Summer University

Speech Recognition

01 July – 30 August 2002

3rd Lecture & Computer Experiments

Digital Signal Processing

Andreas Wendemuth

Schedule Lecture 3: Stochastic Signals & Systems

- Refresh
- Stochastic Signals / Autocorrelation
- System‘s response
- Model systems: analysis and prediction

Random (stochastic) signals

- Stochastic Signal x
- Probability P(x) if discrete
- Probability density p(x) if continuous:
  P(a<x<a+da) = p(x) * da
- Expectation E[f(x)] of a function f(x), where x is distributed with p(x):

$$E[f(x)] = \sum_{alle\,x} f(x)P(x) = \int_x f(x)P(x)\,d$$

Expectations

- Expectation E[.] is a linear function, i.e. it commutes with other linear functions

  example: $f(x) = x^2$ is not linear. Therefore $E[x^2] \neq E[x]^2$
- If $f(x) = x$, expectation gives the mean μ
- If $f(x) = (x-\mu)^2$, expectation gives the variance σ
- Note $E[f(x)*g(y)] = E[f(x)]*E[g(y)]$ only if $P(x,y) = P(x)*P(y)$, i.e. for uncorrelated x,y

Transformation of random signals through LTI systems

$$\mu_{aus} = \mu_{ein} \sum_{n=-\infty}^{\infty} h[n] = \mu_{ein} H(e^{j0}) = \mu_{ein} H(z=1)$$

$$\sigma_{aus} = \sigma_{ein} \sum_{k=-\infty}^{\infty} h[k]^2 = \sigma_{ein} \frac{1}{2\pi j} \oint H(z) H(1/z) z^{-1} \, dz$$

Creation of random signals through random discrete processes

- **A (random discrete) process is stationary, if its properties do not change over time:**

$$E\left\{ g\left(X[k] \right) \right\} = \text{konstant} \qquad \text{für alle} k$$

- **A stationary process is ergodic if the Estimation over different realisations P(x) of the process equals the Estimation over instances (time k) within the process. („spatial mean = temporal mean“)**

$$E\left\{ g\left(X[k] \right) \right\} = <g\left(x[k] \right)> = \text{konstant} \qquad \text{für alle} k$$

Under ergodicity...

- **Mean:**

$$\mu = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{k=-N}^{+N} x[k]$$

- **Variance:**

$$\sigma = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{k=-N}^{+N} | x[k] - \mu |^2$$

Autocorrelation

- **Autocorrelation:**

$$s_{XX}(k_1, k_2) = E\{X(k_1) \cdot X(k_2)\} = E\{X_1 \cdot X_2\}$$

- **If stationary:**

$$s_{XX}(\kappa) = E\{X^*(k) \cdot X(k+\kappa)\}$$

- **... And ergodic:**

$$s_{XX}(\kappa) = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{k=-N}^{+N} x[k] \cdot x[k+\kappa$$

Finite series

$$s_{XX}(\kappa) = \lim_{N \to \infty} \frac{1}{2N+1} \sum_{k=-N}^{+N} x[k] \cdot x[k+\kappa$$

Finite:

$$\hat{s}_{xx}(\kappa) = \frac{1}{N} \sum_{k=0}^{N-1} x[k] \cdot x[k+\kappa]$$

Compute with Fourier transformation:

Define periodic signal with L = power of 2:

$$\tilde{x}[k] = \begin{cases} x[k] & 0 \leqslant k \leqslant N-1 \\ 0 & N \leqslant k \leqslant L-1 \end{cases}$$

$$\tilde{x}[k] = \tilde{x}[k+iL] \qquad \text{mit } i \in \mathbb{Z}$$

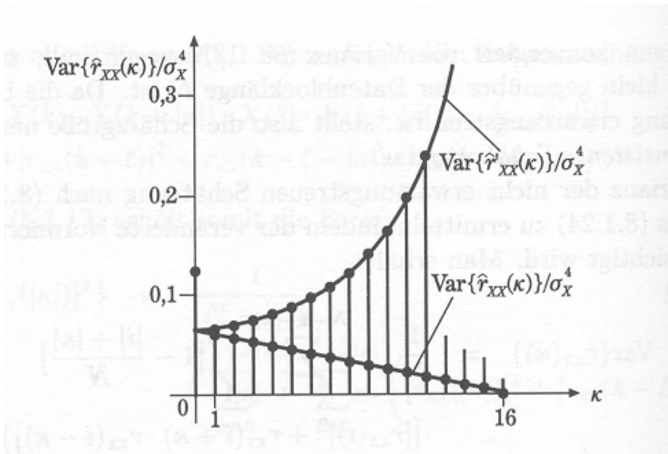
$$\mathrm{FFT}\{\tilde{s}_{xx}(\kappa)\} = \frac{1}{N} X[n] \cdot X[n]$$

Boundaries

True expectation: $\hat{s}_{xx}{}'(|\kappa|) = \frac{1}{N-|\kappa|} \sum_{k=0}^{N-1-|\kappa|} x[k] \cdot x[k+\kappa]$

Konsistent : $\hat{s}_{xx}(|\kappa|) = \frac{1}{N} \sum_{k=0}^{N-1-|\kappa|} x[k] \cdot x[k+\kappa]$

- **Variance (top: true expectation, bottom: konsistent)**



Correlations tells us...

- **Power of a process:**

$$s_{XX}(0) = E\{X^*(k) \cdot X(k+0)\} = E\{|x(k)|^2\} = \sigma + |\mu|$$

- **Uncorrelated processes**

$$s_{XY}(\kappa) = E\{X^*(k) \cdot Y(k+\kappa)\} = E\{X^*(k)\} \cdot E\{Y(k+\kappa)\}$$

- **Orthogonal Processes:**

$$s_{XY}[\kappa] = E\{X^*(k) \cdot Y(k+\kappa)\} = 0$$

Transformation by LTI systems

- **System correlation:**

$$\sum_{i=-\infty}^{\infty} h^*[i] \cdot h[i+m] \;=\; h[m] \;*\; h^*[-m] \;=\; s_{hh}[m]$$

- **Transformation:**

$$s_{YY}[\kappa] \;=\; \sum_{m-\;\;\infty}^{\infty} s_{hh}[m] \cdot s_{XX}[\kappa-m]$$

$$s_{YY}[\kappa] \;=\; s_{hh}[\kappa] \;*\; s_{XX}[\kappa] \;=\; h[\kappa] \;*\; h^*[-\kappa] \;*\; s_{XX}[\kappa]$$

... And after Fourier transform

-
$$S_{YY}(\mathrm{e}^{j\Phi}) \;=\; \boldsymbol{FT}\{s_{yy}[\kappa]\}$$
$$S_{YY}(\mathrm{e}^{j\Phi}) \;=\; \boldsymbol{FT}\{h[\kappa] \;*\; h^*[-\kappa] \;*\; s_{XX}[\kappa]\}$$
$$S_{YY}(\mathrm{e}^{j\Phi}) \;=\; \boldsymbol{FT}\{h[\kappa]\} \;\cdot\; \boldsymbol{FT}\{h^*[-\kappa]\} \;\cdot\; \boldsymbol{FT}\{s_{XX}[\kappa]$$
$$S_{YY}(\mathrm{e}^{j\Phi}) \;=\; H(\mathrm{e}^{j\Phi}) \;\cdot\; H^*(\mathrm{e}^{j\Phi}) \;\cdot\; S_{XX}(\mathrm{e}^{j\Phi})$$
$$S_{YY}(\mathrm{e}^{j\Phi}) \;=\; \left|H(\mathrm{e}^{j\Phi})\right|^2 \;\cdot\; S_{XX}(\mathrm{e}^{j\Phi})$$

- **For „white“ input process with**

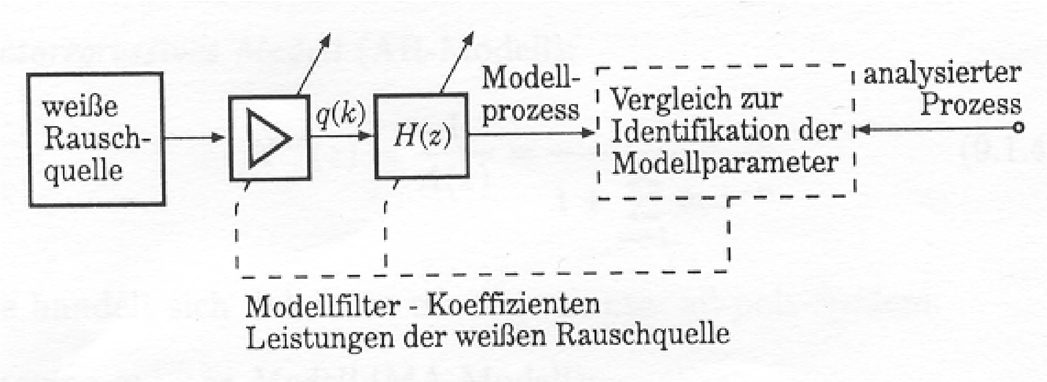
$$s_{XX}[\kappa] \;=\; \delta[\kappa] \cdot \sigma_X \qquad\qquad S_{XX}(\mathrm{e}^{j\Phi}) \;=\; \sigma_X$$

$$S_{YY}(\mathrm{e}^{j\Phi}) \;=\; \left|H(\mathrm{e}^{j\Phi})\right|^2 \cdot S_{XX}(\mathrm{e}^{j\Phi}) \;=\; \sigma_X \cdot \left|H(\mathrm{e}^{j\Phi})\right|^2$$

- **Compare with delta series system input (deterministic)**

Model Systems

- Use difference equations (forward and backward loop!) to model systems with noisy input.



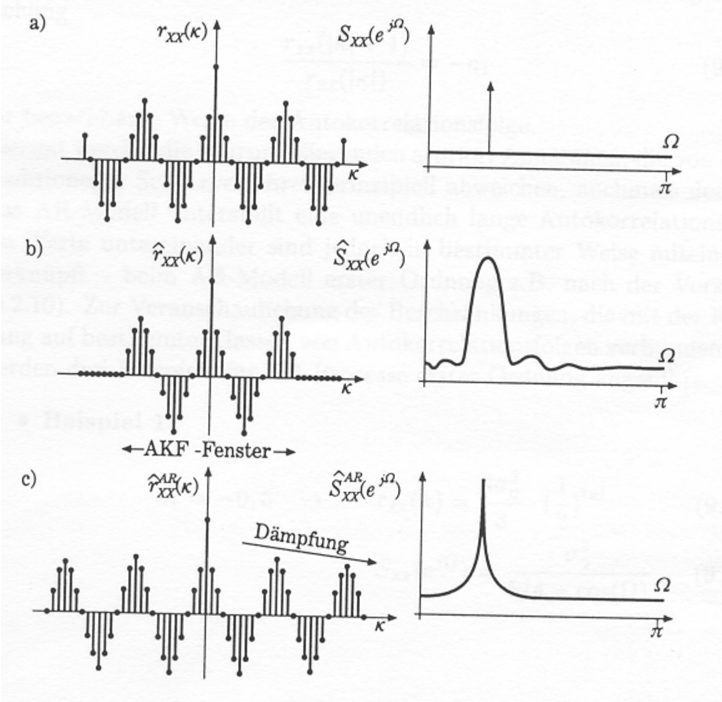
- The systems parameters have to be chosen such that the desired process (normally characterised by ACF) is modelled.
- This works the better, the more components in H(z)

Autoregressive (AR) Model

$$H(z) = \frac{1}{A(z)} = \frac{1}{1 + \sum_{\nu=1}^{n} a_\nu z^{-\nu}}$$

$$x[k] = q[k] - \sum_{\nu=1}^{n} a_\nu \, x[k-\nu]$$

ACF estimation vs. Model System



a) True b) ACF c) AR-Model

Matching Model parameters and ACF values

$$x[k] = q[k] - \sum_{\nu=1}^{n} a_\nu \, x[k-\nu]$$

Compute ACF, compare for all k:

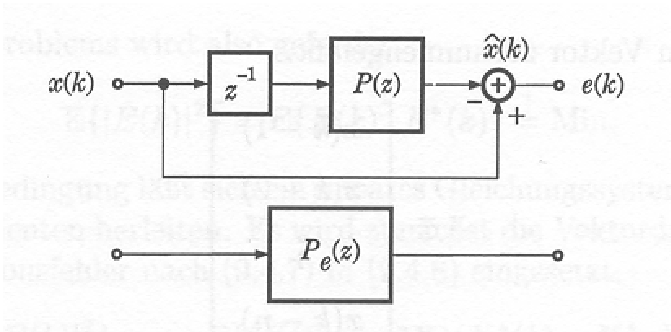
$$\begin{bmatrix} s_{XX}[0] & s_{XX}[-1] & \cdots & s_{XX}[-(n-1)] \\ s_{XX}[1] & s_{XX}[0] & \cdots & s_{XX}[-(n-2)] \\ \vdots & \vdots & \ddots & \vdots \\ s_{XX}[n-1] & s_{XX}[n-2] & \cdots & s_{XX}[0] \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} -s_{XX}[1] \\ -s_{XX}[2] \\ \vdots \\ -s_{XX}[n] \end{bmatrix}$$

Yule-Walker-Equation:

$$\boldsymbol{a} = -\boldsymbol{S}_{XX}^{-1} \cdot \boldsymbol{s}_{XX}$$

Linear Prediction (LP)

- Yule-Walker-Method works as well for LP:



- Result is the Wiener-Hopf-equation (sign!):

$$-\boldsymbol{a} = \boldsymbol{S}_{XX}^{-1} \cdot \boldsymbol{s}_{XX}$$

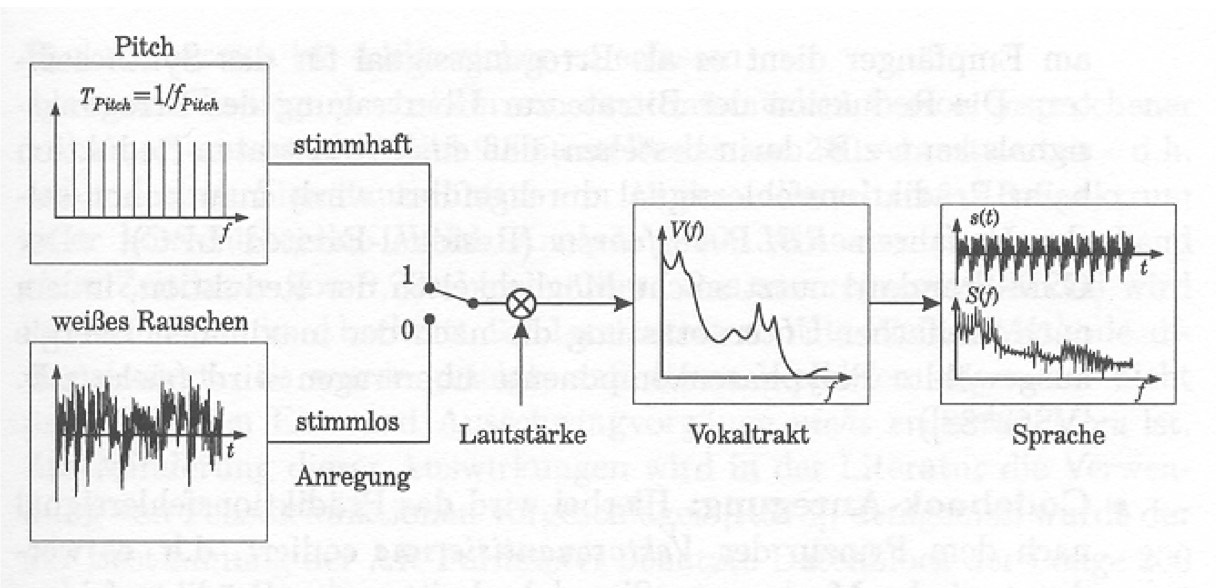
Matrixinversion

- In both the Wiener-Hopf and the Yule-Walker equation, a matrix must be inverted. This Matrix is of special (Toeplitz) type.
- There exist efficient methods for doing this in LINEAR time:
- The Levinson-Durbin recursion computes the parameters of a model of order R, from these the parameters for R+1, etc., until order N.
- So after Levinson-Durbin, we have the coefficients for ALL models of order R <= N.

Prediction and AR-Model

- Suppose the REAL Process was a linear process of order N which was excited by white noise.
- Then if the AR-Model used to analyse this process is also of order N (or higher!), the output of the AR model will be white noise again. So there is a perfect match.
- If the AR model has order < N, or if the real process was not linear, the AR model will produce a best match in terms of minimal quadratic error of the ACF.

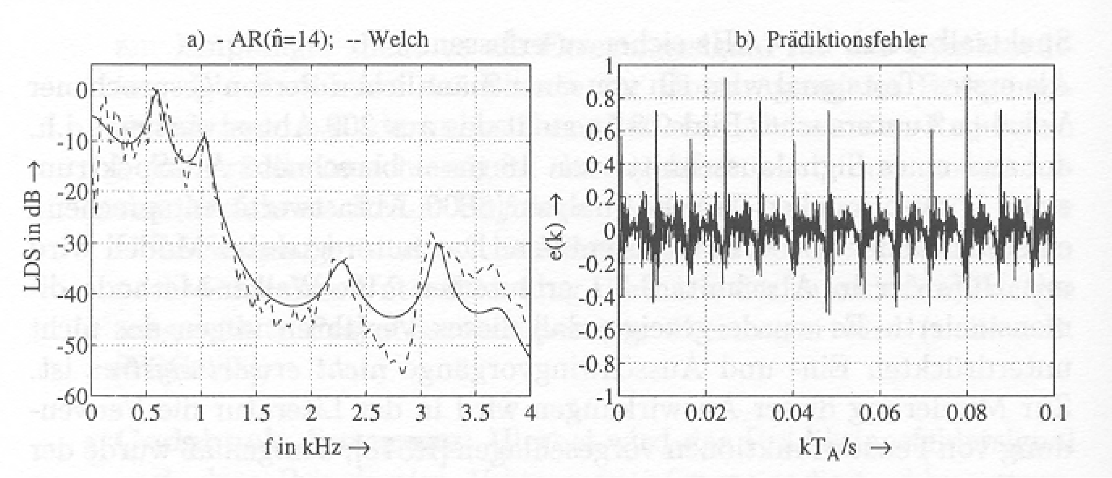
Speech Production: Source-Filter-Model



Modelling phonems

» *Spektralanalyse des Vokals "a"*
a) Compare
- AR-Modell (N= 200, n=14)
- Welch-Schätzung (N=8000, L=256)

» *B) prediction error*



Review Lecture 3: Stochastic Signals & Systems

- **Refresh**
- **Stochastic Signals / Autocorrelation**
- **System‘s response**
- **Model systems: analysis and prediction**