Nanofluid Drop Evaporation: Experiment, Theory, and Modeling

by

William James Gerken

A Thesis Submitted to the Graduate
Faculty of Rensselaer Polytechnic Institute
in Partial Fulfillment of the
Requirements for the degree of
DOCTOR OF PHILOSOPHY
Major Subject: Aeronautical Engineering

Approved by the
Examining Committee:

_____
Matthew A. Oehlschlaeger, Thesis Adviser

_____
Theodorian Borca-Tasciuc, Member

_____
Nikhil A. Koratkar, Member

_____
Joel L. Plawsky, Member

Rensselaer Polytechnic Institute
Troy, New York

July, 2014
(For Graduation August 2014)

CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

a- radius of the average nanoparticle

A_{ij}- Lindsay Bromley correlation constant

B- Spalding mass or temperature transfer number (B_M or B_T)

C- sessile drop concentration of nanoparticles

$c_{p,g}$ - gas phase specific heat

$c_{p_{mix}}$- mixture specific heat

D - drop diameter or diffusion rate

d_f - fractal dimension

d_{np} - nanoparticle diameter

e_l - specific energy

F_D - drag force on a sphere from stokes

g - gravitational constant

h - heat transfer coefficient or drop height

h_{fg} - latent heat of formation of a liquid

J_s - sessile drop evaporation flux

k - evaporation rate

k_B - Boltzman constant

Le - Lewis number

$\dot{m}$ - mass flow rate

m_{sphere} - mass of a spherical drop

MW_i - molecular weight of species i

N_c - number of backbone particles

n_j - number of agglomerates in bin j

Nu - Nusselt number

p - pressure

Pr - Prandtl number

$P_{-\frac{1}{2}+\iota\tau}$ - Legendre function of the first kind

q_c - heat transfer due to convective heat transfer

q_{CHF} - heat transfer at the critical heat flux

q_{evap} - heat transfer due to evaporation

$\dot{q}_{g,cond}$ - gas phase heat conduction rate

$\dot{q}_l$ - rate of liquid heating

q_{nc} - natural convective heat transfer

Q_s - heat of reaction of species *s*

q_{tot} - total heat transfer

r - radius

R- initial drop radius or ideal gas constant

Re - Reynolds number

R_g - nanoparticle agglomerate radius

R_s - spherical drop vertical radius

Sh - Sherwood number

S_i - probability balance equation breakup rate

S_{ij} - Sutherland constant for Lindsay Bromley correlation

St - Stanton number

T- temperature (Kelvin)

t_{agg} - timescale of diffusion driven agglomeration

t_d - drop lifetime

t_f - sessile drop lifetime

t_H - drop heating time

t_L - drop lifetime time

T_{sat}- temperature of liquid-vapor saturation

t_{ss} - sessile drop substrate thickness

t_* - steady state drop heating time

u - velocity

V_B - Brownian velocity of the nanoparticle in solution

Δv_{fg} - change in specific volume due to liquid-vapor phase change

V_j^s - diffusion velocity of species *s* in *j*

$\dot{w}_s$ - reaction rate of species *s* by volume (fuel or oxidizer)

x - distance

X_g - mole fraction of gas phase

X_l - mole fraction of liquid phase

Y_s - mass fraction of species *s*

z - vertical sessile drop coordinate

α - thermal diffusivity

$\alpha_{j,i}$ - probability balance equation collision efficiency of bins *j* and *i*

$\beta_{j,i-j}$ - probability balance equation collision frequency of bins *j* and *i* - *j*

Γ_{ij} - probability balance equation distribution of agglomerate fragments

δ - separation distance between suspended nanoparticles

θ_c - sessile drop contact angle, radians

λ_a - combination conductivity of dead end and chains

λ_l - liquid thermal conductivity

λ_{nc} - dead end particle thermal conductivity

λ_p - particle thermal conductivity

μ_f - pure fluid viscosity

μ_s - colloid viscosity with spherical particles

ρ - density

ρ_{nf} - nanofluid density

ρ_{np} - nanoparticle density

σ - surface tension

τ_{ij} - viscous stress tensor using Stokes hypothesis

φ - volume fraction of nanoparticles

φ_a - agglomerate volume fraction in base liquid

φ_c - backbone particle volume fraction

φ_{int} – internal agglomerate volume fraction

φ_{max} - maximum total volume fraction

φ_{nc} - dead end particle volume fraction

φ_p - total nanoparticle volume fraction

ACKNOWLEDGMENT

I would like to thank all those who have helped me get to this point in my life. I would especially like to thank my Parents, and Grandparents for instilling in me the value of an education. My mentors in Hawken, at NASA GRC, and at RPI were essential in my education. I am indebted to you all.

ABSTRACT

Nanofluids, stable colloidal suspensions of nanoparticles in a base fluid, have potential applications in the heat transfer, combustion and propulsion, manufacturing, and medical fields. Experiments were conducted to determine the evaporation rate of room temperature, millimeter-sized pendant drops of ethanol laden with varying amounts (0-3% by weight) of 40-60 nm aluminum nanoparticles (nAl). Time-resolved high-resolution drop images were collected for the determination of early-time evaporation rate ($D^2/D_0^2 > 0.75$), shown to exhibit D-square law behavior, and surface tension. Results show an asymptotic decrease in pendant drop evaporation rate with increasing nAl loading. The evaporation rate decreases by approximately 15% at around 1% to 3% nAl loading relative to the evaporation rate of pure ethanol. Surface tension was observed to be unaffected by nAl loading up to 3% by weight.

A model was developed to describe the evaporation of the nanofluid pendant drops based on D-square law analysis for the gas domain and a description of the reduction in liquid fraction available for evaporation due to nanoparticle agglomerate packing near the evaporating drop surface. Model predictions are in relatively good agreement with experiment, within a few percent of measured nanofluid pendant drop evaporation rate.

The evaporation of pinned nanofluid sessile drops was also considered via modeling. It was found that the same mechanism for nanofluid evaporation rate reduction used to explain pendant drops could be used for sessile drops. That mechanism is a reduction in evaporation rate due to a reduction in available ethanol for evaporation at the drop surface caused by the packing of nanoparticle agglomerates near the drop surface. Comparisons of the present modeling predictions with sessile drop evaporation rate measurements reported for nAl/ethanol nanofluids by Sefiane and Bennacer [11] are in fairly good agreement.

Portions of this abstract previously appeared as:
W. J. Gerken, A. V. Thomas, N. Koratkar and M. A. Oehlschlaeger, "Nanofluid pendant droplet evaporation: Experiments and modeling," *Int. J. Heat Mass Transfer,* vol. 74, pp. 263-268, July 2014.

W. J. Gerken, M. A. Oehlschlaeger, "Nanofluid Pendant Droplet Evaporation", in Proceedings of the ASME 2013 Summer Heat Transfer Conference, Minneapolis, MN, 2013, p. V001T03A018.

1. Introduction

Recent advances in nanomaterial engineering offer potential for novel modifications or enhancements of fundamental physical processes and material properties (e.g., transport, mechanical, electronic) and the improvement of existing and development of new technologies. For example applications for nanomaterials exist in medical processes, power plant operation, chemical production, lubrication enhancement, and heat transfer, which may in the future contribute to improvements in the standard of living and global condition [1],[2].

Nanofluids, resulting from the suspension of nanoparticles in a base fluid, are one classification of nanomaterials which can exhibit unique properties. Nanofluids have been shown to in some cases have significantly different properties than the base fluid with the change in properties highly dependent on the nanoparticle material, volume fraction, size, degree of agglomeration, and also specific chemical and physical interactions between particle and base fluid [2],[3]. For example, nano-ferrofluids, magnetic nanofluids whose phase (liquid versus solid) can be controlled through an applied magnetic field, are an interesting class of nanofluids that require a high volume fraction of nanoparticles [4]. Other, more subtle property modifications, for example the modification of fluid thermal conductivity [5] or surface tension [6], can be observed with smaller nanoparticle volume fractions. For the past two decades, there have been a significant number of research efforts focused on characterizing nanofluid properties and attempting to understand the influence of suspended nanoparticles on fluid properties.

The properties of nanofluids, once understood and predictable, can be used in engineering applications. For example, nanofluids used in boiling heat transfer applications have been shown to extend the regime of maximum heat transfer [7] or tailor the heat transfer rate for a specific application [2]. Another such application is in the field of combustion, where suspension of energetic nanoparticles in a liquid fuel has been shown to increase the probability of ignition [8], or enhance fuel reactivity, and improve the energy density of the liquid fuel [9].

The evaporation of nanofluid drops, which could have implications for spray cooling or spray combustion utilizing nanofluids, has been studied in a number of recent papers [10],[11],[12],[13]. Most of these studies have reported experimental results for

pendant or sessile drop evaporation leading to the conclusion that the addition of nanoparticles to an evaporating drop reduces the evaporation rate [11],[14]. However, Chen et al. [13] reported the opposite, an increase in evaporation rate with iron oxide nanoparticles added to deionized water. Hence, there is some conflict among the literature for the influence of nanoparticles on pendant and sessile drop evaporation rate. While experimental data for nanofluid drop evaporation does exist in the literature, albeit somewhat conflicting, to the author's knowledge, there has been no prior attempt to present a theory for nanofluid evaporation and model the influence of nanoparticles on the evaporation of either a pendant or sessile drop evaporation until the current work.

In the order of appearance this thesis will focus on nanofluid properties in Chapter 2, where the focus is the current state of the art in nanofluid property evaluation and experimentation. Properties of pure fluids are discussed with the observed changes due to nanoparticle addition. In some cases, theories exist for nanofluid property modification relative to base fluids. In these cases, theory is discussed.

Next, Chapter 3 will reduce the full Navier-Stokes equations to a set of equations applicable to pendant drop modeling. Several different solution techniques for these reduced equations are discussed, ranging from finite element method numeric solutions, and simplified time-stepping models such as that used in the current work, to the simple analytic models that can be developed after further assumptions are made, such as the D-square law.

Chapter 4 will describe and present experimental results for nanofluid pendant drop evaporation conducted as part of this research. Experimental methods, data processing, and interpretation procedures are explained in detail. Results are presented and compared to prior literature studies.

Chapter 5 presents modeling of the pendant drop experiment described in Chapter 4 using equations derived in chapter 3 with the implementation of a new theory for nanofluid evaporation. A simplified numerical time stepping method is coupled with a finite difference solution to gauge the effectiveness of the nanofluid evaporation theory presented in this Chapter. Modeling inputs are also described in detail. The theory and modeling results presented is shown to have excellent agreement with experimental data reported in Chapter 4.

In Chapter 6 the theory developed in Chapter 5 is extended to the sessile drop configuration and numerical modeling results are presented and compared to the nanofluid sessile drop evaporation study reported by Sefiane and Bennacer [11]. These prior experiments, the extension of the theory to sessile drops, and the modeling method and results are described in detail. It is concluded that the proposed theory adequately captures the first-order behavior of nanofluid sessile drop evaporation. The evaluation of the nanofluid evaporation theory for a different experimental configuration and for an experimental carried out in another laboratory is an important validation step.

Lastly, Chapter 7 provides a summary of the present work and offers proposals for future research. For example, experimental studies are needed at a broader range of conditions and for a wider range of nanofluids. Additionally the particle-fluid interaction physics responsible for unique properties of nanofluids need further isolated study.

2. Nanofluid Properties

The suspension of nanoparticles in a fluid has an effect on a great number of the base fluid's properties. The property change generated is a function of nanoparticle concentration, nanoparticle size, nanoparticle stability, and base fluid [2]. Different effects are possible with a seemingly similar combination of parameters, changing one factor among those mentioned while fixing the remainder does not guarantee a similar outcome.

Some properties of nanofluids, such as specific heat and density follow comparatively simple, easily understood rules. The density of a nanofluid mixture is the linear combination of the two (or more) components' densities weighted by the mass fraction. Likewise, the specific heat of a nanofluid is also the linear combination of the mass weighted component specific heats. While other properties, for example thermal conductivity, exhibit more complicated behaviors.

The remainder of this section will discuss literature findings and trends on chosen properties, in order to acquaint the reader with the many dependencies of nanofluid properties.

2.1 Thermal Conductivity

Much of the motivation for nanofluid development is due to their potential for increased thermal conductivity. At the time of the field's genesis, experimental results showed promise for widely increased the thermal conductivity, on the order of 40%-150% as is seen in the review of Das et al. [2] and discussed by Keblinksi [15]. Increased thermal conductivity could allow for smaller or more efficient cooling systems for electronics, transportation systems, or potentially any heat transfer limited application.

The initially reported results of dramatically increased thermal conductivity baffled researchers as the results were greatly under predicted by classical effective field theory, which was used with great success for fluids containing particles with larger dimensions (microns) [16]. As a result, many new theories for nanofluid thermal conductivity enhancement were proposed. These theories, relying on Brownian motion, interfacial

liquid layers, and near field radiation were moderately successful in their ability to describe experimental observations [2], [17], [18], [19], [20].

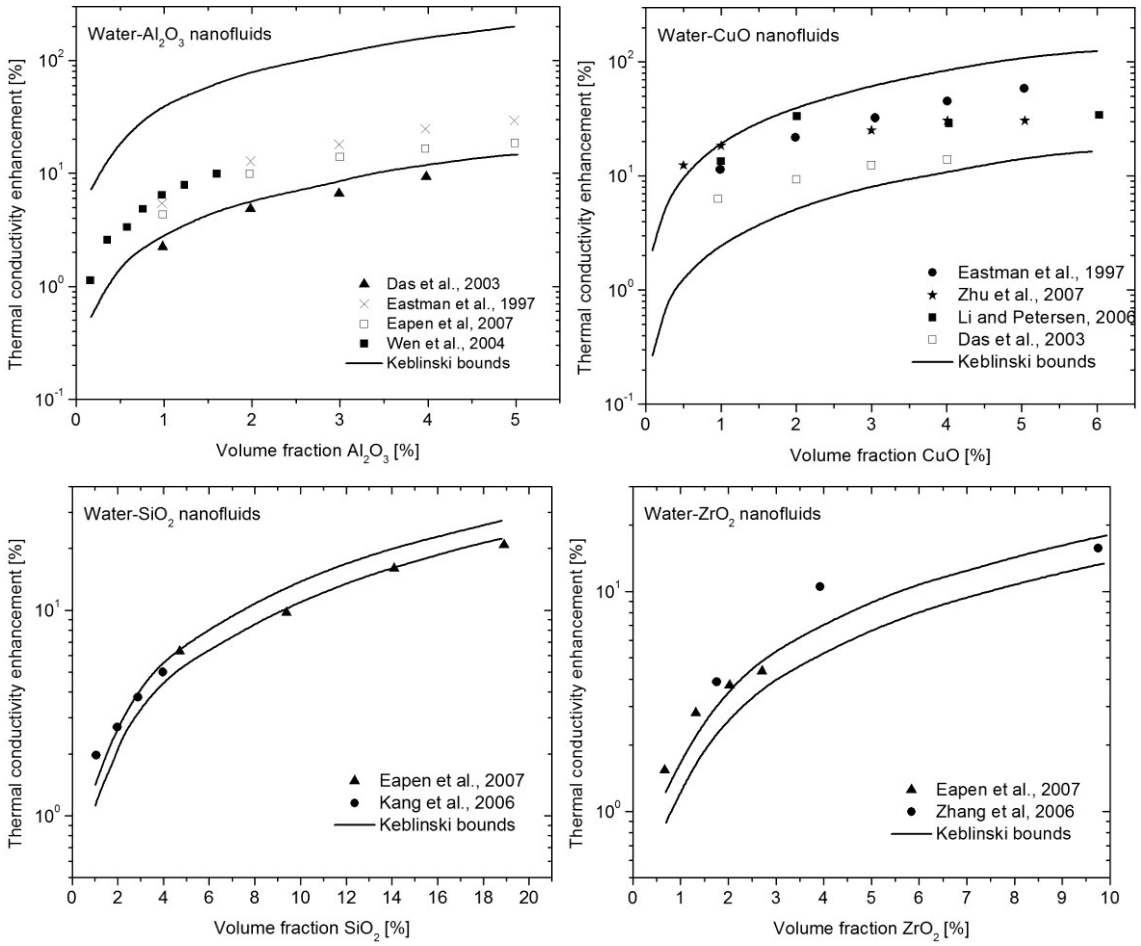

Figure 2.1. Keblinski et al. bounding curves for thermal conductivity enhancement, adapted from reference **[15]**. Graphs illustrate that the thermal conductivity enhancement observed experimentally is within the range expected via the Keblinski et al. implementation of modified effective field theory **[15]**.

In an attempt to resolve the apparent disagreement between effective field theory and measured nanofluid thermal conductivity, it was proposed in 2008 by Keblinski et al. [16] that the differences were due to the assumption of a perfectly dispersed nanofluid. Keblinski et al. proposed, and showed through Hashin-Shtrikman bounding analysis as seen in Figure 2.1., that it was likely that the majority of the nanoparticle samples were not dispersed well enough for the application of the classical effective field theory of Maxwell. Once this assumption was removed, and an effective field theory with agglomeration was introduced, the apparent abnormally high thermal

conductivity results were in the range of theoretic predictions [19]. This model is described in the next subsection.

Additionally, the initial apparent conflict between the prevailing theory (effective field theory) and experimental results is explained through the analysis presented by Keblinski et al. in the same paper [16]. This paper is the one of many to rely heavily on the inferred inconsistency of nanofluid samples to explain differing experimental results and introduces the importance of nanofluid characterization (in this case the particle and agglomerate size distribution) for experimental work.

2.1.1 Prasher et al. Thermal Conductivity Model

The modified effective field theory of Prasher et al. [19] model relies on the fractal nature of nanoparticle agglomerates existing in nanofluids as a way to estimate the conductivities of nanofluids. It was assumed that agglomerates have particles which formed two classifications of structures. These structures were described as either 'backbones' or 'dead ends', each of which may or may not span the entire length of the agglomerate. Numeric values for the respective volume fraction of these structures were found through fractal analysis. In the fractal analysis used by Prasher et al. [19] the total volume fraction of the nanoparticles in the liquid, φ_p, was given as:

$$\varphi_p = \varphi_a \varphi_{int}, \tag{1}$$

where φ_{int} and φ_a are the volume fractions of the nanoparticles within the agglomerate and the agglomerate in the fluid, respectively. These values were found through approximation of the fractal dimension, d_f, and the average radius of a sphere containing an agglomerate, R_g the radius of gyration. Written explicitly φ_{int} is:

$$\varphi_{int} = \left(\frac{R_g}{a}\right)^{d_f - 3}, \tag{2}$$

where a is the radius of the average nanoparticle. Assuming closely packed spherical particles *or* a mixture of entirely agglomerate volumes the volume fraction of the agglomerate within the fluid is:

$$\varphi_a = \frac{\pi}{6} \; or \; 1. \tag{3}$$

From these bulk values, Prasher et al. [19] wrote the number of backbone particles, dead end particles and their volume fraction as:

$$N_c = \left(\frac{R_g}{a}\right)^{d_l}, \tag{4}$$

and,

$$\varphi_c = \left(\frac{R_g}{a}\right)^{d_l-3}, \tag{5}$$

and,

$$\varphi_{nc} = \varphi_{int} - \varphi_c\ , \tag{6}$$

where d_l specifies the ratio of backbone to dead end particles, where all backbone particles exist when d_l is equal to d_f.

Once these input parameters are known, three different thermal conductivities can be written for the dead end particles and the backbone particles and an effective thermal conductivity for the nanofluid. The dead end conductivity is calculated using the Bruggman model [21]. Assuming a homogeneous distribution of perfectly spherical particles for the dead end components only, the following can be written:

$$\frac{(1-\varphi_{nc})(\lambda_l-\lambda_{nc})}{\lambda_l-2\lambda_{nc}} + \frac{\varphi_{nc}(\lambda_p-\lambda_{nc})}{\lambda_l+2\lambda_{nc}} = 0\ , \tag{7}$$

where the λ_l, λ_{nc}, and λ_p are the thermal conductivity of the liquid, the dead end component, and nanoparticle as proposed by Prasher et al. [19].

This dead end conductivity is then used in the cylindrical agglomerate model of Nan et al. [22] to calculate the combination conductivity of the dead end and the chains. Per Prasher et al. [19] this yields:

$$\lambda_a = \lambda_{nc}\left(\frac{3+\varphi_c[2\beta_{11}(1-L_{11})+\beta_{33}(1-L_{33})]}{3-\varphi_c[2\beta_{11}L_{11}+\beta_{33}L_{33}]}\right), \tag{8}$$

and,

$$L_{11} = \frac{0.5p^2}{P^2-1} - 0.5pcosh^{-1}\left(\frac{p}{p^2-1}\right),\ \ p = \frac{R_g}{a}, \tag{9}$$

$$L_{33}\ = 1 - 2L_{11}, \tag{10}$$

$$\beta_{11} = \frac{\lambda_{nc}-\lambda_l}{\lambda_l + L_{11}(\lambda_{nc}-\lambda_l)}. \tag{11}$$

Finally, the total effective nanofluid conductivity is calculated as the solution to the Maxwell-Garnet equation as a function of the chain portion, λ_a, and its volume fraction, as well as the base fluid conductivity:

$$\frac{\lambda}{\lambda_l} = \frac{\lambda_a + 2\lambda_l + 2\varphi_a(\lambda_a - \lambda_l)}{\lambda_a + 2\lambda_l - \varphi_a(\lambda_a - \lambda_l)}. \quad (12)$$

2.2 Nanofluid Viscosity

The viscosity of a nanofluid, which increases with added nanoparticles [23], is not so easily determined, however important. An increase in viscosity results in an increase in required pumping power for heat transfer applications, potentially reducing any efficiency gained through increased thermal conductivity or enhanced phase-change properties for nanofluids. Hence, it is essential to accurately predict the viscosity of the fluid in order to optimize nanofluid nanoparticle volume fraction among other things for heat transfer applications.

Initial theoretical models for the viscosity of fluids containing particles produced by Einstein were intended for inclusion of any sized spherical particles at low (less than 2%) volume fractions as described by Masoumi et al. [24] and Namburu et al. [25], and as presented below:

$$\mu_s = \mu_f \left(1 + \frac{5}{2}\varphi\right), \quad (13)$$

where φ is the volume fraction, and μ_s and μ_f the solution and base fluid viscosities, respectively.

A more recent comprehensive model was proposed by Masoumi et al. in 2009 [24]. In their paper, they pointed out that previous models, like that of Einstein, do not reproduce the experimentally observed dependency of nanofluid viscosity on temperature, particle diameter, or type of nanoparticle. Masoumi et al. proposed a model for viscosity dependent on mean nanoparticle diameter, temperature, nanoparticle volume fraction, and density, as well as the base fluid properties. This model assumes an ideally distributed homogeneous nanofluid, with no agglomeration or chemical interaction between particles, and purely Stokes flow around the particles. The effective fluid viscosity from this model, μ_s, is given by [24]:

$$\mu_s = \mu_f + \frac{\rho_{np} V_B d_{np}^2}{72C\delta}, \quad (14)$$

where μ_f the base fluid viscocity, ρ_{np} and d_{np} the nanoparticle density and diameter, V_B the Brownian velocity of the particle, δ the distance between neighboring particle

centers, and *C* a correction factor to account for boundary layer assumptions made in the derivation of the shear stress on the particle. Values for the Brownian velocity and spacing δ are given as [24]:

$$V_B = \frac{1}{d_{np}} \sqrt{\frac{18\lambda_b T}{\pi \rho_{np} d_{np}}}, \tag{15}$$

and,

$$\delta = \sqrt[3]{\frac{\mu_f}{6\varphi}}\, d_{np}, \tag{16}$$

where λ_b is the base fluid conductivity, T the temperature, ρ_{np}, φ, and d_{np} the nanoparticle density, volume fraction, and diameter, respectively. As shown in Figure 2.2., the Masoumi et al. model provides excellent agreement with measured nanofluid viscosities regardless of base fluid or nanoparticle type or size.

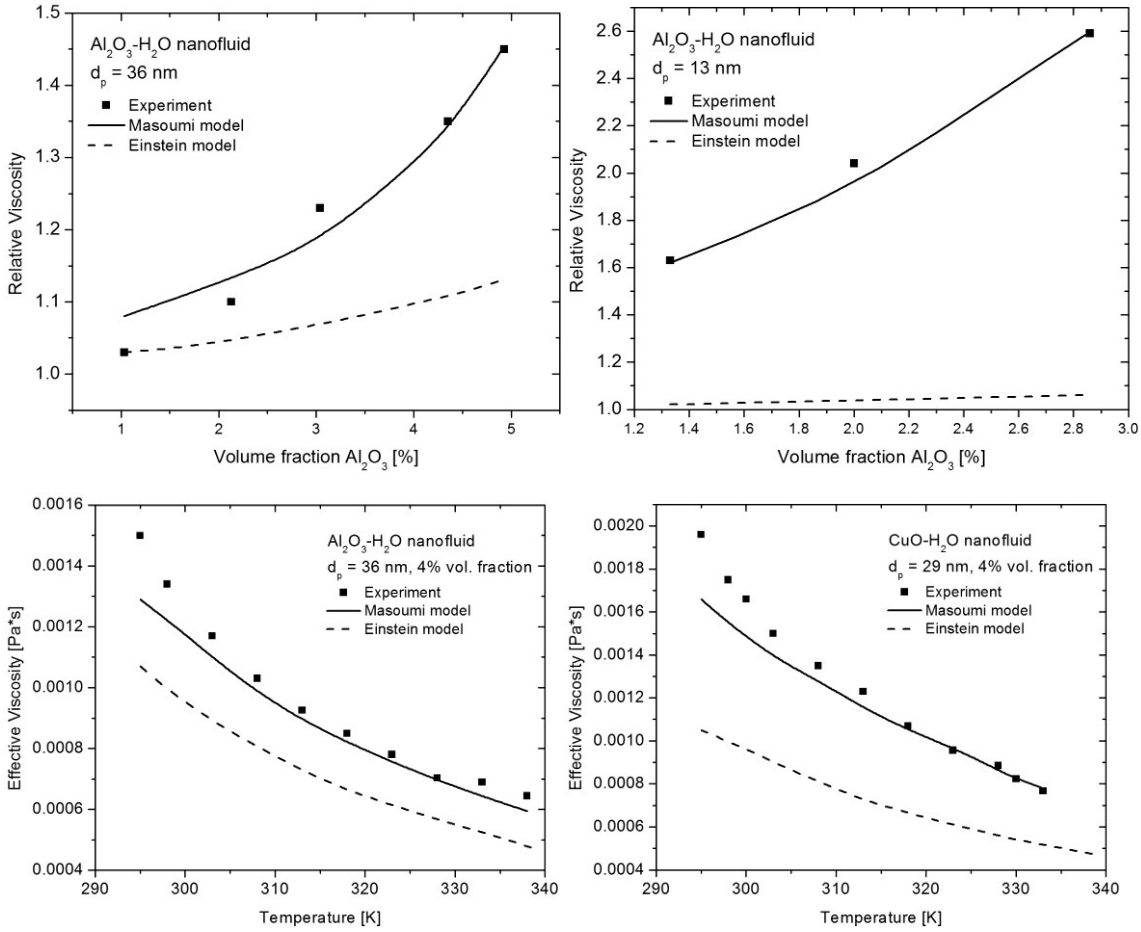

Figure 2.2. Comparison of Masoumi et al. [24] modeling results with experiments, adapted from reference [24].

2.3 Nanofluid Surface Tension

Surface tension is the artificially discrete force applied most notably at the interface between a liquid and a gas. Although usually described with diagrams such as Figure 2.3., surface tension does not occur as a sharp transition from the bulk liquid to the gas phase. The surface is actually a continuous region of decreasing concentration of fluid molecules on the order of tens of molecules thick. As described in Figure 2.3., a simple approach towards understanding the effects of surface tension relies on the attractive van der Waals force of like molecules in the liquid phase, which is greater than the liquid to gas attraction as discussed by Tanvir et al. [26]. At the surface of the liquid, there is a greatly reduced attractive force pulling liquid molecules outward towards the

vapor, while the force pulling the molecule towards the liquid bulk is maintained. As a result, a net force or pressure acting along or across the surface of the liquid exists and can be modeled as a function of curvature via the Young-Laplace equation for surface tension:

$$\Delta p = \sigma\left(\frac{1}{r_1} + \frac{1}{r_2}\right), \tag{17}$$

where Δp is the change in pressure from interior to exterior, σ is the liquid surface tension of units force per length, and r_1 and r_2 are the radius of curvature in orthogonal directions, respectively.

The surface tension of several nanofluids was reported by Tanvir et al. [26] for a variety of base fluids, nanoparticles, and surfactants. Water, ethanol and n-decane were investigated with aluminum oxide (Al_2O_3), aluminum, and boron nanoparticles and multi-walled carbon nanotubes (MWCNTs). Results for all nanofluids (see examples in Figure 2.4.) show at low weight percent concentrations little measurable influence of nanoparticles on surface tension; however, at higher nanoparticle concentrations the surface tension is observed to increase. Tanvir et al. hypothesize that an increase in free surface energy due to the van der Waals particle-particle interaction causes the increased surface tension. They also state that this occurs only at high loadings due to the increased inter-particle distance of particles at the lower concentrations.

Several other published results show inconsistent findings on the effect of nanoparticles on the surface tension of a fluid. A number of studies, Moosavi et al. [27], Kim et al. [28], Kumar and Milanova [6], and Murshed et al. [29] show an increase in the surface tension of a nanofluid with increasing concentration of nanoparticles. However, even in this subset, there is disagreement in the concentration required to see this effect. Kim et al. [28] observed an increase of surface tension at very low concentrations around 0.01 percent by volume. The remaining studies conclude that concentrations on the order of 1-3% and higher by volume are required to see the increase in surface tension. In another study, Murshed et al. [30] reported in 2008 an opposite trend, a decreasing surface tension with the addition of nanoparticles of TiO_2 in deionized water. Yet other studies showed little change in the surface tension with

nanoparticle loading, as reported by Sefiane and Bennacer [11]. These discrepancies could be due to the inconsistent characterization of nanofluids prior to experimentation.

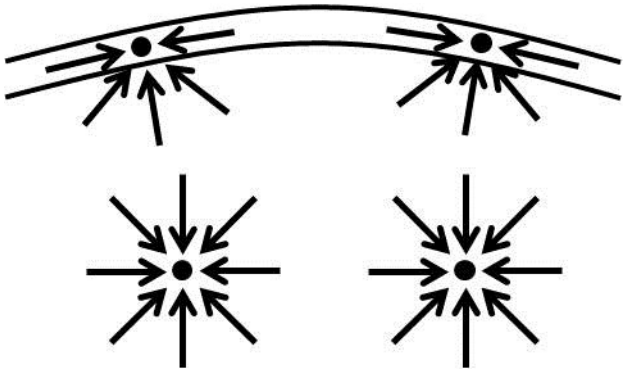

Figure 2.3. A simple diagram of forces acting on molecules in the bulk and at the surface of a drop. In order for the surface molecules to remain on the surface, an additional force, the surface tension, is required when compared to the molecules in the bulk. Arrows here denote application location of force on each molecule.

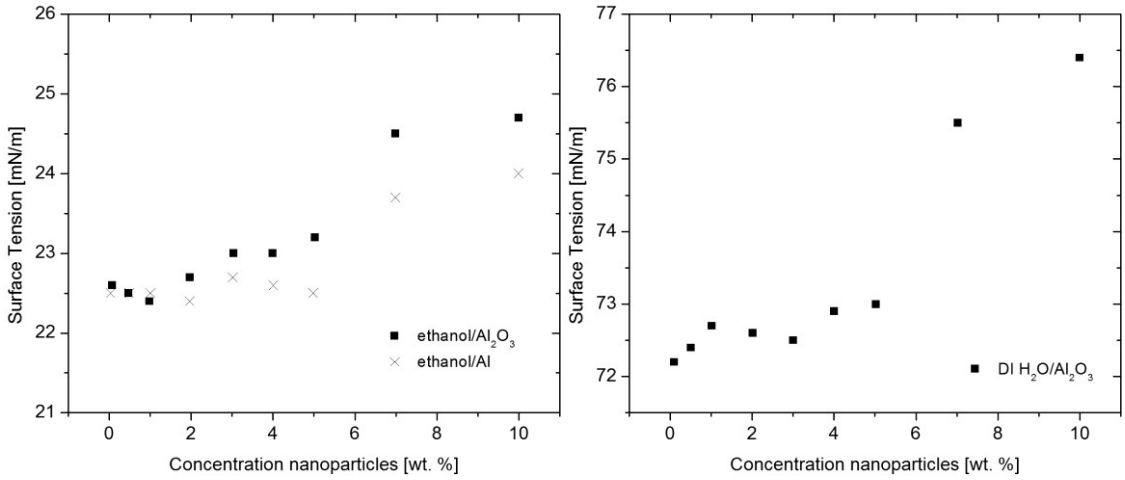

Figure 2.4. Nanofluid surface tension as a function of increasing nanoparticle concentration as reported by Tanvir et al. [26].

2.4 Nanofluid Boiling

Boiling heat transfer is a process involving fluid mechanics (surrounding geometry, and flow), phase change, and surface tension. These elements may be utilized in some combination in applications to impart increased heat transfer or storage in the form of the latent heat of evaporation. Studies of the influence of added nanoparticles on the

boiling process has been initially inconclusive, with studies finding both an increase or decrease in pool boiling heat transfer for nanofluids relative to the base liquids [2]. Relating to the present research on evaporation, the relationships between nanofluid boiling and evaporation may offer insight given the similarities of these processes.

Pure liquid boiling has three approximate regions of heat transfer. These regions, convection, nucleate boiling, and film boiling can be seen in Figure 2.5.. The quiet boiling region, which has the lowest heat flux of the three regions, is an extension of the convective heat transfer regime, and as such has heat transfer coefficients related to the temperature as proportional to the liquid-wall temperature difference:

$$h\ \alpha \begin{cases} \Delta T^{1/4} & laminar\ flow \\ \Delta T^{1/3} & turbulent\ flow \end{cases}. \tag{18}$$

As the solid surface temperature increases to higher values of superheat (the excess temperature above the liquid saturation temperature) the rate of bubble formation increases. These vapor bubbles initially form and collapse at low values of superheat, but with increasing wall superheat they detach and are convected upwards, in what is known as the nucleate boiling region. This region has heat transfer coefficients proportional to the cube of the temperature difference:

$$h\ \alpha\ \Delta T^3. \tag{19}$$

As the wall superheat is increased further, the rate of bubble departure increases to a level at which liquid changes phase so quickly, that the increasing majority of the solid surface interfaces with vapor. The vapor-surface contact causes a drastic decrease in heat transfer flux due to the vapor's decrease in thermal conductivity, when compared to the liquid. This departure from nucleate boiling is called the critical heat flux, or CHF, and is an area to avoid in all heat transfer applications. With increasing temperature difference, the decrease of the heat flux continues as more of the solid surface is covered in vapor until finally the system reaches the Leidenfrost point. At this point, the vapor layer has completely covered the surface of the solid, and radiation heat transfer begins to increase the heat flux with increasing superheat.

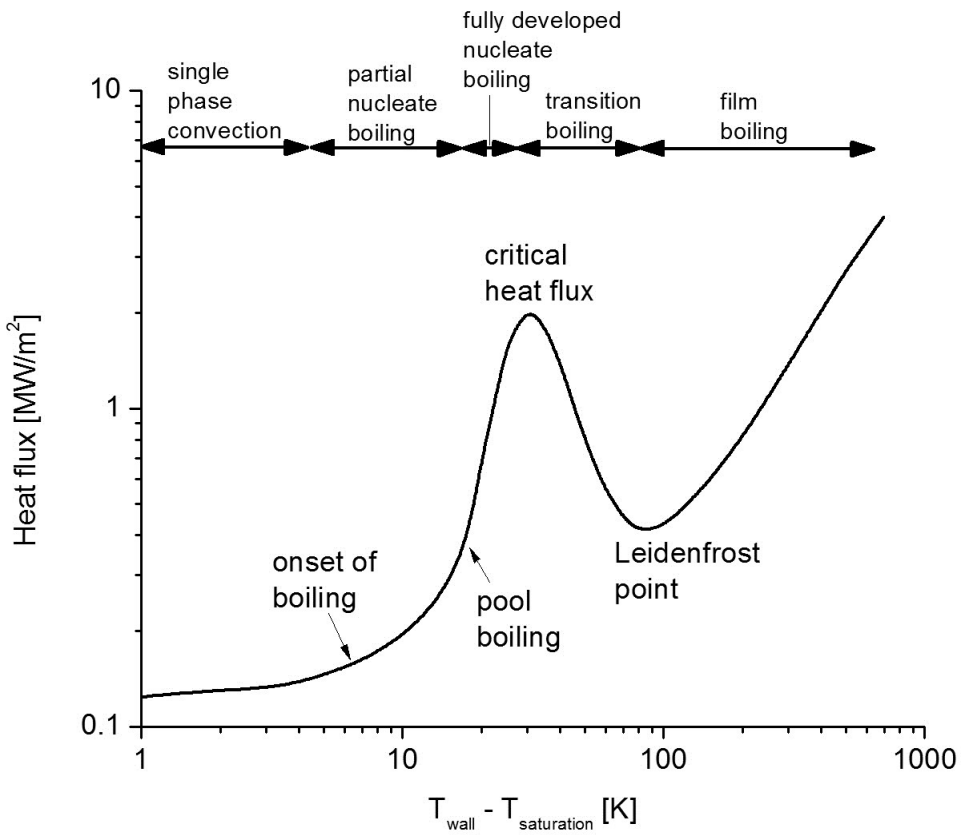

Figure 2.5. Boiling curve of a generic fluid, adapted from [31]. The three main regions of convection, nucleate boiling, and film boiling can be seen in this figure.

Following the description found in [2], this process of pure fluid boiling can be simply modeled using surface tension force balances and thermodynamic relations for phase transformation and heat transfer. Assuming a spherical bubble, we can estimate the equilibrium of pressure forces at its liquid-vapor interface:

$$P_g = P_l + \frac{2\sigma}{r}, \tag{20}$$

where P_g and P_l are the gas and liquid pressures, and σ and r are the surface tension and radius of the bubble. Again, note we have assumed a simple spherical bubble. We can then estimate the amount of excess heat required using the Clausius-Clapeyron relation. This relation relates the change of pressure and temperature with the latent heat, temperature, and change in specific volume of a system after phase change:

$$\frac{dP}{dT} = \frac{h_{fg}}{\Delta v_{fg} T_{sat}}. \tag{21}$$

Further simplifying to a discretization:

$$\frac{P_g - P_l}{T_l - T_{sat}} = \frac{h_{fg}}{\Delta v_{fg} T_{sat}}, \tag{22}$$

which when combined with the pressure equilibrium described above relates the superheat of a solid to a bubble radius:

$$T_l - T_{sat} = \frac{2\sigma \Delta v_{fg} T_{sat}}{r h_{fg}}. \tag{23}$$

This relation tells us, for fixed properties, that the smaller the bubble the larger the super heat required to maintain that bubble. Or that more energy is required for smaller bubble formation. From equation 23 it is possible to infer the general trends relating the boiling heat transfer characteristics of fluids.

The heat transfer rate of a boiling fluid can be estimated as by Mikic and Rohsenow [32] in 1969 as the combination of micro-layer evaporation, transient conduction during boundary layer formation (just after bubble departure) and natural convection induced by bubble motion:

$$q_{tot} = \frac{q_{evap} t_b + q_c t_w}{t_b + t_w} + q_{nc}\,, \tag{24}$$

where q_{evap} is the heat transfer rate due to thin layer evaporation found between a bubble and the solid surface, q_c is the conductive heat transfer rate due to the entrainment of localized hot fluid as a bubble departs, q_{nc} is the natural convection of the base fluid not influence by bubbles, and the times t_b and t_w are the bubble growth time scale, and the bubble waiting timescale.

The critical heat flux of a boiling fluid can be calculated based upon the Kelvin-Helmholtz instability of a viscous jet. Whalley [33] in 1996 proposed:

$$q_{CHF} = 0.149 h_{fg} \rho_g^{\frac{1}{2}} [\sigma(\rho_l - \rho_g) g]^{\frac{1}{4}}, \tag{25}$$

where the notations used are consistent with those above. This relation linearly relates increasing critical heat flux (CHF) with increasing latent heat of vaporization, and with the quartic root of surface tension.

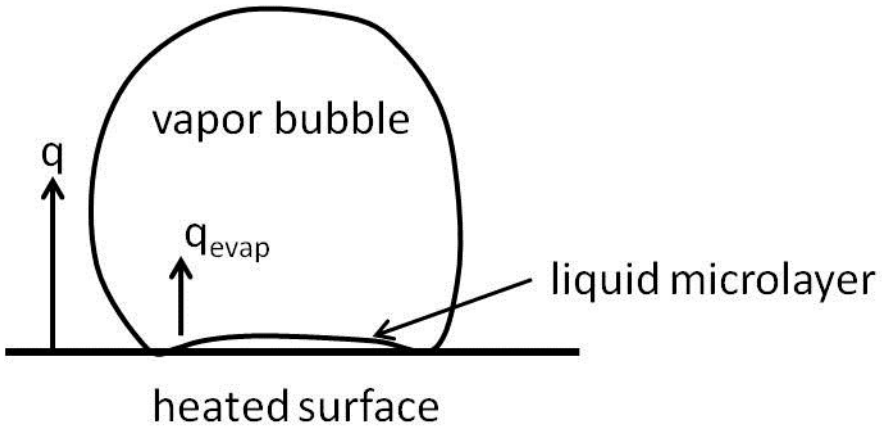

Figure 2.6. Microlayer heat transfer phenomena.

Initial experimental work in nanofluid boiling by Das et al. in 2003 [34] found that nanoparticle additions of Al_2O_3 on the order of 20 to 50 nm in diameter, decreased the boiling heat flux from that of the pure fluid (water) for nanoparticle addition at concentrations from 0.1 to 4% by weight. This was thought to be due to a change in surface roughness as a result of the nanoparticle surface interactions. In essence, the smaller nanoparticles were decreasing the initially rough (0.2 to 1.2 um) surface, and therefore reducing localized heat transfer (increasing the wall superheat through a reduction in nucleation points). At increasing concentrations, this reduction in surface roughness could even deposit a full layer of particles, further reducing surface heat transfer, and overcoming any potential increase in thermal conductivity. This decreased heat transfer could lead to potentially overheated surfaces in any application with potential for pool boiling. As is typical of nanofluid experimental results and repeatability, these boiling heat transfer reduction results are a strong function of preparation methodologies (surfactants, sonication, etc.), nanoparticle size, and concentration, making experiment-experiment comparisons and experimental interpretation difficult.

However, the general trends of experimental findings in the investigation of critical heat flux are in agreement. It has been reported, as summarized by Das et al. [2], that inclusion of nanoparticles in a base fluid will increase the CHF by 30 to 300%. These findings, observed at low nanofluid concentrations (<1%), allow for the dramatic

increase in heat transfer; allowing for the increase of wall superheat without risk of failure. Das et al. [2] noted that this has potentially major applications and benefits for power generation, particularly nuclear.

The increase in critical heat flux can be inferred from relation for pure fluid critical heat flux, equation 25. A factor of two increase in CHF requires either factor of two change in the latent heat of vaporization or an increase in the $\sigma(\rho_l - \rho_g)g$ term by a factor of 16. Such large changes in latent heat or the surface tension term seem unrealistic, indicating that it is probable the underlying physics are not yet understood.

2.5 Combustion Related Enhancements

The present research is partly motivated by the potential for combustion enhancement through the addition of energetic nanomaterials to liquid fuels. Nanomaterial additives can increase liquid fuel energy density and potentially modify the chemical reaction or burning rate through catalysis [35], increased mass transfer, and non-linear fluid-particle chemical interactions as noted by Jones et al. [36], [37]. Additionally, the added heat release due to metallic particle combustion, similar to that of solid rocket additives, is a sought benefit of nanofuel combustion. The potential for increased heat transfer or heat storage capability of a fuel is also beneficial, as many aircraft rely upon the sensible heat sink of the fuel to control component temperatures, as discussed in the author's M.S. thesis on endothermic fuels [38].

Increased thermal conductivity of nanofluid fuels may allow for a faster heating of the fuel, and therefore faster vaporization, mixing (promoting efficiency and reduced emissions), and chemical combustion of the fuel and nanoparticle additive. Additionally, nanoparticle additives have the potential to reduce the emissions of compression ignition engines by introducing new mixing and combustion mechanisms [35]. Ganesh and Gowrishankar [35] reported that the introduction of colbalt oxide (Co_3O_4) and magnalium (Al-Mg) nanoparticles at concentrations of 100 mg/l reduced NOx emissions of a Jatropha biodiesel engine on the order of 45% and 30%, respectively. They proposed that the reduction of emissions, and increase in efficiency, was the result of micro explosions within heated fuel drops causing increased mixing and therefore more stoichiometric combustion.

In 2013, Lenin et al. [39] also reported reduced emissions (primarily CO) from a diesel engine with the addition of manganese oxide (MnO) and copper oxide (CuO) nanoparticles to diesel fuel. This decrease in CO emissions is hypothesized to be due to the oxygen donors in the nanoparticle additives, which may help complete combustion under oxygen-deficient conditions that might otherwise terminate in CO. Additionally, Lenin et al. noted that a reduced ignition delay with nanoparticle addition was likely also occurring in-cylinder and further helped reduce pollutant emissions by increasing the duration of combustion, although the ignition time itself was not measured.

Conventional analysis of drop combustion depicts the combustion time for a fuel drop as the sum of the times for evaporation, mixing, and burning. As previously mentioned, studies have shown that nanofluids offer a favorable enhancement to the ignition probability and energy release under combustion conditions; however, little quantitative work has investigated the three phenomena (evaporation, mixing, and burning) in controlled isolation. In the present study, the timescale for the evaporation of nanofluid fuel drops is examined, as evaporation is often the rate controlling phenomenon in drop or spray combustion. Investigation into the relatively simple (1-D) room temperature evaporation of large drops will allow for a quantitative investigation necessary for the development of theories and models which can be extrapolated to combustion conditions (e.g., high pressure, temperature, and small drop size). Once the nanofluid evaporation physics are understood, the developed modeling advances can be incorporated into tools used to model multi-phase combustion which account for three-dimensional turbulent flow fields, chemical reactions, spray breakup, and drop interactions.

3. Theory of Nanofluid Pendant Drop Evaporation

Pure and colloid fluid evaporation was investigated experimentally and theoretically by Ranz and Marshal in the 1950s [40]. Their notable results provided experimental and theoretical background on the mechanisms behind millimeter sized drop evaporation, culminating in the well-known experimental heat transfer relation for forced convection on spheres [40]:

$$Nu = 2 + 0.6Re^{1/2}Pr^{1/3} \; for \; low \; Re \; and \; \mathrm{Pr} \; (< 200), \tag{26}$$

and eventually the D-square law:

$$\frac{dD^2}{dt} = -\frac{8\lambda_g}{\rho_l c_{p,g}} \ln\left(\frac{c_{p,g}\,(T_\infty - T_b)}{h_{fg}} + 1\right), \tag{27}$$

where the time of drop evaporation is estimated as:

$$t_d = \frac{D_o^2}{\left[\frac{8\lambda_g}{\rho_l C_{p,g}} \ln\left(\frac{c_{p,g}(T_\infty - T_{boil})}{h_{fg}}\right) + 1\right]}. \tag{28}$$

This law linearly related the drop lifetime to the square of the drop diameter, and is a strong function of ambient temperature, pressure, fluid vapor pressure, specific heat, and drop radius. This model was initially valid for stationary convection free drops, in thermal temperature equilibrium, as noted by Sirignano in his book on the fluid dynamics and transport of drops and sprays [41]. Charlesworth and Marshall [42] experimentally investigated the evaporation phases of colloid drops on the same size scale. They visually recorded the different phases of slurry drop evaporation, taking note of shell formation, and proposing the formation of an internal vapor bubble (all of which invalidates the pure fluid D-Squared model).

Since Ranz and Marshall's initial contribution [40] to drop evaporation, the use of their model in the combustion community has allowed for relatively simple analysis of many spray combustion processes (compression ignition, rocket engines, combustion stability). It has been expanded upon in greater detail, and complexity by the likes of Sirgniano [40] who has proposed theoretical methods for capturing gas phase combustion processes and boundary layer effects, unsteady heat transfer within the drop, and internal circulation among others. These theoretical methods retain the relative simplicity of the D-square law, compared to the required numerical computational

solution methods (e.g., computational fluid dynamics) required by the governing Navier-Stokes equations.

3.1 Full System of Equations

The governing equations for the compressible multi-species evaporation and burning of a drop are the continuity, momentum, energy, and species conservation equations. We will derive two sets of governing equations, for the liquid and gas phase analysis. Starting with gas phase analysis, the continuity and momentum equations are the classic Navier-Stokes equations under the assumption of a Newtonian fluid:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u_j)}{\partial x_j} = 0\,, \tag{29}$$

$$\frac{\partial (\rho u_i)}{\partial t} + \frac{\partial \left(\rho u_i u_j\right)}{\partial x_j} + \frac{\partial p}{\partial x_i} = \frac{\partial \tau_{ij}}{\partial x_j} + \rho g_i\,. \tag{30}$$

However, the energy equation includes radiation heat flux, Fourier heat flux, diffusive heat transfer of each species, and one step chemical reaction rate term also of each species, as derived by Sirignano [40]:

$$\overbrace{\frac{\partial (\rho h)}{\partial t} + \frac{\partial \left(\rho u_j h\right)}{\partial x_j}}^{Total\ Derivative} - \overbrace{\frac{\partial}{\partial x_j}\left(\lambda \frac{\partial T}{\partial x_j}\right)}^{Fourier\ H.F.} + \overbrace{\frac{\partial q_{rad,j}}{\partial x_j}}^{Radiation} + \overbrace{\frac{\partial}{\partial x_j}\left(\sum_{s-species} \rho V_j^s\, h_s Y_s\right)}^{Diffusive\ .of\ Species} = \overbrace{\frac{\partial p}{\partial t} + u_j \frac{\partial p}{\partial x_j}}^{Pressure\ Field} + \overbrace{\tau_{ij} \frac{\partial u_i}{\partial x_j}}^{\substack{Viscous \\ Dissipation}} + \overbrace{\sum_{s-species} \rho \dot{w}_s Q_s}^{Chemical\ Reaction}\,, \tag{31}$$

where the Stokes hypothesis for Newtonian fluids gives:

$$\tau_{ij} = \mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) - \frac{2}{3}\frac{\partial}{\partial x_j}\left(\mu \frac{\partial u_k}{\partial x_k}\right)\delta_{ij}, \tag{32}$$

and enthalpy h is the integral of a mixture of specific heats, through the linear combination equation:

$$h = \int_{T0}^{T} c_p dT = \int_{T0}^{T} \left(\sum_s Y_s c_{ps}(T) \right) dT = \sum_s Y_s h_s. \tag{33}$$

In addition to the classical Navier-Stokes equations, an additional conservation equation is required for each added species. Sirignano [40] has reported this as:

$$\overbrace{\frac{\partial(\rho Y_s)}{\partial t} + \frac{\partial(\rho Y_s u_j)}{\partial x_j}}^{Total\ Derivative} + \overbrace{\frac{\partial(\rho Y_s V_j^s)}{\partial x_j}}^{\substack{Diffusive \\ Mass\ Transfer}} = \overbrace{\dot{\rho w_s}}^{\substack{Chemical \\ Reaction \\ Mass\ Transfer}}, \tag{34}$$

$$s = 1\ to\ Species\ count$$

where V_j^s is the diffusion velocity in j of species s, Y_s is the mass fraction of species s, and u_j is the velocity in direction j. If we further follow the setup of Sirignano [40] and assume the diffusion rate is independent of direction, i.e. D_{ij} is equal to D_{ji}, and equal molecular weights for all species we arrive at the Fickian diffusion expression for the gas phase:

$$Y_s V_i^s = -D \frac{\partial Y_s}{\partial x_i}. \tag{35}$$

This assumption of Fickian diffusion in the scope of drop evaporation and burning has be justified by Megaridis and Sirignano [43]. Finally, we will also use the ideal gas relationship:

$$p = \rho R T. \tag{36}$$

For the liquid phase governing equations, we will start from the same governing Navier-Stokes relationships, but will be able to make assumptions regarding the flow speed (we will assume incompressible flow), radiation heat transfer, chemical reaction, and viscous dissipation of the problem, as proposed by Sirignano in [40]. These assumptions yield, denoting the liquid phase with the subscript l, for continuity (incompressible):

$$\frac{\partial u_{l,j}}{\partial x_j} = 0, \tag{37}$$

for momentum (incompressible):

$$\frac{\partial u_{l,i}}{\partial t} + \frac{\partial u_{l,j} u_{l,i}}{\partial x_j} \tag{38}$$
$$= -\frac{1}{\rho_l}\frac{\partial p}{\partial x_j} + \frac{\mu_l}{\rho_l}\frac{\partial^2 u_{l,i}}{\partial x_j^2}$$
$$+ \frac{1}{\rho_l}\frac{\partial \mu_l}{\partial x_j}\left(\frac{\partial u_{l,i}}{\partial x_j} + \frac{\partial u_{l,j}}{\partial x_j}\right) + g_{i,}$$

for energy (incompressible, neglecting radiation, chemical effects, and viscous dissipation):

$$\rho_l \frac{\partial e_l}{\partial t} + \rho_l u_{l,j}\frac{\partial e_l}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\lambda_l \frac{\partial T_l}{\partial x_j}\right), \tag{39}$$

and finally, for species conservation, again with Fickian diffusion (incompressible, Fickian diffusion):

$$\frac{\partial Y_{l,m}}{\partial t} + u_{l,j}\frac{\partial Y_{l,m}}{\partial x_j} = \frac{\partial}{\partial x_j}\left(D_l \frac{\partial Y_{l,m}}{\partial x_j}\right). \tag{40}$$

The solution to this final set of equations requires further assumptions on the flow pattern external and internal to the drop [40]. These flow assumptions are commonly axisymmetric flow or spherically-symmetric evaporation. The axisymmetric formulation allows for fluid flow internal of the drop, a circulation which can be solved for using a variety of methods, as touched upon by [40] can be seen in Figure 3.1. The spherically symmetric evaporation is the most basic formulation, and is the foundation for the simplified D-square law mentioned earlier. These different formulations will apply different sets of boundary conditions to the problem, and have varying advantages and disadvantages based upon the complexity of the solution process.

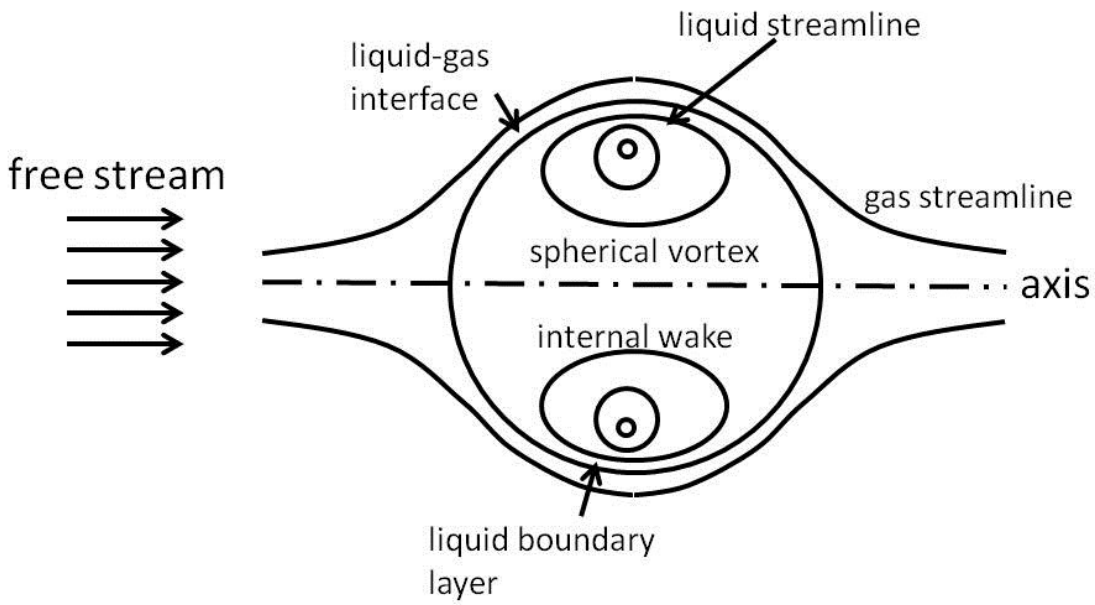

Figure 3.1. Example of axis symmetric velocity contours, adapted from Sirignano **[40]**.

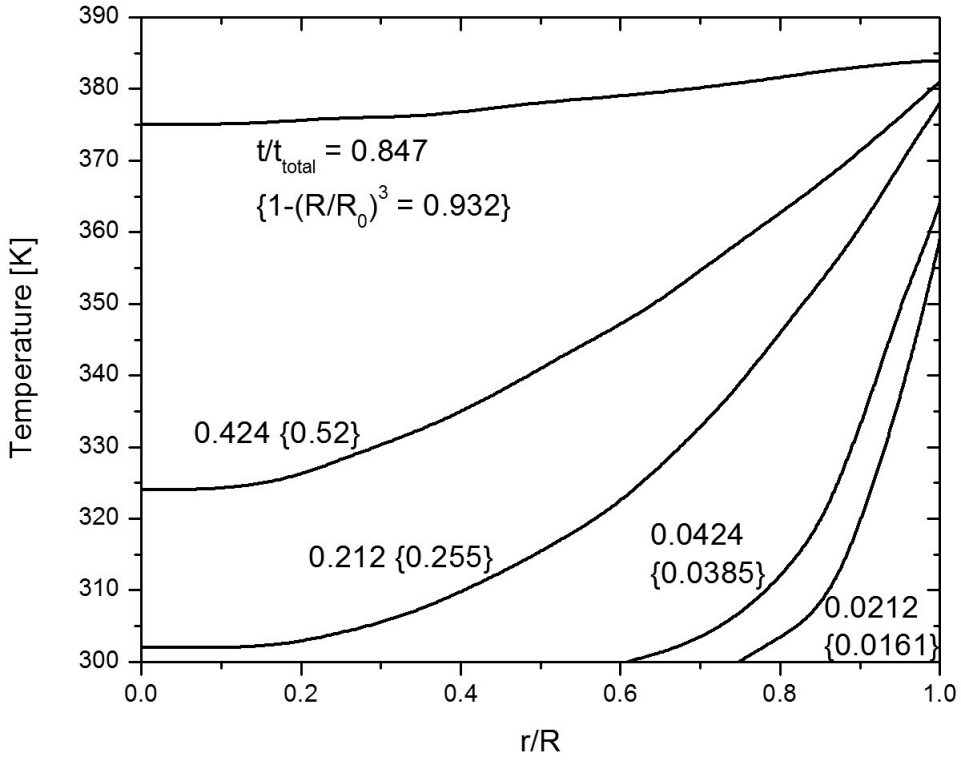

Figure 3.2. Generic spherically symmetric temperature profiles as a function of time into the drop lifetime, adapted from Sirignano **[40]**.

3.2 Spherically Symmetric Gas Phase Formulation

The governing equations for a spherically symmetric burning drop can be written as a specialized set of the general Navier-Stokes equations, as reported by Sirignano [40]. The initial physical problem, a liquid drop combusting in a non-zero mean flow, is first simplified to a drop of liquid combusting in a flow field of zero relative velocity within the drop (spherically symmetric). All gas phase velocity generated in this problem is radial (problem dimension is 1-D), and is assumed to be a consequence of the drop evaporation. We will first consider the gas phase, and separately formulate the liquid phase problem. We will neglect gravity (and therefore buoyancy effects), viscosity and assume a spatially-uniform pressure field. The gas phase momentum equation is [40]:

$$\frac{\partial \rho r^2 u}{\partial t} + \frac{\partial \rho r^2 u^2}{\partial r} + r^2 \frac{\partial p}{\partial r} = \frac{4r^2}{3}\frac{\partial}{\partial r}\left(\mu \frac{\partial u}{\partial r} - \frac{\mu u}{r}\right) + 4r^2 \mu \frac{\partial}{\partial r}\left(\frac{u}{r}\right), \tag{41}$$

and after applying Sirignano's [40] assumptions:

$$\frac{\partial \rho r^2 u}{\partial t} + \frac{\partial \rho r^2 u^2}{\partial r} = 0 \tag{42}$$

The continuity, energy, and species conservation equations for the gas phase problem can be written as [40]:

continuity:

$$r^2 \frac{\partial \rho}{\partial t} + \frac{\partial}{\partial r}(\rho u r^2) = 0, \tag{43}$$

energy:

$$\frac{\partial}{\partial t}\left(\rho r^2 c_{p_{mix}} \Delta T\right) + \frac{\partial}{\partial r}\left(\rho u r^2 c_{p_{mix}} \Delta T\right) = \frac{\partial}{\partial r}\left(\lambda r^2 \frac{\partial T}{\partial r}\right) + \frac{\partial}{\partial r}\left(\sum_{species} \rho D c_{p_s} \Delta T \frac{\partial Y_s}{\partial r}\right) + r^2 \frac{\partial p}{\partial t} - \rho r^2 Q \dot{w}_f, \tag{44}$$

species conservation (fuel):

$$\frac{\partial}{\partial t}\left(\rho r^2 Y_f\right) + \frac{\partial}{\partial r}\left(\rho u r^2 Y_f\right) - \frac{\partial}{\partial r}\left(\rho D r^2 \frac{\partial Y_f}{\partial r}\right) = \rho r^2 \dot{w}_f, \tag{45}$$

species conservation (oxidizer):

$$\frac{\partial}{\partial t}(\rho r^2 Y_o) + \frac{\partial}{\partial r}(\rho u r^2 Y_o) - \frac{\partial}{\partial r}\left(\rho D r^2 \frac{\partial Y_o}{\partial r}\right) = \rho r^2 \dot{w}_o = \frac{\rho r^2 \dot{w}_f}{\upsilon}, \tag{46}$$

species conservation (nitrogen):

$$\frac{\partial}{\partial t}(\rho r^2 Y_N) + \frac{\partial}{\partial r}(\rho u r^2 Y_N) - \frac{\partial}{\partial r}\left(\rho D r^2 \frac{\partial Y_N}{\partial r}\right) = 0, \tag{47}$$

species conservation (products):

$$\frac{\partial}{\partial t}(\rho r^2 Y_P) + \frac{\partial}{\partial r}(\rho u r^2 Y_P) - \frac{\partial}{\partial r}\left(\rho D r^2 \frac{\partial Y_P}{\partial r}\right) = -\rho r^2 \dot{w}_f \left(\frac{\upsilon + 1}{\upsilon}\right), \tag{48}$$

species conservation (total):

$$Y_f + Y_o + Y_N + Y_P = 1. \tag{49}$$

With this formulation, we have assumed that there is a uniform pressure field in space, constant specific heat, invisid gas fluid flow, Fickian mass diffusion, Fourier heat conduction, and single-step chemistry (fuel + oxidizer → products). This current formulation is also valid for non-burning drops (the current study), where the rate of fuel consumption is set to zero, $\dot{w}_f = 0$.

3.3 Spherically Symmetric Liquid Phase Formulation

The spherically symmetric assumptions of Sirignano [40] greatly simplify the liquid phase equations. We assume a stagnant drop interior, with uniform pressure which immediately reduces the momentum and conservation equations to trivial solutions. The liquid phase energy and species conservation equations can then be written as [40]:

$$\rho_l \frac{\partial}{\partial t}(r^2 e_l) - \frac{\partial}{\partial r}\left(\lambda_l r^2 \frac{\partial T_l}{\partial r}\right) = 0, \tag{50}$$

and

$$\frac{\partial}{\partial t}(r^2 Y_{l,s}) - \frac{\partial}{\partial r}\left(r^2 D_l \frac{\partial Y_{l,s}}{\partial r}\right) = 0. \tag{51}$$

These equations allow for variation in time and space of both liquid temperature and species concentration.

The above equations have the potential to model unsteady and spatially varying heating of the liquid phase during drop evaporation, as can be seen in Figure 3.2. This analysis is required when the drop lifetime timescale of interest is on is on the order of the drop heating time.

3.4 Spherically Symmetric Drop Evaporation Solutions

Further classification of the D-square law can result from assumptions regarding the temporally and spatially varying temperature profile within the liquid drop. Sirignano [40] highlights these different solution methods in his book. If steady-state drop evaporation is estimated to occur, then the liquid phase of the drop can be neglected, otherwise, a solution using the spherically symmetric liquid phase equations is required.

3.5 Steady State Heat flux (No Drop Heating)

If we start from the spherically symmetric formulation of drop evaporation, the simplest solution results from a one component, quasi-steady, spatially invariant temperature assumption in the liquid phase. This assumption dictates that all energy coming from the gas phase is transferred directly into the change of phase of the liquid phase (there is no drop heating). Mathematically, this takes the form of a boundary condition on heat transfer at the drop surface, as presented by Sirignano [40]:

$$\dot{q}_{g,cond} = \dot{m} h_{fg}, \tag{52}$$

where $\dot{q}_{g,cond}$ is the gas phase conduction to the drop surface, $\dot{m}$ is the mass flow rate of the evaporating species, and h_{fg} is the latent heat of vaporization of the evaporating species. In this formulation, we neglect the entire liquid phase as it has no impact on the problem.

Additionally, we assume diffusion of heat, and mass within the gaseous phase occurs at the same rate. This assumption corresponds to a Lewis number of one, where the Lewis number is defined as:

$$Le = \frac{\alpha}{D} = \frac{k}{\rho c_p D}, \tag{53}$$

where α is thermal diffusivity, D is the diffusion coefficient, k the thermal conductivity, c_p the specific heat, and ρ the density. This assumption allows for the solution of only one equation, neglecting either temperature or concentration, where the solution of one will be proportionally applied to the other through a Spalding number as described by Sirignano [40]. The previously defined steady continuity equation:

$$\frac{\partial}{\partial r}(\rho u r^2) = 0, \tag{54}$$

along with the spherically symmetric energy equation without our current assumptions:

$$\begin{aligned}\frac{\partial}{\partial t}\left(\rho r^2 c_{p_{mix}} \Delta T\right) + \frac{\partial}{\partial r}\left(\rho u r^2 c_{p_{mix}} \Delta T\right) \\ = \frac{\partial}{\partial r}\left(\lambda r^2 \frac{\partial T}{\partial r}\right) + \frac{\partial}{\partial r}\left(\sum_{species} \rho D c_{p_s} \Delta T \frac{\partial Y_s}{\partial r}\right) \\ + r^2 \frac{\partial p}{\partial t} - \rho r^2 Q \dot{w}_f,\end{aligned} \tag{55}$$

where the assumptions described by Sirignano [41] transform the relation to:

$$\frac{\partial}{\partial r}\left(\rho u r^2 c_{p_{mix}} \Delta T\right) = \frac{\partial}{\partial r}\left(\lambda r^2 \frac{\partial T}{\partial r}\right). \tag{56}$$

From the continuity equation, we know the mass transfer from the surface is steady:

$$\dot{m} = \rho u r^2. \tag{57}$$

This now modifies the energy equation to:

$$\frac{\partial}{\partial r}\left(\dot{m} c_{p_{mix}} \Delta T\right) = \frac{\partial}{\partial r}\left(\lambda r^2 \frac{\partial T}{\partial r}\right), \tag{58}$$

where we can use the product rule rule to achieve:

$$\dot{m} c_{p_{mix}} \frac{\partial T}{\partial r} = \frac{\partial}{\partial r}\left(\lambda r^2 \frac{\partial T}{\partial r}\right), \tag{59}$$

which can be solved analytically:

$$T(r) = \frac{(T_\infty - T_b) e^{-\frac{c_p \dot{m}}{4\pi k r}} - T_\infty e^{-\frac{c_p \dot{m}}{4\pi k R}} + T_b}{1 - e^{-\frac{c_p \dot{m}}{4\pi k R}}}. \tag{60}$$

Applying the assumed relationship at the drop surface to the radial derivative of the previously solved temperature expression yields the mass rate of evaporation:

$$\dot{m} = \frac{4\pi \lambda_g R}{c_{p,g}} \ln(1 + B), \tag{61}$$

or, due to a unity Lewis number from Sirignano [40]:

$$\dot{m} = 4\pi R \rho_g D \ln(1 + B), \tag{62}$$

where the Spalding number is:

$$B = \frac{c_{p,g}\,(T_\infty - T_b)}{h_{fg}}, \tag{63}$$

gas phase thermal conduction is:

$$\dot{q}_{g,cond} = 4\pi r^2 \lambda_g \left.\frac{\partial T}{\partial r}\right|_R, \tag{64}$$

and the previously mentioned temperature gradient is:

$$\left.\frac{\partial T}{\partial r}\right|_R = \frac{c_p \dot{m}}{4\pi k R^2}\left\{\frac{(T_\infty - T_b)e^{-\frac{c_p \dot{m}}{4\pi k R}}}{1 - e^{-\frac{c_p \dot{m}}{4\pi k R}}}\right\}. \tag{65}$$

This relation for the mass transfer rate can be used to find the estimated drop lifetime as from Sirignano [40]:

$$m_{sphere} = \rho_l \frac{4}{3}\pi\left(\frac{D}{2}\right)^3, \tag{66}$$

from which we can estimate the rate of change of the drops mass:

$$\frac{dm}{dt} = \rho_l \frac{\pi}{6} 3D^2 \frac{dD}{dt}, \tag{67}$$

pairing this with the steady state solution:

$$\frac{4\pi\lambda_g R}{c_{p,g}} \ln(1 + B) = \rho_l \frac{\pi}{6} 3D^2 \frac{dD}{dt}, \tag{68}$$

furthermore, using the inverse chain rule, and rearranging the D-square law is evident:

$$\frac{dD^2}{dt} = -\frac{8\lambda_g}{\rho_l c_{p,g}} \ln(B + 1). \tag{69}$$

In the case of a non-unity Lewis number, the equations can be re-cast as described by Sirignano [40]:

$$\dot{m} = \frac{4\pi\lambda_g R}{c_{p,g}} \ln(1 + B_M), \tag{70}$$

where B_M is the Spalding mass transfer number:

$$B_M = \frac{Y_{Fs} + \nu Y_{O_2,\infty}}{1 - Y_{Fs}}. \tag{71}$$

In this formulation, Y_{FS} is the local mass fraction of fuel at the drop surface, $Y_{O_2,\infty}$ is the local mass fraction of oxygen at infinity, and ν is the stoichiometric fuel to oxidizer mass ratio.

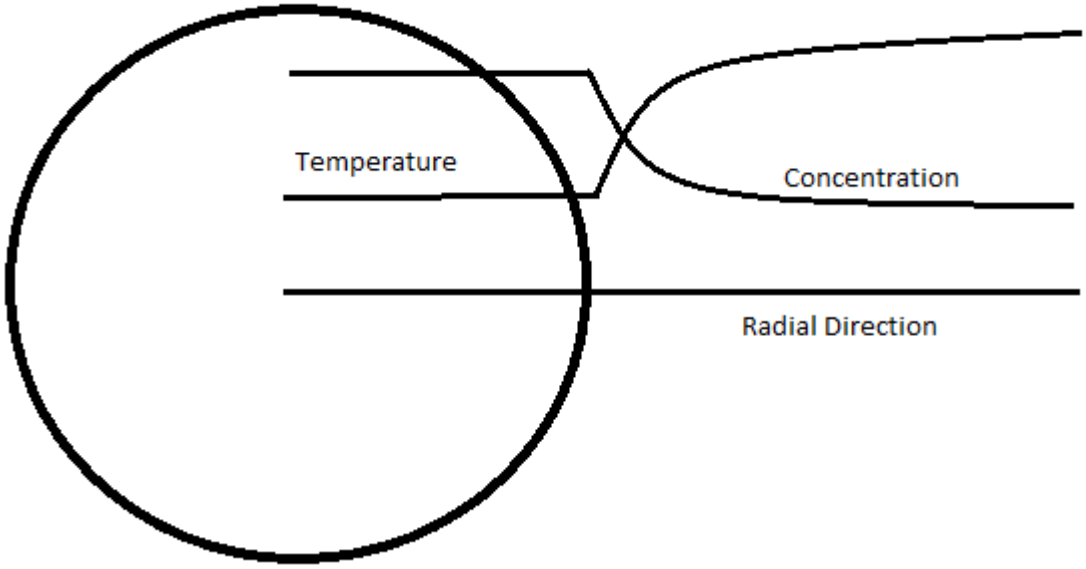

Figure 3.3. Generic quasi-steady spherically-symmetric evaporating drop temperature and concentration profiles.

3.6 Spherically Symmetric Unsteady Liquid Heating

It is also possible to allow for liquid heating in the formulation of the governing equations for a spherically-symmetric evaporating drop. This can take the form of a both time and spatially varying temperature profile within the liquid, while the gas phase analysis is much the same as for the steady solution (the Lewis number is still unity). Allowing for liquid heating modifies the heat balance at the drop surface from Sirignano [40]:

$$\dot{q}_{g,cond} = \dot{m}h_{fg} + \dot{q}_l, \tag{72}$$

as before, $\dot{q}_{g,cond}$ is the gas phase conduction to the drop surface, $\dot{q}_l$ is the liquid phase heat transfer to the drop interior, $\dot{m}$ is the mass flow rate of the evaporating species, and h_{fg} is the latent heat of vaporization of the evaporating species. Here, it is possible to

see that for a given gaseous heating rate, an increase in drop interior heating will reduce the evaporation rate, therefore the problem can vary greatly depending on the magnitude of drop heating. This liquid heating term couples with the liquid energy governing equation:

$$\rho_l \frac{\partial}{\partial t}(r^2 e_l) - \frac{\partial}{\partial r}\left(\lambda_l r^2 \frac{\partial T_l}{\partial r}\right) = \dot{q}_l. \tag{73}$$

Finally, as a result of allowing drop heating, the solution for the evaporation of the drop will no longer be closed form; an iterative solution will be required.

From this heat balance, it is possible to further classify our desired liquid phase solution as steady or unsteady and uniform or varying temperature field within the drop. A uniform unsteady (assume temperature is constant in R) expression allows for the calculation of the radially constant temperature as the bulk sensible heat of the drop from Sirignano [40]:

$$\dot{q}_{l,sensible} = \rho_l \frac{4}{3} \pi R^3 c_{p,l} \frac{dT_l}{dt} \tag{74}$$

where ρ_l is the liquid density, R the quasi steady radius, $c_{p,l}$ the liquid specific heat, and $\frac{dT_l}{dt}$ the differential temperature. If we further discretize the equation using finite differences we can step through this solution using a given initial temperature, and a discrete timestep, Δt. The selection of this timestep is primarily determined in practice through the stability of the solution routine used. Following which the timestep's influence on simulation accuracy should be investigated.

3.7 Solution Methods

The solution of the full liquid equations requires a numerical approach. In many cases it is desirable to avoid the need for a numerical solution method, and as such simple analysis of the timescales of interest will help validate assumptions such as the radially constant or zero drop heating (D-square law) assumptions. Proposed by Sirignano [41], the ratio of the drop heating time to the drop life time will help gauge whether or not the uniform temperature assumption applies. From Sirignano [40] this ratio is estimated as:

$$\frac{t_H}{t_L} \sim \frac{2\lambda_g c_{p,l} \ln(1+B)}{\lambda_l c_{p,g}}. \tag{75}$$

Sirignano's ratio was found based upon the assumption that the time required for thermal diffusion into the drop center is:

$$t_H \sim \frac{\rho_l c_l R^2}{\lambda_l}, \tag{76}$$

and that the drop lifetime is estimated through the D-square law:

$$t_L \sim \frac{\rho_l R^2}{2\rho_g D \ln(1+B)}. \tag{77}$$

Using this ratio, we are able to gauge the importance and applicability of various assumptions made in the solutions methods presented. When drop heating is much less than the drop lifetime, it is safe to use the simplified assumption of bulk sensible heating. This is typical for large (millimeter) diameter drops. Typically in the combustion environment the drop heating and lifetime are on the same order, leading to the required numeric solution to the liquid phase energy equation.

In cases in which the drop heating is much less than the drop lifetime, Sirignano [40] proposed that a further classification can be made. The ratio of time required for the bulk liquid phase to reach the steady state temperature (when the drop temperature causes the heat flux due to evaporation to equal the gas phase conduction) to the drop heating is:

$$\frac{t_*}{t_H} \sim \frac{1}{3}\frac{\lambda_l}{\lambda}\left(\frac{1}{\ln\frac{1+vY_{O\infty}}{1-Y_{F_s}}}\right)\left[\frac{vY_{O\infty}+Y_{F_s}}{1-Y_{F_s}+\frac{h_{fg}}{c_p R}\frac{Y_{F_s}}{T_S^2}\left(\frac{1+vY_{O\infty}}{1-Y_{F_s}}\right)}\right]. \tag{78}$$

where λ_l and λ are the liquid and gas phase thermal conductivity, v is the fuel to oxidizer ratio, $Y_{O\infty}$ and Y_{F_s} are the oxidizer and fuel mass fractions at infinity and the surface of the drop, h_{fg} is the heat of vaporization, c_p the specific heat of the liquid, R the drop radius, and T_s the liquid saturation temperature.

Using this relationship, we are able to gauge further the applicability of our solution method. If the steady state time is of the same order as the drop heating time, which are both much less than drop lifetime, the D-square law is applicable. However, if the steady state time is greater than the drop heating time (which is much less than the

drop lifetime), but still less than or equal to the drop lifetime, use of the bulk sensible relationship is applicable. If the drop heating time is greater than the drop lifetime, we must use the fully nonlinear temperature relationship and a numeric solution method.

3.8 Proposed Theory of Nanofluid Drop Evaporation

The addition of nanoparticles changes property evaluation and introduces a new species equation to the pure fluid evaporation equations. The inclusion of a diffusion equation within the liquid phase will account for gradient driven nanoparticle motion in an assumed spherically symmetric drop. Effects of the nanoparticles properties such as density, specific heat, viscosity, and thermal conductivity are included where appropriate.

Nanofluid viscosity is ignored in a spherically symmetric formulation, as we assume no internal fluid motion. Inclusion of the modified viscosity, and therefore a model of nanofluid viscosity would be required for any model using non-zero liquid fluid flow. This will become important for forced convection drop evaporation models, and models where fluid flow is on the order of the diffusion velocity:

$$u \sim \frac{D}{R}. \tag{79}$$

Addition of nanoparticles will provide an enhancement to the thermal conductivity of the fluid. This enhancement will only serve to increase the applicability of the D-square law, or the sensible heat assumption model. As such, if these models are already valid for a pure fluid, the need to include the liquid phase non-uniform temperature modeling is reduced through the addition of nanoparticles.

The specific heat of the nanofluid will increase (in general, if the nanoparticle specific heat is larger than the fluid) as a result of the ideal mixture of the liquid and nanoparticles. Thus the inclusion of the modified specific heat may be warranted in the sensible heat model. However, in practice this value is very small due to the very small weight percentage of nanoparticles added (typically 0-3 wt%).

Modification of nanofluid density can also be included in a sensible type model. However, as with the specific heat, percentages of nanoparticle typically only change the density by a small amount.

It is proposed that the model most advantageous for room temperature nanofluid drop evaporation is the sensible heat model. This model will allow for the transient temperatures induced from the changing nanoparticle concentration on the drop surface throughout evaporation. Additionally, this will allow the model to be independent of any thermal conductivity modification, and various other nanofluid properties which entail newer, less tested models for property evaluation, thus eliminating unnecessary uncertainties.

3.8.1 Agglomeration

Particle agglomeration is expected to occur to some extent in the drop evaporation experiment. This further grouping of particles will change the nanofluid properties as a function of time. Previously, population balance models have been used to model the particle size distribution with time and space for colloids by Gan and Qiao in [44], [45]. As the nanofluid ages within the drop, particle agglomerate size is expected to change the maximum surface concentration and therefore further affect drop evaporation rate. Data taken with a Scanning Electron Microscope (SEM) is used to approximate the initial conditions of this agglomeration process.

Population balance models proposed for agglomeration vary in complexity. As used by Gan and Qiao [44], [45] for drop studies, and investigated by Meng et al. [46] the rate of change of particle number groupings n_i with respect to time can be written as in equation 80. This model further assumes three body collisions are rare, Brownian motion dominates, particle agglomerates form spheres and that there is no breakup ($S_i = \Gamma_{ij} = 0$), repeated from Gan and Qiao [44]:

$$\frac{dn_i}{dt} = \frac{1}{2}\sum_{j=1}^{i-1} \alpha_{j.i}\beta_{j,i-j}n_j n_{i-j} - \sum_{j=1}^{i-1} \alpha_{ij}\beta_{ij}n_i n_j - S_i n_i + \sum_{j=1}^{n} \Gamma_{ij} S_j n_j, \tag{80}$$

where n_i and n_j the number of agglomerates in previously determined number bins, $\alpha_{j.i}$ is the collision efficiency, β is the collision frequency, S_i is the breakup rate, and Γ_{ij} is distribution of agglomerate fragments.

In this model, collision frequency β can be written as a consequence of the assumed spherical nature of particles and the Einstein-Stokes relation as by Gan and Qiao [44]:

$$\beta_{ij} = \frac{2k_B T}{3\mu} \frac{\left(r_i + r_j\right)^2}{r_i r_j}, \tag{81}$$

with variables representing the Boltzmann constant k, temperature T (Kelvin), viscosity μ, and particle radius r of grouping i.

However, as this model does not capture agglomerate breakup or any non-spherical agglomeration it may be required to increase the model complexity. One possible method has been described in detail by Kim and Kramer in [47] explicitly for use in fractal colloids. Additionally, solution methods for population balance equations are discussed in detail by Kumar [48].

Inclusion of this time varying effect will be determined through timescale analysis. Like drop heating, the timescale of agglomeration is compared to the drop lifetime as derived from the Smolucheowski equations for diffusion limited agglomeration from Lin et al. in 1990 [49]:

$$t_{agg} \sim \frac{3\mu V}{8k_B T N_0}. \tag{82}$$

Here μ is the fluid viscosity, N_0 the initial number of particles, V the initial volume of the system, k_B the Boltzmann constant, and T the temperature. This timescale is the time required for an increase in the average agglomerate mass, and can therefore be used to infer the rate of diffusion limited agglomeration for nanofluid drop evaporation. Finally, the ratio of the two timescales can be estimated as:

$$\frac{t_{agg}}{t_L} \sim \frac{\frac{3\mu \frac{4}{3}\pi R^3}{8k_B T N_0}}{\frac{\rho_l R^2}{2\rho_g D \ln(1+B)}} = \frac{\mu(T_l)\pi R \rho_g D \ln\left(1 + \frac{c_{p,g}\left(T_\infty - T_l\right)}{h_{fg}}\right)}{k_B T_l N_0 \rho_l}. \tag{83}$$

When the ratio shown is small (less than 1), inclusion of agglomerate modeling is required. As can be seen, the ratio is 8.1 for ethanol in air at standard conditions, a 0.1 volume fraction of 60 nm radius particles in a 1 mm spherical drop, meaning that the total lifetime of the ethanol drop is approximately eight times smaller than characteristic agglomeration time of the particles. In the current experiment, we exclude more than half of the evaporation time, at least doubling this ratio, and allowing the model to assume a constant agglomerate shape, size, and fractal distribution during the current simulation. It is interesting to see here that it is the ratio of the gas phase gradient (within the Spalding number) to the liquid drop temperature which determines the relative importance of agglomeration. As drop radius, liquid viscosity, gaseous density and diffusion rate increase, the drop lifetime decreases, reducing the relative importance of agglomeration.

3.8.2 Diffusion Rate of Nanoparticles

The diffusion rate of a nano-sized particle in a liquid can be estimated through use of the Einstein-Stokes relation. Using this relation, an assumed spherical particle is in a low Reynolds number flow where viscosity dominates. With these assumptions, the drag is calculated through the Stokes spherical drag relation as:

$$F_D = 6\pi\mu r, \tag{84}$$

where μ is the fluid viscocity, r is the effective particle radius, and is used with the Einstein diffusion relation:

$$D = \frac{k_B T}{\lambda}, \tag{85}$$

where λ is the force inhibiting diffusion (i.e. drag) ,k_B the Boltzmann constant, and T the temperature. The final approximation for the diffusion rate of nanoparticles in a liquid can be seen as:

$$D = \frac{k_B T}{6\pi\mu r}. \tag{86}$$

3.8.3 Fractal Properties

Sensitive to the preparation method and stability of the nanofluid, the fractal dimension, d_f, of the nanofluid will determine the maximum allowable solid volume,

and therefore maximum obtainable mass fraction within the nanofluid. This maximum volume/mass fraction will determine the point at which the surface concentration is saturated and no longer changes, which reduces the rate of change of evaporation rate. A larger allowable mass or volume fraction will lengthen the range in which evaporation rate is diminished with added nanoparticles.

However, this effect is in practice dwarfed by the effect of the initial concentration loading. If an initially high (compared to the max allowable) amount of nanoparticle is suspended in the fluid as evaporation begins, the effect of this transient increasing concentration at the surface is diminished, and we expect a linear decrease in squared diameter with time (another D-square law). If the initial concentration is low compared to the max allowed, the drop will remain in a period in which the evaporation rate is dominated by the increasing surface concentration, resulting in a non-linear change of the diameter squared with time (departure from a D-square law).

The maximum possible volume fraction of nanoparticles can be estimated by assuming a loosely packed structure of spheres, a volume fraction of π/6. Within these spheres are fractal nano-agglomerates as described by Prasher et al. in [19] and [23], with volume fraction as:

$$\phi_{internal} = \left(\frac{R_g}{a}\right)^{df-3}. \tag{87}$$

The agglomerate radius R_g, nanoparticle radius a, and degree of freedom of the agglomerate are used as parameters in this calculation. From literature and theory, values for the degree of freedom of colloidal nanoparticle agglomerate range from 1.8 to 2.7 as reported by Prasher et al. [19] and Kanniah et al. [50]. The total maximum volume fraction possible is then the product of the two volume fractions:

$$\phi_{max} = \phi_{int} * \left(\frac{\pi}{6}\right). \tag{88}$$

The effect of varying fractal dimension and agglomerate radius on the maximum volume fraction according to the above relations can be seen in Figure 3.4., and Figure 3.5..

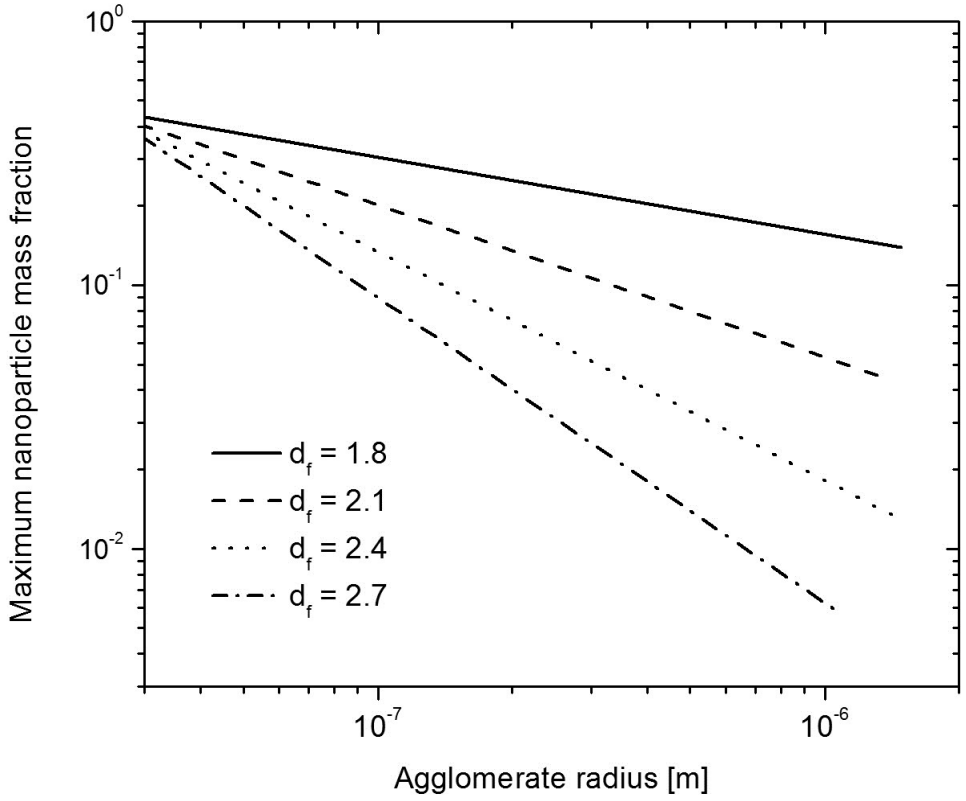

Figure 3.4. Maximum mass fraction of nanoparticles as a function of fractal dimension (d_f) and agglomerate size.

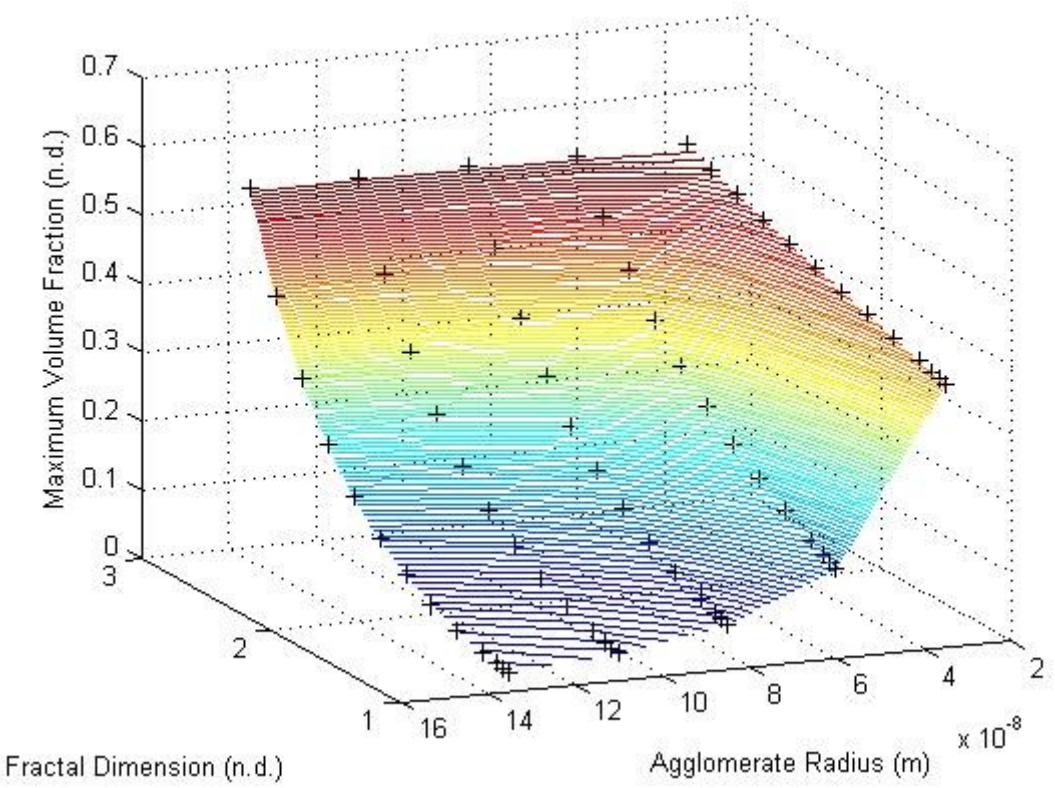

Figure 3.5. Three dimensional figure equating the maximum volume fraction to fractal dimension and agglomerate radius. In this current study, nanoparticle radius was fixed at 30 nm.

4. Pendant Drop Evaporation Experiment

The goal of the experimental work presented here is to provide quantitative information on nanofluid pendant droplet evaporation for development of mechanistic understanding and modeling treatment of the influence of added nanoparticles on drop evaporation.

4.1 Experimental Procedures

The evaporation of nAl/ethanol nanofluid pendant droplets was monitored using a Ramé Hart goniometer (model 250). The goniometer, when coupled with the Ramé Hart DROPimage Advanced software [52], provides grayscale images of backlit suspended pendant droplets and, following automated image processing utilizing a contrast driven surface finding routine, provides droplet geometry and surface tension. The goniometer has ~10 μm spatial resolution and for the present study a 1 Hz image acquisition rate was used. A photograph of the goniometer is shown in Figure 4.1.

The goniometer collects images of the drop at an acquisition rate of 1 Hz in the present study and the data analysis software reports pendant drop geometry and surface tension as a function of time. Figure 4.2 exhibits a drop image and Figure 4.3 and example experimental data file in the format produced by the DROPimage software.

Portions of this chapter previously appeared as:
W. J. Gerken, A. V. Thomas, N. Koratkar and M. A. Oehlschlaeger, "Nanofluid pendant droplet evaporation: Experiments and modeling," *Int. J. Heat Mass Transfer,* vol. 74, pp. 263-268, July 2014.

W. J. Gerken, M. A. Oehlschlaeger, "Nanofluid Pendant Droplet Evaporation", in Proceedings of the ASME 2013 Summer Heat Transfer Conference, Minneapolis, MN, 2013, p. V001T03A018.

Figure 4.1. Ramé Hart goniometer used for droplet evaporation rate and surface tension measurements: camera on left, needle for droplet suspension in middle, and backlight on right. Repeated with permission from [51].

Figure 4.2. An image captured by the goniometer and used for determination of drop geometry and surface tension.

No.	Time	Gamma	Beta	R0	Area	Volume	Theta	Height	Width	Opt	Messages
1	0.028	34.15	0.185	0.804	7.00	2.23	118.07	1.362	1.663	2	
2	0.942	33.73	0.187	0.803	6.97	2.22	117.97	1.358	1.661	2	
3	1.950	33.92	0.185	0.802	6.94	2.21	117.93	1.355	1.659	2	
4	2.952	34.10	0.184	0.801	6.91	2.19	117.80	1.349	1.657	2	
5	3.941	34.31	0.183	0.800	6.87	2.18	117.63	1.344	1.656	2	
6	4.942	34.27	0.182	0.799	6.84	2.16	117.61	1.339	1.654	2	
7	5.942	34.20	0.182	0.798	6.80	2.14	117.34	1.334	1.651	2	
8	6.945	34.27	0.181	0.797	6.77	2.13	117.29	1.329	1.647	2	
9	7.958	34.18	0.181	0.795	6.73	2.12	117.13	1.325	1.645	2	
10	8.954	34.30	0.180	0.794	6.70	2.10	117.10	1.319	1.643	2	
11	9.950	34.27	0.180	0.793	6.66	2.09	116.87	1.314	1.642	2	
12	10.95	34.52	0.178	0.792	6.63	2.07	116.79	1.309	1.639	2	
13	11.94	34.43	0.178	0.791	6.59	2.06	116.59	1.303	1.636	2	
14	12.95	34.60	0.177	0.790	6.56	2.04	116.51	1.298	1.632	2	
15	13.94	34.30	0.178	0.789	6.53	2.03	116.36	1.295	1.630	2	
16	14.95	34.53	0.176	0.788	6.51	2.02	116.32	1.292	1.628	2	
17	15.95	34.61	0.175	0.788	6.48	2.01	116.23	1.287	1.626	2	
18	16.94	34.38	0.176	0.787	6.45	2.00	116.07	1.284	1.625	2	
19	17.96	34.80	0.174	0.786	6.43	1.99	116.06	1.280	1.623	2	
20	18.95	33.85	0.177	0.784	6.39	1.98	115.86	1.277	1.620	2	
21	19.95	34.51	0.174	0.784	6.37	1.97	115.75	1.272	1.621	2	
22	20.95	34.85	0.172	0.783	6.34	1.95	115.60	1.267	1.618	2	
23	21.95	34.34	0.174	0.782	6.30	1.94	115.41	1.262	1.615	2	
24	22.94	34.77	0.172	0.781	6.27	1.93	115.39	1.257	1.612	3	
25	23.94	34.52	0.172	0.780	6.23	1.91	115.09	1.252	1.610	4	

Figure 4.3. Example Rame Hart DROPimage software output. R0 is the drop radius (unites of mm) and gamma is the surface tension (units of mN/m). Time has units of seconds.

4.1.1 Sample Preparation

Ethanol (99.5% from Aldrich) was chosen as a base liquid due to the stability of nAl in ethanol without surfactants [45] and the relatively high vapor pressure of ethanol. Aluminum nanoparticles from Sky-Springs Nanomaterials (99.9%, 40-60 nm diameter, spherical morphology) were used; a scanning-electron microscope (SEM) image of a dry nAl powder sample is shown in Figure 4.4. The Al nanoparticles were exposed to air and therefore have an oxidized (Al_2O_3) shell. The Al nanoparticles were suspended in ethanol at concentrations of 0, 0.1%, 0.27%, 1%, and 3.06% nAl by weight. Nanoparticles were first hand mixed into ethanol, followed by sonication for 10-30 minutes with a duty cycle of 10 seconds of mixing followed by 5 seconds off, as recommended by Gan and Qiao [44]. The nanofluids were stable, with no visual sedimentation, for a minimum of 3 hours at the highest nAl concentrations (3.06%) and for several days at the lowest concentrations (0.1%).

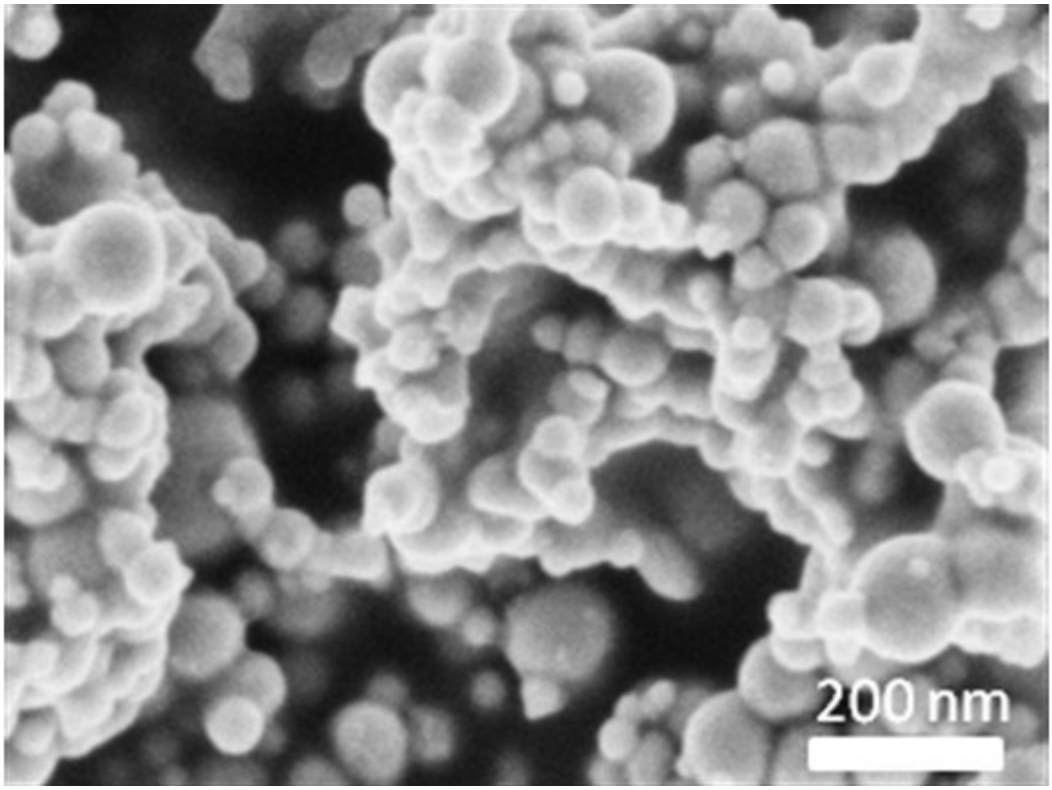

Figure 4.4. SEM image of dry nAl powder. Repeated with permission from [51].

4.1.2 Sample Agglomeration

The important geometric properties of the average agglomerate, radius of gyration R_g and fractal dimension d_f, used in modeling calculations presented in Chapter 5 were determined from analyzing SEM images of aluminum nanoparticle agglomerates from nanofluids (1.0% and 3.06% nAl/ethanol, see Figure 4.5) that were spread on a glass substrate and fully evaporated. SEM images were analyzed using a box counting method (FracLac/ImageJ) which returned a distribution of particle Feret diameter and fractal dimension [53]. The recovered distribution of Feret diameter is shown in Figure 4.6. An average radius of gyration was defined as half the average Feret diameter, R_g = 125 nm, and an average fractal dimension was determined to be d_f = 1.7 from a measured distribution of d_f = 1.6-2. These "average" agglomerate properties were used in the Chapter 5 modeling calculations throughout the droplet spatial domain at all time steps. Hence, further nanoparticle agglomeration during droplet evaporation was not accounted for and, as will be shown in Chapter 5, the modeling calculations slightly over predict the measured nanofluid evaporation rates.

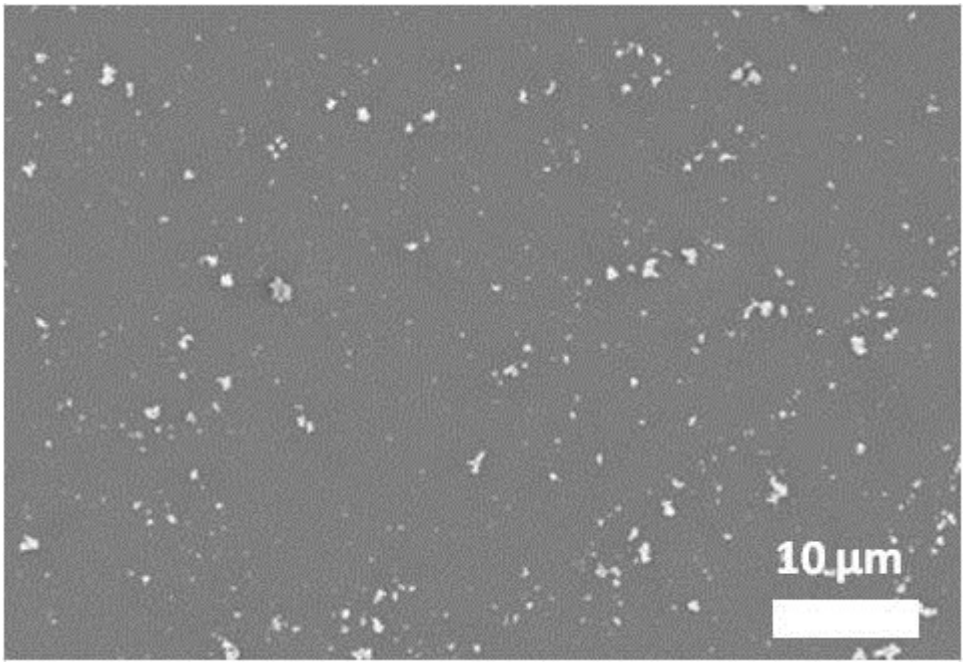

Figure 4.5. Example SEM image used for determining agglomerate Feret diameter and fractal dimension. Repeated with permission from [51].

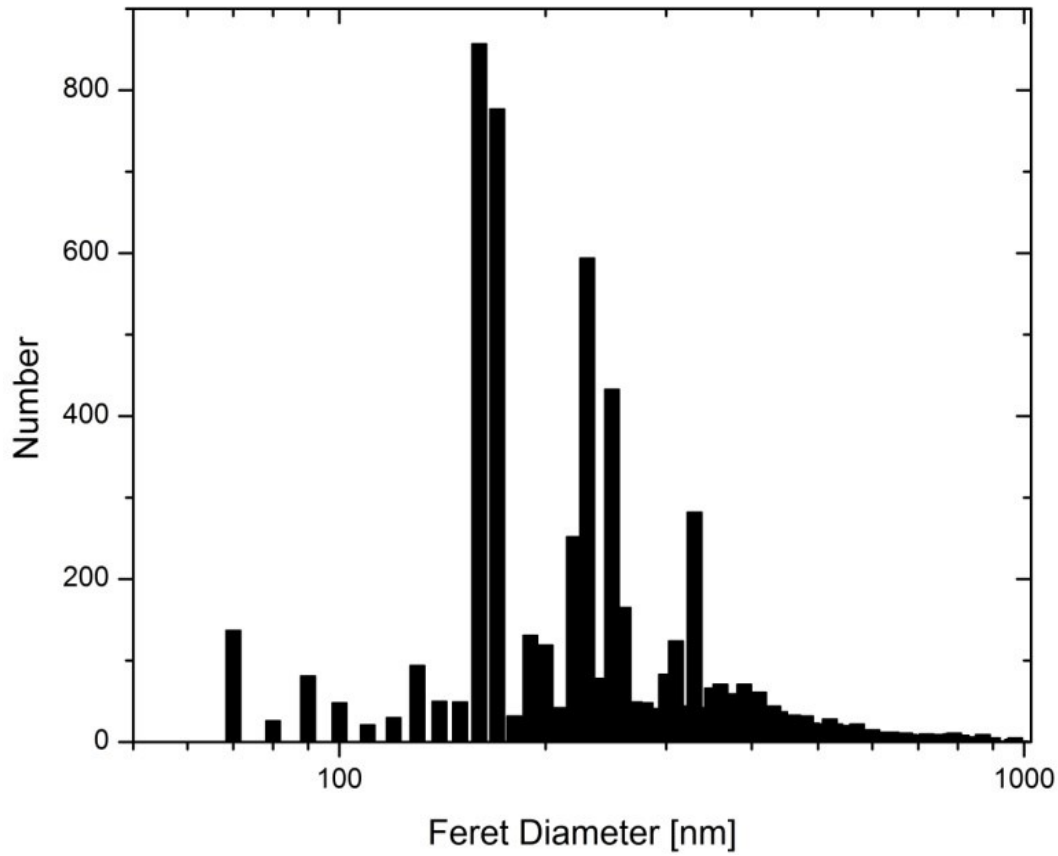

Figure 4.6. Distribution of Feret diameter for 5867 particles. Peaks are at diameters of approximately 170, 250, and 330 nm. The average diameter is 250 nm which was used in the Chapter 5 modeling simulations. Repeated with permission from [51].

4.1.3 Data Collection

Droplets of approximately 2 mm in diameter were suspended by pipette from a 0.1 mm stainless steel wire with a ~0.5 mm diameter sphere at the tip. The wire was cleaned with ethanol between experiments and periodically replaced. It can also be neglected as a major source of heat transfer, as shown by Ranz and Marshall [40] and Law et al. [54]. Once the droplet was suspended on the wire, it evaporated in the room-temperature (297 K), low-humidity (35-45% relative humidity), no air flow (natural convection) controlled laboratory environment. Law et al. [54] have shown little dependence of the evaporation rate of ethanol droplets of similar size to those studied here in room temperature air for relative humidity from 0% to approximately 50%.

Example experimental results for the time histories of the diameter squared and surface tension for pure ethanol and a 1% nAl/ethanol nanofluid are shown in Figures 4.7 and 4.8, respectively. The nAl/ethanol nanofluid has reduced pendant droplet

evaporation rate and indistinguishable surface tension relative to pure ethanol. Within experimental resolution, the evaporation rate in all experiments was observed to obey the D-square law (i.e., a constant slope on the Figure 4.7 axes) for approximately the first half of the droplet lifetime ($D^2/D_0^2 > 0.5$):

$$\frac{D^2}{D_0^2} = 1 - \frac{k}{D_0^2}t \tag{89}$$

where D is the droplet diameter [mm], D_0 the initial droplet diameter [mm], k the evaporation rate [mm^2/s], and t time [s].

At longer times the droplets become unstable (i.e., move on the wire) making geometry and surface tension determinations uncertain. The instability is more pronounced and begins earlier in the droplet lifetime for higher concentration nanofluids (i.e., the 1% and 3% nAl/ethanol nanofluids) because the nanoparticles begin to agglomerate and form a porous shell at the droplet surface; shell formation has also been observed by Javed et al. [55]. For the present study, we choose to only consider the early time period where $D^2/D_0^2 > 0.75$ and pendant droplets for all nanofluids are far from instability and shell formation and evaporation rate and surface tension measurements are unambiguous.

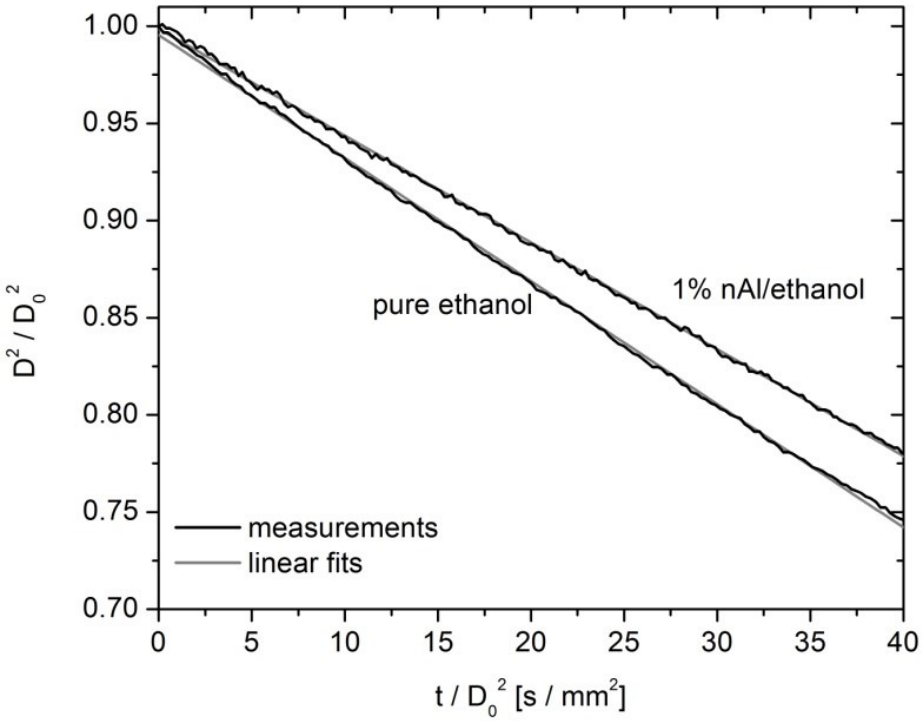

Figure 4.7. Example evaporation rate measurements for pure ethanol and a 1% nAl/ethanol nanofluid. Axes are normalized by the initial droplet diameter squared, D_0^2, so that the slope of the linear fit is the evaporation rate k [units: mm^2/s]. Repeated with permission from [51].

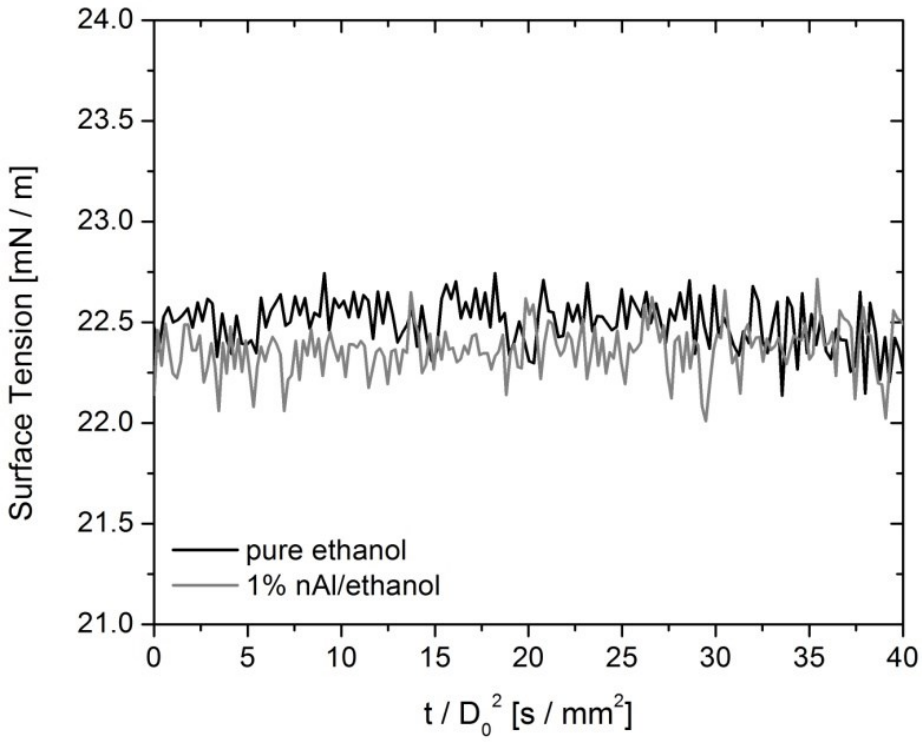

Figure 4.8. Example surface tension measurements for pure ethanol and a 1% nAl/ethanol nanofluid. Repeated with permission from [51].

For each nanofluid mixture (0, 0.1, 0.25, 1.0, and 3.06% nAl by weight in ethanol), ten droplet evaporation experiments were performed. The average D-square evaporation rate k and surface tension for each nanofluid is reported in Figure 4.9 with error bars representing two standard deviations in the data. Comparisons are made to a pendant droplet evaporation rate measurement made for pure ethanol by Law et al. [56] and pendant droplet surface tension measurements made for nAl/ethanol nanofluids by Sefiane and Bennacer [11] and Tanvir and Qiao [26] in Figure 4.10; additionally, the accepted value for the surface tension of pure ethanol [56] is given for comparison.

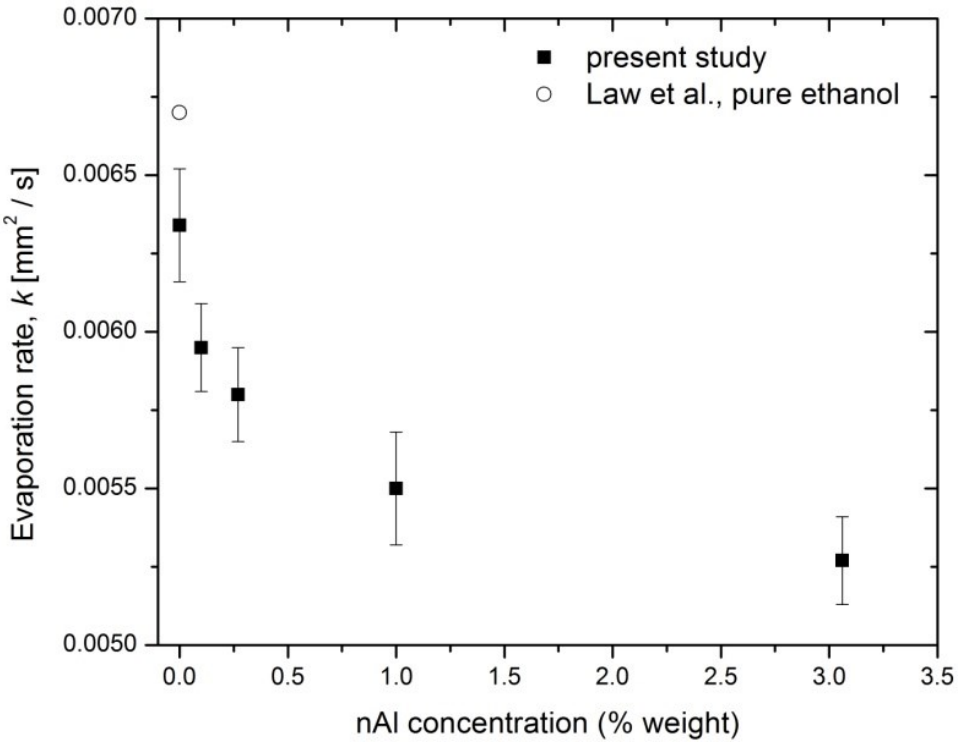

Figure 4.9. Evaporation rate measurements for nAl/ethanol nanofluids. Repeated with permission from [51].

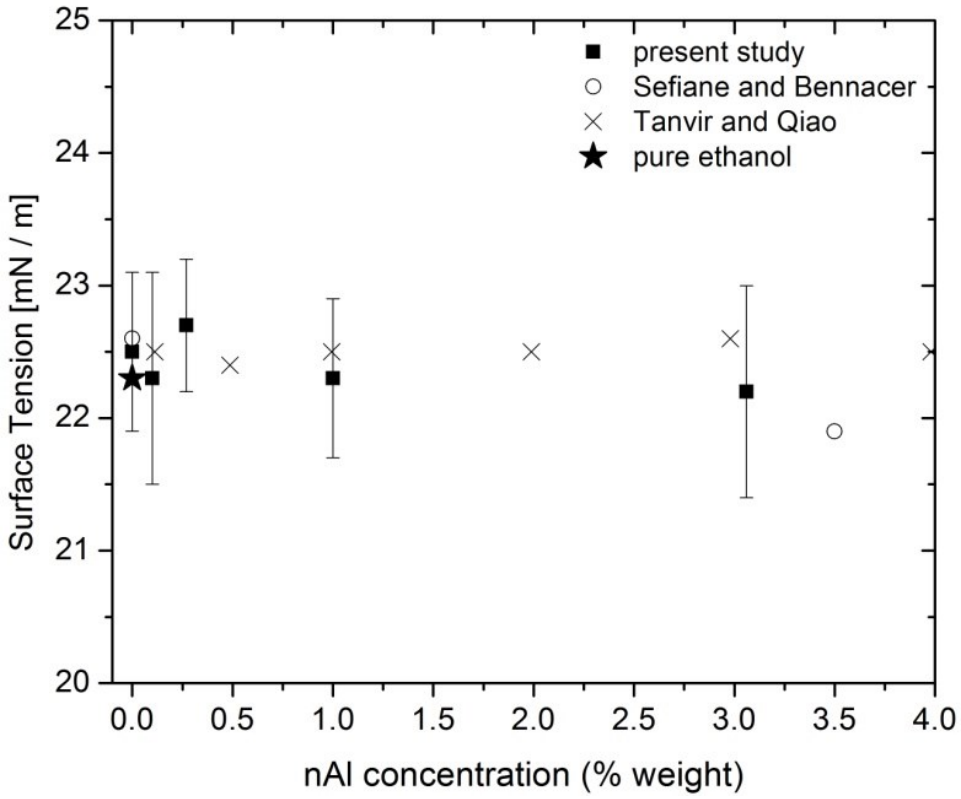

Figure 4.10. Surface tension measurements for nAl/ethanol nanofluids. Repeated with permission from [51].

While prior measurements suitable for direct comparison to the present study are not available in the literature, in Figure 4.11 the nAl/ethanol evaporation rate measurements are compared to: 1) pendant droplet evaporation rate measurements reported by Javed et al. [55] for nAl/n-heptane nanofluids at a temperature of 200 °C (well in excess of the boiling point of n-heptane, 98 °C) and 2) room temperature sessile droplet evaporation rate measurements by Sefiane and Bennacer for 5% nAl/ethanol [11]. Because these measurements were carried out with different configurations and under different conditions, the comparisons are indirect and require normalization. For comparison with the experiments of Sefiane and Bennacer, sessile droplet surface evaporation rates (mm^2/s) were extracted from their measurements for the time period prior to depinning of the sessile droplet contact line (i.e., the time period over which the droplet-surface contact angle reduces, while the droplet base diameter remains constant). The indirect comparisons made in Figure 4.11 show relative agreement for the evaporation rate reduction with increased nAl loading, lending confidence to the present measurements.

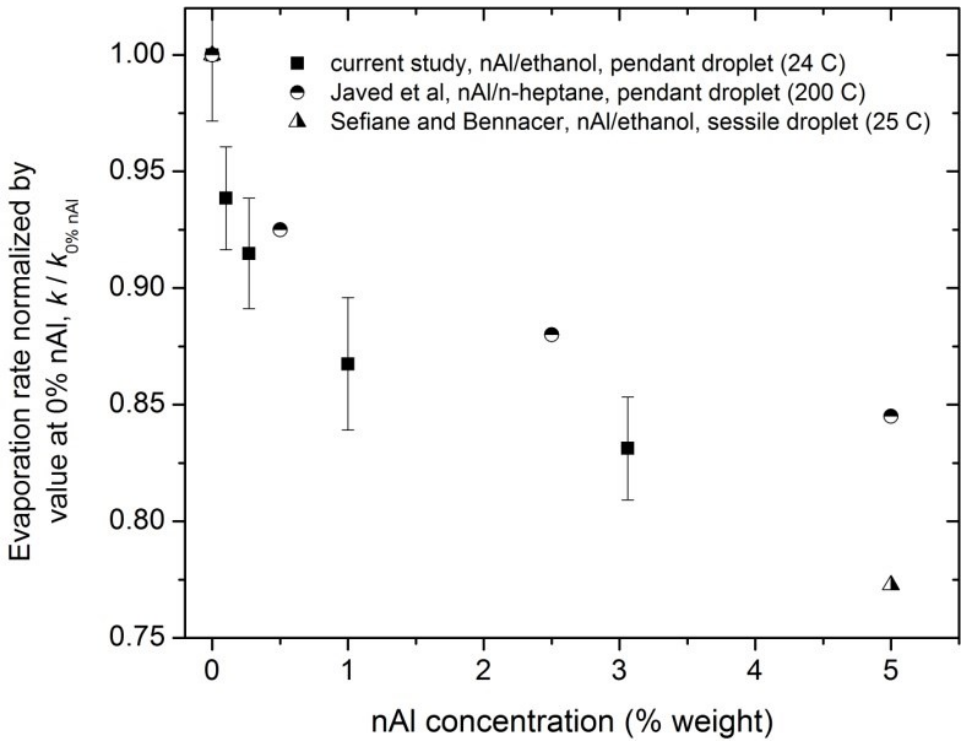

Figure 4.11. Normalized evaporation rate for nanofluids containing nAl: comparison of present study with Javed et al. [57] and Sefiane and Bennacer [11]. Repeated with permission from [51].

5. Pendent Drop Modeling and Simulation

5.1 General Solution Routine

Starting with the one-dimensional diffusion, continuity, and temperature equation the derivation of the D-square law assumes steady evaporation, one-dimensional variation of temperature and concentration in the gas phase, and vapor-liquid equilibrium at the drop surface (drop internal heat flux is zero). Repeated below from [40], the equations for concentration:

$$\frac{\partial Y_i}{\partial t} = D_i \left(\frac{\partial^2 Y_i}{\partial r^2} + \frac{2}{r} \frac{\partial Y_i}{\partial r} \right), \tag{90}$$

continuity:

$$r^2 \frac{\partial \rho}{\partial t} + \frac{\partial}{\partial r} (\rho u r^2) = 0 \ \therefore \tag{91}$$

$$\rho u r^2 = cons. = \frac{\dot{m}}{4\pi}, \tag{92}$$

and temperature:

$$\frac{\partial T}{\partial t} = \alpha \left(\frac{\partial^2 T}{\partial r^2} + \frac{2}{r} \frac{\partial T}{\partial r} \right), \tag{93}$$

provide the practical basis for the current liquid phase solution method.

Through relaxing the zero drop heat flux, as described previously, a numeric solution for the liquid phase temperature as a function of time and space can be found. Brenn, Deviprasath, and Durst [57] and [58] use the classical vapor phase solution and a spatially constant and temporally changing drop temperature to allow for bulk drop heating in a numeric routine (which corresponds to the analytic D-square solution, once equilibrium is reached), as initially proposed by Abramzon and Sirignano [59]. The assumption of uniform temperature is only valid for specific thermal loadings, where drop lifetime is much greater than drop heating time which is the case in the experiments described above.

For cases where the spatially constant drop temperature does not hold, it is possible to estimate the spatial and temporal evolution of the liquid temperature using finite differences. For example, Jiang and Homer [60], [61], [62] used finite difference to model both liquid and gas phase transport during solution drop evaporation. However,

the use of finite differences causes difficulties with mass conservation due to the constantly changing boundary and node placements during the drop surface regression. This requires a very small numerical time step and the requisite computational power.

The current solution method for nanofluid pendant drop evaporation involves a modification to a simple approach to pure fluid evaporation, which relies on the analytic solution for the gas phase temperature and concentration, assumes a uniform drop temperature, and utilizes a time-stepping routine to solve for the drop temperature, as described by Sirignano [59], and implemented by Brenn, Deviprasath, and Durst [57], [58] more recently for a pure, or mixed fluid. These assumptions and the current method are valid for the experimental conditions considered here where liquid-phase heat diffusion is fast relative to drop evaporation. The detail for these assumptions and the full derivation is described previously in Chapter 3.

A modification to the pure fluid method is required to describe the evaporation of nanofluids because, while heat diffusion is fast in a nanofluid drop relative to the evaporation rate, the rate of nanoparticle diffusion (actually nanoparticle agglomerate diffusion) inside a nanofluid drop is not fast and must be resolved. To resolve the internal diffusion of nanoparticle agglomerates a finite difference routine has been implemented.

5.2 Gas Phase Solution Modeling

Analysis of the drop interface presents us with the essence of the drop evaporation phenomenon. Heat transfer from the gas phase is applied to either drop heating or evaporation. Key to the D-square analytic solution is the assumption that drop heating is negligible and, therefore, all energy transferred from the gas to the drop is transferred into evaporation. Coupling the continuity equation with the assumption that gas phase mass and heat transfer occur at the same rate (Lewis number is unity) the result is a solution is shown in:

$$\dot{m} = 4\pi r \rho D \, ln(1 + B), \tag{94}$$

and,

$$B = \frac{Y_{Fs} - Y_{F\infty}}{1 - Y_{Fs}} = \frac{C_p(T_\infty - T_s)}{L}. \tag{95}$$

Further analysis of the temperature and mass gradients in the gas phase allow for the definition of the mass and temperature Spalding transfer numbers (B_M and B_T). Upon relaxation of the unity Lewis number assumption, these terms can be related to one another, via specific heats and the Sherwood, Lewis, and Nusselt numbers, as defined by Abramazon and Sirignano [59]:

$$B_M = \frac{Y_{Fs} - Y_{F\infty}}{1 - Y_{Fs}}, \tag{96}$$

$$B_T = \frac{C_p(T_\infty - T_s)}{L + q_L/\dot{m}}, \tag{97}$$

and

$$B_T = (1 + B_M)^{\phi} + 1, \tag{98}$$

where

$$\phi = \left(\frac{C_{pF}}{C_{pg}}\right)\left(\frac{Sh}{Nu}\right)\left(\frac{1}{Le}\right). \tag{99}$$

This decoupling of non-unity Lewis number solutions allows for a forced convective gas phase solution to be found. However, in the present study the ratio of the Sherwood and Nusselt numbers is assumed to be unity.

Using the Clausius-Clayperion relation,

$$X_g = X_l \exp\left(\frac{h_{fg}}{R}\left(\frac{1}{T_{boil}} - \frac{1}{T_s}\right)\right), \tag{100}$$

at the surface temperature to find the surface vapor concentration,

$$Y_i = \frac{X_i MW_i}{\sum_j X_j MW_j}, \tag{101}$$

it is possible to calculate the Spalding mass transfer number, and therefore mass flow from the drop as described by Equations 95 and 94 respectively. However, it is desirable to calculate the heat transfer to the liquid drop, which is expected to affect the evaporation rate as described in detail in Chapter 3. Using the relation between mass and heat transfer Spalding numbers allows for the solution of heat transfer to the drop, $\dot{q}_L$, from equation 97, written explicitly as:

$$\dot{q}_l = \dot{m}\left\{\frac{c_p(T_\infty - T_s)}{B_T} - L\right\}, \tag{102}$$

and

$$\dot{q}_l = \frac{4}{3}\pi r^3 \rho_l c_{pl} \frac{dT}{dt}. \tag{103}$$

Results for the evaporation rates of pure fluids, shown in Figure 5.1., were obtained using the above analysis procedures match well with the experimental and computational results of Ranz and Marshall [40], Jiang [61], [62], and Law et al. [54] for both water and ethanol evaporation.

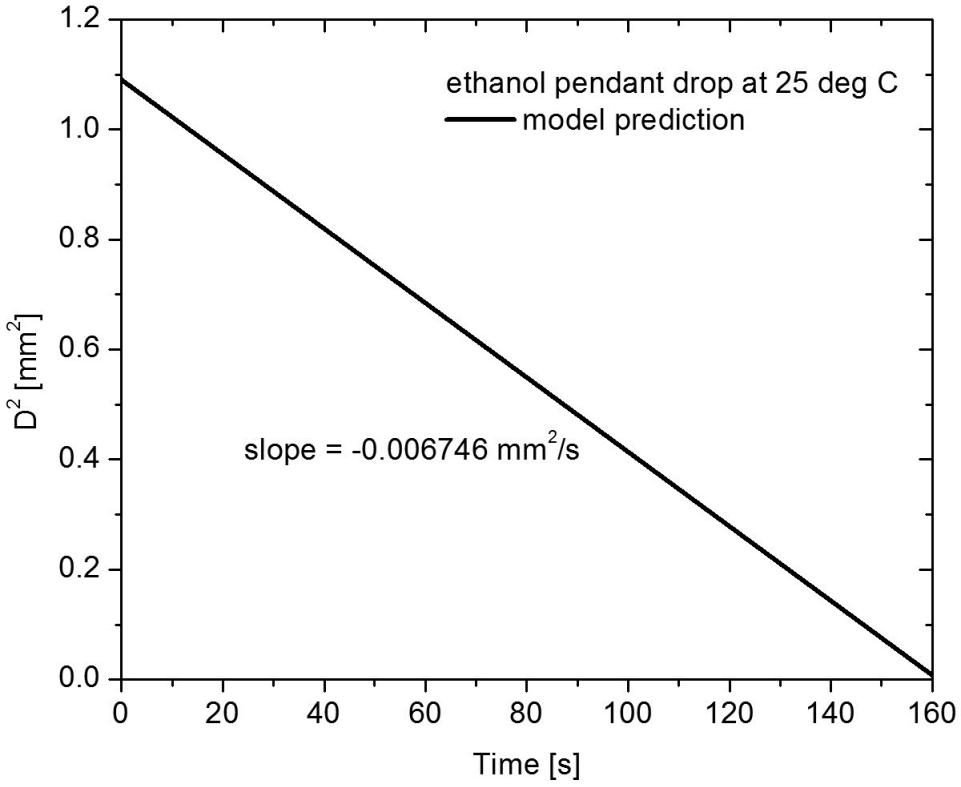

Figure 5.1. Pure ethanol pendant drop evaporation prediction using the current model. The modeled evaporation rate is 0.006746 mm^2/s. Law et al. [54] measured an evaporation rate of 0.0067 mm^2/s for the same conditions and in the present study an evaporation rate of 0.00635 mm^2/s was measured for the same condition.

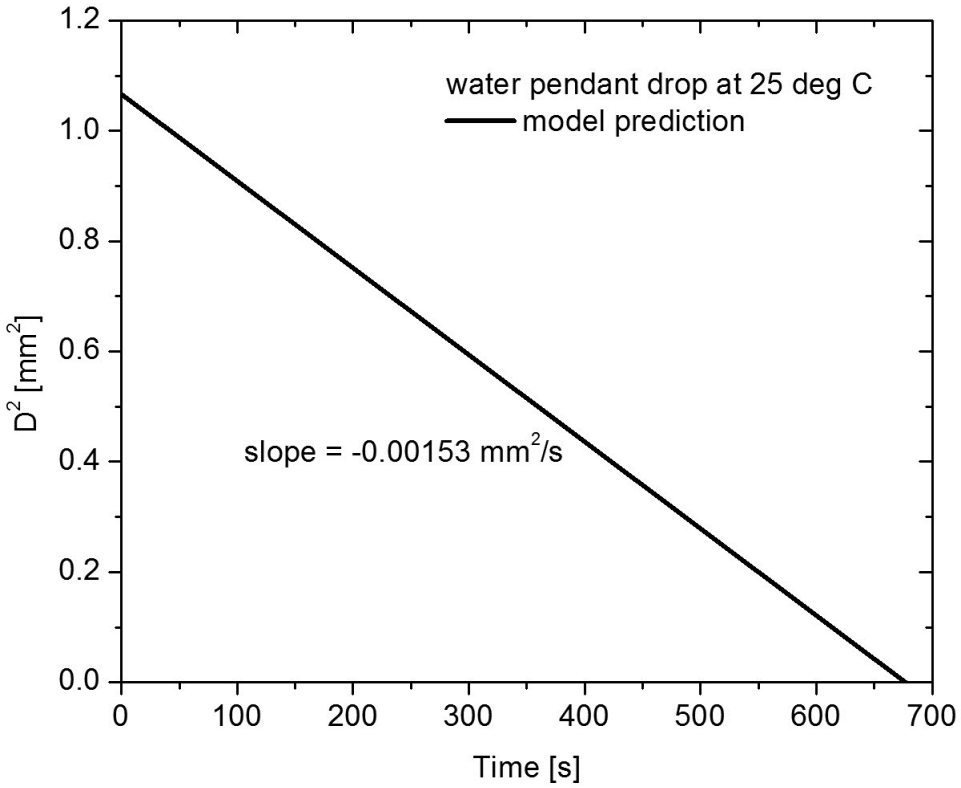

Figure 5.2. Pure water pendant drop evaporation prediction using the current model. The modeled evaporation rate is 0.00153 mm^2/s. The results again match well to measured from Jiang et al. **[61] [62]** and Ranz and Marshal [40] of 0.0015 mm^2/s.

5.3 Liquid Phase Solution Modeling

The mass fraction of nAl was allowed to vary as a function of radius via the diffusion equation:

$$\frac{\partial Y}{\partial t} = \frac{1}{r^2}\frac{\partial}{\partial r}\left(r^2 D \frac{\partial Y}{\partial r}\right), \tag{104}$$

which following discretization using finite differences can be expressed as:

$$Y_r^{t+1} = \frac{\Delta t * D}{r * \Delta r}\left\{(Y_{r+1}^t - Y_{r-1}^t) + \frac{Y_{r+1}^t - 2Y_r^t + Y_{r-1}^t}{\Delta r}\right\} + Y_r^t. \tag{105}$$

Matlab was used to implement a numeric solution to the diffusion equation in concert with the analytic gas phase evaporation analysis. Nanoparticle-modified thermal conductivity was ignored due to the fact that we are assuming of infinitely fast liquid thermal conductivity, and therefore temperature was not modeled using finite differences. However, the mass of the nanoparticles was included in calculating a

modified specific heat for the drop, for use with the bulk temporal temperature change. The effect of the liquid specific heat modification was found to be small.

During each time step the solid surface mass is calculated based on the known liquid mass at the outermost discretization from the prior time step, the solid mass fraction from the prior time step, and mass of evaporated liquid from the D-square law in the current time step. This allows for determination of a new solid volume fraction in the outermost discretization at the drop surface. This new solid volume fraction is then compared to the maximum allowable solid volume fraction based on the fractal character of the agglomerated nanoparticles, as calculated in chapter 3. If the new volume fraction was above the maximum limit, the excess solid fraction is pushed inward within the drop to the next discretization in the finite difference model. Additionally, the solid nanoparticle agglomerates are allowed to diffuse, via the diffusion equation (equation 104), during each time step. The liquid diffusion rate of nanoparticles was estimated using the Einstein-Stokes diffusion equation. The mixture density is given as a function of mass fraction:

$$\rho_{nf}(Y) = \frac{\rho_l}{1 + Y_{np}\frac{\rho_l}{\rho_{np}} - Y_{np}}, \tag{106}$$

and the mass fraction as a function of volume fraction:

$$Y_{max} = \frac{\phi_{max}\rho_{np}}{(\phi_{max}\rho_{np} + (1 - \phi_{max})\rho_l)}, \tag{107}$$

where ϕ_{max} is calculated using agglomeration studies described previously in chapter 3.

The current modeling approach utilizing a linked gas and liquid phase solution with a modification to the ethanol fraction at the drop surface due to the presence of nanoparticle agglomerates allows for prediction of temperature profiles and evaporation rates as described in Figure 5.3, where the increased particle loading at the drop surface reduces the evaporation rate and causes a rise in drop temperature. The evaporation rate is reduced relative to the pure fluid case, even with these increasing temperatures, due to the ever increasing layering of maximum volume fraction nanoparticles at the outermost portion of the drop. This means that the current model can diverge from the classic D-square law of linear evaporation rate at long times as shown in Figure 5.3.

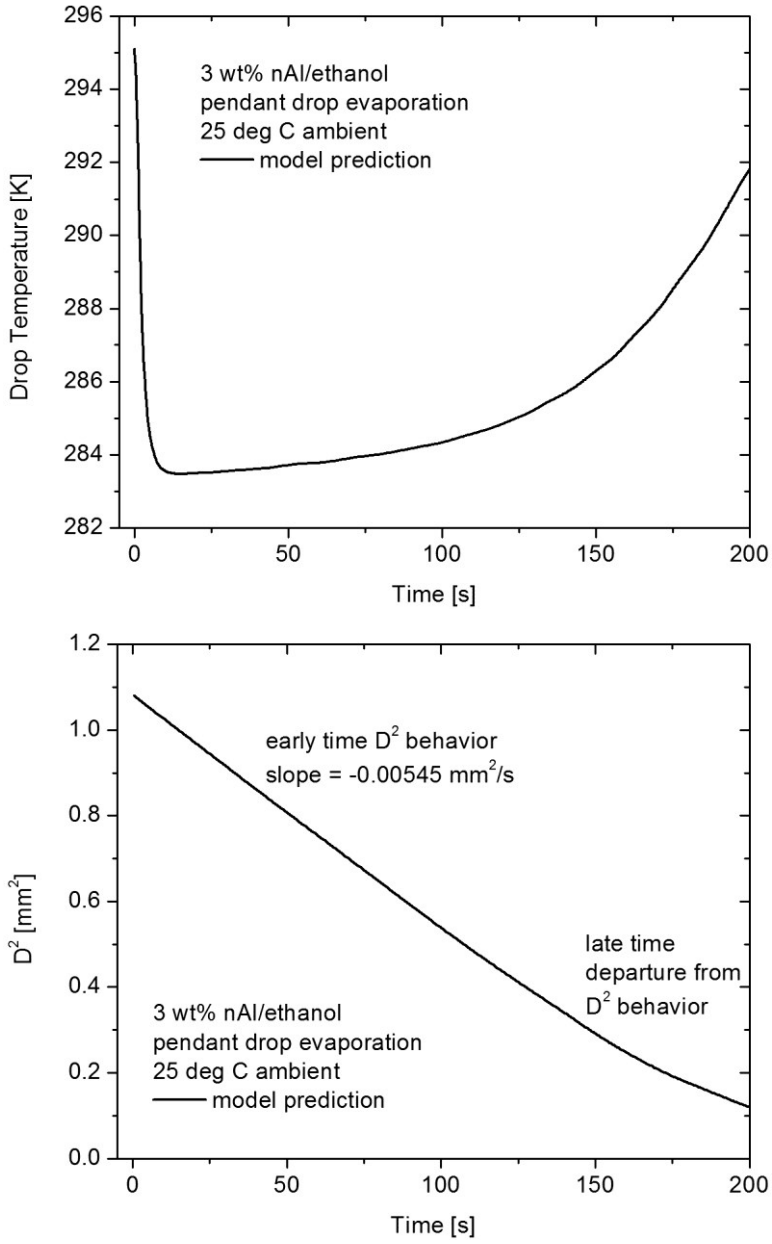

Figure 5.3. Modeling predictions for the evaporation of 3 wt% nAl/ethanol nanofluid. Time varying bulk drop temperature versus time (top) and drop diameter squared versus time (bottom). At long times the drop temperature rises sharply and the evaporation rate drops due to accumulation of nanoparticle agglomerates near the drop surface. This results here are not physically realizable due to slurry/solid shell formation at the drop surface which therefore invalidates the late time solution of the current model. The current study uses the first 40 seconds of this simulation to determine evaporation rate.

5.4 Input Values

Properties used in calculations are from Perry's Chemical Engineering Handbook [63], assuming atmospheric pressures, and a range of temperatures from approximately 0 to 30 degrees Celsius. Parameters which required calculation, such as the diffusion rate of ethanol in air, were calculated using a variety of theoretical relationships. Thermal conductivity of the gas phase was calculated using the Lindsay-Bromley correlation [64], while the diffusion rate of vapor was estimated using Fuller, Schettler, and Giddings correlation (FSG) [65]. Gas phase mixture density was estimated using the perfect gas assumption. Many of these estimation procedures (FSG, Lindsay- Bromley) were used in by Jiang [61], [62] and Homer [60] to great success for multiple evaporation species. The following subsections explicitly state the method of evaluation in the current study.

5.4.1 Liquid Phase Specific Heats

The liquid phase specific heat of ethanol was calculated using Thermal Fluids Central [66] temperature dependent fit:

$$c_{p_{ethanol}}\left[\frac{J}{kgK}\right] = \frac{1000}{MW}\left(98.39 + 0.5368(T - 273.15)\right), \tag{108}$$

where MW is the molecular weight, and T is temperature in Kelvin. Nanoparticle specific heats were evaluated as the bulk specific heat of the material:

$$c_{p_{ethanol}}\left[\frac{J}{kgK}\right] = 910. \tag{109}$$

The mixture specific heat was estimated using the ideal mixture rule:

$$c_{p_{mix}}\left[\frac{J}{kgK}\right] = Y_{nano}c_{p_n} + (1 - Y_{nano})c_{p_l}. \tag{110}$$

5.4.2 Liquid Phase Density

The density of the liquid phase is estimated as:

$$\rho_{liquid}\left[\frac{kg}{m^3}\right] = -0.8544(T - 273.15) + 806.43, \tag{111}$$

in degrees Kelvin from Thermal Fluids Central [66]. This evaluation is independent of nanoparticle loading concentration and is valid due to the small concentrations used in the current experiment.

5.4.3 Liquid Phase Vapor Pressure

The evaluation of the vapor pressure for use in the nanoparticle evaporation models can be estimated through the Clausius-Clayperion relationship, or through a temperature-dependent fit of vapor pressure data. The latent heat of evaporation fit from Thermal Fluids Central [66] is:

$$L\left[\frac{J}{kg}\right] = \frac{1000^2}{MW} 50.43 exp^{\left(\frac{0.4475T}{513.9}\right)} \left(1 - \frac{T}{513.9}\right)^{0.4989}, \tag{112}$$

where MW is the molecular weight, and T is in degrees Kelvin. The experimental fit for the vapor pressure of ethanol, again from Thermal Fluids Central [66], as a function of temperature in Kelvin is:

$$\begin{aligned} P_v\left[\frac{N}{m^2}\right] = 133.3224 & \\ * [4.0325 \times 10^{-4} (T - 273.15)^3 & \\ + 2.7952 \times 10^{-2} (T - 273.15)^2 & \\ + 0.81796(T - 273.15) + 11.574]. & \end{aligned} \tag{113}$$

5.4.4 Liquid Phase Viscosity

The liquid phase viscosity of ethanol was evaluated using Thermal Fluids Central's [66] temperature dependent fit to experimental data:

$$\begin{aligned} \mu_{liquid}\left[\frac{Ns}{m^2}\right] = \frac{1}{1000} \exp(5.8942 \times 10^{-1} & \\ - 2.254 \times 10^{-2}(T - 273.15) & \\ + 1.0283 \times 10^{-4} (T - 273.15)^2 & \\ - 8.857 \times 10^{-7}(T - 273.15)^3 & \\ + 4.7884 \times 10^{-9}(T - 273.15)^4 & \\ - 9.7493 \times 10^{-12}(T - 273.15)^5). & \end{aligned} \tag{114}$$

5.4.5 Pressure

Pressure was assumed to be constant throughout the experiment, and held fixed at an atmospheric pressure of 101325 Pa.

5.4.6 Gas Phase Thermal Conductivity

The gas phase thermal conductivity was evaluated using the Lindsay Bromley correlation [64], as used by Jiang et al. [61] [62]. This correlation first evaluates parameters:

$$A_{ij} = \frac{1}{4}\left(1 + \frac{v_i}{v_j}\left(\frac{MW_j}{MW_i}\right)^{\frac{3}{4}}\left(\frac{T+S_i}{T+S_j}\right)^{\frac{1}{2}}\right)^2 \frac{T+S_{ij}}{T+S_i}, \tag{115}$$

where v, MW, T, and S are respectively, the viscosity, molecular weights, temperature, and Sutherland constant of species i and j. Sutherland's constant is evaluated as:

$$S_i = 1.5 * T_{boil,i}, \tag{116}$$

and

$$S_{ij} = 0.73 * \left(S_i S_j\right)^{0.5}, \tag{117}$$

where $T_{boil,i}$ is the boiling temperature of species i in degrees Kelvin, and we have assumed that the gases are not polar. For air and ethanol, T_{boil} is respectively 77.36 K and 351.8 K. Once values of parameter A_{ij} are evaluated for each combination, (which in this case is a total of four parameters; i.e., ethanol-ethanol, air-air, ethanol-air, and air-ethanol) they can be combined for calculation of the gas phase mixture thermal conductivity:

$$\lambda_{mix}\left[\frac{w}{mK}\right] = \frac{X_i\lambda_i(T)}{X_iA_{ii} + X_jA_{ij}} + \frac{X_j\lambda_j(T)}{X_iA_{ji} + X_jA_{jj}}, \tag{118}$$

where X and λ refer to the mole fraction, and thermal conductivity of species i (in this case air and ethanol). The evaluation of the separate thermal conductivities of these species is performed through correlations from Thermal Fluids Central [66]:

$$\lambda_{ethanol\ vapor}\left[\frac{w}{mK}\right] = 4.1841 \times 10^{-7}T^2 - 1.6423 \times 10^{-4}T + 0.026248. \tag{119}$$

and

$$\lambda_{air}\left[\frac{W}{mK}\right] = 7.071 \times 10^{-5}(T - 273.15) + 2.428 \times 10^{-2}, \tag{120}$$

where temperature is in degrees Kelvin.

5.4.7 Gas Phase Diffusion Rate

The diffusion rate of two gas phase species can be evaluated through the Fuller, Schettler, and Giddings (FSG) correlation [65]. This has also been used by Jiang [62]. Repeated here is the FSG correlation for binary gas diffusion:

$$D_{12}\left[\frac{m^2}{s}\right] = \frac{1}{100^2}\frac{10^{-3}T^{1.75}\left(\frac{1}{MW_1} + \frac{1}{MW_2}\right)^{\frac{1}{2}}}{\frac{P}{101325}\left(V_1^{\frac{1}{3}} + V_2^{\frac{1}{3}}\right)^2}, \tag{121}$$

where T is temperature in Kelvin, MW is the molecular weight of a species, P is pressure in Pascals, and V is the diffusion volume (20.1 for air, and 50.36 for ethanol), and the subscripts 1 and 2 here refer to air and ethanol in the present use.

5.4.8 Gas Phase Specific Heat

The specific heat of air was assumed to be constant:

$$c_{p_{air}}\left[\frac{J}{kgK}\right] = 1005. \tag{122}$$

While the specific heat of ethanol vapor was found through a temperature varying correlation at constant pressure from Thermal Fluids Central [66]:

$$c_{p_{ethanol\ vapor}}\left[\frac{J}{kgK}\right] = \tag{123}$$

$$\frac{1000}{MW_{ethanol}}(-8.28925 \times 10^{-5}T^2 + 0.216104T + 8.28126),$$

where T is the temperature in Kelvin, and MW is the molecular weight of ethanol. The gas phase specific heat of the mixture was estimated using the linear mixture rule:

$$c_{p_{mix}}\left[\frac{J}{kgK}\right] = Y_{ethanol}c_{p_{ethanol\ vapor}} + (1 - Y_{ethanol})c_{p_{air}}. \tag{124}$$

5.4.9 Gas Phase Viscosity

The viscosity of the surrounding air was used in the calculation of the thermal conductivity of the gas phase mixture of ethanol and air. The viscosity of the pure air can be expressed as reported from Thermal Fluids Central [67]:

$$\mu_{air}\left[\frac{Ns}{m^2}\right] = 4.7225 \times 10^{-8}(T - 273.15) + 1.7238 \times 10^{-5}, \quad (125)$$

where T is temperature in degrees Kelvin. Additionally, from Thermal Fluids Central [66], the viscosity of ethanol vapor can be expressed as:

$$\mu_{ethanol\ vapor}\left[\frac{Ns}{m^2}\right] = 1.4991 \times 10^{-7} + 3.0741 \times 10^{-8}T - 4.4479 \times 10^{-12}T^2, \quad (126)$$

where once again T is the temperature expressed in degrees Kelvin.

5.4.10 Gas Phase Density

The gas phase density of air is calculated using a cubic fit of the data reported in Table 1 from Perry's chemical engineering handbook [63].

Table 1. Density of air at 1 atm as a function of temperature [64].

Temperature (Kelvin)	150	200	250	300	350
Density $\left[\frac{kg}{m^3}\right]$	2.3364	1.7458	1.3947	1.1614	0.995

The expression for the gas phase density of ethanol as a function of temperature (in degrees Kelvin) is from Thermal Fluids Central [67]:

$$\rho_{ethanol\ vapor}\left[\frac{kg}{m^3}\right] = \exp(-3.3681 + 5.2492 \times 10^{-2}(T - 273.15) + 5.163 \times 10^{-5}(T - 273.15)^2 - 1.9542 \times 10^{-6}(T - 273.15)^3 + 8.6893 \times 10^{-9}(T - 273.15)^4 - 1.1451 \times 10^{-11}(T - 273.15)^5). \quad (127)$$

5.5 Numeric Sensitivities

Sensitivity analysis was carried out on the initial conditions and some property parameters. A variation of 1% of the mixed nanofluid density results in a negligible evaporation rate change of 0.03%. Simulations show that specific heat would have to change by an order of magnitude to have any significant effect on the present modeling predictions. Pressure and temperature of the surrounding ambient were also tested within reasonable bounds. Pressure was varied from $1x10^6$ Pa to $1.05x10^6$ Pa. Results show that pressure variation changes the evaporation rate by 3% at most. Analysis of the time-step sensitivity showed no change with decreased time step size from the nominal value of 0.1 ms. Increasing the number of spatial nodes by a factor of 2 from the nominal simulation value of 2000 resulted in a 0.6% change in evaporation rate, showing that the finite difference scheme is sufficiently spatially resolved; see Figure 5.4 for the grid study. Initial ambient temperature changes from room temperature result in a change in evaporation rate of 1-2% per degree Celsius. Hence, model-experiment comparisons are relatively insensitive to properties considered and the ambient conditions, known to a fraction of a kPa and °C.

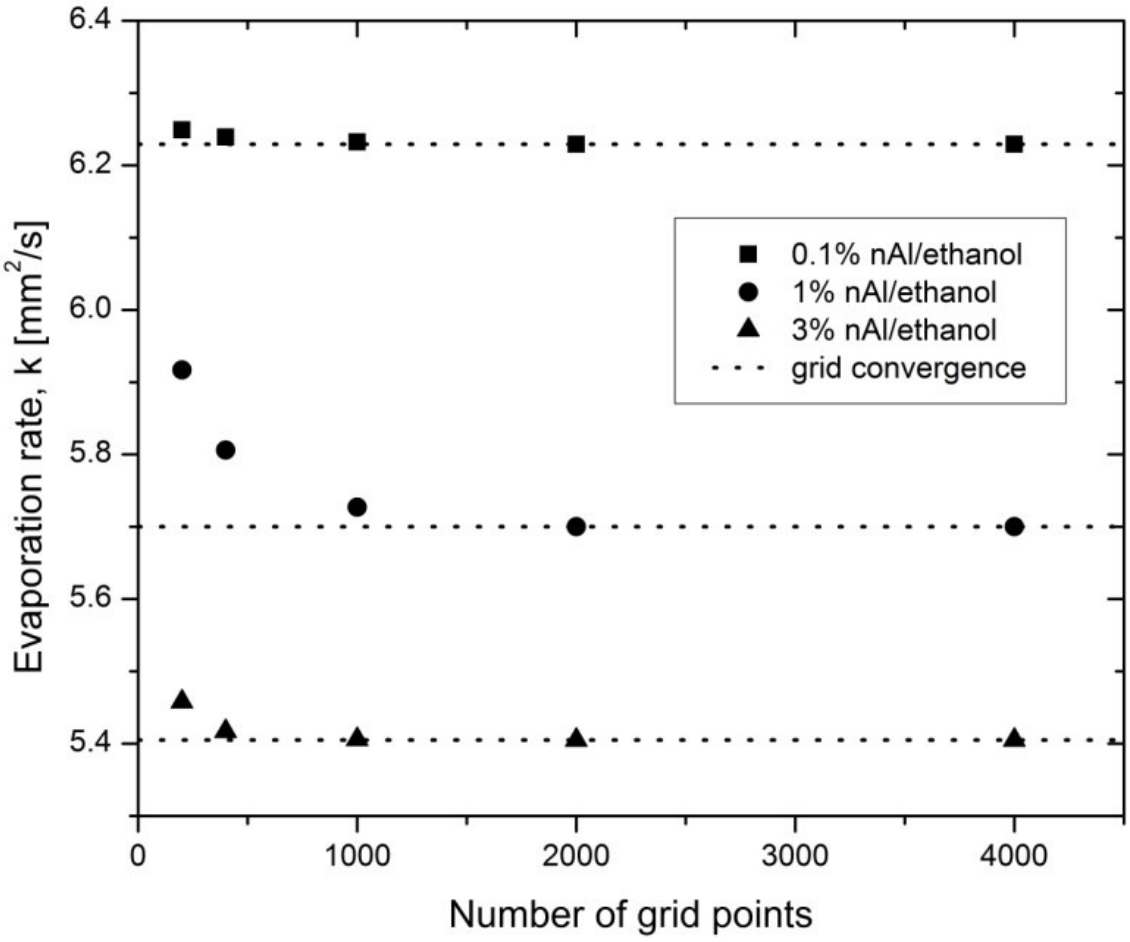

Figure 5.4. Investigation on the solution dependence on liquid drop grid resolution. A grid size of 2000 radial points was chosen for the presented modeling. Repeated with permission from [51].

5.6 Results

The proposed nanofluid pendant drop evaporation model was run at discrete conditions corresponding to 0, 0.05, 0.1, 0.15, 0.25, 0.5, 1, 1.5, 2, 2.5, and 3 weight percent nAl added to ethanol. The results of this study show an very good agreement with the experimental findings of Chapter 4, with an absolute agreement to within 7% of measured evaporation rate as is seen in Figure 5.5.. The current model predicts both the asymptotic reduction in evaporation rate with increasing nAl concentration, the location of the asymptote around 1.5 wt% nAl, and the quantitative evaporation rate at all nAl concentrations.

The current model does somewhat under predict the sharp decrease in evaporation rate at relatively low nAl concentrations, most notably from 0 to 0.25 wt%. It is expected

that this deviation is due to second order effects such as intermolecular forces and agglomeration rate and agglomerate state during evaporation.

Further steps towards improving this model might take the form of extending the range of application to include the formation, and impact of a partially dried nanoparticle shell at the drop exterior. This shell would dictate the use of the Kelvin equation, as well as potentially a permeability velocity analysis and differential pressure calculations to estimate the likelihood of internal vapor bubble formation. These next steps require great model complexity and, therefore, their implementation may necessitate the use of a fully nonlinear finite element method approach as opposed to the simplified approach utilized here.

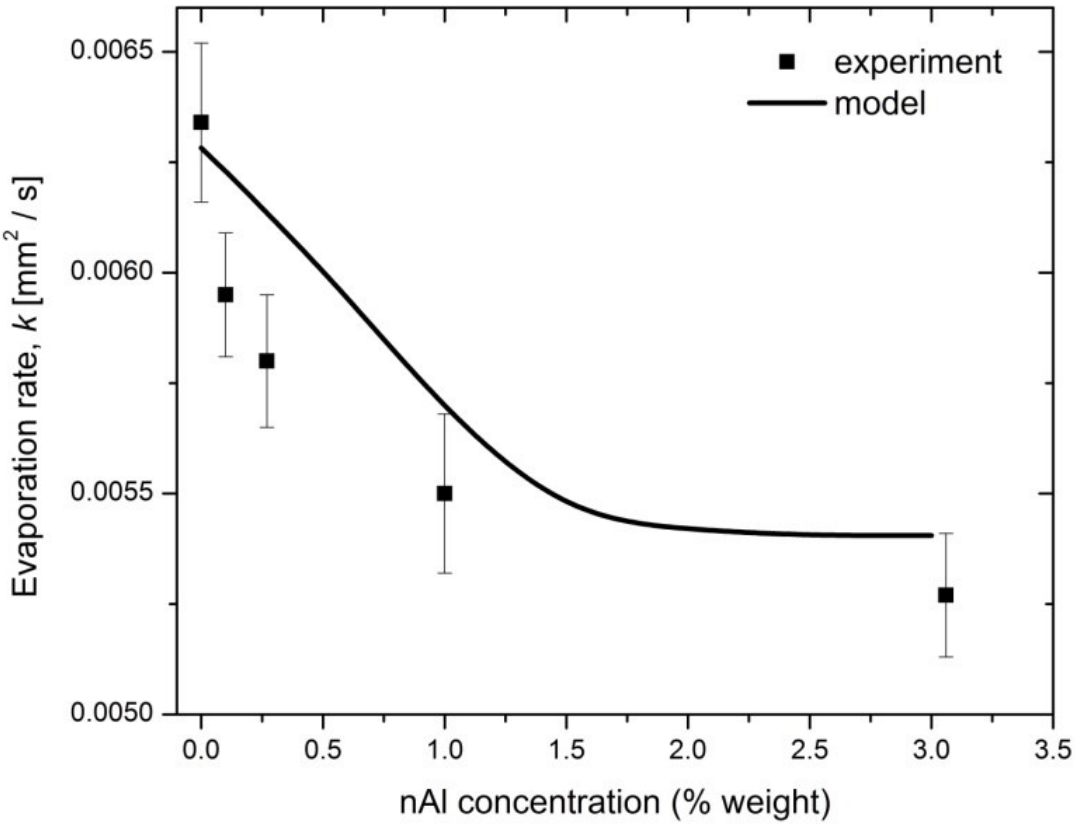

Figure 5.5. Nanofluid pendant drop evaporation, comparison of modeling and experimental results and experimental results for nAl/ethanol nanofluids.. Repeated with permission from [51].

6. Sessile Drop Modeling

Evaporation of a pinned drop on a solid surface, referred to as sessile drop evaporation, differs from pendant drop evaporation in a number of ways and is an important process in applications such as spray cooling and ink-jet printing [68]. Many research efforts have used sessile drops as an experimental indicator of relative surface energies at an instant in time, thus interrogating the interactions between liquid, solid, and gas phases. The interaction between solid, liquid, and gaseous surface energies dictates the angle at which the drop attaches to the solid surface, determining hydrophobic or hydrophilic behavior. Additionally, this contact angle behavior can have notable hysteresis effects, where advancing and retreating contact angles can differ greatly [69], [70]. A pinned sessile drop can be geometrically described by its contact radius, contact angle, and height; Figure 6.1 shows a diagram of a spherical sessile drop with definition of the geometries and Figure 6.2 shows the interfacial forces.

The relative surface energies determine the behavior of a sessile drop as it evaporates. For pure fluids, the relative surface energies are constant during evaporation, although effects due to prior wetting in the form of hysteresis can be observed. The evaporation of sessile drops can be subdivided into two stages. Evaporation in the first stage occurs at a constant drop radius with reducing contact angle and is referred to as the 'pinned' stage [71]. At some point during evaporation a critical contact angle is reached and the force balance at the contact line changes resulting in a 'depinning' of the contact point and a new stage of evaporation. This second stage of evaporation is characterized by a retreating contact radius, and is referred to as the 'unpinned' or 'depinned' stage [70], [72]. The transition from pinned to depinned is not binary, and even for pure fluids, calculating this critical contact angle for depinning during drop evaporation is not well understood [73]. The geometric nature of these evaporation phases can be seen in Figure 6.3.

Colloid sessile drops do not have a constant relative energy at their interfaces. Internal fluid motion can create conditions in which large concentrations of particles deposit near the outer ring of the drop creating sediment [74], [75], [76], [77]. This has the potential to greatly change the surface energy balance at the contact line as a function of deposition rate and drop geometry and substrate surface properties. As the interfacial

energies change, so do the evaporation characteristics. The critical contact angle is now also a function of the particle dimensions (nanoparticle agglomerates here), deposition rate, and sediment cross section [78], [74]. The details of these evaporation mechanics will be discussed in the following sections.

The assumptions used in the derivation of the D-square law for pendant drops are no longer valid for sessile drops; spherically symmetric diffusion driven transport is no longer applicable. Instead, gravity and surface tension balance forces at the drop contact line dictate the binary behavior of sessile drop evaporation. Hence, a new formulation of the governing equations will be required for modeling sessile drop evaporation.

Here modeling predictions are compared to the nanofluid sessile drop evaporation experiment of Sefiane and Bennacer [11]. The present modeling approach for sessile drops uses the mechanism for particle influence on evaporation rate, through the reduction in available vapor pressure at the drop surface due to agglomerated nanoparticles, previously developed for pendant drops. Comparison with the Sefiane and Bennacer [11] results provides additional evidence of the proposed mechanism for nanofluid evaporation rate reduction. The Sefiane and Bennacer [11] experimental work investigated with the pinned and unpinned stages of drop evaporation; however, the current modeling effort focuses solely on the pinned stage of drop evaporation. Depinned drop evaporation and the critical depinning point are dependent on the rate of sedimentation and cross sectional shape of the sediment, seen in Figure 6.4, all of which involve different physics, not focused on in the present study and not well understood based on current literature [73].

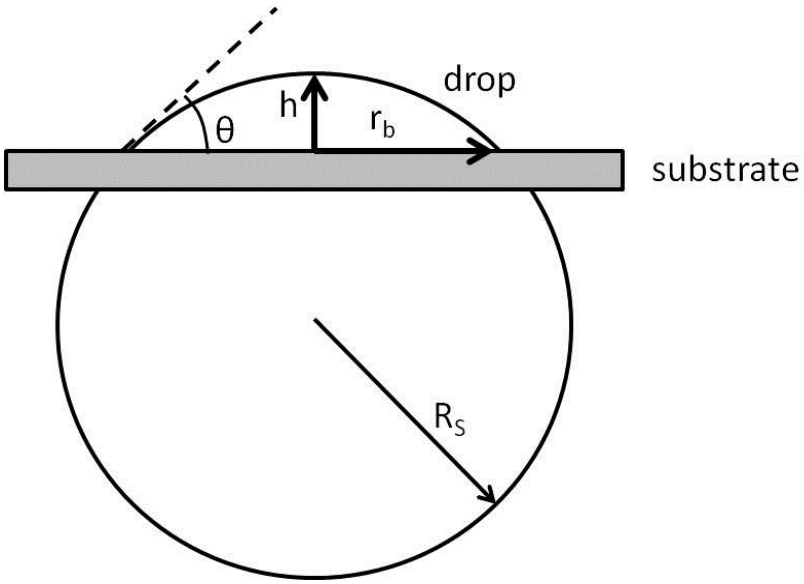

Figure 6.1 Spherical sessile drop geometry.

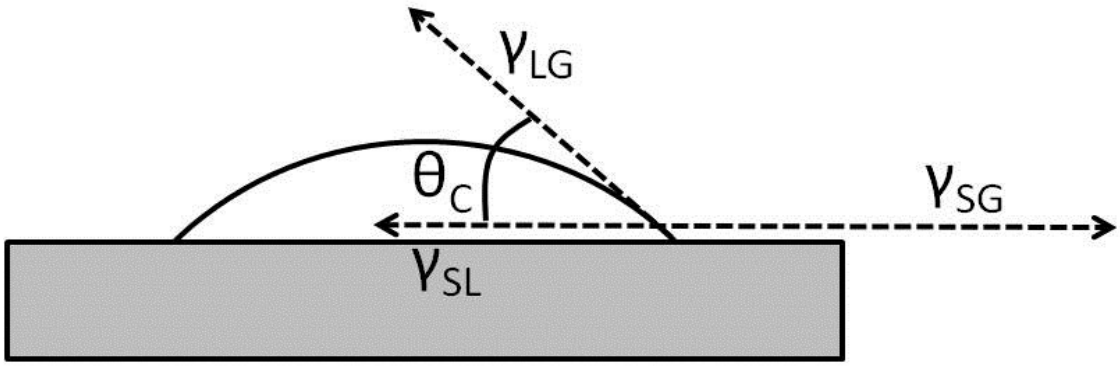

Figure 6.2 Sessile drop interfacial forces. Dotted lines denote force vectors which correspond to the liquid-gas, solid-liquid, and solid-gas surface tension vectors.

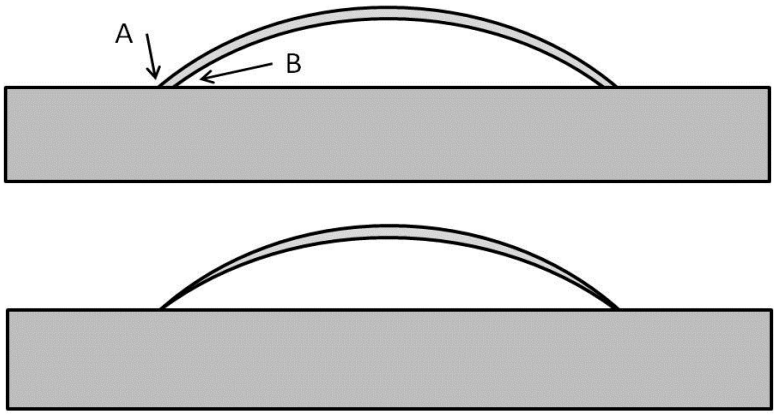

Figure 6.3 Sessile drop evaporation in depinned (top) and pinned (bottom) stages. In order for the radius to remain constant in the pinned evaporation phase, a radially outward flow is required. The depinned drop (top) shows the potential for a constant contact angle with decreasing radius; evaporated volume comes from the reduced contact radius. In the pinned drop (bottom) the evaporated volume comes from a reduced contact angle. Figure adapted from Deegan et al. [79].

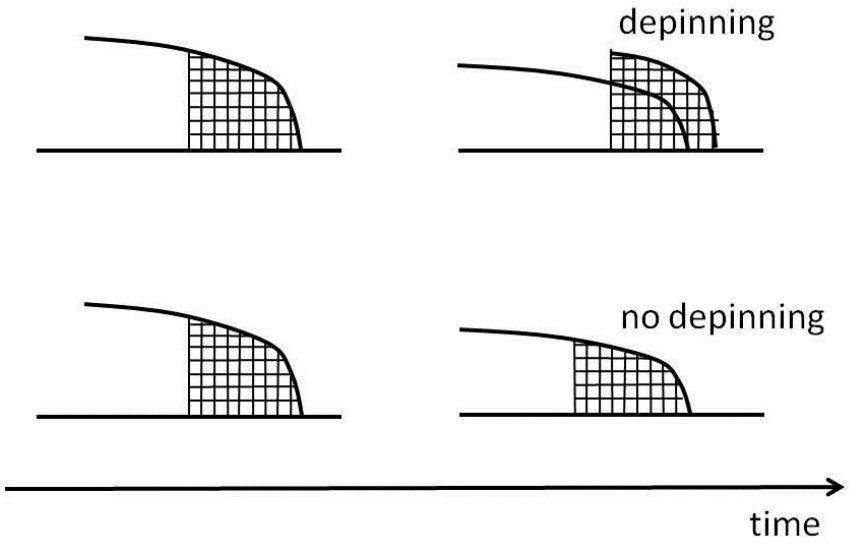

Figure 6.4 Cross section of particle deposition along the outside of the particle ring. The present model does not consider evaporation of depinning drops that typically occurs at sessile drop evaporation longer timescales. Figure adapted from reference [74].

6.1 Sefiane and Bennacer Sessile Drop Evaporation Experiment

Results published by Sefiane and Bennacer [11] in 2008 for ethanol and nAl/ethanol nanofluid (5 wt% nAl) sessile drops on heated PTFE are here used for comparisons to modeling predictions based on the current nanofluid evaporation theory. While other studies [80], [81], [77], [76] have investigated sessile drop evaporation for

various nanofluids, Sefiane and Bennacer performed experiments for nanofluids similar to those investigated here (ethanol and aluminum nanoparticles with no surfactant). This similarity between nanofluids provides a good test of the theory developed here based on pendant drop experimental results. A brief overview of the Sefiane and Bennacer [11] experiment and results is presented here.

Sefiane and Bennacer investigated the evaporation of sessile drops of pure ethanol and a nanofluid containing 5 wt% of aluminum nanoparticles in ethanol on a heated PTFE substrate. The nanoparticle size distribution was characterized using Atomic Force Microscopy and Transmission Electron Microscopy of dried samples, and reported as 60 nm in diameter on average as shown in Figure 6.5. As in the present study, Sefiane and Bennacer also noted that the nanoparticles had a thin oxide layer which may have some additional effect on the nanoparticle stability and/or nanoparticle-fluid interactions. Additionally, nanofluid density, viscosity, and surface tension were reported. Nanofluid density followed the linear mixture rule, as expected, and viscosity increased linearly with increasing nanoparticle volume fraction (Figure 6.6). Surface tension (Figure 6.6) remained constant across the volume fractions investigated (0-2%) which is in agreement with the present experimental results (Chapter 4).

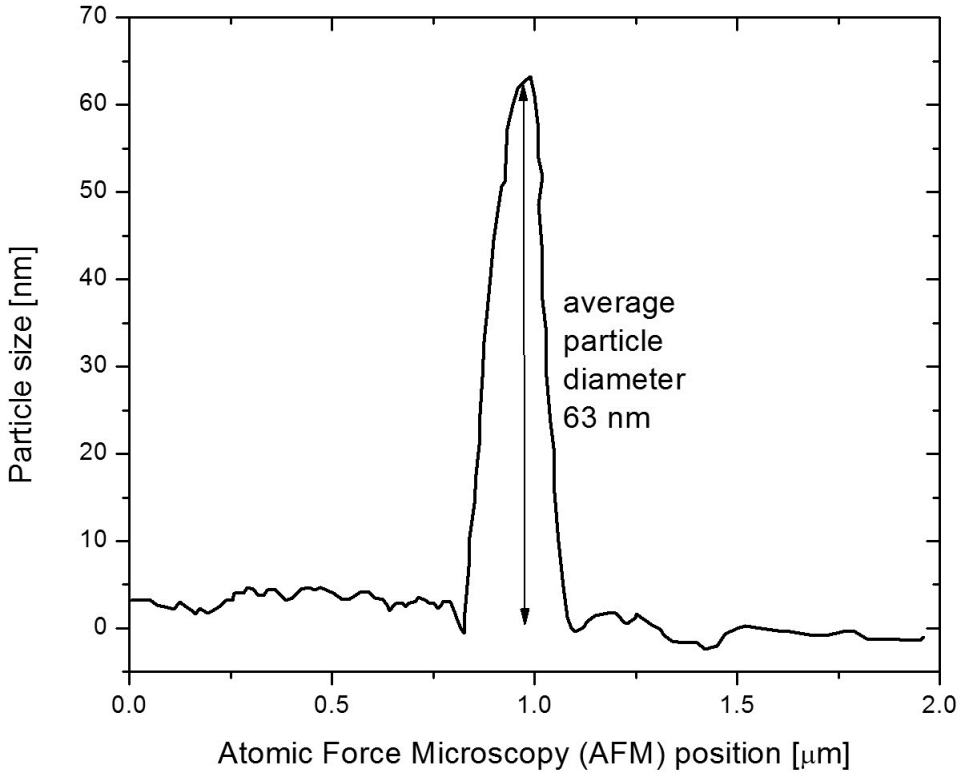

Figure 6.5 Sefiane and Bennacer [11] size distribution of nanoparticles from AFM measurements; average size is around 60 nm.

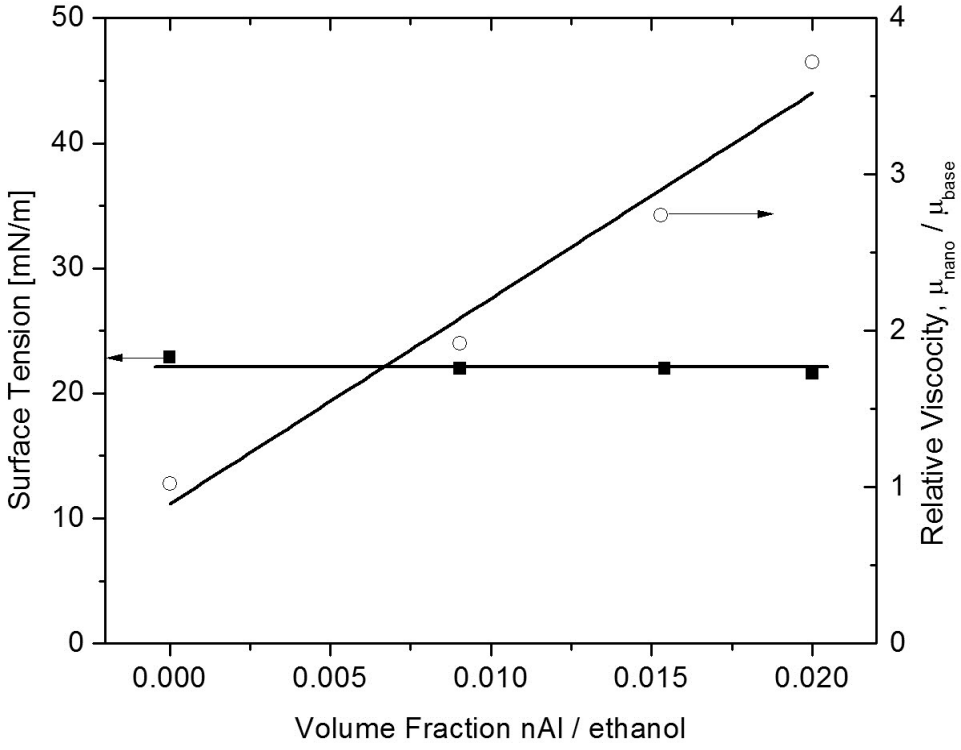

Figure 6.6 Sefiane and Bennacer surface tension and viscosity measurements [11].

The Sefiane and Bennacer [11] experiment utilized a PTFE substrate heated to a range of temperatures (25-95 °C) using a PID controlled imbedded electrical heater. Here only the 25 °C data of Sefiane and Bennacer is compared to the present modeling results, as all the present pendant drop studies were carried out at room temperature. The heater PID control used a thermocouple imbedded inside the PTFE substrate at a depth of 1.5 mm, and as such the reported temperatures are for the thermocouple, and do not report the temperature of the PTFE surface or the liquid drop itself. Sefiane and Bennacer reported substrate temperatures at a depth of 1.5 mm with estimated accuracy of ±0.5 °C.

The drop radius, and contact angle were measured as a function of time using a goniometer until the drop was fully evaporated. This resulted characterization of both the pinned and depinned evaporation stages. Complete experimental results for Sefiane and Bennacer [11] for pure and nanofluid (5 wt% nAl in ethanol) are shown in Figure 6.7. Their results show a decrease in nanofluid evaporation time relative to ethanol. It was noted that the nanofluid drops remained pinned for a longer period of time, causing this decrease in total evaporation time due to the greater evaporation rate for a pinned drop relative to a depinned drop. This is due to the greater contact area of the pinned drops. However, the drop evaporation rate in either the pinned or depinned states was reduced for nAl/ethanol nanofluid relative to pure ethanol. This decrease in evaporation rate is in agreement with the current findings for pendant drops. The evaporation rates for ethanol and 5 wt% nAl/ethanol nanofluids for a substrate temperature of 65 °C is shown in Figure 6.7. It was hypothesized by Sefiane and Bennacer [11] that the internal velocity driven nanoparticle deposition of at the drop edge was the primary factor in delaying the transition to depinned evaporation. This has been numerically studied by others in more detail, including Chan et al. [10] and Bhardwaj et al. [74].

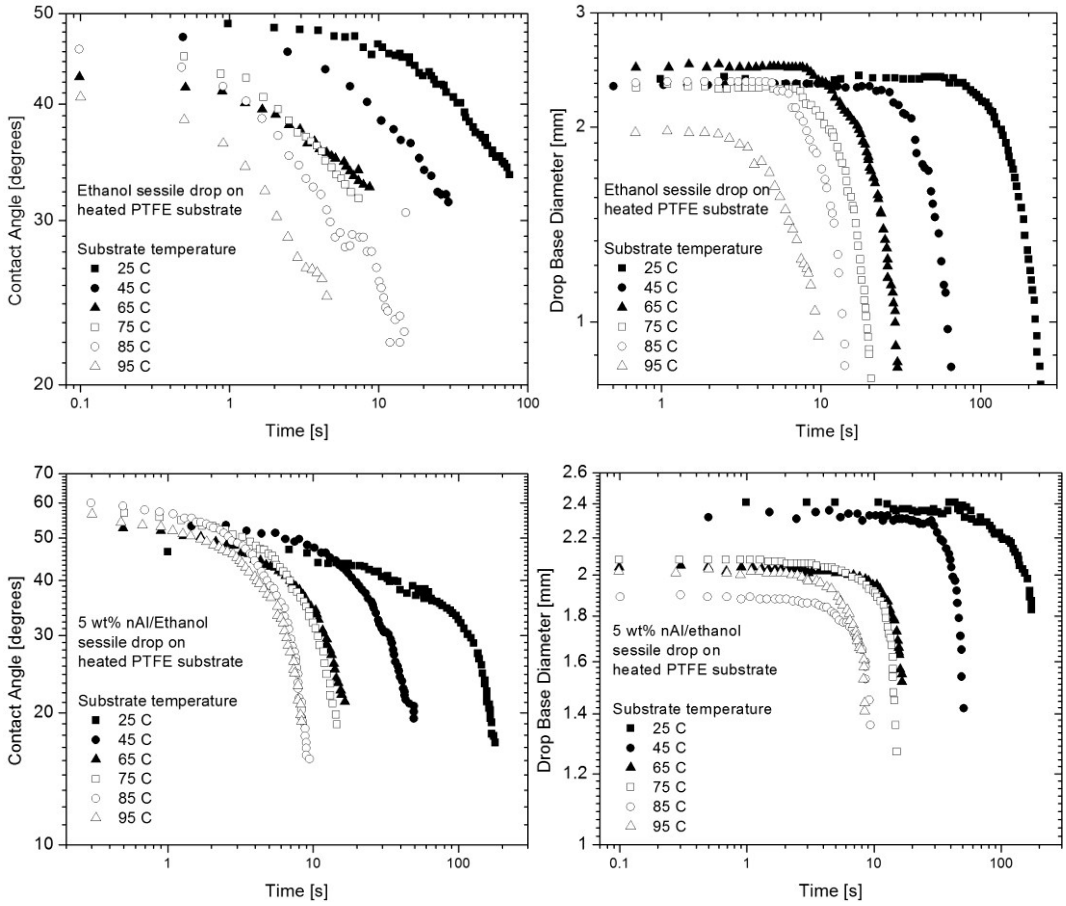

Figure 6.7 Time varying drop contact angle and base diameter from Sefiane and Bennacer [11].

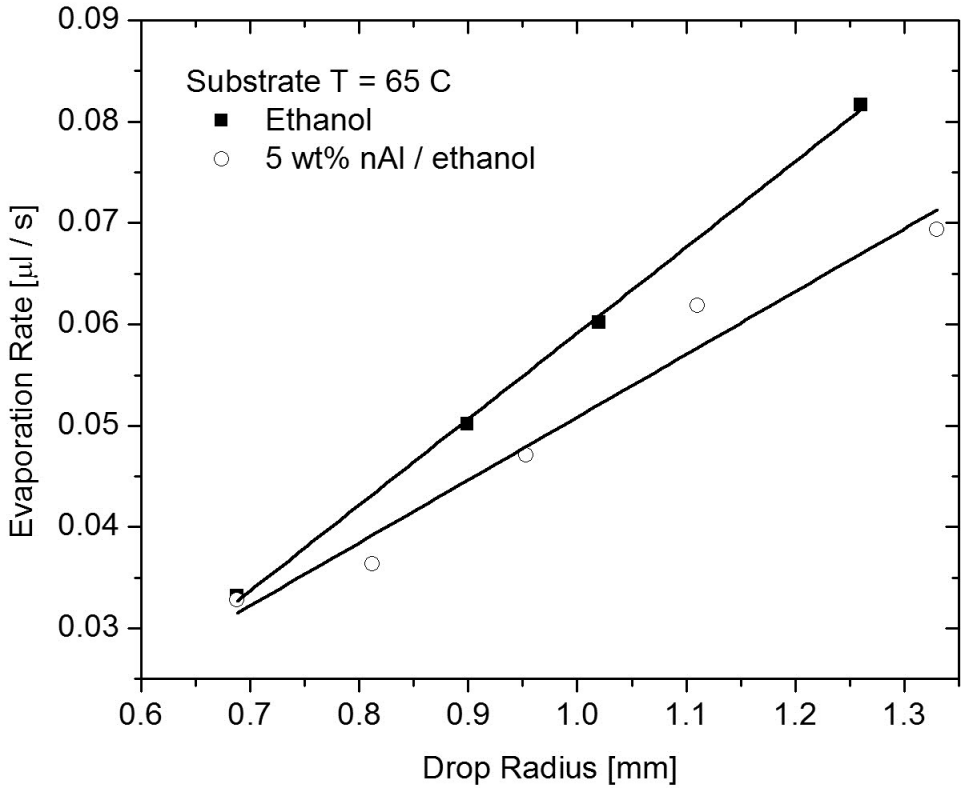

Figure 6.8 Evaporation rate for ethanol and a nAl/ethanol nanofluid as a function of drop radius from Sefiane and Bennacer [11]. This shows the reduction in the evaporation rate for a nAl/ethanol nanofluid relative to pure ethanol.

6.2 Sessile Drop Evaporation Governing Equations

Starting with the cylindrical Navier-Stokes equations and simplifying for steady, two-dimensional, incompressible, and low Reynolds number assumptions, the governing liquid phase equations for the sessile drop evaporation can be written as [83]:

continuity:

$$\frac{1}{r}\frac{\partial(ru_r)}{\partial r}+\frac{\partial u_z}{\partial z}=0, \tag{128}$$

r momentum:

$$\mu\left(\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial}{\partial r}(ru_r)\right)+\frac{\partial^2 u_r}{\partial z^2}\right)=\frac{\partial P}{\partial r}, \tag{129}$$

and for *z* momentum:

$$\mu\left(\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u_z}{\partial r}\right)+\frac{\partial^2 u_z}{\partial z^2}\right)=\frac{\partial P}{\partial z}, \tag{130}$$

and finally for the temperature equation (here we relax the steady assumption):

$$\rho c_p \left(\frac{\partial T}{\partial t} + \vec{u} \cdot \nabla T \right) + \lambda \nabla^2 T = 0, \tag{131}$$

where $\rho, c_{p,}$and λ are the density, specific heat, and thermal conductivity, respectively. Through the non-dimensional analysis presented by Hu et al. [83], the inverse Stanton number,

$$St^{-1} = \frac{\rho c_p u_r R}{\lambda}, \tag{132}$$

allows for an estimation of the ratio of convective heat transfer to conductive heat transfer. When this ratio is favorable (conduction is much faster than convective heat transfer) we can eliminate the time varying and spatially varying convective components of the temperature equation, once again as Hu et al. [83] proposed:

$$\nabla^2 T = 0. \tag{133}$$

The gas phase concentration can be described mathematically through the Laplace equation as noted by Hu et al. [83], Dunn et al. [71], and first proposed by Deegan et al. [79]:

$$\nabla^2 C = 0. \tag{134}$$

This set of equations can now be solved numerically [84], or further simplified with an assumed evaporation flux and drop profile [71], [79].

A full solution to these governing equations through numerical methods requires either a finite element or finite difference scheme. Boundary conditions for a numeric solution dictate the interfaces between liquid, gas, and solid. At the liquid-solid interface a continuous heat flux will be applied. At the liquid-gas interface, a liquid phase heating will be related to an evaporation volume as noted by Bhardwaj et al. [74], Widijaja et al. [76], Saada et al. [85], and Hu et al. [83], [84]:

$$h_{fg} J_s \cdot n = -\lambda \nabla T, \tag{135}$$

where h_{fg} is the latent heat of vaporization, J_s the evaporation flux, n the normal vector, λ the thermal conductivity. The continuity and momentum equations are used to ensure continuous shear stress.

6.3 Simplified Approach

Like the simplification of the governing equations for pendant drop evaporation which leads to the D-square law the solution, the procedure used here will be the result

of segmented analysis. Two separate solutions, one for the drop evaporation rate, and one expression for the drop volume are used to estimate the drop time history. The drop evaporation rate is itself a function of thermal analysis, and in the case of added nanoparticles a radial velocity convecting a volume fraction outward.

6.3.1 Vapor Phase Solution

A solution to the gas phase Laplace equation, equation 134, with similar boundary conditions to the sessile drops considered here has been given by Deegan et al. [79]:

$$C(\alpha,\beta) = \tag{136}$$
$$C_\infty + (C_s - C_\infty)\sqrt{2(\cosh\alpha - \cos\beta)}$$
$$* \int_0^\infty P_{-\frac{1}{2}+\iota\tau}(cosh\alpha)\frac{\cosh(\theta_c\tau)\cosh(2\pi-\beta)\tau}{\cosh(\pi\tau)\cosh(\pi-\theta_c)\,\tau}d\tau,$$

where $P_{-\frac{1}{2}+\iota\tau}$ is the Legendre function of the first kind and α and β are the toroidal coordinates. Evaporation at the surface of the drop can be estimated as:

$$J_s = -D\nabla C, \tag{137}$$

which expands to:

$$J_s = D(\cosh\alpha + \cos\beta)\partial C|_{surface}, \tag{138}$$

and finally [79]:

$$J_s = \tag{139}$$
$$-D(C_s - C_\infty)\left(\frac{\sin\theta_c}{2} + \sqrt{2}(x+\cos\theta_c)^{\frac{3}{2}}\right.$$
$$\left.* \int_0^\infty P_{-\frac{1}{2}+\iota\tau}(x)\tau\frac{\cosh(\theta_c\tau)\tanh(\pi-\theta_c)\tau}{\cosh\pi\tau}\,d\tau\right).$$

This relationship is then used to generate a simple fit to the evaporation flux as recommended by Hu et al. [84] based upon comparison with FEM results:

$$J_s = J_0(\theta)\left(1-\left(\frac{r}{R}\right)^2\right)^{-\left(\frac{1}{2}-\frac{\theta}{\pi}\right)}, \tag{140}$$

where:

$$J_0(\theta) = \frac{Dc_v(1-Y_\infty)}{R}(0.27\theta^2 + 1.3) * \left(0.6381 - 0.2239\left(\theta - \frac{\pi}{4}\right)^2\right). \quad (141)$$

Here D, c_v, R, Y_∞ , and θ are the diffusion rate coefficient, the vapor concentration at the surface of the drop, the radius of the drop, mass fraction at infinity, and contact angle, respectively. This correlation has been verified through comparison with finite element method solutions of the full governing equations by Hu et al. [84], [86].

6.3.2 Volume Change Evaluation

This mass flux is then related to a change in total drop volume:

$$\rho\frac{dV}{dt} = \int J_s dS, \quad (142)$$

and assuming a drop geometry allows for further simplification. If a spherical geometry is assumed, drop volume is as reported from Erbil et al. [68]:

$$V = \left(\frac{R}{\sin\theta}\right)^3 \pi\frac{(1-\cos\theta)^2(2+\cos\theta)}{3}, \quad (143)$$

and drop height is:

$$h(r) = R_s\cos\left(\operatorname{asin}\frac{r}{R_s}\right) - R_s. \quad (144)$$

Finally, the relation between evaporation flux and rate of volume change in time is [71]:

$$-\frac{dV}{dt} = \frac{2\pi}{\rho}\int_0^R J_s(r) r dr. \quad (145)$$

6.3.3 Heat Transfer Evaluation

Unlike the unsteady pendant drop evaporation described previously, the heat balance on the drop surface of this sessile drop involves on liquid heating through the substrate for supplying the energy that leads to phase change; hence the vapor concentration at the drop surface is dependent on the thermal analysis. A schematic of the problem can be seen in Figure 6.9. When compared to the ambient convective and conductive heating at the drop surface, the solid and liquid phase heat conduction are at least an order of magnitude greater, and therefore gaseous heating can be ignored for sessile drops, as shown by Dunn et al. [71]. Furthermore, analysis of the Stanton

number concludes that transient heating is minimal, and the sessile drop is at thermal quasi steady state for all but the shortest time periods. Applying an energy balance at the drop gas-liquid interface results in:

$$h_{fg}J_s = -\lambda \frac{\partial T_l}{\partial z}, \tag{146}$$

where h_{fg} is the latent heat of vaporization, J_s the evaporation flux, and λ the liquid thermal conductivity. Here we have assumed a 1-D thermal transport in the z direction:

$$\frac{\partial T}{\partial z} \gg \frac{\partial T}{\partial r}, \tag{147}$$

instead of calculating the component tangent to the liquid surface. Additionally, as proposed by Dunn et al. [71], because the drop height and substrate thickness are small when compared to the drop diameter the second derivatives of both the liquid and substrate temperature profiles can be neglected. This means that all energy coming into the drop from the surface substrate is vertically funneled to the heat required for change of phase. The solution to this temperature profile, as reported by Dunn et al. [71], for the sessile drop is:

$$T_l(z) = T_\infty - h_{fg}J_s\left(\frac{z}{\lambda_l} + \frac{t_{ss}}{\lambda_s}\right), \tag{148}$$

and for the substrate the temperature profile is:

$$T_s(z) = T_\infty - \frac{h_{fg}J_s}{\lambda_s}(z + t_{ss}), \tag{149}$$

where T_l is the liquid drop temperature, z the height coordinate internal to the drop, T_∞ is the ambient temperature, h_{fg} is the latent heat of vaporization, J_s the evaporation flux, λ the thermal conductivity of the liquid and solid as the subscript denotes, and t_{ss} is the thickness of the solid substrate. With a known varying temperature at the drop surface, it is possible to evaluate the vapor pressure through the same relations used in the pendant drop model. For known vapor pressure, the vapor concentration and evaporation mass flux J_s can be solved. The influence of the substrate on sessile drop evaporation rate is exhibited through the series addition of the thermal conductivities. An example of the solution using the temperature profiles described in equations 148 and 149 can be seen in Figure 6.10.

6.3.4 Liquid Velocity

Evaluation of the liquid velocity internal to the drop was evaluated as described by Hu et al.[85], [83] and Deegan et al. [79] by rearranging the conservation of volume equation for a radially discretized cylindrically symmetric sessile drop:

$$\rho\frac{\partial h}{\partial t} = -\rho\frac{1}{r}\frac{\partial}{\partial r}(rhu) - J_s(r,t)\sqrt{1+\left(\frac{\partial h}{\partial r}\right)^2}, \tag{150}$$

and solving for the radially averaged velocity u at a radial position r and time t:

$$u(r,t) = -\frac{1}{\rho r h}\int_0^r r\left(J_s(r,t)\sqrt{1+\left(\frac{\partial h}{\partial r}\right)^2} + \rho\frac{\partial h}{\partial t}\right)dr, \tag{151}$$

where ρ, h, and J_s are the liquid density, height of the drop at radius r, and evaporative flux at radial position r, respectively. This description of velocity can be evaluated numerically using equation 151. Alternatively, the velocity can be non-dimensionalized and expressed as [86], [83]:

$$u(r) = \frac{R}{t_f} * \frac{1}{4}\frac{1}{1-\frac{t}{t_f}}\frac{R}{r}\left[\left(1-\left(\frac{r}{R}\right)^2\right)^{-\left(\frac{1}{2}-\frac{\theta}{\pi}\right)} - \left(1-\left(\frac{r}{R}\right)^2\right)\right], \tag{152}$$

where R is the drop radius, t_f and t the estimated drop lifetime and current simulation time, r the radial location, and lastly θ is the drop contact angle in radians. In order to facilitate comparisons to the experimental results of Sefiane and Bennacer [11], their experimental drop lifetime was used for non-dimensionalization.

6.3.5 Particle Convection

For modeling the concentration of the nanoparticles within the sessile drop Deegan et al. [79] proposed the following conservation equation:

$$\frac{\partial}{\partial t}(ch) + \frac{1}{r}\frac{\partial}{\partial r}(rchu) = 0, \tag{153}$$

where $r, c, h,$ and u are the radial coordinate, the concentration, height, and radial velocity at the coordinate, respectively. Similarly, an algebraic relation formed using a discretized control volume analysis results:

$$c_i = \frac{1}{2\pi r_i h_i dr}[2\pi dt(c_{i-1}u_{i-1}r_{i-1}h_{i-1} - c_i u_i r_i h_i) + c_i dr 2\pi r_i h_i], \tag{154}$$

where $r, c, h,$ and u are as described above, dt and dr are the radial and time discretization, and subscripts i and $i-1$ are the current and previous radial discretized indices. As the solid particle concentration at a given radial location within the sessile drop increases it is checked against the maximum allowable solid volume fraction calculated through identical means as described for the pendant drop evaporation model in Chapter 5. Inputs to this maximum volume fraction model include nanoparticle agglomerate radius and fractal dimension. Because Sefine and Bennacer [11] did not characterize the fractal dimension of their nanofluid, here, for modeling comparison to their experiments, a range of fractal dimensions will be considered and therefore a range of maximum allowable solid volume fraction. Modeling calculations are performed to determine the evaporation rate of the pinned sessile drop as a function of maximum allowable volume fraction, from which an estimate of fractal dimension can be made for a best fit of the Sefine and Bennacer [11] experiment. If the estimated fractal dimension required to match experimental results is realistic (1.7-2.3, as reported by Keblinski et al. [15] [16]) then the current theory has not been disproven.

Each evaluation of nanoparticle concentration is used for estimating the nanofluid mass fraction modified vapor pressure at the liquid temperature. In this way, all radially varying components correctly contribute to the overall drop evaporation rate.

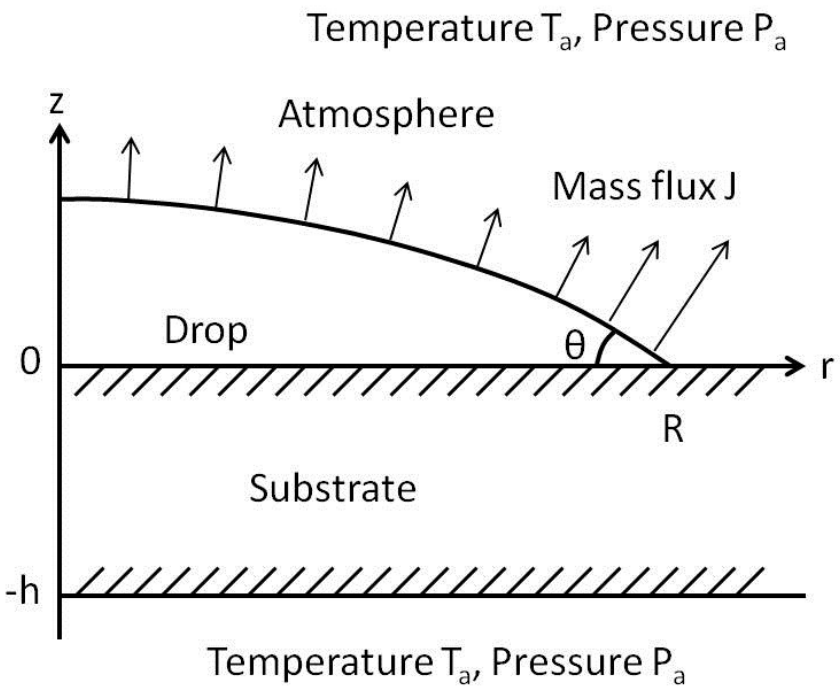

Figure 6.9. Schematic of drop on substrate, adapted from Dunn et al [71].

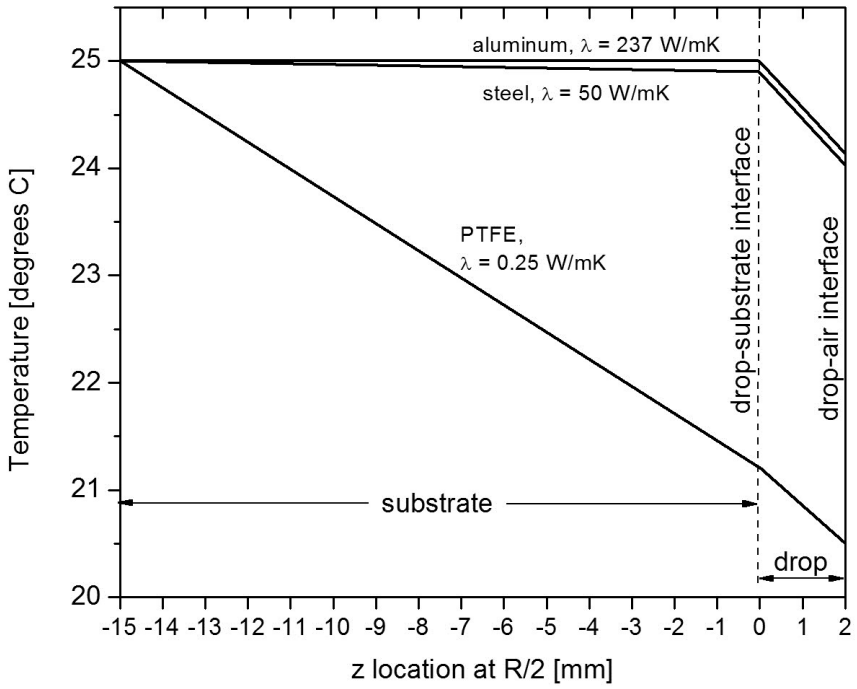

Figure 6.10 Z-direction (vertical in Figure 6.9) temperature profile for substrate and drop at a location of half the drop radius (R/2). Simulations carried out for three different substrate thermal conductivities (PTFE, steel, and aluminum.

6.4 Modeling

Simulation results are reported for the time varying velocity, temperature, evaporation flux, and nanoparticle concentration profiles for an evaporating pinned sessile drop. From these results, it is possible to gain insight into the proposed method to describe the of evaporation rate reduction for nanofluids and how the important variables evolve in time.

The problem is initialized with an initial drop radius, contact angle, and radially uniform concentration of nanoparticle agglomerates. From this initial state, all parameters of interest can be solved as the sessile drop evaporates. The solution routine assumes a spatially uniform drop temperature field (Figure 6.11) equal to the ambient temperature. At this temperature the drop surface vapor concentration is calculated using either the Clausius-Clayperion relation or determined from experimental vapor pressure curve fit [66]. This vapor concentration is then used in the calculation of the mass flux, J_s, as seen in Figure 6.12. From the mass flux, we calculate the change of volume of the drop which is then used to solve for a new contact angle and drop height profile, assuming a fixed radius of contact with the substrate (Figure 6.13). Next, the change in drop height is used to calculate a new increased concentration of nanoparticle agglomerates. The convective transport of this new solid concentration is then solved using equation 154, based upon the calculated radial velocity, seen in Figure 6.14. The new nanoparticle agglomerate concentration distribution is next tested to see if it exceeds the maximum allowable volume fraction dictated by the nanoparticle agglomerate fractal properties. If it does, excess concentration is transported towards the drop interior one mesh step at a time in an iterative method. A drop temperature profile is then calculated using equations 148 and 149 which is then used in the next simulation time step. This process is repeated for each time step until the simulation has reached the end time, a negative drop volume is reached, or a fully packed sessile drop of nanoparticles is found. Drop volume is plotted versus time and best fit for comparison to the experimental results of Sefiane and Bennacer [11], an example of which is seen in Figure 6.19.

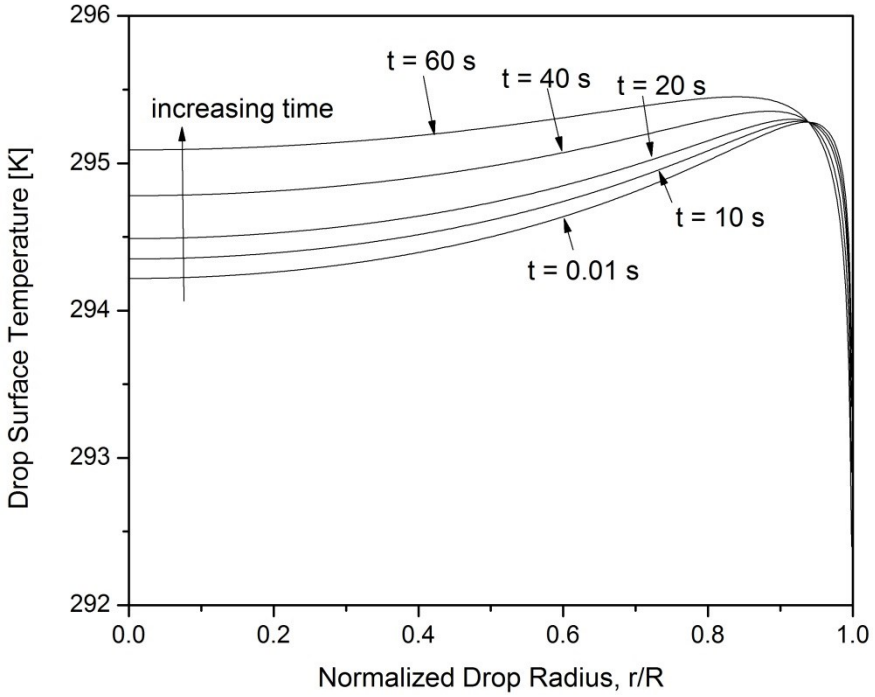

Figure 6.11 Pure ethanol drop modeling results for drop surface temperature as a function of time and drop radius. Drop surface temperature is used in equations 139 and 140 for calculation of the evaporation flux.

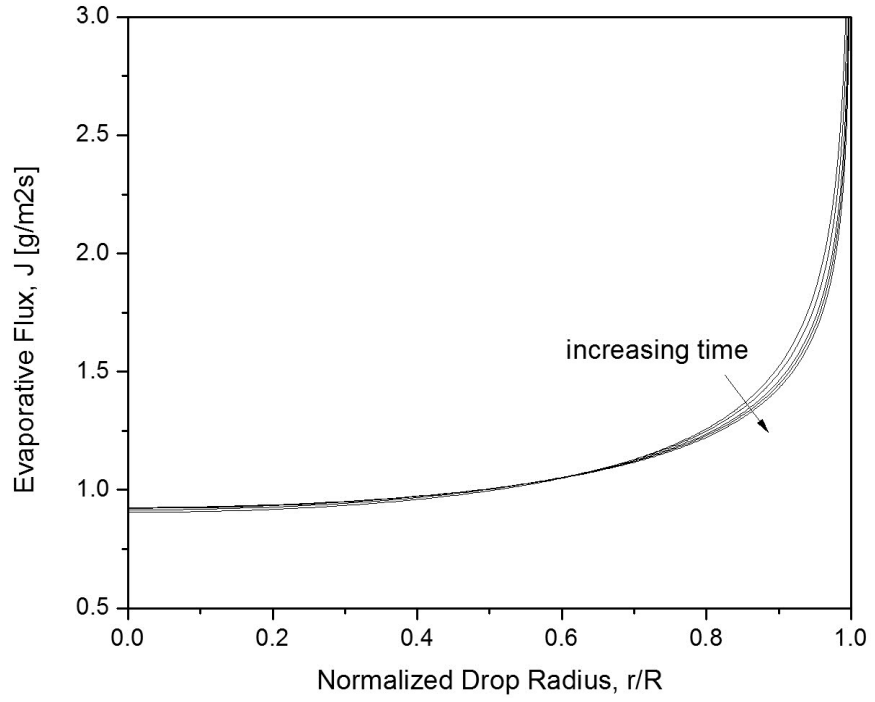

Figure 6.12 Pure ethanol drop modeling results for the evaporation flux at the drop surface as a function of normalized drop radius; this graph reflects the solution to equations 139 and 140.

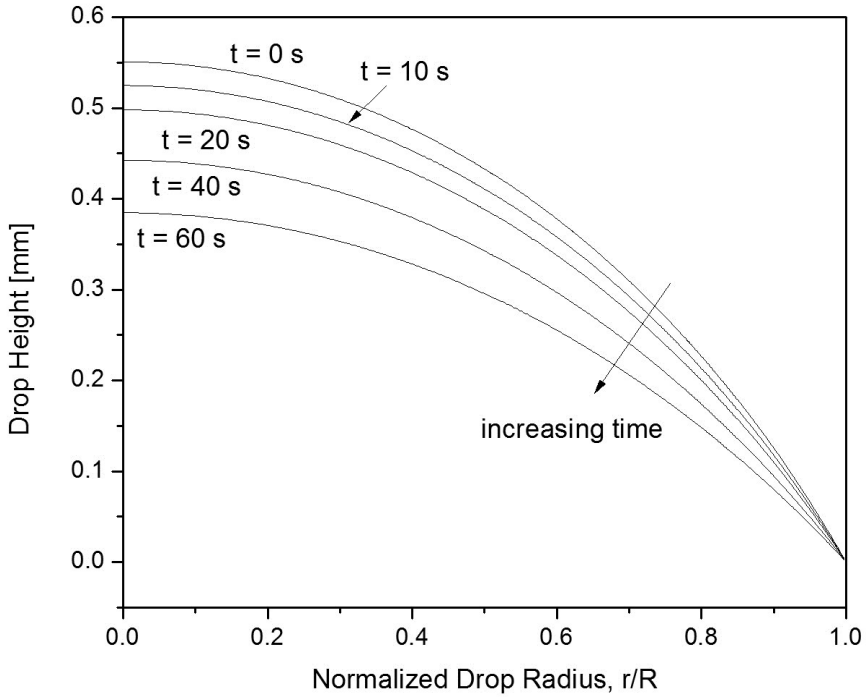

Figure 6.13 Pure ethanol drop modeling results for surface profile as a function of time.

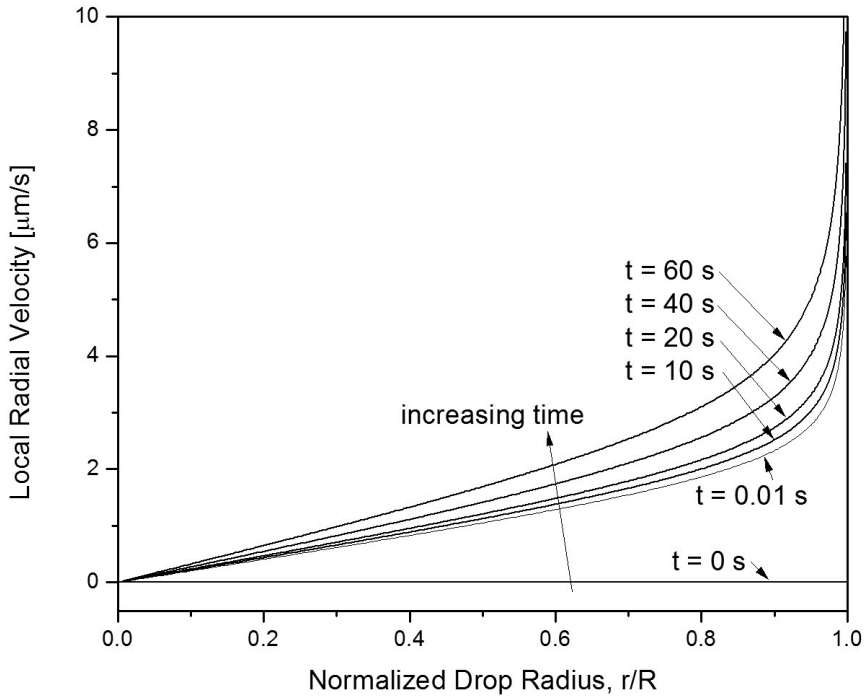

Figure 6.14 Pure ethanol drop modeling results for local radial velocity as a function of time and normalized drop radius.

Initial loadings of nanoparticles influence the radial profiles of temperature, Figure 6.15, evaporation flux, Figure 6.16, and velocity, Figure 6.17, due to the decreased availability of ethanol at the surface for evaporation. The radial and time varying volume fraction profile is strongly dependent on the maximum volume fraction allowed. When this value is reached at the drop exterior, the next inward discretization now becomes the recipient of the excess volume fraction from the exterior shell. This process will continue inward until the simulation is stopped, or a maximum volume fraction is reached throughout the entire drop. The effect can be seen to dominate the natural convective towards the radius in the sharp discontinuity in concentration in Figure 6.18.

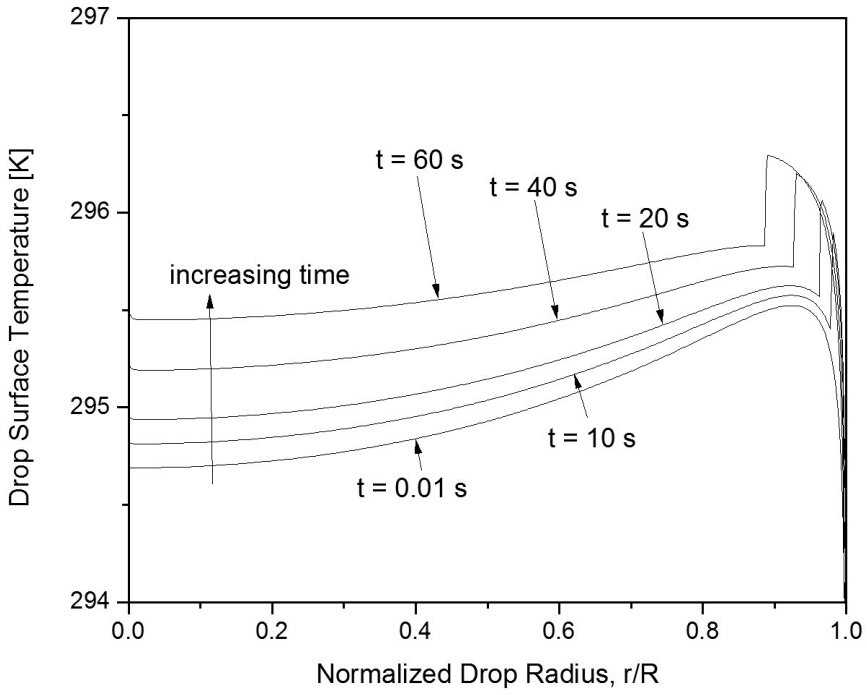

Figure 6.15 Nanofluid (5 wt%/ethanol) drop modeling results for drop surface temperature as a function of time and drop radius. The inclusion of nanoparticles cause higher drop surface temperatures compared to pure ethanol (Figure 6.14) and cause a discontinuity in temperature near the outer radius when the nanoparticle agglomerates reach a maximum allowable volume fraction.

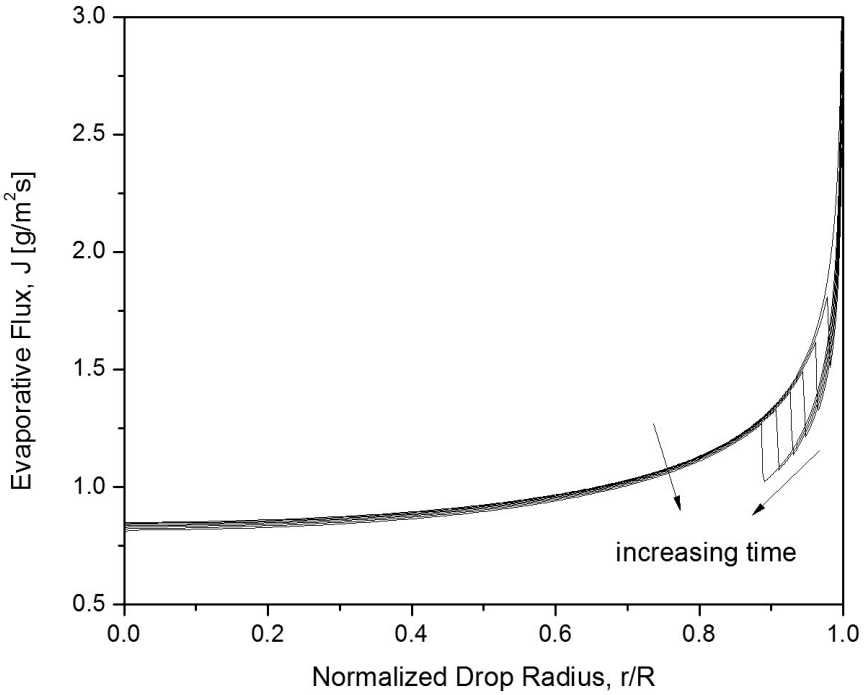

Figure 6.16 Nanofluid drop modeling results for the evaporation flux at the drop surface as a function of normalized drop radius. A reduction in evaporation (discontinuity in curves) occurs near the outer radius when the nanoparticle agglomerates reach a maximum allowable volume fraction.

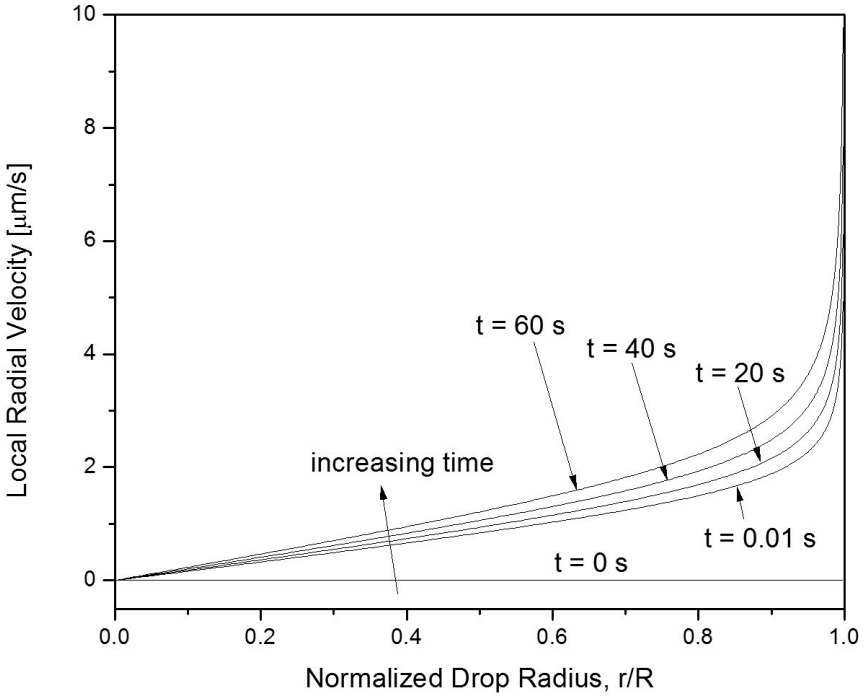

Figure 6.17 Modeled internal drop radial velocity profile for a 5 wt% nAl/ethanol nanofluid drop. It is noticeable here that the inclusion of nanoparticles does not play a significant role in the behavior of the velocity profile, except for the relative magnitudes near the drop radius.

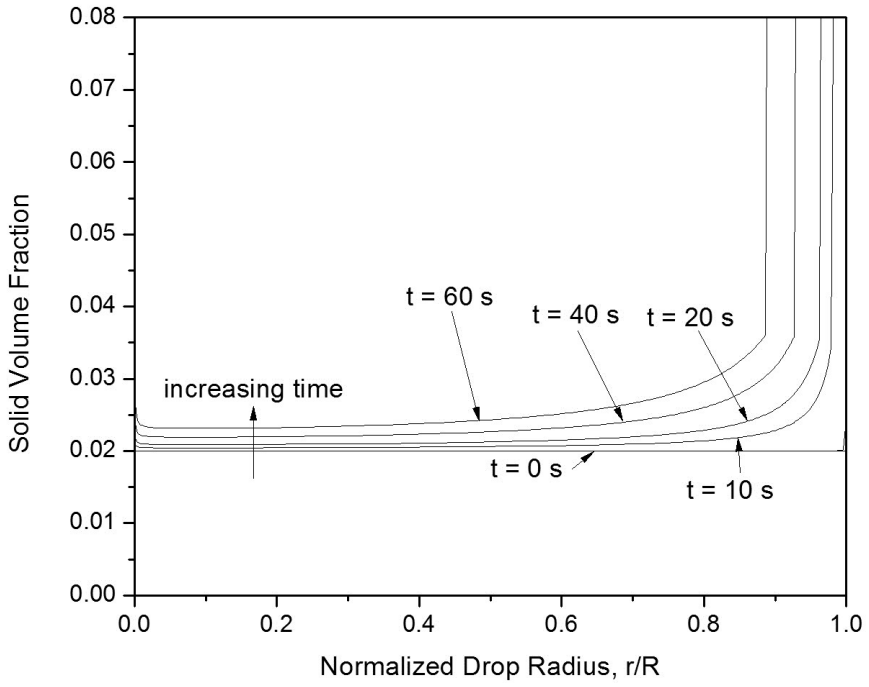

Figure 6.18 Modeled radially and temporally varying solid concentration profiles for a sessile drop with a maximum allowable solid volume fraction of 0.071.

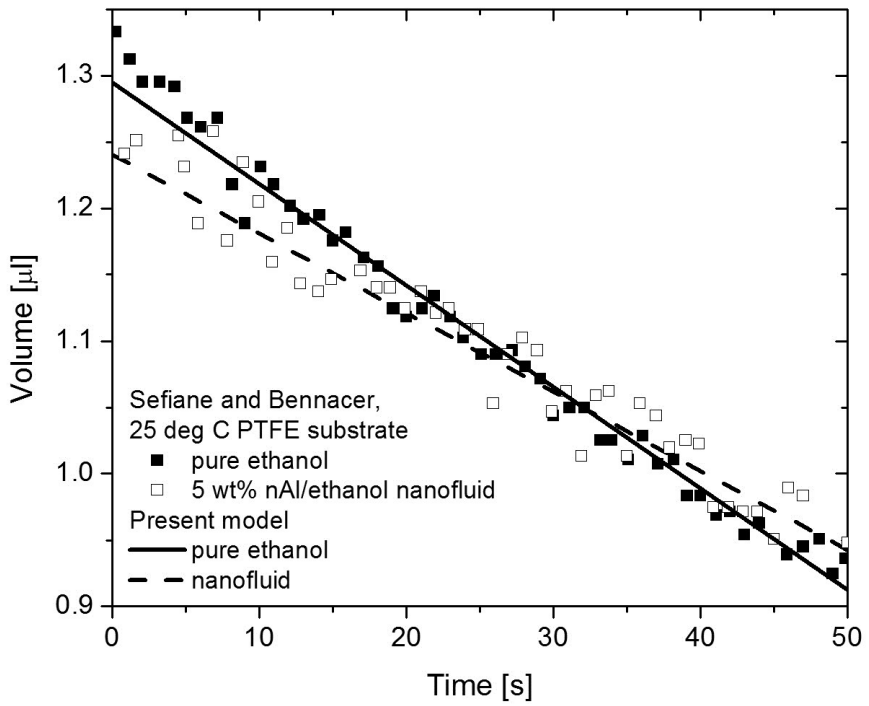

Figure 6.19 Sessile drop volume as a function of time, Sefiane and Bennacer [11] experiment versus present modeling simulations.

6.4.1 Model Inputs

For the simulation comparisons with the Sefiane and Bennacer [11] sessile drop experiments carried out at a substrate temperature of 25 °C, the ambient temperature was fixed at 298 K. Nanoparticle diameter was set at 60 nm and agglomerate size was estimated as the same from the nanofluids used here in pendant drop experiments, 250 nm in diameter. Values for liquid thermal conductivity, particle diffusion coefficient, liquid density, and vapor pressure were evaluated using the same temperature dependent relationships used in the pendant drop modeling (values are in Chapter 5). The linear mixture rule was applied to determine specific heat and density as before. The PTFE substrate thermal conductivity was set to 1 W/mK. Although this value of thermal conductivity is greater by a factor of four than the commonly reported PTFE values, this value was used for simulation in order to better match the pure ethanol evaporation rate measured in their experiment, as discussed later.

6.4.2 Numeric Sensitivities

The numerical sensitivity of the current sessile drop model was investigated in order understand to potential sources of uncertainty. Variations in the substrate thermal conductivity, ambient temperature, vapor pressure determination, number of radial discretization points, and the size of a time step were investigated. Investigation into the effect of the ambient temperature revealed little influence on the drop evaporation, due to the fact that the heat transfer rate at the drop liquid-gas interface is small relative to that at the liquid-solid interface, as discussed before. Radial discretizations from 100 to 1000 radial grid points within the drop were examined. A final value of 500 radial grid points was chosen because simulations at 500 grid points replicated all parameters calculated in 1000 point simulations to three significant figures and were approximately a factor of two faster in computational speed. Two methods for vapor pressure determination were tested, the Clausius-Clayperion relation and a polynomial fit for the experimentally measured temperature dependence of ethanol vapor pressure at atmospheric pressure. Simulations show negligible differences between the two vapor pressure evaluation methods. A time step study was carried out to determine the required time step for numerical convergence. A time step of 0.001 seconds was chosen as it

yielded results for all parameters in agreement, to three significant figures, with simulations carried out for a time step half as large. Unlike the pendant drop model, there is no interdependence of the time and radial step sizes for numerical stability. This is advantageous as it allows for much faster computation and is a result of using an algebraic conservation of volume approach, instead of a central differences finite differences numeric scheme as used in the pendant drop modeling.

The influence of the substrate thermal conductivity on the simulation results is large for values of thermal conductivity similar to those of the liquid drop. While for situations where thermal conductivity of the substrate is high relative to the liquid drop (e.g., for a steel or aluminum substrate where thermal conductivities are 50 and 237 W/mK, respectively), the substrate thermal conductivity is of negligible sensitivity. However for decreasing values of substrate thermal conductivity, the interface temperature between the liquid and solid also reduces, which has a strong influence on the evaporation rate of the drop through a reduction in energy transfer to the drop, the drop temperature, and hence the vapor pressure. Figure 6.10 shows the influence of thermal conductivity on temperature, from which it is possible to see why the influence is more pronounced for diminishing substrate thermal conductivity.

6.5 Results

Initial simulations carried out using a substrate thermal conductivity of 0.25 W/mK corresponding to a PTFE substrate resulted in pure and nanofluid evaporation rates significantly lower than those reported experimentally by Sefiane and Bennacer [11] for ethanol and 5 wt% nAl/ethanol nanofluid drops on PTFE. In searching for the source of the model-experiment discrepancy, modeling assumptions (most notably the spherical drop shape assumption) and input value errors and discrepancies (e.g. an incorrect value of thermal conductivity, temperature or pressure input to the model) were considered.

Meric and Erbil [86] showed that the effect of the non-spherical drop shape only minimally effects the evaporation rate. They showed that models for drop profile of increasing complexity and accuracy manage most notably to predict the evolution of drop contact angle and height with more precision. However, this added accuracy does

not change the evaporation rate, measured as the rate of change of volume in the current study.

A potential model deficiency is noted in the work of Dunn et al. [71]. They investigated the use of the present simplified model for drop evaporation for various substrate materials of different thermal conductivity. Dunn et al. showed excellent agreement between the simplified model and sessile drop evaporation experiments for a high thermal conductivity aluminum substrate but consistently under predicted the observed evaporation rate on PTFE substrate. This error was shown to increase with an increasing evaporation rate (vapor pressure) of a species; see Figure 6.20 for the results of Dunn et al. [71]. Based on this issue with low thermal conductivity substrates, an artificial increase in substrate thermal conductivity was implemented in order to better compare with the Sefiane and Bennacer [11] experiment. While the requirement for this "tuning" may illustrate some deficiency in either the reported substrate temperature in their experiments or the simplified model itself, it is important to point out that the substrate thermal conductivity "tuning" influences both pure fluid (ethanol) and nanofluid (5 wt% nAl/ethanol) simulations equally. Thus, the modeling results still focus on the reduction in evaporation rate for nanofluids and the ability of the proposed mechanism, based on a reduction in vapor pressure due to a displacement of ethanol at the drop surface by nanoparticle agglomerates, in predicting the evaporation rate change.

Prior to the modification of the PTFE substrate thermal conductivity, modeling results show qualitative agreement for the Sefiane and Bennacer [11] evaporation rate reduction but lack quantitative agreement. This is shown in Figure 6.22 and Figure 6.24. After adjustment of the substrate thermal conductivity (1 W/mK), an excellent match of the pure fluid evaporation rate is observed in Figure 6.21 and Figure 6.23. Figure 6.23 and Figure 6.24 show a comparison of modeling results for evaporation rate on an absolute and normalized basis for a range of maximum allowable nanoparticle agglomerate volume fraction and the Sefiane and Bennacer [11] results for pure ethanol and 5 wt% nAl/ethanol. The intersection of the modeling curves at variable maximum allowable volume fraction with the experimental result for the single nanofluid yield a volume fractions from 0.1 to 0.25 which result in fractal dimensions in the range of 1.8 to 2.3, reasonable when compared to literature values [16] [15] indicating that the

mechanism used for nanofluid evaporation rate reduction in the present model is not disproven.

Further modeling errors due to the simplifying assumptions of the present analysis are unlikely, due to the well reported nature of pinned drop evaporation. Hu et al. [83] have reported on the accuracy of simplified models during pinned phase evaporation, such as utilized here, and specifically compare analytic velocity profiles to those found using FEM numerical and experimental methods, illustrating that the assumptions employed in the simplified approaches were verified under similar experimental conditions.

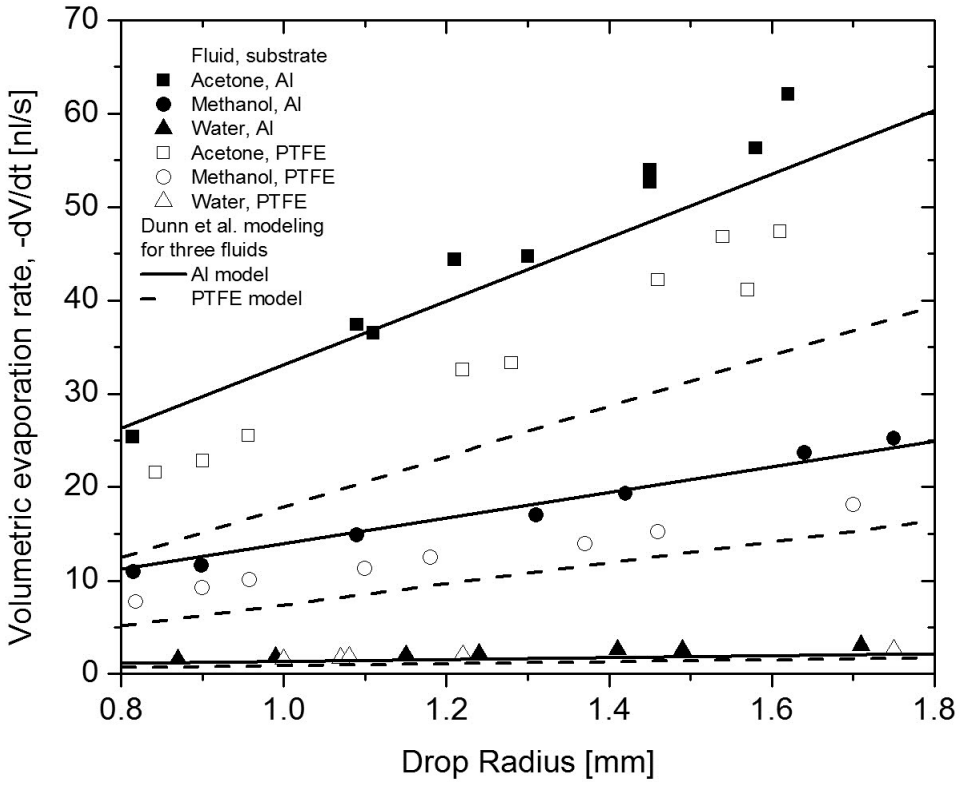

Figure 6.20 Sessile drop evaporation rate as a function of initial drop radius for three pure fluids on Al and PTFE substrates [71]. Accuracy of the Dunn et al. [71] modeling predictions for large substrate thermal conductivities (i.e., Al) is excellent; however, the model consistently under predicts evaporation rates on a PTFE substrate.

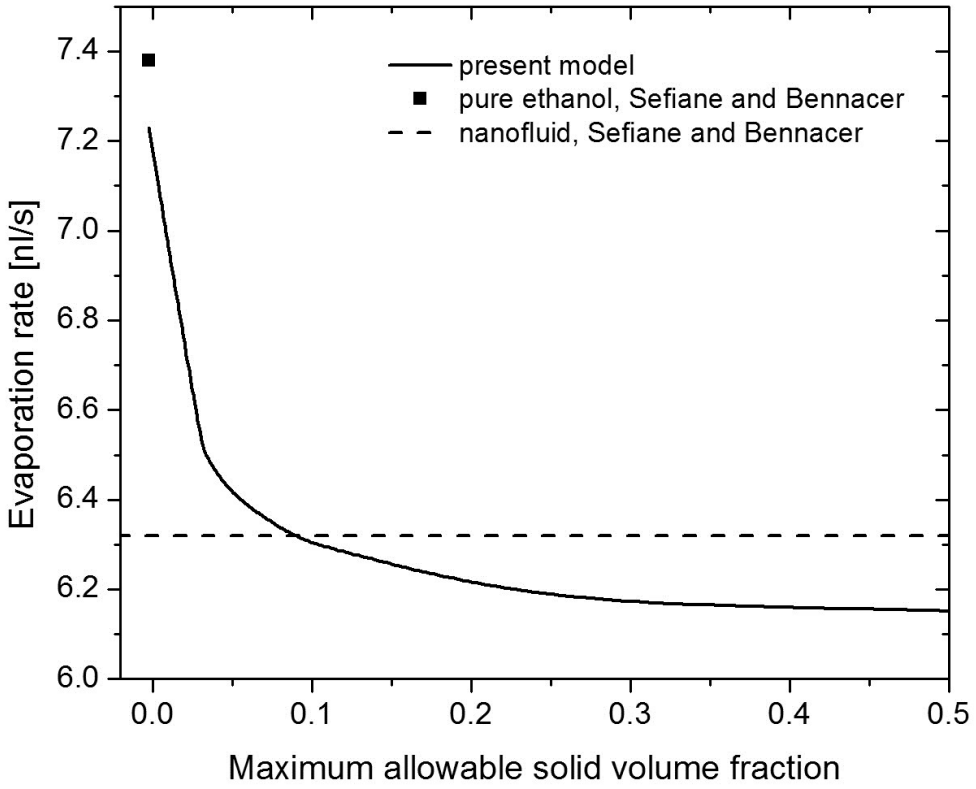

Figure 6.21 Modeled evaporation rate with increasing maximum allowable solid volume fraction for a fixed initial volume fraction of 0.02 and substrate thermal conductivity of 1 W/mK with comparison to the pure ethanol and nanofluid results of Sefiane and Bennacer [11]. The nanofluid results have unknown degree of agglomeration (agglomerate fractal dimension and size); hence, the maximum allowable solid volume fraction based on the present definition is unknown.

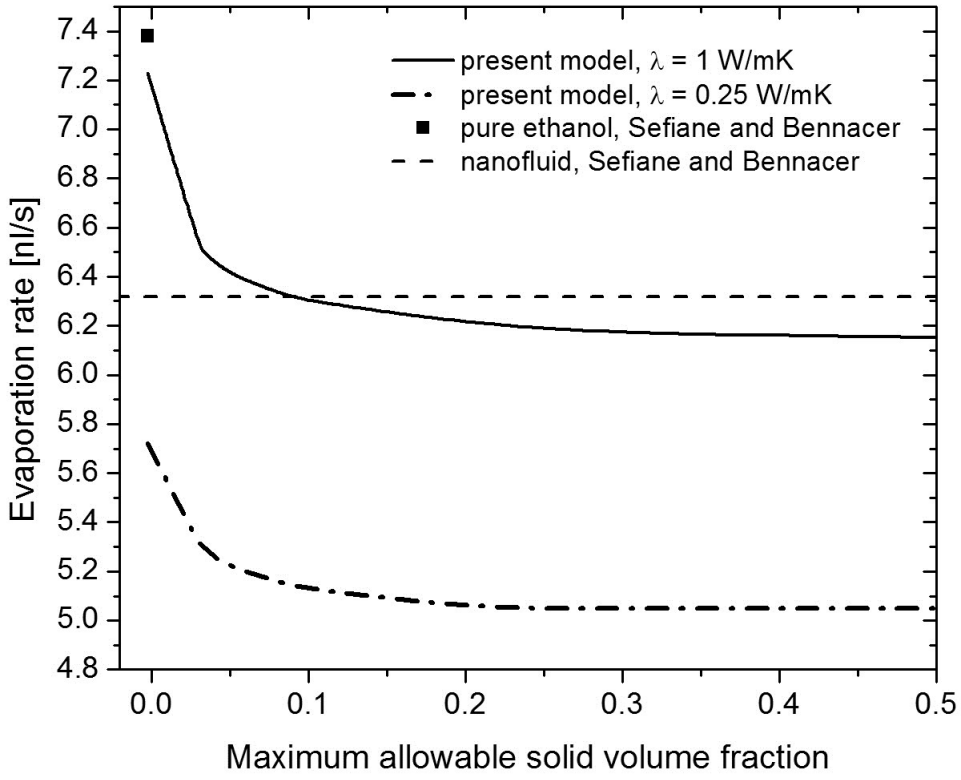

Figure 6.22 Modeled evaporation rate with increasing maximum allowable solid volume fraction for a fixed initial volume fraction of 0.02 and substrate thermal conductivity of 1 W/mK and 0.25 W/mK with comparison to the pure ethanol and nanofluid results of Sefiane and Bennacer [11]. The nanofluid results have unknown degree of agglomeration (agglomerate fractal dimension and size); hence, the maximum allowable solid volume fraction based on the present definition is unknown.

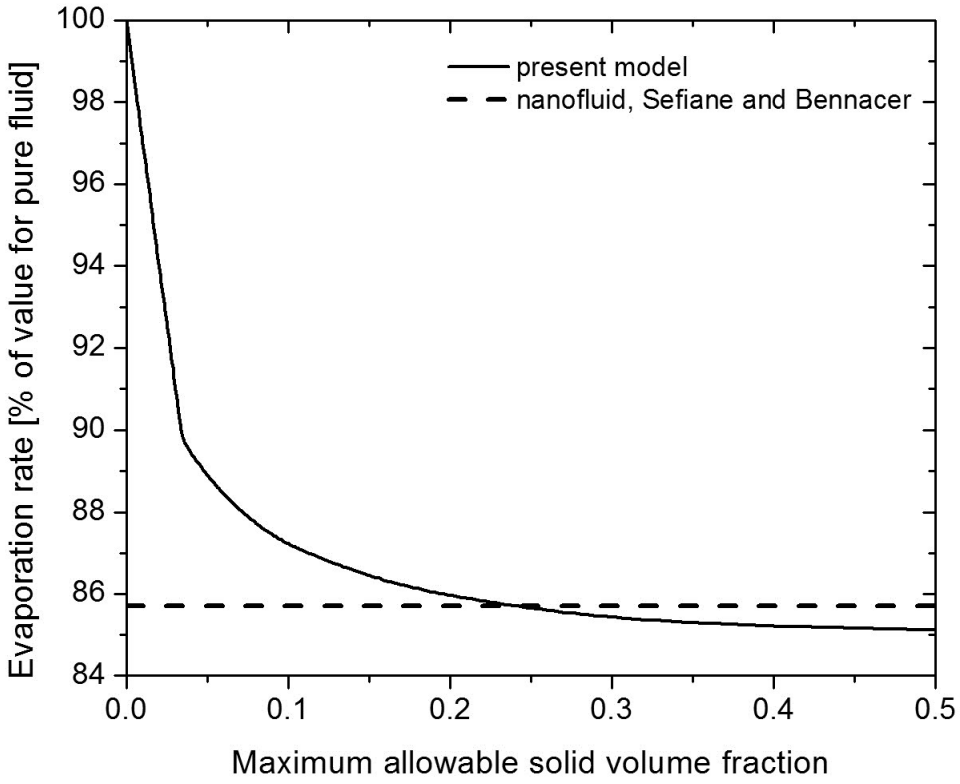

Figure 6.23 Modeled evaporation rate with increasing maximum allowable solid volume fraction for a fixed initial volume fraction of 0.02 and substrate thermal conductivity of 1 W/mK comparison to the nanofluid result of Sefiane and Bennacer [11]. The experimental nanofluid has an unknown degree of agglomeration (agglomerate fractal dimension and size); hence, the maximum allowable solid volume fraction based on the present definition is unknown.

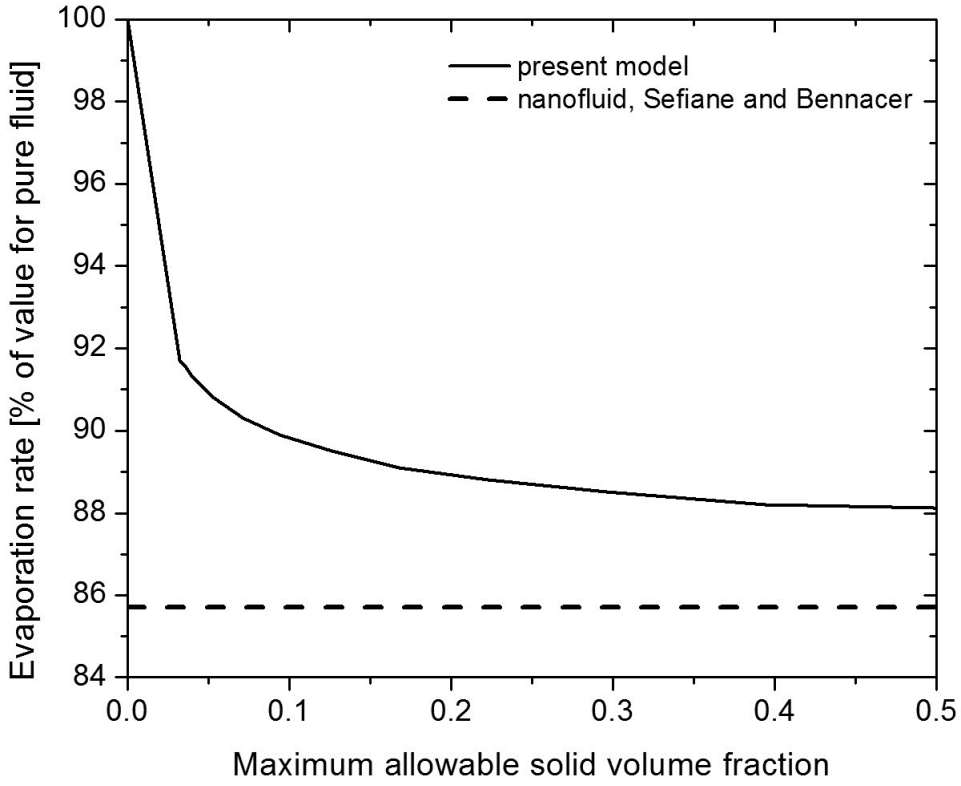

Figure 6.24 Modeled evaporation rate with increasing maximum allowable solid volume fraction for a fixed initial volume fraction of 0.02 and substrate thermal conductivity of 0.25 W/mK comparison to the nanofluid result of Sefiane and Bennacer [11]. The experimental nanofluid has an unknown degree of agglomeration (agglomerate fractal dimension and size); hence, the maximum allowable solid volume fraction based on the present definition is unknown.

7. Conclusion

The present study presents experimental results for the reduction in evaporation rate for nanofluid pendant drops compared to drops of the neat base fluid. A simple, and intuitive mechanism for the reduction of evaporation rate with increasing nanoparticle loading is proposed, based on a reduction in available vapor pressure at the drop surface due to nanoparticle agglomerates at the drop surface, and modeling predictions for pendant and sessile drop evaporation are presented based on that theory. These modeling predictions are compared to the present pendant drop experiments and to literature results for nanofluid sessile drops.

The work highlights that the proposed mechanism is able to explain and predict in modeling calculations the evaporation rate reduction observed in nanofluid drop evaporation experiments. The mechanism depends on the fractal nature of nanoparticle agglomerates in solution as the reduction in vapor pressure is dependent on the solid volume fraction at the surface. The conclusion here that the nanoparticle agglomeration plays a large role in nanofluid transport phenomena is consistent with the findings in the literature for nanofluid thermal conductivity [16], [17] and places an emphasis for future characterization of nanofluids agglomeration in all studies.

The proposed mechanism for nanofluid evaporation is included in a pendant drop evaporation modeling simulation and is found to replicate the current pendant drop evaporation results. The model predicts well the two most important features observed in the experimental data: the evaporation rate and the asymptote in nanofluid evaporation rate with increasing nanoparticle concentration.

Furthermore, a second sessile drop experiment conducted by Sefiane and Bennacer [11] was compared to modeling results carried out with the same mechanism for the influence of nanoparticles on evaporation. Simulation show reasonable experiment-model agreement, lending further support that the mechanism proposed here is applicable to the sessile drop and other configurations.

7.1 Future Work

In order to further evaluate the validity of the proposed mechanism for the influence of nanoparticles on drop evaporation, further efforts in the both modeling and

experimentation are necessary. Implementation of well designed experiments would allow for further investigation of the proposed mechanism as well as discovery of influence of nanoparticles on evaporation at broader regime of temperature, pressure, and drop/droplet size. Additionally, characterization of early time non-steady evaporation would be of interest. Such experiments would provide investigation of the potential for nanofluid use in applications such as spray cooling, biological/agricultural applications, and combustion, among others.

Future experiments would be most useful if they were to report a complete set of parameters necessary for modeling, including temperature, humidity, velocity profiles, all geometrical parameters, and all material properties. Evaporation is extremely sensitive to a number of parameters and model validation requires accurate boundary conditions and properties. As part of nanofluid experimental efforts a standard set of nanofluid properties are required. This characterization should at the very least include agglomerate and nanoparticle size and fractal dimension. Of course, other properties can also be required depending on the problem of interest.

The investigation of nanofluids comprised of other base fluids (e.g., hydrocarbons, water, and methanol) would provide a widening of the nanofluids investigated. However, these fluids would require different evaporation environments to provide a feasible and accurate experiment. Nanofluids comprised of different non-spherical nanomaterials would also be of interest but would potentially require a change to the solid volume fraction approach utilized here, most notably in the case of carbon nanotubes, graphene sheets, or ceramic disk type particles. Additionally, stability and surface chemistry of these different nanomaterials could invalidate the current assumptions of slow agglomeration rate.

Accurate and a more complete experimental characterization of nanofluid evaporation would allow for the interrogation of simplified modeling approaches as reported here and detailed approaches wherein computational method such as finite differences or finite element schemes are used to solve the fully descriptive conservation and transport equations. Efforts to characterize nanofluid evaporation and other transport phenomena and explain and model the influence of nanoparticles on these transport phenomena will be required to assess the potential of nanofluids in application.

REFERENCES

[1] D. O'Neal, L. Hirsch, N. Halas, J. Payne and J. West, "Photo-thermal tumor ablation in mice using near infrared-absorbing nanoparticles," *Cancer Lett.,* vol. 109, no. 2, pp. 171-176, Feb. 2004.

[2] S. K. Das, S. U. Choi, W. Yu and T. Pradeep, *Nanofluids Science and Technology*, Hoboken, NJ: John Wiley & Sons, 2008.

[3] W. Yu and H. Xie, "A Review on Nanofluids: Preparation, Stability Mechanisms, and Applications," *J. of Nanomaterials,* vol. 2012, no. 1, p. 435873, Jan. 2012.

[4] N. Nguyen, A. Beyzavi, K. Ng and X. Huang, "Kinematics and deformation of ferrofluid droplets under magnetic actuation," *Microfluid Nanofluid,* vol. 3, no. 5, pp. 571-579, Jan. 2007.

[5] S. S. Gupta, V. M. Shiva, S. Krishnan, T. Sreeprasad and P. K. Singh, "Thermal conductivity enhancement of nanofluids containing graphene nanosheets," *J. Appl. Physics,* vol. 110, no.1, p. 084302, Oct. 2011.

[6] R. Kumar and D. Milanova, "Effect of surface tension on nanotube nanofluids," *Appl. Physics Lett,* vol. 94, no. 1, p. 073107, Feb. 2009.

[7] P. Vassallo, R. Kumar and S. D'Amico, "Pool boiling heat transfer experiments in silica-water nano-fluids," *Int. J. Heat Mass Transfer,* vol. 47, no. 2, pp. 407-411, Jan. 2004.

[8] H. Tyagi, P. Phelan, R. Prasher, T.Lee, J. Pacheo and P. Arentzen, "Increased hot-plate ignition probability for nanoparticle-laden diesel fuel," *Nano Lett.,* vol. 8, no. 5, pp. 1410-1416, Apr. 2008.

[9] J. S. Basha and R. Anand, "Role of nanoadditive blended biodiesel emulsion fuel on the working characteristics of a diesel engine," *J. Renewable Sustainable Energy,* vol. 3, no. 1, p. 023106, Apr. 2011.

[10] C. H. Chon, S. Paik, J. B. T. Jr. and K. D. Kihm, "Effect of nanoparticle sizes and number densities on the evaporation and dryout characteristics for strongly pinned nanofluid droplets," *Langmuir,* vol. 23, no. 6, pp. 2953-2960, Dec. 2007.

[11] K. Sefiane and R. Bennacer, "Nanofluids droplets evaporation kinetics and wetting dynamics on rough heated substrates," *Advanced Colloid Interface Sci.,* vol. 147-148, no. 1, pp. 263-271, June 2009.

[12] R. Chen, T. Phuoc and D.Martello, "Surface tension of evaporating nanofluid droplets," *Int. J. Heat Transfer,* vol. 54, no. 11-12, pp. 2459-2466, Jan. 2011.

[13] R. Chen, T. Phuoc and D. Martello, "Effects of nanoparticles on nanofluid droplet evaporation," *Int. J. Heat Mass Transfer,* vol. 53, no. 19-20, pp. 3677-3682, Sept. 2010.

[14] J. R. Moffat, "Experimental investigation into the evaporation behavior of pure and nanofluid droplets," PhD. dissertation, School of Eng., Univ. of Edinburgh, Edinburgh, UK, 2010.

[15] P. Keblinski, J. Eastman and D. Cahill, "Heat transfer characteristics of nanofluids: a review," *Materials Today,* vol. 8, no. 1, pp. 36-44, June 2005.

[16] P. Keblinski, R. Prasher and J. Eapen, "Thermal conductance of nanofluids: is the controversy over?" *J. Nanoparticle Research,* vol. 10, no. 7, pp. 1089-1097, Oct. 2008.

[17] S. K. Das, S. Choi and H.E. Patel, "Heat transfer in nanofluids- a review," *Heat Transfer Eng.,* vol. 27, no. 10, pp. 3-19, Feb. 2006.

[18] D. Lee, "Thermophysical properties of interfacial layer in nanofluids," *Langmuir,* vol. 23, no. 11, pp. 6011-6018, Apr. 2007.

[19] R. Prasher, W. Evans, P. Meakin, J. Fish and P. Keblinski, "Effect of aggregation on thermal conduction in colloidal nanofluids," *Appl. Physics Lett.,* vol. 89, no. 14, pp. 143119-143119-3, Oct. 2006.

[20] P. Keblinski, S. Phillipot, S. Choi and J. Eastman, "Mechanisms of heat flow in suspensions of nano-sized particles (nanofluids)," *Int. J. Heat Mass Transfer,* vol. 45, no. 4, pp. 855-863, Feb. 2002.

[21] C. Richter, H. J. Viljoen and N.F.J. van Rensburg, "Effective thermal conductivity of estimates of particulate mixtures," *J. Appl. Physics,* vol. 93, no. 5, pp. 2663-2670, Mar. 2003.

[22] N. Ce-Wen, et al. "Effective thermal conductivity of particulate composites with interfacial thermal resistance," *J. Appl. Physics,* vol. 81, no. 1, pp. 6692-6699, Feb. 1997.

[23] R. Prasher, D. Song, J. Wang and P. Phelan, "Measurements of nanofluid viscosity and its implications for thermal applications," *Appl. Physics Lett.,* vol. 89, no. 13, pp. 133108, Sept. 2006.

[24] N. Masoumi, N. Sohrabi and A. Behzadmehr, "A new model for calculating the effective viscosity of nanofluids," *J. Physics D: Appl. Physics,* vol. 42, no. 1, pp. 055501, Feb. 2009.

[25] P. K. Namburu, D. P. Kulkarni, D. Mirsa and D. K. Das, "Viscosity of copper oxide nanoparticles dispersed in etheylene glycol and water mixture," *Experimental Thermal and Fluid Sci.,* vol. 32, no. 2, pp. 397-402, May 2007.

[26] S. Tanvir and L. Qiao, "Surface tension of nanofluid-type fuels containing suspended nanomaterials," *Nanoscale Research Lett.,* vol. 7, no. 7, p. 226, Apr. 2012.

[27] M. Moosavi, E. Goharshadi and A. Youssefi, "Fabrication, characterization, and measurement of some physicochemical properties of ZnO nanofluids," *J. Heat Fluid Flow,* vol. 31, no. 4, pp. 599-605, Aug. 2010.

[28] S. Kim, I. Bang, J. Buongiorno and L. Hu, "Surface wettability change during pool boiling of nanofluids and its effect on critical heat flux," *Int. J. Heat Mass Transfer,* vol. 50, no. 19-20, pp. 4105-4116, Sept. 2007.

[29] S. Murshed, K. Leong and C. Yang, "Thermophysical and electrokinetic properties of nanofluids – A critical review," *Appl. Thermal Eng.,* vol. 28, no. 17-18, pp. 2109-2125, Dec. 2008.

[30] S. Murshed, S.H. Tan and N.T. Nguyen, "Temperature dependence of interfacial properties and viscosity of nanofluids for droplet-based microfluidics," *J. Physics D,* vol. 41, no. 8, p. 085502, Mar. 2008.

[31] T. Sattelmayer, *Influence of Turbulence and Secondary Flow on the Subcooled Flow Boiling in Characteristical Reactor Configurations*, [Online]. Available: http://www.td.mw.tum.de/tum-td/de/forschung/themen/subcooled_flow_boiling [1 May 2014]

[32] B. Mikic and W. Rohsenow, "A new correlation of pool-boiling data including the effect of heating surface characterisitcs," *Heat Transfer,* vol. 91, no. 2, pp. 245-250, Aug. 1969.

[33] P. B. Whalley, *Two-Phase Flow and Heat Transfer*, Oxford, England: Oxford University Press, 1996.

[34] S. Das, N. Putra, P. Thiesen and W. Roetzel, "Temperature dependence of thermal conductivity enhancement for nanofluids," *J. Heat Transfer,* vol. 125, no. 4, pp. 567-574, July 2003.

[35] D. Ganesh and G. Gowrishankar, "Effect of nano-fuel additive on emission reduction in a biodiesel fuelled CI engine," in *International Conference on Electrical and Control Engineering (ICECE),* Yichang, China, 2011, pp. 3453-3459.

[36] M. Jones, "Ignition and Combustion Characteristics of Nanoscale Metqal and Metal Oxide Additives in Biofuel(Ethanol) and Hydrocarbons," M.S. thesis, Dept. Mech. Eng., The Univ. of Toledo, Toledo, OH, 2011.

[37] M. Jones, C. Li, A. Afjeh and G. Peterson, "Experimental study of combustion characteristics of nanoscale metal and metal oxide additives in biofuel (ethanol)," *Nanoscale Research Lett.,* vol. 6, no. 1, pp. 1-12, Mar. 2011.

[38] W. J. Gerken, "The Autoignition of Tetralin, an endothermic fuel," M.S. thesis, MANE, RPI, Troy, NY, 2013.

[39] M. Lenin, M. Swaminathan and G. Kumaresan, "Performance and emission characteristics of a DI diesel engine with a nanfuel additive," *Fuel,* vol. 109, no. 1, pp. 362-365, July 2013.

[40] W. Ranz and W. Marshall Jr., “Evaporation from drops,” *Chem. Eng. Progress,* vol. 48, no. 3, pp. 173-180, Apr. 1952.

[41] W. Sirignano, *Fluid Dynamics and Transport of Droplets and Sprays*, New York, NY: Cambridge University Press, 2010.

[42] D. Charlesworth and W. Marshall, “Evaporation from drops containing dissolved solids,” *AlChE,* vol. 6, no. 1, pp. 9-23, June 1960.

[43] C. Megaridis and W. Sirignano, "Numerical modeling of a vaporizing multicomponent droplet," in *International Symposium on Combustion*, Orleans, France, vol. 23, no. 1, pp. 1413-1421, 1990.

[44] Y. Gan and L. Qiao, “Evaporation characteristics of fuel droplets with the addition of nanoparticles under natural and forced convections,” *Int. J. Heat Mass Transfer,* vol. 54, no. 23-24, pp. 4913-4922, Nov. 2011.

[45] Y. Gan and L. Qiao, “Combustion characteristics of fuel droplets with addition of nano and micron-sized aluminum particles,” *Comb. and Flame,* vol. 158, no. 1, pp. 354-368, Feb. 2011.

[46] Z. Meng, S. M. Hashmi and M. Elimelech, “Aggregation rate and fractal dimension of fullerene nanoparticles via simultaneous multiangle static and dynamic light scattering measurement,” *J. of Colloid and Interface Sci.,* vol. 392, no. 15, pp 27-33, Feb. 2013.

[47] J. Kim and T. A. Cramer, "Improved orthokinetic coagulation model for fractal colloids: aggregation and breakup," in *Advances in population balance modelling, Second International Conference on Population Balance Modeling,* Valencia, Spain, vol. 61, pp. 45-53, 2004.

[48] J. Kumar, "Numerical approximations of population balance equations in particulate systems," PhD. dissertation, Dept. of Math., Otto-von-Guericke-Universitat, Magdeburge, Germany, 2006.

[49] M. Lin et al., “Universal diffusion-limited colloid aggregation,” *J. Physics Condensed Matter,* vol. 2, no. 13, p. 3093-3113, Apr. 1990.

[50] V. Kanniah et al., “Fractal analysis as a complimentary technique for characterizing nanoparticle size distributions,” *Powder Technology,* vol. 226, no. 1, pp. 189-198, Aug. 2012.

[51] W. J. Gerken, A. V. Thomas, N. Koratkar and M. A. Oehlschlaeger, “Nanofluid pendant droplet evaporation: Experiments and modeling,” *Int. J. Heat Mass Transfer,* vol. 74, no. 1, pp. 263-268, July 2014.

[52] *DROP Image Advanced*, 2nd ed. Rame-Hart Instrument Co., Succasunna, NJ, 2008.

[53] A. Karperien, "FracLac for ImageJ," [Online]. Available: http://rsb.info.nih.gov/ij/plugins/fraclac/FLHelp/Introduction.htm [21 July 2014].

[54] C. Law, T.Y. Xiong and C. Wang, “Alchohol droplet vaporization in humid air,” *Int. J. Heat Mass Transfer,* vol. 30, no. 7, pp. 1435-1443, July 1987.

[55] I. Javed, S.W.Baek and K.Waheed, “Evaporation characteristics of heptane droplets with the addition of aluminum nanoparticles at elevated temperatures,” *Combust. Flame,* vol. 160, no. 1, pp. 170-183, Jan. 2013.

[56] *Lange's Handbook of Chemistry*, 10th ed., New York, NY: McGraw-Hill, pp. 1661-1665, 1967.

[57] G. Brenn et al., “Evaporation of acoustically levitated mulit-component liquid droplets,” *Int. J. Heat Mass Transfer,* vol. 50, no. 25-26, pp. 5073-5086, Dec. 2007.

[58] G. Brenn, L. Deviprasath and F. Durst, "Components and experiments on the evaporation of multi-component droplets," in *Proc. 9th Int. Conf. Liquid Atomiz. Spray Syst.*, Sorrento Italy, p. 1309, 2003.

[59] B. Abramzon and W. Sirignano, “Droplet vaporization model for spray combustion calculations,” *Int. J. Heat Mass Transfer,* vol. 32, pp. 1605-1618, Sept. 1989.

[60] Homer and J. Christopher et al., “Measurements and simulations of the near-surface composition of evaporating ethanol-water droplets,” *Physical Chemistry Chemical Physics,* vol. 11, no. 36, pp. 7780-7791, June 2009.

[61] J. Xingmao et al., “Numerical Simulation of Ethanol-Water-NaCl Droplet Evaporation,” *Industrial & Eng. Chemistry Research,* vol. 49, no. 12, pp. 5631-5643, May 2010.

[62] J. Xingmao, "Engineering of Aerosol Nanoparticle Architectures," PhD. dissertation, ChNE, The Univ. of New Mexico, Albuquerque, NM, 2006.

[63] *Perry's Chemical Engineers' Handbook*, New York, NY: McGraw-Hill, 2008.

[64] A. L. Lindsay and L. A. Bromley, ”Thermal Conductivity of Gas Mixtures,” *Industrial & Eng. Chemistry,* vol. 42, no. 8, pp. 1508-1511, Aug. 1950.

[65] E. N. Fuller, P. D. Schettler and J. C. Giddings, “New method for prediction of binary gas-phase diffusion coefficients,” *Industrial & Eng. Chemistry,* vol. 58, no. 5, pp. 18-27, May 1966.

[66] *Thermophysical Properties : Ethanol* [Online]. Available: https://www.thermalfluidscentral.org/encyclopedia/index.php/Thermophysical_Properties:_Ethanol. [21 July 2014].

[67] *Thermophysical Properties: Air at 1 atm* [Online]. Available: https://www.thermalfluidscentral.org/encyclopedia/index.php/Thermophysical_Properties:_Air_at_1_atm. [21 July 2014].

[68] H. Y. Erbil, “Evaporation of pure liquid sessile and spherical suspended drops: A review,” *Advances in Colloid and Interface Sci.,* vol. 170, no. 1-2, pp. 67-86, Jan. 2012.

[69] J. Drelich, J. D. Miller and R. J. Good, “The effect of drop (bubble) size on advancing and receding contact angles for heterogeneous and rought solid surfaces as observed with sessile-drop and captive-bubble techniques,” *J. Colloid and Interfacial Sci.,* vol. 179, no. 1, pp. 37-50, Apr. 1996.

[70] M. E. R. Shanahan, ”Simple theory of ‘Stick-Slip’ wetting hysteresis,” *Langmuir,* vol. 11, no. 3, pp. 1041-1043, July 1995.

[71] G. Dunn, S. Wilson, B. Duffy, S. David and K. Sefiane, “A mathematical model for the evaporation of a thin sessile liquid droplet: Comparison between experiment and theory,” *Colloids and Surfaces A,* vol. 323, no. 1-3, pp. 50-55, June 2007.

[72] H. Y. Erbil, G. McHale and M. I. Newton, "Drop evaporation on solid surfaces: constant contact angle mode," *Langmuir,* vol. 18, no. 7, pp. 2636-2641, Mar. 2002.

[73] H.-J. Butt, D. S. Golovko and E. Bonaccurso, "On the derivation of young's equation for sessile drops: nonequilibrium effects due to evaporation," *J. Physical Chem. B,* vol. 111, no. 19, pp. 5277-5283, Apr. 2007.

[74] R. Bhardwaj, X. Fang and D. Attinger, "Pattern formation during the evaporation of colloidal nanolieter drop: a numerical and experimental study," *New J. of Physics,* vol. 11, no. 1, p. 075020, July 2009.

[75] Y. Y. Tarasevich, "Simple analytical model of capillary flow in an evaporating sessile drop," *Physical Review E,* vol. 71, p. 027301, Feb. 2005.

[76] E. Widjaja and M. T. Harris, "Particle deposition study during sessile drop evaporation," *AIChE ,* vol. 54, no. 9, pp. 2250-2260, July 2008.

[77] V. N. Truskett and K. J. Stebe, "Influence of surfactants on an evaporation drop: fluorescence images and particle deposition patterns," *Langmuir,* vol. 19, no. 20, pp. 8271-8279, Sept. 2003.

[78] H. Song, Y. Lee, S. Jin, H.Y. Kim and J. Y. Yoo, "Prediction of sessile drop evaporation considering surface wettability," *Microelectronic Eng.,* vol. 88, no. 11, pp. 3249-3255, Nov. 2011.

[79] R. D. Deegan, O. Bakajin, T. F. Dupont, G. Huber, S. R. Nagel and T. A. Witten, "Contact line deposits in an evaporating drop," *Physical Review E,* vol. 62, no. 1, pp. 756-765, July 2000.

[80] F. Fan and K. J. Stebe, "Assembly of colloidal particles by evaporation on surfaces with patterned hydrophobicity," *Langmuir,* vol. 20, no. 8, pp. 3062-3067, Mar. 2004.

[81] K. Sefiane, L. Tadrist and M. Douglas, "Experimental study of evaporation water-ethanol mixture sessile drop: influence of concentration," *Int. J. Heat Mass Transfer,* vol. 46, no. 23, pp. 4527-4534, Nov. 2003.

[82] H. Hu and R. G. Larson, "Analysis of the effects of marangoni stresses on the microflow in an evaporating sessile droplet," *Langmuir,* vol. 21, no. 9, pp. 3972-3980, Mar. 2005.

[83] H. Hu and R. G. Larson, "Analysis of the microfluid flow in an evaporating sessile droplet," *Langmuir,* vol. 21, no. 9, pp. 3963-3971, Mar. 2005.

[84] M. A. Saada, S. Chikh and L. Tadrist, "Numerical investigation of heat and mass transfer of an evaporating sessile drop on a horizontal surface," *Physics of Fluids,* vol. 22, no. 1, p. 112115, Nov. 2010.

[85] H. Hu and R. G. Larson, "Evaporation of a sessile droplet on a substrate," *J. Physical Chemistry B,* vol. 106, no. 6, pp. 1334-1344, Jan. 2002.

[86] R. A. Meric and H. Y. Erbil, "Evaporation of sessile drops on solid surfaces: pseudospherical cap geometry," *Langmuir,* vol. 14, no. 7, pp. 1915-1920, Mar. 1998.

APPENDIX A (CODE)

Sessile_Driver.m

```
% will Gerken
% Sessile Driver

clear all
close all

dbstop if error

tic
df      =[1.8
1.8];%[1.0,1.05,1.1,1.2,1.4,1.6,1.8,2,2.2,2.4,2.6,2.8,3];%[1.3 1.3 1.8
2 3];%
Rnano   =(1/2)*60e-9;%(1/2).*[30:30:90].*1e-9;%
Rg      =(1/2)*250e-9;%(1/2).*[70:50:300].*1e-9;
Sublayer= {'PTFE_Mod'};
skip=0;

for lk=1:length(Sublayer)
    for i=1:size(df,2)
        for j=1:size(Rnano,2)
            for k=1:size(Rg,2)
                if i==1
                    if skip ~= 1
                        disp('Running Pure fluid...')
                        [Vdot(i,j,k)
Volf(i,j,k),bf,nf]=sessile_evap_v6(df(i),Rnano(j),Rg(k),0,Sublayer{lk})
;
                        Volf(i,j,k)=0;
                    else
                        [Vdot(i,j,k) Volf(i,j,k) bf]=[-7.3e-12 0 ];
                    end
                else
                    [Vdot(i,j,k)
Volf(i,j,k),bf,nf]=sessile_evap_v6(df(i),Rnano(j),Rg(k),0.02,Sublayer{l
k});
                end
            end
        end
    end
end
h=figure(100*j)
plot(Volf(:,1,1),squeeze(-Vdot(:,1,1)),'+-');
hold on
if i>2
    plot(0,-bf,'ko','LineWidth',2);
    hold on;
end
if i==size(df,2)
    plot([0 max(Volf)],[-nf -nf],'ro-','LineWidth',2);
end
plot(0:0.1:max(Volf),nf*ones(size(0:0.1:max(Volf)),1),'r+-')
axis([0 1 0 (10)*10^(-12)]);
```

```
xlabel('Maximum Volume Fraction')
ylabel('Evaporation rate')
legend('Simulation','Base-Fluid Sefiane and Bennacer @ 25C','Nanofluid
Sefiane and Bennacer @ 25C')
grid on;

disp(['Nominal Evap of: ', num2str(Vdot(end,1,1))]);
disp('Percent Decrease of :');
PD=100*abs((Vdot(:,:,:)-Vdot(1,1,1))/Vdot(1,1,1))
print(h,'-dpdf',['absolute_study.pdf']);


h=figure(200)
plot(squeeze(Volf(:,:,:)),100-PD,'+-');
hold on;

plot(0:0.1:max(Volf),(nf/bf)*100*ones(size(0:0.1:max(Volf))),'r+-');
grid on;
xlabel('Max Volume Fraction');
ylabel('Percent of nominal evap')
legend('Simulation','Sefiane and Bennacer @ 25C')
print(h,'-dpdf',['percent_study.pdf']);
toc

% figure(300)
% hold on;
% Rg = (1/2).*[70:50:300].*1e-9;
% Rnano   =(1/2)*60e-9;%(1/2).*[30:30:90].*1e-9;%
% df =[1.0,1.05,1.1,1.2,1.4,1.6,1.8,2,2.2,2.4,2.6,2.8,3];%[1.3 1.3 1.8
2 3];%
% for i=1:length(df)
%     for j=1:length(Rg)
%         volfM           = pi()/6;% max volume fraction
%         ra              =((Rg(j)/Rnano)^df(i)*Rnano^2)^(1/2);% from
Keblinski Surface Area Equivilent radius
%         inVolf          = (Rg(j)/Rnano)^(df(i)-3);
%         MaxVolf(i,j)    = volfM*inVolf;
%         plot3(df(i),Rg(j),MaxVolf(i,j),'k+');
%     end
% end
% hold on;
% contour3(repmat(df,[5,1])',repmat(Rg,[13,1]),MaxVolf,100);
% grid on
% xlabel('Fractal Dimension (n.d.)');
% ylabel('Agglomerate Radius (m)');
% zlabel('Maximum Volume Fraction (n.d.)');
```

Sessile_Evap_v6

```
function [V_return, MaxVolf, base_fluid,
nano_fluid]=sessile_evap_v6(df, Rnano, Rg, V0, stringSublayer)

close all

%%
% ****************************************
% Load In Data from Sefiane and Bennacer
% @ 25 Celcius
% ****************************************
BaseFluid       = load('Base_Fluid.mat');
BF_RADIUS       = 0.0024/2;
BF_t            = BaseFluid.data(4:end,1);% Seconds
loc = find(BF_t>60,1);
BF_t            = BaseFluid.data(4:loc,1);% Seconds
BF_theta        = BaseFluid.data(4:loc,2);%Degrees
BF_dtdt         = gradient(BF_theta,BF_t);
    BF_Dradius      = BaseFluid.data(4:loc,3)/1000;% mm to m
    BF_height       = BaseFluid.data(4:loc,4)/1000;% mm to m
    BF_volume       = BaseFluid.data(4:loc,5)/(1000^3);% mm^3 to m^3
    BF_surfaceArea  = BaseFluid.data(4:loc,6)/(1000^2);% mm^2 to m^2
    BF_mass         = BaseFluid.data(4:loc,7);% kg
    BF_dmdt         = gradient(BF_mass,BF_t);% kg/s
NanoFluid       = load('Nano_Fluid_Data.mat');
NF_RADIUS       = 0.0024/2;
NF_t            = NanoFluid.data(4:end,1);% Seconds
loc = find(NF_t>45,1);
NF_t            = NanoFluid.data(4:loc,1);% Seconds
NF_theta        = NanoFluid.data(4:loc,2);%Degrees
NF_dtdt         = gradient(NF_theta,NF_t);
    NF_Dradius      = NanoFluid.data(4:loc,3)/1000;% mm to m
    NF_height       = NanoFluid.data(4:loc,4)/1000;% mm to m
    NF_volume       = NanoFluid.data(4:loc,5)/(1000^3);% mm^3 to m^3
    NF_surfaceArea  = NanoFluid.data(4:loc,6)/(1000^2);% mm^2 to m^2
    NF_mass         = NanoFluid.data(4:loc,7);% kg
    NF_dmdt         = gradient(NF_mass,BF_t);% kg/s
%%
% ****************************************
% Inputs
% ****************************************

% System
P               =101325; % Pa
Yinf            =0;%Gas Mass Fraction @ infinity
%T0              = 273.15 + 25;%; 21.85   65
%V0              = 0.02;
if V0==0
    Rs(1)           = BF_RADIUS;% - 0.2*BF_RADIUS;
    Theta(1)        = BF_theta(1)*pi/180;% radians
else
    Rs(1)           = NF_RADIUS;
    Theta(1)        = NF_theta(1)*pi/180;% radians
end
seconds         = floor(BF_t(end)) + 10;
```

```
dt              = 0.001;
loop            = seconds./dt;
interval        = 10; %
Tinf            = 273.15 + 25;
T0=Tinf;

% Solid Surface parameters
if strcmpi(stringSublayer,'Al')
    lams=237;
elseif strcmpi(stringSublayer,'Steel')
    lams=50;
elseif strcmpi(stringSublayer,'PTFE_Mod')
    lams            = 1;%.91;%0.25;%0.25;%237 ;% 237; % conductivity of
aluminum from in W./mK
elseif strcmpi(stringSublayer,'PTFE');
    lams            =0.25;
end
%0.91 matches well to base fluid results
hs              = 0.0015;%0.0015;%1.5 mm thickness

% Gas Parameters
mwg             =28.97;% kg/kmol
Tbg             =77.36;
Cpg             =@(T) (10.^3).*1.005; %J./kgK
X               =[150;200;250;300;350];
Y               =[2.3364;1.7458;1.3947;1.1614;.995];
rhog            =fit(X,Y,'cubicspline');%kg./m.^3
lamg            =@(T) (7.071e-5).*(T-273.15)+2.428e-2;%W./mK
visg            =@(T) (4.7225e-8).*(T-273.15)+(1.7238e-5); %(Ns./m.^2)
viscosity of air

% Liquid Parameters
mwl             =46.07;% (kg./kmol)
Tbf             =351.8;% in K
Dg              =@(T) (1./100).*(1./100).*(((10.^-
3).*T.^1.75.*(1./mwg+1./mwl).^(1./2))./((P./101325).*(50.36.^(1./3)+20.
1.^(1./3)).^2)); %(m.^2./s) Diffusion of liquid into air from FSG
Cpf             =@(T) (1e3./mwl).*((-8.28925e-
5).*T.^2+(0.216104).*T+8.28126);%(J./kgK)  Specific heat of liquid in
vapor phase
Cpl             =@(T) (1e3./mwl).*(98.39+0.5368.*(T-273.15));%(J./kgK)
Specific heat of liquid as liquid
rhol            =@(T) (-0.8544).*(T-273.15)+806.43;% (kg./m.^3)
rhogf           =@(T) exp((-3.3681)+(5.2492e-2).*(T-273.15)+(5.1630e-
5).*(T-273.15).^2+(-1.9542e-6).*(T-273.15).^3+(8.6893e-9).*(T-
273.15).^4+(-1.1451e-11).*(T-273.15).^5); % from THermal FLuids
Online....
Pv              =@(T) (133.3224).*((4.0325e-4).*(T-273.15).^3+(2.7952e-
2).*(T-273.15).^2+(0.81796).*(T-273.15)+(11.574));% (Pa) vapor pressure
at T and 1 atm in pascals
%L=@(T,Yp)
max(1,(.0839.*log(Yp)+1.7831)).*(1e3).*(1e3./mwl).*(50.43).*exp(0.4475.
*T./513.9).*(1-T./513.9).^(0.4989);%(J./kg) Latent Heat fit
L               =@(T,Xlf)
Xlf.*(1e3).*(1e3./mwl).*(50.43).*exp(0.4475.*T./513.9).*(1-
T./513.9).^(0.4989);%(J./kg) Latent Heat fit
```

```
lamf            =@(T) (4.1841e-7).*T.^2+(-1.6423e-4).*T+.026248;%
(W./mK) Thermal conductivity of fuel vapor
visf            =@(T) (1.4991e-7)+(3.0741e-8).*T+(-4.4479e-12).*T.^2;
%(Ns./m.^2) Viscosity of fuel vapor
visfl           =@(T) (1./1000).*exp((5.8942e-1)+(-2.254e-2).*(T-
273.15)+(1.0283e-4).*(T-273.15).^2+(-8.8574e-7).*(T-
273.15).^3+(4.7884e-9).*(T-273.15).^4+(-9.7493e-12).*(T-273.15).^5);%
Ns./m.^2 from Thermal Fluids ONline
epsilon         =(.58e-9); % approx size (max dimension) of liquid
molecule in m
lamff           =@(T) exp(-1.6976 + -1.2505e-3.*(T-273.15) + 7.59291e-
7.*(T-273.15).^2 +5.2361e-8.*(T-273.15).^3 + -3.4986e-10.*(T-273.15).^-
4 + 6.4599e-13.*(T-273.15).^5);%liquid conductivity W/mK
s               =@(T) 24.419 + -8.1477e-2.*(T-273.15) + -1.1450e-4.*(T-
273.15).^2 + 8.6540e-7.*(T-273.15).^3 + -7.6432e-9.*(T-273.15).^4 +
1.9148e-11.*(T-273.15).^5;% surface tension N/m

% Mixture Properties of Gas Phase
Sg              =1.5.*Tbg;
Sf              =1.5.*Tbf;
Sfg             =.73.*(Sg.*Sf).^.5;
Sgf             =Sfg;
Sff             =.73.*(Sf.*Sf).^.5;
Sgg             =.73.*(Sg.*Sg).^.5;
A               =@(T,vi,vj,Mi,Mj,Si,Sj,Sij)
.25.*(1+((((vi)./(vj)).*((Mj)./(Mi)).^.75 ...
                    .*((T+Si)./(T+Sj))).^(1./2)).^2).*(T+Sij)./(T+Si);
CpMix           =@(T,Ygf) Ygf.*Cpf(T)+(1-Ygf).*Cpg(T);     % linear
mixture rule
rhogMix         =@(T,Ygf) Ygf.*rhogf(T)+(1-Ygf).*rhog(T)'; % linear
mixture rule

% Gas Lambda Mix Calc Variables ----------------------------------------
---
Sg=1.5*Tbg;
Sf=1.5*Tbf;
Sfg=.73*(Sg*Sf)^.5;
Sgf=Sfg;
Sff=.73*(Sf*Sf)^.5;
Sgg=.73*(Sg*Sg)^.5;
A=@(T,vi,vj,Mi,Mj,Si,Sj,Sij)
.25*(1+((((vi)/(vj))*((Mj)/(Mi))^.75*((T+Si)/(T+Sj)))^(1/2))^2)*(T+Sij)
/(T+Si);

% NANO PROPERTIES
mwn             = 26.98;% (kg/kmol)
rhon            = 2700;%(kg/m^3)% bulk
Cpn             = (10^3)*0.91;%(J/kgK)
volfM           = pi()/6;% max volume fraction
%Rnano           =(1/2)*60e-9;% radius of avg nano particle in meters
%df              = 1.8;
%Rg              = (1/2)*250e-9;
ra              =((Rg/Rnano)^df*Rnano^2)^(1/2);% from Keblinski Surface
Area Equivilent radius
volf            =@(T,Y) (rhon*(1/Y-1)/rhol(T)+1)^(-1);
inVolf          = (Rg/Rnano)^(df-3);
```

```
MaxVolf          = volfM*inVolf

%rhon             = rhon*MaxVolf;% nano
rholMix          =@(T,Yn) ((1-Yn)./rhol(T)+Yn./rhon).^-1;
%%
% ****************************************
% Simulation
% ****************************************
numdr            = 500;
R                = Rs(1);
dr               = R./numdr;
r                = 0:dr:Rs(1);
J                = 0.0001*ones(loop,numdr);
RS               = R/cos(pi/2-Theta);
h_max            = -(sqrt(RS^2 - R^2)-RS);
h                = RS*cos(asin(r(1:end-1)./RS)) - (RS-h_max);
Cap              = (2*s(T0)/rhol(T0)/9.81)^0.5;
C                = rhon*V0.*ones(1,size(r,2)-1);% Concentration from
volume fraction
U                = 0.*C;

% Volume representation
%V(1)             = pi.*R^3.*Theta(1)./4;% Dunn et al. (Parabolic)
Beta             = (1-cos(Theta))^2*(2+cos(Theta));
V(1)             = (R/sin(Theta))^3*pi*(Beta)/3;% spherical from Erbil
review 2012
Qtp              = zeros(1,size(r,2)-1);
YpS = zeros(1,size(r,2)-1); % Mass Fraction of NANO
%Tls(1,:) = T0.*ones(1,size(r,2)-1);
count =0;
Yprev =YpS;
T                = T0.*ones(1,size(r,2)-1);
dsdT =-(s(T0+.1) - s(T0))/(T0+.1 -T0);
if V0==0
    tf = V/7.37e-12;
else
    tf=V/5.4e-12;
end
M0=mean(C)*V;

for i=2:loop
    % SAVE VALUES
    if i == 2
        count=count+1;
        disp(['saved at ' num2str(i*dt) ' seconds'] );
        loc(count)              = i;
        t(count)                = 0;
        Tlsave(count,:)         = T;
        Rsave(count,:)          = R;
        hsave(count,:)          = h;
        Vsave(count)            = V;
        V_dotsave(count)        = 0;
        Thetasave(count)        = Theta;
        Csave(count,:)          = C;
        Usave(count,:)          = U;
        VNsave(count)           = sum(C.*2*pi.*r(1:end-1).*dr.*h);
```

```
        Cmodsave(count,:)       = 1.*ones(1,size(r,2)-1);
    end

    % ***************************
    % Converge on a temperature
    % ***************************
    Tp                  = 0.99*T;
    Vp                  =  V;
    Thetap              = Theta;
    index_not_converged = ones(size(T,2),1);
    iloop               = 0;
    converge            = abs(T - Tp)./Tp;
    while converge(index_not_converged) > 0.0001 & iloop < 201
        V           = Vp;
        Theta       = Thetap;
        iloop        = iloop+1;

        % ***************************************
        % Evaporation Mass Flux, J (Dunn et al 2007)
        % ***************************************
        Volf            = C./rhon;% m^3 Al per m^3 solution
        %               |kg nal | |m3 nAl | |m3 nal |
        %               |-------|*|-------|=|-------|
        %               |m3 tot | |kg nAl | |m3 tot |

        Y               = Volf.*rhon./((1-Volf).*rhol(T) +
Volf.*rhon);% kg al per kg solution
        Y               = C./rholMix(T,Y);
        tY              = C.*rhon./((1-C).*rhol(T) + Y.*rhon);
        Xlf             = ((1-Y)./mwl)./(((1-Y)./mwl)+(Y./mwn));
% Convert from mass fraction to mole
        %Xlf             = ((1-YpS)./mwl)./(((1-YpS)./mwl)+(YpS./mwn));

        % Clausius
        Xgf             =
Xlf.*(101325/P).*exp((L(T,1)./(8.3144./mwl.*1000)).*(1./Tbf - 1./T));
% Clausius Clayperion, gas phase mole frac of evap
        % Vapor Pressure
        tXgf             = Xlf.*(Pv(T)/P);

        Xgg             = ((1-Yprev)./mwg)./((1-
Yprev)./mwg+Yprev./mwl);          % Gas phase mole fraction of nonevap
        if imag(Xgf) >0
            disp('Xgf is imag...');
            keyboard
        end
        Ygf        = Xgf.*mwl./(Xgf.*mwl+Xgg.*mwg);
% Mass fraction of fuel in vapor
        tYgf        = Xgf.*mwl./(Xgf.*mwl+(1-Xgf).*mwg);
% Mass fraction in vapor
        tCsat       = (rhogf(T).*(1./Ygf-1)./transpose(rhog(T))+1).^(-
1);                     % Concentration in vapor
        t2Csat        =Ygf.*(1./rhogMix(T,Ygf));
        t3Csat      = (1./(Xgf.*mwl + Xgg.*mwg)).*rhogMix(T,Ygf);
        Volfg       = (rhogf(T).*(1./Ygf-1)./rhog(T)'+1).^(-1);
```

```
        %Csat        =Volfg.*rhogf(T);
                     % |m3 eth | |kg eth | |kg eth |
                     % |-------|*|-------|=|-------|
                     % |m3 tot | |m3 eth | |m3 tot |

        Csat         =Ygf.*(rhogMix(T,Ygf));% units of kg/m^3
                     % |kg eth | |kg tot | |kg eth|
                     % |-------|*|-------|=|------|
                     % |kg tot | |m3 tot | |m3 tot|

        %Csat = 0.186 + 9.47*10^-3*(T-Tinf); % METHANOL VALUES
        %DgM =@(T) 1.5*10^(-5);              % METHANOL VALUES
        %L   =@(T,Y) 1155;                   % METHANOL VALUES

        %MASS FLUX FOR VARYING DEGREE CONTACT ANGLE
        J90          = Dg(T).*(1-Yinf).*Csat./R;% Units of Kg/m^2
        lambda      = 1/2 - Theta / pi;% found to be best match to FEM
results
        J0           = J90.*(0.27*Theta^2 + 1.30).*(0.6381 -
0.2239*(Theta - pi/4)^2);
        J           = J0.*(1-r(1:end-1).^2./R.^2).^-lambda;

%         loca       = find(1./(1-r(1:end-1).^2./R.^2).^lambda > 7 ,1);
%         J(loca:end)    = J(loca).*exp(7 - 1./(1-r(loca:end-
1).^2./R.^2).^lambda);

        %
**************************************************************
        if iloop > 100
            loca       = find(abs(T-Tp)./Tp > 0.01,1);
            J(loca:end)    = J(loca).*exp(1./(1-
r(loca).^2./R.^2).^lambda - 1./(1-r(loca:end-1).^2./R.^2).^lambda);
            %disp('iloop greater than 100')
            %sum(converge(index_not_converged))
        end
        %
**************************************************************

        % ****************************************
        % Volume change
        % ****************************************
        VolFlux      = 2*pi*r(1:end-1).*dr.*J./rhol(T);
        V_dot        = -2.*pi.*sum(J.*r(1:end-1).*dr./rhol(T));

        % ****************************************
        % Convert rate of change of volume to current volume
        % ****************************************
        V            = V + V_dot.*dt;

        if V < 0
            disp(['Complete evap at ' num2str(i*dt) ' seconds of
simulation time.'])
            count=count+1;
            disp(['saved at ' num2str(i*dt) ' seconds'] );
            loc(count)       = i;
```

```
            t(count)              = i*dt;
            Tlsave(count,:)       = T;
            Rsave(count,:)        = R;
            hsave(count,:)        = h;
            hh(count)             = int_h;
            Vsave(count)          = V;
            V_dotsave(count)      = V_dot;
            VolFlux_s(count,:)      = VolFlux;
            Jsave(count,:)        = J;
            Csatsave(count,:)     = Csat;
            Thetasave(count)      = Theta;
            MaranSave(count)      = Maran;
            Csave(count,:)        =C;
            Usave(count,:)        =U;
            return
        end

        int_theta   = Theta;
        conver      = .5;
        prev_theta  = int_theta*.5;
        prev_vol    = V*.5;
        while conver > 0.0001;
            RS          = R/cos(pi/2-int_theta);
            int_h       = -(sqrt(RS^2 - R^2)-RS);
            int_Vol     = (pi*int_h/6)*(3*R^2+int_h^2);
            int_theta_n = int_theta - (int_Vol - V)*(int_theta-
prev_theta)/(int_Vol-prev_vol);
            prev_vol    = int_Vol;
            prev_theta  = int_theta;
            int_theta   = int_theta_n;
            conver      = abs(int_Vol-V)/V;
        end

        % ****************************************
        % Evaluate Contact Angle, and height at new Volume
        % ****************************************
        Theta = int_theta;
        hprev = h;
        h = RS*cos(asin(r(1:end-1)./RS)) - (RS-int_h);
        delh = h - hprev;

        %****************************************
        % Esimate the local concentration as a result of Evap Flux J
        %****************************************
        C = C.*hprev./h;

        %****************************************
        % Estimate Velocity through change in volume
        %****************************************
        %dVh = (delh./dt);%.*2*pi*r(1:end-1).*dr;
        %dVJ = J./rhol(T);%.*2*pi*r(1:end-1).*dr;
        %U = dVJ - dVh;
        %keyboard
        for rloc = 2:size(r,2)-1
            U(rloc) = (-1/rhol(T(rloc))./r(rloc)./h(rloc)).* ...
```

```
                sum(r(1:rloc).*dr.*(J(1:rloc) +
rhol(T(1:rloc)).*delh(1:rloc)./dt));
        end
        tt=i*dt/tf;
        rt=r(1:end-1)./R;
        u=(1/4).*(1./(1-tt)).*1./rt.*((1-rt.^2).^-lambda - (1-rt.^2));
        U1=u.*R./tf;
        U=U1;
        U(1) = 0;
        loc=find(isnan(U));
        if ~isempty(loc)
            keyboard
            U(loc)=0;
        end

        % ****************************************
        % Modify Concentration due to convection
        % ****************************************
        Cprev=C;
        for rloc = 2:size(r,2)-1
            C(rloc) = (C(rloc-1)*U(rloc-1)*dt*2*pi*r(rloc-1)*h(rloc-1)
...
                - C(rloc)  *U(rloc)  *dt*2*pi*r(rloc)  *h(rloc)...
                + C(rloc)  *dr          *2*pi*r(rloc)  *h(rloc))...
                /(2*pi*r(rloc)*h(rloc)*dr);
        end
        Cmod = C./Cprev;
        if ~isempty(find(C<0))
            keyboard
        end

        % ******************************************************
        % Find if concentration is over max, then place inward
        %******************************************************
        locmax = find(C > MaxVolf*rhon );
        if ~isempty(locmax)
            for invi =1:size(r,2)-1
                rloc = size(r,2)-invi;
                if C(rloc)>MaxVolf*rhon
                    if rloc-1 == 0
                        disp(['Max volume @ center @' num2str(MaxVolf)
,' Volume fraction, and ' num2str(i*dt) ' seconds'  ])
                        V_dot = NaN;
                        V_return=V_dot;
                        return
                    end
                    %disp('Max Volf...')
                    Excess          = 2*pi*r(rloc)*dr*h(rloc)*(C(rloc)-
MaxVolf*rhon);
                    C(rloc)         = MaxVolf*rhon;
                    C(rloc-1)       = C(rloc-1) + Excess/(2*pi*r(rloc-
1)*dr*h(rloc-1));

                    totError=M0 - sum(C.*pi.*r(1:end-1).*2*dr.*h)...
                        /(sum(2*pi.*r(1:end-1).*h.*dr))*V;
```

```
                    C(rloc-1)       =C(rloc-1) + totError/(2*pi*r(rloc-
1)*dr*h(rloc-1));
                    if C(rloc-1)<0
                        disp('C less than 0...')
                        beep
                        keyboard
                    end
                end
            end
        end
        totError     =M0 - sum(C.*pi.*r(1:end-1).*2*dr.*h)...
                        /(sum(2*pi.*r(1:end-1).*h.*dr))*V;
        C(end)       =C(end) + totError/(2*pi*r(end-1)*dr*h(end-1));
        % ****************************************
        % Marangoni Calc
        % ****************************************
        Maran       = mean(dsdT.*L(T,1).*Dg(T).*(1-
Yinf).*Csat./Theta./s(T)./lamff(T));

        % ****************************************
        % Gravity Calc
        % ****************************************
        Grav = (9.81*(rhon-
rhol(T))/rhon)./(18/4*visfl(T).*(delh./dt)./ra^2);

        % ****************************************
        % Spatially varying Temp (Dunn et al. 2007)
        % ****************************************
        % Added Heat conduction/diffusive transport in Qair
        %                       Bm          = (Ygf-Yinf)./(1-Ygf);
        %                       Aff         =
A(T,visf(T),visf(T),mwl,mwl,Sf,Sf,Sff);
        %                       Agg         =
A(T,visg(T),visg(T),mwg,mwg,Sg,Sg,Sgg);
        %                       Agf         =
A(T,visg(T),visf(T),mwg,mwl,Sg,Sf,Sgf);
        %                       Afg         =
A(T,visf(T),visg(T),mwl,mwg,Sf,Sg,Sfg);
        %                       lamMix      =
Xgf.*lamf(T)./(Xgf.*Aff+Xgg.*Afg)+Xgg.*lamg(T)./(Xgf.*Agf+Xgg.*Agg);
        %                       rhoMix      = ((Ygf)./rhogf(T)+(1-
Ygf)./(rhog(T)')).^1;
        %                       Le          =
((rhoMix.*CpMix(T,Ygf).*Dg(T)./lamMix)).^-1;
        %                       CpgBAR      = (Cpg(T)+Cpg(Tinf))./2;% air
avg. specific heat
        %                       CpfBAR      =
(2.*Cpf(T)+Cpf(Tinf))./3;%avg. gas phase fuel specific heat
        %                       phi         = (CpfBAR./CpgBAR).*(1./Le);
        %                       Bt          = ((1+Bm).^phi)-1;
        %                   Qair        = rhol(T).*V_dot.*(
CpfBAR.*(Tinf-T)./Bt);
        %
        %                   Tn           = Tinf - ((L(T,1).*J + Qair).*(
h./lamff(T) + hs./lams));
```

```
        %Tgrad       = T0.*(1-r(1:end-1)/R*.01);
        %Tinf        = Tgrad;
        Tn           = Tinf - ((L(T,1).*J).*( h./lamff(T) + hs./lams));

        loca = find(r(:) > R/2 ,1);
        e=0:(h(loca))/100:h(loca);
        Tfuns        =@(z) Tinf - L(T(loca),1)*J(loca)/lams*(z+hs);
        Tfun         =@(z)  Tinf -
L(T(loca),1)*J(loca)*(z./lamff(T(loca)) + hs./lams);
        BL(1,:) = e;
        BL(2,:) = Tfun(e);
        BL(3,:) = -hs:hs/100:0;
        BL(4,:) = Tfuns(BL(3,:));

        Tinf_forced = ((L(T,1).*J).*( h./lamff(T) + hs./lams)) + 298;

        loc=find(isnan(Tn));
        loc1=find(Tn<0);
        if ~isempty(loc) | ~isempty(loc1) | ~isempty(find(Tn>Tinf))
            disp('Temperature acting all funky...')
            beep
            keyboard;
            loc=find(Tn>Tinf | isnan(Tn) | Tn<0)
            Tn(loc)=Tn(loc-1);
        end

        Ts           = Tinf - L(T,1).*J/lams*(hs);
        Qsurface     = lams*(Ts-Tinf)/hs;

        %*************************************************************
        % Neglect unsteady temperature via hu et al 2005 Appendix A
        % *************************************************************

        % look at Stanton Number/
        Tp           = T;
        T            = Tn;
        clear index_not_converged
        index_not_converged = find(abs(T - Tp)./Tp  > 0.0001);
        converge             = abs(T - Tp)./Tp;
    end

    % ****************************************
    % PREVIOUS VALUES
    % ****************************************
    Yprev = Ygf;

    % ****************************************
    % SAVE VALUES
    % ****************************************
    if mod((dt*i),interval) ==0 | i==10
        count=count+1;
        disp(['saved at ' num2str(i*dt) ' seconds'] );
        disp(['     mean (Csat) ' num2str(mean(Csat))]);
        disp(['     mean (tCsat) ' num2str(mean(tCsat))]);
        disp(['     mean (t2Csat) ' num2str(mean(t2Csat))]);
```

```
        disp(['     mean (t3Csat) ' num2str(mean(t3Csat))]);
        disp(['     mean (Ygf)  ' num2str(mean(Ygf))]);
        disp(['     mean (tYgf)  ' num2str(mean(tYgf))]);
        disp(['     mean (Xgf)  ' num2str(mean(Xgf))]);
        disp(['     mean (tXgf)  ' num2str(mean(tXgf))]);
        disp(['     mean (Xlf)  ' num2str(mean(Xlf))]);
        disp(['     Partial pressure = Xgf (using Pv) : '
num2str(mean(Pv(T)./P))])
        loc(count)            = i;
        t(count)              = i*dt;
        Tlsave(count,:)       = T;
        Tinf_forcesave(count,:) =Tinf_forced;
        Rsave(count,:)        = R;
        hsave(count,:)        = h;
        hh(count)             = int_h;
        Vsave(count)          = V;
        V_dotsave(count)      = V_dot;
        VolFlux_s(count,:)    = VolFlux;
        Jsave(count,:)        = J;
        Csatsave(count,:)     = Csat;
        Ygfsave(count,:)      = Ygf;
        Thetasave(count)      = Theta;
        MaranSave(count)      = Maran;
        GravSave(count,:)     = Grav;
        Csave(count,:)        = C;
        Xlfsave(count,:)      = Xlf;
        Ysave(count,:)        = Y;
        Usave(count,:)        = U;
            U1save(count,:)   = U1;
        VNsave(count)         = sum(C.*2*pi.*r(1:end-1).*dr.*h);
        Cmodsave(count,:)     = Cmod;
        BL_s(count,:,:)       = BL;
    end
end
%disp(['Vdot of ' num2str(V_dot) ' @ ' num2str(numdr) ' Radial Steps,
and dt of ' num2str(dt) ' Seconds'])

figplot=1;
if figplot==1
    % ****************************************
    % Plotting
    % ****************************************
    loc = 1:count;
    figure(4)
    plot(t(loc),Thetasave(loc)*180/pi)
    hold on;
    plot(BF_t,BF_theta,'r');
    plot(NF_t,NF_theta,'g');
    grid on
    xlabel('Seconds');
    ylabel('Contact Angle (Degrees)')
    p=polyfit(t(loc),Thetasave(loc)*180/pi,1);
    plot(t(loc),polyval(p,t(loc)),'-.k','LineWidth',2);

    figure(9)
    plot(BF_t,BF_dtdt,'r');
```

```
    hold on
    plot(NF_t,NF_dtdt,'g');
    dtdt=gradient(180/pi*Thetasave(loc),t(loc));
    plot(t(loc),dtdt)
    p=polyfit(BF_t,BF_dtdt,1);
    plot(BF_t,polyval(p,BF_t),'-.k','LineWidth',2);
    xlabel('Seconds');
    ylabel('Contact Angle Rate (Degrees/sec)')

    h=figure(5)
    plot(t(loc),Vsave(loc),'LineWidth',2)
    hold on;
    plot(BF_t,BF_volume,'r')
    plot(NF_t,NF_volume,'g')
    pb=polyfit(BF_t,BF_volume,1);
    plot(BF_t,polyval(pb,BF_t),'r-.','LineWidth',2)
    pn=polyfit(NF_t,NF_volume,1);
    plot(NF_t,polyval(pn,NF_t),'g-.','LineWidth',2)
    grid on
    xlabel('Seconds');
    ylabel('Volume (m^3)')
    p=polyfit(t(loc),Vsave(loc),1);
    plot(t(loc),polyval(p,t(loc)),'-.k','LineWidth',2);
    legend(['Simulation, fit :' num2str(mean(V_dotsave(2:end)))],['Base
Fluid, fit: ' num2str(pb(1)) ],['Nanofluid, fit: ' num2str(pn(1))]);
    print(h,'-dpdf',['fractal_dimension' num2str(df) '_VolF'
num2str(V0) num2str(Rg) '_Evap_Rate.pdf']);


    figure(6)
    plot(t(loc),abs(V_dotsave(loc))/(1e-12),'o')
    hold on;
    p=polyfit(t(loc),abs(V_dotsave(loc))/(1e-12),1);
    %plot(BF_t,polyval(p,BF_t),'-.','LineWidth',2);
    plot(BF_t,-BF_dmdt/790/(1e-12),'r');
    plot(NF_t,-NF_dmdt/790/(1e-12),'g');
    p=polyfit(BF_t,-BF_dmdt/790/(1e-12),1);
    plot(polyval(p,BF_t),'-.r','LineWidth',2);
    p=polyfit(NF_t,-NF_dmdt/790/(1e-12),1);
    plot(polyval(p,NF_t),'-.g','LineWidth',2);
    %plot(-gradient(polyval(pv,BF_t))/(1e-12),'g-.','LineWidth',2)
    grid on
    grid
    xlabel('Seconds');
    ylabel('- Volume rate of change (nl/s)')
    grid
    disp(['Vdot of ' num2str(V_dot) ' @ ' num2str(numdr) ' Radial
Steps, and dt of ' num2str(dt) ' Seconds'])
    disp(['     Vdot mean of ' num2str(mean(V_dotsave(2:end)))]);

    figure(7)
    plot(R,abs(V_dotsave(loc))/(1e-12),'o')
    hold on;
    plot(R,-BF_dmdt/790/(1e-12),'r+')
    plot(R,-NF_dmdt/790/(1e-12),'g+')
    grid on
```

```
    grid
    xlabel('Radius (m)');
    ylabel('- Volume rate of change (nl/s)')
    axis([0 2e-3 0 30 ])
    grid

    figure(10)
    plot(R,(abs(dtdt(2))./(abs(V_dotsave(loc(2)))/(1e-12))),'o');
    grid on
    hold on
    xlabel('Initial radius')
    ylabel('Rate of change of theta  over rate of change of volume')

    figure(11)
    plot(R,(abs(dtdt(2))),'o');
    grid on
    hold on
    plot(R,BF_dtdt,'r+')
    xlabel('Initial radius')
    ylabel('Rate of change of theta')

    figure(8)
    plot(t(loc),MaranSave(loc))
    hold on
    plot([t(loc(1)) t(loc(end))],[0.05 0.05],'r');
    axis([0 i*dt 0 1])
    grid
    xlabel('Seconds');
    ylabel('Instantanious Marangoni number')
    toc

    figure(20)
    plot(t(loc),VNsave(loc))
    hold on;
    plot(0,M0,'ro');
    grid on

    for j =1:count
        c='';
%         if j==1
%             c='r';
%         elseif j==size(loc,2)
%             c='k';
%         end
if V0==0
    c='';
else
    c='r';
end
        figure(1)
        plot(r(1:end-1),hsave(j,:),c)
        hold on

        figure(2)
        plot(r(1:end-1),Tlsave(j,:)-273.15,c)
        hold on
```

```
        figure(3)
        plot(r(1:end-1)./R,Jsave(j,:)*1000,c)
        hold on;

        figure(12)
        plot(r(1:end-1)./R,VolFlux_s(j,:),c);
        hold on;

        figure(13)
        plot(r(1:end-1)./R,Csave(j,:)/rhon,c);
        hold on;

        figure(17)
        plot(r(1:end-1)./R,Ysave(j,:),c);
        hold on;

%         figure(18)
%         plot(r(1:end-1)./R,GravSave(j,:),c);
%         hold on;

        %     figure(14)
        %     plot(r(1:end-1)./R,Usave(j,:),c);
        %     hold on;
        %
        figure(15)
        plot(r(1:end-1)./R,Csatsave(j,:),c);
        hold on;

        figure(19)
        plot(r(1:end-1)./R,Ygfsave(j,:),c);
        hold on;

        figure(21)
        plot(r(1:end-1)./R,Tinf_forcesave(j,:),c);
        hold on;
        grid on

        figure(16)
        plot(r(1:end-1)./R,Usave(j,:),c)
        hold on;
        plot(r(1:end-1)./R,U1save(j,:),[''])

        if j>1
            figure(22)
            plot(squeeze(BL_s(loc(j),2,:))-
273.15,squeeze(BL_s(loc(j),1,:)),'LineWidth',2);
            hold on
            plot(squeeze(BL_s(loc(j),4,:))-
273.15,squeeze(BL_s(loc(j),3,:)),'LineWidth',2);
            grid on
            hold on
            xlabel('Temperature');
            ylabel(['Height @ Half Radius and time '
num2str(t(loc(j)))])
```

```
        end

    end
    h=figure(1)
    xlabel('Radius (m)');
    ylabel('Height (m)');
    title(['Evaporation every ' num2str(interval) ' seconds'])
    str(1)={['Fractal Dimension: ' num2str(df)]};
    str(2)={['Nanoparticle Radius: ' num2str(Rnano) ' m']};
    str(3)={['Agglomerate Radius: ' num2str(Rg) ' m']};
    str(4)={['Initial volume fraction: ' num2str(V0)]};
    str(5)={['Maximum Volume fraction: ' num2str(MaxVolf)]};
    text(0.1e-3,0.1*hsave(1,1),str,'BackgroundColor','w');
    grid on
    axis equal
    print(h,'-dpdf',['fractal_dimension' num2str(df) '_VolF'
num2str(V0) num2str(Rg) '_height.pdf']);

    h=figure(2)
    xlabel('Radius (m)');
    ylabel('Temperature (Celcius)');
    title(['Evaporation every ' num2str(interval) ' seconds'])
    text(.1e-3,Tlsave(end,1)-273.15+2,str,'BackgroundColor','w');
    grid on
print(h,'-dpdf',['fractal_dimension' num2str(df) '_VolF' num2str(V0)
num2str(Rg) '_Temp.pdf']);

    h=figure(3)
    xlabel('Normalized Radius (n.d.)');
    ylabel('Evaporative Flux J (g/m^2s)');
    title(['Evaporation every ' num2str(interval) ' seconds'])
    text(0.1,3,str,'BackgroundColor','w');
    grid on
print(h,'-dpdf',['fractal_dimension' num2str(df) '_VolF' num2str(V0)
num2str(Rg) '_Flux.pdf']);

    figure(12)
    xlabel('Normalized Radius (n.d.)');
    ylabel('m^3/s Evaporation Volume per time at each radial station')
    grid on

   h= figure(13)
    hold on
    plot(r(1:end-1)/R,ones(size(r(1:end-1)/R))*MaxVolf,'r-.');
    xlabel('Normalized Radius (n.d.)');
    ylabel('Local Volume Fraction')
    text(0.1,.16,str,'BackgroundColor','w');
    grid on
print(h,'-dpdf',['fractal_dimension' num2str(df) '_VolF' num2str(V0)
num2str(Rg) '_VolFrac.pdf']);

    h=figure(17)
    xlabel('Normalized Radius (n.d.)');
    ylabel('Local Mass Fraction')
    text(0.1,.16,str,'BackgroundColor','w');
    grid on
```

```
print(h,'-dpdf',['fractal_dimension' num2str(df) '_VolF' num2str(V0)
num2str(Rg) '_MassFrac.pdf']);

    h=figure(16)
    title(['Evaporation every ' num2str(interval) ' seconds'])
    xlabel('Normalized Radius (n.d.)');
    ylabel('Local Radial Velocity')
    text(0.1,0.8e-5,str,'BackgroundColor','w');
    grid on
    print(h,'-dpdf',['fractal_dimension' num2str(df) '_VolF'
num2str(V0) num2str(Rg) '_Vel.pdf']);

    figure(15)
    xlabel('Normalized Radius (n.d.)');
    ylabel('Saturation Concentration')
    grid on

    figure(19)
    xlabel('Normalized Radius (n.d.)');
    ylabel(' Gas Phase Fuel mass fraction at the surface (Ygf)')
    grid on
end
V_dotsave
base_fluid= pb(1);
nano_fluid = pn(1);
V_return = mean(V_dotsave(2:end));
%V_return =  V_dot;
p_test=polyfit(t,Vsave,1);
disp([' Slope fit of : ' num2str(p_test(1))]);

save(['sublayer_' stringSublayer '_fractal_dimension' num2str(df)
'_VolF' num2str(V0) num2str(Rg) '_MassFrac.mat']);
return
end
```

Tot_Exp_for pub.m

```
% Test Space
% Will Gerken
% Nanofluid Drop Evaporation

clear all;
%close all;

% Data points of interest
points=...
        [0,         0.001,  0.005,  0.01,   0.02,   0.03];
eResults= ...
        [0.0063,    0.0062, 0.0058, 0.0056, 0.0055, 0.0054];
points_Eval=[0.001,0.005,0.01,0.02,0.03];%...
      %[0  ,0.0005, 0.001,0.0015,0.0025, 0.005, 0.01,0.015, 0.02,0.025
, 0.03];%[0.001,0.005,0.01,  0.03];
 %points_Eval=...
 %       [0, 0.001, 0.005, 0.01, 0.02 , 0.03];%[0.001,0.005,0.01,
0.03];
%Experimental Conditions
T0=282;
Tinf=296.75;
time=30;
df=1.7;
Rg=250e-9;
%NL=[50, 200, 400, 1000, 2000];
NL=[200];%[2000];
YmaxGiven=0.115;

% df=[1.6,1.65,1.7,1.75,1.8];
% Rg=[170,190,200,210,220,250]*10^-9;
% Ran and found : volf with best fit ~ 0.0374-0.038
% 220 nm, 1.8
% 190 nm, 1.7
% 200 nm, 1.75
df=[1.7];
Rg=[250/2]*10^-9;%[220,190,200,250]*10^-9;
%Rg=[250]*10^-9;


%Load
%load('FDMLD.mat');
%load('FD_STOCK.mat');
%load('PM.mat');
%load('PM.mat');
clock
t1=clock;

for iNL=1:size(NL,2)
    for i=1:size(points_Eval,2)

        %     dfs=1.7;
        %     Rgs=250e-9;
        %     disp('%%%%1%%%%')
```

```
        %
Results_SEM(i,:)=[evalFDMDL_PLAY_v2(points_Eval(i),T0,Tinf,dfs,Rgs,NL,t
ime),points_Eval(i)];
        idf=1;
        %for idf=1:size(df,2)
        if points_Eval(i)==0 && iNL==1

Evap(i,idf,iNL)=evalFDMDL_SurWeight(points_Eval(i),T0,Tinf,df(idf),Rg(i
df),NL(iNL),time);
        elseif points_Eval(i)==0
            Evap(i,idf,iNL)=Evap(i,idf,1);
        else

Evap(i,idf,iNL)=evalFDMDL_SurWeight(points_Eval(i),T0,Tinf,df(idf),Rg(i
df),NL(iNL),time);
        end
        save('Evap_Results_NL200','Evap','points_Eval','df','Rg')
    end
end

iRg=1;

for k=1:size(df,2)
    for l=1:size(Rg,2)

        Rnano=(1/2)*50e-9;
        inVolf=(Rg(l)/Rnano)^(df(k)-3);
        vol=pi()/6;%
        volfM(k,l)=vol*inVolf;
    end
end

h=figure(7);
c={'';'-.';'.-';'+-';'>-'};
for iNL=1:size(NL,2)
plot(points_Eval,-Evap(:,idf,iNL),c{iNL});
hold on;
grid on;
end
plot(points, eResults,'r<-');
%legend( ['Rg= ' num2str(Rg(idf)) ' df = ' num2str(df(idf)) ' NL = '
num2str(NL(1))],'Exp');%...
  %  ['Rg= ' num2str(Rg(idf)) ' df = ' num2str(df(idf)) ' NL = '
num2str(NL(2))],...
  %  ['Rg= ' num2str(Rg(idf)) ' df = ' num2str(df(idf)) ' NL = '
num2str(NL(3))],...
  %  ['Rg= ' num2str(Rg(idf)) ' df = ' num2str(df(idf)) ' NL = '
num2str(NL(4))],...
  %  ['Rg= ' num2str(Rg(idf)) ' df = ' num2str(df(idf)) ' NL = '
num2str(NL(5))],
%print(h,'-dpdf',['tot_exp_' num2str(df(idf)) '_df_' num2str(Rg(iRg))
'_mVol_' num2str(volfM(idf,iRg)) '_NL_' num2str(NL) '.pdf'])
clock
t2=clock;
disp('Eval time (hrs):')
etime(t1,t2)/60/60
```

evalFDMDL_SurWeight.m

```
function
[slope,volfM]=evalFDMDL_SurWeight(massFrac,T0,Tinf,df,Rg,NL,seconds)

%Coding Constants-------------------------------------------------------
----
%seconds=30;
dt=.0001;%seconds
loop=seconds/dt;

% System Variables-------------------------------------------------------
---
P=101325;%Pa
%T0=282;%K
%Tinf=296;%K
Yp0=massFrac;% initial mass fraction nano
gamma=0;% 0-1 activity factor for shell formation (1- All Shell, 0- No
Shell)
Yp=Yp0;
Yprev=0;
YpS=Yp;%surface mass fraction nano
%NL=200;
%Yfd(:,:)=zeros(loop,NL);
%Yfd(1,:)=Yp;
Yfinf=0;% mass fraction fuel at inf
Ts(1)=T0;%(K)
Rs(1)=sqrt(.011)/100/2;%(m)
Mshell(1)=0;
Kb=1.3806503e-23;% Boltzman Constant Kg m^2/s/K
drp=Rs(1)/NL;
NMass=zeros(1,loop);
flag=0;% steady state calc variable
Tprev=T0;
th(1)=0; % thickness of particle sublayer
mdot(1)=0;
tFull=-1;
Ysw_s(1)=massFrac;



% Gas Parameters (AIR)---------------------------------------------------
---
mwg=28.97;
Tbg=77.36;
Cpg=@(T) (10^3)*1.005; %J/kgK
X=[150;200;250;300;350];
Y=[2.3364;1.7458;1.3947;1.1614;.995];
rhog=fit(X,Y,'cubicspline');%kg/m^3
lamg=@(T) (7.071e-5)*(T-273.15)+2.428e-2;%W/mK
visg=@(T) (4.7225e-8)*(T-273.15)+(1.7238e-5); %(Ns/m^2)  viscosity of
air

% Liquid Properties (Ethanol) From Jiang Thesis--------------------------
---
mwl=46.07;% (kg/kmol)
```

```
Tbf=351.8;% in K
Dg=@(T) (1/100)*(1/100)*(((10^-
3)*T^1.75*(1/mwg+1/mwl)^(1/2))/((P/101325)*(50.36^(1/3)+20.1^(1/3))^2))
; %(m^2/s) Diffusion of liquid into air from FSG
Cpf=@(T) (1e3/mwl)*((-8.28925e-5)*T^2+(0.216104)*T+8.28126);%(J/kgK)
Specific heat of liquid in vapor phase
Cpl=@(T) (1e3/mwl)*(98.39+0.5368*(T-273.15));%(J/kgK) Specific heat of
liquid as liquid
rhol=@(T) (-0.8544)*(T-273.15)+806.43;% (kg/m^3)
rhogf=@(T) exp((-3.3681)+(5.2492e-2)*(T-273.15)+(5.1630e-5)*(T-
273.15)^2+(-1.9542e-6)*(T-273.15)^3+(8.6893e-9)*(T-273.15)^4+(-1.1451e-
11)*(T-273.15)^5); % from THermal FLuids Online....
Pv=@(T) (133.3224)*((4.0325e-4)*(T-273.15)^3+(2.7952e-2)*(T-
273.15)^2+(0.81796)*(T-273.15)+(11.574));% (Pa) vapor pressure at T and
1 atm in pascals
%L=@(T,Yp)
max(1,(.0839*log(Yp)+1.7831))*(1e3)*(1e3/mwl)*(50.43)*exp(0.4475*T/513.
9)*(1-T/513.9)^(0.4989);%(J/kg) Latent Heat fit
L=@(T,Xlf) Xlf*(1e3)*(1e3/mwl)*(50.43)*exp(0.4475*T/513.9)*(1-
T/513.9)^(0.4989);%(J/kg) Latent Heat fit
lamf=@(T) (4.1841e-7)*T^2+(-1.6423e-4)*T+.026248;% (W/mK) Thermal
conductivity of fuel vapor
visf=@(T) (1.4991e-7)+(3.0741e-8)*T+(-4.4479e-12)*T^2; %(Ns/m^2)
Viscosity of fuel vapor
visfl=@(T) (1/1000)*exp((5.8942e-1)+(-2.254e-2)*(T-273.15)+(1.0283e-
4)*(T-273.15)^2+(-8.8574e-7)*(T-273.15)^3+(4.7884e-9)*(T-273.15)^4+(-
9.7493e-12)*(T-273.15)^5);% Ns/m^2 from Thermal Fluids ONline
epsilon=(.58e-9); % approx size (max dimension) of liquid molecule in m

%     % NL Calc
%     Evap_L=0.001*22.8/Pv(T0);
%     n_evap=3;
%     NL=round(Rs(1)/(Evap_L/n_evap));
Yfd(:,:)=zeros(loop,NL);
Yfd(1,:)=Yp;

% Nano Parameters ------------------------------------------------------
---
mwn=26.98;% (g/mol)
rhon=2700;%(kg/m^3)
Cpn=(10^3)*0.91;%(J/kgK)
volfM=pi()/6;% max volume fraction
%Rg=;%250e-9;% radius of agglomerate in meters
Rnano=(1/2)*50e-9;% radius of avg nano particle in meters
%df=2.3;%1.70;% fractal dimension
rhofdp=zeros(1,NL);
rhofdp(:)=(Yfd(1)/rhon+(1-Yfd(1))/rhol(T0))^(-1);
ra=((Rg/Rnano)^df*Rnano^2)^(1/2);% from Keblinski Surface Area
Equivilent radius
volf=@(T,Y) (rhon*(1/Y-1)/rhol(T)+1)^(-1);
inVolf=(Rg/Rnano)^(df-3);
volfM=volfM*inVolf
Ymax=@(T) (volfM*rhon)/((volfM*rhon)+((1-volfM)*rhol(T)));
Ymax(T0)
```

```
% Gas Lambda Mix Calc Variables ----------------------------------------
---
Sg=1.5*Tbg;
Sf=1.5*Tbf;
Sfg=.73*(Sg*Sf)^.5;
Sgf=Sfg;
Sff=.73*(Sf*Sf)^.5;
Sgg=.73*(Sg*Sg)^.5;
A=@(T,vi,vj,Mi,Mj,Si,Sj,Sij)
.25*(1+((((vi)/(vj))*((Mj)/(Mi))^.75*((T+Si)/(T+Sj)))^(1/2))^2)*(T+Sij)
/(T+Si);

% Gas Cp Mix Calc -------------------------------------------------------
---
CpMix=@(T,Ygf) Ygf*Cpf(T)+(1-Ygf)*Cpg(T);

% Liquid-Nano Mixture Properties ----------------------------------------
---
rholMix=@(T,Yn) ((1-Yn)/rhol(T)+Yn/rhon)^-1;
CplMix=@(T,Yn) (Yn*Cpn+(1-Yn)*Cpl(T));
Mn0=Yp*(4/3)*pi()*Rs(1)^3*rholMix(T0,Yp0);%(Kg) Initial mass of
aluminum
Mn(1)=Mn0;
volf0=volf(T0,Yp0);%Yp*rholMix(T0,Yp0)/rhon;

if Yp0>Ymax(T0)
    disp('Initial Mass Fraction Greater than Max allowed')
    slope=NaN;
    return
end



tic
% Solve
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
for i=2:loop

    % Temporary Variables---------------------------------------------
---
    R=Rs(i-1);
    T=Ts(i-1);
%
%     % NL Calc
%     Evap_L=0.001*22.8/Pv(T);
%     n_evap=5;
%     NL=round(R/(Evap_L/n_evap));

    % Permiability Study Stuff
    mdot_n(i)=mdot/(1-YpS)-mdot;
    vdot_n(i)=mdot_n(i)/rhol(T)/volfM;
    %vdot_n_tot(i)=sum(vdot_n);% total reduction of volume at
i(integral)
    Requiv=R-th(i-1);
    del_th(i)=Requiv-(-3*dt*vdot_n(i)/4/pi()+Requiv^3)^(1/3);
```

```
th(i)=th(i-1)+del_th(i);
if Yp0==0
    th(i)=0;
end

delta_p(i)=2*22.8*0.001/R;%0.001 from conversion to N/m from mN/m

% Porosity Model (Carman Kozeny via Wiki)
pf=5;% packing factor
k=(1-volfM)^3*(2*Rg)^2/36/pf/(volfM)^2; % initial fractal limit
%k=(1-pi()/6)^3*(2*Rg)^2/36/pf/(pi()/6)^2; % Perfect particle limit

%k=.05e-15;
perm_Vel=k*delta_p(i)/visfl(T)/th(i);
k_req(i)=mdot/rhol(T)/(4)/pi()/R^2/delta_p(i)*visfl(T)*th(i);
perm_Vel_s(i)=perm_Vel;
m_Vel_s(i)=mdot/rhol(T)/(4*pi()*R^2);
mdot_p(i)=rhol(T)*(4)*pi()*R^2*perm_Vel;
mdot_p_req(i)=rhol(T)*(4)*pi()*R^2*perm_Vel*k_req(i)/k;


% Print Statment
if i-2 == 1 | i-loop/2==0 | i==loop-1
   disp('-----Output-----');
   disp(['For: ' num2str(Yp0) ' Initial Mass fraction']);
   disp('Time: ')
   i*dt
   disp('Max Mass Fraction:')
   Ymax(T)
   disp('Surface Mass Fraction:')
   YpS
%        disp('Molar Volume :')
%        Vm
%        disp('Kelvin Factor:')
%        KelvinFactor
%        disp('Nominal Latent Heat:')
%        L(T,1)
%        disp('Effective Latent Heat:')
%        Leff
%        disp('Effective Test Latent Heat:')
%        Leff_t
%        disp('Test Kelvin: ')
%        KelvinFactor_test
%        disp('Test Xfg');
%        Xgf_test
       disp('%%%%%%%%%%%%%%%%%%%%')
   disp(' Xfg');
   Xgf
   disp(' Xfg From Gibbs');
   %Xgf_Gibbs
   disp('Ratio (Gibbs/Nominal): ')
   %Xgf_Gibbs/Xgf
   disp('%%%%%%%%%%%%%%%%%%%%')
%        disp(' NNp');
%        Nnp
%       disp(' MASS_kg');
```

```
%        MASS_kg
%        disp('Radial Step:');
%        dr
%        disp('NL :')
%        NL
%        disp('Evap Length: ')
%        Evap_L
%keyboard
    end

    % Calc Mole, Mass Fraction, and Transfer number of Evap -----------
---
    %YpS=Yp;% Use Average Mass fraction from entire drop
    %Yps=Yfd(i-1,NL);% Use surface concentration from last timestep
    YpS=Ysw_s(i-1);


    Xlf=((1-YpS)/mwl)/(((1-YpS)/mwl)+(YpS/mwn));
    Xlfs(i)=Xlf;
    %Xgf=Xlf*Pv(T)/P; % From Data
    %Xgf=Xlf*(1/(P/101325))*exp(L(T,1)/(1000/mwl*8.3144*Tbf))*exp(-
L(T,Xlf)/(1000/mwl*8.3144*T));
    %Xgf=Xlf*exp((L(T,Xlf)/(8.3144/mwl*1000))*(1/Tbf - 1/T));

    ffactor=5;
    %(Xlf*(mwl)+(1-Xlf)*mwn)
    %rholMix(T,YpS)
    Vm=(Xlf*(mwl)+(1-Xlf)*mwn)/rholMix(T,YpS);% Volume per mole

    %0.022*Vm*2/Rnano

    MASS_kg=((4/3)*pi()*R^3)*rholMix(T,YpS);%kg
    NMoles=(MASS_kg*1000)/((mwl+mwn)/2);


    Nnp=Mn(i-1)/rhon/(4/3*pi()*Rg^3);
    SAnp=Nnp*(4*pi()*Rg^2);

    Vm=Vm*volf(T,YpS);%SAnp*epsilon*rhol(T)*1000*(1/mwl);% Volume
    Vm_s(i)=Vm;
    %L(T,(1+(1-Xlf)*((mwn*Cpn)/(mwl*Cpl(T)))));

    % GOOD AGREEMENT
    %KelvinFactor= 0.022*20*Vm*2/Rg;   % ~1000
    %Leff=KelvinFactor+L(T,(1+(1-Xlf)*((mwn*Cpn)/(mwl*Cpl(T)))));
    Leff=L(T,1);
    Xgf=Xlf*exp((Leff/(8.3144/mwl*1000))*(1/Tbf - 1/T));

%      KelvinFactor_test= 0.022*Vm*2/Rg;
%      Leff_t=KelvinFactor_test+L(T,1);
%      Xgf_test=Xlf*exp((Leff_t/(8.3144/mwl*1000))*(1/Tbf - 1/T));
%
%     %%%%%%%%%% Gibbs free energy approach %%%%%%%%%%
%     %R = 8.3144 J/K/mol
%     Adhesion=(1+cosd(3))*22.8*0.001*(2/Rnano)*volf(T,YpS);
```

```
%      %Adhesion = 0;
%
%      Tavg=(2*T+Tinf)/3;
%      Thermal = rhol(T)*L(T,1)*(T-Tinf)/Tavg;
%      %Thermal = rhol(T)*L(T,1)*(T-Tbf)/Tbf;
%      Thermal_s(i)=Thermal;
%
%      Mix = rhol(T)*8.3144*1000*T/mwl*log(1/Xlf);
%      %Mix = 0;
%
%      (-Adhesion+Mix)/(-Adhesion+Thermal+Mix);
%      TotMix=(Adhesion+Thermal+Mix);
%      TotMix_s(i)=TotMix/(rholMix(T,YpS)*8.3144/mwl*1000*T);
%      Xgf_Gibbs=Xlf*exp((TotMix)/(rholMix(T,YpS)*8.3144/mwl*1000*T));

    %Xgf=Xgf_Gibbs;

    % Current Theory  : value of Thermal is not negative and large
enough
    % at high initial loadings to provide reduction desired.  This is
    % thought to be due to the error in changing the Temperature terms
in
    % THERMAL.  In order to see if this is correct, Thermal will be
saved
    % and plotted, with Mix and Adhesion.


%keyboard

    %Xgf=exp(((L(T,(1+(1-
Xlf)*((mwn*Cpn)/(mwl*Cpl(T)))))+KelvinFactor)/(8.3144/mwl*1000))*(1/Tbf
- 1/T));
    %Xgf=exp((L(T,1+1-Xlf)/(8.3144/mwl*1000))*(1/Tbf - 1/T));

    Xgg=((1-Yprev)/mwg)/((1-Yprev)/mwg+Yprev/mwl);
    Ygf=Xgf*mwl/(Xgf*mwl+Xgg*mwg);
    Yprev=Ygf;
    Bm=(Ygf-Yfinf)/(1-Ygf);
    %Bms(i)=Bm;

    %Lamda Calc ----------------------------------------------------
---
    Aff=A(T,visf(T),visf(T),mwl,mwl,Sf,Sf,Sff);
    Agg=A(T,visg(T),visg(T),mwg,mwg,Sg,Sg,Sgg);
    Agf=A(T,visg(T),visf(T),mwg,mwl,Sg,Sf,Sgf);
    Afg=A(T,visf(T),visg(T),mwl,mwg,Sf,Sg,Sfg);
    lamMix=Xgf*lamf(T)/(Xgf*Aff+Xgg*Afg)+Xgg*lamg(T)/(Xgf*Agf+Xgg*Agg);
    %lam(i)=lamMix;

    % rho mix Calc ---------------------------------------------------
---
    rhoMix=((Ygf)/rhogf(T)+(1-Ygf)/(rhog(T)))^1;

    % Avg Cp Values for Phi Calculation ------------------------------
---
```

```
    CpgBAR=(Cpg(T)+Cpg(Tinf))/2;% air avg. specific heat
    CpfBAR=(2*Cpf(T)+Cpf(Tinf))/3;%avg. gas phase fuel specific heat
    %Cpfbs(i)=CpfBAR;
    %Cpfs(i)=Cpf(T);

    %Lewis Number Calc based on Mix properties ----------------------
---
    Le=((rhoMix*CpMix(T,Ygf)*Dg(T)/lamMix))^-1;

    % Phi Calc based off avg Cp values and Lewis Number --------------
---
    phi=(CpfBAR/CpgBAR)*(1/Le);
    %phis(i)=phi;

    % Calc of evap rate using const T Evap ---------------------------
---
    mdot=4*pi()*R*rhog((T+2*Tinf)/3)*Dg((T+2*Tinf)/3)*log(1+Bm);
    mdots(i)=mdot;

    % Calculate Evap Rate K of ideal droplet -------------------------
----
    K=8*mdot/rhol(T)/pi()/R;

    % Calc of new Radius based off Mdot ------------------------------
---
    %R=(((4/3)*pi()*R^3*rholMix(T,Yp)-
mdot*dt)/((4/3)*pi()*rholMix(T,Yp)))^(1/3);
    %volf=1/(rhon*(1/Yp-1)/rhol+1);%Yp*rholMix(T,Yp)/rhon;
    R=((3/4/pi())*((4/3)*pi()*R^3-mdot*dt/rholMix(T,YpS)/(1-
volf(T,YpS)*gamma)))^(1/3);%Nano Shell formulation
    Rs(i)=R;

    % Calculate new Yp based on shell formation ----------------------
---
    MnEvap=rhon*(4/3)*pi()*(Rs(i-1)^3-R^3)*volf(T,Yp)*gamma;% shell
mass
    Yp=(Mn(i-1)-MnEvap)/((4/3)*pi*Rs(i-1)^3*rholMix(T,Yp)-mdot*dt-
MnEvap);
    %Ypsaved(i)=Yp;
    Mn(i)=Mn(i-1)-MnEvap;
    Mshell(i)=Mshell(i-1)+MnEvap;

    % Calculate Diffusion Rate of Nanoparticles ----------------------
---
    D=Kb*T/6/pi()/visfl(T)/ra;%(m^2/s) Einstien - Stokes Diffusion
    D=D*10^-4;
    % Calculate Max Mass Fraction Based off max Vol Fraction----------
---
    %Ymax(i)=;%volfM*rholMix(T,Yfd(i-1,NL))/rhon;
    Ycompare(i)=Yfd(i-1,NL)/Ymax(T);

    % Finite Difference Calc of Yp -----------------------------------
---
```

```
    %Calculate estimate of YpS here????-------------------------------
-------------------------------------------------------------------------
---------------------------

    Yavg1=zeros(1,NL);
    Yavg=0;
    dr=R/NL;
    dr2dt=0;%((dr*1)^2-(drp*1)^2)/(dt);
    Yfd(i,1)=Yfd(i-1,2)+(dt/1/dr)*(D/dr)*(-dr2dt*Yfd(i-1,1)/1/dr);
    %Yfd(i-1,2)+(dt/1/dr)*(D/dr)*(-dr2dt*Yfd(i-1,1)/1/dr)
    Yavg=Yfd(i,1)*dr^3;
    for j=2:NL-1
        %dr2dt=((dr*j)^2-(drp*j)^2)/(dt);
        Yfd(i,j)=(dt/j/dr)*((D/dr)*(Yfd(i-1,j+1)-Yfd(i-1,j-1)+(Yfd(i-
1,j+1)-2*Yfd(i-1,j)+Yfd(i-1,j-1))/dr))+Yfd(i-1,j); % surface increase
in concentration
        Yavg=Yavg+Yfd(i,j)*((dr*j)^3-(dr*(j-1))^3);

        %(Yfd(i-1,j+1)-Yfd(i-1,j-1)+(Yfd(i-1,j+1)-2*Yfd(i-1,j)+Yfd(i-
1,j-1))/dr)
        %(dt/j/dr)*((D/dr))
        %D
        %pause;
        if Yfd(i,j)>1
            disp(['Yfd at NL is greater than 1 at ' num2str(i) '
timestep']);
            beep;
            toc;
            pause;
        end
        if Yfd(i,j)<0
            disp(['Yfd at NL is less than 0 at ' num2str(i) '
timestep']);
            beep;
            toc;
            pause;
        end
    end
    YfdN=Yfd(i-1,NL)-Yfd(i-1,NL-2)+Yfd(i-1,NL-1);
    dr2dt=((dr*NL)^2-(drp*NL)^2)/(dt);
    D=D*10^-4;% reduction of diffusion rate at surface
    Yfd(i,NL)=(dt/NL/dr)*((D/dr)*(YfdN-Yfd(i-1,NL-1)+(YfdN-2*Yfd(i-
1,NL)+Yfd(i-1,NL-1))/dr)-dr2dt*Yfd(i-1,NL)/NL/dr)+Yfd(i-1,NL);% modify
here with mdot evap mass????????????????????????????? ----------------
--------------------

    Yavg=Yavg+Yfd(i,NL)*((dr*NL)^3-(dr*(NL-1))^3);
    Yavg=Yavg/(dr*NL)^3;%
    Yp=Yavg;

    rhoNL=rholMix(T,Yfd(i,NL));%(Yfd(i,NL)/rhon+(1-
Yfd(i,NL))/rhol(T))^(-1);
    rhoAVG=rholMix(T,Yavg);%(Yavg/rhon+(1-Yavg)/rhol(T))^(-1);
    TMass=rhoAVG*(4/3)*pi()*R^3;%
    NMass=TMass*Yavg;%
    %nanoM=NMass(i);
```

```
    Madd=Mn(i)-NMass;%
    MNL=rhoNL*(4/3)*pi()*((dr*NL)^3-(dr*(NL-1))^3);%
    Yfd(i,NL)=(MNL*Yfd(i,NL)+Madd)/(MNL-mdot*dt+Madd-
MnEvap);%Yfd(i,NL)=(MNL*Yfd(i,NL)+Madd)/(MNL-mdot*dt+Madd-MnEvap);-----
-------------------------------------------------------------------
-------------  Dont include Mdot here?

    % Volume Fraction Check ----------------------------------------
---
    for k=1:NL
        j=(NL-k+1);
        if Yfd(i,j)>Ymax(T)
            if j==NL & tFull==-1
                tFull=i*dt;
            end
            if j==2
                Ts(i)=Ts(i-1);
                shell(i)=shell(i-1);
                loop=i;
                break
            end

            Mover(j)=(4/3)*pi()*((dr*j)^3-(dr*(j-
1))^3)*(Yfd(i,j)*rholMix(T,Yfd(i,j))-Ymax(T)*rholMix(T,Ymax(T)));
            NanoOver(j)=(Yfd(i,j-1)*rholMix(T,Yfd(i,j-
1))*(4/3)*pi()*((dr*(j-1))^3-(dr*(j-2))^3)+Mover(j));
            Yfd(i,j-1)=NanoOver(j)/(rholMix(T,Yfd(i,j-
1))*(4/3)*pi()*((dr*(j-1))^3-(dr*(j-2))^3)+Mover(j));
            Yfd(i,j)=Ymax(T);

        end
    end


    % Surface Weighting
    Yavg_s(i)=Yp;
    Ysw=0;

for k=1:NL-1
    Ysw=Ysw+Yfd(i,NL)*(4/3)*pi()*((dr*(NL-k))^3-(dr*(NL-1-k))^3);%
Volume weighted Avg.
end
    Ysw_s(i)=Ysw/(4/3)/pi()/(R^3);

    % Old Values and YpS Initialization----------------------------
---
    drp=dr;
    YpS=Yfd(i,NL);
    %YpS=Ysw_s(i);

    % Calculate Shell Thickness ------------------------------------
---
    shell(i)=(Mshell(i)*3/rhon/volfM/4/pi()+R^3)^(1/3)-R;
```

```
    % Calc of thermal transfer number using Phi and Bm ----------------
---
    Bt=((1+Bm)^phi)-1;
    %Bts(i)=Bt;

    % Calc of Heat transfer to droplet liquid interior ----------------
---
    %Ql=mdot*(CpfBAR*(Tinf-T)/Bt-L(T));
    if i~= 2
        Tnext=T+(T-Ts(i-2));% numeric estimation of next temp
    end
    if i==2
        Tnext=298;
    end
    Ql=mdot*(CpfBAR*(Tinf-T)/Bt-L(T,Xlf))-Mshell(i)*Cpn*(Tnext-T); %
nano propeties
    %Ql=mdot*(CpfBAR*(Tinf-T)/Bt-L(T,Yp));
    Qls(i)=Ql;

    % Calc of Droplet Temp ---------------------------------------------
---
    Ts(i)=real(T+dt*3*Ql/4/pi()/R^3/rholMix(T,Yp)/CplMix(T,Yp)); % Nano
properties
    %Ts(i)=real(T+dt*3*Ql/4/pi()/R^3/rhol(T)/Cpl(T));

    % End Calculation when evap mostly complete ----------------------
---
    if Rs(i)<(Rs(1)/8)
        loop=i;
        break;
    end
    if volf(T,Yp)>=volfM
        disp(['Complete shell formed at volume fraction of ',
num2str(volf(T,Yp))]);
        loop=i;
        break;
    end

    % Steady State estimate based off temp
    if T>Tprev && flag~=1
        flag=1;
        imin=i;
%         disp('Tmin reached');
%         T
%         i*dt
        %beep;
        %pause;
        %break;
    end
    Tprev=T;
end
toc

disp('Plotting...')
```

```
%Plot ------------------------------------------------------------------
---
t=1:loop;
t=t.*dt;
tnorm=t./((2*Rs(1))^2);

if flag~=1
    imin=10*(1/dt);
end

D2=((2*(Rs(:)+shell(:))).^2)/((2*Rs(1))^2);
fD2=fit(tnorm',D2,'poly1');
coeff=coeffvalues(fD2);

for i=1:loop-imin
Dtemp(i)=D2(i+imin);
ttemp(i)=tnorm(i+imin);
end
fDtemp=fit(ttemp',Dtemp','poly1');
coeff2=coeffvalues(fDtemp);

slope=coeff2(1)*10^6;
disp(['Initial Mass Fraction: ' num2str(massFrac)]);
disp(['Total Evap Rate :' num2str(-slope)]);

%keyboard
% figure();
% plot(t,Vm_s);
% grid on;
% title(['Yp0: ' num2str(Yp0) ' NL : ' num2str(NL)]);

% figure();
% plot(t,TotMix_s);
% hold on;
% plot(t,Thermal_s);
% grid on;


% hold on;
% plot(t,th,'r');
% grid on;
% legend('Radial Step','Calculated Thickness');
% title(['Initial Mass Fraction: ' num2str(massFrac) ' Full Shell at: '
num2str(tFull)])
%
% subplot(3,1,2)
% plot(t,mdots)
% hold on
% plot(t,-mdot_p,'r');
% grid on
% %axis([0 dt*loop 0 1.2*mdots(2)]);
%
% legend('mdot','Permeability mdot');
%
%
```

```
%subplot(3,1,3)
% figure
% plot(t,Ysw_s,'-.')
% hold on
% % plot(t,Ysw_v_s,'r');
% hold on;
% plot(t,Yfd(:,NL));
% hold on;
% plot(t,Yfd(:,NL-1),'r');
% hold on;
% plot(t,Yfd(:,NL-2),'r-.');
% grid on
% plot(t,Ysw_NOM,'r-.');
% hold on;
% plot(t,Ysw_s./Ysw_NOM,'g.');
% hold on;
% plot(t,Ysw_v_s./Ysw_NOM,'b.');
% hold on;
% plot(t,Yavg_s.*(Rs(:)./Rs(1)-log(Rs(:)./Rs(1))),'b-.');
% hold on;
% plot(t,Yavg_s.*(Rs(:)./Rs(1)-2/(Rs(:)./Rs(1)).^(-3)),'+');

%axis([0 dt*loop 0 1.2*mdots(2)]);

%legend('Area, Surface weighted','Volume, Surface Weighted','Surface
Only','Avg Weighted','Ratio of Area, Surf','Ratio of Volume SUrf');
%save(['DATA_' num2str(massFrac)]);
% figure
% plot(t,mdot_n)
% title(['Nano Evap Yp0: ' num2str(Yp0) ' NL : ' num2str(NL)]);
%
% figure
% plot(t,YpS_Calc,'r');
% hold on;
% plot(t,Yfd(:,NL));
% plot(t,Ymax(T));
% title(['Yp0: ' num2str(Yp0) ' NL : ' num2str(NL)]);
% legend('Calc YpS','YpS Used')
end
```