NASA/CR—2015-218918

LSPRAY-V: A Lagrangian Spray Module

M.S. Raju
Vantage Partners, LLC, Brook Park, Ohio

November 2015

NASA STI Program . . . in Profile

NASA/CR—2015-218918

LSPRAY-V: A Lagrangian Spray Module

M.S. Raju
Vantage Partners, LLC, Brook Park, Ohio

Prepared under Contract NNC12BA01B

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

November 2015

Acknowledgments

This work was supported by NASA with Dr. Jeff Moder acting as the technical monitor. The author would like to extend his sincere appreciation to him for his many valuable contributions to the spray code.

This work was sponsored by the Fundamental Aeronautics Program at the NASA Glenn Research Center.

Level of Review: This material has been technically reviewed by NASA technical management OR expert reviewer(s).

Available from

NASA STI Program
Mail Stop 148
NASA Langley Research Center
Hampton, VA 23681-2199

National Technical Information Service
5285 Port Royal Road
Springfield, VA 22161
703-605-6000

This report is available in electronic form at http://www.sti.nasa.gov/ and http://ntrs.nasa.gov/

LSPRAY-V: A Lagrangian Spray Module

M.S. Raju
Vantage Partners, LLC
Brook Park, Ohio 44142

ABSTRACT

LSPRAY-V is a Lagrangian spray solver developed for application with unstructured grids and massively parallel computers. It is mainly designed to predict the flow, thermal and transport properties of a rapidly vaporizing spray encountered over a wide range of operating conditions in modern aircraft engine development. It could easily be coupled with any existing gas-phase flow and/or Monte Carlo Probability Density Function (PDF) solvers. The manual provides the user with an understanding of various models involved in the spray formulation, its code structure and solution algorithm, and various other issues related to parallelization and its coupling with other solvers. With the development of LSPRAY-V, we have advanced the state-of-the-art in spray computations in several important ways.

- It facilitates the use of both structured & unstructured grids and parallel computing and, thereby, facilitates large-scale combustor computations involving complex geometrical configurations. The solver accommodates the use of an unstructured mesh with mixed elements of either triangular, quadrilateral, and/or tetrahedral type.

- In order to deal with modern gas-turbine fuels that are mixtures of many compounds, it takes into account the modeling of multicomponent liquid fuels with variable properties.

- Various well-established vaporization and atomization models are incorporated into our spray code to cover a wide range of engine operating conditions: low to high pressures including supercritical conditions, and superheat conditions associated with flash vaporization. The initial droplet conditions could be prescribed based on either a single-point or multi-point droplet injection. The multi-point injection could be in the form of a line or circular point injection. The initial droplet conditions could be specified entirely in the form of a table based on known experimental data, or some of the initial conditions could be calculated based on several widely-used droplet-size distribution functions or primary atomization models incorporated into our spray code. For low pressures, we have incorporated the following primary atomization models: (a) sheet breakup, (b) air blast, (c) blob jet, or (d) BLS (Boundary-Layer Stripping). We have also incorporated a flash induced atomization model to cover superheat conditions. The secondary droplet breakup following primary atomization could be modeled by one of the following models: (a) Rayleigh-Taylor, (b) TAB (Taylor Analogy Breakup), or (c) ETAB (Enhanced Taylor Analogy Breakup).

- The spray module has a multi-liquid and multi-injector capability.

- Our spray code supports all of the boundary conditions of the national combustion code including particle movement through very complex periodic boundary conditions. Upon impact with the wall, a droplet may shatter, rebound, or stick to the wall depending on the level and conditions of collision impact. We have implemented several droplet-wall interaction models into our spray code to cover a wide range of conditions.

- Our spray code can be used in the calculation of both steady as well as unsteady computations.

- The spray module is designed in such a way so that it could easily be coupled with other CFD codes.

With the aim of improving the overall solution procedure of the national combustion code involving sprays, we have made several other relevant contributions to the gas-side of computations:

- In order to demonstrate the importance of chemistry/turbulence interactions in the modeling of reacting sprays, we have extended the joint scalar Monte Carlo PDF (Probability Density Function) approach to the modeling of spray calculations with unstructured grids, and parallel computing.
- In order to account for non-ideal gas behavior associated with high-pressure conditions, we have completed the implementation of a high-pressure EOS (equation of state). Also, we have implemented several high-pressure corrections that go into the calculation of gas-phase transport properties.

The spray solution procedure provided favorable results when applied to the modeling of various reacting/non-reacting flows encountered in gas-turbine combustors, stratified-charge rotary combustion (Wankel) engines, supersonic and pulse detonation combustion devices. Its use has been demonstrated in various NASA projects: the NASA's fundamental aeronautics program initiative on emissions through the Supersonics (SUP) and Subsonic Fixed Wing (SFW) project offices, Ultra-Efficient Engine Technology (UEET), Pulse Detonation Combustion Technology (PDCT), & Rotary Combustion Engine Technology Enablement Project (RCETEP).

NOMENCLATURE

a	liquid jet radius, m, or parent drop radius, m
a_o	initial droplet radius, m
$\underline{a}_n$	outward area normal vector of the nth surface, m^2
A	a constant
B_k	Spalding mass transfer number
B_t	Spalding heat transfer number
B_o	a constant
c_{freq}	a constant
C	a constant
C_D	drag coefficient
C_p	specific heat, J/(kg K)
D	turbulent diffusion coefficient, m^2/s
D_{AB}	diffusion coefficient, cm^2/s
d	droplet diameter, m
d_b	droplet diameter after primary breakup, m
d_L	ligament diameter, m
d_o	orifice diameter, m
d_{parent}	parent drop diameter, m
$d_{product}$	product drop diameter, m
dt	time increment, s
$d\theta$	half-cone angle, deg.
E_{parent}	energy contained in the parent drop, (kg m^2)/s^2
$E_{product}$	energy contained in the product drop, (kg m^2)/s^2
h	specific enthalpy, J/kg, liquid sheet thickness, m
I_0, I_1	modified Bessel functions of the first kind
k	turbulence kinetic energy, $\mathrm{m}^2/\mathrm{s}^2$, wave number, 1/m, or thermal conductivity, J/(ms K)
k_{ij}	binary interaction parameter
K_b	a breakup constant, 1/s
K_{sb}	wave number associated with sheet breakup, 1/m
k_1, k_2	constants
K_{Lb}	wave number associated with ligament breakup, 1/m
K_0, K_1	modified Bessel functions of the second kind
l	modified wave number, 1/m
l_k	mixture latent heat of evaporation, J/kg
$l_{k,eff}$	effective latent heat of evaporation, J/kg (defined in Eq. (35))
l_{kn}	heat of vaporization at normal boiling point, J/kg
L	liquid sheet breakup length, m
$\dot{m}$	liquid mass flow rate, kg/s
$\dot{m}_k$	droplet vaporization rate, kg/s
$\dot{m}_{k,flash}$	droplet vaporization rate under flash evaporating conditions, kg/s
$\dot{m}_{ko}$	initial mass flow rate associated with kth droplet group, kg/s
$\dot{m}_{k,t}$	droplet vaporization rate due to heat transfer, kg/s
m_o	mass of the parent drop, kg
$\bar{m}(t)$	mean mass of the product drop distribution, kg
M	molecular weight, kg/kg-mole
M_a	molecular weight of gas excluding fuel species, kg/kg-mole
M_f	molecular weight of fuel, kg/kg-mole
M_i	molecular weight of ith species, kg/kg-mole
M_{shed}	shed drop mass, kg
MMD	mass mean diameter, m
n_k	number of droplets in kth group
n_{parent}	drop number in the parent group
$n_{product}$	drop number in the product group
n(t)	non-dimensional number ($=m_0/\bar{m}(t)$)
N	drop number
N_o	parent drop number
N_f	number of surfaces contained in a given computational cell
N_p	total number of computational cells
Nu	Nusselt number
P	pressure, $\mathrm{N/m}^2$
P_c	critical pressure, $\mathrm{N/m}^2$
Pr	Prandtl number
P_r	reduced pressure, P/P_c
P_{sat}	saturation pressure, $\mathrm{N/m}^2$
Q	density ratio ($=\rho_g/\rho_l$)
$\bar{r}$	radius ($=\sqrt{a^3/r_{SMR}}$), m
r_k	droplet radius, m
r_{ko}	initial droplet radial location, m
r_{SMR}	Sauter mean radius, m
R_u	gas constant, J/(kg K)
Re	Reynolds number
RND	random number
Sh	Sherwood number
s_k	droplet radius-squared ($=\mathrm{r}_k^2$), m^2
s_{mlc}	source term contribution from liquid exchange to mass conservation, kg/s
s_{mle}	source term contribution from liquid exchange to energy conservation, J/s
s_{mlm}	source term contribution from liquid exchange to momentum conservation, kg $\mathrm{m/s}^2$

s_{mls}	source term contribution from liquid exchange to species conservation, kg/s
SMD	Sauter mean diameter, m
t	time, s
t_d	non-dimensional time $(=\frac{2\rho_l a^2}{5\mu_l})$
T	temperature, K, or non-dimensional number $(=ZWe_g^{0.5})$
T_b	boiling temperature, K
T_{nb}	normal boiling temperature, K
T_c	critical temperature, K
T_k	kth droplet temperature, K
T_r	reduced temperature, T/T_c
$\dot{u}_d$	drop deceleration, m/s^2
u_i	ith velocity component, m/s
u_{ik}	ith velocity component of kth droplet group, m/s
U	relative velocity between gas and liquid, m/s, or relative drop velocity, m/s
v_T	product drop velocity component, m/s
V	initial liquid velocity, m/s
V_c	volume of the computational cell, m^3, or critical molar volume, m^3/kg-mole
V_n	molar volume at normal pressure, m^3/kg-mole
We	Weber number $(=\frac{\rho_g h U^2}{\sigma})$
$\dot{w}_j$	gas-phase chemical reaction rate, 1/s
x	Cartesian coordinate, m, or drop deformation distance, m
x_i	Cartesian coordinate in the ith direction, m
y	Cartesian coordinate, m, or non-dimensional deformation parameter $(=\frac{2x}{a})$
y_j	mass fraction of jth species
$\underline{x}$	spatial vector
z	Cartesian coordinate, m
Z	compressibility factor $(=\frac{PV}{RT})$, or non-dimensional number $(=\frac{We_l^{0.5}}{Re_l})$
Z_c	critical compressibility factor $(=\frac{P_c V_c}{RT_c})$
α	represents a coordinate related to a Hill's Vortex, spray cone rotation angle, deg., or a constant
α_s	overall heat transfer coefficient, kJ/sm^{2o}K
α_1	a constant
α_2	a constant
β	spray cone rotation angle, deg.
χ	mole fraction
δ	Dirac-delta function, or initial liquid sheet thickness, m
Δp	pressure drop in the injector, N/m^2
Δt_f	local time step used in the flow solver, s
Δt_{gl}	global time step used in the spray solver, s
Δt_{il}	fuel injection time step, s
Δt_{ml}	allowable time step in the spray solver, s
ΔV	computational cell volume, m^3
ϵ	rate of turbulence dissipation, m^2/s^3
ϵ_j	fractional mass evaporating rate of species at the droplet surface
η	wave amplitude, m
Γ_ϕ	turbulent diffusion coefficient, kg/ms, or a factor $(=210\big[\frac{T_c M^3}{P_c^4}\big]^{1/6})$
λ	thermal conductivity, J/(ms K), or wavelength, m
Λ	wavelength associated with Ω, or wavelength associated with Rayleigh-Taylor breakup, m
μ	dynamic viscosity, kg/ms
ν	kinematic viscosity, m^2/s
ω	turbulence frequency, 1/s, frequency $(=\frac{8\sigma}{\rho_l a^3} - \frac{1}{t_d^2})$, 1/s, complex growth rate, 1/s, or acentric factor
Ω	maximum growth rate, 1/s
ρ	density, kg/m^3
ρ_{ln}	liquid density (at 1 bar, 273.15 K), kg/m^3
ρ_r	reduced density $(=\rho/\rho_c)$
σ	potential length constant, Angstrom (=0.1 nm), surface tension, kg/s^2, or characteristic diameter of a molecule in Table 1
τ	stress tensor term, kg/ms^2, or characteristic breakup time $(=1/K_b)$, s
θ	void fraction, or spray cone angle, deg.

Subscripts

A	A-th species component
b	breakup conditions
B	B-th species component
c	critical conditions
f	fuel
g	gas or global
i	i-th coordinate, i-th species component, summation index, or imaginary component
inj	injector
j	j-th coordinate, j-th species component, or summation index
k	k-th droplet group, k-th coordinate, summation index, or liquid conditions associated with k-th droplet group

l	liquid
L	liquid ligament
m	multi-component mixture
n	normal, or nth-face of the computational cell
o	initial conditions, orifice exit conditions, injector initial conditions, or oxidizer
p	particle, drop, or conditions associated with a grid cell
r	reduced, radial coordinate, real component, or a reference value
RT	Rayleigh-Taylor
s	droplet surface, or adjacent computational cell
t	time
x	axial or x-coordinate
y	y-coordinate
z	z-coordinate
,	partial differentiation with respect to the variable followed by it

Superscripts

$\bar{\ }$	mean, or average
$\dot{\ }$	first order differentiation, or flow rate
$\ddot{\ }$	second order differentiation
$''$	fluctuations

1 INTRODUCTION

There are many occurrences of sprays in a variety of industrial and power applications and materials processing [1]. A liquid spray is a two phase flow with the gas as the continuous phase and the liquid as the dispersed phase in the form of droplets or ligaments [1]. The coupling between the two phases occurs through the exchanges of mass, momentum, and energy involving a wide range of thermal, mass, and fluid dynamic factors. A number of finite-difference formulations have been advanced over the years for predicting the flow (mass and momentum) and thermal properties of a rapidly vaporizing spray. Some of the pros and cons of various formulations can be found in [1-5].

The state of the art in multi-dimensional combustor modeling, as evidenced by the level of sophistication employed in terms of modeling and numerical accuracy considerations, is also dictated by the available computer memory and turnaround times afforded by present-day computers. With the aim of advancing the current multi-dimensional computational tools used in the design of advanced technology combustors, a solution procedure is developed that combines the novelty of the coupled CFD/spray/scalar Monte Carlo PDF (Probability Density Function) computations on unstructured grids with the ability to run on parallel architectures. In this approach, the mean gas-phase velocity and turbulence fields are determined from the solution of a conventional CFD method, the scalar fields of species and enthalpy from a modeled PDF transport equation using a Monte Carlo method, and a Lagrangian-based dilute spray model is used for the liquid-phase representation.

The gas-turbine combustor flows are often characterized by a complex interaction between various rate-controlling processes associated with turbulent transport, mixing, chemical kinetics, evaporation and spreading rates of spray, convective and radiative heat transfer, among others [10]. The phenomena to be modeled as controlled by these processes often interact with each other at various disparate time and length scales. In particular, turbulence plays an important role in determining the rates of mass and heat transfer, chemical reactions, and liquid-phase evaporation in many practical combustion devices. The influence of turbulence in a diffusion flame manifests itself in several forms, ranging from the so-called wrinkled or stretched flamelets regime to the distributed combustion regime, depending upon how turbulence interacts with various flame scales [69-70].

Most of the turbulence closure models for reactive flows have difficulty in treating nonlinear reaction rates [69-70]. The use of assumed shape PDF methods was found to provide reasonable predictions of pattern factors and NO_X emissions at the combustor exit [71]. However, their extension to multi-scalar chemistry becomes quite intractable. The solution procedure based on the modeled joint-composition PDF transport equation has an advantage in that it treats the nonlinear reaction rates without any approximation. This approach holds the promise of modeling various important combustion phenomena relevant to practical combustion devices such as flame extinction and blow-off limits, and unburnt hydrocarbons (UHC), CO, and NO_X predictions [71].

With the aim of demonstrating the viability of the PDF approach to the modeling of practical combustion flows, we have undertaken the task of extending this technique to the modeling of sprays as a part of the NCC (National Combustion Code) development program [7-15]. In order to facilitate large-scale combustor computations, we have extended our previous work on the combined CFD/spray/PDF computations to parallel computing [10-15]. The use of parallel computing offers enormous computational power and memory as it can make use of hundreds of processors in concert to solve a complex problem. The trend towards parallel computing is driven by two major developments: the widespread use of distributed computing and the recent advancements in MPPs (Massively Parallel Processors). The solver is designed to be massively parallel and automatically scales with the number of available processors. A current status of the use of the parallel computing in turbulent reacting flows involving sprays, scalar Monte Carlo PDF, and unstructured grids was described in Ref. [11]. It outlined several numerical techniques developed for overcoming some of the high computer CPU-time and memory-storage requirements associated with the use of Monte Carlo solution methods. The parallel performance of both the PDF and CFD modules was found to be excellent but the results were mixed for the spray computations showing reasonable performance on massively parallel computers like Cray T3D; but its performance was poor on the workstation clusters [10]. In order to improve the parallel performance of the spray module, two different domain decomposition strategies were developed and the results from both strategies were summarized

[10-14].

It is also well known that considerable effort usually goes into gridding up of complex gas-turbine combustor geometries. In order to allow representation of complex geometries with relative ease, we have extended our previous work on the combined CFD/spray/PDF computations to unstructured meshes [11-15]. The grid generation time associated with gridding up practical combustor geometries, which tend to be very complex in shape and configuration, could be reduced considerably by making use of existing unstructured grid generators. The solver accommodates the use of an unstructured mesh with mixed elements: triangular and/or quadrilateral for 2D (two-dimensional) geometries and tetrahedral for 3D.

With the development of two computer modules, LSPRAY - a Lagrangian spray solver [14] and EUPDF - an Eulerian Monte Carlo PDF solver [15], we were able to demonstrate the use of the joint scalar Monte Carlo PDF method in the modeling of complex multidimensional reacting flows (e.g., gas-turbine combustor flows) [10-15]. In this manual, we only concentrate on providing the details of our spray module. However, further details on the application of the joint scalar Monte Carlo PDF method to two-phase flows could be found elsewhere in [10-13,15].

In this manual, we summarize some important aspects of our spray formulation without making any attempt to provide an in-depth review on the fluid dynamic and transport behavior of reacting sprays [6-15]. Depending on the nature of the spray, an appropriate selection could be made from the choice of various well-known spray formulations (multicontinua, discrete-particle, or probabilistic) based on either a Lagrangian or an Eulerian representation for the liquid-phase equations by making use of appropriate droplet sub-grid models. The present solution procedure could be used within the context of both multicontinua and probabilistic spray formulations, as it allows for resolution on a scale greater than the average spacing between two neighboring droplets [1]. For NCC, the adopted choice for the gas phase was an Eulerian scheme. The liquid-phase equations form a system of hyperbolic equations and they could be solved by means of either an Eulerian or a Lagrangian representation. A Lagrangian scheme is chosen as it reduces the errors associated with numerical diffusion. The liquid-phase formulation is based on various well-established models for droplet drag; the vaporization models of a polydisperse spray take into account the transient effects associated with the droplet internal heating and the forced convection effects associated with droplet internal circulation; and it employs models for gas-film valid over a wide range of low to intermediate droplet Reynolds numbers [7]. Our current formulation is applicable for flows with a dilute spray approximation where the droplet loading is low. The numerical method could be used within the context of both steady and unsteady calculations [8-13]. Not considered in the present release of the code are the effects associated with droplet/shock interaction and dense spray effects.

Currently, most of the finite-difference techniques used for predicting the two-phase flows make use of the physics derived from single-component liquid droplet studies with constant properties. However, it is well known that most of the gas-turbine fuels are multicomponent mixtures of many compounds with a wide distillation curve [7,18]. The multicomponent nature of the liquid sprays is becoming evident with the increasing need to use jet fuels derived from heavier petroleum compounds. The gasification behavior of a multicomponent fuel droplet may differ significantly over that of a pure single component fuel droplet [18]. Also, the calculation of the variable thermo-transport properties of the liquid-mixtures becomes more important at high pressures. The flame ignition characteristics (such as the phenomena associated with flame blow-off and extinction conditions) could also be influenced by the nonuniform concentration of the fuels with different volatilities. However, the importance of the multicomponent liquid fuels with variable properties received little attention in the modeling of comprehensive gas-turbine combustor spray computations. With this in mind, we have extended the spray formulation to multicomponent liquid sprays in order to deal with the gas turbine fuels that are mixtures of many compounds. This implementation also takes into account the effect of variable liquid properties.

In order to reduce some uncertainty associated with the specification of initial droplet conditions, we have undertaken the task of integrating an atomization module into our spray solution procedure. The atomization module was developed by CFDRC Inc. [47] in collaboration with the university of Wisconsin (UW) [45-46]. Our computational experience with the assessment of various atomization models was summarized in [59-60,78]. A complete description of all the primary atomization as well as the secondary droplet breakup models contained in the

CFDRC/UW atomization module is provided in the appendix.

An understanding of droplet behavior under superheat conditions is identified to be a topic of importance in the design of modern supersonic engines, scramjet, and ramjet afterburners [61-64, & 74]. It is because (1) higher heat-load requirements may lead to the use of the same fuel as a coolant, (2) under some engine operating conditions the nozzles are required to operate under low backpressures, and (3) under normal gas-turbine operating conditions, it is estimated that a small fraction of the liquid fuel may be released by flash boiling [62]. There are some reported incidences of engine damage due to flash atomization in gasoline direct-injection internal combustion engines [62]. Because of its importance in some NASA-supported future engine development projects, we have undertaken an assessment study to establish the baseline accuracy of various existing vaporization and atomization models used in the calculation of a superheated spray. As a part of this effort, we have incorporated a solution procedure based on the modeling approach developed by [62-64, & 74], Our computational experience with the assessment of this modeling approach is summarized in [78-79].

There is a need to understand the droplet behavior under supercritical pressures as some modern gas-turbine combustors tend to operate under increasingly high pressures [19-20, & 81-86]. Based on some important aspects of the modeling approach derived from [20, & 83-86], we have implemented a high-pressure droplet vaporization model into our spray code. As a part of this effort, we have also implemented the Peng-Robinson equation of state following the approach of [16-17, & 28], the high-pressure corrections to the gas-phase transport properties from [16, 19, & 29-31], and the high-pressure corrections to the liquid-phase transport properties from [16-18].

One important aspect fo spray modeling is how to deal with the collision dynamics of a droplet impact with the wall. Upon its impact with the wall, a droplet may shatter, rebound, or stick to the wall. In order to deal with several possible outcomes, we have incorporated the modelng approach of [80-81] based on the coding partially received from CFDRC.

Some other important aspects of the spray module are summarized below:

- An efficient particle tracking algorithm was developed and implemented into the Lagrangian spray solver in order to facilitate particle movement in an unstructured grid of mixed elements.

- LSPRAY-V is currently coupled with an unstructured flow (CFD) solver of NCC [21-23], and an Eulerian-based Monte Carlo probability density function solver - EUPDF [15], which were selected to be as the integral components of the NCC cluster of modules. EUPDF provides the solution for the scalar fields of species and enthalpy from a modeled PDF transport equation using a Monte Carlo method.

- The spray solver receives the mean velocity and turbulence fields from the flow solver. The solution for the scalar (energy and species) fields could be provided by means of either a conventional CFD solver or a Monte Carlo PDF solver depending on the choice of the solver.

- The spray solver supplies the spray source-term contributions arising from the exchanges of mass, momentum and energy with the liquid-phase to the flow solvers (CFD and/or Monte Carlo PDF). This information could be used in either conservative or non-conservative finite-difference formulations of the gas phase equations.

The furnished code demonstrates the successful methods used for parallelization and coupling of the spray to the flow code. First, complete details of the spray solution procedure is presented along with several other numerical issues related to the coupling between the CFD, LSPRAY-V, and EUPDF solvers. It is followed by a brief description of the combined parallel performance of the three solvers (CFD, EUPDF, and LSPRAY-V) along with a brief summary of the validation cases.

2 GOVERNING EQUATIONS FOR GAS PHASE

Here, we summarize the governing gas-phase conservation equations in Eulerian coordinates [1]. This is done for the purpose of identifying the interphase source terms arising from the exchanges of mass, momentum, and energy with the liquid phase. They are valid for a dilute spray with a void fraction of the gas, θ, close to unity. The void fraction is defined as the ratio of the equivalent volume of gas to a given volume of a gas and liquid mixture.

The conservation of the mass leads to:

$$[\bar{\rho}V_c]_{,t} + [\bar{\rho}V_c u_i]_{,x_i} = s_{mlc} = \sum_k n_k \, \dot{m}_k \tag{1}$$

The source term is given as a summation over different classes of droplets. Each class represents the average properties of a different polydisperse group of droplets. Here, n_k represents the number of droplets in a given class and $\dot{m}_k$ represents the corresponding mass vaporization rate.

For the conservation of the jth species, we have:

$$[\bar{\rho}V_c y_j]_{,t} + [\bar{\rho}V_c u_i y_j]_{,x_i} - [\bar{\rho}V_c D y_{j,x_i}]_{,x_i} - \bar{\rho}V_c \dot{w}_j =$$

$$s_{mls} = \sum_k \epsilon_j \, n_k \, \dot{m}_k \tag{2}$$

where

$$\sum_j \dot{w}_j = 0 \; and \; \sum_j \epsilon_j = 1$$

For the species conservation, the source term contains an additional variable, ϵ_j, which is defined as the fractional vaporization rate for species j.

For the momentum conservation, we have:

$$[\bar{\rho}V_c u_i]_{,t} + [\bar{\rho}V_c u_i u_j]_{,x_j} + [pV_c]_{,x_i} -$$

$$[\theta V_c \tau_{ij}]_{,x_j} - [(1-\theta)V_c \tau_{lij}]_{,x_j} = s_{mlm} =$$

$$\sum_k n_k \, \dot{m}_k \, u_{ki} - \sum_k \frac{4\pi}{3} \, \rho_k \, r_k^3 \, n_k \, u_{ki,t} \tag{3}$$

where the shear stress τ_{ij} in Eq. (3) is given by:

$$\tau_{ij} = \mu[u_{i,x_j} + u_{j,x_i}] - \frac{2}{3}\mu\delta_{ij}u_{i,x_j}$$

For the momentum conservation, the first source term represents the momentum associated with liquid fuel vapor and the second represents the momentum change associated with droplet drag.

For the energy conservation, we have:

$$[\bar{\rho}V_c h]_{,t} + [\bar{\rho}V_c u_i h]_{,x_i} - [\theta V_c \lambda T_{,x_i}]_{,x_i} - [(1-\theta)V_c \lambda_l T_{,x_i}]_{,x_i}$$

$$-[\theta V_c p]_{,t} = s_{mle} = \sum_k n_k \, \dot{m}_k \Big(h_s - l_{k,eff}\Big) \tag{4}$$

Similarly, the energy conservation has a source term contribution from the added liquid fuel vapor and an additional contribution associated with the effective latent heat of vaporization which accounts for the heat flux (loss or gain) to the droplet interior from the ambient.

The main purpose of the spray solver is to calculate the source terms arising from the exchanges of the mass, momentum, and energy. In the case of NCC, it provides the calculated source terms to both the CFD and Monte Carlo PDF solvers.

3 HIGH PRESSURE EQUATION OF STATE

In order to calculate the high pressure gas behavior, the Peng-Robinson EOS (Equation-Of-State) is employed for a multi-component mixture in the following form [16-17,28]:

$$P = \frac{RT}{V - b_m} - \frac{a_m}{V^2 + 2b_m V - b_m^2} \tag{5}$$

where

$a_m = \sum_i \sum_j y_i y_j (a_i a_j)^{1/2}(1 - k_{ij})$,

$b_m = \sum_i y_i b_i$,

$b_i = \frac{0.07780 R T_{ic}}{P_{ic}}$,

$a_i = \frac{0.45724 R^2 T_{ic}^2}{P_{ic}}[1 + f_{i\omega}(1 - T_{ir}^{1/2})]^2$,

$T_{ir} = T/T_{ic}$,

$f_{i\omega} = 0.37464 + 1.54226\omega_i - 0.26992{\omega_i}^2$,

ω_i is known as the acentric factor of the molecules which is a measure of non-sphericity of the molecules, and k_{ij} is known as the binary interaction coefficient. However, the Peng-Robinson EOS is rewritten as a cubic EOS in terms of the compressibility factor, Z $(= \frac{PV}{RT})$, before it is solved:

$$Z^3 - (1 - B^\star)Z^2 + (A^\star - 2B^\star - 3B^{\star 2})Z$$

$$-A^\star B^\star + B^{\star 2} + B^{\star 3} = 0 \tag{6}$$

where

$A^\star = \frac{a_m P}{R^2 T^2}$, and $B^\star = \frac{b_m P}{RT}$.

Table 1. Physical constants.

Species	T_{nb}	T_c	P_c	ρ_c	l_{kn}	V_c	V_n	ω	σ	σ / κ
	(K)	(K)	(atm)	(Kg/m^3)	(KJ/kg)	(cm^3/g-mole)	(cm^3/g-mole)		(A ^{o})	(K)
C_6H_{14}	341.9	507.4	30.0	660.0	334.8	370.0	140.06	0.296	5.949	399.3
C_7H_{16}	371.6	540.2	27.0	682.0	316.3	432.0	162.00	0.351	6.297	419.031
C_8H_{18}	398.82	568.8	24.6	718.5	301.3	492.0	188.8	0.394	6.62	488.15
$C_{10}H_{22}$	447.3	617.6	20.8	728.3	276.1	603.0	233.68	0.490	7.16	540.06
$C_{12}H_{26}$	489.5	658.3	18.0	748.0	256.3	713.0	278.54	0.562	7.655	583.68
$C_{14}H_{30}$	526.7	694.0	16.0	763.0	240.1	830.0	326.62	0.679	8.067	629.08
N_2	77.4	126.2	33.9	807.1	197.6	90.1	31.87	0.039	3.681	91.5
O_2	90.2	154.6	50.4	1135.7	212.3	74.4	26.08	0.025	3.433	113.0
CO_2	00.0	304.1	73.8	000.0	000.0	94.0	33.32	0.239	3.996	190.0
H_2O	373.2	647.3	221.2	958.1	2257.2	57.1	19.76	0.344	2.641	809.1

Table 2. Binary Interaction Parameters, $\bar{k}_{ij}$.

	C_7H_{16} (j=1)	O_2 (j=2)	N_2 (j=3)	CO$_2$ (j=4)	H_2O (j=5)
C_7H_{16} (i=1)	0.0000	0.1321	0.1440	0.1000	0.1484
O_2 (i=2)	0.1321	0.0000	-0.0119	-0.0289	0.0910
N_2 (i=3)	0.1440	-0.0119	0.0000	-0.0170	0.1030
CO_2 (i=4)	0.1000	-0.0289	-0.0170	0.0000	0.1200
H_2O (i=5)	0.1484	0.0910	0.1030	0.1200	0.0000

We have chosen the Peng-Robinson EOS because of its simplicity and, more importantly, it proved to be very useful in the supercritical droplet vaporization studies of [19-20].

Table 1 lists various physical constants for some of the species that we found to be of interest in our spray computations. It contains values for the boiling temperature at normal pressure, the critical temperature, the critical pressure, the critical density, the latent heat of vaporization at normal pressure, the critical molar volume, the molar volume at normal pressure, and the acentric factor of the molecules. Most of this data is collected from [16-18] and is useful in both the evaluation of the Peng-Robinson EOS and the calculation of various other variable properties. And Table 2 lists the binary interaction parameters, k_{ij}, used in a calculation for the multi-component mixture of n-heptane, O_2, N_2, CO_2, & H_2O. While the data for most of the binary pairs was obtained from various reference books, some of the missing data, however, is replaced with known data found for other binary pairs of molecules with similar molecular weights.

4 HIGH PRESSURE CORRECTIONS FOR GAS-PHASE TRANSPORT PROPERTIES

The effect of pressure on the viscosity of pure gases is determined by making use of the Reichenberg method [16,19,29]:

$$\frac{\mu}{\mu_n} = 1 \; + \; Q_v \; \frac{A_v P_r^{3/2}}{B_v P_r \; + \; (1 + C_v P_r^{D_v})^{-1}} \qquad (7)$$

where

$$A_v = \frac{\alpha_1}{T_r} exp(\alpha_2 T_r^a) \qquad B_v = A_v(\beta_1 T_r - \beta_2)$$
$$C_v = \frac{\gamma_1}{T_r} exp(\gamma_2 T_r^c) \qquad D_v = \frac{\lambda_1}{T_r} exp(\lambda_2 T_r^d)$$

$\alpha_1 = 1.982510^{-03}$	$\alpha_2 = 5.2683$	$a = -0.5767$
$\beta_1 = 1.6552$	$\beta_2 = 1.2760$	
$\gamma_1 = 0.1319$	$\gamma_2 = 3.7035$	$c = -79.8678$
$\lambda_1 = 2.9496$	$\lambda_2 = 2.9190$	$d = -16.6169$

and $Q_v = 1.0$ for non-polar molecules.

The effect of pressure on the thermal conductivity of pure gases is determined by making use of the Stiel & Thodos method [16,19,30]:

$$(\lambda - \lambda_n)\Gamma Z_c^5 = 1.2210^{-2}[exp(0.535\rho_r) - 1] \qquad (8)$$

when $\rho_r < 0.5$

$$(\lambda - \lambda_n)\Gamma Z_c^5 = 1.1410^{-2}[exp(0.67\rho_r) - 1.069] \qquad (9)$$

when $0.5 < \rho_r < 2.0$

$$(\lambda - \lambda_n)\Gamma Z_c^5 = 2.6010^{-2}[exp(1.155\rho_r) + 2.016] \qquad (10)$$

when $2.0 < \rho_r < 2.8$

where λ is in W/(m.K), $\Gamma = 210\left[\frac{T_c M^3}{P_c^4}\right]^{1/6}$, Z_c is the critical compressibility, and ρ_r is the reduced density $\rho/\rho_c = V_c/V$.

The effect of pressure & temperature on diffusion coefficients in gases is determined by making use of Takahashi correlation [16,19,31]:

$$\frac{D_{AB}P}{(D_{AB}P)^+} \; = \; f(T_r, P_r) \qquad (11)$$

where D_{AB} = diffusion coefficient, cm^2/s, P = pressure, bar, the superscript $^+$ indicates that low-pressure values are to be used, and the reduced pressure and temperature in the above equation are calculated as follows:

$$\begin{array}{lcl} T_r & = & \frac{T}{T_{cAB}} \\ T_{cAB} & = & y_A T_{cA} \; + \; y_B T_{cB} \\ P_r & = & \frac{P}{P_{cAB}} \\ P_{cAB} & = & y_A P_{cA} \; + \; y_B P_{cB} \end{array}$$

This correlation is valid for a system involving a trace solute diffusing in a supercritical fluid. It is shown to provide satisfactory results but it is based on a very limited set of available experimental data. The graphical representation of the Takahashi function is given in Fig. 1.

It is noteworthy that at low pressures, the polynomial fits for the variable thermodynamic properties are taken from the data set compiled by McBride et el [24]. The transport properties involving the thermal conductivity and molecular viscosity for individual species is estimated based on the Chapman-Enskog collision theory [25-27]. And Wilke's formulae is used to determine the properties of mixture [25-27]. The binary-diffusion coefficients are determined based on the Chapman-Enskog theory and the Lennard-Jones potential [25-27].

5 LIQUID-PHASE EQUATIONS

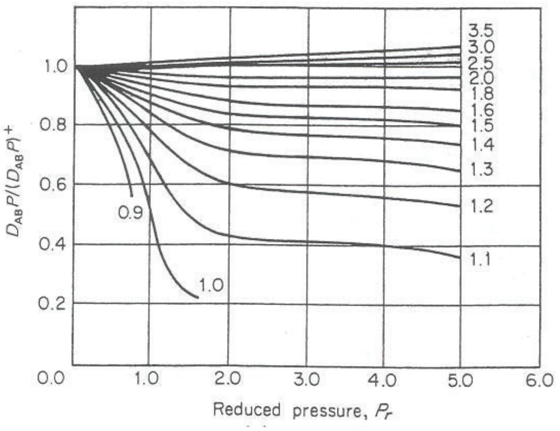


Figure 1 Takahashi correlation showing the variation of the binary diffusion coefficient versus reduced pressure at different reduced temperatures.

Here, we summarize the governing equations for the liquid-phase based on a Lagrangian formulation where the equations for particle position and velocity are described by a set of ordinary differential equations. For the particle position of the kth droplet group, we have:

$$\frac{dx_{ik}}{dt} = u_{ik} \tag{12}$$

For the droplet velocity:

$$\frac{du_{ik}}{dt} = \frac{3}{16}\frac{C_D \mu_{gs} Re_k}{\rho_k r_k^2}\left[u_{ig} + u'_g - u_{ik}\right] \tag{13}$$

where

$$C_D = \frac{24}{Re_k}\left(1 + \frac{Re_k^{2/3}}{6}\right) \tag{14}$$

According to Yuen and Chen [32], the droplet drag for a vaporizing droplet could be calculated by a solid-sphere drag correlation but suggested using a correction in the evaluation of the droplet Reynolds number for the average viscosity based on a 1/3 weighting rule as given by Eq. (46). The droplet Reynolds number is given by:

$$Re_k = 2\frac{r_k \rho_g}{\mu_{gs}}\left[\left(u_g + u'_g - u_k\right).\left(u_g + u'_g - u_k\right)\right]^{1/2} \tag{15}$$

By following the approach taken from KIVA [33], a gas turbulence velocity, u'_g, is added to the local mean gas velocity when calculating the droplet drag and vaporization rates. The gas velocity fluctuations is calculated by randomly sampling a Gaussian distribution with mean square deviation, $2/3k$. The Gaussian is given by

$$G(u'_g) \;=\; (4/3\pi k)^{-3/2} exp[-3|u'_g|^2/4k] \tag{16}$$

The gas fluctuating component is calculated once at every turbulence interaction time, t_{tur}, and is otherwise held constant [33]. The correlation time is taken to be the minimum of either the eddy time or the transit time taken by the droplet to traverse the eddy. It is given by

$$t_{tur} = min\Big(\frac{k}{\epsilon}, c_{tt}\frac{k^{3/2}}{\epsilon}\frac{1}{|u_g + u'_g - u_k|}\Big) \tag{17}$$

where c_{tt} is an empirical constant with a value of 0.16432.

The liquid mixture density, ρ_k, in Eq. (13) is given by [16,18]:

$$\rho_k = \sum_i y_{ki}\rho_{ki} \tag{18}$$

and the individual component liquid density is given by:

$$\rho_{ki} = \frac{M_{ki}}{V_{ki}} \tag{19}$$

where the molar volume, V_{ki}, is

$$V_{ki} = V_{ci}(0.29056 - 0.8775\omega_i)^{c_v} \tag{20}$$

and

$$c_v = \left[1 - \frac{T_k}{T_{ci}}\right]^{\frac{2}{7}}$$

The droplet regression rate is determined from one of three different correlations depending upon the droplet Reynolds number range. The first correlation as given by Eq. (21) is based on a gas-film analysis developed by Tong & Sirignano [34]. It is based on a a combination of stagnation and flat-plate boundary-layer analysis and is valid for Reynolds numbers in the intermediate range. The last two correlations as given by Eqs. (22) & (23) are taken from Clift et al [35] and are valid when $Re_k \leq 20$. In fact, when the droplet Reynolds number goes to zero, Eq. (23) becomes identical to the droplet regression rate of a vaporizing droplet in a quiescent medium [27].

$$\frac{ds_k}{dt} = -2\frac{\rho_g D_g}{\rho_k}\left[\frac{2}{\pi}Re_k\right]^{1/2} f(B_k) \tag{21}$$

$if\ Re_k > 20$

$$\frac{ds_k}{dt} = -\frac{\rho_g D_g}{\rho_k}\left[1+(1+Re_k)^{1/3}\right] Re_k^{0.077} ln(1+B_k) \tag{22}$$

$if\ 1 < Re_k \leq 20$

$$\frac{ds_k}{dt} = -\frac{\rho_g D_g}{\rho_k}\left[1+(1+Re_k)^{1/3}\right] ln(1+B_k) \tag{23}$$

$if\ \ Re_k < 1$

where B_k is the Spalding mass transfer number and is given by:

$$B_k = \frac{(y_{fs} - y_f)}{(1-y_{fs})} \tag{24}$$

and y_{fs} is given as a summation over the fuel-species mass fractions at the droplet interface:

$$y_{fs} = \sum_i y_{is} \tag{25}$$

and y_f is given as a summation over the fuel-species mass fractions in the ambient:

$$y_f = \sum_i y_{fi} \tag{26}$$

and the function $f(B_k)$ is similar to that of a Blassius function [1, 36] and is obtained from an analysis similar to that of Emmon's boundary-layer flow over a flat plate with blowing [36]. The range of validity of this function was extended in Raju and Sirignano [7] to consider the effects associated with a boundary-layer flow with suction.

The internal droplet temperature is determined based on a vortex model [34]. The governing equation for the internal droplet temperature is given by:

$$\frac{\partial T_k}{\partial t} = 17\frac{\lambda_l}{C_{pl}\rho_l r_k^2}\left[\alpha\frac{\partial^2 T_k}{\partial\alpha^2} + (1+C(t)\alpha)\frac{\partial T_k}{\partial\alpha}\right] \tag{27}$$

where

$$C(t) = \frac{3}{17}\left[\frac{C_{pl}\rho_l}{\lambda_l}\right] r_k\frac{dr_k}{dt} \tag{28}$$

where α represents the coordinate normal to the stream-surface of a Hill's Vortex in the circulating fluid, and $C(t)$ represents a nondimensional form of the droplet regression rate. The initial and boundary conditions for Eq. (27) are given by:

$$t = t_{injection}, \quad T_k = T_{k,o} \tag{29}$$

$$\alpha = 0, \quad \frac{\partial T_k}{\partial\alpha} = \frac{1}{17}\left[\frac{C_{pl}\rho_l}{\lambda_l}\right] r_k^2\frac{\partial T_k}{\partial t} \tag{30}$$

$$\alpha = 1, \quad \frac{\partial T_k}{\partial\alpha} = -\frac{3}{32}\frac{\rho_k}{\lambda_l}\left[l_{k,eff} - l_k\right]\frac{ds_k}{dt} \tag{31}$$

where $\alpha = 0$ refers to the vortex center, and $\alpha = 1$ refers to the droplet surface, and the mixture latent heat of vaporization, l_k, is given by

$$l_k = \sum_i \epsilon_i l_{ki} \tag{32}$$

and the individual component latent heat of vaporization, l_{ki}, is given by [16,18]:

$$l_{ki} \ = \ l_{kin}\Big(\frac{T_{ci} - T_k}{T_{ci} - T_{bi}}\Big)^{0.38} \tag{33}$$

and the droplet boiling temperature is given by

$$T_{bi} = \frac{l_{kin}M_i/R_u}{l_{kin}M_i/(R_u t_{bni}) - ln(P)} \tag{34}$$

and, finally, the effective latent heat of vaporization, $l_{k,eff}$, is defined as:

$$l_{k,eff} = l_k + 4\pi\frac{\lambda_l r_k^2}{\dot{m}_k}\left(\frac{\partial T_k}{\partial r}\right)_s \tag{35}$$

It is a very useful parameter as it represents the total energy loss associated with the latent heat of vaporization in addition to the the heat loss to the droplet interior. $l_{k,eff}$ is calculated by means of the following relationship [18]:

$$l_{k,eff} = \frac{C_p(T_g - T_{ks})}{(1+B_k)^{1/Le} - 1} \tag{36}$$

Similar to the internal thermal transport, the internal mass transport of a multi-component fuel is given by:

$$\frac{\partial y_{ki}}{\partial t} = 17\frac{D_k}{r_k^2}\left[\alpha\frac{\partial^2 y_{ki}}{\partial \alpha^2} + (1 + C(t)\alpha)\,\frac{\partial y_{ki}}{\partial \alpha}\right] \tag{37}$$

The initial and boundary conditions for Eq. (37) are given by:

$$t = t_{injection}, \quad y_{ki} = y_{ki,o} \tag{38}$$

$$\alpha = 0, \quad \frac{\partial y_{ki}}{\partial \alpha} = \frac{1}{17}\left[\frac{r_k^2}{D_k}\right]\frac{\partial y_{ki}}{\partial t} \tag{39}$$

$$\alpha = 1, \quad \frac{\partial y_{ki}}{\partial \alpha} = -\frac{3}{32}\frac{1}{D_k}\left[y_{kis} - \epsilon_i\right]\frac{ds_k}{dt} \tag{40}$$

By knowing the mass fractions of the liquid species at the droplet surface, the corresponding mole fractions are determined by

$$x_{iks} = \frac{y_{iks}/M_i}{\sum_i y_{yks}/M_i} \tag{41}$$

At the droplet interface, the mole fractions of the gas species are obtained by means of Raoult's law:

$$x_{is} = \frac{1}{P}x_{iks}P_{is} \tag{42}$$

where the partial pressure, P_{is}, is determined by means of the Clausius-Clapeyron relationship:

$$P_{is} = exp\left[\frac{l_{ki}}{R_u}\left(\frac{1}{T_{bi}} - \frac{1}{T_{ks}}\right)\right] \tag{43}$$

Then, the corresponding mass fractions in the gas-phase at the droplet interface are given by

$$y_{is} = \frac{x_{is}M_i}{M_a(1 - \sum_i x_{is}) + \sum_i M_i x_{is}} \tag{44}$$

where M_a is the molecular weight of the gas excluding fuel vapor. And the fractional mass vaporization rate of liquid species, ϵ_i, is given by

$$\epsilon_i = y_{is} + (1 - y_{fs})\frac{y_{is} - y_{fi}}{y_{fs} - y_f} \tag{45}$$

It is noteworthy that the thermodynamic and transport properties at the gas film are calculated at the temperature and composition as determined by the following one-third rule:

$$\phi_{avg} = \frac{1}{3}\phi_g + \frac{2}{3}\phi_{ks} \tag{46}$$

The correlations for the gas-phase thermodynamic and transport properties are described in Section 4. In a similar way, the liquid-phase thermodynamic and transport properties are determined based on the correlations described in the next section.

6 LOW PRESSURE LIQUID THERMODYNAMIC & TRANSPORT PROPERTIES

The specific heat at constant pressure, C_{pl}, thermal conductivity, λ_l, and viscosity, μ_l, are evaluated by means of the following expressions:

$$C_{pl} = C_{pl0} + C_{pl1}T + C_{pl2}T^2 + C_{pl3}T^3 + C_{pl4}T^4 \tag{47}$$

$$\lambda_l = \lambda_{l0} + \lambda_{l1}T + \lambda_{l2}T^2 + \lambda_{l3}T^3 + \lambda_{l4}T^4 + \lambda_{l5}T^5 \tag{48}$$

$$ln\ \mu_l = \mu_{l0} + \mu_{l1}/T + \mu_{l2}T + \mu_{l3}T^2 \tag{49}$$

where C_{pl} is in J/(kg K), μ_l in (μPA s), and λ_l in (W/m K).

Tables 3-5 provide the polynomial constants used in Eqs. (47)-(49) for some of the species that we found to be of interest in our spray computations. Table 3 provides the constants for the liquid specific heat, C_{pl}, Table 4 for the liquid thermal conductivity, λ_l, and Table 5 for the liquid molecular viscosity, μ_l. These tables are compiled with the data taken mostly from the references of [16,18].

The binary diffusion coefficient, D_{ij}, is evaluated as follows [16]:

$$D_{ij} = \frac{K_{dif}T}{\mu_j V_i^{1/3}} \tag{50}$$

where $K_{dif} = 8.210^{-8}(1 + [\frac{3V_j}{V_i}]^{2/3})$. One should be careful in using this approximation as it is based on a scarce set of available experimental data.

The specific heat for a multicomponent mixture is given by:

$$C_{pm} = \sum_{i=1}^{n} y_i C_{pi} \tag{51}$$

Table 3. Polynomial constants for liquid specific heat.

Fuel	c_{pl0}	c_{pl1}	c_{pl2}	c_{pl3}	c_{pl4}
C_6H_{14}	2.4169	-5.9866e-03	2.0959e-05	-8.4489e-09	0.0
C_7H_{16}	4.8227	-3.6980e-02	1.6777e-04	-3.0987e-07	2.2081e-10
C_8H_{18}	9.2189	-8.8314e-02	3.8869e-04	-7.2539e-07	5.0776e-10
$C_{10}H_{22}$	4.7991	-2.8643e-02	9.3619e-05	-8.9516e-08	0.0
$C_{12}H_{26}$	4.7900	-2.8643e-02	9.3619e-05	-8.9516e-08	0.0
$C_{14}H_{30}$	4.7991	-2.8643e-02	9.3619e-05	-8.9516e-08	0.0

Table 4. Polynomial constants for liquid thermal conductivity.

Fuel	λ_{l0}	λ_{l1}	λ_{l2}	λ_{l3}	λ_{l4}	λ_{l5}
C_6H_{14}	0.37078	-5.4313e-03	4.628e-05	-1.8002e-07	3.2243e-10	-2.1832e-13
C_7H_{16}	0.13236	9.4441e-04	-6.588d-06	1.4617e-08	-1.1244e-11	0.0
C_8H_{18}	0.25652	-7.5401e-04	1.5872e-06	-1.6795e-09	-1.3375e-16	0.0
$C_{10}H_{22}$	0.22179	-2.3699e-04	-6.94e-07	2.0415e-09	-1.5741e-12	0.0
$C_{12}H_{26}$	0.17609	4.2463e-05	-7.4467e-07	6.9446e-10	0.0	0.0
$C_{14}H_{30}$	0.18801	-9.1399e-05	-2.1464e-07	1.1655e-10	0.0	0.0
N_2	-2.629E-1	-1.545E-3	-9.450E-7	0.0	0.0	0.0
O_2	2.444E-1	-8.813E-4	-2.023E-6	0.0	0.0	0.0
CO_2	4.070E-1	-8.438E-4	-9.626E-7	0.0	0.0	0.0
H_2O	-3.838E-1	5.254E-3	-6.369E-6	0.0	0.0	0.0

Table 5. Polynomial constants for liquid molecular viscosity.

Fuel	μ_{l0}	μ_{l1}	μ_{l2}	μ_{l3}
C_6H_{14}	-4.034E+00	8.354E+02	0.0000000	0.0000000
C_7H_{16}	-4.325E+00	1.006E+03	0.0000000	0.0000000
C_8H_{18}	-4.333E+00	1.091E+03	0.0000000	0.0000000
$C_{10}H_{22}$	-4.460E+00	1.286E+03	0.0000000	0.0000000
$C_{12}H_{26}$	-4.562E+00	1.454E+03	0.0000000	0.0000000
$C_{14}H_{30}$	-4.615E+00	1.588E+03	0.0000000	0.0000000
N_2	-2.795E+01	8.660E+02	2.763E-01	-1.084E-03
O_2	-4.771E+00	2.146E+02	1.389E+02	-6.255E-05
CO_2	-3.097E+00	4.886E+01	2.381E-02	-7.840E-05
H_2O	-2.471E+01	4.209E+03	4.527E-02	-3.376E-05

And the thermal conductivity of a multicomponent mixture is calculated by means of the Li method [16].

$$\lambda_m = \sum_{i=1}^{n}\sum_{j=1}^{n} \phi_i \phi_j \lambda_{ij} \tag{52}$$

where

$$\begin{array}{lcl} \lambda_{ij} & = & 2(\lambda_i^{-1} + \lambda_j^{-1})^{-1} \\ \phi_i & = & \frac{x_i V_i}{\sum_{j=1}^{n} x_j V_j} \end{array}$$

where x_i is the mole fraction of the species i, ϕ_i is a volume fraction of the ith species, and V_i is the molar volume of the pure fluid.

7 HIGH-PRESSURE LIQUID TRANSPORT PROPERTIES

The high-pressure correction for thermal conductivity is calculated by means of a correlation taken from Refs. [16-18].

$$W_\lambda = 1.0 + (P_r/31.6 + T_r^2(1.0 - exp(-0.2P_r)))/Q_\lambda, P_r < 12.0, \tag{53}$$

$$W_\lambda = 1.0 + (P_r^{0.7})(0.0667 + 0.1597T_r^2)/Q_\lambda, P_r > 12.0 \tag{54}$$

where $Q_\lambda = 8.6514 - 3.78T_r - T_r^2$.

The high pressure correction for viscosity is also calculated by means of a correlation taken from Refs. [16-18].

$$W_\mu = (1.0 + B_\mu(0.47214\Delta P_r)^{A_\mu})/(1.0 + C_\mu \omega \Delta P_r) \tag{55}$$

where

$$\begin{array}{lcl} \Delta P_r & = & (P - P_{sat})/P_c \\ A_\mu & = & 0.999 - 4.442E - 04/(T_r^{-0.03877} - 0.999) \\ B_\mu & = & 0.3257/(1.0039 - T_r^{2.573})^{0.2906} - 0.2086 \\ C_\mu & = & -0.0792 + 2.1616T_r - 13.404T_r^2 + 44.1706T_r^3 \\ & & -84.829T_r^4 + 96.1209T_r^5 - 59.8127T_r^6 + 15.6719T_r^7 \end{array}$$

It is noteworthy that in calculating the properties of a multi-component mixture that we have adopted the same kind of mixing rules as described in the previous section.

8 VAPORIZATION MODELING OF A SUPERHEATED DROPLET

Flashing phenomena refers to a process that is in thermodynamic non-equilibrium when a liquid is superheated [72-73]. The reasons for its occurrence are mainly two-fold [72-73]: (1) a liquid fuel can be heated to a temperature above its saturation temperature, and (2) when a liquid is depressurized rapidly it can lead to flashing as the thermal inertia tends to maintain its bulk internal temperature above the saturation temperature. Although flash evaporation is considered to be detrimental to engine performance under normal circumstances, it can have some potential benefits. It is known to produce a fine spray with enhanced atomization, increase effective spray cone angle, and decrease spray penetration [61].

An understanding of flash injection is of importance in some applications involving scramjet and ramjet afterburners because the same liquid fuel is often used as a coolant. Also, the engine conditions are such that nozzles operate at low back pressures with supersonic outflow [61]. The objective of our work is to establish a baseline accuracy for existing atomization and vaporization models used in the prediction of a superheated spray by undertaking a critical review of existing experimental data available in the literature for validation. This work is funded through the supersonics (SUP) and subsonic fixed wing (SFW) project office initiatives on high altitude emissions of the NASA's fundamental aeronautics program.

In what follows we first provide some details of the two superheat vaporization models that we have implemented into NCC. It is followed by other modeling considerations that need to be taken into account as described in Sections 8.3 & 8.4.

8.1 Superheat Vaporization Model of Zuo, Gomes, & Rutland [62], and Schmehl & Steelant [63-64]

It is based on an extension of the classical D^2-theory. In the classical evaporation model, the thermal energy needed for evaporation is mostly furnished by the external heat transfer from the surrounding gas. Under superheat conditions, the characteristic vaporization time resulting from the external heat transfer from the surrounding gas is of the same order of magnitude as that resulting from the flash evaporation. The energy needed for vaporization at the droplet surface is partly provided by the superheat energy stored within the droplet and it is controlled by the droplet internal heat transfer. The modeling approach differs from the classical droplet vaporization models in three important ways: (1) under superheat conditions, the droplet surface mass fraction,

Y_{fs}, approaches unity as the droplet surface temperature is maintained at the corresponding liquid boiling temperature; (2) under superheat conditions, all the external heat transfer from the surrounding gas is made available to the vaporization process with no apparent increase in the droplet surface temperature; and (3) the flow of fuel vapor imparted by flash vaporization partly counterbalances the flow generated by external heat transfer and may significantly reduce the energy transferred from the surrounding gas.

Based on the governing equations of conservation for an isolated spherically symmetric droplet, Zuo et al [62] and Schmehl and Steelant [63-64] showed that the total evaporation rate, $\dot{m}_k$, can be calculated as

$$\dot{m}_k \ = \ \dot{m}_{k,flash} \ + \ \dot{m}_{k,t} \qquad (56)$$

where the flash boiled vaporization rate, $\dot{m}_{k,flash}$, is given by

$$\dot{m}_{k,flash} \ = \ 4\pi r_k^2 \alpha_s \frac{(T_k - T_b)}{l_k} \qquad (57)$$

where T_k is the internal droplet temperature and the overall heat transfer coefficient, α_s $(= kJ/s \ m^2 \ {}^oK)$ is given by the Adachi correlation [65]:

$$\begin{array}{lll} & = \ 0.76(T_k - T_b)^{0.26} & (0 \leq T_{ks} - T_b \leq 5) \\ \alpha_s & = \ 0.027(T_k - T_b)^{2.33} & (5 \leq T_{ks} - T_b \leq 25) \\ & = \ 13.8(T_k - T_b)^{0.39} & (T_{ks} - T_b \geq 25) \end{array} \qquad (58)$$

The Adachi correlation is valid over a wide range of superheat conditions.

The vaporization rate due to external heat transfer, $\dot{m}_{k,t}$, in Eq. (56) is given by

$$\dot{m}_{k,t} = 2\pi r_k \frac{k}{C_p} \frac{Nu}{1 + \frac{\dot{m}_{k,flash}}{\dot{m}_{k,t}}} ln[1 + (1 + \frac{\dot{m}_{k,flash}}{\dot{m}_{k,t}})B_t] \qquad (59)$$

where the Spalding heat transfer number, B_t, and the Nusselt number, Nu, are given by

$$B_t \ = \ \frac{C_p(T_g - T_{ks})}{l_{k,eff}} \qquad (60)$$

$$Nu \ = \ 2(1 + 0.3Re^{1/2}Pr_g^{1/3}) \qquad (61)$$

The corresponding droplet regression rate, $\frac{ds_k}{dt}$, is given by

$$\frac{ds_k}{dt} = -\frac{\dot{m}_k}{2\pi r_k \rho_l} \qquad (62)$$

This model is valid over an entire range of superheat conditions as long as there is some amount of superheat energy available within the droplet ($T_k > T_b$).

8.2 Superheat Vaporization model of Lee et al. [74]

This modeling approach is also based on an extension of the classical D^2-theory but it differs from the previous model of [62, 63-64] in several important ways: (1) neglects the effects of internal nucleation and convection, (2) instead of invoking the Adachi correlation for the flash induced vaporization rate, the heat loss from the droplet interior is modeled by $\hat{q}_l \ = \ \frac{T_{interior} - T_s(P_\infty)}{r_{k,t=0}}$, (3) the formulation is valid only when $T_\infty \ \geq \ T_{bub}(P_\infty)$, which means for positive Spalding heat transfer numbers.

With the assumptions employed, Lee et al. [74] arrived at the following equation for the evaporation rate, $\hat{m}_v$.

$$\hat{m}_v \frac{\hat{T}_\infty - \hat{T}_s}{e^{\hat{m}_v} - 1} \ = \ \hat{m}_v \hat{L} - \hat{q}_l \qquad (63)$$

where
$\hat{T} \ = \ C_{pg}T/L_r$, $\hat{m}_v \ = \ (4\pi r_k^2 \rho_g u)/(4\pi \lambda_g r_{k,t=0}/C_{pg})$, $\hat{L} \ = \ L_k/L_r$. The solution for the above equation is obtained by employing the standard Newton's iteration.

It is noteworthy that because of the assumptions invoked in this model, the applicability of this model may be limited in several important ways: (1) our experience shows that the assumption of $T_\infty \ \geq \ T_{bub}(P_\infty)$ becomes too restrictive in most non-reactive calculations and also in some regions of most reactive spray calculations, and (2) the $\hat{q}_l$ term is modeled based on the assumption that the overall droplet life is very short (sub millisecond range) but in most moderate superheat conditions the droplet lifetimes are on par with those vaporizing under non-superheat conditions. So its use is likely to be restricted for the modeling of smaller droplets produced after flash induced atomization.

The solution for Eq. (63) and all the JP8 fluid properties needed in this calculation were obtained by the computer coding provided by UTRC.

8.3 Vaporization Model Valid Under Non-Superheat Conditions

Under moderate initial superheat conditions, only a fraction of the vaporization takes place under superheat conditions ($T_k > T_b$) and the remainder takes place under non-superheat evaporating conditions ($T_k \leq T_b$). So there is a need to revert back to a vaporization model valid under stable evaporating conditions when the internal droplet temperature approaches the boiling temperature. In the present calculations, the vaporization rate under normal evaporating conditions is evaluated by means of a simplified classical D^2-theory:

$$\dot{m}_k = 2\pi r_k \rho_g \; D_{fgs} \; Sh \; ln(1 + B_k) \tag{64}$$

where the Spalding mass transfer number, B_k, is given by Eq. (24) and the Sherwood number, Sh, is given by

$$Sh \; = \; 2(1 + 0.3 Re^{1/2} Sc_g^{1/3}) \tag{65}$$

8.4 Internal Droplet Temperature Calculation

Our experience with the validation studies showed us that there is a definite need to include a calculation involving the internal droplet temperature valid under both superheat and normal evaporating conditions. In our present calculations, it was evaluated by means of a simple infinite conductivity model.

$$\frac{dT_k}{dt} = \frac{3[l_{k,eff} - l_k]}{2C_{pl} r_k^2} \frac{ds_k}{dt} \tag{66}$$

$$if \; T_k \; < \; T_b, \quad and$$

$$\frac{dT_k}{dt} = -\frac{3\alpha_s}{r_k \rho_l C_{pl}} (T_k - T_b)$$

$$(for \; the \; superheat \; model \; of \;\; [62, 63-64])$$

$$or$$

$$\frac{dT_k}{dt} = -\frac{3\lambda_l}{r_k \rho_l C_{pl}} \frac{\partial T}{\partial r}|_{ls} \tag{67}$$

$$(for \; the \; superheat \; model \; of \; [74])$$

$$if \; T_k \; \geq \; T_b$$

9 SUPERCRITICAL DROPLET VAPORIZATION

There is a need for understanding droplet vaporization behavior at supercritical conditions because of the increasingly high operating pressures encountered in some gas-turbine combustors. The engine operating pressures can exceed the supercritical pressure of liquid fuels such as jet-a. Because of its practical and fundamental importance, several experimental and theoretical investigations were undertaken to understand droplet gasification occurring at supercritical conditions [19-20,82-86]. Most of the numerical investigations that appeared in the literature to study the vaporization behavior of an isolated single spherical droplet were based on the coupled twophase, unsteady Navier-Stokes equations in 1-D spherical coordinates [20,83-86]. However, it is noteworthy that the resulting gas-liquid interface analysis at the droplet surface becomes extremely complicated for a multi-component mixture. Also, the numerical solution is complicated by several factors as it needs to take into account the high-pressure corrections associated with various transport and thermo-physical properties. The supercritical droplet vaporization differs from low-pressure droplet vaporization models in several important ways. For example, the results of Zuo et al [85-86] showed that under low to moderate temperatures, first there would be an increase in droplet lifetime before it starts to fall off with an increase in pressure. Their results also showed that there would be a monotonic decrease in droplet lifetime with increase in pressure at high ambient temperatures.

However, none of these analyses could be directly incorporated into a comprehensive spray calculation because of any such detailed treatment would be prohibitively expensive in terms of the required computer CPU resources. Also, they fail to address how to handle the supercritical droplet vaporization in a real gas-turbine environment in which a droplet traverses through a non-stagnant gas. Since our main interest lies in implementing a viable high-pressure droplet vaporization model into our spray code, we attempted to incorporate some important aspects of high pressure modeling derived from Refs. [20,83-86] into the existing framework of our spray formulation. In what follows we describe some important aspects of supercritical droplet vaporization:

(1) The non-idealities in gas-phase become increasingly more pronounced as the pressure approaches a supercritical state. There are several widely used equations of state that provide accurate representation at high pressures, namely the Peng-

Robinson (PR), Redlich-kwong (RK), and Soave-Redlich-Kwong (SRK). In our calculations, we implemented the PR EOS because of its simplicity and also known to provide accurate representation [85-86]. Also, it is known to provide reasonable results when calculating the phase equilibrium at the droplet surface.

(2) At low pressure conditions, the solubility of the ambient gases in liquid phase can be neglected. The corresponding gas-phase composition at the droplet interface can be calculated by means of Raoult's law, Eq. (42). At high pressures, the phase equilibrium calculations at the droplet surface need to take into account fugacity of each individual species. More on how to evaluate the phase quilibrium at the droplet surface can be found in Section 9.1.

(3) Both liquid and gas compositions encountered in a gas turbine combustion device are truly multi-component in nature. However, all the numerical investigations reported so far on supercritical droplet vaporization are based on a binary mixture [19-20,83-86]. Even for a binary mixture, the phase equilibrium calculations at high pressures present a formidable challenge. As it becomes very difficult to solve anything other than a binary mixture, we restrict the phase equilibrium calculations to a binary mixture involving a combination of a single-component surrogate fuel and nitrogen. Such a combination seemed to provide a reasonable representation for a fuel droplet vaporizing at high pressures [85-86].

(4) The transport properties in both liquid and gas phases become increasingly pressure dependent at high pressures. More on how we evaluate the transport properties at high pressures can be found in Sections [4] & [7].

(5) As the droplet surface approaches a trans-critical state, the latent heat of vaporization diminishes to zero. In our current calculations, the latent heat of vaporization is calculated by a formulation as described in Section 9.2.

(6) The solubility of the ambient gases in liquid phase becomes increasingly important at high pressures. However in our present calculation, we ignore to take into consideration the multi-component nature of droplet behavior for the following reasons: For gas turbine combustors of our interest, the gas pressure seldom exceeds more than twice the critical pressure of jet fuels. Within that pressure range, the liquid phase solubility of gas could be ignored since it could be shown from phase equilibrium calculations that the liquid-phase mass fraction of nitrogen remains less than three percent even when the droplet surface temperature reaches near critical temperature.

(7) Zhu et al. [85-86] studied the influence of gas-phase unsteadiness on droplet vaporization, and also quantified to some degree the resulting differences from the quasi-steady and transient models. Their results show that unsteadiness seemed to persist over a wide region during a brief early transient period after a droplet is suddenly introduced into an otherwise stagnant gas. After this initial transition period, some unsteadiness remains persistent in a small region closer to the droplet surface. Initially, the quasi-steady model seemed to produce a smaller regression rate when compared with the transient model. But in the later stages of droplet lifetime, it seemed to produce a much higher regression rate. The influence of unsteadiness seemed to increase with an increase in both ambient pressure and temperature. However in our present calculations, we expect for the quasi-steady model to provide a useful approximation for the following reasons: (1) We are interested mainly in gas pressures not too far above the critical pressure of the liquid fuel, and (2) Also, we are interested in a droplet moving a stratified gas where the unsteadiness associated with initial transient can be neglected. It is because our droplet models are based on what happens after the initial atomization phase. So the neglected influence of unsteadiness is mostly limited to a small region near the droplet surface.

9.1 Equilibrium Relations Valid at High Pressure Conditions

At high pressures, the equilibrium needs to satisfy

$$T^L = T^V \tag{68}$$

$$P^L = P^V \tag{69}$$

$$f_i^L = f_i^V \quad => \quad x_i \phi_i^L = y_i \, \phi_i^V \tag{70}$$

where f_i is the fugacity of the ith species, x_i is the mole fraction species in the liquid phase, y_i is the mole fraction species in the vapor phase, and ϕ_i is the corresponding fugacity coefficient of the ith species which is given by

$$ln\ \phi_i = \frac{b_i}{b_m}(Z-1)-ln(Z-B^\star)+\frac{A^\star}{2\sqrt{2}B^\star}(\frac{b_i}{b_m}-\delta_i)ln\ \frac{Z+B^\star(1+\sqrt{2})}{Z+B^\star(1-\sqrt{2})} \tag{71}$$

where $\frac{b_i}{b_m} = \frac{T_{ci}/P_{ci}}{\sum_i y_j T_{cj}/P_{cj}}$, and $\delta_i = \frac{2a_i^{1/2}}{a_m}\sum_i x_j a_j^{1/2}(1-k_{ij})$. There are two more additional equations that need to be satisfied at the droplet interface,

$$\sum_i y_i = 1 \tag{72}$$

$$\sum_i x_i = 1 \tag{73}$$

For a given P and T of a binary mixture, there are six equations and six unknowns: Z_v, Z_l, y_1, y_2, x_1, and x_2. The Peng-Robinson EOS yields solution for Z_v and Z_l and the other four unknowns are calculated from the solution of Eqs. (70) to (73). The solution for this highly non-linear set of equations is obtained from the use of a Newton-Raphson iterative method.

9.2 Calculation of Latent Heat of Vaporization at High Pressure

The enthalpy of the species can be calculated from

$$h_i = h_i^o - RT^2\left[\frac{\partial ln\ \phi_i}{\partial T}\right]_{p,x_i} \tag{74}$$

The related derivatives in the above equation are given by

$$\begin{aligned} \frac{\partial ln\ \phi_i}{\partial T} &= \frac{b_i}{b}\frac{\partial Z}{\partial T} - \frac{(\frac{\partial Z}{\partial T} + \frac{B^\star}{T})}{Z-B^\star} - \frac{A^\star(\frac{b_i}{b}-\delta_i)(\frac{Z}{T}+\frac{\partial Z}{\partial T})}{Z^2-B^{\star 2}+2ZB^\star} \\ &+ \frac{1}{2\sqrt{2}B^\star}(\frac{b_i}{b}-\delta_i)(\frac{\partial A^\star}{\partial T}+\frac{A^\star}{T})ln\ \frac{Z+B^\star(1+\sqrt{2})}{Z+B^\star(1-\sqrt{2})} \\ &- \frac{A^\star}{2\sqrt{2}B^\star}\frac{\partial \delta_i}{\partial T}ln\ \frac{Z+B^\star(1+\sqrt{2})}{Z+B^\star(1-\sqrt{2})} && (75) \\ \frac{\partial a_i^{1/2}}{\partial T} &= -\frac{Rf_{wi}}{2}\left(\frac{0.45724T_{ci}}{TP_{ci}}\right)^{1/2} && (76) \\ \frac{\partial a}{\partial T} &= -\frac{R}{2}\left(\frac{0.45724}{T}\right)^{1/2}\sum_i\sum_j y_i y_j(1-k_{ij}) \\ &\left[f_{wi}\left(\frac{a_j T_{ci}}{P_{ci}}\right)^{1/2} + f_{wj}\left(\frac{a_i T_{cj}}{P_{cj}}\right)^{1/2}\right] && (77) \\ \frac{\partial A^\star}{\partial T} &= A^\star[\frac{1}{a}\frac{\partial a}{\partial T} - \frac{2}{T}] && (78) \\ \frac{\partial B^\star}{\partial T} &= -\frac{B}{T} && (79) \\ & && (80) \end{aligned}$$

$$\frac{\partial Z}{\partial T} =$$

$$\frac{[-\frac{\partial A}{\partial T}(Z-B^\star)+\frac{B^\star}{T}[Z^2-2Z(1+3B^\star)-A^\star+2B^\star+3B^{\star 2}]}{3Z^2-2(1-B^\star)Z+A^\star-2B^\star-3B^{\star 2}} \tag{81}$$

$$\frac{\partial \delta_i}{\partial T} =$$

$$\delta_i[\frac{1}{a_i^{1/2}}\frac{\partial a_i^{1/2}}{\partial T} - \frac{1}{a}\frac{\partial a}{\partial T}] + 2\frac{a_i^{1/2}}{a}\sum_j y_j(1-k_{ij})\frac{\partial a_i^{1/2}}{\partial T} \tag{82}$$

Finally, once the enthalpies in both phases are known, the latent heat of vaporization is simply calculated as follows

$$L_i = h_i^v - h_i^l \tag{83}$$

9.3 ncc_hp_eos_vle_table.dat: A Table Containing the Equilibrium Properties of a Binary Mixture

Instead of performing these calculations on the fly, it would be a lot more economical to create a table containing the equilibrium properties of a binary mixture. Especially in gas-turbine combustor calculations we need to tabulate these properties for one given pressure. Based on the calculations described in Sections 9.2 to 9.3, we developed the coding necessary to create a table, ncc_hp_eos_vle_table.dat, that contains values for the following variables: latent heat and mole fractions of both species, density, compressibility factor (Z), and specific volume in both liquid and gas phases. The table contains incremental values over a wide range of expected liquid droplet temperatures for any given pressure. The values for any required temperature is calculated by means of linear interpolation.

10 DETAILS OF ATOMIZATION MODELING

Atomization refers to a process of the liquid jet breakup into droplets. There are many processes associated with the liquid jet breakup. They can broadly be classified into two regimes: (1) the inner nozzle flow, and (2) liquid behavior after exiting from the nozzle. In the inner nozzle flow, several factors (such as injector type, geometry, and size) influence the conditions at the injector exit (such as the velocity, the initial sheet or jet thickness, and the angle of droplet dispersion). One way to specify the initial spray conditions at the injector exit is to rely on the widely-used correlations. For a better description, a more accurate analysis is needed which takes into account the physics associated with inside bubble growth, cavity formation, and internal turbulence. Once a liquid exits outside of the nozzle, it becomes unstable under the influence of aerodynamic instabilities and finally breaks up into droplets. The widely known aerodynamic instabilities are of Rayleigh-Taylor and Kelvin-Helmholtz kind [41, 43-44]. The Rayleigh-Taylor instability is due to inertia of the denser fluid opposing the system acceleration in a direction perpendicular to the interface of the denser fluid and the Kelvin-Helmholtz instability is caused by the viscous forces due to the relative motion of the fluids [43-44]. When the maximum amplitude of the most unstable wave reaches a critical value, some liquid is stripped of the main liquid core in the form of primary droplets. These droplets may further breakup into smaller droplets due to a process known as the secondary droplet breakup mechanism.

Based on a linear instability analysis of a 2D viscous incompressible fluid moving thorough an inviscid incompressible gas, Reitz and Bracco [41] characterized the break-up regimes to be four-fold: (1) Rayleigh breakup, (2) first wind-induced breakup, (3) second wind-induced breakup, (4) atomization. In the first two regimes, drops of sizes greater than or equal to the nozzle diameter are produced at distances far from the nozzle exit. In the applications of our interest, the last two regimes are more important where droplets much smaller than the nozzle diameter are produced near the nozzle exit. The knowledge gained from the instability analysis of various kinds [41, 45-46] is combined with some experimental observations to form the basis for various models developed for both primary atomization & secondary droplet breakup. In this aproach, the jet breakup is modeled first by making use of a drop representation approach in which discrete parcels of liquid are injected in the form of blobs with a characteristic size representative of the nozzle diameter instead of tracking an actual intact liquid core that forms at the nozzle exit. In the case of a planar or conical liquid sheet, the discrete parcels essentially represent liquid ligaments. Before the jet breakup the discrete parcels stay inside of the liquid core or sheet but after the jet breakup they move independently. The breakup criterion, atomization rate, drop size and velocity and the location of the newly formed droplets are primarily determined based on an instability analysis derived from the conservation equations of mass, momentum and energy. The analysis of the jet or sheet breakup into ligaments or droplets, the stripping of the liquid into fragments or droplets, and the formation of more droplets from further breakup of ligaments or fragments are all described under the classification of primary atomization.

Some of the large droplets that are formed immediately after the primary liquid jet breakup may further breakup into smaller droplets under the influence of aerodynamic instabilities. The large droplets first tend to flatten under the influence of aerodynamic pressure. Then large amplitude long wavelength waves caused by drop deceleration induce a Rayleigh-Taylor instability on the flattened drop causing it to breakup further into several relatively large-size product droplets. While at the same time short surface waves induce a Kelvin-Helmholtz instability on the windward side of the parent drop resulting in the generation of much smaller product droplets. The breakup of the larger droplets into smaller droplets is described under the classification of secondary droplet breakup.

Most of the coding required for both the primary atomization and secondary droplet breakup models was provided by CFDRC. These models are applicable under both non-superheat and superheat conditions. In the superheated regime, the thermophysical liquid properties are replaced by the two-phase properties of a superheated fluid [74]. The CFDRC atomization module contains the following four primary atomization models: (1) the sheet breakup primary atomization model, (2) the blob jet primary atomization model, (3) the air-blast atomization model & (4) the BLS (Boundary-Layer Stripping) primary atomization model, and it also contains the following three secondary droplet breakup models: (1) the Rayleigh-Taylor secondary droplet breakup model, (2) the TAB (Taylor Analogy Breakup) secondary droplet breakup model, and (3) the ETAB (Enhanced TAB) secondary droplet breakup model. The choice between various models depends on the specific application. Complete details of the models contained in the atomization module can be found in the appendix.

10.1 Flash Atomization Modeling

There are three components to the atomization modeling of a superheat spray: (1) primary atomization, (2) secondary droplet breakup, and (3) flash induced atomization. Further details of both primary atomization and secondary droplet breakup models are described in a previous section. For the superheated sprays, we have to deal with a third component known as flash induced atomization. We have followed the modeling approach of Lee et al. [74] for the flash induced atomization. In what follows we describe some details of their modeling approach.

Flash atomization is a result of rapid evaporation associated with strong nucleation and bubble growth occurring inside of a non-equilibrium two-phase fluid [74]. When the void fraction within the rapidly evaporating fluid reaches a certain critical void fraction, the inner liquid core would shatter into pieces. It seemed to have a value of 0.61 based on the research identified in [74]. Incidentally, this condition for criticality also coincides with the packing limit of spheres beyond which it can not accommodate any more bubbles and results in shattering [75]. In order to calculate this state of critical limit, they employed the homogeneous relaxation model (HRM) for calculating the mass fraction of gas, x, also known as quality.

$$\frac{dx}{dt} \;=\; \frac{x_{eq}-x}{\theta};\;\; \theta \;=\; \theta_o \epsilon^{\alpha} \phi^{\beta}; \tag{84}$$

where the values for α ($=-0.54$), β ($=-1.76$), and θ_o ($=6.51.10^{04}s$) are taken from [76], and the void fraction, ϵ, and the degree of superheat, ϕ, are defined as follows:

$$\epsilon \;=\; \frac{\rho_l-\rho}{\rho_l-\rho_v};\;\; \phi \;=\; \frac{P_b(h)-P}{P_c-P_b(h)} \tag{85}$$

Once the critical limit for the void fraction is reached, the resulting droplets produced as a result of flash atomization are obtained by a droplet number density correlation developed by Kawano et al. [77].

$$n \;=\; Ae^{\frac{-E}{(T_l-T_{bub})}}e^{at+b} \tag{86}$$

From Eq. (86), the mean droplet size, $\bar{R}$, is calculated by equating total liquid mass before and after the droplets are formed.

$$\bar{R}^3 \;=\; \frac{1-\epsilon_c}{(4/3)\pi n} \tag{87}$$

The solution for Eqs. (84) to (87) and all the JP8 fluid properties needed in this calculation were obtained by the computer coding provided by UTRC. Also, it is noteworthy that the properties for JP8 were predicted by means of a surrogate model of Violi. Further details can be found in [74].

The flash induced atomization is combined with the primary and secondary droplet breakup models following the approach of Lee et al. [74]. The aerodynamic instability mode will atomize the liquid core when the surface disturbance reaches a certain critical value and the flash induced atomization is responsible for shattering the liquid core when the void fraction reaches the critical value. Therefore, the control mechanism depends on whichever mechanism is first responsible for reaching the critical condition.

10.2 Some Effects of Superheat on Primary Atomization

Flash evaporation is known to produce a fine spray with enhanced atomization, increase effective spray cone angle, and decrease spray penetration [61]. Here, we consider the effects of flash evaporation on the sheet breakup primary atomization model by following the approach of Zuo et al [62]. Its effect on the initial droplet size generated immediately after the first ligament breakup, d_{is}, is given as a function of both engine pressure and a superheat parameter.

$$d_{is} \;=\; d_{in}(\frac{P}{P_{atm}})^{0.27}[1-\chi(\frac{P_{atm}}{P})^{0.135}] \quad 0 \leq \chi(\frac{P_{atm}}{P})^{0.135} \leq 1 \tag{88}$$

where d_{in} is the corresponding droplet droplet size occurring under normal evaporating conditions without flash evaporation, and χ, a superheat parameter, is defined as follows:

$$\chi \;=\; \frac{I(T_k)-I(T_b)}{l(T_b)} \tag{89}$$

where I is the internal energy of the liquid. Its value varies between $0 < \chi < 1$ with $\chi = 0$ referring to zero flash evaporation and $\chi = 1$ to full flash evaporation.

In Eq. (88), the increase in d_{is} due to an increase in engine pressure by a factor of $(\frac{P}{P_{atm}})^{0.27}$ is based on an experimental correlation obtained from Lefebvre [66]. It reflects the influence of chamber pressure on wave propagation as it damps wave growth. But the decrease by a factor of $(1-\chi(\frac{P_{atm}}{P})^{0.135})$ is due to a significant reduction in droplet size caused by both cavitation and bubble growth under flash evaporation conditions. It was introduced based on the experimental data obtained from VanDerWege et al [67] and Reitz [68].

As the liquid approaches boiling, it also causes a substantial decrease in both intact liquid core length and core droplet size leading to a modification of the nominal cone angle, θ, as given by

$$\theta \ = \ \theta_n \ + \ (144 - \theta_n)\chi^2 \qquad (90)$$

where θ_n is in degrees for a spray vaporizing under normal conditions without flash evaporation. This correlation was developed based on the experimental data of Reitz [68].

These two correlations were developed in conjunction with the sheet breakup primary atomization model but their application with other primary atomization models need further investigation.

11 DESCRIPTION OF INITIAL SPRAY CONDITIONS

The spray computations facilitate the use of multiple fuel injectors. The same or a different type of liquid fuel can be specified for each one of different injectors. The initial droplet temperature is assumed to be the same for all different droplet groups of a given injector. The liquid fuel injection is simulated by introducing a number of discretized parcels of liquid mass at the beginning of every fuel-injection time step, Δt_{il}. The following three variables play an important role in simulating the injector initial conditions:

- no_of_holes() - the number of holes per injector,
- no_of_streams() - the number of droplet streams per hole - it is introduced to distinguish the initial velocity variation within different droplet classes arising from the geometric considerations of a chosen spray, &
- no_of_droplet_groups() - the number of droplet groups per stream - it is introduced to distinguish the droplet-size variation within different droplet classes of a polydisperse spray.

The total number of droplet groups introduced at the beginning of every different injection time step for a given injector is therefore equal to a value given by the multiplication of these three variables. Further details on some of the spray input variables can be found in the spray input file, ncc_injector.in.1 of Table 6, where the integer number followed by the last dot represents the injector number, which in this case happens to be one. The table provides a complete description of the following input variables: nolc(n_i), out_string, (ymki(n_i,n_l), n_l=1,nolc(n_i)), tdrop(n_i), atomization(n_i), drop_breakup_model(n_i), spray_table(n_i), steady_spray_model, no_of_holes(n_i), no_of_streams(n_i), no_of_droplet_groups(n_i), lmdis(n_i), smdm(n_i), cone(n_i), size_min(n_i), size_max(n_i), stochastic(n_i), ((x_inj(n_i,nx), y_inj(n_i,nx), z_inj(n_i,nx), z_inj(n_i,nx), flowf(n_i,nx), v_inj(n_i,nx), alpha_inj(n_i,nx), beta_inj(n_i,nx), theta_inj(n_i,nx), dtheta_inj(n_i,nx), swlr_angle(n_i,nx)), nx=1,no_of_holes(n_i)), (atom_type(n_i,nx), breakup_type(n_i,nx), dia_hole(n_i,nx), delp_inj(n_i,nx), liq_vel(n_i,nx), gas_u(n_i,nx), gas_v(n_i,nx), gas_w(n_i,nx), pcl_start(n_i,nx), pcl_end(n_i,nx)), nx=1,no_of_holes(n_i)).

The initial droplet distribution for a given injector could be specified by making use of one of the three available options: (1) by providing a complete specification of the initial conditions through a spray table, (2) by invoking certain pre-defined correlations, or (3) by chosing one of four available primary atomization models. When Option (3) is selected, the logical variable, atomization(), is set to .true. or otherwise it is set to .false..

Option (1):

When a spray table is defined, one should provide the information on the (x,y z) components of the initial droplet location, the (u, v, w) components of the droplet velocity, and the initial mass flow rate associated with each different droplet group as described in ncc_spray_table.1 of Table 7. This table provides a complete description of the following variables: nos(n_i), (ni,xx_inj, yy_inj, zz_inj, uu_inj, vv_inj, ww_inj, r_inj, fld_d), ,ni=1,nos(n_i)). The initial inputs specified through a spray table should be representative of the integrated averages of the experimental conditions [10-13].

Option (2):

In this option, we need to specify several parameters including no_of_holes(), no_of_streams(), & no_of_droplet_groups(). And depending on what is specified for the input of the integer variable, lmdis, in ncc_injector.in.1, the droplet-size distribution within each one of the streams is determined based on the following three choices:

- In one choice, it is calculated based on a correlation typical of those widely used in describing the initial droplet size distribution [4]:

Table 6. Input file: ncc_liquid_injector.in.01.

Input file content	comments
heading	title of a description of this file
heading	title of property
nolc(n_i)	denotes the total number of initial liquid components in the n_i-th injector.
heading	title of property
out_string	chemical symbols of initial liquid species: e.g., C6H12.
(ymki(n_i,n_l), n_l=1,nolc(n_i))	initial mass fractions of liquid species
heading	title of controlling parameters
tdrop(n_i), isuperhd(n_i), ijetapr(n_i), vmfdrop(n_i)	tdrop(n_i) denotes the initial droplet temperature. If isuperhd(n_i) = 1, it invokes the superheat vaporization model of Zuo et al. [62], and Schmehl et al. [63-64]; If isuperhd(n_i) = 2, it invokes the superheat vaporization model of Lee et al. [74]; & if isuperhd(n_i) = 0, it selects a non-superheat vaporization model. if ijetapr(n_i) = .true., all the JP8 fluid properties are calculated by the computer coding provided by UTRC as described in Lee et al. [74]. vmfdrop(n_i) denotes the initial quality (= gas mass fraction) of a superheated fluid. It only applies when ijetapr(n_i) = .true.
heading	title of controlling parameters
atomization(n_i), drop_breakup_model(n_i), line_injection(n_i), spray_table(n_i), steady_spray_model	If atomization(n_i) = .true., the initial droplet conditions are determined based on the use of a primary atomization model. Otherwise they are determined from either known experimental conditions or a widely-used correlation. If drop_breakup_model(n_i) = .true., it invokes the secondary droplet breakup option. if line_injection(n_i) = .true., it invokes a multi-point spray injection option. If spray_table = .true., initial droplet location, velocity, size, and mass flow rate are input through the file - ncc_liquid_table.in.1. Otherwise they are determined by some other means. If steady_spray_model = .true., it invokes a steady state spray model commonly used in many spray codes, where whenever the spray solver is called, after first introducing a new group of spray particles, it continues with the liquid-phase computation until after all the particles are taken out of the computational domain (note: It is **NOT** recommended to use this option as a steady state calculation could be better arrived at by making use of some features of the unsteady option). The solution from the unsteady option (steady_spray_model = .false.) is determined based on the values assigned to the controlling time steps - dtml, dtil, & dtgl, which are internal to the spray solver.

Table 6. Input file: ncc_liquid_injector.in.01 (continued).

heading	title of controlling parameters
no_of_holes(n_i), no_of_streams(n_i), & no_of_droplet_groups(n_i)	no_of_holes(n_i) = The number of holes per injector (note: when atomization() = .true. or spray_table = .true., the no_of_holes(n_i) is set equal to 1).
	no_of_streams(n_i) = The number of droplet streams per hole. This variable is introduced mainly to distinguish the variation in the droplet groups based on angular orientation (note: when atomization() .true. or spray_table = .true., the no_of_streams is set equal to 1). no_of_droplet_groups(n_i) = The number of droplet groups per stream. This variable is introduced mainly to distinguish the droplet groups based on the droplet-size variation (note: when spray_table = .true. or atomization() = .true., all different injected droplets are lumped together into a single variable, no_of_droplet_groups()).
heading	title of controlling parameters
lmdis(n_i), smdm(n_i), cone(n_i), size_min(n_i), size_max(n_i), & stochastic(n_i)	If lmdis(n_i) = 1, it invokes the droplet size distribution as given by Eq. (91); If lmdis(n_i) = 2, it assumes a uniform droplet distribution between the maximum and minimum limits as specified by size_max() & size_min(); & If lmdis(n_i) = 3, it invokes the droplet size distribution as given by Eq. (92). smdm(n_i) = Sauter mean diameter If cone(n_i) = .true., it activates a 3D solid or hollow cone spray configuration as shown in Fig. 2. Otherwise it activates a 2D configuration depending on the value chosen for the logical variable - axisymmetric. If it is set equal to .true., it invokes the axis-of-symmetry case as shown in Fig. 3 otherwise it invokes the planar case as in Fig. 4. size_min(n_i) & size_max(n_i) = The variables are associated with the lmdis(n_i) = 2 option. stochastic(n_i) is used in conjunction with Option (2) of Sec. 9. If set equal to .true., it introduces some randomness into the determination of initial droplet velocity distribution.
heading	title of controlling parameters
((x_inj(n_i,nx), y_inj(n_i,nx), z_inj(n_i,nx), flowf(n_i,nx), v_inj(n_i,nx), alpha_inj(n_i,nx), beta_inj(n_i,nx),	(x_inj(n_i,nx),y_inj(n_i,nx),z_inj(n_i,nx)) = spatial coordinates of the initial injector location. flowf(n_i,nx) = injector mass flow rate, (units - kgm/s for 3D & axis-of-symmetry & kgm/s/m for 2D planar. REMEMBER FOR AXIS-OF-SYMMETRY, IT IS THE TOTAL FLOW RATE OVER 360 DEGS.)

Table 6. Input file: ncc_liquid_injector.in.01 (continued).

theta_inj(n_i,nx), dtheta_inj(n_i,nx), swlr_angle(n_i,nx)), nx=1,no_of_holes(n_i))	v_inj(n_i,nx) = droplet injection velocity, m/s. alpha_inj(n_i,nx) = angle of rotation from the x-y plane. beta_inj(n_i,nx) = angle of rotation from the x-axis. theta_inj(n_i,nx) = cone angle. dtheta_inj(n_i,nx) = half-cone angle (note: Although dtheta_inj(n_i,nx) = theta_inj(n_i,nx)/2 for SOLID CONE SPRAY, it is invoked by setting dtheta_inj(n_i,nx) either equal to 0 or theta_inj(n_i,nx)/2). Refer to Figs. 2-4, for a better understanding of the angular representation. swlr_angle(n_i,nx) = swirl angle.
heading	title of controlling parameters
(atom_type(n_i,nx), breakup_type(n_i,nx), dia_hole(n_i,nx), delp_inj(n_i,nx), liq_vel(n_i,nx), gas_u(n_i,nx), gas_v(n_i,nx), gas_w(n_i,nx), pcl_start(n_i,nx), pcl_end(n_i,nx)), nx=1,no_of_holes(n_i))	atom_type(n_i,nx) = 0 => no primary atomization specified, = 1 => blob jet primary atomization model, = 2 => sheet breakup primary atomization model, = 3 => air blast primary atomization model, = 4 => BLS primary atomization model. breakup_type(n_i,nx) = 0 => no secondary droplet breakup model specified, = 1 => Rayleigh-Taylor secondary droplet breakup model, = 2 => TAB secondary droplet breakup model, = 3 => ETAB secondary droplet breakup model. dia_hole(n_i,nx) represents the injector orifice diameter (m). delp_inj(n_i,nx) represents the pressure drop across the injector (Pa) (note: needed for pressure swirl atomization only). liq_vel(n_i,nx) represents the liquid velocity from annulus (m/s) (note: needed for air-blast atomization only). gas_u(n_i,nx), gas_v(n_i,nx), gas_w(n_i,nx): gas velocity components in annulus adjacent to liquid film (m/s) (note: needed for air-blast atomization only). pcl_start(n_i,nx): starting parcel number for the annular air-blast injector. pcl_end(n_i,nx)): ending parcel number for the annular air-blast injector.

Table 6. Input file: ncc_liquid_injector.in.01 (continued).

heading	title of controlling parameters
(line_injection_type(n_i,nx), xi_inj(n_i,nx), yi_inj(n_i,nx), zi_inj(n_i,nx), xf_inj(n_i,nx), yf_inj(n_i,nx), zf_inj(n_i,nx), ri_inj(n_i,nx))	line_injection_type(n_i,nx) = 1 => line injection with full cone angle, = 2 => line injection with half cone angle, = 3 => circular injection with full cone angle, = 4 => circular injection with half cone angle. (xi_inj(n_i,nx),yi_inj(n_i,nx),zi_inj(n_i,nx)) = In the case of line injection, they represent the coordinates of the first end point of a line. Otherwise in the case of circular injection, they represent the coordinates of its center. In the case of line injection, (xf_inj(n_i,nx),yf_inj(n_i,nx),zf_inj(n_i,nx)) represent the coordinates of the second end point of a line. Otherwise in the case of circular injection, xf_inj(n_i,nx) & yf_inj(n_i,nx) represent the angular location of its two end points in degrees. In the case of circular injection, ri_inj(n_i,nx) represents its radius.

Table 7. Input file: ncc_spray_table.in.01.

Input file content	**comments**
nos(n_i)	denotes the total number of droplet groups in the n_i-th injector.
(ni,xx_inj,yy_inj, zz_inj,uu_inj,vv_inj, ww_inj,r_inj,fld_d) ,ni=1,nos(n_i))	ni = number of the droplet group. Its value ranges between 1 to nos(n_i). (xx_inj,yy_inj,zz_inj) = spatial coordinates. (uu_inj,vv_inj,ww_inj) = velocity components. r_inj = droplet size in radius. fld_d = mass flow rate of the ni-th droplet group (note: SUMMATION OF fld_d OVER ALL THE nos(n_i) DROPLET GROUPS IS EQUAL TO THE TOTAL MASS FLOW RATE OF THE INJECTOR).

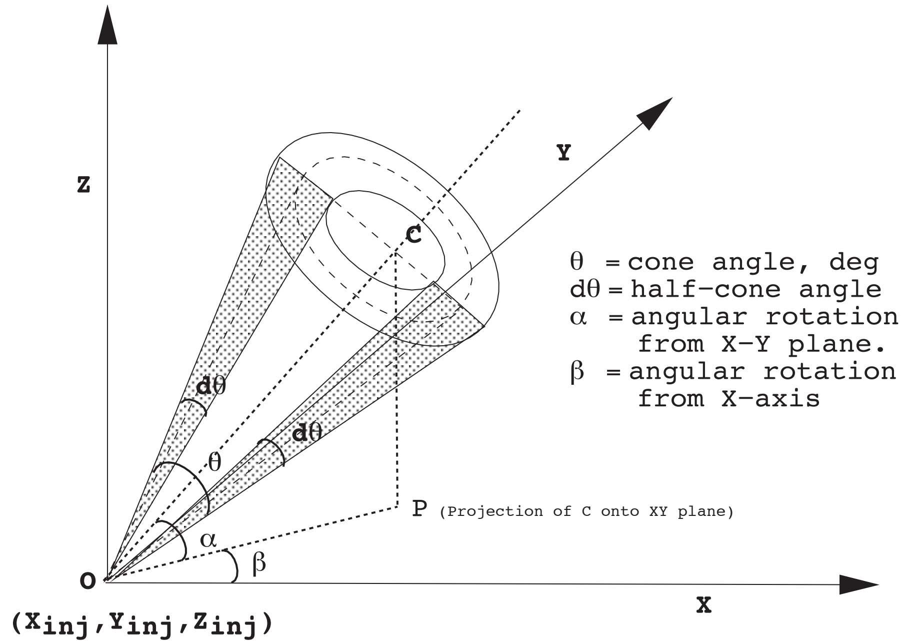

Z
Y
C
θ = cone angle, deg
dθ = half-cone angle
α = angular rotation from X-Y plane.
β = angular rotation from X-axis
dθ
dθ
θ
P (Projection of C onto XY plane)
α
β
O
X
(X_{inj}, Y_{inj}, Z_{inj})

Fig 2a. Geometrical details of fuel injection for a 3D solid or hollow cone spray.

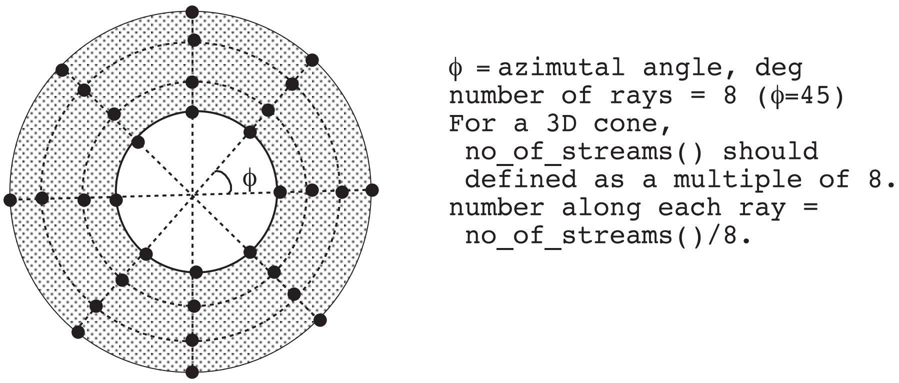


Fig 2b. Initial spray particle orientation in a circular cross-section.

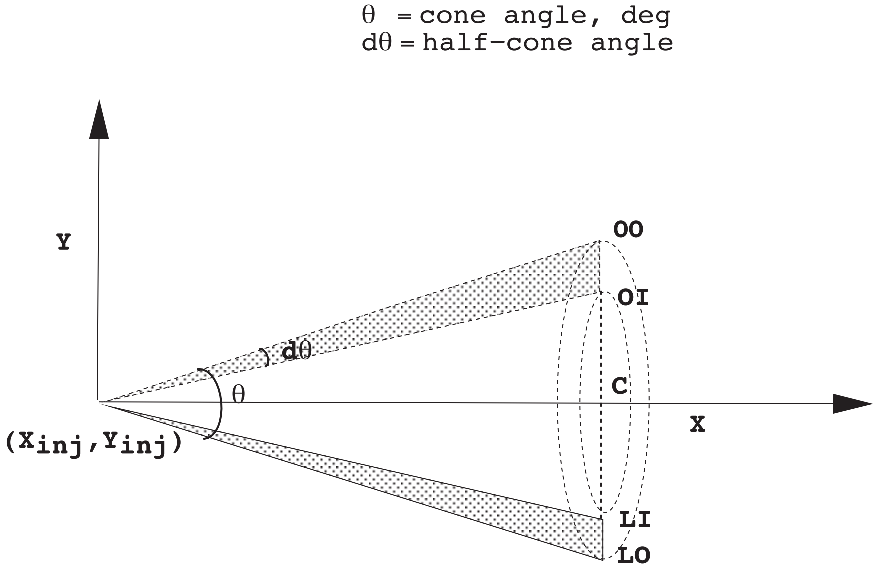
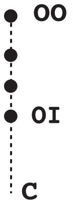


Fig 3a. Geometrical details of fuel injection for an axis-of-symmetry case.



For axis of symmetry:
no_of_streams() is distributed
between **OI** to **OO**.

Fig 3b. Initial spray particle concentration in an axis-of-symmetry case.

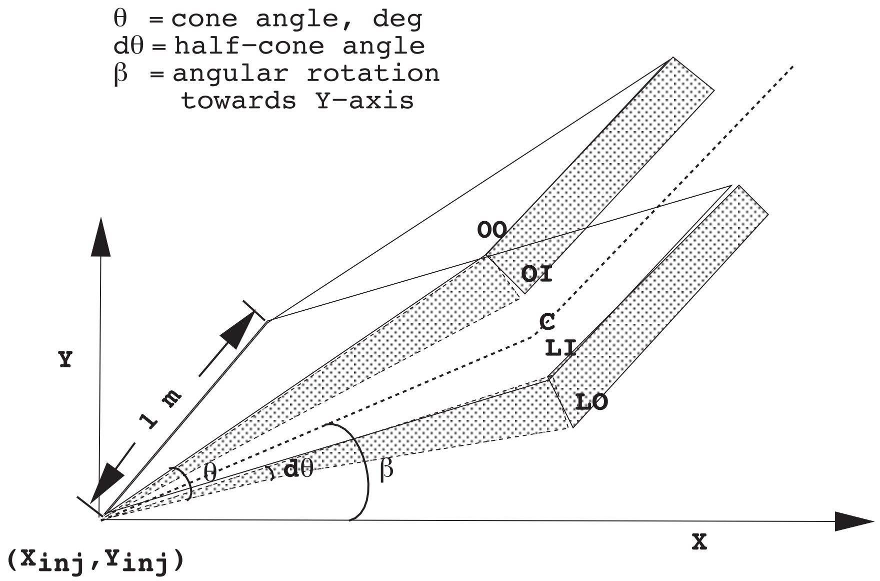
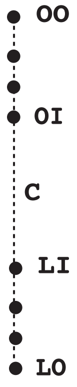


Fig 4a. Geometrical details of fuel injection for a 2D planar case.



For 2D:
no_of_streams()/2 is between **OI** to **OO**, and
no_of_streams()/2 is between **LO** to **LI**.

Fig 4b. Initial spray particle concentration in a 2D planar case.

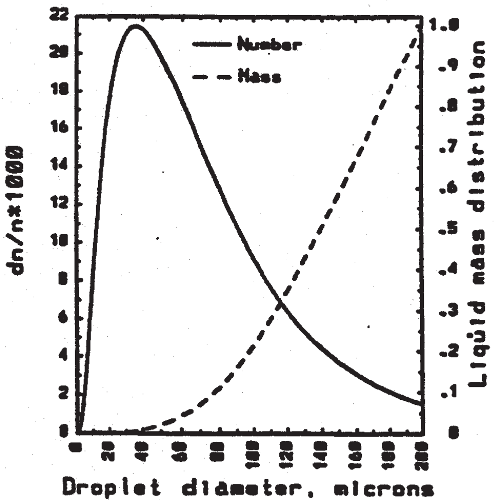


Figure 5 Droplet-size distribution.

$$\frac{dn}{n} = 4.21\ x\ 10^6\ \left[\frac{d}{d_{32}}\right]^{3.5} e^{-16.98\left(\frac{d}{d_{32}}\right)^{0.4}} \frac{dd}{d_{32}} \tag{91}$$

where n is the total number of droplets and dn is the number of droplets in the size range between d and $d + dd$. This correlation also requires the specification of Sauter mean diameter, d_{32}. Fig. 5 shows the droplet size distribution generated by Eq. (91) for a case studied in [10]. The solid line shows the droplet number variation versus drop size and the dashed line shows the integrated mass variation with drop size. The drop size distribution within the spray is represented by a finite number of droplet classes as given by the variable, no_of_droplet_groups().

- In the second choice, the initial droplet sizes are distributed evenly between two specified maximum and minimum drop sizes - size_max() & size_min() as specified in Table 6. And the size interval depends on the value specified for the variable, no_of_droplet_groups(). However, this option is applicable only in certain special cases.

- In the third choice, the initial droplet size distribution is calculated by making use of a cliped probability distribution function as given by

$$P(\chi)\ =\ \int_0^{\chi} f(x)dx/\int_0^{12} f(x)dx \tag{92}$$

where $f(x)\ =\ (1.0-exp(-x)(1+x+\frac{1}{2}x^2+\frac{1}{6}x^3)$. After a random sampling of $0 < \chi < 12.0$, the initial droplet diameter is determined as follows: $(0.3 < 4P(\chi) < 1.5)d_{32}$. If the value for $4P(\chi)$ goes beyond the limits of 0.3 and 1.5, the random sampling is reiterated until that number falls within the range of the lower and upper bounds.

Depending on what is specified for the logical variable, cone, of Table 6, the droplet velocity distribution amongst various streams of a given hole is calculated by assuming the spray to be either a solid or hollow cone spray. A graphical illustration of three different cone configurations are shown in Figs. 2 to 4. Figs. 2a & 2b refer to a 3D case, Figs. 3a & 3b to an axisymmetric case, and Figs. 4a & 4b to a 2D planar case. It is noteworthy that in an axisymmetric case, the x-axis is assumed to be aligned with the axis-of-symmetry.

Fig. 2a shows the geometric details of a hollow cone spray in 3D where θ is known as the cone angle, $d\theta$ is the half-cone angle, and for a solid cone spray $d\theta\ =\ \theta/2$. And α represents the angular rotation from the x-y plane and β is the angular rotation from the x-axis. Fig. 2b shows initial spray stream orientation in a circular cross section. We try to simulate the spray by a finite number of streams as given by the variable, no_of_streams(). Each one of the dark circles in Fig. 2b represents a different stream. The streams are distributed evenly along each one of the different rays which are separated from each other with an angle of separation as given by ϕ. Currently, we have hard-coded the angle of separation to be 45^0 which means we have restricted the number of rays to be eight in 3D. Therefore, when specifying a number for no_of_streams(), it should be borne in mind that it should be a multiple of eight. And the number of streams along each one of the rays, therefore, becomes equal to no_of_streams()/8. In Fig. 2b, the no_of_streams() has a value of 32 and, therefore, the number of streams along each one of the rays for this case is 4.

Fig. 3a. shows the geometric details for an axisymmetric case. Since the computations are performed only in the first quadrant of the x-y plane, all of the specified number of streams, no_of_streams(), are distributed evenly between the lower and upper halfs as shown in Fig. 3a and 3b. Here, the upper half refers to the region between OI to OO and the lower half refers to LO to LI. It is noteworthy that each droplet group in a given stream represents a circular ring of liquid for an axisymmetric case.

The geometric details for a 2D case are shown in Figs. 4a & 4b. Here, it is noteworthy that each droplet group in a given stream represents a planar sheet of liquid. All the specified number of streams, no_of_streams(), are distributed evenly over both sides of the cone center as shown in Fig. 4b.

For each one of the different injector holes, no_of_holes(), it requires the specification of the following parameters as described in ncc_injector.in.1 of Table 6 (However, because of the geometric differences that exist between 3D, 2D, and axisymmetric configurations, some of the input parameters may have different units as noted below):

- The initial (x, y, z) coordinates of the hole location.
- The mass flow rate per hole - however, the definition of the units for the injector mass flow rate per hole differs: it is kgm/s for 3D & the axis-of-symmetry and kgm/s/m for 2D planar (note: In an axis-of-symmetry, the specified mass flow rate per hole refers to the entire mass flow rate over 360 degrees).
- The following variables define the angular orientation: α_{inj} =angle of rotation from the x-y plane, β_{inj} = angle of rotation from the x-axis, θ_{inj} = cone angle, & $d\theta_{inj}$ = half-cone angle (note: Although $d\theta_{inj} = \theta_{inj}/2$ for a solid cone spray, a specified value of zero for $d\theta_{inj}$ also invokes a solid-cone spray configuration).
- The variable, swlr_angle(), allows a means to specify the tangential component in the case of both 3D and axisymmetric sprays.

Stochastic Option

In the above description of Option (2), the initial droplet conditions are determined based on a deterministic approach. However, such an approach when employed in 3D may lead to time consuming calculations requiring the use of many droplet groups. With this in mind, we wanted to introduce some randomness into the calculation of the initial droplet velocity distribution as an option. This option can be invoked by setting the logical variable, stochastic(), to be .true.. In a stochastic approach, the initial angular orientation of a particle could be randomized in a number of different ways. The obvious way would be to determine the initial droplet direction based on a complete randomization in a field as defined by the geometric parameters: solid cone angle, θ, half cone angle, $d\theta$ and/or azimuthal angle, ϕ. Another way is try to retain the basic features of the deterministic approach where the initial droplet directions are taken as the means in a random field with the randomness localized to within the surrounding angular sub-segment of the respective droplet as represented by $\delta\theta$, $\delta d\theta$, $\delta\theta\delta\phi$, or $\delta d\theta\delta\phi$. We have chosen to go with the later option as it preserves the same basic structure but provides some benefits of the stochastic approach.

Option (3):

Here many aspects of fuel injection remain the same as in Option (2) but the initial droplet size and velocity distribution may differ depending upon the primary atomization model selected. More details on the available primary atomization models can be found in the appendix. For each one of the different injector holes, no_of_holes() it requires the specification of all the parameters as in Option (2). However, it doesn't make use of the values set for size_max() & size_min(). Also, it requires the specification of some additional parameters as described in ncc_injector.in.1 of Table 6. However, there exist some differences with regard to their usage depending upon the primary atomization model. Some major differences are highlighted below:

- The integer variable, atom_type() of Table 6, is used to select the primary atomization model of one's choice.
- dia_hole() needs to be specified for all primary atomization models.
- delp_inj() needs to be specified only with the sheet breakup primary atomization model and it is used in Eq. (9) of the appendix.
- liq_vel(n_i,nx), gas_u(n_i,nx), gas_v(n_i,nx), gas_w(n_i,nx), pcl_start(n_i,nx), & pcl_end(n_i,nx)) are required only for the air-blast primary atomization model.
- In all of the primary atomization models except for the air-blast, the total number of droplet groups (particles) to be injected at the time of every injection period is specified though the use of the variable, no_of_droplet_groups().

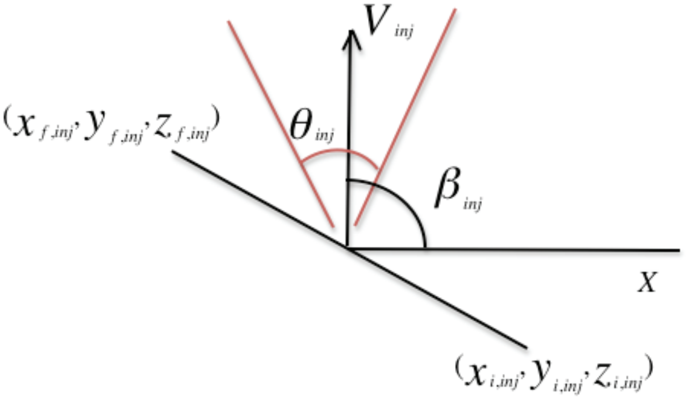

V_{inj}
$(x_{f,inj}, y_{f,inj}, z_{f,inj})$
θ_{inj}
β_{inj}
X
$(x_{i,inj}, y_{i,inj}, z_{i,inj})$

Fig. 6 A graphic illustration of a multi-point (line) injection.

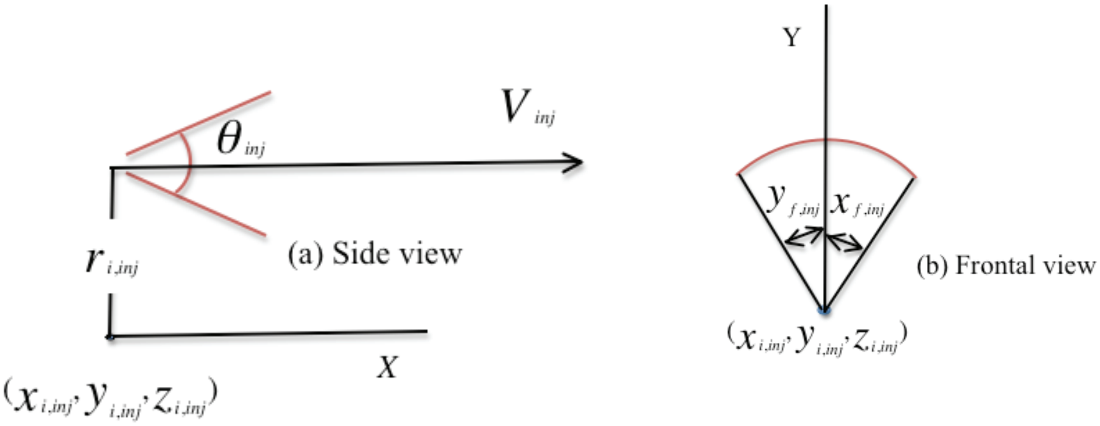


Fig. 7 A graphic illustration of a multi-point (circular) injection.

For the air-blast, it is given by pcl_end(n_i,nx)-pcl_start(n_i,nx)+1.

- In this option, all of the injected droplet groups as specified by either no_of_droplet_groups() or pcl_end(n_i,nx)-pcl_start(n_i,nx)+1 are distributed randomly across the entire cross-section of any selected 2D, axisymmetric, or 3D spray configuration. So it is recommended that no_of_streams() be set equal to one as no distinction is made between streams.

The secondary droplet breakup option can be invoked regardless of how the initial droplet conditions are generated using Options (1) to (3). The secondary droplet breakup option can be invoked by setting the logical variable, drop_breakup_model(), to be .true.. And then the integer variable, breakup_type() of Table 6, is used to select the secondary droplet breakup model of one's choice. More details on the available secondary droplet breakup models can be found in the appendix.

12 MULTI-POINT (LINE OR CIRCULAR) SPRAY INJECTION

In order to perform some reacting spray calculations involving a GE Taps concept which was developed as a part of the NASA's fundamental aeronautics/supersonic initiative on high altitude emissions, we implemented some new methods of spray injection into our spray code: (1) a line injection, and (2) an injection along a circular arc.

In line injection, a new droplet is injected randomly between two end points of a line as specified by $(x_{i,inj}, y_{i,inj}, z_{i,inj})$ and $(x_{f,inj}, y_{f,inj}, z_{f,inj})$. The initial droplet velocity is defined by a velocity magnitude as specified by V_{inj} and its orientation as specified by an angle, β_{inj}, in degrees. Within the line injection, we have added two different options: (1) line injection with a full cone angle, and (2) line injection with a half cone angle. In both these options the angle is given by θ_{inj} in degrees, and a newly injected droplet is chosen randomly to fall within the specified cone angle. A graphic illustration of the line injection with a full cone angle is shown in Fig. 6.

In circular injection, a new droplet is injected randomly along a circular arc between two end points along the circumference of a circle. For this case, the coordinates of the circular arc are specified by the following parameters: its center location, $(x_{i,inj}, y_{i,inj}, z_{i,inj})$, its radius, $r_{i,inj}$, and the angular location of its two end points, $x_{f,inj}$ and $y_{f,inj}$ in degrees. The initial droplet velocity is defined by a velocity magnitude as specified by V_{inj} and its orientation as specified by an angle, β_{inj}, in degrees. Within the circular injection, we have added two different options: (1) circular injection with a full cone angle, and (2) circular injection with a half cone angle. In both these options the angle is given by θ_{inj} in degrees, and a newly injected droplet is chosen randomly to fall within the specified cone angle. A graphic illustration of the circular injection with a full cone angle is shown in Fig. 7.

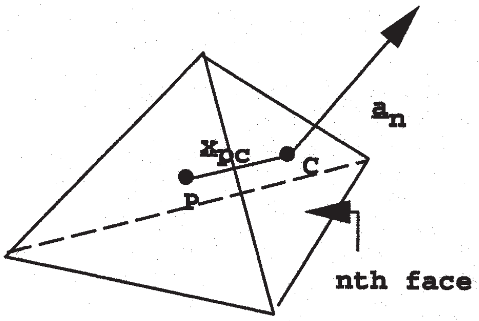


Fig. 8 A vector illustration used in the particle search analysis.

It is also important to note that the initial droplet size is determined based on a given value for SMD and a clipped probability density function as given by Eq. (92). And the number of droplets in a given parcel, (n_k), is determined based on the initial liquid mass flow rate and the specified value for no_of_droplet_groups(n_i).

13 SPRAY SOLUTION ALGORITHM

In order to evaluate the initial conditions that are needed in the integration of the liquid-phase equations, we first need to know the surrounding gas-phase properties at each particle location. But in order to evaluate the gas-phase properties, it is first necessary to identify the computational cell in which a given particle is located. It is a trivial task to track a particle in the regular rectangular coordinates. However, the particle tracking becomes complicated when the computational cell is no longer rectangular in the physical domain and it becomes even more complicated when the particle search is performed within the context of parallel computing.

We have developed and implemented an efficient particle-tracking algorithm for use with parallel computing in an unstructured grid. The search is initiated in the form of a local search originating from the computational cell in which the same particle was found to be located in the previous time step. The location of the computational cell is then determined by first evaluating the dot product of $\underline{x}_{pc} \cdot \underline{a}_n = |x_{pc}|$ $|a_n|$ $cos\,(\phi)$, where $\underline{x}_{pc}$ is the vector defined by the distance between the particle location and the center of the n-face of the computational cell, $\underline{a}_n$ is the outward area normal of the n-face as shown in Fig. 8, and ϕ is the angle between the two vectors.

A simple test for the particle location requires that the dot product be negative over each and every one of the n-faces of the computational cell. If the test fails, the particle search is carried over to the adjacent cells of those faces for which the dot product turns out to be positive. Some of those n-faces might represent the boundaries of the computational domain while the others represent the interfaces between two adjoining interior cells. The search is first carried over to the adjacent interior cells in the direction pointed out by the positive sign of the dot products. The boundary conditions are only implemented after making sure that all other remaining possibilities point towards a search exterior of the computational domain. This implementation ensures against any inadvertent application of the boundary conditions before the correct interior cell could be identified.

After the gas-phase properties at the particle location are known, the solution for the ordinary differential equations of particle position, size, and velocity are advanced by making use of a second-order accurate Runge-Kutta method. The partial differential equations governing the droplet internal thermal and mass transport are integrated by making use of a fully implicit Newton-Raphson iteration method.

Finally, the liquid-phase source terms of the gas-phase conservation equations (1-4) are evaluated by making use of a time-averaging method.

14 DESCRIPTION OF COMPUTER FLOWCHART, & THE TIME-AVERAGING SCHEME USED IN THE GAS-PHASE SOURCE-TERM CALCULATIONS

- In order to know more about the time-averaging method, we need to know first about the three different time steps that are internal to the spray code: Δt_{ml}, Δt_{il}, and Δt_{gl}.

  Δt_{ml} - the actual time step used in integrating the liquid-phase equations which is determined based on the smallest of the different time scales associated with various rate-controlling phenomena of a rapidly vaporizing droplet.

  Δt_{il} - the injection time step. It is the time step at which a new discretized parcel of different droplet groups are introduced into the computation.

  Δt_{gl} - the global time step. Its introduction seems to provide better convergence in both unsteady and steady-state computations.

- When the spray solver is called it advances the liquid phase equations over a number of iterations as determined by the ratio of $\Delta t_{gl}/\Delta t_{ml}$.

- It then evaluates the time-averaged contribution of the liquid-phase source terms, S_{gl}, of the gas-phase governing equations (1-4) as follows:

$$S_{gl} = \sum_{m=1}^{M} \frac{\Delta t_{ml}}{\Delta t_{gl}}\, S_{ml} \tag{93}$$

where

$$\sum_{m=1}^{M} \Delta t_{ml} \;=\; \Delta t_{gl} \tag{94}$$

- The values for Δt_{ml}, Δt_{gl}, & Δt_{il} are specified in the input file, ncc_liquid_solver.in, of Table 8.

- In steady-state computations, it is recommended to use for both Δt_{gl} and Δt_{il} a value of about 1 ms which is roughly equivalent to the average lifetime of the droplets for a typical reacting spray encountered in conventional low-pressure gas-turbine combustors.

The averaging scheme could be explained better through the use of a flow chart shown in Fig. 9. The main spray solver is invoked with a controlling routine, DCLR, which, then, executes the following steps:

1. It first initializes the source terms to zero.
2. Updates the global time, t_{gl}, based on Δt_{gl}.
3. Checks to see if $t_{ml} < t_{gl} < t_{ml} + \Delta t_{ml}$. If it is, it returns control over to the calling routine and supplies the other flow solvers, e.g., flow or EUPDF, with the source terms, S_{gl}, of Eqs. (1)-(4). If not, it proceeds with the next step.

Table 8. Input file: ncc_liquid_solver.in.

Input file content	**comments**
heading	title of controlling parameters
ldread,ispray_mod, (when_start_spray(n_i), n_i=1,no_of_injectors)	If ldread = .true., restarts the calculation from the data stored from the previous computation. Otherwise initiates a new spray computation. ispray_mod controls the calls to the spray solver. The spray solver is called once at every other CFD iteration as given by the number, ispray_mod. when_start_spray() represents the CFD iteration where the computations for the n_ith injector are initiated.
heading	title of controlling parameters
iswim, wall_film_thickness, offset, & coef_of_restitution	iswim = 0, default spray wall interaction model based on a single outcome of droplet moving along a wall, & = 1, new model based on four different possible outcomes. wall_film_thickness = 0, if dry wall, any positive number if wet wall. offset = offset fraction for boundaries. coef_of_restitution = coefficient of restitution at impact wall for drop.
heading	title of controlling parameters
hp_eosl	hp_eosl = 0, default (low pressure) spray model, & hp_eosl = 1, high pressure spray model as described in Sec. 9. This option requires the data contained in the file, ncc_hp_eos_vle_table.dat, as described in Section 9.3.
heading	title of controlling parameters
dtml, dtgl, & dtil	dtml = time step for advancing the liquid phase equations. dtgl = global time step. Whenever the spray solver is called, it advances the spray computations over a period of dtgl before returning control over to the calling routine. To be more precise, it advances the liquid phase equations over a number of time steps as determined by (dtgl/dtml). dtil = injection time step. It is where a new group of droplets are introduced into the computation.

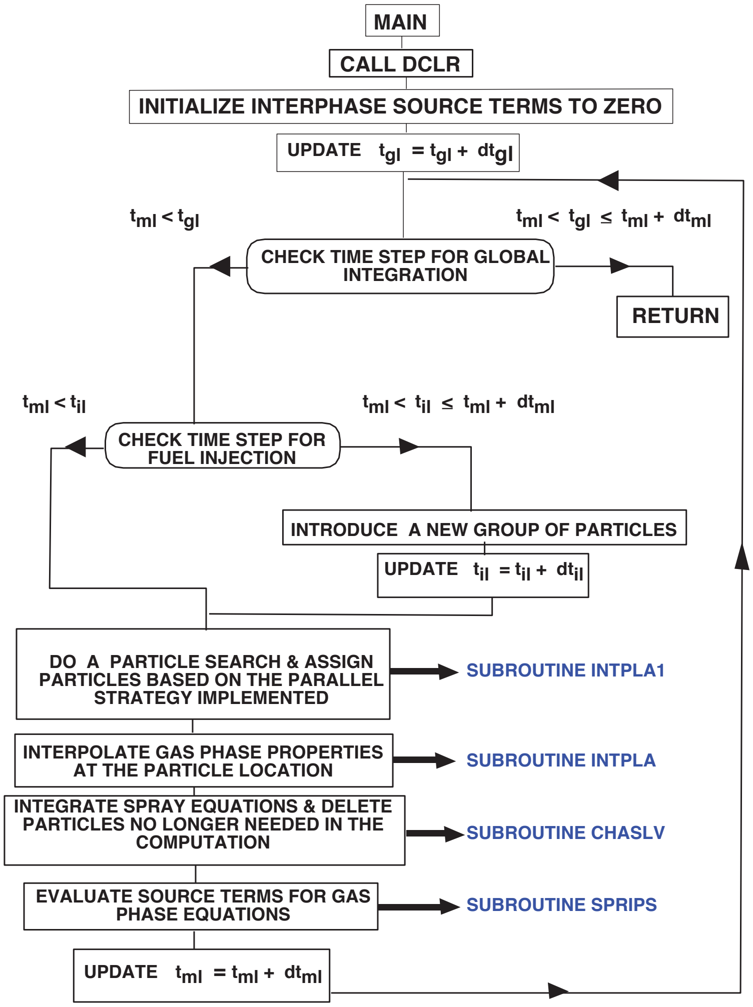

MAIN
CALL DCLR
INITIALIZE INTERPHASE SOURCE TERMS TO ZERO
UPDATE $t_{gl} = t_{gl} + dt_{gl}$
$t_{ml} < t_{gl}$
$t_{ml} < t_{gl} \leq t_{ml} + dt_{ml}$
CHECK TIME STEP FOR GLOBAL INTEGRATION
RETURN
$t_{ml} < t_{il}$
$t_{ml} < t_{il} \leq t_{ml} + dt_{ml}$
CHECK TIME STEP FOR FUEL INJECTION
INTRODUCE A NEW GROUP OF PARTICLES
UPDATE $t_{il} = t_{il} + dt_{il}$
DO A PARTICLE SEARCH & ASSIGN PARTICLES BASED ON THE PARALLEL STRATEGY IMPLEMENTED
SUBROUTINE INTPLA1
INTERPOLATE GAS PHASE PROPERTIES AT THE PARTICLE LOCATION
SUBROUTINE INTPLA
INTEGRATE SPRAY EQUATIONS & DELETE PARTICLES NO LONGER NEEDED IN THE COMPUTATION
SUBROUTINE CHASLV
EVALUATE SOURCE TERMS FOR GAS PHASE EQUATIONS
SUBROUTINE SPRIPS
UPDATE $t_{ml} = t_{ml} + dt_{ml}$

Fig. 9 The flowchart of LSPRAY-V.

4. Checks to see if it it is time to introduce a new group of particles.
5. Proceeds with solving the liquid-phase equations with calls to the following routines:
   - Calls the particle tracking routine and assigns particles based on the parallel strategy implemented.
   - Interpolates gas-phase properties at the particle location.
   - Advances liquid-phase equations and, then, deletes any particles that are no longer needed in the computations.
6. Evaluates the liquid-phase source-term contributions, S_{ml}, of Eq. 93.
7. Updates the time, t_{ml}, based on Δt_{ml}.
8. It then goes back to step (3) and repeats the whole process again until the computations are completed over a global time step: $\sum_{m=1}^{M} \Delta t_{ml} = \Delta t_{gl}$.

15 IMPLEMENTATION OF BOUNDARY CONDITIONS

The spray code supports all of the boundary conditions that exist in the current version of the NCC CFD module. First, we are going to provide a summary of the droplet and wall interaction models. There are several possible outcomes following a droplet impact with a wall. They can be broadly classified into four categories: (1) the droplet may move along a wall after its impact but still keeps on vaporizing, (2) the droplet may rebound after its impact with the wall, (3) the droplet may stick to the wall after its impact but keeps on vaporizing, or (4) the droplet may be shattered after its impact with the wall. The first outcome is the default boundary condition that was originally implemented into our code. It is based on the assumption that a droplet, after having lost most of its momentum upon impingement with the walls, moves along the wall surface with a velocity closer to that of the surrounding gas-film. The last three outcomes are based on the models described in Refs. [80-81]. They are implemented partially based on the coding received from CFDRC.

In order to implement the droplet and wall interaction models, there is a need to determine the intersection location of a particle crossing a wall, $\underline{x}_{bl,k}$. It can be determined based on the the known initial particle location, $\underline{x}_{ik}$, particle velocity, $\underline{V}_{ik}$, and other geometrical considerations of the grid cell. Based on vector analysis, we have determined it as follows

$$\underline{x}_{bl,k} = (x_{ik} + u^{\star} u_{ink}, y_{ik} + u^{\star} v_{ink}, z_{ik} + u^{\star} w_{ink}) \tag{95}$$

where

$$u^{\star} = \frac{\underline{a}_n \cdot (\underline{x}_{cwf} - \underline{x}_{ik})}{\underline{a}_n \cdot \underline{V}_{ink}},$$

$\underline{V}_{ink}$, is the velocity normal ($= \underline{V}_{ik}/|\underline{V}_{ik}|$), and $\underline{x}_{cwf}$ is the center location of the wall face of the grid cell. In what follows, we define various variables - Reynolds number, $Re_{sw,k}$, Weber number, $We_{sw,k}$, Ohnesorge number, $Oh_{sw,k}$, surface energy, E_{sw}, and viscous dissipation, V_{dis} - used in the spray-wall inraction modeling.

$$Re_{sw,k} = \frac{2\rho_k r_k |\underline{V}_{ik}|}{\mu_k} \tag{96}$$

$$We_{sw,k} = \frac{2\rho_k r_k |\underline{V}_{ik}|^2}{\sigma_k} \tag{97}$$

$$Oh_{sw,k} = \frac{\sqrt{We_{sw,k}}}{Re_{sw,k}} \tag{98}$$

$$E_{sw} = \pi \sigma_k (2 r_k \beta_{max})^2 (1 - cos\ \alpha)/4 \tag{99}$$

where

$$\beta_{max} = \frac{\sqrt{12 + We_{sw,k}}}{3(1 - cos\ \alpha)} + \frac{4We_{sw,k}}{\sqrt{Re_{sw,k}}},$$

the droplet impact angle, α, is given by

$$\alpha = \frac{\pi}{2} - cos^{-1}\ \gamma,$$

and the droplet frequency, γ, is

$$\gamma = \frac{(\underline{x}_{bl,k} - \underline{x}_{ik}) \cdot \underline{a}_n}{2 r_k},$$

and

$$V_{dis} = \frac{2\pi r_k \rho_k |\underline{V}_{ik}|^2 (2 r_k \beta_{max})^2}{3\sqrt{Re_{sw,k}}} \tag{100}$$

The droplet outcome after the interaction with the wall is determined based on the following criteria:

Shattering Droplet:

A droplet shatters upon impact with a wall when $We_{sw,k} > We_{crit}$, where the critical Weber number is defined as follows

$$We_{crit} \;=\; 9.9^{10}\, Oh^{2.8} \qquad (101)$$

The average size of the shatterred droplet size after impact is given by

$$r_{ave,shat} \;=\; max(r_{l1,shat}\;, r_{l2,shat}) \qquad (102)$$

where $r_{l1,shat}$ and $r_{l2,shat}$ represent the two limits of droplet sizes as determined by

$$r_{l1,shat} \;=\; \frac{\sigma_k We_{crit}}{2\rho_k|\underline{V}_{ik}|^2} \qquad (103)$$

$$r_{l2,shat} \;=\; \frac{4\pi r_k^3 \sigma_k}{E_{sw}} \qquad (104)$$

Once $r_{ave,shat}$ is determined, the total number of shattered droplets produced is given by

$$N_{shat} \;=\; int[(\frac{r_k}{r_{ave,shat}})^3] \qquad (105)$$

However, it is important to note that the number of shattered droplets generated is limited to 2 as shown below

$$N_{shat} \;=\; min(2\;,\; int[(\frac{r_k}{r_{ave,shat}})^3]) \qquad (106)$$

The actual size of the shattered droplets is calculated by introduing some randomness as shown below

$$r_{shat} \;=\; r_{ave,shat}(0.85 + 0.30RND) \qquad (107)$$

where RND is a random number between (0,1). The droplet position and velocity of the shattered droplets is determined by the following criteria.

Upon impact with a wall, a shattered droplet emerges from the wall in a direction determined by the modified reflection normal vector, $\underline{x}_{rnk}$.

$$\underline{x}_{rnk} \;=\; 2(0.5 - RND)(\underline{x}_{ink} \;-\; C_{res}(\underline{x}_{ink}{\cdot}\underline{a}_n)\underline{a}_n) \qquad (108)$$

where C_{res} stands for coefficient of restitution. Its value depends on the properties of the wall but has a value of 2 under normal reflecting conditions. Also, RND is added to introduce some randomness into the direction of reflection vector. Once $\underline{x}_{rnk}$ is determined, the new droplet position and velocity are determined.

$$If\;\; \underline{x}_{rnk}{\cdot}\underline{a}_n \;>\; 0,$$

$$\underline{V}_{nk} \;=\; |\underline{V}_{ik}|\underline{x}_{rnk} \qquad (109)$$

$$\underline{x}_{nk} \;=\; \underline{x}_{bl,k} + |\underline{x}_{bl,k} - \underline{x}_{ik}|\underline{x}_{rnk} + C_{off}\underline{a}_n \qquad (110)$$

where C_{off} stands for offset and is usually assigned a tiny value but in our present calculations it is assigned a value of $0.01|x_{cwf} - x_{cgc}|$, where x_{cgc} is the center location of grid cell.

$$but\;\; if\;\; \underline{x}_{rnk}{\cdot}\underline{a}_n \;\leq\; 0,$$

$$\underline{V}_{nk} \;=\; 10^{-06}\underline{a}_n \qquad (111)$$

$$\underline{x}_{nk} \;=\; \underline{x}_{bl,k} + C_{off}\underline{a}_n \qquad (112)$$

Once the droplet size, location, and velocity of the shattered droplets are determined, the rest of the droplet properties can easily be deduced from the initial conditions of incident droplet.

Sticking Droplet to the Wall:

A droplet sticks to the wall when $E_{sw} > V_{dis}$. After being stuck to the wall, the impacted droplet size remains the same but continue to vaporize in time. The droplet velocity and position are assigned the following values:

$$\underline{V}_{nk} \;=\; 10^{-06}\underline{a}_n \qquad (113)$$

$$\underline{x}_{nk} \;=\; \underline{x}_{bl,k} + C_{off}\underline{a}_n \qquad (114)$$

The rest of the droplet properties are deduced from the initial conditions of incident droplet.

Rebounding Droplet:

If none of the above conditions are met for the outcomes of either droplet shattering or droplet sticking to a wall, the droplet is assumed to rebound after its impact with a wall. After the rebound, the droplet size remains intact as in the incident droplet. Both the droplet velocity and position are calculated as in the droplet shattering but randomness is removed from the reflection vector. So it is calculated as in Eqs. (109-112) but RND is assigned a value of zero. The rest of the droplet properties are deduced from the initial conditions of incident droplet.

Moving Droplet Along the Wall:

For those droplets that were produced as a result of droplet shattering, they are assumed to move along next to the wall surface after its subsequent impact with a wall. Such a droplet would experience no change in its size after its impact but it is allowed to keep on vaporizing as it moves along the wall surface. The droplet velocity and position are calculated as follows

$$\underline{V}_{nk} = \underline{V}_{gs} \tag{115}$$

$$\underline{x}_{nk} = \underline{x}_{bl,k} + C_{off}\underline{a}_n \tag{116}$$

where $\underline{V}_{gs}$ is the velocity of the the surrounding gas.

Finally, in what follows we highlight how some other boundary conditions are implemented:

- The implementation of the periodic boundary conditions becomes rather complicated. It is because one needs to keep track of a particle leaving the computational domain from one periodic boundary, and for every particle that leaves the domain, a second particle reenters the computational domain through a corresponding second periodic boundary. Also, one needs to take into account the possibility of the two computational cells where the particles leave and reenter the domain being assigned to a different processor. We incorporated the periodic implementation as a part of the particle search algorithm. It is important to note that the boundary conditions are implemented with the help of some appropriately defined transformation matrices. Also, the particle velocities of the entering particle need to be adjusted accordingly based on the orientation of respective periodic boundaries.

- The symmetric boundary condition is implemented in such a way to satisfy the criterion that for every particle crossing the symmetry line, a similar one re-enters the domain in a direction determined by the reflection vector.

- When the particles move out of the exit boundary, they are taken out of the computation.

16 DETAILS OF COUPLING BETWEEN LSPRAY-V AND OTHER SOLVERS

- The spray code is designed to be a stand-alone module such that it could easily be coupled with any other unstructured-grid CFD solver and the same holds true for EUPDF. (However, some grid-related parameters such as area vectors, grid connectivity parameters, etc. need to be supplied separately).

- The spray solver needs information on the gas velocity and scalar fields from the other solvers and, then, it in turn supplies the liquid-phase source terms.

- The PDF solver needs information on the mean gas velocity, turbulent diffusivity and frequency from the CFD solver and the liquid-phase source terms from the spray solver, and then it in turn provides the solution for the scalar (species and energy) fields to the flow and spray solvers.

- It should also be noted that both the PDF and spray solvers are called once at every other specified number of CFD iterations.

- All of the three solvers (LSPRAY-V, EUPDF, and CFD) are advanced sequentially in an iterative manner until a converged solution is obtained.

- All three codes (EUPDF, CFD, and LSPRAY-V) were coupled and parallelized in such a way to achieve maximum efficiency.

The coupling issues could be better understood through the use of a flow chart shown in Fig. 10. It shows the overall flow structure of the combined CFD, LSPRAY-V, and EUPDF modules. Both the PDF and spray codes are loosely coupled with the CFD code. The spray code is designed in such a way that only a minimal amount of effort is needed for its coupling with the flow and PDF solvers. The present version of the spray module relies entirely on the use of Fortran common blocks for its information exchange with other modules. Even this reliance should entail only few changes to be made within the

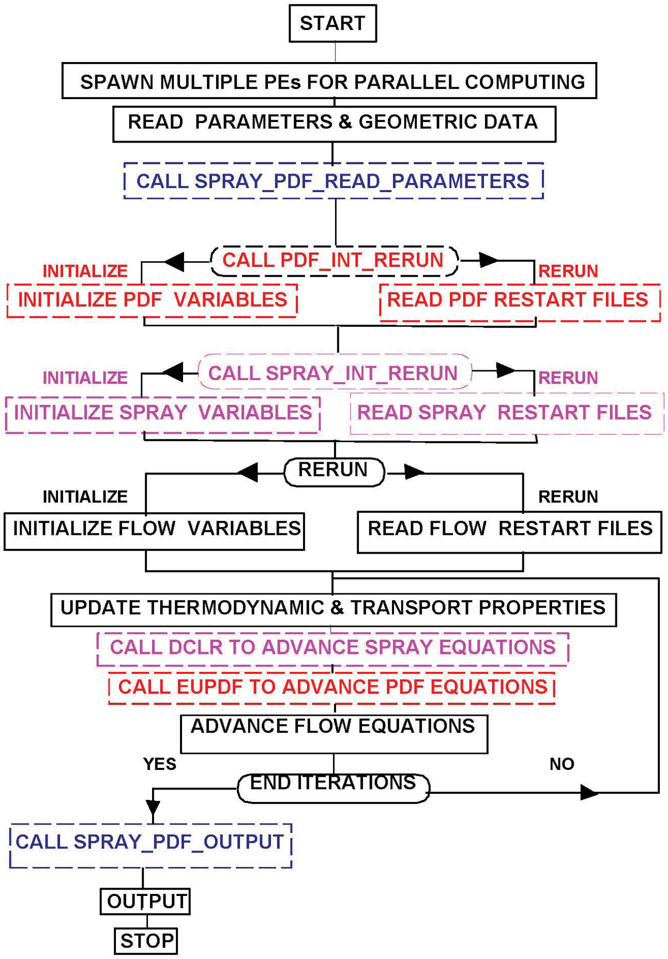

START
SPAWN MULTIPLE PEs FOR PARALLEL COMPUTING
READ PARAMETERS & GEOMETRIC DATA
CALL SPRAY_PDF_READ_PARAMETERS
CALL PDF_INT_RERUN
INITIALIZE
RERUN
INITIALIZE PDF VARIABLES
READ PDF RESTART FILES
CALL SPRAY_INT_RERUN
INITIALIZE
RERUN
INITIALIZE SPRAY VARIABLES
READ SPRAY RESTART FILES
RERUN
INITIALIZE
RERUN
INITIALIZE FLOW VARIABLES
READ FLOW RESTART FILES
UPDATE THERMODYNAMIC & TRANSPORT PROPERTIES
CALL DCLR TO ADVANCE SPRAY EQUATIONS
CALL EUPDF TO ADVANCE PDF EQUATIONS
ADVANCE FLOW EQUATIONS
YES
NO
END ITERATIONS
CALL SPRAY_PDF_OUTPUT
OUTPUT
STOP

Fig. 10 The overall flowchart of the combined CFD, LSPRAY, and EUPDF solvers.

spray code for its linkage with different solvers. The PDF code is also structured along similar coupling principles.

The flow chart of Fig. 10 contains several blocks - some shown in black and/or solid lines and the others in color and/or dashed lines. The ones in solid blocks represent the flow chart that is typical of a CFD solver. The ones in dashed blocks represent the additions arising from the coupling of the spray and PDF solvers.

The coupling starts with the calling of the subroutine - **spray_pdf_read_parameters**, which then reads the spray control parameters from the input file, *ncc_liquid_solver.in* of Table 8. This table provides a detailed description of the following input file variables - ldread, ispray_mod, (when_start_spray(n_i), n_i=1,no_of_injectors), iswim, wall_film_thickness, offset, coef_of_restitution, hp_eosl, dtml, dtgl, & dtil. The coupling is then followed by the calling of the **pdf_int_rerun** subroutine. It initializes PDF computations and also it may restart the PDF computations if needed from the data stored from a previous iteration. Similarly, we call **spray_int_rerun** for the spray computations. It is noteworthy that the spray computations can be restarted by reading the data from the restart files: *ncc_liquid_params.out*, *ncc_liquid_results.db*, & *ncc_liquid_results_mc.db*. The restart capability is invoked by setting the logical variable, ldread, of the input file, *ncc_liquid_solver.in* of Table 8, to be true. Otherwise, the spray computations are initialized to start from the beginning. Then, the coupling proceeds with the calling of the following subroutines: **dclr** for integrating the spray calculations and **eupdf** for the Monte Carlo PDF. The input variable, ispray_mod of Table 8, controls the calls to the spray integrating routine. The spray solver is called once at every other number of CFD iterations as specified by ispray_mod. And the first call to the spray solver is controlled by the input variable, when_start_spray() of Table 8, which represents the starting CFD iteration number from which the spray computations are initiated. Finally, the coupling ends with the calling of a subroutine, **spray_pdf_output**, which will create a set of new restart files.

17 DETAILS OF FORTRAN SUBROUTINES & FUNCTIONS

Table 9 provides a list of all the Fortran subroutines developed as a part of the spray module. This table also provides information on all the Fortran functions. It also describes the purpose of all the individual subroutines and functions. Similary, Table 10 provides a list of all the Fortran functions and subroutines associated with the implementation of the high pressure equation of state as described in Sec. 3, and the high pressure behavior on the gas-phase transport properties as in Sec.4.

18 DETAILS OF PARALLELIZATION

There are several issues associated with the parallelization of both the spray & PDF computations. The goal of the parallel implementation is to extract maximum parallelism so as to minimize the execution time for a given application on a specified number of processors [37]. Several types of overhead costs are associated with parallel implementation which include data dependency, communication, load imbalance, arithmetic, and memory overheads. The term arithmetic overhead is the extra arithmetic operations required by the parallel implementation. Memory overhead refers to the extra memory needed. Excessive memory overhead reduces the size of a problem that can be run on a given system and the other overheads result in performance degradation [37]. Any given application usually consists of several different phases that must be performed in certain sequential order. The degree of parallelism and data dependencies associated with each of the subtasks can vary widely [37]. The goal is to achieve maximum efficiency with a reasonable programming effort [37].

In our earlier work, we discussed the parallel implementation of a spray algorithm developed for the structured grid calculations on a Cray T3D [10]. These computations were performed in conjunction the Monte Carlo PDF method. The parallel algorithm made use of the shared memory constructs exclusive to Cray MPP (Massively Parallel Processing) Fortran and the computations showed a reasonable degree of parallel performance when they were performed on a NASA LeRC Cray T3D with the number of processors ranging between 8 to 32 [10]. Later on, the extension of this method to unstructured grids and parallel computing written in Fortran 77 with PVM or MPI calls was reported in [11-15]. The latest version in Fortran 77/90 offers greater computer platform independence. In this section, we only highlight some important aspects of parallelization from Refs. [10-15].

Table 9. Description of LSPRAY-V Fortran subroutines & functions.

Function	**Purpose of the Function**
blasiu(x)	This function returns a solution for the function, $f(B_k)$, of Eq. (21) for use in computing the droplet regression rate.
Subroutine	**Purpose of the Subroutines**
axisym_part_adjust	This routine was written by Jeff Moder and essentially performs the same functions as the intpla routine but modified for application with a 2d-axisymmetric case.
calc_F_suvw()	It evaluates the right-hand sides of ODEs for both droplet size and velocities, and then provides the solution to the calling routine, chaslv.
chaslv	This routine has two main functions: (1) It integrates the liquid phase equations. (2) It removes the particles that are no longer needed in the computation.
dclr	This routine is called once at every other CFD iteration as specified by ispray_mod. It is primarily a controlling routine for spray computations. This routine has the following functions: (1) It initializes the source terms to zero. (2) Checks to see if new particles need to be introduced. (3) Advances liquid phase equations over an allowable or pre-specified time step, dtml, with calls to the following routines: intpla1 - Interpolates the gas phase properties at the particle location. chaslv - Advances liquid phase equations. intpla - Identifies computational cells and PEs associated with spray particles. sprips - Evaluates the liquid phase source term contributions of the CFD and PDF equations. (4) (a) For unsteady spray computations (steady_spray_model=F), this routine is called once at the beginning of every global time-step, dtgl. This is accomplished by continuing

Table 9. Description of LSPRAY-V Fortran subroutines & functions (continued).

Subroutine	**Purpose of the Subroutines**
	with steps (2) and (3) until the computations are completed over one single global time step, dtgl. (b) For steady spray spray computations (steady_spray_model=T), this routine is only called once to solve entire lifetime of all particles injected. It repeats step (3) until there are no more spray particles left to be integrated (note: The particles are taken of computation when they reach either a certain size of negligible proportion or exit out of the computational domain). (5) Returns control over to other solvers, e.g. CFD and EUPDF, and supply them with the source terms averaged over dtgl.
dropdis(rhol, flowdum,sr, fld,smd,nofg)	This routine provides initial droplet distribution from the following correlation (also, appears as Eq. 91): $dn/n = a((D/D_{32})^{alp})exp(-b((D/D_{32})^{bet}))dD/D_{32}$ where a, b, alp, and bet are constants.
drop_ic(n_i, nmih,nmis, nmip)	It is called by the spray_main_injection routine in conjunction with spray_table(n_i))=F at the beginning of every injection time step, dtil, for introducing a new group of spray particles. It applies for the introduction of new particles in conjunction with Options (2) or (3) of Sec. 9. In Opt. (2) the initial conditions are specified from the known correlations and in Opt. (3) from the available primary atomization models.
find_transport_ds(ijle,chem_model, number_of_species,hp_eos, temp_gas,pres_gas,y_gas, rhogm,visgm,congm,difgm)	It computes the following properties of a gas mixture at the droplet interface by making use of the one-third rule of Eq. (46): molecular viscosity, gas density, thermal conductivity, and diffusion coefficient.
find_xyzface (i,xcfac,ycfac, zcfac)	This routine computes x, y, and z locations of all the face centers of an element, i. This information is used in the particle search algorithm.
get_liq_tat_properties (tmp,pmp,j)	It computes the following variable properties of a liquid mixture: density, specific heat, molecular viscosity, gas density, thermal conductivity, and diffusion coefficient. This routine also computes the surface tension of a multi-component liquid fuel.
get_liq_tat_properties1 (tmp,pmp,boil_tem, cp_sph,hvap_sph,icuin)	This subroutine is used in conjunction with the superheat vaporization model and computes the following variable properties of the liquid fuel: boiling temperature, specific heat at constant pressure, surface tension, heat of vaporization, molecular viscosity, & liquid density and volume.
get_liq_tat_properties_sp (tmp1,tmp2,cplt1, cplt2,j)	This subroutine is used in conjunction with the superheat vaporization model and computes the following variable properties of the liquid fuel: specific heat at droplet temperature, & specific heat at boiling temperature.

Table 9. Description of LSPRAY-V Fortran subroutines & functions (continued).

Subroutine	**Purpose of the Subroutines**
init_psi2_dsd	This routine provides initial droplet distribution from a clipped probability distribution function as given by Eq. 92.
intpla	This routine performs three main functions: (1) Particle Tracking - It identifies the computational cell in which a particle is located. In parallel computing, it also means identifying the corresponding processor of the computational domain in which a particle is located. (2) It implements appropriate boundary conditions. (3) It reassigns the particles between different PEs based on the parallel strategy employed.
intpla1	This routine interpolates the gas-phase properties at the particle location. In the present case, a simple first-order interpolation is employed.
mimd_spray	For computational elements where the neighboring cells are partitioned between different processors, this routine initializes the arrays, ipr_fr_id() and ile_fr_id(), by storing the information on the corresponding processor and element ID numbers of all the neighboring cells. This information is needed in order to track the particle movement between neighboring cells.
mimd_spray_recv (i_recvfrom)	This subroutine is called by mimd_spray in order to gather some relevant information from the neighboring processors.
mimd_spray_send (i_sendto)	This subroutine is called by mimd_spray in order to send some relevant information to the neighboring processors.
par_loc(xparz, yparz,zparz, ipare,iparp)	Given the x,y,z coordinates of a particle location, the algorithm identifies the corresponding computational cell in which a particle is located. It also identifies the corresponding processor of the computational grid in which a particle is located. This information is used to locate newly injected droplet groups.
prnspr	It provides written output of the spray data in a standard format.
spray_int_rerun	This routine has two functions: (1) It initializes the spray solver based on the data read from various input files. (2) If it is a rerun, it restarts the computations from the stored data of a previous computation.

Table 9. Description of LSPRAY-V Fortran subroutines & functions (continued).

Subroutine	**Purpose of the Subroutines**
spray_main_injection()	This routine is called to inject a new group of spray particles for each injector at the beginning of every injection time step, dtil.
spray_output	This routine is called to create appropriate restart files for the spray computations and also to provide for debugging purposes written output of some useful gas and spray data in a standard format.
spray_plot_output	It provides written output of some useful spray data in a standard format. It can be used in conjunction with steady_spray_model = F.
spray_read_parameters (ipid,ncells))	This routine is called once in the beginning to read and initialize some of controlling parameters associated with the spray computations.
spray_read_transport ((ipid,fid,filename)	This routine is called once to read liquid thermo and tranport data: various constants used in the evaluation the polynomials in temperature for constant pressure specific heat, thermal conductivity, and viscosity.
sprips	This routine is called to provide the liquid-phase source terms of Eqs. (1)-(4) for use in both CFD and Monte Carlo PDF solvers: smlc(i) = liquid-phase contribution of Eq. (1) of Section 2. smlmx(i), smlmy(i), smlmz(i) = liquid-phase contribution of Eq. (3) of Section 2. smle(i) = liquid-phase contribution of Eq. (4) of Section 2.
sy(il,iu,bb, dd,aa,cc)	It is a tri-diagonal matrix solver. It is used in the solution of both Eqs. (27) & (37).
uvw_par(swlr_angle(nx), angle_work,nx,ny,nn, v_inj,nmip,t_rotation, cone,n_cone_rays, cone_rotation,uloc, vloc,wloc,n_l)	It computes the initial particle injection velocity for different geometric configurations of spray as shown in Figs. 2 to 4.

Table 10. Description of PENG_ROB Fortran subroutines & functions.

Function	**Purpose of the Function**
peng_rob_density (chem_model,ycomp, tij,pressure, number_of_species)	It provides a solution for the density of a multi-component mixture based on the Peng-Robinson equation of state.
peng_rob_pressure (chem_model,ycomp, tij,density, number_of_species)	It provides a solution for the pressure of a multi-component mixture based on the Peng-Robinson equation of state.
Subroutine	**Purpose of the Subroutines**
hp_diff_table (ct,pt,dabcvi)	This subroutine provides a value for $(D_{AB}\ P)/(D_{AB}\ P)^{+}$ based on the Takahashi high-pressure correlation for the calculation of the diffusion coefficients in a gas mixture. It is based on the interpolation from the data derived from a graphical illustration of Reid et al [16], "The Properties Gases and Liquids." The retrieved data is stored in hp_diff_table.dat as a table of f(tpr_diff,ppr_diff) values.
hp_diffusion (tij,pij,its, nwarn,diin_hp)	It is called to evaluate the gas-phase mixture-averaged diffusion coefficients valid at high pressures.
hp_transport (calc_name,its, tij,pij,nwarn, avisi_hp,condi_hp)	It is called to evaluate the gas-phase transport properties valid at high pressures: viscosity and conductivity.
peng_rob_all_rhos(chem_model,ycomp, tij,pressure, number_of_species, rho1, rho2, rho3, vv1, vv2, vv3, z1, z2, z3, aprgm,bprgm)	It provides the roots for the cubic equation, Eq. (6), involving the compressibility factor, Z. One of the three roots of Eq. (6) yields Z_v and the other Z_l. Jeff Moder made significant improvements to this routine.
peng_rob_gen (xsp,ppr,tpr, its,rho0)	It takes the following as the input variables: xsp() - the mass fraction of the species, ppr - pressure, tpr - temperature, and its - the number of species. And it returns as the output, rho0 - density based on the solution of the Peng-Robinson EOS.
peng_rob_zcrit_rhor (ppr,tpr,its)	It provides a solution for Z_{ic} and ρ_{ir} for ith species based on the solution of the Peng-Robinson EOS for a given value of P_{ir} and T_{ir}. It also requires the specification of the total number of species, its.

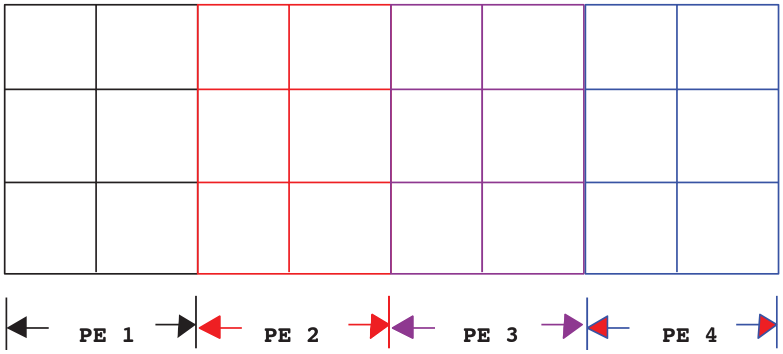


Fig. 11a An illustration of the parallelization strategy employed in the gas flow computations.

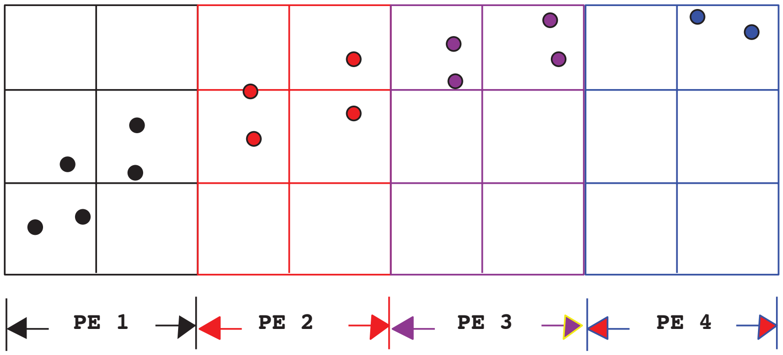


Fig. 11b An illustration of the parallelization strategy employed in the spray computations.

Table 11. CPU time (sec) per cycle versus number of PEs.

		Number of processors		
Solver	Characteristic	2	5	10
CFD	5 steps/cycle	2.50	1.25	0.75
EUPDF	1 step/cycle	6.5	2.9	1.9
LSPRAY	100 steps/cycle	1.70	0.64	0.53

Table 12. CPU Time (Sec) Per Cycle Versus Number of PEs.

		Number of Processors				
Solver	Characteristic	1	2	4	8	16
Spray	100 steps/cycle	6.83	5.29	2.94	1.64	0.87
Max. Spray Particles in a PE		2695	2097	1165	623	312
Min. Spray Particles in a PE		2695	598	118	14	0

- Both the EUPDF and CFD modules are well suited for parallel implementation. For the gas-phase computations, the domain of computation is simply divided into n-Parts of nearly equal size and each part is solved by a different processor. Fig. 11a illustrates a simple example of the domain decomposition strategy adopted for the gas-phase computations where the total domain is simply divided equally amongst the available computer processing elements (PEs). In this case, we assumed the number of available PEs to be equal to four.

- But the spray computations are more difficult to parallelize for the reasons summarized below:

  (1) Non-uniform nature of spray distribution: Most of the particles are usually confined to a small region where the atomizer is located.

  (2) Dynamic nature of Lagrangian particles: Particles keep moving between different subdivided domains of an Eulerian grid (grid used in the CFD computation) which are assigned to different processors. While some new particles are introduced at the time of fuel injection, some others are taken out of computation.

Conceptually, there are several ways to parallelize the spray computations, we, however, developed and tested two different domain decomposition strategies [10-14].

- Strategy I:

  The Lagrangian particles were assigned fairly uniform amongst the available processors but the calculations associated with the particle tracking, the interpolation of the gas-phase properties, and the source-term evaluation were computed on the processor of the computational grid in which a particle is located.

  This strategy leads to an uniform loading during integration but leads to excessive message passing.

- Strategy II:

  The Lagrangian particles were assigned to the processor of the computational grid where the particle is located. Fig. 11b illustrates a simple example of the domain decomposition strategy adopted for the liquid-phase computations where the corresponding gas-flow computational domain is divided into equal parts between the four available PEs. In this strategy, the Lagrangian particles are assigned to the processor of the computational grid where a particle is located.

This strategy lead to a non-uniform loading during integration but leads to less message passing.

Our experience has shown that Strategy II seems to work well on different computer platforms: both massively parallel computers as well as heterogeneous cluster of workstations. So in the present version of the code, we have opted to implement Strategy II over Strategy I.

19 DETAILS OF PARALLEL PERFORMANCE

The details of the combined parallel performance of the CFD, EUPDF, and LSPRAY codes involving several different cases can be found in Refs. [10-15]. Here, we only summarize briefly the parallel-performance results for two different cases. One is a 3D test case and more details on this case can be found in the reference [13]. For this case, the calculations were performed on a computational grid comprising of 8430 tetrahedral elements and 100 Monte Carlo PDF particles per cell. The computations were performed on one of the NASA Ames Research Center's parallel computer platforms called Turing which is a SGI Origin work-station with 24 PEs (Processor Elements). Table 11 summarizes the CPU times per cycle taken by the EUPDF, LSPRAY, and CFD solvers vs the number of PEs. Both the CFD and PDF solvers show good parallel performance with an increase in the number of processors but for the spray solver it shows reasonable parallel performance.

Next, we would like to summarize the results from [13, 38] showing only the results of spray computations. The results are summarized in Table 12. The computations were performed on Turing at NASA ARC (Ames Research Center) - it is a SGI Origin work-station with a maximum of 24 PEs (Processor Elements). The computations made use of an unstructured grid with a mesh size of 3600 tetrahedral elements. And it made use of about 2,695 Lagrangian particles for the spray computations and one hundred Monte Carlo particles per element for the PDF computations. In a given cycle of one global time step, dt_{gl}, the spray equations were advanced over one hundred time steps as given by $dt_{gl}/dt_{ml} = 100$. The first row of Table 12 summarizes the CPU times per PE per cycle taken by the spray solver vs the number of PEs. As expected, the CPU time goes down with an increase in the number of processors. In an ideal case, one would expect an inverse reduction in cpu time with an increase in the number of processors. Here, we don't get such an ideal performance because of the resulting non-uniform distribution of spray particles, between various participating processors, from the implementation of Stratergy II. To get an idea of the spray particle distribution, we have tabulated the maximum and minimum number of particles found between various processors. When we go from 1 to 2 PEs, 2097 particles are assigned to one and the rest to the second. With four they are distributed between 1165 and 118, with eight between 623 and 14, and with sixteen from 312 to 0. The results clearly show that the reduction in the CPU time varies almost linearly with the reduction in the number of maximum particles.

20 SUMMARY OF SOME VALIDATION CASES INVOLVING BOTH REACTING AND NON-REACTING SPRAY COMPUTATIONS

A total of the following nine cases were validated:

1. A reacting methanol spray with no-swirl,
2. A non-reacting methanol spray with no-swirl,
3. A confined swirl-stabilized n-heptane reacting spray,
4. An unconfined swirl-stabilized n-heptane reacting spray,
5. A confined swirl-stabilized kerosene reacting spray,
6. A flashing jet generated by the sudden release of pressurized R134A from cylindrical nozzle,
7. A liquid jet atomizing in a subsonic cross flow,
8. A Parker-Hannifin pressure swirl atomizer, &
9. A single-element LDI (Lean Direct Injector) combustor experiment.

The experimental data for the first two cases was provided by McDonell & Samuelsen from the University of California at Irvine [38]. Both the cases are without swirl; one is a reacting case and the other is non-reacting. The data for the third and fourth cases was provided by Bulzan from the NASA Glenn

Research Center [39-40]. Both the cases are swirl-stabilized reacting cases, one is an unconfined flame and the other is confined. The data for the fifth case was provided by El Banhawy & Whitelaw from Imperial College [4]. It is a confined swirl-stabilized kerosene spray flame. Further details of the last four cases can be found in [78]. These cases were chosen because of their importance in some aerospace applications.

Here, we provide a brief summary of our validation effort but a detailed presentation of the results and discussion can be found elsewhere in the papers [10-13, 78]. The validation is based on some 3D and axisymmetric calculations involving both reacting and non-reacting sprays. Some of them were performed on unstructured grids and the others on structured grids.

The comparisons involved both gas and drop velocities, drop size distributions, drop spreading rates, and gas temperatures. In general, the predicted results provide reasonable agreement for both mean droplet sizes (D_{32}) and average droplet velocities but mostly underestimate the droplets sizes in the inner radial region of a cylindrical jet.

The comparisons also involved the results obtained from the use of the Monte Carlo PDF method as well as those obtained from a conventional CFD solution without the Monte Carlo PDF method. For the first case of McDonell & Samuelsen's reacting spray flame, the detailed comparisons clearly highlighted the importance of chemistry/turbulence interactions in the modeling of reacting sprays [13]. The results from the PDF and non-PDF methods were found to be markedly different with the PDF solution providing a better approximation to the reported experimental data. The PDF solution showed that most of the combustion occurred in a predominantly diffusion-type of flame environment and the rest occurring in a predominantly premixed-type of flame environment. However, the non-PDF predictions showed incorrectly that most of the combustion occurred in a predominantly vaporization-controlled regime. The Monte Carlo temperature distribution showed that the functional form of the PDF for the temperature fluctuations varied substantially from point to point. The results brought to the fore some of the deficiencies associated with the use of assumed-shape PDF methods in spray computations.

21 CONCLUDING REMARKS

- This manual provides a complete description of LSPRAY-V - a Lagrangian spray solver developed for application with parallel computing and unstructured grids.

- It facilitates the calculation of the multi-component liquid sprays with variable properties valid over a wide range of pressure conditions encountered in gas-turbine engines.

- It provides the user with a basic understanding of the the spray formulation and the LSPRAY-V code structure, and complete details on how to couple the spray code to any other flow code.

- The basic structure adopted for the grid representation and parallelization for the gas side of the flow computations follows the guidelines established for NCC.

- Also, we have extended the joint scalar Monte Carlo PDF method to two-phase flows and, thereby, demonstrating the importance of chemistry/turbulence interactions in the modeling of reacting sprays.

- Based on the validation studies involving several confined and unconfined spray flames, the results were found to be encouraging in terms of their ability to capture the overall structure of a spray flame.

- The source code of LSPRAY-V will be available with OpenNCC as a complete package.

APPENDIX

DETAILS OF THE PRIMARY ATOMIZATION MODELS

1. Sheet Breakup Primary Atomization Model (Likely Application: Pressure Swirl Atomizer)

Here, we summarize the details of the sheet breakup model taken from Schmidt et al [48]. The growth of an infinitesimal disturbance as given by

$$\eta \ = \ \eta_o \ exp(ikx \ + \ \omega t) \tag{1}$$

was analyzed based on a linear stability analysis of a two-dimensional, viscous, incompressible liquid sheet of thickness $2h$ which moves through an inviscid, incompressible gas medium. This was analyzed in a coordinate system moving with the sheet with a relative velocity of U where η_o is the initial wave amplitude, k $(= 2\pi/\lambda)$ is the wave number, and $\omega \ = \ \omega_r \ + \ i\omega_i$ is the complex growth rate. The most unstable disturbance responsible for the sheet breakup is denoted by Ω.

Based on the linearized liquid and gas continuity and momentum equations subject to the linearized boundary conditions at the gas and liquid interfaces, a sinuous mode dispersion relation was obtained by [49],

$$\begin{aligned} &\omega^2[tanh(kh) + Q] + \omega[4\nu_l k^2 tanh(kh) + 2iQkU] + \\ &4\nu_l^2 k^4 tanh(kh) - 4\nu_l^2 k^3 l \ tanh(lk) - QU^2k^2 + \tfrac{\sigma k^3}{\rho_l} \quad = 0(2) \end{aligned}$$

where $Q \ = \ \rho_g/\rho_l$, $l^2 \ = \ k^2 \ + \ \omega/\nu_l$, and U is the relative velocity between the liquid and gas. Inviscid analysis also indicates that for low gas Weber number ($We = \rho_g h U^2/\sigma$) flows, long waves tend to grow leading to liquid sheet breakup but for higher Weber numbers, short waves produce a maximum growth rate followed by breakup. The critical Weber number that leads to the transition from the long wavelength regime to the short wavelength regime was shown to be $We_g \ = \ 27/16$. For most modern fuel injection systems, the film Weber number is well above this critical limit. The growth rate for the sinuous mode, ω_r, based on an order of magnitude analysis of the dispersion relation yields,

$$\omega_r = -\frac{2\nu_l k^2 tanh(kh)}{tanh(kh) + Q} + \sqrt{\frac{4\nu_l^2 k^4 tanh^2(kh) - Q^2U^2k^2 - [tanh(kh) + Q](-QU^2k^2 + \sigma k^3/\rho_l)}{tanh(kh) + Q}} \tag{3}$$

For short waves in the limit of $Q << 1$ for high-speed sheets, it yields

$$\omega_r \ = \ -2\nu_l k^2 + \sqrt{4\nu_l^2 k^4 + QU^2k^2 - \frac{\sigma k^3}{\rho_l}} \tag{4}$$

Following Dombrowski & Johns [51], the sheet disintegration leads to the formation of ligaments once the unstable waves reach a critical amplitude and Eq. (4) shows that the growth rate of short waves is independent of the sheet thickness. The corresponding breakup time τ and the breakup length L are given by:

$$\eta_b \ = \ \eta_o \ exp(\Omega\tau) \ \ => \ \ \tau \ = \ \frac{1}{\Omega} ln \frac{\eta_b}{\eta_o} \tag{5}$$

$$L \ = \ V\tau \ = \ \frac{V}{\Omega} ln(\frac{\eta_b}{\eta_o}) \tag{6}$$

where Ω is the maximum growth rate as determined by Eq. (4), the term $ln(\frac{\eta_b}{\eta_o})$ has an assigned value of 12 as suggested by Dombrowski & Hooper [50], and V is the absolute velocity of the liquid.

The initial diameter of the ligaments is derived from a mass balance relationship. For long waves, it is assumed that the ligaments are formed from tears in the sheet once per wavelength and the resulting diameter is given by,

$$d_L \ = \ \sqrt{\frac{8h}{K_{sb}}} \tag{7}$$

where K_{sb} is the wave number corresponding to the maximum growth rate Ω as obtained from Eq. (3) and the film thickness, h, is calculated from the breakup length, L, the radial distance from the centerline to the mid-line of the liquid sheet at the atomizer exit, r_o, and the spray angle, θ, as follows: $h \ = \ \frac{\dot{m}}{2\pi\rho_l V(r_o + Lsin(\theta/2))}$. For short waves, the ligament diameter is independent of the liquid sheet thickness and is assumed to be proportional to the wave length associated with the maximum growth rate Ω as follows: $d_L \ = \ \frac{2\pi C_L}{K_{sb}}$ where the ligament constant, C_L, is equal to 0.5.

For both long and short waves, Dombrowski & Johns [51] developed a linear stability analysis for the further breakup from ligaments to droplets based on the Weber's analysis on capillary instability. The analysis shows that the breakup occurs when the amplitude of the unstable waves nears the radius of the ligament. And the corresponding most unstable wavenumber, K_{Lb}, is given by:

$$K_{Lb}d_L \;=\; [\frac{1}{2} + \frac{3\mu_l}{2\sqrt{\rho_l \sigma d_L)}}]^{-\frac{1}{2}} \qquad (8)$$

This analysis thus yields the most probable droplet size based on a simple mass balance calculation where $d_b^3 \;=\; 3\pi d_L^2/K_{Lb}$.

1.1 Application to pressure swirl atomization

For the pressure swirl atomizer, the initial injector exit velocity and liquid sheet thickness are calculated following the approach of Schmidt et al [48]. It assumes a uniform velocity profile for the initial liquid velocity, V, as given by,

$$V \;=\; max\{0.7, \frac{4\dot{m}}{\pi d_o^2 \rho_l cos\theta}\sqrt{\frac{\rho_l}{2\Delta p}}\}\sqrt{\frac{\Delta p}{\rho_l}} \qquad (9)$$

where $\dot{m}$ and θ are the measured mass flow rate and spray angle, respectively, d_o is the injector exit diameter, and Δp is the pressure drop in the injector. Once V is known, the corresponding axial component of the sheet velocity is calculated via $u \;=\; V\ cos\theta$. And the initial sheet thickness h_o is calculated from the conservation of mass:

$$\dot{m} \;=\; \pi\rho_l u h_o(d_o - h_o) \qquad (10)$$

At the point of primary breakup, the actual drop size is chosen from a Rosin-Rammler distribution with the mean size as given by d_b of the sheet breakup model. Further movement of the droplets is tracked by making use of a Lagrangian formulation.

2. Blob Jet Primary Atomization Model (Likely Application: Single-Orifice Nozzles)

Here we summarize the details of the blob jet primary atomization model taken from Reitz & Bracco [41] and Reitz [52,45]. It applies for a cylindrical liquid jet issuing into an incompressible gas. The following dispersion relation was obtained based on the stability analysis of a cylindrical liquid surface subjected to linear perturbations:

$$\begin{aligned}\omega^2 = 2\nu_l k^2\omega\{\frac{I_1'(ka)}{I_0(ka)} - \frac{2kl}{k^2+l^2}\frac{I_1(ka)}{I_0(ka)}\frac{I_1'(la)}{I_0(la)}\}\\ = \frac{\sigma k}{\rho_l a^2}(1-k^2a^2)\{\frac{l^2-k^2}{l^2+k^2}\}\frac{I_1(ka)}{I_0(ka)}\\ +\frac{\rho_g}{\rho_l}(U - i\omega lk)^2k^2\{\frac{l^2-k^2}{l^2+k^2}\}\frac{I_1(ka)}{I_0(ka)}\frac{K_0(ka)}{K_1(ka)} \qquad (11)\end{aligned}$$

where I_0, I_1, and K_0, K_1 are the modified Bessel functions of the first and the second kinds.

Reitz [52,45] generated curve-fits of numerical solutions to Eq. (11) for the maximum growth rate $(\omega = \Omega)$ and the corresponding wavelength $(\lambda = \Lambda)$:

$$\frac{\Lambda}{a} \;=\; 9.02\frac{(1+0.45Z^{0.5})(1+0.4T^{0.7})}{(1+0.87We_g^{1.67})^{0.6}} \qquad (12)$$

$$\Omega\{\frac{\rho_l a^3}{\sigma}\}^{0.5} \;=\; \frac{0.34+0.38We_g^{1.5}}{(1+Z)(1+1.4T^{0.6})} \qquad (13)$$

where $Z = \frac{We_l^{0.5}}{Re_l}$, $T = ZWe_g^{0.5}$, $We_l = \frac{\rho_l U^2 a}{\sigma}$, $We_g = \frac{\rho_g U^2 a}{\sigma}$, and $Re_l = \frac{Ua}{\nu_l}$. A core region is predicted with the blob injection method because there is a region of large discrete liquid parcels near the nozzle. Based on the jet stability theory, new drops are formed from a parent drop or blob. It is assumed that small droplets (with radius, r) are formed from the parent drops (with radius, a) with drop size proportional to the wavelength of the fastest-growing or most-unstable wave,

$$\begin{aligned} r \;&=\; B_o\Lambda \qquad (if\ B_o\Lambda \leq a)\ \ or\\ r \;&=\; min\{(3\pi a^2U/2\Omega)^{0.33},\ (3a^2\Lambda/4)^{0.33}\}\\ &\qquad (if\ B_o\Lambda > a,\ one\ time\ only) \qquad (14)\end{aligned}$$

where $B_o = 0.61$ according to Reitz [52,45]. In the above, it is assumed for the $(B_o\Lambda \leq a)$ condition that small droplets are formed with the dropsize proportional to the wavelength of the fastest growing mode and the second $(B_o\Lambda > a)$ condition applies to drops larger than the jet and it assumes that the jet disturbance has frequency $\Omega/2\pi$ (a drop is formed each wave period) or that the drop size is determined from the volume of liquid contained under one surface wave. And the rate of change of droplet radius due to breakup is given by

$$da/dt \;=\; -(a-r)/\tau \qquad (r \leq a) \qquad (15)$$

where τ is the breakup time, $\tau \;=\; 3.726B_1a/\Lambda\Omega$, and the value for the breakup time constant, B_1, depends on the injector characteristics and its value ranges between 1.732 to 40. And $a(t = t_o) \;=\; a_o$ is the initial drop radius at time, t_o.

After the breakup, a new parcel containing product drops of size, r, is created and added to the computations. This was done as long as the mass of the liquid removed from the parent $(\rho_l 4\pi(a_o^3 - a^3)/3)$ reached or exceeded 3% of the average injected parcel mass and if the number of product drops is greater than or equal to the number of parent drops [45]. While waiting for sufficient product drops to accumulate, the parent drop number was adjusted so that $Na^3 = N_o a_o^3$ but the parent drop number, N_o, was then restored following the creation of the new product parcel [45].

In the case of $(B_o\Lambda > a)$, the parent parcel was replaced by a new parcel containing drops with size given by Eq. 14 after a time equal to τ (with $N = N_o a_o^3/r^3$) [45]. This breakup procedure was allowed only once for each injected parcel [45].

Validation of the model for a single hole orifice in a typical diesel engine was demonstrated by Reitz and Diwaker [46] and Reitz [52,45].

3. Air Blast Primary Atomization Model (Likely Application: Air Blast Atomizers)

The air blast atomization model is essentially based on the idea of pressure-swirl atomization model (Section 1.1) as the primary atomization of an air blast injector is based on the aerodynamic analysis involving the Kelvin-Helmholtz instability of a liquid jet in an incompressible gas. It, however, differs from the pressure-swirl atomization model in the determination of the initial sheet velocity and thickness, which are given by

$$V_{sheet} \;=\; \alpha V_l \;+\; (1-\alpha)V_g \tag{16}$$

$$\delta \;=\; r\big[1 - \sqrt{1 - \frac{\dot{m}}{\pi r^2 \rho V_{sheet}}}\big] \tag{17}$$

where α has a value of 0.12 to 1 depending on the fuel filmer characteristics, r is the radius of the fuel filmer, and $\dot{m}$ is the fuel flow rate. The continuous annular sheet is represented by a finite number of point injectors located randomly along the circular ring of the liquid sheet.

4. Modified BLS Primary Atomization Model (Likely Application: Liquid Jet in a Cross Flow)

Here we summarize the details of the BLS atomization model taken from Khosla and Crocker [47]. In this model both surface shear breakup and column breakup modes are included. Before column breakup, fragments may be formed due to boundary layer stripping depending on the local Weber number and q $(= \rho_l u_l^2/\rho_g u_g^2)$. When the jet reaches the column breakup time, the entire jet breaks into fragments. It also allows for further breakup of the fragments based on a modified boundary layer stripping. And it is followed by a final breakup step based on the Rayleigh-Taylor secondary droplet breakup model.

Fragments are stripped from the liquid column if the We_g satisfies the following criteria:

$$\begin{aligned} We_g &> 50Re_g^{1/2}\, q^{-1/0.81}\; and \\ We_g &> 15 \end{aligned} \tag{18}$$

where

$$Re_g = \rho_g d_j u_g/\mu_g$$
$$we_g = u_g^2 d_j \rho_g/\sigma_l$$

where Re_g and We_g are based on the gas velocity component normal to the liquid jet direction instead of the relative velocity. If column stripping does occur, the amount of mass removed from the column is given by

$$M_{shed} = \frac{3}{4}\pi d\rho_l \frac{t_b}{t^\star} u_{rel} A\alpha \sqrt{\frac{\pi d}{4}}\Delta t \tag{19}$$

where,

$$A \;=\; \big[\frac{\rho_g}{\rho_l}\big]^{1/3}\big[\frac{\mu_g}{\mu_l}\big]^{1/3} \tag{20}$$

$$\alpha \;=\; \big[\frac{8\mu_l}{3Au_{rel}\rho_l}\big]^{1/2} \tag{21}$$

$$t^\star \;=\; \frac{d_o\sqrt{\frac{\rho_l}{\rho_g}}}{u_g} \tag{22}$$

where t_b is the liquid column drop lifetime. The addition of the factor $t^\star/t_b$ causes the shedding rate to increase essentially linearly with distance away from the injection location. This accounts for the lack of shedding close to the injection location and the subsequent buildup of shedding over the life of the liquid column. The shed drop SMD (Sauter Mean Diameter) is given by

$$SMD \ = \ 3.1\frac{t_b}{t^\star}d^{1/2}\big[\frac{\rho_l}{\rho_g}\big]^{1/4}\big[\frac{\mu_l}{u_{rel}\rho_l}\big]^{1/2} \tag{23}$$

The $t_b/t^\star$ factor is included with the effect of producing smaller drops near the injection location. However, $t_b/t^\star$ is limited to a minimum value of 2.5 which is never exceeded for some cases. The amount of mass shed is tracked and 10 new parcels are created when the cumulative shed mass after a time step exceeds 1% of the mass of the parcel. Each parcel is allocated an equal amount of the shed mass and the size for each new parcel is selected randomly from a uniform distribution between 0.4 and 1.6 times the mass mean diameter, MMD. And the drop velocities were given by

$$u_d = u_p + O.3(RND - 0.2)(u_g - u_p) \tag{24}$$

$$v_d = v_p + +0.3(RND - 0.2)(v_g - v_p) \tag{25}$$

$$w_d = w_p + 0.25(RND - 0.5)(u_{rel} - w_p) \tag{26}$$

where RND is a random number. Column stripping occurs, assuming the above criteria are met, until the column breakup time is exceeded. Parcels created through the column stripping mechanism are considered to be fragments which may undergo further breakup as discussed below. First, though, the column breakup mechanism, which also produces fragments, is described. The liquid jet column breakup time is given by

$$t_b = A_b We^{0.62}t^\star \tag{27}$$

The constant, A_b, has a value of 25. Since the present model includes a fragment breakup step after the column breakup, the We dependence for the breakup onset was retained.

After the column breakup time is reached, the column is broken into 18 new parcels with $MMD = 0.45d_j$. The new parcels are also designated as fragments. The size of the fragments still tend to be large, so the ultimate drop size from the primary breakup process is mostly determined from the fragment breakup process. The size distribution is the same as described above for the column stripping. The cross flow and normal velocity components are the same as Eqs. (24) and (25). The lateral velocity component is given by

$$w_d = w_p + 0.1(RND - 0.5)(u_{rel} - w_p) \tag{28}$$

Fragments are further broken into small drops according to a modified version of the boundary layer stripping model based on the following criteria,

$$\begin{aligned} We_g &> \sqrt{Re}\, and \\ We_g &> 15 \end{aligned} \tag{29}$$

where

$$we_g = u_{rel}^2 d_d \rho_g/\sigma_l$$

$$Re_g = \rho_g d_d u_{rel}/\mu_g$$

The criteria are generally the same as for column stripping except that the dependence on q is not needed. Also note that We and Re are now determined using the relative velocity instead of the cross flow velocity. Again, the mass shed from a fragment in a time step and the SMD are given by

$$M_{shed} = 1.2\pi d\rho_l u_{rel} A\alpha\sqrt{\frac{\pi d}{4}}\Delta t \tag{30}$$

$$SMD \ = \ 3.6d^{1/2}\big[\frac{\rho_l}{\rho_g}\big]^{1/4}\big[\frac{\mu_l}{u_{rel}\rho_l}\big]^{1/2} \tag{31}$$

where A and a are given by Eqs. (20) and (21), respectively. The new droplet velocities are given by Eqs. (24), (25), and (28). The broken fragments produce 3 new parcels with size distribution the same as described above when the shed mass from the fragment exceeds 20% of the fragment mass. A fragment can continue to breakup until it no longer meets the criteria of Eq. (29) or until its size is lower than the newly created drops. Once the fragment breakup process is complete, drops may breakup further based on a Rayleigh-Taylor secondary breakup method.

Khosla and Crocker [47] applied the model to predict the properties of Jet A-1 kerosene fuel injected into a cross-flowing air stream.

DETAILS OF THE SECONDARY DROPLET BREAKUP MODELS

Recent studies by Reitz et al [55, 52] have examined the breakup of single droplets moving in a transverse, high-velocity air jet. The high-speed photography provided new insights into the details of the breakup mechanism of a single drop. The droplet breakup regimes are classified as bag, stripping (shear) and catastrophic (surface wave) based on an increasing size of Weber number. In the bag breakup mode (at low Weber number), the drop is flattened by the aerodynamic pressure, then turned inside out, forming a thin hollow bag which is tied together with a circular belt-like structure on the windward side. The bag eventually bursts into smaller liquid fragments, whereas the belt decays into larger ligaments and droplets. In the stripping regime thin sheets or ligaments of fluid are continuously shed from the periphery of the distorting parent drop as a consequence of a K-H instability, causing these sheets to disintegrate into tiny droplets. This process always leaves a coherent residual parent drop. The catastrophic drop breakup takes place in two stages leading to a collection of larger and tiny product droplets: Large amplitude long-wavelength waves caused by drop deceleration induce a R-T instability on the flattened drop which leads to a breakup into large product droplets, while at the same time short surface waves induce a K-H instability on the windward side of the parent drop resulting in a collection of much smaller product droplets. In diesel or other high-pressure gas-turbine sprays the droplets span a wide range of velocities and hence Weber numbers, and thus it is expected that all three droplet breakup mechanisms are relevant in the breakup modeling.

In what follows, we provide some details of the secondary droplet breakup models contained in the CFDRC/UW atomization module: (1) Rayleigh-Taylor, (2) TAB, and (3) ETAB. These details are taken from [52-56, 42-44].

1. Rayleigh-Taylor Secondary Droplet Breakup Model

Here we summarize the Rayleigh-Taylor secondary breakup model developed by Patterson and Reitz [57]. It is based on the analysis developed by Taylor [53,57] that accounts for the disturbances caused by droplet deceleration. In the Rayleigh-Taylor breakup mechanism, the breakup wavelength, Λ, is given by

$$\Lambda \; = \; 0.2\pi\sqrt{3\sigma/|\dot{u}_d|(\rho_l-\rho_g)} \qquad (32)$$

where $\dot{u}_d$ is the drop deceleration $(= \frac{3}{4}\frac{c_d\rho_g U^2}{\rho_l d}$, c_d is the aerodynamic drag coefficient, U is the drop relative velocity, & d is the drop diameter). Furthermore, the breakup wavelength Λ is limited by

$$\Lambda \; = \; max(0.8d, \Lambda)[1 + 0.2(RND - 0.5)] \qquad (33)$$

and the breakup time, $t_{b,RT}$, is given by

$$t_{b,RT} \; = \; c_{freq}\sqrt{0.5\sqrt{\sigma}(\rho_l+\rho_g)(\frac{3}{|\dot{u}_d|(\rho_l-\rho_g)})^{1.5}} \qquad (34)$$

However, the value assigned for the constant, c_{freq}, depends on the droplet classification - parent, product, or default. For more details, one can refer to Patterson and Reitz [21]. After the breakup, no new drop parcels are created and there is no change in velocities between the parent and product drops. However, the drop number in a given parcel changes to $n_{product}$ as given by $n_{parent}(d_{parent}/\Lambda)^3$ due to the change in the sizes between the parent and product drops.

2. The TAB Secondary Droplet Breakup Model (Likely Application: Lenticular-Shaped Droplet Deformations)

In an attempt to provide a description of the droplet and jet disintegration in the modeling of diesel sprays, O'Rourke & Amsden introduced the TAB model [42]. Here we summarize the TAB model taken from [43-44].

The TAB breakup model is based on Taylor's analogy between an oscillating, distorting drop and a spring-mass system. A detailed analysis of this model together with a discussion of its numerical implementation can be found in [42 & 58]. In this model, the drop motion is governed by a linear ordinary differential equation for a forced, damped harmonic oscillator. The forcing term is given by the aerodynamic droplet-gas interaction, the damping is due to the liquid viscosity and the restoring force is supplied by the surface tension. The parameters and constants have been determined partly from theoretical considerations and partly from experimental observations.

The TAB model describes the distortion of the drop by the deformation parameter, $y \; = \; 2x/a$, where x denotes the increase in the radius increase of the equator from its equilibrium position and a

is the drop radius. The equation for the distortion parameter y is given by

$$\ddot{y} + \frac{5\mu_l}{\rho_l a^2}\dot{y} + \frac{8\sigma}{\rho_l a^3}y = \frac{2\rho_g |U|^2}{3\rho_l a^2} \tag{35}$$

Assuming a constant relative drop-gas velocity, U (which is satisfied in the numerical solution process during a given time step), the solution to Eq. (35) is given by

$$y(t) = \frac{We}{12} + e^{\frac{-t}{t_d}}\big([y(o) - \frac{We}{12}]cos\omega t$$

$$+ [\frac{\dot{y}}{\omega} + \frac{y(0) - \frac{We}{12}}{\omega t_d}]sin\omega t\big) \tag{36}$$

where $We = \rho_g a U^2/\sigma$ and

$$t_d = \frac{2\rho_l a^2}{5\mu_l} \tag{37}$$

$$\omega^2 = \frac{8\sigma}{\rho_l a^3} - \frac{1}{t_d^2} \tag{38}$$

In this model, it is assumed that a necessary condition for drop breakup is reached when $We > We_{crit}$. And the value for the critical Weber number is determined experimentally to be 6. For an inviscid liquid with initial deformation conditions $y(0) = \dot{y}(0) = 0$, the solution to Eq. (35) reduces to $y(t) = We(1 - cos\omega_o t)/12$, where $\omega_o^2 = \frac{8\sigma}{\rho_l a^3}$ [44]. Consequently, the breakup occurs when $y(t) > 1$. The drop size after breakup is determined by an energy balance equation between the parent and the product droplets which equates the surface energies of parent drops with the energies of product drops due to oscillation and distortion. Also, the product droplets are initially equipped with the additional deformation velocity $\dot{x} = \alpha\dot{y}/2$, which acts normal to the path of the parent drop and is responsible for the formation of the spray angle.

One major advantage of this model is that it is based on a simple linear equation and it can be used effectively to describe the lenticular-shaped droplet deformations as observed in the experiments of [56 & 54].

3. The ETAB Secondary Droplet Breakup Model (Likely Application: High-Pressure Diesel Engine)

Here we provide some details of the ETAB model taken from Tanner [43-44]. The ETAB model is based on the following modifications to the standard TAB model: (1) the droplet disintegration is modeled via an exponential law which relates the mean product droplet size to the breakup time of the parent drop; and (2) an energy balance consideration between the parent and product droplets yields an expression for the normal velocity component of the product droplet.

When the breakup condition of $We > We_{crit} = 6$ and $y(t) = 1$ is met, then the parent drop breaks up into a collection of product droplets, subject to a size distribution function which, in general, depends on the breakup mechanism. In the ETAB model, the rate of product droplet generation, dn(t)/dt, is given by

$$\frac{dn(t)}{dt} = 3K_b n(t) \tag{39}$$

where $n(t) = m_0/\bar{m}(t)$ and m_0 is the mass of the parent drop and $\bar{m}$ the mean mass of the product droplet distribution. Utilizing the fact that $dn/dt = -(m_0/\bar{m}^2)(d\bar{m}/dt)$, leads to the breakup law which relates the product drop size to the breakup time as determined by the TAB model.

$$\frac{d\bar{m}}{dt} = -3K_b\bar{m} \tag{40}$$

The breakup constant K_b depends on the breakup regime and is given by parent drop properties only. Bag breakup occurs if $We = We_t$ and stripping breakup happens if $We > We_t$. And it is given by

$$\begin{aligned} K_b &= k_1\omega \quad if\, We \le We_t \quad or \\ K_b &= k_2\omega\sqrt{We} \quad if\, We > We_t \end{aligned} \tag{41}$$

The values for We_t, k_1 and k_2 have been determined experimentally and has been set to $k_1 \approx k_2 = 1/4.5$ and $We_t = 80$.

In this model, a uniform product droplet size distribution is assumed. It is also noted that the choice of uniform distribution is not expected to be realistic but may produce good approximations when averaged over many drop breakups, because parent drops of different sizes and Weber numbers will in general yield a wide range of duct droplet sizes. With this assumption, Eq. (40) becomes

$$\frac{r}{a} = e^{-K_b t} \tag{42}$$

where a and r are the radii of the parent and product drops, respectively.

After breakup of a parent drop, the initial deformation parameters of the product droplets are set to $y(0) = \dot{y}(0) = 0$. Also, the product droplets are initially supplied with a velocity component perpendicular to the path of the parent drop with a value $v_T = A\dot{x}$, where A is a constant determined from the following energy balance consideration. The energy of the parent drop is the sum of the surface tension energy and the droplet deformation energy. The second is computed as the product of the aerodynamic drag and the drop deformation at the stagnation point, estimated to be $5a/9$. This leads to

$$E_{parent} = 4\pi\sigma a^2 + 5\pi c_d \rho_g a^3 |U|^2/18 \qquad (43)$$

And the energy of the product droplets in the frame of reference of the parent drop is given by

$$E_{product} = 4\pi\sigma a^3/r_{SMR} + A^2\pi\rho_l a^5 \dot{y}^2/6 \qquad (44)$$

where the Sauter mean radius, r_{SMR}, enters via the relation $\bar{r}^2 = a^3/r_{SMR}$. From Eqs. (43) and (44) one obtains the relation

$$A^2 = 3[1 - a/r_{SMR} + 5c_d We/72]\omega^2/\dot{y}^2 \qquad (45)$$

where $\omega^2 = \frac{8\sigma}{\rho_l a^3}$.

Tanner [43-44] analysis yields an approximate value of 0.69 for A showing that only 70% of the parent drop deformation velocity goes into the normal velocity component of the product droplets, where as it is 100% in the standard TAB model.

Also, the characteristic time, $\tau\ (= \frac{1}{K_b})$, for breakup in Eq. (42) for an inviscid liquid ($\mu_l = 0$) is given by $\alpha_1\sqrt{\frac{\rho_l a^3}{\sigma}}\ if\ We \leq We_t$ or $\alpha_2\sqrt{\frac{\rho_l}{\rho_g}}\frac{a}{|U|}\ if\ We > We_t$, where the suggested values are for $\alpha_1 = (\sqrt{8}k_1)^{-1}$ and $\alpha_2 = (\sqrt{8}k_2)^{-1}$.

The application of the ETAB model in the simulation of a high pressure liquid jet breakup can be found in [43-44].

REFERENCES

1. W.A. Sirignano, "Fluid Dynamics of Sprays," Journal of Fluids Engineering, vol. 115, no. 3, pp. 345-378, September 1993.

2. Crowe, C.T., "Numerical Models for Dilute Gas-Particle Flows," Journal of Fluids, Vol. 104, pp. 297-301, 1982.

3. C.T. Crowe, M.P. Sharma, and D.E. Stock, The Particle-Source-in Cell (PSI-CELL) Model for Gas-Droplet Flows, J. Fluids Eng., vol. 99, pp. 325, 1977.

4. Y. El Banhawy and J.H. Whitelaw, Calculation of the Flow Properties of a Confined Kerosene-Spray Flame, AIAA J., vol. 18, no. 12, pp. 1503-1510, 1980.

5. Faeth, G.M., "Mixing, transport, and Combustion in Sprays," Progress Energy Combustion Science, Vol. 13, pp. 293-345, 1987.

6. Raju, M.S., and Sirignano, W.A., "Spray Computations in a Centerbody Combustor," Proceedings of the 1987 ASME-JSME Thermal Engineering Joint Conference, Vol. 1, pp. 61-71, Honolulu, HI, March 1987. Also see Journal of Engineering for Gas Turbines and Power, Vol. 1, No. 4, pp. 710-718, October 1989.

7. Raju M.S., and Sirignano, W.A., "Multi-Component Spray Computations in a Modified Centerbody Combustor," Journal of Propulsion and Power, Vol. 6, No. 2, March-April 1990.

8. Raju, M.S., "AGNI-3D: A Computer Code for the Three-Dimensional Modeling of a Wankel Engine," Computers in Engine Technology, Proceedings IMechE, London, United Kingdom, pp. 27-37, 1991.

9. Raju, M.S., "Heat Transfer and Performance Characteristics of a Dual-Ignition Wankel Engine," Journal of Engines, the 1992 SAE Transactions, Section 3, pp. 466-509.

10. M.S. Raju, Application of Scalar Monte Carlo Probability Density Function Method For Turbulent Spray Flames, Numerical Heat Transfer, Part A, vol. 30, pp. 753-777, 1996.

11. M.S. Raju, Current Status of the Use of Parallel Computing in Turbulent Reacting Flows: Computations Involving Sprays, Scalar Monte Carlo Probability Density Function & Unstructured Grids, Advances in Numerical Heat Transfer, vol. 2, ch. 8, pp.259-287, 2000.

12. M.S. Raju, Scalar Monte Carlo PDF Computations of Spray Flames on Unstructured Grids With Parallel Computing, Numerical Heat Transfer, Part B, No. 2, Vol. 35, pp. 185-209, March 1999.

13. M.S. Raju, On the Importance of Chemistry/Turbulence Interactions in Spray Computations, Numerical Heat Transfer, Part B: Fundamentals, No. 5, Vol. 41, pp. 409-432, 2002.

14. M.S. Raju, LSPRAY - A Lagrangian Spray Solver - User's Manual, NASA/CR-97-206240, NASA Lewis Research Center, Cleveland, Ohio, November 1997.

15. M.S. Raju, EUPDF - An Eulerian-Based Monte Carlo Probability Density Function (PDF) Solver - User's Manual, NASA/CR-1998-20401, NASA Lewis Research Center, Cleveland, Ohio, April, 1998.

16. Reid, R.C., Prausnitz, J.M., and Poling, B.E., "The Properties of Gases and Liquids," 4th Edition, McGraw-Hill Book Company, 1987.

17. Prausnitz, J.M., and Chueh, P.L., "Computer Calculation For High pressure Vapor-Liquid Equilibria," Prentice-Hall, Englewood Cliffs, N.J., 1968.

18. Aggarwal, S.K., Shu, Z., Mongia, H., and Hura, H., "Multicomponent Fuel Effects on the Vaporization of a Surrogate Single-Component Fuel Droplet," AIAA Paper 98-0157, 36th Aerospace Sciences Meeting, Reno, Nevada, Jan. 12-15, 1998.

19. Aggarwal, S.K., Shu, Z., Mongia, H., and Hura, H., "Multicomponent Single-Component Fuel Droplet Evaporation Under High Pressure Conditions," 34th AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, Cleveland, OH, July 13-15, 1998.

20. Bellan, J., "Supercritical (and Subcritical) fluid Behavior and Modelling: Drops, Streams, Shear

and Mixing Layers, Jets, and sprays," Prog. Energy Combust. Sci., Vol. 26, pp. 329-366, 2000.

21. R. Ryder, CORSAIR User's Manual: Version 1.0, SID: Y965, Pratt and Whitney Engineering, United Technologies Corporation, 25 January 1993.

22. N.S. Liu and R.M. Stubbs, Preview of National Combustion Code, AIAA 97-3114, AIAA/ASME/SAE/ASEE 33rd Joint Propulsion Conference, Seattle, Wash., July 6-9, 1997.

23. K.-H. Chen, A.T. Norris, A. Quealy, and N.-S. Liu, Benchmark Test Cases For the National Combustion Code, AIAA 98-3855, AIAA/ASME/SAE/ASEE 34th Joint Propulsion Conference, Cleveland, Ohio, July 13-15, 1998.

24. McBride, S. Gordan, and M. Reno, "Coefficients for Calculating Thermodynamic and Transport Properties of Individual species," NASA TM-4513, NASA Lewis Research Center, 1993.

25. C.R. Wilke, "A Viscosity Equation for Gas Mixture," Chem. Physics, Vol. 18, no. 4, pp. 517-519, April 1950.

26. F.M. White, Viscous Flows, McGraw-Hill Inc., 1974.

27. K. Kuo, Principles of Combustion, Wiley and Sons, 1976.

28. Peng, D.Y., and Robinson, D.B., Ind. Eng. Chem. Fundam., Vol. 15, pp. 59, 1976.

29. Reichenberg, D., "The Viscosities of Gases at High Pressures," Natl. Eng. Lab., Rept. Chem. 38, East Kilbride, Glascow, Scotland, August 1975.

30. Stiel, L.I., and Thodos, G., AICHE J., Vol. 10, pp. 26, 1964.

31. Takahashi, S., J. Chem. Eng. Japan, Vol.7, pp. 417, 1974.

32. Yuen, M.C., and Chen, L.W., "On Drag of Evaporating Droplets," Combust. Sci. Technol., Vol. 14, pp. 147-154, 1976.

33. A.A. Amsden, P.J. O'Rourke, and T.D. Butler, "KIVA-II: A Computer program For Chemically Reactive Flows With Sprays," LA-11560-MS, Los Alamos National Laboratory, Los Alamos, New Mexico 87545.

34. A.Y. Tong and W.A. Sirignano, Multi-component Transient Droplet Vaporization With Internal Circulation: Integral Formulation and Approximate Solution, Numerical Heat Transfer, vol. 10, pp. 253-278, 1986.

35. R. Clift, J.R. Grace, and M.E. Weber, Bubbles, Drops, and Particles, Academic, New York, 1978.

36. H. Schlichting, Boundary-Layer Theory, McGraw-Hill Series in Mechanical Engineering: McGraw-Hill, Inc., New York, 1968.

37. J.S. Ryan and S.K. Weeratunga, Parallel Computation of 3-D Navier-Stokes Flowfields for Supersonic Vehicles, AIAA 93-0064, AIAA 31st Aerospace Sciences Meeting, Reno, Nevada, 1993.

38. V.G. McDonell and G.S. Samuelson, An Experimental Data Base for the Computational Fluid Dynamics of Reacting and Nonreacting Methanol Sprays, J. Fluids Engineering, vol. 117, pp.145-153, 1995.

39. D.L. Bulzan, Structure of a Swirl-Stabilized, Combusting Spray, NASA Technical Memorandum: NASA TM-106724, Lewis Research Center, Cleveland, Ohio, 1994.

40. D.L. Bulzan, Velocity and Drop Size Measurements in a Confined, Swirl-Stabilized Combusting Spray, AIAA 96-3164, 32rd AIAA/ ASME/ SAE/ ASEE Joint Propulsion Conference, July 01-03, 1996/Buena Vista, FL.

41. Reitz, R.D., & Bracco, F.V., "Mechanism of Atomization of a Liquid Jet," Phy. Fluids, Vol. 25, ch. 10, Oct., 1982.

42. O'Rourke, P.J., and Amsden, A.A., "The TAB Method for Numerical Calculation of Spray Droplet Breakup," SAE Technical Paper 872089, 1987.

43. Tanner, F.X., "Liquid Jet Atomization and Droplet Breakup Modeling of Non-evaporating Diesel Fuel sprays," SAE Technical Paper 970050, 1998. Also SAE 1997 Transactions: Journal of Engines, Vol. 106, Sec. 3, pp. 127-140, 1998.

44. Tanner, F.X., “A Cascade Atomization and Drop Breakup Model for the Simulation of High-Pressure Liquid Jets,” SAE Paper 2003-01-1044, 2003.

45. Reitz, R.D., “Modeling Atomization Processes in High-Pressure vaporizing Sprays,” Atomization and Spray Technology, Vol. 3, pp. 309-337, 1987.

46. Reitz, R.D., and Diwakar, R. “Structure of High-Pressure Fuel Sprays,” SAE paper 870598, 1987.

47. Khosla, S., and Crocker, D.S., “CFD Modeling of the Atomization of Plain Jets in Cross Flow for Gas Turbine Applications”' IGTI Turbo Expo: Combustion & Fuels, GT2004-54269, Vienna, Austria, June 2004.

48. Schmidt, D.P., Nouar, I., Senecal, P.K., Hoffman, J., Rutland, C.J., Martin. J., & Reitz, R.D., “Pressure-Swirl Atomization in the Near Field,” 1999 SAE Congress, SAE 1999-01-0496.

49. Senecal, P.K., Schmidt, D.P., Nouar, I., Rutland, C.J., & Reitz, R.D., “Modeling High Speed Viscous Liquid Sheet Approximation,” 1998.

50. Dombrowski, N., & Hooper, P.C., “The Effect of Ambient Density on Droplet Formation in Sprays,” Chem. Eng. Sci. vol. 17, pp. 291, 1962.

51. Dombrowski, N., & Johns, W.R., “The Aerodynamic Instability and Disintegration of Liquid Sheets,” Chem. Eng. Sci. vol. 1718, pp. 203, 1963.

52. Reitz, R.D., “Computer Modeling of Sprays,” Spray Technology Short Course, Pittsburg, PA May 7, 1997.

53. Taylor, G.I., “The instability of Surfaces When Accelerated in a Direction Perpendicular to Their Planes,” Proc. Royal Soc., A, Vol. 201, pp.192-196, 1950.

54. Hwang, S.S., Liu, Z., and Reitz, R.D., “Breakup Mechanisms and Drag Coefficients of High-Speed Vaporizing Liquid Drops,” Atomization and Sprays, Vol. 6, pp. 353-376, 1996.

55. Liu, A.B., Mather, D., and Reitz, R.D., “Modeling the Effects of Droplet Drag and Breakup on Fuel sprays,” SAE Technical paper 930072, 1993.

56. Liu, A.B., and Reitz, R.D., “Mechanisms of Air-Assisted Liquid Atomization,” Atomization and Sprays, Vol. 3, pp. 55-75, 1993.

57. Patterson, M.A., and Reitz, R.D., “Modeling the Effects of Fuel Characteristics on Diesel Engine Combustion and Emission,” SAE 980131,1998.

58. Amsden, A.A., O'Rouke, P.J., and Butler, T.D., “KIVA II: A Computer Program for Chemically Reactive Flows With Sprays,” Technical report LA-11560-MS, Los Alamos laboratory, May 1989.

59. Raju, M.S., “Our Computational Experience With the Integration of a Spray Atomization Module,” ICCES0520050720481, International Conference on Computational & Experimental Engineering Sciences, Chennai, India, Dec. 1-10, 2005.

60. Raju, M.S., “Numerical Investigation of Various Atomization Models in the Modeling of a Spray Flame,” AIAA Paper 2006-0176, 44th Aerospace Sciences Meeting and Exhibit, Reno, Nevada, Jan. 9-12, 2006.

61. Lucas, K.D., Tseng, C.C., Pourpoint, T.L., Lucht, R.P., and Anderson, W.E., “Imaging Flashing Injection of Acetone at Jet Engine Augmentor Conditions,” AIAA Paper 2007-1182, 45th AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, Jan. 8-11, 2007.

62. Zuo, B., Gomes, A.M., and Rutland, C.J., “Modeling Superheated Fuel Sprays and Vaporization,” Inj. J. Engine Research, vol. 1, no. 4, pp. 321-336.

63. Schmehl, R, and Steelant, J., “Flash-Evaporation of Oxidizer During Start-Up of an Upper-Stage Rocket Engine,” AIAA Paper 2003-5075, 39th AIAA/ASME SAE/ASEE Joint Propulsion Conference and Exhibit, Huntsville, Alabama, July 20-23 2003.

64. Schmehl, R, and Steelant, J., “Evaluation of Oxidizer Temperature Drop in a Combustion Chamber,” 4th International Conference on Launcher Technology ”Space Launcher Liquid Propulsion, Liege, Belgium, December 3-6 2002.

65. Adachi, M., McDonnel, V.G., Tanaka, D., Senda, J, and Fujimoto, H., “Characteristics of

Fuel Vapor Concentration Inside a Flash Boiling Spray," SAE Paper 970871, 1997.

66. Lefebvre, A., "Atomization and Sprays," Hemisphere Publishing Company, New York, pp. 165-222, 1989.

67. VanDerWege, B.A., Lounberry, T.H., and Hochgreb, S., "Numerical Modeling of Fuel Sprays in DISI engines Under Early-Injection Operating Conditions," SAE Paper 2000-01-0273, 2000.

68. Reitz, R.D., "A Photographic Study of Flash-Boiling Atomization," Aerosol Science Technology, Vol, 12, pp. 561-569, 1990.

69. R. Borghi, Turbulent Combustion Modeling, Prog. Energy Combust. Sci., vol. 14, pp. 245-292, 1988.

70. S.B. Pope, PDF Methods for Turbulent Reactive Flows, Prg. Energy Combust. Sci., vol. 11, pp. 119-192, 1985.

71. S.M. Correa, Development and Assessment of Turbulence-Chemistry Models in Highly Strained Non-Premixed Flames, AFOSR/NA Contractor Report, 110 Duncan Avenue, Bolling AFB, DC 20332-0001, 31 October 1994.

72. Yildiz, D., Rambaud, P., Van Beeck, J., Buchlin, J.-M., "Characterization of Superheated Liquid Jet Atomization Phase Doppler Anemometer (PDA) and High-Speed Imaging," Proceedings of FEDSM2006: 2006 ASME Joint U.S.-European Fluids Engineering Summer Meeting, July 17-20, 2006, Miami, Florida.

73. Yildiz, D., Rambaud, P., Van Beeck, J., Buchlin, J.-M., "Evolution of the Spray Characteristics in Superheated Liquid Jet Atomization in Function of Initial Flow Conditions," ICLASS-2006, Paper ID ICLASS06-122, Aug. 27-Sept 1, 2006, Kyoto, Japan.

74. Lee J., Fotache C., Gopalakrishnan, S., & Schmidt, D., "Validated tools for Low Emissions Injector Design Using Superheated/Supercritical Fuels," NASA/CR–Year-2010/NNC07CB05C, Glenn Research Center, Cleveland, Ohio.

75. Sher, E., & Elata, C., "Spray Formation from Pressure Cans by Flashing," Ind. Eng. Chem. Process Des. Dev., Vo. 16, number 2, pp. 237-242, 1977.

76. Downar-Zapolski, P., Bilicki, Z., Bolle, L., & Franco, J., "The Non-Equilibrum Model One-Dimensional Flashing Liquid Flow," Int. J. of Multiphase Flow, Vol. 22, No. 3, pp. 473-483, 1996.

77. Kawano, D., Ishii, H., Suzuki, H., Gotto, Y., Odaka, M., & Senda, J., "Numerical Study on Flash-Boiling Spray of Multicomponent Fuel," Heat-Transfer-Asian Research (Wiley Periodicals, Inc.), Vol. 35, No. 5, pp. 369-385, 2006.

78. Raju, M.S., & Bulzan, D., "Assessment of Some Atomization Models used in Spray Calculations," ICCES1120110131487, Nanjing, China, 18-21 April, 2011, also as NASA/TM-2011-217029, Glenn Research Center, Cleveland, Ohio, March 2011.

79. Raju, M.S., & Bulzan, D., Current Status of Superheat Spray Modeling With NCC," AIAA 2012-3707, 48th Joint Propulsion Conference and Exhibit cosponsored by AIAA, ASME, SAE, and ASEE, Atlanta, Georgia, July 30August 1, 2012, also as NASA/TM2012-217295, Glenn Research Center, Cleveland, Ohio, March 2012.

80. DesJardin, P.E., Presser, C., & Disimile, P.J., " A Phenomenological Droplet Impact Model for Lagrangian Spray Transport", AIAA 2003-1322, AIAA 41st Aerospace Sciences Meeting and Exhibit, Reno NV, January 6-9, 2003.

81. Stanton, D.W., & Rutland, C.J., "Multi-Dimensional Modeling of Thin Liquid Films and Spray-Wall Interactions Resulting From Impinging Sprays", Intl. J. Heat and Mass Transfer, Vol. 41, pp. 3037-3054, 1998.

82. Umemura, A., "Supercritical Liquid Fuel Combustion," Proc. of the Combust. Inst., Vol. 21, pp. 463-471, 1986

83. Delplanque, J.-P., Sirignano, W.A., "Numerical study of the transient vaporization of an oxygen droplet at sub- and super-critical conditions," Int. J. of Heat and Mass Transfer, Vol. 36, No. 2, pp. 303314, 1993.

84. Gilver, S.D., and Agraham, J., " Supercirtical Droplet Vaporization and Combustion Studies," Prog. Energy Combust. Sci., Vol. 22, pp. 183-197, 1996.

85. Zhu, G., Reitz, R.D., and Aggarwal, S.K., ”Gas-Phase Unsteadiness and Its Influence on Droplet vaporization in Sub- and Super-Critical Environments,” Int. J. of Heat & Mass Transfer, Vol. 44, no. 16, pp. 3081-3093, 2001.

86. Zhu, G., and Aggarwal, S.K., ”Fuel Droplet Evaporation in a Supercritical Environment,” Transactions of the ASME-J. of Engineering for Gas Turbines and Power, Vol. 124, October, pp. 762-770, 2002.