

Problem 1 - Cut-on Frequencies in Ducts and Pipes

The Matlab code for this problem is listed in Appendix [1](#).

Problem 1a

The lowest cut-on frequency for a rectangular duct with air flow is given by equation,

$$f_{\text{cut-on}} = 0.5 \cdot \frac{c}{L} \quad (1)$$

where c is the speed of sound in air, $343 \frac{\text{m}}{\text{s}}$, and L is the largest side of the rectangular cross-section.

With cross-sectional dimensions of $L_x = 12 \text{ cm}$ and $L_y = 20 \text{ cm}$, the lowest cut-on frequency for this rectangular duct is,

$$f_{\text{cut-on}} = 0.5 \cdot \frac{343 \frac{\text{m}}{\text{s}}}{0.20 \text{ m}} = \mathbf{857.5 \text{ Hz}}$$

Problem 1b

The lowest cut-on frequency for a circular duct with air flow with the same cross-sectional area as the rectangular duct in part (a.) can be calculated using equation,

$$f_{\text{cut-on}} = 0.568 \cdot \frac{c}{d} \quad (2)$$

where c is the speed of sound in air, $343 \frac{\text{m}}{\text{s}}$, and d is diameter of the circular duct.

The cross-sectional area of the rectangular duct is,

$$\text{Area}_{\text{rectangular duct}} = 0.12 \text{ m} \cdot 0.20 \text{ m} = 0.024 \text{ m}^2$$

The corresponding diameter for this area is,

$$\text{diameter} = \sqrt{\frac{0.024 \text{ m}^2}{\pi}} \cdot 2 = 0.17 \text{ m}$$

Using Eq. [2](#), the lowest cut-on frequency for this circular duct with air flow is,

$$f_{\text{cut-on}} = 0.568 \cdot \frac{1,500 \frac{\text{m}}{\text{s}}}{0.17 \text{ m}} = \mathbf{1,114.5 \text{ Hz}}$$

Problem 1c

The lowest cut-on frequency for this circular duct with water flow can be calculated using Eq. 2,

$$f_{\text{cut-on}} = 0.568 \cdot \frac{1,500 \frac{\text{m}}{\text{s}}}{0.17 \text{ m}} = 4,873.9 \text{ Hz}$$

The lowest cut-on frequency for water is considerable larger than it is for air flow.

Problem 1d

The speed of sound in air is calculated by,

$$c = \sqrt{\gamma \cdot R \cdot T_K} \quad (3)$$

where $\gamma = 1.4$ is the ratio of specific heats, $R = 287 \frac{\text{J}}{\text{kg} \cdot \text{K}}$ is the gas constant, and T_K is the absolute temperature in Kelvin.

Figure 1 illustrates how the lowest cut-on frequency changes as the air heats from 0° to 500° Celsius.

The square-root relationship between temperature and the speed of sound in air is apparent and governs the behaviour of the cut-on frequency.

Lowest Cut-on Frequency for a Circular Pipe with Air Flow Versus Air Temperature

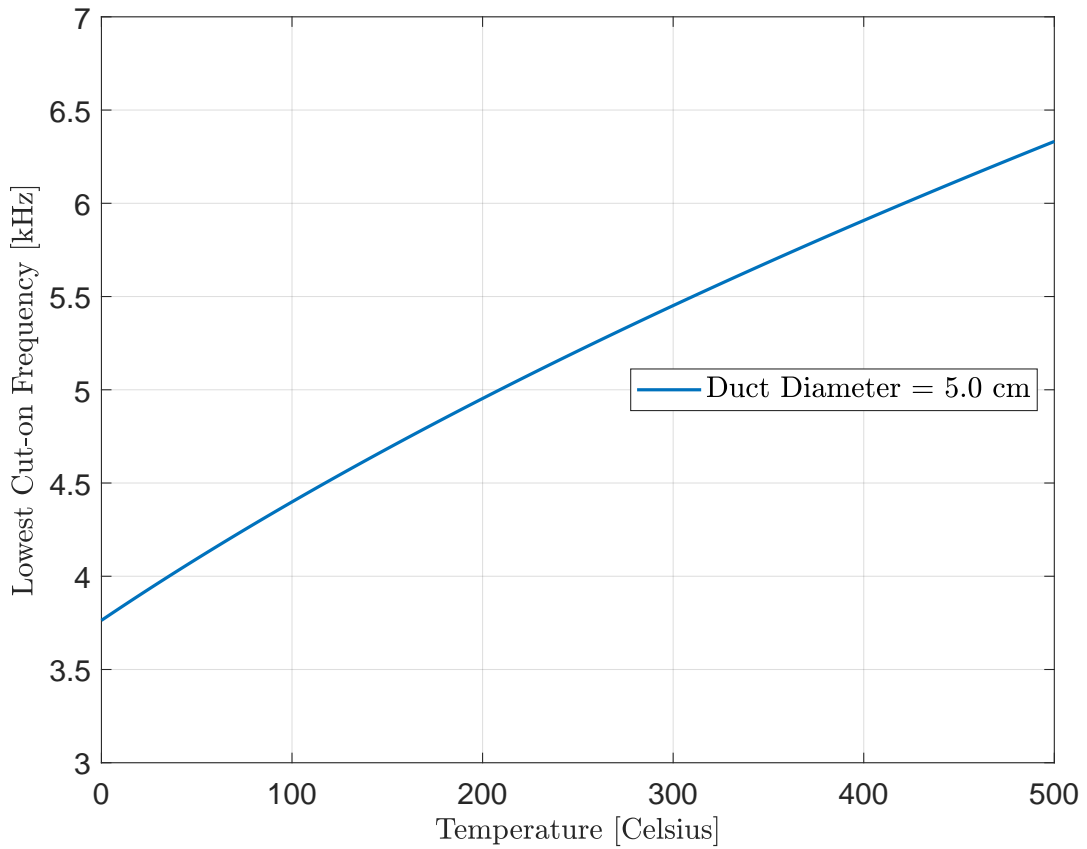


Figure 1: Lowest cut-on frequency for a circular 5 cm diameter duct versus air temperature.

Problem 1e

Question: Are cut-on frequencies higher for a circular or rectangular duct for a given cross-sectional area?

The lowest cut-on frequency is higher for a circular duct than for a rectangular duct for a given cross-sectional area.

For the dimensions given in class, the rectangular duct is not square. This produces a larger dimension and thus a smaller, lowest cut-on frequency. If the rectangular duct is square dimensions on the order of the circular duct diameter with the same cross-sectional area, the cut-on frequencies are approximately equal.

Question: What about in air versus water?

The lowest cut-on frequency is larger for water than for air. The cut-on frequency is proportional to the speed of sound and the speed of sound in water is greater than the speed of sound in air.

Question: What about cold versus hot air?

The lowest cut-on frequency is higher for warm air than it is for cold air.

Problem 2 - Muffler Design Comparison

The Matlab code for this problem is listed in Appendix 2.

For these muffler comparisons, the following assumptions were made:

- There is no flow.
- There are no resistive terms.
- The load impedance was not included because the transmission loss does not require them.
- For Parts b, c, and d, the side branch length offset, L_o , was set to zero.

Problems 2a, 2b, and 2c

Figure 2 shows the transmission loss profiles for a simple expansion chamber, a double-tuned expansion chamber, and a cascaded double-tuned expansion chamber muffler.

The peaks for the simple expansion chamber (red, dashed line) are approximately 22 dB and occur at frequencies with a wavelength that is a quarter of the length of the expansion chamber. Minimal loss occurs at half wavelength multiples.

The addition of the extension tube inside the muffler produces a quarter wavelength resonator. The side branch of Ji (2005; Slide 11, Lecture 3 notes) was used to calculate L_o . For the cascaded double-tuned expansion chamber, the extension tubes produce a secondary quarter wavelength resonator.

As noted in the office hours session, there is no damping which produces artificially high resonances.

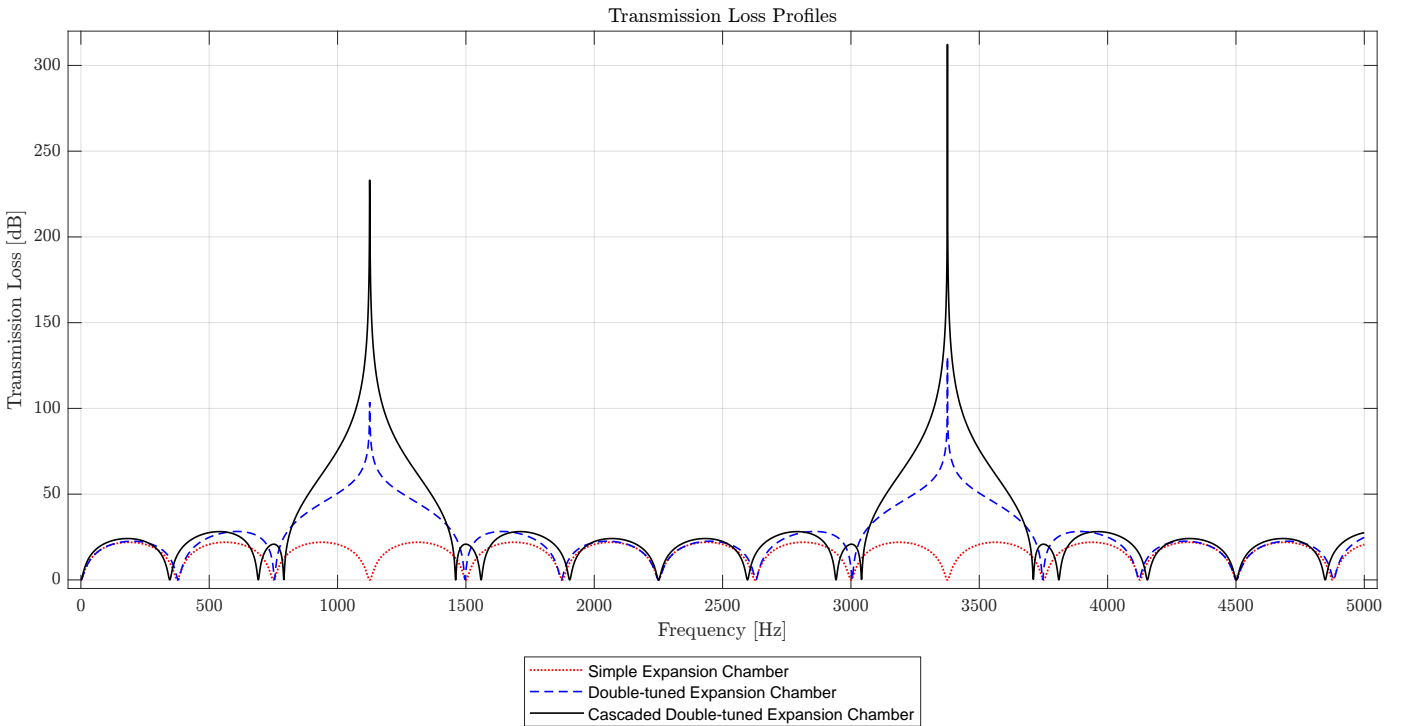


Figure 2: Transmission loss profiles for a simple expansion chamber, a double-tuned expansion chamber, and a cascaded double-tuned expansion chamber mufflers.

Problem 2d

Figure 3 shows the transmission loss profiles for a cascaded double-tuned expansion chamber, and a modified cascaded double-tuned expansion chamber muffler.

Two modifications were made to the original system:

1. The left 3" extension tube in the left chamber was shortened to 2" inches, making the respective muffler section 1" longer.
2. The left 3" extension tube in the right chamber was lengthened to 4", making the respective muffler section 1" shorter.

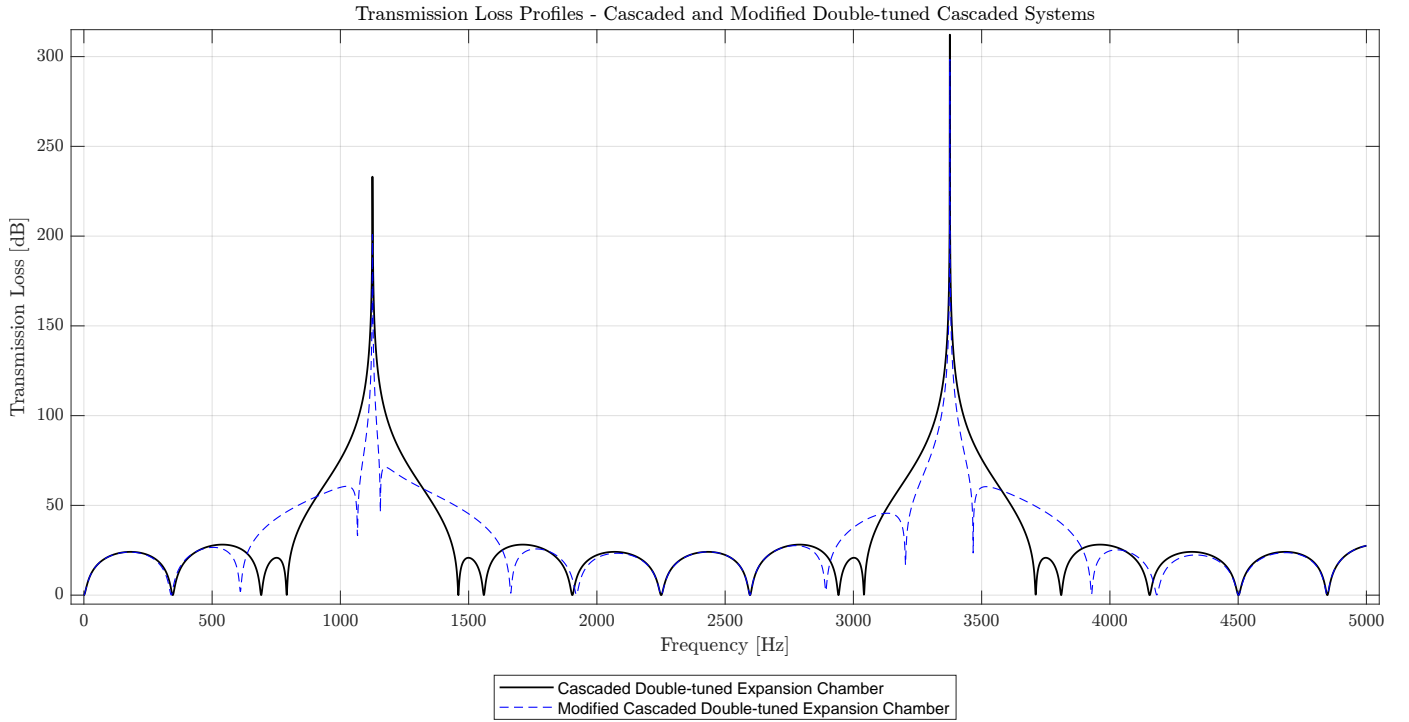


Figure 3: Transmission loss profiles for a cascaded double-tuned expansion chamber muffler and a modified version of this muffler.

These modifications change the symmetry of the cascaded system, and allow the resonate frequencies to be independently changed.

Problem 3 - Bugle Recorder

Problem 3a

The Matlab code for this problem is listed in Appendix 3.

Table 1 lists the length of the pipe section and the mouthpiece.

Item	Length [mm]
Pipe	145
Mouthpiece	90

Table 1: Calculated length of the pipe and length of the mouthpiece.

Problem 3b

Table 2 summarizes the placement of the holes for each note relative to the end of the pipe.

Note	Frequency [Hz]	Length from End of Pipe [mm]
C5	523	n/a
F5	698	87.75
A5	880	138.25
C6	1,046	0.168

Table 2: Hole placement distances.

Figure 4 shows the respective spectrum for each of the bugle recorder notes.

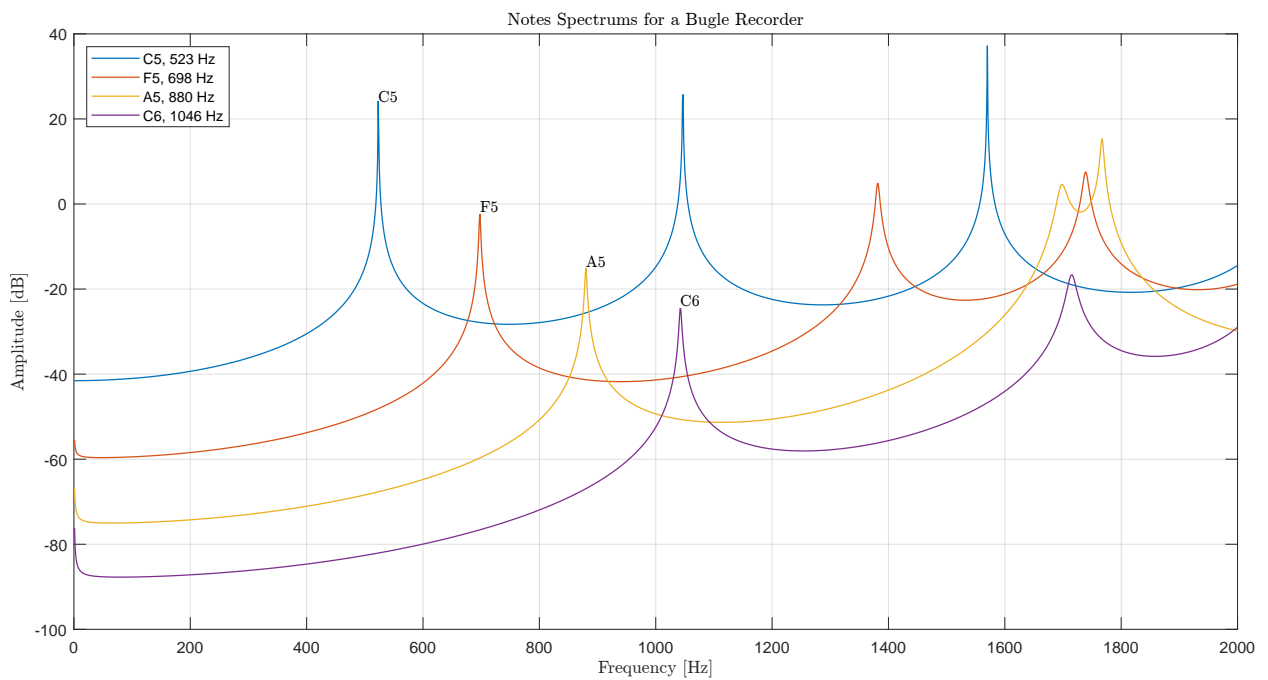


Figure 4: Spectrum for the C5, F5, A5, and C6 bugle recorder notes.

Problem 4 - Intake Duct

The Matlab code for this problem is listed in Appendix 4.

Problem 4a

Table 3 lists the Mach numbers for each pipe section. The flow rate is $0.017462 \frac{\text{m}^3}{\text{s}}$.

Pipe	Area [m ²]	Mach Number [unitless]
Inlet	0.000507	-0.10047
Outlet	0.00811	-0.0062795

Table 3: Calculated Mach numbers.

Problems 4b, 4c, and 4d

Figure 5 shows the transmission loss profiles.

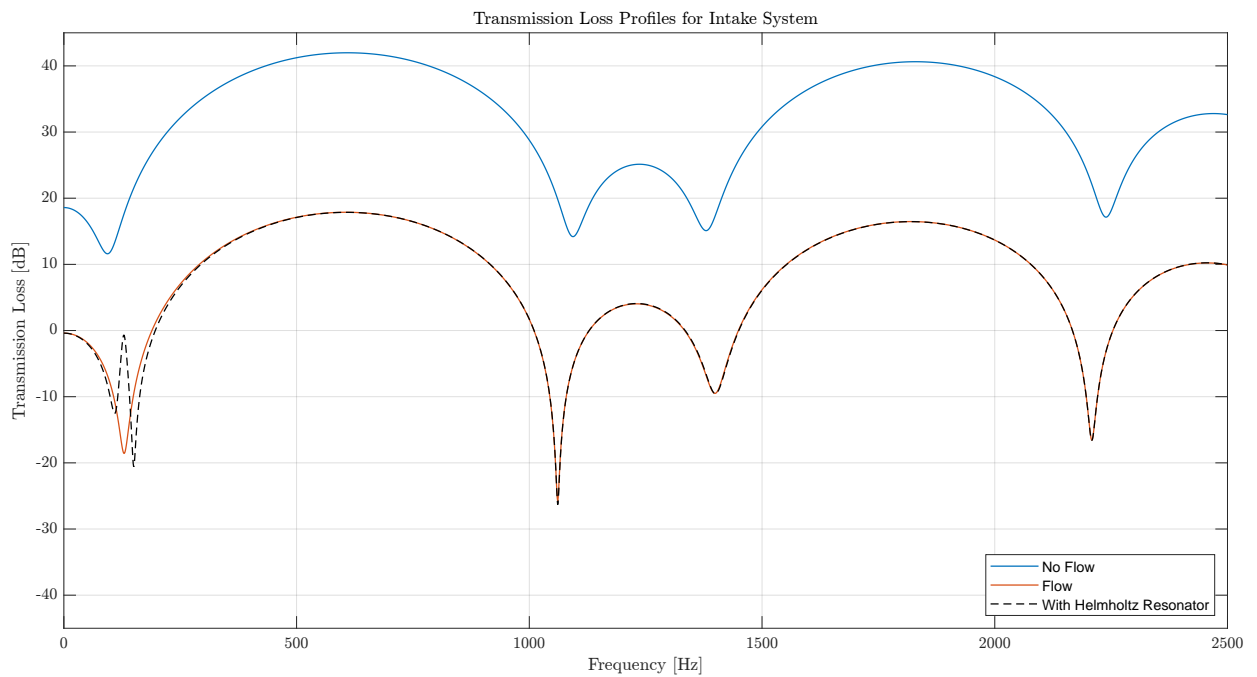


Figure 5: Transmission loss profiles for no flow, flow, and flow with a lossy Helmholtz resonator.

The addition of flow to the intake system introduces a slight phase delay, a lower overall level of loss (approximately 22 dB), and greater loss at the dips. The phase delay is easier to see at respective dips in the loss profile.

A phase delay is introduced by each of the pipe segments by the complex-exponential term in the transfer function. The attenuation is introduced by the expansion transfer function.

The target dip frequency is 129 Hz.

Table 4 lists the dimensions of the lossy Helmholtz resonator.

Item	Measure]
Cavity Diameter	0.254 m
Neck Diameter	0.0254 m
Neck Length	0.127 m
Neck Area	0.5e-3 m ²
Length Correction 1, L _{o1}	0.711 m
Length Correction 2, L _{o2}	0.3 m
Cavity Volume	8.0e-5 m ³
Q Factor	10

Table 4: Dimensions of the lossy Helmholtz resonator (see slide 15 of Lecture 3 notes).

The addition of the Helmholtz resonator offset the attenuation of the dip to approximately 0 dB.

Problem 5 - Intake Duct Silencer

The Matlab code for this problem is listed in Appendix 5.

My respective dip occurs at approximately 1,060 Hz, which has a loss of -25 dB. **For this problem I used the 940 Hz dip noted in the assignment discussion, requiring a 14 dB correction.**

Problem 5a

As noted in the discussion, there are two cases to be considered here.

First, with a liner thickness of 0.0381 meters and a half-liner width (circular duct) of 0.0113 meters, the expansion ratio, m , is 2.7. From Figure 8.37, a 14 dB overall loss has a total liner attenuation of 10 dB. The attenuation rate is about $78.4 \frac{\text{dB}}{\text{m}}$.

Second,

Problem 5b

l , the liner thickness, is 0.0381 m. h , the liner half width, is 0.0113 m.

Problem 5c

The liner thickness ratio is 3.385 [unitless]. The normalized frequency is 0.0617 [unitless] for 940 Hz.

Problem 5d

With a liner thickness ratio of 3.385, the left-side, middle attenuation rate curve is applicable.

With $\frac{1}{h}$ of 4, curve 5 is used

Problem 5e

The flow resistivity, R_1 , is about $1.74\text{e}5 \frac{\text{kg}}{\text{m}^2 \cdot \text{s}}$ or rayl.

1 Appendix - Matlab Code for Problem 1

```
1
2
3
4 %% Synopsis
5
6 % Question 1 – Cut-on Frequencies in Ducts and Pipes
7
8
9
10 %% Environment
11
12 close all; clear; clc;
13 % restoredefaultpath;
14
15 % addpath( genpath( '' ), '-begin' );
16 addpath( genpath( '../40 Assignments/00 Support' ), '-begin' );
17
18 % set( 0, 'DefaultFigurePosition', [ 400 400 900 400 ] ); % [ left bottom width height ]
19 set( 0, 'DefaultFigurePaperPositionMode', 'manual' );
20 set( 0, 'DefaultFigureWindowStyle', 'normal' );
21 set( 0, 'DefaultLineLineWidth', 1.5 );
22 set( 0, 'DefaultTextInterpreter', 'Latex' );
23
24 format ShortG;
25
26 pause( 1 );
27
28 PRINT_FIGURES = 0;
29
30
31
32 %% Define Constants and Anonymous Functions
33
34 c_air = 343; % The speed of sound in air (meters per second).
35 c_water = 1500; % The speed of sound in water (meters per second).
36
37 gamma = 1.4; % The ratio of specific heats [unitless].
38 R = 287; % The gas constant [Joules per (kilogram * Kelvin)].
39
40
41 h_f_cut_on_rectangular_duct = @( c, L ) 0.5 .* c ./ L;
42 %
43 % c – The speed of sound.
44 % L – The largest cross-section dimension of the rectangular duct.
45
46
47 h_f_cut_on_circular_duct = @( c, d ) 0.568 .* c ./ d;
48 %
49 % c – The speed of sound.
50 % L – The diameter of the circular duct.
51
52
53 h_speed_of_sound_in_air = @( gamma, R, temperature_Kelvin ) sqrt( gamma .* R .*
    temperature_Kelvin );
54
55
56
57 %% Problem 1a
58
59 % The cross-sectional dimensions for the rectangular duct are:  $L_x = 12$  cm and  $L_y = 20$  cm.
60
61 % The largest dimension is  $L_y = 20$  cm or 0.2 m.
62
63 % The cut-on frequency is,
64 h_f_cut_on_rectangular_duct( c_air, 0.2 ); % 857.5 Hz (shown in class 858 Hz)
65 fprintf( 1, '\n Problem 1a: The lowest cut-on frequency for the rectangular pipe with air is
    %3.1f Hz.\n', h_f_cut_on_rectangular_duct( c_air, 0.2 ) );
66
67
68
69 %% Problem 1b
70
71 % The cross-sectional dimensions for the rectangular duct are:  $L_x = 12$  cm and  $L_y = 20$  cm.
72
73 % The cross-sectional area of the rectangular duct is  $12$  cm *  $20$  cm =  $240$  cm2 or  $0.024$  m2.
```

```

74 rectangular_duct_cross_sectional_area = 0.12 * 0.20; % 0.024 m^2
75
76 % The diameter of the circular pipe is,
77 circular_duct_diameter = sqrt( 0.024 / pi ) * 2; % 0.17481 meters
78 %
79 % Check:
80 % pi * ( circular_duct_diameter / 2 )^2 CHECKED
81
82
83 % The cut-on frequency for the circular duct is,
84 h_f_cut_on_circular_duct( c_air, circular_duct_diameter ); % 1,114.5 Hz
85 fprintf( 1, '\n Problem 1b: The lowest cut-on frequency for the circular pipe (of equal area
    ) with air is %3.1f Hz.\n', h_f_cut_on_circular_duct( c_air, circular_duct_diameter ) );
86
87
88
89 %% Problem 1c
90
91 % The cut-on frequency for the circular duct with water is,
92 h_f_cut_on_circular_duct( c_water, circular_duct_diameter ); % 4,873.9 Hz
93 fprintf( 1, '\n Problem 1c: The lowest cut-on frequency for the circular pipe (of equal area
    ) with water is %3.1f Hz.\n', h_f_cut_on_circular_duct( c_water, circular_duct_diameter ) );
94
95 % The cut-on frequency should be higher because it is proportional to the
96 % speed of sound in a given medium.
97
98
99
100 %% Problem 1d
101
102 fprintf( 1, '\n Problem 1d: See the figure.\n' );
103
104 temperature_range_celsius = 0:0.1:500; % Celsius
105 temperature_range_kelvin = temperature_range_celsius + 273.15; % Kelvin
106
107
108 FONT_SIZE = 14;
109
110 figure( ); ...
111 plot( temperature_range_celsius, h_f_cut_on_circular_duct( h_speed_of_sound_in_air( gamma, R,
    temperature_range_kelvin ), 0.05 ) ./ 1e3 ); grid on;
112 legend( 'Duct Diameter = 5.0 cm', 'Location', 'East', 'FontSize', FONT_SIZE, 'Interpreter
    ', 'Latex' );
113 set( gca, 'FontSize', FONT_SIZE );
114 %
115 xlabel( 'Temperature [Celsius]', 'FontSize', FONT_SIZE );
116 % xl = get( gca, 'xlabel' ); pxl = get( xl, 'position' ); pxl( 2 ) = 1.1 * pxl( 2 );
117 % set( xl, 'position', pxl );
118 %
119 ylabel( 'Lowest Cut-on Frequency [kHz]', 'FontSize', FONT_SIZE );
120 % yl = get( gca, 'ylabel' ); pyl = get( yl, 'position' ); pyl( 1 ) = 1.2 * pyl( 1 );
121 % set( yl, 'position', pyl );
122 %
123 caption = sprintf( 'Lowest Cut-on Frequency for a Circular Pipe with Air Flow Versus Air
    Temperature\n' );
124 title( caption, 'FontSize', FONT_SIZE );
125 %
126 ylim( [ 3 7 ] );
127
128
129
130 %% Problem 1e
131
132 fprintf( 1, '\n Problem 1e: See Section Problem 1e of the Matlab script for the answers.\n\n' );
133
134
135 % Question: Are cut-on frequencies higher for a circular or rectangular duct for a given cross-
    sectional area?
136
137 % The lowest cut-on frequency is higher for a circular duct than for a
138 % rectangular duct for a given cross-sectional area.
139
140 % For the dimensions given in class, the rectangular duct is not square.
141 % This produces a larger dimension and thus a smaller, lowest cut-on
142 % frequency.
143
144 % If the rectangular duct is square dimensions on the order of the circular
145 % duct diameter with the same cross-sectional area, the the cut-on

```

```

146 % frequencies are approximately equal.
147
148
149 % Question: What about in air versus water?
150
151 % The lowest cut-on frequency is larger with water than air. This due to
152 % the fact that the cut-on frequency is proportional to the speed of sound
153 % and the speed of sound in water is greater than it is in air.
154
155
156 % Question: What about cold versus hot air?
157
158 % For a circular pipe, the cut-on frequency is higher in warm air than cold
159 % air.
160
161
162
163 %% Clean-up
164
165 if ( ~isempty( findobj( 'Type', 'figure' ) ) )
166     monitors = get( 0, 'MonitorPositions' );
167     if ( size( monitors, 1 ) == 1 )
168         autoArrangeFigures( 2, 2, 1 );
169     elseif ( 1 < size( monitors, 1 ) )
170         autoArrangeFigures( 2, 2, 1 );
171     end
172 end
173
174 if ( PRINT_FIGURES == 1 )
175     saveas( gcf, 'Cut-on Frequency Versus Temperature — Sunday, January 19, 2025.pdf' );
176 end
177
178 fprintf( 1, '\n\n*** Processing Complete ***\n\n' );

```

2 Appendix - Matlab Code for Problem 2

```
1
2
3
4 %% Synopsis
5
6 % Question 2 – Muffler Design Comparison
7
8
9
10 %% Environment
11
12 close all; clear; clc;
13 % restoredefaultpath;
14
15 % addpath( genpath( '' ), '-begin' );
16 addpath( genpath( '../00 Support' ), '-begin' );
17
18 % set( 0, 'DefaultFigurePosition', [ 400 400 900 400 ] ); % [ left bottom width height ]
19 set( 0, 'DefaultFigurePaperPositionMode', 'manual' );
20 set( 0, 'DefaultFigureWindowStyle', 'normal' );
21 set( 0, 'DefaultLineLineWidth', 1.5 );
22 set( 0, 'DefaultTextInterpreter', 'Latex' );
23
24 format ShortG;
25
26 pause( 1 );
27
28 PRINT_FIGURES = 0;
29
30
31
32 %% Constants
33
34 rho0 = 1.21; % Air density (kg per m^3).
35 c = 343; % Speed of sound in air (meters per second).
36
37 frequency_set = 0:1:5e3; % Hertz
38
39
40
41 %% Dimensions
42
43 convert.inches_to_meters = 0.0254;
44 convert.foot_to_meters = 0.3048;
45
46 dimensions.inlet_diameter_meters = 2 * convert.inches_to_meters; % 0.0508 meters
47 dimensions.inlet_length_meters = 6 * convert.foot_to_meters; % 1.82 meters
48
49 dimensions.muffler_diameter_meters = 10 * convert.inches_to_meters; % 0.254 meters
50 dimensions.muffler_length_meters = 18 * convert.inches_to_meters; % 0.4572 meters
51
52 dimensions.outlet_diameter_meters = 2 * convert.inches_to_meters; % 0.0508 meters
53 dimensions.outlet_length_meters = 1 * convert.foot_to_meters; % 0.3048 meters
54
55 outlet_flanged = false;
56
57 dimensions.overhang = 3 * convert.inches_to_meters; % 0.0762 meters
58
59 segment_diameters = [ ...
60     dimensions.outlet_diameter_meters, ...
61     dimensions.muffler_diameter_meters, ...
62     dimensions.inlet_diameter_meters, ...
63 ].';
64 %
65 h_area_from_diameter = @( d ) pi .* d.^2 ./ 4;
66 %
67 segment_areas = h_area_from_diameter( segment_diameters );
68
69 segment_lengths = [ ...
70     dimensions.outlet_length_meters, ...
71     dimensions.muffler_length_meters, ...
72     dimensions.inlet_length_meters, ...
73     dimensions.overhang, ...
74 ].';
75
```

```

76
77
78 %% Part a - Simple Expansion Chamber
79
80 nFreq = length( frequency_set );
81 TL = zeros( nFreq, 1 );
82
83 for frequency_index = 1:1:nFreq
84
85     f = frequency_set( frequency_index );
86
87     T_total = [ 1 0; 0 1 ];
88
89     T1 = duct_segment_transfer_matrix( f, rho0, c, 0.3048, 0.0020268 ); % Duct - Outlet
90     T2 = duct_segment_transfer_matrix( f, rho0, c, 0.4572, 0.050671 ); % Duct
91     T3 = duct_segment_transfer_matrix( f, rho0, c, 1.8288, 0.0020268 ); % Duct - Inlet
92
93     T_net = T3 * T2 * T1 * T_total;
94     % T_net = T_inlet * T_total; % Zero transmission loss for a straight duct.
95
96     T11 = T_net(1, 1); T12 = T_net(1, 2); T21 = T_net(2, 1); T22 = T_net(2, 2);
97     TL( frequency_index ) = 10 * log10( abs( ( T11 + 0.0020268*T12/(rho0*c) + (rho0*c)*
98         T21/0.0020268 + T22 ) / 2 )^2 );
99     %
100     % The transmission loss calculation does not require a load impedance.
101 end
102
103 TL_parta = TL; % The maximum peak value should be about 22 (21.952) dB.
104 %
105 % Expected behaviour:
106 %
107 % 1.) 0 dB at 0 Hz.
108 % 2.) The transmission loss of a straight duct section is zero; energy out equals energy in.
109
110 max( TL_parta ); % 22 dB
111
112
113
114 %% Part b - Double-tuned Expansion Chamber
115
116 annulus_area_squared_meters = pi/4 * ( 0.254^2 - 0.0508^2 );
117
118 L_o = 0; % Assume that the L_o extension is negligible.
119
120
121 nFreq = length( frequency_set );
122 TL = zeros( nFreq, 1 );
123
124 for frequency_index = 1:1:nFreq
125
126     f = frequency_set( frequency_index );
127
128     T_total = [ 1 0; 0 1 ];
129
130     T1 = duct_segment_transfer_matrix( f, rho0, c, (0.3048 + 0.0762), 0.0020268 );
131     T3 = duct_segment_transfer_matrix( f, rho0, c, (0.4572 - 2*0.0762), 0.050671 );
132     T5 = duct_segment_transfer_matrix( f, rho0, c, (1.8288 + 0.0762), 0.0020268 );
133
134     k = 2*pi*f/c;
135     Z_A = -1j*rho0*c/annulus_area_squared_meters*cot( k * ( 0.0762 + L_o ) );
136     T2 = [ 1 0; 1/Z_A 1 ];
137     T4 = T2;
138
139     T_net = T5 * T4 * T3 * T2 * T1 * T_total;
140
141
142     T11 = T_net(1, 1); T12 = T_net(1, 2); T21 = T_net(2, 1); T22 = T_net(2, 2);
143     TL( frequency_index ) = 10 * log10( abs( ( T11 + 0.0020268*T12/(rho0*c) + (rho0*c)*
144         T21/0.0020268 + T22 ) / 2 )^2 );
145     %
146     % The transmission loss calculation does not require a load impedance.
147 end
148
149 TL_partb = TL;
150 %
151 % Expected behaviour:

```

```

152 %
153 % 1.) 0 dB at 0 Hz.
154 % 2.) 0 dB at same locations as a simple expansion chamber.
155 % 3.) Peaks at 1,125 Hz and 3,376 Hz;
156
157 % Frequency at which the quarter-wavelength is 0.0762 meters.
158 %  $343 / (4 * 0.0762)$ ; % 1,125 Hz.
159
160 % Also work at three-quarter-wavelength.
161 %  $3 * 1125$ ; % 3,375 Hz
162
163
164
165 %% Part c - Cascaded, Double-tuned Expansion Chamber
166
167 annulus_area_squared_meters = pi/4 * ( 0.254^2 - 0.0508^2 );
168
169 L_o = 0; % Assume that the L_o extension is negligible.
170
171
172 nFreq = length( frequency_set );
173 TL = zeros( nFreq, 1 );
174
175 for frequency_index = 1:1:nFreq
176
177     f = frequency_set( frequency_index );
178     k = 2*pi*f/c;
179     Z_A = -lj*rho0*c/annulus_area_squared_meters*cot(k * ( 0.0762 + L_o ) );
180
181     T_total = [ 1 0; 0 1 ];
182
183     T1 = duct_segment_transfer_matrix( f, rho0, c, (0.3048 + 0.0762), 0.0020268 ); % Duct -
184     % Outlet
185     T2 = [ 1 0; 1/Z_A 1 ]; % Straight Side Branch
186     T3 = duct_segment_transfer_matrix( f, rho0, c, (0.2286 - 2*0.0762), 0.050671 ); % Duct
187     T4 = T2; % Straight Side Branch
188     T5 = duct_segment_transfer_matrix( f, rho0, c, 2*0.0762, 0.050671 ); % Duct
189     T6 = T2; % Straight Side Branch
190     T7 = T3; % Duct
191     T8 = T2; % Straight Side Branch
192     T9 = duct_segment_transfer_matrix( f, rho0, c, (1.8288 + 0.0762), 0.0020268 ); % Duct -
193     % Inlet
194
195     T_net = T9 * T8 * T7 * T6 * T5 * T4 * T3 * T2 * T1 * T_total;
196
197     T11 = T_net(1, 1); T12 = T_net(1, 2); T21 = T_net(2, 1); T22 = T_net(2, 2);
198     TL( frequency_index ) = 10 * log10( abs( ( T11 + 0.0020268*T12/(rho0*c) + (rho0*c)*
199     T21/0.0020268 + T22 ) / 2 )^2 );
200
201 end
202
203 TL_partc = TL;
204
205
206 %% Part d - Cascaded, Double-tuned Expansion Chamber
207
208 nFreq = length( frequency_set );
209 TL = zeros( nFreq, 1 );
210
211 for frequency_index = 1:1:nFreq
212
213     f = frequency_set( frequency_index );
214     k = 2*pi*f/c;
215     Z_A = -lj*rho0*c/annulus_area_squared_meters*cot(k * ( 0.0762 + L_o ) );
216
217     T_total = [ 1 0; 0 1 ];
218
219     T1 = duct_segment_transfer_matrix( f, rho0, c, (0.3048 + 0.0762), 0.0020268 ); % Duct -
220     % Outlet
221     T2 = [ 1 0; 1/Z_A 1 ]; % Straight Side Branch
222     T3 = duct_segment_transfer_matrix( f, rho0, c, 0.0076, 0.050671 ); % Duct
223     T4 = T2; % Straight Side Branch
224     T5 = duct_segment_transfer_matrix( f, rho0, c, 2*0.0762, 0.050671 ); % Duct
225     T6 = T2; % Straight Side Branch
226     T7 = duct_segment_transfer_matrix( f, rho0, c, (0.29718 - 2*0.0762), 0.050671 ); % Duct
227     T8 = T2; % Straight Side Branch
228     T9 = duct_segment_transfer_matrix( f, rho0, c, (1.8288 + 0.0762), 0.0020268 ); % Duct -

```

```

226 Inlet
227 T_net = T9 * T8 * T7 * T6 * T5 * T4 * T3 * T2 * T1 * T_total;
228
229 T11 = T_net(1, 1); T12 = T_net(1, 2); T21 = T_net(2, 1); T22 = T_net(2, 2);
230 TL( frequency_index ) = 10 * log10( abs( ( T11 + 0.0020268*T12/(rho0*c) + (rho0*c)*
231 T21/0.0020268 + T22 ) / 2 )^2 );
232 end
233
234 TL_partd = TL;
235
236
237
238 %% Plot Transmission Loss Profiles
239
240 Y_LIMITS = [ -5 320 ];
241
242 h_figure_1 = figure( ); ...
243 plot( frequency_set, TL_parta, 'LineWidth', 1.0, 'LineStyle', ':', 'Color', 'r' ); hold on;
244 plot( frequency_set, TL_partb, 'LineWidth', 0.9, 'LineStyle', '—', 'Color', 'b' );
245 plot( frequency_set, TL_partc, 'LineWidth', 0.9, 'LineStyle', '—', 'Color', 'k' ); grid on;
246 legend( ...
247     'Simple Expansion Chamber', ...
248     'Double-tuned Expansion Chamber', ...
249     'Cascaded Double-tuned Expansion Chamber', ...
250     'Location', 'SouthOutside' );
251 xlabel( 'Frequency [Hz]' ); ylabel( 'Transmission Loss [dB]' );
252 title( 'Transmission Loss Profiles' );
253 %
254 Ax = gca;
255 Ax.XAxis.TickLabelInterpreter = 'latex';
256 Ax.YAxis.TickLabelInterpreter = 'latex';
257 %
258 axis( [ -50 5e3+50 Y_LIMITS ] );
259
260
261 Y_LIMITS = [ -5 315 ];
262
263 h_figure_2 = figure( ); ...
264 plot( frequency_set, TL_partc, 'LineWidth', 1.0, 'LineStyle', '—', 'Color', 'k' ); hold on;
265 plot( frequency_set, TL_partd, 'LineWidth', 0.6, 'LineStyle', '—', 'Color', 'b' ); grid on;
266 legend( ...
267     'Cascaded Double-tuned Expansion Chamber', ...
268     'Modified Cascaded Double-tuned Expansion Chamber', ...
269     'Location', 'SouthOutside' );
270 xlabel( 'Frequency [Hz]' ); ylabel( 'Transmission Loss [dB]' );
271 title( 'Transmission Loss Profiles – Cascaded and Modified Double-tuned Cascaded Systems' );
272 %
273 Ax = gca;
274 Ax.XAxis.TickLabelInterpreter = 'latex';
275 Ax.YAxis.TickLabelInterpreter = 'latex';
276 %
277 axis( [ -50 5e3+50 Y_LIMITS ] );
278
279
280
281 %% Clean-up
282
283 if ( ~isempty( findobj( 'Type', 'figure' ) ) )
284     monitors = get( 0, 'MonitorPositions' );
285     if ( size( monitors, 1 ) == 1 )
286         autoArrangeFigures( 2, 2, 1 );
287     elseif ( 1 < size( monitors, 1 ) )
288         autoArrangeFigures( 2, 2, 1 );
289     end
290 end
291
292 if ( PRINT_FIGURES == 1 )
293     exportgraphics( h_figure_1, 'Assignment 1 – Question 2 Figure All TL Profiles.pdf', '
Append', true );
294     exportgraphics( h_figure_2, 'Assignment 1 – Question 2 Figure Comparison TL Plot For
Cascaded Systems.pdf', 'Append', true );
295 end
296
297 fprintf( 1, '\n\n\n*** Processing Complete ***\n\n\n' );

```


3 Appendix - Matlab Code for Problem 3

```
1
2
3
4 %% Synopsis
5
6 % Question 3 – Bugle Recorder
7
8
9
10 %% Environment
11
12 close all; clear; clc;
13 % restoredefaultpath;
14
15 % addpath( genpath( '' ), '-begin' );
16 addpath( genpath( '../00 Support' ), '-begin' );
17
18 % set( 0, 'DefaultFigurePosition', [ 400 400 900 400 ] ); % [ left bottom width height ]
19 set( 0, 'DefaultFigurePaperPositionMode', 'manual' );
20 set( 0, 'DefaultFigureWindowStyle', 'normal' );
21 set( 0, 'DefaultLineLineWidth', 0.8 );
22 set( 0, 'DefaultTextInterpreter', 'Latex' );
23
24 format ShortG;
25
26 pause( 1 );
27
28 PRINT_FIGURES = 0;
29
30
31
32 %% Constants and Anonymous Functions
33
34 rho0 = 1.21; % Density of air (kg per cubic-meter).
35 c = 343; % Speed of sound in air (meters per second).
36
37 h_RA_term_1 = @( rho0, c, S, k, delta_mu, D, w ) ( rho0*c/S ) * ( ( k * sqrt( (2*3.178e-5) / (
    rho0*w ) ) * D * 0.004 ) / (2*S) * 1.4364 );
38 h_RA_term_2 = @( rho0, c, S, k, delta_mu, D, w, h ) ( rho0*c/S ) * 0.288*k*3.178e-5*log10((4*
    S)/(pi*h^2));
39 h_RA_term_3 = @( rho0, c, S, k, delta_mu, D, w, h ) ( rho0*c/S ) * (0.5*S*k^2)/(2*pi);
40 %
41 % See Equation 8.34 on page 479 of Bies et al (2024).
42
43
44
45 %% Part a
46
47 % The estimated total length of the recorder is 0.325 meters.
48
49 % Estimation of hole location was done by trial-and-error.
50
51 pipe_net_length = 0.325;
52 pipe_area = pi*0.009^2/4;
53
54 flanged = false;
55
56
57 frequency_set = 1:1:2e3;
58
59 nFreq = length( frequency_set );
60 A = zeros( nFreq, 1 );
61
62 for frequency_index = 1:1:nFreq
63
64     f = frequency_set( frequency_index );
65
66     T_total = [ 1 0; 0 1 ];
67
68     T_segment = duct_segment_transfer_matrix( f, rho0, c, pipe_net_length, pipe_area );
69
70     T_total = T_segment * T_total;
71
72     Z = open_end_impedance( f, rho0, c, 0, pipe_area, flanged );
73
```

```

74     T11 = T_total(1, 1); T12 = T_total(1, 2);
75     A( frequency_index ) = -10*log10( abs( T11 + T12 / Z )^2 );
76
77 end
78
79 A_parta = A; clear A;
80
81 % figure( ); ...
82 % plot( frequency_set, A_parta, 'LineWidth', 0.8 ); grid on;
83 % xlabel( 'Frequency [Hz]' ); ylabel( 'Amplitude [dB]' );
84 % legend( 'C5' );
85
86 % return
87
88 %% Part b
89
90 epsilon = 0.006 / 0.004; % Diameter of the hold divided by diameter of pipe section (1.5).
91
92 switch ( 3 )
93
94     case 0 % Original Value
95
96         L_o = 0.00001; % Estimate
97         L_e = 0.004 + 2*L_o;
98
99         fprintf( 1, '\nZero - Percentage change in pipe thickness: %3.1f%%\n\n', ( L_e - 0.004
100 ) / 0.004 * 100 );
101
102     case 1 % Ingard (2010)
103
104         a = 0.006 / 2;
105
106         L_o = ( 0.6*a + 0.85*a ) / 2;
107         L_e = 0.004 + 2*L_o;
108
109         fprintf( 1, '\nIngard (2010) - Percentage change in pipe thickness: %3.1f%%\n\n', ( L_e
110 - 0.004 ) / 0.004 * 100 );
111
112     case 2 % Kurze and Riedel (2013)
113
114         e = epsilon^2;
115
116         a = 0.006 / 2;
117
118         L_o = pi*a*( 1 - 1.47*e^0.5 + 0.47*e^1.5 );
119         L_e = 0.004 + 2*L_o;
120
121         fprintf( 1, '\nKurze and Riedel (2013) - Percentage change in pipe thickness: %3.1f%%\n
122 \n', ( L_e - 0.004 ) / 0.004 * 100 );
123
124     case 3 % Ji (2005)
125
126         a = 0.006 / 2;
127
128         L_o = a*( 0.9326 - 0.6196*epsilon );
129         L_e = 0.004 + 2*L_o;
130
131         fprintf( 1, '\nJi (2005) - Percentage change in pipe thickness: %3.1f%%\n\n', ( L_e -
132 0.004 ) / 0.004 * 100 );
133
134     otherwise
135         error ( '*** Invalid SWITCH Index ***' );
136
137 end
138
139 duct_lengths = 0.235/4 * ones( 4, 1 ); % 523 Hz, all holes covered.
140
141 frequency_set = 0:1:2e3;
142
143 nFreq = length( frequency_set );
144 A = zeros( nFreq, 1 );
145
146 for frequency_index = 1:1:nFreq
147     f = frequency_set( frequency_index );

```

```

148 T_total = [ 1 0; 0 1 ];
149
150
151 Z_A = 1j * rho0 * (2 * pi * f) * L_e / ( pi*0.006^2/4 );
152 %
153 term_1 = h_RA_term_1( rho0, c, pi*(0.006)^2/4, 2*pi*f/c, sqrt( (2 * 3.178e-5) / ( 2*pi*f *
rho0 ) ), pi * 0.006, 2*pi*f );
154 term_2 = h_RA_term_2( rho0, c, pi*(0.006)^2/4, 2*pi*f/c, sqrt( (2 * 3.178e-5) / ( 2*pi*f *
rho0 ) ), pi * 0.006, 2*pi*f, 0.3 );
155 term_3 = h_RA_term_3( rho0, c, pi*(0.006)^2/4, 2*pi*f/c, sqrt( (2 * 3.178e-5) / ( 2*pi*f *
rho0 ) ), pi * 0.006, 2*pi*f, 0.3 );
156 R_A = term_1 + term_2 + term_3;
157 %
158 Z_A = Z_A + R_A;
159 T_Hole = [ 1 0; 1/Z_A 1 ];
160
161
162 switch ( 0 )
163
164     case 0 % All holes covered. 523 Hz
165         T1 = duct_segment_transfer_matrix( f, rho0, c, 0.05875, pipe_area ); % Duct - Outlet
166         T2 = [ 1 0; 0 1 ]; % Hole
167         T3 = duct_segment_transfer_matrix( f, rho0, c, 0.05875, pipe_area ); % Duct
168         T4 = [ 1 0; 0 1 ]; % Hole
169         T5 = duct_segment_transfer_matrix( f, rho0, c, 0.05875, pipe_area ); % Duct
170         T6 = [ 1 0; 0 1 ]; % Hole
171         T7 = duct_segment_transfer_matrix( f, rho0, c, 0.05875, pipe_area ); % Duct
172
173         % Check
174         % 4 * 0.05875 = 0.235;
175
176     case 1 % Hole 1 uncovered. 698 Hz
177         T1 = duct_segment_transfer_matrix( f, rho0, c, 0.08775, pipe_area ); % Duct - Outlet
178         T2 = T_Hole;
179         T3 = duct_segment_transfer_matrix( f, rho0, c, 0.02975, pipe_area ); % Duct
180         T4 = [ 1 0; 0 1 ]; % Hole
181         T5 = duct_segment_transfer_matrix( f, rho0, c, 0.05875, pipe_area ); % Duct
182         T6 = [ 1 0; 0 1 ]; % Hole
183         T7 = duct_segment_transfer_matrix( f, rho0, c, 0.05875, pipe_area ); % Duct
184
185         % Check
186         % 0.08775 + 0.02975 + 2*0.05875 = 0.235;
187
188     case 2 % Hole 2 uncovered. 880 Hz
189
190         offset = 0.00625; % 874
191
192         T1 = duct_segment_transfer_matrix( f, rho0, c, 0.08775, pipe_area ); % Duct - Outlet
193         T2 = T_Hole;
194         T3 = duct_segment_transfer_matrix( f, rho0, c, 0.0505, pipe_area ); % Duct
195         T4 = T_Hole;
196         T5 = duct_segment_transfer_matrix( f, rho0, c, 0.038, pipe_area ); % Duct
197         T6 = [ 1 0; 0 1 ]; % Hole
198         T7 = duct_segment_transfer_matrix( f, rho0, c, 0.05875, pipe_area ); % Duct
199
200         % Check
201         % 0.08775 + 0.0505 + 0.038 + 0.05875 = 0.235
202
203     case 3 % Hole 3 uncovered. 1,046 Hz
204
205         T1 = duct_segment_transfer_matrix( f, rho0, c, 0.08775, pipe_area ); % Duct - Outlet
206         T2 = T_Hole;
207         T3 = duct_segment_transfer_matrix( f, rho0, c, 0.0505, pipe_area ); % Duct
208         T4 = T_Hole;
209         T5 = duct_segment_transfer_matrix( f, rho0, c, 0.02975, pipe_area ); % Duct
210         T6 = T_Hole;
211         T7 = duct_segment_transfer_matrix( f, rho0, c, 0.067, pipe_area ); % Duct
212
213         % Check
214         % 0.08775 + 0.0505 + 0.02975 + 0.067 = 0.235 m
215
216 end
217
218 T8 = duct_segment_transfer_matrix( f, rho0, c, 0.09, pipe_area ); % Duct - Inlet
219
220 T_total = T8 * T7 * T6 * T5 * T4 * T3 * T2 * T1 * T_total;
221
222 Z = open_end_impedance( f, rho0, c, 0, pipe_area, flanged );

```

```

223
224         T11 = T_total(1, 1); T12 = T_total(1, 2);
225         A( frequency_index ) = -10*log10( abs( T11 + T12 / Z )^2 );
226
227     end
228
229
230 % figure( ); ...
231 %     plot( frequency_set, A ); grid on;
232 %     xlabel( 'Frequency [Hz]' ); ylabel( 'Amplitude [dB]' );
233
234
235
236 %% Plot Note Set
237
238 load( 'A_C5_Data.mat' ); % Variable(s): A_C5
239 load( 'A_F5_Data.mat' ); % Variable(s): A_F5
240 load( 'A_A5_Data.mat' ); % Variable(s): A_A5
241 load( 'A_C6_Data.mat' ); % Variable(s): A_C6
242
243
244 h_figure_1 = figure( ); ...
245     plot( frequency_set, A_C5 ); hold on;
246     text( 523, 25, 'C5' );
247     plot( frequency_set, A_F5 );
248     text( 698, -1, 'F5' );
249     plot( frequency_set, A_A5 );
250     text( 880, -14, 'A5' );
251     plot( frequency_set, A_C6 ); grid on;
252     text( 1042, -23, 'C6' );
253     %
254     legend( 'C5, 523 Hz', 'F5, 698 Hz', 'A5, 880 Hz', 'C6, 1046 Hz', 'Location', 'NorthWest'
255 );
256 xlabel( 'Frequency [Hz]' ); ylabel( 'Amplitude [dB]' );
257 title( 'Notes Spectrums for a Bugle Recorder' );
258
259
260 %% Clean-up
261
262 if ( ~isempty( findobj( 'Type', 'figure' ) ) )
263     monitors = get( 0, 'MonitorPositions' );
264     if ( size( monitors, 1 ) == 1 )
265         autoArrangeFigures( 2, 2, 1 );
266     elseif ( 1 < size( monitors, 1 ) )
267         autoArrangeFigures( 2, 2, 1 );
268     end
269 end
270
271 if ( PRINT_FIGURES == 1 )
272     exportgraphics( h_figure_1, 'Assignment 1 - Question 3 Bugle Recorder Note Spectrums.pdf'
273 , 'Append', true );
274 end
275
276 fprintf( 1, '\n\n*** Processing Complete ***\n\n' );
277
278
279 %% Reference(s)

```

4 Appendix - Matlab Code for Problem 4

```
1
2
3
4 %% Synopsis
5
6 % Question 4 – Intake Duct
7
8
9
10 %% Environment
11
12 close all; clear; clc;
13 % restoredefaultpath;
14
15 % addpath( genpath( '' ), '-begin' );
16 addpath( genpath( '../40 Assignments/00 Support' ), '-begin' );
17
18 % set( 0, 'DefaultFigurePosition', [ 400 400 900 400 ] ); % [ left bottom width height ]
19 set( 0, 'DefaultFigurePaperPositionMode', 'manual' );
20 set( 0, 'DefaultFigureWindowStyle', 'normal' );
21 set( 0, 'DefaultLineLineWidth', 0.8 );
22 set( 0, 'DefaultTextInterpreter', 'Latex' );
23
24 format ShortG;
25
26 pause( 1 );
27
28 PRINT_FIGURES = 0;
29
30
31
32 %% Constants
33
34 rho0 = 1.21; % Density of air (kg per cubic-meter).
35 c = 343; % Speed of sound in air (meters per second).
36
37 h_area = @( diameter ) pi * diameter^2 / 4;
38
39
40
41 %% Define Shape
42
43 % Source
44 duct_1.diameter_meters = 0.0254; % meters
45 duct_1.area = h_area( duct_1.diameter_meters ); % squared-meters
46 duct_1.length_meters = 0.1524; % meters
47
48
49 % Outlet
50 duct_2.diameter_meters = 0.1016; % 4 inches
51 duct_2.area = h_area( duct_2.diameter_meters );
52 duct_2.length_meters = 0.127; % 5 inches
53
54 % Flanged.
55
56 flow_rate_cubic_meters_per_second = 1.04772 / 60; % or 37 cubic-feet per minute
57
58
59
60 %% Part a – Mach Numbers
61
62 % Calculate the Mach number of the flow in both pip sections.
63
64 duct_1.Mach = -1.0 * flow_rate_cubic_meters_per_second / ( (pi * duct_1.diameter_meters^2 / 4 ) *
65     c ); % 0.100 unitless
66
67 duct_2.Mach = -1.0 * flow_rate_cubic_meters_per_second / ( (pi * duct_2.diameter_meters^2 / 4 ) *
68     c ); % 0.00628 unitless
69
70
71 %% Part b – No Flow in Intake System
72
73 % The flange is not considered because transmission loss is calculated.
```

```

74 frequency_set = 0:1:2.5e3;
75 nFreq = length( frequency_set );
76 TL = zeros( nFreq, 1 );
77
78 for frequency_index = 1:1:nFreq
79
80     f = frequency_set( frequency_index );
81
82     T_total = [ 1 0; 0 1 ];
83
84     T1 = duct_segment_transfer_matrix( f, rho0, c, duct_2.length_meters, duct_2.area );
85     T2 = [ 1 0; 0 duct_2.area/duct_1.area ];
86     T3 = duct_segment_transfer_matrix( f, rho0, c, duct_1.length_meters, duct_1.area );
87
88     T_net = T3 * T2 * T1 * T_total;
89
90     T11 = T_net(1, 1); T12 = T_net(1, 2); T21 = T_net(2, 1); T22 = T_net(2, 2);
91     TL( frequency_index ) = 10 * log10( abs( ( T11 + duct_1.area*T12/(rho0*c) + (rho0*c)*
92     T21/duct_2.area + T22 ) / 2 )^2 );
93 end
94
95 TL_part_b = TL;
96
97
98
99


---


100 %% Part c - Flow in Intake System
101
102 % The flange is not considered because transmission loss is calculated.
103
104
105 frequency_set = 0:1:2.5e3;
106 nFreq = length( frequency_set );
107 TL = zeros( nFreq, 1 );
108
109
110 for frequency_index = 1:1:nFreq
111
112     f = frequency_set( frequency_index );
113
114     T_total = [ 1 0; 0 1 ];
115
116     T1 = duct_segment_transfer_matrix_flow( f, rho0, c, duct_2.length_meters, duct_2.area, duct_2
117     .Mach );
118     T2 = duct_expansion_connection_transfer_matrix( rho0, c, duct_2.area, duct_1.area, duct_1.
119     Mach );
120     T3 = duct_segment_transfer_matrix_flow( f, rho0, c, duct_1.length_meters, duct_1.area, duct_1
121     .Mach );
122
123     T_net = T3 * T2 * T1 * T_total;
124
125     T11 = T_net(1, 1); T12 = T_net(1, 2); T21 = T_net(2, 1); T22 = T_net(2, 2);
126     TL( frequency_index ) = 10 * log10( abs( ( T11 + duct_1.area*T12/(rho0*c) + (rho0*c)*
127     T21/duct_2.area + T22 ) / 2 )^2 );
128 end
129
130 TL_part_c = TL;
131
132
133


---


134 %% Part d - Add a Lossy Helmholtz Resonator
135
136 % Use volume to tune resonator.
137
138
139 w_o = 2*pi*129; % Target resonate frequency; estimated from plot.
140
141 d_cavity = 10*0.0254; % meters
142 d_neck = 0.0254; % meters
143 L_neck = 5*0.0254; % meters
144
145 S = d_neck^2*pi / 4;
146
147 Lo1 = 0.82 * ( 1 - 1.33*( d_neck / d_cavity ) );

```

```

147 Lo2 = 0.3;
148 Le = L_neck + Lo1 + Lo2;
149
150 V = S / ( (w_o/c)^2 * Le );
151
152
153 R_A = rho0*c / 10 * sqrt( Le / ( S * V ) );
154
155 h_Z_A = @( f, Le, S, V, R_A ) 1j*rho0*2*pi*f*Le/S - 1j*rho0*c^2/(V*2*pi*f) + R_A;
156
157
158 frequency_set = 0:1:2.5e3;
159 nFreq = length( frequency_set );
160 TL = zeros( nFreq, 1 );
161
162 for frequency_index = 1:1:nFreq
163
164     f = frequency_set( frequency_index );
165
166     Z_A = h_Z_A( f, Le, S, V, R_A );
167
168     T_total = [ 1 0; 0 1 ];
169
170     T1 = duct_segment_transfer_matrix_flow( f, rho0, c, duct_2.length_meters, duct_2.area, duct_2
171     .Mach );
172     T2 = duct_expansion_connection_transfer_matrix( rho0, c, duct_2.area, duct_1.area, duct_1.
173     Mach );
174     T3 = [ 1 0; 1/Z_A 1 ];
175     T4 = duct_segment_transfer_matrix_flow( f, rho0, c, duct_1.length_meters, duct_1.area, duct_1
176     .Mach );
177
178     T_net = T4 * T3 * T2 * T1 * T_total;
179
180     T11 = T_net(1, 1); T12 = T_net(1, 2); T21 = T_net(2, 1); T22 = T_net(2, 2);
181     TL( frequency_index ) = 10 * log10( abs( ( T11 + duct_1.area*T12/(rho0*c) + (rho0*c)*
182     T21/duct_2.area + T22 ) / 2 )^2 );
183
184 end
185
186 TL_part_d = TL;
187
188
189 %% Plot
190
191 figure( ); ...
192 plot( frequency_set, TL_part_b ); hold on;
193 plot( frequency_set, TL_part_c );
194 plot( frequency_set, TL_part_d, 'Color', 'k', 'LineStyle', '—' ); grid on;
195 legend( 'No Flow', 'Flow', 'With Helmholtz Resonator', 'Location', 'SouthEast' );
196 xlabel( 'Frequency [Hz]' ); ylabel( 'Transmission Loss [dB]' );
197 title( 'Transmission Loss Profiles for Intake System' );
198 ylim( [ -45 45 ] );
199 %
200 Ax = gca;
201 Ax.XAxis.TickLabelInterpreter = 'latex';
202 Ax.YAxis.TickLabelInterpreter = 'latex';
203
204 %% Clean-up
205
206 if ( ~isempty( findobj( 'Type', 'figure' ) ) )
207     monitors = get( 0, 'MonitorPositions' );
208     if ( size( monitors, 1 ) == 1 )
209         autoArrangeFigures( 2, 2, 1 );
210     elseif ( 1 < size( monitors, 1 ) )
211         autoArrangeFigures( 2, 2, 1 );
212     end
213 end
214
215 if ( PRINT_FIGURES == 1 )
216     exportgraphics((gcf, 'Intake System.pdf', 'Append', true );
217 end
218
219 fprintf( 1, '\n\n*** Processing Complete ***\n\n' );
220

```

221
222
223 *%% Reference (s)*

5 Appendix - Matlab Code for Problem 5

```
1
2
3 %% Synopsis
4
5 % Question 5 – Intake Duct Silencer
6
7
8
9 %% Environment
10
11 close all; clear; clc;
12 % restoredefaultpath;
13
14 % addpath( genpath( '' ), '-begin' );
15 addpath( genpath( '../40 Assignments/00 Support' ), '-begin' );
16
17 % set( 0, 'DefaultFigurePosition', [ 400 400 900 400 ] ); % [ left bottom width height ]
18 set( 0, 'DefaultFigurePaperPositionMode', 'manual' );
19 set( 0, 'DefaultFigureWindowStyle', 'normal' );
20 set( 0, 'DefaultLineLineWidth', 1.5 );
21 set( 0, 'DefaultTextInterpreter', 'Latex' );
22
23 format ShortG;
24
25 pause( 1 );
26
27
28
29 %% Part a
30
31 % The considered dip frequency 940 Hz.
32
33 % The additional attenuation required is 14 dB.
34
35 liner_thickness = 0.0381; % meters
36 h = sqrt( ( pi*( 0.1016 - 2*liner_thickness )^2 ) / 4 ); % 0.0113 meters
37
38 m = 1 + liner_thickness/h; % 2.7
39
40 % As noted in the discussion, there are two cases to be considered here.
41 %
42 % Figure 8.37 considers the combined effect of the expansion and the liner.
43 % In Problem 4, the effect of the expansion was calculated, so including it
44 % here would include its effect twice. Therefore, with an m of 1, there is
45 % not additional attenuation with the linear.
46 %
47 % The second case is to consider the m value of 2.7, which provides
48 % additional attenuation. From Figure 8.37, the total attenuation of
49 % the lining is 10 dB.
50 attenuation_rate = 10 / 0.127; % 78.4 dB per meter
51
52
53
54 %% Part b
55
56 % For a circular duct.
57
58 liner_thickness = 0.0381; % meters
59 half_diameter_of_open_orifice = 0.0127; % meters
60
61 h = sqrt(pi) * half_diameter_of_open_orifice / 2; % 0.0113 meters
62
63
64
65 %% Part c
66
67 % The liner thickness ratio is,
68 liner_thickness / h; % 3.3852 unitless
69
70 % The normalized frequency is,
71 ( 2 * h ) / ( 343 / 940 ); % 0.06169 unitless
72
73
74
75 %% Part d
```

```

76
77 % The approximate resistivity parameter is 4.
78
79
80
81 %% Part e
82
83 % Calculate the flow resistivity.
84
85 rho0 = 1.21; % Density of air (kg per cubic-meter).
86 c = 343; % Speed of sound in air (meters per second).
87
88 Rl = 16 * rho0*c / liner_thickness; % 1.74e5 kg / m^3*s
89
90
91 %% Clean-up
92
93 if ( ~isempty( findobj( 'Type', 'figure' ) ) )
94     monitors = get( 0, 'MonitorPositions' );
95     if ( size( monitors, 1 ) == 1 )
96         autoArrangeFigures( 2, 2, 1 );
97     elseif ( 1 < size( monitors, 1 ) )
98         autoArrangeFigures( 2, 2, 1 );
99     end
100 end
101
102
103 fprintf( 1, '\n\n*** Processing Complete ***\n\n' );
104
105
106
107 %% Reference(s)

```