8-2009

Unique Characteristics of Liquid Metal Extended Meniscus Evaporation

Joseph Brown Tipton, Jr.
University of Tennessee - Knoxville

Recommended Citation

Tipton, Jr., Joseph Brown, "Unique Characteristics of Liquid Metal Extended Meniscus Evaporation. " PhD diss., University of Tennessee, 2009.
https://trace.tennessee.edu/utk_graddiss/64

To the Graduate Council:

I am submitting herewith a dissertation written by Joseph Brown Tipton, Jr. entitled "Unique Characteristics of Liquid Metal Extended Meniscus Evaporation." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Mechanical Engineering.

Kenneth D. Kihm, Major Professor

We have read this dissertation and recommend its acceptance:

Masood Parang, Jay I. Frankel, Arthur E. Ruggles

Accepted for the Council:
Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

To the Graduate Council:

I am submitting herewith a dissertation written by Joseph Brown Tipton, Jr. entitled "Unique Characteristics of Liquid Metal Extended Meniscus Evaporation." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Mechanical Engineering.

Kenneth D. Kihm

Major Professor

We have read this dissertation
and recommend its acceptance:

Masood Parang

Jay I. Frankel

Arthur E. Ruggles

Accepted for the Council:

Carolyn R. Hodges

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Unique Characteristics of Liquid Metal Extended Meniscus Evaporation

A Dissertation

Presented for the

Doctor of Philosophy Degree

The University of Tennessee, Knoxville

Joseph Brown Tipton, Jr.

August 2009

Dedication

This dissertation is dedicated to the family of believers at Laurel Church of Christ - my brothers, sisters, teachers, mentors, friends, and fellow followers of the Way for over ten years.

Acknowledgments

I would first like to acknowledge the role of Dr. Kenneth Kihm in my doctoral studies. It was because of him that I chose to pursue doctoral work at the University of Tennessee, and I am grateful for the opportunity that he extended to me. I would also like to thank the other members of my doctoral committee, Dr. Masood Parang, Dr. Jay Frankel, and Dr. Arthur Ruggles. Their guidance, patience, and support throughout this process has proved invaluable.

In addition, I would like to recognize the past and present members of the MINSFET research lab with whom I have worked over the past four years. Doctor Changkyoung Choi, Dr. Iltai Kihm, Dr. Seonghwan Kim, and Hunju Yi adopted me into their Korean culture in the lab and made me feel at ease both as a colleague and as a friend. Doctor Chanhee Chon especially provided support and guidance through my first years in the lab without which I would have been lost. Finally, Dr. Chuck Margraves has kept me sane throughout this doctoral program at different times by being my proofreader, voice of reason, sounding board, running partner, and groomsman. I count him and all of these fine gentlemen as good friends and Christian brothers.

The logistical aspects of my graduate work would have been impossible without the kind and patient help of the MABE department office staff including Pam Ward, Judy Melton, and Cheryl Treece. Also, I would like to offer special thanks to the department electronics technician, Dennis Higdon, who has given me a good education in electronics and computer systems over the course of several years.

My spiritual family has also blessed me with support throughout my graduate school and doctoral journey. Doctor Arnold Lumsdaine, Julian Reese, Dr. Andy Fischer, and Dr. Chuck Margraves helped me to study my calling in life as a Christian academic. The staff at the UT Christian Student Center, Todd Tarbett, Mike Buckley, and Connie Harrell gave me spiritual guidance, friendship, and a place to live during my doctoral studies. I would also be remiss not to mention my adopted grandmother, Caroline Evans, who on countless occasions opened her home, pantry, and heart to a weary graduate student and his wife.

The completion of this dissertation also represents the culmination of over twenty-four years of formal education in my life. As such, there have been many, many kind souls who have invested in me along the way and who deserve a mention of thanks in these acknowledgments. I will single out two grade school teachers who will have to accept thanks in place of all of those who taught me over the years. My fifth grade teacher, Mrs. Irene Vorbusch, deserves a special thanks for setting me on the right scholastic path at a crucial time in my life. Also, I wish to thank my high school band director, Mr. Randy Box, for patiently teaching me the virtues of teamwork, dedication, focus, and organization that have served me so well in my professional work.

My family has also been an important source of encouragement and support along the way. My Uncle, Captain Danny McClendon, deserves special thanks for being my childhood pilot hero. My love of all things aerospace has a lot to do with him. I thank my grandparents, Dan and Virginia McClendon, for always keeping me in their daily prayers. And I thank my grandfather especially for quietly aiding in my informal education through the years. When places were still accessible, he took me to visit a radio transmission tower, the Nashville Airport Control Tower, and in a wonderful display of sacrificial love four Star Trek conventions.

My wonderful parents, Joe and Sherrie Tipton, deserve more thanks than I know how to write. They have been a steady rock in my life, even when they had no idea what I was doing or going through. I simply thank God for such devoted parents.

Finally, I am grateful most of all for the friend and helpmate I found during graduate school, my wife Loraine. She was the “for better” when graduate school insisted upon being the “for worse,” and my gratitude to her will be shown through a lifetime of love and cherishing... and perhaps a trip to the spa.

Abstract

The research in this dissertation addresses the steady evaporation of a capillary pore with a liquid metal working fluid. First, the interline region of an extended meniscus thin film is considered for the unique physical case of a liquid metal. A new thin film evaporation model is presented that captures the unsimplified dispersion force along with an electronic disjoining pressure component that is unique to liquid metals. The resulting nonlinear 4th-order ODE is solved using an implicit orthogonal collocation technique along with the Levenberg-Marquardt method. Results show that the electronic component of the disjoining pressure should be considered when modeling liquid metal extended meniscus evaporation for a wide range of work function boundary values, which represent physical properties of different liquid metals. For liquid sodium, as an example test material, variation in the work function produces order-of-magnitude differences in the film thickness and evaporation profile.

Second, the extended meniscus thin film model is spliced with a CFD model of the evaporating bulk meniscus. The result is a multiscale model of the total evaporating capillary meniscus with a nonisothermal interface and non-equilibrium evaporation. Integration of the evaporative mass flux across the total meniscus surface area produces total capillary evaporative mass flow rates and enables comparisons between electronic disjoining pressure states. The clear trend from these comparisons is that a larger electronic component of the disjoining pressure leads towards larger extended meniscus thin film surface area, larger total capillary meniscus surface area, and larger net evaporative mass flow rate (which corresponds with larger heat flow rate, as well).

Finally, an outline is presented of the scope of the general problem in the application of nonlinear stability theory to a liquid metal evaporating thin film.

Table of Contents

List of Tables

List of Figures

List of Symbols

$a_{1,2,3}$	thin film boundary conditions
A	Hamaker constant $[J]$
B	disjoining pressure electronic component constant $[N]$
Bo	Bond number
Ca	capillary number $[\mathrm{Ca} = \mu_l \cdot u_0/\gamma]$
c	speed of light in a vacuum $[m/s]$
$c_{1,2,3,4}$	cubic spline coefficients
c_P	specific heat capacity at constant pressure $[J/kg \cdot K]$
D	diameter $[m]$
E	energy $[eV]$
f	collocated residual function
f	fugacity $[N/m^2]$
$\mathbf{F}$	forcing vector
g	mathematical forcing function
g	gravitational acceleration $[m/s^2]$
G	Gibb's free energy $[J]$
G	modified Jacobian matrix
$\hbar$	reduced Planck constant (Dirac's constant) $[J \cdot s]$
h_{fg}	latent heat of evaporation $[J/kg]$
H	film thickness $[m]$
H_0	adsorbed film thickness $[m]$

H_{tr}	transition film thickness (where disjoining pressure balances capillary pressure) $[m]$
i	imaginary number $[\sqrt{-1}]$
I	identity matrix
J	Jacobian matrix
k	$d\gamma/dT$ $[N/m \cdot K]$
k_B	Boltzmann constant $[m^2 \cdot kg/s^2 \cdot K]$
K	thin film curvature $[m^{-1}]$
l_{max}	right hand limit to function domain $[m]$
m_e	electron mass $[kg]$
$\dot{m}''_{evp}$	evaporative mass flux $[kg/s \cdot m^2]$
$\dot{M}''$	nondimensional evaporative mass flux
$M[\cdot]$	nonlinear operator
$\mathcal{M}$	molar mass $[kg/mol]$
Ma	Marangoni number $[\mathrm{Ma} = \rho \cdot c_P \cdot k \cdot H_{tr} \cdot \Delta T/\mu \cdot \lambda]$
n	index of refraction
n	mass density $[mol/m^3]$
N	maximum number of terms in an expansion
N_e	valence electron number density $[m^{-3}]$
p	variable of integration or iterate
P	pressure $[N/m^2]$
Pr	Prandtl number $[\mathrm{Pr} = c_P \cdot \mu/\lambda]$
q_e	electron charge $[C]$
q_k	orthogonal collocation coefficients
q''	heat flux $[W/m^2]$
$\mathbf{q}$	collocation coefficient difference vector
$\mathbf{Q}$	L_2 norm of the residual
r	radial coordinate (cylindrical CS) $[m]$
r_m	orthogonal collocation coefficients
R	pore radius $[m]$

R_{IHT}	thermal interfacial resistance $[m^2/W \cdot K]$
R_C	thermal conductive resistance $[m^2/W \cdot K]$
R_N	residual function
$\mathcal{R}$	universal gas constant $[N \cdot m/K \cdot mol]$
s	volumetric entropy $[J/K \cdot m^3]$
$s_{1,2}$	equation placeholders
S	cubic spline equation
S	surface domain of integration
SA	surface area $[m^2]$
t	scalar step-size parameter
T	temperature $[K]$
T_m	Chebychev polynomials of the first kind
ΔT	liquid overheat $[K]$
u	horizontal velocity $[m/s]$
v	vertical velocity $[m/s]$
v	specific volume $[m^3/kg]$
v_e	plasma frequency of an electron gas $[Hz]$
V_M	molar volume $[m^3/mol]$
W	work function $[J]$
x	axial coordinate (Cartesian CS) $[m]$
y	vertical coordinate (Cartesian CS) $[m]$
z	vertical coordinate (cylindrical CS) $[m]$
Z	gas compressibility factor

Greek Symbols

α	evaporation coefficient
γ	surface tension $[N/m]$
Γ	liquid mass flow rate per unit width $[kg/s \cdot m]$

Γ^*	nondimensional liquid mass flow rate per unit width
δ	Dirac delta function
ϵ_0	permittivity of free space $[s^4 \cdot A^2/m^3 \cdot kg]$
ϵ_1	relative permittivity of container
ϵ_2	relative permittivity of vapor
ϵ_3	relative permittivity of liquid thin film
ζ	energy level width $[J]$
η	nondimensional axial coordinate $[\eta = x/x_0]$
θ	angular coordinate (cylindrical CS) $[rad]$
θ	nondimensional film thickness $[\theta(\eta) = H/H_0]$
$\hat{\theta}$	nondimensionalized thin film thickness in ξ $[\hat{\theta} = \theta\,(\phi(1 + \xi))]$
$\hat{\Theta}$	approximate solution in ξ using truncated terms
Θ	Heaviside function
κ	ratio of evaporative interfacial resistance to conductive resistance
κ_n	electronic disjoining pressure work function parameter
ƛ	wavelength $[m]$
λ	thermal conductivity $[W/m \cdot K]$
μ	viscosity $[N \cdot s/m^2]$
μ	chemical potential $[J/mol]$
ν	kinematic viscosity $[m^2/s]$
ξ	nondimensionalized distance mapped for Chebyshev polynomials
Π	disjoining pressure $[N/m^2]$
Π^*	nondimensional disjoining pressure $[\Pi^* = \Pi/\Pi_0]$
ρ	density $[kg/m^3]$
ρ_e	electron density of states $[electrons/J \cdot m^3]$
σ	optical conductivity $[S/m]$
$\Sigma_{1,2}$	equation placeholders
τ	relaxation time $[s]$
ϕ	linear domain transformation parameter

χ	electronic disjoining pressure boundary condition
$\hat{\Psi}$	collocation inhomogeneous, linear, boundary conditions
Ψ_k	collocation Chebyshev polynomials with homogeneous boundary conditions
ω	frequency $[rad/s]$
ω_e	plasma frequency of an electron gas $[rad/s]$
ω_n	electromagnetic wave frequency $[rad/s]$

Subscripts

0	reference state
A	dispersion component of the disjoining pressure
B	electronic component of the disjoining pressure
F	Fermi
i	cubic spline piece
lv	liquid/vapor interface
l	liquid
n	summation index
tf	thin film
v	vapor
w	wall

Chapter 1

Introduction / Literature Survey

The evaporation of fluids provides an efficient method for heat transfer and passive cooling in devices such as heat pipes and capillary pumped loops. A liquid metal working fluid enables operation in extremely high temperature environments with the added benefits of a high latent heat of evaporation and high heat transfer coefficient. Numerical and experimental studies have applied liquid metal heat pipes and capillary pumped loops in nuclear [1–3], hypersonic [4–8], and space based [9–14] systems. In the case of aerospace systems, minimization of system mass calls for a consideration of micro-scale heat pipes and capillary pumped loops. Micro-scale heat pipes have already found application in the cooling of high performance electronics [15–21]. In such micro-scale systems using conventional coolant fluids, such as water or refrigerants, the thin film region has been shown to contribute greatly towards meniscus stability and evaporation [22, 23].

High temperature, liquid metal evaporation on the micro-scale, however, has received little attention in the literature [24, 25]. One of the main difficulties in analytically studying liquid metal evaporation arises from the complexity of the disjoining pressure, which renders traditional models and solution schemes invalid. To address this knowledge gap, this research proposes new models for the extended meniscus evaporation of alkaline metal, such as liquid sodium, under capillary and dispersion forces as well as a relatively newly proposed force due to degeneracy of the free electrons in a liquid metal thin film. In the process, the general extended meniscus model is to be combined with a CFD model of the bulk evaporating meniscus to create a true multiscale model of the evaporating capillary meniscus.

Finally, Capillary Pumped Loops (CPL) and Loop Heat Pipes (LHP) are "real world" heat transfer devices that utilize the unique physics of capillary evaporation. The research so-far mentioned models the

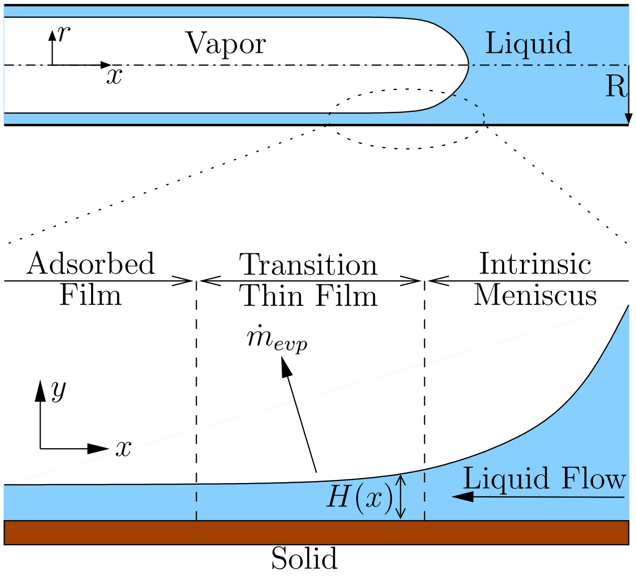


Figure 1.1: Schematic of a cylindrical capillary geometry identifying the distinct regions of the extended evaporating meniscus. The majority of heat and mass transfer occurs in the transition thin film region. For large enough pore radii, the transition thin film region may be simplified from a cylindrical to Cartesian domain.

evaporating liquid metal capillary under the important assumption of a constant, steady-state meniscus profile. Experimental investigations of CPLs and LHPs with traditional working fluids, however, have demonstrated performance degradations due to temporal fluctuations [26]. Thus, this dissertation also seeks to outline the scope of the general problem in the application of nonlinear stability theory to a liquid metal evaporating thin film in an effort to direct future research.

1.1 Thin Film Modeling

As shown in Figure 1.1, the interline or contact line region of an evaporating extended meniscus consists of three subregions. In the adsorbed region, a disjoining pressure dominates the local atomic forces. In the intrinsic or bulk meniscus region, the interfacial curvature governs the driving physics through surface tension. The transition or thin-film region exists between the intrinsic meniscus and adsorbed regions where both the disjoining pressure and the interfacial curvature share a comparable influence.

Previous studies addressed numerical heat and mass transfer solutions for steady extended meniscus evaporation. Here, steady connotes a static interline region continually replenished by fluid from the intrinsic meniscus [22, 27–35]. Wayner and Schonberg [36] developed a governing equation for the film height of a symmetric meniscus as a function of distance between two feed ports. Their development draws upon the pioneering thin film experiments of Derjaguin [37], Schrage's [38] relationship for net mass flux across a liquid/vapor interface, and the evaporating extended meniscus models of Wayner *et al.* [39] and Potash and Wayner [40]. Later, Chebaro and Hallinan [41], Chebaro *et al.* [42], and Hallinan *et al.* [43] introduced new nondimensional variables, which re-expressed Wayner and Schonberg's model in a more meaningful manner. They also created an explicit Runge-Kutta numerical solution procedure, which meets the correct system boundary conditions by way of the "Shooting Method".

1.2 Disjoining Pressure

Derjaguin and Kusakov [44, p.27] first coined the term "wedging-apart pressure" (a literal translation from Russian) to account for pressure differences experimentally obtained between a thin film and its bulk phase. The more popular English paraphrase "disjoining pressure" is now more popularly used. Davis provides a lucid description of the concept [45, p.371]:

> The concept of "disjoining" is that to squeeze a film to thickness H, an excess pressure of $\Pi(H)$ must be applied to offset the tendency of the film phase to separate or disjoin the confining phases. If Π is negative, the film material wants to retreat from the region between the confining phase. Thus, if $\Pi > 0$, the film material is wetting, and if $\Pi < 0$, it is nonwetting.

This work shall adopt the terminology and convention of the Russian school of thought although there do exist differences of opinion regarding its clarity and usefulness [46, p.268]. As such, the long-range nature of the disjoining pressure can include ionic-electrostatic, molecular, structural, adsorption, and electronic components. The ionic-electrostatic component incorporates double-layer interactions. The molecular component deals with van der Waals interactions. The structural component includes solvation forces created by molecules near surface interfaces [46, p.268]. The adsorption component covers interactions of adsorbed layers of nonionic surfactants and macromolecules [47]. Finally, the electronic component consists of forces induced by the electron gas in a liquid metal [48]. The present work considers only the molecular and electronic components of the disjoining pressure as the working fluid is chosen to be a liquid metal.

1.2.1 van der Waals Component

The van der Waals forces account for long-range atomic forces between neutral atoms and can be subdivided into three subcategories. Molecules with permanent dipole moments experience Keesom orientation forces due to dipole-dipole interaction and Debye induction forces due to dipole-induced dipole interactions. All molecules, both polar and non-polar, experience London dispersion forces as a result of induced dipole-induced dipole interactions.

Hamaker [49] first described the London dispersion forces between two media acting in a third medium by considering the microscopic interaction between two molecules and summing under the assumption of additivity. Hamaker's theory does not include the Keesom and Debye forces of polar molecules, the effects of temperature, or time delay effects in the communication of electromagnetic fields between atoms at larger distances (retardation). Lifshitz [50] later pursued a macroscopic approach which modeled the bulk interaction between two media in a vacuum by considering the fluctuation of electromagnetic fields between the two media. Lifshitz's theory solves the shortcomings of Hamaker's theory but is limited to interactions in a vacuum. Finally Dzyaloshinskii, Lifshitz, and Pitaevskii (DLP) used quantum electrodynamics to derive the first general theory of van der Waals forces [51]. While much more complex mathematically, the DLP theory successfully includes the interaction of two media in a third medium. The only major restriction is the assumption of planar geometries.

Since the DLP theory originates from a macroscopic perspective, the van der Waals forces can be described using continuum properties of the participating media in the form of their frequency-dependent, dielectric permittivities. When temperature effects can be neglected and when the film thickness is small compared to the absorption wavelengths of the participating media, the retardation effects of the time delay in the communication of electric fields between atoms may be neglected, and the full DLP theory can be greatly simplified. The resulting nonretarded force is proportional to the inverse cube of the film thickness. This relationship is the macroscopic analogue of the microscopic Hamaker theory. It is most often used in the engineering literature in the form of the well known Hamaker constant [52, pp.137-152]. Thus, the Hamaker constant represents the limiting case of nonretarded dispersion forces. Prevailing convention assigns a negative value for the Hamaker constant for the case of spreading films, although the literature can be confusing [29, 31–34, 53–55].

At the opposite limit of a thick film, which results in a fully retarded dispersion force, several papers have mentioned an analytical solution proportional to the inverse forth power of the film thickness [32, 54–56]. These works fail to mention the requirements for this solution, namely dielectric materials, film thicknesses much greater than "the wavelengths which characterise the absorptions spectra of the given bodies," and film thicknesses much less than the temperature requirement $H \ll \hbar c/k_B T$ [51]. Only a select number of working fluids and operating temperatures meet these requirements. The case of a high temperature, liquid metal, evaporating thin film, however, invalidates each of these assumptions.

1.2.2 Electronic Component

The original disjoining pressure concept, first proposed by Derjaguin in the 1930s, addresses the additional thin film pressure created by van der Waals and electrostatic forces. Almost fifty years later, Derjaguin, Leonov, and Roldughin [48] and Derjaguin and Roldughin [57] proposed the existence of an additional form of disjoining pressure in liquid metal films. Inspired by a theoretical prediction of anomalous effects in nanoscale metallic particles [58], they surmised that the free electrons in a thin metal film, modeled as a fermion gas, would experience a confinement in their position. According to Heisenberg's uncertainty principle, this confinement correlates with an increase in momentum. This electron degeneracy creates an increase of the energy density in the thin film and produces an effective "electron pressure" (for a good summary, see Roldughin [59]).

Derjaguin and Roldughin assumed films sufficiently thick such that the dispersion forces could be neglected. In addition, they assumed thick films, negligible exchange, correlation, and electrostatic interactions, a perfectly smooth surface, and a model of the electron as a non-interacting particle. In this way, they were able to derive a relationship between the change in kinetic energy of free electrons in the thin film and the disjoining pressure using quantum mechanical theory. The resulting electron degeneracy disjoining pressure varies in intensity and sign depending upon the work function (energy needed to move an electron from the liquid metal to the solid surface) of the system.

Derjaguin *et al.* indirectly proved the existence of the electronic component to the disjoining pressure by experiment. The DLP Theory of van der Waals forces predicts any two identical media will attract each other, irregardless of the media in between. Thus, a *free* liquid metal thin film should experience a negative disjoining pressure which would render the film unstable. In contrast, Derjaguin *et al.* demonstrated the stability of free films of liquid mercury in certain organic liquids, which could only be possible if a larger, positive disjoining pressure component was present.

1.2.3 Total Disjoining Pressure

To the author's knowledge, the only previous attempt to model a liquid metal thin film using both the London dispersion force and electron degeneracy force as components of a disjoining pressure was by Ajaev and Willis [60, 61]. They were concerned with "thermocapillary flow and rupture in films of molten metal on a substrate" when heated by a Gaussian laser beam. Ajaev and Willis correctly identified the need for both components of the disjoining pressure and, with neither a fundamental physics model nor experimental measurements available, suggested a linear combination of the two. Their paper presented a general parametric study that looked at model trends only. No attempt was made to calculate the appropriate value for either component of the disjoining pressure. In addition, the electronic component was treated as a constant, positive value under the simplifying assumption of an infinite potential energy well boundary condition.

1.3 Comprehensive Multiscale Capillary Modeling

Several authors have attempted to model the full capillary evaporating meniscus at steady-state with varying degrees of complexity and success. Swanson and Herdt [30] attempted to model the entire micro- and macro-capillary domain using one characteristic set of equations. Chebaro *et al.* [42] pointed out several flaws:

> "Swanson and Herdt's analysis inexactly made assumptions pertaining to the curvature of the interface in the interline region, the radial pressure gradient in the meniscus, and the tangential shear stress boundary condition at the interface in the meniscus."

Stephan and Busse [62] sought to model a groove heat pipe wall geometry. Their thin film extended meniscus model only included an isothermal interface and thermocapillary forces were assumed negligible. The wall temperature in the micro region was assumed and the thin film solution yielded the curvature of the bulk meniscus, the temperature distribution at the interface, and the total heat transferred in the micro region. Heat transfer in the bulk meniscus fluid region and groove walls was solved via a FEM conduction model that did not consider fluid flow. The capillary surface was considered static and nonevaporative. The micro and macro region models were iterated until they agreed on the wall temperature and heat flux at their interface.

Schonberg *et al.* [34] also modeled the thin film extended meniscus with an isothermal interface and without thermocapillary forces. The bulk meniscus region was solved via a FEM conduction heat transfer model only. The curvature was assumed to be constant. Walls were constant temperature. It was assumed that evaporative heat transfer did not take place in the bulk meniscus which effectively decoupled the micro and macro models. In fact, the truncation surfaces on the bulk meniscus model were designed to be insulative.

Khrustalev and Faghri [63, 64] built a multiscale numerical model of an evaporating capillary meniscus. Their model included "two-dimensional steady-state momentum conservation and energy equations for both the vapor and liquid phases, and incorporate[d] the existing simplified one-dimensional model of the evaporating microfilm." Thermocapillary effects were not considered. The meniscus was considered to be a constant shape, and the phase change across the interface was modeled using equilibrium mass and energy conservation.

Kim [65] and Ji *et al.* [66] both created numerical models of an evaporating capillary that included fluid flow and thermocapillary effects. Both models also included an evaporative boundary condition via the Hertz-Knudsen relationship. Heat transfer occurred via a constant wall temperature boundary condition. Ji *et al.* only considered a simplified (an unrealistic) rectangular domain. Kim utilized a constant meniscus profile that was transformed to a rectangular domain for ease of computation using a boundary fitted coordinate system. It does not appear that Kim considered the velocity at the evaporative surface to be specified through the evaporative mass flux. Neither studies considered the effects of thin film extended meniscus evaporation in the micro region.

1.4 Thesis Statement

This research seeks to model the evaporation of high temperature, liquid metal, thin films and thus distinguishes itself from previous thin film evaporation studies of more conventional liquids. The novel aspects of this research include:

(i) an accurate model of the retarded dispersion force component of the disjoining pressure,

(ii) the incorporation of the electronic component of the disjoining pressure,

(iii) a parametric study of the thin film solution over a range of pore radii, liquid overheats, and system work functions.

(iv) integration of the extended meniscus thin film model with a CFD model of the bulk evaporating meniscus to create a comprehensive multiscale model of a liquid metal evaporating capillary, and

(v) an outline of the scope of the general problem in the application of nonlinear stability theory to a liquid metal evaporating thin film under thermocapillary effects.

Chapter 2

Extended Meniscus Evaporation Model

2.1 Interfacial Evaporative Mass Flux

The Hertz-Knudsen-Langmuir (HKL) equation [67, p.341] uses kinetic theory to model the net mass flux of a liquid/vapor phase interface in the intermediate range between equilibrium and free evaporation. For the case of net evaporation

$$\dot{m}''_{evp} = \alpha \left(\frac{\mathcal{M}}{2\pi\mathcal{R}} \right)^{1/2} \left(\frac{P_{lv}}{T_{lv}^{1/2}} - \frac{P_v}{T_v^{1/2}} \right) \approx \alpha \left(\frac{\mathcal{M}}{2\pi\mathcal{R}T_v} \right)^{1/2} (P_{lv} - P_v) \tag{2.1}$$

where P_{lv} is the saturation pressure of the liquid at the liquid/vapor interface at temperature T_{lv} while P_v is the pressure of the vapor at temperature T_v. The HKL relationship depends upon several assumptions, e.g.

1. independence of the two molecular fluxes ($\rho_v \ll \rho_l$),
2. equivalent evaporation and condensation coefficients ($\alpha_{evap} = \alpha_{cond} = \alpha$),
3. no temperature jump across the interface ($T_{l,lv} = T_{v,lv} = T_{lv}$),
4. use of an equilibrium molecular distribution function under nonequilibrium conditions,
5. the vapor modeled as an ideal gas ($\rho_v = \frac{P_v\mathcal{M}}{\mathcal{R}T_v}$), and
6. no molecular backscattering near the liquid surface.

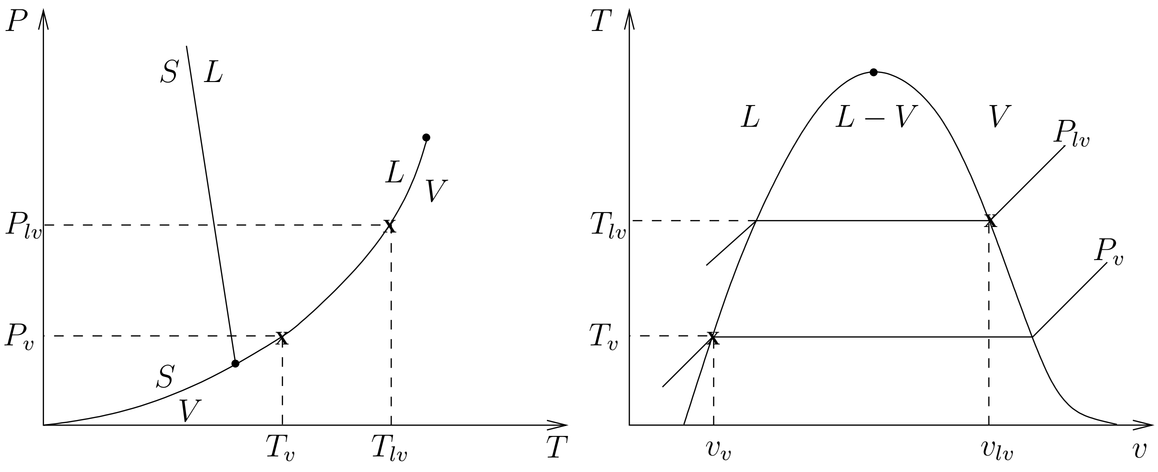


Figure 2.1: Diagrams of the thermodynamic states of the liquid and vapor which drive the evaporation process. S, L, and V indicate the solid, liquid, and vapor phases respectively. P, v, and T indicate pressure, specific volume, and temperature respectively.

Schrage [38, pp.34-36] further generalized the HKL theory when he incorporated nonequilibrium molecular distribution with the model of uniform gas motion at a planar phase interface. To a first approximation, the nonlinearities introduced by the bulk gas velocities can be linearized, and the net evaporative mass flux can be re-expressed as

$$\dot{m}''_{evp} = \left(\frac{2\alpha}{2-\alpha}\right)\left(\frac{\mathcal{M}}{2\pi\mathcal{R}T_v}\right)^{1/2}(P_{lv}-P_v)\,,\ \frac{P_v}{P_{lv}} \to 1 \tag{2.2}$$

which is referred to as the Hertz-Knudsen-Schrage (HKS) equation [67, p.346]. Thus, the net evaporative mass flux is driven by the the pressure difference across the fluid interface.

The Clapeyron equation characterizes the phase transition between a liquid and vapor. Assuming that the phase transition takes place at equilibrium and at constant pressure and temperature, then

$$\frac{dP}{dT} = \frac{h_{fg}}{T\Delta v} \tag{2.3}$$

which describes the coexistence curve on a pressure-temperature diagram, such as Figure 2.1. If the final state can be modeled as an ideal gas, then integration of the above equation produces

$$\ln\left(\frac{P_v}{P_{lv}}\right) = \frac{\mathcal{M}h_{fg}}{\mathcal{R}}\left(\frac{1}{T_{lv}} - \frac{1}{T_v}\right) \tag{2.4a}$$

or

$$\ln\left(\frac{P_{lv}}{P_v}\right) = \frac{\mathcal{M}h_{fg}}{\mathcal{R}T_vT_{lv}}\left(T_{lv} - T_v\right) \tag{2.4b}$$

which is known as the Clausius-Clapeyron equation.

In addition, the Kelvin equation describes the change of vapor pressure during the liquid/vapor phase change due to van der Waals intermolecular forces

$$\ln\left(\frac{P_{lv}}{P_v}\right) = -\Delta P\frac{V_{M,l}}{\mathcal{R}T_{lv}} \tag{2.5}$$

where ΔP represents the equilibrium pressure difference sustained across the interface between two static fluids, such as water and air, due to the intermolecular forces. From this relationship, it is evident that positive van der Waals forces cause $P_{lv} < P_v$ which acts to inhibit evaporation from the interface. The Augmented Laplace-Young equation relates the van der Waals intermolecular forces as a combination of capillary and disjoining pressures

$$\Delta P = P_v - P_l = \gamma K + \Pi \tag{2.6}$$

where K is the mean curvature of the liquid/vapor interface. The concept of a linear combination of the disjoining and capillary pressures is not new [68] and has been experimentally validated [54].

The mean curvature for a circular pore is given by Philip [69]

$$K = \left\{\frac{1}{r\left[1+\left(\frac{dr}{dx}\right)^2\right]^{1/2}} - \frac{\frac{d^2r}{dx^2}}{\left[1+\left(\frac{dr}{dx}\right)^2\right]^{3/2}}\right\} \tag{2.7a}$$

where the first and second terms represent the circumferential and axial curvatures, respectively. If only large pore radii are considered, the circumferential curvature may be neglected and the axial curvature may be modeled in a 2D Cartesian frame of reference

$$K \approx \frac{d^2H}{dx^2}\left[1+\left(\frac{dH}{dx}\right)^2\right]^{-3/2} \tag{2.7b}$$

using the relationship $H = R - r$. Finally, it is assumed that the slope is small in the thin-film region ($dH/dx \approx 0$) which reduces the curvature to the second derivative of the film thickness

$$K \approx \frac{d^2H}{dx^2} \ . \tag{2.7c}$$

The effects of the Clausius-Clapeyron and Kelvin relationships on the evaporating thin film are combined by way of Rusanov's "surface excess convention" [70, pp.22-34]. The Gibbs-Duhem equations for the bulk liquid and vapor phases are

$$dP_l = s_l dT + n_l d\mu_l \tag{2.8a}$$

and

$$dP_v = s_v dT + n_v d\mu_v \ . \tag{2.8b}$$

These two equations may be combined through the Augmented Laplace-Young relationship [39] seen in Equation (2.6) to give

$$d\mu = -V_{M,l} d(\Delta P) + \frac{\mathcal{M} h_{fg}}{T} dT \tag{2.9}$$

where $(\Delta n)^{-1} \approx V_{M,l}/\rho_l$ and $d\mu_l = d\mu_v = d\mu$ according to local equilibrium. This can also be written as

$$d(\ln f) = -\frac{V_{M,l}}{\mathcal{R}T} dP + \frac{\mathcal{M} h_{fg}}{RT^2} dT \tag{2.10}$$

using the fugacity concept where $d\mu = RTd(\ln f)$. The fugacity can be replaced by the vapor pressure for small changes in the fugacity. Upon integration the above equation produces

$$P_{lv} - P_v = -\frac{V_{M,l} P_v}{\mathcal{R} T_{lv}} (\gamma K + \Pi) + \frac{P_v V_{M,l} h_{fg}}{R T_{lv} T_v} (T_{lv} - T_v) \tag{2.11}$$

where the natural logarithm is approximated by the first term of its Taylor series. The above equation is known as an extended Clapeyron equation and was first used by Wayner *et al.* [39] and later derived by Wayner [71]. If the overheat is considered small, $T_{lv}^{1/2} \approx T_v^{1/2}$ and Equations (2.2) and (2.11) may be combined to yield

$$\dot{m}''_{evp} = \left(\frac{2\alpha}{2-\alpha}\right)\left(\frac{\mathcal{M}}{2\pi\mathcal{R}T_v}\right)^{1/2}\left[\frac{P_v\mathcal{M}h_{fg}}{\mathcal{R}T_vT_{lv}}(T_{lv}-T_v)-\frac{V_lP_v}{\mathcal{R}T_{lv}}(\gamma K+\Pi)\right] \tag{2.12}$$

which describes the net evaporative mass flux at the liquid/vapor interface as a function of the liquid overheat (Clausius-Clapeyron effect) and van der Waals intermolecular forces (Kelvin effect).

If the liquid/vapor interface is considered nonisothermal, then the temperature at the interface (T_{lv}) is a function of heat transfer through the thin film. The energy equation may be approximated as

$$\frac{d^2T}{dy^2} = 0 \tag{2.13a}$$

under the assumptions of negligible heat convection, negligible axial heat conduction, and constant wall temperature (T_w). At the substrate boundary, the temperature must equal the temperature of the wall

$$T(0) = T_w \tag{2.13b}$$

while at the liquid/vapor interface, conduction must equal the net evaporative heat flux

$$-\lambda\frac{dT}{dy}\bigg|_{y=H} = \dot{m}''_{evp}h_{fg} \quad . \tag{2.13c}$$

The solution of Equation (2.13a) with the established boundary conditions gives an equation for temperature through the thin film

$$T(y) = T_w - \left(\frac{\dot{m}''_{evp}h_{fg}}{\lambda}\right)y \tag{2.14}$$

which, when evaluated at the thin film height H, then yields the temperature at the liquid/vapor interface

$$T_{lv} = T_w - \left(\frac{\dot{m}''_{evp}h_{fg}}{\lambda}\right)H + T_w \quad . \tag{2.15}$$

At this point, we may follow Hallinan *et al.* [43] and simplify our development with additional appropriate nondimensional variables

$$\theta = H/H_0 \tag{2.16a}$$

$$\eta = x/x_0 \tag{2.16b}$$

$$\Pi^* = \Pi/\Pi_0 \tag{2.16c}$$

$$\dot{m}_0'' = \rho_l u_0 \tag{2.16d}$$

$$\dot{m}_0'' = \left(\frac{2\alpha}{2-\alpha}\right)\left(\frac{\mathcal{M}}{2\pi\mathcal{R}T_v}\right)^{1/2}\left(\frac{P_v\mathcal{M}h_{fg}}{\mathcal{R}T_vT_{lv}}\right)(T_w - T_v) \tag{2.16e}$$

$$Ca = \mu_l u_0/\gamma \tag{2.16f}$$

$$x_0 = (\gamma H_0/\Pi_0)^{1/2} \tag{2.16g}$$

$$\Pi_0 = \frac{\mathcal{M}h_{fg}\Delta T}{V_l T_v} \tag{2.16h}$$

$$\Delta T_0 = T_w - T_v \tag{2.16i}$$

$$T^* = \frac{T_{lv} - T_v}{T_w - T_v} \tag{2.16j}$$

$$\kappa = \frac{\dot{m}_{evp}'' h_{fg}}{(\lambda H_0)} \tag{2.16k}$$

where H_0 is defined as the adsorbed film thickness and is determined by solving Equation (2.12) with negligible evaporative mass flux and curvature and assuming that $T_{lv} \approx T_w$. Equations (2.12) and (2.15) combined with the nondimensionalized definitions of Equation (2.16) produce an expression for the nondimensionalized evaporative mass flux

$$\dot{M}'' = \frac{\dot{m}_{evp}''}{\dot{m}_0''} = T^* - \theta(\eta)'' - \Pi^* \tag{2.17}$$

where the nondimensional temperature is

$$T^* = \frac{\Delta T_0 + \kappa\left[\theta(\eta)\theta(\eta)'' + \theta(\eta)\Pi^*\right]}{\Delta T_0 + \kappa\theta(\eta)}\ . \tag{2.18}$$

2.2 An Aside: Interfacial Heat Transfer Resistance Concept

Following Kamotani [72], we observe the extended meniscus evaporation model from a heat transfer frame of reference. Doing so elucidates the role of the disjoining pressure and liquid/vapor interface. The specific heat flux is related to the net evaporative mass flux by

$$q'' = h_{fg}\dot{m}''_{evp} \tag{2.19}$$

which, when combined with Equation (2.12) becomes

$$q'' = h_{fg}\left(\frac{2\alpha}{2-\alpha}\right)\left(\frac{\mathcal{M}}{2\pi\mathcal{R}T_v}\right)^{1/2}\left[\frac{\rho_v h_{fg}}{T_{lv}}(T_{lv}-T_v)-\frac{\rho_v}{\rho_l}(P_v-P_l)\right] . \tag{2.20}$$

If conduction of heat through the thin film is taken into account, the temperature of the liquid/vapor interface may be represented as a function of the wall temperature

$$T_{lv} = T_w - \frac{q''H}{\lambda} . \tag{2.21}$$

Upon combination of the previous two equations and considerable rearrangement, and with T_v readily interchanged with T_{lv} when necessary, Kamotani obtained the following representation of specific heat flux in the thin film system

$$q'' = \frac{T_w - T_v\left[1+\frac{(P_v-P_l)}{\rho_l h_{fg}}\right]}{R_C + R_{IHT}} \tag{2.22}$$

where R_C and R_{IHT} represent heat transfer resistances due to conduction

$$R_C = \frac{H}{\lambda} \tag{2.23}$$

and interfacial heat transfer

$$R_{IHT} = \left(\frac{2-\alpha}{2\alpha}\right)\left(\frac{2\pi\mathcal{R}T_v}{\mathcal{M}}\right)^{1/2}\left(\frac{T_v}{\rho_v h_{fg}^2}\right) . \tag{2.24}$$

In this way, we can clearly see that the interfacial evaporative process serves to govern the heat transfer process, especially for the case of a liquid metal working fluid for which the conductive resistance is negligible. Furthermore, the presence of curvature and disjoining pressures are seen to reduce heat transfer in the thin film system as if the vapor temperature were increased.

2.3 Fluid Flow

The geometry and scale of the thin film region validate an assumption of incompressible, laminar, parallel, 1D liquid flow. We additionally assume a steady-state source of fluid exists outside the analyzed region such that the meniscus shape remains constant. Lubrication theory then models fluid flow in the thin film

$$\mu_l \frac{d^2u}{dy^2} = \frac{dP_l}{dx} \tag{2.25a}$$

with a no-slip boundary condition at the substrate

$$u(0) = 0 \tag{2.25b}$$

and a negligible surface tension gradient at the liquid-vapor interface

$$\frac{du(H)}{dy} = 0 \, . \tag{2.25c}$$

The one dimensional, second-order differential equation can be solved by integrating twice and applying the boundary conditions to yield

$$u(y) = \frac{1}{\mu_l}\frac{dP_l}{dx}\left(\frac{y^2}{2} - Hy\right) \tag{2.26}$$

which represents the liquid velocity profile as a function of the axial pressure gradient in the liquid. Using the velocity profile, the total thin film mass flow rate per unit width is determined to be

$$\Gamma = \frac{\rho_l}{\mu_l}\frac{dP_l}{dx}\int\limits_{y=0}^{H}\left(\frac{y^2}{2} - Hy\right)dy = \frac{-H^3}{3\nu_l}\frac{dP_l}{dx} = \frac{-H_0^3}{3\nu_l x_0}\theta(\eta)^3\frac{dP_l}{d\eta} \, . \tag{2.27}$$

If we define a nondimensional mass flow rate $\Gamma^* = \Gamma\frac{x_0\nu_l}{\Pi_0 H_0^3}$ and assume a constant vapor pressure across the thin film interface such that $dP_v/d\eta = 0$, then Equations (2.6), (2.7), and (2.16) may be combined with Equation (2.27) to yield

$$\Gamma^* = \frac{1}{3}\theta(\eta)^3\theta(\eta)''' + \frac{1}{3}\theta(\eta)^3(\Pi^*)' \, . \tag{2.28}$$

2.4 Mass Balance

Conservation of mass along the thin film requires that the reduction in the liquid flow rate equal the net evaporative mass flux. Thus

$$\frac{d\Gamma}{dx} = -\dot{m}''_{evp} \tag{2.29a}$$

or, in nondimensional form

$$\frac{d\Gamma^*}{d\eta} = -\frac{\text{Ca}}{\left(\frac{H_0\Pi_0}{\gamma}\right)^2}\,\dot{M}'' \ . \tag{2.29b}$$

Substitution of Equations (2.17) and (2.28) into (2.29b) yields an equation that models the steady extended meniscus evaporation as a nonlinear, inhomogeneous, fourth-order, ordinary differential equation (ODE)

$$\left[\theta(\eta)^3\theta(\eta)''' + \theta(\eta)^3\Pi^*(\theta)'\right]' = \frac{-3\,\text{Ca}}{\left(\frac{H_0\Pi_0}{\gamma}\right)^2}\left[T^* - \theta(\eta)'' - \Pi^*(\theta)\right],\ \eta \in [0,\infty) \tag{2.30a}$$

where $\eta = 0$ represents the adsorbed film and $\eta = \infty$ represents the bulk meniscus region. The boundary conditions chosen to describe the system are

$$\theta(0) = a_1 \tag{2.30b}$$

$$\theta'(0) = a_2 \tag{2.30c}$$

$$\theta''(\infty) = a_3 \tag{2.30d}$$

$$\theta'''(0) = 0\ . \tag{2.30e}$$

The initial perturbations of the independent variable and its first derivative are necessary to avoid a trivial solution and do correspond to physical realities as described in Hallinan *et al.* [43] (albeit somewhat tenuously) where $a_1 = 1.030$ and $a_2 = 0.0004$. The boundary condition on the second derivative of the independent variable is $a_3 = K$ where K is the curvature of the bulk meniscus region. Thus, in practice, $\eta = \infty$ is taken to be a point in the far-field, l_{max}, where the second derivative approaches an asymptotic value that is the reciprocal of the pore radius ($K = 1/R$).

Chapter 3

Disjoining Pressure Development

3.1 Dispersion Force Component

3.1.1 General Theory

The Dzyaloshinskii, Lifshitz, and Pitaevskii (DLP) General Theory of Van der Waals Forces [51] describes the dispersion force per unit area between two smooth media with nonpolluted surfaces (1 and 2) while separated by a gap (H) that is filled with a third medium (3) as illustrated in Figure 3.1. The dispersion force created by the interactions of the long range atomic forces of the three media is calculated in terms of macroscopic, frequency-dependent, dielectric permittivities

$$\begin{aligned}-\Pi_A(H) = \frac{k_B T}{\pi c^3} \sum_{n=0}^{\infty}{}' \epsilon_3^{3/2} \omega_n^3 \\ \times \int_{p=1}^{\infty} p^2 \left\{ \left[\frac{(s_1+p)(s_2+p)}{(s_1-p)(s_2-p)} exp\left(\frac{2p\omega_n H}{c}\sqrt{\epsilon_3} \right) - 1 \right]^{-1} \right. \\ \left. + \left[\frac{(s_1+p\epsilon_1/\epsilon_3)(s_2+p\epsilon_2/\epsilon_3)}{(s_1-p\epsilon_1/\epsilon_3)(s_2-p\epsilon_2/\epsilon_3)} exp\left(\frac{2p\omega_n H}{c}\sqrt{\epsilon_3} \right) - 1 \right]^{-1} \right\} dp \end{aligned} \tag{3.1a}$$

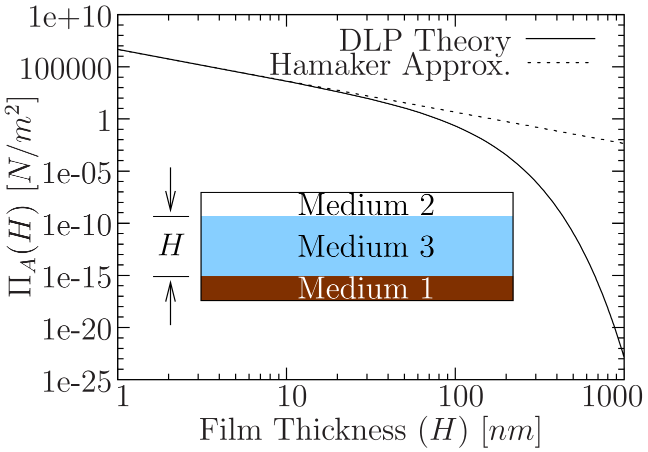


Figure 3.1: A comparison of the retarded (DLP theory) and nonretarded (Hamaker approximation) London dispersion component of the disjoining pressure for the case of type 304 stainless steel (Medium 1) and vapor (Medium 2) interacting across liquid sodium (Medium 3). The retarded disjoining pressure is calculated from Equation (3.2) while the nonretarded disjoining pressure is calculated from Equation (3.1).

where

$$s_1 = \sqrt{\epsilon_1/\epsilon_3 - 1 + p^2} \tag{3.1b}$$

$$s_2 = \sqrt{\epsilon_2/\epsilon_3 - 1 + p^2} \tag{3.1c}$$

$$\omega_n = 2\pi n k_B T/\hbar \tag{3.1d}$$

$$\epsilon = \epsilon(i\omega_n)\ . \tag{3.1e}$$

The prime notation on the summation symbol in Equation (3.1a) indicates the term with $n = 0$ is divided by half. The disjoining pressure of the liquid (3) is interpreted as the negative of the dispersion force per unit area between the planar surfaces (1 and 2). For the case of an evaporating thin film, medium 1 represents the solid surface (chosen to be type 304 stainless steel with a smooth, nonpolluted surface), medium 2 symbolizes an inert gas at standard atmospheric pressure (hereafter approximated by $\epsilon_2 = 1$), and medium 3 corresponds to the liquid (chosen to be pure, perfectly wetting sodium). Thus, when the dispersion force is positive, media 1 and 2 are attractive, the disjoining pressure is negative, and the thin film is unstable. Conversely, when the dispersion force is negative, media 1 and 2 are repulsive, the disjoining pressure is positive, and the thin film is stable and spreading.

3.1.2 Hamaker Approximation

The full DLP theory may be simplified if temperature has little effect on the interactions between the media ($H \ll c\hbar/k_B T$) and if the film thicknesses are small compared to the absorption wavelength of the participating media ($H \ll \lambda\!\!\!^{-}_{1,2,3}$). Under these restrictions, the disjoining pressure approaches an asymptotic relationship

$$\Pi_A(\delta) \approx \frac{A}{6\pi H^3} = \frac{\bar{A}}{H^3} \tag{3.2a}$$

where A stands for the macroscopic analogue to the Hamaker constant

$$A = \frac{3\hbar}{4\pi} \int\limits_{\omega=\omega_1}^{\infty} \left(\frac{\epsilon_1(i\omega) - \epsilon_3(i\omega)}{\epsilon_1(i\omega) + \epsilon_3(i\omega)}\right)\left(\frac{\epsilon_2(i\omega) - \epsilon_3(i\omega)}{\epsilon_2(i\omega) + \epsilon_3(i\omega)}\right) d\omega\,. \tag{3.2b}$$

The Hamaker approximation of the dispersion force is said to be nonretarded since any retardation effects in the time delay in the communication of electric fields between atoms may be neglected.

As stated earlier, the sign convention used with the Hamaker approximation is quite varied in the literature. Table 3.1 provides a select summary of the chronological variation in the terminology across several authors. The present work assumes the convention of a negative Hamaker constant resulting in a positive disjoining pressure for the case of a wetting film. It should be noted that Chebaro and Hallinan [41] and Chebaro *et al.* [42] used an incorrect relationship to calculate the Hamaker constant. They quoted an approximation derived by Israelachvili [52, p.147] to describe two identical metals interacting across a vacuum. This incorrect relationship was utilized and mistakenly treated as a positive disjoining pressure to fortuitously result in stable, thin film solutions.

3.1.3 Complex Dielectric Permittivities

A frequency-dependent electromagnetic field affects the complex dielectric permittivity of a given metal through the phenomenon of electromagnetic dispersion. In 1902, Drüde [73, pp.396-399] proposed an empirical model of electromagnetic dispersion that is especially suitable for the alkali metals. Drüde assumed the valence electrons in a metal could be modeled as simple harmonic oscillators. The valence electrons are considered free and unbound since they are subject to inertia and dampening forces but not a restoring force.

Table 3.1: A brief literature survey of the sign convention used for the Hamaker approximation of the dispersion force component of the disjoining pressure for the case of perfectly wetting thin films.

Author(s)	**Date**	**Disjoining Pressure**	**Hamaker Relationship**	**Hamaker Constant**
Wayner et al. [39]	1976	$P_v - P_l = -\Pi$	$\Pi \approx -\frac{\bar{A}}{H^3}$	$\bar{A} > 0$
Wayner [53]	1989	$P_v - P_l = -\frac{\bar{A}}{H^3}$	?	$\bar{A} < 0$
Wayner & Schonberg [36]	1990	$P_v - P_l = \gamma K + \Pi$	$\Pi \approx -\frac{\bar{A}}{H^3}$	$\bar{A} < 0$
Schonberg & Wayner [28]	1991	$P_v - P_l = \Pi$	$\Pi \approx -\frac{\bar{A}}{H^3}$	$\bar{A} < 0$
Wayner [71]	1991	$P_v - P_l = \Pi$	$\Pi \approx -\frac{\bar{A}}{H^3}$	?
DasGupta et al. [31]	1993a	$P_v - P_l = \gamma K - \Pi$	$\Pi \approx \frac{\bar{A}}{H^3}$	$\bar{A} < 0$
DasGupta et al. [32]	1993b	$P_v - P_l = \gamma K + \Pi$	$\Pi \approx -\frac{\bar{A}}{H^3}$	$\bar{A} < 0$
DasGupta et al. [33]	1994	$P_v - P_l = \gamma K - \Pi$	$\Pi \approx \frac{\bar{A}}{H^3}$	$\bar{A} < 0$
Schonberg et al. [34]	1995	$P_v - P_l = \gamma K + \Pi$	$\Pi \approx -\frac{\bar{A}}{H^3}$	$\bar{A} < 0$
Wayner [55]	1999	$P_v - P_l = \gamma K + \Pi$	$\Pi \approx -\frac{\bar{A}}{H^3}$	$\bar{A} < 0$
Wee et al. [22]	2005	$P_v - P_l = \gamma K + \Pi$	$\Pi \approx \frac{\bar{A}}{H^3}$	$\bar{A} > 0$

As a result, the complex dielectric permittivity is

$$\epsilon(\omega) = \epsilon'(\omega) + i\epsilon''(\omega) \tag{3.3a}$$

$$\epsilon'(\omega) = 1 - \frac{\omega_e^2 \tau^2}{1 + \omega^2 \tau^2} \tag{3.3b}$$

$$\epsilon''(\omega) = \frac{\omega_e^2 \tau}{\omega(1 + \omega^2 \tau^2)} \tag{3.3c}$$

where τ represents the relaxation time, which is related to the DC conductivity via the Lorentz-Sommerfeld relation [74]

$$\tau = m_e \sigma_0 / N_e q_e^2 \ , \tag{3.3d}$$

and ω_e symbolizes the plasma frequency of the free electron gas

$$\omega_e = \left(N_e q_e^2 / \epsilon_0 m_e \right)^{1/2} \ . \tag{3.3e}$$

Hodgson [75, pp.332-337] provides a detailed derivation and explanation of the pertinent simplifying assumptions. Above all, it should be noted that this development ignores the magnetic permeability in accordance with Maxwell's relation (i.e. $\epsilon(\omega) \approx n^2(\omega)$). Inagaki *et al.* [76] found good correlation between the Drüde Theory and experimental results for liquid Sodium at 120^0C at lower frequencies of excitation. The discrepancy at higher frequencies arises from the assumption that the dielectric permittivity is independent of the wavenumber of the incoming electromagnetic wave [77]. It is not modeled in this case for the sake of simplicity.

In reality, electrons experience influence from the positive ions in the metal as well as other electrons. The electron mass, m_e, or free electron density, N_e, are multiplied by an empirical "fudge factor" in an effort to accommodate these influences and make this extremely simplified model more closely resemble experimental data. The presence of a superscript * indicates the use of an effective value. For liquid Sodium, Shimoji [78, p.288] reported an effective valence electron number density of $N_e^*/N_e = 0.85$ at 100^0C. Inagaki *et al.* [76] reported an effective mass $m_e^*/m_e = 1.17$ at 120^0C. These empirical terms are essentially equivalent since $N_e^*/N_e = m_e/m_e^*$. In the absence of any further experimental results, we assume this value holds at the melting point of liquid sodium, as well. The plasma frequency for liquid sodium at the melting point is calculated to be $v_{e,3} = 1.0675 \times 10^{15} Hz$ using Equation (3.3e) with the effective mass and the properties listed in Table 3.2.

The solid substrate is chosen to be AISI type 304 stainless steel (SS304). To the author's best knowledge, no information exists regarding effective electron masses, relaxation times, or correlation with the Drüde theory for this alloy. In the absence of such information, the simplified Drüde model is used which assumes no dampening forces

$$\epsilon(\omega) = 1 - \left(\frac{\omega_e}{\omega}\right)^2 \tag{3.4}$$

where ω_e is the plasma frequency of the electron gas as given in Equation (3.3e). The composition is approximated as Fe (71%), Cr (19%), Ni (9%) yielding an atomic weight of 54.81 with 1.79 valence electrons per molecule and a density of 8000 kg/m^3. These values yield a plasma frequency for solid SS304 of $v_{e,1} = 3.5615 \times 10^{15} Hz$ using Equation (3.3e).

It is important to note that the DLP equation requires the three media to be modeled in terms of their respective dielectric permittivities for imaginary frequencies. This is related to the imaginary part of the

Table 3.2: Fluidic and thermodynamic properties of liquid sodium at atmospheric pressure. The evaporation coefficient of sodium was reported by Takens et al [79]. The resistivity was extrapolated from curve fits summarized by Wilson [80]. All other properties were obtained from the Argonne National Laboratory International Nuclear Safety Center Material Properties Database as reported by Fink and Leibowitz [81].

Property	**Symbol**	**Units**	**Value**
Vapor Temperature	T_v	(K)	1154.7
Molecular Weight	$\mathcal{M}$	(kg/mol)	0.02299
Density	ρ	(kg/m^3)	742.86
Dynamic Viscosity	μ	$(N \cdot s/m2)$	1.5856E-04
Surface Tension	γ	(N/m)	0.1199
$d\gamma/dT$	k	$(N/m \cdot K)$	0.0001
Thermal Conductivity	λ	$(W/m \cdot K)$	48.6562
Latent Heat of Vaporization	Δh_{fg}	(KJ/kg)	3881.5
Vapor Pressure	P_v	(MPa)	0.10133
Conductivity	σ	(S/m)	25.3605E+05
Evaporation Coefficient	α		1.0

dielectric permittivity for real frequencies through the relationship

$$\epsilon(i\omega) = 1 + \frac{2}{\pi} \int_{x=0}^{\infty} \frac{x\epsilon''(x)}{x^2 + \omega^2}\, dx \tag{3.5}$$

which was derived from the Kramers-Kronig relation using contour integrals [82, p.262]. Here, the imaginary part of the complex dielectric permittivity “is always positive and determines the dissipation of energy in an electromagnetic wave propagated in the medium” [51]. For liquid sodium, substitution of Equation (3.3c) into Equation (3.5) yields

$$\epsilon_3(i\omega) = 1 + \frac{\omega_{e,3}^2 \tau(1 - \omega\tau)}{\omega(1 - \omega^2\tau^2)} \,. \tag{3.6}$$

For the solid stainless steel substrate, Equation (3.4) does not contain a complex part. Thus, the dielectric permittivity for imaginary frequencies is

$$\epsilon_1(i\omega) = 1 + \left(\frac{\omega_{e,1}}{\omega}\right)^2 \tag{3.7}$$

using the substitution of $i\omega$ for ω. For the sodium vapor, the dielectric permittivity for imaginary frequencies is simply unity (i.e. $\epsilon_2(i\omega) = 1$).

Figure 3.1 plots the retarded dispersion force from Equation (3.1) and nonretarded dispersion force from Equation (3.2) for a liquid sodium thin film on a type 304 stainless steel plate over a range of film thicknesses using Equations (3.6) and (3.7). Equation (3.1) was solved numerically using adaptive Lobatto quadrature in the MATLAB® programming environment. The summation was carried out until a relative convergence of 1×10^{-6} was obtained. The Hamaker constant, given in Equation (3.2), was solved numerically as well using the Maple™ computer algebra environment to yield the value $A = -1.0199 \times 10^{-19} J$. The program codes are listed in Appendix A for reference.

From Figure 3.1, it is evident that, for expected thin film thicknesses, the liquid sodium system dispersion force cannot be modeled by the simplified Hamaker approximation. This is to be expected since the absorption spectra and elevated temperature of a liquid metal thin film system prohibit any simplifications to the DLP dispersion force model. Instead, the DLP theory in its full, retarded form must be used. To the author's knowledge, this has not been attempted by any research groups to date.

3.1.4 Curve Fit

A variety of numerical methods could be applied to incorporate the full, retarded form of the DLP dispersion force model into the extended meniscus thin film model. This research expresses the dispersion force curve of Equation (3.1) with a simpler function via cubic spline interpolation. Cubic spline interpolation fulfills the requirements of modeling over many orders of magnitude and second-order differentiability. It furthermore presents a piecewise continuous curve that enables the use of continuous solution schemes to the thin film equation, such as a projection method.

Consider a set of discrete data points $[x_i, y_i]$ where $i = 0, 1, 2, ..., N$ that belong to some function $y = f(x)$. This discrete set consists of $N + 1$ points with N intervals in between. Cubic spline interpolation furnishes N cubic equations to model or interpolate in between the known data points. Each segment has a respective cubic spline equation with four coefficients as

$$S_i(x) = c_{1,i}(x - x_i)^3 + c_{2,i}(x - x_i)^2 + c_{3,i}(x - x_i) + c_{4,i}, \ x \in [x_i, x_{i+1}] \, . \tag{3.8a}$$

Therefore, the total spline model contains $4N$ unknowns. Continuity through the second derivative,

$$S_i(x_i) = y_i \tag{3.8b}$$

$$S_i(x_{i+1}) = y_{i+1} \tag{3.8c}$$

$$S'_{i-1}(x_i) = S'_i(x_i) \tag{3.8d}$$

$$S''_{i-1}(x_i) = S''_i(x_i) \tag{3.8e}$$

provides $4N-2$ equations. Either the first or second derivatives at the endpoints supply the final 2 constraining equations.

A cubic spline interpolation model for the retarded dispersion force is bounded to the left by the thickness of the adsorbed film region and to the right by computer-limited round-off errors. Experimentation indicates a good model with negligible error is created when the retarded dispersion force curve for a domain $10nm \leq H \leq 771nm$ is split into 75 piecewise continuous cubic splines described by

$$\Pi_{A,i}(H) = c_{1,i}(H-H_i)^3 + c_{2,i}(H-H_i)^2 + c_{3,i}(H-H_i) + c_{4,i},\ H \in [H_i, H_{i+1}],\ \ i = 1, 2, \ldots, 75\ . \tag{3.9a}$$

The final two constraining equations are specified in the second derivative

$$\Pi''_{A,1}(H_1) = \frac{2A}{\pi H_0^5} \tag{3.9b}$$

$$\Pi''_{A,75}(H_{75}) = 0 \tag{3.9c}$$

where asymptotic analysis at the limit of a thin film gives the Hamaker approximation which can be used for the left endpoint and the right endpoint is a so-called ‘natural’ spline. The dispersion force and its derivatives are treated as negligible for film thicknesses $H > 771nm$. Figure 3.2 displays the results.

3.2 Electronic Component

Statistical quantum mechanics successfully models the thermodynamic and electrical properties of the metallic state of matter. The outermost valence electron in a metal can be well approximated as a free

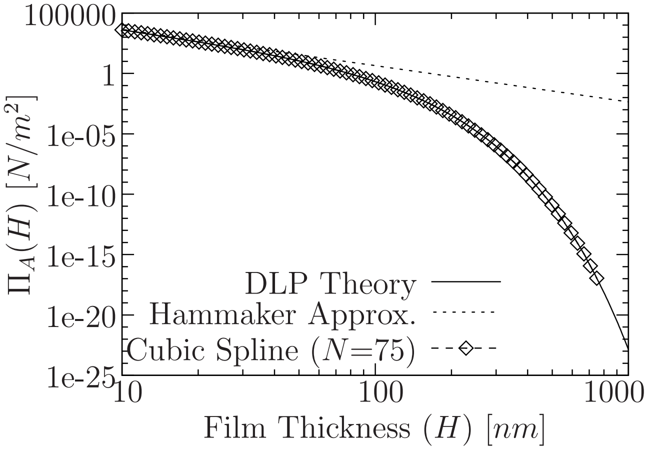


Figure 3.2: The cubic spline fit of the DLP model of the dispersion force.

electron as the presence of other electrons serves to shield it from the effects of positive ions. According to quantum mechanics, the valence electron of a metallic atom is restricted to very discrete energy levels. As more and more metallic atoms are brought together, however, the wave functions of the valence electrons overlap and the discrete energy levels broaden into apparent bands. The approximation of a continuous energy band of valence electrons in a metal facilitates the theoretical calculations of the desirable macroscopic thermodynamic and electrical properties.

We first consider a bulk, alkali metal. The alkali metals are monovalent and have low electronegativities. It is thus reasonable to assume that each atom contributes one free electron to the system. If the electrons are modeled as noninteracting particles, then Fermi-Dirac statistics can be used to model the resulting electron gas in the metal. At the limit of absolute zero temperature, a Fermi gas of free electrons will fill all available energy levels from the ground state up to the so-called Fermi energy, E_F^0. Considering the translational energy states along with electron spin degeneracy in a continuous energy band, we model the free electron gas density of states as

$$\rho_e(E) = \frac{m_e V}{\pi \hbar^2 H} \int_{n=0}^{\infty} \Theta(E - E_n) dn \tag{3.10}$$

where $\Theta(x)$ represents the Heaviside step function. The energy spectrum of the electrons corresponds to that of particles in an infinitely high potential well,

$$E_n = \frac{\pi^2\hbar^2}{2m_e H^2} n^2 \tag{3.11}$$

thus the electrons may not leave the metal system. The value of the Fermi energy is calculated from the free electron gas density of states

$$N_e = \int_{E=0}^{E_F^0} \rho_e(E) dE \tag{3.12}$$

which results in

$$E_F^0 = (3\pi^2)^{2/3} \left(\frac{\hbar^2}{2m_e}\right) \left(\frac{N_e}{V}\right)^{2/3} . \tag{3.13}$$

The Fermi energy is also the chemical potential of the electron system at absolute zero. Thus, the thermodynamic "pressure" caused by the electrons can be described by the derivative of the Gibbs thermodynamic potential per unit surface area with respect to the system thickness

$$P(H) = -\left(\frac{\partial G/\mathrm{SA}}{\partial H}\right)_T \tag{3.14}$$

and, for $V = \mathrm{SA} \cdot H$, yields

$$P = -\frac{N_e}{V} E_F^0 \tag{3.15}$$

which, for equilibrium to exist, must be balanced by the"pressure" introduced by the potential of the positive ion core.

As an aside, the assumption of absolute zero temperature seems quite inapplicable to a liquid metal evaporation system. The Fermi energy, however, proves quite insensitive to temperature. A procedure to calculate the correction to the Fermi energy for nonzero temperatures is outlined by Coutts [83, pp.30-32] and results in a series expansion of the form

$$E_F = E_F^0 \left[1 - \frac{\pi^2}{12} \left(\frac{k_B T}{E_F^0}\right)^2 - \frac{\pi^4}{80} \left(\frac{k_B T}{E_F^0}\right)^4 + \ldots \right]. \tag{3.16}$$

For a metal, the Fermi energy is of the order of magnitude $10^5 \times k_B$ which justifies the use of the absolute zero limiting case even at the boiling point of an alkali metal.

For small dimensions, such as in thin metallic films, the finiteness of the physical domain invalidates the assumption of a continuous electron energy band [58]. Derjaguin, Leonov, and Roldughin were the first to consider this effect in liquid metal thin films [48]. They described the electron gas density of states in a discrete form

$$\rho_e(E) = \frac{m_e V}{\pi^2 \hbar^2 H} \sum_n \left[\pi \Theta(E - E_n) + tan^{-1} \left(\frac{\zeta}{E_n - E} \right) \right] \tag{3.17}$$

where the last term accounts for a widening of the energy levels due to electron-electron interactions, electron-ion interactions, and impurities. They neglected higher-order terms to obtain a solution for a new Fermi energy via Equation (3.12). Now the pressure induced by the free electron gas in the thin film as defined by Equation (3.14) leads to the bulk pressure component seen in Equation (3.15) along with monotonous and exponentially decaying oscillatory contributions due to the discrete nature of the thin film. The monotonous pressure contribution is manifested in the thin film system as an excess or disjoining pressure,

$$\Pi_B(H) \approx \frac{B}{H^2}; \;\; B = \frac{\hbar^2}{2m_e} \frac{N_e}{V} \frac{3\pi^2}{16} \tag{3.18}$$

which is proportional to the square of the film thickness.

Shortly thereafter, Derjaguin and Roldughin [57] derived the electron gas density of state for a more complicated case where surface effects are considered. The surface effects are manifested in the boundary conditions for the electronic wave function at the top and bottom boundaries of the liquid metal thin film. Following the same procedure as outlined above and assuming symmetrical boundary conditions, Derjaguin and Roldughin derived a more general form of the electronic component of the disjoining pressure

$$\Pi_B(H) \approx \frac{B}{H^2}; \;\; B = \frac{\hbar^2}{2m} \frac{N}{V} \chi(\kappa_n) \tag{3.19}$$

where the parameter, χ, reflects a dependence upon the boundary conditions of the system. Roldughin [59] described κ_n as “the distance by which electrons are ‘allowed’ to go out into the external environment.” In so doing, they reduce the energy density of the fermion gas and lower the electronic disjoining pressure. As shown below, κ_n is closely related to the work function, W, or energy needed to move an electron from the liquid metal to the solid surface

$$\chi(\kappa_n) = \Sigma_1 \Sigma_2 - \frac{1}{4} \Sigma_1^2 \tag{3.20a}$$

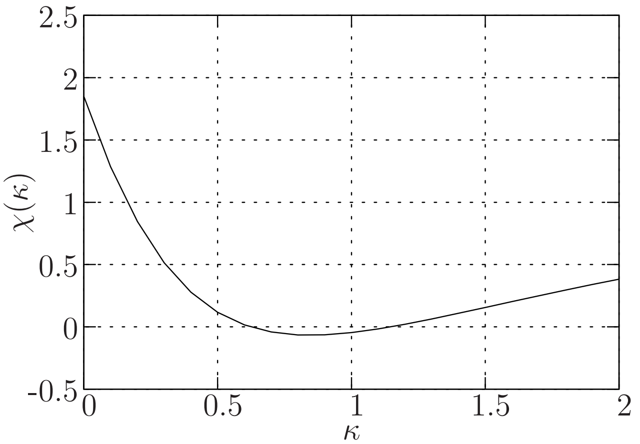


Figure 3.3: Dependence of $\chi(\kappa_n)$ on the work function related parameter, κ_n, as related via Equation (3.20). This function determines the boundary condition for Derjaguin's electronic component of the disjoining pressure.

where

$$\Sigma_1 = \frac{\pi}{2} - 2\left[tan^{-1}(\kappa_n) + \kappa_n - \kappa_n^2 tan^{-1}\left(\frac{1}{\kappa_n}\right)\right] \tag{3.20b}$$

$$\Sigma_2 = \frac{\pi}{2} - 2tan^{-1}(\kappa_n) \tag{3.20c}$$

$$\kappa_n \approx \sqrt{1 + \frac{W}{E_F}}\,. \tag{3.20d}$$

Figure 3.3 displays this function in graphical form. Depending on the work function of the system, the electronic disjoining pressure can vary in intensity and even become negative, resulting in an unstable film. The minimum value $\chi = -0.066873$ occurs at $\kappa_n = 0.844664$. Also, as κ_n approaches the limits of zero and infinity, the adjusting parameter χ approaches the same limit of $3\pi^2/16$. The infinite limit represents the simplified assumption of an infinitely deep potential pit at the liquid/surface boundary that prevents electrons from emerging from the film as seen in Equation (3.18).

Knowledge of the proper work function for a given system proves intractable at the present as this boundary condition depends heavily on a quantum mechanical description of the system that is intimately tied to the surface conditions between the solid and liquid. Instead, we seek to define the range of values which the work function might take in a liquid sodium thin film system. The constant part of B is calculated from Equation (3.19) at the melting point of liquid sodium.

$$B = \frac{\hbar^2}{2m_e}\frac{N_e}{V}\chi(\kappa_n) = 1.1873 \times 10^{-10} \cdot \chi(\kappa_n) \tag{3.21}$$

The boundary condition parameter will vary within a set range ($-0.066873 \leq \chi(\kappa_n) \leq 1.850551$) as shown in Figure 3.3. This yields a range of possible values for B ($-6.837223 \times 10^{-12}\ N \leq B \leq 1.892039 \times 10^{-10}\ N$), which a parametric study should include.

Chapter 4

Numerical Analysis Techniques

4.1 Runge-Kutta Method

Chebaro and Hallinan [41] and Chebaro *et al.* [42] solved the thin film equation with an explicit Runge-Kutta numerical solution procedure. The one far-field boundary condition, seen in Equation (2.30d), is met with a shooting method. Since negligible curvature exists in the interline region, the governing equation reduces to a second-order ODE that is solved with slight perturbations in the independent variable and its first derivative. The endpoints of this solution then become the boundary conditions for the full, fourth-order ODE. The missing far-field boundary condition is satisfied with the shooting method whereby the second derivative near-field boundary condition is iterated upon until the solution approaches an asymptotic value in the far-field equal to the curvature of the bulk meniscus (the inverse of a simulated pore radius).

4.2 Orthogonal Collocation Method

Orthogonal collocation [84], a subset of the weighted residual method, provides a more favorable numerical solution scheme for the problem at hand in that it:

(i) is a fully implicit form of the Runge-Kutta method [85, 86] and thereby eliminates the need for the shooting method,

(ii) results in a continuous approximation to the solution (consisting of a series expansion of a basis function operated on by a collocation coefficient) that can later be analytically manipulated for post-processing needs,

(iii) readily handles the possible stiffness problems associated with the high degrees of nonlinearity inherent in the problem, and

(iv) permits incorporation of the cubic spline interpolation model of the dispersion force.

First we map the domain from $\eta \in [0, l_{max}]$ to $\xi \in [-1, 1]$ which later enables easy implementation of the Chebyshev polynomial basis function. A linear transformation accomplishes this task by letting $\hat{\theta}(\xi) = \theta\left(\phi(1+\xi)\right)$ where $\phi = l_{max}/2$. After applying the chain rule, Equation (2.30) can be rewritten as

$$\frac{3}{\phi^4}\hat{\theta}(\xi)^2\hat{\theta}(\xi)'\hat{\theta}(\xi)''' + \frac{1}{\phi^4}\hat{\theta}(\xi)^3\hat{\theta}(\xi)'''' + \frac{3}{\phi}\hat{\theta}(\xi)^2\hat{\theta}(\xi)'\Pi^*(\hat{\theta})' + \hat{\theta}(\xi)^3\Pi^*(\hat{\theta})'' + \frac{3Ca}{\left(\frac{H_0\Pi_0}{\gamma}\right)^2}\left[T^* - \frac{1}{\phi^2}\hat{\theta}(\xi)'' - \Pi^*(\hat{\theta})\right] = 0, \quad \xi \in [-1, 1] \tag{4.1a}$$

where

$$\hat{\theta}(-1) = a_1 \tag{4.1b}$$

$$\hat{\theta}'(-1) = \phi a_2 \tag{4.1c}$$

$$\hat{\theta}''(1) = \phi^2 a_3 \tag{4.1d}$$

$$\hat{\theta}'''(-1) = 0 \tag{4.1e}$$

which is amenable to the desired numerical solution scheme.

The disjoining pressure is treated as a linear combination of the dispersion force and electronic components. Thus, Equations (3.9) and (3.19) are mapped into the new Chebyshev polynomial friendly domain and added to yield

$$\begin{aligned}\Pi^*(\hat{\theta}) &= \frac{\Pi_{A,i}(\hat{\theta})}{\Pi_0} + \frac{\Pi_B(\hat{\theta})}{\Pi_0} \\ &= \frac{c_{1,i}H_0^3}{\Pi_0}(\hat{\theta}-\hat{\theta}_i)^3 + \frac{c_{2,i}H_0^2}{\Pi_0}(\hat{\theta}-\hat{\theta}_i)^2 + \frac{c_{3,i}H_0}{\Pi_0}(\hat{\theta}-\hat{\theta}_i) + \frac{c_{4,i}}{\Pi_0} \\ &+ \frac{B\chi(\kappa_n)}{H_0^2\Pi_0}\frac{1}{\hat{\theta}^2}, \\ &\hat{\theta} \in [\hat{\theta}_i, \hat{\theta}_{i+1}], \quad i = 1, 2, \ldots, 75 \ ,\end{aligned} \tag{4.1f}$$

which consists of 75 different equations due to the cubic spline interpolation of the dispersion force.

With the problem defined in a Chebyshev polynomial-friendly domain, we seek an analytical solution using the Ansatz

$$\hat{\theta}(\xi) = \sum_{m=0}^{\infty} r_m T_m(\xi),\ \xi \in (-1, 1) \tag{4.2}$$

where $\{T_m(\xi)\}_{m=0}^{\infty}$ represents Chebyshev polynomials of the first kind. These were chosen over monomials as an othogonal basis set due to their demonstrated ability to remain numerically independent at higher orders [84]. Chebyshev polynomials of the first kind [87, 88] are given with the recursive relationship

$$T_0(\xi) = 1 \tag{4.3}$$

$$T_1(\xi) = \xi \tag{4.4}$$

$$T_{m+1}(\xi) = 2\xi T_m(\xi) - T_{m-1}(\xi) \tag{4.5}$$

and form an orthogonal basis set per

$$\int_{\xi=-1}^{1} \frac{T_m(\xi)T_n(\xi)d\xi}{\sqrt{1-\xi^2}} \tag{4.6}$$

where $w(x) = 1/\sqrt{1-\xi^2}$ is the weight function. Thus, the need to utilize the computational domain $\xi \in (-1, 1)$ is realized.

The approximate analytical solution is obtained by truncating the infinite series of the Ansatz to $N + 3$ terms such that

$$\hat{\theta}(\xi) \approx \hat{\Theta}_{N+3}(\xi) = \sum_{m=0}^{N+3} r_m T_m(\xi),\ \xi \in (-1, 1)\ . \tag{4.7}$$

The first four terms in the expansion of Equation (4.7) are obtained explicitly by enforcing the four boundary conditions specified in Equation (4.1). Like terms are gathered, the summation is re-indexed, and we formally present the approximate analytical solution as a linear combination

$$\hat{\theta}(\xi) \approx \hat{\theta}_N(\xi) = \hat{\Psi}(\xi) + \sum_{k=1}^{N} q_k^N \Psi_k(\xi), \ \xi \in [-1, 1] \tag{4.8a}$$

where

$$\hat{\Psi}(\xi) = a_1 + \phi(\xi + 1)a_2 + \phi^2 \left(\frac{\xi^2}{2} + \xi + \frac{1}{2} \right) a_3 \tag{4.8b}$$

satisfies the inhomogeneous, linear, boundary conditions while

$$\begin{aligned} \Psi(\xi) = {} & T_k(\xi) - T_k(-1) - (\xi + 1)T_k'(-1) \\ & - \left(\frac{\xi^2}{2} + \xi + \frac{1}{2} \right) T_k''(1) + \frac{1}{6} \left(-\xi^3 + 3\xi^2 + 9\xi + 5 \right) T_k'''(-1) \end{aligned} \tag{4.8c}$$

satisfies the original problem statement with homogeneous boundary conditions. This representation uses a reformulated subscript such that $m = k + 3$. Thus the k represents N integers and the reason for the earlier truncation to $N + 3$ terms becomes clear. Equations for the derivatives of the approximate analytical solution are found by differentiating Equation (4.8).

Since the series truncation produces an *approximate* analytical solution, Equation (4.8) will not fully satisfy Equation (4.1). Instead, we introduce the local residual function $R_N \left(\hat{\theta}_N(\xi) \right)$ to satisfy the problem statement such that

$$R_N \left(\hat{\theta}_N(\xi) \right) + M[\hat{\theta}_N(\xi)] + g = 0, \ \xi \in [-1, 1] \, . \tag{4.9}$$

where $M[\cdot]$ indicates a nonlinear ODE operator and g indicates forcing data as a representation of Equation (4.1). The definition of the residual function becomes

$$R_N \left(\hat{\theta}_N(\xi) \right) = -M[\hat{\theta}_N(\xi)] - g, \ \xi \in [-1, 1] \, . \tag{4.10}$$

The collocation method minimizes the local residual function and determines the expansion coefficients $\left\{q_k^N\right\}_{k=1}^N$ by way of the Sifting property [89]

$$\left\langle R_N\left(\hat{\theta}_N(\xi)\right), \delta(\xi-\xi_j)\right\rangle_1 = \left\langle -M[\hat{\theta}_N(\xi)] - g, \delta(\xi-\xi_j)\right\rangle_1 = 0, \quad j = 1, 2, 3, \ldots, N \tag{4.11a}$$

where the brackets follow Dirac's notation [90, pp.18-22] and denote an inner product of the residual function with a Dirac delta test function with unity weight. As a result, the residual function is evaluated at each of the collocation points to produce a series of algebraic equations

$$R_N\left(\hat{\theta}_N(\xi_j)\right) = 0 \tag{4.11b}$$

or

$$M[\hat{\theta}_N(\xi_j)] + g = 0 \tag{4.11c}$$

where the collocation points are defined by the standard Chebyshev-Gauss-Lobatto open rule formula [91]

$$\xi_j = cos\left(\frac{2j-1}{2N}\pi\right), \; j = 1, 2, \ldots, N\,. \tag{4.12}$$

The nonlinearities in the residual operator $M[\cdot]$ must be treated before we employ a linear equation solver to obtain the collocation coefficients $\left\{q_k^N\right\}_{k=1}^N$.

4.3 Spatial Convergence Accuracy

Spatial convergence is analyzed *a posteriori* by integrating the approximate analytical solution over the domain of interest. Thus,

$$\int\limits_{\xi=-1}^{1} \hat{\theta}_N(\xi)d\xi = \int\limits_{\xi=-1}^{1} \hat{\Psi}(\xi)d\xi + \sum_{k=1}^{N} q_k^N \int\limits_{\xi=-1}^{1} \Psi_k(\xi)d\xi, \;\; \xi \in [-1, 1] \tag{4.13a}$$

where

$$\int\limits_{\xi=-1}^{1} \hat{\Psi}(\xi)d\xi = 2a_1 + 2\phi a_2 + \frac{4}{3}\phi^2 a_3 \tag{4.13b}$$

$$\int\limits_{\xi=-1}^{1} \Psi_k(\xi)d\xi = \int\limits_{\xi=-1}^{1} T_k(\xi)d\xi - 2T_k(-1) - 2T'_k(-1) - \frac{5}{3}T''_k(1) + 2T'''_k(-1) \tag{4.13c}$$

and

$$\int\limits_{\xi=-1}^{1} T_k(\xi)d\xi = \begin{cases} 0 & \text{if } k = odd, \\ \frac{-2}{(k+1)(k-1)} & \text{if } k = even \enspace . \end{cases} \tag{4.13d}$$

To observe spatial convergence, we increment the number of terms N in the approximate analytical solution, integrate the solution over the domain space using Equation (4.13), and observe the relative error between successive increments.

4.4 Nonlinear Solver: Newton-Raphson Method

The method of orthogonal collocation delivers a set of simultaneous nonlinear equations to solve. The final step towards a solution linearizes these equations for solution with common linear matrix solution routines. First, we represent the residual function as a function of the unknown collocation coefficients

$$f_j\left(\left\{q_k^N\right\}_{k=1}^N\right) = R_N\left(\hat{\theta}_N(\xi_j)\right) = 0 \tag{4.14}$$

which can be expanded in a linear, multivariable Taylor series polynomial about the roots [84]

$$\begin{aligned} f_j\left(\left\{q_k^N\right\}_{k=1}^N\right) = 0 \approx f_j&\left(\left\{\hat{q}_k^N\right\}_{k=1}^N\right) \\ &+ \frac{\partial f_j}{\partial c_1^N}\left(\left\{q_k^N\right\}_{k=1}^N\right)\Bigg|_{\left\{\hat{q}_k^N\right\}_{k=1}^N} (q_1^N - \hat{q}_1^N) \\ &+ \frac{\partial f_j}{\partial c_2^N}\left(\left\{q_k^N\right\}_{k=1}^N\right)\Bigg|_{\left\{\hat{q}_k^N\right\}_{k=1}^N} (q_2^N - \hat{q}_2^N) \\ &+ \ldots \\ &+ \frac{\partial f_j}{\partial c_N^N}\left(\left\{q_k^N\right\}_{k=1}^N\right)\Bigg|_{\left\{\hat{q}_k^N\right\}_{k=1}^N} (q_N^N - \hat{q}_N^N) \\ &+ H.O.T., \quad j = 1, 2, \ldots, N \end{aligned} \tag{4.15}$$

where it is understood that the set $\left\{q_k^N\right\}_{k=1}^N$ represents the unknown being resolved at the fixed iterate p and that the set $\left\{\hat{q}_k^N\right\}_{k=1}^N$ contains known values obtained from the previous iterate, $p-1$. Here, H.O.T. represents higher-order terms. The first-order terms of the Taylor series creates a linear matrix equation at each iterate p, namely

$$J\mathbf{q} = -\mathbf{F} \tag{4.16a}$$

where $\mathbf{q} = \left[(q_1^N - \hat{q}_1^N), (q_2^N - \hat{q}_2^N), \ldots, (q_N^N - \hat{q}_N^N)\right]^T$ contains the unknown collocation coefficients of interest, $\mathbf{F} = \left[f_1\left(\left\{\hat{q}_k^N\right\}_{k=1}^N\right), f_2\left(\left\{\hat{q}_k^N\right\}_{k=1}^N\right), \ldots, f_N\left(\left\{\hat{q}_k^N\right\}_{k=1}^N\right)\right]^T$ is a known forcing vector and J represents the Jacobian matrix whose terms are given by

$$\begin{gathered} J_{jm} = \frac{\partial f_j}{\partial c_m^N}\left(\left\{q_k^N\right\}_{k=1}^N\right)\bigg|_{\left\{\hat{q}_k^N\right\}_{k=1}^N} , \\ j = 1, 2, \ldots, N,\ m = 1, 2, \ldots, N\, . \end{gathered} \tag{4.16b}$$

4.5 Iterative Convergence Accuracy

The L_2 norm of the residual measures the iterative convergence accuracy of successive solutions, namely

$$Q = \|R_N\|_2 = \sqrt{(J\mathbf{q} + \mathbf{F}) \cdot (J\mathbf{q} + \mathbf{F})}\ . \tag{4.17}$$

An end to the iterative solution process is defined once the relative error between successive residual norms falls below a desired tolerance threshold of 1×10^{-8}.

4.6 Nonlinear Solver: Levenberg-Marquardt Method

The stiffness associated with a highly non-linear ODE can cause the Newton-Raphson approach to flounder as the Jacobian becomes poorly conditioned. In such a case the method of steepest descent can be used, albeit with extremely slow convergence. The Levenberg-Marquardt method [92, 93, pp.202-204] performs an optimum interpolation between the Newton-Raphson method and the method of steepest descent (or gradient) method

$$(G + tI)\mathbf{q} = -\nabla\mathbf{Q} \tag{4.18}$$

where $G = 2J^T J$, t is a scalar step-size parameter, I is the identity matrix, and $\nabla\mathbf{Q} = -2J^T\mathbf{F}$. When $t = 0$, Equation (4.18) represents the Newton-Raphson method of Equation (4.16). On the other hand, when $t \to \infty$, Equation (4.18) resembles the method of steepest descent. The solution procedure to obtain the set of collocation coefficients is then

(i) initially guess $\bar{q} = \{0\}$,

(ii) compute $Q^{(p)}$ using Equation (4.17),

(iii) pick an arbitrary t (e.g., $t = 0.001$),

(iv) † solve Equation (4.18),

(v) compute new $Q^{(p+1)}$ using Equation (4.17),

(vi) if $Q^{(p+1)} \geq Q^{(p)}$ then let $t = 10t$ and go to †,

(vii) else, if $Q^{(p+1)} < Q^{(p)}$ then let $t = t/10$ and go to †,

(viii) stop when $Q^{(p+1)} < Q^{(p)}$ and $Q^{(p+1)} < tol$.

Chapter 5

Bulk Meniscus Modeling

5.1 Finite Element CFD Model

This research distinguishes itself from previous works as it models multiscale liquid metal capillary evaporation with a nonisothermal interface and non-equilibrium meniscus evaporation. The continuity and momentum equations for the bulk domain in cylindrical coordinates are

$$\frac{1}{r}\frac{\partial}{\partial r}(ru) + \frac{\partial v}{\partial z} = 0 \tag{5.1a}$$

$$\rho\left(u\frac{\partial u}{\partial r} + v\frac{\partial u}{\partial z}\right) = -\frac{\partial p}{\partial r} + \mu\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial u}{\partial r}\right) + \frac{\partial^2 u}{\partial z^2} - \frac{u}{r^2}\right] \tag{5.1b}$$

$$\rho\left(u\frac{\partial v}{\partial r} + v\frac{\partial v}{\partial z}\right) = -\frac{\partial p}{\partial z} + \mu\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial v}{\partial r}\right) + \frac{\partial^2 v}{\partial z^2}\right] \tag{5.1c}$$

for an incompressible fluid with constant density and viscosity. Buoyancy forces are considered to be negligible for the micro-geometries to be considered in this research. The energy equation is

$$\rho c_p\left(u\frac{\partial T}{\partial r} + v\frac{\partial T}{\partial z}\right) = \lambda\left[\frac{1}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right) + \frac{\partial^2 T}{\partial z^2}\right] \tag{5.1d}$$

assumes constant density, specific heat, and thermal conductivity which is consistent with the small temperature overheats used in the thin film solutions.

Figure 5.1 details the full problem geometry and boundary conditions. The right side represents the

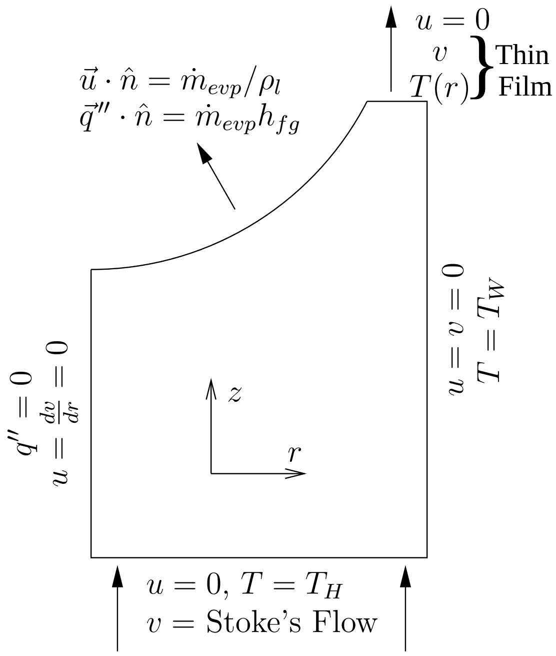


Figure 5.1: A schematic of the bulk evaporating capillary meniscus CFD geometry and boundary conditions.

capillary wall with no-slip velocity and constant temperature boundary conditions. The left side represents the symmetry line through the center of the capillary. As such, the slip/symmetry boundary conditions are utilized in the momentum equations, and the energy equation boundary condition is adiabatic. The top surface represents the outflow boundary, namely the evaporating bulk meniscus. At the surface of the evaporating bulk meniscus, the velocity and heat flux are specified as functions of surface temperature via the Hertz-Knudsen-Schrage (HKS) relationship, Equation (2.2) which describes the net evaporative mass flux under non-equilibrium conditions [67, p.346].

The curvature of the meniscus is considered constant (per previous developments) and inversely proportional to the radius of the capillary tube. The bulk meniscus profile is thus given by

$$z(r) = \frac{r^2}{2R} + \frac{(R - H_{tf})^2}{2R}, \quad r \in [0, R - H_{tf}] \tag{5.2}$$

where, at the capillary centerline, the meniscus slope is considered to be zero. Instead of approaching the wall and creating a singularity condition, the bulk meniscus is ended at a point ($r = R - H_{tf}$) that matches the far-field solution of the thin film, extended meniscus evaporation model presented previously. This interface must match with the thin film model in thickness, mass flow, and temperature/heat flux. The mass flow boundary condition is met by establishing a uniform outflow velocity over the thin film thickness that equals the total evaporative mass flow of the evaporating thin film. This is consistent with the assumption of lubrication theory fluid flow that was used to construct the extended meniscus thin film model. Similarly, a 1D conduction model was assumed to model heat transfer through the extended meniscus thin film. Thus, the temperature is specified using 1D conduction

$$T(r) = T_w - \left(\frac{\dot{m}''_{evp} h_{fg}}{\lambda} \right) (R - r) \tag{5.3}$$

where x is the distance along the interface between the bulk meniscus and the extended meniscus thin film.

Finally, the bottom surface of the capillary represents the inflow boundary. Fully developed flow is assumed, thus the velocity profile is that of Stoke's Flow and the flow temperature is that of the wall. The velocity profile is scaled to conserve mass according to the specified outflow conditions along the meniscus interface to ensure that the meniscus boundary remains static in space and time.

Chapter 6

Results

6.1 High Temperature, Liquid Metal, Extended Meniscus, Evaporation

The liquid metal thin film system described by Equation (4.1) contains five basic parameters:

1. the ODE boundary condition at $\hat{\theta}(-1)$,
2. the ODE boundary condition at $\hat{\theta}'(-1)$,
3. the ODE boundary condition at $\hat{\theta}''(1)$,
4. the liquid overheat ΔT, and
5. the disjoining pressure electronic component boundary condition $\chi(\kappa_n)$.

As explained previously, the first two ODE boundary conditions are nonzero to avoid a trivial solution and are tenuously related to physical characteristics of the system. They are thus considered as constants for this study, which reduces a parametric study to three variables. The size of the pore drives the second derivative boundary condition such that $\hat{\theta}''(1) = K = 1/R$. The applied heat flux to the system controls the liquid overheat ΔT. Finally, the disjoining pressure electronic component boundary condition $\chi(\kappa_n)$ sets the magnitude of the disjoining pressure as well as the relative importance of the dispersion force compared to the electronic force components.

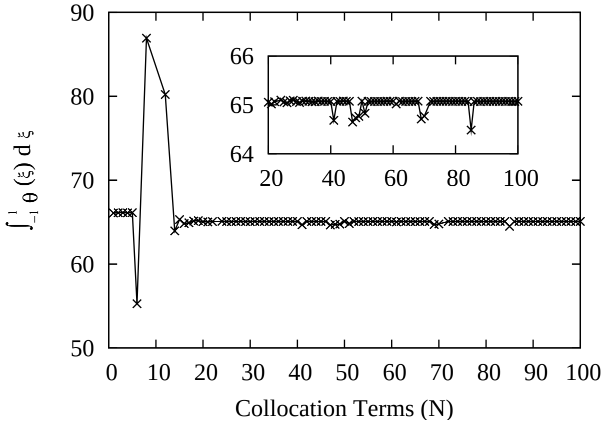
Figures 6.1, 6.2, and 6.3 demonstrate the spatial [Equation (4.13)] and iterative [Equation (4.17)] convergences of solutions to the steady extended meniscus evaporation model [Equation (4.1)] using an Ansatz

[Equation (4.8)] with increasing terms via orthogonal collocation as solved by the Levenberg-Marquardt Method. In Figure 6.1, the electronic component of the disjoining pressure is negligible and Equation (4.1) is simplified such that only the dispersion force component is included in the numerical solution using cubic spline interpolation. In Figure 6.2, the disjoining pressure is considered negligible compared to the electronic component, and Equation (4.1) is simplified such that only the electronic component is included in the numerical solution. The slight but constant decrease in the spatial convergence after approximately 70 collocation terms is deemed to be the result of numerical round-off due to computational limitations. In Figure 6.3, both the dispersion force and electronic components of the disjoining pressure are equal in magnitude and Equation (4.1) is unmodified in the numerical solution. All three scenarios reveal that extending the approximate analytical solution to 100 terms provides an accurate, converged solution.

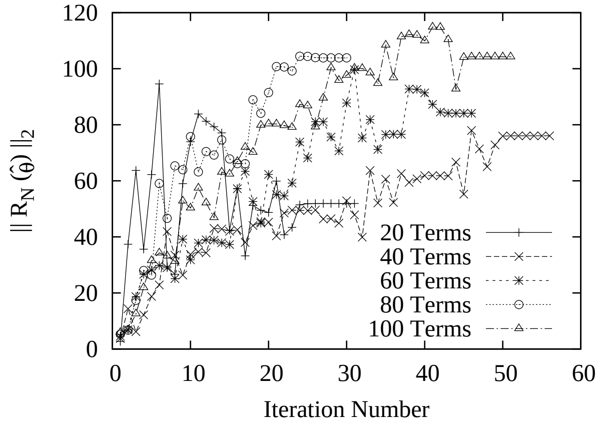
6.1.1 Effect of Liquid Overheat

Figure 6.4 illustrates the solutions to the extended meniscus evaporation model for varying liquid overheats. The boundary condition of the electronic component of the disjoining pressure is modeled at the limiting condition of an infinite potential energy well ($\chi(\kappa_n) = 3\pi/16$). The dispersion force component is negligible and removed from the governing equations. The pore radius is $200\mu m$. The liquid overheat varies from $0.0003K$ to $0.0007K$ which is analogous to an increasing, applied heat flux to the substrate.

Hallinan *et al.* [43] analyzed solutions for increasing heat flux in a similar thin film model. Different physical constants were used compared to the current study, however, and the disjoining pressure was modeled by the Hamaker approximation. Thus, only a qualitative comparison is available between the two. As expected, Figure 6.4 shows that increasing the heat flux to the system results in an increased evaporative mass flux associated with an increased liquid pressure gradient to supply liquid from the bulk meniscus. Also, both models predict a reduction in the adsorbed film thickness as the heat flux increases. In contrast the liquid metal extended meniscus evaporation model predicts that increases in the heat flux result in an associated increase in the film length. This makes sense as an increase in film length, and thus film area, would accommodate the required increase in net evaporative mass flux.

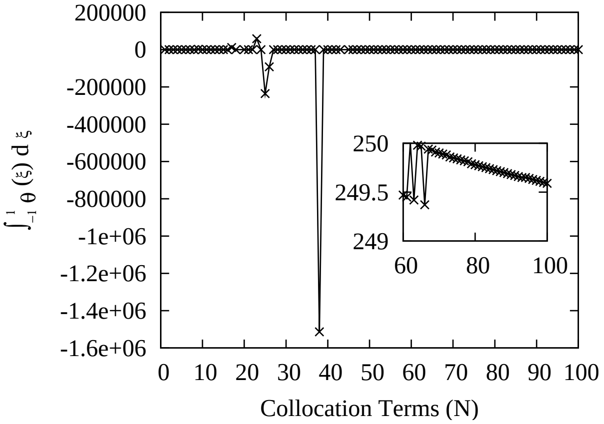


(a) Spatial convergence accuracy.

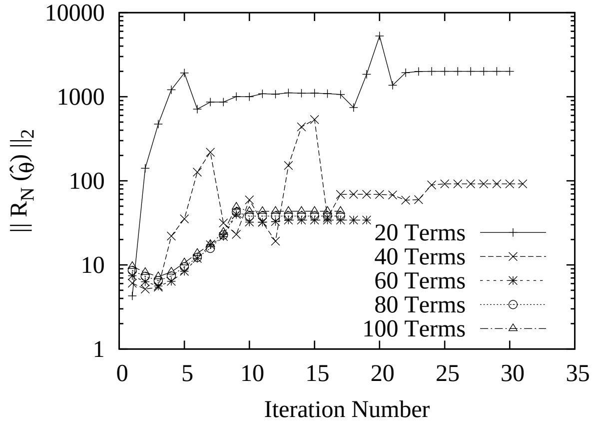


(b) Iterative convergence accuracy.

Figure 6.1: A sample graph demonstrating spatial and iterative convergence of solutions to the steady extended meniscus evaporation model [Equation (4.1)] using the Ansatz [Equation (4.8)] with increasing terms via orthogonal collocation with the Levenberg-Marquardt Method. For this case, $R = 200\mu m$, $\Delta T = 0.0005K$, and $\chi(\kappa_n) = 0$ (Case F: $\Pi_B/\Pi_A = 0$).

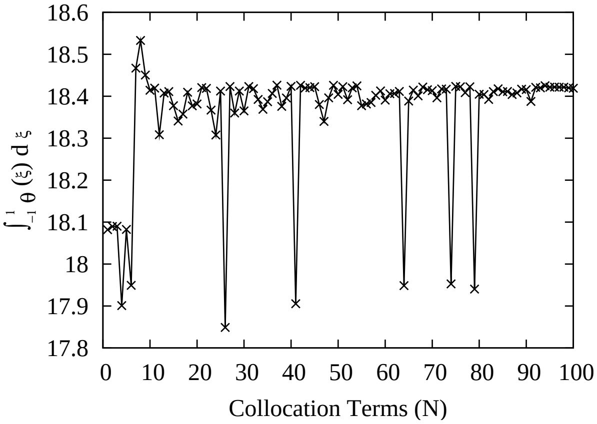


(a) Spatial convergence accuracy.

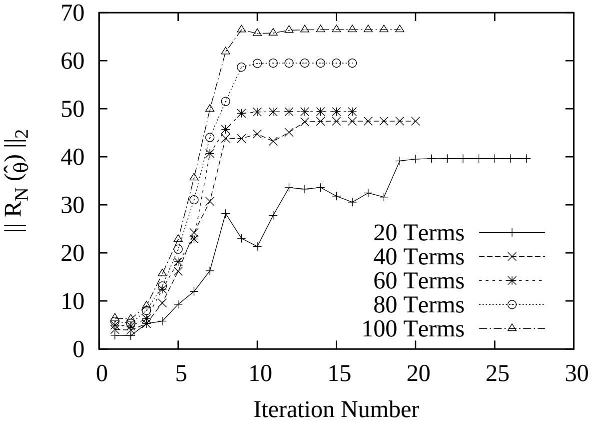


(b) Iterative convergence accuracy.

Figure 6.2: A sample graph demonstrating spatial and iterative convergence of solutions to the steady extended meniscus evaporation model [Equation (4.1)] using the Ansatz [Equation (4.8)] with increasing terms via orthogonal collocation with the Levenberg-Marquardt Method. For this case, $R = 200\mu m$, $\Delta T = 0.0005K$, and $\chi(\kappa_n) = 3\pi/16$ (Case A: $\Pi_B/\Pi_A \gg 1$).



(a) Spatial convergence accuracy.



(b) Iterative convergence accuracy.

Figure 6.3: A sample graph demonstrating spatial and iterative convergence of solutions to the steady extended meniscus evaporation model [Equation (4.1)] using the Ansatz [Equation (4.8)] with increasing terms via orthogonal collocation with the Levenberg-Marquardt Method. For this case, $R = 200\mu m$, $\Delta T = 0.0005K$, and $\chi(\kappa_n) = 0.001650$ (Case E: $\Pi_B/\Pi_A = 1.0$).

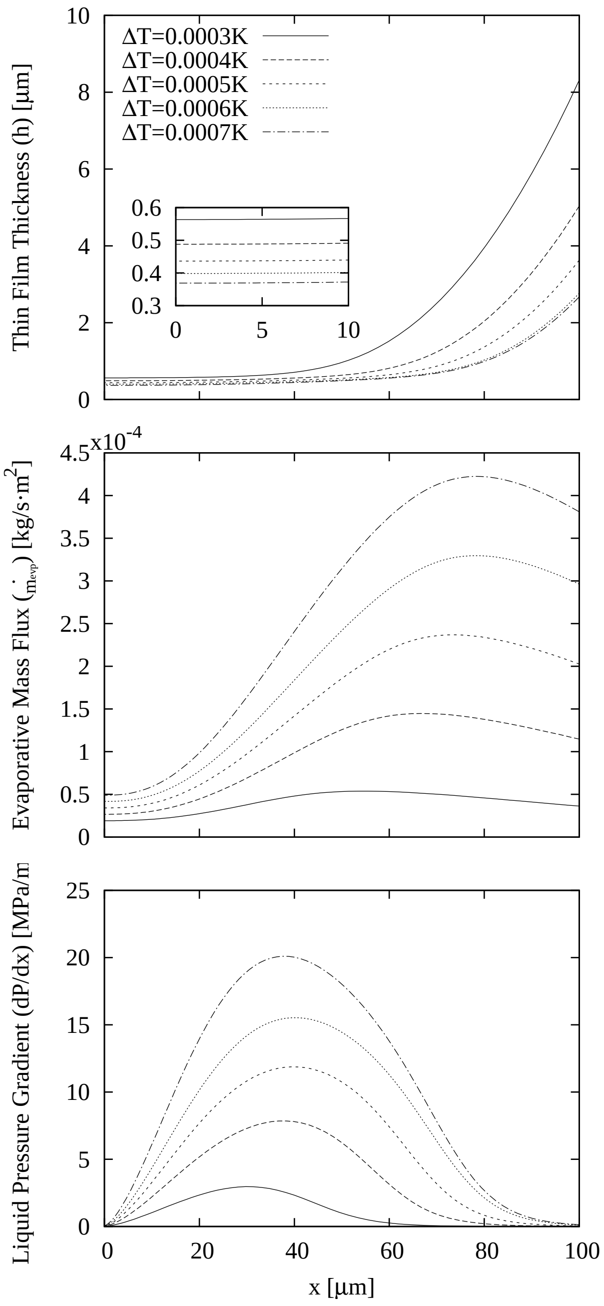


Figure 6.4: Steady thin-film evaporation solutions for $R = 200\mu m$ and $\chi(\kappa_n) = 3\pi/16$ over a range of possible liquid overheats.

6.1.2 Effect of Pore Radius

To study the effects of the pore radius on the steady liquid metal extended meniscus evaporation model, the liquid overheat is held constant at $0.0005K$. The electronic component of the disjoining pressure is treated as previously while the dispersion force component is considered negligible and removed from the mathematical model. The pore radius varies from $200\mu m$ to $500\mu m$. Figure 6.5 shows the results. As expected, the pore radius affects the second-order boundary condition through the simplified curvature assumption. Thus, the adsorbed film thickness stays constant while, for increasing pore radii, the evaporating film lengthens. The evaporative mass flux correspondingly increases, and the liquid pressure gradient broadens to sustain the film.

The simplifying assumption of negligible circumferential curvature prohibits consideration of pore radii less than $200\mu m$. Hallinan *et al.* [43] previously observed that, for this assumption to hold true, the circumferential curvature must have a negligible effect on both the flow-field in the thin film as well as the adsorbed film thickness. The latter requirement constrains the most and yields

$$R >> \frac{\gamma T_v}{\rho h_{fg} \Delta T} . \tag{6.1}$$

The smallest liquid overheat $\Delta T = 0.0003K$, corresponds to the largest constraining radius $R \leq 160\mu m$.

6.1.3 Effect of Electronic Disjoining Pressure Component

Table 6.1 lists the parametric variations of the electronic disjoining pressure Π_B resulting from a variation of the system work function boundary condition $\chi(\kappa_n)$ for a given liquid overheat, $\Delta T = 0.0005K$, and pore radius, $R = 200\mu m$. In each case, adsorbed film thicknesses, H_0, are solved using Equation (2.17) when the mass flux is zero and curvature is negligible.

Case A represents the upper limit to the electronic disjoining pressure boundary condition ($3\pi^2/16$) as previously described. It, along with Cases B-D, result in a system where the electronic disjoining pressure (Π_B) dominates over the retarded dispersion component (Π_A). The London dispersion component of the disjoining pressure is negligible and can be removed from the governing equation. In cases A-D, the electrons barely penetrate into the substrate which keeps the electron "pressure" high.

In Case E, the system work function boundary condition parameter $\chi(\kappa_n)$ is chosen such that both com-

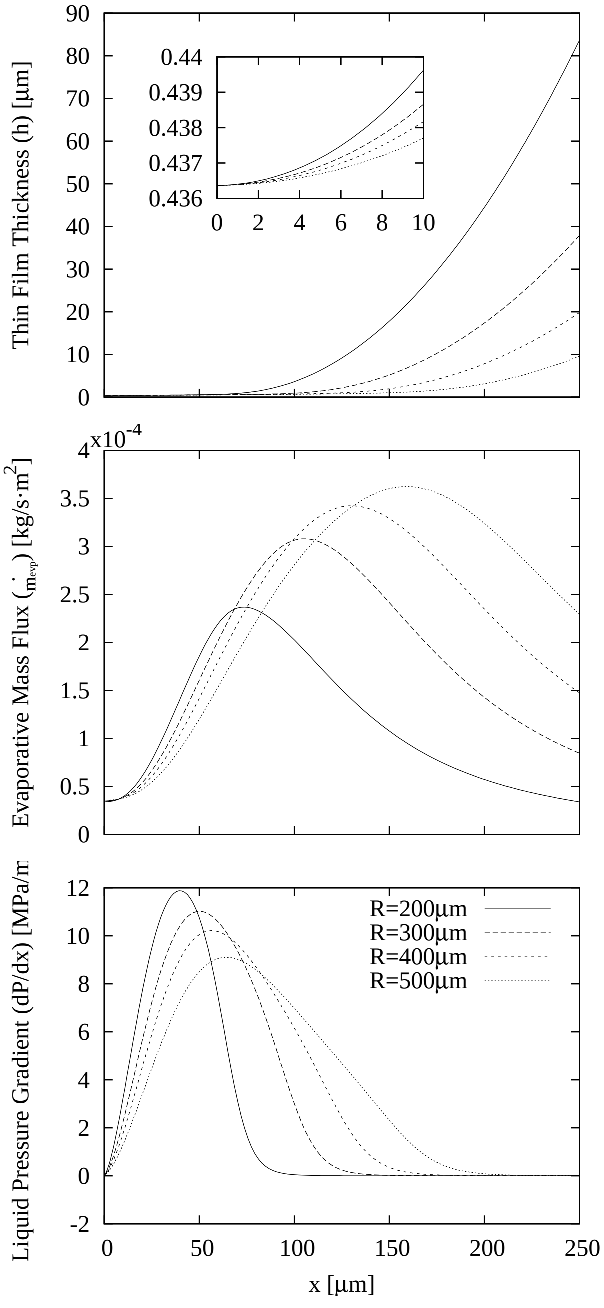


Figure 6.5: Steady thin-film evaporation solutions for $\Delta T = 0.0005K$ and $\chi(\kappa_n) = 3\pi/16$ over a range of possible intrinsic meniscus radii.

Table 6.1: Possible variations in the boundary condition term of Derjaguin's electronic component of the disjoining pressure and its effect on the adsorbed film thickness as well as the scaling of the nondimensionalized liquid metal thin film equation.

CASE	κ_n	$\chi(\kappa_n)$	$B[N]$	$H_0[nm]$	Π_B/Π_A
A	0,∞	1.850551	2.198E-10	419.59	2.172E+10
B	0.059, 11.366	1.5	1.782E-10	377.76	3.243E+09
C	0.162, 4.249	1.0	1.188E-10	308.44	1.290E+08
D	0.305, 2.291	0.5	5.939E-11	218.10	1.657E+06
E	0.619, 1.150	0.001706	2.026E-13	18.017	1.000
F	0.621, 1.146	0	0	14.591	0
G	0.626, 1.137	-0.002500	-2.970E-13	9.8976	-0.703
H	0.845	-0.066873	-7.943E-12	N/A	N/A

ponents of the disjoining pressure are equal in magnitude. In Case F, the electronic disjoining pressure is zero and the system retains only the dispersion force component of the disjoining pressure. Case G represents the lowest electronic disjoining pressure boundary condition possible for a stable thin film solution and was obtained by a trial and error approach. This results in a situation where the positive London dispersion disjoining pressure is just able to overcome the negative (repulsive) electronic disjoining pressure. In cases E-G, electrons penetrate into the substrate, effectively lowering the electron "pressure."

Finally, Case H represents the lower limit to the electronic disjoining pressure boundary condition. It results in a negative electronic disjoining pressure that overcomes the positive London dispersion disjoining pressure. As such, no steady thin film solution is possible.

Figure 6.6 shows the corresponding results of a variation in the electronic disjoining pressure boundary condition parameter, $\chi(\kappa_n)$, on the (a) liquid metal thin film profile, (b) evaporative mass flux, and (c) liquid pressure gradient. It is interesting to note that the solutions tend to follow two very distinct thin film profiles. Either the electronic component of the disjoining pressure causes a drastic change in the extended evaporating meniscus or it doesn't. As a particular consequence, the adsorbed film thickness tends towards two distinct ranges of values that are an order of magnitude apart.

For the majority of work function boundary condition values, the electronic component of the disjoining pressure causes a drastic change in the extended evaporating meniscus (Cases A-D in Fig. 6.1.3) as compared to the case of no electronic component (Case F). Second, work function boundary conditions near zero (Case E and G) do not result in drastic deviations from that of Case F in the meniscus profiles. More importantly, the adsorbed film thickness substantially increases, more than one order of magnitude from ten nanometers to hundreds of nanometers, with increasing work function boundary conditions for the modeled range, in other words, with increasing electron degeneracy contribution.

Figure 6.1.3 shows the liquid pressure gradient that is needed to replenish fluid in the thin film from the bulk meniscus to maintain a steady evaporating thin film profile. Considering Case F (no electronic component) as the baseline, we see that an increasing work function boundary condition, from Case E to A, results in an increasing total disjoining pressure which thickens and lengthens the thin film profile. This, in turn, broadens the evaporative mass flux curve and the liquid pressure gradient is reduced. For Case G, a negative electronic disjoining pressure component results in a thinner and shorter thin film profile. This is seen to sharpen the evaporative mass flux curve. The liquid pressure gradient needed to support this is much higher. Due to the steep thin film profile and elevated liquid pressure gradient, it is unknown if this evaporating thin film scenario could be stable.

The net evaporative mass flux plot in Figure 6.1.3 shows that all solution curves reach a peak within the thin film regime and then begin to decrease as the bulk meniscus region is approached. This evaporation reduction past the peak corresponds to the increasing heat transfer resistance of the thickening film. We see that substantial evaporation continues to occur at the end of the thin film for liquid sodium. In contrast, the net evaporative mass flux is practically zero at the beginning of the bulk meniscus regime for the published results of more traditional coolants. The difference is attributed to the high thermal conductivity of liquid metal which, in turn, reduces heat transfer resistance for comparable film thicknesses. Clearly, the net evaporative mass flux in the bulk meniscus region must also be modeled to obtain a true picture of the total capillary evaporation potential.

Another feature of interest from the net evaporative mass flux plot is the evaporation near the adsorbed film regime. Technically, $x = 0$ should correspond to the adsorbed film with no evaporation possible. The fact that evaporation does occur at $x = 0$ corresponds to the choice of boundary conditions in the governing thin film equation, per Section 2.4 (The reader will remember that the independent variable and its first derivative were perturbed slightly to avoid a trivial solution). This still does not explain the variance in initial evaporation fluxes for the difference disjoining pressure cases. The answer here lies in the fact that the net evaporative mass flux, as seen in Equation (2.17), depends in large part upon the second derivative of the film thickness (i.e. the curvature). By definition, the boundary condition for the curvature was fixed at the far-field condition. The curvature value at $x = 0$ is then not fixed and left to vary with the solution. Hence, the "initial" net evaporative mass flux at $x = 0$ is seen to vary for the different thin film solutions.

It is possible to integrate the net evaporative mass flux solutions in Figure 6.1.3 to obtain the total rate of evaporation for the extended meniscus thin film. The net evaporative mass flux must be integrated along the surface area (not the substrate area) from the adsorbed film ($x = 0$) to a point where the bulk meniscus is reached ($x = x_{tf}$), as specified in the far-field boundary condition of Equation (2.30d). Since the net evaporative mass flux is already normal to the thin film surface, Equation (2.17) may be applied in a line integration

$$\dot{m}_{evp} = \iint\limits_{S} \vec{\dot{m}}''_{evp} \cdot \hat{n}\, dS \approx 2\pi R \int\limits_{x=0}^{x_{tf}} \dot{m}''_{evp}(x) \sqrt{1 + \left(\frac{dy}{dx}\right)^2}\, dx \ , \tag{6.2}$$

and this line integral is then extruded around the capillary via the $2\pi R$ term. The approximate symbol, $\approx$, serves as a reminder of the simplifying curvature assumption in Equation (2.7) under which the governing equation was constructed.

The results of the total rate of evaporation calculation for the seven disjoining pressure cases are listed in Table 6.3. The results show that, on the whole, as the electronic component of the disjoining pressure decreases, the total rate of evaporation in the thin film also decreases. The one exception to this trend is seen between Cases E and F. In Case E, the electronic and dispersion components of the disjoining pressure are of equal magnitude. In Case F, the electronic component is negligible, and yet the total rate of evaporation increases approximately 15%. As before, the culprit of this discrepancy is the second derivative of the film thickness (i.e. the curvature). Even though the two cases present very similar thin film profiles, the curvatures at both the adsorbed film and bulk meniscus borders create differences in the net evaporative mass flux curve and film cut-off point, respectively.

6.1.4 Justification of Assumptions

Pure Substances

The assumption of pure substances and atomically smooth surfaces is a major assumption in our work. The existence of impurities can significantly alter model assumptions and, therefore, results. Wayner [55]

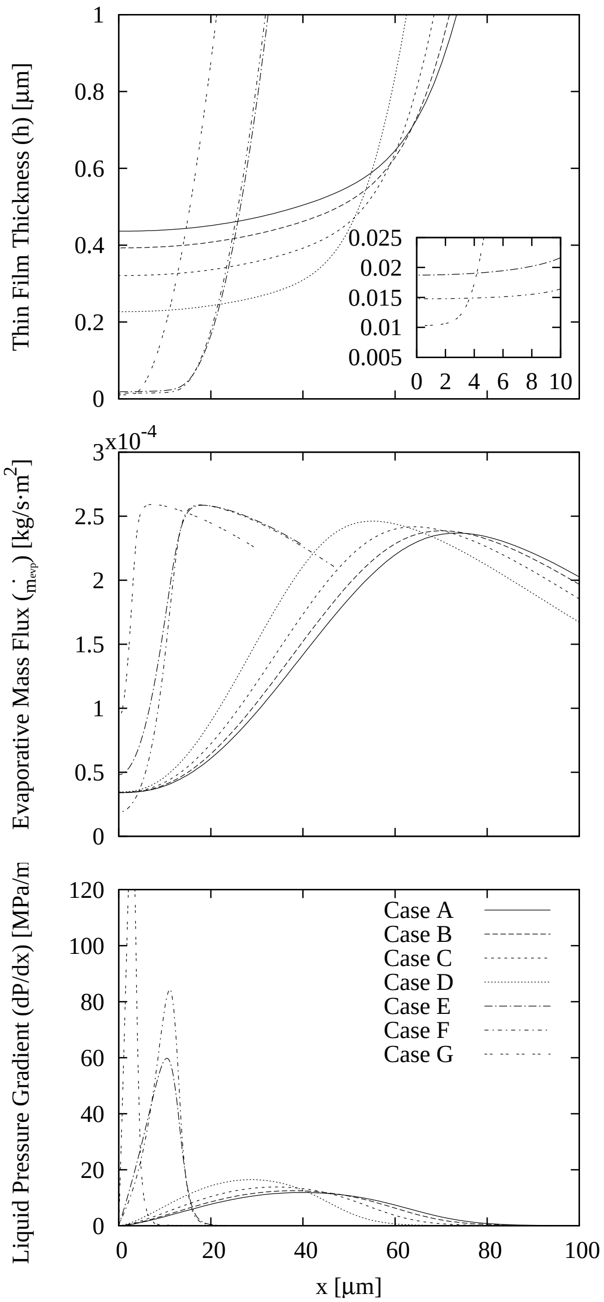


Figure 6.6: Steady thin-film evaporation solutions as functions of x measured from the absorption thickness H_0 over the range of possible disjoining pressures: (a) thin film thickness, (b) evaporative mass flux, and (c) liquid pressure gradient. Cases A to G, listed in Table 6.1 represent the effects of varying strengths of the electron degeneracy component of the disjoining pressure depending upon the electronic work function boundary condition [$R = 200\mu m$ and $\Delta T = 0.0005K$].

discusses an important point regarding the validity of this assumption.

> To connect these concepts and the Hamaker constant with experimental observations, we assume (at times) that there is no practical difference between these processes of interfacial formation occurring in a vacuum and an environment saturated with vapor or gas. However, it is also important to realize that the interfacial free energy values can, in some cases, be substantially different in laboratory air because of the adsorption of foreign vapor molecules like water and hydrocarbons. At liquid-vapor interfaces, impurities may or may not concentrate at the surface and thereby affect the value of the interfacial free energy. A further complication can arise if the environment has a foreign gas which can adsorb on the liquid substrate and change [the surface tension].

As such, significant departures from simplified mathematical models and actual engineering systems are to be expected.

The concept of a perfectly wetting system is closely related to the assumption of pure substances. Perfect wetting and spreading requires the adsorption of a layer of liquid metal atoms onto the solid metal surface. Foreign matter in the liquid metal can chemically attack the solid surface at high temperatures and leave behind surface impurities. Several papers have explored the corrosion of stainless steel by impurities in liquid sodium. Ratz and Brickner observed the adsorption of nitrogen on SS304 pipe that had been exposed to liquid sodium [94]. Barker and Wood reported the presence of corrosion products $NaCrO_2$ and Na_4FeO_3 on stainless steel after exposure to liquid sodium [95]. Moberly *et al.* exposed SS304 to reactor grade liquid sodium and found evidence of grain attack with precipitates, possibly carbon [96]. Table 6.2 lists the impurities they reported in the reactor grade sodium. Clearly, care must be taken to remove as many impurities from liquid sodium samples as possible.

In addition, oxidation of the metal surface and the adsorption of a foreign gas onto the solid surface are two prominent vehicles for contamination that will drastically alter liquid metal wetting and spreading. Barlow and Planting measured contact angles of liquid mercury droplets on several metals including iron and nickel at room temperature [97]. They discovered that mercury would not wet these metals even after they had been carefully electropolished and degreased. Spreading only occurred during or immediately after argon ion-bombardment of the metal surface. They surmised the ion-bombardment was necessary to

Table 6.2: Expected impurities and amounts in nuclear reactor grade sodium as reported by Moberly et al. [96].

Element	ppm	Element	ppm	Element	ppm
Fe	3	Cr	1	Sr	1
B	5	Si	15	Ba	3
Co	5	Ti	5	Ca	5
Mn	1	Ni	1	Li	1
Al	2	Mo	5	K	40
Mg	1	V	1	Rb	10
Sn	5	Be	1	O_2	10
Cu	2	Ag	1	C	22
Pb	5	Zr	10		
Y	5	Bi	5		

remove oxidation and gas adsorption on the metal surfaces and enable liquid mercury adsorption. Longson and Prescott explored the wetting of stainless steel by liquid sodium and found it difficult to obtain a zero contact angle around $200 - 250^0C$ even after electropolishing [98]. Since wettability is known to increase with temperature, it seems reasonable to expect liquid sodium will readily wet a properly electropolished, degreased, and ion-bombarded SS304 surface at temperatures near its melting point.

Ideal Gas

Both the Hertz-Knudsen-Schrage and extended Clapeyron relations, Equations (2.2) and (2.11) respectively, intrinsically model the alkali vapor as an ideal gas. Hensel and Warren warn that the vapors of alkali metals cannot be accurately modeled as monatomic gases even at relatively low densities [99, p.98]. According to these authors, researchers have had some success fitting alkali vapor data to equations of state using virial expansions that include coefficients for one, two, and three molecule interactions. The departure of liquid sodium vapor from an ideal gas is gauged by the compressibility factor, Z,

$$\frac{P_v \mathcal{M}}{\rho_v \mathcal{R} T_v} = Z \tag{6.3}$$

Using the data reported by Fink and Leibowitz [81] it is determined that $Z = 0.87$ at the boiling point of sodium. Thus, the ideal gas assumption used in this work is expected to result in a 13% error from the true vapor properties.

Additivity of Disjoining Pressure Components

Finally, we observe that the simple additivity of the dispersion force and electronic components of the disjoining pressure remains in the realm of educated conjecture until verified by experimentation.

6.1.5 Identification of Future Work

At this point, it remains an open question as to which of the work function boundary condition cases best represents an experimental reality. The work function for a liquid sodium thin film on a stainless steel substrate is unknown. In experiments with mercury films surrounded by organic fluids, Derjaguin and Roldughin found that organic fluids with approximately the same physical parameters caused extremely different results in mercury film stability [57]. As such, theoretical knowledge of the work function seems improbable and must be obtained experimentally. Even if such a measurement could be made, another rather large obstacle exists as the liquid sodium system must be isolated in a high temperature, low oxygen environment that is rather inaccessible for delicate and operator intensive operations.

Visualization of a capillary evaporation experiment will also be difficult. Sodium low- and high- pressure lamps are a mature illumination technology that provide useful insight into the detrimental effects of sodium vapor. It is known that sodium vapor reacts chemically with silica (including Pyrex glass and Quartz) to form sodium silicate [100]. As a result, the glass container is chemically etched and develops orange/brown deposits which preclude visualization [101]. Commercial lighting applications have developed proprietary coatings of polycrystalline alumina which resist chemical attack but at the expense of transparency [102, pp.190,199]. Sapphire tubing appears to be the only material available that can provide transparency and resistance to sodium vapor attack at high temperatures [103, p.234]. The anisotropic nature of this crystalline material, however, makes it prone to cracks and sealing problems. Truly, experimental visualization of liquid sodium capillary evaporation will prove difficult.

The present study, however, leads us to propose a somewhat more tractable test. The largest unknown in this study is the proper magnitude of the electronic component of the disjoining pressure. As Figure 6.6 illustrates, if the adsorbed film thickness of an evaporating extended meniscus could be measured with just enough accuracy to distinguish between a value that is on the order of tens of nanometers or hundreds of nanometers, this could serve to determine the approximate range of work function and resulting magnitude of the electronic component of the disjoining pressure. Such a work could be regarded as a significant breakthrough in the study of alkaline liquid metal evaporating thin films.

In light of these previous statements, it would now be worthwhile to shift experimental goals from visualization of liquid sodium evaporation to visualization of an adsorbed film, which could be performed on a solidified, room temperature sample. Ahn and Berghezan researched the infiltration of liquid metal into metal capillaries by cooling the samples and using a Scanning Electron Microscope (SEM) to image the propagation front [104]. They reported the ubiquitous presence of a “precursor film” in the rise of a perfectly wetting molten metal in a solid metal capillary for a variety of metal combinations and attributed the film to chemical adsorption and capillary condensation. Similarly, a prepared SS304 sample could be wetted with liquid sodium, cooled, and transferred to either a SEM or an Atomic Force Microscope (AFM) to measure the topology of the adsorbed precursor film. Here, the major technological challenge would be the transfer of the sample to an SEM or AFM without oxidation of the sample. Either a vacuum would have to be maintained during the transfer, or an inert gas would have to be used.

6.2 Comprehensive and Multiscale Modeling of a Liquid Metal Evaporating Capillary

6.2.1 Effect of Electronic Disjoining Pressure Component

Table 6.3 lists the parameters used to splice the extended meniscus thin film model to the CFD model of the bulk meniscus. Seven CFD models are needed for the seven representative disjoining pressure conditions listed in Table 6.1. The "Cutoff Height" parameter represents the vertical distance from the bottom of the capillary meniscus (located at $r = 0$) to the top of the thin film boundary. The "Thin Film Evaporation" parameter is utilized to specify the temperature boundary condition along the thin film interface using Equation (5.3) and is calculated using Equation (6.2). The "1D Velocity" parameter specifies the normal velocity boundary condition into the thin film. It is calculated as

$$v = \frac{\dot{m}_{evp}}{\rho\,\mathrm{SA}} = \frac{\dot{m}_{evp}}{\rho\pi\left[R^2 - (R - H_{tf})^2\right]} \tag{6.4}$$

where both $\dot{m}_{evp}$ (the thin film evaporation) and H_{tf} (the film thickness at the bulk meniscus interface) are listed in the table. This velocity boundary condition is uniform due to the approximation of lubrication theory that was utilized to construct the extended meniscus thin film model.

The seven CFD models of the bulk meniscus were created and solved using the COMSOL® Finite Element software package. Figures 6.7-6.14 show these models and their graphical results. In each figure, the leftmost image details the triangular mesh used for the finite elements. A mesh refinement study was performed to ensure that the solutions were no longer mesh dependent. The central image depicts an velocity vector field plot overlaying a surface contour plot of the temperature overheat field. The velocity vector field shows the flow field correctly changing from fully developed Stoke's flow to meet the specified evaporation boundary conditions. The 1D velocity boundary condition at the extended meniscus thin film interface clearly represents the largest velocity in the model, much larger than the outlet velocities at the evaporating bulk meniscus. The temperature overheat plot shows clear striation, indicating conduction dominant heat transfer, and the presence of evaporation at the meniscus appropriately reduces the overheat towards the capillary centerline. Finally, the rightmost images give a surface contour plot of the velocity field. These again show the large fluid flow that is needed to replenish the evaporating extended meniscus thin film.

Table 6.3: Parameters used to splice the extended meniscus thin film model to the CFD model of the bulk meniscus. The cases represent possible variations in the boundary condition term of Derjaguin's electronic component of the disjoining pressure as listed in Table 6.1. The cutoff height represents the vertical distance from the bottom of the capillary meniscus of Figure 6.7 at which the bulk meniscus is cut off to join with the extended meniscus thin film model.

CASE	**Film Thickness** (μm)	**Cutoff Height** (μm)	**Thin Film Evaporation** (kg/s)	**1D Velocity** ($\mu m/s$)	**Surface Overheat** (K)
A	17.72	83.07	2.949e-11	1.867	3.472e-4
B	13.20	87.23	2.728e-11	2.290	3.661e-4
C	10.25	90.01	2.505e-11	2.688	3.825e-4
D	6.892	93.23	2.163e-11	3.422	4.073e-4
E	1.811	98.20	1.013e-11	6.022	4.670e-4
F	3.019	97.00	1.165e-11	4.169	4.492e-4
G	1.962	98.05	0.862e-11	4.733	4.646e-4

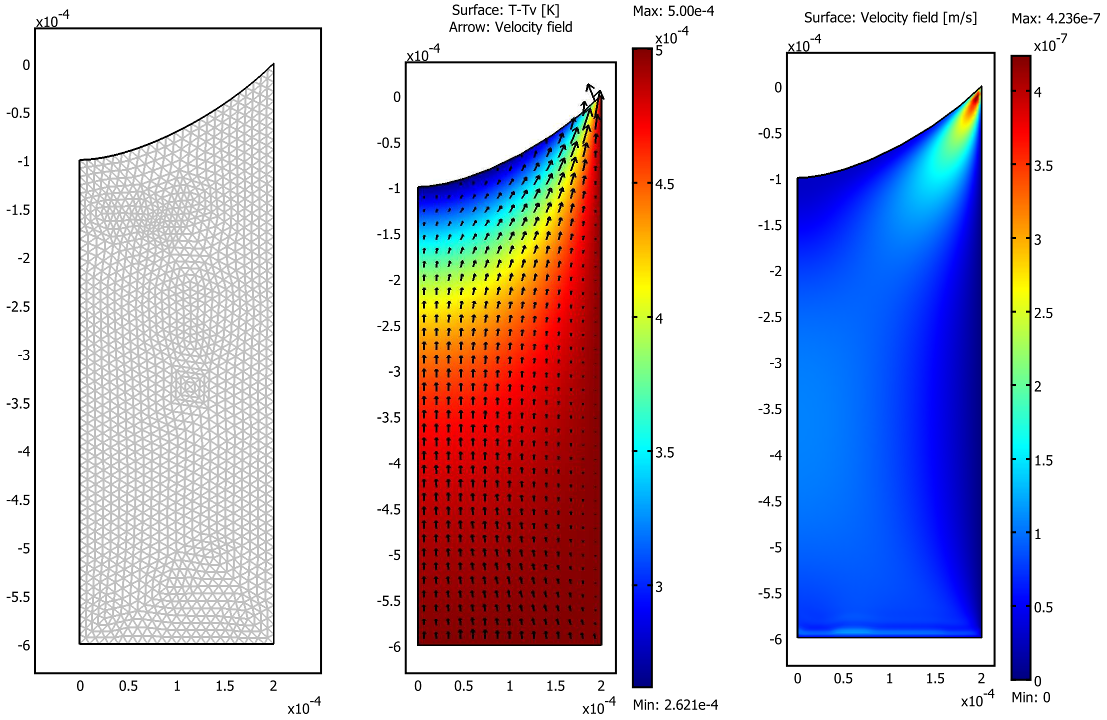


Figure 6.7: A COMSOL® CFD model of an evaporating capillary meniscus with no extended thin film. The plots represent (a) element meshing, (b) relative velocity field vectors overlaid against temperature overheat contours, and (c) velocity field contours [$R = 200\mu m$ and $\Delta T = 0.0005K$].

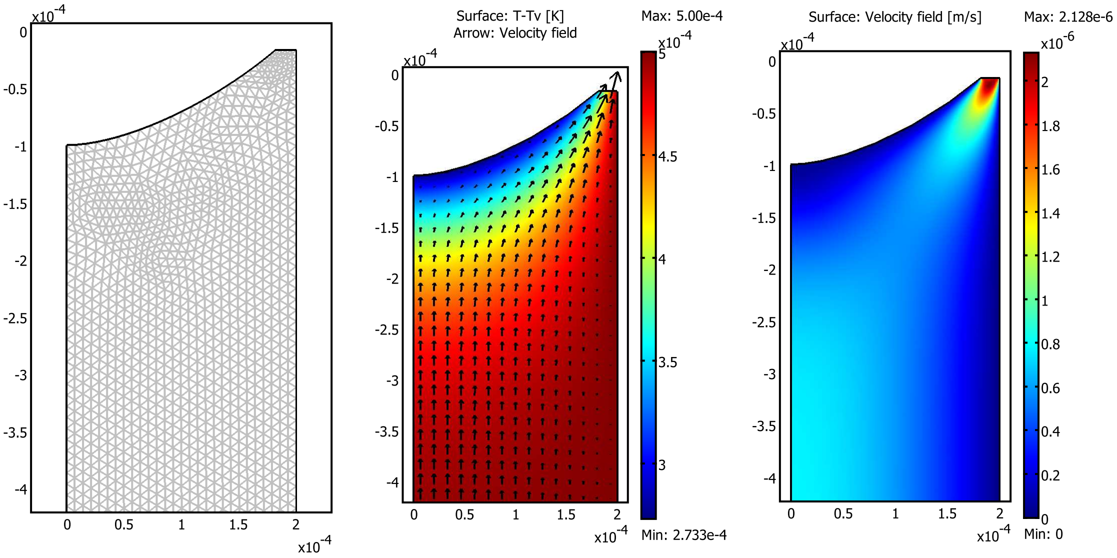


Figure 6.8: A COMSOL® CFD model of an evaporating capillary meniscus. The blocked-off meniscus is spliced to extended meniscus thin film Case A as listed in Tables 6.1 and 6.3. The plots represent (a) element meshing, (b) relative velocity field vectors overlaid against temperature overheat contours, and (c) velocity field contours [$R = 200\mu m$ and $\Delta T = 0.0005K$].

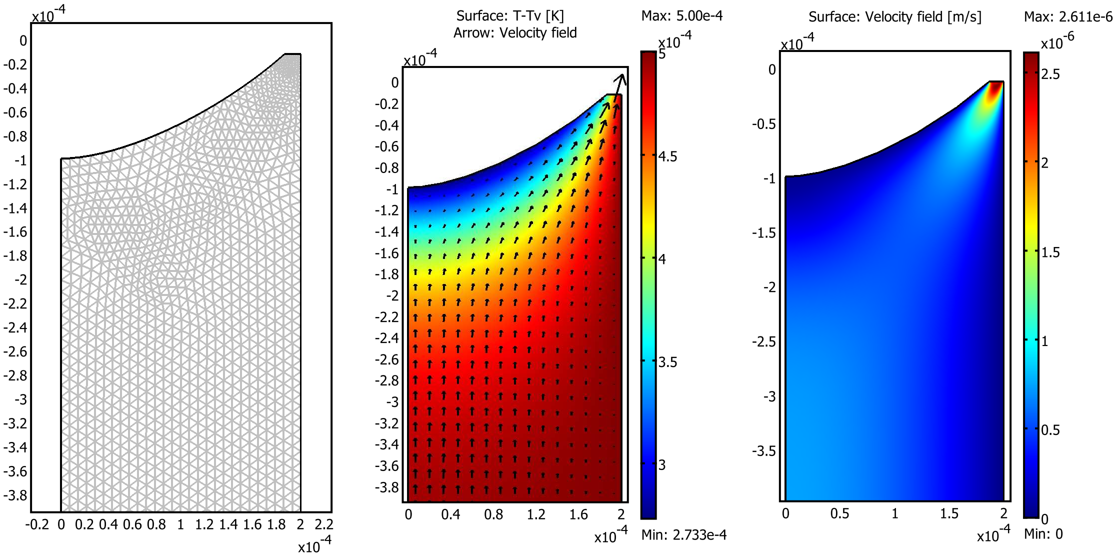


Figure 6.9: A COMSOL® CFD model of an evaporating capillary meniscus. The blocked-off meniscus is spliced to extended meniscus thin film Case B as listed in Tables 6.1 and 6.3. The plots represent (a) element meshing, (b) relative velocity field vectors overlaid against temperature overheat contours, and (c) velocity field contours [$R = 200\mu m$ and $\Delta T = 0.0005K$].

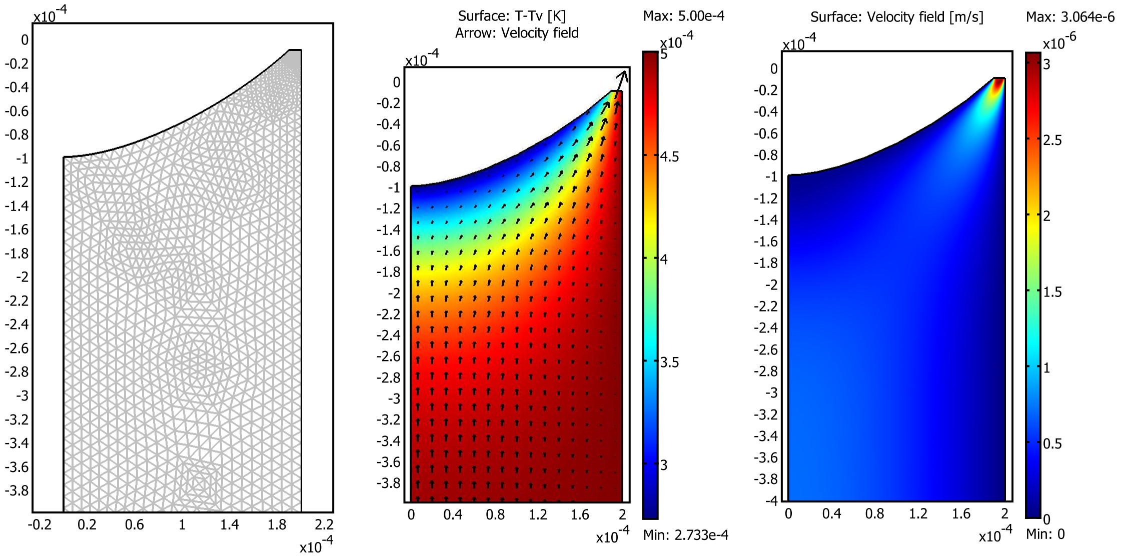


Figure 6.10: A COMSOL® CFD model of an evaporating capillary meniscus. The blocked-off meniscus is spliced to extended meniscus thin film Case C as listed in Tables 6.1 and 6.3. The plots represent (a) element meshing, (b) relative velocity field vectors overlaid against temperature overheat contours, and (c) velocity field contours [$R = 200\mu m$ and $\Delta T = 0.0005K$].

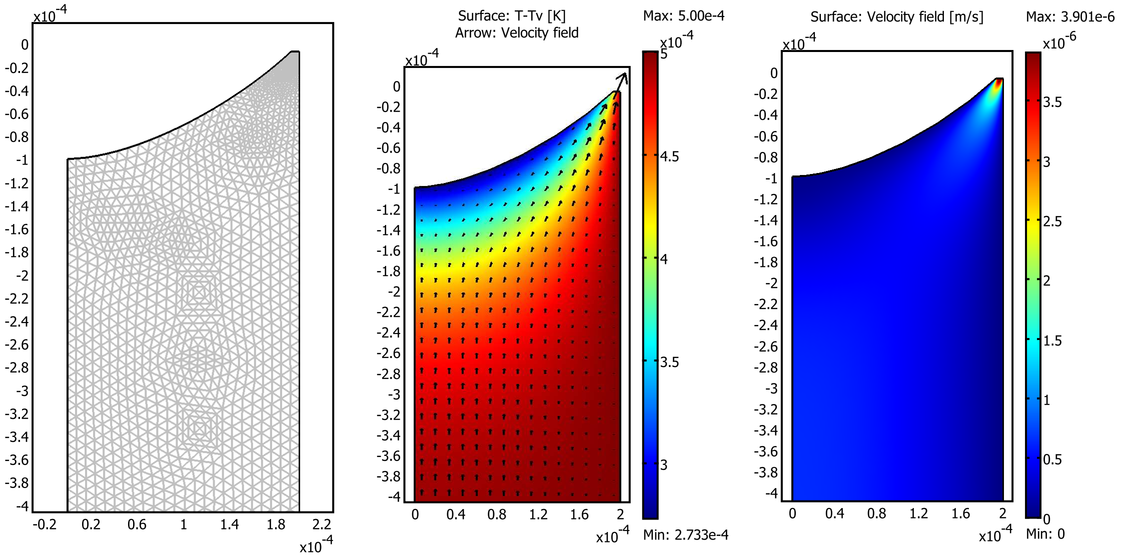


Figure 6.11: A COMSOL® CFD model of an evaporating capillary meniscus. The blocked-off meniscus is spliced to extended meniscus thin film Case D as listed in Tables 6.1 and 6.3. The plots represent (a) element meshing, (b) relative velocity field vectors overlaid against temperature overheat contours, and (c) velocity field contours [$R = 200\mu m$ and $\Delta T = 0.0005K$].

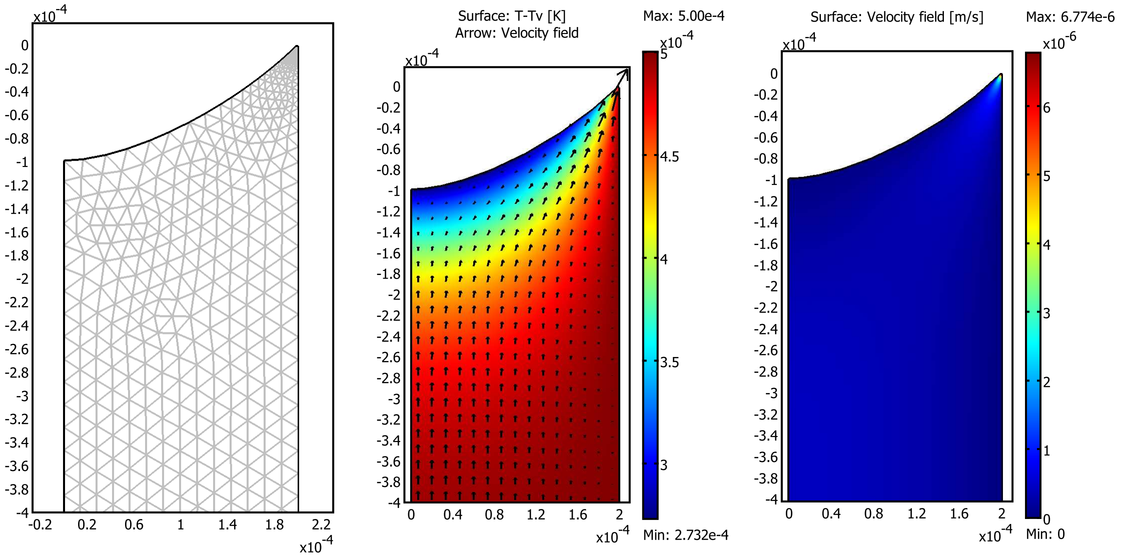


Figure 6.12: A COMSOL® CFD model of an evaporating capillary meniscus. The blocked-off meniscus is spliced to extended meniscus thin film Case E as listed in Tables 6.1 and 6.3. The plots represent (a) element meshing, (b) relative velocity field vectors overlaid against temperature overheat contours, and (c) velocity field contours [$R = 200\mu m$ and $\Delta T = 0.0005K$].

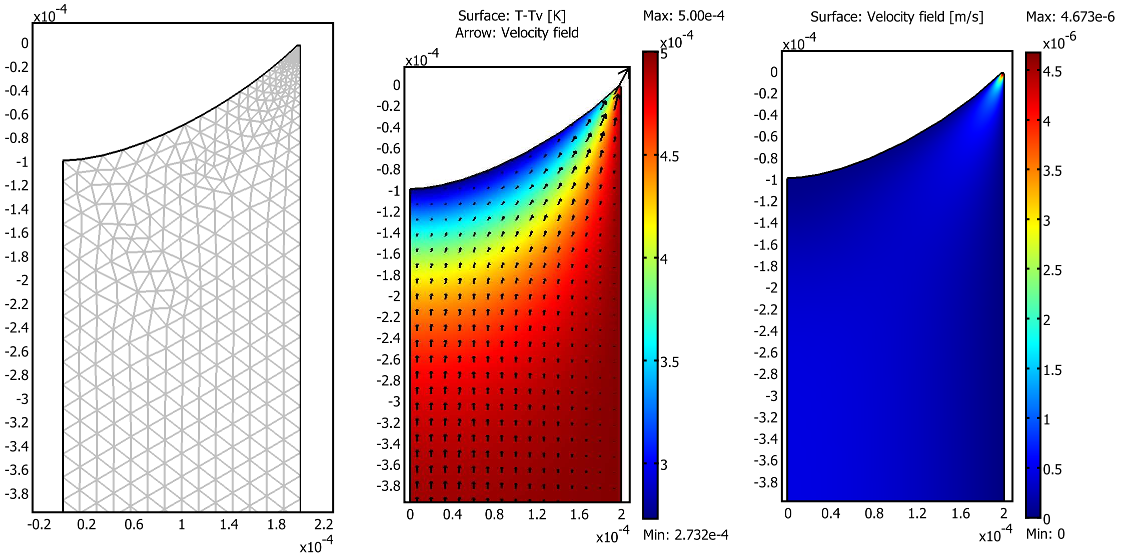


Figure 6.13: A COMSOL® CFD model of an evaporating capillary meniscus. The blocked-off meniscus is spliced to extended meniscus thin film Case F as listed in Tables 6.1 and 6.3. The plots represent (a) element meshing, (b) relative velocity field vectors overlaid against temperature overheat contours, and (c) velocity field contours [$R = 200\mu m$ and $\Delta T = 0.0005K$].

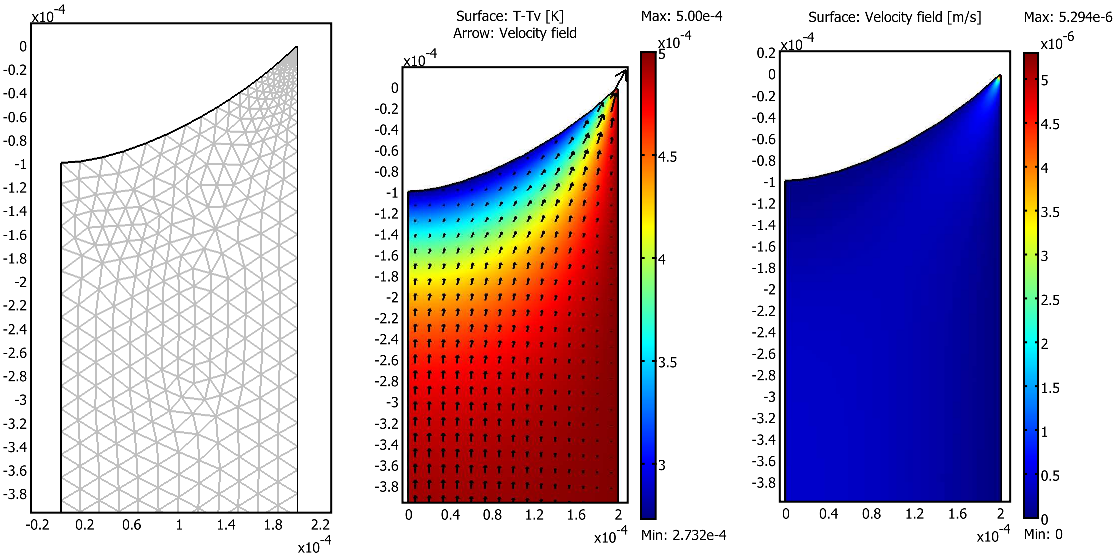


Figure 6.14: A COMSOL® CFD model of an evaporating capillary meniscus. The blocked-off meniscus is spliced to extended meniscus thin film Case G as listed in Tables 6.1 and 6.3. The plots represent (a) element meshing, (b) relative velocity field vectors overlaid against temperature overheat contours, and (c) velocity field contours [$R = 200\mu m$ and $\Delta T = 0.0005K$].

Figure 6.15 splices together the extended meniscus thin film evaporation solutions of Figure 6.6 with the CFD bulk meniscus solutions of Figures 6.8-6.14. The result is a truly comprehensive multiscale model of a liquid metal evaporating capillary. The top two plots in Figure 6.15 detail the total capillary meniscus profile. A slope discontinuity is clearly evident in the transition of each curve from the bulk meniscus model to the extended meniscus thin film model. This is a direct result of the curvature approximation, seen in Equation (2.7), upon which the extended meniscus thin film model is built. As a result, the total capillary meniscus profile and its second derivative (i.e. the curvature) are continuous while slope continuity is not enforced. This result does not deter us, however, from making the important observation that capillary meniscus surface area increases along with increasing electronic component of the disjoining pressure.

The bottom plot in Figure 6.15 gives the net evaporative mass flux across the entire capillary meniscus. A slight discontinuity in the slope is observed where the extended meniscus thin film model abruptly changes to the CFD model, as is to be expected. In addition, a jump discontinuity is present in the curves for cases A and B. It is surmised that this jump discontinuity again results from the simplified curvature assumption present in the extended meniscus thin film model. As can be seen in Table 6.3, the thin film heights at the far-field for cases A and B are $17.7\mu m$ and $13.2\mu m$, respectively. These yield H_{tf}/R values of 9% and 7%, respectively. Essentially, results from a Cartesian geometry are being combined with the results from a cylindrical geometry, and the error in this approximation is becoming large enough in the thicker films of cases A and B to cause a jump discontinuity.

Integration of the evaporative mass flux across the total capillary meniscus surface area yields total evaporative mass flow rates. To this end, the extended meniscus thin film integrations performed using Equation (6.2) are added to the following bulk meniscus surface area integration

$$\dot{m}_{evp} = \iint_S \vec{\dot{m}}''_{evp} \cdot \hat{n}\, dS = \int_{\theta=0}^{2\pi} d\theta \int_{r=0}^{R-H_{tf}} \dot{m}''_{evp}(r) r \sqrt{1 + \left(\frac{dz}{dr}\right)^2}\, dr \ , \tag{6.5}$$

where axial symmetry is assumed using a cylindrical coordinate system. The slope in this calculation comes from Equation (5.2). Figure 6.16 presents columnstacked bar charts that compare the meniscus surface area, net evaporative mass flow rate, and total capillary heat flux for the varying cases of thin film disjoining pressures. It is seen that the bulk meniscus constitutes from 40% to 80% of the total meniscus surface area for Cases A to G respectively. This corresponds with a 3% to 16% bulk meniscus contribution to the evaporative mass flow rate (and, hence, heat flow rate) for Cases A to G respectively. Finally, the bulk meniscus region contributes a roughly constant 5% of the total capillary heat flux in each of the disjoining pressure cases. Clearly, appreciable heat and mass transfer takes place in the bulk meniscus region of an evaporating liquid metal capillary.

The overall trend from these plots is that a larger electronic component of the disjoining pressure leads towards larger extended meniscus thin film surface area, larger total capillary meniscus surface area, and larger net evaporative mass flow rate (which corresponds with larger heat flow rate). Cases A-D are obviously desirable, while the stability of Cases E-G are questionable due to the higher sustained heat fluxes that are necessary to support evaporation in the extended meniscus thin film.

The most optimum situation appears to be disjoining pressure case Case A, in which the electron degeneracy disjoining pressure is at a maximum. This occurs when the boundary condition parameter $\chi(\kappa_n)$ is at the theoretical maximum limit of $3\pi/16$. This indicates that the work function, W, or energy needed to move an electron from the liquid metal to the solid surface, is at an infinite limit. As first derived in Derjaguin *et al.* [48], this infinite limit represents the simplified assumption of an infinitely deep potential pit at the liquid/substrate boundary that prevents electrons from emerging from the film. In light of the presents results, future research should attempt to identify substrate metallurgies and treatments to induce the desired work function.

6.2.2 Justification of Assumptions

Negligible Buoyancy Forces

Equations (5.1) model the fluid flow and heat transfer in the capillary. It is noted that the Boussinesq approximation, typically used to model buoyancy forces, is absent. The nondimensional Bond number

$$\mathrm{Bo} = \frac{g(\rho_l - \rho_v)D^2}{\gamma} = \frac{g(\rho_l - \frac{P_v}{RT_v})D^2}{\gamma} \tag{6.6}$$

gives the ratio of gravitational to surface tension forces. For the range of capillary pore sizes studied in this research, $\mathrm{Bo} = 0.0097$ for $R = 200\mu m$ and $\mathrm{Bo} = 0.0608$ for $R = 500\mu m$. Clearly, the buoyancy forces are negligible for the micro-pore geometries considered.

Bulk Meniscus Profile

The capillary meniscus profile is found from the curvature equation. As discussed in the extended meniscus thin film models via Equation (2.7), the curvature equation can be significantly simplified when the slope is considered small compared to the second derivative. From Equation (5.2), we observe that the second derivative of the meniscus is on the order of 5,000 (since $z'' \approx 1/R$) while the first derivative is on the order of unity (since $z' \approx r/R$, $0 \leq z'(r) \leq 1$). Clearly, this assumption holds for the pore geometries considered.

6.2.3 Identification of Future Work

The curvature equation simplification constrains the solution parameters of this study the most. It would be beneficial in future work to attempt to incorporate the full meniscus curvature equation into the multiscale capillary model. Wee *et al.* [22] achieved this for an extended meniscus thin film model using traditional coolants. Extension of this to the current system, while not trivial, seems feasible. The main benefit is removal of the current limits on available liquid overheat and capillary pore size. Equation (6.1), discussed previously, led to the current solution limits of $0.0003K \leq \Delta T \leq 0.0007K$ and $200\mu m \leq R \leq 500\mu m$. Smaller capillary pore sizes and larger overheats would be possible with the full curvature equation.

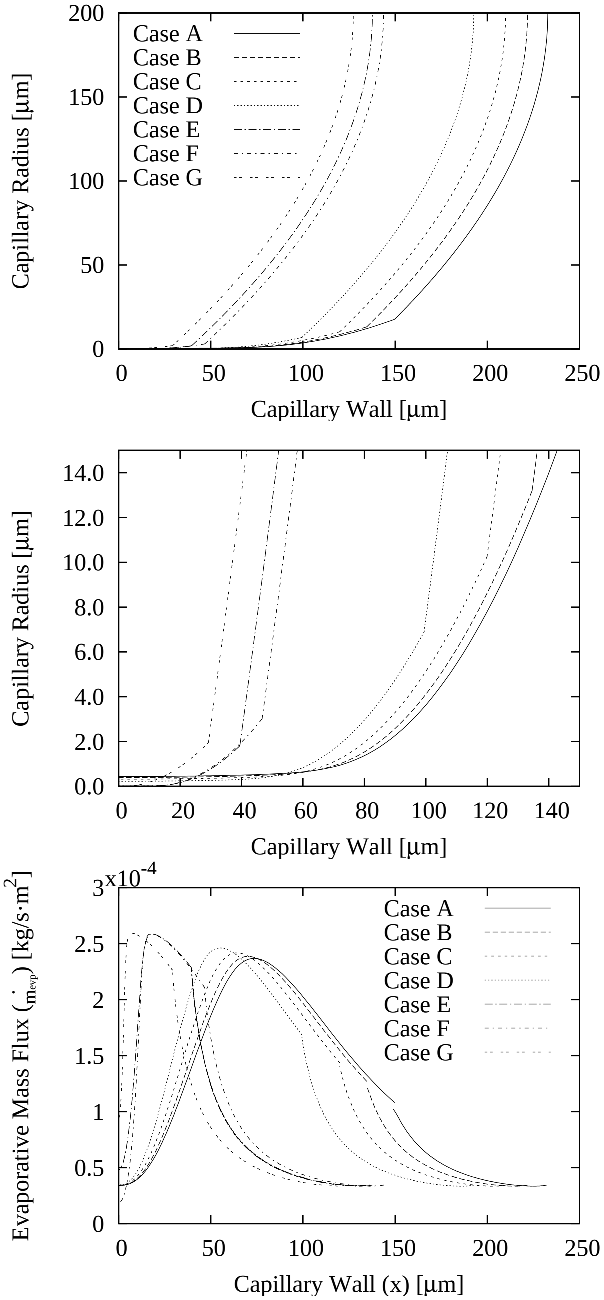


Figure 6.15: Steady total capillary meniscus evaporation solutions measured from the absorption thickness H_0 to the capillary centerline: (a)-(b) thin film thickness and (c) evaporative mass flux. Cases A to G represent the range of possible disjoining pressures as referenced in Table 6.1 [$R = 200\mu m$ and $\Delta T = 0.0005K$].

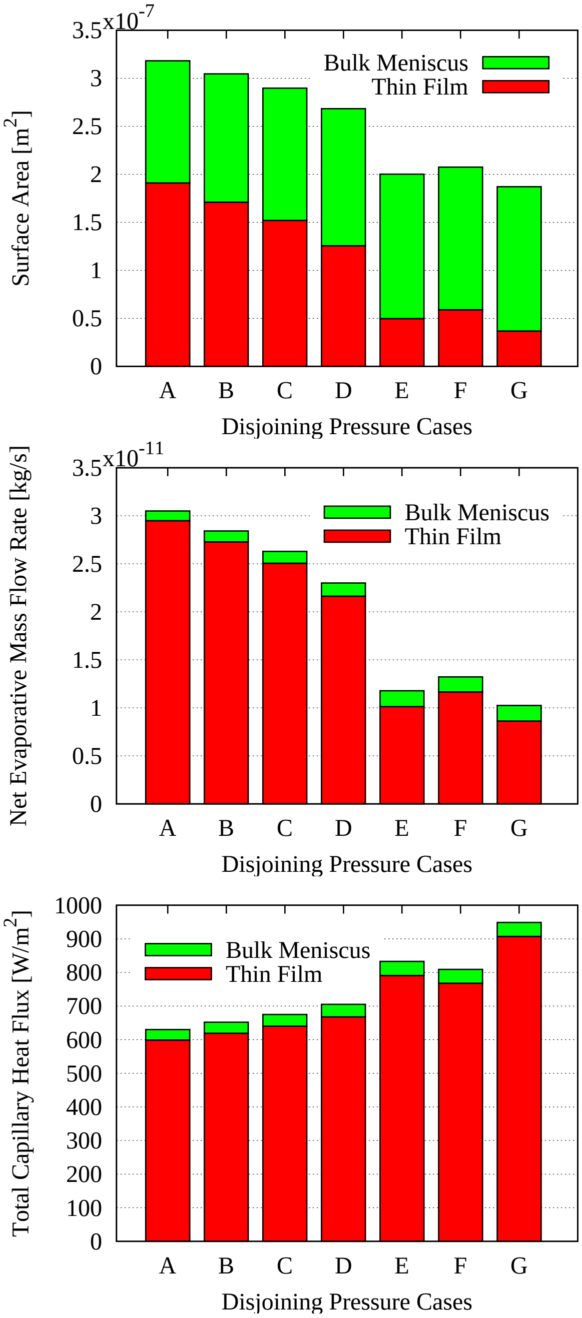


Figure 6.16: Columnstacked bar charts which compare the extended meniscus thin film and bulk meniscus contributions to capillary meniscus (a) surface area, (b) net evaporative mass flow rate, and (c) total heat flux. Cases A to G represent the range of possible disjoining pressures as referenced in Table 6.1 [$R = 200\mu m$ and $\Delta T = 0.0005K$].

Chapter 7

Liquid Metal Capillary Evaporation Stability: Future Research

7.1 Liquid Metal Extended Meniscus Stability

Capillary Pumped Loops (CPL) and Loop Heat Pipes (LHP) are “real world,” passive, heat transfer devices that utilize the unique physics of capillary evaporation. Experimental investigations of CPLs and LHPs with traditional working fluids have demonstrated performance degradations due to temporal fluctuations [26]. One of the major assumptions used in the research presented in this dissertation (as well as in the majority of research to date) only considers the case of a static meniscus where the fluid lost by evaporation is continually replenished by flow from the bulk capillary. In light of this steady-state assumption, it would seem that future research in liquid metal extended meniscus evaporation should consider temporal stability.

He and Hallinan [35] were the first to explore the role of thermocapillary effects on evaporating extended meniscus stability. They included thermocapillary forces in a thin film evaporation model. The solution results were used to anecdotally explain the change in the Hamaker constant for the experimental results of Wayner. They also used scaling analysis to identify a critical Marangoni number that corresponded with the thermocapillary limit of an evaporating thin film. Later, Pratt and Hallinan [105] experimentally tested capillary stability using pentane as the working fluid. They observed degraded meniscus wettability due to thermocapillary stresses for large wall-temperature gradients in the vicinity of the vapor/thin film/wall contact line. Finally, Pratt *et al.* [26] attempted to apply the nonlinear stability theory of planar evaporating films by Burelbach *et al.* [106] to the stability of a heated, curved meniscus. Their analysis identified a critical wall temperature difference per length scale that was compared with experimental results using pentane.

The first step in exploring thermocapillary effects on liquid metal thin film stability is to include the surface tension dependence on temperature. Several authors have attempted this for thin films of conventional coolants. Mirzamoghadam and Catton [29] incorporated thermocapillary effects for the case of an evaporating meniscus on an inclined plate. Swanson and Herdt [30] included thermocapillary effects in a circular pore geometry for a traditional coolant. He and Hallinan assumed that circumferential curvature was negligible and solved the thin film equation to a first approximation by linearizing the dependent variables to simplify the governing equations. Finally Wee *et al.* [22] were the first to solve the extended meniscus, thin film equation for full pore curvatures using a nonisothermal liquid/vapor interface along with surface tension temperature dependence. They focused on slip/no slip substrate boundary effects, however, using only traditional coolants for which the Hamaker approximation of the disjoining pressure applied. The main difficulty with extension of this model for a liquid metal is in the complexity of the cubic spline interpolation of the dispersion component of the disjoining pressure. Formulation of the Jacobian matrix for the Newton-Raphson method would be quite difficult, but not impossible.

Second, it would be beneficial to explore nonlinear stability theory in a liquid metal evaporating thin film. The theory of Burelbach *et al.* [106] was for planar evaporating thin films using the Hamaker approximation for the disjoining pressure. Pratt *et al.* [26] applied this theory to curved films which has merits, but also raises questions. As previously stated, Pratt *et al.* used Burelbach *et al.*'s theory to identify a critical interfacial temperature slope. The theory of Burelbach *et al.*, however, was built upon the assumption of a constant temperature substrate which might not be valid for liquid metal coolants with high thermal conductivity. Furthermore, Ma *et al.* [107] discounted the role of thermocapillary stresses on capillary performance degradation due to the small temperature gradients existing in the thin-film region.

Pratt *et al.* derived the following meniscus thermocapillary stability criterion for a curved film

$$\frac{\mathrm{Ma}}{4\,\mathrm{Pr}} \geq \Pi^{*}\,. \tag{7.1}$$

The nondimensionalizing length scale was chosen to be where the disjoining pressure balances the capillary pressure. We can model a liquid sodium film disjoining pressure using the electron degeneracy component under the boundary condition of an infinite potential energy well. The scaling film thickness then becomes

$$\frac{B}{H_{tr}^2} = \frac{2\gamma}{R} \implies H_{tr} = \pm\sqrt{\frac{B \cdot R}{2\gamma}} \tag{7.2}$$

where, based on physical reality, only the positive root is considered. Substitution of the scaling parameters into Equation (7.1) produces the critical interfacial temperature difference

$$\Delta T_c \geq \frac{4}{k}\sqrt{\frac{2B\gamma}{R}} \tag{7.3}$$

which is over a critical length scale defined by Pratt *et al.* as $x_c = \sqrt{RH_{tr}}$. For the liquid sodium case specified, using the fluid properties listed in Table 3.2, the critical interfacial temperature slope is calculated to be $21,000K/cm$. Obviously, under the geometry and assumptions considered by Pratt *et al.*, it would seem that liquid metal evaporating extended meniscus stability would not be affected by thermocapillary forces. It would thus seem prudent in a thermocapillary stability analysis to focus upon the more realistic substrate heating case under a liquid metal scenario. This would require 2D conduction through the substrate to be included in the analysis for both constant temperature and constant heat flux cases.

7.2 Liquid Metal Evaporating Capillary Boiling Stability

Third, it would be beneficial to consider the relative importance of thermocapillary stability effects in the extended meniscus thin film compared to possible boiling instabilities in the bulk evaporating capillary. Using conventional incipience of boiling models, Ruggles [108] showed that “liquid metals can easily achieve bulk superheat prior to nucleation when micro-channels are used.” Furthermore, Ruggles postulated that, in such situations, “the bubble departure diameter may be of the same order as the diameter of the first nucleation site to activate.” In other words, cross sectional changes could serve as nucleation sites for the rapid boiling incipience of a superheated liquid metal. Indeed, the appearance of rapid boiling has been a larger problem than meniscus stability in preliminary research with liquid metal capillary evaporation [109]. The best course of action to ensure flow and evaporation stability might be to focus on engineering capillary surfaces to initiate boiling at desired locations, as Ruggles has suggested.

Chapter 8

Conclusions

8.1 High Temperature, Liquid Metal, Extended Meniscus, Evaporation

The present study seeks to expand existing extended meniscus evaporation models to properly capture the unique disjoining pressure characteristics of liquid alkali metals. Where previous studies have only used the non-retarded dispersion force via Hamaker theory, this research incorporates the full (unsimplified) retarded dispersion force (Π_A) using the DLP theory and its representation by cubic spline interpolation. Additionally, this research incorporates an electronic disjoining pressure component (Π_B) that is unique to liquid metals by performing a parametric study on the work function boundary condition. The results for a liquid sodium thin film in a $200\mu m$ diameter capillary with a $0.0005K$ overheat indicate that adsorbed film thicknesses can vary from $8nm$ (Case G: $\Pi_B/\Pi_A \approx 0$) to $420nm$ (Case A: $\Pi_B/\Pi_A \approx \infty$) depending on the work function boundary condition. Thin film profiles (and thus meniscus surface areas) exhibit large changes, as well.

The important conceptual results identified from this work include the following:

1. Accurate high temperature, liquid metal, extended meniscus evaporation models should account for both retarded dispersion force and electronic disjoining pressures.

2. Cubic spline interpolation is an acceptable vehicle to model the retarded dispersion force and can be implemented within the framework of the orthogonal collocation solution method.

3. Results indicate the electronic component of the disjoining pressure is not negligible for a wide range of work function boundary values and must be included in models of liquid metal extended meniscus

evaporation.

4. Numerical solutions to the thin film governing equation for isothermal sodium coolant predicts thin film thickness profiles, mass flux distributions, and pressure gradient along the substrate of stainless steel.
5. Continuing studies require greater physical insight into the work function for a liquid sodium thin film on a stainless steel substrate.
6. A coarse experimental measurement of the adsorbed film thickness could identify a finer range of electronic disjoining pressure component boundary conditions and result in a refinement of high temperature, liquid sodium, extended meniscus evaporation models.

8.2 Comprehensive and Multiscale Modeling of a Liquid Metal Evaporating Capillary

When coupled to a CFD model of the evaporating bulk meniscus, the problem as described above also yields a multiscale numerical model of an evaporating liquid metal in a capillary tube. The model correctly considers the unique disjoining pressure effects at the near wall region, including the extended meniscus thin film profile, and captures the heat and fluid transfer through the bulk meniscus region. Multiscale integration along the total capillary surface area shows a range of heat transfer rates, from $40\mu W$ @ $950 W/m^2$ (Case G: $\Pi_B/\Pi_A \approx 0$) to $116\mu W$ @ $630 W/m^2$ (Case A: $\Pi_B/\Pi_A \approx \infty$), are possible.

The important conceptual results identified from this work include the following:

1. Integration of the evaporative mass flux across the total meniscus surface area produces total capillary evaporative mass flow and heat transfer rates and enables comparisons between electronic disjoining pressure states.
2. Unlike more traditional coolants, evaporative mass and heat flow occurs in the bulk meniscus region of evaporating micro-capillaries and should be modeled.
3. The clear trend from these comparisons is that a larger electronic component of the disjoining pressure leads towards larger extended meniscus thin film surface area, larger total capillary meniscus surface

area, and larger net evaporative mass flow rate (which corresponds with larger heat flow rate).

4. To ensure maximum heat transfer in an evaporating liquid sodium capillary, it is desirable to create a liquid sodium work function environment that prevents electrons from emerging from the evaporating thin film.

List of References

List of References

[1] T. Yamamoto, Y. Tanaks, M. Sasaki, and H. Hatori, “Study on Life of Sodium Heat Pipe,” in D. Reay, ed., *Advances in Heat Pipe Technology: Proceedings of the 4th International Heat Pipe Conference, London, UK, 7-10 September 1981* (Pergamon Press, Oxford, 1981), pp. 389–398.

[2] M. Razzaque, “On Application of Heat Pipes for Passive Shutdown Heat Removal in Advanced Liquid Metal and Gas-Cooled Reactor Designs,” Ann. Nucl. Energy **17**, 139 (1990).

[3] Y. Yamamoto, M. Mochizuki, and S. Sugihara, “Sodium heat Pipe Heat-Exchanger for High Temperature Waste Heat Recovery,” in L. L. Vasiliev, ed., *Heat Pipe Technology: Proceedings of the 7th International Heat Pipe Conference, Minsk, USSR, 21-25 May 1990* (Begell House, New York, 1993), pp. 495–505, vol. 2.

[4] C. Camarda, “Thermostructural Applications of Heat Pipes for High-Speed Aerospace Vehicles,” in *Proceedings of the Third International Heat Pipe Conference* (Tsukaba, Japan, 1988), pp. 31–43.

[5] Y. Cao, A. Faghri, and E. Mahefkey, “Micro/Miniature Heat Pipes and Operating Limitations,” in *Heat Pipes and Capillary Pumped Loops: Proceedings of the 29th National Heat Transfer Conference, Atlanta, Georgia, 8-11 Aug 1993* (ASME Heat Transfer Division, New York, 1993), pp. 55–62.

[6] G. Coldwell, “Some Techniques for Cooling Hypersonic Structures,” in *Heat Pipes and Capillary Pumped Loops: Proceedings of the 29th National Heat Transfer Conference, Atlanta, Georgia, 8-11 Aug 1993* (ASME Heat Transfer Division, New York, 1993), pp. 39–46.

[7] G. T. Coldwell, “Cooling Hypersonic Vehicle Structures,” in L. L. Vasiliev, ed., *Heat Pipe Technology: Proceedings of the 7th International Heat Pipe Conference, Minsk, USSR, 21-25 May 1990* (Begell House, New York, 1993), pp. 507–516, vol. 1.

[8] D. Glass, C. Camarda, M. Merrigan, and J. Sena, “Fabrication and Testing of a Leading-Edge-Shaped Heat Pipe,” J. Spacecraft Rockets **36**, 921 (1999).

[9] F. Prenger, Jr., “Performance Predictions and Measurements for Space Power System Heat Pipes,” in D. Reay, ed., *Advances in Heat Pipe Technology: Proceedings of the 4th International Heat Pipe Conference, London, UK, 7-10 September 1981* (Pergamon Press, Oxford, 1981), pp. 575–588.

[10] W. Ranken, “Heat Pipe Development for the SPAR Space Power System,” in D. Reay, ed., *Advances in Heat Pipe Technology: Proceedings of the 4th International Heat Pipe Conference, London, UK, 7-10 September 1981* (Pergamon Press, Oxford, 1981), pp. 561–574.

[11] O. Brost, M. Groll, and H. Mack, “High Temperature Lithium Heat Pipe Furnace for Space Applications,” in L. L. Vasiliev, ed., *Heat Pipe Technology: Proceedings of the 7th International Heat Pipe Conference, Minsk, USSR, 21-25 May 1990* (Begell House, New York, 1993), pp. 169–179, vol. 2.

[12] W. Anderson, “High Temperature Capillary Pumped Loops,” in *Heat Pipes and Capillary Pumped Loops: Proceedings of the 29th National Heat Transfer Conference, Atlanta, Georgia, 8-11 Aug 1993* (ASME Heat Transfer Division, New York, 1993), pp. 93–101.

[13] J. Ku, “Overview of Capillary Pumped Loop Technology,” in *Heat Pipes and Capillary Pumped Loops: Proceedings of the 29th National Heat Transfer Conference, Atlanta, Georgia, 8-11 Aug 1993* (ASME Heat Transfer Division, New York, 1993), pp. 1–17.

[14] T. Fujihara, T. Hoshino, S. Ogiwara, K. Eguchi, S. Sugimoto, and M. Shimizu, “Design and Evaluations of a Sodium Heat Pipe Model for Solar Stirling Power in Space,” in J. Andrews, A. Akbarzadeh, and I. Sauciuc, eds., *Heat Pipe Technology: Theory, Applications, and Prospects: Proceedings of the 5th International Heat Pipe Symposium, Melbourne, Australia, 17-20 November 1996* (Elsevier, Oxford, 1997), pp. 446–452.

[15] Y. Cao and A. Faghri, “Conjugate Modeling of High-Temperature Nosecap and Wing Leading Edge Heat Pipes,” J. Heat Transfer **115**, 819 (1993).

[16] D. Khrustalev and A. Faghri, “Thermal Analysis of a Micro Heat Pipe,” J. Heat Transfer **116**, 189 (1994).

[17] A. Miner and U. Ghoshal, “Cooling of High-Power-Density Microdevices Using Liquid Metal Coolants,” Appl. Phys. Lett. **85**, 506 (2004).

[18] A. Jiao, R. Riegler, H. Ma, and G. Peterson, “Thin Film Evaporation Effect on Heat Transport Capability in a Grooved Heat Pipe,” Microfluid Nanofluid **1**, 227 (2005).

[19] K.-Q. Ma and J. Liu, “Heat-driven liquid metal cooling device for the thermal management of a computer chip,” J. Phys. D: Appl. Phys. **40**, 4722 (2007).

[20] C. Sobhan, R. Rag, and G. Peterson, “A Review and Comparative Study of the Investigations on Micro Heat Pipes,” Int. J. Energy Res. **31**, 664 (2007).

[21] A. Jiao, H. Ma, and J. Critser, “Heat Transport Characteristics in a Miniature Flat Heat Pipe with Wire Core Wicks,” J. Heat Transfer **130** (2008), 051501.

[22] S.-K. Wee, K. D. Kihm, and K. P. Hallinan, “Effects of the Liquid Polarity and the Wall Slip on the Heat and Mass Transport Characteristics of the Micro-Scale Evaporating Transition Film,” Int. J. Heat Mass Transfer **48**, 265 (2005).

[23] G. Peterson and H. Ma, “Temperature Response of Heat Transport in a Micro Heat Pipe,” J. Heat Transfer **121**, 438 (1999).

[24] B. Badran, J. M. Albayyari, F. M. Gerner, P. Ramadas, H. T. Henderson, and K. W. Baker, “Liquid-Metal Micro Heat Pipes,” in *Heat Pipes and Capillary Pumped Loops: Proceedings of the 29th National Heat Transfer Conference, Atlanta, Georgia, 8-11 Aug 1993* (ASME Heat Transfer Division, New York, 1993), pp. 71–85.

[25] P. Ramadas, H. Henderson, B. Badran, F. Gerner, and K. Baker, “Liquid-Metal Micro Heat Pipes Incorporated in Waste-Heat Radiator Panels,” in M. El-Genk and M. D. Hoover, eds., *Proceedings*

of the 10th Symposium on Space Nuclear Power and Propulsion, Albuquerque NM, January 10-14, 1993 (American Institute of Physics, New York, 1993), pp. 551–557, CONF-930103.

[26] D. Pratt, J. Brown, and K. Hallinan, “Thermocapillary Effects on the Stability of a Heated, Curved Meniscus,” J. Heat Transfer **120**, 220 (1998).

[27] F. Holm and S. Goplen, “Heat Transfer in the Meniscus Thin-Film Region,” J. Heat Transfer **101**, 543 (1979).

[28] J. Schonberg and P. Wayner, Jr., “Analytical Solution for the Integral Contact Line Evaporative Heat Sink,” J. Thermophys Heat Transfer **6**, 128 (1991).

[29] A. Mirzamoghadam and I. Catton, “A Physical Model of the Evaporating Meniscus,” J. Heat Transfer **110**, 201 (1988).

[30] L. Swanson and G. Herdt, “Model of the Evaporating Meniscus in a Capillary Tube,” J. Heat Transfer **114**, 434 (1992).

[31] S. DasGupta, J. Schonberg, I. Kim, and P. Wayner, Jr., “Use of the Augmented Young-Laplace Equation to Model Equilibrium and Evaporating Extended Menisci,” J. Colloid Interface Sci. **157**, 332 (1993).

[32] S. DasGupta, J. Schonberg, and P. Wayner, Jr., “Investigation of an Evaporating Extended Meniscus Based on the Augmented Young-Laplace Equation,” J. Heat Transfer **115**, 201 (1993).

[33] S. DasGupta, I. Kim, and P. Wayner, Jr., “Use of the Kelvin-Clapeyron Equation to Model an Evaporating Curved Microfilm,” J. Heat Transfer **116**, 107 (1994).

[34] J. Schonberg, S. DasGupta, and P. Wayner, Jr., “An Augmented Young-Laplace Model of an Evaporating Meniscus in a Microchannel with High Heat Flux,” Exp. Therm Fluid Sci. **10**, 163 (1995).

[35] Q. He and K. Hallinan, “Thermocapillary Effects on the Evaporating Extended Meniscus,” ASME HTD **290**, 71 (1994).

[36] P. Wayner, Jr. and J. Schonberg, “Heat Transfer and Fluid Flow in an Evaporating Extended Meniscus,” in G. Hestroni, ed., *Heat Transfer 1990: Proceedings of the Ninth International Heat Transfer*

Conference, Jerusalem, Israel, 1990 (Hemisphere Publishing Corp., New York, 1990), pp. 228–234, vol. 4.

[37] B. Derjaguin, S. Nerpin, and N. Churaev, “Effect of Film Transfer Upon Evaporation of Liquids from Capillaries,” RILEM Bul. **29**, 93 (1965).

[38] R. W. Schrage, *A Theoretical Study of Interphase Mass Transfer* (Columbia University Press, New York, 1953).

[39] P. Wayner, Jr., Y. Kao, and L. LaCroix, “The Interline Heat-Transfer Coefficient of an Evaporating Wetting Film,” Int. J. Heat Mass Transfer **19**, 487 (1976).

[40] M. Potash, Jr. and P. Wayner, Jr., “Evaporation from a Two-Dimensional Extended Meniscus,” Int. J. Heat Mass Transfer **15**, 1851 (1972).

[41] H. Chebaro and K. Hallinan, “Boundary Conditions for an Evaporating Thin Film for Isothermal Interfacial Conditions,” J. Heat Transfer **115**, 816 (1993).

[42] H. Chebaro, K. Hallinan, S. Kim, and W. Chang, “Evaporation from a Porous Wick Heat Pipe for Isothermal Interfacial Conditions,” ASME HTD **221**, 23 (1992).

[43] K. Hallinan, H. Chebaro, S. Kim, and W. Chang, “Evaporation from an Extended Meniscus for Nonisothermal Interfacial Conditions,” J. Thermophys Heat Transfer **8**, 709 (1994).

[44] B. Derjaguin, N. Churaev, and V. Muller, *Surface Forces* (Consultants Bureau, New York, 1987).

[45] H. T. Davis, *Statistical Mechanics of Phases, Interfaces, and Thin Films* (VCH Publishers, Inc., New York, 1996).

[46] C. J. van Oss, *Interfacial Forces in Aqueous Media* (Taylor and Francis, New York, 2006), second ed.

[47] T. F. Tadros, “Steric Interactions in Thin Liquid Films,” in I. Ivanov, ed., *Thin Liquid Films: Fundamentals and Applications* (Marcel Dekker, Inc., New York, 1988), pp. 331–377.

[48] B. Derjaguin, L. Leonov, and V. Roldughin, “Disjoining Pressure in Liquid Metallic Films,” J. Colloid Interface Sci. **108**, 207 (1985).

[49] H. Hamaker, “London-van der Waals Attraction Between Spherical Particles,” Physica **4**, 1058 (1937).

[50] E. Lifshitz, “The Theory of Molecular Attractive Forces Between Solids,” Sov. Phys.-JETP (USA) **2**, 73 (1955).

[51] I. Dzyaloshinskii, E. Lifshitz, and L. Pitaevskii, “The General Theory of Van der Waals Forces,” Adv. Phys. **10**, 165 (1961).

[52] J. N. Israelachvili, *Intermolecular and Surface Forces: With Applications to Colloidal and Biological Systems* (Academic Press, New York, 1985).

[53] P. Wayner, Jr., “A Dimensionless Number for the Contact Line Evaporative Heat Sink,” J. Heat Transfer **111**, 813 (1989).

[54] J. G. Truong and P. C. Wayner, Jr., “Effects of Capillary and van der Waals Dispersion Forces on the Equilibrium Profile of a Wetting Liquid: Theory and Experiment,” J. Chem. Phys. **87**, 4180 (1987).

[55] P. C. Wayner, Jr., “Intermolecular Forces in Phase-Change Heat Transfer: 1998 Kern Award Review,” AlChE J. **45**, 2055 (1999).

[56] S. Moosman and G. Homsy, “Evaporating Menisci of Wetting Fluids,” J. Colloid Interface Sci. **73**, 212 (1979).

[57] B. Derjaguin and V. Roldughin, “Influence of the Ambient Medium on the Disjoining Pressure of Liquid Metallic Films,” Surf. Sci. **159**, 69 (1985).

[58] R. Kubo, “Electronic Properties of Metallic Fine Particles,” J. Phys. Soc. Jpn. **17**, 975 (1962).

[59] V. Roldughin, “Quantum-Size Colloid Metal Systems,” Russ. Chem. Rev. **69**, 821 (2000).

[60] V. Ajaev and D. Willis, “Thermocapillary Flow and Rupture in Films of Molten Metal on a Substrate,” Phys. Fluids **15**, 3144 (2003).

[61] V. S. Ajaev and D. A. Willis, “Heat Transfer, Phase Change, and Thermocapillary Flow in Films of Molten Metal on a Substrate,” Numer. Heat Transfer, Part A **50**, 301 (2006).

[62] P. Stephan and C. Busse, “Analysis of the Heat Transfer Coefficient of Grooved Heat Pipe Evaporator Walls,” Int. J. Heat Mass Transfer **35**, 383 (1992).

[63] D. Khrustalev and A. Faghri, “Heat Transfer During Evaporation on Capillary-Grooved Structures of Heat Pipes,” J. Heat Transfer **117**, 740 (1995).

[64] D. Khrustalev and A. Faghri, “Fluid Flow Effects in Evaporation From Liquid-Vapor Meniscus,” J. Heat Transfer **118**, 725 (1996).

[65] H. Kim, *Temperature and Velocity Field Mappings for Micro-Scale Heat and Mass Transport Phenomena*, Ph.D. thesis, Texas A&M University (2001).

[66] Y. Ji, Q.-S. Liu, and R. Liu, “Coupling of Evaporation and Thermocapillary Convection in a Liquid Layer with Mass and Heat Exchanging Interface,” Chinese Physics Letters **25**, 608 (2008).

[67] H. K. Cammenga, “Evaporation Mechanisms of Liquids,” Current Topics in Materials Science **5**, 335 (1980).

[68] J. Padday, “Cohesive Properties of Thin Films of Liquids Ahering to a Solid Surface,” Spec. Discuss. Faraday Soc. **1**, 64 (1970).

[69] J. Philip, “Unitary Approach to Capillary Condensation and Adsorption,” J. Chem. Phys. **66**, 5069 (1977).

[70] J. A. de Feijter, “Thermodynamics of Thin Liquid Films,” in I. Ivanov, ed., *Thin Liquid Films: Fundamentals and Applications* (Marcel Dekker, Inc., New York, 1988), pp. 1–48.

[71] P. C. Wayner Jr., “The Effect of Interfacial Mass Transport on Flow in Thin Liquid Films,” Colloids Surf. **52**, 71 (1991).

[72] Y. Kamotani, “Evaporator Film Coefficients of Grooved Heat Pipes,” in *Proceedings of the International Heat Pipe Conference, 3rd, Palo Alto, California, 22-24 May 1978* (AIAA, New York, 1978), pp. 128–130.

[73] P. Drüde, *The Theory of Optics* (Longmans, Green, and Co., New York, 1902).

[74] H. Bennett and J. Bennett, “Validity of the Drüde Theory for Silver, Gold and Aluminum in the Infrared,” in F. Abelès, ed., *Optical Properties and Electronic Structure of Metals and Alloys* (North-Holland Publishing Company, Amsterdam, 1966), pp. 175–195, proceedings of the International Colloquium held at Paris, 13-16 September 1965.

[75] J. Hodgson, “The Optical Properties of Liquid Metals,” in S. Z. Beer, ed., *Liquid Metals: Chemistry and Physics* (Marcel Dekker, Inc., New York, 1972), pp. 331–371.

[76] T. Inagaki, E. Arakawa, R. Birkhoff, and M. Williams, “Optical Properties of Liquid Na between 0.6 and 3.8 eV,” Phys. Rev. B **13**, 5610 (1976).

[77] N. March and B. Paranjape, “Mean Free Path Effects in the Dielectric Function of a Liquid Metal,” Phys. Chem. Liq. **17**, 55 (1987).

[78] M. Shimoji, *Liquid Metals: An Introduction to the Physics and Chemistry of Metals in the Liquid State* (Academic Press, New York, 1977).

[79] W. Takens, W. Mischke, J. Korving, and J. Beenakker, “A Spectroscopic Study of Free Evaporation of Sodium,” in *Rarefied Gas Dynamics: Proceedings of the 14th International Symposium on Rarefied Gas Dynamics, Tsukuba, Science City, Japan, 16-20 July 1984* (Univ. of Tokyo Press, Tokyo, 1984), pp. 967–974.

[80] J. Wilson, “The Structure of Liquid Metals and Alloys,” Met. Revs. **10**, 381 (1965).

[81] J. Fink and L. Leibowitz, “A Consistent Assessment of the Thermophysical Properties of Sodium,” High Temp. Mater. Sci. **35**, 65 (1996).

[82] L. Landau and E. Lifshitz, *Electrodynamics of Continuous Media* (Pergamon Press, Oxford, 1960).

[83] T. Coutts, *Electrical Conductivity in Thin Metal Films* (Elsevier Scientific Publishing Company, New York, 1974).

[84] J. I. Frankel, “ME599 Special Topics: Advanced Engineering Analysis,” Unpublished Lecture Notes (2007), The University of Tennessee, Knoxville.

[85] A. Guillou and F. Soull, “La Resolution Numerique des Problemes Differentielles aux Conditions Initials par des Methodes de Collocation,” RAIRO Inform. Théor. Appl. **3**, 17 (1969).

[86] K. Wright, “Some Relationships Between Implicit Runge-Kutta, Collocation and Lanczos T-method, and Their Stability Properties,” BIT **10**, 217 (1970).

[87] T. J. Rivlin, *The Chebyshev Polynomials* (Wiley, New York, 1974).

[88] L. C. Andrews, *Special Functions of Mathematics for Engineers* (McGraw-Hill, New York, 1992), second ed.

[89] R. Bracewell, *The Fourier Transform and Its Applications* (McGraw-Hill, New York, 2000), third ed.

[90] P. Dirac, *The Principles of Quantum Mechanics* (Oxford University Press, Oxford, 1958), fourth ed.

[91] L. Delves and J. Mohamed, *Computational Methods for Integral Equations* (Cambridge University Press, New York, 1985).

[92] D. W. Marquardt, “An Algorithm for Least-Squares Estimation of Nonlinear Parameters,” J. Soc. Indust. Appl. Math. **11**, 431 (1963).

[93] E. J. Henley and E. M. Rosen, *Material and Energy Balance Computations* (John Wiley and Sons, New York, 1969).

[94] G. Ratz and K. Brickner, “Effect of Liquid Sodium on Welded AISI Type-304 Stainless-Steel Pipe,” Nucl. Appl. **4**, 154 (1968).

[95] M. Barker and D. Wood, “The Corrosium of Cromium, Iron, and Stainless Steel in Liquid Sodium,” J. Less-Common Met. **35**, 315 (1974).

[96] J. Moberly, M. Barlow, M. Garrison, and P. Planting, “Electron Microscope Observations of Liquid Sodium Interaction with 304 Stainless Steel,” J. Nucl. Mater. **29**, 223 (1969).

[97] M. Barlow and P. J. Planting, “The Wetting of Metal Surfaces by Liquid Mercury,” Z. Metallkde. **60**, 817 (1969).

[98] B. Longson and J. Prescott, “Some Experiment on the Wetting of Stainless Steel, Nickel and Iron in Liquid Sodium,” in H. Oguchi, ed., *Proceedings of the International Conference on Liquid Alkali Metals, Nottingham, UK, 4-6 April 1973* (British Nuclear Energy Society, London, 1973), pp. 171–176, vol. II.

[99] F. Hensel and J. William W. Warren, *Fluid Metals: The Liquid-Vapor Transition of Metals* (Princeton University Press, Princeton, New Jersey, 1999).

[100] W. Louden and K. Schmidt, “High-Pressure Sodium Discharge Arc Lamps,” Illum. Eng. **60**, 696 (1965).

[101] E. da C. Andrade and L. Tsien, “On Surface Cracks in Glasses,” Proc. R. Soc. Lond. (UK) **159A**, 346 (1937).

[102] J. F. Waymouth, *Electric Discharge Lamps* (MIT Press, Cambridge, Massachusetts, 1971).

[103] J. de Groot and J. van Vliet, *The High-Pressure Sodium Lamp* (MacMillan Education LTD., London, 1986).

[104] J.-H. Ahn and A. Berghezan, “Scanning Electron Microscopy of Liquid Metal Infiltration of Capillaries,” Mater. Sci. Tech. Ser. **7**, 643 (1991).

[105] D. M. Pratt and K. P. Hallinan, “Thermocapillary Effects on the Wetting Characteristics of a Heated Curved Meniscus,” J. Thermophys Heat Transfer **11**, 519 (1997).

[106] J. Burelbach, S. Bankoff, and S. Davis, “Nonlinear Stability of Evaporating/Condensing Liquid Films,” J. Fluid Mech. **195**, 463 (1988).

[107] H. Ma, G. Peterson, and D. Pratt, “Disjoining Pressure Effect on the Wetting Characteristics in a Capillary Tube,” Microscale Therm. Eng. **2**, 283 (1998).

[108] A. E. Ruggles, “Micro-scale Liquid Metal Evaporator/Vapor Generator Designs for Earth and Space Power Applications,” in *American Nuclear Society Embedded Topical Meeting - 2005 Space Nuclear Conference* (American Nuclear Society, San Diego, 2005), pp. 178–187.

[109] K. D. Kihm (2008), Personal Communications.

[110] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numerical Recipes in FORTRAN* (Cambridge University Press, Cambridge, 1992), second ed.

Appendices

Appendix A

Hamaker Constant Calculation

The Hamaker constant is found by numerically solving Equation (3.2). The complex dielectric permittivities are modeled using classical Drüde theory where the liquid sodium coolant is given as Equation (3.6), the SS304 substrate is given as Equation (3.7), and the liquid sodium vapor is taken to be completely dielectric with a complex dielectric permittivity of unity. The numerical calculations are performed within the Maple$^{\text{TM}}$ computer algebra system.

```
# Using Maple v12.0
> restart;
> with(ScientificConstants);
> GetConstant(h);
                                                              -34
           Planck_constant, symbol = h, value = 6.62606876 10   ,

                                 -41
             uncertainty = 5.2 10   , units = J s
> GetConstant(N[A]);
                                                      A[r](e) M[u]
           Avogadro_constant, symbol = N[A], derive = ------------
                                                          m[e]
> GetConstant(epsilon[0]);
                                                                 1
       permittivity_of_vacuum, symbol = epsilon[0], derive = --------
                                                                     2
                                                             mu[0] c
> GetConstant(m[e]);
                                                  2 R[infinity] h
           electron_mass, symbol = m[e], derive = ---------------
                                                             2
                                                     c alpha
> GetConstant(e);
```

```
                                                                (1/2)
                                                 (1/2) /alpha h\
        elementary_charge, symbol = e, derive = 2      |-------|
                                                       \mu[0] c/
> GetConstant(k);
                                                          R
                Boltzmann_constant, symbol = k, derive = ----
                                                         N[A]
> h := evalf(Constant(h));
                                           -34
                              6.62606876 10
> N[A] := evalf(Constant(N[A]));
                                            23
                              6.022141986 10
> epsilon[0] := evalf(Constant(epsilon[0]));
                                            -12
                              8.854187815 10
> m := evalf(Constant(m[e]));
                                            -31
                              9.109381882 10
> e := evalf(Constant(e));
                                            -19
                              1.602176462 10
> k := evalf(Constant(k));
                                            -23
                              1.380650277 10
# Define electromagnetic wave frequency at the melting point of sodium:
> v[n] := (2*evalf(Pi)*k*1154.7)/h;
                                            14
                              1.511738871 10
> omega[n] := 2*evalf(Pi)*v[n];
                                            14
                              9.498535464 10
# Define the dielectric permittivity of the sodium metal using Drude Theory with Relaxation Time:
> A[Na] := 0.2299e-1;
                                   0.02299
> rho[Na] := 742.8591;
                                  742.8591
> Z[Na] := 1;
                                      1
> sigma[Na] := 0.253605e7;
                                           6
                                 2.53605 10
> tau[3] := 1.17*m*sigma[Na]*A[Na]/(.85*N[A]*Z[Na]*rho[Na]*e^2);
                                            -15
                              6.366115225 10
> omega[3] := sqrt(.85*N[A]*Z[Na]*rho[Na]*e^2/(1.17*A[Na]*epsilon[0]*m));
                                            15
                              6.707602118 10
> v[3] := omega[3]/(2*evalf(Pi));
                                            15
                              1.067548033 10
> epsilon[3] := 1+omega[3]^2*tau[3]*(1-x*tau[3])/(x*(1-x^2*tau[3]^2));
                                 17 /                  -15  \
                   2.864237862 10   \1 - 6.366115225 10    x/
```

```
               1 + ------------------------------------------
                            /                  -29  2\
                          x \1 - 4.052742306 10    x /
# Define the dielectric permittivity of the SS304 metal using Drude theory without Relaxation Time:
> A[SS] := 0.5481e-1;
                                   0.05481
> rho[SS] := 8000;
                                    8000
> Z[SS] := 1.79;
                                    1.79
> omega[1] := sqrt(N[A]*Z[SS]*rho[SS]*e^2/(A[SS]*epsilon[0]*m));
                                            16
                              2.237734878 10
> v[1] := omega[1]/(2*evalf(Pi));
                                            15
                              3.561465671 10
> epsilon[1] := 1+(omega[1]/x)^2;
                                              32
                                5.007457384 10
                            1 + ----------------
                                        2
                                       x
# Define the dielectric permittivity of the sodium vapor as that of a vacuum:
> epsilon[2] := 1;
                                      1
# Solve for the Hamaker Constant (via Israelachvili)
> freq := (epsilon[1]-epsilon[3])*(epsilon[2]-epsilon[3])/((epsilon[1]+epsilon[3])*(epsilon[2]+epsilon[3]));
  /                 /              32
  |              17 |5.007457384 10
- |2.864237862 10   |----------------
  |                 |        2
  \                 \       x

                   17 /                  -15  \\                          \//
     2.864237862 10   \1 - 6.366115225 10    x/| /                  -15  \| |
   - ------------------------------------------| \1 - 6.366115225 10    x/| |
              /                  -29  2\       |                          | |
            x \1 - 4.052742306 10    x /       /                          / \

  /                  32                  17 /                  -15  \\
  |    5.007457384 10      2.864237862 10   \1 - 6.366115225 10    x/|   /
  |2 + ---------------- + ------------------------------------------| x \1
  |            2                   /                  -29  2\       |
  \           x                  x \1 - 4.052742306 10    x /       /

                           /                  17 /                  -15  \\\
                   -29  2\ |    2.864237862 10   \1 - 6.366115225 10    x/||
   - 4.052742306 10    x / |2 + ------------------------------------------||
                           |             /                  -29  2\       ||
                           \           x \1 - 4.052742306 10    x /       //
> A := 3*h*(int(freq, x = omega[n] .. infinity))/(8*evalf(Pi)^2);
                                            -19
                             -1.015143464 10
>
```

Appendix B

Dispersion Force Calculation

The retarded dispersion force curve for a liquid sodium thin film on a stainless steel substrate, shown in Figure 3.1, is found via solution of Equation (3.1). The complex dielectric permittivities are modeled using classical Drüde theory where the liquid sodium coolant is given as Equation (3.6), the SS304 substrate is given as Equation (3.7), and the liquid sodium vapor is taken to be completely dielectric with a complex dielectric permittivity of unity. The numerical calculations are performed within the MATLAB® programming environment using adaptive Lobatto quadrature with a relative convergence of 1×10^{-6}.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%
% File:     dispersion_force.m
% Language: MATLAB
% Author:   jtipton2
% Date:     12/23/2008
% Summary:  Calculates dispersion force curve using DLP theory for the case of
%           liquid sodium thin film on a stainless steel substrate.
%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
clear all
clc
syms p n delta w
%
% LOAD PHYSICAL CONSTANTS
%
h = 6.62606876e-34;                                     % Planck's Constant  (m^2 kg/s)
N_A = 6.022141986e23;                                   % Avogadro's Number (atoms/mole)
epsilon_0 = 8.854187815e-12;                            % Permittivity of Free Space  (s^4 A^2 / m^3 / kg) NOTE: A = C/s
m = 9.109381882e-31;                                    % Electron Mass  (kg)
e = 1.602176462e-19;                                    % Electron Charge  (C)
k = 1.380650277e-23;                                    % Boltzmann's Constant  (m^2 kg / s^2 / K)
c = 2.99792458e8;                                       % Speed of Light in Vacuum  (m/s)
```

```
R_g = 8.314472;                                         % universal gas constant (N-m/K-mol)
%
% DEFINE ELECTROMAGNETIC WAVE FREQUENCY AT SODIUM MELTING POINT
%
T = 1154.7;                                             % Melting point of Sodium (K)
T_v = T;                                                % vapor temperature (K)
omega_n = 4*pi^2*n*k*T/h;                               % frequency of incoming radiation (rad/s)
%
% DEFINE DIELECTRIC PERMITTIVITY OF SODIUM
% (Drude Model)
%
rho_Na = 219.00 + 275.32 * (1 - T/2503.7) + ...
    511.58 * sqrt(1 - T/2503.7);                        % kg/m^3        U ~ 0.4%
sigma_Na = 25.3605e+5;                                  % (S/m or 1/Ohm-m or s^3 A^2/m^3/kg)
M_Na = 22.99/1000;                                      % (kg/mol)
NV_Na = 1;                                              % valence electrons/atom
A_Na = 0.02299;                                         % atomic weight (kg/mol)
m_eff_Na = 1.17 * m;                                    % effective mass (kg)
edensity_Na = 0.85*N_A*rho_Na/A_Na;                     % valence electron density (electrons/m^3)
omega_3 = sqrt(edensity_Na*e^2/epsilon_0/m_eff_Na);     % plasma frequency of free electron gas (rad/s)
tau_3 = m_eff_Na*sigma_Na/edensity_Na/e^2;              % relaxation factor (s)
epsilon_3 = 1 + omega_3^2 * tau_3 * (1 - omega_n*tau_3) ...
    / omega_n / (1 - omega_n^2*tau_3^2);                % complex dielectric permittivity of Na
%
% DEFINE DIELECTRIC PERMITTIVITY OF SS304
% (Drude Model - No Relaxation Time)
%
A_SS = 0.05481;                                         % atomic weight kg/mol
rho_SS = 8000;                                          % kg/m^3
NV_SS = 1.79;                                           % valence electrons/molecule
omega_1 = sqrt(N_A*rho_SS*e^2/A_SS/epsilon_0/m);        % plasma frequency of free electron gas (rad/s)
epsilon_1 = 1 + (omega_1/omega_n)^2;                    % complex dielectric permittivity of SS304
%
% DEFINE DIELECTRIC PERMITTIVITY OF SODIUM VAPOR
%
epsilon_2 = 1;
%
% HAMAKER "CONSTANT" CALCULATION
% modified Lifshitz theory created by Israelachvili (p. 142, Eq 11.8)
% corresponds to DLP asymptotic expression for small delta (p.191, Eq 4.18)
%
A = -1.015143464e-19;
%__________________________________________________________________________
%
% DLP THEORY CALCULATION FOR DISPERSION FORCE
% (force between SS204 and vapor when separated by liquid Na)
%__________________________________________________________________________
%

s_1 = sqrt(epsilon_1/epsilon_3 - 1 + p^2);
s_2 = sqrt(epsilon_2/epsilon_3 - 1 + p^2);

inside_int = p^2 * ((s_1 + p)*(s_2 + p)*exp(2*p*omega_n*sqrt(epsilon_3)*delta/c)/(s_1 - p)/(s_2 - p) - 1)^-1 ...
    + p^2 * ((s_1 + p*epsilon_1/epsilon_3)*(s_2 + p*epsilon_2/epsilon_3)*exp(2*p*omega_n*sqrt(epsilon_3)*delta/c) ...
    /(s_1 - p*epsilon_1/epsilon_3)/(s_2 - p*epsilon_2/epsilon_3) - 1)^-1;
```

```
inside_int = eval(['@(p,n,delta)' vectorize(inside_int)]);

%l = linspace(1e-9,1e-6,100);
%l = logspace(-8,log10(3)-6,100);
l = logspace(-9,log10(7.5)-7,80);

warning off all
for i = 1:length(l)
    F_diff = 1;
    F_old = 1;
    F = 0;
    j = 1;
    while abs(F_diff) > 0.000001
        if l(i) <= 2.1e-7
            tol = 1.e-10;
        elseif l(i) > 2.1e-7 & l(i) <= 6.8e-7
            tol = 1.e-20;
        else
            tol = 1.e-90;
        end
        F = F + (k*T/pi/c^3) * subs(epsilon_3,n,j)^1.5 * subs(omega_n,n,j)^3 * quad(inside_int,1,2000,tol,[],j,l(i));
        F_diff = (F - F_old)/F_old;
        F_old = F;
        j = j + 1;
    end
    disp(j);
    F_l(i) = F;
end

format long g

[l',-F_l',-A./6./pi./l'.^3]

figure(1)
loglog(l,-F_l,'bo',l,-A./6./pi./l.^3,'r--')
xlabel('\delta␣(m)');
ylabel('-F(\delta)␣␣␣(N/m^2)');
xlim([1e-9 1e-6]);
legend('DLP␣Theory','Hamaker␣Approx.','Location','NorthEast')

%
% Print data to file for use in FORTRAN program.
%
fid = fopen('DISPERSION_DATA','wt');
fprintf(fid,'%15.15E',-F_l);
```

Appendix C

Thin Film Solutions

The following three programs solve Equation (4.1) for the unique case of high temperature, liquid metal, extended meniscus evaporation on a stainless steel substrate. The first program, Appendix C.1, solves the thin film governing equation where the electronic component of the disjoining pressure is large enough to render the dispersion force component negligible. This represents Cases A-D in Table 6.1. The second program, Appendix C.2, solves the thin film governing equation where the electronic and dispersion force components of the disjoining pressure are of equal order of magnitude. This represents Cases E and G in Table 6.1. The third program, Appendix C.3, solves the thin film governing equation where the electronic component of the disjoining pressure is not present. This represents Case F in Table 6.1. These programs were written in the FORTRAN 90 programming language and were compiled on a Sun Microsystems Sun Fire$^{\text{TM}}$ V880 server running the Solaris® 10 operating system using quadruple precision. The FORTRAN modules they require are listed in Appendix D.

C.1 Disjoining Pressure Cases A-D

```
PROGRAM HTLMTF_7B2
!************************************************************************************************************************
!
! File:      HTLMTF_7B2.f90
! Language: FORTRAN 90
! Author:    jtipton2
! Summary:   Thin film model of K.P. Hallinan et al.,  "Evaporation from an Extended Meniscus for Nonisothermal
!            Interfacial Conditions,"  Journal of Thermophysics and Heat Transfer,  Vol. 8,  1994,  pp. 709-716.
```

```
!           + LIQUID METAL PROPERTIES
!           + LECTRONIC COMPONENT OF THE DISPERSION FORCE **ONLY**
!           + WITH NONLINEARITIES SOLVED VIA THE LEVENBURG-MARQUARDT METHOD
!
! Redord of Revision:
!   Date        Programmer          Description of Change
!   ====        ==========          =====================
!    3/31/08    JBT                 Initial creation
!    1/02/09    JBT                 Changed T_V from 1156.09_16 to 1154.7_16
!                                   Moved all subroutines to modules to improve programming
!                                   Changed INDX to an integer to match the subroutine
!                                   Added LABEL parameter and used it to print result headers
!************************************************************************************************************************
    USE MATSOLV
    USE CUBICSPLINE
    USE CHEBYSHEV
    USE LMPROPERTIES
    IMPLICIT NONE
    INTEGER, PARAMETER :: NN_MAX = 200, TERMS = 200
    INTEGER :: II, JJ, KK, NN, LOOPNUM, INDX(NN_MAX)
    REAL*16, PARAMETER :: PI = 3.14159265358979_16
    CHARACTER (LEN=*), PARAMETER :: FORM1 = "(//␣1X␣A␣/␣1X,␣50('='))"
    CHARACTER (LEN=40), PARAMETER :: LABEL = "(//8(15X,A)/␣1X,␣50('='))"
    REAL*16 :: T_V, R_G, V_L, R, C_L, DT_0, ALPHA, BETA, GAMMA, LAMBDA, COEFF, B_ELEC, CHI, RHO, H_FG, MU, P_V, &
               SIGMA, K, MW, H_0, PI_0, X_0, M_ID, U_0, CA, KAPPA
    REAL*16 :: T_BC(NN_MAX+4), T_I_BC(NN_MAX+4), T_II_BC(NN_MAX+4), T_III_BC(NN_MAX+4), &
               T(NN_MAX+4), T_I(NN_MAX+4), T_II(NN_MAX+4), T_III(NN_MAX+4), T_IIII(NN_MAX+4), &
               U, U_I, U_II, U_III, U_IIII, DU, DU_I, DU_II, DU_III, DU_IIII, PI_STAR, PI_STAR_I, PI_STAR_II
    REAL*16 :: C_OLD(NN_MAX), C_NEW(NN_MAX), DC(NN_MAX), JACOBIAN(NN_MAX,NN_MAX), F(NN_MAX), Q_OLD, Q_NEW, MARQ
    REAL*16 :: JACOBIAN_C_PLUS_F(NN_MAX), G(NN_MAX,NN_MAX), GRADQ(NN_MAX), G_PLUS_MARQ_I(NN_MAX,NN_MAX)
    REAL*16 :: IDENTITY(NN_MAX,NN_MAX), ERROR, XI, KEY
    REAL*16 :: ETA(TERMS), THETA(TERMS,5), X(TERMS), H(TERMS,5), M_EVP(TERMS), DPDX(TERMS)
    REAL*16 :: Q(NN_MAX), OMEGA(NN_MAX)
    REAL*16 :: NONISO_DIFF, NONDTEMP

    ! Program Constants
    T_V = 1154.7_16                                     ! sodium temperature of vaporization (K)
    CALL LM_PROPS (T_V, RHO, K, H_FG, MU, MW, P_V, SIGMA, B_ELEC)
    R_G = 8.314472_16                                   ! universal gas constant (N-m/K-mol)
    V_L = MW/RHO                                        ! liquid molar volume (m^3/mol)
    R = 400E-6_16                                       ! radius or width of pore (m)
    C_L = 2._16                                         ! accomodation coefficient
    DT_0 = 5E-4_16                                      ! wall/vapor temperature difference (K)
    CHI = 3._16*PI**2._16/16._16                        ! electronic disjoining pressure boundary condition term

    ! Nondimensional Variables and Scales:
    H_0 = SQRT(V_L*T_V*B_ELEC*CHI/MW/H_FG/DT_0)         ! reference film thickness of the adsorbed film (m)
    PI_0 = MW*H_FG*DT_0/V_L/T_V                         ! reference disjoining pressure (N/m^2)
    X_0 = SQRT(SIGMA*H_0/PI_0)                          ! axial length scale (m)
    M_ID = C_L * SQRT(MW/2._16/PI/R_G/T_V) * P_V*MW*H_FG*DT_0/(R_G*T_V*(T_V+DT_0))   ! ideal evaporative flux (kg/s/m^2)
    U_0 = M_ID/RHO                                      ! liquid characteristic velocity (m/s)
    CA = MU*U_0/SIGMA                                   ! capillary number
    KAPPA = H_FG*M_ID*H_0/K                             ! ratio of evaporative interfacial resistance
                                                        ! to conductive resistance in the thin film

```

```
! Normalized problem boundary conditions
ALPHA = 1.04_16
BETA = 1E-4_16
GAMMA = X_0**2._16/R/H_0
LAMBDA = 20._16
COEFF = 3._16 * CA / (H_0*PI_0/SIGMA)**2._16

! Create Chebyshev polynomials of the first kind
! for the boundary conditions
CALL CHEBY_T (NN_MAX+4,0,-1._16,T_BC)
CALL CHEBY_T (NN_MAX+4,1,-1._16,T_I_BC)
CALL CHEBY_T (NN_MAX+4,2, 1._16,T_II_BC)
CALL CHEBY_T (NN_MAX+4,3,-1._16,T_III_BC)



DO NN = 1,100,99
    ERROR = 1._16
    LOOPNUM = 0
    C_NEW = 0._16
    Q_NEW = 0._16
    Q = 0._16

    DO II = 1,NN
        DO JJ = 1,NN
            IF (II==JJ) THEN
                IDENTITY(II,JJ) = 1._16
            ELSE
                IDENTITY(II,JJ) = 0._16
            END IF
        END DO
    END DO

    PRINT FORM1,'Q = '

    DO WHILE (ERROR >= 1E-8_16 .AND. Q_NEW < 0.1E17_16)
        LOOPNUM = LOOPNUM + 1
        C_OLD = C_NEW
        Q_OLD = Q_NEW
        DO II = 1,NN
            !
            ! The "i" subscript refers to matrix rows which represent the
            ! functions evaluated at different values of the collocated domain
            ! variable, "XI".
            !
            XI = COS((2._16*II - 1._16)*PI/2._16/NN)
            !
            ! Create Chebyshev Polynomials of the First Kind
            ! and Their Derivatives
            !
            CALL CHEBY_T (NN_MAX+4,0,XI,T)
            CALL CHEBY_T (NN_MAX+4,1,XI,T_I)
            CALL CHEBY_T (NN_MAX+4,2,XI,T_II)
            CALL CHEBY_T (NN_MAX+4,3,XI,T_III)
            CALL CHEBY_T (NN_MAX+4,4,XI,T_IIII)
```

```
!
! Establish the Approximate Analytical Series Solution
! and It's Derivatives
!
U = ALPHA + LAMBDA*BETA*(XI + 1._16) + (0.5_16*XI**2._16 + XI + 0.5_16)*GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T(5:NN+4) &
- T_BC(5:NN+4) &
- (XI + 1._16)*T_I_BC(5:NN+4) &
- (0.5_16*XI**2._16 + XI + 0.5_16)*T_II_BC(5:NN+4) &
+ (-XI**3._16 + 3._16*XI**2._16 + 9._16*XI + 5._16)*T_III_BC(5:NN+4)/6._16))

U_I = LAMBDA*BETA + (1._16 + XI)*GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T_I(5:NN+4) &
- T_I_BC(5:NN+4) &
- (XI + 1._16)*T_II_BC(5:NN+4) &
+ (-0.5_16*XI**2._16 + XI + 1.5_16)*T_III_BC(5:NN+4)))

U_II = GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T_II(5:NN+4) &
- T_II_BC(5:NN+4) &
+ (1._16 - XI)*T_III_BC(5:NN+4)))

U_III = SUM(C_OLD(1:NN) * (T_III(5:NN+4) - T_III_BC(5:NN+4)))

U_IIII = SUM(C_OLD(1:NN) * (T_IIII(5:NN+4)))

!
! Create the [F] matrix
!
NONISO_DIFF = DT_0 + KAPPA*U
NONDTEMP = (DT_0 + KAPPA*(U*U_II/LAMBDA**2._16 + 1._16/U)) / (DT_0 + KAPPA*U)

F(II) = 2._16*U_II/LAMBDA**2._16 &
      - 3._16*U**2._16*U_I*U_III/LAMBDA**4._16 &
      - U**3._16*U_IIII/LAMBDA**4._16 &
      - COEFF * (NONDTEMP - U_II/LAMBDA**2._16 - 1._16/U**2._16)

!
! Create the [E] matrix
!
DO JJ = 1,NN

    DU = T(JJ+4) - T_BC(JJ+4) - (XI + 1._16)*T_I_BC(JJ+4) &
            - (0.5_16*XI**2._16 + XI + 0.5_16)*T_II_BC(JJ+4) &
            + (-XI**3._16 + 3._16*XI**2._16 + 9._16*XI + 5._16)*T_III_BC(JJ+4)/6._16
    DU_I = T_I(JJ+4) - T_I_BC(JJ+4) - (XI + 1._16)*T_II_BC(JJ+4) + (-0.5_16*XI**2._16 + XI + 1.5_16) &
            *T_III_BC(JJ+4)
    DU_II = T_II(JJ+4) - T_II_BC(JJ+4) + (1._16 - XI)*T_III_BC(JJ+4)
    DU_III = T_III(JJ+4) - T_III_BC(JJ+4)
    DU_IIII = T_IIII(JJ+4)

    JACOBIAN(II,JJ) = (6._16*U*U_I*U_III/LAMBDA**4._16) * DU &
                    + (3._16*U**2._16*U_III/LAMBDA**4._16) * DU_I &
                    + (3._16*U**2._16*U_I/LAMBDA**4._16) * DU_III &

```

```
                        + (3._16*U**2._16*U_IIII/LAMBDA**4._16) * DU &
                        + (U**3._16/LAMBDA**4._16) * DU_IIII &

                        - (2._16/LAMBDA**2._16) * DU_II &

                        - (COEFF/LAMBDA**2._16) * DU_II &

                        + (2._16*COEFF/U**3._16) * DU &

                        !
                        ! Nonisothermal interface terms
                        !
                        + (COEFF*DT_0*KAPPA/NONISO_DIFF**2._16) * DU &

                        + (COEFF*KAPPA*U_II/LAMBDA**2._16/NONISO_DIFF) * DU &
                        + (COEFF*KAPPA*U/LAMBDA**2._16/NONISO_DIFF) * DU_II &
                        + (COEFF*KAPPA**2._16*U*U_II/LAMBDA**2._16/NONISO_DIFF**2._16) * DU &

                        - (COEFF*KAPPA*B_ELEC/H_0**2._16/PI_0/U**2._16/NONISO_DIFF) * DU &
                        - (COEFF*KAPPA**2._16*B_ELEC/H_0**2._16/PI_0/U/NONISO_DIFF**2._16) * DU

    END DO
END DO

!**LEVENBERG-MARQUARDT METHOD**********************************************
!
! Formulation described by:
! Henley and Rosen,  "Material and Energy Balance Computations,"
! John Wiley & Sons, 1969, pp. 171-173, 192-204.
!
!
! Solution procedure described by:
! "Numerical Recipes in Fortran:  The Art of Scientific Computing"
! 2nd Edition, pp. 679.
!
! (1)  compute Q
! (2)  pick MARQ = 0.001
! (3)  solve linear system (G + MARQ*I) DC = -GRADQ
! (4)  compute new Q(C) = SUM (JACOBIAN DC + F)^2
! (4a) if Q_NEW >= Q_OLD then MARQ = MARQ*10 | goto (3)
! (4b) if Q_NEW <  Q_OLD then MARQ = MARQ/10 | goto (3)
! (5)  if Q_NEW < Q_OLD AND Q_NEW < tol then STOP
!
!**************************************************************************
IF (LOOPNUM == 1) THEN
    Q_NEW = SQRT(DOT_PRODUCT(F,F))
    Q_OLD = 1._16
    MARQ = 0.0001_16
ELSE
    JACOBIAN_C_PLUS_F = MATMUL(JACOBIAN,C_OLD) + F
    Q_NEW = SQRT(DOT_PRODUCT(JACOBIAN_C_PLUS_F,JACOBIAN_C_PLUS_F))
END IF

IF (Q_NEW .GE. Q_OLD) THEN
    MARQ = MARQ * 10._16
```

```
        ELSE
            MARQ = MARQ / 10._16
        END IF

        G = 2._16 * MATMUL(TRANSPOSE(JACOBIAN),JACOBIAN)
        GRADQ = 2._16 * MATMUL(TRANSPOSE(JACOBIAN),F)
        G_PLUS_MARQ_I = G + MARQ*IDENTITY
        CALL LUDCMP (G_PLUS_MARQ_I,NN,NN_MAX,INDX,KEY)
        DC = GRADQ
        CALL LUBKSB (G_PLUS_MARQ_I,NN,NN_MAX,INDX,DC)
        C_NEW = DC + C_OLD

        ERROR = ABS(Q_NEW-Q_OLD)/Q_OLD
        Q(LOOPNUM) = Q_NEW
        PRINT '(1G32.16)', Q(LOOPNUM)

    END DO

    !
    ! SPATIAL CONVERGENCE ACCURACY
    ! (Integrate approximate analytical solution over the domain space)
    !
    OMEGA(NN) = 2._16*ALPHA + 2._16*LAMBDA*BETA + 4._16*LAMBDA**2._16*GAMMA/3._16

    DO JJ = 1,NN/2
        KK = 2._16*JJ
        OMEGA(NN) = OMEGA(NN) + C_NEW(KK) * ( -2._16*T_BC(KK) - 2._16*T_I_BC(KK) - 5._16*T_II_BC(KK)/3._16 &
                                             + 2._16*T_III_BC(KK) - 2._16/(KK+1._16)/(KK-1._16) )
    END DO

END DO

    PRINT FORM1,'OMEGA = '
    PRINT '(1G32.16)', OMEGA(1:NN-1)

    !
    ! CONVERT TO ORIGINAL NONDIMENSIONALIZED THIN FILM EVAPORATION EQUATION
    !
    DO II = 1,TERMS
        XI = COS((2._16*II - 1._16)*PI/2._16/TERMS)
        ETA(II) = LAMBDA*(1._16 + XI)
        !
        ! Create Chebyshev Polynomials of the First Kind
        ! and Their Derivatives
        !
        CALL CHEBY_T (NN_MAX+4,0,XI,T)
        CALL CHEBY_T (NN_MAX+4,1,XI,T_I)
        CALL CHEBY_T (NN_MAX+4,2,XI,T_II)
        CALL CHEBY_T (NN_MAX+4,3,XI,T_III)
        CALL CHEBY_T (NN_MAX+4,4,XI,T_IIII)
        !
        ! Establish the Approximate Analytical Series Solution
        ! and It's Derivatives
        !
        U = ALPHA + LAMBDA*BETA*(XI + 1._16) + (0.5_16*XI**2._16 + XI + 0.5_16)*GAMMA*LAMBDA**2._16 &
```

```
        + SUM(C_OLD(1:NN) * (T(5:NN+4) &
        - T_BC(5:NN+4) &
        - (XI + 1._16)*T_I_BC(5:NN+4) &
        - (0.5_16*XI**2._16 + XI + 0.5_16)*T_II_BC(5:NN+4) &
        + (-XI**3._16 + 3._16*XI**2._16 + 9._16*XI + 5._16)*T_III_BC(5:NN+4)/6._16))

        U_I = LAMBDA*BETA + (1._16 + XI)*GAMMA*LAMBDA**2._16 &
        + SUM(C_OLD(1:NN) * (T_I(5:NN+4) &
        - T_I_BC(5:NN+4) &
        - (XI + 1._16)*T_II_BC(5:NN+4) &
        + (-0.5_16*XI**2._16 + XI + 1.5_16)*T_III_BC(5:NN+4)))

        U_II = GAMMA*LAMBDA**2._16 &
        + SUM(C_OLD(1:NN) * (T_II(5:NN+4) &
        - T_II_BC(5:NN+4) &
        + (1._16 - XI)*T_III_BC(5:NN+4)))

        U_III = SUM(C_OLD(1:NN) * (T_III(5:NN+4) - T_III_BC(5:NN+4)))

        U_IIII = SUM(C_OLD(1:NN) * (T_IIII(5:NN+4)))
        !
        ! Convert to from THETA_HAT to THETA
        !
        THETA(II,1) = U
        THETA(II,2) = U_I/LAMBDA
        THETA(II,3) = U_II/LAMBDA**2._16
        THETA(II,4) = U_III/LAMBDA**3._16
        THETA(II,5) = U_IIII/LAMBDA**4._16
        !
        ! Calculate nondimensional disjoining pressures
        !
        PI_STAR = 1._16 / U**2._16
        PI_STAR_I = -2._16 * U_I / U**3._16 / LAMBDA
        !
        ! Calculate net evaporative mass flux and liquid pressure gradient
        !

        NONDTEMP = (DT_0 + KAPPA*(THETA(II,1)*THETA(II,3) + THETA(II,1)*PI_STAR)) / (DT_0 + KAPPA*THETA(II,1))
        M_EVP(II) = M_ID * (NONDTEMP - THETA(II,3) - PI_STAR)
        DPDX(II) = -SIGMA*H_0*THETA(II,4)/X_0**3._16 - PI_0*PI_STAR_I/X_0

    END DO

    !
    ! CONVERT TO ORIGINAL DIMENSIONAL THIN FILM EVAPORATION EQUATION
    !
    X = X_0*ETA
    H(:,1) = H_0*THETA(:,1)
    H(:,2) = H_0*THETA(:,2)/X_0
    H(:,3) = H_0*THETA(:,3)/X_0**2._16
    H(:,4) = H_0*THETA(:,4)/X_0**3._16
    H(:,5) = H_0*THETA(:,5)/X_0**4._16

    ! DISPLAY RESULTS
    PRINT LABEL,'X','H','H_I','H_II','H_III','H_IIII','M_EVP','DPDX'
```

```
        DO II=1,TERMS
            PRINT '(1X, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8)', &
            X(II), H(II,1), H(II,2), H(II,3), H(II,4), H(II,5), M_EVP(II), DPDX(II)
        END DO

END
```

C.2 Disjoining Pressure Cases E,G

```
PROGRAM HTLMTF_7AB
!************************************************************************************************************************
!
! File:     HTLMTF_7A-B.f90
! Language: FORTRAN 90
! Author:   jtipton2
! Summary:  Thin film model of K.P. Hallinan et al.,  "Evaporation from an Extended Meniscus for Nonisothermal
!           Interfacial Conditions,"  Journal of Thermophysics and Heat Transfer,  Vol. 8,  1994,  pp. 709-716.
!           + LIQUID METAL PROPERTIES
!           + GENERAL DGP DISPERSION FORCE MODELED WITH CUBIC SPLINE INTERPOLATION
!           + LECTRONIC COMPONENT OF THE DISPERSION FORCE
!           + WITH NONLINEARITIES SOLVED VIA THE LEVENBURG-MARQUARDT METHOD
!
! Redord of Revision:
!   Date        Programmer          Description of Change
!   ====        ==========          =====================
!    1/24/08    JBT                 Initial creation
!    1/05/09    JBT                 Changed T_V from 1156.09_16 to 1154.7_16
!                                   Moved all subroutines to modules to improve programming
!                                   Changed INDX to an integer to match the subroutine
!                                   Added LABEL parameter and used it to print result headers
!                                   Changed CS_N from 75 to 80
!                                   Changed LEFT_LIM_EXP from -8._16 to -9._16
!                                   Changed spline BC at X(1) from 5.30344944662574E20_16 to 6.492683078E25_16
!************************************************************************************************************************
    USE MATSOLV
    USE CUBICSPLINE
    USE CHEBYSHEV
    USE LMPROPERTIES
    IMPLICIT NONE
    REAL*16, PARAMETER :: PI = 3.14159265358979_16
    INTEGER, PARAMETER :: NN_MAX = 200, CS_N = 80, TERMS = 200
    INTEGER :: II, JJ, KK, NN, LOOPNUM, INDX(NN_MAX)
    CHARACTER (LEN=*), PARAMETER :: FORM1 = "(//␣1X␣A␣/␣1X,␣50('='))"
    CHARACTER (LEN=40), PARAMETER :: LABEL = "(//8(15X,A)/␣1X,␣50('='))"
    REAL*16 :: T_V, R_G, V_L, R, C_L, DT_0, ALPHA, BETA, GAMMA, LAMBDA, COEFF, B_ELEC, CHI, RHO, H_FG, MU, P_V, &
               SIGMA, K, MW, H_0, PI_0, X_0, M_ID, U_0, CA, HAMAKER, KAPPA
    REAL*16 :: T_BC(NN_MAX+4), T_I_BC(NN_MAX+4), T_II_BC(NN_MAX+4), T_III_BC(NN_MAX+4), &
               T(NN_MAX+4), T_I(NN_MAX+4), T_II(NN_MAX+4), T_III(NN_MAX+4), T_IIII(NN_MAX+4), &
               U, U_I, U_II, U_III, U_IIII, DU, DU_I, DU_II, DU_III, DU_IIII, PI_STAR, PI_STAR_I, PI_STAR_II
    REAL*16 :: C_OLD(NN_MAX), C_NEW(NN_MAX), DC(NN_MAX), JACOBIAN(NN_MAX,NN_MAX), F(NN_MAX), Q_OLD, Q_NEW, MARQ
    REAL*16 :: JACOBIAN_C_PLUS_F(NN_MAX), G(NN_MAX,NN_MAX), GRADQ(NN_MAX), G_PLUS_MARQ_I(NN_MAX,NN_MAX)
    REAL*16 :: ERROR, XI, KEY, IDENTITY(NN_MAX,NN_MAX)
    REAL*16 :: LEFT_LIM_EXP, RIGHT_LIM_EXP
    REAL*16 :: CS_XI(CS_N), CS_PI(CS_N), CS_PI2(CS_N), CS_X_I, CS_A, CS_B, CS_C, CS_D, CS_DIFF
    REAL*16 :: NONISO_DIFF, NONDTEMP
    REAL*16 :: ETA(TERMS), THETA(TERMS,5), X(TERMS), H(TERMS,5), M_EVP(TERMS), DPDX(TERMS)
    REAL*16 :: Q(NN_MAX), OMEGA(NN_MAX)

    ! Program Constants
    T_V = 1154.7_16                                      ! sodium temperature of vaporization (K)
    CALL LM_PROPS (T_V, RHO, K, H_FG, MU, MW, P_V, SIGMA, B_ELEC, HAMAKER)
    R_G = 8.314472_16                                    ! universal gas constant (N-m/K-mol)
```

```
V_L = MW/RHO                                        ! liquid molar volume (m^3/mol)
R = 200E-6_16                                       ! radius or width of pore (m)
C_L = 2._16                                         ! accomodation coefficient
DT_0 = 5E-4_16                                      ! wall/vapor temperature difference (K)
CHI = 0.0017060360_16                               ! electronic disjoining pressure boundary condition term


!----------------------------------------------------------------------------------------------------
! Get DLP Dispersion Force Data
!----------------------------------------------------------------------------------------------------

! Load discrete DLP Dispersion Force data from DISP_FORCE.f90
OPEN (UNIT = 12, FILE = 'DISPERSION_DATA', STATUS = 'OLD')
READ (12, '(100E22.15)') CS_PI
CLOSE (12)

LEFT_LIM_EXP = -9._16
RIGHT_LIM_EXP = LOG10(7.5_16)-7._16
DO II = 1,CS_N
    CS_XI(II) = 10._16**(LEFT_LIM_EXP + (II-1._16)*(RIGHT_LIM_EXP - LEFT_LIM_EXP)/(CS_N-1._16))
ENDDO

CALL SPLINE (CS_XI,CS_PI,CS_N,6.492683078E25_16,0._16,CS_PI2)
! Boundary condition at X(1) --> Calculated from Hamaker Approximation (F'' = -2A/pi/x^5)
!                       X(N) --> "Natural" spline condition (F'' = 0)

!----------------------------------------------------------------------------------------------------


! Nondimensional Variables and Scales:
CALL ADSORBED_THICKNESS_7AB &
(MW, H_FG, DT_0, V_L, T_V, B_ELEC, CHI, HAMAKER, CS_N, CS_XI, CS_PI, CS_PI2, 1E-6_16, 1E-11_16, H_0)
PI_0 = MW*H_FG*DT_0/V_L/T_V                         ! reference disjoining pressure (N/m^2)
X_0 = SQRT(SIGMA*H_0/PI_0)                          ! axial length scale (m)
M_ID = C_L * SQRT(MW/2._16/PI/R_G/T_V) * P_V*MW*H_FG*DT_0/(R_G*T_V**2._16)   ! ideal evaporative flux (kg/s/m^2)
U_0 = M_ID/RHO                                      ! liquid characteristic velocity (m/s)
CA = MU*U_0/SIGMA                                   ! capillary number
KAPPA = H_FG*M_ID*H_0/K                             ! ratio of evaporative interfacial resistance
                                                    ! to conductive resistance in the thin film

! Normalized problem boundary conditions
ALPHA = 1.04_16
BETA = 1E-4_16
GAMMA = X_0**2._16/R/H_0
LAMBDA = 5._16
COEFF = 3._16 * CA / (H_0*PI_0/SIGMA)**2._16

! Create Chebyshev polynomials of the first kind
! for the boundary conditions
CALL CHEBY_T (NN_MAX+4,0,-1._16,T_BC)
CALL CHEBY_T (NN_MAX+4,1,-1._16,T_I_BC)
CALL CHEBY_T (NN_MAX+4,2, 1._16,T_II_BC)
CALL CHEBY_T (NN_MAX+4,3,-1._16,T_III_BC)


```

```fortran
DO NN = 1,100,99
    ERROR = 1._16
    LOOPNUM = 0
    C_NEW = 0._16
    Q_NEW = 0._16
    Q = 0._16

    DO II = 1,NN
        DO JJ = 1,NN
            IF (II==JJ) THEN
                IDENTITY(II,JJ) = 1._16
            ELSE
                IDENTITY(II,JJ) = 0._16
            END IF
        END DO
    END DO

    PRINT FORM1,'Q = '

    DO WHILE (ERROR >= 1E-8_16 .AND. Q_NEW < 0.1E17_16)
        LOOPNUM = LOOPNUM + 1
        C_OLD = C_NEW
        Q_OLD = Q_NEW
        DO II = 1,NN
            !
            ! The "i" subscript refers to matrix rows which represent the
            ! functions evaluated at different values of the collocated domain
            ! variable, "XI".
            !
            XI = COS((2._16*II - 1._16)*PI/2._16/NN)
            !
            ! Create Chebyshev Polynomials of the First Kind
            ! and Their Derivatives
            !
            CALL CHEBY_T (NN_MAX+4,0,XI,T)
            CALL CHEBY_T (NN_MAX+4,1,XI,T_I)
            CALL CHEBY_T (NN_MAX+4,2,XI,T_II)
            CALL CHEBY_T (NN_MAX+4,3,XI,T_III)
            CALL CHEBY_T (NN_MAX+4,4,XI,T_IIII)
            !
            ! Establish the Approximate Analytical Series Solution
            ! and It's Derivatives
            !
            U = ALPHA + LAMBDA*BETA*(XI + 1._16) + (0.5_16*XI**2._16 + XI + 0.5_16)*GAMMA*LAMBDA**2._16 &
            + SUM(C_OLD(1:NN) * (T(5:NN+4) &
            - T_BC(5:NN+4) &
            - (XI + 1._16)*T_I_BC(5:NN+4) &
            - (0.5_16*XI**2._16 + XI + 0.5_16)*T_II_BC(5:NN+4) &
            + (-XI**3._16 + 3._16*XI**2._16 + 9._16*XI + 5._16)*T_III_BC(5:NN+4)/6._16))

            U_I = LAMBDA*BETA + (1._16 + XI)*GAMMA*LAMBDA**2._16 &
            + SUM(C_OLD(1:NN) * (T_I(5:NN+4) &
            - T_I_BC(5:NN+4) &
            - (XI + 1._16)*T_II_BC(5:NN+4) &
```

```
+ (-0.5_16*XI**2._16 + XI + 1.5_16)*T_III_BC(5:NN+4)))

U_II = GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T_II(5:NN+4) &
- T_II_BC(5:NN+4) &
+ (1._16 - XI)*T_III_BC(5:NN+4)))

U_III = SUM(C_OLD(1:NN) * (T_III(5:NN+4) - T_III_BC(5:NN+4)))

U_IIII = SUM(C_OLD(1:NN) * (T_IIII(5:NN+4)))

NONISO_DIFF = DT_0 + KAPPA*U

!
! Create the [F] matrix
!
IF (U*H_0 > 7.07493733732976559433421216887514375e-07_16) THEN
    PI_STAR = (B_ELEC*CHI/PI_0/H_0**2._16) / U**2._16
    PI_STAR_I = (-2._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA) * U_I / U**3._16
    PI_STAR_II = (6._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA**2._16) * U_I**2._16 / U**4._16 &
               - (2._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA**2._16) * U_II / U**3._16
    CS_A = 0._16
    CS_B = 0._16
    CS_C = 0._16
    CS_D = 0._16
ELSE
    CALL SPLINTS (CS_XI,CS_PI,CS_PI2,CS_N,U*H_0,CS_X_I,CS_A,CS_B,CS_C,CS_D)
    CS_DIFF = H_0*U - CS_X_I

    PI_STAR = CS_A*CS_DIFF**3._16/PI_0 + CS_B*CS_DIFF**2._16/PI_0 &
        + CS_C*CS_DIFF/PI_0 + CS_D/PI_0 &
        + (B_ELEC*CHI/PI_0/H_0**2._16) / U**2._16

    PI_STAR_I = 3._16*CS_A*H_0*CS_DIFF**2._16*U_I/LAMBDA/PI_0 &
        + 2._16*CS_B*H_0*CS_DIFF*U_I/LAMBDA/PI_0 &
        + CS_C*H_0*U_I/LAMBDA/PI_0 &
        - (2._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA) * U_I / U**3._16

    PI_STAR_II = 6._16*CS_A*H_0**2._16*CS_DIFF*U_I**2._16/LAMBDA**2._16/PI_0 &
     + 3._16*CS_A*H_0*CS_DIFF**2._16*U_II/LAMBDA**2._16/PI_0 &
     + 2._16*CS_B*H_0**2._16*U_I**2._16/LAMBDA**2._16/PI_0 &
     + 2._16*CS_B*CS_DIFF*H_0*U_II/LAMBDA**2._16/PI_0 + CS_C*H_0*U_II/LAMBDA**2._16/PI_0 &
     + (6._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA**2._16) * U_I**2._16 / U**4._16 &
     - (2._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA**2._16) * U_II / U**3._16
ENDIF

F(II) = -3._16*U**2._16*U_I*U_III/LAMBDA**4._16 &
        - U**3._16*U_IIII/LAMBDA**4._16 &
        - 3._16*U**2._16*U_I*PI_STAR_I/LAMBDA &
        - U**3._16*PI_STAR_II &
        + COEFF*U_II/LAMBDA**2._16 &
        + COEFF*PI_STAR &
        ! terms due to nonisothermal interface assumption
        - COEFF*DT_0/(DT_0+KAPPA*U) &
        - COEFF*KAPPA*U*U_II/LAMBDA**2._16/(DT_0+KAPPA*U) &
```

```
        - COEFF*KAPPA*U*PI_STAR/(DT_0+KAPPA*U)
!
! Create the [E] matrix
!
DO JJ = 1,NN

    DU = T(JJ+4) - T_BC(JJ+4) - (XI + 1._16)*T_I_BC(JJ+4) &
        - (0.5_16*XI**2._16 + XI + 0.5_16)*T_II_BC(JJ+4)&
        + (-XI**3._16 + 3._16*XI**2._16 + 9._16*XI + 5._16)*T_III_BC(JJ+4)/6._16
    DU_I = T_I(JJ+4) - T_I_BC(JJ+4) - (XI + 1._16)*T_II_BC(JJ+4) &
        + (-0.5_16*XI**2._16 + XI + 1.5_16)*T_III_BC(JJ+4)
    DU_II = T_II(JJ+4) - T_II_BC(JJ+4) + (1._16 - XI)*T_III_BC(JJ+4)
    DU_III = T_III(JJ+4) - T_III_BC(JJ+4)
    DU_IIII = T_IIII(JJ+4)

    JACOBIAN(II,JJ) = (6._16*U*U_I*U_III/LAMBDA**4._16) * DU &
                    + (3._16*U**2._16*U_III/LAMBDA**4._16) * DU_I &
                    + (3._16*U**2._16*U_I/LAMBDA**4._16) * DU_III &

                    + (3._16*U**2._16*U_IIII/LAMBDA**4._16) * DU &
                    + (U**3._16/LAMBDA**4._16) * DU_IIII &

                    - (COEFF/LAMBDA**2._16) * DU_II &

                    - (3._16*COEFF*CS_A*H_0*CS_DIFF**2._16/PI_0) * DU &
                    - (2._16*COEFF*CS_B*H_0*CS_DIFF/PI_0) * DU &
                    - (COEFF*CS_C*H_0/PI_0) * DU &

                    + (18._16*CS_A*H_0**2._16*CS_DIFF*U**2._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                    + (18._16*CS_A*H_0*CS_DIFF**2._16*U*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                    + (18._16*CS_A*H_0*CS_DIFF**2._16*U**2._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                    + (6._16*CS_B*H_0**2._16*U**2._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                    + (12._16*CS_B*H_0*CS_DIFF*U*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                    + (12._16*CS_B*H_0*CS_DIFF*U**2._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                    + (6._16*CS_C*H_0*U*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                    + (6._16*CS_C*H_0*U**2._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                    + (6._16*CS_A*H_0**3._16*U**3._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                    + (18._16*CS_A*H_0**2._16*CS_DIFF*U**2._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                    + (12._16*CS_A*H_0**2._16*CS_DIFF*U**3._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                    + (6._16*CS_A*H_0**2._16*CS_DIFF*U**3._16*U_II/LAMBDA**2._16/PI_0) * DU &
                    + (9._16*CS_A*H_0*CS_DIFF**2._16*U**2._16*U_II/LAMBDA**2._16/PI_0) * DU &
                    + (3._16*CS_A*H_0*CS_DIFF**2._16*U**3._16/LAMBDA**2._16/PI_0) * DU_II &

                    + (6._16*CS_B*H_0**2._16*U**2._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                    + (4._16*CS_B*H_0**2._16*U**3._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                    + (2._16*CS_B*H_0**2._16*U**3._16*U_II/LAMBDA**2._16/PI_0) * DU &
                    + (6._16*CS_B*H_0*CS_DIFF*U**2._16*U_II/LAMBDA**2._16/PI_0) * DU &
                    + (2._16*CS_B*H_0*CS_DIFF*U**3._16/LAMBDA**2._16/PI_0) * DU_II &

                    + (3._16*CS_C*H_0*U**2._16*U_II/LAMBDA**2._16/PI_0) * DU &
```

```
                        + (CS_C*H_0*U**3._16/LAMBDA**2._16/PI_0) * DU_II &

                        !
                        ! Electronic disjoining pressure terms
                        !
                        - (2._16*B_ELEC*CHI/H_0**2._16/PI_0/LAMBDA**2._16) * DU_II &

                        + (2._16*COEFF*B_ELEC*CHI/H_0**2._16/PI_0/U**3._16) * DU &

                        !
                        ! Nonisothermal interface terms
                        !
                        + (COEFF*DT_0*KAPPA/NONISO_DIFF**2._16) * DU &

                        + (COEFF*KAPPA*U_II/LAMBDA**2._16/NONISO_DIFF) * DU &
                        + (COEFF*KAPPA*U/LAMBDA**2._16/NONISO_DIFF) * DU_II &
                        + (COEFF*KAPPA**2._16*U*U_II/LAMBDA**2._16/NONISO_DIFF**2._16) * DU &

                        + (COEFF*KAPPA*CS_A*CS_DIFF**3._16/NONISO_DIFF/PI_0) * DU &
                        + (3._16*COEFF*KAPPA*CS_A*H_0*U*CS_DIFF**2._16/NONISO_DIFF/PI_0) * DU &
                        - (COEFF*KAPPA**2._16*CS_A*U*CS_DIFF**3._16/NONISO_DIFF**2._16/PI_0) * DU &

                        + (COEFF*KAPPA*CS_B*CS_DIFF**2._16/NONISO_DIFF/PI_0) * DU &
                        + (2._16*COEFF*KAPPA*CS_B*H_0*U*CS_DIFF/NONISO_DIFF/PI_0) * DU &
                        - (COEFF*KAPPA**2._16*CS_B*U*CS_DIFF**2._16/NONISO_DIFF**2._16/PI_0) * DU &

                        + (COEFF*KAPPA*CS_C*CS_DIFF/NONISO_DIFF/PI_0) * DU &
                        + (COEFF*KAPPA*CS_C*H_0*U/NONISO_DIFF/PI_0) * DU &
                        - (COEFF*KAPPA**2._16*CS_C*U*CS_DIFF/NONISO_DIFF**2._16/PI_0) * DU &

                        + (COEFF*KAPPA*CS_D/NONISO_DIFF/PI_0) * DU &
                        - (COEFF*KAPPA**2._16*CS_D*U/NONISO_DIFF**2._16/PI_0) * DU &

                        - (COEFF*KAPPA*B_ELEC/H_0**2._16/PI_0/U**2._16/NONISO_DIFF) * DU &
                        - (COEFF*KAPPA**2._16*B_ELEC/H_0**2._16/PI_0/U/NONISO_DIFF**2._16) * DU
    END DO
END DO

!**LEVENBERG-MARQUARDT METHOD**********************************************
!
! Formulation described by:
! Henley and Rosen,  "Material and Energy Balance Computations,"
! John Wiley & Sons, 1969, pp. 171-173, 192-204.
!
!
! Solution procedure described by:
! "Numerical Recipes in Fortran:  The Art of Scientific Computing"
! 2nd Edition, pp. 679.
!
! (1)  compute Q
! (2)  pick MARQ = 0.001
! (3)  solve linear system (G + MARQ*I) DC = -GRADQ
! (4)  compute new Q(C) = SUM (JACOBIAN DC + F)^2
! (4a) if Q_NEW >= Q_OLD then MARQ = MARQ*10 | goto (3)
! (4b) if Q_NEW <  Q_OLD then MARQ = MARQ/10 | goto (3)
```

```
            ! (5)  if Q_NEW < Q_OLD AND Q_NEW < tol then STOP
            !
            !***************************************************************************
            IF (LOOPNUM == 1) THEN
                Q_NEW = SQRT(DOT_PRODUCT(F,F))
                Q_OLD = 1._16
                MARQ = 0.0001_16
            ELSE
                JACOBIAN_C_PLUS_F = MATMUL(JACOBIAN,C_OLD) + F
                Q_NEW = SQRT(DOT_PRODUCT(JACOBIAN_C_PLUS_F,JACOBIAN_C_PLUS_F))
            END IF

            IF (Q_NEW .GE. Q_OLD) THEN
                MARQ = MARQ * 10._16
            ELSE
                MARQ = MARQ / 10._16
            END IF

            G = 2._16 * MATMUL(TRANSPOSE(JACOBIAN),JACOBIAN)
            GRADQ = 2._16 * MATMUL(TRANSPOSE(JACOBIAN),F)
            G_PLUS_MARQ_I = G + MARQ*IDENTITY
            CALL LUDCMP (G_PLUS_MARQ_I,NN,NN_MAX,INDX,KEY)
            DC = GRADQ
            CALL LUBKSB (G_PLUS_MARQ_I,NN,NN_MAX,INDX,DC)
            C_NEW = DC + C_OLD

            ERROR = ABS(Q_NEW-Q_OLD)/Q_OLD
            Q(LOOPNUM) = Q_NEW
            PRINT '(1G32.16)', Q(LOOPNUM)

        END DO

        !
        ! SPATIAL CONVERGENCE ACCURACY
        ! (Integrate approximate analytical solution over the domain space)
        !
        OMEGA(NN) = 2._16*ALPHA + 2._16*LAMBDA*BETA + 4._16*LAMBDA**2._16*GAMMA/3._16

        DO JJ = 1,NN/2
            KK = 2._16*JJ
            OMEGA(NN) = OMEGA(NN) + C_NEW(KK) * ( -2._16*T_BC(KK) - 2._16*T_I_BC(KK) - 5._16*T_II_BC(KK)/3._16 &
                                                 + 2._16*T_III_BC(KK) - 2._16/(KK+1._16)/(KK-1._16) )
        END DO

    END DO

    PRINT FORM1,'OMEGA = '
    PRINT '(1G32.16)', OMEGA(1:NN-1)

        !
        ! CONVERT TO ORIGINAL NONDIMENSIONALIZED THIN FILM EVAPORATION EQUATION
        !
        DO II = 1,TERMS
            XI = COS((2._16*II - 1._16)*PI/2._16/TERMS)
            ETA(II) = LAMBDA*(1._16 + XI)
```

```
!
! Create Chebyshev Polynomials of the First Kind
! and Their Derivatives
!
CALL CHEBY_T (NN_MAX+4,0,XI,T)
CALL CHEBY_T (NN_MAX+4,1,XI,T_I)
CALL CHEBY_T (NN_MAX+4,2,XI,T_II)
CALL CHEBY_T (NN_MAX+4,3,XI,T_III)
CALL CHEBY_T (NN_MAX+4,4,XI,T_IIII)
!
! Establish the Approximate Analytical Series Solution
! and It's Derivatives
!
U = ALPHA + LAMBDA*BETA*(XI + 1._16) + (0.5_16*XI**2._16 + XI + 0.5_16)*GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T(5:NN+4) &
- T_BC(5:NN+4) &
- (XI + 1._16)*T_I_BC(5:NN+4) &
- (0.5_16*XI**2._16 + XI + 0.5_16)*T_II_BC(5:NN+4) &
+ (-XI**3._16 + 3._16*XI**2._16 + 9._16*XI + 5._16)*T_III_BC(5:NN+4)/6._16))

U_I = LAMBDA*BETA + (1._16 + XI)*GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T_I(5:NN+4) &
- T_I_BC(5:NN+4) &
- (XI + 1._16)*T_II_BC(5:NN+4) &
+ (-0.5_16*XI**2._16 + XI + 1.5_16)*T_III_BC(5:NN+4)))

U_II = GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T_II(5:NN+4) &
- T_II_BC(5:NN+4) &
+ (1._16 - XI)*T_III_BC(5:NN+4)))

U_III = SUM(C_OLD(1:NN) * (T_III(5:NN+4) - T_III_BC(5:NN+4)))

U_IIII = SUM(C_OLD(1:NN) * (T_IIII(5:NN+4)))
!
! Convert to from THETA_HAT to THETA
!
THETA(II,1) = U
THETA(II,2) = U_I/LAMBDA
THETA(II,3) = U_II/LAMBDA**2._16
THETA(II,4) = U_III/LAMBDA**3._16
THETA(II,5) = U_IIII/LAMBDA**4._16
!
! Calculate nondimensional disjoining pressures
!
IF (U*H_0 > 7.07493733732976559433421216887514375e-07_16) THEN
    PI_STAR = (B_ELEC*CHI/PI_0/H_0**2._16) / U**2._16
    PI_STAR_I = (-2._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA) * U_I / U**3._16
    PI_STAR_II = (6._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA**2._16) * U_I**2._16 / U**4._16 &
                   - (2._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA**2._16) * U_II / U**3._16
    CS_A = 0._16
    CS_B = 0._16
    CS_C = 0._16
    CS_D = 0._16
ELSE
```

```
                CALL SPLINTS (CS_XI,CS_PI,CS_PI2,CS_N,U*H_0,CS_X_I,CS_A,CS_B,CS_C,CS_D)
                CS_DIFF = H_0*U - CS_X_I

                PI_STAR = CS_A*CS_DIFF**3._16/PI_0 + CS_B*CS_DIFF**2._16/PI_0 + CS_C*CS_DIFF/PI_0 + CS_D/PI_0 &
                            + (B_ELEC*CHI/PI_0/H_0**2._16) / U**2._16

                PI_STAR_I = 3._16*CS_A*H_0*CS_DIFF**2._16*U_I/LAMBDA/PI_0 + 2._16*CS_B*H_0*CS_DIFF*U_I/LAMBDA/PI_0 &
                              + CS_C*H_0*U_I/LAMBDA/PI_0 &
                              - (2._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA) * U_I / U**3._16

                PI_STAR_II = 6._16*CS_A*H_0**2._16*CS_DIFF*U_I**2._16/LAMBDA**2._16/PI_0 &
                     + 3._16*CS_A*H_0*CS_DIFF**2._16*U_II/LAMBDA**2._16/PI_0 &
                     + 2._16*CS_B*H_0**2._16*U_I**2._16/LAMBDA**2._16/PI_0 &
                     + 2._16*CS_B*CS_DIFF*H_0*U_II/LAMBDA**2._16/PI_0 + CS_C*H_0*U_II/LAMBDA**2._16/PI_0 &
                     + (6._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA**2._16) * U_I**2._16 / U**4._16 &
                     - (2._16*B_ELEC*CHI/PI_0/H_0**2._16/LAMBDA**2._16) * U_II / U**3._16
            ENDIF
            !
            ! Calculate net evaporative mass flux and liquid pressure gradient
            !
            NONDTEMP = (DT_0 + KAPPA*(THETA(II,1)*THETA(II,3) + THETA(II,1)*PI_STAR)) / (DT_0 + KAPPA*THETA(II,1))
            M_EVP(II) = M_ID * (NONDTEMP - THETA(II,3) - PI_STAR)
            DPDX(II) = -SIGMA*H_0*THETA(II,4)/X_0**3._16 - PI_0*PI_STAR_I/X_0
        END DO

        !
        ! CONVERT TO ORIGINAL DIMENSIONAL THIN FILM EVAPORATION EQUATION
        !
        X = X_0*ETA
        H(:,1) = H_0*THETA(:,1)
        H(:,2) = H_0*THETA(:,2)/X_0
        H(:,3) = H_0*THETA(:,3)/X_0**2._16
        H(:,4) = H_0*THETA(:,4)/X_0**3._16
        H(:,5) = H_0*THETA(:,5)/X_0**4._16

        ! DISPLAY RESULTS
        PRINT LABEL,'X','H','H_I','H_II','H_III','H_IIII','M_EVP','DPDX'
        DO II=1,TERMS
            PRINT '(1X, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8)', &
            X(II), H(II,1), H(II,2), H(II,3), H(II,4), H(II,5), M_EVP(II), DPDX(II)
        END DO

END
```

C.3 Disjoining Pressure Case F

```fortran
PROGRAM HTLMTF_7A2
!***********************************************************************************************************************
!
! File:     HTLMTF_7A2.f90
! Language: FORTRAN 90
! Author:   jtipton2
! Summary:  Thin film model of K.P. Hallinan et al.,  "Evaporation from an Extended Meniscus for Nonisothermal
!           Interfacial Conditions,"  Journal of Thermophysics and Heat Transfer,  Vol. 8,  1994,  pp. 709-716.
!           + LIQUID METAL PROPERTIES
!           + GENERAL DGP DISPERSION FORCE MODELED WITH CUBIC SPLINE INTERPOLATION
!           + WITH NONLINEARITIES SOLVED VIA THE LEVENBURG-MARQUARDT METHOD
!
! Redord of Revision:
!   Date        Programmer          Description of Change
!   ====        ==========          ====================
!    1/25/08    JBT                 Initial creation
!   12/10/08    JBT                 Changed T_V from 1156.09_16 to 1154.7_16
!   12/22/08    JBT                 Changed CS_N from 75 to 80
!                                   Changed LEFT_LIM_EXP from -8._16 to -9._16
!                                   Changed spline BC at X(1) from 5.30344944662574E20_16 to 6.492683078E25_16
!                                   Removed "Hammaker" variable from program since it is not used
!                                   Moved all subroutines to modules to improve programming
!    1/02/09    JBT                 Added Electronic Dispersion Force Constant calculation to LM_Props subroutine
!    1/05/09    JBT                 Renamed ADSORBED_THICKNESS subroutine to ADSORBED_THICKNESS_7A2
!***********************************************************************************************************************
    USE MATSOLV
    USE CUBICSPLINE
    USE CHEBYSHEV
    USE LMPROPERTIES
    IMPLICIT NONE
    INTEGER :: II, JJ, KK, NN, LOOPNUM
    REAL*16, PARAMETER :: PI = 3.14159265358979_16
    INTEGER, PARAMETER :: NN_MAX = 200, CS_N = 80, TERMS = 200
    CHARACTER (LEN=40), PARAMETER :: FORM1 = "(//␣1X␣A␣/␣1X,␣50('='))"
    CHARACTER (LEN=40), PARAMETER :: LABEL = "(//8(15X,A)/␣1X,␣50('='))"
    REAL*16 :: T_V, R_G, V_L, R, C_L, DT_0, ALPHA, BETA, GAMMA, LAMBDA, COEFF, RHO, H_FG, MU, P_V, SIGMA, K, MW, &
               H_0, PI_0, X_0, M_ID, U_0, CA, KAPPA
    REAL*16 :: T_BC(NN_MAX+4), T_I_BC(NN_MAX+4), T_II_BC(NN_MAX+4), T_III_BC(NN_MAX+4), &
               T(NN_MAX+4), T_I(NN_MAX+4), T_II(NN_MAX+4), T_III(NN_MAX+4), T_IIII(NN_MAX+4), &
               U, U_I, U_II, U_III, U_IIII, DU, DU_I, DU_II, DU_III, DU_IIII, PI_STAR, PI_STAR_I, PI_STAR_II
    REAL*16 :: C_OLD(NN_MAX), C_NEW(NN_MAX), DC(NN_MAX), JACOBIAN(NN_MAX,NN_MAX), F(NN_MAX), Q_OLD, Q_NEW, MARQ
    REAL*16 :: JACOBIAN_C_PLUS_F(NN_MAX), G(NN_MAX,NN_MAX), GRADQ(NN_MAX), G_PLUS_MARQ_I(NN_MAX,NN_MAX)
    REAL*16 :: IDENTITY(NN_MAX,NN_MAX)
    INTEGER :: INDX(NN_MAX)
    REAL*16 :: ERROR, XI, KEY
    REAL*16 :: LEFT_LIM_EXP, RIGHT_LIM_EXP, CS_XI(CS_N), CS_PI(CS_N), CS_PI2(CS_N)
    REAL*16 :: CS_X_I, CS_A, CS_B, CS_C, CS_D, CS_DIFF
    REAL*16 :: ETA(TERMS), THETA(TERMS,5), X(TERMS), H(TERMS,5), M_EVP(TERMS), DPDX(TERMS), PI_STAR_STORE(TERMS)
    REAL*16 :: Q(NN_MAX), OMEGA(NN_MAX)
    REAL*16 :: NONISO_DIFF, NONDTEMP

    ! Program Constants
    T_V = 1154.7!1156.09_16                                    ! sodium temperature of vaporization (K)
```

```
CALL LM_PROPS (T_V, RHO, K, H_FG, MU, MW, P_V, SIGMA)
R_G = 8.314472_16                                   ! universal gas constant (N-m/K-mol)
V_L = MW/RHO                                        ! liquid molar volume (m^3/mol)
R = 200E-6_16                                        ! radius or width of pore (m)
C_L = 2._16                                         ! accomodation coefficient
DT_0 = 5E-4_16                                      ! wall/vapor temperature difference (K)


!----------------------------------------------------------------------------------------------------
! Get DLP Dispersion Force Data
!----------------------------------------------------------------------------------------------------

! Load discrete DLP Dispersion Force data from DISPERSION_DATA
OPEN (UNIT = 12, FILE = 'DISPERSION_DATA', STATUS = 'OLD')
READ (12, '(100E22.15)') CS_PI
CLOSE (12)

LEFT_LIM_EXP = -9._16
RIGHT_LIM_EXP = LOG10(7.5_16)-7._16
DO II = 1,CS_N
    CS_XI(II) = 10._16**(LEFT_LIM_EXP + (II-1._16)*(RIGHT_LIM_EXP - LEFT_LIM_EXP)/(CS_N-1._16))
ENDDO

CALL SPLINE (CS_XI,CS_PI,CS_N,6.492683078E25_16,0._16,CS_PI2)
! Boundary condition at X(1) --> Calculated from Hamaker Approximation (F'' = -2A/pi/x^5)
!                       X(N) --> "Natural" spline condition (F'' = 0)

!----------------------------------------------------------------------------------------------------


! Nondimensional Variables and Scales:
CALL ADSORBED_THICKNESS_7A2 (MW, H_FG, DT_0, V_L, T_V, CS_N, CS_XI, CS_PI, CS_PI2, 1.32E-8_16, 1.42E-8_16, H_0)
PI_0 = MW*H_FG*DT_0/V_L/T_V                         ! reference disjoining pressure (N/m^2)
X_0 = SQRT(SIGMA*H_0/PI_0)                          ! axial length scale (m)
M_ID = C_L * SQRT(MW/2._16/PI/R_G/T_V) * P_V*MW*H_FG*DT_0/(R_G*T_V*(T_V+DT_0))   ! ideal evaporative flux (kg/s/m^2)
U_0 = M_ID/RHO                                      ! liquid characteristic velocity (m/s)
CA = MU*U_0/SIGMA                                   ! capillary number
KAPPA = H_FG*M_ID*H_0/K                       ! ratio of evaporative interfacial resistance to conductive resistance

! Normalized problem boundary conditions
ALPHA = 1.04_16
BETA = 1E-4_16
GAMMA = X_0**2._16/R/H_0
LAMBDA = 10._16
COEFF = 3._16 * CA / (H_0*PI_0/SIGMA)**2._16

! Create Chebyshev polynomials of the first kind
! for the boundary conditions
CALL CHEBY_T (NN_MAX+4,0,-1._16,T_BC)
CALL CHEBY_T (NN_MAX+4,1,-1._16,T_I_BC)
CALL CHEBY_T (NN_MAX+4,2, 1._16,T_II_BC)
CALL CHEBY_T (NN_MAX+4,3,-1._16,T_III_BC)


```

```
DO NN = 1,100,99
    ERROR = 1._16
    LOOPNUM = 0
    C_NEW = 0._16
    Q_NEW = 0._16
    Q = 0._16

    DO II = 1,NN
        DO JJ = 1,NN
            IF (II==JJ) THEN
                IDENTITY(II,JJ) = 1._16
            ELSE
                IDENTITY(II,JJ) = 0._16
            END IF
        END DO
    END DO

    PRINT FORM1,'Q = '

    DO WHILE (ERROR >= 1E-8_16 .AND. Q_NEW < 0.1E17_16)
        LOOPNUM = LOOPNUM + 1
        C_OLD = C_NEW
        Q_OLD = Q_NEW
        DO II = 1,NN
            !
            ! The "i" subscript refers to matrix rows which represent the
            ! functions evaluated at different values of the collocated domain
            ! variable, "XI".
            !
            XI = COS((2._16*II - 1._16)*PI/2._16/NN)
            !
            ! Create Chebyshev Polynomials of the First Kind
            ! and Their Derivatives
            !
            CALL CHEBY_T (NN_MAX+4,0,XI,T)
            CALL CHEBY_T (NN_MAX+4,1,XI,T_I)
            CALL CHEBY_T (NN_MAX+4,2,XI,T_II)
            CALL CHEBY_T (NN_MAX+4,3,XI,T_III)
            CALL CHEBY_T (NN_MAX+4,4,XI,T_IIII)
            !
            ! Establish the Approximate Analytical Series Solution
            ! and It's Derivatives
            !
            U = ALPHA + LAMBDA*BETA*(XI + 1._16) + (0.5_16*XI**2._16 + XI + 0.5_16)*GAMMA*LAMBDA**2._16 &
            + SUM(C_OLD(1:NN) * (T(5:NN+4) &
            - T_BC(5:NN+4) &
            - (XI + 1._16)*T_I_BC(5:NN+4) &
            - (0.5_16*XI**2._16 + XI + 0.5_16)*T_II_BC(5:NN+4) &
            + (-XI**3._16 + 3._16*XI**2._16 + 9._16*XI + 5._16)*T_III_BC(5:NN+4)/6._16))

            U_I = LAMBDA*BETA + (1._16 + XI)*GAMMA*LAMBDA**2._16 &
            + SUM(C_OLD(1:NN) * (T_I(5:NN+4) &
            - T_I_BC(5:NN+4) &
            - (XI + 1._16)*T_II_BC(5:NN+4) &
            + (-0.5_16*XI**2._16 + XI + 1.5_16)*T_III_BC(5:NN+4)))
```

```fortran
U_II = GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T_II(5:NN+4) &
- T_II_BC(5:NN+4) &
+ (1._16 - XI)*T_III_BC(5:NN+4)))

U_III = SUM(C_OLD(1:NN) * (T_III(5:NN+4) - T_III_BC(5:NN+4)))

U_IIII = SUM(C_OLD(1:NN) * (T_IIII(5:NN+4)))

NONISO_DIFF = DT_0 + KAPPA*U

!
! Create the [F] matrix
!
IF (U*H_0 > 7.07493733732976559433421216887514375e-07_16) THEN
    PI_STAR = 0._16
    PI_STAR_I = 0._16
    PI_STAR_II = 0._16
    CS_A = 0._16
    CS_B = 0._16
    CS_C = 0._16
    CS_D = 0._16
ELSE
    CALL SPLINTS (CS_XI,CS_PI,CS_PI2,CS_N,U*H_0,CS_X_I,CS_A,CS_B,CS_C,CS_D)
    CS_DIFF = H_0*U - CS_X_I

    PI_STAR = CS_A*CS_DIFF**3._16/PI_0 + CS_B*CS_DIFF**2._16/PI_0 + CS_C*CS_DIFF/PI_0 + CS_D/PI_0

    PI_STAR_I = 3._16*CS_A*H_0*CS_DIFF**2._16*U_I/LAMBDA/PI_0 + 2._16*CS_B*H_0*CS_DIFF*U_I/LAMBDA/PI_0 &
    + CS_C*H_0*U_I/LAMBDA/PI_0

    PI_STAR_II = 6._16*CS_A*H_0**2._16*CS_DIFF*U_I**2._16/LAMBDA**2._16/PI_0 &
    + 3._16*CS_A*H_0*CS_DIFF**2._16*U_II/LAMBDA**2._16/PI_0 &
    + 2._16*CS_B*H_0**2._16*U_I**2._16/LAMBDA**2._16/PI_0 &
    + 2._16*CS_B*CS_DIFF*H_0*U_II/LAMBDA**2._16/PI_0 + CS_C*H_0*U_II/LAMBDA**2._16/PI_0
ENDIF

F(II) = -3._16*U**2._16*U_I*U_III/LAMBDA**4._16 &
        - U**3._16*U_IIII/LAMBDA**4._16 &
        - 3._16*U**2._16*U_I*PI_STAR_I/LAMBDA &
        - U**3._16*PI_STAR_II &
        + COEFF*U_II/LAMBDA**2._16 &
        + COEFF*PI_STAR &
        ! terms due to nonisothermal interface assumption
        - COEFF*DT_0/(DT_0+KAPPA*U) &
        - COEFF*KAPPA*U*U_II/LAMBDA**2._16/(DT_0+KAPPA*U) &
        - COEFF*KAPPA*U*PI_STAR/(DT_0+KAPPA*U)

!
! Create the [E] matrix
!
DO JJ = 1,NN

    DU = T(JJ+4) - T_BC(JJ+4) - (XI + 1._16)*T_I_BC(JJ+4) &
```

```
        - (0.5_16*XI**2._16 + XI + 0.5_16)*T_II_BC(JJ+4)&
        + (-XI**3._16 + 3._16*XI**2._16 + 9._16*XI + 5._16)*T_III_BC(JJ+4)/6._16
DU_I = T_I(JJ+4) - T_I_BC(JJ+4) - (XI + 1._16)*T_II_BC(JJ+4) + (-0.5_16*XI**2._16 + XI + 1.5_16) &
        *T_III_BC(JJ+4)
DU_II = T_II(JJ+4) - T_II_BC(JJ+4) + (1._16 - XI)*T_III_BC(JJ+4)
DU_III = T_III(JJ+4) - T_III_BC(JJ+4)
DU_IIII = T_IIII(JJ+4)

JACOBIAN(II,JJ) = (6._16*U*U_I*U_III/LAMBDA**4._16) * DU &
                + (3._16*U**2._16*U_III/LAMBDA**4._16) * DU_I &
                + (3._16*U**2._16*U_I/LAMBDA**4._16) * DU_III &

                + (3._16*U**2._16*U_IIII/LAMBDA**4._16) * DU &
                + (U**3._16/LAMBDA**4._16) * DU_IIII &

                - (COEFF/LAMBDA**2._16) * DU_II &

                - (3._16*COEFF*CS_A*H_0*CS_DIFF**2._16/PI_0) * DU &
                - (2._16*COEFF*CS_B*H_0*CS_DIFF/PI_0) * DU &
                - (COEFF*CS_C*H_0/PI_0) * DU &

                + (18._16*CS_A*H_0**2._16*CS_DIFF*U**2._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                + (18._16*CS_A*H_0*CS_DIFF**2._16*U*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                + (18._16*CS_A*H_0*CS_DIFF**2._16*U**2._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                + (6._16*CS_B*H_0**2._16*U**2._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                + (12._16*CS_B*H_0*CS_DIFF*U*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                + (12._16*CS_B*H_0*CS_DIFF*U**2._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                + (6._16*CS_C*H_0*U*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                + (6._16*CS_C*H_0*U**2._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                + (6._16*CS_A*H_0**3._16*U**3._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                + (18._16*CS_A*H_0**2._16*CS_DIFF*U**2._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                + (12._16*CS_A*H_0**2._16*CS_DIFF*U**3._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                + (6._16*CS_A*H_0**2._16*CS_DIFF*U**3._16*U_II/LAMBDA**2._16/PI_0) * DU &
                + (9._16*CS_A*H_0*CS_DIFF**2._16*U**2._16*U_II/LAMBDA**2._16/PI_0) * DU &
                + (3._16*CS_A*H_0*CS_DIFF**2._16*U**3._16/LAMBDA**2._16/PI_0) * DU_II &

                + (6._16*CS_B*H_0**2._16*U**2._16*U_I**2._16/LAMBDA**2._16/PI_0) * DU &
                + (4._16*CS_B*H_0**2._16*U**3._16*U_I/LAMBDA**2._16/PI_0) * DU_I &

                + (2._16*CS_B*H_0**2._16*U**3._16*U_II/LAMBDA**2._16/PI_0) * DU &
                + (6._16*CS_B*H_0*CS_DIFF*U**2._16*U_II/LAMBDA**2._16/PI_0) * DU &
                + (2._16*CS_B*H_0*CS_DIFF*U**3._16/LAMBDA**2._16/PI_0) * DU_II &

                + (3._16*CS_C*H_0*U**2._16*U_II/LAMBDA**2._16/PI_0) * DU &
                + (CS_C*H_0*U**3._16/LAMBDA**2._16/PI_0) * DU_II &

                !
                ! Nonisothermal interface terms
                !
                + (COEFF*DT_0*KAPPA/NONISO_DIFF**2._16) * DU &

```

```
                        + (COEFF*KAPPA*U_II/LAMBDA**2._16/NONISO_DIFF) * DU &
                        + (COEFF*KAPPA*U/LAMBDA**2._16/NONISO_DIFF) * DU_II &
                        + (COEFF*KAPPA**2._16*U*U_II/LAMBDA**2._16/NONISO_DIFF**2._16) * DU &

                        + (COEFF*KAPPA*CS_A*CS_DIFF**3._16/NONISO_DIFF/PI_0) * DU &
                        + (3._16*COEFF*KAPPA*CS_A*H_0*U*CS_DIFF**2._16/NONISO_DIFF/PI_0) * DU &
                        - (COEFF*KAPPA**2._16*CS_A*U*CS_DIFF**3._16/NONISO_DIFF**2._16/PI_0) * DU &

                        + (COEFF*KAPPA*CS_B*CS_DIFF**2._16/NONISO_DIFF/PI_0) * DU &
                        + (2._16*COEFF*KAPPA*CS_B*H_0*U*CS_DIFF/NONISO_DIFF/PI_0) * DU &
                        - (COEFF*KAPPA**2._16*CS_B*U*CS_DIFF**2._16/NONISO_DIFF**2._16/PI_0) * DU &

                        + (COEFF*KAPPA*CS_C*CS_DIFF/NONISO_DIFF/PI_0) * DU &
                        + (COEFF*KAPPA*CS_C*H_0*U/NONISO_DIFF/PI_0) * DU &
                        - (COEFF*KAPPA**2._16*CS_C*U*CS_DIFF/NONISO_DIFF**2._16/PI_0) * DU &

                        + (COEFF*KAPPA*CS_D/NONISO_DIFF/PI_0) * DU &
                        - (COEFF*KAPPA**2._16*CS_D*U/NONISO_DIFF**2._16/PI_0) * DU
    END DO
END DO

!**LEVENBERG-MARQUARDT METHOD**********************************************
!
! Formulation described by:
! Henley and Rosen,  "Material and Energy Balance Computations,"
! John Wiley & Sons, 1969, pp. 171-173, 192-204.
!
!
! Solution procedure described by:
! "Numerical Recipes in Fortran:  The Art of Scientific Computing"
! 2nd Edition, pp. 679.
!
! (1)  compute Q
! (2)  pick MARQ = 0.001
! (3)  solve linear system (G + MARQ*I) DC = -GRADQ
! (4)  compute new Q(C) = SUM (JACOBIAN DC + F)^2
! (4a) if Q_NEW >= Q_OLD then MARQ = MARQ*10 | goto (3)
! (4b) if Q_NEW <  Q_OLD then MARQ = MARQ/10 | goto (3)
! (5)  if Q_NEW < Q_OLD AND Q_NEW < tol then STOP
!
!**************************************************************************
IF (LOOPNUM == 1) THEN
    Q_NEW = SQRT(DOT_PRODUCT(F,F))
    Q_OLD = 1._16
    MARQ = 0.0001_16
ELSE
    JACOBIAN_C_PLUS_F = MATMUL(JACOBIAN,C_OLD) + F
    Q_NEW = SQRT(DOT_PRODUCT(JACOBIAN_C_PLUS_F,JACOBIAN_C_PLUS_F))
END IF

IF (Q_NEW .GE. Q_OLD) THEN
    MARQ = MARQ * 10._16
ELSE
    MARQ = MARQ / 10._16
END IF
```

```
        G = 2._16 * MATMUL(TRANSPOSE(JACOBIAN),JACOBIAN)
        GRADQ = 2._16 * MATMUL(TRANSPOSE(JACOBIAN),F)
        G_PLUS_MARQ_I = G + MARQ*IDENTITY
        CALL LUDCMP (G_PLUS_MARQ_I,NN,NN_MAX,INDX,KEY)
        DC = GRADQ
        CALL LUBKSB (G_PLUS_MARQ_I,NN,NN_MAX,INDX,DC)
        C_NEW = DC + C_OLD

        ERROR = ABS(Q_NEW-Q_OLD)/Q_OLD
        Q(LOOPNUM) = Q_NEW
        PRINT '(1G32.16)', Q(LOOPNUM)

    END DO

    !
    ! SPATIAL CONVERGENCE ACCURACY
    ! (Integrate approximate analytical solution over the domain space)
    !
    OMEGA(NN) = 2._16*ALPHA + 2._16*LAMBDA*BETA + 4._16*LAMBDA**2._16*GAMMA/3._16

    DO JJ = 1,NN/2
        KK = 2._16*JJ
        OMEGA(NN) = OMEGA(NN) + C_NEW(KK) * ( -2._16*T_BC(KK) - 2._16*T_I_BC(KK) - 5._16*T_II_BC(KK)/3._16 &
                                             + 2._16*T_III_BC(KK) - 2._16/(KK+1._16)/(KK-1._16) )
    END DO

END DO

    PRINT FORM1,'OMEGA = '
    PRINT '(1G32.16)', OMEGA(1:NN-1)

    !
    ! CONVERT TO ORIGINAL NONDIMENSIONALIZED THIN FILM EVAPORATION EQUATION
    !
    DO II = 1,TERMS
        XI = COS((2._16*II - 1._16)*PI/2._16/TERMS)
        ETA(II) = LAMBDA*(1._16 + XI)
        !
        ! Create Chebyshev Polynomials of the First Kind
        ! and Their Derivatives
        !
        CALL CHEBY_T (NN_MAX+4,0,XI,T)
        CALL CHEBY_T (NN_MAX+4,1,XI,T_I)
        CALL CHEBY_T (NN_MAX+4,2,XI,T_II)
        CALL CHEBY_T (NN_MAX+4,3,XI,T_III)
        CALL CHEBY_T (NN_MAX+4,4,XI,T_IIII)
        !
        ! Establish the Approximate Analytical Series Solution
        ! and It's Derivatives
        !
        U = ALPHA + LAMBDA*BETA*(XI + 1._16) + (0.5_16*XI**2._16 + XI + 0.5_16)*GAMMA*LAMBDA**2._16 &
        + SUM(C_OLD(1:NN) * (T(5:NN+4) &
        - T_BC(5:NN+4) &
        - (XI + 1._16)*T_I_BC(5:NN+4) &
```

```
- (0.5_16*XI**2._16 + XI + 0.5_16)*T_II_BC(5:NN+4) &
+ (-XI**3._16 + 3._16*XI**2._16 + 9._16*XI + 5._16)*T_III_BC(5:NN+4)/6._16))

U_I = LAMBDA*BETA + (1._16 + XI)*GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T_I(5:NN+4) &
- T_I_BC(5:NN+4) &
- (XI + 1._16)*T_II_BC(5:NN+4) &
+ (-0.5_16*XI**2._16 + XI + 1.5_16)*T_III_BC(5:NN+4)))

U_II = GAMMA*LAMBDA**2._16 &
+ SUM(C_OLD(1:NN) * (T_II(5:NN+4) &
- T_II_BC(5:NN+4) &
+ (1._16 - XI)*T_III_BC(5:NN+4)))

U_III = SUM(C_OLD(1:NN) * (T_III(5:NN+4) - T_III_BC(5:NN+4)))

U_IIII = SUM(C_OLD(1:NN) * (T_IIII(5:NN+4)))
!
! Convert to from THETA_HAT to THETA
!
THETA(II,1) = U
THETA(II,2) = U_I/LAMBDA
THETA(II,3) = U_II/LAMBDA**2._16
THETA(II,4) = U_III/LAMBDA**3._16
THETA(II,5) = U_IIII/LAMBDA**4._16
!
! Calculate nondimensional disjoining pressures
!
IF (U*H_0 > 7.07493733732976559433421216887514375e-07_16) THEN
    PI_STAR = 0._16
    PI_STAR_I = 0._16
    PI_STAR_II = 0._16
    CS_A = 0._16
    CS_B = 0._16
    CS_C = 0._16
    CS_D = 0._16
ELSE
    CALL SPLINTS (CS_XI,CS_PI,CS_PI2,CS_N,U*H_0,CS_X_I,CS_A,CS_B,CS_C,CS_D)
    CS_DIFF = H_0*U - CS_X_I

    PI_STAR = CS_A*CS_DIFF**3._16/PI_0 + CS_B*CS_DIFF**2._16/PI_0 + CS_C*CS_DIFF/PI_0 + CS_D/PI_0

    PI_STAR_I = 3._16*CS_A*H_0*CS_DIFF**2._16*U_I/LAMBDA/PI_0 + 2._16*CS_B*H_0*CS_DIFF*U_I/LAMBDA/PI_0 &
        + CS_C*H_0*U_I/LAMBDA/PI_0

    PI_STAR_II = 6._16*CS_A*H_0**2._16*CS_DIFF*U_I**2._16/LAMBDA**2._16/PI_0 &
        + 3._16*CS_A*H_0*CS_DIFF**2._16*U_II/LAMBDA**2._16/PI_0 &
        + 2._16*CS_B*H_0**2._16*U_I**2._16/LAMBDA**2._16/PI_0 &
        + 2._16*CS_B*CS_DIFF*H_0*U_II/LAMBDA**2._16/PI_0 + CS_C*H_0*U_II/LAMBDA**2._16/PI_0
ENDIF
PI_STAR_STORE(II) = PI_STAR
!
! Calculate net evaporative mass flux and liquid pressure gradient
!
NONDTEMP = (DT_0 + KAPPA*(THETA(II,1)*THETA(II,3) + THETA(II,1)*PI_STAR)) / (DT_0 + KAPPA*THETA(II,1))
```

```
            M_EVP(II) = M_ID * (NONDTEMP - THETA(II,3) - PI_STAR)
            DPDX(II) = -SIGMA*H_0*THETA(II,4)/X_0**3._16 - PI_0*PI_STAR_I/X_0
        END DO

        !
        ! CONVERT TO ORIGINAL DIMENSIONAL THIN FILM EVAPORATION EQUATION
        !
        X = X_0*ETA
        H(:,1) = H_0*THETA(:,1)
        H(:,2) = H_0*THETA(:,2)/X_0
        H(:,3) = H_0*THETA(:,3)/X_0**2._16
        H(:,4) = H_0*THETA(:,4)/X_0**3._16
        H(:,5) = H_0*THETA(:,5)/X_0**4._16

        ! DISPLAY RESULTS
        PRINT LABEL,'X','H','H_I','H_II','H_III','H_IIII','M_EVP','DPDX'
        DO II=1,TERMS
            PRINT '(1X, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8, G16.8)', &
            X(II), H(II,1), H(II,2), H(II,3), H(II,4), H(II,5), M_EVP(II), DPDX(II), PI_STAR_STORE(II)
        END DO

END
```

Appendix D

Thin Film Solution Modules

The following four programs are FORTRAN modules utilized by the thin film solution programs given in Appendix C. The first module, Appendix D.1, evaluates the Chebyshev polynomials of the first kind up to the fourth derivative using recursive functions. The second module, Appendix D.2, uses cubic spline interpolation to curve-fit the dispersion force curve calculated in Appendix B. The third module, Appendix D.3, calculates the pertinent physical property values for liquid sodium metal at a specified temperature using the references presented in Table 3.2. The module also uses the bisection root finding algorithm to calculate the thickness of the adsorbed film for all of the cases presented in Table 6.1. Finally, the fourth module, Appendix D.4, contains the linear algebra solvers used during the numerical solution procedure. This module performs forward and back substitution along with Crout's Method with partial pivoting. Where mentioned in the program comments, subroutines have been used with permission from *Numerical Recipes in FORTRAN* [110].

D.1 Chebyshev Polynomials

```
!
! File:   CHEBYSHEV.f90
! Author: jtipton2
!
! Created on January 5, 2009, 5:41 PM
!

MODULE CHEBYSHEV
    CONTAINS
    !**CHEBY_T*****************************************************************
```

```
    !
    ! Evaluates the Chebyshev polynomials T(N)(X) of the first kind
    ! up to the 4th derivative
    !
    !  Parameters:
    !
    !    Input, integer MM, the highest polynomial to compute.
    !
    !    Input, integer DD, the derivative requested.
    !
    !    Input, real X, the point at which the polynomials are to be computed.
    !
    !    Output, real CX(1:MM), the values of the MM Chebyshev polynomials.
    !
    !*************************************************************************
    SUBROUTINE CHEBY_T (MM, DD, X, CX)
        IMPLICIT NONE
        INTEGER, INTENT(IN) :: MM, DD
        REAL*16, INTENT(IN) :: X
        REAL*16, INTENT(OUT) :: CX(MM)
        REAL*16 :: CX_TEMP(MM,5)
        INTEGER :: KK

        IF (MM <= 0) THEN
            RETURN
        END IF

        CX_TEMP(1,:) = (/1._16, 0._16, 0._16, 0._16, 0._16/)

        IF (MM == 1) THEN
            RETURN
        END IF

        CX_TEMP(2,:) = (/X, 1._16, 0._16, 0._16, 0._16/)

        DO KK = 3,MM
            CX_TEMP(KK,1) =                         2._16*X*CX_TEMP(KK-1,1) - CX_TEMP(KK-2,1)
            CX_TEMP(KK,2) = 2._16*CX_TEMP(KK-1,1) + 2._16*X*CX_TEMP(KK-1,2) - CX_TEMP(KK-2,2)
            CX_TEMP(KK,3) = 4._16*CX_TEMP(KK-1,2) + 2._16*X*CX_TEMP(KK-1,3) - CX_TEMP(KK-2,3)
            CX_TEMP(KK,4) = 6._16*CX_TEMP(KK-1,3) + 2._16*X*CX_TEMP(KK-1,4) - CX_TEMP(KK-2,4)
            CX_TEMP(KK,5) = 8._16*CX_TEMP(KK-1,4) + 2._16*X*CX_TEMP(KK-1,5) - CX_TEMP(KK-2,5)
        END DO

        CX = CX_TEMP(:,DD+1)
    END SUBROUTINE CHEBY_T
END MODULE CHEBYSHEV
```

D.2 Cubic Spline Interpolation

```
!
! File:   CUBICSPLINE.f90
! Author: jtipton2
!
! Created on January 5, 2009, 5:41 PM
!

MODULE CUBICSPLINE
    CONTAINS
    !--SPLINE--------------------------------------------------------------------------------------------
    !
    ! Curve-fit data using cubic spline interpolation.
    ! For description, see http://www.physics.utah.edu/~detar/phycs6720/handouts/cubic_spline/cubic_spline/node1.html
    ! Algorithm from "Numerical Recipes in Fortran: The Art of Scientific Computing" 2nd Ed.
    !
    !  Parameters:
    !
    !    Input, real X(N), domain values of the function.
    !
    !    Input, real Y(N), tabulated function values corresponding to X(N).
    !
    !    Input, integer N, the size of the tabulated function values.
    !
    !    Input, real YPP1, second derivative of the interpolating function at X(1).
    !
    !    Input, real YPP2, second derivative of the interpolating function at X(N).
    !
    !    Output, real Y2(N), second derivatives of the interpolating function at tabulated points X(N).
    !
    !----------------------------------------------------------------------------------------------------
    SUBROUTINE SPLINE(X,Y,N,YPP1,YPPN,Y2)
        IMPLICIT NONE
        INTEGER, INTENT (IN) :: N
        REAL*16, INTENT (IN) :: YPP1, YPPN, X(N), Y(N)
        REAL*16, INTENT (OUT) :: Y2(N)
        INTEGER :: NMAX     ! The largest anticipated value of N
        PARAMETER (NMAX=500)
        INTEGER :: J
        REAL*16 :: A(N), B(N), C(N), R(N)

        ! Setup the initial boundary condition (i.e. known second derivatives)
        B(1) = 1._16
        C(1) = 0._16
        R(1) = YPP1

        ! Construct the tridiagonal matrix (out of 3 vectors) and vector of known data
        DO J = 2,N-1
            A(J) = (X(J) - X(J-1))/6._16
            B(J) = (X(J+1) - X(J-1))/3._16
            C(J) = (X(J+1) - X(J))/6._16
            R(J) = (Y(J+1)-Y(J))/(X(J+1)-X(J)) - (Y(J)-Y(J-1))/(X(J)-X(J-1))
        END DO
```

```
    ! Setup the final boundary condition
    A(N) = 0._16
    B(N) = 1._16
    R(N) = YPPN

    CALL TRIDAG (A,B,C,R,Y2,N)

    RETURN

END SUBROUTINE SPLINE


!--TRIDAG-------------------------------------------------------------------------------------------
!
! Tridiagonal equation solution routine.
! Algorithm from "Numerical Recipes in Fortran: The Art of Scientific Computing" 2nd Ed.
!
!  Parameters:
!
!    Input, real A(N), Lower diagonal of the coefficient matrix.
!
!    Input, real B(N), Middle diagonal of the coefficient matrix.
!
!    Input, real C(N), Upper diagonal of the coefficient matrix.
!
!    Input, real R(N), Forcing data array.
!
!    Input, integer N, Array size.
!
!    Output, real U(N), Matrix solution.
!
!---------------------------------------------------------------------------------------------------
SUBROUTINE TRIDAG(A,B,C,R,U,N)
    IMPLICIT NONE
    INTEGER, INTENT (IN) :: N
    REAL*16, INTENT (IN) :: A(N), B(N), C(N), R(N)
    REAL*16, INTENT (OUT) :: U(N)
    INTEGER :: NMAX             ! The largest anticipated value of N
    PARAMETER (NMAX=500)
    INTEGER :: J
    REAL*16 :: BET, GAM(NMAX)   ! One vector of workspace is needed

    IF (B(1) .EQ. 0._16) PAUSE 'TRIDAG: REWRITE EQUATION'   ! If this happens this you should rewrite your equations
                                                            ! as a set of order N-1, with u(2) trivially eliminated.
    BET = B(1)
    U(1) = R(1)/BET
    DO J = 2,N                  ! Decomposition and forward substitution
        GAM(J) = C(J-1)/BET
        BET=B(J)-A(J)*GAM(J)
        IF (BET .EQ. 0._16) PAUSE 'TRIDAG FAILED'
        U(J) = (R(J)-A(J)*U(J-1))/BET
    END DO

    DO J=N-1,1,-1               ! Backsubstitution
        U(J) = U(J) - GAM(J+1)*U(J+1)
```

```
    END DO

    RETURN

END SUBROUTINE TRIDAG


!--SPLINT-------------------------------------------------------------------------------------------
!
! Subroutine to return cubic-spline interpolated value of "y".
! Algorithm from "Numerical Recipes in Fortran: The Art of Scientific Computing" 2nd Ed.
!
!  Parameters:
!
!    Input, real XA(N), domain values of the function.
!
!    Input, real YA(N), tabulated function values corresponding to X(N).
!
!    Input, real Y2A(N), second derivatives of the interpolating function at tabulated points X(N).
!
!    Input, integer N, the size of the tabulated function values.
!
!    Input, real X, desired point in the domain.
!
!    Output, real Y, cubic-spline interpolated value of the function at X.
!
!---------------------------------------------------------------------------------------------------
SUBROUTINE SPLINT (XA,YA,Y2A,N,X,Y)
    IMPLICIT NONE
    INTEGER, INTENT (IN) :: N
    REAL*16, INTENT (IN) :: X,XA(N),Y2A(N),YA(N)
    REAL*16, INTENT (OUT) :: Y
    INTEGER :: K, KHI, KLO
    REAL*16 :: A,B,C,D,H

    KLO = 1
    KHI = N
    ! Find the right place in the table by means of bisection.
    DO WHILE (KHI-KLO .GT. 1)
        K = (KHI+KLO)/2
        IF (XA(K) .GT. X) THEN
            KHI = K
        ELSE
            KLO = K
        ENDIF
    ENDDO   ! KLO and KHI now bracket the input value of x.

    H = XA(KHI) - XA(KLO)

    IF (H .EQ. 0._16) PAUSE 'BAD XA INPUT IN SPLINT'    ! The XA's must be distinct

    ! Now evaluate the cubic spline.
    A = (Y2A(KHI) - Y2A(KLO)) / 6._16 / H
    B = Y2A(KLO) / 2._16
    C = (YA(KHI)-YA(KLO))/H - H*Y2A(KLO)/3._16 - H*Y2A(KHI)/6._16
```

```
    D = YA(KLO)

    Y = A*(X-XA(KLO))**3._16 + B*(X-XA(KLO))**2._16 + C*(X-XA(KLO)) + D

    RETURN
END SUBROUTINE SPLINT


!--SPLINTS-------------------------------------------------------------------------------------------
!
! Subroutine to return cubic-spline interpolated coefficients.
! Algorithm from "Numerical Recipes in Fortran: The Art of Scientific Computing" 2nd Ed.
!
!  Parameters:
!
!    Input, real XA(N), domain values of the function.
!
!    Input, real YA(N), tabulated function values corresponding to X(N).
!
!    Input, real Y2A(N), second derivatives of the interpolating function at tabulated points X(N).
!
!    Input, integer N, the size of the tabulated function values.
!
!    Input, real X, desired point in the domain.
!
!    Output, real A, first cubic spline coefficient
!    Output, real B, second cubic spline coefficient
!    Output, real C, third cubic spline coefficient
!    Output, real D, fourth cubic spline coefficient
!
!---------------------------------------------------------------------------------------------------
SUBROUTINE SPLINTS (XA,YA,Y2A,N,X,XI,A,B,C,D)
    IMPLICIT NONE
    INTEGER, INTENT (IN) :: N
    REAL*16, INTENT (IN) :: X,XA(N),Y2A(N),YA(N)
    REAL*16, INTENT (OUT) :: XI, A, B, C, D
    INTEGER :: K, KHI, KLO
    REAL*16 :: H

    KLO = 1
    KHI = N
    ! Find the right place in the table by means of bisection.
    DO WHILE (KHI-KLO .GT. 1)
        K = (KHI+KLO)/2
        IF (XA(K) .GT. X) THEN
            KHI = K
        ELSE
            KLO = K
        ENDIF
    ENDDO   ! KLO and KHI now bracket the input value of x.

    H = XA(KHI) - XA(KLO)

    IF (H .EQ. 0._16) PAUSE 'BAD XA INPUT IN SPLINT'    ! The XA's must be distinct

```

```
        ! Now evaluate the cubic spline coefficients
        XI = XA(KLO)
        A = (Y2A(KHI) - Y2A(KLO)) / 6._16 / H
        B = Y2A(KLO) / 2._16
        C = (YA(KHI)-YA(KLO))/H - H*Y2A(KLO)/3._16 - H*Y2A(KHI)/6._16
        D = YA(KLO)

        RETURN
    END SUBROUTINE SPLINTS
END MODULE CUBICSPLINE
```

D.3 Liquid Metal Thermophysical Properties

```
!
! File:   LMPROPERTIES.f90
! Author: jtipton2
!
! Created on January 5, 2009, 5:42 PM
!

MODULE LMPROPERTIES
    CONTAINS
    !**ADSORBED_THICKNESS_7A2*****************************************************************
    !
    ! Calculates the thickness of the adsorbed film
    ! using the bisection root finding algorithm
    !
    !
    !  Parameters:
    !
    !    Input,  real MW,  molecular weight
    !    Input,  real H_FG,  latent heat of vaporization
    !    Input,  real DT_0,  liquid overheat
    !    Input,  real V_L,  liquid volume
    !    Input,  real T_V,  vapor temperature
    !    Input,  integer N,  dispersion force cubic spline array size
    !    Input,  real X(1:N), cubic spline array film thicknesses
    !    Input,  real F(1:N), cubic spline dispersion force values
    !    Input,  real F2(1:N), second derivative of cubic spline dispersion force values
    !    Input,  real GUESS_L,  left bound of root
    !    Input,  real GUESS_R,  right bound of root
    !
    !    Output, real ANSWER,  root
    !
    !*****************************************************************************************
    SUBROUTINE ADSORBED_THICKNESS_7A2 (MW, H_FG, DT_0, V_L, T_V, N, X, F, F2, GUESS_L, GUESS_R, ANSWER)
        USE CUBICSPLINE
        IMPLICIT NONE
        INTEGER :: I, J
        INTEGER, INTENT(IN) :: N
        REAL*16, INTENT(IN) :: MW, H_FG, DT_0, V_L, T_V, X(N), F(N), F2(N), GUESS_L, GUESS_R
        REAL*16, INTENT(OUT) :: ANSWER

        REAL*16 :: FX1, FX3, X1, X2, X3

        X1 = GUESS_L
        X2 = GUESS_R

        DO WHILE (ABS(X1-X2)/X1 >= 1E-8_16)
            X3 = (X1 + X2)/2._16
            CALL SPLINT (X,F,F2,N,X1,FX1)
            FX1 = MW*H_FG*DT_0/V_L/T_V - FX1
            CALL SPLINT (X,F,F2,N,X3,FX3)
            FX3 = MW*H_FG*DT_0/V_L/T_V - FX3
            IF (FX3*FX1 < 0._16) THEN
                X2 = X3
```

```
        ELSE
            X1 = X3
        END IF
    END DO

    ANSWER = X3

    END SUBROUTINE ADSORBED_THICKNESS_7A2


!**ADSORBED_THICKNESS_7AB***************************************************************
!
! Calculates the thickness of the adsorbed film
! using the bisection root finding algorithm
!
!
!  Parameters:
!
!    Input,  real MW,  molecular weight
!    Input,  real H_FG,  latent heat of vaporization
!    Input,  real DT_0,  liquid overheat
!    Input,  real V_L,  liquid volume
!    Input,  real T_V,  vapor temperature
!    Input,  integer N,  dispersion force cubic spline array size
!    Input,  real X(1:N), cubic spline array film thicknesses
!    Input,  real F(1:N), cubic spline dispersion force values
!    Input,  real F2(1:N), second derivative of cubic spline dispersion force values
!    Input,  real GUESS_L,  left bound of root
!    Input,  real GUESS_R,  right bound of root
!
!    Output, real ANSWER,  root
!    Output, real RATIO,  ratio of electronic to dipsersion disjoining pressures
!
!**************************************************************************************
SUBROUTINE ADSORBED_THICKNESS_7AB (MW, H_FG, DT_0, V_L, T_V, B, CHI, A, N, X, F, F2, GUESS_L, GUESS_R, ANSWER, RATIO)
    USE CUBICSPLINE
    IMPLICIT NONE
    INTEGER :: I, J
    INTEGER, INTENT(IN) :: N
    REAL*16, INTENT(IN) :: MW, H_FG, DT_0, V_L, T_V, B, CHI, A, X(N), F(N), F2(N), GUESS_L, GUESS_R
    REAL*16, INTENT(OUT) :: ANSWER
    REAL*16, OPTIONAL, INTENT(OUT) :: RATIO

    REAL*16 :: FX1, FX3, X1, X2, X3

    X1 = GUESS_L
    X2 = GUESS_R

    DO WHILE (ABS(X1-X2)/X1 >= 1E-8_16)
        X3 = (X1 + X2)/2._16

        IF (X1 > 6.89712477526423354387130105695148843e-07_16) THEN
            ! NOTE:  The last point of the dispersion curve at 750nm acts a little 'funny' due to the natural
            ! spline BC in order to get find a root here, we must assume the second to last data point of CS_XI
            ! (690nm) is the last point in the dispersion force curve.
```

```
            FX1 = 0._16
        ELSEIF (X1 < 1E-8_16) THEN
            FX1 = A/X1**3._16
        ELSE
            CALL SPLINT (X,F,F2,N,X1,FX1)
        END IF
        FX1 = MW*H_FG*DT_0/V_L/T_V - B*CHI/X1**2._16 - FX1

        IF (X3 > 6.89712477526423354387130105695148843e-07_16) THEN
            FX3 = 0._16
        ELSEIF (X1 < 1E-8_16) THEN
            FX3 = A/X3**3._16
        ELSE
            CALL SPLINT (X,F,F2,N,X3,FX3)
        END IF
        FX3 = MW*H_FG*DT_0/V_L/T_V - B*CHI/X3**2._16 - FX3

        IF (FX3*FX1 < 0._16) THEN
            X2 = X3
        ELSE
            X1 = X3
        END IF
    END DO

    ANSWER = X3

    IF(PRESENT(RATIO))THEN
        CALL SPLINT (X,F,F2,N,X3,FX3)
        RATIO = B*CHI/X3**2._16/FX3
    ENDIF

END SUBROUTINE ADSORBED_THICKNESS_7AB


!**LM_PROPS*****************************************************
!
! Calculates pertinent physical property values for liquid
! sodium metal at the specified temperature.
!
! J.K. Fink and L. Leibowitz. A consistent assessment of the thermophysical
! properties of sodium. High Temp. Mater. Sci., 35:65  103 , 1996.
!
! Parameters:
!
!    Input,  real TEMP,  temperature
!
!    Output, real RHO,  density
!    Output, real K,  thermal conductivity
!    Output, real H_FG,  latent heat of evaporation
!    Output, real MU,  dynamic viscosity
!    Output, real MW,  molecular weight
!    Output, real P_V,  vapor pressure
!    Output, real SIGMA,  density
!    Output, real B,  electronic disjoining pressure constant
!    Output, real A,  Hamaker constant
```

```
    !
    !****************************************************************
    SUBROUTINE LM_PROPS (TEMP, RHO, K, H_FG, MU, MW, P_V, SIGMA, B, A)
        !
        ! Load Physical Constants
        !
        IMPLICIT NONE
        REAL*16, PARAMETER :: PI = 3.14159265358979_16
        REAL*16, PARAMETER :: NA = 6.0221415E+23_16         !Avogadro's Number
        REAL*16, PARAMETER :: QE = 1.60217646E-19_16        !Electron Charge (Coulomb = A s)
        REAL*16, PARAMETER :: EO = 8.85418782E-12_16        !Permittivity of Free Space (A**2 s**4 / m**3 / kg)
        REAL*16, PARAMETER :: ME = 9.10938188E-31_16        !Electron Mass (kg)
        REAL*16, PARAMETER :: BOLTZMANN = 1.3806503E-23_16  !Boltzmann Constant (m**2 kg / s**2 / K)
        REAL*16, PARAMETER :: PLANCK = 6.626068E-34_16      !Planck's Constant (m**2 kg / s)
        !
        ! Solid Type 304 Stainless Steel Properties
        !
        REAL*16, PARAMETER :: RHO_SS304 = 8000._16          ! kg/m**3
        REAL*16, PARAMETER :: M_SS304 = 0.05481_16          ! kg/mol
        REAL*16, PARAMETER :: NV_SS304 = 1.79_16            ! # Valence Electrons / molecule
        !
        ! Liquid Sodium Properties
        ! "Thermodynamic and Transport Properties of Sodium Liquid and Vapor" ANL/RE-95/2
        !
        REAL*16, INTENT(IN) :: TEMP
        REAL*16, INTENT(OUT) :: RHO, K, MU, SIGMA, P_V, H_FG, MW
        REAL*16, OPTIONAL, INTENT(OUT) :: A, B
        REAL*16 :: CP, NV
        CP = 1000._16 * &
            (1.6582_16 - 8.4790E-4_16 * TEMP + 4.4541E-7_16 * TEMP**2._16 - 2992.6_16 / TEMP**2._16)        ! J/kg/K
        RHO = 219._16 + 275.32_16 * (1._16 - TEMP/2503.7_16) + 511.58_16 * SQRT(1._16 - TEMP/2503.7_16)     ! kg/m**3
        K = 124.67_16 - 0.11381_16 * TEMP + 5.5226E-5_16 * TEMP**2._16 - 1.1842E-8_16 * TEMP**3._16         ! W/m/K
        MU = EXP( -6.4406_16 - 0.3958_16 * LOG(TEMP) + 556.835_16/TEMP )                                    ! Pa-s
        SIGMA = 240.5_16 * (1._16 - TEMP/2503.7_16)**1.126_16 / 1000._16                                    ! N/m
        P_V = 1E6_16 * EXP( 11.9463_16 - 12633.73_16/TEMP - 0.4672_16*LOG(TEMP) )                           ! Pa
        H_FG = 393370._16 * (1._16 - TEMP/2503.7_16) + 4398600._16 * (1._16 - TEMP/2503.7_16)**0.29302_16   ! J/kg
        MW = 0.02299_16                                                                                     ! kg/mol
        NV = 1._16                                                                    ! # Valence Electrons / molecule
        !
        ! Electronic Dispersion Force Constant
        !
        IF(PRESENT(A))THEN
            A = -1.015143464E-19_16                                            ! Hamaker constant calculated from MAPLE (J)
        ENDIF
        IF(PRESENT(B))THEN
            B = (1._16/8._16/PI**2._16) * (PLANCK**2._16/ME) * (NA*RHO*NV/MW)
        ENDIF
    END SUBROUTINE LM_PROPS
END MODULE LMPROPERTIES
```

D.4 Matrix Algebra

```
!
! File:   MATSOLV.f90
! Author: jtipton2
!
! Created on January 5, 2009, 5:40 PM
!

MODULE MATSOLV
    CONTAINS
    !--LUDCMP---------------------------------------------------------------------------------
    !
    ! Crout's Method with Partial Pivoting
    ! "Numerical Recipes in Fortran:  The Art of Scientific Computing"
    ! 2nd Edition
    !
    !-----------------------------------------------------------------------------------------
        SUBROUTINE LUDCMP(a,n,np,indx,d)
            IMPLICIT NONE
            INTEGER, INTENT(IN) :: n, np
            INTEGER, INTENT(OUT) :: indx(np)
            INTEGER, PARAMETER :: NMAX = 500
            REAL*16, INTENT(INOUT) :: a(np,np)
            REAL*16, INTENT(OUT) :: d
            INTEGER :: i, imax, j, k
            REAL*16 ::  aamax, dum, sum, vv(NMAX), TINY

            TINY = 1E-40_16

            d=1._16
            DO i=1,n
                aamax=0._16
                DO j=1,n
                    IF (abs(a(i,j)) > aamax) aamax=abs(a(i,j))
                ENDDO
                IF (aamax == 0._16) pause 'singular matrix in ludcmp'
                vv(i)=1._16/aamax
            ENDDO
            DO j=1,n
                DO i=1,j-1
                    sum=a(i,j)
                    DO k=1,i-1
                        sum=sum-a(i,k)*a(k,j)
                    ENDDO
                    a(i,j)=sum
                ENDDO
                aamax=0._16
                DO i=j,n
                    sum=a(i,j)
                    DO k=1,j-1
                        sum=sum-a(i,k)*a(k,j)
                    ENDDO
                    a(i,j)=sum
                    dum=vv(i)*abs(sum)
```

```
                IF (dum >= aamax) THEN
                    imax=i
                    aamax=dum
                ENDIF
            ENDDO
            IF (j <> imax) THEN
                DO k=1,n
                    dum=a(imax,k)
                    a(imax,k)=a(j,k)
                    a(j,k)=dum
                ENDDO
                d=-d
                vv(imax)=vv(j)
            ENDIF
            indx(j)=imax
            IF(a(j,j) == 0._16)a(j,j)=TINY  !In case matrix is singular
            IF(j.ne.n) THEN
                dum=1._16/a(j,j)
                DO i=j+1,n
                    a(i,j)=a(i,j)*dum
                ENDDO
            ENDIF
        ENDDO
        RETURN
    END SUBROUTINE LUDCMP

!--LUBKSB---------------------------------------------------------------------------------
!
! Forward substitution and Back Substitution for Use with LU Decomposition
! "Numerical Recipes in Fortran:  The Art of Scientific Computing"
! 2nd Edition
!
!-----------------------------------------------------------------------------------------
    SUBROUTINE LUBKSB (a,n,np,indx,b)
        IMPLICIT NONE
        INTEGER, INTENT(IN) :: n, np, indx(np)
        REAL*16, INTENT(IN) :: a(np,np)
        REAL*16, INTENT(INOUT) :: b(np)
        INTEGER :: i, ii, j, ll
        REAL*16 :: sum

        ii = 0
        DO i = 1, n
            ll = indx(i)
            sum = b(ll)
            b(ll) = b(i)
            IF (ii <> 0) THEN
                DO j = ii, i-1
                    sum = sum - a(i,j)*b(j)
                ENDDO
            ELSE IF (sum <> 0._16) THEN
                ii = i
            ENDIF
            b(i) = sum
        ENDDO
```

```
            DO i = n,1,-1
                sum = b(i)
                DO j = i+1,n
                    sum = sum - a(i,j)*b(j)
                ENDDO
                b(i) = sum/a(i,i)
            ENDDO

            RETURN
        END SUBROUTINE LUBKSB
END MODULE MATSOLV
```

Vita

Joseph Brown Tipton, Jr. was born in Nashville, TN on April 9, 1979. In 1997, he graduated in the top 10% of his class at Brentwood High School in Brentwood, Tennessee. He received a Bachelor of Science degree in Aerospace Engineering with *Magna Cum Laude* honors from the University of Tennessee in August 2002. After a year-long sabbatical in Central America, he returned to the University of Tennessee to pursue graduate studies and received a Master of Science degree in Mechanical Engineering in May 2006. Upon graduation, he will be pursuing a career in engineering education as an Assistant Professor of Mechanical Engineering at the University of Evansville in Evansville, IN. Up to date information on Joseph can be located on the internet at `http://www.JosephTipton.com`.