Faculty of Applied Science and Technology

Laboratory 2 Basic Signal Operations

Student Name Michael McCorkell Student Number N01500049 Date: January 31th 2025

Note 1: This is an individual lab. Please bring your laptop and install required software to complete this lab.
Note 2: For your lab report, please take necessary screenshots and proper captions and include them in the corresponding parts of the lab. Demonstrate your work during the lab time. Submit the report file on the course website.

1. Learning outcome:

1.1 Familiarize with signal visualization in MATLAB.

1.2 Implement differentiation and integration in MATLAB.

1.3 Implement difference and accumulation in MATLAB.

2. Signals processing in MATLAB:

In this lab, we are going to continue working on signals in the time domain. We are going to explore some of the most common signal processing tools on both CT and DT signals. We are also going to examine the relation between CT and DT by applying some sampling concepts to CT and convert CT signals into their corresponding DT signals. Specifically, the categories of operations in the time domain that are commonly used are:

- CT signals: integration, differentiation, signal energy and power calculation, shifting and scaling.
- CT to DT signals: sampling.
- DT signals: differencing, accumulation, signal energy and power calculation, shifting and scaling.

In MATLAB, the integration of a mathematical expression can be computed through two types of methods: numerical integration, or analytical integration. Here, analytical integration refers to finding the closed form solution of the integration first. It can be used to compute indefinite integral or definite integral of the expression from a to b.

3. Procedures

3.1 Signal energy of CT signals

In MATLAB, analytical integration can be implemented using int() from Symbolic Math Toolbox.
Run the following three lines to see the results.

```
syms x
expr = -x^2+4*sqrt(x)+3;
F = int(expr)
```

Task 1 (5%) Modify the code above and compute the definite integral of the same expression $f(x) = -x^2 + 4\sqrt{x} + 3$ for $x \in [0,4]$. Record the code and the integral result below:

```
syms x
expr = -x^2+4*sqrt(x)+3;
F = int(expr, x, 0, 4)
disp(int(expr))
```

Task 2 (10%): Based on the information on how int() is implemented, compute the total signal energy $E_\infty = \int_{-\infty}^{\infty} |f(t)|^2 dt$ of the following signal $f(t)$:

$$f(t) = \begin{cases} 1 + \left|\frac{t}{3}\right|, -4 < t < 3 \\ 0, otherwise \end{cases}$$

Record the code and the result below:

```
syms t
task2 = 1+abs(t/3);
enrgy = int(task2^2, t, -4,  3)
```

Question 1 (10%): Calculate the total signal energy E_∞ of the signals $x(t)$ and $y(t)$ by hand, without using MATLAB.

a) $x(t) = e^{-t}[u(t) - u(t-1)]$

b) $y(t) = e^{-2t}[u(2t) - u(2t-1)]$

Show all steps of your work here.

a) $\int_0^1 \|e^{-t}\|^2$ 1. $\int_0^1 e^{-2t} \rightarrow \frac{e^{-at}}{-a}$ 2. $\left[\frac{e^{-2t}}{-2}\right]_0^1 \rightarrow \frac{e^{-2(1)}-e^{-2(0)}}{-2}$ 3. $x(t) = \frac{e^{-2}-1}{-2} \rightarrow \frac{1-e^{-2}}{-2}$	b) $\int_0^{\frac{1}{2}} \|e^{-2t}\|^2$ 1. $\int_0^{\frac{1}{2}} e^{-4t} \rightarrow \frac{e^{-at}}{-a}$ 2. $\left[\frac{e^{-4t}}{-4}\right]_0^{\frac{1}{2}} \rightarrow \frac{e^{-4\left(\frac{1}{2}\right)}-e^{-4(0)}}{-4}$ 3. $y(t) = \frac{e^{-2}-1}{-4} \rightarrow \frac{1-e^{-2}}{-4}$

Question 2 (10%): Comparing the results above, what conclusion can be drawn about the energy changes when the signal is compressed or expanded?

This shows that the energy of y(t) is exactly half of x(t), likely due to the time compression by a factor of 2.

3.2 Signal power of CT signals

In MATLAB, the limit of symbolic expressions can be computed using limit() from Symbolic Math Toolbox. Run the following three lines to see the results.

```
syms x h
f = sin(x)/x;
limit(f,x,0)
```

Task 3 (10%): Based on the information on how limit() is implemented, compute the signal power $P_\infty = \lim_{T\to\infty} \frac{1}{2T} E_\infty$ of the same signal $f(t)$ from task 2. Record the code and the result below:

```
syms t T
task2 = 1+abs(t/3);
enrgy = int(task2^2, t, -4, 3)
pimp = limit( (1/(2*T))*enrgy, T, inf)
```

3.3 Signal sampling from CT to DT

In this section, we are going to examine how sampling is conducted on a CT signal. This is an important concept that we will be repeatedly going back to in future. This example gives you a concept of how sampling can be used to approximate a CT signal in the DT domain. Given a CT signal $f(t) = \sin(2\pi f_0 t + \theta)$. Please plot the DT approximation of this CT signal with $t \in [-1,1]$.

```
f0 = 400;
fs = 8e3;
ts = 1./fs;
dt = 0.00004;
theta = pi./4;
t_axis = -1:dt:1;
t_axis_sample = -1:ts:1;
x_ct = sin(2*pi*f0*t_axis+theta);
x_dt = sin(2*pi*f0*t_axis_sample+theta);
figure; hold on;
plot (t_axis,x_ct,'green');
stem (t_axis_sample,x_dt);
```

Task 4 (20%): Based on the code above, fill in the following table.

Table 1. Parameters for signal sampling	
CT signal name	x_ct
Fundamental period of CT signal	$T_0 = \frac{1}{400} = 0.0025\ sec$
Fundamental frequency of CT signal	$f_0 = 400\ Hz$
Fundamental angular frequency of CT signal	$w_0 = 2\pi f_0 = 800\pi\ rad/sec$
DT signal name	x_dt
Fundamental period of DT signal	1/8000 =0.000125
Fundamental frequency of DT signal	400 Hz
Fundamental angular frequency of DT signal	$\frac{2\pi}{20}$ rad/sample
Sampling frequency	8000 hz
Purpose of dt	Defines the time resolution for continuous-time signal representation.
How to improve the final visualization?	- Increase the number of samples for x_dt to make it smoother. - Use different colors and line styles for better contrast.

Task 5 (15%): MATLAB has a function sound(audio_signal, sampling_frequency) that can send a digital signal out to the soundcard available on your platform. Now send the sampled data to the soundcard and listen to the result. Double the fundamental frequency of the original CT signal to see if that will make an impact. What about doubling the sampling frequency? Please explain your observation.

Sampled data to the soundcard and listen to the result: It's a low hum, with a lot of bass.
Double the fundamental frequency of the original CT signal to see if that will make an impact: It's making a higher pitched sound.
What about doubling the sampling frequency: I believe nothing changes from the noise and the previous test.

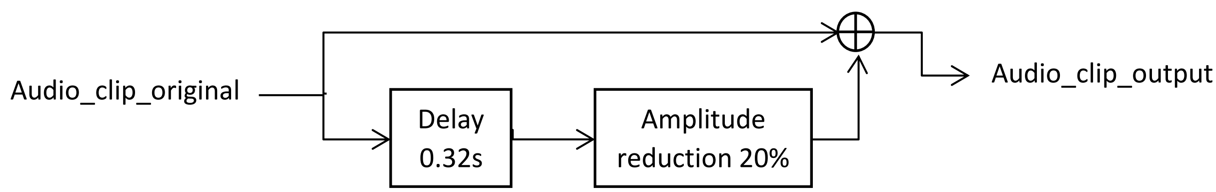
3.5 Application of shifting and scaling

Recall our activity in Lab0, we are going to conduct some basic signal processing on DT signals. First record an audio clip of 5 seconds [1]. Store this data as audio_clip_original. Use the DT shifting and scaling to create an echo effect. Below is one of the suggested ways to achieve that:

Step 1. Shift the audio_clip_original by 0.32s and reduce the shifted signal amplitude by 20%. Decide on the direction of the shift and how its implemented in MATLAB.

Step 2. Add the original signal and the delayed signal together to create a new audio_clip_output.

Step 3. Similar to 3.4, use sound() to send the processed DT signal directly to the speaker. Decide on the sampling frequency and other needed parameters to implement this step. Hint: the sampling frequency is determined at the recording step, and is part of the properties of audiorecorder object that you have created previously.



Task 6 (20%): Based on the information given, please create an audio clip with echo effect in an auditorium of the size (500 by 500 ft). Record your MATLAB code below as well as plot the original audio clip and the processed one side by side to compare the difference. Include relevant screenshots. Please justify the design parameters that you used in the space below. Hint: the speed of sound is 1125 ft/s.

References:
[1] *The MathWorks, Inc.* "Record and Play Audio," *MATLAB Documentation*. [Online]. Available: https://www.mathworks.com/help/matlab/import_export/record-and-play-audio.html
Accessed:Jan.10.

TASK 6 NEXT PAGE

In an auditorium of size **500 × 500 ft**, the time delay for an echo depends on the speed of sound (1125 ft/s). The longest reflection path is approximately the **diagonal** of the auditorium:

$$d = \sqrt{500^2 + 500^2} = 707.1ft \qquad t_{delay} = \frac{d}{sos} = \frac{707.1}{1125} \approx 0.63\ s$$

```
recAud = audiorecorder(44100,16,1);
disp("Start Speaking...")
recordblocking(recAud,5);
disp("Done Recording");
audio_clip_original = getaudiodata(recAud);
delay = 0.63; amp_red=0.6; sr = recAud.SampleRate;
delaySamp = round(delay*sr);
audioDelay = [zeros(delaySamp, 1); audio_clip_original] * amp_red;
audioDelay = audioDelay(1:length(audio_clip_original));
audio_clip_output = audioDelay+audio_clip_original;
time = (0:length(audio_clip_original)-1) / sr;
figure; subplot(2,1,1);
plot(time, audio_clip_original); title('Original Audio'); grid; xlabel('Time (s)'); ylabel('Amplitude');
subplot(2,1,2);
plot(time, audio_clip_output); title('Output Audio'); grid; xlabel('Time (s)'); ylabel('Amplitude');
```