NASA CR-165,816

NASA Contractor Report 165816

NASA-CR-165816
19840021082

A NUMERICAL SIMULATION OF THE DISPERSAL OF AERIAL SPRAYS

M. B. Bragg

THE OHIO STATE UNIVERSITY
Aeronautical and Astronautical Research Laboratory
Columbus, Ohio 43220

NASA Purchase Order L-7965B
November 1981

National Aeronautics and Space Administration

Langley Research Center
Hampton, Virginia 23665

All Blank Pages

Intentionally Left Blank

To Keep Document Continuity

TABLE OF CONTENTS

NOMENCLATURE

A	Aircraft aspect ratio
b	Aircraft semispan, m
C	Concentration, volume/semispan^2
C_D	Drag coefficient, D/qs
C_L	Lift coefficient, L/qs
D_V	Diffusivity, cm^2/sec
F	Propeller thrust, N
F_r	Froude number, $U/\sqrt{bg}$
g	Gravitational constant, m/sec^2
h	Aircraft altitude, semispans
K	Inertia parameter, $\sigma\delta^2 U/18b\mu$
M_m	Mean molecular weight of gas mixture in transfer path
M_V	Mean molecular weight of evaporating material
m	Mass, Kg
P	Pressure, Kg/m^2 (lb/in^2)
P_f	Partial pressure of air, N/m^2
P_V	Vapor pressure of air, N/m^2
PK1 to PK4	Propeller constants
Q	Nozzle flow rate, volume/semispan along flight path
Q_p	Propeller torque, N-m
r	Radial coordinate, in propeller radii
R	Droplet Reynolds number $\rho\delta U\lvert\bar{u} - \dot{\bar{\eta}}\rvert/\mu$
R_U	Droplet free stream Reynolds number, $\rho\delta U/\mu$
RP	Propeller radius, m
S_c	Schmidt number, $\rho D_V/\mu$
T	Temperature, K(^{o}R)
t	Time, secs
U	Free stream velocity, m/sec
u	Local flowfield velocity, dimensionless with U
V	Cummulative volume percent
V_{cw}	Crosswind velocity component, m/sec
x, y, z	Cartesian coordinates
x_o, y_o, z_o	Initial droplet position (nozzle location), semispans
σ	Droplet density, Kg/m^3
σ_s	Standard deviation, microns
Γ	Aircraft circulation strength, m^2/sec
δ	Droplet diameter, microns
δ_m	Volume median droplet diameter, microns
η	Droplet position, dimensionless with b
μ	Air absolute viscosity, Kg/m-sec
ρ	Air density, Kg/m^3
τ	Dimensionless time, Ut/b

Superscripts

$\dot{}$	Derivative with respect to τ
$\bar{}$	Vector quantity

Subscripts

g	Final value at ground plane
wb	Wet bulb condition
db	Dry bulb condition

I. INTRODUCTION

The use of aerospace technology to improve aerial applications promises to yield significant improvements in the efficiency of operation, the environmental acceptability, and many other areas. One area of study is the interaction of the spray droplet with the aircraft wake. This interaction improves the overall efficiency of the operation by increasing the swath width, but also results in the drift of particles suspended in the wake to off target sites. The propeller slipstream and other nonuniform portions the wake also lead to nonuniform deposits of material inside the swath. This research uses an analytical approach to study this wake-particle interaction. Ultimately the goal of such a research effort will be to tailor the aircraft wake and dispersal system to produce a wide uniform distribution of material with minimum drift.

Reed[1] in 1954 presented a two-dimensional analysis of aerial applications. This report, aided by the techniques for trajectory analysis developed in the aircraft icing program, established the fundamentals of the method. This work was hampered by the limited computational capabilities of the time and Reed presents only a limited number of trajectories for his simple model. Bragg[2] improved this model by performing a three dimensional analysis including propeller slipstream effects Trayford and Welch[3] have made similar improvements including the effect of crosswind on the distribution. The present method is based on reference 2 but includes many improvements which have been developed by the author since the publication of reference 2.

The goal of this research was to document a method, and develop a computer code to simulate the dispersal of liquid sprays from agricultural aircraft. The program was to require small computation times so mathematical models and numerical procedures were kept as simple as possible while still modelling the fundamental physical phenomena. The program was written in such a way that a user could modify the wake model easily, while still using the particle dynamics model and the distribution and drift numerical procedures. The user may provide a flowfield code which can be inserted into this program by use of the USERV subroutine. This allows for the analysis of more complicated flowfields which may contain fuselage vortices, vortices off of flaps due to variable span loading, winglets, or more complicated flowfields due to unconventional aircraft configurations

The computer code consists of three main parts; the droplet dynamics and evaporation models, the aircraft wake models, and the distribution and drift prediction method. The theoretical basis for these models is derived in Section II and the method is compared to other results in Section III. Section IV is the computer program users' manual containing a description of the code and the input, output, and error messages. In the Appendices are sample cases and a complete listing of the code.

II. THEORETICAL DEVELOPMENT

In this section the mathematical models used for the droplet dynamics and the wake flowfield will be described. The numerical procedures used to solve these equation will also be presented.

Droplet Dynamics

By application of Newton's Second Law of Motion to a water droplet the differential equation describing the particle trajectory may be derived. Here particles in the 100 to 500 micron size range are considered. These particles are nearly spherical in shape[4] permitting the use of standard sphere drag data. In addition the analysis does not include the effect on the trajectory of atmospheric and aircraft induced turbulence, liquid droplet deformation and internal circulation, magnus forces, multiple particle interaction, and electric charge. For the aerial application problem these forces are considered negligible.

The equation of motion for a non-evaporating single particle moving in the wake of an aircraft can then be written as

$$m\left(\frac{d^2\bar{X}}{dt^2}\right) = \bar{D} + \bar{P} + \bar{M}_a + \bar{B} + m\bar{g} \tag{1}$$

The apparent mass, $\bar{M}_a$, the pressure gradient term, $\bar{P}$, and the Bassett force, $\bar{B}$, become important only if the density of the particle is of lower, or similar order of magnitude as that of air[5]. In addition, the apparent mass and Bassett force terms could also become significant if the particle experiences a large acceleration. The particle could experience a sizeable acceleration when initially injected into the flow. This acceleration is, however, of short duration and in the streamwise direction so as to have little effect on the final spanwise location of the particle. Therefore, noting that the density of water is much greater than that of air, and ignoring the initial acceleration effects; the apparent mass, the pressure gradient, and the Bassett force terms may be dropped from equation (1).

Now writing this equation in nondimensional form,

$$K\ddot{\bar{\eta}} = \frac{C_D R}{24}(\bar{u} - \dot{\bar{\eta}}) + \frac{1}{F_r{}^2}\frac{\bar{g}}{g} \qquad (2)$$

where the nondimensional numbers K, the inertia parameter, and F_r, the Froude number are

$$K = \frac{\sigma\delta^2 U}{18b\mu} \quad ; \quad F_r = \frac{U}{\sqrt{bg}} \qquad (3)\text{a,b}$$

An additional nondimensional number, R_U, the free stream droplet Reynolds number

$$R_U = \frac{\rho\delta U}{\mu} \qquad (4)$$

results from the $\frac{C_D R}{24}$ term in equation (2). The droplets studied here experience relatively low Reynolds numbers well within the range of zero to one thousand. Several curve fits to the standard sphere drag curve are available in this range and all yield similar results. Here the equation by Langmuir[6]

$$\frac{C_D R}{24} = 1. + 0.19R^{0.63} + 2.6 \times 10^{-4}R^{1.38} \qquad (5)$$

is used to calculate the sphere drag when computing the particle trajectories. Note that here R is the Reynolds number experienced by the droplet based on the particle velocity relative to the surrounding fluid

$$R = \frac{\rho \delta U |\bar{u} - \dot{\bar{\eta}}|}{\mu} \tag{6}$$

The trajectory of a particle is then governed by equation (2) which is a non-linear, second order, ordinary differential equation which must in general be solved numerically The equation may be reduced to first order and separated into its three components to form a system of six simultaneous differential equations Given the six initial conditions of particle position and velocity when first injected into the aircraft flowfield, the trajectory may be calculated using a step integration method. The program uses a variable step size, predictor-corrector scheme suitable for stiff systems[7,8]. When compared against non-stiff methods this scheme shows a considerable decrease in computational time to achieve the same accuracy. The trajectory is terminated when the particle intersects the ground plane or becomes entrained in a vortex.

Evaporation Model

For small liquid droplets under roughly 150 microns in diameter evaporation effects often become significant As a result of evaporation the ground distribution of material is reduced and the material lost due to drift is greatly increased. To include the effect of evaporation, the analysis of Goering[9] is used.

Since the mass of the particle is no longer constant the differential equation (1) must be modified. Having already incorporated the assumptions described early it becomes

$$m\left(\frac{d^2\bar{x}}{dt^2}\right) + \left(\frac{d\bar{x}}{dt}\right)\frac{dm}{dt} = \bar{D} + m\bar{g} \tag{7}$$

Nondimensionalizing, this equation becomes

$$K\left(\frac{d^2\eta}{d\tau^2}\right) = \left(\frac{C_D R}{24}\right)(\bar{u} - \dot{\bar{\eta}}) - \frac{3\dot{\bar{\eta}}K}{(\delta/b)}\left(\frac{d(\delta/b)}{d\tau}\right) + \frac{1}{F_r^2}\left(\frac{\bar{g}}{g}\right) \tag{8}$$

Equation (8) is identical to equation (2) except for the addition of the second term on the right hand side which is due to the $\frac{dm}{dt}$ term in equation (7). An additional variable, δ, the particle diameter is introduced in equation (8) and another equation must be added to the system to allow a solution.

The expression for the rate of change of the particle diameter with respect to time is[9]

$$\left(\frac{d(\delta/b)}{d\tau}\right) = -\left(\frac{2}{U}\right)\left(\frac{M_V}{M_m}\right)\left(\frac{D_V}{\delta}\right)\left(\frac{\rho}{\sigma}\right)\left(\frac{\Delta P}{P_f}\right)(2 + 0\ 6\ S_c^{1/3}R^{1/2}) \tag{9}$$

The terms on the right hand side may be evaluated in the following way.

It is assumed that the water droplet is diffusing into the air, therefore, the molecular weights are $M_m = 29\ 0$ and $M_V = 18.0$. The diffusion coefficient for water vapor into air is given as a function of temperature[9]

$$D_V = 5.28 \times 10^{-6}\ T_K^{1.88} \tag{10}$$

The term $\Delta P/P_f$ is shown by Goering[9] to be

$$\frac{\Delta P}{P_f} = \frac{P_{swb} - P_V}{P_{atm} - P_V} \tag{11}$$

Brooker[10] has developed a mathematical model for the psychrometric chart which is useful in evaluating equation (11). Input to the computer code are the atmospheric pressure, P_{atm}, and the wet and dry bulb temperatures, T_{wb} and T_{db} The saturation vapor pressure, P_{swb}, at wet bulb temperature, T_{wb}, is given by

$$P_{swb} = e^{54.6329 - \frac{12301.688}{T_{wb}} - 5.16923 \ln(T_{wb})} \qquad (12)$$

and the vapor pressure, P_V, is

$$P_V = P_{swb} + \frac{0.2405(P_{swb} - P_{atm})}{0.62194(1075.8965 - 0.56983(T_{wb} - 491.69))}(T_{db}-T_{wb}) \qquad (13)$$

In equation (12) and (13) the temperatures are in ^{o}R and the pressures are in psi. Equation (12) is valid only for temperatures above $32^{o}F$. Note that by definition $RH = P_V/P_s$, if the relative hunidity, RH, is available when T_{wb} is not.

The particle trajectory including the evaporation effect is calculated in the same manner as before, here using the differential equation (8) and adding to the system equation (9). Additional complications due to droplet evaporation which arise in the distribution and drift calculations will be discussed latter

Wake Flowfield

The flowfield model incorporated in the computer code is a horseshoe vortex system in ground effect. This simple model in either its 2-D or 3-D forms provides the wake velocities with a minimum of computer time, allowing for the small execution times characteristic of this program. In addition, a simple propeller, crosswind, and tunnel flowfield models are included in the code.

If these simple models are inadequate, the user may supply an alternate flowfield model by providing the proper subroutine. Here the flowfield models included in the code are discussed.

Basic Wake Model: The wake of an aircraft in ground effect may be modelled in three dimensions using a horse shoe vortex system. This model is reasonably accurate provided that, the wake roll-up occurs in a time much less than that of the particle trajectory, the particle trajectory intersects the ground before this system decays appreciably, and the particle remain outside of the viscous vortex cores and in the potential flow region. A two dimensional model can be formed consisting of only two doubly infinite straight trailing vortices and their reflection through the ground plane, figure 1. The strength of these vortices is calculated assuming an elliptical loading

$$\Gamma = \frac{4C_L Ub}{\pi A} \tag{14}$$

The motion of these vortices is given by Bragg[2] and the velocity can be calculated using the Biot-Savart law[1]. This model can be extended to three dimensions by adding a bound vortex of strength Γ on the aircraft quarter chord line, figure 2. The Biot-Savart Law is used to calculate the velocity induced in the x direction[2] and the y and z velocities are the same as in the two dimensional model.

It should be noted that the two dimensional model uses Reed's[1] correction for the bound vortex. If the user inputs zero for the initial droplet velocity, the program assigns to the droplet an initial velocity equal to the velocity the particle would have if falling in free air in a constant flowfield with the velocity components equal to that existing in the 2-D model at the point of particle injection. This correction is ignored if instead non-zero initial droplet velocities are input to the computer program.

Tunnel Model. The two dimensional model of reference 2 has been included in this program. This permits the calculation of droplet trajectories from an aircraft model in a wind tunnel to evaluate the influence of the tunnel walls and the floor. The tunnel ceiling is assumed to have a negligible effect on the droplet trajectories since a model in ground effect will be in close proximity to the tunnel floor. The simple geometry of this model is shown in figure 3. A conformal mapping method is used to determine the velocity at a point in the tunnel due to the two dimensional vortex system. The complex function describing the velocity potential and stream function in free air is transformed to the plane with infinite reflections of this vortex system in the y direction by using a complex sine mapping function. The vortex position is determined by step integration using a truncated series to express the induced velocities on a vortex filament. The derivative of this complex function with respect to the complex particle position, yields the y and z velocities at any point in the tunnel.

Propeller Model: A propeller slipstream is characterized by a rotation in the y-z plane due to the torque and an increased axial velocity due to the generated thrust. Functional forms for the axial and rotational velocities as a function of radial distance from the propeller hub have been assumed. These equations are of a form compatible with existing data on the thrust and torque loadings on general aviation propeller blades. The equation for the axial velocity is

$$V_{axial} = PK1(1. - r) + PK2(r)\ \sin(\pi r) \tag{15}$$

and the rotational velocity is given by

$$V_{rot} = PK3(1. - r) + PK4(r)\ \sin(\pi r) \tag{16}$$

Figure 4 shows the velocity distributions for a typical propeller model. PK1 and PK3 govern the velocities at the centerline while PK2 and PK4 control primarily the maximum velocities. For most applications PK3 = 0 and PK1 is set equal to some small fraction of PK2. Using equations (15) and (16) the slipstream can be integrated to determine the thrust, F, and the torque, Q_p. The thrust is

$$\begin{aligned} F = (RP)^2\pi\rho\Big[&2U(0.166667PK1 + 0.189304PK2) \\ &+ 0.0833333(PK1)^2 + 0.129006(PK1)(PK2) \\ &+ 0.0870040(PK2)^2\Big] \end{aligned} \tag{17}$$

and for the torque,

$$\begin{aligned} Q_p = 2\pi\rho(RP)^3\Big[&0.033333(PK1)(PK3) + 0.036657\big[(PK1)(PK4) \\ &+ (PK2)(PK3)\big] + 0.0507039(PK2)(PK4) + 0\ 083333U(PK3) \\ &+ 0.124801U(PK4)\Big] \end{aligned} \tag{18}$$

Equations (17) and (18) are then used to determine the constants PK1, PK2, PK3, and PK4 knowing the propeller thrust and torque This method assumes that the propeller slipstream extends downstream from the propeller disc plane to infinity with no dissipation.

In reality the rotational velocities are noticeably reduced due to viscous dissipation and the straighting effects of the main wing, fuselage, landing gear, horizontal, and vertical stabilizers. This reduction in rotational velocity is strongly dependent on the particular aircraft being modelled. It is recommended therefore that V_{rot}, that is PK3 and Pk4, be reduced from those calculated from equation (16). Although little experimental data is available on which to base this, a reduction to 75 percent of the calculated values seems reasonable from preliminary studies of conventional low wing monoplanes.

Crosswind Model: The component of the wind perpendicular to the flight path and parallel to the ground plane is modelled by the equation[11]

$$V_{cw} = VCW(z)^{0.25} \tag{19}$$

Here VCW is the crosswind component in m/sec 1 meter above the ground and z is the height above the ground plane in meters. The crosswind not only contributes a velocity in the y direction, but also causes a lateral translation of the entire aircraft wake system. This is modelled in the program by including the first order differential equations for the positions of the trailing vortices and propeller slipstream with the system of equations for the droplet trajectory and solving this enlarged system by the same step integration technique.

Distribution and Drift Model

By combining several droplet trajectories from a single nozzle location the ground deposition from the nozzle can be calculated and the drift estimated. This procedure is straight forward for the case of no crosswind, evaporation, or propeller and was first developed for the aerial spray problem by Reed[1]. Here an improved form of Reed's analysis is used for the simple case and major changes are made to handle the more difficult case of the unsymmetric and often double valued distributions.

The concentration, C, from a single nozzle can be written as

$$C = Q \frac{dV}{d\delta} \frac{d\delta}{dy_g} \tag{20}$$

By superposition the concentration from any number of nozzles can be found The derivative $d\delta/dy_g$ can be found from the plot of the particle diameter as a function of the final lateral position where the particle intersects the ground plane. These points are cubic splined to obtain $d\delta/dy_g$. This value must then be multiplied by $dV/d\delta$, a function of δ and the particular nozzle used, which represents the volume of material released by the nozzle as a function of particle diameter. Here a normal distribution of material about some volume median diameter, δ_m, is assumed which gives

$$\frac{dV}{d\delta} = \frac{1}{\sqrt{2\pi}\ \sigma_s} e^{-\frac{1}{2}\left(\frac{\delta - \delta_m}{\sigma_s}\right)^2} \tag{21}$$

where σ_s is the standard deviation The Q in equation (20) is the volume flow rate of the nozzle The concentration of material on the ground is then given by equation (20) and all that remains is to determine the drift.

The drift is estimated by an extrapolation procedure which finds the smallest particle, within some maximum allowable error, which reaches the ground. The amount of material in the distribution from particles smaller than this drifted particle is found by integration of the normal distribution function. The allowable error is a function of the number of standard deviations the particle size is from δ_m to control the error in terms of volume of material. Particles which encircle the wing-tip vortices are considered as drift and their trajectory calculations are terminated. In addition, off target distribution is also considered drift The user can select a lateral semispan distance beyond which the material is considered to be drift.

This procedure must be modified for unsymmetrical distributions or cases including droplet evaporation. As a result of the propeller slipstream effects or the crosswind model the calculation of $d\delta/dy_g$ can become more difficult. Here it is not uncommon for several particle sizes to intersect the ground at the same y_g location, thus making the function double valued The double valued and often erratic behavior of the curve makes the use of the cubic spline very difficult. Many points would be required to obtain a reasonable fit When these complications arise the program abandons the cubic spline and instead uses linear interpolation between the calculated trajectory points. In regions of large changes in slope the program automatically inserts additional particle trajectories using again an error parameter weighted from the volume median diameter. This method provides good results in a minimum of computer time. Here the drift is estimated in the same manner as before.

When the droplet evaporation is considered the mass lost through evaporation must be accounted for. This results in an increase in material lost due to drift and a decrease in the ground deposition. The final droplet diameter, δ_f, is determined as a function of the initial diameter, δ_o, by using a cubic spline on the results of the trajectory calculations. The distribution is then reduced by the ratio of δ_f^3/δ_o^3. In addition to the material which does not reach the ground or lies outside the target area, the amount of material which evaporates is determined by integration and included in the total drift. By using this procedure accurate drift and distributions are determined from the calculated particle trajectories.

III. CODE VALIDATION

Initial validation of the computer code was conducted by comparing the results of this method to the early analytical work of Reed[1]. Reed's method uses a two dimensional wake model without propeller, evaporation, or crosswind effects. The distribution is calculated using a least-squares fit of a second order polynomial to the trajectory data. A comparison of the two methods is shown in figure 5. Here the circulation, Γ, has been matched since the early work assumed a rectangular lift distribution. The comparison is quite good considering that modern digital computers were not available for Reed to make his trajectory calculations.

Comparisons of the code to experimental data is difficult because of the number of variables involved which are difficult to control or measure. Recent small scale model tests of agricultural aircraft dispensing scaled particles have generated some distribution data which may be used for comparison[12,13]. Figure 6 shows the measured distributions from three nozzles and the corresponding calculated distributions. The model used was unpowered with a wing semispan of 0.914m moving at 20.6 m/sec at an altitude of 0.51 semispans. The particles used were glass microbeads with a mean diameter of 125 microns and a density of 2.42 g/m^3. Since the test program was designed only to validate the scaling laws, no measurements of particle size distribution or nozzle volumetric flow rate was made. These parameters were estimated for this comparison. Note also that the concentration reported in reference 12 is in terms of the numbers of particles per unit area where this method calculates concentration in terms of volume of material per unit area. This accounts for the difference in the areas under the curve and also for the larger measured concentrations to the right of the peak concentration where the particles are smaller in diameter. The difference in the distributions at the y_o = 0.30 nozzle location is due in part to the fuselage vortex not accounted for in the present flowfield model. The comparison in figure 5 is however, quite good overall and supports the analysis.

In figure 7, the present method is again compared to data from reference 12. Here the same model is used as in figure 5 but the altitude is 0.35 semispans. The lateral position of the nozzle is at 0.40 semispans and the particles have a mean diameter of 140 microns and a density of 0.58 g/m^3. These particles scaled to a 200 micron water droplet for the full scale aircraft. Here again the comparison is quite good considering the differences in the definition of concentration. The amount of lateral transport of the material and the general shape and magnitude of the concentration curves compare quite well.

The present method was also compared qualitatively to the powered test conducted in reference 12. Again considering the influence of the fuselage vortex the comparisons were encouraging. The amount of lateral transport due to the propeller was approximately the same if the calculated swirl velocity was reduced by 25% as discussed earlier to account for straighting effects. Sufficient data were not available to make a detailed comparison of the distributions.

Detailed comparisons of this method have been conducted with both analytical and experimental data. The method has performed well over the range of parameters tested The code appears to do a good job of predicting the lateral transport of droplets in the wake of aircraft in ground effect.

IV. COMPUTER PROGRAM MANUAL

A computer program has been developed to predict the ground deposition resulting from the dispersal of water droplets into the wake of an aircraft in ground effect. A complete listing of this program has been provided in Appendix B. After a general description of the program, the input, output, error messages, and test cases will be presented in this section.

General Description

The computer code has been written in FORTRAN and developed on the AARL Harris Computer System. The code is in single precision and should execute on CDC systems with only minor changes. Conversion of the code to execute on IBM type machines should involve little more than a conversion to double precision and modifying the Hollerith codes. The code is approximately 2500 card images in length and requires 15 to 45 seconds of execution time per single nozzle deposition on the Harris SLASH 6 Computer. The Harris /6 is slightly slower than a CDC Cyber 73 and about 6 times slower than an IBM 370/168. Additional storage must be supplied to the program on octal file 10.

A plot package has been provided which is completely contained in subroutine PLOT1. A single call to this subroutine occurs as the last executable statement in the main program. Since the plot subroutine used here may not be compatible to those available on other systems, the program was written to allow for easy modification of the plot package.

Certain limitations on the size of the analysis that can be conducted considering the present DO loop parameters and array dimensions should be discussed. The maximum swath width that can be used is ± 10 semispans and the number of nozzles input can not exceed 100. The program is limited to 15 particle trajectories per nozzle including those determined internally to estimate the drift and improve accuracy in a region of a large change in lateral transport. Therefore no more than 10 particle diameters should ever be input and in general 5 to 7 is usually sufficient. No more than 1000 steps can be taken in the calculation of a single particle trajectory. The number of particles calculated and the number of steps can be controlled by use of the user input error parameters.

A flow chart including the major subroutines is shown in figure 8.

Input Data

The program reads the input from device 5 in the manner described below. Several options are available to the user with the acceptable combination of options shown in figure 9. In addition the user may supply the wake flowfield to be used by the inclusion of a subroutine USERV. This allows the present particle dynamics, including evaporation effects distribution, and drift models to be used with a different or more sophisticated wake model. USERV is described by comment statements in the listing. It may use by inclusion of the proper COMMON statements or subroutine calls use other variables and sections of the code as desired.

The inputs to the program are primarily physical variables in SI units. Some variables have been nondimensionalized where it seemed the most logical approach It is hoped that this type of input will make the code more useful. It should be noted however, that the problem can be nondimensionalized as in reference 11. The number of independent parameters is much less than the number of physical variables input to this code. The inputs could have been in terms of completely nondimensional numbers such as $R_{\bar{U}}$, K, and F_r as discussed earlier. The use of this code for sensitivity analysis can, therefore, be greatly simplified by considering the nondimensional parameters.

Card 1· (Format 13A6) Title

Card 2: (Format I5)
NDROPS Number of trajectories to be run. Set = 1 if NDIST = 1

Cards 3-8 repeated NDROPS times

Card 3: (Format 9I2, 2X, E15.6)

N2D	= 0	For a 3D run
	1	For a 2D run
	2	For a 2D run using the user supplied USERV subroutine
N3D	= 0	For a 2D run
	1	For a 3D run
	2	For a 3D run using the user supplied USERV subroutine
NEVAP	= 0	No droplet evaporation
	1	Droplet evaporation included
NPROP2	= 0	No propeller effects
	1	Propeller slipstream model included
NCW	= 0	No crosswind effects (set VCW = 0)
	1	Crosswind effects included
NTUN	= 0	No tunnel model
	1	Tunnel model included
NDIST	= 0	Single particle trajectory run
	1	Ground depositions calculated

NPRINT = 0 Short output option
1 Long output option

NPLOT = 0 No plotting
1 Plot distribution of trajectories

EPS Step integration error control parameter, usually set equal to 1.E-5.

Card 4 included only if NEVAP = 1

Card 4: (Format 3F10.6)

PA Atomspheric pressure, N/m^2

TDB Dry bulb temperature, ^oC

TWB Wet bulb temperature, ^oC

Card 5 included only if NPROP2=1

Card 5: (Format 6F10.4)

ZPROP2 Z coordinate of propeller hub, semispans

PK1 to 4 Propeller slipstream constants as defined in Section II

RP Slipstream radius (usually equal to propeller radius), semispans

Card 6 (Format 7F10.4)

A Aircraft aspect ratio

CL Aircraft lift coefficient

B Aircraft semispan, m

U Aircraft speed, m/sec

G Gravitational constant, m/sec^2

ZYO Initial Y coordinate of trailing vortex, semispans

ZZO Initial Z coordinate of trailing vortex, semispans

Card 7: (Format 2E20.6, 2F10.4)

DA	Air Density (ignored if NEVAP = 1), Kg/m^3
VIS	Absolute air viscosity (ignored if NEVAP = 1), Kg/m-sec
VCW	Crosswind velocity (=0. if NCW = 0), m/sec
D	Tunnel width (ignored if NTUN = 0), m

Card 8 included only if NDIST = 0

Card 8: (Format 8F10.5)

DIA	Particle diameter, microns
DD	Particle density, Kg/m^3
X, Y, Z	Initial particle position, semispans
UD, VD, WD	Initial particle velocity relative to the aircraft fixed coordinate system, nondimensional with respect to U

Cards 9 to 16 included if NDIST = 1

Card 9: (Format 6I5)

NCOL	Number of nozzles
NROW	Number of initial particle sizes input
NDIAMN =	0 All nozzles the same DIAMN 1 DIAMN input for each nozzle
NSTDEV =	0 All nozzle have the same STDEV 1 STDEV input for each nozzle
NQ =	0 All nozzles have the same Q 1 Q input for each nozzle
NZNOZ =	0 All nozzles have same Z coordinates 1 Z input for each nozzle

Card 10: (Format 8F10 5)

DD	Particle density, Kg/m^3
XO	Nozzle X coordinate (ignored if N2D = 1), semispans
UDO, VDO, WDO	Initial particle velocity relative to the aircraft fixed coordinate system, nondimensional with respect to U
SWIDTH	One-half the swadth width calculated, semispans

DERR — Constant by which default error limits are multiplied, normally set equal to one

DWIDTH — Material landing beyond $\pm$ this value are considered drift, semispans

Card 11: (Formant 8F10.5)

YNOZ — Y coordinate of each nozzle (NCOL values), semispans

Card 12. (Format 8F10.5)

SIZE — Particle sizes input to represent distribution (NROW values), microns

Card 13: (Format 8F10.5)

DIAMN — Median particle diameter, microns
Input one value if NDIAMN = 0
Input NCOL values if NDIAMN = 1

Card 14: (Format 8F10 5)

STDEV — Standard deviation of particle size distribution, microns
Input one value if NSTDEV = 0
Input NCOL values if NSTDEV = 1

Card 15. (Format 8F10.5)

Q — Nozzle flow rate, unit volume/semispan along flight path
Input one value if NQ = 0
Input NCOL values if NQ = 1

Card 16 (Format 8F10 5)

ZNOZ — Z coordinate of nozzle, semispan
Input one value if NZNOZ = 0
Input NCOL values if NZNOZ = 1

Cards 1-16 may be repeated to run several cases simultaneously.

Output

The program first outputs all the basic input data. The variables output are all in the same units as the corresponding inputs. After this initial listing, the output depends primarily on only the choice of NDIST and NPRINT. The program's calculated output is all dimensionless unless indicated otherwise.

For the NDIST = 1 case and NPRINT = 1, the distribution from each individual nozzle is listed. The output is self explanatory, note that concentration has units of volume per semispan squared. If more than one particle size lands at the same lateral position, YG, the diameter, DIA, column is omitted. The rest of the output is the same for both the NPRINT = 0 and 1 options A table including the input values for each nozzle is output along with the drift in percent of total for each nozzle. Next a trajectory summary listing all the trajectories calculated is included before the final distribution The final distribution is a compilation of the distributions from all nozzles and is output in 0.01 semispan increments. The total drift given is in percent of total material and is weighted by the volume flow rate of each nozzle. As a check, the total material reaching the tarjet area is also given. This is determined by a simple trapezoidal integration of the final integration and is meant only as a check. The accuracy of this estimate will be low for complicated distributions.

For the single particle trajectory calculations of NDIST = 0 the output is simple for NPRINT = 0. Here the initial inputs are output as before, followed by the final value of the variables in their normal units. The final values include particle position, velocity, droplet diameter for NEVAP = 1, and the nondimensional time, τ, required for the trajectory. This output is repeated for each particle trajectory.

If the user selects NPRINT = 1, all the dependent variables and the independent variable, τ, are output at each time step in the integration process. This can result in a great many lines in some cases. Since so many combinations of variables are possible depending of the user options selected, no header is provided for this output. The variables are output in the following manner. Column 1 is always the step number and 2 the nondimensional time, τ. Columns 3-6 always contain, the lateral position, Y, the lateral velocity, VD, the vertical position, Z, and the vertical velocity, WD If the run is 3-D, columns 7 and 8 contain the streamwise position, X, and the corresponding velocity, UD. If NEVAP = 1 the last column always contains the current droplet diameter. The columns between WD or UD and the droplet diameter contain the vortex and propeller slipstream position. If options are selected making the entire flowfield transport laterally, the columns will be Y and Z coordinate of the left, then right trailing vortex and then the Y and Z position of the propeller slipstream. If all these variables are not present, the initial values make it clear which one is being output.

Error Messages

The program provides several internal error messages and warnings. These messages and the most likely user actions needed to correct the problem are given here

MESSAGE	COMMENT
Max Number of Runs Exceeded Before Drift Calculation Completed (E)	Number of Particle Trajectories Exceeds 15, Reduce NROW or Increase DERR
Trajectory Terminated, Propeller Slipstream Entrainment (W)	Particle Considered Drift

MESSAGE	COMMENT
DO Loop Parameter Exceeded in SUB2D (E)	Number of Steps Exceeds 1000, Increase EPS
Trajectory Completed, Z less Than Zero	Normal Trajectory Termination
Trajectory Terminated, Vortex Entrainment (W)	Particle Considered Drift
DO Loop Parameter Exceeded in SUB3D (E)	Number of Steps Exceeds 1000, Reduce EPS
NVAR Incorrect (E)	Illegal Combination of Options
Cubic Spline Problem Nozzle No DIA = (E)	Cubic Spline Provides Poor Fit to Trajectory Data, Change Error Parameters or Input Particle Diameters
Insufficient Data To Calculate Distribution (E)	Not Enough Trajectories to Cubic Spline Particle Diameters for NEVAP = 1 Case, Change Input Particle Diameter
Insufficient Data To Estimate Drift, Nozzle No. Drift Set Equal To Zero (E)	(Same as Above)
WDR Exceeded DO Loop in INCON (E)	Error in 2D Initial Droplet Velocity Calculation, Input Non Zero Initial Velocity, Often Due to Illegal Input
Right Endpoint in INCON Incorrect (E)	(Same as Above)
DO Loop Parameter in INCON Exceeded (E)	(Same as Above)

(E) ERROR (W) WARNING

Sample Cases

Five sample cases have been provided in Appendix A. These cases are intended both as examples and as check cases after the program has been installed on a new system. The complete input, output, and plots are given for each of the cases. With the use of the input and output descriptions in this section, the cases should be self explanatory.

V SUMMARY

A method has been developed and a computer program written to predict the dispersal of aerial sprays from agricultural aircraft. The method has been compared to early analytical results and recent experimental studies and found to compare well with these results A user's manual for the program including sample cases and FORTRAN program listing have been provided The code was written to allow the flexibility to install a more sophisticated wake model to increase the usefullness of the program. The computer program provides a useful tool to aid in agricultural aircraft and dispersal system design as well as for aerial applications research

REFERENCES

1 Reed, Wilmer H., "An Analytical Study of the Effects of Airplane Wake on the Lateral Dispersion of Aerial Sprays," NACA Report 1196, 1954.

2. Bragg, M.B., "The Trajectory of a Liquid Droplet Injected Into the Wake of an Aircraft in Ground Effect," Aeronautical and Astronautical Engineering Department, University of Illinois, Technical Report 77-7, UIUC-ENG 770507, May 1977.

3. Trayford, R.S. and Welch, L.W., "Aerial Spraying: A Simulation of Factors Influencing the Distribution and Recovery of Liquid Droplets," J. agric. Engng Res. (1977)22, pp. 183-196.

4. Hughes, R.R. and Gilliland, E.R., "The Mechanics of Drops," Chemical Engineering Progress 48(10), 1952, pp. 497-504.

5. Soo, S.L., "Fluid Dynamics of Multiphase Systems," Blaisdell Publishing Company, Waltham Mass., 1967.

6. Langmuir, Irving and Blodgett, Katherine B., "A Mathematical Investigation of Water Droplet Trajectories," Army Air Forces Technical Report No. 5418 (Contract No. W-33-038-ac-9151), Feb. 1946.

7. Gear, C.W., "The Automatic Integration of Ordinary Differential Equations," Comm. ACM 14(March 1971) pp. 176-179.

8. Gear, C.W., "DIFSUB for Solution of Ordinary Differential Equations," Comm. ACM 14(March 1971) pp. 185-190.

9. Goering, C.E., Bode, L.W. and Sebhardt, M.R., "Mathematical Modeling of Spray Droplet Deceleration and Evaporation," TRANSACTIONS of the ASAE 15(2), 1972, pp. 220-225.

10. Brooker, D.B., "Mathematical Model of the Psychrometric Chart," TRANSACTIONS of the ASAE 10(4), 1967, pp. 558-560, 563.

11. Seiger, R., "The Climate Near the Ground," Harvard University Press, 1966.

12. Ormsbee, Allen I., Bragg, Michael B. and Maughmer, Mark D., "The Development of Methods for Predicting and Measuring Distribution Patterns of Aerial Sprays," University of Illinois, Aviation Research Laboratory Report ARL 79-1, June 1979.

13. Ormsbee, A.I., Bragg, M.B. and Maughmer, M.D , "Scaling Wake-Particle Interactions for Aerial Applications Research," AIAA Journal of Aircraft, Vol. 18, No. 7, July 1981, pp. 592-596.

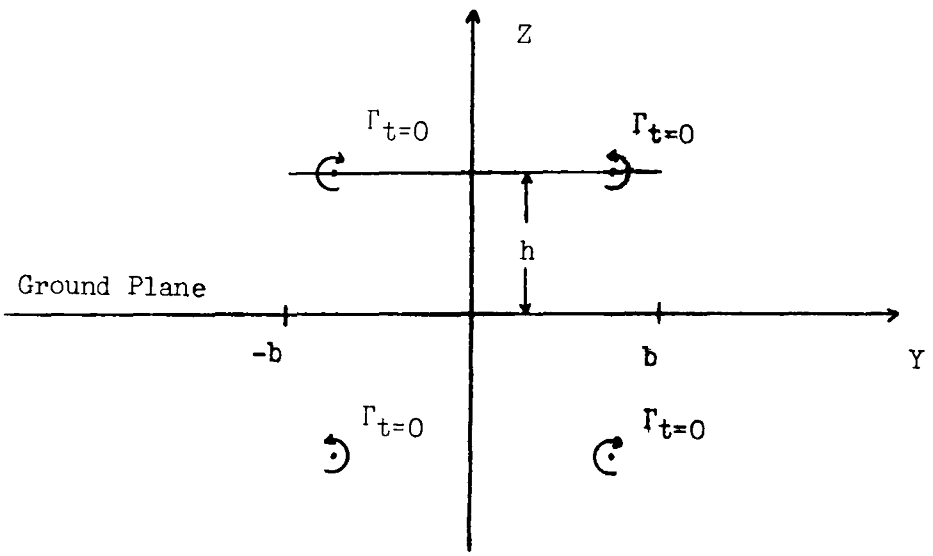


Figure 1. Two Dimensional Wake Vortex Model

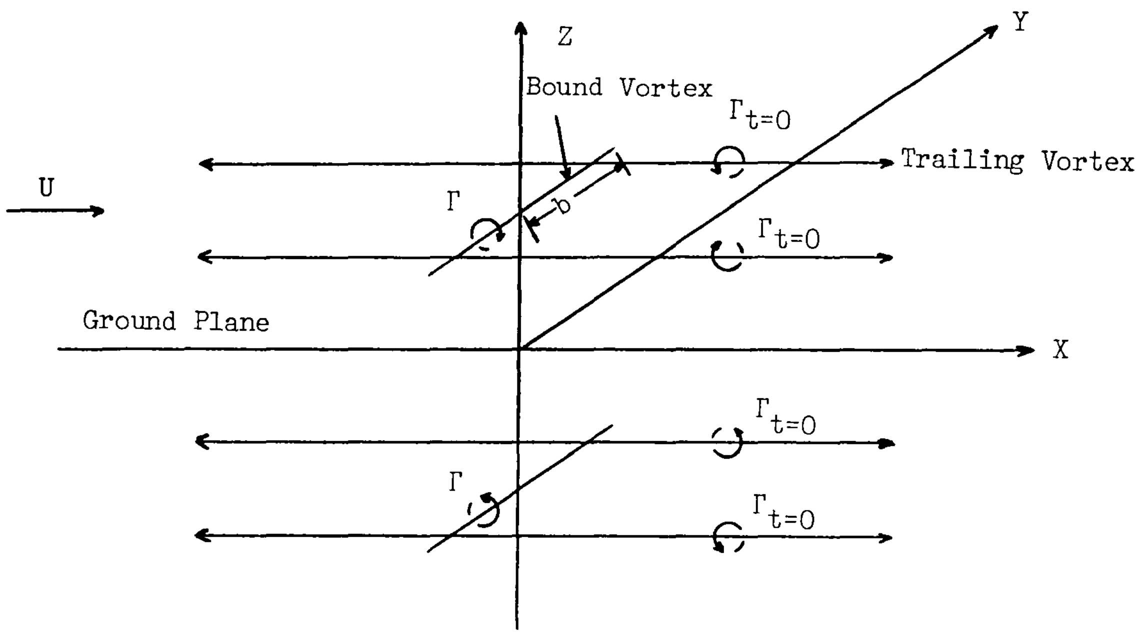


Figure 2. Three Dimensional Wake Vortex Model

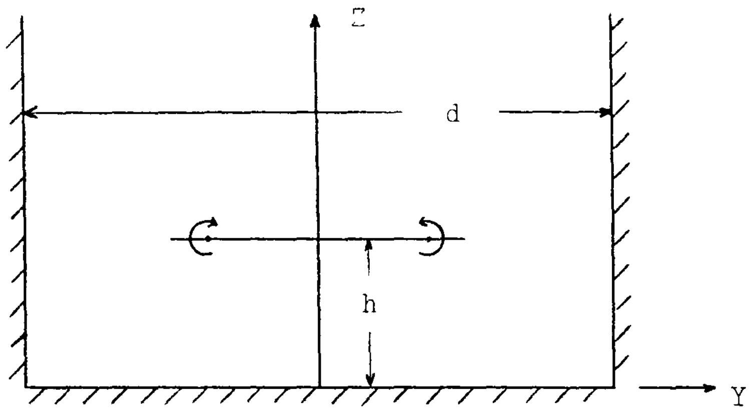


Figure 3 Two Dimensional Tunnel Model

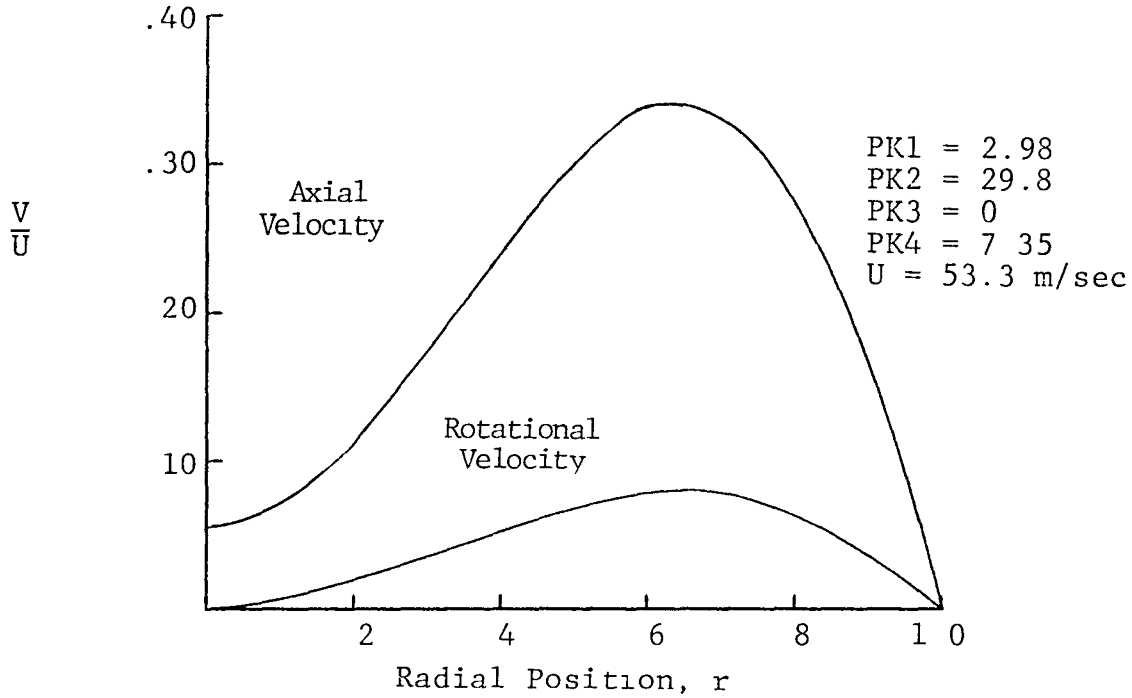


Figure 4. Typical Velocities for the Propeller Model

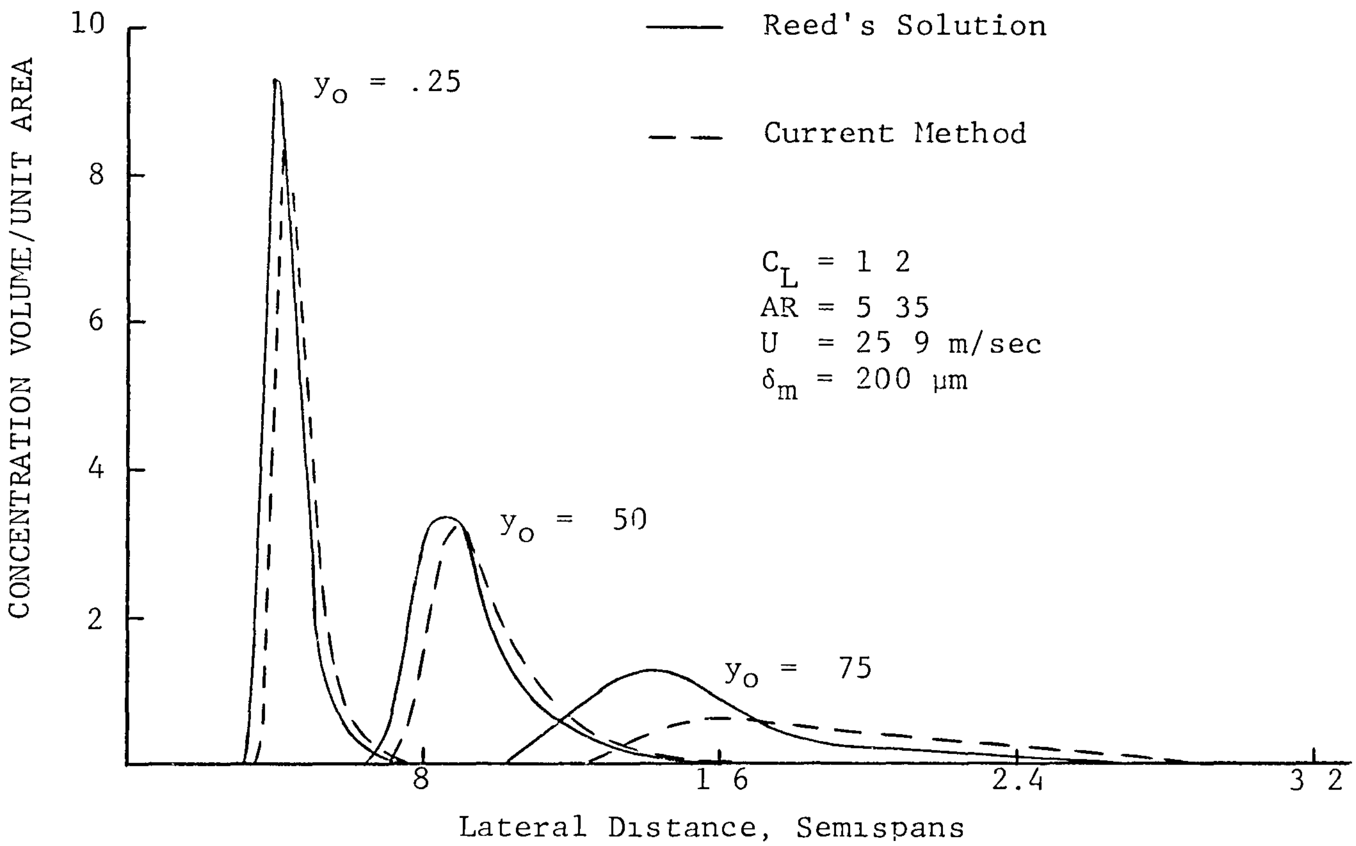


Figure 5. Present Method Compared to that of Reed

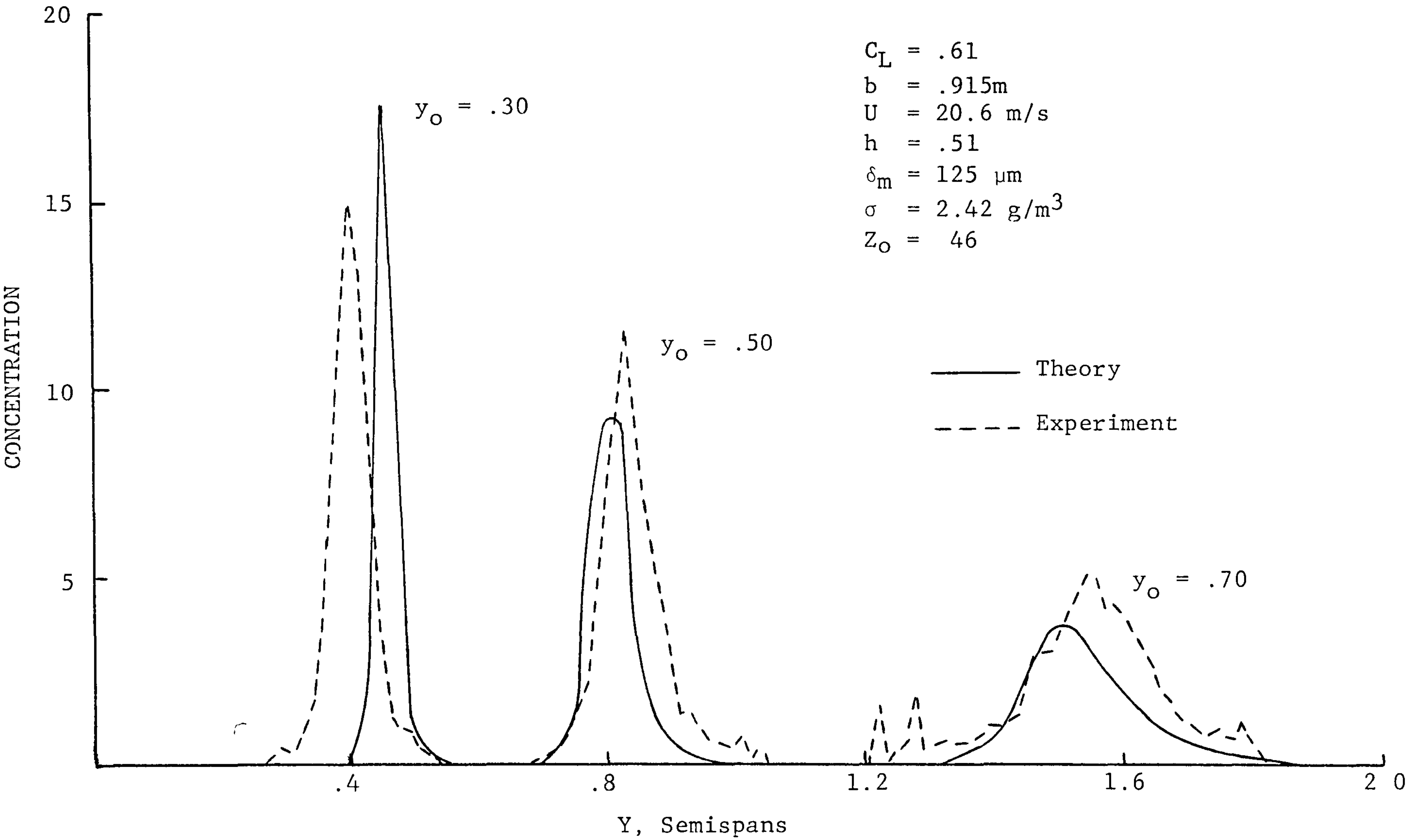


Figure 6. Comparison of Theory to Experiment for Three Nozzle Locations

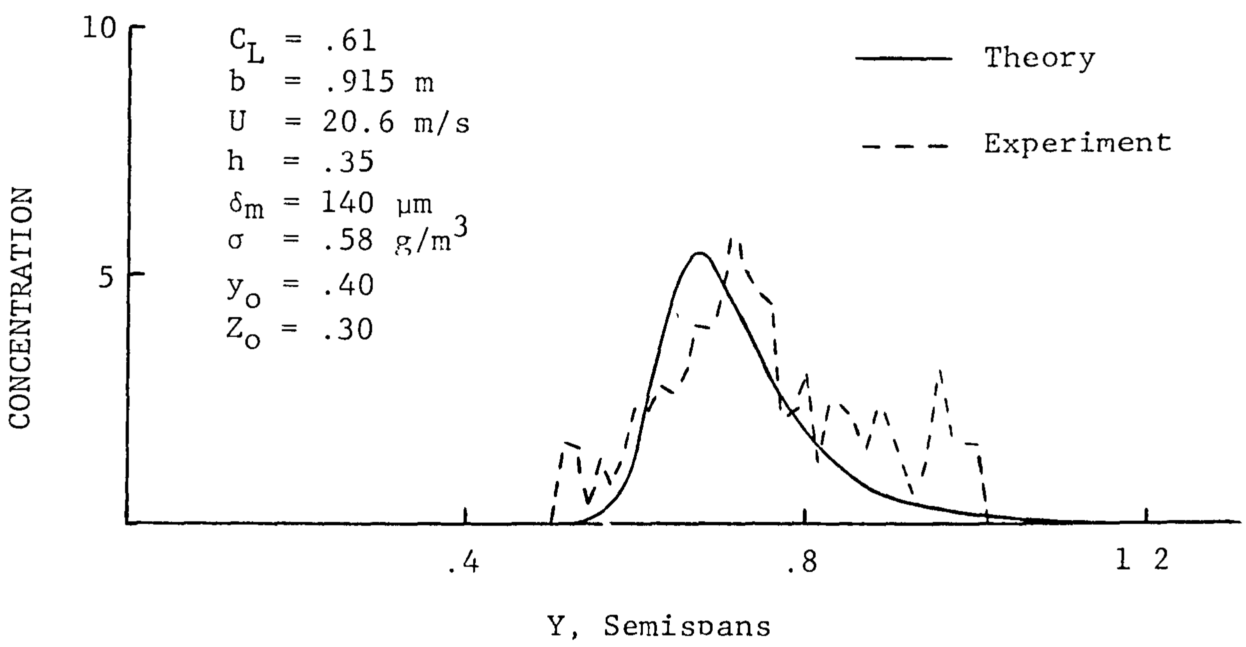


Figure 7 Comparison of Theory to Experiment

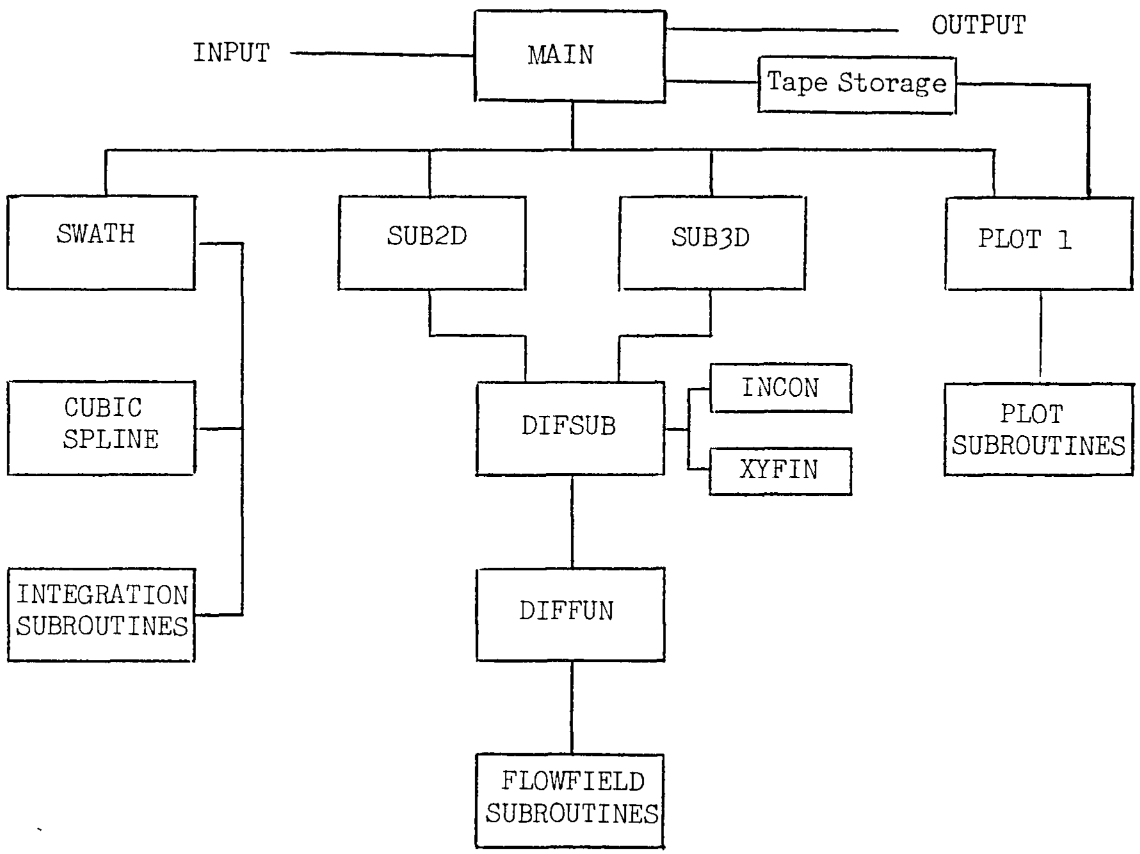


Figure 8. Computer Program Flow Chart

	N2D=1	N2D=2	N3D=1	N3D=2	NEVAP	NPROP2	NCW	NTUN	NDIST	NPLOT	NPRINT
N2D=1	X	-	-	-	X	X	X	X	X	X	X
N2D=2	-	X	-	-	X	-	-	-	X	X	X
N3D=1	-	-	X	-	X	X	X	-	X	X	X
N3D=2	-	-	-	X	X	-	-	-	X	X	X
NEVAP	X	X	X	X	X	X	X	X	X	X	X
NPROP2	X	-	X	-	X	X	X	X	X	X	X
NCW	X	-	X	-	X	X	X	-	X	X	X
NTUN	X	-	-	-	X	X	-	X	X	X	X
NDIST	X	X	X	X	X	X	X	X	X	X	X
NPRINT	X	X	X	X	X	X	X	X	X	X	X
NPLOT	X	X	X	X	X	X	X	X	X	X	X

X Allowable Option
- Unacceptable Option

Figure 9. Program Option Available

APPENDIX A

FIVE SAMPLE CASES

All five sets of input data are listed first, then each sample output including the plot.

```
ASP TEST CASE 1
1
 0 1 0 1 1 0 0 1 1                 1.E-5
.50       1.793      17.93      0.         6.029      0.20
6.        .6         6.096      53.035     9.8066     .8         .5
            1.225E00              1.7932E-5 2.5       0.
200.      1000.      0.         .10        .40        0.         0.         0.
$EOJ
```

```
ASP TEST CASE 2
1
 1 0 0 0 0 0 1 1 1                 1.E-5
6.        .6         6.096      53.035     9.8066     .8         .5
            1.225E00              1.7932E-5 0.        0.
1    6    1    1     1    1
1000.     0.         0.         0.         0.         4.0        1.         5.
.40
600.      300.       250.       200.       150.       100.
200.
50.
1.
.4
```

```
ASP TEST CASE 3
1
 1 0 1 0 0 0 1 0 1                 1.E-5
101352.1  23.8889    18.3333
6.        .6         6.096      53.035     9.8066     .8         .5
            1.225E00              1.7932E-5 0.        0.
1    6    1    1     1    1
1000.     0.         0.         0.         0.         4.0        1.         5.
.40
600.      300.       250.       200.       150.       100.
200.
50.
1.
.4
```

```
ASP TEST CASE 4
1
 1 0 0 0 0 1 1 0 1                 1.E-5
6.        .6        6.096     53.035    9.8066    .8        .5
            1.225F00           1.7932E-5 0.       36.576
1    6    1    1    1    1
1000.     0.        0.        0.        0.        4.0       1.        5.
.40
600.      300.      250.      200.      150.      100.
200.
50.
1.
.4
```

```
ASP TEST CASE 5
1
 0 1 1 1 1 0 1 0 1                 1.E-5
101352.1  23.8889   18.3333
.50       1.793     17.93     0.        6.029     .20
6.        .6        6.096     53.035    9.8066    .8        .5
            1.225F00           1.7932E-5 1.0      0.
3    6    1    1    1    1
1000.     0.        0.        0.        0.        4.0       1.        5.
-.30      0.        .30
600.      300.      250.      200.      150.      100.
200.      200.      200.
50.       50.       50.
1.        1.        1.
.4        .4        .4
```

OSU / AARL

AERIAL SPRAY PROGRAM

LAST UPDATE 2/1/81

ASP TEST CASE 1

PROGRAM CONTROL

```
N2D= 0              N3D= 1              NEVAP= 0            NPROP2= 1           NCW= 1
NTUN= 0             NDIST= 0            NPRINT= 1           NPLOT= 1            EPS= 0.100000E-04
```

PARAMETER VALUES

```
A=     6.0000       CL=     0.6000      B=     6.0960       U=   53.0350          G=    9.8066
ZY0=     0.8000     ZZ0=     0.5000     DA= 0.122500E+01    VIS= 0.179320E-04    VCW=     2.5000
D=     0.0000       DD= 1000.0000
```

PROP2 INPUTS

```
ZPROP2=     0.5000    PK1=     1.7930     PK2=    17.9300     PK3=     0.0000     PK4=     6.0290
RP=     0.2000
```

INITIAL VALUES

```
X=  0.000000        Y=  0.100000        Z=  0.400000        UD=  0.000000       VD=  0.000000
WD=  0.000000       DIA=  200.0000
```

1	0.00010	0.10000	0.00003	0.40000	-0.00011	0.00000	0.00142	0.80001	0.50000	-0.80000	0.50000	0.00001	0.50000
2	0.00020	0.10000	0.00005	0.40000	-0.00022	0.00000	0.00283	0.80002	0.50000	-0.79999	0.50000	0.00001	0.49999
3	0.00116	0.10000	0.00030	0.40000	-0.00124	0.00001	0.01625	0.80009	0.50000	-0.79994	0.50000	0.00007	0.49996
4	0.00212	0.10000	0.00054	0.40000	-0.00225	0.00003	0.02937	0.80016	0.49999	-0.79990	0.49999	0.00013	0.49993
5	0.00308	0.10000	0.00077	0.39999	-0.00323	0.00007	0.04219	0.80024	0.49999	-0.79985	0.49999	0.00019	0.49990
6	0.00472	0.10000	0.00116	0.39999	-0.00487	0.00015	0.06347	0.80036	0.49998	-0.79977	0.49998	0.00029	0.49985
7	0.00636	0.10001	0.00153	0.39998	-0.00644	0.00027	0.08395	0.80049	0.49998	-0.79970	0.49998	0.00040	0.49980
8	0.00801	0.10001	0.00189	0.39997	-0.00797	0.00043	0.10368	0.80062	0.49997	-0.79962	0.49997	0.00050	0.49975
9	0.00965	0.10001	0.00224	0.39995	-0.00945	0.00061	0.12269	0.80074	0.49997	-0.79954	0.49997	0.00060	0.49970
10	0.01185	0.10002	0.00268	0.39993	-0.01136	0.00091	0.14711	0.80091	0.49996	-0.79943	0.49996	0.00074	0.49963
11	0.01405	0.10002	0.00310	0.39990	-0.01321	0.00126	0.17040	0.80108	0.49995	-0.79933	0.49995	0.00088	0.49957
12	0.01625	0.10003	0.00350	0.39987	-0.01499	0.00166	0.19263	0.80125	0.49994	-0.79922	0.49994	0.00101	0.49950
13	0.01845	0.10004	0.00388	0.39984	-0.01671	0.00211	0.21387	0.80142	0.49993	-0.79912	0.49993	0.00115	0.49943
14	0.02225	0.10005	0.00449	0.39977	-0.01957	0.00299	0.24838	0.80171	0.49992	-0.79894	0.49992	0.00139	0.49931
15	0.02605	0.10007	0.00505	0.39969	-0.02231	0.00399	0.28043	0.80200	0.49991	-0.79876	0.49991	0.00162	0.49920
16	0.02986	0.10009	0.00557	0.39960	-0.02494	0.00512	0.31025	0.80229	0.49989	-0.79858	0.49989	0.00186	0.49908
17	0.03366	0.10011	0.00606	0.39950	-0.02749	0.00635	0.33808	0.80259	0.49988	-0.79839	0.49988	0.00210	0.49896
18	0.03746	0.10014	0.00650	0.39939	-0.02998	0.00768	0.36411	0.80288	0.49987	-0.79821	0.49987	0.00233	0.49884
19	0.04210	0.10017	0.00701	0.39924	-0.03294	0.00944	0.39372	0.80324	0.49995	-0.79799	0.49985	0.00262	0.49870
20	0.04674	0.10020	0.00747	0.39908	-0.03584	0.01134	0.42117	0.80359	0.49983	-0.79777	0.49983	0.00291	0.49856
21	0.05139	0.10024	0.00789	0.39891	-0.03870	0.01335	0.44670	0.80395	0.49982	-0.79755	0.49982	0.00320	0.49842
22	0.05603	0.10028	0.00828	0.39872	-0.04152	0.01548	0.47052	0.80431	0.49980	-0.79733	0.49980	0.00349	0.49827
23	0.06067	0.10032	0.00864	0.39853	-0.04432	0.01772	0.49281	0.80466	0.49978	-0.79711	0.49978	0.00378	0.49813

25	0.07287	0.10043	0.00945	0.39794	-0.05157	0.02406	0.54516	0.80560	0.49974	-0.79652	0.49974	0.00454	0.49775
26	0.07897	0.10049	0.00980	0.39761	-0.05516	0.02745	0.56851	0.80607	0.49972	-0.79623	0.49972	0.00492	0.49757
27	0.08506	0.10055	0.01011	0.39727	-0.05873	0.03099	0.59031	0.80654	0.49970	-0.79594	0.49970	0.00529	0.49738
28	0.09116	0.10061	0.01040	0.39690	-0.06228	0.03465	0.61073	0.80701	0.49968	-0.79565	0.49968	0.00567	0.49719
29	0.09726	0.10067	0.01065	0.39651	-0.06579	0.03843	0.62994	0.80748	0.49965	-0.79536	0.49965	0.00605	0.49700
30	0.10709	0.10078	0.01102	0.39583	-0.07136	0.04477	0.65870	0.80823	0.49962	-0.79489	0.49962	0.00666	0.49670
31	0.11692	0.10089	0.01133	0.39511	-0.07679	0.05137	0.68515	0.80899	0.49959	-0.79442	0.49959	0.00728	0.49640
32	0.12675	0.10100	0.01159	0.39433	-0.08203	0.05823	0.70963	0.80974	0.49955	-0.79396	0.49955	0.00789	0.49610
33	0.13658	0.10112	0.01181	0.39349	-0.08705	0.06532	0.73244	0.81050	0.49952	-0.79349	0.49952	0.00850	0.49580
34	0.14641	0.10123	0.01199	0.39261	-0.09179	0.07262	0.75378	0.81125	0.49948	-0.79302	0.49948	0.00911	0.49550
35	0.15623	0.10135	0.01214	0.39169	-0.09624	0.08013	0.77381	0.81201	0.49945	-0.79255	0.49945	0.00972	0.49520
36	0.17090	0.10153	0.01231	0.39023	-0.10229	0.09169	0.80153	0.81314	0.49939	-0.79185	0.49939	0.01063	0.49475
37	0.18556	0.10171	0.01242	0.38869	-0.10758	0.10363	0.82692	0.81427	0.49934	-0.79115	0.49934	0.01154	0.49430
38	0.20023	0.10190	0.01249	0.38708	-0.11213	0.11593	0.85024	0.81539	0.49929	-0.79045	0.49929	0.01245	0.49385
39	0.21489	0.10208	0.01251	0.38541	-0.11596	0.12856	0.87167	0.81652	0.49924	-0.78976	0.49924	0.01336	0.49340
40	0.22956	0.10226	0.01251	0.38368	-0.11910	0.14149	0.89137	0.81765	0.49919	-0.78906	0.49919	0.01427	0.49296
41	0.24422	0.10245	0.01247	0.38192	-0.12163	0.15470	0.90948	0.81878	0.49914	-0.78836	0.49914	0.01518	0.49251
42	0.26215	0.10267	0.01240	0.37972	-0.12396	0.17119	0.92965	0.82016	0.49907	-0.78751	0.49907	0.01630	0.49197
43	0.28008	0.10289	0.01231	0.37748	-0.12557	0.18802	0.94785	0.82153	0.49901	-0.78665	0.49901	0.01741	0.49142
44	0.29801	0.10311	0.01220	0.37522	-0.12656	0.20516	0.96427	0.82291	0.49895	-0.78580	0.49895	0.01852	0.49088
45	0.31594	0.10333	0.01207	0.37294	-0.12704	0.22259	0.97911	0.82429	0.49889	-0.78495	0.49889	0.01963	0.49033
46	0.33387	0.10354	0.01194	0.37066	-0.12702	0.24026	0.99252	0.82567	0.49882	-0.78409	0.49882	0.02074	0.48979
47	0.35180	0.10376	0.01181	0.36839	-0.12679	0.25817	1.00464	0.82705	0.49876	-0.78324	0.49876	0.02185	0.48925
48	0.37972	0.10408	0.01160	0.36486	-0.12575	0.28646	1.02126	0.82920	0.49866	-0.78191	0.49866	0.02358	0.48840
49	0.40764	0.10440	0.01141	0.36137	-0.12420	0.31517	1.03552	0.83135	0.49857	-0.78059	0.49857	0.02531	0.48755
50	0.43556	0.10472	0.01123	0.35793	-0.12229	0.34426	1.04777	0.83349	0.49847	-0.77926	0.49847	0.02704	0.48672
51	0.46348	0.10503	0.01109	0.35454	-0.12012	0.37366	1.05830	0.83564	0.49837	-0.77793	0.49837	0.02876	0.48588
52	0.49140	0.10534	0.01097	0.35122	-0.11778	0.40334	1.06737	0.83779	0.49828	-0.77660	0.49828	0.03049	0.48504
53	0.51932	0.10564	0.01088	0.34797	-0.11534	0.43325	1.07518	0.83994	0.49818	-0.77528	0.49818	0.03222	0.48421
54	0.55757	0.10606	0.01082	0.34362	-0.11190	0.47457	1.08415	0.84288	0.49805	-0.77346	0.49805	0.03458	0.48306
55	0.59583	0.10647	0.01082	0.33941	-0.10843	0.51619	1.09145	0.84583	0.49791	-0.77165	0.49791	0.03694	0.48192
56	0.63409	0.10689	0.01089	0.33532	-0.10500	0.55806	1.09736	0.84877	0.49778	-0.76983	0.49778	0.03930	0.48078
57	0.67235	0.10731	0.01102	0.33137	-0.10162	0.60014	1.10212	0.85172	0.49765	-0.76802	0.49765	0.04166	0.47965
58	0.71060	0.10773	0.01121	0.32755	-0.09832	0.64238	1.10590	0.85466	0.49752	-0.76620	0.49752	0.04402	0.47852
59	0.74886	0.10817	0.01147	0.32385	-0.09512	0.68474	1.10885	0.85761	0.49739	-0.76439	0.49739	0.04637	0.47739
60	0.80074	0.10877	0.01190	0.31902	-0.09093	0.74235	1.11175	0.86161	0.49721	-0.76193	0.49721	0.04956	0.47586
61	0.85263	0.10940	0.01243	0.31441	-0.08690	0.80009	1.11358	0.86560	0.49704	-0.75947	0.49704	0.05275	0.47433
62	0.90451	0.11006	0.01306	0.31000	-0.08302	0.85789	1.11452	0.86960	0.49686	-0.75701	0.49686	0.05594	0.47282
63	0.95639	0.11076	0.01377	0.30579	-0.07929	0.91573	1.11469	0.87360	0.49668	-0.75456	0.49668	0.05913	0.47130
64	1.00828	0.11149	0.01457	0.30177	-0.07570	0.97355	1.11420	0.87759	0.49651	-0.75210	0.49651	0.06231	0.46980
65	1.06016	0.11227	0.01545	0.29793	-0.07226	1.03133	1.11315	0.88159	0.49634	-0.74965	0.49634	0.06549	0.46829
66	1.13080	0.11341	0.01675	0.29299	-0.06780	1.10990	1.11092	0.88704	0.49610	-0.74631	0.49610	0.06981	0.46625
67	1.20144	0.11464	0.01814	0.28835	-0.06363	1.18827	1.10797	0.89248	0.49586	-0.74298	0.49586	0.07413	0.46423
68	1.27209	0.11597	0.01959	0.28399	-0.05978	1.26642	1.10446	0.89793	0.49563	-0.73964	0.49563	0.07845	0.46221
69	1.34273	0.11741	0.02108	0.27990	-0.05624	1.34431	1.10055	0.90338	0.49539	-0.73631	0.49539	0.08276	0.46020
70	1.41337	0.11895	0.02257	0.27604	-0.05303	1.42191	1.09639	0.90883	0.49516	-0.73298	0.49516	0.08707	0.45820
71	1.48401	0.12060	0.02404	0.27240	-0.05013	1.49921	1.09209	0.91428	0.49493	-0.72965	0.49493	0.09137	0.45621
72	1.57953	0.12299	0.02596	0.26778	-0.04669	1.60324	1.08624	0.92165	0.49462	-0.72515	0.49462	0.09717	0.45353
73	1.67504	0.12555	0.02777	0.26346	-0.04374	1.70671	1.08048	0.92902	0.49431	-0.72066	0.49431	0.10297	0.45087
74	1.77055	0.12829	0.02946	0.25941	-0.04123	1.80964	1.07496	0.93639	0.49400	-0.71617	0.49400	0.10876	0.44823
75	1.86606	0.13118	0.03102	0.25557	-0.03911	1.91206	1.06973	0.94377	0.49369	-0.71168	0.49369	0.11454	0.44560
76	1.96158	0.13421	0.03244	0.25193	-0.03730	2.01399	1.06482	0.95114	0.49338	-0.70720	0.49338	0.12031	0.44299
77	2.08133	0.13819	0.03404	0.24757	-0.03542	2.14116	1.05915	0.96039	0.49300	-0.70159	0.49300	0.12753	0.43975
78	2.20108	0.14235	0.03545	0.24343	-0.03389	2.26768	1.05397	0.96965	0.49262	-0.69598	0.49262	0.13474	0.43653
79	2.32083	0.14668	0.03671	0.23945	-0.03263	2.39361	1.04925	0.97890	0.49225	-0.69037	0.49225	0.14194	0.43334
80	2.44058	0.15114	0.03782	0.23560	-0.03159	2.51900	1.04494	0.98816	0.49187	-0.68478	0.49187	0.14912	0.43017
81	2.56033	0.15573	0.03881	0.23187	-0.03072	2.64389	1.04098	0.99743	0.49150	-0.67918	0.49150	0.15628	0.42703
82	2.68008	0.16043	0.03970	0.22824	-0.02999	2.76832	1.03732	1.00669	0.49113	-0.67360	0.49113	0.16343	0.42392
83	2.83746	0.16676	0.04074	0.22358	-0.02919	2.93123	1.03288	1.01887	0.49065	-0.66626	0.49065	0.17281	0.41987
84	2.99485	0.17325	0.04167	0.21904	-0.02852	3.09346	1.02877	1.03106	0.49017	-0.65894	0.49017	0.18216	0.41587
85	3.15224	0.17987	0.04252	0.21460	-0.02796	3.25507	1.02491	1.04325	0.48970	-0.65162	0.48970	0.19149	0.41192
86	3.30962	0.18663	0.04331	0.21024	-0.02748	3.41609	1.02123	1.05545	0.48923	-0.64432	0.48923	0.20079	0.40800
87	3.46701	0.19350	0.04405	0.20595	-0.02705	3.57653	1.01769	1.06765	0.48877	-0.63702	0.48877	0.21007	0.40414
88	3.62439	0.20049	0.04476	0.20172	-0.02667	3.73643	1.01424	1.07985	0.48831	-0.62973	0.48831	0.21932	0.40032
89	3.75188	0.20623	0.04533	0.19834	-0.02640	3.86556	1.01149	1.08974	0.48793	-0.62384	0.48793	0.22679	0.39726
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

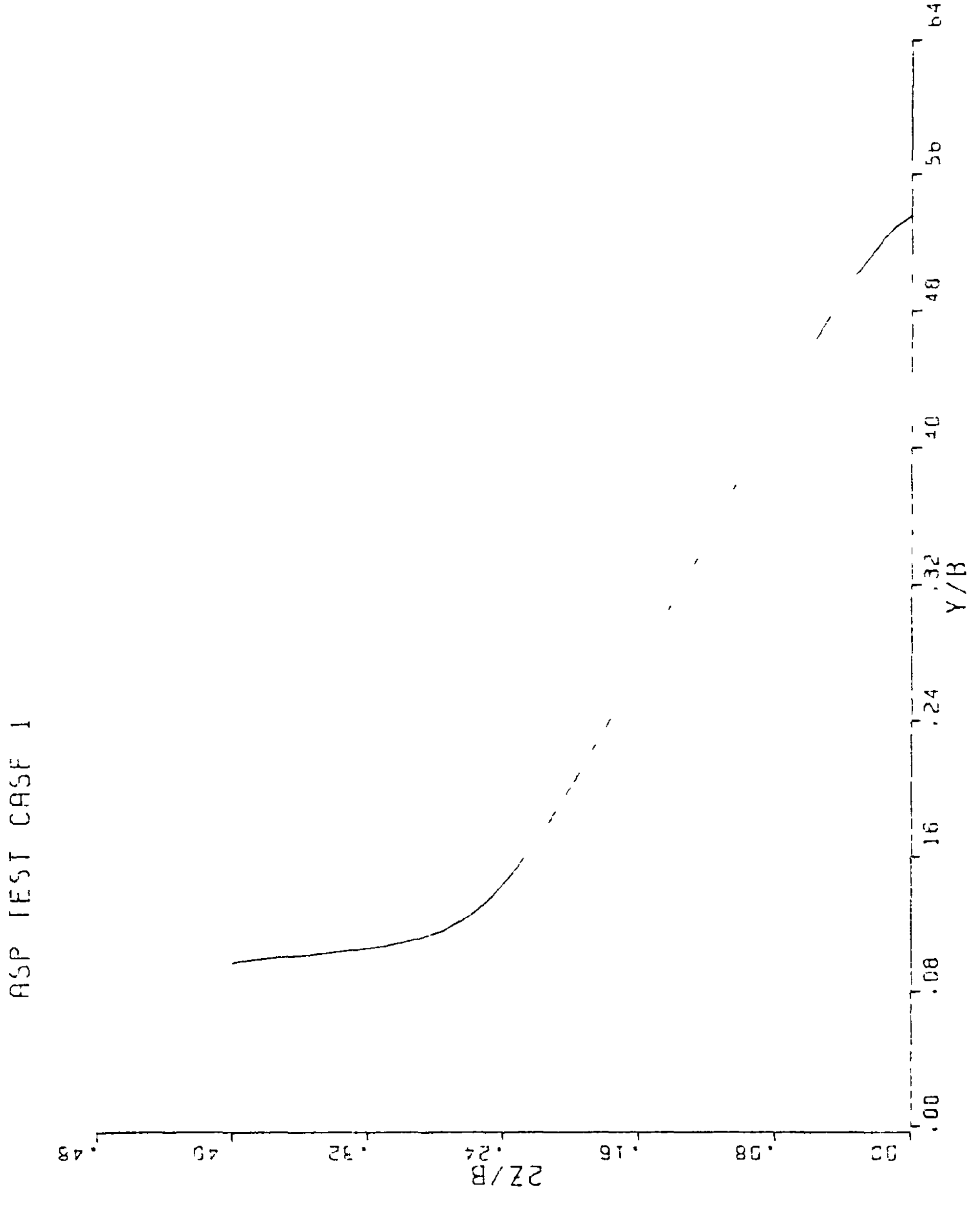
90	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
91	3.79186	0.20805	0.04549	0.19729	-0.02632	3.90597	1.01067	1.09284	0.48782	-0.62199	0.48782	0.22913	0.39630
92	3.81903	0.20929	0.04559	0.19657	-0.02627	3.93342	1.01014	1.09495	0.48774	-0.62073	0.48774	0.23072	0.39566
93	3.84620	0.21053	0.04568	0.19586	-0.02623	3.96086	1.00964	1.09706	0.48766	-0.61948	0.48766	0.23231	0.39501
94	3.87337	0.21177	0.04577	0.19515	-0.02619	3.98828	1.00917	1.09917	0.48758	-0.61822	0.48758	0.23390	0.39437
95	3.95217	0.21539	0.04597	0.19309	-0.02607	4.06776	1.00793	1.10528	0.48736	-0.61459	0.48736	0.23850	0.39251
96	4.03097	0.21901	0.04612	0.19104	-0.02598	4.14714	1.00687	1.11140	0.48713	-0.61095	0.48713	0.24310	0.39066
97	4.10977	0.22265	0.04622	0.18899	-0.02588	4.22644	1.00596	1.11752	0.48691	-0.60732	0.48691	0.24769	0.38882
98	4.18857	0.22630	0.04628	0.18696	-0.02579	4.30568	1.00517	1.12364	0.48669	-0.60369	0.48669	0.25227	0.38699
99	4.29082	0.23103	0.04630	0.18433	-0.02567	4.40842	1.00430	1.13158	0.48640	-0.59898	0.48640	0.25821	0.38464
100	4.39308	0.23577	0.04628	0.18171	-0.02554	4.51108	1.00358	1.13953	0.48611	-0.59427	0.48611	0.26413	0.38230
101	4.49533	0.24049	0.04622	0.17910	-0.02541	4.61366	1.00297	1.14747	0.48583	-0.58957	0.48583	0.27005	0.37998
102	4.59759	0.24522	0.04613	0.17651	-0.02526	4.71620	1.00246	1.15542	0.48554	-0.58488	0.48554	0.27596	0.37768
103	4.69984	0.24993	0.04600	0.17394	-0.02511	4.81868	1.00204	1.16337	0.48526	-0.58018	0.48526	0.28185	0.37540
104	4.87481	0.25795	0.04574	0.16957	-0.02483	4.99395	1.00147	1.17698	0.48478	-0.57216	0.48478	0.29191	0.37154
105	5.04977	0.26593	0.04542	0.16525	-0.02453	5.16913	1.00105	1.19059	0.48431	-0.56415	0.48431	0.30194	0.36773
106	5.22473	0.27384	0.04506	0.16098	-0.02422	5.34425	1.00075	1.20421	0.48384	-0.55615	0.48384	0.31193	0.36397
107	5.39970	0.28169	0.04466	0.15677	-0.02391	5.51933	1.00052	1.21784	0.48337	-0.54816	0.48337	0.32190	0.36026
108	5.57466	0.28947	0.04424	0.15262	-0.02359	5.69437	1.00036	1.23147	0.48291	-0.54018	0.48291	0.33183	0.35661
109	5.84230	0.30122	0.04356	0.14637	-0.02311	5.96208	1.00019	1.25233	0.48222	-0.52800	0.48222	0.34695	0.35111
110	6.10994	0.31279	0.04285	0.14025	-0.02263	6.22976	1.00009	1.27320	0.48153	-0.51584	0.48153	0.36201	0.34573
111	6.37758	0.32416	0.04211	0.13425	-0.02217	6.49741	1.00003	1.29409	0.48085	-0.50370	0.48085	0.37698	0.34047
112	6.64523	0.33533	0.04135	0.12838	-0.02171	6.76506	1.00000	1.31499	0.48019	-0.49159	0.48019	0.39188	0.33531
113	6.91287	0.34629	0.04057	0.12263	-0.02127	7.03269	0.99998	1.33590	0.47953	-0.47950	0.47953	0.40670	0.33027
114	7.18051	0.35704	0.03977	0.11699	-0.02084	7.30033	0.99996	1.35682	0.47888	-0.46744	0.47888	0.42145	0.32534
115	7.55511	0.37173	0.03863	0.10930	-0.02026	7.67491	0.99996	1.38613	0.47799	-0.45059	0.47799	0.44196	0.31861
116	7.92970	0.38598	0.03745	0.10181	-0.01971	8.04949	0.99996	1.41546	0.47712	-0.43379	0.47712	0.46231	0.31208
117	8.30430	0.39978	0.03624	0.09453	-0.01918	8.42407	0.99996	1.44481	0.47627	-0.41704	0.47627	0.48251	0.30574
118	8.67890	0.41313	0.03500	0.08744	-0.01867	8.79866	0.99997	1.47419	0.47543	-0.40033	0.47543	0.50256	0.29960
119	9.05349	0.42600	0.03371	0.08053	-0.01819	9.17324	0.99997	1.50359	0.47461	-0.38366	0.47461	0.52246	0.29364
120	9.42809	0.43838	0.03238	0.07381	-0.01773	9.54783	0.99997	1.53302	0.47380	-0.36703	0.47380	0.54220	0.28786
121	10.05584	0.45798	0.03005	0.06291	-0.01700	10.17556	0.99998	1.58237	0.47249	-0.33926	0.47249	0.57494	0.27856
122	10.68359	0.47607	0.02756	0.05246	-0.01631	10.80329	0.99998	1.63180	0.47123	-0.31161	0.47123	0.60725	0.26972
123	11.31133	0.49254	0.02486	0.04242	-0.01567	11.43103	0.99998	1.68128	0.47000	-0.28407	0.47000	0.63913	0.26131
124	11.93908	0.50723	0.02190	0.03277	-0.01506	12.05877	0.99999	1.73082	0.46882	-0.25664	0.46882	0.67058	0.25331
125	12.56683	0.51995	0.01857	0.02350	-0.01448	12.68651	0.99999	1.78042	0.46767	-0.22931	0.46767	0.70160	0.24570
126	13.19458	0.53041	0.01464	0.01459	-0.01390	13.31425	0.99999	1.83008	0.46656	-0.20209	0.46656	0.73219	0.23845
127	13.53726	0.53500	0.01209	0.00988	-0.01359	13.65693	0.99999	1.85721	0.46597	-0.18728	0.46597	0.74872	0.23463
128	13.79823	0.53786	0.00982	0.00637	-0.01333	13.91789	0.99999	1.87788	0.46553	-0.17601	0.46553	0.76122	0.23180
129	13.98649	0.53953	0.00788	0.00388	-0.01313	14.10615	0.99999	1.89280	0.46521	-0.16790	0.46521	0.77019	0.22978
130	14.09471	0.54032	0.00657	0.00247	-0.01301	14.21438	0.99999	1.90138	0.46503	-0.16324	0.46503	0.77533	0.22864
131	14.17140	0.54078	0.00548	0.00147	-0.01291	14.29106	0.99999	1.90746	0.46491	-0.15994	0.46491	0.77896	0.22783
132	14.22379	0.54104	0.00462	0.00080	-0.01282	14.34345	0.99999	1.91162	0.46482	-0.15768	0.46482	0.78144	0.22729
133	14.25067	0.54116	0.00410	0.00045	-0.01277	14.37034	0.99999	1.91375	0.46478	-0.15653	0.46478	0.78271	0.22701
134	14.26808	0.54123	0.00372	0.00023	-0.01274	14.38775	0.99999	1.91513	0.46475	-0.15578	0.46475	0.78354	0.22682
135	14.27811	0.54126	0.00347	0.00010	-0.01271	14.39778	0.99999	1.91593	0.46473	-0.15535	0.46473	0.78401	0.22672
136	14.28217	0.54128	0.00335	0.00005	-0.01270	14.40184	0.99999	1.91625	0.46473	-0.15517	0.46473	0.78420	0.22668
137	14.28461	0.54128	0.00328	0.00002	-0.01269	14.40428	0.99999	1.91644	0.46472	-0.15507	0.46472	0.78432	0.22665
138	14.28627	0.54129	0.00321	0.00000	-0.01268	14.40594	0.99999	1.91657	0.46472	-0.15500	0.46472	0.78440	0.22664
139	14.28793	0.54129	0.00314	-0.00002	-0.01268	14.40760	0.99999	1.91670	0.46472	-0.15492	0.46472	0.78447	0.22662

**** TRAJECTORY COMPLETED, Z LESS THAN ZERO ****

FINAL VALUES

X= 14.405961 Y= 0.541290 Z= 0.000000 UD= 0.999992 VD= 0.003211
WD= -0.012684 DIA= 200.0000 T= 14.286297

HAKGND: STOP


ASP TEST CASE 1
Y/B
2Z/B
.00
.08
.16
.24
.32
.40
.48
.56
.64

OSU / AARL

AERIAL SPRAY PROGRAM

LAST UPDATE 2/1/81

ASP TEST CASE 2

PROGRAM CONTROL

N2D= 1	N3D= 0	NEVAP= 0	NPROP2= 0	NCW= 0
NTUN= 0	NDIST= 1	NPRINT= 1	NPLOT= 1	EPS= 0.100000E-04

PARAMETER VALUES

A= 6.0000	CL= 0.6000	B= 6.0960	U= 53.0350	G= 9.8066
ZY0= 0.8000	ZZ0= 0.5000	DA= 0.122500E+01	VIS= 0.179320E-04	VCW= 0.0000
D= 0.0000	DD= 1000.0000			

DISTRIBUTION INPUTS

NCOL= 1	NROW= 6	NDIAMN= 1	NSTDEV= 1	NQ= 1
NZNOZ= 1	DD=1000.00000	X= 0.00000	UD= 0.00000	VD= 0.00000
WD= 0.00000	SWIDTH= 4.00000	DERR= 1.00000	DWIDTH= 5.00000	

YNOZ=0.400

YG	DIA	CONCENTRATION	YG	DIA	CONCENTRATION	YG	DIA	CONCENTRATION
0.640000E+00	0.441061E+03	0.110564E-03	0.650000E+00	0.426213E+03	0.408700E-03	0.660000E+00	0.412488E+03	0.126168E-02
0.670000E+00	0.399762E+03	0.334558E-02	0.680000E+00	0.387928E+03	0.779777E-02	0.690000E+00	0.376896E+03	0.162809E-01
0.700000E+00	0.366585E+03	0.309327E-01	0.710000E+00	0.356927E+03	0.541859E-01	0.720000E+00	0.347861E+03	0.884828E-01
0.730000E+00	0.339334E+03	0.135946E+00	0.740000E+00	0.331299E+03	0.198073E+00	0.750000E+00	0.323714E+03	0.275513E+00
0.760000E+00	0.316543E+03	0.367965E+00	0.770000E+00	0.309752E+03	0.474192E+00	0.780000E+00	0.303311E+03	0.592145E+00
0.790000E+00	0.297195E+03	0.719134E+00	0.800000E+00	0.291380E+03	0.851925E+00	0.810000E+00	0.285844E+03	0.987170E+00
0.820000E+00	0.280570E+03	0.112161E+01	0.830000E+00	0.275540E+03	0.125221E+01	0.840000E+00	0.270738E+03	0.137634E+01
0.850000E+00	0.266150E+03	0.149180E+01	0.860000E+00	0.261762E+03	0.159689E+01	0.870000E+00	0.257562E+03	0.169039E+01
0.880000E+00	0.253538E+03	0.177154E+01	0.890000E+00	0.249681E+03	0.184000E+01	0.900000E+00	0.245979E+03	0.189559E+01
0.910000E+00	0.242426E+03	0.193848E+01	0.920000E+00	0.239012E+03	0.196920E+01	0.930000E+00	0.235731E+03	0.198850E+01
0.940000E+00	0.232576E+03	0.199723E+01	0.950000E+00	0.229539E+03	0.199634E+01	0.960000E+00	0.226616E+03	0.198681E+01
0.970000E+00	0.223799E+03	0.196964E+01	0.980000E+00	0.221084E+03	0.194580E+01	0.990000E+00	0.218467E+03	0.191622E+01
0.100000E+01	0.215941E+03	0.188178E+01	0.101000E+01	0.213502E+03	0.184330E+01	0.102000E+01	0.211147E+03	0.180153E+01
0.103000E+01	0.208872E+03	0.175716E+01	0.104000E+01	0.206672E+03	0.171079E+01	0.105000E+01	0.204544E+03	0.166297E+01
0.106000E+01	0.202485E+03	0.161418E+01	0.107000E+01	0.200492E+03	0.156483E+01	0.108000E+01	0.198561E+03	0.151536E+01
0.109000E+01	0.196691E+03	0.146626E+01	0.110000E+01	0.194877E+03	0.141774E+01	0.111000E+01	0.193117E+03	0.136998E+01
0.112000E+01	0.191409E+03	0.132313E+01	0.113000E+01	0.189751E+03	0.127730E+01	0.114000E+01	0.188139E+03	0.123259E+01
0.115000E+01	0.186572E+03	0.118905E+01	0.116000E+01	0.185048E+03	0.114675E+01	0.117000E+01	0.183566E+03	0.110572E+01
0.118000E+01	0.182122E+03	0.106597E+01	0.119000E+01	0.180717E+03	0.102752E+01	0.120000E+01	0.179347E+03	0.990365E+00
0.121000E+01	0.178013E+03	0.954498E+00	0.122000E+01	0.176711E+03	0.919905E+00	0.123000E+01	0.175442E+03	0.886566E+00
0.124000E+01	0.174204E+03	0.854457E+00	0.125000E+01	0.172995E+03	0.823551E+00	0.126000E+01	0.171815E+03	0.793816E+00
0.127000E+01	0.170662E+03	0.765222E+00	0.128000E+01	0.169536E+03	0.737734E+00	0.129000E+01	0.168436E+03	0.711317E+00
0.130000E+01	0.167360E+03	0.685937E+00	0.131000E+01	0.166308E+03	0.661557E+00	0.132000E+01	0.165279E+03	0.638143E+00
0.133000E+01	0.164272E+03	0.615658E+00	0.134000E+01	0.163286E+03	0.594069E+00	0.135000E+01	0.162322E+03	0.573341E+00
0.136000E+01	0.161377E+03	0.553441E+00	0.137000E+01	0.160452E+03	0.534334E+00	0.138000E+01	0.159546E+03	0.515991E+00
0.139000E+01	0.158657E+03	0.498379E+00	0.140000E+01	0.157787E+03	0.481469E+00	0.141000E+01	0.156934E+03	0.465232E+00
0.142000E+01	0.156097E+03	0.449639E+00	0.143000E+01	0.155276E+03	0.434664E+00	0.144000E+01	0.154471E+03	0.420280E+00
0.145000E+01	0.153681E+03	0.406463E+00	0.146000E+01	0.152906E+03	0.393188E+00	0.147000E+01	0.152146E+03	0.380433E+00
0.148000E+01	0.151399E+03	0.368174E+00	0.149000E+01	0.150665E+03	0.356392E+00	0.150000E+01	0.149945E+03	0.345064E+00
0.151000E+01	0.149238E+03	0.334066E+00	0.152000E+01	0.148543E+03	0.323321E+00	0.153000E+01	0.147861E+03	0.312838E+00
0.154000E+01	0.147192E+03	0.302624E+00	0.155000E+01	0.146536E+03	0.292683E+00	0.156000E+01	0.145893E+03	0.283019E+00
0.157000E+01	0.145262E+03	0.273632E+00	0.158000E+01	0.144644E+03	0.264523E+00	0.159000E+01	0.144038E+03	0.255689E+00
0.160000E+01	0.143445E+03	0.247128E+00	0.161000E+01	0.142864E+03	0.238838E+00	0.162000E+01	0.142295E+03	0.230813E+00
0.163000E+01	0.141738E+03	0.223049E+00	0.164000E+01	0.141192E+03	0.215542E+00	0.165000E+01	0.140659E+03	0.208284E+00
0.166000E+01	0.140136E+03	0.201271E+00	0.167000E+01	0.139626E+03	0.194496E+00	0.168000E+01	0.139126E+03	0.187953E+00
0.169000E+01	0.138637E+03	0.181636E+00	0.170000E+01	0.138159E+03	0.175537E+00	0.171000E+01	0.137691E+03	0.169650E+00
0.172000E+01	0.137234E+03	0.163969E+00	0.173000E+01	0.136788E+03	0.158487E+00	0.174000E+01	0.136351E+03	0.153197E+00
0.175000E+01	0.135924E+03	0.148094E+00	0.176000E+01	0.135507E+03	0.143170E+00	0.177000E+01	0.135100E+03	0.138419E+00
0.178000E+01	0.134701E+03	0.133837E+00	0.179000E+01	0.134312E+03	0.129415E+00	0.180000E+01	0.133932E+03	0.125150E+00
0.181000E+01	0.133561E+03	0.121034E+00	0.182000E+01	0.133199E+03	0.117063E+00	0.183000E+01	0.132845E+03	0.113232E+00
0.184000E+01	0.132499E+03	0.109534E+00	0.185000E+01	0.132162E+03	0.105966E+00	0.186000E+01	0.131833E+03	0.102522E+00
0.187000E+01	0.131511E+03	0.991974E-01	0.188000E+01	0.131197E+03	0.959882E-01	0.189000E+01	0.130891E+03	0.928897E-01
0.190000E+01	0.130592E+03	0.898979E-01	0.191000E+01	0.130300E+03	0.870086E-01	0.192000E+01	0.130016E+03	0.842382E-01
0.193000E+01	0.129738E+03	0.816006E-01	0.194000E+01	0.129466E+03	0.790880E-01	0.195000E+01	0.129201E+03	0.766933E-01
0.196000E+01	0.128942E+03	0.744098E-01	0.197000E+01	0.128689E+03	0.722311E-01	0.198000E+01	0.128442E+03	0.701514E-01
0.199000E+01	0.128200E+03	0.681652E-01	0.200000E+01	0.127963E+03	0.662675E-01	0.201000E+01	0.127731E+03	0.644532E-01
0.202000E+01	0.127503E+03	0.627181E-01	0.203000E+01	0.127281E+03	0.610577E-01	0.204000E+01	0.127063E+03	0.594682E-01
0.205000E+01	0.126849E+03	0.579458E-01	0.206000E+01	0.126639E+03	0.564870E-01	0.207000E+01	0.126433E+03	0.550886E-01
0.208000E+01	0.126231E+03	0.537473E-01	0.209000E+01	0.126033E+03	0.524604E-01	0.210000E+01	0.125839E+03	0.512251E-01
0.211000E+01	0.125647E+03	0.500387E-01	0.212000E+01	0.125460E+03	0.488990E-01	0.213000E+01	0.125275E+03	0.478034E-01
0.214000E+01	0.125093E+03	0.467500E-01	0.215000E+01	0.124915E+03	0.457366E-01	0.216000E+01	0.124739E+03	0.447614E-01
0.217000E+01	0.124567E+03	0.438225E-01	0.218000E+01	0.124397E+03	0.429181E-01	0.219000E+01	0.124229E+03	0.420468E-01
0.220000E+01	0.124064E+03	0.412068E-01	0.221000E+01	0.123902E+03	0.403969E-01	0.222000E+01	0.123742E+03	0.396155E-01
0.223000E+01	0.123584E+03	0.388614E-01	0.224000E+01	0.123429E+03	0.381334E-01	0.225000E+01	0.123275E+03	0.374303E-01
0.226000E+01	0.123124E+03	0.367510E-01	0.227000E+01	0.122975E+03	0.360944E-01	0.228000E+01	0.122828E+03	0.354595E-01
0.229000E+01	0.122682E+03	0.348454E-01	0.230000E+01	0.122539E+03	0.342512E-01	0.231000E+01	0.122397E+03	0.336761E-01
0.232000E+01	0.122257E+03	0.331191E-01	0.233000E+01	0.122119E+03	0.325796E-01	0.234000E+01	0.121983E+03	0.320569E-01
0.235000E+01	0.121848E+03	0.315501E-01	0.236000E+01	0.121714E+03	0.310587E-01	0.237000E+01	0.121582E+03	0.305821E-01
[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

```
0.241000E+01 0.121069E+03 0.288111E-01   0.242000E+01 0.120945E+03 0.283996E-01   0.243000E+01 0.120821E+03 0.279996E-01
0.244000E+01 0.120699E+03 0.276107E-01   0.245000E+01 0.120577E+03 0.272325E-01   0.246000E+01 0.120458F+03 0.268645E-01
```

DRIFT= 0.558032E+01

NOZZLE NO.	YNOZ	ZNOZ	DIAMN	STDEV	Q	DRIFT
1	0.4000	0.4000	200.0000	50.0000	1.0000	5.5803

SINGLE DROPLET LATERAL DISPLACEMENT

NOZZLE: 1

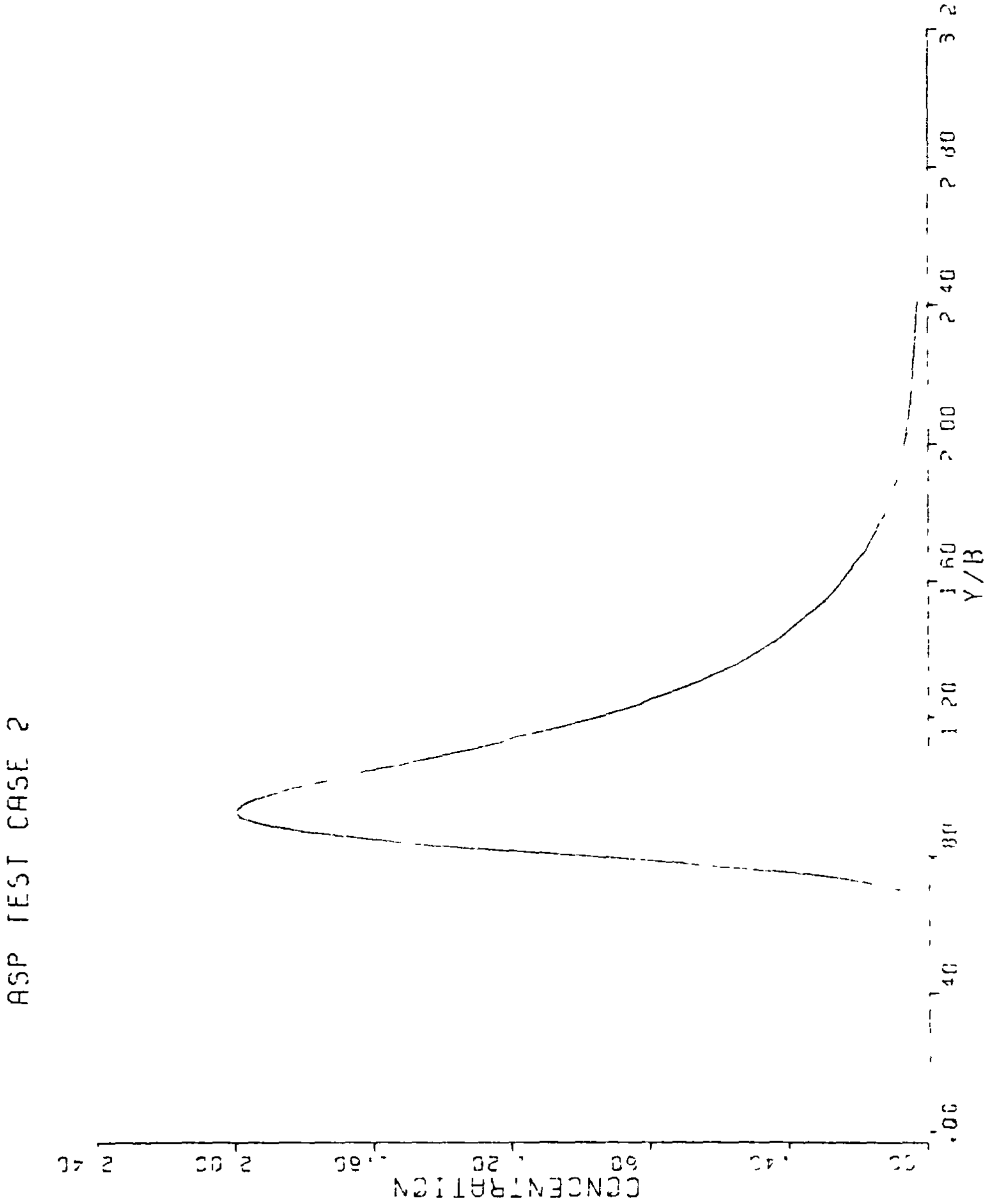
YG	DIA
0.567768E+00	0.600000E+03
0.785351E+00	0.300000E+03
0.889156E+00	0.250000E+03
0.107252E+01	0.200000E+03
0.149923F+01	0.150000E+03
0.191002E+01	0.130300E+03
0.246069E+01	0.120449E+03
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000F+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01

FINAL DISTRIBUTION

YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION
0.64	0.110564E-03	0.65	0.408700E-03	0.66	0.126168E-02	0.67	0.334558E-02	0.68	0.779777E-02
0.69	0.162809E-01	0.70	0.309327E-01	0.71	0.541859E-01	0.72	0.884828E-01	0.73	0.135946E+00
0.74	0.198073E+00	0.75	0.275513E+00	0.76	0.367965E+00	0.77	0.474192E+00	0.78	0.592145E+00
0.79	0.719134E+00	0.80	0.851925E+00	0.81	0.987170E+00	0.82	0.112161E+01	0.83	0.125221E+01
0.84	0.137634E+01	0.85	0.149180E+01	0.86	0.159689E+01	0.87	0.169039E+01	0.88	0.177154E+01
0.89	0.184000E+01	0.90	0.189559E+01	0.91	0.193848E+01	0.92	0.196920E+01	0.93	0.198850E+01
0.94	0.199723E+01	0.95	0.199634E+01	0.96	0.198681E+01	0.97	0.196964E+01	0.98	0.194580E+01
0.99	0.191622E+01	1.00	0.188178E+01	1.01	0.184330E+01	1.02	0.180153E+01	1.03	0.175716E+01
1.04	0.171079E+01	1.05	0.166297E+01	1.06	0.161418E+01	1.07	0.156483E+01	1.08	0.151536E+01
1.09	0.146626E+01	1.10	0.141774E+01	1.11	0.136998E+01	1.12	0.132313E+01	1.13	0.127730E+01
1.14	0.123259E+01	1.15	0.118905E+01	1.16	0.114675E+01	1.17	0.110572E+01	1.18	0.106597E+01
1.19	0.102752E+01	1.20	0.990365E+00	1.21	0.954498E+00	1.22	0.919905E+00	1.23	0.886566E+00
1.24	0.854457E+00	1.25	0.823551E+00	1.26	0.793816E+00	1.27	0.765222E+00	1.28	0.737734E+00
1.29	0.711317E+00	1.30	0.685937E+00	1.31	0.661557E+00	1.32	0.638143E+00	1.33	0.615658E+00
1.34	0.594069E+00	1.35	0.573341E+00	1.36	0.553441E+00	1.37	0.534334E+00	1.38	0.515991E+00
1.39	0.498379E+00	1.40	0.481469E+00	1.41	0.465232E+00	1.42	0.449639E+00	1.43	0.434664E+00
1.44	0.420280E+00	1.45	0.406463E+00	1.46	0.393188E+00	1.47	0.380433E+00	1.48	0.368174E+00
1.49	0.356392E+00	1.50	0.345064E+00	1.51	0.334066E+00	1.52	0.323321E+00	1.53	0.312838E+00
1.54	0.302624E+00	1.55	0.292683E+00	1.56	0.283019E+00	1.57	0.273632E+00	1.58	0.264523E+00
1.59	0.255689E+00	1.60	0.247128E+00	1.61	0.238838E+00	1.62	0.230813E+00	1.63	0.223049E+00
1.64	0.215542E+00	1.65	0.208284E+00	1.66	0.201271E+00	1.67	0.194496E+00	1.68	0.187953E+00
1.69	0.181636E+00	1.70	0.175537E+00	1.71	0.169650E+00	1.72	0.163969E+00	1.73	0.158487E+00
1.74	0.153197E+00	1.75	0.148094E+00	1.76	0.143170E+00	1.77	0.138419E+00	1.78	0.133837E+00
1.79	0.129415E+00	1.80	0.125150E+00	1.81	0.121034E+00	1.82	0.117063E+00	1.83	0.113232E+00
1.84	0.109534E+00	1.85	0.105966E+00	1.86	0.102522E+00	1.87	0.991974E-01	1.88	0.959882E-01
1.89	0.928897E-01	1.90	0.898979E-01	1.91	0.870086E-01	1.92	0.842382E-01	1.93	0.816006E-01
1.94	0.790880E-01	1.95	0.766933E-01	1.96	0.744098E-01	1.97	0.722311E-01	1.98	0.701514E-01
1.99	0.681652E-01	2.00	0.662675E-01	2.01	0.644532E-01	2.02	0.627181E-01	2.03	0.610577E-01
2.04	0.594682E-01	2.05	0.579458E-01	2.06	0.564870E-01	2.07	0.550886E-01	2.08	0.537473E-01
2.09	0.524604E-01	2.10	0.512251E-01	2.11	0.500387E-01	2.12	0.488990E-01	2.13	0.478034E-01
2.14	0.467500E-01	2.15	0.457366E-01	2.16	0.447614E-01	2.17	0.438225E-01	2.18	0.429181E-01
2.19	0.420468E-01	2.20	0.412068E-01	2.21	0.403969E-01	2.22	0.396155E-01	2.23	0.388614E-01
2.24	0.381334E-01	2.25	0.374303E-01	2.26	0.367510E-01	2.27	0.360944E-01	2.28	0.354595E-01
2.29	0.348454E-01	2.30	0.342512E-01	2.31	0.336761E-01	2.32	0.331191E-01	2.33	0.325796E-01
2.34	0.320569E-01	2.35	0.315501E-01	2.36	0.310587E-01	2.37	0.305821E-01	2.38	0.301195E-01
2.39	0.296705E-01	2.40	0.292346E-01	2.41	0.288111E-01	2.42	0.283996E-01	2.43	0.279996E-01
2.44	0.276107E-01	2.45	0.272325E-01	2.46	0.268645E-01				

TOTAL DRIFT= 0.558032E+01

TOTAL= 0.944312E+02
BAKGND: STOP


ASP TEST CASE 2
CONCENTRATION
Y/B

OSU / AARL

AERIAL SPRAY PROGRAM

LAST UPDATE 2/1/81

ASP TEST CASE 3

PROGRAM CONTROL

N2D= 1	N3D= 0	NEVAP= 1	NPROP2= 0	NCW= 0
NTUN= 0	NDIST= 1	NPRINT= 0	NPLOT= 1	EPS= 0.100000E-04

PARAMETER VALUES

A= 6.0000	CL= 0.6000	B= 6.0960	U= 53.0350	G= 9.8066
ZY0= 0.8000	Z70= 0.5000	DA= 0.118901E+01	VIS= 0.181330E-04	VCW= 0.0000
D= 0.0000	DD= 1000.0000			

EVAPORATION INPUTS

PA= 101352.10 TDB= 23.888900 TWB= 18.333300

DISTRIBUTION INPUTS

NCOL= 1	NROW= 6	NDIAMN= 1	NSTDEV= 1	NQ= 1
NZNOZ= 1	DD=1000.00000	X= 0.00000	UD= 0.00000	VD= 0.00000
WD= 0.00000	SWIDTH= 4.00000	DERR= 1.00000	DWIDTH= 5.00000	

NOZZLE NO.	YNOZ	ZNOZ	DIAMN	STDEV	Q	DRIFT
1	0.4000	0.4000	200.0000	50.0000	1.0000	21.2474

SINGLE DROPLET LATERAL DISPLACEMENT

NOZZLE: 1

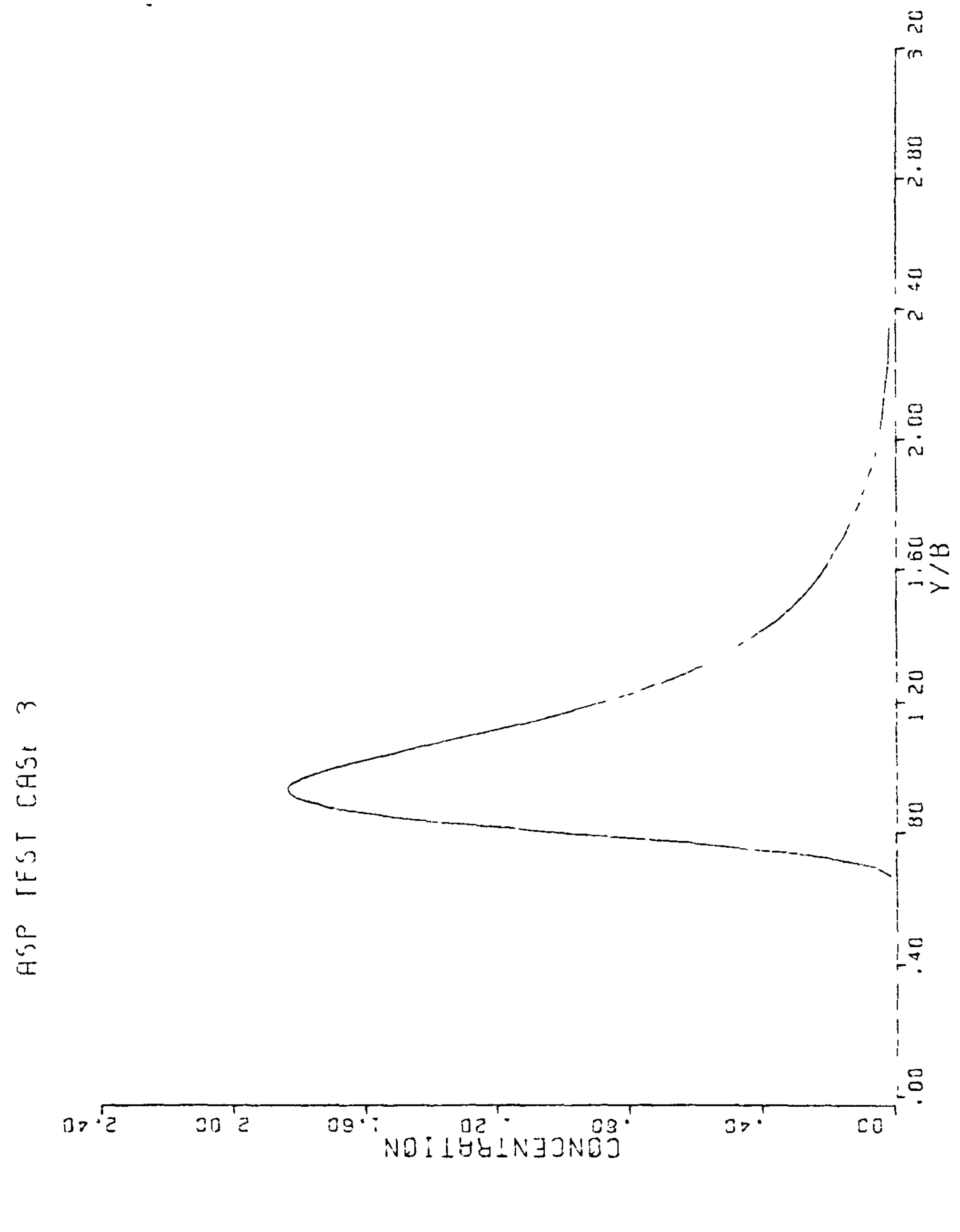
YG	DIA	DIAG
0.566534E+00	0.600000E+03	0.598786E+03
0.786095E+00	0.300000E+03	0.297565E+03
0.892304E+00	0.250000E+03	0.247031E+03
0.108410E+01	0.200000E+03	0.196103E+03
0.155559E+01	0.150000E+03	0.143102E+03
0.204485E+01	0.131864E+03	0.119342E+03
0.235994E+01	0.127329E+03	0.109949E+03
0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01

FINAL DISTRIBUTION

YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION
0.64	0.124109E-03	0.65	0.445961E-03	0.66	0.134425E-02	0.67	0.349339E-02	0.68	0.800428E-02
0.69	0.164709E-01	0.70	0.309088E-01	0.71	0.535747E-01	0.72	0.866979E-01	0.73	0.132177E+00
0.74	0.191308E+00	0.75	0.264595E+00	0.76	0.351662E+00	0.77	0.451289E+00	0.78	0.561524E+00
0.79	0.679842E+00	0.80	0.803113E+00	0.81	0.928154E+00	0.82	0.105193E+01	0.83	0.117164E+01
0.84	0.128490E+01	0.85	0.138974E+01	0.86	0.148466E+01	0.87	0.156864E+01	0.88	0.164107E+01
0.89	0.170174E+01	0.90	0.175078E+01	0.91	0.178863E+01	0.92	0.181580E+01	0.93	0.183294E+01
0.94	0.184078E+01	0.95	0.184014E+01	0.96	0.183185E+01	0.97	0.181677E+01	0.98	0.179573E+01
0.99	0.176951E+01	1.00	0.173887E+01	1.01	0.170452E+01	1.02	0.166711E+01	1.03	0.162724E+01
1.04	0.158543E+01	1.05	0.154217E+01	1.06	0.149790E+01	1.07	0.145297E+01	1.08	0.140773E+01
1.09	0.136244E+01	1.10	0.131730E+01	1.11	0.127254E+01	1.12	0.122836E+01	1.13	0.118492E+01
1.14	0.114237E+01	1.15	0.110081E+01	1.16	0.106032E+01	1.17	0.102097E+01	1.18	0.982809E+00
1.19	0.945863E+00	1.20	0.910151E+00	1.21	0.875679E+00	1.22	0.842445E+00	1.23	0.810440E+00
1.24	0.779645E+00	1.25	0.750041E+00	1.26	0.721601E+00	1.27	0.694295E+00	1.28	0.668093E+00
1.29	0.642961E+00	1.30	0.618865E+00	1.31	0.595770E+00	1.32	0.573638E+00	1.33	0.552436E+00
1.34	0.532126E+00	1.35	0.512674E+00	1.36	0.494045E+00	1.37	0.476205E+00	1.38	0.459121E+00
1.39	0.442760E+00	1.40	0.427091E+00	1.41	0.412084E+00	1.42	0.397710E+00	1.43	0.383941E+00
1.44	0.370749E+00	1.45	0.358108E+00	1.46	0.345994E+00	1.47	0.334383E+00	1.48	0.323252E+00
1.49	0.312579E+00	1.50	0.302342E+00	1.51	0.292523E+00	1.52	0.283103E+00	1.53	0.274062E+00
1.54	0.265384E+00	1.55	0.257052E+00	1.56	0.249039E+00	1.57	0.241244E+00	1.58	0.233645E+00
1.59	0.226239E+00	1.60	0.219023E+00	1.61	0.211994E+00	1.62	0.205151E+00	1.63	0.198490E+00
1.64	0.192008E+00	1.65	0.185704E+00	1.66	0.179573E+00	1.67	0.173614E+00	1.68	0.167824E+00
1.69	0.162200E+00	1.70	0.156739E+00	1.71	0.151439E+00	1.72	0.146296E+00	1.73	0.141308E+00
1.74	0.136471E+00	1.75	0.131783E+00	1.76	0.127240E+00	1.77	0.122840E+00	1.78	0.118579E+00
1.79	0.114455E+00	1.80	0.110463E+00	1.81	0.106601E+00	1.82	0.102866E+00	1.83	0.992536E-01
1.84	0.957618E-01	1.85	0.923869E-01	1.86	0.891259E-01	1.87	0.859754E-01	1.88	0.829325E-01
1.89	0.799939E-01	1.90	0.771566E-01	1.91	0.744176E-01	1.92	0.717737E-01	1.93	0.692221E-01
1.94	0.667599E-01	1.95	0.643842E-01	1.96	0.620922E-01	1.97	0.598811E-01	1.98	0.577482E-01
1.99	0.556909E-01	2.00	0.537067E-01	2.01	0.517930E-01	2.02	0.499474E-01	2.03	0.481675E-01
2.04	0.464509E-01	2.05	0.447997E-01	2.06	0.432341E-01	2.07	0.417527E-01	2.08	0.403504E-01
2.09	0.390227E-01	2.10	0.377650E-01	2.11	0.365733E-01	2.12	0.354437E-01	2.13	0.343724E-01
2.14	0.333562E-01	2.15	0.323918E-01	2.16	0.314761E-01	2.17	0.306064E-01	2.18	0.297799E-01
2.19	0.289942E-01	2.20	0.282470E-01	2.21	0.275361E-01	2.22	0.268593E-01	2.23	0.262148E-01
2.24	0.256007E-01	2.25	0.250153E-01	2.26	0.244571E-01	2.27	0.239245E-01	2.28	0.234160E-01
2.29	0.229304E-01	2.30	0.224664E-01	2.31	0.220229E-01	2.32	0.215986E-01	2.33	0.211926E-01
2.34	0.208038E-01	2.35	0.204314E-01						

TOTAL DRIFT= 0.212474E+02

TOTAL= 0.860481E+02
BAKGND: STOP


Y/B
CONCENTRATION

OSU / AARL

AERIAL SPRAY PROGRAM

LAST UPDATE 2/1/81

ASP TEST CASE 4

PROGRAM CONTROL

N2D= 1	N3D= 0	NEVAP= 0	NPROP2= 0	NCW= 0
NTUN= 1	NDIST= 1	NPRINT= 0	NPLOT= 1	EPS= 0.100000E-04

PARAMETER VALUES

A= 6.0000	CL= 0.6000	B= 6.0960	U= 53.0350	G= 9.8066
ZY0= 0.8000	ZZ0= 0.5000	DA= 0.122500E+01	VIS= 0.179320E-04	VCW= 0.0000
D= 36.5760	DD= 1000.0000			

DISTRIBUTION INPUTS

NCOL= 1	NROW= 6	NDIAMN= 1	NSTDEV= 1	NQ= 1
NZNOZ= 1	DD=1000.00000	X= 0.00000	UD= 0.00000	VD= 0.00000
WD= 0.00000	SWIDTH= 4.00000	DERR= 1.00000	DWIDTH= 5.00000	

NOZZLE NO.	YNOZ	ZNOZ	DIAMN	STDEV	Q	DRIFT
1	0.4000	0.4000	200.0000	50.0000	1.0000	5.5777

SINGLE DROPLET LATERAL DISPLACEMENT

NOZZLE: 1

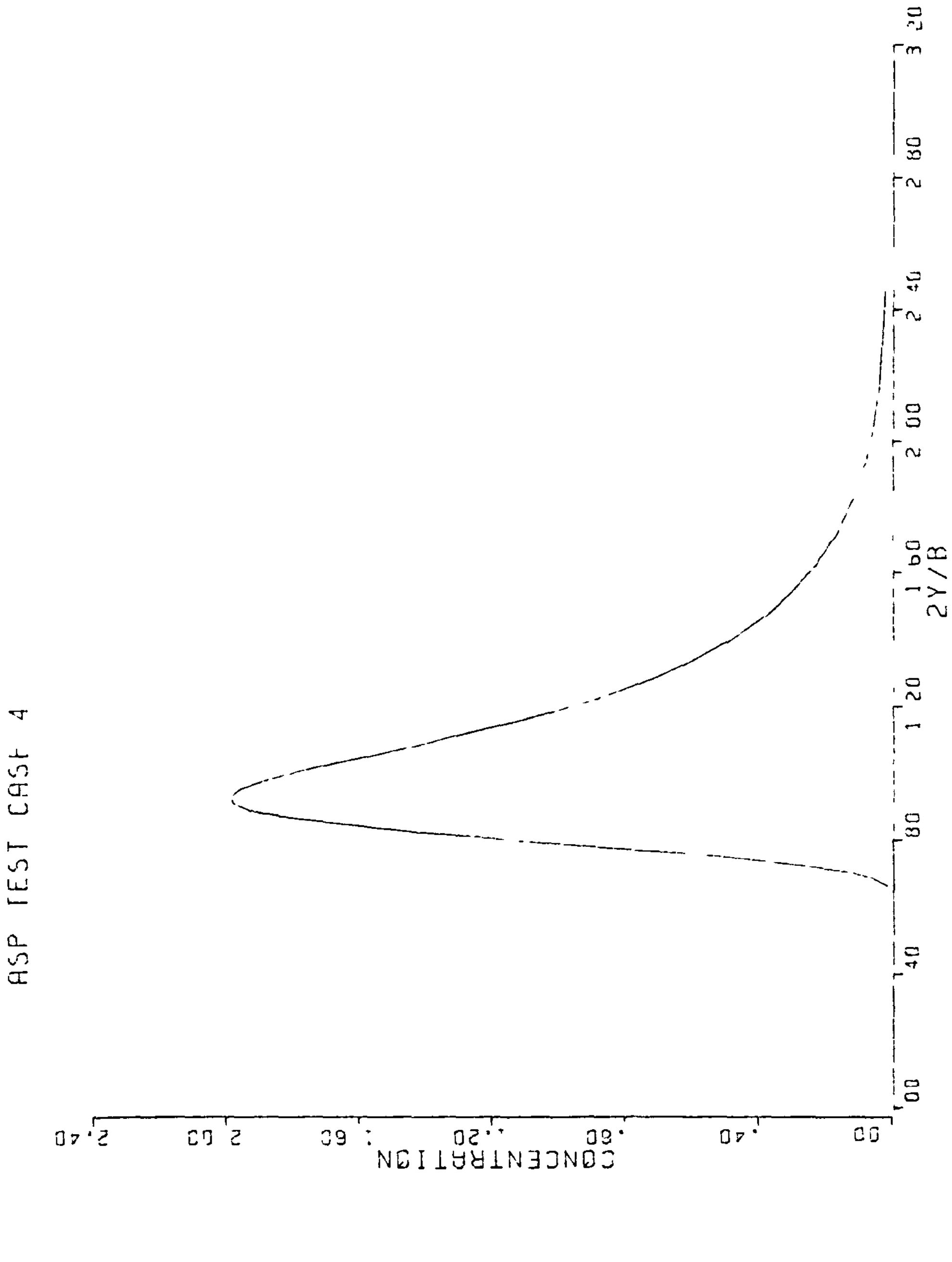
YG	DIA
0.567303E+00	0.600000E+03
0.785011E+00	0.300000E+03
0.888992E+00	0.250000E+03
0.107310E+01	0.200000E+03
0.150001E+01	0.150000E+03
0.191081E+01	0.130292E+03
0.248190E+01	0.120437E+03
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01

FINAL DISTRIBUTION

YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION
0.64	0.117847E-03	0.65	0.431420E-03	0.66	0.132101E-02	0.67	0.347890E-02	0.68	0.806143E-02
0.69	0.167483E-01	0.70	0.316871E-01	0.71	0.553080E-01	0.72	0.900375E-01	0.73	0.137970E+00
0.74	0.200565E+00	0.75	0.278435E+00	0.76	0.371242E+00	0.77	0.477720E+00	0.78	0.595802E+00
0.79	0.722783E+00	0.80	0.855396E+00	0.81	0.990285E+00	0.82	0.112419E+01	0.83	0.125410E+01
0.84	0.137738E+01	0.85	0.149188E+01	0.86	0.159591E+01	0.87	0.168829E+01	0.88	0.176829E+01
0.89	0.183559E+01	0.90	0.189011E+01	0.91	0.193209E+01	0.92	0.196208E+01	0.93	0.198083E+01
0.94	0.198918E+01	0.95	0.198809E+01	0.96	0.197851E+01	0.97	0.196144E+01	0.98	0.193781E+01
0.99	0.190854E+01	1.00	0.187451E+01	1.01	0.183650E+01	1.02	0.179526E+01	1.03	0.175145E+01
1.04	0.170567E+01	1.05	0.165845E+01	1.06	0.161025E+01	1.07	0.156150E+01	1.08	0.151260E+01
1.09	0.146403E+01	1.10	0.141601E+01	1.11	0.136872E+01	1.12	0.132230E+01	1.13	0.127688E+01
1.14	0.123253E+01	1.15	0.118933E+01	1.16	0.114733E+01	1.17	0.110657E+01	1.18	0.106707E+01
1.19	0.102884E+01	1.20	0.991886E+00	1.21	0.956193E+00	1.22	0.921753E+00	1.23	0.888547E+00
1.24	0.856552E+00	1.25	0.825743E+00	1.26	0.796091E+00	1.27	0.767564E+00	1.28	0.740131E+00
1.29	0.713757E+00	1.30	0.688409E+00	1.31	0.664052E+00	1.32	0.640652E+00	1.33	0.618174E+00
1.34	0.596584E+00	1.35	0.575848E+00	1.36	0.555935E+00	1.37	0.536811E+00	1.38	0.518445E+00
1.39	0.500807E+00	1.40	0.483868E+00	1.41	0.467598E+00	1.42	0.451970E+00	1.43	0.436958E+00
1.44	0.422535E+00	1.45	0.408677E+00	1.46	0.395361E+00	1.47	0.382563E+00	1.48	0.370261E+00
1.49	0.358435E+00	1.50	0.347063E+00	1.51	0.336030E+00	1.52	0.325237E+00	1.53	0.314697E+00
1.54	0.304418E+00	1.55	0.294405E+00	1.56	0.284664E+00	1.57	0.275197E+00	1.58	0.266004E+00
1.59	0.257084E+00	1.60	0.248437E+00	1.61	0.240058E+00	1.62	0.231946E+00	1.63	0.224095E+00
1.64	0.216501E+00	1.65	0.209158E+00	1.66	0.202061E+00	1.67	0.195204E+00	1.68	0.188580E+00
1.69	0.182184E+00	1.70	0.176009E+00	1.71	0.170048E+00	1.72	0.164295E+00	1.73	0.158743E+00
1.74	0.153386E+00	1.75	0.148218E+00	1.76	0.143231E+00	1.77	0.138420E+00	1.78	0.133779E+00
1.79	0.129302E+00	1.80	0.124982E+00	1.81	0.120815E+00	1.82	0.116794E+00	1.83	0.112915E+00
1.84	0.109171E+00	1.85	0.105559E+00	1.86	0.102073E+00	1.87	0.987078E-01	1.88	0.954596E-01
1.89	0.923239E-01	1.90	0.892963E-01	1.91	0.863727E-01	1.92	0.835671E-01	1.93	0.808973E-01
1.94	0.783554E-01	1.95	0.759341E-01	1.96	0.736265E-01	1.97	0.714261E-01	1.98	0.693269E-01
1.99	0.673232E-01	2.00	0.654097E-01	2.01	0.635815E-01	2.02	0.618341E-01	2.03	0.601629E-01
2.04	0.585640E-01	2.05	0.570334E-01	2.06	0.555678E-01	2.07	0.541635E-01	2.08	0.528176E-01
2.09	0.515269E-01	2.10	0.502887E-01	2.11	0.491003E-01	2.12	0.479593E-01	2.13	0.468633E-01
2.14	0.458100E-01	2.15	0.447975E-01	2.16	0.438237E-01	2.17	0.428867E-01	2.18	0.419848E-01
2.19	0.411164E-01	2.20	0.402798E-01	2.21	0.394737E-01	2.22	0.386965E-01	2.23	0.379469E-01
2.24	0.372237E-01	2.25	0.365258E-01	2.26	0.358519E-01	2.27	0.352009E-01	2.28	0.345720E-01
2.29	0.339640E-01	2.30	0.333761E-01	2.31	0.328075E-01	2.32	0.322572E-01	2.33	0.317246E-01
2.34	0.312088E-01	2.35	0.307091E-01	2.36	0.302249E-01	2.37	0.297556E-01	2.38	0.293004E-01
2.39	0.288590E-01	2.40	0.284305E-01	2.41	0.280147E-01	2.42	0.276109E-01	2.43	0.272187E-01
2.44	0.268375E-01	2.45	0.264671E-01	2.46	0.261069E-01	2.47	0.257566E-01	2.48	0.254158E-01

TOTAL DRIFT= 0.557766E+01

TOTAL= 0.944302E+02
BAKGND: STOP



ASP TEST CASE 4
CONCENTRATION
.00
.40
.80
1.20
1.60
2.00
2.40
2Y/B
.00
.40
.80
1.20
1.60
2.00
2.40
2.80
3.20

OSU / AARL

AERIAL SPRAY PROGRAM

LAST UPDATE 2/1/81

ASP TEST CASE 5

PROGRAM CONTROL

N2D= 0	N3D= 1	NEVAP= 1	NPROP2= 1	NCW= 1
NTUN= 0	NDIST= 1	NPRINT= 0	NPLOT= 1	EPS= 0.100000E-04

PARAMETER VALUES

A= 6.0000	CL= 0.6000	B= 6.0960	U= 53.0350	G= 9.8066
ZY0= 0.8000	ZZ0= 0.5000	DA= 0.118901E+01	VIS= 0.181330E-04	VCW= 1.0000
D= 0.0000	DD= 1000.0000			

EVAPORATION INPUTS

PA= 101352.10	TDB= 23.888900	TWB= 18.333300

PROP2 INPUTS

ZPROP2= 0.5000	PK1= 1.7930	PK2= 17.9300	PK3= 0.0000	PK4= 6.0290
RP= 0.2000				

DISTRIBUTION INPUTS

NCOL= 3	NROW= 6	NDIAMN= 1	NSTDEV= 1	NQ= 1
NZNOZ= 1	DD=1000.00000	X= 0.00000	UD= 0.00000	VD= 0.00000
WD= 0.00000	SWIDTH= 4.00000	DFRR= 1.00000	DWIDTH= 5.00000	

NOZZLE NO.	YNOZ	ZNOZ	DIAMN	STDEV	Q	DRIFT
1	-0.3000	0.4000	200.0000	50.0000	1.0000	7.1442
2	0.0000	0.4000	200.0000	50.0000	1.0000	13.0321
3	0.3000	0.4000	200.0000	50.0000	1.0000	9.2220

SINGLE DROPLET LATERAL DISPLACEMENT

NOZZLE:	1			2			3	
YG	DIA	DIAG	YG	DIA	DIAG	YG	DIA	DIAG
-0.320151E+00	0.600000E+03	0.598080E+03	-0.876441E-01	0.600000E+03	0.597780E+03	0.544745E+00	0.600000E+03	0.598063E+03
-0.407045E+00	0.300000E+03	0.297088E+03	-0.515855E-01	0.300000E+03	0.296151E+03	0.708199E+00	0.300000E+03	0.297044E+03
-0.459665E+00	0.250000E+03	0.246594E+03	-0.539186E-01	0.275000E+03	0.270796E+03	0.781975E+00	0.250000E+03	0.246526E+03
-0.557848E+00	0.200000E+03	0.195780E+03	-0.593184E-01	0.250000E+03	0.245340E+03	0.896006E+00	0.200000E+03	0.195665E+03
-0.783947E+00	0.150000E+03	0.144136E+03	-0.842100E-01	0.200000E+03	0.193927E+03	0.109198E+01	0.150000E+03	0.144039E+03
-0.147024E+01	0.100000E+03	0.740140E+02	-0.139088E+00	0.150000E+03	0.140996E+03	0.152290E+01	0.100000E+03	0.900272E+02
-0.151925E+01	0.984375E+02	0.632314E+02	-0.172918E+00	0.125000E+03	0.112875E+03	0.209856E+01	0.770271E+02	0.613805E+02
0.000000E+01	0.000000E+01	0.000000E+01	0.321932E+00	0.100000E+03	0.710852E+02	0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01	0.325157E+00	0.725000E+02	0.451472E+02	0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01	0.704499E-01	0.690625E+02	0.157827E+02	0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01
0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01	0.000000E+01

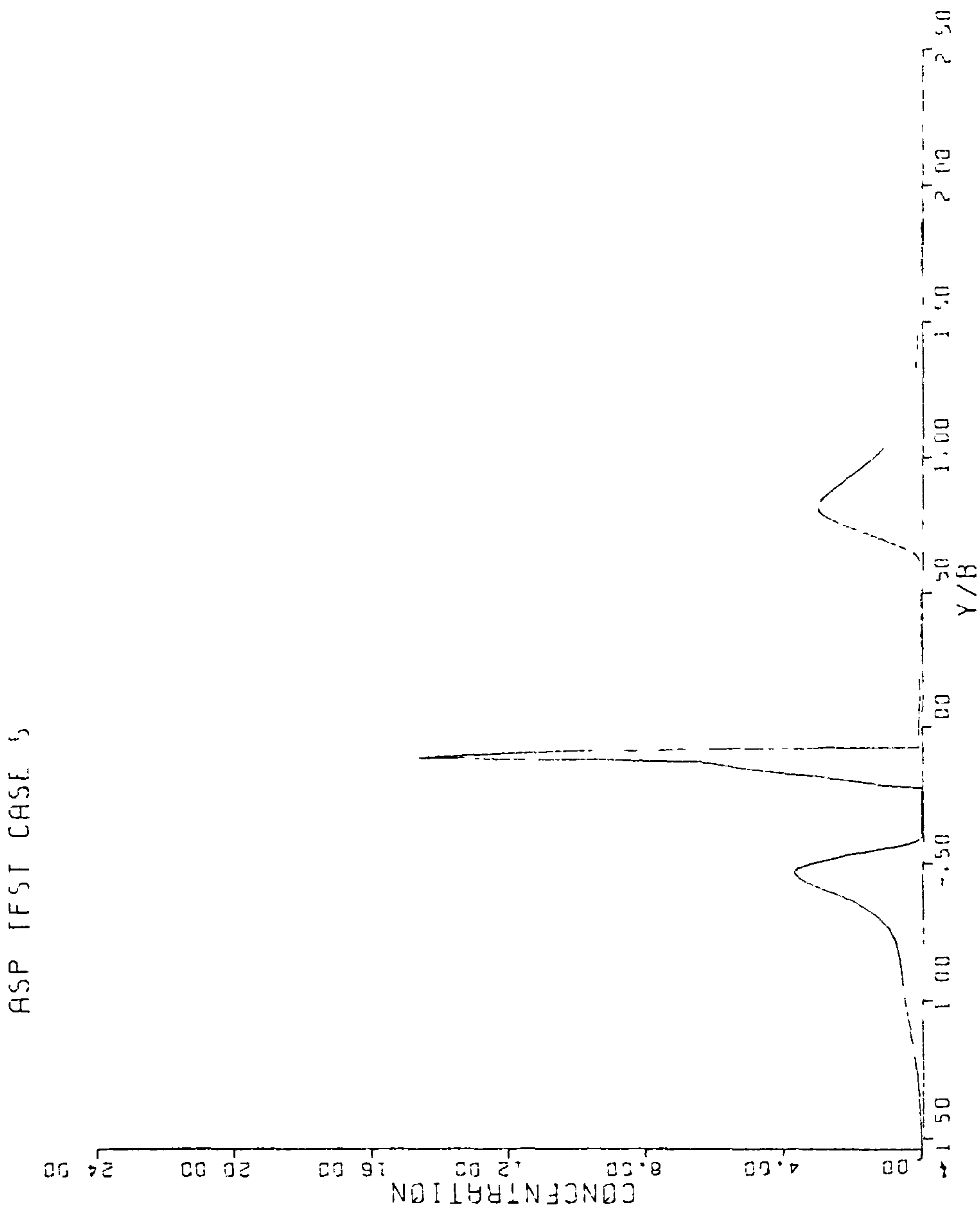
FINAL DISTRIBUTION

YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION	YG	CONCENTRATION
-1.51	0.919440E-02	-1.50	0.103502E-01	-1.49	0.116264E-01	-1.48	0.130289E-01	-1.47	0.145629E-01
-1.46	0.162340E-01	-1.45	0.180510E-01	-1.44	0.200232E-01	-1.43	0.221601E-01	-1.42	0.244715E-01
-1.41	0.269678E-01	-1.40	0.296594E-01	-1.39	0.325572E-01	-1.38	0.356725E-01	-1.37	0.390167E-01
-1.36	0.426016E-01	-1.35	0.464390E-01	-1.34	0.505412E-01	-1.33	0.549204E-01	-1.32	0.595890E-01
-1.31	0.645596E-01	-1.30	0.698445E-01	-1.29	0.754561E-01	-1.28	0.814066E-01	-1.27	0.877078E-01
-1.26	0.943714E-01	-1.25	0.101408E+00	-1.24	0.108829E+00	-1.23	0.116644E+00	-1.22	0.124861E+00
-1.21	0.133489E+00	-1.20	0.142533E+00	-1.19	0.152000E+00	-1.18	0.161892E+00	-1.17	0.172212E+00
-1.16	0.182959E+00	-1.15	0.194131E+00	-1.14	0.205723E+00	-1.13	0.217726E+00	-1.12	0.230131E+00
-1.11	0.242923E+00	-1.10	0.256085E+00	-1.09	0.269595E+00	-1.08	0.283428E+00	-1.07	0.297556E+00
-1.06	0.311945E+00	-1.05	0.326556E+00	-1.04	0.341349E+00	-1.03	0.356277E+00	-1.02	0.371289E+00
-1.01	0.386331E+00	-1.00	0.401346E+00	-0.99	0.416273E+00	-0.98	0.431049E+00	-0.97	0.445610E+00
-0.96	0.459892E+00	-0.95	0.473831E+00	-0.94	0.487368E+00	-0.93	0.500447E+00	-0.92	0.513020E+00
-0.91	0.525048E+00	-0.90	0.536505E+00	-0.89	0.547382E+00	-0.88	0.557687E+00	-0.87	0.567455E+00
-0.86	0.576747E+00	-0.85	0.585659E+00	-0.84	0.594326E+00	-0.83	0.602928E+00	-0.82	0.611698E+00
-0.81	0.620931E+00	-0.80	0.630991E+00	-0.79	0.642328E+00	-0.78	0.655514E+00	-0.77	0.671325E+00
-0.76	0.690124E+00	-0.75	0.712217E+00	-0.74	0.737946E+00	-0.73	0.767698E+00	-0.72	0.801901E+00
-0.71	0.841038E+00	-0.70	0.885644E+00	-0.69	0.936312E+00	-0.68	0.993698E+00	-0.67	0.105852E+01
-0.66	0.113158E+01	-0.65	0.121374E+01	-0.64	0.130592E+01	-0.63	0.140916E+01	-0.62	0.152453E+01
-0.61	0.165320E+01	-0.60	0.179643E+01	-0.59	0.195549E+01	-0.58	0.213168E+01	-0.57	0.232615E+01
-0.56	0.253965E+01	-0.55	0.276672E+01	-0.54	0.299187E+01	-0.53	0.320669E+01	-0.52	0.340104E+01
-0.51	0.356281E+01	-0.50	0.367833E+01	-0.49	0.373321E+01	-0.48	0.371375E+01	-0.47	0.360894E+01
-0.46	0.341294E+01	-0.45	0.313619E+01	-0.44	0.279241E+01	-0.43	0.238913E+01	-0.42	0.194197E+01
-0.41	0.147452E+01	-0.40	0.100968E+01	-0.39	0.581835E+00	-0.38	0.260214E+00	-0.37	0.796776E-01
-0.36	0.136499E-01	-0.35	0.938853E-03	-0.17	0.168032E+01	-0.16	0.221909E+01	-0.15	0.265141E+01
-0.14	0.302279E+01	-0.13	0.430308E+01	-0.12	0.496400E+01	-0.11	0.561741E+01	-0.10	0.618396E+01
-0.09	0.657872E+01	-0.08	0.146893E+02	-0.07	0.128646E+02	-0.06	0.969981E+01	-0.05	0.625907E-01
-0.04	0.601423E-01	-0.03	0.577604E-01	-0.02	0.554479E-01	-0.01	0.532070E-01	0.00	0.510397E-01
0.01	0.489476E-01	0.02	0.469318E-01	0.03	0.449931E-01	0.04	0.431322E-01	0.05	0.413491E-01
0.06	0.396439E-01	0.07	0.380163E-01	0.08	0.365174E-01	0.09	0.350553E-01	0.10	0.336704E-01
0.11	0.323619E-01	0.12	0.311288E-01	0.13	0.299698E-01	0.14	0.288839E-01	0.15	0.278698E-01
0.16	0.269261E-01	0.17	0.260517E-01	0.18	0.252451E-01	0.19	0.245051E-01	0.20	0.238307E-01
0.21	0.232205E-01	0.22	0.226735E-01	0.23	0.221889E-01	0.24	0.217659E-01	0.25	0.214037E-01
0.26	0.211019E-01	0.27	0.208601E-01	0.28	0.206784E-01	0.29	0.205569E-01	0.30	0.204959E-01
0.31	0.204961E-01	0.32	0.205586E-01	0.61	0.398603E-03	0.62	0.188731E-02	0.63	0.691297E-02
0.64	0.204101E-01	0.65	0.502728E-01	0.66	0.106330E+00	0.67	0.197834E+00	0.68	0.330412E+00
0.69	0.503837E+00	0.70	0.711515E+00	0.71	0.941824E+00	0.72	0.118517E+01	0.73	0.143373E+01
0.74	0.167951E+01	0.75	0.191594E+01	0.76	0.213787E+01	0.77	0.234145E+01	0.78	0.252389E+01
0.79	0.268062E+01	0.80	0.280446E+01	0.81	0.289454E+01	0.82	0.295156E+01	0.83	0.297733E+01
0.84	0.297448E+01	0.85	0.294617E+01	0.86	0.289584E+01	0.87	0.282701E+01	0.88	0.274313E+01
0.89	0.264744E+01	0.90	0.254306E+01	0.91	0.243368E+01	0.92	0.232165E+01	0.93	0.220864E+01
0.94	0.209605E+01	0.95	0.198500E+01	0.96	0.187638E+01	0.97	0.177087E+01	0.98	0.166897E+01
0.99	0.157106E+01	1.00	0.147738E+01	1.01	0.138806E+01	1.02	0.130318E+01	1.03	0.122273E+01
1.04	0.114665E+01	1.05	0.107486E+01	1.06	0.100723E+01	1.07	0.943617E+00	1.08	0.883859E+00
1.09	0.827786E+00	1.10	0.775062E+00	1.11	0.725256E+00	1.12	0.678293E+00	1.13	0.634094E+00
1.14	0.592563E+00	1.15	0.553596E+00	1.16	0.517082E+00	1.17	0.482905E+00	1.18	0.450947E+00
1.19	0.421089E+00	1.20	0.393215E+00	1.21	0.367209E+00	1.22	0.342959E+00	1.23	0.320356E+00
1.24	0.299298E+00	1.25	0.279683E+00	1.26	0.261418E+00	1.27	0.244412E+00	1.28	0.228580E+00
1.29	0.213842E+00	1.30	0.200122E+00	1.31	0.187350E+00	1.32	0.175459E+00	1.33	0.164388E+00
1.34	0.154078E+00	1.35	0.144474E+00	1.36	0.135528E+00	1.37	0.127192E+00	1.38	0.119422E+00
1.39	0.112178E+00	1.40	0.105422E+00	1.41	0.991196E-01	1.42	0.932381E-01	1.43	0.877476E-01
1.44	0.826202E-01	1.45	0.778301E-01	1.46	0.733534E-01	1.47	0.691679E-01	1.48	0.652532E-01
1.49	0.615902E-01	1.50	0.581613E-01	1.51	0.549501E-01	1.52	0.519417E-01	1.53	0.491255E-01
1.54	0.464975E-01	1.55	0.440436E-01	1.56	0.417504E-01	1.57	0.396055E-01	1.58	0.375977E-01
1.59	0.357167E-01	1.60	0.339532E-01	1.61	0.322984E-01	1.62	0.307446E-01	1.63	0.292843E-01
1.64	0.279110E-01	1.65	0.266185E-01	1.66	0.254011E-01	1.67	0.242538E-01	1.68	0.231716E-01

[illegible]	[illegible]	1.70	[illegible]	1.71	[illegible]	1.72	[illegible]	1.73	0.185959E-01
1.74	0.178233E-01	1.75	0.170913E-01	1.76	0.163973E-01	1.77	0.157389E-01	1.78	0.151139E-01
1.79	0.145204E-01	1.80	0.139563E-01	1.81	0.134200E-01	1.82	0.129097E-01	1.83	0.124241E-01
1.84	0.119615E-01	1.85	0.115208E-01	1.86	0.111006E-01	1.87	0.106998E-01	1.88	0.103174E-01
1.89	0.995225E-02	1.90	0.960347E-02	1.91	0.927016E-02	1.92	0.895150E-02	1.93	0.864670E-02
1.94	0.835502E-02	1.95	0.807580E-02	1.96	0.780837E-02	1.97	0.755215E-02	1.98	0.730654E-02
1.99	0.707104E-02	2.00	0.684512E-02	2.01	0.662831E-02	2.02	0.642017E-02	2.03	0.622028E-02
2.04	0.602825E-02	2.05	0.584368E-02	2.06	0.566624E-02	2.07	0.549559E-02	2.08	0.533141E-02
2.09	0.517340E-02								

TOTAL DRIFT= 0.979943E+01

TOTAL= 0.873970E+02
BAKGND: STOP


ASP TEST CASE 5
CONCENTRATION
Y/B

APPENDIX B

FORTRAN COMPUTER PROGRAM LISTING

```
C
C        ASP  -   AERIAL SPRAY PROGRAM
C
C        ASP HAS BEEN DESIGNED TO PREDICT THE GROUND DEPOSITION
C        AND PERCENT OF MATERIAL LOST DUE TO DRIFT FROM AN
C        AGRICULTURAL AIRCRAFT.  THE PROGRAM IS WRITTEN TO INPUT
C        DATA IN SI UNITS AND INCLUDES THE FOLLOWING OPTIONS:
C            1) 2D OR 3D WAKE MODEL AND DROPLET DYNAMICS
C            2) A DROPLET EVAPORATION MODEL
C            3) A PROPELLER SLIPSTREAM MODEL
C            4) A CROSSWIND MODEL
C            5) A TUNNEL WALL MODEL
C            6) THE ABILITY FOR THE USER TO SUPPLY HIS OWN
C               FLOWFIELD MODEL.
C
C        DOCUMENTATION AVAILABLE IN THE FORM OF A NASA CONTRACTORS
C        REPORT
C
C        CODE WRITTEN AT THE AERONAUTICAL AND ASTRONAUTICAL ENGINEERING
C        RESEARCH LABORATORY,  THE OHIO STATE UNIVERSITY,  COLUMBUS,
C        OHIO  1981    AUTHOR  MICHAEL B. BRAGG
C
C        LAST UPDATE 2/1/81
C
C
CDC   PROGRAM MAIN(INPUT,OUTPUT,TAPE8,TAPE5=INPUT,TAPE6=OUTPUT)
      DIMENSION YNOZ(100),SIZE(15),DIST(1001),DRIFT(100),NRUN(100)
      DIMENSION DIAMN(100),STDEV(100),Q(100),ZNOZ(100),DIAG(15,100)
      DIMENSION IRUN(50),TRAJ(15,100),SIZ(15,100)
     &          ,SLP(15),XLIST(5),DLIST(5)
      DIMENSION YY(13,13),SAVE(13,13),CSAVE(13,3),YMAX(13),MET(100)
      DIMENSION ERROR(13),DY(13),YY1(13),PW(13,13),IP(13),TITLE(13)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMMON /AREA2/ N2D,N3D,NEVAP,EPS,NPROP2,NCW,NTUN,NDIST,NPRINT
      COMMON /AREA3/ PSWB,PAE,PV,SCN,DV
      COMMON /AREA5/ ZPROP2,PK1,PK2,PK3,PK4,RP
      COMMON /AREA6/ JJ,IRUN,ICOUNT
      COMMON /AREA7/ NVAR
      COMMON /AREA10/ TRAJ,DIAG,SIZ
C
C        INPUT THE NECESSARY PARAMETERS ACCORDING TO THE OPTIONS
C        SELECTED AND OUTPUT HEADER AND TITLE
C
  800 CONTINUE
      REWIND 8
      READ(5,3,END=700) (TITLE(K),K=1,13)
    3 FORMAT(13A6)
      WRITE(6,5) (TITLE(K),K=1,13)
    5 FORMAT("1",T50,"OSU / AARL"/"0",T50,
     &       "AERIAL SPRAY PROGRAM"/"0",T50,
     &       "LAST UPDATE 2/1/81"/////"0",13A6//)
      DO 1 K=1,100
      DRIFT(K)=0.
      NRUN(K)=0
      DO 1 J=1,15
```

```
      SIZ(J,K)=0.
      DIAG(J,K)=0.
    1 TRAJ(J,K)=0.
      DO 2 K=1,1001
    2 DIST(K)=0.
      ICOUNT=1
      JJ=0
      READ(5,10)NDROPS
   10 FORMAT(I5)
      DO 210 I=1,NDROPS
      READ(5,20) N2D,N3D,NEVAP,NPROP2,NCW,NTUN,NDIST,
     &            NPRINT,NPLOT,EPS
   20 FORMAT(9I2,2X,E15.6)
      IF(NEVAP.EQ.1) READ(5,30) PA,TDB,TWB
   30 FORMAT(3F10.6)
      IF(NPROP2.EQ.1) READ(5,50) ZPROP2,PK1,PK2,PK3,PK4,RP
   50 FORMAT(6F10.4)
      READ(5,60) A,CL,B,U,G,ZY0,ZZ0
   60 FORMAT(7F10.4)
      READ(5,70) DA,VIS,VCW,D
   70 FORMAT(2E20.6,2F10.4)
      IF(NDIST.EQ.1) GO TO 90
      READ(5,80) DIA,DD,X,Y,Z,UD,VD,WD
   80 FORMAT(8F10.5)
      GO TO 120
   90 CONTINUE
      READ(5,95) NCOL,NROW,NDIAMN,NSTDEV,NQ,NZNOZ
   95 FORMAT(6I5)
      READ(5,100)DD,X0,UD0,VD0,WD0,SWIDTH,DERR,DWIDTH
  100 FORMAT(8F10.5)
      READ(5,110) (YNOZ(K),K=1,NCOL)
  110 FORMAT(8F10.5)
      READ(5,110) (SIZE(K),K=1,NROW)
      IF(NDIAMN.EQ.1) READ(5,110) (DIAMN(K),K=1,NCOL)
      IF(NDIAMN.EQ.0) READ(5,110) DIAMN1
      IF(NSTDEV.EQ.1) READ(5,110) (STDEV(K),K=1,NCOL)
      IF(NSTDEV.EQ.0) READ(5,110) STDEV1
      IF(NQ.EQ.1) READ(5,110) (Q(K),K=1,NCOL)
      IF(NQ.EQ.0) READ(5,110) Q1
      IF(NZNOZ.EQ.1) READ(5,110) (ZNOZ(K),K=1,NCOL)
      IF(NZNOZ.EQ.0) READ(5,110) ZNOZ1
      DO 115 K=1,NCOL
      IF(NDIAMN.EQ.0) DIAMN(K)=DIAMN1
      IF(NQ.EQ.0) Q(K)=Q1
      IF(NSTDEV.EQ.0) STDEV(K)=STDEV1
      IF(NZNOZ.EQ.0) ZNOZ(K)=ZNOZ1
  115 CONTINUE
  120 CONTINUE
      C=3.141592653589793
C
C        CALCULATE EVAPORATION PARAMETERS
C
      IF(NEVAP.EQ.0) GO TO 130
      TDBK=TDB+273.15
      TDBR=TDB*(9./5.)+32.+459.67
```

```
      TWBR=TWB*(9./5.)+32.+459.67
      PAE=PA*1.4504E-4
      PS=EXP(54.6329-12301.688/TDBR-5.16923*ALOG(TDBR))
      PSWB=EXP(54.6329-12301.688/TWBR-5.16923*ALOG(TWBR))
      PV=PSWB-((.2405*(PAE-PSWB))/(.62194*(1075.8965-.56983*
     &    (TWBR-491.69))))*(TDBR-TWBR)
      RH=PV/PS
      DA=((PAE*144.)/(G*3.280839*53.34*TDBR))*515.4
      DV=(5.28E-6*TDBK**1.88)*1.E-4
      VIS=1.795E-5*(TDBK/293.16)**1.5*((293.16+110.)/(TDBK+110.))
      SCN=(DV*DA)/VIS
  130 CONTINUE
C
C        DETERMINE NVAR, THE NUMBER OF SIMULTANEOUS FIRST ORDER
C        DIFFERENTIAL EQUATIONS TO BE SOLVED
C
      NVAR=4
      IF(N3D.NE.0) NVAR=6
      IF(NEVAP.EQ.1) NVAR=NVAR+1
      IF(N2D.EQ.2.OR.N3D.EQ.2) GO TO 138
      IF(NTUN.EQ.1) GO TO 136
      IF(NCW.EQ.1.AND.NPROP2.EQ.1) GO TO 132
      IF(NCW.EQ.1) GO TO 134
      IF(NPROP2.EQ.1) NVAR=NVAR+1
      GO TO 138
  132 NVAR=NVAR+6
      GO TO 138
  134 NVAR=NVAR+4
      GO TO 138
  136 NVAR=NVAR+2
      IF(NPROP2.EQ.1) NVAR=NVAR+1
      GO TO 138
  138 CONTINUE
C
C        OUTPUT THE CONSTANTS THAT WERE INPUT
C
      WRITE(6,150) N2D,N3D,NEVAP,NPROP2,NCW,NTUN,NDIST,NPRINT,
     &             NPLOT,EPS
  150 FORMAT("0","PROGRAM CONTROL",//" ",T5,"N2D=",I2,T25,"N3D=",I2,
     &       T45,"NEVAP=",I2,T65,"NPROP2=",I2,T85,"NCW=",I2/" "
     &       ,T5,"NTUN=",I2,T25,"NDIST=",I2,T45,"NPRINT=",I2,T65,
     &       "NPLOT=",I2,T85,"EPS=",E13.6/)
      WRITE(6,160) A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
  160 FORMAT("0","PARAMETER VALUES"//" ",T5,"A=",F10.4,T25,"CL=",
     &       F10.4,T45,"B=",F10.4,T65,"U=",F10.4,T85,"G=",F10.4/" ",
     &       T5,"ZY0=",F10.4,T25,"ZZ0=",F10.4,T45,"DA=",E13.6,T65,
     &       "VIS=",E13.6,T85,"VCW=",F10.4/" ",T5,"D=",F10.4,
     &       T25,"DD=",F10.4/)
      IF(NEVAP.EQ.1) WRITE(6,170) PA,TDB,TWB
  170 FORMAT("0","EVAPORATION INPUTS"//" ",T5,"PA=",F10.2,T25,
     &       "TDB=",F10.6,T45,"TWB=",F10.6/)
      IF(NPROP2.EQ.1) WRITE(6,190) ZPROP2,PK1,PK2,PK3,PK4,RP
  190 FORMAT("0","PROP2 INPUTS"//" ",T5,"ZPROP2=",F10.4,T25,
     &       "PK1=",F10.4,T45,"PK2=",F10.4,T65,"PK3=",F10.4
     &       ,T85,"PK4=",F10.4/" ",T5,"RP=",F10.4/)
```

```
C         CHANGE FROM RECTANGULAR TO ELLIPTICAL LOAD DISTRIBUTION
C         BY ADJUSTING CL
      CL=(4./C)*CL
      IF(NDIST.EQ.1) GO TO 220
      WRITE(6,195) X,Y,Z,UD,VD,WD,DIA
  195 FORMAT("0","INITIAL VALUES"//" ",T5,"X=",F10.6,T25,
     &        "Y=",F10.6,T45,"Z=",F10.6,T65,"UD=",F10.6,T85,
     &        "VD=",F10.6/" ",T5,"WD=",F10.6,T25,"DIA=",
     &        F10.4/)
C
C         CALL SUB2D OR SUB3D TO CALCULATE THE DROPLET TRAJECTORY
C
      IF(N2D.NE.0) CALL SUB2D(NVAR,Y,Z,VD,WD,T,DIA,YY,SAVE
     &             ,CSAVE,YMAX,ERROR,DY,YY1,PW,IP)
      IF(N3D.NE.0) CALL SUB3D(NVAR,X,Y,Z,UD,VD,WD,T,DIA,YY,SAVE
     &             ,CSAVE,YMAX,ERROR,DY,YY1,PW,IP)
      ICOUNT=ICOUNT+1
      WRITE(6,200) X,Y,Z,UD,VD,WD,DIA,T
  200 FORMAT("0","FINAL VALUES"//" ",T5,"X=",F10.6,T25,"Y=",
     &       F10.6,T45,"Z=",F10.6,T65,"UD=",F10.6,T85,"VD=",
     &       F10.6/" ",T5,"WD=",F10.6,T25,"DIA=",F10.4,T45,
     &       "T=",F10.6/)
  210 CONTINUE
      GO TO 320
  220 CONTINUE
C
C         THIS PART OF THE MAIN PROGRAM CONTROLS THE NDIST=1 CASE
C
      WRITE(6,225) NCOL,NROW,NDIAMN,NSTDEV,NQ,NZNOZ,DD,
     &             X0,UD0,VD0,WD0,SWIDTH,DERR,DWIDTH
  225 FORMAT("0","DISTRIBUTION INPUTS"//" ",T5,"NCOL=",I2,
     &       T25,"NROW=",I2,T45,"NDIAMN=",I2,T65,"NSTDEV=",I2,
     &       T85,"NQ=",I2/" ",T5,"NZNOZ=",I2,T25,"DD=",F10.5,
     &       T45,"X=",F10.5,T65,"UD=",F10.5,T85,"VD=",F10.5/
     &       " ",T5,"WD=",F10.5,T25,"SWIDTH=",F10.5,T45,"DERR=",
     &       F10.5,T65,"DWIDTH=",F10.5)
C
C         PERFORM ALL THE TRAJECTORIES REQUIRED TO CALCULATE THE
C         REQUIRED DISTRIBUTION
C
      DO 250 I=1,NCOL
      DO 240 J=1,NROW
      DIA=SIZE(J)
      DIA1=DIA
      K=K-1
      T=0.
      X=X0
      UD=UD0
      VD=VD0
      WD=WD0
      Z=ZNOZ(I)
      Y=YNOZ(I)
      IF(N2D.NE.0) CALL SUB2D(NVAR,Y,Z,VD,WD,T,DIA,YY,SAVE,
     &              CSAVE,YMAX,ERROR,DY,YY1,PW,IP)
      IF(N3D.NE.0) CALL SUB3D(NVAR,X,Y,Z,UD,VD,WD,T,DIA,YY,SAVE,
```

```
     &                CSAVE,YMAX,ERROR,DY,YY1,PW,IP)
      TRAJ(J,I)=Y
      DIAG(J,I)=DIA
      SIZ(J,I)=SIZE(J)
      IF(Z.NE.0.) GO TO 241
      NRUN(I)=NRUN(I)+1
      IF(Z.EQ.0.AND.ABS(Y).GT.DWIDTH) GO TO 241
  240 CONTINUE
C
C         THIS SECTION UP TO STMT NO 250 CONTROLS THE CALCULATION
C         OF ADDITIONAL TRAJECTORIES IN ORDER TO ESTIMATE THE DRIFT
C         AND INSERTS PARTICLES IN REGIONS OF A LARGE GRADIENT
C         DDIA/DYG
C
      KMISS=0
      SIZMIS=0.
  241 CONTINUE
      J=NRUN(I)+1
      MET(I)=0
      IF(J.EQ.1) DRIFT(I)=SIZ(J,I)
      IF(J.EQ.1) GO TO 250
      DSIZ1=ABS(DIAMN(I)-SIZ(J-1,I))
      K250=0
      GO TO 243
  234 CONTINUE
      IF(Z.EQ.0.AND.ABS(Y).GT.DWIDTH) GO TO 250
      IF((DSIZ1/STDEV(I)).LT.4.) GO TO 242
      DRIFT(I)=0.
      GO TO 250
  242 CONTINUE
C
C         DEFINE ACCEPTABLE ERROR IN TERMS OF STDEV
C
      ERR=.5*STDEV(I)
      IF(DSIZ1.LT.(2.*STDEV(I))) ERR=.30*STDEV(I)
      IF(DSIZ1.LT.STDEV(I)) ERR=.20*STDEV(I)
      ERR=ERR*DERR
  246 CONTINUE
      IF(Z.NE.0.) KMISS=1
      IF(Z.NE.0.) SIZMIS=SIZ(J,I)
      IF(J.GE.3) GO TO 239
      IF(J.EQ.2) SIZ(J,I)=SIZ(J-1,I)-(SIZ(J-1,I)-SIZMIS)/2.
      GO TO 245
  239 CONTINUE
C
C         DETERMINE PARTICLE SIZE FOR DRIFT ESTIMATE
C         AND CALCULATE TRAJECTORIES
C
      YGM1=TRAJ(J-2,I)
      YG=TRAJ(J-1,I)
      Y1=ABS(1./(YG-YNOZ(I)))
      Y2=ABS(1./(YGM1-YNOZ(I)))
      SLOPE=(SIZ(J-2,I)-SIZ(J-1,I))/(Y2-Y1)
      SIZ(J,I)=SIZ(J-1,I)-SLOPE*Y1/2
      IF((SLOPE*Y1/2.).LT.ERR) SIZ(J,I)=SIZ(J-1,I)-ERR
```

```
      IF(SIZ(J,I).GT.SIZMIS) GO TO 245
      SIZ(J,I)=SIZMIS+(SIZ(J-1,I)-SIZMIS)/2.
  245 CONTINUE
      DIA=SIZ(J,I)
      DIA1=DIA
      T=0.
      X=X0
      UD=UD0
      VD=VD0
      WD=WD0
      Z=ZNOZ(I)
      Y=YNOZ(I)
      IF(N2D.NE.0) CALL SUB2D(NVAR,Y,Z,VD,WD,T,DIA,YY,SAVE,
     &                CSAVE,YMAX,ERROR,DY,YY1,PW,IP)
      IF(N3D.NE.0) CALL SUB3D(NVAR,X,Y,Z,UD,VD,WD,T,DIA,YY,SAVE,
     &                CSAVE,YMAX,ERROR,DY,YY1,PW,IP)
C
C        STORE TRAJECTORY INFORMATION, IF ERROR.LT.ERR GO TO 250
C        OTHERWISE ITERATE
C
      TRAJ(J,I)=Y
      DIAG(J,I)=DIA
      IF(MET(I).EQ.1) GO TO 228
      IF(Z.NE.0.) GO TO 246
      NRUN(I)=NRUN(I)+1
      IF(ABS(Y).LT.DWIDTH) GO TO 330
      GO TO 250
  330 CONTINUE
      IF((SIZ(J,I)-SIZMIS).GT.ERR) GO TO 244
      DRIFT(I)=SIZ(J,I)
      K250=1
      GO TO 243
  244 CONTINUE
      J=J+1
      IF(J.LE.15) GO TO 246
      WRITE(6,247)
  247 FORMAT(" ","*** MAX NUMBER OF RUNS EXCEEDED BEFORE DRIFT",
     &       " CALCULATION COMPLETED")
      DRIFT(I)=SIZ(J,I)
      GO TO 250
  243 CONTINUE
C
C        THIS SECTION HANDLLES THE CASE WHERE DYG/DDIA CHANGES
C        SIGN BETWEEN POINTS
C
      IF(J.EQ.2) GO TO 227
      J2=J-2
      DO 410 K=1,J2
      SLP(K)=(TRAJ(K+1,I)-TRAJ(K,I))/(SIZ(K+1,I)-SIZ(K,I))
  410 CONTINUE
      DO 221 K=2,J2
      IF(SLP(K).GT.0.AND.SLP(K-1).LT.0.) GO TO 222
      IF(SLP(K).LT.0.AND.SLP(K-1).GT.0.) GO TO 222
  221 CONTINUE
      IF(K250.EQ.1) GO TO 250
```

```
      GO TO 234
  222 CONTINUE
C
C        INSERT PARTICLES WHERE DYG/DDIA CHANGES SIGN
C
      MET(I)=1
      IF(K250.EQ.1) GO TO 250
      IADD=0
      DO 223 K2=2,J2
      K=K2
      IF(SLP(K).LT.0.AND.SLP(K-1).LT.0.) GO TO 223
      IF(SLP(K).GT.0.AND.SLP(K-1).GT.0.) GO TO 223
      IF(IADD.NE.0) K=K2+IADD
      DELTA=ABS(SIZ(K,I)-DIAMN(I))/STDEV(I)
      ER=1.0*STDEV(I)
      IF(DELTA.LT.2.) ER=.75*STDEV(I)
      IF(DELTA.LT.1.) ER=.50*STDEV(I)
      ER=ER*DERR
      IDELTA=ABS(SIZ(K+1,I)-SIZ(K,I))/ER
      IADD=IADD+IDELTA
      DELTA=(SIZ(K+1,I)-SIZ(K,I))/(IDELTA+1)
      K1=K+1
      J1=J-1
      DO 224 K5=K1,J1
      KK=K1-K5+J1
      TRAJ(KK+IDELTA,I)=TRAJ(KK,I)
      DIAG(KK+IDELTA,I)=DIAG(KK,I)
      SIZ(KK+IDELTA,I)=SIZ(KK,I)
  224 CONTINUE
      DO 226 KK=1,IDELTA
      SIZ(K+KK,I)=SIZ(K,I)+KK*DELTA
      DIA=SIZ(K+KK,I)
      T=0.
      X=X0
      UD=UD0
      VD=VD0
      WD=WD0
      Z=ZNOZ(I)
      Y=YNOZ(I)
      IF(N2D.NE.0) CALL SUB2D(NVAR,Y,Z,VD,WD,T,DIA,YY,SAVE,
     &                 CSAVE,YMAX,ERROR,DY,YY1,PW,IP)
      IF(N3D.NE.0) CALL SUB3D(NVAR,X,Y,Z,UD,VD,WD,T,DIA,YY,SAVE,
     &                 CSAVE,YMAX,ERROR,DY,YY1,PW,IP)
      TRAJ(KK+K,I)=Y
      DIAG(KK+K,I)=DIA
      NRUN(I)=NRUN(I)+1
      J=J+1
  226 CONTINUE
      NRR=NRUN(I)
  223 CONTINUE
  227 CONTINUE
C
C        DETERMINE PARTICLE SIZE FOR DRIFT ESTIMATE AND
C        CALCULATE THE TRAJECTORY
C
```

```
      DSIZ1=ABS(DIAMN(I)-SIZ(J-1,I))
      IF((DSIZ1/STDEV(I)).LT.4) GO TO 237
      DRIFT(I)=0.
      GO TO 250
  237 CONTINUE
      ERR=.5*STDEV(I)
      IF(DSIZ1.LT.(2.*STDEV(I))) ERR=.30*STDEV(I)
      IF(DSIZ1.LT.STDEV(I)) ERR=.20*STDEV(I)
      ERR=ERR*DERR
  236 CONTINUE
      IF(KMISS.EQ.0) SIZ(J,I)=SIZ(J-1,I)-1.1*ERR
      IF(KMISS.EQ.1) SIZ(J,I)=SIZ(J-1,I)-(SIZ(J-1,I)-SIZMIS)/2.
      GO TO 245
  228 CONTINUE
C
C         IF PARTICLE WITHIN ERR GO TO 250 OTHERWISE ITERATE
C
      IF(Z.EQ.0.) GO TO 229
      KMISS=1
      SIZMIS=SIZ(J,I)
      GO TO 227
  229 CONTINUE
      NRUN(I)=NRUN(I)+1
      IF(ABS(Y).LT.DWIDTH) GO TO 340
      DRIFT(I)=SIZ(J-1,I)-ABS((SIZ(J,I)-SIZ(J-1,I))/(Y-TRAJ(J-1,I)))*
     &          (DWIDTH-ABS(TRAJ(J-1,I)))
      GO TO 250
  340 CONTINUE
      IF(ABS(SIZ(J,I)-SIZ(J-1,I)).GT.ERR) GO TO 233
      DRIFT(I)=SIZ(J,I)
      GO TO 250
  233 CONTINUE
      J=J+1
      IF(J.LE.15) GO TO 236
      WRITE(6,247)
  250 CONTINUE
C
C         COMBINE TRAJECTORY INFORMATION TO BUILD-UP THE DISTRIBUTION
C         AND ESTIMATE THE DRIFT OUTPUT RESULTS
C
      CALL SWATH(NCOL,NROW,Q,DIAMN,STDEV,YNOZ,DWIDTH,NRUN,MET,
     &            SWIDTH,DIST,DRIFT)
      WRITE(6,252)
  252 FORMAT("0","NOZZLE NO.",8X,"YNOZ",11X,"ZNOZ",8X,
     &       "DIAMN",10X,"STDEV",13X,"Q",12X,"DRIFT"/)
      DO 255 K=1,NCOL
      WRITE(6,253) K,YNOZ(K),ZNOZ(K),DIAMN(K),STDEV(K),Q(K),DRIFT(K)
  253 FORMAT(" ",4X,I3,6F15.4)
  255 CONTINUE
      WRITE(6,256)
  256 FORMAT("1"//" SINGLE DROPLET LATERAL DISPLACEMENT"//)
C
C         OUTPUT TRAJECTORY INFORMATION TO PRINTER
C
      NP=NCOL/3
```

```
      NLEFT=NCOL-NP*3
      KP=-2
      IF(NP.EQ.0) GO TO 425
      DO 405 K1=1,NP
      KK=IFIX(K1/2.)*2
      IF(K1.EQ.KK) WRITE(6,415)
  415 FORMAT("1","  ")
      KP=K1*3-2
      KP1=KP+1
      KP2=KP+2
      IF(NEVAP.EQ.1) GO TO 450
      WRITE(6,420) KP,KP1,KP2
  420 FORMAT("0","NOZZLE:",T13,I3,T55,I3,T97,I3/" ",T7,"YG",T21,"DIA",
     &       T49,"YG",T63,"DIA",T91,"YG",T105,"DIA")
      DO 430 K2=1,15
      WRITE(6,440) TRAJ(K2,KP),SIZ(K2,KP),TRAJ(K2,KP+1),SIZ(K2,KP+1),
     &             TRAJ(K2,KP+2),SIZ(K2,KP+2)
  440 FORMAT(" ",2E14.6,14X,2E14.6,14X,2E14.6)
  430 CONTINUE
      GO TO 405
  450 CONTINUE
      WRITE(6,460) KP,KP1,KP2
  460 FORMAT("0","NOZZLE:",T20,I3,T62,I3,T104,I3/" ",T7,"YG",T21,"DIA",
     &       T35,"DIAG",T49,"YG",T63,"DIA",T77,"DIAG",T91,"YG",T105,
     &       "DIA",T119,"DIAG")
      DO 470 K2=1,15
      WRITE(6,480) TRAJ(K2,KP),SIZ(K2,KP),DIAG(K2,KP),TRAJ(K2,KP+1),
     &             SIZ(K2,KP+1),DIAG(K2,KP+1),TRAJ(K2,KP+2),SIZ(K2,KP+2)
     &             ,DIAG(K2,KP+2)
  480 FORMAT(" ",9E14.6)
  470 CONTINUE
  405 CONTINUE
  425 CONTINUE
      IF(NLEFT.EQ.0) GO TO 600
      KK=IFIX(K1/2.)*2
      KP3=KP+3
      KP4=KP+4
      IF(K1.NE.KK) WRITE(6,415)
      IF(NLEFT.EQ.1) GO TO 550
      IF(NEVAP.EQ.1) GO TO 510
      WRITE(6,490) KP3,KP4
  490 FORMAT("0","NOZZLE:",T13,I3,T55,I3/" ",T7,"YG",T21,"DIA",T49,"YG",
     &       T63,"DIA")
      DO 500 K2=1,15
      WRITE(6,440) TRAJ(K2,KP+3),SIZ(K2,KP+3),TRAJ(K2,KP+4),SIZ(K2,KP+4)
  500 CONTINUE
      GO TO 600
  510 CONTINUE
      WRITE(6,520) KP3,KP4
  520 FORMAT("0","NOZZLE:",T20,I3,T62,I3/" ",T7,"YG",T21,"DIA",T35,
     &       "DIAG",T49,"YG",T63,"DIA",T77,"DIAG")
      DO 530 K2=1,15
      WRITE(6,480) TRAJ(K2,KP+3),SIZ(K2,KP+3),DIAG(K2,KP+3),
     &             TRAJ(K2,KP+4),SIZ(K2,KP+4),DIAG(K2,KP+4)
  530 CONTINUE
```

```
      GO TO 600
  550 CONTINUE
      IF(NEVAP.EQ.1) GO TO 580
      WRITE(6,560) KP3
  560 FORMAT("0","NOZZLE:",T13,I3/" ",T7,"YG",T21,"DIA")
      WRITE(6,570) (TRAJ(K2,KP+3),SIZ(K2,KP+3),K2=1,15)
  570 FORMAT(" ",2E14.6)
      GO TO 600
  580 CONTINUE
      WRITE(6,585) KP3
  585 FORMAT("0","NOZZLE:",T20,I3/" ",T7,"YG",T21,"DIA",T35,"DIAG")
      WRITE(6,590) (TRAJ(K2,KP+3),SIZ(K2,KP+3),DIAG(K2,KP+3),K2=1,15)
  590 FORMAT(" ",3F14.6)
  600 CONTINUE
      WRITE(6,260)
  260 FORMAT("1","FINAL DISTRIBUTION"//"0",5(3X," YG",6X,
     &        "CONCENTRATION")//)
C
C        WRITE TO FILE 8 AND OUTPUT TO 6 THE FINAL DISTRIBUTION
C
      KJ=0
      TOT=0.
      START=-SWIDTH
      NPTS=SWIDTH*200+1
      JJ=NPTS
      DO 310 J=1,NPTS
      XPL=START+(J-1)*.01
      WRITE(8,302) XPL,DIST(J)
  302 FORMAT(2E13.6)
      TOT=TOT+DIST(J)
      IF(DIST(J).LT.1.E-4) GO TO 310
      IF(KJ.EQ.5) KJ=0
      KJ=KJ+1
      XLIST(KJ)=XPL
      DLIST(KJ)=DIST(J)
      IF(KJ.EQ.5) WRITE(6,300) (XLIST(K),DLIST(K),K=1,5)
  300 FORMAT(" ",5(3X,F6.2,3X,E13.6))
  310 CONTINUE
      IF(KJ.NE.5) WRITE(6,300) (XLIST(K),DLIST(K),K=1,KJ)
C
C        DETERMINE TOTAL DRIFT
C
      QF=0.
      DRIFTF=0.
      DO 261 K=1,NCOL
      DRIFTF=DRIFTF+DRIFT(K)*Q(K)
      QF=QF+Q(K)
  261 CONTINUE
      DRIFTF=DRIFTF/QF
      WRITE(6,265) DRIFTF
      TOT=TOT/QF
  265 FORMAT("0","TOTAL DRIFT=",E13.6)
      WRITE(6,266) TOT
  266 FORMAT("0","TOTAL=",E14.6)
  320 CONTINUE
```

```
      IF(NPLOT.EQ.1) CALL PLOT1(TITLE)
      GO TO 800
  700 CONTINUE
      STOP
      END
C
C        SUBROUTINE SUB2D: CALLED BY THE MAIN PROGRAM, SUB2D CONTROLS
C        THE TRAJECTORY CALCULATION FOR ALL 2D CASES. GIVEN THE
C        INITIAL VALUES AND USER SELECTED OPTIONS, SUB2D CALCULATES
C        THE TRAJECTORY AND RETURNS THE FINAL VALUES OF THE VARIABLES
C
      SUBROUTINE SUB2D(NVAR,Y,Z,VD,WD,T,DIA,YY,SAVE,CSAVE,
     &                 YMAX,ERROR,DY,YY1,PW,IP)
      DIMENSION YY(13,NVAR),SAVE(13,NVAR),CSAVE(NVAR,3),YMAX(NVAR)
      DIMENSION ERROR(NVAR),DY(NVAR),YY1(NVAR)
      DIMENSION PW(NVAR,NVAR),IP(NVAR),IRUN(50)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMMON /AREA2/ N2D,N3D,NEVAP,EPS,NPROP2,NCW,NTUN,NDIST,NPRINT
      COMMON /AREA5/ ZPROP2,PK1,PK2,PK3,PK4,RP
      COMMON /AREA6/ JJ,IRUN,ICOUNT
      COMMON /AREA8/ DIA1
C
C        INITIALIZE THE DEPENDENT VARIABLES AND STORE IN ARRAY
C        YY(1,NVAR)
C
      DIA=DIA*1.E-6
      DIA1=DIA
      C=3.141592653589793
      T=0.E0
      YY(1,1)=Y
      YY(1,2)=VD
      YY(1,3)=Z
      YY(1,4)=WD
      IF(NEVAP.EQ.1) YY(1,NVAR)=DIA
      IF(N2D.EQ.2) GO TO 4
      IF(NVAR.GE.10) GO TO 1
      IF(NVAR.GE.8.AND.NPROP2.EQ.0) GO TO 2
      IF(NTUN.EQ.1.AND.NPROP2.EQ.1) YY(1,7)=ZPROP2
      IF(NTUN.EQ.1) GO TO 3
      IF(NPROP2.EQ.1) YY(1,5)=ZPROP2
      GO TO 4
    1 YY(1,10)=ZPROP2
      YY(1,9)=0.
    2 YY(1,8)=ZZ0
      YY(1,7)=-ZY0
    3 YY(1,6)=ZZ0
      YY(1,5)=ZY0
    4 CONTINUE
      IF(VD.EQ.0..AND.WD.EQ.0.) CALL INCON(YY,DIA,NVAR)
      IF(YY(1,4).EQ.0.E0.AND.YY(1,2).EQ.0.E0)GO TO 20
C
C        INITIALIZE VALUES FOR DIFSUB
C
      H=1.E-4
      MF=2
```

```
      DO 5 I=1,NVAR
      YMAX(I)=1.E0
    5 CONTINUE
      MAXDER=6
      HMIN=1.E-16
      HMAX=1.E0
      JSTART=0
      LOOP=0
      YPROP=0.
C
C        PERFORM THE STEP INTEGRATION  BY SUCCESSIVE CALLS OF DIFSUB
C
C        DIFSUB IS A FORTUOI SUBROUTINE TO SOLVE A SYSTEM OF N SIMULTANE
C        DIFFERENTIAL EQUATIONS
C
      DO 14 K=1,1000
      DO 9 I=1,NVAR
      YY1(I)=YY(1,I)
    9 CONTINUE
      T1=T
C
      CALL DIFSBM(NVAR,T,YY,SAVE,CSAVE,H,HMIN,HMAX,EPS,MF,YMAX,ERROR,
     &            KFLAG,JSTART,MAXDER,PW,IP)
C
C        MONITOR VARIABLES AFTER EACH TIME-STEP TO CONTROL OUTPUT
C        AND STORAGE OF TRAJECTORY AND TERMINATE TRAJECTORY IF
C        PARTICLE BECOMES ENTRAINED IN THE VORTEX OR HITS THE GROUND
C
      DIAPL=YY(1,NVAR)/1.E-6
      IF(NEVAP.EQ.1) YY(1,NVAR)=YY(1,NVAR)/1.E-6
      IF(NPRINT.EQ.1.AND.NDIST.EQ.0) WRITE (6,11) K,T,(YY(1,I),I=1,NVAR)
   11 FORMAT(" ",I4,4X,12F10.6)
      IF(NEVAP.EQ.1) YY(1,NVAR)=YY(1,NVAR)*1.E-6
      L=K+JJ
      IF(YY(1,3).LT.0.E0)GO TO 16
      IF(NDIST.EQ.1) GO TO 10
      WRITE(8,22) T,YY(1,1),YY(1,3)
   22 FORMAT(3E13.6)
   10 CONTINUE
      IF(NEVAP.EQ.1.AND.DIAPL.LT.15) GO TO 20
      IF(NPROP2.NE.1) GO TO 25
      IF(NCW.EQ.1) YPROP=YY(1,10)
      IF(YY1(1).LE.YPROP.AND.YY(1,1).GT.YPROP) LOOP=LOOP+1
      IF(LOOP.LT.2) GO TO 25
      IF(NPRINT.EQ.1.AND.NDIST.EQ.0) WRITE(6,30)
   30 FORMAT("*** TRAJECTORY TERMINATED, PROPELLER SLIPSTREAM ",
     &       "ENTRAINMENT ***")
      GO TO 20
   25 CONTINUE
      IF(YY(1,3).GT.YY1(3))GO TO 12
      GO TO 14
   12 IF(ABS(YY(1,1)).GT.1.E0)GO TO 13
      GO TO 14
   13 IF(YY(1,3).GT.ZZ0.AND.ABS(YY(1,1)).LT.ABS(YY1(1))) GO TO 18
   14 CONTINUE
```

```
      K=K-1
      WRITE (6,15)
   15 FORMAT("0","***** DO LOOP PARAMETER EXCEEDED IN SUB2D *****")
      GO TO 20
   16 CONTINUE
      IF(NDIST.EQ.0.AND.NPRINT.EQ.1) WRITE (6,17)
   17 FORMAT("0","***** TRAJECTORY COMPLETED, Z LESS THAN ZERO *****")
      CALL YZFIN(YY,YY1,NVAR,T,T1,L)
      GO TO 20
   18 CONTINUE
      IF(NDIST.EQ.0.AND.NPRINT.EQ.1) WRITE (6,19)
   19 FORMAT("0","***** TRAJECTORY TERMINATED, VORTEX ENTRAINMENT *****")
   20 CONTINUE
      JJ=JJ+K
      IRUN(ICOUNT)=K
      Y=YY(1,1)
      VD=YY(1,2)
      Z=YY(1,3)
      WD=YY(1,4)
      IF(NEVAP.EQ.1) DIA=YY(1,NVAR)/1.E-6
      IF(NEVAP.EQ.0) DIA=DIA1/1.E-6
      RETURN
      END
C
C        SUBROUTINE SUB3D: CALLED BY THE MAIN PROGRAM, SUB3D CONTROLS
C        THE TRAJECTORY CALCULATION FOR ALL 3D CASES. GIVEN THE
C        INITIAL VALUES AND USER SELECTED OPTIONS, SUB3D CALCULATES
C        THE TRAJECTORY AND RETURNS THE FINAL VALUES OF THE VARIABLES
C
      SUBROUTINE SUB3D(NVAR,X,Y,Z,UD,VD,WD,T,DIA,YY,SAVE,CSAVE,
     &                 YMAX,ERROR,DY,YY1,PW,IP)
      DIMENSION YY(13,NVAR),SAVE(13,NVAR),CSAVE(NVAR,3),YMAX(NVAR)
      DIMENSION ERROR(NVAR),DY(NVAR),YY1(NVAR)
      DIMENSION PW(NVAR,NVAR),IP(NVAR),IRUN(50)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMMON /AREA2/ N2D,N3D,NEVAP,FPS,NPROP2,NCW,NTUN,NDIST,NPRINT
      COMMON /AREA5/ ZPROP2,PK1,PK2,PK3,PK4,RP
      COMMON /AREA6/ JJ,IRUN,ICOUNT
      COMMON /AREA8/ DIA1
C
C        INITIALIZE THE DEPENDENT VARIABLES AND STORE IN ARRAY
C        YY(1,NVAR)
C
      DIA=DIA*1.E-6
      DIA1=DIA
      C=3.14159265358979З
      T=0.E0
      YY(1,1)=Y
      YY(1,2)=VD
      YY(1,3)=Z
      YY(1,4)=WD
      YY(1,5)=X
      YY(1,6)=UD
      IF(NEVAP.EQ.1) YY(1,NVAR)=DIA
      IF(N3D.EQ.2) GO TO 4
```

```
      IF(NVAR.GE.12) GO TO 1
      IF(NVAR.GE.10) GO TO 2
      IF(NPROP2.EQ.1) YY(1,7)=ZPROP2
      GO TO 4
    1 YY(1,12)=ZPROP2
      YY(1,11)=0.
    2 YY(1,10)=ZZ0
      YY(1,9)=-ZY0
    3 YY(1,8)=ZZ0
      YY(1,7)=ZY0
    4 CONTINUE
C
C        INITIALIZE VALUES FOR DIFSUB
C
      H=1.E-4
      MF=2
      DO 5 I=1,NVAR
      YMAX(I)=1.E0
    5 CONTINUE
      MAXDER=6
      HMIN=1.E-16
      HMAX=1.E0
      JSTART=0
      LOOP=0
      YPROP=0.
C
C        PERFORM THE STEP INTEGRATION  BY SUCCESSIVE CALLS OF DIFSUB
C
C        DIFSUB IS A FORTUOI SUBROUTINE TO SOLVE A SYSTEM OF N SIMULTANE
C        DIFFERENTIAL EQUATIONS
C
      DO 14 K=1,1000
      DO 9 I=1,NVAR
      YY1(I)=YY(1,I)
    9 CONTINUE
      T1=T
C
      CALL DIFSBM(NVAR,T,YY,SAVE,CSAVE,H,HMIN,HMAX,EPS,MF,YMAX,ERROR,
     &             KFLAG,JSTART,MAXDER,PW,IP)
C
C        MONITOR VARIABLES AFTER EACH TIME-STEP TO CONTROL OUTPUT
C        AND STORAGE OF TRAJECTORY AND TERMINATE TRAJECTORY IF
C        PARTICLE BECOMES ENTRAINED IN THE VORTEX OR HITS THE GROUND
C
      DIAPL=YY(1,NVAR)/1.E-6
      IF(NEVAP.EQ.1) YY(1,NVAR)=YY(1,NVAR)/1.E-6
      IF(NPRINT.EQ.1.AND.NDIST.EQ.0) WRITE (6,11) K,T,(YY(1,I),I=1,NVAR)
   11 FORMAT(" ",I4,4X,14F9.5)
      IF(NEVAP.EQ.1) YY(1,NVAR)=YY(1,NVAR)*1.E-6
      L=K+JJ
      IF(YY(1,3).LT.0.E0)GO TO 16
      IF(NDIST.EQ.1) GO TO 10
      WRITE(8,22) T,YY(1,1),YY(1,3)
   22 FORMAT(3F13.6)
   10 CONTINUE
```

```
      IF(NEVAP.EQ.1.AND.DIAPL.LT.15) GO TO 20
      IF(NPROP2.NE.1) GO TO 25
       IF(NCW.EQ.1) YPROP=YY(1,10)
      IF(YY1(1).LE.YPROP.AND.YY(1,1).GT.YPROP) LOOP=LOOP+1
      IF(LOOP.LT.2) GO TO 25
      IF(NPRINT.EQ.1.AND.NDIST.EQ.0) WRITE(6,30)
   30 FORMAT("*** TRAJECTORY TERMINATED, PROPELLER SLIPSTREAM ",
     &        "ENTRAINMENT ***")
      GO TO 20
   25 CONTINUE
      IF(YY(1,3).GT.YY1(3))GO TO 12
      GO TO 14
   12 IF(ABS(YY(1,1)).GT.1.E0)GO TO 13
      GO TO 14
   13 IF(YY(1,3).GT.ZZ0.AND.ABS(YY(1,1)).LT.ABS(YY1(1))) GO TO 18
   14 CONTINUE
      K=K-1
      WRITE (6,15)
   15 FORMAT("0","***** DO LOOP PARAMETER EXCEEDED IN SUB3D *****")
      GO TO 20
   16 CONTINUE
      IF(NDIST.EQ.0.AND.NPRINT.EQ.1) WRITE (6,17)
   17 FORMAT("0","***** TRAJECTORY COMPLETED, Z LESS THAN ZERO *****")
      CALL YZFIN(YY,YY1,NVAR,T,T1,L)
      GO TO 20
   18 CONTINUE
      IF(NDIST.EQ.0.AND.NPRINT.EQ.1) WRITE (6,19)
   19 FORMAT("0","***** TRAJECTORY TERMINATED, VORTEX ENTRAINMENT *****")
   20 CONTINUE
      JJ=JJ+K
      IRUN(ICOUNT)=K
      Y=YY(1,1)
      VD=YY(1,2)
      Z=YY(1,3)
      WD=YY(1,4)
      X=YY(1,5)
      UD=YY(1,6)
      IF(NEVAP.EQ.1) DIA=YY(1,NVAR)/1.E-6
      IF(NEVAP.EQ.0) DIA=DIA1/1.E-6
      RETURN
      END
C
C        PEDERV IS REQUIRED BY DIFSUB, BUT NOT NEEDED FOR THIS METHOD
C
      SUBROUTINE PEDERV(A,B,C,D)
      RETURN
      END
C
C        DIFFUN IS A SUBROUTINE CALLED BY DIFSUB TO EVALUATE THE RIGHT-
C        HAND SIDE OF THE SYSTEM OF DIFFERENTIAL EQUATIONS
C        RHS IS RETURNED TO DIFSUB IN ARRAY DY(NVAR)
C
      SUBROUTINE DIFFUN(NVAR,T,YY,DY)
      DIMENSION YY(13,NVAR),DY(NVAR)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
```

```
      COMMON /AREA2/ N2D,N3D,NEVAP,EPS,NPROP2,NCW,NTUN,NDIST,NPRINT
      COMMON /AREA3/ PSWB,PA,PV,SCN,DV
      COMMON /AREA5/ ZPROP2,PK1,PK2,PK3,PK4,RP
      COMMON /AREA8/ DIA1
      IF(NEVAP.EQ.0) DIA=DIA1
      Y=YY(1,1)
      VD=YY(1,2)
      Z=YY(1,3)
      WD=YY(1,4)
      IF(N3D.EQ.1) X=YY(1,5)
      IF(N3D.EQ.1) UD=YY(1,6)
C
C        CROSSWIND MODEL
C
      ZCW=Z
      IF(ZCW.LT.0.) ZCW=0.
      ZCW=ZCW*B
      IF(NCW.EQ.0) VCW=0.
C
C        BRANCH TO CORRECT SECTION OF CODE DEPENDING ON OPTIONS
C        SELECTED
C
       IF(N2D.EQ.2.OR.N3D.EQ.2) GO TO 45
      IF(N3D.EQ.1) GO TO 15
      KK=0
      IF(NVAR.EQ.11.OR.NVAR.EQ.10) GO TO 10
      IF(NCW.EQ.1) GO TO 20
      IF(NTUN.EQ.1) GO TO 30
      IF(NPROP2.EQ.1) GO TO 35
      IF(NVAR.EQ.5.OR.NVAR.EQ.4) GO TO 40
      WRITE (6,9)
    9 FORMAT (" ","***** NVAR INCORRECT *****")
   15 CONTINUE
      KK=2
      IF(NVAR.GE.12) GO TO 10
      IF(NVAR.GE.10) GO TO 20
      IF(NVAR.GE.7.AND.NPROP2.EQ.1) GO TO 35
      IF(NVAR.GE.6) GO TO 40
      WRITE(6,9)
   10 CONTINUE
C
C         BOTH VORTICES AND PROP POSITIONS ARE
C        DETERMINED BY STEP INTEGRATION
C        HERE NCW=1 AND NPROP2=1
C
      CALL VELTV(YY(1,5+KK),YY(1,6+KK),YY(1,7+KK),YY(1,8+KK),Y,Z,V1,W1)
      CALL PROP2D(Y,Z,YY(1,9+KK),YY(1,10+KK),V2,W2)
      VA=V1+V2+(VCW*ZCW**.25)/U
      WA=W1+W2
      IF(N3D.EQ.0) GO TO 17
      CALL BOUND(X,Y,Z,UA,W3)
      CALL PROPX(Y,Z,YY(1,11),YY(1,12),UAP)
      UA=UA+UAP
      WA=WA+W3
   17 CONTINUE
```

```
      CALL VELTV(YY(1,5+KK),YY(1,6+KK),YY(1,7+KK),YY(1,8+KK),
     $          YY(1,9+KK),YY(1,10+KK),VP1,WP1)
      DY(9+KK)=VP1+(VCW*(YY(1,10+KK)*B)**.25)/U
      DY(10+KK)=WP1
      IF(YY(1,10+KK).LT.RP) DY(10+KK)=0.
      CALL VELC(YY(1,5+KK),YY(1,6+KK),YY(1,7+KK),YY(1,8+KK),V1,W1)
      DY(5+KK)=V1+(VCW*(YY(1,6+KK)*B)**.25)/U
      DY(6+KK)=W1
      CALL VELC(YY(1,7+KK),YY(1,8+KK),YY(1,5+KK),YY(1,6+KK),V1,W1)
      DY(7+KK)=-V1+(VCW*(YY(1,8+KK)*B)**.25)/U
      DY(8+KK)=-W1
      GO TO 50
   20 CONTINUE
C
C        NCW=1: VORTEX POSITIONS DETERMINED BY STEP INTEGRATION,
C        NO PROP
C
      CALL VELTV(YY(1,5+KK),YY(1,6+KK),YY(1,7+KK),YY(1,8+KK),Y,Z,V1,WA)
      VA=V1+(VCW*ZCW**.25)/U
      IF(N3D.EQ.0) GO TO 25
      CALL BOUND(X,Y,Z,UA,W1)
      WA=WA+W1
   25 CONTINUE
      CALL VELC(YY(1,5+KK),YY(1,6+KK),YY(1,7+KK),YY(1,8+KK),V1,W1)
      DY(5+KK)=V1+(VCW*(YY(1,6+KK)*B)**.25)/U
      DY(6+KK)=W1
      CALL VELC(YY(1,7+KK),YY(1,8+KK),YY(1,5+KK),YY(1,6+KK),V1,W1)
      DY(7+KK)=-V1+(VCW*(YY(1,8+KK)*B)**.25)/U
      DY(8+KK)=-W1
      GO TO 50
   30 CONTINUE
C
C        NTUN=1: NPROP2D=0 OR 1
C
      VP1=0.
      WP1=0.
      CALL TUNVEL(Y,Z,YY(1,5),YY(1,6),V1,W1)
      IF(NPROP2.EQ.1) CALL PROP2D(Y,Z,0.,YY(1,7),VP1,WP1)
      VA=V1+VP1+(VCW*ZCW**.25)/U
      WA=W1+WP1
      CALL VORTUN(YY(1,5),YY(1,6),DY(5),DY(6))
      IF(NPROP2.EQ.0) GO TO 50
      CALL TUNVEL(0.,YY(1,7),YY(1,5),YY(1,6),VDUM,DY(7))
      GO TO 50
   35 CONTINUE
C
C        NPROP2=1: WITHOUT CROSSWIND OR TUNNEL
C
      CALL VEL2D(T,0.,YY(1,5+KK),V1,W1)
      IF(YY(1,5+KK).LE.RP) W1=0.
      DY(5+KK)=W1
      CALL VEL2D(T,Y,Z,VA,WA)
      CALL PROP2D(Y,Z,0.,YY(1,5+KK),VP1,WP1)
      VA=VA+VP1
      WA=WA+WP1
```

```
      IF(N3D.EQ.0) GO TO 37
      CALL BOUND(X,Y,Z,UA,W2)
      CALL PROPX(Y,Z,0.,YY(1,7),UAP)
      UA=UA+UAP
      WA=WA+W2
   37 CONTINUE
      GO TO 50
   40 CONTINUE
C
C         N2D=1 OR N3D=1: NO OPTIONS EXCEPT NEVAP MAY EQUAL 1
C
      CALL VEL2D(T,Y,Z,VA,WA)
      IF(N3D.EQ.0) GO TO 50
      CALL BOUND(X,Y,Z,UA,W1)
      WA=WA+W1
      GO TO 50
C
C         N2D=2 OR N3D=2
C         GET VELOCITY FROM USER SUPPLIED SUBROUTINE USERV
C
   45 CONTINUE
      CALL USERV(T,X,Y,Z,UA,VA,WA)
   50 CONTINUE
C
C         CALCULATE RHS OF DROPLET DYNAMICS EQUATIONS AND EVAPORATION
C
      IF(NEVAP.EQ.1) DIA=YY(1,NVAR)
      R=U*((DA*DIA)*SQRT((VA-VD)**2+(WA-WD)**2))/VIS
      IF(N3D.EQ.1) R=U*((DA*DIA)*SQRT((UA-UD)**2+(VA-VD)**2
     &           +(WA-WD)**2))/VIS
      CDR=1.0+0.197*R**0.63+0.26E-3*R**1.38
      S=18.*(B*DA)/(DIA*DD)
      RU=(DA*DIA*U)/VIS
      IF(NEVAP.EQ.0) GO TO 60
      DY(NVAR)=-2.*(B/U)*(18.016/28.97)*(DV/DIA)*(DA/DD)*((PSWB-PV)
     &          /(PA-PV))*(2.+.6*SCN**(1./3.)*R**.5)
   60 CONTINUE
      DY(1)=YY(1,2)
      DY(2)=((S*CDR)/RU)*(VA-VD)
      DY(3)=YY(1,4)
      DY(4)=((S*CDR)/RU)*(WA-WD)-(B*G)/U**2
      IF(NEVAP.EQ.1) DY(2)=DY(2)-(3./DIA)*VD*DY(NVAR)
      IF(NEVAP.EQ.1) DY(4)=DY(4)-(3./DIA)*WD*DY(NVAR)
      IF(N3D.EQ.0) GO TO 70
      DY(5)=YY(1,6)
      DY(6)=((S*CDR)/RU)*(UA-UD)
      IF(NEVAP.EQ.1) DY(6)=DY(6)-(3./DIA)*UD*DY(NVAR)
   70 RETURN
      END
C
C         SUBROUTINE USERV......USER SUPPLIED
C
C         WITH N2D=2 OR N3D=2 USERV IS CALLED BY DIFFUN FOR THE
C         VELOCITY IN THE WAKE.  THIS ALLOWS THE USER TO REPLACE
C         THE CODE'S WAKE MODEL WITH ONE OF THEIR OWN.  THE NEVAP
```

```
C        OPTION IS STILL AVAILABLE AS IS NDIST, NPRINT, AND NPLOT.
C        THE PROPELLER AND TUNNEL MODELS ARE AVAILABLE TO THE
C        USER BY INCLUDING THE PROPER INPUTS AND SUBROUTINE CALLS.
C
C        PASSED TO USERV IS THE NONDIMNSIONAL TIME,T, AND THE
C        NONDIMENSIONAL POSITION X,Y,Z.  THE USER RETURNS THE
C        CORRESPONDING NONDIMENSIONAL VELOCITIES UA,VA,WA.
C
      SUBROUTINE USERV(T,X,Y,Z,UA,VA,WA)
      RETURN
      END
C
C
C        SUBROUTINE VEL2D: INPUT DIMENSIONLESS TIME AND POSITION(Y,Z)
C        VEL2D RETURNS DIMENSIONLESS INDUCED VELOCITIES(UA,VA) FROM
C        A VORTEX PAIR UNAFFECTED BY CROSSWIND
C
      SUBROUTINE VEL2D(T,Y,Z,VA,WA)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
C
C        CALCULATE THE POSITION OF THE VORTICES
C
      C=3.141592654
      C1=1/(ZY0**2)+1/(ZZ0**2)
      C2=-((C1*ZY0**2-2.0)/SQRT(C1*ZY0**2-1.0))
      C3= ((C1*ZZ0**2-2.0)/SQRT(C1*ZZ0**2-1.0))
      C4=C2*((4.E0*C*A)/(CL*C1))
      C5=C3*((4.E0*C*A)/(CL*C1))
      IF(T.GT.C4) GO TO 10
      ZY=SQRT(((((T*CL*C1)/(4.0*C*A))**2)*C1-(C2*T*CL*C1**2)/(2.0*C*
     &   A)+C1*C2**2+4.0*C1)-SQRT(((((T*CL*C1)/(4.0*C*A))**2)*C1-(C2
     &   *T*CL*C1**2)/(2.0*C*A)+C1*C2**2+4.0*C1)**2-4.0*C1**2*(((T*CL*C1
     &   )/(4.0*C*A))**2-(C2*T*CL*C1)/(2.0*C*A)+C2**2+4.0)))/(2.0
     &   *C1**2))
      GO TO 20
   10 ZY=SQRT(((((T*CL*C1)/(4.0*C*A))**2)*C1-(C2*T*CL*C1**2)/(2.0*C*
     &   A)+C1*C2**2+4.0*C1)+SQRT(((((T*CL*C1)/(4.0*C*A))**2)*C1-(C2
     &   *T*CL*C1**2)/(2.0*C*A)+C1*C2**2+4.0*C1)**2-4.0*C1**2*(((T*CL*C1
     &   )/(4.0*C*A))**2-(C2*T*CL*C1)/(2.0*C*A)+C2**2+4.0)))/(2.0
     &   *C1**2))
   20 IF(T.LT.C5) GO TO 30
      ZZ=SQRT(((((T*CL*C1)/(4.0*C*A))**2)*C1-(C3*T*CL*C1**2)/(2.0*C*
     &   A)+C1*C3**2+4.0*C1)-SQRT(((((T*CL*C1)/(4.0*C*A))**2)*C1-(C3
     &   *T*CL*C1**2)/(2.0*C*A)+C1*C3**2+4.0*C1)**2-4.0*C1**2*(((T*CL*C1
     &   )/(4.0*C*A))**2-(C3*T*CL*C1)/(2.0*C*A)+C3**2+4.0)))/(2.0
     &   *C1**2))
      GO TO 40
   30 ZZ=SQRT(((((T*CL*C1)/(4.0*C*A))**2)*C1-(C3*T*CL*C1**2)/(2.0*C*
     &   A)+C1*C3**2+4.0*C1)+SQRT(((((T*CL*C1)/(4.0*C*A))**2)*C1-(C3
     &   *T*CL*C1**2)/(2.0*C*A)+C1*C3**2+4.0*C1)**2-4.0*C1**2*(((T*CL*C1
     &   )/(4.0*C*A))**2-(C3*T*CL*C1)/(2.0*C*A)+C3**2+4.0)))/(2.0
     &   *C1**2))
   40 CONTINUE
C
C        CALCULATE THE INDUCED VELOCITIES
```

```
C
      VA=(CL/(2.0*C*A))*((ZZ+Z)/((ZZ+Z)**2+(ZY-Y)**2)-(ZZ+Z)/((ZZ+Z)
     &   **2+(ZY+Y)**2)-(ZZ-Z)/((ZZ-Z)**2+(ZY+Y)**2)+(ZZ-Z)/((ZZ-Z)**2
     &   +(ZY-Y)**2))
      WA=(CL/(2.0*C*A))*((ZY-Y)/((ZZ+Z)**2+(ZY-Y)**2)+(ZY+Y)/((ZZ+Z)
     &   **2+(ZY+Y)**2)-(ZY+Y)/((ZZ-Z)**2+(ZY+Y)**2)-(ZY-Y)/((ZZ-Z)**2
     &   +(ZY-Y)**2))
      RETURN
      END
C
C        SUBROUTINE BOUND: CALCULATES THE INDUCED VELOCITIES (UA,WA)
C        DUE TO THE BOUND VORTEX AT A POINT (X,Y,Z)
C
      SUBROUTINE BOUND(X,Y,Z,UA,WA)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      R5=SQRT(X**2+(ZZ0-Z)**2)
      R6=SQRT(X**2+(ZZ0+Z)**2)
      UA=1.-(CL/(4.*3.141592654*A))*(((1.-Y)/SQRT(R5**2+(1.-Y)**2)+
     &   (1.+Y)/SQRT(R5**2+(1.+Y)**2))*((ZZ0-Z)/R5**2)+((1.-Y)/SQRT
     &   (R6**2+(1.-Y)**2)+(1.+Y)/SQRT(R6**2+(1.+Y)**2))*((ZZ0+Z)
     &   /R6**2))
      WA=(CL/(4.*3.141592654*A))*(-((1.-Y)/SQRT(R5**2+(1.-Y)**2)+
     &   (1.+Y)/SQRT(R5**2+(1.+Y)**2))*(X/R5**2)+((1.-Y)/SQRT(R6**2
     &   +(1.-Y)**2)+(1.+Y)/SQRT(R6**2+(1.+Y)**2))*(X/R6**2))
      RETURN
      END
C
C        SUBROUTINE PROPX: RETURNS THE X VELOCITY IN THE PROPELLER
C        SLIPSTREAM UAP DUE TO A PROP CENTERED AT (YP,ZP) AT A
C         POINT (Y,Z)
C
      SUBROUTINE PROPX(Y,Z,YP,ZP,UAP)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMMON /AREA5/ ZPROP2,PK1,PK2,PK3,PK4,RP
      R1=SQRT((Y-YP)**2+(Z-ZP)**2)
      IF(R1.LT.RP) GO TO 10
      UAP=0.
      RETURN
   10 CONTINUE
      R2=R1/RP
      UAP=(PK1*(1.-R2)+PK2*R2*SIN(R2*3.141592654))/U
      RETURN
      END
C
C
C        SUBROUTINE PROP2D: RETURNS DIMENSIONLESS PROPELLER INDUCED
C        VELOCITIES(V,W) AT POSITION (Y,Z) FOR A PROP AT (YP,ZP)
C        (ALL POSITIONS DIMENSIONLESS)
C
      SUBROUTINE PROP2D(Y,Z,YP,ZP,V,W)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMMON /AREA5/ ZPROP2,PK1,PK2,PK3,PK4,RP
      R1=SQRT((Y-YP)**2+(Z-ZP)**2)
      IF(R1.LT.RP) GO TO 10
      V=0.
```

```
      W=0.
      RETURN
   10 CONTINUE
      R2=R1/RP
      SV1=(PK3*(1.0-R2)+PK4*R2*SIN(R2*3.141592654))/U
      V=((Z-ZP)/R1)*SV1
      W=((YP-Y)/R1)*SV1
      RETURN
      END
C
C
C        SUBROUTINE VELTV: RETURNS INDUCED VELOCITIES (V,W) AT
C        POSITION (Y,Z) FOR VORTICES LOCATED AT (ZYR,ZZR) AND
C        (ZYL,ZZL)   (ALL QUANTITIES DIMENSIONLESS)
C
      SUBROUTINE VELTV(ZYR,ZZR,ZYL,ZZL,Y,Z,V,W)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMPLEX GC,ZCR,ZCL,WZ,ZD
      GAMMA=CL/(A*2.*3.141592654)
      GC=CMPLX(0.,GAMMA)
      ZD=CMPLX(Y,Z)
      ZCR=CMPLX(ZYR,ZZR)
      ZCL=CMPLX(ZYL,ZZL)
      WZ=-GC*(1./(ZD-ZCR)-1./(ZD-ZCL)+1./(ZD-CONJG(ZCL))
     &   -1./(ZD-CONJG(ZCR)))
      V=REAL(WZ)
      W=-AIMAG(WZ)
      RETURN
      END
C
C
C        SUBROUTINE VELC: RETURNS INDUCED VELOCITIES (V,W) ON A
C        VORTEX CORE AT (Y1,Z1) AND (Y2,Z2)   (ALL QUANTITIES ARE
C        DIMENSIONLESS)
C
      SUBROUTINE VELC(Y1,Z1,Y2,Z2,V,W)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMPLEX GC,ZC1,ZC2,WZ
      GAMMA=CL/(2.*3.141592654*A)
      GC=CMPLX(0.,GAMMA)
      ZC2=CMPLX(Y2,Z2)
      ZC1=CMPLX(Y1,Z1)
      WZ=-GC*(-1./(ZC1-ZC2)+1./(ZC1-CONJG(ZC2))-1./(ZC1-CONJG(ZC1)))
      V=REAL(WZ)
      W=-AIMAG(WZ)
      RETURN
      END
C
C
C        SUBROUTINE VORTUN: RETURNS INDUCED VELOCITIES (VV,WW)
C        AT A POINT (ZY,ZZ) FOR THE TUNNEL CASE WHERE VORTICES
C        ARE LOCATED AT (ZY,ZZ) AND (ZY,-ZZ)   (ALL QUANTITIES ARE
C        DIMENSIONLESS)
C
      SUBROUTINE VORTUN(ZY,ZZ,VV,WW)
```

```
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMPLEX Z1,W,CONST
      C=3.141592654
      DTUN=D/B
      Z1=CMPLX(ZY,ZZ)
      GAMMA=CL/(2.*C*A)
      E=0.
      CONST=CMPLX(E,GAMMA)
      W=(0.,0.)
C
C        CALCULATING THE COMPLEX VELOCITY AT A VORTEX
C
      DO 10 I=1,21
      N=I-11
      IF(N.EQ.0) GO TO 10
      W=W+CONST*(-(1./(N*DTUN))+(1./(2.*ZY+N*DTUN))-(1./(2.*Z1
     &           +N*DTUN))+(1./(Z1-CONJG(Z1)+N*DTUN)))
   10 CONTINUE
      W=W+CONST*((1./(2.*ZY))-(1./(2.*Z1))+(1./(Z1-CONJG(Z1))))
      VV=REAL(W)
      WW=-AIMAG(W)
      RETURN
      END
C
C        TUNVEL SOLVES FOR THE INDUCED VELOCITY (VA,WA) AT A POINT IN TH
C        TUNNEL (YD,ZD) AND A GIVEN VORTEX POSITION (ZY,+-ZZ).
C        (ALL QUANTITIES ARE DIMENSIONLESS)
C
      SUBROUTINE TUNVEL(YD,ZD,ZY,ZZ,VA,WA)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMPLEX C,Z,CONST,W
      CC=3.141592654
      GAMMA=(CL*U*B)/A
      CD=GAMMA/(2.*CC)
      E=0.
      CONST=CMPLX(E,CD)
      F=CC/D
      DYD=YD*B
      DZD=ZD*B
      DZY=ZY*B
      DZZ=ZZ*B
      C=CMPLX(DZY,DZZ)
      Z=CMPLX(DYD,DZD)
      W=-CONST*(((CSIN(F*Z))**2-(CSIN(F*CONJG(C)))**2)/((CSIN(F*Z))**2
     &  -(CSIN(F*C))**2))*(((2.*F*CSIN(F*Z)*CCOS(F*Z))*((CSIN(F*C))**2
     &  -(CSIN(F*CONJG(C)))**2))/((CSIN(F*Z))**2-(CSIN(F*CONJG(C)))**2
     &  )**2)
      W=CONJG(W)/U
      VA=REAL(W)
      WA=AIMAG(W)
      RETURN
      END
C
C        SUBROUTINE SWATH: CALCULATES THE DISTRIBUTION FROM EACH NOZZLE
C        AND SUMS THESE UP TO GENERATE A TOTAL DISTRIBUTION AND
```

```
C        ESTIMATES THE DRIFT. IF NPRINT=1 EACH NOZZLE'S DISTRIBUTION
C        AND DRIFT IS OUTPUT
C
      SUBROUTINE SWATH(NCOL,NROW,Q,DIAMN,STDEV,YNOZ,DWIDTH,
     &                 NRUN,MET,WIDTH,DIST,DRIFT)
      DIMENSION TRAJ(15,100),YNOZ(100),SIZE(15,100),NRUN(100),DIST(1)
      DIMENSION DRIFT(100),YI(15),C(4,15),SIZ(15),SIZ1(15)
      DIMENSION DIAG(15,100),S(15,5),MET(100),XLIST(4),SLIST(4)
      DIMENSION DIAMN(100),STDEV(100),Q(100),C1(4,15),DLIST(4)
      EXTERNAL CINT,BINT
      COMMON /AREA2/ N2D,N3D,NEVAP,EPS,NPROP2,NCW,NTUN,NDIST,NPRINT
      COMMON /AREA9/ SIZ,SIZ1,NR,C1,DIAMNI,STDEVI
      COMMON /AREA10/ TRAJ,DIAG,SIZE
      DATA S/75*0./
      KJ=0
      START=-WIDTH
      NWIDTH=100*WIDTH+1-START*100
      DO 10 I=1,NCOL
      NR=NRUN(I)
      IF(NR.EQ.0) GO TO 200
      IF(MET(I).EQ.1.OR.NR.LT.4) GO TO 100
C
C        IF MET(I)=0 THE I'TH NOZZLE'S DISTRIBUTION IS CALCULATED
C        USING A CUBIC SPLINE
C
      IF(NPRINT.EQ.0) GO TO 46
      WRITE (6,40) YNOZ(I)
   40 FORMAT("1","YNOZ=",F5.3)
      WRITE (6,45)
   45 FORMAT("0",3(7X,"YG",11X,"DIA",5X,"CONCENTRATION"))
   46 CONTINUE
      DO 20 J=1,NR
      K=NR-J+1
      SIZ(J)=SIZE(K,I)
      SIZ1(J)=DIAG(K,I)
      YI(J)=1./(TRAJ(K,I)-YNOZ(I))
   20 CONTINUE
      IBC=0
      AA=0.
      BB=0.
      IF(NEVAP.EQ.1) CALL CSPL(SIZ,SIZ1,NR,C1,S,IBC,AA,BB)
      CALL CSPL(YI,SIZ,NR,C,S,IBC,AA,BB)
      IF(ABS(TRAJ(NR,I)).LT.DWIDTH) GO TO 21
      DELTAY=1./(DWIDTH-YNOZ(I))
      CALL CSFIND(YI,SIZ,NR,C,DELTAY,DRIFT(I),DUMI,DUMY)
   21 CONTINUE
C
C        CALCULATE THE DISTRIBUTION AND ADD TO DIST ARRAY,
C        OUTPUT TO 6 IF NPRINT=1
C
      NPTS=(TRAJ(1,I)-START)*100+1
      NEND=NWIDTH
      IF(TRAJ(NR,I).LT.TRAJ(1,I)) NEND=1
      NSTART=NPTS
      NSTOP=NEND
```

```
      IF(NEND.NE.1) GO TO 25
      NSTART=NEND
      NSTOP=NPTS
   25 CONTINUE
      DIAOLD=10000.
      DO 30 KKK=NSTART,NSTOP
      KK=KKK
      IF(NEND.EQ.1) KK=NSTOP-(KKK-1)
      YG=START+(KK-1)*.01
      DELTAY=1./(YG-YNOZ(I))
      CALL CSFIND(YI,SIZ,NR,C,DELTAY,DIA,DDYG,DUMY)
      IF(DIA.GE.DIAOLD) WRITE(6,27) I,DIA
   27 FORMAT("0","***  CUBIC SPLINE PROBLEM  NOZZLE NO.",I3,"  DIA=",
     &        E13.6,"  ***")
      DIAOLD=DIA
      IF(DIA.LT.DRIFT(I)) GO TO 35
      IF(DIA.GT.SIZ(NR)) GO TO 30
      DVDD=(EXP(-.5*((DIA-DIAMN(I))/STDEV(I))**2))
     &        /(STDEV(I)*SQRT(2.*3.141592654))
      IF(NEVAP.EQ.1) CALL CSFIND(SIZ,SIZ1,NR,C1,DIA,DIA1,DUMY,DUMMY)
      IF(NEVAP.EQ.1) DVDD=DVDD*(DIA1**3/DIA**3)
      DYG=-1./((YG-YNOZ(I))**2)
      DDYG=DDYG*DYG
      DIST1=ABS(Q(I)*DVDD*DDYG)
      DIST(KK)=DIST1+DIST(KK)
      IF(NPRINT.EQ.0) GO TO 30
      IF(DIST1.LT.(1.E-4/Q(I))) GO TO 30
      IF(KJ.EQ.3) KJ=0
      KJ=KJ+1
      XLIST(KJ)=YG
      DLIST(KJ)=DIST1
      SLIST(KJ)=DIA
      IF(KJ.EQ.3) WRITE(6,80) (XLIST(K1),SLIST(K1),DLIST(K1),K1=1,3)
   80 FORMAT(" ",3(E15.6,2E13.6))
   30 CONTINUE
   35 CONTINUE
      IF(KJ.NE.3.AND.NPRINT.EQ.1) WRITE(6,80) (XLIST(K1),SLIST(K1),
     &                           DLIST(K1),K1=1,KJ)
      GO TO 200
  100 CONTINUE
C
C        IF MET(I)=1 THE I'TH NOZZLE'S DISTRIBUTION IS CALCULATED
C        USING A LINEAR FIT
C
      KJ=0
      IF(NR.LT.2) GO TO 200
      IF(NPRINT.EQ.1) WRITE(6,105) YNOZ(I)
  105 FORMAT("1","YNOZ=",F5.3/"0",4(8X,"YG",7X,"CONCENTRATION")/)
      DO 110 J=1,NR
      K=NR-J+1
      SIZ(J)=SIZE(K,I)
      SIZ1(J)=DIAG(K,I)
      YI(J)=TRAJ(K,I)
  110 CONTINUE
      IF(NEVAP.EQ.0) GO TO 115
```

```
      IF(NR.GE.4) GO TO 112
      WRITE(6,113) I
  113 FORMAT("0","*** INSUFFICIENT DATA TO CALCULATE DISTRIBUTION ***"/
     &       " *** NOZZLE NO. ",I2," DISTRIBUTION SET = TO ZERO ***")
      GO TO 300
  112 CONTINUE
      IBC=0.
      AA=0.
      BB=0.
      CALL CSPL(SIZ,SIZ1,NR,C1,S,IBC,AA,BB)
  115 CONTINUE
C
C        CALCULATE DISTRIBUTION AND ADD TO DIST ARRAY,
C        IF NPRINT=1 OUTPUT TO 6
C
      YMIN=100
      YMAX=-100
      DO 120 K=1,NR
      IF(YI(K).LT.YMIN) YMIN=YI(K)
      IF(YI(K).GT.YMAX) YMAX=YI(K)
  120 CONTINUE
      ISTART=(YMIN-START)*100+1
      DO 130 K=ISTART,NWIDTH
      YG=(K-1)*.01+START
      IF(ABS(YG).GT.DWIDTH) GO TO 130
      DIST1=0.
      IF(YG.GT.YMAX) GO TO 199
      J1=NR-1
      DO 140 J=1,J1
      IF(YG.LT.YI(J).AND.YG.LT.YI(J+1)) GO TO 140
      IF(YG.GT.YI(J).AND.YG.GT.YI(J+1)) GO TO 140
      SLOPE=(SIZ(J+1)-SIZ(J))/(YI(J+1)-YI(J))
      DIA=SIZ(J)+SLOPE*(YG-YI(J))
      DVDD=(EXP(-.5*((DIA-DIAMN(I))/STDEV(I))**2))
     &      /(STDEV(I)*SQRT(2.*3.141592654))
      IF(NEVAP.EQ.0) GO TO 135
      CALL CSFIND(SIZ,SIZ1,NR,C1,DIA,DIA1,DMY,DMMY)
      DVDD=DVDD*(DIA1**3/DIA**3)
  135 CONTINUE
      DIST1=DIST1+Q(I)*ABS(SLOPE*DVDD)
  140 CONTINUE
      DIST(K)=DIST1+DIST(K)
      IF(NPRINT.EQ.0) GO TO 130
      IF(KJ.EQ.4) KJ=0
      KJ=KJ+1
      XLIST(KJ)=YG
      DLIST(KJ)=DIST1
      IF(KJ.EQ.4) WRITE(6,141) (XLIST(K1),DLIST(K1),K1=1,4)
  141 FORMAT(" ",4(E16.6,E14.6))
  130 CONTINUE
  199 CONTINUE
      IF(KJ.NE.4.AND.NPRINT.EQ.1) WRITE(6,141) (XLIST(K1),DLIST(K1)
     &                               ,K1=1,KJ)
  200 CONTINUE
C
```

```
C         DRIFT ESTIMATION
C
      DRFT=DRIFT(I)
      IF(DRIFT(I).EQ.0) GO TO 300
      IF(DRIFT(I).NE.SIZE(1,I)) GO TO 205
      DRIFT(I)=100.
      GO TO 350
  205 CONTINUE
      AA=0.
      BB=(DRIFT(I)-DIAMN(I))/STDEV(I)
      ISEG=ABS(BB*20)/2
      ISEG=ISEG*2
      CALL SIMP(AA,BB,ISEG,CINT,ANS)
      ANS=ANS*SQRT(1./(2.*3.141592654))
      DRIFT(I)=(.50+ANS)*100.
  300 CONTINUE
      IF(NEVAP.EQ.0) GO TO 350
C
C         INCLUDE THE EFFECT OF EVAPORATION IN THE DRIFT
C
      IF(NR.GE.4) GO TO 320
      WRITE(6,330) I
  330 FORMAT("0","***INSUFFICIENT DATA TO ESTIMATE DRIFT ***"/
     &       " *** NOZZLE NO. ",I2," DRIFT SET EQUAL TO ZERO ***")
      DRIFT(I)=0.
      GO TO 10
  320 CONTINUE
      AA=DRFT
      BB=SIZ(NR)
      ISEG=IFIX((SIZ(NR)-SIZ(1))/20)*20
      IF(ISEG.LT.2) ISEG=2
      DIAMNI=DIAMN(I)
      STDEVI=STDEV(I)
      CALL SIMP(AA,BB,ISEG,BINT,ANS)
      EVAP=1.-ANS
      DRIFT(I)=EVAP*100.+DRIFT(I)
  350 CONTINUE
      IF(NPRINT.EQ.1) WRITE (6,60) DRIFT(I)
   60 FORMAT("0","DRIFT=",E14.6)
   10 CONTINUE
      RETURN
      END
C
C         SIMP IS A SIMPSON'S RULE INTEGRATION ROUTINE
C
C         AA,BB  ARE THE LIMITS OF INTEGRATION
C         ISEG   IS THE EVEN NUMBER OF SEGMENTS AA TO BB IS DIVIDED INTO
C         CINT   IS THE SUBROUTINE WHICH EVALUATES THE INTEGRAND
C         ANS    RETURNS THE ANSWER
C
      SUBROUTINE SIMP(AA,BB,ISEG,CINT,ANS)
      DELTA=(BB-AA)/ISEG
      ISEG1=ISEG-1
      SUM=0.
      DO 10 I=1,ISEG1
```

```
      FAC=4.
      K=I/2
      IF((K*2).EQ.I) FAC=2.
      X=AA+I*DELTA
      CALL CINT(X,Y)
      SUM=SUM+Y*FAC
   10 CONTINUE
      CALL CINT(AA,Y1)
      CALL CINT(BB,Y2)
      SUM=SUM+Y1+Y2
      ANS=(DELTA/3.)*SUM
      RETURN
      END
C
C        CINT EVALUATES INTEGRAND WHEN NEVAP=0
C
      SUBROUTINE CINT(X,Y)
      Y=EXP(-.50*X**2)
      RETURN
      END
C
C        BINT EVALUATES INTEGRAND WHEN NEVAP=1
C
      SUBROUTINE BINT(X,Y)
      DIMENSION SIZ(15),SIZ1(15),C1(4,15)
      COMMON /AREA9/ SIZ,SIZ1,NR,C1,DIAMNI,STDEVI
      CALL CSFIND(SIZ,SIZ1,NR,C1,X,DIA1,DUMY,DUMMY)
      DIA=X
      DVDD=(EXP(-.5*((DIA-DIAMNI)/STDEVI)**2))
     &        /(STDEVI*SQRT(2.*3.141592654))
      Y=(DIA1/DIA)**3*DVDD
      RETURN
      END
C
C        INCON EVALUATES THE INITIAL VELOCITY OF THE DROPLET
C
C        TO CORRECT FOR THE EFFECT OF THE BOUND VORTEX THE DROPLET IS
C        GIVEN AN INITIAL VELOCITY EQUAL TO ITS TERMINAL VELOCITY AT
C        THAT POINT IN THE FLOW FIELD
C
      SUBROUTINE INCON(YY,DIA,NVAR)
      DIMENSION YY(13,NVAR)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMMON /AREA2/ N2D,N3D,NEVAP,EPS,NPROP2,NCW,NTUN,NDIST,NPRINT
      C=3.141592654
      ZY=ZY0
      ZZ=ZZ0
      Y=YY(1,1)
      Z=YY(1,3)
      IF(NTUN.EQ.1) GO TO 5
      VA=(CL/(2.0*C*A))*((ZZ+Z)/((ZZ+Z)**2+(ZY-Y)**2)-(ZZ+Z)/((ZZ+Z)
     &    **2+(ZY+Y)**2)-(ZZ-Z)/((ZZ-Z)**2+(ZY+Y)**2)+(ZZ-Z)/((ZZ-Z)**2
     &    +(ZY-Y)**2))
      WA=(CL/(2.0*C*A))*((ZY-Y)/((ZZ+Z)**2+(ZY-Y)**2)+(ZY+Y)/((ZZ+Z)
     &    **2+(ZY+Y)**2)-(ZY+Y)/((ZZ-Z)**2+(ZY+Y)**2)-(ZY-Y)/((ZZ-Z)**2
```

```
     &    +(ZY-Y)**2))
    5 CONTINUE
      IF(NTUN.EQ.1) CALL TUNVEL(Y,Z,ZY,ZZ,VA,WA)
      S=18.*(R*DA)/(DIA*DD)
      RU=(DA*DIA*U)/VIS
      VD=VA
      YY(1,2)=VA
      WDL=WA
      WDR=WA
      DO 10 J=1,100
      WDR=WDR+WA
      R=U*ABS(((DA*DIA)*(WA-WDR))/VIS)
      CDR=1.0+0.197*R**0.63+0.26E-03*R**1.38
      FR=((S*CDR)/RU)*(WA-WDR)-(B*G)/U**2
      IF(FR.GT.0.E0)GO TO 40
   10 CONTINUE
      WRITE (6,20)
   20 FORMAT("0","***** WDR EXCEEDED DO LOOP IN INCON *****")
      WRITE (6,30)
   30 FORMAT("1","***** RIGHT ENDPOINT IN INCON INCORRECT *****")
      YY(1,4)=0.E0
      GO TO 90
   40 CONTINUE
C
C         SOLVE FOR THE VELOCITY BY THE METHOD OF BISECTION
C
      DO 50 I=1,500
      WDM=(WDL+WDR)/2.E0
      R=ABS(((DA*DIA)*(WA-WDM))/VIS)
      R=R*U
      CDR=1.0+0.197*R**0.63+0.26E-03*R**1.38
      FM=((S*CDR)/RU)*(WA-WDM)-(B*G)/U**2
      FF=ABS(FM)
      IF(FF.LT.EPS)GO TO 70
      IF(FM.GT.0.E0) WDR=WDM
      IF(FM.LT.0.E0) WDL=WDM
   50 CONTINUE
      WRITE (6,60)
   60 FORMAT("1","***** DO LOOP PARAMETER IN INCON EXCEEDED *****")
      YY(1,4)=0.E0
      GO TO 90
   70 YY(1,4)=WDM
      IF(NDIST.EQ.0) WRITE (6,80) VD,WDM
   80 FORMAT("0","INCON VALUES",T25,"VD=",E13.6,T45,"WD=",E13.6//)
   90 RETURN
      END
C
C         YZFIN EVALUATES THE DROPLET LOCATION AND VELOCITY
C         WHEN IT INTERSECTS THE GROUND PLANE
C         VALUES FOR THE NEXT TO LAST TIME-STEP ARE IN THE ARRAY
C         YY1 AND T1. THE VALUES FOR THE LAST STEP YY AND T ARE
C         REPLACED BY THE LINEARLY INTERPOLATED FINAL VALUES
C         TO BE RETURNED
C
      SUBROUTINE YZFIN(YY,YY1,NVAR,T,T1,L)
```

```
      DIMENSION YY(13,NVAR),YY1(NVAR),IRUN(50)
      COMMON /AREA1/ A,CL,B,U,G,ZY0,ZZ0,DA,VIS,VCW,D,DD
      COMMON /AREA2/ N2D,N3D,NEVAP,EPS,NPROP2,NCW,NTUN,NDIST,NPRINT
      COMMON /AREA6/ JJ,IRUN,ICOUNT
      CONV=(0.E0-YY1(3))/(YY(1,3)-YY1(3))
      DO 10 I=1,NVAR
      YY(1,I)=(YY(1,I)-YY1(I))*CONV+YY1(I)
   10 CONTINUE
      YY(1,3)=0.
      T=((T-T1)*CONV)+T1
      IF(NDIST.EQ.1) GO TO 20
      WRITE(8,30) T,YY(1,1),YY(1,3)
   30 FORMAT(3E13.6)
   20 RETURN
      END
C
C     SUBROUTINE CSPL
C     CSPL FITS A CUBIC SPLINE THROUGH THE M INPUT POINTS (X,Y).
C     THE EQUATION FOR THE CUBIC WHEN X IS GREATER THAN X(I) AND
C     LESS THAN X(I+1) IS:
C     Y(X)=C(1,I)*(X(I+1)-X)**3+C(2,I)*(X-X(I))**3+C(3,I)*
C          (X(I+1)-X)+C(4,I)*(X-X(I))
C
C     PARAMETERS:
C
C     X        AN ARRAY INPUT AND DIMENSIONED X(M) CONTAINING THE
C              INDEPENDENT VARIABLE IN INCREASING ORDER
C     Y        AN ARRAY INPUT AND DIMENSIONED Y(M) CONTAINING THE
C              DEPENDENT VARIABLE
C     M        NUMBER OF (X,Y) INPUT
C     C        AN ARRAY OUTPUT WITH DIMENSION C(4,M-1) CONTAINING
C              THE DESIRED COEFFICIENTS
C     S        A WORK ARRAY DIMENSIONED S(M,5)
C     IBC      =0  SECOND DERIVATIVE OF Y WRT X SET EQUAL TO A AT
C                  X(1) AND B AT X(M)
C              =1  FIRST DERIVATIVE OF Y WRT X SET EQUAL TO A AT
C                  X(1) AND B AT X(M)
C     A,B      INPUT VALUE OF FIRST OR SECOND DERIVATIVE DEPENDING
C              ON IBC
C
C     PROGRAMED BY M. BRAGG   AARL/OSU   MARCH,1980
C
      SUBROUTINE CSPL(X,Y,M,C,S,IBC,A,B)
      DIMENSION X(1),Y(1),C(4,1),S(M,5)
      DO 2 K2=1,5
      DO 1 K1=1,M
    1 S(K1,K2)=0.
    2 CONTINUE
      IF(IBC.NE.0) GO TO 10
      S(1,2)=1.
      S(M,2)=1.
      S(1,4)=A
      S(M,4)=B
      GO TO 20
   10 CONTINUE
```

```
      S(1,2)=2.
      S(1,3)=1.
      S(1,4)=6.*((Y(2)-Y(1))/((X(2)-X(1))**2))-(6.*A/(X(2)-X(1)))
      S(M,1)=1.
      S(M,2)=2.
      S(M,4)=6.*((Y(M-1)-Y(M))/((X(M)-X(M-1))**2))+(6.*B/(X(M)-X(M-1)))
   20 CONTINUE
      M1=M-1
       DO 30 I=2,M1
      DX=X(I+1)-X(I)
      DX1=X(I)-X(I-1)
      S(I,1)=DX1/DX
      S(I,2)=(2.*DX+2.*DX1)/DX
      S(I,3)=1.
      S(I,4)=6.*(((Y(I+1)-Y(I))/DX**2)-((Y(I)-Y(I-1))/(DX*DX1)))
   30 CONTINUE
C
C     USE THOMAS ALGORITHUM TO SOLVE TRI-DIAGONAL SYSTEM
C
      DO 40 I=2,M
      S(I,2)=S(I,2)-S(I,1)*S(I-1,3)/S(I-1,2)
      S(I,4)=S(I,4)-S(I,1)*S(I-1,4)/S(I-1,2)
   40 CONTINUE
      S(M,5)=S(M,4)/S(M,2)
      DO 50 I=2,M
      J=M-I+1
      S(J,5)=(S(J,4)-S(J,3)*S(J+1,5))/S(J,2)
   50 CONTINUE
      DO 60 I=1,M1
      DX=X(I+1)-X(I)
      C(1,I)=S(I,5)/(6.*DX)
      C(2,I)=S(I+1,5)/(6.*DX)
      C(3,I)=(Y(I)/DX)-((S(I,5)*DX)/6.)
      C(4,I)=(Y(I+1)/DX)-((S(I+1,5)*DX)/6.)
   60 CONTINUE
      RETURN
      END
C
C     SUBROUTINE CSFIND
C
C     SCFIND USES THE X,Y,M,C FROM A CALL TO CSPL AND RETURNS FOR
C     ANY INDEPENDENT VARIABLE XX; THE CUBIC SPLINE VALUE OF THE
C     DEPENDENT VARIABLE YY, THE FIRST DERIVATIVE YP, AND THE
C     SECOND DERIVATIVE Y2P.
C
      SUBROUTINE CSFIND(X,Y,M,C,XX,YY,YP,Y2P)
      DIMENSION X(1),Y(1),C(4,1)
      M1=M-1
      IF(X(1).GT.X(M)) GO TO 20
      DO 10 I=1,M1
      IF(X(I).LE.XX.AND.X(I+1).GT.XX) GO TO 40
   10 CONTINUE
      IF(XX.LT.X(1)) I=1
      IF(XX.GE.X(M)) I=M1
      GO TO 40
```

```
   20 CONTINUE
      DO 30 I=1,M1
      IF(X(I).GE.XX.AND.X(I+1).LT.XX) GO TO 40
   30 CONTINUE
      IF(XX.GT.X(1)) I=1
      IF(XX.LT.X(M)) I=M1
   40 CONTINUE
      YY=C(1,I)*(X(I+1)-XX)**3+C(2,I)*(XX-X(I))**3+C(3,I)*(X(I+1)-XX)
     &   +C(4,I)*(XX-X(I))
      YP=-3.*C(1,I)*(X(I+1)-XX)**2+3.*C(2,I)*(XX-X(I))**2-C(3,I)
     &   +C(4,I)
      Y2P=6.*C(1,I)*(X(I+1)-XX)+6.*C(2,I)*(XX-X(I))
      RETURN
      END
C
C          SUBROUTINE PLOT1
C
C          PLOT1 IS CALLED IF NPLOT=1 AND PLOTS PARTICLE TRAJECTORIES
C          IF NDIST=0 AND THE FINAL DISTRIBUTION IF NDIST=1
C
      SUBROUTINE PLOT1(TITLE)
      DIMENSION IRUN(50),YPLOT(1001),ZPLOT(1001),TITLE(13)
      COMMON /AREA2/ N2D,N3D,NEVAP,EPS,NPROP2,NCW,NTUN,NDIST,NPRINT
      COMMON /AREA6/ JJ,IRUN,ICOUNT
      NTOT=JJ
      REWIND 8
      CALL PLOTF(120,4)
      IF(NDIST.EQ.1) GO TO 200
      DO 10 I=1,NTOT
      READ(8,16) T,YPLOT(I),ZPLOT(I)
   16 FORMAT(3F13.6)
   10 CONTINUE
      ZMAX=0.
      DO 20 I=1,NTOT
      IF(ZPLOT(I).GT.ZMAX) ZMAX=ZPLOT(I)
   20 CONTINUE
      YPLOT(NTOT+1)=0.
      NTOT1=NTOT+1
      CALL SCALE(YPLOT,8.,NTOT1,FY,DY)
      FZ=0.
      DZ=DY
      IZ=ZMAX/DY+1.
      IF(IZ.LE.6) GO TO 30
      CALL SCALE(ZPLOT,6.,NTOT,FZ,DZ)
      DY=DZ
   30 CONTINUE
      AIZ=IZ
      CALL PLOT(0.5,1.5,-3)
      CALL AXIS(0.,0.,'2Y/B',-4,8.,0.,FY,DY)
      CALL AXIS(0.,0.,'2Z/B',4,AIZ,90.,FZ,DZ)
      CALL SYMBOL(1.0,6.5,.14,TITLE,0.,36)
      K=1
      DO 40 I=1,ICOUNT
      II=IRUN(I)+K-1
      DO 50 J=K,II
```

```
      JJ=J-K+1
      YPLOT(JJ)=YPLOT(J)
      ZPLOT(JJ)=ZPLOT(J)
   50 CONTINUE
      NLAST=IRUN(I)
      CALL LINE(YPLOT,FY,DY,ZPLOT,FZ,DZ,NLAST,0,1)
      K=K+IRUN(I)
   40 CONTINUE
      CALL PLOT(11.,-1.5,999)
      RETURN
  200 CONTINUE
      YMAX=-100.
      YMIN=100.
      DO 220 I=1,JJ
      READ(8,230) YPLOT(I),ZPLOT(I)
  230 FORMAT(2F13.6)
      IF(ZPLOT(I).GT.1.E-2.AND.YPLOT(I).LT.YMIN) YMIN=YPLOT(I)
      IF(ZPLOT(I).GT.1.E-2.AND.YPLOT(I).GT.YMAX) YMAX=YPLOT(I)
  220 CONTINUE
      CALL PLOT(0.,1.5,-3)
      K=0
      DO 210 I=1,JJ
      IF(YPLOT(I).LT.YMIN) GO TO 210
      IF(YPLOT(I).GT.YMAX) GO TO 210
      K=K+1
      YPLOT(K)=YPLOT(I)
      ZPLOT(K)=ZPLOT(I)
  210 CONTINUE
      CALL SCALE(YPLOT,8.,K,FX,DX)
      CALL SCALE(ZPLOT,6.,K,FD,DD)
      CALL AXIS(0.,0.,'2Y/B',-4,8.,0.,FX,DX)
      CALL AXIS(0.,0.,'CONCENTRATION',13,6.,90.,FD,DD)
      CALL SYMBOL(1.0,6.5,.14,TITLE,0.,36)
      CALL LINE(YPLOT,FX,DX,ZPLOT,FD,DD,K,0,1)
      CALL PLOT(11.,-1.5,999)
      RETURN
      END
```

```
C* THIS SUBROUTINE INTEGRATES A SET OF N ORDINARY DIFFERENTIAL FIRST   *
C* ORDER EQUATIONS OVER ONE STEP OF LENGTH H AT EACH CALL.  H CAN BE   *
C* SPECIFIED BY THE USER FOR EACH STEP, BUT IT MAY BE INCREASED OR     *
C* DECREASED BY DIFSUB WITHIN THE RANGE HMIN TO HMAX IN ORDER TO       *
C* ACHIEVE AS LARGE A STEP AS POSSIBLE WHILE NOT COMMITTING A SINGLE   *
C* STEP ERROR WHICH IS LARGER THAN EPS IN THE L-2 NORM, WHERE EACH     *
C* COMPONENT OF THE ERROR IS DIVIDED BY THE COMPONENTS OF YMAX.        *
C*                                                                     *
C* THE PROGRAM REQUIRES FOUR SUBROUTINES NAMED                         *
C*    DIFFUN(N,T,Y,DY)    --USER-SUPPLIED--
C*    DECOMP(N,M,PW,IP)       --RENAMED TO NDI01Z IN THIS SOURCE--     *
C*    SOLVE(N,M,PW,CSAVE(1,1),IP)  --RENAMED TO NDI02Z--               *
C*    PEDERV(T,Y,PW,M)     --USER-SUPPLIED--                           *
C* THE FIRST, DIFFUN, EVALUATES THE DERIVATIVES OF THE DEPENDENT       *
C* VARIABLES STORED IN Y(1,I) FOR I = 1 TO N, AND STORES THE           *
C* DERIVATIVES IN THE ARRAY DY.  THE NEXT TWO ARE CALLED ONLY IF THE   *
C* METHOD FLAG MF IS SET TO 1 OR 2 FOR STIFF METHODS.  DECOMP IS A     *
C* STANDARD LU DECOMPOSITION WITH PIVOTING THAT DECOMPOSES THE MATRIX  *
C* PW, LEAVING THE PIVOTS IN THE INTEGER ARRAY IP.  M IS THE DECLARED  *
C* SIZE OF PW.  IP(N) IS SET TO 0 IF PW IS SINGULAR.  SOLVE PERFORMS   *
C* BACK SUBSTITUTION ON THE CONTENTS OF CSAVE(I,1), LEAVING THE        *
C* RESULTS THERE.
C* PEDERV IS USED ONLY IF MF IS 1, AND COMPUTES THE PARTIAL            *
C* DERIVATIVES OF THE DIFFERENTIAL EQUATIONS AS DESCRIBED UNDER THE    *
C* MF PARAMETER.                                                       *
C*                                                                     *
C* THE TEMPORARY STORAGE SPACE IS PROVIDED BY THE CALLER IN THE        *
C* INTEGER ARRAY IP, THE  ARRAY PW, AND THE                            *
C* ARRAYS SAVE AND CSAVE.  THE ARRAY PW IS USED ONLY TO HOLD           *
C* THE MATRIX OF THE SAME NAME, AND SAVE IS USED TO SAVE THE VALUES    *
C* OF Y IN CASE A STEP HAS TO BE REPEATED, BUT CSAVE IS USED TO HOLD   *
C* SEVERAL ARRAYS.  THE REGIONS USED ARE                               *
C*    CSAVE(I,1)    IS USED MAINLY TO HOLD THE CORRECTION TERMS IN THE *
C*                  CORRECTOR LOOP, AND HOLDS THE DERIVATIVES DURING   *
C*                  JACOBIAN EVALUATIONS.                              *
C*    CSAVE(I,2)    IS USED TO SAVE THE VALUES OF THE SUMS OF ALL OF THE*
C*                  CORRECTION TERMS IN THE PREVIOUS STEP AFTER THEY   *
C*                  HAVE BEEN ACCUMULATED IN THE ARRAY ERROR IN THE    *
C*                  CURRENT STEP.  THIS ENABLES THE BACKWARDS DIFFERENCE*
C*                  OF ERROR TO BE FORMED.  IT IS USED TO ESTIMATE THE *
C*                  STEP SIZE FOR ONE ORDER HIGHER THAN CURRENT.       *
C*    CSAVE(I,3)    IS USED TO STORE THE DERIVATIVES WHEN THEY ARE     *
C*                  COMPUTED BY DIFFUN.                                *
C*                                                                     *
C* THE PARAMETERS TO THE SUBROUTINE DIFSBM HAVE                        *
C* THE FOLLOWING MEANINGS..                                            *
C*                                                                     *
C*   N        THE NUMBER OF FIRST ORDER DIFFERENTIAL EQUATIONS.  N     *
C*              MAY BE DECREASED ON LATER CALLS IF THE NUMBER OF       *
C*              ACTIVE EQUATIONS REDUCES, BUT IT MUST NOT BE           *
C*              INCREASED WITHOUT CALLING WITH JSTART = 0.             *
C*   T        THE INDEPENDENT VARIABLE.                                *
C*   Y        A 13 BY N ARRAY CONTAINING THE DEPENDENT VARIABLES AND   *
C*              THEIR SCALED DERIVATIVES.  Y(J+1,I) CONTAINS           *
C*              THE J-TH DERIVATIVE OF Y(I) SCALED BY                  *
```

```
C*                 H**J/FACTORIAL(J) WHERE H IS THE CURRENT                *
C*                 STEP SIZE. ONLY Y(1,I) NEED BE PROVIDED BY              *
C*                 THE CALLING PROGRAM ON THE FIRST ENTRY.                 *
C*                   IF IT IS DESIRED TO INTERPOLATE TO NON MESH POINTS    *
C*                 THESE VALUES CAN BE USED.  IF THE CURRENT STEP SIZE     *
C*                 IS H AND THE VALUE AT T + E IS NEEDED, FORM             *
C*                 S = E/H, AND THEN COMPUTE                               *
C*                                     NQ                                  *
C*                   Y(I)(T+E) =  SUM  Y(J+1,I)*S**J                       *
C*                                     J=0                                 *
C*    SAVE       A BLOCK OF AT LEAST 13*N FLOATING POINT LOCATIONS.        *
C*    CSAVE      N*3 FLOATING POINT LOCATIONS USED BY THE SUBROUTINES.     *
C*    H          THE STEP SIZE TO BE ATTEMPTED ON THE NEXT STEP.           *
C*                 H MAY BE ADJUSTED UP OR DOWN BY THE PROGRAM             *
C*                 IN ORDER TO ACHEIVE AN ECONOMICAL INTEGRATION.          *
C*                 HOWEVER, IF THE H PROVIDED BY THE USER DOES             *
C*                 NOT CAUSE A LARGER ERROR THAN REQUESTED, IT             *
C*                 WILL BE USED.  TO SAVE COMPUTER TIME, THE USER IS       *
C*                 ADVISED TO USE A FAIRLY SMALL STEP FOR THE FIRST        *
C*                 CALL.  IT WILL BE AUTOMATICALLY INCREASED LATER.        *
C*    HMIN       THE MINIMUM STEP SIZE THAT WILL BE USED FOR THE           *
C*                 INTEGRATION.  NOTE THAT ON STARTING THIS MUST BE        *
C*                 MUCH SMALLER THAN THE AVERAGE H EXPECTED SINCE          *
C*                 A FIRST ORDER METHOD IS USED INITIALLY.                 *
C*    HMAX       THE MAXIMUM SIZE TO WHICH THE STEP WILL BE INCREASED      *
C*    EPS        THE ERROR TEST CONSTANT.  SINGLE STEP ERROR ESTIMATES     *
C*                 DIVIDED BY  YMAX(I)  MUST BE LESS THAN THIS             *
C*                 IN THE EUCLIDEAN NORM. THE STEP AND/OR ORDER IS         *
C*                 ADJUSTED TO ACHEIVE THIS.                               *
C*    MF         THE METHOD INDICATOR. THE FOLLOWING ARE ALLOWED..         *
C*                         0    AN ADAMS PREDICTOR CORRECTOR IS USED.      *
C*                         1    A MULTI-STEP METHOD SUITABLE FOR STIFF     *
C*                                SYSTEMS IS USED. IT WILL ALSO WORK FOR   *
C*                                NON STIFF SYSTEMS.  HOWEVER THE USER     *
C*                                MUST PROVIDE A SUBROUTINE PEDERV WHICH   *
C*                                EVALUATES THE PARTIAL DERIVATIVES OF     *
C*                                THE DIFFERENTIAL EQUATIONS WITH RESPECT  *
C*                                TO THE Y'S.   THIS IS DONE BY CALL       *
C*                                PEDERV(T,Y,PW,M). PW IS AN N BY N ARRAY  *
C*                                WHICH MUST BE SET TO THE PARTIAL OF      *
C*                                THE I-TH EQUATION WITH RESPECT           *
C*                                TO THE J DEPENDENT VARIABLE IN PW(I,J).  *
C*                                PW IS ACTUALLY STORED IN AN M BY M       *
C*                                ARRAY WHERE M IS THE VALUE OF N USED ON  *
C*                                THE FIRST CALL TO THIS PROGRAM.          *
C*                         2    THE SAME AS CASE 1, EXCEPT THAT THIS       *
C*                                SUBROUTINE COMPUTES THE PARTIAL          *
C*                                DERIVATIVES BY NUMERICAL DIFFERENCING    *
C*                                OF THE DERIVATIVES. HENCE PEDERV IS      *
C*                                NOT CALLED.                              *
C*    YMAX       AN ARRAY OF  N LOCATIONS WHICH CONTAINS THE MAXIMUM       *
C*                 OF EACH Y SEEN SO FAR.  IT SHOULD NORMALLY BE SET TO    *
C*                 1 IN EACH COMPONENT BEFORE THE FIRST ENTRY. (SEE THE    *
C*                 DESCRIPTION OF EPS.)                                    *
C*    ERROR      AN ARRAY OF  N ELEMENTS WHICH CONTAINS THE ESTIMATED      *
```

```
C*              ONE STEP ERROR IN EACH COMPONENT.                              *
C*   KFLAG    A COMPLETION CODE WITH THE FOLLOWING MEANINGS..                  *
C*                       +1  THE STEP WAS SUCCESFUL.                           *
C*                       -1  THE STEP WAS TAKEN WITH H = HMIN, BUT THE         *
C*                              REQUESTED ERROR WAS NOT ACHIEVED.              *
C*                       -2  THE MAXIMUM ORDER SPECIFIED WAS FOUND TO          *
C*                              BE TOO LARGE.                                  *
C*                       -3  CORRECTOR CONVERGENCE COULD NOT BE                *
C*                              ACHIEVED FOR H .GT. HMIN.                      *
C*                       -4  THE REQUESTED ERROR IS SMALLER THAN CAN           *
C*                              BE HANDLED FOR THIS PROBLEM.                   *
C*   JSTART   AN INPUT INDICATOR WITH THE FOLLOWING MEANINGS..                 *
C*                       -1  REPEAT THE LAST STEP WITH A NEW H                 *
C*                        0  PERFORM THE FIRST STEP.  THE FIRST STEP           *
C*                              MUST BE DONE WITH THIS VALUE OF JSTART         *
C*                              SO THAT THE SUBROUTINE CAN INITIALIZE          *
C*                              ITSELF.                                        *
C*                       +1  TAKE A NEW STEP CONTINUING FROM THE LAST.         *
C*              JSTART IS SET TO NQ, THE CURRENT ORDER OF THE METHOD           *
C*              AT EXIT.  NQ IS ALSO THE ORDER OF THE MAXIMUM                  *
C*              DERIVATIVE AVAILABLE.                                          *
C*   MAXDER   THE MAXIMUM DERIVATIVE THAT SHOULD BE USED IN THE                *
C*              METHOD.  SINCE THE ORDER IS EQUAL TO THE HIGHEST               *
C*              DERIVATIVE USED, THIS RESTRICTS THE ORDER. IT MUST             *
C*              BE LESS THAN 13 FOR ADAMS AND 7 FOR STIFF METHODS.             *
C*   PW       A BLOCK OF AT LEAST N**2 FLOATING POINT LOCATIONS.               *
C*   IP       A BLOCK OF AT LEAST N INTEGERS.                                  *
C*******************************************************************************
      SUBROUTINE DIFSRM (N,T,Y,SAVE,CSAVE,H,HMIN,HMAX,EPS,MF,YMAX,ERROR,
     1KFLAG,JSTART,MAXDER,PW,IP)
      DIMENSION Y(13,1), YMAX(1), SAVE(13,1), ERROR(1), PW(1), A(13), PE
     1RTST(12,2,3), CSAVE(N,3), IP(1)
C*******************************************************************************
C* THE COEFFICIENTS IN PERTST ARE USED IN SELECTING THE STEP AND               *
C* ORDER, THEREFORE ONLY ABOUT ONE PERCENT ACCURACY IS NEEDED.                 *
C*******************************************************************************
      DATA PERTST/2.,4.5,7.333,10.42,13.7,17.15,1.,1.,1.,1.,1.,1.,2.,12.
     1.24.,37.89,53.33,70.08,87.97,100.9,126.7,147.3,168.8,191.4,3.,6.,9
     2.167,12.5,15.98,1.,1.,1.,1.,1.,1.,1.,12.,24.,37.89,53.33,70.08,87.
     397,100.9,126.7,147.3,168.9,191.4,211.,1.,1.,.5,.1667,.04133,.00826
     47,1.,1.,1.,1.,1.,1.,1.,1.,2.,1.,.3157,.07407,.0139,.0021818,.00029
     5,.000035,.0000037,.00000035/
      DATA A(2)/-1.0/
      IRET=1
      KFLAG=1
      IF (JSTART.LE.0) GO TO 50
C*******************************************************************************
C* BEGIN BY SAVING INFORMATION FOR POSSIBLE RESTARTS AND CHANGING              *
C* H BY THE FACTOR R IF THE CALLER HAS CHANGED H.  ALL VARIABLES               *
C* DEPENDENT ON H MUST ALSO BE CHANGED.                                        *
C* E IS A COMPARISON FOR ERRORS OF THE CURRENT ORDER NQ. EUP IS                *
C* TO TEST FOR INCREASING THE ORDER, EDWN FOR DECREASING THE ORDER.            *
C* HNEW IS THE STEP SIZE THAT WAS USED ON THE LAST CALL.                       *
C*******************************************************************************
   10 DO 20 I=1,N
```

```
      DO 20 J=1,K
   20   SAVE(J,I)=Y(J,I)
      HOLD=HNEW
      IF (H.EQ.HOLD) GO TO 40
   30 IRET1=1
      GO TO 750
   40 NQOLD=NQ
      TOLD=T
      IF (JSTART.GT.0) GO TO 300
      GO TO 80
   50 IF (JSTART.EQ.-1) GO TO 70
C**********************************************************************
C* ON THE FIRST CALL, THE ORDER IS SET TO 1 AND THE INITIAL           *
C* DERIVATIVES ARE CALCULATED.                                        *
C**********************************************************************
      RR=1.0
      NQ=1
      N3=N
      N4=N**2
      CALL DIFFUN (N,T,Y,CSAVE(1,3))
      DO 60 I=1,N
   60   Y(2,I)=CSAVE(I,3)*H
      HNEW=H
      K=2
      GO TO 10
C**********************************************************************
C* REPEAT LAST STEP BY RESTORING SAVED INFORMATION.                   *
C**********************************************************************
   70 IF (NQ.EQ.NQOLD) JSTART=1
      T=TOLD
      NQ=NQOLD
      K=NQ+1
      GO TO 30
C**********************************************************************
C* SET THE COEFFICIENTS THAT DETERMINE THE ORDER AND THE METHOD       *
C* TYPE.  CHECK FOR EXCESSIVE ORDER.  THE LAST TWO STATEMENTS OF      *
C* THIS SECTION  SET IWEVAL .GT.0 IF PW IS TO BE RE-EVALUATED         *
C* BECAUSE OF THE ORDER CHANGE, AND THEN REPEAT THE INTEGRATION       *
C* STEP IF IT HAS NOT YET BEEN DONE (IRET = 1) OR SKIP TO A FINAL     *
C* SCALING BEFORE EXIT IF IT HAS BEEN COMPLETED (IRET = 2).           *
C**********************************************************************
   80 IF (MF.EQ.0) GO TO 90
      IF (NQ.GT.6) GO TO 100
      GO TO (230,240,250,260,270,280), NQ
   90 IF (NQ.GT.12) GO TO 100
      GO TO (110,120,130,140,150,160,170,180,190,200,210,220), NQ
  100 KFLAG=-2
      RETURN
C**********************************************************************
C* THE FOLLOWING COEFFICIENTS SHOULD BE DEFINED TO THE MAXIMUM        *
C* ACCURACY PERMITTED BY THE MACHINE.  THEY ARE, IN THE ORDER USED..  *
C*                                                                    *
C* -1                                                                 *
C* -1/2,-1/2                                                          *
C* -5/12,-3/4,-1/6                                                    *
```

```
C* -3/8,-11/12,-1/3,-1/24                                                   *
C* -251/720,-25/24,-35/72,-5/48,-1/120                                      *
C* -95/288,-137/120,-5/8,-17/96,-1/40,-1/720                                *
C* -19087/60480,-49/40,-203/270,-49/192,-7/144,-7/1440,-1/5040              *
C* -5257/17280,-363/280,-469/540,-967/2880,-7/90,-23/2160,-1/1260,
C*      -1/40320
C* -1070017/3628800,-761/560,-29531/30240,-267/640,-1069/9600,-3/160,
C*      -13/6720,-1/8960,-1/362880
C* -4165506/14515200,-7129/5040,-6515/6048,-4523/9072,-19/128,
C*        -3013/1036800,-5/1344,-29/96768,-1/72576,-1/3628800
C* -134211265/479001600,-7381/5040,-177133/151200,-84095/145152,
C*      -341693/1814400,-8591/207360,-7513/1209600,-121/193536,
C*           -11/272160,-11/7257600,-1/39916800
C* -262747265/958003200,-83711/55440,-190553/151200,-341747/518400,
C*      -139381/604800,-2667907/47900160,-1903/201600,-10831/9676800,
C*           -11/120960,-1/207369,-1/6652800,-1/479001600
C*                                                                          *
C*                                                                          *
C* -1                                                                       *
C* -2/3,-1/3                                                                *
C* -12/25,-7/10,-1/5,-1/50                                                  *
C* -120/274,-225/274,-85/274,-15/274,-1/274                                 *
C* -180/441,-58/63,-15/36,-25/252,-3/252,-1/1764                            *
C***************************************************************************
  110 A(1)=-1.0
      GO TO 290
  120 A(1)=-.50000000000
      A(3)=-0.500000000
      GO TO 290
  130 A(1)=-0.4166666666666667
      A(3)=-0.750000000
      A(4)=-0.1666666666666667
      GO TO 290
  140 A(1)=-0.3750000000000
      A(3)=-0.9166666666666667
      A(4)=-0.3333333333333333
      A(5)=-0.0416666666666667
      GO TO 290
  150 A(1)=-0.3486111111111111
      A(3)=-1.0416666666666667
      A(4)=-0.4861111111111111
      A(5)=-0.1041666666666667
      A(6)=-0.00833333333333333
      GO TO 290
  160 A(1)=-0.3298611111111111
      A(3)=-1.1416666666666667
      A(4)=-0.625000000
      A(5)=-0.1770833333333333
      A(6)=-0.0250000000
      A(7)=-0.00138888888888889
      GO TO 290
  170 A(1)=-0.3155919312169312
      A(3)=-1.225000000
      A(4)=-0.7518518518518519
      A(5)=-0.2552083333333333
```

```
      A(6)=-0.04861111111111111
      A(7)=-0.004861111111111111
      A(8)=-0.0001984126984126984
      GO TO 290
  180 A(1)=-0.3042245370370370
      A(3)=-1.296428571428485
      A(4)=-0.8685185185185185
      A(5)=-0.3357638888888888
      A(6)=-0.07777777777777777
      A(7)=-0.01064814814814815
      A(8)=-0.0007936507936507937
      A(9)=-0.00002480158730158730
      GO TO 290
  190 A(1)=-0.2948680004409171
      A(3)=-1.358928571428557
      A(4)=-0.976554232804233
      A(5)=-0.41718750000
      A(6)=-0.111354166666667
      A(7)=-0.0187500000
      A(8)=-0.001934523809523809
      A(9)=-0.0001116071428571043
      A(10)=-0.00000275573192239859
      GO TO 290
  200 A(1)=-0.2869754464285714
      A(3)=-1.41448412698413
      A(4)=-1.07721560846561
      A(5)=-0.498567019400353
      A(6)=-0.148437500000
      A(7)=-0.0290605709876543
      A(8)=-0.00372023809523810
      A(9)=-0.000299685846560847
      A(10)=-0.0000137786596119929
      A(11)=-0.000000275573192239860
      GO TO 290
  210 A(1)=-0.280189564439367
      A(3)=-1.46448412698413
      A(4)=-1.17151455026455
      A(5)=-0.579358190035273
      A(6)=-0.188322861552028
      A(7)=-0.0414303626543210
      A(8)=-0.00621114479894180
      A(9)=-0.000625206679894180
      A(10)=-0.0000404174015285126
      A(11)=-0.00000151565255731922
      A(12)=-0.0000000250521083854417
      GO TO 290
  220 A(1)=-0.274265540031599
      A(3)=-1.50993867243867
      A(4)=-1.26027116402116
      A(5)=-0.659234182098765
      A(6)=-0.230458002645503
      A(7)=-0.0556972461052322
      A(8)=-0.00943948412698413
      A(9)=-0.00111927496693122
      A(10)=-0.0000909391534391534
```

```
      A(11)=-0.00000482253086419753
      A(12)=-0.000000150312650312650
      A(13)=-0.00000000208767569878681
      GO TO 290
  230 A(1)=-1.000000000
      GO TO 290
  240 A(1)=-0.6666666666666667
      A(3)=-0.3333333333333333
      GO TO 290
  250 A(1)=-0.5454545454545455
      A(3)=A(1)
      A(4)=-0.0909090909090909091
      GO TO 290
  260 A(1)=-0.480000000
      A(3)=-0.700000000
      A(4)=-0.200000000
      A(5)=-0.020000000
      GO TO 290
  270 A(1)=-0.437956204379562
      A(3)=-0.8211678832116788
      A(4)=-0.3102189781021898
      A(5)=-0.05474452554744526
      A(6)=-0.003649635036496350 4
      GO TO 290
  280 A(1)=-0.4081632653061225
      A(3)=-0.9206349206349206
      A(4)=-0.4166666666666667
      A(5)=-0.0992063492063492
      A(6)=-0.0119047619047619
      A(7)=-0.000566893424036282
  290 K=NQ+1
      IDOUB=K
      MTYP=(4-MF)/2
      ENQ2=.5/FLOAT(NQ+1)
      ENQ3=.5/FLOAT(NQ+2)
      ENQ1=0.5/FLOAT(NQ)
      PEPSH=EPS
      EUP=(PERTST(NQ,MTYP,2)*PEPSH)**2
      E=(PERTST(NQ,MTYP,1)*PEPSH)**2
      EDWN=(PERTST(NQ,MTYP,3)*PEPSH)**2
      IF (EDWN.EQ.0) GO TO 780
      BND=(EPS*ENQ3)**2
      IWEVAL=MF
      GO TO (300,670), IRET
C**********************************************************************
C* THIS SECTION COMPUTES THE PREDICTED VALUES BY EFFECTIVELY          *
C* MULTIPLYING THE SAVED INFORMATION BY THE PASCAL TRIANGLE           *
C* MATRIX.                                                            *
C**********************************************************************
  300 T=T+H
      DO 310 J=2,K
        DO 310 J1=J,K
        J2=K-J1+J-1
        DO 310 I=1,N
  310   Y(J2,I)=Y(J2,I)+Y(J2+1,I)
```

```
C*********************************************************************
C*   UP TO 2 CORRECTOR ITERATIONS ARE TAKEN.  CONVERGENCE IS TESTED BY *
C*   REQUIRING THE L2 NORM OF CHANGES TO BE LESS THAN BND WHICH IS     *
C*   DEPENDENT ON THE ERROR TEST CONSTANT.                             *
C*      THE SUM OF THE CORRECTIONS IS ACCUMULATED IN THE ARRAY         *
C*   ERROR(I).  IT IS EQUAL TO THE K-TH DERIVATIVE OF Y MULTIPLIED     *
C*   BY  H**K/(FACTORIAL(K-1)*A(K)), AND IS THEREFORE PROPORTIONAL     *
C*   TO THE ACTUAL ERRORS TO THE LOWEST POWER OF H PRESENT. (H**K)     *
C*********************************************************************
      DO 320 I=1,N
  320   ERROR(I)=0.0
      DO 430 L=1,2
        CALL DIFFUN (N,T,Y,CSAVE(1,3))
C*********************************************************************
C*   IF THERE HAS BEEN A CHANGE OF ORDER OR THERE HAS BEEN TROUBLE     *
C*   WITH CONVERGENCE, PW IS RE-EVALUATED PRIOR TO STARTING THE        *
C*   CORRECTOR ITERATION IN THE CASE OF STIFF METHODS.  IWEVAL IS      *
C*   THEN SET TO -1 AS AN INDICATOR THAT IT HAS BEEN DONE.             *
C*********************************************************************
        IF (IWEVAL.LT.1) GO TO 390
        IF (MF.EQ.2) GO TO 360
        CALL PEDERV (T,Y,PW,N3)
        R=A(1)*H
        DO 330 I=1,N4
  330     PW(I)=PW(I)*R
C*********************************************************************
C* ADD THE IDENTITY MATRIX TO THE JACOBIAN AND DECOMPOSE INTO LU = PW  *
C*********************************************************************
  340   DO 350 I=1,N
  350     PW(I*(N3+1)-N3)=1.0+PW(I*(N3+1)-N3)
        IWEVAL=-1
        CALL NDI01Z (N,N3,PW,IP)
        IF (IP(N).NE.0) GO TO 390
        GO TO 440
C*********************************************************************
C* EVALUATE THE JACOBIAN INTO PW BY NUMERICAL DIFFERENCING.  R IS THE  *
C* CHANGE MADE TO THE ELEMENT OF Y.  IT IS EPS RELATIVE TO Y WITH      *
C* A MINIMUM OF EPS**2.  F STORES THE UNCHANGED VALUE OF Y.            *
C*********************************************************************
  360   DO 380 J=1,N
          F=Y(1,J)
          R=EPS*AMAX1(EPS,ABS(F))
          Y(1,J)=Y(1,J)+R
          D=A(1)*H/R
          CALL DIFFUN (N,T,Y,CSAVE(1,1))
          DO 370 I=1,N
  370       PW(I+(J-1)*N3)=(CSAVE(I,1)-CSAVE(I,3))*D
  380     Y(1,J)=F
        GO TO 340
  390   DO 400 I=1,N
  400     CSAVE(I,1)=Y(2,I)-CSAVE(I,3)*H
        IF (MF.EQ.0) GO TO 410
        CALL NDI02Z (N,N3,PW,CSAVE(1,1),IP)
C*********************************************************************
C* CORRECT AND COMPARE DEL, THE L2 NORM OF CHANGE/YMAX, WITH BND.      *
```

```
C* ESTIMATE THE VALUE OF THE L2 NORM OF THE NEXT CORRECTION BY          *
C* BR*2*DEL AND COMPARE WITH BND.  IF EITHER IS LESS, THE CORRECTOR     *
C* IS SAID TO HAVE CONVERGED.                                           *
C***********************************************************************
  410   DEL=0.0
        DO 420 I=1,N
          Y(1,I)=Y(1,I)+A(1)*CSAVE(I,1)
          Y(2,I)=Y(2,I)-CSAVE(I,1)
          ERROR(I)=ERROR(I)+CSAVE(I,1)
          DEL=DEL+(CSAVE(I,1)/YMAX(I))**2
  420     CONTINUE
        IF (L.GE.2) BR=AMAX1(.9*BR,DEL/DEL1)
        DEL1=DEL
        IF (AMIN1(DEL,BR*DEL*2.0).LE.BND) GO TO 480
  430   CONTINUE
C***********************************************************************
C* THE CORRECTOR ITERATION FAILED TO CONVERGE IN 2 TRIES.  VARIOUS      *
C* POSSIBILITIES ARE CHECKED FOR.  IF H IS ALREADY HMIN AND             *
C* THIS IS EITHER ADAMS METHOD OR THE STIFF METHOD IN WHICH THE         *
C* MATRIX PW HAS ALREADY BEEN RE-EVALUATED, A NO CONVERGENCE EXIT       *
C* IS TAKEN. OTHERWISE THE MATRIX PW IS RE-EVALUATED AND/OR THE         *
C* STEP IS REDUCED TO TRY AND GET CONVERGENCE.                          *
C***********************************************************************
  440 T=TOLD
      IF (ABS(H).LE.HMIN*1.00001.AND.(IWEVAL-MTYP).LT.-1) GO TO 450
      IF ((MF.EQ.0).OR.(IWEVAL.NE.0)) H=H*0.25
      IWEVAL=MF
      IRET1=2
      GO TO 750
  450 KFLAG=-3
      NQ=NQOLD
  460 DO 470 I=1,N
        DO 470 J=1,K
  470   Y(J,I)=SAVE(J,I)
      H=HOLD
      JSTART=NQ
      RETURN
C***********************************************************************
C* THE CORRECTOR CONVERGED AND CONTROL IS PASSED TO STATEMENT 520       *
C* IF THE ERROR TEST IS O.K.,  AND TO 540 OTHERWISE.                    *
C* IF THE STEP IS O.K. IT IS ACCEPTED. IF IDOUB HAS BEEN REDUCED        *
C* TO ONE,  A TEST IS MADE TO SEE IF THE STEP CAN BE INCREASED          *
C* AT THE CURRENT ORDER OR BY GOING TO ONE HIGHER OR ONE LOWER.         *
C* SUCH A CHANGE IS ONLY MADE IF THE STEP CAN BE INCREASED BY AT        *
C* LEAST 1.1.  IF NO CHANGE IS POSSIBLE IDOUB IS SET TO 8 TO            *
C* PREVENT FUTHER TESTING FOR 8 STEPS.                                  *
C* IF A CHANGE IS POSSIBLE, IT IS MADE AND IDOUB IS SET TO              *
C* NQ + 1     TO PREVENT FURTHER TESING FOR THAT NUMBER OF STEPS.       *
C* IF THE ERROR WAS TOO LARGE, THE OPTIMUM STEP SIZE FOR THIS OR        *
C* LOWER ORDER IS COMPUTED, AND THE STEP RETRIED.  IF IT SHOULD         *
C* FAIL TWICE MORE IT IS AN INDICATION THAT THE DERIVATIVES THAT        *
C* HAVE ACCUMULATED IN THE Y ARRAY HAVE ERRORS OF THE WRONG ORDER       *
C* SO THE FIRST DERIVATIVES ARE RECOMPUTED AND THE ORDER IS SET         *
C* TO 1.                                                                *
C***********************************************************************
```

```
  480 D=0.0
      DO 490 I=1,N
  490   D=D+(ERROR(I)/YMAX(I))**2
      IWEVAL=0
      IF (D.GT.E) GO TO 530
      IF (K.LT.3) GO TO 510
C*********************************************************************
C* COMPLETE THE CORRECTION OF THE HIGHER ORDER DERIVATIVES AFTER A   *
C* SUCCESFUL STEP.                                                   *
C*********************************************************************
      DO 500 J=3,K
        DO 500 I=1,N
  500   Y(J,I)=Y(J,I)+A(J)*ERROR(I)
  510 KFLAG=+1
      HNEW=H
      IF (IDOUB.LE.1) GO TO 540
      IDOUB=IDOUB-1
      IF (IDOUB.GT.1) GO TO 690
      DO 520 I=1,N
  520   CSAVE(I,2)=ERROR(I)
      GO TO 690
C*********************************************************************
C* REDUCE THE FAILURE FLAG COUNT TO CHECK FOR MULTIPLE FAILURES.     *
C* RESTORE T TO ITS ORIGINAL VALUE AND TRY AGAIN UNLESS THERE HAVE   *
C* THREE FAILURES.  IN THAT CASE THE DERIVATIVES ARE ASSUMED TO HAVE *
C* ACCUMULATED ERRORS SO A RESTART FROM THE CURRENT VALUES OF Y IS   *
C* TRIED. THIS IS CONTINUED UNTIL SUCCESS OR H = HMIN.               *
C*********************************************************************
  530 KFLAG=KFLAG-2
      IF (ABS(H).LE.(HMIN*1.00001)) GO TO 740
      T=TOLD
      IF (KFLAG.LE.-5) GO TO 710
C*********************************************************************
C* PR1, PR2, AND PR3  WILL CONTAIN THE AMOUNTS BY WHICH THE STEP SIZE *
C* SHOULD BE DIVIDED AT ORDER ONE LOWER, AT THIS ORDER, AND AT ORDER *
C* ONE HIGHER RESPECTIVELY.                                          *
C*********************************************************************
  540 PR2=(D/E)**ENQ2*1.2
      PR3=1.E+20
      IF ((NQ.GE.MAXDER).OR.(KFLAG.LE.-1)) GO TO 560
      D=0.0
      DO 550 I=1,N
  550   D=D+((ERROR(I)-CSAVE(I,2))/YMAX(I))**2
      PR3=(D/EUP)**ENQ3*1.4
  560 PR1=1.E+20
      IF (NQ.LE.1) GO TO 580
      D=0.0
      DO 570 I=1,N
  570   D=D+(Y(K,I)/YMAX(I))**2
      PR1=(D/EDWN)**ENQ1*1.3
  580 CONTINUE
      IF (PR2.LE.PR3) GO TO 640
      IF (PR3.LT.PR1) GO TO 650
  590 R=1.0/AMAX1(PR1,1.E-4)
      NEWQ=NQ-1
```

```
  600 IDOUB=8
      IF ((KFLAG.EQ.1).AND.(R.LT.(1.1))) GO TO 690
      IF (NEWQ.LE.NQ) GO TO 620
C***********************************************************************
C* COMPUTE ONE ADDITIONAL SCALED DERIVATIVE IF ORDER IS INCREASED.     *
C***********************************************************************
      DO 610 I=1,N
  610   Y(NEWQ+1,I)=ERROR(I)*A(K)/FLOAT(K)
  620 K=NEWQ+1
      IF (KFLAG.EQ.1) GO TO 660
      H=H*R
      IRET1=3
      GO TO 750
  630 IF (NEWQ.EQ.NQ) GO TO 300
      NQ=NEWQ
      GO TO 80
  640 IF (PR2.GT.PR1) GO TO 590
      NEWQ=NQ
      R=1.0/AMAX1(PR2,1.E-4)
      GO TO 600
  650 R=1.0/AMAX1(PR3,1.E-4)
      NEWQ=NQ+1
      GO TO 600
  660 IRET=2
      R=AMIN1(R,HMAX/ABS(H))
      H=H*R
      HNEW=H
      IF (NQ.EQ.NEWQ) GO TO 670
      NQ=NEWQ
      GO TO 80
  670 R1=1.0
      DO 680 J=2,K
        R1=R1*R
        DO 680 I=1,N
  680   Y(J,I)=Y(J,I)*R1
      IDOUB=K
  690 DO 700 I=1,N
  700   YMAX(I)=AMAX1(YMAX(I),ABS(Y(1,I)))
      JSTART=NQ
      RETURN
  710 IF (NQ.GT.1) GO TO 720
      IF (ABS(H).LE.2.*HMIN) GO TO 780
      GO TO 540
  720 R=H/HOLD
      DO 730 I=1,N
        Y(1,I)=SAVE(1,I)
  730   Y(2,I)=SAVE(2,I)*R
      NQ=1
      KFLAG=1
      GO TO 80
  740 KFLAG=-1
      HNEW=H
      JSTART=NQ
      RETURN
C***********************************************************************
```

```
C* THIS SECTION SCALES ALL VARIABLES CONNECTED WITH H AND RETURNS      *
C* TO THE ENTERING SECTION.                                            *
C**********************************************************************
  750 H=SIGN(AMAX1(HMIN,AMIN1(ABS(H),HMAX)),H)
      R1=1.0
      DO 760 J=2,K
        R=H/HOLD
        R1=R1*R
        DO 760 I=1,N
  760   Y(J,I)=SAVF(J,I)*R1
      DO 770 I=1,N
  770   Y(1,I)=SAVF(1,I)
      IDOUB=K
      GO TO (40,300,630), IRET1
  780 KFLAG=-4
      GO TO 460
      END
      SUBROUTINE NDI01Z (N,NDIM,A,IP)
C--MOLER'S "DECOMP".
C
C     MATRIX TRIANGULARIZATION BY GAUSSIAN ELIMINATION.
C
C    INPUT...
C      N = ORDER OF MATRIX
C      NDIM = DECLARED DIMENSION OF ARRAY A.
C      A = MATRIX TO BE TRIANGULARIZED. (FOR STIFF METHODS, A IS SINGLE
C                  PRECISION;  ALL OTHER VARIABLES ARE DOUBLE PRECISION.)
C    OUTPUT...
C      A(I,J),  I.LE.J  = UPPER TRIANGULAR FACTOR, U.
C      A(I,J),   I.GT.J   = MULTIPLIERS  = LOWER TRIANGULAR FACTOR, I-L.
C      IP(K),  K.LT.N = INDEX OF K-TH PIVOT ROW.
C      IP(N)  = (-1)**(NUMBER OF INTERCHANGES) OR 0.
C    USE 'SOLVE' (NDI02Z)  TO OBTAIN SOLUTION OF LINEAR SYSTEM.
C    DETERM(A) = IP(N)*A(1,1)*A(2,2)*...*A(N,N).
C    IF IP(N) = 0, A IS SINGULAR, SOLVE WILL DIVIDE BY ZERO.
C
      DIMENSION A(NDIM,NDIM), IP(NDIM)
      IP(N)=1
      DO 50 K=1,N
        IF (K.EQ.N) GO TO 50
        KP1=K+1
        M=K
        DO 10 I=KP1,N
          IF (ABS(A(I,K)).GT.ABS(A(M,K))) M=I
   10     CONTINUE
        IP(K)=M
        IF (M.NE.K) IP(N)=-IP(N)
        T=A(M,K)
        A(M,K)=A(K,K)
        A(K,K)=T
        IF (T.EQ.0) GO TO 50
        DO 20 I=KP1,N
   20     A(I,K)=-A(I,K)/T
        DO 40 J=KP1,N
          T=A(M,J)
```

```
      A(M,J)=A(K,J)
      A(K,J)=T
      IF (T.EQ.0.) GO TO 40
      DO 30 I=KP1,N
30      A(I,J)=A(I,J)+A(I,K)*T
40    CONTINUE
50  IF (A(K,K).EQ.0.) IP(N)=0
60  CONTINUE
    RETURN
    END
    SUBROUTINE NDI02Z (N,NDIM,A,B,IP)
C--MOLER'S "SOLVE".
C
C   SOLUTION OF LINEAR SYSTEM, A*X = B.
C
C   INPUT...
C     N = ORDER OF MATRIX.
C     NDIM = DECLARED DIMENSION OF ARRAY A.
C     A = TRIANGULARIZED MATRIX OBTAINED FROM 'DECOMP' (NDI01Z).
C     B = RIGHT HAND SIDE VECTOR.
C     IP = PIVOT VECTOR OBTAINED FROM 'DECOMP'.
C   OUTPUT...
C     B = SOLUTION VECTOR, X.
    DIMENSION A(NDIM,NDIM), B(NDIM), IP(NDIM)
    IF (N.EQ.1) GO TO 30
    NM1=N-1
    DO 10 K=1,NM1
      KP1=K+1
      M=IP(K)
      T=B(M)
      B(M)=B(K)
      B(K)=T
      DO 10 I=KP1,N
10    B(I)=B(I)+A(I,K)*T
    DO 20 KB=1,NM1
      KM1=N-KB
      K=KM1+1
      B(K)=B(K)/A(K,K)
      T=-B(K)
      DO 20 I=1,KM1
20    B(I)=B(I)+A(I,K)*T
30 B(1)=B(1)/A(1,1)
    RETURN
    END
```

1 Report No NASA CR-165816	2 Government Accession No	3 Recipient's Catalog No
4 Title and Subtitle A NUMERICAL SIMULATION OF THE DISPERSAL OF AERIAL SPRAYS		5 Report Date November 1981
		6 Performing Organization Code
7 Author(s) M B BRAGG		8 Performing Organization Report No AARL TR 81-0
9 Performing Organization Name and Address The Ohio State University, Aeronautical and Astronautical Research Lab, 2300 West Case Road, Columbus, Ohio 43220		10 Work Unit No
		11 Contract or Grant No L-7965B
12 Sponsoring Agency Name and Address National Aeronautics and Space Administration, Washington, DC 20546		13 Type of Report and Period Covered Contractor report
		14 Sponsoring Agency Code

15 Supplementary Notes

Langley technical monitor Dana J Morris

16 Abstract

A computer program has been developed to predict the trajectory, ground deposition, and drift of liquid sprays injected into the wake of an aircraft in ground effect The program uses a horseshoe vortex wake model and includes the effects of liquid droplet evaporation, crosswind, the propeller slipstream, ground effect, and tunnel walls on small scale models. This paper is intended as a user's guide and includes several sample cases demonstrating the various user options. A complete listing of the FORTRAN computer program is provided.

17 Key Words (Suggested by Author(s))	18 Distribution Statement
Aerial Application Technology; Wake Vortex - Dispersal Interaction; Computer Simulation	FEDD Distribution

19 Security Classif (of this report)	20 Security Classif (of this page)	21 No of Pages	22 Price
Unclassified	Unclassified	109	

Available: NASA's Industrial Applications Centers

End of Document