Random signals II.

Jan Černocký

ÚPGM FIT VUT Brno, cernocky@fit.vutbr.cz

Temporal estimate of autocorrelation coefficients

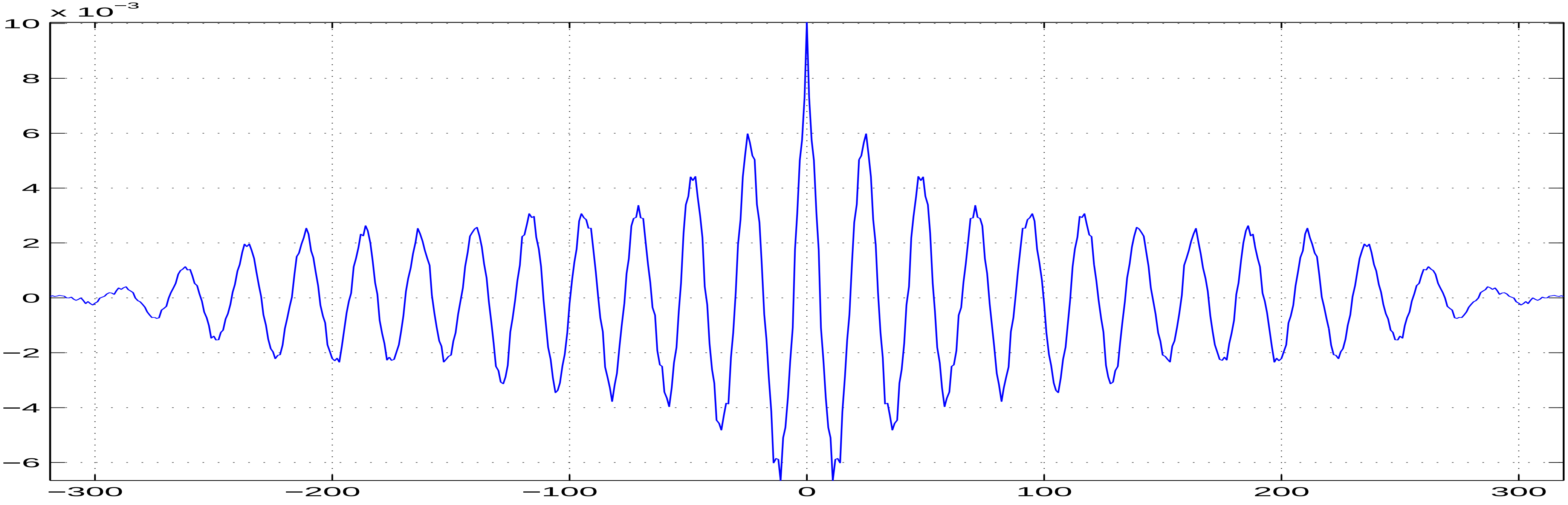
for ergodic discrete-time random process.

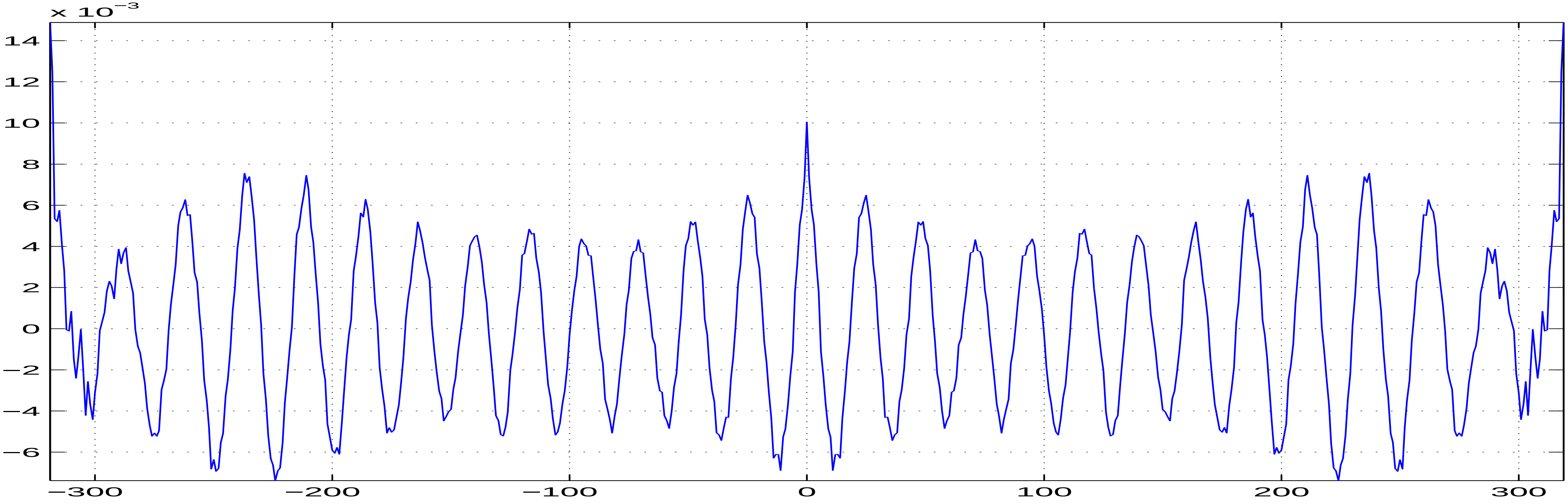
- $$\hat{R}[k] = \frac{1}{N} \sum_{n=0}^{N-1} x[n]x[n+k],$$

  where N is the number of samples we have at our disposal, is called **biased estimate**. When we "pull" the signals from each other, $R(k)$ is estimated from only $N-k$ samples, but we are always dividing by N, so that the values will decrease toward the edges.

- $$\hat{R}[k] = \frac{1}{N-|k|} \sum_{n=0}^{N-1} x[n]x[n+k],$$

  is **unbiased estimate**, where the division is done by the actual number of overlapping samples. On the other hand, the coefficients on the edges: $k \rightarrow N-1, \quad k \rightarrow -N+1$ are very badly estimated, as only few samples are available for their estimation. Therefore, we prefer biased estimates.


x 10^-3
10
8
6
4
2
0
−2
−4
−6
−300
−200
−100
0
100
200
300


x 10^-3
14
12
10
8
6
4
2
0
−2
−4
−6
−300
−200
−100
0
100
200
300

Power spectral density – PSD – continuous time

We want to study the behavior in frequency also for random processes, but:

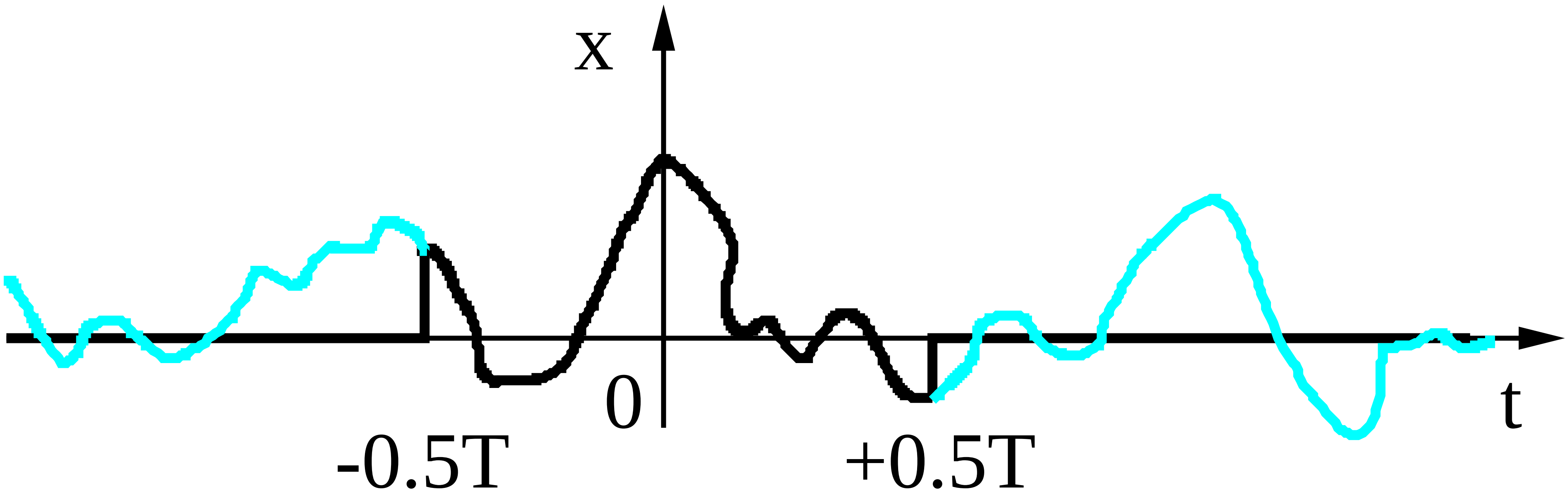
- we can not use Fourier series, as random signals are not periodic.
- we can not use Fourier transform, as random signals have infinite energy (FT can be applied to such signals, but only to some special cases).

We will consider only ergodic random signals, one realization: $x(t)$.

Power spectral density – PSD – derivation.

- let us define an interval of length T: from $-T/2$ till $T/2$:

$$x_T(t) = \begin{cases} x(t) & \text{for} \quad |t| < T/2 \\ 0 & \text{elsewhere} \end{cases}$$



- We will define a Fourier image:

$$X_T(j\omega) = \int_{-\infty}^{+\infty} x_T(t) e^{-j\omega t} dt$$

- We will define energy spectral density (see the lecture on Fourier transform):

$$\int_{-\infty}^{\infty} x_T^2(t)dt = \ldots = \frac{1}{2\pi}\int_{-\infty}^{\infty} X_T(j\omega)X_T(-j\omega)d\omega = \int_{-\infty}^{\infty} L_T(j\omega)d\omega$$

$L_T(\omega)$ will be called **(double-sided) energy spectral density**

$$L_T(j\omega) = \frac{|X(j\omega)|^2}{2\pi}$$

- We will try to "stretch" T till ∞. In this case however, the energy (and also its density) would grow to infinity. We will therefore define **(double-sided) power spectral density** by dividing the energy by T (analogy: $P = E/T$):

$$G_T(j\omega) = \frac{L_T(j\omega)}{T}$$

- now, the "stretching" can be done and power spectral density can be computed not only for the interval of length T, but for the whole signal:

$$G(j\omega) = \lim_{T\to\infty} G_T(j\omega) = \lim_{T\to\infty} \frac{|X_T(j\omega)|^2}{2\pi T}$$

Properties of $G(j\omega)$

- For spectral function of a real signal $X_T(j\omega)$:

$$X_T(j\omega) = X_T^\star(-j\omega)$$

$G(j\omega)$ is given by squares of its values so that it will be **purely real and even**.

- The mean power of random process in an interval $[\omega_1,\ \omega_2]$ can be computed by:

$$P_{[\omega_1,\ \omega_2]} = \int_{\omega_1}^{\omega_2} G(j\omega)d\omega + \int_{-\omega_2}^{-\omega_1} G(j\omega)d\omega = 2\int_{\omega_1}^{\omega_2} G(j\omega)d\omega.$$

- And the **total** mean power of random process is expressed as:

$$P = \int_{-\infty}^{+\infty} G(\omega)d\omega$$

- In communication technology, often the mean value $a = 0$. Then, the mean power is equal to the variance: $P = D$ and the effective value is equal to the standard deviation: $X_{ef} = \sigma$.

Wiener-Chinchin's equations

The power spectral density is linked to the auto-correlation function by $R(\tau)$ by FT (usually, we compute PSD in this way – it is more tractable than increasing $T \to \infty$):

$$G(j\omega) = \frac{1}{2\pi} \int_{-\infty}^{+\infty} R(\tau) e^{-j\omega\tau} d\tau$$

$$R(\tau) = \int_{-\infty}^{+\infty} G(j\omega) e^{+j\omega\tau} d\omega$$

Power spectral density – discrete time

PSD of discrete-time random process will be defined directly using auto-correlation coefficients (note the terminology: auto-correlation **function** $R(\tau)$ for continuous time but auto-correlation **coefficients** $R[k]$ for discrete time):

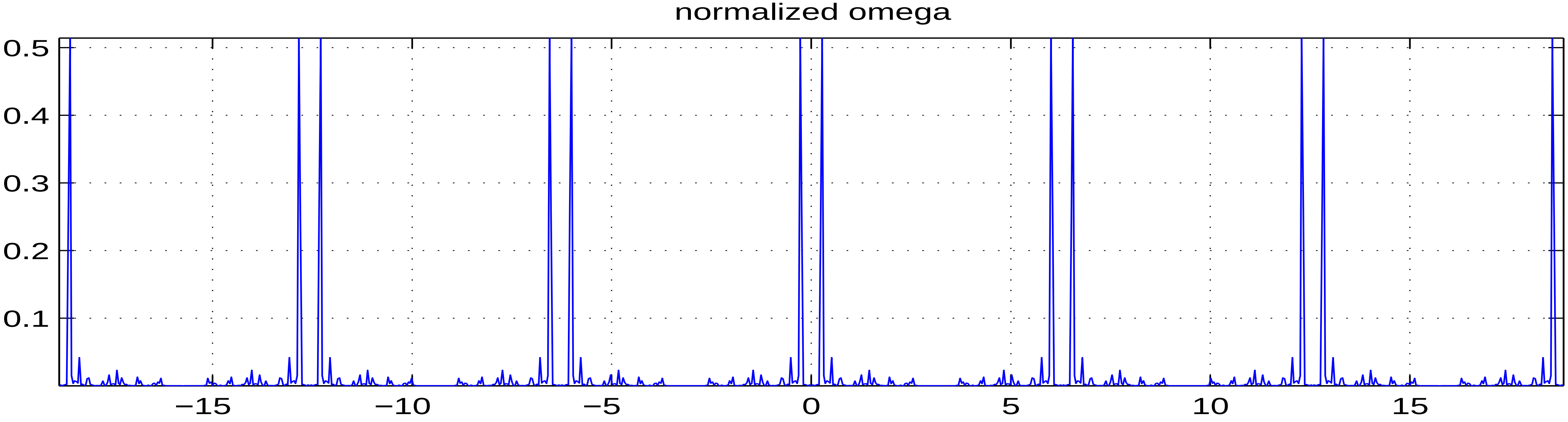
$$G(e^{j\omega}) = \sum_{k=-\infty}^{\infty} R[k]e^{-j\omega k}$$

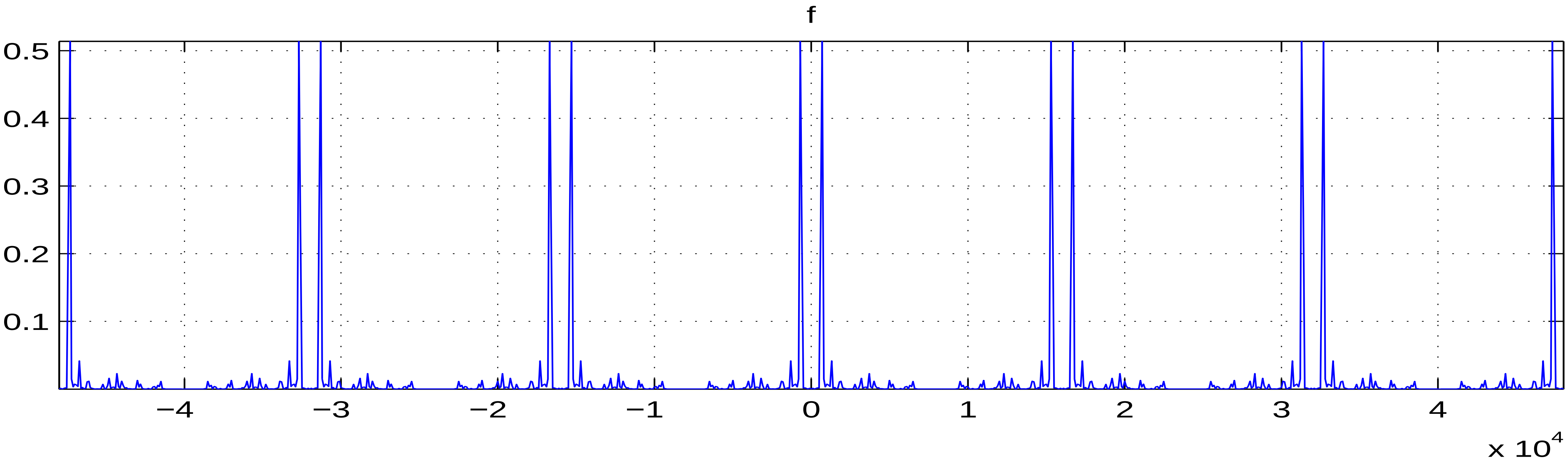
(question: which angular frequency is ω in this equation ?). $G(e^{j\omega})$ is Discrete-time Fourier transform (DTFT) of auto-correlation coefficients. In case we estimate these (ensemble or temporal estimate), we can compute $G(e^{j\omega})$.

Back from $G(e^{j\omega})$ to auto-correlation coefficients (inverse DTFT):

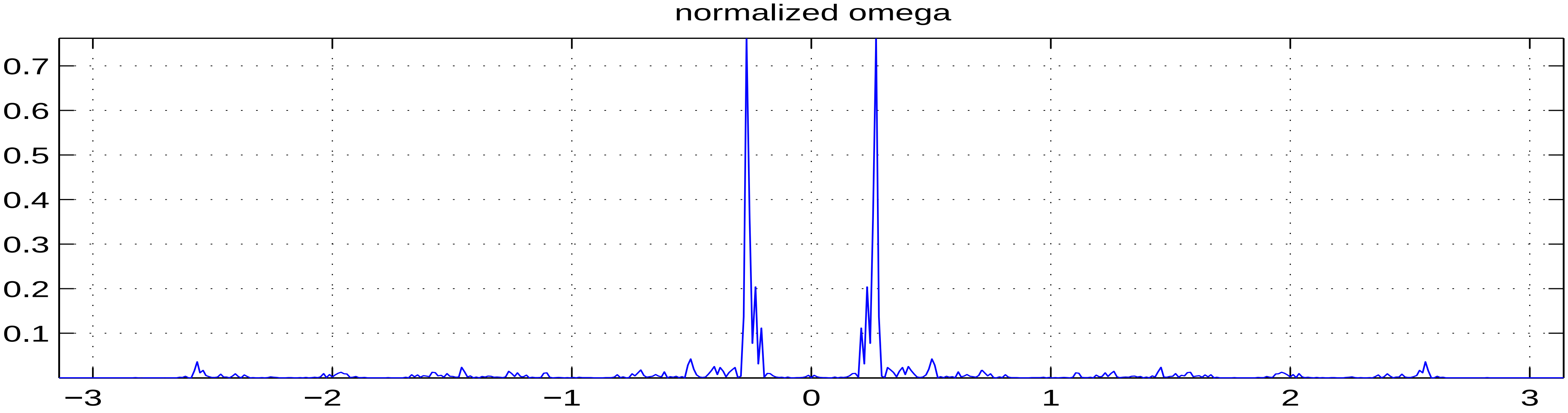
$$R[k] = \frac{1}{2\pi}\int_{-\pi}^{\pi} G(e^{j\omega})e^{+j\omega k}d\omega$$

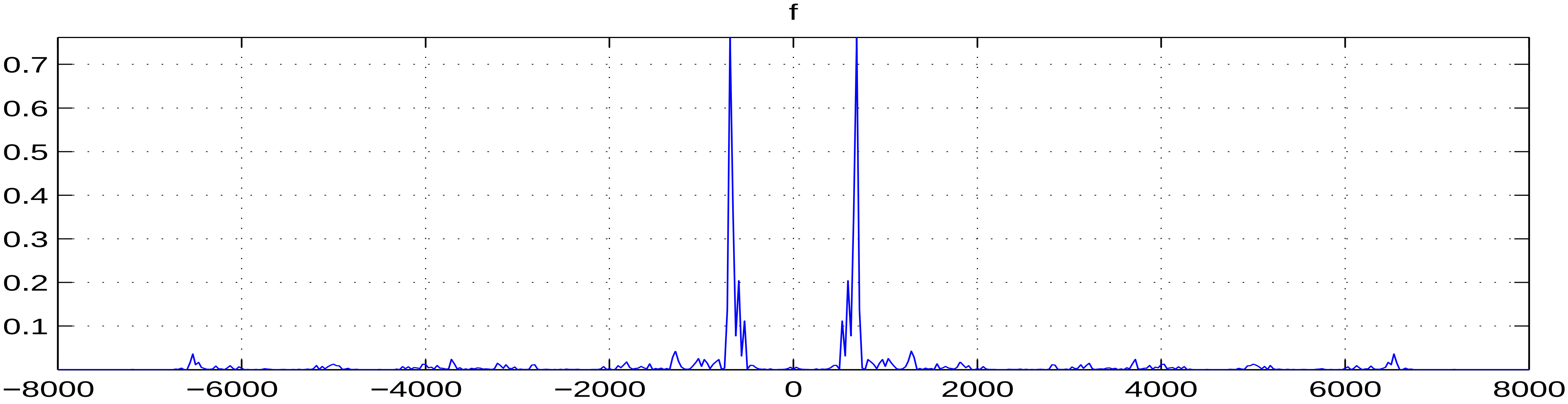
Example: flowing water ($R[k]$ estimated from one realization):





zoom from $-Fs/2$ to $Fs/2$:





$\Rightarrow$ the flowing water has a peak in the power at 700 Hz (a resonance of tube in my ex-apartment ?)

Properties of $G(e^{j\omega})$ are again given by standard properties of DTFT-image of a real signal:

- auto-correlation coefficients are real, therefore

$$G(e^{j\omega}) = G^{\star}(e^{-j\omega}),$$

- autocorrelation coefficients are symmetric (even), so that $G(e^{j\omega})$ will be purely real.
- by combining the two properties above, we can conclude, that $G(e^{j\omega})$ will be real and even, similarly to $G(j\omega)$.
- it is periodic (with period $2\pi,\ 1,\ F_s$ or $2\pi F_s$ — depends on which frequency you'll choose) as the signal is discrete.
- The mean power of random process in interval $[\omega_1,\ \omega_2]$ can be computed as:

$$P_{[\omega_1,\ \omega_2]} = \frac{1}{2\pi}\int_{\omega_1}^{\omega_2} G(e^{j\omega})d\omega + \frac{1}{2\pi}\int_{-\omega_2}^{-\omega_1} G(e^{j\omega})d\omega = \frac{1}{\pi}\int_{\omega_1}^{\omega_2} G(e^{j\omega})d\omega.$$

- For the **total** mean power of random signal, we have:

$$P = \frac{1}{2\pi}\int_{-\pi}^{+\pi} G(e^{j\omega})d\omega$$

... but this is a value of inverse DTFT for $k = 0$:

$$R[0] = \frac{1}{2\pi} \int_{-\pi}^{\pi} G(e^{j\omega}) e^{+j\omega 0} d\omega = R[0] = \frac{1}{2\pi} \int_{-\pi}^{+\pi} G(e^{j\omega}) d\omega$$

so that we have obtained:

$$R[0] = P$$

Estimation of power spectral density $G(e^{j\omega})$ using DFT

In case we dispose of one realization of random process $x[n]$ with samples N, we can estimate the PSD using Discrete Fourier transform (DFT … this is the only one that we can actually compute!). Reminder:

$$X[k] = \sum_{n=0}^{N-1} x[n] e^{-j\frac{2\pi}{N}kn}$$

$G(e^{j\omega})$ will be obtained only for discrete frequencies: $\omega_k = \frac{2\pi}{N}k$:

$$\hat{G}(e^{j\omega_k}) = \frac{1}{N}|X[k]|^2.$$

This estimate is usually very unreliable (noisy), so that we often use **Welch method** – averaging of several PSD estimates over several time-segments:

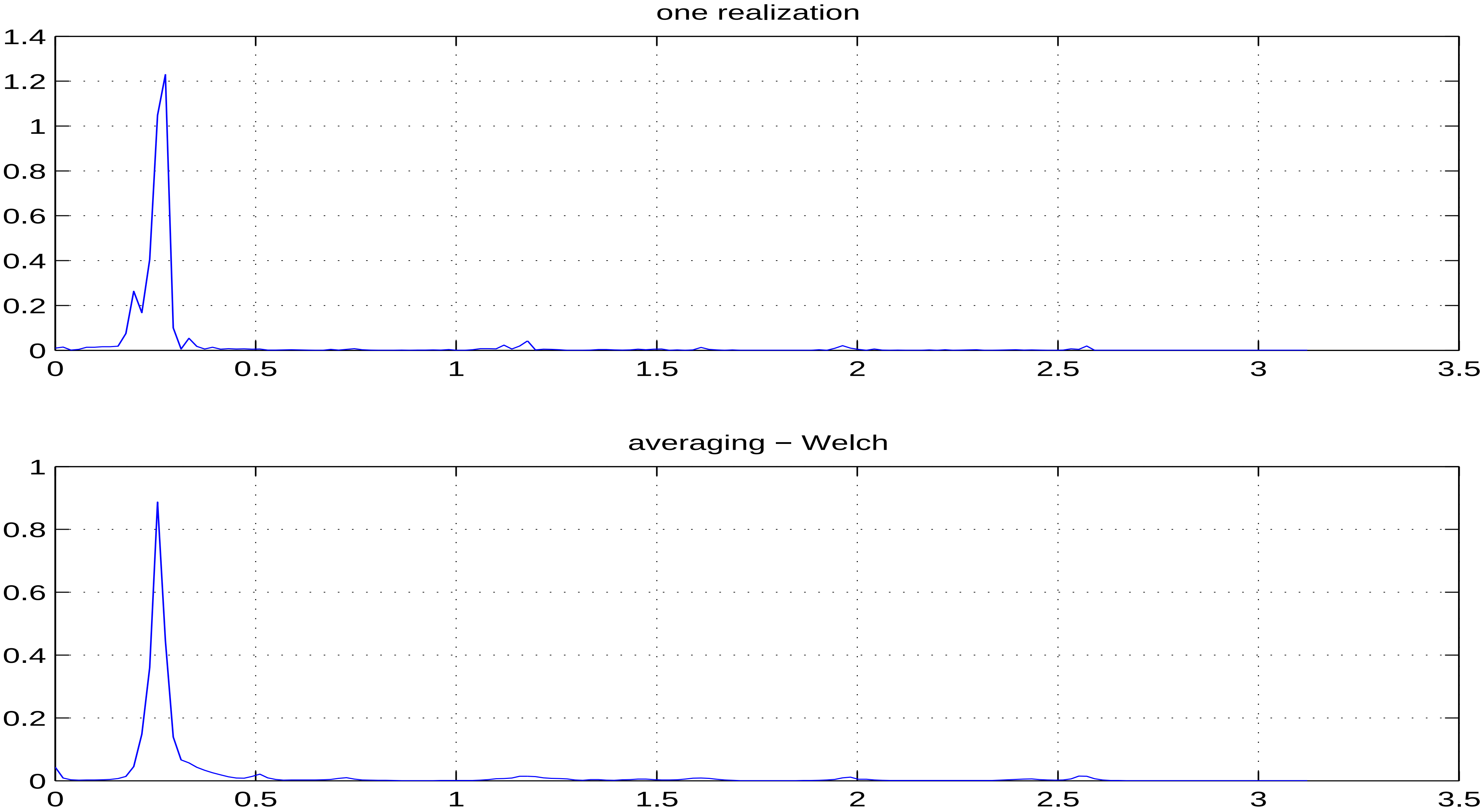
- The signal is divided into M segments, each with N samples and DFT is computed for each segment:

$$X_m[k] = \sum_{n=0}^{N-1} x_m[n] e^{-j\frac{2\pi}{N}kn} \quad \text{for} \quad 0 \leq m \leq M-1$$

- The PSD-estimate is done:

$$\hat{G}_W(e^{j\omega_k}) = \frac{1}{M} \sum_{m=0}^{M-1} \frac{1}{N} |X_m[k]|^2.$$

Example of PSD estimate from one 320-sample segment and average from 1068 such segments:



$\Rightarrow$ the estimate obtained by averaging is much smoother.

Random signals processed by linear systems

- **continuous time:** the linear system has complex frequency response $H(j\omega)$. For input signal with PSD $G_x(j\omega)$ the output PSD is:

$$G_y(j\omega) = |H(j\omega)|^2 G_x(j\omega)$$

- **discrete time:** the linear system has complex frequency response $H(e^{j\omega})$. For input signal with PSD $G_x(e^{j\omega})$, the output PSD is:

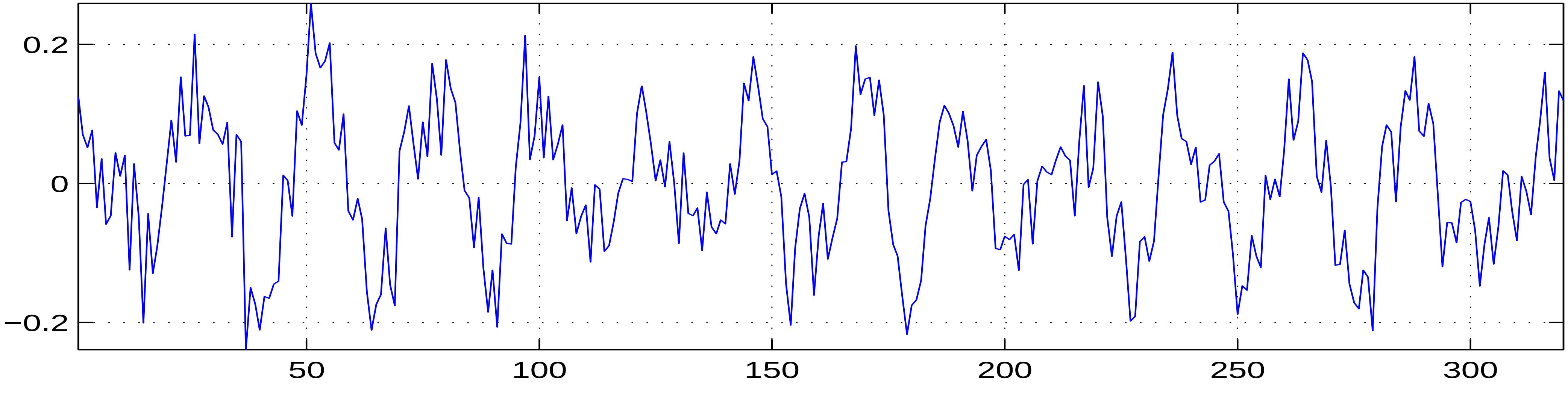
$$G_y(e^{j\omega}) = |H(e^{j\omega})|^2 G_x(e^{j\omega})$$

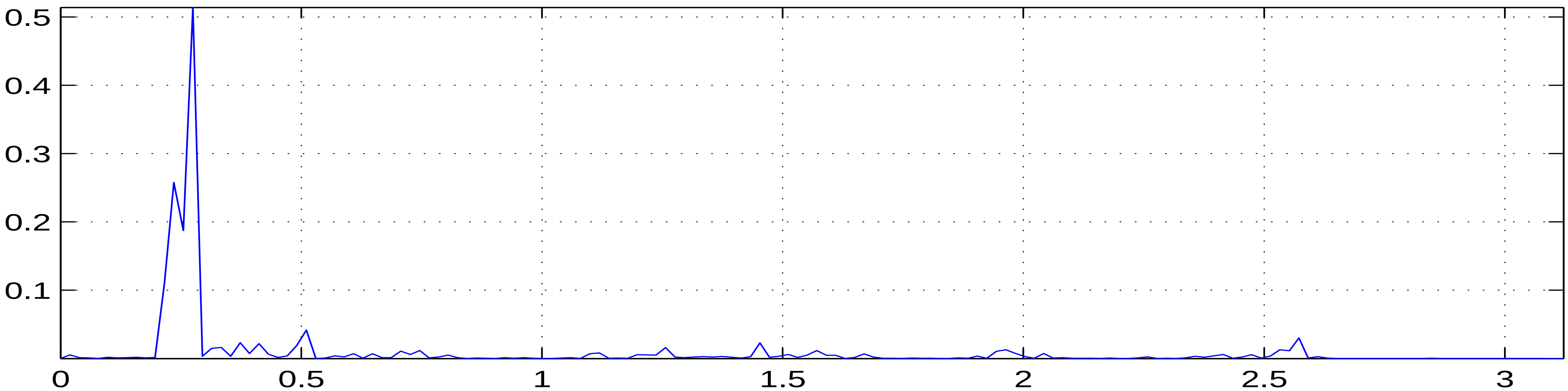
In both cases, the input PSD is multiplied by the square of the **magnitude** of complex frequency response.

The shape of probability density function is **not changed** by the linear system – only the parameters change.

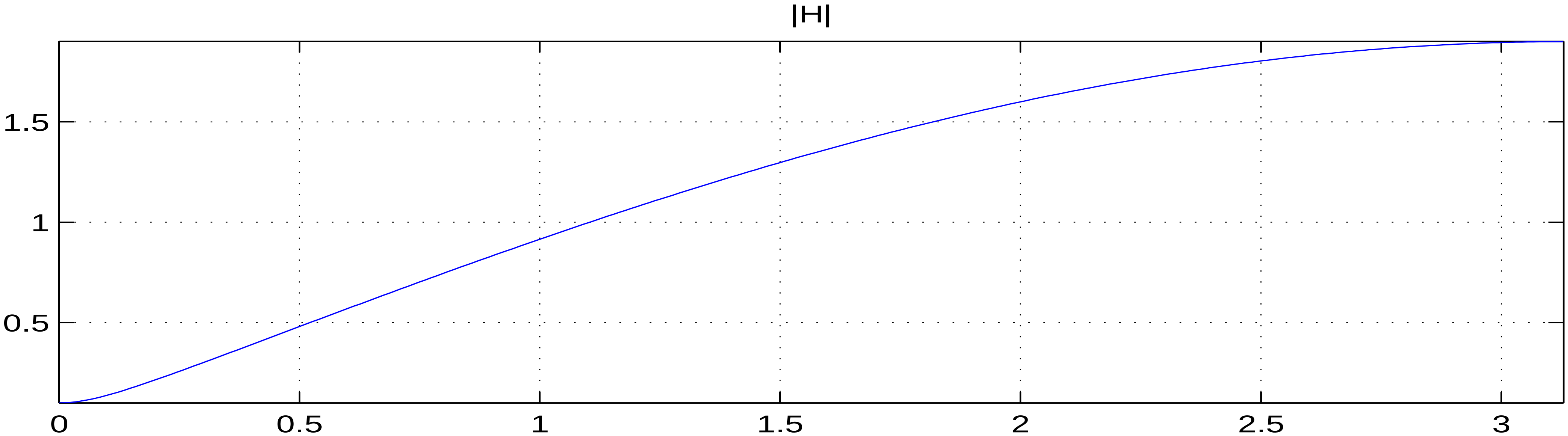
Example: filtering of one realization of the water flowing by filter $H(z) = 1 - 0.9z^{-1}$.

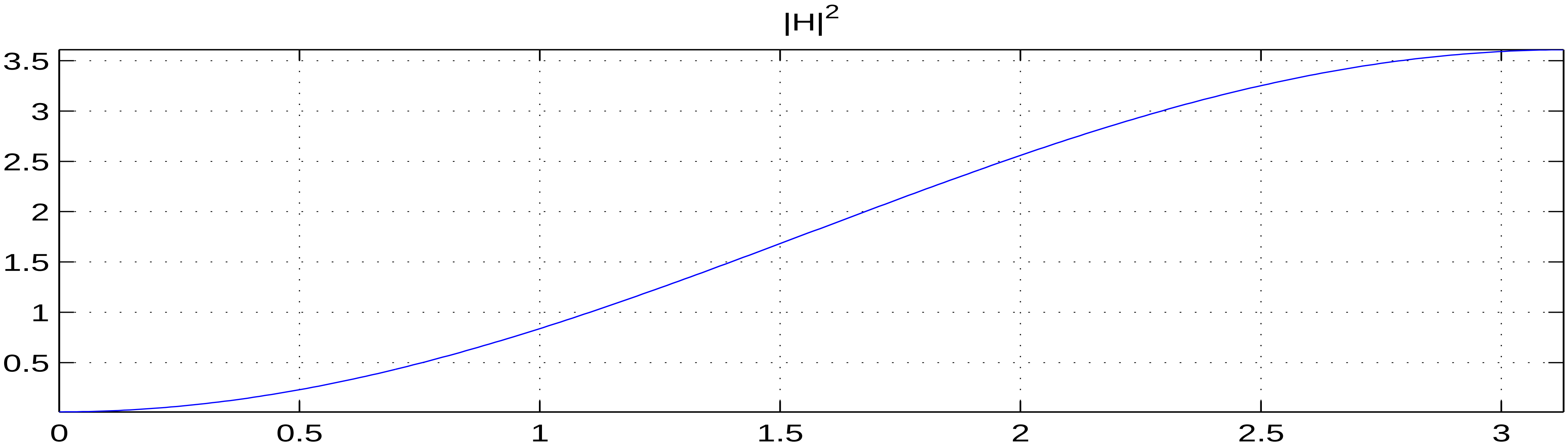
Input signal and its PSD:



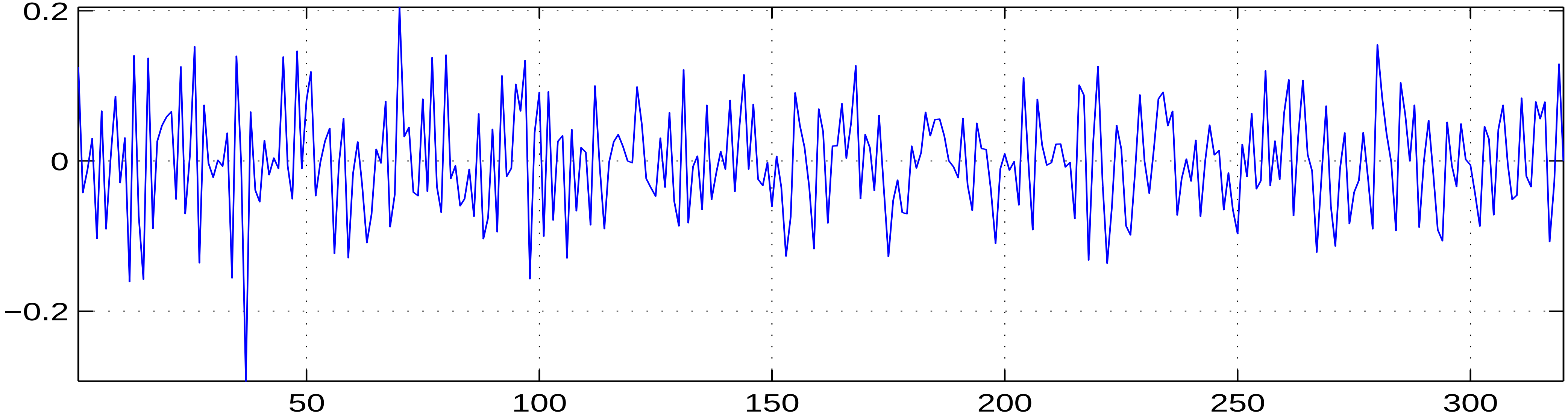


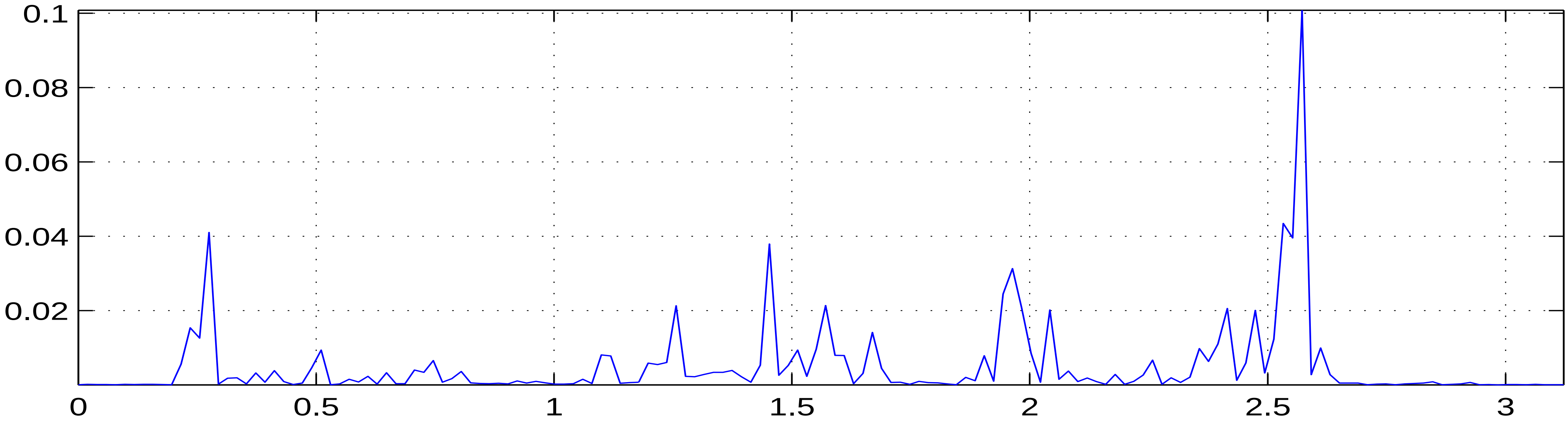
Magnitude of the complex frequency response and its square:





Output signal and its PSD:

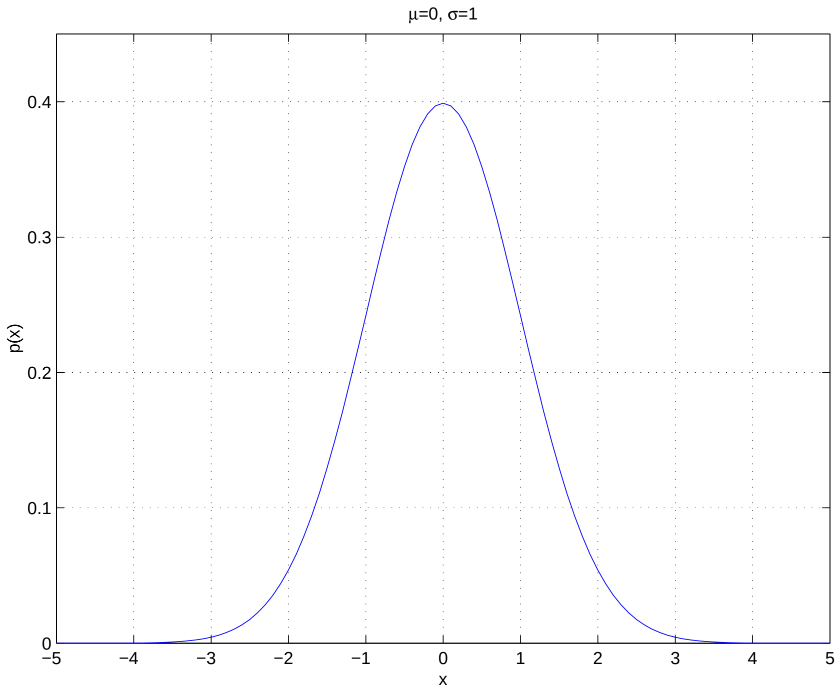


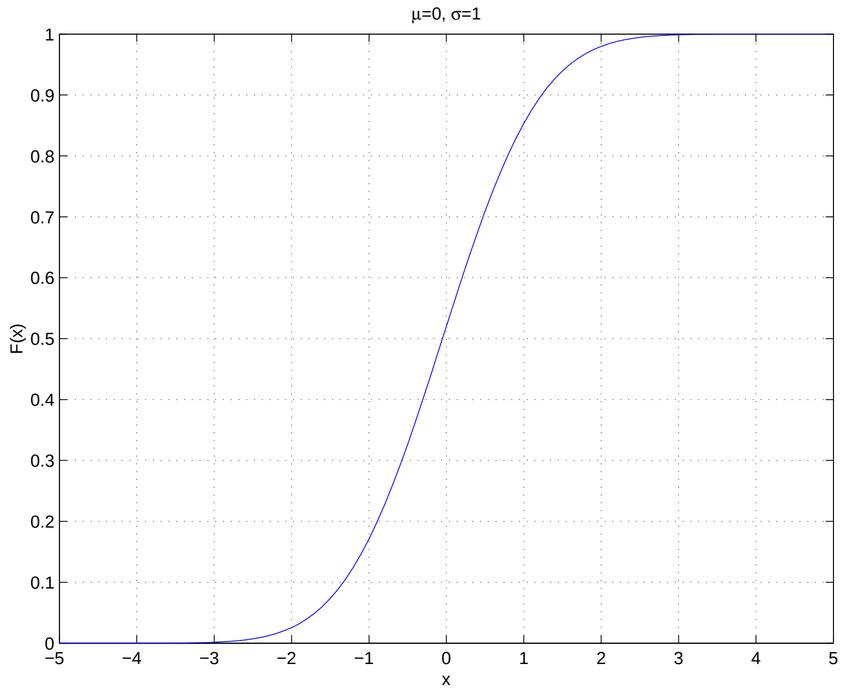


Example of random process – Gaussian white noise

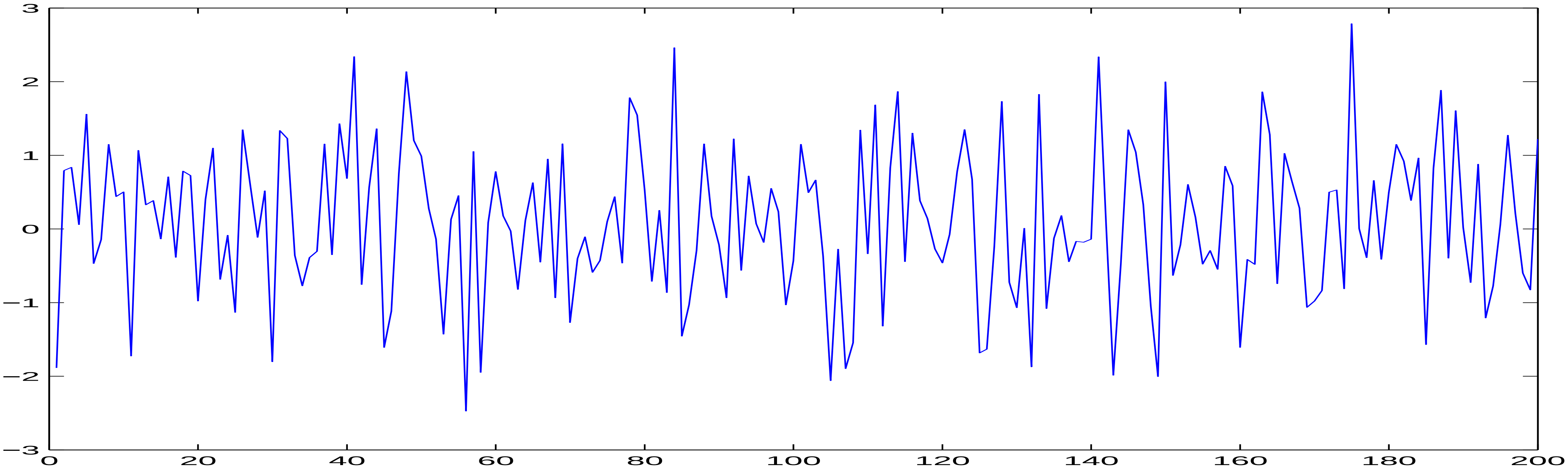
In Matlab, function `randn` – the samples are independent and the PDF is given by Gaussian distribution:

$$p(x) = \mathcal{N}(x;\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{[x-\mu]^2}{2\sigma^2}}$$



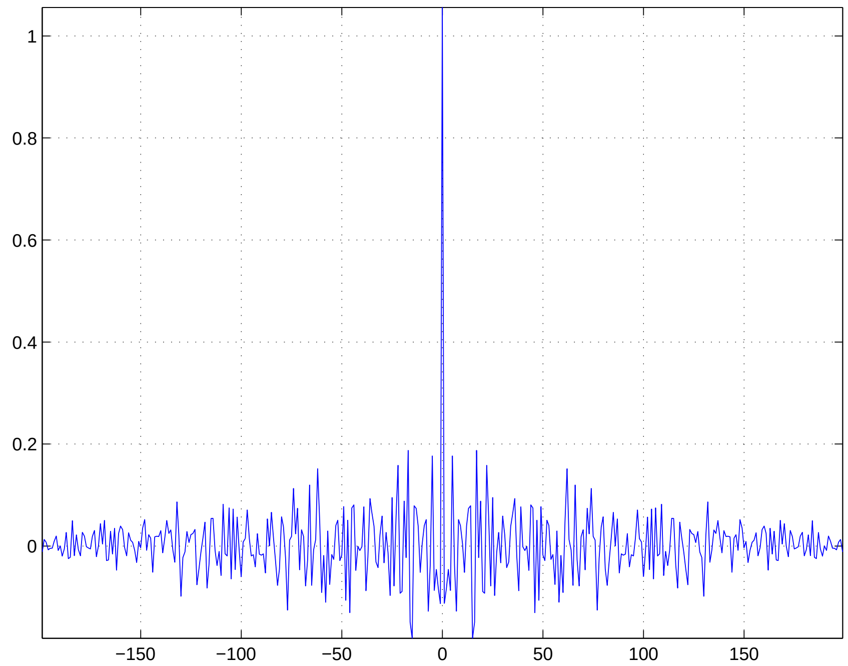


Generation in Matlab: `x = sigma*randn(1,N) + mu`



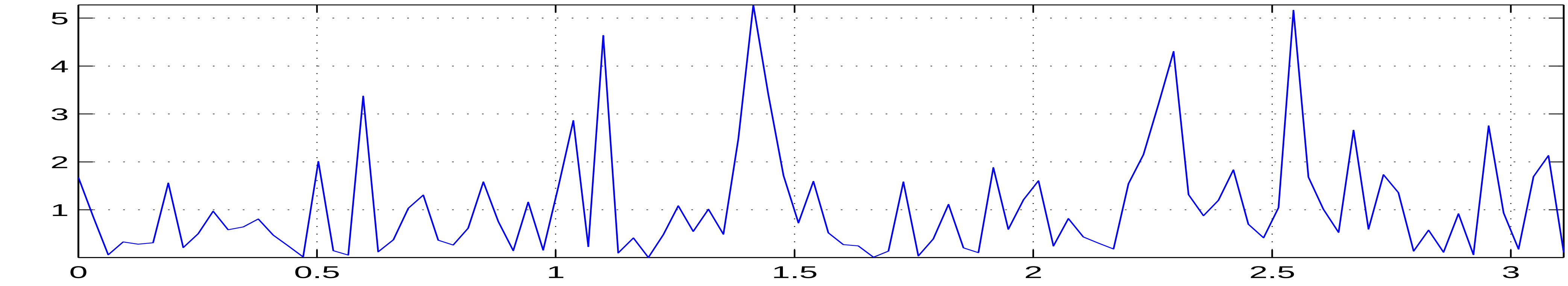
Autocorrelation coefficients:

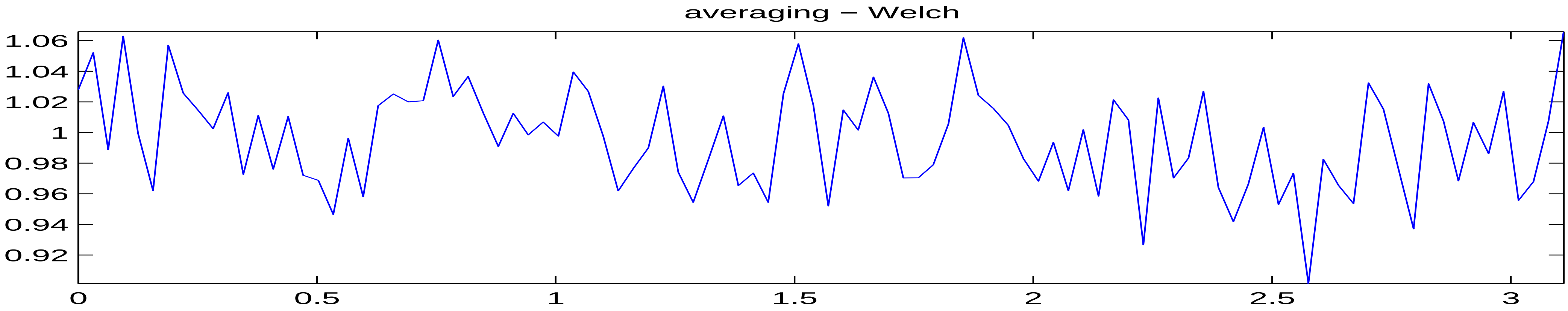
$$R[0] = \mu^2 + D, \qquad R[k] = \mu^2 \text{ pro } k \neq 0$$



PSD for white noise is **constant** (therefore "white"):

$$G(e^{j\omega}) = R[0]$$



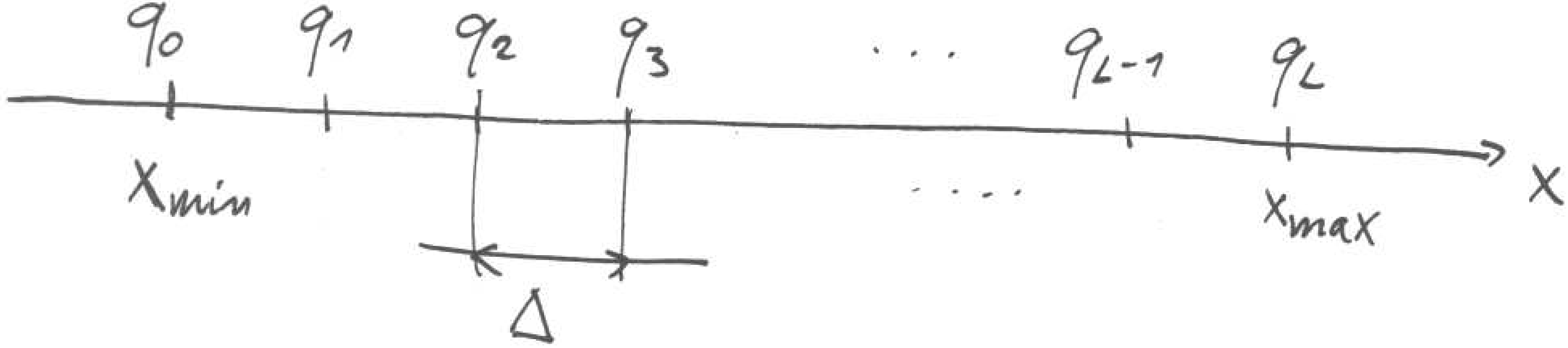


For continuous time, a true white noise ca not be generated (in case $G(j\omega)$ is be non-zero for all ω, the noise would have infinite energy. . .)

Quantization

We can not represent samples of discrete signal $x[n]$ with arbitrary precision $\Rightarrow$ **quantization**. Most often, we will **round** to a fixed number L **quantization levels**, which are numbered 0 to $L-1$. In case we dispose of b bits, $L = 2^b$.

Uniform quantization has uniform distribution of quantization levels $q_0 \dots q_{L-1}$ from minimum value of the signal x_{min} till its maximum value x_{max}:



Quantization step Δ is given by:

$$\Delta = \frac{x_{max} - x_{min}}{L-1},$$

for large L, we can approximate by:

$$\Delta = \frac{x_{max} - x_{min}}{L}.$$

Quantization: for value $x[n]$, the index of the best quantization level is given by:

$$i[n] = \arg \min_{l=0 \dots L-1} |x[n] - q_l|,$$

and the quantized signal is:

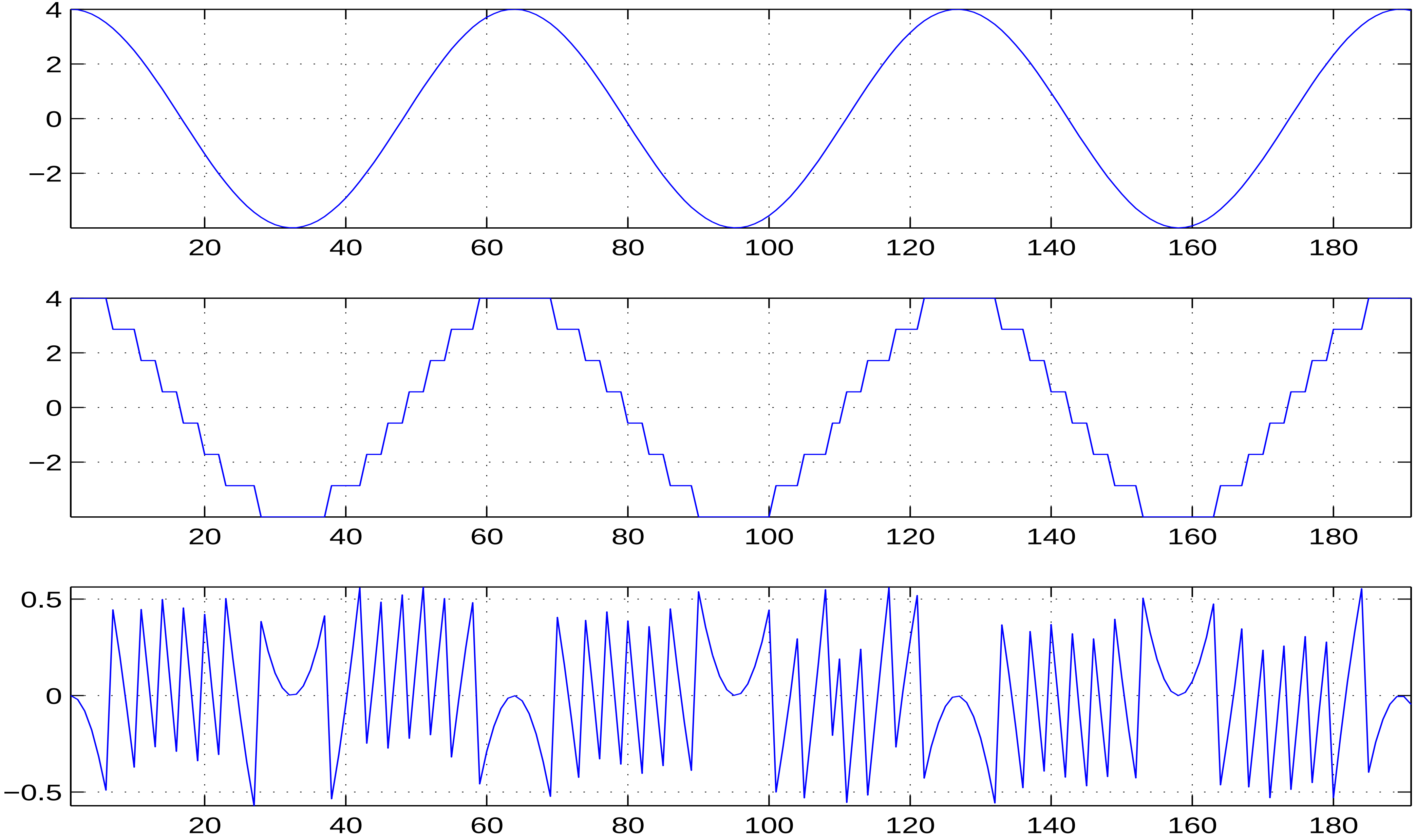
$$x_q[n] = q_{i[n]}.$$

Quantization error:

$$e[n] = x[n] - x_q[n].$$

can also be considered as a **signal** !

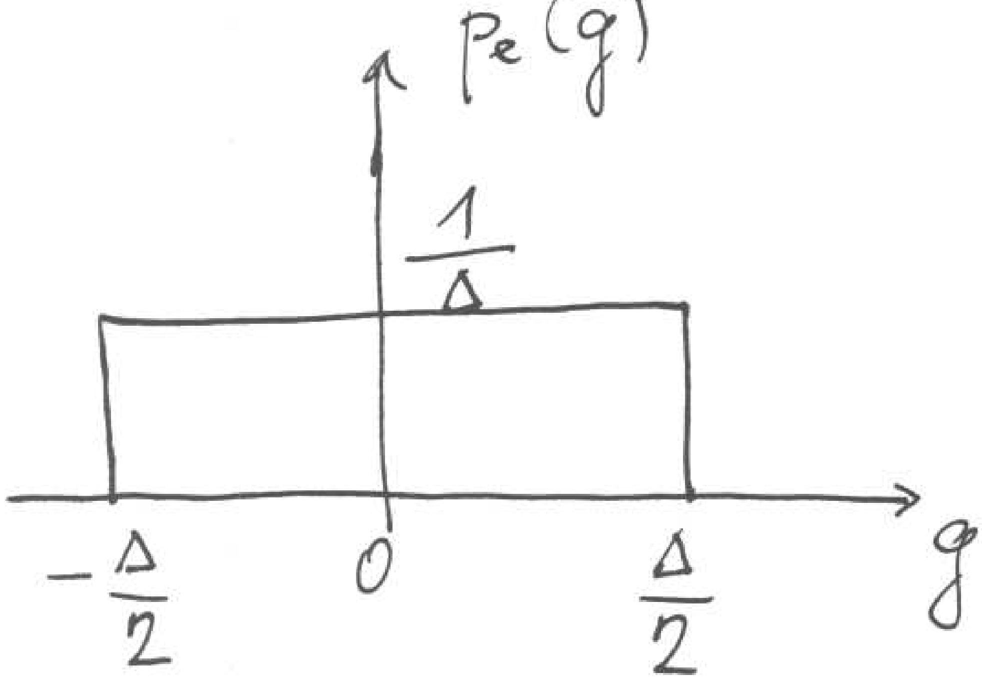
Illustration of quantization on a cosine: $x[n] = 4\cos(0.1n)$, $L = 8$:



To learn, how the signal was distorted by the quantization, we will compute the **power** of the useful signal P_s and compare it to the power of noise P_e: **signal-to-noise ratio (SNR)**:

$$SNR = 10 \log_{10} \frac{P_s}{P_e} \quad \text{[dB]}.$$

For computing the power of error signal, we will make use of the theory of **random processes**: we do not know the values of $e[n]$, but we know that they will be in the interval $[-\frac{\Delta}{2}, +\frac{\Delta}{2}]$ and that they will be uniformly distributed. Probability density function for $e[n]$ will therefore be:



…its height $\frac{1}{\Delta}$ is given by the fact that the surface:

$$\int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} p_e(g)dg \stackrel{!}{=} 1.$$

This process has a zero mean value (we would easily obtain $\int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} gp_e(g)dg = 0$), so that its power will be equal to the **variance**:

$$P_e = D_e = \int_{-\infty}^{\infty} g^2 p_e(g)dg = \int_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} g^2 p_e(g)dg = \frac{1}{\Delta}\left[\frac{g^3}{3}\right]_{-\frac{\Delta}{2}}^{\frac{\Delta}{2}} = \frac{1}{3\Delta}\left(\frac{\Delta^3}{8} + \frac{\Delta^3}{8}\right) = \frac{\Delta^2}{12}$$

SNR computation for a cosine

amplitude A, the cosine has power $P_s = \frac{A^2}{2}$
$x_{min} = -A$, $x_{max} = A$, so that

$$\Delta = \frac{2A}{L} \qquad P_e = \frac{\Delta^2}{12} = \frac{4A^2}{12L^2} = \frac{A^2}{3L^2}.$$

Signal-to-noise ratio:

$$SNR = 10\log_{10}\frac{P_s}{P_e} = 10\log_{10}\frac{\frac{2A}{L}}{\frac{A^2}{3L^2}} = 10\log_{10}\frac{3L^2}{2}.$$

In case we dispose of b bits and the number of quantization levels is $L = 2^b$:

$$SNR = 10\log_{10}\frac{3}{2}(2^b)^2 = 10\log_{10}\frac{3}{2} + 10\log_{10} 2^{2b} = 1.76 + 20b\log_{10} 2 = 1.76 + 6\,b \text{ dB}.$$

The constant 1.76 depends on the character of signal (cos, noise), but it always holds that **adding/removing 1 bit improves/decreases the SNR by 6 dB**.

Example: cosine with $A = 4$ quantized at $L = 8$ levels:

$$SNR_{teor} = 1.76 + 3 \times 6 = 19.76 \text{ dB}$$

$$SNR_{exp} = 10 \log_{10} \frac{\frac{1}{N} \sum_{n=0}^{N-1} s^2[n]}{\frac{1}{N} \sum_{n=0}^{N-1} e^2[n]} = 19.36 \text{ dB}$$

Matlab: `snr = 10*log10 (sum(x.^2) / sum(e.^2))`

. . . quite matching ;-)