10/26/20

Today's lecture

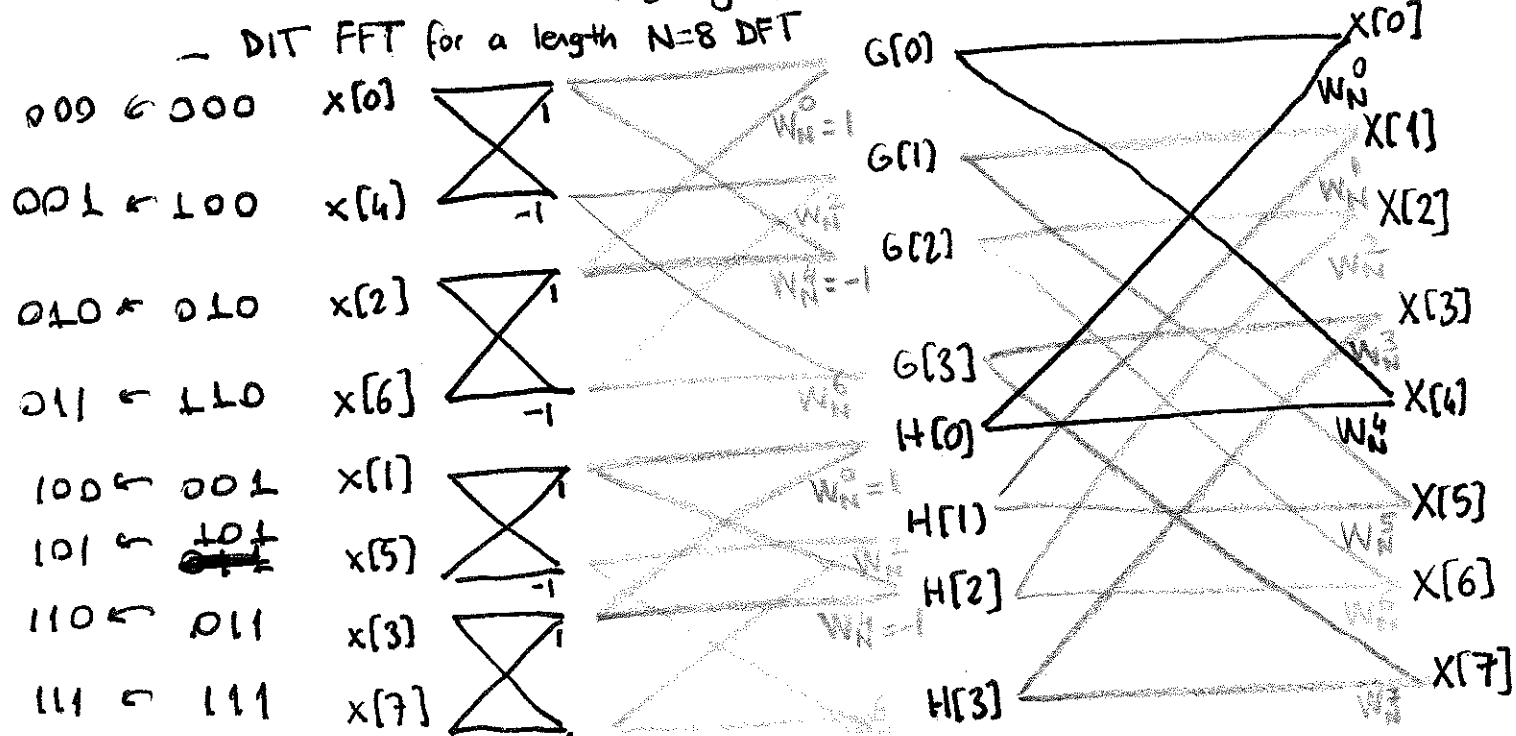
- Decimation in frequency (DIF) FFT
- The sampling theorem

Readings : 8-1, 8-2 FFT
6-1, 6-2 Sampling

Question on Piazza: Post the solutions today.

Last lecture : DIT FFT

- The radix-2 DIT : It divides a DFT of size N into 2 interleaved DFTs of size $\frac{N}{2}$ with each recursive stage.
- The radix-2 DIT first computes the DFTs of the even indexed input $x[2\ell]$ and odd indexed input $x[2\ell+1]$, $\ell = 0, 1, \dots, \frac{N}{2}-1$, and then combines those two results to produce the DFT.
- Applying radix-2 DIT recursively the overall runtime is reduced to $O(N\log N)$
- DIT FFT for a length N=8 DFT



$$\begin{array}{cccccccc} 0^{th} & & 2^{nd} & & 4^{th} & & 6^{th} & \end{array}$$

$$\begin{bmatrix} X[0] \\ X[1] \\ X[2] \\ X[3] \\ X[4] \\ X[5] \\ X[6] \\ X[7] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & W_8 & W_8^2 & W_8^3 & -1 & -W_8 & -W_8^2 & -W_8^3 \\ 1 & W_8^2 & -1 & -W_8^2 & 1 & W_8^2 & -1 & -W_8^2 \\ 1 & W_8^3 & -W_8^2 & W_8 & -1 & -W_8^3 & W_8^2 & -W_8 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 \\ 1 & -W_8 & W_8^2 & -W_8^3 & -1 & W_8 & -W_8^2 & W_8^3 \\ 1 & -W_8^2 & -1 & W_8^2 & 1 & -W_8^2 & -1 & W_8^2 \\ 1 & -W_8^3 & -W_8^2 & -W_8 & -1 & W_8^3 & W_8^2 & W_8 \end{bmatrix} \begin{bmatrix} x[0] \\ x[1] \\ x[2] \\ x[3] \\ x[4] \\ x[5] \\ x[6] \\ x[7] \end{bmatrix}$$

8x8 matrix

"Dftmtx"

$N=8$

Even Columns

$$\begin{bmatrix} I_{4\times4} \\ \\ I_{4\times4} \end{bmatrix} \underbrace{\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & W_8^2 & -1 & -W_8^2 \\ 1 & -1 & 1 & -1 \\ 1 & -W_8^2 & -1 & W_8^2 \end{bmatrix}}_{F_4} \quad \text{where} \quad I_{4\times4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

DFT for $N=4$

Odd Columns

$$\begin{bmatrix} I_{4\times4} \\ \\ -I_{4\times4} \end{bmatrix} \begin{bmatrix} 1 & 1 & 1 & 1 \\ W_8 & W_8^3 & -W_8 & -W_8^3 \\ W_8^2 & -W_8^2 & W_8^2 & -W_8^2 \\ W_8^3 & W_8 & -W_8^3 & -W_8 \end{bmatrix} = \begin{bmatrix} I_{4\times4} \\ \\ -I_{4\times4} \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & W_8 & 0 & 0 \\ 0 & 0 & W_8^2 & 0 \\ 0 & 0 & 0 & W_8^3 \end{bmatrix} F_4$$

$$F_8 = \underbrace{\begin{bmatrix} I_{4\times4} & I_{4\times4} \\ I_{4\times4} & -I_{4\times4} \end{bmatrix}}_{8\times8 \text{ matrix}} \underbrace{\begin{bmatrix} I_{4\times4} & O \\ O & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & W_8 & 0 & 0 \\ 0 & 0 & W_8^2 & 0 \\ 0 & 0 & 0 & W_8^3 \end{bmatrix} \end{bmatrix}}_{8\times8 \quad \text{'Twiddle factors'}} \underbrace{\begin{bmatrix} F_4 & 0 \\ 0 & F_4 \end{bmatrix}}_{8\times8}$$

What about when N is not even?

1 Cooley - Tukey FFT

FFT for general N

2 Good - Thomas FFT

FFT to get rid of the twiddle factors W_8^{lr}.

Decimation in Frequency (N even)

$$X[k] = \sum_{n=0}^{N-1} x[n] W_N^{nk}$$

Let's consider the even samples of $X[k]$:

$$X[2l] = \sum_{n=0}^{N-1} x[n] W_N^{n.2l}$$

$$= \sum_{n=0}^{N/2-1} x[n] W_N^{n.2l} + \sum_{n=N/2}^{N-1} x[n] W_N^{n.2l}$$

$$= \sum_{n=0}^{N/2-1} x[n] W_{N/2}^{nl} + \sum_{n=0}^{N/2-1} x[n+\tfrac{N}{2}] \underbrace{W_N^{(n+\frac{N}{2})2l}}_{W_N^{n2l} \cdot \underbrace{W_N^{Nl}}_{1}}$$

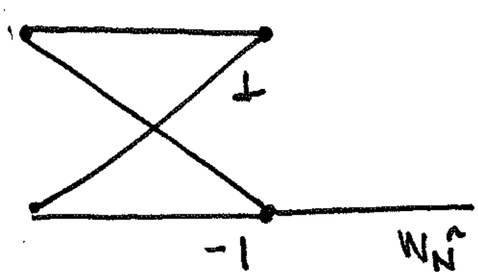
$W_N^{n2l} \rightarrow W_{N/2}^{nl}$

$$X[2\ell] = \sum_{n=0}^{N/2-1} \left(x[n] + x\left[n+\tfrac{N}{2}\right]\right) W_{N/2}^{n\ell}$$

which is like a N/2 DFT of the summed input (top half & bottom half)

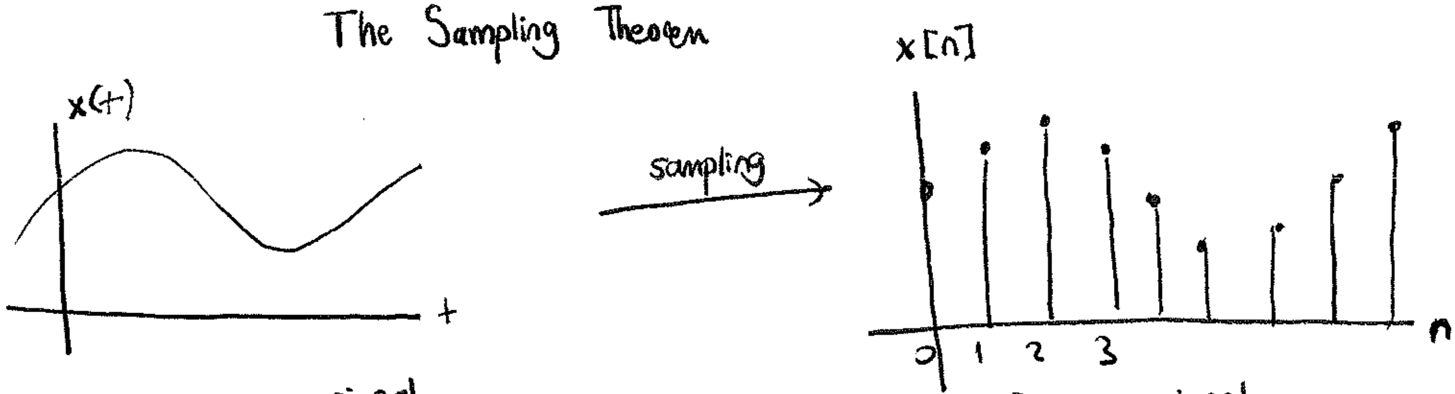
For the odd samples of $X[k]$ we have

$$X[2\ell+1] = \sum_{n=0}^{N/2-1} \left(x[n] - x\left[n+\tfrac{N}{2}\right]\right) \underbrace{W_N^n}_{\text{Twiddle factor}} W_{N/2}^{n\ell}$$



butterfly for DIF FFT

The Sampling Theorem



Continuous signal

Discrete signal

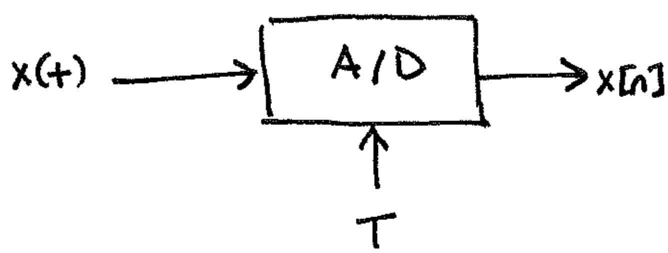
When the inputs are quantized
$\Rightarrow$ digital signal.

Periodic Sampling

$x[n] = x(nT)$ where n is an integer, T is sampling period

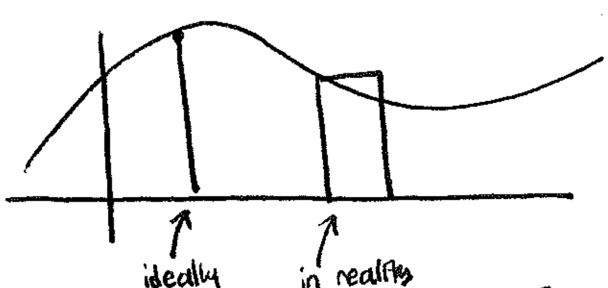
$w_s = \frac{2\pi}{T}$ (radians) sampling frequency

$f_s = \frac{1}{T}$ (Hertz, Hz) sampling frequency



Non-ideal effects

1. Ideally we multiply $x(t)$ with a shifted impulse train. Instead of sampling $x(t)$ we end up sampling $x(t) * h(t)$
(impulse response of the filter)



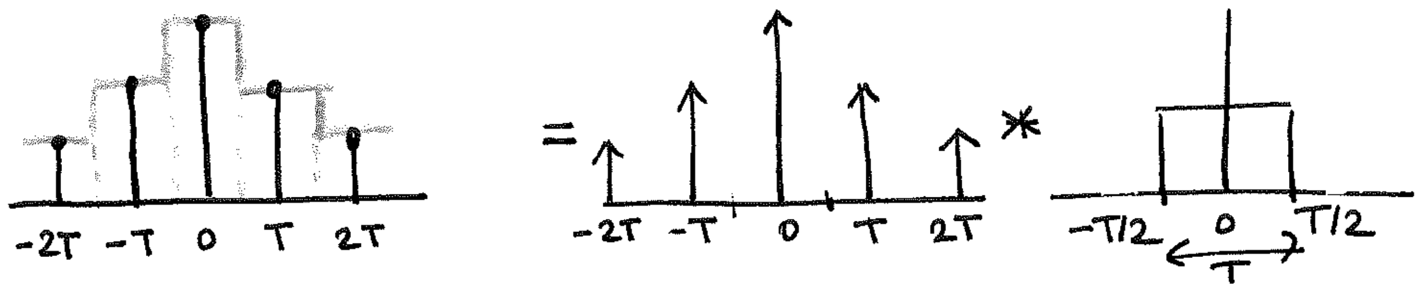
2. Noise or distortion

$x[n] = y(nT) + z[n]$
(noise)

Reconstructing CT signal $x_c(t)$ given $x[n]$.

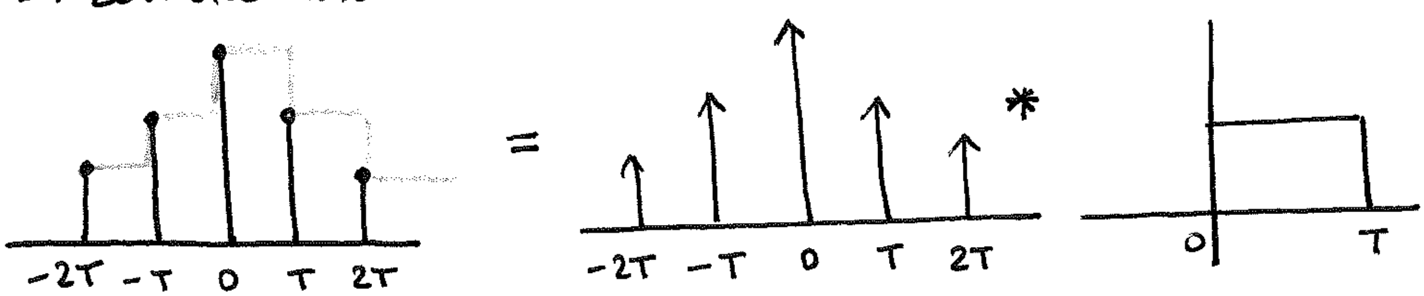
Assume $x[n] = x(nT)$

1. Nearest neighbor

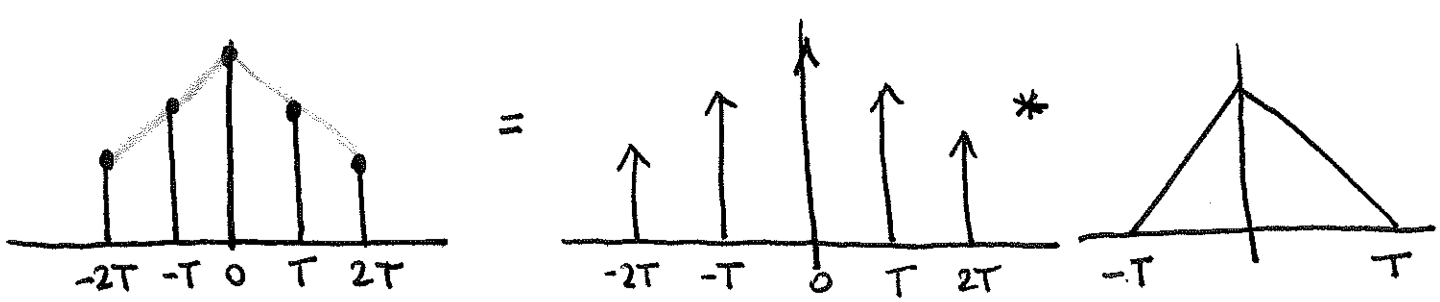


Non causal

2. Zero-order hold



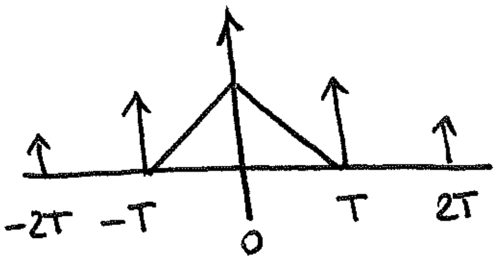
3. First-order hold



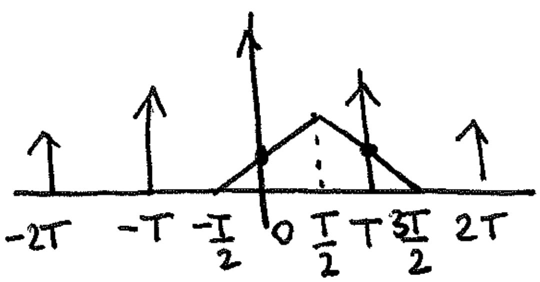
- Zero-order hold and nearest neighbor only consider individual samples.

- First-order hold linearly weights 2 consecutive samples:

$$y(t) = x(t) * h(t) = \int_{-\infty}^{+\infty} x(\tau)h(t-\tau)\,d\tau$$

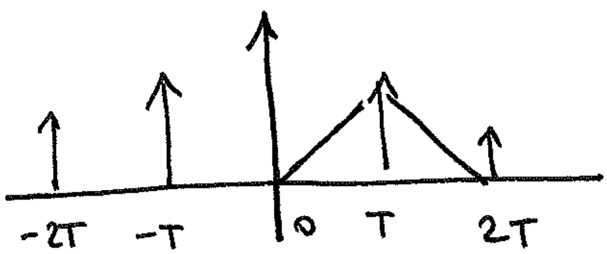


$$y(0) = \int_{-T}^{T} x(\tau)h(\tau)\,d\tau = x(0)$$



$$y\left(\tfrac{T}{2}\right) = \int_{-\frac{T}{2}}^{\frac{3T}{2}} x(\tau)h\left(\tau-\tfrac{T}{2}\right)d\tau$$

$$= \frac{x(0)+x(T)}{2}$$



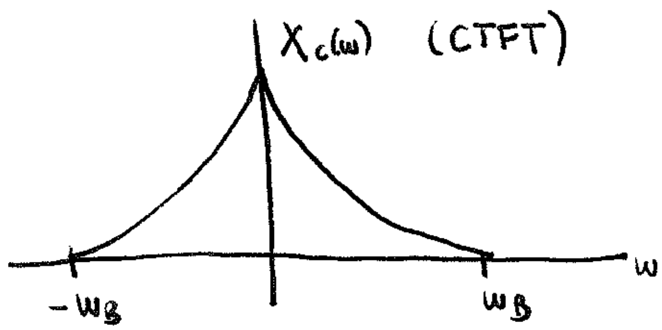
$$y(T) = \int_{0}^{2T} x(\tau)\,h(\tau-T)\,d\tau$$

$$= x(T)$$

Question: What is the correct interpolator?

Answer: Sinc interpolation

- We require the input signal to be bandlimited.

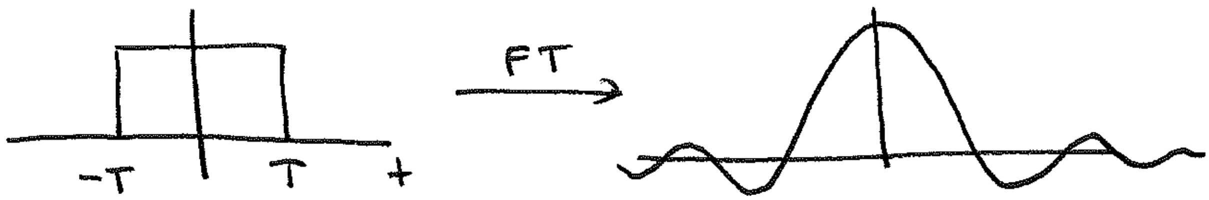


$$X_c(\omega) = 0, \quad |\omega| > \omega_B$$

* If we have finite duration in time (time-limited) → not band-limited.

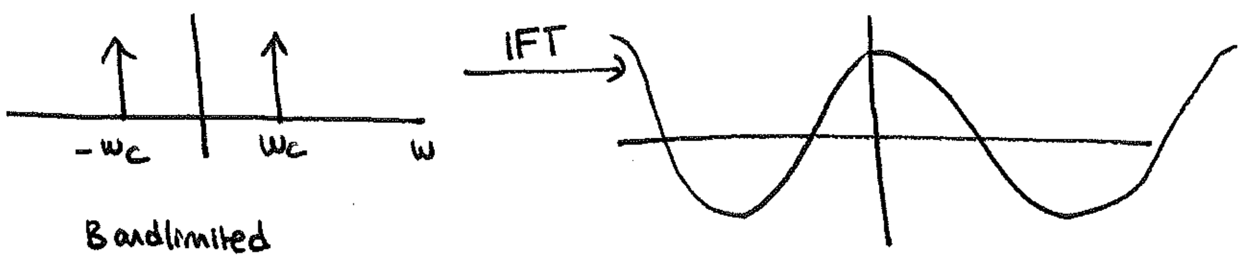
bandlimited signal → not time limited.

For example



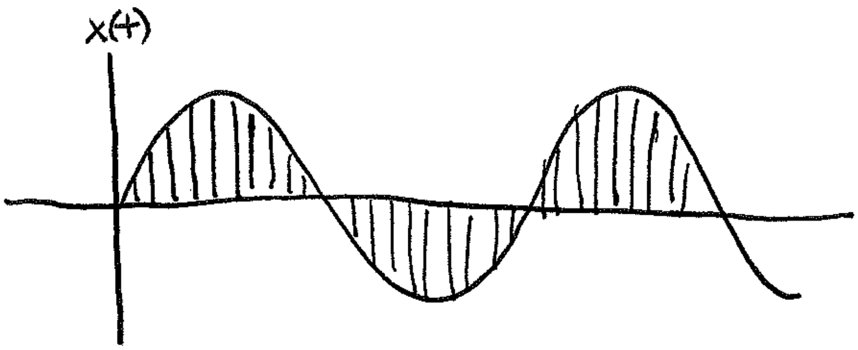
Time limited

Not band-limited

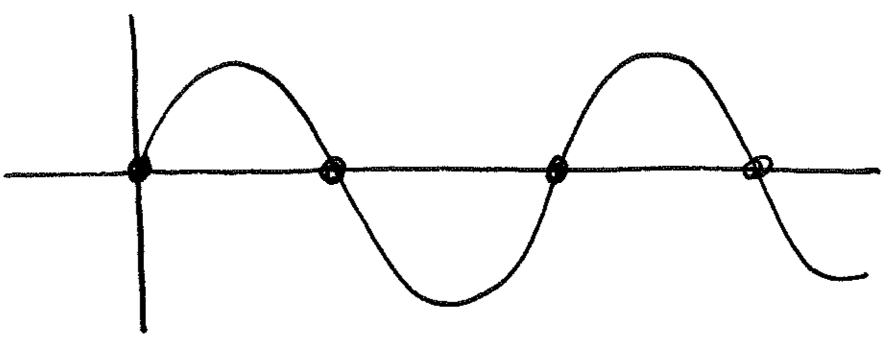


Bandlimited

Not time-limited



← Good example
Samples are close enough



← Bad example
samples are far away

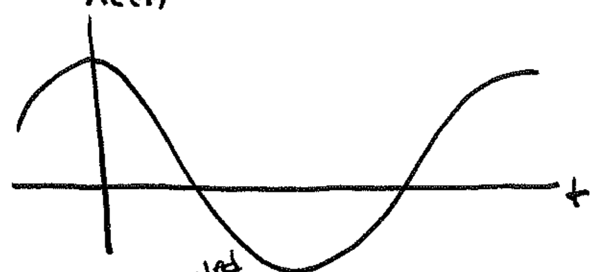
Nyquist - Shannon Sampling Theorem

A bandlimited signal with maximum frequency w_B can be perfectly reconstructed from its evenly spaced samples if the sampling frequency w_s satisfies

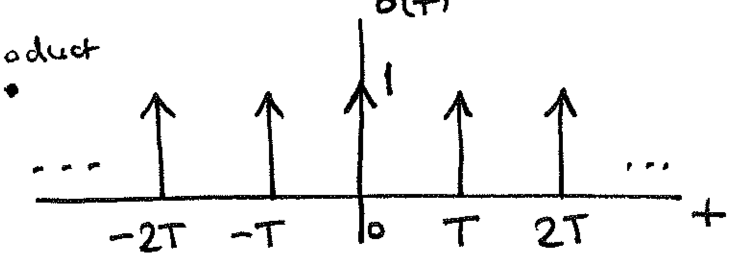
$$\underbrace{w_s > \underbrace{2w_B}_{\text{The Nyquist rate}}}_{\text{Strict inequality}}$$

continuous

$x_c(t)$



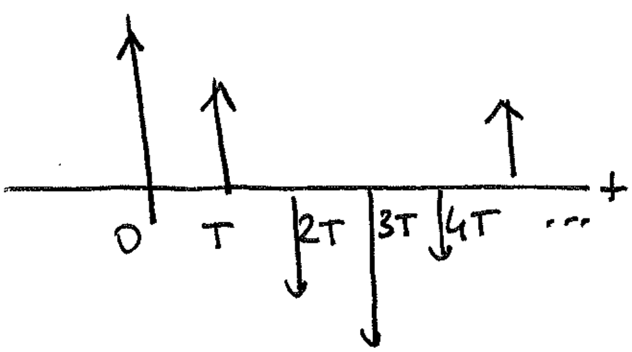
product ·



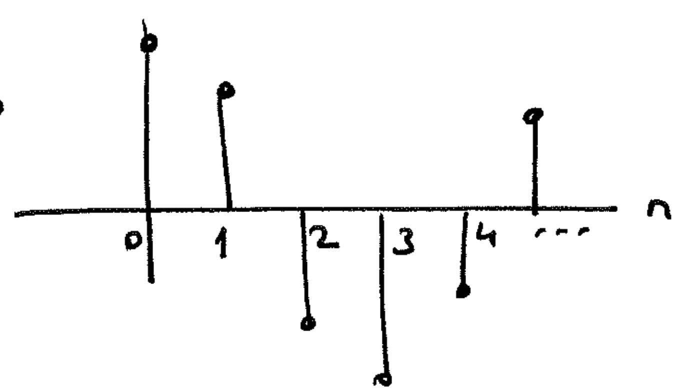
sampled

$x_s(t) = x_c(t) \cdot d(t)$

$x[n]$



$\Longrightarrow$



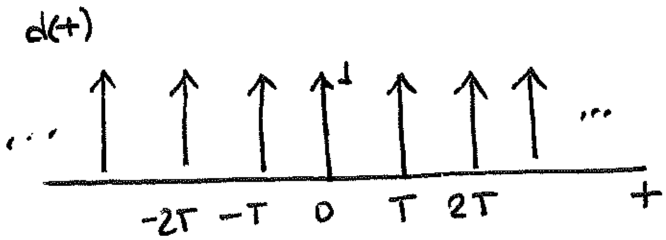
$d(t)$ is periodic $\Longrightarrow$ FS expansion $\quad d(t) = \sum_{k=-\infty}^{+\infty} a_k e^{j\frac{2\pi}{T}kt}$

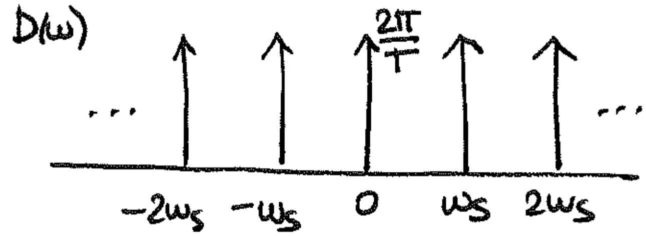
where $\quad a_k = \frac{1}{T}\int_{-T/2}^{T/2} d(t)\, e^{-j\frac{2\pi}{T}kt}\, dt$

$$= \frac{1}{T}$$

$$x_s(t) = x_c(t) \cdot d(t) \xleftrightarrow{FT} \frac{1}{2\pi} X_c(\omega) * D(\omega) \qquad \text{CTFT}$$

$$d(t) = \frac{1}{T}\sum_{k=-\infty}^{+\infty} e^{j\frac{2\pi}{T}kt} \xleftrightarrow{FT} D(\omega) = \frac{2\pi}{T}\sum_{k=-\infty}^{+\infty} \delta\left(\omega - \frac{2\pi}{T}k\right)$$





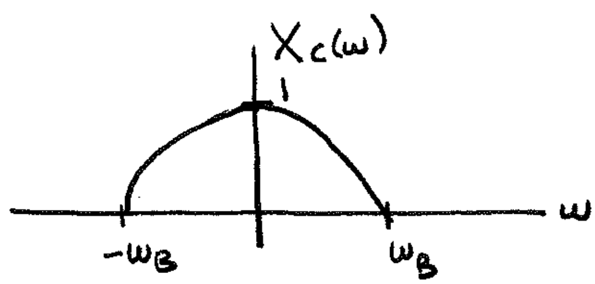
$$\omega_s = \frac{2\pi}{T}$$

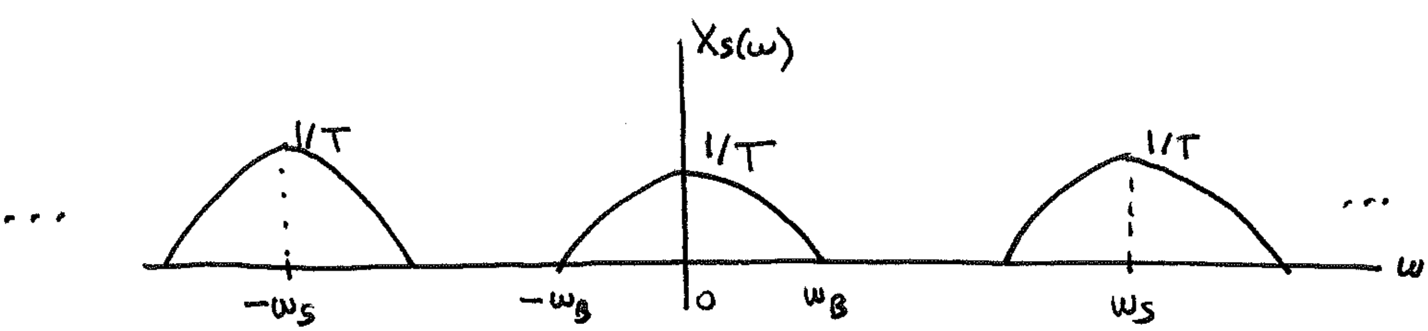
Therefore,

$$X_s(\omega) = \frac{1}{2\pi} X_c(\omega) * D(\omega)$$

$$= \frac{1}{2\pi} X_c(\omega) * \frac{2\pi}{T}\sum_{k=-\infty}^{+\infty} \delta(\omega - \omega_s k)$$

$$= \frac{1}{T}\sum_{k=-\infty}^{+\infty} X_c(\omega - \omega_s k)$$



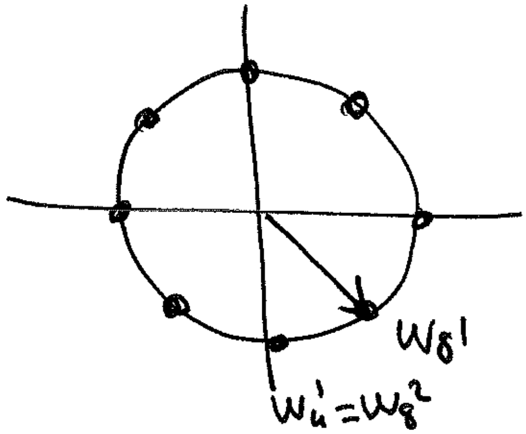


* Need to low pass filter $X_s(\omega)$ to get $X_c(\omega)$ (original signal)

Question: When can we perfectly recover $X_c(\omega)$?

$$y[n] = \underbrace{x[0]}_{\text{even}}\ \overbrace{x[2]}^{\text{odd}}\ \underbrace{x[4]}_{\text{even}}\ \overbrace{x[6]}^{\text{odd}}$$

$$z[n] = \underbrace{x[1]}_{\text{even}}\ \overbrace{x[3]}^{\text{odd}}\ \underbrace{x[5]}_{\text{even}}\ \overbrace{x[7]}^{\text{odd}}$$



FS $\quad a_k = \frac{1}{T}\int_{-T/2}^{T/2} x(t)\, e^{-j\frac{2\pi}{T}kt}\, dt$

$= \frac{1}{T}$