4CM00: Control Engineering

Frequency response measurements

Dr.ir. Gert Witvoet

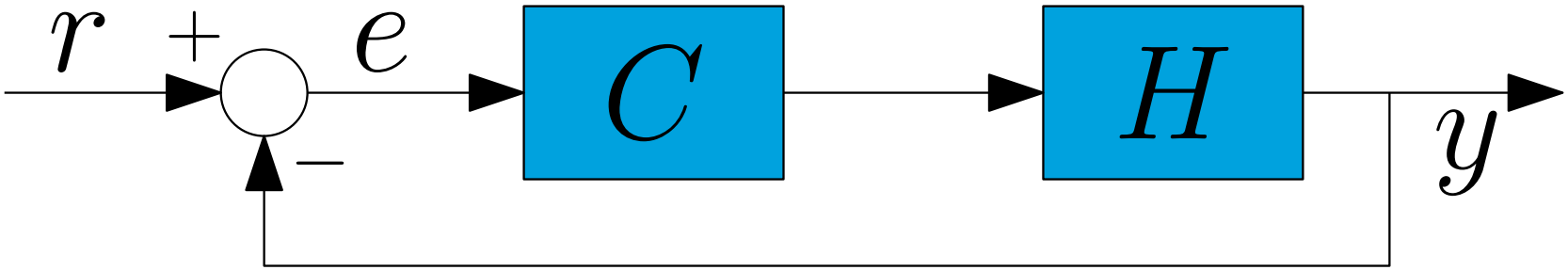
Technische Universiteit
Eindhoven
University of Technology

September 2020

Where innovation starts

The context

Goal of this course:



Given: the plant $H(s)$

Design: the controller $C(s)$

The design of $C(s)$ depends on the specific characteristics of $H(s)$

- But how to design $C(s)$ if $H(s)$ is unknown?

The context: model-based

Approach 1: Make a model based on physics, e.g. $H(s) = \frac{1}{ms^2+ds+k}$

The context: model-based

Approach 1: Make a model based on physics, e.g. $H(s) = \frac{1}{ms^2+ds+k}$

Problem:
How to derive models of complex systems?

The context: model-based

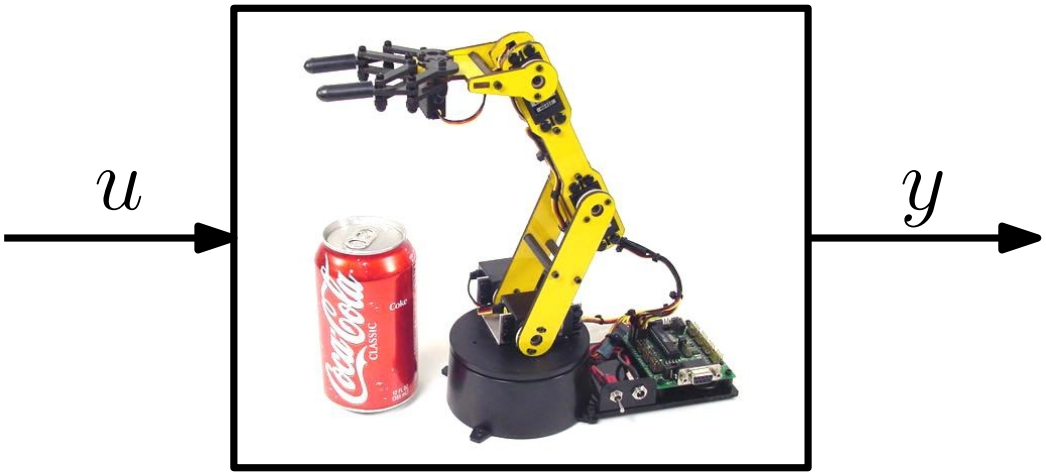
Approach 1: Make a model based on physics, e.g. $H(s) = \frac{1}{ms^2+ds+k}$

Problem:
How to derive models of complex systems?

Problem:
How to determine system parameter values?

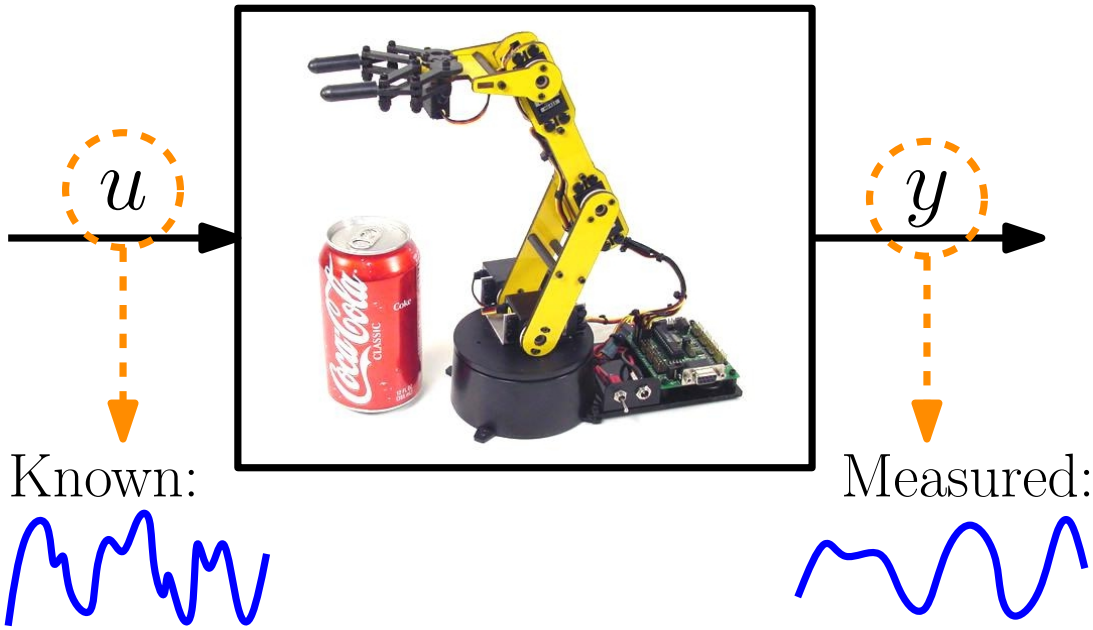
The context: data-based

Approach 2: Identify the system by measuring it



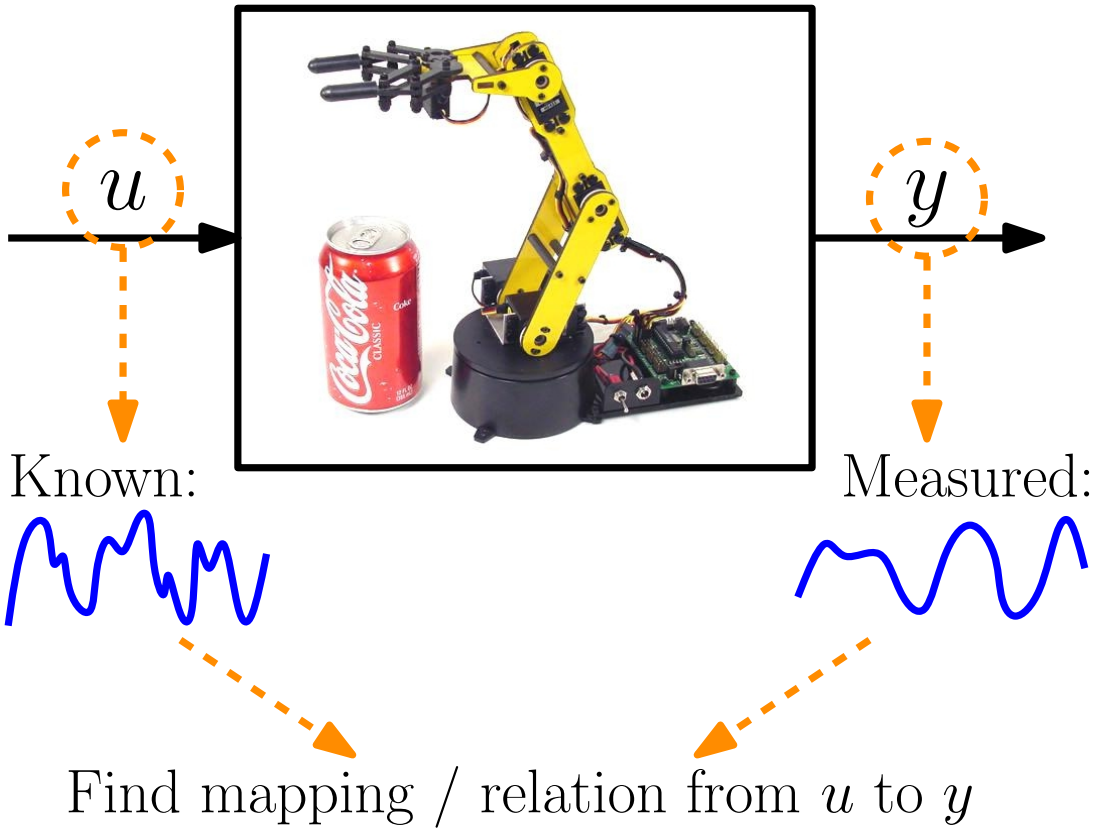
The context: data-based

Approach 2: Identify the system by measuring it



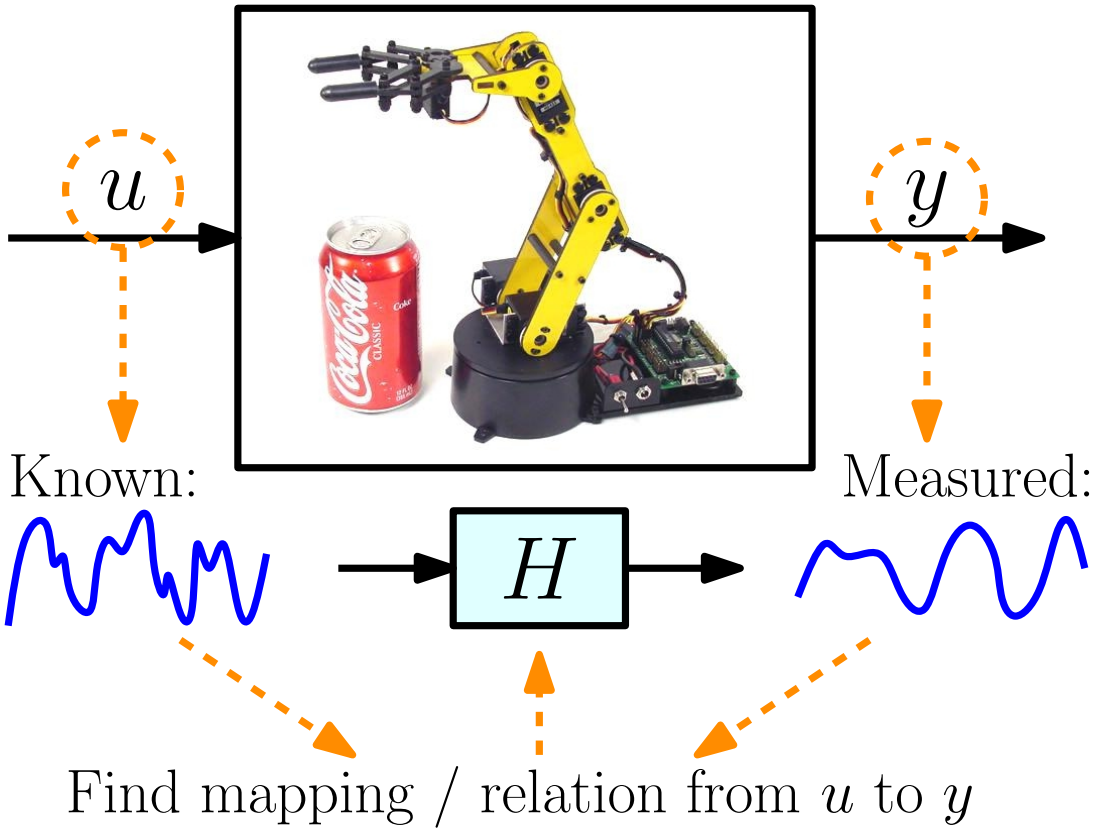
The context: data-based

Approach 2: Identify the system by measuring it



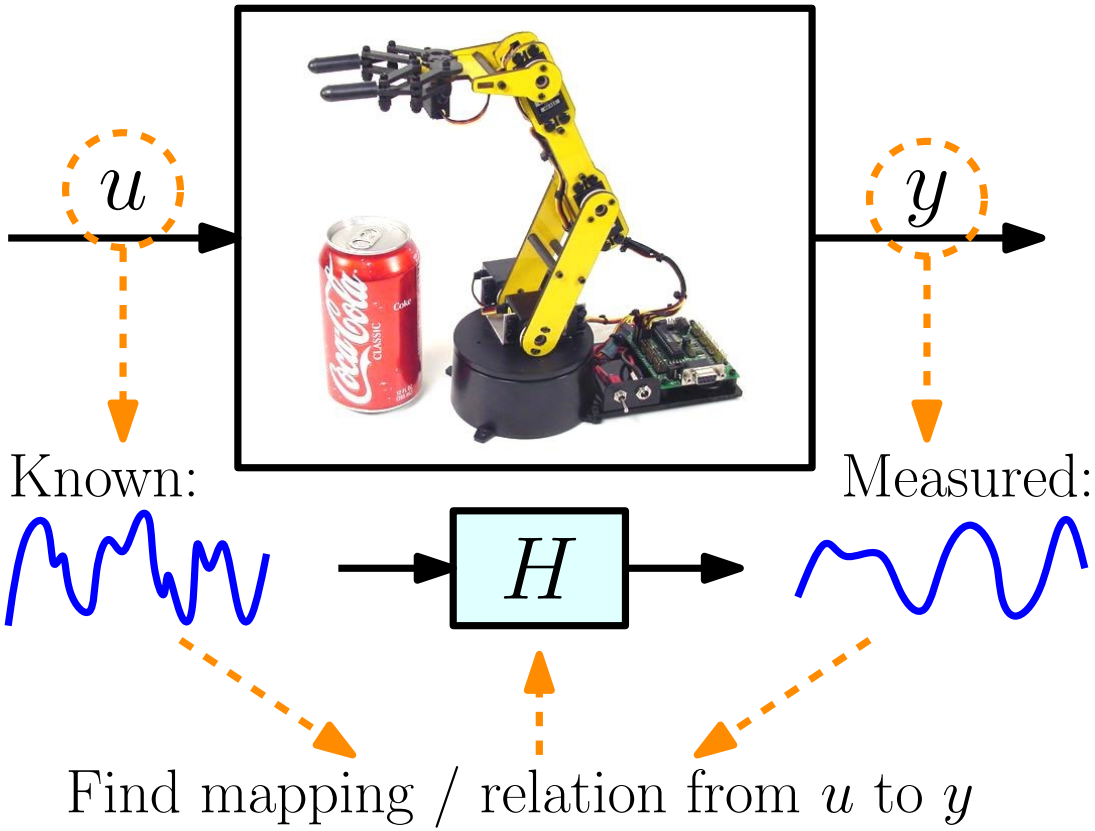
The context: data-based

Approach 2: Identify the system by measuring it



The context: data-based

Approach 2: Identify the system by measuring it



Disadvantage: Obtained parameters do not have any physical meaning

- Irrelevant for controller design

The context: identification

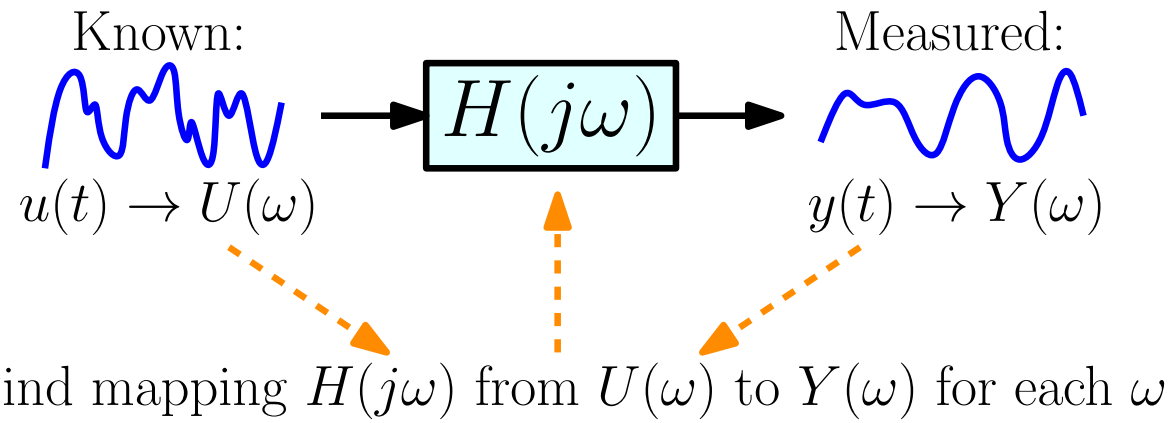
There are various data-based identification techniques:

- Prediction error identification
- Subspace identification
- Minimal realization techniques
- Frequency response measurements
- ...

The context: identification

There are various data-based identification techniques:

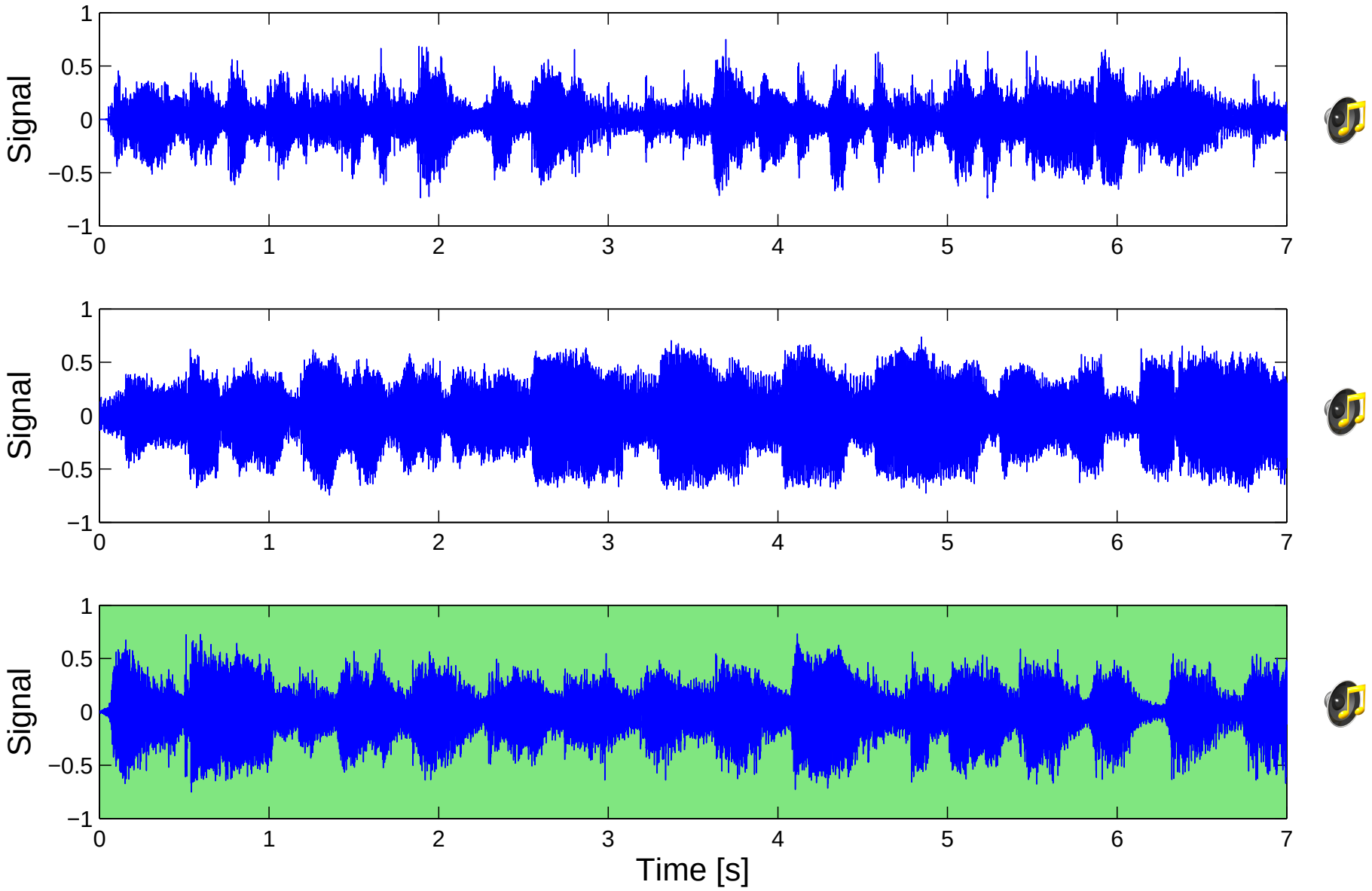
- Prediction error identification
- Subspace identification
- Minimal realization techniques
- Frequency response measurements
- ...



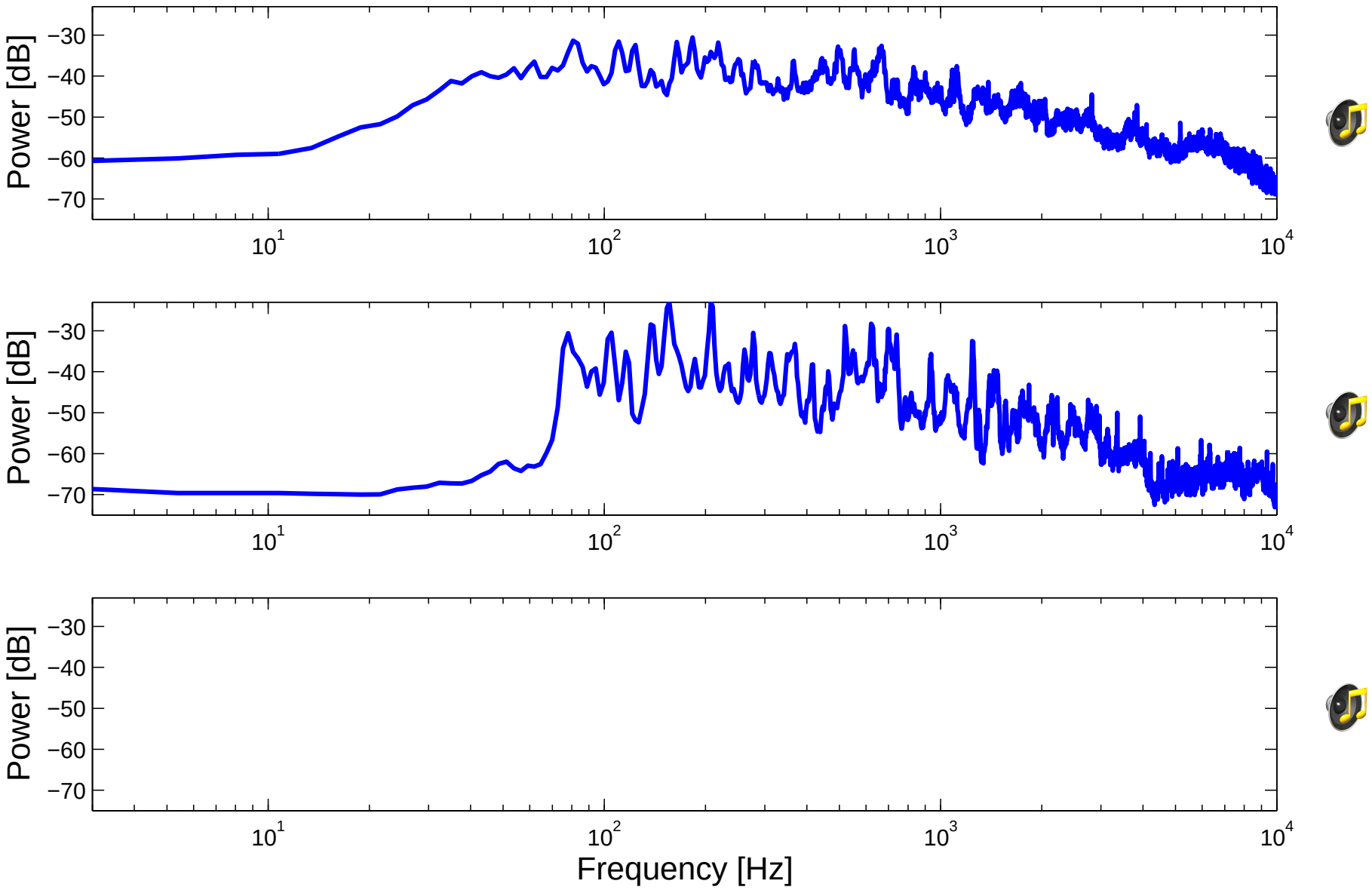
Note: Only applicable for linear systems!

Why frequency domain?

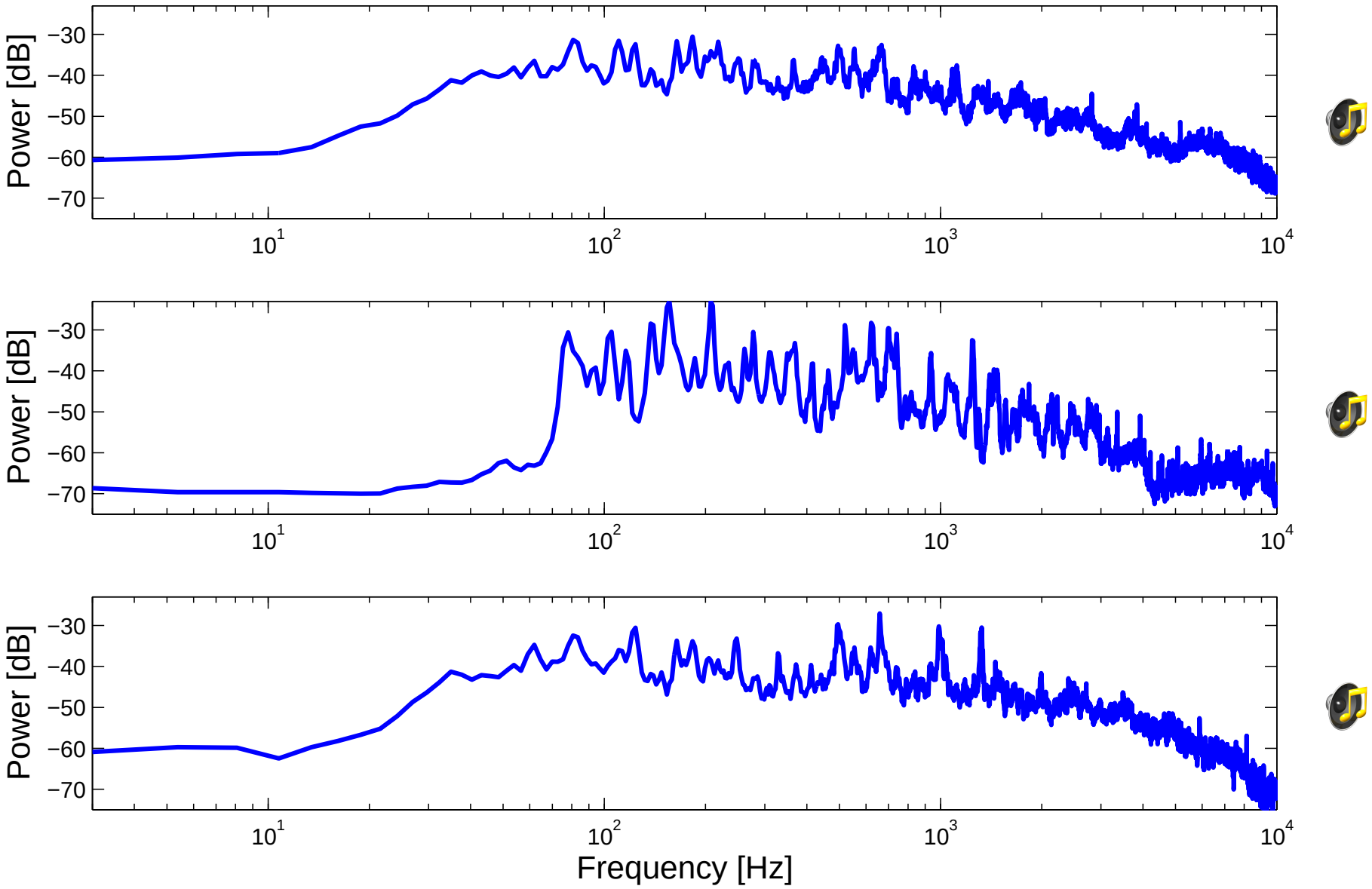
Time vs frequency



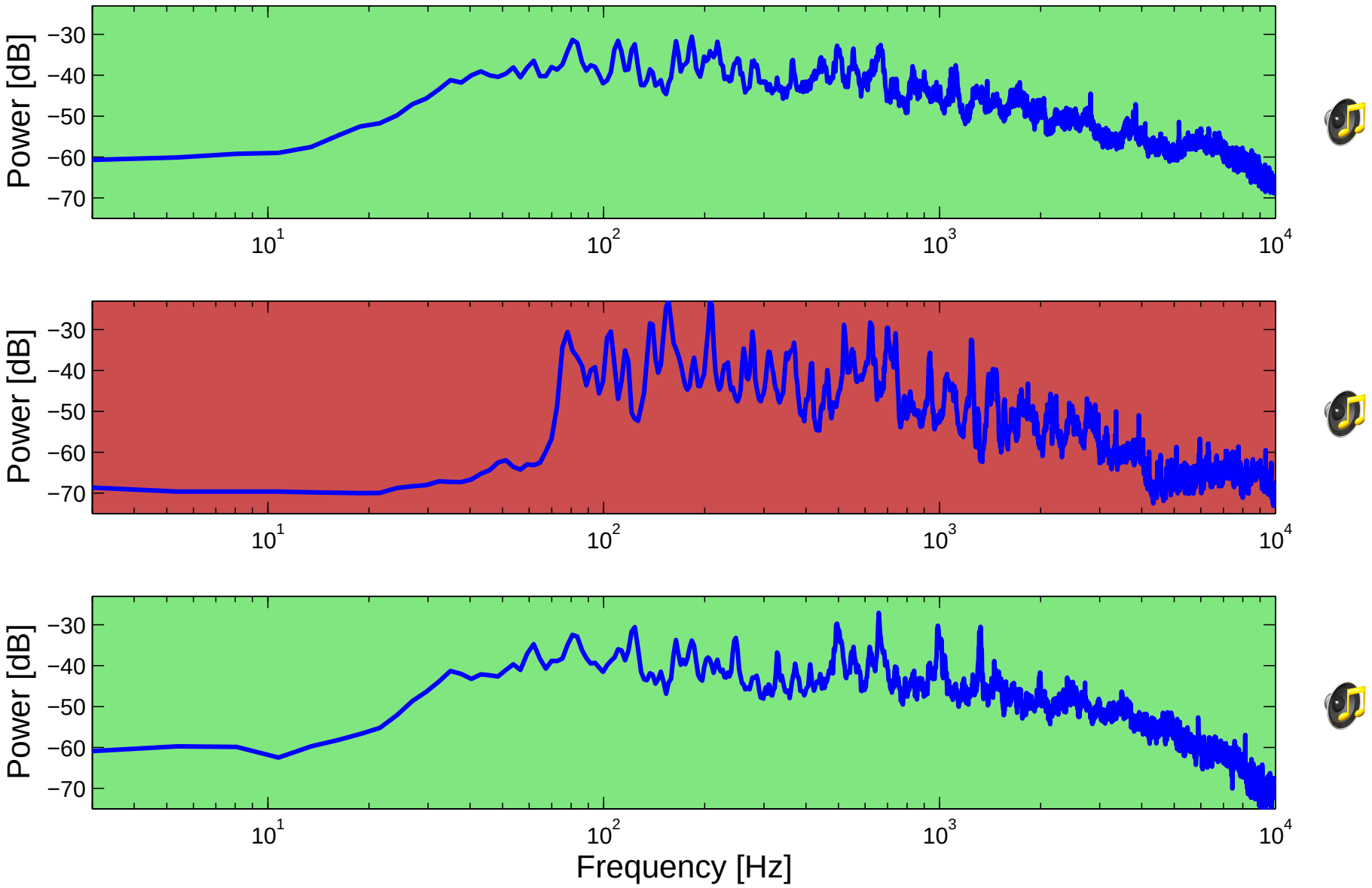
Time vs frequency



Time vs frequency



Time vs frequency



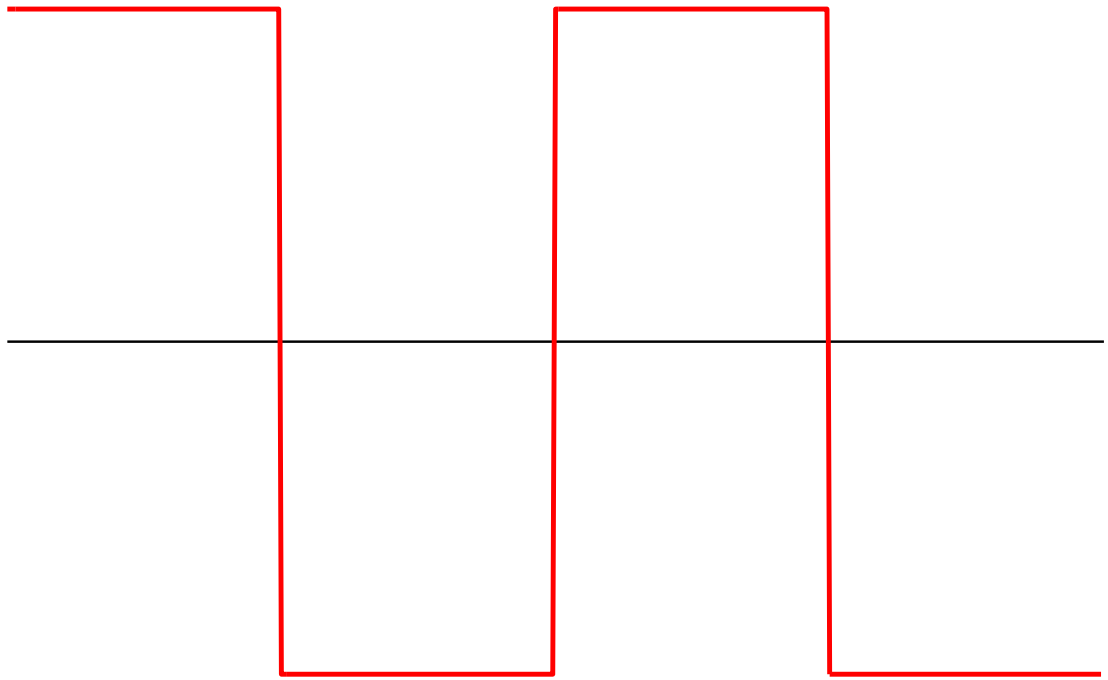
How to transform time into frequency?

Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

Joseph Fourier

Example: square wave

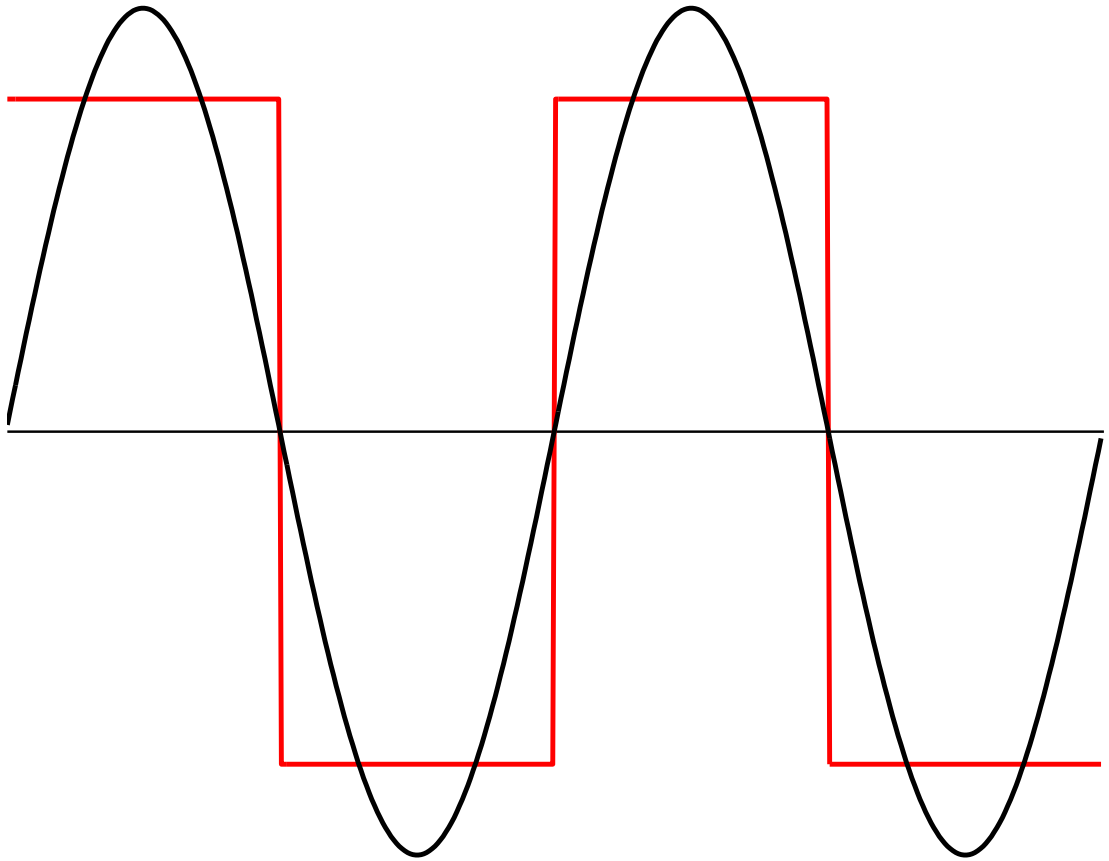


Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

Joseph Fourier

Example: square wave

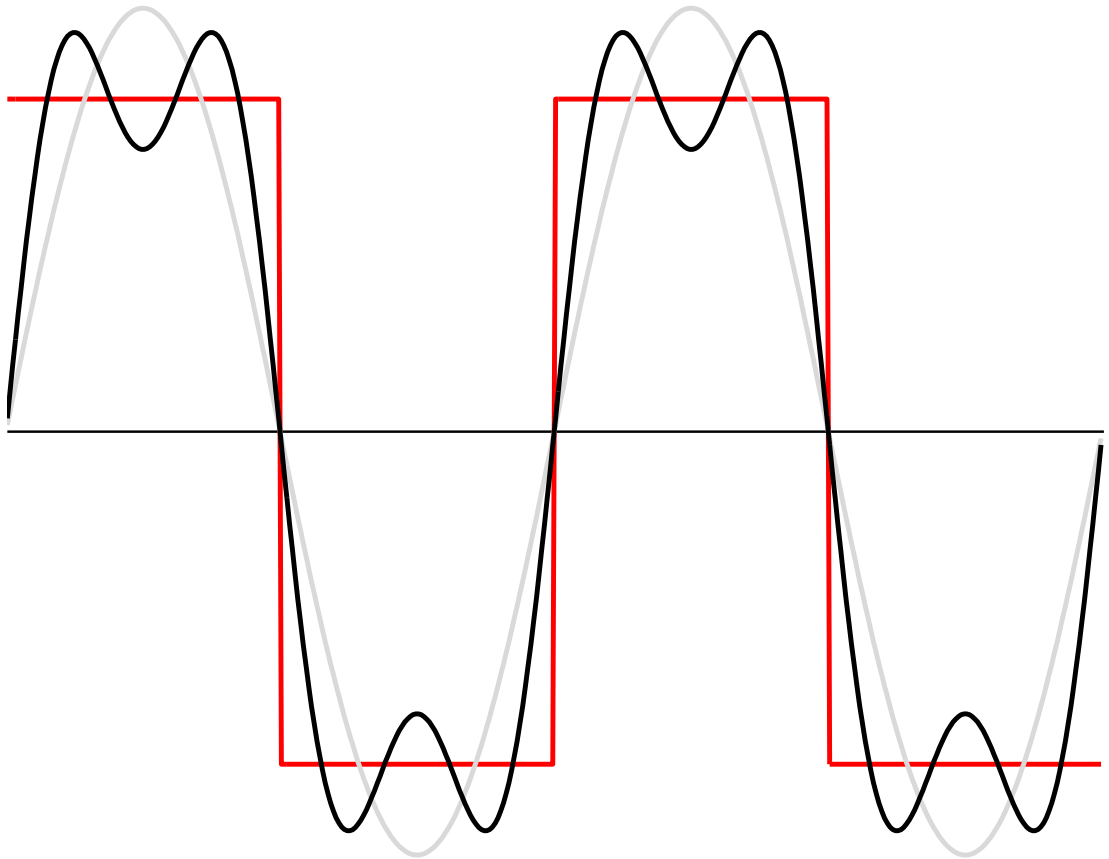


Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

Joseph Fourier

Example: square wave

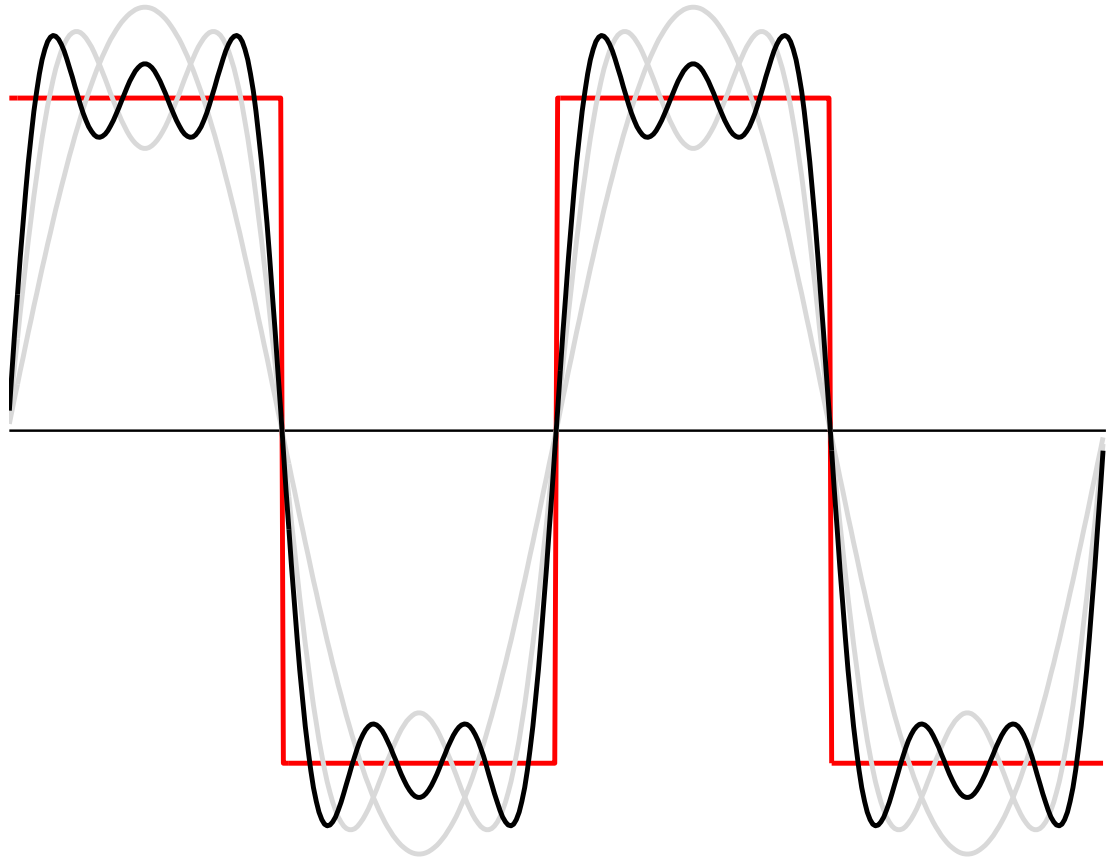


Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

Joseph Fourier

Example: square wave

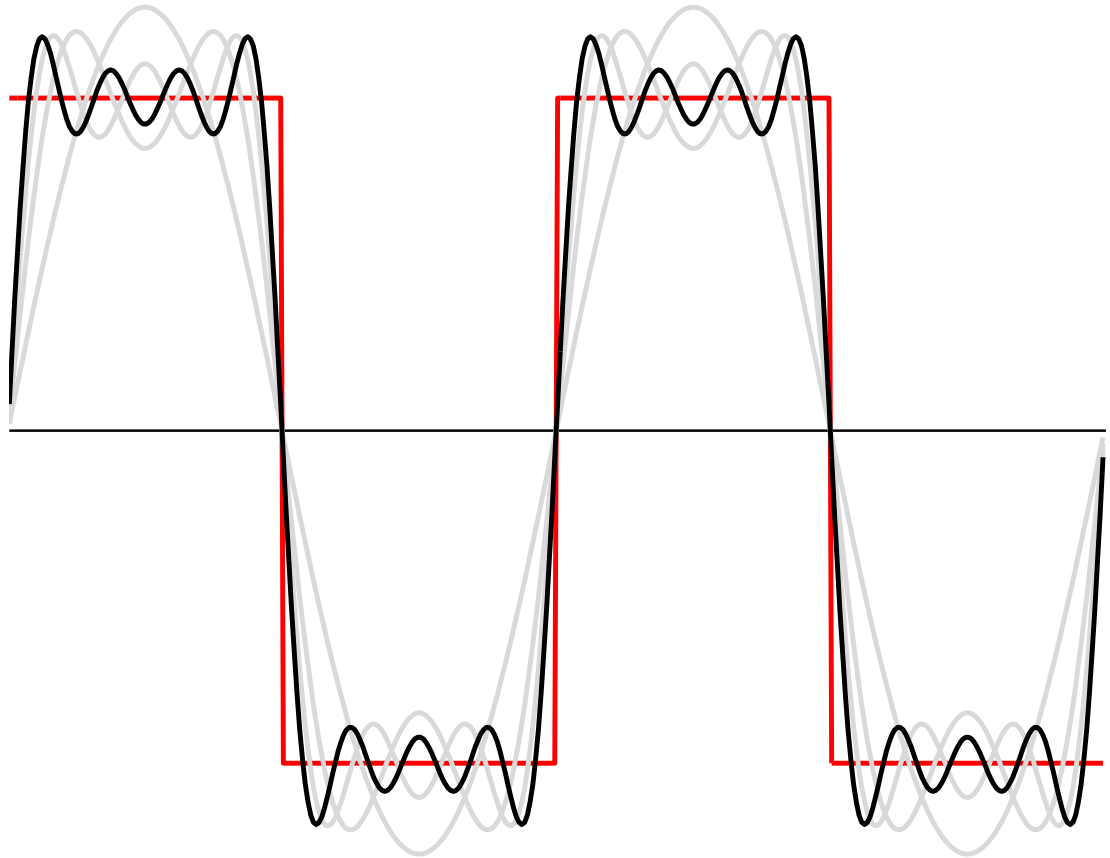


Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

Joseph Fourier

Example: square wave

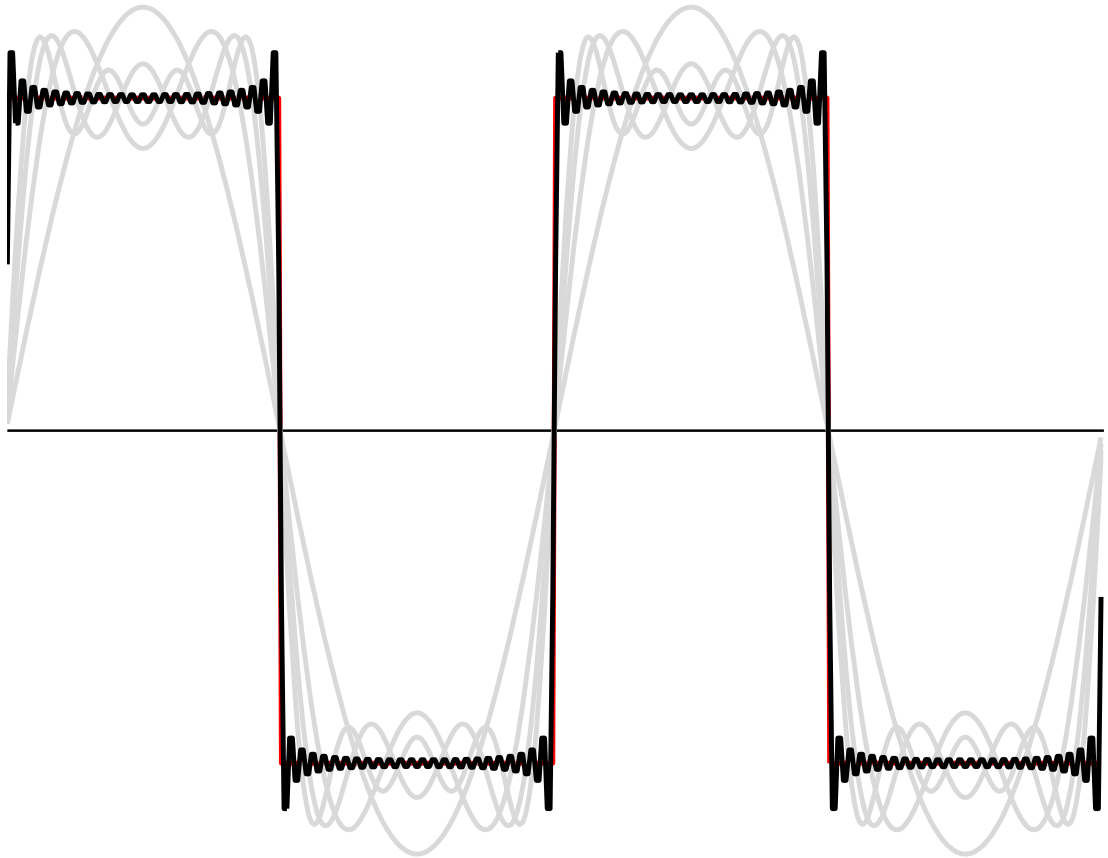


Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

Joseph Fourier

Example: square wave

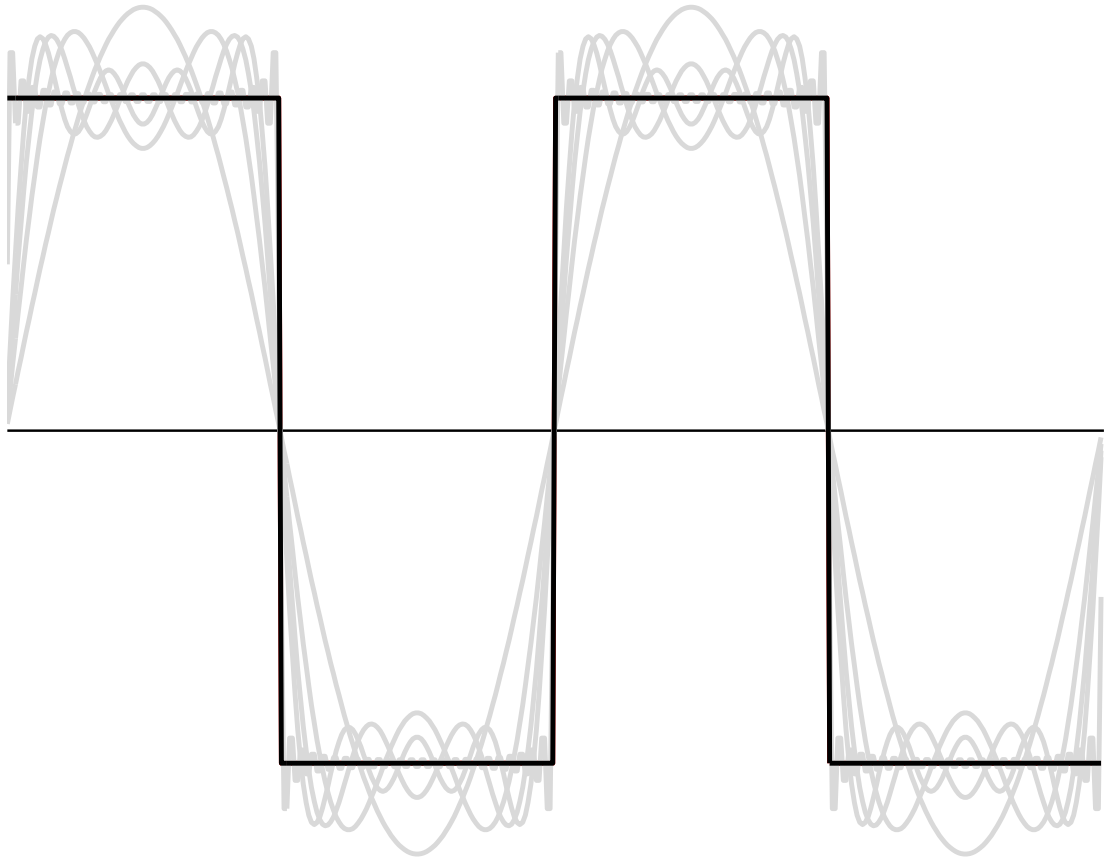


Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

Joseph Fourier

Example: square wave



Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

$$x(t) = \sum_{k=0}^{\infty} \left[a_k \cos\left(2\pi \tfrac{k}{T} t\right) + b_k \sin\left(2\pi \tfrac{k}{T} t\right) \right] \quad (1)$$

$$= A_0 + \sum_{k=1}^{\infty} A_k \cos\left(k \tfrac{2\pi}{T} t + \phi_k\right) \quad (2)$$

Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

$$x(t) = \sum_{k=0}^{\infty} \left[a_k \cos\left(2\pi \tfrac{k}{T} t\right) + b_k \sin\left(2\pi \tfrac{k}{T} t\right) \right] \quad (1)$$

$$= A_0 + \sum_{k=1}^{\infty} A_k \cos\left(k \tfrac{2\pi}{T} t + \phi_k\right) \quad (2)$$

where

$\frac{2\pi}{T}$: Fundamental frequency (period T)

k : Harmonic number

$A_0 = a_0$: Constant signal offset

$A_k = \sqrt{a_k^2 + b_k^2}$: Amplitude of k-th harmonic

$\phi_k = -\arctan \frac{b_k}{a_k}$: Phase of k-th harmonic

Note: assumes the signal is periodic: $x(t + T) = x(t)$

Signal analysis: Fourier series

Any signal $x(t)$ of length T can be represented as a sum of harmonics.

$$x(t) = \sum_{k=0}^{\infty} \left[a_k \cos\left(2\pi \tfrac{k}{T} t\right) + b_k \sin\left(2\pi \tfrac{k}{T} t\right) \right]$$

Let's try to rewrite this equation.

Using Euler's formula

$$e^{jt} = \cos(t) + j \sin(t), \tag{3}$$

we can write

$$\cos\left(2\pi \tfrac{k}{T} t\right) = \tfrac{1}{2} \left(e^{j2\pi \frac{k}{T} t} + e^{-j2\pi \frac{k}{T} t} \right) \tag{4a}$$

$$\sin(2\pi \tfrac{k}{T} t) = -j \tfrac{1}{2} \left(e^{j2\pi \frac{k}{T} t} - e^{-j2\pi \frac{k}{T} t} \right). \tag{4b}$$

Then $x(t)$ can be written as

$$\begin{aligned} x(t) &= \sum_{k=0}^{\infty} \left[\tfrac{1}{2} a_k \left(e^{j2\pi \frac{k}{T} t} + e^{-j2\pi \frac{k}{T} t} \right) - j \tfrac{1}{2} b_k \left(e^{j2\pi \frac{k}{T} t} - e^{-j2\pi \frac{k}{T} t} \right) \right] \\ &= \sum_{k=0}^{\infty} \left[\tfrac{1}{2} \left(a_k - j b_k \right) e^{j2\pi \frac{k}{T} t} + \tfrac{1}{2} \left(a_k + j b_k \right) e^{-j2\pi \frac{k}{T} t} \right] \\ &= a_0 + \sum_{k=1}^{\infty} \left[X(k) e^{j2\pi \frac{k}{T} t} + X(-k) e^{-j2\pi \frac{k}{T} t} \right] \\ &= \sum_{k=-\infty}^{\infty} X(k) e^{j2\pi \frac{k}{T} t} \end{aligned} \tag{5}$$

where we have defined the coefficients

$$X(0) = a_0, \quad X(k) = \tfrac{1}{2}(a_k - j b_k), \quad X(-k) = \tfrac{1}{2}(a_k + j b_k).$$

Fourier series expansion

Hence, the complex form of the *Fourier series* is

$$x(t) = \sum_{k=-\infty}^{\infty} X(k) e^{j2\pi \frac{k}{T} t}. \tag{6}$$

- $X(k)$ are the *complex valued* Fourier series expansion coefficients

Fourier series expansion

Hence, the complex form of the *Fourier series* is

$$x(t) = \sum_{k=-\infty}^{\infty} X(k) e^{j2\pi \frac{k}{T} t}. \tag{6}$$

- $X(k)$ are the *complex valued* Fourier series expansion coefficients

How to find these Fourier coefficients?

- by using the 'inverse' of (6):

$$X(k) = \frac{1}{T} \int_{t_0}^{t_0+T} x(t) e^{-j2\pi \frac{k}{T} t} dt, \qquad k = 0, \pm 1, \pm 2, \ldots \tag{7}$$

Fourier series expansion

Hence, the complex form of the *Fourier series* is

$$x(t) = \sum_{k=-\infty}^{\infty} X(k) e^{j2\pi \frac{k}{T} t}. \tag{6}$$

- $X(k)$ are the *complex valued* Fourier series expansion coefficients

How to find these Fourier coefficients?

- by using the 'inverse' of (6):

$$X(k) = \frac{1}{T} \int_{t_0}^{t_0+T} x(t) e^{-j2\pi \frac{k}{T} t} dt, \qquad k = 0, \pm 1, \pm 2, \ldots \tag{7}$$

Note: the proof for (7) is based on orthogonality of sines and cosines

$$\int_{t_0}^{t_0+T} \cos(2\pi \tfrac{k}{T}) \cos(2\pi \tfrac{\ell}{T}) dt = \begin{cases} 0 & \text{for } k \neq \ell \\ \frac{T}{2} & \text{for } k = \ell \end{cases}$$

$$\int_{t_0}^{t_0+T} \cos(2\pi \tfrac{k}{T}) \sin(2\pi \tfrac{\ell}{T}) dt = 0 \quad \text{for all } k, \ell$$

Fourier series expansion: interpretation

Theory:

$$X(k) = \frac{1}{T} \int_{t_0}^{t_0+T} x(t) e^{-j2\pi \frac{k}{T} t} dt$$

In practice (Matlab):

```
X = fft(x)/length(x);
```

How to interpret these complex coefficients $X(k)$?

Fourier series expansion: interpretation

Theory:

$$X(k) = \frac{1}{T}\int_{t_0}^{t_0+T} x(t)e^{-j2\pi\frac{k}{T}t}\,dt$$

In practice (Matlab):

```
X = fft(x)/length(x);
```

How to interpret these complex coefficients $X(k)$?

$$X(k) = \tfrac{1}{2}(a_k - jb_k), \quad X(-k) = \tfrac{1}{2}(a_k + jb_k), \quad X(0) = a_0.$$

Fourier series expansion: interpretation

Theory:

$$X(k) = \frac{1}{T}\int_{t_0}^{t_0+T} x(t)e^{-j2\pi\frac{k}{T}t}\,dt$$

In practice (Matlab):

```
X = fft(x)/length(x);
```

How to interpret these complex coefficients $X(k)$?

$$X(k) = \tfrac{1}{2}(a_k - jb_k), \quad X(-k) = \tfrac{1}{2}(a_k + jb_k), \quad X(0) = a_0.$$

- meaning of amplitude and phase:

$$|X(0)| = a_0 = A_0 \tag{8a}$$

$$|X(k)| = \tfrac{1}{2}\sqrt{a_k^2 + b_k^2} = \tfrac{1}{2}A_k \tag{8b}$$

$$\arg X(k) = -\arg X(-k) = -\arctan\frac{b_k}{a_k} = \phi_k \tag{8c}$$

Fourier series expansion: interpretation

Theory:

$$X(k) = \frac{1}{T}\int_{t_0}^{t_0+T} x(t)e^{-j2\pi\frac{k}{T}t}\,dt$$

In practice (Matlab):

```
X = fft(x)/length(x);
```

How to interpret these complex coefficients $X(k)$?

$$X(k) = \tfrac{1}{2}(a_k - jb_k), \quad X(-k) = \tfrac{1}{2}(a_k + jb_k), \quad X(0) = a_0.$$

- meaning of amplitude and phase:

$$|X(0)| = a_0 = A_0 \tag{8a}$$

$$|X(k)| = \tfrac{1}{2}\sqrt{a_k^2 + b_k^2} = \tfrac{1}{2}A_k \tag{8b}$$

$$\arg X(k) = -\arg X(-k) = -\arctan\frac{b_k}{a_k} = \phi_k \tag{8c}$$

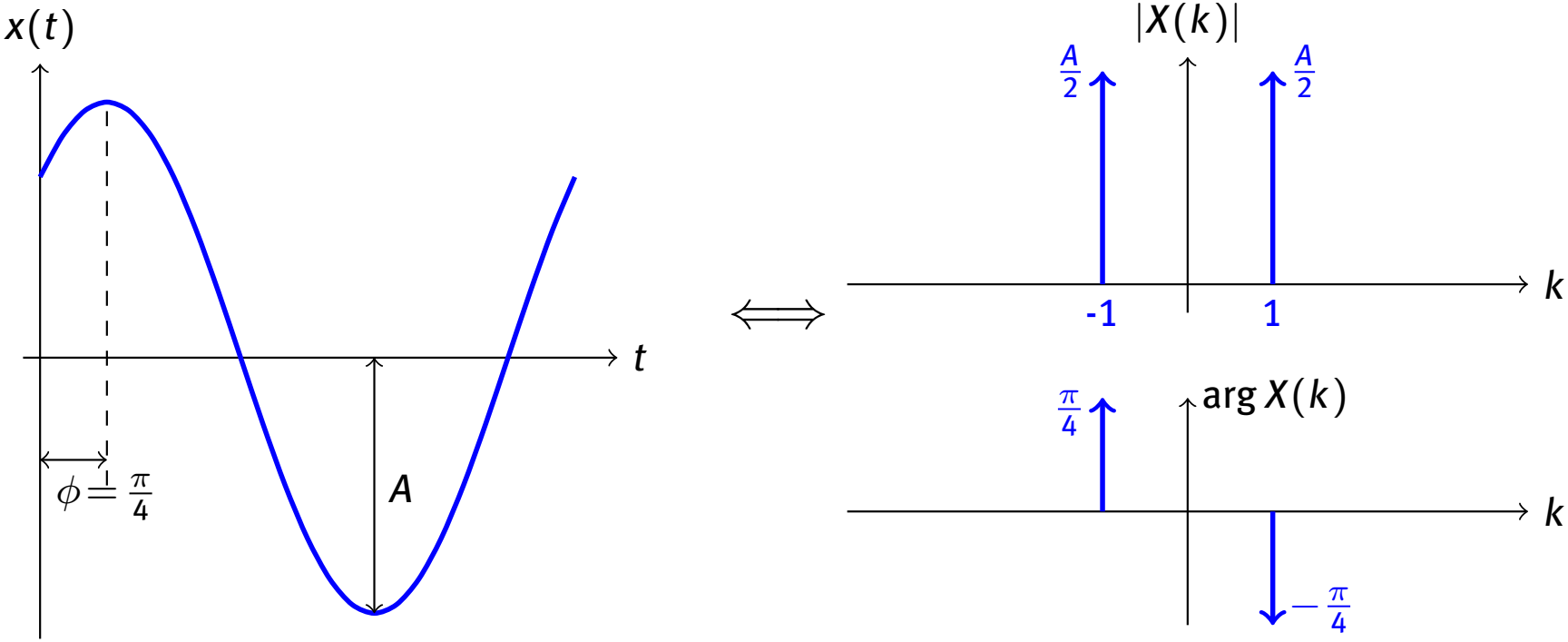
- which can be directly linked to the signal $x(t)$:

$$x(t) = A_0 + \sum_{k=1}^{\infty} A_k \cos\left(k\tfrac{2\pi}{T}t + \phi_k\right)$$

Fourier series: examples

Fourier series maps time into frequency:

Continuous time t $\Longleftrightarrow$ Discrete frequency $\frac{2\pi}{T}k$

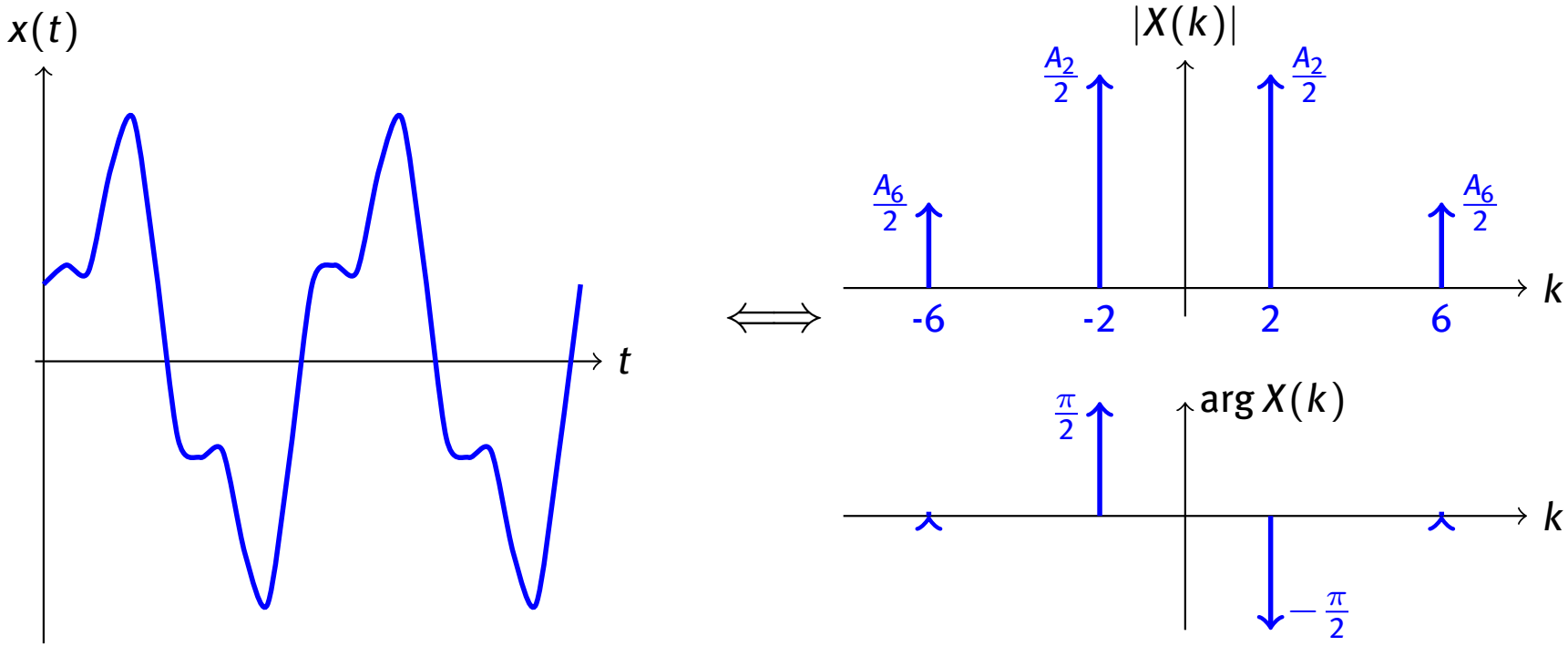


Signal $\Longleftrightarrow$ Spectrum

Fourier series: examples

Fourier series maps time into frequency:

Continuous time t $\Longleftrightarrow$ Discrete frequency $\frac{2\pi}{T}k$

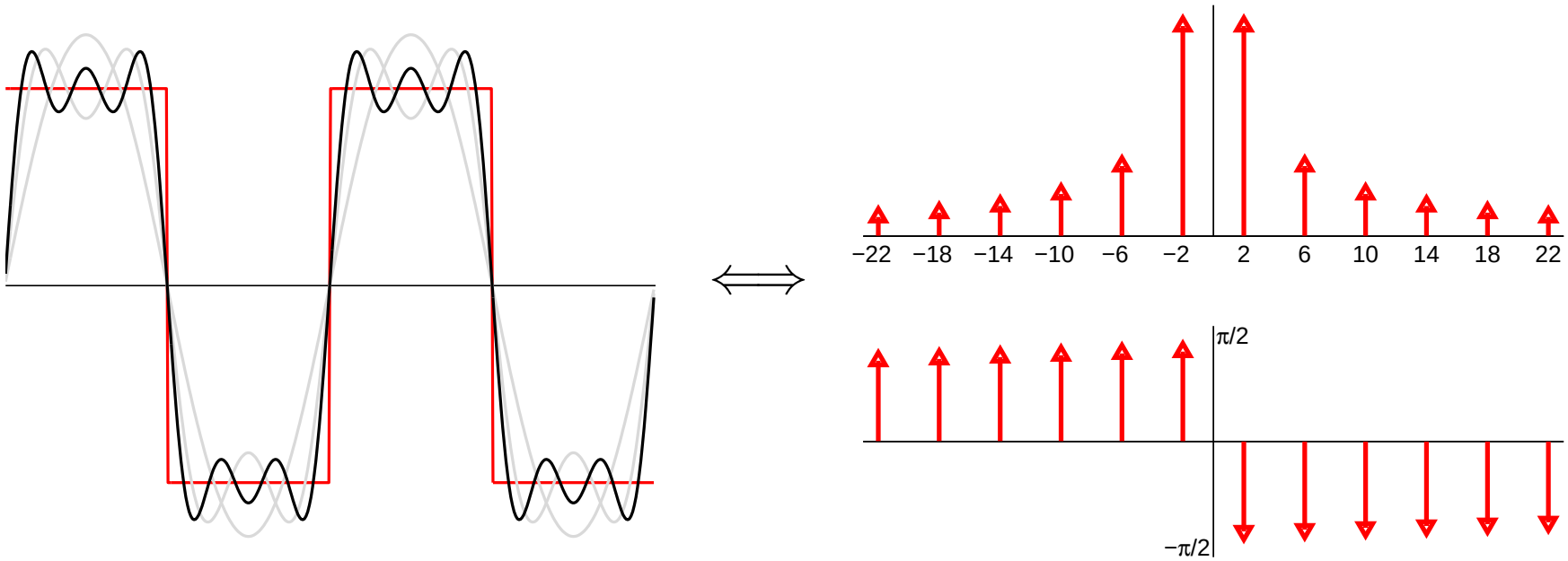


Signal $\Longleftrightarrow$ Spectrum

Fourier series: examples

Fourier series maps time into frequency:

Continuous time t $\Longleftrightarrow$ Discrete frequency $\frac{2\pi}{T}k$



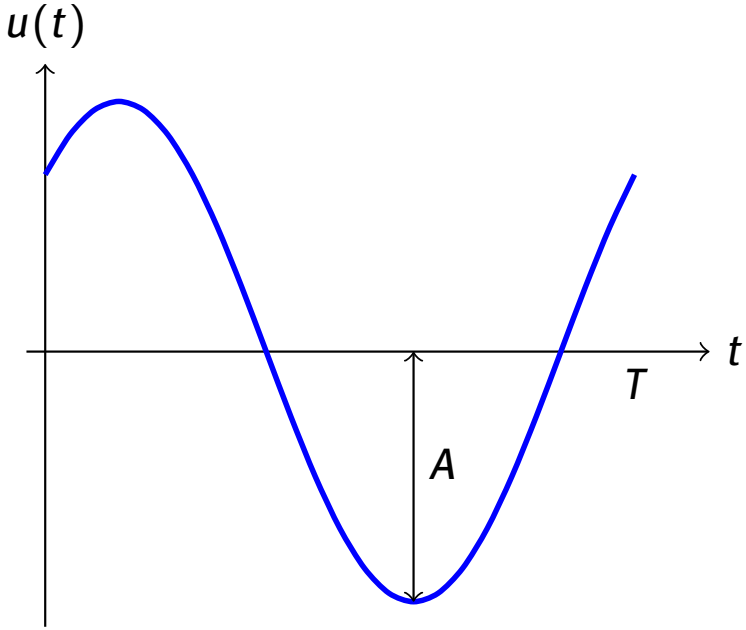
Signal $\Longleftrightarrow$ Spectrum

Fourier series, energy and power

Signal energy and power

Periodic signals have finite power P

- Power = Energy per second
- Power = RMS^2 (Root Mean Squared)

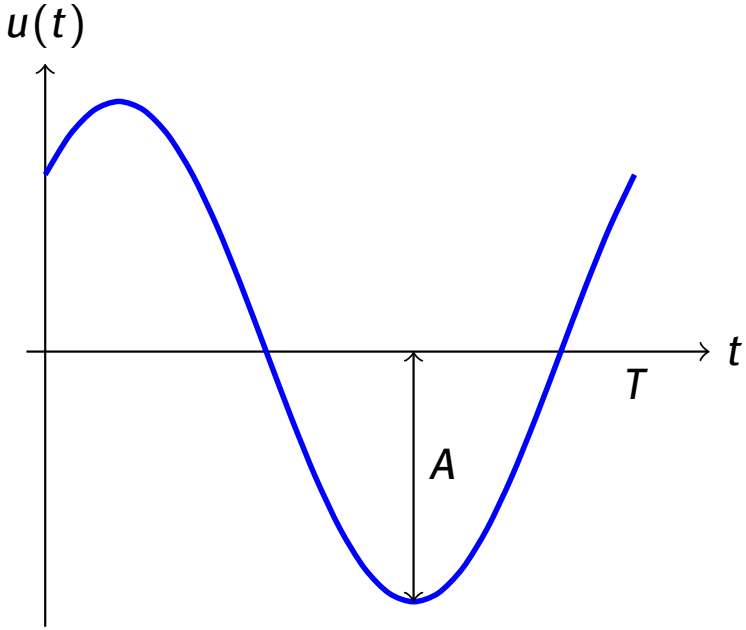


$$
\begin{aligned}
P &= \frac{1}{T}\int_0^T u^2(t)\,dt \\
&= \frac{1}{T}\int_0^T A^2\cos^2\left(\frac{2\pi}{T}t\right)dt \\
&= \frac{1}{T}\int_0^T \frac{1}{2}A^2 + \frac{1}{2}A^2\cos\left(\frac{4\pi}{T}t\right)dt \\
&= \frac{1}{2}A^2
\end{aligned}
$$

Signal energy and power

Periodic signals have finite power P

- Power = Energy per second
- Power = RMS^2 (Root Mean Squared)



$$
\begin{aligned}
P &= \frac{1}{T}\int_0^T u^2(t)\,dt \\
&= \frac{1}{T}\int_0^T A^2\cos^2\left(\frac{2\pi}{T}t\right)dt \\
&= \frac{1}{T}\int_0^T \frac{1}{2}A^2 + \frac{1}{2}A^2\cos\left(\frac{4\pi}{T}t\right)dt \\
&= \frac{1}{2}A^2
\end{aligned}
$$

Can we calculate this also from the frequency spectrum $X(k)$?

Signal energy and power

Total power in a signal $u(t)$:

$$
\begin{aligned}
P &= \frac{1}{T}\int_0^T u^2(t)dt \\
&= \frac{1}{T}\int_0^T u(t)\left[\sum_{k=-\infty}^{\infty} U(k)e^{j2\pi\frac{k}{T}t}\right]dt \\
&= \sum_{k=-\infty}^{\infty} U(k)\frac{1}{T}\int_0^T u(t)e^{j2\pi\frac{k}{T}t}dt \\
&= \sum_{k=-\infty}^{\infty} U(k)U(-k) = \sum_{k=-\infty}^{\infty} U(k)U^*(k) \qquad (9)
\end{aligned}
$$

Known as *Parseval's theorem*:

- Power in time domain = power in frequency domain

Signal energy and power

Power per frequency = power spectral *density* (PSD)

$$\begin{aligned} S_{uu}(k) &= U(k)U^*(k) \\ &= \tfrac{1}{2}(a_k - jb_k) \cdot \tfrac{1}{2}(a_k + jb_k) = \tfrac{1}{4}\left(a_k^2 + b_k^2\right) \end{aligned} \tag{10}$$

Or expressed in actual frequencies $f = \frac{k}{T}$:

$$S_{uu}(f) = U(f)U^*(f) \tag{11}$$

Signal energy and power

Power per frequency = power spectral *density* (PSD)

$$\begin{aligned} S_{uu}(k) &= U(k)U^*(k) \\ &= \tfrac{1}{2}(a_k - jb_k) \cdot \tfrac{1}{2}(a_k + jb_k) = \tfrac{1}{4}\left(a_k^2 + b_k^2\right) \end{aligned} \tag{10}$$

Or expressed in actual frequencies $f = \frac{k}{T}$:

$$S_{uu}(f) = U(f)U^*(f) \tag{11}$$

- PSD in Matlab (be aware of scaling!):
  `pwelch.m, periodogram.m`
- PSD has no phase information

Signal energy and power

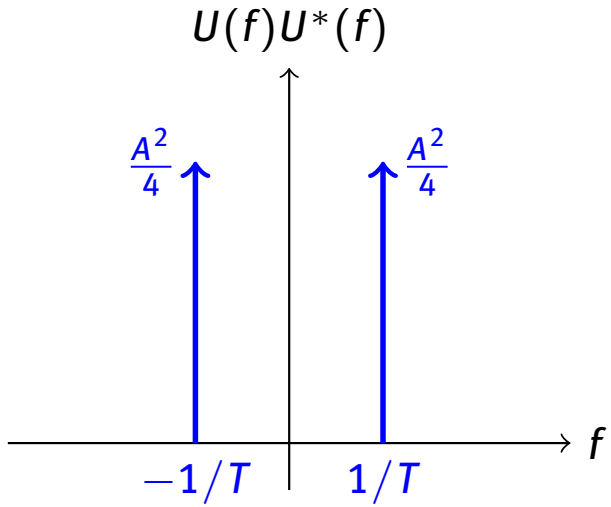
Power per frequency = power spectral *density* (PSD)

$$\begin{aligned} S_{uu}(k) &= U(k)U^*(k) \\ &= \tfrac{1}{2}(a_k - jb_k) \cdot \tfrac{1}{2}(a_k + jb_k) = \tfrac{1}{4}\left(a_k^2 + b_k^2\right) \end{aligned} \tag{10}$$

Or expressed in actual frequencies $f = \frac{k}{T}$:

$$S_{uu}(f) = U(f)U^*(f) \tag{11}$$

- PSD in Matlab (be aware of scaling!): **`pwelch.m, periodogram.m`**
- PSD has no phase information



Signal energy and power

Power per frequency = power spectral *density* (PSD)

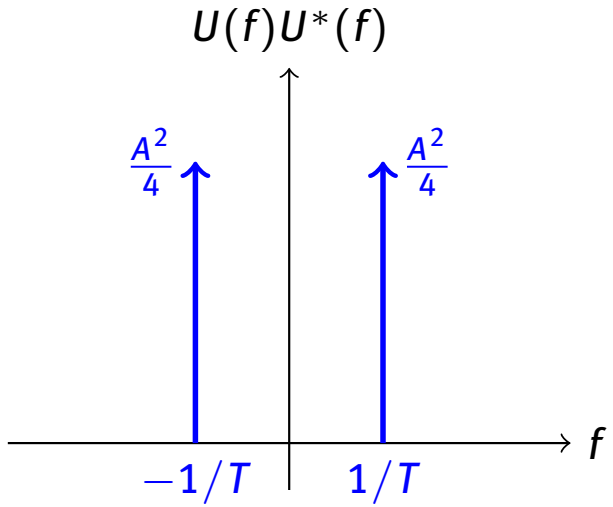
$$\begin{aligned} S_{uu}(k) &= U(k)U^*(k) \\ &= \tfrac{1}{2}(a_k - jb_k) \cdot \tfrac{1}{2}(a_k + jb_k) = \tfrac{1}{4}\left(a_k^2 + b_k^2\right) \end{aligned} \tag{10}$$

Or expressed in actual frequencies $f = \frac{k}{T}$:

$$S_{uu}(f) = U(f)U^*(f) \tag{11}$$

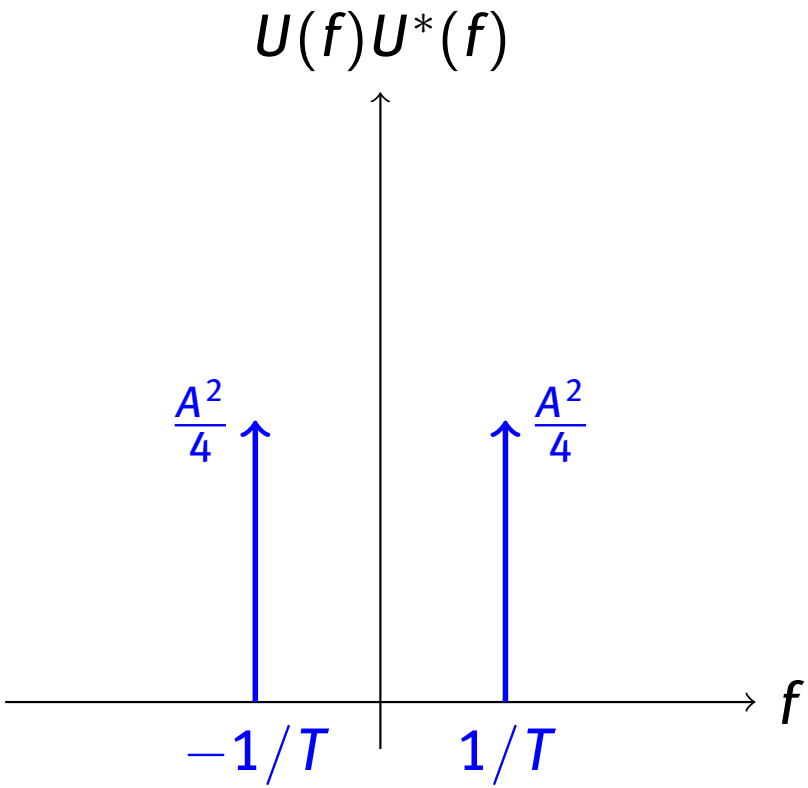
- PSD in Matlab (be aware of scaling!): **`pwelch.m, periodogram.m`**
- PSD has no phase information
- Total power = sum of PSD over all f:

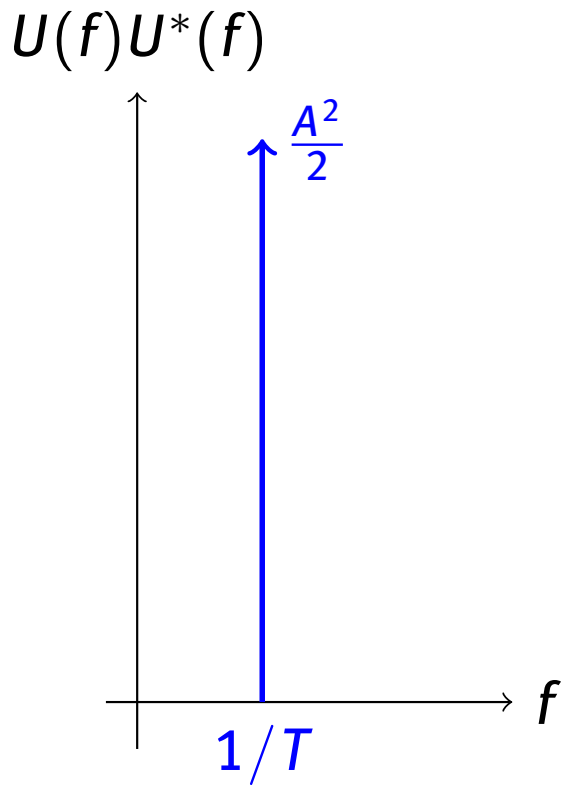
$$P = \sum U(f)U^*(f) = \frac{A^2}{4} + \frac{A^2}{4} = \frac{1}{2}A^2$$



Positive and negative frequencies

Spectral densities can be either *twosided* or *onesided*:





- Negative frequencies are added to positive frequencies
- Known as *Auto* Power Spectral Density

Common power between signals

So the product of the Fourier series $U(f)$ and $U^*(f)$ of a single signal $u(t)$ returns the energy within a single signal per frequency.

$$S_{uu}(f) = U(f)U^*(f)$$

Common power between signals

So the product of the Fourier series $U(f)$ and $U^*(f)$ of a single signal $u(t)$ returns the energy within a single signal per frequency.

$$S_{uu}(f) = U(f)U^*(f)$$

Then the product of the Fourier series of different signals $u(t)$ and $v(t)$ returns their *common* power per frequency.

$$S_{uv}(f) = U(f)V^*(f)$$

- Known as *Cross* Power Spectral Density (CPSD)
- can indicate 'common power' between the signals
- signals could be related if this common power repeats

Cross power density

Cross Power Spectral Density (CPSD):

$$S_{uv}(f) = U(f)V^*(f)$$

Interpretation:

- For each frequency: how much power of u is present in v?
- Can indicate (cross-)covariance between u and v

Cross power density

Cross Power Spectral Density (CPSD):

$$S_{uv}(f) = U(f)V^*(f)$$

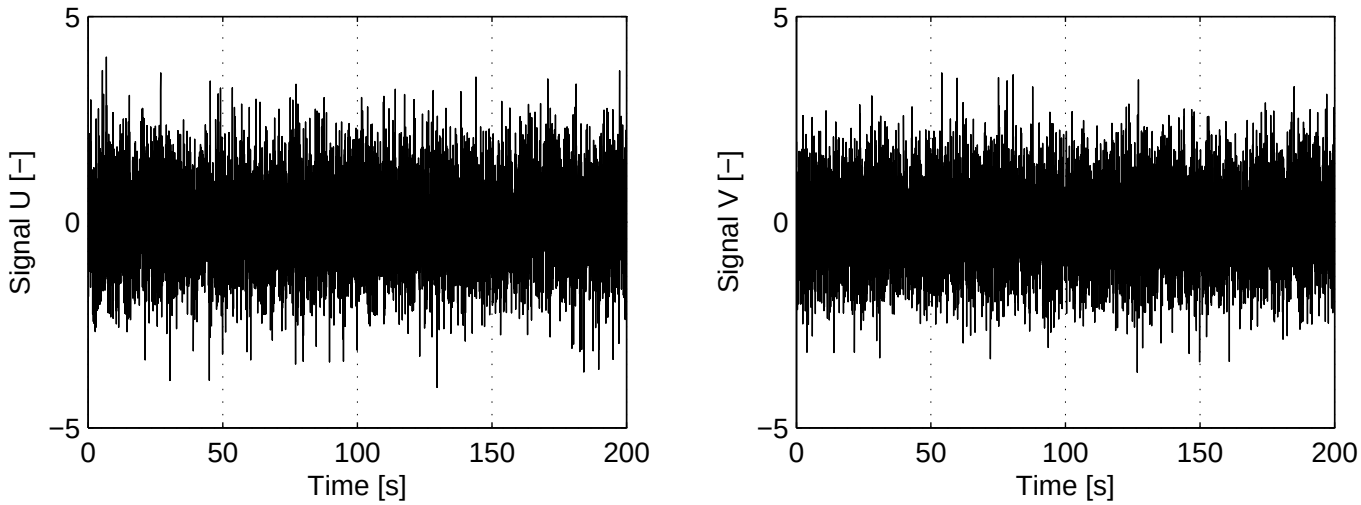
Interpretation:

- For each frequency: how much power of u is present in v?
- Can indicate (cross-)covariance between u and v

Remarks:

- $S_{uv}(f)$ does contain phase information
  - Indicates how much v is shifted w.r.t. u for each frequency f
- $S_{vu}(f) = S^*_{uv}(f)$
  - same magnitude, negative phase
- If $u = v$, then Cross PSD reduces to Auto PSD
- Matlab: `cpsd.m` (again, be aware of scaling!)

Cross power: example



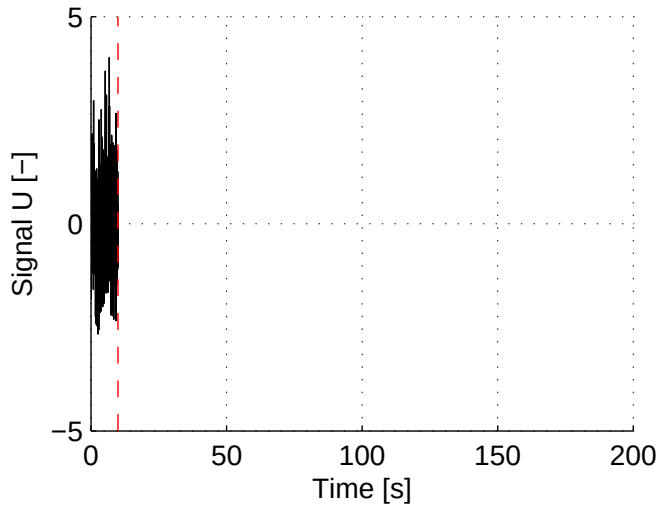
Assume two (apparently) noisy signals $u(t)$ and $v(t)$.

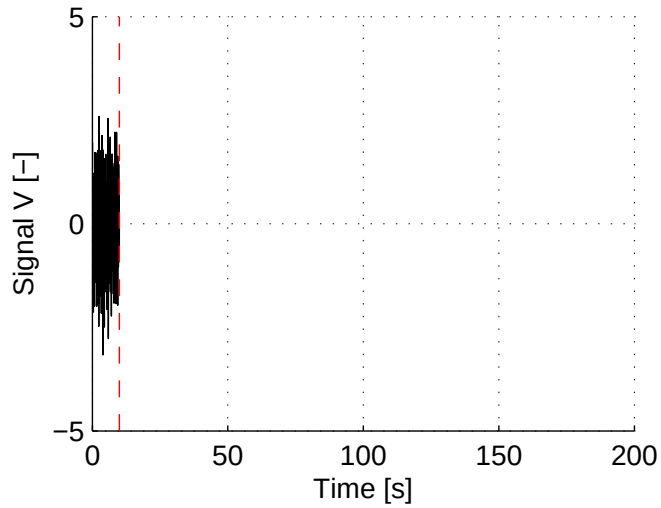
- Do these signals have common ground?
- I.e. is there a repeated common frequency in both?
- Compute CPSD of several pieces (frames) of the signals, and *average* them

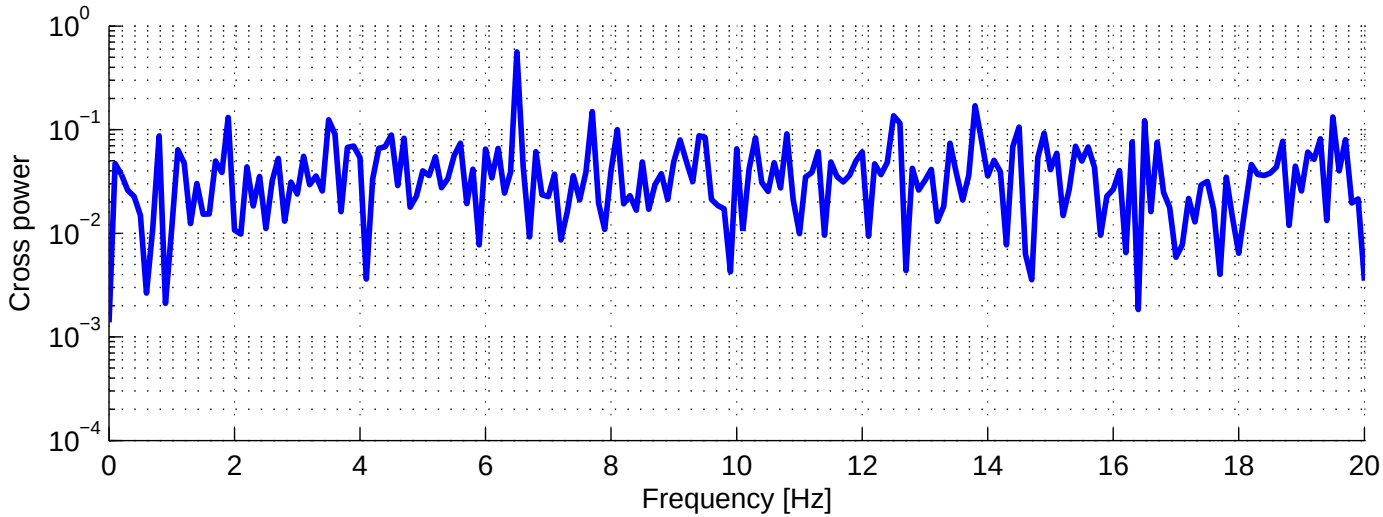
Cross power and importance of averaging

Frames of 10s.

- 1 frame
- 2 frames
- 4 frames
- 10 frames
- 400 frames



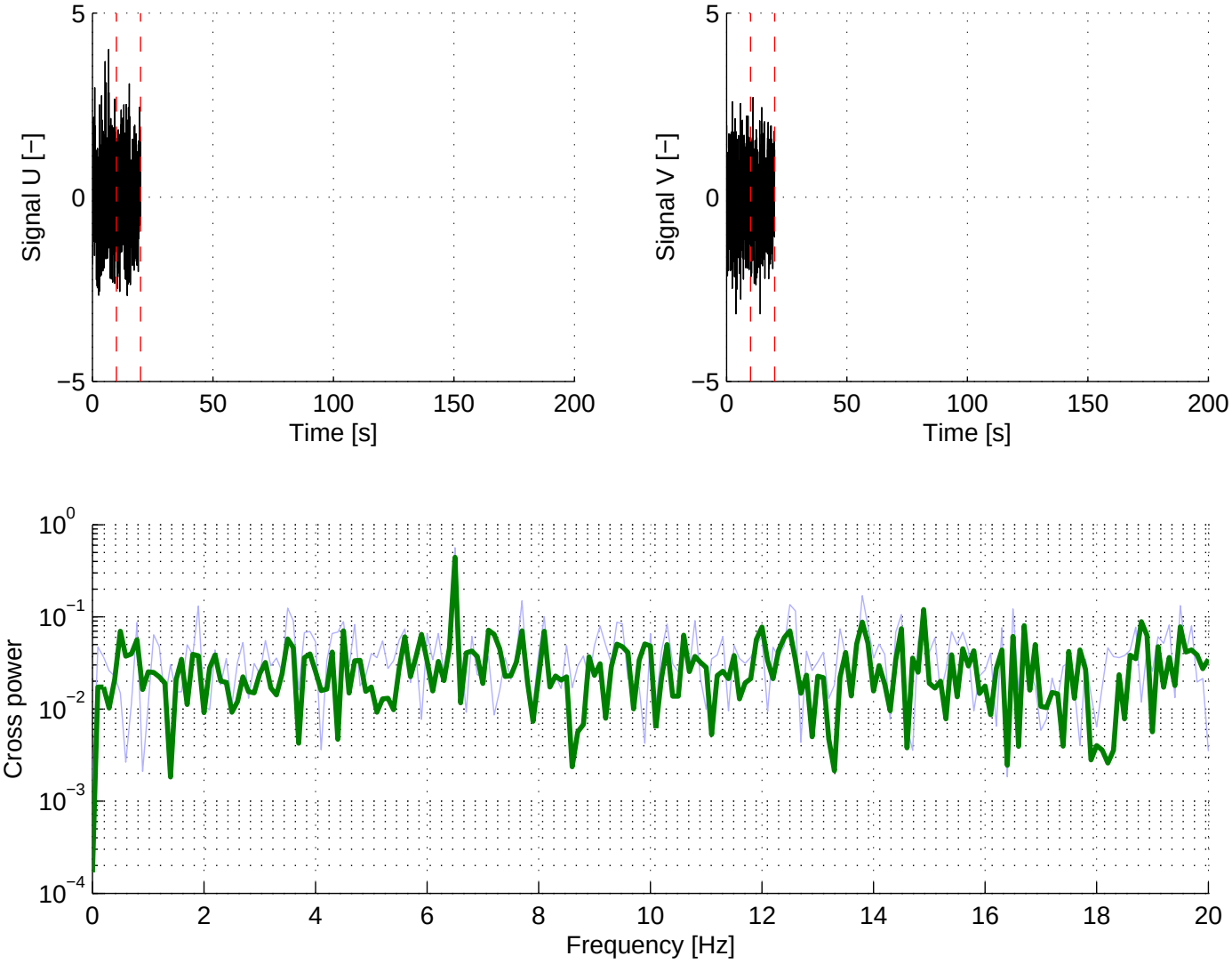




Cross power and importance of averaging

Frames of 10s.

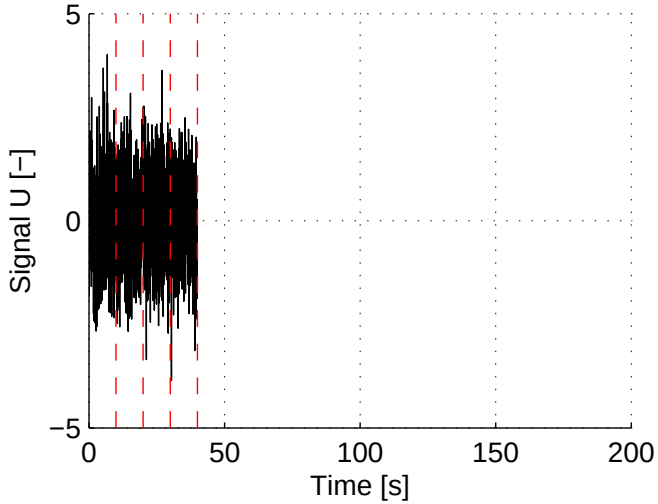
- 1 frame
- 2 frames
- 4 frames
- 10 frames
- 400 frames

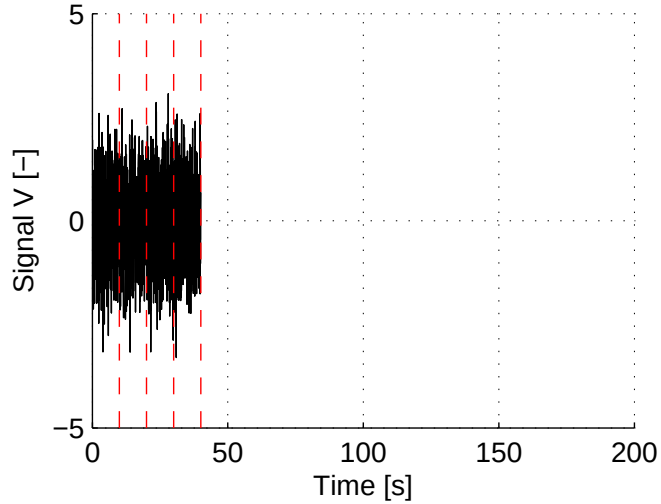


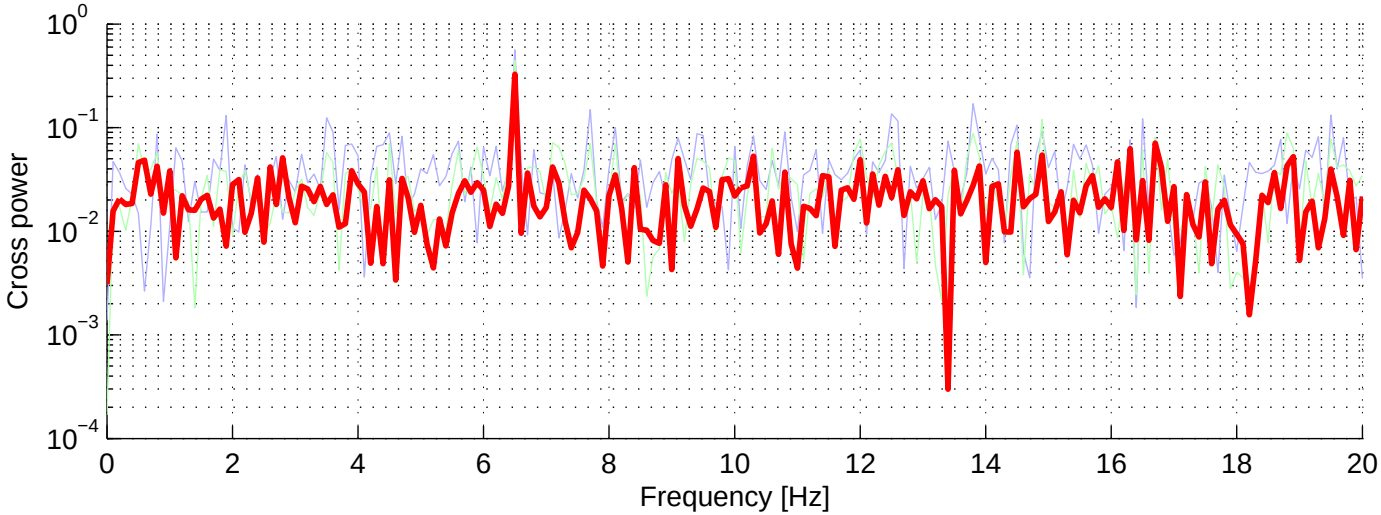
Cross power and importance of averaging

Frames of 10s.

- 1 frame
- 2 frames
- 4 frames
- 10 frames
- 400 frames



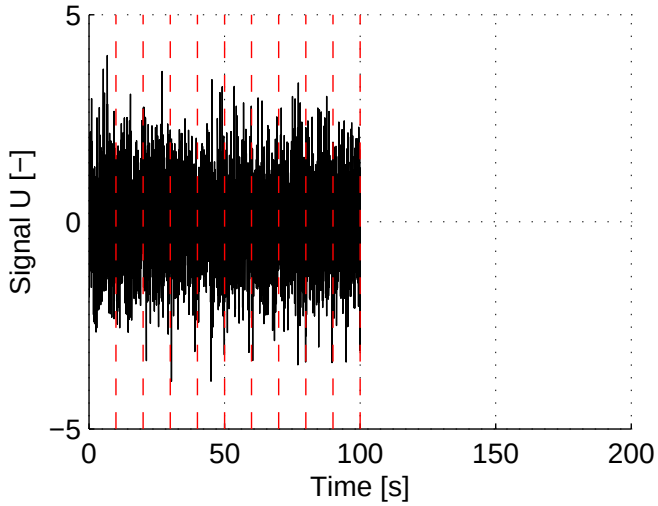


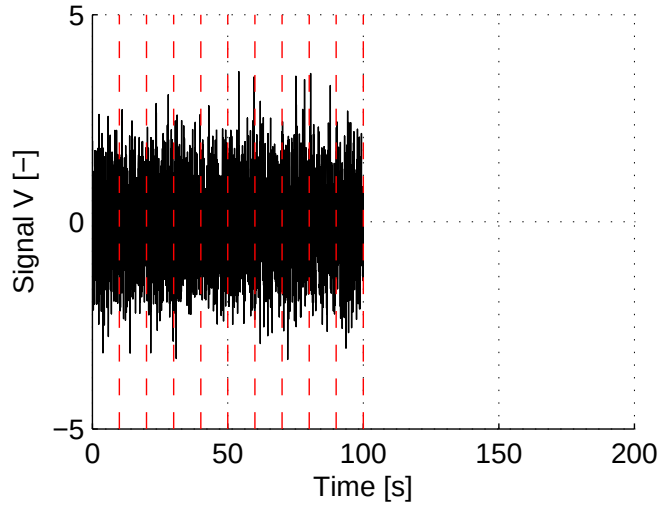


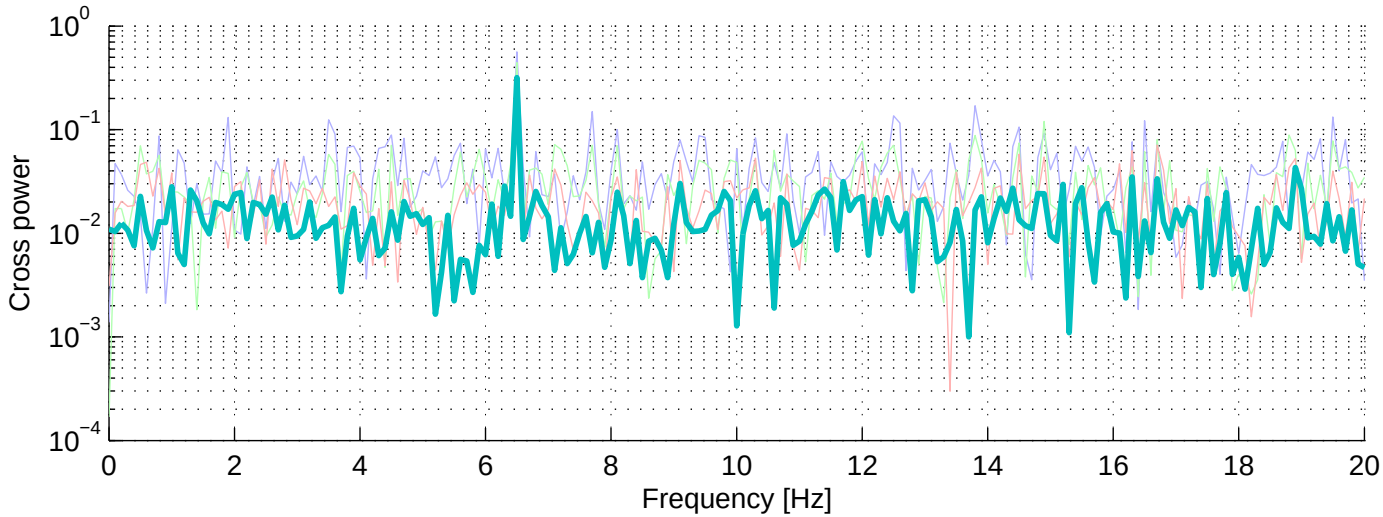
Cross power and importance of averaging

Frames of 10s.

- 1 frame
- 2 frames
- 4 frames
- 10 frames
- 400 frames



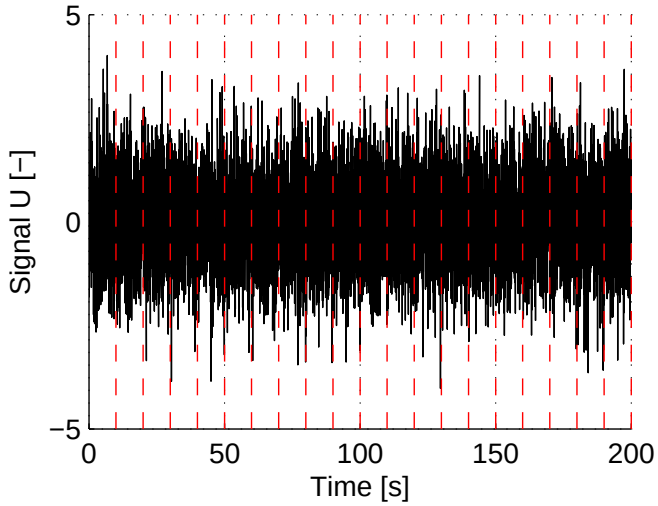


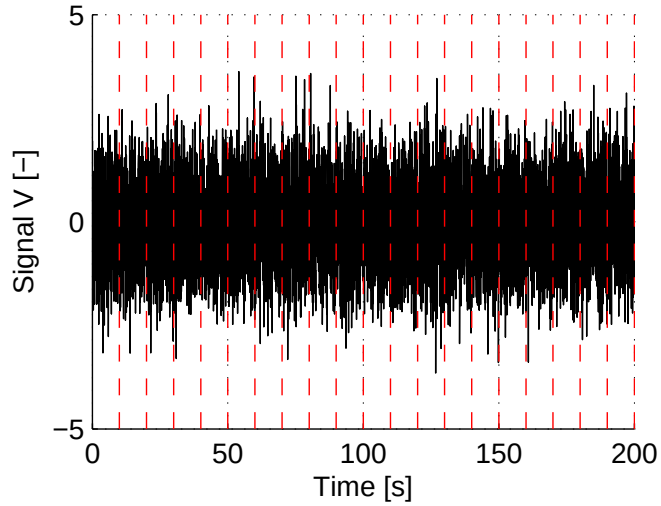


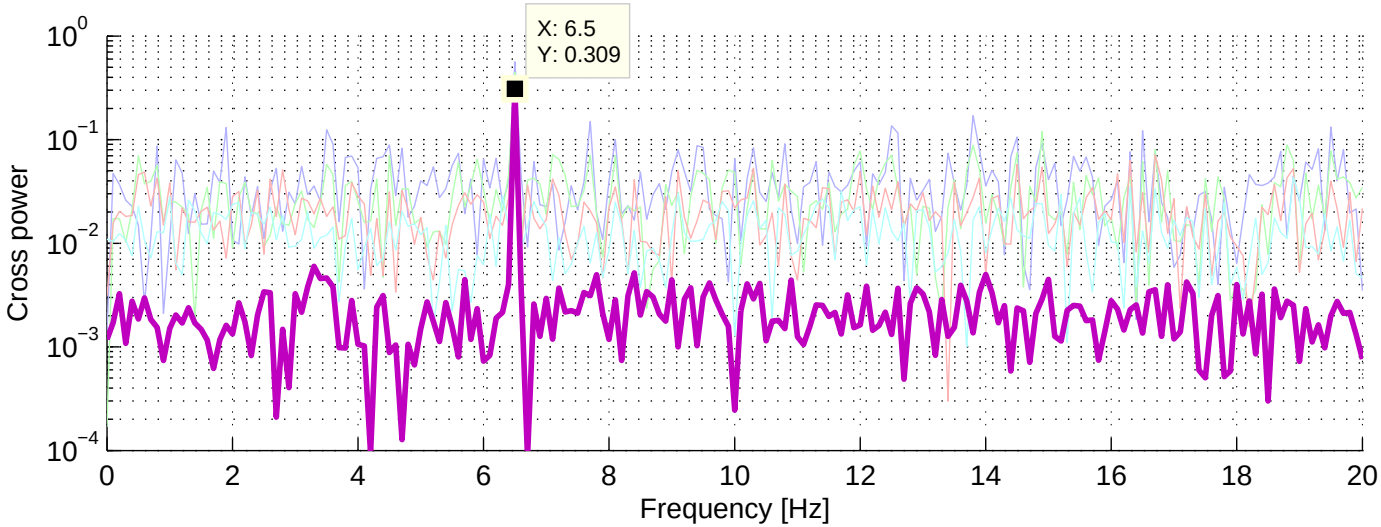
Cross power and importance of averaging

Frames of 10s.

- 1 frame
- 2 frames
- 4 frames
- 10 frames
- 400 frames





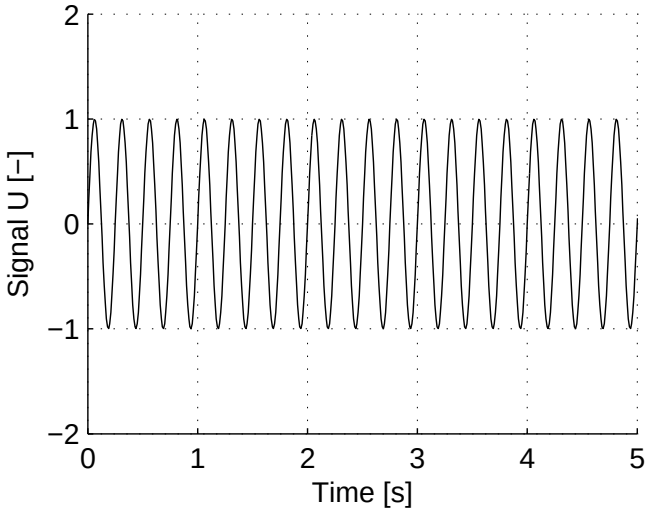


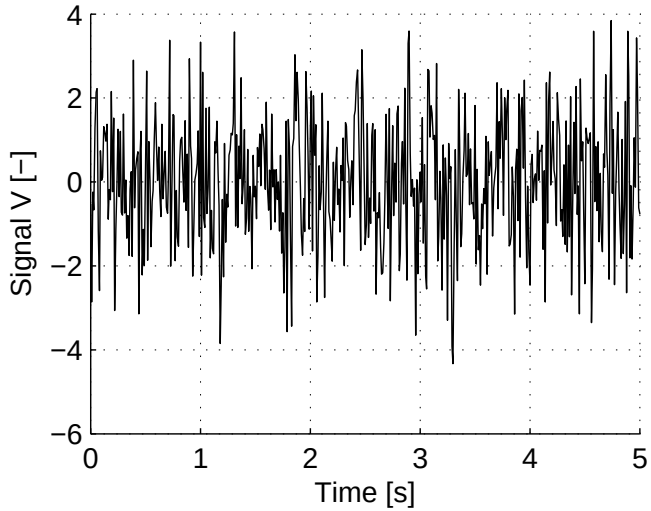
Importance of averaging

Hence, averaging in computing a CPSD is important to

- emphasize repeating commonness between two signals
- but also to minimize the effect of randomness or noise

Example:



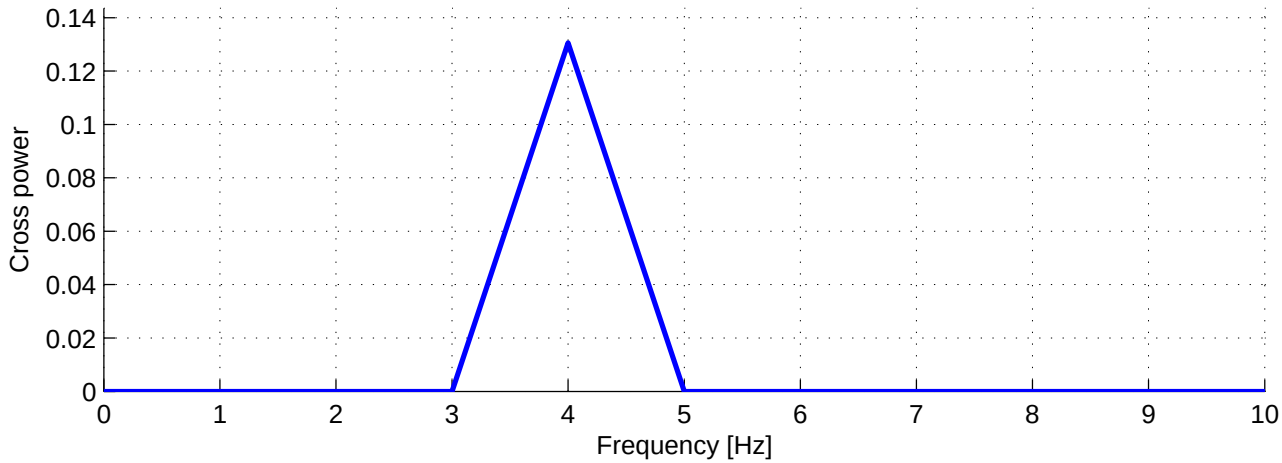


- $u(t)$ is a 4 Hz sine wave
- $v(t)$ is a random number sequence

CPSD with a random number

Frames of 1s.

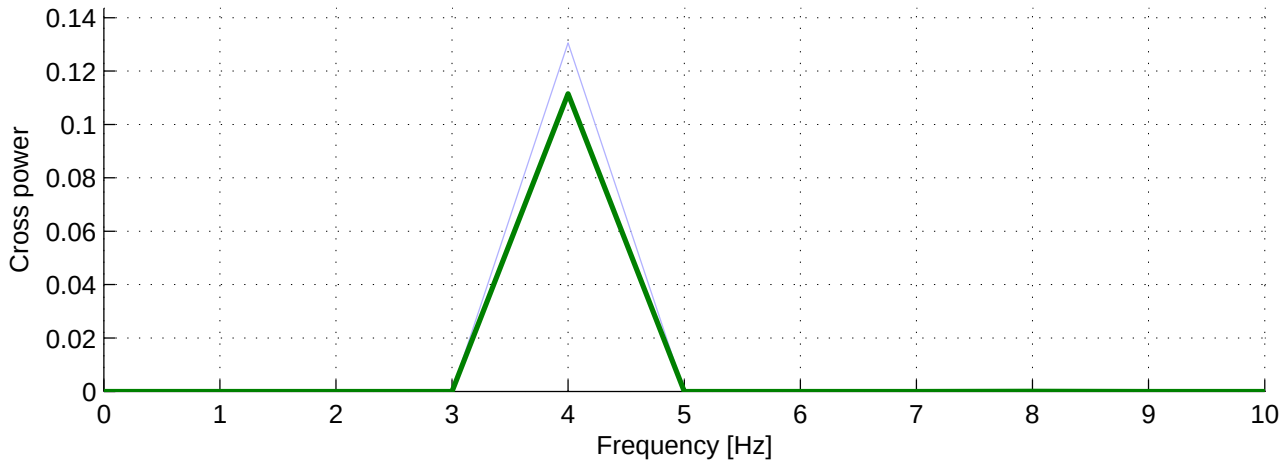
- 1 frame
- 3 frames
- 10 frames
- 100 frames
- 1000 frames



CPSD with a random number

Frames of 1s.

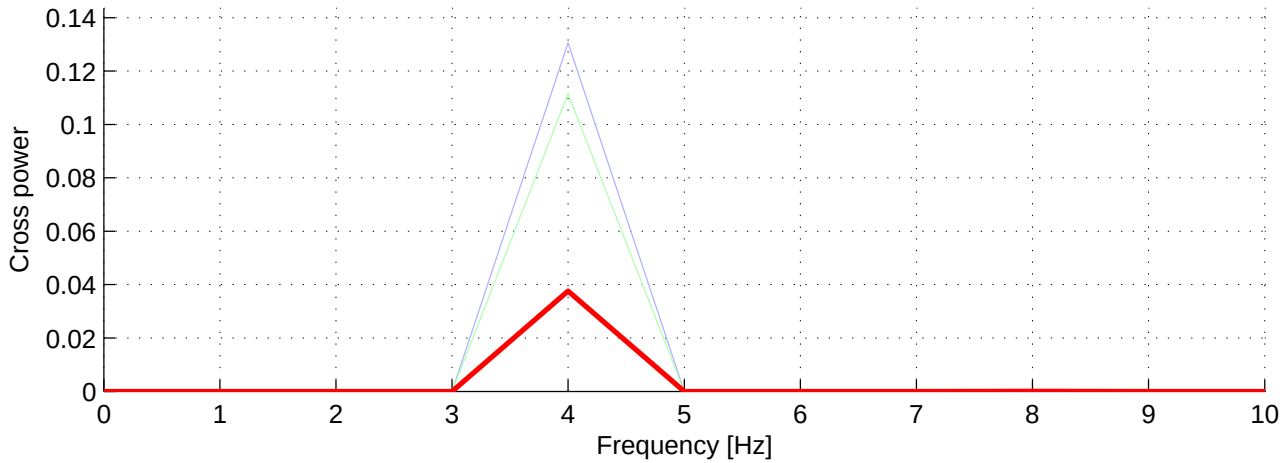
- 1 frame
- 3 frames
- 10 frames
- 100 frames
- 1000 frames



CPSD with a random number

Frames of 1s.

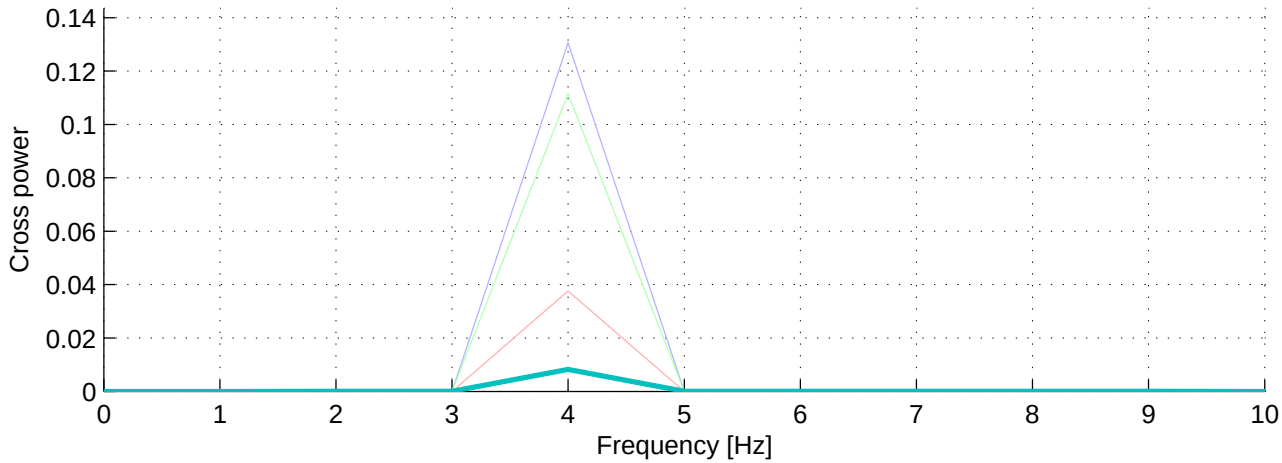
- 1 frame
- 3 frames
- 10 frames
- 100 frames
- 1000 frames



CPSD with a random number

Frames of 1s.

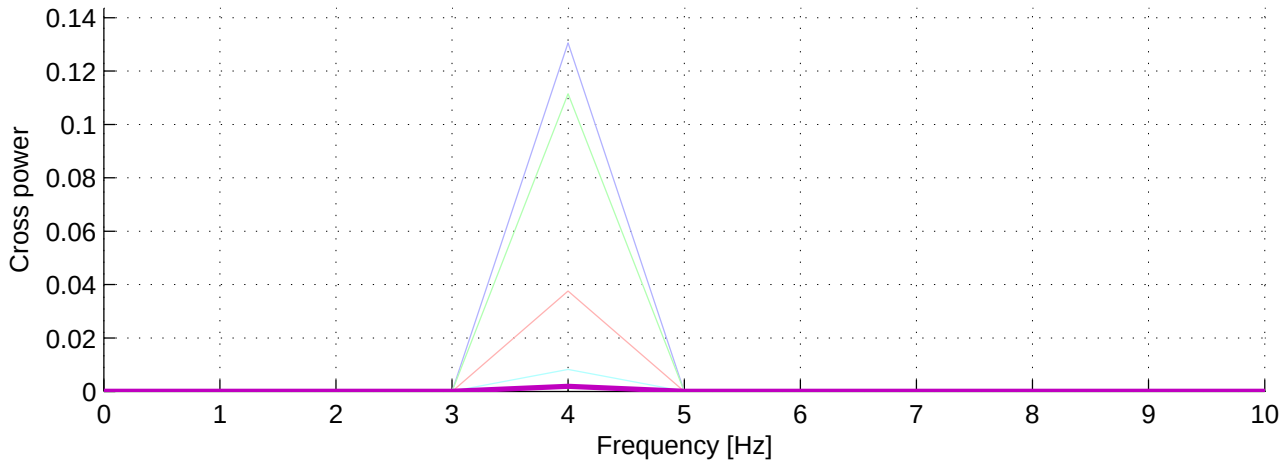
- 1 frame
- 3 frames
- 10 frames
- **100 frames**
- 1000 frames



CPSD with a random number

Frames of 1s.

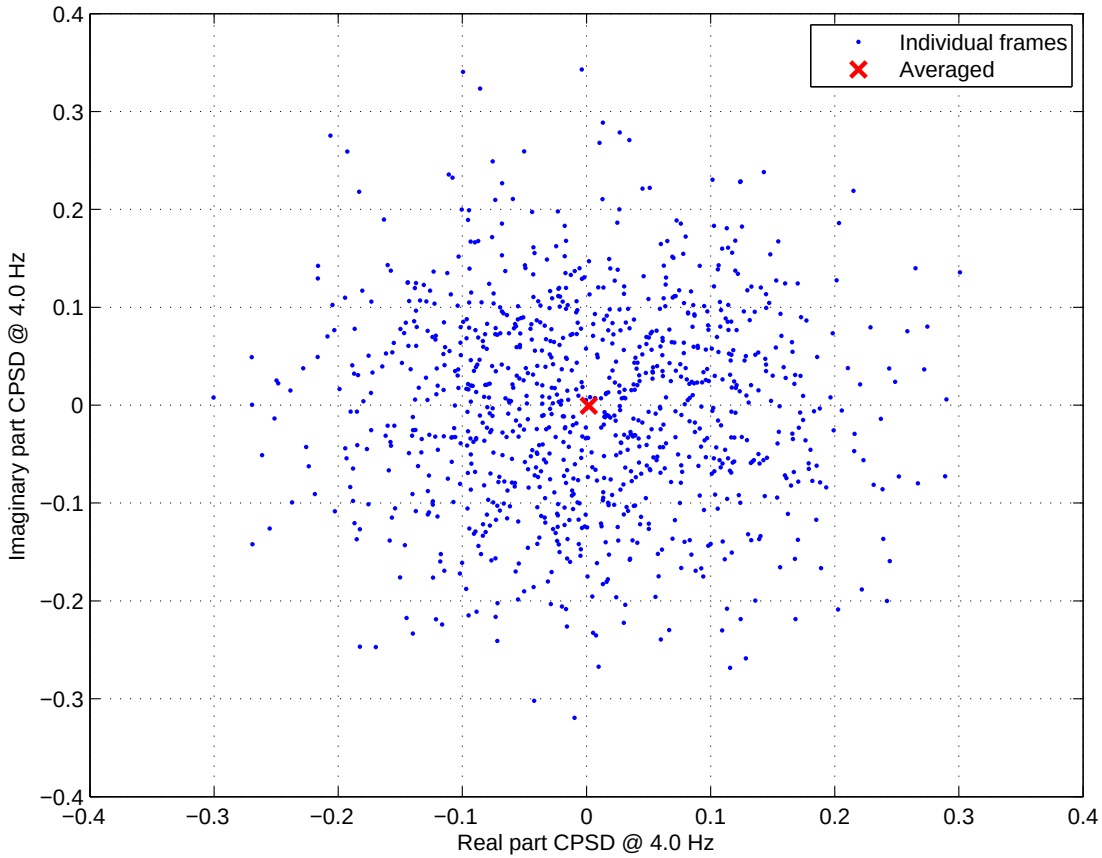
- 1 frame
- 3 frames
- 10 frames
- 100 frames
- 1000 frames



Note that $U(f)V^*(f)$ with $U(f) \neq V(f)$ is a complex number.
Hence, it has phase!

- We're averaging vectors with random size and random angles
- Hence, average CPSD with a random number converges to zero

Averaging a complex number



Conclusion:

- always calculate average CPSD: $\frac{1}{N}\sum_{i=1}^{N} S_{uv}^{i}(f) = \frac{1}{N}\sum_{i=1}^{N} U_i(f)V_i^*(f)$

Averaging the Auto PSD

Note that averaging the *auto* PSD for noisy signals is also a good idea

- APSD is a non-negative real-valued function $S_{uu}(f) \in \mathbb{R}_{\geq 0}$
- Hence, phase is constant at 0°
- Averaged APSD $A_u(f)$ will thus not converge to zero

Due to Parseval we know that

$$\sum_f A_u(f) = \sum_f \left(\frac{1}{N} \sum_{i=1}^{N} S_{uu}^i(f) \right) = P > 0.$$

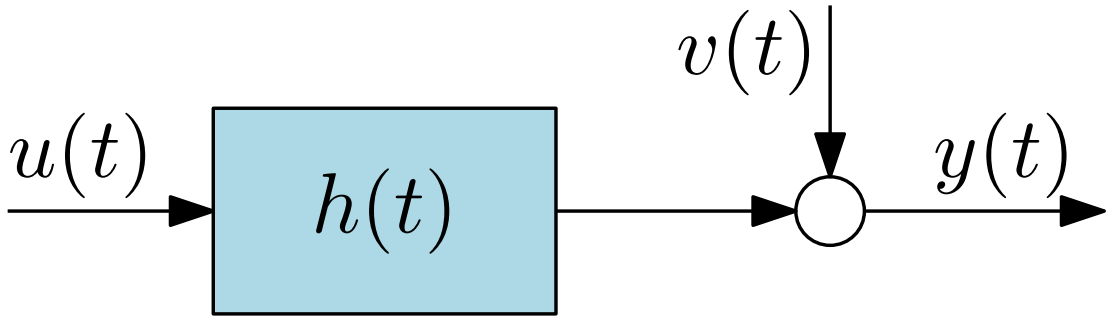
For zero-mean white noise the APSD converges to flat spectrum, hence

$$\lim_{N \to \infty} A_u(f) = A_u = \frac{\sigma^2}{N_f},$$

where σ^2 is the variance of the white noise signal $u(t)$, and N_f is the number of frequency points in the APSD.

How do we use this to ‘measure’ a system?

Direct open loop method

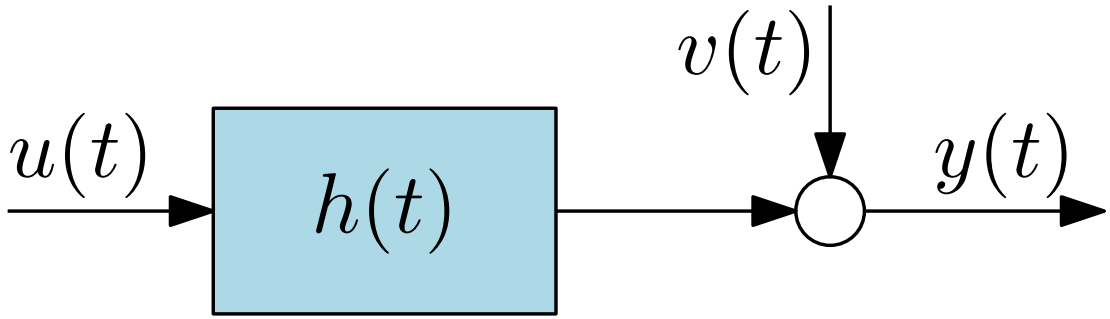


- impulse response $h(t)$ is unknown (but approximately linear)
- $v(t)$ is (unknown) disturbance or noise

Direct method:

- no controller present: open-loop
- directly measuring plant input u and plant output y

Direct open loop method



- impulse response $h(t)$ is unknown (but approximately linear)
- $v(t)$ is (unknown) disturbance or noise

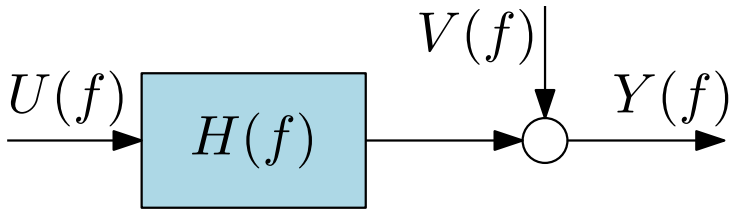
Direct method:

- no controller present: open-loop
- directly measuring plant input u and plant output y

$$y(t) = h(t) \otimes u(t) + v(t)$$

But how to obtain an estimate for $H(s) = \mathcal{L}\{h(t)\}$?

Direct open loop method

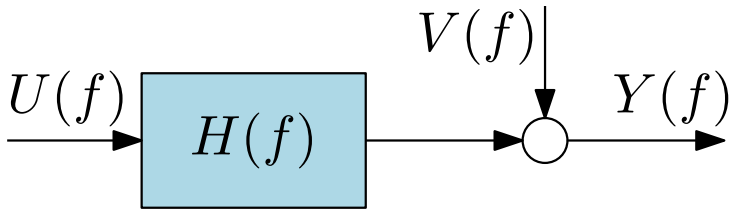


First convert all signals to frequency domain:

$$u(t) \xrightarrow{\texttt{fft}} U(f)$$
$$y(t) \xrightarrow{\texttt{fft}} Y(f)$$
$$v(t) \xrightarrow{\texttt{fft}} V(f)$$

(Note that $v(t)$ and $V(f)$ are actually unknown).

Direct open loop method



First convert all signals to frequency domain:

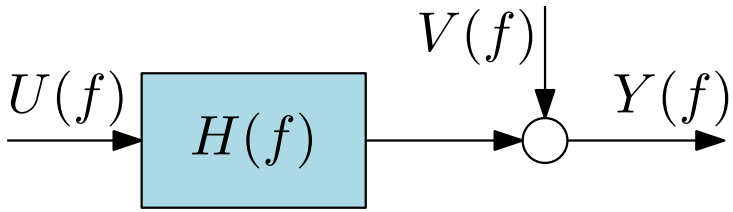
$$u(t) \xrightarrow{\texttt{fft}} U(f)$$
$$y(t) \xrightarrow{\texttt{fft}} Y(f)$$
$$v(t) \xrightarrow{\texttt{fft}} V(f)$$

(Note that $v(t)$ and $V(f)$ are actually unknown).

Then, due to superposition:

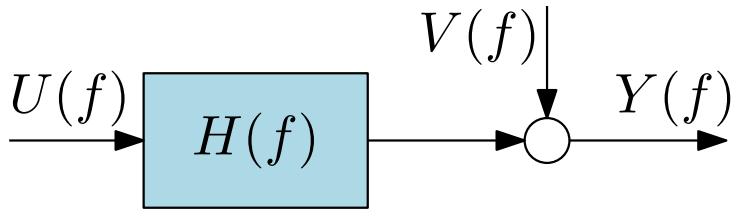
$$Y(f) = H(f)U(f) + V(f)$$

Direct open loop method



$$
\begin{aligned}
Y(f) &= H(f)U(f) + V(f) \\
Y(f)U^*(f) &= H(f)U(f)U^*(f) + V(f)U^*(f) \\
S_{yu}(f) &= H(f)S_{uu}(f) + \underbrace{S_{vu}(f)}_{???}
\end{aligned}
$$

Direct open loop method



$$
\begin{aligned}
Y(f) &= H(f)U(f) + V(f) \\
Y(f)U^*(f) &= H(f)U(f)U^*(f) + V(f)U^*(f) \\
S_{yu}(f) &= H(f)S_{uu}(f) + \underbrace{S_{vu}(f)}_{???}
\end{aligned}
$$

If we can make sure that $S_{vu}(f) \approx 0$, then

$$H(f) \approx \frac{S_{yu}(f)}{S_{uu}(f)}$$

Enhancing the spectral estimation

How to minimize the effect of $v(t)$?

- By making sure $u(t)$ and $v(t)$ are uncorrelated
  - e.g. by choosing $u(t)$ white noise, or other random signal
- By taking multiple data series N and *averaging* them

Enhancing the spectral estimation

How to minimize the effect of $v(t)$?

- By making sure $u(t)$ and $v(t)$ are uncorrelated
  - e.g. by choosing $u(t)$ white noise, or other random signal
- By taking multiple data series N and *averaging* them

Then $\lim_{N\to\infty} \frac{1}{N} \sum_{i=1}^{N} S_{uv}^{i}(f) = 0$, so that

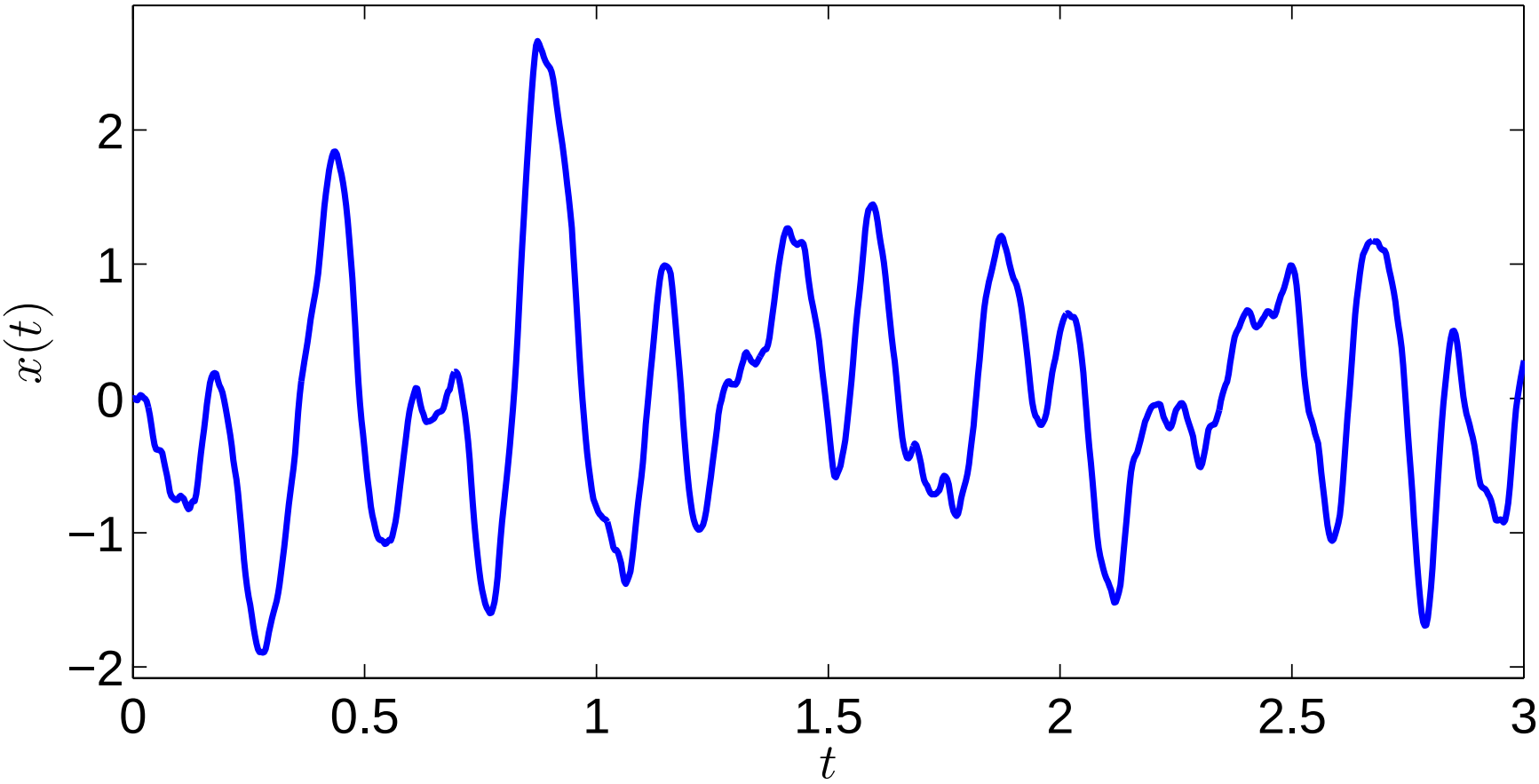
$$H(f) \approx \frac{\sum_{i=1}^{N} S_{yu}^{i}(f)}{\sum_{i=1}^{N} S_{uu}^{i}(f)} = \frac{\sum_{i=1}^{N} Y_i(f) U_i^*(f)}{\sum_{i=1}^{N} U_i(f) U_i^*(f)} \tag{12}$$

Of course, averaging measurements allows you to

- minimize the effect of disturbances, measurement noise, etc.
- reduce the uncertainty on your end result (scales with $\sqrt{N}$)

Transfer function estimates using Welch

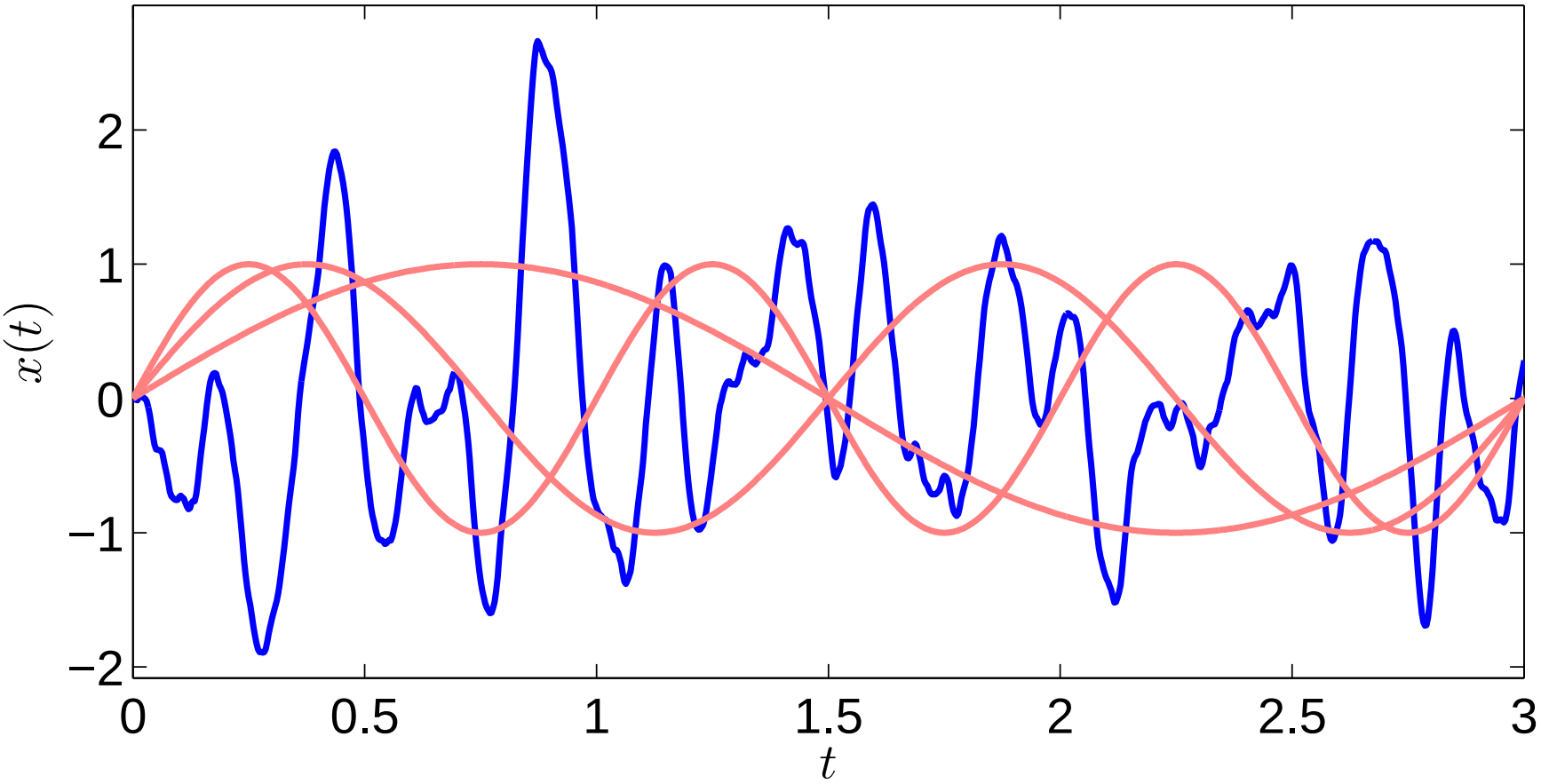
Welch's averaged modified periodogram method



- Example: one data serie of 3 seconds (for both input and output)

Transfer function estimates using Welch

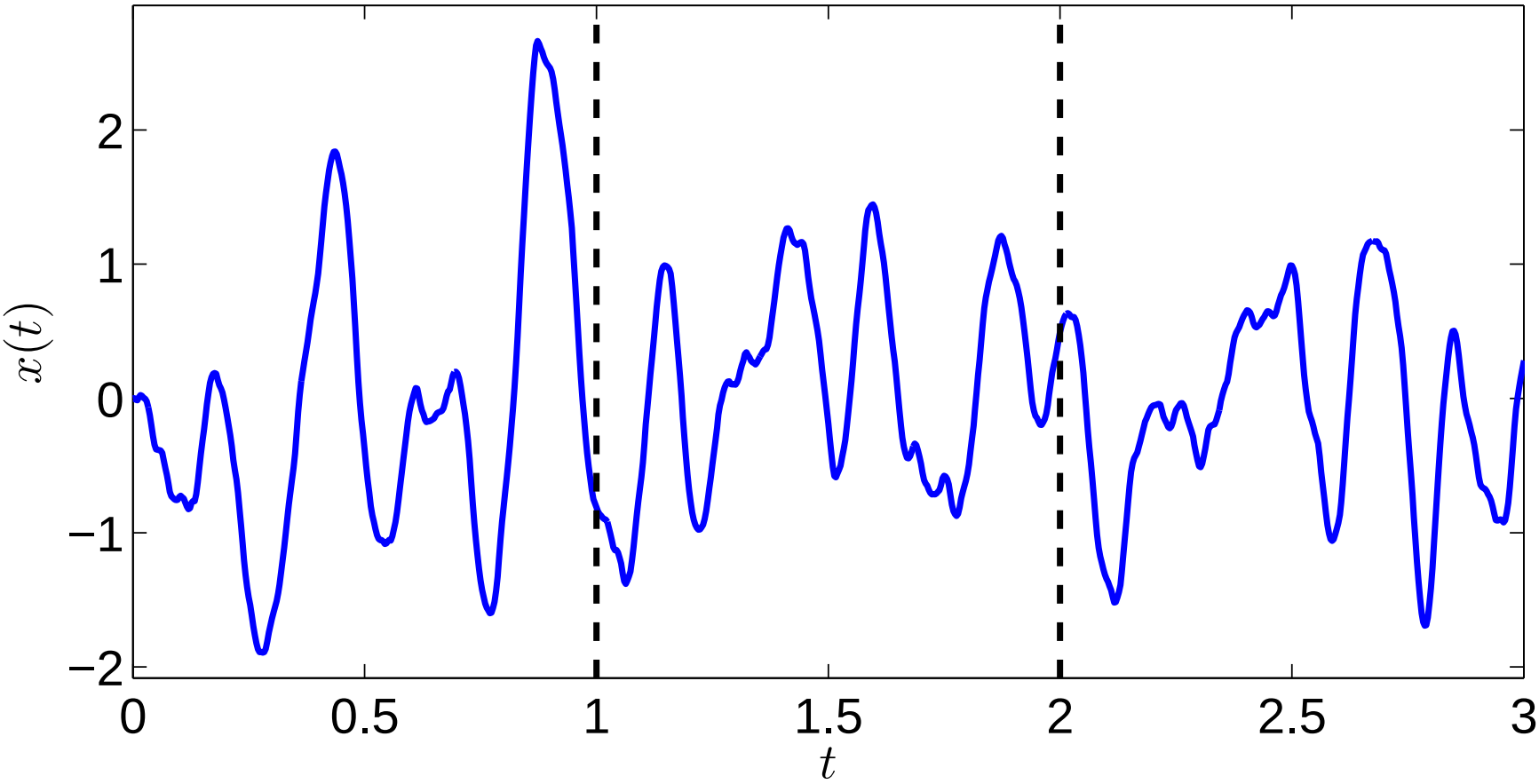
Welch's averaged modified periodogram method



- Calculated spectrum: $\frac{1}{3}$Hz, $\frac{2}{3}$Hz, 1Hz, ...
- Frequency resolution: $\frac{1}{3}$Hz

Transfer function estimates using Welch

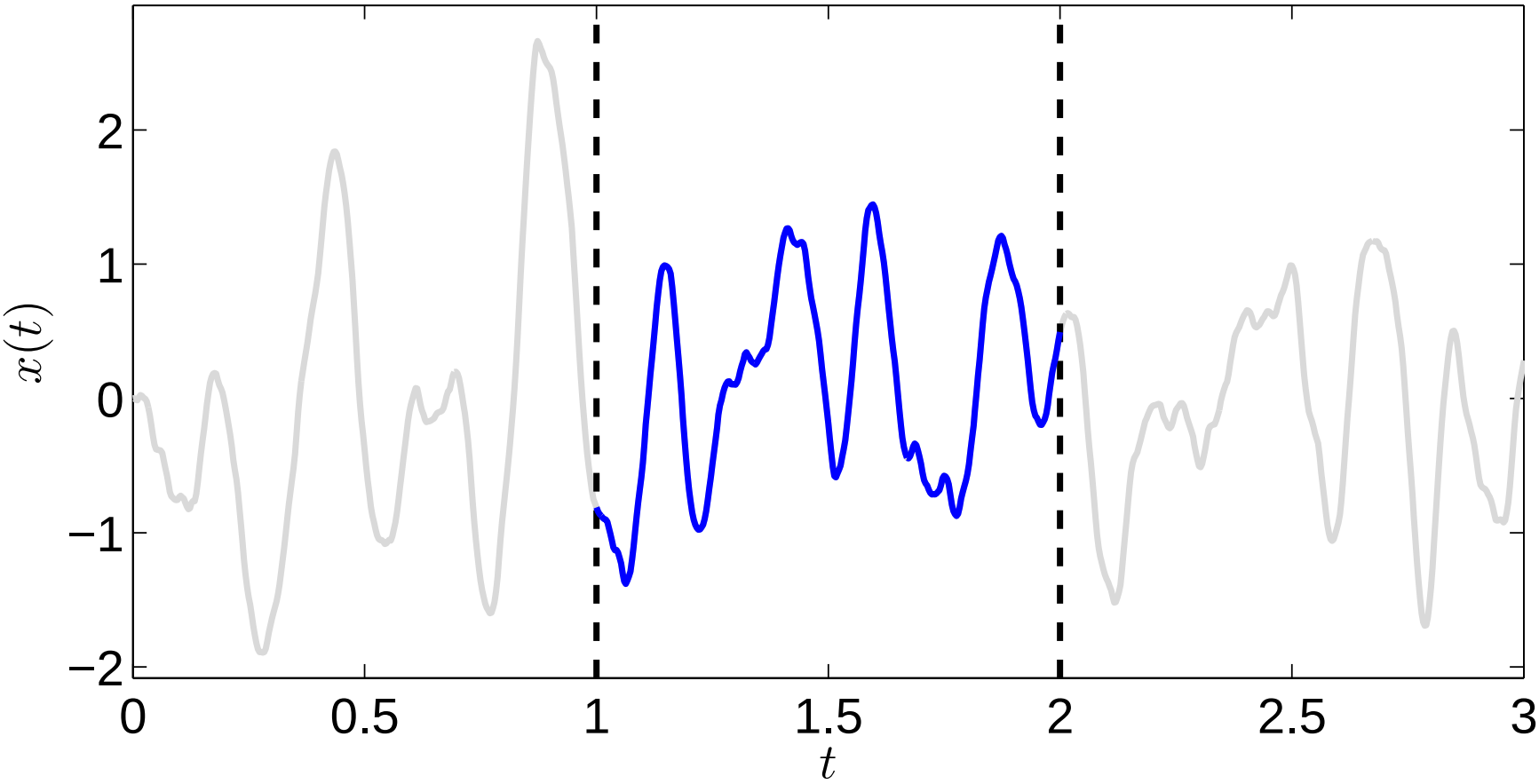
Welch's averaged modified periodogram method



- Create three data *frames* by cutting the original one

Transfer function estimates using Welch

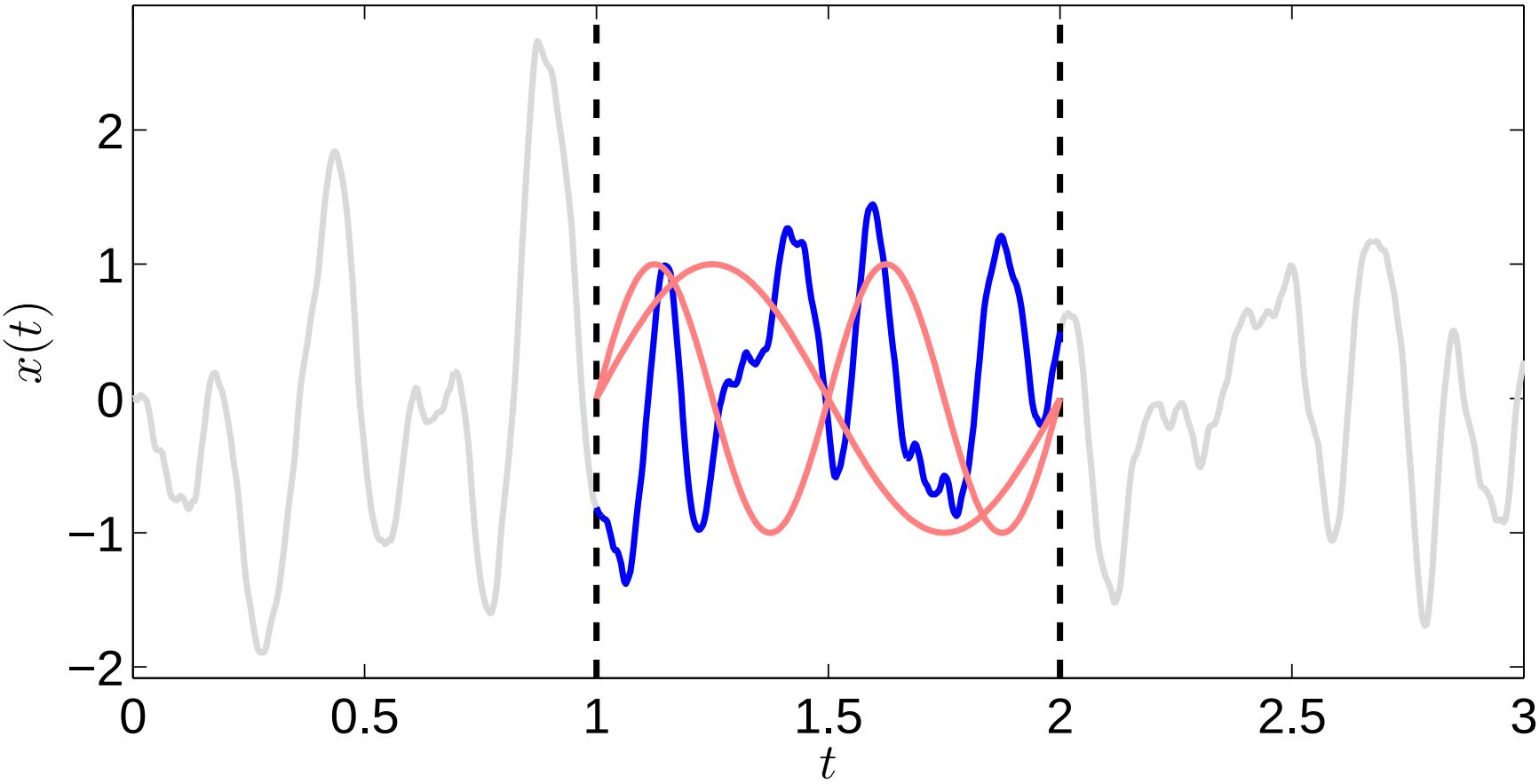
Welch's averaged modified periodogram method



- Calculate Fourier series (and APSD and CPSD) for each block...

Transfer function estimates using Welch

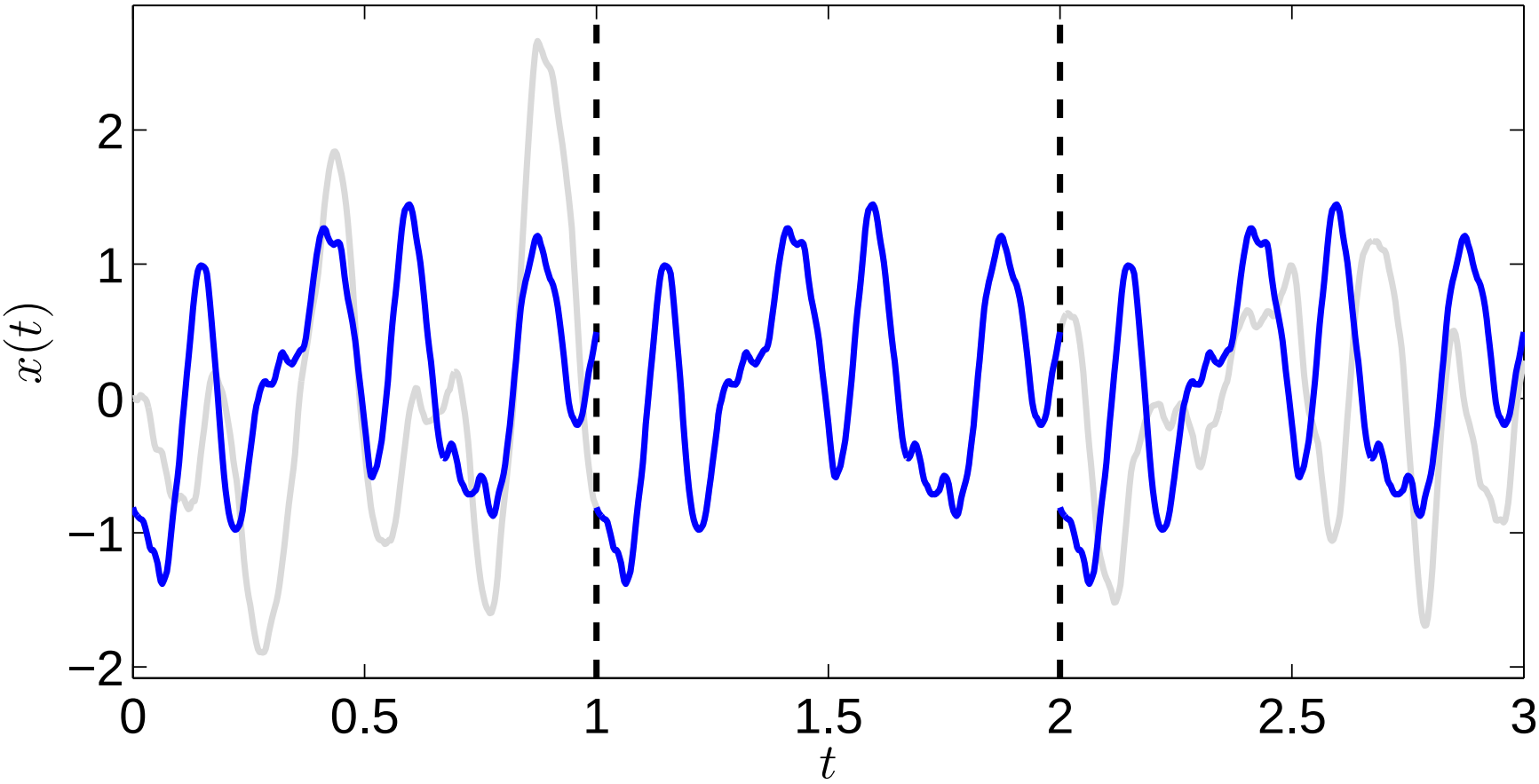
Welch's averaged modified periodogram method



- Calculated spectrum: 1Hz, 2Hz, 3Hz, ...
- Frequency resolution: 1Hz $\Rightarrow$ *loss of resolution*

Transfer function estimates using Welch

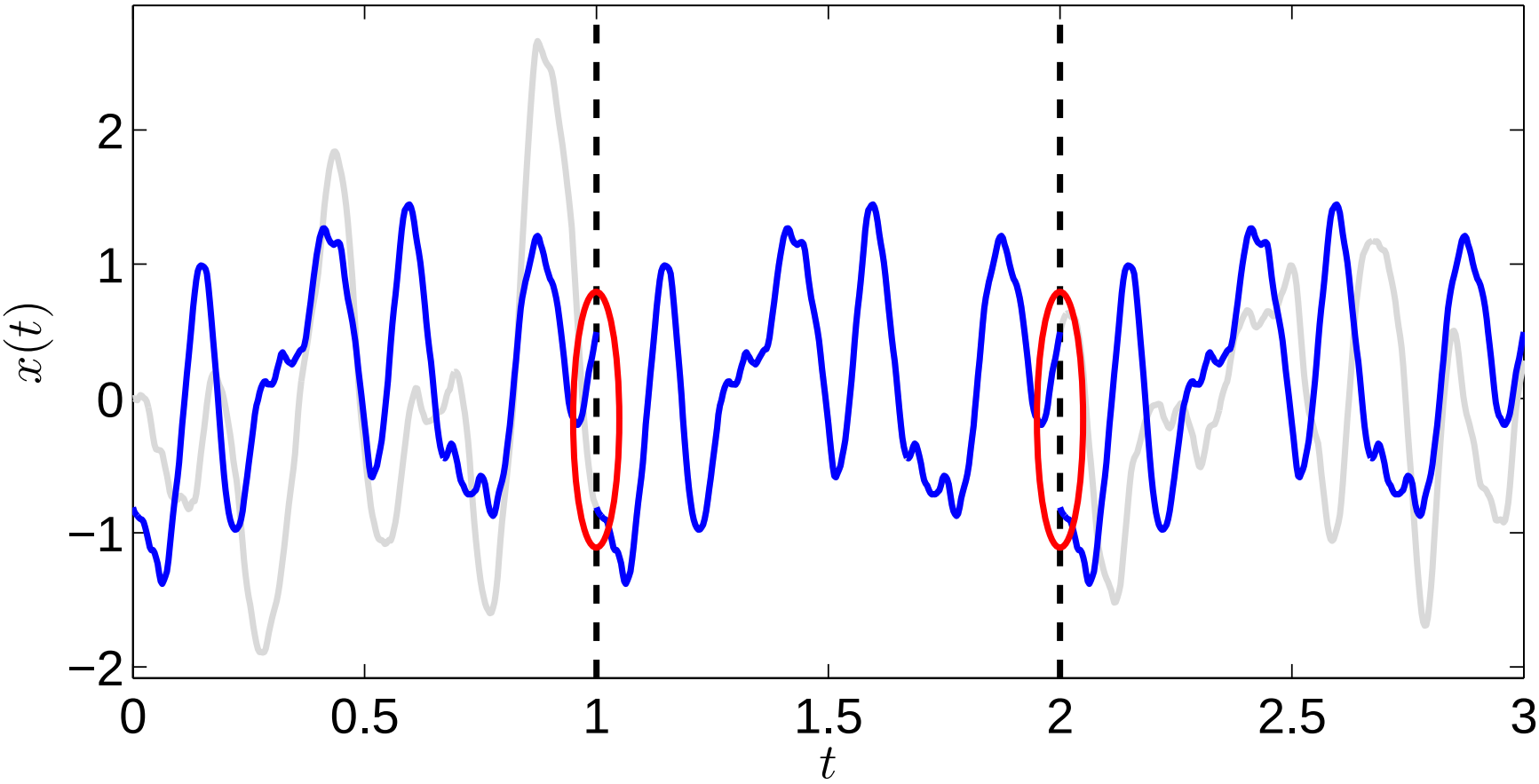
Welch's averaged modified periodogram method



- Fourier assumes the data is periodic...

Transfer function estimates using Welch

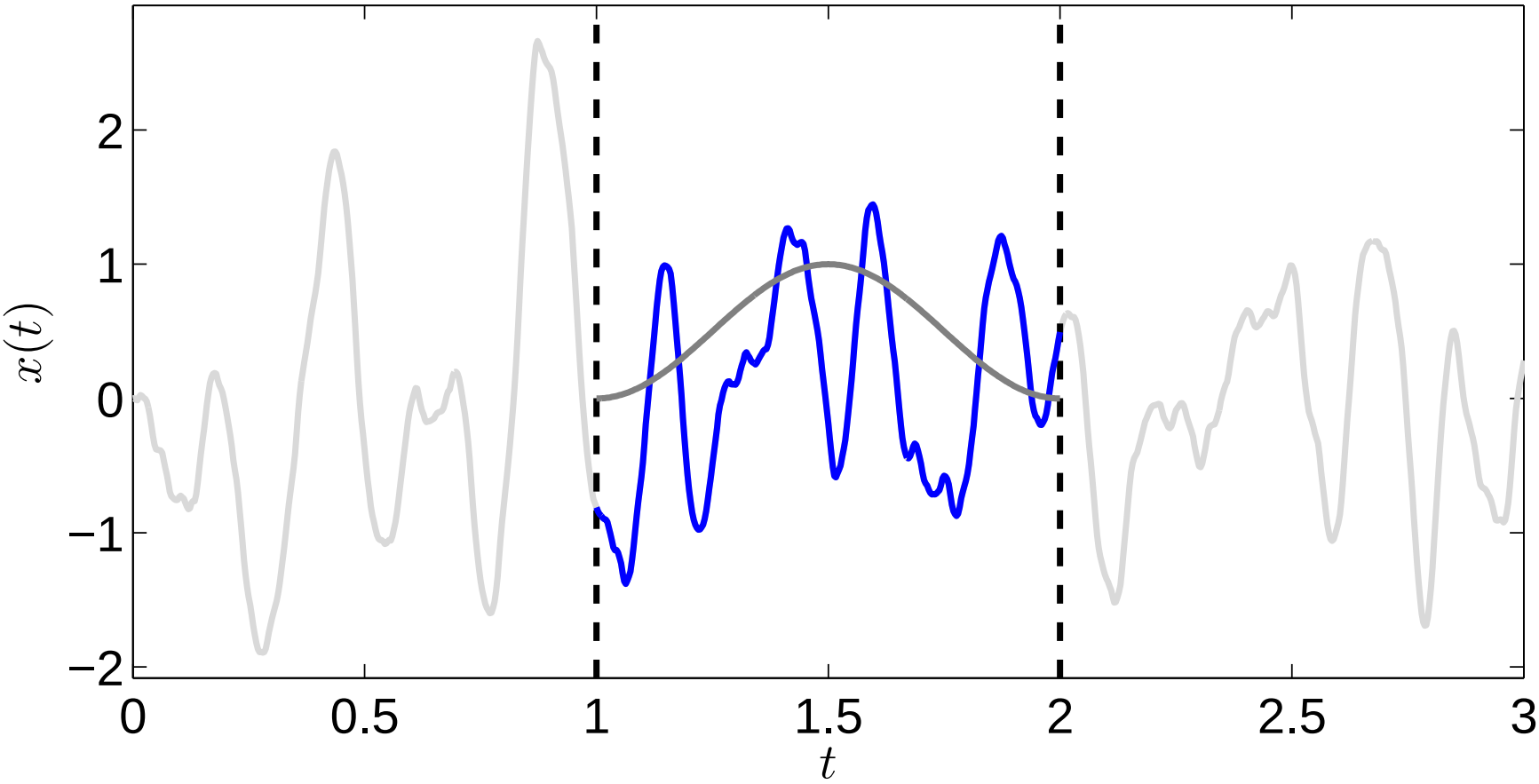
Welch's averaged modified periodogram method



- ...which introduces non-present frequencies

Transfer function estimates using Welch

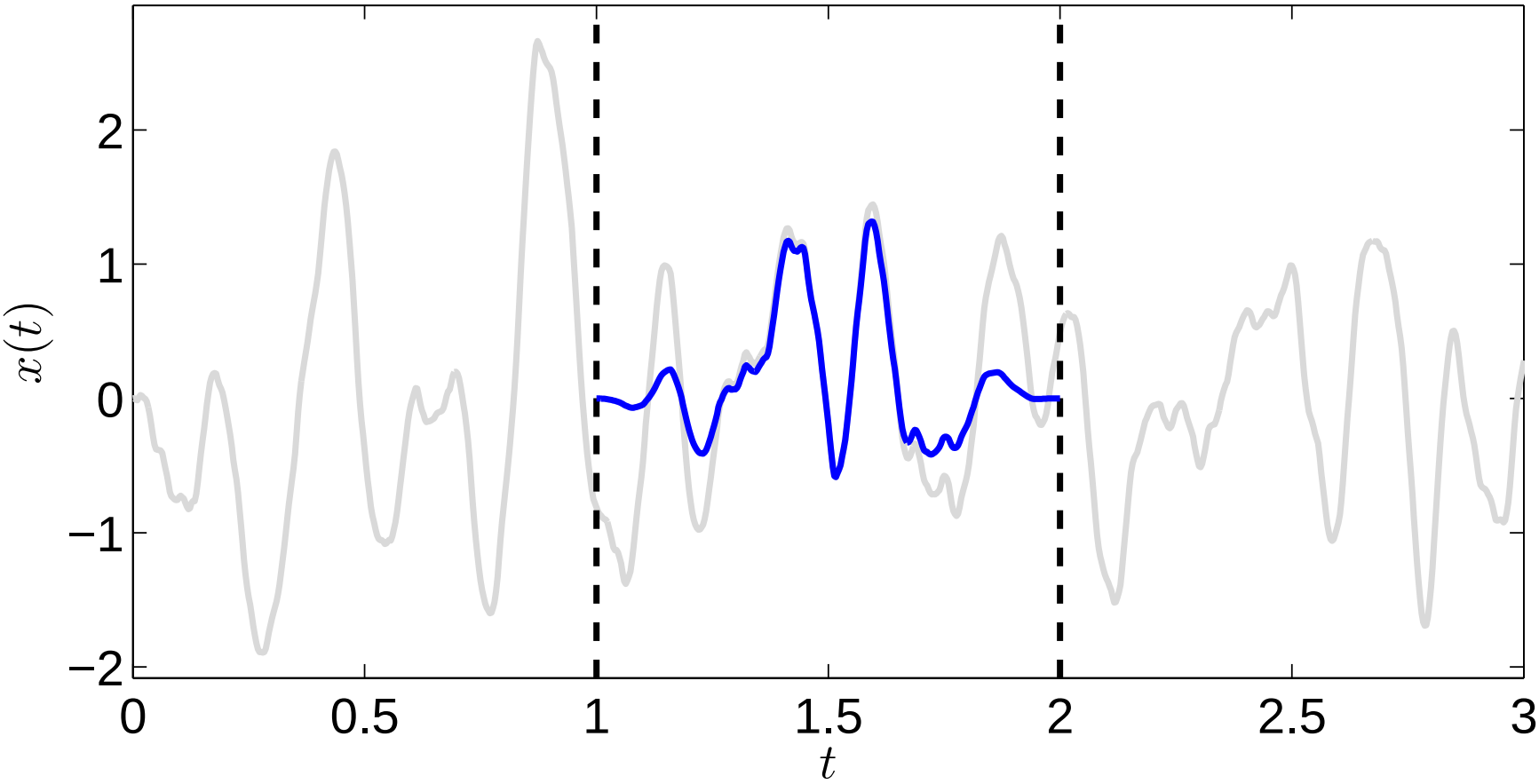
Welch's averaged modified periodogram method



- Solution: multiply signal with a *window*

Transfer function estimates using Welch

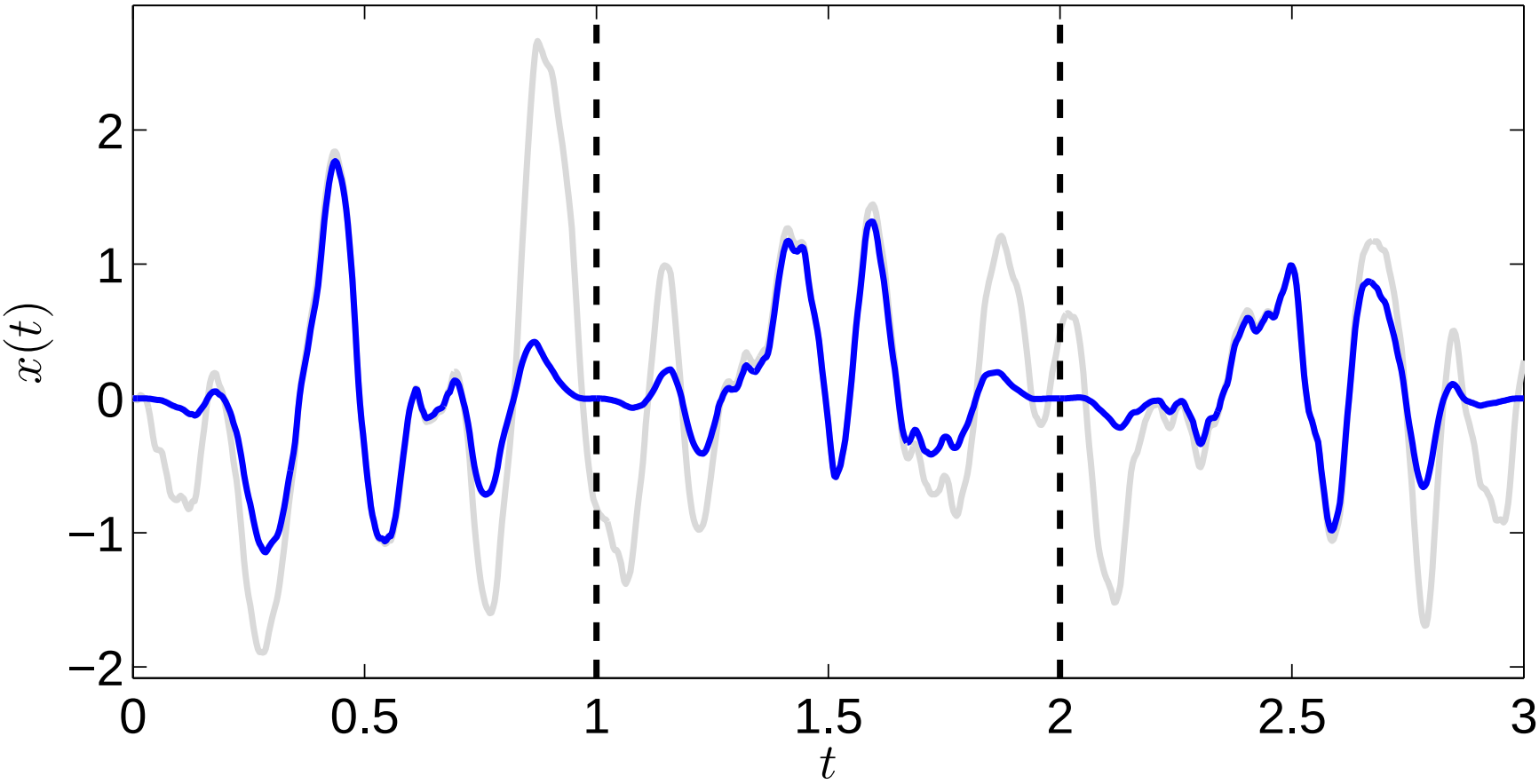
Welch's averaged modified periodogram method



- Solution: multiply signal with a *window*

Transfer function estimates using Welch

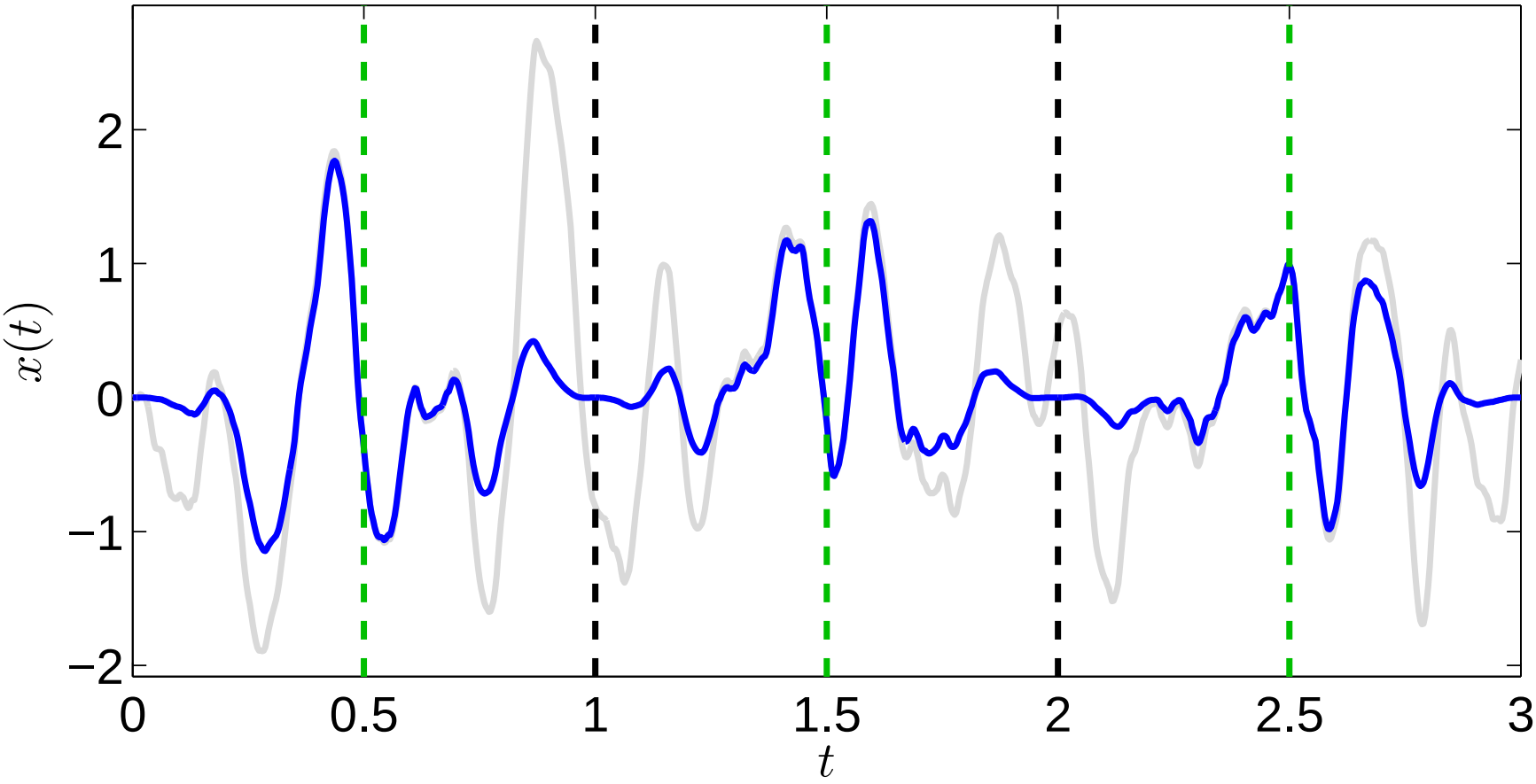
Welch's averaged modified periodogram method



- Disadvantage: loss of data at frame boundaries

Transfer function estimates using Welch

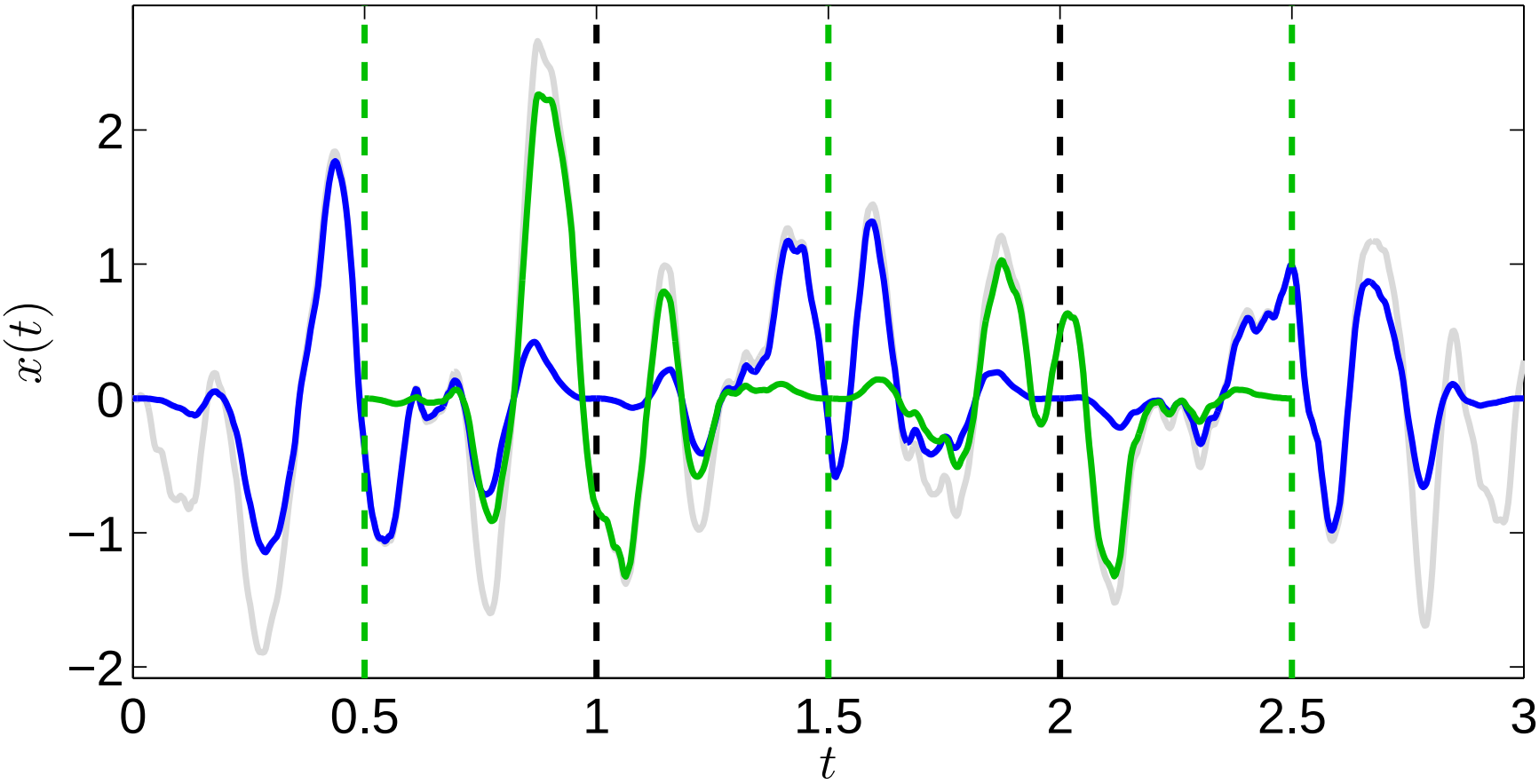
Welch's averaged modified periodogram method



- Solution: create more frames by *overlapping*

Transfer function estimates using Welch

Welch's averaged modified periodogram method



- Now: averaging over 5 frames, resolution of 1Hz

Estimating the transfer function

Transfer function estimation:

$$H(f) \approx \frac{\sum_{i=1}^{N} S_{yu}^{i}(f)}{\sum_{i=1}^{N} S_{uu}^{i}(f)} = \frac{\sum_{i=1}^{N} \bar{Y}_i(f)\bar{U}_i^*(f)}{\sum_{i=1}^{N} \bar{U}_i(f)\bar{U}_i^*(f)}, \quad (13)$$

where $\bar{U}$ and $\bar{Y}$ indicate windowed Fourier coefficients of u and y.

In Matlab:

```
[H,hz] = tfestimate(u,y,window,noverlap,nfft,fs)
```

Estimating the transfer function

Transfer function estimation:

$$H(f) \approx \frac{\sum_{i=1}^{N} S_{yu}^{i}(f)}{\sum_{i=1}^{N} S_{uu}^{i}(f)} = \frac{\sum_{i=1}^{N} \bar{Y}_i(f)\bar{U}_i^*(f)}{\sum_{i=1}^{N} \bar{U}_i(f)\bar{U}_i^*(f)}, \quad (13)$$

where $\bar{U}$ and $\bar{Y}$ indicate windowed Fourier coefficients of u and y.

In Matlab:

```
[H,hz] = tfestimate(u,y,window,noverlap,nfft,fs)
```

- **`window`**: shape of the window, normally **`hann(nfft)`**
- **`noverlap`**: number of overlap between frames (in samples)
- **`nfft`**: frame length in samples
- **`fs`**: sample frequency of **`u`** and **`y`**, to calculate frequency vector **`hz`**

Estimating the transfer function

Transfer function estimation:

$$H(f) \approx \frac{\sum_{i=1}^{N} S_{yu}^{i}(f)}{\sum_{i=1}^{N} S_{uu}^{i}(f)} = \frac{\sum_{i=1}^{N} \bar{Y}_i(f)\bar{U}_i^*(f)}{\sum_{i=1}^{N} \bar{U}_i(f)\bar{U}_i^*(f)}, \quad (13)$$

where $\bar{U}$ and $\bar{Y}$ indicate windowed Fourier coefficients of u and y.

In Matlab:

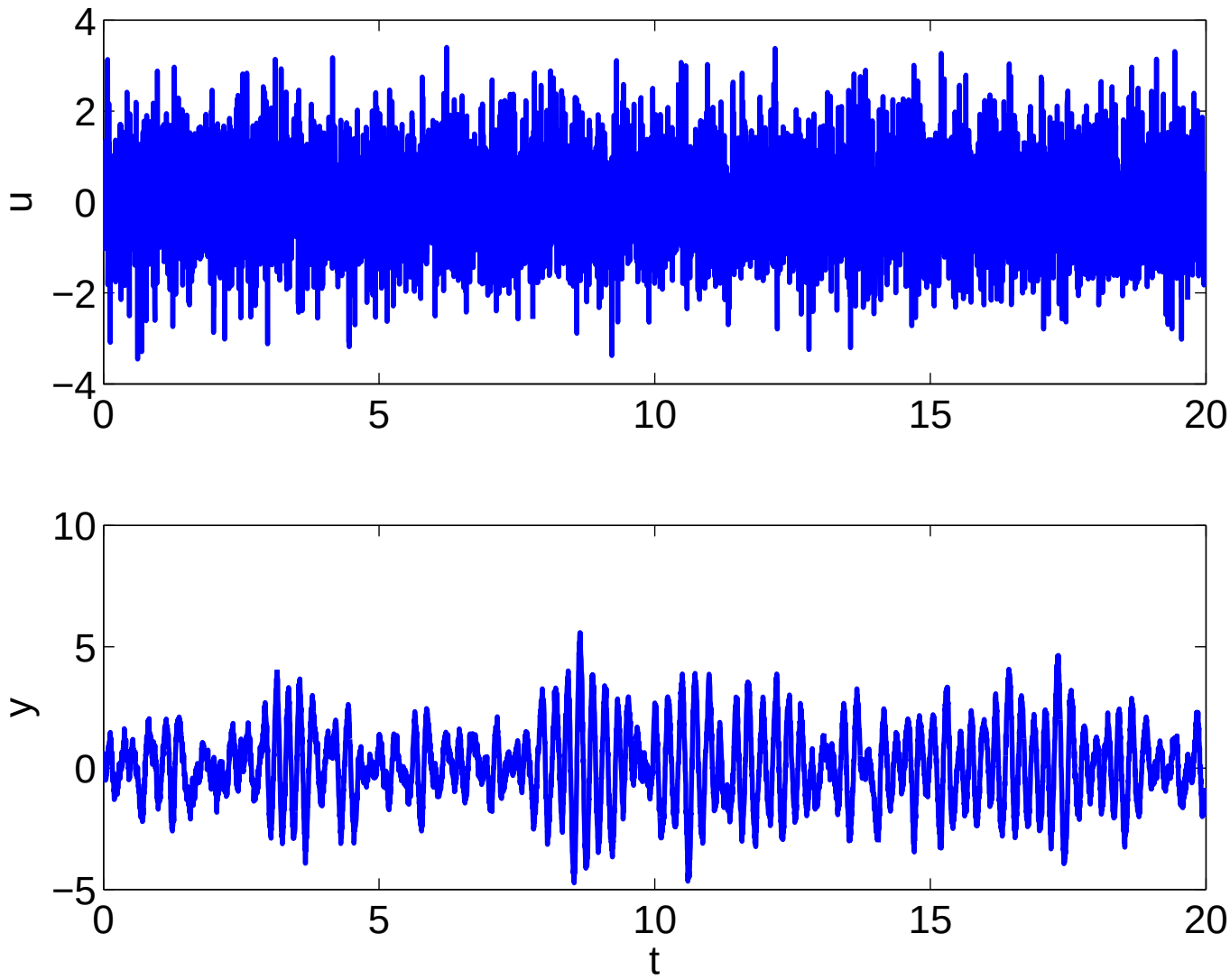
```
[H,hz] = tfestimate(u,y,window,noverlap,nfft,fs)
```

- **`window`**: shape of the window, normally **`hann(nfft)`**
- **`noverlap`**: number of overlap between frames (in samples)
- **`nfft`**: frame length in samples
- **`fs`**: sample frequency of **`u`** and **`y`**, to calculate frequency vector **`hz`**

Note: Frequency resolution = **`fs/nfft`** Hz

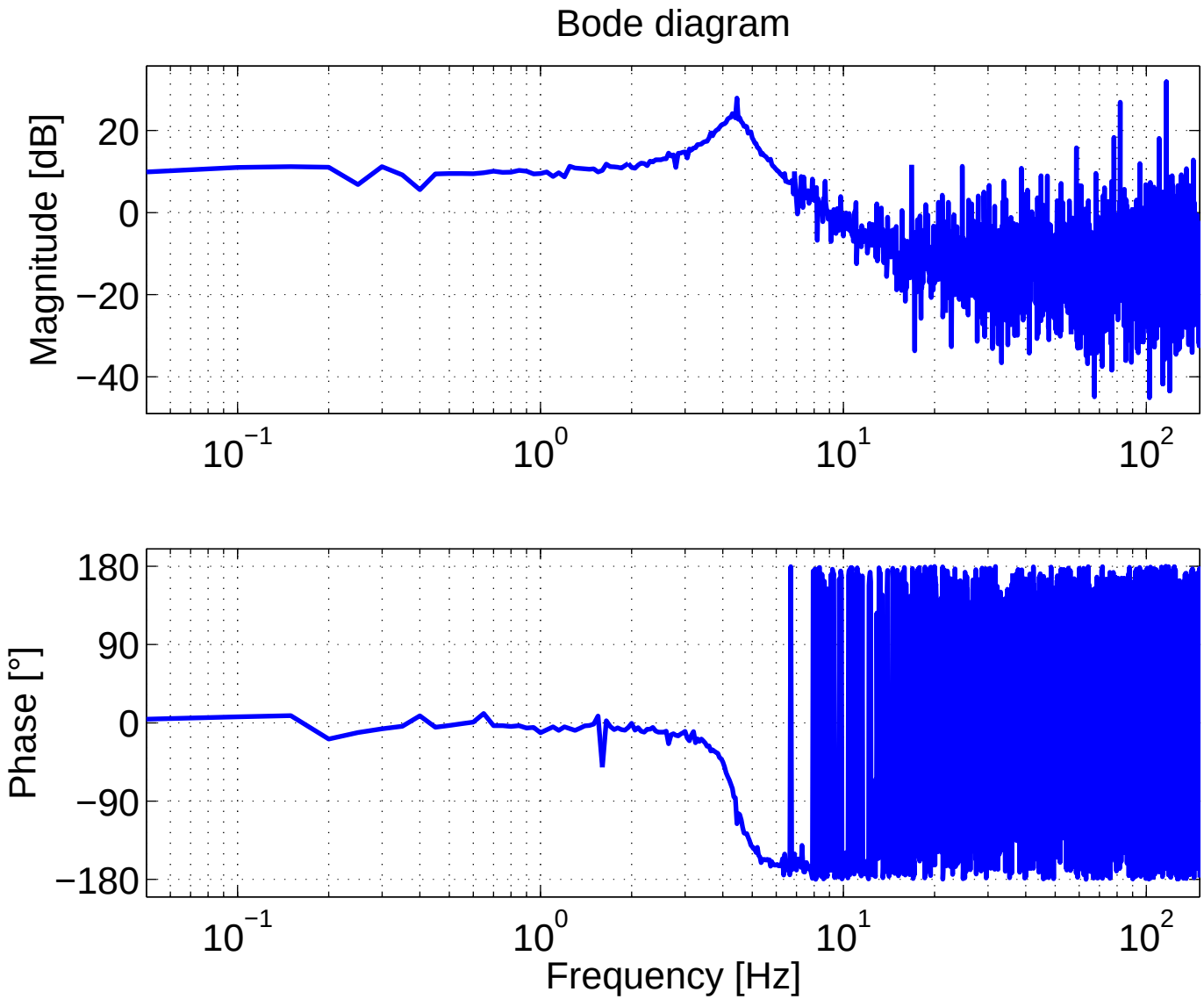
Estimating the transfer function: example

- 20 seconds of input- and output-data
- sampled with 300Hz



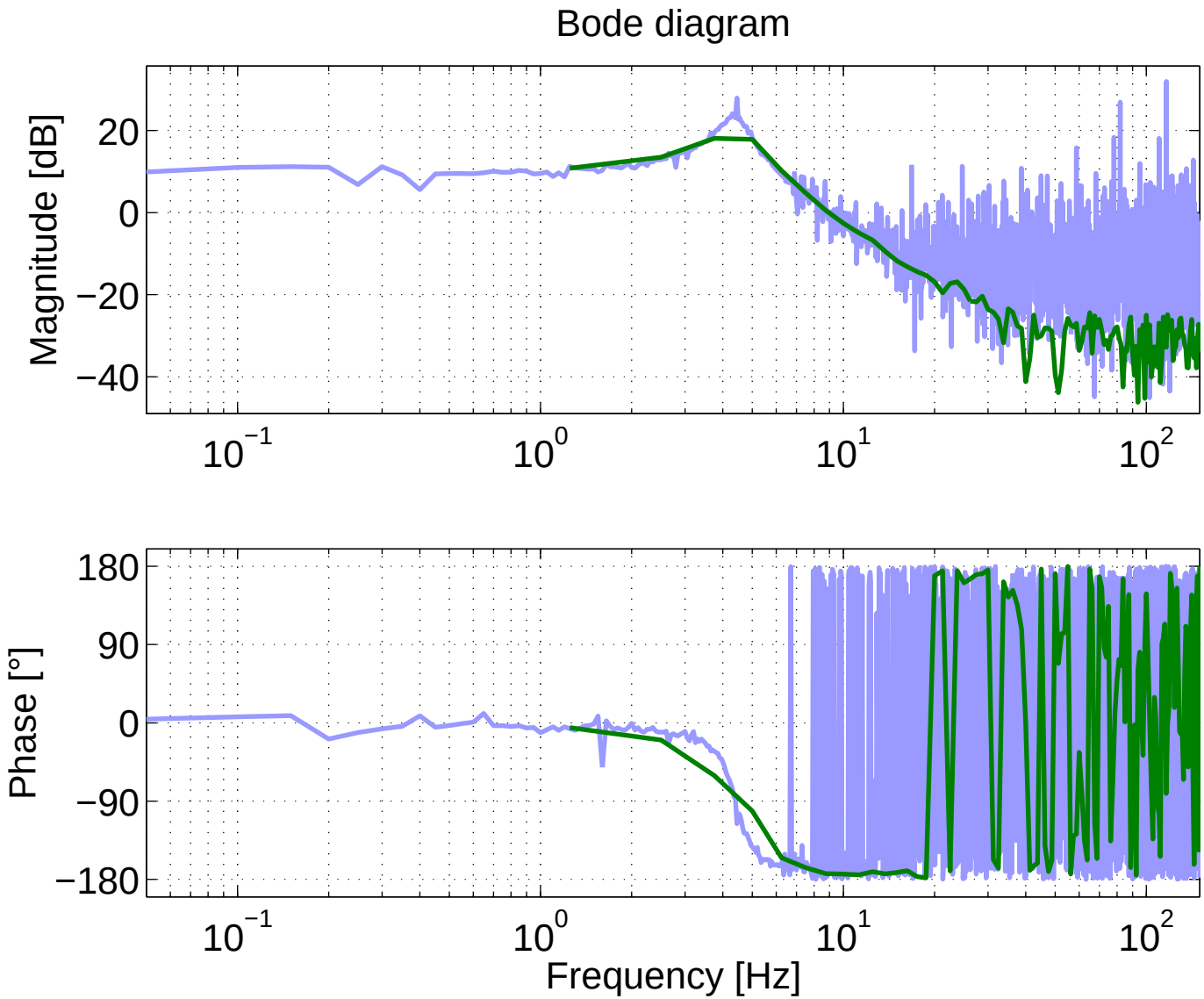
Estimating the transfer function: example

- frame length = 6000 (20s)
- 1 frame
- Resolution: 0.05Hz



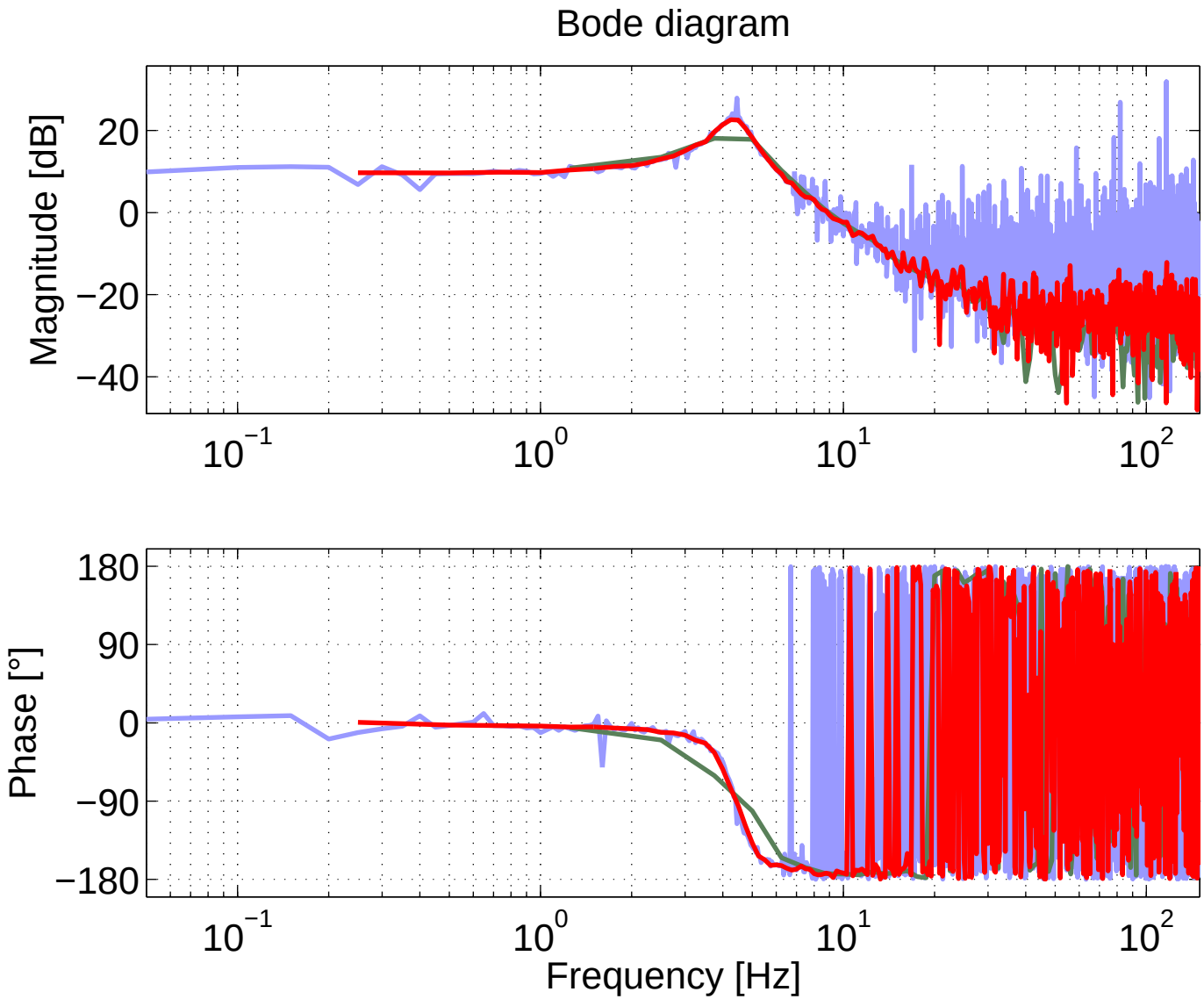
Estimating the transfer function: example

- frame length = 240 (0.8s)
- 49 frames
- Resolution: 1.25Hz



Estimating the transfer function: example

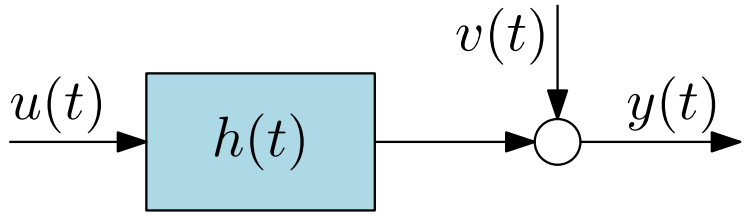
- frame length = 1200 (4s)
- 9 frames
- Resolution: 0.25Hz



What about the quality of the measurement?

Coherence

How reliable is our measurement?



$$
\begin{aligned}
Y(f) &= H(f)U(f) + V(f) \\
Y(f)Y^*(f) &= H(f)U(f)Y^*(f) + V(f)Y^*(f) \\
S_{yy}(f) &= H(f)S_{uy}(f) + \underbrace{S_{vy}(f)}_{???}
\end{aligned}
$$

Quality is determined by $S_{vy}(f)$:

- minimal when $|S_{yy}(f)| = |H(f)S_{uy}(f)|$
- when output power can be explained by plant

Coherence

This motivates the idea behind the *coherence function*:

$$C(f) = \left| \frac{H(f)S_{uy}(f)}{S_{yy}(f)} \right| = \left| \frac{S_{yu}(f)S_{uy}(f)}{S_{uu}(f)S_{yy}(f)} \right| = |H(f)|^2 \frac{S_{uu}(f)}{S_{yy}(f)}$$

(where we assumed that $S_{uv}(f) = 0$).

Coherence

This motivates the idea behind the *coherence function*:

$$C(f) = \left| \frac{H(f)S_{uy}(f)}{S_{yy}(f)} \right| = \left| \frac{S_{yu}(f)S_{uy}(f)}{S_{uu}(f)S_{yy}(f)} \right| = |H(f)|^2 \frac{S_{uu}(f)}{S_{yy}(f)}$$

(where we assumed that $S_{uv}(f) = 0$).

More precisely, when averaging is used:

$$\begin{aligned} C(f) &= \left| \frac{\sum_{i=1}^{N} S_{yu}^{i}(f) \cdot \sum_{i=1}^{N} S_{uy}^{i}(f)}{\sum_{i=1}^{N} S_{uu}^{i}(f) \cdot \sum_{i=1}^{N} S_{yy}^{i}(f)} \right| \\ &= \left| \frac{\left(\sum_{i=1}^{N} \bar{Y}_i(f) \bar{U}_i^*(f) \right)^2}{\sum_{i=1}^{N} \bar{U}_i(f) \bar{U}_i^*(f) \cdot \sum_{i=1}^{N} \bar{Y}_i(f) \bar{Y}_i^*(f)} \right| \end{aligned} \tag{14}$$

Coherence: interpretation

$$C(f) = \left| \frac{\sum_{i=1}^{N} S_{yu}^{i}(f) \cdot \sum_{i=1}^{N} S_{uy}^{i}(f)}{\sum_{i=1}^{N} S_{uu}^{i}(f) \cdot \sum_{i=1}^{N} S_{yy}^{i}(f)} \right|$$

When is a measurement reliable?

- when each data serie in N yields the same PSD's
  - i.e. when $S_{\cdot\cdot}^{i}(f) \approx S_{\cdot\cdot}^{j}(f)$ for all $i \neq j$
- then $C(f) \approx 1$
  - i.e. then relation from u to y is mostly linear for that f

Coherence: interpretation

$$C(f) = \left| \frac{\sum_{i=1}^{N} S_{yu}^{i}(f) \cdot \sum_{i=1}^{N} S_{uy}^{i}(f)}{\sum_{i=1}^{N} S_{uu}^{i}(f) \cdot \sum_{i=1}^{N} S_{yy}^{i}(f)} \right|$$

When is a measurement reliable?

- when each data serie in N yields the same PSD's
  - i.e. when $S_{\cdot\cdot}^{i}(f) \approx S_{\cdot\cdot}^{j}(f)$ for all $i \neq j$
- then $C(f) \approx 1$
  - i.e. then relation from u to y is mostly linear for that f

Hence, when $C(f) < 1$ at a certain f, there are

- non-linearities in $H(f)$
- dominant (measurement) noise sources
- other disturbances or inputs present in y

Coherence: example

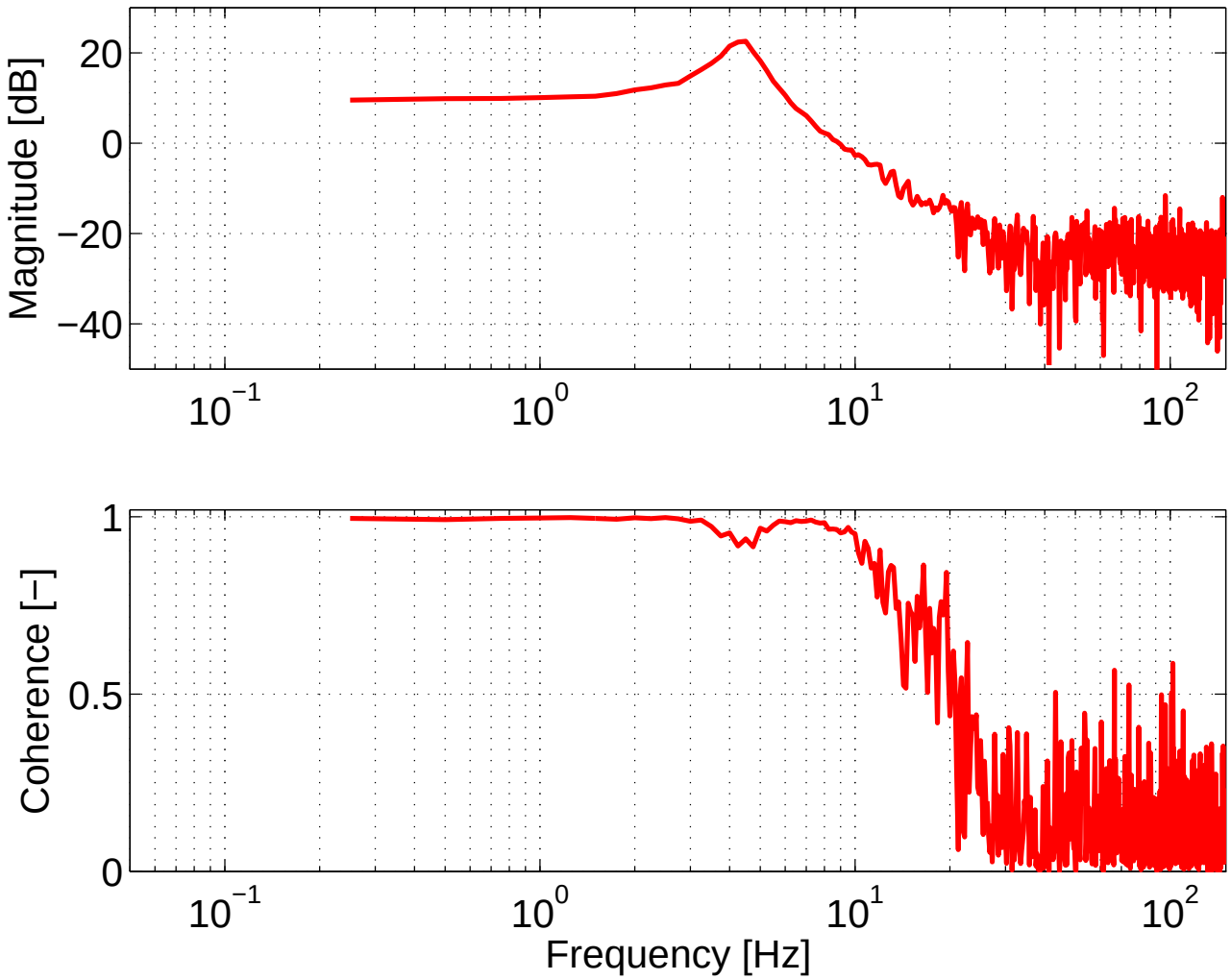
Coherence

$\Updownarrow$

signal-to-noise ratio

$C(f) < 1$ if

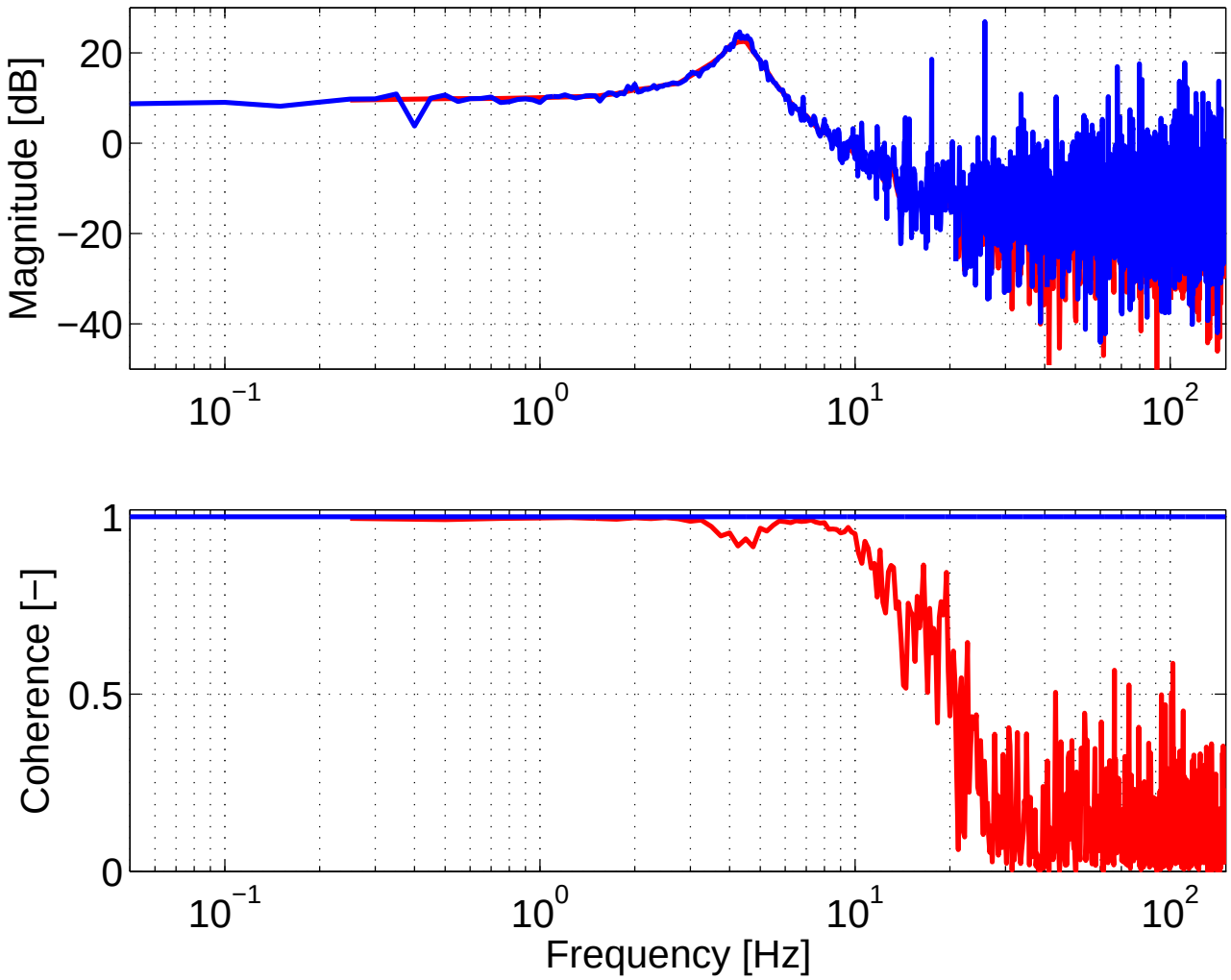
- noise is large
- 'signal' is small



Coherence: example

If $N = 1$

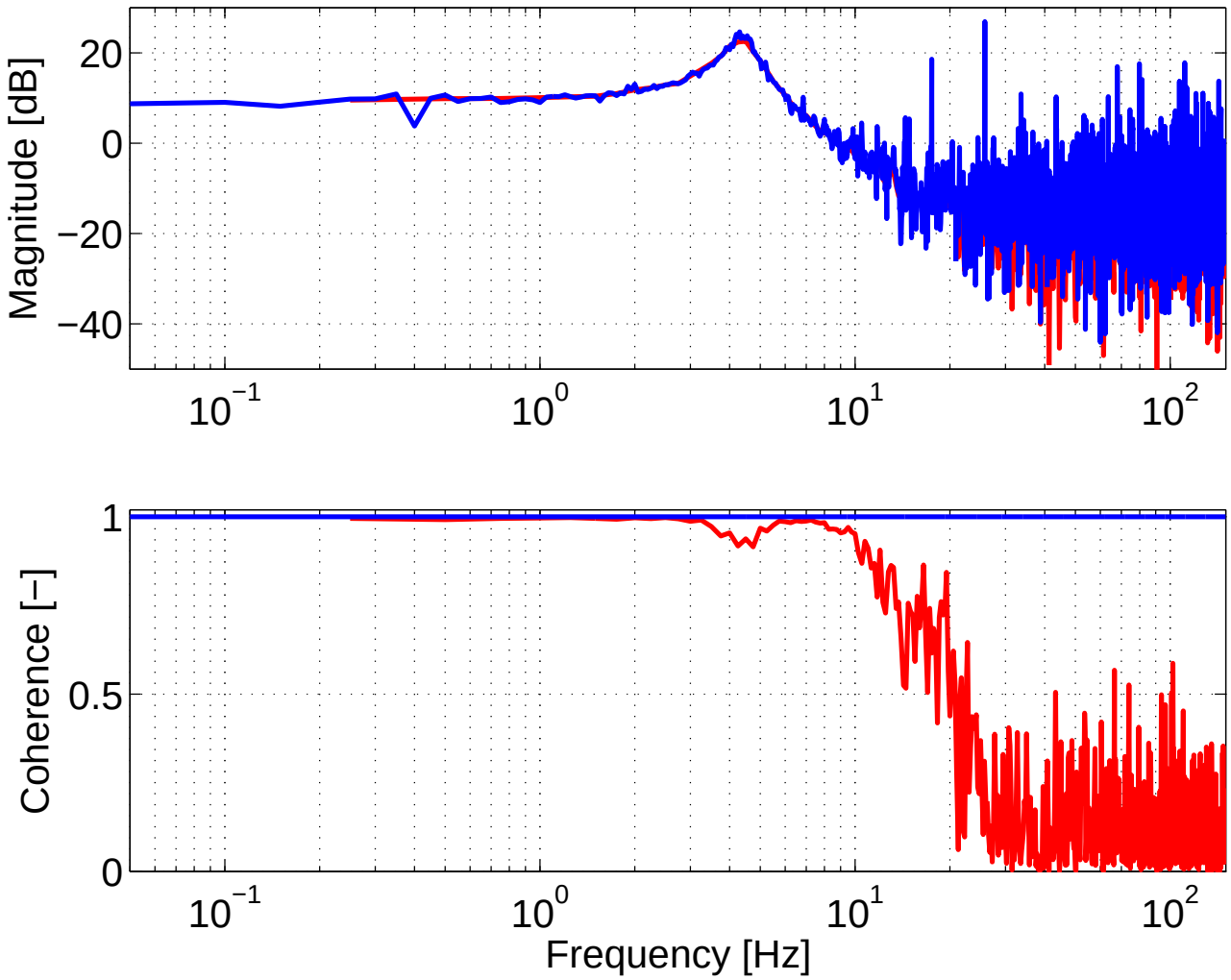
- just one frame
- no averaging
- noise is seen as linear dynamics
- $C(f) = 1$ for all f



Coherence: example

If $N = 1$

- just one frame
- no averaging
- noise is seen as linear dynamics
- $C(f) = 1$ for all f



Coherence is not ‘holy’: handle with care!

Can we also measure in closed loop?

Closed loop measurements

Open loop measurements of the plant not always a good idea:

- when plant is unstable
- when plant movement is restricted

Then plant has to be controlled before an experiment can take place.

Closed loop measurements

Open loop measurements of the plant not always a good idea:

- when plant is unstable
- when plant movement is restricted

Then plant has to be controlled before an experiment can take place.

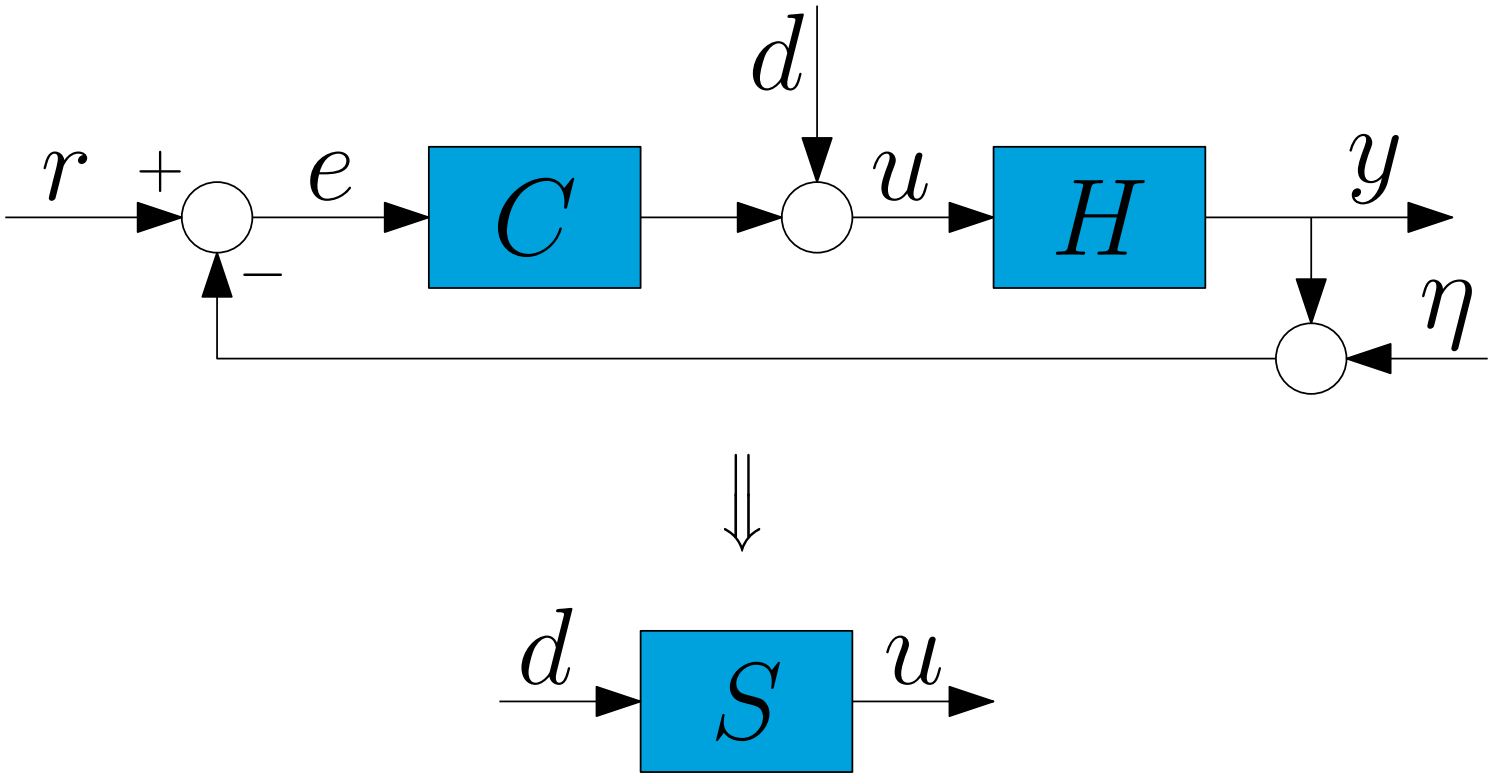
Solution: Indirect closed loop measurement:

- Given any stabilizing controller
- Measure closed loop transfer function
  - e.g. Sensitivity or Process Sensitivity
- Derive plant $H(f)$ or open loop $C(f)H(f)$ from that

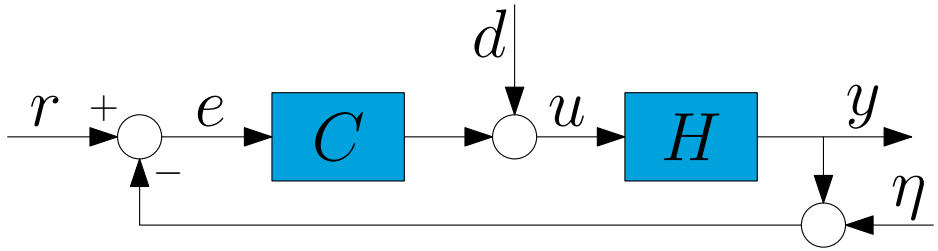
Indirect method: two-point

Two-point measurement:

- measure Sensitivity directly, then derive open-loop
- apply disturbance d, measure u

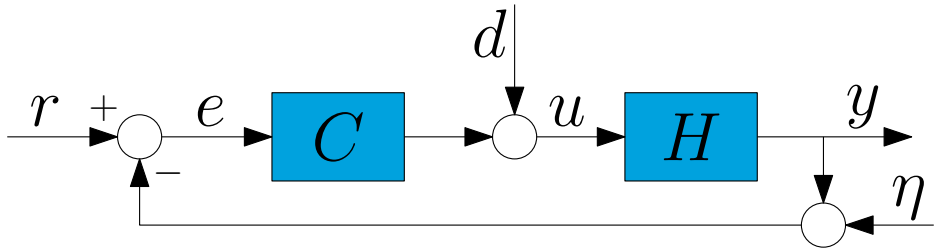


Indirect method: two-point



$$
\begin{aligned}
U(f) &= D(f) + C(f)R(f) - C(f)N(f) - C(f)H(f)U(f) \\
S_{ud}(f) &= S_{dd}(f) + C(f)S_{rd}(f) - C(f)S_{\eta d}(f) - C(f)H(f)S_{ud}(f) \\
&= \frac{1}{1 + C(f)H(f)} S_{dd}(f) + \frac{C(f)}{1 + C(f)H(f)} \left[S_{rd}(f) - S_{\eta d}(f) \right]
\end{aligned}
$$

Indirect method: two-point

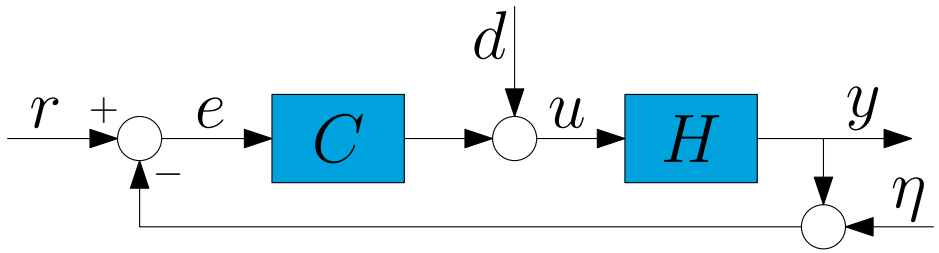


$$
\begin{aligned}
U(f) &= D(f) + C(f)R(f) - C(f)N(f) - C(f)H(f)U(f) \\
S_{ud}(f) &= S_{dd}(f) + C(f)S_{rd}(f) - C(f)S_{\eta d}(f) - C(f)H(f)S_{ud}(f) \\
&= \frac{1}{1 + C(f)H(f)} S_{dd}(f) + \frac{C(f)}{1 + C(f)H(f)} \left[S_{rd}(f) - S_{\eta d}(f) \right]
\end{aligned}
$$

When r, d and η are uncorrelated:

$$
S(f) = \frac{1}{1 + C(f)H(f)} \approx \frac{S_{ud}(f)}{S_{dd}(f)}
$$

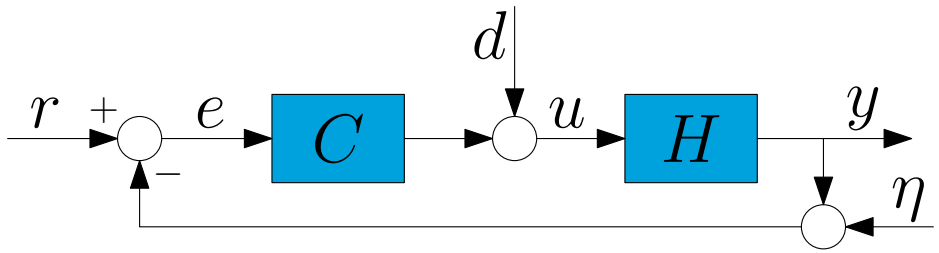
Indirect method: two-point



Or when averaging is used (as usual):

$$S(f) = \frac{1}{1 + C(f)H(f)} \approx \frac{\sum_{i=1}^{N} S_{ud}^{i}(f)}{\sum_{i=1}^{N} S_{dd}^{i}(f)}$$

Indirect method: two-point



Or when averaging is used (as usual):

$$S(f) = \frac{1}{1 + C(f)H(f)} \approx \frac{\sum_{i=1}^{N} S_{ud}^{i}(f)}{\sum_{i=1}^{N} S_{dd}^{i}(f)}$$

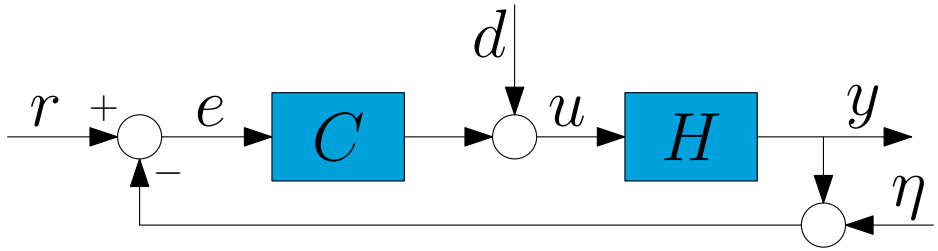
Then the open-loop is:

$$L(f) = C(f)H(f) = S^{-1}(f) - 1$$

And since $C(f)$ is known (or can be measured):

$$H(f) = C^{-1}(f)\left[S^{-1}(f) - 1\right]$$

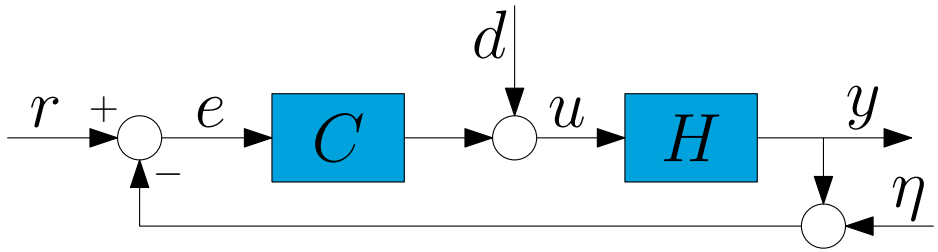
Indirect method: two-point



How to obtain $S_{rd}(f) \approx 0$ and $S_{\eta d}(f) \approx 0$?

- choose $r = 0$
- choose d as white noise

Indirect method: two-point



How to obtain $S_{rd}(f) \approx 0$ and $S_{\eta d}(f) \approx 0$?

- choose $r = 0$
- choose d as white noise

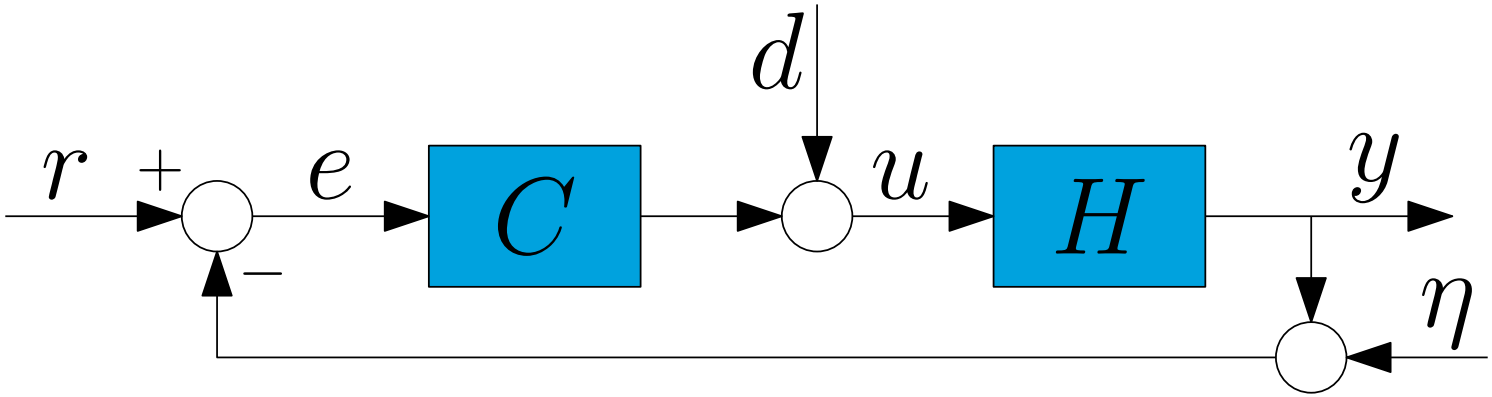
Otherwise

$$S_{ud}(f) = \frac{1}{1 + C(f)H(f)} S_{dd}(f) + \underbrace{\frac{C(f)}{1 + C(f)H(f)} \left[S_{rd}(f) - S_{\eta d}(f) \right]}_{\text{bias}}$$

Results in a wrong estimation of the Sensitivity $S(f)$.

Indirect method: three-point

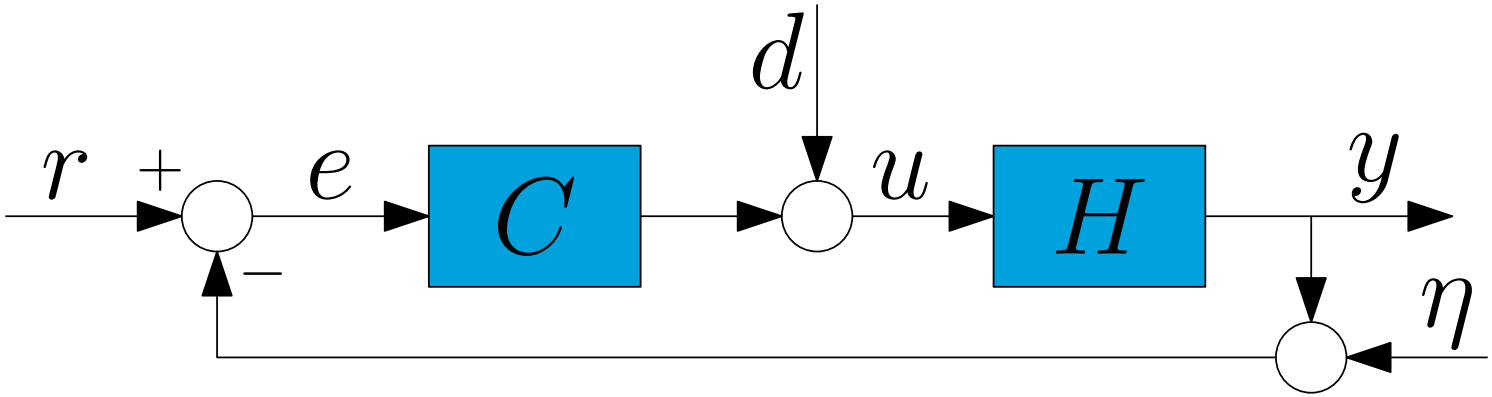
What if $C(f)$ is not (exactly) known?



- measure both Sensitivity and Process Sensitivity directly
- apply d, measure both u and e (or y)

Indirect method: three-point

What if $C(f)$ is not (exactly) known?

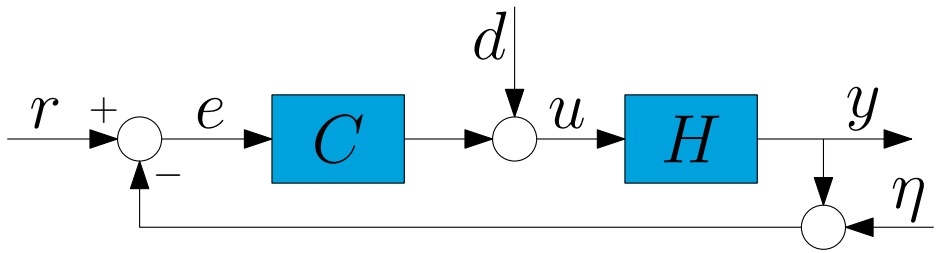


$$\left.\begin{array}{l} \frac{u}{d} = S(f) \\ \frac{e}{d} = -PS(f) \end{array}\right\} \Rightarrow \quad \frac{PS(f)}{S(f)} = \frac{H(f)}{1+C(f)H(f)} \cdot \frac{1+C(f)H(f)}{1} = H(f)$$

- No (separate) measurement of $C(f)$ necessary
- Check coherence of both $d \rightarrow u$ and $d \rightarrow e$

Note: make sure that r, d and η are uncorrelated
Be aware of bias otherwise!

Bad idea...



Some people are tempted to do a direct measurement in closed loop...

- apply d, measure u and y, calculate $H(f) = \frac{S_{yu}(f)}{S_{uu}(f)}$

Problem: η and u are *correlated*:

$$S_{yu}(f) = S_{\eta u}(f) + H(f)S_{uu}(f), \quad S_{\eta u}(f) \neq 0,$$

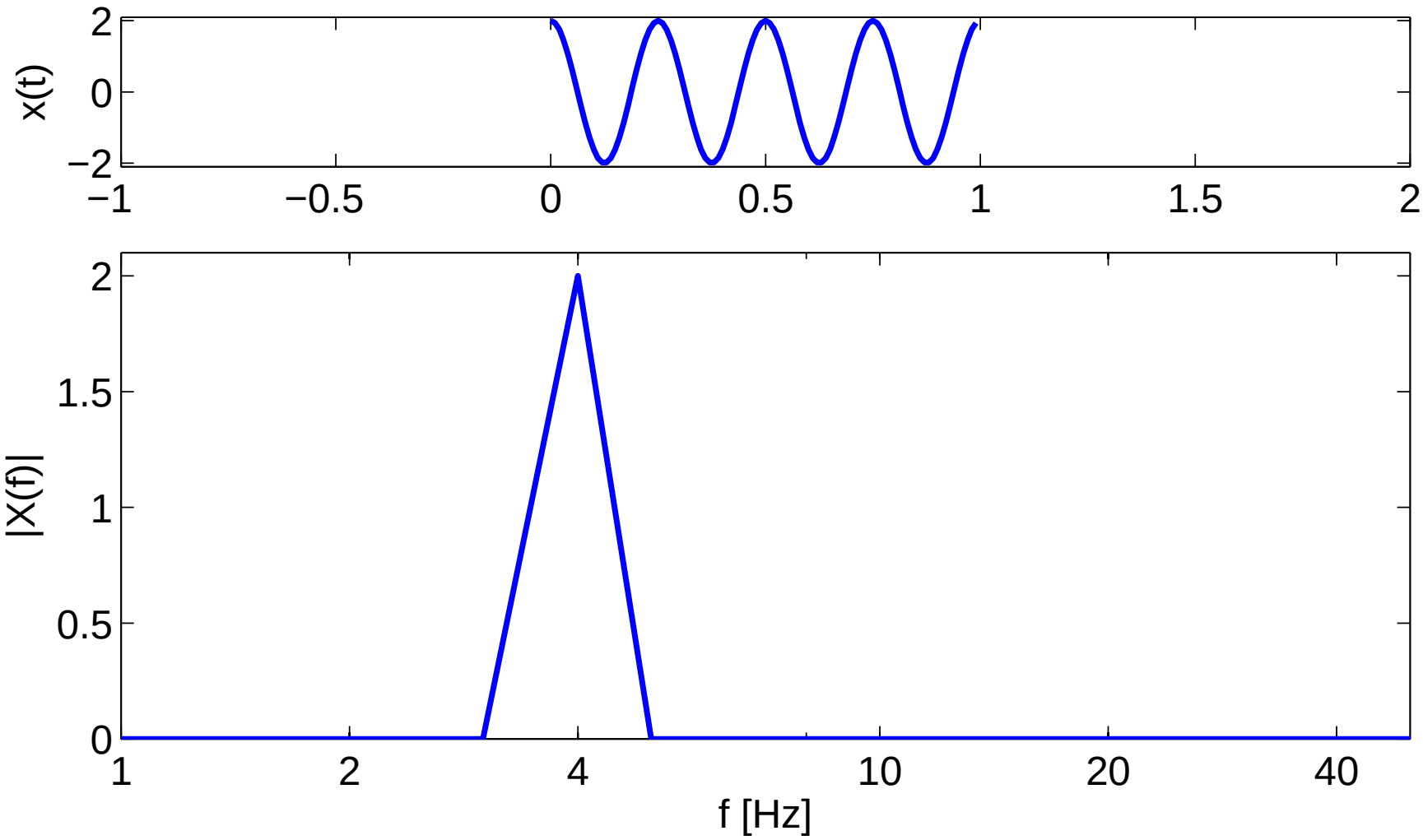
since $U(f) = D(f) - C(f)N(f) - C(f)H(f)U(f)$ we have

$$\begin{aligned} S_{u\eta}(f) &= S(f)S_{d\eta}(f) - CS(f)S_{\eta\eta}(f) \quad S_{d\eta}(f) \approx 0 \\ S_{yu}(f) &= H(f)S_{uu}(f) - \underbrace{CS(f)S_{\eta\eta}(f)}_{\text{bias}} \end{aligned}$$

Practical tricks and issues to be aware of

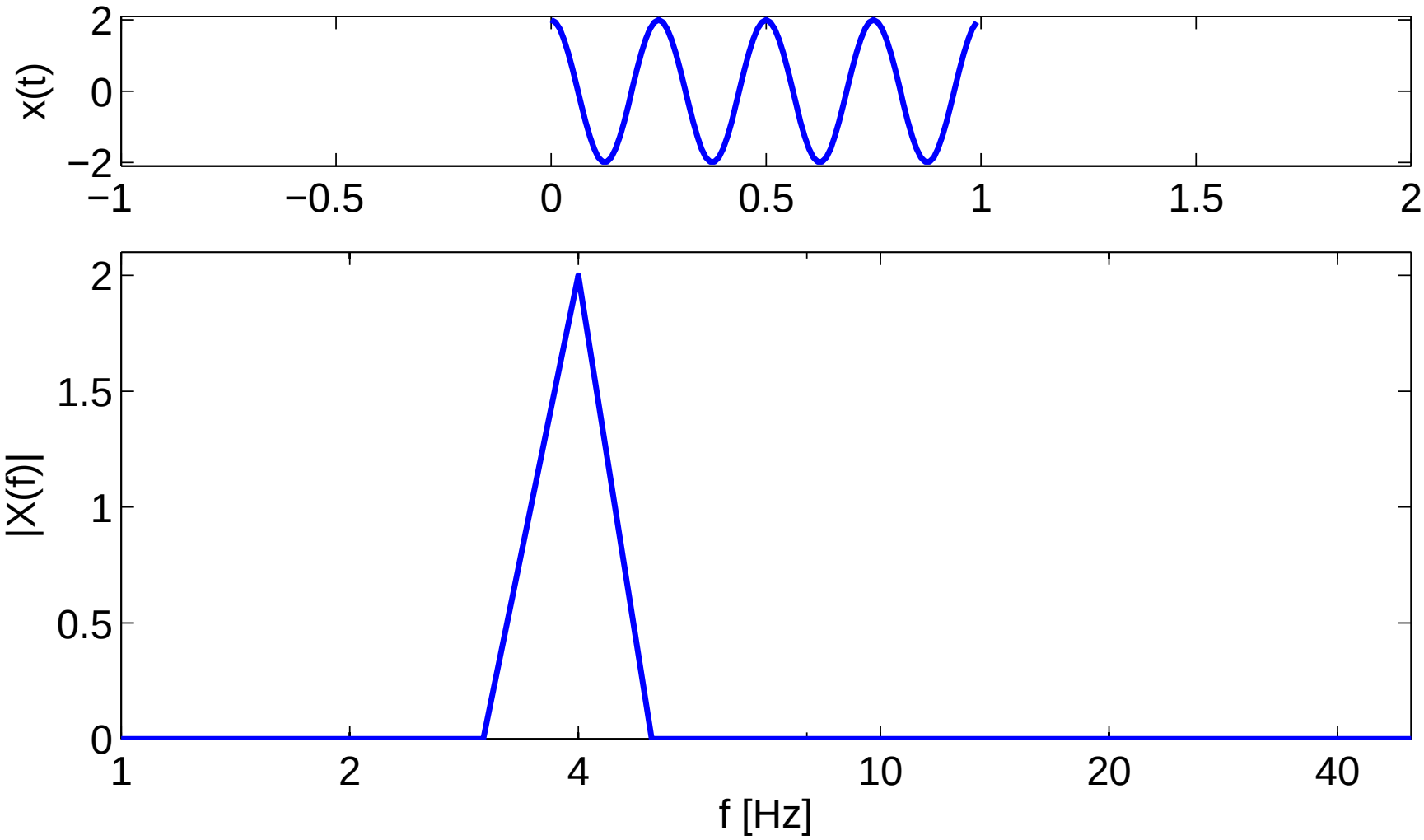
Leakage

Suppose: $x(t) = A\cos(2\pi ft)$, with $t \in [0, T]$ and $f \cdot T$ an integer



Leakage

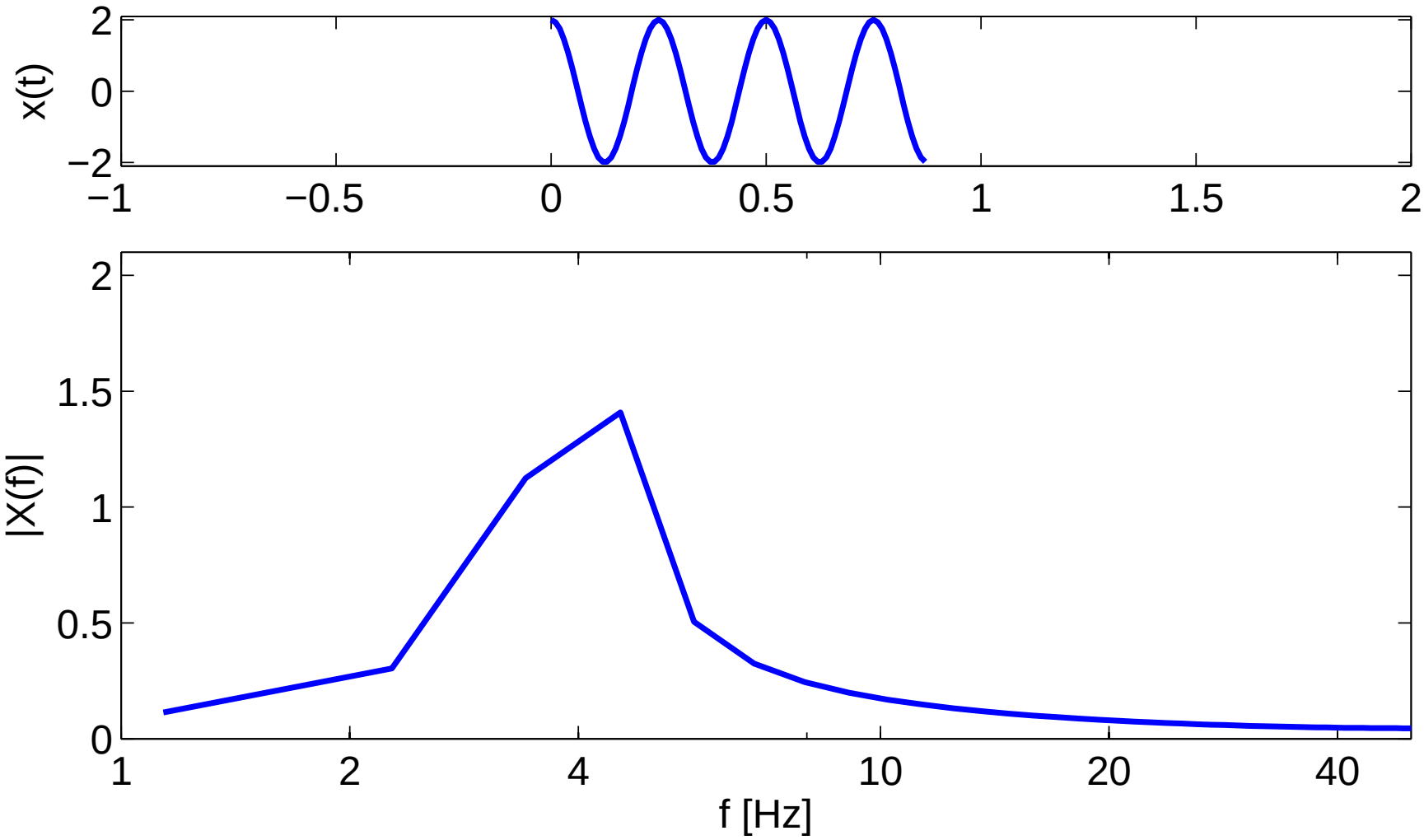
Suppose: $x(t) = A\cos(2\pi ft)$, with $t \in [0, T]$ and $f \cdot T$ an integer



Only then $x(t)$ is exactly periodic: spectrum is exact.

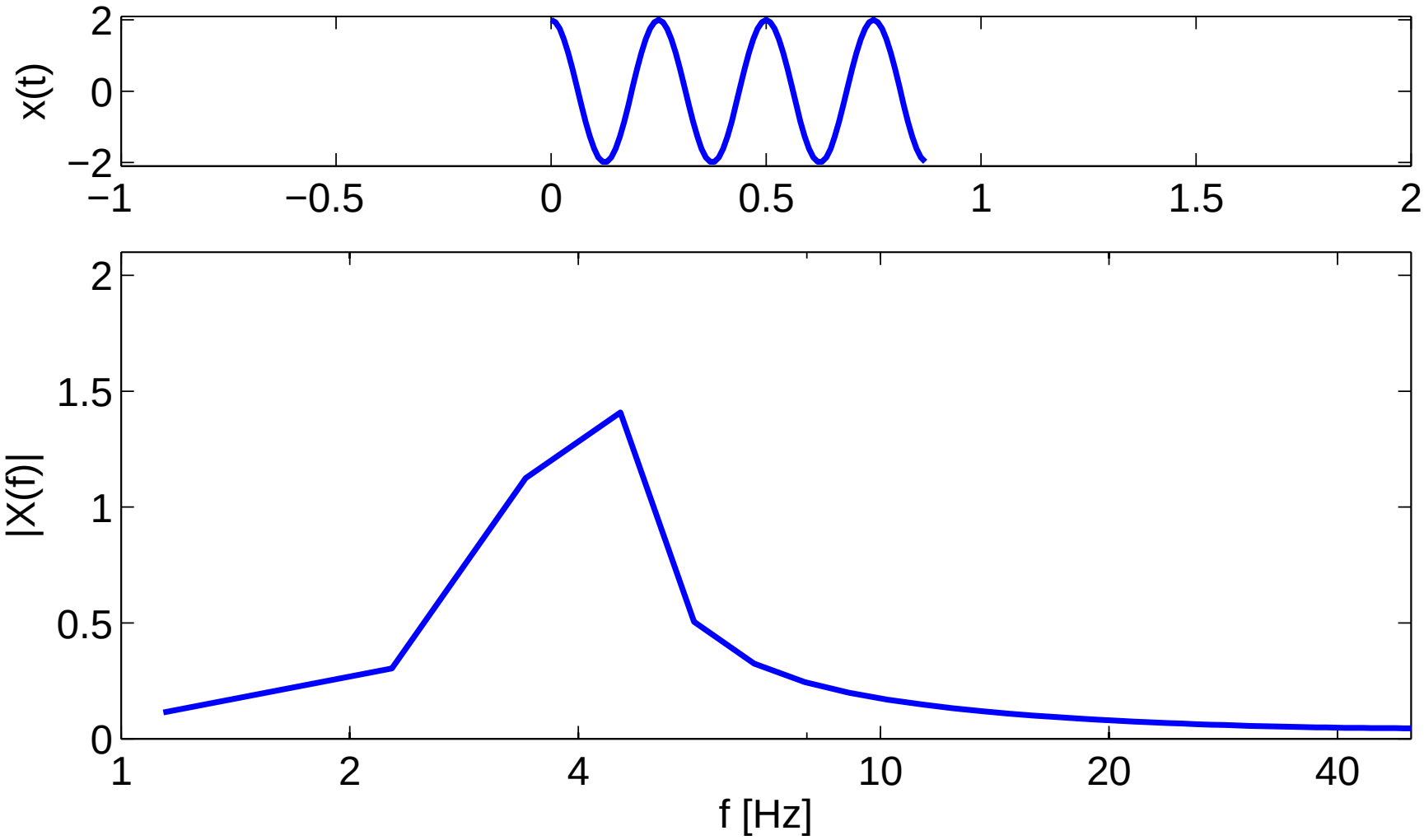
Leakage

Suppose: $x(t) = A\cos(2\pi ft)$, with $t \in [0, T]$ and T is random



Leakage

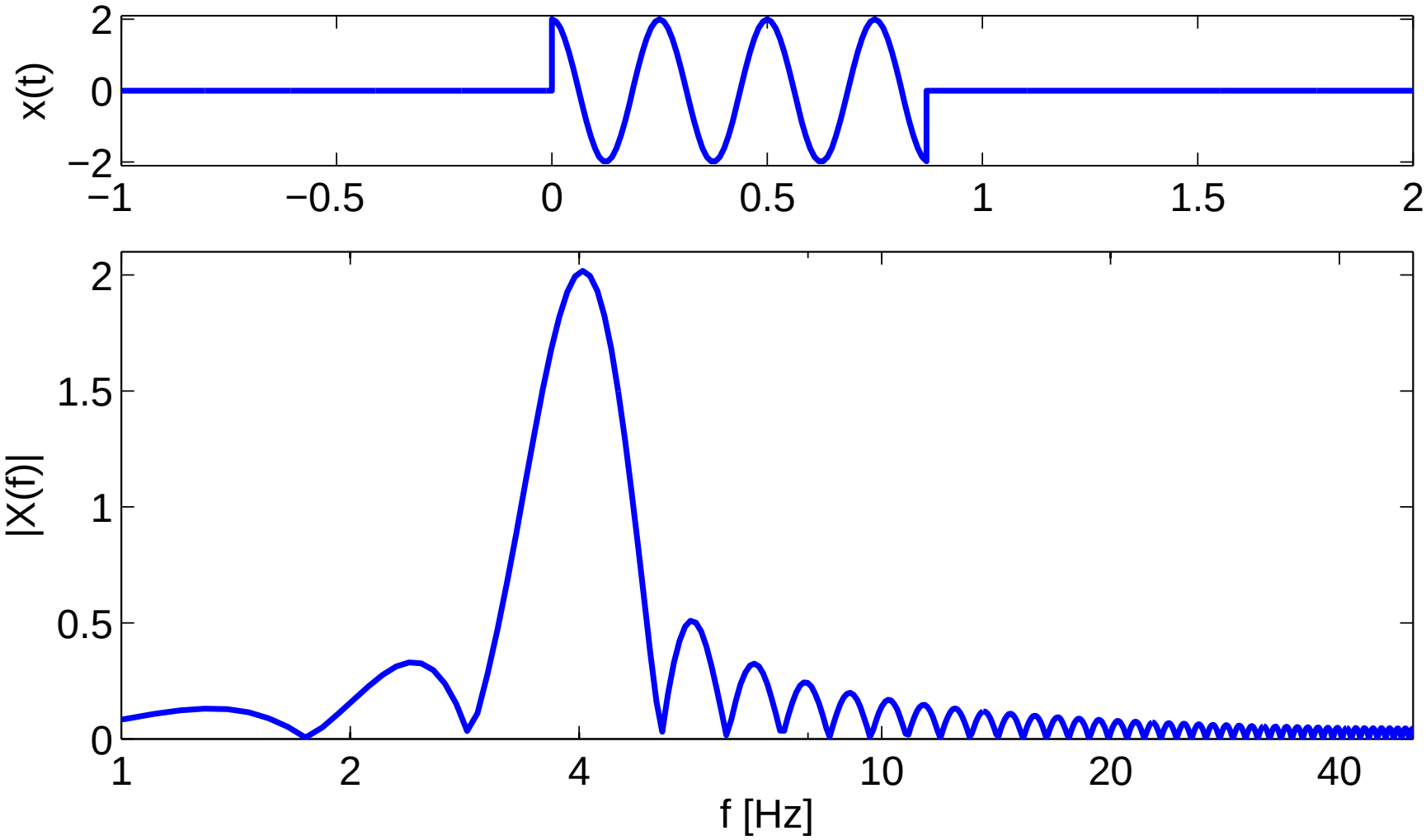
Suppose: $x(t) = A\cos(2\pi ft)$, with $t \in [0, T]$ and T is random



Then f is not part of spectrum ($x(t)$ is aperiodic): *leakage*.

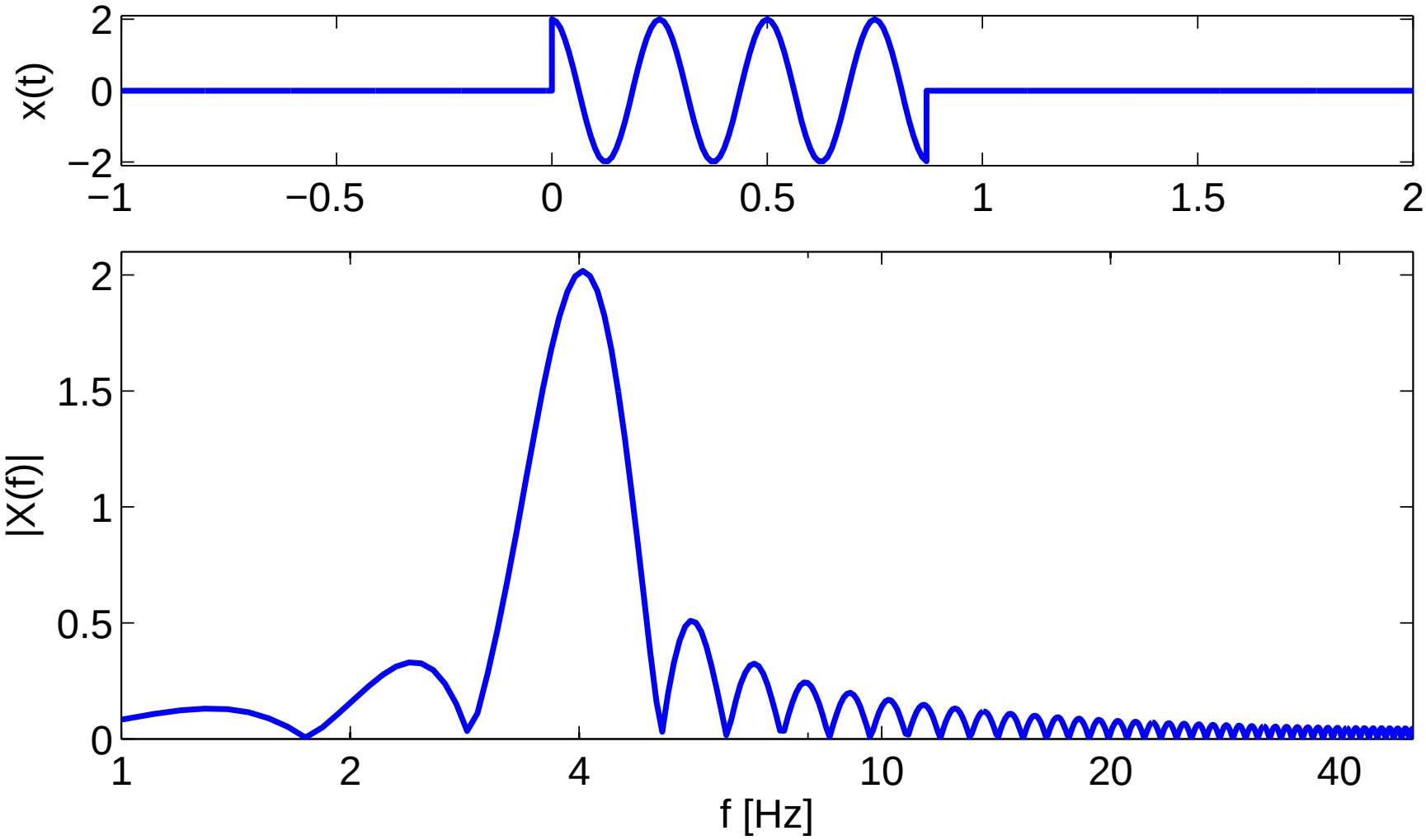
Zero-padding

Solution: add zeros to increase frequency resolution



Zero-padding

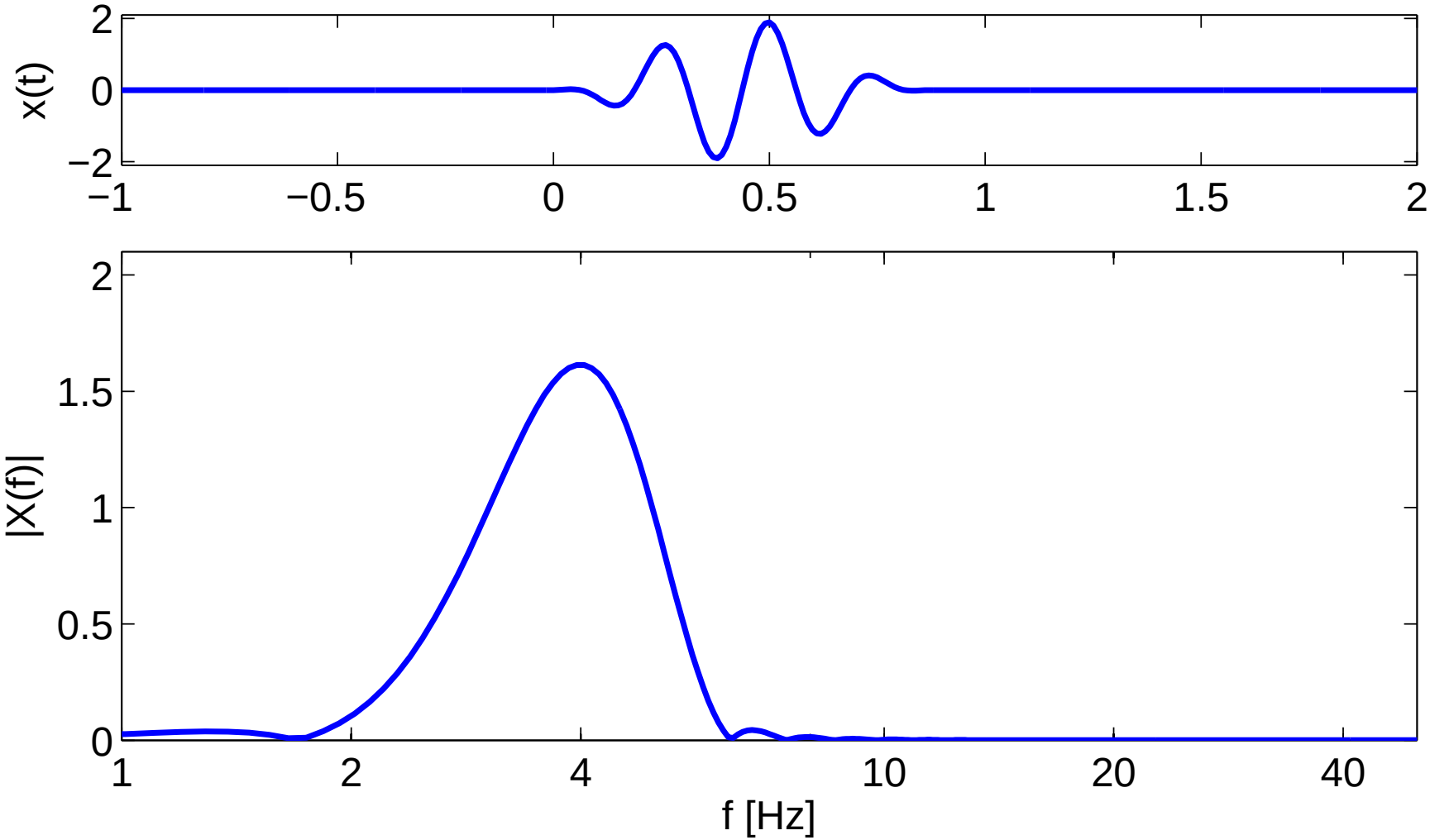
Solution: add zeros to increase frequency resolution



Zero-padding increases resolution, but introduces *ripple*.

Windowing

Solution: apply window to remove jumps in time-signal

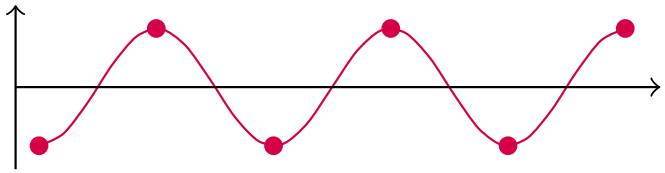


Reduced ripple and leakage, but spectrum is still not exact.

Sampling

In practice signals are sampled with sample frequency F_s

- at least two samples are needed to identify an harmonic:

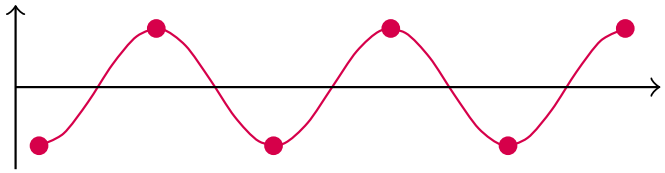


  Highest frequency recognized in Fourier analysis is thus $F_s/2$
- $F_s/2$ is known as the *Nyquist frequency*

Sampling

In practice signals are sampled with sample frequency F_s

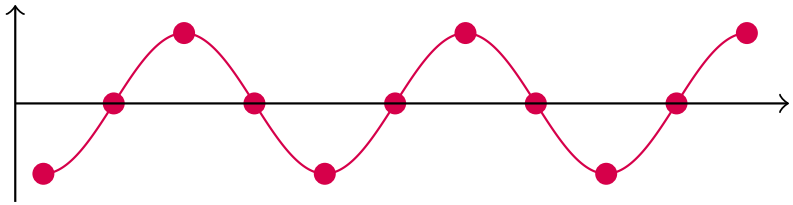
- at least two samples are needed to identify an harmonic:

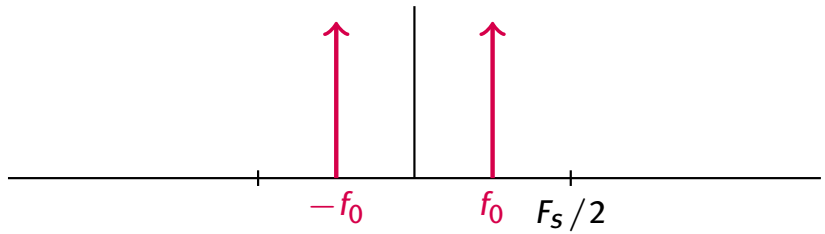


  Highest frequency recognized in Fourier analysis is thus $F_s/2$

- $F_s/2$ is known as the *Nyquist frequency*

Spectrum for sampled signals is repetitive

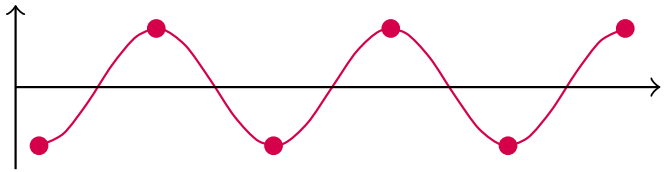




Sampling

In practice signals are sampled with sample frequency F_s

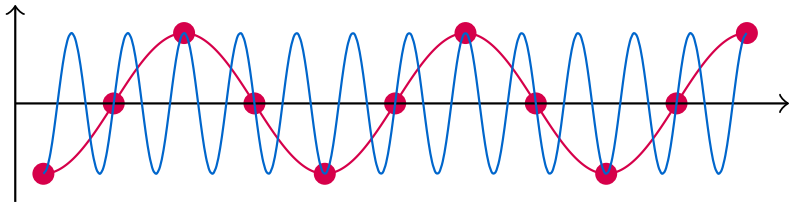
- at least two samples are needed to identify an harmonic:

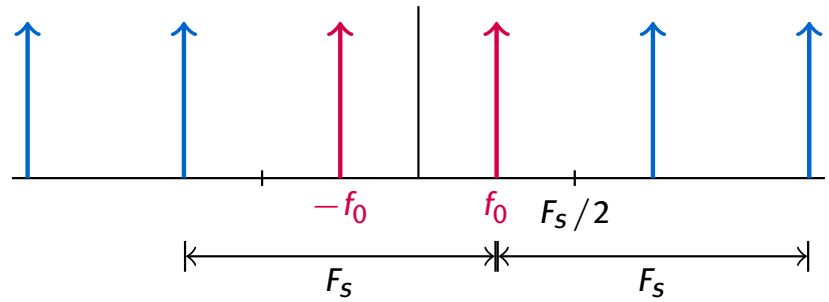


  Highest frequency recognized in Fourier analysis is thus $F_s/2$

- $F_s/2$ is known as the *Nyquist frequency*

Spectrum for sampled signals is repetitive

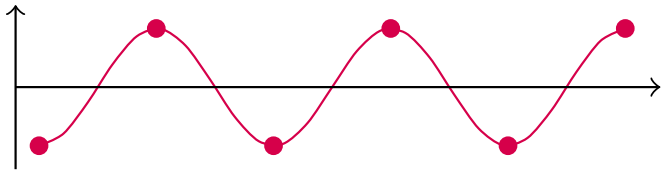




Sampling

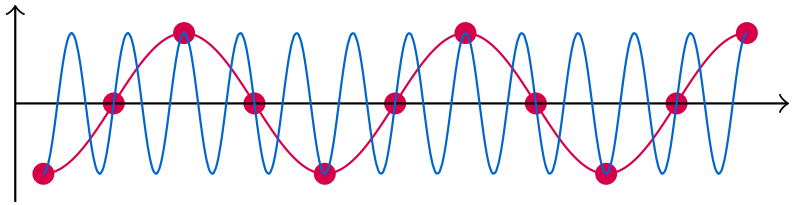
In practice signals are sampled with sample frequency F_s

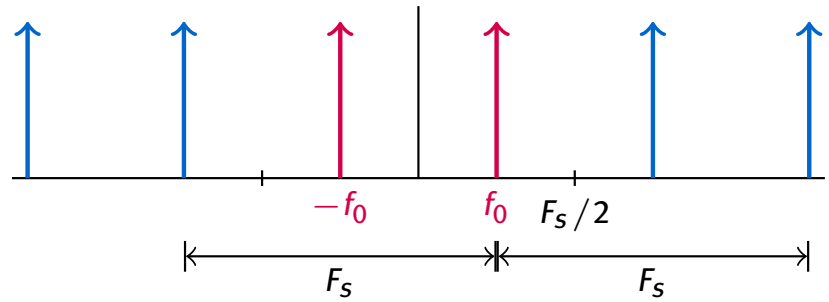
- at least two samples are needed to identify an harmonic:



  Highest frequency recognized in Fourier analysis is thus $F_s/2$
- $F_s/2$ is known as the *Nyquist frequency*

Spectrum for sampled signals is repetitive



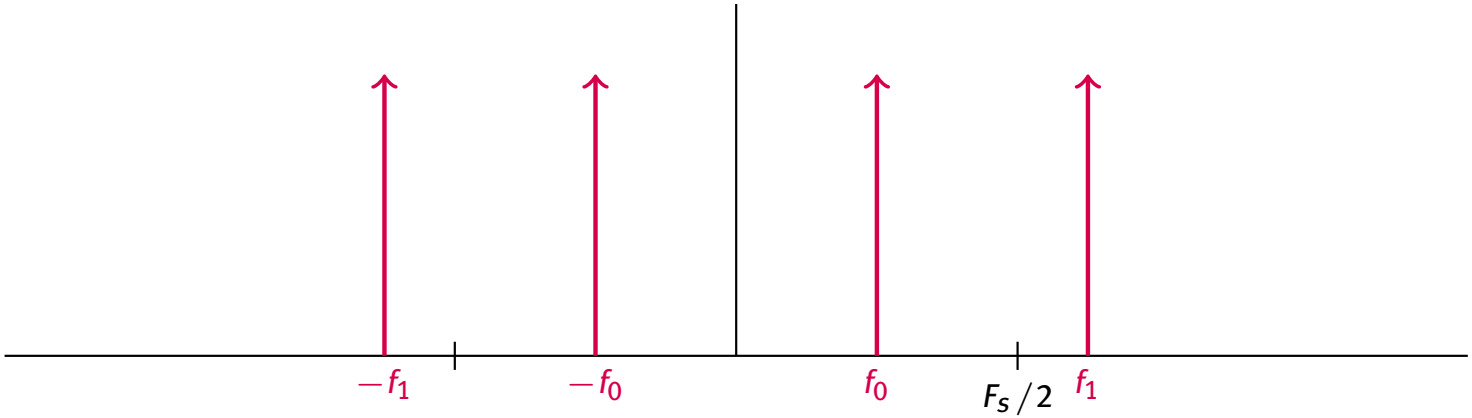


- only frequencies between $-F_s/2$ and $F_s/2$ are relevant

Aliasing

What happens if frequencies higher than $F_s/2$ are present?

- these frequencies are mapped (or folded) to lower frequencies

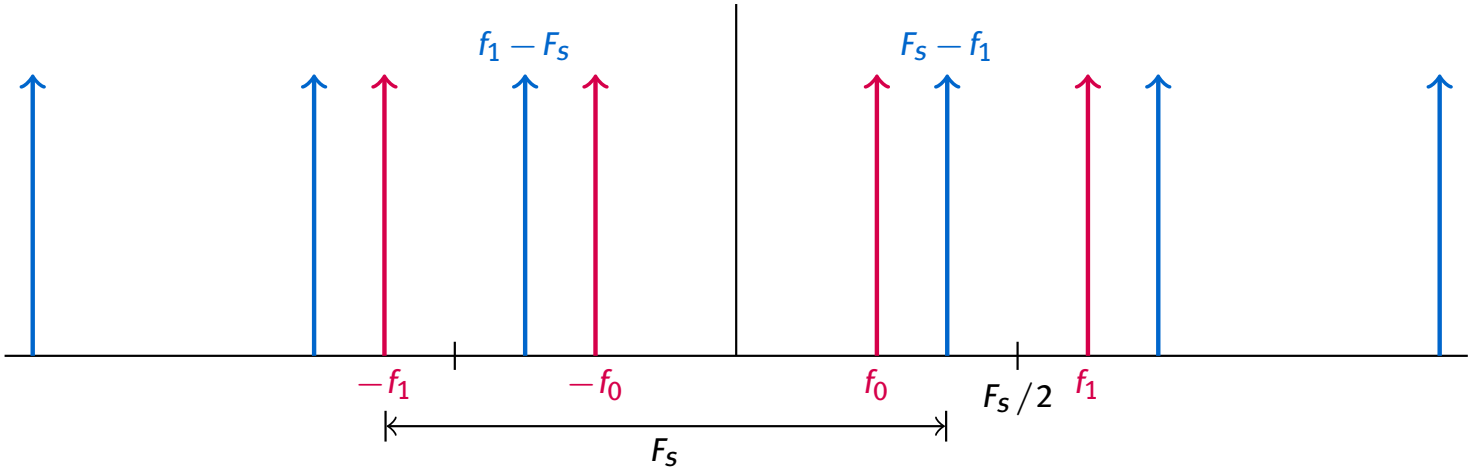


Note: Aliased frequencies will change when sample frequency F_s changes, while f_0 remains the same. This can be used to check whether aliasing occurs.

Aliasing

What happens if frequencies higher than $F_s/2$ are present?

- these frequencies are mapped (or folded) to lower frequencies

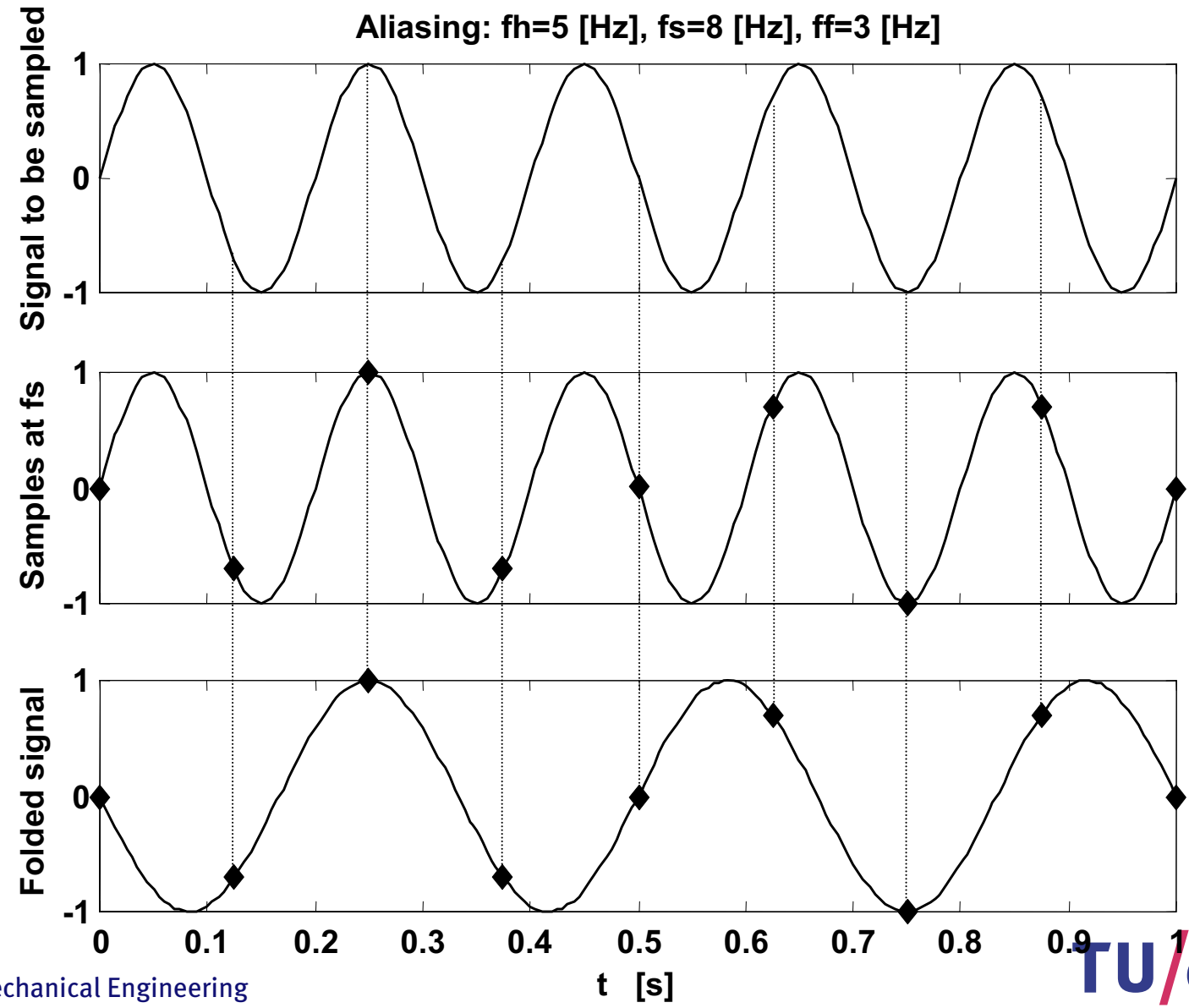


- Hence, f_1 is wrongly identified as $F_s - f_1$

Note: Aliased frequencies will change when sample frequency F_s changes, while f_0 remains the same. This can be used to check whether aliasing occurs.

Aliasing

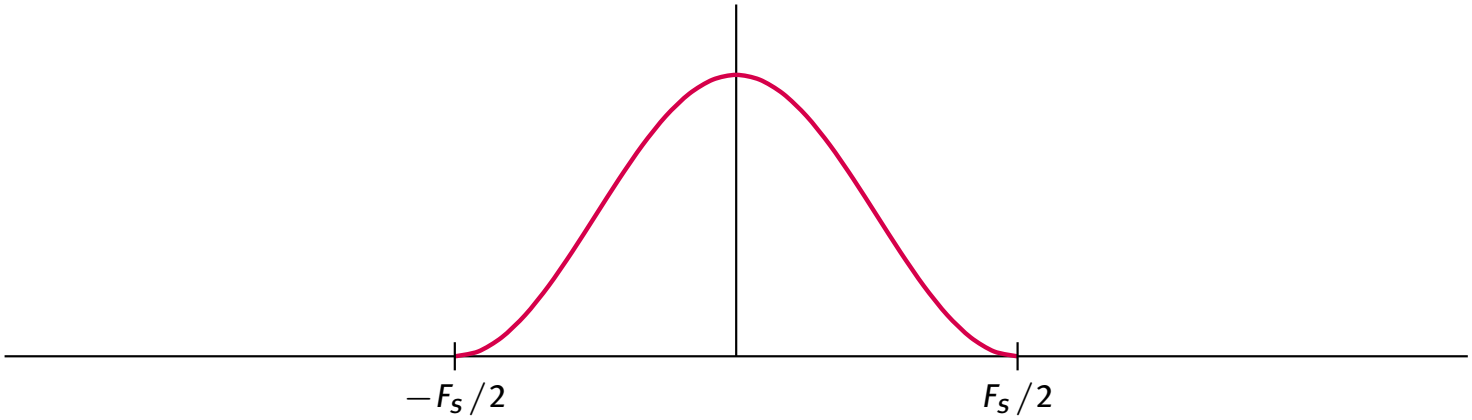
What does that mean in time domain?



Aliasing

Hence, to prevent aliasing:

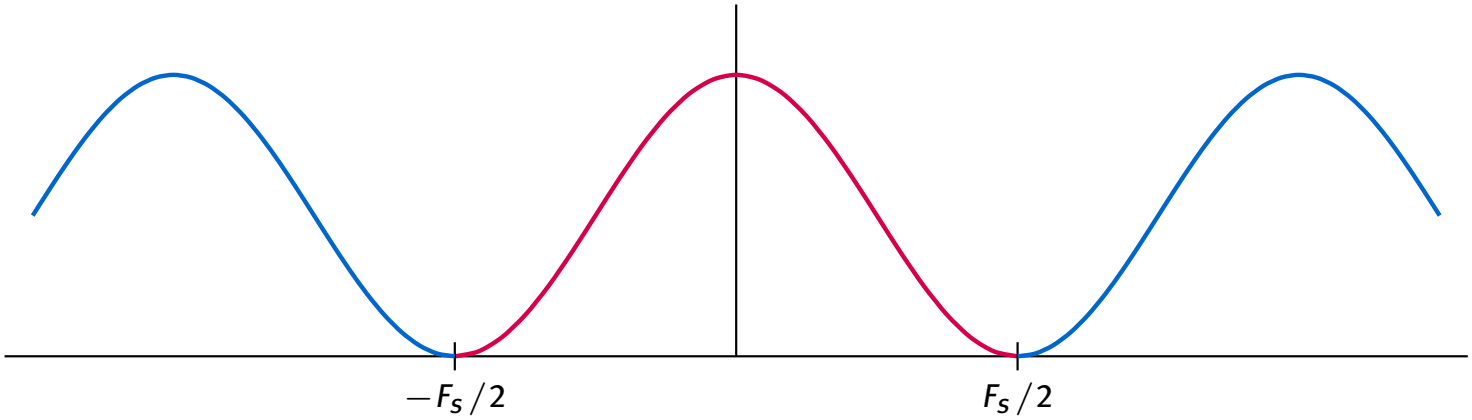
- Sampling frequency should be at least *twice* the highest frequency present in the signal



Aliasing

Hence, to prevent aliasing:

- Sampling frequency should be at least *twice* the highest frequency present in the signal



- Known as *Shannon* sampling theorem

And now it’s your turn!