Finite-Rate Water Condensation in Combustion-Heated Wind Tunnels

Wayne D. Erickson
Langley Research Center
Hampton, Virginia

Gerald H. Mall
Computer Sciences Corporation
Hampton, Virginia

Ramadas K. Prabhu
PRC Systems Services
A Division of Planning Research Corporation
Hampton, Virginia
Symbols

\( A \)  
nozzle cross-sectional area, m\(^2\)

\( A_{n,k} \)  
constant coefficients in equation (36) for \( C_{p,k} \)

\( a_{2,c} \)  
initial approximation of mols of CO\(_2\) per total mol equivalence of C in mixture

\( B_{n,j} \)  
constant coefficients in equation (35) for \( K_{p,j} \)

\( C_{p,k} \)  
molar heat capacity of species \( k \), J/mol-K

\( c_{p,c} \)  
specific heat capacity of carrier gas, J/kg-K

\( c_{p,k} \)  
specific heat capacity of species \( k \), J/kg-K

\( D_{n,k} \)  
constant coefficients in equation (63) for \( \mu_k \)

\( E_{n,k} \)  
constant coefficients in equation (66) for \( \lambda_k \)

\( F \)  
function defined by equation (111)

\( f \)  
factor to account for carrier gas defined by equation (74)

\( G \)  
function defined by equation (46)

\( h \)  
specific enthalpy of total mixture, J/kg

\( h_G \)  
specific enthalpy of gaseous mixture, J/kg

\( h_L \)  
specific enthalpy of liquid water, J/kg

\( h_o \)  
stagnation specific enthalpy, J/kg

\( \tilde{h}_F, \tilde{h}_{O_2}, \tilde{h}_{N_2} \)  
specific enthalpy of fuel, O\(_2\), and N\(_2\) in feed, J/kg

\( \tilde{h}_o \)  
total specific enthalpy of feed to combustion chamber, J/kg

\( \Delta h^0_{L,G,373.15} \)  
specific enthalpy of evaporation at 373.15 K, J/kg

\( J \)  
nucleation rate, droplets formed/m\(^3\)-s

\( \bar{J}_j \)  
mean nucleation rate defined by equation (108), droplets formed/m\(^3\)-s

\( K_{p,j} \)  
equilibrium constant for reaction \( j \)

\( K_n \)  
Knudsen number defined in terms of \( r \)

\( K_{n*} \)  
Knudsen number defined in terms of \( r_* \)

\( k \)  
Boltzmann constant, J/molecule-K

\( L \)  
latent heat of evaporation at \( T_s(p_1) \), J/kg

\( \bar{\ell} \)  
mean free path, m

\( M \)  
mass of a water droplet, kg

\( m \)  
average molecular mass of gaseous mixture, kg/molecule

\( m_c \)  
average molecular mass of carrier gas, kg/molecule

\( m_1 \)  
molecular mass of water, kg/molecule

\( \dot{m} \)  
mass flow rate, kg total mixture/s

\( n \)  
iteration index and atoms of carbon per molecule of fuel
\[ \Delta n^i \] number of droplets of kind \( i \) per mass of total mixture, droplets/kg

\[ \text{Pr} \] Prandtl number defined in terms of \( c_p,1 \) (eq. (69))

\[ p \] pressure, N/m\(^2\)

\[ p_1 \] partial pressure of water vapor, N/m\(^2\)

\[ p_\infty \] vapor pressure of water over a flat surface, N/m\(^2\)

\[ p^o \] standard state pressure, 0.101325 MN/m\(^2\)

\[ Q \] Kantrowitz correction factor, equation (62)

\[ q_c \] condensation coefficient

\[ q_e \] evaporation coefficient

\[ R \] universal gas constant, 8.314 J/mol-K

\[ \bar{R} \] specific gas constant for mixture, \( R/W, J/kg-K \)

\[ R' \] universal gas constant, 1.987 cal/mol-K

\[ \bar{R}_c \] specific gas constant for carrier gas, \( R/W_c, J/kg-K \)

\[ \bar{R}_1 \] specific gas constant for water, \( R/W_1, J/kg-K \)

\[ r \] droplet radius, m

\[ r_w \] nozzle wall radius, m

\[ r_* \] critical droplet radius, m

\[ s \] specific entropy of total mixture, J/kg-K

\[ s^o_{G,1} \] specific entropy of water vapor at standard state pressure, J/kg-K

\[ s_L \] specific entropy of liquid water, J/kg-K

\[ \Delta s^o_{L,G} \] specific entropy of evaporation at standard state pressure, J/kg-K

\[ T \] temperature, K

\[ T_G \] temperature of gaseous mixture, K

\[ T_L \] temperature of liquid water, K

\[ T_R \] reduced temperature, \( T/T_c \)

\[ T_c \] critical temperature of water, 647.3 K

\[ T_s(p_1) \] saturation temperature based on partial pressure of water vapor, K

\[ t \] time, s

\[ \Delta t \] time step, s

\[ U \] velocity, m/s

\[ W \] average molecular weight of gaseous mixture, kg/mol (also \( W = 1/Y \))

\[ W_c \] average molecular weight of carrier gas, kg/mol

\[ W_k \] molecular weight of species \( k \), kg/mol
mass of liquid water per mass of total mixture
mass of carrier gas per mass of total mixture
distance along nozzle measured from throat, m
computational step along nozzle, m
mol number of mixture, mols/kg
equivalent total mols of elements C, H, O, and N per mass of total mixture, mols/kg
mol number of species k, mols/kg
mol number of species k in gaseous mixture at point in nozzle where $J \equiv J_{\text{min}}$, mols/kg
mols of fuel, O₂, and N₂ per mass of total mixture in feed, mols/kg
mol fraction of species k
parameter defined by equation (112)
ratio of droplet radius to critical radius, $r/r_*$
constant in equation (78)
thermal accommodation coefficient for carrier gas interaction with water droplets
constant in equation (75); also, Langmuir parameter
ratio of specific heats
parameter defined by equation (82)
nozzle boundary layer displacement thickness, m
allowable iteration error
elemental ratio of hydrogen to carbon in mixture
elemental ratio of nitrogen to oxygen in mixture
parameter defined by equation (75)
parameter defined by equation (76)
thermal conductivity of gaseous mixture, J/s-m-K
viscosity of gaseous mixture, N·s/m²
parameter defined by equation (78)
parameter defined by equation (77)
mass density, kg/m³
surface tension, N/m
equivalence ratio
parameter defined by equation (80)
parameter defined by equation (73)
carrier gas
$G$  
gaseous mixture

$i$  
properties at interface between free molecular and continuum regimes (see appendix A)

$j$  
location index along nozzle axis

$k$  
chemical species index (where $k = 1$ denotes water vapor)

$L$  
liquid water

$\text{min}$  
minimum

$n$  
summation index

$o$  
stagnation conditions

$s$  
saturation condition except in equation (12) where it is stoichiometric condition

$1$  
water vapor

373.15  
evaluated at 373.15 K

Superscripts:

$i$  
droplet type index

$o$  
conditions at point in nozzle where nucleation just begins, except $p^o$ denotes standard state pressure of 0.101325 MN/m$^2$

','  
approximation

'"  
alternate approximation

'*  
sonic condition at nozzle throat

A caret (^) over a symbol indicates the feed condition to combustor.
Introduction

Experimental studies of propulsion systems and aerothermal-structural systems concepts for hypersonic flight require wind tunnels that generate high enthalpy flow. One approach to achieving such flows utilizes combustion at a relatively high pressure followed by expansion of the resulting combustion products to form the test stream. A number of combustion-driven wind tunnels burning various fuels and of various sizes have been built and operated.

Fuels for these facilities include hydrogen, methane (as the major component of natural gas), and isobutane each burning with air or oxygen-enriched air. In all cases, the products of combustion contain a substantial amount of water vapor. Conditions for which condensation of water could occur depend on the equivalence ratio, temperature, and pressure in the tunnel combustion chamber, and the extent of the expansion of the combustion products in the nozzle. If liquid water does form, the process experiences entropy production and the flow properties are altered relative to an isentropic expansion. There is therefore a need to be able to predict and analyze the effects of nucleation and water droplet growth in combustion-driven wind tunnels.

A great deal of computational and experimental work has been carried out over the years on the nucleation and droplet growth of liquid water from pure steam. References 1 and 2 include reviews of previous studies and present computations and comparisons with experimental data for pure steam. One of the most recent studies has been presented by Young in references 3 and 4 and is the starting point and basis for the present work. The method presented by Young for steam is modified and extended to the rapid expansion of combustion products. This requires the addition of a scheme to compute the properties of the reacting gas mixture in the combustion chamber and the subsequent expansion along the nozzle prior to the formation of liquid water and the modification of the nucleation and droplet growth equations to account for the noncondensable components of the combustion products.

Development of Equations

The development of an appropriate set of equations for water condensation from a rapidly expanding flow of combustion products follows closely the approach presented by Young (refs. 3 and 4) for condensation of pure steam. In the present analysis, the tunnel stagnation conditions and the subsequent expansion along the nozzle are described by a reacting mixture of ideal gases in chemical equilibrium. The condition of chemical equilibrium is applied along the expansion until nucleation just begins, at which point the temperature is low enough so that no further reaction needs to be considered. Also the nucleation and droplet growth processes take place in a gaseous mixture of water vapor and a noncondensable carrier gas composed of the other combustion products rather than pure steam. It should be noted that no liquid water is formed while the condition of chemical equilibrium is imposed and that no chemical reaction takes place during nucleation and droplet growth. Figure 1 is a schematic of a combustion-heated wind tunnel to which this analysis is applied.

The description of the nozzle flow process from the combustion chamber through the throat and until nucleation just begins requires a set of flow equations, stoichiometric relationships, a set of equations for computing the equilibrium chemical composition, thermodynamic relations and properties for a reacting gas mixture, and a nucleation rate equation. This process is indicated as part 1 and part 2 in figure 1. Computation of the flow process for continued expansion during nucleation and droplet growth requires additional equations for transport properties, droplet growth rate, and entropy production. This process is indicated as part 3 in figure 1. The following paragraphs present the specific equations required for these processes.

Flow Equations

The continuity equation for quasi-one-dimensional nozzle flow is

\[ \frac{dn}{l-w} = \frac{pG \Delta U}{1-w} \]  (1)

where it has been assumed that liquid water and the gaseous mixture have the same velocity. Equation (1) applies along the entire expansion prior to liquid water formation with \( w = 0 \) as well as with liquid water present.

The momentum equation is

\[ \frac{(1-w)dp}{\rho_G} = -U \, dU \]  (2)
where again the liquid water and gaseous mixture have the same velocity.

The energy equation for adiabatic steady flow is

$$h_o = (1 - w)h_G + \sum_i w_i h_i^L + \frac{U^2}{2}$$  (3)

where the first term on the right-hand side of equation (3) can be written as

$$h_G = (1 - w - w_c)h_{G,1} + w_c h_{G,c}$$  (4)

Now in the region where water vapor condenses and no further chemical reaction occurs in the gas phase, the differential form of the energy equation (eq. (3)) becomes

$$(1 - w - w_c) d h_{G,1} + w_c d h_{G,c} + U dU = h_{G,1} dw - \sum_i w_i d h_i^L - \sum_i h_i^L d w_i$$  (5)

### Stoichiometric Relationships

The computation of the chemical composition of a reacting gas mixture composed of the elements C, H, O, and N requires a specification of the elemental constants $Y_C$, $Y_H$, $Y_O$, and $Y_N$ which represent the equivalent total number of mols of each element per mass of mixture. These four constants can be determined by specifying the elemental nitrogen-to-oxygen ratio $\eta_{N,O}$, the elemental hydrogen-to-carbon ratio $\eta_{H,C}$, the equivalence ratio $\phi$, and

$$\phi = \frac{Y_H/2 + 2Y_C}{Y_O}$$  (6)

and the four elemental constants are related by the identity

$$12Y_C + Y_H + 16Y_O + 14Y_N = 1000$$  (7)

It follows then that

$$Y_O = \frac{500}{8 + 7\eta_{N,O} + \phi[(12 + \eta_{H,C})/(4 + \eta_{H,C})]}$$  (8)

$$Y_C = \frac{2\phi Y_O}{4 + \eta_{H,C}}$$  (9)

$$Y_N = \eta_{N,O} Y_O$$  (10)

$$Y_H = \eta_{H,C} Y_C$$  (11)

These elemental constants will be used in the set of equations appearing in the next section.

The chemical composition of the feed stream to the combustion chamber is also required. Although it is possible to present a scheme for a more general fuel type, attention is limited to aliphatic hydrocarbons burning with a mixture of oxygen and nitrogen. For this class of fuels, the stoichiometric reaction is

$$C_n H_m + \left(n + \frac{m}{4}\right) O_2 = nCO_2 + \frac{m}{2}H_2O$$

and the stoichiometric ratio of fuel to oxygen in the feed is

$$\left(\frac{\dot{Y}_F}{\dot{Y}_O_2}\right)_s = \frac{4}{n(4 + \eta_{H,C})}$$  (12)

so that

$$\dot{Y}_F = \left[\frac{4\phi}{n(4 + \eta_{H,C})}\right] \dot{Y}_O_2$$  (13)

Also of course

$$\dot{Y}_{N_2} = \eta_{N,O} \dot{Y}_O_2$$  (14)

and since

$$\dot{Y}_O_2 = \frac{Y_O}{2}$$  (15)

it follows that

$$\dot{Y}_O_2 = \frac{250}{8 + 7\eta_{N,O} + \phi[(12 + \eta_{H,C})/(4 + \eta_{H,C})]}$$  (16)

In the flow region where nucleation and droplet growth occur, no further chemical reaction takes place so that the mixture composition changes only as a result of water condensation. The gaseous mixture composition in this region can be expressed in the following way in terms of the mass fraction of liquid water formed and the mol numbers of each species in the gaseous mixture just before any liquid water is formed. Let $w$ represent the mass of liquid water in the stream per unit mass of total mixture (liquid water plus gaseous mixture). Also let $Y_k$ represent the mol number of species $k$ in the gaseous mixture and $Y_k^o$ denote the mol number of species $k$ in the gaseous mixture just before any liquid water is formed. Now since the number of mols of liquid water formed per unit mass of the total mixture is $w/W_1$ and the mass of gaseous mixture per mass of total
mixture is \((1 - w)\), it follows that the mol number of water vapor in the gaseous mixture is
\[
Y_1 = \frac{Y^o_1 - w/W_1}{1 - w}
\]  
(17)
in the region where nucleation and droplet growth occur. The mol numbers for the other species, that is for \(k > 1\), are
\[
Y_k = \frac{Y^o_k}{1 - w}
\]  
(18)
It also follows, since
\[
W = \frac{1}{\sum_{k=1}^{10} Y_k}
\]  
(19)
that the average molecular weight of the gaseous mixture is
\[
W = \frac{1 - w}{Y^o - w/W_1}
\]  
(20)
where
\[
Y^o = \sum_{k=1}^{10} Y^o_k
\]  
(21)
Also, the mol fraction of any species in the gaseous mixture is
\[
y_k = Y_k W
\]  
(22)
The average molecular weight of the carrier gas, that is for all species in the gaseous mixture except water vapor, is
\[
W_c = \sum_{k=2}^{10} \frac{Y_k W_k}{Y - Y_1}
\]  
(23)
or noting that \(Y = 1/W\) and using the foregoing expressions for \(y_k\), \(Y_1\), \(Y_k\), and \(W\), it follows that
\[
W_c = \sum_{k=2}^{10} \frac{y_k^o W_k}{1 - y_1^o}
\]  
(24)
which is of course independent of the amount of water condensed, \(w\).

**Equilibrium Chemical Composition**

The equilibrium chemical composition of combustion products includes 10 species numbered from 1 to 10 in the following order—\(H_2O\), \(CO_2\), \(CO\), \(O_2\), \(H_2\), \(N_2\), \(H\), \(O\), \(OH\), and \(NO\). The composition is determined by the simultaneous solution of the following six chemical equilibrium relations corresponding to the six listed reactions:

For reaction I, \(CO_2 + H_2 = CO + H_2O\),
\[
K_{p,1} = \frac{Y_1 Y_3}{Y_2 Y_5}
\]  
(25)
For reaction II, \(2CO_2 = 2CO + O_2\),
\[
K_{p,2} = \frac{Y_3^2 Y_4}{Y_2 Y_5^2 Y_6^2}
\]  
(26)
For reaction III, \(H_2 + O_2 = 2OH\),
\[
K_{p,3} = \frac{Y_6^2}{Y_4 Y_5}
\]  
(27)
For reaction IV, \(H_2 = 2H\),
\[
K_{p,4} = \frac{Y_8^2}{Y_4 Y_5}
\]  
(28)
For reaction V, \(O_2 = 2O\),
\[
K_{p,5} = \frac{Y_8^2}{Y_4 Y_5}
\]  
(29)
For reaction VI, \(O_2 + N_2 = 2NO\),
\[
K_{p,6} = \frac{Y_10^2}{Y_4 Y_6}
\]  
(30)
and the four elemental balance equations:
\[
Y_H = 2Y_1 + 2Y_5 + Y_7 + Y_9
\]  
(31)
\[
Y_O = Y_1 + 2Y_2 + Y_3 + 2Y_4 + Y_8 + Y_9 + Y_{10}
\]  
(32)
\[
Y_N = 2Y_6 + Y_{10}
\]  
(33)
\[
Y_C = Y_2 + Y_3
\]  
(34)
The expression for \(K_{p,j}\) is
\[
K_{p,j} = \exp \left( \sum_{n=1}^{6} B_{n,j} T_G^{-n-2} \right)
\]  
(35)
where the constant coefficients \(B_{n,j}\) are given in appendix B. The constants \(Y_C\), \(Y_H\), \(Y_O\), and \(Y_N\) in the four elemental balance equations are determined.
from equations (8) through (11) and the specification
of $\eta_{N,O}$, $\eta_{H,C}$, and $\phi$.

**Thermodynamic Relations and Properties**

The starting point for determining the thermodynamic properties of a mixture is an equation for the molar heat capacity at constant pressure of each chemical species expressed as

$$C_{p,k} = \sum_{n=1}^{6} A_{n,k} T_G^{n-2}$$

where $A_{n,k}$ is a set of constant coefficients given in appendix B for each species over a temperature range. In some cases, the specific heat capacity at constant pressure is useful and can be written as

$$c_{p,k} = \frac{C_{p,k}}{W_k}$$

In addition to this expression for the $k$th species, expressions for the specific heat capacity at constant pressure and the ratio of specific heats for water vapor are needed; that is,

$$c_{p,1} = \frac{\sum_{n=1}^{6} A_{n,1} T_G^{n-2}}{W_1}$$

and

$$\gamma_1 = \frac{c_{p,1}}{c_{p,1} - R_1}$$

Similar expressions for the carrier gas are also needed for computations of droplet growth and can be written as

$$c_{p,c} = \frac{\sum_{k=2}^{10} y_k^p C_{p,k}}{(1 - y_1^p) W_c}$$

and

$$\gamma_c = \frac{c_{p,c}}{c_{p,c} - R_c}$$

The specific enthalpy of the gaseous mixture can be obtained from the equation for $C_{p,k}$, integration, and summation over all species to give

$$h_G = \sum_{k=1}^{10} Y_k \left[ A_{1,k} \ln T_G + \sum_{n=2}^{6} \frac{A_{n,k} T_G^{n-1}}{n-1} + A_{7,k} \right]$$

where the additional constant coefficients $A_{7,k}$ include the standard state enthalpy of formation of each species at the reference temperature of 298.15 K. Numerical values of $A_{7,k}$ are given in appendix B. The units for $h_G$ are joules per kilogram of gaseous mixture. An expression for the specific enthalpy of water vapor in the mixture is also needed; that is,

$$h_{G,1} = \frac{A_{1,1} \ln T_G + \sum_{n=2}^{6} \frac{A_{n,1} T_G^{n-1}}{n-1} + A_{7,1}}{W_1}$$

The units of $h_{G,1}$ are joules per kilogram of water vapor.

The specific entropy of the gaseous mixture can also be obtained from the expression for $C_{p,k}$, integration, and summation over all species to give

$$s_G = \sum_{k=1}^{10} Y_k \left[ \frac{-A_{1,k}}{T_G} + (A_{2,k} - R') \ln T_G + \sum_{n=3}^{6} \frac{A_{n,k} T_G^{n-2}}{n-2} + A_{8,k} \right] - R \ln \rho_G$$

$$- R \ln \left( \frac{R}{p^o} \right) - R \sum_{k=1}^{10} Y_k \ln Y_k$$

where $A_{8,k}$ includes the standard state entropy of formation of species $k$ at the reference temperature of 298.15 K. Numerical values are given in appendix B. For some applications it is convenient to invert this equation to express $\rho_G$ as a function of $s_G$, $T_G$, and the set of mol numbers $Y_k$; that is,

$$\rho_G = \exp \left[ G - s_G - R \ln \left( \frac{R}{p^o} \right) - R \sum_{k=1}^{10} Y_k \ln Y_k \right]$$

where

$$G = \sum_{k=1}^{10} Y_k \left[ \frac{-A_{1,k}}{T_G} + (A_{2,k} - R') \ln T_G + \sum_{n=3}^{6} \frac{A_{n,k} T_G^{n-2}}{n-2} + A_{8,k} \right]$$

The equation of state for the gaseous mixture is

$$p = \rho_G R T_G$$

where $\bar{R} = \frac{R}{W}$ and $W = \left( \sum_{k=1}^{10} Y_k \right)^{-1}$.
Computations in the flow region where nucleation and droplet growth can occur require additional equations for the enthalpy and entropy of liquid water. An equation for the specific enthalpy of liquid water is obtained by starting with the specific enthalpy of evaporation of water at 1 atm and 373.15 K from reference 5; that is, $\Delta h^o_{L,G,373.15} = 2.2570 \times 10^6$ J/kg. Then using equation (43) at $T = 373.15$ K and the coefficients $A_{n,1}$ given in table B1(a), it follows that $h^o_{G,1,373.15} = -13.2933 \times 10^6$ J/kg. Also, the heat capacity of liquid water in the region of interest is essentially constant and is $c_L = 4.2 \times 10^3$ J/kg-K. Now since $h^o_L = h^o_{G,1,373.15} - h_{L,373.15}$ (48) and $\Delta h^o_{L,G,373.15} = h^o_{G,1,373.15} - h_{L,373.15}$ (49) it follows that the specific enthalpy of liquid water is $h_L = (4.2 \times 10^3)T - (17.1175 \times 10^6)$ (50) where the units of $h_L$ are joules per kilogram of liquid water.

The specific entropy of liquid water can be obtained in a similar way. The specific entropy of evaporation of water at 1 atm and 373.15 K is $\Delta s^o_{L,G,373.15} = \frac{\Delta h^o_{L,G,373.15}}{373.15} = 6.0485 \times 10^3$ J/kg-K. The entropy of water vapor at 1 atm and 373.15 K is obtained by using the equation $s^o_{G,1} = \left[ -A_{1,1}/T_G + A_{2,1} \ln T_G + \sum_{n=3}^{6} \frac{A_{n,1}T_G^{n-2}}{n-2} + A_{8,1} \right]/W_1$ (51) and the coefficients $A_{n,1}$ given in table B1(a) so that $s^o_{G,1,373.15} = 10.905 \times 10^3$ J/kg-K. Now since $s_L = s^o_L - s_{L,373.15} = (4.2 \times 10^3) \ln \frac{T}{373.15}$ (52) and $\Delta s^o_{L,G,373.15} = s^o_{G,1,373.15} - s_{L,373.15}$ (53) it follows that the specific entropy of liquid water is $s_L = 4.2 \times 10^3 \ln \frac{T}{T_G} - 20.0149 \times 10^3$ (54) where the units of $s_L$ are joules per kilogram of liquid water-kelvin.

An equation for the specific latent heat of evaporation of water at a temperature $T$ is also needed and is $L = h^o_{G,1} - h_L$ (55) It then follows from equation (50) that $L = h^o_{G,1} - 4.2 \times 10^3 T + 17.1175 \times 10^6$ (56) where $h^o_{G,1}$ is evaluated at $T$ and the units of $L$ are joules per kilogram of water.

The vapor pressure of water is also required. An expression for the vapor pressure of water over a flat surface as a function of temperature was obtained by fitting a curve to the tabulated values given in reference 5; that is, $p_\infty = \exp \left( 55.897 - \frac{6641.7}{T} - 4.4864 \ln T \right)$ (57) where $p_\infty$ has units of newtons per meter$^2$.

The surface tension of water is represented by the equation given by Young (ref. 4); that is, $\sigma = (82.27 + 75.612T_R - 256.889T^2_R + 95.928T_R) \times 10^{-3}$ (58) where $T_R = \frac{T}{T_c}$ with $T_c$ denoting the critical temperature of water. The units of $\sigma$ are newtons per meter.

Finally, the critical droplet radius is obtained from the well-known equation $r_* = \frac{2\sigma}{\rho_L R_i T_G \ln (p_1/p_\infty)}$ (59) where $p_1 = y_1 p$ and $r_*$ has units of meters. Droplets larger than $r_*$ tend to grow while droplets less than $r_*$ tend to evaporate.

**Nucleation Rate**

The nucleation rate equation given by Young can be modified to take into account a carrier gas by replacing the gas density in Young’s equation which is for pure steam by the density of water vapor. This follows from the argument that nucleation depends on collisions between water molecules in the gaseous phase and clusters of water molecules, both of which are proportional to the density of water vapor. Now since the density of water vapor in a gaseous mixture can be expressed as
the nucleation rate equation for a gaseous mixture that includes a carrier gas becomes

\[ J = \frac{q_c}{1 + Q} \left( \frac{y_1 m_1}{m} \right)^2 \left( \frac{2 \pi m_1^2}{\pi m_1^2} \right)^{1/2} \frac{\rho_G^2}{\rho_L} \times \exp \left( \frac{-4 \pi r^2 \sigma}{3 k T_G} \right) \]  

(61)

where \( q_c \) is the condensation coefficient which Young argued to be unity and

\[ Q = \frac{2(\gamma_1 - 1)}{\gamma_1 + 1} \frac{L}{R_1 T_G} \left( \frac{L}{R_1 T_G} - \frac{1}{2} \right) \]  

(62)

is the nonisothermal correction factor given by Kantrowitz in reference 6.

**Transport Properties**

The need to consider transport processes is limited to the flow region in which droplet growth can occur. The temperature in this region is relatively low so that viscosity and thermal conductivity data are required over a rather narrow temperature range. Furthermore, only seven chemical species (H\(_2\)O, CO\(_2\), CO, O\(_2\), H\(_2\), N\(_2\), and NO) are present in significant quantities at these temperatures.

The viscosity of the gaseous mixture can be obtained by starting with an expression for the viscosity of each species in the pure state and then applying a mixing rule. The viscosity of each pure species can be represented by a polynomial

\[ \mu_k = \sum_{n=1}^{4} D_{n,k} T_G^{n-1} \]  

(63)

where the constant coefficients for each species \( D_{n,k} \) are given in appendix B. The viscosity of the gaseous mixture containing these species can then be obtained from the expression (ref. 7)

\[ \mu = \sum_{k=1}^{10} \frac{\mu_k}{1 + \frac{10}{Y_{\ell} \phi_{k,\ell}}} \]  

(64)

where

\[ \phi_{k,\ell} = \left[ 1 + \left( \mu_k / \mu_\ell \right)^{1/2} (W_\ell / W_k)^{1/4} \right] \left( 8 [1 + (W_k / W_\ell)] \right)^{1/2} \]  

(65)

The thermal conductivity of the gaseous mixture can also be obtained by starting with a polynomial expression in terms of temperature for each pure species

\[ \lambda_k = \sum_{n=1}^{4} E_{n,k} T_G^{n-1} \]  

(66)

where the constant coefficients for each species \( E_{n,k} \) are given in appendix B. The thermal conductivity of the gaseous mixture can then be obtained by using the expression (ref. 8)

\[ \lambda = \frac{1}{2} \left[ W \sum_{k=1}^{10} Y_k \lambda_k + \frac{1}{W \sum_{k=1}^{10} Y_k / \lambda_k} \right] \]  

(67)

Finally, the equations for the mean free path, Prandtl number, and Knudsen number are

\[ \lambda = \frac{1.5 \mu \sqrt{RT_G}}{p} \]  

(68)

\[ \text{Pr} = \frac{c_p \mu}{\lambda} \]  

(69)

\[ \text{Kn} = \frac{\lambda}{2r} \]  

(70)

and

\[ \text{Kn*} = \frac{\lambda}{2r*} \]  

(71)

where the specific heat at constant pressure for water vapor is used to define the Prandtl number.

**Droplet Growth Rate**

The integrated droplet growth equation given by Young for droplets with \( z = \frac{t}{r} \geq 1.1 \) can be written as

\[ \frac{\theta^3}{\theta + 1} \ln \frac{z_{j+1} + \theta}{z_j + \theta} + \left( \frac{1}{\theta + 1} + \Omega \right) \ln \frac{z_{j+1} - 1}{z_j - 1} \]

\[ + \left( \Omega + 1 - \theta \right) (z_{j+1} - z_j) \]

\[ + \frac{1}{2} \left( z_{j+1}^2 - z_j^2 \right) = \Lambda \Delta t \]  

(72)

\(^1\) The constant of 1.5 used by Young is somewhat smaller than 3\( \sqrt{3/8} = 1.88 \) from simplified kinetic theory and somewhat larger than \( \sqrt{8/3} = 1.26 \) from a considerably more elaborate calculation.
This equation also applies in the case with a carrier gas present when \( \Omega \) is defined as

\[
\Omega = \frac{\xi (1 - \nu) \text{Kn}_*}{\text{Pr}_f (73)}
\]

where

\[
f = y_1 \left( \frac{m_1}{m} \right)^{1/2} + (1 - y_1) \left( \frac{m_c}{m} \right)^{1/2} \left[ \frac{(\gamma_c + 1) \gamma_c \text{c_p,c}}{\gamma_c + 1} \right] \alpha_c (74)
\]

The derivation of the factor \( f \) is given in appendix A.

Expressions for the other parameters \( \theta \), \( \Lambda \), \( \xi \), and \( \nu \) are

\[
\theta = 2\beta \text{Kn}_* (75)
\]

\[
\Lambda = \frac{\lambda [T_s(p_1) - T_G]}{L \rho L \tau^2} (76)
\]

\[
\xi = \sqrt{2\pi} \frac{2\gamma_1}{1.5} \frac{\text{Pr}_L}{\gamma_1 + 1} (77)
\]

and

\[
\nu = \frac{R_1 T_s(p_1)}{L} \left[ \alpha + \frac{1}{2} \frac{2 - q_c \gamma_1 + 1}{2 \gamma_1} \frac{\text{c_p1}_s T_s(p_1)}{L} \right] \frac{f}{y_1} \left( \frac{m}{m_1} \right)^{1/2} (78)
\]

The saturation temperature \( T_s(p_1) \) is based on the partial pressure of water vapor in the mixture, and \( \gamma_1 \) and \( \text{c_p1}_s \) are for water vapor. Also note that the expression for \( \nu \) contains a term +1/2 rather than -1/2 given by Young. This difference is believed to be due to a typographical error in Young's paper.

The integrated droplet growth equation given by Young for droplets with \( z = \frac{r}{r_*} < 1.1 \) is

\[
z_{j+1} = 1 + \frac{r_*}{r_{*,j+1}} (z_j - 1) \exp(\psi \Delta t)
- \frac{1}{\psi r_{*,j+1}} \frac{dr_*}{dt} \exp(\psi \Delta t - 1) (79)
\]

where the subscripts \( j \) and \( j+1 \) denote adjacent locations in the flow direction and \( \Delta t \) is the corresponding time interval. This equation applies to the case with a carrier gas present when \( \psi \) is defined as

\[
\psi = \frac{f_{p} \gamma_1 + 1}{2 \gamma_1} \frac{1}{\text{Pr}_L \gamma_1 - \nu} \frac{1}{[T_s(p_1) - T_G]} (80)
\]

where \( f \) and \( \nu \) are given by equations (74) and (78).

The temperature of the liquid droplets is also required. Young has developed the following equation for \( T_L \) which depends on \( T_G \), \( T_s(p_1) \), \( r_* \), and \( r \):

\[
T_L = T_G + \frac{(1 - r_*/r) [T_s(p_1) - T_G]}{1 - \nu \delta} (81)
\]

where

\[
\delta = \frac{\xi \text{Kn}_*/\text{Pr}_f}{1 + 2\beta \text{Kn}_* + \xi \text{Kn}_*/\text{Pr}_f} (82)
\]

and \( \xi \) is given by equation (77). It should be noted that equations (72) and (79) imply that the droplets retain their identity as they grow and do not agglomerate.

**Entropy Production Equation**

The approach taken by Young was to replace the differential form of the momentum equation with an expression for the increase in entropy due to water condensation. The differential change in entropy given by Young due to liquid water formation also applies to a mixture containing a carrier gas if expressed as

\[
dx = \{ L - \text{c_p1}_s [T_s(p_1) - T_G] \} \left[ \frac{1}{T_G} - \frac{1}{T_s(p_1)} \right] dw (83)
\]

where the saturation temperature \( T_s(p_1) \) is based on the partial pressure of water vapor in the mixture \( p_1 \), the specific heat of water vapor \( \text{c_p1}_s \) is used, and the latent heat \( L \) is evaluated at \( T_s(p_1) \).

**Numerical Solution**

The foregoing set of equations can now be used to obtain a numerical solution for the adiabatic expansion of combustion products with nucleation and water droplet growth. The numerical solution is divided into three successive parts as indicated in figure 1.

**Part 1 of Numerical Solution—Stagnation Conditions and Mass Flow Rate**

Part 1 of the numerical solution involves the specification of the feed conditions to the combustor, the determination of the adiabatic flame temperature,
the computation of the chemical composition of combustion products in the combustor at a stagnation temperature that allows for heat loss, and the determination of the mass flow rate.

**Feed conditions.** The first step in computing the stagnation conditions in the combustion chamber is to define the hydrogen-to-carbon ratio \( n_{H/C} \) and the number of atoms of carbon in a molecule of fuel \( n \); the nitrogen-to-oxygen ratio \( n_{N/O} \); the fuel equivalence ratio \( \phi \); the feed temperatures of the fuel, air, and oxygen to the combustion chamber, \( T_F, \ T_{air}, \ \) and \( T_{O_2}; \) and the pressure in the combustion chamber, \( p_o. \) The selection of the fuel and the feed temperatures determines the specific enthalpy of the fuel \( h_F \) as well as \( h_{O_2} \) and \( h_{N_2}. \)

The feed composition in terms of \( Y_F, \ Y_{O_2}, \) and \( Y_{N_2} \) may be found by first solving equation (16) for \( Y_{O_2} \) and then equations (13) and (14) for \( Y_F \) and \( Y_{N_2}. \) The specific total enthalpy of the feed can then be computed from the equation

\[
\hat{h}_o = \hat{Y}_F h_F + \hat{Y}_{O_2} h_{O_2} + \hat{Y}_{N_2} h_{N_2} \quad (84)
\]

**Adiabatic flame temperature.** The adiabatic flame temperature and the corresponding chemical equilibrium composition are determined by the simultaneous solution of the elemental balance expressions (eqs. (31) through (34)), the equilibrium relations (eqs. (25) through (30)), and the specific enthalpy of the mixture with \( h_G = \hat{h}_o \) (eq. (42)). Auxiliary equations for the equilibrium constants (eq. (35)) and the elemental constants (eqs. (8) through (11)) are also required. A solution is obtained by iteration for flame temperature \( T_f. \)

Equations (25) through (34) are solved simultaneously by the method of successive approximations. For flame temperatures up to 2000 K and fuel lean conditions, the major chemical species are \( H_2O, \ CO_2, \ CO, \ O_2, \) and \( N_2. \) Although significant amounts of the other five species may be present, estimates for the five major species noted can be used for a reasonable initial approximation. If an initial estimate of the ratio of the number of mols of \( CO_2 \) to total mols of carbon in the mixture \( a_{2,C} \) is made, and equations (31) through (34) are used with \( Y_5 = Y_7 = Y_8 = Y_9 = Y_{10} = 0, \) it follows that a first approximation for the mol numbers of the five major species and the total mol number of the mixture is

\[
Y_1 = \frac{Y_H}{2} \quad (85)
\]

\[
Y_2 = a_{2,C} Y_C \quad (86)
\]

\[
Y_3 = Y_C - Y_2 \quad (87)
\]

\[
Y_4 = \frac{Y_O - Y_C - Y_1 - Y_2}{2} \quad (88)
\]

\[
Y_5 = \frac{Y_N}{2} \quad (89)
\]

\[
Y = \frac{Y_H + Y_O + Y_N + Y_C - Y_1 - Y_2}{2} \quad (90)
\]

If equations (25) through (30) are now also used along with equations (31) through (34), the following set of equations can be used to determine the chemical composition for all 10 species through a series of successive approximations:

\[
Y_3 = Y_2 \left( \frac{K_{p,2} Y p^0}{Y_4} \right)^{1/2} \quad (91)
\]

\[
Y_5 = \frac{Y_1 Y_3}{K_{p,1} Y_2} \quad (92)
\]

\[
Y_{10} = (K_{p,6} Y_4 Y_6)^{1/2} \quad (93)
\]

\[
Y_6 = \frac{Y_N - Y_{10}}{2} \quad (94)
\]

\[
Y_9 = (K_{p,3} Y_4 Y_5)^{1/2} \quad (95)
\]

\[
Y_7 = \left( \frac{K_{p,4} Y_3 Y_5 p^0}{p} \right)^{1/2} \quad (96)
\]

\[
Y_8 = \left( \frac{K_{p,5} Y_4 Y_5 p^0}{p} \right)^{1/2} \quad (97)
\]

\[
Y_1 = \frac{Y_H - 2Y_5 - Y_7 - Y_9}{2} \quad (98)
\]

\[
Y_2 = Y_C - Y_3 \quad (99)
\]

\[
Y_4 = \frac{Y_O - Y_C - Y_1 - Y_2 - Y_8 - Y_9 - Y_{10}}{2} \quad (100)
\]

\[
Y_4 = \frac{\left[ Y_4 + Y_O - Y_C - Y_1 - Y_2 - Y_8 - Y_9 - Y_{10} \right]}{2} \quad (101)
\]

\[
Y = \sum_{k=1}^{10} Y_k \quad (102)
\]

It should be noted that equation (100) led to step-to-step oscillations in the solution for higher temperatures. In order to avoid this problem, equation (100) was rewritten as equation (101) to provide some damping. The numerical value of \( Y_4 \) computed in the previous iteration step is used in the
right-hand side of equation (101). After a stable solution for the mol numbers \( Y_k \) for \( k = 1 \) to 10 is achieved for a given temperature, the corresponding specific enthalpy of the mixture \( h_G \) is computed from equation (42) and compared with the known total feed enthalpy \( \hat{h}_o \). If \( h_G \) turns out to be less than \( \hat{h}_o \), the next approximation to the temperature is increased and if greater, decreased. The final solution for the adiabatic flame temperature is obtained when \( h_G = \hat{h}_o \).

**Stagnation conditions.** This study does not attempt to compute the heat loss from the combustor, however, the influence of an arbitrary heat loss has been taken into account by assuming a stagnation temperature \( T_o \) that is somewhat less than the adiabatic flame temperature. The chemical composition at \( T_o \) can then be determined directly by using the scheme just presented. The stagnation enthalpy \( \hat{h}_o \) and the stagnation entropy \( s_o \) are then computed from equations (42) and (44).

**Mass flow rate.** The mass flow rate is determined in the following way. An isentropic expansion from the combustion chamber is assumed and the mass flux \( \rho_G U \) computed along the nozzle to find the maximum value of \( \rho_G U \). Since \( h_o \) and \( s \) are constant along the expansion, all flow properties can be computed at a selected temperature that is less than \( T_o \). This computation uses an iteration on the pressure.

The choice of an approximate pressure, say \( p' \), at the selected temperature \( T \) allows the computation of a chemical equilibrium composition. This in turn permits the direct computation of the density by equations (45) and (46). The resulting density \( \rho_G \) and composition can then be used in equation (47) to compute a pressure \( p \). A comparison between \( p' \) and \( p \) indicates the need to continue the iteration process. When the iteration scheme gives a value of \( p \) sufficiently close to \( p' \), the pressure is determined.

The flow velocity \( U \) at this state is then computed from equation (3) with no liquid water present; that is,

\[
U = \left[2(h_o - h_G)\right]^{1/2} \tag{103}
\]

The mass flux \( \rho_G U \) can therefore be found at selected temperatures corresponding to successive locations along the nozzle. The maximum mass flux \( \rho_G U^* \) occurs at the nozzle throat where the nozzle cross-sectional area is \( A^* \). The mass flow rate is given then by equation (1) with no liquid water present; that is,

\[
\dot{m} = \rho_G U^* A^* \tag{104}
\]

This is the mass flow rate of the total mixture along the entire length of the nozzle including the downstream conditions with liquid water formation.

**Part 2 of Numerical Solution—Isentropic Expansion and Beginning of Nucleation**

Part 2 of the numerical solution involves the computation from the nozzle throat to the beginning of nucleation. It is assumed that the flow expands isentropically and remains in chemical equilibrium with no liquid water formation. At each step along the nozzle the nucleation rate is computed to determine at what point to include nucleation and liquid droplet growth.

**Isentropic expansion.** The isentropic expansion from the nozzle throat is obtained by noting that \( s = s_o \) and \( h_o = \) Constant along the nozzle. Starting at \( T^* \) and \( p^* \), a temperature \( T \) less than \( T^* \) is selected. The pressure is then determined by iteration and the values of \( \rho_G \), \( h_G \), and \( U \) computed by the scheme given in part 1 of the numerical solution. The area of the nozzle can then be computed from equation (1) with \( w = 0 \); that is,

\[
A = \frac{\dot{m}}{\rho_G U} \tag{105}
\]

The location of this particular point along the nozzle is then determined by using the relationship that defines the cross-sectional area as a function of nozzle position \( x \). This computation in general requires the inversion of the expression of \( A \) as a function of \( x \) or an iteration to find \( x \) for a given value of \( A \).

**Beginning of nucleation.** The nucleation rate \( J \) is computed from equation (61) where the appropriate values of \( \gamma_1 \), \( h_{G,1} \), \( L \), \( p_\infty \), \( \sigma \), and \( r_* \) are obtained from equations (39), (43), (56), (57), (58), and (59), respectively. This computed value of \( J \) is compared with a value \( J_{\text{min}} \) chosen such that a smaller value of \( J \) does not generate enough nuclei to affect the flow throughout the nozzle. Successively lower values of \( T \) are taken along the nozzle until \( J \geq J_{\text{min}} \). At this point and downstream, the number of nuclei formed is taken into account, and the droplet growth equations are used.

**Part 3 of Numerical Solution—Nucleation and Droplet Growth**

Part 3 of the numerical solution deals with the computation of the nucleation process to determine the number of droplets formed at successive locations along the nozzle and the subsequent growth of the liquid water droplets. The starting point is
the condition determined in part 2 of the numerical solution where the nucleation rate just exceeds a threshold value. The index $j$ is used to denote a location along the nozzle, starting with $j = 1$ at the beginning of part 3 (fig. 1), where all properties are known. This also includes the chemical composition of the gas phase. Since essentially no further chemical reaction takes place once nucleation and droplet growth occur, the only change in the gaseous mixture composition is due to the formation of liquid water. The appropriate expressions for $Y_1, Y_k$ (with $k = 2, 3, \ldots, 6$), and $W$ at any position along the nozzle are thus given by equations (17), (18), and (19). The computation proceeds in general at location $j$ where $T, p, \rho, y_1, U, s, A, x$, and $w$ are known. All other required parameters at $j$ can be computed from this set. Note also that all properties at $j = 1$ are known.

**Computational steps along nozzle and nozzle area.**

The first step in the determination of properties at $j+1$ is to increase $x_j$ by $\Delta x$ so that

$$x_{j+1} = x_j + \Delta x$$

(106)

and to determine the nozzle area from the expression

$$A_{j+1} = f(x_{j+1})$$

(107)

An iteration scheme based on adjusting $T_{j+1}$ as the primary variable is then used to find a solution for all properties at $j+1$. Secondary iterations within the loop for $T_{j+1}$ are also required.

**Gas temperature.** The first approximation for the gas temperature at $j+1$ is $T_{j+1} = T_j$ and it is further assumed that initially $p_{j+1} = p_j$ and $U_{j+1} = U_j$. The value of $T_{j+1}$ is adjusted as a result of each iteration step, and $T_{j+1}$ is defined at this point and used to test for convergence.

**Formation of new droplets.** The nucleation rate is assumed to vary as $\exp(kx)$ as suggested by Young, so that the appropriate mean nucleation rate expressed in terms of new droplets formed per unit volume per unit time at $j$ over the volume from $j-\frac{1}{2}$ to $j+\frac{1}{2}$ is the logarithmic mean value; that is,

$$J_j = \frac{J_{j+1/2} - J_{j-1/2}}{\ln J_{j+1/2} - J_{j-1/2}}$$

(108)

where $J_{j+1/2}$ and $J_{j-1/2}$ are computed from equation (61). The number of new droplets formed at $j$ per unit mass of total mixture is therefore

$$\Delta n^i_{j+1} = \frac{J_j A_j \Delta x}{m}$$

(109)

Although the droplet size of this set formed at $j$ will change as it moves downstream, the number $\Delta n^i$ in the set will remain constant unless some of the droplets evaporate. Agglomeration of droplets is not taken into account in these calculations. Note that a set of droplets characterized by a single index $i$ will be formed at successive positions along the nozzle. At any position $j$, there could be as many as $j$ sets of droplets each with a radius of $r_j^i$ in the amount of $\Delta n^i$ droplets per mass of total mixture.

**Droplet growth.** The computation of the droplet radius at $j + 1$ employs equation (72) and appropriate auxiliary equations for $z_j = r_j^i / r_{*,j} \geq 1.1$ or equation (79) and auxiliary equations for $z_j = r_j^i / r_{*,j} < 1.1$. Either equation requires the time interval that corresponds to the step size $\Delta z$; that is,

$$\Delta t = \frac{2\Delta z}{U_j + U_{j+1}}$$

(110)

It should be noted that the derivation of equation (72) by Young used the argument that the properties of the gaseous mixture do not change between locations $j$ and $j+1$. This means that equation (72) can be used to compute $r_{j+1}^i$ with only knowledge of properties at $j$. The solution for droplets in the size range of $z_j \geq 1.1$ can be obtained by rewriting equation (72) as

$$F = Z_{j+1} - Z_j - \Delta \Delta t$$

(111)

where

$$Z_j = \frac{\theta^3}{\theta + 1} \ln \left(z_j + \theta\right) + \left(\frac{1}{\theta + 1} + \Omega\right) \ln \left(z_j - 1\right) + \left(\Omega + 1 - \theta\right) z_j^i + \frac{1}{2} z_j^{2i}$$

(112)

The application of a Newton iteration scheme requires $dF/dz_{j+1}^i$ which is

$$\frac{dF}{dz_{j+1}^i} = \frac{\theta^3}{(\theta + 1) \left(z_{j+1}^i + \theta\right)} + \frac{1 + \Omega(\theta + 1)}{(\theta + 1) \left(z_{j+1}^i - 1\right)} + z_{j+1}^i + \left(\Omega + 1 - \theta\right)$$

(113)
so that the \( n \)th iteration gives

\[
z^i_{j+1}(n+1) = z^i_{j+1}(n) - \frac{F(n)}{dF(n)/dz^i_{j+1}}
\]

(114)

The droplet radius at \( j + 1 \) when the iteration has been completed is

\[
r^i_{j+1} = z^i_{j+1}r^*_i
\]

(115)

Equation (79) is used for the smaller droplets when \( z^i_j < 1.1 \). In this case, the solution for \( z^i_{j+1} \) requires knowledge of \( r^*_i_{j+1} \) as well as properties at \( j \). An expression for \( dr_i/r \) is

\[
\frac{dr_i}{dt} = r_i d_r - r_{i-1}
\]

(116)

It should be noted that Young's analysis assumed that \( \psi \) in equation (79) and given by equation (80) is a constant over the interval from \( j \) to \( j + 1 \) but that the solution for \( z^i_{j+1} \) from equation (79) requires knowledge of \( r^*_i_{j+1} \). This means that the computation of \( r^i_{j+1} \) when \( z^i_j < 1.1 \) must use successively revised values of \( r^*_i_{j+1} \) that correspond to the successively revised values of \( T^i_{j+1} \).

**Mass of liquid water.** The mass of liquid water droplets that have a radius \( r^i_{j+1} \) is \( 4\pi r^i_{j+1} \cdot 3 \) so that the mass of liquid water per total mass of mixture for all droplets of type \( i \) at \( j + 1 \) is

\[
w^i_{j+1} = \frac{4\pi r^i_{j+1}}{3} \Delta n^i
\]

(117)

The total mass of liquid water per total mass of mixture at \( j + 1 \) is

\[
w_{j+1} = \sum_{i=1}^{j+1} w^i_{j+1}
\]

(118)

**Enthalpy and entropy of liquid water.** Now in order to compute the enthalpy and entropy of the liquid water in the mixture, the temperature of the liquid droplets of each type \( i \), \( T^i_{L,j+1} \), must be determined from equation (81). The enthalpy of the liquid at \( j + 1 \) can then be computed from equation (50) using \( T^i_{L,j+1} \) to get \( h^i_{L,j+1} \) and \( h_{L,j+1} \) is obtained from

\[
h_{L,j+1} = \sum_i w^i_{j+1} h^i_{L,j+1}
\]

(119)

The entropy of the liquid water is computed likewise from equation (54) using \( T^i_{L,j+1} \) to get \( s^i_{L,j+1} \) so that

\[
s^i_{L,j+1} = \sum_i w^i_{j+1} s^i_{L,j+1}
\]

(120)

**Entropy production.** The next step is to compute the entropy increase due to liquid water formation by equation (83) with \( ds \) replaced by \( \Delta s \) and \( dw \) by \( \Delta w \) where

\[
\Delta w = w_{j+1} - w_j
\]

(121)

The temperature and pressure used to determine properties in equation (83) are

\[
T_G = \frac{T_G^{j+1} + T_G^j}{2}
\]

(122)

and

\[
p = \frac{p_{j+1} + p_j}{2}
\]

(123)

The value of \( L \) is evaluated at \( T_s(p_1) \), and \( c_p,1 \) is a very weak function of temperature so it can be evaluated at \( T_G \). The entropy of the total mixture can therefore be written as

\[
s_{j+1} = s_j + \Delta s_j
\]

(124)

**Two-level iteration to determine \( T_G \).** At this point, the iteration scheme proceeds with an attempt to find a value of \( T^i_{G,j+1} \) that satisfies the independent computation of \( s^i_{G,j+1} \). Simultaneously an iteration is carried out to determine a value of \( p^i_{G,j+1} \) by using the known area at \( j + 1 \), \( A^i_{j+1} \), and the mass flow equation. The two iteration schemes are as follows:

1. Iteration using entropy of gas mixture: One equation for the entropy of the gaseous mixture can be written as

\[
s^G_{G,j+1} = \frac{s^i_{j+1} - s^i_{L,j+1}}{1 - w^i_{j+1}}
\]

(125)

where all terms on the right-hand side have been computed. A second and independent equation is given by equation (44) from which \( s^G_{G,j+1} \) is computed using \( p^i_{G,j+1} \) and \( T^i_{G,j+1} \). The iteration at this point assumes that \( s''^G_{G,j+1} \) is correct, fixes \( p^i_{G,j+1} \), and adjusts \( T^i_{G,j+1} \) until

\[
(s''^G - s^G)_{j+1} < \epsilon
\]

(126)

where

\[
c_p = \sum_k Y_k C_{p,k}
\]
If this condition is not satisfied, the value of \( T_{G,j+1} \) is revised according to the expression

\[
T_{j+1}(n+1) = T_{j+1}(n) \exp \left[ \frac{(s_G^0 - s_G)^{j+1}}{c_p - \bar{R}} \right]
\]  
(127)

where \( T_{j+1} = T_{G,j+1} \).

2. Iteration using nozzle area: After the condition in equation (126) is satisfied, \( T_{j+1} \) is used in equation (42) to compute \( h'_{G,j+1} \). The specific enthalpy of the total mixture is then computed from

\[
h'_{j+1} = (1 - w_{j+1})h'_{G,j+1} + h_{L,j+1}
\]  
(128)

The value of \( U \) at \( j + 1 \) is then computed from

\[
U_{j+1} = \left[ 2(h_0 - h'_{j+1}) \right]^{1/2}
\]  
(129)

A value for the cross-sectional area of the nozzle at \( j + 1 \) is then computed from equation (1); that is,

\[
A^h_{j+1} = \frac{\dot{m}(1 - w_{j+1})}{\rho G_{j+1} U_{j+1}}
\]  
(130)

and compared with the known area \( A_{j+1} \) by using the expression

\[
\frac{|A^h_{j+1} - A_{j+1}|}{A_{j+1}} < \epsilon
\]  
(131)

If this condition is not satisfied, the density of the gaseous mixture is computed from the equation

\[
\rho G_{j+1} = \frac{\dot{m}(1 - w_{j+1})}{A_{j+1} U_{j+1}}
\]  
(132)

and the iteration process returns to the computation of \( s'_{G,j+1} \) using this value of \( \rho G_{j+1} \) and the last computed value of \( T_{G,j+1} \). The iteration continues until the condition in equation (131) is satisfied after which \( p_{j+1} \) is computed from equation (47).

The iteration for \( T_{j+1} \) now includes the test that

\[
\frac{|T_{j+1} - T'_{j+1}|}{T'_{j+1}} < \epsilon
\]  
(133)

If this test is not satisfied, the iteration process returns to the beginning (gas temperature) with \( T_{j+1} \) equal to its last value. When this test is satisfied \( j \) is increased by 1 with \( z \) increased by \( \Delta z \) (step along the nozzle and nozzle area). This process is continued to the nozzle exit.

### Computer Program

A computer program, FIRACON, has been developed to solve the finite-rate condensation problem using the solution technique described in the present paper. The program is written in FORTRAN Version 5 language for the CDC® CYBER series digital computer system, model 860, with Network Operating System. The program requires approximately 134000 octal locations of core storage, and a typical case requires approximately 200 seconds of central processing unit time. Program input requirements and operating instructions are summarized in the code which is included as appendix C.

### Application of Computation Scheme to Langley 8-Foot High-Temperature Tunnel

The numerical scheme presented in the previous section has been applied to the Langley 8-Foot High-Temperature Tunnel (8'HTT) for a selected set of operating conditions. The 8'HTT is a large combustion-driven wind tunnel that burns natural gas (mostly methane) and air at high pressure in a cylindrical combustor approximately 1 m in diameter and 7 m long. The combustion products which are primarily \( H_2O, CO_2, O_2, \) and \( N_2 \) with smaller amounts of \( H_2, CO, H, O, NO, \) and \( OH \), expand as the test gas from the combustor through an axisymmetric conical-contoured nozzle to a test section with a diameter of approximately 2.4 m (8 ft). The nozzle length is approximately 16 m.

Numerical results are presented for five cases that correspond to operating conditions of the 8'HTT. The combustor temperature, pressure, and fuel equivalence ratio for these cases are given in table 1. The mole fraction of water vapor formed in the combustor for each case is also listed. The conditions for case 1 were established by selecting a combustor pressure of 50 bars and an equivalence ratio that gave a flame temperature of 2000 K. An arbitrary heat loss from the combustor was included by selecting a stagnation temperature that was 100 K less than the flame temperature, that is, 1900 K. Conditions for case 2 were obtained likewise but for a pressure of 250 bars. Conditions for case 3 and case 4 were obtained in a similar fashion but for a temperature of 1600 K. Conditions for case 5 were obtained for \( T_0 = 1900 \) K and \( p_0 = 250 \) bars, the same as for case 2, but for additional oxygen and an equivalence ratio that gave a mole fraction of 0.21 for \( O_2 \) in the test stream. Note that the mole fraction of water vapor in the combustor is 0.154 for cases 1 and 2, 0.122 for cases 3 and 4, and 0.156 for case 5. The values used for the coefficients \( q_c, \alpha, \alpha_c, \beta, \) and \( J_{min} \) are also indicated in table 1.
The nozzle contour of the 8'HTT was used to define the variation of the cross-sectional area along the nozzle. Tabulated values of selected wall coordinates \( r_w \) and computed boundary-layer-displacement thicknesses \( \delta^* \) are tabulated in table 2. The numerical values of \( \delta^* \) were computed from a modified version of the boundary-layer scheme presented in reference 9. The inviscid nozzle contour that was used for the quasi-one-dimensional computation was obtained by subtracting \( \delta^* \) from \( r_w \) at each location of \( x \) listed in table 2. A spline fitting scheme for intermediate locations was applied.

The computed results for case 1 with \( T_0 = 1900 \text{ K}, p_0 = 50 \text{ bars}, \phi = 0.798 \) are shown in figure 2. Figure 2(a) shows the difference between saturation temperature and the gas temperature \( T_s(p_1) - T_G \), the nucleation rate \( J \), and the mass fraction of liquid water formed. The temperature difference \( T_s(p_1) - T_G \) is a measure of the degree of supercooling and provides the driving force for nucleation and droplet growth. The test gas first becomes saturated with respect to water vapor when \( T_s(p_1) = T_G \). This occurs at a nozzle location just beyond \( x = 6.5 \text{ m} \). The temperature difference \( T_s(p_1) - T_G \) then increases to a maximum value of about 45 K at approximately \( x = 12 \text{ m} \) after which \( T_s(p_1) - T_G \) decreases rapidly. Figure 2(a) also shows that the nucleation rate \( J \) rises exponentially and then decreases rapidly as \( T_s(p_1) - T_G \) decreases. The mass fraction of liquid water begins to be significant at about the point of the maximum nucleation rate. The liquid water then continues to increase to the end of the nozzle.

The formation of liquid water under these conditions results in entropy production as shown in figure 2(b). The flow is isentropic until liquid water is formed. The rather rapid formation of liquid water initially causes a correspondingly rapid increase in entropy followed by a more gradual increase as the rate of liquid water formation decreases. This trend is to be expected from equation (83).

The entropy production due to liquid water formation affects the static pressure distribution along the nozzle as shown in figure 2(c). The dashed curve indicates the pressure distribution for an isentropic expansion if no liquid water was formed. The solid curve represents the computation that takes into account nucleation and droplet growth. Note that the process follows the isentropic (dashed) curve to just beyond a nozzle location of \( x = 12 \text{ m} \). At this point the static pressure decreases less rapidly than indicated by the isentropic process. The pressure then increases somewhat to a maximum value after which it decreases in parallel to the isentropic curve. This departure from the isentropic pressure distribution is due to the transfer of heat to the gaseous phase from the liquid droplets. At the nozzle exit the static pressure for case 1 with condensation is approximately 25 percent larger than that computed for isentropic flow without condensation.

The growth history of the liquid droplets is shown in figure 2(d). The single curve labeled \( r_* \) is the critical droplet radius computed from equation (59). Droplets that are larger than \( r_* \) grow, whereas those less than \( r_* \) evaporate. The series of curves indicates the growth of the sets of droplets that were initially formed at successive stations along the nozzle. Droplet sets that are formed but eventually evaporate are also shown. Only a portion of the numerous sets of droplets formed is included in figure 2(d).

The computed results for case 2 are given in figure 3 and are also for \( T_0 = 1900 \text{ K} \) but at a higher pressure of \( p_0 = 250 \text{ bars} \). Figure 3(a) shows that \( T_s(p_1) - T_G \) increases earlier and more rapidly for \( p_0 = 250 \text{ bars} \) than shown in figure 2(a) for \( p_0 = 50 \text{ bars} \). Case 2 shows a slightly lower maximum value of \( T_s(p_1) - T_G \) and also an earlier and more rapid decrease in \( T_s(p_1) - T_G \) along the nozzle. The nucleation process also begins further upstream for the higher pressure case and displays a more rapid increase and decay. The maximum nucleation rate also achieves a higher value for the higher pressure case. Note also that liquid water forms earlier and more rapidly for \( p_0 = 250 \text{ bars} \) compared with \( p_0 = 50 \text{ bars} \). The total amount of liquid water formed for the higher pressure case is approximately 60 percent more than for case 1.

Figure 3(b) shows the entropy production for case 2. A comparison with figure 2(b) shows that the entropy rises more rapidly for the higher pressure case. This is due to the more rapid formation of liquid water. The entropy also attains a nearly level value, but the total entropy increase is less for \( p_0 = 250 \text{ bars} \) than for \( p_0 = 50 \text{ bars} \). This smaller entropy increase is a direct result of the earlier collapse of the supercooling.

The static pressure distribution along the nozzle for case 2 is shown in figure 3(c). A comparison of figure 3(c) with figure 2(c) shows that the higher pressure case results in an earlier and more pronounced departure from an isentropic process. The static pressure at the nozzle exit for case 2 with condensation is approximately 33 percent larger than that computed for isentropic flow without condensation.

The droplet growth history for case 2 is shown in figure 3(d). The results are similar to case 1 except the liquid droplets form earlier, grow more rapidly, and attain larger sizes.

The results from case 3 are included to indicate the effect of a lower temperature and are
presented in figure 4. A comparison of these results for $T_o = 1600$ K with those for case 1 with $T_o = 1900$ K shows that the lower temperature case results in an earlier onset of nucleation and condensation. More liquid water is formed even though the total water vapor initially present in the test gas is less for case 3 than for case 1. The static pressure distribution plotted in figure 4(c) for case 3 shows a more pronounced departure from an isentropic process than was noted for either case 1 or case 2. The static pressure at the nozzle exit computed for flow with condensation is approximately 57 percent greater than that computed for isentropic flow without condensation as shown in figure 4(c). The results for case 4 with $T_o = 1600$ K, $p_o = 250$ bars, and $\phi = 0.62$ are presented in figure 5 and can be compared with the other cases. The results for case 5, which are for additional oxygen to give a test stream containing 0.21 mole fraction of O$_2$, are presented in figure 6. A comparison of these results with case 2 (fig. 3), which is also for 1900 K and 250 bars, shows very little difference. The somewhat different composition of the combustion products produces an expansion that gives a water vapor saturation point for case 5 that is slightly downstream of that for case 2. All the curves shown in figure 6 for case 5 are therefore shifted slightly downstream relative to case 2.

The purpose of presenting these five cases is to show a set of results that indicates the magnitude and the general trends for the effects of water condensation in the Langley 8-Foot High-Temperature Tunnel (8'HTT). The numerical values used for the coefficients $q_c = 1$, $\alpha = 8$, and $\beta = 2$ are those suggested by Young for pure steam. The value of $\alpha_c = 1$ was used to account for the non-condensable carrier gas. Computations with values of $q_c < 1$ were also carried out but are not included, and they give similar but less pronounced results. The value of $J_{\min}$ used was adjusted until a smaller value had no effect. More appropriate values of the empirical coefficients $\alpha, \alpha_c$, and $\beta$ might be determined from experimental data. These data would include static pressure distributions along the nozzle and water droplet size and concentration measurements for supersonic expanding flow of combustion products.

### Concluding Remarks

An analysis and numerical scheme has been developed to treat the supersonic expansion of combustion products that takes into account nucleation and finite-rate growth of liquid water droplets. The scheme has two limitations: the flow is assumed to be quasi-one-dimensional and empirical coefficients are required. This analysis has specific application to the computation of flow properties in combustion-heated wind tunnels such as the Langley 8-Foot High-Temperature Tunnel (8'HTT).

Sample computations included in this paper are based on the nozzle contour of the Langley 8'HTT. The numerical values used for the empirical coefficients in these calculations are those suggested by Young for pure steam. The results indicate that the free-stream static pressure can be significantly greater than that computed for isentropic nozzle flow without condensation. The computed entropy production also indicates a loss in total pressure relative to isentropic flow. These computed results suggest that this scheme can be used as a tool to interpret the calibration and flow measurements in combustion-heated wind tunnels. A calibration of combustion-heated wind tunnels must take the effects of water condensation into account.

NASA Langley Research Center
Hampton, VA 23665-5225
July 29, 1988
Appendix A
Extension of Young's Droplet Growth Equations To Include Carrier Gas

This appendix extends the droplet growth equations given by Young in references 3 and 4 for pure steam to the case of water vapor in the presence of a carrier gas. The carrier gas takes into account the noncondensable gases contained in the products of combustion. The equations for mass and heat transfer given by Young for water vapor are used and the appropriate equations for a carrier gas are introduced.

Young's analysis begins by writing the equations for the free molecular transport regime and for the continuum transport regime and then combines these equations to find droplet growth equations that apply to the intermediate regime. Young postulates that the transport processes can be described by free molecular transport from the droplet surface out to a distance of a few mean free paths and by continuum transport beyond. The interface between these two regimes is taken to be at \( r + f_\ell \) where \( f_\ell \) is a parameter of the order of unity. Young has chosen a value of \( f_\ell = 2 \). At this interface the pressure and temperature are denoted by \( P_i \) and \( T_i \).

**Free Molecular Transport Regime (Kn \( \gg 1 \))**

The mass transfer equation for water in the free molecular regime from the droplet surface out to the interface given by Young is

\[
\frac{dM}{dt} = 2q_c \frac{4\pi r^2}{2 - q_c} \left[ \frac{P_{1,i}}{2\pi R_1 T_i} - \frac{q_c p_s(T_L, r)}{2\gamma_1 \sqrt{2\pi R_1 T_L}} \right]
\]

and the energy transfer equation in this regime is

\[
\left( L \frac{dM}{dt} \right) = 4\pi r^2 \frac{P_{1,i}}{2\pi R_1 T_i} \left[ \frac{1}{2\gamma_1} \right] c_{p,1} (T_L - T_i)
\]

The mass transfer equation is appropriate as it stands with \( P_{1,i} \) representing the partial pressure of water vapor at the interface and \( R_1 = R/W_1 \). Equation (A2), however, includes only the energy transfer due to water. An additional equation is required to account for the parallel energy transfer due to the carrier gas. The equation for the energy transfer due to the carrier gas has the same form as equation (A2); that is,

\[
\left( L \frac{dM}{dt} \right)_c = 4\pi r^2 \frac{p_{c,i}}{2\pi R_c T_i} \left[ \frac{1}{2\gamma_c} \right] c_{p,c} \alpha_c (T_L - T_i)
\]

where \( p_{c,i} \) is the partial pressure of the carrier gas at the interface, \( R_c = R/W_c \), and \( \alpha_c \) is the thermal accommodation coefficient for the interaction between the carrier gas and a water droplet. It should be noted that the thermal accommodation coefficient for water vapor with a water droplet was assumed to be unity by Young as is implied by equation (A2). The total energy transfer in the free molecular regime due to both water vapor and carrier gas is then the sum of equations (A2) and (A3); that is,

\[
L \frac{dM}{dt} = 4\pi r^2 \frac{p_i}{2\pi R T_i} \left[ \frac{1}{2\gamma_1} \right] c_{p,1} (T_L - T_i) f
\]

where

\[
f = y_1 \left( \frac{m_1}{m} \right)^{1/2} + (1 - y_1) \left( \frac{m_c}{m} \right)^{1/2} \left[ \frac{(\gamma_c + 1) \gamma_1 c_{p,c}}{(\gamma_1 + 1) \gamma_c c_{p,1}} \right] \alpha_c
\]

This expression for \( f \) is equation (74). Note that \( f \) reduces to unity for the pure steam case.

**Continuum Transport Regime (Kn \( \ll 1 \))**

Young points out that even when the condensation rate is high, the pressure drop beyond the interface due to bulk flow of water vapor towards the droplet is negligible so that

\[
p_i = p
\]

The energy transfer equation in the continuum regime given by Young is

\[
L \frac{dM}{dt} = 4\pi (r + \beta \ell)^2 \frac{\lambda}{r + \beta \ell} (T_i - T_G)
\]

Equation (A6) can be used as it stands and so can equation (A7) if \( \lambda \) is taken to be the thermal conductivity of the gas mixture.

**Intermediate Regime**

The mass transfer equation in the intermediate regime is obtained by combining equations (A1) and (A6); that is,

\[
\frac{dM}{dt} = 2q_c \frac{4\pi r^2}{2 - q_c} \left[ \frac{p_i}{2\pi R T_i} - \frac{q_c p_s(T_L, r)}{2\gamma_1 \sqrt{2\pi R T_L}} \right]
\]

The energy transfer equation in the intermediate regime can be developed by replacing \( p_i \) in equation (A4) with \( p \) and combining that result with
equation (A7) to obtain
\[ \frac{dM}{dt} = \frac{4\pi r^2(\lambda/r)(T_L - T_G)}{r \left[ \frac{1}{r + \beta t} + \frac{\lambda}{2\pi \sigma^2 r^2} \frac{2\gamma_1}{\gamma_1 + 1} \right]} \quad (A9) \]

Now if equation (68) is used to eliminate \( p \) and equations (69) and (70) are used for \( \text{Pr} \) and \( \text{Kn} \), it follows that
\[ \frac{dM}{dt} = \frac{4\pi r^2(\lambda/r)(T_L - T_G)}{1 + \frac{1}{2\beta \text{Kn}} + \xi(\text{Kn}/\text{Pr}) \sqrt{T_i/T_G}} \quad (A10a) \]

and with the additional approximation that \( \sqrt{T_i/T_G} \) can be taken as unity,
\[ \frac{dM}{dt} = \frac{4\pi r^2(\lambda/r)(T_L - T_G)}{1 + \frac{1}{2\beta \text{Kn}} + \xi(\text{Kn}/\text{Pr})} \quad (A10b) \]

where
\[ \xi = \frac{\sqrt{8\pi}}{1.5} \frac{2\gamma_1}{\gamma_1 + 1} \quad (A11) \]

Young continues his analysis by developing a relationship between \( T_L - T_G \) and \( T_s(p_1) - T_G \). First,
\[ (T_L - T_G) = [T_s(p_1) - T_G] - (T'_s - T_L) \]
\[ + [T'_s - T_s(p_1)] \quad (A12) \]

where \( T'_s \) is the saturation temperature that corresponds to \( p_s(T_L, r) \). Young states that a very good approximation to the second term is
\[ (T'_s - T_L) = \frac{r^*}{r} [T_s(p_1) - T_G] \quad (A13) \]

The third term, \( T'_s - T_s(p_1) \), is obtained in the following way. Equation (A4) with \( p_i = p \) and equation (A8) are used to eliminate \( dM/dt \) to obtain an expression for \( p_s(T_L, r)/p_1 \); that is,
\[ \frac{p_s(T_L, r)}{p_1} = \frac{q_c}{\eta} \sqrt{\frac{T_L}{T_i}} \left[ 1 - \frac{2 - q_c \gamma_1 + 1}{2q_c} \frac{c_p}{2\gamma_1} \right] \]
\[ \times (T_L - T_i) \frac{f}{y_1} \left( \frac{m}{m_1} \right)^{1/2} \quad (A14) \]

Young then uses an approximate integral form of the Clausius-Clapeyron equation,
\[ T'_s - T_s(p_1) = \frac{R_1}{L} [T_s(p_1)]^2 \ln \frac{p_s(T_L, r)}{p_1} \quad (A15) \]

An expression for \( T_L - T_i \) in equation (A14) can be obtained by equating equation (A4) with \( p_i = p \) to equation (A10a) to get
\[ T_L - T_i = \delta(T_L - T_G) \quad (A16) \]

where
\[ \delta = \frac{\xi \text{Kn}/\text{Pr}}{1 + \frac{1}{2\beta \text{Kn}} + \xi \text{Kn}/\text{Pr}} \quad (A17a) \]

The additional approximation used by Young that \( \sqrt{T_G/T_i} \) is unity leads to
\[ \delta = \frac{\xi \text{Kn}/\text{Pr}}{1 + \frac{1}{2\beta \text{Kn}} + \xi \text{Kn}/\text{Pr}} \quad (A17b) \]

which is equation (82). Young also postulates that
\[ \frac{q_c}{\eta} = 1 + \frac{\alpha(T_L - T_i)}{T_s(p_1)} + \ldots \quad (A18) \]

where \( \alpha \) is a constant and higher order terms are neglected. Also note that
\[ \sqrt{\frac{T_L}{T_i}} = \left( \frac{T_L - T_i + 1}{T_i} \right)^{1/2} \quad (A19) \]

Substituting equations (A18) and (A19) into equation (A14), taking the logarithm of both sides of the resulting equation, and then using the approximation \( \ln (1 + x) = x \) for \( x \ll 1 \) yield
\[ \ln \frac{p_s(T_L, r)}{p_1} = \frac{\alpha(T_L - T_i)}{T_s(p_1)} + \frac{1}{2} \frac{(T_L - T_i)}{T_i} \]
\[ - \frac{2 - q_c \gamma_1 + 1}{2q_c} \frac{c_p}{2\gamma_1} \]
\[ \times \frac{L}{y_1} (T_L - T_i) \frac{f}{y_1} \left( \frac{m}{m_1} \right)^{1/2} \quad (A20) \]

Substituting this expression into equation (A15) and using equation (A16) give
\[ T'_s - T_s(p_1) = \nu \delta(T_L - T_G) \quad (A21) \]

where
\[ \nu = \frac{R_1 T_s(p_1)}{L} \left[ \frac{\alpha + \frac{1}{2} \frac{T_s(p_1)}{T_i}}{2} - \frac{2 - q_c \gamma_1 + 1}{2q_c} \right] \]
\[ \times \frac{c_p}{L} T_s(p_1) \frac{f}{y_1} \left( \frac{m}{m_1} \right)^{1/2} \quad (A22a) \]
The additional approximation that $T_s(p_1)/T_i$ is unity leads to
\[
\nu = \frac{R T_s(p_1)}{L} \left[ \frac{1}{2} + 1 - \frac{2 - \alpha c_1 T_s(p_1)}{2 \gamma_1} \frac{\gamma_1 + 1 \ c_{p_1} T_s(p_1)}{2} \right] \times \frac{f (m/L)}{y_1^{1/2}} \]  
(A22)

which is equation (78). A relationship between $(T_L - T_G)$ and $(T_s(p_1) - T_G)$ can now be formed by using equations (A13) and (A21) in equation (A12) to obtain
\[
(T_L - T_G) = \frac{1 - \tau_s/r}{1 - \nu \delta} [T_s(p_1) - T_G] \]  
(A23)

This expression for $T_L - T_G$ can be substituted into equation (A10b) and since the mass of a liquid water droplet is
\[
M = \frac{4}{3} \pi r^3 \rho_L \]  
(A24)

the droplet growth equation becomes
\[
\frac{dr}{dt} = \frac{\lambda (1 - \tau_s/r)[T_s(p_1) - T_G]}{(L \rho L r)^{1/2} \left[ 1 + 2 \gamma_1 Kn + \xi (Kn/Pr) (1 - \nu) \right]} \]  
(A25)

This equation reduces to that given by Young for pure steam for $f = 1$.

For droplets with $z = r/r_s > 1.1$, Young defines a Knudsen number based on a critical droplet radius as
\[
Kn_s = \frac{r}{2r_s} \]  
(A26)

and assumes that the vapor properties remain constant over an integration step so that equation (A25) can be written as
\[
\frac{dz}{dt} = \frac{\lambda (1 - 1/\Delta z) [T_s(p_1) - T_G]}{(L \rho L r)^{1/2} \left[ \frac{z^2}{z + 2 \gamma_1 Kn_s} + \xi (Kn_s/Pr)(1 - \nu) \right]} \]  
(A27)

This equation can be integrated over a time step $\Delta t$ from $z_j$ to $z_{j+1}$ to give equation (72) where the terms $\Omega, f, \theta, \lambda, \xi, J, \nu$ are defined by equations (73) through (78).

For droplets with $r/r_s < 1.1$, Young uses the energy transfer equation appropriate in the free molecular regime; that is, equation (A4) with $p_i = p$ and $T_i = T_G$. Again if equation (A23) but with $\delta = 1$ and equation (A24) are used, it follows that for $r/r_s < 1.1$,
\[
\frac{dr}{dt} = \frac{p}{\rho L \sqrt{2 \pi RT_G}} \gamma_1 + \frac{\tau_s/r}{1 - \nu} [T_s(p_1) - T_G] \]  
(A28)

Now define
\[
\psi = \frac{p}{r \sqrt{2 \pi RT_G}} \left( \frac{\gamma_1 + 1}{2 \gamma_1} \right) \frac{c_{p_1}}{L \rho L} 1 - \frac{r_s/r}{1 - \nu} [T_s(p_1) - T_G] \]  
(A29)

so that
\[
\frac{dr}{dt} = r \psi \left( 1 - \frac{r_s}{r} \right) \]  
(A30)

Subtraction of $dr_*/dt$ from both sides of this equation gives
\[
\frac{d(r - r_*)}{dt} = \psi (r - r_*) - \frac{dr_*}{dt} \]  
(A31)

At this point, Young makes the approximation that $\psi$ is constant over an integration step so that equation (A31) can be integrated over a time interval $\Delta t$ to give equation (79). An expression for $\psi$ given by equation (80) is obtained from equation (A29) with $r$ set equal to $r_j$, and all other quantities in the expression for $\psi$ are also evaluated at station $j$. 

17
Appendix B

Thermodynamic and Transport Property Data for Products of Combustion

This appendix lists a set of coefficients from which thermodynamic and transport properties can be computed for products of combustion. The thermodynamic properties can be computed over the temperature range from 200 K to 3000 K and the transport properties up to 500 K. The coefficients for the thermodynamic properties were obtained from a least-squares fit to tabulated heat capacity and equilibrium constant values given in the JANAF Thermochemical Tables (ref. 10). The coefficients for computing viscosity were determined from a least-squares fit to tabulated data presented by Touloukian, Saxena, and Hestermans (ref. 7) and the coefficients for computing thermal conductivity using tabulated data presented by Touloukian, Liley, and Saxena (ref. 8).

Table B1 lists coefficients $A_{n,k}$ ($n = 1, 2, \ldots, 6$) that appear in equation (36) for the heat capacity, the integration constant for the enthalpy $A_{7,k}$ that appears in equation (42), and the integration constant for the entropy $A_{8,k}$ that appears in equation (44). The numerical values listed correspond to units for $C_p,k$ in calories per mol-kelvin. The coefficients listed in table B1(a) apply to a temperature range of 200 K $\leq T \leq$ 1000 K and table B1(b) apply to 1000 K $\leq T \leq$ 3000 K.

Table B1. Heat Capacity Coefficients $A_{n,k}$ ($n = 1, 2, \ldots, 6$) and Constants of Integration

<table>
<thead>
<tr>
<th>$A_{n,k}$ for $n =$</th>
<th>$A_{7,k}$</th>
<th>$A_{8,k}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k$</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>(a) 200 K $\leq T \leq$ 1000 K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>6.41000E+01</td>
<td>7.59151E+00</td>
</tr>
<tr>
<td>2</td>
<td>1.04750E+02</td>
<td>3.37574E+00</td>
</tr>
<tr>
<td>3</td>
<td>-1.60500E+02</td>
<td>9.06284E+00</td>
</tr>
<tr>
<td>4</td>
<td>4.94000E+01</td>
<td>6.94324E+00</td>
</tr>
<tr>
<td>5</td>
<td>-4.26900E+01</td>
<td>9.83533E+00</td>
</tr>
<tr>
<td>6</td>
<td>-1.51550E+02</td>
<td>8.84079E+00</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>4.96800E+00</td>
</tr>
<tr>
<td>8</td>
<td>1.06200E+02</td>
<td>5.01442E+00</td>
</tr>
<tr>
<td>9</td>
<td>1.65300E+02</td>
<td>6.18389E+00</td>
</tr>
<tr>
<td>10</td>
<td>-9.10512E+00</td>
<td>8.49215E+00</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>$A_{n,k}$ for $n =$</th>
<th>$A_{7,k}$</th>
<th>$A_{8,k}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$k$</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>(b) 1000 K $\leq T \leq$ 3000 K</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>3.49199E+03</td>
<td>-4.6184E+04</td>
</tr>
<tr>
<td>2</td>
<td>-2.22684E+03</td>
<td>1.37397E+01</td>
</tr>
<tr>
<td>3</td>
<td>-3.58341E+02</td>
<td>6.31771E+00</td>
</tr>
<tr>
<td>4</td>
<td>-1.53922E+02</td>
<td>1.07380E+01</td>
</tr>
<tr>
<td>5</td>
<td>1.48203E+02</td>
<td>2.67165E+00</td>
</tr>
<tr>
<td>6</td>
<td>5.22642E+02</td>
<td>3.50182E+00</td>
</tr>
<tr>
<td>7</td>
<td>0</td>
<td>4.96800E+00</td>
</tr>
<tr>
<td>8</td>
<td>8.12266E+01</td>
<td>4.87159E+00</td>
</tr>
<tr>
<td>9</td>
<td>1.94043E+03</td>
<td>5.88699E-01</td>
</tr>
<tr>
<td>10</td>
<td>-3.78024E+02</td>
<td>6.62079E+00</td>
</tr>
</tbody>
</table>
Table B2 lists coefficients $B_{n,j}$ that appear in equation (35) for the six equilibrium constants $K_{p,j}$ that correspond to reactions I through VI (eqs. 30). The numerical values of $B_{n,j}$ listed correspond to units for $K_{p,j}$ that are nondimensional for $j = 1, 3,$ and $6$ and are in atmospheres for $j = 2, 4,$ and $5$.

Table B2. Equilibrium Constant Coefficients $B_{n,j}$

<table>
<thead>
<tr>
<th>$j$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
<th>6</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$5.09440E+03$</td>
<td>$-5.98336E+00$</td>
<td>$1.65031E-03$</td>
<td>$-4.45924E-07$</td>
<td>$4.89373E-11$</td>
<td>$8.42816E-16$</td>
</tr>
<tr>
<td>2</td>
<td>$6.80400E+04$</td>
<td>$-2.02038E+01$</td>
<td>$2.04607E-03$</td>
<td>$1.83617E-06$</td>
<td>$6.39945E-10$</td>
<td>$8.33955E-14$</td>
</tr>
<tr>
<td>3</td>
<td>$1.04351E+05$</td>
<td>$-1.40641E+00$</td>
<td>$3.07030E-03$</td>
<td>$-1.40641E-06$</td>
<td>$3.75299E-10$</td>
<td>$-4.25248E-14$</td>
</tr>
<tr>
<td>4</td>
<td>$3.26442E-03$</td>
<td>$1.26601E+01$</td>
<td>$3.06797E-05$</td>
<td>$-2.40321E-07$</td>
<td>$1.01593E-10$</td>
<td>$-1.35585E-14$</td>
</tr>
<tr>
<td>5</td>
<td>$3.06797E-05$</td>
<td>$-2.10521E+04$</td>
<td>$2.64558E-03$</td>
<td>$1.65031E-03$</td>
<td>$4.89037E-11$</td>
<td>$-6.39945E-10$</td>
</tr>
<tr>
<td>6</td>
<td>$-5.98336E+00$</td>
<td>$-2.02038E+01$</td>
<td>$1.04351E+05$</td>
<td>$-2.10521E+04$</td>
<td>$3.26442E-03$</td>
<td>$8.42816E-16$</td>
</tr>
</tbody>
</table>

Table B3 lists coefficients $D_{n,k}$ that appear in equation (63) for the viscosity of the seven species with $k = 1, 2, \ldots, 6, 10$. The units for $\mu_k$ based on this set of coefficients are newtons-second per meter$^2$. Table B4 lists coefficients $E_{n,k}$ in equation (66) for the thermal conductivity of the seven species with $k = 1, 2, \ldots, 6, 10$. The units for $\lambda_k$ based on this set of coefficients are joules per second-meter-kelvin. Note that the viscosity and thermal conductivity data are only required in the region where water droplet growth occurs, which is generally at a temperature less than 500 K. Also, at these temperatures the species corresponding to $k = 7, 8,$ and $9$ do not appear in significant amounts.

Table B3. Coefficients in Expression for Viscosity $D_{n,k}$

<table>
<thead>
<tr>
<th>$k$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$-1.5371E-05$</td>
<td>$1.5429E-07$</td>
<td>$-3.1981E-10$</td>
<td>$2.7958E-13$</td>
</tr>
<tr>
<td>2</td>
<td>$-2.5212E-07$</td>
<td>$5.5024E-08$</td>
<td>$-1.2381E-11$</td>
<td>$5.5279E-15$</td>
</tr>
<tr>
<td>3</td>
<td>$-4.9877E-07$</td>
<td>$7.8689E-08$</td>
<td>$-6.9794E-11$</td>
<td>$3.6438E-14$</td>
</tr>
<tr>
<td>4</td>
<td>$-5.7709E-07$</td>
<td>$8.8852E-08$</td>
<td>$-7.0382E-11$</td>
<td>$3.4305E-14$</td>
</tr>
<tr>
<td>5</td>
<td>$1.6225E-06$</td>
<td>$2.9338E-08$</td>
<td>$-1.9878E-11$</td>
<td>$1.1391E-14$</td>
</tr>
<tr>
<td>6</td>
<td>$-6.0917E-08$</td>
<td>$7.6311E-08$</td>
<td>$-6.4962E-11$</td>
<td>$3.2374E-14$</td>
</tr>
<tr>
<td>10</td>
<td>$-8.4886E-07$</td>
<td>$8.5390E-08$</td>
<td>$-7.1595E-11$</td>
<td>$3.5702E-14$</td>
</tr>
</tbody>
</table>

Table B4. Coefficients in Expression for Thermal Conductivity $E_{n,k}$

<table>
<thead>
<tr>
<th>$k$</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$-2.0450E-01$</td>
<td>$1.7566E-03$</td>
<td>$-4.6279E-06$</td>
<td>$4.2102E-09$</td>
</tr>
<tr>
<td>2</td>
<td>$-1.4403E-05$</td>
<td>$2.5166E-05$</td>
<td>$1.3405E-07$</td>
<td>$-1.1069E-10$</td>
</tr>
<tr>
<td>3</td>
<td>$5.3712E-04$</td>
<td>$8.8425E-05$</td>
<td>$-1.5431E-08$</td>
<td>$-1.8246E-11$</td>
</tr>
<tr>
<td>4</td>
<td>$-3.8305E-03$</td>
<td>$1.3168E-04$</td>
<td>$-1.2385E-07$</td>
<td>$8.1162E-11$</td>
</tr>
<tr>
<td>5</td>
<td>$-5.5274E-02$</td>
<td>$1.2443E-03$</td>
<td>$-1.9310E-06$</td>
<td>$1.3762E-09$</td>
</tr>
<tr>
<td>6</td>
<td>$-3.0706E-03$</td>
<td>$1.3215E-04$</td>
<td>$-1.4900E-07$</td>
<td>$1.0324E-10$</td>
</tr>
<tr>
<td>10</td>
<td>$-1.8772E-03$</td>
<td>$1.1087E-04$</td>
<td>$-6.9710E-08$</td>
<td>$2.7900E-11$</td>
</tr>
</tbody>
</table>
Appendix C

Computer Program

This appendix presents a computer program, FIRACON, which has been developed to solve the finite-rate condensation problem with the solution technique of the present paper. The program is written in FORTRAN Version 5 language for the CDC® CYBER 180 series digital computer system, model 860, with Network Operating System. The program requires approximately 134000 octal locations of core language and a typical case requires approximately 200 central processing unit seconds. Program input requirements and operating instructions are summarized in the program.

---

```
PROGRAM FIRACON(INPUT,OUTPUT,TAPE1=INPUT,TAPE6=OUTPUT,
                 1 SAVDAT,TAPE7,SAYDAT,ROIST,TAPE33=ROIST)

******************************************************************************

FINITE RATE CONDENSATION
******************************************************************************

PROGRAM FIRACON PROVIDES A NUMERICAL SOLUTION FOR THE
ADIABATIC EXPANSION OF COMBUSTION PRODUCTS WITH NUCLEATION
AND WATER DROPLET GROWTH. THE NUMERICAL SOLUTION IS
DIVIDED INTO THREE SUCCESSIVE PARTS. THE FIRST PART
INCLUDES THE COMPUTATION OF THE STAGNATION CONDITIONS
IN THE COMBUSTION CHAMBER AND THE DETERMINATION OF THE
MASS FLOW RATE. THE SECOND PART INVOLVES THE COMPUTATION
OF ISENTROPIC FLOW FROM THE NOZZLE THROAT TO A POINT IN
THE NOZZLE WHERE NUCLEATION JUST BEGINS. THE THIRD PART
DEALS WITH THE COMPUTATION OF THE NUCLEATION PROCESS
TO DETERMINE THE NUMBER OF DROPLETS FORMED AND THE
SUBSEQUENT GROWTH OF THE LIQUID WATER DROPLETS.

COMMON BLOCKS ARE USED EXTENSIVELY TO PASS INFORMATION
BETWEEN THE VARIOUS SUBPROGRAMS. THE CONTENTS OF THE
PRIMARY COMMON BLOCKS (PARTICULARLY AS THEY RELATE TO
INPUT DATA AND OPERATING INSTRUCTIONS) ARE DESCRIBED
IN THE DATA INPUT AND INITIALIZATION SUBROUTINE, DATAI.
THE USE OF COMMON BLOCKS IS SUMMARIZED IN THE FOLLOWING
TABLE:

COMMON BLOCK

| SUBPROGRAM | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O | P | Q | R | S | T | U | V | W | X | Y |
| FIRACON     |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| AVSX        | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| CCP         |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| CDEN        |   |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| CECOMP      |   |   |   | X | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| CEDATA      |   |   |   |   |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| CHG         |   |   |   |   |   |   |   | X |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
```
WHERE THE COLUMN HEADINGS REFER TO THE FOLLOWING
COMMON BLOCKS:

A) /ADJUST/   H) /CONSTS/   N) /K1K2ETC/   T) /SIGCOM/
B) /ARECUMI   I) /ECOM/     O) /OUTII U/   V) /SWx2/
C) /CMDUTZI   J) /EXPDAT/   P) /OUTII V/   W) /SWITCH/
D) /CMDUT3/   K) /FEED/     Q) /PLTB/   X) /SWIP/   Y) /SWIT/   
E) /CMDUT3/   L) /GAMOUT/   R) /PTOUT/   X) /TANUP/   
F) /COEFFU/   M) /INPT/     S) /RESLTS/   
G) /CONST/   C) 

COMMON/SWITCH/SWEND/SWPO/SWGU
INTEGER SWEND,SWPO,SWGO
COMMON DUMMY(10002)
SUBROUTINE AVSX(X, A)

SUBROUTINE AVSX CALCULATES THE NOZZLE AREA AS A FUNCTION OF X POSITION.
ACOEFS CONTAINS THE SPLINE COEFFICIENTS DETERMINED BY THE LIBRARY ROUTINE CSOS.

COMMON/ARECON/NA, X, ACOEF
COMMON/CONST/TC, K, CAPR, M, W(10), RMOL, RBAR, PI, ALPHAC
REAL K, M
DIMENSION ACOEF(10, 4)
DIMENSION X(10)

NPD = NA - 1
XXI = X
DO 10 IJ = 1, NPD
10 CONTINUE
IJ = NPD

H = XXI - XA(IJ)
RBARX = ((ACOEF(IJ, 6) * H + ACOEF(IJ, 3)) * H
1 + ACOEF(IJ, 2)) * H + ACOEF(IJ, 1)
A = PI * RBARX**2
RETURN
END

SUBROUTINE CCP(T, CP1, CP, CPC)

SUBROUTINE CCP CALCULATES THE SPECIFIC HEAT CAPACITY FOR WATER VAPOR AND THE CARRIER GAS.

COMMON/CONST/TC, K, CAPR, M, W(10), RMOL, RBAR, PI, ALPHAC
REAL K, M
COMMON/ACOEFC/OEF(10, 4)
COMMON/SIGCOM/SIGMA(10), SIG, PSUPO

VAL = 0.
J = 1
IF(J .LT. 1000.) GO TO 10
J = 2
10 CONTINUE

DO 20 I = 1, 10
20 VAL = VAL + SIGMA(I) * (COEF(I, 1, J) / T + COEF(I, 2, J)
1 + COEF(I, 3, J) * T + COEF(I, 4, J) * T**2 + COEF(I, 5, J) * T**3
2 + COEF(I, 6, J) * T**4)
IF(I.EQ.1) CP1 = VAL

DO 25 I = 1, 10
25 CP1 = CP1 + SIGMA(I) * W(I)

CP1 = CP1 / SIGMA(1) * 4.184 / W(I)
CP = CP1 / (SIG - SIGMA(1)) * 4.184

WC = 0.
DO 30 I = 2, 10
30 WC = WC + SIGMA(I) * W(I)

WC = WC / (SIG - SIGMA(1))

CPC = CPC / WC
CP = 4.184 * VAL
RETURN
END
SUBROUTINE COEN(RHO St T)

SUBROUTINE COEN CALCULATES THE MASS DENSITY OF THE GASEOUS MIXTURE BY INVERTING EQUATION (44)

COMMON/CONST/TC, K, CAPR, M, W(10), RHO, BAR, PI, ALPHAC
REAL K, M
COMMON/CODES/CNOEF(I:8,2)
COMMON/SIGCOM/SIGMA(I:10), SIG, PSUPO

COMMON/ICONST/ICAPR, M, W(10), RHO, BAR, PI, ALPHAC
REAL K, M
COMMON/CODES/CNOEF(I:8,2)
COMMON/SIGCOM/SIGMA(I:10), SIG, PSUPO

VAL = 0.
RP = CAPR/4.184
J = 1
IF(T,LT,1000.) GO TO 10
J = 2
10 CONTINUE

J = 1
DO 10 I = 1, 10
VAL = VAL + SIGMA(I)*(-COEF(I,1,J)/T + (COEF(I,2,J) - RP)*ALOG(T)
10 CONTINUE

VAL = 4.184*VAL

J = 2
DO 20 I = 1, 10
VAL = VAL + SIGMA(I)*(-COEF(I,1,J)/T + (COEF(I,2,J) - RP)*ALOG(T)
20 CONTINUE

VAL = 4.184*VAL

J = 3
DO 30 I = 1, 10
VAL = VAL + SIGMA(I)*(-COEF(I,1,J)/T + (COEF(I,2,J) - RP)*ALOG(T)
30 CONTINUE

VAL = -VAL/((4*1.256*CAPR)
RHOD = EXP(VAL)
RETURN
END

23
SUBROUTINE CECOMP

SUBROUTINE CECOMP CALCULATES THE CHEMICAL EQUILIBRIUM COMPOSITION OF COMBUSTION PRODUCTS

DIMENSION KPI(6), BB(I,J)
REAL KPI(1), B(1,J), X(1,J)
REAL KPI
COMMON/K1K2/K1,K2,K3,K4,K5,K6
COMMON/SIGMAS/S, S0, SN, SC
COMMON/CONSTS/FACTUR, EPS
COMMON/RESULT/S(11)
COMMON/INPT/PO, XLAST, PHI1, RHC, RNO
COMMON/TANDP/XX, PX
COMMON/SCOM/SCOM(I)

EQUILIBRIUM CONSTANT COEFFICIENTS, B(I,J)
(N = 1, 3, 5, 7, 9, 11)

DATA BB/5.0943976E+3,-5.98335611.650312E-3,
1.4.4592415E-7,-4.893668E-11.8.4281563E-16,2.8940026E+4,
2.2.023927E+1-2.0460652E+3,1.8361607E-6,2.3993532E-10,
3.3.395483E-14,-5.2550958E+4,1.0434537E+2,3.0703042E-3,
4.4.1.0460056E-6,3.7228596E-10,-4.2520015E-16,
5.5.9770152E-4,1.2660142E+1,3.2644200E-3,1.7662154E-6,
6.6.1.0028356E-10,-5.9550966E-14,-9.405111E+3,
7.7.3.9949255E-3,3.0679650E-5-2.4032609E-7,1.0159289E-10,
8.8.1.3584656E-14,-2.1052085E+4,8.6629408E-11,2.6459821E-3,
9.9.1.5772980E-6,4.6339262E-10,-5.4650140E-14,
DATA PSUP0/1.01325E+5/
DATA FACTOR/0.5/
DATA EPS/1.0E-8/

EQUATION (35)

DO 10 J=1,6
KPI(J) = KPI(J)*BB(1,J), XX)
CONTINUE

CONTINUE
K1 = 1./KPI(1)
K2 = (XX/PSUP0)*KPI(2)
K3 = 1./K2
K4 = KPI(3)/KPI(2)
K5 = KPI(4)/KPI(2)
K6 = KPI(5)
RHC = PHI1/(2.0 + 0.5*RHC)
SC = 1000.0/RHC + 12.0 + (14.0*RNO + 16.0)/RCD
SN = RHC/SC
SO = SC/RCD
SN = RND/SD
CALL ELEMBAL
CONTINUE
SIGMA(I) = SIG(I)
CONTINUE
SIG = SIG(11)
RETURN
END
BLOCK DATA CEDATA

HEAT CAPACITY COEFFICIENTS A(N,K) (N=1,2,...,9)
AND CONSTANTS OF INTEGRATION FOR ENTHALPY A(N,K)
AND ENTROPY A(N,K) ... TABLE B1

THESE COEFFICIENTS CORRESPOND TO THE SPECIES
H2O, CO2, CO, O2, H2, N2, H, O, OH, AND NO
IN THAT ORDER.

COMMON/COEFCO/COEF(110,8;2)

DATA (COEF(i,j,k)) i=1,100, j=1,101, k=1,8

<p>| | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>4</td>
<td>5</td>
<td>6</td>
<td>7</td>
<td>8</td>
</tr>
</tbody>
</table>

ORIGIN. PAGE IS OF POOR QUALITY
SUBROUTINE CHG(T,HG,HGI)
C
SUBROUTINE CHG CALCULATES THE SPECIFIC ENTHALPY
OF THE GASEOUS MIXTURE AND OF WATER VAPOR
C
COMMON/COEFCO/COEF(10,8,2)
COMMON/CONST/C,K,CAPR,M1,W(10),RHOL,RBAR,PI,ALPHAC
REAL K,M1
COMMON/SIGCON/SIGMA(10),SIG,PSUP0
C
VAL = 0.
J = 1
IF(T.LT.1000.) GO TO 10
J = 2
10 CONTINUE
C
EQUATION (42)
DO 20 I=1,10
VAL = VAL + SIGMA(I)*(COEF(I,1,J)*ALOG(T) + COEF(I,2,J)*T
1 + COEF(I,3,J)*T**2/2. + COEF(I,4,J)*T**3/3.
2 + COEF(I,5,J)*T**4/4. + COEF(I,6,J)*T**5/5. + COEF(I,7,J))
IF(J.EQ.1) HGI = VAL/SIGMA(I)
20 CONTINUE
HG = 4.184*VAL
C
EQUATION (43)
HG1 = HG1 + 4.184/W(1)
C
RETURN
END

SUBROUTINE CS(S,RHO,T)
C
SUBROUTINE CS CALCULATES THE SPECIFIC ENTROPY
OF THE GASEOUS MIXTURE
C
COMMON/CONST/C,K,CAPR,M1,W(10),RHOL,RBAR,PI,ALPHAC
REAL K,M1
COMMON/COEFCO/COEF(10,8,2)
COMMON/SIGCON/SIGMA(10),SIG,PSUP0
C
VAL = 0.
RP = CAPR/4.184
J = 1
IF(T.LT.1000.) GO TO 10
J = 2
10 CONTINUE
C
EQUATION (44)
DO 20 I=1,10
IF(SIGMA(I),EQ,0,) GO TO 30
VAL = VAL - CAPR_SIGMA(I)_ALOG(SIGMA(I))
30 CONTINUE
S = VAL
RETURN
END
SUBROUTINE DATAII

SUBROUTINE DATAII HANDLES THE DATA INPUT AND
INITIALIZATION. THE SUBROUTINE CSOS IS A PART OF THE
MATHEMATICAL LIBRARY AT LANGLEY. FOR MORE INFORMATION,
SEE "MATHEMATICAL AND STATISTICAL SOFTWARE AT LANGLEY",
CENTRAL SCIENTIFIC COMPUTING COMPLEX DOCUMENT N2-3G.

DIMENSION ACOEF(10),+1, DY(10), XA(10), RWA(10), DELSTR(10)
DIMENSION Y(10)
DIMENSION WK(79)

"EXPDAT" VARIABLES

NEXP - NUMBER OF EXPERIMENTAL DATA POINTS TO BE
DISPLAYED ON PRESSURE PLOT. IF NEXP IS
ZERO, NO EXPERIMENTAL DATA IS PLOTTED.
IF NEXP IS NOT ZERO, THE EXPERIMENTAL
DATA IS INCLUDED ON PLOT TYPE 1.
XEXP - X POSITIONS FOR WHICH EXPERIMENTAL DATA
IS TO BE DISPLAYED ON PRESSURE PLOT
PEXP - EXPERIMENTAL VALUES OF PRESSURE TO BE
DISPLAYED ON PRESSURE PLOT
PREF - REFERENCE PRESSURE - I.E., THE VALUE PEXP/PREF
IS DRAWN ON THE PRESSURE PLOT

COMMON/EXPDAT/NEXP, XEXP(50), PEXP(50), PREF

"SWITCH" VARIABLES

SWEND - END OF DATA SWITCH
0 - DATA ENCOUNTERED, PROCESS IT
1 - END OF DATA ENCOUNTERED, STOP
SWPO - PRINTED OUTPUT SWITCH
0 - DO NOT PRINT OUTPUT
1 - PRINT OUTPUT (DEFAULT)
SWGO - GRAPHIC OUTPUT SWITCH
0 - DO NOT PLOT OUTPUT
1 - PLOT OUTPUT (DEFAULT)

COMMON/SWITCH/SWEND, SWPO, SWGO
INTEGER SWEND, SWPO, SWGO

"ARECOM" VARIABLES

NA - NUMBER OF X POSITIONS DEFINING NOZZLE GEOMETRY
XA - NOZZLE X COORDINATES
ACOEF - SPLINE COEFFICIENTS DEFINING AREA VS X

COMMON/ARECOM/NA, XA, ACOEF

"FEED" VARIABLES

HFUEL - ENTHALPY OF THE FUEL AT THE FEED
TEMPERATURE, CAL/MOL
TFEEED - TEMPERATURE OF THE FEED, K
NATOM - NUMBER OF ATOMS OF CARBON IN A MOLECULE
OF FUEL

COMMON/FEED/HFUEL, TFEED, NATOM

"PLTBLK" VARIABLES

PLTBLK CONTAINS THE MINIMUM AND MAXIMUM VALUES
FOR ALL PLOT OUTPUT. IF ANY PARAMETERS ARE NOT
SPECIFIED, THE CORRESPONDING PLOT IS SCALED
AUTOMATICALLY

XMIN
COMMON/PLTBLK/XMIN,XMAX,PMIN,PMAX,TMIN,TMAX,
1 YMIN,YMAX,DTMIN,DTMAX,DMIN,DMAX,RMIN,RMAX,
2 PRMIN,PRMAX,TMIN,TLMAX,RHOMIN,RHOMAX,
3 SMIN,SMAX,UMIN,UMAX,XMIN,XMAX,IPSCLT(15)

"CONST* VARIABLES

TC - CRITICAL TEMPERATURE OF WATER, K
K - BOLTZMANN CONSTANT, J/MOLECULE*K
C CAPR - UNIVERSAL GAS CONSTANT, J/MOL-K
C M1 - MOLECULAR MASS OF WATER, KG/MOLECULE
C W - MOLECULAR WEIGHT TABLE, KG/MOL
C RHOL - DENSITY OF WATER, KG/M**3
C RBAR - SPECIFIC GAS CONSTANT FOR WATER, R/M1, J/KG-K
C PI - AREA OF UNIT CIRCLE
C ALPHAC - THERMAL ACCOMMODATION COEFFICIENT FOR CARRIER
  GAS INTERACTION WITH WATER DROPLETS
C COMMON/CONST/TG,CAFR,M1,W(10),RHOL,RBAR,PI,ALPHAC
C REAL K,M1
C "ADJUST" VARIABLES
C GAMMA - RATIO OF SPECIFIC HEATS
C QC - CONDENSATION COEFFICIENT
C ALPHA - CONSTANT IN EQUATION (76)
C BETA - LAMMOURIR PARAMETER
C DSTAR - DIAMETER OF NOZZLE THROAT, M**3
C ASTAR - AREA OF NOZZLE THROAT, M**3
C DT2 - TEMPERATURE STEP FOR PART 2, K
C JMIN - BEGINNING NUCLEATION RATE FOR PART 3,
  DROPLETS FORMED/M**3-S
C DELX - STEP WIDTH FOR PART 3 CALCULATIONS, M
C EPS - CONVERGENCE ARRAY
C 1 - NEWTON ITERATION TO SOLVE EQUATION (57)
C 2 - NEWTON ITERATION TO SOLVE EQUATION (111)
C 3 - PRESSURE CONVERGENCE IN PART 1
C 4 - MASS FLUX CONVERGENCE IN PART 1
C 5 - PRESSURE CONVERGENCE IN PART 2
C 6 - NEWTON ITERATION FOR T(J-1) IN PART 3
C 7 - TEMPERATURE CONVERGENCE, EQUATION (133), IN PART 3
C 8 - NEWTON ITERATION FOR X AS A FUNCTION OF A
C IDOUT - OUTPUT LEVEL
C 0 - STANDARD SUMMARY OUTPUT
C 1 - EXTENDED OUTPUT
C 2 - DEBUG OUTPUT
C DT1 - TEMPERATURE STEP FOR PART 1, K
C CVRAT - CONVERGENCE RATIO FOR DELS/CV, EQUATION (126)
C ARAT - CONVERGENCE RATIO FOR A^+A, EQUATION (131)
C JSTOP - MAXIMUM NUMBER OF BANDS IN PART 3
C DIMENSION STATEMENTS LIMIT THIS VALUE TO 400. ANY
C SMALLER NUMBER WILL PROVIDE A MEANS TO STOP THE
C CALCULATIONS SHORT OF THE END OF THE NOZZLE.
C JOB - DEBUG PRINT CONTROL FOR PART 3
C IF THIS VALUE IS DEFINED, 'EXTENSIVE' PRINTOUT
C IS PROVIDED FOR DEBUGGING BOTH BEFORE AND AFTER
C THE DBLRT ROUTINE FOR ALL BAND NUMBERS GREATER
C THAN OR EQUAL TO JOB
C CP - SPECIFIC HEAT CAPACITY OF MIXTURE, J/KG-K
C CP1 - SPECIFIC HEAT CAPACITY OF WATER, J/KG-K
C COMMON/ADJUST/GAMMA,QC,ALPHA,BETA,DSTAR,ASTAR,DT2,
C JMIN,DELX,EPS,B,1,IDOUT,DT1,CVRAT,ARAT,JSTOP,JDB,CP,CP1
C REAL JMIN
C "INPT" VARIABLES
C
C TO - INITIAL TEMPERATURE, K
C PO - INITIAL PRESSURE, M/M**2
C XLAST - FINAL X, M (USED TO TERMINATE CALCULATIONS)
C PHI1 - EQUIVALENCE RATIO
C RHC - ELEMENTAL RATIO OF HYDROGEN TO CARBON
C RNO - ELEMENTAL RATIO OF NITROGEN TO OXYGEN
C COMMON/INPT/TO,PO,XLAST,PHI1,RHC,RNO
C
NAMELIST PARAMETERS
C
DESCRIPTIONS OF MOST NAMELIST PARAMETERS CAN BE
FOUND ELSEWHERE IN THIS SUBROUTINE. THE FOLLOWING
PARAMETERS ARE LOCAL TO THIS ROUTINE.
C
IPLTRD - INTEGER FLAG TO INDICATE IF AUTO-SCALING
IS TO BE USED ON THE PLOTS.
O - USE AUTO-SCALING
1 - DO NOT USE AUTO-SCALING. THE NAMELIST
"PLTOVR" MUST BE INCLUDED IN THE INPUT DATA.
C
RWA - WALL COORDINATES FOR DEFINING TUNNEL GEOMETRY
C
DELSSTR - BOUNDARY LAYER DISPLACEMENT THICKNESSES FOR
DEFINING TUNNEL GEOMETRY
C
NAMELIST/DATA/INPT/TO,PO,XLAST,SWPD,SWGO,QC,ALPHA,BETA,
1 DSTAR,CYRAT,ARAT,DT1,DTOUT,DT2,JMIN,DELX,EPS,
2 JSTDP,JOB,PHI1,RHC,RNO, IPLTRD,NAX,RAW,DELSSTR,ALPHAC,
3 NEWP, XEP, PEP, PREP, LPSLC, TFUEL, TFEED, NATION
NAMELIST/PLTOVR/XMIN,XMAX,PMIN,PMAX,TMIN,TMAX,
1 YMIN,YMAX,DTMIN,DTMAX,DNMIN,DNMAX,RMIN,RMAX,
2 PRMIN,PRMAX,TLMIN,TLMAX,RHOMIN,RHOMAX,
3 SMIN,SMAX,UMIN,UMAX,XMIN,XMAX
C
SET CONSTANTS
C
DATA TC,K,CAPR,M,K,RHOL/6.73.286,1.3801*23,8.314,2.99E+26,
1 0.018,0.044,0.0294,0.032,0.002,0.028,0.001,0.016,0.017,
1 0.030,1.0/E3/
DATA DT/1.0/E2/
DATA DT2/1.0/E1/
DATA DTMIN/0.0,E-6/
DATA DTMAX/1.0,E6/
DATA DNMIN/0.0,E-6/
DATA DNMAX/1.0,E6/
DATA XHIN,XMAX,PXMIN,PXMAX,TXMIN,TXMAX,
1 YHIN,YMAX,DXMIN,DXMAX,RXMIN,RXMAX,
2 PRXMIN,PRXMAX,TLXMIN,TLXMAX,RHXMIN,RHXMAX,
3 SXMIN,SMAX,UXMIN,UXMAX,XXMIN,XXMAX
C
SET DEFAULTS FOR ADJUSTABLE CONSTANTS
C
DATA ALPHAC/1.0/E3/
DATA TO/0.0/E3/
DATA HFUEL,TFEED,NATOM/-1.7895,2.98,1.51/
DATA IPSLCT/1.5/E3/
DATA XEP/0.0/E3/
DATA XEP/0.0/1.5E3/
DATA XEP/92.2,140,164,118,224,230,240,
1 260,272,284,308,320,332,344,356,368,380,392,400/,
2 416,430,450,474,496,500,516,530,546,560,576,592/,
DATA QC,ALPHA,BETA,DSTAR/1.0,E9,0,
1 2.0,0.01128/
DATA SWEND/0.0/E3/
DATA IPLTRD/0.0/E3/
DATA XMIN,XMAX,PXMIN,PXMAX,TXMIN,TXMAX,
1 YMIN,YMAX,DXMIN,DXMAX,RXMIN,RXMAX,
2 PRXMIN,PRXMAX,TLXMIN,TLXMAX,RHXMIN,RHXMAX,
3 SXMIN,SMAX,UXMIN,UXMAX,XXMIN,XXMAX/1.5E3/
DATA CYRAT/1.0,E-3/
DATA ARAT/1.0,E-3/
DATA EPS/8.1E-6/
DATA PH1/0.8/
DATA RHC/4.0/
DATA RNO/3.7619/
DATA IOUT/1.0/
DATA DT1/1.0/
DATA DT2/0.2/
DATA NPASS/0.0/
DATA SWPD,SWGO/1.1,E3/
DATA JSTDP/1.0/
DATA JOB/1000/
DATA 1 FORMAT(1H1,20X,22HCONDENSATION EVOLUTION//)
C
SET COMPUTED CONSTANTS

RBAR = HAPP/W(1)
P1 = ACOS(-1.)
READ(5,DATAIN;END=10)
10 IF(EDF(5).NE.0) GO TO 30
   WRITE(33) TO:PO
   IF(IPLTRD.NE.0) READ(5,PLTOVR;END=20)
20 IF(EDF(5).EQ.0) GO TO 40
30 SWEND = 1
40 CONTINUE
   IF(TO,NE.0) GO TO 50
   TO = 1500.
   CALL FTEMP(I=FLAME)
   NTENS = (FLAME - 45.)/10.
   TO = 10.*FLOAT(NTENS)
50 CONTINUE
   IF(NPASS.EQ.0) WRITE(6,1)
   IF(NPASS.EQ.0) WRITE(6,DATAIN)
   NPASS = NPASS + 1
   IERR = 0
   NPM = 10
   S = 0.0001
   IPT = -1
   DO 60 I=1,NA
      Y(I) = RMA(I) - DELSTR(I)
   60 CONTINUE
   CALL CSDS(NPM,NA,XA,YA,DY,S,IPT,ACDEF,WK,IERR)
RETURN
END
SUBROUTINE DBLIT

SUBROUTINE DBLIT DETERMINES THE TEMPERATURE FOR WHICH TWO INDEPENDENT CALCULATIONS OF ENTROPY ARE EQUAL AND THE DENSITY FOR WHICH THE MASS FLOW AND CROSS SECTIONAL AREA ARE CONSISTENT

COMMON/SIGCOM(SIGMA(10)),SIG,PSUP0
COMMON T(J+1),X(J+1),DELN(400),CAPJX(400),RT(400),
    1 MU(400),TL(400),T(400),PC(400),RSTAR(400),
    2 UI(400),Sl(400),SL(400),HL(400),RG(400),CAPJY(400),
    3 P(400),G(400),YS(400),DELY(400),DELS(400),
    4 RPERM(400),TS(400),OUTMCH(400),CPJP(400)
REAL MU
COMMON/CONST/Tc,K,CR,M1,w(10),RHO*,BRAR,PI,ALPHAC
REAL K,.
COMMON/ADJUST/GAMMA1,OC,ALPHA,BETA,DSTAR,ASTAR,DT2,
    1 JMIN,DELX,EPS(8),IOUT,DUM,CVRAT,ARAT,JSTOP,JDB,CP,CP1
REAL JMIN
COMMON/DUT1/SO,H0,MDOT,RHOSTR,TSTAR
REAL MOOT
C
SGPP = (S(J+1) - SL(J+1))/(1. - YS(J+1))
10 CONTINUE
EQUATION (44)
    CALL CS(SGP,RHO(J+1),T(J+1))
    CALL CCP(T(J+1),CP,CP1,DUM)
EQUATION (126)
    USOCV = (SGPP - SGPI/CP - CAPR*SIG)
    IF(ABS(DSOCV).LT.CYRAT) GO TO 20
GO TO 10
TEMPERATURE ITERATION HAS CONVERGED
20 CONTINUE
EQUATION (128)
    HP = (1. - YS(J+1))*HGP + HL(J+1)
    CALL CHG(T(J+1),HGP,DUM)
CALCULATE SPECIFIC ENTHALPY USING EQUATION (128)
    U(J+1) = SORT(2.*(H0 - HP))
CALCULATE FLOW VELOCITY USING EQUATION (129)
    AP = MDOT*(1. - YS(J+1))/(RHO(J+1)*U(J+1))
CALCULATE CORRESPONDING AREA USING EQUATION (130)
    APRAT = ABS(AP - AJ(J+1))/A(J+1)
    IF(APRAT.LT.ARAT) GO TO 30
ADJUST RHO*, EQUATION (132), AND GO BACK
RHO(J+1) = MDOT*(1. - YS(J+1))/(A(J+1)*U(J+1))
GO TO 10
CALCULATE PRESSURE USING EQUATION (47)
DENSITY ITERATION HAS CONVERGED
30 CONTINUE
C
CALL SATTEM(TS(J+1),P(J+1))
RETURN
END
SUBROUTINE ELENBAL
REAL K1,K2,K3,K4,K5,K6
COMMON/RESULTS/S(I1)
COMMON/SIGMAS/SH,SN,SC
COMMON/CONSTS/FACTOR,EPS
COMMON/K1K2ETC/K1,K2,K3,K4,K5,K6
1 FORMAT(*** ELENBAL EXCEEDS 50 ITERATIONS ***)
S(2)=5*FACTOR
S(3)=SH/2.0
S(4)=(SH-SC-S(I1)-S(2)/2.0
S(6)=SN/2.0
S(I1)=(SH+SN+SC-S(I1)-S(2))/2.0
NITER=0
10 CONTINUE
YOLD=S(2)
ZOLD = S(4)
NITER=NITER+1
IF(NITER.LT.31) GO TO 10
WRITE(6,11)
STOP
20 CONTINUE
C COMPUTE THE REST OF THE VARIABLES........
25 S(3)=S(2)*SQRK2*S(I1)/S(4)
26 S(5)=S(I1)*S(3)/S(I2)*K1
27 S(I0) = SQRK6*S(4)/S(6))
28 S(6) = (SN-S(10))/2.0
29 S(9)=SQRK3*S(4)*S(5)
30 S(7)=SQRK4*S(5)*S(11)
31 S(6)=SQRK5*S(4)*S(11)
32 S(I1)=(SH-2.0*S(5)-S(7)-S(9))/2.0
33 S(2)=SC-S(3)
34 S(4)=(S(3)+S(I1)-S(2)-S(8)-S(9)-S(10))/2.0 /2.0
36 C CHECK FOR CONVERGENCE........
37 C
39 DY=YOLD-S(2)
40 DZ = ZOLD - S(4)
41 IF(NITER.EQ.1) GO TO 10
42 IF(ABS(DY/S(2)).LE.EPS) GO TO 30
43 GO TO 10
44 30 CONTINUE
45 IF(ABS(DZ)/S(4)).LE.EPS) GO TO 40
46 GO TO 10
47 40 CONTINUE
S(3)=S(2)*SQRK2*S(I1)/S(4)
S(5)=S(I1)*S(3)/S(I2)*K1
S(I0) = SQRK6*S(4)/S(6))
S(6) = (SN-S(10))/2.0
S(9)=SQRK3*S(4)*S(5)
S(7)=SQRK4*S(5)*S(11)
S(6)=SQRK5*S(4)*S(11)
S(8)=SQRK5*S(4)*S(11)
S(2)=SC-S(3)
RETURN
END
FUNCTION EXPP(VAL)

FUNCTION EXPP IS A UTILITY USED TO AVOID THE UNDERFLOW
MESSAGE ASSOCIATED WITH VERY SMALL ARGUMENTS

EXPP = 0.
IF(VAL.LT.-670.) GO TO 10
EXPP = EXPP(VAL)
10 CONTINUE
RETURN
END

SUBROUTINE FITLAM(LAMBDA,T)

SUBROUTINE FITLAM DETERMINES THE THERMAL CONDUCTIVITY
OF EACH SPECIES AS A POLYNOMIAL IN TERMS OF TEMPERATURE

DIMENSION LAMBOA(10)
REAL LAMBDA
DIMENSION COEF(4,10)

COEFFICIENTS IN EXPRESSION FOR THERMAL
CONDUCTIVITY E(N,K) ... TABLE B4

DATA COEF/-0.2045t1.7565E-3,-4.6279E-6,4.210CE-9,
1 1.14403E-5,-5.5166E-5,1.3405E-7,-1.1069E-10,
2 5.3712E-4,8.8425E-5,1.5431E-6,1.8246E-11,
3 -3.8505E-3,1.3168E-4,-1.2385E-7,-8.1162E-11,
4 -5.5274E-2,1.2443E-3,-1.9310E-6,1.3762E-9,
5 -3.0706E-3,1.2159E-4,1.4907E-7,1.1032E-10,
6 12*0.0,-1.8772E-3,1.1087E-4,-6.9710E-8,2.7900E-11/

EQUATION (66)

DO 20 I=1,10
LAMBA(I) = 0.
DO 10 J=1,4
LAMBA(I) = LAMBA(I) + COEF(J,I)*T**(J-1)
10 CONTINUE
20 CONTINUE
RETURN
END

SUBROUTINE FITMU(MU,T)

SUBROUTINE FITMU DETERMINES THE VISCOSITY
OF EACH SPECIES AS A POLYNOMIAL IN TERMS OF TEMPERATURE

DIMENSION MU(10)
REAL MU
DIMENSION COEF(4,10)

COEFFICIENTS IN EXPRESSION FOR VISCOITY
DIN(N,K) ... TABLE B3

DATA COEF/-1.5371E-5,1.5429E-7,-3.1981E-10,2.7958E-13,
1 -2.5212E-7,5.5029E-9,1.2381E-11,-9.527E-15,
2 -4.9877E-7,7.8689E-8,-6.9794E-11,3.6489E-14,
3 -5.7709E-7,8.0892E-8,7.0082E-11,3.4305E-14,
4 -1.6225E-6,2.9536E-8,1.9878E-11,1.3916E-14,
5 -6.0817E-8,4.311E-9,3.4462E-11,3.3734E-14,
6 12*0.0,-4.8886E-7,6.5396E-8,7.4195E-11,3.572E-14/

EQUATION (63)

DO 20 I=1,10
MU(I) = 0.
DO 10 J=1,4
MU(I) = MU(I) + COEF(J,I)*T**(J-1)
10 CONTINUE
20 CONTINUE
RETURN
END
SUBROUTINE FTEMP(TFLAME)

SUBROUTINE FTEMP COMPUTES THE FLAME TEMPERATURE OF HYDROCARBON COMBUSTION IN AIR WITH OR WITHOUT OXYGEN ENRICHMENT.

CONSTANT PRESSURE FORMULATION.

REFERENCE TEMPERATURE IS 298.15 KELVIN.

COMMON/NPT/TX,PO,XLAST,PHII,RHC,RNO
COMMON/TANDP/TXX,FXX
COMMON/FEED/HFUEL,TFEED,NATOM
COMMON/COEFCU/COEF(10,8,2)

1 FORMAT(I2X,'DID NOT CONVERGE IN 20 ITERATIONS')
2 FORMAT(I2X,'ENTHALPY OF THE FEED (CAL/KG) = ',F12.4)
3 FORMAT(I2X,'FLAME TEMPERATURE (KELVIN) = ',F8.2)
4 FORMAT(I2X,'DENSITY OF THE MIXTURE (KG/CM) = ',E12.6)
5 FORMAT(I2X,'ENTHALPY OF THE MIXTURE (CAL/KG) = ',F12.4)

1 + COEF(1,3,J)*T/2 + COEF(1,4,J)*T**2/2 + COEF(1,5,J)*T**3/3.
2 + COEF(1,6,J)*T**4/4 + COEF(1,7,J)*T**5/5 + COEF(1,8,J)

C FUEL = ENTHALPY OF THE FUEL AT THE FEED TEMPERATURE (CAL/MOL).
C TFEED = TEMPERATURE OF THE FUEL (KELVIN).
C N = NUMBER OF ATOMS OF CARBON IN A MOLECULE OF FUEL.
C TFO = FIRST GUESS OF FLAME TEMPERATURE, KELVIN.
C PO = PRESSURE OF THE FUEL (BARS).
C
C COMPUTE SIGMA(H), SIGMA(O), SIGMA(N), AND SIGMA(C)
C IN 1 KG OF THE MIXTURE.

EQUATION (8)

SD*500.0/(8.0*7.0*RNO*PHII*(12.0*RHC)/(4.0*RHC))

EQUATION (9)

SC = 2.0*PHII*50/(4.0 + RHC).

EQUATION (11)

SH=RHC*SC

EQUATION (10)

SN=50*RNO

ENTHALPY OF THE FEED.

FUEL HAS THE FORMULA C-N H-M.
THERE ARE N MOLS OF CARBON IN ONE MOL OF THE FUEL.

HFEED=HFUEL * SC * FLOAT(NATOM)
HFEED=HFEED*VAL(4.1,1,TFEED)*50/2.0*VAL(6.1,TFEED)*5N/2.0

COMPUTE THE EQUILIBRIUM COMPOSITION AT TFO AND PO.

COMPUTE THE ENTHALPY.

TXX = T0
PXX = PO
CALL CECOMP
TFO = TXX
CALL CHG(TFO,HO,DUMMY)

NOTE: SUBR CHG RETURNS H IN JOULES/KG.

T1=TFO-100.0
DH1=HFEED-H0/4.184
DO 10 ITER=1,20
10 TXX = T1
CALL CECOMP
CALL CHG(T1,H1,DUMMY)
DH2=HFEED-H1/4.184
TFLAME=(DH2-TFO-DH1)/(DH2-DH1)
IF(ABS(TFLAME-T1) .LE. 0.001) GO TO 20
TFO=T1
T1=TFLAME
DH1=DH2
10 CONTINUE
WRITE(6,*)
STOP
SUBROUTINE GRL(ZZ)

SUBROUTINE GRL CALCULATES THE GROWTH RATE
FOR LARGE DROPLETS (I.E., Z > 1.1)

COMMON/ECOM/DELTAT,CAVL,NU,PSI,DRUT,
1 T,LTILDA,KN,THETA,LAMBDA,PR,XI,OMEGA,CAVLAM
REAL NU,LTILDA,KN,LAMBDA
COMMON/CONST/TC,K+CAPR + M1, +10,RHOL,REBAR,PI,ALPHA,CAVL
REAL K, M1
COMMON/ADJUST/GAMMA,ALPHA,BETA,DSTAR,ASTAR,DTZ
1 JMIN,DELX,EPS(8),D,DT1,CTT,AT,ST,JSTOP,JDB,CF,CP1
REAL JMIN
COMMON J,J(400),J(400),DELT(400),CAPJ(400),R(400),
1 MU(400),T(400),F(400),RSTAR(400),
2 U(400),S(400),SL(400),HL(400),RHD(400),CAPJY(400),
3 F(400),G(400),TS(400),DEL(400),DEL(400),
4 RP(400),TS(400),OUTMCH(400),CAPJ(400),
5 REAL MU

ZOLD = ZZ
RAT = R(I)/RSTAR(J)
CAPDEL = CAPLAMDELTAT
ZPRIME = (THETA**3/(THETA + 1.))*ALOG(RAT - 1.1)
1 + (1./(THETA + 1.1 + OMEGA))ALOG(1/RAT - 1.1)
2 + OMEGA + 1.1 - THETA)*RAT + 0.5*RAT**2 + CAPDEL

CONTINUE

C

EQUATION (112)

FZZ = (THETA**3/(THETA + 1.))*ALOG(ZOLD + THETA)
1 + (1./(THETA + 1.1 + OMEGA))ALOG(ZOLD - 1.1)
2 + OMEGA + 1.1 - THETA)*ZOLD + 0.5*ZOLD**2 - ZPRIME

CONTINUE

C

EQUATION (113)

FZZP = THETA**3/(THETA + 1.1)*(ZOLD + THETA)
1 + (1./THETA + 1.1 + OMEGA)*ZOLD - 1.1)
2 + OMEGA + 1.1 - THETA

CONTINUE

C

EQUATION (114)

ZNEW = ZOLD - FZZ/FZZP
IF(ABS(ZNEW - ZOLD)/ZNEW.LT.EPS(2)) GO TO 20
ZOLD = ZNEW
GO TO 10
CONTINUE

Z = ZNEW
RETURN
END
SUBROUTINE GRLPRP
C C SUBROUTINE GRLPRP CALCULATES SEVERAL VARIABLES
C PREPARATORY FOR EVALUATION OF THE GROWTH RATE FOR
C LARGE DROPLETS
C
COMMON/SIGCOM/SIGMA(10), SIGHVSDUG
DIMENSION WORK(10)
COMMON/ECOM/DELTAT,CAPL,NU,FSl,DRDT,
1 TR,TILDA,KN,THETA,LAMBDA,PRXI,OMEGA,CAPLAM
REAL NU,TILDA,KN,LAMBDA
COMMON/IPMT/TO,XO,XLAST,PHI1,RH0,RH2,DOM
COMMON/SWITCH/SWEND,SWPO,SWGO
INTEGER SWEND,SWPO,SWGO
COMMON/COUT/DIST,NSIG,NSIG2,NSIG3,NSIG4,NSIG5,NSIG6,NSIG7,NSIG8
COMMON/CMOUT/NDIST,NSIG,NSIG2,NSIG3,NSIG4,NSIG5,NSIG6,NSIG7,NSIG8
COMMON/AOJUST/ALPHAI,BETAI,ALPHA,DSTAR,DSTAR
CALL SATTEM(TS,P8)
CALL CHQ(TS,HGX,HGI)
C C EQUATION (56)
C CAPL = HG1 + 4.2*3*TS(J) + 17.175365BE+6
CALL CGP(TS),CP,CP1,CP2)
WAVG = 0.
WC = 0.
DO 10 IJ=1,10
WAVG = WAVG + W(IJ)*SIGMA(IJ)
10 CONTINUE
WAVG = WAVG/SIG
WC = WC/SIG = SIGMA(IJ)
Y1 = SIGMA(IJ)/SIG
50 C C EQUATION (39)
C GAMMA1 = CP1/(CP1 - CAPR/W(1))
54 C C EQUATION (41)
C GAMMAC = CPC/(CP2 - CAPR/WC)
56 C C EQUATION (74)
C FF = Y1*SQRT(WRAT) + (1. - Y1)*SQRT(WC/WAVG)*(GAMMA1/GAMMAC)
61 1 = (GAMMAC + 1.)/GAMMAC + 1.)*GAMMAC + 1.)*GAMMAC
C C EQUATION (78)
C NU = ((CAPR/W1)*TSP/CAPL)*ALPHA + 0.5 - ((Z2 - QC)/(2.*WC))
67 1 = (GAMMA1 + 1.)/(2.*GAMMA1)*((CP1*TSP)/CAPL)
68 2 = FF/Y1)/SQRT(WRAT)
69 TR = T11/TC
CALL FITMU(MRR,TR)
CALL MIXMU(MRR,TR)
C C EQUATION (68)
C TILDA = 1.5*MRR*SQRT(CAPR*SIG*T(J))/P(J)
70 C C EQUATION (71)
C C C EQUATION (75)
C C C EQUATION (75)
C
FUNCTION GRS(ZZ)

FUNCTION GRS CALCULATES THE GROWTH RATE FOR SMALL DROPLETS (I.E., Z < 1.1)

COMMON/SIGCOM/SIGMA(10),SIG,PSTU,COM/EOCOM/DELTA,T,CAPL,NUS,PSTU,DROT,
1 LTR,LTILDA,KN,THEA,LAGMA,PR,XT,OMEGA,CAPLAM
REAL NUS,LTILDA,KN,LAGMA
COMMON/CNST/TC,K,CAPR,MU,WJ,FBAR,PI,ALPHAC
REAL KIN
COMMON/ADJกอง/กAMMA1,UC,ALPHA,BETAS,STAR,STAR,DT2,
1 J(2),DELX(3),OUTDT,TVRAT,ARAT,JSTOP,JD,CP,CP1
REAL JMIN
COMMON/AJUST/GAMMA,UC,ALPHA,BETAS,STAR,STAR,DT2,
1 J(2),DELX(3),OUTDT,TVRAT,ARAT,JSTOP,JD,CP,CP1
REAL NUS
COMMON/1/(400),X(400),DELN(400),CAPJX(400),R(400),
1 MU(400),TL(400),P(400),RSTAR(400),
2 L(400),S(400),H(400),H(400),RMD(400),CAPJY(400),
3 A(400),ST(400),DTY(400),DEL(400),DEL(400),
4 F(400),ST(400),OUTMCH(400),CAPJY(400)
REAL MU
CALL CCP(T(J),CP,CP1,CP2)
_0,0 = 0.
00 = 0.0
DO 10 J=1,10
WAVG = WAVG + W(IJ)*SIGMA(IJ)
10 CONTINUE
WAVG = WAVG/SIG
W = WC/SIG - SIGMA(J)
WRAT = W/JWAVE
Y1 = SIGMA(J)/SIG
EQUATION (39)
GAMMA = CPI*(CAPR/W(11))
EQUATION (41)
GAMMAC = CPC/(CAPR/WC)
EQUATION (74)
FF = Y1*SOUT(WRAT) + (1. - Y1)*SOUT(W/WAVG)*(GAMMA1/GAMMAC)
1 #((GAMMA1 + 1.)/(GAMMA1 + 1.))*CPC/CPF1*ALPHAC
EQUATION (61)
PS1 = FF*PSJX(I)*CAPR*SIG/T(J)*(GAMMA1 + 1.)/(2.*GAMMA1)
1 #((CAPR/TPH/BAR*(1. - NU)*(TS(J) - T(J))
PSBAR = PSI/(R(I))*STARR(A)
DROT = (RSTAR(J+1) - RSTAR(I))/DELTAT
EQUATION (79)
PSL = 1. + RSTAR(1)/STARR(1)*ZL - 1.)*EXPP(PSIBAR*DELTAT)
1 - (1./PSIBAR*RSTAR(J+1))*DROT*EXPP(PSIBAR*DELTAT) - 1.1
GRS = SDL
RETURN
END
REAL FUNCTION KPJ(B,T)
DIMENSION Bl(3)
CC
EVALUATE EQUATION (35) FOR THE EQUILIBRIUM CONSTANTS
KPJ = EXP(B(1)/T + B(2) + B(3)*T + B(4)*T**2
1 + B(5)*T**3 + B(6)*T**4)
RETURN
END

SUBROUTINE MIXLAM(LAMDA,LAMDAI)
SUBROUTINE MIXLAM CALCULATES THE THERMAL CONDUCTIVITY
OF THE MIXTURE
DIMENSION LAMDAI(6)
REAL LAMDA,LAMDAI
COMMON/SIGCOM/SIGMA(IO),SIG,PSUPO
EQUATION (67)
SUM = 0.
DO 10 I=1,10
IF(LAMDAI(I).EQ.0.) GO TO 10
SUM = SUM + SIGMA(I)/LAMDAI(I)
10 CONTINUE
LAMDA = SIG/SUM
SUM = 0.
DO 20 I=1,10
SUM = SUM + SIGMA(I)*LAMDAI(I)
20 CONTINUE
LAMDA = 0.5*(SUM/SIG + LAMDA)
RETURN
END

SUBROUTINE MIXMU(MUI,MUI)
SUBROUTINE MIXMU CALCULATES THE VISCOSITY OF THE
GASEOUS MIXTURE
COMMON/CONST/TC,K,CPAP,M1,M10),RHOL,RBAR,PI,ALPHAC
REAL K,M1
DIMENSION MUI(IO)
REAL MUI
COMMON/SIGCOM/SIGMA(IO),SIG,PSUPO
EQUATION (64)
MU = 0.
DO 20 I=1,10
SUM = 0.
DO 10 J=1,10
IF(J.EQ.I) GO TO 10
IF(MUI(J).EQ.0.) GO TO 10
PHI = (1. + SQRT(MUI(I)/MUI(J)))/(W(J)/W(I))**0.25)**2
PHI = PHI/(SQRT(8.)*SQRT(1. + W(1)/W(J)))
SUM = SUM + SIGMA(J)*PHI
10 CONTINUE
IF(SIGMA(I).EQ.0.) GO TO 20
MU = MU + MUI(I)/(1. + SUM/SIGMA(I))
20 CONTINUE
RETURN
END
SUBROUTINE NUCRAT(CAPJ, T, P)

SUBROUTINE NUCRAT Calculates the nucleation rate, J

COMMON/SIGCOM/SIGMA(10), SIGMA(1), SIGMA(11), SIGMA(13), SIGMA(12), SIGMA(14), SIGMA(15), SIGMA(16), SIGMA(17), SIGMA(18), SIGMA(19), SIGMA(20), SIGMA(21), SIGMA(22), SIGMA(23), SIGMA(24), SIGMA(25), SIGMA(26), SIGMA(27), SIGMA(28), SIGMA(29), SIGMA(30), SIGMA(31), SIGMA(32), SIGMA(33), SIGMA(34), SIGMA(35), SIGMA(36), SIGMA(37), SIGMA(38), SIGMA(39), SIGMA(40), SIGMA(41), SIGMA(42), SIGMA(43), SIGMA(44), SIGMA(45), SIGMA(46), SIGMA(47), SIGMA(48), SIGMA(49), SIGMA(50), SIGMA(51), SIGMA(52), SIGMA(53), SIGMA(54), SIGMA(55), SIGMA(56), SIGMA(57), SIGMA(58), SIGMA(59), SIGMA(60), SIGMA(61), SIGMA(62), SIGMA(63), SIGMA(64), SIGMA(65), SIGMA(66), SIGMA(67), SIGMA(68), SIGMA(69), SIGMA(70), SIGMA(71), SIGMA(72), SIGMA(73), SIGMA(74), SIGMA(75), SIGMA(76), SIGMA(77), SIGMA(78), SIGMA(79), SIGMA(80), SIGMA(81), SIGMA(82), SIGMA(83), SIGMA(84), SIGMA(85), SIGMA(86), SIGMA(87), SIGMA(88), SIGMA(89), SIGMA(90), SIGMA(91), SIGMA(92), SIGMA(93), SIGMA(94), SIGMA(95), SIGMA(96), SIGMA(97), SIGMA(98), SIGMA(99), SIGMA(100)

REAL A, B


REAL K, M

COMMON/MOD/ALPHA, BETA, DSTAR, ASTAR, DT2, JMIN, DELX, EPS(0), IDUT, DT1, CVRAT, ARAT, JSTOP, JDB, CP, CP1

REAL JMIN

EQUATION (57)

PINF = EXP(35.897 - 664.7/T - 4.4864*LOG(T))

PBAR = (P/PINF)*(SIGMA(1)/SIG)

EQUATION (58)

TR = T/TC

SIGMAX = (182.27 + 10.412*TR - 256.889*TR**2 + 95.928*TR**3)

EQUATION (59)

RSTARX = 2.5*SIGMAX/(RHOBAR*TR*LOG(PBAR))

CALL CHG(T, HGX, HG1)

EQUATION (56)

CAPL = HG1 - 1.2E+3*T + 17.175308E+6

RHOX = P/(CAPR*SIG*T)

GAMMA1 = CPI/CP1 - CAPR/(M1))

EQUATION (62)

Q = 2.*(GAMMA1 - 1.)*(CAPL/(M1) - 0.51/(RBA*K*T)

1.*(GAMMA1 + 1.))

WAVG = 0.

DO 10 I = 1, I0

WAVG = WAVG + SIGMA(I)*W(I)

10 CONTINUE

WAVG = WAVG/SIG

WRAT = (W(I)/WAVG)**2

EQUATION (61)

EQUATION (56)

CAPJ = (1./(1. + Q))**(WRAT**2/2)*SIGMAX/(PI*M1**3))

1.*(RHOX**2/RHOL)*EXP(-4.*P1*SIGMAX*RSTARX**2/(3.*K*T))

CAPJ = CAPJ*(SIGMA(1)/SIG)**2

RETURN

END
SUBROUTINE PARTZ

STAGNATION CONDITIONS AND MASS FLOW RATE

COMMON/CONST/TC,K,CAPR,M1,PI,ALPHAC
REAL K,M
COMMON/GAMMA/GAMBAR
COMMON/ADJUST/GAMMA1,GC,ALPHA,BETA,OSTAR,ASTAR,DTZ,
JMIN,JDELX,DELX,DT1,DT1,CYRAT,ART1,JSTOP,JOP,CP1
REAL JMIN
COMMON/SWITCH/SWEND,SWPO,SWGO
INTEGER SWEND,SWPO,SWGO
COMMON/INPUT/TO,P0,PO,XLAST,PHI1,RCM,RND
COMMON/TANDP/TXX,PXX
COMMON/INPT/TO_PO,XLAST,PHI1,URHO_RHOFI,CP,CP1
COMMON/CMOUT/ENTRY,DT,RHOD,T1,U1,USTAR,URHO,URHO_RHOFI
COMMON/CMOUT/INENTRY,URHO_RHOFI,TO
COMMON/SIGCOM/SIGMA(IO),SIG,PSUPO

C COMPUTE ASTAR, AREA AT NOZZLE THROAT

C ENTRY = 0
ASTAR = PI/4.*DSTAR**2

C COMPUTE HO = HG USING EQUATION (42) WITH T = TO
TXX = TO
PXX = P0
CALL CECEMP
CALL CHGITO_HO
RHOD = P0/(CAPR*SIG*TO)

C COMPUTE SO = SG USING EQUATION (44) WITH T = TO, RHO = RHOD
CALL CS(SO,RHOD,TO)

C MAKE INITIAL GUESS FOR T
TI = 0.9*TO
TXX = T1
PXX = P0
DT = DT1
URHD = 0.
CONTINUE
ENTRY = ENTRY + 1
CONTINUE

C COMPUTE DENSITY USING EQUATION (45)
CALL CDEN(RHOFI,SO,T1)

C COMPUTE PRESSURE USING EQUATION (47)
P = RHDFI*CAPR*SIG*TI
DIFF = ABS(PXX - P)

C TEST FOR CONVERGENCE ON PRESSURE
IF(DIFF/P,LT.EPSI_) GO TO 30
PXX = P
GO TO 20

C COMPUTE FLOW VELOCITY USING EQUATION (103)
UF = SQRT((RHO - HGFI)/H1)
H1 = HCF
U1 = UF

C COMPUTE MASS FLUX
URHO = RHDFI*UF
GAMBAR = ALOG(T1/TO1)/ALOG(P/P0)
GAMBAR = 1./U1 - GAMBAR
IF(DTOUT(1),1,1) AND (SWPO.EQ.1) CALL PRTOUt(1,2)

C IF PAST A MAXIMUM FOR MASS FLUX BRANCH TO CHANGE
C RESOLUTION TO LOCATE THE MAXIMUM
IF(URHO_LUT.URHD) GO TO 40

41
URHOX = URHO
T1 = T1 - DT1
TXX = T1
GO TO 10
C
C     PAST A MAXIMUM, RESTORE LAST T AND DECREASE
C     DELTA T - GO BACK AND TRY AGAIN
C
40 TST = ABS(URHO - URHOX)/URHO
      IF(TST.LT.EPS(4)) GO TO 50
      T1 = T1 - 2.5*DT1
      TXX = T1
      DT1 = 0.5*DT1
      URHOX = 0.
      GO TO 10
C
C     HAVE FOUND A MAXIMUM MASS FLUX
C
100 C     SET TSTAR = T, RHOSTR = RH0, USTAR = U
101 C
102 50 TSTAR = T1
103      RHOSTR = RHOF1
104      USTAR = UF
105 C
106 C     COMPUTE MASS FLOW RATE, MDOT USING EQUATION (104)
107 C
108      MDOT = RHOF1*UF*ASTAR
109      IF(SWPO.EQ.1) CALL PRTOUT(I,0)
110      RETURN
111 END
SUBROUTINE PART2

C  C  ISENTROPIC EXPANSION FROM NOZZLE THROAT
C  C
COMMON/SIGCOM/SIGMA(10),SIG,PSUP0
COMMON/SVXZ/XZ2,YZ2
COMMON/GAMDUT/GAMBAR
COMMON/SWEND/SWPO,SWGD
COMMON/PLOTIL/PLOTIL(600),PLOTIL(600),PLOTIL(600),PLOTIL(600)
1  DELTT(600),UDOUT(600),OUTMAC(600)
11 INTEGER SWEND,SWPO,SWGD
12 COMMON/INPT/TO,PO,XLAST,PH11,RMC,RND
13 COMMON/TANDP/TX3,PX3
14 COMMON/OUT1/50,H0,MDOT,RHOSTR,TSTAR
15 REAL RHOSTR
16 COMMON/CMOUT1/NENTRY,DT,RHOD,T1,H1,U1,USTAR,RHOSTR,RHOF1
17 COMMON/ADJUST/GAMMA1,GC,ALPHA,BETA,DSTAR,ASTAR,DT2,
18 JMIN,DELX,EPS(8),IOUT,DT1,CYRAT,ARAT,JSTD,FDB,CP,CP1
19 REAL JMIN
20 COMMON/CONST/TC,K,CAPR,M1,(10),RHO1,RBAR,P,ALPHAC
21 REAL K,M1
22 COMMON/OUT2/U2,RHODZ,CAPZ2,TZ,TZ,RSTAR2,XZ,A2,SIG2111),DUMC
23 COMMON/CHOUT2/NE,TZM, SigMAX,UZM,RHODZ,P,PINF,PBAR,TR,H2K,RSTAR,CP,L,
24 REAL J
25 REAL J
26 C
27 1 FORMAT(1HO,****** ARRAY SIZE EXCEEDED IN PART 2 *****)
28 2 FORMAT(1HO,****** END OF NOZZLE ENCOUNTERED WITH INSUFFICIENT
29   J VALUE OF J *****)
30 C
C START WITH T = TSTAR AND RHO = RHOSTR
31 C
C HENTRY = 0
32 C NENTRYZ = 0
33 IBR = 0
34 T0LO = 0.
35 TZ_ = -TSTAR
36 C DECREASE T BY DT2
37 C
38 NE = 0
39 P = RHOSTR*CAP*SIG*TSTAR
40 10 CONTINUE
41 NE = NE + 1
42 T2W = T2W - DT2
43 TXX = T2W
44 20 CONTINUE
45 PXX = P
46 CALL CCCEMP
47 C
48 C CALCULATE DENSITY USING EQUATION (45)
49 C
50 CALL CDEN(RHOZZ,50,T2W)
51 C
52 C CALCULATE PRESSURE USING EQUATION (47)
53 C
54 PZZ = RHOZZ*CAP*SIG*T2W
55 DIF = ABS(P - PZZ)
56 IF(DIF/PZZ LT EPS(5)) GO TO 30
57 P = PZZ
58 GO TO 20
59 30 CONTINUE
60 C
61 C CONVERGENCE IN PRESSURE ITERATION
62 C CALCULATE ENTHALPY USING EQUATION (42)
63 C
64 CALL CHGT(T2W,HG,DUM)
65 C
66 C CALCULATE FLOW VELOCITY USING EQUATION (103)
67 C
68 C U2W = SQRT(T2W*(HO - HG))
69 H2W = HG
70 RHD = RHOZ
71 NOUT = NE
72 IOUT(NE) = T2W/T0
73 POUT(NE) = P/PO
74 UDOUT(NE) = U2W
75 CALL SATTEMITS(P)
76 DELTT(NOUT) = TS - T2W
77 IF(DELTT(NOUT) LT 0.) DELTT(NOUT) = 0.
78 C
79 C CALCULATE CPI USING EQUATION (38)
CALL CCP(TZW, CP1, CP1, CPC)

CALCULATE GAMMA1 USING EQUATION (39)

GAMMA1 = CP1/(CP1 - CAPR/W(1))

OUTMAC(NOUT) = GAMMA1*CAPR*SIG*TZW

OUTMAC(NOUT) = U2W/SORT(OUTMAC(NOUT))

CALCULATE PINF USING EQUATION (57)
PINF = EXP(55.897 - 6641.7/TZW - 4.886*ALOG(T2M))

PBAR = (P/PINF)*(SIGMA1/SIG)

CALCULATE AREA USING EQUATION (105)

ALST = MOOT/(RHO*U2W)

GET X POSITION CORRESPONDING TO THIS AREA

CALL XYSA(ALST, XLST)

IF(XLIST - XOLD) .GT. 0.9#DT2

XOLD = XLIST

IF(NOUT .GT. 595) WRITE(6, (4))

IF(INOUT .EQ. 0) STOP

XOUT(INOUT) = XLST

X2 = XLST

GARBAR = ALOG(TZW/TO)/ALOG(P/PO)

GAMBAR = 1./(1. - GAMBAR)

NENTRY = NENTRY + 1

IF(PBAR .GE. 1.0 AND NENTRY .EQ. 0) NENTRY = 1

IF(DOUT.IN.GT.6) CALL PRTOUT(Z)

IF(IBR .EQ. 0) AND XZ .GT. XLAST)

WRITE(Z)

IF(IBR .EQ. 0) AND XZ .GT. XLAST) STOP

END

CALCULATE SIGMA FROM EQUATION (59)

SIGMAX = (82.27 + 75.62*TR - 25.88*TR**2 + 5.928*TR**3)

CALCULATE HG1 USING EQUATION (43)

CALL CHG(TZW, DUH, HG1)

CALCULATE LATENT HEAT OF EVAPORATION USING EQUATION (56)

CAPL = HG1 - 4.2E+3*TZW + 17.11753658E+6

CALCULATE CRITICAL DROPLET RADIUS USING EQUATION (59)

RSTAR = 2.*SIGMAX/(RHO*PBAR*TZW*ALOG(PBAR))

CALL CCP(TZW, CP1, CPC)

GAMMA1 = CP1/(CP1 - CAPR/W(1))

WAVG = 0.

DO 40 I=1,10

WAVG = WAVG + SIGMA(I)*W(I)

40 CONTINUE

WAVG = WAVG/SIG

WRAT = (W(1/L)*WAVG)**2

CALCULATE Q USING EQUATION (62)

U = 2.*GAMMA1 - 1.)*CAPL/(CAPL*IRBAR*TZW) - 0.5)/(IRBAR*TZW

1.)*(GAMMA1 + 1.))

CALCULATE NUCLEATION RATE USING EQUATION (61)

J = (1./(1. + Q))*Q*WRAT*SORT(2.*SIGMAX/(PI*M1**3))/(RHO*W2)

1./RHO*W2*EXP(-4.4*PI*M1*SIGMAX*STAR**2/I3.**4*TZW))

J = J*(SIGMA1/SIG)**2

IF(DOUT.IN.GT.6) CALL PRTOOUT(Z, 2)

END

BRANCH BACK IF J.LT. JMIN
IF(J.LT.JMIN) GO TO 10
IF(IBR.EQ.1.AND.XLST.LE.XLAST) GO TO 10
IF(IBR.EQ.1.AND.XLST.GT.XLAST) GO TO 60
U2 = U2W
XSAV = XLST
AZ = ALST
KHO2 = KHO
CAPJ2 = J
T2 = TZW
P2 = P
RSTAR2 = RSTAR
GUMC = OUTMAC(NOUT)
DO 50 IJ=1,10
SIGZ(IJ) = SIGMA(IJ)
50 CONTINUE
SIGZ(11) = SIG
IBR = 1
DO 70 IJ=1,10
SIGMA(IJ) = SIG2(IJ)
70 CONTINUE
SIG = SIG2(11)
XZ = XSAV
IF(SWPO.EQ.1) CALL PRTOUT(Z)
SUBROUTINE PART3

NUCLEATION AND DROPLET GROWTH

COMMON/COEFFCD/COEFF(10,6,2)
COMMON/GAMOUT/GAMBAR
DIMENSION HL(I)(400),SLI(600),T(J)(400),YI(J)(400)
COMMON/SIGMAM/SIGMA(10),SIG,PSUP0
COMMON/EGCDM/DELTAT,CAPL,NU,PS,DS,T,
1 TLITILDA,KH,THETA,LAMBDA,PA,XI,OMEGA,CAPLAM
COMMON/INPT/TO,PS,XLAST,PHI11,RH1,RND
COMMON/SWICH/SWEND,SWPO,SWGU
INTEGER SWEND,SWPO,SWGO
COMMON/CUT2U2,RHOD2,CAPJ2,T2,P2,STAR2,Z2,A2,SIG2(11),UUMC
COMMON/CMT3U3/NENT
COMMON/CUT1/50,HG,MDT,MRH,STAR,TSTAR
REAL MDT
COMMON/CMT1U1/ENTRY,DT,RHOD,T2,H1,U1,USTAR,H2,STAR1,RHOP1
DIMENSION SIGM(10)
COMMON/ADJUST/GAMMA,ALPHA,ALPHA-Star,STAR,OT2,
1 JMIN,DELY,EPS(8),TOUT,DT1,CVRAT,ARAT,STOP,JD,CP,CPI
REAL JMIN
COMMON/CONST/TCR,PHI1,MU1,RH01,RBAR,PI1,ALPHAC
REAL K,M1
COMMON I,XJ,A(400),X(400),DELNX(400),CAPJX(400),R(400),
1 MU(400),TL(400),T(400),P(400),STAR(400),
2 U(400),S(400),SL(400),HL(400),RHOD(400),CAPJY(400),
3 F(400),G(400),Y(400),DELTY(400),DELS(400),
4 RP(400),TS(400),OUTMCH(400),CAPPJ(400)
REAL MU
REAL LBAR
DIMENSION INDTYP(400)

1 FORMAT(' **** PROGRAM STOP... EXCESSIVE LOOPING *****')
2 FORMAT('#### BEFORE DBLT')
3 FORMAT(IH,T(I),T(J),T(J1),J(J1),15.3E16.8)
4 FORMAT(' RSTER(J),CAPJ(J),CAPJ(J),15.3E16.8)
5 FORMAT(' DELNX(I,J),T(J1),T(J1),15.3E16.8)
6 FORMAT(' F(J,J1),G(J,J1),YS(J1),15.3E16.8)
7 FORMAT(' HL(J,J1),SL(J,J1),DELY(J),15.3E16.8)
8 FORMAT(' RHOD(J1),AT(J1),15.3E16.8)
9 FORMAT(' RTL(J1),15.3E16.8)
10 FORMAT(' (TL(J1),15.3E16.8)
11 FORMAT(' (SL(J1),15.3E16.8)
12 FORMAT(' (HL(J1),15.3E16.8)
13 FORMAT(' (Y(J,J1),15.3E16.8)
14 FORMAT(' AFTER DBLT')
MAXJ = 1000
JTOP = MAXJ
DO 20 JD=1,400
RPERM(JD) = 0.
20 CONTINUE

C TAKE VALUES FROM PART 2 AS A STARTING POINT

57 T(I) = T2
58 P(I) = P2
59 RHOD(J) = RHOD2
60 U(I) = U2
61 V(I) = V2
62 A(J) = A2
63 OUTMC(J) = UUMC
64 X(J) = X2
65 STAR(J) = STAR2
66 YS(J) = 0.

C CHEMICAL COMPOSITION REMAINS CONSTANT THROUGHOUT

C PART 3 EXCEPT FOR FORMATION OF LIQUID WATER

C SAVE PART 2 RESULTS

DO 30 JD=1,10
SIGSV(J) = SIGM(J)
30 CONTINUE

C EQUATION (18)
DO 50 JMK = 2, 10
   SIGMA(JMK) = SIGMSV(JMK)/(1. - YS(JMK))
50 CONTINUE

C EQUATION (17)
C
SIGMA(1) = SIGMSV(1) - YS(J1)/0.018
SIGMA(J1) = SIGMA(1)/(1. - YS(J1))
SIG = SIGMSV - YS(J1)/0.018
SIG = SIG/(1. - YS(J1))
CALL SATTEM(TS(J1), P(J1))
UM(J1) = U(J1)
C INCREASE THE SIZE OF DELTA X FOR POSITIONS 20
C STEPS PAST THE DROPLET FORMATION CUTOFF
C
IF(ICHG.LT.15.AND.(J-Z0).GT.MAXJ) DELX = 1.08*DEIX
IF((J-Z0).GT.MAXJ) ICHG = ICHG + 1
C INCREASE X BY DELTA X, EQUATION (106)
C
X(J+1) = X(J) + DELX
C IF AT END OF NOZZLE, CALCULATION IS FINISHED
C
IF(X(J+1).GT.XLAST) GO TO 240
FRCBND = 1.
IF(J.EQ.1) GO TO 60
C
FRCBND = (X(J) - X(J-1))/(X(J+1) - X(J))
60 CONTINUE
C
GET THE CROSS SECTIONAL AREA FOR THE NEW
C X POSITION, EQUATION (107)
C
CALL AVSX(X(J+1), AJ(J+1))
CALL SATTEM(TS(J), P(J))
PFIN = EXP(55.807 - 6641.7/T(J)) - 4.4864*ALOG(T(J))
PBAR = (P(J)/PFIN)*SIGMA(1)/SIG
TR = T(J)/TC
SIGMAX = (0.227 + 75.612*TR - 256.892*TR**2 + 95.928*TR**3)*
1.181E-3
RSTAR(J) = 2.*SIGMAX/(RHOL*RBAR*SIG*TR**3)*
R(J) = 1.001*RSTAR(J)
CALL GRLPRP
IF(J.NE.1) GO TO 80
CALL AVSX(X(J)) - DELX, AJ(J)
CAPGAM = MDT/AJM
TAUOLD = T(J)
70 CONTINUE
C
CALL CDEHN(RHOF, SO, TAUOLD)
CALL CHG(TAUOLD, HGF, DUM)
CALL CEPTAUOLD(C1, C1PCC)
UF = SORT(2.41*HOF - HGF)
FF = RHDF/UF - CAPGAM
DRHDFD = (C1 - CAPR*SIG)*RHDF/TAUOLD
DUDT = -C9/UF
FDF = UF*DRHDFD + RHDF*DUDT
DEL = ABS(TAUNEW-TAUNOLD)/TAUNEW
TAUNOLD = TAUNEW
IF((DEL.GT.0.016)) GO TO 70
TJM1 = TAUNEW
PJM1 = RHDF*CAPR*SIG*TAUNEW
147 GO TO 90
80 CONTINUE
TJM1 = T(J-1)
PJM1 = P(J-1)
90 CONTINUE
TJM1 = (TJM1 + (FRCBND - 1.)*T(J))/FRCBND
PJM1 = (PJM1 + (FRCBND - 1.)*P(J))/FRCBND
TJM = 0.5*(TJM1 + T(J))
PJM = 0.5*(PJM1 + P(J))
C
GET NUCLEATION RATE AT J - 1/2
C
CALL NUCRAT(CAPRJX(J), TJM, PJM)
DO 100 I = 1, J
RPERM(I) = R(I)
100 CONTINUE
100 NFLOOP = 0
110 CONTINUE
NFLOOP = NFLOOP + 1
IF(NFLOOP.GT.4000) WRITE(6, 1)
111

C GET NUCLEATION RATE AT J + 1/2
C CALL NUCRAT(CAPJY(J),TJP,PJP)
DELN(J) = 0.
CAPP(J) = 0.
IF(CAPJY(J).LT.1.1) GO TO 120
IF(CAPJX(J).LT.1.1) GO TO 120
C CALCULATE MEAN NUCLEATION RATE, EQUATION (108)
C CAPP(J) = (CAPJY(J) - CAPJX(J))ALOG(CAPJY(J)/CAPJX(J))
C CALCULATE NUMBER OF DROPLETS FORMED, EQUATION (109)
C DELN(J) = CAPP(J)*A(J)*DELX/MDOT
120 CONTINUE
C SET FLAG TO INDICATE DROPLET FORMATION HAS STOPPED
C IF(DELN(J).EQ.0..AND.,MAXJ.EQ.1000) MAXJ = J
C CALCULATE R* FOR DROPLET GROWTH
C EQUATIONS (57) - (59)
C PINF = EXPP(55.897 - 6641.7/T(J+1) - 4.4864*ALOG(T(J+1)))
PBAR = (P(J+1)/PINF)*SIGMA(I)/SIG
TR = T(J+1)/TC
SIGMAX = (82.27 + 75.612*TR - 256.88*TR*2 + 95.928*TR*3)1.1E-3
RSTAR(J) = 2.*SIGMAX/(RHO*RBAR*T(J+1)*ALOG(PBAR))
C CALCULATE THE TIME STEP CORRESPONDING TO THIS
C DELTA X, EQUATION (110)
C DELTAT = 2.*DELX/(U(J) + U(J+1))
DD 130 I=1.J
R(I) = RPERM(I)
130 CONTINUE
C CALL GRLPRP
DO 140 I=1.J
IF(NLOOP.EQ.I) INDTP(I) = 27
140 CONTINUE
C SET DROPLET RADIUS TO ZERO FOR POSITIONS PAST
C THE DROPLET FORMATION CUTOFF
C IF(J.GT.MAXJ) R(J) = 0.
JTOP = J
IF(J.GT.MAXJ) JTOP = MAXJ
DD 160 I=1.JTOP
IF(R(I)/RSTAR(J).LT.1.0 AND NLOOP.EQ.I) INDTP(I) = 26
IF(INDTP(I).EQ.26) GO TO 150
Z = 1.0001
C CALCULATE DROPLET GROWTH FOR LARGE
C DROPLETS, EQUATION (72)
C IF(NLOOP.EQ.I)
1CALL GRL(I)
C IF(NLOOP.EQ.I)
1R(I) = Z*GRS(I)
1IF(NLOOP.EQ.I) RPERM(I) = R(I)
GO TO 160
160 CONTINUE
C IF(R(I).EQ.0.) GO TO 160
Z = R(I)/RSTAR(J)
C CALCULATE DROPLET GROWTH FOR SMALL
C DROPLETS, EQUATION (79)
C Z2 = GRS(I)
IF(Z2.LT.0.) Z2=0.
R(I) = Z2*RSTAR(J)
160 CONTINUE
R(J+1) = 1.001*RSTAR(J+1)
RHO(J+1) = P(J+1)/(CAPR*SIGT(J+1))
FPJ(J+1) = 0.
JP1 = JTOP + 1
C CALL NUCRAT(CAPJY(J+1),T(J+1),P(J+1))
DELN(J+1) = A(J+1)*DELX*CAPJPI/NDOT
CALCULATE MASS OF DROPLETS FOR TYPE 1

\[ \text{MU}(I) = (4/3) \pi \rho_{\text{HOL}} \text{R}(I)^3 \]

CALCULATE MASS RATIO OF LIQUID WATER FOR TYPE 1, EQUATION (117)

\[ \text{YIJ}(I) = \text{MU}(I) \times \text{DEL}(I) \]

CALCULATE TOTAL MASS RATIO OF LIQUID WATER, EQUATION (118)

\[ \text{YSIJ} = \text{YIJ}(I) + \frac{\text{YS}(J+1)}{\text{YS}(J)} \]

CONTINUE

\[ \text{DEL}(J) = \text{DEL}(J+1) - \text{DEL}(J) \]

DO 200 I = 1, JPP1

TL(I) = T(J+1) + DEL(I) * TL(I)

IF(I,GT.JPP1) GO TO 200

IF(R(1).LT.RSTAR(J+1)) GO TO 190

KN = LTILDA(JZ, *R(I))

\[ \Delta = X_{N\text{K}} \text{P}R \]

\[ \text{CALL CCP(TIJ, CP, CP1, CPC)} \]

WAVG = 0.

W = 0.

UD = 100 I = 1, 10

WAVG = WAVG + W(IJ) * SIGMA(IJ)

IF(IJ.NE.I) WC = WC

WAVG = WAVG/\text{SIG}

WC = WC/\text{SIG}

\[ \text{WRAT} = W(I)/WAVG \]

\[ \text{YI} = \text{SIGMA}(I)/\text{SIG} \]

\[ \text{GAMMA1} = \text{CP1}/\text{CP} - \text{CAPR}/W(I) \]

\[ \text{GAMMAC} = \text{CP}/\text{CP} - \text{CAPR}/W(C) \]

\[ \text{FFF} = Y(1) \times \text{SORT(WRAT)} + (1. - Y(1)) \times \text{SORT(WC/WAVG)} \times \text{GAMMA1}/\text{GAMMAC} \]

\[ 1 \times (\text{GAMMA1} + 1)/((\text{GAMMA1} + 1) \times \text{CP}1/\text{CP} + \text{ALPHAC}) \]

\[ \Delta = \text{DEL}(X_{\text{N\text{K}}}) + 2.5 \times \text{ALPHA} \]

\[ \text{TL}(I) = T(J+1) + (1/1 - NU \times \text{DEL}) \times T(J+1) \]

100 CONTINUE

CALCULATE ENTHALPY OF LIQUID, EQUATION (119)

\[ \text{HLI}(I) = 4.2 \times 10^3 \times \text{TL}(I) - 17,117,356,583 \times 10^6 \]

\[ \text{HL}(J+1) = \text{HL}(J) + \text{YIJ}(I) \times \text{HLI}(I) \]

CALCULATE ENTROPY OF LIQUID, EQUATION (120)

\[ \text{SLI}(I) = 4.2 \times 10^3 \times \text{ALOG}(\text{TL}(I)) - 2.001496280 \times 10^6 \]

\[ \text{SL}(J+1) = \text{SL}(J) + \text{YIJ}(I) \times \text{SLI}(I) \]

200 CONTINUE

CALCULATE CHANGE IN ENTHALPY, EQUATION (83)

\[ \text{DELS}(I) = (\text{LBAR} - \text{CPZ} \times \text{TSJ} - \text{TJ}) \times \text{DELY}(J) \]

\[ 1 \times (\text{TJ} - \text{TJ} + 1)/\text{TJS} \]

CALCULATE CHANGE IN ENTROPY, EQUATION (83)

\[ \text{DELS}(I) = (\text{LBAR} - \text{CPZ} \times \text{TSJ} - \text{TJ}) \times \text{DELY}(J) \]

\[ 1 \times (\text{TJ} - \text{TJ} + 1)/\text{TJS} \]

CALCULATE ENTROPY OF THE MIXTURE, EQUATION (124)

\[ \text{S}(J+1) = \text{S}(J) + \text{DELS}(I) \]

\[ \text{TOLD} = \text{T}(J+1) \]

IF(IJ.LT.JPP) GO TO 210

WRITE(6,2)

WRITE(6,3) J, TOLD, T(J+1), P(J+1)

WRITE(6,4) RSTAR(J+1), CPX(I), CPY(I)

WRITE(6,5) DELM(I), U(I+1), TJS(I+1), CP1
335  WRITE(6,6) F(J+1),C(J+1),YS(J+1),DELY(J)
336  WRITE(6,7) HL(J+1),SL(J+1),DELS(J),SL(J+1)
337  WRITE(6,8) RH(J+1),A(J+1)
338  WRITE(6,9) R(IJK),IJK=1,J
339  WRITE(6,10) (TL(IJK),IJK=1,J)
340  WRITE(6,11) (SL(IJK),IJK=1,J)
341  WRITE(6,12) (ML(IJK),IJK=1,J)
342  WRITE(6,13) (YI(JJK),IJK=1,J)
343
344  CONTINUE
345
346  CALL DLBL
347  CALL CCP(T(J+1),CP5,CP1,CPG)
348  GAMMAS = CP1/(CP - CAPR/W(J+1))
349  OUTMCH(J+1) = GAMMAS*CAPR*SIGM(T(J+1))
350  OUTMCH(J+1) = U(J+1)/SQRT(OUTMCH(J+1))
351  IF(JL.TJ0) GO TO 220
352  WRITE(6,14)
353  WRITE(6,3) JTOLG,T(J+1),P(J+1)
354  WRITE(6,4) KSTART(J+1),CAPJX(J),CAPJY(J)
355  WRITE(6,5) DELN(J),U(J+1),TS(J+1),CPA
356  WRITE(6,6) G(J+1),G(J+1),YS(J+1),DELY(J)
357  WRITE(6,7) HL(J+1),SL(J+1),DELS(J),SL(J+1)
358  WRITE(6,8) RHO(J+1),A(J+1)
359  WRITE(6,9) R(IJK),IJK=1,J
360  WRITE(6,10) (TL(IJK),IJK=1,J)
361  WRITE(6,11) (SL(IJK),IJK=1,J)
362  WRITE(6,13) (YI(JJK),IJK=1,J)
363
364  C  CONVERGENCE TEST
365  C  DEL = (T(J+1) - TOLD)/TOLD
366  TOLD = T(J+1)
367
368  C  TEST FOR TEMPERATURE CONVERGENCE EQUATION (133)
369  C  IF NO CONVERGENCE GO BACK TO DROPLET GROWTH
370  C
371  C  IFABS(DEL,GT,EPS(7)) GO TO 110
372  C  CONVERGENCE...OUTPUT RESULTS AND GO ON
373  C  TO THE NEXT VALUE OF J
374  C  WRITE(7) J,(R(I),ML(I),HL(I),SL(I),TL(I),YI(I),I=1,J)
375  GAMBAR = ALOG(T(J+1)/T0)/ALOG(P(J+1)/P0)
376  GAMBAR = 1./(1. - GAMBAR)
377  IF(TOUT.EQ.2.AND.SWP.EQ.1) CALL PRTOUT(3,1)
378  UO 230 = 1
379  IF(INDTYP(1).EQ.36 | R(1).GT.RR(1)) CALL PRTOUT(3,1)
380  C  CONTINUE
381  C  IF AT TERMINAL J, CALCULATION IS FINISHED
382  C  IF(J,GT,JS) GO TO 240
383  C  GO TO 40
384  C  CONTINUE
385  C  IF(SWP.EQ.1) CALL PRTOUT(3,0)
386  RETURN
387  END
SUBROUTINE PLTOUT

SUBROUTINE PLTOUT GENERATES PLOT OUTPUT OF PROGRAM RESULTS. SEVERAL SUPPORTING SUBROUTINES ARE A PART OF THE LANGLEY GRAPHICS SYSTEM. THESE INCLUDE:

- ASCALE
- AXES
- CALPLOT
- CHARST3
- CHARST4
- CHARST12
- DASHPLT
- LINLINE
- NFRAPE
- NOTATE
- PNTPLT
- REPCHAR

FOR MORE INFORMATION ON THESE ROUTINES SEE "LANGLEY GRAPHICS SYSTEM", CENTRAL SCIENTIFIC COMPUTING COMPLEX DOCUMENT G-3.

COMMON I,J,A(400),X(400),DELN(400),CAPX(400),R(400),
1 MU(400),TL(400),T(400),PI(400),RSTAR(400),
2 F(400),G1(400),YS1(400),DEL4(400),DELS1(400),
3 RPERR(400),TS(400),OUTCH(400),CAPPJ(400),
4 REAL M

COMMON DUM(400)

DIMENSION DUM(500)

COMMON/SVXZ/XZZ/YP(400)

DIMENSION PAT(Z),PATX(4),PATY(4)

DATA PAT/O.059,0.25/

DATA PATX/O.125,0.1,

DATA PATY/O.58,0.58,0.1,/

CALL CHARS12

CALL REPCHAR(2,12)

CALL CHARST4

CALL CHARST3

NP = J - 1

XSV = X(J)

X(J) = 0.

CALL ASCALE(X,8.,NP,1,10.)

IF(XMIN.NE.0.,OR.XMAX.NE.0.) X(NP+1) = XMIN

IF(XMIN.NE.0.,OR.XMAX.NE.0.) X(NP+2) = (XMAX - XMIN)/8.

XOUT(NOUT+1) = (XMIN+1)

XOUT(NOUT+2) = X(NP+2)

X(J) = XSV

IF(IPSLCT(2).EQ.0) GO TO 40

10 CONTINUE

YP(IJ) = POUT(IJK) + (X(IJ) - XOUT(IJK))

1 *(YP(IJK+1) - POUT(IJK))/XOUT(IJK + 1) - XOUT(IJK))

YP(IJ) = P(IJ)/YP(IJ)*PO

30 CONTINUE

CALL ASCALE(YP,6.,NP,1,10.)

IF(PRMIN.NE.0.,OR.PRMAX.NE.0.) YP(NP+1) = PRMIN

IF(PRMIN.NE.0.,OR.PRMAX.NE.0.) YP(NP+2) = (PRMAX - PRMIN)/6.

CALL AXES(0.,0.,0.,X(NP+1),X(NP+2),1.,0.,0.,1.)

1 IXK,0.2,-1)

CALL AXES(0.,0.,0.,YP(NP+1),YP(NP+2),1.,0.,0.,1.)

1 THR RATIO=0.2,7)

CALL LINLINE(X,YP,NP,1,0,0,0)
C     CRITICAL RADIUS VS X
C
DO 50 J=1,NP
YP(IJ) = RSTAR(IJ)
50 CONTINUE
C
CALL ASCALE(YP(6..NP+1,10.), RMIN)
IF(RMIN.NE.0. OR RMAX.NE.0.1) YP(NP+1) = RMIN
IF(RMIN.NE.0. OR RMAX.NE.0.) YP(NP+2) = (RMAX-RMIN)/6.
CALL AXES(0.0,0.0,0.0,X(NP+1),X(NP+2),1.0,1.0)
CALL CALPLT(4.5,1.5,-3)
CALL NFRAME
CALL CALPLT(4.5,1.5,-3)
IF(IPSLCT(7).EQ.0) GO TO 60
C
C     DROPLET RADIUS VS X
C
NPP = NP - IRAD + 2
YP(IRAD-1) = 0.
DO 124 IJ=1,NPP
KRAD = IRAD + IJ - 2
YP(KRAD) = ALOG10(YP(KRAD))
124 CONTINUE
C
IF(IRAD.EQ.1) CALL ASCALE(YP(IRAD-1),6..NPP,1,10.)
IF(IRAD.EQ.1) GO TO 100
IF(RMIN.NE.0. OR RMAX.NE.0.) YP(NP+1) = RMIN
IF(RMIN.NE.0. OR RMAX.NE.0.) YP(NP+2) = (RMAX-RMIN)/6.
100 CONTINUE
C
CALL AXES(0.0,0.0,0.8,X(NP+1),X(NP+2),1.0,0.1)
1 'DISTANCE ALONG X OZ X M, Z, Z;
CALL AXES(0.0,0.0,0.8,X(NP+1),X(NP+2),1.0,0.1)
2 'DROPLET IRADIUS, SDSO GD SUSIN AND SDSO GR ND M, M',
3 0.2,55
NPP = NPP - 1
YP(IRAD), RMIN) = ALOG10(RSTAR(IRAD))
C
CALL LINPLT(X(IRAD),YP(IRAD),NPP,1.0,0.0,0.0)
IRAD = IRAD + 10
120 CONTINUE
DO 146 IJ=1,NP
YP(IJ) = RSTAR(IJ)
146 CONTINUE
C
IF(IPSLCT(19).EQ.0) GO TO 280
C
C     TEMPERATURE OF LIQUID VS X
C
DO 158 IJ=1,NP
YP(IJ) = T5(IJ)
158 CONTINUE
C
NPP = NP/10
NPP = NPP - 1
IF(IRAD.GT.10) NPP = 10
IRAD = 5
DO 170 IJ=1,NPP
REWIND 7
70 CONTINUE
READ(7,END=80) JJJ,(DUM(IJ)+DUM1+DUM2+DUM3+DUM4+DUM5,1J=1,1JJ)
80 IF(EOF(7).NE.0) GO TO 90
IF(JJJ,LG,IRAO) GO TO 70
YP(JJJ) = DUM(IJ)
90 CONTINUE
NPP = NPP - IRAD + 2
YP(IRAD-1) = 0.
DO 190 IJ=1,NPP
REWIND 7
190 CONTINUE
READ(7,END=80) JJJ,(DUM(IJ)+DUM1+DUM2+DUM3+DUM4+DUM5,1J=1,1JJ)
IF(JJJ,G.E.1) GO TO 200
IF(JJJ,LT,IRAD) GO TO 70
YP(JJJ) = DUM(IJ)
200 CONTINUE
NPP = NPP - IRAD + 2
YP(IRAD-1) = 0.
DO 226 IJ=1,NPP
REWIND 7
226 CONTINUE
READ(7,END=80) JJJ,(DUM(IJ)+DUM1+DUM2+DUM3+DUM4+DUM5,1J=1,1JJ)
IF(JJJ,G.E.1) GO TO 210
IF(JJJ,LT,IRAD) GO TO 70
YP(JJJ) = DUM(IJ)
210 CONTINUE
NPP = NPP - IRAD + 2
YP(IRAD-1) = 0.
DO 246 IJ=1,NPP
REWIND 7
246 CONTINUE
READ(7,END=80) JJJ,(DUM(IJ)+DUM1+DUM2+DUM3+DUM4+DUM5,1J=1,1JJ)
IF(JJJ,G.E.1) GO TO 250
IF(JJJ,LT,IRAD) GO TO 70
YP(JJJ) = DUM(IJ)
250 CONTINUE
NPP = NPP - IRAD + 2
YP(IRAD-1) = 0.
CONTINUE
READ(7,END=170) JJJ,(DUM4,DUM1,DUM2,DUM3,DUM(IJ),DUM5,IJ=1,330)
170 IF(EOF(7).NE.O) GO TO 180
    IF(JJJ.LT.IRAD) GO TO 160
    YP(JJJ) = DUM(IRAD)
    GO TO 180
180 CONTINUE
    NPP = NP - IRAD + 2
    YP(IRAD-1) = 0.
    IF(IRAD.EQ.1) CALL ASCALE(YP(IRAD-1),6.,NPP,1.10.)
    IF(IRAD.EQ.2) GO TO 190
    IF(TLMIN.NE.0..OR.TLMAX.NE.0.) YP(NP+1) = TLMIN
    IF(TLMIN.NE.0..OR.TLMAX.NE.0.) YP(NP+2) = (TLMAX - TLMIN)/6.
190 CONTINUE
    IF(IRAD.EQ.1)
182 ICALL AXES(I0.0,0.0,0.0,XY(NP+1),X(NP+2),1.,0.,
183 1 IHX,0.2,-1)
184 IF(IRAD.EQ.2)
185 ICALL AXES(I0.0,0.0,0.0,XY(NP+1),Y(NP+2),1.,0.,
186 1 IMY,0.2,-1)
187 NPP = NPP - 1
188 CALL LINPLT(X(IRAD),YP(IRAD),NPP,1,0,0,0,0)
189 IRAD = IRAD + 10
200 CONTINUE
    DO 210 IJ=1,NP
210 YP(IJ) = T(IJ)
210 CONTINUE
    CALL LINPLT(X,YP,NP,1,0,0,0,0)
220 DO 230 IJ=1,NOUT
230 TOUT(IJ) = TOUT(IJ)*10
220 CONTINUE
    CMAX = YP(NP+1) + 6.*YP(NP+2)
230 DO 240 IJ=1,NOUT
240 IF(TOUT(IJ).LE.CMAX) GO TO 250
240 CONTINUE
    IJ = 1
240 IBEG = IJ
240 CONTINUE
    DO 250 IJ=1,NOUT
250 IF(KOUT(IJ)=62.0) GO TO 260
250 CONTINUE
    IJ = NOUT
260 IEND = IJ
260 CONTINUE
    NOUTX = IEND - IBEG + 1
270 TSERV = TOUT(IEND+1)
270 TSERV2 = TOUT(IEND+2)
270 TOUT(IEND+1) = YP(NP+1)
270 TOUT(IEND+2) = YP(NP+2)
270 TRPS = XOUT(IEND+1)
270 TRPS6 = XOUT(IEND+2)
270 XOUT(IEND+1) = XOUT(NOUT+1)
270 XOUT(IEND+2) = XOUT(NOUT+2)
270 CALL DASHPLT(XOUT(IBEG),TOUT(IBEG),NOUTX,1,FAT,2)
270 XOUT(IEND+1) = TRPS
270 XOUT(IEND+2) = TRPS6
270 TOUT(IEND+1) = TSERV
270 TOUT(IEND+2) = TSERV2
270 DO 280 IJ=1,NOUT
280 TOUT(IJ) = TOUT(IJ)/10
280 CONTINUE
    CALL NFRAME
    CALL CALPLT(1.5,1.5,3)
290 IF(IPSLCT(IO).EQ.0) GO TO 300
290 CONTINUE
    CALL ASCALE(XD,C,NOUT,1,10.)
290 CONTINUE
    IF(XMIN.NE.0. OR XMAX.NE.0.) XD(NP+1) = XMIN
    IF(XMIN.NE.0. OR XMAX.NE.0.) XD(NP+2) = (XMAX - XMIN)/6.
290 CONTINUE
    CALL ASCALE(YP,6.,NPP,1.10.)
    YP(NP+1) = RHOMIN
    YP(NP+2) = RHOMAX
    CALL AXES(0.0,0.0,8.,XYP(NP+1),XYP(NP+2),1.,0.,
7 1 IHX,0.2,-1)
290 CONTINUE
    CALL AXES(0.0,0.0,8.,XYP(NP+1),YYP(NP+2),1.,0.,
7 1 IMY,0.2,-1)
290 CONTINUE
    CALL AXES(0.0,0.0,8.,XYP(NP+1),YYP(NP+2),1.,0.,
7 1 IHY,0.2,-1)
300 IF(IPSLCT(13).EQ.0) GO TO 340
300 CONTINUE
    CALL NFRAME
    CALL CALPLT(4.5,1.5,3)
310 IF(IPSLCT(10).EQ.0) GO TO 350
310 CONTINUE
    CALL ASCALE(OUTMAC,6.,NOUT,1,10.)
310 CONTINUE
    IF(XMIN.NE.0. OR XMEN.NE.0.) OUTMAC(NOUT+1) = XMIN
320 CONTINUE
    CALL NFRAME
    CALL CALPLT(1.5,1.5,3)
IF (XMIN,NE.,0.,OR,XMAX,NE.,G.) OUTMAC(NOUT+2) = (XMAX - XMIN)/6.
CALL AXES(0.,0.,0.,X(NP+1),X(NP+2),1.,0.,)
1 1MX,0.,-1
CALL AXES(0.,0.,90.,6.,OUTMAC(NOUT+1),OUTMAC(NOUT+2),1.,0.,
4 1 AHMAC(0.,0.,4.)
DO 310 IJ=1,NOUT
IF (OUTMAC(IJ).NE.GE.OUTMAC(NOUT+1)) GO TO 320
310 CONTINUE
I J = 1
320 NOUTX = NOUT - I J + 1
CALL LINPLOT(NOUT(IJ),OUTMAC(IJ),NOUTX,1,0,0,0,0)
DO 330 IJ=1,NP
YP(IJ) = OUTMAC(IJ)
330 CONTINUE
YP(NP+1) = OUTMAC(NOUT+1)
YP(NP+2) = OUTMAC(NOUT+2)
CALL LINPLOT(X,YP,NP,1,0,0,0,0)
CALL NFRAME
CALL CALPLOT(4.,5.,1.,5.,3.)
340 IF (IPSLECT(I),EQ.,0.,AND.,IPSLECT(14),EQ.,0.) GO TO 400
340
C
C
350 CONTINUE
IJ = 1
370 NOUTX = NOUT - I J + 1
CALL DASHPLOT(NOUT(IJ),POUT(IJ),NOUTX,1,PAT,2)
360 CONTINUE
IF (NEXP.EQ.,C.) GO TO 390
DO 380 IJ=1,NEXP
XPLOT = XEXP(IJ)/39.37
XPLOT = (XPLOT - X(NP+1))/X(NP+2)
YPLOT = YEXP(IJ)/PREF
YPLOT = (YPLOT - YP(NP))/YP(NP+2)
380 CONTINUE
CALL PNTPLOT(XPLOT,YPLOT,90.1,1)
390 CONTINUE
XPLOT = (X22 - X(NP+1))/X(NP+2)
YPLOT = (Y22 - YP(NP+1))/YP(NP+2)
CALL PNTPLOT(XPLOT,YPLOT,90.1,1)
390 CONTINUE
CALL CALPLOT(4.,5.,1.,5.,3.)
400 IF (IPSLECT(13),EQ.,0.,AND.,IPSLECT(14),EQ.,0.) GO TO 440
400
C
C
410 CONTINUE
IJ = 1
420 CONTINUE
YP(IJ) = T(IJ)/T0
420 CONTINUE
TMPL = T(PNP)
YPNP = AMIN1(TPNP/T0,TOUT(NOUT))
CALL ASCALE(YP,6.,NP+1)
IF (TMNP,NE.,0.,OR,TMAX,NE.,G.) YPNP = TMIN
IF (TMNP,NE.,0.,OR,TMIN,NE.,0.) YPNP = TMAX-TMIN
YPNP = TNP
YPNP = TNP
ICALL AXES(0.,0.,0.,8.,X(NP+1),X(NP+2),1.,0.,
4 1 1NX,0.,2.,-1)
IF(IPSLCT(3).NE.0)
ICALL AXES(0.,0.,90.,0.,YP(NP+1),YP(NP+2),1.,0.,
1.4MT/TO,0.,2.,4.)
IF(IPSLCT(14).NE.0)
ICALL AXES(-1.,0.,90.,0.,YP(NP+1),YP(NP+2),1.,0.,
1.4MT/TO,0.,2.,4.)
CALL LINPLT(X,YP,NP,1.,0.,0.,0.,0.)
TMP3 = X(NP+1)
TMP2 = X(NP+2)
TMP1 = YP(NP+1)
TMP4 = YP(NP+2)
XOUT(NOUT+1) = TMP1
XOUT(NOUT+2) = TMP2
XOUT(NOUT+1) = TMP3
XOUT(NOUT+2) = TMP4
CMAX = YP(NP+1) + 6.*YP(NP+2)
DO 420 IJ = 1,NOUT
IF(XOUT(IJ) .LE. CMAX) GO TO 430
420 CONTINUE
430 NOUTX = NOUT - IJ + 1
CALL LINPLT(XOUT(IJ),TOUT(IJ),NOUTX,1.,0.,0.,0.,0.)
IF(IPSLCT(14).NE.0) GO TO 440
CALL NFRAME
CALL CALPLT(4.,5.,1.,5.,-3)
440 IF(IPSLCT(12).EQ.0.AND.IPSLCT(14).EQ.0) GO TO 510
C
C FLOW VELOCITY VS X
C
DO 450 IJ = 1,NP
YP(IJ) = U(IJ)
450 CONTINUE
CALL ASCALE(YP,NP,NP,1.,0.1.)
IF(UMIN.NE.0..OR.UMAX.NE.0.) YP(NP+1) = UMIN
I
IF(UMIN.NE.0..OR.UMAX.NE.0.) YP(NP+2) = (UMAX - UMIN)/6.
UOUT(NOUT+1) = YP(NP+1)
UOUT(NOUT+2) = YP(NP+2)
IF(IPSLCT(12).NE.0) ICALL AXES(0.,0.,90.,0.,YP(NP+1),YP(NP+2),1.,0.,
1.4MT/TO,0.,2.,4.)
ICALL AXES(0.,0.,90.,0.,XP(NP+1),XP(NP+2),1.,0.,
1.4MT/TO,0.,2.,4.)
ICALL AXES(0.,0.,90.,0.,YP(NP+1),YP(NP+2),1.,0.,
1.4MT/TO,0.,2.,4.)
CALL LINPLT(X,YP,NP,1.,0.,0.,0.,0.)
DO 460 IJ = 1,NOUT
IF(UOUT(IJ) .GE. YP(NP+1)+6.*YP(NP+2)) GO TO 470
460 CONTINUE
NOUTY = NOUT
GO TO 480
470 NOUTY = IJ - 1
480 CONTINUE
DO 490 IJ = 1,NOUT
IF(UOUT(IJ) .GE. YP(NP+1)) GO TO 500
490 CONTINUE
IJ = 1
500 NOUTX = NOUT - IJ + 1
XOUT(NOUTY+1) = XOUT(NOUT+1)
XOUT(NOUTY+2) = XOUT(NOUT+2)
UOUT(NOUT+1) = YP(NP+1)
UOUT(NOUT+2) = YP(NP+2)
CALL LINPLT(XOUT(IJ),UOUT(IJ),NOUTX,1.,0.,0.,0.)
CALL NFRAME
CALL CALPLT(4.,5.,1.,5.,-3)
510 IF(IPSLCT(15).EQ.0.AND.IPSLCT(15).EQ.0) GO TO 530
C
C MASS FRACTION VS X
C
NPM = NP - 1
GO 520 IJ = 1,NP
YP(IJ) = YS(IJ)
520 CONTINUE
CALL ASCALE(YP,NP,NP,1.,1.)
IF(YMIN.NE.0..OR.YMAX.NE.0.) YP(NP+1) = YMIN
IF(YMIN.NE.0..OR.YMAX.NE.0.) YP(NP+2) = (YMAX - YMIN)/6.
CALL AXES(0.,0.,90.,0.,XP(NP+1),XP(NP+2),1.,0.,
1.4MT/TO,0.,2.,4.)
2 'DISTANCE JAPLOG (GZ L Er. Ml.)0.2,-32)
CALL AXE5(0.,0.,90.,0.,YP(NP+1),YP(NP+2),1.,0.,
1.'M ASS I.FRACTION J LIQUID J(A T.R.E, W)
1.0.2,-60)
CALL LINPLT(X,YP,NP,1.,0.,0.,0.)
IF(IPSLCT(15).NE.0) GO TO 530
530
CALL NFRAME
CALL CALPLOT(4,5,1,5,-3)
530 IF(IPSLCT(5).EQ.0.AND.IP5LCT(15).EQ.0) GO TO 580
540 CONTINUE
DO 540 IJ=1,NP
YP(IJ) = TS(IJ) - T(IJ)
550 CONTINUE
CALL ASCALE(YP,6,NP,1,10.)
560 CONTINUE
DO 560 IJ=L,NP
IF(YP(IJ) .LT. 1.0E-10.) YP(IJ) = 1.0E-10.
570 CONTINUE
CALL CALPLOT(-4,5,1,5,-3)
580 IF(IPSLCT(5).EQ.0,A.ND.IP5LCT(15).EQ.0) GO TO 610
590 CONTINUE
DO 590 IJ=1,NP
YP(IJ) = CAPP(IJ)
599 CONTINUE
CALL ASCALE(YP,6,NP,1,10.)
600 CONTINUE
DO 600 IJ=1,NP
IF(YP(IJ) .LT. 1.0E-10.) YP(IJ) = 1.0E-10.
609 CONTINUE
CALL CALPLOT(-4,5,1,5,-3)
610 IF(IPSLCT(5).EQ.0.AND.IP5LCT(15).EQ.0) GO TO 630
620 CONTINUE
DO 620 IJ=1,NP
YP(IJ) = SI(IJ) - S0
630 CONTINUE
CALL CALPLOT(-4,5,1,5,-3)
640 IF(S0 .LT. 0.0) S0 = 0.0
650 CONTINUE
DO 650 IJ=1,NP
IF(YP(IJ) .LT. S0) YP(IJ) = S0
659 CONTINUE
CALL CALPLOT(-4,5,1,5,-3)
CALL LINPLT(X,Y,NP,1,0,0,0)
CALL NFRAHE
CALL CALPLT(4,5,1,5,-3)
CALL NOTATE(0.,5.5,0.16,'TS(N(BRPSN(S(F(D18NS)-)TSDGSN'),0.,27)
CALL NOTATE(0.,4.5,0.16,'j',0.,1)
CALL NOTATE(0.,3.5,0.16,'(W)',0.,3)
CALL NOTATE(0.,2.5,0.16,'(SATURATION)',0.,12)
CALL NOTATE(0.,1.5,0.16,'(R)SDGSN',0.,8)
CALL LINPLT(PATX,PATY,2,1,0,0,0,0)
CALL NOTATE(1.7,0.5,0.16,'(NUCLEATION AND CONDENSATION)',0.,29)
PATY(1) = PATY(1) - 0.5
PATY(2) = PATY(2) - 0.5
CALL DASHPLT(PATX,PATY,2,1,PAT,2)
CALL NOTATE(1.7,0.5,0.16,'(ISENTROPIC WITHOUT CONDENSATION)',0.,33)
630 CONTINUE
518 RETURN
519 END
SUBROUTINE PRTOUT(I3,I1)

SUBROUTINE PRTOUT GENERATES PRINT OUTPUT OF THE
PROGRAM RESULTS

DIMENSION HLI3(400),SLI3(400),YIJ3(400)
COMMON/CMOUT3/ENT
COMMON/CAMUT/CAMBAR
COMMON/INFNT/TO,PO,SLAST,PHII,RHC,RHO
COMMON(I3,J3,A3(400),X3(400),DELM3(400),CAPJX3(400),R3(400),
1  NJ3(400),T3(400),F3(400),G3(400),Y3(400),
2 DELY3(400),DELS3(400),RPEM3(400),TS3(400),
3 OUTCMH3(400),CAPPJ3(400)
REAL MU3
COMMON/OUT1/50,HQ,MDO,T,RH0,STAR,TSTAR
REAL MDO
COMMON/CMOUT1/NENTRY,DT,RH0,T,PHII,U,RH0,RHOF
COMMON/ICOM/SIGMA(10),SIG,P
COMMON/OUT2/U2,RHO2,CAPJ2,T2,STAR2,H2,STAR,CAPL2
COMMON/ADJOUT/GAMMA1,OCALPHA,BETA,STAR,OT2
1 JMIN,DELT,EP1(11),1OUT,DT1,CYRAT,ARAT,JSTOP,JDB,CF,CP1
REAL JMIN
COMMON/SWITCH/SWEND,SWPO,SWGD
COMMON/CMOUT2/NE,T2,W,SIGMAX,U2,W,RHO,P,PINF,PBAR,TR,H2,W,RSTAR,CAPL,
PHI_J
REAL J
INTEGER SWEND,SWPO,SWGD
DATA N_/O/
DATA IFLG/I/

C FORMAT(1HI,ZCX,1_HPART I SUMMARY///
1 1HO,17X,WHSOG = ,1PE16.8/
1 21HO,17X,WHD = ,1PE16.8/
2 31HO,17X,WMMDO = ,1PE16.8/
3 41HO,17X,WHRH0* = ,1PE16.8/
4 51HO,17X,WHT = ,1PE16.8/
5 61HO,17X,WHT* = ,1PE16.8/
6 71HO,17X,WHSIGMA = ,1PE16.8/
7 8(1HO_2bX(1PE16.8))
8 91HO,WHERS(1PE16.8/)
9 2 FORMAT(I1HI,2X,2X,Z2HPART 1 EXTENDED OUTPUT///
10 111HO,4X,X,9H,5H__,+9X,3H,RH0,9X,5,3H,RH0 10X,8X
11 213H,RH0,11X,1HT,12X,11H,12X,11H,10X,5H#RHO,8X,6H,GAMMA1/)
12 3 FORMAT(3X,101PE13,5/)
13 4 FORMAT(I1HI,2X,2X,Z2HPART 2 SUMMARY///
14 11HO,5X,9HU = ,1PE16.8/
14 21HO,5X,9HRH0 = ,1PE16.8/
15 31HO,5X,9HT = ,1PE16.8/
16 41HO,5X,9HRH0* = ,1PE16.8/
17 51HO,5X,9HT* = ,1PE16.8/
18 61HO,5X,9HRSTARI = ,1PE16.8/
19 71HO,5X,9HGSIGMA = ,1PE16.8/
20 891HO,5X,WHERS(1PE16.8/)
21 5 FORMAT(I1HI,2X,2X,Z2HPART 2 EXTENDED OUTPUT///
22 6 FORMAT(11HO,4X,X,4H,7HT/TO,11X,1HH,12X,11H,11X,3H,RH0,9X,4H,P/PO,9X,5H,PINF,
23 8X,4,HPBAR,11X,1HHX,10X,6H,GAMMA1)
24 7 FORMAT(11HO,5X,3H TR,9X,6HSIGMAX,8X,5HRSTAR,10X,1HH,11X,3H,MHI,
25 21X,1HJ)
26 8 FORMAT(I1HI,2X,2X,Z2HPART 3 SUMMARY///
27 11HO,5X,9HU = ,1PE16.8/
28 21HO,5X,9HRH0 = ,1PE16.8/
29 31HO,5X,9HT = ,1PE16.8/
30 41HO,5X,9HRH0* = ,1PE16.8/
31 51HO,5X,9HT* = ,1PE16.8/
32 61HO,5X,9HRSTARI = ,1PE16.8/
33 71HO,5X,9HGSIGMA = ,1PE16.8/
34 891HO,5X,WHERS(1PE16.8/)
35 5 FORMAT(I1HI,2X,2X,Z2HPART 2 EXTENDED OUTPUT///
36 6 FORMAT(11HO,4X,X,4H,7HT/TO,11X,1HH,12X,11H,11X,3H,RH0,9X,4H,P/PO,9X,5H,PINF,
37 8X,4,HPBAR,11X,1HHX,10X,6H,GAMMA1)
38 7 FORMAT(11HO,4X,X,4H,7HT/TO,11X,1HH,12X,11H,11X,3H,RH0,9X,4H,P/PO,9X,5H,PINF,
39 8X,4,HPBAR,11X,1HHX,10X,6H,GAMMA1)
40 8 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
41 9 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
42 10 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
43 11 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
44 12 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
45 13 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
46 14 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
47 15 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
48 16 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
49 17 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
50 18 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
51 19 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
52 20 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
53 21 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
54 22 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
55 23 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
56 24 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
57 25 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
58 26 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
59 27 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
60 28 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
61 29 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
62 30 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
63 31 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
64 32 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
65 33 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
66 34 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
67 35 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
68 36 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
69 37 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
70 38 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
71 39 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
72 40 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
73 41 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
74 42 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
75 43 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
76 44 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
77 45 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
78 46 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
79 47 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
80 48 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
81 49 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///
82 50 FORMAT(I1HI,2X,2X,Z2HPART 3 EXTENDED OUTPUT///

IF(NENTRY.EQ.11) NLINE = 0
GO TO (120,40,70),IP
PART 1 OUTPUT

20 CONTINUE
IF(IL.NE.0) GO TO 30

STANDARD OUTPUT

WRITE(6,1) S0,HO,HODT,RHODRT,TSTAR,USTAR,SIGMA
GO TO 140

30 CONTINUE
IF(IL.NE.2) GO TO 140

DEBUG OUTPUT

IF(NENTRY.EQ.1,OR,NLINE.EQ.0) WRITE(6,2)
IF(NLINE.EQ.0) NLINE = 6
NLINE = NLINE + 1
WRITE(6,3) ASTAR,HO,RHOD,SO,RHOF1,T1,H1,U1,URHO,GAMBAR
GO TO 140

PART 2 OUTPUT

40 CONTINUE
IF(IL.NE.0) GO TO 50

STANDARD OUTPUT

WRITE(6,4) U2,RH02,CAIP2,T2,P2,RSTAR2,SIG2
GO TO 140

50 CONTINUE
IF(IL.NE.1) GO TO 60

EXTENDED OUTPUT - 1

IF(NENTRY.EQ.1,OR,NLINE.EQ.0) WRITE(6,5)
IF(NENTRY.LT.N2) IFLG = 2
WRITE(6,6) NLINE = NLINE + 6
IF(NLINE.EQ.0) NLINE = NLINE + 6
IF(NLINE.EQ.1) NLINE = NLINE + 1
IF(NLINE.GT.45) NLINE = NLINE + 1
WRITE(6,7) XTT,PP,P/PO
WRITE(6,8) XTT,TZ,PP,P/PO
N2 = NENTRY
GO TO 140

EXTENDED OUTPUT - 2

NLINE = NLINE + 1
IF(NENTRY.GT.45) NLINE = 0
WRITE(6,9) T3,ASTAR,CAPI,PPBAR,PHI,J
GO TO 140

PART 3 OUTPUT

70 CONTINUE
IF(IL.NE.0) GO TO 120

STANDARD OUTPUT

WRITE(6,10) NLINE = 6
J3M1 = J3 - 1
DO 80 I=1,J3M1
IF(NLINE.EQ.0) WRITE(6,8)
IF(NLINE.EQ.0) NLINE = NLINE + 6
NLINE = NLINE + 1
IF(NLINE.GT.45) NLINE = 0
PP = P3/PI
TT = T3/TO
WRITE(6,11) X3,PP,TT,CAIP3,CAIP3,DELM3,U3,Y3
80 CONTINUE
IF(IL.NE.100) GO TO 120
REWIND 7
REWIND 7

READ(7,END=100) JJJ,R3,MM3,H13,SL3,Y3
100 IF(E0F(7).NE.0) GO TO 140
SUBROUTINE SATTEM(TS,PX)
C
C SUBROUTINE SATTEM CALCULATES THE SATURATION TEMPERATURE
C
COMMON/SIGCOM/SIGMA(10), SIG, PSUP0
COMMON/ADJUST/GAMMA1, Q1, ALPHA, BETA, DSTAR, ASTAR, DT2,
1 JMIN, DELX, EPS(1), IOUT, DT1, CVRAT, ARAT, JSTOP, JDB, CP, CPI
REAL JMIN
C
C SOLVE EQUATION (57) FOR T
C
TOLD = 300.
ALOP = ALOG(PX*SIGMA(10)/SIG)
10 CONTINUE
F = 55.897 - 6641.7/TOLD - 4.4864*ALOG(TOLD) - ALOP
FP = 6641.7/TOLD*2 - 4.4864/TOLD
THEN = TOLD - F/FP
IF(ABS(THEN-TOLD)/TOLD.LT.EPS(1)) GO TO 20
20 THEN = TOLD
GO TO 10
C
20 CONTINUE
TS = THEN
RETURN
END
SUBROUTINE XVSA (ALST, XLST)

SUBROUTINE XVSA calculates the x position in the nozzle as a function of area. ACOEF contains the spline coefficients determined by the library routine CSDS.

COMMON/ARECM/NA, XA, ACOEF

DIMENSION ACOEF(10, 4)

DIMENSION A(I(10))

COMMON/CONST/TCS, CAPR, M1, W(10), RHOL, RBA, PI, ALPHAC

REAL K, M

COMMON/ADJUST/GAMMA1, O, ALPHAB, BETA, DSTAR, ASAP, DT2,

1 JMIN, DELX, EPS(8), IOUT, DT1, CVRT, ARAT, JSTOP, JDB, CP, CP1

REAL JMIN

DIMENSION XA(I(10))

DO I0 I = 1, NA

A(I) = ACOEF(I, 1)

A(I) = PI*A(I)**2

10 CONTINUE

AREA = ALST

RBA = SORT(AREA/PI)

NPD = NA - 1

DO 20 IJ = 1, NPD

IF(AREA.GE.A(IJ) .AND. AREA.LE.A(IJ + 1)) GO TO 30

20 CONTINUE

IJ = NPD

30 CONTINUE

0.5*XA(IJ + 1) - XA(IJ)

40 CONTINUE

FXX = ((ACOEF(IJ, 4) + ACOEF(IJ, 3))*HOLD + ACOEF(IJ, 2))*HOLD + ACOEF(IJ, 1)

44 CONTINUE

FXX = FXX - RBA

FXXP = (3. * ACOEF(IJ, 4) + 2. * ACOEF(IJ, 3))*HOLD + ACOEF(IJ, 2)

48 CONTINUE

HNEW = HOLD - FXX/FXXP

IFABS(HNEW + HOLD)/HOLD .LT. EPS(8)) GO TO 50

40 CONTINUE

GO TO 40

50 CONTINUE

XLST = XA(IJ) + HNEW

IF(AREA.LT.A(I)) XLST = (AREA/A(I))*XA(I)

RETURN

END

ORIGINAL PAGE IS OF POOR QUALITY
References


   *Part I—Nucleation and Droplet Growth Theory*. CUED/A-Turbo/TR 97, Pt. I.
   *Part II—Numerical Methods and Comparison With Experimental Results*. CUED/A-Turbo/TR 97, Pt. II.


Table 1. Conditions for Numerical Results
[For all cases: \( q_e = 1, \alpha = 8, \alpha_c = 1, \beta = 2, \) and \( J_{\text{min}} = 10^{15} \)]

<table>
<thead>
<tr>
<th>Case</th>
<th>( T_0, \text{K} )</th>
<th>( p_0, \text{bars} )</th>
<th>( \phi )</th>
<th>( y_{H_2O} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1900</td>
<td>50</td>
<td>0.798</td>
<td>0.154</td>
</tr>
<tr>
<td>2</td>
<td>1900</td>
<td>250</td>
<td>.797</td>
<td>.154</td>
</tr>
<tr>
<td>3</td>
<td>1600</td>
<td>50</td>
<td>.620</td>
<td>.122</td>
</tr>
<tr>
<td>4</td>
<td>1600</td>
<td>250</td>
<td>.620</td>
<td>.122</td>
</tr>
<tr>
<td>5</td>
<td>1900</td>
<td>250</td>
<td>.427</td>
<td>.156</td>
</tr>
</tbody>
</table>

Table 2. Nozzle Coordinates for Langley 8’HTT and Computed Boundary-Layer-Displacement Thickness

<table>
<thead>
<tr>
<th>( x, \text{m} )</th>
<th>( r_w, \text{m} )</th>
<th>( \delta^*, \text{m} ), for—</th>
<th>Case 1</th>
<th>Case 2 and 5</th>
<th>Case 3</th>
<th>Case 4</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>.0710</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>.0860</td>
<td>.0714</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2.5400</td>
<td>.3538</td>
<td>.0072</td>
<td>.0059</td>
<td>.0073</td>
<td>.0060</td>
<td></td>
</tr>
<tr>
<td>5.4703</td>
<td>.6911</td>
<td>.0271</td>
<td>.0213</td>
<td>.0275</td>
<td>.0217</td>
<td></td>
</tr>
<tr>
<td>6.4230</td>
<td>.7938</td>
<td>.0358</td>
<td>.0281</td>
<td>.0363</td>
<td>.0286</td>
<td></td>
</tr>
<tr>
<td>9.8560</td>
<td>1.0434</td>
<td>.0670</td>
<td>.0523</td>
<td>.0675</td>
<td>.0528</td>
<td></td>
</tr>
<tr>
<td>12.8949</td>
<td>1.1660</td>
<td>.0911</td>
<td>.0713</td>
<td>.0916</td>
<td>.0718</td>
<td></td>
</tr>
<tr>
<td>15.7116</td>
<td>1.2192</td>
<td>.1089</td>
<td>.0851</td>
<td>.1093</td>
<td>.0855</td>
<td></td>
</tr>
</tbody>
</table>
Figure 1. Schematic of combustion-heated wind tunnel.
Figure 2. Numerical results for case 1 with $T_o = 1900$ K, $p_o = 50$ bars, and $\phi = 0.798$. 

(a) Temperature difference, nucleation rate, and mass fraction liquid water.

(b) Entropy production.

(c) Static pressure distribution.

(d) Droplet growth.
Figure 3. Numerical results for case 2 with $T_o = 1900 \, \text{K}$, $p_o = 250 \, \text{bars}$, and $\phi = 0.797$. 

(a) Temperature difference, nucleation rate, and mass fraction liquid water.

(b) Entropy production.

(c) Static pressure distribution.

(d) Droplet growth.
Figure 4. Numerical results for case 3 with $T_0 = 1600$ K, $p_0 = 50$ bars, and $\phi = 0.620$. 

(a) Temperature difference, nucleation rate, and mass fraction liquid water.

(b) Entropy production.

(c) Static pressure distribution.

(d) Droplet growth.
Figure 5. Numerical results for case 4 with $T_o = 1600$ K, $p_o = 250$ bars, and $\phi = 0.620$. 

(a) Temperature difference, nucleation rate, and mass fraction liquid water. 

(b) Entropy production. 

(c) Static pressure distribution. 

(d) Droplet growth.
Figure 6. Numerical results for case 5 with $T_0 = 1900$ K, $p_0 = 250$ bars, and $\phi = 0.427$. 

(a) Temperature difference, nucleation rate, and mass fraction liquid water.

(b) Entropy production.

(c) Static pressure distribution.

(d) Droplet growth.
**Finite-Rate Water Condensation in Combustion-Heated Wind Tunnels**

Wayne D. Erickson, Gerald H. Mall, and Ramadas K. Prabhu

**Report Date**
September 1988

**Performing Organization Name and Address**
NASA Langley Research Center
Hampton, VA 23665-5225

**Sponsoring Agency Name and Address**
National Aeronautics and Space Administration
Washington, DC 20546-0001

**Abstract**
A quasi-one-dimensional method for computing finite-rate nucleation and droplet growth of water in a supersonic expansion of combustion products is presented. Sample computations are included for the Langley 8-Foot High-Temperature Tunnel, but the method can also be applied to other combustion-heated wind tunnels. The sample results indicate that the free-stream static pressure can be in the range of 25 to 60 percent greater than that computed for isentropic nozzle flow without water condensation. The method provides a tool for examining the effects of water condensation on static state properties and velocity of the supersonic stream in combustion-heated wind tunnels.

**Key Words**
- Combustion-heated wind tunnels
- Finite-rate water condensation
- Nucleation
- Droplet growth

**Distribution Statement**
Unclassified—Unlimited

**Subject Category**
09

**Security Classification**
Unclassified