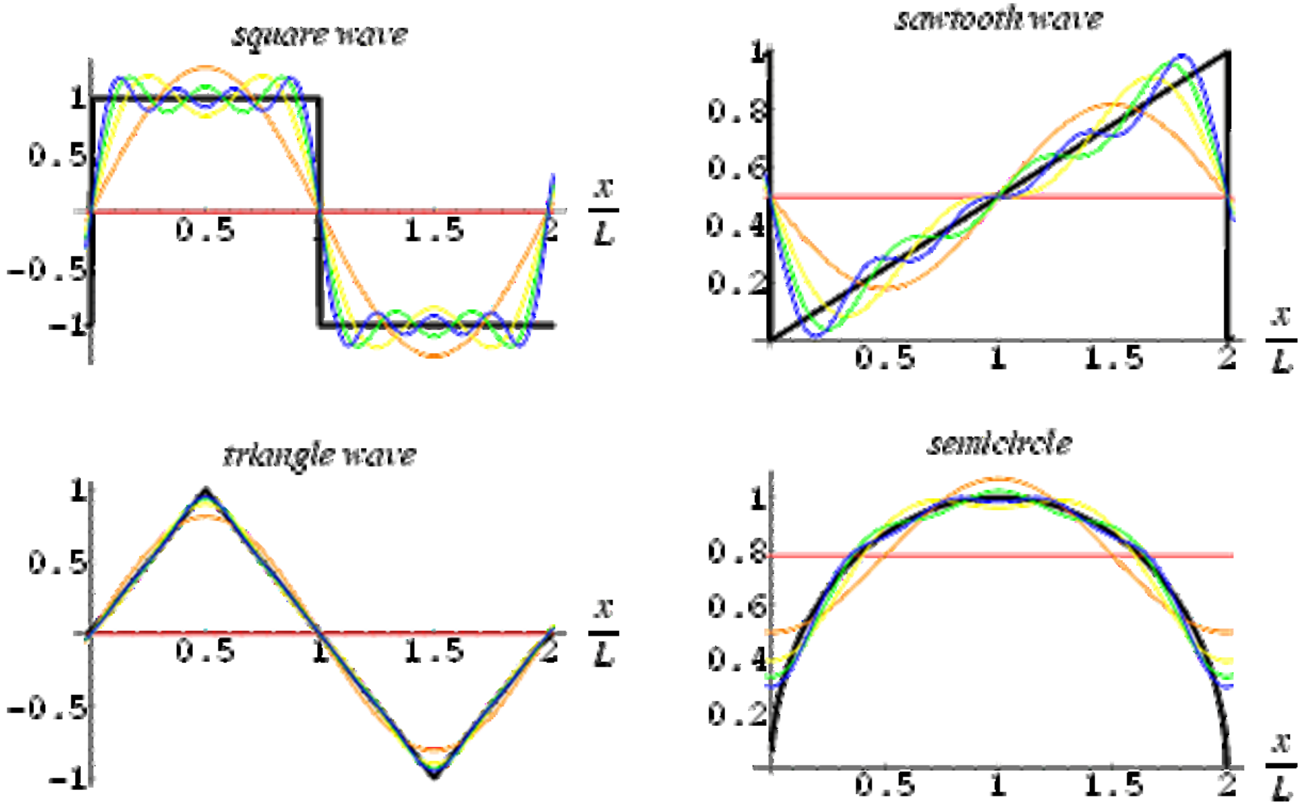
Fourier Transform based signal analysis

All periodic waveforms can be described as an infinite series of sines and cosines with different amplitudes



$$y(t) = A_o + \sum_{n=1}^{\infty} \left(A_n \cos(nt) + B_n \sin(nt) \right)$$

The linear coefficients of a periodic waveform are determined by integrating the series from -π to π

$$y(t) = A_o + \sum_{n=1}^{\infty}\left(A_n\, cos(nt) + B_n\, sin(nt)\right)$$

$$\int_{-\pi}^{\pi} y(t)\,dt = A_o \int_{-\pi}^{\pi} dt + \sum_{n=1}^{\infty}\left(A_n \int_{-\pi}^{\pi} cos(nt)\,dt + B_n \int_{-\pi}^{\pi} sin(nt)\,dt\right)$$

$$A_o = \frac{1}{2\pi}\int_{-\pi}^{\pi} y(t)\,dt \left[Because \int_{-\pi}^{\pi} cos(nt)\,dt = 0;\ \int_{-\pi}^{\pi} sin(nt)\,dt = 0\right]$$

$$A_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} y(t)\, cos(nt)\,dt \left[Multiply\ by\ cos(mt),\ integrate\right]$$

$$B_n = \frac{1}{2\pi}\int_{-\pi}^{\pi} y(t)\, sin(nt)\,dt \left[Multiply\ by\ sin(mt),\ integrate\right]$$

If we have a function with an arbitrary period, T, we integrate from –T/2 to T/2

$$y(t) = A_o + \sum_{n=1}^{\infty}\left(A_n\, cos(n\omega t) + B_n\, sin(n\omega t)\right)$$

$$A_o = \frac{1}{2T}\int_{-T/2}^{T/2} y(t)\,dt$$

$$A_n = \frac{1}{2T}\int_{-T/2}^{T/2} y(t)\,cos(n\omega t)\,dt$$

$$B_n = \frac{1}{2T}\int_{-T/2}^{T/2} y(t)\,sin(n\omega t)\,dt$$

If we remove the constraint that the function is periodic, we integrate from -∞ to ∞ and get coefficients that are a function of ω only. This is the Fourier Transform

$$A(\omega)=\int_{-\infty}^{\infty} y(t)\cos(\omega t)\,dt$$

$$B(\omega)=\int_{-\infty}^{T/2} y(t)\sin(\omega t)\,dt$$

$$Y(\omega)=A(\omega)-iB(\omega)$$

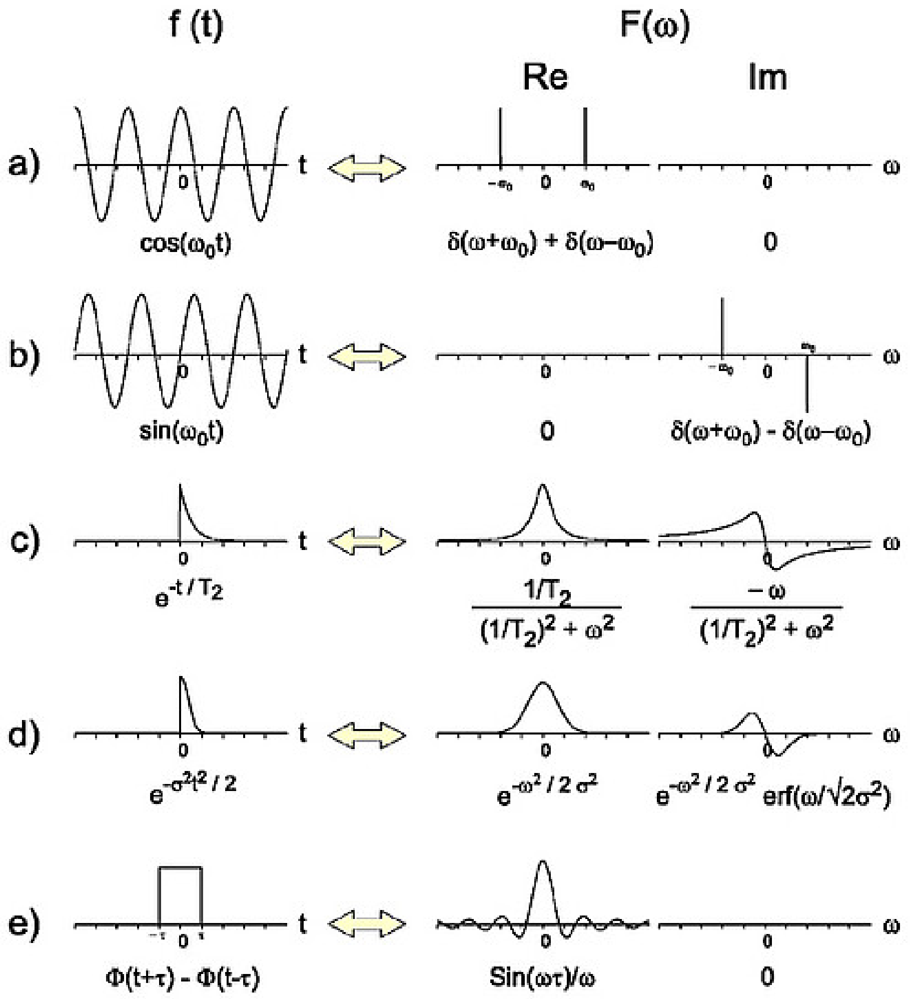
$$Y(\omega)=\int_{-\infty}^{\infty} y(t)\left(\cos\omega t-i\sin(\omega t)\right)dt$$

$$Y(\omega)=\int_{-\infty}^{\infty} y(t)e^{-i\omega t}\,dt$$

$$Y(f)=\int_{-\infty}^{\infty} y(t)e^{-i2\pi ft}\,dt$$

$$y(t)=\int_{-\infty}^{\infty} Y(f)e^{-i2\pi ft}\,df$$

A number of analytic functions can be explicitly integrated to Fourier space. They are called Fourier Transform pairs



The discrete Fourier Transform (DFT) is motivated by the fact that the N data points are recorded at discrete intervals.

$$\left\{y\left(r\,\delta t\right)\right\} = y\left(t\right)\delta\left(t - r\delta t\right) \quad r = 0,1,2,3...,N-1$$

$$Y\left(f_k\right) = \frac{2}{N}\sum_{r=0}^{N/2-1} y\left(r\delta t\right)e^{-i2\pi rk/N}$$

$$f_k = k\delta f \quad k = 0,1,2,3.....N/2-1$$

$$\delta f = 1/N\delta t$$

Consider calculating the fifth frequency component of 10 points recorded of a waveform

$$Y\left(f_k\right)=\frac{2}{N}\sum_{r=0}^{N/2-1} y\left(r\delta t\right)e^{-i2\pi rk/N}$$

$$k=0,1,2,3.....N/2-1$$

$$Y\left(f_0\right)=\frac{2}{10}\left(y\left(\delta t\right)e^{-i2\pi 0/10}+y\left(2\delta t\right)e^{-i2\pi(2)0/10}+y\left(3\delta t\right)e^{-i2\pi(3)0/10}+y\left(4\delta t\right)e^{-i2\pi(4)0/10}\right)$$

$$=\frac{2}{10}\left(y\left(\delta t\right)+y\left(2\delta t\right)+y\left(3\delta t\right)+y\left(4\delta t\right)\right)$$

$$Y\left(f_1\right)=\frac{2}{10}\left(y\left(\delta t\right)e^{-i2\pi/10}+y\left(2\delta t\right)e^{-i2\pi(2)/10}+y\left(3\delta t\right)e^{-i2\pi(3)/10}+y\left(4\delta t\right)e^{-i2\pi(4)/10}\right)$$

$$Y\left(f_2\right)=\frac{2}{10}\left(y\left(\delta t\right)e^{-i2\pi 2/10}+y\left(2\delta t\right)e^{-i2\pi(2)2/10}+y\left(3\delta t\right)e^{-i2\pi(3)2/10}+y\left(4\delta t\right)e^{-i2\pi(4)2/10}\right)$$

$$Y\left(f_3\right)=\frac{2}{10}\left(y\left(\delta t\right)e^{-i2\pi 3/10}+y\left(2\delta t\right)e^{-i2\pi(2)3/10}+y\left(3\delta t\right)e^{-i2\pi(3)3/10}+y\left(4\delta t\right)e^{-i2\pi(4)3/10}\right)$$

$$Y\left(f_4\right)=\frac{2}{10}\left(y\left(\delta t\right)e^{-i2\pi 4/10}+y\left(2\delta t\right)e^{-i2\pi(2)4/10}+y\left(3\delta t\right)e^{-i2\pi(3)5/10}+y\left(4\delta t\right)e^{-i2\pi(4)5/10}\right)$$

DFT vs. FFT

This approach requires $O(N^2)$ computations. The Fast Fourier Transform is an algorithm that provides the same information with $O(N\log(N))$ computations making it significantly faster.

The FFT algorithm requires that one has 2^n data points. The data array is "padded" with zeros to ensure the correct number of points. This can cause some numerical artifacts.

The frequency resolution is determined by the period of data acquisition. The algorithm cannot detect partial waveform components. The maximum frequency is based on the requirement that there are at least two data points per frequency component. So, f_{max}=(N/2-1)/sampling period~(N/2)/sampling period.

The amplitude and phase at a given frequency are given by:

$$\left|Y\left(f_k\right)\right| = \sqrt{Re\left|Y\left(f_k\right)\right|^2 + Im\left|Y\left(f_k\right)\right|^2}$$

$$\phi\left(f_k\right) = tan^{-1}\frac{Im\left|Y\left(f_k\right)\right|}{Re\left|Y\left(f_k\right)\right|}$$