Fourier transform

Eugeniy E. Mikhailov

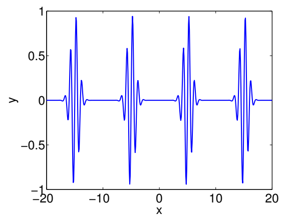
The College of William & Mary

Lecture 25

Notes

Fourier series

Any periodic single value function with a finite number of discontinuities, and for which $\int_0^T |f(t)|dt$ is finite, can be presented as



$$y(t) = \frac{a_0}{2} + \sum_1^\infty (a_n \cos(n\omega_1 t) + b_n \sin(n\omega_1 t))$$

T is the period, i.e., $y(t) = y(t+T)$

$\omega_1 = 2\pi/T$ is the fundamental frequency

$$\begin{pmatrix} a_n \\ b_n \end{pmatrix} = \frac{2}{T}\int_0^T dt \begin{pmatrix} \cos(n\omega_1 t) \\ \sin(n\omega_1 t) \end{pmatrix} y(t)$$

At a discontinuity, the series approaches the mid point

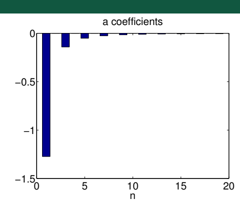
Notes

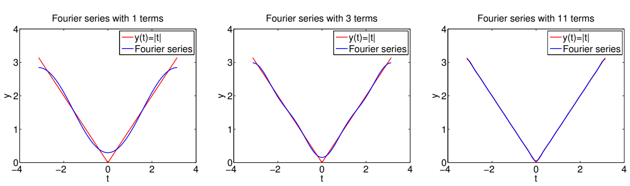
Fourier series example: |t|

$y(t) = |t|, \quad -pi < t < pi$

Since the function is even all $b_n = 0$

$$\begin{cases} a_0 = \pi, & \\ a_n = 0, & n \text{ is even} \\ a_n = -\frac{4}{\pi n^2}, & n \text{ is odd} \end{cases}$$





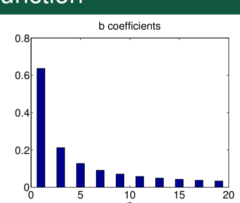
Notes

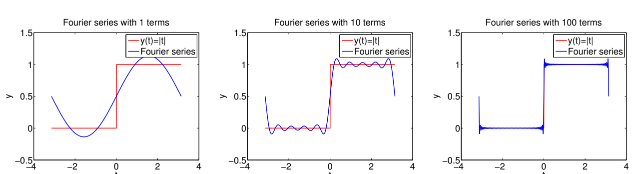
Fourier series example: step function

$$\begin{cases} 0, & -\pi < x < 0, \\ 1, & 0 < x < \pi \end{cases}$$

Since the function is odd all $a_n = 0$ except $a_0 = 1$

$$\begin{cases} b_n = 0, & n \text{ is even} \\ b_n = \frac{2}{\pi n}, & n \text{ is odd} \end{cases}$$





Notes

Complex representation

Recall that

$$\exp(i\omega t) = \cos(\omega t) + i\sin(\omega t)$$

It can be shown that

$$y(t) = \sum_{n=-\infty}^{\infty} c_n \exp(in\omega_1 t)$$
$$c_n = \frac{1}{T}\int_0^T y(t)\exp(-i\omega_1 nt)dt$$

$$a_n = c_n + c_{-n}$$
$$b_n = i(c_n - c_{-n})$$

Notes

What to do if function is not periodic?

- $T \to \infty$
- $\sum \to \int$
- discrete spectrum $\to$ continuous spectrum
  - $c_n \to c_\omega$

$$y(t) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} c_\omega \exp(i\omega t)d\omega$$
$$c_\omega = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty} y(t)\exp(-i\omega t)dt$$

Above requires that $\int_{-\infty}^{\infty} dt\, y(t)$ exists and is finite.
Note that c_ω has the extra $\sqrt{2\pi}$ when compared to c_n, and T is gone.

Notes

Discrete Fourier transform (DFT)

In reality, we cannot have

- infinitely large interval
- infinite amount of points to calculate true integral

Assuming that $y(t)$ has a period T and we took N equidistant points

$$\Delta t = \frac{T}{N} \text{ samples spacing, } f_s = \frac{1}{\Delta t} \text{ sampling rate}$$
$$f_1 = \frac{1}{T} = \frac{1}{N\Delta t} \text{ smallest observed frequency, also resolution bandwidth}$$
$$t_k = \Delta t \times (k-1)$$
$$y(t_{k+N}) = y(t_k) \text{ periodicity condition}$$
$$y_k = y(t_k) \text{ shortcut notation}$$
$$y_1, y_2, y_3, \cdots, y_N \text{ data set}$$

We replace the integral in the Fourier series with the sum

Notes

DFT

$$y_k = \frac{1}{N}\sum_{n=0}^{N-1} c_n \exp\left(i\frac{2\pi(k-1)n}{N}\right) \text{ inverse Fourier transform}$$
$$c_n = \sum_{k=1}^{N} y_k \exp\left(-i\frac{2\pi(k-1)n}{N}\right) \text{ Fourier transform}$$
$$n = 0, 1, 2, \cdots, N-1$$

Confusion keeps increasing: where are the negative coefficients c_{-n} ?
In DFT, they moved to the right end of the c_n vector :

$$c_{-n} = c_{N-n}$$

Notes

Fast Fourier transform (FFT)

Fast numerical realization of DFT is FFT. This is just the smart way to do DFT. Matlab has one built in

- y is a matlab vector of data points (y_k)
- `c=fft(y)` Fourier transform
- `y=ifft(c)` inverse Fourier transform

Notice that `fft` does not normalize by N. So to get Fourier series c_n, you need to calculate `fft(y)/N`.
However `y = ifft( fft(y) )`
Notice one more point of confusion: Matlab does not have index=0, so actual $c_n = c_{matlab\ fft}(n-1)$, so $c_0 = c_{matlab\ fft}(1)$

Notes

Notes

Notes

Notes