Discrete Fourier Transform (DFT)

Earlier we started with periodic function $f(x)$ of period $2L$ & sent $L \to \infty$. This took us from the Fourier series to the Fourier integral transform.

Now let us retrace our steps. The reason is that we would like to learn how to handle Fourier transforms when we only have a finite no. of measurements of some function.

Start with $f(t)$ with Fourier transform $\tilde{f}(\omega)$

$$\therefore \quad f(t) = \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} e^{-i\omega t} \tilde{f}(\omega)$$

&

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

If $f(t)$ is periodic with period T, then let us prove that its Fourier transform must be "quantized", i.e. ω can only take values that are integer multiples of a fundamental frequency $\Delta\omega = 2\pi/T$.

Periodicity means $f(t+nT) = f(t)$ for integers n.

i.e $0 = f(t) - f(t+nT)$

$$= \int_{-\infty}^{\infty} \frac{d\omega}{2\pi} e^{-i\omega t} (1 - e^{-i\omega nT}) \tilde{f}(\omega)$$

which is only possible if $\tilde{f}(\omega)$ only picks out integer multiples of $2\pi/T$.

i.e $\tilde{f}(\omega) = 2\pi \sum_{m=-\infty}^{\infty} \tilde{f}_m \delta_D(\omega - \frac{2\pi m}{T})$

which is a "modulated" Dirac comb

~~This~~ gives us

$$f(t) = \sum_{m=-\infty}^{\infty} \int_{-\infty}^{\infty} d\omega \tilde{f}_m \delta_D(\omega - \frac{2\pi m}{T}) e^{-i\omega t}$$

i.e $f(t) = \sum_{n=-\infty}^{\infty} \tilde{f}_n e^{2\pi i n t/T}$ $[n = -m]$

$\longrightarrow$ Fourier series

Orthogonality of $e^{2\pi i n t/T}$ on $t \in [0, T)$ then gives

$$\tilde{f}_n = \frac{1}{T} \int_0^T dt\, e^{-2\pi i n t/T} f(t)$$

which completes the description.

Another manifestation of discreteness in frequency space arises due to "sampling"

Consider a general non-periodic function $f(t)$ which is sampled at discrete intervals of length Δt. This can be represented as the product of $f(t)$ with a Dirac comb:

$$f_s(t) \equiv f(t)\, C_D(t)\, \Delta t \qquad \text{[presence of } \Delta t \text{ ensures that } f_s \text{ has the same dimensions as } f\text{]}$$

$$= \sum_{n=-\infty}^{\infty} \Delta t\, f(t)\, \delta_D(t - n\Delta t)$$

i.e $$f_s(t) = \sum_{n=-\infty}^{\infty} \Delta t\, f(n\Delta t)\, \delta_D(t - n\Delta t)$$

which is another modulated Dirac comb

The Fourier transform of the sampled function is called the Discrete-Time Fourier Transform (DTFT) of the original function f.

$$\tilde{f}_s(\omega) = \frac{\Delta t}{2\pi} (\tilde{f} * \tilde{C}_D)(\omega) \qquad \text{[convolution theorem]}$$

$$= \frac{\Delta t}{2\pi} \int_{-\infty}^{\infty} d\omega'\, \tilde{f}(\omega')\, \tilde{C}_D(\omega - \omega')$$

$$= \sum_{n=-\infty}^{\infty} \frac{\Delta t}{2\pi} \cdot \frac{2\pi}{\Delta t} \int_{-\infty}^{\infty} d\omega'\, \tilde{f}(\omega')\, \delta_D(\omega - \omega' - 2\pi n/\Delta t)$$

from F.T. defn.

i.e $$\tilde{f}_s(\omega) = \sum_{n=-\infty}^{\infty} \tilde{f}\left(\omega - \frac{2\pi n}{\Delta t}\right) = \sum_{n=-\infty}^{\infty} \Delta t\, f(n\Delta t)\, e^{i n \omega \Delta t}$$

So the DTFT of $f(t)$ at frequency ω gets contributions not just from ω, but all frequencies at integer multiples of $\Delta\omega = \frac{2\pi}{\Delta t}$ away from ω.

These ideas come together nicely when we sample a periodic function f, provided the sampling is done with the appropriate rate.

Suppose $f(t)$ is periodic with period T. Let us choose a Dirac comb whose interval is

$\Delta t = T/N$ where $N > 0$ is an integer.

i.e. $G(t)$ is periodic with period Δt & $f(t)$ is periodic with period $N\Delta t$

This means that the sampled function f_s is also periodic with period $N\Delta t = T$

[Would this still be the case if N is not an integer?]

This means $f_s(t)$ can be expanded in a Fourier series

$$f_s(t) \equiv \Delta t\, f(t)\, e_D(t)$$

$$= \sum_{m=-\infty}^{\infty} \tilde{f}_m\, e^{2\pi i m t/T}$$

~~& we know that its Fourier transform is~~

$\cancel{\tilde{f}_s(\omega) = 2\pi}$

Fourier transforming, we find

$$\tilde{f}_s(\omega) = \int_{-\infty}^{\infty} dt\, e^{i\omega t}\, f_s(t)$$

$$= \sum_{m=-\infty}^{\infty} \tilde{f}_m \int_{-\infty}^{\infty} dt\, e^{i(\omega - \frac{2\pi m}{T})t}$$

$$= 2\pi \sum_{m=-\infty}^{\infty} \tilde{f}_m\, \delta_D\left(\omega - \frac{2\pi m}{T}\right)$$

as expected from periodicity of f_s.

Now we exploit the sampling to show that
(a) Each $\tilde{f}_m$ only gets contributions from N samples of f in a single fundamental period and
(b) $\tilde{f}_m$ is periodic with period N.

$$\tilde{f}_m = \frac{1}{T}\int_0^T dt\, e^{-2\pi i m t/T} f_s(t)$$

$$= \frac{1}{T}\int_0^T dt\, e^{-2\pi i m t/T} \cdot \Delta t \sum_{n=-\infty}^{\infty} f(n\Delta t)\, \delta_D(t-n\Delta t)$$

$$= \underbrace{\frac{\Delta t}{T}}_{\frac{1}{N}} \sum_{n=-\infty}^{\infty} f(n\Delta t) \underbrace{\int_0^T dt\, e^{-2\pi i m t/T}\, \delta_D(t-n\Delta t)}$$

will only pick those n values for which $0 \le n\Delta t < T$

$$\therefore \boxed{\tilde{f}_m = \frac{1}{N}\sum_{n=0}^{N-1} f(n\Delta t)\, e^{-2\pi i m n/N}}$$

– This proves (a) & gives definition of the Discrete Fourier Transform (DFT)

We can clearly see that

$$\tilde{f}_{m+jN} = \frac{1}{N}\sum_{n=0}^{N-1} f(n\Delta t)\, e^{-2\pi i m n/N}\, \underbrace{e^{-2\pi i j n}}_{=1} = \tilde{f}_m$$

which proves (b)

Since $f(t)$ is periodic, $\tilde{f}_m$ can be written as a sum over N ~~any~~ samples from any single period, not necessarily $0 \le n < N$

$$\left[f((n+jN)\Delta t)\, e^{-\frac{2\pi i m}{N}(n+jN)} = f(n\Delta t + jT)\, e^{-2\pi i m n/N} = f(n\Delta t)\, e^{-2\pi i m n/N} \right]$$

$0 \le n < N$ & integer j

To invert the DFT, we can use the orthogonality (or completeness) of the $e^{2\pi i mn/N}$ in $0 \leq m < N-1$

Write $\boxed{f_n \equiv \cancel{f(n\Delta t)} \sum_{m=0}^{N-1} \tilde{f}_m e^{+2\pi i mn/N}}$

→ We want to check whether $f_n = f(n\Delta t)$.

then

$$f_n = \sum_{m=0}^{N-1} \sum_{n'=0}^{N-1} \frac{1}{N} e^{2\pi i m(n-n')/N} f(n'\Delta t)$$

$$= \frac{1}{N} \sum_{n'=0}^{N-1} \left[\sum_{m=0}^{N-1} e^{2\pi i \frac{(n-n')}{N} m} \right] f(n'\Delta t)$$

$$= \frac{1}{N} \sum_{n'=0}^{N-1} \sum_{m=0}^{N-1} (\gamma_{n,n'})^m \, f(n'\Delta t), \quad \gamma_{n,n'} \equiv e^{2\pi i (n-n')/N}$$

Now

$$\sum_{m=0}^{N-1} (\gamma_{n,n'})^m = \begin{cases} N, & \gamma_{n,n'} = 1 \\ \dfrac{1-(\gamma_{n,n'})^N}{1-\gamma_{n,n'}}, & \gamma_{n,n'} \neq 1 \end{cases}$$

$$= \begin{cases} N, & n = n' \\ 0, & n \neq n' \end{cases}$$

$$= N \delta_{nn'}$$

$$\therefore f_n = \sum_{n'=0}^{N-1} \delta_{nn'} f(n'\Delta t) = f(n\Delta t) \checkmark$$

[since $0 \leq n < N-1$]

As with Fourier series & transforms, we can adjust the overall scaling of the DFT pairs ~~to ens~~ provided the inversion is unaffected.

Our derivation gives us the pair

$$\tilde{f}_m = \frac{1}{N}\sum_{n=0}^{N-1} f_n e^{-2\pi i m n/N}$$

$$\& \quad f_n = \sum_{m=0}^{N-1} \tilde{f}_m e^{2\pi i m n/N}$$

, $f_n \equiv f(n\Delta t)$ where $f(t)$ is periodic with period $N\Delta t$.

Both $\tilde{f}_m$ & f_n are periodic with period N

DFT is therefore very useful in manipulating & analysing periodic functions. We will discuss non-periodic functions later; the DFT is still very useful, but one must take care to choose the sampling interval properly.

Some nomenclature:

Δt = sampling interval = (Nyquist rate)$^{-1}$

$\Delta\omega \equiv \frac{2\pi}{T} = \frac{2\pi}{N\Delta t}$ = Nyquist frequency sampling interval

$\omega_c \equiv \pi/\Delta t = N\frac{\Delta\omega}{2}$ = Nyquist critical frequency

$\nu_c \equiv \frac{1}{2\Delta t}$ = $\frac{1}{2}\times$ Nyquist rate

Frequencies are sampled with interval $\Delta\omega$ ~~rate~~ ~~$2\pi/\Delta t$~~ in the range $[0, 2\omega_c)$ or $[-\omega_c, \omega_c)$

Loosely, T sets smallest frequency while Δt sets the largest frequency.

Fast Fourier Transform (FFT)

For convenience let's rescale the DFT pair to write

$$\tilde{f}_m = \sum_{n=0}^{N-1} f_n e^{-2\pi i mn/N} \; ; \quad f_n = \frac{1}{N}\sum_{m=0}^{N-1} \tilde{f}_m e^{2\pi i mn/N}$$

Define $W_{(N)} \equiv e^{-2\pi i/N}$

Then $\tilde{f}_m = \sum_{n=0}^{N-1} W_{(N)}^{mn} f_n$ is a matrix operation with the $N \times N$ matrix $W_{(N)}^{mn}$ acting on the $N \times 1$ vector f_n.

Computationally this is an $O(N^2)$ operation [N^2 complex multiplications & $N(N-1)$ additions + generating the numbers W^{mn}.]

FFT is an algorithm that enables the DFT to be computed in $O(N \log_2 N)$ time, leading to tremendous gains for large matrices. E.g.: Consider the case $N \sim 10^6$ ~~with~~ where the difference is between ~ 30s & ~ 2 weeks.

Algorithm is due to Danielson & Lanczos (1942), generalised by Cooley & Tukey (1965). We will study the Danielson-Lanczos version ~~to~~ for the case that N is a power of 2.

First note that the DFT of length N can be written as the sum of 2 DFT's of length N/2 whenever N is even:

$$\tilde{f}_m = \sum_{n=0}^{N-1} f_n W_{(N)}^{mn}$$

$$= \sum_{j=0}^{N/2-1} W_{(N)}^{m\cdot(2j)} f_{2j} + \sum_{j=0}^{N/2-1} W_{(N)}^{m\cdot(2j+1)} f_{2j+1}$$

But $W_{(N)}^{m\cdot(2j)} = e^{-2\pi i\cdot m\cdot 2j/N}$

$$= e^{-2\pi i\cdot m\cdot j/(N/2)}$$

$$= W_{(N/2)}^{m\cdot j}$$

Similarly $W_{(N)}^{m\cdot(2j+1)} = W_{(N)}^{m}\cdot W_{(N/2)}^{m\cdot j}$

$$\therefore \underbrace{\tilde{f}_m}_{1\times(N\times N)\text{ operation}} = \sum_{j=0}^{N/2-1} W_{(N/2)}^{mj} f_{2j} + W_{(N)}^{m} \sum_{j=0}^{N/2-1} W_{(N/2)}^{mj} f_{2j+1}$$

$$\equiv \underbrace{\tilde{f}_m^{(e)}}_{\text{even}} + W_{(N)}^{m} \underbrace{\tilde{f}_m^{(o)}}_{\text{odd}}$$

$$= 2\times(N/2\times N/2) \text{ operations}$$

$\Rightarrow$ factor 2 gain in computation time

~~Recall that~~

This is possible because $W_{(N/2)} = W_{(N)}^2$ and because the matrix is $(W)^{m*n}$, making the matrix of the DFT highly redundant, with many repeated values.

Notice, however, that $\tilde{f}_m^{(e)}$ & $\tilde{f}_m^{(o)}$ are DFT's of length $N/2$, i.e $m = 0, 1, \ldots, N/2 - 1$.

So how will we get $\tilde{f}_m$ for $m = N/2, N/2+1, \ldots, N-1$

→ Recall $\tilde{f}_{m+jN} = \tilde{f}_m$ for any DFT of length N for any integer j

$$\Rightarrow \tilde{f}^{(e/o)}_{m+jN/2} = \tilde{f}^{(e/o)}_m \quad \text{for any integer } j$$

So we can simply write

$$\tilde{f}_m = \begin{cases} \tilde{f}_m^{(e)} + W_{(N)}^m \tilde{f}_m^{(o)}, & 0 \leq m \leq N/2 - 1 \\ \tilde{f}^{(e)}_{m-N/2} + W_{(N)}^m \tilde{f}^{(o)}_{m-N/2}, & N/2 \leq m \leq N-1 \end{cases}$$

giving us the DFT over the full range

Now use this method recursively, i.e, apply the same trick to $\tilde{f}_m^{(e)}$ & $\tilde{f}_m^{(o)}$ as DFT's of length $N/2$ each, getting a sum of 4 terms

$$\tilde{f}_m = (\ )\tilde{f}_m^{(ee)} + (\ )\tilde{f}_m^{(eo)} + (\ )\tilde{f}_m^{(oe)} + (\ )\tilde{f}_m^{(oo)}$$

giving us a $4 \times (N/4)^2$ operation

Obviously this is only valid if $N/2$ is even, and so we would clearly like $N = 2^M$ in order to push further.

With N a power of 2, this trick can be pushed until we have written $\tilde{f}_m$ as a sum of DFT's of length 1. But DFT of length 1 is just the identity : $\tilde{f}_0 = f_0$

With $N = 2^M$ this gives

$$\tilde{f}_m = (\,)\,\tilde{f}_m^{(\overbrace{ee--e}^{M\text{ values}})} \underbrace{+ \cdots +}_{2^M = N \text{ terms (one for each combin- of e's \& o's)}} (\,)\,\tilde{f}_m^{(\overbrace{oo--o}^{M\text{ values}})}$$

with $\tilde{f}_m^{(eeoe--eo)} = f_m^{(eeoe--eo)}$ [DFT of length 1]

$= f_n$ for some $0 \leq n \leq N-1$
[because we only have N samples]

We will be through if we know which pattern of e's & o's corresponds to which n.

Note: Since all ~~N perm~~ $= 2^M$ arrangements of M e's & o's contribute to each $\tilde{f}_m$, it shouldn't be surprising that $\tilde{f}_m^{(\underbrace{ee--oe}_{M\text{ values}})}$ is independent of m.

The m-dependence only appears in the coefficients through powers of $W_{(N)}$.

Claim [w/o proof]: If we set 'e' = 0 & 'o' = 1 & think of the sequence of M 0's & 1's as the binary representation of $N = 2^M$ integers, then the <u>bit reversed</u> sequence of e's & o's gives the binary representation of n.

~~i.e n = bit reversed~~

i.e $f_n = f^{(eeeo---eo)}$ if

$$n = \text{bit reversal of } (eeeo---eo)$$
$$(0001---01)$$

$$[= (10---1000)]$$

Example below.

<u>Summary</u>:

- Take original vector f_n & arrange in bit-reversed order. These are 1-point transforms
- Combine these adjacent pairwise (with easily predictable coefficients) to get 2-point transforms
- Continue M times to get $N = 2^M$-point transform. [Final step will be combination of single pair consisting of ½ the data set in each part.]
- There are $M = \log_2 N$ combinations (or steps) and each step involves $O(N)$ operations. So overall complexity is $O(N \log_2 N)$ [Assuming bit reversal is not more complex than this]

Example:

Let $N=4$.

DFT is $\quad \tilde{f}_m = \sum_{n=0}^{3} W_{(4)}^{mn} f_n \quad , \quad W_{(4)} = e^{-2\pi i/4} = -i$

i.e.
$$\begin{pmatrix} \tilde{f}_0 \\ \tilde{f}_1 \\ \tilde{f}_2 \\ \tilde{f}_3 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -i & -1 & i \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \end{pmatrix} \begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{pmatrix}$$

$$\therefore \quad \begin{aligned} \tilde{f}_0 &= 1\cdot f_0 + 1\cdot f_1 + 1\cdot f_2 + 1\cdot f_3 \\ \tilde{f}_1 &= 1\cdot f_0 - i\cdot f_1 - 1\cdot f_2 + i\cdot f_3 \\ \tilde{f}_2 &= 1\cdot f_0 - 1\cdot f_1 + 1\cdot f_2 - 1\cdot f_3 \\ \tilde{f}_3 &= 1\cdot f_0 + i\cdot f_1 - 1\cdot f_2 - i\cdot f_3 \end{aligned}$$

$\longrightarrow$ $4\times 4 = 16$ multiplic^ns & $3\times 4 = 12$ additions

~ 28 operations

FFT :- First bit reverse $\begin{pmatrix} f_0 \\ f_1 \\ f_2 \\ f_3 \end{pmatrix} \sim \begin{pmatrix} 00 \\ 01 \\ 10 \\ 11 \end{pmatrix} \rightarrow \begin{pmatrix} 00 \\ 10 \\ 01 \\ 11 \end{pmatrix} \sim \begin{pmatrix} f_0 \\ f_2 \\ f_1 \\ f_3 \end{pmatrix}$

- Now use pairs (f_0, f_2) & (f_1, f_3) to form the 2-point DFT's.

$f_0 + f_2 \,,\, f_0 - f_2$

and also $f_1 + f_3 \,,\, f_1 - f_3$ ~~by periodicity~~

[For $N=2$ we would have $W_{(2)} = e^{-2\pi i/2} = -1$

So $\tilde{g}_0 = g_0 + g_1$ & $\tilde{g}_1 = g_0 - g_1$]

$\longrightarrow$ This involves 4 additions

~~- Now form the final four combinations~~

$\longrightarrow$

... in the signal".

Now form the final four combinations [recall $W_{(4)} = -i$ & $\tilde{f}_m = \tilde{f}^{(e)}_m + W^m_{(N)} \tilde{f}^{(o)}$ at last step]

$\cancel{\tilde{f}_{n}=}$

$$\tilde{f}_0 = \underbrace{(\overset{(ee)}{f_0} + \overset{(eo)}{f_2})}_{\tilde{f}^{(e)}_0} + \underbrace{1}_{W^{0}_{(4)}} \cdot \underbrace{(\overset{(oe)}{f_1} + \overset{(oo)}{f_3})}_{\tilde{f}^{(o)}_0}$$

$$\tilde{f}_1 = \underbrace{(f_0 - f_2)}_{\tilde{f}^{(e)}_1} - \underbrace{i}_{W^1_{(4)}} \cdot \underbrace{(f_1 - f_3)}_{\tilde{f}^{(o)}_1}$$

and, by periodicity

$$\tilde{f}_2 = (f_0 + f_2) - \underbrace{1}_{W^2_{(4)}} \cdot (f_1 + f_3)$$

$$\tilde{f}_3 = (f_0 - f_2) + \underbrace{i}_{W^3_{(4)}} \cdot (f_1 - f_3)$$

Same as DFT

$\longrightarrow$ total 4 multiplications & 4 additions ~ 8 operations

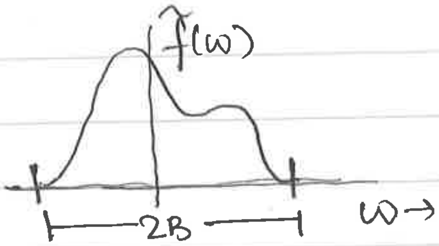
$\Longrightarrow$ FFT used ~ 12 operations apart from bit reversal

Sampling theorem & Aliasing

Consider a signal $f(t)$ which is bandwidth limited.

i.e: $\tilde{f}(\omega)$ only gets contribution from a finite range $\omega \in [-B, B)$ of length $2B$.

B is called the bandwidth.



[Note that sometimes $B/2\pi$ is called the Nyquist frequency & ω_c/π simply the sampling rate.]

Suppose we are able to sample $f(t)$ discretely with sampling interval Δt.

Question: Is this discrete sampling sufficient to reconstruct the entire signal? [i.e, without spoiling it in frequency domain]

Sampling theorem says "yes", iff the sampling rate (equivalently the Nyquist critical frequency) is large enough.

Specifically, the sequence $\{f(n\Delta t)\}$ completely characterises $f(t)$ if

$$\omega_c = \pi/\Delta t > B$$

i.e Sampling rate $\frac{1}{\Delta t} > 2 \cdot \left(\frac{B}{2\pi}\right) = 2 \times$ largest freq. in the signal.

Let us see why.

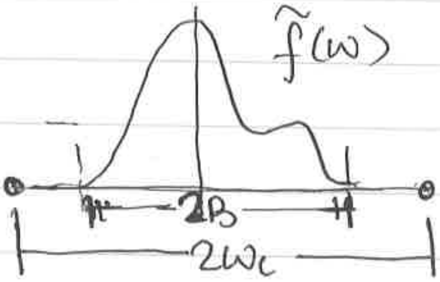
Recall that the sampled function can be written as a modulated Dirac comb:

$$f_s(t) = \sum_n \Delta t\, f(n\Delta t)\, \delta_D(t - n\Delta t)$$

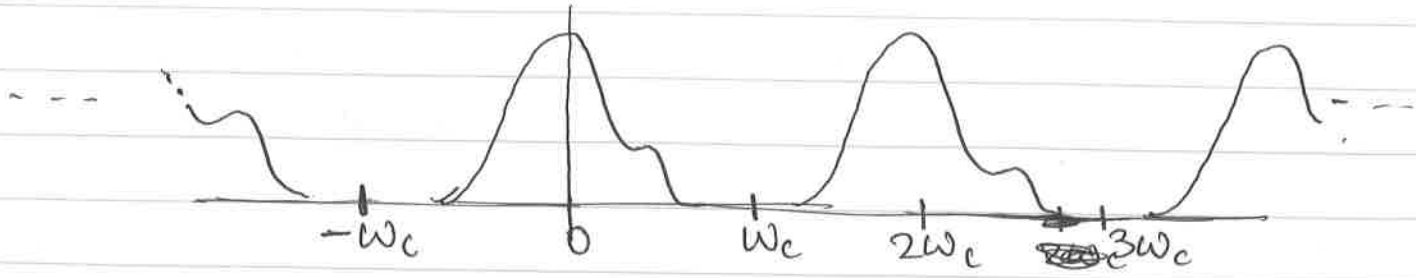
with Fourier transform

$$\tilde{f}_s(\omega) = \sum_n \tilde{f}(\omega - 2n\omega_c)$$

Suppose $\omega_c > B$



Then clearly $\hat{f}_s(\omega)$ is a sequence of non-overlapping copies of $\tilde{f}(\omega)$



In this case we can isolate one copy of $\tilde{f}(\omega)$ by multiplying $\tilde{f}_s(\omega)$ with a TopHat filter

$$\tilde{f}_{is}(\omega) \equiv \tilde{f}_s(\omega)\,\hat{W}(\omega) \quad \text{where} \quad \tilde{W}(\omega) = \Theta(\omega_c - |\omega|)$$

$\downarrow$
"isolated"

$\rightarrow$ picks out the copy in $\omega \in [-\omega_c, \omega_c)$ which is nothing but the original $\tilde{f}(\omega)$

i.e $\tilde{f}_{is}(\omega) = \tilde{f}(\omega)$

From the convolution theorem we have

$$f_{is}(t) = (f_s * W)(t)$$

where $W(t) = \frac{\omega_c}{\pi}\,\mathrm{sinc}(\omega_c t)$

$$\therefore \; f_{is}(t) = \int_{-\infty}^{\infty} dt'\, f_s(t')\, W(t-t')$$

$$= \sum_{n=-\infty}^{\infty} \Delta t\, f(n\Delta t) \int_{-\infty}^{\infty} dt'\, \delta_D(t' - n\Delta t)\, W(t-t')$$

$$= \sum_{n=-\infty}^{\infty} \Delta t \cdot \frac{\omega_c}{\pi}\, f(n\Delta t)\, \mathrm{sinc}(\omega_c t - n\omega_c \Delta t)$$

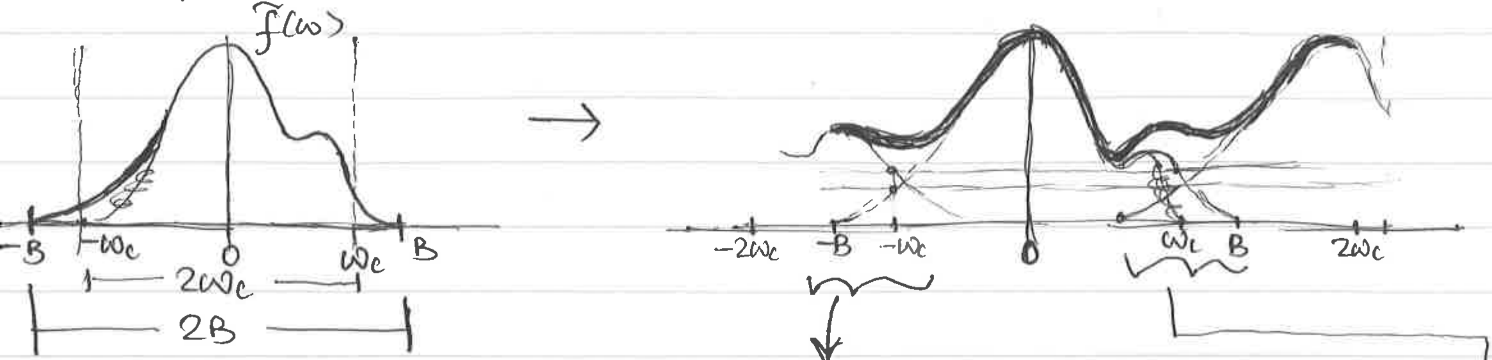
But $\omega_c = \pi/\Delta t$

$$\therefore \; f_{is}(t) = \sum_{n=-\infty}^{\infty} f(n\Delta t)\, \mathrm{sinc}\left(\pi(t/\Delta t - n)\right)$$

Since $\mathrm{sinc}(\pi(m-n)) = \delta_{m,n}$, this is an interpolation of the sequence $\{f(n\Delta t)\}$, with Fourier transform identical to $\tilde{f}(\omega)$.

[Proves one direction in the theorem]

Now suppose $\omega_c < B$ & we sample the function. Then $\tilde{f}_s(\omega)$ [shift & add] starts mixing copies of $\tilde{f}(\omega)$



Frequencies $\omega_c < \omega < B$ contribute to $\omega \gtrsim -\omega_c$.
Similarly frequencies $-B < \omega < -\omega_c$ contribute to $\omega \lesssim \omega_c$

This is called aliasing [one frequency mimicking another]

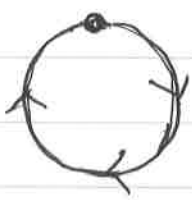
Clearly we can no longer isolate any original copy of $\tilde{f}(\omega)$, meaning that the sequence $\{f(n\Delta t)\}$ cannot correctly describe the full function. [The function is "undersampled".]
[Proves converse direction of theorem.]

Possible techniques to avoid aliasing:
- Ideally, oversampling [i.e. having a large enough ω_c or small enough Δt]
  → There may be hardware restrictions on this
- Anti-aliasing or low-pass filtering
  → "Colour" the signal by restricting its frequency range before sampling. [Think optical filters.]

Aliasing shows up in many places
- Digital signal processing
  [Audio / Single images / Movies]
- Radio Astronomy
- Simulating galaxy clustering

Simple example to understand aliasing

Consider a ~~bright~~ reflective dot (say, a small mirror), going in a circle clockwise with period $T = 1s$

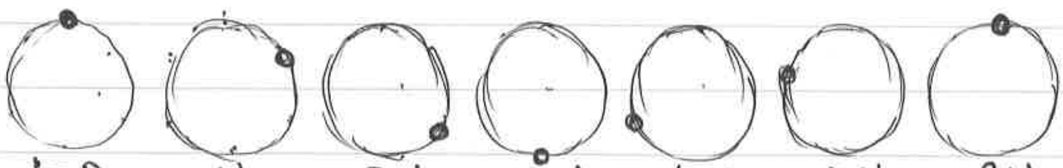


$T = 1s$

We see this dot using a strobe light (or a flashing torch or whatever) ~~with~~ which samples with interval Δt.

- If $\Delta t \ll T$, then we have many samples in a single period

Eg: $\Delta t = 1/6$ sec

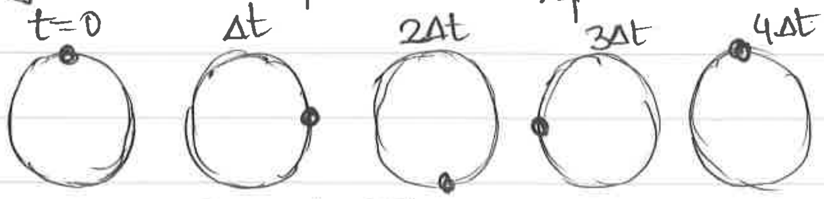


$t=0$ $\quad \Delta t$ $\quad 2\Delta t$ $\quad 3\Delta t$ $\quad 4\Delta t$ $\quad 5\Delta t$ $\quad 6\Delta t$

So $T_{obs} = 6\Delta t = 1s = T$ ✓

- When Δt is close to T, things become interesting

- E.g. $\Delta t = 5/4$ s [$> T$]



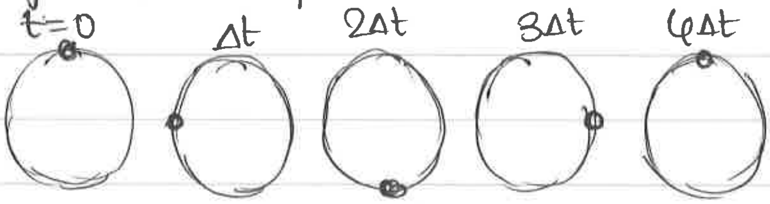
[see why?]

$\therefore T_{obs} = 4\Delta t = 5$ s $> T$

[Notice $\omega_{obs} \equiv 2\pi/T_{obs} = \omega_{true} - 2\omega_c$]

($\omega_c = \pi/\Delta t$)

- And if $\Delta t = 3/4$ s [$< T$].



[see why?]

→ Going backwards! [Seen this effect anywhere else?]

Now $T_{obs} = -4\Delta t = -3$ s

[And again: $\omega_{obs} = \omega_{true} - 2\omega_c$]

↳ Consistent with being the $n = -1$ term in the series $\tilde{f}_s(\omega) = \sum_n \tilde{f}(\omega - 2n\omega_c)$

[Think of $\tilde{f}(\omega)$ as a narrow Gaussian centered at $\omega = \omega_{true}$]

These examples also show that we need $\Delta t \leq T/2$ in order to get $T_{obs} = T$
→ i.e. $\omega_c \geq 2\pi/T$
— Sampling theorem.