Experiment 3 - Fourier Synthesis of Periodic Waveforms

Department of Electrical Engineering and Electronics

September 2020, Ver. 5.0

Experiment specifications

Module(s)	ELEC270
Experiment code	3
Semester	1
Level	2
Lab location	Computer lab, third/fourth floor, check the lab timetable
Work	In groups
Timetabled time	7 hrs
Subject(s) of relevance	Fourier theorem
Assessment method	Report template
Submission deadline	7 days after lab date

Instructions:

- **Read this script before attempting the experiment.**
- **The Pre-Lab Questions should be answered before the lab day. They are available online and worth 10%.**
- **The script questions should be answered while carrying out the experimental procedure.**
- **The report template should be completed with the graphs and answers to the questions and submitted before the deadline.**

1 Objectives

This experiment is aimed at:

- Studying quantitatively Fourier components of some basic periodic waveforms.
- Verifying that periodic waveforms can be synthesised by the superposition of sinusoids having harmonically related frequencies.
- Studying the effect of filters on waveforms.
- Investigating the effect of phase on sound.

2 Apparatus

- A computer with Matlab

Students of the University of Liverpool can download a copy of Matlab for use in their personal computers/laptops from the following link:
https://www.liverpool.ac.uk/csd/software/software-downloads/#matlab

3 Introduction

In the fields of communications, signal processing and in electrical engineering more generally, a **signal** is any time-varying or spatial-varying quantity. A graph of the amplitude of this signal plotted against time produces a waveform (an example is shown in Figure 1). Perhaps the most familiar type of waveforms is the sinusoid shown in Figure 1 (b). Here, the voltage (V) at any time (t) is given by the equation:

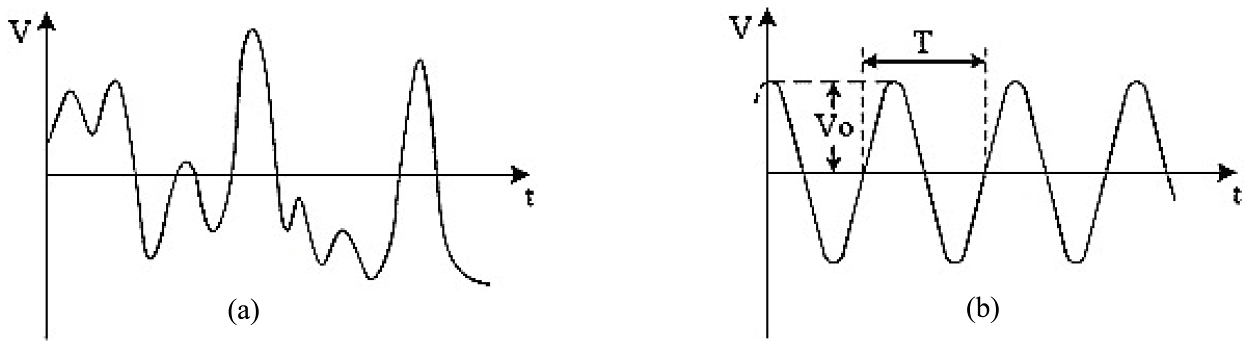

Figure 1: Examples of waveforms.

$$V(t) = V_o sin(2\pi ft + \phi) \tag{1}$$

where V_o is the amplitude of the sinusoid, f is the frequency and ϕ is the phase angle measured in radians.

It may be seen from the figure that this sinusoidal waveform is periodic with period T, i.e. the waveform repeats itself after every T seconds, or $V(t+nT) = V(t)$ for all t, where n is an integer. The frequency f, measured in Hertz, and the period T are related by $T = 1/f$. The phase angle determines the starting point of the sine wave at $t = 0$.

Analysis:

Many real-life signals are periodic but non-sinusoidal. Examples are shown in Figure 2. Fourier analysis is based on a theorem which states that periodic waveforms may be expressed as the sum of infinite simple sines and cosines (plus possibly a constant). If T is the period of a periodic signal $f(t)$, which satisfies certain conditions (normally satisfied by signals of practical interest), then $f(t)$ can be expressed as the sum of sine and cosine functions called a *Fourier series* that is uniquely defined by constants known as *Fourier coefficients*. Hence:

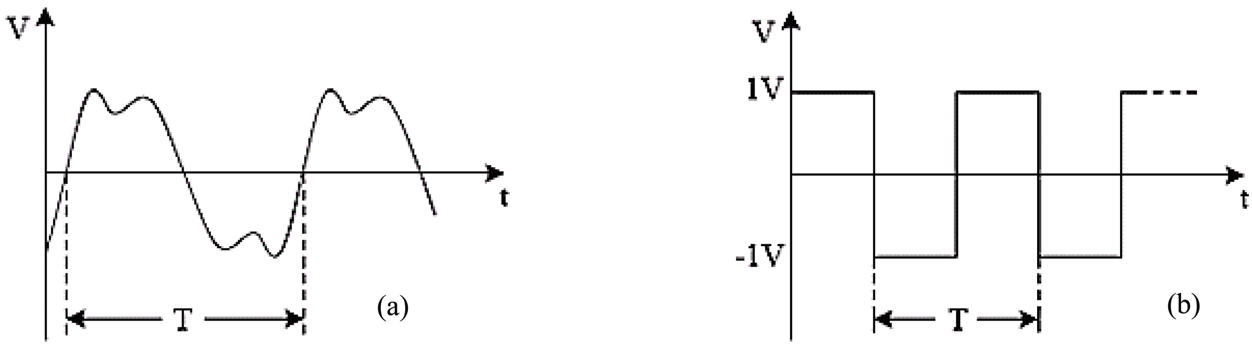

Figure 2: Non-sinusoidal periodic waveforms.

$$f(t) = a_o + \sum_{n=1}^{\infty} a_n \cos(2\pi n f_o t) + \sum_{n=1}^{\infty} b_n \sin(2\pi n f_o t) \tag{2}$$

where $f_o = 1/T$ and is termed as the *fundamental frequency*, a_o, a_n and b_n (for $n = 1, 2, 3, \ldots$) are the Fourier coefficients that can be calculated from the following expressions:

$$a_o = \frac{1}{T}\int_0^T f(t)dt \tag{3}$$

$$a_n = \frac{2}{T}\int_0^T f(t)\cos(2\pi n f_o t)dt \tag{4}$$

$$b_n = \frac{2}{T}\int_0^T f(t)\sin(2\pi n f_o t)dt \tag{5}$$

Equation 2 may also be written in an alternative form:

$$f(t) = c_o + \sum_{n=1}^{\infty} c_n \cos(2\pi n f_o t + \phi_n) \tag{6}$$

where $c_n = \sqrt{(a_n^2 + b_n^2)}$ and $\tan\phi_n = -b_n/a_n$.

Note that $f(t)$ has been expressed as the sum of a constant (DC) voltage c_o and sinusoids of frequencies f_o, $2f_o$, $3f_o$, $4f_o$ and so on. The sinusoid at frequency f_o, i.e. $c_1 \sin(2\pi f_o t + \phi_1)$, is called the *fundamental component* of $f(t)$. The other sinusoids are called *harmonic components*; the component at frequency $2f_o$ being the second harmonic; the component at $3f_o$ being the third harmonic and so on.

Representing a periodic function $f(t)$ as a series of sinusoids of specific amplitudes is referred to as **Fourier Analysis**.

Example: Fourier analysis of a triangular wave.

It is required to analyse a triangular wave (Figure 3) using Fourier analysis. First, we need to establish an expression for the time domain waveform $f(t)$. It can be concluded easily that $f(t)$ can be written as:

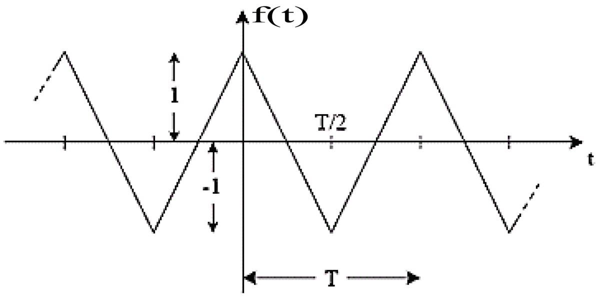

Figure 3: Triangular wave.

$$f(t) = \begin{cases} 1 - \frac{4t}{T} & 0 \leq t \leq \frac{T}{2} \\ \frac{4t}{T} - 3 & \frac{T}{2} \leq t \leq T \end{cases} \tag{7}$$

Now, using the definition Equations 3, 4 and 5, we can find that:

$$a_o = \frac{1}{T}\int_0^{T/2} \left(1 - \frac{4t}{T}\right) dt + \frac{1}{T}\int_{T/2}^{T} \left(\frac{4t}{T} - 3\right) dt = 0 \tag{8}$$

and,

$$a_n = \begin{cases} \frac{8}{(n\pi)^2} & n\ odd \\ 0 & n\ even \end{cases} \tag{9}$$

Similarly, it can be found that b_n= 0 for all n (the triangular wave has even symmetry). Hence, the Fourier series for $f(t)$ is:

$$f(t) = \frac{8}{\pi^2}\left[\cos(2\pi f_o t) + \frac{1}{9}\cos(2\pi 3 f_o t) + \frac{1}{25}\cos(2\pi 5 f_o t) + \ldots\right] \tag{10}$$

or:

$$f(t) = \frac{8}{\pi^2}\sin(2\pi f_o t + \frac{\pi}{2}) + \frac{8}{9\pi^2}\sin(2\pi 3 f_o t + \frac{\pi}{2}) + \ldots \tag{11}$$

The spectrum of $f(t)$ (frequency domain view) contains an infinite number of frequency components, hence, the bandwidth of $f(t)$ is infinite. A practical system would restrict $f(t)$ to a finite bandwidth by removing all frequency components above certain upper limit. This would distort the shape of the triangular waveform. Fortunately, the first few terms in the Fourier series of a triangular wave are the most important since the amplitudes, a_n, decrease rapidly as n increases.

Synthesis:

The addition of a series of harmonically-related sinusoids in order to generate a periodic waveform is referred to as Fourier synthesis, and is the reverse process of analysis.

To synthesise a periodic wave with a good approximation, the sine and cosine waves with the proper amplitudes (as defined by the coefficients) must be electronically generated and combined up to the highest possible (and practical) value of n. The larger the value of n for which sine and cosine wave signals are generated the more nearly the synthesised waveform matches the desired waveform. Figure 4 is an example of successive approximation of a *sawtooth wave* by adding harmonics with amplitude inversely proportional to the harmonic number. The resultant waveform at each stage of addition is shown at right.

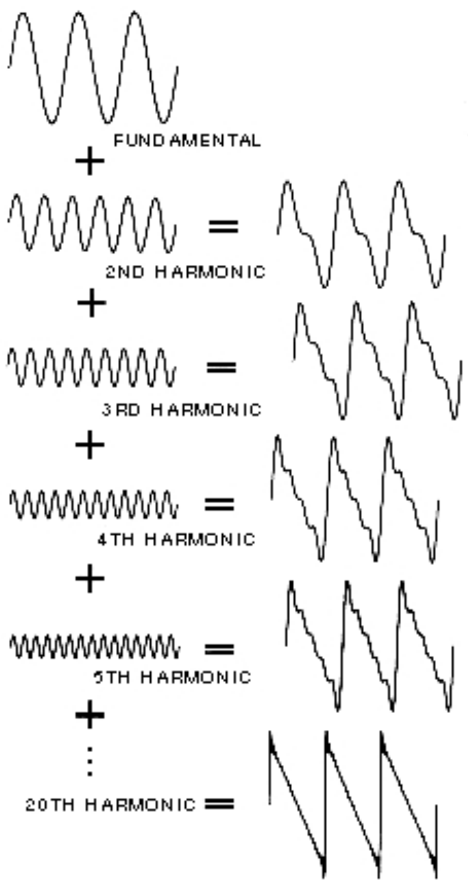

Figure 4: Successive approximation of sawtooth signal.

Fourier synthesis is used in many practical applications. For example, it is used extensively in electronic music applications to generate waveforms that mimic the sounds of familiar musical instruments. Although many musicians can create very clear-tones on their instruments, all instruments have their own characteristic voice. Voices are differentiated by the shape of their associated sound waves. All instrument voices display an array of frequencies which sum together to produce a wave of characteristic shape. Fourier synthesis is also employed in laboratory instruments known as *function (signal) generators* that are used to generate different waveforms for various purposes like electronic and communication systems test.

Example: Fourier synthesis of a triangular wave with Matlab.

The triangular wave analysed above can be synthesised in Matlab by generating the right sine/cosine components at the right harmonic frequencies and calculating a weighted sum of them. The Matlab script below shows how to do this based on the expressions shown in equations 8, 9 and 10:

```
clear

% Set signal period to 1 second
T = 1;

% Time variable (includes 1 signal period)
t = linspace(0, T, 1000);

% Triangular signal (evaluated over one period)
triang = zeros(size(t));
triang(t <= T/2) = 1 - 4*t(t <= T/2)/T;
triang(t >  T/2) = 4*t(t > T/2)/T - 3;

% Synthesised signal (evaluated over one period)
max_harmonics = 3;
a = zeros(1, max_harmonics+1);
b = zeros(1, max_harmonics+1);
synth_triang = zeros(size(t));
for n = 0:max_harmonics
    if mod(n,2)
        a(n+1) = 8/((n*pi)^2);
        synth_triang = synth_triang + a(n+1)*cos(2*pi*n*(1/T)*t);
    end
end

% Concatenate three periods for a better visualisation
t = linspace(0, 3*T, 3*1000);
triang = repmat(triang, 1, 3);
synth_triang = repmat(synth_triang, 1, 3);

% Plot results
plot(t, triang, 'b-', t, synth_triang, 'r--')
legend('Triangular wave', 'Synthesised triangular wave')
```

Figure 5: Matlab code to synthesise a triangular waveforms (3 harmonics).

Figure 6 compares the original triangular wave and its synthesised version with 1, 3, 5, and 9 harmonic components (left-to-right, top-to-bottom). It can be seen that the synthesised waveform will reproduce the original triangular signal more accurately as more harmonic components are added. However, only the first few terms in the Fourier series of a triangular wave are the most important.

4 Experimental work

As mentioned above, Fourier synthesis is a method of electronically constructing a signal with a specific, desired periodic waveform. It works by combining sinusoidal harmonics (signals at multiples of the lowest or fundamental frequency) in certain proportions. In this experiment you are going to use Matlab to synthesis arbitrary waveforms using harmonics as shown in the example above (you can use this example as a reference).

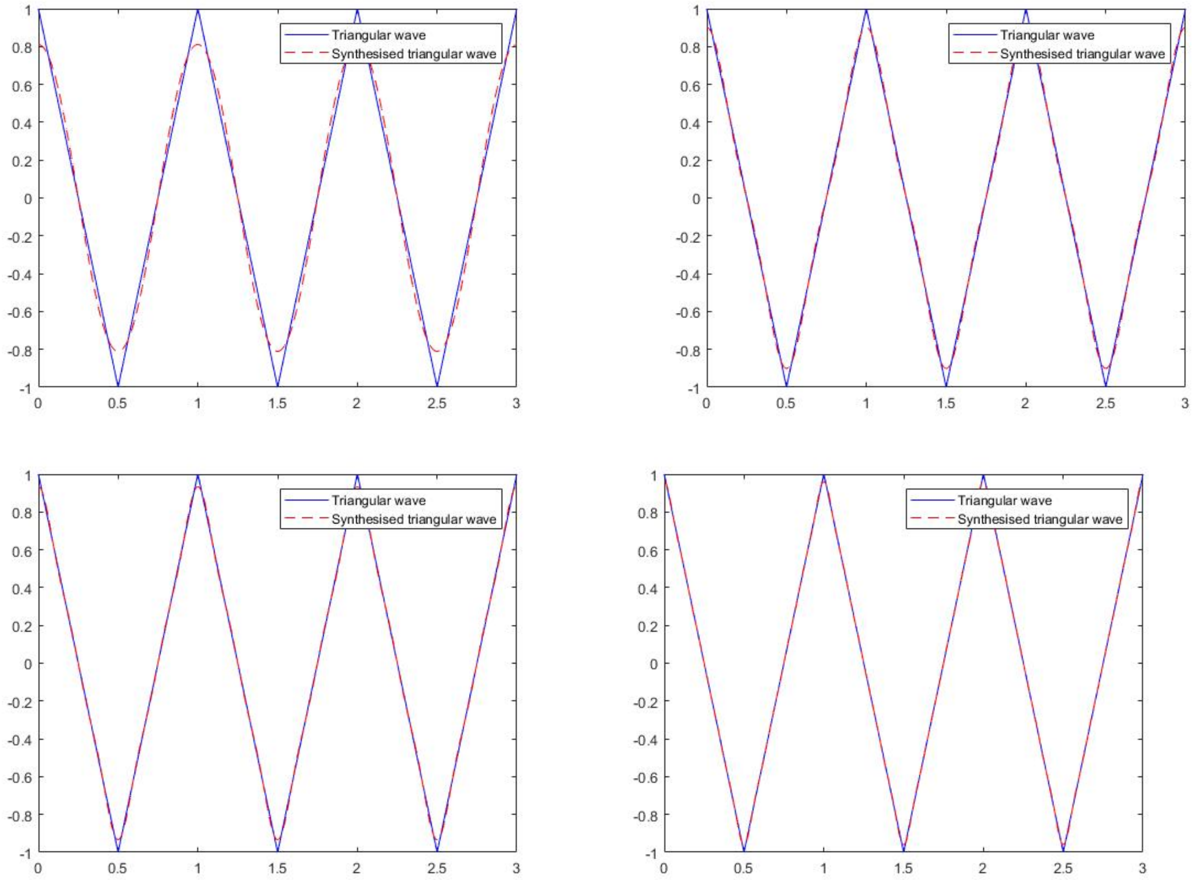

Figure 6: Original triangular wave and its synthesised version with 1, 3, 5, and 9 harmonic components (left-to-right, top-to-bottom).

4.1 Part A (30 Marks)

Objectives: To familiarise yourself with Fourier synthesis using Matlab and to investigate the extent of the accompanied error.

It is required to synthesise the following function:

$$f(t) = V_o \left(\sin \omega t + \frac{1}{3} \sin 3\omega t + \frac{1}{5} \sin 5\omega t + \frac{1}{7} \sin 7\omega t + \frac{1}{9} \sin 9\omega t \right) \tag{12}$$

where V_o is the maximum applicable voltage (you can select an arbitrary value, for example $V_o = 1$ volt) and $\omega = 2\pi f_o$.

Requirements:

- Provide the Matlab code required to synthesise the waveform in equation 12 and the resulting waveform. **[5 marks]**

Questions:

- Explain the features of the resulting waveform (peak-to-peak amplitude, symmetry, ripple, etc.) **[5 marks]**

$2V_o$ odd Gibbs phenomenon

square wave

- How do you think the waveform would look if an unlimited number of harmonics was available (i.e. n goes to ∞)? To support your answer, provide a couple of figures along with their associated code. [**5 marks**]
- Referring to Equation 3, what is the value of a_o? [**5 marks**] 0
- Find an expression (in terms of n) for a_n and b_n. [**5 marks**] $a_n = 0 \quad b_n = \frac{1}{2n-1}$
- Plot the spectrum (frequency domain view) of $f(t)$ using $c_n = \sqrt{(a_n^2 + b_n^2)}$. Provide the figure and the Matlab code used to obtain it (you can use the `stem` function from Matlab to plot the frequency components). [**5 marks**]

4.2 Part B (20 Marks)

Objective: To synthesise and study a sawtooth waveform.

The periodic waveform shown in Figure 7 is called a sawtooth function, and can be represented mathematically as follows:

$$f(t) = V - \frac{2V}{T}t, \quad 0 \leq t < T \tag{13}$$

Using, the Fourier analysis equations (Equations 3, 4 and 5), the Fourier coefficients can be shown to be a_o=0, a_n=0 and b_n=$2V/\pi n$.

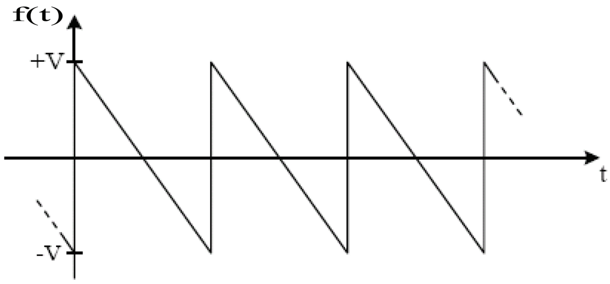

Figure 7: Sawtooth waveform.

$f(t) = \sum_{n=1}^{\infty} \frac{2V}{\pi n} \sin 2\pi n f_0 t$

Requirements:

1. Write $f(t)$ in Fourier synthesis form, i.e. as in Equation 2. [**4 marks**]
2. Calculate the first 10 sinewave coefficients (i.e. b_1, b_2, $\ldots b_{10}$). [**4 marks**] $b_n = \frac{2V}{\pi n}$
3. Synthesise the first 10 harmonics of this waveform and plot the result (provide your Matlab code as well). [**4 marks**]

Questions:

- Plot in the same figure the original and synthesised sawtooth waveforms (provide your Matlab code). Compare the resulting waveform with what you expected to see and discuss the results. [**4 marks**]
- If the number of harmonics is reduced to 5, comment on the changes that will be observed practically. [**4 marks**]

$$f(t) = V - \frac{2V}{T}t \qquad t \in [0, T)$$

$f(t)$ is an odd function, so it does not have even componets; hence $a_n = 0$

$f(t)$ is symmetric to the x-axis, so $a_0 = 0$

$$\begin{aligned} a_0 &= \frac{1}{T}\int_0^T f(t)\,dt \\ &= \frac{1}{T}\int_0^T V - \frac{2V}{T}t\,dt \\ &= \frac{1}{T}\left[Vt - \frac{V}{T}t^2\right]_0^T \\ &= \frac{1}{T}\left(VT - \frac{V}{T}T^2\right) \\ &= 0 \end{aligned}$$

4.3 Part C (15 Marks)

Objective: To synthesise and study various waveforms.

Requirements:

1. Synthesise the waveforms below. Plot the resulting waveforms and provide the Matlab code used to obtain the plots as well. **[1, 1, 1 and 2 marks, respectively]**
   - (a) $f_1(t) = \sin\omega_o t - \frac{1}{9}\sin 3\omega_o t + \frac{1}{25}\sin 5\omega_o t - \frac{1}{49}\sin 7\omega_o t + \ldots$.
   - (b) $f_2(t) = 0.1(\cos\omega_o t + \cos 2\omega_o t + \cos 3\omega_o t + \cos 4\omega_o t + \ldots)$.
   - (c) $f_3(t) = \sin\omega_o t - \frac{4}{3\pi}\cos 2\omega_o t - \frac{4}{15\pi}\cos 4\omega_o t - \frac{4}{35\pi}\cos 6\omega_o t - \frac{4}{63\pi}\cos 8\omega_o t - \frac{4}{99\pi}\cos 10\omega_o t$.
   - (d) The waveform in (c) above with the fundamental component ($\sin\omega_o t$) switched off.

Question:

- Comment on your results for each waveform. **[3, 3, 2 and 2 marks, respectively]**

4.4 Part D (15 Marks)

Objectives: To study and simulate the effect of a linear filter on a wave.

The effect of a linear filter on a certain waveform can be simulated by passing each Fourier component of that waveform through the filter and observing the effect on the amplitude and phase of each separate component.

Requirements:

Consider an ideal square wave with frequency of 1 kHz expressed as a Fourier series:

$$V(t) = \frac{3}{\pi}\left(\sin(2\pi f_o t) + \frac{1}{3}\sin(2\pi 3 f_o t) + \frac{1}{5}\sin(2\pi 5 f_o t) + \ldots\right) \quad (14)$$

1. Synthesise and plot this square wave (provide your Matlab code as well). Calculate the percentage overshoot of the synthesised waveform (compared with the ideal waveform) at the discontinuity. How does this compare with the expected limit of 17.9%? **[2 marks]**
2. What is the name of this overshoot? Explain it. **[2 marks]**
3. Give the Fourier series in each case for the resulting waveform if the above square wave is used as an input for:
   - (a) A low-pass filter with gain and phase responses as given in Figures 8 and 10 respectively. **[1 mark]**
   - (b) A low-pass filter with gain and phase responses as given in Figures 8 and 11 respectively. **[1 mark]**
   - (c) A band-pass filter with gain and phase responses as given in Figures 9 and 10 respectively. **[1 mark]**
4. Synthesise the above waveforms and draw the obtained waveforms for filters (a), (b) and (c), providing the Matlab code as well. **[3 marks]**

Question:

- What is the fundamental frequency of the output from filter (c)? Why? **[5 marks]**

$3f_o$, because by definition, it is the lowest

frequency of a Fourier series.

$$\frac{3}{\pi}\left(\cos(2\pi f_0 t) - \frac{1}{3}\cos(2\pi 3 f_0 t) \ldots + \frac{1}{9}\cos(2\pi 9 f_0 t)\right)$$

$$\frac{3}{\pi}\left(\cos(2\pi f_0 t) + \frac{1}{3}\cos(2\pi 3 f_0 t) \ldots \frac{1}{9}\cos(2\pi 9 f_0 t)\right)$$

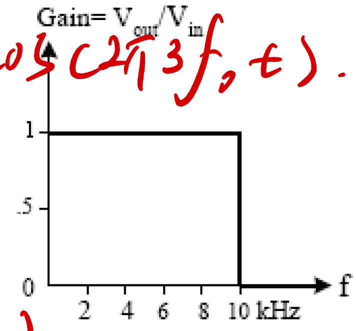

Figure 8: Low-pass filter gain response.

$$\frac{3}{\pi}\Big(\frac{1}{2}\times\left(-\frac{1}{3}\cos 2\pi 3 f_0 t\right) + \frac{1}{5}\cos 2\pi 5 f_0 t - \frac{1}{7}\cos 2\pi 7 f_0 t + \frac{1}{2}\times\left(\frac{1}{9}\cos 2\pi 9 f_0 t\right)\Big)$$

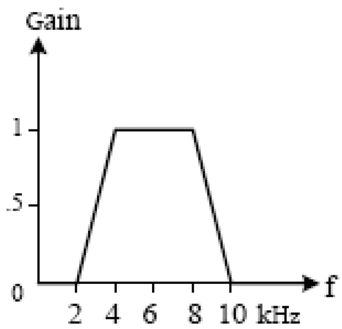

Figure 9: Band-pass filter gain response.

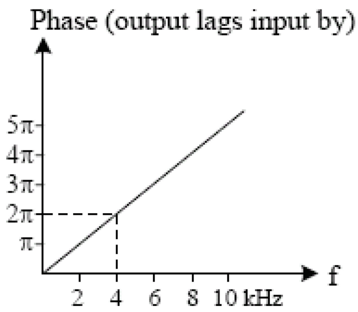

Figure 10: Example of a phase response.

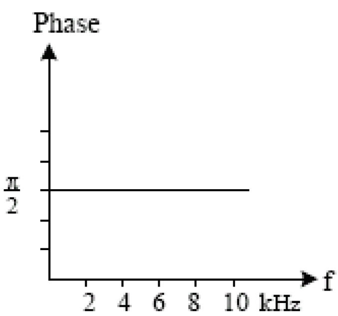

Figure 11: Example of another phase response.

How to implement these filter traits in MATLAB?

4.5 Part E (10 Marks)

Objectives: To listen to harmonics and chord, and investigate the effect of phase on sound.

In this part, you need to listen to the sound generated by the speaker of your computer when playing one or more harmonic components of a signal.

The example below shows how to play in Matlab a tone of 500 Hz with a duration of 3 seconds. The amplitude can be modified to adjust the volume. The sampling frequency $f_s = 44.1$ kHz is commonly used by most computer sound cards.

```
fs = 44100; % Sampling frequency
duration = 3;
t = 0:1/fs:duration;
amplitude = 10;
f = 500;
wave = amplitude * sin(2*pi*f*t);
sound(wave,fs)
```

Figure 12: Example of Matlab code to play a tone.

To plot a signal generated this way, it is convenient to clip it first. Notice that the tone generated in the example above has a frequency of 500 Hz and its period is 2 ms. This period is repeated 1500 times in 3 seconds, so plotting the complete signal will not be very useful. The example below shows how to clip the signal generated in the example above to just 3 periods before plotting it.

```
T_max = 3;
clipped_t = t(1:find(t <= T_max/f, 1, 'last'));
clipped_wave = wave(1:find(t <= T_max/f, 1, 'last'));
plot(clipped_t, clipped_wave)
```

Figure 13: Example of Matlab code to clip a long audio signal for plotting.

Consider the following wave with a fundamental frequency $f_o = 500$ Hz:

$$V(t) = 10\left[\sin(2\pi f_o t) + \sin(2\pi 2 f_o t) + \sin(2\pi 3 f_o t)\right] \tag{15}$$

Requirements:

✓ 1. For the wave in equation 15, listen to harmonics 1, 2 and 3 individually then as a chord. Plot the chord waveform as well. Provide the Matlab code used to listen to the harmonics/chord and plot the chord. [**3 marks**]

✓ 2. Alter the phase of the third harmonic in equation 15 by 90° and repeat the tasks in the point above. [**3 marks**]

Questions:

same changed

- Discuss the effect of altering the phase of the third harmonic, both on the sound and plot of the chord. Do the sound or plot change as you alter the phase? Why? [**2 marks**]
- What does the above tell you about the human ear? [**2 marks**]

sound doesn't change, because timbre depends primarily upon the frequency spectrum human ear can distinguish distinct sounds by frequency and combination of them.

5 Assessment and Marking Scheme

The marking scheme for this experiment is as follows:

- Results of Part A: **30 Marks**
- Results of Part B: **20 Marks**
- Results of Part C: **15 Marks**
- Results of Part D: **15 Marks**
- Results of Part E: **10 Marks**
- The pre-lab test: **10 Marks**

6 Plagiarism and Collusion

Plagiarism and collusion or fabrication of data is always treated seriously, and action appropriate to the circumstances is always taken. The procedure followed by the University in all cases where plagiarism, collusion or fabrication is suspected is detailed in the University's Policy for Dealing with Plagiarism, Collusion and Fabrication of Data, Code of Practice on Assessment, Category C, available on `https://www.liverpool.ac.uk/media/livacuk/tqsd/code-of-practice-on-assessment/appendix_L_cop_assess.pdf`.

Follow the following guidelines to avoid any problems:

(1) Do your work yourself.

(2) Acknowledge all your sources.

(3) Present your results as they are.

(4) Restrict access to your work.

References

[1] Manual, "Fourier Synthesiser ST2603 training system", Scientech, 2008.

[2] W Al-Nuaimy, "Lecture notes, ELEC270, Signals and Systems", University of Liverpool.

Version history

Name	Date	Version
Dr M López-Benítez	September 2020	Ver. 5.0
Dr M López-Benítez	September 2019	Ver. 4.4
Dr A Al-Ataby	August 2014	Ver. 4.3
Dr A Al-Ataby	October 2013	Ver. 4.2
Dr A Al-Ataby and Dr W Al-Nuaimy	October 2012	Ver. 4.1
Dr A Al-Ataby and Dr W Al-Nuaimy	October 2011	Ver. 4.0
Dr L Momani	September 2009	Ver. 3.0
Dr K Nuttall	October 2003	Ver. 2.0
Dr J Marsland	August 2000	Ver. 1.0