Lab 1 - Fourier Series Analysis and Synthesis

Objectives

In this lab we will numerically study Fourier series (FS) analysis and synthesis. Periodic signals can be represented by (possibly infinite) sum of harmonically related complex exponentials given by the Fourier series representation. We will study

- Computing Fourier series coefficients numerically
- Fourier series reconstruction and approximation errors
- Gibbs Phenomenon
- Properties of Fourier series.

1.1 Finding Fourier series coefficients

For a periodic signal $x(t)$ Fourier series coefficients are determined by

$$a_k = \frac{1}{T}\int_{\langle T\rangle} x(t)\mathrm{e}^{-jk\omega_0 t}\,dt, \qquad k \in \mathbb{Z} \tag{1}$$

Write a Matlab function "fourierCoeff" to compute FS coefficients for a given periodic signal x(t). your function should take input as:

- N, to compute Fourier Series coefficients for $k = -N{:}N$, i.e. $\{\, a_{-N} \ldots, a_{-1,} a_{0,} a_{1,} \ldots\ a_{N,}\}$
- T, period of the signal
- t, symbolic variable for integration (see Hints below)
- xt, original signal as a function of 't'
- a, b are limits where the expression xt is valid (else zero otherwise)

and return F, which corresponds to the Fourier series coefficients $\{\, a_{-N} \ldots, a_{-1,} a_{0,} a_{1,} \ldots\ a_{N,}\}$ arranged in a vector.

Write a matlab script file which calls "fourierCoeff" function for the periodic triangle wave. Consider the periodic triangular wave over one period (T) as given by

$$x(t) = \begin{cases} \left(\dfrac{1}{4} - |t|\right), & -\mathrm{T}_1 \le t \le \mathrm{T}_1 \\ 0, & \mathrm{T}_1 < |t| < T/2 \end{cases}$$

where $\mathrm{T} = 1$ and $\mathrm{T} = 4\mathrm{T}_1$.

a) Calculate FS coefficients for $x(t)$ using "fourierCoeff" function and plot them for N = 10.
b) Solve part(a) analytically and compare with FS coefficients obtained using "fourierCoeff".
c) Try any other periodic signal $x(t)$ of your choice and find FS coefficients.

Hints:

1. Fundamental period (T) and fundamental frequency (w_0) are related as $w_0 = \frac{2\pi}{T}$
2. Making use of symmetry, it is convenient to perform integration over $-T/2 \leq t \leq T/2$
3. Use MATLAB command ***int (expr, t, a, b)*** for integration; **expr** represents the function to be integrated, **t** is a symbolic variable for the symbolic expression, and **a** and **b** are limits of the definite integral. You need to define **t** as symbolic variable using a command ***syms.*** Type 'doc int' in command prompt to get its documentation. An example is given below

```
syms t;
expr = t*(1+t);
F = int (expr,t, [0 1]);
```

4. Use different matlab files for scripts and functions. Use the following code templates to get started for this problem (copy-paste into matlab and edit):

<table>
<tr><th>Matlab Script:</th><th>Matlab Function</th></tr>
<tr><td>

```
% define relevant parameters
...

% define relevant expressions
syms t;
xt = ...  % triangle wave

% function call to find FS coefficients
F = fourierCoeff(N,T,t,xt,a,b);

% plotting
FS_idx = -N:N;
figure; stem(FS_idx,F); grid on;
```

</td><td>

```
function F = fourierCoeff(N,T,t,xt,a,b)
% function to find FS coefficients

% initialize
w0 = ...
F = zeros(2*N+1,1);
FS_idx = ...

% for-loop to find coefficents
for nn = 1:2*N+1
    expr = ...
    F(nn) = ...
end

end
```

</td></tr>
</table>

5. Avoid using i or j as loop variables. In Matlab they can be interpreted as the imaginary number iota i.e. square root of -1. Also, while using imaginary number iota in your code it is preferable to write it as 1i which is unambiguous, for example $e^{j\frac{\pi}{3}}$ is exp(1i*pi/3).

1.2 FS reconstruction and finite FS approximation error

For a signal $x(t)$ with Fourier series coefficients $\{a_k\}$, a partial Fourier sum (of order N) is given as

$$\hat{x}(t) = \sum_{k=-N}^{N} a_k \, e^{jk\omega_0 t} \qquad (2)$$

As the order N is increased (to infinity) the partial sum approaches the original signal $x(t)$.

Write a Matlab function, "partialfouriersum", to compute a partial Fourier sum from a given vector of Fourier series coefficients $\{ a_k \}$. Your function should take as input

- A, a 2*N+1 vector of Fourier Series coefficients { a_{-N}..., $a_{-1}, a_0, a_1, \dots a_N,$}
- T, the period of the signal
- t, a time vector

and should return y, which corresponds to the Fourier series reconstruction $y(t)$ obtained from computing the partial Fourier sum from -N to N. Use the following template:

```
function y = partialfouriersum (A, T,t)
% Compute N based on the length of a
y = zeros(size(t));
for k = -N:N
    y = y + ...
end
end
```

a) Write a script which takes the output of part (a) in 1.1 and reconstructs one period of the triangle wave, T = 1 and t = -0.5:0.01:0.5. In reality, time is continuous, but for sake of plotting and analysis we consider a fine time grid (0.01 spacing) to imitate continuous time. You can increase spacing to 0.001 for a finer resolution (but more computations).
b) Plot $x(t)$, original signal (see hint) given in 1.1 and the reconstructed signal in the same plot (use plot function here and not stem, what is difference between these two functions?).
c) For a fixed set of input parameters, from the outputs of the above function compute the following two types of error:
   1) The maximum absolute error between original signal and reconstructed signal.
   2) The mean squared error between y and original signal and reconstructed signal.
d) Write a Matlab script which computes these two errors as N is increased from 1 to 100 and plots them. What are your observations?
e) Commit all your codes GitHub at this point!!

Hints:

1. Use the following matlab code snippet to plot original signal (triangle wave) in one period:

```
M = length(t); K = floor(M/4);
xt = zeros(M,1);
nz_idx = K+1:3*K+1; % indices where xt is non-zero
xt(nz_idx) = 0.25 - abs(t(nz_idx));
figure; plot(t,xt); grid on;
```

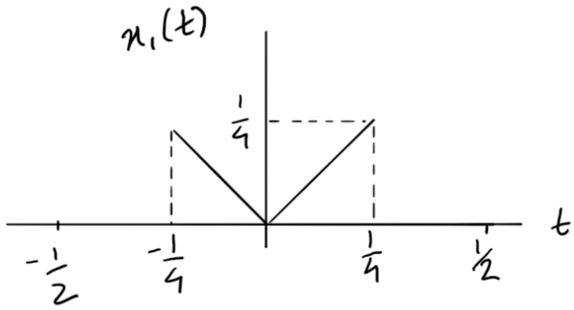
1.3 Gibbs Phenomenon – square wave

a) What are the Fourier Series coefficients { a_k } for a real, periodic square wave that has amplitude 1 in [-T_1, T_1] and period T? Note that we require T_1 < T/2.
b) Find these FS coefficients by calling the function "fourierCoeff". Use T_1 = 0.1, T=1, N = 100*T. What happens as T is increased to T = 10, T = 100, and more?

c) Do partial Fourier sum reconstruction of square wave using the function "partialfouriersum" and plot it. Use T_1 = 0.1, T = 1, t = -0.5:0.01:0.5, N = 10. Repeat this as N is increased (N = 50,100,1000) and note your observations.
d) Comment on nature of oscillations with respect to increasing N (Optional reading: Gibbs Phenomenon, Section 3.4 in book OWN).

1.4 Fourier series – more examples and symmetry properties

a) Team member 1 - write a Matlab script to find FS coefficients of $x_1(t)$ using "fourierCoeff".
b) Team member 2 - write a Matlab script to find FS coefficients of $x_2(t)$ using "fourierCoeff".
c) What do you expect the results to be? What symmetry properties hold for these examples?
d) Don't forget to commit your codes and answers to GitHub!



Period $T = 1$