5-2006

Theoretical Study of Noble Gas Bubble Behavior in Mercury

Bo Lu
University of Tennessee - Knoxville

To the Graduate Council:

I am submitting herewith a thesis written by Bo Lu entitled "Theoretical Study of Noble Gas Bubble Behavior in Mercury." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Nuclear Engineering.

Arthur E. Ruggles, Major Professor

We have read this thesis and recommend its acceptance:

Laurence F. Miller, Belle R. Upadhyaya

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 thesis written by Bo Lu entitled "Theoretical Study of Noble Gas Bubble Behavior in Mercury." I have examined the final electronic copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Nuclear Engineering.

Arthur E. Ruggles
Major Professor

We have read this thesis and
recommend its acceptance:

Laurence F. Miller

Belle R. Upadhyaya

Acceptance for the Council:
Anne Mayhew
Vice Chancellor and
Dean of Graduate Studies

(Original signatures are on file with official student records)

Theoretical Study of Noble Gas Bubble Behavior in Mercury

A Thesis

Presented for the

Master of Science Degree

The University of Tennessee, Knoxville

Bo Lu

May 2006

Acknowledgements

First, I would like to express my sincere gratitude to my research advisor, Dr. Arthur E. Ruggles, for his invaluable guidance, encouragement and financial support throughout the research project and this thesis work. I would also like to thank the High Power Target project sponsors at Oak Ridge National Laboratory for providing me the opportunities to work with them and attend numerous wonderful workshops and field trips. Special thanks go to Bernard W. Riemer, Thomas W. Burgess, and Mark W. Wendel. Former fellow graduate student, Nathan Delauder, must be mentioned for his help and discussion in the research and the preparation of this thesis.

Sincere thanks also go to my committee members, Dr. Laurence. F. Miller and Dr. Belle R. Upadhyaya, for their comments and help on my research and thesis. I would also like to show my appreciation to all other professors and staff at the University of Tennessee, Nuclear Engineering Department, especially to Dr. Harold L. Dodds for offering me the opportunity to study and work here.

Personally, I want to thank my parents and my sister who are always supporting my goal for pursuing higher education. Finally, thanks go to my wife Na Wang, for being so kind and considerate to me whenever things went well or not. I appreciate sincerely her understanding and support of my ambitions.

Abstract

Spallation Neutron Source (SNS) uses heavy liquid metal (mercury) as the target material for high power proton beam bombardment to produce neutrons for scientific research. Though the liquid target is not subject to material degradation due to radiation damage, the stainless steel pressure boundary confining the liquid metal flow is damaged by radiation and cavitation erosion induced by the thermal shock waves caused by the deposition of the incoming high-power proton beam. This puts a limit on the lifetime of the target holder.

To mitigate the cavitation-induced erosion damage to the target holder, it is aimed to introduce microbubbles to the target mercury with expected nominal size of 30μm diameter and volume fraction of 0.5%, which can substantially lower the pressure amplitude resulting from the proton beam deposition due to the added compressibility.

The noble gas bubble behavior in mercury is studied in this thesis. The acoustics of the two-phase mixture under the perturbation due to beam deposition, specifically acoustic streaming**,** is simulated in a bubbly two-phase flow for the first time in the literature. The numerical simulation shows the magnitude of obtained streaming velocity is much smaller than the pumped mercury flow in the target and will not cause distortion to flow patterns and heat transfer in the target.

Single bubble dynamics, which includes noble gas solubility evaluation in mercury and the bubble radius evolution under the effect of mass diffusion across the bubble wall, is also simulated. Two different profiles of bubble size distribution are studied. The solubility evaluation provides a theoretical basis for the inert gas solubility measurement experiments. The mass diffusion induced bubble behavior simulation based on the solubility results indicates that xenon bubbles creates a more viable and stable bubble population in mercury than helium bubbles, which means xenon is a possible better candidate to add compressibility to pure mercury in the SNS target.

Table of Contents

1 Introduction 1
1.1 Background 1
1.2 Purpose and Organization of the Thesis 3
2 Acoustic Streaming in Mercury-Helium Two-Phase Flow 5
2.1 Literature Review 5
2.2 Physical Models 6
2.3 Problem Description 9
2.4 Numerical Methods 10
2.5 Results and Discussion 12
2.5.1 Case one with V_m=0.2m/s 12
2.5.2 Case two with V_m=0.4m/s 19
2.5.3 Conclusions 25
3 Theoretical Investigation of Inert Gas Solubility in Mercury 26
3.1 Introduction 26
3.2 Model for Inert Gas Solubility Evaluation in Mercury 26
3.3 Numerical Results of Solubility Evaluation 28
3.4 Henry's Law Constant 30
4 Mass Diffusion Induced Inert Gas Bubble Behavior in Mercury 39
4.1 Introduction 39
4.2 Diffusion Coefficient of Helium and Xenon in Mercury 39
4.3 Model for Predicting Bubble Growth Rate 40
4.4 Numerical Analysis of One-Size Group of Bubbles 42
4.5 Numerical Analysis of Two-Size Group of Bubbles 42
5 Conclusions and Suggestions for Future Work 48
List of References 49
Appendices 54
Appendix A 55
1-D Acoustic Streaming Simulation Code in FORTRAN 90 for Helium-Mercury

Two-Phase Bubbly Flow 55
Appendix B 65
MATLAB Script for Inert Gas Solubility Evaluation in Mercury 65
Appendix C 68
MATLAB Script for One-Group Bubble Growth Rate Simulation 68
C.1 Master Script 68
C.2 Fourth Order Runge-Kutta Function 68
C.3 Bubble Growth Rate Function 69
C.4 Gas Concentration Changing Rate 70
Appendix D 71
MATLAB Script for Two-Group Bubble Growth Rate Simulation 71
D.1 Master Script 71
D.2 Fourth-Order Runge-Kutta Function 72
D.3 Bubble Growth Rate Function 73
Vita 75

List of Tables

Table 1 Nuclear and Physical Properties of Noble Gases and Mercury 29
Table 2 Calculated Solubility Values of Noble Gases in Mercury..... 29
Table 3 Henry's Law Constants of Noble Gases in Mercury..... 37
Table 4 Inert Gas Diffusion Coefficients in Mercury 41

List of Figures

Figure 1 Target Vessel Module [Pointer, 2001] 3
Figure 2 Schematic of the Simulation Domain 10
Figure 3 Schematic of Numerical Computational Domain 11
Figure 4 Mixture Pressure Profile in the Enclosure at Four Different Phases (ωt=0, π/2, π, 3π/2) during the Cycle of No.2000 13
Figure 5 Mixture Velocity Profile in the Enclosure at Four Different Phases (ωt=0, π/2, π, 3π/2) during the Cycle of No.2000 14
Figure 6 Streaming Velocity Profile in the Enclosure at t=0.2s with Vibration Velocity Amplitude of 0.2m/s 16
Figure 7 Time-Dependent Pressure Profile at the Left Wall for the Cycle of No.1999 and No.2000 17
Figure 8 Frequency Spectrum of the Pressure Fluctuations at the Left Wall 18
Figure 9 Mixture Pressure Profile in the Enclosure at Four Different Phases (ωt=0, π/2, π, 3π/2) during the Cycle of No.2000 20
Figure 10 Mixture Velocity Profile in the Enclosure at Four Different Phases (ωt=0, π/2, π, 3π/2) during the Cycle of No.2000 21
Figure 11 Streaming Velocity Profile in the Enclosure at t=0.2s with Vibration Velocity Amplitude of 0.4m/s 22
Figure 12 Time-Dependent Pressure Profile at the Left Wall for the Cycle of No.1999 and No.2000 23
Figure 13 Frequency Spectrum of the Pressure Fluctuations at the Left Wall 24
Figure 14 Temperature-Dependent Helium Solubility in Mercury (bar^{-1}) 31
Figure 15 Temperature-Dependent Neon Solubility in Mercury (bar^{-1}) 32
Figure 16 Temperature-Dependent Argon Solubility in Mercury (bar^{-1}) 33
Figure 17 Temperature-Dependent Krypton Solubility in Mercury (bar^{-1}) 34
Figure 18 Temperature-Dependent Xenon Solubility in Mercury (bar^{-1}) 35
Figure 19 Comparison of Inert Gas Solubility in Mercury at 300K 36

Figure 20 Comparison of Henry's Law Constants for Inert Gases in Mercury at 300K .. 38
Figure 21 Time-Dependent Helium Bubble Radius and Gas Concentration in Mercury . 43
Figure 22 Time-Dependent Xenon Bubble Radius and Gas Concentration in Mercury .. 44
Figure 23 Time-Dependent Helium Bubble Radii and Gas Concentration in Mercury ... 46
Figure 24 Time-Dependent Xenon Bubble Radii and Gas Concentration in Mercury..... 47

1 Introduction

1.1 Background

Neutron scattering is an advanced experimental method used in a wide variety of research activities to study the arrangement, motion, and interaction of atoms in materials. The advantage of using neutron scattering is that it can provide important information on material structures that often cannot be revealed by other methods, such as electron microscopy and X-ray diffraction method.

The Spallation Neutron Source (SNS), under construction in the Oak Ridge National Laboratory (ORNL), is aimed to be the most powerful spallation source in the world upon completion. The accelerator-based pulsed neutron source will have a 1-GeV energy proton beam of up to 1.4 megawatts power impinging on the target to produce neutrons as the result of the collision between the heavy target material atoms and the proton beam. The beam energy is deposited in pulses of one microsecond duration, with sixty pulses delivered each second. The neutrons are subsequently collimated and moderated to accommodate the needs in the scientific research activities. The SNS will greatly benefit the scientific community by providing valuable information on material structures that cannot be obtained using other methods and facilitating research activities in material, biological and physical sciences.

The SNS is also to be the first scientific device using pure mercury as the target material for neutron production. The usage of liquid mercury can exclude the material degradation effect due to the radiation damage common to solid target materials. Mercury atoms are also rich in neutrons and susceptible to spallation reactions. However, the deposition of each proton beam pulse in mercury will result in high amplitude pressure waves in the target region. The interaction between the pressure waves and the target vessel will cause cavitation damage to the pressure boundary. This will reduce the lifetime and may limit beam power handling capacity of the target.

One promising method to mitigate the cavitation damage to the pressure vessel is to inject very small helium gas bubbles into mercury. The reason for using microbubbles

is that they can respond to the one-microsecond pressure rise caused by the proton pulse deposition due to their high resonance frequencies. A suitable bubble size distribution in mercury will add compressibility and lower the amplitude of the pressure waves, which will reduce cavitation damage. Also the small bubbles tend to stay entrained in mercury flow, while large bubbles are more buoyant and will stratify in the target mercury flow circuit.

A full-scale test facility of the SNS target loop (TTF) was constructed to perform the flow test of the mercury and obtain expertise with target body remote handling. The TTF is also incorporated with a gas injection system for the purpose of testing gas bubble deployment at a full-scale level [Riemer, et al, 2004]. The components in the TTF are identical to those planned for application in the actual SNS target system.

The TTF has been used in a series of tests to characterize the transient and steady state behavior of the target flow. Also initial helium bubble injection experiments have been performed in the TTF, which exposed the difficulties in obtaining the desired gas volume fraction of 0.5% and bubble diameter near 30μm. This motivates the careful study of small bubble behaviors in mercury. Inert gas types other than helium may also be good candidates for bubble injection and they are also studied along with helium. For example, the solubility of xenon is much smaller than helium in mercury, which promotes long-term stability of xenon bubbles when deployed in the mercury flow circuit.

It is also interesting to examine the flow field induced by periodic proton beam insertion near the target window, which causes window deflection. The SNS target module consists of the mercury target vessel and the mercury feed and return lines. The target module is illustrated in Figure 1. The green arrows indicate the mercury flow direction in the target and the cyan arrow shows the orientation of the incoming proton beam. With gas bubble injection, the target contains bubbly flow, which will be perturbed by the deposition of a proton beam pulse of one microsecond duration, sixty pulses per second. The beam deposition will cause the target structure to ring at resonance. The acoustic drive is modeled here as a sinusoidally vibrating window, which drives flow and also forms the boundary condition and initial condition for the numerical simulation. The flow field is modeled by solving the response of the mercury-helium system subject to a

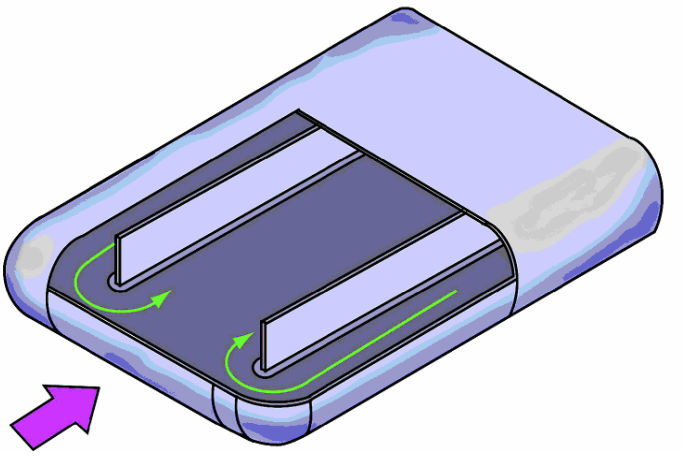

Figure 1 Target Vessel Module [Pointer, 2001]

sinusoidally vibrating target window with frequency of 10kHz. The bubble resonance frequency of a 30μm diameter bubble in mercury is very high (around 100kHz for isothermal gas behavior), so the chosen driving frequency is well below the resonance value.

1.2 Purpose and Organization of the Thesis

The interaction between gas bubbles and the liquid mercury is idealized and studied numerically using an extension of techniques developed to examine the acoustic streaming phenomenon in single-phase systems. The simulation was developed to study flows induced by pressure waves caused by proton pulses. Such flows were not considered in the initial target design and are scaled herein. The target is considered to be a one-dimensional duct containing bubbly two-phase flow. The structure response to the beam pulses is modeled as periodic boundary condition at the beam window, which drives flow in the target.

The design of the inert gas injection system requires understanding of the behavior of the gas bubbles in mercury and the interaction between the gas bubbles and mercury. To understand the bubble behavior, the gas solubility in mercury is essential to know, since the bubble lifetime greatly depends on how much gas will dissolve in mercury. In this thesis, the solubility of different inert gases in mercury is theoretically evaluated. And the bubble behavior due to mass diffusion is studied based on the

solubility results.

The acoustics of the two-phase mixture under the perturbation due to beam deposition, specifically acoustic streaming**,** is simulated in a bubbly two-phase flow for the first time in the literature. The numerical simulation shows the magnitude of obtained streaming velocity is much smaller than the pumped mercury flow in the target and will not cause distortion to flow patterns and heat transfer in the target.

Inert gas solubility in mercury is theoretically investigated to provide a theoretical basis for the inert gas solubility measurement experiments, as there is still no available theoretical and experimental data available in the literature. This information is important for successful operation of inert gas bubble injection experiments in the SNS.

The bubble behavior under the effect of mass diffusion across the bubble wall is also simulated. Two different profiles of bubble size distribution are studied. The mass diffusion induced bubble behavior simulation based on the solubility results indicates that xenon bubbles creates a more viable and stable bubble population in mercury than helium bubbles, which shows xenon is another more promising candidate to add compressibility to pure mercury in the SNS target other than helium.

2 Acoustic Streaming in Mercury-Helium Two-Phase Flow

2.1 Literature Review

Since the first work by Rayleigh [1884] addressing the circulation phenomenon of air in Kundt's tubes, acoustic streaming has been extensively investigated. Acoustic streaming refers to the time-average velocity fields induced in single-phase compressible gas or liquid by a steadily oscillating acoustic field. The nonlinear acoustic field variables can be expanded about their equilibrium values, where the first order term is the primary sinusoidally oscillating field. The second order terms contain both the steady and second-order harmonic fields. The mechanism inducing the steady flow is the interaction between second order terms in the nonlinear sound field. The second harmonic term is filtered out using time average over a single primary cycle to generate the steady term. Though the magnitude of the streaming velocity remains quite small relative to the primary oscillatory particle velocity amplitude even at high sound intensity for most applications, this phenomenon has important potential engineering applications in micromechanical flow systems that use acoustic pumping.

Both Nyborg [1965, 1998] and Riley [2001] reviewed the theoretical fundamentals of acoustic streaming. Nyborg adopted an approach of successive approximation to the sound field variables. The steady streaming is included in the second-order approximation equation system. In Riley's work a time-average operation is directly applied to the governing partial differential equations, and the solution of the resulting equations gives the time averaged streaming velocity. Both methods render analytical solutions of the streaming velocity field and other field variables. However, due to the complexity of the governing equations, analytical solutions are only available for single-phase flow cases where the boundary geometries are simply planes, cylinders or spheres.

More recent work has been dedicated to the numerical solutions of the streaming velocity field. Kawahashi [1996], Yano [1999] and Farouk [2004] successfully simulated acoustic streaming phenomena in resonators containing single-phase gases, which have a

piston or moving wall vibrating harmonically to motivate the flow in a closed two-dimensional duct and create the sound field in the duct. The results show that streaming field has a circulatory pattern and the magnitude of the streaming velocity increases with increasing velocity amplitude of the oscillating piston or moving wall.

Some historical evaluations take into consideration the engineering applications of the acoustic streaming phenomenon. The secondary flow induced by acoustic streaming may enhance heat transfer. Wan and Kuznetsov [2001] simulated the cooling effect of a computer chip subjected to traveling waves and standing waves and found that the addition of an acoustic field increased the cooling efficiency up to three times the value of conventional convective cooling method. Deymier [2000] developed the theory for particle removal forces induced by second order terms in an acoustic field in a cleaning process for silicon wafers. The results show that the cleaning efficiency could be improved by subjecting the wafer to an incident acoustic wave of megahertz frequency in a wide range of incidence angles.

While a large amount of work has been done to investigate acoustic streaming in single-phase fluid flow, relatively much less effort is dedicated to two-phase (or two-component) flow systems. Wu [1997] investigated the streaming field induced by a vibrating bubble in an ultrasound field. He evaluated streaming patterns both inside and outside the bubble. Yarin [2001] solved analytically the streaming field due to the shape oscillation of a liquid droplet in an immiscible fluid, and evaluated the effect on mass transfer in a liquid-liquid system. Both of these studies focus on the streaming field inside a single bubble or droplet and in the surrounding fluid, without the consideration of the situation with a large number of dispersed bubbles or drops in the ambient liquid.

The reviews indicate that current studies on acoustic streaming contribute to the single-phase fluid flows. In this thesis project, the numerical simulation of acoustic streaming is applied to mercury-helium two-phase flow.

2.2 Physical Models

The numerical simulation of acoustic streaming in gas-liquid two-phase flow requires suitable modeling of two-phase flow and an accurate numerical scheme to solve

the governing equations under certain initial conditions and boundary conditions.

The two-fluid model has been extensively studied to model two-phase fluid flow by a large number of researchers [Ishii, 1975; Delhaye, 1981; Drew, 1983]. The model assumes inter-penetrating continuum for both two phases, so distinct partial differential equations describing single-phase fluid flow can be applied to two phases for mass, momentum and energy conservation. The two phases interact through interfacial mass, momentum and energy transfer. A complete two-fluid model needs detailed formulation of the interaction terms to form a solvable equation system.

For simplicity, only mass and momentum equations are illustrated from Ishii & Kocamystafafaogullari [1982]:

- Mass continuity equation

$$\frac{\partial \alpha_k \rho_k}{\partial t} + \nabla \cdot \left(\alpha_k \rho_k v_k \right) = \Gamma_k \qquad \textbf{(1)}$$

- Momentum equation

$$\begin{aligned} &\frac{\partial \alpha_k \rho_k v_k}{\partial t} + \nabla \cdot \left(\alpha_k \rho_k v_k v_k \right) = \\ &- \alpha_k \nabla p_k + \nabla \cdot \alpha_k \tau_k + \alpha_k \rho_k g + v_{\alpha,i} \Gamma_k + M_{i,k} - \nabla \alpha_k \cdot \tau_{i,k} \end{aligned} \qquad \textbf{(2)}$$

where:

- α_k is the volume fraction for phase k
- Γ_k is the mass transfer to phase k;
- $M_{i,k}$ is the interfacial drag
- τ_k is the shear stress tensor
- $\tau_{i,k}$ is the shear stress evaluated at the interface

In general, the dependent variable vector from the governing equations can be rewritten into the following forms, which are not exactly in conservative form.

$$\frac{\partial U(x,t)}{\partial t} + A(U(x,t)) \frac{\partial U(x,t)}{\partial x} = G(U(x,t)) \qquad \textbf{(3)}$$

If the matrix A has complex eigenvalues, the equation set is ill-posed, which

implies that small perturbations may grow unconditionally if not damped in some way. Though in some problems the ill-posed equations still render meaningful solutions, it is better to find a way to modify the original equations or modeling of interaction terms to obtain a well-posed system.

Besides the two-fluid model, a one-dimensional homogeneous two-phase model is also successfully applied in those cases where the relative motion between two phases is negligible [van Wijngaarden, 1972; Ruggles, 1987; Wang & Brennen, 1998; Preston, et al, 2000]. The homogeneous model uses mixture state variables.

For mercury-helium bubbly flow considered here, a continuum model for bubbly flow is adopted from van Wijngaarden [1972]. This model neglects the relative motion between the phases and compressibility of the liquid phase. The gas inside the bubble excludes mercury vapor and only consists of helium gas. The model couples the conventional continuity and momentum equations for the mixture with the Rayleigh-Plesset equation, which governs bubble dynamics. These equations govern pressure field and bubble size evolution. Bubbles remain spherical during oscillations and are permanent. The volume fraction of the gas phase is initially near 0.5%. The applicability of this model to helium-mercury bubbly flow here can be approximately validated by evaluating helium gas bubble terminal velocity in mercury with a bubble radius of 15μm [Lamb, 1932],

$$v_r = \frac{g\rho_l R^2}{3\mu_f} \qquad \textbf{(4)}$$

where:

- ρ_l is the density of mercury
- R is the bubble radius
- μ_f is the dynamic viscosity coefficient of mercury

Evaluated at the given mercury properties and bubble radius,

$$v_r = 0.0064\, m/s \qquad \textbf{(5)}$$

The bubble terminal velocity is small compared to convective velocities and

acoustic velocities in this system, indicating the homogeneous two-phase model can be applied to mercury-helium bubbly flow. Using the Rayleigh-Plesset equation to couple the bubble dynamics with fluid mechanics, the system is well posed. The complete system in one-dimensional format is described as follows with mixture state variables,

- Mass and momentum equation:

$$\frac{\partial \rho}{\partial t} + \frac{\partial (\rho u)}{\partial x} = 0 \qquad \textbf{(6)}$$

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} + \frac{1}{\rho}\frac{\partial p}{\partial x} = 0 \qquad \textbf{(7)}$$

- Rayleigh-Plesset equation

$$\rho_l \left(R\frac{D^2 R}{Dt^2} + \frac{3}{2}\left(\frac{DR}{Dt}\right)^2 + \frac{4\nu_l}{R}\frac{DR}{Dt} \right) + \frac{2S}{R} - \left(p_0 + \frac{2S}{R_0} \right)\frac{R_0^3}{R^3} + p = 0 \qquad \textbf{(8)}$$

where S is the surface tension of mercury.

Neglecting relative motion between phases and gas contribution to the mixture density implies a relationship between mixture density and the bubble radius,

$$\rho = \rho_l \left[1 + \frac{\alpha_0}{1-\alpha_0}\left(\frac{R}{R_0}\right)^3 \right]^{-1} \qquad \textbf{(9)}$$

where α_0 and R_0 are initial gas volume fraction and bubble radius, respectively.

2.3 Problem Description

To simulate the flow field in the target induced by the deposition of the proton beam near the target window, the target is considered as a one-dimensional closed duct initially filled with mercury and uniformly dispersed with uniform-sized helium bubbles. The structure response from the beam insertion is approximated as the sinusoidally vibrating window left wall, which performs as a sound source in the simulation and drives the flow in the duct. The duct is chosen with a length of 0.2m similar to the dimension of the actual target (Figure 2).

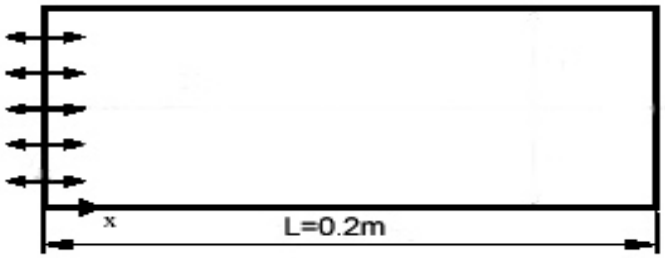

Figure 2 Schematic of the Simulation Domain

The vibrating frequency of the left wall is fixed at 10kHz. Two different vibrating velocity amplitudes of the left wall are used to simulate streaming velocity field in response to changes in sound intensity. The harmonically oscillating velocity of the left wall is given by $u_w=Asin(\omega t)$, where A is the maximum velocity of the left wall, and ω is the angular frequency of the vibration ($\omega=2\pi f$). Reflecting boundary conditions are used for both ends of the enclosure. A suitable numerical scheme is applied to solve the governing equations. The streaming velocity is obtained by a time-averaged operation on the instantaneous acoustic velocity field.

2.4 Numerical Methods

To solve the governing partial differential equations in Eulerian framework, the scheme is required to reformulate the equations into a discretized form using finite difference or finite volume methods on a fixed-grid basis. The difficulty and complexity in this framework stems from the discretization of Rayleigh-Plesset equation, which is a second-order partial differential equation.

A method developed by Colonius, et al [2000] treats the equations in the Lagrangian coordinate system through a finite volume scheme. The one-dimensional duct is initially divided into N subdivisions of equal length with N+1 interfaces on which field variables are defined and to be solved (Figure 3). Mass and momentum balance equations are integrated over each of these N cells. The key here is to convert the partial differential equations into ordinary differential equations, which excludes the spatial difference for the discretization of the partial differential equations. The Rayleigh-Plesset equation will

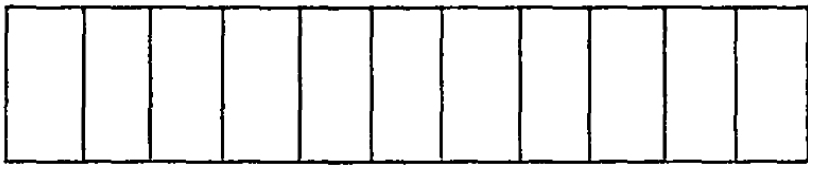

Figure 3 Schematic of Numerical Computational Domain

also be solved at each interface to update the radius values with the isothermal gas behavior determined from the work by Sawyer and Ruggles [2004].

For the subdivision with interface number of j and j+1, the integrated mass and momentum balance equations are transformed into the following equations,

$$\frac{d}{dt}\int_{x_j}^{x_{j+1}} \rho dx = 0 \qquad \textbf{(10)}$$

$$\frac{d}{dt}\int_{x_j}^{x_{j+1}} \rho u dx = p_j - p_{j+1} \qquad \textbf{(11)}$$

The integration of mass and momentum over the control volume is approximated by the following formula suggested by Colonius [2000],

$$\int_{x_j}^{x_{j+1}} f dx = \frac{x_{j+1} - x_j}{2}\left(f_j + f_{j+1}\right) + O(\Delta^3) \qquad \textbf{(12)}$$

In using Lagrangian methods, the positions on which the variables are defined are moving and the cell size also changes with the time. The compression and decompression the cell vary from time step to time step. Taking into account the moving cell boundaries, the spatial integration over each cell results in a system of ordinary differential equations for all cells. The accuracy of the numerical scheme will only depend on the integration method and the method for solving the ordinary differential equations.

Another useful equations addressing moving cell walls is as follows relating each wall position (x_j) and velocity (u_j),

$$\frac{dx_j}{dt} = u_j \qquad \textbf{(13)}$$

Combined with the Rayleigh-Plesset equation for each cell interfaces, the whole system contains 5(N+1)-2 ordinary differential equations for 5(N+1) unknown variables on $N+1$ cell boundaries. Two boundary conditions complete the system. As suggested by Colonius [2000], an implicit time marching scheme shall be used. At each time step, Newton's method is adopted to obtain the solution from the nonlinear system of discretized equations. In this case, the Jacobian matrix is band diagonal and easy to get root correction for convergence. A first-order Euler's method is chosen for the ordinary differential equations and small time step shall be employed. A mesh structure of 3000×5000 is used for space and time respectively in each vibration period.

2.5 Results and Discussion

The numerical simulation of acoustic streaming generated by the sinusoidal motion of the left wall is performed in the one-dimensional enclosure containing a mixture of mercury and uniformly dispersed helium gas bubbles. The initial pressure in the bulk liquid is taken as 3bar and the initial bubble radius is 15μm. The volume fraction of gas is 0.5% before the left wall begins to vibrate. The study is focused on the effect of two different velocity amplitudes of the left vibrating wall on the streaming velocity distribution at a fixed vibrating frequency. The computations begin with a quiescent mixture. A large number of vibration cycles are used to form an approximately steady acoustic field in the one-dimensional enclosure. Up to 2,000 cycles are run in the current numerical investigation. Two different cases are run with different left wall vibration amplitudes.

2.5.1 Case one with V_m=0.2m/s

In the first case, the wall velocity amplitude is chosen at 0.2m/s. Figure 4 and Figure 5 plot the pressure and velocity profiles respectively for four different times in the No. 2000 cycle, when ωt=0, π/2, π and 3π/2. The profiles are plotted only for the first twentieth portion of the enclosure.

The acoustic field created in the enclosure is approximately stable. The nonlinearity of the acoustic field is apparent. The profiles are propagated to the right and

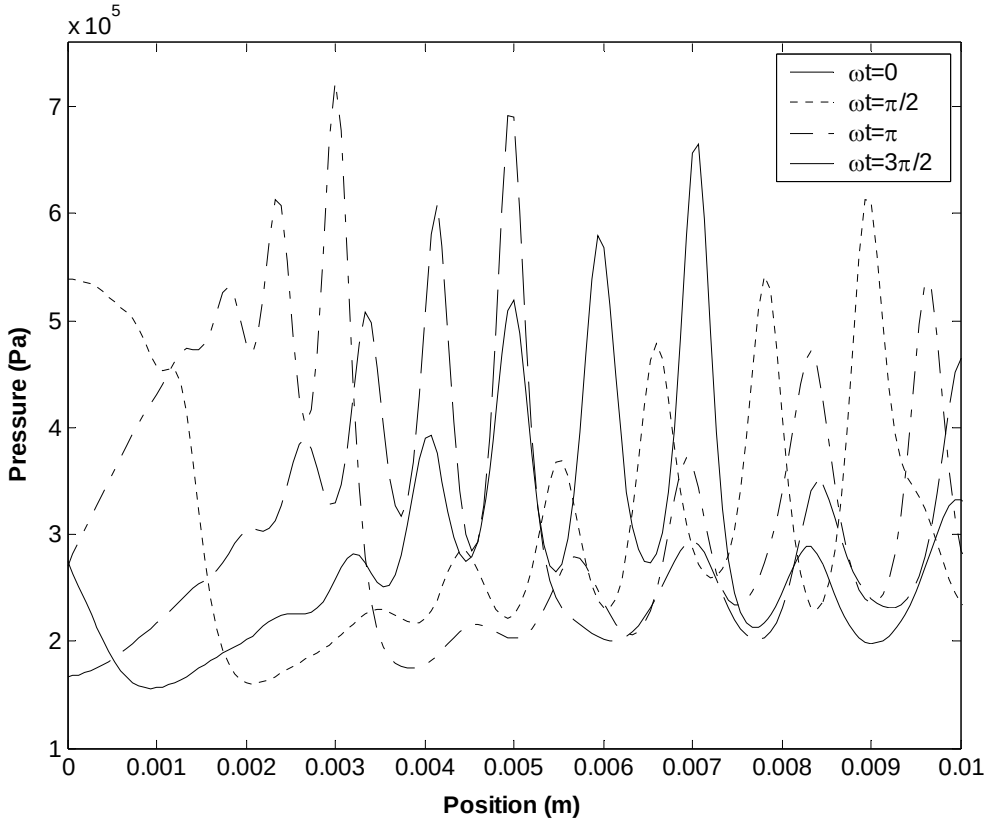

Figure 4 Mixture Pressure Profile in the Enclosure at Four Different Phases (ωt=0, π/2, π, 3π/2) during the Cycle of No.2000

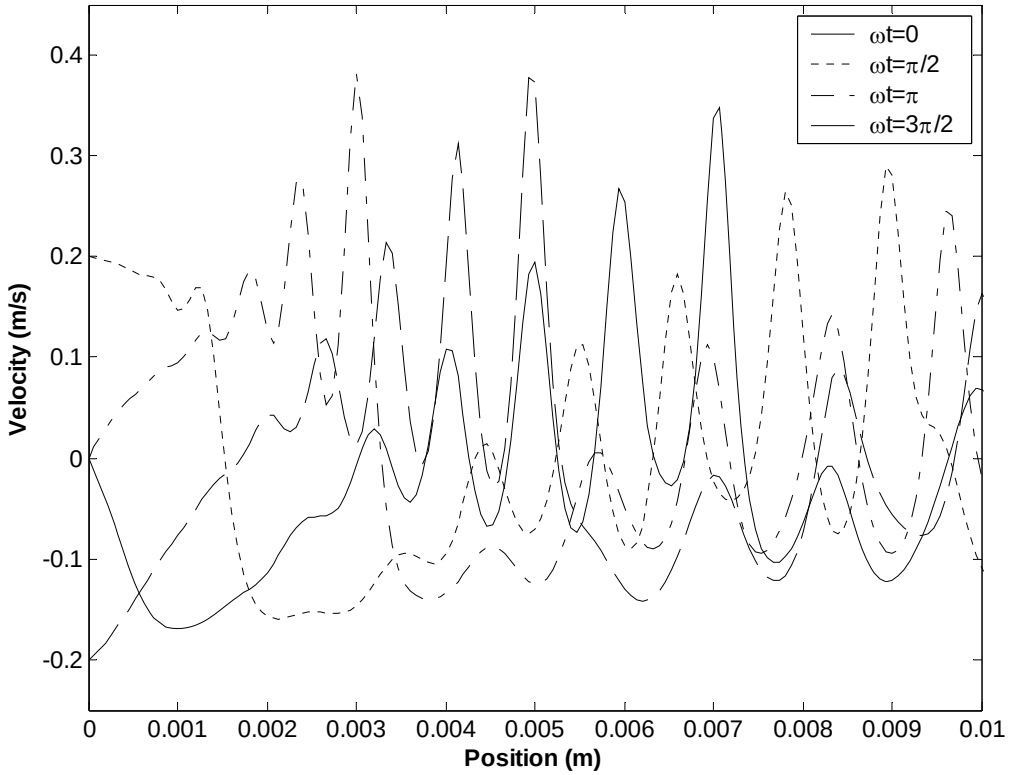

Figure 5 Mixture Velocity Profile in the Enclosure at Four Different Phases (ωt=0, π/2, π, 3π/2) during the Cycle of No.2000

reflected at the right wall. The profiles at subsequent times are similar. At ωt=0 and ωt=π, the pressure at the left wall reaches the maximum value, and the velocities are also maximum which demonstrates the near zero phase shift between pressure and velocity. This is expected for a weakly nonlinear system where the pressure and velocity are related by:

$$\Delta p \approx \rho c \Delta u \qquad \textbf{(14)}$$

where c is the sound velocity in the two-phase mixture. The driving frequency of 10kHz is much less than the bubble resonance frequency, which also contributes to keeping pressure and velocity in phase. The flow field is approximately periodic in time, and the maximum displacement velocity is about 0.4m/s, which is two times the vibrating velocity amplitude of the left wall.

A time average of the displacement velocity field for the last 100 cycles preceding the No. 2000 cycle is performed to obtain steady streaming velocity profile. After around 2000 cycles of vibrating excitation, the acoustic field is approximately stable in the enclosure. Figure 6 shows the streaming velocity distribution along the enclosure axis at t =0.2s at full enclosure length. The streaming velocity distribution fluctuates along the axis in part because a standing wave is present in the enclosure.

The shape of obtained streaming velocity profile (Figure 6) qualitatively agrees with single phase streaming evaluations investigated in Nyborg's review [Nyborg, 1965], where the streaming velocity is opposite to the direction of the traveling wave in the enclosure. It is noted that this one-dimensional evaluation creates streaming velocity solutions that do not satisfy mass continuity condition for the enclosure. A multi-dimensional representation would allow patterns of circulation to form satisfying continuity for the time averaged flow field.

The existence of acoustic streaming is further validated by the frequency spectra of left wall pressure history. Figure 7 and Figure 8 plot the time variation of pressure at the left wall (No.1999 & No.2000 cycle) and the power spectral density of the pressure variation at the left wall. The pressure shape is distorted from the pure sine waveform as shown in Figure 7. A frequency spectral analysis of the pressure fluctuation (Figure 8)

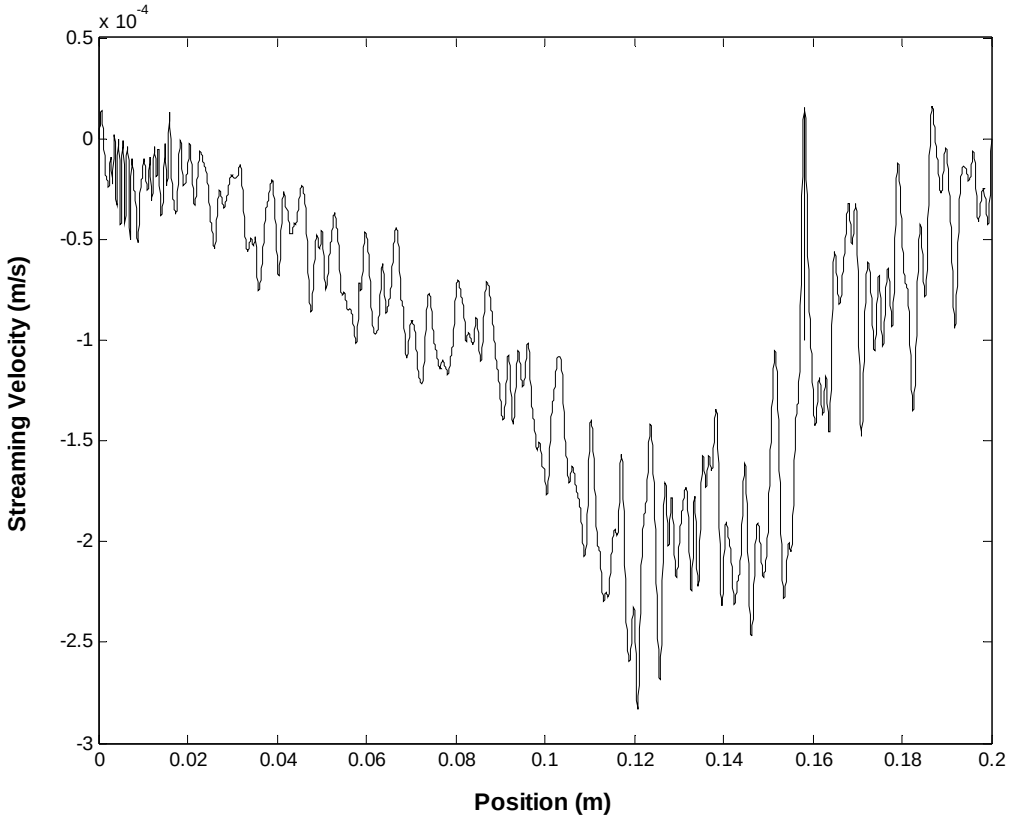

Figure 6 Streaming Velocity Profile in the Enclosure at t=0.2s with Vibration Velocity Amplitude of 0.2m/s

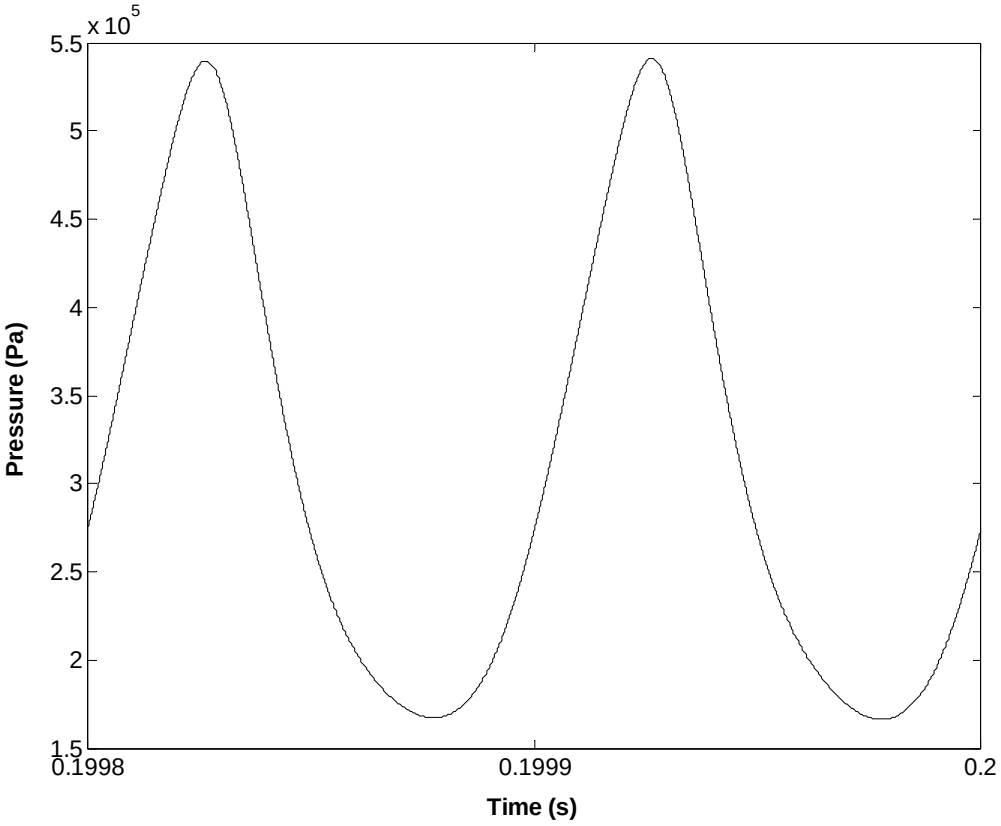

Figure 7 Time-Dependent Pressure Profile at the Left Wall for the Cycle of No.1999 and No.2000

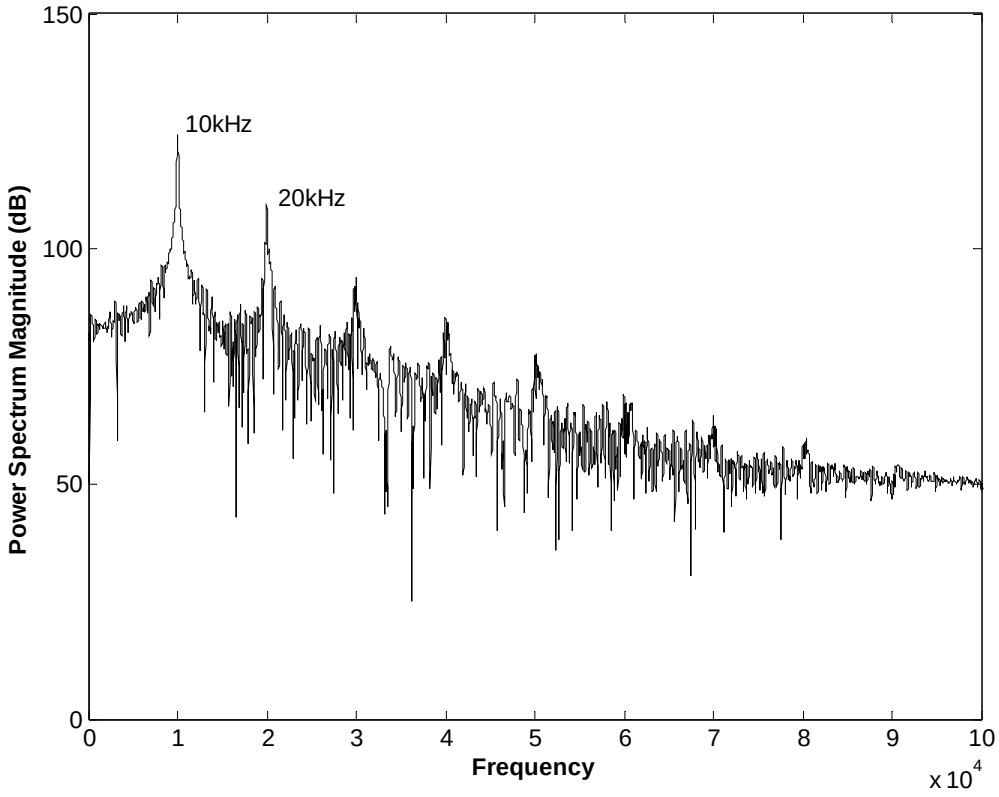

Figure 8 Frequency Spectrum of the Pressure Fluctuations at the Left Wall

shows the second-order harmonic component is present, and higher harmonic components are also visible. The distortion in the pressure is caused by the higher-order harmonic components. This type of distortion in the pressure is typical for situations in single-phase flow where streaming is present.

2.5.2 Case two with V_m=0.4m/s

In the second case, the velocity amplitude is increased to 0.4m/s to study the effect of elevated amplitude on the profile of streaming velocity. The increased intensity of the sound source results in the strengthened nonlinearity in the sound field (Figure 9 and Figure 10). The pressure perturbation signal and velocity history have more sharp spikes than the first case. The pressure and instantaneous velocity are still in phase with each other, though the sound field is more nonlinear.

The streaming velocity distribution is again obtained through the time-average of the instantaneous velocity profiles in the last 100 cycles preceding the No.2000 cycle. After around 2000 cycles of vibrating excitation, the acoustic field is approximately stable in the enclosure. Figure 11 shows the streaming flow field along the enclosure axis at t=0.2s with the vibrating velocity amplitude of 0.4m/s. It can be seen that the increased vibrating velocity amplitude has changed the distribution of streaming velocity and pressure amplitude. More violent vibrating motion of the left wall causes the acoustic field to be more nonlinear, which increases the streaming velocity.

Figure 12 and Figure 13 plot the pressure variation in time and frequency spectrum, respectively. The frequency spectrum shows more harmonic components are present, which contribute to the distorted pressure waveform at the left wall.

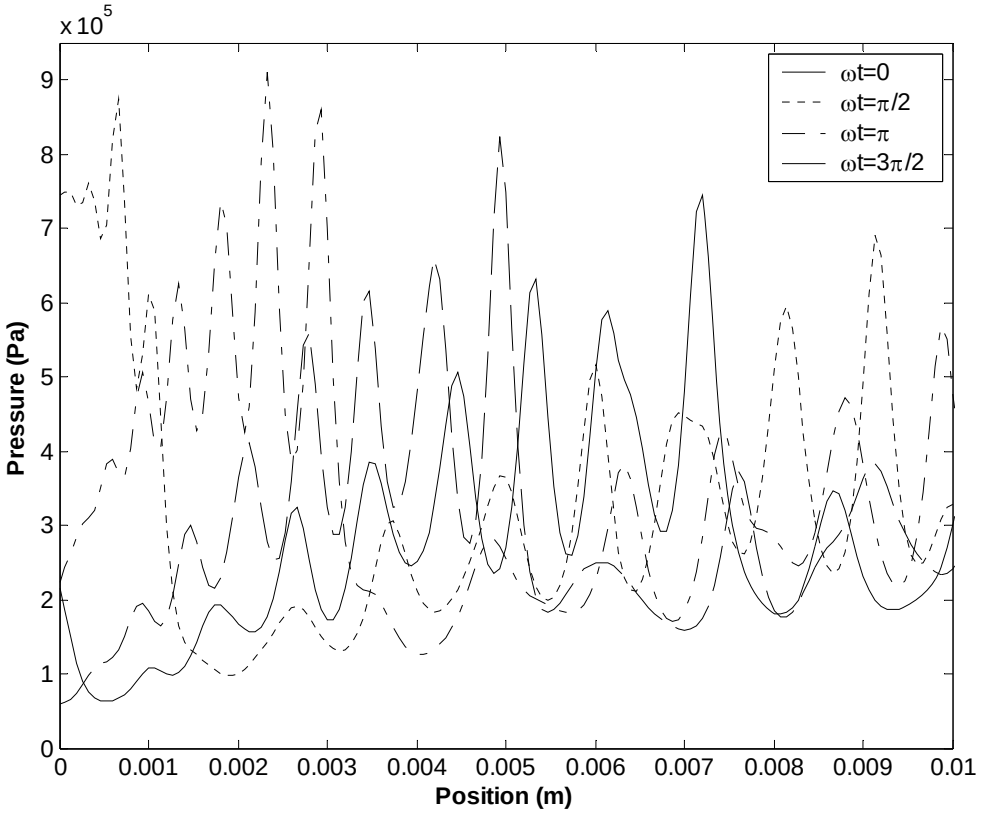

Figure 9 Mixture Pressure Profile in the Enclosure at Four Different Phases (ωt=0, π/2, π, 3π/2) during the Cycle of No.2000

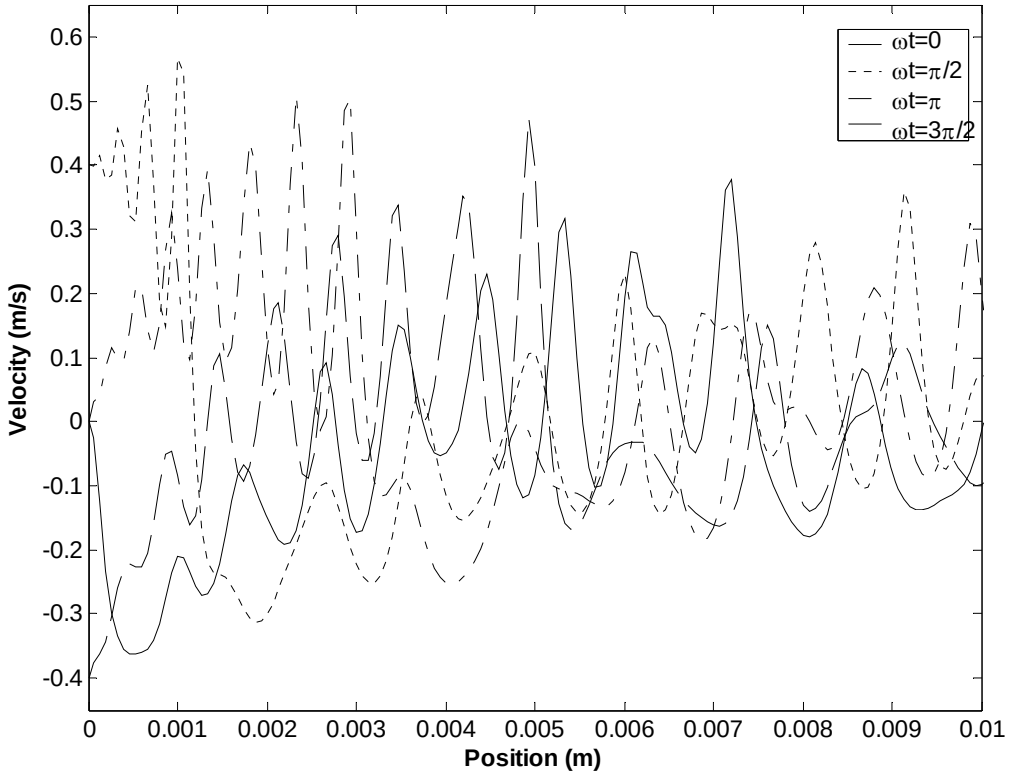

Figure 10 Mixture Velocity Profile in the Enclosure at Four Different Phases (ωt=0, π/2, π, 3π/2) during the Cycle of No.2000

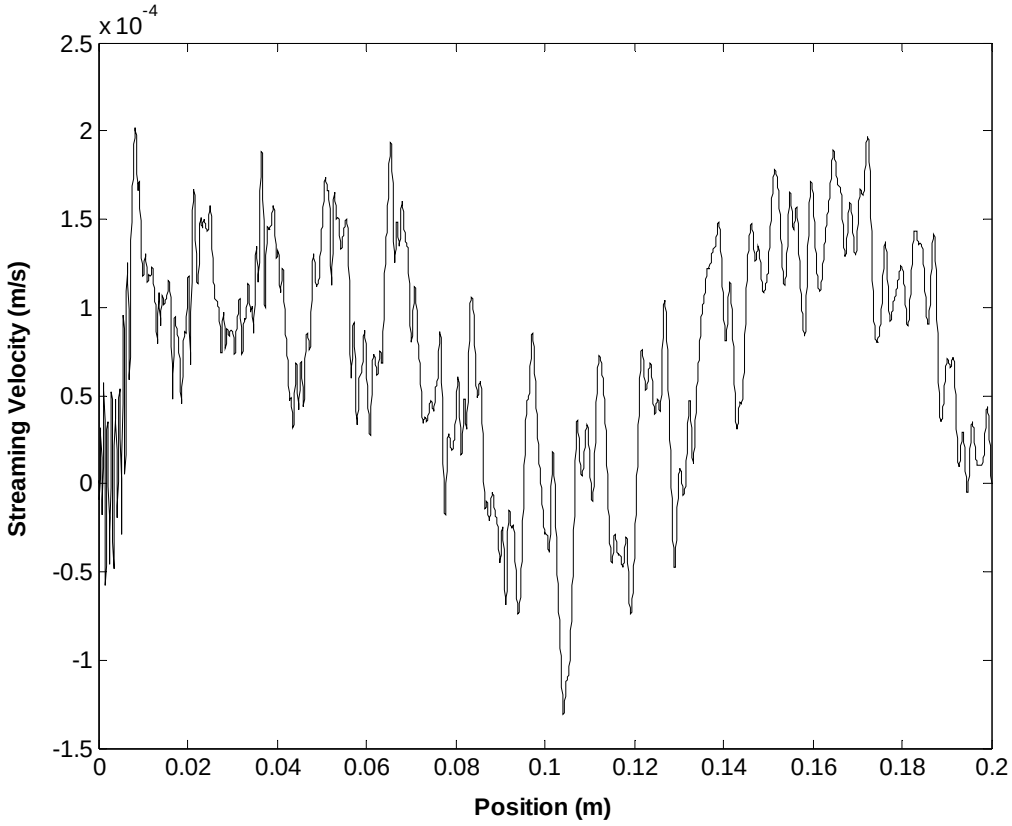

Figure 11 Streaming Velocity Profile in the Enclosure at t=0.2s with Vibration Velocity Amplitude of 0.4m/s

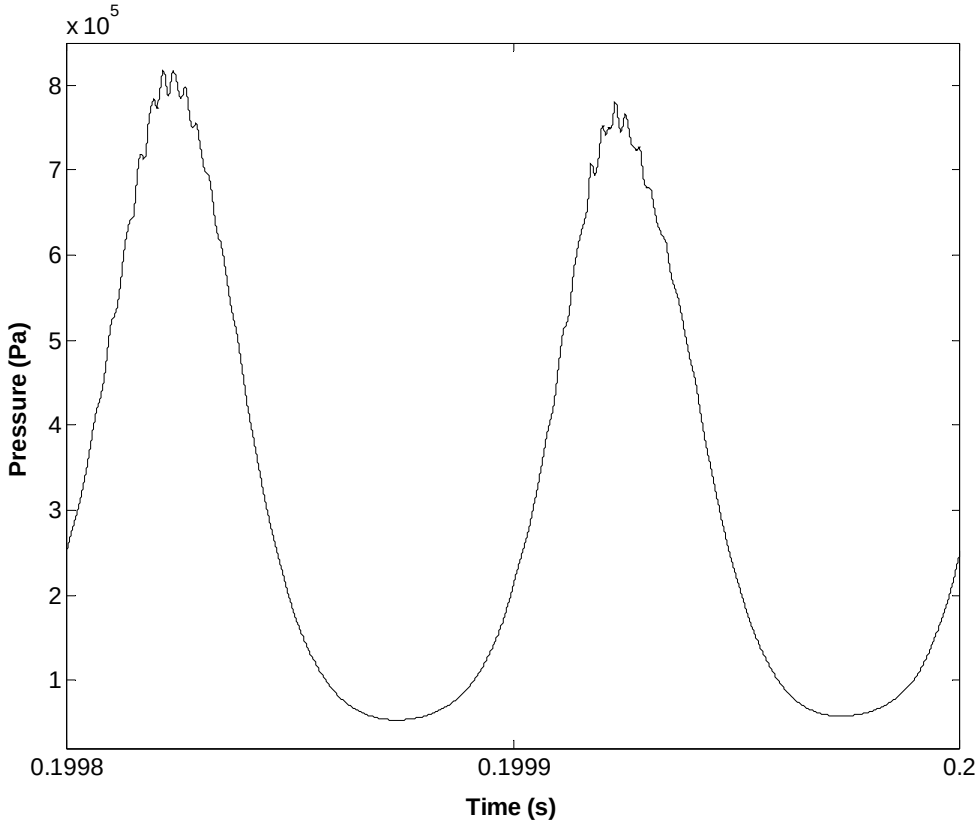

Figure 12 Time-Dependent Pressure Profile at the Left Wall for the Cycle of No.1999 and No.2000

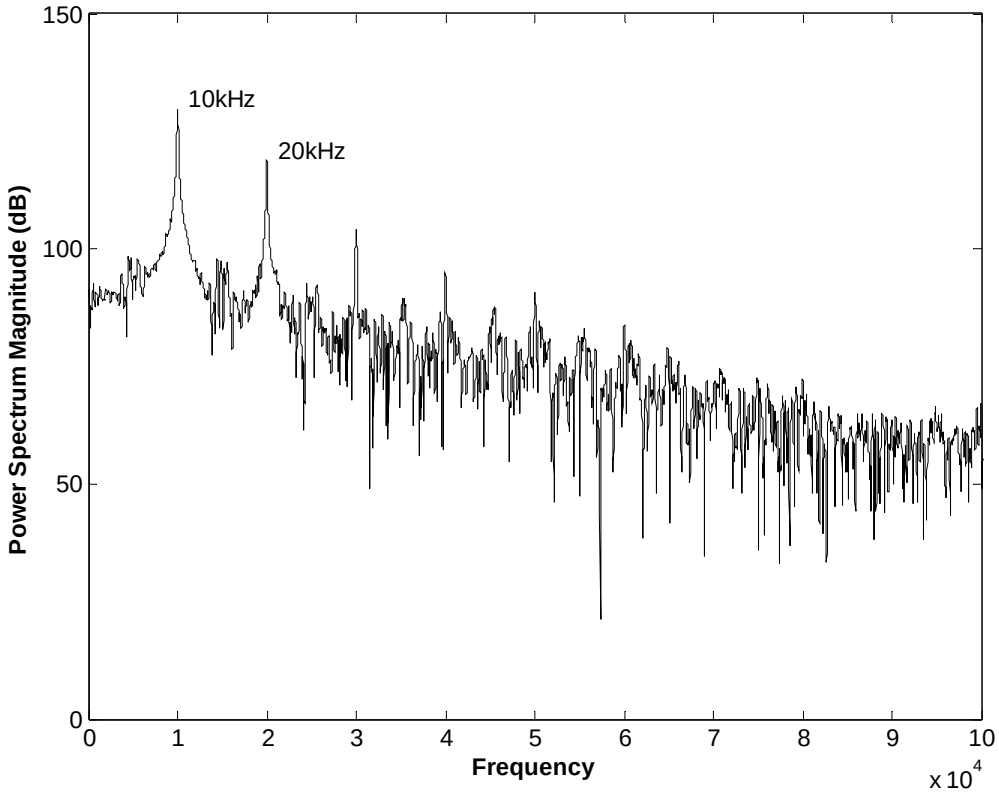

Figure 13 Frequency Spectrum of the Pressure Fluctuations at the Left Wall

2.5.3 Conclusions

The simulation of acoustic streaming in gas-liquid two-phase flow is performed based on the numerical solution of a one-dimensional bubbly flow model using a finite-volume Lagrangian method. The bubbly flow model does not include damping due to thermal transport between the two phases. The only attenuation considered is caused by viscous damping effect. The homogeneous model also excludes the relative motion between gas and liquid, which may also contribute to damping and attenuation.

The streaming velocity distribution in the target is obtained after applying time average to the instantaneous velocity field in the computation domain. Acoustic field and fluid flow are generated due to the harmonic vibration of the left wall. Compared with the convective velocity due to the pumped mercury flow in the target, with mean flow velocity of order 2m/s, the predicted magnitude of streaming velocity is quite small, and will not cause distortion to flow patterns and heat transfer in the target.

3 Theoretical Investigation of Inert Gas Solubility in Mercury

3.1 Introduction

Initial helium gas bubble injection experiments at a full-scale level in the TTF showed the difficulties in obtaining anticipated gas volume fraction and bubble size distribution in mercury [Riemer, et al, 2004]. Much smaller gas volume fraction was observed than expected. These results motivated the study of the mixture properties and the bubble behavior in the solution. The gas solubility in mercury must be known to understand the bubble behavior, since the bubble lifetime will depend on how much gas will dissolve in mercury [Epstein & Plesset, 1950]. Inert gas solubility in liquid metals is also an important parameter for applications in nuclear reactor engineering, where the formation and dissolution of inert gases from nuclear reactions can affect the heat transfer capability of liquid metal as heat-transfer agents, such as helium in liquid sodium. In this part of the thesis, the solubility of different inert gases in mercury is theoretically evaluated.

3.2 Model for Inert Gas Solubility Evaluation in Mercury

A number of papers have been dedicated to the theoretical investigation of gas solubility in liquid, most of which are based on the thermodynamic equilibrium between the solvent and the solute [Pierotti, 1965; Shoor and Gubbins, 1969; Neff and McQuarrie, 1973; Fukase and Satoh, 1976; Thormeier, 1970; Shpil'rain, et al, 2000]. Thermodynamic equilibrium state determines the amount of solvent and solute that can coexist under certain thermodynamic conditions.

Among previous work, the methods used by Thormeier [1970] and Shpil'rain, et al [2000] render good agreement with available experimental data. Though the methods are developed for liquid alkali metals, they are independent of the chosen substances. The theoretical evaluation for inert gases in mercury adopts the method from Shpil'rain, et al [2000]. This approach starts from formulating the free energy expression of the binary solution of inert gas in mercury using the Fowler-Guggenheim expression [Fowler and

Guggenheim, 1949], which is a function of quantities of the substances, individual properties and system temperature and pressure,

$$F = n_1\left\{-\Phi_1 - kT\left[\ln\left(Z_1 v_1\right)+1-\ln\frac{n_1}{n_1+n_2}\right]\right\} + n_2\left\{-\Phi_2 - kT\left[\ln\left(Z_2 v_2\right)+1-\ln\frac{n_2}{n_1+n_2}\right]\right\}+\frac{n_1 n_2}{n_1+n_2}\Delta\Phi_{12} \qquad \textbf{(15)}$$

where n_1 and n_2 are the number of moles of the solvent and the solute, respectively; Φ_1, Φ_2 and $\Delta\Phi_{12}$ are the molar binding energy of the two components and the correlation part, respectively; v_1 and v_2 denote molar volumes of the two substances; and Z_1 and Z_2 are two state variables. This expression is used to evaluate the chemical potentials for the two components.

The solubility is usually expressed as the ratio of the solute's number of moles to the total number of moles,

$$x_2^{(1)} = \frac{n_2}{n_1+n_2} \qquad \textbf{(16)}$$

which can be used to rewrite the free energy expression. Thermodynamic equilibrium state requires the equality between the chemical potential of gas in gas phase ($\varphi_2^{(2)}$) and the chemical potential of gas in mercury ($\varphi_2^{(1)}$). For the chemical potential of gas in mercury, it takes the partial derivative of free energy (F) against the number of moles of the solute in the solution (n_2). And the chemical potential of gas in gas phase, it has an explicit form [Neff and McQuarrie, 1973],

$$\varphi_2^{(1)} = \frac{\partial F}{\partial n_2}, \qquad \varphi_2^{(2)} = -kT\left[\ln Z_2 - \ln\left(\frac{p_2^{(2)}}{kT}\right)\right] \qquad \textbf{(17)}$$

Equating the two chemical potentials with some mathematical manipulations, a formulation for solubility evaluation is obtained [Shpil'rain, et al, 2000],

$$x_2^{(1)} = \frac{v_1 p_2^{(2)}}{RT} \exp\left(\frac{-F_{12}}{RT}\right) \qquad \textbf{(18)}$$

where $p_2^{(2)}$ is the cover gas pressure above the solution; F_{12} is the free energy of the gas in the solution and can be written in the following form,

$$F_{12} = U_{12} - TS_{12} \qquad \textbf{(19)}$$

in which U_{12} is the temperature-dependent molar internal energy required to introduce a gas atom into mercury; S_{12} is the vibration entropy of gas dissolved in liquid. Both of them have certain formulations and can be evaluated using specific property values of the solute and the solvent [Shpil'rain, et al, 2000].

It remains to select property values of mercury and the chosen gases. Unfortunately, a range of nuclear or physical property values exists in the literature.

3.3 Numerical Results of Solubility Evaluation

Typical property values are excerpted from the literature [Radzig and Smirnov, 1985; Perry and Green, ed., 1997; Lide, ed., 2005]. The solubility results are dependent on the chosen values and may be very sensitive to some values chosen, especially for atomic radius. Either metallic radius or covalent radius are used to measure the atomic radius depending on the bonding types. The method used to measure atomic radius for elements other than noble gases is to measure the distance between two touching atoms. Since noble gas atoms do not form bonds, the values of noble gas atomic radius can only be measured as van der Waals radius. The values in the literature range from 30 picometers (pm) to 130 pm for the helium atomic radius. The choice of property values would be improved if experimental solubility results are available for validation.

Using Table 1, inert gas solubility is evaluated using the aforementioned model. The solubility depends on system temperature and pressure. The solubility of helium, neon, argon, krypton and xenon in mercury in a temperature range from 300K to 600K is computed. Since the solubility is approximately proportional to cover gas pressure (mercury vapor pressure is negligible compared to the cover gas pressure), the results will be represented in the unit of molar fraction per bar of system pressure [Table 2].

Table 1 Nuclear and Physical Properties of Noble Gases and Mercury

Substance	Atomic Weight (a.m.u)	Atomic Radius (10^{-10}m)	Effective Charge	Polarizability (10^{-30} m^3)
Helium	4.0026	0.31	1.7	0.205
Neon	20.18	0.71	8.8	0.396
Argon	39.948	0.98	12.65	1.641
Krypton	83.8	1.12	20.1	2.484
Xenon	131.29	1.31	24.3	4.044
Mercury	200.59	1.51	2	5.02

Table 2 Calculated Solubility Values of Noble Gases in Mercury

T (K)	Inert Gas Type				
	Helium	Neon	Argon	Krypton	Xenon
300.0	8.3658×10^{-5}	2.8808×10^{-6}	6.0368×10^{-8}	5.6825×10^{-9}	9.7188×10^{-11}
350.0	7.8241×10^{-5}	2.7868×10^{-6}	5.8931×10^{-8}	5.5873×10^{-9}	9.5879×10^{-11}
400.0	7.4308×10^{-5}	2.7175×10^{-6}	5.7866×10^{-8}	5.5165×10^{-9}	9.4904×10^{-11}
450.0	7.1329×10^{-5}	2.6644×10^{-6}	5.7047×10^{-8}	5.4619×10^{-9}	9.4150×10^{-11}
500.0	6.8995×10^{-5}	2.6224×10^{-6}	5.6397×10^{-8}	5.4184×10^{-9}	9.3550×10^{-11}
550.0	6.7120×10^{-5}	2.5883×10^{-6}	5.5869×10^{-8}	5.3831×10^{-9}	9.3061×10^{-11}
600.0	6.5580×10^{-5}	2.5601×10^{-6}	5.5431×10^{-8}	5.3537×10^{-9}	9.2654×10^{-11}

The results show that the solubility values for the same gas slightly decrease as the system temperature increases, but have significant dependence on the gas atomic number (atom size) and nuclear and physical properties. The values drop dramatically with decreasing atomic number from 4 (helium) to 54 (xenon). For helium and xenon, the difference is up to 6 orders of magnitude. The largest part of the total energy in introducing the gas atoms into mercury is the energy of the formation of holes in liquid for gas atoms [Shpil'rain, Skovorod'ko and Mozgovoi, 2000]. This explains that the lighter noble gas types have bigger solubilities than the heavier ones. So the results are very sensitive on the chosen radius values. The smaller the gas atomic radius used, the bigger the solubility predicted.

Figure 14 to Figure 18 plot the temperature-dependent inert gas solubility in mercury with the temperature ranging from 300K to 600K. The solubility is shown to drop approximately exponentially against system temperature. The results also indicate that the temperature rise due to proton beam deposition in the SNS target will drive the dissolved gas out of mercury. A comparison of different inert gas solubility in mercury is plotted in Figure 19 at the system temperature of 300K.

3.4 Henry's Law Constant

Gases dissolve into liquids to form solutions. The dissolution is an equilibrium process and some equilibrium constants can be used to correlate the equilibrium state. For most gases, the concentration of a solute gas in a solution is directly proportional to the partial pressure of that gas above the solution. This relationship can be modeled in an empirical law named after J. W. Henry using so-called Henry's law constant. The constant then can be used to determine the gas concentration in the solution approximately under certain cover gas pressure, which is another method to represent the solubility [Barton, 1991]. The Henry's law states as follows,

$$p = K_H c \qquad \textbf{(20)}$$

where p is the gas partial pressure above the solution, K_H *is* Henry's law constant and c is the gas concentration in the solution. As yet another form to represent the gas solubility in liquid, Henry's law constant can be evaluated from the calculated solubility results (molar

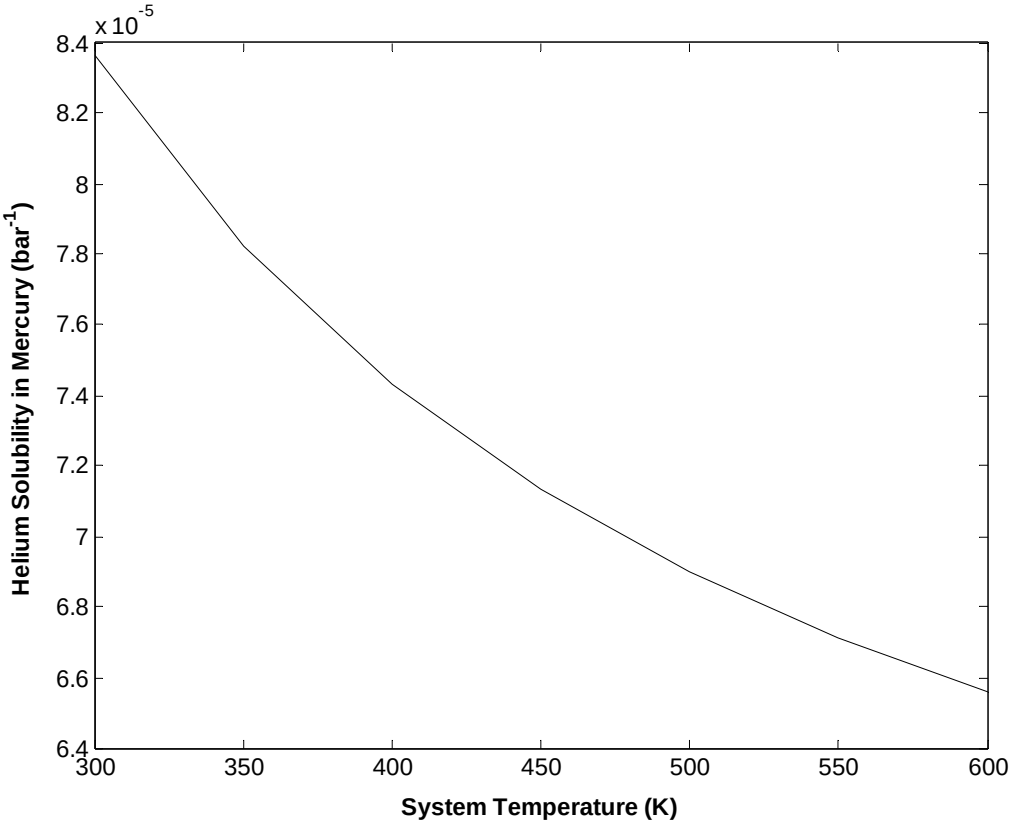

Figure 14 Temperature-Dependent Helium Solubility in Mercury (bar^{-1})

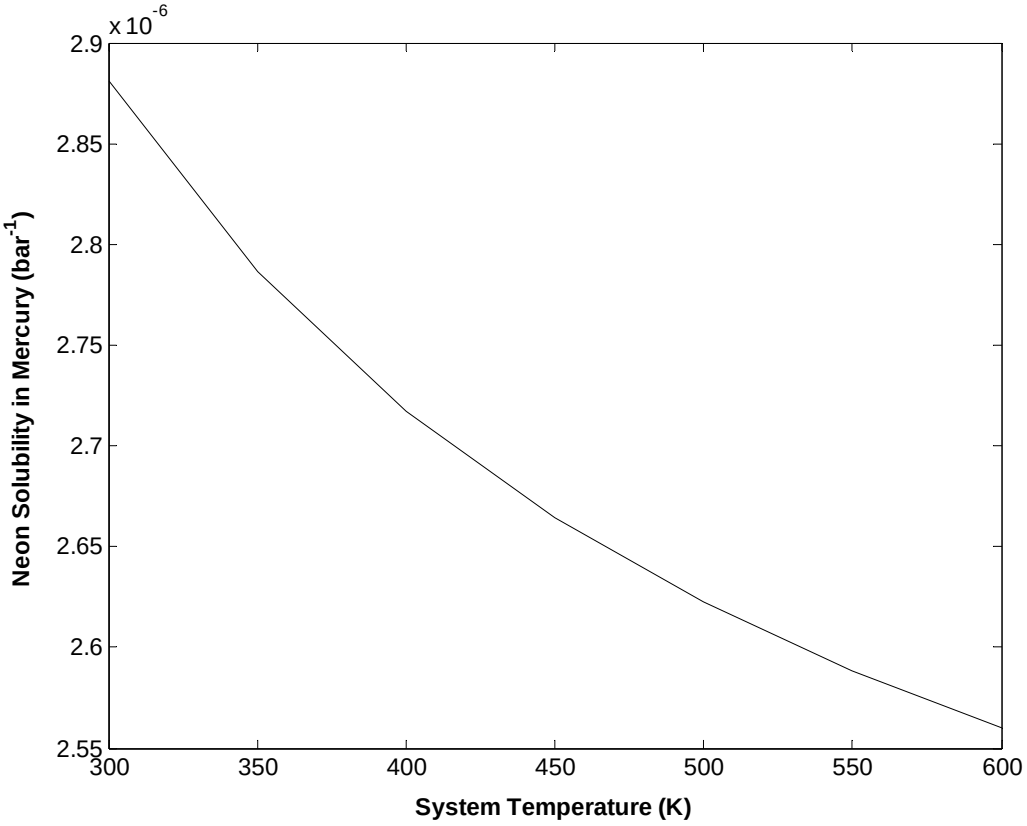

Figure 15 Temperature-Dependent Neon Solubility in Mercury (bar^{-1})

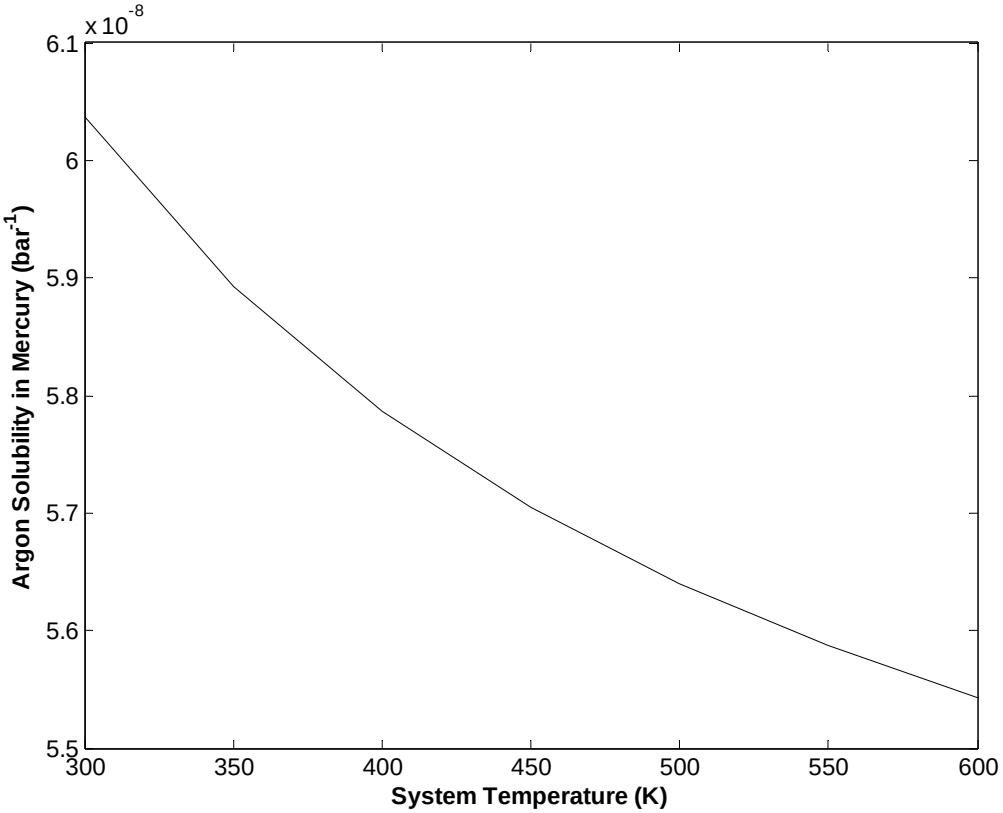

Figure 16 Temperature-Dependent Argon Solubility in Mercury (bar^{-1})

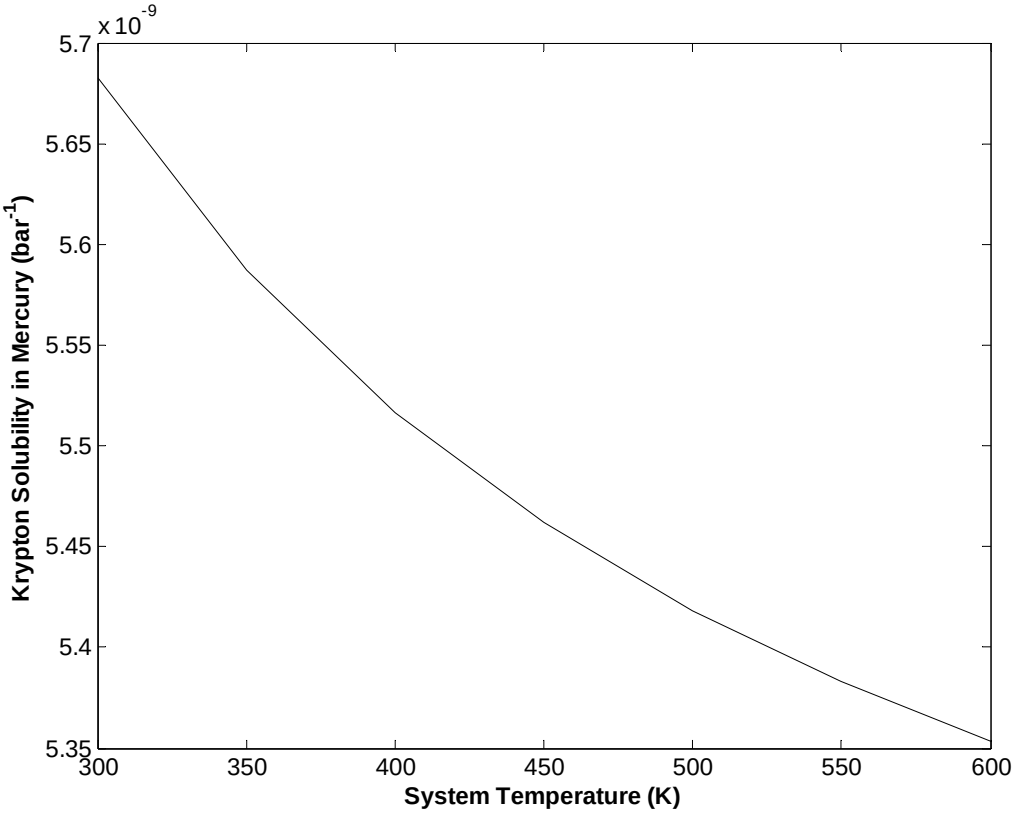

Figure 17 Temperature-Dependent Krypton Solubility in Mercury (bar^{-1})

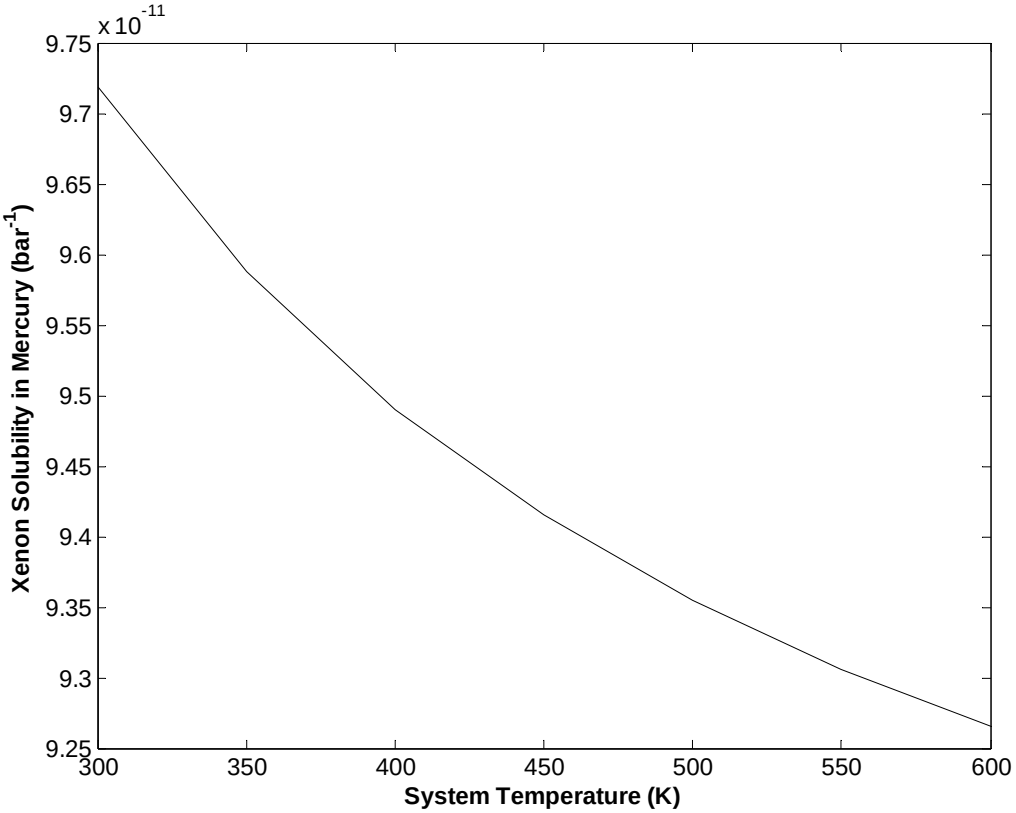

Figure 18 Temperature-Dependent Xenon Solubility in Mercury (bar^{-1})

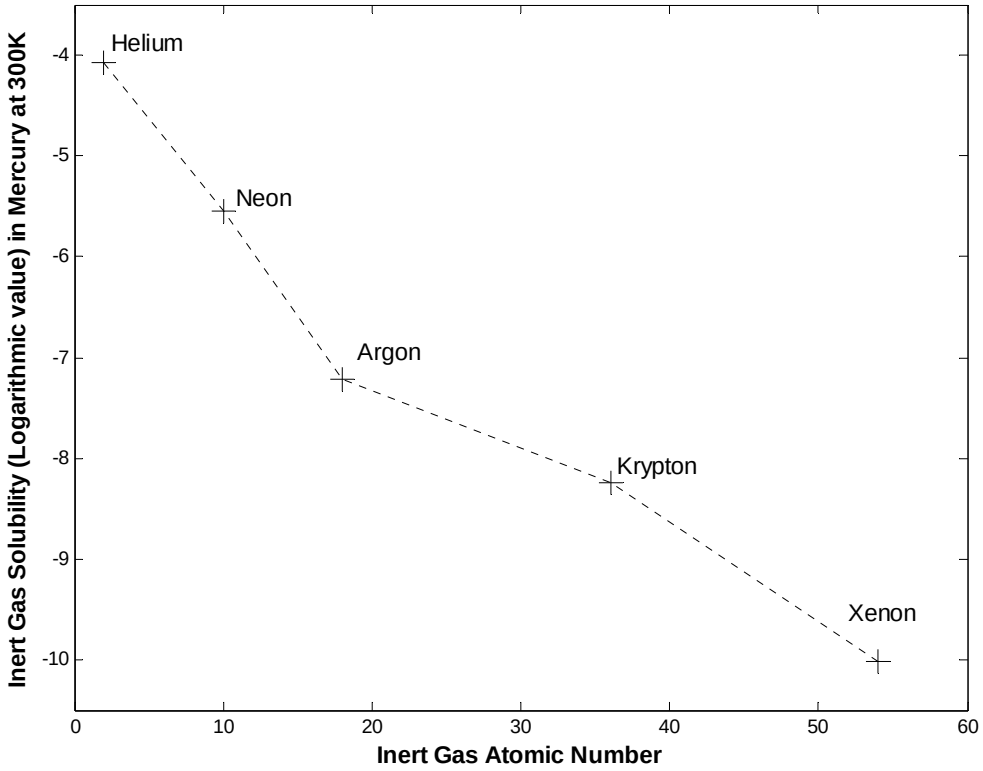

Figure 19 Comparison of Inert Gas Solubility in Mercury at 300K

fraction). To facilitate the study of helium and xenon bubble behavior in mercury in the following chapter, the unit of gas concentration in mercury is taken as kg/m^3, which is the same as the density unit. The computed values are in Table 3.

Again the values differ by several orders of magnitude. A comparison of Henry's law constants is plotted in Figure 20. With evaluated Henry's law constant, gas solubility at different gas partial pressure can be computed easily. For example, the gas concentration near the gas bubble wall in the liquid is conveniently obtained using Henry's law constant and gas pressure inside the bubble.

Table 3 Henry's Law Constants of Noble Gases in Mercury

T (K)	Inert Gas Type				
	Helium	Neon	Argon	Krypton	Xenon
300.0	4.426×10^{6}	2.5495×10^{7}	6.1460×10^{8}	3.1125×10^{9}	1.1616×10^{11}
350.0	4.7328×10^{6}	2.6356×10^{7}	6.2959×10^{8}	3.1656×10^{9}	1.1775×10^{11}
400.0	4.9833×10^{6}	2.7028×10^{7}	6.4117×10^{8}	3.2062×10^{9}	1.1895×10^{11}
450.0	5.1915×10^{6}	2.7567×10^{7}	6.5038×10^{8}	3.2382×10^{9}	1.1991×10^{11}
500.0	5.3670×10^{6}	2.8008×10^{7}	6.5788×10^{8}	3.2642×10^{9}	1.2068×10^{11}
550.0	5.5170×10^{6}	2.8377×10^{7}	6.6410×10^{8}	3.2857×10^{9}	1.2131×10^{11}
600.0	5.6465×10^{6}	2.8690×10^{7}	6.6934×10^{8}	3.3037×10^{9}	1.2184×10^{11}

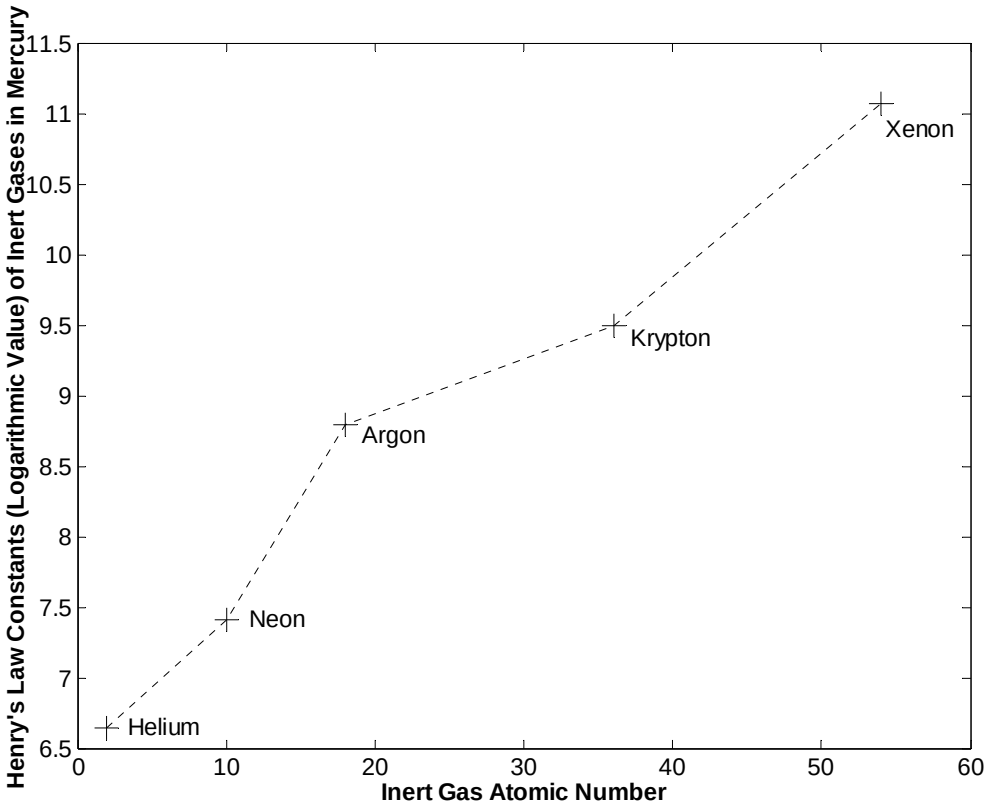

Figure 20 Comparison of Henry's Law Constants for Inert Gases in Mercury at 300K

4 Mass Diffusion Induced Inert Gas Bubble Behavior in Mercury

4.1 Introduction

Inert gas is to be injected into mercury to form a void fraction of 0.5%. The gas bubble size will vary in time under the effect of several different mechanisms. For example, in a time-dependent pressure field, bubble growth can be studied using the Rayleigh-Plesset equation. Another important effect is mass diffusion induced bubble growth, resulting from the concentration gradient at the gas bubble wall. In the current study, mercury is initially saturated with gas under a cover gas pressure. The injected bubbles will shrink if the gas concentration in mercury is less than that at the bubble wall, which causes gas atoms to diffuse into the mercury across the bubble wall. The objective here is to study bubble behavior under the mass diffusion effect. The phenomenon of mass diffusion induced bubble growth is important in the effort of maintaining a desired volume fraction of gas bubbles in mercury and the desired bubble diameter.

Many analytical models describing bubble growth rate have been developed in the literature [Epstein and Plesset, 1950; Scriven, 1959; Manley, 1960; Barlow and Langlois, 1962]. Most of these models exclude the convective term caused by relative velocity between gas bubble and liquid at the bubble wall. The model by Epstein and Plesset [1950] provides an approximate solution to the bubble behavior in good agreement with experimental data. This model will be used to predict helium and xenon bubble behavior here. However, this model doesn't consider the change in ambient gas concentration caused by gas dissolution from the bubbles. This effect is included to complete the bubble behavior study.

4.2 Diffusion Coefficient of Helium and Xenon in Mercury

The study of mass diffusion induced bubble behavior requires the knowledge of gas diffusion coefficient in the liquid. The diffusion coefficients of helium and xenon in mercury are not available in the literature, so some existing theoretical models are used to

estimate the values. These models are useful in obtaining orders of magnitude for the diffusion coefficient values. Here, the classical Stokes-Einstein equation is used, which applies for the binary mixture considered,

$$D = \frac{kT}{6\pi\mu_l r_g} \qquad \textbf{(21)}$$

where D is the diffusion coefficient; k is the Boltzmann constant; μ_l is the liquid (solvent) dynamic viscosity and r_g is the gas (solute) atomic radius.

The formula is shown to be applicable in obtaining the diffusion coefficient for spherical solute particles in the solvent, which appears as a continuum for the solute atoms [Thormeier, 1970]. This also requires the solute particle having a much smaller size than the solvent. The equation is applicable for He-Hg system. While for Xe-Hg system it will likely predict a diffusion coefficient value greater than the actual one, which is conservative in estimating the bubble growth or dissolution rate. Using the gas and mercury properties at system temperature of 300K, the diffusion coefficient is evaluated for helium, neon, argon, krypton and xenon in mercury. The computed results are listed in Table 4. The diffusion coefficient goes as one over the atomic radius in the Stokes-Einstein model. The values for helium and xenon are used to study helium and xenon bubble behavior in the following sections.

4.3 Model for Predicting Bubble Growth Rate

As indicated in the first section of this chapter, the model from Epstein and Plesset [1950] is used to predict the bubble behavior under the effect of mass diffusion. The model is an approximate analytical solution to the governing gas diffusion equation coupled with mass balance equation in the bubble,

$$R\frac{dR}{dt} = \frac{D}{\rho_g}\frac{c_\infty - c_s}{1 + 4S/3Rp_\infty}\left\{1 + R\left(\pi Dt\right)^{-\frac{1}{2}}\right\} \qquad \textbf{(22)}$$

where R is the bubble radius; D is the diffusion coefficient; S is the surface tension of mercury; p_∞ is the liquid pressure around the bubble; c_∞ is the gas concentration in the

Table 4 Inert Gas Diffusion Coefficients in Mercury

T (K)	Inert Gas Type				
	Helium	Neon	Argon	Krypton	Xenon
300.0	4.6431×10^{-9}	2.0272×10^{-9}	1.4687×10^{-9}	1.2851×10^{-9}	1.0987×10^{-9}
350.0	5.4169×10^{-9}	2.3651×10^{-9}	1.7135×10^{-9}	1.4993×10^{-9}	1.2819×10^{-9}
400.0	6.1907×10^{-9}	2.7030×10^{-9}	1.9583×10^{-9}	1.7135×10^{-9}	1.4650×10^{-9}
450.0	6.9646×10^{-9}	3.0409×10^{-9}	2.2031×10^{-9}	1.9276×10^{-9}	1.6481×10^{-9}
500.0	7.7384×10^{-9}	3.3787×10^{-9}	2.4479×10^{-9}	2.1419×10^{-9}	1.8312×10^{-9}
550.0	8.5122×10^{-9}	3.7166×10^{-9}	2.6927×10^{-9}	2.3561×10^{-9}	2.0144×10^{-9}
600.0	9.2861×10^{-9}	4.0545×10^{-9}	2.9374×10^{9}	2.5703×10^{-9}	2.1975×10^{-9}

infinity; c_s is the concentration at the bubble wall; ρ_g is gas density in the bubble at the gas pressure of p_∞.

Though the above model is shown to be valid in estimating the bubble growth rates [Epstein and Plesset, 1950], it does not include the effect of increased gas concentration in the ambient mercury due to the inert gas bubble dissolution. This effect is taken into account in this thesis by adding an ordinary differential equation to the above bubble growth model to form a system of ordinary differential equations. The first-order approximation for updating the liquid concentration can be incorporated to update the ambient gas concentration at each time step.

$$\frac{dc_\infty}{dt} = -\rho_g \frac{d\alpha}{dt} = -\rho_g \left(n_b 4\pi R^2 \frac{dR}{dt} \right) \qquad \textbf{(23)}$$

The system is solved using classical 4th-order Runge-Kutta method to obatin accurate numerical results.

The numerical analysis assumes that mercury is initially saturated with cover gas at the pressure of 3 bars. The initial concentration and the concentration at the bubble wall are evaluated using Henry's law. Two different cases are studied. The first one assumes the bubbles are initially of the same radius of 15μm. In the second case the

bubble size distribution is modeled by using two groups of bubbles with different radii (10μm and 15μm) of same number density. Time-dependent bubble radii and concentration are numerically computed.

4.4 Numerical Analysis of One-Size Group of Bubbles

If one assumes the gas bubbles are of the same size of radius R_b, the initial bubble number density in mercury is

$$n_b = \alpha / \left(\frac{4}{3} \pi R_b^{\,3} \right) \qquad \textbf{(24)}$$

For 30μm-diameter and 0.5%-volume fraction, the bubble number density is about $3.537 \times 10^{11} m^{-3}$. This indicates the bubbles are widely separated from others at a distance of more than 100 times the radius value.

Figure 21 shows all the helium bubbles dissolve completely in about 0.62 seconds after being injected into mercury. This is because of the higher pressure at the bubble wall than in mercury due to the surface tension effect. For 15μm-radius helium bubble, the pressure difference between gas and liquid (p_g-p_l) is about 1/2 bar according to the Laplace equation. The bulk concentration in mercury increases as a result of bubble dissolution, but the concentration is still lower than at the bubble wall, which causes the bubbles to dissolve completely.

In contrast, the xenon bubbles show a totally different behavior as shown in Figure 22. Since xenon has much smaller solubility in mercury than helium, slightly dissolving the gas bubble increases the gas concentration in mercury sufficiently to stop bubble dissolution. The bubble radius remains at approximately the initial value. Xenon is more viable than helium to form a specified volume fraction.

4.5 Numerical Analysis of Two-Size Group of Bubbles

A realistic study on bubble behavior must include the effect of bubble size distribution. Any method for bubble injection cannot create a perfectly mono-sized bubble population. Bubbles of different size will have different behaviors due to different internal pressures, causing different concentrations at the bubble wall. For simplicity, a

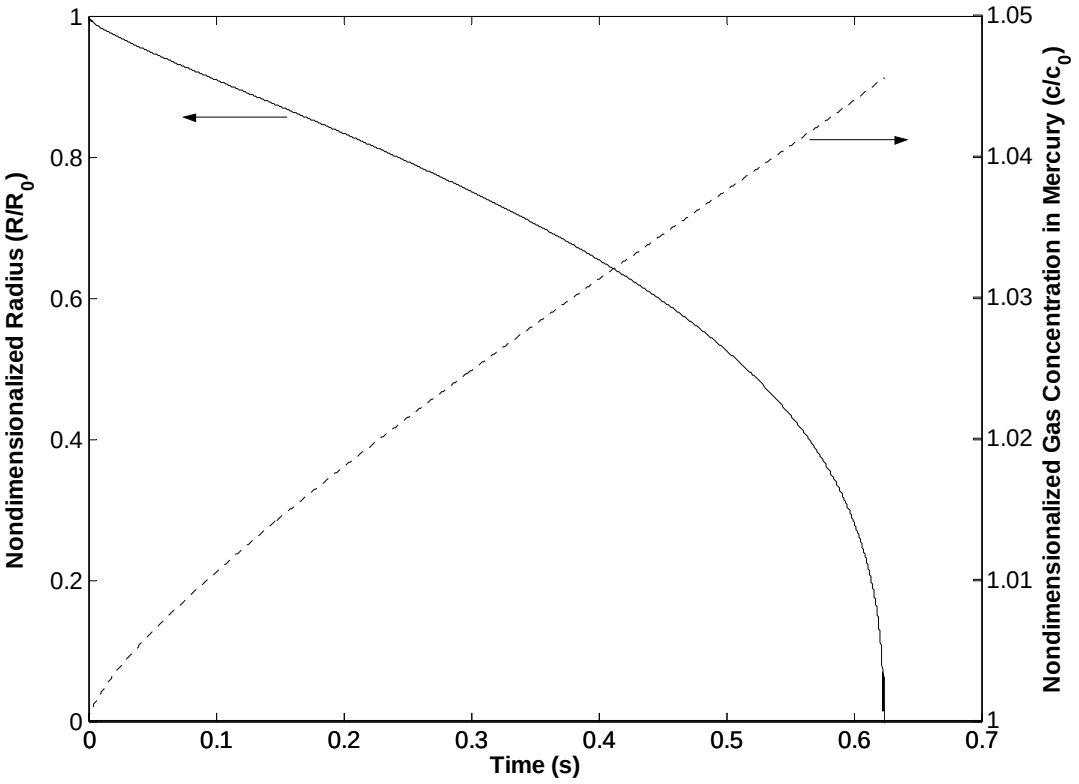

Figure 21 Time-Dependent Helium Bubble Radius and Gas Concentration in Mercury

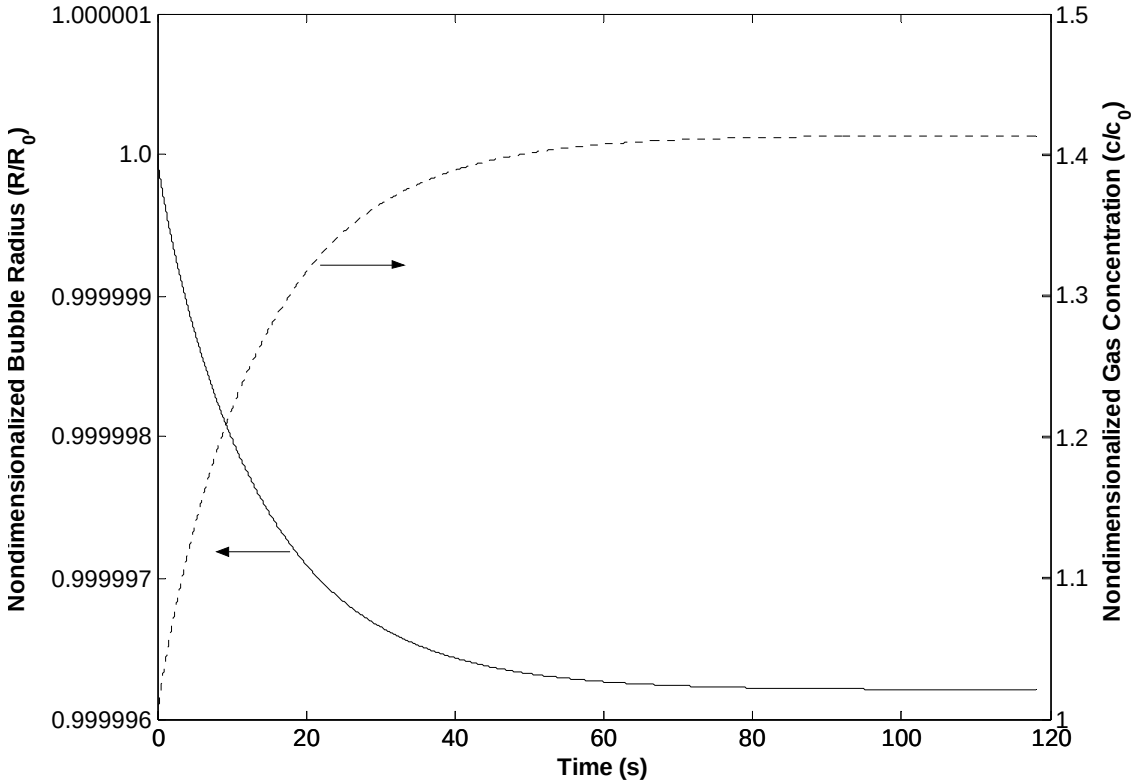

Figure 22 Time-Dependent Xenon Bubble Radius and Gas Concentration in Mercury

two-group distribution is included here to display the physical behavior. The study begins by introducing bubbles with two sizes (10μm & 15μm in radius) and equal number densities. The dominant effect is still the mass diffusion across the bubble wall and the concentration in the ambient mercury is updated at each time step.

Due to the relatively high solubility in mercury, helium bubbles still dissolve away in the two-group situation. Numerical results show the small bubbles dissolve in less than 0.2s, while the larger bubbles shrink to zero radii in about 0.64s, which contributes to the continuously increasing concentration displayed in Figure 23. Xenon has much smaller solubility in mercury, so little gas from the bubbles dissolves. However, smaller bubbles move gas into the mercury that later devovles into the larger bubbles, causing smaller bubbles to get smaller, and bigger bubbles to get bigger, as shown in Figure 24. The concentration stops increasing after a steep initial change. This indicates the number of gas atoms diffusing out of the bubbles of the smaller group is nearly equal to the number of xenon gas atoms entering the bigger group, which has little net effect on the gas concentration in the ambient liquid.

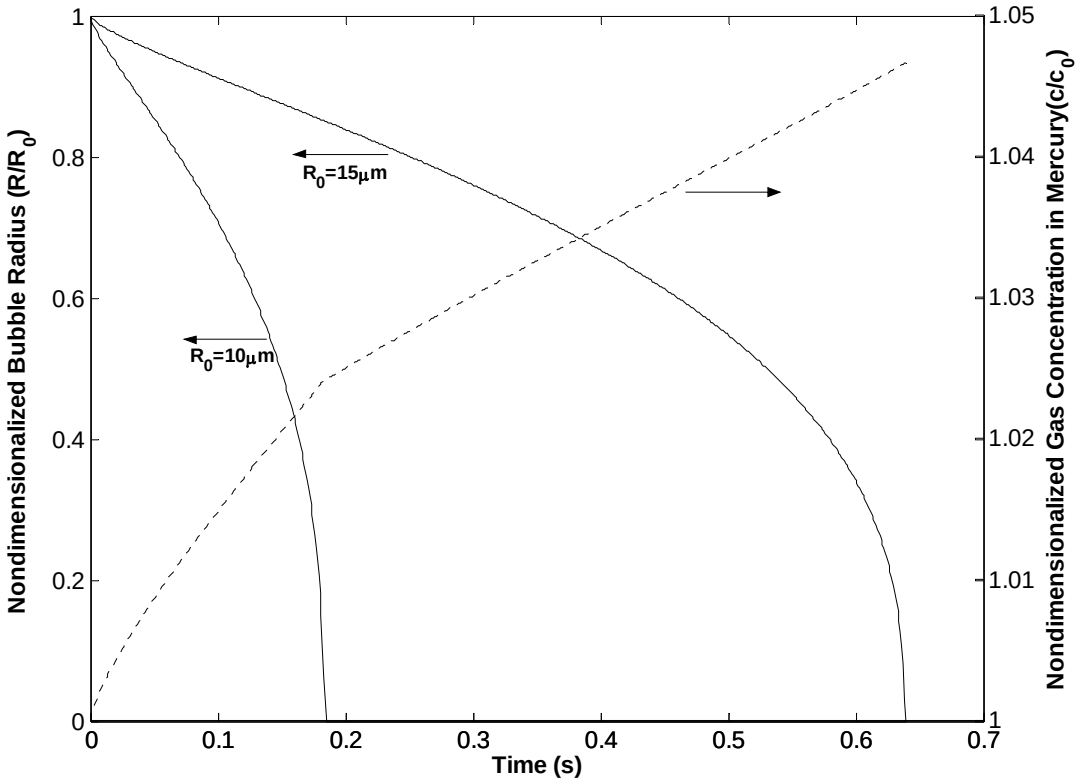

Figure 23 Time-Dependent Helium Bubble Radii and Gas Concentration in Mercury

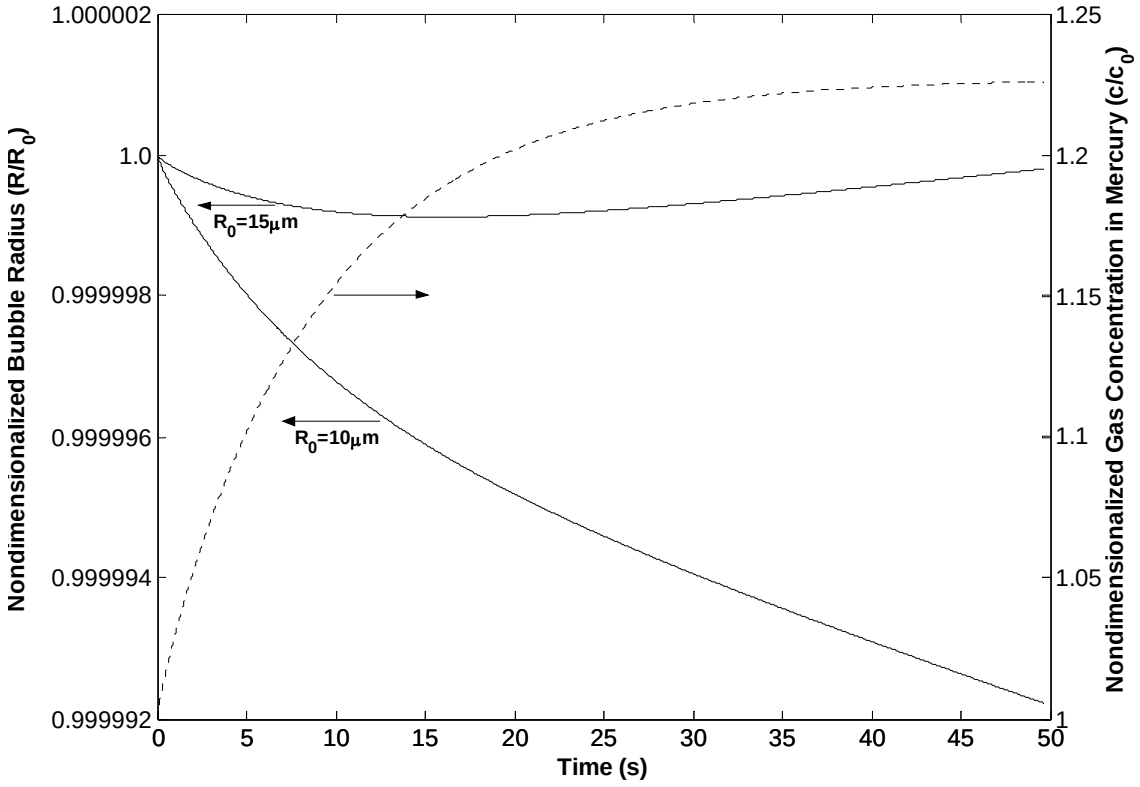

Figure 24 Time-Dependent Xenon Bubble Radii and Gas Concentration in Mercury

5 Conclusions and Suggestions for Future Work

Inert gas solubility and bubble behavior in mercury is intensively investigated in this thesis project. The bubbly dynamics is coupled with acoustic simulation to produce numerical simulation of acoustic streaming in helium-mercury two-phase flow. The results indicate that the acoustic streaming velocities are small relative to convective velocities expected in the SNS target flow.

Inert gas solubility and gas diffusivity in mercury are theoretically evaluated. These are used to simulate inert helium and xenon gas bubble behavior under the effect of mass diffusion across the bubble wall. The simulations show the injected helium bubbles are dissolved if mercury is not oversaturated initially with helium. The case of xenon bubble injection results in a more stable bubble population due to the much smaller solubility than helium in mercury.

The process of computing theoretical inert gas solubility in mercury indicates the results are very sensitive to the chosen inert gas and mercury property values, which have a range of values reported in the literature. Well-designed solubility measurement experiments shall be performed to validate the computed results. This will allow the simulation results of bubble behavior to be of higher confidence.

More accurate bubble behavior simulation must include the effect of bubble size distribution, since any method of bubble formation method will not create a bubble mono-sized distribution. The bubble size distribution function from actual bubble injection experiments will be included for bubble behavior simulation, combined with inert gas solubility measurement results.

List of References

List of References

Arakawa, M., and Kawahashi, M., *Nonlinear Phenomena Induced by Finite-Amplitude Oscillation of Air Column in a Closed Duct (Analysis of Acoustic Streaming)*, JSME International Journal, Vol.39, No.2, pp.280-286, 1996

Barlow, E. and Langlois, W., *Diffusion of Gas from a Liquid into an Expanding Bubble*, IBM Journal of Research and Development, Vol.6, No.3, pp.329-337, 1962

Barton, A., ed., *CRC Handbook of Solubility Parameters and Other Cohesion Parameters*, CRC Press, Boca Baton, FL, 1991

Colonius, T., d'Auria, F. and Brennen C., *Acoustic Saturation in Bubbly Cavitating Flow Adjacent to an Oscillating wall*, Physics of Fluids, Vol.12, No.11, pp.2752-2761, 2000

Delhaye, J., Giot, M. and Riethmuller, M., *Thermohydraulics of Two-Phase Systems for Industrial Design and Nuclear Engineering*, Hemisphere Publishing, 1981

Deymier, P., et al., *Streaming and Removal Forces due to Second-Order Sound Field during Megasonic Cleaning of Silicon Wafers*, Journal of Applied Physics, Vol.88, No.11, pp.6821-6835, 2000

Drew, D., *Mathematical Modeling of Two-Phase Flow*, Annual Review Fluid. Mechanics, vol.15, pp.261-291, 1983

Epstein, P. and Plesset, M, *On the Stability of Gas Bubbles in Liquid-Gas Solutions,* Journal of Chemical Physics, Vol.18, No.11, pp.1505-1509, 1950

Farouk, B. and Aktas, M., *Simulation of Acoustic Streaming in a Resonator*, Proceeding of HT-FED04, ASME Heat Transfer/Fluids Engineering Division Summer Conference, Charlotte, North Carolina, USA, July 11-15, 2004

Fowler, R. and Guggenheim, E., *Statistical Thermodynamics*, Cambridge University Press, Cambridge, 1949

Fukase, S. and Satoh, T., *Solubilities of Rare Gases in Liquid Sodium and Farber's Formula for Vacancy Formation Energies*, Journal of Physics F: Metal Physics, Vol.6, No.7, pp.1233-1242, 1976

Ishii, M., *Thermo-Fluid Dynamic Theory of Two-Phase Flow*, Eyrolles, Paris, 1975

Ishii, M. & Kocamystafafaogullari, G., *Two-Phase Flow Models and Their Limitations*, Proceedings of the NATO Advanced Research Workshop on the Advances in Two-Phase Flow and Heat Transfer, Spitzingsee, FRG, Aug.31-Sept.3, 1982

Lamb, H., *Hydrodynamics*, Cambridge University Press, Cambridge, 1932

Lide, D., ed., *CRC Handbook of Chemistry and Physics*, CRC Press, Boca Baton, FL, 2005

Manley, D., *Change of Size of Air Bubbles in Water Containing a Small Dissolved Air Content*, British Journal of Applied Physics, Vol.11, pp.38-42, 1960

Neff, R. and McQuarrie, D, *A Statistical Mechanical Theory of Solubility*, Journal of Physical Chemistry, Vol.77, No.3, pp.413-417, 1973

Nyborg, W., *Acoustic Streaming*, in Physical Acoustics, edited by Mason, W., Vol. II, Part. B, pp. 265-331, Academic Press, New York, 1965

Nyborg, W., *Acoustic Streaming*, in *Nonlinear Acoustic,* edited by M. F. Hamilton, pp. 207-231, Academic Press, San Diego, CA 1998

Perry, R and Green, D., ed., *Perry's Chemical Engineers' Handbook (7th Edition),* McGraw-Hill, New York, 1997

Pierotti, R., *Aqueous Solutions of Nonpolar Gases*, Journal of Physical Chemistry, Vol.69, No.1, pp.281-288, 1965

Pointer, D., *Experimental Characterization of Flow Patterns and Flow Stability in the Bulk Mercury Flow Field of the Spallation Neutron Source Mercury Target*, Ph.D Dissertation, University of Tennessee, 2001

Preston, A., Colonius, T. and Brennen, C., *A Numerical Investigation of Unsteady Bubbly Cavitating Nozzle Flows,* Proceedings of ASME FEDSM00, ASME 2000 Fluids Engineering Division Summer Meeting, Boston, MA, USA, Jun.11-15, 2000

Radzig, A. and Smirnov, B, *Reference Data on Atoms, Molecules, and Ions*, Springer, Berlin, 1985

Rayleigh, L., *On the Circulation of Air Observed in Kundt's Tubes and on Some Allied Acoustical Problems*, Philosophical Transactions of the Royal Society of London, Vol.175, pp.1-21, 1884

Riemer, B, Soltner, H. and Wendel, M., *Bubble Injection Experiments on the Target Test*

Facility, Presentation, Spallation Neutron Source, Oak Ridge National Laboratory, June 3, 2004,

Riley, N., *Steady Streaming*, Annual Review of Fluid Mechanics, Vol. 33, pp. 43-65, 2001

Ruggels, A., *The Propagation of Pressure Perturbations in Bubbly Air/Water Flows,* Ph.D Dissertation, Rensselaer Polytechnic Institute, 1987

Sawyer, A. and Ruggles, A., *Fluid Transients in Mercury and Mercury-Helium Flows*, Proceeding of HT-FED04, ASME Heat Transfer/Fluids Engineering Division Summer Conference, Charlotte, North Carolina, USA, July 11-15, 2004

Scriven, L., *On the Dynamics of Phase Growth*, Chemical Engineering Science, Vol.10, Issue.1-2, pp.1-13, 1959

Shoor, S. and Gubbins, K., *Solubility of Nonpolar Gases in Concentrated Electrolyte Solutions*, Journal of Physical Chemistry, Vol.73, No.3, pp.498-504, 1969

Shpil'rain, E., Skovorod'ko, S. and Mozgovoi, A. *The Solubility of Inert Gases in Liquid-Metal Heat-Transfer Agents*, High Temperature, Vol.38, No.3, pp.384-388, 2000

Thormeier, K, *Solubility of the Noble Gases in Liquid Sodium*, Nuclear Engineering and Design, Vol.14, pp.69-82, 1970

van Wijngaarden, L., *One-Dimensional Flow of Liquids Containing Small Gas Bubbles*, Annual Review of Fluid Mechanics, Vol.4, pp.369-396, 1972

Wan, Q., and Kuznetsov, A., *Numerical Efficiency Comparison of Heat Transfer Enhancement between Two Parallel Beams by Acoustic Streaming Induced by Standing and Traveling Waves*, Proceeding of NSF–ECS/EPSCoR National Grantees Conference on Electronics, Photonics, and Device Technologies (EPDT), University of Arkansas, Fayetteville, AR, USA, August 16-17, 2001

Wu, J., and Du, G., *Streaming Generated by a Bubble in an Ultrasound Field*, Journal of the Acoustical Society of America, Vol.101, No.4, pp.1899-1907, 1997

Yarin, A., *Stationary D.C. Streaming due to Shape Oscillations of a Droplet and its Effect on Mass Transfer in Liquid-Liquid Systems*, Journal of Fluid Mechanics, Vol.44, pp.321-342, 2001

Yano, T., *Turbulent Acoustic Streaming Excited by Resonant Gas Oscillation with*

Periodic Shock Waves in a Closed Tube, Journal of the Acoustic Society of America, Vol.106, No.1, pp.7-12, 1999

Appendices

Appendix A

1-D Acoustic Streaming Simulation Code in FORTRAN 90 for Helium-Mercury Two-Phase Bubbly Flow

```
!**********************************************************************
! Finally, the new method of Colonius (1998) is implemented in a more simple way only using Implicit
Euler method, instead of Richardson Extrapolation method. Thanks goes to Dr. Al Preston for providing me
his old codes. Some of the subroutines here are modified from them.
!**********************************************************************
MODULE consts
  ! Define constants here for use
  IMPLICIT NONE
  REAL(8),PARAMETER::P0=3.0D5,R0=15.0D-6,Rho_Hg=13.546D3,S=425.41D-3
  REAL(8),PARAMETER::Mu_Hg=1.552D-3,Nu_Hg=Mu_Hg/Rho_Hg,Beta0=0.5D-2
  REAL(8),PARAMETER::Pg0=P0+2.D0*S/R0,C2=Pg0*R0**3,C3=Rho_Hg*R0**3*(1-Beta0)/Beta0
  REAL(8),PARAMETER::C4=R0**3*(1-Beta0)/Beta0,xmax=0.2D0,eps=1.D-10
END MODULE consts

!**********************************************************************
MODULE vars
  ! Define variables
  IMPLICIT NONE
  REAL(8)::delt,A,fn,pi,emax1,emax2
  ! emax1 for r & emax2 for x
  REAL(8),ALLOCATABLE::rn(:),ro(:),xn(:),xo(:),pn(:),po(:),un(:),uo(:),vn(:),vo(:)
  REAL(8),ALLOCATABLE::jac(:,:),pw(:),f(:),jacl(:,:),det(:),f1(:),rhs(:),pr(:)
  INTEGER,ALLOCATABLE::indx(:)
  INTEGER::nx,nt,l,i,j,k,TN1,TN2

END MODULE vars

!**********************************************************************
MODULE some_functions
! Define some functions here

CONTAINS
!**********************************************************************
  FUNCTION rhof(r)
  ! Calculate the density of corresponding radius value

    USE consts
    IMPLICIT NONE
    REAL(8)::rhof,r
    rhof=C3/(C4+r**3)

  END FUNCTION rhof

!**********************************************************************
  FUNCTION drhof(r)
  ! Calculate d(rho)/d(r) for Jacobian Matrix
```

```
   USE consts
   IMPLICIT NONE
   REAL(8)::r,drhof

   drhof=-3.D0*(rhof(r)*r)**2.D0/C3
  END FUNCTION drhof

!***********************************************************************
  FUNCTION pf(rn,ro,vn,vo,delt)
  ! calculate the pressure

   USE consts
   IMPLICIT NONE
   REAL(8)::rn,ro,vn,vo,delt,pf

   pf=(rn*vn-ro*vo)/delt+0.5D0*vn*vn+4.D0*Nu_Hg*vn/rn
   pf=pf*Rho_Hg+2.D0*S/rn
   pf=C2/rn**3.D0-pf
!    IF (pf.lt.0.D0) THEN
!      PRINT*,'p= ',pf
!    END IF
  END FUNCTION pf

!***********************************************************************
  FUNCTION dpf(rn,ro,vn,vo,delt)
  ! Calculate dp/dr for Jacobian Matrix

  USE consts
  IMPLICIT NONE
  REAL(8)::rn,ro,vn,vo,delt,dpf

  dpf=(vn+rn/delt)/delt+vn/delt-4.D0*Nu_Hg*vn/rn**2.D0+4.D0*Nu_Hg/rn/delt
  dpf=-dpf*Rho_Hg-3.D0*C2/rn**4.D0+2.D0*S/rn**2.D0
  END FUNCTION dpf

!***********************************************************************
END MODULE some_functions

!***********************************************************************
!***********************************************************************
PROGRAM AC1D
  USE consts
  USE vars
  USE some_functions

  IMPLICIT NONE
  !REAL(8)::Omega0
  Real(8)::sv(3000)
  do i=1,3000
    sv(i)=0.D0
  end do
  PRINT*,'Input the number of spatial mesh steps:'
  READ*,nx
```

```
  PRINT*,'Input the number of time mesh steps:'
  READ*,nt

  PRINT*,'Input the number of periods to compute:'
  print*,'TN1:'
  READ*,TN1
  print*,'TN2:'
  read*,TN2
  !TN1=1401
  !TN2=1500


! Omega0=sqrt(3.D0*P0/Rho_Hg)/R0

  fn=1.D4
  A=0.4D0
  k=1
  l=0 ! Count the number of vibrating cycles used
  pi=dacos(-1.D0)
  delt=1.D0/fn/(nt-1)

  ! roots only include x(2)-x(nx) and r(1)-r(nx), totally 2*nx-2
  ! Solution: r1,x2,r2,x3,...,r(nx-2),x(nx-1),r(nx-1),r(nx)
  ALLOCATE (rn(nx))
  ALLOCATE (ro(nx))
  ALLOCATE (xn(nx))
  ALLOCATE (xo(nx))
  ALLOCATE (pn(nx))
  ALLOCATE (po(nx))
  ALLOCATE (un(nx))
  ALLOCATE (uo(nx))
  ALLOCATE (vn(nx))
  ALLOCATE (vo(nx))

  ALLOCATE (f(2*nx-2))
  ALLOCATE (f1(2*nx-2))
  ALLOCATE (indx(2*nx-2))
  ALLOCATE (det(2*nx-2))
  ALLOCATE (rhs(2*nx-2))

  ALLOCATE (jac(2*nx-2,5))
  ALLOCATE (jacl(2*nx-2,2))

  ALLOCATE (pw(nt))
  ALLOCATE (pr(nt))

  ! Set up initial field
  DO i=1,nx
    po(i)=P0
    ro(i)=R0
    xo(i)=xmax*(i-1)/(nx-1)
    uo(i)=0.D0
    vo(i)=0.D0
```

```
   pn(i)=po(i)
   xn(i)=xo(i)
   rn(i)=ro(i)
   vn(i)=vo(i)
   un(i)=uo(i)
  END DO
  pw(1)=P0
  CALL output2(1,pw(1))
  pr(1)=P0
  CALL output3(1,pr(1))

! Compute field in enough periods till it stops changing,
DO WHILE(k.gt.0)
  l=l+1
  if (l.eq.TN1) then
    sv=sv+uo
  end if
  IF (MOD(l,TN2).eq.0) THEN
    CALL output(nx,po,ro,xo,uo)
  END IF
  DO i=2,nt
    ! boundary conditions
    un(1)=A*dsin(2.D0*pi*(i-1)/(nt-1))
    xn(1)=xo(1)+delt*un(1)
  DO WHILE(.true.)
    ! get the function values of the current field
      DO j=1,nx-1
        f(2*j-1)=0.5D0*(xn(j+1)-xn(j))*(rhof(rn(j))+rhof(rn(j+1)))- &
          0.5D0*(xo(j+1)-xo(j))*(rhof(ro(j))+rhof(ro(j+1)))
        f(2*j)=0.5D0*(xn(j+1)-xn(j))*(rhof(rn(j))*un(j)+ &
          rhof(rn(j+1))*un(j+1))-(xo(j+1)-xo(j))/2.D0* &
          (rhof(ro(j))*uo(j)+rhof(ro(j+1))*uo(j+1))- &
          delt*(pf(rn(j),ro(j),vn(j),vo(j),delt)- &
          pf(rn(j+1),ro(j+1),vn(j+1),vo(j+1),delt))
      END DO
      rhs=-f
      ! calculate the Jacobian Matrix first
      CALL compute_jacobian(jac,xn,xo,rn,ro,vn,vo,nx,delt,rhs)

      ! find the solution for the root correction
      CALL bandec(jac,2*nx-2,2,2,2*nx-2,5,jacl,2,indx,det)
      CALL banbks(jac,2*nx-2,2,2,2*nx-2,5,jacl,2,indx,rhs)

      ! update the field of x,r,v,u
      DO j=1,nx-2
        rn(j)=rn(j)+rhs(2*j-1)
        xn(j+1)=xn(j+1)+rhs(2*j)
      END DO
      rn(nx-1)=rn(nx-1)+rhs(2*nx-3)
      rn(nx)=rn(nx)+rhs(2*nx-2)

      DO j=2,nx-1
        un(j)=(xn(j)-xo(j))/delt
        vn(j)=(rn(j)-ro(j))/delt
```

```
      END DO
      vn(1)=(rn(1)-ro(1))/delt
      vn(nx)=(rn(nx)-ro(nx))/delt
      un(nx)=0.D0
      xn(nx)=xmax

      DO j=1,nx
        pn(j)=pf(rn(j),ro(j),vn(j),vo(j),delt)
      END DO
      DO j=1,nx-1
        f1(2*j-1)=0.5D0*(xn(j+1)-xn(j))*(rhof(rn(j))+rhof(rn(j+1)))- &
          0.5D0*(xo(j+1)-xo(j))*(rhof(ro(j))+rhof(ro(j+1)))
        f1(2*j)=0.5D0*(xn(j+1)-xn(j))*(rhof(rn(j))*un(j)+ &
          rhof(rn(j+1))*un(j+1))-(xo(j+1)-xo(j))/2.D0* &
          (rhof(ro(j))*uo(j)+rhof(ro(j+1))*uo(j+1))- &
          delt*(pf(rn(j),ro(j),vn(j),vo(j),delt)- &
          pf(rn(j+1),ro(j+1),vn(j+1),vo(j+1),delt))
      END DO
      emax1=0.D0 ! for variable values
      emax2=0.D0 ! for function values

      ! check for convergence
      DO j=1,2*nx-2
        emax1=MAX(DABS(rhs(j)),emax1)
        emax2=MAX(DABS(f1(j)-f(j)),emax2)
      END DO
!     PRINT*,l,i,emax1,emax2
      IF ((emax1.lt.eps).or.(emax2.lt.eps)) THEN
!       PRINT*,i,' EXIT'
!       DO j=1,nx
!         IF (pn(j)<0.d0) THEN
!           PRINT*,i,j,pn(j)
!           STOP
!         END IF
!       END DO
        EXIT
      ENDif
    END DO ! inner while loop

    IF (MOD(l,TN2).eq.0) THEN
      CALL output(nx,pn,rn,xn,un)
    END IF
    if (l.ge.TN1) then
      sv=sv+un
    end if

    ! save the pressure history near wall
    pw(i)=pn(1)
    pr(i)=pn(nx)

    ! update the old field for the next time step
    xo(1:nx)=xn(1:nx)
    uo(1:nx)=un(1:nx)
    po(1:nx)=pn(1:nx)
```

```
    vo(1:nx)=vn(1:nx)
    ro(1:nx)=rn(1:nx)
  END DO ! time step stops here
  CALL output2(nt-1,pw(2:nt))
  CALL output3(nt-1,pr(2:nt))
  pw(1)=pw(nt)

  PRINT*,'No',l,' period of calculation completed!'
  PRINT*,sv(1)

  IF (MOD(l,TN2).eq.0) THEN
    PRINT*,'to stop, input 0; to go on, input 1:'
    !READ*,k
    k=0
  END IF
END DO ! outer WHILE loop
sv=sv/((TN2-TN1+1.D0)*5000.D0-TN2+TN1)
open(1,file='sv.dat',form='formatted',status='unknown')
write(1,'(e14.6)'),(sv(i),i=1,3000)
close(1)

stop
END

!**********************************************************************
SUBROUTINE compute_jacobian(jac,xn,xo,rn,ro,vn,vo,nx,delt,f)
! Calculates the Jacobian matrix for Newton's Method
  USE consts
  USE some_functions

  IMPLICIT NONE
  INTEGER::nx,i,j
  REAL(8),DIMENSION(2*nx-2,5)::jac
  REAL(8),DIMENSION(nx)::rn,xn,ro,xo,vn,vo
  REAL(8)::delt,f(2*nx-2),maxjac

  DO i=1,nx-1
    IF(i.eq.1) THEN
    ! d/dr1,d/dx2,d/dr2
      jac(2*i-1,1)=0.D0
      jac(2*i-1,2)=0.D0
      jac(2*i-1,3)=0.5D0*(xn(i+1)-xn(i))*drhof(rn(i))
      jac(2*i-1,4)=0.5D0*(rhof(rn(i))+rhof(rn(i+1)))
      jac(2*i-1,5)=0.5D0*(xn(i+1)-xn(i))*drhof(rn(i+1))

      jac(2*i,1)=0.D0
      jac(2*i,2)=0.5D0*(xn(i+1)-xn(i))*(xn(i)-xo(i))/delt*drhof(rn(i))- &
        delt*dpf(rn(i),ro(i),vn(i),vo(i),delt)
      jac(2*i,3)=0.5*(rhof(rn(i))*(xn(i)-xo(i))/delt+ &
        rhof(rn(i+1))*(xn(i+1)-xo(i+1))/delt)+ &
        0.5D0*(xn(i+1)-xn(i))*rhof(rn(i+1))/delt
      jac(2*i,4)=0.5D0*(xn(i+1)-xn(i))*(xn(i+1)-xo(i+1))/delt*drhof(rn(i+1))+ &
        delt*dpf(rn(i+1),ro(i+1),vn(i+1),vo(i+1),delt)
      jac(2*i,5)=0.D0
```

```
   ELSEIF(i.eq.(nx-1)) THEN
   ! d/dx(nx-1),d/dr(nx-1),d/dr(nx)
    jac(2*i-1,1)=0.D0
    jac(2*i-1,2)=-0.5D0*(rhof(rn(i))+rhof(rn(i+1)))
    jac(2*i-1,3)=0.5D0*(xn(i+1)-xn(i))*drhof(rn(i))
    jac(2*i-1,4)=0.5D0*(xn(i+1)-xn(i))*drhof(rn(i+1))
    jac(2*i-1,5)=0.D0

    jac(2*i,1)=-0.5D0*(rhof(rn(i))*(xn(i)-xo(i))/delt+ &
      rhof(rn(i+1))*(xn(i+1)-xo(i+1))/delt)+ &
      0.5D0*(xn(i+1)-xn(i))*rhof(rn(i))/delt
    jac(2*i,2)=0.5D0*(xn(i+1)-xn(i))*drhof(rn(i))*(xn(i)-xo(i))/delt- &
      delt*dpf(rn(i),ro(i),vn(i),vo(i),delt)
    jac(2*i,3)=0.5D0*(xn(i+1)-xn(i))*drhof(rn(i+1))*(xn(i+1)-xo(i+1))/delt+ &
      delt*dpf(rn(i+1),ro(i+1),vn(i+1),vo(i+1),delt)
    jac(2*i,4)=0.D0
    jac(2*i,5)=0.D0
   ELSE
   ! d/dx(i),d/dr(i),d/dx(i+1),d/dr(i+1)
    jac(2*i-1,1)=0.D0
    jac(2*i-1,2)=-0.5D0*(rhof(rn(i))+rhof(rn(i+1)))
    jac(2*i-1,3)=0.5D0*(xn(i+1)-xn(i))*drhof(rn(i))
    jac(2*i-1,4)=0.5D0*(rhof(rn(i))+rhof(rn(i+1)))
    jac(2*i-1,5)=0.5D0*(xn(i+1)-xn(i))*drhof(rn(i+1))

    jac(2*i,1)=-0.5D0*(rhof(rn(i))*(xn(i)-xo(i))/delt+ &
      rhof(rn(i+1))*(xn(i+1)-xo(i+1))/delt)+ &
      0.5D0*(xn(i+1)-xn(i))*rhof(rn(i))/delt
    jac(2*i,2)=(xn(i+1)-xn(i))/2.D0*(xn(i)-xo(i))/delt*drhof(rn(i))- &
      delt*dpf(rn(i),ro(i),vn(i),vo(i),delt)
    jac(2*i,3)=0.5D0*(rhof(rn(i))*(xn(i)-xo(i))/delt+ &
      rhof(rn(i+1))*(xn(i+1)-xo(i+1))/delt)+ &
      0.5D0*(xn(i+1)-xn(i))*rhof(rn(i+1))/delt
    jac(2*i,4)=(xn(i+1)-xn(i))/2.D0*(xn(i+1)-xo(i+1))/delt*drhof(rn(i+1))+ &
      delt*dpf(rn(i+1),ro(i+1),vn(i+1),vo(i+1),delt)
    jac(2*i,5)=0.D0
   END IF
  END DO
  DO j=1,2*nx-2
   maxjac=0.D0
   DO i=1,5
    maxjac=MAX(DABS(jac(j,i)),maxjac)
   END DO
   jac(j,1:5)=jac(j,1:5)/maxjac
   f(j)=f(j)/maxjac
  END DO
END SUBROUTINE compute_jacobian

!**********************************************************************
SUBROUTINE output(nx,pn,rn,xn,un)
! Output the data

  IMPLICIT NONE
  INTEGER::i,nx
```

```
  REAL(8)::pn(nx),rn(nx),xn(nx),un(nx)

  open(1,file='p.dat',form='formatted',status='unknown',position='append')
  open(2,file='r.dat',form='formatted',status='unknown',position='append')
  open(3,file='x.dat',form='formatted',status='unknown',position='append')
  open(4,file='u.dat',form='formatted',status='unknown',position='append')
  WRITE(1,9010),(pn(i),i=1,nx)
  WRITE(2,9010),(rn(i),i=1,nx)
  WRITE(3,9010),(xn(i),i=1,nx)
  WRITE(4,9010),(un(i),i=1,nx)
  9010 format(10000e14.6)

  CLOSE(1)
  CLOSE(2)
  CLOSE(3)
  CLOSE(4)
END SUBROUTINE output

!**********************************************************************
SUBROUTINE output2(nt,pw)
! Output the pressure history at wall

  IMPLICIT NONE
  INTEGER nt,i
  REAL(8)::pw(nt)

  open(5,file='pw.dat',form='formatted',status='unknown',position='append')
  DO i=1,nt
     WRITE(5,'(e14.7)'),pw(i)
  END DO
  CLOSE(5)
END SUBROUTINE output2

!**********************************************************************
SUBROUTINE output3(nt,pr)
! Output the pressure history at wall

  IMPLICIT NONE
  INTEGER nt,i
  REAL(8)::pr(nt)

  open(5,file='pr.dat',form='formatted',status='unknown',position='append')
  DO i=1,nt
     WRITE(5,'(e14.7)'),pr(i)
  END DO
  CLOSE(5)
END SUBROUTINE output3

!**********************************************************************
SUBROUTINE bandec(a,n,m1,m2,np,mp,al,mpl,indx,d)
       IMPLICIT NONE
       INTEGER, INTENT(IN) :: n,m1,m2,np,mp,mpl
       REAL(8), DIMENSION(np,mp), INTENT(INOUT) :: a
       REAL(8), DIMENSION(np,mpl), INTENT(OUT) :: al
```

```
      INTEGER, DIMENSION(n), INTENT(OUT) :: indx
      REAL(8), INTENT(OUT) :: d
      REAL(8), PARAMETER :: TINY=1.0D-20
      INTEGER :: i,j,k,l,mm
      REAL(8) :: dum

      mm = m1 + m2 + 1
      IF(mm > mp .OR. m1 > mpl .OR. n > np ) PAUSE 'bad args in bandec'
      l = m1
      DO i = 1,m1
          a(i,m1+2-i-l:mm-l) = a(i,m1+2-i:mm)
          l = l-1
          a(i,mm-l:mm) = 0.D0
      END DO
      d = 1.D0
      l = m1

      DO k = 1,n
          dum = a(k,1)
          i = k
          IF ( l < n ) l = l+1

          DO j = k+1,l
              IF ( ABS( a(j,1) ) > ABS(dum) ) THEN
             dum = a(j,1)
             i = j
              END IF
          END DO
          indx(k) = i
          IF (dum == 0.D0) THEN
              WRITE(*,*) 'Matrix is singular, using TINY pivot'
              a(k,1) = TINY
          END IF
          IF (i .NE. k) THEN
              d = -d
              DO j = 1,mm
             dum = a(k,j)
             a(k,j) = a(i,j)
             a(i,j) = dum
              END DO
          END IF
          DO i = k+1,l
              dum = a(i,1) / a(k,1)
              al(k,i-k) = dum
              a(i,1:mm-1) = a(i,2:mm) - dum*a(k,2:mm)
              a(i,mm) = 0.D0
          END DO
      END DO
      RETURN
END SUBROUTINE bandec

!*****************************************************************
SUBROUTINE banbks(a,n,m1,m2,np,mp,al,mpl,indx,b)
      IMPLICIT NONE
```

```
      INTEGER, INTENT(IN) :: n,m1,m2,np,mp,mpl
      REAL(8), DIMENSION(np,mp), INTENT(IN) :: a
      REAL(8), DIMENSION(np,mpl), INTENT(IN) :: al
      INTEGER, DIMENSION(n), INTENT(IN) :: indx
      REAL(8), DIMENSION(n), INTENT(INOUT) :: b
      INTEGER :: i,k,l,mm
      REAL(8) :: dum

      mm = m1+m2+1
      IF(mm > mp .OR. m1 > mpl .OR. n > np ) PAUSE 'bad args in banbks'
      l = m1
      DO k = 1,n
          i = indx(k)
          IF ( i .NE. k ) THEN
              dum = b(k)
              b(k) = b(i)
              b(i) = dum
          END IF
          IF ( l < n ) l = l+1
          b(k+1:l) = b(k+1:l) - al(k,1:l-k)*b(k)
      END DO
      l = 1
      DO i = n,1,-1
          dum = b(i)
          DO k = 2,l ; dum = dum - a(i,k)*b(k+i-1) ; END DO
          b(i) = dum / a(i,1)
          IF( l < mm ) l = l+1
      END DO
      RETURN
END SUBROUTINE banbks
!*******************************************************************
```

Appendix B

MATLAB Script for Inert Gas Solubility Evaluation in Mercury

```
% MATLAB script to compute the solubility of noble gases in mercury
% based on the method by Shpilrain, E.E., et.al, High Temperature, 38(3),
% 407-411, 2000
% sub- & super- script 1 means mercury(solvent)
% sub- & super- script 2 means gas phase(solute)
% output: x21 (molar fraction of component 2 in component 1, dimensionless);
% kh: Henry's law constants; D: gas diffusion coefficient

clear all;
% ************************common constants************************
% universal gas constant (J/mol*K)
R=8.31441;
% electron charge(C)
e=-1.60221892e-19;
% electron mass(kg);
me=9.109534e-31;
% planck's constant (J*s)
h=6.626176e-34;
% Avogadro's Constant (1/mol)
A=6.022045e+23;
% atomic mass unit (kg)
amu=1.6605655e-27;
% Boltzmann's constant (J/K)
k=1.3800662e-23;

% ********************common variable*************************
% system    temperature(K), can use more values
Ts=300.0:50.0:600.0;
% gas cover pressure(bar), can use more values
p2s=1.0e5;
% mercury vapor pressure (pa): it varies with temperature,
% but it's small enough to be neglected. Typical value at room temperture
% adopted (From CRC Handbook of Chemistry and Physics, 85th version,
% pp.6-147, 2004-2005
p12=3.68e-4*1e3;

p2s=p2s-p12;
% ********* MERCURY (1) properties to be used    ************
% From: Lange's Handbook of Chemistry (15th Edition), Table 4.6
% atomic radius and diameter (m)
r1=1.51e-10;
d1=2*r1;

% atomic mass
m1=200.59*amu;
% molar weight/mass (kg/mol)
M1=m1*A;
```

```
% density (kg/m^3)
rho1=13.5336e+3;
% number density (1/m^3)
rho1a=rho1*A/M1;
% molar volume of mercury (m^3/mol)
v1=M1/rho1;

% From CRC Handbook of Chemistry and Physics, 85th version,
% pp.6-134, pp.6-186, pp.10-167, 2004-2005
% thermal expansion coefficient 100^oC (1/K) TEMPERATURE DEPENDENT
alpha=1.81e-4;
% isothermal compressibility 100^oC(/Pa) TEMPERATURE DEPENDENT
beta=4.410e-11;
% dynamic viscosity (Pa*s) at 25^oC
mu1=1.526e-3;
% polarizability of mercury atom (m^3)
alpha1=5.02e-30;
% number of electrons in the outer shell of atoms of mercury
y1=41.76;

% eta in U^in
eta=pi*rho1a*d1^3/6;
% Debye temperature of mercury
deb1=3/4*mu1*h/k*((A/M1)^2/rho1)^(1/3);

% ********************gas(2) properties for calculation***************
% in the list, the sequence is HELIUM, NEON, ARGON, KRYPTON & XENON
% initial values
% r2=[0.49 0.55 0.88 1.03 1.24]*1.0e-10;
% shpilrain
% r2=[1.35 1.60 1.71 1.83 2.03]*1.0e-10;
% thormeier
% r2=[1.30 1.60 1.70 1.86 2.05]*1.0e-10;
% From chemicool.com
r2=[31 71 98 112 131]*1.0e-12;

% From CRC Handbook of Chemistry and Physics, 85th version,
% pp.10-167, 2004-2005;
% polarizability of gass atom (m^3)
alpha2=[0.204956e-30 0.3956e-30 1.6411e-30 2.4844e-30 4.044e-30];

% number of electrons in the outer shell of gas atoms
y2=[1.7 8.8 12.65 20.1 24.3];

% gas atomic mass
m2=[4.002602 20.1797 39.948 83.8 131.29];
% Gas molar mass
m2a=m2*1.0e-3;
% Gas atomic mass
m2=m2*amu;

% use the loop to compute for each pressure and temperature
for i=1:length(Ts)
      T=Ts(i);
```

```
        % gas partial pressure above the solution(Pa)

        % internal presure
        P1=T*alpha/beta;

        % dk=r1+r2 (m)
        dk=r1+r2;

        %********************U^in_12********************%
        q0=R*T*(-log(1-eta)+4.5*(eta/(1-eta))^2)-pi*A*d1^3*P1/6;
        q1=-R*T/d1*(6*eta/(1-eta)+18*(eta/(1-eta))^2)+pi*A*d1^2*P1;
        q2=R*T/d1^2*(12*eta/(1-eta)+18*(eta/(1-eta))^2)-2*pi*A*d1*P1;
        q3=4*pi*A*P1/3;
        Uin=q0+q1*dk+q2*dk.^2+q3*dk.^3;
        %********************U^p_12 ********************%
        C=3*e*h/4/pi/sqrt(me)*alpha1*alpha2;
        C=C./(sqrt(alpha1/y1)+sqrt(alpha2./y2));

        Up=-8*pi*rho1a*A/9*C./dk.^3;
        %********************U^os_12********************%
        % Debye temperature of gases
        deb2=deb1.*sqrt(m1./m2);
        Uos=R*deb2./(exp(deb2/T)-1);

        %********************U12************************%
        U12=Uin+Up+Uos;
        %********************S12                                    entropy********************%
S12=deb2/T./(exp(deb2/T)-1)-log(1-exp(-deb2/T));
        %****x21(molar fraction): output for each pressure and temperature****%
        x21(i,:)=p2s*v1/R/T*exp(-U12/R/T+S12);
        kh(i,:)=1.0e5*v1./(x21(i,:).*m2a);
        D(i,:)=k*T/6/pi/mu1./r2;
end
disp('!!********************compuatation finished*****************************!!');
disp(['Temperature from ',num2str(300),'K ','to ',num2str(600),'K ',...
        'with step of ', num2str(50),'K']);
x21
kh
D
save sol.mat
```

Appendix C

MATLAB Script for One-Group Bubble Growth Rate Simulation

Only the code for helium is attached. For xenon case, see the comment in the script. One only needs to change some values for some constants.

C.1 Master Script

```
clear all;
global D R T S kh cinf nb pinf p0 % define global variables to be used in functions
% temperature at T=300K

% constants
S=425.41e-3; % surface tension of mercury
p0=3.0e5; % cover gas pressure
pinf=3.0e5; % ambient liquid pressure
kh=1/4.4264e6; % Henry's law constant; for xenon, change to xenon's Henry's law constant
D=4.643e-9; % diffusion coefficient; for xenon, change to xenon's diffusion coefficient
alpha=0.005; % initial gas volume fraction
r0=15e-6; % initial gas bubble radius
nb=alpha*3/4/pi/r0^3; % gas bubble number density

pg=pinf+2*S/r0; % gas pressure
rhog=pg*4.0e-3/8.3125/300; % gas density; for xenon, 4.0e-3 should be replace as xenon's molar mass
cinf0=p0*kh; % initial concentration at infinity

f=[@fun1,@fun2]; % function handle vector
ic=[r0,cinf0]; % initial values vector
rk4(f,0,1,ic,8092); % call r-k method
save He1G.mat % save results
plotyy(T,R/R(1),T,cinf/cinf(1)); % plot time-dependent radius and concentration
```

C.2 Fourth Order Runge-Kutta Function

```
function rk4(f,a,b,y0,n)
% 4th order classic Runge-Kutta
%
% input: function handle(f);solution interval([a,b]); initial value(y0); no.of subdivisions(n)
%
% output: y(b)
% When the R becomes less than zero or stops changing,
% the computation is stopped
global R T S cinf nb
% calculate the step size according
% to no.of subdivisions prescirbed
h=(b-a)/n;
% set the starting time value
```

```
tn=a;
% set y initial value
y=y0; % Y1
T(1)=a;
R(1)=y0(1);
cinf(1)=y0(2);
% temporary value for old step
y_temp0=y0;
for i=1:n
    % add the first approximation term to y
    for j=1:2
        y_temp1(j)=0.5*h*feval(f(j),tn,y_temp0);
    end
    y=y+y_temp1/3; % Y1 added
    y_temp2=y_temp0+y_temp1; % Y2
    % add the second approximation term to y
    for j=1:2
        y_temp1(j)=0.5*h*feval(f(j),tn+0.5*h,y_temp2);
    end
    y=y+y_temp1*2/3; %Y2 added
    y_temp2=y_temp0+y_temp1; % Y3
    % add the third approximation term to y
    for j=1:2
        y_temp1(j)=h*feval(f(j),tn+0.5*h,y_temp2);
    end
    y=y+y_temp1/3; % Y3 added
    y_temp2=y_temp0+y_temp1; % Y4
    % add the fourth approximation to y
    for j=1:2
        y_temp1(j)=h*feval(f(j),tn+h,y_temp2);
    end
    y=y+y_temp1/6; % Y4 added
    % update old step value for y
    y_temp0=y;

    % advance in time
    R(i+1)=y(1);
    cinf(i+1)=y(2);

    tn=tn+h;
    T(i+1)=tn;
    e1=abs(R(i+1)-R(i))/R(i);
    if (R(i+1)<0)|(e1<1e-5)
        break;
    end
end
return;
```

C.3 Bubble Growth Rate Function

```
function z=fun1(ts,rc)
% for dR/dt=fun1(t,R)
```

```
global D R T S kh cinf nb pinf p0
%pg=pinf+2*S/rc(1);
rhog=pinf*4.0e-3/8.3125/300; % for xenon, 4.0e-3 changes to xenon's molar mass

cs=pinf*kh; % saturation concentration
if ts==0 z=0;
else
      z=D/rc(1)/rhog*(rc(2)-cs*(1+2*S/rc(1)/pinf));
      z=z/(1+4*S/3/rc(1)/pinf);
      z=z*(1+rc(1)/sqrt(pi*D*ts));
end
return;
```

C.4 Gas Concentration Changing Rate

```
function z=fun2(ts,rc)
% for dcinf/dt=fun2(t,cinf)

global D R T S kh cinf nb pinf p0
pg=pinf+2*S/rc(1);
rhog=pg*4.0e-3/8.3125/300; % for xenon, 4.0e-3 should be replace as xenon's molar mass

z=-rhog*nb*4*pi*rc(1)^2*fun1(ts,rc);
return;
```

Appendix D

MATLAB Script for Two-Group Bubble Growth Rate Simulation

Only the code for helium is attached. For xenon case, see the comment in the script. One only needs to change some values for some constants.

D.1 Master Script

```
clear all;
clc;
global D S kh nb pinf p0 rhog alpha

% constants
alpha=.005; % volume fraction
pinf=3.e5; % pressure in mercury
p0=3.e5; % cover gas pressure before bubble injection
kh=1/4.426e6; % For xenon, changes to xenon's Henry's law constant
r0=15e-6; % nomial bubble radius
D=4.643e-9; % diffusion coefficient. For xenon, change to xenon's value
S=425.41e-3; % surface tension
rhog=pinf*4.0e-3/8.3125/300; % gas density inside bubble; For xenon, 4.0e-3 changes to xenon's molar %
% mass

% initial radius group
m=2; % group number
R=[10 15]; % micro-m
R=R*1.0e-6;
f=[0.5 0.5];
% total bubble number density
nb=alpha/(4*pi/3*sum(R(1:m).^3.*f(1:m)));

% time grid
dt=1.0e-3; % time step. Can be changed for xenon
t(1)=0;

% initial concentration
c(1)=p0*kh;
vf(1)=alpha;
fn=@fun1;
rh(1,:)=R;

i=1;
% time step advancing
while (1)
    i=i+1;
    t(i)=t(i-1)+dt;
    lt=length(R);

    % update the radius value
```

```
    for j=1:lt
        if R(j)>0
            ic=[R(j),c(i-1)];
            Rtemp(j)=rk4(fn,t(i-1),dt,ic);
        else
            Rtemp(j)=0;
        end
    end
    % update the concentration
    dc=0;
    for j=1:lt
        if R(j)>0
            if Rtemp(j)>0
                dc=dc-rhog*(1+2*S/R(j)/pinf)*4*pi/3*(Rtemp(j)^3-R(j)^3)*f(j)*nb;
            else
                dc=dc+4*pi/3*rhog*(1+2*S/R(j)/pinf)*R(j)^3*f(j)*nb;
            end
        end
    end
    c(i)=dc+c(i-1);
    rh(i,:)=Rtemp;
    temp=find(rh(i,:)<0);
    rh(i,temp)=0;

    lt=length(Rtemp);
    vf(i)=0;
    for j=1:lt
        if Rtemp(j)>0
            vf(i)=vf(i)+Rtemp(j)^3*f(j)*4*pi/3*nb;
        end
    end
    %vf(i)=sum(R(1:lt).^3.*f(1:lt))*4*pi/3*nb;
    temp=find(Rtemp>0);
    if (isempty(temp))
        disp(['All bubbles have dissolved at time: ',num2str(t(i))]);
        %keyboard;
        break;
    end
    e1=dc/c(i-1);
    e2=(vf(i)-vf(i-1))/vf(i-1);

    if (abs(e1)<=1.0e-8)&(abs(e2)<=1.0e-6)
        disp('Bubble size distribution stops changing!!!');
        break;
    end
    R=Rtemp;
end
save HeBPM.mat
```

D.2 Fourth-Order Runge-Kutta Function

```
function [nr,nc]=rk4(f,t0,dt,y0)
```

```
% 4th order classic Runge-Kutta
%
% input: function handle(f);solution interval([a,b])
% initial value(y0);no.of subdivisions(n)
%
% output: y(b)
% When the R becomes less than zero or stops changing,
% the computation is stopped
tn=t0;
h=dt;
% set y initial value
y=y0; % Y1
% temporary value for old step
y_temp0=y0;

    % add the first approximation term to y
    for j=1:1
        y_temp1(j)=0.5*h*feval(f(j),tn,y_temp0);
    end
    y=y+y_temp1/3; % Y1 added
    y_temp2=y_temp0+y_temp1; % Y2
    % add the second approximation term to y
    for j=1:1
        y_temp1(j)=0.5*h*feval(f(j),tn+0.5*h,y_temp2);
    end
    y=y+y_temp1*2/3; %Y2 added
    y_temp2=y_temp0+y_temp1; % Y3
    % add the third approximation term to y
    for j=1:1
        y_temp1(j)=h*feval(f(j),tn+0.5*h,y_temp2);
    end
    y=y+y_temp1/3; % Y3 added
    y_temp2=y_temp0+y_temp1; % Y4
    % add the fourth approximation to y
    for j=1:1
        y_temp1(j)=h*feval(f(j),tn+h,y_temp2);
    end
    y=y+y_temp1/6; % Y4 added


    nr=y(1);
    %nc=y(2);

return;
```

D.3 Bubble Growth Rate Function

```
function z=fun1(ts,rc)
% for dR/dt=fun1(t,R)
global D S kh nb pinf p0 rhog
cs=pinf*kh;
if ts==0 z=0;
else
    z=D/rc(1)/rhog*(rc(2)-cs*(1+2*S/rc(1)/pinf));
```

```
        z=z/(1+4*S/3/rc(1)/pinf);
        z=z*(1+rc(1)/sqrt(pi*D*ts));
end
return;
```

Vita

Bo Lu was born in Xinchang, Zhejiang Province, People's Republic of China on March 23, 1980. From September 1986 to July 1998, he attended the public schools in Xinchang. He entered the Department of Engineering Physics at Tsinghua University, Beijing in September 1998, with a major in Nuclear Engineering and Technology. He received the Bachelor of Engineering degree in July 2002. Following the graduation, he joined the Institute of Plasma Physics, Chinese Academy of Sciences and worked there mainly on soft and hard X-ray radiation detection to determine the temperature of electrons in high-temperature plasma from October 2002 to November 2003. In January 2004, he entered Nuclear Engineering Department at the University of Tennessee, Knoxville for graduate study. He has been working in the High Power Target Group as a graduate research assistant on a DOE funded project towards developing methods in understanding the inert gas bubble behavior in the mercury target of the Spallation Neutron Source at Oak Ridge National Laboratory.