CONTRACTOR REPORT

SAND96–2878
Unlimited Release
UC–702

PROGRAM DROP: A Computer Program for Prediction of Evaporation from Freely Falling Multicomponent Drops

Patrick M. Gavin
Gavin Consulting
1599 Krebs Court
Newark, Ohio 43055

Prepared by
Sandia National Laboratories
Albuquerque, New Mexico 87185 and Livermore, California 94550
for the United States Department of Energy
under Contract DE-AC04-94AL85000

Printed December 1996

Issued by Sandia National Laboratories, operated for the United States Department of Energy by Sandia Corporation.

Available to DOE and DOE contractors from
Office of Scientific and Technical Information
PO Box 62
Oak Ridge, TN 37831

Prices available from (615) 576-8401, FTS 626-8401

Available to the public from
National Technical Information Service
US Department of Commerce
5285 Port Royal RD
Springfield, VA 22161

NTIS price codes
Printed copy: A03
Microfiche copy: A01

SAND96-2878
Unlimited Release
Printed December 1996

Distribution
Category UC-702

PROGRAM DROP: A Computer Program for Prediction of Evaporation from Freely Falling Multicomponent Drops

Patrick M. Gavin
Gavin Consulting
1599 Krebs Court
Newark, Ohio 43055

Sandia Contract No. AJ-3260

Abstract

PROGRAM DROP consists of a series of FORTRAN routine which together are used to model the evaporation of a freely falling, multicomponent drop composed of an arbitrary number of volatile species and a single nonvolatile, inert component. The physics underlying the model are clearly identified, and the model's relationship to previous work in the literature is described. Test cases are used to illustrate the viability of the model and to highlight its potential usefulness in the accurate prediction of multicomponent droplet vaporization in a variety of applications.

Acknowledgment

The author wishes to acknowledge the guidance of his contract manager, Bruce Boughton, in the completion of the current work. Discussions with Ronald O. Pennsyle of the Advanced Systems Concepts Directorate at Aberdeen Proving Ground, Maryland, were also of value in setting the direction for the current modeling effort.

List of Figures

Table of Contents

List of Figures

Nomenclature

English Symbols

$a_{11}, a_{12}, a_{13}, a_{14}$	constants for determination of droplet Reynolds number at small X [Equation (22)]
$a_{20}, a_{21}, a_{22}, a_{23}$	constants for determination of droplet Reynolds number at large X [Equation (23)]
a_{Di}, b_{Di}	constants for determination of diffusion coefficient of component i [Equation (25)]
a_{Li}, b_{Li}	constants for determination of latent heat of vaporization of component i [Equation (5)]
a_{Ri}, b_{Ri}	constants for determination of density of component i [Equation (24)]
A_S	droplet surface area
c_{pa}	specific heat of air
C	parameter for assessment of importance of radial convection [Equation (1)]
C_D	droplet drag coefficient
d	droplet diameter [Equation (11)]
D_i	diffusion coefficient of component i [Equation (25)]
f	droplet thermal equilibrium function [Equation (18)]
g	acceleration of gravity
h_{Di}	mass transfer coefficient for component i
h_k	heat transfer coefficient
i	volatile component identification number
k	thermal conductivity of air [Equation (30)]
m	droplet total mass

English Symbols (continued)

m_i	mass of component i in droplet
m_I	mass of inert component in droplet
M_i	molecular weight of component i
n	number of volatile components in droplet
Nu	droplet Nusselt number [Equations (15) & (16)]
P_{Ai}	vapor pressure of component i in ambient air
P_{Bi}	base vapor pressure for computation of saturation vapor pressure of component i [Equation (6)]
P_{fi}	film vapor pressure for component i [Equation (28)]
P_{Si}	saturation vapor pressure for component i [Equation (6)]
P_T	ambient total pressure
Pr	droplet Prandtl number [Equation (17)]
R_a	ideal gas constant for air
R_i	ideal gas constant for component i
Re	droplet Reynolds number [Equation (9)]
Sc_i	Schmidt number for component i [Equation (10)]
Sh_i	Sherwood number for component i [Equations (7) & (8)]
t	time
T	droplet temperature
T_A	ambient temperature
T_{Bi}	base temperature for computation of saturation vapor pressure of component i [Equation (6)]

English Symbols (continued)

T_f	film temperature [Equation (26)]
T_{new}	corrected droplet temperature in Newton-Raphson iteration [Equation (19)]
u	droplet settling velocity
$\forall$	droplet volume [Equation (12)]
x_i	mole fraction of component i in droplet [Equation (3)]
X	dimensionless variable for calculation of droplet Reynolds number [Equation (21)]

Greek Symbols

λ_i	latent heat of vaporization of component i [Equation (5)]
μ	dynamic viscosity of air [Equation (29)]
ρ_g	density of ambient air [Equation (27)]
ρ_i	density of component i in liquid phase [Equation (24)]
ρ_I	density of inert component

1. Introduction

This report documents work performed under Contract No. AJ-3260, the main focus of which was the development of a FORTRAN computer code to model the evaporation of a freely falling, multicomponent liquid drop containing a single nonvolatile, inert component.

Because of the wide variety of applications in which droplet evaporation plays a significant role, there exists a large body of literature on droplet evaporation and related problems. A complete review of this literature is beyond the scope of the present project, and the reader is referred to the work by Gavin (1983) for a comprehensive critical review of the field. The evaporation model developed under the current contract is based largely on the subset of literature devoted to evaporation from drops composed of multiple miscible, volatile components.

Section 2 describes the theory behind the current model, and specific literature references on which the code is based are cited therein. Section 3 provides an overview of the computer code and includes the results of several runs made to benchmark the code against theoretical and experimental data available in the literature.

2. THEORETICAL BACKGROUND

2.1 Mass Transfer

Two groups of multicomponent droplet mass transfer models can be distinguished in the literature: the so-called "well-mixed" models [e.g., Newbold and Amundson (1973); Law (1976)], which assume uniform concentrations of all species within the drop; and the "unmixed" models [e.g., Landis and Mills (1974); Law (1978)], which assume that internal species concentrations are governed by diffusion.

The current model falls in the "well-mixed" category. This approach was selected for the following reasons:

- Freely falling drops are generally subject to internal circulation induced by air drag on the droplet surface; this tends to "stir" the drop and prevent significant internal concentration gradients from forming.

- The few experimental data which are available in the literature [e.g., El Wakil et al. (1956); Wood et al. (1960)] tend to support the well-mixed assumption.

- The liquid-phase diffusion coefficient data needed to model species diffusion within the drop are often unavailable, as is the case for chemical agents [Pennsyle (1994)], so that the additional complications resulting from use of an unmixed model are of little value in simulating actual applications.

The current model further neglects the effects of radial convection, also known as Stefan flow, on mass transfer from the droplet. This simplification was made because the present model is targeted primarily at droplet evaporation in the open atmosphere, where relatively low temperatures and evaporation rates will be encountered. Stefan flow was examined in some detail by Newbold and Amundson (1973), who observed that inclusion of radial convection effects is important for accurate modeling of droplet evaporation in high-temperature environments, when the droplet temperature may approach the boiling point of one or more of its liquid constituents. They defined a dimensionless parameter C which can be used to determine whether or not radial convection is significant in a particular problem:

$$C = \frac{P_T - \sum_{i=1}^{n} P_{Ai}}{P_T - \sum_{i=1}^{n} x_i P_{Si}}, \qquad (1)$$

where symbols are as defined in the Nomenclature section at the beginning of this report. When the value of C is near unity, radial convection phenomena are unimportant; and the linearized

treatment of mass transfer used by the current model is adequate. Computation of C for the benchmark cases examined later in this report (see Section 3) certainly supports the validity of the linearized mass transfer analysis.

With these assumptions in hand, the evaporation rate of a given species from a multicomponent drop can be expressed as follows:

$$\frac{dm_i}{dt} = -h_{Di} A_s \frac{x_i P_{si} - P_{Ai}}{R_i T_f} . \qquad (2)$$

In Equation (2), Raoult's law for ideal solutions has been used to determine the vapor pressure of the volatile component at the surface of the drop. That is, the surface vapor pressure of a component is determined by multiplying its saturation vapor pressure by its mole fraction in the liquid phase, defined as

$$x_i = \frac{m_i / M_i}{\sum_{i=1}^{n} m_i / M_i} . \qquad (3)$$

The use of Raoult's law is common in the multicomponent droplet literature, presumably because there are usually no property data available to justify the use of more complicated non-ideal solution analyses in the prediction of surface vapor pressure. This appears to be the case for chemical agent applications as well [Pennsyle (1994)].

The present model uses an integrated form of the Clausius-Clapeyron equation to determine the saturation vapor pressure of the volatile components. In its differential form, the Clausius-Clapeyron equation,

$$\frac{dP_{si}}{dT} = \frac{\lambda_i P_{si}}{R_i T^2} , \qquad (4)$$

is an exact thermodynamic relationship between species vapor pressure and temperature, involving only two assumptions [Hsieh (1975)]: 1) vapor density is much smaller than liquid density; and 2) the vapor obeys the ideal gas law. Because the latent heat of vaporization varies slowly with temperature, it is generally adequate to assume a linear relationship between the two, as follows:

$$\lambda_i = a_{Li} + b_{Li} T. \qquad (5)$$

Over narrow temperature ranges, or in the absence of detailed property data, it can also be acceptable to assume constant latent heat of vaporization, which of course can be handled via Equation (5) by setting b_{Li} equal to zero. Inserting Equation (5) into Equation (4) and integrating, we obtain

$$P_{Si}(T) = P_{Bi} \left(\frac{T}{T_{Bi}}\right)^{b_{Li}/R_i} \exp\left[\frac{a_{Li}}{R_i}\left(\frac{1}{T_{Bi}} - \frac{1}{T}\right)\right]. \qquad (6)$$

In using Equation (6), the base values of species vapor pressure and temperature (P_{Bi} and T_{Bi}) can be chosen as any consistent set of values. Normally, however, one is constrained by available physical property data to use species normal boiling conditions as the base point, i.e., P_{Bi} equal to one atmosphere and T_{Bi} equal to the boiling temperature at one atmosphere. For water, it can be shown that Equation (6) gives vapor pressure values accurate to within about 2.5% between 0 and 100 C, when normal boiling conditions are used for base pressure and temperature (one atmosphere and 100 C, respectively). If the latent heat of vaporization is held constant at its value at 50 C, similar calculations for water show that Equation (6) is accurate to within 5.5% between 0 and 100 C.

The mass transfer coefficient in Equation (2) is determined from the Ranz-Marshall correlation [Ranz and Marshall (1952)], which was found by Gavin (1983) to yield very accurate predictions of evaporation rates of freely falling drops:

$$Sh_i = 2 + 0.6 Re^{1/2} Sc_i^{1/3}, \qquad (7)$$

where the Sherwood number is defined as

$$Sh_i = h_{Di} d / D_i, \qquad (8)$$

the Reynolds number is defined as

$$Re = \rho_g u d / \mu, \qquad (9)$$

and the Schmidt number is defined as

$$Sc_i = \mu / \rho_g D_i . \qquad (10)$$

The droplet diameter needed for computation of surface area and mass transfer coefficient is determined from droplet volume per

$$d = (6\forall/\pi)^{1/3}, \qquad (11)$$

where the droplet volume is calculated as the sum of the volumes of all the individual species present in the drop:

$$\forall = m_I/\rho_I + \sum_{i=1}^{n} m_i/\rho_i ; \qquad (12)$$

note that the inert component (if present) affects evaporation through its influence on droplet size, but does not otherwise participate in the transfer processes.

Using the above relationships, Equation (2) can now be written as

$$\frac{dm_i}{dt} = -\pi d \frac{D_i Sh_i}{R_i T_f} (x_i P_{si} - P_{Ai}) , \qquad (13)$$

which is the expression used in the computer program to determine species evaporation rates.

2.2 Heat Transfer

For purposes of heat transfer calculations, the drop temperature is assumed to be spatially uniform, i.e., "well-mixed," as described above in connection with mass transfer. Gavin (1983) showed rigorously that for small drops (100-1000 microns diameter), even in the absence of internal circulation, the external resistance to heat transfer far exceeds the internal resistance, making the uniform temperature assumption entirely appropriate.

The current model further assumes that droplet heat transfer is quasi-steady, i.e., that the drop is in thermal equilibrium at any given instant in time. This is a common assumption in the multicomponent droplet literature; and Gavin (1983) showed that the unsteady period of droplet

heat transfer, wherein the droplet is adjusting from its initial temperature to its quasi-steady temperature, is a negligibly short fraction of the total evaporation time for small drops (100-1000 microns diameter). For chemical agent applications, it is also likely that droplet initial conditions will be difficult to specify, making the use of the quasi-steady assumption even more attractive.

With these assumptions, droplet thermal equilibrium can be expressed as a balance between the sensible heat convected to the drop from the surrounding air and the latent heat released from the drop by evaporation of the volatile components:

$$h_k A_S (T_A - T) = -\sum_{i=1}^{n} \lambda_i (dm_i/dt) . \qquad (14)$$

The heat transfer coefficient is calculated from the Ranz-Marshall correlation [Ranz and Marshall (1952)]:

$$Nu = 2 + 0.6 Re^{1/2} Pr^{1/3}, \qquad (15)$$

where the Nusselt number is defined as

$$Nu = h_k d/k, \qquad (16)$$

and the Prandtl number is defined as

$$Pr = \mu c_{pa}/k. \qquad (17)$$

Using the above relationships, and combining Equation (14) with Equation (13), the expression for droplet thermal equilibrium becomes

$$f(T) = \sum_{i=1}^{n} \lambda_i \frac{D_i Sh_i}{R_i T_f} (x_i P_{Si} - P_{Ai}) - kNu(T_A - T) = 0 . \qquad (18)$$

Note that Equation (18) is implicit in droplet temperature and must be solved iteratively. The current computer code solves Equation (18) using the Newton-Raphson method, which begins with an initial guess for droplet temperature and computes corrected approximations from

$$T_{new}=T-f(T)/f'(T), \tag{19}$$

where the derivative of Equation (18) with respect to temperature is approximated by

$$f'(T)=\sum_{i=1}^{n}[\lambda_i\frac{D_iSh_i}{R_iT_f}(\frac{\lambda_i}{R_i}\frac{x_iP_{Si}}{T^2}-\frac{x_iP_{Si}-P_{Ai}}{2T_f})]+kNu. \tag{20}$$

This solution technique has been found to work quite well, converging quickly to a final value for droplet temperature within a few iterations.

2.3 Dynamics

The present model assumes that the droplet is at all times falling at the terminal, or settling, velocity consistent with its current size and mass. Some of the most accurate and widely used data for the settling velocity of water drops are those of Gunn and Kinzer (1949) and Beard and Pruppacher (1969). The present model uses polynomial fits to the data of these researchers created by Berry and Pranger (1974) in order to determine the droplet fall velocity.

In order to avoid having to perform an iterative solution of a drag coefficient versus Reynolds number relationship, both sides of which would contain the droplet velocity, Berry and Pranger (1974) define the dimensionless variable X,

$$X=C_DRe^2=\frac{8\rho_g g(m-\rho_g\forall)}{\pi\mu^2}, \tag{21}$$

which contains only the droplet mass and volume, and relate it explicitly to the droplet Reynolds number through the following polynomials:

$$\begin{aligned} 0<X<175.270:\quad & Re=a_{11}X+a_{12}X^2+a_{13}X^3+a_{14}X^4 \\ & a_{11}=+0.412657\times10^{-1} \\ & a_{12}=-0.150074\times10^{-3} \\ & a_{13}=+0.758804\times10^{-6} \\ & a_{14}=-0.168841\times10^{-8} \end{aligned} \tag{22}$$

and

$$175.270 \le X \le 10^7: \quad \ln Re = a_{20} + a_{21}\ln X + a_{22}(\ln X)^2 + a_{23}(\ln X)^3$$

$$a_{20} = -0.236534 \times 10^{+1}$$
$$a_{21} = +0.767787 \times 10^{+0} \qquad (23)$$
$$a_{22} = +0.535826 \times 10^{-2}$$
$$a_{23} = -0.763554 \times 10^{-3}$$

Equations (22) and (23), although derived from experimental data on pure water drops, are expected to give reasonable predictions for small droplets (or solid spheres) of other compositions, as small drops should not deviate significantly from spherical shape.

The sequence of operations followed by the present computer program is to determine X from Equation (21) and then compute Re from Equations (22) and (23); the droplet settling velocity u is then found by simple rearrangement of Equation (9).

2.4 Physical Properties

The key physical properties which must be specified for each of the volatile components in the droplet are the vapor pressure, diffusion coefficient, latent heat of vaporization, and density. For vapor pressure, the present model requires a pressure-temperature base point for the integrated Clausius-Clapeyron equation [Equation (6)], as described in Section 2.1. Two parameters are also required to describe the relationship between latent heat of vaporization and temperature, per Equation (5). Because liquid densities are generally only weakly dependent on temperature, the linear relationship between species density and temperature assumed by the present model,

$$\rho_i = a_{Ri} + b_{Ri} T, \qquad (24)$$

should be more than adequate. Similar to the situation for latent heat of vaporization, assuming a constant value for density (b_{Ri} set to zero) can be entirely acceptable over narrow temperature ranges. For example, the density of water changes by less than 4.5% between 0 and 100 C.

Following Dunn (1977) and Law (1976), the species diffusion coefficient and absolute temperature are assumed to be related via a power-law expression:

$$D_i = a_{Di} T_f^{b_{Di}}. \qquad (25)$$

Note that vapor pressure, latent heat, and density are evaluated at the drop temperature, whereas the diffusion coefficient is evaluated at the film temperature,

$$T_f=(T+T_A)/2. \tag{26}$$

The only physical property required by the model for the nonvolatile, inert component in the drop is its density, which is assumed to be constant.

The model assumes the drop is falling through air. The ambient air density is calculated from the ideal gas law, corrected for the presence of droplet volatile species in the gas phase near the drop:

$$\rho_g=\frac{1}{T_f}\left[\frac{P_T-\sum_{i=1}^{n}P_{fi}}{R_a}+\sum_{i=1}^{n}\frac{P_{fi}}{R_i}\right], \tag{27}$$

where the film vapor pressure for each volatile component is calculated from

$$P_{fi}=(x_iP_{Si}+P_{Ai})/2. \tag{28}$$

The dynamic viscosity of air is calculated from the formula supplied by Beard and Pruppacher (1971),

$$\mu\ (kg/ms)\ =1.4963\times10^{-6}T_f^{1.5}/(T_f+120), \tag{29}$$

for film temperature in degrees Kelvin. The thermal conductivity of air is determined from data supplied by Ranz (1950),

$$k\ (W/mK)\ =2.367\times10^{-2}+6.865\times10^{-5}T_f, \tag{30}$$

for film temperature in degrees Celsius. No correction for presence of volatile droplet components in the gas phase is applied to either air viscosity or conductivity. The specific heat of air is assumed constant at a value of 1006 J/kg-K, as it is very insensitive to temperature over the range of temperatures likely to be encountered in the atmosphere.

3. Computer Program

A complete listing of the computer program written to analyze multicomponent droplet vaporization, titled PROGRAM DROP, is included in Appendix A. An overall description of the program is given in the following section; additional details can be obtained from the program listing, which is fully annotated with COMMENT statements.

3.1 Description of Code

PROGRAM DROP consists of six separate routines:

- The main program, which handles input and output, orchestrates integration of the rate equations, and terminates the run when the drop evaporates completely or hits the ground.

- FUNCTION TEMP, which determines the quasi-steady droplet temperature, via the method outlined in Section 2.2, given the droplet composition and position.

- FUNCTION REYN, which calculates the droplet Reynolds number from the dimensionless term X, as described in Section 2.3.

- SUBROUTINE PROPS, which calculates the physical properties of the drop and surrounding air, given the current drop temperature and position, per the techniques described in Section 2.4.

- SUBROUTINE AMB, which returns the local atmospheric temperature and total pressure, as well as ambient vapor pressures, if any, of volatile species in the drop, given the droplet vertical elevation. In the version of the code provided, the ambient temperature and total pressure have been set at uniform values of 20 C and 101325 Pa (one atmosphere), respectively, and all ambient volatile species concentrations have been set to zero. If the user wishes to change these defaults, the new values must be specified in SUBROUTINE AMB and the subroutine recompiled.

- SUBROUTINE DDROP, which calculates the first derivatives with respect to time of all variables for which the program computes time histories, i.e., the masses of all volatile species in the drop and the drop vertical elevation.

In addition to the above routines, the program uses the packaged integrator SUBROUTINE ODE to perform the numerical integration of the rate equations. For completeness, a listing of SUBROUTINE ODE is included in Appendix B. Note that ODE requires setting two machine-dependent constants (TWOU and FOURU) related to the numerical accuracy of the computer on which the calculations are done. These have been set to characterize the computer on which PROGRAM DROP was developed and may need to be changed, per the instructions given in the listing in Appendix B, when the code is run on different machines.

For typical cases run to date, PROGRAM DROP has required just a few seconds execution time on an IBM-compatible personal computer with a 386-25 chip and math co-processor.

3.2 Benchmark Cases

In all of the benchmark cases described below, the ambient air temperature and total pressure were assumed to be invariant with vertical position; and the values used to generate the cases are specified on the accompanying graphs. In all cases, the ambient volatile species concentrations were assumed to be zero (i.e., "dry" air).

The experimental data of Gavin (1983) were used to check the performance of PROGRAM DROP for single-component, pure water drops. These data were obtained by suspending the drops at terminal velocity on a vertical air stream in a wind tunnel, and as such they represent the evaporation behavior of freely falling drops. The data take the form of histories of the vertical wind tunnel velocity required to keep the evaporating drops stationary, and thus represent time histories of the droplet settling velocities.

The input and output files from PROGRAM DROP for the two cases examined, representing drops of two different initial sizes, are included in Appendix C. Graphs comparing the predictions of the code with the experimental data are shown in Figures 1 and 2. As can be seen from the graphs, the agreement is excellent. For completeness, the predicted time histories of droplet mass, diameter, and temperature for the larger drop are also included in Figures 3-5, respectively. Note that the droplet temperature is essentially constant throughout its lifetime, at a value roughly equal to the wet bulb temperature.

To check the performance of PROGRAM DROP in the case of multicomponent evaporation, its predictions were compared to the predictions of the model by Law (1976) for the case of a two-component drop consisting of hexane and octane vaporizing in 300 K (26.85 C) air. Law's predicted time histories of normalized drop surface area; change in drop temperature from initial quasi-steady value; drop composition; and component fractional vaporization are shown in Figures 6-9, respectively. Law studied two cases of interest to the present effort: Case I, in which the initial mole fractions are 75% hexane and 25% octane; and Case IV, in which the initial mole fractions are 25% hexane and 75% octane. The remaining two cases illustrated in Figures 6-9 apply to vaporization and combustion in high-temperature air, and as such are not treated here. Also, the references to the "Shell Model" in Figures 6 and 8 are to the predictions of a simplified model developed by Law (1976) in which the volatile species are assumed to be arranged in shells, in order of volatility, with the most volatile component in the outermost shell. He found that gross droplet characteristics (e.g., overall vaporization time) could be fairly well predicted with this simplified model. The present computer model predictions should be compared to those of Law's complete model (solid lines in Figures 6 and 8), not to those of the simplified Shell Model (dashed lines in Figures 6 and 8).

It must also be noted that Law's model applies to a stationary drop (no convection) and was formulated nondimensionally. As such, he does not specify an absolute initial drop size for the cases studied; and the results are plotted for nondimensional time varying from zero (initiation of vaporization) to one (drop completely vaporized). For comparison purposes, the initial drop size for the runs of PROGRAM DROP was arbitrarily selected as 500 microns, and the program was run until complete vaporization was reached. After importing the output from PROGRAM DROP into a LOTUS spreadsheet, the results were then plotted in normalized fashion, following Law (1976), so as to facilitate direct comparison to his curves. Physical property data for the runs were taken directly from Law (1976); because Law assumed unity Lewis number for both

Figure 1: Water Drop Settling Velocity

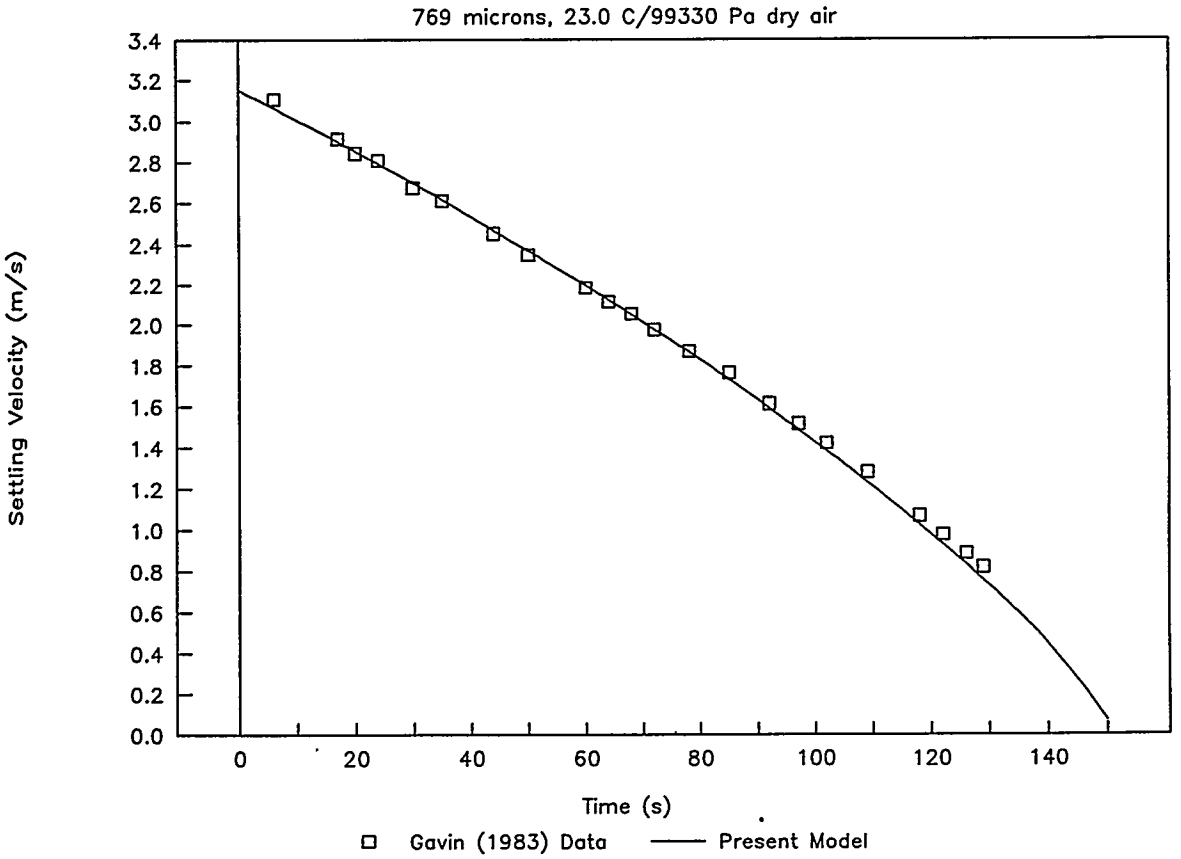


Figure 2: Water Drop Settling Velocity

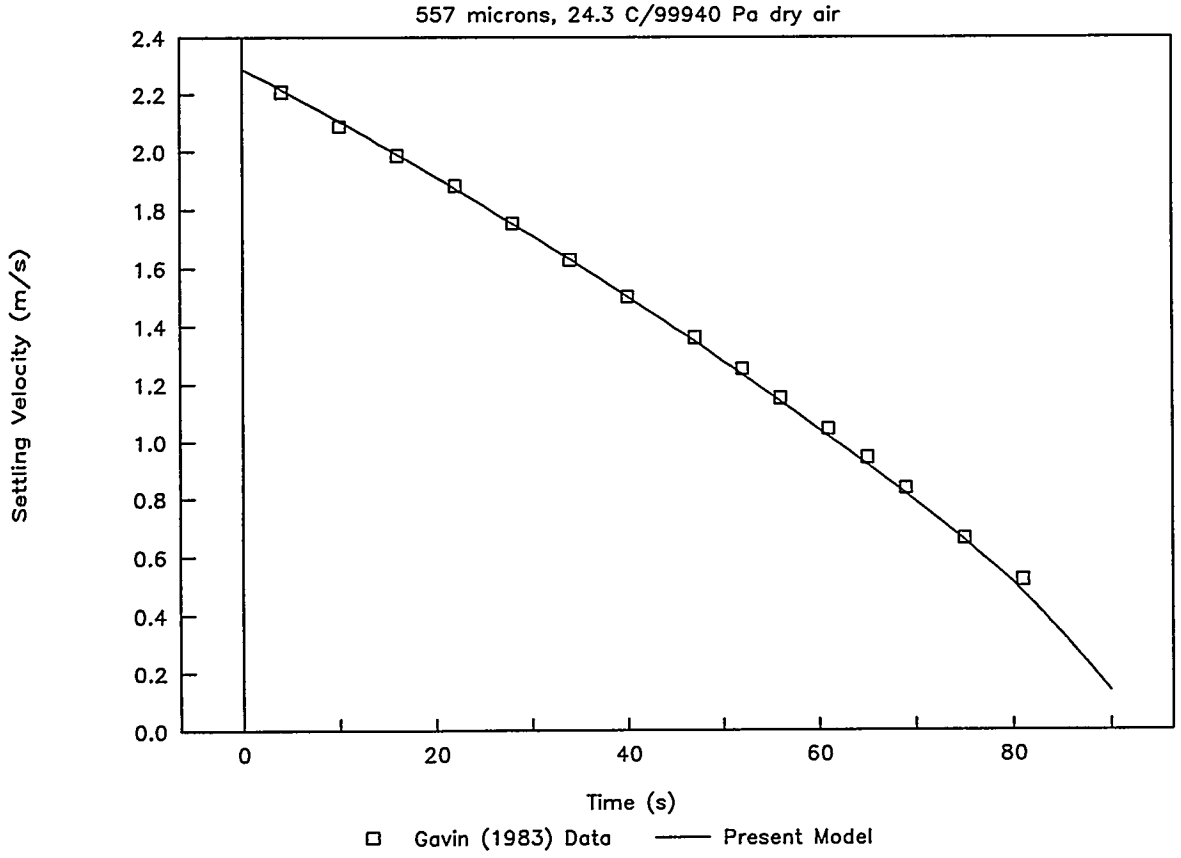


Figure 3: Water Drop Mass History

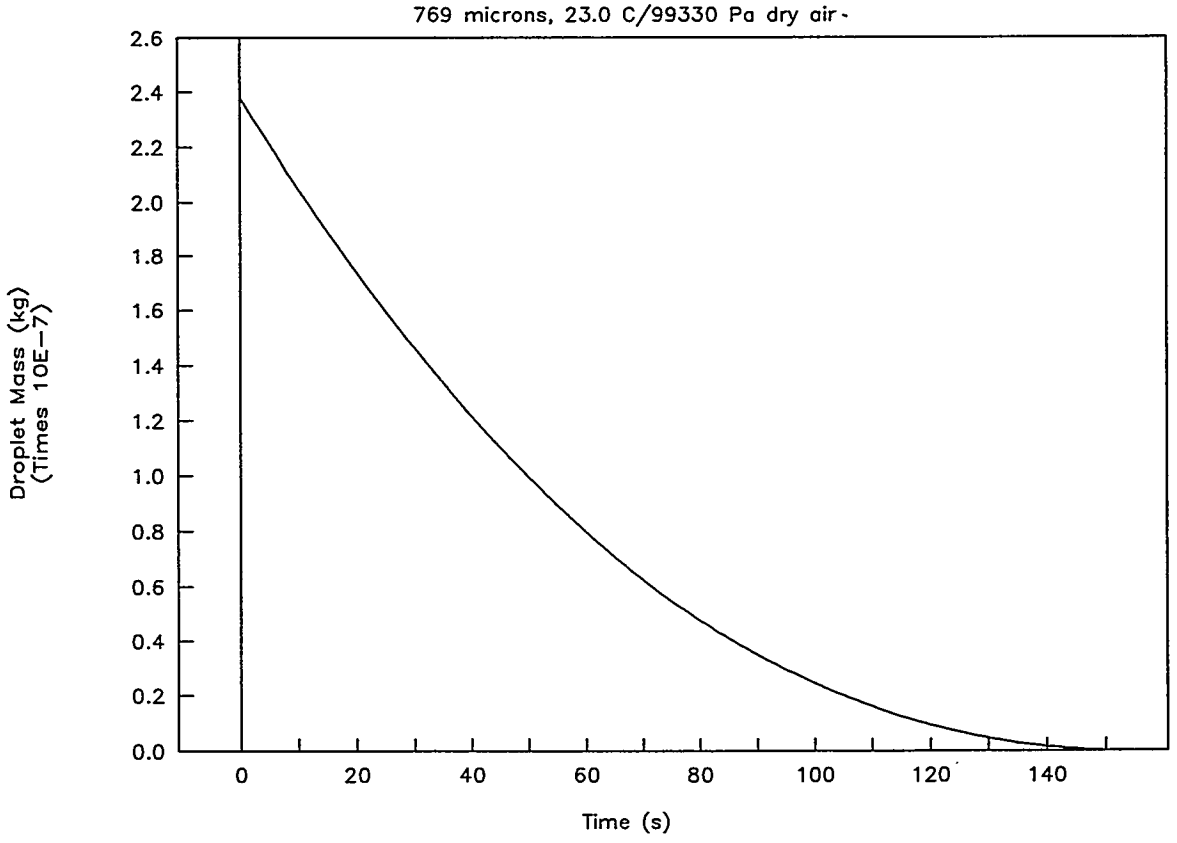


Figure 4: Water Drop Diameter History

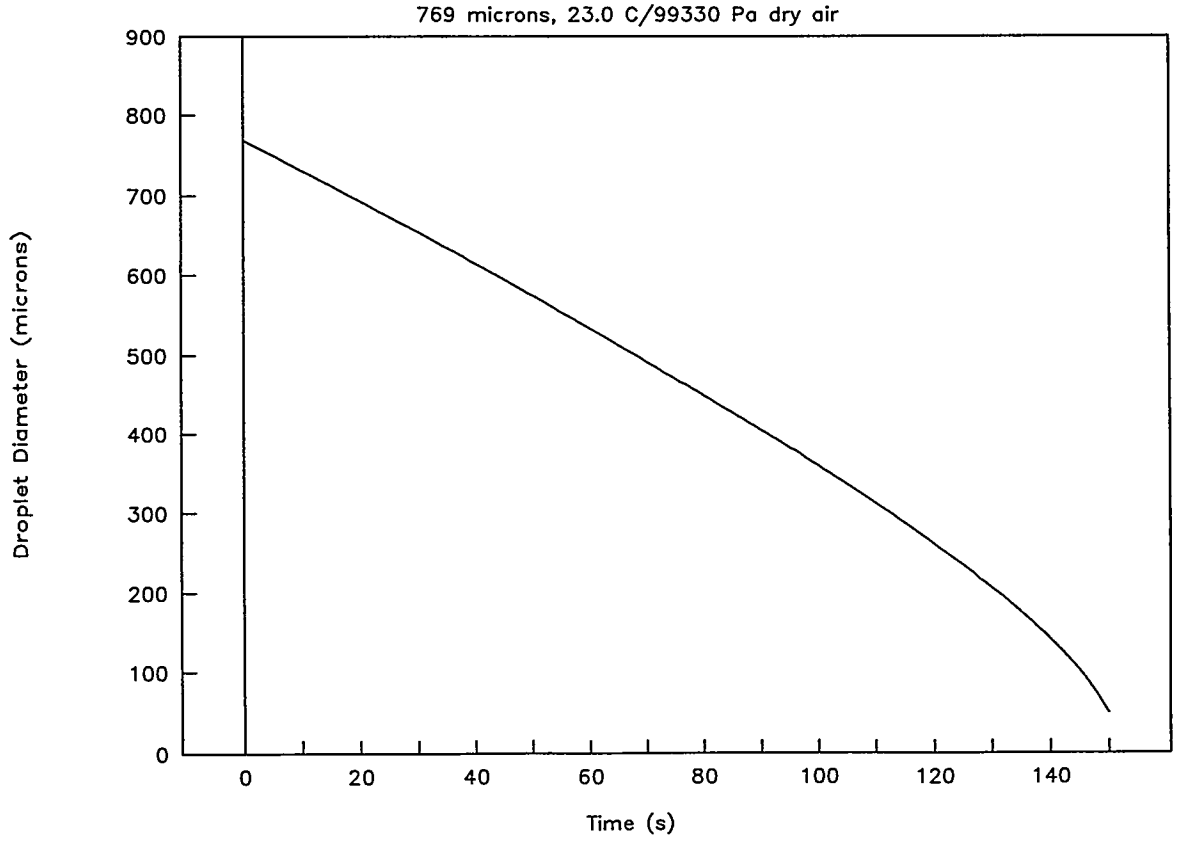
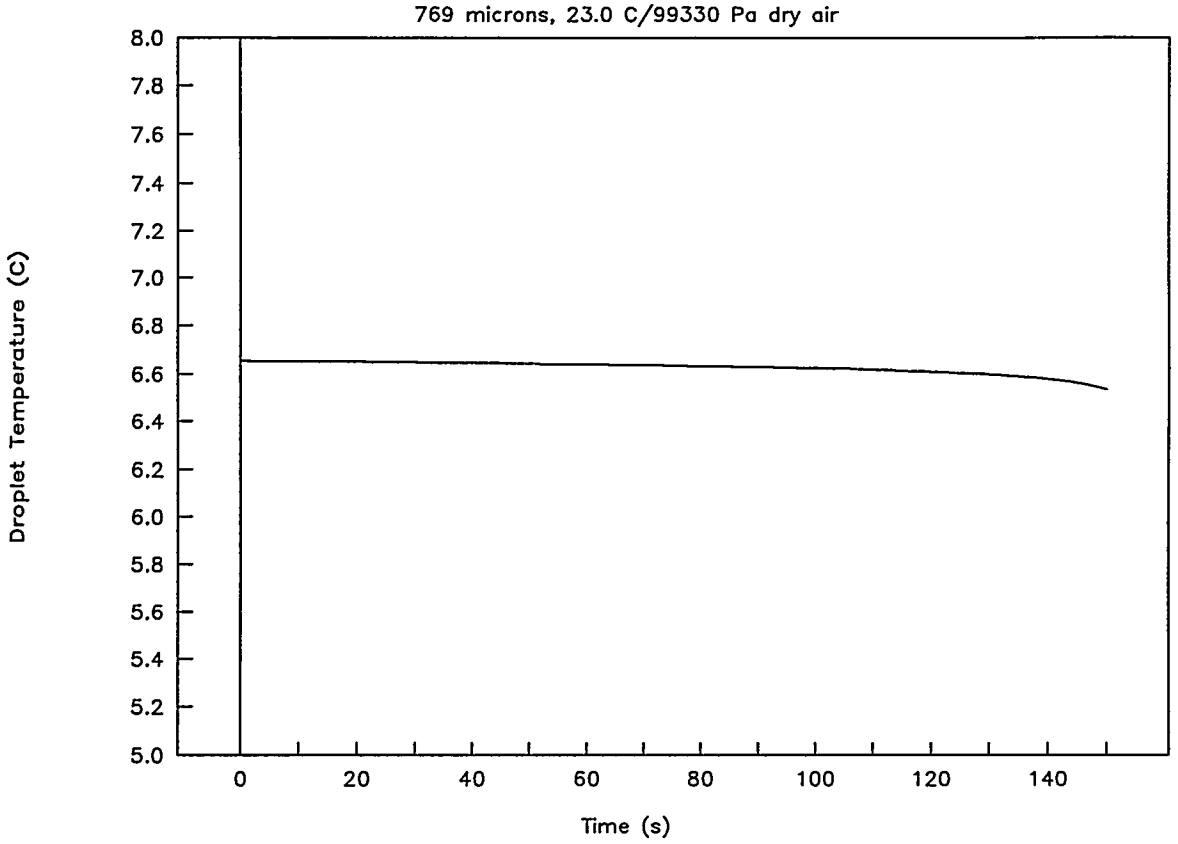


Fig. 5: Water Drop Temperature History



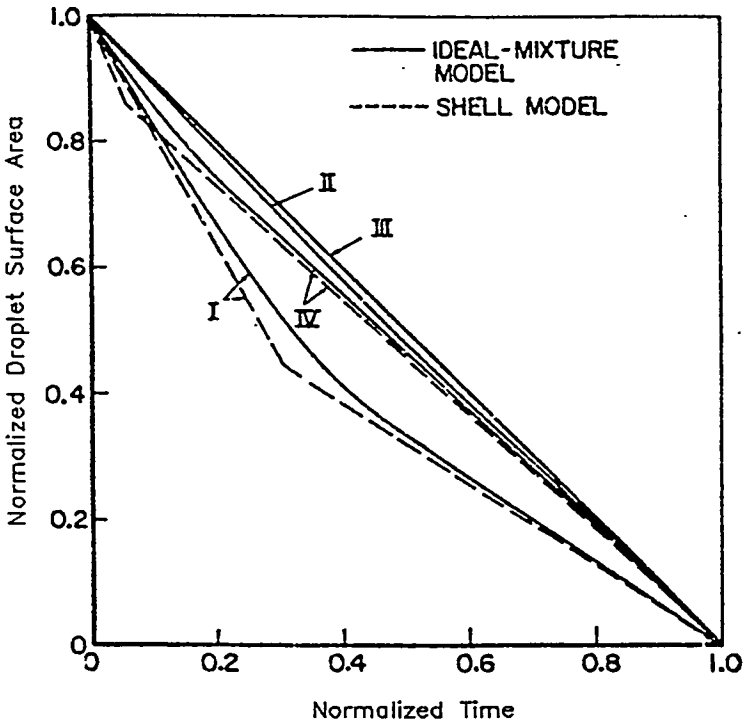


Figure 6: Temporal variation of the droplet surface area for a hexane–octane droplet [after Law (1976)]

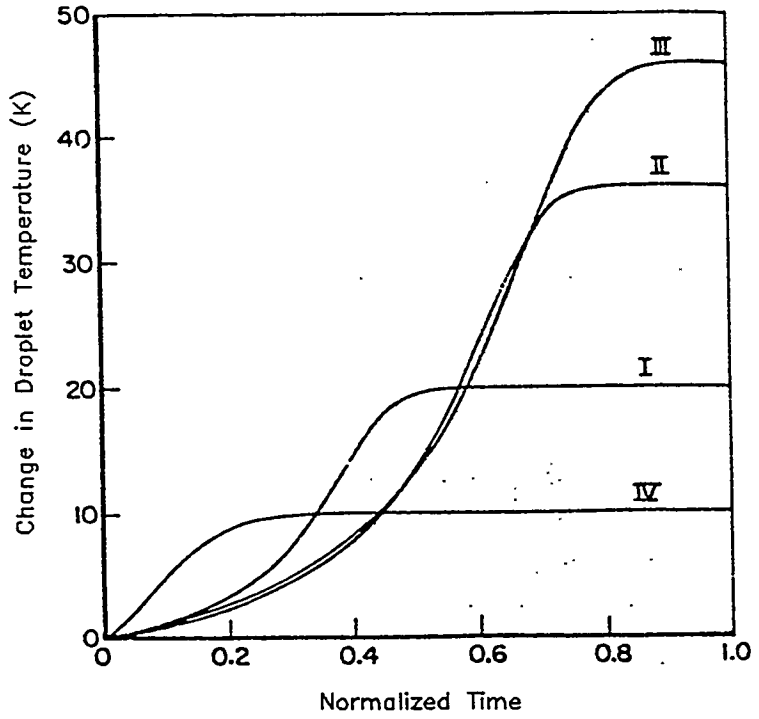


Figure 7: Temporal variation of the droplet temperature for a hexane–octane droplet [after Law (1976)]

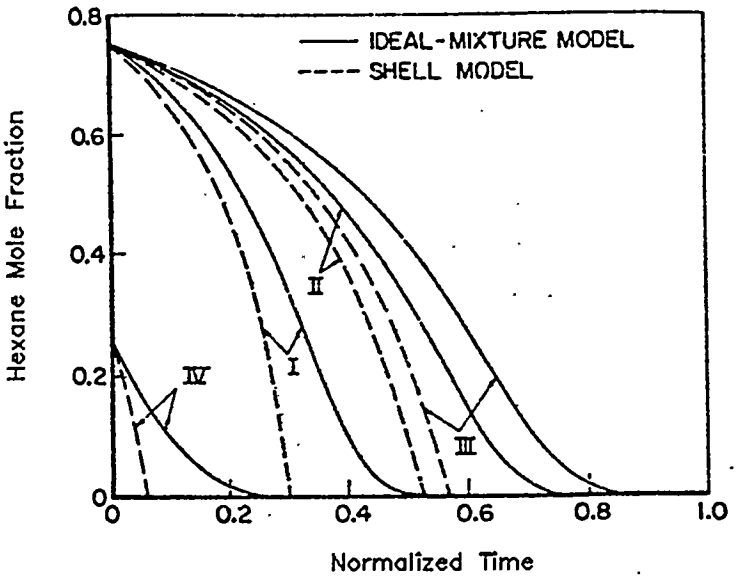


Figure 8: Temporal variation of the droplet composition for a hexane–octane droplet [after Law (1976)]

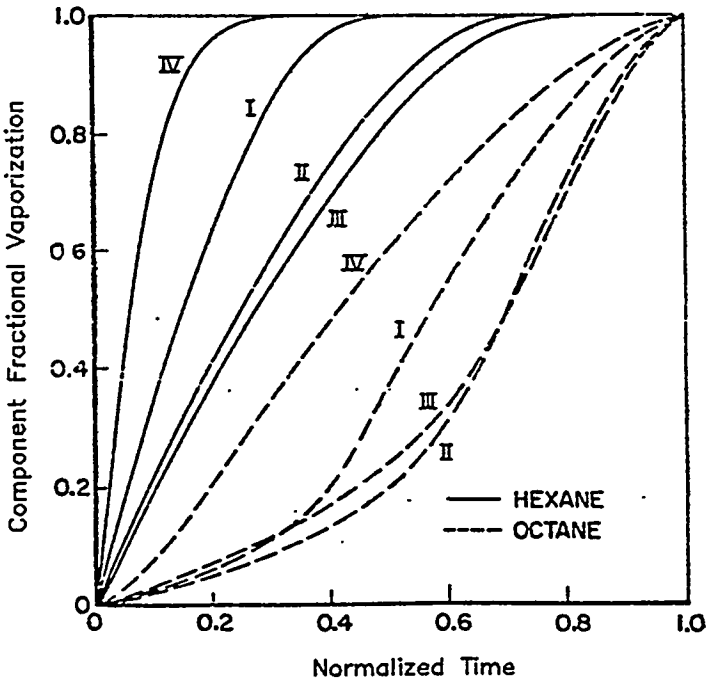


Figure 9: Temporal variation of the fractional amount of each component which has been vaporized for a hexane–octane droplet [after Law (1976)]

components, the diffusion coefficients for both hexane and octane were set equal to the thermal diffusivity of air.

The results of PROGRAM DROP runs for Law's Cases I and IV are shown in Figures 10-13; input and output files for these runs are included in Appendix C. In the figures, the symbols represent the actual output from PROGRAM DROP at discrete time intervals; these symbols are used to distinguish the various curves in the graphs from one another, and do not represent experimental data. In all cases, the qualitative behavior of the curves is in excellent agreement with the results of Law's model shown in Figures 6-9. Again, exact quantitative agreement should not be expected, since the present model treats a freely falling drop and Law's model is for a stationary drop. Nevertheless, the nature of the vaporization process predicted by the two models is clearly similar. Both show the more volatile of the two components (hexane) being preferentially vaporized at the beginning of the droplet lifetime, with the latter portion of the lifetime being dominated by vaporization of essentially pure octane. The temperature curves are also similar, showing an initially cold temperature as the more volatile hexane evaporates, with rising temperature as vaporization shifts to the less volatile octane. Note the significant difference in droplet temperature history for a multicomponent drop (Figures 7 and 11) as opposed to a single component drop (Figure 5). Both models also indicate a more rapid shrinkage in drop size (Figures 6 and 10) early in the drop lifetime as the hexane evaporates than later in the drop lifetime as the octane vaporizes. Agreement between the models' predictions of the droplet compositional variation with time (Figures 8 and 12; Figures 9 and 13) is also reasonable.

To illustrate the use of the model with a nonvolatile, inert component included in the drop, two additional cases were run: a 500-micron, pure water drop, falling in 20 C dry air; and a drop containing the same amount of water, falling in the same atmosphere, but also containing 50% (by mass) solid inert material with density the same as water. Due to addition of the inert material, the initial drop diameter in the latter case was 630 microns. Both cases were run until all of the water had vaporized completely. Input and output files for these runs are included in Appendix C; and time histories of droplet mass, diameter, and settling velocity for these two cases are shown in Figures 14-16, respectively.

As shown in Figure 14, the droplet containing the inert material evaporates much faster than the pure water drop. This is because the drop with inert material is both larger (more surface area for evaporation) and heavier (faster fall velocity, leading to enhanced convective heat and mass transfer rates) than the pure water drop. Figures 15 and 16 clearly show the drop with inert material evaporating to a final, dry particle with nonzero diameter and settling velocity. PROGRAM DROP continues to track the final particle until it impacts the ground.

Finally, a quick glance through Appendix C reveals that the nondimensional parameter C [Equation (1)] for all of the benchmark cases is very close to unity, typically being near 1.01. As discussed in Section 2.1, this supports the present model's exclusion of radial convection and its use of a linearized treatment of mass transfer.

Figure 10: Surface Area History

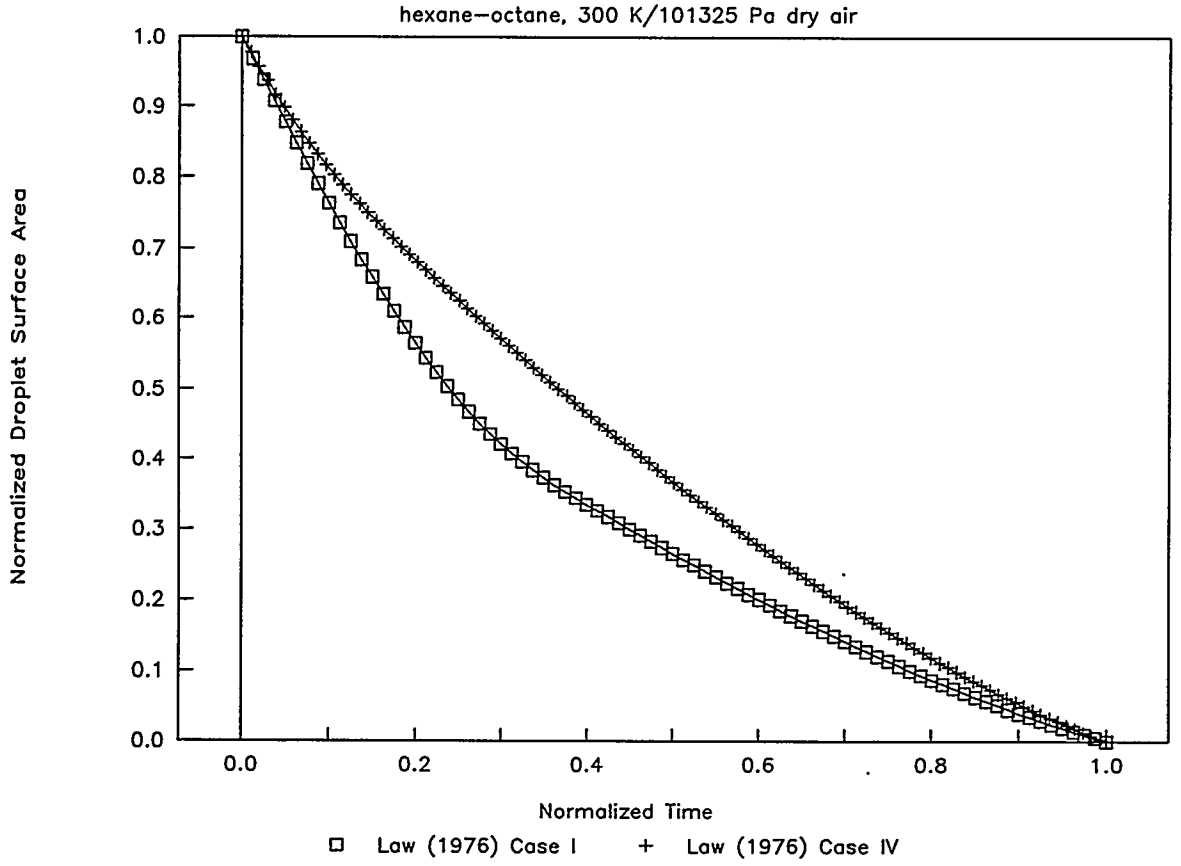


Figure 11: Temperature History

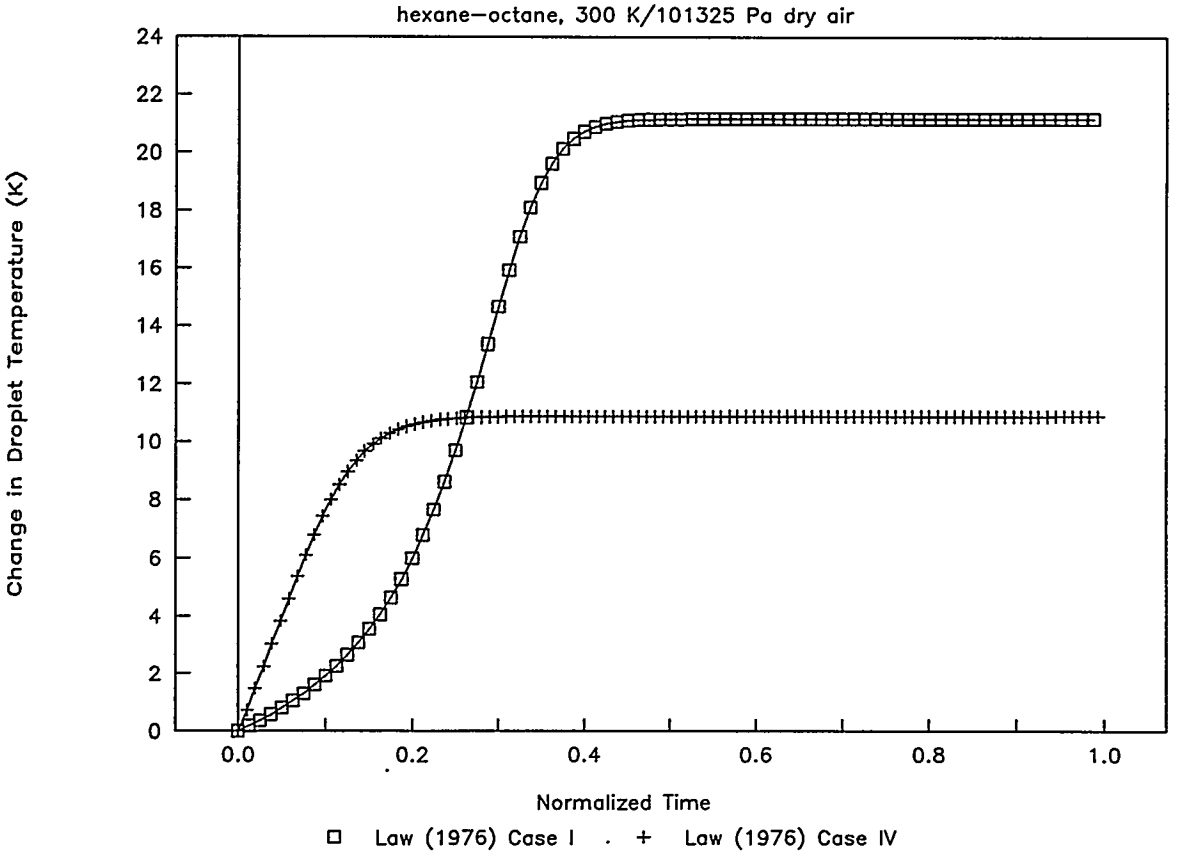


Figure 12: Composition History

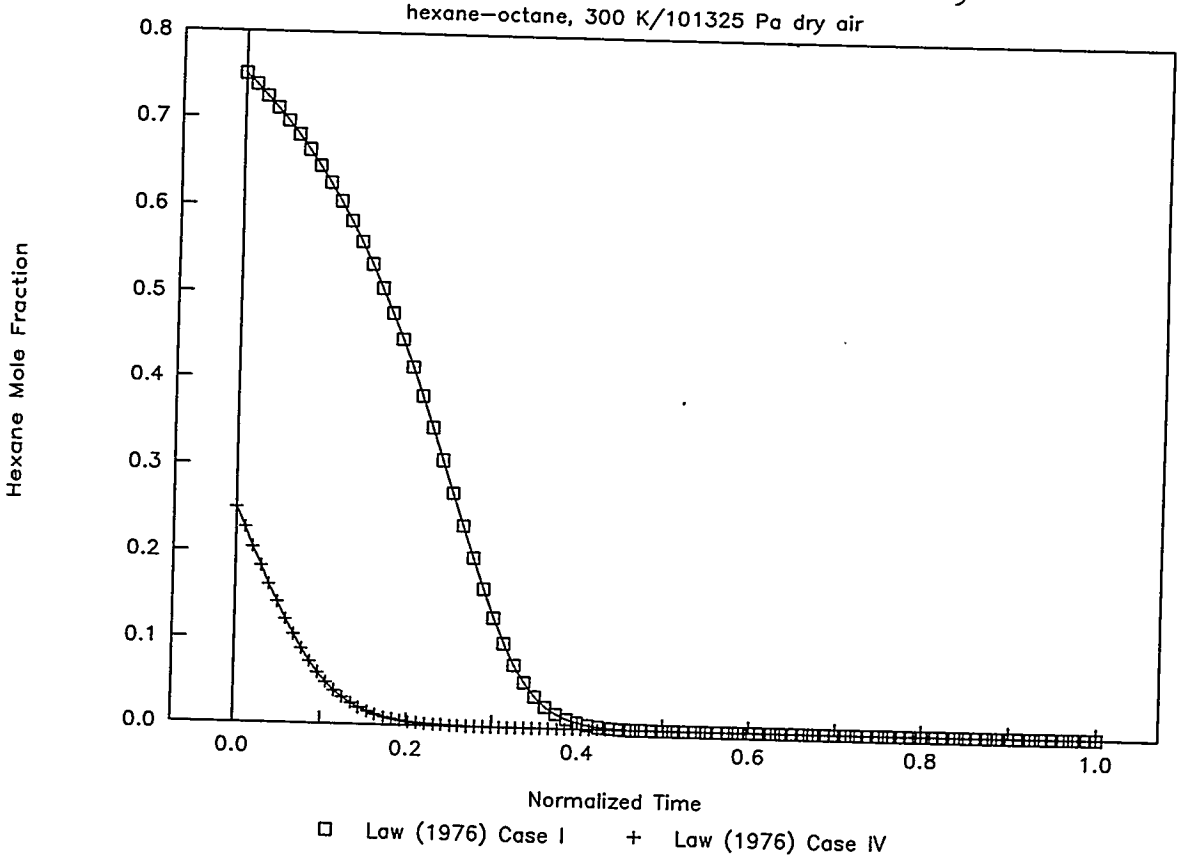


Figure 13: Species Evaporation History

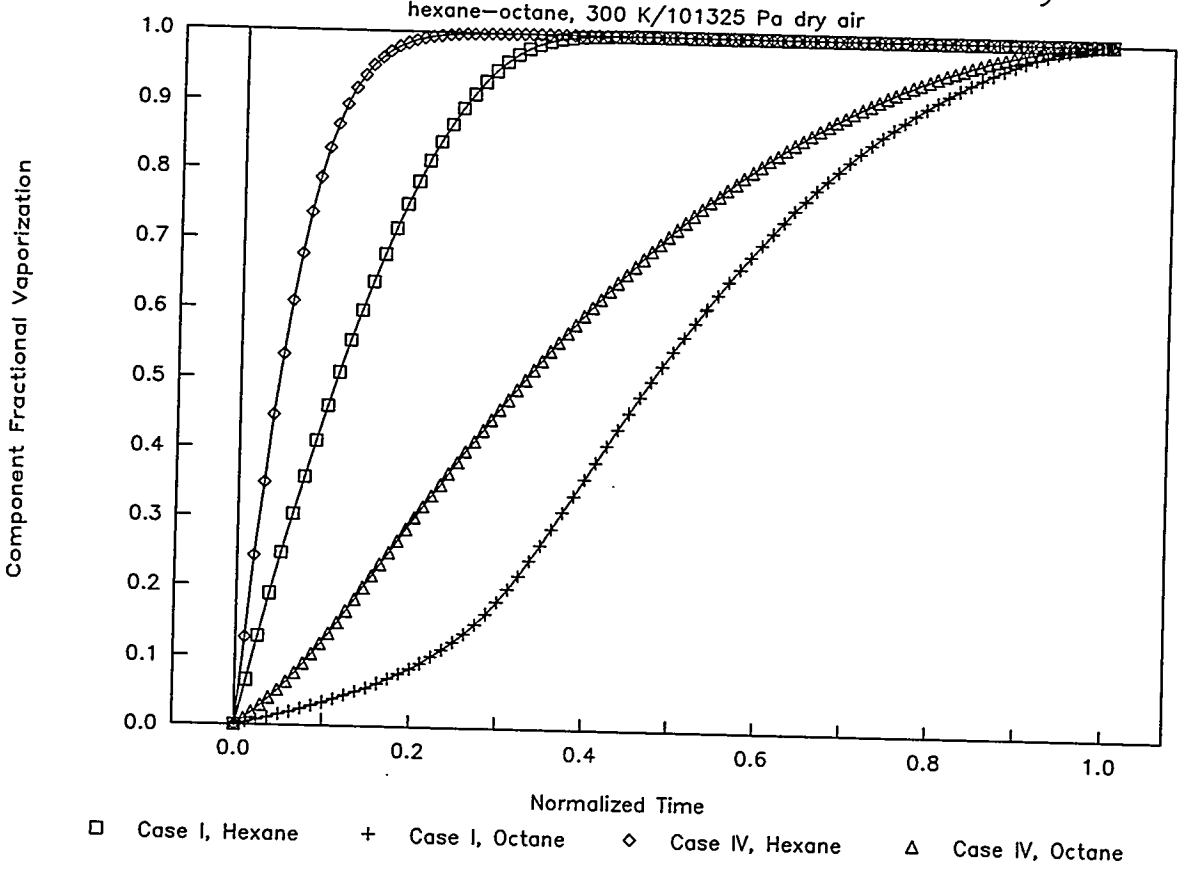


Figure 14: Effect of Inert Component

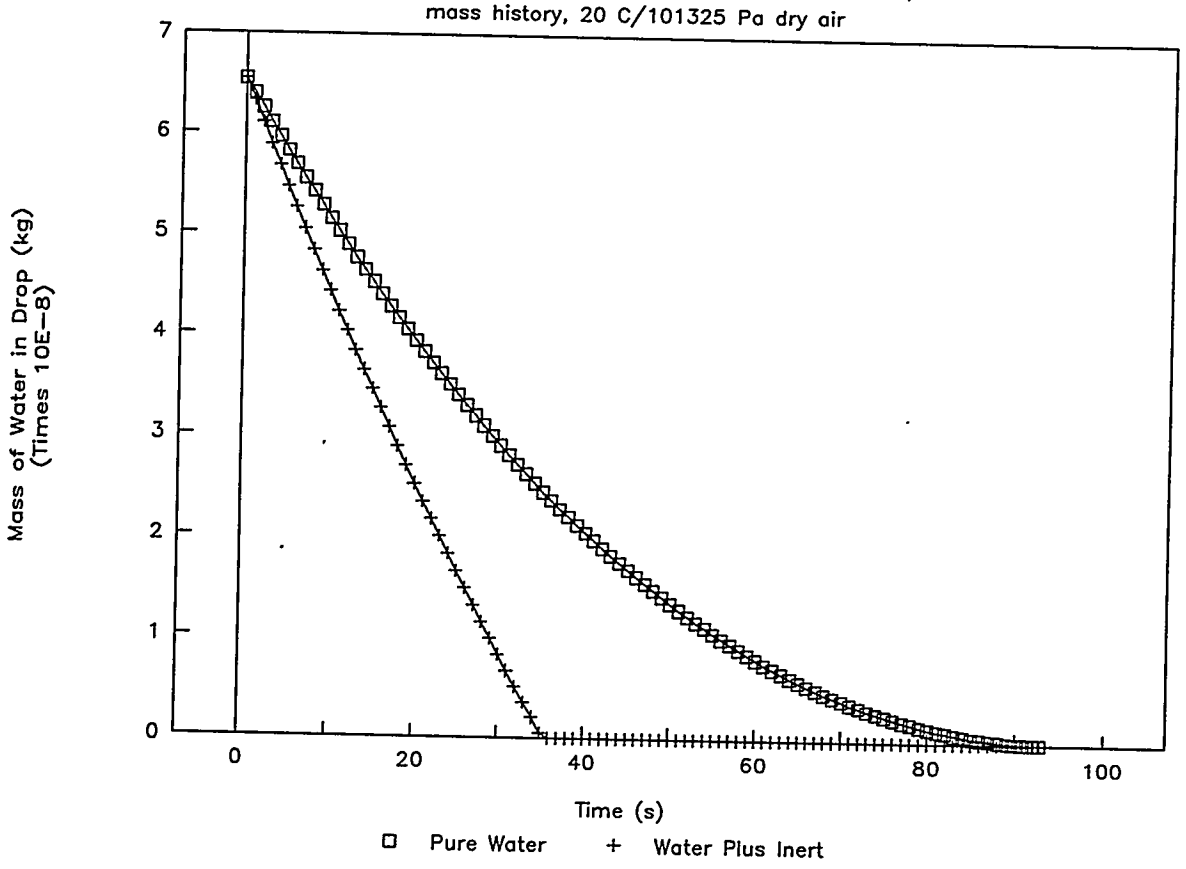


Figure 15: Effect of Inert Component

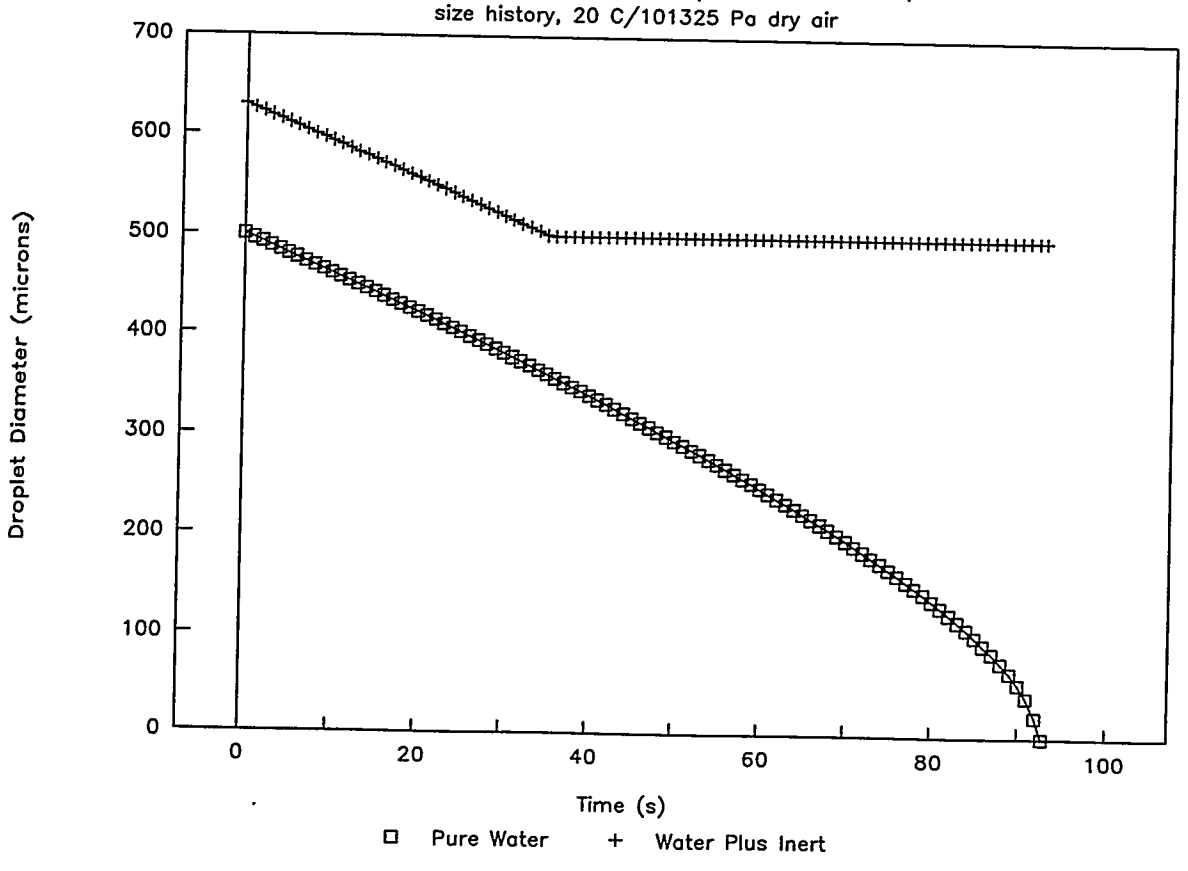
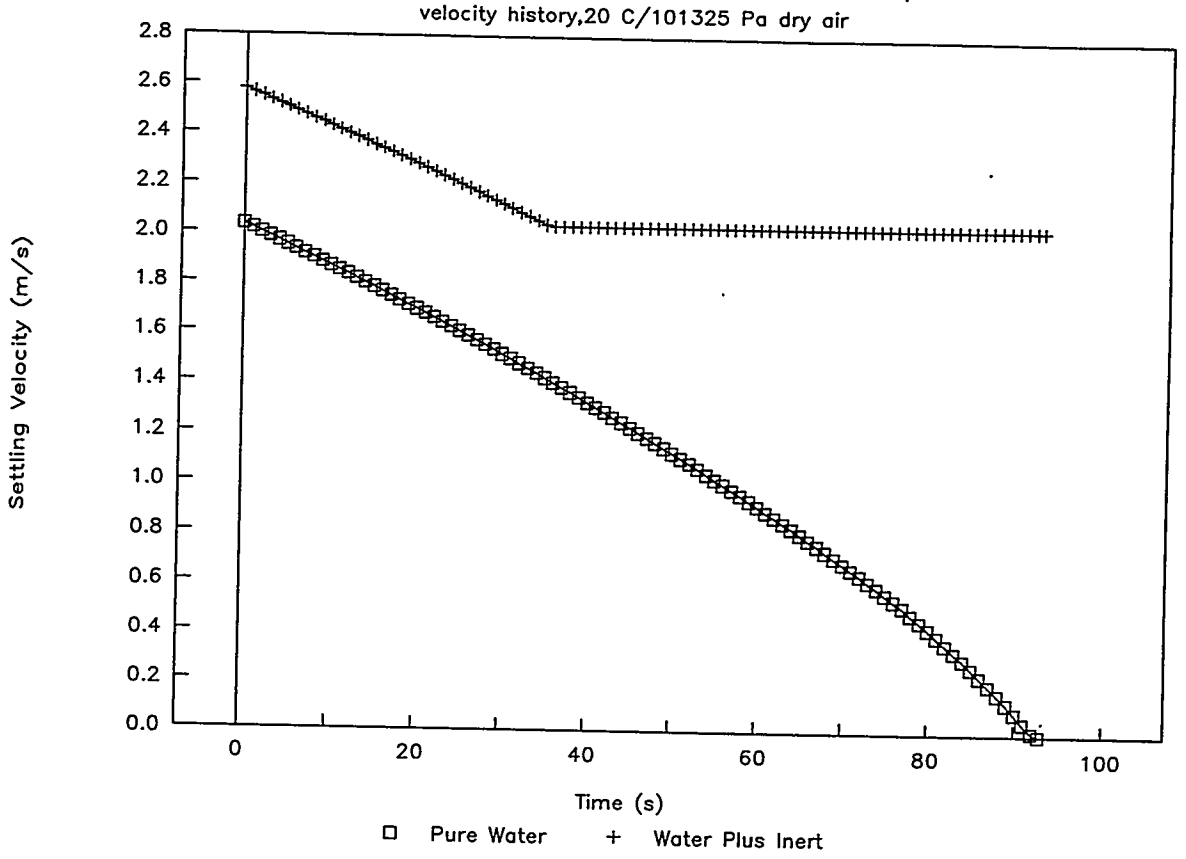


Figure 16: Effect of Inert Component



4. CONCLUSION

A computer program has been written to predict the evaporation of a freely falling, multicomponent drop composed of an arbitrary number of volatile species and a single nonvolatile, inert component. The physics underlying the model have been clearly identified, and the model's relationship to previous work in the literature has been described. The model has been coded in FORTRAN, in a program titled PROGRAM DROP, and has been used to run several benchmark cases. The test cases have illustrated the viability of the model and highlight its potential usefulness in the accurate prediction of multicomponent droplet vaporization in a variety of applications.

References

Beard, K. V., and H. R. Pruppacher, 1969: "A Determination of the Terminal Velocity and Drag of Small Water Drops by Means of a Wind Tunnel," J. Atmos. Sci., 26, 1066-1072.

Beard, K. V., and H. R. Pruppacher, 1971: "A Wind Tunnel Investigation of the Rate of Evaporation of Small Water Drops Falling at Terminal Velocity in Air," J. Atmos. Sci., 28, 1455-1464.

Berry, Edwin X., and Maarten R. Pranger, 1974: "Equations for Calculating the Terminal Velocities of Water Drops," J. Appl. Meteor., 13, 108-113.

Dunn, W. E., 1977: "Heat, Mass and Momentum Transfer in Multidrop Systems," Dept. of Mechanical and Industrial Engineering, Report No. UILU-ENG-77-4006, University of Illinois at Urbana-Champaign, Urbana, IL.

El Wakil, M. M., R. J. Priem, H. J. Brikowski, P. S. Meyers, and O. A. Uyehara, 1956: "Experimental and Calculated Temperature and Mass Histories of Vaporizing Fuel Droplets," NACA TN 3490.

Gavin, Patrick M., 1983: "A Theoretical and Experimental Investigation of Evaporation from Drops Containing Nonvolatile Solutes," Ph.D. Thesis, University of Illinois, Urbana, IL.

Gunn, Ross, and Gilbert D. Kinzer, 1949: "The Terminal Velocity of Fall for Water Droplets in Stagnant Air," J. Meteorology, 6, 243-248.

Hsieh, Jui Sheng, 1975: Principles of Thermodynamics (New York: McGraw-Hill).

Landis, R. B., and A. F. Mills, 1974: "Effect of Internal Diffusional Resistance on the Evaporation of Binary Droplets," Proc. Fifth Int. Heat Transfer Conf., Pap. B7.9, 4, 345-349.

Law, C. K., 1976: "Multicomponent Droplet Combustion with Rapid Internal Mixing," Comb. Flame, 26, 219-233.

Law, C. K., 1978: "Internal Boiling and Superheating in Vaporizing Multicomponent Droplets," A.I.Ch.E.J., 24, 626-632.

Newbold, Fred R., and Neal R. Amundson, 1973: "A Model for Evaporation of a Multicomponent Droplet," A.I.Ch.E.J., 19, 22-30.

Pennsyle, R. O., 1994: Personal communication.

Ranz, William Edwin, 1950: "Drying of Drops," Ph.D. Thesis, University of Wisconsin, Madison, WI.

Ranz, W. E., and W. R. Marshall, Jr., 1952: "Evaporation from Drops - Part I," Chem. Eng. Prog., 48, 141-146.

Ranz, W. E., and W. R. Marshall, Jr., 1952: "Evaporation from Drops - Part II," Chem Eng. Prog., 48, 173-180.

Wood, B. J., H. Wise, and S. H. Inami, 1960: "Heterogeneous Combustion of Multicomponent Fuels," Comb. Flame, 4, 235-242.

Appendix A
Listing of PROGRAM DROP

```
C  PROGRAM DROP
C
C  Computes time histories of mass, velocity, and fall distance
C    for a freely falling multicomponent drop consisting of an
C    arbitrary number of volatile species and a single nonvolatile,
C    inert component.
C
C  The program is currently set up to handle a maximum of five
C    volatile components in the drop, but this can easily be
C    increased by increasing the size of the species arrays.
C
C  Written by:  Patrick M. Gavin
C               Gavin Consulting
C               1599 Krebs Court
C               Newark, OH  43055
C               October, 1994
C
C  Definition of program variables:
C
C  A11,A12   constants in fourth-order polynomials relating droplet
C  A13,A14   Reynolds number (RE) to the dimensionless term XRE,
C  A20,A21   as described by Berry & Pranger (1974)
C  A22,A23
C  ABSERR    absolute error tolerance for integration routine ODE
C  ABSIT     absolute error tolerance in iterative calculation of
C            droplet temperature
C  AD,BD     constants for computing species diffusion coefficient
C            in air:
C            diffusion                      absolute
C            coefficient (m^2/s) = AD * temperature (K) ** BD
C  AL,BL     constants for computing species latent heat of
C            vaporization:
C            latent heat (J/kg) = AL + BL * temperature (C)
C  AP,BP     constants for computing species saturation vapor
C            pressure in accordance with Clausius-Clapeyron
C            equation; AP is the saturation vapor pressure (N/m^2)
C            at temperature BP (K; input in C).  Usually, one will
C            specify species normal boiling conditions, i.e.,
C            AP = 101325 N/m^2 (1 atmosphere), and BP = species
C            normal boiling temperature.
C  AR,BR     constants for computing species density:
C            density (kg/m^3) = AR + BR * temperature (C)
C  CINT      linear interpolation constant used to determine time at
C            which a droplet component evaporates completely or at
C            which the droplet strikes the ground
C  COND      thermal conductivity of air (W/m-K)
C  CPA       constant-pressure specific heat of air (J/kg-K)
C  CRIT      criterion to assess validity of using linearized
C            analysis of multicomponent vaporization, per
C            Newbold & Amundson (1973)
C  D         droplet diameter (m; input in microns)
```

```
C  DIFF     species diffusion coefficient in air (m^2/s)
C  DINERT   density of inert component (kg/m^3)
C  DRY      .TRUE. if all volatile components have evaporated
C           completely; otherwise .FALSE.
C  DT       time increment at which output is desired (s)
C  DYDT     derivative with respect to time of variables (Y)
C           integrated by ODE (kg/s for Y(1)..Y(N), m/s for Y(N+1))
C  F        species mass fraction
C  FINERT   mass fraction of inert component
C  FLUX     species vaporization rate divided by droplet
C           circumference (kg/m-s)
C  FLUXA    average of FLUXM and FLUXT
C  FLUXM    latent heat consumed by vaporization of all volatile
C           species in drop, per unit of droplet circumference (W/m)
C  FLUXT    sensible heat transferred to droplet from air, per
C           unit of droplet circumference (W/m)
C  FNAME    file name for program input and output
C  FPT      first derivative of droplet energy balance function (FT)
C           with respect to temperature (W/m-K)
C  FRHO     sum of inverse species densities, weighted by their
C           mass fractions; used to calculate initial droplet
C           mass (m^3/kg)
C  FT       droplet energy balance function, defined as the difference
C           between FLUXM and FLUXT; driven to near zero via Newton-
C           Raphson iteration in FUNCTION TEMP to determine droplet
C           temperature (W/m)
C  FTOT     sum of species mass fractions; must equal unity or
C           program will abort
C  GRAV     acceleration of gravity (m/s^2)
C  I        DO-loop index variable; species number
C  IFLAG    error flag for integration routine ODE
C  K        number of equations integrated by ODE, which is
C           one plus the number of volatile species, to account
C           for integration of drop velocity to obtain fall
C           distance
C  L        DO-loop index at which a zero-crossing occurred,
C           either in species mass or droplet height
C  LAMBDA   species latent heat of vaporization (J/kg)
C  LNX      natural logarithm of XRE
C  MASS     droplet mass (kg)
C  MINERT   mass of inert component in drop (kg)
C  MW       species molecular weight
C  N        number of volatile species in drop
C  NU       droplet Nusselt number
C  PA       species vapor pressure in surrounding air (N/m^2)
C  PAMB     ambient air (barometric) pressure (N/m^2)
C  PERM     intermediate variable used in calculation of FT and FPT;
C           see FUNCTION TEMP
C  PFIRI    term used in SUBROUTINE PROPS to correct ambient air
C           density for presence of droplet volatile species
C           in gas phase (kg-K/m^3)
```

```
C  PI       pi (3.141593)
C  PI8      eight divided by pi
C  PID      droplet circumference * -1 (m)
C  PISIX    pi divided by six
C  PR       Prandtl number of air
C  PS       species saturation vapor pressure (N/m^2)
C  R        species ideal gas constant (N-m/kg-K)
C  RA       air ideal gas constant (N-m/kg-K)
C  RE       droplet Reynolds number
C  RELERR   relative error tolerance for integration routine ODE
C  RELIT    relative error tolerance in iterative calculation of
C           droplet temperature
C  RHO      species density (kg/m^3)
C  RHOG     density of air (kg/m^3)
C  RU       universal ideal gas constant (N-m/kgmole-K)
C  SC       species Schmidt number
C  SH       droplet Sherwood number
C  SIXPI    six divided by pi
C  SIXRE2   0.6 times the square root of droplet Reynolds number
C  SUMPA    sum of ambient species vapor pressures
C  SUMPS    sum of species vapor pressures at droplet surface
C  T        droplet temperature (C)
C  TA       ambient air temperature (C)
C  TF       film temperature for physical property evaluation (C)
C  TFK      film temperature for physical property evaluation (K)
C  THIRD    one-third (1/3)
C  TIME     time elapsed since droplet release (s)
C  TK       droplet temperature (K)
C  TOUT     time at which output is desired from ODE (s)
C  TSEED    initial guess for iterative calculation of droplet
C           temperature (C)
C  U        droplet settling velocity (m/s)
C  VINERT   volume of inert component in drop
C  VISC     viscosity of air (kg/m-s)
C  VOLUM    droplet volume (m^3)
C  WORK     calculation space required by ODE
C  X        species mole fraction in drop
C  XRE      dimensionless term (equal to drag coefficient times
C           Reynolds number squared) used to determine droplet
C           settling velocity given its mass, as described by
C           Berry & Pranger (1974)
C  XTOT     total moles of volatile species in drop
C  Y        array of variables to be integrated by ODE:
C           Y(1) - Y(N):  mass of volatile species 1-N (kg)
C           Y(N+1):  droplet fall distance (m)
C  YPREV    values of variables in array Y from previous
C           output time increment
C  Z        height above ground at which drop is released (m)
C  ZKELV    absolute temperature conversion factor (K)
C
C  Begin main program
```

```
C
C  The main program handles input and output, orchestrates
C  integration of the rate equations, and terminates the
C  run when the drop evaporates completely or hits the ground
C
      EXTERNAL DDROP
C
      REAL MW(5),MASS,MINERT
      CHARACTER*80 FNAME
C
      DIMENSION R(5),F(5),Y(6),FLUX(5),YPREV(6),WORK(226)
      DIMENSION AP(5),BP(5),AD(5),BD(5),AL(5),BL(5),AR(5),BR(5)
C
      COMMON/VAPORS/N,MW(5),R(5)
      COMMON/COEFFS/AP(5),BP(5),AD(5),BD(5),AL(5),BL(5),AR(5),BR(5)
      COMMON/TOTEMP/TSEED,VINERT,MINERT
      COMMON/FRTEMP/VOLUM,FLUX(5),RE,VISC,RHOG,CRIT
C
      DATA PISIX/0.523599/,ZKELV/273.15/,RU/8315./
      DATA SIXPI/1.909859/,THIRD/0.333333/
      DATA RELERR/1.0E-5/,ABSERR/1.0E-13/
C
C  Prompt user for input and output filenames
C
      PRINT * "ENTER INPUT FILE NAME"
      READ * FNAME
      OPEN (UNIT=5,FILE=FNAME)
      PRINT * "ENTER OUTPUT FILE NAME"
      READ * FNAME
      OPEN (UNIT=6,FILE=FNAME)
C
C  Read initial input data (note drop diameter input in microns)
C
      READ(5,500) D,T,Z,N,DT
      READ(5,501) FINERT,DINERT
      FRHO=FINERT/DINERT
      FTOT=FINERT
C
C  Read physical property data for volatile species
C
      DO 10 I=1,N
        READ(5,501) F(I),MW(I)
        READ(5,501) AP(I),BP(I),AD(I),BD(I),AL(I),BL(I),AR(I),BR(I)
        BP(I)=BP(I)+ZKELV
        AL(I)=AL(I)-BL(I)*ZKELV
        RHO=AR(I)+BR(I)*T
        FRHO=FRHO+F(I)/RHO
        FTOT=FTOT+F(I)
10    CONTINUE
C
C  Check that mass fractions add up to one; if not, abort
```

```
C
      IF(FTOT.EQ.1.0) GO TO 20
      WRITE(6,600)
      STOP
C
C  Compute initial masses of volatile and inert components
C
20    MASS=PISIX*(D*1.E-6)**3/FRHO
      DO 30 I=1,N
      Y(I)=F(I)*MASS
      R(I)=RU/MW(I)
30    CONTINUE
      MINERT=FINERT*MASS
      VINERT=MINERT/DINERT
C
C  Initialize variables for integration routine
C
      TSEED=T
      K=N+1
      Y(K)=Z
      TIME=0.0
      IFLAG=1
C
C  Begin calculation loop
C
C  Determine droplet temperature, diameter, and velocity at
C  output time increment
C
40    T=TEMP(Y)
      D=(SIXPI*VOLUM)**THIRD
      U=0.0
      IF(RE.GT.0.0) U=-VISC*RE/(RHOG*D)
C
C  Write values to output file
C
      WRITE(6,*) TIME,T,D,U,(Y(I),I=1,K),CRIT
C
C  If drop has struck ground or vaporized completely, stop program
C
      IF(Y(K).EQ.0.0.OR.RE.LE.0.0) STOP
C
C  Increment output time, and take another time step
C
      TOUT=TIME+DT
      CALL ODE (DDROP,K,Y,TIME,TOUT,RELERR,ABSERR,IFLAG,WORK)
C
C  If integration routine error, stop the program
C
      IF(IFLAG.EQ.2) GO TO 45
      WRITE(6,601) IFLAG
      STOP
```

```
C
C  Determine if one of the volatile species has vaporized
C  completely, or if the drop has struck ground, during the
C  current time step (i.e., a zero crossing)
C
45    DO 50 I=1,K
        IF(Y(I).GE.0.0) GO TO 50
        L=I
        GO TO 60
50    CONTINUE
      GO TO 80
C
C  Zero crossing has occurred, so do linear interpolation to
C  determine variable values at the zero crossing
C
60    CINT=(0.0-YPREV(L))/(Y(L)-YPREV(L))
      TIME=TIME-(1.0-CINT)*DT
      DO 70 I=1,K
        Y(I)=YPREV(I)+(Y(I)-YPREV(I))*CINT
70    CONTINUE
      Y(L)=0.0
C
C  Reset integration flag to re-start integration from point
C  of zero crossing
C
      IFLAG=1
C
C  Store variable values at this time step in case a zero
C  crossing occurs during the next time step
C
80    DO 90 I=1,K
        YPREV(I)=Y(I)
90    CONTINUE
C
C  Return to start of calculation loop
C
      GO TO 40
C
500   FORMAT(3E10.0,I10,2E10.0)
501   FORMAT(8E10.0)
C
600   FORMAT(/," ERROR IN MASS FRACTION SPECIFICATION")
601   FORMAT(/," IFLAG =",I5,"; EXECUTION TERMINATED")
C
      END
```

```
      FUNCTION TEMP (Y)
C
C  FUNCTION TEMP determines the quasi-steady droplet temperature
C  given the droplet composition and position
C
      LOGICAL DRY
C
      REAL MW(5),MASS,MINERT,NU,LAMBDA(5)
C
      DIMENSION Y(*)
      DIMENSION RHO(5),X(5),SC(5),R(5),FLUX(5),DIFF(5),PS(5),PA(5)
      DIMENSION PERM(5)
C
      COMMON/VAPORS/N,MW(5),R(5)
      COMMON/TOTEMP/TSEED,VINERT,MINERT
      COMMON/FRTEMP/VOLUM,FLUX(5),RE,VISC,RHOG,CRIT
C
      DATA PI8/2.546479/,GRAV/9.801/,THIRD/0.333333/,ZKELV/273.15/
      DATA RELIT/1.0E-5/,ABSIT/0.0/
C
C  To start, assume there are volatile species present, and
C  compute mole fraction of each volatile component as well
C  as total moles and mass of volatile components in drop
C
      DRY=.FALSE.
      XTOT=0.0
      DO 10 I=1,N
        X(I)=Y(I)/MW(I)
        XTOT=XTOT+X(I)
10    CONTINUE
      MASS=0.0
      DO 20 I=1,N
        IF(XTOT.GT.0.0) X(I)=X(I)/XTOT
        MASS=MASS+Y(I)
20    CONTINUE
C
C  Set droplet temperature at initial guess
C
      TEMP=TSEED
C
C  Check if there are any volatile components left
C
      IF(MASS.GT.0.0) GO TO 25
C
C  No volatile components left, so set all mass flux terms
C  to zero, and set droplet temperature equal to ambient
C  air temperature
C
      DRY=.TRUE.
      DO 21 I=1,N
        FLUX(I)=0.0
```

```
21    CONTINUE
      CALL AMB (Y(N+1),TA,PA,PAMB)
      TEMP=TA
C
C  Add in mass of inert component to arrive at total drop mass
C
25    MASS=MASS+MINERT
C
C  Begin Newton-Raphson iteration loop
C
      DO 60 J=1,50
C
C  Compute physical properties of drop and ambient at current
C  drop temperature and position
C
        CALL PROPS (TEMP,Y(N+1),X,TA,TF,PA,PAMB,RHOG,VISC,COND,PR,
     $    PS,DIFF,LAMBDA,RHO,SC)
C
C  Compute droplet volume as sum of volumes of individual species
C
        VOLUM=VINERT
        DO 30 I=1,N
          VOLUM=VOLUM+Y(I)/RHO(I)
30      CONTINUE
C
C  Compute Berry & Pranger (1974) term "X" (drag coefficient times
C  Reynolds number squared) from current droplet mass and volume,
C  and use it to calculate droplet Reynolds number
C
        XRE=PI8*RHOG*(MASS-RHOG*VOLUM)*GRAV/(VISC*VISC)
        RE=REYN(XRE)
C
C  If no volatile species are present, no need to proceed further,
C  so return to calling program
C
        IF(DRY) RETURN
C
C  Calculate sensible heat flux to droplet from ambient using heat
C  transfer coefficient determined from Ranz & Marshall (1952)
C  correlation
C
        SIXRE2=0.6*SQRT(RE)
        NU=2.0+SIXRE2*PR**THIRD
        FPT=COND*NU
        FLUXT=FPT*(TA-TEMP)
C
C  Calculate latent heat flux from droplet due to evaporation of
C  all volatile species, using mass transfer coefficient determined
C  from Ranz & Marshall (1952) correlation
C
        FLUXM=0.0
```

```
      TFK=TF+ZKELV
      DO 40 I=1,N
        SH=2.0+SIXRE2*SC(I)**THIRD
        PERM(I)=DIFF(I)*SH/(R(I)*TFK)
        FLUX(I)=0.0
        IF(Y(I).GT.0.0) FLUX(I)=PERM(I)*(X(I)*PS(I)-PA(I))
        FLUXM=FLUXM+LAMBDA(I)*FLUX(I)
40    CONTINUE
C
C  Droplet is at quasi-steady temperature when sensible and latent
C  heat fluxes balance...compute difference between latent and
C  sensible fluxes and compare to average of latent and sensible
C  fluxes to determine if Newton-Raphson iteration has converged
C
      FLUXA=0.5*(FLUXM+FLUXT)
      FT=FLUXM-FLUXT
      IF(ABS(FT).LT.RELIT*ABS(FLUXA)+ABSIT) THEN
C
C  Iteration has converged; set seed for next call to TEMP at
C  current computed temperature, and calculate Newbold &
C  Amundson (1973) criterion (CRIT) for validity of linearized
C  analysis of multicomponent vaporization
C
        TSEED=TEMP
        SUMPA=0.0
        SUMPS=0.0
        DO 45 I=1,N
          SUMPA=SUMPA+PA(I)
          SUMPS=SUMPS+X(I)*PS(I)
45        CONTINUE
        CRIT=(PAMB-SUMPA)/(PAMB-SUMPS)
        RETURN
C
C  Iteration has not converged; compute Newton-Raphson correction term
C
      ELSE
        TK=TEMP+ZKELV
        FPT=FPT-0.5*FLUXM/TFK
        DO 50 I=1,N
          FPT=FPT+PERM(I)*LAMBDA(I)**2*X(I)*PS(I)/(R(I)*TK*TK)
50        CONTINUE
        TEMP=TEMP-FT/FPT
      END IF
C
C  Return to start of Newton-Raphson iteration loop
C
60    CONTINUE
C
C  Convergence has not been achieved after fifty iterations;
C  write error message and halt program execution
C
```

```
      WRITE(6,600)
      STOP
C
600   FORMAT(/," TEMPERATURE ITERATION DID NOT CONVERGE")
C
      END
```

```
      FUNCTION REYN (XRE)
C
C  FUNCTION REYN calculates droplet Reynolds number given the
C  dimensionless term XRE, equal to the drag coefficient times
C  Reynolds number squared
C
C  The correlations used herein are taken directly from Berry
C  & Pranger, 1974:  "Equations for Calculating the Terminal
C  Velocities of Water Drops," J. Appl. Meteor., 13, 108-113
C
      REAL LNX
C
      DATA A11/+0.412657E-1/,A12/-0.150074E-3/
      DATA A13/+0.758804E-6/,A14/-0.168841E-8/
      DATA A20/-0.236534E+1/,A21/+0.767787E+0/
      DATA A22/+0.535826E-2/,A23/-0.763554E-3/
C
      IF (XRE.LE.175.270) THEN
        REYN=XRE*(A11+XRE*(A12+XRE*(A13+XRE*A14)))
      ELSE
        LNX=LOG(XRE)
        REYN=EXP(A20+LNX*(A21+LNX*(A22+LNX*A23)))
      END IF
C
      RETURN
      END
```

```
      SUBROUTINE PROPS (T,Z,X,TA,TF,PA,PAMB,RHOG,VISC,COND,PR,
     $  PS,DIFF,LAMBDA,RHO,SC)
C
C  SUBROUTINE PROPS calculates the physical properties of the drop
C  and surrounding air, given the current drop temperature and
C  position
C
      REAL LAMBDA(5),MW(5)
      DIMENSION PS(5),PA(5),DIFF(5),RHO(5),SC(5),R(5),X(5)
      DIMENSION AP(5),BP(5),AD(5),BD(5),AL(5),BL(5),AR(5),BR(5)
C
      COMMON/VAPORS/N,MW(5),R(5)
      COMMON/COEFFS/AP(5),BP(5),AD(5),BD(5),AL(5),BL(5),AR(5),BR(5)
C
      DATA ZKELV/273.15/,RA/287.04/,CPA/1006./
C
C  Obtain ambient atmospheric properties at current drop position
C
      CALL AMB (Z,TA,PA,PAMB)
C
C  Calculate film temperature for physical property evaluation
C
      TF=0.5*(T+TA)
      TFK=TF+ZKELV
      TK=T+ZKELV
C
C  Determine latent heat of vaporization, saturation vapor pressure,
C  diffusion coefficient, and liquid-phase density for each
C  volatile component in the drop
C
C  Also compute correction to ambient air density due to presence
C  of droplet volatile species in the gas phase
C
      PFIRI=0.0
      DO 10 I=1,N
        LAMBDA(I)=AL(I)+BL(I)*TK
        PS(I)=AP(I)*(TK/BP(I))**(BL(I)/R(I))*
     $    EXP(AL(I)/R(I)*(1.0/BP(I)-1.0/TK))
        DIFF(I)=AD(I)*TFK**BD(I)
        RHO(I)=AR(I)+BR(I)*T
        PFIRI=PFIRI+(X(I)*PS(I)+PA(I))*(1.0/R(I)-1.0/RA)
10    CONTINUE
C
C  Compute air density corrected for presence of volatile species
C
      RHOG=(PAMB/RA+0.5*PFIRI)/TFK
C
C  Compute air viscosity, thermal conductivity, and Prandtl number
C
      VISC=1.4963E-6*TFK**1.5/(TFK+120.)
      COND=2.367E-2+6.865E-5*TF
```

```
      PR=VISC*CPA/COND
C
C  Compute Schmidt number for each volatile component in drop
C
      DO 20 I=1,N
        SC(I)=VISC/(RHOG*DIFF(I))
20    CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE AMB (Z,TA,PA,PAMB)
C
C  Given the droplet vertical elevation, SUBROUTINE AMB returns
C  the local atmospheric temperature and total pressure, as well
C  as ambient vapor pressures, if any, of volatile species
C  present in the drop
C
C  Currently, ambient temperature and pressure are set at constant
C  values for purposes of illustrating performance of the code.
C  In actual applications, temperature and pressure may be profiled
C  with height above ground.
C
C  Vapor pressures of volatile drop species are also currently set
C  to zero, which may be a reasonable general assumption except in
C  cases where the droplet contains water.  In these cases, care
C  must be taken to include the presence of atmospheric water vapor
C  (humidity).
C
      REAL MW(5)
C
      DIMENSION PA(5)
C
      COMMON/VAPORS/N,MW(5),R(5)
C
      TA=20.0
C
      DO 10 I=1,N
        PA(I)=0.0
10    CONTINUE
C
      PAMB=101325.
C
      RETURN
      END
```

```
      SUBROUTINE DDROP (TIME,Y,DYDT)
C
C  SUBROUTINE DDROP calculates the first derivatives with respect
C  to time of all variables integrated by ODE, namely, the masses
C  of all volatile species in the drop (Y(1)..Y(N)) and the drop
C  vertical elevation (Y(N+1))
C
      REAL MW(5)
C
      DIMENSION Y(*),DYDT(*)
      DIMENSION R(5),FLUX(5)
C
      COMMON/VAPORS/N,MW(5),R(5)
      COMMON/FRTEMP/VOLUM,FLUX(5),RE,VISC,RHOG,CRIT
C
      DATA SIXPI/1.909859/,THIRD/0.333333/,PI/3.141593/
C
C  Compute current droplet temperature and related parameters
C  passed through common block FRTEMP
C
      T=TEMP(Y)
C
C  Determine droplet diameter and circumference
C
      D=(SIXPI*VOLUM)**THIRD
      PID=-PI*D
C
C  Compute volatile species evaporation rates from mass fluxes
C
      DO 10 I=1,N
        DYDT(I)=PID*FLUX(I)
10    CONTINUE
C
C  If drop has not evaporated completely, compute rate of change
C  of droplet vertical elevation, which is negative of settling
C  velocity
C
      DYDT(N+1)=0.0
      IF(RE.GT.0.0) DYDT(N+1)=-VISC*RE/(RHOG*D)
C
      RETURN
      END
```

Appendix B
Listing of SUBROUTINE ODE

```
      subroutine ode(f,neqn,y,t,tout,relerr,abserr,iflag,work)
c
c...  this version of ode has a new interpolant as described in
c...  "a smoother interpolant for de/step, intrp and deabm,"
c...  l. f. shampine and h. a. watts, sand83-1226.
c
c
c   subroutine  ode  integrates a system of neqn first order
c   ordinary differential equations of the form
c            dy(i)/dt=f(t,y(1),y(2),...,y(neqn))
c            y(i) given at  t .
c   the subroutine integrates from  t  to  tout .  on return the
c   parameters in the call list are set for continuing the integration.
c   the user has only to define a new value  tout  and call  ode  again.
c
c   the differential equations are actually solved by a suite of codes
c   de ,  step , and  intrp .  ode  allocates virtual storage in the
c   arrays  work  and  iwork  and calls  de .  de  is a supervisor which
c   directs the solution.  it calls on the routines  step  and  intrp
c   to advance the integration and to interpolate at output points.
c   step  uses a modified divided difference form of the adams pece
c   formulas and local extrapolation.  it adjusts the order and step
c   size to control the local error per unit step in a generalized
c   sense.  normally each call to  step  advances the solution one step
c   in the direction of  tout .  for reasons of efficiency  de
c   integrates beyond  tout  internally, though never beyond
c   t+10*(tout-t), and calls  intrp  to interpolate the solution at
c   tout .  an option is provided to stop the integration at  tout  but
c   it should be used only if it is impossible to continue the
c   integration beyond  tout .
c
c   this code is completely explained and documented in the text,
c   computer solution of ordinary differential equations:  the initial
c   value problem  by l. f. shampine and m. k. gordon.
c
c   the parameters represent:
c      f -- subroutine f(t,y,yp) to evaluate derivatives yp(i)=dy(i)/dt
c      neqn -- number of equations to be integrated
c      y(*) -- solution vector at  t
c      t -- independent variable
c      tout -- point at which solution is desired
c      relerr,abserr -- relative and absolute error tolerances for local
c           error test.  at each step the code requires
c             abs(local error).le.abs(y)*relerr+abserr
c           for each component of the local error and solution vectors
c      iflag -- indicates status of integration
c      work(*) -- array to hold information internal to code
c           which is necessary for subsequent calls
c
c   first call to ode --
c
c   the user must provide storage in his calling program for the arrays
c   in the call list,
c      y(neqn), work(100+21*neqn),
c   declare  f  in an external statement, supply the subroutine
c   f(t,y,yp) to evaluate
c      dy(i)/dt=yp(i) = f(t,y(1),y(2),...,y(neqn))
c   and initialize the parameters:
c      neqn -- number of equations to be integrated
c      y(*) -- vector of initial conditions
c      t -- starting point of integration
c      tout -- point at which solution is desired
```

```
c      relerr,abserr -- relative and absolute local error tolerances
c      iflag -- +1,-1.  indicator to initialize the code.  normal input
c           is +1.  the user should set iflag=-1 only if it is
c           impossible to continue the integration beyond  tout .
c   all parameters except  f ,  neqn  and  tout  may be altered by the
c   code on output so must be variables in the calling program.
c
c   output from  ode  --
c
c      neqn -- unchanged
c      y(*) -- solution at  t
c      t -- last point reached in integration.  normal return has
c           t=tout .
c      tout -- unchanged
c      relerr,abserr -- normal return has tolerances unchanged.  iflag=3
c           signals tolerances increased
c      iflag = 2 -- normal return.  integration reached  tout
c            = 3 -- integration did not reach  tout  because error
c                   tolerances too small.  relerr ,  abserr  increased
c                   appropriately for continuing
c            = 4 -- integration did not reach  tout  because more than
c                   500 steps needed
c            = 5 -- integration did not reach  tout  because equations
c                   appear to be stiff
c            = 6 -- invalid input parameters (fatal error)
c           the value of  iflag  is returned negative when the input
c           value is negative and the integration does not reach  tout ,
c           i.e., -3, -4, -5.
c      work(*),iwork(*) -- information generally of no interest to the
c           user but necessary for subsequent calls.
c
c   subsequent calls to  ode --
c
c   subroutine  ode  returns with all information needed to continue
c   the integration.  if the integration reached  tout , the user need
c   only define a new  tout  and call again.  if the integration did not
c   reach  tout  and the user wants to continue, he just calls again.
c   the output value of  iflag  is the appropriate input value for
c   subsequent calls.  the only situation in which it should be altered
c   is to stop the integration internally at the new  tout , i.e.,
c   change output  iflag=2  to input  iflag=-2 .  error tolerances may
c   be changed by the user before continuing.  all other parameters must
c   remain unchanged.
c
c*********************************************************************
c*  subroutines  de  and  step  contain machine dependent constants.  *
c*  be sure they are set before using  ode .                          *
c*********************************************************************
c
      logical start,phase1,nornd
      dimension y(*),work(*),iwork(5)
      external f
      data ialpha,ibeta,isig,iv,iw,ig,iphase,ipsi,ix,ih,ixold,ihold,
     1 istart,itold,idelsn/1,13,25,38,50,62,75,76,88,89,90,91,92,93,94/
      iyy=100
      iwt=iyy+neqn
      ip=iwt+neqn
      iyp=ip+neqn
      iypout=iyp+neqn
      iphi=iypout+neqn
      if(iabs(iflag).eq.1) goto 1
      start=work(istart).gt.0.
```

```
      phase1=work(iphase).gt.0.
      nornd=iwork(2).ne.-1
     1 call de(f,neqn,y,t,tout,relerr,abserr,iflag,work(iyy),
     1 work(iwt),work(ip),work(iyp),work(iypout),work(iphi),
     2 work(ialpha),work(ibeta),work(isig),work(iv),work(iw),work(ig),
     3 phase1,work(ipsi),work(ix),work(ih),work(ixold),work(ihold),
     4 start,work(itold),work(idelsn),iwork(1),nornd,iwork(3),iwork(4),
     5 iwork(5))
      work(istart)=-1.
      if(start) work(istart)=1.
      work(iphase)=-1.
      if(phase1) work(iphase)=1.
      iwork(2)=-1
      if(nornd) iwork(2)=1
      return
      end
      subroutine de(f,neqn,y,t,tout,relerr,abserr,iflag,yy,wt,p,yp,
     1 ypout,phi,alpha,beta,sig,v,w,g,phase1,psi,x,h,xold,hold,
     2 start,told,delsgn,ns,nornd,k,kold,isnold)
c
c   ode  merely allocates storage for  de  to relieve the user of the
c   inconvenience of a long call list.  consequently  de  is used as
c   described in the comments for  ode .
c
c   this code is completely explained and documented in the text,
c   computer solution of ordinary differential equations:  the initial
c   value problem  by l. f. shampine and m. k. gordon.
c
      logical stiff,crash,start,phase1,nornd
      dimension y(*),yy(*),wt(*),phi(neqn,16),p(*),yp(*),ypout(*),
     1 psi(12),alpha(12),beta(12),sig(13),v(12),w(12),g(13)
      external f
c
c*********************************************************************
c*  the only machine dependent constant is based on the machine unit   *
c*  roundoff error  u  which is the smallest positive number such that *
c*  1.0+u.gt.1.0 .  u  must be calculated and  fouru=4.0*u  inserted *
c*  in the following data statement before using  de .  the routine    *
c*  machin  calculates  u .  fouru  and  twou=2.0*u  must also be      *
c*  inserted in subroutine  step  before calling  de .                 *
      data fouru/2.17e-19/
c*********************************************************************
c
c   the constant  maxnum  is the maximum number of steps allowed in one
c   call to  de .  the user may change this limit by altering the
c   following statement
      data maxnum/500/
c
c           ***            ***            ***
c   test for improper parameters
c
      if(neqn.lt.1) goto 10
      if(t.eq.tout) goto 10
      if(relerr.lt.0. .or. abserr.lt.0.) goto 10
      eps=amax1(relerr,abserr)
      if(eps.le.0.) goto 10
      if(iflag.eq.0) goto 10
      isn=isign(1,iflag)
      iflag=iabs(iflag)
      if(iflag.eq.1) goto 20
      if(t.ne.told) goto 10
      if(iflag.ge.2 .and. iflag.le.5) goto 20
```

```
   10 iflag=6
      return
c
c    on each call set interval of integration and counter for number of
c    steps.  adjust input error tolerances to define weight vector for
c    subroutine  step
c
   20 del=tout-t
      absdel=abs(del)
      tend=t+10.*del
      if(isn.lt.0) tend=tout
      nostep=0
      kle4=0
      stiff=.false.
      releps=relerr/eps
      abseps=abserr/eps
      if(iflag.eq.1) goto 30
      if(isnold.lt.0) goto 30
      if(delsgn*del.gt.0.) goto 50
c
c    on start and restart also set work variables x and yy(*), store the
c    direction of integration and initialize the step size
c
   30 start=.true.
      x=t
      do 40 l=1,neqn
   40 yy(l)=y(l)
      delsgn=sign(1.,del)
      h=sign(amax1(abs(tout-x),fouru*abs(x)),tout-x)
c
c    if already past output point, interpolate and return
c
   50 if(abs(x-t).lt.absdel) goto 60
      call intrp(x,yy,tout,y,ypout,neqn,kold,phi,psi,xold,p)
      iflag=2
      t=tout
      told=t
      isnold=isn
      return
c
c    if cannot go past output point and sufficiently close,
c    extrapolate and return
c
   60 if(isn.gt.0 .or. abs(tout-x).ge.fouru*abs(x)) goto 80
      h=tout-x
      call f(x,yy,yp)
      do 70 l=1,neqn
   70 y(l)=yy(l)+h*yp(l)
      iflag=2
      t=tout
      told=t
      isnold=isn
      return
c
c    test for too many steps
c
   80 if(nostep.lt.maxnum) goto 100
      iflag=isn*4
      if(stiff) iflag=isn*5
      do 90 l=1,neqn
   90 y(l)=yy(l)
      t=x
```

```
      told=t
      isnold=1
      return
c
c   limit step size, set weight vector and take a step
c
  100 h=sign(amin1(abs(h),abs(tend-x)),h)
      do 110 l=1,neqn
  110 wt(l)=releps*abs(yy(l))+abseps
      call step(x,yy,f,neqn,h,eps,wt,start,hold,k,kold,crash,phi,
     1 p,yp,psi,alpha,beta,sig,v,w,g,phase1,ns,nornd,xold)
c
c   test for tolerances too small
c
      if(.not.crash) goto 130
      iflag=isn*3
      relerr=eps*releps
      abserr=eps*abseps
      do 120 l=1,neqn
  120 y(l)=yy(l)
      t=x
      told=t
      isnold=1
      return
c
c   augment counter on number of steps and test for stiffness
c
  130 nostep=nostep+1
      kle4=kle4+1
      if(kold.gt.4) kle4=0
      if(kle4.ge.50) stiff=.true.
      goto 50
      end
      subroutine step(x,y,f,neqn,h,eps,wt,start,hold,k,kold,crash,phi,
     1 p,yp,psi,alpha,beta,sig,v,w,g,phase1,ns,nornd,xold)
c
c   subroutine  step  integrates a system of first order ordinary
c   differential equations one step, normally from x to x+h, using a
c   modified divided difference form of the adams pece formulas.  local
c   extrapolation is used to improve absolute stability and accuracy.
c   the code adjusts its order and step size to control the local error
c   per unit step in a generalized sense.  special devices are included
c   to control roundoff error and to detect when the user is requesting
c   too much accuracy.
c
c   this code is completely explained and documented in the text,
c   computer solution of ordinary differential equations:  the initial
c   value problem  by l. f. shampine and m. k. gordon.
c
c
c   the parameters represent:
c      x -- independent variable
c      y(*) -- solution vector at x
c      yp(*) -- derivative of solution vector at  x  after successful
c           step
c      neqn -- number of equations to be integrated
c      h -- appropriate step size for next step.  normally determined by
c           code
c      eps -- local error tolerance.  must be variable
c      wt(*) -- vector of weights for error criterion
c      start -- logical variable set .true. for first step,  .false.
c           otherwise
```

```
c       hold -- step size used for last successful step
c       k -- appropriate order for next step (determined by code)
c       kold -- order used for last successful step
c       crash -- logical variable set .true. when no step can be taken,
c            .false. otherwise.
c    the arrays  phi, psi  are required for the interpolation subroutine
c    intrp .  the array  p  is internal to the code.
c
c    input to  step
c
c       first call --
c
c    the user must provide storage in his driver program for all arrays
c    in the call list, namely
c
c      dimension y(neqn),wt(neqn),phi(neqn,16),p(neqn),yp(neqn),psi(12)
c
c    the user must also declare  start  and  crash  logical variables
c    and  f  an external subroutine, supply the subroutine  f(x,y,yp)
c    to evaluate
c       dy(i)/dx=yp(i) = f(x,y(1),y(2),...,y(neqn))
c    and initialize only the following parameters:
c       x -- initial value of the independent variable
c       y(*) -- vector of initial values of dependent variables
c       neqn -- number of equations to be integrated
c       h -- nominal step size indicating direction of integration
c            and maximum size of step.  must be variable
c       eps -- local error tolerance per step.  must be variable
c       wt(*) -- vector of non-zero weights for error criterion
c       start -- .true.
c
c    step  requires the l2 norm of the vector with components
c    local error(l)/wt(l)  be less than  eps  for a successful step.  the
c    array  wt  allows the user to specify an error test appropriate
c    for his problem.  for example,
c       wt(l)=1.0  specifies absolute error,
c            =abs(y(l))  error relative to the most recent value of the
c                  l-th component of the solution,
c            =abs(yp(l))  error relative to the most recent value of
c                  the l-th component of the derivative,
c            =amax1(wt(l),abs(y(l)))  error relative to the largest
c                  magnitude of l-th component obtained so far,
c            =abs(y(l))*relerr/eps+abserr/eps  specifies a mixed
c                  relative-absolute test where  relerr  is relative
c                  error,  abserr  is absolute error and  eps =
c                  amax1(relerr,abserr) .
c
c       subsequent calls --
c
c    subroutine  step  is designed so that all information needed to
c    continue the integration, including the step size  h  and the order
c    k , is returned with each step.  with the exception of the step
c    size, the error tolerance, and the weights, none of the parameters
c    should be altered.  the array  wt  must be updated after each step
c    to maintain relative error tests like those above.  normally the
c    integration is continued just beyond the desired endpoint and the
c    solution interpolated there with subroutine  intrp .  if it is
c    impossible to integrate beyond the endpoint, the step size may be
c    reduced to hit the endpoint since the code will not take a step
c    larger than the  h  input.  changing the direction of integration,
c    i.e., the sign of  h , requires the user set  start=.true. before
c    calling  step  again.  this is the only situation in which  start
```

```
c   should be altered.
c
c   output from  step
c
c      successful step --
c
c   the subroutine returns after each successful step with  start  and
c   crash  set .false. .  x  represents the independent variable
c   advanced one step of length  hold  from its value on input and  y
c   the solution vector at the new value of  x .  all other parameters
c   represent information corresponding to the new  x  needed to
c   continue the integration.
c
c      unsuccessful step --
c
c   when the error tolerance is too small for the machine precision,
c   the subroutine returns without taking a step and  crash=.true. .
c   an appropriate step size and error tolerance for continuing are
c   estimated and all other information is restored as upon input
c   before returning.  to continue with the larger tolerance, the user
c   just calls the code again.  a restart is neither required nor
c   desirable.
c
      logical start,crash,phase1,nornd
      dimension y(*),wt(*),phi(neqn,16),p(*),yp(*),psi(12),alpha(12),
     1 beta(12),sig(13),w(12),v(12),g(13),gstr(13),two(13)
c*********************************************************************
c*  the only machine dependent constants are based on the machine unit *
c*  roundoff error  u  which is the smallest positive number such that *
c*  1.0+u.gt.1.0  .  the user must calculate  u  and insert          *
c*  twou=2.0*u  and  fouru=4.0*u  in the data statement before calling *
c*  the code.  the routine  machin  calculates  u .                   *
      data twou,fouru/1.08e-19,2.17e-19/
c*********************************************************************
      data two/2.0,4.0,8.0,16.0,32.0,64.0,128.0,256.0,512.0,1024.0,
     1 2048.0,4096.0,8192.0/
      data gstr/0.500,0.0833,0.0417,0.0264,0.0188,0.0143,0.0114,0.00936,
     1 0.00789,0.00679,0.00592,0.00524,0.00468/
c     data g(1),g(2)/1.0,0.5/,sig(1)/1.0/
c
c
c       ***     begin block 0     ***
c   check if step size or error tolerance is too small for machine
c   precision.  if first step, initialize phi array and estimate a
c   starting step size.
c                  ***
c
c   if step size is too small, determine an acceptable one
c
      crash=.true.
      if(abs(h).ge.fouru*abs(x)) goto 5
      h=sign(fouru*abs(x),h)
      return
    5 p5eps=0.5*eps
c
c   if error tolerance is too small, increase it to an acceptable value
c
      round=0.
      do 10 l=1,neqn
   10 round=round+(y(l)/wt(l))**2
      round=twou*sqrt(round)
      if(p5eps.ge.round) goto 15
```

```
      eps=2.*round*(1.+fouru)
      return
   15 crash=.false.
      g(1)=1.
      g(2)=0.5
      sig(1)=1.
      if(.not.start) goto 99
c
c   initialize.  compute appropriate step size for first step
c
      call f(x,y,yp)
      sum=0.
      do 20 l=1,neqn
      phi(l,1)=yp(l)
      phi(l,2)=0.
   20 sum=sum+(yp(l)/wt(l))**2
      sum=sqrt(sum)
      absh=abs(h)
      if(eps.lt.16.0*sum*h*h) absh=0.25*sqrt(eps/sum)
      h=sign(amax1(absh,fouru*abs(x)),h)
      hold=0.
      k=1
      kold=0
      start=.false.
      phase1=.true.
      nornd=.true.
      if(p5eps.gt.100.*round) goto 99
      nornd=.false.
      do 25 l=1,neqn
   25 phi(l,15)=0.
   99 ifail=0
c       ***     end block 0     ***
c
c       ***     begin block 1     ***
c   compute coefficients of formulas for this step.  avoid computing
c   those quantities not changed when step size is not changed.
c                   ***
c
  100 kp1=k+1
      kp2=k+2
      km1=k-1
      km2=k-2
c
c   ns is the number of steps taken with size h, including the current
c  .ne. when k.lt.ns, no coefficients change
c
      if(h.ne.hold) ns=0
      if (ns.le.kold) ns=ns+1
      nsp1=ns+1
      if (k.lt.ns) goto 199
c
c   compute those components of alpha(*),beta(*),psi(*),sig(*) which
c   are changed
c
      beta(ns)=1.
      realns=ns
      alpha(ns)=1./realns
      temp1=h*realns
      sig(nsp1)=1.
      if(k.lt.nsp1) goto 110
      do 105 i=nsp1,k
      im1=i-1
```

```
      temp2=psi(im1)
      psi(im1)=temp1
      beta(i)=beta(im1)*psi(im1)/temp2
      temp1=temp2+h
      alpha(i)=h/temp1
      reali=i
  105 sig(i+1)=reali*alpha(i)*sig(i)
  110 psi(k)=temp1
c
c   compute coefficients g(*)
c
c   initialize v(*) and set w(*).  g(2) is set in data statement
c
      if(ns.gt.1) goto 120
      do 115 iq=1,k
      temp3=iq*(iq+1)
      v(iq)=1./temp3
  115 w(iq)=v(iq)
      goto 140
c
c   if order was raised, update diagonal part of v(*)
c
  120 if(k.le.kold) goto 130
      temp4=k*kp1
      v(k)=1./temp4
      nsm2=ns-2
      if(nsm2.lt.1) goto 130
      do 125 j=1,nsm2
      i=k-j
  125 v(i)=v(i)-alpha(j+1)*v(i+1)
c
c   update v(*) and set w(*)
c
  130 limit1=kp1-ns
      temp5=alpha(ns)
      do 135 iq=1,limit1
      v(iq)=v(iq)-temp5*v(iq+1)
  135 w(iq)=v(iq)
      g(nsp1)=w(1)
c
c   compute the g(*) in the work vector w(*)
c
  140 nsp2=ns+2
      if(kp1.lt.nsp2) goto 199
      do 150 i=nsp2,kp1
      limit2=kp2-i
      temp6=alpha(i-1)
      do 145 iq=1,limit2
  145 w(iq)=w(iq)-temp6*w(iq+1)
  150 g(i)=w(1)
  199 continue
c       ***     end block 1     ***
c
c       ***     begin block 2     ***
c   predict a solution p(*), evaluate derivatives using predicted
c   solution, estimate local error at order k and errors at orders k,
c   k-1, k-2 as if constant step size were used.
c                   ***
c
c   change phi to phi star
c
      if(k.lt.nsp1) goto 215
```

```
      do 210 i=nsp1,k
      temp1=beta(i)
      do 205 l=1,neqn
  205 phi(l,i)=temp1*phi(l,i)
  210 continue
c
c   predict solution and differences
c
  215 do 220 l=1,neqn
      phi(l,kp2)=phi(l,kp1)
      phi(l,kp1)=0.
  220  p(l)=0.
      do 230 j=1,k
      i=kp1-j
      ip1=i+1
      temp2=g(i)
      do 225 l=1,neqn
      p(l)=p(l)+temp2*phi(l,i)
  225 phi(l,i)=phi(l,i)+phi(l,ip1)
  230 continue
      if(nornd) goto 240
      do 235 l=1,neqn
      tau=h*p(l)-phi(l,15)
      p(l)=y(l)+tau
  235 phi(l,16)=(p(l)-y(l))-tau
      goto 250
  240 do 245 l=1,neqn
  245 p(l)=y(l)+h*p(l)
  250 xold=x
      x=x+h
      absh=abs(h)
      call f(x,p,yp)
c
c   estimate errors at orders k,k-1,k-2
c
      erkm2=0.
      erkm1=0.
      erk=0.
      do 265 l=1,neqn
      temp3=1.0/wt(l)
      temp4=yp(l)-phi(l,1)
      if(km2)265,260,255
  255 erkm2=erkm2+((phi(l,km1)+temp4)*temp3)**2
  260 erkm1=erkm1+((phi(l,k)+temp4)*temp3)**2
  265 erk=erk+(temp4*temp3)**2
      if(km2)280,275,270
  270 erkm2=absh*sig(km1)*gstr(km2)*sqrt(erkm2)
  275 erkm1=absh*sig(k)*gstr(km1)*sqrt(erkm1)
  280 temp5=absh*sqrt(erk)
      err=temp5*(g(k)-g(kp1))
      erk=temp5*sig(kp1)*gstr(k)
      knew=k
c
c   test if order should be lowered
c
      if(km2)299,290,285
  285 if(amax1(erkm1,erkm2).le.erk) knew=km1
      goto 299
  290 if(erkm1.le.0.5*erk) knew=km1
c
c   test if step successful
c
```

```
  299 if(err.le.eps) goto 400
c       ***     end block 2     ***
c
c       ***     begin block 3     ***
c   the step is unsuccessful.  restore  x, phi(*,*), psi(*) .
c   if third consecutive failure, set order to one.  if step fails more
c   than three times, consider an optimal step size.  double error
c   tolerance and return if estimated step size is too small for machine
c   precision.
c                   ***
c
c   restore x, phi(*,*) and psi(*)
c
      phase1=.false.
      x=xold
      do 310 i=1,k
      temp1=1.0/beta(i)
      ip1=i+1
      do 305 l=1,neqn
  305 phi(l,i)=temp1*(phi(l,i)-phi(l,ip1))
  310 continue
      if(k.lt.2) goto 320
      do 315 i=2,k
  315 psi(i-1)=psi(i)-h
c
c   on third failure, set order to one.  thereafter, use optimal step
c   size
c
  320 ifail=ifail+1
      temp2=0.5
      if(ifail-3) 335,330,325
  325 if(p5eps.lt.0.25*erk) temp2=sqrt(p5eps/erk)
  330 knew=1
  335 h=temp2*h
      k=knew
      if(abs(h).ge.fouru*abs(x)) goto 340
      crash=.true.
      h=sign(fouru*abs(x),h)
      eps=eps+eps
      return
  340 goto 100
c       ***     end block 3     ***
c
c       ***     begin block 4     ***
c   the step is successful.  correct the predicted solution, evaluate
c   the derivatives using the corrected solution and update the
c   differences.  determine best order and step size for next step.
c                   ***
  400 kold=k
      hold=h
c
c...  correct and evaluate
c
      temp1=h*g(kp1)
      if(nornd) goto 410
      do 405 l=1,neqn
      temp3=y(l)
      rho=temp1*(yp(l)-phi(l,1))-phi(l,16)
      y(l)=p(l)+rho
      phi(l,15)=(y(l)-p(l))-rho
  405 p(l)=temp3
      goto 420
```

```
  410 do 415 l=1,neqn
      temp3=y(l)
      y(l)=p(l)+temp1*(yp(l)-phi(l,1))
  415 p(l)=temp3
  420 call f(x,y,yp)
c
c   update differences for next step
c
      do 425 l=1,neqn
      phi(l,kp1)=yp(l)-phi(l,1)
  425 phi(l,kp2)=phi(l,kp1)-phi(l,kp2)
      do 435 i=1,k
      do 430 l=1,neqn
  430 phi(l,i)=phi(l,i)+phi(l,kp1)
  435 continue
c
c   estimate error at order k+1 unless:
c     in first phase when always raise order,
c     already decided to lower order,
c     step size not constant so estimate unreliable
c
      erkp1=0.0
      if(knew.eq.km1 .or. k.eq.12) phase1=.false.
      if(phase1) goto 450
      if(knew.eq.km1) goto 455
      if(kp1.gt.ns) goto 460
      do 440 l=1,neqn
  440 erkp1=erkp1+(phi(l,kp2)/wt(l))**2
      erkp1=absh*gstr(kp1)*sqrt(erkp1)
c
c   using estimated error at order k+1, determine appropriate order
c   for next step
c
      if(k.gt.1) goto 445
      if(erkp1.ge.0.5*erk) goto 460
      goto 450
  445 if(erkm1.le.amin1(erk,erkp1)) goto 455
      if(erkp1.ge.erk .or. k.eq.12) goto 460
c
c   here erkp1.lt.erk.lt.amax1(erkm1,erkm2) else order would have
c   been lowered in block 2.  thus order is to be raised
c
c   raise order
c
  450 k=kp1
      erk=erkp1
      goto 460
c
c   lower order
c
  455 k=km1
      erk=erkm1
c
c   with new order determine appropriate step size for next step
c
  460 hnew=h+h
      if(phase1) goto 465
      if(p5eps.ge.erk*two(k+1)) goto 465
      hnew=h
      if(p5eps.ge.erk) goto 465
      temp2=k+1
      r=(p5eps/erk)**(1.0/temp2)
```

```
      hnew=absh*amax1(0.5,amin1(0.9,r))
      hnew=sign(amax1(hnew,fouru*abs(x)),h)
  465 h=hnew
      return
c       ***     end block 4     ***
      end
      subroutine intrp(x,y,xout,yout,ypout,neqn,kold,phi,psi,ox,oy)
c
c   the methods in subroutine  step  approximate the solution near  x
c   by a polynomial.  subroutine  intrp  approximates the solution at
c   xout  by evaluating the polynomial there.  information defining this
c   polynomial is passed from  step  so  intrp  cannot be used alone.
c
c   this code is completely explained and documented in the text,
c   computer solution of ordinary differential equations:  the initial
c   value problem  by l. f. shampine and m. k. gordon.
c
c   input to intrp --
c
c   the user provides storage in the calling program for the arrays in
c   the call list
      dimension y(neqn),yout(neqn),ypout(neqn),phi(neqn,16),psi(12),
     1 oy(neqn)
c   and defines
c      xout -- point at which solution is desired.
c   the remaining parameters are defined in  step  and passed to  intrp
c   from that subroutine
c
c   output from  intrp --
c
c      yout(*) -- solution at  xout
c      ypout(*) -- derivative of solution at  xout
c   the remaining parameters are returned unaltered from their input
c   values.  integration with  step  may be continued.
c
      dimension g(13),rho(13),u(13),v(13),w(13)
c
      ki=kold+1
      kip1=ki+1
      kip2=ki+2
c
      hi=xout-x
      h=x-ox
      hioh=hi/h
c
c...  initialize w(*) for computing g(*) and u(*) for computing v(*)
c
      do 1 i=1,ki
      temp1=i*(i+1)
      w(i)=hioh/temp1
    1 u(i)=-1./temp1
c
c...  compute g(*), v(*) and rho(*)
c
      g(1)=1.
      g(2)=0.5*hioh
      v(1)=1.
      v(2)=-0.5
      rho(1)=1.
      rho(2)=hioh
      term=h
      if(ki.ge.3) then
```

```
        do 3 j=3,ki
        jm1=j-1
        psijm1=psi(jm1)
        gamma=(hi+term)/psijm1
        eta=hi/psijm1
        gamman=(-h+term)/psijm1
        etaxn=h/psijm1
        l=kip2-j
        do 2 i=1,l
        w(i)=gamma*w(i)-eta*w(i+1)
    2   u(i)=gamman*u(i)+etaxn*u(i+1)
        g(j)=w(1)
        v(j)=u(1)
        rho(j)=gamma*rho(jm1)
    3   term=psijm1
      endif
c
c...  define correction terms to yield smooth interpolant
c
      qxout=hi*((hi+term)*w(1)-hi*w(2))
      qxn=-h*((-h+term)*u(1)+h*u(2))
      qpxout=(hi+term)*rho(ki)
      qr=qxout/qxn
      qpr=qpxout/qxn
c
c...  interpolate for the solution, yout, and for the derivative of the
c...  solution, ypout
c
      do 4 l=1,neqn
      ypout(l)=0.
    4 yout(l)=0.
      do 6 j=1,ki
      i=kip1-j
      temp2=hi*g(i)+qr*h*v(i)
      temp3=rho(i)+qpr*h*v(i)
      do 5 l=1,neqn
      yout(l)=yout(l)+temp2*phi(l,i)
    5 ypout(l)=ypout(l)+temp3*phi(l,i)
    6 continue
      do 7 l=1,neqn
      yout(l)=((1.-qr)*y(l)+qr*oy(l))+yout(l)
    7 ypout(l)=ypout(l)+qpr*(oy(l)-y(l))
      return
      end
```

Appendix C
Input and Output Files from PROGRAM DROP for Benchmark Cases

PROGRAM DROP INPUT AND OUTPUT FILES FOR 769-MICRON PURE WATER DROP (FIGURES 1, 3, 4, AND 5)

INPUT FILE (WATER1.INP)

```
     768.9       6.8     500.0         1       1.0
       0.0     1000.
       1.0      18.0
   101325.     100.0 4.250E-10     1.916  2.5044E6   -2442.5      1000.       0.0
```

OUTPUT FILE (WATER1.OUT)

0.0000000E+000	6.655145	7.6890585E-04	-3.151295	2.3801725E-07	500.0000	1.010194
1.000000	6.654982	7.6515734E-04	-3.136883	2.3455313E-07	496.8559	1.010194
2.000000	6.654836	7.6140428E-04	-3.122428	2.3111861E-07	493.7262	1.010194
3.000000	6.654682	7.5764651E-04	-3.107927	2.2771354E-07	490.6111	1.010194
4.000000	6.654540	7.5388409E-04	-3.093378	2.2433791E-07	487.5104	1.010194
5.000000	6.654386	7.5011689E-04	-3.078783	2.2099161E-07	484.4243	1.010194
6.000000	6.654231	7.4634497E-04	-3.064140	2.1767458E-07	481.3528	1.010194
7.000000	6.654081	7.4256817E-04	-3.049451	2.1438673E-07	478.2961	1.010194
8.000000	6.653919	7.3878653E-04	-3.034715	2.1112798E-07	475.2540	1.010194
9.000000	6.653756	7.3499995E-04	-3.019930	2.0789825E-07	472.2266	1.010193
10.00000	6.653609	7.3120848E-04	-3.005099	2.0469749E-07	469.2141	1.010193
11.00000	6.653449	7.2741194E-04	-2.990217	2.0152558E-07	466.2164	1.010193
12.00000	6.653282	7.2361040E-04	-2.975288	1.9838247E-07	463.2337	1.010193
13.00000	6.653110	7.1980373E-04	-2.960310	1.9526806E-07	460.2659	1.010193
14.00000	6.652956	7.1599201E-04	-2.945282	1.9218228E-07	457.3131	1.010193
15.00000	6.652781	7.1217504E-04	-2.930205	1.8912507E-07	454.3753	1.010193
16.00000	6.652615	7.0835283E-04	-2.915080	1.8609632E-07	451.4527	1.010193
17.00000	6.652437	7.0452539E-04	-2.899903	1.8309596E-07	448.5452	1.010193
18.00000	6.652267	7.0069259E-04	-2.884677	1.8012392E-07	445.6529	1.010192
19.00000	6.652097	6.9685443E-04	-2.869400	1.7718011E-07	442.7758	1.010192
20.00000	6.651912	6.9301081E-04	-2.854072	1.7426446E-07	439.9141	1.010192
21.00000	6.651731	6.8916171E-04	-2.838692	1.7137687E-07	437.0677	1.010192
22.00000	6.651552	6.8530702E-04	-2.823262	1.6851726E-07	434.2368	1.010192
23.00000	6.651371	6.8144681E-04	-2.807779	1.6568558E-07	431.4212	1.010192
24.00000	6.651206	6.7758095E-04	-2.792243	1.6288172E-07	428.6212	1.010192
25.00000	6.650994	6.7370932E-04	-2.776657	1.6010559E-07	425.8368	1.010192
26.00000	6.650806	6.6983199E-04	-2.761018	1.5735712E-07	423.0679	1.010191
27.00000	6.650613	6.6594878E-04	-2.745325	1.5463624E-07	420.3148	1.010191
28.00000	6.650430	6.6205970E-04	-2.729578	1.5194284E-07	417.5773	1.010191
29.00000	6.650236	6.5816467E-04	-2.713779	1.4927684E-07	414.8556	1.010191
30.00000	6.650033	6.5426360E-04	-2.697924	1.4663817E-07	412.1497	1.010191
31.00000	6.649839	6.5035647E-04	-2.682016	1.4402674E-07	409.4598	1.010191
32.00000	6.649630	6.4644322E-04	-2.666051	1.4144247E-07	406.7857	1.010190
33.00000	6.649446	6.4252369E-04	-2.650033	1.3888526E-07	404.1277	1.010190
34.00000	6.649221	6.3859794E-04	-2.633959	1.3635504E-07	401.4857	1.010190
35.00000	6.649006	6.3466578E-04	-2.617828	1.3385171E-07	398.8598	1.010190

36.00000	6.648794	6.3072721E-04	-2.601642	1.3137519E-07	396.2500	1.010190
37.00000	6.648581	6.2678213E-04	-2.585398	1.2892539E-07	393.6565	1.010190
38.00000	6.648365	6.2283041E-04	-2.569098	1.2650223E-07	391.0793	1.010190
39.00000	6.648145	6.1887212E-04	-2.552740	1.2410561E-07	388.5183	1.010190
40.00000	6.647935	6.1490701E-04	-2.536325	1.2173544E-07	385.9738	1.010189
41.00000	6.647697	6.1093515E-04	-2.519852	1.1939166E-07	383.4457	1.010189
42.00000	6.647473	6.0695637E-04	-2.503319	1.1707416E-07	380.9341	1.010189
43.00000	6.647270	6.0297060E-04	-2.486727	1.1478285E-07	378.4391	1.010189
44.00000	6.647009	5.9897773E-04	-2.470076	1.1251765E-07	375.9606	1.010189
45.00000	6.646781	5.9497770E-04	-2.453367	1.1027846E-07	373.4989	1.010189
46.00000	6.646535	5.9097039E-04	-2.436595	1.0806519E-07	371.0539	1.010188
47.00000	6.646291	5.8695575E-04	-2.419762	1.0587775E-07	368.6257	1.010188
48.00000	6.646047	5.8293366E-04	-2.402869	1.0371605E-07	366.2144	1.010188
49.00000	6.645798	5.7890400E-04	-2.385915	1.0158001E-07	363.8200	1.010188
50.00000	6.645545	5.7486672E-04	-2.368897	9.9469524E-08	361.4426	1.010188
51.00000	6.645300	5.7082169E-04	-2.351817	9.7384515E-08	359.0822	1.010188
52.00000	6.645063	5.6676875E-04	-2.334675	9.5324879E-08	356.7390	1.010187
53.00000	6.644784	5.6270789E-04	-2.317467	9.3290524E-08	354.4129	1.010187
54.00000	6.644518	5.5863889E-04	-2.300198	9.1281350E-08	352.1041	1.010187
55.00000	6.644238	5.5456173E-04	-2.282861	8.9297274E-08	349.8125	1.010187
56.00000	6.643979	5.5047619E-04	-2.265461	8.7338194E-08	347.5384	1.010187
57.00000	6.643694	5.4638227E-04	-2.247995	8.5404025E-08	345.2816	1.010186
58.00000	6.643408	5.4227980E-04	-2.230462	8.3494662E-08	343.0424	1.010186
59.00000	6.643135	5.3816859E-04	-2.212863	8.1610018E-08	340.8207	1.010186
60.00000	6.642848	5.3404854E-04	-2.195196	7.9750002E-08	338.6167	1.010186
61.00000	6.642576	5.2991958E-04	-2.177460	7.7914507E-08	336.4304	1.010186
62.00000	6.642255	5.2578148E-04	-2.159658	7.6103440E-08	334.2618	1.010185
63.00000	6.641963	5.2163412E-04	-2.141786	7.4316702E-08	332.1111	1.010185
64.00000	6.641649	5.1747740E-04	-2.123843	7.2554194E-08	329.9782	1.010185
65.00000	6.641335	5.1331107E-04	-2.105831	7.0815815E-08	327.8634	1.010185
66.00000	6.641033	5.0913502E-04	-2.087748	6.9101482E-08	325.7666	1.010184
67.00000	6.640716	5.0494913E-04	-2.069593	6.7411079E-08	323.6879	1.010184
68.00000	6.640386	5.0075317E-04	-2.051365	6.5744523E-08	321.6274	1.010184
69.00000	6.640061	4.9654709E-04	-2.033064	6.4101712E-08	319.5852	1.010184
70.00000	6.639752	4.9233052E-04	-2.014691	6.2482542E-08	317.5613	1.010184
71.00000	6.639394	4.8810345E-04	-1.996243	6.0886919E-08	315.5558	1.010183
72.00000	6.639036	4.8386559E-04	-1.977719	5.9314729E-08	313.5688	1.010183
73.00000	6.638693	4.7961678E-04	-1.959120	5.7765877E-08	311.6004	1.010183
74.00000	6.638334	4.7535676E-04	-1.940443	5.6240264E-08	309.6506	1.010182
75.00000	6.637976	4.7108540E-04	-1.921691	5.4737782E-08	307.7196	1.010182
76.00000	6.637613	4.6680242E-04	-1.902858	5.3258333E-08	305.8073	1.010182
77.00000	6.637244	4.6250763E-04	-1.883947	5.1801816E-08	303.9139	1.010182
78.00000	6.636861	4.5820078E-04	-1.864956	5.0368122E-08	302.0394	1.010182
79.00000	6.636470	4.5388166E-04	-1.845885	4.8957151E-08	300.1840	1.010181
80.00000	6.636112	4.4954996E-04	-1.826732	4.7568797E-08	298.3477	1.010181
81.00000	6.635686	4.4520549E-04	-1.807495	4.6202956E-08	296.5305	1.010181
82.00000	6.635280	4.4084794E-04	-1.788176	4.4859526E-08	294.7327	1.010180

83.00000	6.634863	4.3647704E-04	-1.768772	4.3538392E-08	292.9542	1.010180
84.00000	6.634449	4.3209249E-04	-1.749282	4.2239460E-08	291.1952	1.010180
85.00000	6.634011	4.2769400E-04	-1.729706	4.0962615E-08	289.4557	1.010180
86.00000	6.633588	4.2328128E-04	-1.710041	3.9707754E-08	287.7358	1.010179
87.00000	6.633147	4.1885395E-04	-1.690288	3.8474766E-08	286.0356	1.010179
88.00000	6.632687	4.1441171E-04	-1.670445	3.7263547E-08	284.3553	1.010179
89.00000	6.632226	4.0995420E-04	-1.650511	3.6073988E-08	282.6948	1.010178
90.00000	6.631804	4.0548106E-04	-1.630485	3.4905980E-08	281.0543	1.010178
91.00000	6.631281	4.0099191E-04	-1.610364	3.3759413E-08	279.4339	1.010178
92.00000	6.630790	3.9648631E-04	-1.590149	3.2634180E-08	277.8336	1.010177
93.00000	6.630292	3.9196393E-04	-1.569837	3.1530170E-08	276.2536	1.010177
94.00000	6.629797	3.8742422E-04	-1.549428	3.0447268E-08	274.6939	1.010177
95.00000	6.629276	3.8286680E-04	-1.528920	2.9385371E-08	273.1548	1.010176
96.00000	6.628746	3.7829121E-04	-1.508311	2.8344363E-08	271.6361	1.010176
97.00000	6.628201	3.7369688E-04	-1.487600	2.7324132E-08	270.1382	1.010175
98.00000	6.627655	3.6908334E-04	-1.466785	2.6324569E-08	268.6610	1.010175
99.00000	6.627093	3.6445001E-04	-1.445863	2.5345559E-08	267.2047	1.010175
100.0000	6.626566	3.5979634E-04	-1.424835	2.4386988E-08	265.7693	1.010174
101.0000	6.625936	3.5512171E-04	-1.403698	2.3448747E-08	264.3550	1.010174
102.0000	6.625326	3.5042551E-04	-1.382449	2.2530719E-08	262.9619	1.010173
103.0000	6.624712	3.4570706E-04	-1.361087	2.1632792E-08	261.5902	1.010173
104.0000	6.624081	3.4096561E-04	-1.339610	2.0754849E-08	260.2398	1.010172
105.0000	6.623444	3.3620046E-04	-1.318015	1.9896778E-08	258.9110	1.010172
106.0000	6.622786	3.3141079E-04	-1.296301	1.9058460E-08	257.6039	1.010172
107.0000	6.622120	3.2659582E-04	-1.274464	1.8239783E-08	256.3185	1.010171
108.0000	6.621415	3.2175460E-04	-1.252502	1.7440628E-08	255.0550	1.010171
109.0000	6.620706	3.1688626E-04	-1.230412	1.6660881E-08	253.8135	1.010170
110.0000	6.620052	3.1198974E-04	-1.208193	1.5900424E-08	252.5942	1.010170
111.0000	6.619237	3.0706407E-04	-1.185839	1.5159145E-08	251.3972	1.010169
112.0000	6.618478	3.0210809E-04	-1.163350	1.4436926E-08	250.2226	1.010168
113.0000	6.617696	2.9712060E-04	-1.140720	1.3733651E-08	249.0706	1.010168
114.0000	6.616877	2.9210036E-04	-1.117947	1.3049202E-08	247.9412	1.010167
115.0000	6.616050	2.8704599E-04	-1.095027	1.2383461E-08	246.8348	1.010167
116.0000	6.615196	2.8195599E-04	-1.071956	1.1736311E-08	245.7513	1.010166
117.0000	6.614316	2.7682883E-04	-1.048729	1.1107633E-08	244.6910	1.010166
118.0000	6.613414	2.7166278E-04	-1.025342	1.0497307E-08	243.6539	1.010165
119.0000	6.612489	2.6645596E-04	-1.001790	9.9052144E-09	242.6404	1.010164
120.0000	6.611532	2.6120653E-04	-0.9780674	9.3312442E-09	241.6504	1.010164
121.0000	6.610523	2.5591237E-04	-0.9541696	8.7752872E-09	240.6843	1.010163
122.0000	6.609496	2.5057123E-04	-0.9300904	8.2372287E-09	239.7422	1.010162
123.0000	6.608443	2.4518062E-04	-0.9058231	7.7169533E-09	238.8243	1.010161
124.0000	6.607356	2.3973778E-04	-0.8813607	7.2143442E-09	237.9308	1.010161
125.0000	6.606210	2.3423971E-04	-0.8566957	6.7292847E-09	237.0619	1.010160
126.0000	6.605042	2.2868307E-04	-0.8318192	6.2616579E-09	236.2177	1.010159
127.0000	6.603817	2.2306415E-04	-0.8067220	5.8113452E-09	235.3985	1.010158
128.0000	6.602548	2.1737887E-04	-0.7813936	5.3782281E-09	234.6045	1.010157
129.0000	6.601237	2.1162257E-04	-0.7558221	4.9621871E-09	233.8359	1.010156

130.0000	6.599923	2.0579039E-04	-0.7299954	4.5631263E-09	233.0930	1.010155
131.0000	6.598433	1.9987709E-04	-0.7039016	4.1809614E-09	232.3761	1.010154
132.0000	6.596942	1.9387633E-04	-0.6775256	3.8155878E-09	231.6854	1.010153
133.0000	6.595368	1.8778096E-04	-0.6508495	3.4669025E-09	231.0212	1.010152
134.0000	6.593737	1.8158265E-04	-0.6238538	3.1348018E-09	230.3838	1.010151
135.0000	6.592096	1.7527258E-04	-0.5965182	2.8192197E-09	229.7738	1.010150
136.0000	6.590208	1.6884101E-04	-0.5688224	2.5201172E-09	229.1917	1.010149
137.0000	6.588306	1.6227603E-04	-0.5407402	2.2374327E-09	228.6382	1.010147
138.0000	6.586287	1.5556354E-04	-0.5122405	1.9711073E-09	228.1139	1.010146
139.0000	6.583984	1.4868654E-04	-0.4790734	1.7210825E-09	227.6196	1.010144
140.0000	6.581563	1.4162583E-04	-0.4462888	1.4873531E-09	227.1563	1.010142
141.0000	6.578897	1.3435936E-04	-0.4120204	1.2699611E-09	226.7252	1.010141
142.0000	6.576062	1.2685875E-04	-0.3771818	1.0689264E-09	226.3278	1.010139
143.0000	6.573003	1.1908786E-04	-0.3419919	8.8427821E-10	225.9656	1.010136
144.0000	6.569680	1.1099909E-04	-0.3062799	7.1605211E-10	225.6402	1.010134
145.0000	6.566115	1.0252579E-04	-0.2697105	5.6426824E-10	225.3526	1.010131
146.0000	6.561988	9.3571805E-05	-0.2319527	4.2896423E-10	225.1024	1.010129
147.0000	6.557442	8.3996121E-05	-0.1928079	3.1028707E-10	224.8900	1.010125
148.0000	6.552185	7.3603325E-05	-0.1523890	2.0877479E-10	224.7174	1.010122
149.0000	6.546090	6.1997351E-05	-0.1109179	1.2476849E-10	224.5858	1.010117
150.0000	6.538863	4.8356909E-05	-6.8909414E-02	5.9205328E-11	224.4961	1.010112
151.0000	6.529458	3.0148300E-05	-2.7185814E-02	1.4347392E-11	224.4482	1.010106
151.7466	23.00000	0.0000000E+000	0.0000000E+000	0.0000000E+000	224.4393	1.010106

PROGRAM DROP INPUT AND OUTPUT FILES FOR 557-MICRON PURE WATER DROP (FIGURE 2)

INPUT FILE (WATER2.INP)

```
     557.0       7.4     500.0         1       1.0
       0.0     1000.
       1.0      18.0
   101325.     100.0 4.250E-10     1.916  2.5044E6   -2442.5     1000.       0.0
```

OUTPUT FILE (WATER2.OUT)

0.0000000E+000	7.263837	5.5700442E-04	-2.289084	9.0482459E-08	500.0000	1.010568
1.000000	7.263520	5.5274263E-04	-2.270979	8.8421409E-08	497.7199	1.010568
2.000000	7.263191	5.4847175E-04	-2.252805	8.6387580E-08	495.4581	1.010568
3.000000	7.262879	5.4419157E-04	-2.234556	8.4380858E-08	493.2144	1.010568
4.000000	7.262534	5.3990196E-04	-2.216236	8.2401137E-08	490.9890	1.010567
5.000000	7.262192	5.3560280E-04	-2.197842	8.0448309E-08	488.7819	1.010567
6.000000	7.261869	5.3129386E-04	-2.179376	7.8522270E-08	486.5933	1.010567
7.000000	7.261513	5.2697508E-04	-2.160834	7.6622911E-08	484.4232	1.010567
8.000000	7.261169	5.2264618E-04	-2.142219	7.4750112E-08	482.2716	1.010566
9.000000	7.260808	5.1830715E-04	-2.123526	7.2903774E-08	480.1388	1.010566
10.00000	7.260450	5.1395770E-04	-2.104759	7.1083775E-08	478.0246	1.010566
11.00000	7.260089	5.0959765E-04	-2.085913	6.9290003E-08	475.9293	1.010566
12.00000	7.259723	5.0522678E-04	-2.066989	6.7522350E-08	473.8528	1.010565
13.00000	7.259328	5.0084502E-04	-2.047986	6.5780704E-08	471.7953	1.010565
14.00000	7.258943	4.9645209E-04	-2.028905	6.4064949E-08	469.7569	1.010565
15.00000	7.258559	4.9204787E-04	-2.009743	6.2374973E-08	467.7375	1.010564
16.00000	7.258171	4.8763200E-04	-1.990499	6.0710661E-08	465.7374	1.010564
17.00000	7.257787	4.8320441E-04	-1.971174	5.9071905E-08	463.7566	1.010564
18.00000	7.257353	4.7876479E-04	-1.951767	5.7458585E-08	461.7951	1.010564
19.00000	7.256933	4.7431292E-04	-1.932275	5.5870583E-08	459.8531	1.010563
20.00000	7.256513	4.6984860E-04	-1.912700	5.4307783E-08	457.9305	1.010563
21.00000	7.256085	4.6537150E-04	-1.893038	5.2770069E-08	456.0277	1.010563
22.00000	7.255647	4.6088145E-04	-1.873290	5.1257324E-08	454.1445	1.010562
23.00000	7.255195	4.5637810E-04	-1.853455	4.9769429E-08	452.2811	1.010562
24.00000	7.254745	4.5186124E-04	-1.833531	4.8306269E-08	450.4376	1.010562
25.00000	7.254286	4.4733053E-04	-1.813517	4.6867722E-08	448.6141	1.010561
26.00000	7.253816	4.4278573E-04	-1.793414	4.5453678E-08	446.8106	1.010561
27.00000	7.253361	4.3822647E-04	-1.773218	4.4064009E-08	445.0273	1.010561
28.00000	7.252845	4.3365243E-04	-1.752930	4.2698591E-08	443.2642	1.010560
29.00000	7.252350	4.2906328E-04	-1.732548	4.1357303E-08	441.5215	1.010560
30.00000	7.251839	4.2445865E-04	-1.712071	4.0040025E-08	439.7991	1.010560
31.00000	7.251328	4.1983821E-04	-1.691497	3.8746634E-08	438.0973	1.010559
32.00000	7.250804	4.1520153E-04	-1.670825	3.7477012E-08	436.4162	1.010559
33.00000	7.250267	4.1054827E-04	-1.650054	3.6231036E-08	434.7557	1.010558
34.00000	7.249710	4.0587797E-04	-1.629184	3.5008583E-08	433.1161	1.010558
35.00000	7.249165	4.0119025E-04	-1.608211	3.3809531E-08	431.4973	1.010558

36.00000	7.248631	3.9648463E-04	-1.587135	3.2633757E-08	429.8997	1.010557
37.00000	7.248006	3.9176061E-04	-1.565953	3.1481129E-08	428.3231	1.010557
38.00000	7.247405	3.8701767E-04	-1.544666	3.0351515E-08	426.7678	1.010556
39.00000	7.246803	3.8225533E-04	-1.523270	2.9244797E-08	425.2338	1.010556
40.00000	7.246191	3.7747298E-04	-1.501764	2.8160841E-08	423.7213	1.010556
41.00000	7.245554	3.7267013E-04	-1.480147	2.7099526E-08	422.2303	1.010555
42.00000	7.244911	3.6784614E-04	-1.458415	2.6060727E-08	420.7610	1.010555
43.00000	7.244249	3.6300038E-04	-1.436568	2.5044317E-08	419.3135	1.010554
44.00000	7.243560	3.5813224E-04	-1.414604	2.4050170E-08	417.8879	1.010554
45.00000	7.242939	3.5324099E-04	-1.392521	2.3078162E-08	416.4843	1.010553
46.00000	7.242158	3.4832588E-04	-1.370315	2.2128150E-08	415.1029	1.010553
47.00000	7.241442	3.4338608E-04	-1.347984	2.1200004E-08	413.7437	1.010552
48.00000	7.240691	3.3842080E-04	-1.325527	2.0293593E-08	412.4070	1.010551
49.00000	7.239935	3.3342911E-04	-1.302941	1.9408784E-08	411.0927	1.010551
50.00000	7.239160	3.2841013E-04	-1.280222	1.8545451E-08	409.8011	1.010550
51.00000	7.238351	3.2336285E-04	-1.257368	1.7703462E-08	408.5323	1.010550
52.00000	7.237538	3.1828630E-04	-1.234376	1.6882687E-08	407.2865	1.010549
53.00000	7.236684	3.1317933E-04	-1.211243	1.6083000E-08	406.0636	1.010548
54.00000	7.235899	3.0804085E-04	-1.187966	1.5304270E-08	404.8640	1.010548
55.00000	7.234932	3.0286951E-04	-1.164541	1.4546361E-08	403.6877	1.010547
56.00000	7.234022	2.9766400E-04	-1.140964	1.3809137E-08	402.5350	1.010547
57.00000	7.233084	2.9242283E-04	-1.117231	1.3092466E-08	401.4059	1.010546
58.00000	7.232104	2.8714447E-04	-1.093338	1.2396212E-08	400.3006	1.010545
59.00000	7.231102	2.8182726E-04	-1.069281	1.1720243E-08	399.2194	1.010544
60.00000	7.230077	2.7646942E-04	-1.045054	1.1064425E-08	398.1622	1.010544
61.00000	7.229023	2.7106894E-04	-1.020652	1.0428622E-08	397.1294	1.010543
62.00000	7.227919	2.6562379E-04	-0.9960697	9.8126973E-09	396.1210	1.010542
63.00000	7.226802	2.6013161E-04	-0.9713008	9.2165173E-09	395.1373	1.010541
64.00000	7.225629	2.5459001E-04	-0.9463386	8.6399545E-09	394.1785	1.010540
65.00000	7.224423	2.4899640E-04	-0.9211771	8.0828881E-09	393.2448	1.010540
66.00000	7.223164	2.4334785E-04	-0.8958085	7.5451858E-09	392.3363	1.010539
67.00000	7.221861	2.3764119E-04	-0.8702244	7.0267183E-09	391.4533	1.010538
68.00000	7.220523	2.3187287E-04	-0.8444161	6.5273533E-09	390.5961	1.010537
69.00000	7.219128	2.2603894E-04	-0.8183735	6.0469598E-09	389.7648	1.010536
70.00000	7.217680	2.2013501E-04	-0.7920852	5.5854050E-09	388.9596	1.010535
71.00000	7.216160	2.1415616E-04	-0.7655391	5.1425548E-09	388.1809	1.010534
72.00000	7.214585	2.0809678E-04	-0.7387205	4.7182751E-09	387.4288	1.010532
73.00000	7.212991	2.0195090E-04	-0.7116154	4.3124539E-09	386.7036	1.010531
74.00000	7.211222	1.9571201E-04	-0.6842086	3.9249990E-09	386.0057	1.010530
75.00000	7.209423	1.8937229E-04	-0.6564812	3.5557919E-09	385.3354	1.010529
76.00000	7.207530	1.8292258E-04	-0.6284109	3.2047118E-09	384.6929	1.010527
77.00000	7.205616	1.7635257E-04	-0.5999734	2.8716549E-09	384.0787	1.010526
78.00000	7.203433	1.6965072E-04	-0.5711432	2.5565488E-09	383.4932	1.010524
79.00000	7.201228	1.6280301E-04	-0.5418875	2.2593016E-09	382.9367	1.010523
80.00000	7.198876	1.5579247E-04	-0.5121674	1.9798221E-09	382.4097	1.010521
81.00000	7.196223	1.4860081E-04	-0.4776652	1.7181072E-09	381.9133	1.010519
82.00000	7.193324	1.4121323E-04	-0.4433605	1.4743915E-09	381.4511	1.010517

83.00000	7.190272	1.3360678E-04	-0.4075763	1.2487402E-09	381.0256	1.010515
84.00000	7.186866	1.2574432E-04	-0.3712409	1.0410022E-09	380.6366	1.010512
85.00000	7.183314	1.1757873E-04	-0.3345049	8.5108459E-10	380.2838	1.010510
86.00000	7.179297	1.0905029E-04	-0.2971171	6.7899536E-10	379.9679	1.010507
87.00000	7.175014	1.0007482E-04	-0.2586785	5.2475990E-10	379.6900	1.010504
88.00000	7.170024	9.0538226E-05	-0.2188727	3.8858142E-10	379.4512	1.010500
89.00000	7.164552	8.0255806E-05	-0.1775373	2.7065453E-10	379.2529	1.010496
90.00000	7.157957	6.8937872E-05	-0.1348431	1.7153758E-10	379.0966	1.010491
91.00000	7.150372	5.5991572E-05	-9.1181450E-02	9.1908023E-11	378.9836	1.010486
92.00000	7.140914	4.0029230E-05	-4.7495428E-02	3.3582782E-11	378.9146	1.010479
93.00000	7.125828	9.3550680E-06	-2.6230845E-03	4.2867074E-13	378.8879	1.010468
93.08334	24.30000	0.0000000E+000	0.0000000E+000	0.0000000E+000	378.8878	1.010468

PROGRAM DROP INPUT AND OUTPUT FILES FOR LAW (1976) CASE I (FIGURES 10-13)

INPUT FILE (LAW1.INP)

```
     500.0      -8.25      500.0         2        0.1
       0.0      1000.
    0.6935       86.0
    101325.      68.85 3.886E-10     1.910   3.765E5    -606.7     676.2    -0.874
    0.3065      114.0
    101325.    125.85 3.886E-10     1.910   3.809E5    -636.0     718.2    -0.852
```

OUTPUT FILE (LAW1.OUT)

```
 0.0000000E+000  -8.251300      5.0000375E-04  -1.567121      3.1577752E-08  1.3956138E-08  500.0000       1.033148
  0.1000000      -8.076053      4.9208099E-04  -1.540863      2.9524035E-08  1.3904826E-08  499.8446       1.032970
  0.2000000      -7.886409      4.8415866E-04  -1.514558      2.7538629E-08  1.3851722E-08  499.6919       1.032778
  0.3000000      -7.680762      4.7623951E-04  -1.488216      2.5621222E-08  1.3796676E-08  499.5417       1.032570
  0.4000000      -7.457317      4.6832653E-04  -1.461848      2.3771527E-08  1.3739519E-08  499.3942       1.032343
  0.5000000      -7.214056      4.6042324E-04  -1.435469      2.1989305E-08  1.3680059E-08  499.2494       1.032096
  0.6000000      -6.948630      4.5253371E-04  -1.409092      2.0274348E-08  1.3618084E-08  499.1071       1.031825
  0.7000000      -6.658408      4.4466282E-04  -1.382738      1.8626526E-08  1.3553336E-08  498.9675       1.031529
  0.8000001      -6.340357      4.3681604E-04  -1.356424      1.7045751E-08  1.3485533E-08  498.8306       1.031204
  0.9000001      -5.990994      4.2899972E-04  -1.330174      1.5531961E-08  1.3414359E-08  498.6962       1.030847
   1.000000      -5.606431      4.2122148E-04  -1.304014      1.4085286E-08  1.3339413E-08  498.5645       1.030452
   1.100000      -5.182162      4.1348999E-04  -1.277973      1.2705865E-08  1.3260262E-08  498.4354       1.030015
   1.200000      -4.713038      4.0581531E-04  -1.252085      1.1393960E-08  1.3176399E-08  498.3089       1.029531
   1.300000      -4.193386      3.9820952E-04  -1.226389      1.0150066E-08  1.3087185E-08  498.1850       1.028993
   1.400000      -3.616734      3.9068604E-04  -1.200928      8.9746912E-09  1.2991937E-08  498.0636       1.028394
   1.500000      -2.976168      3.8326104E-04  -1.175754      7.8687048E-09  1.2889764E-08  497.9448       1.027726
   1.600000      -2.264114      3.7595263E-04  -1.150921      6.8330204E-09  1.2779694E-08  497.8285       1.026980
   1.700000      -1.473060      3.6878197E-04  -1.126496      5.8689391E-09  1.2660495E-08  497.7146       1.026146
   1.800000     -0.5958471      3.6177252E-04  -1.102548      4.9778706E-09  1.2530787E-08  497.6031       1.025216
   1.900000      0.3730052      3.5495058E-04  -1.079158      4.1615631E-09  1.2388893E-08  497.4941       1.024181
   2.000000       1.435860      3.4834413E-04  -1.056409      3.4218330E-09  1.2232936E-08  497.3873       1.023037
   2.100000       2.589277      3.4198203E-04  -1.034391      2.7605760E-09  1.2060791E-08  497.2827       1.021784
   2.200000       3.821201      3.3589144E-04  -1.013188      2.1793591E-09  1.1870234E-08  497.1804       1.020432
   2.300000       5.106899      3.3009567E-04 -0.9928812      1.6792188E-09  1.1659047E-08  497.0801       1.019006
   2.400000       6.407962      3.2460821E-04 -0.9735255      1.2596713E-09  1.1425488E-08  496.9818       1.017544
   2.500000       7.672989      3.1942973E-04 -0.9551448      9.1826380E-10  1.1168630E-08  496.8854       1.016106
   2.600000       8.844561      3.1454518E-04 -0.9377197      6.4994826E-10  1.0888836E-08  496.7907       1.014757
   2.700000       9.871565      3.0992346E-04 -0.9211788      4.4686244E-10  1.0588074E-08  496.6978       1.013561
   2.799999       10.72084      3.0552203E-04 -0.9054083      2.9901956E-10  1.0269747E-08  496.6065       1.012562
   2.899999       11.38593      3.0129196E-04 -0.8902612      1.9513921E-10  9.9383639E-09  496.5167       1.011773
   2.999999       11.88050      2.9718713E-04 -0.8755886      1.2459703E-10  9.5985211E-09  496.4284       1.011182
   3.099999       12.23274      2.9316719E-04 -0.8612520      7.8068607E-11  9.2544736E-09  496.3416       1.010760
   3.199999       12.47493      2.8919979E-04 -0.8471355      4.8113381E-11  8.9097325E-09  496.2562       1.010469
   3.299999       12.63691      2.8526055E-04 -0.8331484      2.9208441E-11  8.5669516E-09  496.1722       1.010273
```

3.399999	12.74301	2.8133192E-04	-0.8192248	1.7470447E-11	8.2280129E-09	496.0896	1.010146
3.499999	12.81123	2.7740162E-04	-0.8053181	1.0289163E-11	7.8941849E-09	496.0083	1.010063
3.599999	12.85435	2.7346128E-04	-0.7913966	5.9677159E-12	7.5662854E-09	495.9285	1.010011
3.699999	12.88054	2.6950528E-04	-0.7774395	3.4607920E-12	7.2448088E-09	495.8500	1.009980
3.799999	12.89677	2.6552897E-04	-0.7634302	1.9842090E-12	6.9301178E-09	495.7730	1.009961
3.899998	12.90687	2.6152929E-04	-0.7493581	1.1143187E-12	6.6223960E-09	495.6974	1.009949
3.999998	12.91303	2.5750409E-04	-0.7352171	6.1521632E-13	6.3217316E-09	495.6231	1.009942
4.099998	12.91662	2.5345155E-04	-0.7210016	3.4303975E-13	6.0281664E-09	495.5503	1.009938
4.199998	12.91888	2.4936991E-04	-0.7067069	1.8553689E-13	5.7417280E-09	495.4789	1.009935
4.299998	12.92140	2.4525714E-04	-0.6923279	2.6255392E-14	5.4624429E-09	495.4090	1.009933
4.309507	12.92183	2.4486889E-04	-0.6909716	0.0000000E+000	5.4365676E-09	495.4025	1.009932
4.409507	12.92207	2.4072177E-04	-0.6765018	0.0000000E+000	5.1649938E-09	495.3341	1.009932
4.509507	12.92224	2.3654071E-04	-0.6619425	0.0000000E+000	4.9005111E-09	495.2672	1.009932
4.609507	12.92246	2.3232427E-04	-0.6472901	0.0000000E+000	4.6430908E-09	495.2018	1.009932
4.709507	12.92270	2.2807071E-04	-0.6325421	0.0000000E+000	4.3927040E-09	495.1378	1.009933
4.809507	12.92292	2.2377830E-04	-0.6176950	0.0000000E+000	4.1493222E-09	495.0753	1.009933
4.909507	12.92317	2.1944511E-04	-0.6027450	0.0000000E+000	3.9129180E-09	495.0142	1.009933
5.009507	12.92341	2.1506901E-04	-0.5876881	0.0000000E+000	3.6834640E-09	494.9547	1.009933
5.109507	12.92367	2.1064778E-04	-0.5725206	0.0000000E+000	3.4609344E-09	494.8967	1.009933
5.209507	12.92392	2.0617890E-04	-0.5572373	0.0000000E+000	3.2453027E-09	494.8402	1.009933
5.309506	12.92419	2.0165967E-04	-0.5418338	0.0000000E+000	3.0365437E-09	494.7852	1.009933
5.409506	12.92446	1.9708715E-04	-0.5263041	0.0000000E+000	2.8346352E-09	494.7318	1.009934
5.509506	12.92475	1.9245812E-04	-0.5106429	0.0000000E+000	2.6395550E-09	494.6800	1.009934
5.609506	12.92505	1.8776891E-04	-0.4948436	0.0000000E+000	2.4512801E-09	494.6297	1.009934
5.709506	12.92536	1.8301549E-04	-0.4788986	0.0000000E+000	2.2697879E-09	494.5811	1.009934
5.809506	12.92567	1.7819372E-04	-0.4628012	0.0000000E+000	2.0950703E-09	494.5340	1.009934
5.909506	12.92601	1.7330049E-04	-0.4465489	0.0000000E+000	1.9271729E-09	494.4886	1.009935
6.009506	12.92638	1.6833159E-04	-0.4266279	0.0000000E+000	1.7661107E-09	494.4450	1.009935
6.109506	12.92676	1.6328139E-04	-0.4084686	0.0000000E+000	1.6118732E-09	494.4033	1.009935
6.209506	12.92716	1.5814330E-04	-0.3895163	0.0000000E+000	1.4644445E-09	494.3634	1.009935
6.309505	12.92758	1.5291001E-04	-0.3701281	0.0000000E+000	1.3238172E-09	494.3254	1.009936
6.409505	12.92803	1.4757349E-04	-0.3505128	0.0000000E+000	1.1899945E-09	494.2894	1.009936
6.509505	12.92849	1.4212415E-04	-0.3307663	0.0000000E+000	1.0629756E-09	494.2553	1.009936
6.609505	12.92897	1.3654974E-04	-0.3109008	0.0000000E+000	9.4274011E-10	494.2232	1.009936
6.709505	12.92948	1.3083537E-04	-0.2908739	0.0000000E+000	8.2926732E-10	494.1932	1.009937
6.809505	12.93002	1.2496361E-04	-0.2706136	0.0000000E+000	7.2255252E-10	494.1651	1.009937
6.909505	12.93059	1.1891391E-04	-0.2500382	0.0000000E+000	6.2261024E-10	494.1390	1.009938
7.009505	12.93120	1.1266065E-04	-0.2290687	0.0000000E+000	5.2946192E-10	494.1151	1.009938
7.109505	12.93185	1.0617238E-04	-0.2076437	0.0000000E+000	4.4315143E-10	494.0932	1.009938
7.209505	12.93256	9.9406905E-05	-0.1857214	0.0000000E+000	3.6371958E-10	494.0735	1.009939
7.309505	12.93333	9.2309609E-05	-0.1632953	0.0000000E+000	2.9124417E-10	494.0561	1.009939
7.409504	12.93417	8.4804895E-05	-0.1403943	0.0000000E+000	2.2582854E-10	494.0409	1.009940
7.509504	12.93509	7.6782359E-05	-0.1170833	0.0000000E+000	1.6760993E-10	494.0280	1.009940
7.609504	12.93611	6.8075242E-05	-9.3478143E-02	0.0000000E+000	1.1681060E-10	494.0175	1.009941
7.709504	12.93726	5.8386795E-05	-6.9707058E-02	0.0000000E+000	7.3698353E-11	494.0094	1.009942
7.809504	12.93859	4.7150021E-05	-4.5974568E-02	0.0000000E+000	3.8811329E-11	494.0036	1.009943
7.909504	12.94019	3.2918575E-05	-2.2599753E-02	0.0000000E+000	1.3207926E-11	494.0002	1.009943
8.008595	26.85000	0.0000000E+000	0.0000000E+000	0.0000000E+000	0.0000000E+000	493.9990	1.009943

PROGRAM DROP INPUT AND OUTPUT FILES FOR LAW (1976) CASE IV (FIGURES 10-13)

INPUT FILE (LAW4.INP)

```
     500.0      2.05     500.0         2       0.1
       0.0     1000.
    0.2009      86.0
    101325.    68.85 3.886E-10     1.910   3.765E5    -606.7     676.2    -0.874
    0.7991     114.0
    101325.   125.85 3.886E-10     1.910   3.809E5    -636.0     718.2    -0.852
```

OUTPUT FILE (LAW4.OUT)

0.0000000E+000	2.048722	5.0000381E-04	-1.589637	9.3040313E-09	3.7007720E-08	500.0000	1.022365
0.1000000	2.768480	4.9447088E-04	-1.571122	8.1288896E-09	3.6677115E-08	499.8420	1.021580
0.2000000	3.515777	4.8909074E-04	-1.553025	7.0453403E-09	3.6325638E-08	499.6858	1.020761
0.3000000	4.285227	4.8387065E-04	-1.535373	6.0538214E-09	3.5951832E-08	499.5314	1.019911
0.4000000	5.069594	4.7881616E-04	-1.518186	5.1542872E-09	3.5554383E-08	499.3787	1.019039
0.5000000	5.859643	4.7393073E-04	-1.501480	4.3460386E-09	3.5132199E-08	499.2277	1.018154
0.6000000	6.644330	4.6921533E-04	-1.485264	3.6275869E-09	3.4684518E-08	499.0784	1.017268
0.7000000	7.411394	4.6466789E-04	-1.469540	2.9964071E-09	3.4211059E-08	498.9306	1.016396
0.8000001	8.147944	4.6028304E-04	-1.454299	2.4488731E-09	3.3712109E-08	498.7845	1.015552
0.9000001	8.841610	4.5605231E-04	-1.439525	1.9802096E-09	3.3188591E-08	498.6398	1.014752
1.000000	9.481638	4.5196430E-04	-1.425193	1.5845659E-09	3.2642102E-08	498.4966	1.014008
1.100000	10.05985	4.4800539E-04	-1.411267	1.2552038E-09	3.2074841E-08	498.3547	1.013332
1.200000	10.57129	4.4416045E-04	-1.397707	9.8478459E-10	3.1489485E-08	498.2143	1.012730
1.300000	11.01452	4.4041380E-04	-1.384467	7.6569273E-10	3.0889066E-08	498.0752	1.012206
1.400000	11.39135	4.3674972E-04	-1.371502	5.9038463E-10	3.0276741E-08	497.9374	1.011758
1.500000	11.70612	4.3315350E-04	-1.358765	4.5170581E-10	2.9655643E-08	497.8009	1.011383
1.600000	11.96481	4.2961177E-04	-1.346215	3.4318667E-10	2.9028687E-08	497.6656	1.011073
1.700000	12.17454	4.2611259E-04	-1.333809	2.5901781E-10	2.8398553E-08	497.5316	1.010822
1.800000	12.34225	4.2264600E-04	-1.321516	1.9436763E-10	2.7767506E-08	497.3989	1.010620
1.900000	12.47513	4.1920310E-04	-1.309304	1.4502079E-10	2.7137542E-08	497.2673	1.010460
2.000000	12.57924	4.1577718E-04	-1.297150	1.0768809E-10	2.6510234E-08	497.1370	1.010335
2.100000	12.66013	4.1236216E-04	-1.285032	7.9574951E-11	2.5886921E-08	497.0079	1.010237
2.200000	12.72261	4.0895349E-04	-1.272934	5.8545405E-11	2.5268628E-08	496.8800	1.010162
2.300000	12.77050	4.0554735E-04	-1.260842	4.2895736E-11	2.4656170E-08	496.7533	1.010104
2.400000	12.80699	4.0214078E-04	-1.248744	3.1302846E-11	2.4050172E-08	496.6278	1.010060
2.500000	12.83472	3.9873138E-04	-1.236634	2.2753100E-11	2.3451111E-08	496.5036	1.010027
2.600000	12.85569	3.9531727E-04	-1.224504	1.6473945E-11	2.2859343E-08	496.3805	1.010002
2.700000	12.87145	3.9189696E-04	-1.212348	1.1884519E-11	2.2275128E-08	496.2587	1.009983
2.799999	12.88330	3.8846923E-04	-1.200162	8.5335176E-12	2.1698661E-08	496.1380	1.009969
2.899999	12.89219	3.8503311E-04	-1.187943	6.0982647E-12	2.1130072E-08	496.0186	1.009958
2.999999	12.89882	3.8158786E-04	-1.175687	4.3369843E-12	2.0569454E-08	495.9005	1.009950
3.099999	12.90383	3.7813271E-04	-1.163394	3.0497078E-12	2.0016873E-08	495.7835	1.009944
3.199999	12.90765	3.7466711E-04	-1.151061	2.1061488E-12	1.9472365E-08	495.6678	1.009940
3.299999	12.91034	3.7119084E-04	-1.138687	1.4688530E-12	1.8935918E-08	495.5533	1.009937

3.399999	12.91238	3.6770318E-04	-1.126270	1.0035537E-12	1.8407565E-08	495.4400	1.009935
3.499999	12.91397	3.6420379E-04	-1.113809	6.6104002E-13	1.7887302E-08	495.3280	1.009933
3.599999	12.91502	3.6069244E-04	-1.101304	4.4904182E-13	1.7375093E-08	495.2173	1.009932
3.699999	12.91594	3.5716858E-04	-1.088753	2.7431350E-13	1.6870954E-08	495.1078	1.009931
3.799999	12.91660	3.5363194E-04	-1.076156	1.5770305E-13	1.6374843E-08	494.9995	1.009930
3.899998	12.91708	3.5008223E-04	-1.063511	8.7695564E-14	1.5886732E-08	494.8925	1.009930
3.995190	12.91765	3.4669193E-04	-1.051434	0.0000000E+000	1.5429711E-08	494.7919	1.009929
4.095190	12.91772	3.4311571E-04	-1.038695	0.0000000E+000	1.4957132E-08	494.6874	1.009929
4.195189	12.91793	3.3952534E-04	-1.025906	0.0000000E+000	1.4492488E-08	494.5842	1.009930
4.295189	12.91801	3.3592034E-04	-1.013067	0.0000000E+000	1.4035739E-08	494.4822	1.009930
4.395189	12.91815	3.3230043E-04	-1.000176	0.0000000E+000	1.3586853E-08	494.3816	1.009930
4.495189	12.91828	3.2866516E-04	-0.9872327	0.0000000E+000	1.3145804E-08	494.2822	1.009930
4.595189	12.91841	3.2501420E-04	-0.9742362	0.0000000E+000	1.2712561E-08	494.1841	1.009930
4.695189	12.91855	3.2134706E-04	-0.9611859	0.0000000E+000	1.2287088E-08	494.0874	1.009930
4.795189	12.91868	3.1766330E-04	-0.9480802	0.0000000E+000	1.1869354E-08	493.9919	1.009930
4.895189	12.91882	3.1396252E-04	-0.9349188	0.0000000E+000	1.1459329E-08	493.8978	1.009930
4.995189	12.91896	3.1024413E-04	-0.9216998	0.0000000E+000	1.1056977E-08	493.8049	1.009930
5.095189	12.91911	3.0650766E-04	-0.9084228	0.0000000E+000	1.0662268E-08	493.7134	1.009930
5.195189	12.91926	3.0275257E-04	-0.8950867	0.0000000E+000	1.0275171E-08	493.6233	1.009930
5.295188	12.91940	2.9897830E-04	-0.8816901	0.0000000E+000	9.8956532E-09	493.5344	1.009930
5.395188	12.91956	2.9518426E-04	-0.8682317	0.0000000E+000	9.5236841E-09	493.4469	1.009931
5.495188	12.91971	2.9136983E-04	-0.8547104	0.0000000E+000	9.1592298E-09	493.3608	1.009931
5.595188	12.91988	2.8753432E-04	-0.8411244	0.0000000E+000	8.8022594E-09	493.2760	1.009931
5.695188	12.92004	2.8367704E-04	-0.8274729	0.0000000E+000	8.4527416E-09	493.1925	1.009931
5.795188	12.92020	2.7979727E-04	-0.8137543	0.0000000E+000	8.1106428E-09	493.1105	1.009931
5.895188	12.92038	2.7589418E-04	-0.7999665	0.0000000E+000	7.7759328E-09	493.0298	1.009931
5.995188	12.92054	2.7196706E-04	-0.7861082	0.0000000E+000	7.4485813E-09	492.9505	1.009931
6.095188	12.92073	2.6801493E-04	-0.7721779	0.0000000E+000	7.1285569E-09	492.8726	1.009931
6.195188	12.92091	2.6403696E-04	-0.7581735	0.0000000E+000	6.8158292E-09	492.7961	1.009931
6.295187	12.92110	2.6003216E-04	-0.7440929	0.0000000E+000	6.5103669E-09	492.7209	1.009932
6.395187	12.92128	2.5599942E-04	-0.7299346	0.0000000E+000	6.2121392E-09	492.6472	1.009932
6.495187	12.92148	2.5193769E-04	-0.7156958	0.0000000E+000	5.9211147E-09	492.5750	1.009932
6.595187	12.92168	2.4784572E-04	-0.7013747	0.0000000E+000	5.6372618E-09	492.5041	1.009932
6.695187	12.92188	2.4372224E-04	-0.6869681	0.0000000E+000	5.3605498E-09	492.4347	1.009932
6.795187	12.92209	2.3956588E-04	-0.6724740	0.0000000E+000	5.0909472E-09	492.3667	1.009932
6.895187	12.92231	2.3537522E-04	-0.6578890	0.0000000E+000	4.8284283E-09	492.3002	1.009932
6.995187	12.92253	2.3114879E-04	-0.6432112	0.0000000E+000	4.5729696E-09	492.2352	1.009932
7.095187	12.92275	2.2688492E-04	-0.6284368	0.0000000E+000	4.3245425E-09	492.1716	1.009933
7.195187	12.92299	2.2258176E-04	-0.6135628	0.0000000E+000	4.0831174E-09	492.1095	1.009933
7.295187	12.92324	2.1823736E-04	-0.5985851	0.0000000E+000	3.8486663E-09	492.0489	1.009933
7.395186	12.92347	2.1384953E-04	-0.5835000	0.0000000E+000	3.6211600E-09	491.9898	1.009933
7.495186	12.92374	2.0941590E-04	-0.5683028	0.0000000E+000	3.4005689E-09	491.9322	1.009933
7.595186	12.92399	2.0493382E-04	-0.5529882	0.0000000E+000	3.1868637E-09	491.8762	1.009933
7.695186	12.92427	2.0040043E-04	-0.5375512	0.0000000E+000	2.9800142E-09	491.8216	1.009934
7.795186	12.92455	1.9581262E-04	-0.5219859	0.0000000E+000	2.7799965E-09	491.7686	1.009934
7.895186	12.92483	1.9116739E-04	-0.5062873	0.0000000E+000	2.5868032E-09	491.7172	1.009934
7.995186	12.92513	1.8646098E-04	-0.4904489	0.0000000E+000	2.4004114E-09	491.6674	1.009934

8.095186	12.92544	1.8168915E-04	-0.4744628	0.0000000E+000	2.2207958E-09	491.6191	1.009934
8.195187	12.92576	1.7684739E-04	-0.4583210	0.0000000E+000	2.0479407E-09	491.5725	1.009934
8.295187	12.92614	1.7193242E-04	-0.4389637	0.0000000E+000	1.8818911E-09	491.5276	1.009935
8.395187	12.92648	1.6693986E-04	-0.4217003	0.0000000E+000	1.7226660E-09	491.4844	1.009935
8.495188	12.92687	1.6186449E-04	-0.4032697	0.0000000E+000	1.5702740E-09	491.4431	1.009935
8.595188	12.92727	1.5670022E-04	-0.3841659	0.0000000E+000	1.4247191E-09	491.4037	1.009935
8.695189	12.92770	1.5143915E-04	-0.3646991	0.0000000E+000	1.2859812E-09	491.3663	1.009936
8.795189	12.92815	1.4607220E-04	-0.3450413	0.0000000E+000	1.1540445E-09	491.3308	1.009936
8.895189	12.92862	1.4058892E-04	-0.3252614	0.0000000E+000	1.0288992E-09	491.2972	1.009936
8.995190	12.92911	1.3497694E-04	-0.3053560	0.0000000E+000	9.1053798E-10	491.2657	1.009937
9.095190	12.92963	1.2922152E-04	-0.2852748	0.0000000E+000	7.9895696E-10	491.2362	1.009937
9.195190	12.93018	1.2330431E-04	-0.2649413	0.0000000E+000	6.9415018E-10	491.2087	1.009937
9.295191	12.93076	1.1720235E-04	-0.2442688	0.0000000E+000	5.9611110E-10	491.1833	1.009938
9.395191	12.93138	1.1088753E-04	-0.2231786	0.0000000E+000	5.0485416E-10	491.1599	1.009938
9.495192	12.93205	1.0432707E-04	-0.2016184	0.0000000E+000	4.2044426E-10	491.1387	1.009938
9.595192	12.93276	9.7475749E-05	-0.1795555	0.0000000E+000	3.4293091E-10	491.1196	1.009939
9.695192	12.93355	9.0274465E-05	-0.1569940	0.0000000E+000	2.7240257E-10	491.1028	1.009939
9.795193	12.93441	8.2639453E-05	-0.1339718	0.0000000E+000	2.0896720E-10	491.0883	1.009940
9.895193	12.93536	7.4447584E-05	-0.1105664	0.0000000E+000	1.5278019E-10	491.0760	1.009941
9.995193	12.93641	6.5509215E-05	-8.6908951E-02	0.0000000E+000	1.0409303E-10	491.0662	1.009941
10.09519	12.93760	5.5478511E-05	-6.3147895E-02	0.0000000E+000	6.3224884E-11	491.0587	1.009942
10.19519	12.93899	4.3758282E-05	-3.9699417E-02	0.0000000E+000	3.1023684E-11	491.0536	1.009943
10.29519	12.94068	2.8148041E-05	-1.6551349E-02	0.0000000E+000	8.2576342E-12	491.0508	1.009944
10.36649	26.85000	0.0000000E+000	0.0000000E+000	0.0000000E+000	0.0000000E+000	491.0501	1.009944

PROGRAM DROP INPUT AND OUTPUT FILES FOR 500-MICRON PURE WATER DROP (FIGURES 14-16)

INPUT FILE (ALLH2O.INP)

```
     500.0       5.2     125.0         1       1.0
       0.0     1000.
       1.0      18.0
    101325.     100.0 4.250E-10     1.916  2.5044E6   -2442.5      1000.       0.0
```

OUTPUT FILE (ALLH2O.OUT)

0.0000000E+000	5.225298	5.0000404E-04	-2.034941	6.5449889E-08	125.0000	1.009042
1.000000	5.224943	4.9619668E-04	-2.018533	6.3966105E-08	122.9733	1.009042
2.000000	5.224575	4.9238082E-04	-2.002064	6.2501691E-08	120.9630	1.009041
3.000000	5.224219	4.8855640E-04	-1.985536	6.1056568E-08	118.9691	1.009041
4.000000	5.223836	4.8472313E-04	-1.968945	5.9630658E-08	116.9919	1.009041
5.000000	5.223449	4.8088102E-04	-1.952292	5.8223890E-08	115.0313	1.009041
6.000000	5.223088	4.7702983E-04	-1.935576	5.6836182E-08	113.0873	1.009040
7.000000	5.222679	4.7316944E-04	-1.918798	5.5467467E-08	111.1601	1.009040
8.000000	5.222277	4.6929964E-04	-1.901956	5.4117660E-08	109.2498	1.009040
9.000000	5.221871	4.6542037E-04	-1.885049	5.2786689E-08	107.3563	1.009040
10.00000	5.221475	4.6153137E-04	-1.868077	5.1474476E-08	105.4797	1.009040
11.00000	5.221093	4.5763250E-04	-1.851040	5.0180944E-08	103.6201	1.009039
12.00000	5.220630	4.5372357E-04	-1.833936	4.8906013E-08	101.7776	1.009039
13.00000	5.220200	4.4980441E-04	-1.816766	4.7649610E-08	99.95227	1.009039
14.00000	5.219765	4.4587482E-04	-1.799527	4.6411653E-08	98.14412	1.009038
15.00000	5.219334	4.4193459E-04	-1.782220	4.5192067E-08	96.35323	1.009038
16.00000	5.218884	4.3798354E-04	-1.764845	4.3990774E-08	94.57970	1.009038
17.00000	5.218433	4.3402147E-04	-1.747398	4.2807695E-08	92.82357	1.009038
18.00000	5.217959	4.3004812E-04	-1.729882	4.1642750E-08	91.08492	1.009037
19.00000	5.217495	4.2606334E-04	-1.712295	4.0495863E-08	89.36382	1.009037
20.00000	5.217013	4.2206683E-04	-1.694635	3.9366956E-08	87.66035	1.009037
21.00000	5.216563	4.1805836E-04	-1.676903	3.8255944E-08	85.97458	1.009036
22.00000	5.216040	4.1403770E-04	-1.659097	3.7162746E-08	84.30657	1.009036
23.00000	5.215535	4.1000458E-04	-1.641215	3.6087282E-08	82.65641	1.009036
24.00000	5.215014	4.0595868E-04	-1.623258	3.5029473E-08	81.02416	1.009035
25.00000	5.214493	4.0189977E-04	-1.605224	3.3989235E-08	79.40991	1.009035
26.00000	5.213963	3.9782756E-04	-1.587113	3.2966486E-08	77.81374	1.009035
27.00000	5.213433	3.9374174E-04	-1.568924	3.1961150E-08	76.23570	1.009034
28.00000	5.212878	3.8964197E-04	-1.550655	3.0973140E-08	74.67590	1.009034
29.00000	5.212316	3.8552799E-04	-1.532306	3.0002383E-08	73.13441	1.009034
30.00000	5.211806	3.8139941E-04	-1.513875	2.9048790E-08	71.61132	1.009033
31.00000	5.211181	3.7725590E-04	-1.495361	2.8112281E-08	70.10670	1.009033
32.00000	5.210582	3.7309705E-04	-1.476763	2.7192767E-08	68.62062	1.009033
33.00000	5.209985	3.6892245E-04	-1.458080	2.6290163E-08	67.15320	1.009032
34.00000	5.209362	3.6473179E-04	-1.439310	2.5404390E-08	65.70449	1.009032
35.00000	5.208745	3.6052454E-04	-1.420453	2.4535360E-08	64.27460	1.009031

36.00000	5.208101	3.5630033E-04	-1.401506	2.3682992E-08	62.86361	1.009031
37.00000	5.207449	3.5205868E-04	-1.382469	2.2847203E-08	61.47161	1.009031
38.00000	5.206778	3.4779913E-04	-1.363340	2.2027910E-08	60.09870	1.009030
39.00000	5.206100	3.4352115E-04	-1.344117	2.1225029E-08	58.74496	1.009030
40.00000	5.205458	3.3922424E-04	-1.324800	2.0438472E-08	57.41050	1.009029
41.00000	5.204710	3.3490776E-04	-1.305385	1.9668150E-08	56.09540	1.009029
42.00000	5.203977	3.3057120E-04	-1.285872	1.8913976E-08	54.79977	1.009028
43.00000	5.203237	3.2621386E-04	-1.266259	1.8175864E-08	53.52369	1.009028
44.00000	5.202477	3.2183516E-04	-1.246543	1.7453731E-08	52.26729	1.009027
45.00000	5.201706	3.1743440E-04	-1.226723	1.6747489E-08	51.03065	1.009027
46.00000	5.200925	3.1301082E-04	-1.206797	1.6057051E-08	49.81388	1.009026
47.00000	5.200107	3.0856373E-04	-1.186762	1.5382335E-08	48.61710	1.009026
48.00000	5.199280	3.0409225E-04	-1.166617	1.4723253E-08	47.44040	1.009025
49.00000	5.198492	2.9959559E-04	-1.146359	1.4079717E-08	46.28391	1.009025
50.00000	5.197561	2.9507279E-04	-1.125985	1.3451638E-08	45.14773	1.009024
51.00000	5.196662	2.9052290E-04	-1.105493	1.2838927E-08	44.03199	1.009024
52.00000	5.195756	2.8594487E-04	-1.084879	1.2241498E-08	42.93681	1.009023
53.00000	5.194806	2.8133765E-04	-1.064141	1.1659264E-08	41.86230	1.009023
54.00000	5.193857	2.7670004E-04	-1.043276	1.1092137E-08	40.80859	1.009022
55.00000	5.192864	2.7203080E-04	-1.022280	1.0540030E-08	39.77582	1.009021
56.00000	5.191839	2.6732864E-04	-1.001150	1.0002856E-08	38.76410	1.009021
57.00000	5.190826	2.6259210E-04	-0.9798812	9.4805284E-09	37.77357	1.009020
58.00000	5.189720	2.5781969E-04	-0.9584705	8.9729584E-09	36.80439	1.009019
59.00000	5.188609	2.5300967E-04	-0.9369137	8.4800584E-09	35.85669	1.009019
60.00000	5.187515	2.4816033E-04	-0.9152052	8.0017406E-09	34.93062	1.009018
61.00000	5.186288	2.4326971E-04	-0.8933408	7.5379196E-09	34.02633	1.009017
62.00000	5.185088	2.3833576E-04	-0.8713153	7.0885116E-09	33.14400	1.009016
63.00000	5.183831	2.3335626E-04	-0.8491221	6.6534311E-09	32.28377	1.009016
64.00000	5.182575	2.2832869E-04	-0.8267558	6.2325931E-09	31.44581	1.009015
65.00000	5.181194	2.2325040E-04	-0.8042091	5.8259140E-09	30.63032	1.009014
66.00000	5.179822	2.1811845E-04	-0.7814748	5.4333107E-09	29.83747	1.009013
67.00000	5.178389	2.1292960E-04	-0.7585450	5.0546993E-09	29.06744	1.009012
68.00000	5.176930	2.0768025E-04	-0.7354106	4.6899991E-09	28.32045	1.009011
69.00000	5.175347	2.0236650E-04	-0.7120623	4.3391335E-09	27.59670	1.009010
70.00000	5.173744	1.9698395E-04	-0.6884893	4.0020240E-09	26.89641	1.009009
71.00000	5.172141	1.9152768E-04	-0.6646798	3.6785930E-09	26.21981	1.009008
72.00000	5.170333	1.8599218E-04	-0.6406206	3.3687679E-09	25.56714	1.009007
73.00000	5.168530	1.8037129E-04	-0.6162975	3.0724812E-09	24.93867	1.009006
74.00000	5.166631	1.7465795E-04	-0.5916936	2.7896652E-09	24.33465	1.009005
75.00000	5.164664	1.6884410E-04	-0.5667905	2.5202553E-09	23.75538	1.009004
76.00000	5.162545	1.6292048E-04	-0.5415667	2.2641953E-09	23.20118	1.009002
77.00000	5.160390	1.5687683E-04	-0.5160003	2.0214510E-09	22.67260	1.009001
78.00000	5.157866	1.5070383E-04	-0.4866891	1.7920889E-09	22.17130	1.008999
79.00000	5.155379	1.4438991E-04	-0.4587480	1.5761487E-09	21.69885	1.008998
80.00000	5.152597	1.3791802E-04	-0.4286157	1.3735665E-09	21.25553	1.008996
81.00000	5.149546	1.3126883E-04	-0.3975306	1.1843263E-09	20.84209	1.008994
82.00000	5.146388	1.2442072E-04	-0.3660537	1.0084737E-09	20.46018	1.008992

83.00000	5.142897	1.1734253E-04	-0.3342832	8.4596574E-10	20.11015	1.008990
84.00000	5.139232	1.0999301E-04	-0.3020559	6.9675732E-10	19.79191	1.008987
85.00000	5.135226	1.0232037E-04	-0.2691114	5.6088334E-10	19.50624	1.008985
86.00000	5.130810	9.4254545E-05	-0.2351960	4.3842271E-10	19.25399	1.008982
87.00000	5.125776	8.5696753E-05	-0.2001441	3.2951797E-10	19.03621	1.008979
88.00000	5.120265	7.6498494E-05	-0.1639303	2.3439292E-10	18.85408	1.008975
89.00000	5.113903	6.6417611E-05	-0.1267031	1.5340357E-10	18.70871	1.008971
90.00000	5.106334	5.5008459E-05	-8.8807702E-02	8.7151279E-11	18.60097	1.008967
91.00000	5.097394	4.1236417E-05	-5.0766520E-02	3.6713663E-11	18.53126	1.008961
92.00000	5.085390	2.1147316E-05	-1.3499921E-02	4.9516498E-12	18.49932	1.008953
92.79070	20.00000	0.0000000E+000	0.0000000E+000	0.0000000E+000	18.49647	1.008952

PROGRAM DROP INPUT AND OUTPUT FILES FOR 630-MICRON WATER DROP WITH INERT COMPONENT (FIGURES 14-16)

INPUT FILE (PARTH2O.INP)

```
     630.0       5.2     300.0         1       1.0
       0.5     1000.
       0.5      18.0
    101325.     100.0 4.250E-10     1.916  2.5044E6   -2442.5      1000.        0.0
```

OUTPUT FILE (PARTH2O.OUT)

0.0000000E+000	5.235469	6.3000491E-04	-2.579755	6.5462189E-08	300.0000	1.009048
1.000000	5.235199	6.2643358E-04	-2.565210	6.3248272E-08	297.4276	1.009048
2.000000	5.234921	6.2285695E-04	-2.550619	6.1056170E-08	294.8696	1.009048
3.000000	5.234766	6.1927480E-04	-2.535981	5.8885828E-08	292.3263	1.009048
4.000000	5.234510	6.1568717E-04	-2.521296	5.6737175E-08	289.7977	1.009048
5.000000	5.234272	6.1209395E-04	-2.506563	5.4610140E-08	287.2838	1.009048
6.000000	5.234017	6.0849515E-04	-2.491782	5.2504667E-08	284.7846	1.009048
7.000000	5.233778	6.0489064E-04	-2.476954	5.0420681E-08	282.3002	1.009047
8.000000	5.233526	6.0128036E-04	-2.462076	4.8358118E-08	279.8307	1.009047
9.000000	5.233268	5.9766433E-04	-2.447150	4.6316909E-08	277.3761	1.009047
10.00000	5.233021	5.9404236E-04	-2.432174	4.4296986E-08	274.9364	1.009047
11.00000	5.232740	5.9041445E-04	-2.417149	4.2298286E-08	272.5117	1.009047
12.00000	5.232489	5.8678049E-04	-2.402074	4.0320739E-08	270.1021	1.009046
13.00000	5.232217	5.8314047E-04	-2.386948	3.8364281E-08	267.7076	1.009046
14.00000	5.231948	5.7949423E-04	-2.371772	3.6428840E-08	265.3282	1.009046
15.00000	5.231678	5.7584181E-04	-2.356546	3.4514350E-08	262.9641	1.009046
16.00000	5.231389	5.7218305E-04	-2.341267	3.2620751E-08	260.6151	1.009046
17.00000	5.231116	5.6851789E-04	-2.325938	3.0747966E-08	258.2815	1.009046
18.00000	5.230834	5.6484627E-04	-2.310556	2.8895933E-08	255.9633	1.009045
19.00000	5.230563	5.6116807E-04	-2.295123	2.7064583E-08	253.6604	1.009045
20.00000	5.230255	5.5748329E-04	-2.279635	2.5253842E-08	251.3731	1.009045
21.00000	5.229960	5.5379176E-04	-2.264096	2.3463638E-08	249.1012	1.009045
22.00000	5.229667	5.5009336E-04	-2.248503	2.1693905E-08	246.8449	1.009045
23.00000	5.229356	5.4638815E-04	-2.232856	1.9944570E-08	244.6042	1.009045
24.00000	5.229048	5.4267584E-04	-2.217155	1.8215564E-08	242.3792	1.009044
25.00000	5.228755	5.3895655E-04	-2.201400	1.6506815E-08	240.1699	1.009044
26.00000	5.228431	5.3522998E-04	-2.185588	1.4818256E-08	237.9764	1.009044
27.00000	5.228121	5.3149619E-04	-2.169722	1.3149813E-08	235.7987	1.009044
28.00000	5.227807	5.2775495E-04	-2.153799	1.1501417E-08	233.6370	1.009043
29.00000	5.227468	5.2400626E-04	-2.137820	9.8729940E-09	231.4911	1.009043
30.00000	5.227137	5.2024995E-04	-2.121785	8.2644709E-09	229.3613	1.009043
31.00000	5.226804	5.1648595E-04	-2.105691	6.6757764E-09	227.2476	1.009043
32.00000	5.226467	5.1271409E-04	-2.089541	5.1068385E-09	225.1500	1.009043
33.00000	5.226128	5.0893432E-04	-2.073332	3.5575836E-09	223.0685	1.009043
34.00000	5.225775	5.0514645E-04	-2.057064	2.0279378E-09	221.0033	1.009042
35.00000	5.225426	5.0135038E-04	-2.040737	5.1782950E-10	218.9544	1.009042

35.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	216.9190	1.009042
36.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	214.8841	1.009042
37.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	212.8492	1.009042
38.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	210.8143	1.009042
39.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	208.7794	1.009042
40.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	206.7446	1.009042
41.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	204.7097	1.009042
42.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	202.6748	1.009042
43.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	200.6399	1.009042
44.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	198.6050	1.009042
45.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	196.5701	1.009042
46.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	194.5352	1.009042
47.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	192.5003	1.009042
48.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	190.4654	1.009042
49.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	188.4306	1.009042
50.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	186.3957	1.009042
51.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	184.3608	1.009042
52.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	182.3259	1.009042
53.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	180.2910	1.009042
54.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	178.2561	1.009042
55.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	176.2212	1.009042
56.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	174.1863	1.009042
57.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	172.1514	1.009042
58.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	170.1165	1.009042
59.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	168.0816	1.009042
60.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	166.0468	1.009042
61.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	164.0119	1.009042
62.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	161.9770	1.009042
63.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	159.9421	1.009042
64.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	157.9072	1.009042
65.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	155.8723	1.009042
66.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	153.8374	1.009042
67.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	151.8025	1.009042
68.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	149.7677	1.009042
69.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	147.7328	1.009042
70.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	145.6979	1.009042
71.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	143.6630	1.009042
72.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	141.6281	1.009042
73.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	139.5932	1.009042
74.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	137.5583	1.009042
75.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	135.5234	1.009042
76.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	133.4885	1.009042
77.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	131.4536	1.009042
78.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	129.4187	1.009042
79.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	127.3839	1.009042
80.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	125.3490	1.009042
81.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	123.3141	1.009042

82.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	121.2792	1.009042
83.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	119.2443	1.009042
84.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	117.2094	1.009042
85.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	115.1745	1.009042
86.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	113.1396	1.009042
87.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	111.1047	1.009042
88.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	109.0698	1.009042
89.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	107.0350	1.009042
90.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	105.0001	1.009042
91.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	102.9652	1.009042
92.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	100.9303	1.009042
93.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	98.89540	1.009042
94.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	96.86051	1.009042
95.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	94.82562	1.009042
96.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	92.79073	1.009042
97.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	90.75584	1.009042
98.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	88.72095	1.009042
99.99976	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	86.68606	1.009042
100.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	84.65117	1.009042
101.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	82.61628	1.009042
102.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	80.58139	1.009042
103.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	78.54650	1.009042
104.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	76.51161	1.009042
105.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	74.47672	1.009042
106.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	72.44183	1.009042
107.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	70.40694	1.009042
108.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	68.37206	1.009042
109.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	66.33717	1.009042
110.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	64.30228	1.009042
111.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	62.26738	1.009042
112.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	60.23249	1.009042
113.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	58.19761	1.009042
114.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	56.16272	1.009042
115.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	54.12783	1.009042
116.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	52.09293	1.009042
117.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	50.05804	1.009042
118.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	48.02316	1.009042
119.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	45.98827	1.009042
120.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	43.95338	1.009042
121.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	41.91849	1.009042
122.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	39.88360	1.009042
123.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	37.84871	1.009042
124.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	35.81382	1.009042
125.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	33.77893	1.009042
126.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	31.74404	1.009042
127.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	29.70915	1.009042
128.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	27.67426	1.009042

129.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	25.63937	1.009042
130.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	23.60448	1.009042
131.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	21.56959	1.009042
132.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	19.53470	1.009042
133.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	17.49981	1.009042
134.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	15.46492	1.009042
135.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	13.43003	1.009042
136.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	11.39514	1.009042
137.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	9.360253	1.009042
138.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	7.325364	1.009042
139.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	5.290474	1.009042
140.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	3.255584	1.009042
141.9998	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	1.220695	1.009042
142.5996	20.00000	5.0003536E-04	-2.034890	0.0000000E+000	0.0000000E+000	1.009042

Distribution:

10	Patrick M. Gavin Gavin Consulting 1599 Krebs Court Newark, Ohio 43055	
10	MS 0767	B. A. Boughton, 5514
5	MS 0767	B. W. Marshall, 5514
1	MS 9018	Central Technical Files, 8523-2
5	MS 0899	Technical Library, 4414
2	MS 0619	Review & approval Desk, 12630 For DOE/OSTI