Algorithms and Data Structures

COMP261
Fast Fourier Transform 1

Yi Mei

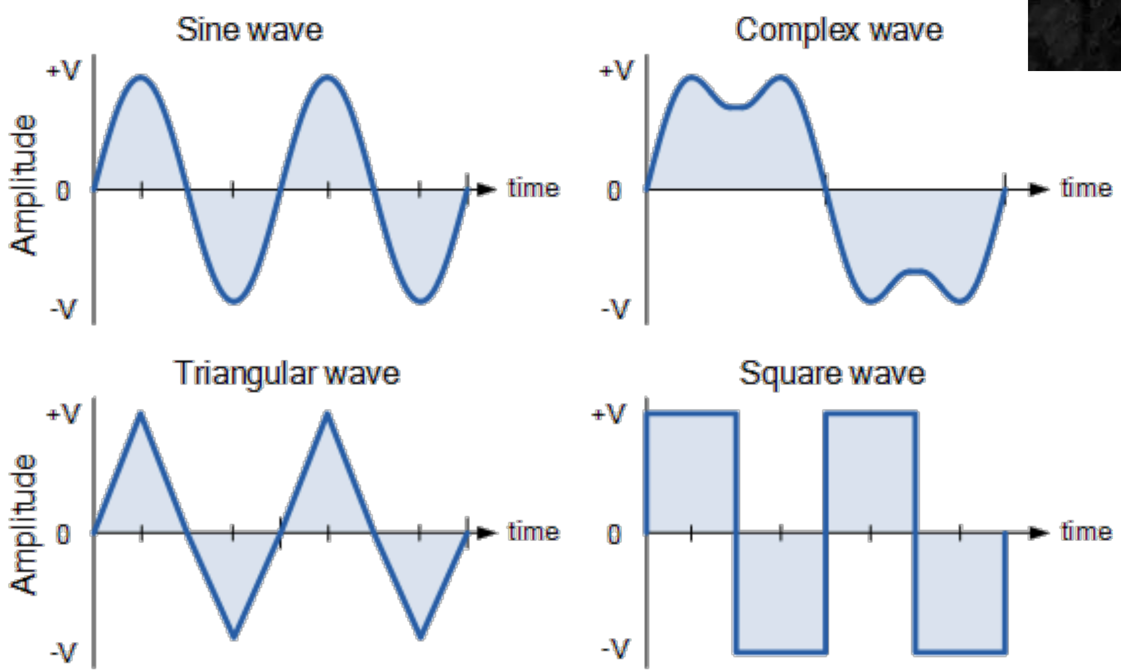
yi.mei@ecs.vuw.ac.nz

Outline

- Waveform (Signal) processing
- Fourier Series (Periodic waveform)
  - Real numbers
  - Complex numbers
- Fourier Transform (General waveform)
- Discrete Fourier Transforms

Waveform Processing

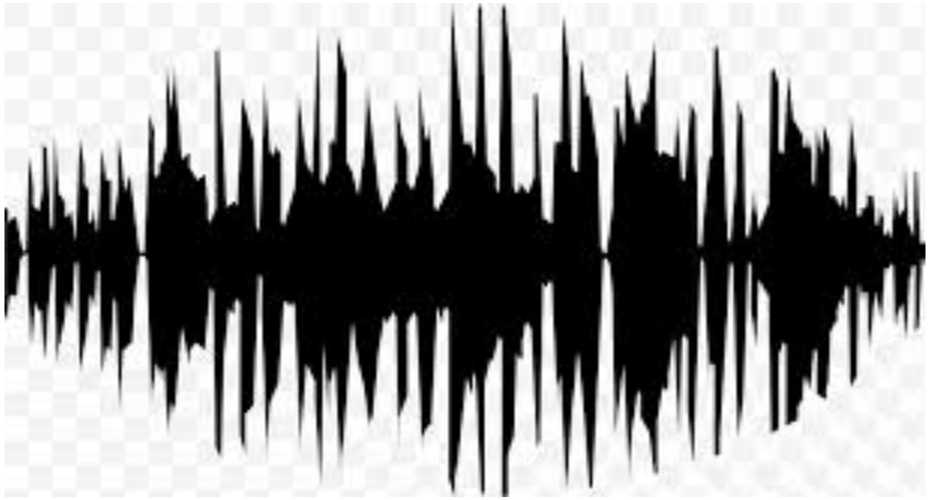
- Virtually everything in the world can be described via a waveform
  - Sound waves
  - Electromagnetic fields
  - Stock price series



Duc de Broglie

Waveform Processing

- Two representations

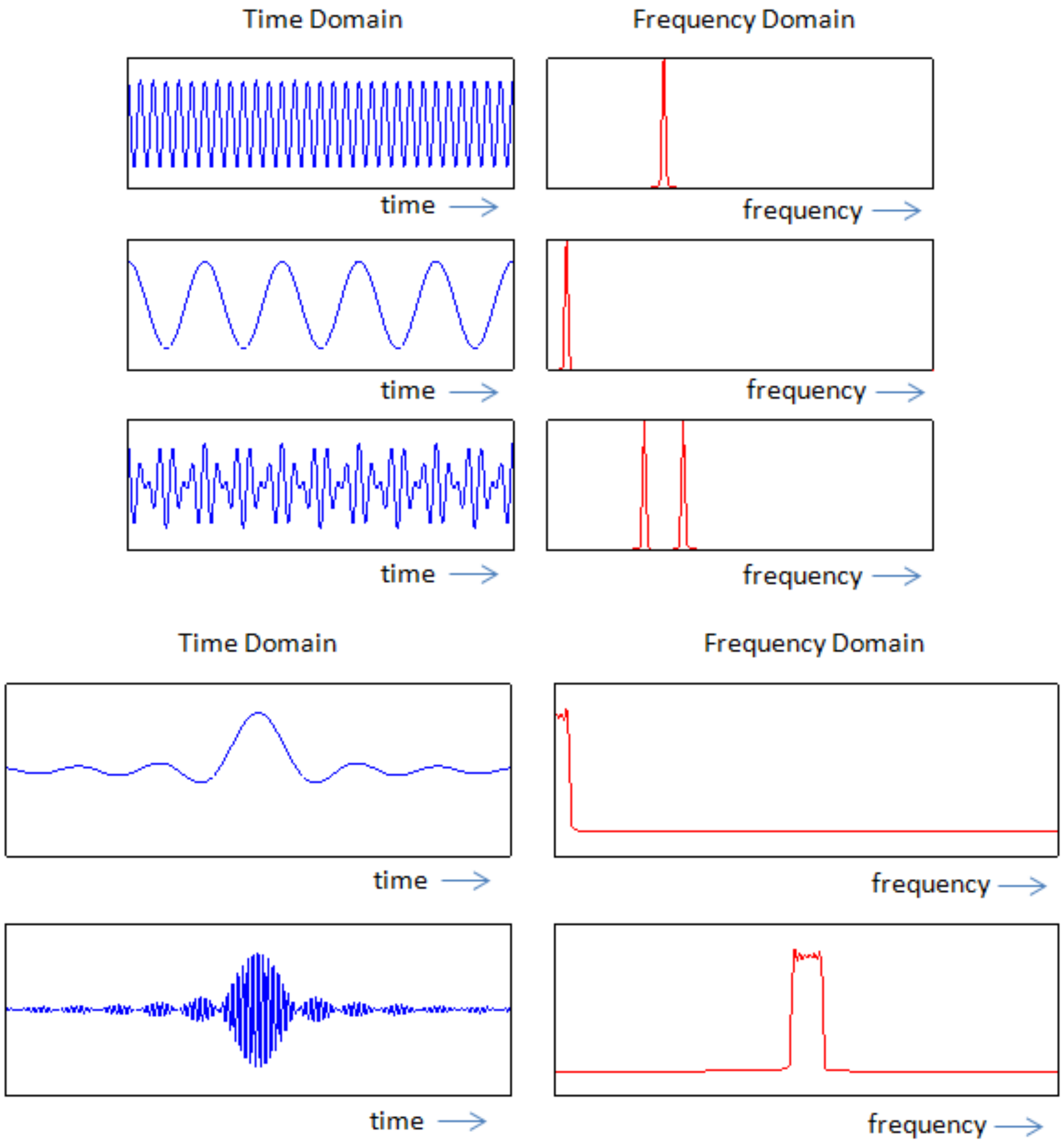


Time domain

Frequency domain

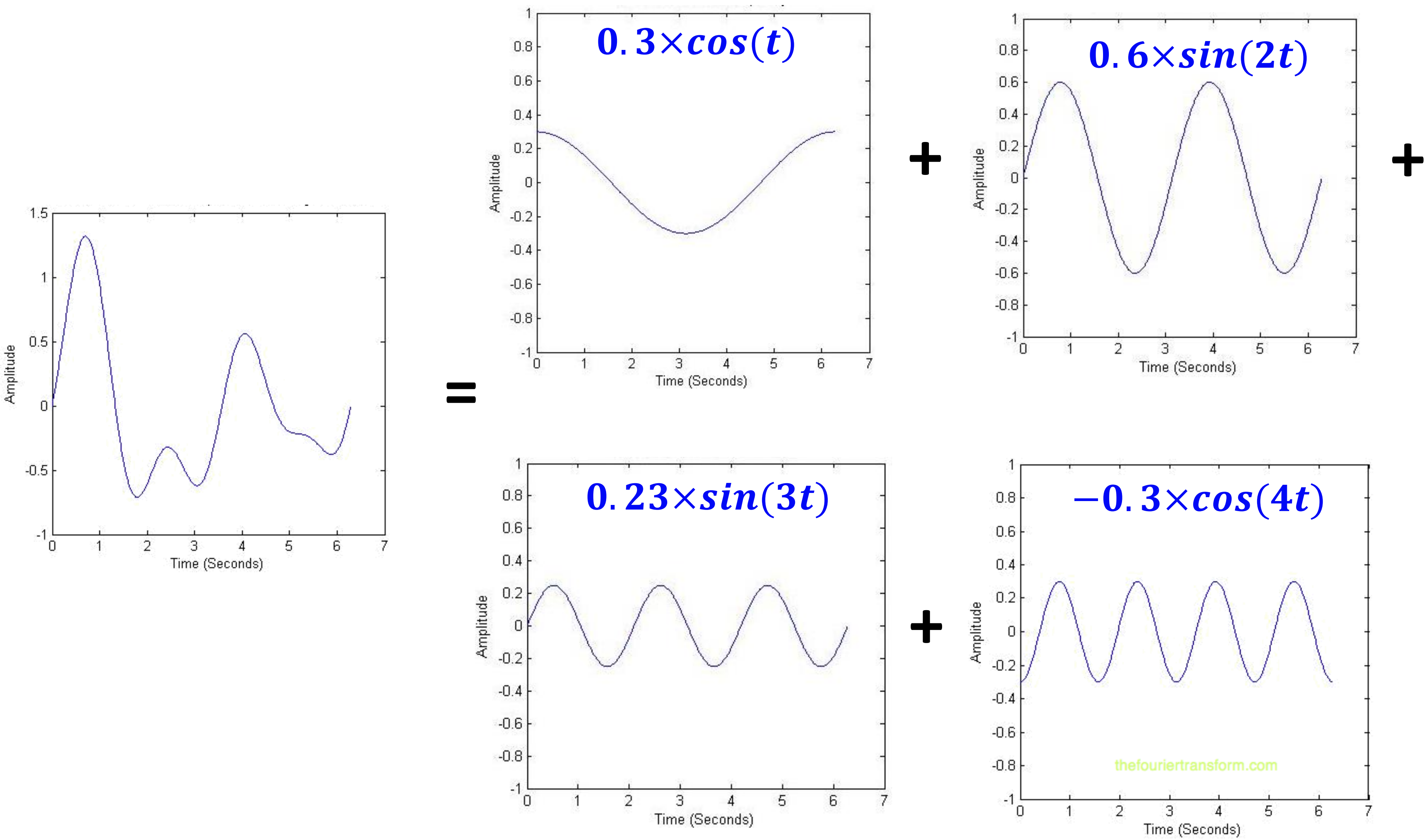
Waveform Processing

- Time domain = Frequency domain
  - **Fourier Transform**



Fourier Transform

- **Any** waveform can be decomposed into a number of sinusoids (sine/cosine functions)



Coefficient Calculation

- If we know the waveform has a period of $\boldsymbol{T}$

Fourier series: $$f(t) = a_0 + \sum_{m=1}^{\infty} a_m \cos\left(\boldsymbol{m} \times t \times \frac{2\pi}{T}\right) + \sum_{n=1}^{\infty} b_n \sin\left(\boldsymbol{n} \times t \times \frac{2\pi}{T}\right)$$

Coefficient (average): $$a_0 = \frac{1}{T}\int_0^T f(t)dt$$

Coefficient (correlation): $$a_m = \frac{2}{T}\int_0^T f(t)\cos(mt\frac{2\pi}{T})\,dt$$

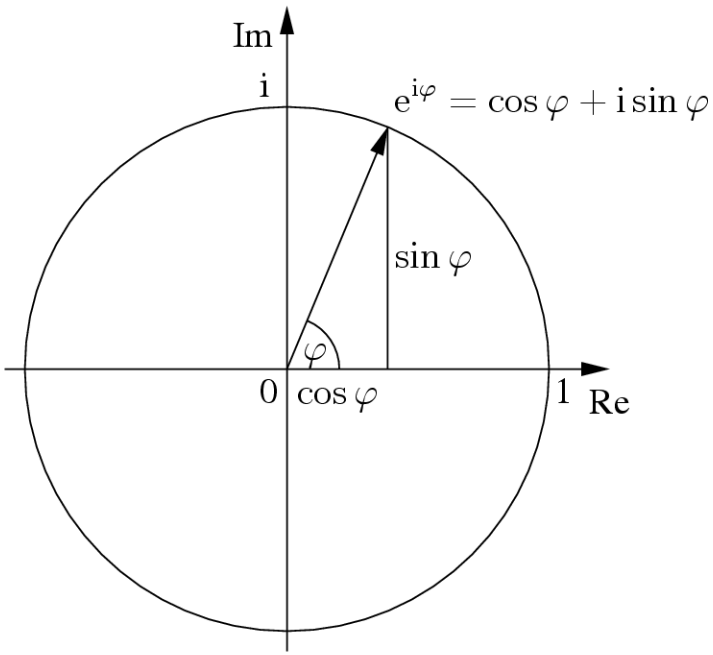
$$b_n = \frac{2}{T}\int_0^T f(t)\sin(nt\frac{2\pi}{T})\,dt$$

A Better Representation

- Complex number
  - $x = x.Re + i * x.Im$ (Re = real part, Im = imaginary part)
  - $i = \sqrt{-1}$ $(i^2 = -1)$

https://hackaday.com/2015/09/17/visualizing-the-fourier-transform/

- Euler's formula
  - $\boldsymbol{e^{it} = \cos t + i \cdot \sin t}$



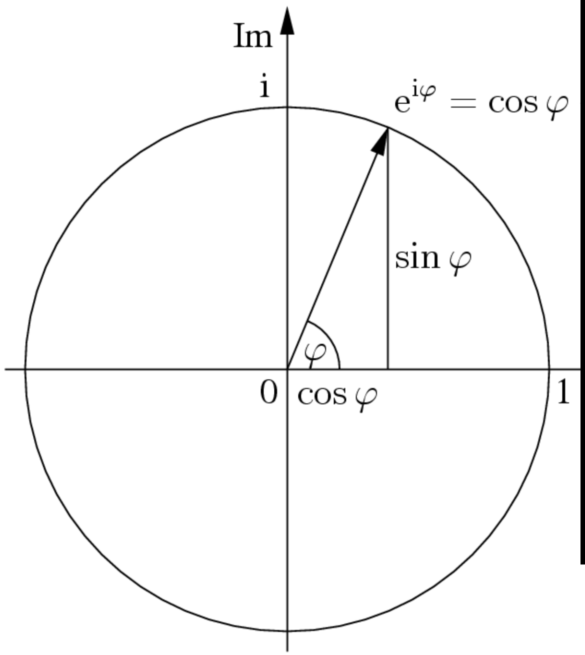
Leonhard Euler
(1707-1783)

A Better Representation

- Complex number
  - $x = x.Re + i * x.Im$ (Re = real part, Im = imaginary part)
  - $i = \sqrt{-1}$ $(i^2 = -1)$

https://hackaday.com/2015/09/17/visualizing-the-fourier-transform/

- Euler's formula
  - $e^{it} = \cos t + i$ [redacted]



Leonhard Euler
(1707-1783)

A Better Representation

- Rewrite into complex number

Fourier series: $$f(t) = a_0 + \sum_{m=1}^{\infty} a_m \cos\left(\boldsymbol{m} \times t \times \frac{2\pi}{T}\right) + \sum_{n=1}^{\infty} b_n \sin\left(\boldsymbol{n} \times t \times \frac{2\pi}{T}\right)$$

Euler formula: $$\boldsymbol{e^{it} = \cos t + i \cdot \sin t}$$

$$f(t) = \sum_{n=-\infty}^{\infty} c_n e^{i*n*t*\frac{2\pi}{T}}$$

$$c_n = \frac{1}{T}\int_0^T f(t) e^{-i*n*t*\frac{2\pi}{T}} dt$$

General Fourier Transform

- If we don't know the period (or non-periodic)
  - Infinite period: $T \rightarrow \infty$
  - Use frequency ω rather than period, so ω can be 0

Fourier transform:

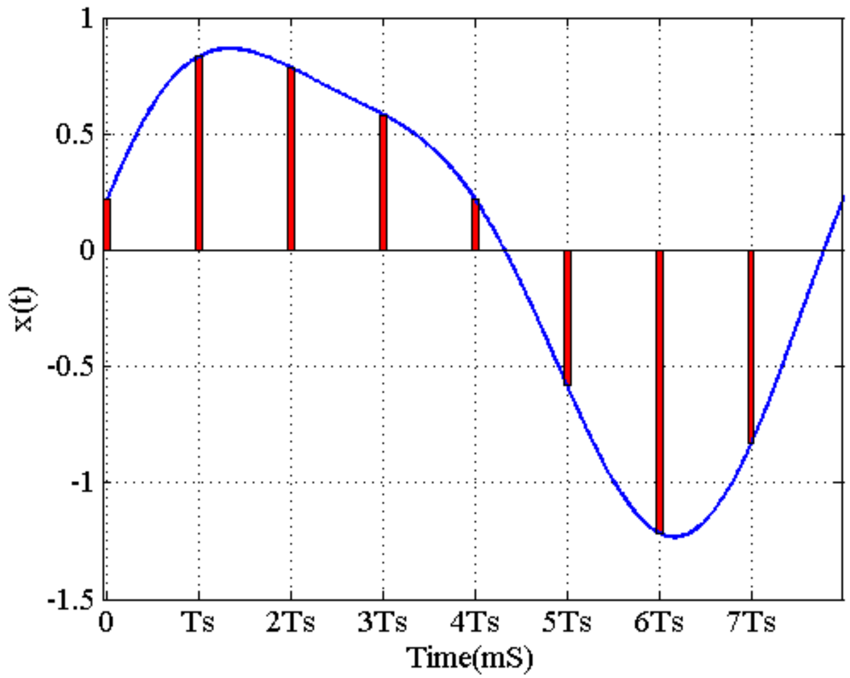
$$f(t) = \int_{-\infty}^{\infty} F(\omega) e^{i*\omega*t} d\omega$$

Inverse Fourier transform:

$$F(\omega) = \frac{1}{2\pi} \int_{-\infty}^{\infty} f(t) e^{-i*\omega*t} dt$$

Discrete Fourier Transform

- Everything in computer is discrete
- Sample signals at a certain frequency
  - E.g. Every T seconds
  - Frequency $f = \frac{1}{T}$



n	0	1	2	3	4	5	6	7
x(n)	0.2165	0.8321	0.7835	0.5821	0.2165	-0.5821	-1.2165	-0.8321

Discrete Fourier Transform

- Given a discrete sequence with N samples $[x(0), x(1), \dots, x(N-1)]$

- Time -> Frequency (Discrete-Time Fourier Transform)

$$X(k) = \sum_{n=0}^{N-1} x(n) e^{-i*n*k*\frac{2\pi}{N}}, k = 0,1, \dots, N-1$$

- Frequency -> Time (Discrete Fourier Transform)

$$x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{i*n*k*\frac{2\pi}{N}}, n = 0, 1, \dots, N-1$$

Summary

- Fourier Series (periodic continuous)
- Complex number representation
- Non-periodic – general series
- Discrete Fourier Transform