

09.2 four.fsex a Complex Fourier series example

1 There are several flavors of Fourier series problem: trigonometric/exponential, analysis/synthesis, plotting partial sums/plotting spectra. Of course, problems just present us an opportunity to traverse part of the landscape (to mix two metaphors like 31 similies).

Example 09.2 four.fsex a-1

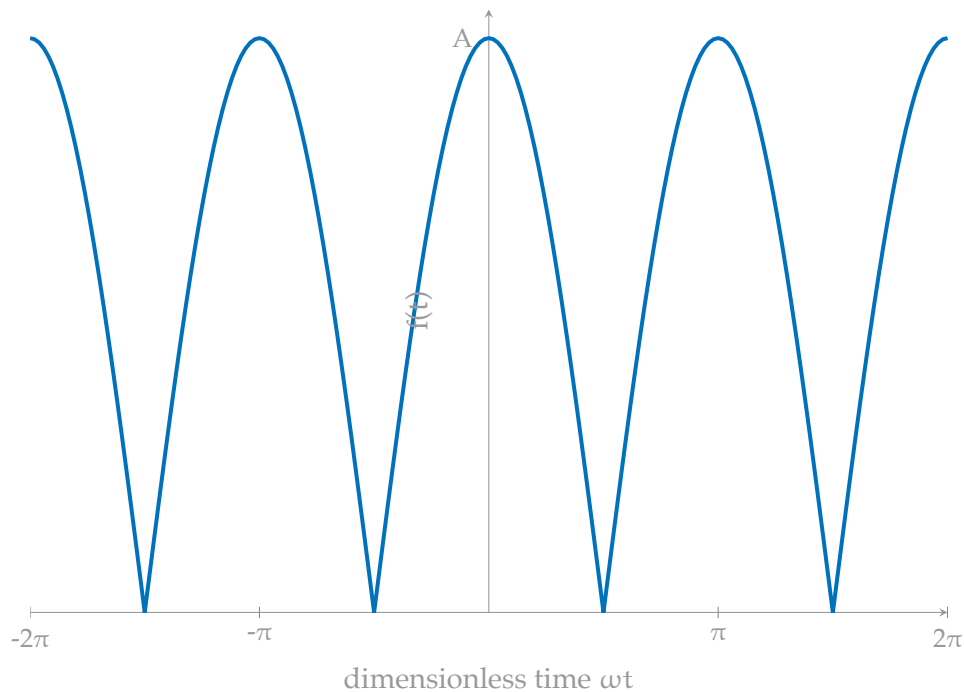


Figure fsex a.1: the function $f(t) = |A \cos(\omega t)|$ plotted for several periods.

2 Consider a rectified sinusoid

$$f(t) = |A \cos(\omega t)|$$

••• for $A, \omega, t \in \mathbb{R}$, shown in Fig. fsex.1. The fundamental period is $T = \pi/\omega$, half the unrectified period.

- a. Perform a complex Fourier analysis on $f(t)$, computing the complex Fourier components $c_{\pm n}$.
- b. Compute and plot the magnitude and phase spectra.
- c. Convert $c_{\pm n}$ to trigonometric components a_n and b_n .

Part a: complex Fourier analysis

3 The complex Fourier analysis of Definition 09 four.2 will be applied in a moment. However, it is convenient to first convert f into an . We can write f over a single period $t \in [-T/2, T/2)$ as

(absolute value property)

(already positive)

(Euler, Eq. 2)

••• 4 Applying Fourier analysis à la Definition 09 four.2 with harmonic

frequency $\omega_n = 2\pi n/T$,

$$\begin{aligned}
 c_{\pm n} &= \frac{1}{T} \int_{-T/2}^{T/2} f(t) e^{-j\omega_n t} dt \\
 &= \frac{1}{T} \int_{-T/2}^{T/2} |A| \frac{1}{2} (e^{j\omega t} + e^{-j\omega t}) e^{-j\omega_n t} dt \\
 &= \frac{|A|}{2T} \int_{-T/2}^{T/2} (e^{j\omega t} + e^{-j\omega t}) e^{-j\omega_n t} dt \\
 &= \frac{|A|}{2T} \int_{-T/2}^{T/2} (e^{j(\omega - \omega_n)t} + e^{-j(\omega + \omega_n)t}) dt \\
 &= \frac{|A|}{2T} \left(\frac{1}{j(\omega - \omega_n)} e^{j(\omega - \omega_n)t} - \frac{1}{j(\omega + \omega_n)} e^{-j(\omega + \omega_n)t} \right) \Big|_{-T/2}^{T/2} \\
 &= \frac{|A|}{2T} \left(\frac{1}{j(\omega - \omega_n)} e^{j(\omega - \omega_n)T/2} - \frac{1}{j(\omega + \omega_n)} e^{-j(\omega + \omega_n)T/2} + \right. \\
 &\quad \left. - \frac{1}{j(\omega - \omega_n)} e^{-j(\omega - \omega_n)T/2} + \frac{1}{j(\omega + \omega_n)} e^{j(\omega + \omega_n)T/2} \right) \\
 &= \frac{|A|}{j2T(\omega - \omega_n)} (e^{j(\omega - \omega_n)T/2} - e^{-j(\omega - \omega_n)T/2}) + \\
 &\quad + \frac{|A|}{j2T(\omega + \omega_n)} (e^{j(\omega + \omega_n)T/2} - e^{-j(\omega + \omega_n)T/2}) \\
 &= \frac{|A|}{T(\omega - \omega_n)} \sin((\omega - \omega_n)T/2) + \frac{|A|}{T(\omega + \omega_n)} \sin((\omega + \omega_n)T/2).
 \end{aligned}$$

5 This can be simplified further if we substitute $T = \pi/\omega$ and $\omega_n = 2\pi n/T = 2n\omega$,



6 Using a product-to-sum trigonometric identity ([Appendix 01.2 math.trig](#)), this further simplifies to

$$c_{\pm n} = \frac{-2|A|}{\pi(4n^2 - 1)} \cos(\pi n),$$

• which, for n odd or even,

$$c_{\pm n} = \begin{cases} \frac{2|A|}{\pi(4n^2-1)} & n \text{ odd} \\ \frac{-2|A|}{\pi(4n^2-1)} & n \text{ even.} \end{cases}$$

7 Alternatively we could use Matlab's Symbolic Math Toolbox rather straightforwardly.

```
syms A n w wn T t 'real' % symbolic, real
```

8 Now define the function of time f and the known relations in a dictionary.

```
f = abs(A)*cos(w*t);
props.T = pi/w;
props.wn = 2*n*w;
```

9 Now apply the same Fourier analysis as before.

```
c_n1 = 1/T*int(f*exp(-j*wn*t),t,-T/2,T/2);
c_n = simplify(subs(c_n1,props))
```

```
c_n =

-(2*cos(pi*n)*abs(A))/(pi*(4*n^2 - 1))
```

10 Nice! This is the _____. We can even check our odd/even assumptions.

```
assume((n-1)/2,'integer') % odd
simplify(c_n)
assume(n,'clear') % clear assumptions
assume(n/2,'integer') % even
simplify(c_n)
assume(n,'clear') % clear before moving on
assume(n,'real')
```

```

ans =

(2*abs(A))/(pi*(4*n^2 - 1))

ans =

-(2*abs(A))/(pi*(4*n^2 - 1))

```

11 These are also what we got before.

Parb b: harmonic amplitude and phase with spectra

12 According to Eq. 12, the harmonic amplitude is

$$C_n = 2\sqrt{c_n c_{-n}}$$

$$= \frac{4|A|}{\pi|4n^2 - 1|} |\cos(\pi n)|$$

13 Let's check with Matlab.

```

assume(n, 'real');
C_n = simplify(2*sqrt(c_n*subs(c_n,n,-n)))
assume(n, 'clear');
assume(n, 'integer');
C_n = simplify(2*sqrt(c_n*subs(c_n,n,-n)))

```

```

C_n =

(4*abs(A)*abs(cos(pi*n)))/(pi*abs(4*n^2 - 1))

C_n =

(4*abs(A))/(pi*abs(4*n^2 - 1))

```

14 We see that if we assume n is an integer, C_n simplifies even further than we took it by-hand.

- 15 Plotting the harmonic amplitude is straightforward. First make C_n something that can be numerically evaluated and choose parameters.

```
p.A = 1;
C_n_fun = matlabFunction( ...
    subs(C_n, p) ...
);
```

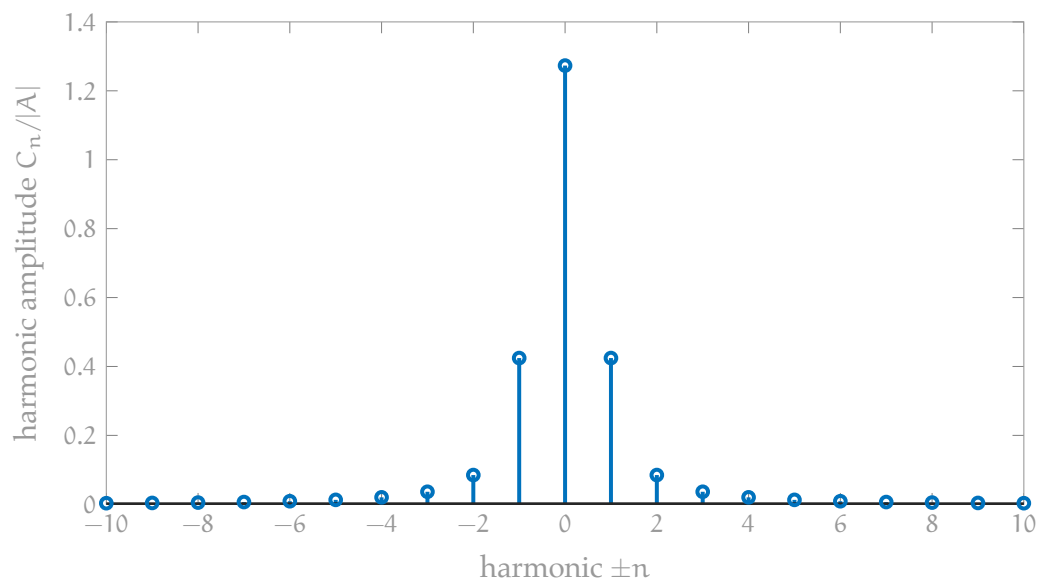


Figure fsex.2: the harmonic amplitude C_n .

- 16 Now we plot.

```
n_a = -10:10;
figure
stem(n_a, C_n_fun(n_a))
xlabel('\pm n')
ylabel('harmonic amplitude C_n/|A|')
```

- 17 Let's find the phase *à la* Eq. 13 with Matlab directly.

```
phase_n = simplify(atan2(imag(c_n),real(c_n)))
```

```
phase_n =
```

```
(pi*sign((-1)^n*abs(A))/(4*n^2 - 1))*(sign((-1)^n*abs(A))/(4*n^2 - 1) +  
↪ 1))/2
```

18 The sign function just returns the sign of its argument. It's difficult to see, but this expression only takes on the following two values:



19 We can plot the phase similarly to how we plotted the amplitude. First we get a numerically evaluable function.

```
phase_n_fun = matlabFunction( ...  
    subs(phase_n, p) ...  
);
```

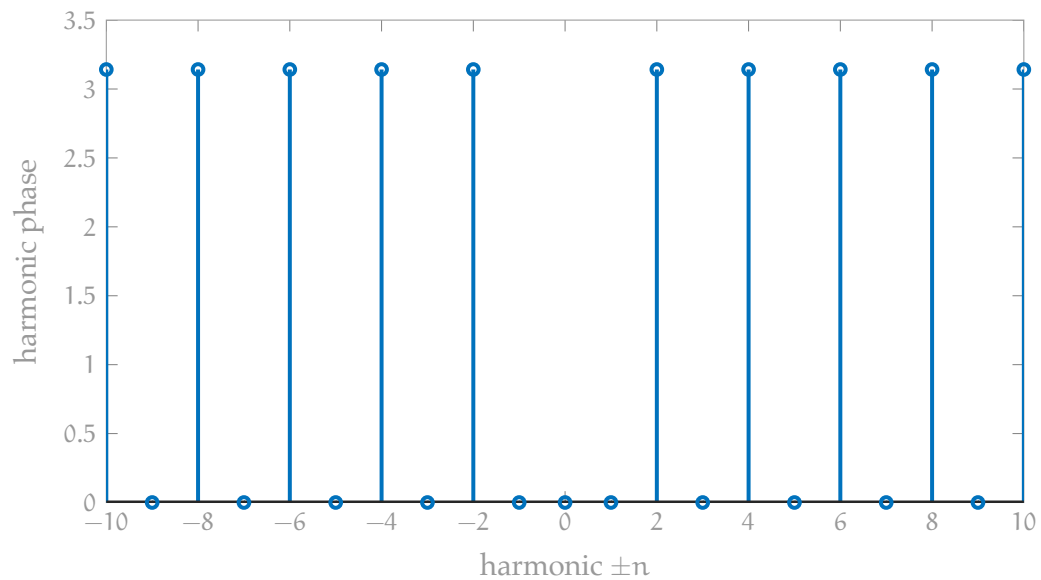


Figure fsex.3: the harmonic phase.

20 Now we plot.

```
figure
stem(n_a, phase_n_fun(n_a))
xlabel('\pm n')
ylabel('harmonic phase')
```

Part c: conversion to trig form

21 According to [Definition 09 four.3](#), the trigonometric components can be computed from the complex components as follows.

```
a_n = simplify(c_n + subs(c_n, n, -n))
b_n = simplify(j*(c_n - subs(c_n, n, -n)))
```

```
a_n =
```

```
-(4*(-1)^n*abs(A))/(pi*(4*n^2 - 1))
```



$$b_n =$$

$$0$$

22 The fact that $b_n = 0$ should not surprise us: $f(t)$ is *even* after all!