1. Code a numerical solution to the RP equation (1).

MATLAB was used to generate a solution to the Rayleigh-Plesset Equation. Variables were re-defined such that y_1 = R and $y_2 = \dot{R}$. This allowed the single 2nd order ODE to become 2 coupled 1^{st} order ODEs. Redefinition of variables resulted in:

$$\dot{y_1} = y_2 \qquad \dot{y_2} = \ddot{R}$$

with

$$\ddot{R} = \left[\frac{P(R,\dot{R},t)}{\rho} - \frac{3}{2}\dot{R}^2\right]\frac{1}{R}$$

where

$$P(R,\dot{R},t) = P_g^e\left(\frac{R_o}{R}\right)^{3k} + P_v - P_\infty - \frac{2\sigma}{R} - \frac{4\mu\dot{R}}{R}$$

$$P_g^e = P_\infty^e - P_v + \frac{2\sigma}{R_o}$$

$$P_\infty = P_\infty^e + P_a sin(\omega t)$$

Shown below are the codes used to obtain solutions to the RP equation

```
% This script will model the Rayleigh Plesset Equations
% Author: Dorien Villafranco
% Department of Mechanical Engineering, Boston University
% Requirements for ME721: Acoustic Bubble Dynamics ~ HW1
clear
clc

t0 = 0;
tf = 0.001;
y0 = [Ro, 0]';
[t,y] = ode23(@Rayleigh,[t0,tf],y0);

figure(1)
hp = plot(t,y(:,1));
grid on
set(hp,'LineWidth',2)
set(gca,'FontSize',12)
xlabel('Time (s)')
ylabel('Bubble Radius (m)')
```

```matlab
function Rdot2 = Rayleigh(t,y)
rho = 998;          % kg/m^3
P_a = 0   ;         % Acoustic Pressure amplitude (Pascals)
w = 2*pi*20*10^3; % Acoustic Driving Frequency (2*pi*20 kHz)
Pv= 2.33*10^3;     % Vapor Pressure (Water/Air) Boundary
 P_inf_e=1*10^5;   % Pressure far away from the bubble at equil(Pa)
k = 1.4;            % Polytropic Exponent of gas (air adiabitc)
sigma = 0.072;     % Surface Tension for Air/Water
nu  = 0.001;       % Shear(dynamic) liquid viscosity for water(Pas)
Ro  = 1*10^-5;     % Equlibrium Bubble Radius in meters
Rdot = y(2);
R = y(1);
% Pressure in the gas @ equlibrium
P_ge = P_inf_e - Pv + (2*sigma/Ro);

% Pressure far away from the bubble
 P_inf = P_inf_e + P_a.*sin(w*t);
P = P_ge*(Ro/R)^(3*k) + Pv - P_inf - (2*sigma/R) - (4*nu*Rdot/R);
Rdot2 = [y(2); (P/rho - (3/2)*y(2).^2)*(1./y(1))];
```

2. Perform the following sanity checks on your code:

---

i. Equilibrium: Set $P_a = 0$; $\dot{R}$(t=0) = 0; R(t=0) = R_o and run the code with these initial conditions for at least t = 0.001s. Plot R(t); is it stable?

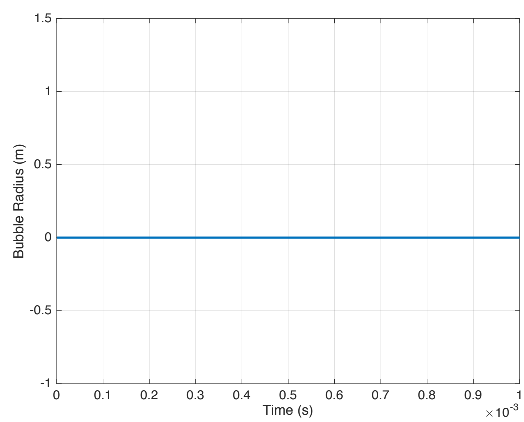

Figure 1: Equilibrium Solution for the Rayleigh Plesset Equation

ii. Ring down: Set run your code for each initial condition. Plot R(t) does it decay to R0? What is the frequency of the ringing? How does it compare with the linear natural frequency.

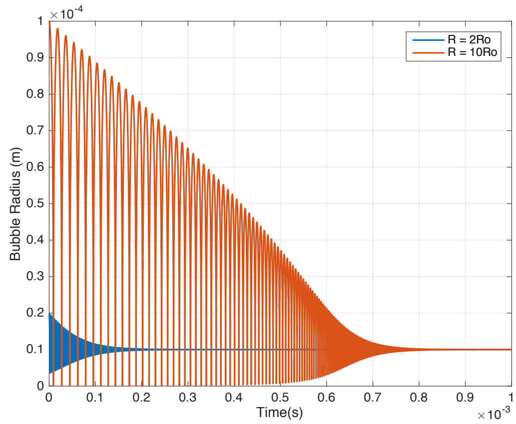

Figure 2: Ring Down test for Rayleigh Plesset Code

iii. Step drive: Set $\dot{R}$(t=0) = 0; R(t=0) = R_0; $P_\infty = P_\infty^e + P_c H(t - t_o)$ where P_c is 1x10^5 Pa and H(t) is the unit step function (Heaviside). Plot R(t) does it make sense?

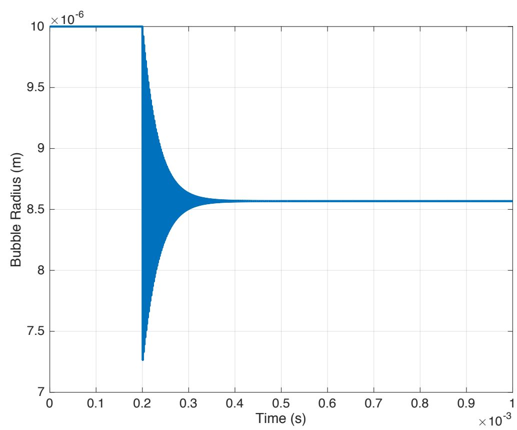

Figure 3: Step Drive test for Rayleigh Plesset Code

3. Benchmark your code against a previously published solution, using their parameter values. You may choose anything nontrivial* in the literature of the past 50 years, but be careful to compare results of the same equation at the same parameters, since there are a variety of Rayleigh-Plesset equations.