DIGITAL SIGNAL PROCESSING
AELE4507
AELE7104

Signal Representation

- Objective is to get a good approximation to a set of “real-world” signals, by linear superposition of weighted basis vectors, ϕ_k.
- For the data-set (typical signals) we want to do this using either:
  - the minimum number of vectors,
  - or the vectors that best discriminate signals
  - or both.

$$f(x) = \sum_{COMPLETE} w_k \phi_k(x) = \vec{w}.\vec{\phi}$$

- Most times we know ϕ_k but sometimes we “train” these from a typical data set.

Fourier series

- We endeavour to make our job easier by representing arbitrary signals by a weighted set of “basis” vectors, so when they are passed through a system such as a filter, we can describe the output in terms of how we know the basis vectors to be altered individually.
- For example, consider a whole lot of cosines basis vectors of increasing frequency passing through a filter. Only those below a certain frequency are passed, so only the contributions of these in the original signal will construct the output signal.
- We also try and choose the basis vectors so they are “orthogonal” and therefore don’t couple together going through a linear system, and we try and choose the “complete” set so that we minimise the number needed to completely represent an arbitrary signal. (Clearly this depends on the type of signals we are dealing with.)

Fourier Series (contd)

- The Fourier series is one such set of basis vectors. Each vector or function is a sinusoidal signal. e.g. cos(wt), cos(2wt), etc.
- Thus we can represent any periodic signal (having period T) by a series

$$f(t) = a_o + \sum_{k=1}^{\infty} a_k \cos\left(\frac{2\pi}{T} kt\right) + \sum_{n=1}^{\infty} b_n \sin\left(\frac{2\pi}{T} nt\right)$$

- Alternatively we can realise that exp(a)=cos(a)+jsin(a), and write this in a simpler form;

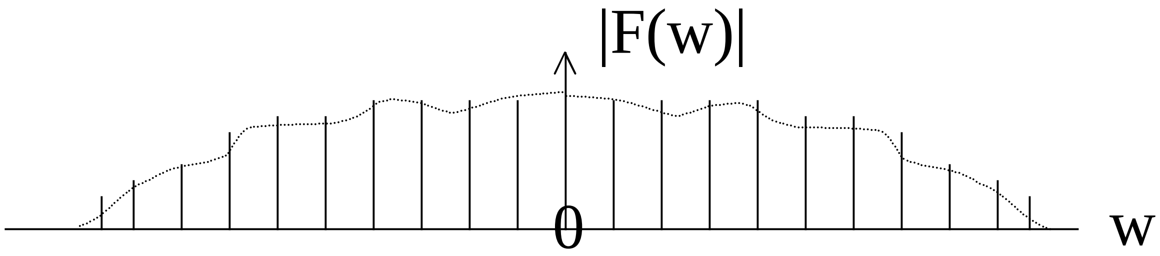
$$f(t) = \sum_{k=-\infty}^{\infty} F_k \exp\left(j\frac{2\pi}{T} kt\right)$$

- where

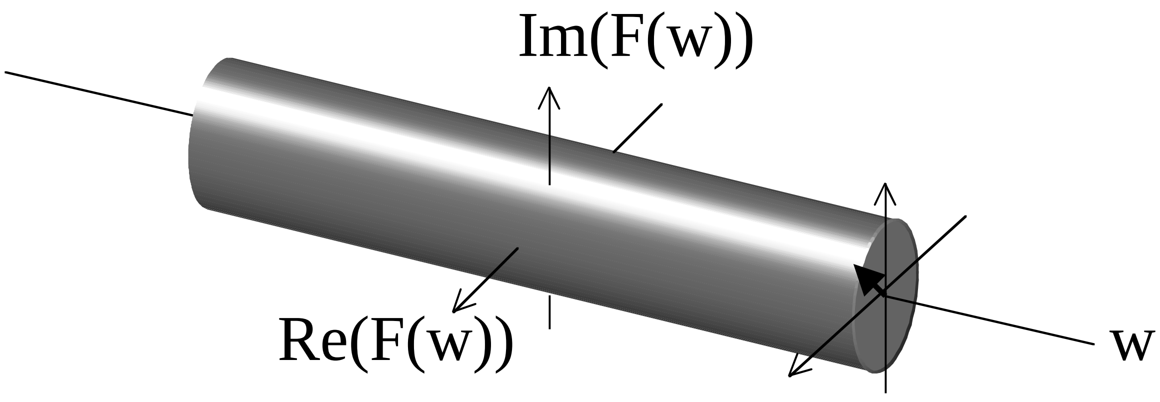
$$F_k = \frac{1}{T} \int_{-T/2}^{T/2} f(t) \exp\left(-j\frac{2\pi}{T} kt\right) dt$$

The frequency domain

- The frequency spectrum of a periodic waveform therefore consists of harmonics of (complex) height F_k.

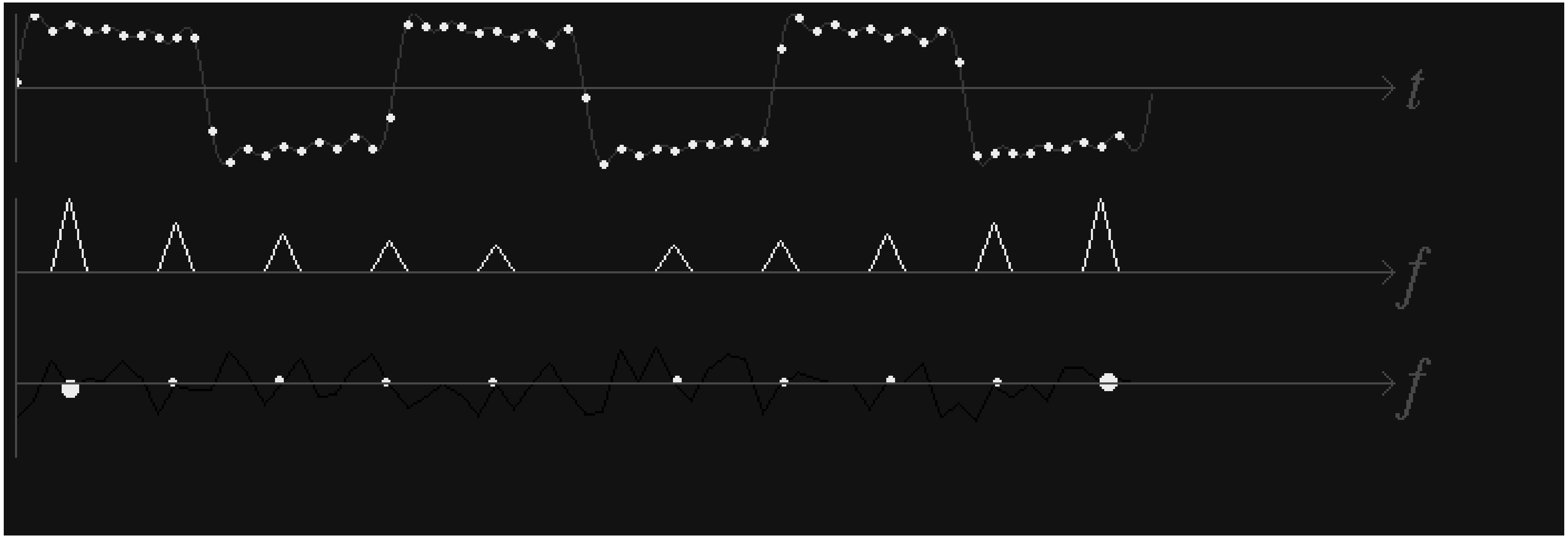


- The frequency domain of a one-dimensional time signal is one-dimensional in frequency, but has complex values.



The importance of phase

- Consider the phase of a single sinusoid frequency: [MOVIE]



- Phase is important for distortionless transmission: [MOVIE]

What is a transform?

- A transform is a (possibly reversible) rearrangement of information in a signal

$$H(u) = \int_{RANGE} h(x)k(x,u)dx$$

where *k* is the kernel, or basis set.

$$H_d(v) = \sum_{RANGE} h(i)k(i,v)$$

where *v* might be discrete or continuous.

Why transform?

- Why do we bother with transforms?
  - To rearrange the signal for more easy/obvious analysis
  - To differentiate better between two similar signals
  - To ease understanding of the filter process
  - To implement faster operations

Fourier Transform

- What about a signal that is not periodic? We can still determine the frequency content, this time using the Fourier transform. (w=2πf)

$$F(\omega) = \int_{-\infty}^{\infty} f(t) \exp(-j\omega t) dt$$

- And we can get back from the frequency domain to the time domain using the inverse Fourier transform.

$$f(t) = \int_{-\infty}^{\infty} F(\omega) \exp(j\omega t) d\omega$$

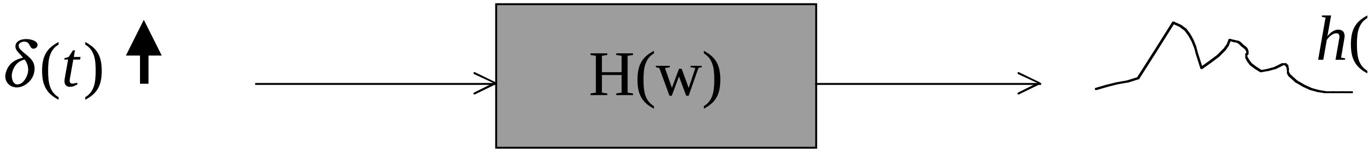
- Note the Fourier transform works for periodic signals too!F_k=F(k2π/T)

Duality of domains

- Why do we need to consider the frequency domain? Isn't it an abstraction?
  - It may be easier to describe a filter in terms of frequency operation.
  - If we have a way to "go" to the frequency domain then operations may be done faster there, than by operating purely in the time domain.
  - Operations performed in one domain have an immediate effect in the other.
  - An operation may be performed in either domain - the result in the other domain will be effected simultaneously. For example, the multiplication of two signals in the time domain results in the convolution of their respective frequency contents, whereas, a multiplication of their frequency contents results in a convolution of the signals in the time domain. (This latter case is "filtering"!).

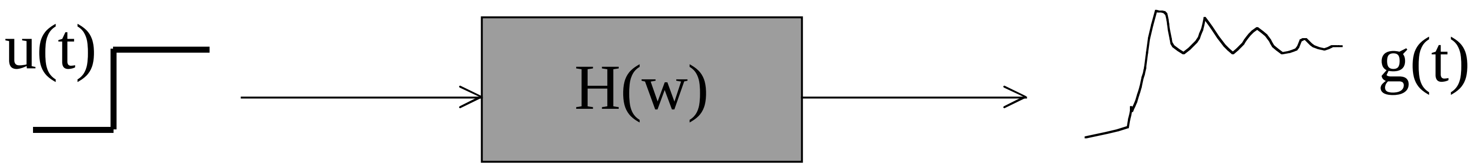
Impulse response

- If we know how a filter alters each frequency of a signal, this is known as the “transfer function” of the filter, H(w). e.g. a low pass filter with cutoff frequency of 1 kHz.
- The inverse Fourier transform of this transfer function is known as the “impulse response” of the filter. This is what emerges from the filter in the time domain if we put an impulse into it.



$$h(t) = \frac{dg(t)}{dt}$$

- It is also the derivative of the step response of the filter



Impulse function

- What is the impulse function? It is the thinnest, tallest, function having area of unity!
- We can construct it by considering any function with unit area being squashed up until it is infinitely thin…

$$r(t) = \begin{cases} 0 & t < -d/2 \\ 1/d & -d/2 \le t \le d/2 \\ 0 & t > d/2 \end{cases}$$

$$\delta(t) = \lim_{d \to 0} r(t)$$

- [MOVIE]

Convolution

- So any signal passed through the filter will have its frequency content multiplied by the transfer function of the filter, or alternatively it will be convolved with the impulse response of the filter. THESE OPERATIONS ARE ENTIRELY EQUIVALENT!

- So what is “convolution”?

$$f(t) \otimes g(t) = \int_{-\infty}^{\infty} f(T)g(t-T)dT$$

- This amounts to constructing the output by reversing one of the signals, shifting it a fraction, multiplying it by the other signal, to find the overlapping area which is the output value at that shift, then shifting a fraction more, and so on….

Mathematically…

- Consider two rectangular pulses..

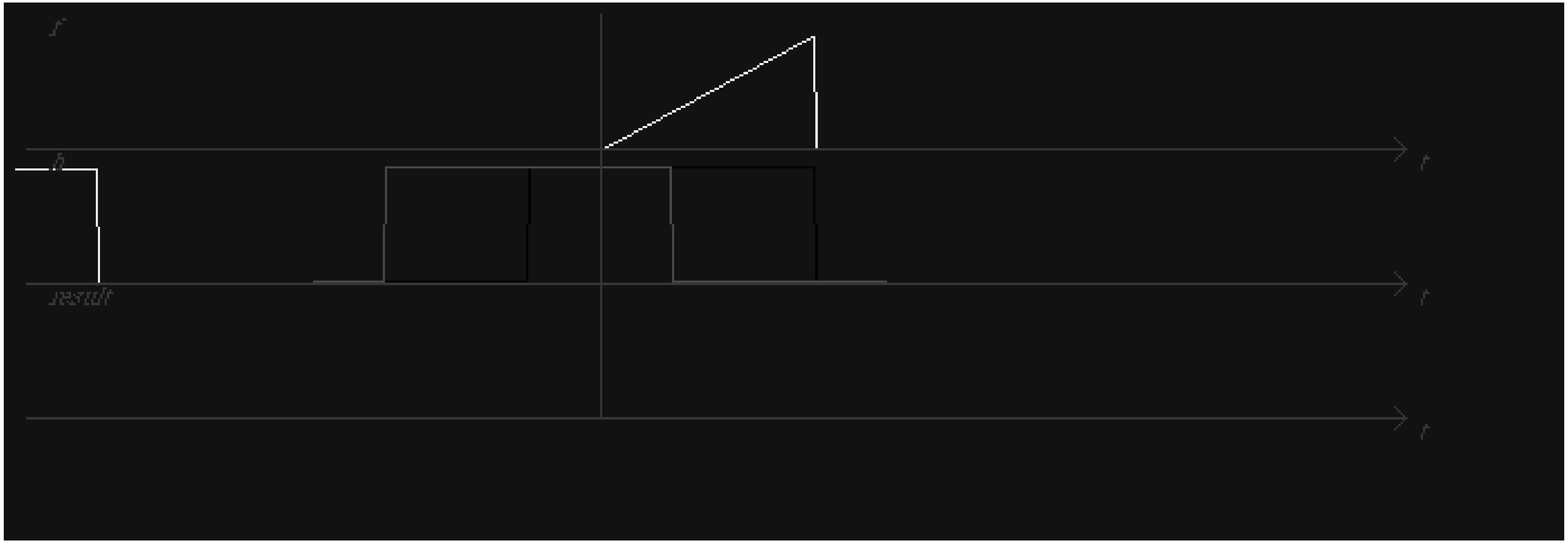
$$\prod(t/T)\otimes\prod(t/2T)=\int_{-\infty}^{\infty}\prod(u/T)\prod((t-u)/2T)du$$

- Because the pulse is zero outside -T/2<u<T/2 and unity elsewhere we can ditch it from the integral if we change the limits to -T/2 to T/2.
- We can also do a substitution of s=t-u to get a result which is…

$$\int_{t-T/2}^{T/2+t}\prod(s/2T)ds=\begin{cases}\int_{t-T/2}^{T}ds & T/2<t<3T/2\\ \int_{t-T/2}^{t+T/2}ds & -T/2<t<T/2\\ \int_{-T}^{t+T/2}ds & -3T/2<t<-T/2\end{cases}=\begin{cases}3T/2-t & T/2<t<3T/2\\ T & -T/2<t<T/2\\ 3T/2-t & -3T/2<t<-T/2\end{cases}$$

Graphically...

- A more complicated example… [MOVIE]



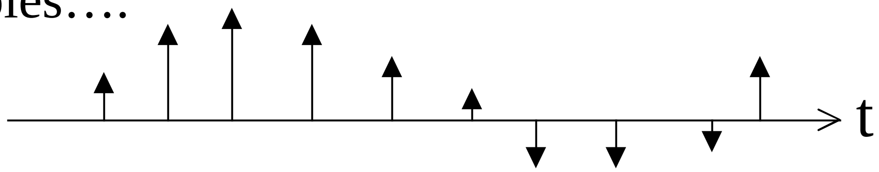
- [INSERT PAGE ON GRAPHICAL CONVOLUTION]

Properties...

- Commutative $$f(t)\otimes g(t) = g(t)\otimes f(t)$$
- Associative $$f(t)\otimes(g(t)\otimes h(t)) = (f(t)\otimes g(t))\otimes h(t)$$
- Distributive $$f(t)\otimes(g(t)+h(t)) = f(t)\otimes g(t) + f(t)\otimes h(t)$$
- Fourier transform $$\mathscr{F}\{f(t)\otimes g(t)\} = F(\omega).G(\omega)$$
- And not surprisingly… $$\mathscr{F}\{f(t).g(t)\} = F(\omega)\otimes G(\omega)$$
- **Shifting property** $$f(t)\otimes\delta(t-t_o) = \int_{-\infty}^{\infty} f(T)\delta(t-t_o-T)dT$$
- Letting $u=T-t_o$… $$\int_{-\infty}^{\infty} f(u-t_o)\delta(t-u)du = f(t-t_o)\otimes\delta(t)$$
- Now intuitively convolving any signal with a delta at the origin just gives the original signal, so CONVOLVING ANY SIGNAL WITH A SHIFTED DELTA JUST SHIFTS THE ORIGIN OF THE SIGNAL!

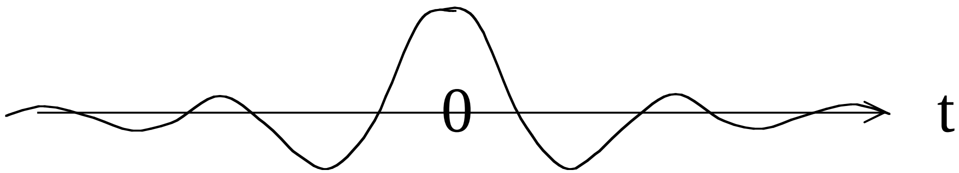
Low pass filtering

- Consider performing a low pass filtering operation using convolution. We will use a signal which might originate from DSP hardware as samples….



$$f(t) = \sum_{n=-\infty}^{\infty} f_n \delta(t - nt_o)$$

- The inverse Fourier transform of an ideal low pass filter is a sine over argument function, i.e., the impulse response of the filter is



$$Sa(t) = \frac{\sin(t)}{t}$$

- Therefore the result of passing the sampled signal through a LPF is…

$$f(t) \otimes Sa(t) = \sum_{n=-\infty}^{\infty} f_n Sa(t - nt_o)$$

- [MOVIE ONE] [MOVIE TWO]

Correlation

- An operation which is very similar to convolution is correlation. The correlation of two signals indicates how similar they are.

$$f(t) * g(t) = \int_{-\infty}^{\infty} f^*(T) g(T + t) dT$$

- A useful property of correlation is that random noise by definition will only "look like itself" if it is superimposed with zero shift. This means the autocorrelation of random noise produces a spike at the origin, but nowhere else. Because deterministic signals have autocorrelations that are also signals, it is reasonably easy to isolate the noise spike, thereby removing the noise completely from the autocorrelation of the signal. (The hard part is then regenerating the signal from its autocorrelation.)
- NOTE: The inverse Fourier transform of the "power spectrum" of a signal is its autocorrelation.

Some Fourier transform pairs...

by Bracewell, R.

100 THE FOURIER TRANSFORM AND ITS APPLICATIONS

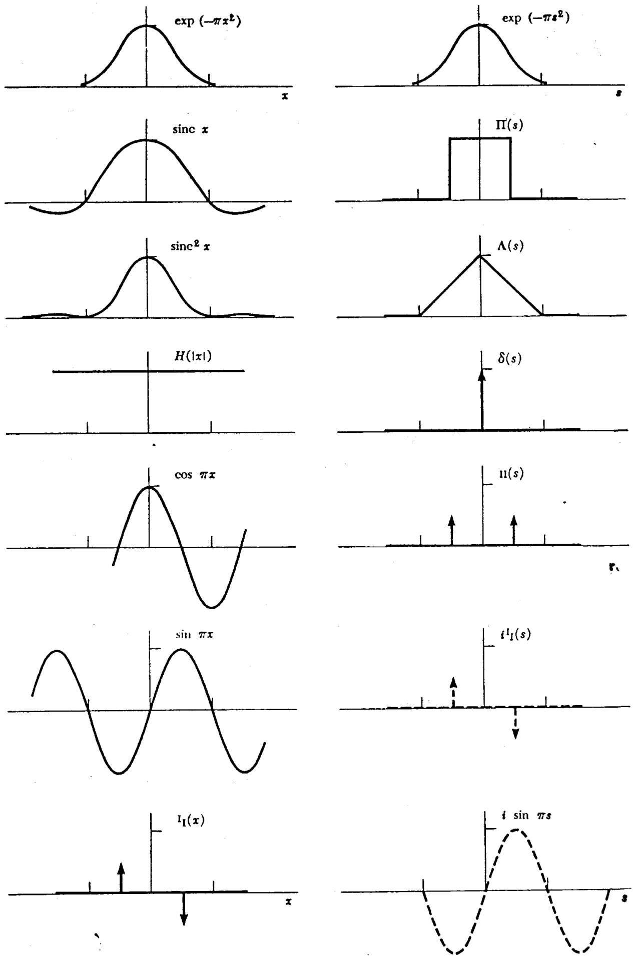


Fig. 6.1 Some Fourier transform pairs for reference.

Properties of the Fourier transform

- Addition: $$\mathscr{F}\{f(t)+g(t)\} = F(\omega)+G(\omega)$$
- Scaling: $$\mathscr{F}\{f(at)\} = \frac{1}{|a|}F\left(\frac{\omega}{a}\right)$$
- Shift (Sifting): $$\mathscr{F}\{f(t-a)\} = e^{-ja\omega}F(\omega)$$
- Modulation: $$\mathscr{F}\{f(t)\cos(\omega_o t)\} = \tfrac{1}{2}F(\omega-\omega_o)+\tfrac{1}{2}F(\omega+\omega_o)$$
- Convolution in time (multiplication in frequency): $$\mathscr{F}\{f(t)\otimes g(t)\} = F(\omega)G(\omega)$$
- Convolution in frequency (multiplication in time): $$\mathscr{F}\{f(t).g(t)\} = F(\omega)\otimes G(\omega)$$
-

continued...

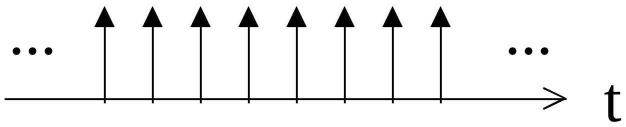
Properties of the Fourier transform….

- Autocorrelation: $\mathscr{F}\{f(t) * f(t)\} = |F(\omega)|^2$
- Derivative: $\mathscr{F}\{\frac{d}{dt} f(t)\} = j\omega F(\omega)$
- Integration: $\mathscr{F}\left\{\int f(t)dt\right\} = \dfrac{F(\omega)}{j\omega}$
- Energy Conservation:

$$\int_{-\infty}^{\infty} |f(t)|^2 dt \equiv \int_{-\infty}^{\infty} |F(\omega)|^2 d\omega$$

Fourier transform of Rack of Nails function

- A function that is useful when talking about samples of a signal is the "rack of nails" or "shah" function:



$$III(t) = \sum_{k=-\infty}^{\infty} \delta(t-k)$$

- Because this is a periodic function we can also write it as a Fourier series..

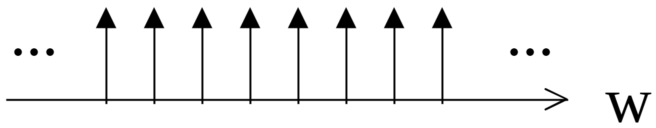
$$III(t) = \sum_{n=-\infty}^{\infty} \exp(j2\pi nt)$$

- So taking the Fourier transform..

$$\mathscr{F}\{III(t)\} = \int_{-\infty}^{\infty} dt \sum_{n=-\infty}^{\infty} \exp(j2\pi nt)\exp(j\omega t)$$

$$= \sum_{n=-\infty}^{\infty} \int_{-\infty}^{\infty} dt \exp(j2\pi nt)\exp(j\omega t) = \sum_{n=-\infty}^{\infty} \mathscr{F}\{\exp(j2\pi nt)\}$$

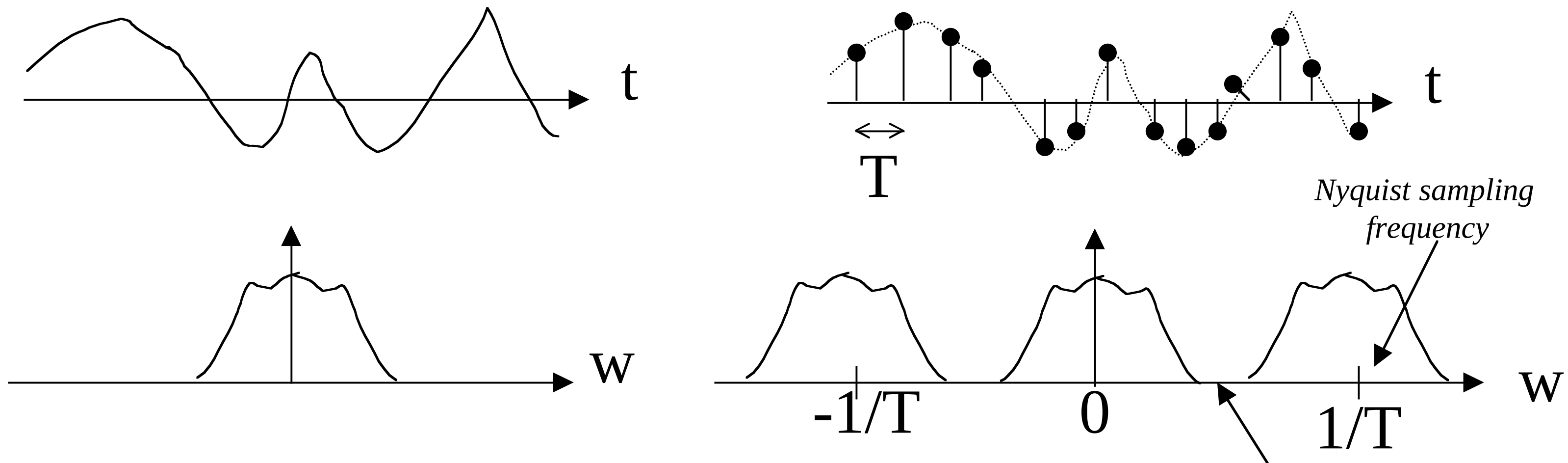
$$= \sum_{n=-\infty}^{\infty} \delta(\omega - 2\pi n) = III(\omega)$$



Going Digital...

- When we sample an analog signal we introduce repeats of the Fourier transform of the original signal…

$$\mathscr{F}\left\{ f(t).III\left({}^{t}\!/_{T} \right) \right\} = T.F(\omega) \otimes III(T\omega)$$

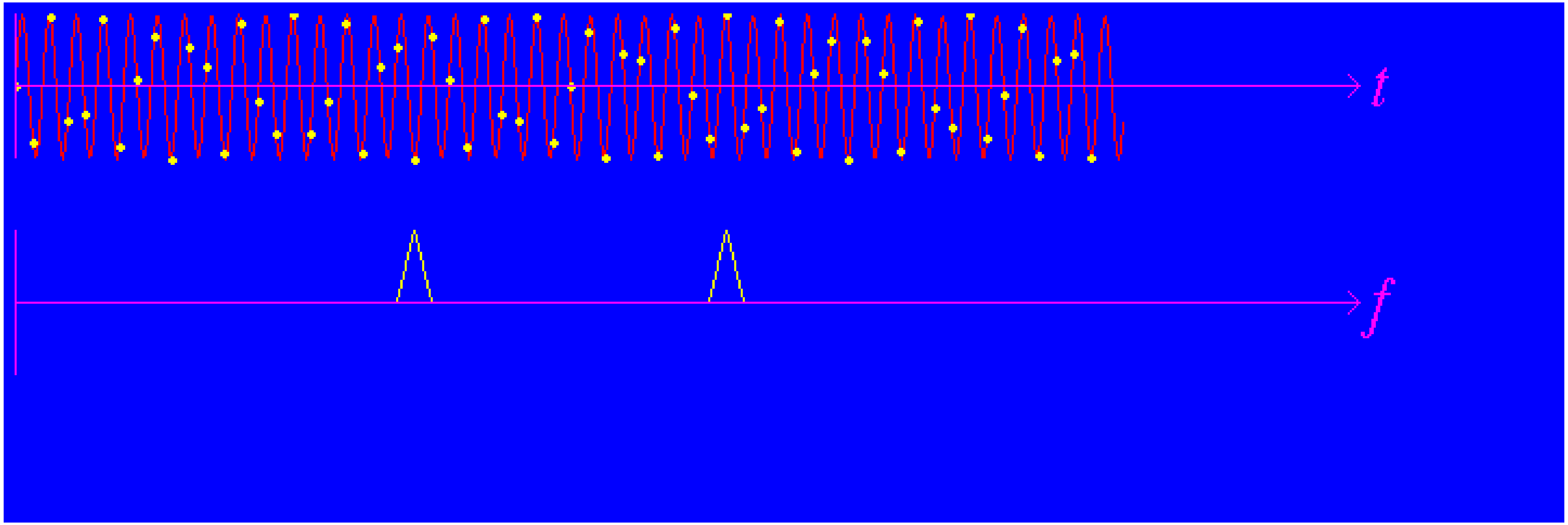


- The spacing of these repeats is critical to the sampling!

Sampling the real world...

- Need to get minimal information as numbers to represent a real world signal, such that we can reconstruct it perfectly again….
- So how much information is sufficient? Generally this means that we remove ambiguities in the reconstruction by ruling out the existence of any ambiguous waveforms, either methodologically or by physically ensuring they will never occur.
- If we fail in this then there is a possibility that signals may masquerade or “alias” as signals in which we are interested. This is a direct implication of not taking enough “samples”, that is, we have “undersampled”.
- If we know the frequency range or “bandlimit” of our input, then we can determine the minimum sampling rate required to ensure proper reconstruction of the signal. To ensure the bandlimit is adhered to, we use an analog “anti-aliasing” filter prior to any A to D conversion.

Sampling and Aliasing



- [MOVIES]

Sampling Theorem

- If we know the signal is bandlimited to B then the minimum sampling rate or “Nyquist sampling frequency” required is **greater than** 2B.
- e.g. A speech signal is easily interpretable if confined to a baseband 3 kHz bandlimit. Therefore the sampling frequency > 6 kHz. That is, a time between samples of less than 1/6000 s = 166 µs.
- Note that this frequency content does not need to be baseband. It can be anywhere in the spectrum, but so long as we use this information about where it is centred in frequency when reconstructing it, we can use its content at baseband.
- e.g. The speech signal above is still intelligible if we remove the frequencies below 300 Hz. Therefore the bandlimit is only 2.7 kHz, so the sampling rate may be reduced to greater than 5.4 kHz. (This minor reduction in sampling rate might make the difference between cheap and expensive hardware!)

Example: Slope Sampling

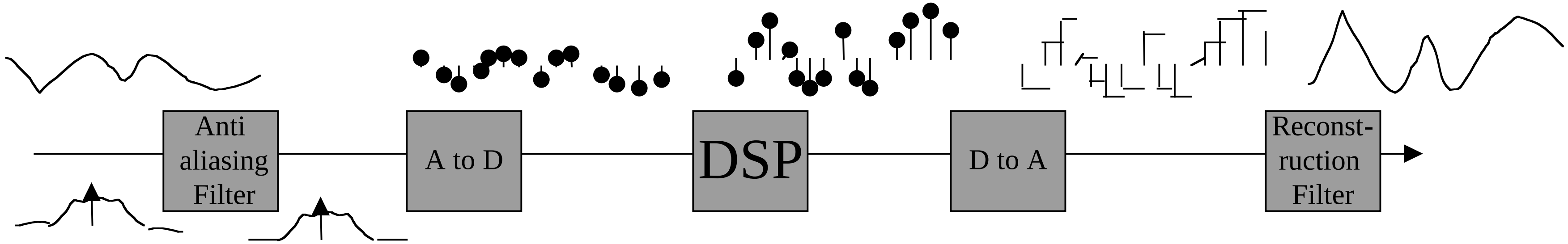
- It is not necessary to gain constant spaced samples to obey the sampling theorem, nor is it necessary to have a sampling rate greater than twice the bandlimit of the signal - these are just easy to arrange.
- We can halve the required sampling rate, for example, and obtain the necessary information per time period by sampling both the analog signal and its analog derivative. We therefore obtain N samples within the same time, but from two sources.
- The original signal may be reconstructed from these two sample streams, mathematically at least, by using different interpolating functions for each. (This requires fashioned reconstruction filters and an adder).

$$f(t) = a(t) \otimes f_D(t) + b(t) \otimes f'_D(t)$$

$$a(t) = \text{sinc}^2(t) \qquad b(t) = t\,\text{sinc}^2(t) \qquad c.f. \quad h(t) = \text{sinc}(t)$$

Reconstruction

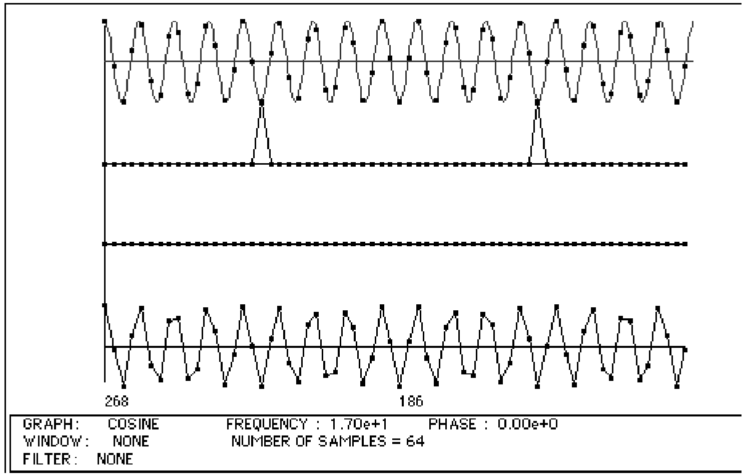
- The reconstruction of an analog signal from sampled data involves passing the output of a D to A converter through the identical filter used to anti-alias the input. (We have already discussed how this is the convolution of the sampled data with the impulse response of the filter)

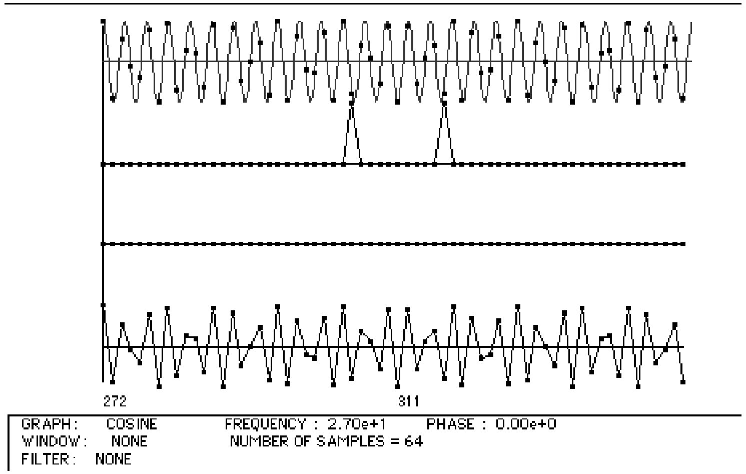


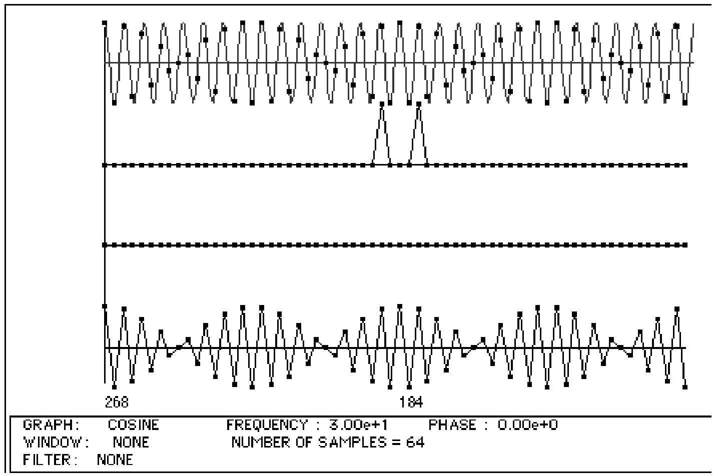
- Consider the effect of the steps in the output of the D to A converter...

Beating

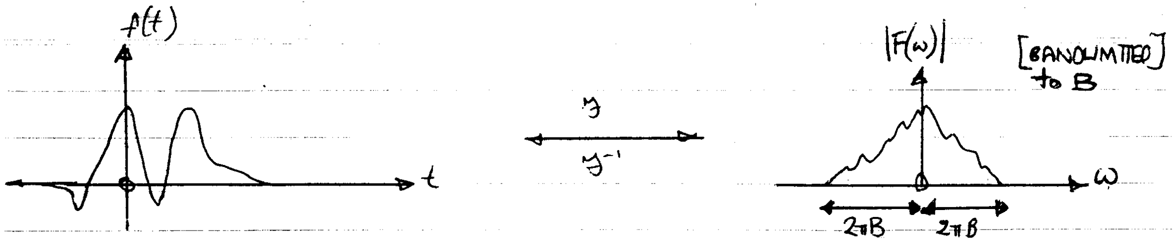
- Even if we obey the sampling theorem,we might still observe "funny" effects:
  - This apparent beating is purely a result of poor reconstruction
  - "join the dots" versus ideal impulse response
  - We must be careful though, since non-ideal filters, or precision of arithmetic during processing might let us down
  - Solution: Guard Bands/Zero Padding in freq.domain.





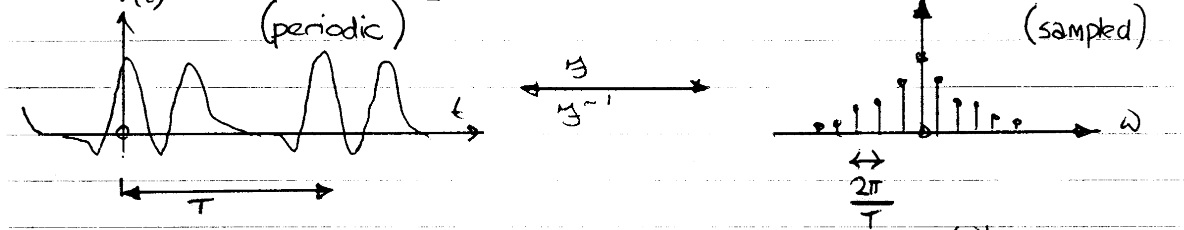


N.B. if $f(t)$ is real then $|F(\omega)|$ is symmetric
(phase is antisymmetric)

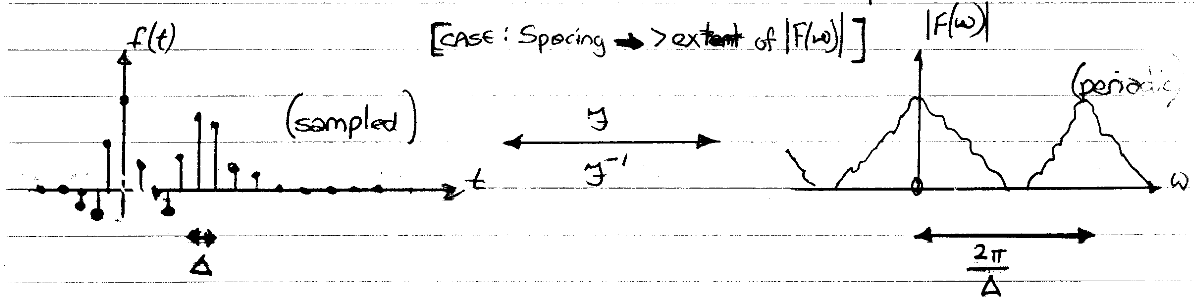


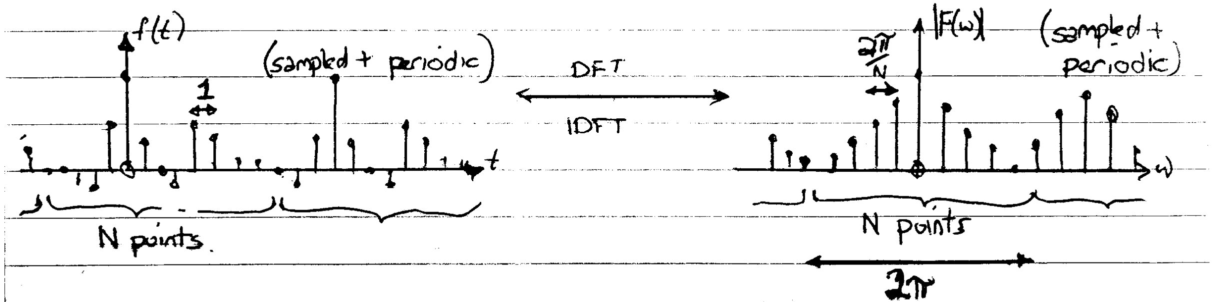
N.B. f(t) need not be of limited extent.

[CASE: T > extent of f(t)]



[CASE: Spacing → > extent of |F(ω)|]





N.B. For DFT uses we drop the 2π in the frequency domain.

THE EFFECTS OF SAMPLING

Block Artefacts...

- To perform any useful processing we must acquire data, and are of course limited in sampling rate and in storage capacity or available time for computation….
- We are forced to take a limited block of N samples and operate on them as though they contain all the information about the entire signal. Clearly, if we do one of two obvious extensions - either assume zero outside this block, or assume the block repeats ad infinitum - then we will often have assumed incorrectly!
- Therefore need to be aware of the problems this “blocking” can introduce, and adopt methods to minimise any error introduced…

Spectral Leakage

- If the block of N samples is a period of a periodic signal, say a sine wave of integer number of cycles within the block, then no artifacts are introduced by forcing periodicity, but if this is not the case, discontinuities are introduced at the ends of the block, and subsequent processing will be corrupted to some extent. This phenomenon is most obvious in the frequency domain as “leakage” of erroneous frequencies into the spectrum…

Windowing...

- Most likely we don't have knowledge of the period of the signal, if any, so we must adopt ways to reduce the spectral leakage.
- The "windows" we apply on the signal are designed to reduce the discontinuities at the ends of the block, while minimising the effect on the rest of the samples.
- A window is a function that is multiplied by the signal, and therefore its Fourier transform is convolved with the signal's spectrum. This is itself introduces leakage, but "LIFE IS A TRADEOFF!"
- The best choice of window is one tailored to the type of data being processed…
- e.g. Hanning, Hamming, Kaiser, (Rectangular), etc.

Circular Convolution

- We can not avoid convolution (in either domain) if we are to process a signal. Any time we multiply (window) a signal by another, their Fourier transforms are convolved, or equivalently, if we multiply (filter) their Fourier transforms, the signals are convolved!
- Unfortunately, when we assume the block of data is repeated to make a periodic signal, the convolution of adjacent blocks “slides” into the next block, or if you prefer it “wraps around” on itself. This is what we call “circularity”.
- The only way to avoid the ensuing self-corruption of the result is to take precautions prior to processing, and introduce “guard bands” of extra zeroes to each signal. This is known as “zero-padding”.
- We MUST do this even when it is not obvious from the processing we plan to do. For example, filtering a signal…..

Circular Convolution contd.

- The circular convolution of two sequences *f* and *h* is represented as:

$$g(k) = \sum_{i=0}^{N-1} f(i)h([k-i]\mathrm{MOD}N) \quad \forall\, k = 0 \ldots N-1$$

- Where the MOD function (is not the integer remainder function found in Pascal) ensures that *k-i* remains between 0 and N-1, by subtracting multiples of N if greater than N, and adding multiples of N if less than 0. (This is “clock” arithmetic base N.)

- [INSERT MOVIE HERE]

Solution: Guard Bands

- Because circular convolution wraps the result around on itself, corrupting the result, we must add “guard” bands of zero space to the signals before convolving. This provides room for the convolution result to spread over “nothing” before it wraps around, therefore not corrupting the result. (Remember that filtering causes a convolution, so we must “zero pad” the signal before taking its FT; this is “interpolating” the FT.)

- [INSERT MOVIE HERE]