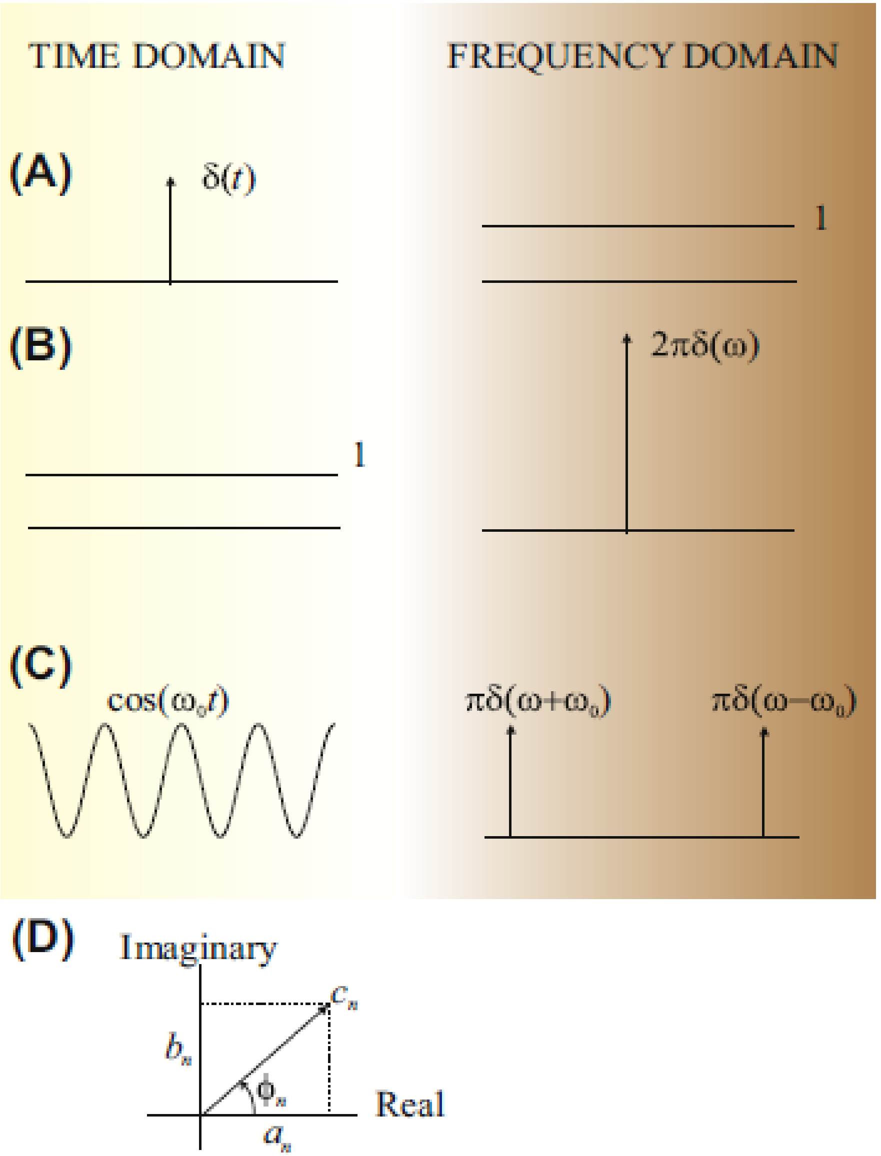
Brandenburgische Technische Universität Cottbus - Senftenberg

Chapter 5

Continuous, Discrete, and Fast Fourier Transform

Examples of Continuous Fourier Transform Pairs



Common Fourier transform pairs.
(A) A Dirac impulse function in the time domain is represented by all frequencies in the frequency domain.
(B) This relationship can be reversed to show that a DC component in the time domain generates an impulse function at a frequency of zero.
(C) A pure (cosine) wave shows peaks at w0 in the frequency domain.
(D) In general a coefficient cn being part of F(jw) may contain both real (an) and imaginary (bn) parts (represented here in a polar plot).

Time/Spatial Domain $f(t)$	Frequency Domain $F(\omega)$
$\delta(t)$	1
1	$2\pi\,\delta(\omega)$
$\cos(\omega_0 t)$	$\pi[\delta(\omega+\omega_0)+\delta(\omega-\omega_0)]$
$\sin(\omega_0 t)$	$j\pi[\delta(\omega+\omega_0)-\delta(\omega-\omega_0)]$

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

Figure 6.2 and table 6.1, Signal processing for Neuroscientist, second edition by Wim van Drongelen

DISCRETE FOURIER TRANSFORM

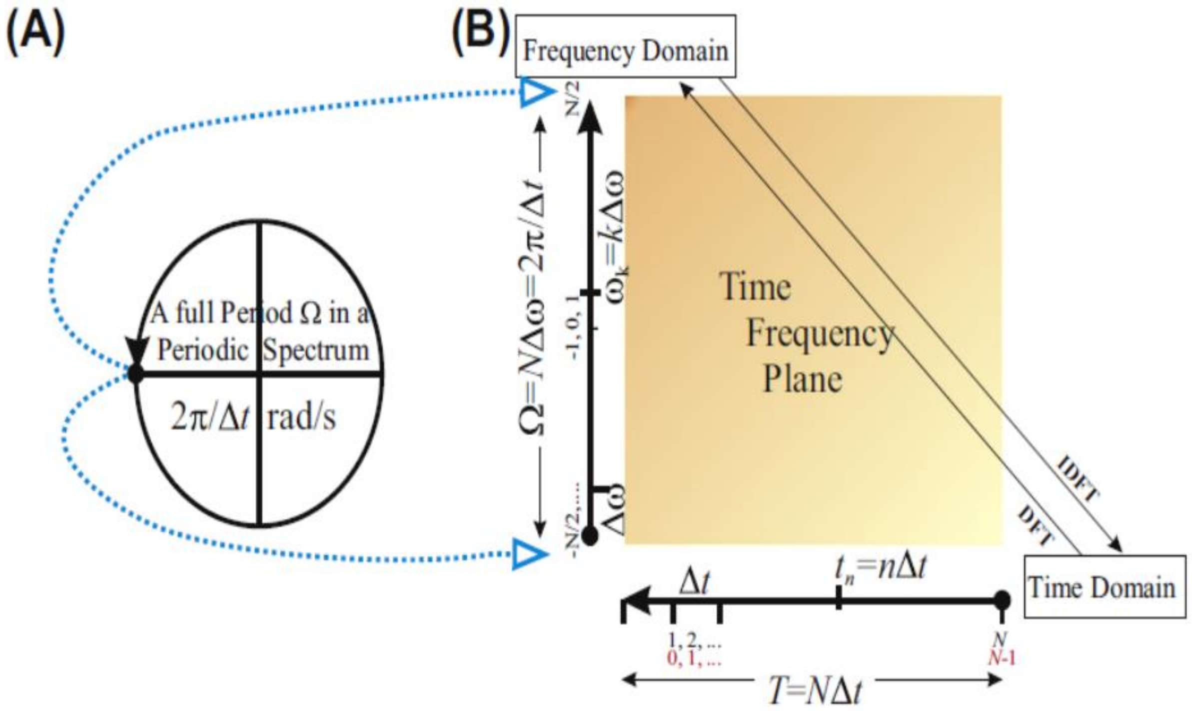
Relationship Between Continuous and Discrete Fourier Transform

The continuous and discrete Fourier transforms and their inverses are related but not identical. For the discrete pair we use a discrete time scale and a discrete frequency scale. Because we want to apply the discrete transform to **sampled real-world signals**, both the time and frequency scales must also necessarily be **finite**. Furthermore, we can establish that both scales must be related. For example, in a signal that is observed over a 10-s interval T and sampled at an interval Δt = 1 ms (0.001 s), these parameters determine the **range** and **precision** of the discrete Fourier transform of that signal.

Precision: In a T-s interval we cannot distinguish frequencies below a precision of Δ f = 1/T if T=10 then Δ f = 0.1 Hz.

Maximum frequency : fits within the sample interval is 1/ Δt = 1/0.001 = 1000 Hz.

In angular frequency terms, the precision and maximum frequency translate into a step size of Δω = 2π× 1/10 rad/s and a range of Ω= 2π× 1000 rad/s.



Fourier transform.

The Fourier spectrum is periodic, represented by a circular scale in (A). This circular frequency domain scale is mapped onto a line represented by the ordinate in (B). The abscissa in (B) is the time domain scale; note that on the frequency (vertical) axis, the point N/2 is included and N/2 is not. Each sample in the time domain represents the preceding sample interval. Depending on which convention is used, the first sample in the time domain is either counted as the zeroth sample (indicated in red) or the first one (indicated in black).

Figure 6.1, Signal processing for Neuroscientist, second edition by Wim van Drongelen

DISCRETE FOURIER TRANSFORM …

- The discrete approximation $F_a(j\omega)$ of CFW $F(j\omega) = \int_{-\infty}^{\infty} f(t)e^{-j\omega t}dt$ ampled over a finite interval including N samples is:

$$F_a(j\omega_k) = \sum_{n=1}^{N} f(t_n)\exp(-j\omega_k t_n)\Delta t$$

If $t_n = n\,\Delta t$ and $\omega_k = k\,\Delta\omega = k\,2\pi/N\Delta t$ → $F_a(j\omega_k) = \Delta t \sum_{n=0}^{N-1} f(t_n)e^{-j\frac{2\pi}{N}kn} = \Delta t \sum_{n=0}^{N-1} f(t_n)W_N^{kn}$

where W_N^{kn} is a notational simplification of the exponential term. Smuggling Δt out of the above expression, changing $f(t_n)$ to $x(n)$, and $F_a(jw_k)$ to $X(k)$ yields the standard definition for the DFT:

$$\boxed{X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn}}$$

Chapter6, Signal processing for Neuroscientist, second edition by Wim van Drongelen

Inverse DFT

Inverse continuous Fourier transform (ICFT) can be approximated by:

$$f_a(t_n) = \frac{1}{2\pi} \sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1} F_a(j\omega_k) e^{j\omega_k t_n} \Delta\omega$$

Note that the upper summation limit does not include N/2; due to the circular scale of ω, N/2 and N/2 are the same. Changing the range of summation from –N/2-N/2-1 into 0-N-1 and Δω = 2π/N Δt , yields:

$$f_a(t_n) = \frac{1}{2\pi} \sum_{k=0}^{N-1} F_a(j\omega_k) e^{j\omega_k t_n} \frac{2\pi}{N\Delta t} = \frac{1}{N\Delta t} \sum_{k=0}^{N-1} F_a(j\omega_k) e^{j\omega_k t_n}$$

We now use $t_n = n\,\Delta t \; and \; w_k = k2\,\pi / N\,\Delta t$, smuggle Δt back, change $f_a(t_n)$ to $x(n)$ and $f_a(t_n)$ to $X(k)$, thereby obtaining the expression for the discrete inverse Fourier transform:

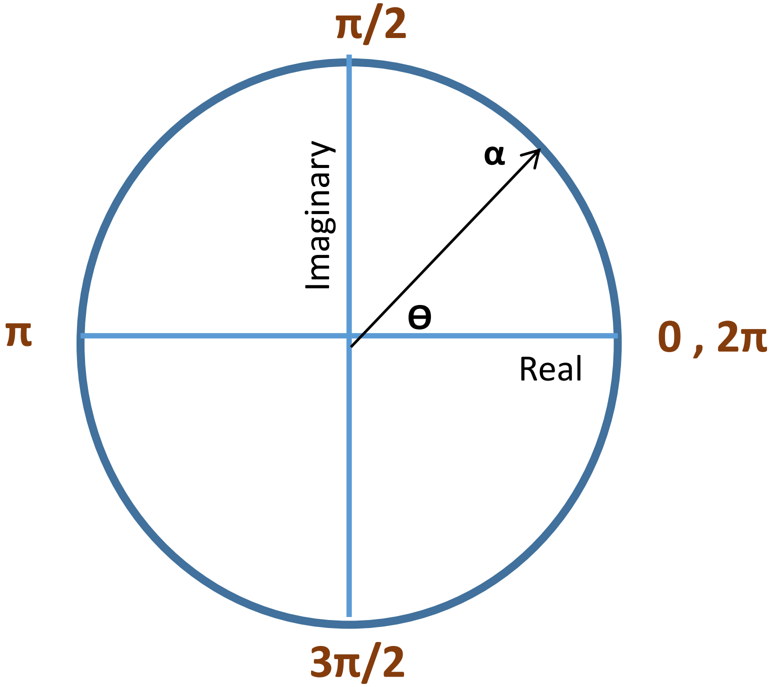
$$\boxed{x(n) = \frac{1}{N} \sum_{k=0}^{N-1} X(k) e^{j\frac{2\pi}{N}kn} = \frac{1}{N} \sum_{k=0}^{N-1} X(k) W_N^{-kn}}$$

Twiddle Factor (W_N)

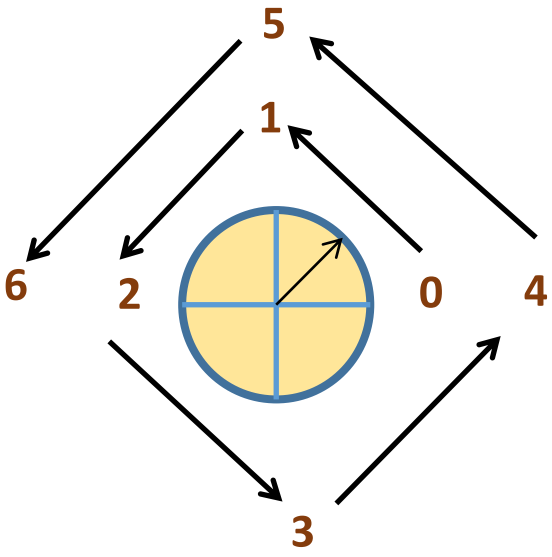
The weighting factor introduced as W_N in the above formulae plays an important role in the practical development of DFT algorithms including the optimized one known as the FFT. The efficiency of this algorithm relies crucially on the fact that this factor, also known as the “twiddle” factor, is periodic.

$$W_4^4 = e^{-\frac{2\pi}{4}j\times 4} = e^{-\frac{2\pi}{4}j\times 0}=1$$

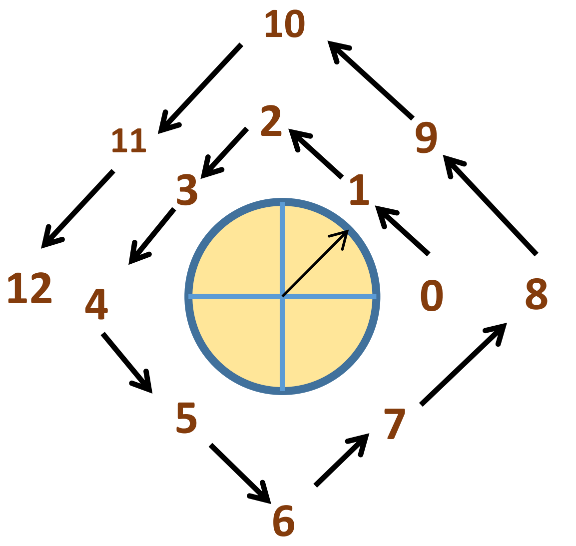
$$W_4^6 = e^{-\frac{2\pi}{4}j\times 6} = e^{-\frac{2\pi}{4}j\times 2}=-1$$



$$W_4 = e^{-\frac{2\pi}{4}j}$$



$$W_8 = e^{-\frac{2\pi}{8}j}$$



The values of **θ** are indicated in red and the real and imaginary components in black. For instance 1+ j0 is associated with **θ=** π, 0 + j is associated with **θ=** π/2, etc. It can be seen that for **θ=** 0 or 2π, the values are identical (1+j0) due to the periodicity of WN.

concrete examples are provided ,for the periodicity of a four-point (W4) and eight-point (W8) algorithm. The numbers correspond to the powers of the twiddle factor (e.g., 0□W_4^0; 1□W_4^1; 2□W_4^2 , etc.): in case of N = 4, a cycle is completed in four steps; whereas for N = 8 the cycle is completed in eight steps. In the first case (B): $W_4^0 = W_4^4$; $W_4^1 = W_4^5$; and $W_4^2 = W_4^6$. In the second example (C): $W_8^0 = W_8^8$; $W_8^2 = W_8^{10}$, etc.

Fast Fourier Transform (FFT)

- The basic idea used to optimize the DFT algorithm involves using the periodicity in the twiddle factor to combine terms and therefore reduce the number of computationally demanding multiplication steps required for a given number of samples (Cooley and Tukey, 1965). Specifically, the standard formulation of the DFT of a time series with N values requires N^2 multiplications for a time series with N points; whereas, the FFT requires only Nlog2(N) multiplications.
- For example, consider a four-point time series: x(0), x(1), x(2), x(3) and its DFT X(0), X(1), X(2), X(3). For N = 4 each of the X values is calculated with

$$\boxed{X(k)=\sum_{n=0}^{N-1} x(n)W_N^{kn}}$$

$$X(k)=\sum_{n=0}^{3} x(n)W_4^{kn}$$

$$X(k)=\sum_{r=0}^{1} x(2r)W_4^{2rk}+\sum_{r=0}^{1} x(2r+1)W_4^{(2r+1)k}$$

$$W_4^{(2r+1)k}=W_4^{2rk}W_4^{k}$$

$$\begin{aligned}X(0)&=x(0)W_4^0+x(2)W_4^0+x(1)W_4^0W_4^0+x(3)W_4^0W_4^0\\&=\left[x(0)+x(2)W_4^0\right]+W_4^0\left[x(1)+x(3)W_4^0\right]\end{aligned}$$

$$\begin{aligned}X(1)&=x(0)W_4^0+x(2)W_4^2+x(1)W_4^1W_4^0+x(3)W_4^1W_4^2\\&=\left[x(0)+x(2)W_4^2\right]+W_4^1\left[x(1)+x(3)W_4^2\right]\end{aligned}$$

$$\begin{aligned}X(2)&=x(0)W_4^0+x(2)W_4^4+x(1)W_4^2W_4^0+x(3)W_4^2W_4^4\\&=\left[x(0)+x(2)W_4^4\right]+W_4^2\left[x(1)+x(3)W_4^4\right]\end{aligned}$$

$$\begin{aligned}X(3)&=x(0)W_4^0+x(2)W_4^6+x(1)W_4^3W_4^0+x(3)W_4^3W_4^6\\&=\left[x(0)+x(2)W_4^6\right]+W_4^3\left[x(1)+x(3)W_4^6\right]\end{aligned}$$

Fast Fourier Transform (FFT)

- The basic idea used to optimize the DFT algorithm involves using the periodicity in the twiddle factor to combine terms and therefore reduce the number of computationally demanding multiplication steps required for a given number of samples (Cooley and Tukey, 1965). Specifically, the standard formulation of the DFT of a time series with N values requires N^2 multiplications for a time series with N points; whereas, the FFT requires only Nlog2(N) multiplications.
- For example, consider a four-point time series: x(0), x(1), x(2), x(3) and its DFT X(0), X(1), X(2), X(3). For N = 4 each of the X values is calculated with

$$X(k) = \sum_{n=0}^{3} x(n) W_4^{kn}$$

$$\boxed{X(k) = \sum_{n=0}^{N-1} x(n) W_N^{kn}}$$

$$X(k) = \sum_{r=0}^{1} x(2r) W_4^{2rk} + \sum_{r=0}^{1} x(2r+1) W_4^{(2r+1)k}$$

$$W_4^{(2r+1)k} = W_4^{2rk} W_4^{k}$$

$$\begin{aligned} X(0) &= x(0)W_4^0 + x(2)W_4^0 + x(1)W_4^0W_4^0 + x(3)W_4^0W_4^0 \\ &= \left[x(0) + x(2)W_4^0\right] + W_4^0\left[x(1) + x(3)W_4^0\right] \end{aligned}$$

$$\begin{aligned} X(1) &= x(0)W_4^0 + x(2)W_4^2 + x(1)W_4^1W_4^0 + x(3)W_4^1W_4^2 \\ &= \left[x(0) + x(2)W_4^2\right] + W_4^1\left[x(1) + x(3)W_4^2\right] \end{aligned}$$

$$\begin{aligned} X(2) &= x(0)W_4^0 + x(2)W_4^4 + x(1)W_4^2W_4^0 + x(3)W_4^2W_4^4 \\ &= \left[x(0) + x(2)W_4^4\right] + W_4^2\left[x(1) + x(3)W_4^4\right] \end{aligned}$$

$$\begin{aligned} X(3) &= x(0)W_4^0 + x(2)W_4^6 + x(1)W_4^3W_4^0 + x(3)W_4^3W_4^6 \\ &= \left[x(0) + x(2)W_4^6\right] + W_4^3\left[x(1) + x(3)W_4^6\right] \end{aligned}$$