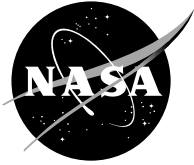# An Interactive Microsoft® Excel Program for Tracking a Single Evaporating Droplet in Crossflow

K.H. Liew, E. Urip, and S.L. Yang
Michigan Technological University, Houghton, Michigan

C.J. Marek
Glenn Research Center, Cleveland, Ohio

The NASA STI Program Office . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) Program Office plays a key part in helping NASA maintain this important role.

The NASA STI Program Office is operated by Langley Research Center, the Lead Center for NASA's scientific and technical information. The NASA STI Program Office provides access to the NASA STI Database, the largest collection of aeronautical and space science STI in the world. The Program Office is also NASA's institutional mechanism for disseminating the results of its research and development activities. These results are published by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA's counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services that complement the STI Program Office's diverse offerings include creating custom thesauri, building customized databases, organizing and publishing research results . . . even providing videos.

For more information about the NASA STI Program Office, see the following:

- Access the NASA STI Program Home Page at *http://www.sti.nasa.gov*

- E-mail your question via the Internet to help@sti.nasa.gov

- Fax your question to the NASA Access Help Desk at 301–621–0134

- Telephone the NASA Access Help Desk at 301–621–0390

- Write to:
  NASA Access Help Desk
  NASA Center for AeroSpace Information
  7121 Standard Drive
  Hanover, MD 21076
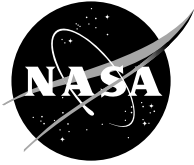
An Interactive Microsoft® Excel Program for Tracking a Single Evaporating Droplet in Crossflow

K.H. Liew, E. Urip, and S.L. Yang
Michigan Technological University, Houghton, Michigan

C.J. Marek
Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center

July 2004

The interactive Microsoft® Excel spreadsheet entitled, "An Interactive Microsoft® Excel Program for Tracking a Single Evaporating Droplet in Crossflow" that computes and plots data with and without spray evaporation is available on CD-ROM as a separate document and can also be accessed from the URL's located on page 17 of this report. The CD-ROM also contains NASA/TM—2002-211710 which includes an interactive spreadsheet that computes and plots data without spray evaporation.

An Interactive Microsoft® Excel Program for Tracking a Single Evaporating Droplet in Crossflow

K.H. Liew, E. Urip, and S.L. Yang
Michigan Technological University
Houghton, Michigan 49931
Ph.: 906–487–2624
Fax: 906–487–2822
E-mail: slyang@mtu.edu

C.J. Marek
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135
Ph.: 216–433–3584
Fax: 216–433–3000
E-mail: Cecil.J.Marek@nasa.gov

Abstract

Droplet interaction with a high temperature gaseous crossflow is important because of its wide application in systems involving two phase mixing such as in combustion requiring quick mixing of fuel and air. The focus of this work is to investigate dispersion of a two-dimensional evaporating spray into a crossflow.

An interactive Microsoft® Excel program for tracking a single droplet in crossflow that has previously been developed was modified to include droplet evaporation computation. In addition to the high velocity airflow, the injected droplets are also subjected to increased combustor temperature and pressure that affect their motion in the flow field. Six ordinary differential equations (namely the time rate of change of $x, z, u_d, w_d, D, and T_s$) are then solved by 4th-order Runge-Kutta method using Microsoft® Excel software.

Visual Basic programming and Excel macrocode are used to calculate the data and plot the droplet's motion in the flow field. This program computes and plots the data sequentially without forcing the user to open other types of plotting programs. A user's manual on how to use the program is included.

Symbol List

A_d Projected area of the droplet

B_M Mass transfer number

B_T Heat transfer number

C_{exp} Expansion coefficient

C_D Drag coefficient

c_p Specific heat at constant pressure

$c_{p,A}$ Specific heat of air

$c_{p,F}$ Specific heat of liquid fuel

$c_{p,Fv}$ Specific heat of fuel vapor

$c_{p,g}$ Specific heat of fuel-air mixture

D Droplet diameter

D_{FA} Diffusion coefficient of fuel in air

D_0 Initial diameter of droplet

F Force

g Gravitational acceleration constant

k_A Thermal conductivity of air

k_{Fv} Thermal conductivity of fuel vapor

k_g Thermal conductivity of fuel-air mixture

L Latent heat of vaporization

L_{Tbn} Latent heat at normal boiling point

M_A Molecular weight of air

M_F Molecular weight of fuel

m_{drop} Mass of a droplet

$\dot{m}$ Evaporation rate

$\dot{m}''$ Evaporation rate per unit surface area

n Exponential constant

Nu Nusselt number

P Ambient pressure

P_F Partial pressure of fuel

P_{vap} Vapor pressure

Pr_A Prandtl number of air

Q_{act} Total heat transfer to droplet

Q_{ss} Steady-state heat transfer to droplet

R_u Universal gas constant

Re_d Reynolds number of droplet

r Spherical radius of the droplet

r_{32} 0.5*(Sauter mean radius)

T_{bn} Normal boiling point

T_{crit} Critical temperature

T_{drop} Droplet temperature

T_r Reference temperature

T_s Droplet surface temperature

T_∞ Ambient temperature

t_e Evaporation time

U_R Relative velocity between the droplet and the gas stream

u_A Velocity of the cross stream (air) in the x-direction

u_d Velocity of the droplet in the x-direction

V_d Droplet Volume

w_A Velocity of the cross stream (air) in the z-direction

w_d Velocity of the droplet in the z-direction

x Horizontal direction (+ equals to the right)

Y_A Mass fraction of air

Y_F Mass fraction of fuel

Z Vertical direction (+ equals up)

α_g Thermal diffusivity of gas

λ Evaporation constant

μ_A Viscosity constant of air

ρ_A Density of the cross stream (air)

ρ_d Density of the droplet

$\rho_{F,288.6K}$ Density of fuel droplet at 288.6K

Δt time step-size

Subscripts

A Air

d Droplet

F Fuel

g Mixture of gaseous phase

r Reference condition

s Droplet surface

st Steady state

1. Introduction

A liquid spray injected into a gaseous crossflow with high temperature is important because of its wide application in systems involving two phase mixing. It is important to be able to compute this flow to optimize the mixing strategy.

An existing Excel program (ref. 1) has previously been developed for tracking a single droplet in crossflow computation. This work is focused on producing a quick computational method for determining spray penetration with evaporation. With this spreadsheet, one can investigate the dispersion of an air-blast atomized spray jet into a high temperature crossflow. During the transverse injection of a spray into high velocity airflow, the droplets (carried along in the gaseous stream of co-flowing air) are not only subjected to forces due to the crossflow motion, but also to increases in the combustor temperature and pressure (see fig. 1).
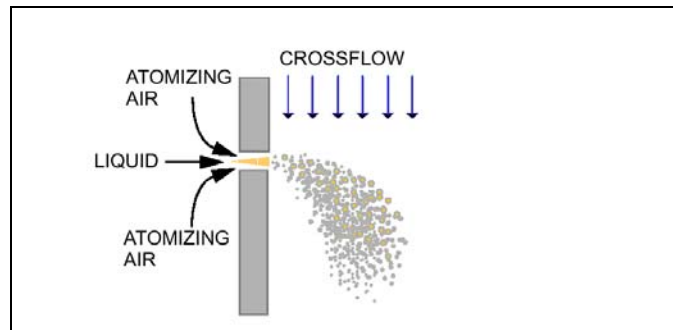

Figure 1.1 (from NASA/CR—2000-210467) (ref. 2)

2. Governing Equations

2.1 Droplet Trajectories and Velocities

The trajectories of the droplets can be tracked by applying a Lagrangian-based analysis to the droplets. The momentum equations for a droplet can be obtained by equating the droplet motion to: (1) The viscosity and pressure-related drag forces, and (2) The pressure gradient and viscous forces related to the fluid surrounding the droplet, and (3) The inertia of the virtual mass, induced when the particle acceleration affects the fluid mass acceleration.

Droplet trajectory and velocity with respect to time can be calculated. Based on these principles along with the following assumptions:

  (1) The droplets are spherical,
  (2) No droplet breakup occurs,
  (3) Vaporization is not considered yet and will be derived separately in section 2.2,
  (4) Lift, virtual mass, and Basset forces which takes into account the acceleration history of the droplet, are neglected,
  (5) Chemical reaction is not included,

These assumptions reduce the droplet momentum equation to include only the effects of the drag and body forces. The general momentum equations for a single droplet injected along the positive x-direction, transversely into a downward-flowing air stream in the positive z-direction, as shown in figure 2, is described by

$$\vec{F}_d = \vec{F}_{drag} + \vec{F}_{body} \tag{1}$$

where the net force $\vec{F}_d$ that drives the droplet motion is balanced by the drag force opposing its motion, and the field forces acting on the droplet. The aerodynamic drag force is given by

$$\vec{F}_{drag} = -\frac{1}{2}\rho_g \vec{U}_R |\vec{U}_R| A_d C_D \tag{2}$$

where ρ_A is the air density, and A_d and C_D, the projected area and the drag coefficient of the droplet, respectively. The relative velocity between the droplet and the crossflow has a magnitude of U_R (see fig. 2). The subscript "d" refers to the droplet and "g" the crossflow air.
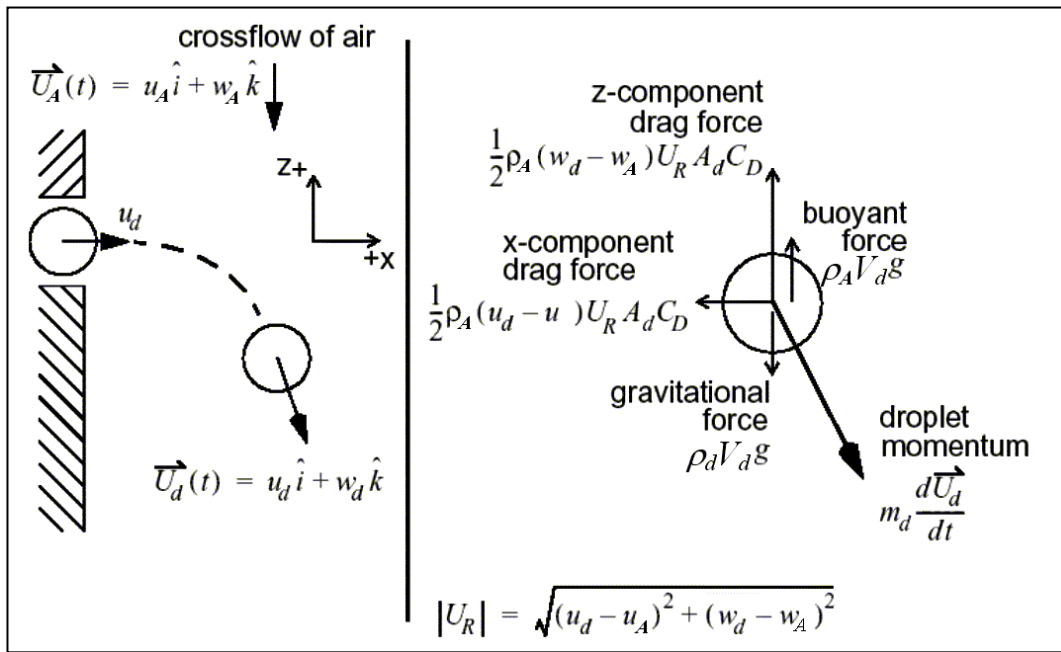

Figure 2.1 (from NASA/CR—2000-210467) (ref. 2)

The body force, resulting from an equivalent volume of air that buoys the droplet, includes the gravitational and buoyancy forces. It is given by

$$\vec{F}_{body} = (\rho_d - \rho_A)V_d\,\vec{g} \tag{3}$$

which says that the body force is equal to the product of relative droplet and air density (ρ_d - ρ_A), the droplet volume V_d, and the gravitational acceleration g.

Substituting equation (2) and equation (3) to equation (1) yields:

$$\rho_d V_d \frac{du_d}{dt} = -\frac{1}{2}\rho_A(u_d - u_A)\ \left|\vec{U}_R\right| A_d C_D \tag{4}$$

$$\rho_d V_d \frac{dw_d}{dt} = -\frac{1}{2}\rho_A(w_d - w_A)\ \left|\vec{U}_R\right| A_d C_D + (\rho_A - \rho_d)V_d\ g \tag{5}$$

$$\frac{dx}{dt} = u_d \tag{6}$$

$$\frac{dz}{dt} = w_d \tag{7}$$

The drag coefficient of the droplet depends on the droplet Reynolds number and is given by

$$C_D = \begin{cases} \dfrac{24}{\mathrm{Re}_d}\left[1+\dfrac{1}{6}\mathrm{Re}_d^{\,2/3}\right] & \mathrm{Re}_d \leq 1000 \\[2em] 0.424 & \mathrm{Re}_d > 1000 \end{cases} \tag{8}$$

where Re_d is the droplet Reynolds number and is defined as follows

$$\mathrm{Re}_d = \frac{2\rho_A\left|\vec{U}_R\right|r_d}{\mu_A} \tag{9}$$

in which r_d is the droplet radius and μ_g is the gas (air) viscosity.

2.2 Droplet Evaporation

To include the effect of evaporation rate on spray penetration, apply a control volume at droplet surface that will change with droplet radius during evaporation process. For simplicity, consider steady state analysis first.

2.2.1. Steady-State Analysis

A fuel droplet rarely reaches a steady state (ref. 3) during its lifetime. This is because most commercial fuels are multi-component, where different fuel compounds posses its own properties, for example kerosene and gasoline. To simplify analysis, 'steady state' term here refers to 'quasi-steady', which allows droplet lifetime and evaporation rate to be estimated to an acceptable level of accuracy.

To simplify the analysis, in addition to the assumptions listed in trajectory analysis as shown in section 2.1:

1. there is no radiation,
2. there is no internal circulation and internal convective heating within droplet,
3. consider only single-component fuel (with well-defined boiling point), and
4. it is quasi-steady flow.

Consider a fuel droplet at low fuel injection temperature that is suddenly exposed to a gaseous crossflow at high temperature. Initially, almost all heat supplied to the droplet serves to raise the droplet temperature. As the droplet temperature rises, fuel vapor will form at the droplet surface and has two main effects:

1. a large portion of heat transferred to droplet is used to vaporize the droplet, and
2. the outward flow of fuel vapor impedes the rate of heat transfer to droplet

Eventually, a stage is reached where all heat transferred to droplet is used as the heat of vaporization and the droplet temperature will stabilize at a steady-state temperature.

Mass Transfer Number

Assume that the thermal diffusion is negligible. Therefore, the concentration gradient is the only driving force considered for species diffusion in the direction of the diffusion path. Then, the following expression for an evaporating fuel droplet of radius r is described by:
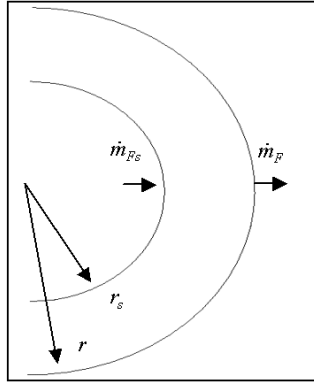
$$\frac{dY_F}{dr} = -\frac{R_u T}{D_{FA} P}\left(\dot{m}_F'' Y_A\right) \qquad (10)$$

where D_{FA} diffusion coefficient of fuel in air
$\dot{m}_F''$ mass rate of diffusion per unit area (mass flux)
P ambient air pressure
R_u universal gas constant
T ambient air temperature
$Y_F(r)$ fuel mass fraction
Y_A air mass fraction at range $r_s < r < \infty$ at any time

$$Y_A = 1 - Y_F \qquad (10a)$$

From the continuity equation applied on the control surface surrounding a droplet, one obtains

$$\dot{m}_F'' = \dot{m}_{F,s}''\left(\frac{r_s}{r}\right)^2 \qquad (11)$$

Figure 2.2 – Sketch of the control surface surrounding a droplet

where $\dot{m}''_{F,s}$ – mass flux at droplet surface

r_s – radius of droplet

r – radius of control surface at time t

Substituting equation (10a) and (11) into (10) yields:

$$\frac{dY_F}{dr} = -\frac{R_u T}{D_{FA} P} \dot{m}''_{F,s} \left(\frac{r_s}{r}\right)^2 (1 - Y_{F,s})$$ (12)

where $Y_{F,s}$ - fuel mass fraction at droplet surface

Assume the ideal gas relation ($\rho = \dfrac{P}{R_u T}$), separating variables, integrating, and rearranging equation (12) yields:

$$\dot{m}''_{F,s} = -\frac{\rho D_{FA}}{r_s} \cdot \ln(1 - Y_{F,s})$$ (13)

where ρ – fuel density

Multiplying by droplet surface area ($A_s = 4\pi r_s$) and for $D = 2r_s$,

$$\dot{m}_{F,s} = -2\pi D \cdot \rho D_{FA} \cdot \ln(1 - Y_{F,s})$$ (14)

where D – diameter of evaporating droplet

Unity Lewis Number (Le)

Assume $Le = 1$, it implies that the mass transfer rate is equal to the heat transfer rate, i.e.

$$D_{FA} = \alpha_g \qquad (15)$$

where

$$\alpha_g = \left(\frac{k}{\rho c_p} \right)_g \qquad (15a)$$

where α_g – thermal diffusivity of gas
$\quad k$ – thermal conductivity
$\quad c_p$ – specific heat at constant pressure

Then,

$$\rho_g D_{FA} = \left(\frac{k}{c_p} \right)_g \qquad (16)$$

Define the mass transfer number, B_M:

$$B_M = \frac{Y_{F,s} - Y_{F,\infty}}{Y_{F,drop} - Y_{F,s}} \qquad (17)$$

Since $Y_{F,\infty} \approx 0$ and $Y_{F,drop} = 1$, equation (17) is simplified as:

$$B_M = \frac{Y_{F,s}}{1 - Y_{F,s}} \qquad (18)$$

$$Y_{F,s} = \left[1 + \left(\frac{P}{P_{F,s}} - 1 \right) \left(\frac{M_A}{M_F} \right) \right]^{-1} \qquad (19)$$

where M_F – molecular weight of fuel [kg/kg-mol]
$\quad M_A$ – molecular weight of air [kg/kg-mol]
$\quad P$ – ambient pressure [kPa]
$\quad P_{F,s}$ – fuel vapor pressure at droplet surface [kPa]

Rearranging $ln(1–Y_F)$ in term of B_M yields:

$$\ln(1 - Y_F) = -\ln(1 + B_M) \qquad (20)$$

Substituting (16) and (20) into (14) yields the rate of evaporation of a fuel drop at the surface:

$$\dot{m}_{F,s} = 2\pi D \left(\frac{k}{c_p}\right)_g \ln(1 + B_M) \tag{21}$$

Reference Conditions

For better accuracy, the choice of values of k_g and $c_{p,g}$ are evaluated at the following reference temperature and composition using the "one-third rule":

$$T_r = T_s + \frac{1}{3}(T_\infty - T_s) \tag{22}$$

$$Y_{F,r} = Y_{F,s} + \frac{1}{3}(Y_{F,\infty} - Y_{F,s}) \tag{23}$$

Since $Y_{F,\infty} \approx 0$,

$$Y_{F,r} = \frac{2}{3}Y_{F,s} \tag{24}$$

$$Y_{A,r} = 1 - Y_{F,r} \tag{25}$$

Fuel-air Mixture

Therefore, the reference thermal conductivity and specific heat at constant pressure are estimated as:

$$k_g = Y_{A,r} \cdot k_A(T_r) + Y_{F,r} \cdot k_{Fv}(T_r) \tag{26}$$

$$c_{p,g} = Y_{A,r} \cdot c_{p,A}(T_r) + Y_{F,r} \cdot c_{p,Fv}(T_r) \tag{27}$$

All properties for air and different fuels are provided in *Appendix A* and *B*.

Evaporation Constant

At steady-state period, the droplet diameter D at any instant may be related to its initial diameter D_0 by D^2-law:

$$D_0^2 - D^2 = \lambda_{st} t \tag{28}$$

where λ_{st} $[\dfrac{m^2}{s}]$ is the steady state evaporation constant

$$\lambda_{st} = \left(\frac{k}{c_p}\right)_g \frac{8\ln(1 + B_M)}{\rho_F} \qquad (28a)$$

The D^2-law states that the square of droplet diameter is a linear function of time where the evaporation constant apparently represents the slope of the equation. The larger the λ_{st}, the shorter the time it takes for the droplet to vaporize completely.

Heat Transfer Number

Consider conductive and convective heat fluxes across a thin shell surrounding the evaporating droplet, the heat transfer number is defined as the ratio of enthalpy available in the surrounding gas to the energy required to vaporize the fuel:

$$B_T = \frac{c_{p,g}(T_\infty - T_s)}{L + c_{p,drop}(T_s - T_{drop})} \qquad (29)$$

where L – latent heat of fuel vaporization corresponding to fuel surface temperature $[\dfrac{kJ}{kg}]$

For simplicity, one can neglect the energy required to raise the droplet temperature to the surface temperature. Then, equation (29) becomes:

$$B_T = \frac{c_{p,g}(T_\infty - T_s)}{L} \qquad (30)$$

When heat transfer dominates the evaporation process, the rate of evaporation of a fuel droplet at the surface is then described by:

$$\dot{m}_{F,s} = 2\pi D \left(\frac{k}{c_p}\right)_g \ln(1 + B_T) \qquad (31)$$

B$_T$ versus B$_M$

Estimation of rate of fuel evaporation using equation (31) is only good for steady-state conditions. Nevertheless, equation (21) applies under all conditions, including the heat-up process of droplet (ref. 6).

However, under steady state conditions, $B_M = B_T = B$ and equation (21) and (31) are identical. Therefore, droplet evaporation rate can be written as:

$$\dot{m}_{Fs} = 2\pi D \left(\frac{k}{c_p} \right)_g \ln(1 + B) \tag{32}$$

2.2.2 Heat-up Process

According to Chin[6], serious error may be incurred in the calculation of fuel evaporation rate and droplet lifetime if the transient heat-up process is neglected. In fact, for many fuels at high ambient pressure and temperature, the transient heat-up process constitutes a significant portion of the droplet evaporation time.

At the steady-state period, the heat used in vaporizing the fuel is given by:

$$Q_{ss} = \dot{m}_F L \tag{33}$$

Substituting equation (14) into (33) yields,

$$Q_{ss} = 2\pi D \left(\frac{k}{c_p} \right)_g \ln(1 + B_M) L \tag{34}$$

Including the heating process, the actual heat transfer is estimated as:

$$Q_{act} = 2\pi D (T_\infty - T_s)_g \frac{\ln(1 + B_M)}{B_M} \tag{35}$$

Then, the rate of change of the droplet surface temperature is given by:

$$\frac{dT_s}{dt} = \frac{Q_{act} - Q_{ss}}{c_{p,F} m_{drop}} \tag{36}$$

Substituting equation (34) and (35) into (36) and rearranging gives,

$$\frac{dT_s}{dt} = \frac{\dot{m}_F L}{c_{p,F} m_{drop}} \left(\frac{B_T}{B_M} - 1 \right) \tag{37}$$

where

$$m_{drop} = \frac{\pi}{6} \rho_F D^3 \tag{37a}$$

Note that

$$\dot{m}_F = \frac{d}{dt}\left(\frac{\pi}{6}\rho_F D^3\right) \tag{38}$$

Equating equation (14) and (38) and rearranging gives the rate of change of droplet size:

$$\frac{dD}{dt} = \frac{4\ln(1+B_M)}{\rho_F D}\left(\frac{k}{c_p}\right)_g \tag{39}$$

2.2.3 Droplet Lifetime

Since the rate of chemical reactions in many practical combustion systems are so high, the burning rate is mainly controlled by the fuel evaporation process. Therefore, droplet lifetime is important in such situations because it determines the residence time needed to ensure completion of combustion.

Assume the final droplet diameter, D_0, equal to zero and rearranging the D^2-*law*, the steady state droplet lifetime is readily obtained by:

$$t_{e,st} = \frac{D_0^2}{\lambda_{st}} \tag{40}$$

2.2.4 Convective Effect

All derivations shown above are only good for droplet at stationary condition. In addition, it is known that convection may enhance both mass and heat transfer during the evaporation process. Moreover, to include the convective effect into the evaporation rate equation is straightforward. Equation (23) is then modified by replacing the coefficient with the Nusselt number correlation (ref. 5):

$$\dot{m}_F = Nu\pi D\left(\frac{k}{c_p}\right)_g \ln(1+B_M) \tag{41}$$

where

$$Nu = 2 + 0.6\,\mathrm{Re}_d^{0.5}\,\mathrm{Pr}_A^{0.33} \tag{41a}$$

$$\mathrm{Re}_d = \frac{\rho_A \cdot U_r \cdot D}{\mu_A} \tag{41b}$$

All physical properties should be evaluated at the reference temperature, T_r, except for μ_A, Pr_A, and ρ_A. Please refer to *Appendix B* for air properties.

3. Numerical Method

Six ordinary differential equations are to be solved for the six dependent variables x, z, u_d, w_d, D, *and* T_s. The droplet trajectory is defined by the set of *x* and *z* values. A 4th-Order Runge-Kutta explicit method[8] was used to solve these equations. The Runge-Kutta explicit method is an ideal numerical scheme for solving ordinary differential equations using Microsoft[®] Excel software. It is a self-starting method with good stability characteristics. The time step-size can be changed as desired without any complications for higher-order schemes.

There are totally six sets of coupled equations, namely the time rate change of *x, z, u_d, w_d, D,* and T_s, along with their solutions, as shown below: (subscript *n* stands for the nth time step)

[1]
$$\frac{dx}{dt} = u_d = f_1(u_d) \tag{1}$$

$$x_{n+1} = x_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \tag{1a}$$

where

$$k_1 = \Delta t \cdot f_1(u_{d,n})$$
$$k_2 = \Delta t \cdot f_1(u_{d,n} + \frac{l_1}{2})$$
$$k_3 = \Delta t \cdot f_1(u_{d,n} + \frac{l_2}{2}) \tag{1b}$$
$$k_4 = \Delta t \cdot f_1(u_{d,n} + l_3)$$

[2]
$$\frac{dz}{dt} = w_d = f_2(w_d) \tag{2}$$

$$z_{n+1} = z_n + \frac{1}{6}(kz_1 + 2kz_2 + 2kz_3 + kz_4) \tag{2a}$$

where

$$kz_1 = \Delta t \cdot f_2(w_{d,n})$$

$$kz_2 = \Delta t \cdot f_2(w_{d,n} + \frac{lz_1}{2})$$

$$kz_3 = \Delta t \cdot f_2(w_{d,n} + \frac{lz_2}{2})$$

$$\quad (2b)$$

$$kz_4 = \Delta t \cdot f_2(w_{d,n} + lz_3)$$

[3]
$$\frac{du_d}{dt} = \frac{\left[-\frac{1}{2} \rho_g (u_d - u_g) |\vec{U}_g| A_d C_D \right]}{\rho_d V_d} = f_3(u_d, w_d, D, T_s) \qquad (3)$$

$$u_{d,n+1} = u_{d,n} + \frac{1}{6}(l_1 + 2l_2 + 2l_3 + l_4) \qquad (3a)$$

where

$$l_1 = \Delta t \cdot f_3(u_{d,n}, w_{d,n}, D_n, T_{s,n})$$

$$l_2 = \Delta t \cdot f_3(u_{d,n} + \frac{l_1}{2}, w_{d,n} + \frac{lz_1}{2}, D_n + \frac{md_1}{2}, T_{s,n} + \frac{mt_1}{2})$$

$$l_3 = \Delta t \cdot f_s(u_{d,n} + \frac{l_2}{2}, w_{d,n} + \frac{lz_2}{2}, D_n + \frac{md_2}{2}, T_{s,n} + \frac{mt_2}{2})$$

$$\quad (3b)$$

$$l_4 = \Delta t \cdot f_3(u_{d,n} + l_3, w_{d,n} + lz_3, D_n + md_3, T_{s,n} + mt_3)$$

[4]
$$\frac{dw_d}{dt} = \frac{\left[-\frac{1}{2} \rho_g (w_d - w_g) |\vec{U}_R| A_d C_D + (\rho_g - \rho_d) V_d g \right]}{\rho_d V_d} = f_4(u_d, w_d, D, T_s) \qquad (4)$$

$$w_{d,n+1} = w_{d,n} + \frac{1}{6}(lz_1 + 2lz_2 + 2lz_3 + lz_4) \qquad (4a)$$

where

$$lz_1 = \Delta t \cdot f_4(u_{d,n}, w_{d,n}, D_n, T_{s,n})$$

$$lz_2 = \Delta t \cdot f_4(u_{d,n} + \frac{l_1}{2}, w_{d,n} + \frac{lz_1}{2}, D_n + \frac{md_1}{2}, T_{s,n} + \frac{mt_1}{2})$$

$$lz_3 = \Delta t \cdot f_4(u_{d,n} + \frac{l_2}{2}, w_{d,n} + \frac{lz_2}{2}, D_n + \frac{md_2}{2}, T_{s,n} + \frac{mt_2}{2})$$

$$\quad (4b)$$

$$lz_4 = \Delta t \cdot f_4(u_{d,n} + l_3, w_{d,n} + lz_3, D_n + md_3, T_{s,n} + mt_3)$$

[5]
$$\frac{dD}{dt} = -\frac{\lambda}{2D} = f_5(D, T_s) \tag{5}$$

$$D_{n+1} = D_n + \frac{1}{6}(md_1 + 2md_2 + 2md_3 + md_4) \tag{5a}$$

where

$$md_1 = \Delta t \cdot f_5(D_n, T_{s,n})$$

$$md_2 = \Delta t \cdot f_5(D_n + \frac{md_1}{2}, T_{s,n} + \frac{mt_1}{2})$$

$$md_3 = \Delta t \cdot f_5(D_n + \frac{md_2}{2}, T_{s,n} + \frac{mt_2}{2}) \tag{5b}$$

$$md_4 = \Delta t \cdot f_5(D_n + md_3, T_{s,n} + mt_3)$$

[6]
$$\frac{dT_s}{dt} = \frac{\dot{m}_F L}{c_{p,F} m_{drop}}\left(\frac{B_T}{B_M} - 1\right) = f_6(u_d, w_d, D, T_s) \tag{6}$$

$$T_{s,n+1} = T_{s,n} + \frac{1}{6}(mt_1 + 2mt_2 + 2mt_3 + mt_4) \tag{6a}$$

where

$$mt_1 = \Delta t \cdot f_6(u_{d,n}, w_{d,n}, D_n, T_{s,n})$$

$$mt_2 = \Delta t \cdot f_6(u_{d,n} + \frac{l_1}{2}, w_{d,n} + \frac{lz_1}{2}, D_n + \frac{md_1}{2}, T_{s,n} + \frac{mt_1}{2})$$

$$mt_3 = \Delta t \cdot f_6(u_{d,n} + \frac{l_2}{2}, w_{d,n} + \frac{lz_2}{2}, D_n + \frac{md_2}{2}, T_{s,n} + \frac{mt_2}{2}) \tag{6b}$$

$$mt_4 = \Delta t \cdot f_6(u_{d,n} + l_3, w_{d,n} + lz_3, D_n + md_3, T_{s,n} + mt_3)$$

Only every n[th] cycle (as specified by the user) is saved for plotting. This greatly saves on storage and increases the speed of post processing. We have chosen to enter the data in SI units in the unlocked cells. The required conversions are done in the locked cells. When the user becomes familiar with the spreadsheet, the spreadsheet can be unlocked as there is no password and the user can adapt the spreadsheet as required.

The report can be accessed on the web at:
http://gltrs.grc.nasa.gov/reports/2004/TM-2004-212910/TM-2004-212910.html
along with the interactive Microsoft® Excel spreadsheet[1] that computes and plots
data with and without spray evaporation can be accessed through hyperlinks located on the back
of the title page and on page 17, or directly on the web at:
http://gltrs.grc.nasa.gov/reports/2004/TM-2004-212910/SprayEvapVBONOFFSept23.xls

The interactive Microsoft® Excel spreadsheet is also available on CD-ROM as a separate
document. Additional copies of the Microsoft® Excel spreadsheet can be requested by e-mailing:
Dan.L.Bulzan@nasa.gov.

The CD-ROM also contains NASA/TM—2002-211710 and supplemental Microsoft® Excel
interactive spreadsheet that computes and plots data without spray evaporation.

---

[1]To access the interactive spreadsheet, Microsoft® Excel 2002 or higher is recommended to view the file and for
proper functionality. It can be opened through your browser, however, saving to the hard drive is recommended. If
you cannot access this file, please contact: Dan.L.Bulzan@nasa.gov.

4. Equations Summary for Tracking an Evaporating Spray in a Crossflow

Inputs

Droplets

      Properties: R_{32}, A_d, V_d, $\rho_{d,288.25K}$, T_{crit}, T_{bn}, C_{exp}, L_{Tbn}, MW_d, a, b

      Initial conditions: X_0, U_{d0}, Z_0, W_{d0}, T_{init},

Crossflow

      Properties: M_g, U_g, W_g

Ambient conditions: T_∞, P, g

Numerical method: Δt, n (total cycle number), n_{data}, ε

Outputs

Ballistics: X, U_d, Z, W_{d0}, C_{dt}

Evaporation: D, T_{surf}

Equations

Mass Transfer Number:

If (Group 1 fuel[2]) then

$$P_{F,s} = P_{vap}(T_s) = \exp\left(a - \frac{b}{T_s - 43}\right) [\text{kPa}]$$

(1)

else

$$\log_{10} P_{vap} = A + \frac{B}{T_s} + C\log_{10} T_s + DT_s + ET_s^2$$

(2)

$$P_{F,s} = P_{vap}(T_s)[\text{mmHg}]$$

(3)

End if

$$Y_{F,s} = \left[1 + \left(\frac{P}{P_{F,s}} - 1\right)\left(\frac{M_A}{M_F}\right)\right]^{-1}$$

(4)

$$B_M = \frac{Y_{F,s}}{1 - Y_{F,s}}$$

(5)

Reference Conditions:

$$T_r = T_s + \frac{1}{3}(T_\infty - T_s)[\text{K}]$$

(6)

$$Y_{F,r} = \frac{2}{3}Y_{F,s}$$

(7)

$$Y_{A,r} = 1 - Y_{F,r}$$

(8)

Thermal Properties of Air

$$k_A = 2\cdot10^{-11}T_r^3 - 5\cdot10^{-8}T_r^2 + 0.0969\cdot10^{-3}T_r + 0.8289\cdot10^{-3}[\frac{W}{m\cdot K}]$$

(9)

---

[2]Group 1 fuels – DF 2, JP 4, JP 5 and n-Heptane
  Group 2 fuels – Jet-A ($C_{12}H_{23}$) and Water (H_2O) (please refer to *Appendix A* for more details.)

$$c_{p,A} = -5 \cdot 10^{-11} T_r^3 + 2 \cdot 10^{-7} T_r^2 - 1 \cdot 10^{-5} T_r + 1.0041 [\frac{kJ}{kg}] \tag{10}$$

$$\mu_A = (7 \cdot 10^{-8} T_\infty^3 - 0.0003 T_\infty^2 + 0.6227 T_\infty + 18.761) \cdot 10^{-7} [\frac{Ns}{m^2}] \tag{11}$$

$$\mathrm{Pr}_A = 2 \cdot 10^{-13} T_\infty^4 - 10^{-9} T_\infty^3 + 2 \cdot 10^{-6} T_\infty^2 - 0.0009 T_\infty + 0.8632 \tag{12}$$

$$\rho_A = 355.91 T_\infty^{-1.0032} [\frac{kg}{m^3}] \tag{13}$$

Thermal properties of hydrocarbon fuel:

If (Group 1 fuel) then

$$\rho_F = \rho_{F,288.6K} \left[1 - 1.8 C_{ex}(T - 288.6) - 0.09 \frac{(T - 288.6)^2}{(T_{crit} - 288.6)^2} \right] [\frac{kg}{m^3}] \tag{14}$$

$$c_{p,Fv} = (0.363 + 0.000467 T_r)(5 - 0.001 \rho_{F,288.6K}) [\frac{kJ}{kgK}] \tag{15}$$

$$c_{p,F} = (0.76 + 0.00335 T_{drop})(0.001 \rho_F)^{0.5} [\frac{kJ}{kgK}] \tag{16}$$

$$n = 2 - 0.0372 \left(\frac{T_r}{T_{bn}} \right)^2 \tag{17}$$

$$k_{Fv} = 10^{-6} [13.2 - 0.0313(T_{bn} - 273)] \left(\frac{T_r}{273} \right)^n [\frac{W}{m \cdot K}] \tag{18}$$

$$L = L_{Tbn} \left(\frac{T_{crit} - T_s}{T_{crit} - T_{bn}} \right)^{0.38} [\frac{kJ}{kg}] \tag{19}$$

Else

$$\rho_F = AB^{-\left(1 - \frac{T_{drop}}{T_c}\right)^n} [\frac{g}{cm^3}] \tag{20}$$

$$k_{Fv} = A + BT_r + CT_r^2 \ [\frac{W}{m \cdot K}] \tag{21}$$

$$C_{p,Fv} = A + BT_r + CT_r^2 + DT_r^3 + ET_r^4 \ [\frac{J}{mol - K}] \tag{22}$$

$$C_{p,F} = A + BT_{drop} + CT_{drop}^2 + DT_{drop}^3 \ [\frac{J}{mol - K}] \tag{23}$$

$$L = A\left(1 - \frac{T_{drop}}{T_c}\right)^n \ [\frac{kJ}{mol}] \tag{24}$$

End if

Thermal properties of fuel-air mixture:

$$k_g = Y_{A,r} \cdot k_A(T_r) + Y_{F,r} \cdot k_v(T_r) \tag{25}$$

$$c_{p,g} = Y_{A,r} \cdot c_{p,A}(T_r) + Y_{F,r} \cdot c_{p,v}(T_r) \tag{26}$$

Droplet trajectories:

$$\vec{U}_R = \sqrt{\left(u_d - u_g\right)^2 + \left(w_d - w_g\right)^2} \tag{27}$$

$$\mathrm{Re}_d = \frac{2\rho_A|\vec{U}_R|r_{d,s}}{\mu_A} \tag{28}$$

$$C_D = \begin{cases} \dfrac{24}{\mathrm{Re}_d}\left[1 + \dfrac{1}{6}\mathrm{Re}_d^{2/3}\right] & \mathrm{Re}_d \leq 1000 \\ \\ 0.424 & \mathrm{Re}_d > 1000 \end{cases} \tag{29}$$

$$\rho_d V_d \frac{du_d}{dt} = -\frac{1}{2}\rho_g\left(u_d - u_g\right)|\vec{U}_R|A_d C_D \tag{30}$$

$$\rho_d V_d \frac{dw_d}{dt} = -\frac{1}{2}\rho_g\left(w_d - w_g\right)|\vec{U}_R|A_d C_D + \left(\rho_g - \rho_d\right)V_d \ g \tag{31}$$

$$\frac{dx}{dt} = u_d \tag{32}$$

$$\frac{dz}{dt} = w_d \tag{33}$$

Convective effect:

$$Nu = 2 + 0.6\,\mathrm{Re}_D^{0.5}\,\mathrm{Pr}_{air}^{0.33} \tag{34}$$

$$\dot{m}_F = Nu\,\pi D\left(\frac{k}{c_p}\right)_g \ln(1 + B_M) \tag{35}$$

Heat transfer number:

$$B_T = \frac{c_{p,g}(T_\infty - T_s)}{L} \tag{36}$$

Evaporation constant:

$$\lambda = \left(\frac{k}{c_p}\right)_g \frac{8\,In(1 + B_M)}{\rho_F}\;[\frac{m^2}{s}] \tag{37}$$

$$m_{drop} = \frac{\pi}{6}\rho_F D^3\;[\mathrm{kg}] \tag{38}$$

Heat-up process:

$$\frac{dT_s}{dt} = \frac{\dot{m}_F L}{c_{p,F}\,m_{drop}}\left(\frac{B_T}{B_M} - 1\right)[\frac{K}{s}] \tag{39}$$

$$\frac{dD}{dt} = -\frac{\lambda}{2D}\;[\frac{m}{s}] \tag{40}$$

Evaporation time:

$$t_{e,st} = \frac{D_0^2}{\lambda}\;[\mathrm{s}] \tag{41}$$

5. User's Manual

This program is written in Microsoft® Visual Basic Excel. There are six sheets in the program, namely the 'CoverPage' sheet, 'Instruction' sheet, 'Process' sheet, 'Data' sheet, 'Trajectory' sheet, and 'Evap'Code sheet.

'CoverPage' sheet
Relevant information about authors is provided in this sheet. If there are any questions or comments regarding this program, please feel free to contact us.

'Instruction' sheet
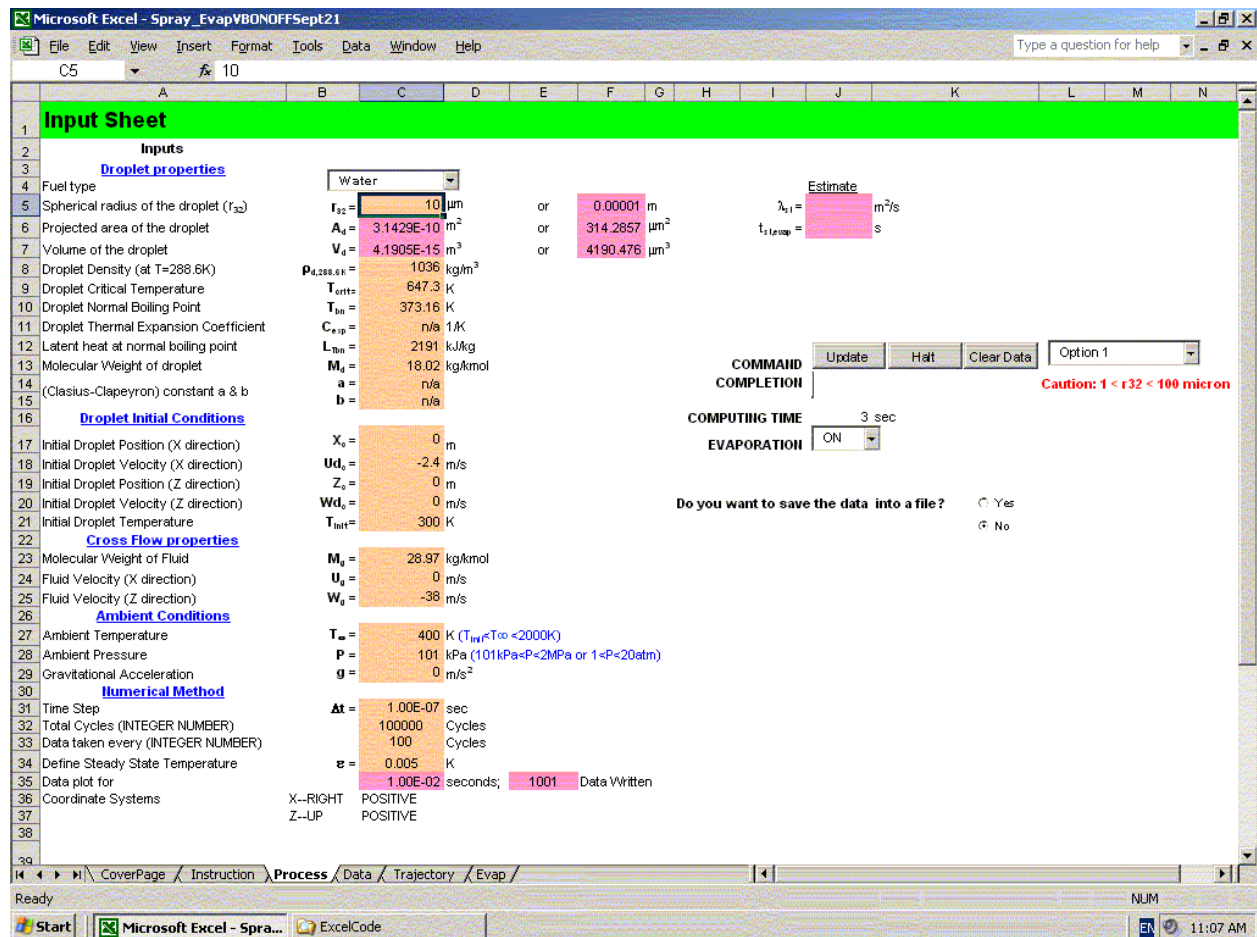The instruction sheet contains a description of each sheet in the program, as well as the user's manual.

Figure 5.1 – 'Process' sheet

'Process' sheet
The process sheet contains all user-inputs necessary for performing computations. First of all, the user needs to choose a fuel type from the pull down menu. This program contains built-in a database of properties of each listed fuel necessary for evaporation computations (see *Appendix A* – Fuel Properties). Other than that, the user can specify any hydrocarbon fuel not listed here, provided that the properties of each specified fuel are available.

The cells highlighted in *light-brown* are the user inputs. The cells highlighted in *pink* contain computed values associated with the light-brown cells; therefore, they are locked to prevent the user from modifying their values. When values are entered into the formula cells, the formulas are erased and linkages to other cells are interrupted. That is why for this version we chose to lock the closed cells without a password.

One feature added to the previous work (ref. 1) is the 'Evaporation ON/OFF' switch. If the user chooses to turn off the 'Evaporation switch', the Microsoft$^®$ Excel program will run as if there is no evaporation and will give the solution for the droplet ballistics only.

If the 'Evaporation switch' is turned on, one main concern is the computational time because it is considered very computation-intensive to use Runge-Kutta method in a Microsoft$^®$ Excel spreadsheet for tracking a single evaporating droplet in crossflow. The computational time varies mostly with the CPU power available and the total cycle number as well as the time step size. To remedy this problem, four options have been proposed and it is up to the user to select one of the four options before running the code:

a. *Option 1* – Properties are calculated four times each cycle (more accurate but very slow)
b. *Option 2* – Properties are calculated one time each cycle (less accurate but fast)
c. *Option 3* – Properties are calculated four times during droplet heating. After droplet reaches steady state temperature, properties are not calculated any more. The values are used from the last cycle of droplet heating. This option requires user to input a constant value of epsilon, ε. It defines the condition when the steady state temperature will occur. ($dT/dt = \varepsilon$)
d. *Evap time estimate* – Before computation, it's always good to estimate the steady state evaporation time (droplet lifetime) of the droplet of size r_{32}. Click the **Update** button to start the estimation. The computational speed depends mostly on the droplet size r_{32} and ambient conditions. For example, a Jet-A droplet of 50-micron diameter (r_{32}=25 micron) is estimated to completely vaporize within 0.004s. Using Δt=1.0e-7s, at least n=40,000 cycles are required.

4th-Order Runge-Kutta method calculates temperature derivative *(dT/dt)* four times in each cycle. *Option 3* will force the program to check all four temperature derivatives in each cycle. If one of the temperature derivatives in a cycle is less than the constant epsilon, properties calculation will not be performed afterward. Based on the experiment, epsilon between 0.1 and 1 is good enough. For instance, a case has been performed using two constant epsilons, say 0.0001 and 0.1. Solutions from both epsilons yield similar results. Note that this may not apply to cases with fuel with properties highly sensitive to temperature.

From figure 4, the last two user inputs, **Total Cycles (C32)** and **Data taken every ## cycles (C33)** are the important features that were added for monitoring the amount of output data. **The number assigned in pink cell "E34"must be kept below 65,536; the cell will turn red if this condition is not satisfied.** Keeping the value below the limit can be done by changing the value in the cell **"C33"**.

After all the inputs have been specified, clicking the **Update** button will instruct the program to update the data in the 'Data' sheet as well as all plots in the 'Trajectory' and 'Evap' sheets.

In summary, the user needs to do the following steps to run the program:

1. Go to the 'Process' Sheet ➜ click on the **Process** tab.
2. Enter input values in the light-brown cells.
3. Adjust the value in the cell C33 so that the computed value in cell E34 is less than 65,536.
4. Choose the fuel type from the '*Fuel*' option menu.
5. Choose the **ON/OFF** evaporation switch from the evaporation option menu.
6. If Evaporation is **ON**, choose one of the four options: *Option 1, 2, 3 or Evap time estimate*.
7. If '*Evap time estimate*' is selected in step 6, click the **Update** button. The code will estimate evaporation constant and time.
8. Based on the estimated value of $t_{st,evap}$ (droplet lifetime), adjust and input appropriate 'Time step Δt' and 'Total cycles n'.
9. Click the **Update** button.
10. Observe the droplet profiles on the solution plots at both 'Trajectory' and 'Evap' sheets.
11. Repeat step 1 through step 10 for different input values
12. Click the **Clear Data** button to clear the data (Optional).

An additional feature in this program is the option to store the computed data into a *TecPlot* format file. This feature provides the user a flexibility to plot the data using other software such as **TecPlot**. In summary, user needs to do the following steps to save the data into TecPlot format file:
1. Select "Yes" to save the data into a file
2. Specify the path and the filename to save into a file (e.g. C:\Document\result.dat)
3. Select the unit length (i.e. meter or millimeter)
4. Click the Update button

Other than computational time, another concern is the computer memory usage because the code needs a lot of memory space for those six variables arrays to be solved. As discussed above, the feature of estimating the steady state droplet lifetime enables us to predict the total cycle number n required to complete a whole evaporation process for each fuel droplet. By making use of this advantage, a so-called '*dynamic array size*' is added to the existing code such that the allocated memory space for each computation will closely follow the total cycle number n entered by user. Using the example in '*Evap Time Estimate*' option where $n=40,000$, each variable array size will be set to $n+2=40002$.

'Data' sheet

This sheet contains the solution data computed by the program. All the solution data will be listed separately at eight columns A to H for plotting. Table 1 shows the corresponding variables of each column:

Column	Variable	Units	Definition
A	Time	sec	Time
B	X	m	Droplet trajectory in x-direction
C	Z	m	Droplet trajectory in z-direction
D	U_d	$\dfrac{m}{s^2}$	Droplet velocity in x-direction
E	W_d	$\dfrac{m}{s^2}$	Droplet velocity in z-direction
F	C_D	n/a	Drag coefficient
G	D/D_0	μm	Normalized droplet diameter square
H	T_s	K	Droplet surface temperature

Table 5.1 – Variables with their corresponding column in 'Data' sheet

'Trajectory' sheet

As shown in figure 5, there are five graphs on this sheet, namely droplet trajectory, droplet velocity profile, drag coefficient, C_D, as a function of time, droplet velocity profiles as a function of time, and droplet trajectory profiles as a function of time.
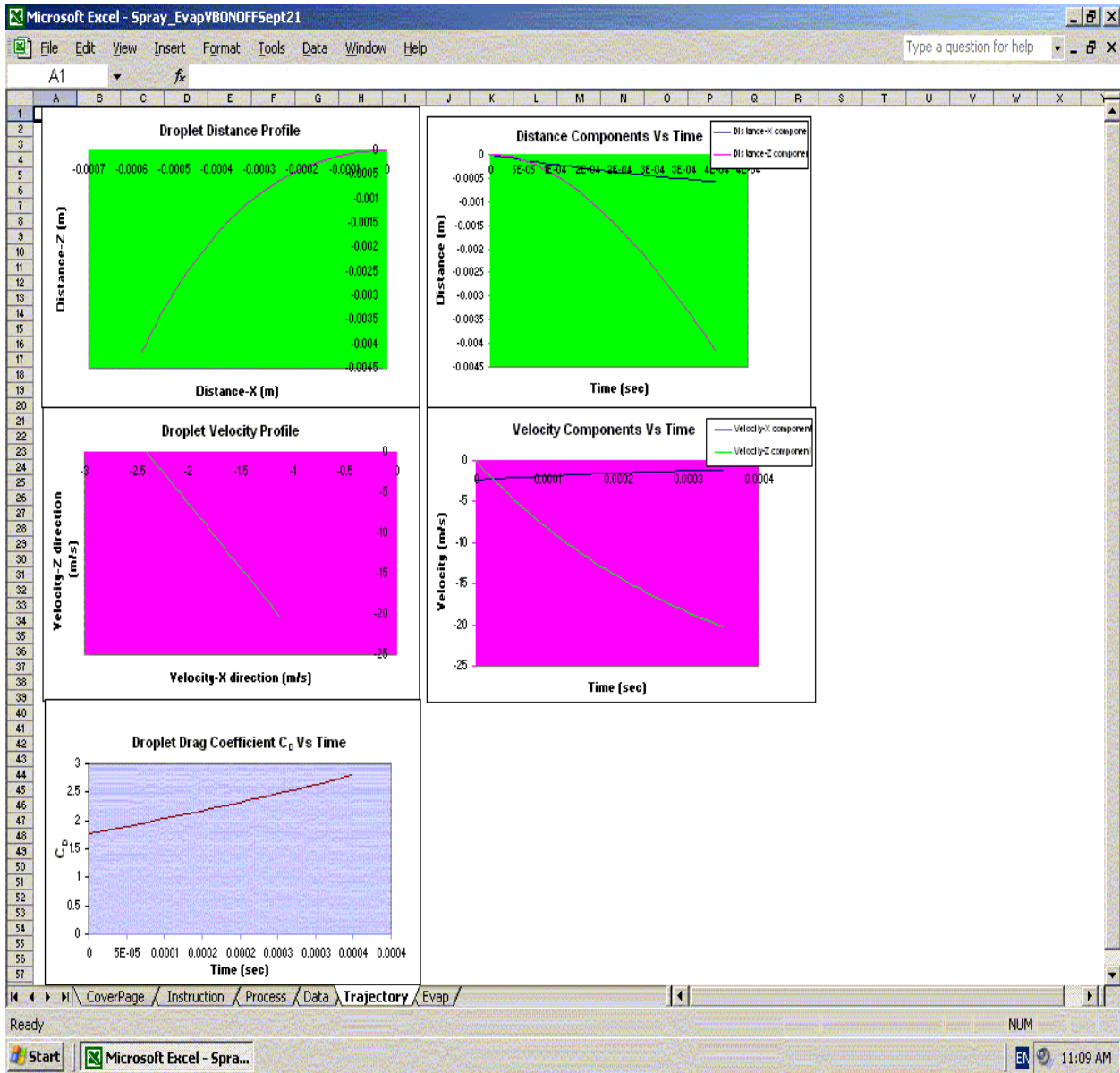
Figure 5.2 – 'Trajectory' sheet

'Evap' sheet

There are two graphs on this sheet, as shown in figure 6, namely droplet size and surface temperature histories. It also shows the values of the estimated steady-state evaporation constant and evaporation time.
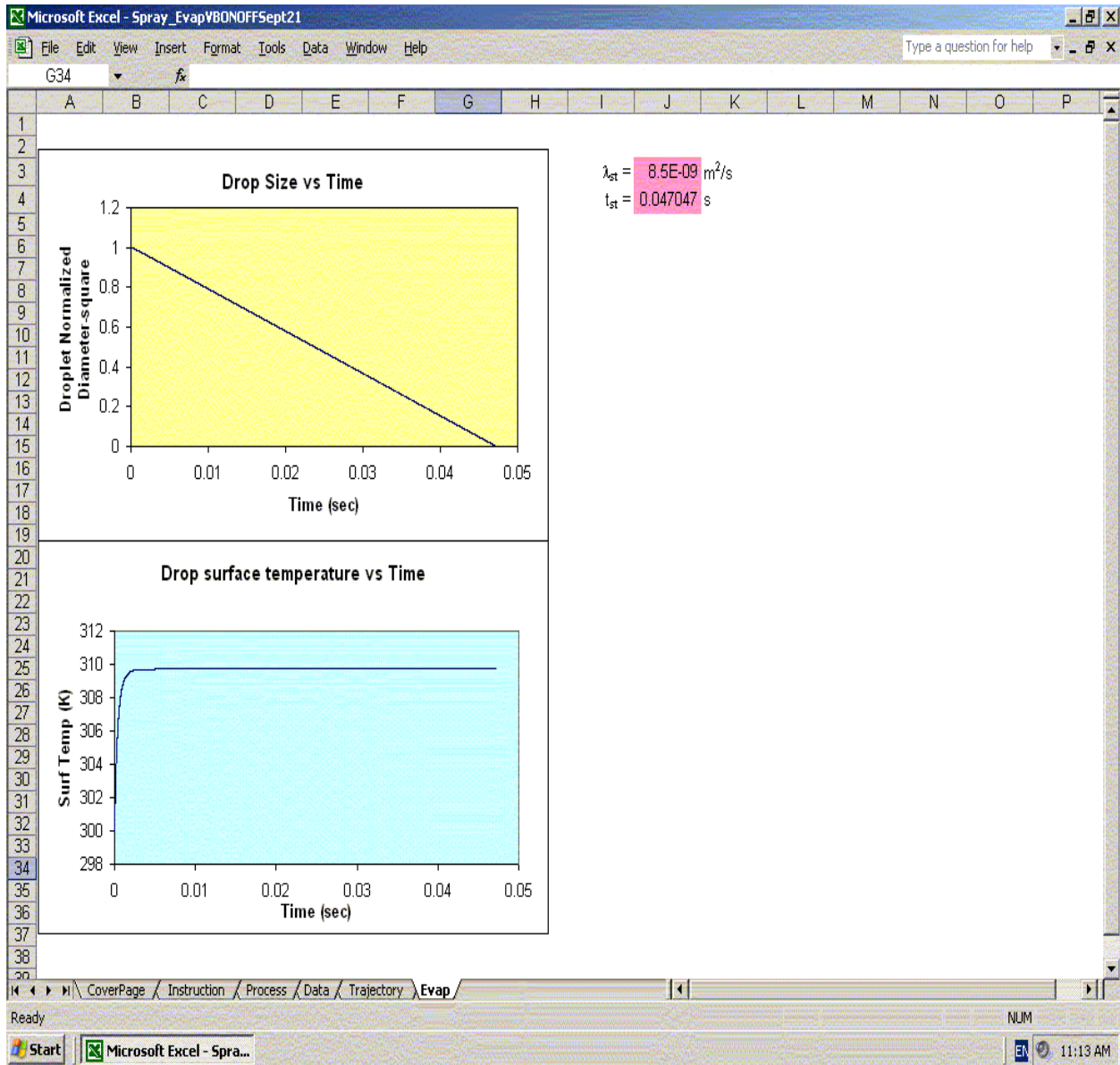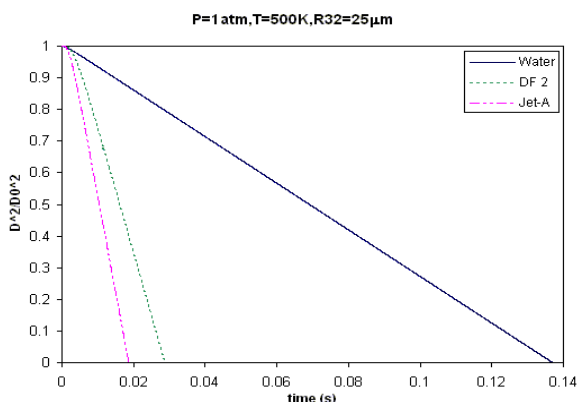

Figure 5.3 – 'Evap' sheet

6. Results and Discussions

To understand the performance of this Excel code, several results have been obtained and displayed in the following pages using the input parameters as shown in Table 6.1. Only the trajectory and the time variation of normalized droplet size and surface temperature are selected as the results to be discussed.

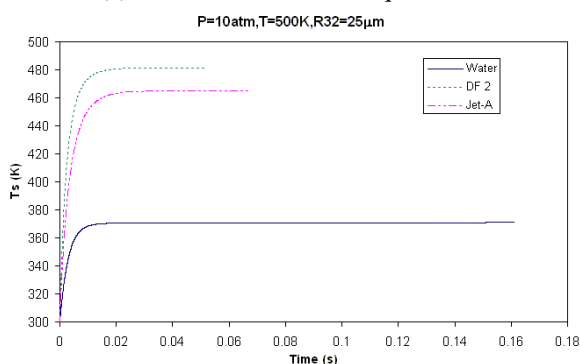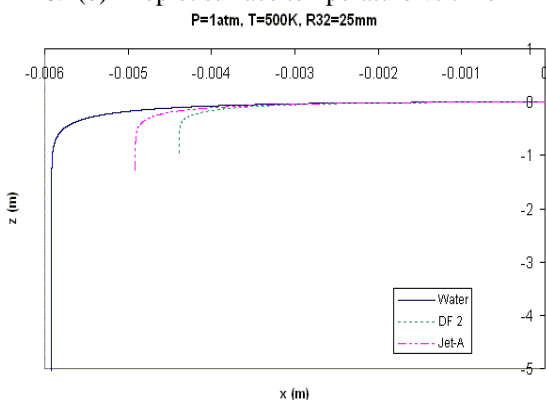6.1(a) Normalized diameter square vs time
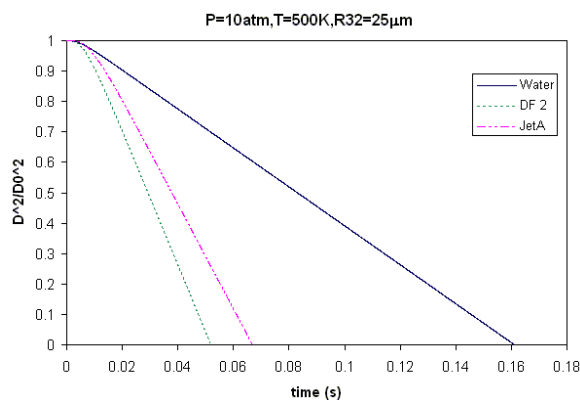
6.2(a) Normalized diameter square vs time
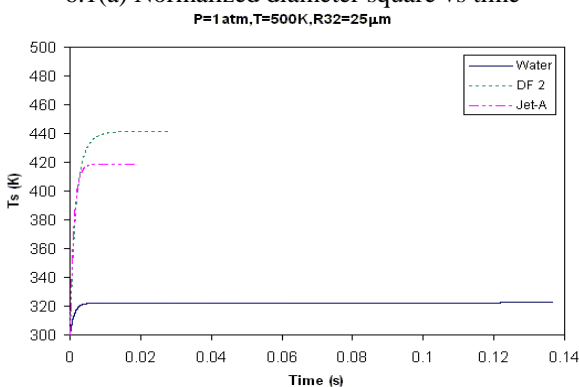
6.1(b) Droplet surface temperature vs time

6.2(b) Droplet surface temperature vs time

6.1(c) Trajectory of fuel droplet after injection
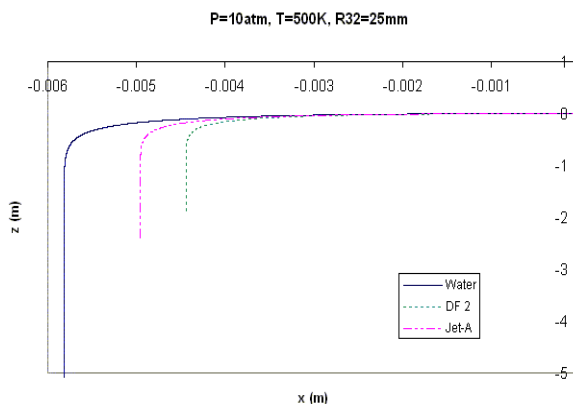
6.1(c) Trajectory of fuel droplet after injection

Figure 6.1 – Injection of fuel droplet at T=500K and P=1atm

Figure 6.2 – Injection of fuel droplet at T=500K and P=10atm

P=1atm,T=1000K,R32=25μm

6.3(a) Normalized diameter square vs time

P=10atm,T=1000K,R32=25μm
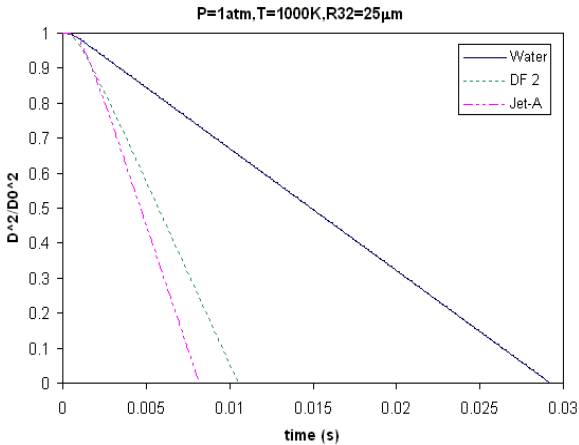
6.4(a) Normalized diameter square vs time

P=1atm,T=1000K,R32=25μm

6.3(b) Droplet surface temperature vs time

P=10atm,T=1000K,R32=25μm
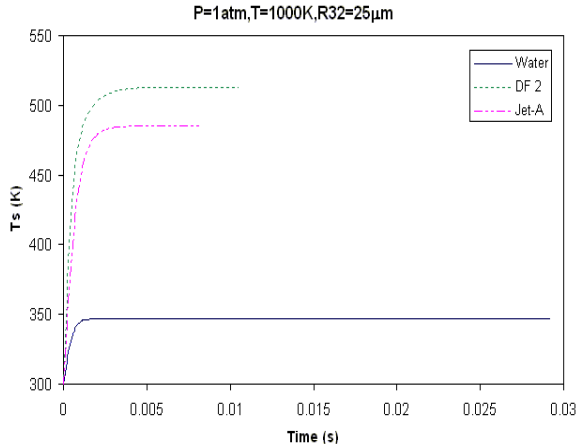
6.4(b) Droplet surface temperature vs time

P=1atm, T=1000K, R32=25mm

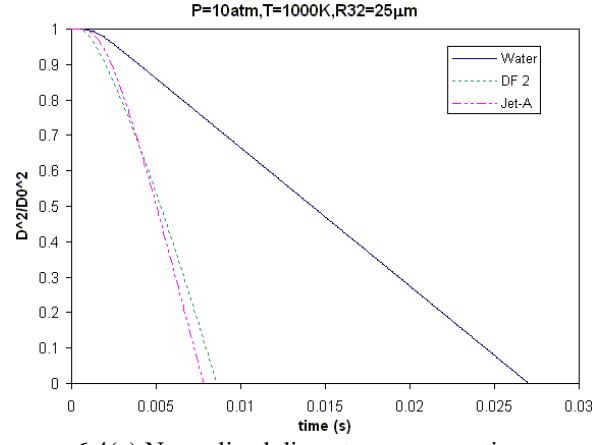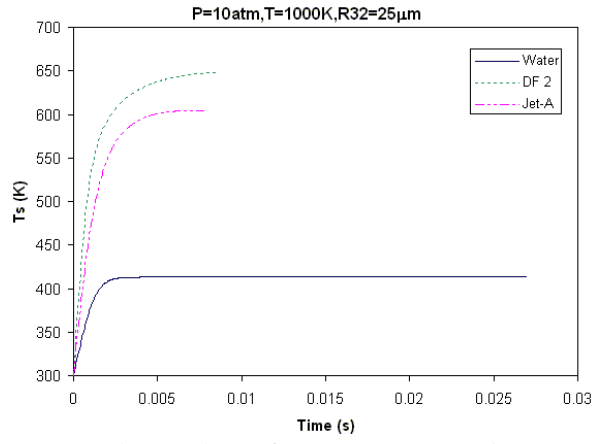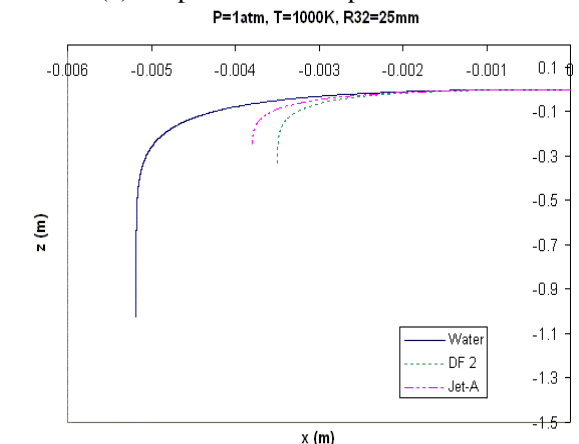6.3(c) Trajectory of fuel droplet after injection

P=10atm, T=1000K, R32=25mm
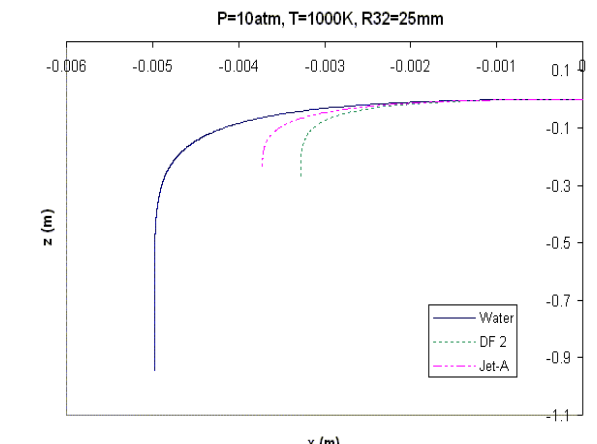
6.4(c) Trajectory of fuel droplet after injection

Figure 6.3 – Injection of fuel droplet at
T=500K and P=1atm

Figure 6.4 – Injection of fuel droplet at
T=500K and P=10atm

In figure 6.1(a), water droplet of diameter 50 micron at 500K 1 atm takes a much longer time to reach complete vaporization i.e. around 0.13s than Jet-A and DF 2 do. As shown in figure 6.2(a), increasing the ambient pressure to 10 atm further extends its droplet lifetime to around 0.16s, as known from the general physics results. Similar conclusions will be obtained when comparing figure 6.3(a) with 6.4(a) except that each fuel droplet lifetime is much shorter at such a high ambient temperature (T=1000K).

Figures 6.1(b) and 6.2(b) show the time variation of droplet surface temperature at the same ambient temperature i.e. 500K and two different ambient pressures, i.e. 1 atm and 10 atm respectively. In figure 6.1(b), each droplet reaches its steady state temperature and it never goes beyond its normal boiling point. While at high pressure as shown in figure 6.2(b), the boiling point increases with pressure and consequently the steady state temperature is higher. These observations also apply to figures 6.3(b) and 6.4(b).

The droplet trajectory after injection and before complete vaporization for each droplet is shown in figures 6.1(c), 6.2(c), 6.3(c) and 6.4(c). Due to the decreasing droplet mass (as a result of vaporization) followed by the momentum loss, the downward crossflow forces the evaporating droplet to drop almost vertically (or in z-direction) at the very end of its drop life. Water, with a higher heat capacity, therefore penetrates deepest across the crossflow among all the fuels.

Input	Value (unit)	Input	Value (unit)
r_{32}	25 μm	W_g	-38 m/s
X_o	0 m	T_∞	1000 K
Ud_o	-2.4 m/s	P	1010 kPa
Z_o	0 m	g	0 m/s^2
Wd_o	0 m/s	Δt	1.00E-07 sec
T_{init}	300 K	n	100000 Cycles
M_g	28.97 kg/kmol	ϵ	1 K
U_g	0 m/s		

Table 6.1 – Input parameters used in discussion

Appendix A—Fuel Properties

In this code, two different groups of fuels are considered. The reason is because the thermal properties for these two groups of fuels are referring to different sources as shown below.

Group	Source	Fuels
1	Chin (refs. 3 and 6)	DF 2 (diesel oil) JP 4 JP 5 n-Heptane
2	Yaws (ref. 7)	Jet-A ($C_{12}H_{23}$) Water (H_2O)

Table A.1 – two groups of fuels

Fuel/properties	DF 2	Jet-A ($C_{12}H_{23}$)	JP 4	JP 5	n-Heptane	Water (H_2O)
$\rho_{d,288.6K}$ (kg/m³)	846	915.91	773	827	687.8	1036
T_{crit} (K)	725.9	737.0	612	648.8	540.17	647.3
T_{bn} (K)	536.4	529.0	420	495.3	371.4	373.16
C_{exp} (1/K)	0.00046	n/a	0.000557	0.000485	0.000715	n/a
L_{Tbn} (kJ/kg)	254	341.200	292	266.5	317.8	2191
M_d (kg/kmol)	198	181.321	125	169	100.16	18.02
a*	15.5274	n/a	15.2323	15.16	14.2146	n/a
b*	5383.59	n/a	3999.66	4768.77	3151.68	n/a

*Clasius-Clapeyron constants
.

Table A.2 – Physical properties of fuels (Group 1 and 2)

All fuel properties are calculated in subroutines `transfernumber(Ts)` and `fuel_thermal_prop(Ts)` where Ts refers to surface temperature of a fuel droplet.

Group 1 Fuels: DF 2 (diesel oil), JP 4, JP 5, and n-Heptane

At any given droplet surface temperature, T_s, fuel vapor pressure at droplet surface, $P_{F,s}$, is approximated by *Clasius-Clapeyron* equation (or called *Antoine equation* (ref. 4)) [kPa]:

$$P_{F,s} = P_{vap}(T_s) = \exp\left(a - \frac{b}{T_s - 43}\right) \tag{A.1}$$

where a and b are constants for certain fuels (provided in Table A.2 shown above).

T_s – droplet surface temperature [K]

Meanwhile, variation of k_{Fv} [$\frac{W}{m \cdot K}$] and $c_{p,Fv}$ [$\frac{kJ}{kg}$] of fuel vapors with temperature are given by Chin (ref. 3):

$$k_{Fv} = 10^{-6}\left[13.2 - 0.0313(T_{bn} - 273)\right]\left(\frac{T_r}{273}\right)^n \tag{A.2}$$

$$c_{p,Fv} = (0.363 + 0.000467T_r)(5 - 0.001\rho_{F,288.6K}) \tag{A.3}$$

where

$$n = 2 - 0.0372\left(\frac{T_r}{T_{bn}}\right)^2 \tag{A.3a}$$

$\rho_{F,288.6K}$ - fuel density in kg/m^3 at 288.6K.

Liquid fuel density[3] [$\frac{kg}{m^3}$] is needed to calculate λ_{st}:

$$\rho_F = \rho_{F,288.6K}\left[1 - 1.8C_{ex}(T_{drop} - 288.6) - 0.09\frac{(T_{drop} - 288.6)^2}{(T_{crit} - 288.6)^2}\right] \tag{A.4}$$

where C_{exp} is the expansion coefficient (provided in Table A.2).

Latent heat of fuel vaporization [$\frac{kJ}{kg}$] corresponding to fuel surface temperature is estimated by:

$$L = L_{T_{bn}} \left(\frac{T_{crit} - T_s}{T_{crit} - T_{bn}} \right)^{0.38} \tag{A.5}$$

L_{Tbn} – Latent heat of fuel vaporization at normal boiling point [$\frac{kJ}{kg}$]

Specific heat at constant pressure of fuel liquid is given by [$\frac{kJ}{kgK}$]:

$$c_{p,F} = (0.76 + 0.00335 T_{drop})(0.001 \rho_F)^{0.5} \tag{A.6}$$

Group 2 Fuels: Jet-A ($C_{12}H_{23}$) and Water (H_2O)

All equations for the fuel properties can be found in Yaws (ref. 7), as shown below:

Fuel liquid density [$\frac{g}{cm^3}$]:

$$\rho_F = AB^{-\left(1-\frac{T_{drop}}{T_c}\right)^n}$$

(A.7)

Coefficients	Jet-A ($C_{12}H_{23}$)	Water (H_2O)
A	0.29292	0.3471
B	0.26661	0.274
n	0.298	0.2857

Table A.3 – Coefficients for *Fuel Liquid Density* [$\frac{g}{cm^3}$] in Group 2 fuel.

Latent heat of vaporization of fuel liquid [$\frac{kJ}{mol}$]:

$$L = A\left(1 - \frac{T_{drop}}{T_c}\right)^n$$

(A.8)

Coefficients	Jet-A ($C_{12}H_{23}$)	Water (H_2O)
A	73.509	52.023
n	0.347	0.321

Table A.4 – Coefficients for *Latent Heat of Vaporization of fuel liquid* [$\frac{kJ}{mol}$] in Group 2 fuel.

Specific heat at constant pressure of fuel liquid $[\dfrac{J}{mol-K}]$:

$$C_{p,F} = A + BT_{drop} + CT_{drop}^2 + DT_{drop}^3 \qquad\qquad (A.9)$$

Coefficients	Jet-A ($C_{12}H_{23}$)	Water (H_2O)
A	142.238	92.053
B	1.5261	-0.039953
C	-0.0034477	-0.00021103
D	0.0000032968	0.0000005347

Table A.5 – Coefficients for *Specific Heat at constant pressure of fuel liquid* $[\dfrac{J}{mol-K}]$ in Group 2 fuel.

Specific heat at constant pressure of fuel vapor $[\dfrac{J}{mol-K}]$:

$$C_{p,Fv} = A + BT_r + CT_r^2 + DT_r^3 + ET_r^4 \qquad\qquad (A.10)$$

Coefficients	Jet-A ($C_{12}H_{23}$)	Water (H_2O)
A	-128.032	33.933
B	1.4622	-0.0084186
C	-0.00086193	0.000029906
D	0.00000018462	-0.000000017825
E	0.0000000000036227	3.6934E-12

Table A.6 – Coefficients for *Specific Heat at constant pressure of fuel vapor* $[\dfrac{J}{mol-K}]$ in Group 2 fuel.

Thermal conductivity of fuel vapor $[\dfrac{W}{m \cdot K}]$:

$$k_{Fv} = A + BT_r + CT_r^2 \qquad\qquad (A.11)$$

Coefficients	Jet-A ($C_{12}H_{23}$)	Water (H_2O)
A	-0.01184	0.00053
B	0.000061839	0.000047093
C	0.000000025082	0.000000049551

Table A.7 – Coefficients for *Thermal Conductivity for fuel vapor* $[\dfrac{W}{m \cdot K}]$ in Group 2 fuel.

Vapor pressure [mmHg]:

$$\log_{10} P_{vap} = A + \frac{B}{T_s} + C \log_{10} T_s + D T_s + E T_s^2 \tag{A.12}$$

Then, let

$$P_{F,s} = P_{vap}(T_s) \tag{A.12a}$$

Coefficients	Jet-A ($C_{12}H_{23}$)	Water (H_2O)
A	-50.5512	29.8605
B	-2705.3	-3152.2
C	28.273	-7.3037
D	-0.045702	0.00000024247
E	0.000020443	0.000001809

Table A.8 – Coefficients for *Vapor Pressure* [mmHg] in Group 2 fuel.

Appendix B—Air Properties

Variation of air properties with temperature (range 100K through 2000K) can be obtained from correlation provided in Incorpera[5]. Air properties are all calculated in subroutine `air()`.

Thermal conductivity [$\frac{W}{m \cdot K}$]:

$$k_A = 2 \cdot 10^{-11} T_r^3 - 5 \cdot 10^{-8} T_r^2 - 0.0969 \cdot 10^{-3} T_r + 0.8289 \cdot 10^{-3} \qquad \text{(B.1)}$$

Specific heat at constant pressure [$\frac{kJ}{kg}$]:

$$c_{p,A} = -5 \cdot 10^{-11} T_r^3 + 2 \cdot 10^{-7} T_r^2 - 1 \cdot 10^{-5} T_r + 1.0041 \qquad \text{(B.2)}$$

Viscosity [$\frac{Ns}{m^2}$]:

$$\mu_A = (7 \cdot 10^{-8} T_\infty^3 - 0.0003 T_\infty^2 + 0.6227 T_\infty + 18.761) \cdot 10^{-7} \qquad \text{(B.3)}$$

Prandtl number:

$$\text{Pr}_A = 2 \cdot 10^{-13} T_\infty^4 - 10^{-9} T_\infty^3 + 2 \cdot 10^{-6} T_\infty^2 - 0.0009 T_\infty + 0.8632 \qquad \text{(B.4)}$$

Density [$\frac{kg}{m^3}$]:

$$\rho_A = 355.91 T_\infty^{-1.0032} \qquad \text{(B.5)}$$

Appendix C—Microsoft® Visual Basic Code

In the Microsoft® Excel spreadsheet, select '**Tools→Macro→Visual Basic Editor**', a new window will open showing all the numerical codes in different sheets or modules.

Main Code

```
Private Sub CommandButton3_Click()
'Declare ballistics variables

    Dim n As Long, nn As Double, nnnn As Long, speedup As Long
    Dim Cdm As Double, Rems As Double
    Dim dtt As Double
    Dim k1 As Double, k2 As Double, k3 As Double, k4 As Double, l1 As Double, l2 As Double, l3 As Double, l4
As Double
    Dim kz1 As Double, kz2 As Double, kz3 As Double, kz4 As Double, lz1 As Double, lz2 As Double, lz3 As
Double, lz4 As Double
    Dim Urm As Double
    Dim a As Double, b As Double
    Dim location
    Dim sf
    Dim a1, b1, c1, d1, e1, f1, g1, h1
    Dim unitconv

'Adding these inputs for drop evaporation '(3/26/03)
    Dim md1 As Double, md2 As Double, md3 As Double, md4 As Double
    Dim mt1 As Double, mt2 As Double, mt3 As Double, mt4 As Double
    Dim evap As Double

'Variable time
    Dim begin, last, switch
        begin = Second(Time) + Minute(Time) * 60 + Hour(Time) * 3600
    switch = 0


'Clear data
    Call Macro2
    Halt = False
    Msg = "Do you want to continue ?"
    Style = vbYesNo
    Style1 = vbOKOnly
    Title = "Jet Flow in CrossFlow"
    Title1 = "Evaporation Time Estimate"
    Msg3 = "Drop vaporizes completely!"
    Msg_size = "Please reduce droplet size!"
    Msg_size1 = "Can't estimate. Drop size too big!"

    Call taperd

'Dynamic array size
    asize = kl + 2
    ReDim xx(0 To asize) As Double, xp(0 To asize) As Double
    ReDim zz(0 To asize) As Double, zp(0 To asize) As Double
    ReDim D(0 To asize) As Double, Ts(0 To asize) As Double
```

```
 'Sheets("Process").Cells(9, 10) = kl
 'Sheets("Process").Cells(10, 10) = asize
 If flag = 4 And r > 0.0001 Then
    Response = MsgBox(Msg_size, Style1, Title1)
    Cells(5, 3).Select
    GoTo 50
 End If

 Pi = 22# / 7#

    Worksheets("Process").CommandButton1.Width = 0
    Worksheets("Process").CommandButton1.Visible = True

    n = 0    'used for array value index
    nn = -1  'used for writing data into cell
    nnn = 2  'used for writing into sequence cell
    dtt = 0
    evap = 0#
    nl = 0   'used to keep track the time level recorded when drop vaporizes completely
    ss = 0   'used to turn on/off switch if temperature has reached steady
    sst = 0  'used to print steady state time
    speedup = Int(kl / 50) 'used to update the percentage bar

    'initial position and velocity
    xx(n) = xxi
    xp(n) = xpi
    zz(n) = zzi
    zp(n) = zpi

    'Assume uniform temperature within the drop
    Ts(n) = Tinit

    D(n) = 2# * r
    D02 = D(n) ^ 2

    If flag = 3 Or flag = 4 Then Sheets("Process").Cells(18, 10) = "Temperature is not steady"

    'Determine fuel type then select one of two approach to calculate fuel properties
    ' 0 - Lebefvre's, 1 - Yaws'
    fuel_type = Sheets("Process").ComboBox1.Value
    Select Case fuel_type
    Case "DF 2"
       otherfuel = 0
    Case "Jet-A (C12H23)"
       otherfuel = 1
       Apv = -50.5512
       Bpv = -2705.3
       Cpv = 28.273
       Dpv = -0.045702
       Epv = 0.000020443
       '
       Arho = 0.29292
       Brho = 0.26661
       nrho = 0.298
       '
```

```
   Akfv = -0.01184
   Bkfv = 0.000061839
   Ckfv = 0.000000025082
   '
   Acpfv = -128.032
   Bcpfv = 1.4622
   Ccpfv = -0.00086193
   Dcpfv = 0.00000018462
   Ecpfv = 3.6227E-13
   '
   Acpf = 142.238
   Bcpf = 1.5261
   Ccpf = -0.0034477
   Dcpf = 0.0000032968
   '
   Alat = 73.509
   nlat = 0.347

Case "JP 4"
   otherfuel = 0
Case "JP 5"
   otherfuel = 0
Case "n-Heptane"
   otherfuel = 0
Case "Water"
   otherfuel = 1
   Apv = 29.8605
   Bpv = -3152.2
   Cpv = -7.3037
   Dpv = 0.00000024247
   Epv = 0.000001809
   '
   Arho = 0.3471
   Brho = 0.274
   nrho = 2# / 7#
   '
   Akfv = 0.00053
   Bkfv = 0.000047093
   Ckfv = 0.000000049551
   '
   Acpfv = 33.933
   Bcpfv = -0.0084186
   Ccpfv = 0.000029906
   Dcpfv = -0.000000017825
   Ecpfv = 3.6934E-12
   '
   Acpf = 92.053
   Bcpf = -0.039953
   Ccpf = -0.00021103
   Dcpf = 0.0000005347
   '
   Alat = 52.023
   nlat = 0.321
Case Else
   otherfuel = 0
End Select
```

```
If onoff_flag = 0 Then
  Tref = Tinit + (Tinf - Tinit) / 3#
  'air: viscosity
  mu = 0.00000007 * Tref ^ 3 - 0.0003 * Tref ^ 2 + 0.6227 * Tref + 18.761
  mu = mu * 0.0000001 'Ns/m^2

  If otherfuel = 0 Then
   'fuel: liquid density
    rhof = rhofr * (1# - 1.8 * cexp * (Tinit - 288.6) - 0.09 * ((Tinit - 288.6) / (Tcrit - 288.6)) ^ 2#) 'kg/m^3
  Else
   'fuel: liquid density
    Tdocrit = Tinit / Tcrit
    rhof = 1000# * Arho * Brho ^ ((-1#) * (1# - Tdocrit) ^ nrho) 'convert g/cm^3' to 'kg/m^3
  End If

  'air: density
   rhog = 355.91 * Tref ^ (-1.0032) 'kg/m^3
End If


Sheets("Process").Cells(14, 12) = "Calculating"
20
    dtt = dtt + dt
    If flag = 1 Or flag = 2 And onoff_flag = 1 Then Call driver_for_runge(Ts(n))
    If flag = 3 And ss = 0 And onoff_flag = 1 Then Call driver_for_runge(Ts(n))
    If flag = 4 And ss = 0 And onoff_flag = 1 Then Call driver_for_runge(Ts(n))
    Call Runge(xp(n), zp(n), D(n), Ts(n))

    If (SuperExit) Then GoTo 50
    k1 = dt * xp(n) 'replaced xp(n) 3/26/03
    l1 = dt * A_x
    kz1 = dt * zp(n) 'replaced zp(n) 3/26/03
    lz1 = dt * A_z
    If onoff_flag = 1 Then
      md1 = dt * (-0.5) * lambda / D(n) '3/28/03
      mt1 = dt * ct
      If ((D(n) + 0.5 * md1) < 0.00000001) Then
        evap = 1#
        nl = n
        GoTo 15
      End If
    End If
    If flag = 1 And onoff_flag = 1 Then Call driver_for_runge(Ts(n) + 0.5 * mt1)
    If flag = 3 And ss = 0 And onoff_flag = 1 Then Call driver_for_runge(Ts(n) + 0.5 * mt1)
    Call Runge(xp(n) + l1 / 2#, zp(n) + lz1 / 2#, D(n) + 0.5 * md1, Ts(n) + 0.5 * mt1)

    If (SuperExit) Then GoTo 50
    k2 = dt * (xp(n) + l1 / 2#) 'replaced xp(n) 3/26/03
    l2 = dt * A_x
    kz2 = dt * (zp(n) + lz1 / 2#) 'replaced zp(n) 3/26/03
    lz2 = dt * A_z
```

```
If onoff_flag = 1 Then
     md2 = dt * (-0.5) * lambda / (D(n) + 0.5 * md1) '3/28/03
     mt2 = dt * ct
     If ((D(n) + 0.5 * md2) < 0.00000001) Then
       evap = 1#
       nl = n
       GoTo 15
     End If
   End If
   If flag = 1 And onoff_flag = 1 Then Call driver_for_runge(Ts(n) + 0.5 * mt2)
   If flag = 3 And ss = 0 And onoff_flag = 1 Then Call driver_for_runge(Ts(n) + 0.5 * mt2)
   Call Runge(xp(n) + l2 / 2#, zp(n) + lz2 / 2#, D(n) + 0.5 * md2, Ts(n) + 0.5 * mt2)

   If (SuperExit) Then GoTo 50
   k3 = dt * (xp(n) + l2 / 2#) 'replaced xp(n) 3/26/03
   l3 = dt * A_x
   kz3 = dt * (zp(n) + lz2 / 2#) 'replaced zp(n) 3/26/03
   lz3 = dt * A_z
   If onoff_flag = 1 Then
     md3 = dt * (-0.5) * lambda / (D(n) + 0.5 * md2) '3/28/03
     mt3 = dt * ct
     If ((D(n) + md3) < 0.00000001) Then
       evap = 1#
       nl = n
       GoTo 15
     End If
   End If
   If flag = 1 And onoff_flag = 1 Then Call driver_for_runge(Ts(n) + mt3)
   If flag = 3 And ss = 0 And onoff_flag = 1 Then Call driver_for_runge(Ts(n) + mt3)
   Call Runge(xp(n) + l3, zp(n) + lz3, D(n) + md3, Ts(n) + mt3)


   If (SuperExit) Then GoTo 50
   k4 = dt * (xp(n) + l3) 'replaced xp(n) 3/26/03
   l4 = dt * A_x
   kz4 = dt * (zp(n) + lz3) 'replaced zp(n) 3/26/03
   lz4 = dt * A_z
   If onoff_flag = 1 Then
     md4 = dt * (-0.5) * lambda / (D(n) + md3) '3/28/03
     mt4 = dt * ct
   End If

   If flag = 3 Or flag = 4 And ss = 0 And onoff_flag = 1 Then
     If Abs(ct) < sstemp Then ss = 1
   End If

   If flag = 3 And ss = 1 And sst = 0 And onoff_flag = 1 Then
     Sheets("Process").Cells(18, 10) = "Temperature is steady at t = " & dtt * 1000# & "milliseconds"
     sst = 1
   End If


   xp(n + 1) = xp(n) + (1# / 6#) * (l1 + 2# * l2 + 2# * l3 + l4)
   zp(n + 1) = zp(n) + (1# / 6#) * (lz1 + 2# * lz2 + 2# * lz3 + lz4)

   xx(n + 1) = xx(n) + (1# / 6#) * (k1 + 2# * k2 + 2# * k3 + k4)
```

```
zz(n + 1) = zz(n) + (1# / 6#) * (kz1 + 2# * kz2 + 2# * kz3 + kz4)

If onoff_flag = 1 Then
 D(n + 1) = D(n) + (1# / 6#) * (md1 + 2# * md2 + 2# * md3 + md4) '3/28/03
 Ts(n + 1) = Ts(n) + (1# / 6#) * (mt1 + 2# * mt2 + 2# * mt3 + mt4)
End If
DoEvents
If (Halt) Then
 DoEvents
 Halt = False
 Response = MsgBox(Msg, Style, Title)
 If Response = vbNo Then
  evap = 2#
  nl = n
  GoTo 15
 End If
End If

If 0 = (n Mod speedup) Or n >= kl Then
 Worksheets("Process").CommandButton1.Width = ((n / kl) * 165.75)
 Worksheets("Process").CommandButton1.Caption = ((n / kl) * 100) & "%"
 Worksheets("Process").CommandButton1.Height = 20.25
End If

If flag = 4 And ss = 1 And onoff_flag = 1 Then
 Sheets("Process").Cells(14, 12) = "Estimation done"
 Sheets("Process").Cells(18, 10) = "Temperature is steady at t = " & dtt & "seconds"
 Sheets("Process").Cells(6, 10) = D02 / lambda
 Sheets("Process").Cells(5, 10) = lambda
 dt_temp = dt
 dt = dt_old
 'Sheets("Process").Cells(9, 10) = n
 'Sheets("Process").Cells(10, 10) = kl
 'Sheets("Process").Cells(7, 10) = dt '9/10/03
 'Sheets("Process").Cells(8, 10) = dt_temp
 GoTo 50
Else
 If flag = 4 And n >= kl And ss = 0 Then
  Response = MsgBox(Msg_size1, Style1, Title1)
  dt_temp = dt
  dt = dt_old
  GoTo 50
 End If
End If

If (D(n + 1) < 0.00000001) And onoff_flag = 1 Then
  evap = 1#
  nl = n
  GoTo 15
End If
```

```
  n = n + 1
      If n <= kl Then
        GoTo 20
      End If


15
      Sheets("Process").Cells(14, 12) = "Writing data into cells"
      If evap = 1# Then
        nf = nl - 1
      ElseIf evap = 2# Then
        nf = nl - 1
      ElseIf evap = 0# Then
        nf = kl
      End If
n = 0
ttldata = Sheets("Process").Cells(35, 5)
sd = Sheets("Process").Cells(35, 5)
speedup = Int(sd / 10)


25
      Urm = (((xp(n) - ug) ^ 2) + ((zp(n) - wg) ^ 2)) ^ 0.5
      If onoff_flag = 1 Then Rems = rhog * Urm * D(n) / mu
      If onoff_flag = 0 Then Rems = rhog * Urm * D(0) / mu

      If Rems <= 1000 Then
        If Rems = 0# Then
          Cdm = 0#
        Else
          Cdm = (((Rems ^ (2# / 3#)) / 6#) + 1#) * 24# / Rems
        End If
      Else
          Cdm = 0.424
      End If
      Sheets("Data").Cells(nnn, 1) = n * dt
      Sheets("Data").Cells(nnn, 2) = xx(n)
      Sheets("Data").Cells(nnn, 3) = zz(n)
      Sheets("Data").Cells(nnn, 4) = xp(n)
      Sheets("Data").Cells(nnn, 5) = zp(n)
      Sheets("Data").Cells(nnn, 6) = Cdm
      If onoff_flag = 1 Then
        Dplot = (D(n) / D(0)) ^ (2#)
        Sheets("Data").Cells(nnn, 7) = Dplot
        Sheets("Data").Cells(nnn, 8) = Ts(n)
      End If

      If 0 = (nnn Mod speedup) Or n >= nf Then
        Worksheets("Process").CommandButton1.Width = (((nnn - 1) / ttldata) * 165.75)
        Worksheets("Process").CommandButton1.Caption = (((nnn - 1) / ttldata) * 100) & "%"
        Worksheets("Process").CommandButton1.Height = 20.25
      End If

      DoEvents
      If (Halt) Then
        DoEvents
        Halt = False
        Response = MsgBox(Msg, Style, Title)
```

```
      If Response = vbNo Then
        GoTo 10
      End If
    End If


    n = n + userchoice
    nnn = nnn + 1
    If n > nf Then
      GoTo 10
    Else
      GoTo 25
    End If

10
  last = Second(Time) + Minute(Time) * 60 + Hour(Time) * 3600
  Sheets("Process").Cells(16, 10) = last - begin
  If onoff_flag = 1 Then
    Sheets("Evap").Cells(4, 10) = D02 / lambda
    Sheets("Evap").Cells(3, 10) = lambda
  End If
  If (evap = 1#) Then
'     nnn = nnn + 1
'     Sheets("Data").Cells(nnn, 1) = (nf + 1#) * dt
'     Sheets("Data").Cells(nnn, 7) = 0#
      Response = MsgBox(Msg3, Style1, Title)
  End If

  Macro1 'plot data on chart
  'sd = Sheets("Process").Cells(35, 5)
    'print variables in a file (Tecplot format)
  If Sheets("Process").Cells(46, 9) = "YES" Then
    location = Sheets("Process").Cells(23, 11)
    Sheets("Process").Cells(14, 12) = "Writing data into file : " + location
    Open location For Output As #1
    Print #1, "TITLE = "; Spc(2); """"; "Spray Jet In CrossFlow"; """"
    If onoff_flag = 1 Then
      If Sheets("Process").Cells(47, 9) = "millimeter" Then
        unitconv = 1000#
        Print #1, "Variables ="; Spc(2); """"; "Time(sec)"; """"; Spc(2); """"; "X(mm)"; """"; Spc(2); """"; "Z(mm)";
""""; Spc(2); """"; "U(mm/s)"; """"; Spc(2); """"; "W(mm/s)"; """"; Spc(2); """"; "Cd"; """"; Spc(2); """";
"D(Normalized)"; """"; Spc(2); """"; "T(K)"; """"
      Else
        unitconv = 1#
        Print #1, "Variables ="; Spc(2); """"; "Time(sec)"; """"; Spc(2); """"; "X(meter)"; """"; Spc(2); """";
"Z(meter)"; """"; Spc(2); """"; "U(m/s)"; """"; Spc(2); """"; "W(m/s)"; """"; Spc(2); """"; "Cd"; """"; Spc(2); """";
"D(Normalized)"; """"; Spc(2); """"; "T(K)"; """"
      End If
    Else
      If Sheets("Process").Cells(47, 9) = "millimeter" Then
        unitconv = 1000#
        Print #1, "Variables ="; Spc(2); """"; "Time(sec)"; """"; Spc(2); """"; "X(mm)"; """"; Spc(2); """"; "Z(mm)";
""""; Spc(2); """"; "U(mm/s)"; """"; Spc(2); """"; "W(mm/s)"; """"; Spc(2); """"; "Cd"; """"
      Else
        unitconv = 1#
        Print #1, "Variables ="; Spc(2); """"; "Time(sec)"; """"; Spc(2); """"; "X(meter)"; """"; Spc(2); """";
"Z(meter)"; """"; Spc(2); """"; "U(m/s)"; """"; Spc(2); """"; "W(m/s)"; """"; Spc(2); """"; "Cd"; """"
```

```
        End If
      End If
      sdd = 0

'plot data in a file (Tecplot format)
      Print #1, "ZONE I ="; nnn - 2 & ","; Spc(2); "F = POINT"
      nnn = nnn - 1
      For sf = 2 To nnn
       a1 = Sheets("Data").Cells(sf, 1)
       b1 = Sheets("Data").Cells(sf, 2)
       b1 = b1 * unitconv
       c1 = Sheets("Data").Cells(sf, 3)
       c1 = c1 * unitconv
       d1 = Sheets("Data").Cells(sf, 4)
       d1 = d1 * unitconv
       e1 = Sheets("Data").Cells(sf, 5)
       e1 = e1 * unitconv
       f1 = Sheets("Data").Cells(sf, 6)
       If onoff_flag = 1 Then
        g1 = Sheets("Data").Cells(sf, 7)
        h1 = Sheets("Data").Cells(sf, 8)
       End If
       If onoff_flag = 1 Then
         Print #1, a1; Spc(2); b1; Spc(2); c1; Spc(2); d1; Spc(2); e1; Spc(2); f1; Spc(2); g1; Spc(2); h1
       Else
         Print #1, a1; Spc(2); b1; Spc(2); c1; Spc(2); d1; Spc(2); e1; Spc(2); f1
       End If
      Next sf
      Close #1
     End If
     Sheets("Process").Cells(14, 12) = "Completed"
50
     SuperExit = False
End Sub
```

Other Subroutines

Sub driver_for_runge(Ts)

```
 Call transfernumber(Ts)
 Call reference_cond(Ts)
 Call air
 Call fuel_thermal_prop(Ts)

 'fuel-air mixture
 kg = yar * ka + yfr * kfv
 cpg = yar * cpa + yfr * cpfv

 BT = cpg * (Tinf - Ts) / lat

 'evaporation contant
 lambda = 8# * Log(1# + BM) * kg / (cpg * rhof)
End Sub
```

Sub transfernumber(Ts)

```
 If (otherfuel = 1) Then
   Log10T = Log(Ts) / Log(10#)
   Log10P = Apv + Bpv / Ts + Cpv * Log10T + Dpv * Ts + Epv * Ts * Ts
   pfs = 10# ^ (Log10P) * 0.13333 'convert 'mmHg' to 'Kpa'
   pfs = Application.WorksheetFunction.Min(p, pfs)
 Else
   pfs = Exp(acc - bcc / (Ts - 43#)) 'unit='Kpa'
   pfs = Application.WorksheetFunction.Min(p, pfs)
 End If

 yfs = 1# / (1# + (p / pfs - 1#) * (mwg / mwd))
 BM = yfs / (1# - yfs)
End Sub
```

Sub reference_cond(Ts)

```
 Tref = Ts + (Tinf - Ts) / 3#
 yfr = 2# * yfs / 3#
 yar = 1# - yfr
End Sub
```

Sub air()
```
 'air: density
 rhog = 355.91 * Tref ^ (-1.0032) 'kg/m^3

 'air: viscosity
 mu = 0.00000007 * Tref ^ 3 - 0.0003 * Tref ^ 2 + 0.6227 * Tref + 18.761
 mu = mu * 0.0000001 'Ns/m^2

 'air: prandtl number
```

Pra = 0.0000000000002 * Tref ^ 4 - 0.000000001 * Tref ^ 3 + 0.000002 * Tref ^ 2 - 0.0009 * Tref + 0.8632

    'air: thermal conductivity
    ka = 0.00000000002 * Tref ^ 3 - 0.00000005 * Tref ^ 2 + 0.0000969 * Tref + 0.0008289 'W/(mK)

    'air: specific heat
    cpa = -0.00000000005 * Tref ^ 3 + 0.0000002 * Tref ^ 2 - 0.00001 * Tref + 1.0041 'kJ/kg
    cpa = cpa * 1000# 'multiply by 1 kilo => J/(kgK)
End Sub

Sub fuel_thermal_prop(Ts)
  If (otherfuel = 1) Then
    'fuel: liquid density
    Tdocrit = Ts / Tcrit
    rhof = 1000# * Arho * Brho ^ ((-1#) * (1# - Tdocrit) ^ nrho) 'convert g/cm^3' to 'kg/m^3

    'fuel: vapor thermal conductivity
    kfv = Akfv + Bkfv * Tref + Ckfv * Tref * Tref 'W/(mK)

    'fuel: vapor specific heat at constant pressure
    cpfv = Acpfv + Bcpfv * Tref + Ccpfv * Tref * Tref + Dcpfv * Tref * Tref * Tref + Ecpfv * Tref * Tref * Tref *
Tref
    cpfv = cpfv * 1000# / mwd 'convert to J/(kgK)

    'fuel: liquid specific heat at constant pressure
    cpf = Acpf + Bcpf * Ts + Ccpf * Ts * Ts + Dcpf * Ts * Ts * Ts
    cpf = cpf * 1000# / mwd 'J/(kgK)

    'fuel: liquid latent heat of vaporization
    lat = Alat * (1# - Tdocrit) ^ nlat
    lat = lat * 1000000# / mwd 'J/kg
  Else
    'fuel: liquid density
    rhof = rhofr * (1# - 1.8 * cexp * (Ts - 288.6) - 0.09 * ((Ts - 288.6) / (Tcrit - 288.6)) ^ 2#) 'kg/m^3

    'fuel: vapor thermal conductivity
    nkfv = 2# - 0.0372 * (Tref / Tbn) ^ 2
    kfv = 0.000001 * (13.2 - 0.0313 * (Tbn - 273#)) * (Tref / 273#) ^ nkfv 'W/(mK)

    'fuel: vapor specific heat at constant pressure
    cpfv = (0.363 + 0.000467 * Tref) * (5# - 0.001 * rhofr) 'kJ/(kgK)
    cpfv = cpfv * 1000# 'J/(kgK)

    'fuel: liquid specific heat at constant pressure
    cpf = (0.76 + 0.00335 * Ts) * (0.001 * rhof) 'kJ/(kgK)
    cpf = cpf * 1000# 'J/(kgK)

    'fuel: liquid latent heat of vaporization
    lat = latbn * ((Tcrit - Ts) / (Tcrit - Tbn)) * (0.38) 'kJ/kg
    lat = lat * 1000# 'J/(kgK)
  End If
End Sub

Sub taperd()

```
    fl = Sheets("Process").ComboBox2.Value
    If fl = "Option 1" Then flag = 1
    If fl = "Option 2" Then flag = 2
    If fl = "Option 3" Then flag = 3
    If fl = "Evap time estimate" Then flag = 4

    fl = Sheets("Process").ComboBox3.Value
    If fl = "ON" Then onoff_flag = 1
    If fl = "OFF" Then onoff_flag = 0

'Getting the input
    r = Sheets("Process").Cells(5, 3)
    r = r / 1000000#
    rhofr = Sheets("Process").Cells(8, 3)
    Tcrit = Sheets("Process").Cells(9, 3)
    Tbn = Sheets("Process").Cells(10, 3)
    cexp = Sheets("Process").Cells(11, 3)
    latbn = Sheets("Process").Cells(12, 3)
    mwd = Sheets("Process").Cells(13, 3)
    acc = Sheets("Process").Cells(14, 3)
    bcc = Sheets("Process").Cells(15, 3)

    xxi = Sheets("Process").Cells(17, 3)
    xpi = Sheets("Process").Cells(18, 3)
    zzi = Sheets("Process").Cells(19, 3)
    zpi = Sheets("Process").Cells(20, 3)
    Tinit = Sheets("Process").Cells(21, 3)

    mwg = Sheets("Process").Cells(23, 3)
    ug = Sheets("Process").Cells(24, 3)
    wg = Sheets("Process").Cells(25, 3)

    Tinf = Sheets("Process").Cells(27, 3)
    p = Sheets("Process").Cells(28, 3)
    g = Sheets("Process").Cells(29, 3)


    dt = Sheets("Process").Cells(31, 3)
    kl = Sheets("Process").Cells(32, 3)
    If flag = 4 And onoff_flag = 1 Then
      dt_old = dt
      dt = 0.000001
      kl = 400000
      If r > 0.00005 Then kl = 800000
    End If
    userchoice = Sheets("Process").Cells(33, 3)
    sstemp = Sheets("Process").Cells(34, 3)

    area_noevap = Sheets("Process").Cells(6, 3)
    volume_noevap = Sheets("Process").Cells(7, 3)

End Sub
```

References

[1] Urip, E., Yang, S.L., and Marek C.J., "An Interactive Excel Program for Tracking a Single Droplet in Crossflow Computation," NASA/TM—2002-211710, August 2002.

[2] Leong, M.Y., McDonell, V.G., and Samuelsen, G.S., 2000, "Mixing of an Airblast-Atomized Fuel Spray Injected Into a Crossflow of Air," NASA/CR—2000-210467.

[3] Chin, J.S., and Lefebvre, A.H., "Steady-state Evaporation Characteristics of Hydrocarbon Fuel Drops," *AIAA journal,* Vol. 21, No. 10, 1983, pp 1437–1443.

[4] Spiers, H.M., ed., *Technical Data on Fuels*, The British Committee, World Power Conference, London, 1961.

[5] Incropera, F.P., and DeWitt D.P., *Fundamentals of Heat and Mass Transfer*, 5th Edition, New York, John Wiley & Sons, 2002.

[6] Chin, J.S., and Lefebvre, A.H., "The role of the Heat-up Period in Fuel Drop Evaporation," *Int. J. Turbo Jet Engines,* Vol. 2, 1985, pp 315–325.

[7] Carl L. Yaws, *Chemical Properties Handbook*, McGraw-Hill, New York, 1999.

[8] Nakamura, Shoichiro, *Applied Numerical Method With Software*, Englewood Cliffs, New Jersey: Prentice Hall, 1991.

1. AGENCY USE ONLY (Leave blank)	2. REPORT DATE July 2004	3. REPORT TYPE AND DATES COVERED Technical Memorandum

4. TITLE AND SUBTITLE	5. FUNDING NUMBERS
An Interactive Microsoft® Excel Program for Tracking a Single Evaporating Droplet in Crossflow	
6. AUTHOR(S) K.H. Liew, E. Urip, S.L. Yang, and C.J. Marek	WBS–22–714–20–10

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
National Aeronautics and Space Administration John H. Glenn Research Center at Lewis Field Cleveland, Ohio 44135–3191	E–14342

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITORING AGENCY REPORT NUMBER
National Aeronautics and Space Administration Washington, DC 20546–0001	NASA TM—2004-212910

12a. DISTRIBUTION/AVAILABILITY STATEMENT	12b. DISTRIBUTION CODE
Unclassified - Unlimited Subject Category: 07 Distribution: Nonstandard Available electronically at http://gltrs.grc.nasa.gov This publication is available from the NASA Center for AeroSpace Information, 301–621–0390.	

13. ABSTRACT *(Maximum 200 words)*

Droplet interaction with a high temperature gaseous crossflow is important because of its wide application in systems involving two phase mixing such as in combustion requiring quick mixing of fuel and air with the reduction of pollutants and for jet mixing in the dilution zone of combustors. Therefore, the focus of this work is to investigate dispersion of a two-dimensional atomized and evaporating spray jet into a two-dimensional crossflow. An interactive Microsoft® Excel program for tracking a single droplet in crossflow that has previously been developed will be modified to include droplet evaporation computation. In addition to the high velocity airflow, the injected droplets are also subjected to combustor temperature and pressure that affect their motion in the flow field. Six ordinary differential equations are then solved by 4th-order Runge-Kutta method using Microsoft® Excel software. Microsoft® Visual Basic programming and Microsoft® Excel macrocode are used to produce the data and plot graphs describing the droplet's motion in the flow field. This program computes and plots the data sequentially without forcing the user to open other types of plotting programs. A user's manual on how to use the program is included.

14. SUBJECT TERMS		15. NUMBER OF PAGES 50
Droplet; Evaporation; Trajectory; Microsoft® Excel spreadsheet; Jet-A; Water; Diesel; JP-4; JP-5; N-heptane		16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	