Optimal Signal Processing

Lesson 4

Chapter 6.
Lattice Filters

LTH

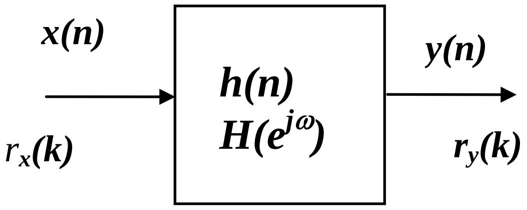
September 2013

Nedelko Grbic

(Mtrl from Bengt Mandersson)

Department of
Electrical and Information Technology, Lund University
Lund University

Chapter 3 Review of filtering random processes



Input-output relation (convolution)

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

Autocorrelation function (deterministic)

$$r_x(k) = \sum_{n} x(n)\ x(n-k) = (r_{xx}(k))$$

Autocorrelation function (random processes)

$$r_x(k) = E\{x(n)\ x(n-k)\} = (r_{xx}(k))$$

Cross correlation function (random processes)

$$r_{yx}(k) = E\{y(n)\ x(n-k)\}$$

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

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

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

Correlation functions

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

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

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

Spectra

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

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

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

$$P_y(z) = P_x(z)\;H(z)\,H(z^{-1})$$

$$P_{yx}(z) = P_x(z)\;H(z)$$

$$P_{xy}(z) = P_x(z)\;H(z^{-1})$$

Chapter 4 Review of the All-pole model.

The difference equation for the input $\delta(n)$ is (deterministic)

$$x(n) + \sum_{k=1}^{p} a_p(k)\; x(n-k) = b(0)\delta(n)$$

and the system function

$$H(z) = \frac{b(0)}{1 + a_p(1)\,z^{-1} + a_p(2)\,z^{-2} + \cdots + a_p(p)\,z^{-p}} = \frac{b(0)}{1 + \sum_{k=1}^{p} a_p(k)\,z^{-k}}$$

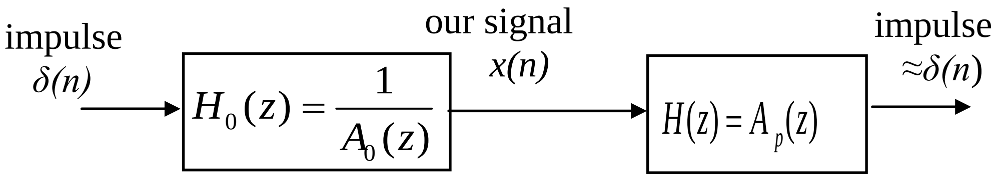
The output should be zero for all $n \neq 0$. We define an error

$$e(n) = x(n) + \sum_{k=1}^{p} a_p(k)\; x(n-k)$$

and we minimize

$$\varepsilon_p = \sum_{n=0}^{\infty} |\, e(n)\,|^2$$

This can be described by the following figure (*b(0)=1*).



$$A_p(z) = 1 + \sum_{k=1}^{p} a_p(k)\,z^{-k}$$

The filter $A_p(z)$ is called the predicting error filter (PEF).

We use a least squares solution to solve the problem.

Take the derivative (for simplicity, we assume real valued signals).

$$\frac{\partial \varepsilon_p}{\partial a_p(k)} = \frac{\partial}{\partial a_p(k)} \sum_{n=0}^{\infty} | e(n) |^2 = 2\sum_{n=0}^{\infty} e(n) \frac{\partial}{\partial a_p(k)} e(n) =$$

$$= 2\sum_{n=0}^{\infty} e(n) \frac{\partial}{\partial a_p(k)} [x(n) + \sum_{l=1}^{p} a_p(l)\ x(n-l)\ ] =$$

$$= 2\underbrace{\sum_{n=0}^{\infty} e(n)x(n-k)}_{\substack{e(n)\ and\ given\ data \\ orthogonal}} = 0 \qquad k = 1, 2, .., p$$

Then $$\sum_{n=0}^{\infty} [x(n) + \sum_{l=1}^{p} a_p(l)\ x(n-l)\ ] x(n-k) = 0$$

With

$$r_x(k) = \sum_{n=0}^{\infty} x(n)\, x(n-k)$$

we get the result

$$r_x(k) + \sum_{l=1}^{p} a_p(l) \underbrace{r_x(l-k)}_{r_x(k-l)} = 0$$

or rewritten

$$\sum_{l=1}^{p} a_p(l)\ r_x(k-l) = -r_x(k) \qquad k = 1, ..., p$$

This equation is called the normal equation or the Yule-Walker equation.

In matrix form

$$\begin{bmatrix} r_x(0) & r_x(1) & r_x(2) & \dots & r_x(p-1) \\ r_x(1) & r_x(0) & r_x(1) & \dots & r_x(p-2) \\ r_x(2) & r_x(1) & r_x(0) & \dots & r_x(p-3) \\ \\ r_x(p-1) & r_x(p-2) & r_x(p-3) & \dots & r_x(0) \\ \end{bmatrix} \cdot \begin{bmatrix} a_p(1) \\ a_p(2) \\ a_p(3) \\ \\ a_p(p) \end{bmatrix} = - \begin{bmatrix} r_x(1) \\ r_x(2) \\ r_x(3) \\ \\ r_x(p) \end{bmatrix}$$

Orthogonality principle.

We can derive the filter in a slightly different way. Writing

$$\varepsilon_p = \sum_{n=0}^{\infty} |e(n)|^2 = \sum_{n=0}^{\infty} e(n)e(n) = \sum_{n=0}^{\infty} e(n)[x(n) + \sum_{k=1}^{p} a_p(k)x(n-k)] =$$

$$= \underbrace{\sum_{n=0}^{\infty} e(n)\,x(n)}_{\varepsilon_{p,\min}} + \sum_{k=1}^{p} a_p(k) \underbrace{\sum_{n=0}^{\infty} e(n)\,x(n-k)}_{=0}$$

called model error

e(n) and given data must be orthogonal

The minimum error (model error) is now found as

$$\varepsilon_{p,\min} = \varepsilon_p = \sum_{n=0}^{\infty} e(n)x(n) = \sum_{n=0}^{\infty} [x(n) + \sum_{k=1}^{p} a_p(k)x(n-k)]x(n) =$$

$$= r_x(0) + \sum_{k=1}^{p} a(k)\ r_x(k)$$

$$\varepsilon_{p,\min} = \varepsilon_p = r_x(0) + \sum_{k=1}^{p} a(k)\ r_x\ (k)$$

This equation can be added to the matrix equation described above.

Then, we get (for real signals $r_x^*(k) = r_x(k)$ **)**

$$\underbrace{\begin{bmatrix} r_x(0) & r_x(1) & r_x(2) & \dots & r_x(p) \\ r_x(1) & r_x(0) & r_x(1) & \dots & r_x(p-1) \\ r_x(2) & r_x(1) & r_x(0) & \dots & r_x(p-2) \\ & & & & \\ r_x(p) & r_x(p-1) & r_x(p-2) & \dots & r_x(0) \end{bmatrix}}_{R_x} \cdot \underbrace{\begin{bmatrix} 1 \\ a_p(1) \\ a_p(2) \\ \\ a_p(p) \end{bmatrix}}_{a_p} = \cdot \underbrace{\begin{bmatrix} \varepsilon_p \\ 0 \\ 0 \\ \\ 0 \end{bmatrix}}_{\varepsilon_p u_1}$$

$$R_x \ a_p = \varepsilon_p u_1$$

This is a symmetrical Toeplitz matrix equation system and can be solved with the Levinson-Durbin algorithm described in chapter 5.

This all-pole model is often called Prediction Error Filter (PEF) or Linear Prediction Coding (LPC).

Levinson-Durbin (chapter 5) solves the normal equations iteratively.

The solution gives $a_j(k)$ **and** Γ_j **in each step j=1,..,p**

Table 5.1 ***The Levinson-Durbin Recursion***

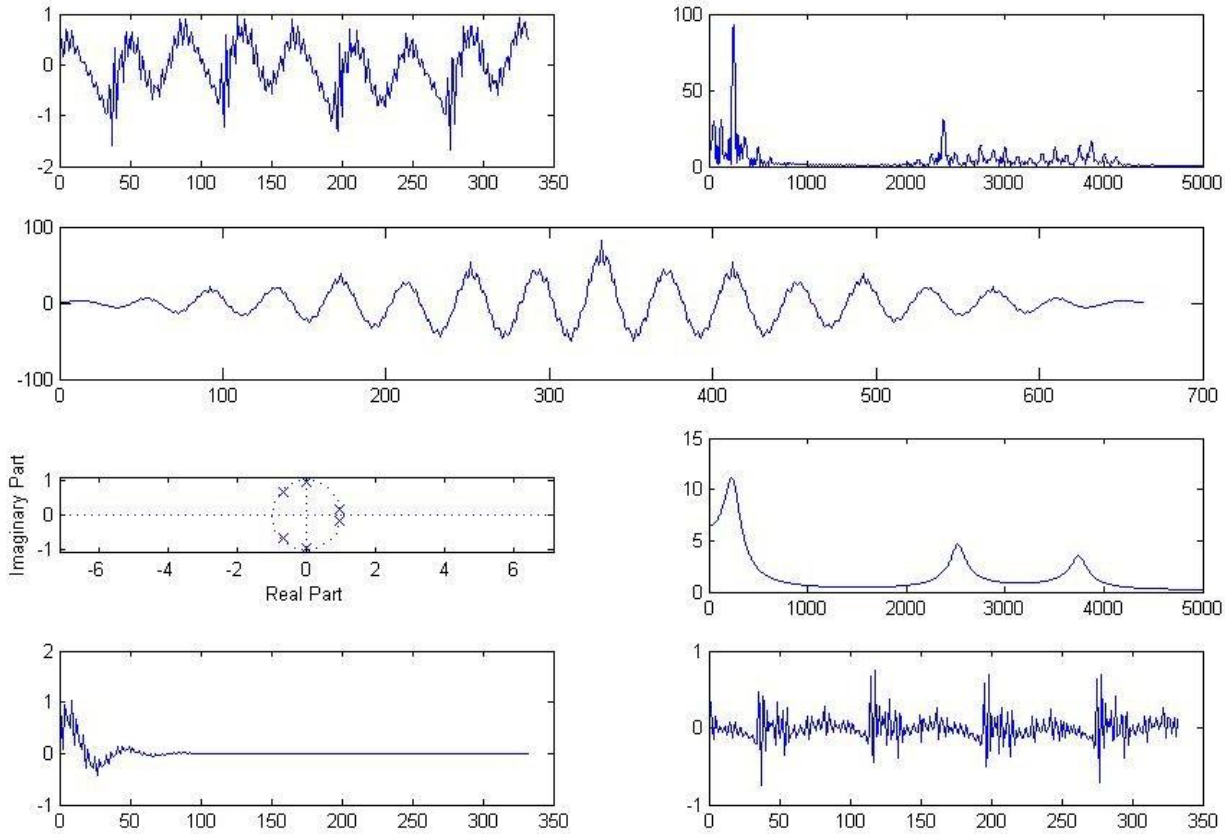
1. Initialize the recursion
   (a) $a_0(0) = 1$
   (b) $\epsilon_0 = r_x(0)$
2. For $j = 0, 1, \ldots, p-1$
   (a) $\gamma_j = r_x(j+1) + \sum_{i=1}^{j} a_j(i) r_x(j-i+1)$
   (b) $\Gamma_{j+1} = -\gamma_j / \epsilon_j$
   (c) For $i = 1, 2, \ldots, j$

   $$a_{j+1}(i) = a_j(i) + \Gamma_{j+1} a_j^*(j-i+1)$$

   (d) $a_{j+1}(j+1) = \Gamma_{j+1}$
   (e) $\epsilon_{j+1} = \epsilon_j \left[1 - |\Gamma_{j+1}|^2\right]$
3. $b(0) = \sqrt{\epsilon_p}$

Example of all-pole model of vowels
Example 1

Vowel 'i' with order p=6



Upper left: Signal, Upper right: Fourier transform (DFT) of the signal

Middle: Autocorrelation sequence of the signal
Pole-zero plot Spectrum from poles

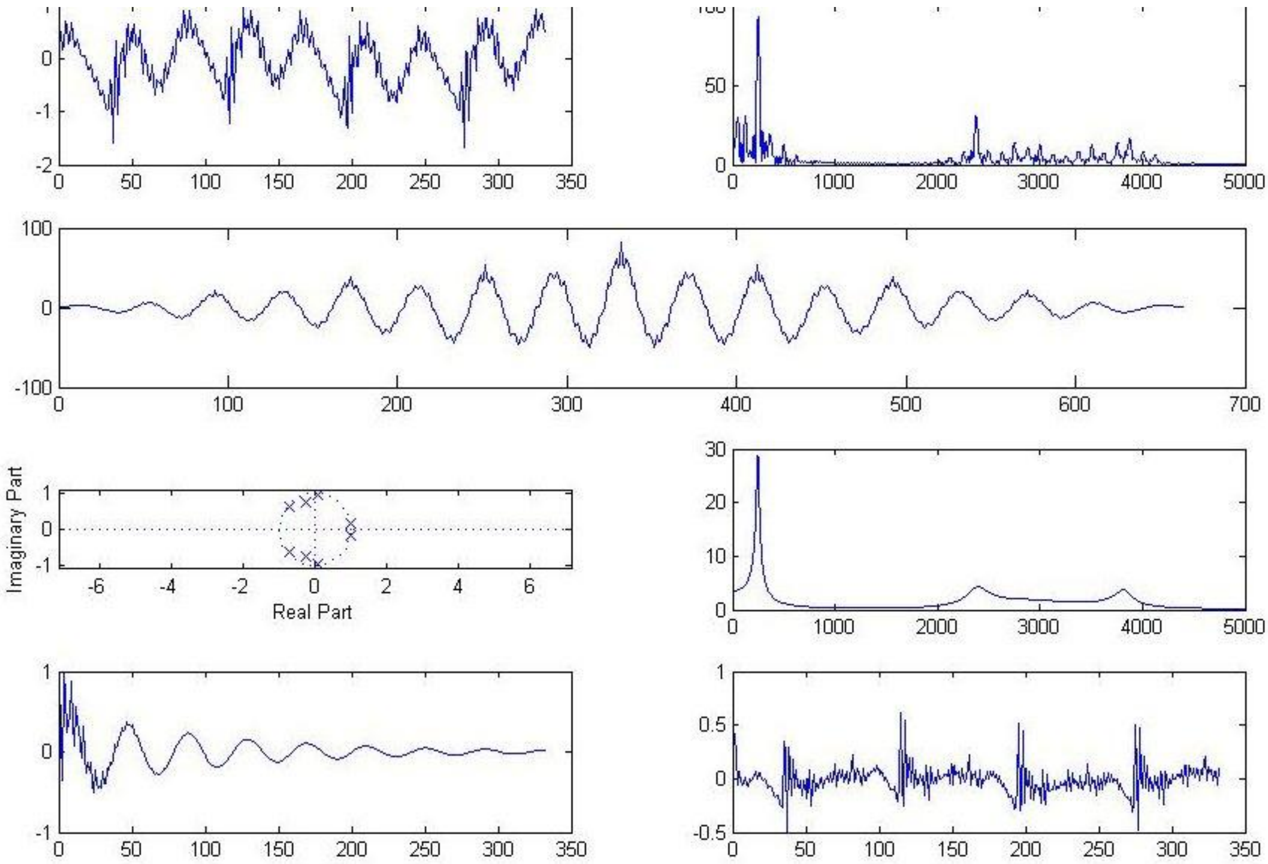
Lower left: Impulse response to $H_{IIR}(z)=1/A(z)$
Lower right: Output from $H_{FIR}(z)=A(z)$.

Coefficients $a_p(k)$ for order p=6.
a=[1.0000 -0.4912 0.1714 -1.0041 0.1397 -0.4127 0.7529]

Reflection coefficients:
Γ= [-0.7021 -0.2154 -0.5704 -0.0168 -0.0990 0.7529]

Example 2

Vowel ‘i’ with order p=8



Upper left: **Signal ,** **Upper right: Fourier transform (DFT) of the signal**

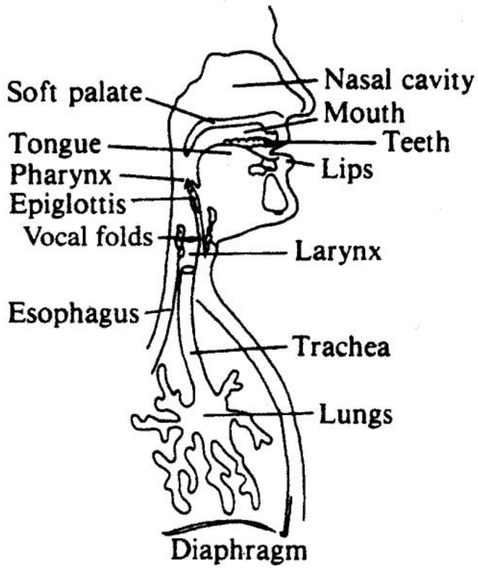
Middle: **Autocorrelation sequence of the signal**
Pole-zero plot **Spectrum from poles**

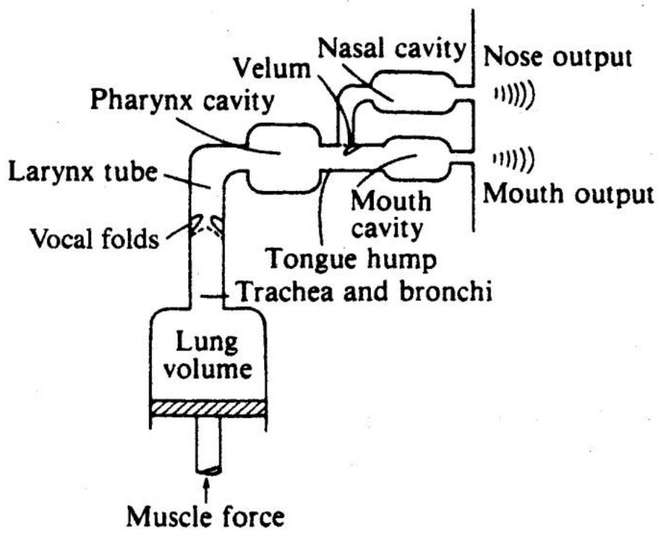
Lower left: **Impulse response to $H_{IIR}(z)=1/A(z)$**
Lower right: **Output from $H_{FIR}(z)=A(z)$.**

Coefficients $a_p(k)$ for order p=8.
a=[1.0000 -0.1191 0.3997 -1.1694 -0.2256 -0.9065 0.6446 0.1265 0.5622]

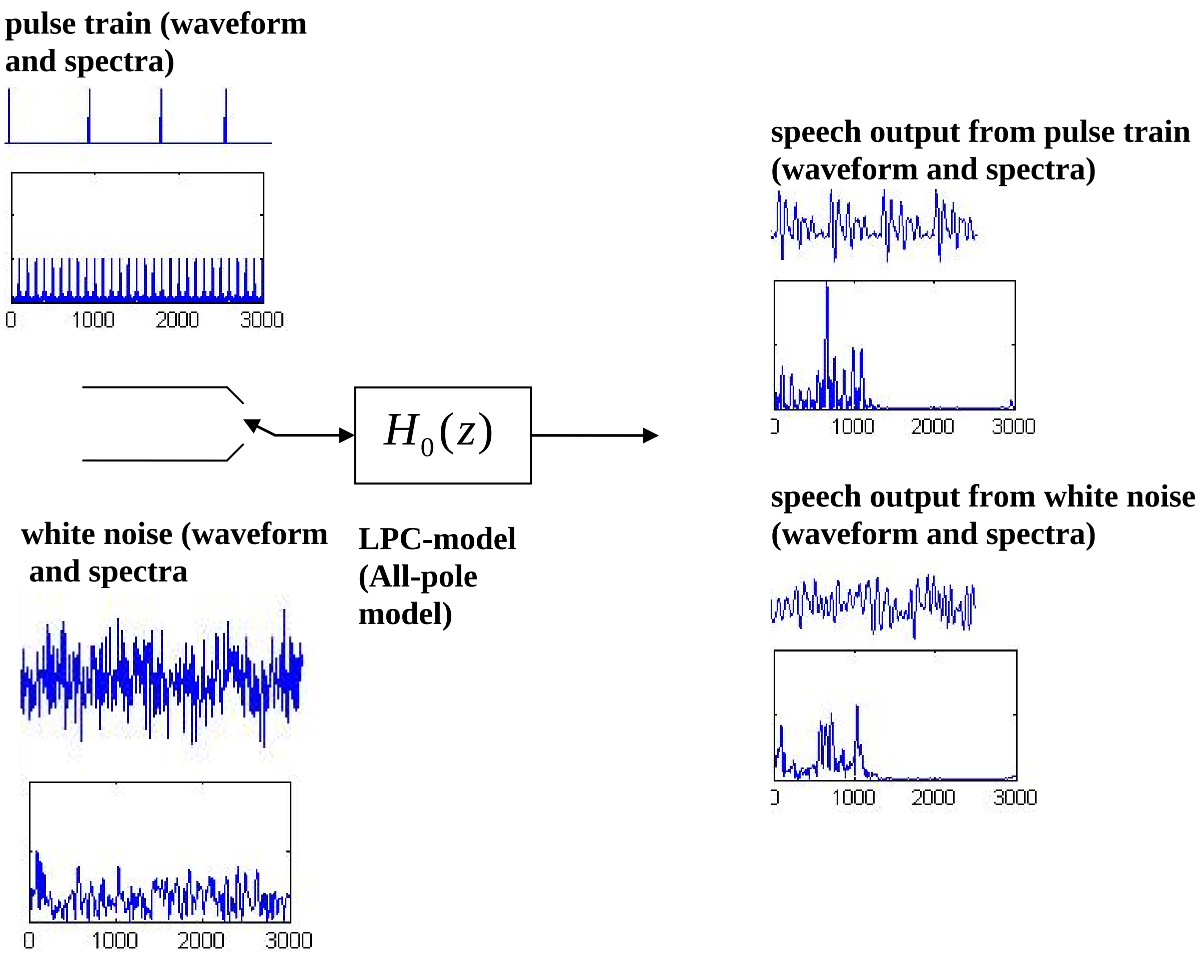
Reflection coefficients:
Γ= [-0.7021 -0.2154 -0.5704 -0.0168 -0.0990 0.7529 0.2829 0.5622]

LPC Speech encoding





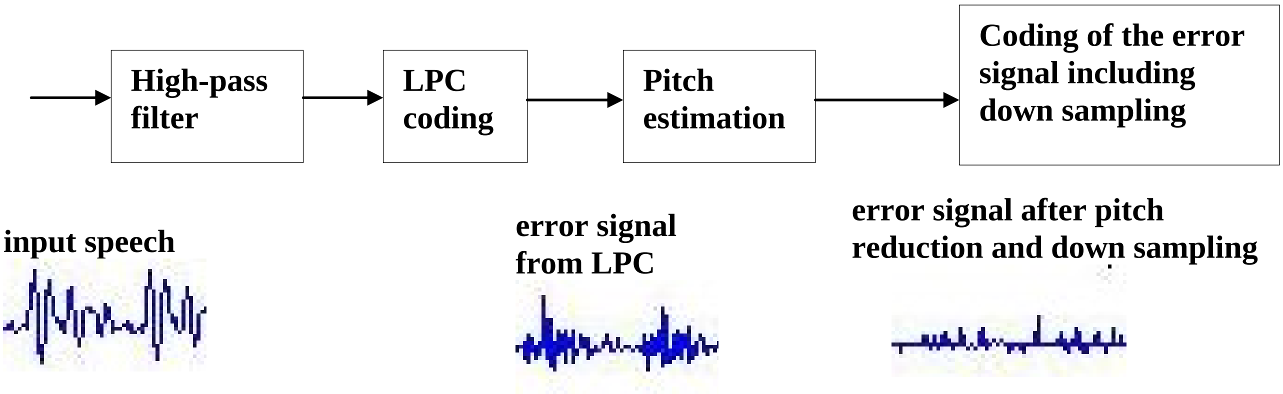
LPC model of syntetic sound production



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

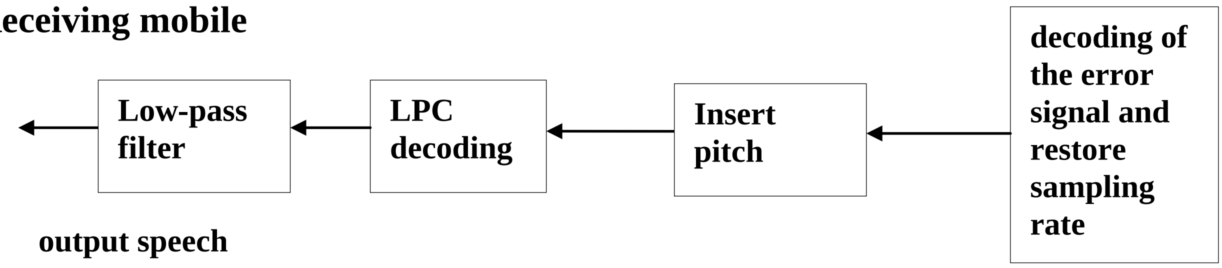
The principles of the speech coding in GSM

Transmitting mobile



Radio channel

Receiving mobile



In the laboratory work 1, we will listen to the signals after each block and we will also plot the waveforms and the spectra after each step.

Chapter 6 Lattice Filters

FIR Lattice Filter structure

Now we look at signals directly in the FIR Lattice Filter structure. We determine the coefficients directly from the signal, i.e. not determining the autocorrelation function r(k) first.

The error signal (output signal) is

$$e(n) = x(n) + \underbrace{\sum_{k=1}^{p} a_p(k)x(n-k)}_{-\hat{x}(n)} = x(n) - \hat{x}(n)$$

with

$$\hat{x}(n) = -\sum_{k=1}^{p} a_p(k)x(n-k)$$

We refer this error as the forward prediction error and use the notation

$$e^+(n) = x(n) + \underbrace{\sum_{k=1}^{p} a_p(k)x(n-k)}_{-\hat{x}(n)} = x(n) - \hat{x}(n)$$

This signal is found as the output from the upper branch in the Lattice FIR filter.
Now, we also define the signal in the lower branch as the backward prediction error.

Forward/Backward Prediction Error

From chapter 5, we have (page 224, 235, 236) (real signals)

$$a_{j+1}(i) = a_j(i) + \Gamma_{j+1}\ a_j^R(i-1)$$

$$a_{j+1}^R(i) = a_j^R(i-1) + \Gamma_{j+1}\ a_j(i)$$

and the transforms

$$A_{j+1}(z) = A_j(z) + z^{-1}\Gamma_{j+1}\ A_j^R(z)$$

$$A_{j+1}^R(z) = z^{-1}A_j^R(z) + \Gamma_{j+1}\ A_j(z)$$

The output in each step is

$$E_j^+(z) = A_j(z)\ X(z)$$

$$E_j^-(z) = A_j^R(z)\ X(z)$$

The error signal is then

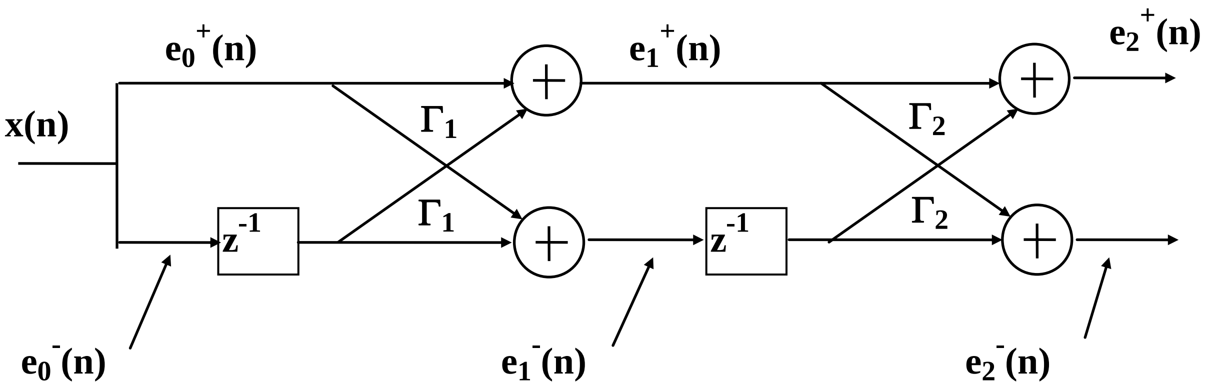
$$e_{j+1}^+(n) = e_j^+(n) + \Gamma_{j+1}\ e_j^-(n-1)$$

$$e_{j+1}^-(n) = e_j^-(n-1) + \Gamma_{j+1}\ e_j(n)$$

Second order Lattice-FIR-filter (real signals)

We can interpret the forward prediction error in the upper branch and the backward prediction error in the lower branch in the Lattice FIR filters shown below.

FIR $H(z)=A(z)$



We now will briefly present methods using the Lattice structure.

Forward Covariance Method, page 308

Backward Covariance Method, page 313

Burgs Method, page 317

Another method is the modified covariance method (page 322) using only FIR structure.

6.4 IIR Lattice filters. Some examples in the exercises

Forward Covariance method, page 308

Given: **The Lattice FIR structure (real signals).**

Task: **Determine the predictor, which minimize the forward prediction error**

$$\varepsilon_j^+ = \sum_{n=j}^{N} | e_j^+(n) |^2$$

where

$$e_j^+(n) = e_{j-1}^+(n) + \Gamma_j^+ e_{j-1}^-(n-1)$$

Solution:

Take the derivative of ε_j^+ **with respect to** Γ_j^+

$$\frac{\delta \varepsilon_j^+}{\delta \, \Gamma_j^+} = 2 \sum_{n=j}^{N} e_j^+(n) \underbrace{\frac{\delta e_j^+}{\delta \, \Gamma_j^+}}_{e_{j-1}^-(n-1)} =$$

$$= 2 \sum_{n=j}^{N} \left[e_{j-1}^+(n) + \Gamma_j^+ e_{j-1}^-(n-1) \right] e_{j-1}^-(n-1) = 0$$

This gives the solution

$$\Gamma_j^+ = - \frac{\sum_{n=j}^{N} e_{j-1}^+(n)(e_{j-1}^-(n-1))}{\sum_{n=j}^{N} | (e_{j-1}^-(n-1)) |^2}$$

Backward Covariance method (page 313-314)

Given: The Lattice FIR structure (real signals).

Task: Determine the predictor, which minimize the forward prediction error

$$\varepsilon_j^- = \sum_{n=j}^{N} | e_j^-(n) |^2$$

with

$$e_j^-(n) = e_{j-1}^-(n-1) + \Gamma_j^- e_{j-1}^+(n)$$

Solution:

Take the derivative of ε_j^- **with respect to** Γ_j^-

This gives the solution

$$\Gamma_j^- = -\frac{\sum_{n=j}^{N} e_{j-1}^+(n)(e_{j-1}^-(n-1))}{\sum_{n=j}^{N} | e_{j-1}^+(n) |^2}$$

Burgs method (page 317-319)

Given: **The Lattice FIR structure (real signals).**

Task: **Determine the predictor, which minimize the sum of the forward and backward prediction error**

$$\varepsilon_j^B = \varepsilon_j^+ + \varepsilon_j^- = \sum_{n=j}^{N} (| e_j^+(n) |^2 + | e_j^-(n) |^2)$$

with

$$e_j^+(n) = e_{j-1}^+(n) + \Gamma_j^B e_{j-1}^-(n-1)$$

$$e_j^-(n) = e_{j-1}^-(n-1) + \Gamma_j^B \quad e_{j-1}^+(n)$$

Solution:

Take the derivative of ε_j^B **with respect to** Γ_j^B

This gives the solution

$$\Gamma_j^B = -\frac{2\sum_{n=j}^{N} \ e_{j-1}^+(n)(e_{j-1}^-(n-1))}{\sum_{n=j}^{N} \ (| e_{j-1}^+(n) |^2 + | e_{j-1}^-(n-1) |^2)}$$

Burgs method step by step

Step 1:

$$\Gamma_1^B = -\frac{2\sum_{n=1}^{N} x(n)(x(n-1))}{\sum_{n=1}^{N} (|x(n)|^2 + |x(n-1)|^2)}$$

Step 2:

$$e_1^+(n) = x(n) + \Gamma_1^B x(n-1), \quad n = 1,\ldots,N$$

$$e_1^-(n) = x(n-1) + \Gamma_1^B x(n)$$

Step 3

$$\Gamma_2^B = -\frac{2\sum_{n=2}^{N} e_1^+(n)(e_1^-(n-1))}{\sum_{n=2}^{N} (|e_1^+(n)|^2 + |e_1^-(n-1)|^2)}$$

Step 4

$$e_2^+(n) = e_1^+(n) + \Gamma_2^B e_1^-(n-1), \qquad n = 2,\ldots,N$$

$$e_2^-(n) = e_1^-(n-1) + \Gamma_2^B e_1^+(n)$$

and so on

Which method is the best?

The forward and backward covariance methods can give reflection coefficients which are not always less than 1 and then, the signal model is not stable. However, they may be useful for short data sequences.

The reflection coefficients estimated using the Burg method are always less than 1 and the signal model is stable.

Burgs method modified with a window

Given: **The Lattice FIR structure (real signals).**

Task: **Determine the predictor, which minimize the sum of the forward and backward prediction error**

$$\varepsilon_j^B = \varepsilon_j^+ + \varepsilon_j^- = \sum_{n=j}^{N} w_j(n)(|e_j^+(n)|^2 + |e_j^-(n)|^2)$$

Solution:

This gives the solution (see exercise 6.10)

$$\Gamma_j^B = -\frac{2\sum_{n=j}^{N} w_{j-1}(n)e_{j-1}^+(n)(e_{j-1}^-(n-1))}{\sum_{n=j}^{N} w_{j-1}(n)(|e_{j-1}^+(n)|^2 + |e_{j-1}^-(n-1))|^2)}$$

Forward/Backward Covariance method, page 322. Gives the coefficients $a_p(k)$ direct.

Given: **The Lattice FIR structure (real signals).**

Task: **Determine the predictor, which minimize the sum of the forward and backward prediction error**

$$\varepsilon_p^M = \sum_{n=p}^{N} (|e_p^+(n)|^2 + |e_p^-(n)|^2)$$

with

$$e_p^+(n) = x(n) + \sum_{k=1}^{p} a_p(k) x(n-k)$$

$$e_p^-(n) = x(n-p) + \sum_{k=1}^{p} a_p(k) x(n-p+k)$$

Solution:

Take the derivative of ε_p^M **with respect to** $a_p^M(k)$

This gives the solution

$$\sum_{k=1}^{p} (r_x(l,k) + r_x(p-k, p-l))\, a_p(k) = -(r_x(l,0) + r_x(p, p-l))$$