Lesson 1

Optimal Signal Processing

Optimal signalbehandling

LTH

September 2013

Statistical Digital Signal Processing and Modeling,

Hayes, M:

John Wiley & Sons, 1996. ISBN 0471594318

Nedelko Grbic

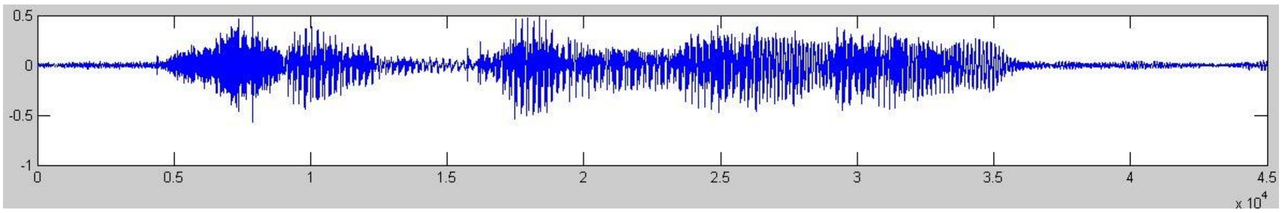
(Mtrl from Bengt Mandersson)

Department of

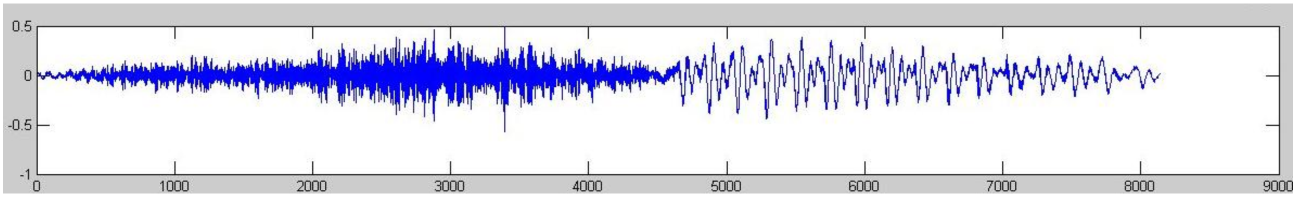
Electrical and Information Technology, Lund University

Lund University

The sound of ‘Signalbehandling’



‘s’ ‘i’ ‘g’ ‘n’ ‘………………………’

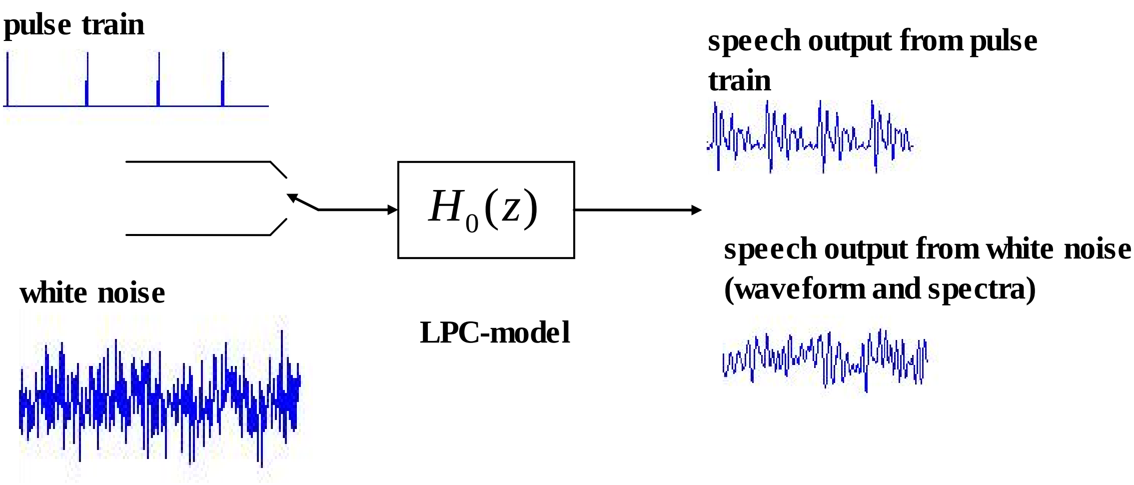


‘s’ ‘i’

noise **harmonic signal**

How can this be generated as output from a linear filter?
Determine the filter and the input signal.

LPC model of syntetic sound production

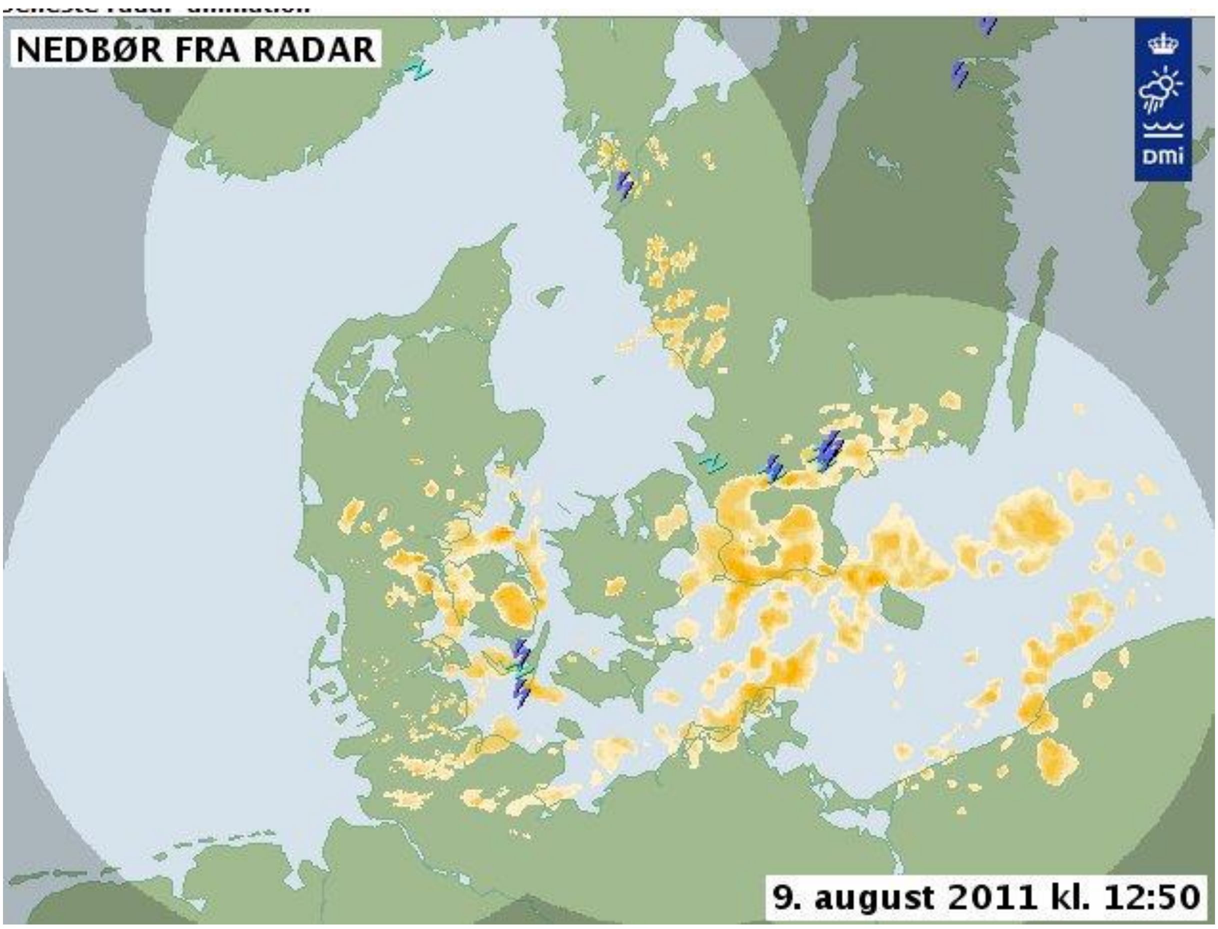


In syntetic speech production, the parameters often are updated every 5 milliseconds.

Chapter 2.	Digital signal processing	
	impulse response, convolution,	
	system function, Fourier, z-transforms	page 7-20
	Matrix description.	page 20-52
	Hints.	page 8-18, 21, 49.
Chapter 3.	Random processing, such as	
	correlation functions, correlation matrices.	
	Random variables	page 58-74
	Random Processes	page 74-119
	Hints.	page 77, 79, 80, 85,
		95, 99, 100, 101, 106
Chapter 4.	Signal models, Deterministic and Stochastic approach.	
	Padé, Prony	page 133-154
	Shank	page 154-160
	All-pole Modeling	page 160,165
	Linear prediction	page 165-174
		4.5 not included
	4.6	page 178-188
	4.7 Stochastic Models	page 188-200
	Hints.	page 130, 135, 138, 147, 148,149
		195, 195
Chapter 5.	Levinson-Durbin recursion.	page 215-225, 233-241
		page 242 – 276 not included
	Hints.	Table 5.1 – 5.4, figure 5.10
Chapter 6.	Latttice FIR and IIR filters,	
	only 6.2 and 6.4.1, 6.4.3	page 289-293, 297, 298, 304-307
	6.5	page 308-324
Chapter 7.	Optimal filters. Linear prediction.	
	Wiener filters. Specially FIR filters.	
	FIR- Wiener filter	page 335-345
	IIR- Wiener filter	page 353-371
	Kalman filters	page 371-379
	Hints.	page 337-339, 354, 355, 358-363, 370
Chapter 8.	Spectrum estimation.	
	Nonparametric methods	page 393-399, 408-425
	8.3 (8.5 see chap 4) , 8.6	page 426-429, 451-472
	Hints.	page 394, 408

Digital Signal Processing application

Radar

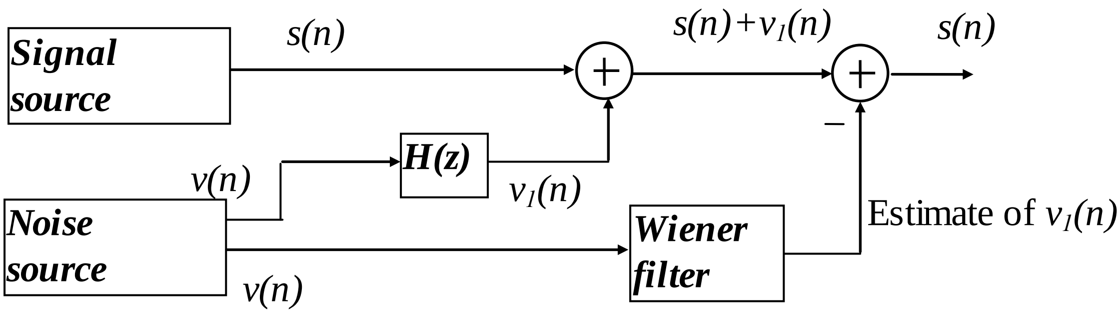


An application from the text book

Noise cancellation (chapter 7, page 349)

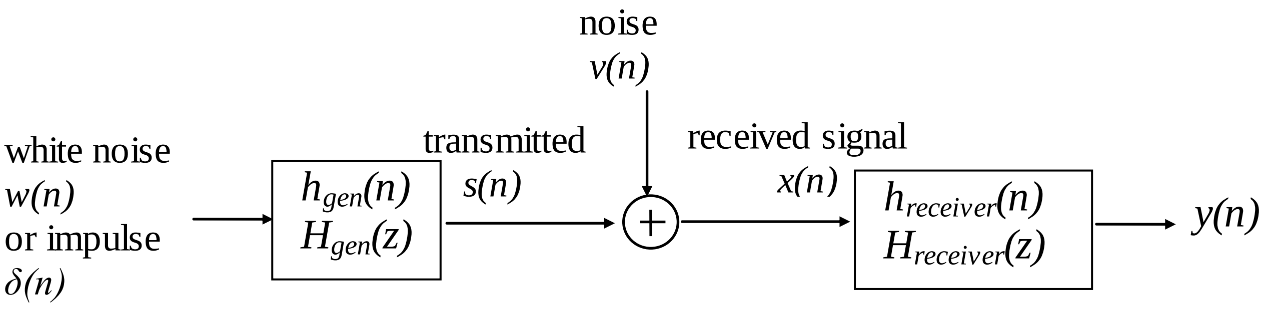
A signal is disturbed by additive noise $v_1(n)$.

Try to measure the noise $v(n)$ from the source and estimate the noise $v_1(n)$ added to the signal. Then subtract the noise $v_1(n)$ from the received signal.



Optimal signal processing in Hay's book

Chapter 2: **Brief review of digital signal processing.**
Chapter 3: **Brief review of random signals.**



Estimate $H_{gen}(z)$ from properties of $s(n)$

Determine $H_{receiver}(z)$

The filters $H_{gen}(z)$ and $H_{receiver}(z)$ are of type

FIR
IIR
all-pole IIR

Chapter 4, 5 and 6: **Make a model $H_{gen}(z)$ from the properties of $s(n)$.**

Chapter 7: **Determine $H_{receiver}(z)$.**

Chapter 8: **Estimation of spectra.**

Chapter 2 Digital Signal Processing

Difference equation

$$y(n) = -\sum_{k=1}^{p} a(k)\ y(n-k) + \sum_{k=0}^{q} b(k) x(n-k)$$

MATLAB: A=[1 0.5 0.5]; B=[1 1]; y=filter(B,A,x);

Convolution

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)\, x(n-k)$$

impulse: $\delta(n) = [0\ 0\ 0\ \underset{\uparrow}{1}\ 0\ 0\ 0]$

unit step: $u(n) = [0\ 0\ 0\ \underset{\uparrow}{1}\ 1\ 1\ 1 \ldots]$

System function

$$H(z) = \frac{B(z)}{A(z)}$$

Frequency function

$$H(e^{j\omega}) = \frac{B(e^{j\omega})}{A(e^{j\omega})}$$

FIR, IIR filters

FIR: Circuit with impulse response with finite length

Example

$$y(n) = x(n) + x(n-1), \quad h(n) = \delta(n) + \delta(n-1)$$

IIR: Circuit with impulse response with infinite length

Example

$$y(n) = 0.5\,y(n-1) + x(n), \quad h(n) = 0.5^n\,u(n)$$

All-pole IIR-filters

IIR-filters with poles only (all zeroes in origin, *B(z)*=constant)

Example

$$H(z) = \frac{1}{1-0.5\,z^{-1}}$$

Solvning the convolution sum.

$$y(n) = \sum_{k=-\infty}^{\infty} h(k)\,x(n-k) = \sum_{k=-\infty}^{\infty} x(k)\,h(n-k)$$

$$y(n) = \cdot\cdot\cdot h(0)\,x(n) + h(1)\,x(n-1) + h(2)\,x(n-2)\ \cdot\cdot\cdot$$

Example $x(n) = [\underset{\uparrow}{1}\ 2\ 3\ 4],\quad h(n) = [\underset{\uparrow}{4}\ 2\ 2]$

Method A: Vector notation

$$y(n) = \left[h(0)\,h(1)...h(N-1)\right]^T \begin{bmatrix} x(n) \\ x(n-1) \\ . \\ . \\ x(n-N+1) \end{bmatrix} = \underline{h}^T\ \underline{x}(n)$$

Method B: Graphical solution

Write

$x(k):\qquad \underset{\uparrow}{1}\ 2\ 3\ 4$

$h(0-k):\quad 2\ 2\ \underset{\uparrow}{4} \qquad\qquad y(0) = 4\cdot 1 = 4$

$h(1-k):\quad 2\ \underset{\uparrow}{2}\ 4 \qquad\qquad y(1) = 2\cdot 1 + 4\cdot 2 = 10$

Gives the output $y(n) = [\underset{\uparrow}{4}\ 10\ 18\ 26\ 14\ 8]$

MATLAB: x=[1 2 3 4]; h=[4 2 2]; y=conv(x,h)

Method C: Convolution matrix

Use matrix notations

$$x(n) = [\underset{\uparrow}{1}\ 2\ 3\ 4],\quad h(n) = [\underset{\uparrow}{4}\ 2\ 2]$$

$$\begin{bmatrix} x(0) & 0 & 0 \\ x(1) & x(0) & 0 \\ x(2) & x(1) & x(0) \\ x(3) & x(2) & x(1) \\ 0 & x(3) & x(2) \\ 0 & 0 & x(3) \end{bmatrix} \cdot \begin{bmatrix} h(0) \\ h(1) \\ h(2) \end{bmatrix} = \begin{bmatrix} y(0) \\ y(1) \\ y(2) \\ y(3) \\ y(4) \\ y(5) \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 \\ 2 & 1 & 0 \\ 3 & 2 & 1 \\ 4 & 3 & 2 \\ 0 & 4 & 3 \\ 0 & 0 & 4 \end{bmatrix} \cdot \begin{bmatrix} 4 \\ 2 \\ 2 \end{bmatrix} = \begin{bmatrix} 4 \\ 10 \\ 18 \\ 26 \\ 14 \\ 8 \end{bmatrix} \qquad X\,h = y$$

In Matlab: **x=[1 2 3 4]’; X=convmtx(x,3)**
h=[4 2 2]', y=X*h

(In signal processing, all vectors are column vectors)

Properties of matrices

The square matrix $A \ (n \times n)$ **is:**

symmetrical if $A = A^T$

Hermitian if $A = (A^T)^* = A^H$

invertable if $AA^{-1} = I$

Toeplitz if all diagonals are identical

$$A = \begin{bmatrix} 3 & 4 & 5 \\ 2 & 3 & 4 \\ 1 & 2 & 3 \end{bmatrix}$$

Hermitian (symmetrical) Toeplitz if

$$A = \begin{bmatrix} 3 & 2 & 1 \\ 2 & 3 & 2 \\ 1 & 2 & 3 \end{bmatrix} \qquad A = Toep[3,2,1]$$

orthogonal if $A^T A \ = I$

Linear equation (page 31-34)

$$A \ is \ a \ \ [n * m] \ \ matrix$$

$$Ax = b$$

gives

$$x = A^{-1}b \quad if \quad n = m, (A \ \ invertable)$$

$$x = (A^H A)^{-1} A^H \, b \quad if \quad n > m$$

(overdetermined, more equations than variables.) Described more in chapter 4

$$x = A^H (A A^H)^{-1} b \quad if \quad n < m$$

(underdetermined, less equations than variables)

Eigenvalue:

$$Av = \lambda v, \quad (A - \lambda I) = 0$$

$$\lambda \ \ eigenvalues, \ \ v \ eigenvectors$$

$$A = V \Lambda V^{-1} \ \ with \ eigenvectors \ in \, the \ columns \ of \, V,$$
$$eigenvalues \ in \ the \ diagonal \, of \ \Lambda$$

Optimisation (minimizing): (page 49)

If z real: $f(z) = z^2$

$$\frac{d}{dz} f(z) = \frac{d}{dz} z^2 = 2z; \qquad \frac{d}{dz} z^2 = 0$$

$$gives \;\; z = 0 \;\; as \;\; \min imum;$$

If z is complex: $f(z) = |z|^2 = z^* z$

(z^* *is the conjugate of* z)

Derivate with respect to z^* *or* z **separately while treating the other as a constant.**

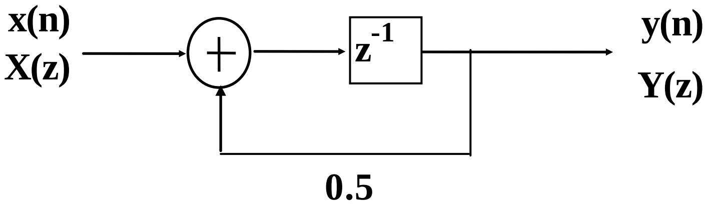
$$\frac{d}{dz} | z^2 | = \frac{d}{dz} z^* z = z^*$$

$$\frac{d}{dz^*} | z^2 | = \frac{d}{dz^*} z^* z = z$$

Setting this derivatives equal to zero gives the same minimum (page 49). This is used sometimes in the textbook.

Example on circuits

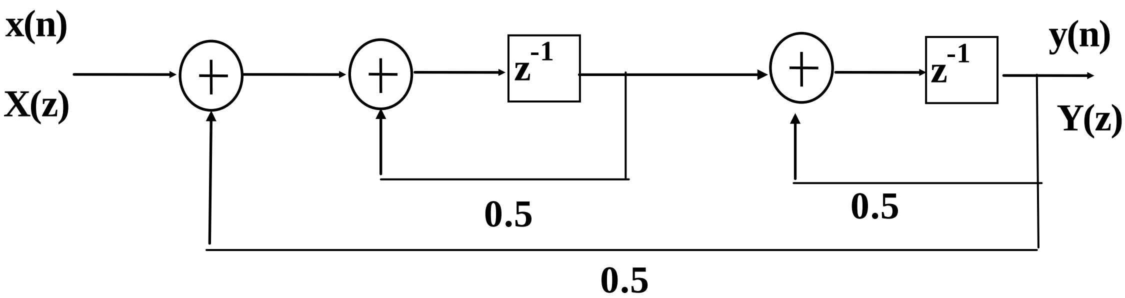
A



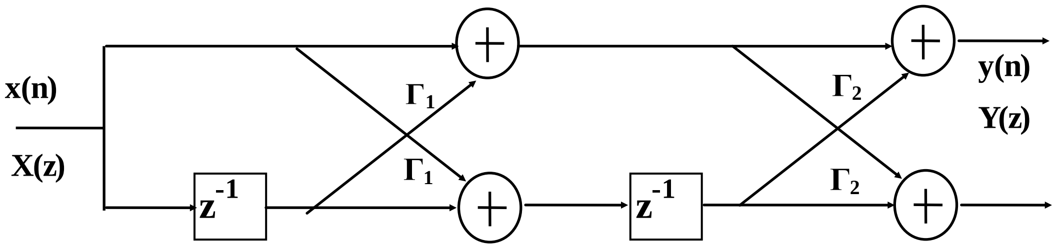
$$y(n) = 0.5\, y(n-1) + x(n-1)$$

$$Y(z) = 0.5\, z^{-1}\; Y(z) + z^{-1}\; X(z)$$

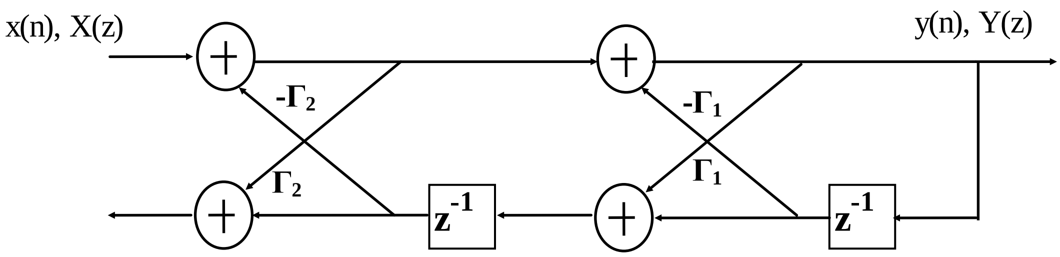
B



C Lattice filters



FIR-lattice filter



IIR-lattice filter

Correlation functions (deterministic)

Autocorrelation function

$$r_x(l) = \sum_{n=-\infty}^{\infty} x(n)\,x(n-l) \quad \left(= r_{xx}(l)\right)$$

Cross-correlation function

$$r_{yx}(l) = \sum_{n=-\infty}^{\infty} y(n)\,x(n-l)$$

$$r_x(l) = x(l) * x(-l)$$

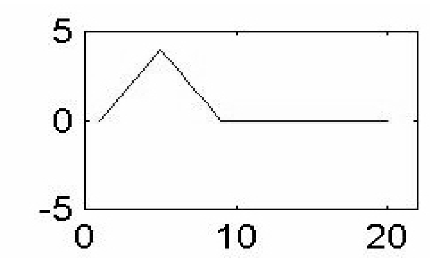
$$r_{yx}(l) = y(l) * x(-l)$$

Relation between input and output

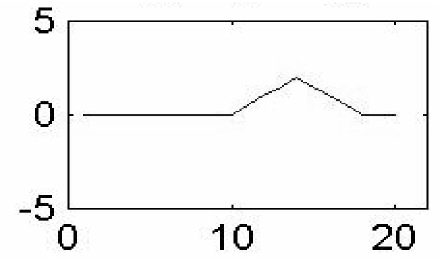
$$r_{yx}(l) = h(l) * r_x(l)$$

$$r_y(l) = r_h(l) * r_x(l)$$

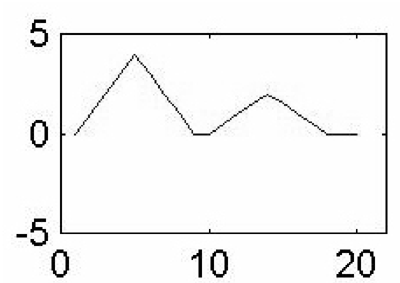
Example on correlation, echo



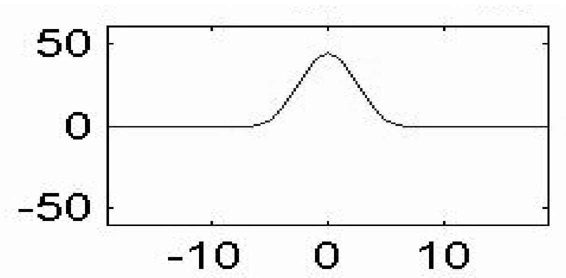
$\mathbf{x_1}$



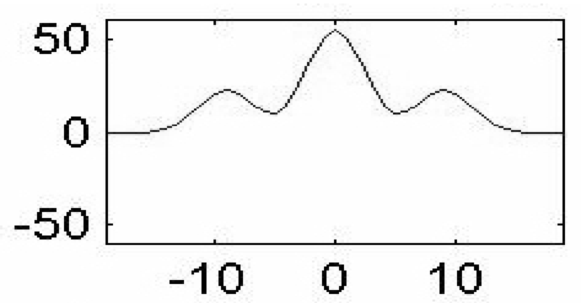
$\mathbf{x_2}$



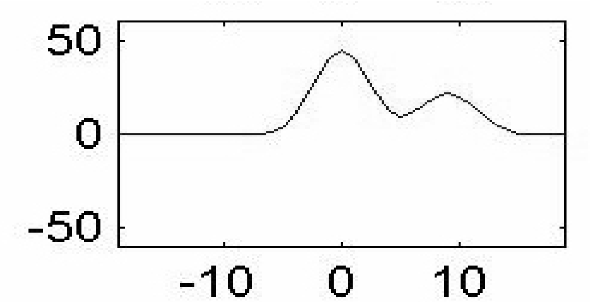
$\mathbf{y=x_1+x_2}$



$\mathbf{r_{x1}}$

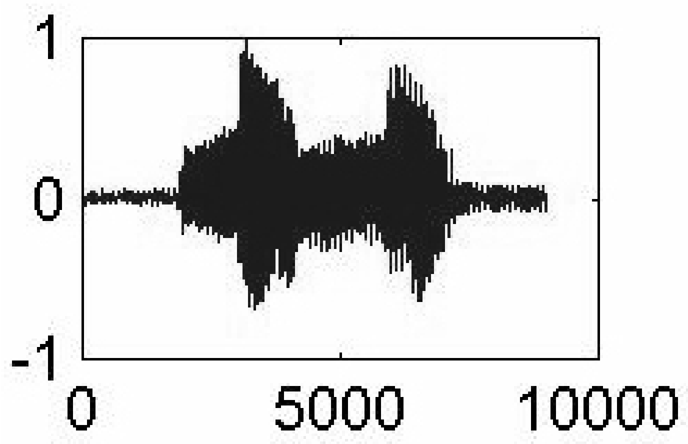


$\mathbf{r_y}$

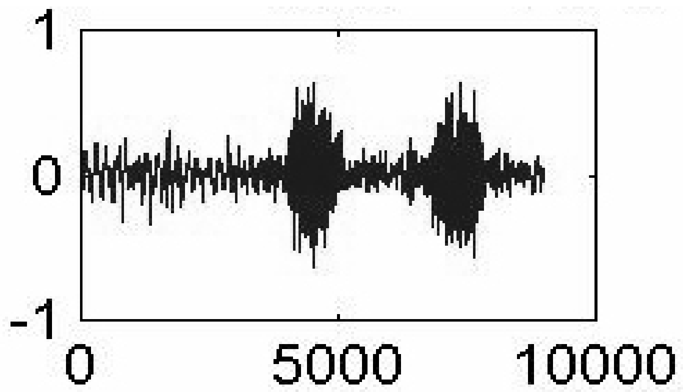


$\mathbf{r_{x1y}}$

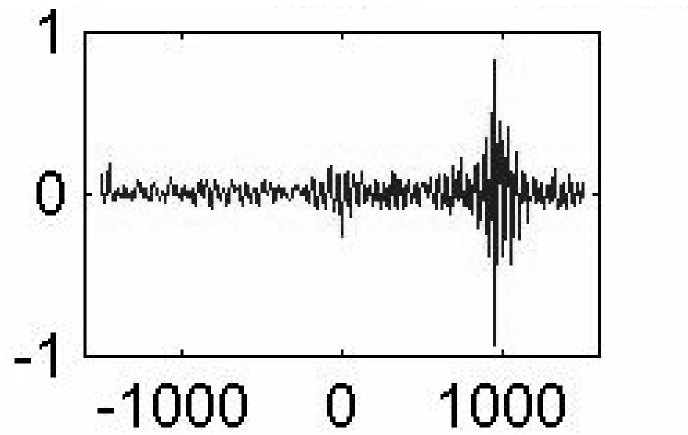
Example of correlation, delay in mobile phones (GSM)



Input signal to the GSM phone



Output signal after GSM



Crosscorrelation

In Matlab: rxy=xcorr(input,output)

Chapter 3 Discrete-Time Random Processes

Random variables (3.2 page 58-74)

Probability density function $f_X(x)$

Probability distribution function: $F_X(x)$

Expected value (mean): $m = E\{x\} = \int x \ f_X(x)\ dx$

Mean-square value: $E\{x^2\} = \int x^2 \ f_X(x)\ dx$

Variance:

$$Var[x] = \sigma_x{}^2 = E\{[x-m]^2\} = \int [x-m]^2 \ f_X(x)\ dx$$

General: $y = g(x); \quad E\{y\} = E\{g(x)\} = \int g(x) \ f_X(x)\ dx$

Relation:

$$Var[x] = E\{[x-m]^2\} = E\{X^2\} - m^2$$

Correlation. Dependency between random variables *x* and *y*

Correlation: $r_{xy} = E\{x\,y\}$

Covariance: $$c_{xy} = E\{[x - m_x][y - m_y]\}$$

Stochastic processes (3.3 page 74)

(Wide-sense stationary processes, WSS)

Example A: Sinusoids with random phase

$$x(n) = A\sin(\omega_0 n + \Phi),$$

Φ **is a random variable and**

$x(n)$ **is a random process.**

Example B: Noise (white noise, colored noise).

Example C: Speech signals.

The autocorrelation sequence and the cross-correlation sequence and their Fourier transforms are important in this course.

Autocorrelation sequence:

$$r_x(m) = E\{x(k)\, x^*(k-m)\}$$

Cross-correlation sequence.

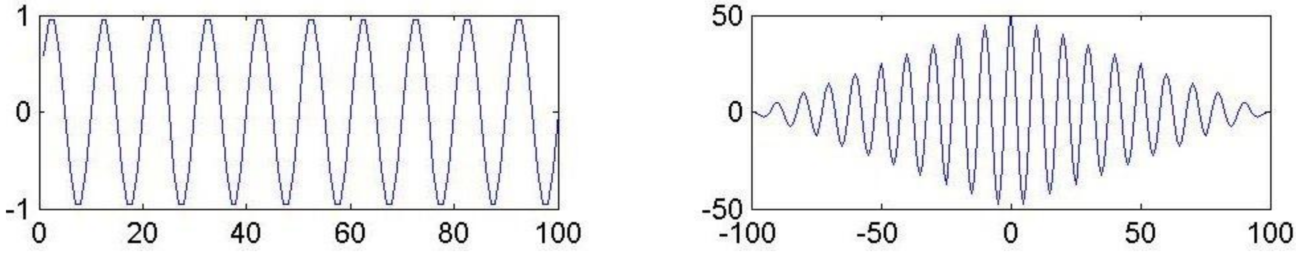
$$r_{xy}(m) = E\{x(k)\, y^*(k-m)\}$$

Estimation of the autocorrelation sequence (ergodic processes)

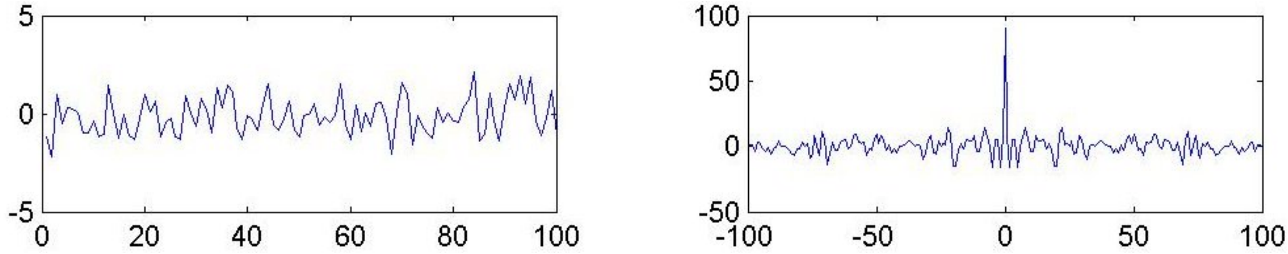
$$r_x(m) = E\{x(k)\, x(k-m)\} = \frac{1}{N} \sum_{\substack{sum\ over \\ N\, values}} x(k)x(k-m)$$

Interpreting of autocorrelation sequence:

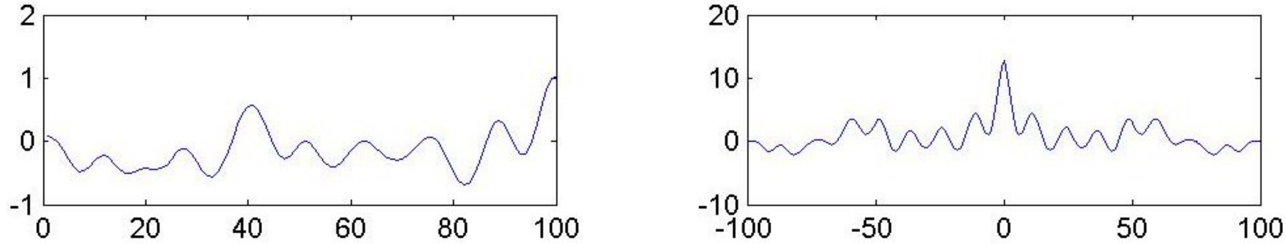
Signal | **Autocorrelation sequence**



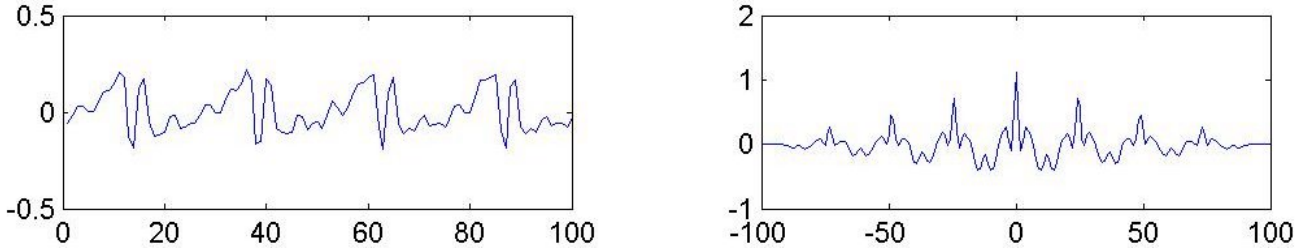
Sinusoid:



White noise.



Colored noise



Speech signal: Vowel.

Properties of autocorrelation sequence (page 83)

(Wide-sense stationary processes, WSS)

Definition:

$$r_x(k) = E\left[x(n)x^*(n-k)\right] = E\left[x^*(n)\ x(n-k)\right]^* =$$
$$= E\left[x(n-k)x^*(n)\right]^* = E\left[x(n)\ x^*(n+k)\right]^* = r_x(k)^*$$

Symmetry:

$$r_x(k) = r_x^*(-k)$$

Mean-square value:

$$r_x(0) = E[|\,x(n)|^2] \geq 0 \qquad\qquad (positive)$$

Maximum value:

$$r_x(0) \geq |\,r_x(k)\,|$$

Non-stationary processes

For signals that are not wide-sense stationary processes, (not WSS), we have to use the definitions (see chapter 4)

$$r_x(k,l) = E\{x(k)\ x^*(l)\}$$
$$r_{yx}(k,l) = E\{y(k)\ x^*(l)\}$$

Correlation matrix (WSS)

$$x = [x(0) \;\; x(1) \ldots x(N-1)]^T$$

$$R_x = E[x\,x^H] =$$

$$= \begin{bmatrix} r(0) & r_x^*(1) & r_x^*(2) & \cdots & r_x^*(p) \\ r_x(1) & r_x(0) & r_x^*(1) & \cdots & r_x^*(p-1) \\ r_x(2) & r_x(1) & r_x(0) & \cdots & r_x^*(p-2) \\ & & . & & \\ r_x(p) & r_x(p-1) & r_x(p-2) & \cdots & r_x(0) \end{bmatrix}$$

Properties of the correlation matrix

Hermitian Toeplitz
Toeplitz if real-valued process
Eigenvalues are real and non-negative

Estimate of the correlation function

$$\hat{r}_x(k) = \frac{1}{N} \sum_{n=0}^{N-1} x(n)\, x^*(n-k)$$

Estimate of the cross-correlation function

$$\hat{r}_{xy}(k) = \frac{1}{N} \sum_{n=0}^{N-1} x(n)\, y^*(n-k)$$

Power spectrum of random process (3.3.8 page 94):

(Wide-sense stationary processes, WSS)

x(n) **is a wide sense stationary random process (WSS,** ***x(n)*** **real-valued,** ***h(n)*** **real) with autocorrelation** $r_x(k)$

The Fourier transform and the z-transform are given by:

The Fourier transform of $r_x(k)$ **:**

$$P_x(e^{j\omega}) = \sum r_x(k)\ e^{-j\omega k}$$

The Z-transform of $r_x(k)$ **:**

$$P_x(z) = \sum r_x(k)\ z^{-k}$$

Properties

Symmetry (real processes)

:

$$P_x(e^{j\omega}) = P_x(e^{-j\omega})$$

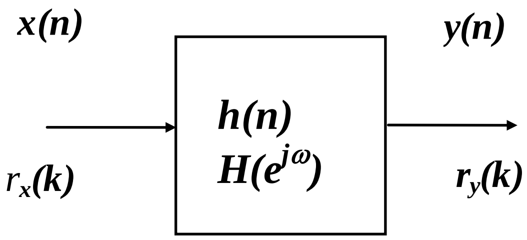
Positive:

$$P_x(e^{j\omega}) \geq 0$$

Total power:

$$r_x(0) = \frac{1}{2\pi}\int_{-\pi}^{\pi} P_x(e^{j\omega})\, d\omega$$

Filtering of random processes, (3.4 page 99, 100, 101):



Input-output relation

$$y(n) = x(n) * h(n) = \sum_{k=-\infty}^{\infty} x(k)h(n-k)$$

Autocorrelation function for the output

$$r_y(k) = E\{y(n)\,y(n-k)\} = \sum_{l=-\infty}^{\infty} \sum_{m=-\infty}^{\infty} h(l)\, r_x(m-l+k)\, h(m)$$

Cross correlation functions

$$r_{yx}(k) = E\{y(n)\,x(n-k)\} = \sum_{l=-\infty}^{\infty} h(l)\, r_x(k-l)$$

$$r_{xy}(k) = E\{x(n)\,y(n-k\} = \sum_{l=-\infty}^{\infty} h(l)\; r_x(k+l)$$

Using convolution and power spectra

$$Define \quad r_h(k) = \sum h(l)h(l+k) = h(k) * h(-k)$$

Correlation functions

$$r_y(k) = r_x(k) * h(k) * h(-k) = r_x(k) * r_h(k)$$

$$r_{yx}(k) = r_x(k) * h(k)$$

$$r_{xy}(k) = r_x(k) * h(-k)$$

Spectra

$$P_y(e^{j\omega}) = P_x(e^{j\omega}) |H(e^{j\omega})|^2$$

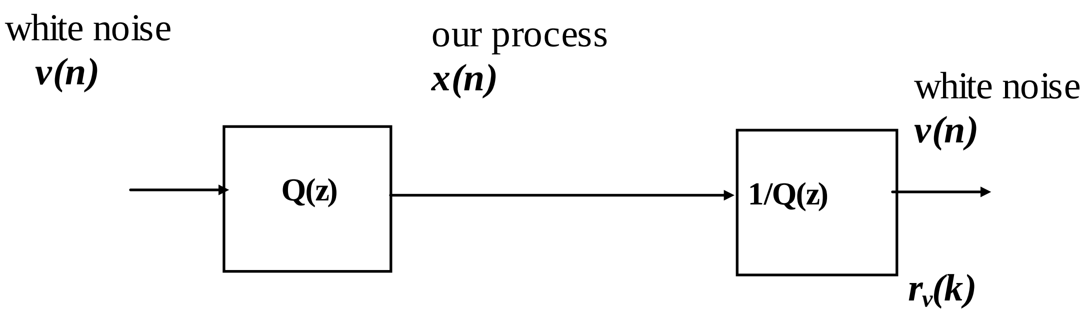
$$P_{yx}(e^{j\omega}) = P_x(e^{j\omega}) H(e^{j\omega})$$

$$P_{xy}(e^{j\omega}) = P_x(e^{j\omega}) \; H^*(e^{j\omega})$$

$$P_y(z) = P_x(z)\, H(z)\, H(\frac{1}{z})$$

Spectral factorization (3.5 page 104)

x(n) **is a WSS process with autocorrelation** $r_x(k)$**. We assume that the process are generated from white noise v(n) filtered in a filter with system function Q(z), Then,** ***v(n)*** **is called the innovation process of the process x(n).**



$$r_v(k) = \sigma_0^2\,\delta(k) \qquad r_x(k)$$

$$P_v(z) = \sigma_0^2 \qquad P_x(z) = \sigma_0^2 Q(z) Q^*(1/z^*)$$

Can we find the filter ***Q(z)*** **from** ***x(n)*** **and** $r_x(k)$**?**
Is ***Q(z)*** **stable and causal?**
Is ***1/Q(z)*** **stable and causal?**