Spectral Analysis using Discrete-time Data

AE4304 lecture # 4, edition 2017-2018

prof. dr ir Max Mulder

m.mulder@tudelft.nl

Spectral Analysis using Discrete-time Data

For this lecture the following material was used:

- Chapter 4 of the lecture notes *AE4304 Aircraft Responses to Atmospheric Turbulence*
- *Systems & Signals* by Oppenheim, Wilsky & Young
- *Digital Control System Analysis and Design* by Phillips & Nagle
- *Digital Signal Processing* by Roberts & Mullis

Sampling

Signal reconstruction

The discrete-time Fourier transform (DTFT)

The Discrete Fourier transform (DFT)

study The Fast Fourier transform (FFT) at home

The calculation of spectral estimates

Miscellaneous

Introduction to spectral analysis in discrete time

In Chapter 3 we have discussed the continuous-time Fourier analysis, which is of great theoretical value for the analysis and insight into the properties of continuous-time stochastic signals. However, any *practical* computer-based application requires the use of a **discrete time series**. In this lecture the spectral analysis of continous-time stochastic processes, using discrete-time data (samples) is treated.

The obvious method to estimate PSD functions from measured data would be to estimate the covariance functions first and then to apply a (discrete-time) Fourier transform. Because of the availability of the Fast-Fourier Transform (FFT), however, which is extremely efficient (fast), it has become a more common procedure to calculate the PSD functions **directly** from the original time series data.

$\bar{x}(t) \to C_{\bar{x}\bar{x}}(\tau) \to S_{\bar{x}\bar{x}}(\omega)$ has become $\bar{x}[n] \to S_{\bar{x}\bar{x}}[k]$

An often used Fourier transform pair:

A block in time-domain becomes a sinc-function in frequency:

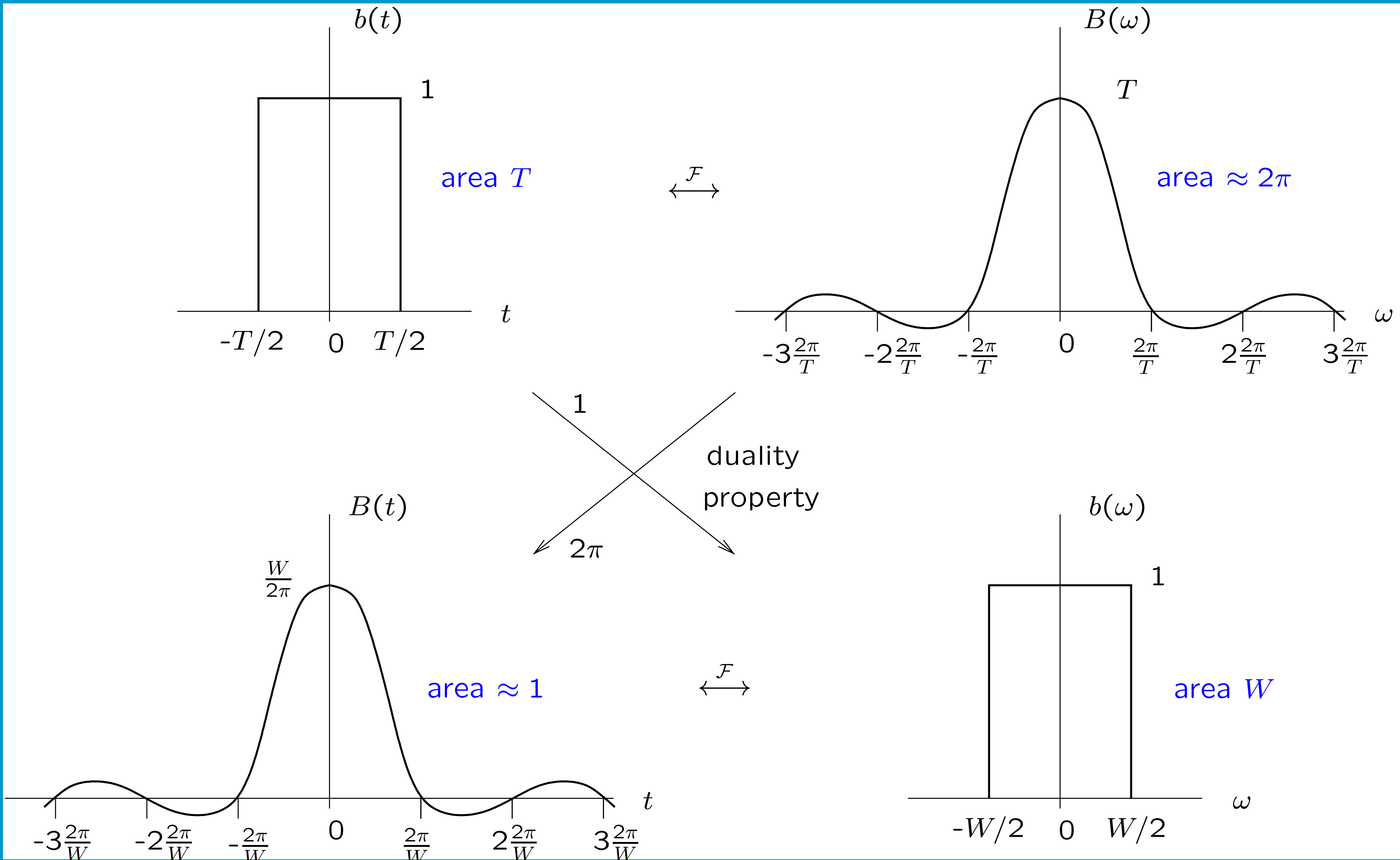
$$\boxed{\Pi\left(\frac{t}{T}\right) \;\xleftrightarrow{\mathcal{F}}\; T\text{sinc}\left(\frac{\omega T}{2\pi}\right)}$$

(area T) in time, zeros at $\omega = m\frac{2\pi}{T}$

A block in frequency-domain becomes a sinc-function in time:

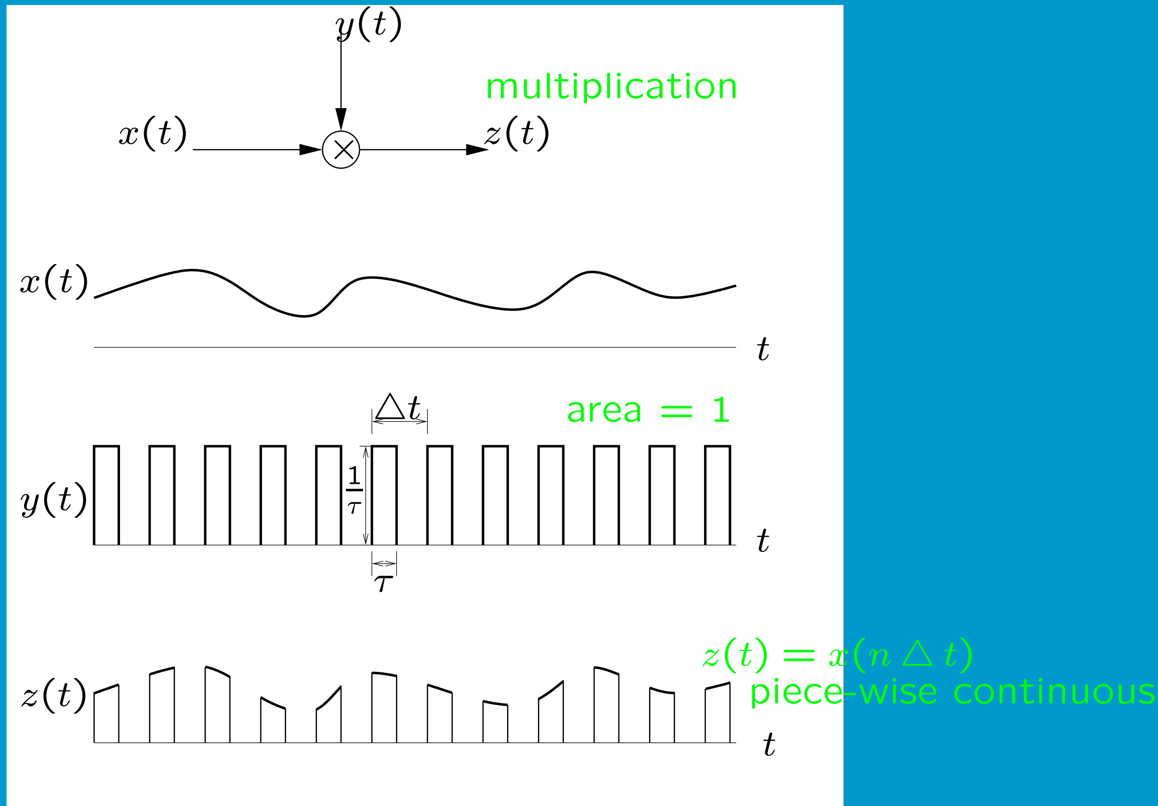
$$\boxed{\frac{W}{2\pi}\text{sinc}\left(\frac{tW}{2\pi}\right) \;\xleftrightarrow{\mathcal{F}}\; \Pi\left(\frac{\omega}{W}\right)}$$

(area W) in frequency, zeros at $t = m\frac{2\pi}{W}$



duality: area in $t \to$ height in ω, area in $\omega \cdot 1/2\pi \to$ height in t.

Sampling: C $\longrightarrow$ D



When τ becomes very small, we speak of **impulse-train sampling**.

It is known from the previous lecture that a multiplication of two signals in the time domain:

$$z(t) = x(t) \cdot y(t), \tag{4.1}$$

corresponds with a **convolution** operation in the frequency domain:

$$Z(\omega) = \frac{1}{2\pi} X(\omega) \star Y(\omega) = \frac{1}{2\pi} \int\limits_{-\infty}^{\infty} X(\xi) Y(\omega - \xi)\, \mathrm{d}\xi \tag{4.2}$$

Now, let $x(t)$ be the continuous-time (possibly aperiodic) signal which is to be sampled, and let $X(\omega)$ denote its continuous-time Fourier transform.

The signal $y(t)$ is selected to be a **pulse train** in the time domain (Table 3.2 of the lecture notes):

$$y(t) = \sum_{n=-\infty}^{\infty} \delta(t - n\triangle t), \qquad \text{Dirac pulses ('comb')} \qquad (4.3)$$

which is referred to as the <u>sampling function</u>, with $\triangle t$ as the **sampling period** and the fundamental frequency $\omega_0 = 2\pi/\triangle t$ as the **sampling frequency** ω_s.

Then, the continuous-time FT of $y(t)$ results in (Table 3.2):

$$\boxed{Y(\omega) = \frac{2\pi}{\triangle t} \sum_{k=-\infty}^{\infty} \delta(\omega - k\frac{2\pi}{\triangle t})}, \qquad \left(= \omega_s \sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_s)\right) \quad (4.4)$$

i.e., a **pulse train in the frequency domain**.

Proof

The proof of the property that a pulse train in the time domain becomes a pulse train in the frequency domain is relatively easy. Yet, to increase our insight, we will consider the problem from two perspectives.

First, let us take an analytical approach. The pulse train $y(t)$ can be considered a *periodic* function in the time domain with fundamental period $\triangle t$ (and fundamental frequency $\omega_0 = \frac{2\pi}{\triangle t}$). Therefore, we can apply the complex CTFS expansion to compute the FS coefficients c_k:

$$c_k \;=\; \frac{1}{\triangle t} \int\limits_{-\triangle t/2}^{\triangle t/2} \delta(t)\, e^{-j\,k\omega_0\,t} \mathrm{d}t \;=\; \frac{1}{\triangle t} e^0 \int\limits_{-\triangle t/2}^{\triangle t/2} \delta(t)\,\mathrm{d}t \;=\; \frac{1}{\triangle t} \qquad \forall\ k \qquad (=\text{“sifting property”})$$

The FS coefficients are therefore *all* equal to $\frac{1}{\triangle t}$, and the CTFS signal approximation becomes:

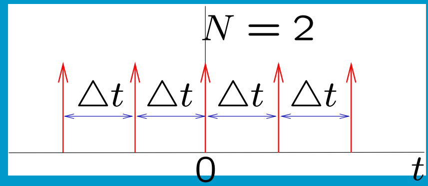
$$\tilde{y}(t) \;=\; \frac{1}{\triangle t} \sum_{k=-\infty}^{\infty} e^{j\,k\omega_0\,t}$$

Computing the CTFT of $\tilde{y}(t)$ yields:

$$\begin{aligned}
\tilde{Y}(\omega) \;&=\; \int\limits_{-\infty}^{\infty} \left(\frac{1}{\triangle t} \sum_{k=-\infty}^{\infty} e^{j\,k\omega_0\,t} \right) e^{-j\,\omega t}\,\mathrm{d}t \;=\; \frac{1}{\triangle t} \sum_{k=-\infty}^{\infty} \int\limits_{-\infty}^{\infty} e^{j\,k\omega_0\,t}\, e^{-j\,\omega t}\,\mathrm{d}t \\
&=\; \frac{1}{\triangle t} \sum_{k=-\infty}^{\infty} \int\limits_{-\infty}^{\infty} e^{-j\,(\omega - k\omega_0)\,t}\,\mathrm{d}t \;=\; \frac{1}{\triangle t} \sum_{k=-\infty}^{\infty} 2\pi\,\delta(\omega - k\omega_0) \\
&=\; \frac{2\pi}{\triangle t} \sum_{k=-\infty}^{\infty} \delta(\omega - k\omega_0) \qquad\qquad \text{q.e.d.}
\end{aligned}$$

Second, let us look at things in a more practical sense. Consider a *finite* set of $2N+1$ Dirac pulses:

$$
\begin{aligned}
x(t) &= \sum_{n=-N}^{n=N} \delta(t - n\triangle t) \\
&= \delta(t) + \sum_{n=1}^{n=N} [\delta(t + n\triangle t) + \delta(t - n\triangle t)]
\end{aligned}
$$



The Fourier transform of this function yields:

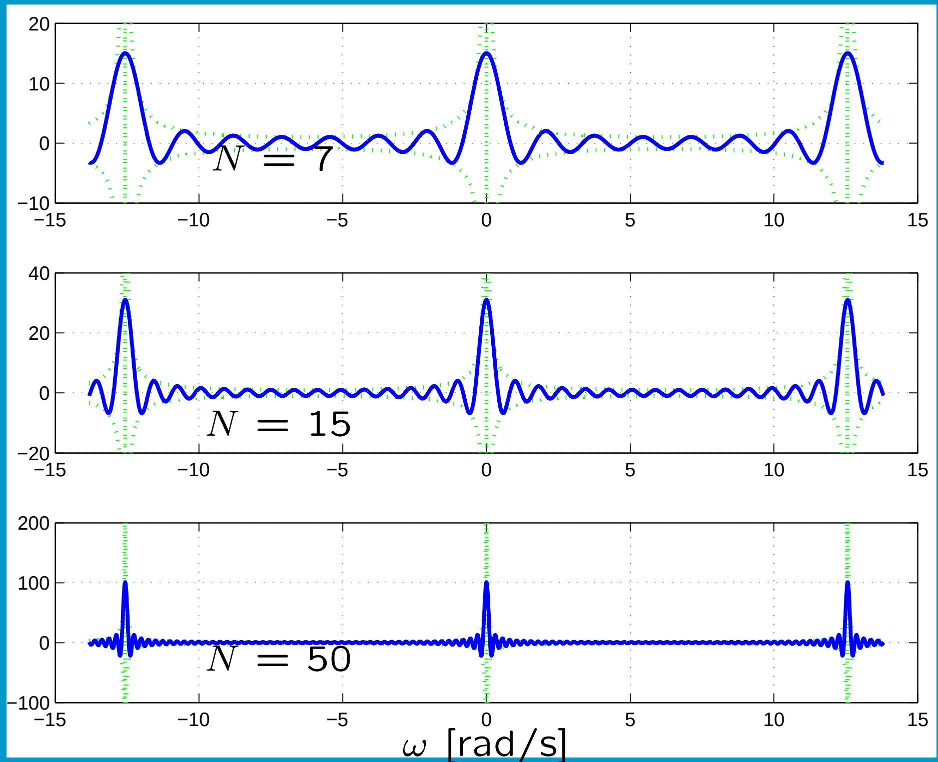
$$
\mathcal{F}\{x(t)\} = 1 + \sum_{n=1}^{n=N} (e^{j\omega n\triangle t} + e^{-j\omega n\triangle t}) = 1 + 2\sum_{n=1}^{n=N} \cos(\omega n\triangle t), \quad \text{so:}
$$

$$
\boxed{X(\omega) = 1 + 2\sum_{n=1}^{n=N} \cos(\omega n\triangle t)} \qquad (*)
$$

This is a periodical function in the frequency domain with "period" F equal to $\frac{1}{\triangle t}$. It is written like a Fourier series expansion, but now not in 'time', but rather in 'frequency' (ω is the running variable), containing only the cosine terms with all 'a_k's equal to 2. But what does the 'original function' in the frequency domain look like of which (*) is the 'Fourier series expansion in the frequency domain'. It can be shown that this 'original' is the following function:

$$
\boxed{X(\omega) = \frac{\sin((2N+1)\omega\triangle t/2)}{\sin(\omega\triangle t/2)}}
$$

Below this function is shown for a sample time $\triangle t$ of 0.5 s. It is periodic with a 'period' of $2\pi/\triangle t$ $= 4\pi$ rad/s. Also, when N increases the function 'compresses' near the periods, while remaining within the **envelopes** defined by the functions $-1/\sin(\omega\triangle t/2)$ and $+1/\sin(\omega\triangle t/2)$.



When $N \to \infty$ the function takes on any value within the contours, except of the points $(k2\pi/\triangle t, -\infty), k =$ $..., -2, -1, 0, 1, 2, ...$ Hence, it can be considered to <u>have all properties of a Dirac function</u>.

Now let us continue with investigating the sampled signal $z(t)$:

$$\begin{aligned}
Z(\omega) &= \tfrac{1}{2\pi} X(\omega) \star Y(\omega) \\
&= \tfrac{1}{2\pi} \int_{-\infty}^{\infty} X(\xi) Y(\omega - \xi)\, \mathrm{d}\xi && \text{definition of convolution} \\
&= \tfrac{1}{2\pi} \int_{-\infty}^{\infty} X(\xi) \left(\frac{2\pi}{\triangle t} \sum_{k=-\infty}^{\infty} \delta(\omega - \xi - k\omega_s) \right) \mathrm{d}\xi \\
&= \frac{1}{\triangle t} \sum_{k=-\infty}^{\infty} \int_{-\infty}^{\infty} X(\xi) \delta(\omega - \xi - k\omega_s)\, \mathrm{d}\xi && \text{reverse } \textstyle\sum \text{ and } \int \\
&= \frac{1}{\triangle t} \sum_{k=-\infty}^{\infty} X(\omega - k\omega_s) && \text{use sifting}
\end{aligned}$$

The Fourier transform $Z(\omega)$ is a **periodic** function of frequency, consisting of an infinite sum of replications of $X(\omega)$, **scaled by** $\frac{1}{\triangle t}$ and shifted by $k\omega_s$, for integer k:

$$\boxed{Z(\omega) = \frac{1}{\triangle t} \sum_{k=-\infty}^{\infty} X(\omega - k\omega_s)}, \quad \text{with } \omega_s = \frac{2\pi}{\triangle t}$$

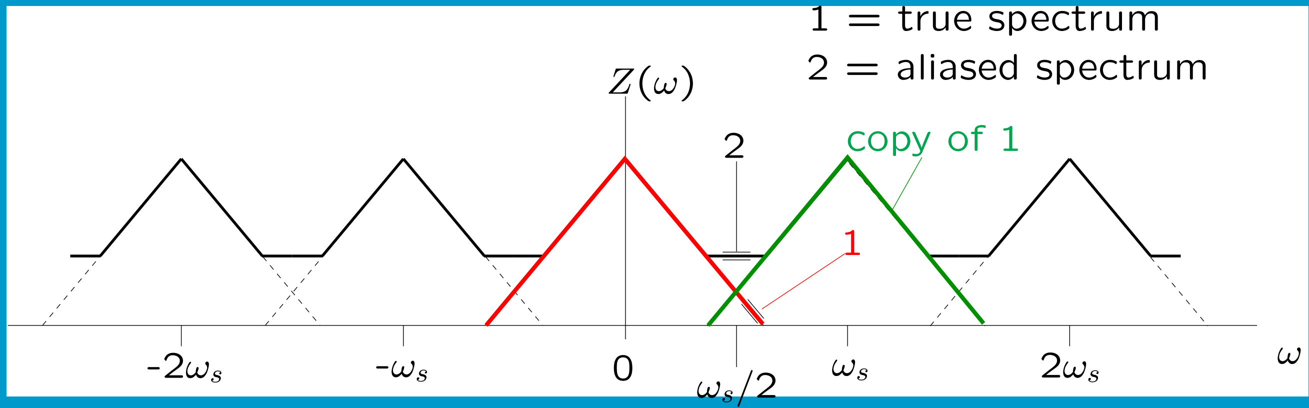
Sampling destroys information:

In the time domain, the signal $z(t)$ is only known at the samples (and is therefore referred to as $z(n\triangle t)$) and gives *no information* of what the original signal $x(t)$ does *between samples*.

In the frequency domain, the infinite ω-axis is cut into strips of $[-\omega_s/2, \omega_s/2]$ (length ω_s), which are then *superimposed* and the frequency-domain components summed to form $Z(\omega)$. This is called **aliasing**.

Aliasing

Selecting a sample frequency ω_s that is smaller than twice the highest frequency appearing in the continuous time spectrum of $X(\omega)$, will lead to an **overlapping** of the alias spectra.

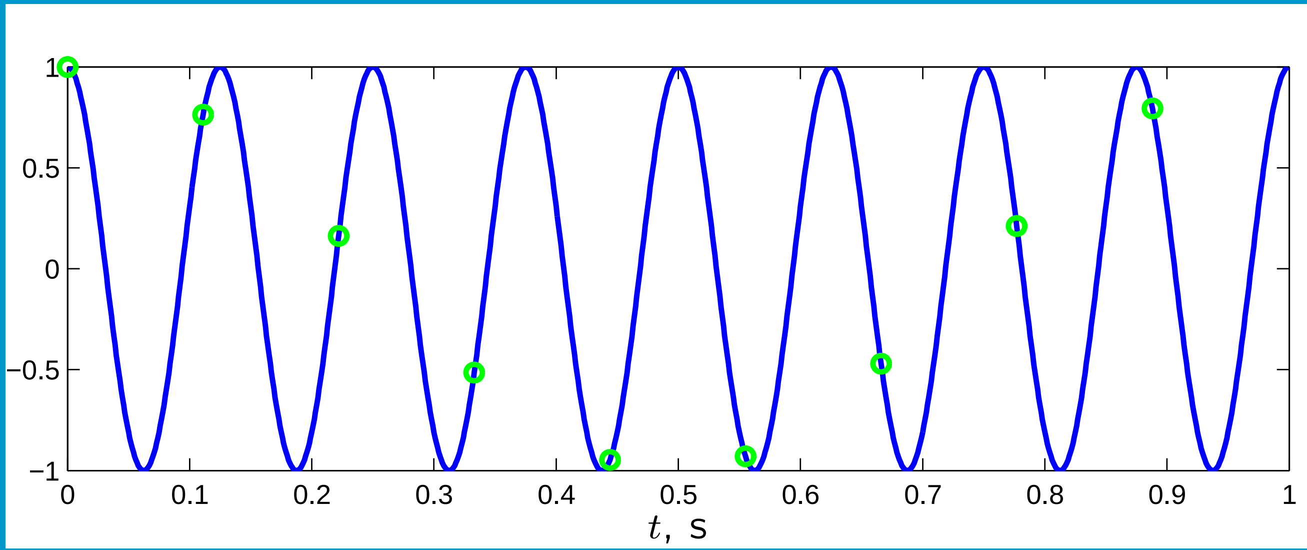


Sometimes the only way to prevent aliasing is to filter the continuous time signal before sampling such that all frequencies above $\frac{\omega_s}{2}$ are being removed **before sampling**.

This frequency is called the **Nyquist frequency** $\omega_n = \omega_s/2$.

Example aliasing

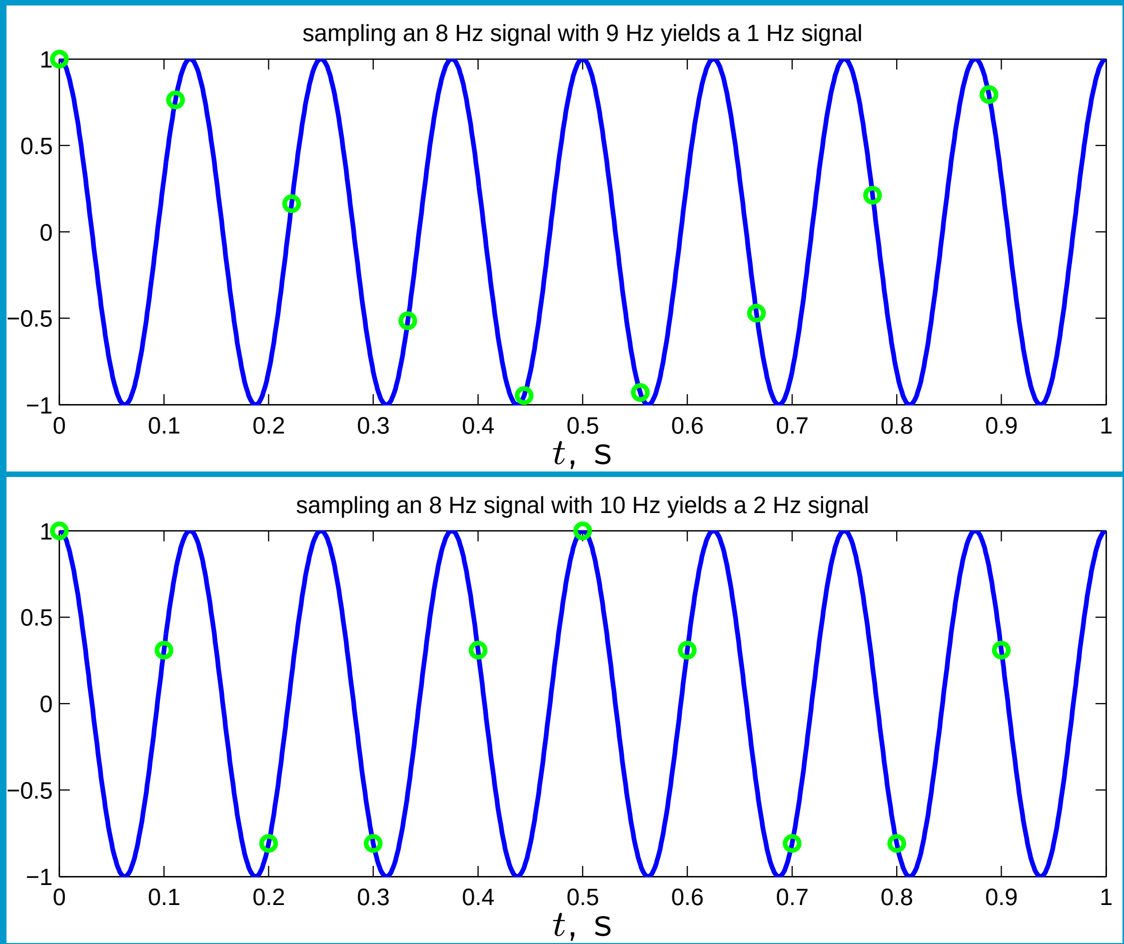
Consider a cosine signal with a frequency $f_1 = 8$ Hz. Sampling this signal with a frequency equal to or lower than 16 Hz will result in aliasing. The figure shows the result of sampling the signal with a sample frequency f_s of 9 Hz.



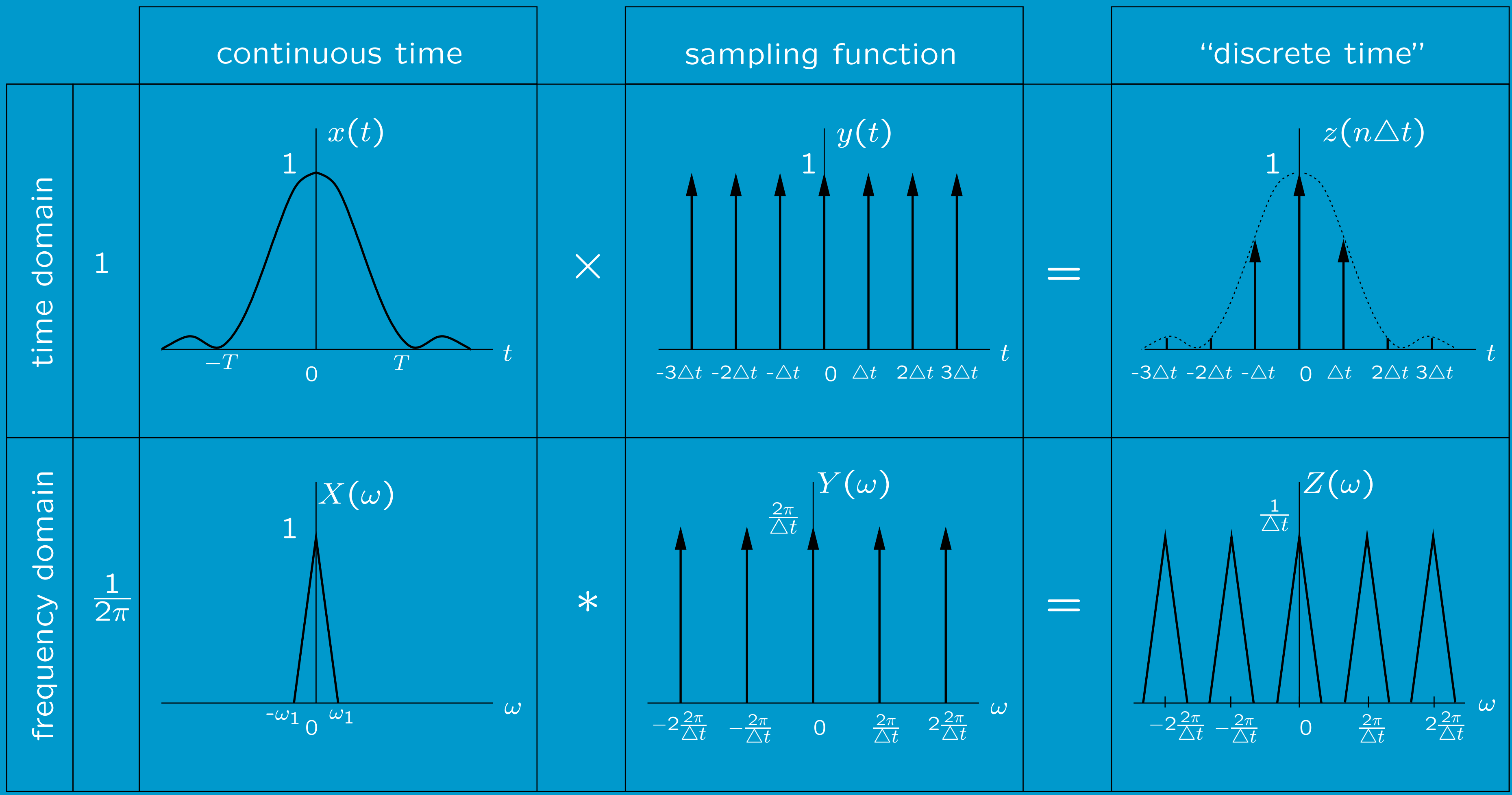
The resulting 'sampled' signal has a frequency f_2 equal to $f_s - f_1 = 1$ Hz.

Explanation: The nyquist rate of the signal is 16 Hz. The 'folding frequency' equals 4.5 Hz, so the cosine signal's Dirac pulse at 8 Hz (positive frequencies) will 'fold' around the 4.5 Hz (8-4.5 = 3.5 Hz) and come at 4.5-3.5 = 1 Hz. The same happens of course for the negative frequencies.

Example aliasing (continued)



Overview: impulse train sampling



Signal reconstruction: D $\longrightarrow$ C

The transformation of the discrete-time signal back to a continuous-time signal is called **signal reconstruction**. Ideally, we would be able to define a reconstruction filter $R(\omega)$ that passes through all signal strength within $[-\omega_s/2, \omega_s/2]$, beyond which all is zero, and would have a gain of $\triangle t$:

$$R(\omega) \;=\; \triangle t\, \sqcap \left(\frac{\omega}{\omega_s}\right), \text{ with } \omega_s = \frac{2\pi}{\triangle t}$$

The time-domain equivalent of this filter is a sinc-function:

$$r(t) \;=\; \triangle t \left(\frac{\omega_s}{2\pi}\,\mathrm{sinc}\left(\frac{t\omega_s}{2\pi}\right)\right) \;=\; \mathrm{sinc}\left(t\frac{1}{\triangle t}\right),$$

which is zero for all t equal to $m\triangle t$ (except m=0).

Multiplication of the signal frequency-domain description $Z(\omega)$ with the reconstruction filter $R(\omega)$ yields the original spectrum (assuming that no aliasing occurs):

$$\begin{aligned} R(\omega)Z(\omega) &= \triangle t \, \sqcap\left(\frac{\omega}{\omega_s}\right) \cdot \frac{1}{\triangle t} \sum_{k=-\infty}^{\infty} X(\omega - k\omega_s) \\ &= X(\omega) \end{aligned}$$

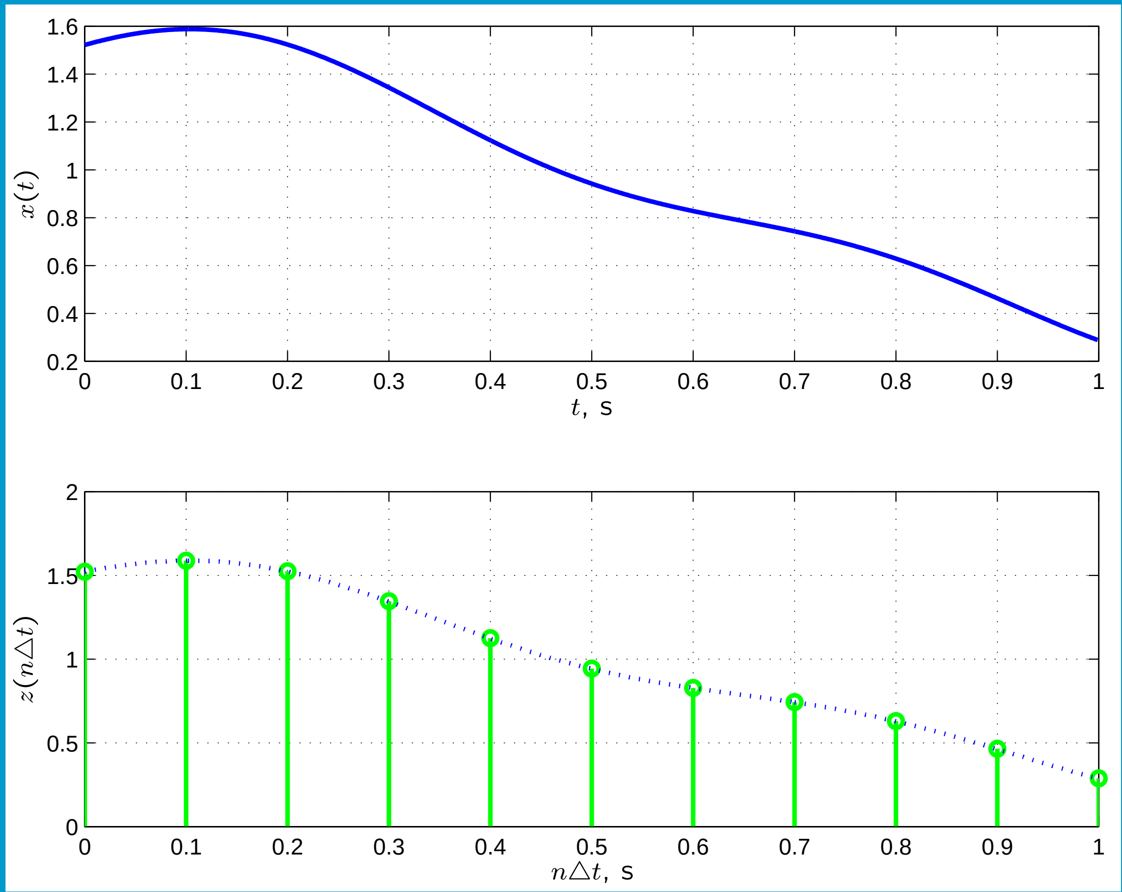
Multiplication in the frequency domain is equivalent to a convolution in the time domain. So, convolving the signal $z(n\triangle t)$ with the sinc-function $r(t)$ results in the original signal $x(t)$:

$$z(n\triangle t) * r(t) = x(t)$$

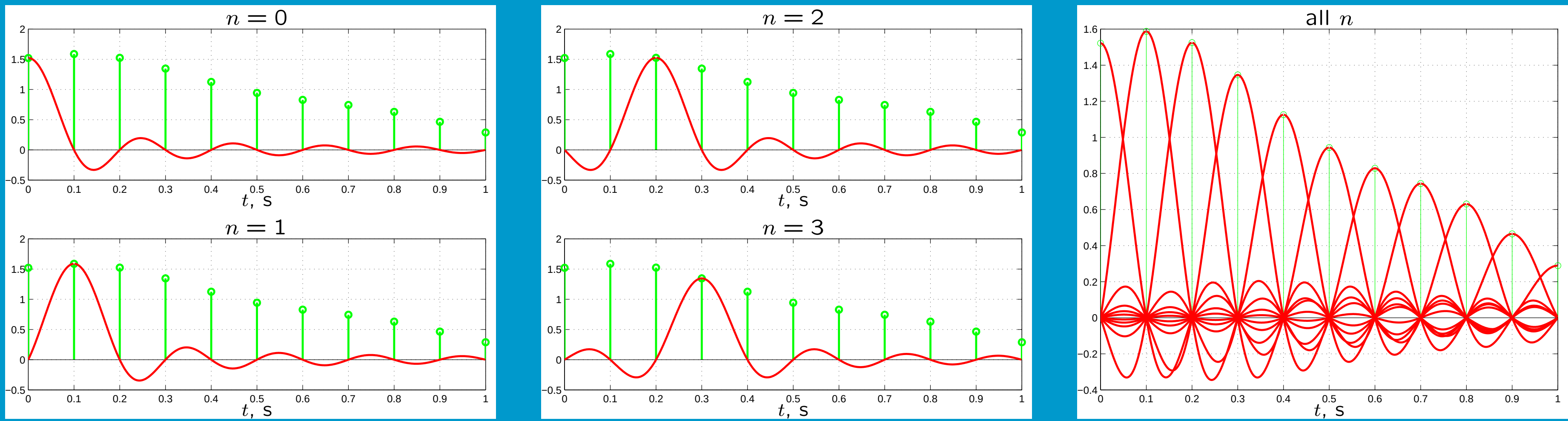
This is shown in the example below.

Example: signal reconstruction

Assume a 1 second measurement of a signal $x(t)$, and sample it with 10 Hz.

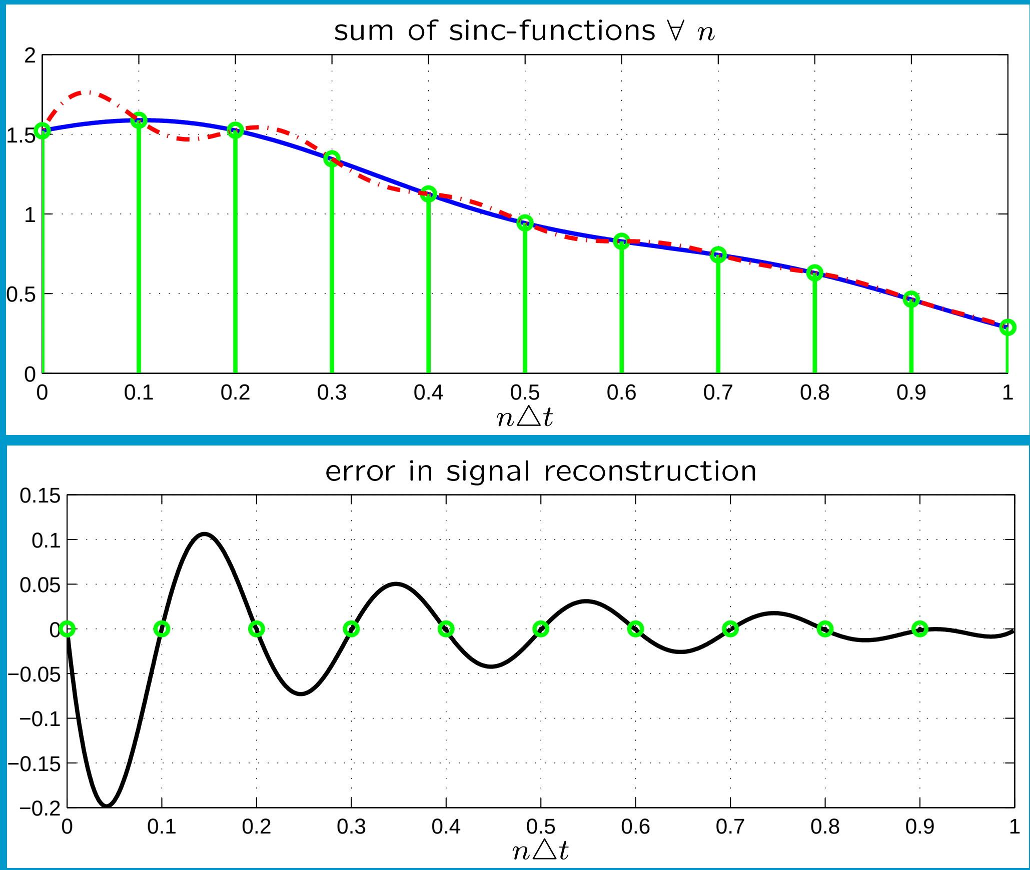


Convolve each sample with a sinc-function, $n = 0, 1, 2, 3, 4 \ldots$, then sum them all up (right).



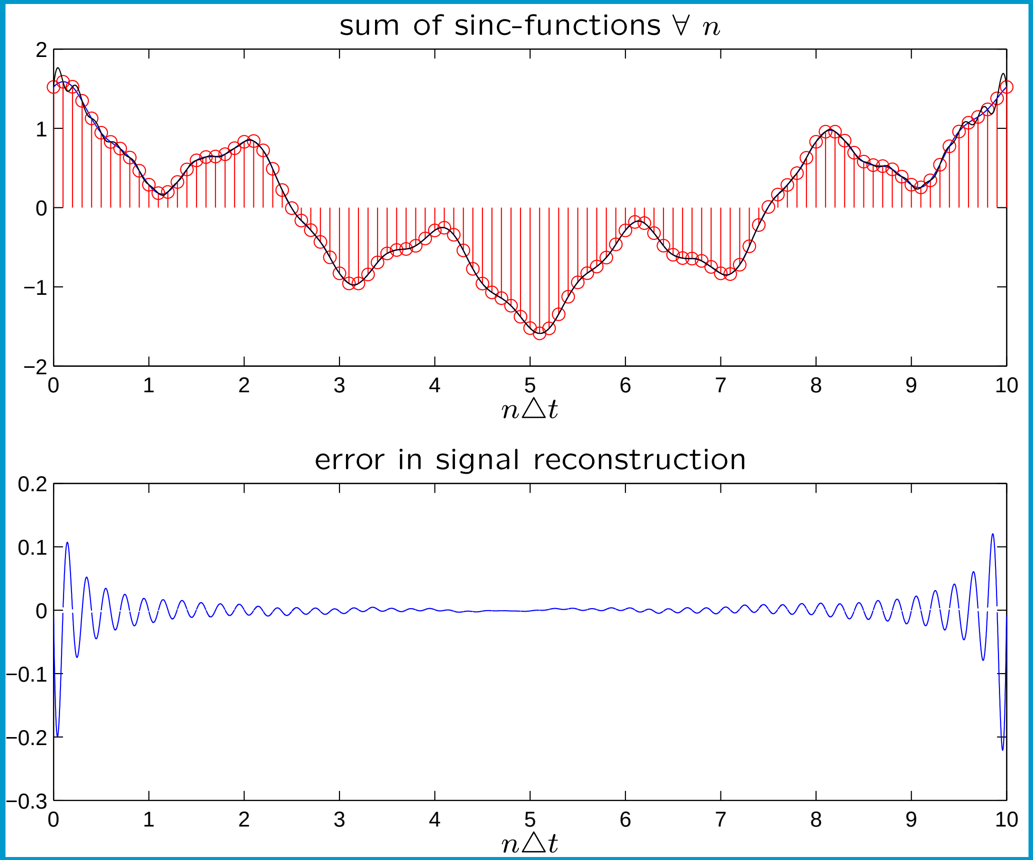
Note: for each n^{th} sample's convolution with the sinc, the fact that the sinc-function equals zero at $(n+m)\triangle t$ (except at $m = 0$, so for that sample n) means that the sinc simply scales with the value of the sample. For each sample we obtain a scaled and continuous sinc function!

The reconstruction (red line) is good but because of the limited observation not perfect. It is not equal to the measurement (blue line).



Note: the error equals zero at the samples' instants (why ??).

The longer we measure, however, the better the approximation. Here the result is shown for an observation of 10 s.



Note: the error is small in the middle, but becomes larger at the edges (why ??).

Zero-order Hold (ZOH)

The 'ideal' signal reconstruction is often not feasible, as it requires knowledge about all (past *and future*) samples.

non-causal filter, only off-line!

A common way to make a discrete-time signal more 'continuous', in an on-line fashion, is the use of **hold filters**, like the zero-order hold (ZOH) and the first-order hold (FOH).

In a ZOH, the signal between samples is kept constant between the samples. That is, when $z(n\triangle t)$ is the sample at $t = n\triangle t$, then the reconstructed or 'continuous time' signal after the ZOH, $x_r(t)$, equals $z(n\triangle t)$ for $n\triangle t \leq t < (n+1)\triangle t$.

This is equivalent to saying that the discrete-time signal $z(n\triangle t)$ is convoluted in the time domain with a block $b(t)$ with height 1 and width $\triangle t$, *shifted in time to the right* with $\triangle t/2$:

$$r(t) \;= b(t - \triangle t/2) \;=\; \sqcap\left(\frac{t - \triangle t/2}{\triangle t}\right)$$

In the frequency-domain this becomes:

$$\begin{aligned} R(\omega) &= e^{-j\omega\triangle t/2}\, B(\omega) \\ &= e^{-j\omega\triangle t/2}\, \triangle t \,\mathrm{sinc}\left(\frac{\omega\triangle t}{2\pi}\right) \end{aligned}$$

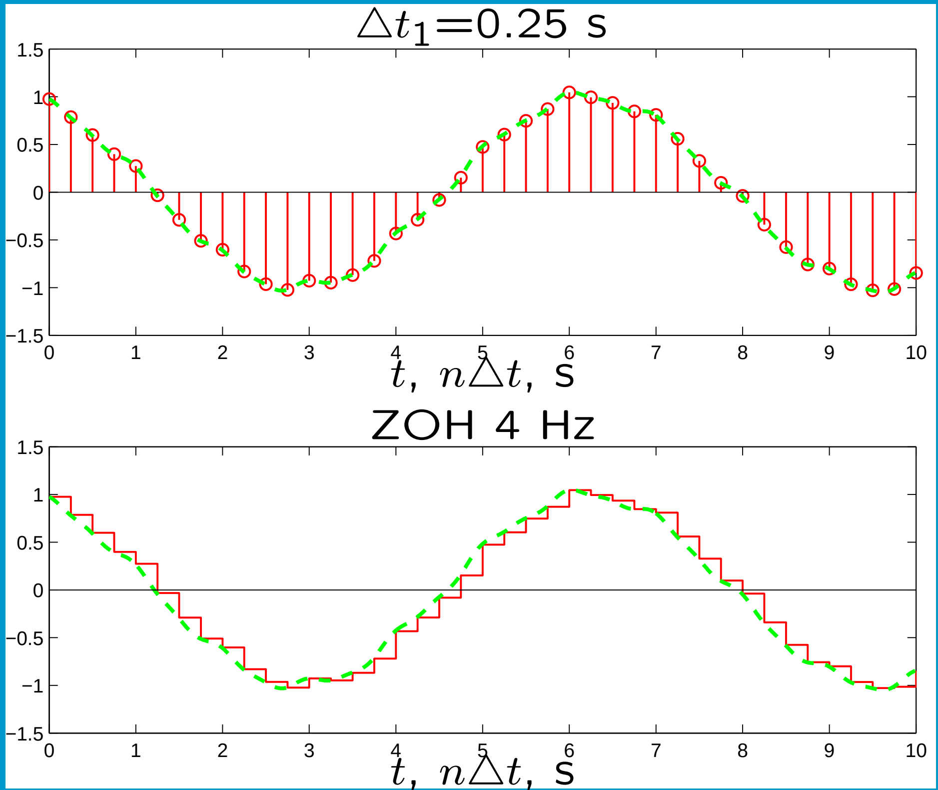
So:

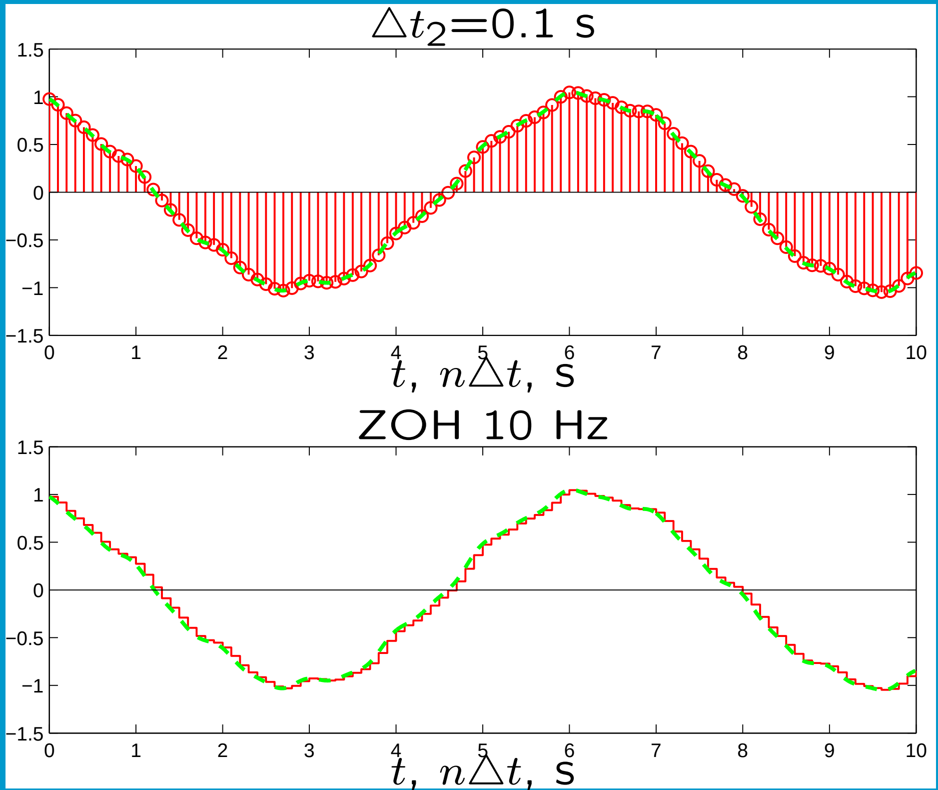
$$X_r(\omega) \;=\; R(\omega) Z(\omega)$$

Let it be clear that, in most cases, $X_r(\omega) \neq X(\omega)$, see the example.

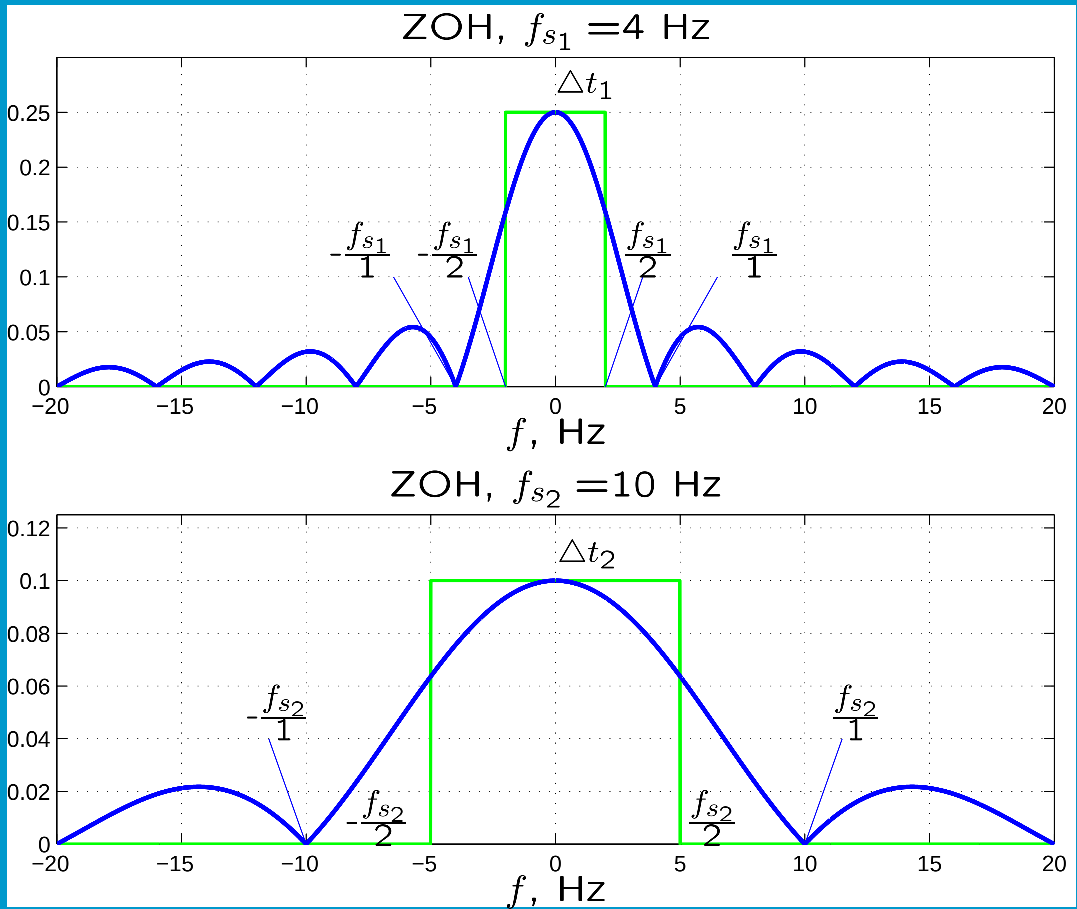
Example of a Zero-Order Hold signal reconstruction

We will be looking at using a ZOH circuit with 4 Hz (left) and 10 Hz (right). That is, what is the effect of sampling and reconstruction at a higher rate?



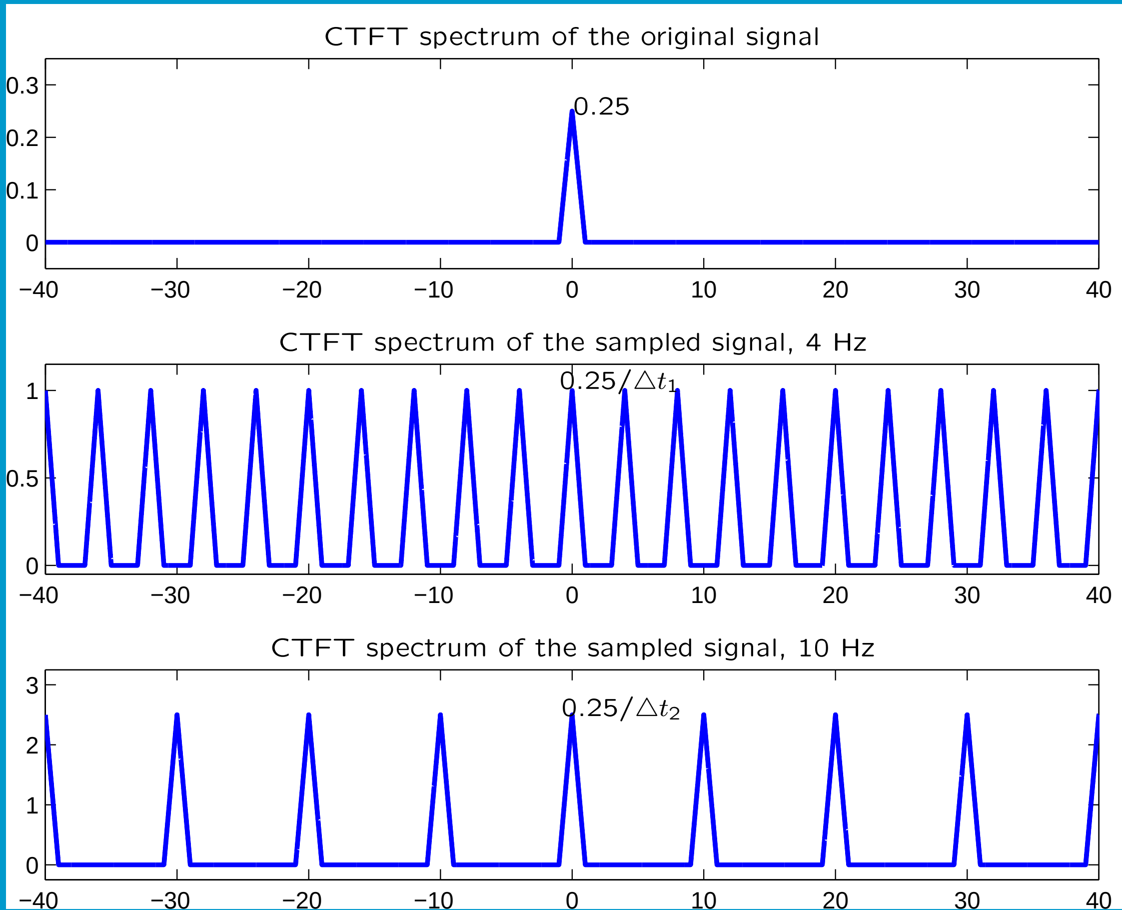


ZOH frequency response $|R_{ZOH}(\omega)|$ (in blue) for the two sample frequencies

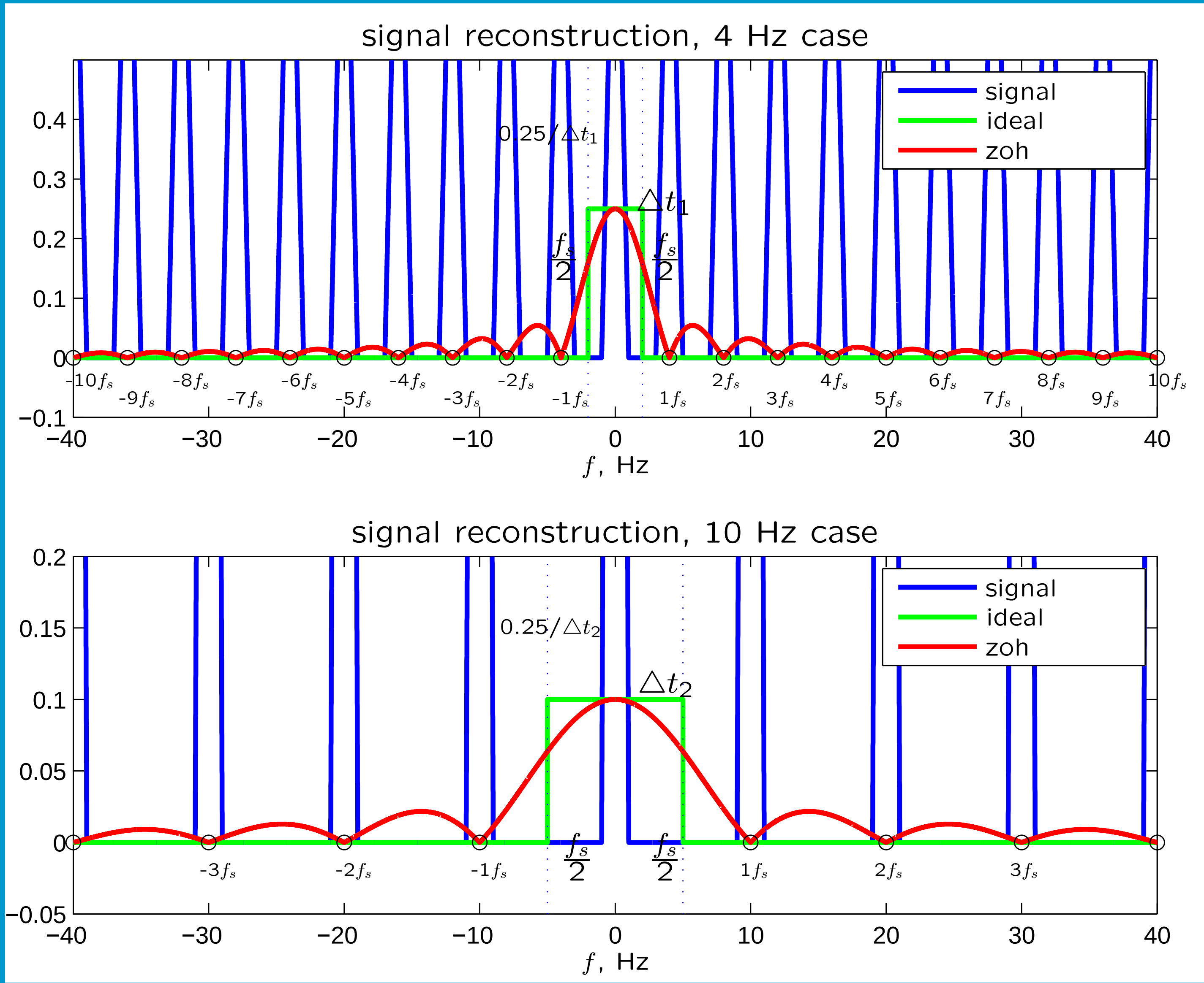


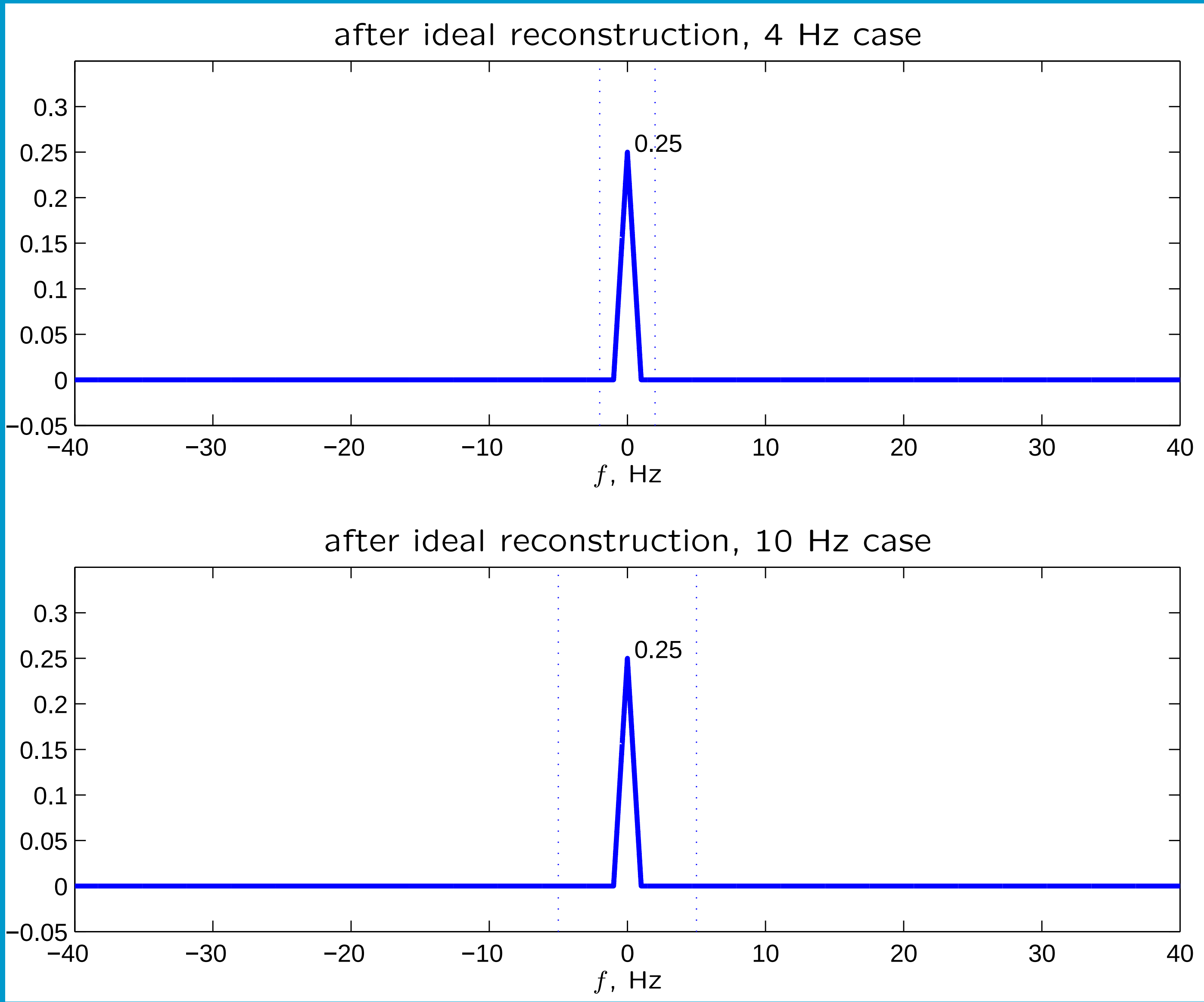
Note: the green lines show the ideal reconstruction filters (4 and 10 Hz).

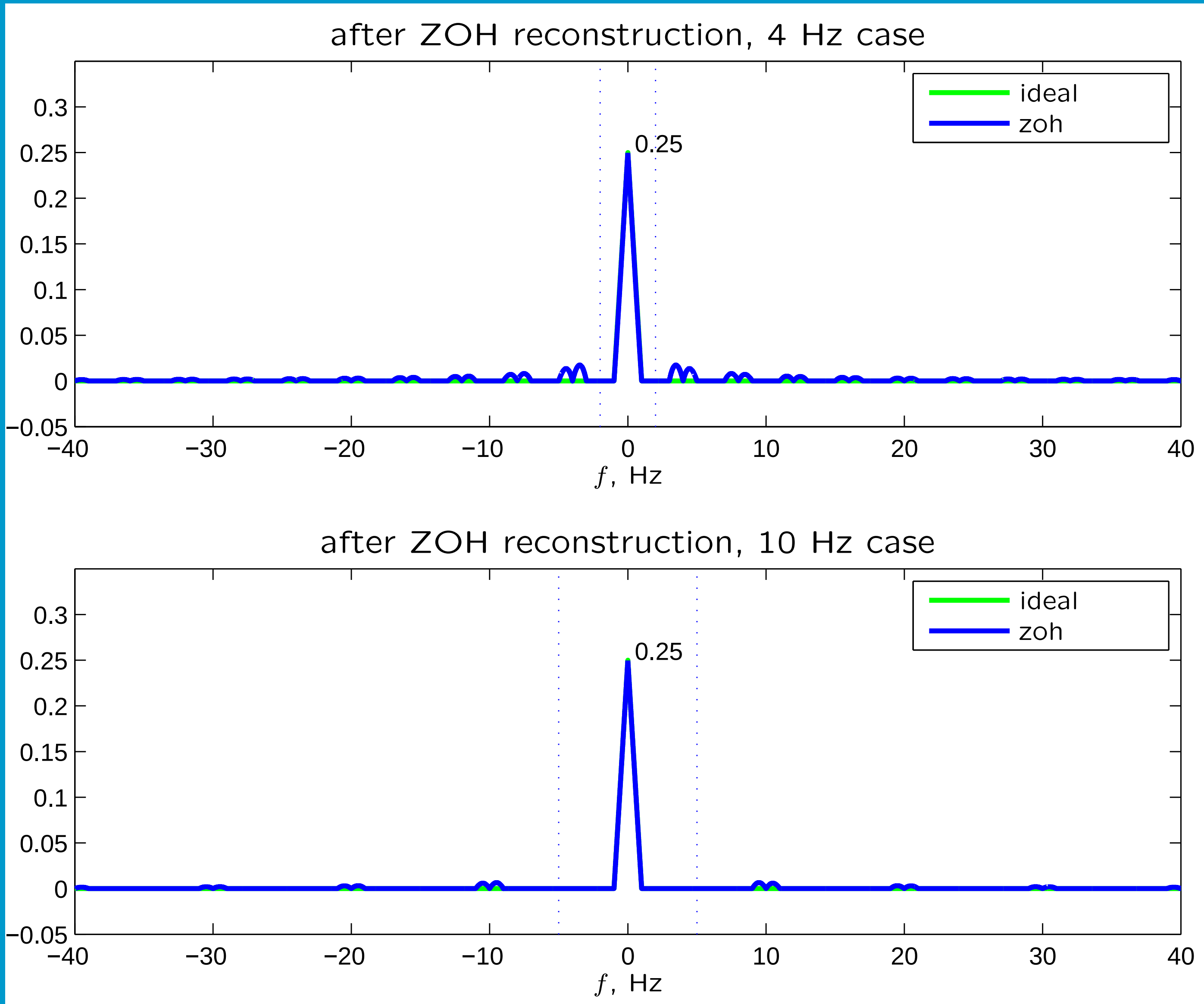
Reconstruction using ZOH versus ‘ideal’ reconstruction filter



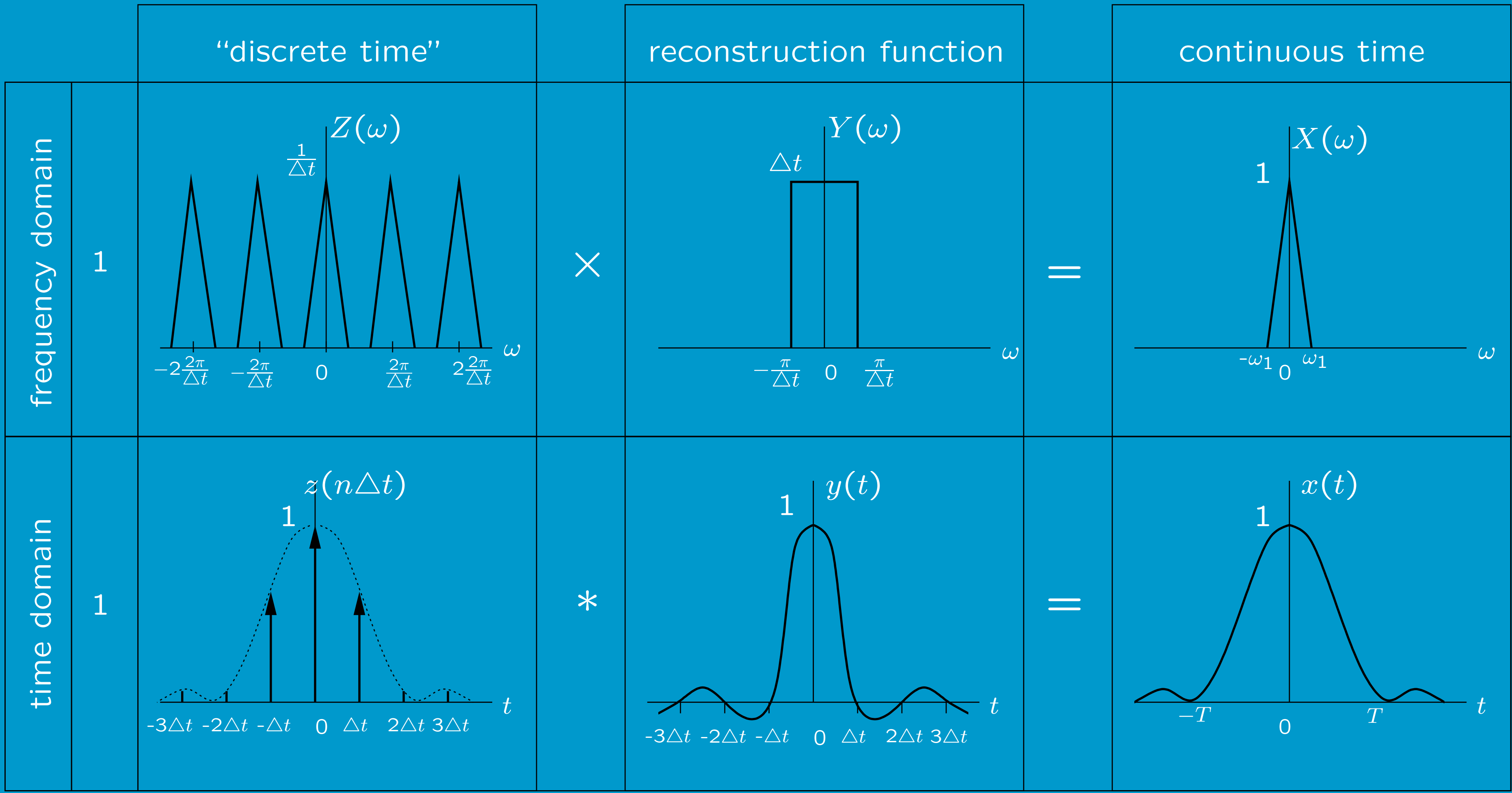
Note: we will only look at the ‘magnitude’ of the ZOH filtering.


signal reconstruction, 4 Hz case
signal
ideal
zoh
0.25/△t1
△t1
fs/2
f, Hz
signal reconstruction, 10 Hz case
0.25/△t2
△t2
f, Hz


after ideal reconstruction, 4 Hz case
0.3
0.25
0.2
0.15
0.1
0.05
0
−0.05
−40
−30
−20
−10
0
10
20
30
40
0.25
f, Hz
after ideal reconstruction, 10 Hz case
0.3
0.25
0.2
0.15
0.1
0.05
0
−0.05
−40
−30
−20
−10
0
10
20
30
40
0.25
f, Hz


after ZOH reconstruction, 4 Hz case
ideal
zoh
0.25
0.3
0.25
0.2
0.15
0.1
0.05
0
−0.05
−40
−30
−20
−10
0
10
20
30
40
f, Hz
after ZOH reconstruction, 10 Hz case
ideal
zoh
0.25
0.3
0.25
0.2
0.15
0.1
0.05
0
−0.05
−40
−30
−20
−10
0
10
20
30
40
f, Hz

Overview: signal reconstruction



Sampling Theorem (Shannon)

Let $x(t)$ be a bandlimited signal with $X(\omega) = 0$ for $|\omega| > \omega_M$. Then $x(t)$ is uniquely determined by its samples $x(n\triangle t)$, $n = 0, \pm 1, \pm 2, \ldots$ if:

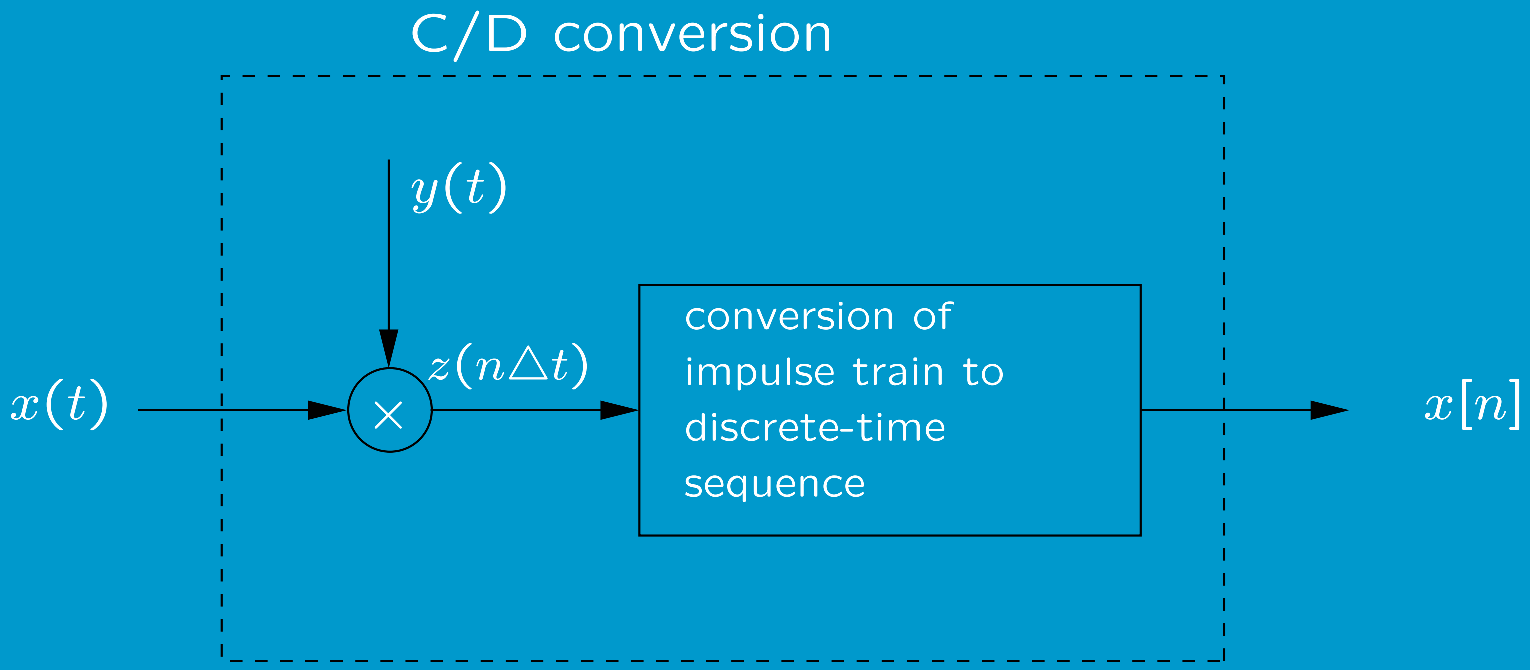
$$\boxed{\omega_s \; > \; 2\omega_M}, \qquad \text{Nyquist rate } 2\omega_M$$

where $\omega_s \; = \; \frac{2\pi}{\triangle t}$.

Given these samples, we can reconstruct $x(t)$ by generating a periodic impulse train in which successive impulses have weights that are equal to the successive sample values. This impulse train is then processed through an ideal low-pass filter with gain $\triangle t$ and cutoff frequency greater than ω_M and less than $(\omega_s - \omega_M)$. The resulting reconstructed signal will exactly equal $x(t)$.

Conversion of the CT impulse train to discrete time series

Did we fully cover the relationship between the continuous time signal $x(t)$ and the discrete-time sequence $x[n]$? The answer is no, we still have to map the impulse train $z(n\triangle t)$ to the sequence $x[n]$.



We have seen that the CTFT of $z(n\triangle t)$ equals:

$$Z(\omega) \;=\; \frac{1}{\triangle t}\sum_{k=-\infty}^{\infty} X(\omega - k\omega_s),$$

where $z(n\triangle t)$ can be described as an impulse train with the weights of the impulses scaled to the values of the samples of $x(t)$ at discrete times $n\triangle t$:

$$z(n\triangle t) \;=\; \sum_{n=-\infty}^{\infty} x(t)\delta(t - n\triangle t)$$

Directly CT Fourier transforming this signal yields:

$$Z(\omega) \;=\; \mathcal{F}\left\{z(n\triangle t)\right\} \;=\; \sum_{n=-\infty}^{\infty} x(n\triangle t)\, e^{-j\,\omega\triangle t\, n} \qquad (*1)$$

The discrete-time Fourier transform (DTFT) of $x[n]$ is defined as:

$$\boxed{X(\Omega) \;=\; \sum_{n=-\infty}^{\infty} x[n]\, e^{-j\,\Omega\,n}},$$

Ω is the 'discrete frequency'

which equals (because $x[n]$ equals $x(n\triangle t)$):

$$X(\Omega) \;=\; \sum_{n=-\infty}^{\infty} x(n\triangle t)\, e^{-j\,\Omega\,n} \qquad (*2)$$

Comparing equations (*1) and (*2), we see that $X(\Omega)$ (the DTFT of the discrete-time sequence $x[n]$) and $Z(\omega)$ (the CTFT of the continuous-time impulse train $z(n\triangle t)$) are related through:

$$\boxed{X(\Omega) \;=\; Z(\omega = \frac{\Omega}{\triangle t})}$$

substitute $\frac{\Omega}{\triangle t}$ for ω

The discrete-time frequency Ω equals $\omega \triangle t$. It is periodic with 2π, i.e.: $\Omega + 2\pi \;=\; \Omega.$

digital frequency is discussed later

It is as if for the discrete-time equivalent of the CTFT, the CT frequency ω is scaled with $\triangle t$, and therefore the CTFT *time* t is normalized with $\frac{1}{\triangle t}$, mapping the time axis t to a DT series.

That is, the sequence:

$$t \in \{..., \; -3\triangle t, \; -2\triangle t, \; -1\triangle t, \; 0\triangle t, \; 1\triangle t, \; 2\triangle t, \; 3\triangle t, ...\}$$

CTFT yields $Z(\omega)$

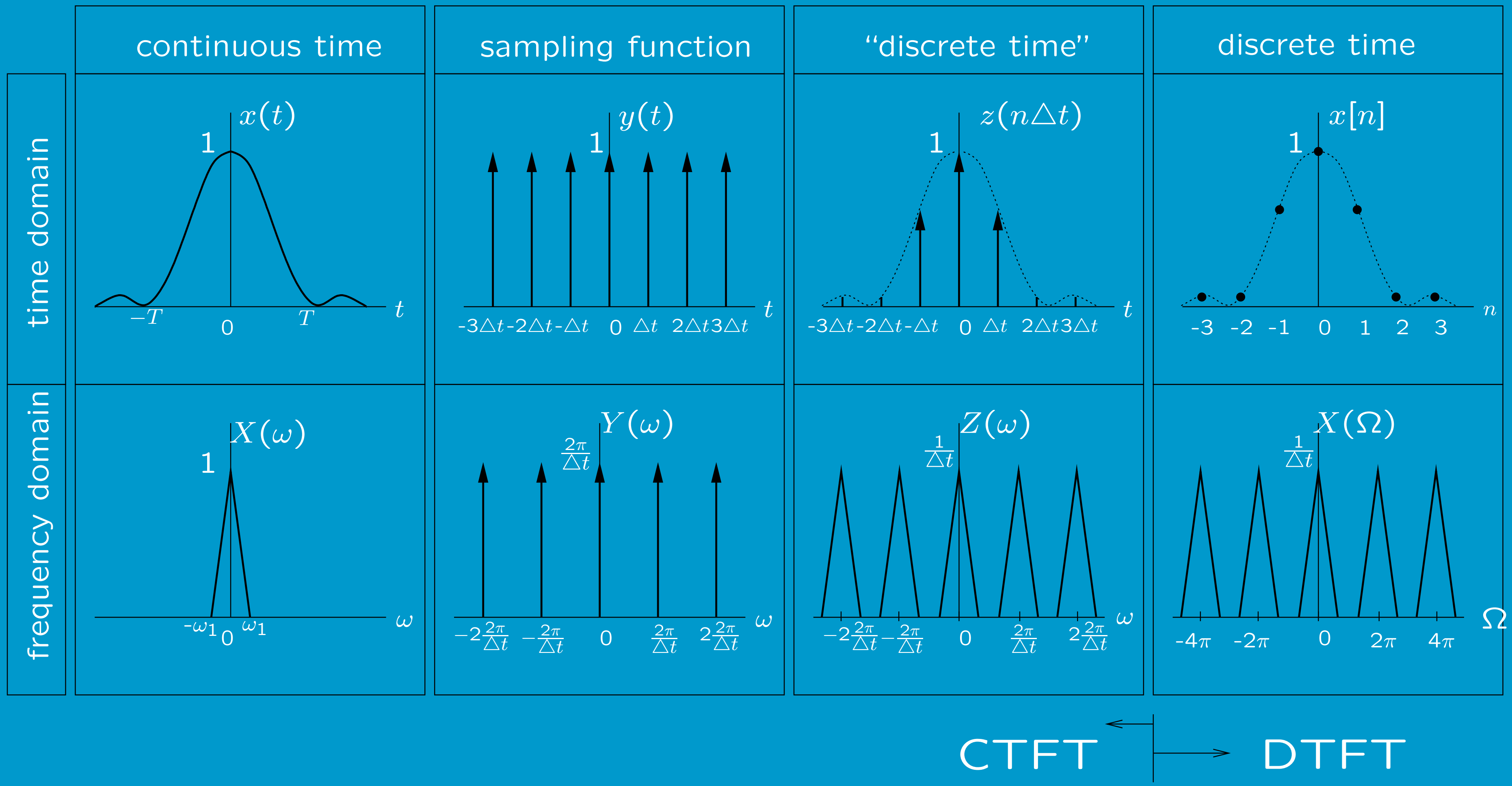
is mapped to:

we are 'normalizing' the time axis here!

$$n \in \{..., \; -3, \; -2, \; -1, \; 0, \; 1, \; 2, \; 3, ...\}$$

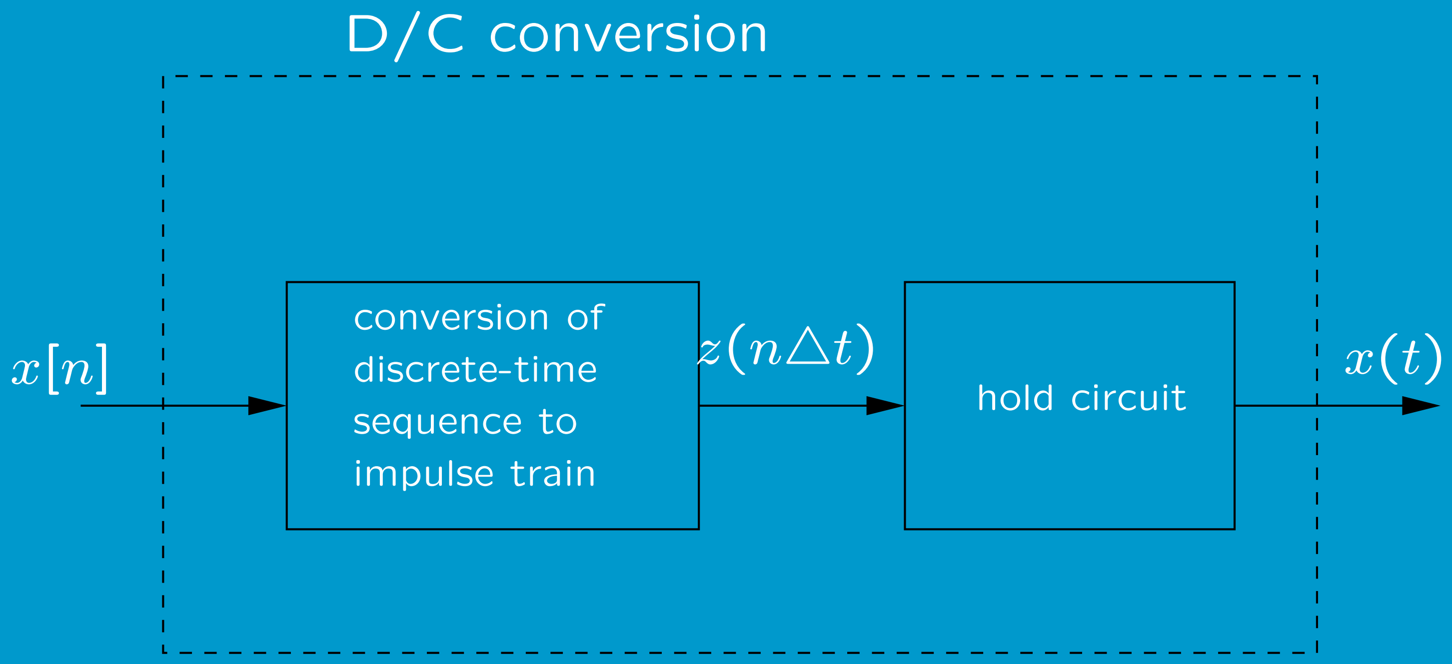
DTFT yields $X(\Omega) = Z\left(\frac{\Omega}{\triangle t}\right)$

Sampling complete

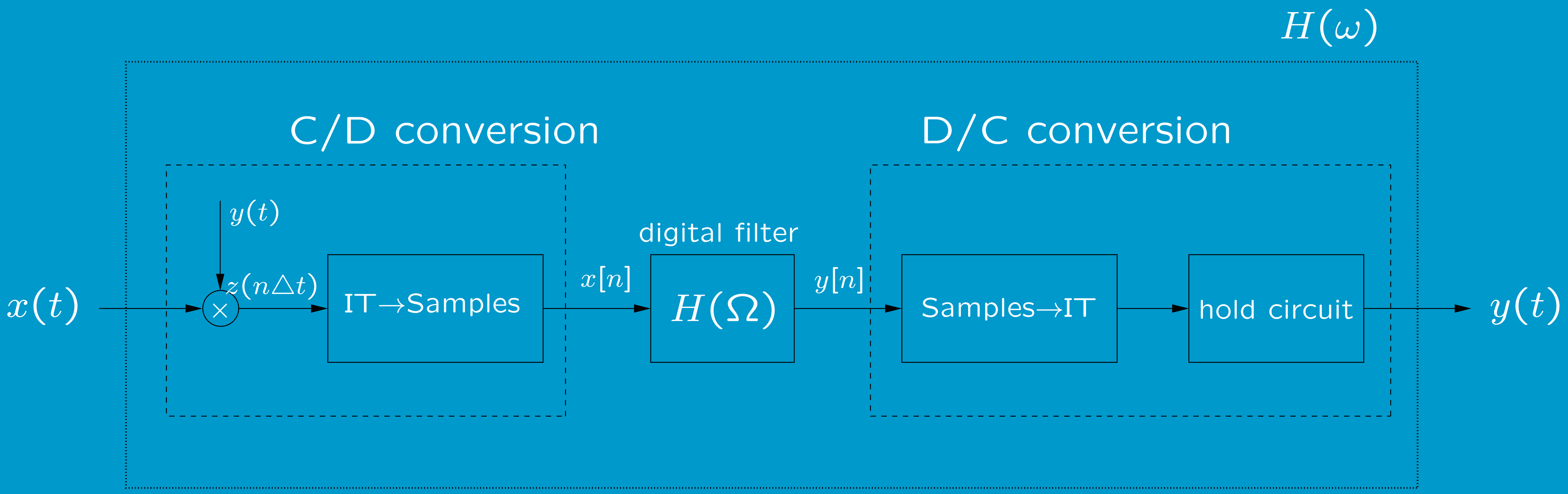


Conversion of the discrete time series to continuous signal

A similar, but reverse, operation is followed to convert the discrete-time sequence $x[n]$ to the continuous-time signal $x(t)$.

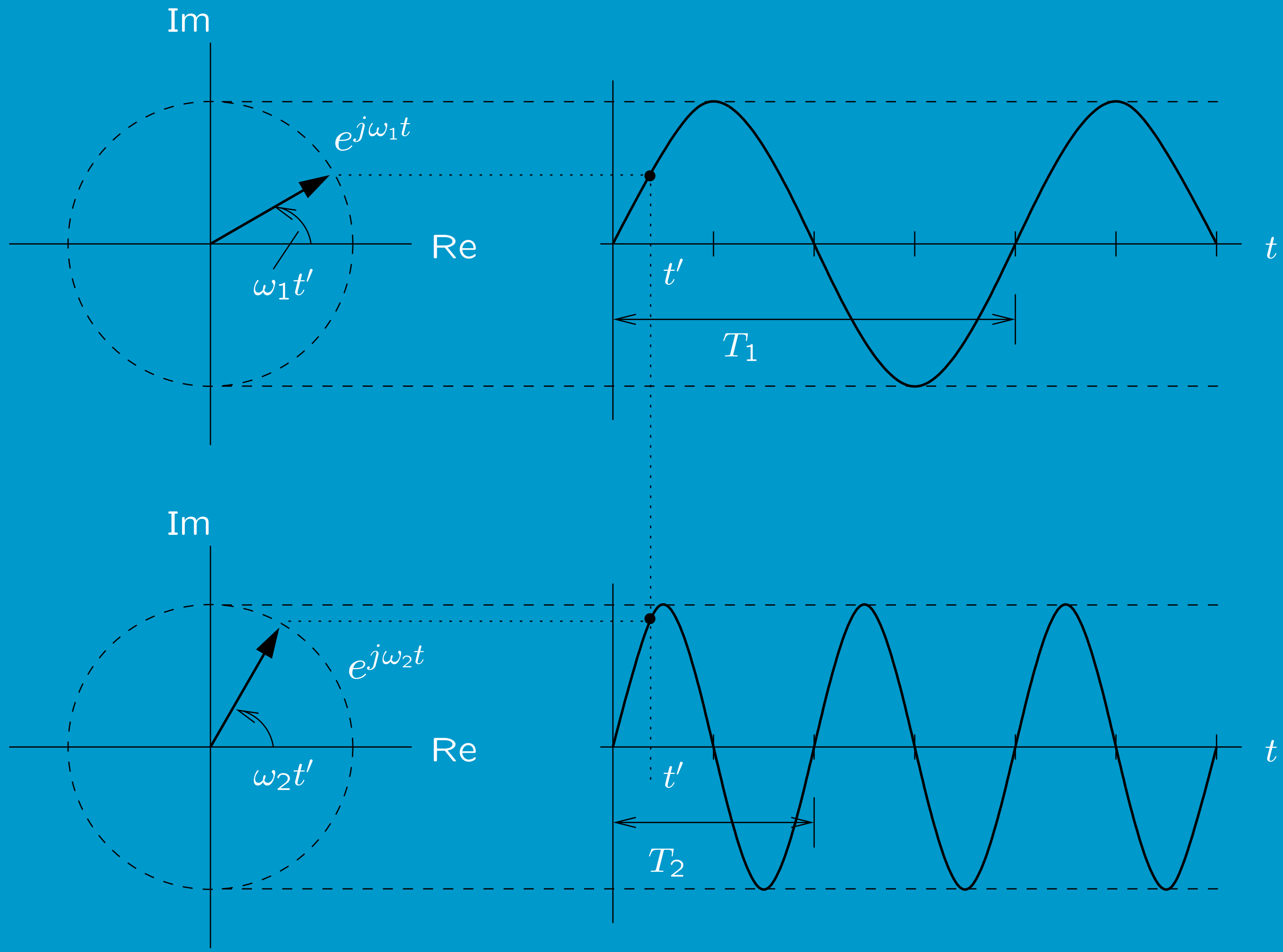


Conclusion: DT equivalent of CT system



The discrete-time system $H(\Omega)$ computes the output sequence $y[n]$ as a function of the input sequence $x[n]$. When done properly, the DT system closely resembles the CT system $H(\omega)$.

Frequency in continuous time (CT)

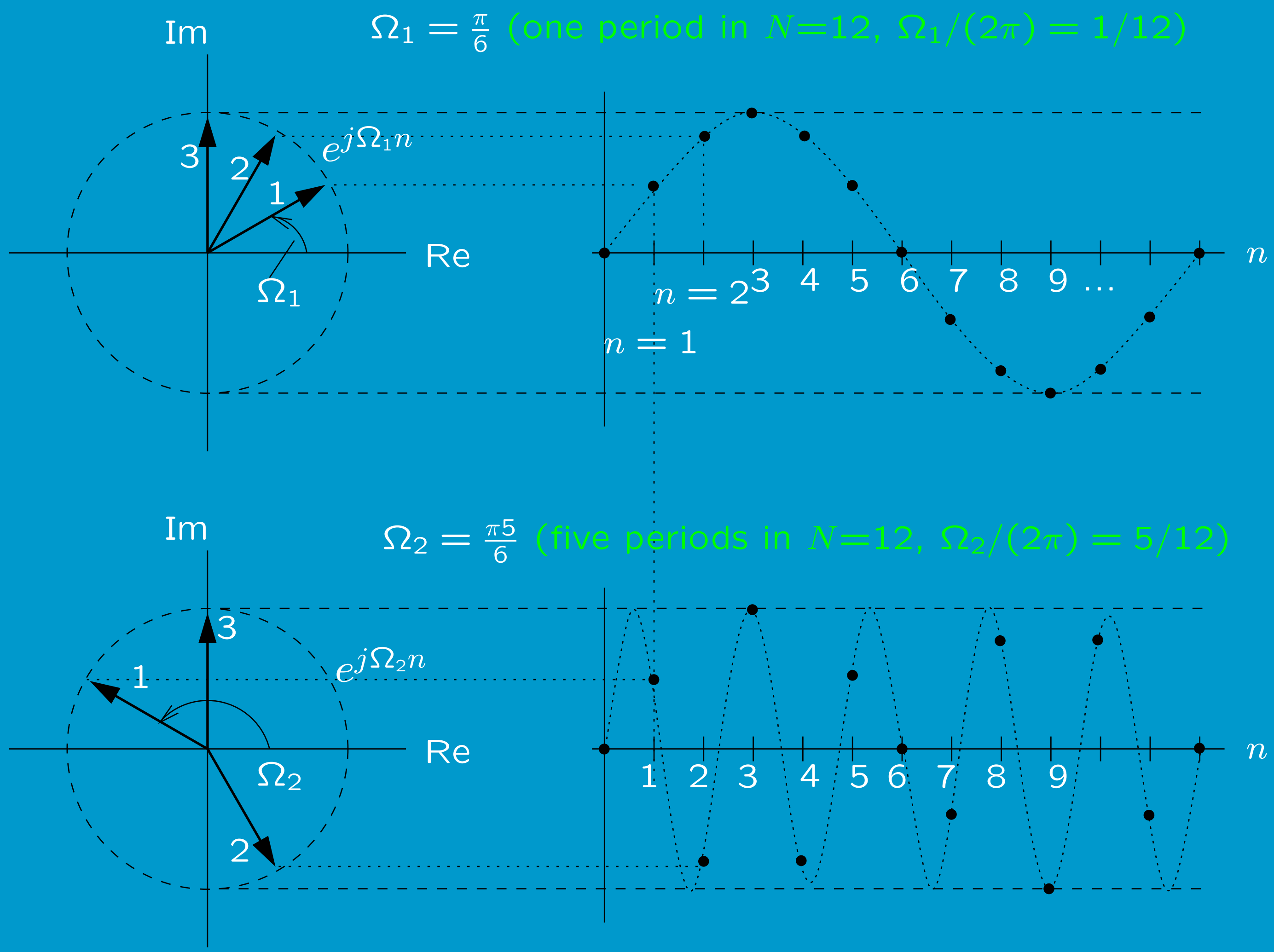


In CT the complex exponentials are **periodic** for all ω

In CT, when ω increases, the higher the rate of the fluctuations

Here $\omega_2 = 2\omega_1$

Frequency in discrete time (DT)



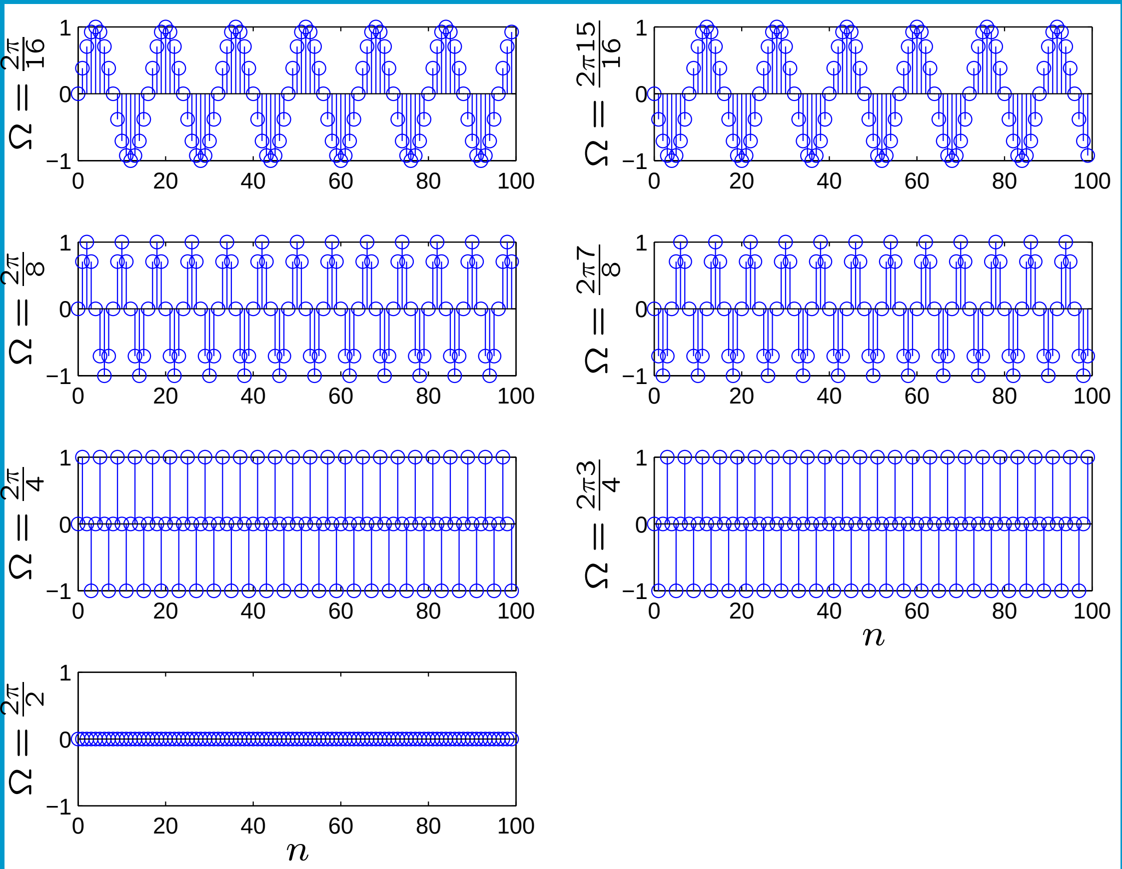
In DT the complex exponentials are only periodic when $\frac{\Omega}{2\pi}$ is a rational number:

$e^{j\Omega n} = e^{j\Omega(n+N)}$
with $e^{j\Omega N} = 1$
$\Rightarrow \quad \Omega N = m2\pi$

In DT, when Ω increases from 0 to π, the higher the rate of the fluctuations.

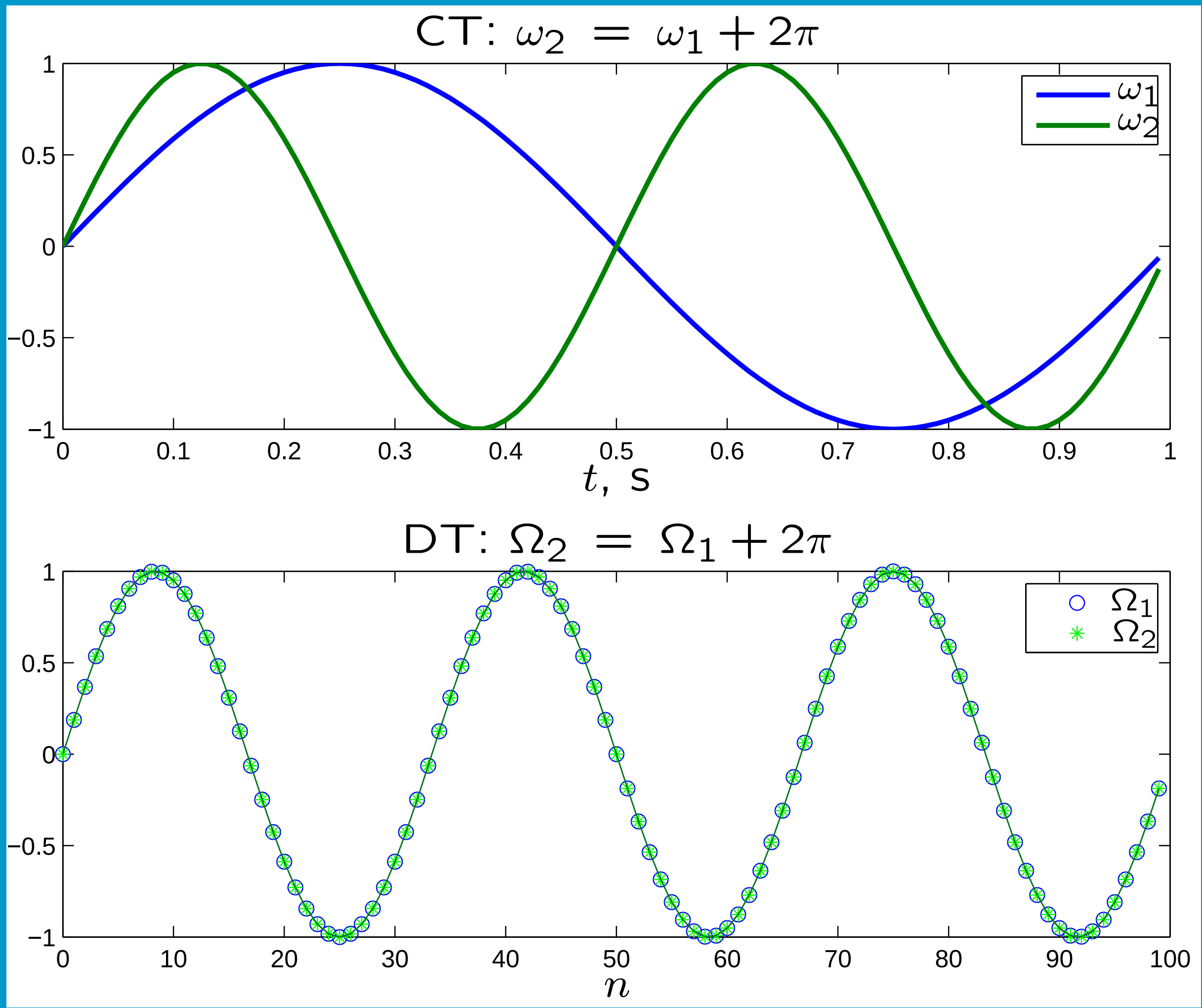
The fluctuations become less when Ω increases from π to 2π.

NOTE: we look at the SINE component, like in the previous sheet.



From Ω =0 to π, oscillations go faster, then from π to 2π the oscillations go slower. In fact, the left column represents the 'positive frequencies', the right column the 'negative frequencies'.

In DT, when $\Omega_2 = \Omega_1 + 2\pi$ the signals are exactly the same!!



Discrete-time Fourier transform (DTFT)

continuous time	discrete time
CTFT	DTFT
$X(\omega) = \mathcal{F}\{x(t)\} = \int\limits_{-\infty}^{\infty} x(t)e^{-j\omega t}\,\mathrm{d}t$ $x(t) = \mathcal{F}^{-1}\{X(\omega)\} = \frac{1}{2\pi}\int\limits_{-\infty}^{\infty} X(\omega)e^{j\omega t}\,\mathrm{d}\omega$	$X(\Omega) = \mathcal{F}\{x[n]\} = \sum\limits_{n=-\infty}^{\infty} x[n]e^{-j\Omega n}$ $x[n] = \mathcal{F}^{-1}\{X(\Omega)\} = \frac{1}{2\pi}\int\limits_{2\pi} X(\Omega)e^{j\Omega n}\,\mathrm{d}\Omega$
Laplace transform	z-transform
$X(s) = \mathcal{L}\{x(t)\} = \int\limits_{-\infty}^{\infty} x(t)e^{-st}\,\mathrm{d}t$ $x(t) = \mathcal{L}^{-1}\{X(s)\} = \frac{1}{2\pi j}\int\limits_{\sigma-j\infty}^{\sigma+j\infty} X(s)e^{st}\,\mathrm{d}s$	$X(z) = \mathcal{Z}\{x[n]\} = \sum\limits_{n=-\infty}^{\infty} x[n]z^{-n}$ $x[n] = \mathcal{F}^{-1}\{X(z)\} = \frac{1}{2\pi j}\oint X(z)z^{n-1}\,\mathrm{d}z$

The DTFT relates to the z-transform in a similar way as the CTFT relates to the Laplace transform. The z-transform is more general, the DTFT only concerns the values of z on the unit circle, i.e., the harmonics.

The DTFT is valid for aperiodic signals (note that it is *continuous* in Ω) and is of great theoretical value. In practical applications, however, it is rarely used, because it implies infinite observation times.

The DTFT is accompanied by its (simpler and less general) counterpart, the discrete Fourier transform DFT, which supported by a very efficient algorithm, the Fast-Fourier transform FFT.

It will be shown that the DFT can be considered as the equivalent, in discrete time, of the continuous time Fourier Series.

The discrete Fourier transform (DFT)

Assume that we have sampled a continuous time signal $x(t)$ with a sufficiently high sampling frequency ($\frac{1}{\triangle t}$), yielding N samples that are equally spaced at $\triangle t$ seconds.

Then, the DFT (which is NOT the discrete-time FT!!!) is defined as:

$$\boxed{X[k] = \sum_{n=0}^{N-1} x[n] e^{-jk\frac{2\pi}{N}n}}, \quad k\frac{2\pi}{N}n$$

N real numbers $\Rightarrow$ N complex numbers

More information??

and its inverse as:

$$\boxed{x[n] = \frac{1}{N} \sum_{k=0}^{N-1} X[k] e^{jn\frac{2\pi}{N}k}}, \quad n\frac{2\pi}{N}k$$

No: $X[k]$ repeats itself, for the positive and negative frequencies!

$X[k]$ for $k = 0, \ldots, \frac{N}{2} - 1$ has all the information

or, with $W = e^{j\frac{2\pi}{N}}$:

$$\boxed{X[k] = \sum_{n=0}^{N-1} x[n] W^{-kn}}, \quad \text{and} \quad \boxed{x[n] = \frac{1}{N} \sum_{n=0}^{N-1} X[k] W^{kn}}$$

The DFT bears a great resemblance with the CT Fourier Series:

CTFS	DFT
$c_k = \frac{1}{T}\int\limits_{t_0}^{t_0+T} x(t)e^{-jk\frac{2\pi}{T}t}\mathrm{d}t$	$X[k] = \sum\limits_{n=0}^{N-1} x[n]e^{-jk\frac{2\pi}{N}n}$
$x(t) = \sum\limits_{k=-\infty}^{\infty} c_k e^{jk\frac{2\pi}{T}t}$	$x[n] = \frac{1}{N}\sum\limits_{k=0}^{N-1} X[k]e^{jn\frac{2\pi}{N}k}$

In the CTFS, $\frac{2\pi}{T}$ is the fundamental frequency, the frequency of the sine that fits exactly within one period T.

In the DFT, $\frac{2\pi}{N}$ is the fundamental frequency, the smallest digital frequency that you can have with N samples. Again, it is the frequency of the sine that fits exactly one period in the N-sample observation.

Notes:

1 In the CT case, T stands for one cycle of a periodic function with period T. In the DT case, N stands for an observation consisting of N samples.

2 The digital frequency Ω equals $\omega\triangle t$, so $\omega = \Omega/\triangle t$. The smallest change in discrete frequency that we can 'look at' is $2\pi/N$, so the smallest change in continuous radial frequency equals $2\pi/(N\triangle t)$ which equals $2\pi/T$, with $T = N\triangle t$ the length of the observation.

3 The smallest change in natural frequency equals $1/(N\triangle t)$ which equals f_s/N. It is known as the **"frequency resolution"** of the DFT.

4 There are other definitions of the DFT as well, e.g.:
$X[k] = \triangle t \sum_{n=0}^{N-1} x[n]e^{-j\frac{2\pi}{N}kn}$ and $x[n] = \frac{f_s}{N} \sum_{k=0}^{N-1} X[k]e^{+j\frac{2\pi}{N}nk} = \frac{1}{T} \sum_{k=0}^{N-1} X[k]e^{+j\frac{2\pi}{N}nk}$.

5 In the definition in Note #4, the "time resolution" $\triangle t$ and the "frequency resolution" f_s/N become more apparent.

6 When N increases, so does the "frequency resolution", f_s/N.

7 When f_s increases, and the measurement time T remains the same, the resolution in time improves, but the resolution in frequency remains the same.

8 The DFT (FFT) definition in Matlab is the same as used in the lecture slides.

9 The DFT is periodic in N: $X[k] = X[k+N]$. Hence, the discrete-time sequence $x[n]$ is "modelled" by the DFT as a realization of a single cycle of a *periodic* time series.

10 Note that $X[-k] = X[N-k] = X^*[k]$, the complex conjugate.

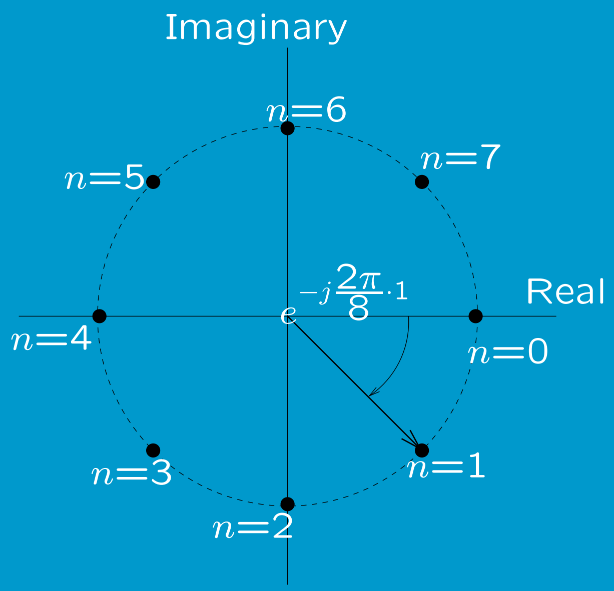
The DFT is periodic, and therefore:

$$\boxed{\begin{array}{rcrcrcr} W^{m+N} & = & W^m & ; & W^{m-N} & = & W^m \\ W^{m+N/2} & = & -W^m & ; & W^{m-N/2} & = & -W^m \end{array}}$$

And also:

$$\boxed{\begin{array}{llclcl} W^{N/2} & = W^{-N/2} & = & -1 & & \\ W^0 & = W^N & = & W^{-N} & = & 1 \end{array}}$$

Example: the DFT of a discrete series of 8 points (N=8)



$$\boxed{X[k] = \sum_{n=0}^{N-1} x[n] e^{-jk\left(\frac{2\pi}{N}\right)n}, \text{ for } k = 0, 1, \ldots, N-1}$$

$$X[0] = \sum_{n=0}^{7} x[n] e^{-j0\frac{2\pi}{8}n} = x[0] + x[1] + x[2] + x[3] + x[4] + x[5] + x[6] + x[7]$$ zero frequency

$$X[1] = \sum_{n=0}^{7} x[n] e^{-j1\frac{2\pi}{8}n}$$ one cycle in N=8

$$X[2] = \sum_{n=0}^{7} x[n] e^{-j2\frac{2\pi}{8}n} = x[0] - jx[1] - x[2] + jx[3] + x[4] - jx[5] - x[6] + jx[7]$$ two cycles

$$X[3] = \sum_{n=0}^{7} x[n] e^{-j3\frac{2\pi}{8}n}$$ three cycles

$$X[4] = \sum_{n=0}^{7} x[n]e^{-j4\frac{2\pi}{8}n} = x[0] - x[1] + x[2] - x[3] + x[4] - x[5] + x[6] - x[7]$$
$$= (x[0] + x[2] + x[4] + x[6]) - (x[1] + x[3] + x[5] + x[7])$$

$$X[5] = \sum_{n=0}^{7} x[n]e^{-j5\frac{2\pi}{8}n}$$

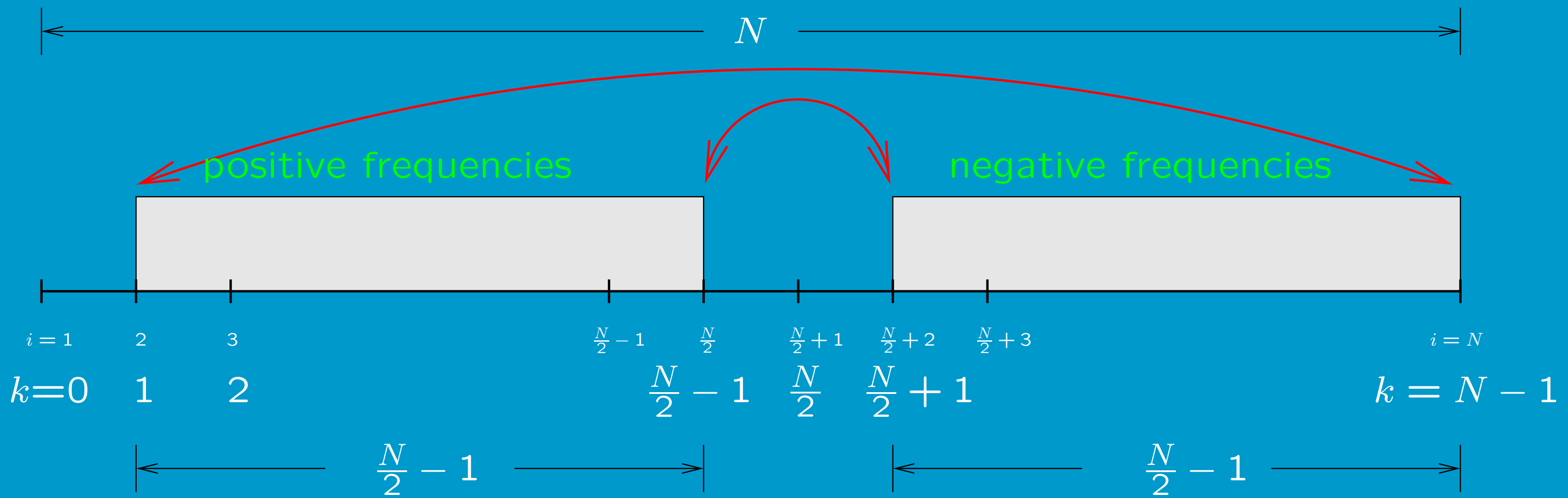
$$X[6] = \sum_{n=0}^{7} x[n]e^{-j6\frac{2\pi}{8}n} = x[0] + jx[1] - x[2] - jx[3] + x[4] + jx[5] - x[6] - jx[7]$$

$$X[7] = \sum_{n=0}^{7} x[n]e^{-j7\frac{2\pi}{8}n}$$

Hence: we get a sequence of 8 complex numbers, with:

i	k	*pos. f*	*pos./neg. f*	*complex number*
1	0	$(\frac{2\pi}{8}) \cdot 0$	$(\frac{2\pi}{8}) \cdot 0$	$X[0]$ is real, it is the sum of all samples.
2	1	$(\frac{2\pi}{8}) \cdot 1$	$(\frac{2\pi}{8}) \cdot 1$	$X[1]$ is complex
3	2	$(\frac{2\pi}{8}) \cdot 2$	$(\frac{2\pi}{8}) \cdot 2$	$X[2]$ is complex
4	3	$(\frac{2\pi}{8}) \cdot 3$	$(\frac{2\pi}{8}) \cdot 3$	$X[3]$ is complex
5	4	$(\frac{2\pi}{8}) \cdot 4$	$(\frac{2\pi}{8}) \cdot 4$	$X[4]$ is real, it equals $\sum$(even samples) - $\sum$(odd samples)
6	5	$(\frac{2\pi}{8}) \cdot 5$	$-(\frac{2\pi}{8}) \cdot 3$	$X[5]$ is the complex conjugate of $X[3]$
7	6	$(\frac{2\pi}{8}) \cdot 6$	$-(\frac{2\pi}{8}) \cdot 2$	$X[6]$ is the complex conjugate of $X[2]$
8	7	$(\frac{2\pi}{8}) \cdot 7$	$-(\frac{2\pi}{8}) \cdot 1$	$X[7]$ is the complex conjugate of $X[1]$

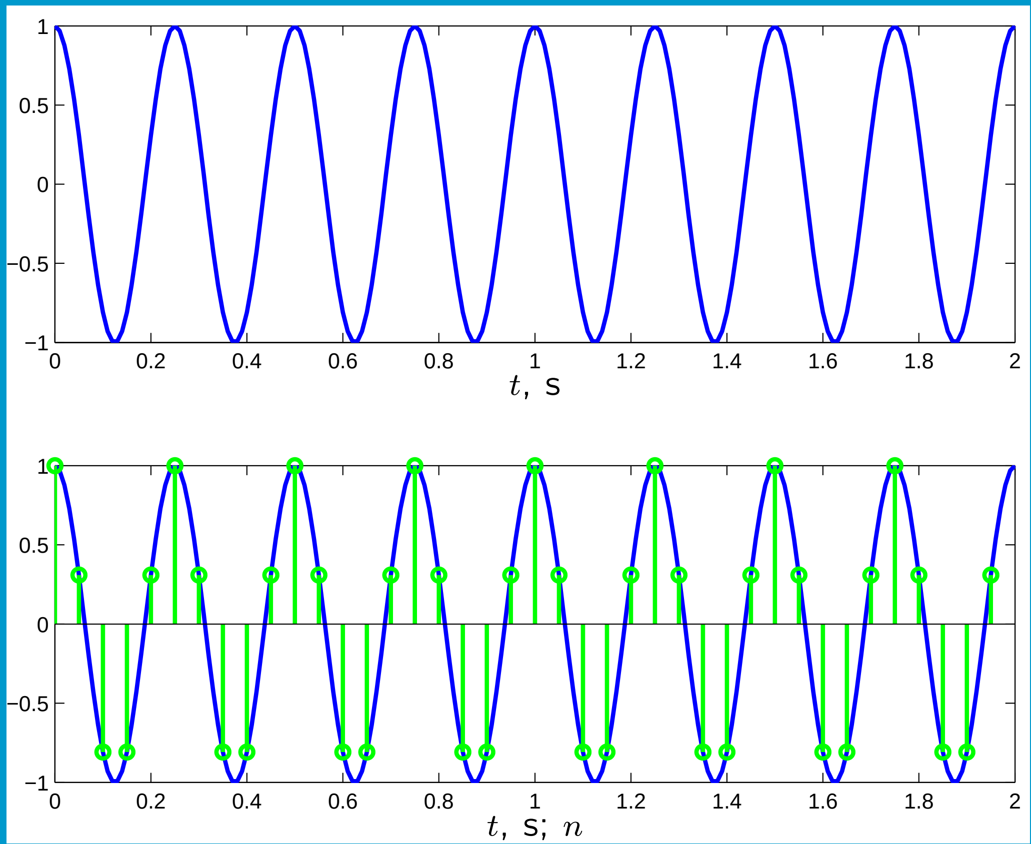
So, the complex sequence $X[i]$ from $i = 1$ $(k = 0)$ to $N/2$ $(k = N/2-1)$ contains all the information we need. We get $N/2$ complex numbers ($= N$ real numbers).



In fact, when we have the DFT array for i=1 until $i = N/2$, the full Fourier transform is known. For negative frequencies, we can just use the complex conjugate of the DFT coefficients for i=2 until $i = N/2$ (i=1 contains the real sum of the discrete time series).

Example : DFT of a cosine

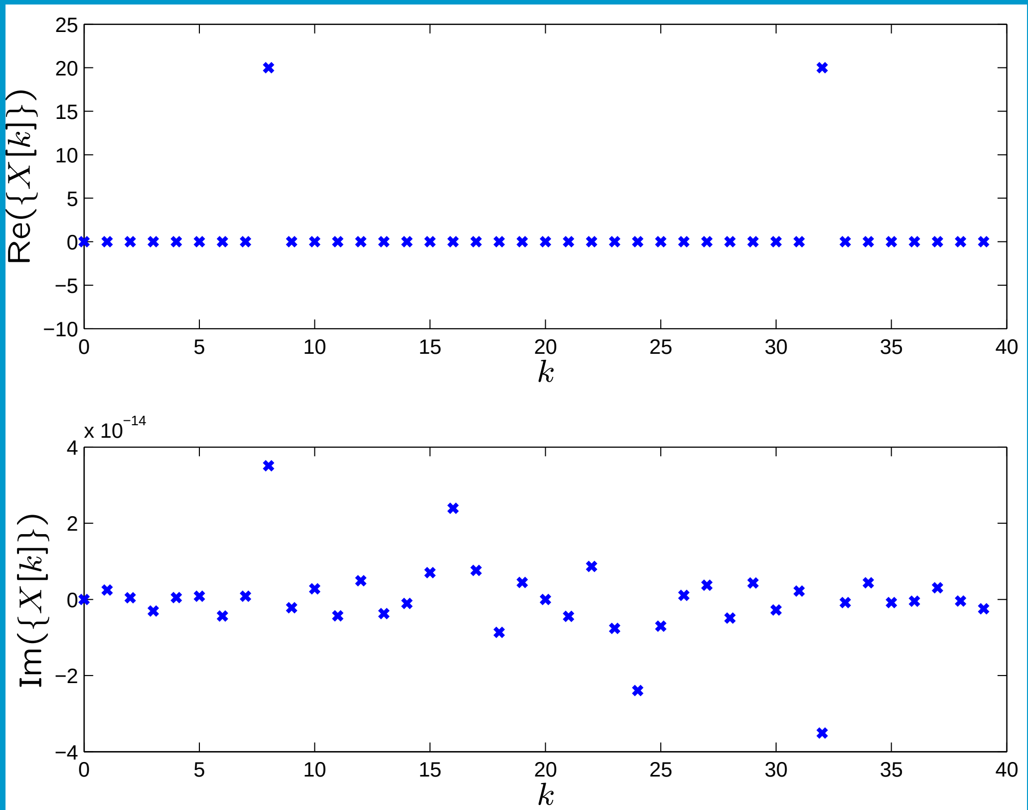
DFT of a cosine function $x[n] = \cos(2\pi f_0 n \triangle t)$.



Assume we have obtained N samples ($n = 0, ..., N-1$) of this function x_n.

When $f_0 = \frac{1}{N_0 \triangle t}$ and $N = rN_0$ then this cosine function 'fits' an integer number of times in the measurement period.

Let's take $T = 2$ s, $f_s = 20$ Hz, and $f_0 = 4$ Hz. Then $N = 40$, $N_0 = 5$ and $r = 8$. The DFT frequency resolution (f_s/N) is 0.5 Hz.

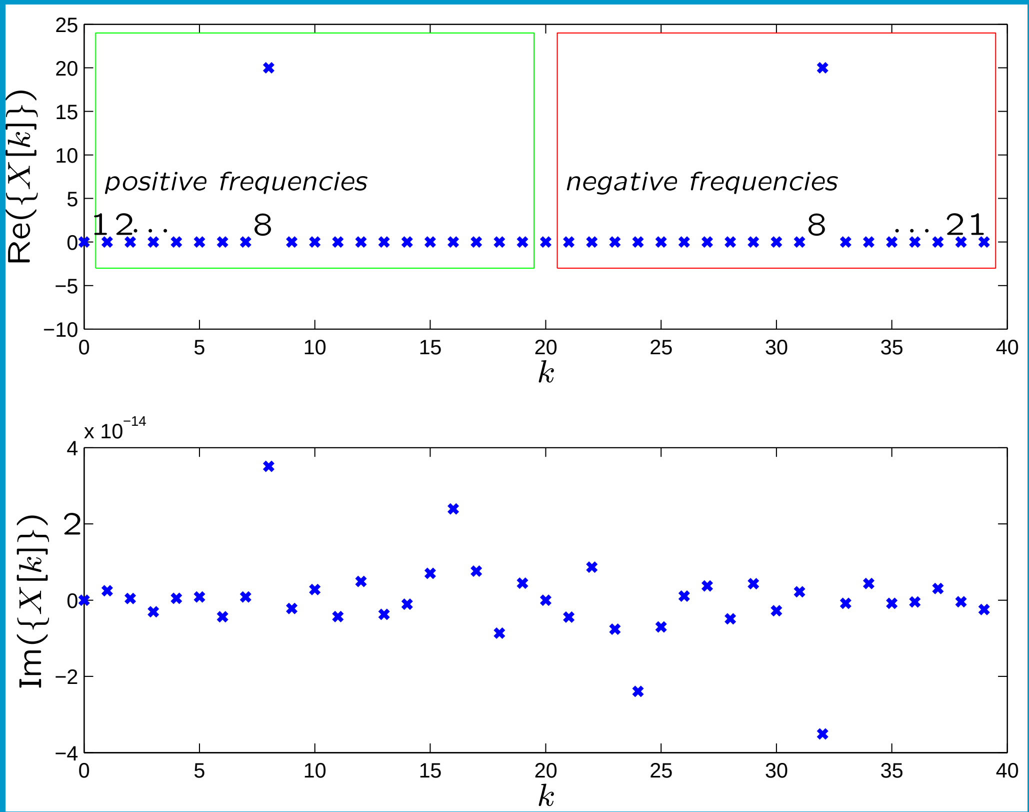


Using MATLAB's `fft`-function, we obtain 40 complex numbers.

The real and imaginary parts of these numbers are plotted in the figure on the left.

Note that the imaginary part is extremely small (10^{-14}), as it should because we are Fourier-transforming a cosine, a real and even function of time.

Remember that $X[0]/N$ is the signal's average, which equals 0.

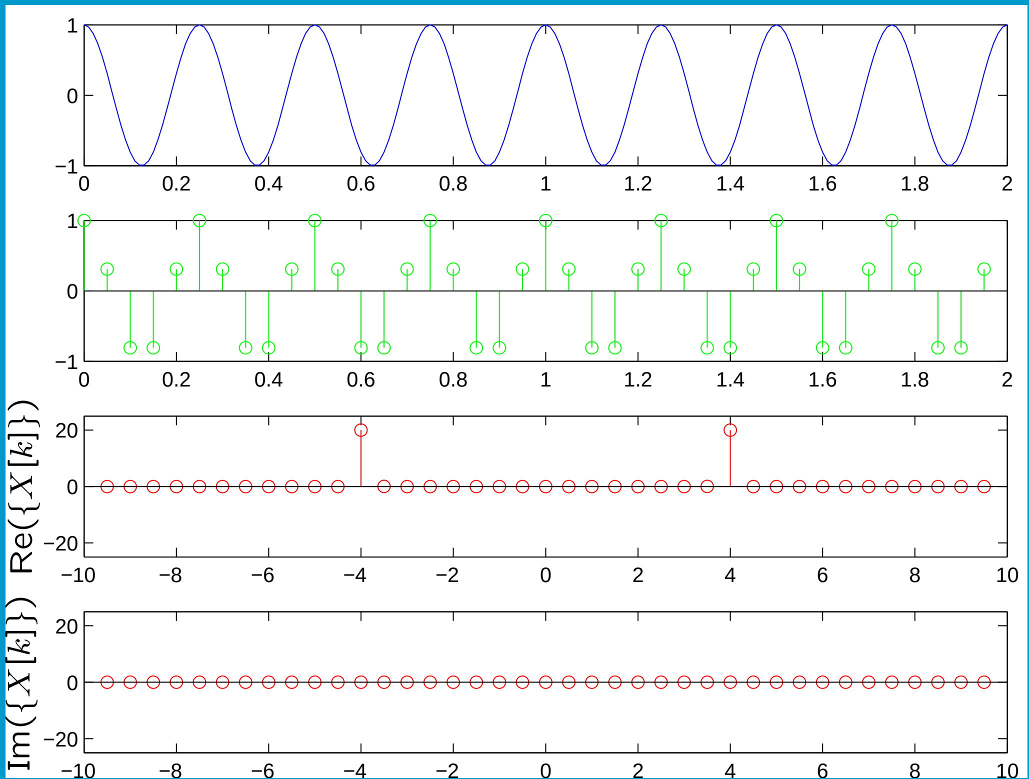


Further remember that the $X[k]$'s for $k = 1$ until $k = \frac{N}{2} - 1$ represent the positive frequencies,

... and the $X[k]$'s for $k = \frac{N}{2} + 1$ until $k = N - 1$ equal the negative frequencies.

$X[k = N/2]$ is the frequency-content at $f = f_s/2$.

To get the two-sided DFT we need to put the negative frequencies at their correct positions: move all numbers in the red box to the left hand-side...



... then multiply the k's with $\frac{f_s}{N}$ (= 0.5 Hz) to get the frequency in Hertz.

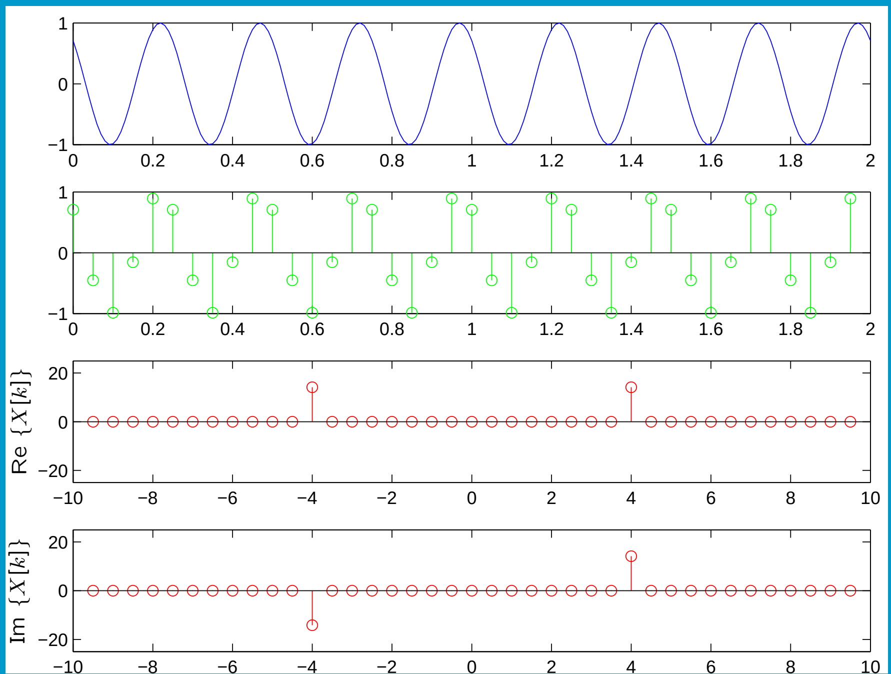
The result is shown on the left, a nice two-sided plot of the real and imaginary values of the DFT coefficients $X[k]$.

As could have been expected, the signal has power at 4 Hz, i.e. at the 8^{th} frequency component $k = 8$.

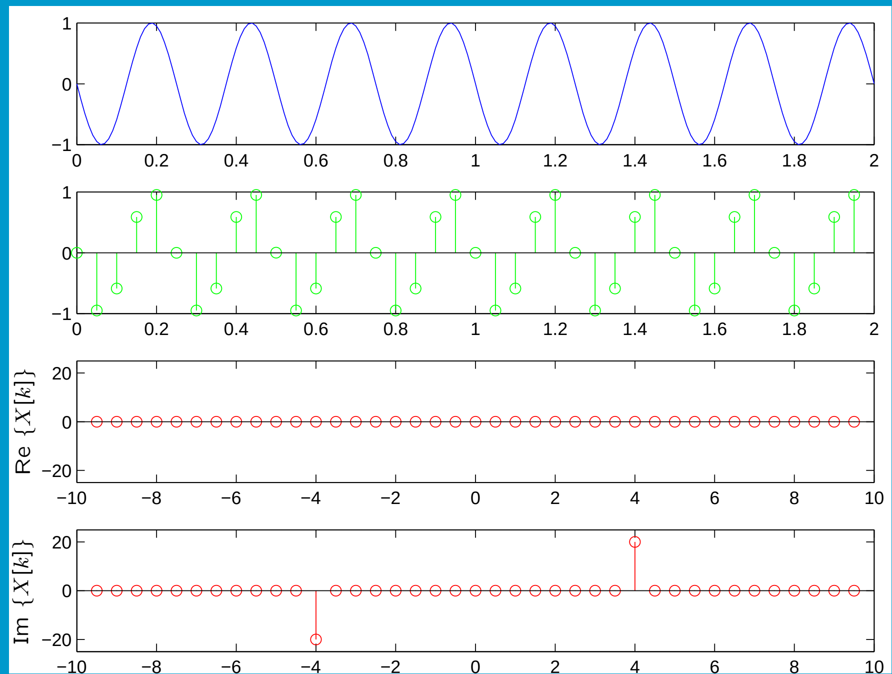
The power at 4 Hz equals $\frac{N}{2}$ (remember that $A = 1$ in this example), so the DFT basically 'integrates' (= sums up) the (squared!) cosine.

When the cosine is phase-shifted in time with 45 and 90 degrees the real and imaginary parts of the DFT will change.

phase shift 45°

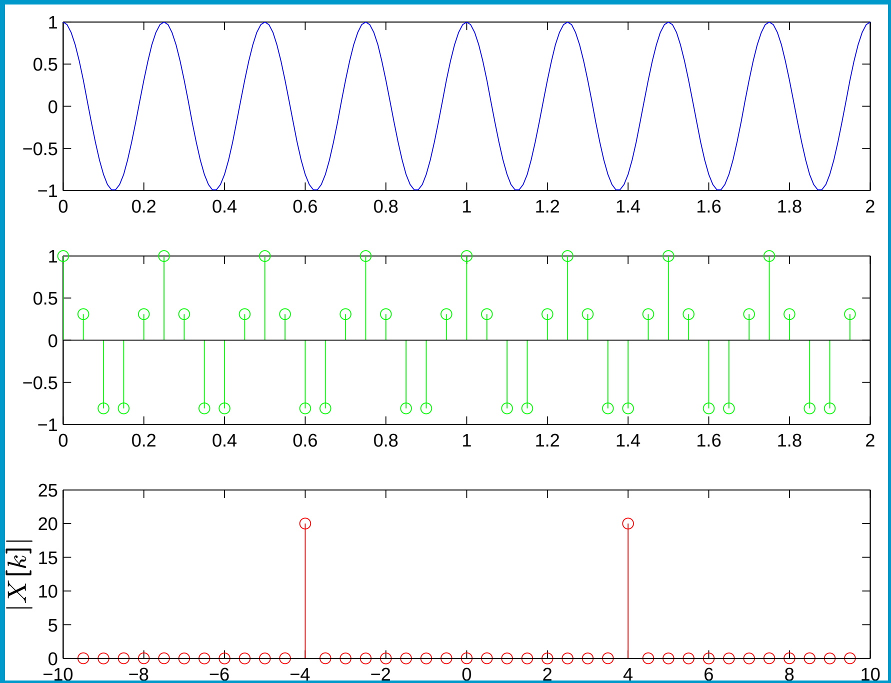


phase shift 90° (= sine function!)

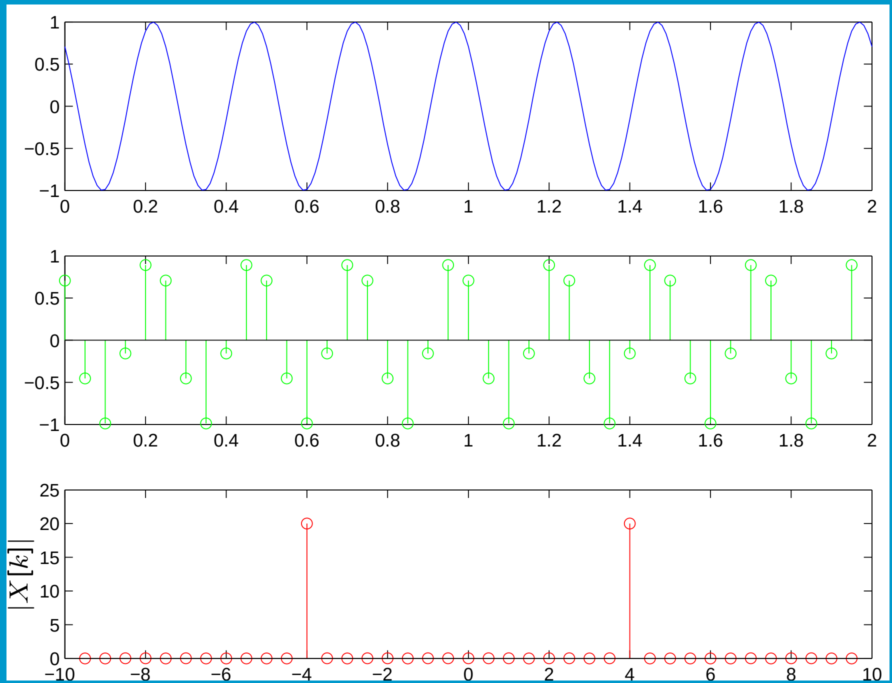


Yet, the magnitude of the complex vector, $|X[k]|$, remains the same:

phase shift 0°

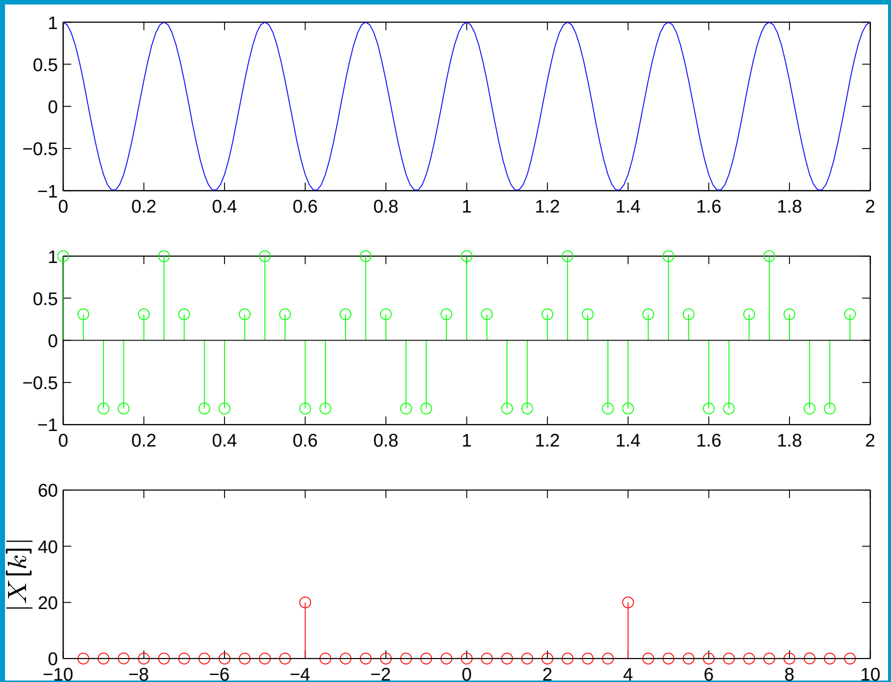


phase shift 45°

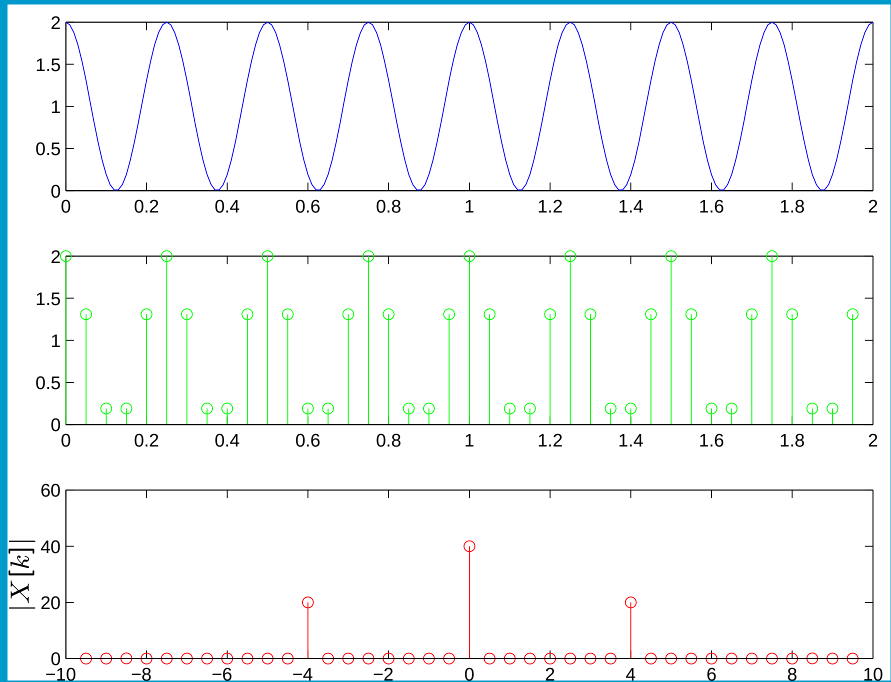


Adding '1' to the cosine function, increasing the signal average, and we get a different DFT coefficient at the 'zero frequency'. The average equals $X[0]/N$.

mean $= 0$

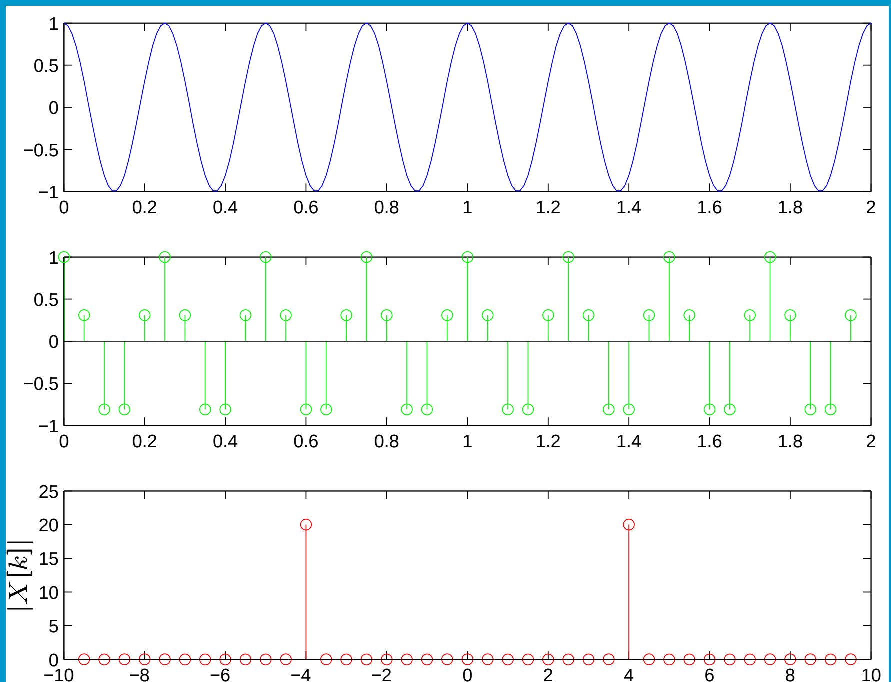


mean $= 1$

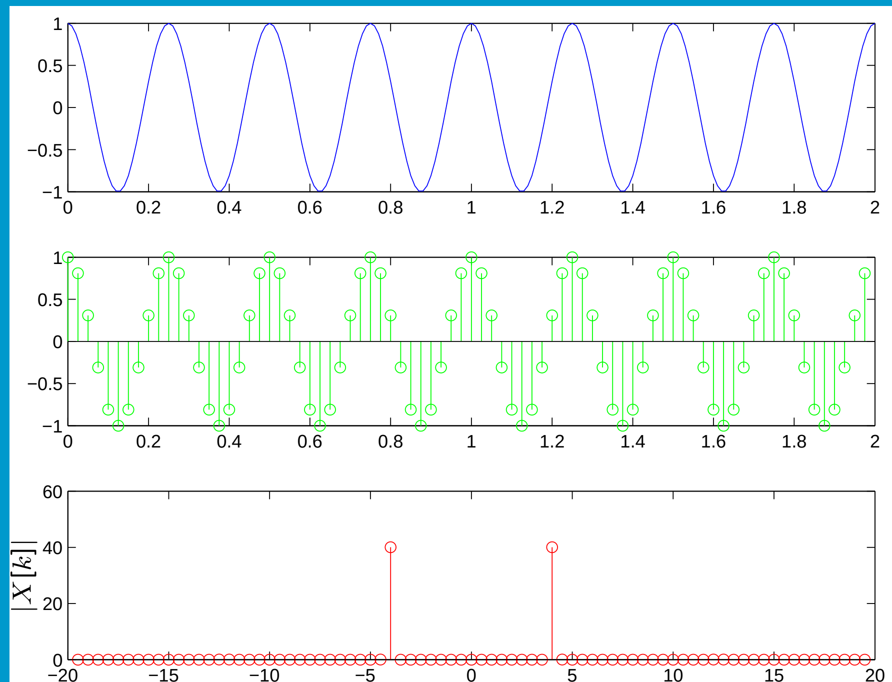


Increasing the sampling frequency f_s to 40 Hz, with the same observation time $T = 2$ s, we get a ‘wider’ frequency range, but the DFT frequency resolution remains the same (0.5 Hz).

$f_s = 20$ Hz ($N = 40$)

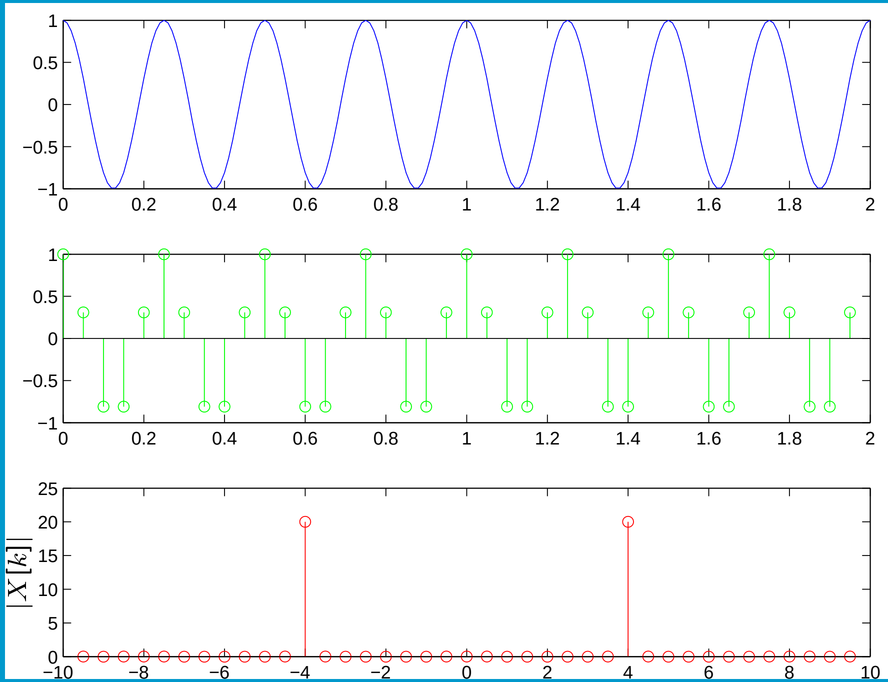


$f_s = 40$ Hz ($N = 80$)

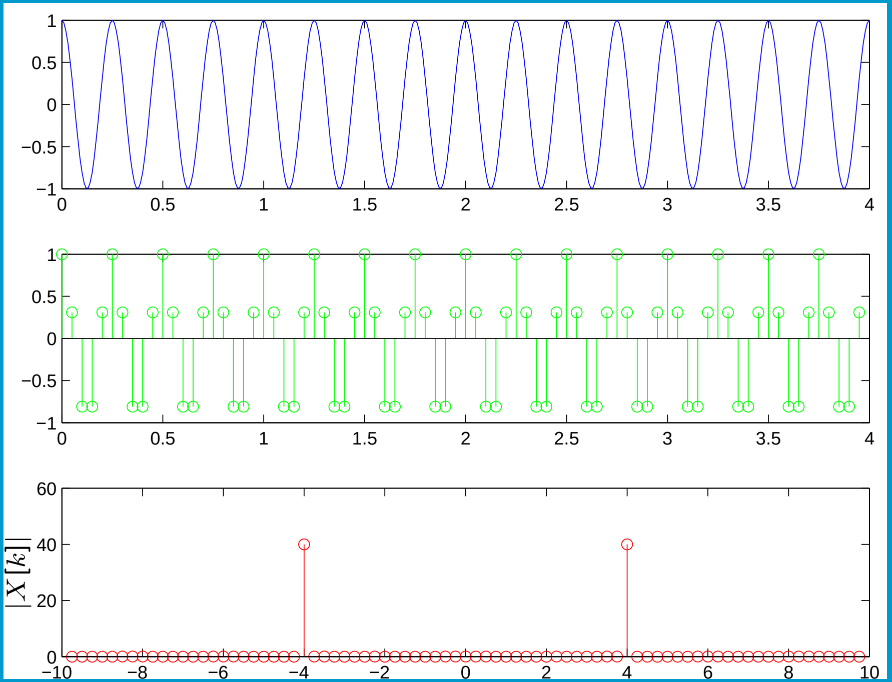


Increasing the observation time T, with constant sampling frequency f_s means that the frequency range remains the same, but we do get a better DFT frequency resolution of 0.25 Hz.

$f_s = 20$ Hz, $T = 2$ s $(N = 40)$

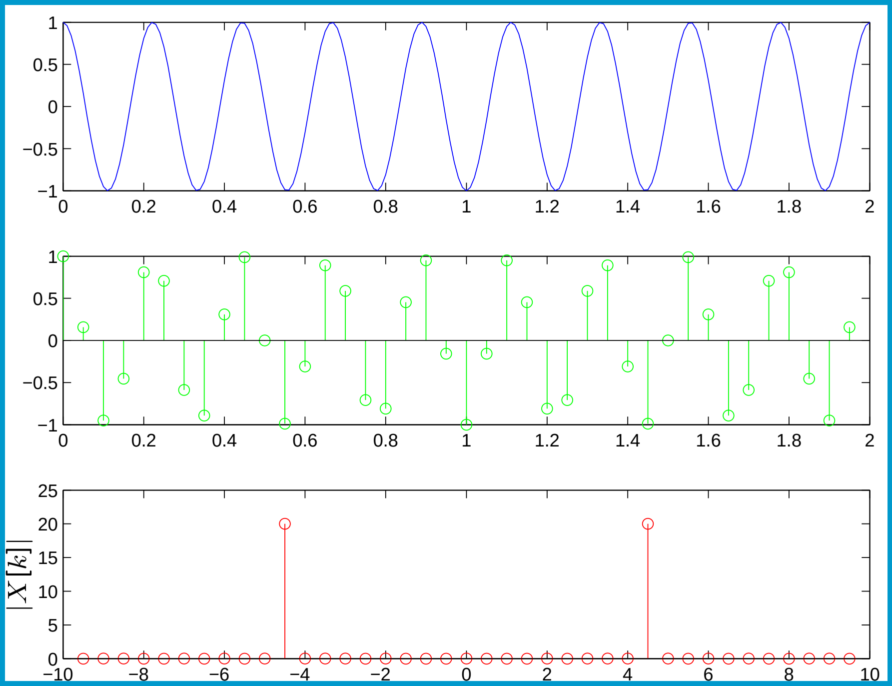


$f_s = 20$ Hz, $T = 4$ s $(N = 80)$

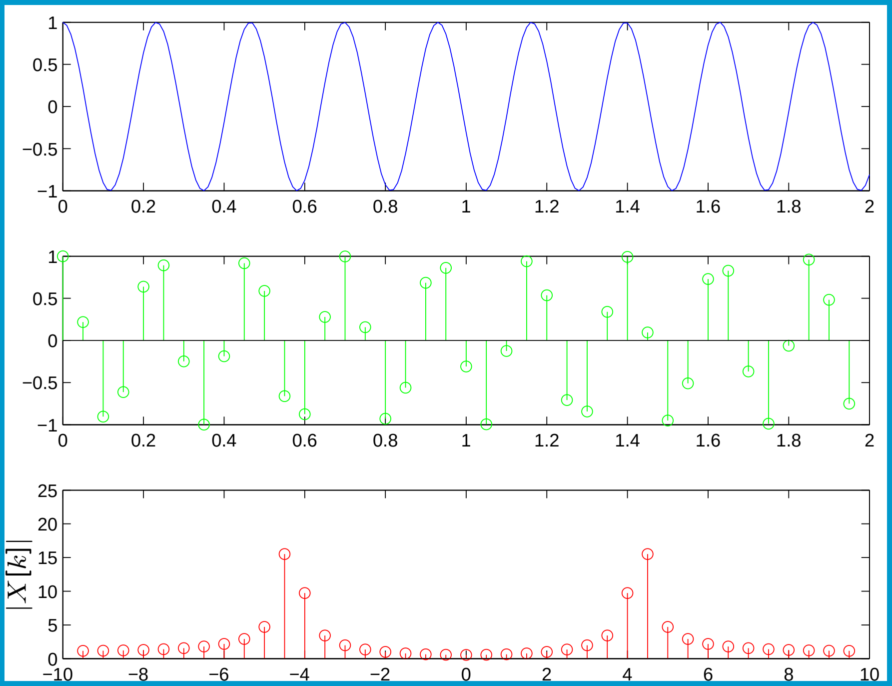


Let us keep the observation time T at 2 seconds and the sampling frequency f_s at 20 Hz. Consider a situation where the frequency of the cosine function becomes $f_0 = 4.5$ Hz (left) or $f_0 = 4.3$ Hz (right):

$f_0 = 4.5$ Hz



$f_0 = 4.3$ Hz



$\boxed{\text{The phenomenon is called } \textbf{spectral leakage}\text{, or } \textbf{leakage} \text{ for short}}$

Leakage makes the application of the DFT a tedious one, i.e., one has to be careful in interpreting its results. In the example above, the leakage appears when the cosine function does not 'fit' an integer number of times in the measurement period.

When $f_0 = 4.5$ Hz we get 9 periods in $T = 2$ seconds OK

When $f_0 = 4.3$ Hz we get 8.6 periods in $T = 2$ seconds not OK

Or, equivalently, the quotient of the frequency of the cosine function f_0 and the frequency resolution f_s/N is not an integer:

When $f_0 = 4.5$ Hz we get the '9th' component of the DFT

When $f_0 = 4.3$ Hz we have to use the components around the '9'th component to 'fit' the DFT

To better understand the cause of leakage, we have to once again consider the relation between the DTFT and the DFT.

Relation of the DFT and the DTFT

Recall that the DTFT was defined as the discrete-time Fourier transform of a sequence $x[n]$ of infinite length. The DFT, on the other hand, is used for measured sequences $y[n]$ that only have length N.

Now suppose $y[n]$ equals $x[n]$ for $n = 0, \ldots, N-1$. Define another infinite length sequence, $z[n]$, which equals $x[n]$ multiplied with a **time window** $w[n]$, defined as:

$$w[n] = \begin{cases} 1 & n = 0, 1, \ldots, N-1 \\ 0 & \text{other } n \end{cases}$$

Clearly, $z[n]$ is zero for all n, but equals $x[n]$ from $n = 0, \ldots, N-1$.

Then, it can be shown that the DTFT of $z[n]$ equals the DFT of $y[n]$ at the discrete set of frequencies of the DFT.

Proof

The DFT of $y[n]$ is defined as:

$$Y[k] = \sum_{n=0}^{N-1} y[n]e^{-jk\frac{2\pi}{N}n} = \sum_{n=0}^{N-1} x[n]e^{-jk\frac{2\pi}{N}n},$$

for the set of frequencies $\Omega_k = k\frac{2\pi}{N}$.

Now, for the continuous discrete frequency Ω, the DTFT of $z[n]$ equals:

$$\begin{aligned} Z(\Omega) &= \sum_{n=-\infty}^{\infty} z[n]e^{-j\Omega n} = \sum_{n=-\infty}^{\infty} w[n]x[n]e^{-j\Omega n} \\ &= \sum_{n=0}^{N-1} x[n]e^{-j\Omega n} \end{aligned}$$

At the discrete set of frequencies of the DFT, $\Omega = \Omega_k = k\frac{2\pi}{N}$ we get:

$$\begin{aligned} Z(\Omega_k) &= \sum_{n=0}^{N-1} x[n]e^{-j\Omega_k n} \\ &= \sum_{n=0}^{N-1} x[n]e^{-jk\frac{2\pi}{N}n} \\ &= Y[k] \end{aligned}$$

qed

Multiplication of $w[n]$ and $x[n]$ in the time domain means that we have a **convolution** of the DTFTs $W(\Omega)$ and $X(\Omega)$ in the frequency domain.

It can be shown that:

$$W(\Omega) = \frac{\sin(\frac{N}{2}\Omega)}{\sin(\frac{1}{2}\Omega)} e^{-j(N-1)\Omega/2}$$

This is a 'sinc-like' function. $W(\Omega)$ is zero when $\frac{N}{2}\Omega = m\pi$, i.e., when $\Omega = m\frac{2\pi}{N}$ for integer $m \in \mathcal{Z}, m \neq 0$. With $\omega = \Omega/\triangle t$ and $f = \omega/(2\pi)$ we see that the function passes zero for $f = m\frac{f_s}{N}$, i.e., exactly at integer multiples of the frequency resolution of the DFT.

$$f = \frac{\omega}{2\pi} = \frac{\Omega}{\triangle t}\frac{1}{2\pi} = m\frac{2\pi}{N}\frac{1}{\triangle t}\frac{1}{2\pi} = m\frac{f_s}{N}$$

Returning to the cosine-function example, the DTFT of $x[n] = \cos(2\pi f_o n \triangle t)$ equals:

$$X(\Omega) = \frac{\pi}{\triangle t}\left(\delta(\Omega - 2\pi f_0 \triangle t) + \delta(\Omega + 2\pi f_0 \triangle t)\right)$$

Convolving $W(\Omega)$ with $X(\Omega)$ means (because we are dealing with Dirac functions in $X(\Omega)$) that in order to get the DTFT of $Y(\Omega)$ we must add the 'copy' of $W(\Omega)$ centered at the positive frequency $\Omega = 2\pi f_0 \triangle t$ and the copy of $W(\Omega)$ centered at the negative frequency $\Omega = -2\pi f_0 \triangle t$.

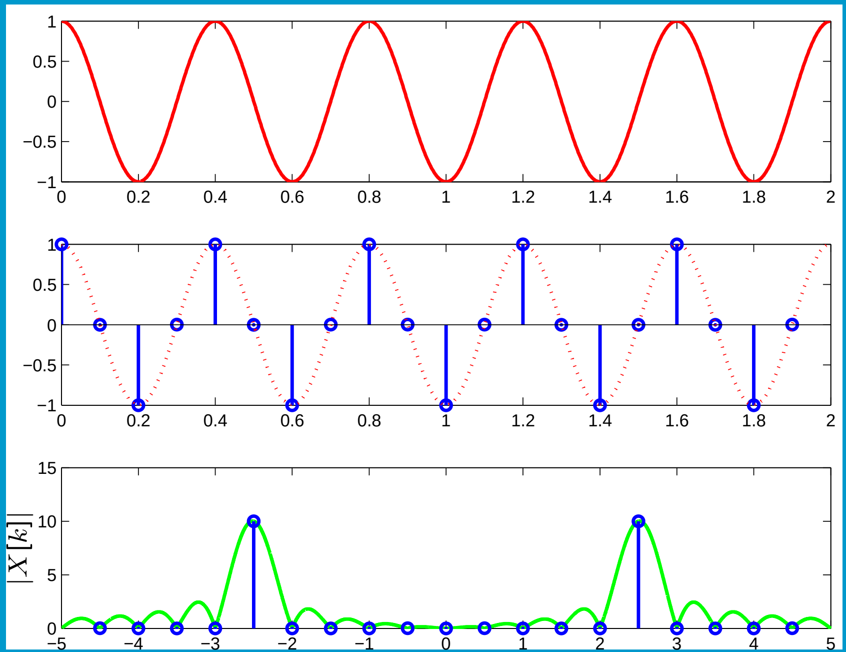
Then, since $W(\Omega)$ is zero at $\Omega = m\frac{2\pi}{N}$ ($m \in \mathcal{Z}$, $m \neq 0$), $Y(\Omega)$ will be zero at frequencies Ω equal to $2\pi f_0 \triangle t + m\frac{2\pi}{N}$, or, equivalently, with $f = \omega/(2\pi) = \Omega/(\triangle t 2\pi)$, at f equal to $f_0 + m\frac{1}{N\triangle t} = f_0 + m\frac{f_s}{N}$ ($m \in \mathcal{Z}$, $m \neq 0$).

Now, when the DFT frequency resolution f_s/N 'fits' an integer number ℓ in f_0 (i.e., $f_0 = \ell\frac{f_s}{N}$), the locations of the zeros in the 'sinc-like' function become $(\ell + m)\frac{f_s}{N}$. That is, they do not change at all, and no leakage appears!

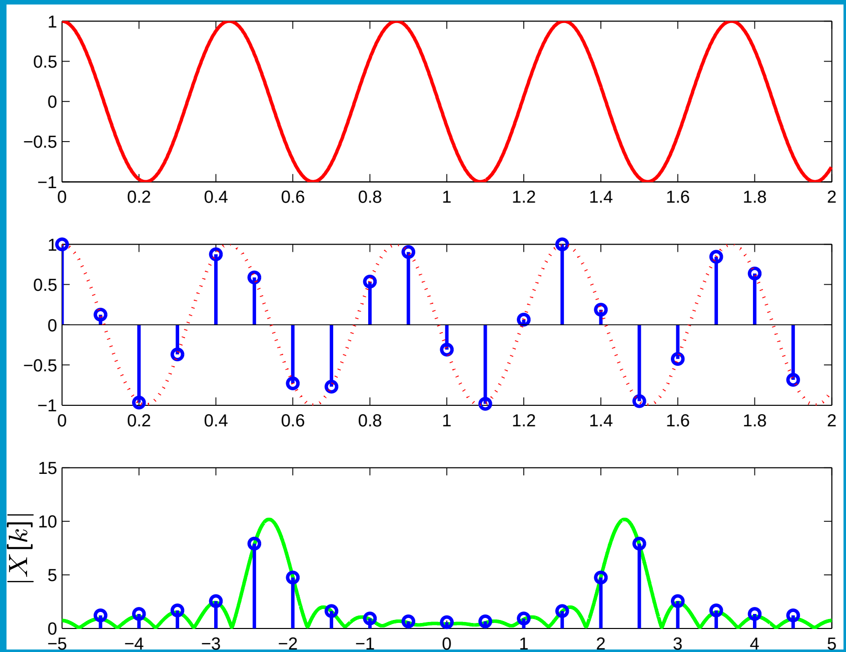
When this is not the case, however, the zeros of the 'sinc-like' function will not coincide with the frequency bins of the DFT, and spectral leakage will appear.

Take an observation time T of 2 seconds and a sampling frequency f_s of 10 Hz. The DFT frequency resolution is then 0.5 Hz. Consider a situation where the frequency of the cosine function becomes f_0=2.5 Hz (left) or f_0=2.3 Hz (right):

$f_0 = 2.5$ Hz



$f_0 = 2.3$ Hz



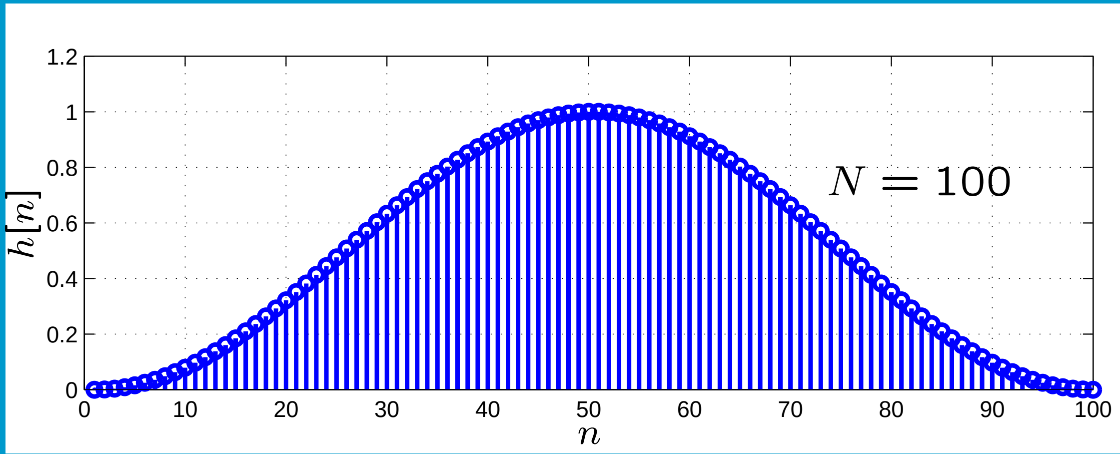
The green lines show the (absolute values of the) 'sinc' function. It is shifted with 2.5 Hz (5 bins) in the left-hand figure, and 2.3 Hz (4.6 bins) in the right-hand figure.

Windowing

In most practical situations, the chance that exactly an integer number of periods of an arbitrary signal 'fit' in the measurement period, is zero. Except in the case we can construct a signal ourselves (which often occurs in system identification applications), we will generally have leakage in the DFT.

One way to counter the effects of leakage is to use a windowing technique. Recall that the finite observation of N samples can be considered as the complete observation *multiplied* with a **rectangular** window $w[n] = 1$ (n=0,. . . , N-1) in the time domain. Other windows exist as well, which generally lead to better results. Examples are the Bartlett, Hanning, Hamming and Chebyshev windows.

Example: the Hanning window

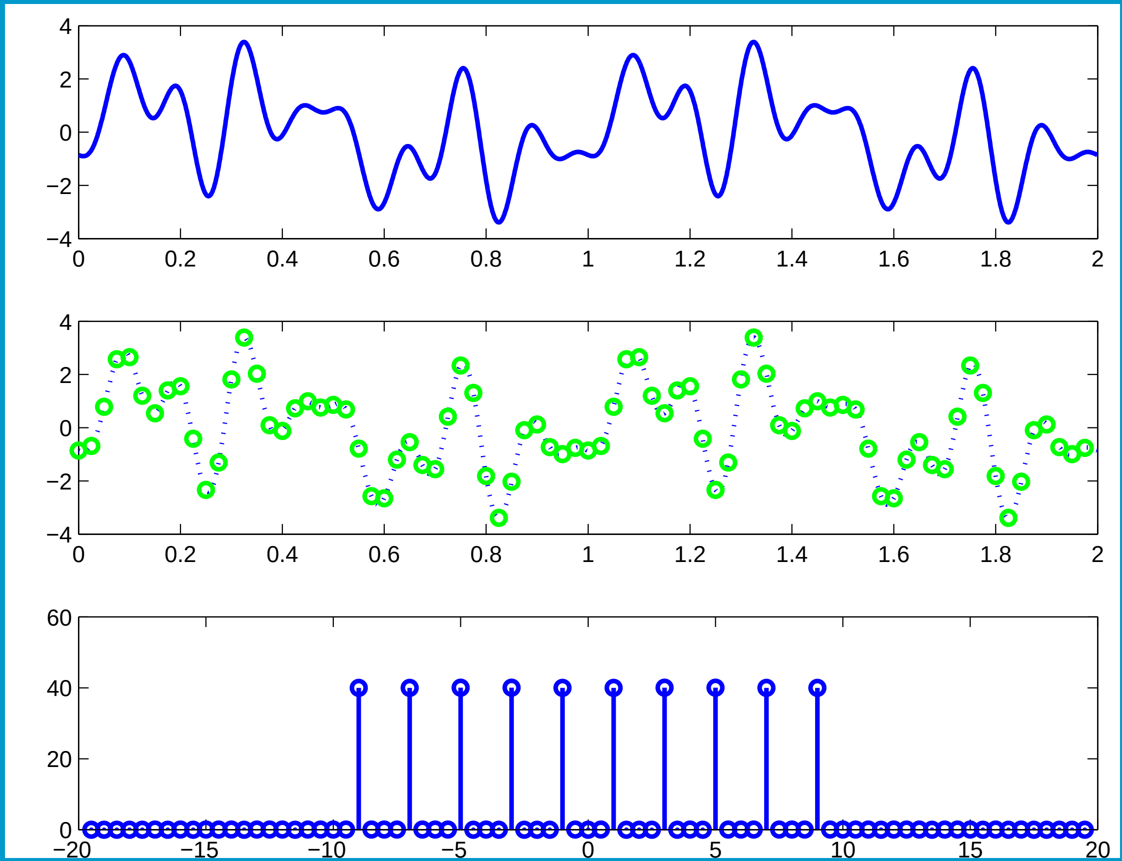


The Hanning window (or cosine window) can be defined as:

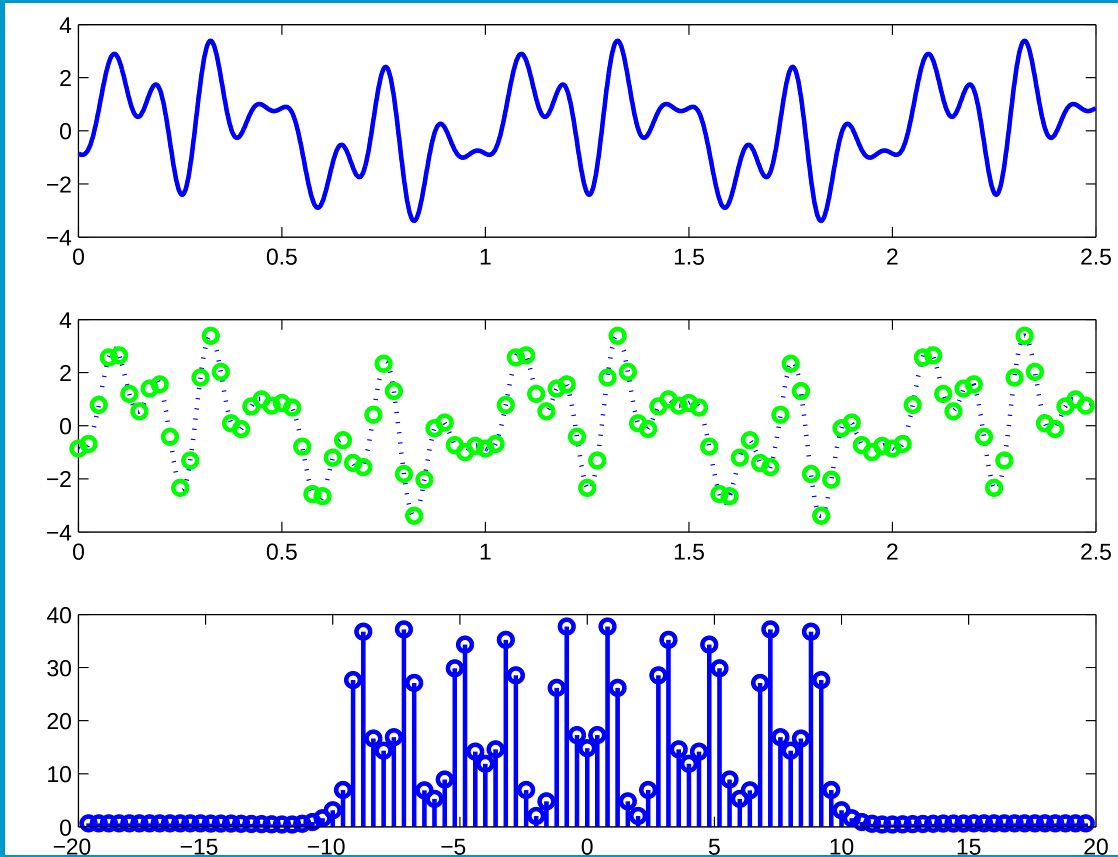
$$h[n] = \begin{cases} 0.5\,(1 - \cos(2\pi n/N)) & \text{for } n = 0, 1, \ldots, N-1 \\ 0 & \text{for other } n \end{cases}$$

First multiplying the observations $x[n]$ with $h[n]$, and then computing the DFT yields better estimate of the signal's 'true' frequency components, independent of the measurement length.

Take an observation time T of 2 seconds and a sampling frequency f_s of 40 Hz. The DFT frequency resolution is then 0.5 Hz. Consider a signal that consists of the sum of 5 cosine functions with frequencies 1, 3, 5, 7 and 9 Hz.

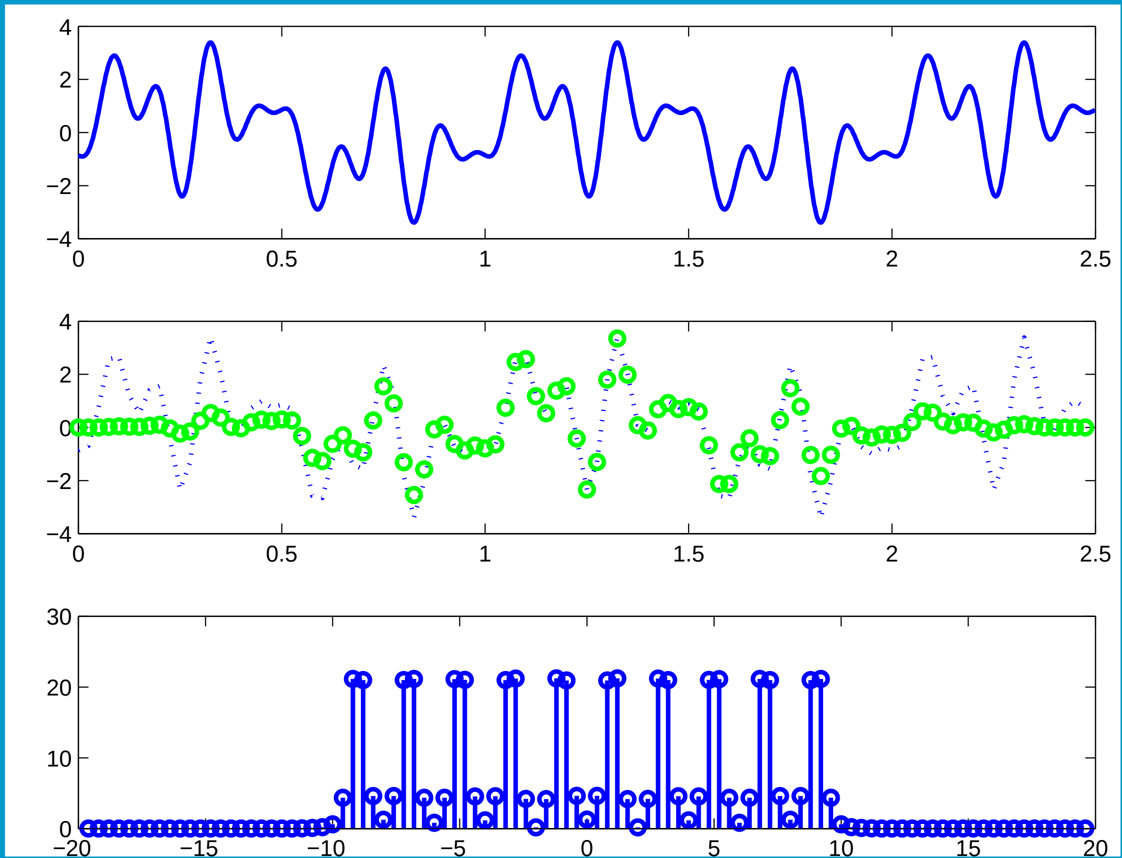


When the observation time T becomes 2.5 seconds (sampling frequency f_s=40 Hz, resolution 0.4 Hz) the signal does not ‘fit’ an integer amount of periods, and leakage occurs



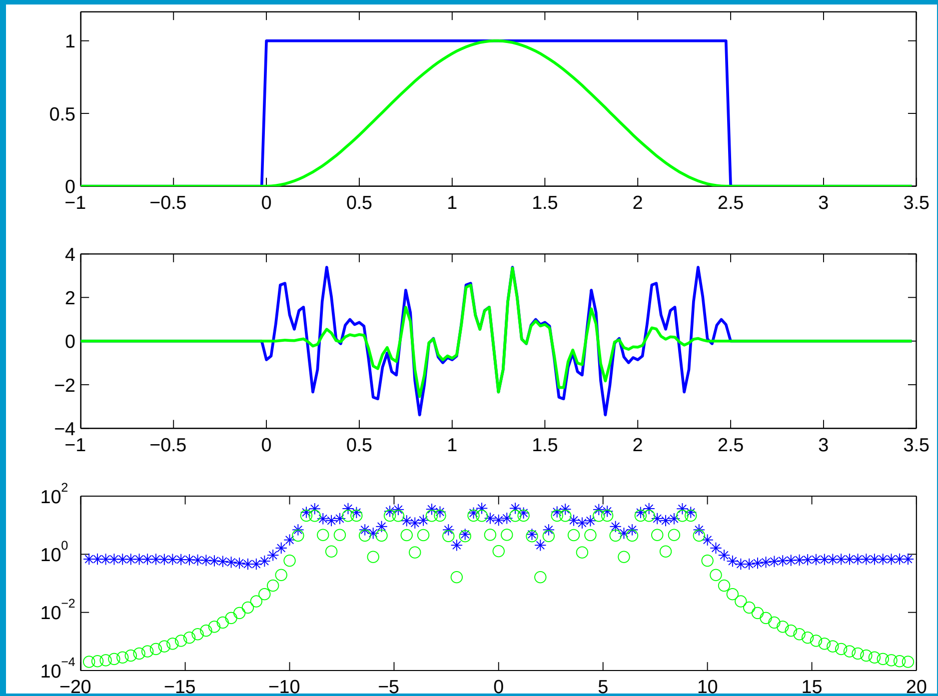
The original frequencies of 1, 3, 5, 7 and 9 Hz can hardly be distinguished !!

First multiplying the signal with the Hanning window, and then taking the DFT yields a considerable improvement:



The original frequencies of 1, 3, 5, 7 and 9 Hz can be distinguished much better.

Showing the operations in the time and frequency domain. The top figure shows the rectangular (blue) and Hanning (green) windows in the time domain. The center figure shows the signal multiplied with the time windows. The bottom figure shows what happens in the frequency domain.



The Fast Fourier Transform (FFT)

The FFT is an extremely efficient algorithm, developed by Cooley and Tukey around 1965. It has revolutionized spectral analysis in the sense that (as we will see) with the FFT the PSD function is now generally computed **directly** using the FFT-ed discrete time series, instead of through the intermediate step of calculating the covariance function and then Fourier transforming that.

When $N = 2^m$, i.e. the number of samples is a power of 2:
(Example: for 4.096 samples $m = 12$)

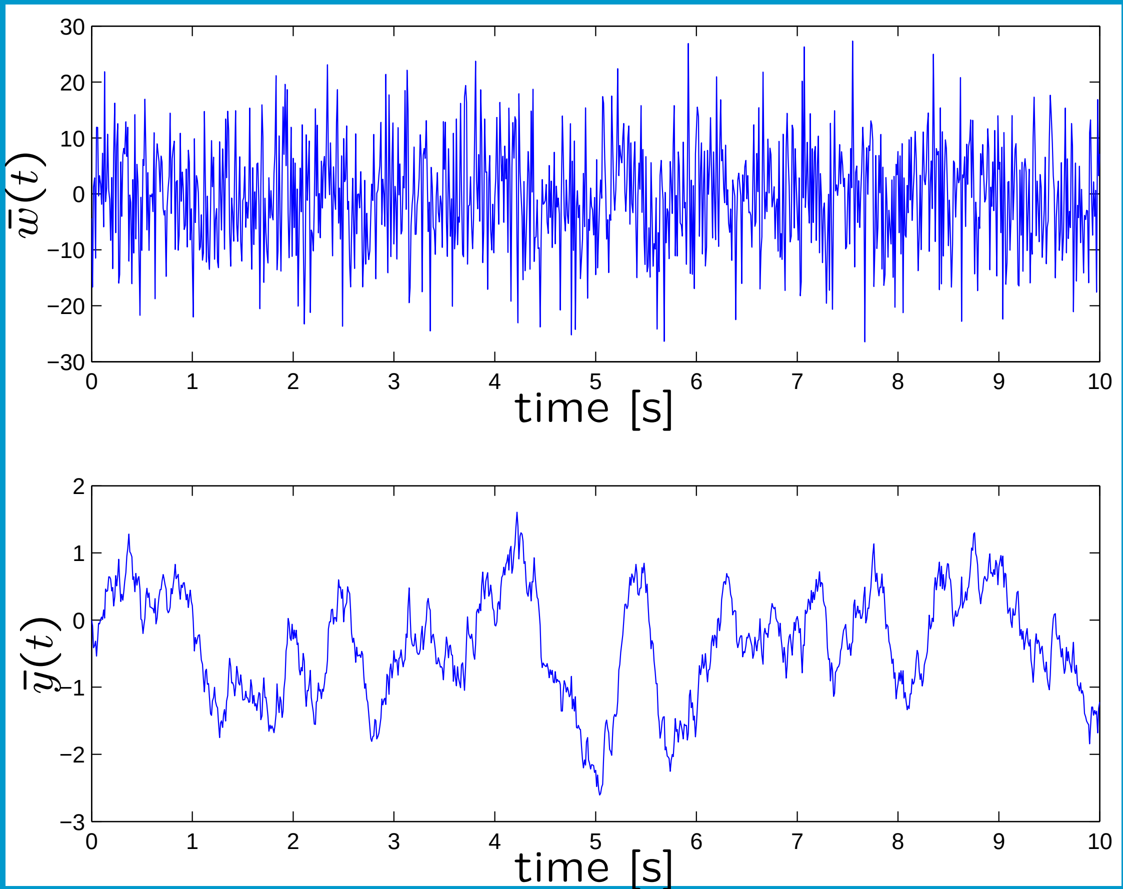
the DFT needs 2^{2m} operations (Ex: 16.777.216)

the FFT needs $m2^m$ operations (Ex: 49.152)

So, here the FFT is $\frac{2^m}{m}$ times faster (Ex: 341) than the DFT. Read the lecture notes for a theoretical background of the FFT.

Matlab example of DFT/FFT

Consider the output $\bar{y}(t)$ of a first order LTI system $\frac{K}{1+j\omega\tau}$ ($K = 1$, $\tau = 0.5$) driven by white noise $\bar{w}(t)$. The sample frequency is 100 [Hz]; resolution 0.5 Hz.



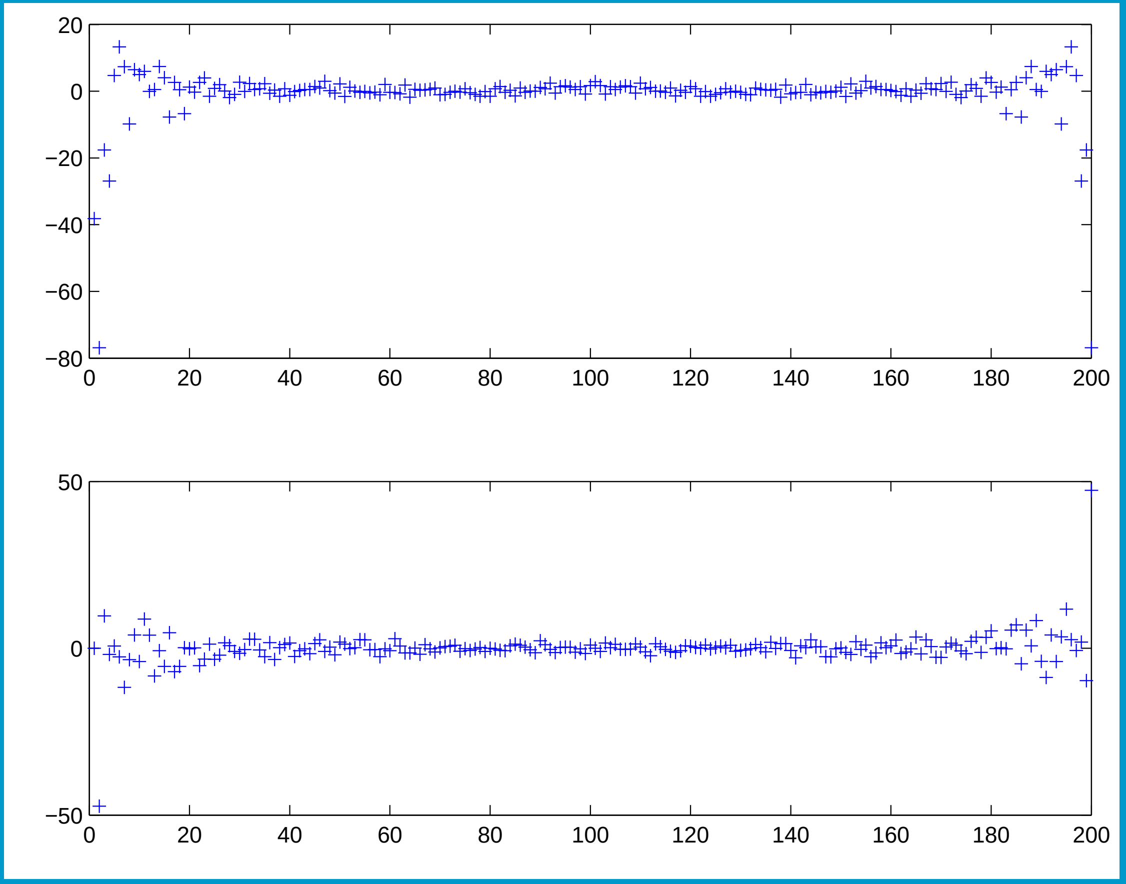
intensity $\bar{w}(t)$: W=1

$$\sigma_{\bar{y}}^2 = W\frac{K^2}{2\tau} = 1$$

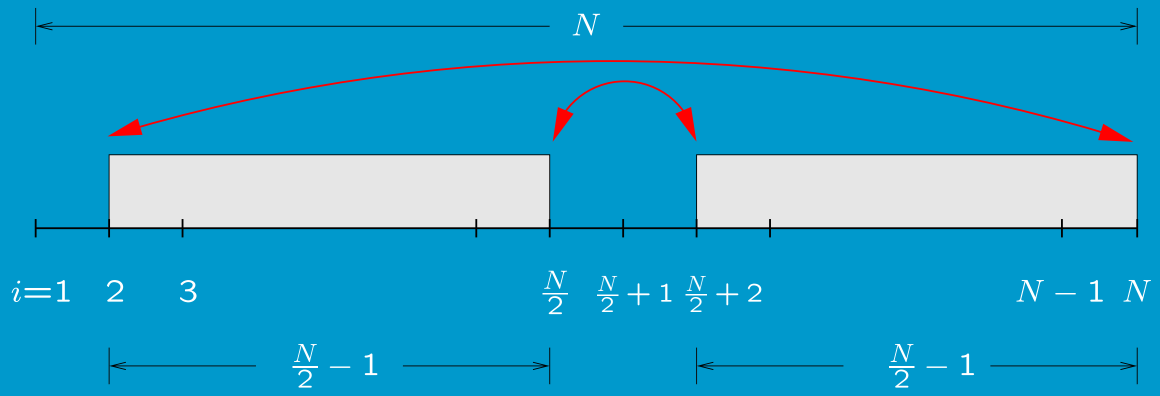
(Table 3.5)

It is a realisation!

Take the last two seconds ($N = 200$ samples) of this response and use Matlab's FFT algorithm (no m-file!) to transform the discrete time series to the frequency domain. We obtain 200 complex numbers $Y[k]$:

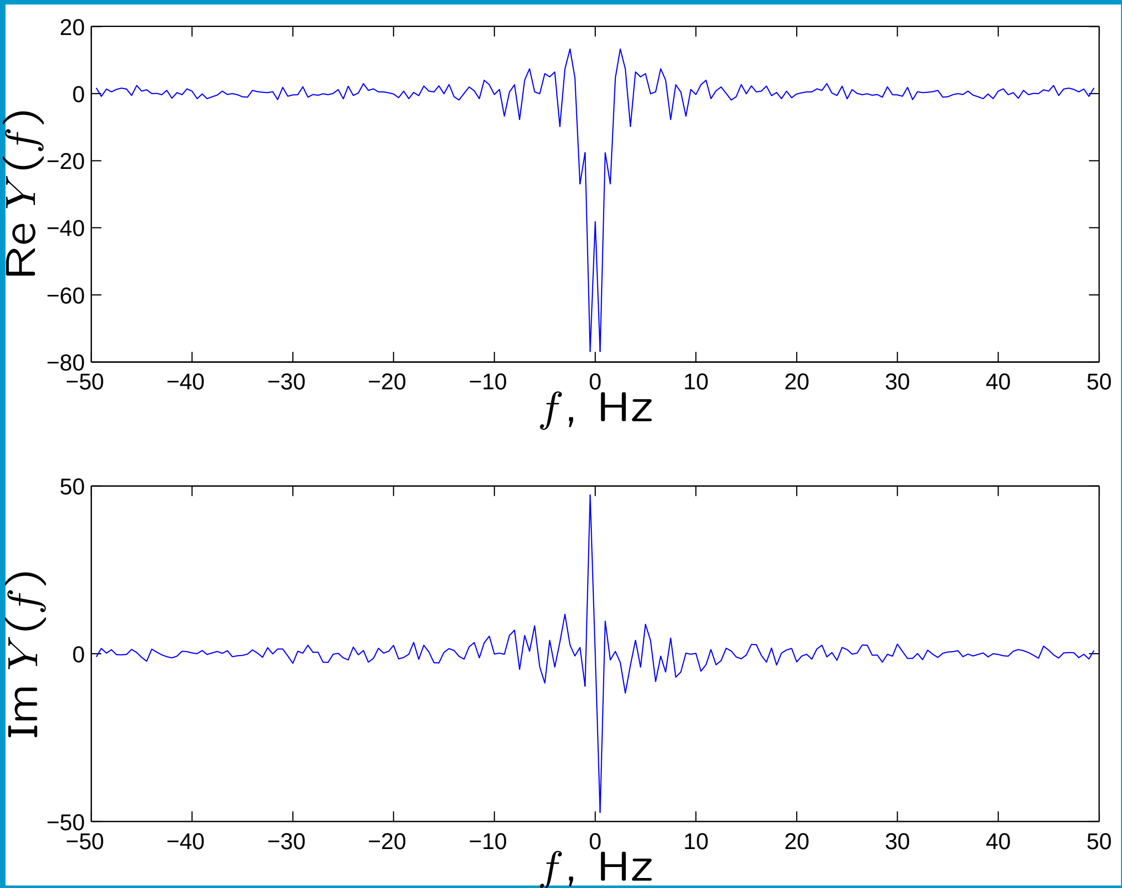


Remember that the complex array $Y[i]$ contains the information twice, i.e. for positive frequencies as well as for negative frequencies.



$Y[1]$ is the 'zero frequency' (sum of all samples), $Y[i]$ for $i = 2$ until $N/2$ ($N/2-1$ numbers) contains the DFT coefficients for positive frequencies, $Y[N/2+1]$ is just ignored, and $Y[i]$ for $i = N/2+2$ until $i = N$ ($N/2-1$ numbers) contains the complex conjugates of the first $N/2-1$ numbers

$$Y[N] = Y^*[2]$$
$$Y[N-1] = Y^*[3]$$
$$\ldots$$
$$Y[N/2+2] = Y^*[N/2]$$



Clearly, the real part of the DFT is even, and the imaginary part odd. With f_s=100 Hz, we can look at $f \in [-50, 50]$.

The calculation of spectral estimates

In continuous time, the Power Spectral Density function is defined as the CTFT transform of the covariance function:

$$S_{\bar{x}\bar{y}}(\omega) \;=\; \mathcal{F}\left\{C_{\bar{x}\bar{y}}(\tau)\right\} \;=\; \int_{-\infty}^{\infty} C_{\bar{x}\bar{y}}(\tau)e^{-j\omega\tau}\mathrm{d}\tau, \tag{3.27}$$

assuming zero-mean signals $\bar{x}$ and $\bar{y}$. So: $S_{\bar{x}\bar{y}}(\omega) = \mathcal{F}\left\{R_{\bar{x}\bar{y}}(\tau)\right\}$!!

In discrete time, assume we have N samples of a random discrete time sequence $\bar{x}[n]$ ($n = 0$ to $N-1$). A necessary assumption is that $\bar{x}[n]$ is considered a *periodic* function, of which only a single cycle is available for analysis. Then, for the DFT of $\bar{x}[n]$:

$$X[k] = \sum_{n=0}^{N-1} \bar{x}[n]e^{-jk\frac{2\pi}{N}n}, \quad \text{for } k = 0,\ 1,\ 2,\ \ldots,\ N\text{-}1$$

Consider calculating the auto-covariance function of $\bar{x}[n]$, which we assume has zero mean. Clearly, we can only *estimate* this function for time lags τ that are integer multiples of the sampling time $\triangle t$, i.e. for $\tau = r\triangle t$:

$$C_{\bar{x}\bar{x}}[r] = \frac{1}{N-|r|} \sum_{s=0}^{s=N-1-|r|} \bar{x}[s]\bar{x}[s+r], \text{ for } s = 0,\ 1,\ 2,\ \ldots\ N\text{-}1$$

Now, because $\bar{x}[n]$ is assumed to be periodic, i.e., $\bar{x}[s+r] = \bar{x}[s+r+N]$ we can always use N samples to compute the **circular** auto-covariance:

$$\boxed{C[r] = \frac{1}{N} \sum_{s=0}^{s=N-1} \bar{x}[s]\bar{x}[s+r]}$$

circular: $C[r+N] = C[r]$ (periodic with N)

Let it be clear that the circular covariance function does **not** represent the true covariance function, because in reality the discrete time series $\bar{x}[n]$ is, generally, **not** periodic. For now, we will assume the circular covariance function $C[r]$ to be the best available *estimate* of the true covariance function.

Because the circular covariance function is periodic with period N, its Fourier transform equals a Fourier series and is called a **periodogram**:

$$\boxed{I_{N\bar{x}\bar{x}}[k] = \sum_{r=0}^{N-1} C[r] e^{-jk\frac{2\pi}{N}r}}$$

by definition!

It is assumed that **the periodogram is the spectrum estimate**:

$$\boxed{\widehat{S}_{\bar{x}\bar{x}}[k] = I_{N\bar{x}\bar{x}}[k]}$$

Then, it can be shown that:

$$\boxed{I_{N_{\underline{x}\underline{x}}}[k] = \frac{1}{N} X^*[k] X[k] = \frac{1}{N} |X[k]|^2} \tag{4.25}$$

With Eq. (4.25) we see that the periodogram is directly related to the DFT $X[k]$, in other words: we do not *need* to calculate the (circular) covariance function, we can directly use the DFT of a discrete time series to become an estimate of the discrete-time power spectral density function.

Similarly, we obtain for the cross-PSD functions:

$$\hat{S}_{\underline{x}\underline{y}}[k] = I_{N_{\underline{x}\underline{y}}}[k] = \frac{1}{N} X^*[k] Y[k], \qquad \text{remember: } S_{\underline{x}\underline{y}}(\omega) = \mathsf{E}\left\{\bar{X}(-\omega)\bar{Y}(\omega)\right\}$$

and:

$$\hat{S}_{\underline{y}\underline{x}}[k] = I_{N_{\underline{y}\underline{x}}}[k] = \frac{1}{N} Y^*[k] X[k] \qquad \text{remember: } S_{\underline{y}\underline{x}}(\omega) = \mathsf{E}\left\{\bar{Y}(-\omega)\bar{X}(\omega)\right\}$$

Proof

$$
\begin{aligned}
I_{N\bar{x}\bar{x}}[k] &= \sum_{r=0}^{N-1} C[r] e^{-jk\frac{2\pi}{N}r} \\
&= \sum_{r=0}^{N-1} \left(\frac{1}{N} \sum_{s=0}^{N-1} \bar{x}[s]\bar{x}[s+r] \right) e^{-jk\frac{2\pi}{N}r} && \text{insert (4.22)} \\
&= \frac{1}{N} \sum_{r=0}^{N-1} \sum_{s=0}^{N-1} \bar{x}[s] e^{jk\frac{2\pi}{N}s} \bar{x}[s+r] e^{-jk\frac{2\pi}{N}(s+r)} && \text{rearrange} \\
&= \frac{1}{N} \sum_{s=0}^{N-1} \sum_{r=0}^{N-1} \bar{x}[s] e^{jk\frac{2\pi}{N}s} \bar{x}[s+r] e^{-jk\frac{2\pi}{N}(s+r)} && \text{change order of summation} \\
&= \frac{1}{N} \sum_{s=0}^{N-1} \bar{x}[s] e^{jk\frac{2\pi}{N}s} \left(\sum_{r=0}^{N-1} \bar{x}[s+r] e^{-jk\frac{2\pi}{N}(s+r)} \right) && \text{substitute } u = s + r \\
&= \frac{1}{N} \underbrace{\sum_{s=0}^{N-1} \bar{x}[s] e^{jk\frac{2\pi}{N}s}}_{X[-k]} \cdot \underbrace{\sum_{u=s}^{N-1+s} \bar{x}[u] e^{-jk\frac{2\pi}{N}(u)}}_{X[k]} && \text{periodic time series} \\
&= \frac{1}{N} X^*[k] \cdot X[k] && \text{q.e.d.}
\end{aligned}
$$

Statistical properties of the periodogram

The periodogram is an **unbiased estimate** of the PSD:

$$\boxed{\lim_{N\to\infty} \mathsf{E}\left\{I_{N\overline{x}\overline{x}}[k]\right\} = S_{\overline{x}\overline{x}}[k]} \tag{4.26}$$

Hence, when the observation time increases, and we have more samples, then the estimate of the PSD improves.

The periodogram is **not a consistent estimate** of the PSD:

$$\boxed{\lim_{N\to\infty} var\{I_{N\overline{x}\overline{x}}[k]\} = \sigma^4_{\overline{x}} \neq 0} \tag{4.27}$$

The variance of the periodogram does not approach zero when the number of measurements N increases, not even when we have an infinite number of measurements!

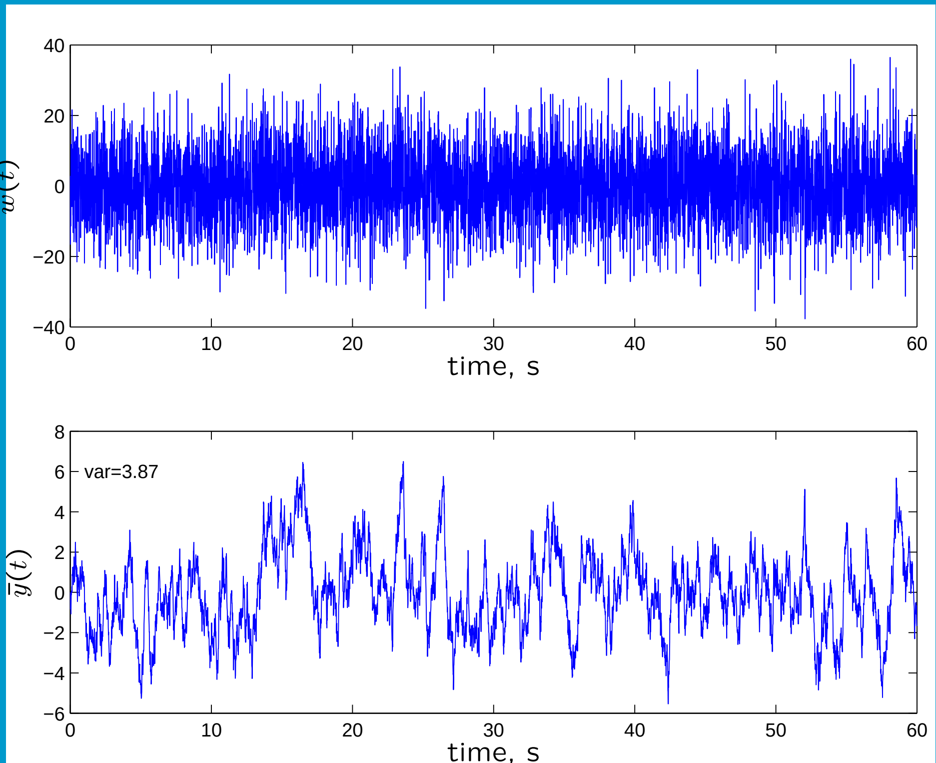
These properties are further elaborated on in the lecture notes.

Let it be clear, though, that the fastest way to obtain the spectral densities is to calculate them directly from the discrete time series using the FFT.

Even if we are only interested in the covariance function, it is faster to first compute the PSD and then use an inverse FFT (IFFT) to transform it back to the time domain.

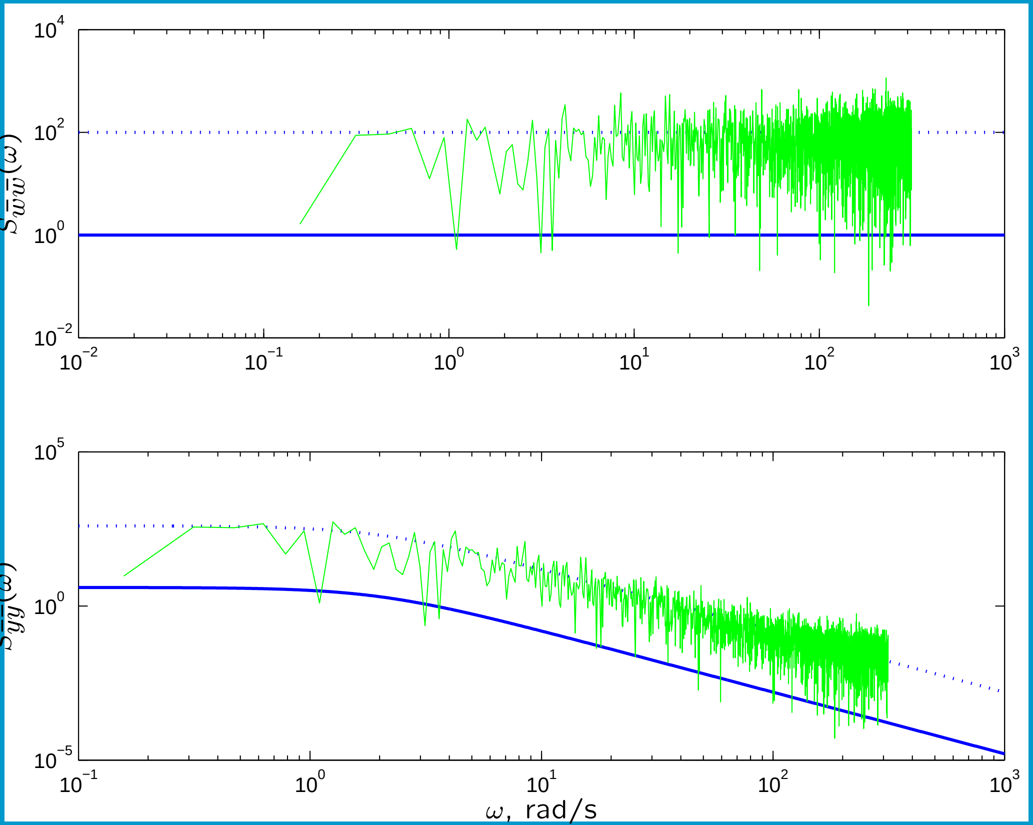
Matlab example of spectral estimate

Consider the output $\bar{y}(t)$ of CT system $H(\omega) = \frac{K}{1+j\omega\tau}$ ($K = 2$, $\tau = 0.5$) driven by white noise $\bar{w}(t)$.



The analytical variance of $\bar{y}(t)$ would be 4 (Table 3.5). In this particular realization it equals 3.87.

Now we take the last forty seconds (sample frequency 100 Hz, $N = 4000$ samples) of this response and compute the signal spectrum using the periodogram. We compare it with the analytical power spectra $S_{\bar{w}\bar{w}}(\omega)$ and $S_{\bar{y}\bar{y}}(\omega)$, which equal 1 ($\forall\ \omega$) and $|H(\omega)|^2$, respectively.



The blue continuous lines show the analytical spectra, $S_{\bar{w}\bar{w}}(\omega)$ and $S_{\bar{y}\bar{y}}(\omega)$.

The green scattered lines show the PSD estimate using the periodogram, $\hat{S}_{\bar{w}\bar{w}}[k]$ and $\hat{S}_{\bar{y}\bar{y}}[k]$.

The blue dotted lines show the analytical spectra, multiplied with $\frac{1}{\triangle t}$.

Note that $\omega_s/2 = 2\pi \cdot 100/2 = 100\pi$ rad/s.

Our mistake has now become clear. With the power spectral density estimate using the periodogram of a DFT-ed discrete-time sequence we have obtained an estimate of the discrete-time PSD.

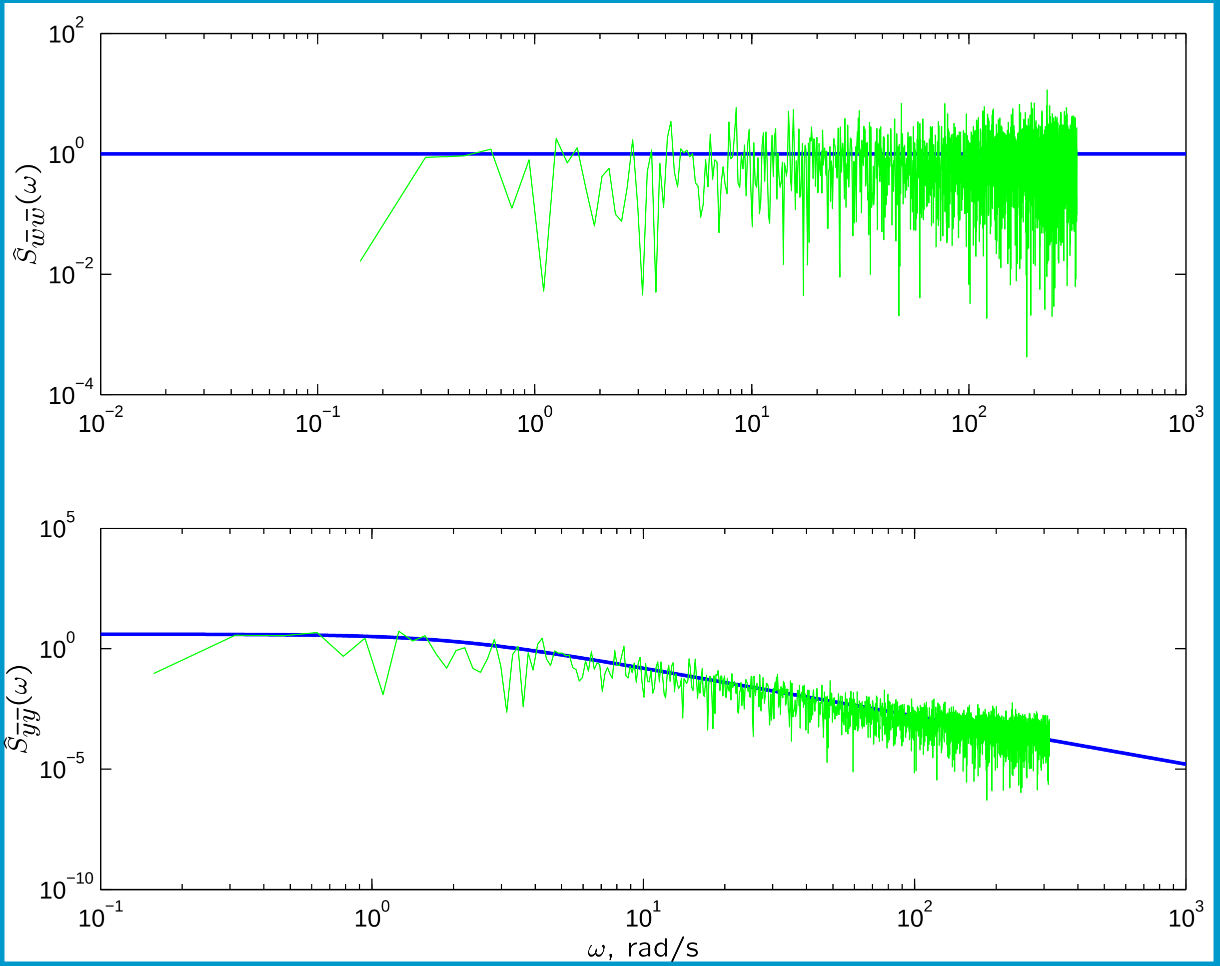
Remember that the DTFT (and thus the DFT) of a discrete-time series that represents samples of a continuous-time process, the discrete-time Fourier transform equals the continuous-time Fourier transform **scaled with** $\frac{1}{\triangle t}$, and repeated in discrete frequency.

Multiplying the discrete-time estimate of the PSD with this sample time $\triangle t$ then gives us the estimate of the continuous-time PSD:

$$\boxed{\widehat{S}_{\overline{y}\overline{y}}(\omega) \;=\; \triangle t \cdot \widehat{S}_{\overline{y}\overline{y}}[k]}$$

Note that we are dealing primarily with CT systems in our domain (a/c, controllers, humans...)

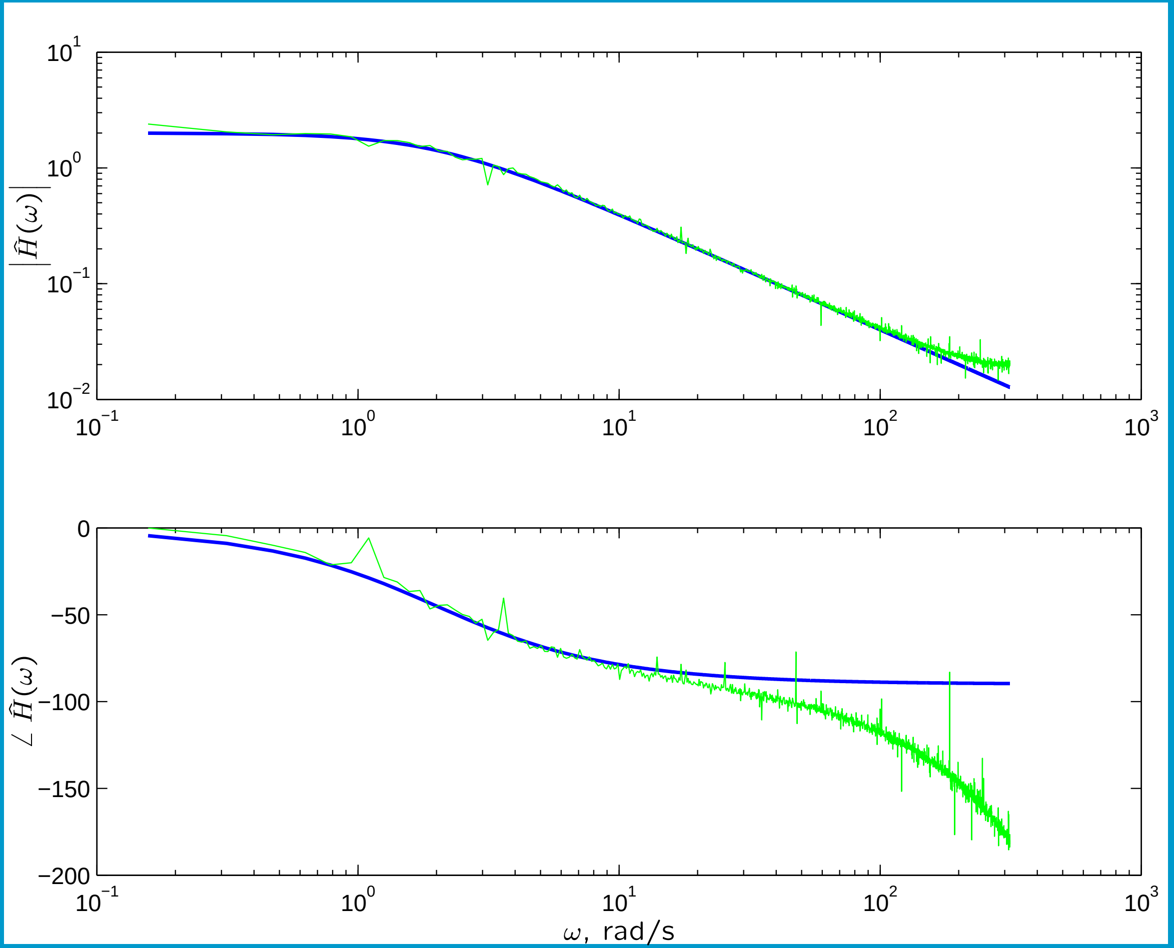
Scaling the DT PSD with $\triangle t$ indeed yields the estimate of the CT PSD:



The blue continuous lines show the analytical spectra, $S_{\overline{w}\overline{w}}(\omega) = W$ and $S_{\overline{y}\overline{y}}(\omega) = |H(\omega)|^2$.

The green scattered lines show the scaled PSD estimates using the periodogram, $\triangle t \cdot \widehat{S}_{\overline{w}\overline{w}}[k]$ (top) and $\triangle t \cdot \widehat{S}_{\overline{y}\overline{y}}[k]$ (bottom).

The system FRF $H(\omega)$ can be estimated through computing the quotient of the cross-spectral density $\hat{S}_{\bar{u}\bar{y}}(\omega)$ and the auto-spectral density $\hat{S}_{\bar{u}\bar{u}}(\omega)$:



The blue continuous lines show the analytical system FRF, $|H(\omega)|$ and $\angle H(\omega)$.

The green scattered lines show the estimate of $H(\omega)$, $\hat{H}(\omega)$:

$$\hat{H}(\omega) = \frac{\hat{S}_{\bar{u}\bar{y}}(\omega)}{\hat{S}_{\bar{u}\bar{u}}(\omega)}$$

(where $\bar{u}(t)$ is white noise)

Note the error in the phase at higher frequencies!! How come??

Smoothed estimators for PSD functions

A straightforward approach in reducing the variance of the periodogram is to average over a number of independent estimates. E.g., one can average the PSDs calculated for a number of realizations of the same stochastic process. Or one can average the PSDs calculated for a number of *segments* of the same realization.

The **Bartlett** approach means that we are dividing our realization of N samples into K segments of M samples each (i.e., $N = K \cdot M$). We compute the K periodograms for each of these segments (using M samples), $I_M^{(i)}[k]$ for $i = 1 \ldots K$, and then take the average of these K periodograms as the estimate for the PSD:

$$\widehat{S}_{\overline{x}\overline{x}}[k] = \frac{1}{K} \sum_{i=1}^{K} I_M^{(i)}[k] \qquad (4.38)$$

The variance of this estimate for the PSD is then:

$$var\{\widehat{S}_{\overline{x}\overline{x}}[k]\} = \frac{1}{K} var\{I_M[k]\} \quad (4.39)$$

So, when N is constant, when the number of periodograms K increases the variance of the spectrum estimate indeed decreases, a nice result. Keep in mind, however, that when K increases for a fixed N, the number of samples M on which the computation of the periodogram is based must decrease as well ($N = M \cdot K$). Recall that:

$$I_M^{(i)}[k] = \frac{1}{M} \left| \sum_{n=0}^{M-1} x^{(i)}[n] e^{-jk\frac{2\pi}{M}n} \right|^2,$$

for each data segment $x^{(i)}[n]$. The **resolution** of the spectrum frequencies is equal to $\frac{2\pi}{M}$, so when M decreases, so does the spectrum resolution!

In other words, we have a trade-off between the variance of our estimate (which decreases for larger K) and the resolution of our spectrum (which increases for larger M).

The lecture notes discuss other ways to smooth the spectrum estimate, like the **Welch** method and using **Hanning** windows, this time not on the time series but on the estimate of the periodogram.

Other, even simpler methods exist to smooth the PSD estimates, based on averaging the PSD estimate *in the frequency domain*, i.e., averaging over a number of frequency bands. These methods are discussed in more detail in other lectures (see e.g., WB2301).