UBAR(KU)=SQRT(.5*(U(K+1)**2+U2(K+1)**2))
      RHABAR=.5*(RHO(K+1)+RHO2(K+1))
  110 RBVG=0.25*(R(K)+R(K-1)+R2(K)+R2(K-1))
      RAVG=0.25*(R(K+1)+R(K)+R2(K+1)+R2(K))
  130 ZMW2=0.0
      DO 200 I=1,NDS
      CABAR(I)=.5*(C(I,K+1)+C2(I,K+1))
  200 ZMW2=ZMW2+CABAR(I)/MW(I)
      ZMW2=1.0/ZMW2
      DO 300 I=1,NDS
  300 XABAR(I)=CABAR(I)*ZMW2/MW(I)
      PABAR=.5*(P(K+1)+P2(K+1))
      GO TO 600
  400 TABAR=T(K+1)
      KU=K+1
      UBAR(KU)=U(K+1)
      RHABAR=RHO(K+1)
```

```
  120 RBVG=0.5*(R(K)+R(K-1))                                              044
      RAVG=0.5*(R(K+1)+R(K))                                              045
      DO 500 I=1,NDS                                                      046
      CABAR(I)=C(I,K+1)                                                   047
  500 XABAR(I)=XS(I,K+1)                                                  048
      PABAR=P(K+1)                                                        049
  600 DO 602 I=1,NDS                                                      050
      IF (XABAR(I)-0.75) 602,601,601                                      051
  601 IF (XBBAR(I)-0.75) 603,604,604                                      052
  602 CONTINUE                                                            053
C
C     TRANSPORT PROPERTIES
C
  603 MULT=0                                                              054
      GO TO 605                                                           055
  604 MULT=1                                                              056
      IBIG=I                                                              057
  605 CALL TRANSP(TABAR,PABAR,K+1,MULT,IBIG)                              058
C
C     DETERMINING SMOOTH PRESSURE GRADIENT FOR PRESSURE DIFFUSION
C
      IF (K-2) 606,606,620                                                059
  606 SUMDOT=0.0                                                          060
      DO 610 MM=1,KMAX                                                    061
  610 SUMDOT=SUMDOT+MDOT(MM)                                              062
      DPDMAV=(P(KMAX)-P(1))/(SUMDOT-0.5*(MDOT(1)+MDOT(KMAX)))             063
      MM=0                                                                064
  620 DPDM=(0.5*(R(MM)+R(KMAX))/R(K))**DELTA*DPDMAV                       065
C
C     CALCULATION OF HEAT FLUX AND SHEAR
C
      DENOMU=(MDOT(K+1)+MDOT(K))/(RHABAR*UBAR(K+1)*  RAVG **DELTA*MU+     066
     *RHBBAR*UBAR(K)*  RBVG **DELTA*MU2)/(2.0**DELTA*PI)                  067
      DENOMT=(MDOT(K+1)+MDOT(K))/(RHABAR*UBAR(K+1)*RAVG     **DELTA*KAY   068
     *+RHBBAR*UBAR(K)* RBVG  **DELTA*KAY2)/(2.0**DELTA*PI)                069
      TAW(K)=(UBAR(K+1)-UBAR(K))/DENOMU                                   070
      Q(K)=(TABAR-TBBAR)/DENOMT                                           071
```

```
  625 IF (MULT) 1880,630,1880
C
C     SPECIES FLUX CALCULATION--MULTICOMPONENT DIFFUSION
C
  630 DO 666 I=1,NDS
      DO 666 J=1,NDS
  666 DENOM(I,J)=(RHABAR**2*DIH(I,J)*UBAR(K+1)*  RAVG   **DELTA+RHBBAR**
     *2*D2IH(I,J)*UBAR(K)*  RBVG **DELTA)*2.0**DELTA*PI
  700 NUSE = NDS
      DO 850 I=1,NDS
      ZJ(I,K)=0.0
      IDROP(I)=0
      IF (XBBAR(I)) 850,750,850
  750 IF (XABAR(I)) 850,800,850
  800 IDROP (I)=1
      NUSE=NUSE-1
  850 CONTINUE
  900 I=1
      II=1
      NOVER=0
  925 IF(IDROP(I)) 1550,950,1550
C
C     DRIVING FORCES
C
  950 BETA(II)=-(XABAR(I)-XBBAR(I))/(MDOT(K+1)+MDOT(K))*2.0
  960 F1=BETA(II)
  990 IF (IPD) 1000,1003,1000
 1000 BETA(II)=BETA(II)-((XABAR(I)-CABAR(I))+(XBBAR(I)-CBBAR(I)))*DPDM/(
     *0.5*(PABAR+PBBAR))
 1002 F2=BETA(II)-F1
 1003 IF (ITD) 1005,1050,1005
 1005 IF (I-1) 1012,1007,1012
 1007 DO 1010 J=2,NDS
 1010 BETA(II)=BETA(II)+(XABAR(I)+XBBAR(I))*(XABAR(J)+XBBAR(J))*(ALOG(
     *TABAR)-ALOG(TBBAR))/(MDOT(K+1)+MDOT(K))*((DT(J)+D2T(J))/(CABAR(J)+
     *CBBAR(J))-(DT(I)+D2T(I))/(CABAR(I)+CBBAR(I)))/(DIH(I,J)+D2IH(I,J))
      GO TO 1014
```

```
 1012 BETA(II)=BETA(II)+(XABAR(I)+XBBAR(I))*(XABAR(1)+XBBAR(1))*(ALOG(
     *TABAR)-ALOG(TBBAR))/(MDOT(K+1)+MDOT(K))*((DT(1)+D2T(1))/(CABAR(1)+
     *CBBAR(1))-(DT(I)+D2T(I))/(CABAR(I)+CBBAR(I)))/(DIH(I,1)+D2IH(I,1))
 1014 F3=BETA(II)-F1-F2
 1030 IF(I-1) 1050,1035,1050
 1035 INTGER=ALPHAP
      IF (MOD(LL,INTGER)) 1050,1040,1050
 1040 WRITE (6,1045) K,F1,F2,F3
 1045 FORMAT (10X,I2,3X,1P1E11.4,3X,1P1E11.4,3X,1P1E11.4)
C
C     MATRIX ELEMENTS
C
 1050 ALFA (II,II) = 0.0
      DO 1200 J=1,NDS
      IF (IDROP (J)) 1200,1100,1200
 1100 IF (J-I) 1150,1200,1150
 1150 ALFA(II,II)=ALFA(II,II)-(XABAR(I)+XBBAR(I))*(XABAR(J)+XBBAR(J))/(
     *DENOM(I,J)*(CABAR(I)+CBBAR(I)))
 1200 CONTINUE
      IF (II-2) 1500,1250,1250
C
 1250 J=1
      JJ=1
 1300 IF(IDROP(J))1450,1350,1450
 1350 ALFA(II,JJ)=(XABAR(I)+XBBAR(I))*(XABAR(J)+XBBAR(J))/(DENOM(I,J)*
     *(CABAR(J)+CBBAR(J)))
      ALFA(JJ,II)=ALFA(II,JJ)*(CABAR(J)+CBBAR(J))/(CABAR(I)+CBBAR(I))
      IF(JJ-II+1) 1400,1500,1500
 1400 JJ=JJ+1
 1450 J=J+1
      GO TO 1300
 1500 II=II+1
 1550 I=I+1
      IF (II-NUSE+1) 925,925,1600
 1600 L=0
      IF (NUSE) 1630,1630,1602
 1602 DO 1610  JJ=1,NUSE
```

```
 1604 L=L+1
      IF(I-L) 1605,1610,1605
 1605 IF (IDROP(L)) 1604,1606,1604
 1606 ALFA(JJ,NUSE)=(XABAR(I)+XBBAR(I))*(XABAR(L)+XBBAR(L))/(DENOM(I,L)*
     *(CABAR(I)+CBBAR(I)))
 1610 ALFA(NUSE,JJ)=1.0
      BETA(NUSE)=0.0
 1625 IF (II-1) 1630,1650,1650
 1630 DO 1640 I=1,NDS
 1640 ZJ(I,K)=0.0
      GO TO 10000
 1650 YYY=0.0
 1652 CALL MATE7(10,NUSE,1,ALFA,BETA,YYV,ZZJJ,J)
      GO TO (1700,1655,1675),J
 1655 WRITE (6,1660)
 1660 FORMAT(40H1DETERMINANT OVERFLOW IN SUBROUTINE FLUX)
      GO TO 1690
 1675 WRITE (6,1680)
 1680 FORMAT (40H1SINGULAR DETERMINANT IN SUBROUTINE FLUX)
 1690 CALL DUMP(A(1,1),N,5)
 1700 DO 1750 II = 1,NUSE
 1750 ZZJJ(II)=ALFA(II,1)
      II=1
 1800 DO 1850 I=1,NDS
      IF (IDROP(I)) 1850,1825,1850
 1825 ZJ(I,K) = ZZJJ(II)
 1840 II=II+1
 1850 CONTINUE
C
C     RECHECK TO SEE THAT MASS FRACTIONS STAY POSITIVE
C
      DO1875 I=1,NDS
      IF (XABAR(I)) 1855,1855,1865
 1855 IF (ZJ(I,K)) 1865,1865,1860
 1860 IDROP(I) =1
      ZJ(I,K)=0.0
      NUSE=NUSE-1
```

```
      NOVER=1
      GO TO 1875
 1865 IF (XBBAR(I)) 1870,1870,1875
 1870 IF(ZJ(I,K)) 1860,1875,1875
 1875 CONTINUE
      IF (NOVER) 900,1878,900
 1878 GO TO 1925
 1880 DO 1890 I=1,NDS
      IF (IBIG-I) 1885,1890,1885
 1885 CDENOM(I)=(MDOT(K)+MDOT(K+1))/(RHBBAR**2*D2EFF(I)*UBAR(K)*PI*(2.0*
     *RBVG)**DELTA+RHABAR**2*DEFF(I)*UBAR(K+1)*PI*(2.0*RAVG)**DELTA)
      ZJX=(CABAR(I)-CBBAR(I))/CDENOM(I)
      PDENOM(I)=CDENOM(I)*ZMW**2/(0.5*(XABAR(I)-CABAR(I)+XBBAR(I)-CBBAR
     *(I))*MW(I)*MW(IBIG)*(.5*(MDOT(K)+MDOT(K+1))))
      ZJP=DPDM/(0.5*(PABAR+PBBAR)*PDENOM(I))
      IF (D2T(I)) 1886,2000,1886
 2000 IF (DT(I)) 1886,2001,1886
 1886 TDENOM(I)=(MDOT(K)+MDOT(K+1))/(RHBBAR**2*D2T(I)*UBAR(K)*PI*(2.0*RB
     *VG)**DELTA+RHABAR**2*DT(I)*UBAR(K+1)*PI*(2.0*RAVG)**DELTA)
      ZJT=(ALOG(TBBAR)-ALOG(TABAR))/TDENOM(I)
      GO TO 2002
 2001 ZJT=0.0
 2002 ZJ(I,K)=ZJX+ZJP+ZJT
      IF (I-1) 1890,1887,1890
 1887 INTGER=ALPHAP
      IF (MOD(LL,INTGER))   1890,1888,1890
 1888 WRITE (6,1045) K,ZJX,ZJP,ZJT
 1890 CONTINUE
      ZJ(IBIG,K)=0.0
      DO 1900 I=1,NDS
      IF (IBIG-I) 1895,1900,1895
 1895 ZJ(IBIG,K)=ZJ(IBIG,K)-ZJ(I,K)
 1900 CONTINUE
 1925 L=0
      DIST=X(L)-X0
      IF(ABS(DIST)-FRAC*Y0) 1830,10000,10000
 1830 FLIMIT=ABS(DIST)/(FRAC*Y0)
```

```
      Q(K)=Q(K)*FLIMIT
      TAW(K)=TAW(K)*FLIMIT
      DO 1835 I=1,NDS
 1835 ZJ(I,K)=ZJ(I,K)*FLIMIT
      GO TO 10000
10000 RETURN
      END
```

```
      SUBROUTINE FLUX
C
C     THIS SUBROUTINE CALCULATES THE FLUXES OF MASS, MOMENTUM, AND HEAT
C     ASSUMING TURBULENT TRANSPORT
C
      COMMON A(10,5)      ,AA(60)       ,ALFA(10,10),ALPHAH    ,ALPHAP
     1       ,ATOL        ,BETAP        ,BMIX       ,C11(10)   ,C(10,60)
     2       ,C12(10)     ,C2(10,60)    ,CABAR(10)  ,CBBAR(10) ,CP
     3       ,CPS(10)     ,CPSH         ,CSH(10)    ,CSH1(10)  ,CSTREM(10)
     4       ,MXSTRM      ,D2IH(10,10)  ,DEFF(10)   ,D2EFF(10) ,DELTA
     5       ,D11         ,DELSS(60)    ,DELS       ,DELSO     ,DLS(60)
     6       ,DIH(10,10)  ,D12          ,DELY(60)   ,D13       ,DPDY(60)
     7       ,D14         ,DPHIDS(60)   ,EPCON
     8       ,EPSLON      ,EXTRA(50)    ,FSTEP      ,FMAX      ,GRAD
     9       ,H11         ,H(60)        ,HH         ,HJ        ,HPM(10)
     *       ,H2PM(10)    ,H3PM(10)     ,ICONST     ,ICOUNT    ,IDENT(10)
     1       ,IERROR      ,IEXTRA(50)   ,IFLAG      ,IKIND
     2       ,IPTUC       ,ISHOCK       ,ITYPE      ,IPD
     3       ,IDIFF       ,K            ,KAY        ,KAYS      ,KAY2
     4       ,KLO         ,KMAX
     5       ,KUP         ,KW           ,LL         ,LPLANE    ,MA
     6       ,MASH        ,MDOT         ,MMAX       ,MUO       ,MU
     7       ,MU2         ,MUS          ,MW         ,MW2       ,MWSH
     8       ,NBOUND      ,NDS          ,NITER      ,NMAX      ,NN
```

```
 COMMON NUCASE         ,OMEGA(10)      ,P11            ,P(60)          ,P12
1         ,P2(60)                      ,PE(50)         ,PABAR          ,PBBAR
2         ,PBS         ,P15            ,PHI(60)                        ,PHIB(50)
3         ,PHISH1      ,PHSTRM                         ,PHIW(50)       ,P18
4         ,PHI2(60)    ,PHIBS          ,PI             ,PK(10)         ,PSH
5         ,PSH1        ,PSI            ,PSTREM                         ,PK(50)
6         ,Q11         ,Q(60)          ,QXTR1          ,QXTR2
7         ,QW(50)      ,R11            ,R(60)                          ,RB(50)
8         ,RBS         ,R13            ,RBAR(60)       ,R14            ,R2(60)
9         ,RCON        ,R15            ,RE(60)         ,RESH           ,R16
*         ,RHO(60)     ,R17            ,RHO2(60)       ,RHS(10)        ,RHSEN
1         ,RHSMOM      ,RHABAR         ,RHBBAR         ,RU             ,RSH
2         ,RSTREM      ,RV             ,R19            ,RW(100)        ,S
3                      ,SB(50)         ,SC(10)                         ,SW(50)
4         ,S13         ,SX(60)         ,T11            ,T(60)          ,T12
5         ,T2(60)      ,TABAR          ,TBBAR          ,TC(10)         ,T14
6         ,TAW(60)     ,TXTR1          ,TXTR2          ,TOL            ,TSH
7         ,TSH1        ,TSTREM                         ,TW(50)         ,TWS
8         ,TS          ,U11            ,U(60)          ,U12            ,U2(60)
 COMMON UXTR1          ,UXTR2          ,U14            ,UBAR(60)       ,USH
1         ,USH1        ,USTREM                         ,UW(50)         ,UWS
2         ,X11         ,X(60)          ,X12            ,X2(60)
3         ,XB(50)      ,XBS            ,XABAR(10)      ,XBBAR(10)      ,X15
4         ,XS(10,60)   ,XSH            ,XSTREM                         ,XW(50)
5                                      ,Y11            ,Y(60)          ,Y12
6         ,Y2(60)      ,YABAR          ,YBBAR          ,ZA(10)         ,Z11(10)
7         ,ZJ(10,60)   ,ZMW            ,S2X            ,R2SH           ,X2SH
8         ,FX(2,60)    ,FR(2,60)       ,FPHI(2,60)     ,FP(2,60)       ,FT(2,60)
9         ,FU(2,60)    ,INDL(2,60)     ,INDR(2,60)     ,FC(2,60,10),CARD1
*         ,M           ,N
 REAL      KAY         ,KAYS(10)       ,KAY2           ,KW             ,MDOT(60)
*         ,MA(60)      ,MU             ,MUS(10)        ,MUC(10)        ,MU2
*         ,MW(10)      ,MW2            ,MASH           ,MASH1          ,MWSH
 COMMON/INPUX/ ITD,XO,YO,FRAC                                                   002
 COMMON/TRANUX/DT(10),D2T(10)                                                   003
 DIMENSION CDENOM(10), PDENOM(10), TDENOM(10)                                   004
 DIMENSION DENOM (10,10)                                                        005
```

```
      DIMENSION BETA(10)
      DIMENSION IDROP(10),ZZJJ(10)
      IF (NBCUND) 10,30,10
   10 Q(K)=0.0
      TAW(K)=0.0
      DO 20 I=1,NDS
   20 ZJ(I,K)=0.0
      GO TO 10000
C
C     SETUP OF AVERAGED FLOW QUANTITIES NEEDED
C
   30 YCBAR=YBBAR
      YBBAR=YABAR
      TBBAR=TABAR
      RHBBAR=RHABAR
      DO 50 I=1,NDS
      CBBAR(I)=CABAR(I)
   50 XBBAR(I)=XABAR(I)
      PBBAR=PABAR
      NOVER=0
      IF (IFLAG) 100,400,100
  100 YABAR=.5*(Y(K+1)+Y2(K+1))
      TABAR=.5*(T(K+1)+T2(K+1))
      KU=K+1
      UBAR(KU)=SQRT(.5*(U(K+1)**2+U2(K+1)**2))
      RHABAR=.5*(RHO(K+1)+RHO2(K+1))
  110 RBVG=0.25*(R(K)+R(K-1)+R2(K)+R2(K-1))
      RAVG=0.25*(R(K+1)+R(K)+R2(K+1)+R2(K))
  130 ZMW2=0.0
      DO 200 I=1,NDS
      CABAR(I)=.5*(C(I,K+1)+C2(I,K+1))
  200 ZMW2=ZMW2+CABAR(I)/MW(I)
      ZMW2=1.0/ZMW2
      DO 300 I=1,NDS
  300 XABAR(I)=CABAR(I)*ZMW2/MW(I)
      PABAR=.5*(P(K+1)+P2(K+1))
      GO TO 605
```

```
  400 TABAR=T(K+1)                                                       040
      KU=K+1                                                             041
      UBAR(KU)=U(K+1)                                                    042
      RHABAR=RHO(K+1)                                                    043
  120 RBVG=0.5*(R(K)+R(K-1))                                             044
      RAVG=0.5*(R(K+1)+R(K))                                             045
      DO 500 I=1,NDS                                                     046
      CABAR(I)=C(I,K+1)                                                  047
  500 XABAR(I)=XS(I,K+1)                                                 048
      PABAR=P(K+1)                                                       049
  605 CALL TRANSP(TABAR,PABAR,K+1,MULT,IBIG)                             058
C
C     CALCULATION OF HEAT FLUX AND SHEAR
C
      DENOMU=(MDOT(K+1)+MDOT(K))/(RHABAR*UBAR(K+1)*  RAVG **DELTA*MU+   066
     *RHBBAR*UBAR(K)*  RBVG **DELTA*MU2)/(2.0**DELTA*PI)                 067
      DENOMT=(MDOT(K+1)+MDOT(K))/(RHABAR*UBAR(K+1)*RAVG     **DELTA*KAY  068
     *+RHBBAR*UBAR(K)* RBVG  **DELTA*KAY2)/(2.0**DELTA*PI)               069
      TAW(K)=(UBAR(K+1)-UBAR(K))/DENOMU                                  070
      Q(K)=(TABAR-TBBAR)/DENOMT                                          071
C
C     CALCULATION OF MASS FLUX
C
      DO 1900 I=1,NDS
      CDENOM(I)=(MDOT(K)+MDOT(K+1))/(RHBBAR**2*D2IH(1,2)*UBAR(K)*PI*(2.0
     * *RBVG)**DELTA+RHABAR**2*DIH(1,2)*UBAR(K+1)*PI*(2.0*RAVG)**DELTA)
 1900 ZJ(I,K)=(CABAR(I)-CBBAR(I))/CDENOM(I)
 1925 L=0                                                                202
      DIST=X(L)-X0                                                       203
      IF(ABS(DIST)-FRAC*Y0) 1830,10000,10000                             204
 1830 FLIMIT=ABS(DIST)/(FRAC*Y0)                                         205
      Q(K)=Q(K)*FLIMIT                                                   206
      TAW(K)=TAW(K)*FLIMIT                                               207
      DO 1835 I=1,NDS                                                    208
 1835 ZJ(I,K)=ZJ(I,K)*FLIMIT                                             209
10000 RETURN                                                             211
      END                                                                212
```

```
      SUBROUTINE LOCATE(XB,YB,X1,Y1,X2,Y2,X3,Y3,IFGOOD)
C
C     THIS SUBROUTINE LOCATES THE UPSTREAM POSITION OF A SHOCK POINT
C     BETWEEN TWO STREAMTUBES FROM A PREVIOUSLY CALCULATED NONUNIFORM
C     FLOWFIELD
C
      DIMENSION XLOC(3)
C     IFGOOD = 1  POINT IS BOUNDED
C            = 2  POINT IS NOT BOUNDED
      DO 22   I=1,3
      GO TO (10,12,14) , I
   10 XX2 = X1
      YY2 = Y1
      GO TO 16
   12 XX2 = X2
      YY2 = Y2
      GO TO 16
   14 XX2 = X3
      YY2 = Y3
   16 D = SQRT((XX2-XB)**2+(YY2-YB)**2)
      COST =(XX2-XB)/D
      IF (YY2.GE.YB) GO TO 20
      XLOC(I) = 3.+COST
      GO TO 22
   20 XLOC(I) = 1.-COST
   22 CONTINUE
      IF (XLOC(3).LT.XLOC(2)) GO TO 55
      IF (XLOC(1).GE.XLOC(2).AND.XLOC(1).LE.XLOC(3)) GO TO 50
   40 IFGOOD = 2
      GO TO 10000
   50 IFGOOD = 1
```

```
      GO TO 10000
   55 IF (XLOC(1).GE.XLOC(2).OR.XLOC(1).LE.XLOC(3)) GO TO 50
      GO TO 40
10000 RETURN
      END
```

```
      SUBROUTINE MATE7(IDIM,NN,MM,A,B,DET,IND,NOGO)
C
C     THIS SUBROUTINE SOLVES THE EQUATION AX=B WHERE A,X, AND B ARE
C     MATRICES
C
      DIMENSION A(IDIM,1),B(IDIM,1),IND(1)
C         SET INITIAL CONSTANTS
      N=NN
      M=MM
      NOGO=1
C         SPECIAL CONSIDERATION WHEN THE ORDER OF MATRIX A IS 1.
      IF(N.NE.1) GO TO 3
      IF(A(1,1).EQ.0.) GO TO 110
      HOLD=A(1,1)
      DO 1 I=1,M
    1 A(1,I)=B(1,I)/HOLD
      DET=DET*HOLD
      RETURN
    3 NM1=N-1
      NP1=N+1
C         INITIALIZE DETERMINANT
      DETM=DET
C         INITIALIZE COLUMN INDICATORS
      DO 5 I=1,N
    5 IND(I)=I
C         BEGIN TRIANGULARIZATION TO GET UPPER TRIANGLE
```

```
      DO 70 K=1,NM1
      KP1=K+1
      KR=K
      KC=K
C        SEARCH FOR PIVOTAL ELEMENT
      BIGA=ABS(A(K,K))
      DO 10 I=K,N
      DO 10 J=K,N
      IF(BIGA.GE.ABS(A(I,J))) GO TO 10
      BIGA=ABS(A(I,J))
      KR=I
      KC=J
   10 CONTINUE
C        TEST FOR SINGULAR MATRIX
      IF(BIGA.EQ.0.) GO TO 110
C        UPDATE DETERMINANT
      DETM=DETM*A(KR,KC)
C        INTERCHANGE ROWS
      IF(KR.EQ.K) GO TO 30
      DO 20 I=K,N
      HOLD=A(K,I)
      A(K,I)=A(KR,I)
   20 A(KR,I)=HOLD
C        INTERCHANGE ELEMENTS OF RIGHT HAND SIDES
      DO 25 L=1,M
      HOLD=B(K,L)
      B(K,L)=B(KR,L)
   25 B(KR,L)=HOLD
C        CHANGE SIGN OF DETERMINANT DUE TO ROW INTERCHANGE
      DETM=-DETM
C        INTERCHANGE COLUMNS
   30 IF(KC.EQ.K) GO TO 55
      DO 40 J=1,N
      HOLD=A(J,K)
      A(J,K)=A(J,KC)
   40 A(J,KC)=HOLD
C        INTERCHANGE COLUMN INDICATORS
```

```
      I=IND(K)
      IND(K)=IND(KC)
      IND(KC)=I
C        CHANGE SIGN OF DETERMINANT DUE TO COLUMN INTERCHANGE
      DETM=-DETM
C        DIVIDE REDUCED EQUATION-ON, BY LEADING ELEMENT
   55 DO 60 I=KP1,N
   60 A(K,I)=A(K,I)/A(K,K)
      DO 63 L=1,M
   63 B(K,L)=B(K,L)/A(K,K)
C        REDUCE MATRIX AND RIGHT HAND SIDES
      DO 70 I=KP1,N
      DO 65 J=KP1,N
   65 A(I,J)=A(I,J)-A(I,K)*A(K,J)
      DO 70 L=1,M
   70 B(I,L)=B(I,L)-A(I,K)*B(K,L)
C        FINAL TEST FOR SINGULAR MATRIX
      IF(A(N,N).EQ.0.) GO TO 110
C        COMPUTE FINAL DETERMINANT
      DET=DETM*A(N,N)
C        BACK SUBSTITUE TO OBTAIN SOLUTION VECTORS
      DO 80 L=1,M
      B(N,L)=B(N,L)/A(N,N)
      DO 80 I=1,NM1
      HOLD=0.
      J=N-I
      DO 75 KC=1,I
      K=NP1-KC
   75 HOLD=HOLD+A(J,K)*B(K,L)
   80 B(J,L)=B(J,L)-HOLD
C        REARRANGE SOLUTION VECTORS TO ORIGINAL ORDER
      DO 90 I=1,N
      J=IND(I)
      DO 90 L=1,M
   90 A(J,L)=B(I,L)
      RETURN
C        SINGULAR MATRIX - REDUNDANT SET OF EQUATIONS
```

```
  110 NOGO=3
      DET=0.
      RETURN
      END
```

```
      SUBROUTINE ORTHOG(SIDE,LIM1,LIM2)                                 C01
C
C     THIS SUBROUTINE CONSTRUCTS NORMALS FROM THE UPSTREAM SHOCK
C     POINT TO THE NEAREST ORTHOGONAL SURFACES FROM A PREVIOUSLY
C     CALCULATED NONUNIFORM FLOWFIELD.
C
      COMMON A(10,5)      ,AA(60)     ,ALFA(10,10),ALPHAH     ,ALPHAP
     1          ,ATOL     ,BETAP      ,BMIX       ,C11(10)    ,C(10,60)
     2          ,C12(10)  ,C2(10,60)  ,CABAR(10)  ,CBBAR(10)  ,CP
     3          ,CPS(10)  ,CPSH       ,CSH(10)    ,CSH1(10)   ,CSTREM(10)
     4          ,MXSTRM   ,D2IH(10,10),DEFF(10)   ,D2EFF(10)  ,DELTA
     5          ,D11      ,DELSS(60)  ,DELS       ,DELSO      ,DLS(60)
     6          ,DIH(10,10) ,D12      ,DELY(60)   ,D13        ,DPDY(60)
     7          ,D14      ,DPHIDS(60) ,EPCON
     8          ,EPSLON   ,EXTRA(50)  ,FSTEP      ,FMAX       ,GRAD
     9          ,H11      ,H(60)      ,HH         ,HJ         ,HPM(10)
     *          ,H2PM(10) ,H3PM(10)   ,ICONST     ,ICOUNT     ,IDENT(10)
     1          ,IERROR   ,IEXTRA(50) ,IFLAG      ,IKIND
     2          ,IPTUC    ,ISHOCK     ,ITYPE      ,IPD
     3          ,IDIFF    ,K          ,KAY        ,KAYS       ,KAY2
     4          ,KLO      ,KMAX
     5          ,KUP      ,KW         ,LL         ,LPLANE     ,MA
     6          ,MASH     ,MDOT       ,MMAX       ,MUO        ,MU
     7          ,MU2      ,MUS        ,MW         ,MW2        ,MWSH
     8          ,NBOUND   ,NDS        ,NITER      ,NMAX       ,NN
     COMMON NUCASE        ,OMEGA(10)  ,P11        ,P(60)      ,P12
     1          ,P2(60)               ,PB(50)     ,PABAR      ,PBBAR
```

```
2        ,PBS          ,P15          ,PHI(60)                    ,PHIB(50)
3        ,PHISH1       ,PHSTRM                     ,PHIW(50)     ,PIS
4        ,PHI2(60)     ,PHIBS        ,PI           ,PR(10)       ,PSH
5        ,PSH1         ,PSI          ,PSTREM                     ,PW(50)
6        ,Q11          ,Q(60)        ,QXTR1        ,QXTR2
7        ,QW(50)       ,R11          ,R(60)                      ,R1(50)
8        ,RBS          ,R13          ,RBAR(60)     ,R14          ,R2(60)
9        ,RCON         ,R15          ,RE(60)       ,RESH         ,R16
*        ,RHO(60)      ,R17          ,RHO2(60)     ,RHS(10)      ,RHSEN
1        ,RHSMOM       ,RHABAR       ,RHBBAR       ,RU           ,RSH
2        ,RSTREM       ,RV           ,R19          ,RW(100)      ,S
3                      ,SB(50)       ,SC(10)                     ,SW(50)
4        ,S13          ,SX(60)       ,T11          ,T(60)        ,T12
5        ,T2(60)       ,TABAR        ,TBBAR        ,TJ(10)       ,T14
6        ,TAW(60)      ,TXTR1        ,TXTR2        ,TOL          ,TSH
7        ,TSH1         ,TSTREM                     ,TW(50)       ,TWS
8        ,TS           ,U11          ,U(60)        ,U12          ,U2(60)
 COMMON  UXTR1         ,UXTR2        ,U14          ,UBAR(60)     ,USH
1        ,USH1         ,USTREM                     ,UW(50)       ,UWS
2        ,X11          ,X(60)        ,X12          ,X2(60)
3        ,XB(50)       ,XBS          ,XABAR(10)    ,XBBAR(10)    ,X15
4        ,XS(10,60)    ,XSH          ,XSTREM                     ,XW(50)
5                                    ,Y11          ,Y(60)        ,Y12
6        ,Y2(60)       ,YABAR        ,YBBAR        ,ZA(10)       ,Z11(10)
7        ,ZJ(10,60)    ,ZMW          ,S2X          ,R2SH         ,X2SH
8        ,FX(2,60)     ,FR(2,60)     ,FPHI(2,60)   ,FP(2,60)     ,FT(2,60)
9        ,FU(2,60)     ,INDL(2,60)   ,INDR(2,60)   ,FC(2,60,10),CARD1
*        ,M            ,N
 REAL    KAY           ,KAYS(10)     ,KAY2         ,KW           ,MDOT(60)
1        ,MA(60)       ,MU           ,MUS(10)      ,MUG(10)      ,MU2
2        ,MW(10)       ,MW2          ,MASH         ,MASH1        ,MWSH
 INTEGER SIDE,DIRCTN,PROP1,PROP2
 COMMON /INTRP1/ NSTRM(2),TSFC(2),PSFC(2),USFC(2),PHISFC(2),
1CSFC(2,10),DPTSFC(2)
 COMMON /IBUGS1/ IBUGSH
 COMMON/CRSHCK/XTRY,RTRY
 AX= FX(SIDE,LIM2)-FX(SIDE,LIM1)
```

```
      BX= FR(SIDE,LIM2)-FR(SIDE,LIM1)
      CX=AX*XTRY+BX*RTRY
      IF(AX.EQ.0.) GO TO 20
      TEMP=-BX/AX*FX(SIDE,LIM1)+FR(SIDE,LIM1)
      TEMP1 = AX+BX*BX/AX
      XINT=(CX-BX*TEMP)/TEMP1
      RINT = TEMP+BX/AX*XINT
      GO TO 30
   20 XINT = FX(SIDE,LIM1)
      RINT=(CX-AX*XINT)/BX
   30 DPTSFC(SIDE) = SQRT((XINT-XTRY)**2+(RINT-RTRY)**2)
      D1SQ = (XINT-FX(SIDE,LIM1))**2+(RINT-FR(SIDE,LIM1))**2
      XMID1=(FX(SIDE,LIM1)+FX(SIDE,LIM2))/2.
      RMID1=(FR(SIDE,LIM1)+FR(SIDE,LIM2))/2.
      PROP1 = LIM2
      D2SQ = (XMID1-FX(SIDE,LIM1))**2+(RMID1-FR(SIDE,LIM1))**2
      DIRCTN = 1
      IF (D2SQ.LT.D1SQ) DIRCTN = 2
      GO TO ( 40 , 47 ), DIRCTN
   40 J = LIM1
   42 IF (J.EQ.1) GO TO 54
      J = J-1
   44 XMID2=(FX(SIDE,LIM1)+FX(SIDE,J))/2.
      RMID2=(FR(SIDE,LIM1)+FR(SIDE,J))/2.
      PROP2 = LIM1
      GO TO 56
   47 J = LIM2
   49 IF (J.EQ.NSTRM(SIDE)) GO TO 54
      J = J+1
   52 XMID2 = (FX(SIDE,LIM2)+FX(SIDE,J))/2.
      RMID2 = (FR(SIDE,LIM2)+FR(SIDE,J))/2.
      PROP2 = J
      GO TO 56
   54 PROP2 = PROP1
      GO TO 57
   56 DPTMD1 = SQRT((XINT-XMID1)**2+(RINT-RMID1)**2)
      DMD12 = SQRT((XMID1-XMID2)**2+(RMID1-RMID2)**2)
```

```
      PRPRAT = DPTMD1/DMD12
   57 TSFC(SIDE) = FT(SIDE,PROP1)+PRPRAT*(FT(SIDE,PROP2)-FT(SIDE,PROP1))
      PSFC(SIDE) = FP(SIDE,PROP1)+PRPRAT*(FP(SIDE,PROP2)-FP(SIDE,PROP1))
      USFC(SIDE) = FU(SIDE,PROP1)+PRPRAT*(FU(SIDE,PROP2)-FU(SIDE,PROP1))
      DO 59   I=1,NCS
   59 CSFC(SIDE,I) = FC(SIDE,PROP1,I)+PRPRAT*(FC(SIDE,PROP2,I)-FC(SIDE,
     1PROP1,I))
      D3SQ = (FX(SIDE,LIM2)-FX(SIDE,LIM1))**2+(FR(SIDE,LIM2)-FR(SIDE,
     1LIM1))**2
      PHISFC(SIDE) = FPHI(SIDE,LIM1)+(FPHI(SIDE,LIM2)-FPHI(SIDE,LIM1))*
     1SQRT(D1SQ/D3SQ)
      IF (IBUGSH.EQ.0) GO TO 10000
      WRITE (6, 62) AX,BX,CX,TEMP,TEMP1,XINT,RINT,DPTSFC(SIDE),D1SQ,XMID
     11,RMID1,PROP1,D2SQ,DIRCTN,XMID2,RMID2,PROP2,DPTMD1,DMD12,PRPRAT,
     2TSFC(SIDE),PSFC(SIDE),USFC(SIDE),D3SQ,PHISFC(SIDE),SIDE,LIM1,LIM2
   62 FORMAT (/20H RESULTS FROM ORTHOG/4H AX=1PE15.6,4H BX=E15.6,4H CX=E
     115.6,6H TEMP=E15.6,7H TEMP1=E15.6/6H XINT=E15.6,6H RINT=E15.6,8H D
     2PTSFC=E15.6,6H D1SQ=E15.6,7H XMID1=E15.6/7H RMID1=E15.6,7H PROP1=I
     33,6H D2SQ=E15.6,8H DIRCTN=I3,7H XMID2=E15.6,7H RMID2=E15.6,7H PROP
     42=I3/8H DPTMD1=E15.6,7H DMD12= E15.6,8H PRPRAT= E15.6,6H TSFC=E
     515.6,6H PSFC=E15.6/6H USFC=E15.6,6H D3SQ=E15.6,8H PHISFC=E15.6,
     66H SIDE=I4,6H LIM1=I4,6H LIM2=I4)
10000 RETURN
      END
```

```
      SUBROUTINE PUTIN
C
C     THIS SUBROUTINE READS IN AND INITIALIZES ALL DATA EXCEPT THAT
C     HAVING TO DO WITH AN EXTERNAL NON-UNIFORM FLOW FIELD
C
      COMMON A(10,5)     ,AA(60)      ,ALFA(10,10),ALPHAH      ,ALPHAP
     1       ,ATOL       ,BETAP       ,BMIX       ,C11(10)     ,C(10,60)
```

```
2       ,C12(10)     ,C2(10,60)   ,CABAR(10)   ,CBBAR(10)   ,CP
3       ,CPS(10)     ,CPSH        ,CSH(10)     ,CSH1(10)    ,CSTREM(10)
4       ,MXSTRM      ,D2IH(10,10),DEFF(10)    ,D2EFF(10)   ,DELTA
5       ,D11         ,DELSS(60)   ,DELS        ,DELSO       ,DLS(60)
6       ,DIH(10,10)  ,D12         ,DELY(60)    ,D13         ,DPDY(60)
7       ,D14         ,DPHIDS(60)  ,EPCON
8       ,EPSLON      ,EXTRA(50)   ,FSTEP       ,FMAX        ,GRAD
9       ,H11         ,H(60)       ,HH          ,HJ          ,HPM(10)
*       ,H2PM(10)    ,H3PM(10)    ,ICONST      ,ICOUNT      ,IDENT(10)
1       ,IERROR      ,IEXTRA(50)  ,IFLAG       ,IKIND
2       ,IPTUC       ,ISHOCK      ,ITYPE       ,IPD
3       ,IDIFF       ,K           ,KAY         ,KAYS        ,KAY2
4       ,KLO         ,KMAX
5       ,KUP         ,KW          ,LL          ,LPLANE      ,MA
6       ,MASH        ,MDOT        ,MMAX        ,MUO         ,MU
7       ,MU2         ,MUS         ,MW          ,MW2         ,MWSH
8       ,NBOUND      ,NDS         ,NITER       ,NMAX        ,NN
 COMMON NUCASE       ,OMEGA(10)   ,P11         ,P(60)       ,P12
1       ,P2(60)                   ,PB(50)      ,PABAR       ,PBBAR
2       ,PBS         ,P15         ,PHI(60)                  ,PHIB(50)
3       ,PHISH1      ,PHSTRM                   ,PHIW(50)    ,P18
4       ,PHI2(60)    ,PHIBS       ,PI          ,PR(10)      ,PSH
5       ,PSH1        ,PSI         ,PSTREM                   ,PW(50)
6       ,Q11         ,Q(60)       ,QXTR1       ,QXTR2
7       ,QW(50)      ,R11         ,R(60)                    ,RB(50)
8       ,RBS         ,R13         ,RBAR(60)    ,R14         ,R2(60)
9       ,RCON        ,R15         ,RE(60)      ,RESH        ,R16
*       ,RHO(60)     ,R17         ,RHO2(60)    ,RHS(10)     ,RHSEN
1       ,RHSMOM      ,RHABAR      ,RHBBAR      ,RO          ,RSH
2       ,RSTREM      ,RV          ,R19         ,RW(100)     ,S
3                    ,SB(50)      ,SC(10)                   ,SW(50)
4       ,S13         ,SX(60)      ,T11         ,T(60)       ,T12
5       ,T2(60)      ,TABAR       ,TBBAR       ,TC(10)      ,T14
6       ,TAW(60)     ,TXTR1       ,TXTR2       ,TOL         ,TSH
7       ,TSH1        ,TSTREM                   ,TW(50)      ,TWS
8       ,TS          ,U11         ,U(60)       ,U12         ,U2(60)
 COMMON UXTR1        ,UXTR2       ,U14         ,UBAR(60)    ,USH
```

```
     1        ,USH1        ,USTREM                       ,UW(50)      ,UWS
     2        ,X11         ,X(60)       ,X12             ,X2(60)
     3        ,XB(50)      ,XBS         ,XABAR(10)       ,XBBAR(10)   ,X15
     4        ,XS(10,60)   ,XSH         ,XSTREM                       ,XW(50)
     5                                  ,Y11             ,Y(60)       ,Y12
     6        ,Y2(60)      ,YABAR       ,YBBAR           ,ZA(10)      ,Z11(10)
     7        ,ZJ(10,60)   ,ZMW         ,S2X             ,R2SH        ,X2SH
     8        ,FX(2,60)    ,FR(2,60)    ,FPHI(2,60)      ,FP(2,60)    ,FT(2,60)
     9        ,FU(2,60)    ,INDL(2,60)  ,INDR(2,60)      ,FC(2,60,10),CARD1
     *        ,M           ,N
      REAL    KAY          ,KAYS(10)    ,KAY2            ,KW          ,MDOT(60)
     *        ,MA(60)      ,MU          ,MUS(10)         ,MUC(10)     ,MU2
     *        ,MW(10)      ,MW2         ,MASH            ,MASH1       ,MWSH
      COMMON /IBUGS1/ IBUGSH                                                    002
      COMMON/INPUX/ ITD,XC,YC,FRAC                                              003
      COMMON/TRNSPT/ ALFTD(10),DEXP(10)                                         004
      COMMON/INPRG/ ICSH,IIN                                                    005
      EQUIVALENCE (IVISC,IEXTRA(1)),(IPCH,IEXTRA(5))                            006
      COMMON/INSHCP/ PCC,TCC,UCC,PHCC,COC(10)                                   007
      COMMON/SHOPUT/ XSHW,RSHW,PSIW,PSHW,TSHW,USHW,CSHW(10)                     008
      COMMON/CHEMI/NU1(20,15),NU2(20,15),RF1(20),RF2(20),RF3(20),
     *RB1(20),RB2(20),RB3(20),NCAT,IBURN,NCR,SPRCD(10),ICAT(5,10)
      COMMON/TURBUL/TLE,TPR,EDDYK,ITURB,DELMIX
      EQUIVALENCE (PWSH,EXTRA(11))                                              009
      IIN=0                                                                     010
C
C     INTEGRAL PARAMETERS,LIMITS,AND SWITCHES
C
      CARD1 = 1.                                                                011
      READ (5,100) KMAX,NN,KP,IP,ITYPE,IKIND,MMAX,NMAX,NDS,IPD,NITER,           012
     *IDIFF,LPLANE,IPTUC,IPCH,ISHOCK,IBUGSH,ITD,ICSH,IBURN,NCR,NCAT,            013
     *ITURB
  100 FORMAT (11I3,1I1,1I5,10I3)                                                014
      EXTRA(3)=KMAX                                                             015
      ALPHAP=KP                                                                 016
C
C     PHYSICAL CONSTANTS AND CONVERSIONS
```

```
C
      READ (5,200) BETAP,HJ,RCON                                              017
  200 FORMAT(6E12.4)                                                          018
C
C     NON-INTEGRAL PARAMETERS
C
      READ (5,250) ALPHAH,EPSLON,TOL,DELTA,ATOL,FSTEP,GRAD,FRAC,FRACTN         019
  250 FORMAT (8E9.4,1E8.4)                                                    020
      EXTRA(4)=FRACTN                                                         021
      IF (ITD) 260,270,260                                                    022
  260 READ (5,1600) (ALFTD(I),I=1,NDS)                                        023
      READ (5,1600) ( DEXP(I),I=1,NDS)                                        024
  270 PI=3.14159265**DELTA                                                    025
      RU=RCON*HJ                                                              026
      RV=RU/BETAP                                                             027
C
C     ENTHALPY POLYNOMIAL CONSTANTS
C
      DO 300 I=1,NDS                                                          028
  300 READ (5,200) (A(I,J),J=1,NN)                                            029
      DO 340 J=3,NN                                                           030
      DO 340 I=1,NDS                                                          031
      IF (A(I,J)) 330,340,330                                                 032
  330 ICONST=1                                                                033
  340 CONTINUE                                                                034
C
C     TRANSPORT PROPERTIES
C
      IVISC=0.0                                                               035
      DO 350 I=1,NDS                                                          036
  350 READ (5,400) IDENT(I),MUC(I),TC(I),OMEGA(I),PR(I),SC(I),MW(I)           037
  400 FORMAT(A4,8X,5E12.4,1E8.2)                                              038
      IF (ITURB.EQ.0) GO TO 450
      READ (5,425) TLE,TPR,EDDYK,DELMIX
  425 FORMAT (4E12.5)
      IVISC=1
      GO TO 650
```

```
  450 DO 600 I=1,NDS                                                  039
      IF (MU0(I)) 500,600,500                                         040
  500 IVISC=1                                                         041
  600 CONTINUE                                                        042
  650 MU=10.0**(-20)                                                  043
      MU2=10.0**(-20)                                                 044
      IF (IBURN-1) 1100,700,1100
C
C     CHEMICAL REACTION INFORMATION
C
  700 J=NDS+NCAT
      DO 800 M1=1,NCR
  800 READ (5,850) (NU1(M1,I),I=1,J), (NU2(M1,I),I=1,J)
  850 FORMAT (30I2)
      DO 900 M1=1,NCR
  900 READ (5,950) RF1(M1),RF2(M1),RF3(M1),RB1(M1),RB2(M1),RB3(M1)
  950 FORMAT (6E12.5)
      IF (NCAT.EQ.0) GO TO 1075
      DO 1000 J=1,NCAT
 1000 READ (5,1050) (ICAT(J,I),I=1,NDS)
 1050 FORMAT (10I2)
 1075 CALL PUTOUT(3)
C
C     INNER STREAMLINE POSITION
C
 1100 K=0                                                             045
      READ (5,200) X(K),R(K),PHI(K)                                   046
      X2(0)=X(0)
      R2(0)=R(0)
      PHI2(0)=PHI(0)
      X0=X(K)                                                         047
      Y(K)=0.0                                                        048
C
C     OTHER STREAMLINE POSITION AND STREAMTUBE PROPER
C
      K=1                                                             049
 1200 READ (5,200) X(K),R(K),PHI(K),P(K),T(K),U(K)                    050
```

```
      B=SQRT((X(K)-X(K-1))**2+(R(K)-R(K-1))**2)
      IF (ABS(PHI(K)-PHI(K-1))-1.0E-06) 1300,1300,1400
 1300 DELY(K)=B
      GO TO 1500
 1400 DELY(K)=(PHI(K)-PHI(K-1))*B/(2.0*SIN(.5*(PHI(K)-PHI(K-1))))
 1500 AA(K)=PI*(R(K)+R(K-1))**DELTA*DELY(K)
      Y(K)=Y(K-1)+DELY(K)
C
C     STREAMTUBE COMPOSITION (SPECIES MASS FRACTIONS)
C
      READ (5,1600) (C(I,K),I=1,NDS)
 1600 FORMAT(10E8.5)
C
C     SETUP CALCULATIONS
C
 1900 ZMW=0.0
      DO 2000 I=1,NDS
 2000 ZMW=ZMW+C(I,K)/MW(I)
      ZMW=1.0/ZMW
      DO 2100 I=1,NDS
 2100 XS(I,K)=C(I,K)*ZMW/MW(I)
 2400 RHO(K)=ZMW*P(K)/(RV*T(K))
      MDOT(K)=RHO(K)*U(K)*AA(K)
      FMAX=1.0
      IF (IVISC) 2406,2402,2406
 2402 DO 2404 I=1,NDS
      CPS(I)=0.0
      DO 2404 J=2,NN
 2404 CPS(I)=CPS(I)+FLOAT(J-1)*A(I,J)*T(K)**(J-2)
      GO TO 2408
 2406 IBIG=1
      MULT=0
      CALL TRANSP (T(K),P(K),K,MULT,IBIG)
 2408 CP=0.0
      DO 2410 I=1,NDS
 2410 CP=CP+CPS(I)*C(I,K)
      IF (KAY-MU*CP) 2430,2430,2420
```

```
 2420 FMAX=KAY/(CP*MU)
 2430 DO 2450 I=1,NDS
      MAX=I
      DO 2450 J=1,MAX
      IF (DIH(I,J)*RHO(K)/MU-FMAX) 2450,2450,2440
 2440 FMAX= DIH(I,J)*RHO(K)/MU
 2450 CONTINUE
      RE(K)=RHO(K)*U(K)*DELY(K)/(MU*FMAX)
      MA(K)=U(K)*SQRT((CP*ZMW-RCON)/(CP*T(K)*RU))
      H(K)=0.0
      DO 2500 I=1,NDS
      DO 2500 J=1,NN
 2500 H(K)=H(K)+A(I,J)*C(I,K)*T(K)**(J-1)
      IF (K-KMAX) 2600,2700,2700
 2600 K=K+1
      GO TO 1200
 2700 YO=Y(KMAX)
C
C     WALL CONDITIONS
C
      DO 2800 I=1,MMAX
 2800 READ (5,200) XW(I),RW(I),PHIW(I),PW(I),SW(I)
 3700 SWA=0.0
      PHIWA=0.0
      M=1
      PWSH=PW(M)
      IF (PHIW(M)) 3900,3800,3900
 3800 PHIW(M)=PHI(M-1)
 3900 M=M+1
      IF (PHIW(M)) 4300,4000,4300
 4000 IF (M-MMAX) 4100,4200,4200
 4100 PHIW(M)=ATAN((RW(M+1)-RW(M-1))/(XW(M+1)-XW(M-1)))
      GO TO 4300
 4200 PHIW(M)=2.0*ATAN((RW(M)-RW(M-1))/(XW(M)-XW(M-1)))
      PHIW(M)=PHIW(M)-PHIW(M-1)
 4300 IF (SW(M)) 4800,4400,4800
 4400 FKW=(SIN(PHIW(M))-SIN(PHIW(M-1)))/(XW(M)-XW(M-1))
```

```
      IF (ABS(FKW/(XW(M)-XW(M-1)))-1.E-06) 4600,4600,4500
 4500 DELSWA=(PHIW(M)-PHIW(M-1))/FKW
      GO TO 4700
 4600 DELSWA=SQRT((XW(M)-XW(M-1))**2+(RW(M)-RW(M-1))**2)
 4700 SWA=SWA+DELSWA
      SW(M)=SWA
 4800 IF (M-MMAX) 3900,4900,4900
 4900 DO 5000 N=1,NMAX
 5000 READ (5,200) XB(N),RB(N),PHIB(N),PB(N),SB(N)
 5700 SWA=0.0
C
C     OUTER BOUNDARY CONDITIONS
C
      N=1
      IF (PHIB(N)) 5900,5800,5900
 5800 PHIB(N)=PHI(KMAX)
 5900 N=N+1
      IF (PHIB(N)) 6300,6000,6300
 6000 IF (N-NMAX) 6100,6200,6200
 6100 PHIB(N)=ATAN((RB(N+1)-RB(N-1))/(XB(N+1)-XB(N-1)))
      GO TO 6300
 6200 PHIB(N)=2.0*ATAN((RB(N)-RB(N-1))/(XB(N)-XB(N-1)))
      PHIB(N)=PHIB(N)-PHIB(N-1)
 6300 IF (SB(N)) 6750,6400,6750
 6400 FKW=(SIN(PHIB(N))-SIN(PHIB(N-1)))/(XB(N)-XB(N-1))
      IF (ABS(FKW/(XB(N)-XB(N-1)))-1.E-06) 6600,6600,6500
 6500 DELSWA=(PHIB(N)-PHIB(N-1))/FKW
      GO TO 6700
 6600 DELSWA=SQRT((XB(N)-XB(N-1))**2+(RB(N)-RB(N-1))**2)
 6700 SWA=SWA+DELSWA
      SB(N)=SWA
 6750 IF (N-NMAX) 5900,6800,6800
 6800 IF (IKIND-3) 6905,6850,6850
C
C     CONDITIONS BEHIND OUTER SHOCK
C
 6850 READ (5,200) XSH,RSH,PSI,PSH,TSH,USH
```

```
      READ (5,6900) (CSH(I),I=1,NDS)
 6900 FORMAT(10E8.2)
      PBS=PSH
C
C     EXTERNAL FLOW PROPERTIES AT FIRST SHOCK POINT
C
      READ (5,200) XSTREM,RSTREM,PHSTRM,PSTREM,TSTREM,USTREM
      READ (5,6900) (CSTREM(I),I=1,NDS)
 6905 IF (ITYPE-3) 7000,6910,6910
C
C     CONDITIONS BEHIND INNER SHOCK
C
 6910 READ (5,200) XSHW,RSHW,PSIW,PSHW,TSHW,USHW
      READ (5,6900) (CSHW(I),I=1,NDS)
      EXTRA(11)=PSHW
 7000 IF (ITYPE-3) 10000,7300,7300
 7300 READ(5,200)XOO,ROO,PHOO,POO,TOO,UOO
      READ (5,6900) (COO(I),I=1,NDS)
10000 CALL PUTOUT(1)
      RETURN
      END
```

```
      SUBROUTINE PUTOUT(IOUT)
C
C     THIS SUBROUTINE WRITES OUT THE FLOW PROPERTIES IN EACH STREAMTUBE
C     AT THE DESIRED ORTHOGONAL SURFACES.
C
      COMMON A(10,5)      ,AA(60)       ,ALFA(10,10),ALPHAH      ,ALPHAP
     1       ,ATOL        ,BETAP        ,BMIX       ,C11(10)     ,C(10,60)
     2       ,C12(10)     ,C2(10,60)    ,CABAR(10)  ,CBBAR(10)   ,CP
     3       ,CPS(10)     ,CPSH         ,CSH(10)    ,CSH1(10)    ,CSTREM(10)
     4       ,MXSTRM      ,D2IH(10,10),DEFF(10)     ,D2EFF(10)   ,DELTA
```

```
     5         ,D11          ,DELSS(60)    ,DELS         ,DELSO        ,DLS(60)
     6         ,DIH(10,10)   ,D12          ,DELY(60)     ,D13          ,DPDY(60)
     7         ,D14          ,DPHIDS(60)   ,EPCON
     8         ,EPSLON       ,EXTRA(50)    ,FSTEP        ,FMAX         ,GRAD
     9         ,H11          ,H(60)        ,HH           ,FJ           ,HPM(10)
     *         ,H2PM(10)     ,H3PM(10)     ,ICONST       ,ICOUNT       ,IDENT(10)
     1         ,IERROR       ,IEXTRA(50)   ,IFLAG        ,IKIND
     2         ,IPTUC        ,ISHOCK       ,ITYPE        ,IPD
     3         ,IDIFF        ,K            ,KAY          ,KAYS         ,KAY2
     4         ,KLO          ,KMAX
     5         ,KUP          ,KW           ,LL           ,LPLANE       ,MA
     6         ,MASH         ,MDOT         ,MMAX         ,MUO          ,MU
     7         ,MU2          ,MUS          ,MW           ,MW2          ,MWSH
     8         ,NBOUND       ,NDS          ,NITER        ,NMAX         ,NN
      COMMON NUCASE        ,OMEGA(10)    ,P11          ,P(60)        ,P12
     1         ,P2(60)                     ,PB(50)       ,PABAR        ,PBBAR
     2         ,PBS          ,P15          ,PHI(60)                    ,PHIB(50)
     3         ,PHISH1       ,PHSTRM                     ,PHIW(50)     ,P18
     4         ,PHI2(60)     ,PHIBS        ,PI           ,PR(10)       ,PSH
     5         ,PSH1         ,PSI          ,PSTREM                     ,PW(50)
     6         ,Q11          ,Q(60)        ,QXTR1        ,QXTR2
     7         ,QW(50)       ,R11          ,R(60)                      ,RB(50)
     8         ,RBS          ,R13          ,RBAR(60)     ,R14          ,R2(60)
     9         ,RCON         ,R15          ,RE(60)       ,RESH         ,R16
     *         ,RHO(60)      ,R17          ,RHO2(60)     ,RHS(10)      ,RHSEN
     1         ,RHSMOM       ,RHABAR       ,RHBBAR       ,RU           ,RSH
     2         ,RSTREM       ,RV           ,R19          ,RW(100)      ,S
     3                       ,SB(50)       ,SC(10)                     ,SW(50)
     4         ,S13          ,SX(60)       ,T11          ,T(60)        ,T12
     5         ,T2(60)       ,TABAR        ,TBBAR        ,TC(10)       ,T14
     6         ,TAW(60)      ,TXTR1        ,TXTR2        ,TOL          ,TSH
     7         ,TSH1         ,TSTREM                     ,TW(50)       ,TWS
     8         ,TS           ,U11          ,U(60)        ,U12          ,U2(60)
      COMMON UXTR1         ,UXTR2        ,U14          ,UBAR(60)     ,USH
     1         ,USH1         ,USTREM                     ,UW(50)       ,UWS
     2         ,X11          ,X(60)        ,X12          ,X2(60)
     3         ,XB(50)       ,XBS          ,XABAR(10)    ,XBBAR(10)    ,X15
```

```
     4      ,XS(10,60)   ,XSH          ,XSTREM                    ,XW(50)
     5                                 ,Y11          ,Y(60)       ,Y12
     6      ,Y2(60)      ,YABAR        ,YBBAR        ,ZA(10)      ,Z11(10)
     7      ,ZJ(10,60)   ,ZMW          ,S2X          ,R2SH        ,X2SH
     8      ,FX(2,60)    ,FR(2,60)     ,FPHI(2,60)   ,FP(2,60)    ,FT(2,60)
     9      ,FU(2,60)    ,INDL(2,60)   ,INDR(2,60)   ,FC(2,60,10),CARD1
     *      ,M           ,N
      REAL   KAY         ,KAYS(10)     ,KAY2         ,KW          ,MDOT(60)
     *      ,MA(60)      ,MU           ,MUS(10)      ,MUC(10)     ,MU2
     *      ,MW(10)      ,MW2          ,MASH         ,MASH1       ,MWSH
      COMMON/SHOPUT/ XSHW,RSHW,PSIW,PSHW,TSHW,USHW,CSHW(10)                002
      COMMON/CHEMI/NU1(20,15),NU2(20,15),RF1(20),RF2(20),RF3(20),
     *RB1(20),RB2(20),RB3(20),NCAT,IBURN,NCR,SPRCD(10),ICAT(5,10)
      DIMENSION SUMI(10)                                                   003
      GO TO (50,1300,1500),IOUT
   50 L=0                                                                  005
      SUMDOT=0.0                                                           006
      DO 75 I=1,NDS                                                        007
   75 SUMI(I)=0.0                                                          008
      WRITE (6,100) LL,X2(L),R2(L),PHI2(L)                                 009
  100 FORMAT ( 1H0,I5,8X,3HXW=,   E11.4,3X,3HRW=,   E11.4,3X,5HPHIW=,      010
     *E11.4)                                                               012
      L=1                                                                  013
  250 IF(LL) 225,260,260                                                   014
  225 WRITE(6,300) L,X(L),R(L),PHI(L),T(L),P(L),U(L)                       015
      GO TO 325                                                            016
  260 WRITE (6,300) L,X2(L),R2(L),PHI2(L),T2(L),P2(L),U2(L)                017
  300 FORMAT(1H0,5X,I5,3X,2HX=,   E11.4,4X,2HR=,   E11.4,4X,4HPHI=,   E1   018
     *1.4,2X,2HT=,   E11.4,4X,2HP=,   E11.4,4X,2HU=,   E11.4)
  325 HTK=H(L)+U(L)**2/(2.0*HJ)                                            020
      PTK=P(L)+RHO(L)*U(L)**2/(2.0*BETAP)                                  021
      SUMDOT=SUMDOT+MDOT(L)                                                022
  375 IF(LL) 350,380,380                                                   023
  350 WRITE (6,400) MA(L),DELY(L),H(L),HTK,TAW(L),Q(L),PTK,RHO(L),SX(L)    024
     *,SUMDOT                                                              025
  400 FORMAT(14X,3HMA=,   E11.4,3X,5HDELY=,   E11.4,1X,2HH=,   E11.4,4X,   026
     *3HHT=,   E11.4,3X,4HTAW=,   E11.4,2X,2HQ=,   E11.4 / 14X,3HPT=,      027
```

```
     *E11.4,3X,4HRHO=,    E11.4,2X,3HSX=,    E11.4,3X,7HSUMDCT=,    E11.4
     *)
      GO TO 410
  380 WRITE (6,400) MA(L),DELY(L),H(L),HTK,TAW(L),Q(L),PTK,RHO2(L),SX(L)
     *,SUMDCT
  410 WRITE (6,1000)
 1000 FORMAT(1H0)
      IF (NDS-1) 450,450,1050
 1050 DO 1299 I=1,NDS
      IF (IEXTRA(I)) 1120,1110,1120
 1110 IF (LL) 1105,1115,1115
 1105 WRITE (6,1200) I,C(I,L),XS(I,L)
      GO TO 1299
 1115 WRITE (6,1200) I,C2(I,L),XS(I,L)
 1200 FORMAT(14X,6HSPECIE,I4,7X,2HC=,    E11.4,4X,2HX=,    E11.4)
      GO TO 1299
 1120 SUMI(I)=SUMI(I)+MDCT(L)*C(I,L)
 1150 IF (LL) 1125,1160,1160
 1125 WRITE (6,1225) I,C(I,L),XS(I,L),SUMI(I),ZJ(I,L)
 1225 FORMAT(14X,6HSPECIE,I4,7X,2HC=,    E11.4,4X,2HX=,    E11.4,3X,3HF1=,
     *    E11.4,3X,3HZJ=,    E11.4)
      GO TO 1299
 1160 WRITE (6,1225) I,C2(I,L),XS(I,L),SUMI(I),ZJ(I,L)
 1299 CONTINUE
  450 IF (L-K) 500,600,600
  500 L=L+1
      GO TO 250
 1300 IF (NBOUND) 1320,1310,1320
 1310 K=0
      DELPHI=PHI(K)-PHISH1
      WRITE (6,1400) LL,K,X2(K),R2(K),PHI2(K),PSIW,DELPHI
      K=1
      GO TO 600
 1320 DELPHI=PHI(KMAX)-PHISH1
      WRITE (6,1400) LL,KMAX,X(KMAX),R(KMAX),PHI(KMAX),PSI,DELPHI
 1400 FORMAT (1H0,5HSHOCK,2X,3HLL=,I4,2X,2HK=,I2,2X,2HX=,    E11.4,2X,2HR
     *=,    E11.4,2X,4HPHI=,    E11.4,2X,4HPSI=,    E11.4,2X,7HDELPHI=,    E
```

```
     *11.4)
      GO TO 600
 1500 WRITE (6,1900)
 1900 FORMAT (1H0,'STOICHIOMETRIC COEFFICIENT MATRIX')
      J=NDS+NCAT
      DO 2000 M1=1,NCR
 2000 WRITE (6,2100) (NU1(M1,I),I=1,J),(NU2(M1,I),I=1,J)
 2100 FORMAT (1H ,30I2)
      WRITE (6,2200)
 2200 FORMAT (1H0,'REACTION RATES')
      DO 2300 M1=1,NCR
 2300 WRITE (6,2400) M1,RF1(M1),RF2(M1),RF3(M1),RB1(M1),RB2(M1),RB3(M1)
 2400 FORMAT (1H ,I2,2X,6E14.5)
  600 RETURN
      END
```



```
      SUBROUTINE SHOCKE
C
C     THIS SUBROUTINE DOES THE SHOCK CALCULATION
C
      COMMON A(10,5)     ,AA(60)      ,ALFA(10,10),ALPHAH      ,ALPHAP
     1       ,ATOL        ,BETAP       ,BMIX        ,C11(10)     ,C(10,60)
     2       ,C12(10)     ,C2(10,60)   ,CABAR(10)   ,CBBAR(10)   ,CP
     3       ,CPS(10)     ,CPSH        ,CSH(10)     ,CSH1(10)    ,CSTREM(10)
     4       ,MXSTRM      ,D2IH(10,10),DEFF(10)    ,D2EFF(10)   ,DELTA
     5       ,D11         ,DELSS(60)   ,DELS        ,DELSC       ,DLS(60)
     6       ,DIH(10,10)  ,D12         ,DELY(60)    ,D13         ,DPDY(60)
     7       ,D14         ,DPHIDS(60)  ,EPCON
     8       ,EPSLON      ,EXTRA(50)   ,FSTEP       ,FMAX        ,GRAD
     9       ,H11         ,H(60)       ,HH          ,HJ          ,HPM(10)
     *       ,H2PM(10)    ,H3PM(10)    ,ICONST      ,ICOUNT      ,IDENT(10)
     1       ,IERROR      ,IEXTRA(50)  ,IFLAG       ,IKIND
```

```
2         ,IPTUC        ,ISHOCK       ,ITYPE        ,IPD
3         ,IDIFF        ,K            ,KAY          ,KAYS         ,KAY2
4         ,KLC          ,KMAX
5         ,KUP          ,KW           ,LL           ,LPLANE       ,MA
6         ,MASH         ,MDOT         ,MMAX         ,MUC          ,MU
7         ,MU2          ,MUS          ,MW           ,MW2          ,MWSH
8         ,NBOUND       ,NDS          ,NITER        ,NMAX         ,NN
 COMMON NUCASE          ,OMEGA(10)    ,P11          ,P(60)        ,P12
1         ,P2(60)                     ,PB(50)       ,PABAR        ,PEBAR
2         ,PBS          ,P15          ,PHI(60)                    ,PHIB(50)
3         ,PHISH1       ,PHSTRM                     ,PHIW(50)     ,P18
4         ,PHI2(60)     ,PHIBS        ,PI           ,PR(10)       ,PSH
5         ,PSH1         ,PSI          ,PSTREM                     ,PW(50)
6         ,Q11          ,Q(60)        ,QXTR1        ,QXTR2
7         ,QW(50)       ,R11          ,R(60)                      ,RB(50)
8         ,RBS          ,R13          ,RBAR(60)     ,R14          ,R2(60)
9         ,RCON         ,R15          ,RE(60)       ,RESH         ,R16
*         ,RHO(60)      ,R17          ,RHO2(60)     ,RHS(10)      ,RHSEN
1         ,RHSMOM       ,RHABAR       ,RHBBAR       ,RU           ,RSH
2         ,RSTREM       ,RV           ,R19          ,RW(100)      ,S
3                       ,SB(50)       ,SC(10)                     ,SW(50)
4         ,S13          ,SX(60)       ,T11          ,T(60)        ,T12
5         ,T2(60)       ,TABAR        ,TBBAR        ,TC(10)       ,T14
6         ,TAW(60)      ,TXTR1        ,TXTR2        ,TCL          ,TSH
7         ,TSH1         ,TSTREM                     ,TW(50)       ,TWS
8         ,TS           ,U11          ,U(60)        ,U12          ,U2(60)
 COMMON UXTR1           ,UXTR2        ,U14          ,UBAR(60)     ,USH
1         ,USH1         ,USTREM                     ,UW(50)       ,UWS
2         ,X11          ,X(60)        ,X12          ,X2(60)
3         ,XB(50)       ,XBS          ,XABAR(10)    ,XBBAR(10)    ,X15
4         ,XS(10,60)    ,XSH          ,XSTREM                     ,XW(50)
5                                     ,Y11          ,Y(60)        ,Y12
6         ,Y2(60)       ,YABAR        ,YBBAR        ,ZA(10)       ,Z11(10)
7         ,ZJ(10,60)    ,ZMW          ,S2X          ,R2SH         ,X2SH
8         ,FX(2,60)     ,FR(2,60)     ,FPHI(2,60)   ,FP(2,60)     ,FT(2,60)
9         ,FU(2,60)     ,INDL(2,60)   ,INDR(2,60)   ,FC(2,60,10)  ,CARD1
*         ,M            ,N
```

```
      REAL    KAY          ,KAYS(10)    ,KAY2        ,KW          ,MDOT(60)
     1        ,MA(60)      ,MU          ,MUS(10)     ,MUO(10)     ,MU2
     2        ,MW(10)      ,MW2         ,MASH        ,MASH1       ,MWSH
      COMMON/SHOPUT/ XSHW,RSHW,PSIW,PSHW,TSHW,USHW,CSHW(10)
      COMMON/SHOSHO/X3SH,R3SH
      COMMON /IBUGS1/ IBUGSH
      COMMON/MAINSH/ KOUNT,KINT
      EQUIVALENCE (PWSH,EXTRA(11))
      DIMENSION G(2),DGD(2,2),XX(2)
      FL=LL
      FPLANE=LPLANE
      FMULT=FL/FPLANE
      IF (EXTRA(4).NE.0.0) GO TO 5
      GSTEP=FSTEP
      GO TO 10
    5 GSTEP=FSTEP-(FSTEP-EXTRA(3))*(1.-FMULT)/EXTRA(4)
   10 IF (LL) 24,24,50
C
C     DETERMINING UPSTREAM PROPERTIES AT INITIAL SHOCK POINT
C
   24 IF(IFLAG) 50,25,50
   25 ZMB=0.0
      GO TO (27,26,27,27,26), ISHOCK
   26 IF (NBOUND) 265,27,265
  265 NK=K
      K=KMAX
      R2SH=RSH
      X2SH=XSH
      CALL SHOPRO
      PHSTRM=PHISH1
      PSTREM=PSH1
      TSTREM=TSH1
      USTREM=USH1
      K=NK
   27 IF(IKIND-3) 305,305,28
   28 CPB=0.0
      DO 30 I=1,NOS
```

```
      ZMB=ZMB+CSH(I)/MW(I)
   30 CPB=CPB+CSH(I)*A(I,2)
      ZMB=1.0/ZMB
      GBOLD=ZMB*PSTREM*USTREM/(RV*TSTREM)
      BMASUM=0.0
      BHSUM=0.0
      IF (ICONST) 302,301,302
  301 FHBOLD=GBOLD*(CPB*TSTREM+USTREM**2/(2.0*HJ))
      GO TO 304
  302 FHBOLD=GBOLD*USTREM**2/(2.0*HJ)
      DO 303 I=1,NDS
      DO 303 J=1,NN
  303 FHBOLD=FHBOLD+GBOLD*A(I,J)*CSH1(I)*TSTREM**(J-1)
  304 PHBOLD=PHSTRM
      ZWM=0.0
      IF (NBOUND) 50,305,50
  305 IF (ISHOCK-4) 32,31,32
   31 MK=K
      K=0
      R3SH=RSHW
      X3SH=XSHW
      CALL SHOPRO
      PHOO=PHISH1
      POO=PSH1
      TOO=TSH1
      UOO=USH1
      K=MK
   32 IF (ITYPE-2) 50,50,33
   33 CPW=0.0
      DO 35 I=1,NDS
      ZWM=ZWM+CSHW(I)/MW(I)
   35 CPW=CPW+CSHW(I)*A(I,2)
      ZWM=1.0/ZWM
      GWOLD=ZWM*POO*UOO/(RV*TOO)
      WMASUM=0.0
      WHSUM=0.0
      IF (ICONST) 42,40,42
```

```
   40 FHWOLD=GWOLD*(CPW*TCO+UCO**2/(2.0*HJ))
      GO TO 46
   42 FHWOLD=GWOLD*UCO**2/(2.0*HJ)
      DO 44 I=1,NDS
      DO 44 J=1,NN
   44 FHWOLD=FHWOLD+GWOLD*A(I,J)*CSH1(I)*TCO**(J-1)
   46 PHWOLD=PHOO
C
C     LOCATING NEW SHOCK POINT
C
   50 IF (NBOUND) 100,200,100
  100 R2SH=(RSH-XSH*TAN(PSI)+R2(K)*TAN(PHI2(K))*TAN(PSI)+X2(K)*TAN(PSI))
     */(1.0+TAN(PSI)*TAN(PHI2(K)))
      X2SH=(R2(K)-R2SH)*TAN(PHI2(K))+X2(K)
      GO TO 300
  200 R3SH=(RSHW-XSHW*TAN(PSIW)+R2(K)*TAN(PHI2(K))*TAN(PSIW)+X2(K)*TAN(P
     *SIW))/(1.0+TAN(PSIW)*TAN(PHI2(K)))
      X3SH=(R2(K)-R3SH)*TAN(PHI2(K))+X2(K)
  300 CALL SHCPRO
      CPSH=0.0
      MWSH=0.0
      DO 500 I=1,NDS
      DO 333 J=1,NN
  333 CPSH=CPSH+A(I,J)*FLOAT(J-1)*CSH1(I)*TSH1**(J-2)
  500 MWSH=MWSH+CSH1(I)/MW(I)
      MWSH=1.0/MWSH
      GAMMA=CPSH/(CPSH-RCON/MWSH)
      MASH1=USH1/SQRT(GAMMA*RU*TSH1/MWSH)
      IF (ICONST) 1500,400,1500
C
C     SHOCK CALCULATION FOR CONSTANT GAMMA
C
C
C     DETERMINATION OF SHOCK ANGLE
C
  400 DELPHI=PHI2(K)-PHISH1
      B1=-(MASH1**2+2.0)/(MASH1**2)-GAMMA*SIN(DELPHI)**2
```

```
      B2=(2.0*MASH1**2+1.0)/(MASH1**4)
      B2=B2+((GAMMA+1.0)**2/4.0+(GAMMA-1.0)/(MASH1**2))*SIN(DELPHI)**2
      B3=COS(DELPHI)**2/(MASH1**4)*(-1.0)
      ITER=0
      IF (NBOUND)620,600,620
  600 IF (DELPHI)625,605,605
  605 ZETA=(1.0/MASH1)**2
      US=USH1*(1.+ABS(DELPHI)/SQRT(MASH1**2-1.))
      TS=TSH1-(US**2-USH1**2)/(2.0*HJ*CPSH)
      PS=PSH1*(TS/TSH1)**(GAMMA/(GAMMA-1.))
      IF (NBOUND) 615,610,615
  610 PSHW=PS
      TSHW=TS
      USHW=US
      GO TO 1350
  615 PSH=PS
      TSH=TS
      USH=US
      GO TO 1175
  620 IF (DELPHI) 605,605,630
  625 ZETA=(SIN(PSIW-PHISH1))**2
      GO TO 700
  630 ZETA=(SIN(PSI-PHISH1))**2
  700 ITER=ITER+1
      G(1)=ZETA**3+B1*ZETA**2+B2*ZETA+B3
      DGDZ=3.0*ZETA**2+2.0*B1*ZETA+B2
      DLZETA=-G(1)/DGDZ
      IF(ABS(DLZETA/ZETA)-TOL) 1100,800,800
  800 IF (ITER-NITER) 900,900,1000
  900 ZETA=ZETA+DLZETA
      GO TO 700
 1000 WRITE (6,1050)
 1050 FORMAT(27H1ITERATIONS FOR ZETA FAILED)
      CALL PDUMP (A(1,1),N,5)
C
C     DETERMINATION OF SHOCKED GAS PROPERTIES
C
```

```
1100 CON4=CPSH*TSH1+USH1**2/(2.0*HJ)
     IF (NBOUND) 1150,1300,1150
1150 PSH=(2.0*GAMMA*MASH1**2*ZETA-GAMMA+1.0)/(GAMMA+1.0)*PSH1
     TTSH=(PSH/PSH1)*((GAMMA-1.0)*MASH1**2*ZETA+2.0)
     TSH=TTSH/((GAMMA+1.0)*MASH1**2*ZETA)*TSH1
     USH=SQRT(2.0*HJ*(CON4-CPSH*TSH))
1175 MASH=MASH1*USH/USH1*SQRT(TSH1/TSH)
     DO 1200 I=1,NDS
1200 CSH(I)=CSH1(I)
     DO 1205 I=1,NDS
     IF (IBUGSH.NE.0) WRITE (6,1210) I,CSH (I)
1210 FORMAT(5H CSH ,I1,1H=,1P1E11.4)
1205 CONTINUE
     IF (IBUGSH.NE.0) WRITE (6,1220) PSH ,TSH ,USH ,ZETA
1220 FORMAT(6H PSH =,1P1E11.4,3X,5HTSH =,1P1E11.4,3X,5HUSH =,1P1E11.4,
    *3X,5HZETA=,1P1E11.4)
     GB=MWSH*PSH1*USH1/(RV*TSH1)
     IF(ICONST)1240,1230,1240
1230 FHB=GB*(CPSH*TSH1+USH1**2/(2.0*HJ))
     GO TO 1260
1240 FHB=GB*USH1**2/(2.0*HJ)
     DO 1250 I=1,NDS
     DO 1250 J=1,NN
1250 FHB=FHB+GB*A(I,J)*CSH1(I)*TSH1**(J-1)
1260 ANGLE=0.5*(PHISH1+PHBOLD)
     GO TO 6000
1300 PSHW=(2.0*GAMMA*MASH1**2*ZETA-GAMMA+1.0)/(GAMMA+1.0)*PSH1
     TTSH=(PSHW/PSH1)*((GAMMA-1.0)*MASH1**2*ZETA+2.0)
     TSHW=TTSH/((GAMMA+1.0)*MASH1**2*ZETA)*TSH1
     USHW=SQRT(2.0*HJ*(CON4-CPSH*TSHW))
1350 MASH=MASH1*USHW/USH1*SQRT(TSH1/TSHW)
     DO 1400 I=1,NDS
1400 CSHW(I)=CSH1(I)
     DO 1405 I=1,NDS
     IF (IBUGSH.NE.0) WRITE (6,1410) I,CSHW(I)
1410 FORMAT(5H CSHW,I1,1H=,1P1E11.4)
1405 CONTINUE
```

```
      IF (IBUGSH.NE.0) WRITE (6,1420) PSHW,TSHW,USHW,ZETA
 1420 FORMAT(6H PSHW=,1P1E11.4,3X,5HTSHW=,1P1E11.4,3X,5HUSHW=,1P1E11.4,
     *3X,5HZETA=,1P1E11.4)
      GW=MWSH*PSH1*USH1/(RV*TSH1)
      IF (ICONST) 1440,1430,1440
 1430 FHW=GW*(CPSH*TSH1+USH1**2/(2.0*HJ))
      GO TO 1460
 1440 FHW=GW*USH1**2/(2.0*HJ)
      DO 1450 I=1,NDS
      DO 1450 J=1,NN
 1450 FHW=FHW+GW*A(I,J)*CSH1(I)*TSH1**(J-1)
 1460 ANGLE=0.5*(PHISH1+PHWOLD)
      GO TO 6000
C
C     SHOCK CALCULATION FOR VARIABLE HEAT CAPACITY
C
 1500 H1=0.0
      DO 1600 J=1,NN
      DO 1600 I=1,NDS
 1600 H1=H1+A(I,J)*CSH1(I)*TSH1**(J-1)
      DO 1900 I=1,NDS
      IF (NBOUND) 1700,1800,1700
 1700 CSH(I)=CSH1(I)
      GO TO 1900
 1800 CSHW(I)=CSH1(I)
 1900 CONTINUE
      ITER=0
      IF (NBOUND) 2000,2100,2000
 2000 TS=TSH
      THS=PSI-PHISH1
      DELPHI=PHI(KMAX)-PHISH1
      IF (DELPHI) 605,605,2200
 2100 TS=TSHW
      THS=PHISH1-PSIW
      DELPHI=PHISH1-PHI(K)
      IF (DELPHI) 2150,2150,2200
 2150 DELPHI=-DELPHI
```

```
      GO TO 605
 2200 ITER=ITER+1
      H2=0.0
      CP=0.0
      DO 2300 I=1,NDS
      DO 2300 J=1,NN
      H2=H2+A(1,J)*CSH1(I)*TS**(J-1)
 2300 CP=CP+FLOAT(J-1)*A(1,J)*TS**(J-2)*CSH1(I)
      SIT=SIN(THS)
      SID=SIN(THS-DELPHI)
      COOT=COS(THS)
      COD=COS(THS-DELPHI)
      TAT=TAN(THS)
      TAD=TAN(THS-DELPHI)
      G(1)=-(H2-H1-USH1**2*(1.-(COOT/COD)**2)/(2.*HJ))
      G(2)=-(PSH1*(TSH1/TAT-TS/TAD)+PSH1*MWSH*USH1**2/(RV*BETAP)*SIT*COO
     *T*(1.-TAD/TAT))
      DGD(1,1)=CP
      DGD(1,2)=USH1**2*COOT*(COOT*TAD-SIT)/(HJ*COD**2)
      DGD(2,1)=-PSH1/TAD
      DGD(2,2)=PSH1*(TS/SID**2-TSH1/SIT**2)+PSH1*MWSH*USH1**2/(RV*BETAP)
     **((COOT**2-SIT**2)*(1.-TAD/TAT)         -COOT**2/COD**2+TAD/TAT)
      IF (IBUGSH.NE.0) WRITE (6,2350) G(1),G(2),TS,THS,DGD(1,1),DGD(1,2)
     *,DGD(2,1),DGD(2,2)
 2350 FORMAT (12X,4E12.5)
      YYY=0.0
      CALL MATE7(2,2,1,DGD,G,YYY,XX,J)
      GO TO (2600,2400,2500),J
 2400 WRITE (6,2405)
 2405 FORMAT (42H1DETERMINANT OVERFLOW IN SUBROUTINE SHOCKE)
      GO TO 2510
 2500 WRITE (6,2505)
 2505 FORMAT (42H1SINGULAR DETERMINANT IN SUBROUTINE SHOCKE)
 2510 CALL PDUMP (A(1,1),N,5)
      CALL EXIT
 2600 DO 2650 I=1,2
 2650 XX(I)=DGD(I,1)
```

```
      IF (ABS(XX(1)/TS)-TOL ) 2800,2700,2700
 2700 TS=TS+XX(1)
      THS=THS+XX(2)
      IF (ITER-NITER) 2200,2200,2750
 2750 WRITE (6,1050)
      GO TO 2510
 2800 IF (ABS(XX(2)/THS )-TOL)  2900,2700,2700
 2900 IF (NBOUND) 3100,3000,3100
 3000 TSHW=TS
      USHW=SQRT(2.*HJ*(H1-H2)+USH1**2)
      PSHW=PSH1*(1.+MWSH/(RV*BETAP*TSH1)*USH1**2*SIT**2*(1.-TAD/TAT))
      ZETA=SIN(THS)**2
      DELPHI=-DELPHI
      GO TO 1350
 3100 TSH=TS
      USH =SQRT(2.*HJ*(H1-H2)+USH1**2)
      PSH =PSH1*(1.+MWSH/(RV*BETAP*TSH1)*USH1**2*SIT**2*(1.-TAD/TAT))
      ZETA=SIN(THS)**2
      GO TO 1175
 6000 IF (NBOUND) 6005,6050,6005
 6005 PBS=PSH
      IF (DELPHI) 6007,6006,6006
 6006 PSITRY=ARSIN(SQRT(ZETA))+PHISH1
      GO TO 6008
 6007 PSITRY=ARSIN(SQRT(ZETA))+PHI2(K)
 6008 IF (IFLAG) 6010,6009,6010
 6009 PSI=0.5*(PSI+PSITRY)
      GO TO 10000
C
C     EXTERNAL SHOCK -- CALCULATION OF PROPERTIES IN STREAMTUBE
C     BETWEEN SHOCK AND OUTERMOST STREAMLINE CARRIED
C
 6010 B=SQRT((R2SH-R(K))**2+(X2SH-X(K))**2)
      BFACT=SQRT((XSH-X2SH)**2+(RSH-R2SH)**2)*ABS(SIN(PSI-ANGLE))
      PROJA=BFACT*(RSH+R2SH)**DELTA*PI
      RESH=RE(KMAX)*PSH*USH*T(KMAX)**(1.0+OMEGA(1))/(DELY(KMAX)*P(KMAX)*
     *U(KMAX)*TSH**(1.0+OMEGA(1)))
```

```
      SUMDOT=0.0
      DO 6015 L=1,KMAX
 6015 SUMDOT=SUMDOT+MDOT(L)*2.0/(R2(L)+R2(L-1))
      BMASUM=BMASUM+PROJA*0.5*(GB+GBOLD)
      BHSUM=BHSUM+PROJA*0.5*(FHB+FHBOLD)
      IF (IBUGSH) 6017,6016,6017
 6017 WRITE (6,6018) BFACT, PROJA, BMASUM, BHSUM, GB, GBOLD, FHB, FHBOLD
     *, ANGLE, MASH1
 6018 FORMAT(7H BFACT=E12.6,8H  PROJA=E12.6,9H  BMASUM=E12.6,8H  BHSUM=E
     *12.6/4H GB=E12.6,7H GBOLD=E12.6,6H  FHB=E12.6,9H  FHBOLD=E12.6,8H
     * ANGLE=E12.6/7H MASH1=E12.6)
      WRITE (6,6019) SUMDOT
 6019 FORMAT(8H SUMDOT=E12.6)
 6016 IF (BMASUM*2.0/(R2SH+R2(KMAX))-1.2*SUMDOT/GSTEP) 7400,6020,6020
C
C     ADDING NEW STREAMTUBE TO SHOCKED FLOW
C
 6020 KMAX=KMAX+1
      X(KMAX)=X2SH
      R(KMAX)=R2SH
      IF (DELPHI) 6025,6025,6022
 6022 PHI(KMAX)=PHI(K)
      GO TO 6027
 6025 PHI(KMAX)=0.5*(PHI(K)+PHISH1)
 6027 P(KMAX)=PSH
      AA(KMAX)=PI*(R(KMAX)+R(K))**DELTA*B
      IF (KOUNT) 6026,6028,6026
 6026 Z=(BMASUM*RV/(AA(KMAX)*MWSH*P(KMAX)))**2/(2.0*HJ)
      IF (ICONST) 4100,4000,4100
 4000 T(KMAX)=(SQRT(CPSH**2+4.0*Z*BHSUM/BMASUM)-CPSH)/(2.0*Z)
      GO TO 4700
 4100 ITER=0
      TS=TSH
 4200 ITER=ITER+1
      GG=Z*TS**2-BHSUM/BMASUM
      DGG=2.0*Z*TS
      DO 4250 I=1,NDS
```

```
      DO 4250 J=1,NN
      GG=GG+A(I,J)*CSH(I)*TS**(J-1)
 4250 DGG=DGG+FLOAT(J-1)*A(I,J)*CSH(I)*TS**(J-2)
      DELT=-GG/DGG
      TS=TS+DELT
      IF (ABS(DELT/TS)-TOL) 4600,4300,4300
 4300 IF (ITER-NITER) 4200,4400,4400
 4400 WRITE (6,4500)
 4500 FORMAT (30H1TEMPERATURE ITERATIONS FAILED)
      GO TO 2510
 4600 T(KMAX)=TS
 4700 U(KMAX)=BMASUM*RV*T(KMAX)/(AA(KMAX)*MWSH*P(KMAX))
      GO TO 6029
 6028 T(KMAX)=TSH
      U(KMAX)=USH
      KCUNT=1
 6029 BMASUM=0.0
      BHSUM=0.0
      DO6030 I=1,NDS
      C(I,KMAX)=CSH(I)
 6030 XS(I,KMAX)=CSH(I)*MWSH/MW(I)
      RHO(KMAX)=MWSH*P(KMAX)/(RV*T(KMAX))
      DELY(KMAX)=B
      Y(KMAX)=Y(K)+B
      MDOT(KMAX)=RHO(KMAX)*U(KMAX)*AA(KMAX)
      RE(KMAX)=RESH*B
      MA(KMAX)=U(KMAX)/(SQRT(CPSH*T(KMAX)*RU/(CPSH*MWSH-RCON)))
      H(KMAX)=0.0
      DO6040 I=1,NDS
      DO6040 J=1,NN
 6040 H(KMAX)=H(KMAX)+A(I,J)*CSH(I)*T(KMAX)**(J-1)
      J=KMAX
      GO TO 6099
 6050 PWSH=PSHW
      IF (DELPHI) 6051,6051,6052
 6051 PSIWTR=PHISH1-ARSIN(SQRT(ZETA))
      GO TO 6053
```

```
 6052 PSIWTR=PHI2(K)-ARSIN(SQRT(ZETA))
 6053 IF (IFLAG) 6055,6054,6055
 6054 PSIW=0.5*(PSIW+PSIWTR)
      GO TO 10000
C
C     INTERNAL SHOCK -- CALCULATION OF PROPERTIES IN STREAMTUBE
C     BETWEEN SHOCK AND INNERMOST STREAMLINE CARRIED
C
 6055 B=SQRT((R3SH-R2(K))**2+(X3SH-X2(K))**2)
      BFACT=SQRT((XSHW-X3SH)**2+(RSHW-R3SH)**2)*ABS(SIN(PSIW-ANGLE))
      PROJAW=BFACT*(RSHW+R3SH)**DELTA*PI
      RESH=RE(1)*PSHW*USHW*T(1)**(1.0+OMEGA(1))/(DELY(1)*P(1)*U(1)*TSHW*
     **(1.0+OMEGA(1)))
      SUMDOT=0.0
      DO 6058 L=1,KMAX
 6058 SUMDOT=SUMDOT+MDOT(L)*2.0/(R2(L)+R2(L-1))
      WMASUM=WMASUM+PROJAW*0.5*(GW+GWOLD)
      WHSUM=WHSUM+PROJAW*0.5*(FHW+FHWOLD)
      IF (IBUGSH) 6057,6061,6057
 6057 WRITE (6,6056) BFACT, PROJAW, WMASUM,WHSUM, GW, GWOLD, FHW, FHWOLD
     *,ANGLE, MASH1
 6056 FORMAT(7H BFACT=E12.6,9H  PROJAW=E12.6,9H  WMASUM=E12.6,8H  WHSUM=
     *E12.6/4H GW=E12.6,7H GWOLD=E12.6,6H  FHW=E12.6,9H  FHWOLD=E12.6,8H
     *  ANGLE=E12.6/7H MASH1=E12.6)
      WRITE (6,6059) SUMDOT
 6059 FORMAT(8H SUMDOT=E12.6)
 6061 J=0
      IF (WMASUM*2.0/(R3SH+R2(J   ))-1.2*SUMDOT/GSTEP) 7600,6060,6060
 6060 KMAX=KMAX+1
C
C     REINDEXING STREAMLINES IN SHOCKED GAS
C
      J=KMAX
 6065 AA(J)=AA(J-1)
      DO 6070 I=1,NDS
      C(I,J)=C(I,J-1)
      C2(I,J)=C2(I,J-1)
```

```
 6070 XS(I,J)=XS(I,J-1)
      DELY(J)=DELY(J-1)
      H(J)=H(J-1)
      MA(J)=MA(J-1)
      MDOT(J)=MDOT(J-1)
      P(J)=P(J-1)
      PHI(J)=PHI(J-1)
      R(J)=R(J-1)
      RE(J)=RE(J-1)
      RHO(J)=RHO(J-1)
      SX(J)=SX(J-1)
      T(J)=T(J-1)
      U(J)=U(J-1)
      X(J)=X(J-1)
      Y(J)=Y(J-1)+B
      P2(J)=P2(J-1)
      PHI2(J)=PHI2(J-1)
      R2(J)=R2(J-1)
      RHO2(J)=RHO2(J-1)
      T2(J)=T2(J-1)
      U2(J)=U2(J-1)
      X2(J)=X2(J-1)
      Y2(J)=Y2(J-1)+B
      DELSS(J)=DELSS(J-1)
      IEXTRA(7)=IEXTRA(7)+1
      J=J-1
      IF (J-1) 6075,6075,6065
C
C     ADDING NEW STREAMTUBE TO SHOCKED FLOW
C
 6075 J=1
      P(J)=PSHW
      P2(J)=PSHW
      AA(J)=PI*(R3SH+R(J-1))**DELTA*B
      IF (KINT) 6076,6077,6076
 6076 Z=       (WMASUM*RV/(AA(J)*MWSH*P(J)))**2/(2.0*HJ)
      IF (ICONST) 5100,5000,5100
```

```
5000 T(J)=(SQRT(CPSH**2+4.0*Z*WHSUM/WMASUM)-CPSH)/(2.0*Z)
     T2(J)=T(J)
     GO TO 5700
5100 ITER=0
     TS=TSHW
5200 ITER=ITER+1
     GG=Z*TS**2-WHSUM/WMASUM
     DGG=2.0*Z*TS
     DO 5250 I=1,NDS
     DO 5250 L=1,NN
     GG=GG+A(I,L)*CSHW(I)*TS**(L-1)
5250 DGG=DGG+FLOAT(L-1)*A(I,L)*CSHW(I)*TS**(L-2)
     DELT=-GG/DGG
     TS=TS+DELT
     IF (ABS(DELT/TS)-TOL) 5600,5300,5300
5300 IF (ITER-NITER) 5200,4400,4400
5600 T(J)=TS
     T2(J)=TS
5700 U(J)=WMASUM*RV*T(J)/(AA(J)*MWSH*P(J))
     U2(J)=U(J)
     GO TO 6078
6077 T(J)=TSHW
     T2(J)=TSHW
     U(J)=USHW
     U2(J)=USHW
     KINT=1
6078 WMASUM=0.0
     WHSUM=0.0
     DO 6080 I=1,NDS
     C(I,J)=CSHW(I)
     C2(I,J)=C(I,J)
6080 XS(I,J)=C(I,J)*MWSH/MW(I)
     RHO(J)=MWSH*P(J)/(RV*T(J))
     RHO2(J)=RHO(J)
     X(J)=X(J-1)
     X2(J)=X2(J-1)
     R(J)=R(J-1)
```

```
      R2(J)=R2(J-1)
      PHI(J)=PHI(J-1)
      PHI2(J)=PHI2(J-1)
      DELY(J)=B
      Y(J)=B
      Y2(J)=B
      MOUT(J)=RHO(J)*U(J)*AA(J)
      RE(J)=RESH*B
      MA(J)=U(J)/(SQRT(CPSH*T(J)*RU/(CPSH*MWSH-RCON)))
      H(J)=0.0
      DO 6090 I=1,NDS
      DO 6090 JJ=1,NN
 6090 H(J)=H(J)+A(I,JJ)*C(I,J)*T(J)**(JJ-1)
      J=0
      X2(J)=X3SH
      R2(J)=R3SH
      IF (DELPHI) 6091,6092,6092
 6091 PHI2(J)=PHI2(1)
      GO TO 6093
 6092 PHI2(J)=0.5*(PHI2(1)+PHISH1)
 6093 Y2(J)=0.0
      DO 7700 J=1,KMAX
 7700 Y2(J)=Y2(J-1)+DELY(J)
 6099 CALL PUTOUT(2)
C
C     PREPARING FOR NEXT STEP DOWNSTREAM
C
 7400 IF (NBOUND) 7500,7600,7500
 7500 XSH=X2SH
      RSH=R2SH
      PSI=PSITRY
      GBOLD=GB
      FHBOLD=FHB
      PHBOLD=PHISH1
      GO TO 10000
 7600 XSHW=X3SH
      RSHW=R3SH
```

```
      PSIW=PSIWTR
      GWOLD=GW
      FHWOLD=FHW
      PHWOLD=PHISH1
      K=1
10000 RETURN
      END
```

```
      SUBROUTINE SHOPRO
C
C     THIS SUBROUTINE DETERMINES THE CONDITIONS UPSTREAM OF THE SHOCK
C
      COMMON A(10,5)     ,AA(60)     ,ALFA(10,10),ALPHAH     ,ALPHAP
     1       ,ATOL       ,BETAP      ,BMIX       ,C11(10)    ,C(10,60)
     2       ,C12(10)    ,C2(10,60)  ,CABAR(10)  ,CBBAR(10)  ,CP
     3       ,CPS(10)    ,CPSH       ,CSH(10)    ,CSH1(10)   ,CSTREM(10)
     4       ,MXSTRM     ,D2IH(10,10),DEFF(10)   ,D2EFF(10)  ,DELTA
     5       ,D11        ,DELSS(60)  ,DELS       ,DELSC      ,DLS(60)
     6       ,DIH(10,10) ,D12        ,DELY(60)   ,D13        ,DPDY(60)
     7       ,D14        ,DPHIDS(60) ,EPCON
     8       ,EPSLON     ,EXTRA(50)  ,FSTEP      ,FMAX       ,GRAD
     9       ,H11        ,H(60)      ,HH         ,HJ         ,HPM(10)
     *       ,H2PM(10)   ,H3PM(10)   ,ICONST     ,ICOUNT     ,IDENT(10)
     1       ,IERROR     ,IEXTRA(50) ,IFLAG      ,IKIND
     2       ,IPTUC      ,ISHOCK     ,ITYPE      ,IPD
     3       ,IDIFF      ,K          ,KAY        ,KAYS       ,KAY2
     4       ,KLO        ,KMAX
     5       ,KUP        ,KW         ,LL         ,LPLANE     ,MA
     6       ,MASH       ,MDOT       ,MMAX       ,MUC        ,MU
     7       ,MU2        ,MUS        ,MW         ,MW2        ,MWSH
     8       ,NBOUND     ,NDS        ,NITER      ,NMAX       ,NN
      COMMON NUCASE      ,OMEGA(10)  ,P11        ,P(60)      ,P12
```

```
1        ,P2(60)                         ,PB(50)      ,PABAR       ,PBBAR
2        ,PBS          ,P15              ,PHI(60)                  ,PHIB(50)
3        ,PHISH1       ,PHSTRM                        ,PHIW(50)    ,P18
4        ,PHI2(60)     ,PHIBS            ,PI          ,PR(10)      ,PSH
5        ,PSH1         ,PSI              ,PSTREM                   ,PW(50)
6        ,Q11          ,Q(60)            ,QXTR1       ,QXTR2
7        ,QW(50)       ,R11              ,R(60)                    ,RB(50)
8        ,RBS          ,R13              ,RBAR(60)    ,R14         ,R2(60)
9        ,RCON         ,R15              ,RE(60)      ,RESH        ,R16
*        ,RHO(60)      ,R17              ,RHO2(60)    ,RHS(10)     ,RHSEN
1        ,RHSMOM       ,RHABAR           ,RHBBAR      ,RU          ,RSH
2        ,RSTREM       ,RV               ,R19         ,RW(100)     ,S
3                      ,SB(50)           ,SC(10)                   ,SW(50)
4        ,S13          ,SX(60)           ,T11         ,T(60)       ,T12
5        ,T2(60)       ,TABAR            ,TBBAR       ,TC(10)      ,T14
6        ,TAW(60)      ,TXTR1            ,TXTR2       ,TOL         ,TSH
7        ,TSH1         ,TSTREM                        ,TW(50)      ,TWS
8        ,TS           ,U11              ,U(60)       ,U12         ,U2(60)
 COMMON  UXTR1         ,UXTR2            ,U14         ,UBAR(60)    ,USH
1        ,USH1         ,USTREM                        ,UW(50)      ,UWS
2        ,X11          ,X(60)            ,X12         ,X2(60)
3        ,XB(50)       ,XBS              ,XABAR(10)   ,XBBAR(10)   ,X15
4        ,XS(10,60)    ,XSH              ,XSTREM                   ,XW(50)
5                                        ,Y11         ,Y(60)       ,Y12
6        ,Y2(60)       ,YABAR            ,YBBAR       ,ZA(10)      ,Z11(10)
7        ,ZJ(10,60)    ,ZMW              ,S2X         ,R2SH        ,X2SH
8        ,FX(2,60)     ,FR(2,60)         ,FPHI(2,60)  ,FP(2,60)    ,FT(2,60)
9        ,FU(2,60)     ,INDL(2,60)       ,INDR(2,60)  ,FC(2,60,10),CARD1
*        ,M            ,N
 REAL    KAY           ,KAYS(10)         ,KAY2        ,KW          ,MDOT(60)
1        ,MA(60)       ,MU               ,MUS(10)     ,MUO(10)     ,MU2
2        ,MW(10)       ,MW2              ,MASH        ,MASH1       ,MWSH
 COMMON /INTRP1/ NSTRM(2),TSFC(2),PSFC(2),USFC(2),PHISFC(2),                  002
1CSFC(2,10),DPTSFC(2)                                                          003
 COMMON /IBUGS1/ IBUGSH                                                        004
 COMMON /JNO/ JNDL(2,120),JNDR(2,120)                                          005
 COMMON/INSHOP/ PCO,TCO,UCO,PHOO,COO(10)                                       006
```

```
      COMMON/INPRO/ ICSH,IIN
      COMMON/CRSHCK/XTRY,RTRY
      COMMON/SHOSHC/X3SH,R3SH
      INTEGER R1S2,R2S2,R1,R3,R2M1,R1P1
      DIMENSION LOCF(2,120),          NTOT(2)
      DIMENSION ARHO(6,6), ATHET(6,6), AP(6,6)
      IF (LL) 130,120,130
  120 RESH=RE(KMAX)
C
C     TESTS FOR UNIFORM OR NON-UNIFORM FLOW
C
  130 GO TO (131,147,141,144,146), ISHOCK
  131 TSH1=TSTREM
      PSH1=PSTREM
      USH1=USTREM
      DO 140  I=1,NDS
  140 CSH1(I) = CSTREM(I)
      PHISH1=PHSTRM
      GO TO 10000
  141 IF (K) 131,142,131
  142 TSH1=TOO
      PSH1=POO
      USH1=UOO
      DO 143 I=1,NDS
  143 CSH1(I)=COO(I)
      PHISH1=PHOO
      GO TO 10000
  144 IF (K) 131,145,131
  145 XTRY=X3SH
      RTRY=R3SH
      GO TO 150
  146 IF (K) 147,142,147
  147 XTRY=X2SH
      RTRY=R2SH
  150 IF (ICSH.NE.0) GO TO 400
C
C     NON-UNIFORM EXTERNAL FLOW WITH TABULAR DATA
```

```
C
      IBEG=1
      IF (IBUGSH.NE.0) WRITE (6,1502) XTRY,RTRY
 1502 FORMAT (///57H BEGIN LOGIC TO FIND PROPERTIES FOR POINT PRECEDING
     1SHOCK/14H POINT IS-- X=E15.6, 3H R=E15.6)
      IF (IFLAG.NE.0) GO TO 1925
      IF (CARD1.NE.1.)GO TO 173
      CARD1 = 2.
      MXSTRM = 60
      MXTCT = 120
      ICTSFC = 0
      I9 = 1
      GO TO 153
  151 I9 = 2
  153 ICTSFC = ICTSFC+1
      J = 0
      IF (IBUGSH.NE.0) WRITE (6,154) ICTSFC
  154 FORMAT (/19H ORTHOGONAL SURFACEI4/2H L9X,1HX14X,1HR13X,3HPHI13X,1H
     1P14X,1HT14X,1HU8X,1CHJNDL   JNDR)
      DO 165 L=1,MXTOT
      J = J+1
      READ (5,155) FX(I9,J),FR(I9,J),FPHI(I9,J),FP(I9,J),FT(I9,J),
     1FU(I9,J),JNDL(I9,L),JNDR(I9,L),(FC(I9,J,I),I=1,NDS)
  155 FORMAT (6E12.0,2I4/10E8.0)
      IF (JNDR(I9,L)) 1551,1559,1551
 1551 RRR=.5*(FR(I9,J)+FR(I9,J-1))
      XXX=.5*(FX(I9,J)+FX(I9,J-1))
      SMW=0.0
      FCP=0.0
      DO 1554 I=1,NDS
 1552 SMW=SMW+FC(I9,J,I)/MW(I)
      FCP=FCP+FC(I9,J,I)*A(I,2)
 1554 CONTINUE
      SMW=1.0/SMW
      DENSTY=SMW*FP(I9,J)/(RV*FT(I9,J))
      FP(I9,J)=DENSTY*(RRR**2+XXX**2)
 1559 IF (JNDR(I9,L).GE.0) GO TO 157
```

```
      J = J-1
      IF (IBUGSH.NE.0) WRITE(6,156)L,JNDL(I9,L),JNDR(I9,L)
  156 FORMAT (I3,90X,2I6)
      GO TO 165
  157 IF (J.NE.1) GO TO 160
      IF (JNDL(I9,L).NE.0) GO TO 9999
      GO TO 162
  160 IF (JNDL(I9,L).EQ.0) GO TO 170
  162 IF (IBUGSH.NE.0) WRITE (6,163) L,FX(I9,J),FR(I9,J),FPHI(I9,J),
     1FP(I9,J),FT(I9,J),FL(I9,J),JNDL(I9,L),JNDR(I9,L)
  163 FORMAT (I3,1P6E15.6,2I6)
  165 LOCF(I9,L) = J
      NSTRM(I9) = J
      NTOT(I9) = MXTOT
      GO TO 172
  170 NSTRM(I9) = J-1
      NTOT(I9) = L-1
  172 IF (I9.NE.2) GO TO 151
  173 NOTUBE = 0
      L = 0
  175 L = L+1
      IF (JNDR(2,L).GE.0)GO TO 177
      IF (L.LT.NTOT (2)) GO TO 175
      GO TO 9999
  177 R1  = L
  178 L = L+1
      IF (JNDR(2,L).GT.0) GO TO 180
  179 IF (L.LT.NTOT (2)) GO TO 178
      IF (NOTUBE.EQ.0) GO TO 9999
      GO TO 300
  180 R3 = L
      IF (IBUGSH.NE.0) WRITE (6,1802) R1,R3
 1802 FORMAT ( 30H POINTS FOUND    ON RIGHT. R1=I3,4H R3=I3)
      NOTUBE = 1
C     TEST TO SEE IF POINT IS BOUNDED BY STREAMTUBE.
      R1S2 = LOCF(2,R1)
      R2S2 = LOCF(2,R3)
```

```
      CALL LOCATE (FX(2,R1S2),FR(2,R1S2),XTRY,RTRY,FX(2,R2S2),FR(2,R2S2)
     1,FX(1,R1),FR(1,R1),IFGOOD)
      GO TO ( 182,181 ), IFGOOD
  181 R1 = R3
      GO TO 178
  182 CALL LOCATE (FX(1,R3),FR(1,R3),XTRY,RTRY,FX(1,R1),FR(1,R1),
     1FX(2,R2S2),FR(2,R2S2),IFGOOD)
      GO TO (185,181), IFGOOD
  185 IF (R3-R1.NE.1) GO TO 186
      L1 = R1
      L2 = R3
      GO TO 192
  186 R2M1 = R3-1
      R1P1 = R1+1
      DO 188 J = R1P1,R2M1
      XXX = FX(2,R1S2)+(FX(2,R2S2)-FX(2,R1S2))*(FX(1,J)-FX(1,R1))/
     1(FX(1,R3)-FX(1,R1))
      RRR = FR(2,R1S2)+(FR(2,R2S2)-FR(2,R1S2))*(FR(1,J)-FR(1,R1))/
     1(FR(1,R3)-FR(1,R1))
      CALL LOCATE (FX(1,J),FR(1,J),XTRY,RTRY,FX(1,J-1),FR(1,J-1),
     1XXX,RRR,IFGOOD)
      GO TO (190,188) , IFGOOD
  188 CONTINUE
      L1 = R2M1
      L2 = R3
      GO TO 192
  190 L1 = J-1
      L2 = J
C     PROJECT POINT BACK TO ORTHOGINAL SURFACES ALONG NORMAL FROM POINT
C     TO THE SURFACES.
  192 IF (IBUGSH.NE.0) WRITE (6,1922) R1,R3,L1,L2
 1922 FORMAT ( 41H POINT BRACKETED BOTH LEFT AND RIGHT. R1=I3,4H R3=I3,
     14H L1=I3,4H L2=I3)
 1925 CALL ORTHOG(2,R1S2,R2S2)
      CALL ORTHOG (1,L1,L2)
      RATIO1 = DPTSFC(2)/(DPTSFC(2)+DPTSFC(1))
      TSH1 = TSFC(2)+RATIO1*(TSFC(1)-TSFC(2))
```

```
      PSH1 = PSFC(2)+RATIO1*(PSFC(1)-PSFC(2))
      USH1 = USFC(2)+RATIO1*(USFC(1)-USFC(2))
      PHISH1 = PHISFC(2)+RATIO1*(PHISFC(1)-PHISFC(2))
  195 DO 200 I=1,NDS
  200 CSH1(I) = CSFC(2,I)+RATIO1*(CSFC(1,I)-CSFC(2,I))
  204 DENSTY=PSH1/(RTRY**2+XTRY**2)
      SMW=0.0
      FCP=0.0
      DO 205 I=1,NDS
      SMW=SMW+CSH1(I)/MW(I)
  205 FCP=FCP+CSH1(I)*A(I,2)
      SMW=1.0/SMW
      PSH1=DENSTY*RV*TSH1/SMW
      IF (IBUGSH.NE.0) WRITE (6,210) TSH1,PSH1,USH1,PHISH1
  210 FORMAT (/34H FINAL PROPERTIES FOR SHOCK POINT./3H T=E15.6,3H P=E15
     1.6,3H U=E15.6,5H PHI=E15.6)
      GO TO 10000
  300 NSTRM2 = NSTRM(I9)
      NTOT2 = NTOT(I9)
      DO 305 J=1,NSTRM2
      FX(1,J) = FX(2,J)
      FR(1,J) = FR(2,J)
      FPHI(1,J) = FPHI(2,J)
      FP(1,J) = FP(2,J)
      FT(1,J) = FT(2,J)
      FU(1,J) = FU(2,J)
      DO 305 I=1,NDS
  305 FC(1,J,I) = FC(2,J,I)
      DO 310 L=1,NTOT2
      JNDL(1,L) = JNDL(2,L)
  310 JNDR(1,L) = JNDR(2,L)
      NSTRM(1) = NSTRM(2)
      NTOT(1) = NTOT(2)
      GO TO 151
C
C     NON-UNIFORM FLOW WITH ANALYTICAL COEFFICIENTS
C
```

```
400 IF (IIN) 420,410,420
410 READ (5,411) THETOO,RHOE,POE,TOE,REX,NEND,NEMD,NENT,NEMT,NENP,NEMP
411 FORMAT (5E12.6,6I3)
    IF (IBUGSH.NE.0) WRITE (6,411) THETOO,RHOE,POE,TOE,REX,NEND,NEMD,
   *NENT,NEMT,NENP,NEMP
    DO 412 I=1,NEND
    READ (5,419) (ARHO(I,J), J=1,NEMD)
    IF (IBUGSH.NE.0) WRITE (6,418) (ARHO(I,J), J=1,NEMD)
412 CONTINUE
    DO 413 I=1,NENT
    READ (5,419) (ATHET(I,J), J=1,NEMT)
    IF (IBUGSH.NE.0) WRITE (6,418) (ATHET(I,J), J=1,NEMT)
413 CONTINUE
    DO 414 I=1,NENP
    READ (5,419) (AP(I,J), J=1,NEMP)
    IF (IBUGSH.NE.0) WRITE (6,418) (AP(I,J), J=1,NEMP)
414 CONTINUE
    SMW=0.0
    DO 415 I=1,NDS
415 SMW=SMW+MW(I)*COC(I)
    RHOO=SMW*POE/(RV*TOE)
    IIN=1
    THT=THETOO-0.5*PI
    XOE= REX*TAN(THT)
    IF (IBUGSH.NE.0) WRITE (6,418) RHOO,XOE
418 FORMAT (6E12.5)
419 FORMAT (6E12.6)
420 ETA=REX/SQRT(RTRY**2+(XTRY-XOE)**2)
    IF (IBUGSH.NE.0) WRITE (6,1502) XTRY,RTRY
    IF (XTRY-XOE) 421,422,423
421 THETA=PI-ATAN(RTRY/(XOE-XTRY))
    GO TO 424
422 THETA=0.5*PI
    GO TO 424
423 THETA=ATAN(RTRY/(XTRY-XOE))
424 CP=0.0
    DO 425 I=1,NDS
```

```
      CSH1(I)=CCC(I)
  425 CP=CP+A(I,2)*CCO(I)
      GAM=CP/(CP-RCON/SMW)
      SUMRHO=0.0
      SUMPHI=0.0
      SUMP=0.0
      DO 430 I=1,NEND
      DO 430 J=1,NEMD
  430 SUMRHO=SUMRHO+ARHO(I,J)*ETA**(J-1)*COS(PI*(FLOAT(I)-0.5)*THETA/
     *THETOO)
      DO 435 I=1,NENT
      DO 435 J=1,NEMT
  435 SUMPHI=SUMPHI+ATHET(I,J)*ETA**(J-1)*COS(.5*PI*(FLOAT(2)-.5)*THETA/
     .*THETOO)
      DO 440 I=1,NENP
      DO 440 J=1,NEMP
  440 SUMP=SUMP+AP(I,J)*ETA**(J-1)*COS(.5*PI*(FLOAT(I)-.5)*THETA/THETOO)
      RHOSH=RHOE*ETA**2*SUMRHO**(2./(GAM-1.))
      PHISH1=THETA*(1.+ETA*SUMPHI)
      POSH=POE*(1.+ETA*SUMPHI)
      RHOOSH=POSH*SMW/(RV*TOE)
      PSH1=(RHOSH/RHOOSH)**GAM*POSH
      TSH1=PSH1*SMW/(RV*RHOSH)
      USH1=SQRT(2.*HJ*CP*(TOE-TSH1))
      IF (IBUGSH.NE.0) WRITE (6,418) ETA,THETA,GAM,POSH,RHOSH
      IF (IBUGSH.NE.0) WRITE (6,210) TSH1,PSH1,USH1,PHISH1
10000 RETURN
 9999 WRITE (6,9998)
 9998 FORMAT ( 16H ERROR IN SHOPRG)
      CALL EXIT
      RETURN
      END
```

```
      SUBROUTINE STABLE                                                   001
C
C     THIS SUBROUTINE DETERMINES STABLE STEPPING DISTANCE AND PUNCHES
C     OUTPUT DATA WHEN CALLED FOR
C
      COMMON A(10,5)     ,AA(60)      ,ALFA(10,10),ALPHAH    ,ALPHAP
     1       ,ATOL       ,BETAP       ,BMIX       ,C11(10)    ,C(10,60)
     2       ,C12(10)    ,C2(10,60)   ,CABAR(10)  ,CBBAR(10)  ,CP
     3       ,CPS(10)    ,CPSH        ,CSH(10)    ,CSH1(10)   ,CSTREM(10)
     4       ,MXSTRM     ,D2IH(10,10),DEFF(10)    ,D2EFF(10)  ,DELTA
     5       ,D11        ,DELSS(60)   ,DELS       ,DELSC      ,DLS(60)
     6       ,DIH(10,10) ,D12         ,DELY(60)   ,D13        ,DPDY(60)
     7       ,D14        ,DPHIDS(60)  ,EPCON
     8       ,EPSLON     ,EXTRA(50)   ,FSTEP      ,FMAX       ,GRAD
     9       ,H11        ,H(60)       ,HH         ,HJ         ,HPM(10)
     *       ,H2PM(10)   ,H3PM(10)    ,ICONST     ,ICOUNT     ,IDENT(10)
     1       ,IERROR     ,IEXTRA(50)  ,IFLAG      ,IKIND
     2       ,IPTUC      ,ISHOCK      ,ITYPE      ,IPD
     3       ,IDIFF      ,K           ,KAY        ,KAYS       ,KAY2
     4       ,KLO        ,KMAX
     5       ,KUP        ,KW          ,LL         ,LPLANE     ,MA
     6       ,MASH       ,MDOT        ,MMAX       ,MUC        ,MU
     7       ,MU2        ,MUS         ,MW         ,MW2        ,MWSH
     8       ,NBOUND     ,NDS         ,NITER      ,NMAX       ,NN
      COMMON NUCASE      ,OMEGA(10)   ,P11        ,P(60)      ,P12
     1       ,P2(60)                  ,PB(50)     ,PABAR      ,PBBAR
     2       ,PBS        ,P15         ,PHI(60)                ,PHIB(50)
     3       ,PHISH1     ,PHSTRM                  ,PHIW(50)   ,P18
     4       ,PHI2(60)   ,PHIBS       ,PI         ,PR(10)     ,PSH
     5       ,PSH1       ,PSI         ,PSTREM                 ,PW(50)
     6       ,Q11        ,Q(60)       ,QXTR1      ,QXTR2
     7       ,QW(50)     ,R11         ,R(60)                  ,RE(50)
     8       ,RBS        ,R13         ,RBAR(60)   ,R14        ,R2(60)
     9       ,RCON       ,R15         ,RE(60)     ,RESH       ,R16
     *       ,RHO(60)    ,R17         ,RHO2(60)   ,RHS(10)    ,RHSEN
     1       ,RHSMOM     ,RHABAR      ,RHBBAR     ,RU         ,RSH
     2       ,RSTREM     ,RV          ,R19        ,RW(100)    ,S
```

```
 3                 ,SB(50)     ,SC(10)                  ,SW(50)
 4       ,S13      ,SX(60)     ,T11        ,T(60)       ,T12
 5       ,T2(60)   ,TABAR      ,TBBAR      ,TC(10)      ,T14
 6       ,TAW(60)  ,TXTR1      ,TXTR2      ,TCL         ,TSH
 7       ,TSH1     ,TSTREM                 ,TW(50)      ,TWS
 8       ,TS       ,U11        ,U(60)      ,U12         ,U2(60)
  COMMON UXTR1     ,UXTR2      ,U14        ,UBAR(50)    ,USH
 1       ,USH1     ,USTREM                 ,UW(50)      ,UWS
 2       ,X11      ,X(60)      ,X12        ,X2(60)
 3       ,XB(50)   ,XBS        ,XABAR(10)  ,XBBAR(10)   ,X15
 4       ,XS(10,60) ,XSH       ,XSTREM                  ,XW(50)
 5                             ,Y11        ,Y(60)       ,Y12
 6       ,Y2(60)   ,YABAR      ,YBBAR      ,ZA(10)      ,Z11(10)
 7       ,ZJ(10,60) ,ZMW       ,S2X        ,R2SH        ,X2SH
 8       ,FX(2,60) ,FR(2,60)   ,FPHI(2,60) ,FP(2,60)    ,FT(2,60)
 9       ,FU(2,60) ,INDL(2,60) ,INDR(2,60) ,FC(2,60,10),CARD1
 *       ,M        ,N
  REAL    KAY      ,KAYS(10)   ,KAY2       ,KW          ,MDCT(60)
 *       ,MA(60)   ,MU         ,MUS(10)    ,MU0(10)     ,MU2
 *       ,MW(10)   ,MW2        ,MASH       ,MASH1       ,MWSH
  DIMENSION INDEX2(120)                                                 002
  IF (LL) 10,10,40                                                      003
C
C    INDEXING
C
10 IEXTRA(6)=0                                                          004
   DO 20 I=1,KMAX                                                       005
20 INDEX2(I)=I                                                          006
   MAX=KMAX+1                                                           007
   DO 30 I=MAX,120                                                      008
30 INDEX2(I)=0                                                          009
40 K=1                                                                  010
   I=1                                                                  012
   DELS=1.0E10                                                          013
45 IF (IEXTRA(5)-LPLANE) 50,50,90                                       014
50 IF (INDEX2(I)) 75,90,90                                              015
75 I=I+1                                                                016
```

```
      GO TO 50                                                    017
C
C     VISCOUS STABILITY CRITERION
C
   90 DEL1=DELY(K)*RE(K)/2.0                                      018
C
C     INERTIAL STABILITY CRITERION
C
      IF (MA(K)-1.0-EPSLON) 100,100,600                           019
  100 WRITE (6,200) K                                             020
  200 FORMAT (25H1FLOW IS SUBSONIC IN TUBE,I5)                    021
      CALL PDUMP (A(1,1),N,5)                                     022
      CALL EXIT                                                   023
  600 DEL2=.5*DELY(K)*(MA(K)**2-1.0)**(.5)                        024
      IF (DEL1.EQ.0.0)  GO TO 590
      DELSS(K)=ALPHAH/(1.0/DEL1+1.0/DEL2)                         025
      GO TO 595
  590 DELSS(K)=ALPHAH*DEL2
  595 CONTINUE
C
C     COMBINING SMALL TUBES
C
      IF (LL) 608,602,602                                         026
  602 FL=LL                                                       027
      FPLANE=LPLANE                                               028
      FMULT=FL/FPLANE                                             029
      GSTEP=FSTEP                                                 030
      IF (EXTRA(4).EQ.0.0)  GO TO 604
       FSTEP=GSTEP-(GSTEP-EXTRA(3))*(1.-FMULT)/EXTRA(4)           031
      GO TO 608
  604 FSTEP=GSTEP
  608 L=K                                                         032
      CALL COMBO(L)                                               033
      FSTEP=GSTEP                                                 034
      IF (L-K) 610,690,620                                        035
  610 K=L                                                         036
      IF (IEXTRA(5)-LPLANE) 630,630,90                            037
```

```
  620 K=L-1
      I=I+1
      IF (IEXTRA(5)-LPLANE) 630,630,90
  630 IJ=I
  640 IJ=IJ-1
      IF (INDEX2(IJ)) 640,650,650
  650 INDEX2(IJ)=-1
      IEXTRA(6)=IEXTRA(6)+1
      GO TO 50
C
C     AREA CHANGE LIMITATION
C
  690 DLNA=(SIN(.5*(PHI(K)+PHI(K-1)))/(.5*(R(K)+R(K-1)))*DELTA+(PHI(K)-
     *PHI(K-1))/DELY(K))*DELSS(K)
      IF (ABS(DLNA)-ATOL)  800,800,700
  700 DELSS(K)=DELSS(K)*ATOL/ABS(DLNA)
  800 IF (DELSS(K)-DELS) 900,1000,1000
  900 DELS=DELSS(K)
      IEXTRA(7)=K
 1000 IF (IEXTRA(5)-LPLANE) 1025,1025,1050
 1025 INDEX2(I)=K
      IEXTRA(8)=I
      I=I+1
 1050 K=K+1
      IF (K-KMAX) 45,45,1055
 1055 IF (KMAX-58) 1057,1057,1060
C
C     COMBINING SMALLEST TUBE WHEN NUMBER OF TUBES GETS TOO LARGE
C
 1057 ITUBE=FSTEP
      IF (KMAX-ITUBE) 1100,1100,1060
 1060 L=IEXTRA(7)
      K=L
      STORE1=GRAD
      STORE2=FSTEP
      STORE3=Y(KMAX)
      GRAD=10.**10
```

```
      FSTEP=10.**(-10)
      Y(KMAX)=10.**20
      CALL COMBE(L)
      GRAD=STORE1
      FSTEP=STORE2
      Y(KMAX)=STORE3
      IF (IEXTRA(5)-LPLANE) 1065,1065,1100
 1065 IF (L-IEXTRA(7)) 1085,1080,1080
 1080 I=IEXTRA(8)
      GO TO 1090
 1085 I=IEXTRA(6)-1
 1090 IEXTRA(6)=IEXTRA(6)+1
      JMAX=KMAX+IEXTRA(6)
      JMIN=I+1
      J=JMAX
 1095 INDEX2(J)=INDEX2(J-1)
      J=J-1
      IF (J-JMIN) 1097,1095,1095
 1097 INDEX2(I)=-1
 1100 K=KMAX+1
 1400 IF (MOD(LL,IEXTRA(5))) 2000,1500,2000
C
C     PUNCHING OUTPUT CARDS
C
 1500 I=0
      WRITE (8,1600) X(I),R(I),PHI(I),P(I),T(I),U(I),I,I
 1600 FORMAT(6E12.5,2I4)
      WRITE (8,1700) (C(J,I),J=1,NDS)
 1700 FORMAT(10F8.5)
      KMXX=KMAX+IEXTRA(6)
      DO 1950 I=1,KMXX
      IF (INDEX2(I)) 1800,1800,1850
 1800 JJ=I
 1825 JJ=JJ+1
      II=INDEX2(JJ)
      IF (II) 1825,1825,1900
 1850 II=INDEX2(I)
```

```
1900 WRITE (6,1600) X(II),R(II),PHI(II),P(II),T(II),U(II),I,INDEX2(I)
1950 WRITE (6,1700) (C(J,II),J=1,NDS)
     IEXTRA(6)=0
     DO 1970 I=1,KMAX
1970 INDEX2(I)=1
     MAX=KMAX+1
     DO 1980 I=MAX,120
1980 INDEX2(I)=0
2000 RETURN
     END
```

```
      SUBROUTINE STEP
C
C     THIS SUBROUTINE CALCULATES STATE PROPERTIES IN A STREAMTUBE
C
      COMMON A(10,5)     ,AA(60)     ,ALFA(10,10),ALPHAH    ,ALPHAP
     1        ,ATOL       ,BETAP      ,BMIX       ,C11(10)    ,C(10,60)
     2        ,C12(10)    ,C2(10,60)  ,CABAR(10)  ,CBBAR(10)  ,CP
     3        ,CPS(10)    ,CPSH       ,CSH(10)    ,CSH1(10)   ,CSTREM(10)
     4        ,MXSTRM     ,D2IH(10,10),DEFF(10)   ,D2EFF(10)  ,DELTA
     5        ,D11        ,DELSS(60)  ,DELS       ,DELSO      ,DLS(60)
     6        ,DIH(10,10) ,D12        ,DELY(60)   ,D13        ,DPDY(60)
     7        ,D14        ,DPHIDS(60) ,EPCON
     8        ,EPSLON     ,EXTRA(50)  ,FSTEP      ,FMAX       ,GRAD
     9        ,H11        ,H(60)      ,HH         ,HJ         ,HPM(10)
     *        ,H2PM(10)   ,H3PM(10)   ,ICONST     ,ICOUNT     ,IDENT(10)
     1        ,IERROR     ,IEXTRA(50) ,IFLAG      ,IKIND
     2        ,IPTUC      ,ISHOCK     ,ITYPE      ,IPD
     3        ,IDIFF      ,K          ,KAY        ,KAYS       ,KAY2
     4        ,KLO        ,KMAX
     5        ,KUP        ,KW         ,LL         ,LPLANE     ,MA
     6        ,MASH       ,MDOT       ,MMAX       ,MUO        ,MU
```

```
7        ,MU2          ,MUS          ,MW           ,MW2          ,MWSH
8        ,NBOUND       ,NDS          ,NITER        ,NMAX         ,NN
 COMMON NUCASE         ,OMEGA(10)    ,P11          ,P(60)        ,P12
1        ,P2(60)                     ,PB(50)       ,PABAR        ,PBBAR
2        ,PBS          ,P15          ,PHI(60)                    ,PHIB(50)
3        ,PHISH1       ,PHSTRM                     ,PHIW(50)     ,P16
4        ,PHI2(60)     ,PHIBS        ,PI           ,PR(10)       ,PSH
5        ,PSH1         ,PSI          ,PSTREM                     ,PW(50)
6        ,Q11          ,Q(60)        ,QXTR1        ,QXTR2
7        ,QW(50)       ,R11          ,R(60)                      ,RB(50)
8        ,RBS          ,R13          ,RBAR(60)     ,R14          ,R2(60)
9        ,RCON         ,R15          ,RE(60)       ,RESH         ,R16
*        ,RHO(60)      ,R17          ,RHO2(60)     ,RHS(10)      ,RHSEN
1        ,RHSMCM       ,RHABAR       ,RHBBAR       ,RU           ,RSH
2        ,RSTREM       ,RV           ,R19          ,RW(100)      ,S
3                      ,SB(50)       ,SC(10)                     ,SW(50)
4        ,S13          ,SX(60)       ,T11          ,T(60)        ,T12
5        ,T2(60)       ,TABAR        ,TBBAR        ,TC(10)       ,T14
6        ,TAW(60)      ,TXTR1        ,TXTR2        ,TOL          ,TSH
7        ,TSH1         ,TSTREM                     ,TW(50)       ,TWS
8        ,TS           ,U11          ,U(60)        ,U12          ,U2(60)
 COMMON UXTR1          ,UXTR2        ,U14          ,UBAR(60)     ,USH
1        ,USH1         ,USTREM                     ,UW(50)       ,UWS
2        ,X11          ,X(60)        ,X12          ,X2(60)
3        ,XB(50)       ,XBS          ,XABAR(10)    ,XBBAR(10)    ,X15
4        ,XS(10,60)    ,XSH          ,XSTREM                     ,XW(50)
5                                    ,Y11          ,Y(60)        ,Y12
6        ,Y2(60)       ,YABAR        ,YBBAR        ,ZA(10)       ,Z11(10)
7        ,ZJ(10,60)    ,ZMW          ,S2X          ,R2SH         ,X2SH
8        ,FX(2,60)     ,FR(2,60)     ,FPHI(2,60)   ,FP(2,60)     ,FT(2,60)
9        ,FU(2,60)     ,INDL(2,60)   ,INDR(2,60)   ,FC(2,60,10)  ,CARD1
*        ,M            ,N
 REAL     KAY          ,KAYS(10)     ,KAY2         ,KW           ,MDOT(60)
*        ,MA(60)       ,MU           ,MUS(10)      ,MUC(10)      ,MU2
*        ,MW(10)       ,MW2          ,MASH         ,MASH1        ,MWSH
 COMMON/CHEMI/NU1(20,15),NU2(20,15),RF1(20),RF2(20),RF3(20),
*RB1(20),RB2(20),RB3(20),NCAT,IBURN,NCR,SPROD(10),ICAT(5,10)
```

```
      EQUIVALENCE (IVISC,IEXTRA(1))                                    002
      EQUIVALENCE (KALARM, IEXTRA(20))                                 003
C
C     EVALUATION OF TRANSPORT TERMS
C
      IF (IVISC) 50,50,200                                             004
   50 RHSMOM=0.0                                                       005
      RHSEN=0.0                                                        006
      DO 100 I=1,NDS                                                   007
  100 RHS(I)=0.0                                                       008
      IF (IFLAG) 125,150,125                                           009
  125 UBAR(K)=U2(K)                                                    010
      GO TO 300                                                        011
  150 UBAR(K)=U(K)                                                     012
      GO TO 300                                                        013
  200 CALL VISCO                                                       014
  300 IF (IBURN.EQ.0) GO TO 400
  350 CALL CHEM
      GO TO 475
  400 DO 450 I=1,NDS
  450 C2(I,K)=C(I,K)+RHS(I)/MOOT(K)
  475 DO 477 I=1,NDS
  476 IF (C2(I,K)) 478,477,477
  477 CONTINUE
      GO TO 480
  478 IF (ICOUNT-2) 479,479,9300
  479 ICOUNT=ICOUNT+1
      KALARM=2
      DELS=DELS/2.0
      GO TO 10000
  480 ZMW=0.0
      DO 500 I=1,NDS                                                   018
  500 ZMW=ZMW+C2(I,K)/MW(I)                                            019
      ZMW=1.0/ZMW                                                      020
      DO 600 J=1,NN                                                    021
      ZA(J)=0.0                                                        022
      DO 600 I=1,NDS                                                   023
```

```
 800 ZA(J)=ZA(J)+A(I,J)*C2(I,K)                                          024
     AOLD=PI*(R(K)+R(K-1))**DELTA*(Y(K)-Y(K-1))                         025
     ITER=0                                                              026
 700 US=UBAR(K)                                                          027
     CP=0.0                                                              028
     DO 702 J=1,NN                                                       029
 702 CP=CP+ZA(J)*FLOAT(J-1)*T(K)**(J-2)                                  030
     ABAR=.5*(AA(K)+AOLD)                                                031
     B1=(RHSMOM+MDOT(K)*U(K))/(BETAP*ABAR)+P(K)                          032
     B2=MDOT(K)/(BETAP*ABAR)                                             033
 703 B3=ZMW*AA(K)/RV/MDOT(K)                                             034
     HF=0.0
     DO 704 I=1,NDS
 704 HF=HF+C2(I,K)*A(I,1)
     IF (ICONST)  705,710,705                                            035
 705 IF (MA(K)-5.0) 800,710,710                                          036
C
C      CONSTANT GAMMA
C
 710 B5=HJ*CP*B3*B1                                                      037
     B6=2.0*HJ*CP*B3*B2-1.0                                              038
     B4=2.0*HJ*RHSEN/MDOT(K)+2.0*HJ* H(K)   +U(K)**2-2.0*HJ*HF           039
     DISCR=B5**2-B6*B4                                                   040
     IF (DISCR) 730,750,750                                              041
 730 IF (ICOUNT-2) 740,740,9400                                          042
 740 ICOUNT=ICOUNT+1                                                     043
     KALARM=2                                                            044
     DELS=DELS/2.0                                                       045
     GO TO 10000                                                         046
 750 US=(B5+SQRT(DISCR))/B6                                              047
     GO TO 1200                                                          048
C
C      VARIABLE HEAT CAPACITY
C
 800 ITER=ITER+1                                                         049
     B4=2.0*HJ*RHSEN/MDOT(K)+2.0*HJ*H(K)+U(K)**2                         050
     DUM1=0.0                                                            051
```

```
      DUM2=0.0
      DO 900 J=1,NN
      DUM1=DUM1+ZA(J)*(B3*(B1-B2*US)*US)**(J-1)
  900 DUM2=DUM2+ZA(J)*FLOAT(J-1)*    (B1*US-B2*US**2) **(J-2)*B3**(J-1)
      G=2.0*HJ*DUM1+US**2-B4
      DGDU=2.0*US+2.0*HJ*DUM2*(B1-2.0*B2*US)
      DELTAU=-G/DGDU
      IF (ABS(DELTAU/US)-TOL) 1200,1000,1000
 1000 IF (ITER-NITER) 1100,1010,1010
 1010 IF (ICOUNT-2) 1020,1020,9500
 1020 ICOUNT=ICOUNT+1
      KALARM=2
      DELS=DELS/2.0
      GO TO 10000
 1100 US=US+DELTAU
      GO TO 800
 1200 U2(K)=US
      P2(K)=B1-B2*US
      T2(K)=B3*P2(K)*US
      RHO2(K)=ZMW*P2(K)/(RV*T2(K))
      IF (P2(K)) 1301,1301,1304
 1301 WRITE (6,1302)
 1302 FORMAT(26H1 PRESSURE BECAME NEGATIVE)
 1303 CALL PDUMP(A(1,1),N,5)
      CALL EXIT
 1304 IF (T2(K)) 1305,1305,1307
 1305 WRITE (6,1306)
 1306 FORMAT(29H1 TEMPERATURE BECAME NEGATIVE)
      GO TO 1303
C
C     STREAMTUBE MACH AND REYNOLDS NUMBERS
C
 1307 FMAX=1.0
      CP=0.0
      DO 1310 I=1,NN
 1310 CP=CP+FLOAT(I-1)*ZA(I)*T2(K)**(I-2)
      IF (IVISC) 1315,1360,1315
```

```
 1315 IF (KAY2-MU2*CP) 1330,1320,1320
 1320 IF (MU2) 1325,1321,1325
 1321 MU2=10.0**(-20)
 1325 FMAX=KAY2/(CP*MU2)
 1330 DO 1350 I=1,NDS
      J=I
 1335 J=J+1
      IF (D2IH(I,J)*RHO2(K)/MU2-FMAX) 1345,1345,1340
 1340 FMAX=D2IH(I,J)*RHO2(K)/MU2
 1345 IF (J-NDS) 1335,1350,1350
 1350 CONTINUE
      IF (FMAX) 1355,1351,1355
 1351 FMAX=1.0
 1355 RE(K)=RHO2(K)*U2(K)*DELY(K)/(MU2*FMAX)
 1360 MA(K)=U2(K)*SQRT((CP*ZMW-RCON)/(CP*T2(K)*RU))
      GO TO 10000
C
C     ERROR MESSAGES
C
 9300 NUCASE=478
      GO TO 9900
 9400 NUCASE=730
      GO TO 9900
 9500 NUCASE=1010
 9900 IERROR=3
10000 RETURN
      END
```

```
      SUBROUTINE TRANSP(TT,PP,KKK,MULT,IBIG)
C
C     THIS ROUTINE CALCULATES THE LAMINAR TRANSPORT PROPERTIES
C
```

```
 COMMON A(10,5)      ,AA(60)      ,ALFA(10,10),ALPHAH      ,ALPHAP
1       ,ATOL        ,BETAP       ,BMIX        ,C11(10)     ,C(10,60)
2       ,C12(10)     ,C2(10,60)   ,CABAR(10)   ,CBBAR(10)   ,CP
3       ,CPS(10)     ,CPSH        ,CSH(10)     ,CSH1(10)    ,CSTREM(10)
4       ,MXSTRM      ,D2IH(10,10) ,DEFF(10)    ,D2EFF(10)   ,DELTA
5       ,D11         ,DELSS(60)   ,DELS        ,DELSO       ,DLS(60)
6       ,DIH(10,10)  ,D12         ,DELY(60)    ,D13         ,DFDY(60)
7       ,D14         ,DPHIDS(60)  ,EPCON
8       ,EPSLON      ,EXTRA(50)   ,FSTEP       ,FMAX        ,GRAD
9       ,H11         ,H(60)       ,HH          ,HJ          ,HPM(10)
*       ,H2PM(10)    ,H3PM(10)    ,ICONST      ,ICOUNT      ,IDENT(10)
1       ,IERROR      ,IEXTRA(50)  ,IFLAG       ,IKIND
2       ,IPTUC       ,ISHOCK      ,ITYPE       ,IPD
3       ,IDIFF       ,K           ,KAY         ,KAYS        ,KAY2
4       ,KLO         ,KMAX
5       ,KUP         ,KW          ,LL          ,LPLANE      ,MA
6       ,MASH        ,MDOT        ,MMAX        ,MUO         ,MU
7       ,MU2         ,MUS         ,MW          ,MW2         ,MWSH
8       ,NBOUND      ,NDS         ,NITER       ,NMAX        ,NN
 COMMON NUCASE       ,OMEGA(10)   ,P11         ,P(60)       ,P12
1       ,P2(60)                   ,PB(50)      ,PABAR       ,PBBAR
2       ,PBS         ,P15         ,PHI(60)                  ,PHIB(50)
3       ,PHISH1      ,PHSTRM                   ,PHIW(50)    ,P18
4       ,PHI2(60)    ,PHIBS       ,PI          ,PR(10)      ,PSH
5       ,PSH1        ,PSI         ,PSTREM                   ,PW(50)
6       ,Q11         ,Q(60)       ,QXTR1       ,QXTR2
7       ,QW(50)      ,R11         ,R(60)                    ,RB(50)
8       ,RBS         ,R13         ,RBAR(60)    ,R14         ,R2(60)
9       ,RCON        ,R15         ,RE(60)      ,RESH        ,R16
*       ,RHO(60)     ,R17         ,RHO2(60)    ,RHS(10)     ,RHSEN
1       ,RHSMOM      ,RHABAR      ,RHBBAR      ,RU          ,RSH
2       ,RSTREM      ,RV          ,R19         ,RW(100)     ,S
3                    ,SB(50)      ,SC(10)                   ,SW(50)
4       ,S13         ,SX(60)      ,T11         ,T(60)       ,T12
5       ,T2(60)      ,TABAR       ,TBBAR       ,TO(10)      ,T14
6       ,TAW(60)     ,TXTR1       ,TXTR2       ,TOL         ,TSH
7       ,TSH1        ,TSTREM                   ,TW(50)      ,TWS
```

```
     8      ,TS         ,U11        ,U(60)      ,U12        ,U2(60)
      COMMON UXTR1      ,UXTR2      ,U14        ,UBAR(60)   ,USH
     1      ,USH1       ,USTREM                 ,UW(50)     ,UWS
     2      ,X11        ,X(60)      ,X12        ,X2(60)
     3      ,XB(50)     ,XBS        ,XABAR(10)  ,XBBAR(10)  ,X15
     4      ,XS(10,60)  ,XSH        ,XSTREM                 ,XW(50)
     5                              ,Y11        ,Y(60)      ,Y12
     6      ,Y2(60)     ,YABAR      ,YBBAR      ,ZA(10)     ,Z11(10)
     7      ,ZJ(10,60)  ,ZMW        ,S2X        ,R2SH       ,X2SH
     8      ,FX(2,60)   ,FR(2,60)   ,FPHI(2,60) ,FP(2,60)   ,FT(2,60)
     9      ,FU(2,60)   ,INDL(2,60) ,INDR(2,60) ,FC(2,50,10),CARD1
     *      ,M          ,N
      REAL   KAY        ,KAYS(10)   ,KAY2       ,KW         ,MDOT(60)
     *      ,MA(60)     ,MU         ,MUS(10)    ,MU0(10)    ,MU2
     *      ,MW(10)     ,MW2        ,MASH       ,MASH1      ,MWSH
      COMMON/TRNSPT/ ALFTD(10),DEXP(10)
      COMMON/TRANUX/ DT(10),D2T(10)
      COMMON/INPUX/ ITD,X0,Y0,FRAC
      DIMENSION ALPHT(10)
      DIMENSION SIG(10),SIGBAR(10,10),ZETA(10,10),RHORHO(10,10)
      DIMENSION XMOLE(10)
      KAY2=KAY
      MU2=MU
C
C     VISCOSITY AND THERMAL CONDUCTIVITY
C
      MU=0.0
      KAY=0.0
      DO 300 I=1,NDS
      MUS(I)=      (TT/T0(I))**OMEGA(I)*MU0(I)
      CPS(I)=0.0
      DO 200 J=2,NN
  200 CPS(I)=CPS(I)+FLOAT(J-1)*A(I,J)*TT**(J-2)
      KAYS(I)=PR(I)*CPS(I)*MUS(I)
      SIG(I)=SQRT(TT*MW(I))/MUS(I)
      IF (KKK-1) 230,230,210
  210 IF (KKK-KMAX) 220,230,230
```

```
  220 IF (LL) 230,240,240
  230 XMOLE(I)=XS(I,KKK)
      GO TO 250
  240 XMOLE(I)=XABAR(I)
  250 MU=MU+MUS(I)*XMOLE(I)
      IF (IBIG) 300,300,260
  260 D2EFF(I)=DEFF(I)
  300 KAY=KAY+KAYS(I)*XMOLE(I)
C
C     DIFFUSION COEFFICIENTS
C
      IF (MULT) 350,400,350
  350 IF (KKK-KMAX) 360,400,400
  360 IF(XS(IBIG,KKK+1)-0.75) 400,610,610
  400 DO 450 I=1,NDS
      DO 420 J=1,NDS
  420 D2IH(I,J)=DIH(I,J)
      JJ=I-1
      DO 450 J=1,JJ
      RHORHO(I,J)=PP*SQRT(MW(I)*MW(J))/(RV*TT)
      ZETA(I,J)=SQRT(TT*(MW(I)+MW(J))/2.0)
      DIH(I,J)=SQRT(SC(I)*SC(J))*ZETA(I,J)/RHORHO(I,J)
      SIGBAR(I,J)=(.5*(SQRT(SIG(I))+SQRT(SIG(J))))**2
  450 DIH(I,J)=DIH(I,J)/SIGBAR(I,J)
      DO 600 J=1,NDS
      II=J+1
      IF (II-(NDS+1)) 550,540,540
  540 II=NDS
  550 DO 600 I=II,NDS
  600 DIH(J,I)=DIH(I,J)
      DO 603 I=1,NDS
      DIH(I,I)=0.0
      DO 603 J=1,I
      IF (IBIG-I) 603,602,603
  602 DEFF(J)=DIH(I,J)
  603 CONTINUE
      GO TO 647
```

```
  610 J=1
  620 DO 630 I=1,NDS
      D2IH(I,J)=DIH(I,J)
      DO 625 L=1,NDS
  625 DIH(I,L)=0.0
      IF (I-J) 627,630,627
  627 RHORHO(I,J)=PP*SQRT(MW(I)*MW(J))/(RV*TT)
      ZETA(I,J)=SQRT(TT*(MW(I)+MW(J))/2.0)
      DIH(I,J)=SQRT(SC(I)*SC(J))*ZETA(I,J)/RHORHO(I,J)
      SIGBAR(I,J)=(.5*(SQRT(SIG(I))+SQRT(SIG(J))))**2
      DIH(I,J)=DIH(I,J)/SIGBAR(I,J)
  630 CONTINUE
      IF (IBIG-J) 635,640,635
  635 J=IBIG
      GO TO 620
  640 DO 645 I=1,NDS
  645 DEFF(I)=DIH(I,J)
  647 IF (ITD) 650,10000,650
C
C     THERMAL DIFFUSION COEFFICIENT
C
  650 DO 700 I=1,NDS
      ALPHT(I)=ALFTD(I)*EXP(DEXP(I)/TO(I)-DEXP(I)/TT)
  700 D2T(I)=DT(I)
      DT(1)=0.0
      DO 800 I=2,NDS
      DT(I)=XMOLE(I)*ALPHT(I)*DIH(I,1)*CABAR(I)*CABAR(1)
  800  DT(1)=DT(1)-DT(I)
10000 RETURN
      END



      SUBROUTINE TRANSP(TT,PP,KKK,MULT,IBIG)
```

```
C
C     THIS SUBROUTINE CALCULATES THE TURBULENT TRANSPORT PROPERTIES
C
      COMMON A(10,5)       ,AA(60)       ,ALFA(10,10),ALPHAH      ,ALPHAP
     1      ,ATOL          ,BETAP        ,BMIX        ,C11(10)     ,C(10,60)
     2      ,C12(10)       ,C2(10,60)    ,CABAR(10)   ,CBBAR(10)   ,CP
     3      ,CPS(10)       ,CPSH         ,CSH(10)     ,CSH1(10)    ,CSTREM(10)
     4      ,MXSTRM        ,D2IH(10,10)  ,DEFF(10)    ,D2EFF(10)   ,DELTA
     5      ,D11           ,DELSS(60)    ,DELS        ,DELSC       ,DLS(60)
     6      ,DIH(10,10)    ,D12          ,DELY(60)    ,D13         ,DPDY(60)
     7      ,D14           ,DPHIDS(60)   ,EPCON
     8      ,EPSLON        ,EXTRA(50)    ,FSTEP       ,FMAX        ,GRAD
     9      ,H11           ,H(60)        ,HH          ,HJ          ,HPM(10)
     *      ,H2PM(10)      ,H3PM(10)     ,ICONST      ,ICOUNT      ,IDENT(10)
     1      ,IERROR        ,IEXTRA(50)   ,IFLAG       ,IKIND
     2      ,IPTUC         ,ISHOCK       ,ITYPE       ,IPD
     3      ,IDIFF         ,K            ,KAY         ,KAYS        ,KAY2
     4      ,KLO           ,KMAX
     5      ,KUP           ,KW           ,LL          ,LPLANE      ,MA
     6      ,MASH          ,MDOT         ,MMAX        ,MUC         ,MU
     7      ,MU2           ,MUS          ,MW          ,MW2         ,MWSH
     8      ,NBOUND        ,NDS          ,NITER       ,NMAX        ,NN
      COMMON NUCASE        ,OMEGA(10)    ,P11         ,P(60)       ,P12
     1      ,P2(60)                      ,PB(50)      ,PABAR       ,PBBAR
     2      ,PBS           ,P15          ,PHI(60)                  ,PHIB(50)
     3      ,PHISH1        ,PHSTRM                    ,PHIW(50)    ,P18
     4      ,PHI2(60)      ,PHIBS        ,PI          ,PR(10)      ,PSH
     5      ,PSH1          ,PSI          ,PSTREM                   ,PW(50)
     6      ,Q11           ,Q(60)        ,QXTR1       ,QXTR2
     7      ,QW(50)        ,R11          ,R(60)                    ,RB(50)
     8      ,RBS           ,R13          ,RBAR(60)    ,R14         ,R2(60)
     9      ,RCON          ,R15          ,RE(60)      ,RESH        ,R16
     *      ,RHO(60)       ,R17          ,RHO2(60)    ,RHS(10)     ,RHSEN
     1      ,RHSMOM        ,RHABAR       ,RHBBAR      ,RU          ,RSH
     2      ,RSTREM        ,RV           ,R19         ,RW(100)     ,S
     3                     ,SB(50)       ,SC(10)                   ,SW(50)
     4      ,S13           ,SX(60)       ,T11         ,T(60)       ,T12
```

```
     5        ,T2(60)       ,TABAR        ,TEBAR        ,T3(10)       ,T14
     6        ,TAW(60)      ,TXTR1        ,TXTR2        ,TCL          ,TSH
     7        ,TSH1         ,TSTREM                     ,TW(50)       ,TWS
     8        ,TS           ,U11          ,U(60)        ,U12          ,U2(60)
      COMMON  UXTR1         ,UXTR2        ,U14          ,UBAR(60)     ,USH
     1        ,USH1         ,USTREM                     ,UW(50)       ,UWS
     2        ,X11          ,X(60)        ,X12          ,X2(60)
     3        ,XB(50)       ,XBS          ,XABAR(10)    ,XBBAR(10)    ,X15
     4        ,XS(10,60)    ,XSH          ,XSTREM                     ,XW(50)
     5                                    ,Y11          ,Y(60)        ,Y12
     6        ,Y2(60)       ,YABAR        ,YBBAR        ,ZA(10)       ,Z11(10)
     7        ,ZJ(10,60)    ,ZMW          ,S2X          ,R2SH         ,X2SH
     8        ,FX(2,60)     ,FR(2,60)     ,FPHI(2,60)   ,FP(2,60)     ,FT(2,60)
     9        ,FU(2,60)     ,INDL(2,60)   ,INDR(2,60)   ,FC(2,60,10)  ,CARD1
     *        ,M            ,N
      REAL    KAY           ,KAYS(10)     ,KAY2         ,KW           ,MDOT(60)
     *        ,MA(60)       ,MU           ,MUS(10)      ,MU0(10)      ,MU2
     *        ,MW(10)       ,MW2          ,MASH         ,MASH1        ,MWSH
      COMMON/TURBUL/TLE,TPR,EDDYK,ITURB,DELMIX
      KAY2=KAY
      MU2=MU
      D2IH(1,2)=DIH(1,2)
      IF (KKK-1) 100,100,450
  100 IF (LL.LT.0) GO TO 400
      PUMAX=RHO(1)*U(1)
      PUMIN=RHO(1)*U(1)
      DO 300 L=1,KMAX
      TEST=RHO(L)*U(L)
      IF (TEST.GT.PUMAX) GO TO 200
      IF (TEST.GE.PUMIN) GO TO 300
      PUMIN=TEST
      GO TO 300
  200 PUMAX=TEST
  300 CONTINUE
      DELMIX=Y(KMAX)
      MU=0.047896+(DELMIX/EDDYK)*(PUMAX-PUMIN)
      GO TO 450
```

```
  400 MU=0.047896+(DELMIX/EDDYK)
  450 CP=0.0
      DO 600 I=1,NDS
      CPS(I)=0.0
      DO 500 J=2,NN
  500 CPS(I)=CPS(I)+FLOAT(J-1)*A(I,J)*TT**(J-2)
      IF (LL.LT.0) GO TO 550
      CP=CP+CPS(I)*CABAR(I)
      GO TO 600
  550 CP=CP+CPS(I)*C(I,KKK)
  600 CONTINUE
      KAY=CP*MU/TPR
      IF (LL.LT.0) GO TO 650
      DIH(1,2)=KAY*TLE/(RHABAR*CP)
      GO TO 10000
  650 DIH(1,2)=KAY*TLE/(RHO(KKK)*CP)
10000 RETURN
      END
```

```
      SUBROUTINE VISCO                                                  001
C
C     THIS SUBROUTINE CALCULATES THE FLUX CONTRIBUTIONS TO THE
C     CONSERVATION EQUATIONS
C
      COMMON A(10,5)      ,AA(60)      ,ALFA(10,10),ALPHAH     ,ALPHAP
     1       ,ATOL        ,BETAP       ,BMIX       ,C11(10)    ,C(10,60)
     2       ,C12(10)     ,C2(10,60)   ,CABAR(10)  ,CBBAR(10)  ,CP
     3       ,CPS(10)     ,CPSH        ,CSH(10)    ,CSH1(10)   ,CSTREM(10)
     4       ,MXSTRM      ,D2IH(10,10),DEFF(10)    ,D2EFF(10)  ,DELTA
     5       ,D11         ,DELSS(60)   ,DELS       ,DELSC      ,DLS(60)
     6       ,DIH(10,10)  ,D12         ,DELY(60)   ,D13        ,DPDY(60)
     7       ,D14         ,DPHIDS(60)  ,EPCON
```

```
8         ,EPSLON     ,EXTRA(50)  ,FSTEP      ,FMAX       ,GRAD
9         ,H11        ,H(60)      ,HH         ,HJ         ,HPM(10)
*         ,H2PM(10)   ,H3PM(10)   ,ICONST     ,ICOUNT     ,IDENT(10)
1         ,IERROR     ,IEXTRA(50) ,IFLAG      ,IKIND
2         ,IPTUC      ,ISHOCK     ,ITYPE      ,IPD
3         ,IDIFF      ,K          ,KAY        ,KAYS       ,KAY2
4         ,KLC        ,KMAX
5         ,KUP        ,KW         ,LL         ,LPLANE     ,MA
6         ,MASH       ,MDOT       ,MMAX       ,MUC        ,MU
7         ,MU2        ,MUS        ,MW         ,MW2        ,MWSH
8         ,NBOUND     ,NDS        ,NITER      ,NMAX       ,NN
 COMMON NUCASE        ,OMEGA(10)  ,P11        ,P(60)      ,P12
1         ,P2(60)                 ,PB(50)     ,PABAR      ,PBBAR
2         ,PBS        ,P15        ,PHI(60)                ,PHIB(50)
3         ,PHISH1     ,PHSTRM                 ,PHIW(50)   ,P18
4         ,PHI2(60)   ,PHIBS      ,PI         ,PR(10)     ,PSH
5         ,PSH1       ,PSI        ,PSTREM                 ,PW(50)
6         ,Q11        ,Q(60)      ,QXTR1      ,QXTR2
7         ,QW(50)     ,R11        ,R(60)                  ,RB(50)
8         ,RBS        ,R13        ,RBAR(60)   ,R14        ,R2(60)
9         ,RCON       ,R15        ,RE(60)     ,RESH       ,R16
*         ,RHO(60)    ,R17        ,RHO2(60)   ,RHS(10)    ,RHSEN
1         ,RHSMOM     ,RHABAR     ,RHBBAR     ,RU         ,RSH
2         ,RSTREM     ,RV         ,R19        ,RW(100)    ,S
3                     ,SB(50)     ,SC(10)                 ,SW(50)
4         ,S13        ,SX(60)     ,T11        ,T(60)      ,T12
5         ,T2(60)     ,TABAR      ,TBBAR      ,TC(10)     ,T14
6         ,TAW(60)    ,TXTR1      ,TXTR2      ,TOL        ,TSH
7         ,TSH1       ,TSTREM                 ,TW(50)     ,TWS
8         ,TS         ,U11        ,U(60)      ,U12        ,U2(60)
 COMMON UXTR1         ,UXTR2      ,U14        ,UBAR(60)   ,USH
1         ,USH1       ,USTREM                 ,UW(50)     ,UWS
2         ,X11        ,X(60)      ,X12        ,X2(60)
3         ,XB(50)     ,XBS        ,XABAR(10)  ,XBBAR(10)  ,X15
4         ,XS(10,60)  ,XSH        ,XSTREM                 ,XW(50)
5                                 ,Y11        ,Y(60)      ,Y12
6         ,Y2(60)     ,YABAR      ,YBBAR      ,ZA(10)     ,Z11(10)
```

```
     7        ,ZJ(10,60)   ,ZMW         ,S2X         ,R2SH        ,X2SH
     8        ,FX(2,60)    ,FR(2,60)    ,FPHI(2,60)  ,FP(2,60)    ,FT(2,60)
     9        ,FU(2,60)    ,INCL(2,60)  ,INDR(2,60)  ,FC(2,60,10),CARD1
     *        ,M           ,A
      REAL    KAY          ,KAYS(10)    ,KAY2        ,KW          ,MDOT(60)
     *        ,MA(60)      ,MU          ,MUS(10)     ,MUC(10)     ,MU2
     *        ,MW(10)      ,MW2         ,MASH        ,MASH1       ,MWSH
      IF (P(K)) 100,400,400
  100 WRITE (6,150)
  150 FORMAT(28H1NEGATIVE PRESSURE IN VISCO. )
      CALL DUMP(A(1,1),N  ,5)
  400 CALL FLUX
  500 IF (IFLAG) 600,700,600
  600 RBAR(K)=.5*(R(K)+R2(K))
      GO TO 800
  700 RBAR(K)=R(K)
  800 IF (RBAR(K-1)) 900,1000,900
  900 DUMM=RBAR(K-1)**DELTA
      GO TO 1200
 1000 IF (DELTA) 900,1100,900
 1100 DUMM=1.0
C
C     MOMENTUM TRANSFER
C
 1200 RHSMOM=(RBAR(K)**DELTA*TAW(K)*DLS(K)-DUMM*TAW(K-1)*DLS(K-1))
     **PI*2.0**DELTA
C
C     KINETIC ENERGY TRANSFER
C
      IF (K-1) 1300,1300,1400
 1300 DSSPTN=RBAR(K)**DELTA*TAW(K)*(UBAR(K)+UBAR(K+1))/2.0*DLS(K)-DUMM*T
     *AW(K)*UWS*DLS(K-1)
      GO TO 1500
 1400 DSSPTN=RBAR(K)**DELTA*TAW(K)*(UBAR(K)+UBAR(K+1))/2.0*DLS(K)-DUMM*T
     *AW(K-1)*(UBAR(K)+UBAR(K-1))/2.0*DLS(K-1)
C
C     ENERGY TRANSFER BY DIFFUSION
```

```
C
 1500 DFFLX=0.0
      DO 1700 I=1,NDS
      H3PM(I)=H2PM(I)
      H2PM(I)=HPM(I)
      HPM(I)=0.0
      IF(K+1-KMAX)1550,1650,1650
 1550 DO 1600 J=1,NN
      IF (IFLAG) 1570,1560,1570
 1560 HPM(I)=HPM(I)+A(I,J)*T(K+1)**(J-1)
      GO TO 1600
 1570 HPM(I)=HPM(I)+A(I,J)*(0.5*(T(K+1)+T2(K+1)))**(J-1)
 1600 CONTINUE
 1650 DFFLX=DFFLX+.5*RBAR(K)**DELTA*ZJ(I,K)*DLS(K)*(HPM(I)+H2PM(I))
     *-.5*DUMM*ZJ(I,K-1)*DLS(K-1)*(H2PM(I)+H3PM(I))
C
C    SPECIES TRANSFER
C
 1700 RHS(I)=(RBAR(K)**DELTA*ZJ(I,K)*DLS(K)-DUMM*ZJ(I,K-1)*DLS(K-1))*PI
     **2.0**DELTA
C
C    TOTAL ENERGY TRANSFER
C
      RHSEN=(DFFLX+DSSPTN/HJ+RBAR(K)**DELTA*Q(K)*DLS(K)-DUMM*Q(K-1)*DLS(
     *K-1))*PI*2.0**DELTA
      RETURN
      END
```