12-02-2020

Experiment 3
Fourier Series in MATLAB

Sahil Bondre (u18co021)

Aim.

1. To compute the Fourier Coefficients of Exponential and Square Wave
2. Plot their Magnitude and Phase Spectrum

Theory.

Fourier Series

Fourier series is a periodic function composed of harmonically related sinusoids, combined by a weighted summation. With appropriate weights *(Fourier Coefficients)*, one cycle (or period) of the summation can be made to approximate an arbitrary function in that interval (or the entire function if it too is periodic). As such, the summation is a synthesis of another function. The discrete-time Fourier transform is an example of the Fourier series.

$$\begin{cases} a_0 &= \dfrac{1}{2\pi}\displaystyle\int_{-\pi}^{\pi} F_n(x)dx, \\ a_k &= \dfrac{1}{\pi}\displaystyle\int_{-\pi}^{\pi} F_n(x)\cos(kx)dx, \quad 1 \le k \le n \\ b_k &= \dfrac{1}{\pi}\displaystyle\int_{-\pi}^{\pi} F_n(x)\sin(kx)dx, \quad 1 \le k \le n. \end{cases}$$

Integration in MATLAB

In contrast to differentiation, symbolic integration is a more complicated task. A number of difficulties can arise in computing the integral:

- The antiderivative, F, may not exist in closed form.

- The antiderivative may define an unfamiliar function.
- The antiderivative may exist, but the software can't find it.

Nevertheless, in many cases, MATLAB can perform symbolic integration successfully.

```
q = integral(fun,xmin,xmax)
```

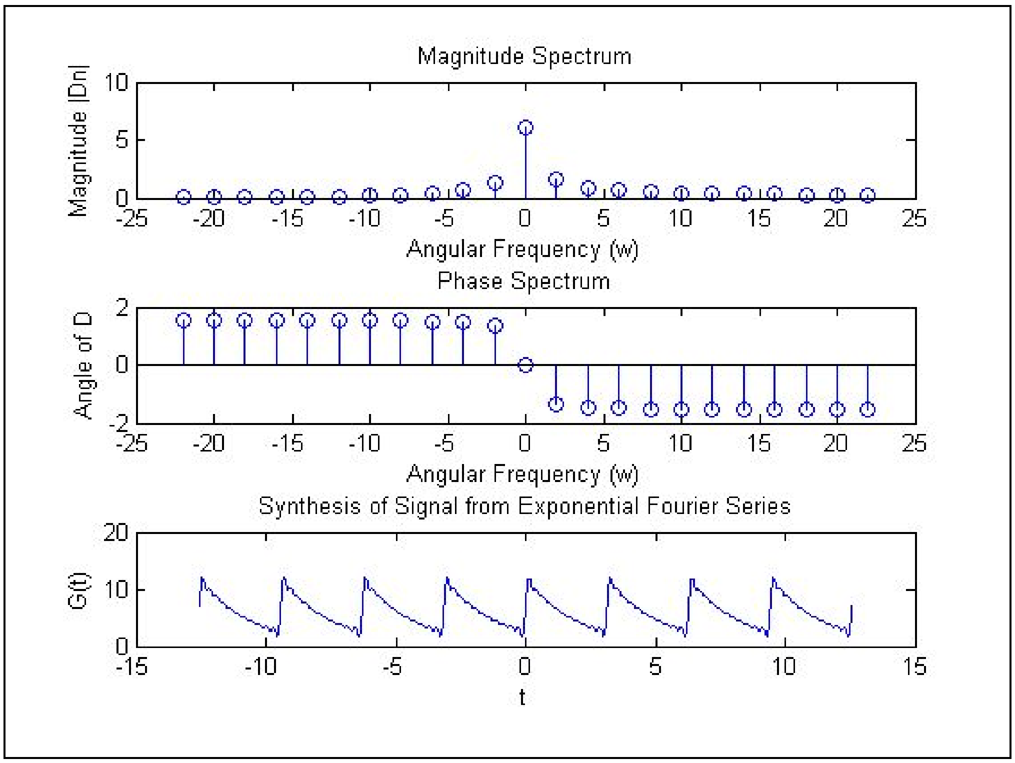
numerically integrates function `fun` from `xmin` to `xmax` using global adaptive quadrature and default error tolerances.

Code.

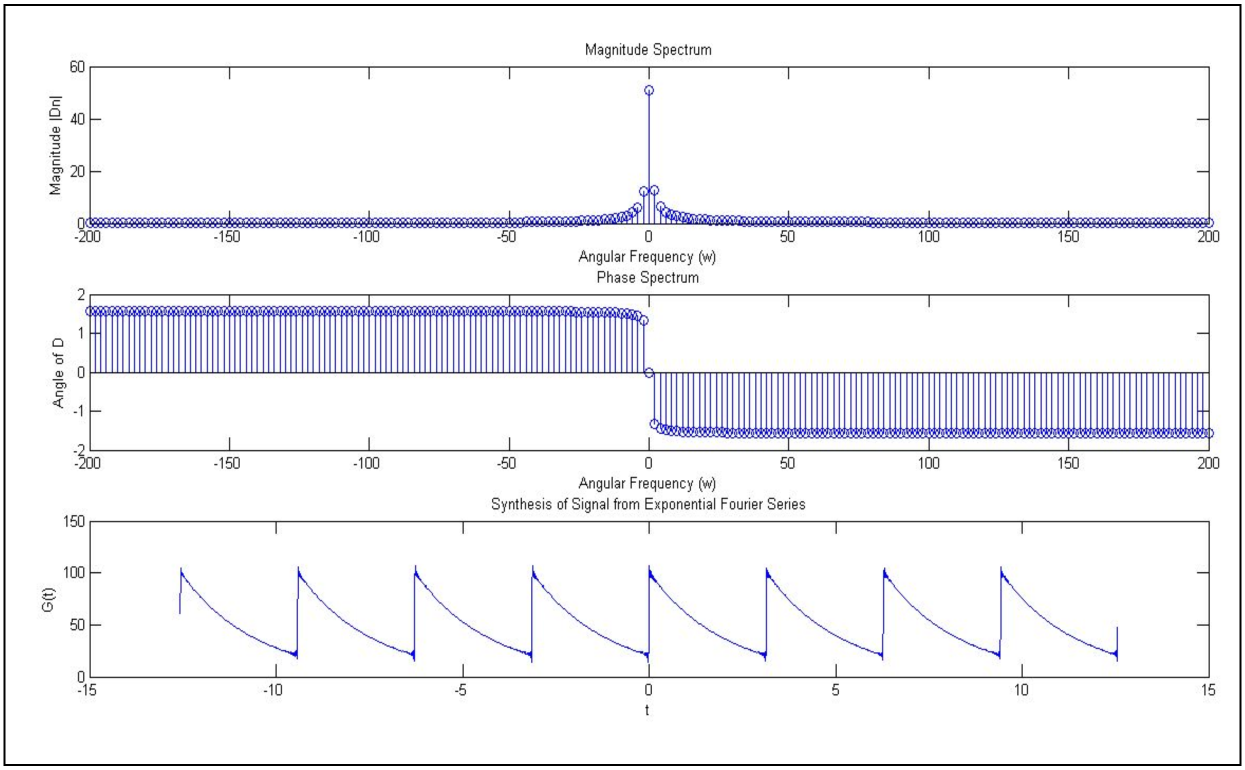
1. Exponent Function

```
clc;
clear all;
close all;

T0 = 2 * pi;
N = 100;
% N = 11;
w0 = 2*pi / T0;
t = -4 * pi : 0.01 : 4 * pi;
w = (-N: N) * (2 * pi / T0);

for i = 1:length(w)
      D(i) = i / T0 * integral(@(t)exp(-t/2).*exp(-1j*w(i)*t), 0, T0);
end

figure;
subplot(3, 1, 1);
stem(w, abs(D));
xlabel('Angular Frequency (w)');
ylabel('Magnitude |Dn|');
title('Magnitude Spectrum');

subplot(3, 1, 2);
stem(w, angle(D));
xlabel('Angular Frequency (w)');
ylabel('Angle of D');
title('Phase Spectrum');
```

```
sum = 0;
for i = 1:length(w)
      sum = sum + D(i)*exp(j*w(i)*t);
end

subplot(3, 1, 3);
plot(t, sum);
xlabel('t');
ylabel('G(t)');
title('Synthesis of Signal from Exponential Fourier Series');
```



Magnitude and Phase Spectrum for Exponent Curve N = 11

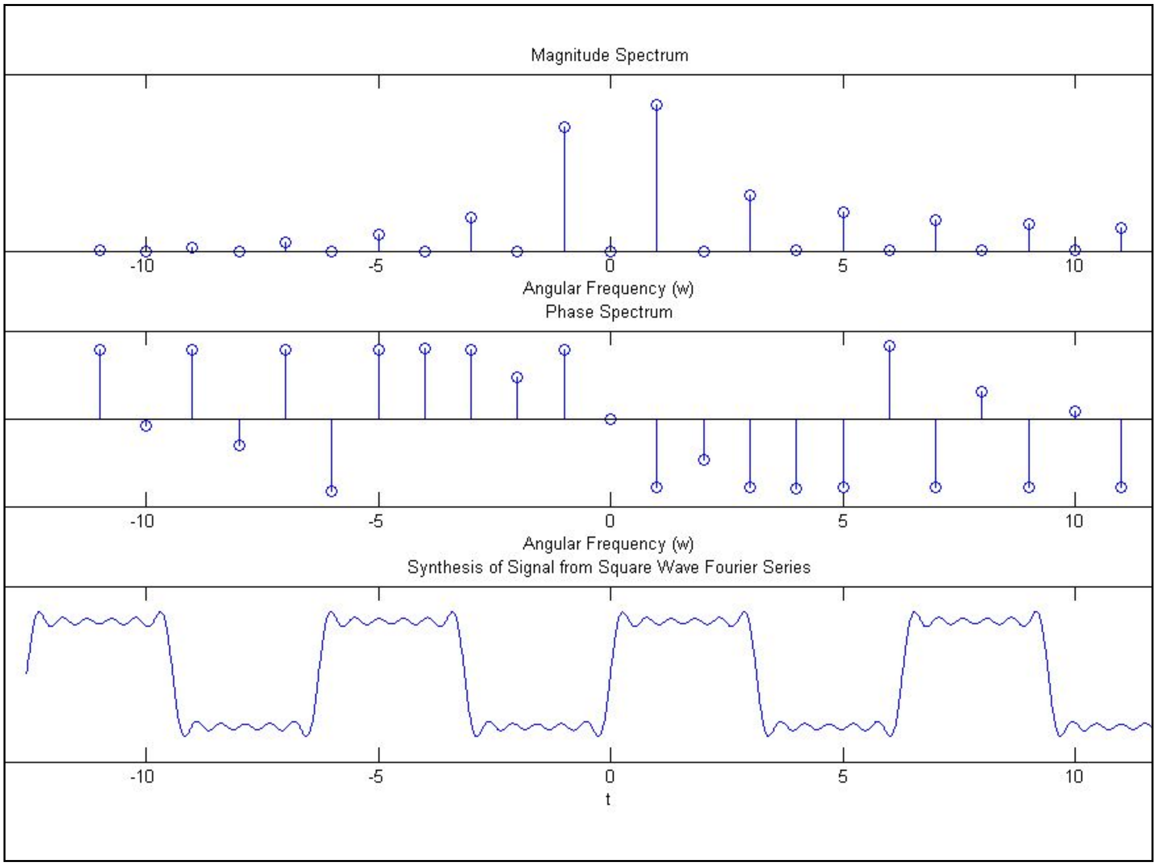


Magnitude and Phase Spectrum for Exponent Curve N = 100

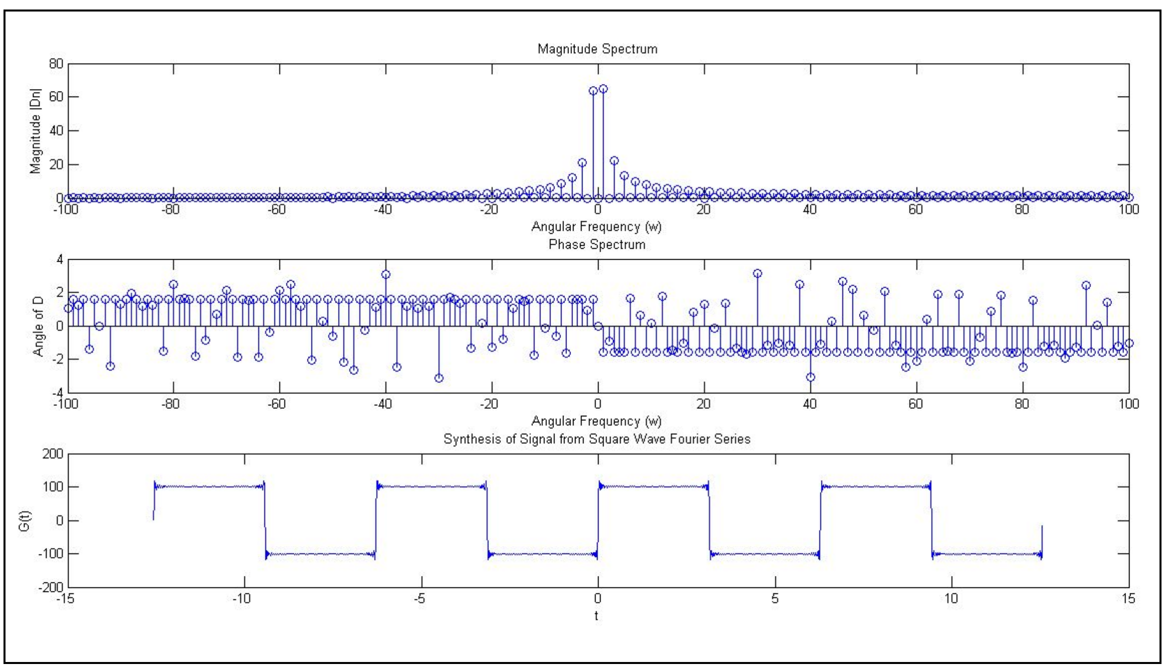
2. Square Wave

```
clc;
clear all;
close all;

T0 = 2 * pi;
N = 100;
% N = 11;
w0 = 2*pi / T0;
t = -4 * pi : 0.01 : 4 * pi;
w = (-N: N) * (2 * pi / T0);

for i = 1:length(w)
      D(i) = i / T0 * integral(@(t)square(t).*exp(-1j*w(i)*t), 0, T0);
end

figure;
subplot(3, 1, 1);
stem(w, abs(D));
xlabel('Angular Frequency (w)');
```

```
ylabel('Magnitude |Dn|');
title('Magnitude Spectrum');

subplot(3, 1, 2);
stem(w, angle(D));
xlabel('Angular Frequency (w)');
ylabel('Angle of D');
title('Phase Spectrum');

sum = 0;
for i = 1:length(w)
      sum = sum + D(i)*exp(j*w(i)*t);
end

subplot(3, 1, 3);
plot(t, sum);
xlabel('t');
ylabel('G(t)');
title('Synthesis of Signal from Square Wave Fourier Series');
```



Magnitude and Phase Spectrum for Square Wave N = 11



Magnitude and Phase Spectrum for Square Wave N = 100

Conclusion.

In this experiment we computed the Fourier Coefficients of Exponent and Square Wave Functions. Using these coefficients, we plotted the Phase Spectrum, Magnitude Spectrum and Synthesised the Original Signals.

Remarks. Signature.