Consider the square wave, $x(t)$, shown below.

Find the Fourier series representation. Find the power in the signal. Find the percent power in the first three harmonics.

The signal $x(t)$ is applied as input to a low-pass RC filter with transfer function $H(s) = \frac{1000}{s + 1000}$. Find the Fourier series of the output of the filter, $y(t)$. Plot $x(t)$ and $y(t)$ using Matlab.

**Period:** $T = \ldots$

**Fundamental Frequency:** $\omega_0 = \ldots$

**Fourier Series Coefficients:**

$a_0 = \ldots$

$a_n = \ldots$

$b_n = \ldots$
Solution: $T = 10$ ms, $\omega_0 = \frac{2\pi}{T} = 200\pi$.

$$a_0 = 0$$

$$a_n = 0, \text{since } x(t) \text{ is odd.}$$

$$b_n = \frac{2}{T} \int_{-T/2}^{T/2} x(t) \sin(n\omega_0 t) \, dt = \frac{2}{T} \left[ \int_{0}^{T/2} (-5) \sin(n\omega_0 t) \, dt + \int_{0}^{T/2} (5) \sin(n\omega_0 t) \, dt \right]$$

$$= \frac{10}{\pi n} (1 - \cos(n\pi)).$$

$$P = \frac{1}{T} \int_{-T/2}^{T/2} |x(t)|^2 \, dt = 25$$

Power in first three harmonics:

- $0.5 b_1^2 = 0.5 \times \frac{20^2}{\pi^2} = 20.26$
- $0.5 b_2^2 = 0$
- $0.5 b_3^2 = 0.5 \times \frac{20^2}{(3\pi)^2} = 2.25$

Percent power in first 3 harmonics is $\frac{20.26 + 2.25}{25} \times 100 = 90\%$.

Filtering:

The signal $x(t)$ is applied as input to a low-pass RC filter with transfer function $H(s) = \frac{1000}{s + 1000}$. Find the Fourier series of the output of the filter, $y(t)$.

Solution: Evaluate the filter transfer function at the harmonic frequencies.

$$H(jn\omega_0) = \frac{1000}{jn\omega_0 + 1000}$$

The sine/cosine form of $x(t)$ and $y(t)$ are as follows.

$$x(t) = a_0 + \sum_{n=1}^{\infty} a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t)$$

$$y(t) = H(j0)a_0 + \sum_{n=1}^{\infty} |H(jn\omega_0)| a_n \cos(n\omega_0 t + \beta_n) + |H(jn\omega_0)| b_n \sin(n\omega_0 t + \beta_n)$$

The magnitude/phase form of $x(t)$ and $y(t)$ are then

$$x(t) = a_0 + \sum_{n=1}^{\infty} A_n \cos(n\omega_0 t - \theta_n)$$

$$y(t) = H(j0)a_0 + \sum_{n=1}^{\infty} |H(jn\omega_0)| A_n \cos(n\omega_0 t - \theta_n + \beta_n)$$
Use Matlab to plot the truncated Fourier series,

\[ x_N(t) = a_0 + \sum_{n=1}^{N} a_n \cos(n\omega_0 t) + b_n \sin(n\omega_0 t), \]

and the truncated Fourier series for \( y(t) \),

\[ y_N(t) = H(j0)a_0 + \sum_{n=1}^{N} \left| H(jn\omega_0) \right| a_n \cos(n\omega_0 t + \beta_n) + \left| H(jn\omega_0) \right| b_n \sin(n\omega_0 t + \beta_n) \] or

\[ y_N(t) = \tilde{a}_0 + \sum_{n=1}^{N} \tilde{a}_n \cos(n\omega_0 t + \beta_n) + \tilde{b}_n \sin(n\omega_0 t + \beta_n) \]

where \( \tilde{a}_0 = H(j0)a_0, \quad \tilde{a}_n = \left| H(jn\omega_0) \right| a_n, \quad \tilde{b}_n = \left| H(jn\omega_0) \right| b_n \) are the new Fourier series coefficients for \( y(t) \).

Note that for a fixed number of terms in the truncated Fourier series (fixed value of \( N \)), the truncated Fourier series for \( y(t) \) appears much smoother than the truncated Fourier series for \( x(t) \). Why?

Matlab code

```matlab
function y = fs_filter(N)
% EE 321 example of filtering a Fourier series
T=0.01;
w0=2*pi/T;
t=[-T:T/1000:T];
x=zeros(1,length(t));
y=zeros(1,length(t));
k=[1:N];
b=(10/pi)*(1-cos(k*pi))./k;
% Filter \( H(jw) = \frac{1000}{jw + 1000} \)
for n=1:N
    x=x+b(n)*sin(2*pi*n*t/T);
    y=y+b(n)*abs(1000/(j*n*w0 + 1000))*sin(2*pi*n*t/T -atan(n*w0/1000));
end
plot(t,x,t,y,'--')
```

Note that for a fixed number of terms in the truncated Fourier series (fixed value of \( N \)), the truncated Fourier series for \( y(t) \) appears much smoother than the truncated Fourier series for \( x(t) \). Why?