Lecture

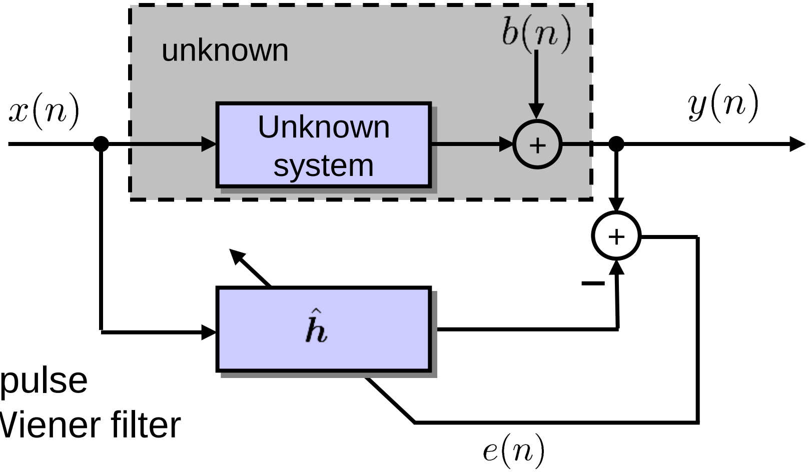
Adaptive Filters

Lecture 2: Wiener Filter

Content

- Introduction and motivation
- Cost functions
- Principle of orthogonality
- Time-domain solution
- Frequency-domain solution
- Application examples:
  - System identification
  - Noise suppression

Setup: Unknown system identification



- Signals and system parameters:

$x(n)$: Excitation signal (accessible)

$b(n)$: Noise / Disturbance (unknown)

$y(n)$: Disturbed system output (accessible)

$\hat{\boldsymbol{h}} = \left[\hat{h}_0, \hat{h}_1, \ldots, \hat{h}_{N-1}\right]^{\mathrm{T}}$: Time-invariant impulse response of the Wiener filter

$\widehat{d}(n) = \sum_{i=0}^{N-1} x(n-i)\,\hat{h}_i$: Output signal of the Wiener filter

$e(n) = y(n) - \sum_{i=0}^{N-1} x(n-i)\,\hat{h}_i$: Error signal

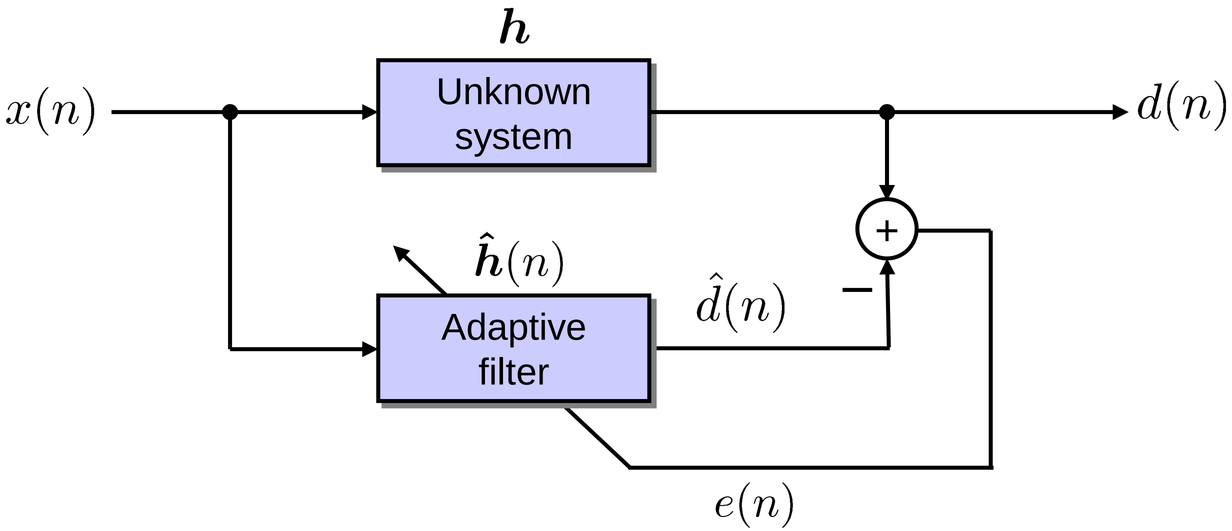
- Assumptions:
  1) All signals can be modeled as real and stationary random processes.
  2) Unknown system is time-invariant (i.e., does not change over time)

- Optimization criterion: Based on the error signal

Cost functions of adaptive filters

TECHNISCHE UNIVERSITÄT DARMSTADT

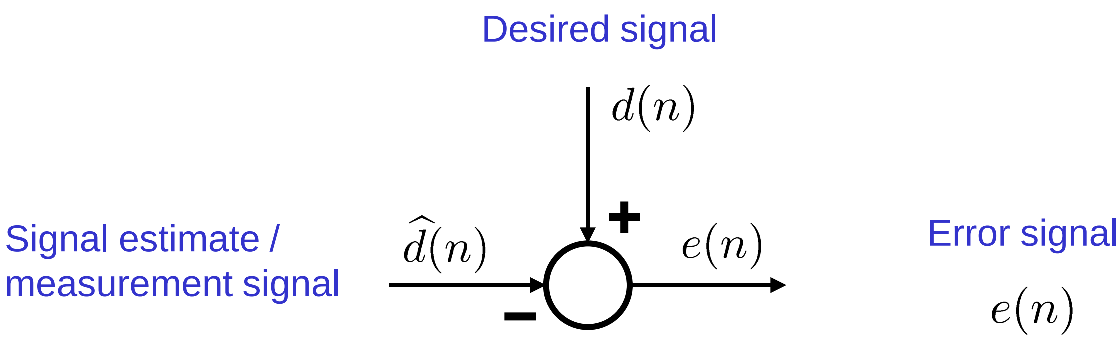
- System identifiction setup (without system noise):



$$
\begin{aligned}
e(n) &= d(n) - \widehat{d}(n) \\
&= \sum_{v=0}^{\infty} h_v\, x(n-v) - \sum_{v=0}^{N-1} \widehat{h}_v(n)\, x(n-v) \\
&\approx \sum_{v=0}^{N-1} \left[h_v - \widehat{h}_v(n)\right] x(n-v)
\end{aligned}
$$

Cost functions of adaptive filters

- Setup of the calculation of an error signal:



$$e(n) \quad = \quad d(n) - \widehat{d}(n)$$

- A cost function is typically a compromise between the requirements and the possibility to allow for a calculation of a solution based on models.

- This depends on which statistical properties are known or can be reliably estimated, e.g., the mean, pdf’s, PSDs, etc.

- The solution should be “robust”, i.e., small deviations from the optimum solution should not provoke strong changes of the optimum solution.

Cost functions of adaptive filters

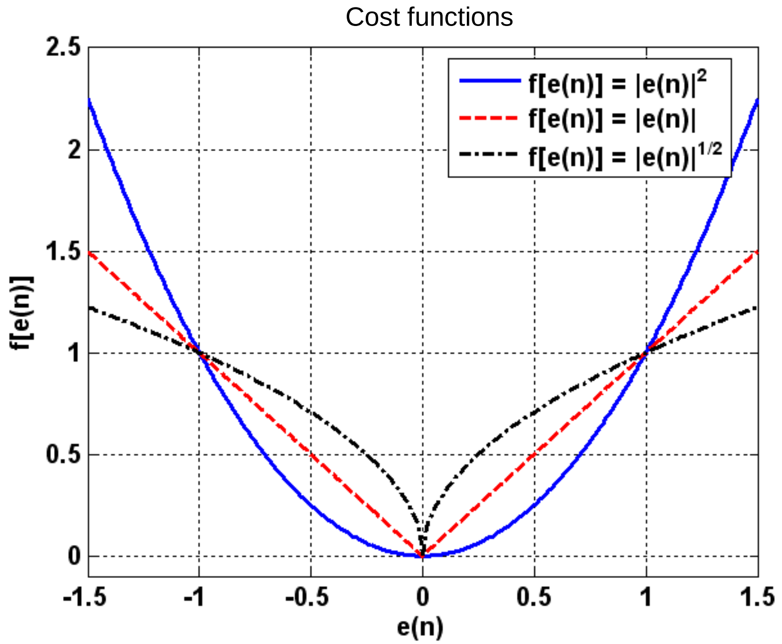
- Desired properties of the error function:
  1. $f[e(n)]$ instead of $f[e(n),\, d[n],\, \widehat{d}(n)]$
  2. $f[e_2(n)] \geq f[e_1(n)]$ for $|e_2(n)| \geq |e_1(n)|$ $\longleftarrow$ necessary
  3. $f[e(n)] = f[-e(n)]$
  4. $f[e(n),\, e(n-1),\, e(n-2),\, \ldots]$

- Explanations:
  1. Only the error signal is used, not the estimated output values
  2. In case the error magnitude increases, also the output of the error function should increase.
  3. For positive or negative error values the error function should give the same result.
  4. The error function may not only depend on the current but also previous error signal samples

Cost functions of adaptive filters

- Examples:
  - $f[e(n)] = |e(n)|$
  - $f[e(n)] = |e(n)|^2$
  - in general: $f[e(n)] = |e(n)|^\alpha$

    $\alpha > 1$ amplifies large errors
    $\alpha < 1$ attenuates large errors



- Deterministic error criterion with memory:

$$f[e(n), e(n-1), \ldots] = \sum_{k=0}^{\infty} \lambda^k \, |e(n-k)|^2 \quad \text{with} \quad 0 < \lambda \leq 1$$

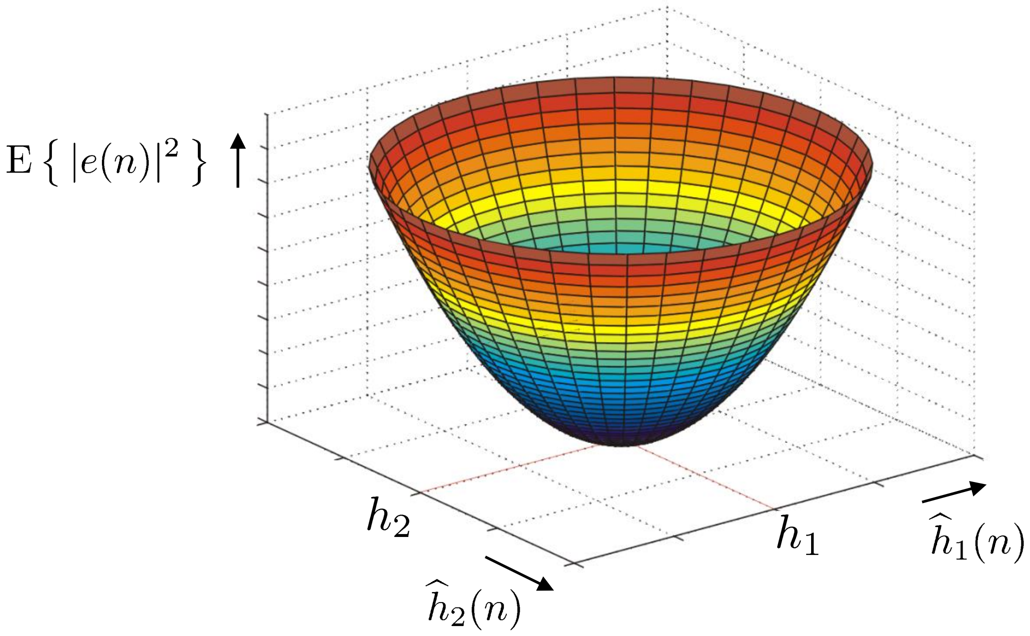
- Error criterion by ensemble mean:

$$f[e(n)] = \mathrm{E}\left\{ \, |e(n)|^2 \, \right\}$$ Mean square error

with $e(n)$: random process

Cost functions of adaptive filters

- Typically preferred cost function: quadratic cost function $\Longrightarrow$ Mean square error



Mean square error:
Allowing to find the global minimum.

Graph shows the error surface
for a two-filter-tap system.

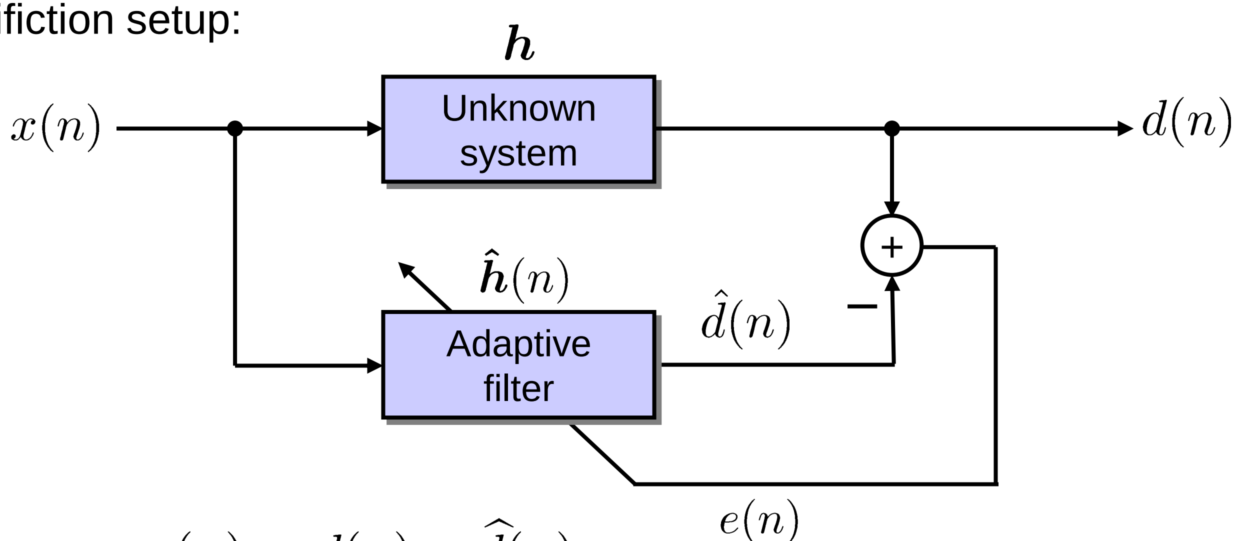
Konvex Problem!
=> unique solution

- With the error signal:

$$e(n) = d(n) - \widehat{d}(n) = \sum_{v=0}^{N-1} \left[h_v - \widehat{h}_v(n)\right] x(n-v)$$

Cost functions of adaptive filters

- System identifiction setup:



Signal error: $e(n) = d(n) - \widehat{d}(n)$

perceba que isso é um escalar

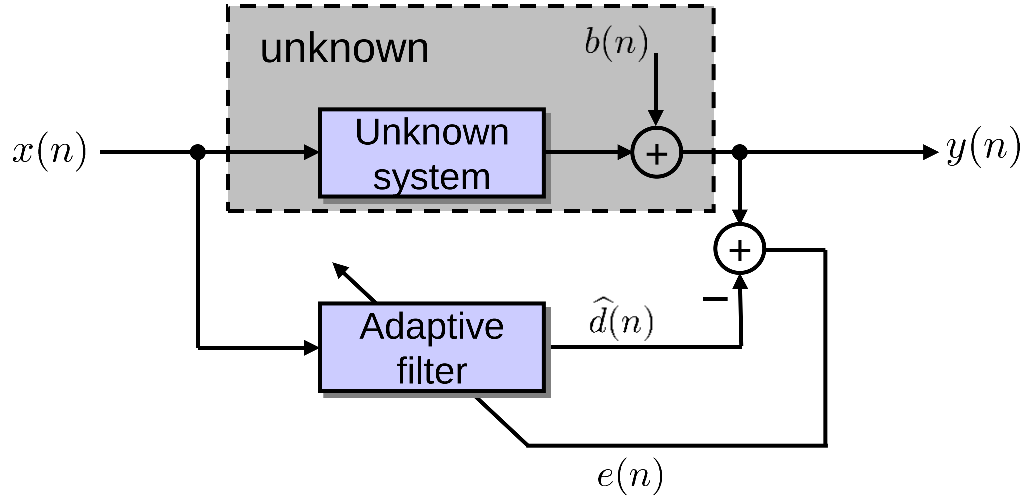
"System error" or
"System distance": $\|\boldsymbol{h}_\Delta(n)\| = \|\boldsymbol{h} - \widehat{\boldsymbol{h}}\boldsymbol{(n)}\|$

An optimum system distance (min. system error) can only be obtained when a "persistant" excitation is present
(=> all frequency components are excited)

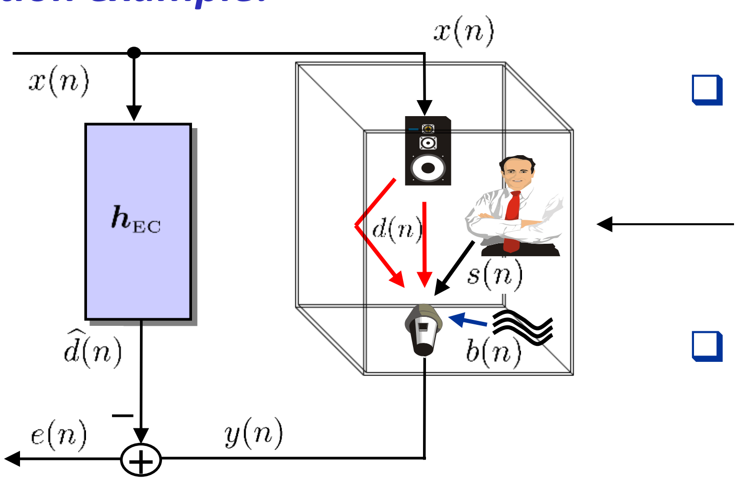
ex: se o filtro adaptativo for exatamente igual em apenas uma frequencia e x estiver naquela frequencia, nunca perceberíamos

System identification

Generalized setup:



Application example:



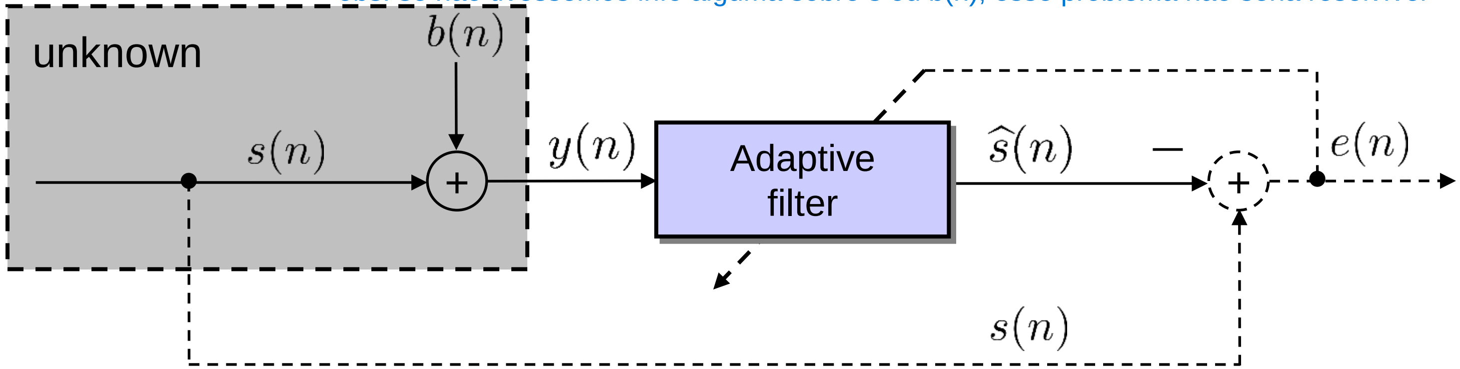
- Echo cancellation
  BUT: signals are non-stationary and system is time-varying
  => no direct Wiener filter application

- Room impulse measurements
  Setup fixed, and measurement signals can be chosen stationary
  => appropriate for direct application

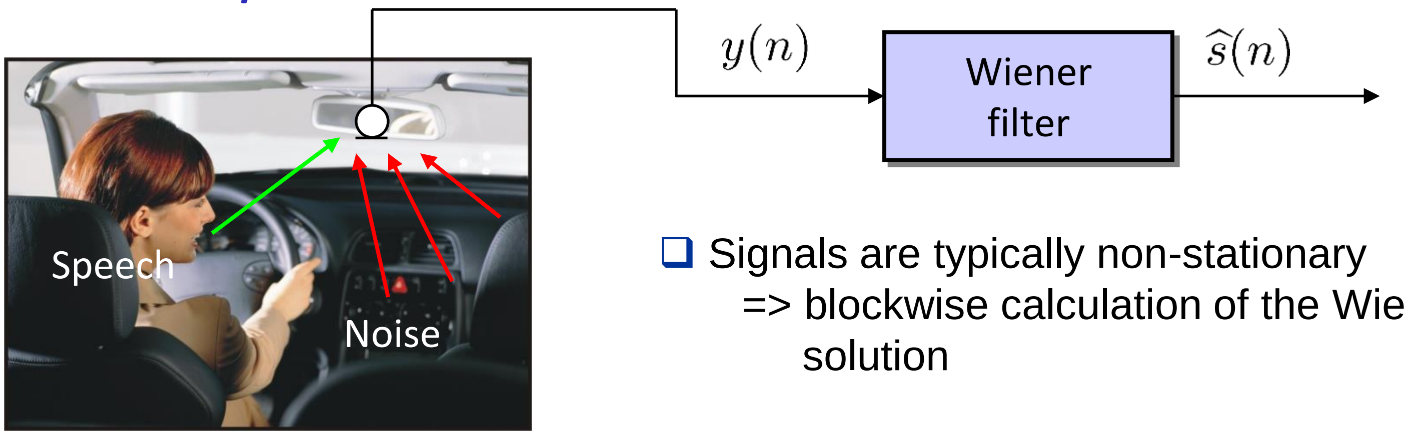
Noise reduction without reference

Generalized setup:

obs: se não tivessemos info alguma sobre s ou b(n), esse problema não seria resolvível



Application example:



- Signals are typically non-stationary
  => blockwise calculation of the Wiener solution

History & Assumptions

Design of filters by means of minimizing the squared error (according to Gauß)

Independent development

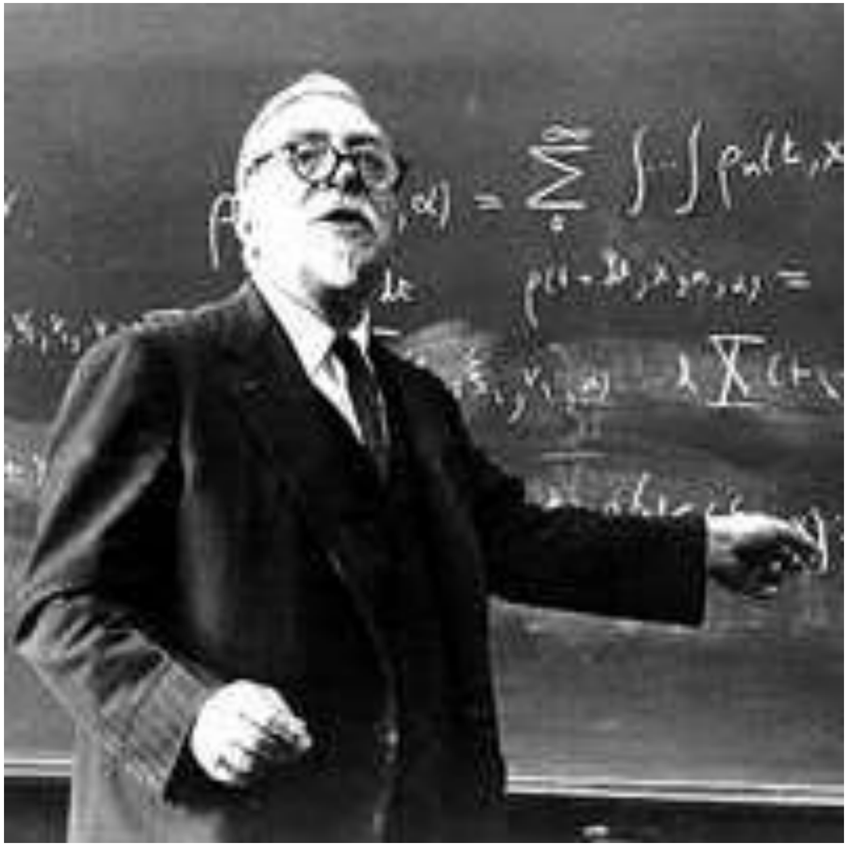
1941: A. Kolmogoroff: *Interpolation und Extrapolation von stationären zufälligen Folgen*, Izv. Akad. Nauk SSSR Ser. Mat. 5, pp. 3 – 14, 1941 (in Russian)

1942: N. Wiener: *The Extrapolation, Interpolation, and Smoothing of Stationary Time Series with Engineering Applications*, J. Wiley, New York, USA, 1949 (originally published in 1942 as MIT Radiation Laboratory Report)

Assumptions / design criteria:

- Design of a filter that separates a desired signal optimally from additive noise
- Both signals are described as stationary random processes
- Knowledge about the statistical properties up to second order necessary

Norbert Wiener



Norbert Wiener (November 26, 1894, Columbia, Missouri – March 18, 1964, Stockholm, Sweden) was an American mathematician.

A famous child prodigy, Wiener later became an early researcher in stochastic and noise processes, contributing work relevant to electronic engineering, electronic communication, and control systems.

Wiener is regarded as the originator of cybernetics, a formalization of the notion of feedback, with many implications for engineering, systems control, computer science, biology, philosophy, and the organization of society.

from Wikipedia

Andrey Nikolaevich Kolmogorov

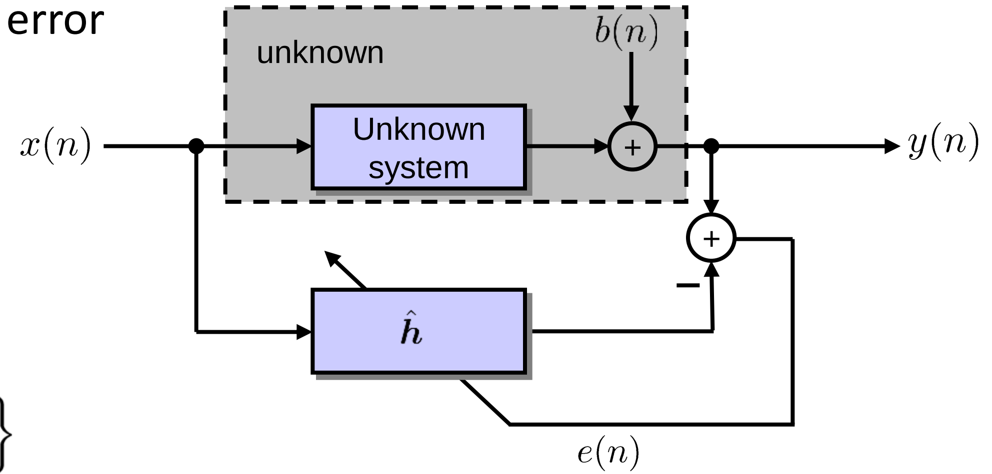
Andrey Nikolaevich Kolmogorov
(25 April 1903 – 20 October 1987)
was a Soviet Russian mathematician,
preeminent in the 20th century,
who advanced various scientific
fields, among them probability
theory, topology, intuitionistic
logic, turbulence, classical
mechanics and computational
complexity.

from Wikipedia

Unknown system identification: Principle of orthogonality

- Optimization criterion: mean square error
  Derivation for real-value signals:

$$\mathrm{E}\{e^2(n)\} \underset{\hat{\boldsymbol{h}}=\hat{\boldsymbol{h}}_{\mathrm{opt}}}{\longrightarrow} \min$$



- Derivation of the cost function according to the coefficients $\hat{h}_j$:

$$\begin{aligned}\frac{\partial \mathrm{E}\left\{e^2(n)\right\}}{\partial \hat{h}_j} &= \mathrm{E}\left\{2\,e(n)\,\frac{\partial e(n)}{\partial \hat{h}_j}\right\} \\ &= \mathrm{E}\left\{2\,e(n)\,\frac{\partial\left(y(n) - \sum\limits_{i=0}^{N-1} x(n-i)\,\hat{h}_i\right)}{\partial \hat{h}_j}\right\} \\ &= -2\,\mathrm{E}\left\{e(n)\,x(n-j)\right\} \quad \text{for} \quad j \in \{0,\,\ldots,\,N-1\}\end{aligned}$$

dy/dh = 0

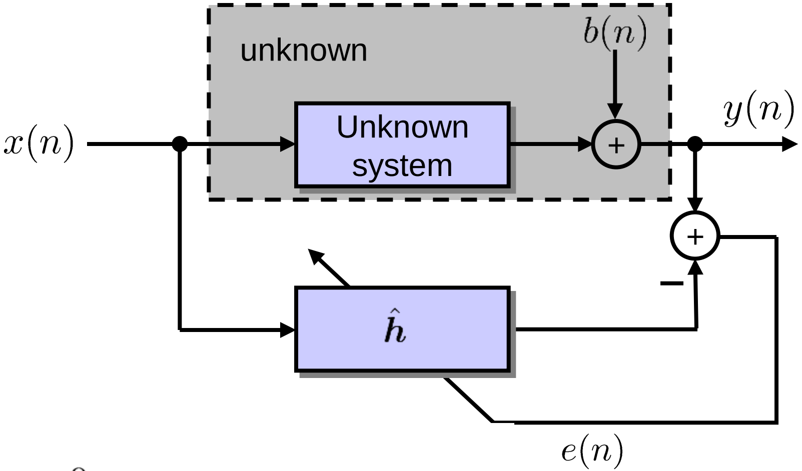
dhi/dhj =0 (pq)

- Principle of orthogonality:

tente explicar com suas proprias palavras o que significa ser "ortogonal"

$$\boxed{\mathrm{E}\left\{e_{\mathrm{opt}}(n)\,x(n-j)\right\} = 0 \quad \text{for} \quad j \in \{0,\,\ldots,\,N-1\}}$$

Optimal Wiener filter solution



- Principle of orthogonality:

$$\boxed{\mathrm{E}\Big\{e_{\mathrm{opt}}(n)\,x(n-j)\Big\} = 0 \quad \text{for} \quad j \in \{0,\,...,\,N-1\}}$$

- Insertion of the error signal:

$$\mathrm{E}\left\{x(n-j)\left[y(n) - \sum_{i=0}^{N-1} \hat{h}_{\mathrm{opt},i}\,x(n-i)\right]\right\} \;=\; 0$$

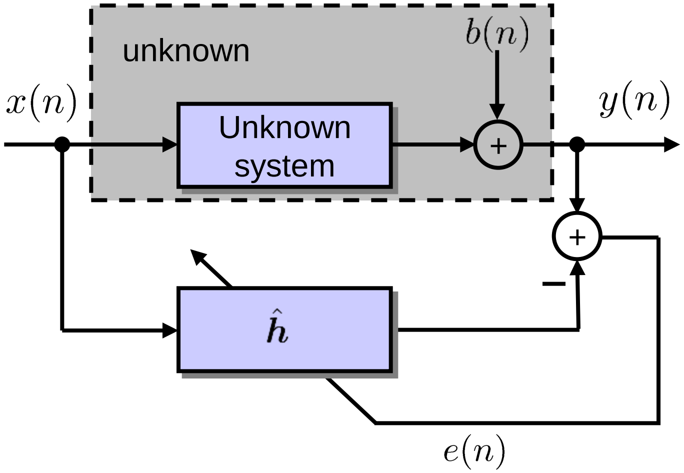
$$\underbrace{\mathrm{E}\Big\{x(n-j)\,y(n)\Big\}}_{r_{xy}(j)} - \sum_{i=0}^{N-1} \hat{h}_{\mathrm{opt},i}\;\underbrace{\mathrm{E}\Big\{x(n-j)\,x(n-i)\Big\}}_{r_{xx}(j-i)} \;=\; 0$$

$$r_{xy}(j) - \sum_{i=0}^{N-1} \hat{h}_{\mathrm{opt},i}\,r_{xx}(j-i) \;=\; 0 \quad \text{for} \quad j \in \{0,\,...,\,N-1\}$$

Optimal Wiener filter solution

- Current solution:

$$r_{xy}(j) - \sum_{i=0}^{N-1} \hat{h}_{\mathrm{opt},i}\, r_{xx}(j-i) \;=\; 0 \quad \text{for} \quad j \in \{0,\, \dots,\, N-1\}$$



- All equations:

$$\begin{array}{cccccc}
r_{xx}(0)\,\hat{h}_{\mathrm{opt},0} & + \;\dots & + \; r_{xx}(N-1)\,\hat{h}_{\mathrm{opt},N-1} & = & r_{xy}(0) \\
r_{xx}(1)\,\hat{h}_{\mathrm{opt},0} & + \;\dots & + \; r_{xx}(N-2)\,\hat{h}_{\mathrm{opt},N-1} & = & r_{xy}(1) \\
\vdots & + \;\ddots\; + & \vdots & = & \vdots \\
r_{xx}(N-1)\,\hat{h}_{\mathrm{opt},0} & + \;\dots & + \quad r_{xx}(0)\,\hat{h}_{\mathrm{opt},N-1} & = & r_{xy}(N-1)
\end{array}$$

- Matrix vector notation:

$$\boldsymbol{R}_{xx}\,\hat{\boldsymbol{h}}_{\mathrm{opt}} = \boldsymbol{r}_{xy}(0)$$

- Wiener solution:

Rxx tem uma propriedade que vimos na aula passada. Qual?

$$\boxed{\hat{\boldsymbol{h}}_{\mathrm{opt}} = \boldsymbol{R}_{xx}^{-1}\,\boldsymbol{r}_{xy}(0)}$$

- Abbreviations:

$$\hat{\boldsymbol{h}}_{\mathrm{opt}} = \left[\hat{h}_{\mathrm{opt},0},\, \hat{h}_{\mathrm{opt},1},\, \dots,\, \hat{h}_{\mathrm{opt},N-1}\right]^{\mathrm{T}}$$

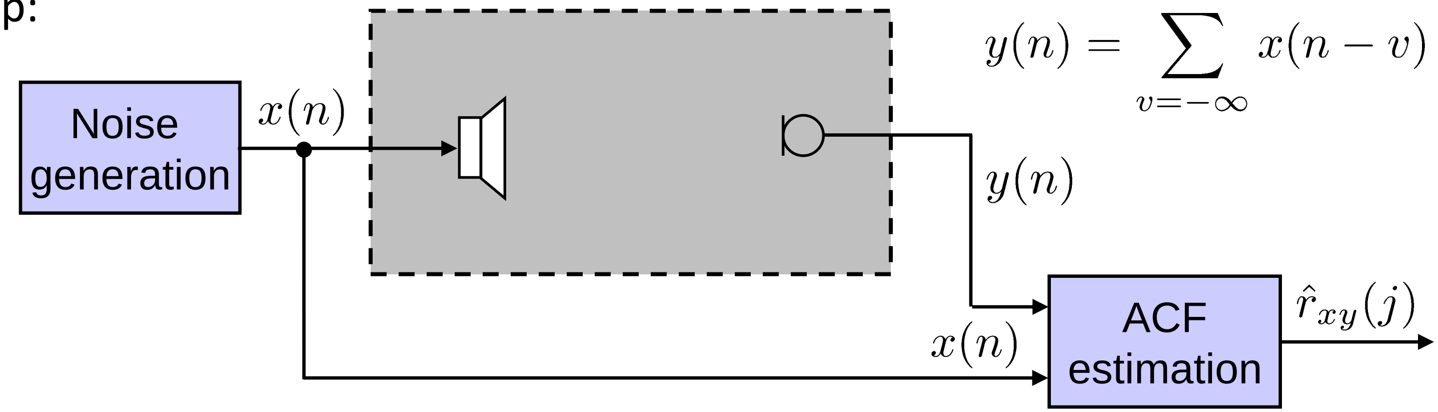
$$\boldsymbol{R}_{xx} = \begin{bmatrix} r_{xx}(0) & r_{xx}(1) & \dots & r_{xx}(N-1) \\ r_{xx}(1) & r_{xx}(0) & \dots & r_{xx}(N-2) \\ \vdots & \vdots & \ddots & \vdots \\ r_{xx}(N-1) & r_{xx}(N-2) & \dots & r_{xx}(0) \end{bmatrix}$$

$$\boldsymbol{r}_{xy}(k) = \left[r_{xy}(k),\, r_{xy}(k+1),\, \dots,\, r_{xy}(k+N-1)\right]^{\mathrm{T}}$$

System identification with the Wiener filter

escolher sinal branco facilita muito a sua vida, pq fica mt facil inverter a matriz de correlação

- **Setup:**

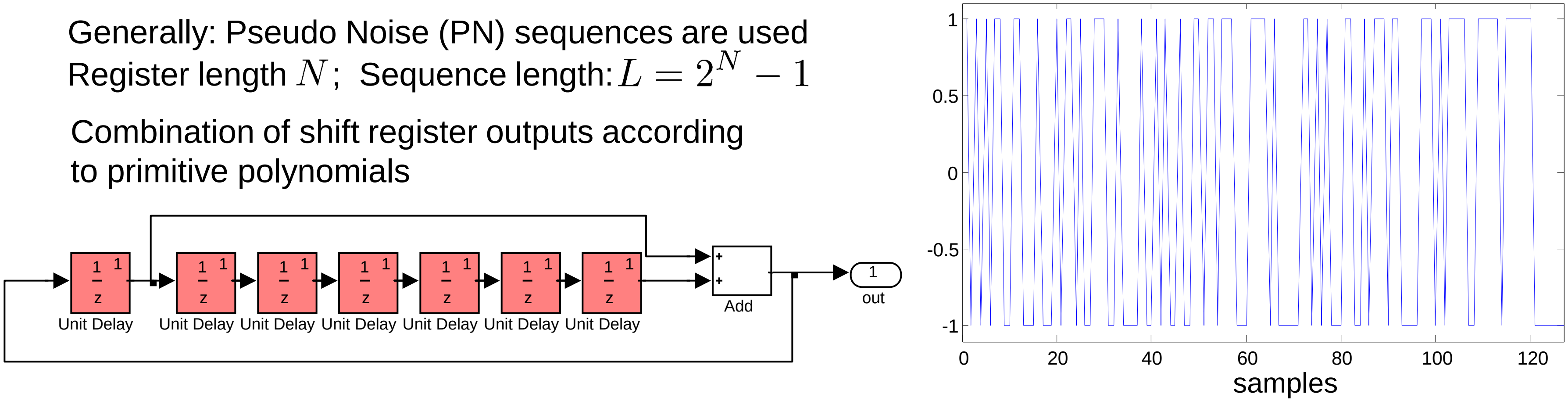


$$y(n) = \sum_{v=-\infty}^{\infty} x(n-v)\, h_v$$

- **Noise generation:**

Generally: Pseudo Noise (PN) sequences are used
Register length N; Sequence length: $L = 2^N - 1$

Combination of shift register outputs according to primitive polynomials

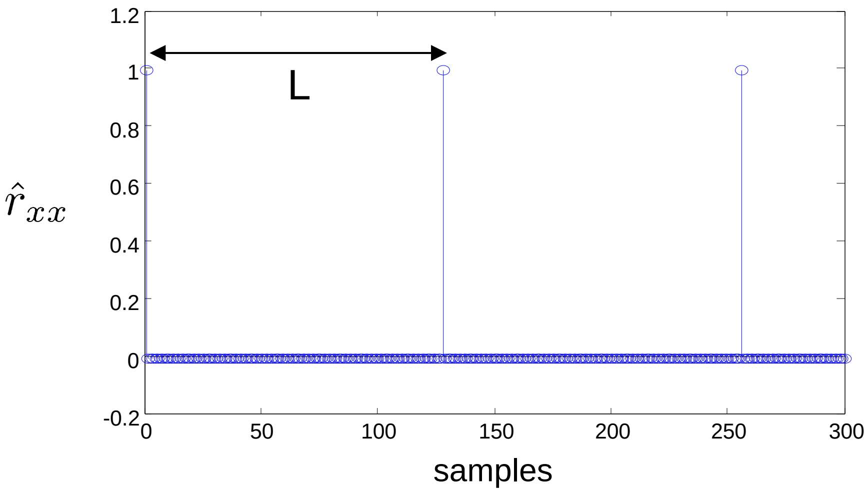


System identification with the Wiener filter

- Properties of PN Sequences:

$$\hat{r}_{xx}(j) = \frac{1}{L+1}\sum_{n=0}^{L-1} x(n)\,x(n+j) = \delta_{K,L}(j) - \frac{1}{L+1}$$

$$= \begin{cases} 1 - \frac{1}{L+1} & j = 0,\ L,\ 2L,\ \ldots \\ -\frac{1}{L+1} & \text{else} \end{cases}$$



no matlab é mais ou menos isso o que acontece tbm com randn(gaussiana) - a autocorrelação não vai ser exatamente 1 ou 0, vc precisaria de um L mto grande pra isso

System identification with the Wiener filter

TECHNISCHE UNIVERSITÄT DARMSTADT

- Properties of PN Sequences:

$$\hat{r}_{xx}(j) = \frac{1}{L+1}\sum_{n=0}^{L-1} x(n)\,x(n+j) = \delta_{K,L}(j) - \frac{1}{L+1}$$

True impulse response of the unknown system

$$y(n) = \sum_{v=-\infty}^{\infty} x(n-v)\,h_v$$

- Impulse response estimation with PN Sequences => periodic repetition:

$$\begin{aligned}
\hat{r}_{xy}(j) &= \frac{1}{L+1}\sum_{n=0}^{L-1} x(n)\,y(n+j) = \frac{1}{L+1}\sum_{n=0}^{L-1} x(n)\sum_{v=-\infty}^{\infty} x(n+j-v)\,h_v \\
&= \sum_{v=-\infty}^{\infty}\frac{1}{L+1}\sum_{n=0}^{L-1} x(n)x(n+j-v)\,h_v = \sum_{v=-\infty}^{\infty}\hat{r}_{xx}(j-v)\,h_v \\
&= \sum_{v=-\infty}^{\infty}\left[\delta_{K,L}(j-v) - \frac{1}{L+1}\right] h_v = \sum_{v=-\infty}^{\infty} h_{j-vL} - \frac{1}{L+1}\sum_{v=-\infty}^{\infty} h_v \\
&\approx \sum_{v=-\infty}^{\infty} h_{j-vL}
\end{aligned}$$

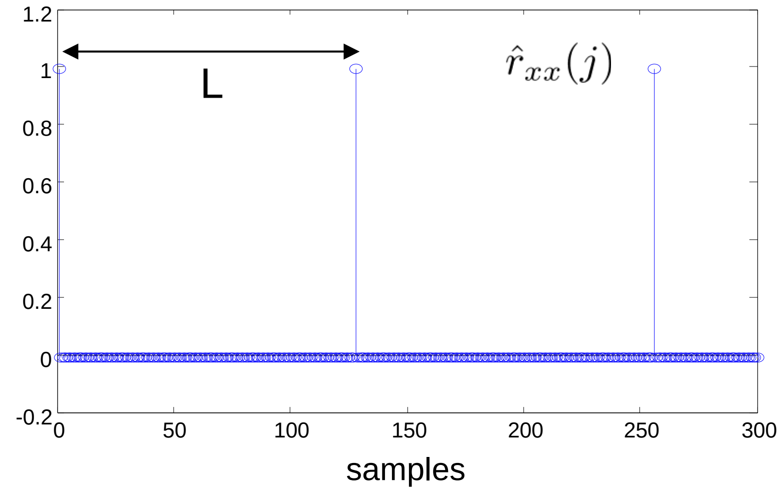
Periodic repetition of impulse response

Mean of impulse response h_v
Typically, close to zero

System identification with the Wiener filter

- Autocorrelation of PN Sequences:

$$\hat{r}_{xx}(j) = \delta_{K,L}(j) - \frac{1}{L+1}$$

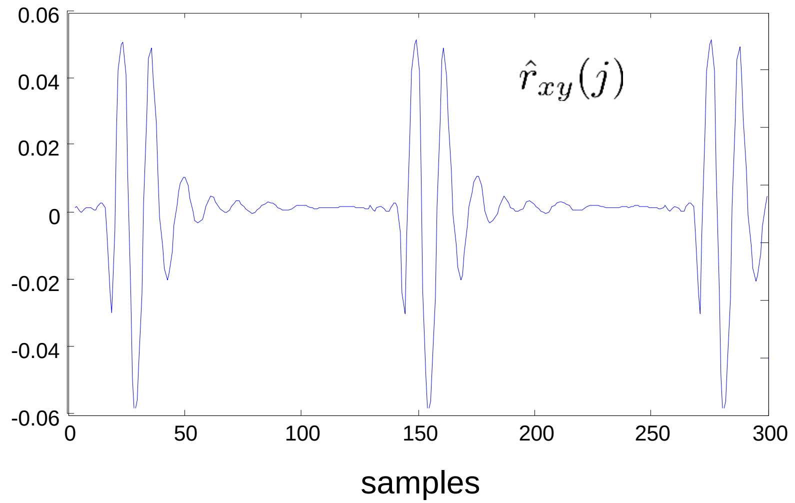


- Cross-correlation of input and output signal in case of PN excitation :

$$\hat{r}_{xy}(j) \approx \sum_{v=-\infty}^{\infty} h_{j-vL}$$

Convolution of the periodic diracs
=> periodic repetition of impulse response

Caution! Cyclic convolution errors if impulse response is longer than L



Sensitivity to noise

TECHNISCHE UNIVERSITÄT DARMSTADT

- With additive noise:

$$y(n) = \sum_{v=-\infty}^{\infty} x(n-v)\, h_v + b(n) \qquad \hat{r}_{xy}(j) = \frac{1}{L+1} \sum_{n=0}^{L-1} x(n)\, y(n+j)$$

- Impulse response estimation with PN Sequences => periodic repetition:

$$\hat{r}_{xy}(j) = \sum_{v=-\infty}^{\infty} h_{j-vL} - \frac{1}{L+1} \sum_{v=-\infty}^{\infty} h_v + \underbrace{\frac{1}{L+1} \sum_{n=0}^{L-1} x(n)\, b(n+j)}_{\hat{r}_{noise}(j)}$$

$$\sigma^2_{\hat{r}_{noise}} = \frac{1}{(L+1)^2} \sum_{n=0}^{L-1} \sum_{u=0}^{L-1} \mathrm{E}\{x(n)\, x(u)\, b(n+j)\, b(u+j)\}$$

$$= \frac{1}{(L+1)^2} \sum_{n=0}^{L-1} \mathrm{E}\{b(n+j)^2\} = \frac{L}{(L+1)^2} \sigma_b^2$$

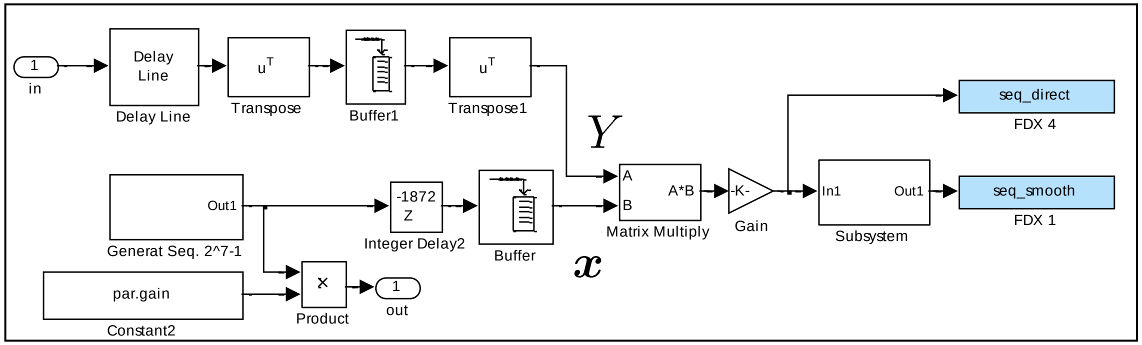
with:

$$\mathrm{E}\{x(n)\, x(u)\} \approx \delta_K(u-n)$$

$$\boxed{\sigma^2_{\hat{r}_{noise}} \approx \frac{1}{(L+1)} \sigma_b^2}$$

=> Reduction of noise power proportional to seq. length

Path measurements: real-time demo



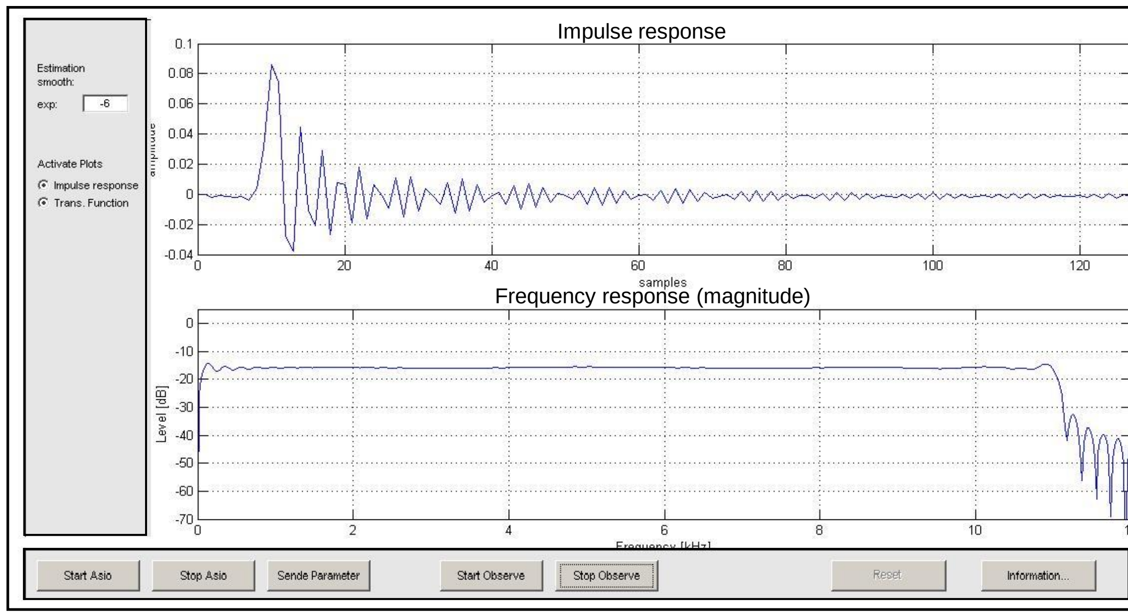
$$\boldsymbol{Y} = \begin{bmatrix} y(n-2N+2) & \ldots & y(n-N+1) \\ y(n-2N+3) & \ldots & y(n-N+2) \\ y(n-N+1) & \ldots & y(n) \end{bmatrix}$$

$$\boldsymbol{x} = [x(n-2N+2)\ \ldots,\ x(n-N+1)]^T$$

$$\hat{r}_{xy}(j) = \frac{1}{L+1} \sum_{l=n-2N+2}^{n-N+1} x(n)\, y(n+j)$$

Simulink Model

for the calculation of the cross correlation



Error power in dependence of the filter coefficients

ele n falou disso, mas e bom revisar

- Error signal:

$$\begin{aligned} e(n) &= y(n) - \sum_{i=0}^{N-1} x(n-i)\,\hat{h}_i \\ &= y(n) - \hat{\boldsymbol{h}}^T \boldsymbol{x}(n) \end{aligned}$$

- Error power for a general filter $\hat{\boldsymbol{h}}$:

$$\begin{aligned} \mathrm{E}\{e^2(n)\} &= \mathrm{E}\Big\{\Big(y(n) - \hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{x}(n)\Big)\Big(y(n) - \hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{x}(n)\Big)\Big\} \\ &= \mathrm{E}\Big\{y(n)\,y(n)\Big\} - 2\,\underbrace{\mathrm{E}\Big\{y(n)\,\hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{x}(n)\Big\}}_{\hat{\mathbf{h}}^T \mathrm{E}\{y(n)\,\mathbf{x}(n)\}} + \underbrace{\mathrm{E}\Big\{\hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{x}(n)\,\hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{x}(n)\Big\}}_{\hat{\mathbf{h}}^T \mathrm{E}\{\mathbf{x}(\mathbf{n})\mathbf{x}(\mathbf{n})^{\mathbf{T}}\}\,\hat{\mathbf{h}}} \\ &= r_{yy}(0) - 2\,\hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{r}_{xy}(0) + \hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{R}_{xx}\,\hat{\boldsymbol{h}} \end{aligned}$$

Error power in dependence of the filter coefficients

- Error power for the optimal filter vector $\hat{\boldsymbol{h}}_{\mathrm{opt}}$:

$$\mathrm{E}\{e^2(n)\}\Big|_{\min} = r_{yy}(0) - 2\,\hat{\boldsymbol{h}}_{\mathrm{opt}}^{\mathrm{T}}\boldsymbol{r}_{xy}(0) + \hat{\boldsymbol{h}}_{\mathrm{opt}}^{\mathrm{T}}\boldsymbol{R}_{xx}\,\hat{\boldsymbol{h}}_{\mathrm{opt}}$$

with: $\hat{\boldsymbol{h}}_{\mathrm{opt}} = \boldsymbol{R}_{xx}^{-1}\boldsymbol{r}_{xy}(0)$

$$\mathrm{E}\{e^2(n)\}\Big|_{\min} = r_{yy}(0) - 2\left(\boldsymbol{R}_{xx}^{-1}\boldsymbol{r}_{xy}(0)\right)^{\mathrm{T}}\boldsymbol{r}_{xy}(0) + \overbrace{\left(\boldsymbol{R}_{xx}^{-1}\boldsymbol{r}_{xy}(0)\right)^{\mathrm{T}}}^{\left(\boldsymbol{R}_{xx}^{-1}\,\boldsymbol{r}_{xy}(0)\right)^{\mathrm{T}} = \boldsymbol{r}_{xy}^{\mathrm{T}}(0)\,\boldsymbol{R}_{xx}^{-1}}\boldsymbol{R}_{xx}\,\boldsymbol{R}_{xx}^{-1}\boldsymbol{r}_{xy}(0)$$

with: $\left(\boldsymbol{R}_{xx}^{-1}\right)^{\mathrm{T}} = \boldsymbol{R}_{xx}^{-1}$

$$\mathrm{E}\{e^2(n)\}\Big|_{\min} = r_{yy}(0) - 2\,\boldsymbol{r}_{xy}^{\mathrm{T}}(0)\,\boldsymbol{R}_{xx}^{-1}\,\boldsymbol{r}_{xy}(0) + \boldsymbol{r}_{xy}^{\mathrm{T}}(0)\,\boldsymbol{R}_{xx}^{-1}\,\boldsymbol{r}_{xy}(0)$$

$$\boxed{\begin{aligned}\mathrm{E}\{e^2(n)\}\Big|_{\min} &= r_{yy}(0) - \boldsymbol{r}_{xy}^{\mathrm{T}}(0)\,\boldsymbol{R}_{xx}^{-1}\,\boldsymbol{r}_{xy}(0)\\ &= r_{yy}(0) - \boldsymbol{r}_{xy}^{\mathrm{T}}(0)\,\hat{\boldsymbol{h}}_{\mathrm{opt}}\end{aligned}}$$

Error power in dependence of the filter coefficients

- Abbreviation:

$$E_{\min} \;=\; \mathrm{E}\{e^2(n)\}\Big|_{\min} \;=\; r_{yy}(0) - \boldsymbol{r}_{xy}^{\mathrm{T}}(0)\,\boldsymbol{R}_{xx}^{-1}\,\boldsymbol{r}_{xy}(0)$$

- Error power in dependence of the minimum power:

$$\mathrm{E}\{e^2(n)\} \;=\; r_{yy}(0) \;-\; 2\,\hat{\boldsymbol{h}}^{\mathrm{T}}\boldsymbol{r}_{xy}(0) \;+\; \hat{\boldsymbol{h}}^{\mathrm{T}}\boldsymbol{R}_{xx}\,\hat{\boldsymbol{h}}$$

with: $r_{yy}(0) = E_{\min} + \boldsymbol{r}_{xy}^{\mathrm{T}}(0)\,\boldsymbol{R}_{xx}^{-1}\,\boldsymbol{r}_{xy}(0)$ (s. above)

$$\mathrm{E}\{e^2(n)\} \;=\; E_{\min} \;+\; \boldsymbol{r}_{xy}^{\mathrm{T}}(0)\,\boldsymbol{R}_{xx}^{-1}\,\boldsymbol{r}_{xy}(0) - 2\,\hat{\boldsymbol{h}}^{\mathrm{T}}\boldsymbol{r}_{xy}(0) \;+\; \hat{\boldsymbol{h}}^{\mathrm{T}}\boldsymbol{R}_{xx}\,\hat{\boldsymbol{h}}$$

with: $\boldsymbol{r}_{xy}(0) \;=\; \boldsymbol{R}_{xx}\,\hat{\boldsymbol{h}}_{\mathrm{opt}}$ Wiener equation

$$\mathrm{E}\{e^2(n)\} \;=\; E_{\min} \;+\; \hat{\boldsymbol{h}}_{\mathrm{opt}}^{\mathrm{T}}\,\boldsymbol{R}_{xx}\,\boldsymbol{R}_{xx}^{-1}\,\boldsymbol{R}_{xx}\,\hat{\boldsymbol{h}}_{\mathrm{opt}} - 2\,\hat{\boldsymbol{h}}^{\mathrm{T}}\boldsymbol{R}_{xx}\,\hat{\boldsymbol{h}}_{\mathrm{opt}} \;+\; \hat{\boldsymbol{h}}^{\mathrm{T}}\boldsymbol{R}_{xx}\,\hat{\boldsymbol{h}}$$

Error power in dependence of the filter coefficients

- Last equation:

$$\mathrm{E}\{e^2(n)\} \;=\; E_{\min} + \hat{\boldsymbol{h}}_{\mathrm{opt}}^{\mathrm{T}} \boldsymbol{R}_{xx} \boldsymbol{R}_{xx}^{-1} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}_{\mathrm{opt}} - 2\,\hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}_{\mathrm{opt}} + \hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}$$

with the scalar element: $\hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}_{\mathrm{opt}} = \left(\hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}_{\mathrm{opt}}\right)^{T} = \hat{\boldsymbol{h}}_{\mathrm{opt}}^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}$

$$\begin{aligned}\mathrm{E}\{e^2(n)\} &= E_{\min} + \hat{\boldsymbol{h}}_{\mathrm{opt}}^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}_{\mathrm{opt}} - \hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}_{\mathrm{opt}} - \hat{\boldsymbol{h}}_{\mathrm{opt}}^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}} + \hat{\boldsymbol{h}}^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}} \\ &= E_{\min} - \left(\hat{\boldsymbol{h}} - \hat{\boldsymbol{h}}_{\mathrm{opt}}\right)^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}_{\mathrm{opt}} + \left(\hat{\boldsymbol{h}} - \hat{\boldsymbol{h}}_{\mathrm{opt}}\right)^{\mathrm{T}} \boldsymbol{R}_{xx} \hat{\boldsymbol{h}}\end{aligned}$$

$$\boxed{\mathrm{E}\{e^2(n)\} \;=\; E_{\min} + \left(\hat{\boldsymbol{h}} - \hat{\boldsymbol{h}}_{\mathrm{opt}}\right)^{\mathrm{T}} \boldsymbol{R}_{xx} \left(\hat{\boldsymbol{h}} - \hat{\boldsymbol{h}}_{\mathrm{opt}}\right)}$$

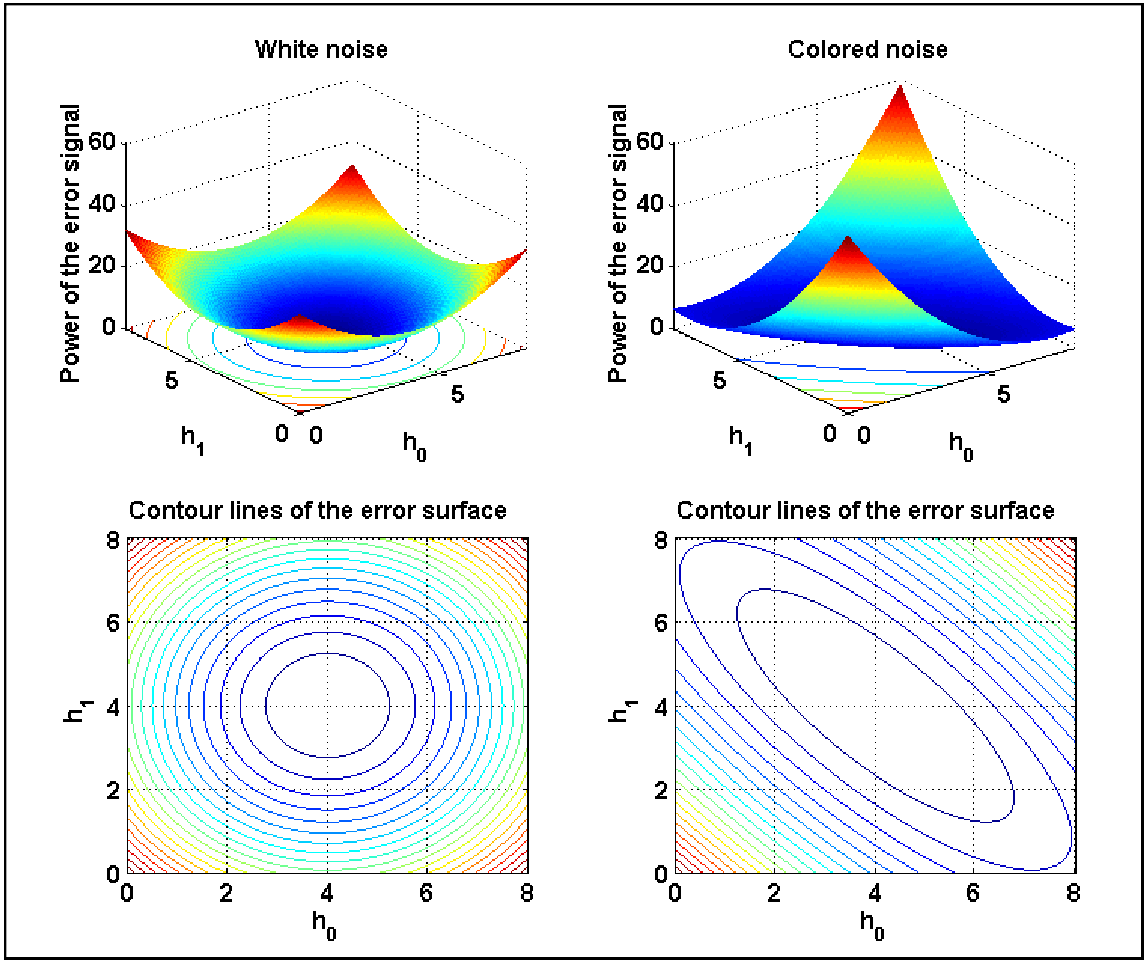
Error surface for filter length N = 2

diferentes propriedades de convergencia do problema convexo (deepest descent do colored noise é diferente do white noise que é simplesmente ortogonal as curvas de nivel)

Error surface for:

- $\boldsymbol{R}_{xx} = \begin{bmatrix} 1.0 & 0 \\ 0 & 1.0 \end{bmatrix}$

- $\boldsymbol{R}_{xx} = \begin{bmatrix} 1.0 & 0.8 \\ 0.8 & 1.0 \end{bmatrix}$

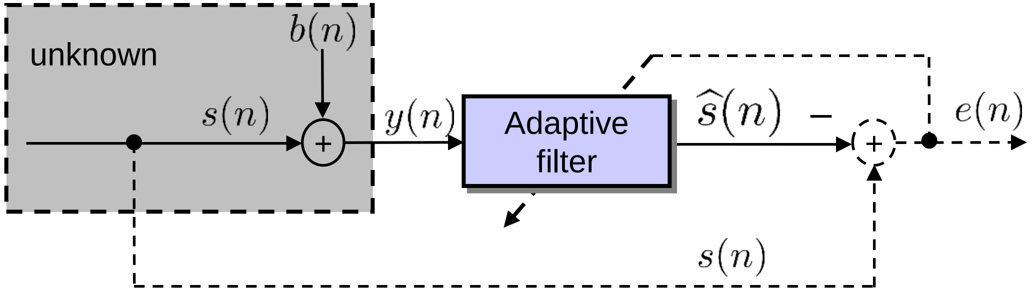


Properties:

- Unique minimum (no local minima)
- Error surface depends on the correlation properties of the input signal

Wiener filter for noise reduction

- Signals and system parameters:



$s(n)$: Desired signal (not accessible but assumed to be known for the filter derivation)

$b(n)$: Noise / Disturbance (unknown)

$y(n)$: Disturbed system output (accessible)

$$\hat{\boldsymbol{h}} = \left[\hat{h}_0,\, \hat{h}_1,\, \dots,\, \hat{h}_{N-1}\right]^{\mathrm{T}}$$: Time-invariant impulse response of the Wiener filter

$$\widehat{s}(n) = \sum_{i=0}^{N-1} y(n-i)\, \hat{h}_i$$: Output signal of the Wiener filter

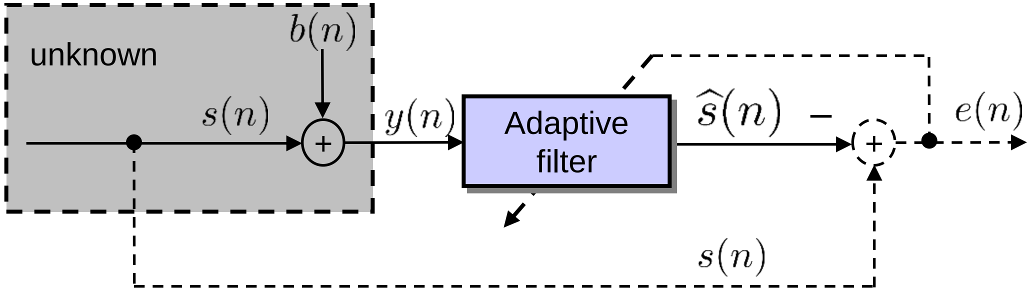
$$\begin{aligned} e(n) &= s(n) - \hat{s}(n) \\ &= s(n) - \sum_{i=0}^{N-1} y(n-i)\, \hat{h}_i \end{aligned}$$: Error signal

- Optimization criterion: $$\mathrm{E}\{e^2(n)\} \underset{\hat{\boldsymbol{h}}=\hat{\boldsymbol{h}}_{\mathrm{opt}}}{\longrightarrow} \min$$

Wiener filter for noise reduction

- Optimization criterion:

$$\mathrm{E}\{e^2(n)\} \underset{\hat{\boldsymbol{h}}=\hat{\boldsymbol{h}}_{\mathrm{opt}}}{\longrightarrow} \min$$



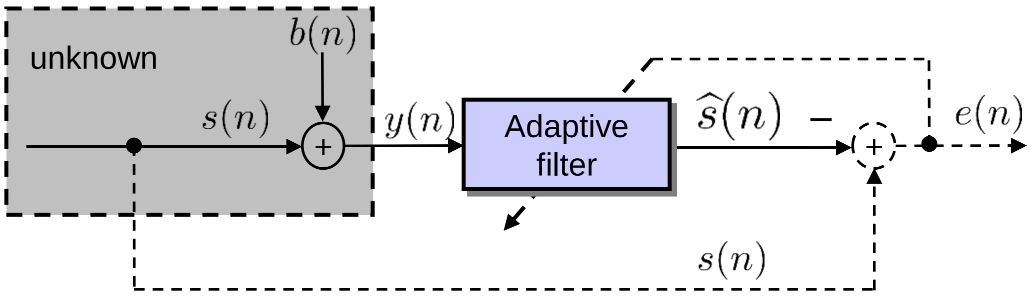
$$\Rightarrow \quad \frac{\partial}{\partial h_j}\mathrm{E}\{e^2(n)\} \stackrel{!}{=} 0$$

$$\Rightarrow \quad 2\mathrm{E}\left\{e(n)\frac{\partial}{\partial h_j}\left(s(n) - \sum_{i=0}^{N-1} y(n-i)\hat{h}_i\right)\right\} = 0$$

$$\Rightarrow \quad \mathrm{E}\left\{e(n)\, y(n-j)\right\} \stackrel{!}{=} 0 \;\text{ for all: } j \in \{0,\ldots,N-1\}$$: Orthogonality theorem

$$\Rightarrow \quad \hat{\boldsymbol{h}}_{\mathrm{opt}} = \boldsymbol{R}_{yy}^{-1}\, r_{ys}(0)$$: by using the relations of the Wiener filter derivation

Wiener filter for noise reduction
=> Frequency domain solution



- Conditions for the filter in the time domain:

$$r_{ys}(i) = \sum_{i=0}^{N-1} \hat{h}_{\mathrm{opt},i}\, r_{yy}(j-i) \qquad \text{for} \quad j \in \{0,\, \ldots,\, N-1\}$$

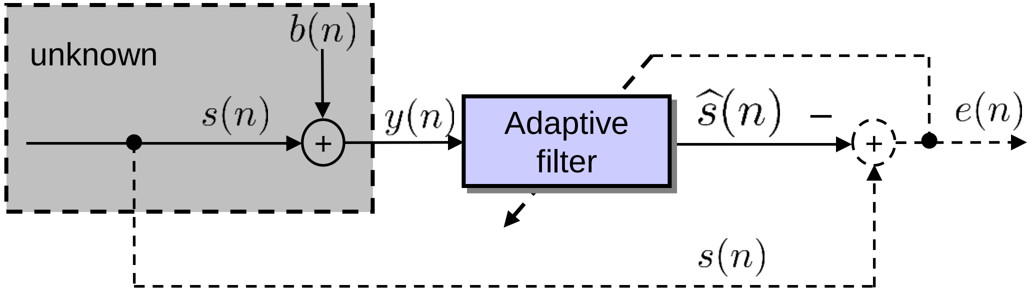
- Assuming a filter of infinite length:

$$r_{ys}(i) = \sum_{i=-\infty}^{\infty} \hat{h}_{\mathrm{opt},i}\, r_{yy}(j-i) \qquad \text{for} \quad j \in \mathbf{Z}$$

- Transformation into the frequency domain:

$$\sum_{u=-\infty}^{\infty} r_{ys}(u)\, e^{-j\,\Omega\,u} \quad = \quad \sum_{u=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} \hat{h}_{\mathrm{opt},i}\, r_{yy}(u-i)\, e^{-j\,\Omega\,u}$$

Wiener filter for noise reduction
=> Frequency domain solution



- Transformation into the frequency domain:

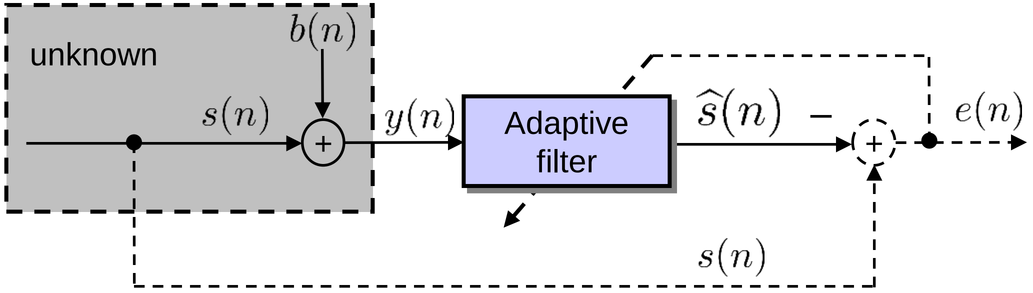
$$\sum_{u=-\infty}^{\infty} r_{ys}(u)\, e^{-j\,\Omega\,u} = \sum_{u=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} \hat{h}_{\mathrm{opt},i}\, r_{yy}(u-i)\, e^{-j\,\Omega\,u}$$

$$\sum_{u=-\infty}^{\infty} r_{ys}(u)\, e^{-j\,\Omega\,u} = \sum_{u=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} \hat{h}_{\mathrm{opt},i}\, r_{yy}(u-i)\, e^{-j\,\Omega\,i}\, e^{-j\,\Omega\,(u-i)}$$

$$\sum_{u=-\infty}^{\infty} r_{ys}(u)\, e^{-j\,\Omega\,u} = \sum_{u=-\infty}^{\infty} \sum_{i=-\infty}^{\infty} \left[\hat{h}_{\mathrm{opt},i}\, e^{-j\,\Omega\,i}\right] r_{yy}(u-i)\, e^{-j\,\Omega\,(u-i)}$$

$$\underbrace{\sum_{u=-\infty}^{\infty} r_{ys}(u)\, e^{-j\,\Omega\,u}}_{S_{ys}(\Omega)} = \underbrace{\sum_{i=-\infty}^{\infty} \left[\hat{h}_{\mathrm{opt},i}\, e^{-j\,\Omega\,i}\right]}_{\hat{H}_{\mathrm{opt}}(e^{j\Omega})} \underbrace{\sum_{u=-\infty}^{\infty} r_{yy}(u)\, e^{-j\,\Omega\,u}}_{S_{yy}(\Omega)}$$

Wiener filter for noise reduction
=> Frequency domain solution



- Solution in the frequency domain:

$$S_{ys}(\Omega) \quad = \quad \hat{H}_{\text{opt}}\left(e^{j\Omega}\right)\, S_{yy}(\Omega)$$

$$\hat{H}_{\text{opt}}\left(e^{j\Omega}\right) \quad = \quad \frac{S_{ys}(\Omega)}{S_{yy}(\Omega)}$$

- Summary of time and frequency domain solutions:

Time domain solution:	Frequency domain solution:
$\hat{\boldsymbol{h}}_{\text{opt}} = \boldsymbol{R}_{yy}^{-1}\, \boldsymbol{r}_{ys}(0)$	$\hat{H}_{\text{opt}}(e^{j\,\Omega}) = \dfrac{S_{ys}(\Omega)}{S_{yy}(\Omega)}$

Noise reduction (I)

Frequency-domain Wiener solution (non-causal):

$$\hat{H}_{\text{opt}}\left(e^{j\Omega}\right) = \frac{S_{ys}(\Omega)}{S_{yy}(\Omega)}$$

Desired signal and noise are orthogonal:

$$y(n) = s(n) + b(n) \qquad S_{ys}(\Omega) = S_{ss}(\Omega) + S_{bs}(\Omega)$$

$$S_{yy}(\Omega) = S_{ss}(\Omega) + S_{bs}(\Omega) + S_{bs}^{*}(\Omega) + S_{bb}(\Omega)$$

$$\mathrm{E}\{s(n)\,b(n)\} = 0$$

=> $S_{bs}(\Omega) = 0$ => $\begin{cases} S_{ys}(\Omega) = S_{ss}(\Omega) \\ S_{yy}(\Omega) = S_{ss}(\Omega) + S_{bb}(\Omega) \end{cases}$

$$\Longrightarrow \qquad \hat{H}_{\text{opt}}\left(e^{j\Omega}\right) = \frac{S_{ys}(\Omega)}{S_{yy}(\Omega)} = \frac{S_{ss}(\Omega) + \cancel{S_{bs}(\Omega)}}{S_{yy}(\Omega)}$$

$$= \frac{S_{ss}(\Omega)}{S_{yy}(\Omega)} = \frac{S_{yy}(\Omega) - S_{bb}(\Omega)}{S_{yy}(\Omega)}$$

$$\boxed{\hat{H}_{\text{opt}}\left(e^{j\Omega}\right) = 1 - \frac{S_{bb}(\Omega)}{S_{yy}(\Omega)}}$$

Noise reduction (II)

nós usamos o assumption de que o ruido é estacionario, e isso faz com que não caiamos no caso em que não sabemos nada sobre o sistema

Frequency-domain solution:

$$\widehat{H}_{\mathrm{opt}}(e^{j\Omega}) \quad = \quad 1 - \frac{S_{bb}(\Omega)}{S_{yy}(\Omega)}$$

Approximation using short-term estimations:

$$\widehat{H}_{\mathrm{opt}}(e^{j\Omega}, n) \quad = \quad \max\left\{1 - \frac{\widehat{S}_{bb}(\Omega, n)}{\widehat{S}_{yy}(\Omega, n)},\, 0\right\}$$

se Syy = Sbb, o filtro atenua tudo
faz sentido pq vc quer cortar tudo

esse 0 aqui faz sentido pq na pratica, como os dois são
estimadores, pode acontecer de ter um valor menor de 0

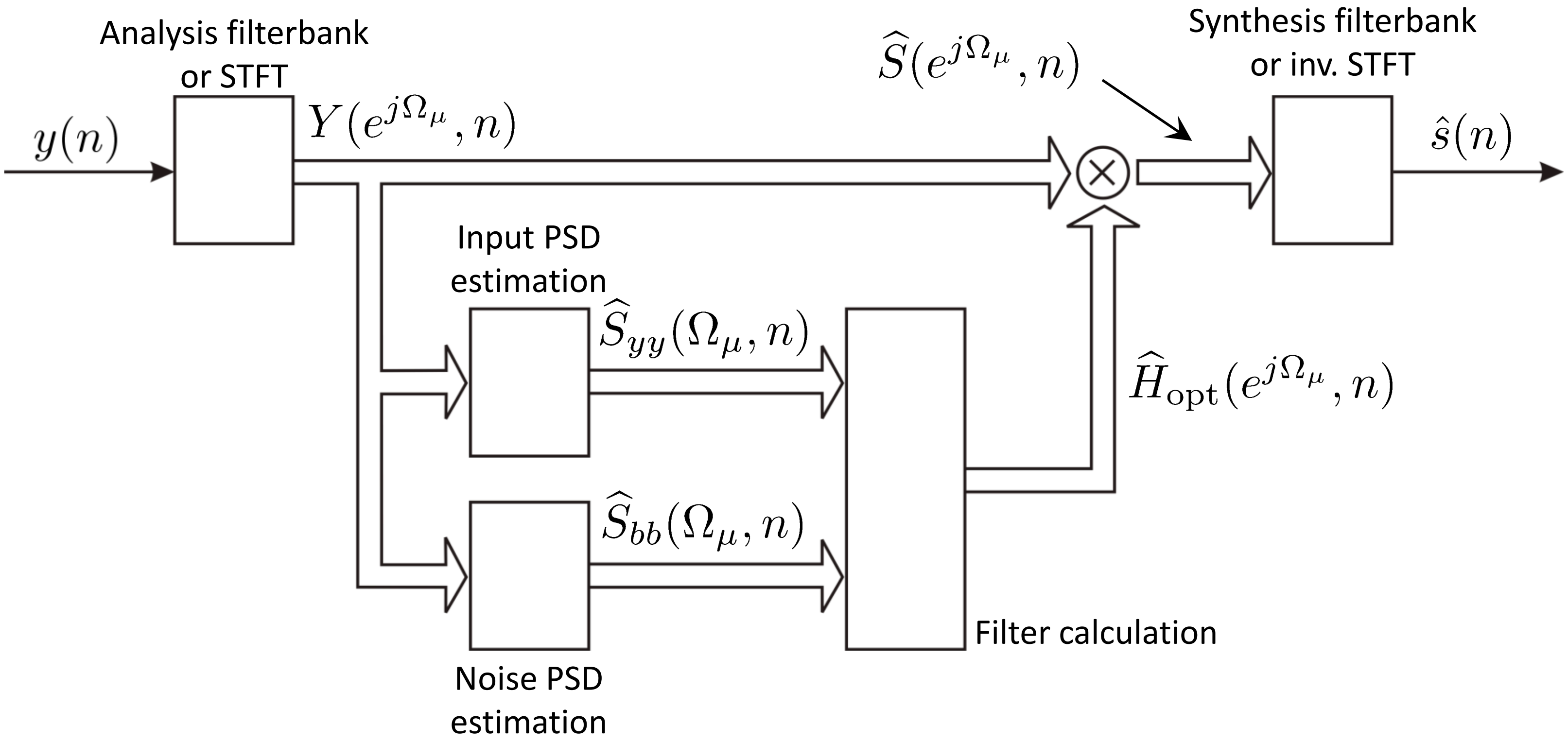
n: frame index

Practical approaches:

- Realization using a filterbank system (time-variant attenuation of subband signals) or an STFT (Short-Time Fourier Transform)
- Analysis filters with length of about 15 to 100 ms
- Frame-based processing with frame shifts between 1 and 20 ms
- The basic Wiener characteristic is usually „enriched“ with several extensions (overestimation, limitation of the attenuation, etc.)

Noise reduction (III): Processing in discrete freq. bands

Processing structure:



PSD = power spectral density

Noise reduction (IV): PSD estimation

TECHNISCHE UNIVERSITÄT DARMSTADT

Power spectral density estimation for the input signal:

$$\widehat{S}_{yy}(\Omega_\mu, n) = \left|Y(e^{j\Omega_\mu}, n)\right|^2$$

Theory behind:
Estimation of PSDs with „periodograms“

Power spectral density estimation for the noise:

Estimation schemes using voice activity detection (VAD)	Tracking of minima of short-term power estimations

Noise reduction (V): Possibilities for noise PSD estimation

Two alternatives:

1) Schemes with voice activity detection (VAD):

tentar simular isso aqui/ como ficaria a representação em frequencia disso?

$$\widehat{S}_{bb}(\Omega_\mu, n) = \begin{cases} \beta\, \widehat{S}_{bb}(\Omega_\mu, n-1) + (1-\beta)\, \widehat{S}_{yy}(\Omega_\mu, n), & \text{during speech pauses,} \\ \widehat{S}_{bb}(\Omega_\mu, n-1), & \text{else.} \end{cases}$$

2) Tracking of minima of the short-term power:

1) Smoothing:

$$\overline{S_{yy}(\Omega_\mu, n)} \quad = \quad \beta\, \overline{S_{yy}(\Omega_\mu, n-1)} + (1-\beta)\, \widehat{S}_{yy}(\Omega_\mu, n)$$

2) Minimum value, with a slight increase to avoid a freezing of the estimate:

$$\widehat{S}_{bb}(\Omega_\mu, n) = \min\left\{\overline{S_{yy}(\Omega_\mu, n)}, \widehat{S}_{bb}(\Omega_\mu, n-1)\right\}\, (1+\epsilon) \text{ with: } \epsilon << 1$$

ϵ : determines the tracking capabilities of the estimator

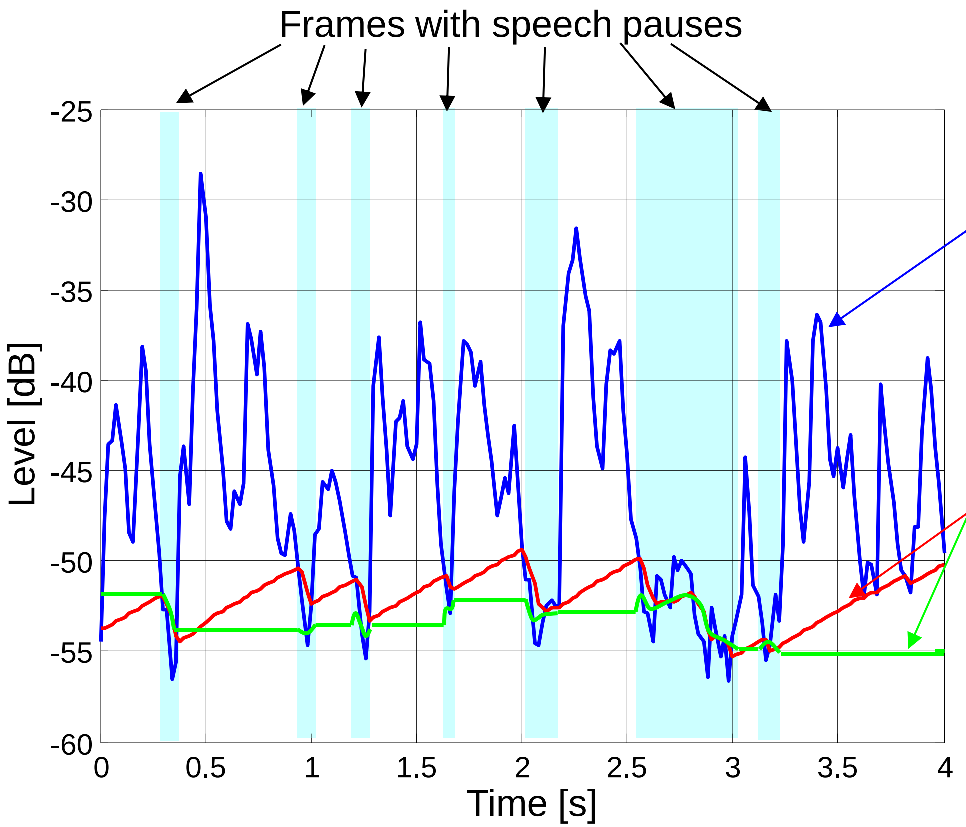
no caso em que o ruido fica estacionario
esse 1+eps não é bom

o eps é determinado no projeto pensando
em quao rapido isso pode subri

Noise reduction (Vb): Possibilities for noise PSD estimation

TECHNISCHE UNIVERSITÄT DARMSTADT

Analysis for one discrete frequency component:



$$\widehat{S}_{yy}(\Omega_\mu, n) = \left|Y(e^{j\Omega_\mu}, n)\right|^2$$

1) Estimation with voice activity detection

2) Estimation with minimum power tracking:
Racing slopes are determined by the constant ϵ

Noise reduction (VI)

TECHNISCHE UNIVERSITÄT DARMSTADT

Problem:

- The short-term power of the input signal usually fluctuates faster than the noise estimate – also during speech pauses. As a result, the filter characteristic opens and closes in a randomized manner, which results in tonal residual noise (so-called musical noise).

Simple solution:

- By inserting a fixed overestimation

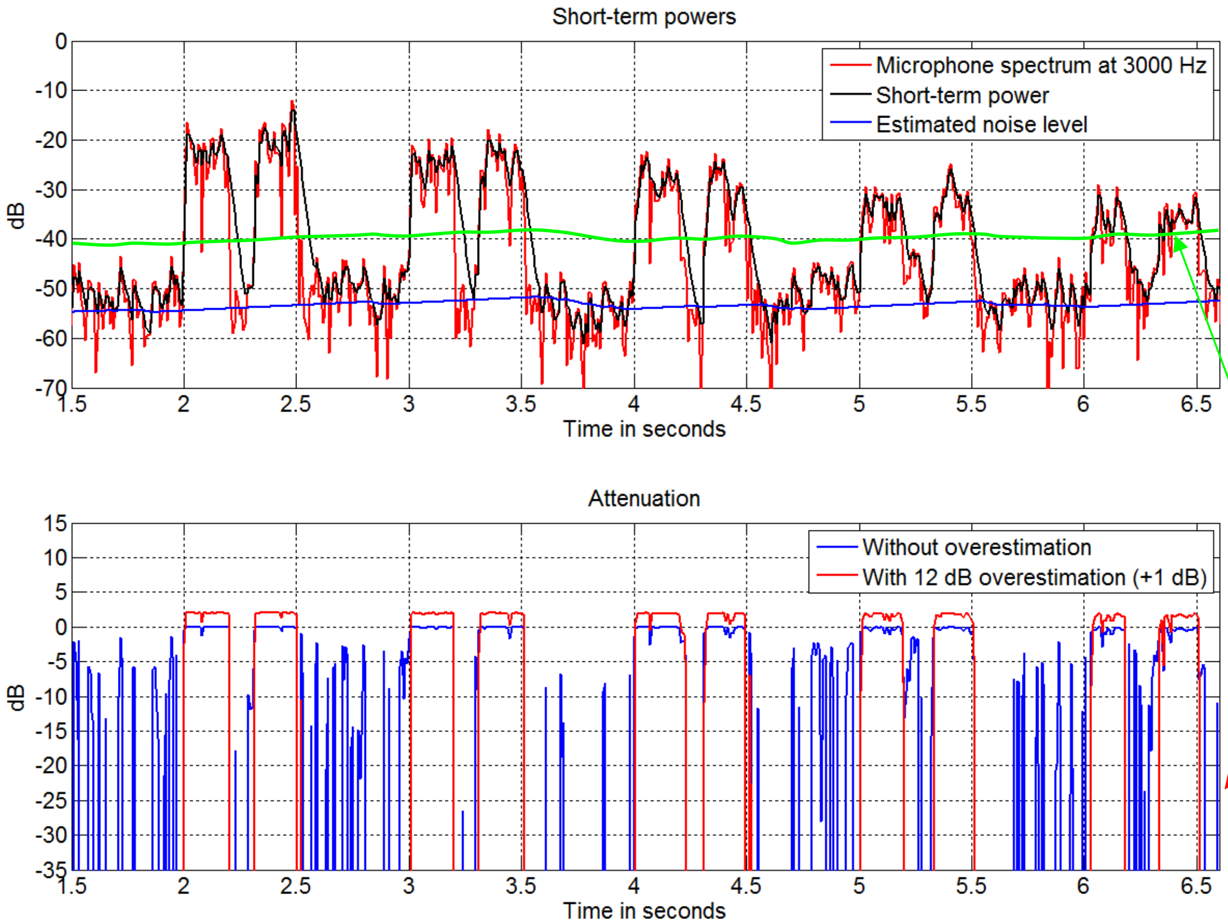
$$\widehat{S}_{bb}(\Omega_\mu, n) \longrightarrow K_{\text{over}}\, \widehat{S}_{bb}(\Omega_\mu, n)$$

  the randomized opening of the filter can be avoided. This comes, however, with a more aggressive attenuation characteristic that attenuates also parts of the speech signal.

Enhanced solutions:

- More enhance solutions will be presented in the lecture “Speech and Audio Processing”.

Noise reduction (VII)



: Microphone signal

: Output without over-estimation

: Output with 12 dB over-estimation

With overestimation

Noise reduction (VII)

Limiting the maximum attenuation:

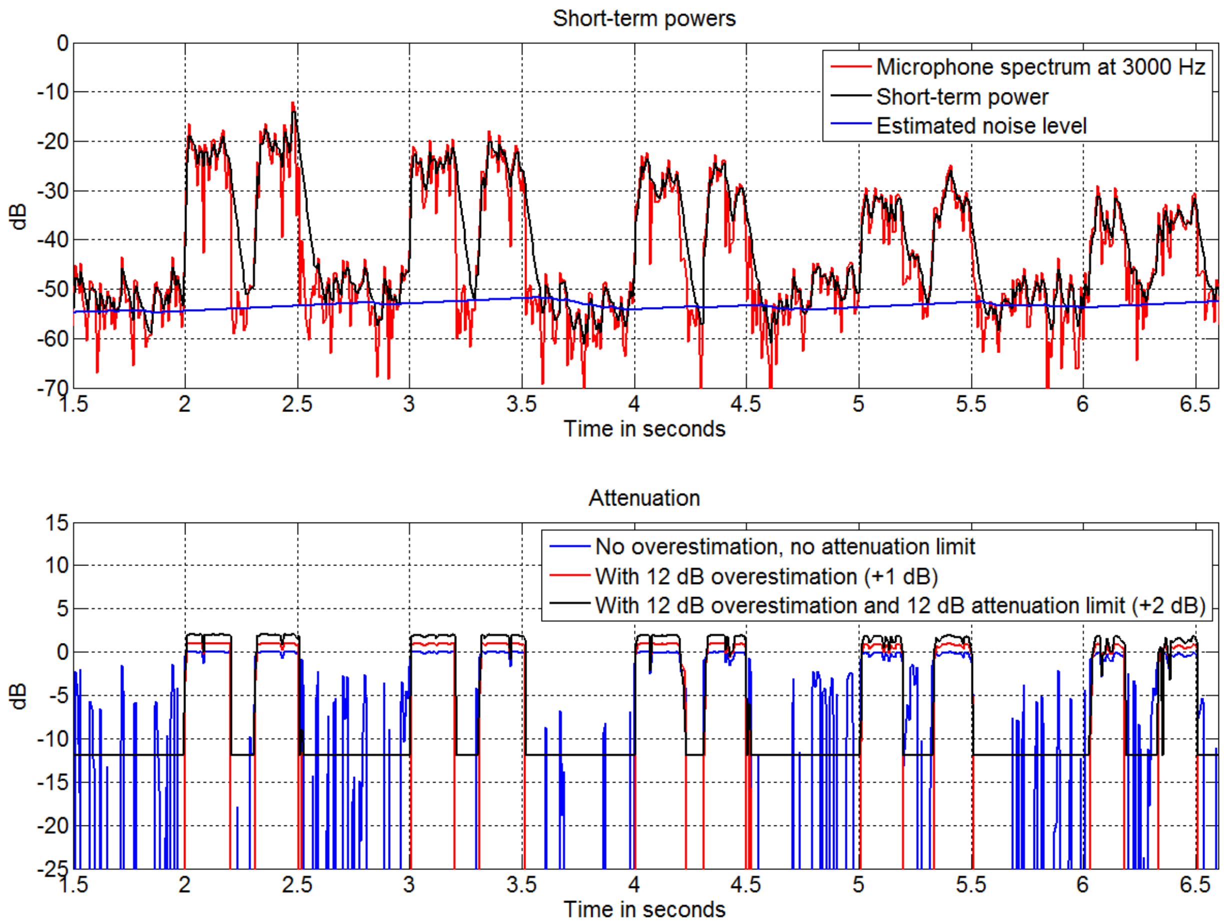
- For several applications, the original shape of the noise should be preserved (the noise should only be attenuated but not completely removed). This could be achieved by inserting a maximum attenuation:

$$H_{\min}(e^{j\Omega_\mu}, n) \quad = \quad H_{\min}.$$

$$\widehat{H}_{\text{opt}}(e^{j\Omega}, n) \quad = \quad \max\left\{1 - K_{\text{over}}\,\frac{\widehat{S}_{bb}(\Omega, n)}{\widehat{S}_{yy}(\Omega, n)},\; H_{\min}\right\}$$

- In addition, this attenuation limits can be varied slowly over time (slightly more attenuation during speech pauses, less attenuation during speech activity).

Noise reduction (IX)



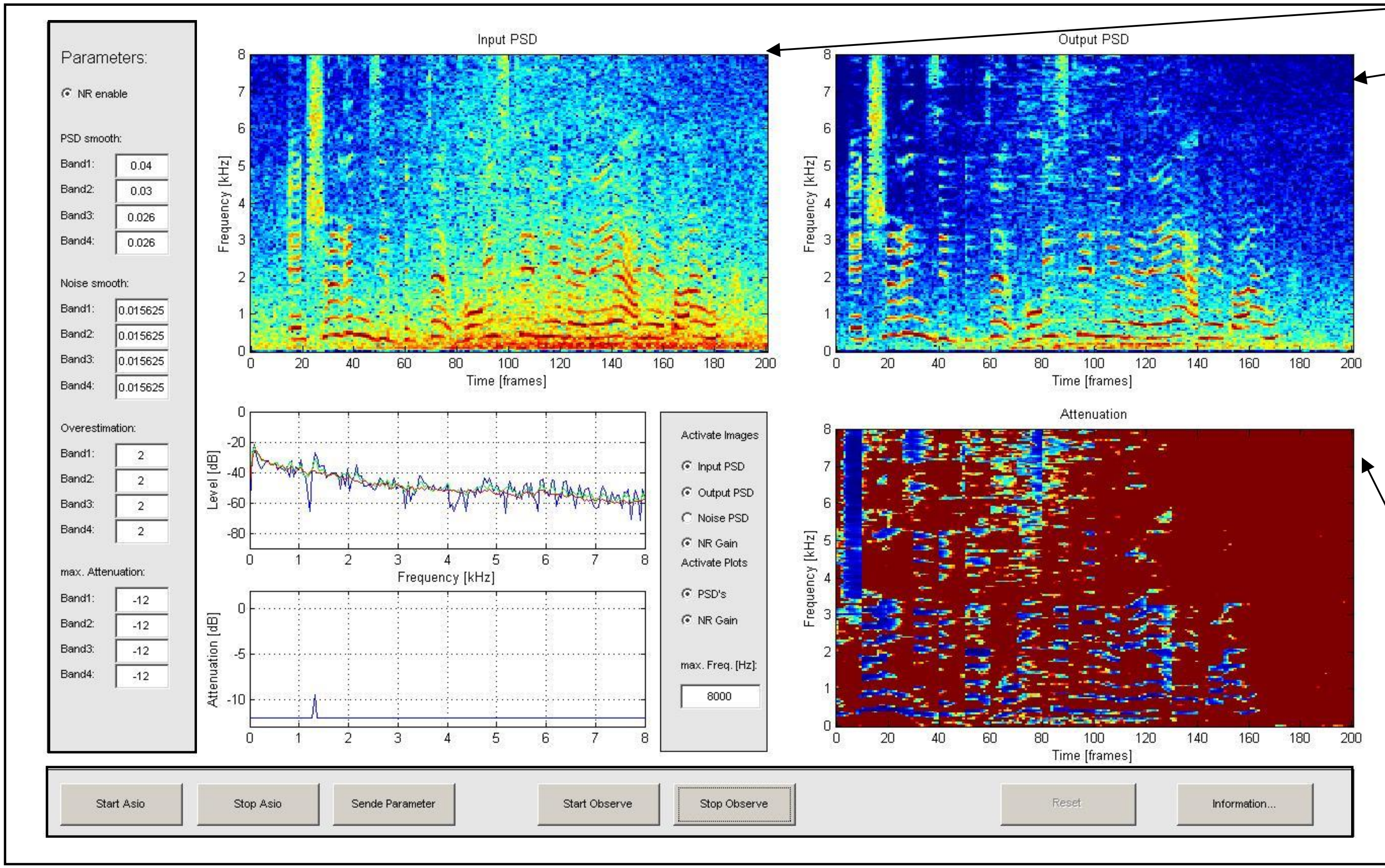
: Microphone signal

: Output without attenuation limit

: Output with attenuation limit

Noise reduction (X): Spectral Comparison

é muito interessante ver esses aqui com microfone e vc fazendo um chiado

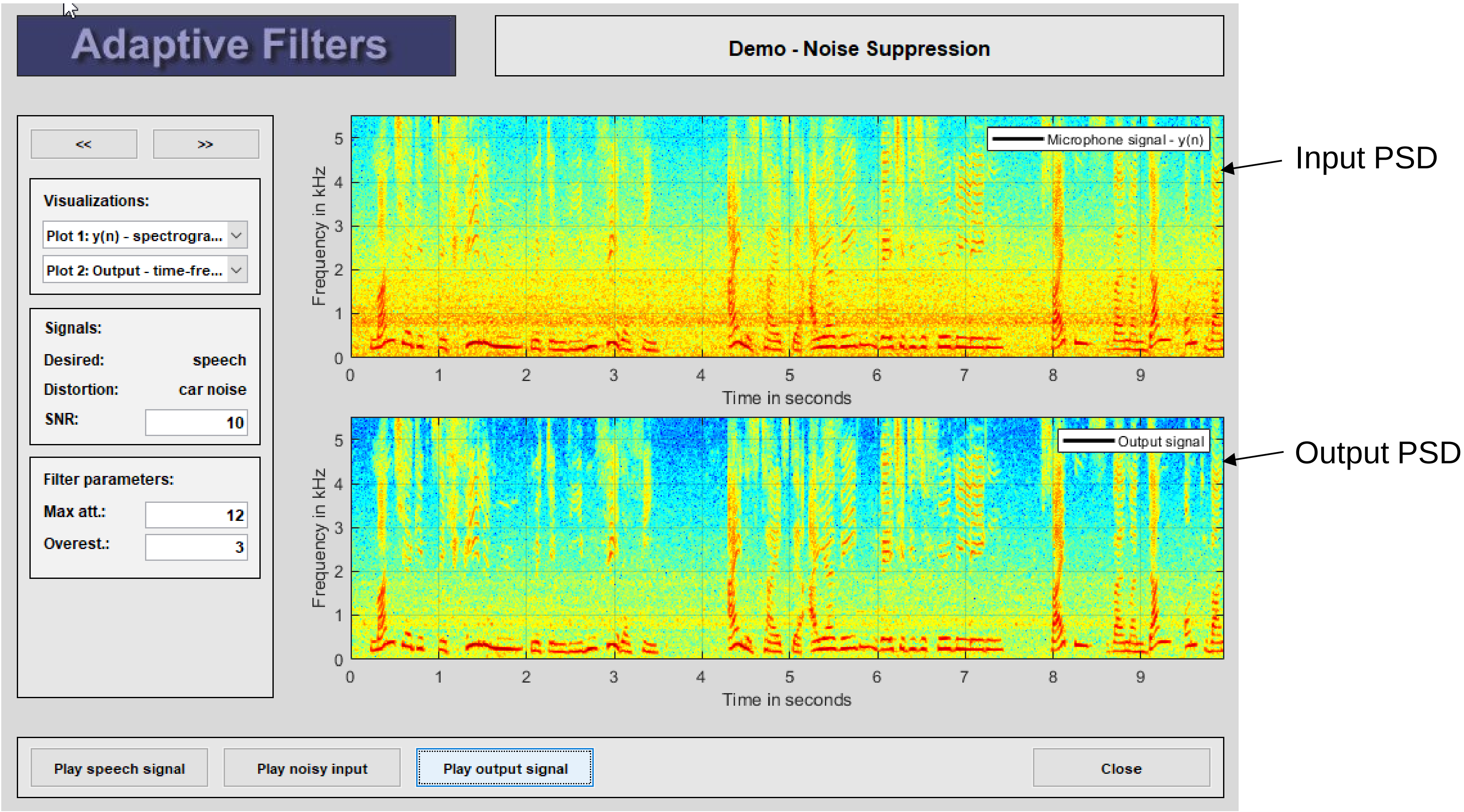


Input PSD

Output PSD

Time-frequency dependent attenuation (**blue**: no attenuation, **red**: max attenuation)

Noise reduction (XI): Demo



Books

Main text:

- E. Hänsler / G. Schmidt: *Acoustic Echo and Noise Control – Kapitel 5 (Wiener Filter),* Wiley, 2004

Additional texts:

- E. Hänsler: *Statistische Signale: Grundlagen und Anwendungen – Kapitel 8 (Optimalfilter nach Wiener und Kolmogoroff),* Springer, 2001 (in German)
- M. S.Hayes: *Statistical Digital Signal Processing and Modeling – Kaptitel 7 (Wiener Filtering),* Wiley, 1996
- S. Haykin: *Adaptive Filter Theory – Kapitel 2 (Wiener Filters),* Prentice Hall, 2002

Noise suppression:

- U. Heute: *Noise Suppression*, in E. Hänsler, G. Schmidt (eds.), Topics in Acoustic Echo and Noise Control, Springer, 2006

Summary & Outlook

TECHNISCHE UNIVERSITÄT DARMSTADT

This week:

- Introduction and motivation
- Principle of orthogonality
- Time-domain solution
- Frequency-domain solution
- Application examples:
  System identification
  Noise suppression

Next week:

- Linear Prediction