Charlie Nitschelm

Pressure Lab

3/26/19

The natural frequency and the damping ratio were extracted from the data using fundamental equations assuming a second order differential system. The damping ratio was calculated by locating all of the major peaks from the data file and using those values as inputs into the damping ratio

(1)

The equation above provides a list of damping ratios for each peak observed in the data sets obtained, including the valve and balloon sets. A mean is then obtained for each data set that displays the predicted overall damping ratio of the system.

Once the damping ratio was determined, the natural frequency of each data set can be determined. To do this, the time of each peak was recorded and used to find the period of the cycle. The period constant can then be inputted into the equation for the damped natural frequency:

(2)

Finally, the damping ratio and damped natural frequency can be the inputs to the natural frequency equation

(3)

Both the damping ratio and the natural frequency of each data set can then be implemented to predict the output of a system given its initial conditions and simulate an arithmetic solution to the curve.

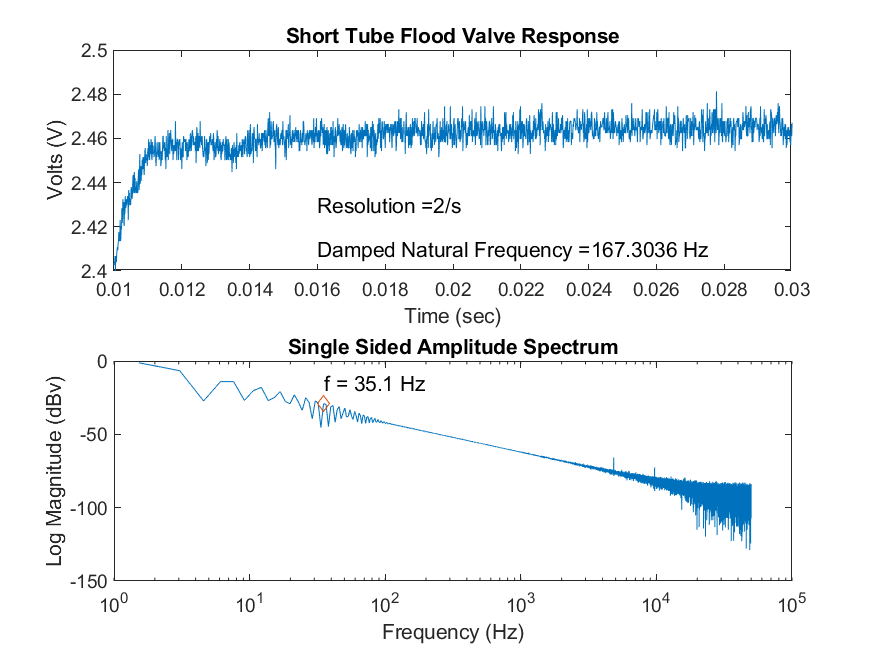


Figure 1 - The 0.17 meter tube with a quick flood valve showing no measurable oscillations impeding data analysis

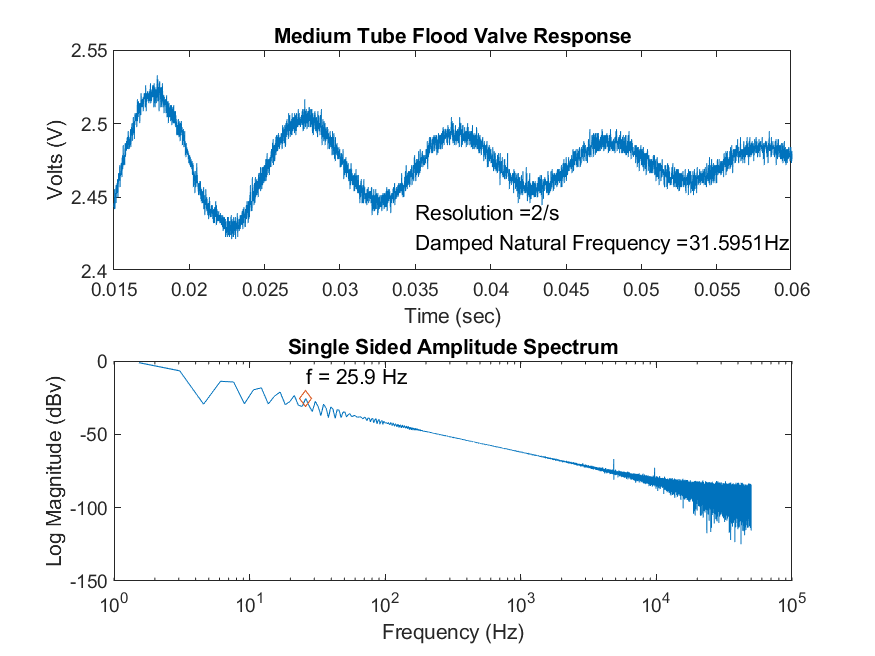


Figure 2 - 0.795 meter long tube with a quick flood valve response imparting measurable peaks for data analysis. FFT calculating the single-sided amplitude spectrum was not able to find the damped natural frequency accurately.

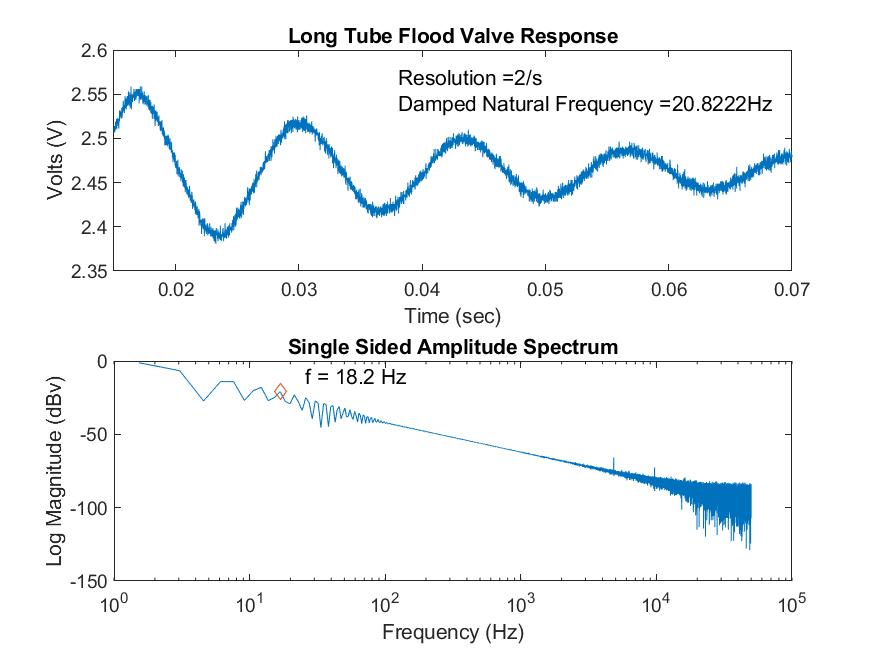


Figure 3 - 1.05 meter long tube with a quick flood valve response imparting measurable peaks for data analysis. FFT calculating the single-sided amplitude spectrum was able to find the damped natural frequency but with significant error.

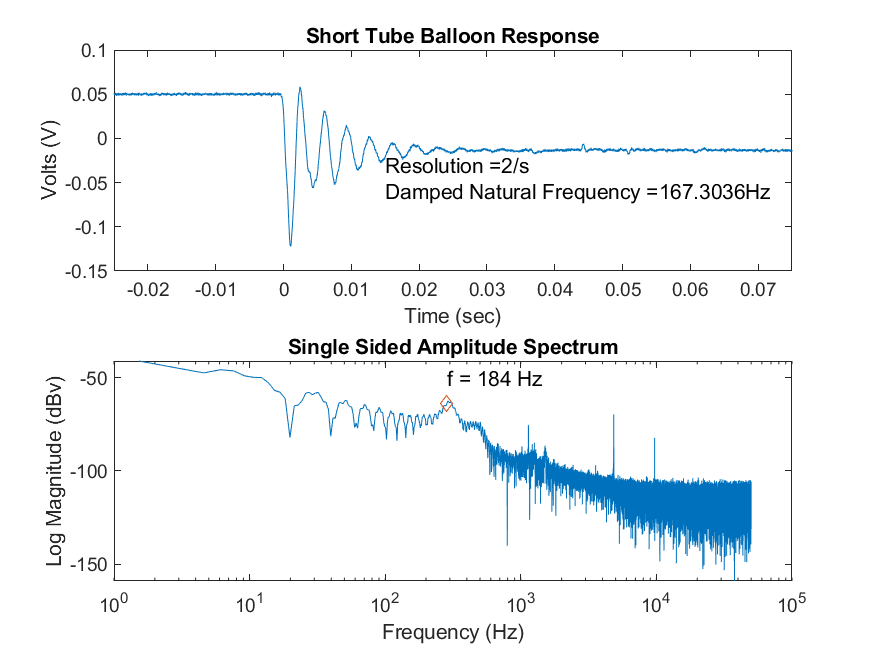


Figure 4 - 0.17 meter long tube response with an inflated balloon attached displaying the damped natural frequency using equations and FFT. Accuracy of the FFT is only 7% off from the actual from the data.

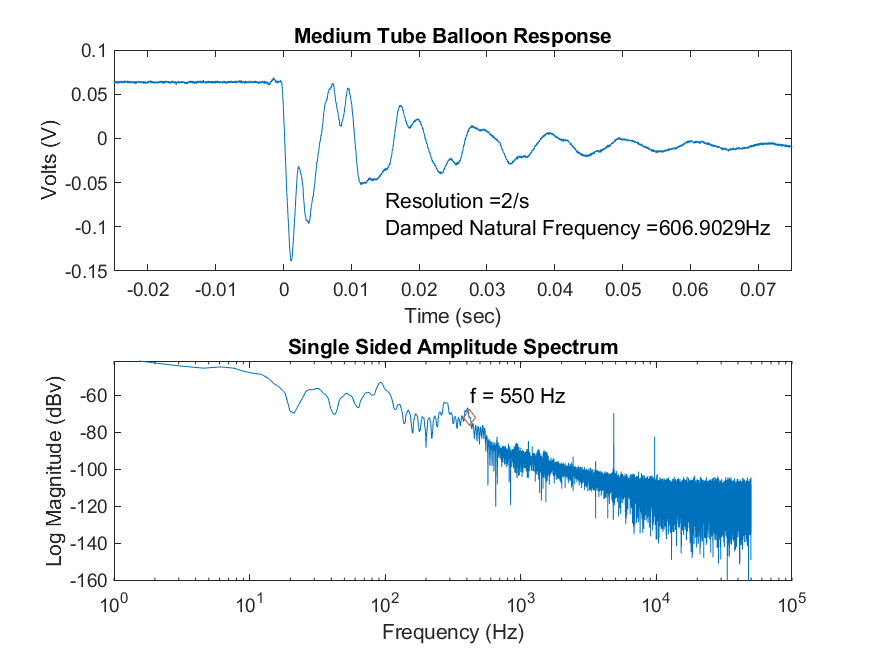


Figure 5 - 0.795 meter long tube response with an inflated balloon attached displaying the damped natural frequency using equations and FFT. Accuracy of the FFT is only 9% off from the actual from the data.

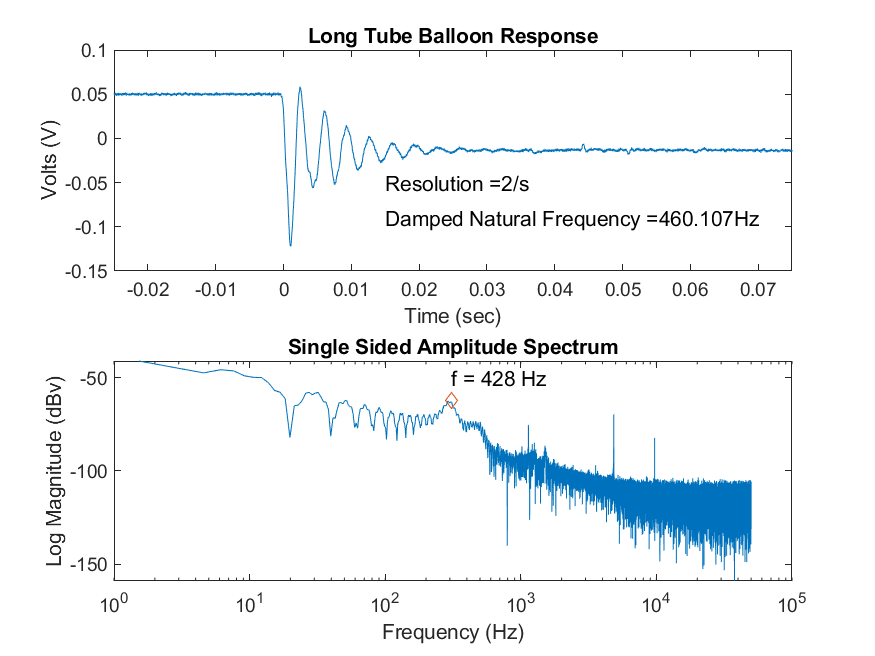


Figure 6 – 1.05 meter long tube response with an inflated balloon attached displaying the damped natural frequency using equations and FFT. Accuracy of the FFT is only 7% off from the actual from the data.

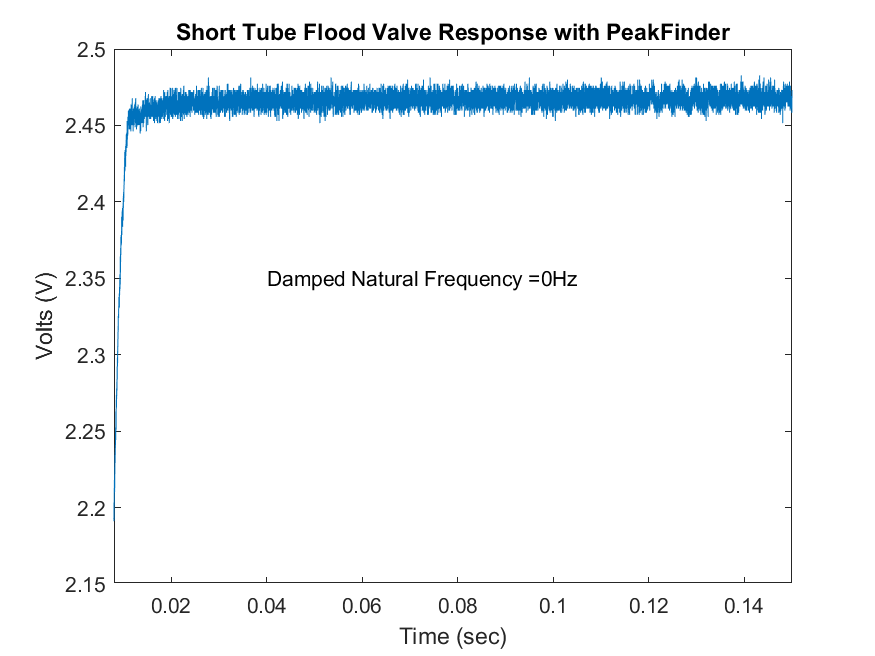


Figure 7 - No peaks found from 0.17 meter flood valve data by not opening the valve quick enough.

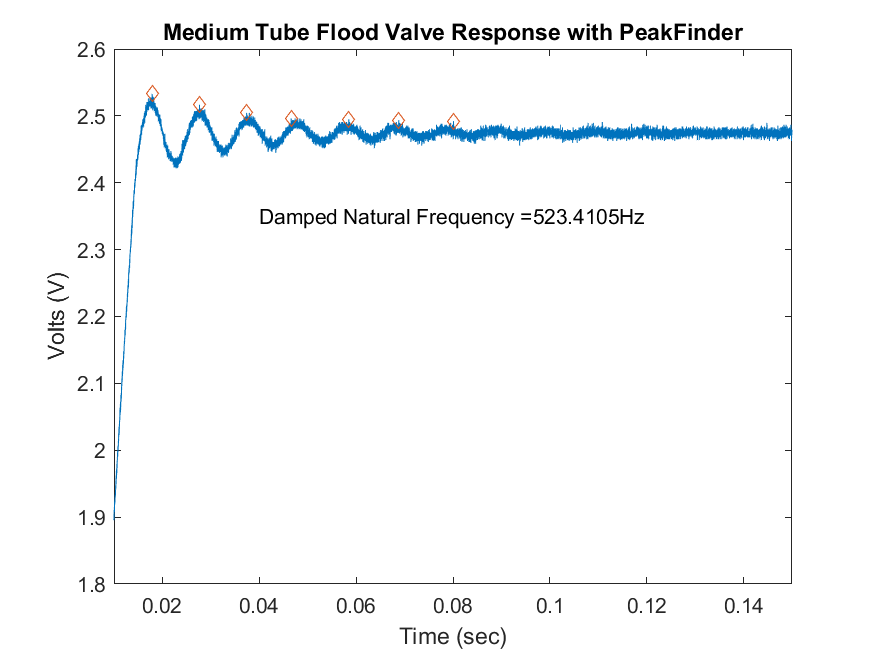


Figure 8 - The calculated damped natural frequency using the 0.795 meter tube and utilizing PeakFinder to identify locations of the peaks in the data

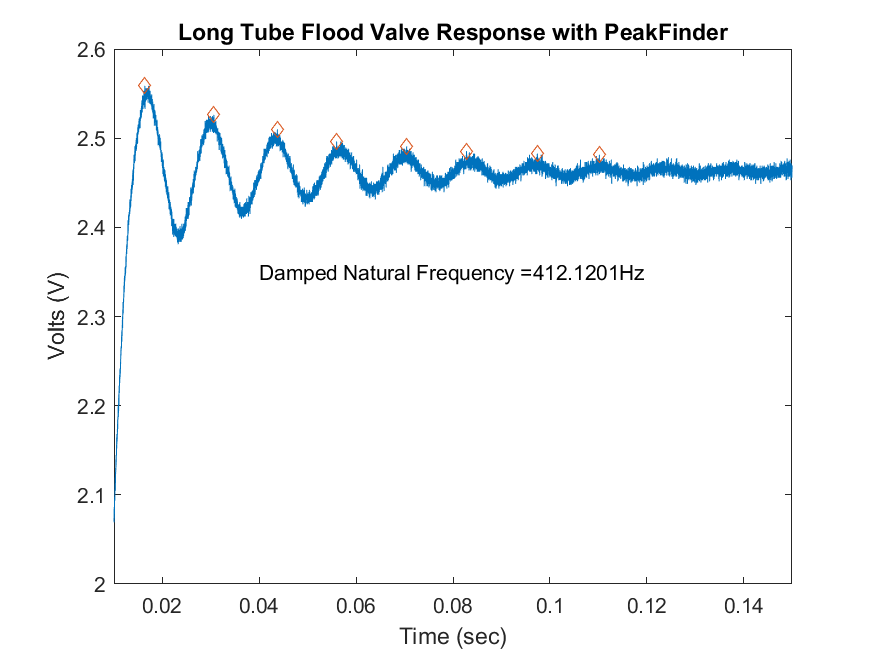


Figure 9 - The calculated damped natural frequency using the 1.05 meter tube and utilizing PeakFinder to identify locations of the peaks in the data

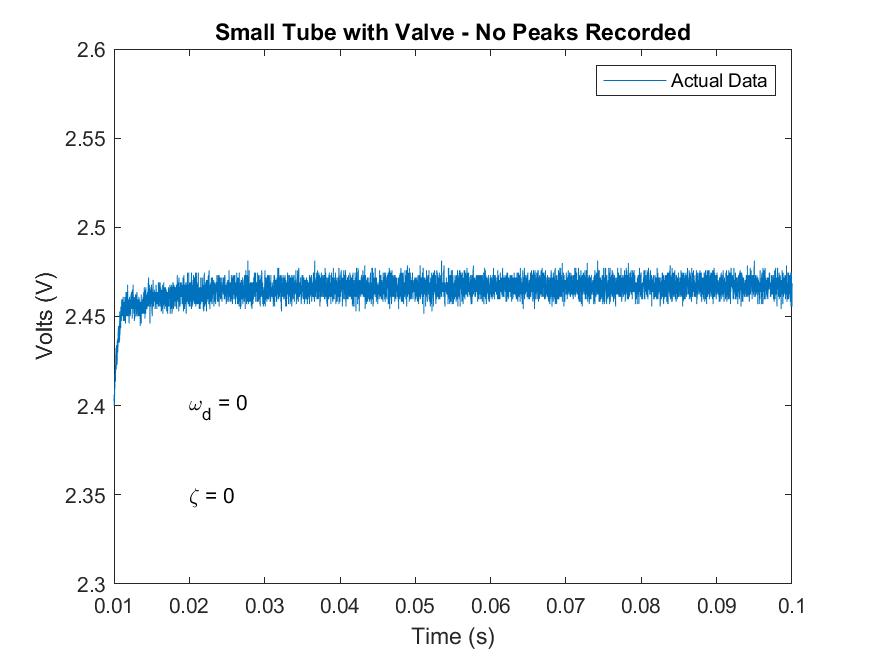


Figure 10 - With no measurable peaks, no prediction analysis could be implemented onto the 0.17 meter tube flood valve data set

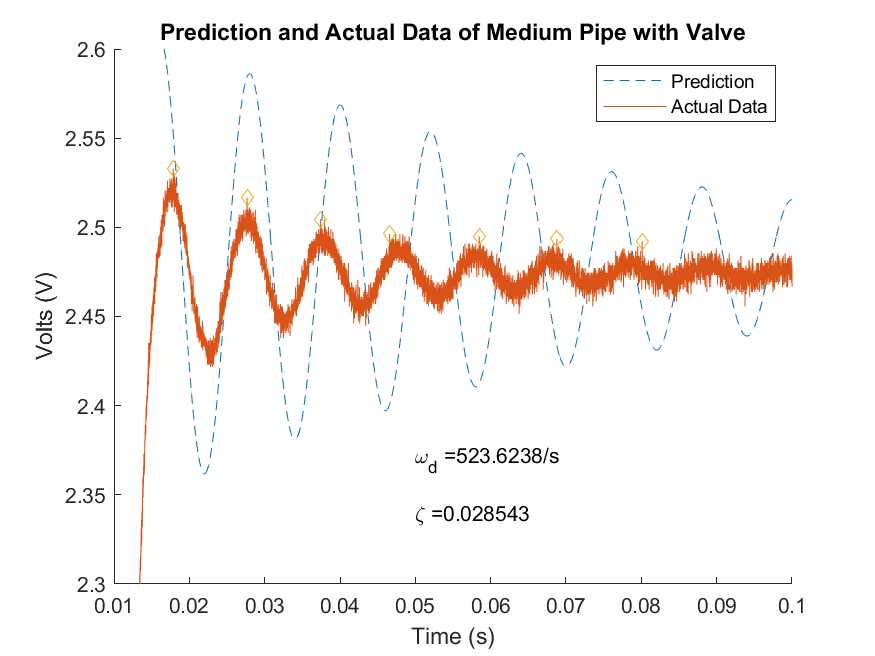
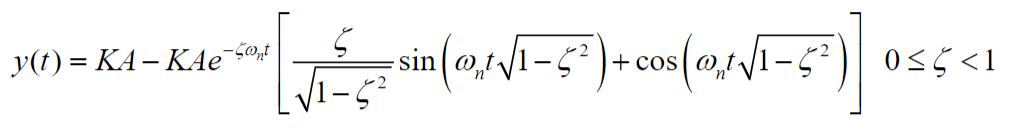


Figure 11 - Prediction and actual data curve utilizing equation 4 with the 0.795 meter tube with flood valve



(4)

Where KA is the settling voltage, and the Natural Frequency and the Damping Ratio were explained above. This way was chosen in competition with the clear code given as a challenge. Although the predictions are not completely in sync with the actual data, the predicted curves follow the same trends with acceptable accuracy. Figure 11 to Figure 15 uses equation 4 to predict the matching curves analytically.

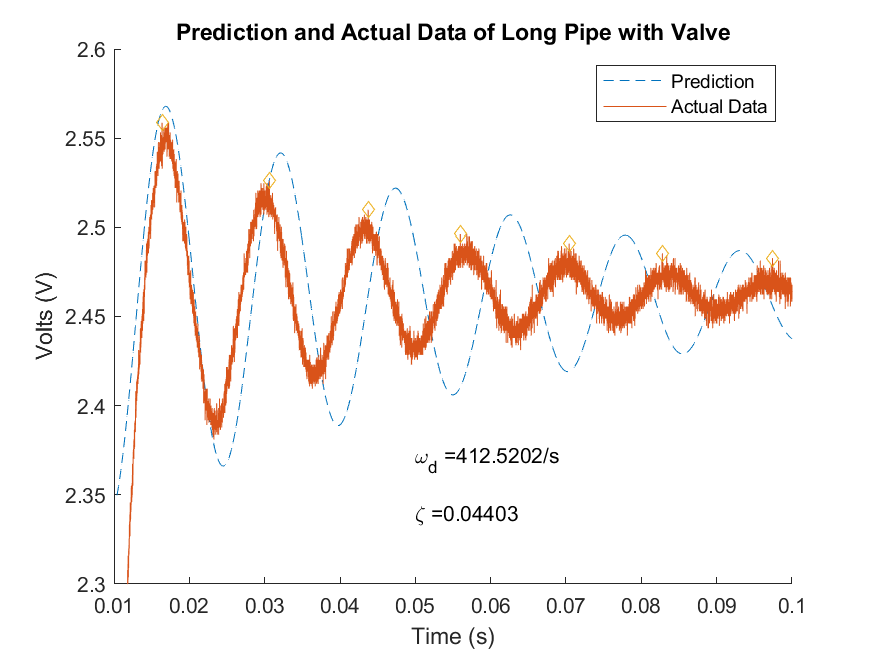


Figure 12 - Prediction and actual data curve utilizing the equation below with the 1.05 meter tube with flood valve using equation 4

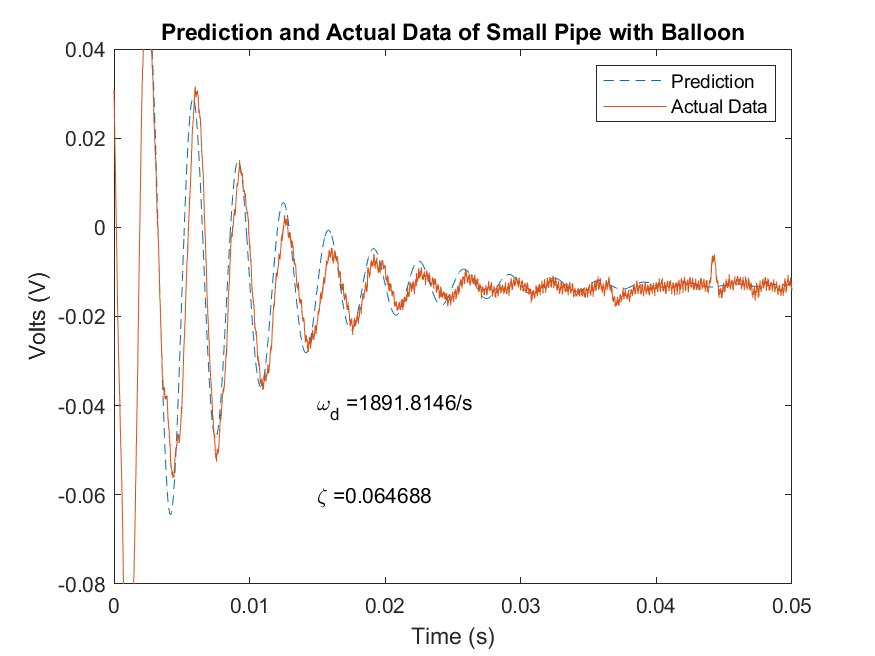


Figure 13 - Prediction and actual data curve utilizing equation 4 with the 0.17 meter tube with an inflated balloon attached

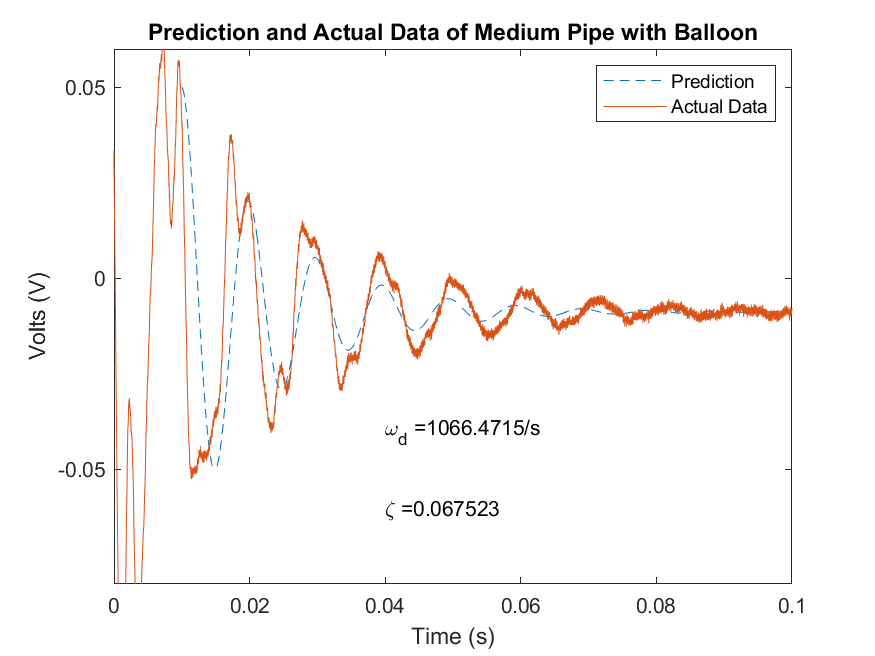


Figure 14 - Prediction and actual data curve utilizing equation 4 with the 0.795 meter tube with an inflated balloon attached

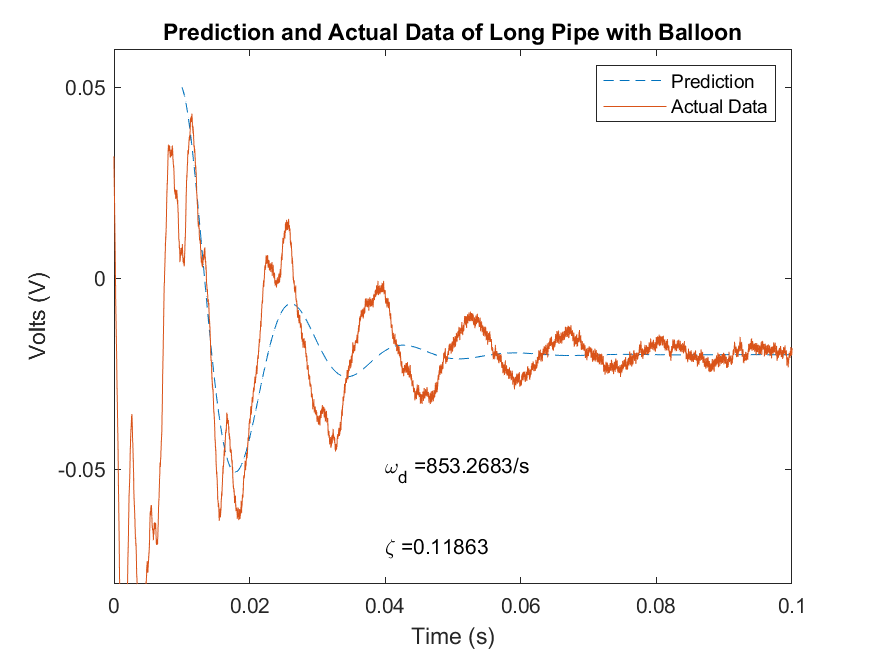


Figure 15 - Prediction and actual data curve utilizing equation 4 with the 1.05 meter tube with an inflated balloon attached

Two fundamental equations were used to calculate a curve to attempt to match with the calculated damping ratio and natural frequencies of each system analyzed. The equation for natural frequency is

(5)

Where a is the speed of sound, 343 m/s, l is the length of the tube used for the test, V is the volume of the pressure transducer, and is the volume of the tube used. By knowing the inputs, a prediction graph could be calculated to obtain a graphical display of predicted and actual natural frequencies.

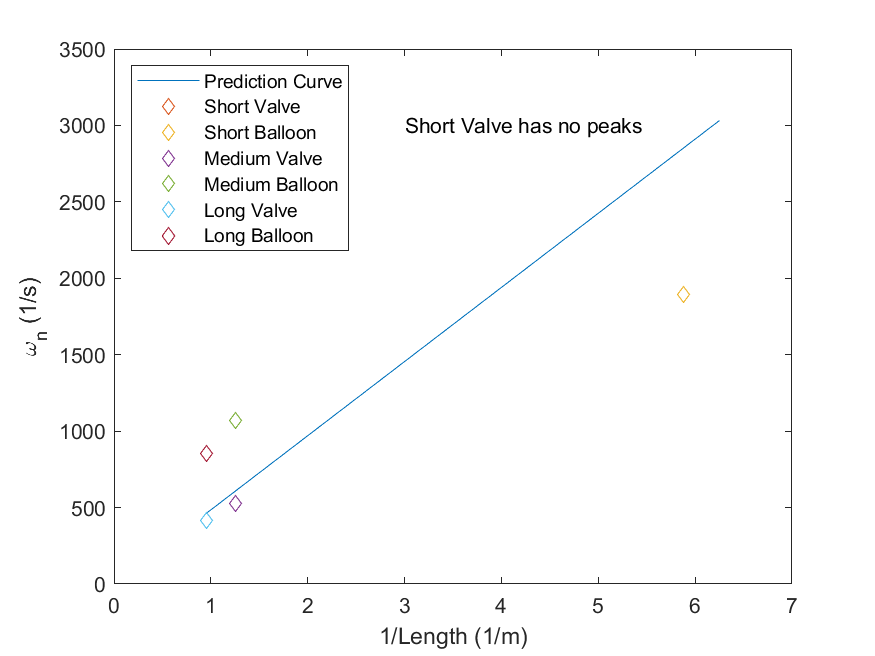


Figure 16 - Predicted curve for the natural frequency of the system with any tube length plotted with actual recorded data

There is a clear linear relationship between natural frequency and the inverse of the tube length. The actual precision of the data to the curve is within +- 500/s for the longest and the second longest tube, while the short tube exhibits and error closer -1000/s. With any data set, a significant amount of error can exist. Within the pressure experiment, the valve data especially exhibited a larger error as the change in voltage from the pressure sensor was quick and large. Second order systems are also known to be difficult to precisely predict an appropriate curve.

The equation for natural frequency is

(6)

Where is kinematic viscosity, 1.81\* kg/ms, l is the length of the tube used for the test, V is the volume of the pressure transducer, and is the volume of the tube used, a is the speed of sound, 343 m/s, is the density of air, 1.225 kg/, and d is the diameter of the tube used. By knowing the inputs, a prediction graph could be calculated to obtain a graphical display of predicted and actual damping ratios.

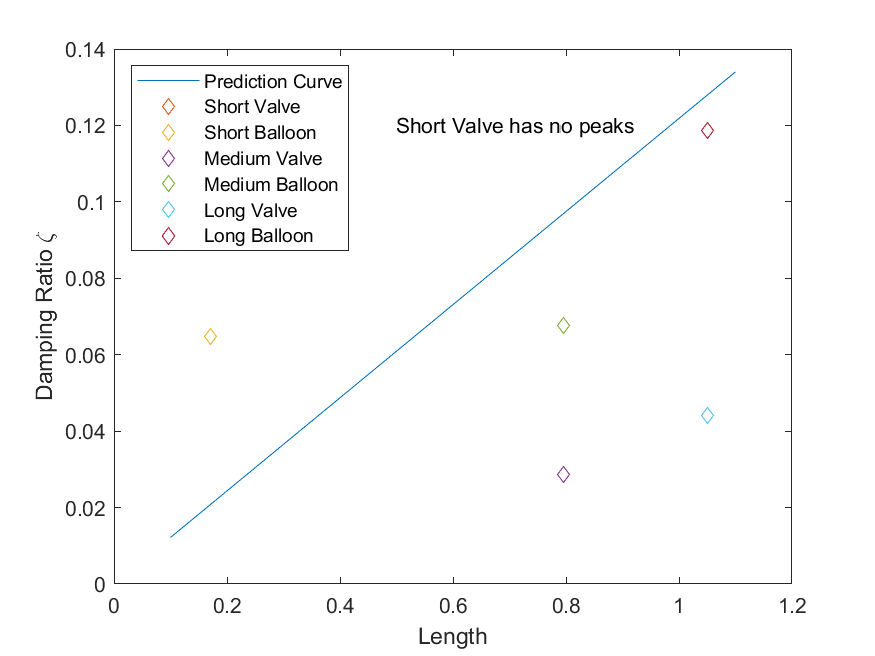


Figure 17 - Predicted curve for the Damping Ratio of the system with any tube length plotted with actual recorded data

There is a clear linear relationship between the damping ratio and the tube length. The actual precision of the data to the curve is within an order of magnitude for the longest and the second longest tube, while the short tube exhibits and error closer to half and order of magnitude. With any data set, a significant amount of error can exist. Within the pressure experiment, the valve data especially exhibited a larger error as the change in voltage from the pressure sensor was quick and large. Second order systems are also known to be difficult to precisely predict an appropriate curve.

For both Figure 16 and Figure 17, a relationship to length for both of these parameters are clear, but significant error is involved limiting the accuracy of the equation to actual calculations.

# Appendix

% Lab 4 %

clear all

close all

## Import Data

% Reading in entire data set

Header=29;

Small\_Flood = importdata('SmallPipe\_Flood.lvm','\t',Header);

Small\_Balloon = importdata('SmallPipe\_Balloon.lvm','\t',Header);

Medium\_Flood = importdata('MediumPipe\_Flood.lvm','\t',Header);

Medium\_Balloon = importdata('MediumPipe\_Balloon.lvm','\t',Header);

Long\_Flood = importdata('LongPipe\_Flood.lvm','\t',Header);

Long\_Balloon = importdata('LongPipe\_Balloon.lvm','\t',Header);

% Separating entire data sets into vectors of Time and Voltage

Time\_Small\_Flood = Small\_Flood.data(:,1);

Time\_Small\_Balloon = Small\_Balloon.data(:,1);

Time\_Medium\_Flood = Medium\_Flood.data(:,1);

Time\_Medium\_Balloon = Medium\_Balloon.data(:,1);

Time\_Long\_Flood = Long\_Flood.data(:,1);

Time\_Long\_Balloon = Long\_Balloon.data(:,1);

Voltage\_Small\_Flood = Small\_Flood.data(:,2);

Voltage\_Small\_Balloon = Small\_Balloon.data(:,2);

Voltage\_Medium\_Flood = Medium\_Flood.data(:,2);

Voltage\_Medium\_Balloon = Medium\_Balloon.data(:,2);

Voltage\_Long\_Flood = Long\_Flood.data(:,2);

Voltage\_Long\_Balloon = Long\_Balloon.data(:,2);

## Flood Tests

% Small Pipe

T = Time\_Small\_Flood(2)-Time\_Small\_Flood(1); % Calculated time interval between data points

Freq = 1/T; % Sampling Frequency

Length = size(Voltage\_Small\_Flood); % Number of points in vector

Power\_2 = 2^nextpow2(Length(1)); % power of 2 from length y

FFT = fft(Voltage\_Small\_Flood,Power\_2)./Length(1); % fft bull

Spaced\_Points = Freq/2\*linspace(0,1,Power\_2/2+1); % spaced point vector using length

FFT\_A = 20\*log10(abs(FFT(1:Power\_2/2+1))); % amplitude

Length\_2 = length(Voltage\_Small\_Flood);

Freq\_Reso\_Flood\_Small = 1/(Length\_2\*T); %FOR GRAPH

a = 343;

l = .17;

V = l\*pi\*.0042^2;

Vt = 6.5\*10^-8;

Damp\_Natural\_Freq\_Small\_Flood = a/(l\*sqrt(0.5+(V/Vt))); % FOR GRAPH

f1 = figure(1)

subplot(2,1,1)

plot (Time\_Small\_Flood,Voltage\_Small\_Flood);

title('Short Tube Flood Valve Response')

xlabel ('Time (sec)')

ylabel ('Volts (V)')

text(.016,2.43,strcat('Resolution =',num2str(Freq\_Reso\_Flood\_Small),'/s'))

text(.016,2.41,strcat('Damped Natural Frequency = ',num2str(Damp\_Natural\_Freq\_Small\_Flood),' Hz'))

xlim([0.01,0.03])

subplot(2,1,2)

semilogx(Spaced\_Points,FFT\_A);

hold on

plot(35.1,-29.14,'d')

text(35.1,-15,'f = 35.1 Hz')

title('Single Sided Amplitude Spectrum')

xlabel ('Frequency (Hz)')

ylabel ('Log Magnitude (dBv)')

% Medium Pipe

T = Time\_Medium\_Flood(2)-Time\_Medium\_Flood(1); % Calculated time interval between data points

Freq = 1/T; % Sampling Frequency

Length = size(Voltage\_Medium\_Flood); % Number of points in vector

Power\_2 = 2^nextpow2(Length(1)); % power of 2 from length y

FFT = fft(Voltage\_Medium\_Flood,Power\_2)./Length(1); % fft bull

Spaced\_Points = Freq/2\*linspace(0,1,Power\_2/2+1); % spaced point vector using length

FFT\_A = 20\*log10(abs(FFT(1:Power\_2/2+1))); % amplitude

Length\_2 = length(Voltage\_Medium\_Flood);

Freq\_Reso\_Flood\_Medium = 1/(Length\_2\*T); %FOR GRAPH

a = 343;

l = .795;

V = l\*pi\*.0022^2;

Vt = 6.5\*10^-8;

Damp\_Natural\_Freq\_Medium\_Flood = a/(l\*sqrt(0.5+(V/Vt)));

f2 = figure(2)

subplot(2,1,1)

plot (Time\_Medium\_Flood,Voltage\_Medium\_Flood);

title('Medium Tube Flood Valve Response')

xlabel ('Time (sec)')

ylabel ('Volts (V)')

text(.035,2.44,strcat('Resolution =',num2str(Freq\_Reso\_Flood\_Medium),'/s'))

text(.035,2.42,strcat('Damped Natural Frequency = ',num2str(Damp\_Natural\_Freq\_Medium\_Flood),'Hz'))

xlim([0.015,0.06])

subplot(2,1,2)

semilogx(Spaced\_Points,FFT\_A);

hold on

plot(25.9,-25.52,'d')

text(25.9,-10,'f = 25.9 Hz')

title('Single Sided Amplitude Spectrum')

xlabel ('Frequency (Hz)')

ylabel ('Log Magnitude (dBv)')

% Long Pipe

T = Time\_Long\_Flood(2)-Time\_Long\_Flood(1); % Calculated time interval between data points

Freq = 1/T; % Sampling Frequency

Length = size(Voltage\_Small\_Flood); % Number of points in vector

Power\_2 = 2^nextpow2(Length(1)); % power of 2 from length y

FFT = fft(Voltage\_Small\_Flood,Power\_2)./Length(1); % fft bull

Spaced\_Points = Freq/2\*linspace(0,1,Power\_2/2+1); % spaced point vector using length

FFT\_A = 20\*log10(abs(FFT(1:Power\_2/2+1))); % amplitude

Length\_2 = length(Voltage\_Long\_Flood);

Freq\_Reso\_Flood\_Long = 1/(Length\_2\*T); %FOR GRAPH

a = 343;

l = 1.05;

V = l\*pi\*.0022^2;

Vt = 6.5\*10^-8;

Damp\_Natural\_Freq\_Long\_Flood = a/(l\*sqrt(0.5+(V/Vt)));

f3 = figure(3)

subplot(2,1,1)

plot (Time\_Long\_Flood,Voltage\_Long\_Flood);

title('Long Tube Flood Valve Response')

xlabel ('Time (sec)')

ylabel ('Volts (V)')

text(.038,2.57,strcat('Resolution =',num2str(Freq\_Reso\_Flood\_Long),'/s'))

text(.038,2.54,strcat('Damped Natural Frequency = ',num2str(Damp\_Natural\_Freq\_Long\_Flood),'Hz'))

xlim([0.015,.07])

subplot(2,1,2)

semilogx(Spaced\_Points,FFT\_A);

hold on

plot(16.78,-20.82,'d')

text(25.78,-10,'f = 18.2 Hz')

title('Single Sided Amplitude Spectrum')

xlabel ('Frequency (Hz)')

ylabel ('Log Magnitude (dBv)')

f1 =

Figure (1) with properties:

Number: 1

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f2 =

Figure (2) with properties:

Number: 2

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f3 =

Figure (3) with properties:

Number: 3

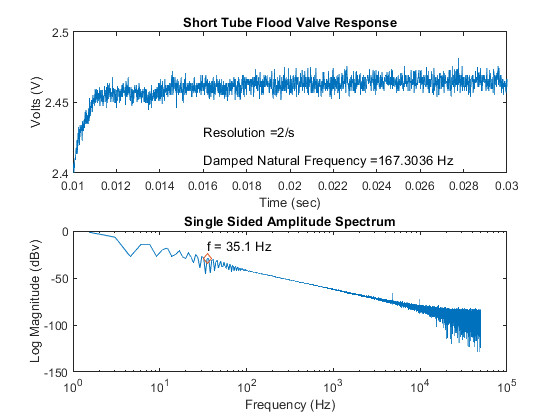
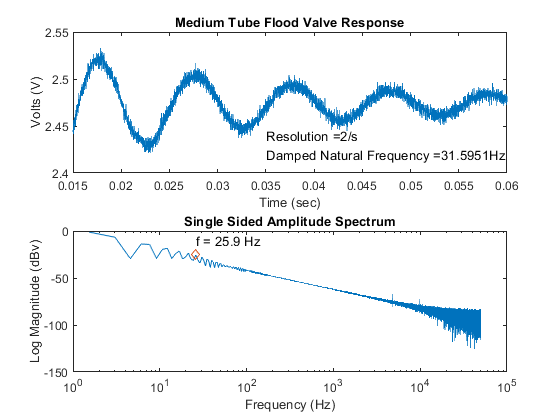
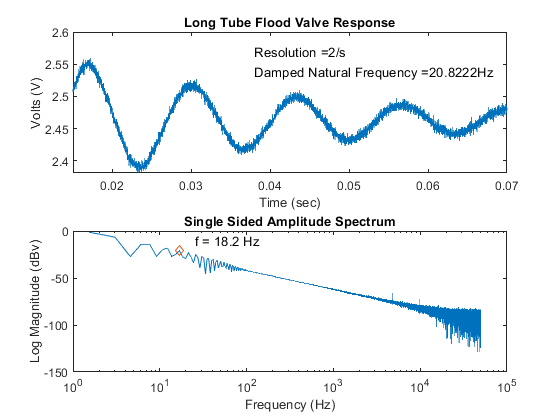
Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

## Balloon Tests

% Small Pipe

T = Time\_Small\_Balloon(2)-Time\_Small\_Balloon(1); % Calculated time interval between data points

Freq = 1/T; % Sampling Frequency

Length = size(Voltage\_Small\_Balloon); % Number of points in vector

Power\_2 = 2^nextpow2(Length(1)); % power of 2 from length y

FFT = fft(Voltage\_Small\_Balloon,Power\_2)./Length(1); % fft bull

Spaced\_Points = Freq/2\*linspace(0,1,Power\_2/2+1); % spaced point vector using length

FFT\_A = 20\*log10(abs(FFT(1:Power\_2/2+1))); % amplitude

Length\_2 = length(Voltage\_Small\_Balloon);

Freq\_Reso\_Balloon\_Small = 1/(Length\_2\*T); %FOR GRAPH

a = 343;

l = .17;

V = l\*pi\*.0042^2;

Vt = 6.5\*10^-8;

Damp\_Natural\_Freq\_Small\_Balloon = a/(l\*sqrt(0.5+(V/Vt)));

f4 = figure(4)

subplot(2,1,1)

plot (Time\_Small\_Balloon,Voltage\_Small\_Balloon);

title('Short Tube Balloon Response')

xlabel ('Time (sec)')

ylabel ('Volts (V)')

text(.015,-.03,strcat('Resolution =',num2str(Freq\_Reso\_Balloon\_Small),'/s'))

text(.015,-.06,strcat('Damped Natural Frequency = ',num2str(Damp\_Natural\_Freq\_Small\_Balloon),'Hz'))

xlim([-.025,.075])

subplot(2,1,2)

semilogx(Spaced\_Points,FFT\_A);

hold on

plot(285,-63.6,'d')

text(285,-50,'f = 184 Hz')

title('Single Sided Amplitude Spectrum')

xlabel ('Frequency (Hz)')

ylabel ('Log Magnitude (dBv)')

% Medium Pipe

T = Time\_Medium\_Balloon(2)-Time\_Medium\_Balloon(1); % Calculated time interval between data points

Freq = 1/T; % Sampling Frequency

Length = size(Voltage\_Medium\_Balloon); % Number of points in vector

Power\_2 = 2^nextpow2(Length(1)); % power of 2 from length y

FFT = fft(Voltage\_Medium\_Balloon,Power\_2)./Length(1); % fft bull

Spaced\_Points = Freq/2\*linspace(0,1,Power\_2/2+1); % spaced point vector using length

FFT\_A = 20\*log10(abs(FFT(1:Power\_2/2+1))); % amplitude

Length\_2 = length(Voltage\_Medium\_Balloon);

Freq\_Reso\_Balloon\_Medium = 1/(Length\_2\*T); %FOR GRAPH

a = 343;

l = .795;

Vt = l\*pi\*.0022^2;

V = 6.5\*10^-8;

Damp\_Natural\_Freq\_Medium\_Balloon = a/(l\*sqrt(0.5+(V/Vt)));

f5 = figure(5)

subplot(2,1,1)

plot (Time\_Medium\_Balloon,Voltage\_Medium\_Balloon);

title('Medium Tube Balloon Response')

xlabel ('Time (sec)')

ylabel ('Volts (V)')

text(.015,-.07,strcat('Resolution =',num2str(Freq\_Reso\_Balloon\_Medium),'/s'))

text(.015,-.10,strcat('Damped Natural Frequency = ',num2str(Damp\_Natural\_Freq\_Medium\_Balloon),'Hz'))

xlim([-.025,.075])

subplot(2,1,2)

semilogx(Spaced\_Points,FFT\_A);

hold on

plot(420,-71.9,'d')

text(420,-60,'f = 550 Hz')

title('Single Sided Amplitude Spectrum')

xlabel ('Frequency (Hz)')

ylabel ('Log Magnitude (dBv)')

% Long Pipe

T = Time\_Long\_Balloon(2)-Time\_Long\_Balloon(1); % Calculated time interval between data points

Freq = 1/T; % Sampling Frequency

Length = size(Voltage\_Small\_Balloon); % Number of points in vector

Power\_2 = 2^nextpow2(Length(1)); % power of 2 from length y

FFT = fft(Voltage\_Small\_Balloon,Power\_2)./Length(1); % fft bull

Spaced\_Points = Freq/2\*linspace(0,1,Power\_2/2+1); % spaced point vector using length

FFT\_A = 20\*log10(abs(FFT(1:Power\_2/2+1))); % amplitude

Length\_2 = length(Voltage\_Long\_Balloon);

Freq\_Reso\_Balloon\_Long = 1/(Length\_2\*T); %FOR GRAPH

a = 343;

l = 1.05;

Vt = l\*pi\*.0022^2;

V = 6.5\*10^-8;

Damp\_Natural\_Freq\_Long\_Balloon = a/(l\*sqrt(0.5+(V/Vt)));

f6 = figure(6)

subplot(2,1,1)

plot (Time\_Long\_Balloon,Voltage\_Small\_Balloon);

title('Long Tube Balloon Response')

xlabel ('Time (sec)')

ylabel ('Volts (V)')

text(.015,-.05,strcat('Resolution =',num2str(Freq\_Reso\_Balloon\_Long),'/s'))

text(.015,-.09,strcat('Damped Natural Frequency = ',num2str(Damp\_Natural\_Freq\_Long\_Balloon),'Hz'))

xlim([-.025,.075])

subplot(2,1,2)

semilogx(Spaced\_Points,FFT\_A);

hold on

plot(308,-62,'d')

text(308,-50,'f = 428 Hz')

title('Single Sided Amplitude Spectrum')

xlabel ('Frequency (Hz)')

ylabel ('Log Magnitude (dBv)')

f4 =

Figure (4) with properties:

Number: 4

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f5 =

Figure (5) with properties:

Number: 5

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f6 =

Figure (6) with properties:

Number: 6

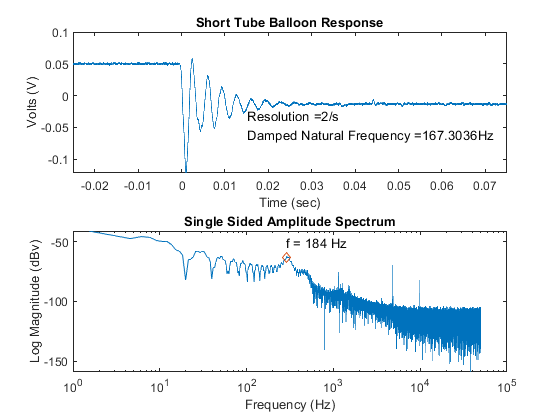
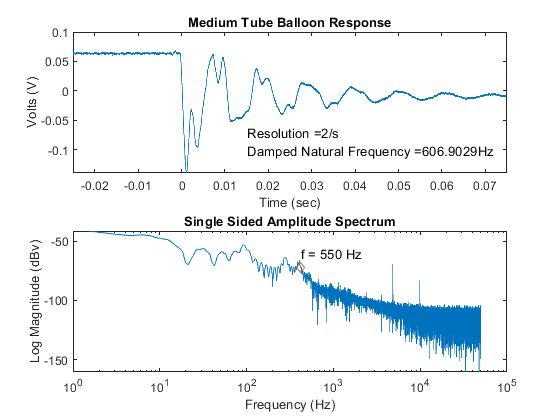
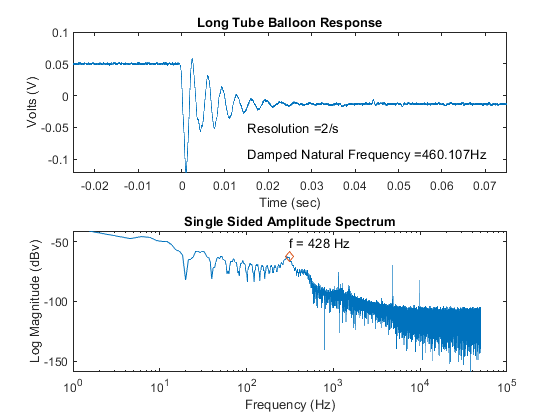
Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

## Peakfinder

% Small Pipe

Constant = 1;

[Volt\_Locations\_Small,Volt\_Small]=peakfinder(Voltage\_Small\_Flood,Constant);

Damping\_Ratio\_Small\_Flood = zeros(length(Volt\_Small),1);

for n = 1:length(Volt\_Small)-1

Period\_Small = Time\_Small\_Flood(Volt\_Locations\_Small(n+1))-Time\_Small\_Flood(Volt\_Locations\_Small(n));

Damping\_Ratio\_Small\_Flood(n) = (2\*pi)/Period\_Small;

end

Damp\_Mean\_Small = mean(Damping\_Ratio\_Small\_Flood);

f7 = figure(7)

plot(Time\_Small\_Flood,Voltage\_Small\_Flood,Time\_Small\_Flood(Volt\_Locations\_Small),Volt\_Small,'d')

title('Short Tube Flood Valve Response with PeakFinder')

text(.04,2.35,strcat('Damped Natural Frequency =',num2str(Damp\_Mean\_Small),'Hz'))

xlabel ('Time (sec)')

ylabel ('Volts (V)')

xlim([.008,.15])

% Medium Pipe

Constant = 0.036;

[Volt\_Locations\_Medium,Volt\_Medium]=peakfinder(Voltage\_Medium\_Flood,Constant);

Damping\_Ratio\_Medium\_Flood = zeros(length(Volt\_Medium),1);

for n = 1:length(Volt\_Medium)-1

Period\_Medium = Time\_Medium\_Flood(Volt\_Locations\_Medium(n+1))-Time\_Medium\_Flood(Volt\_Locations\_Medium(n));

Damping\_Ratio\_Medium\_Flood(n) = (2\*pi)/Period\_Medium;

end

Damp\_Mean\_Medium = mean(Damping\_Ratio\_Medium\_Flood);

f8 = figure(8)

plot(Time\_Medium\_Flood,Voltage\_Medium\_Flood,Time\_Medium\_Flood(Volt\_Locations\_Medium),Volt\_Medium,'d')

title('Medium Tube Flood Valve Response with PeakFinder')

text(.04,2.35,strcat('Damped Natural Frequency =',num2str(Damp\_Mean\_Medium),'Hz'))

xlabel ('Time (sec)')

ylabel ('Volts (V)')

xlim([.01,.15])

% Long Pipe

Constant = 0.035;

[Volt\_Locations\_Long,Volt\_Long]=peakfinder(Voltage\_Long\_Flood,Constant);

Damping\_Ratio\_Long\_Flood = zeros(length(Volt\_Long),1);

for n = 1:length(Volt\_Long)-1

Period\_Long = Time\_Long\_Flood(Volt\_Locations\_Long(n+1))-Time\_Long\_Flood(Volt\_Locations\_Long(n));

Damping\_Ratio\_Long\_Flood(n) = (2\*pi)/Period\_Long;

end

Damp\_Mean\_Long = mean(Damping\_Ratio\_Long\_Flood);

f9 = figure(9)

plot(Time\_Long\_Flood,Voltage\_Long\_Flood,Time\_Long\_Flood(Volt\_Locations\_Long),Volt\_Long,'d')

title('Long Tube Flood Valve Response with PeakFinder')

text(.04,2.35,strcat('Damped Natural Frequency =',num2str(Damp\_Mean\_Long),'Hz'))

xlabel ('Time (sec)')

ylabel ('Volts (V)')

xlim([.01,.15])

f7 =

Figure (7) with properties:

Number: 7

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f8 =

Figure (8) with properties:

Number: 8

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f9 =

Figure (9) with properties:

Number: 9

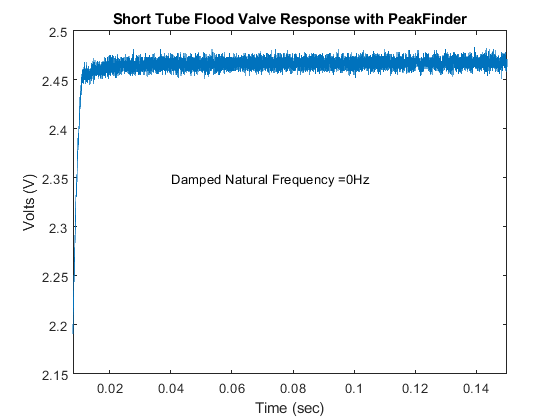
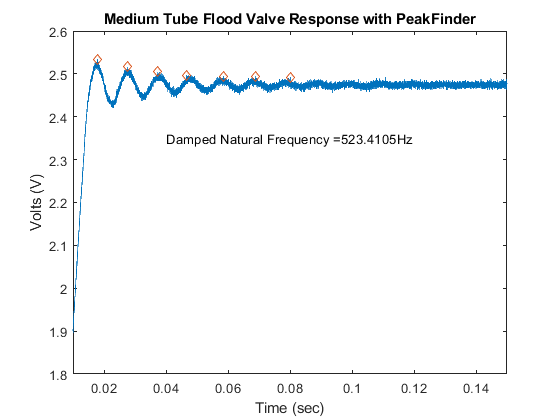
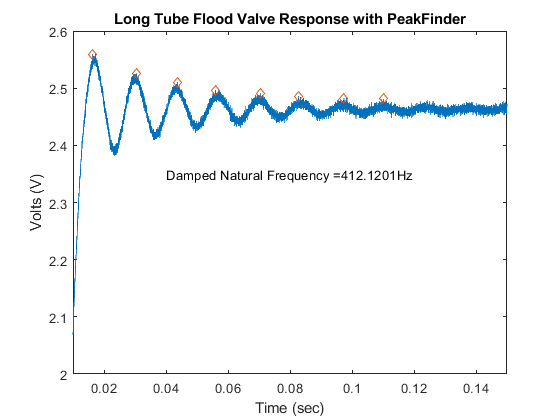
Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

## p-t response and predicted

Volt\_Locations\_Small;

Volt\_Locations\_Medium;

Volt\_Locations\_Long;

Volt\_Small;

Volt\_Medium;

Volt\_Long;

Constant = 0.01;

[Volt\_Locations\_Balloon\_Small,Volt\_Small\_Balloon]=peakfinder(Voltage\_Small\_Balloon,Constant);

Volt\_Locations\_Balloon\_Small = Volt\_Locations\_Balloon\_Small(2:end-1);

Volt\_Small\_Balloon = Volt\_Small\_Balloon(2:end-1);

% figure(10)

% plot(Time\_Small\_Balloon,Voltage\_Small\_Balloon,Time\_Small\_Balloon(Volt\_Locations\_Balloon\_Small),Volt\_Small\_Balloon,'d')

Constant = 0.02;

[Volt\_Locations\_Balloon\_Medium,Volt\_Medium\_Balloon]=peakfinder(Voltage\_Medium\_Balloon,Constant);

Volt\_Locations\_Balloon\_Medium = Volt\_Locations\_Balloon\_Medium(3:end);

Volt\_Medium\_Balloon = Volt\_Medium\_Balloon(3:end);

% figure(11)

% plot(Time\_Medium\_Balloon,Voltage\_Medium\_Balloon,Time\_Medium\_Balloon(Volt\_Locations\_Balloon\_Medium),Volt\_Medium\_Balloon,'d')

Constant = 0.015;

[Volt\_Locations\_Balloon\_Long,Volt\_Long\_Balloon]=peakfinder(Voltage\_Long\_Balloon,Constant);

Volt\_Locations\_Balloon\_Long = Volt\_Locations\_Balloon\_Long(3:end);

Volt\_Long\_Balloon = Volt\_Long\_Balloon(3:end);

% figure(12)

% plot(Time\_Long\_Balloon,Voltage\_Long\_Balloon,Time\_Long\_Balloon(Volt\_Locations\_Balloon\_Long),Volt\_Long\_Balloon,'d')

Damp\_Small\_Valve = zeros(length(Volt\_Locations\_Small)-1,1);

Damp\_Medium\_Valve = zeros(length(Volt\_Locations\_Medium)-1,1);

Damp\_Long\_Valve = zeros(length(Volt\_Locations\_Long)-1,1);

Damp\_Small\_Balloon = zeros(length(Volt\_Locations\_Balloon\_Small)-1,1);

Damp\_Medium\_Balloon = zeros(length(Volt\_Locations\_Balloon\_Medium)-1,1);

Damp\_Long\_Balloon = zeros(length(Volt\_Locations\_Balloon\_Long)-1,1);

for n=2:length(Volt\_Locations\_Small)

Num = (1/(n-1))\*log(Volt\_Small(2)./Volt\_Small(n));

Damp\_Small\_Valve(n-1) = Num./sqrt(4\*pi^2+Num.^2);

end

for n=2:length(Volt\_Locations\_Medium)

Num = 100\*(1/(n-1))\*log(Volt\_Medium(2)/Volt\_Medium(n));

Damp\_Medium\_Valve(n-1) = Num/(sqrt(4\*pi^2+Num^2));

end

for n=2:length(Volt\_Locations\_Long)

Num = 100\*(1/(n-1))\*log(Volt\_Long(2)./Volt\_Long(n));

Damp\_Long\_Valve(n-1) = Num./sqrt(4\*pi^2+Num^2);

end

for n=2:length(Volt\_Locations\_Balloon\_Small)

Num = (1/(n-1))\*abs(log(Volt\_Small\_Balloon(1)./Volt\_Small\_Balloon(n)));

Damp\_Small\_Balloon(n-1) = 0.5\*(Num./sqrt(4\*pi^2+Num^2));

end

for n=2:length(Volt\_Locations\_Balloon\_Medium)

Num = (1/(n-1))\*log(Volt\_Medium\_Balloon(1)./Volt\_Medium\_Balloon(n));

Damp\_Medium\_Balloon(n-1) = Num./sqrt(4\*pi^2+Num^2);

end

for n=2:length(Volt\_Locations\_Balloon\_Long)

Num = (1/(n-1))\*abs(log(Volt\_Long\_Balloon(1)./Volt\_Long\_Balloon(n)));

Damp\_Long\_Balloon(n-1) = Num./sqrt(4\*pi^2+Num^2);

end

Ratio\_Valve\_Small = mean(Damp\_Small\_Valve);

Ratio\_Valve\_Medium = mean(Damp\_Medium\_Valve);

Ratio\_Valve\_Long = mean(Damp\_Long\_Valve);

Ratio\_Balloon\_Small = mean(Damp\_Small\_Balloon);

Ratio\_Balloon\_Medium = mean(Damp\_Medium\_Balloon);

Ratio\_Balloon\_Long = mean(Damp\_Long\_Balloon);

Damp\_Mean\_Small;

Damp\_Mean\_Medium;

Damp\_Mean\_Long;

for n = 1:length(Volt\_Small\_Balloon)-1

Period\_Small = Time\_Small\_Balloon(Volt\_Locations\_Balloon\_Small(n+1))-Time\_Small\_Balloon(Volt\_Locations\_Balloon\_Small(n));

Damping\_Ratio\_Small\_Balloon(n) = 2\*pi/Period\_Small;

end

Damp\_Mean\_Small\_Balloon = mean(Damping\_Ratio\_Small\_Balloon);

for n = 1:length(Volt\_Medium\_Balloon)-1

Period\_Medium = Time\_Medium\_Balloon(Volt\_Locations\_Balloon\_Medium(n+1))-Time\_Medium\_Balloon(Volt\_Locations\_Balloon\_Medium(n));

Damping\_Ratio\_Medium\_Balloon(n) = 2\*pi/Period\_Medium;

end

Damp\_Mean\_Medium\_Balloon = mean(Damping\_Ratio\_Medium\_Balloon);

for n = 1:length(Volt\_Long\_Balloon)-1

Period\_Long = Time\_Long\_Balloon(Volt\_Locations\_Balloon\_Long(n+1))-Time\_Long\_Balloon(Volt\_Locations\_Balloon\_Long(n));

Damping\_Ratio\_Long\_Balloon(n) = 2\*pi/Period\_Long;

end

Damp\_Mean\_Long\_Balloon = mean(Damping\_Ratio\_Long\_Balloon);

Natural\_Freq\_Small\_Valve = Damp\_Mean\_Small/sqrt(1-Ratio\_Valve\_Small^2);

Natural\_Freq\_Medium\_Valve = Damp\_Mean\_Medium/sqrt(1-Ratio\_Valve\_Medium^2);

Natural\_Freq\_Long\_Valve = Damp\_Mean\_Long/sqrt(1-Ratio\_Valve\_Long^2);

Natural\_Freq\_Small\_Balloon = Damp\_Mean\_Small\_Balloon/sqrt(1-Ratio\_Balloon\_Small^2);

Natural\_Freq\_Medium\_Balloon = Damp\_Mean\_Medium\_Balloon/sqrt(1-Ratio\_Balloon\_Medium^2);

Natural\_Freq\_Long\_Balloon = Damp\_Mean\_Long\_Balloon/sqrt(1-Ratio\_Balloon\_Long^2);

% Ratio\_Valve\_Small = mean(Damp\_Small\_Valve);

% Ratio\_Valve\_Medium = mean(Damp\_Medium\_Valve);

% Ratio\_Valve\_Long = mean(Damp\_Long\_Valve);

% Ratio\_Balloon\_Small = mean(Damp\_Small\_Balloon);

% Ratio\_Balloon\_Medium = mean(Damp\_Medium\_Balloon);

% Ratio\_Balloon\_Long = mean(Damp\_Long\_Balloon);

Time\_Small\_Flood2 = Time\_Small\_Flood+.05;

KA = 2.46;

e = exp((-Ratio\_Valve\_Small\*Natural\_Freq\_Small\_Valve.\*Time\_Small\_Flood2));

a = Ratio\_Valve\_Small/(sqrt(1-Ratio\_Valve\_Small^2));

b = Natural\_Freq\_Small\_Valve.\*Time\_Small\_Flood2\*sqrt(1-Ratio\_Valve\_Small^2);

Volt\_Predict\_Valve\_Small = KA-KA.\*e.\*(a.\*sin(b)+cos(b));

f13 = figure(13)

plot(Time\_Small\_Flood,Voltage\_Small\_Flood)

text(.02,2.4,'\omega\_d = 0')

text(.02,2.35,'\zeta = 0')

title('Small Tube with Valve - No Peaks Recorded')

xlabel('Time (s)')

ylabel('Volts (V)')

legend('Actual Data','location','northeast')

xlim([0.01,.10])

ylim([2.3,2.6])

Time\_Medium\_Flood2 = Time\_Medium\_Flood +.0677;

KA = 2.479;

e = .0677.\*exp((-Ratio\_Valve\_Medium\*Natural\_Freq\_Medium\_Valve.\*Time\_Medium\_Flood2));

a = Ratio\_Valve\_Medium/(sqrt(1-Ratio\_Valve\_Medium^2));

b = Natural\_Freq\_Medium\_Valve.\*Time\_Medium\_Flood2\*sqrt(1-Ratio\_Valve\_Medium^2);

Volt\_Predict\_Valve\_Medium = KA-KA.\*e.\*(a.\*sin(b)+cos(b));

f14 = figure(14)

hold on

plot(Time\_Medium\_Flood2-.002,Volt\_Predict\_Valve\_Medium,'--')

plot(Time\_Medium\_Flood,Voltage\_Medium\_Flood,Time\_Medium\_Flood(Volt\_Locations\_Medium),Volt\_Medium,'d')

text(.05,2.37,strcat('\omega\_d = ',num2str(Natural\_Freq\_Medium\_Valve),'/s'))

text(.05,2.34,strcat('\zeta = ',num2str(Ratio\_Valve\_Medium)))

title('Prediction and Actual Data of Medium Pipe with Valve')

xlabel('Time (s)')

ylabel('Volts (V)')

legend('Prediction','Actual Data','location','northeast')

xlim([0.01,.10])

ylim([2.3,2.6])

Time\_Long\_Flood2 = Time\_Long\_Flood+.0664;

KA = 2.46;

e = .0664\*exp((-Ratio\_Valve\_Long\*Natural\_Freq\_Long\_Valve.\*Time\_Long\_Flood2));

a = Ratio\_Valve\_Long/(sqrt(1-Ratio\_Valve\_Long^2));

b = Natural\_Freq\_Long\_Valve.\*Time\_Long\_Flood2\*sqrt(1-Ratio\_Valve\_Long^2);

Volt\_Predict\_Valve\_Long = KA-KA.\*e.\*(a.\*sin(b)+cos(b));

f15 = figure(15)

hold on

plot(Time\_Long\_Flood2-.006,Volt\_Predict\_Valve\_Long,'--')

plot(Time\_Long\_Flood,Voltage\_Long\_Flood,Time\_Long\_Flood(Volt\_Locations\_Long),Volt\_Long,'d')

text(.05,2.37,strcat('\omega\_d = ',num2str(Natural\_Freq\_Long\_Valve),'/s'))

text(.05,2.34,strcat('\zeta = ',num2str(Ratio\_Valve\_Long)))

title('Prediction and Actual Data of Long Pipe with Valve')

xlabel('Time (s)')

ylabel('Volts (V)')

legend('Prediction','Actual Data','location','northeast')

xlim([0.01,.10])

ylim([2.3,2.6])

Time\_Small\_Balloon2 = Time\_Small\_Balloon+.05;

KA = -.014;

e = exp((-Ratio\_Balloon\_Small\*Natural\_Freq\_Small\_Balloon.\*Time\_Small\_Balloon2));

a = Ratio\_Balloon\_Small/(sqrt(1-Ratio\_Balloon\_Small.^2));

b = Natural\_Freq\_Small\_Balloon.\*Time\_Small\_Balloon2\*sqrt(1-Ratio\_Balloon\_Small^2);

Volt\_Predict\_Balloon\_Small = .05+4.5\*(KA-KA.\*e.\*(a.\*sin(b)+cos(b))); % Added addition multiplier to start at peak and multiplier for reaching KA

f16 = figure(16)

plot(Time\_Small\_Balloon2+.0025,Volt\_Predict\_Balloon\_Small,'--',Time\_Small\_Balloon,Voltage\_Small\_Balloon)

text(.015,-.04,strcat('\omega\_d = ',num2str(Natural\_Freq\_Small\_Balloon),'/s'))

text(.015,-.06,strcat('\zeta = ',num2str(Ratio\_Balloon\_Small)))

title('Prediction and Actual Data of Small Pipe with Balloon')

xlabel('Time (s)')

ylabel('Volts (V)')

legend('Prediction','Actual Data','location','northeast')

xlim([0,.05])

ylim([-.08,.04])

Time\_Medium\_Balloon2 = Time\_Medium\_Balloon+.05;

KA = -.008;

e = exp((-Ratio\_Balloon\_Medium\*Natural\_Freq\_Medium\_Balloon.\*Time\_Medium\_Balloon2));

a = Ratio\_Balloon\_Medium/(sqrt(1-Ratio\_Balloon\_Medium^2));

b = 0.6\*Natural\_Freq\_Medium\_Balloon.\*Time\_Medium\_Balloon2\*sqrt(1-Ratio\_Balloon\_Medium^2); % Added multiplier here for aligning data

Volt\_Predict\_Balloon\_Medium = .05+7.35.\*(KA-KA.\*e.\*(a.\*sin(b)+cos(b)));% Added addition multiplier to start at peak and multiplier for reaching KA

f17 = figure(17)

plot(Time\_Medium\_Balloon2+.01,Volt\_Predict\_Balloon\_Medium,'--',Time\_Medium\_Balloon,Voltage\_Medium\_Balloon)

text(.04,-.04,strcat('\omega\_d = ',num2str(Natural\_Freq\_Medium\_Balloon),'/s'))

text(.04,-.06,strcat('\zeta = ',num2str(Ratio\_Balloon\_Medium)))

title('Prediction and Actual Data of Medium Pipe with Balloon')

xlabel('Time (s)')

ylabel('Volts (V)')

legend('Prediction','Actual Data','location','northeast')

xlim([0,.1])

ylim([-.08,.06])

Time\_Long\_Balloon2 = Time\_Long\_Balloon+.05;

KA = -.02;

e = exp((-Ratio\_Balloon\_Long\*Natural\_Freq\_Long\_Balloon.\*Time\_Long\_Balloon2));

a = Ratio\_Balloon\_Long/(sqrt(1-Ratio\_Balloon\_Long^2));

b = .45.\*Natural\_Freq\_Long\_Balloon.\*Time\_Long\_Balloon2\*sqrt(1-Ratio\_Balloon\_Long^2);

Volt\_Predict\_Balloon\_Long = .05+3.5.\*(KA-KA.\*e.\*(a.\*sin(b)+cos(b)));

f18 = figure(18)

plot(Time\_Long\_Balloon2+.01,Volt\_Predict\_Balloon\_Long,'--',Time\_Long\_Balloon,Voltage\_Long\_Balloon)

text(.04,-.05,strcat('\omega\_d = ',num2str(Natural\_Freq\_Long\_Balloon),'/s'))

text(.04,-.07,strcat('\zeta = ',num2str(Ratio\_Balloon\_Long)))

title('Prediction and Actual Data of Long Pipe with Balloon')

xlabel('Time (s)')

ylabel('Volts (V)')

legend('Prediction','Actual Data','location','northeast')

xlim([0,.1])

ylim([-.08,.06])

f13 =

Figure (13) with properties:

Number: 13

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f14 =

Figure (14) with properties:

Number: 14

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f15 =

Figure (15) with properties:

Number: 15

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f16 =

Figure (16) with properties:

Number: 16

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f17 =

Figure (17) with properties:

Number: 17

Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

f18 =

Figure (18) with properties:

Number: 18

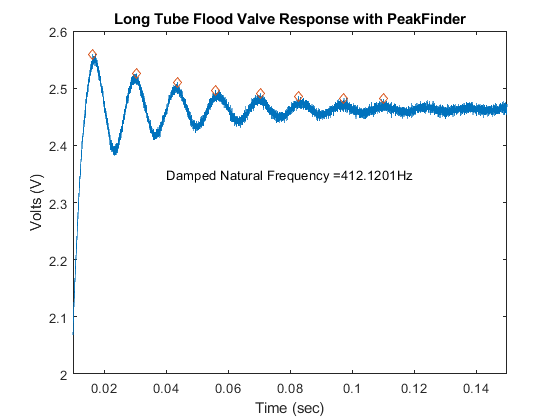
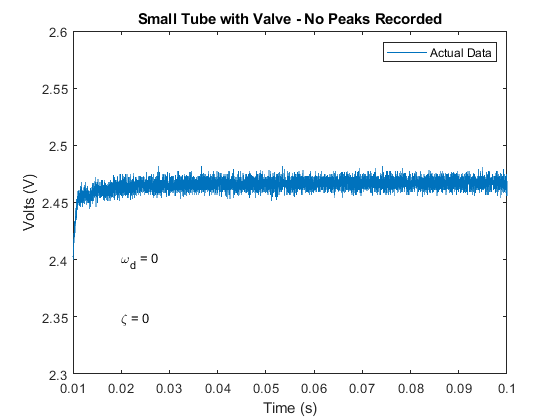
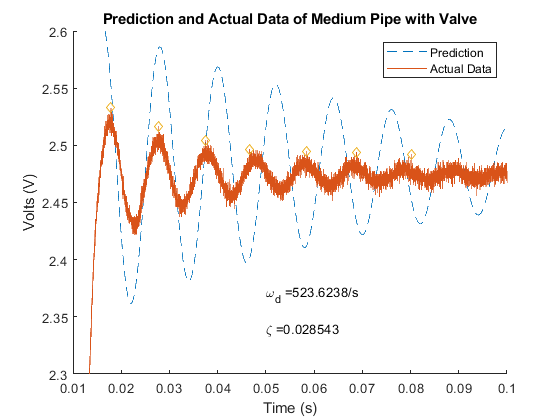
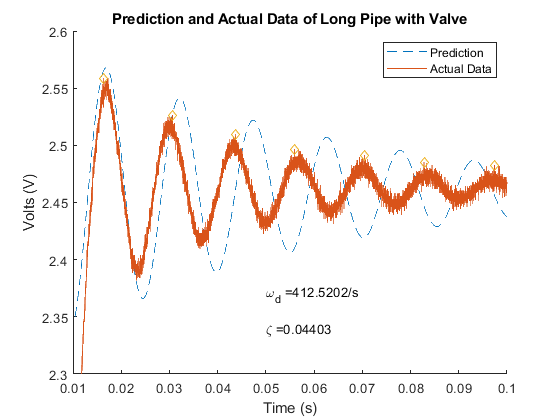
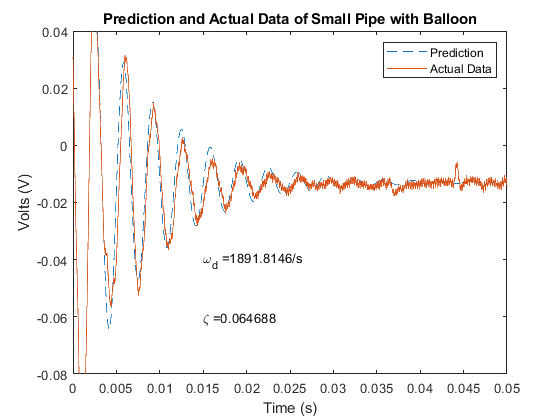
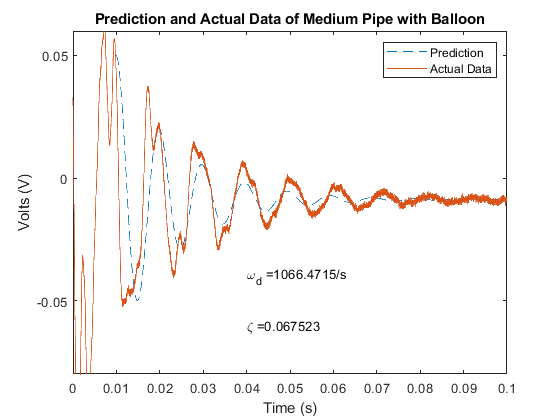
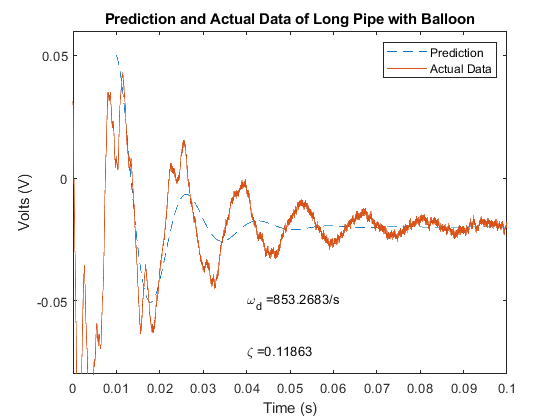
Name: ''

Color: [0.9400 0.9400 0.9400]

Position: [488 342 560 420]

Units: 'pixels'

Use GET to show all properties

## Damping Ratio and Natural Frequency Dependence on Length

a = 343; % m/s

nu = 1.81\*10^-5; % kg/ms

density = 1.225; %kg/m3

l = linspace(.16,1.06,100);

omega\_n = a./(l\*sqrt(0.5));

l\_long = 1.05;

l\_medium = .795;

l\_short = .17;

Natural\_Freq\_Small\_Valve;

Natural\_Freq\_Medium\_Valve;

Natural\_Freq\_Long\_Valve;

Natural\_Freq\_Small\_Balloon;

Natural\_Freq\_Medium\_Balloon;

Natural\_Freq\_Long\_Balloon;

f19 = figure(19)

plot(1./l,omega\_n)

hold on

plot(1/l\_short,Natural\_Freq\_Small\_Valve,'d',1/l\_short,Natural\_Freq\_Small\_Balloon,'d')

plot(1/l\_medium,Natural\_Freq\_Medium\_Valve,'d',1/l\_medium,Natural\_Freq\_Medium\_Balloon,'d')

plot(1/l\_long,Natural\_Freq\_Long\_Valve,'d',1/l\_long,Natural\_Freq\_Long\_Balloon,'d')

text(3,3000,'Short Valve has no peaks')

xlabel('1/Length (1/m)')

ylabel('\omega\_n (1/s)')

legend('Prediction Curve','Short Valve','Short Balloon','Medium Valve','Medium Balloon','Long Valve','Long Balloon','location','northwest')

% Damping Ratio

l = linspace(.10,1.1,100);

Damping = ((16\*nu.\*l)/(a\*density\*.002^2))\*sqrt(.5);

f20 = figure(20)

plot(l,Damping)

hold on

plot(l\_short,Ratio\_Valve\_Small,'d',l\_short,Ratio\_Balloon\_Small,'d')

plot(l\_medium,Ratio\_Valve\_Medium,'d',l\_medium,Ratio\_Balloon\_Medium,'d')

plot(l\_long,Ratio\_Valve\_Long,'d',l\_long,Ratio\_Balloon\_Long,'d')

text(.5,.12,'Short Valve has no peaks')

xlabel('Length')

ylabel('Damping Ratio \zeta')

legend('Prediction Curve','Short Valve','Short Balloon','Medium Valve','Medium Balloon','Long Valve','Long Balloon','location','northwest')

saveas(f1, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure1.png','png');

saveas(f2, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure2.png','png');

saveas(f3, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure3.png','png');

saveas(f4, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure4.png','png');

saveas(f5, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure5.png','png');

saveas(f6, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure6.png','png');

saveas(f7, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure7.png','png');

saveas(f8, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure8.png','png');

saveas(f9, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure9.png','png');

% saveas(f10, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure10.png','png');

% saveas(f11, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure11.png','png');

% saveas(f12, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure12.png','png');

saveas(f13, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure13.png','png');

saveas(f14, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure14.png','png');

saveas(f15, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure15.png','png');

saveas(f16, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure16.png','png');

saveas(f17, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure17.png','png');

saveas(f18, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure18.png','png');

saveas(f19, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure19.png','png');

saveas(f20, 'C:\Users\User\Desktop\Classes\Junior Year\Spring 2019\J-Lab\Lab 4\Figures\figure20.png','png');