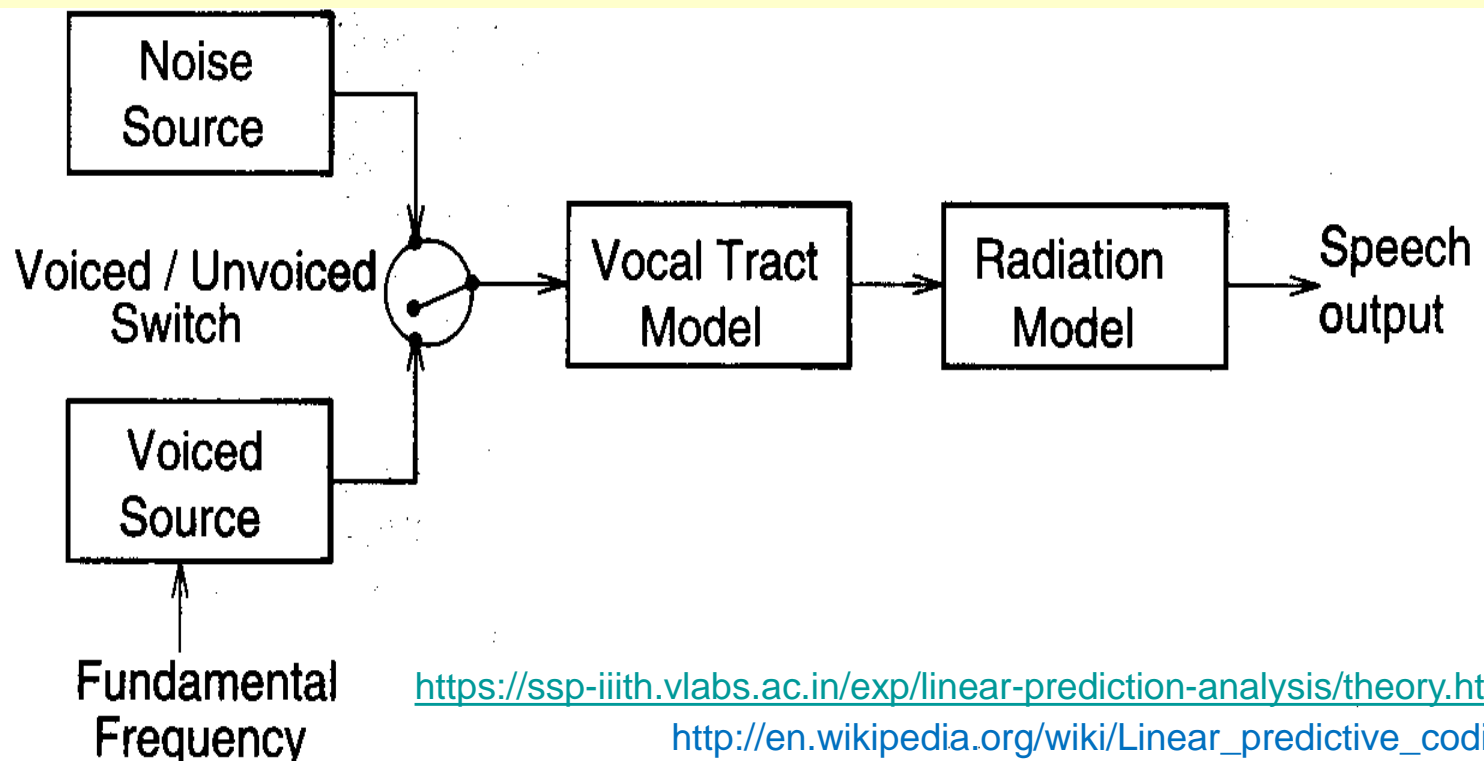


Lecture 5

Frequency domain speech analysis (3)



<https://ssp-iiith.vlabs.ac.in/exp/linear-prediction-analysis/theory.html>

http://en.wikipedia.org/wiki/Linear_predictive_coding

<https://www.ece.ucsb.edu/Faculty/Rabiner/ece259/>

Speech Signal ANALYSIS

Time domain:

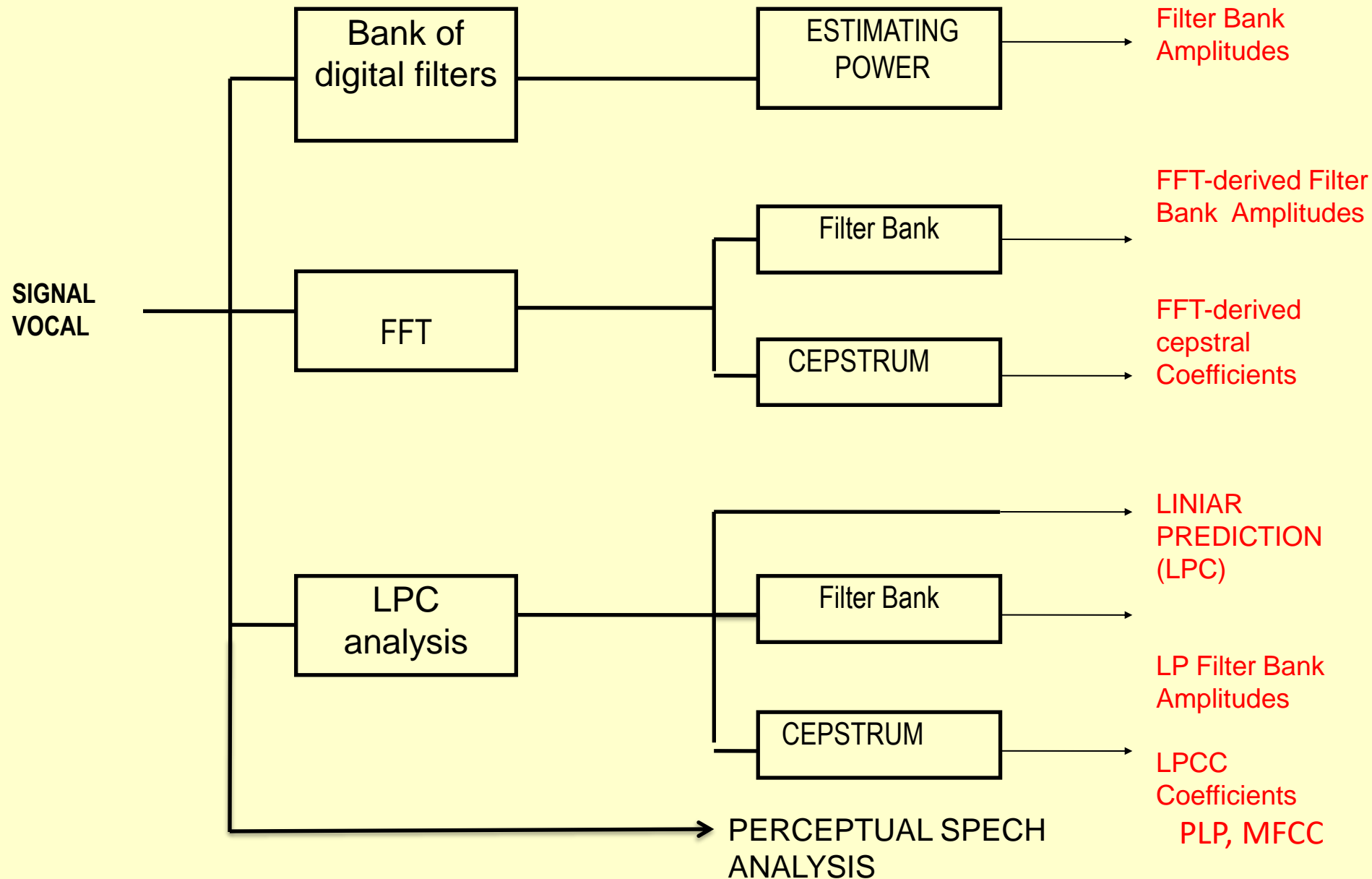
- Average and maximum amplitude
- Amplitude density
- Average energy
- TEAGER energy
- Number of zero crossings
- Fundamental frequency (F0)
- TESPAN coding

Frequency domain:

- DFT (FFT)
- LPC analysis
- Digital filter bank
- Cepstral analysis
- Perceptual analysis

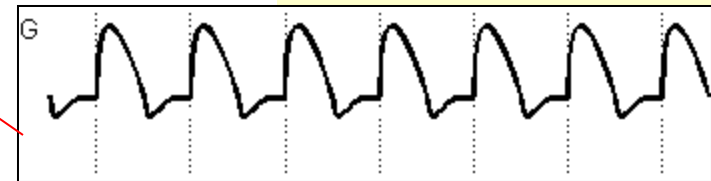
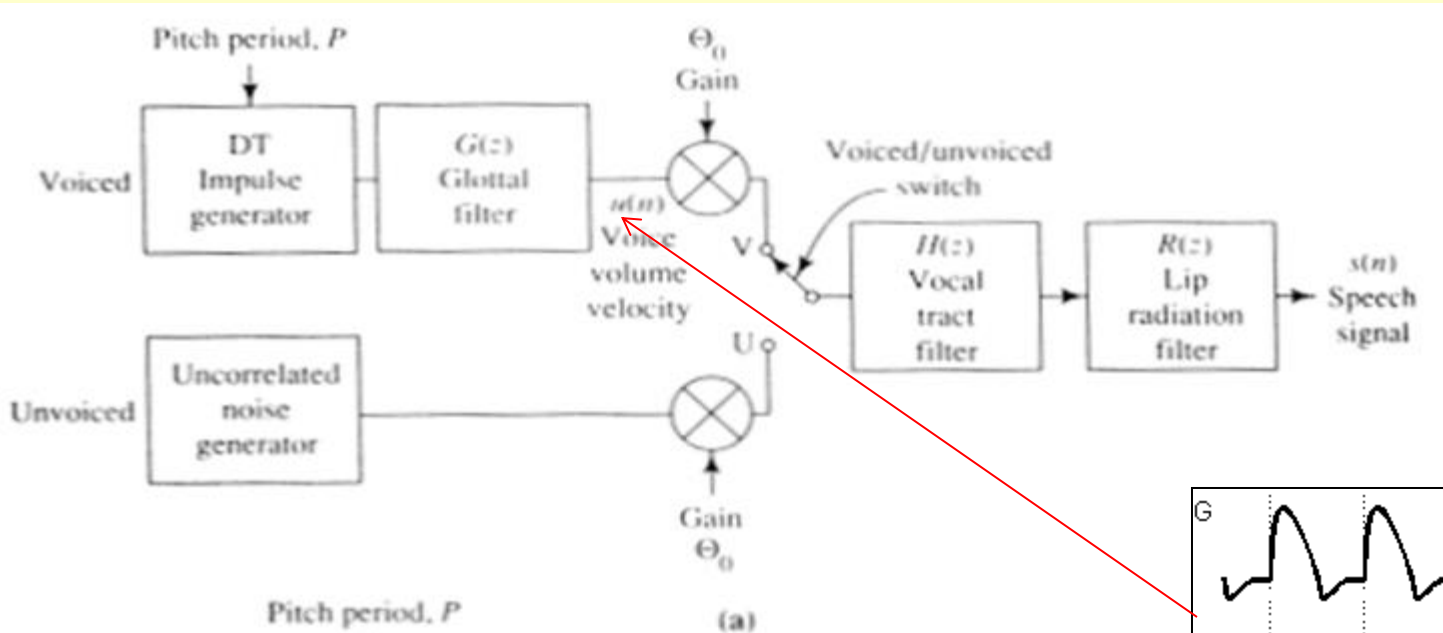
Time-frequency analysis

- Short-time Fourier transform (STFT)
- Discrete wavelet transform (Haar) (DWT)
- Continuous wavelet transform (Morlet) (CWT)
- Pseudo-Wigner distribution

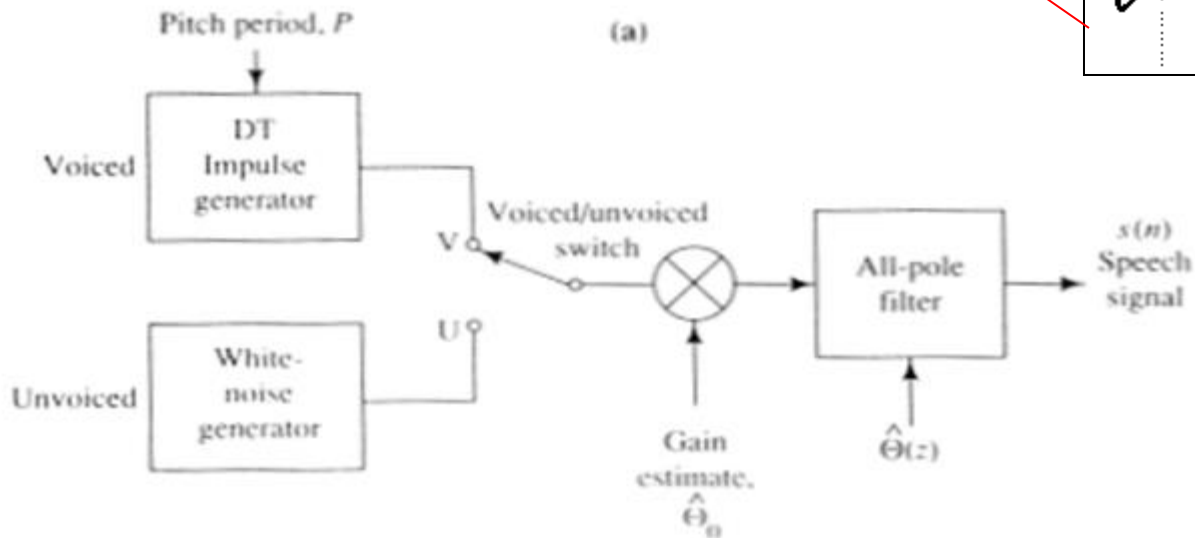


Algorithms for spectral analysis [Picone]

LPC MODELING



Glottic wave



The auto-regressive model

a) ACTUAL

b) MODEL ESTIMATED by prediction

- For a periodic signal, consider that:

$$s(n) \approx s(n - N_p)$$

,where N_p is the period of the signal, but this is not what LP does; it estimates $s(n)$ from the p most recent values ($p \ll N_p$) of $s(n)$, using a linear method

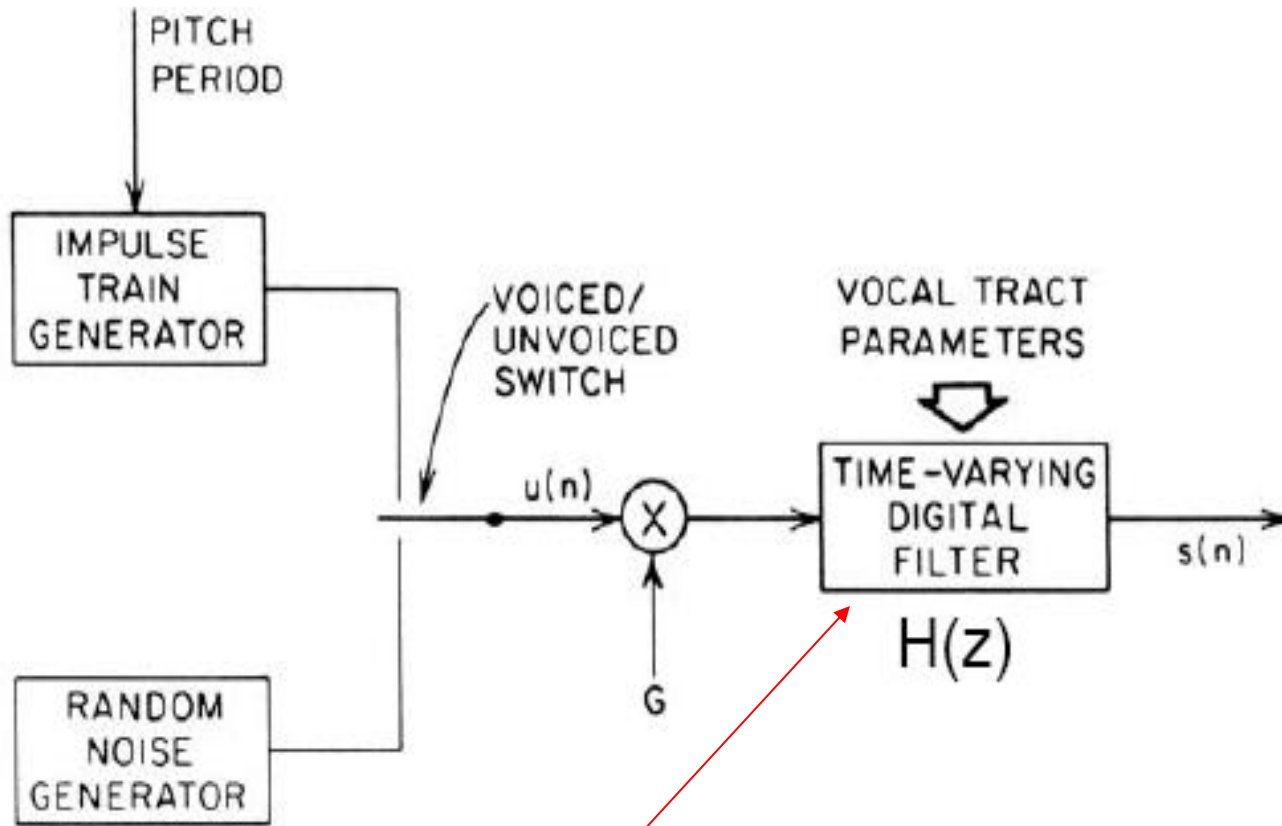
$$x(n) = \sum_{k=1}^p \alpha_k x(n - k)$$

Reasons to use the LPC model:

1. It makes a good \sim of the speech signal, especially in voiced domains
2. Analysis leads to separation of source and vocal tract
3. Low computational volume
4. Suitable for recognition/synthesis applications
5. Suitable for hardware implementation

- *Interpretations of the linear prediction model:*
 - *system identification*
 - *inverse filtering*
 - *linear prediction*
 - *spectral smoothing*

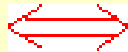
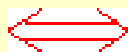
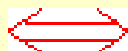
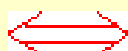
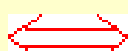

The model parameters



- The relationship between the mathematical model and the physical representation:
 - Filter coefficients(a_i)
 - Gain (G)
 - Voiced/unvoiced decision
 - Fundamental frequency (F_0)

$$s(n) = \sum_{k=1}^p a_k s(n-k) + Gu(n)$$

- The relationship between the mathematical model and the physical representation:

| | | |
|------------------------------|---|---------------------|
| Vocal tract |  | $H(z)$ (LPC Filter) |
| Air |  | $u(n)$ (samples) |
| Vocal cord vibrations |  | V (voiced) |
| Period of vibrations |  | T (pitch period) |
| Fricatives/Plosives |  | UV (unvoiced) |
| Air Volum |  | G (gain) |

- LP is based on models of speech production/synthesis and can be modeled as the output of a time-varying system excited by quasi-periodic pulses or noise.

- LP offers a robust, accurate, and reliable method for estimating the parameters of the linear system (combination of vocal tract, glottal pulses, and characteristic sound radiation).

Remark.

- The LPC10 model can be represented as a vector of the form ($p=10$):

$$\mathbf{A} = (a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, G, V/UV, T)$$

- \mathbf{A} change every 20ms (if we have an 8 kHz sampling frequency), 20 ms is equivalent to 160 samples.
- The sampled SS is divided into frames of $T = 20$ ms, so we have 50 frames/sec.
- The model assumes that

$$\mathbf{A} = (a_1, a_2, a_3, a_4, a_5, a_6, a_7, a_8, a_9, a_{10}, G, V/UV, T)$$

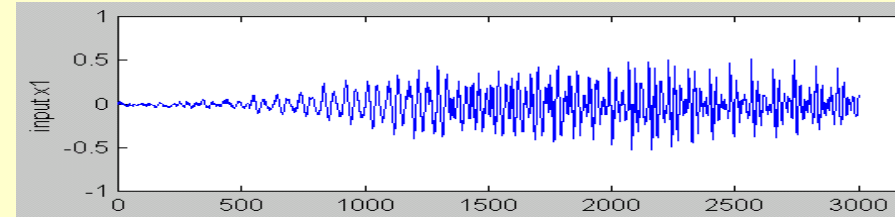
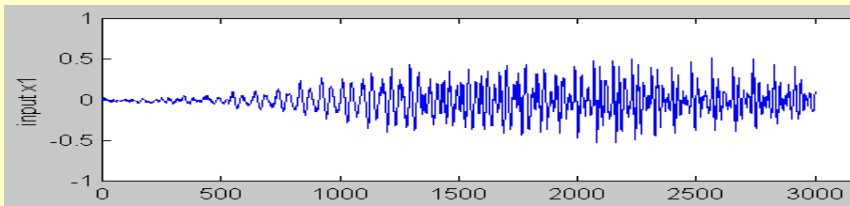
|||

$$\mathbf{S} = (s(0), s(1), \dots, s(159))$$

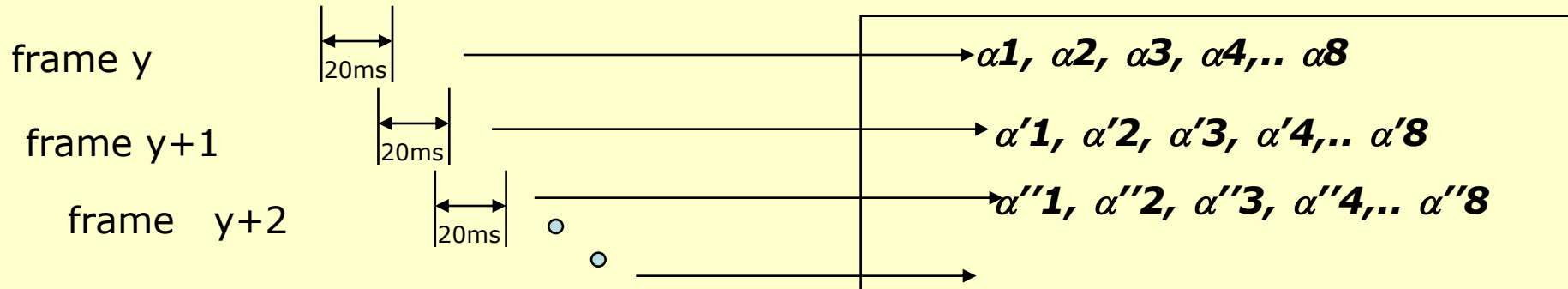
- So : 160 samples of $S(n)$ are represented by the 13 values of \mathbf{A} , from the LPC model

Example

Input Signal



You can reconstruct the signal from
LPC coef. + ...



Examples of LPC encoders

- SS sampled at 8kHz and quantized at 8bits/sample requires a transmission rate of 64 kbps
1. A representation through an LPC frame (m) of 25 ms requires: 1 bit for V_m , 3 bits for G_m , 4 bits for T_m , and 12 coefficients/8 bits each for H_m ; that's a transmission rate of 4.16 kbps.
 2. In the GSM cell phone standard, we use a modified version of LPC coding (RPE-LTP - Regular Pulse Excitation - Long Term prediction), enabling a reduction rate transmitted by phone from 104 kbps (8 kHz, 13 bits) to 13 kbps (20 ms/frame), a compression factor of 8.
 3. LPC10 [FS1015] - coding standard for voice communication at 2.4kbps.
LPC10 uses a $f_{es} = 8\text{kHz}$, 22.5ms frames, and 10 LPC coefficients. Higher compression ratios are possible using more complex algorithms.

Basic PL equations

A p-order predictor is a system of the form:

$$\tilde{s}(n) = \sum_{k=1}^p \alpha_k s(n-k) \Leftrightarrow P(z) = \sum_{k=1}^p \alpha_k z^{-k} = \frac{\tilde{S}(z)}{S(z)}$$

-The prediction error is of the form :

$$e(n) = s(n) - \tilde{s}(n) = s(n) - \sum_{k=1}^p \alpha_k s(n-k)$$

- the error is the output of a system with the transfer function :

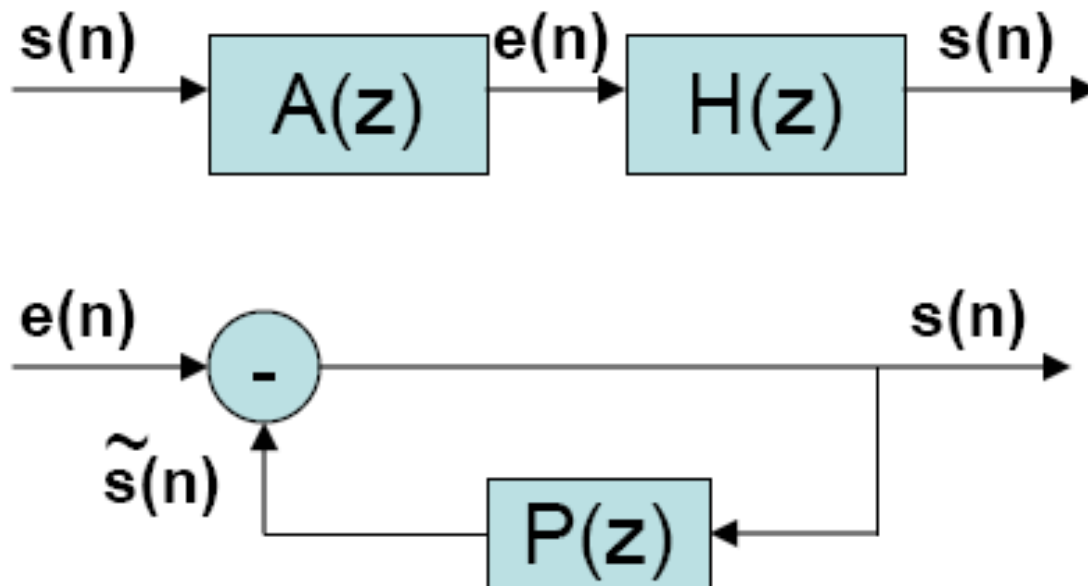
$$A(z) = \frac{E(z)}{S(z)} = 1 - \sum_{k=1}^p \alpha_k z^{-k}$$

- If the SS exactly follows the speech production model and

$$\alpha_k = a_k, 1 \leq k \leq p \Rightarrow e(n) = Gu(n)$$

and $A(z)$ is the inverse filter of $H(z)$.

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



Model estimation

- We need to determine the prediction coefficients α_k , so that the model will best estimate the temporal variation of the speech spectrum
- Estimation will be done on short frames of the SS and will minimize the root mean square error of prediction on these segments;
- The resulting coefficients α_k are assumed to be the coefficients (a_k) of the SS production model

SOLUTION for α_k

Select a segment of the SS around sample n , $s_n(m)=s(m+n)$. The root-mean-square error (RMSE) of short-term prediction E_n :

$$\begin{aligned} E_n &= \sum_m e_n^2(m) = \sum_m \left(s_n(m) - \tilde{s}_n(m) \right)^2 \\ &= \sum_m \left(s_n(m) - \sum_{k=1}^p \alpha_k s_n(m-k) \right)^2 \end{aligned}$$

-The values of α_k are determined by minimizing the prediction error E_n :

$$\frac{\partial E_n}{\partial \alpha_i} = 0, \quad i = 1, 2, \dots, p$$

- results in the set of equations :

$$-2 \sum_m s_n(m-i) \left[s_n(m) - \sum_{k=1}^p \hat{\alpha}_k s_n(m-k) \right] = 0, \quad 1 \leq i \leq p$$

$$-2 \sum_m s_n(m-i) e_n(m) = 0, \quad 1 \leq i \leq p$$

- It is defined as follows

$$\phi_n(i, k) = \sum_m s_n(m-i) s_n(m-k)$$

- The resulting system of equations (p/p), which provides an efficient solution:

$$\sum_{k=1}^p \alpha_k \phi_n(i, k) = \phi_n(i, 0), \quad i = 1, 2, \dots, p$$

- the root-mean-square error on short-term prediction E_n :

$$E_n = \sum_m s_n^2(m) - \sum_{k=1}^p \alpha_k \sum_m s_n(m) s_n(m-k)$$

- It can be written as follows:

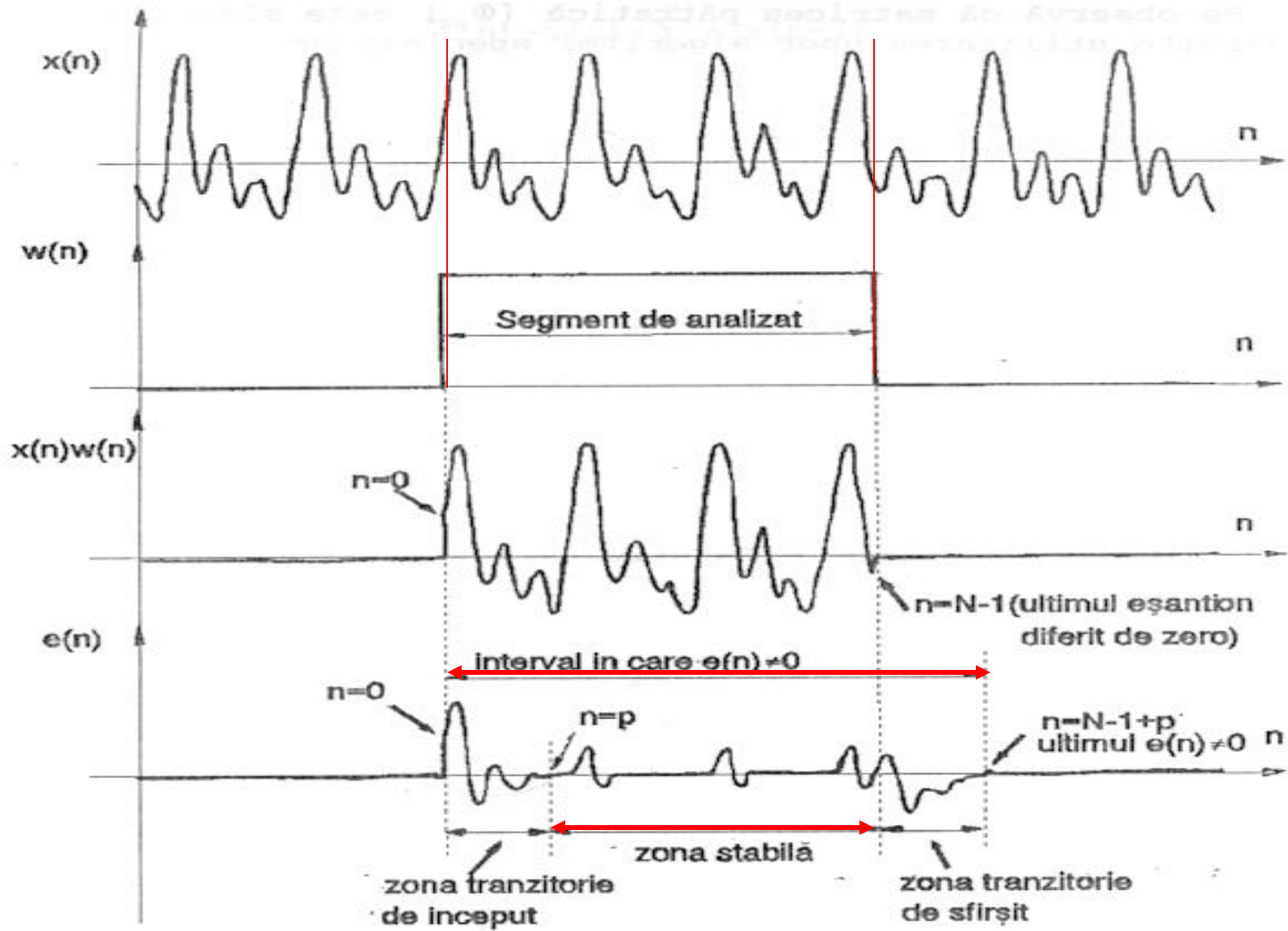
$$E_n = \phi_n(0,0) - \sum_{k=1}^p \alpha_k \phi_n(0,k)$$

is computed:

$$\phi_n(i,k) \quad \text{for} \quad 1 \leq i \leq p, 0 \leq k \leq p$$

- the matrix equation for α_k

- We specify the range of m for the calculation $\phi_n(i,k)$ and $s_n(m)$



There are 2 calculation methods:

Autocorrelation - which means that the signal $S_n(m) \neq 0, 0 < m < N-1$

Covariance - is applied to the area where the error is stable

1) **Autocorrelation** - assumes that the signal $s_n(m) \neq 0$, $0 \leq m \leq N-1$

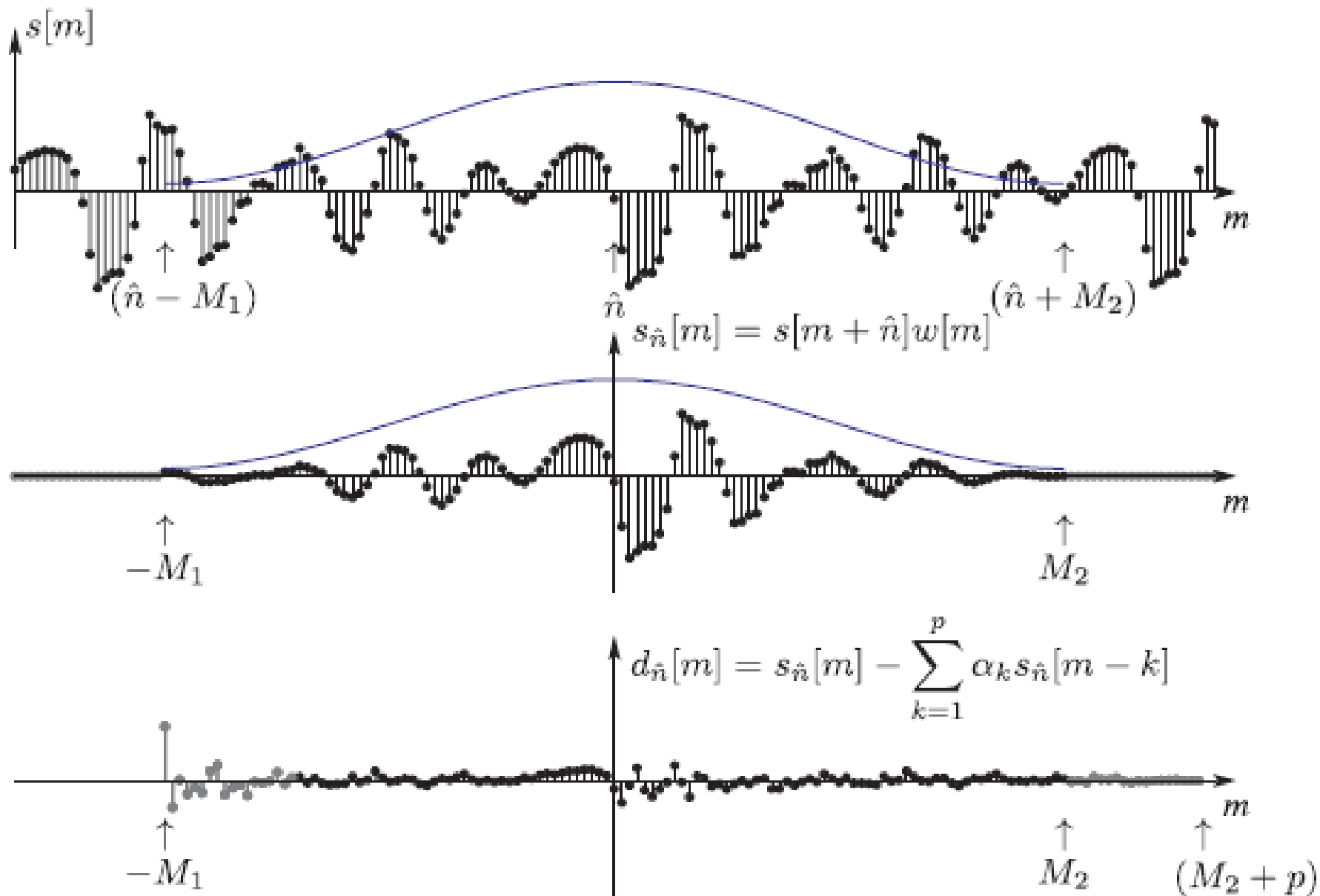
$$s_n(m) = s(m+n) w(m)$$

, where the $w(m)$ finite window of N samples

The signal $s_n(m) \neq 0$ for $0 \leq m \leq N-1$, and the error:

$$e_n(m) = s_n(m) - \sum_{k=1}^p \alpha_k s_n(m-k) \quad \text{is } \neq 0, \text{ for } 0 \leq m \leq N-1+p,$$

$$E_n = \sum_{m=-\infty}^{\infty} e_n^2(m) = \sum_{m=0}^{N-1+p} e_n^2(m)$$



$$s_n(m) = s(m+n)w(m), \quad 0 \leq m \leq N-1$$

$$e_n(m) = s_n(m) - \sum_{k=1}^p \alpha_k s_n(m-k), \quad 0 \leq m \leq N-1+p$$

It is calculated: $\phi_n(i, k)$, where $s_n(m) \neq 0$:

$$\phi_n(i, k) = \sum_{m=0}^{N-1+p} s_n(m-i) s_n(m-k), \quad 1 \leq i \leq p, 0 \leq k \leq p$$

Equivalent to :

$$\phi_n(i, k) = \sum_{m=0}^{N-1+(i-k)} s_n(m) s_n(m+i-k), \quad 1 \leq i \leq p, 0 \leq k \leq p$$

There are $N-|i-k|$ terms $\neq 0$ to the calculation of $\phi_n(i, k)$, for each i and k ;
We can show that:

$$\phi_n(i, k) = f(i-k) = R_n(i-k)$$

Where $R_n(i-k)$ is the short-term autocorrelation of $s_n(m)$ around $(i-k)$ is:

$$R_n(k) = \sum_{m=0}^{N-1-k} s_n(m)s_n(m+k)$$

$R_n(k)$ being an even function :

$$\phi_n(i, k) = R_n(|i - k|), \quad 1 \leq i \leq p, \quad 0 \leq k \leq p$$

and the equations become, with the minimum mean square error E_n , of the form :

$$\sum_{k=1}^p \alpha_k \phi_n(i - k) = \phi_n(i, 0), \quad 1 \leq i \leq p$$

$$\sum_{k=1}^p \alpha_k R_n(|i - k|) = R_n(i), \quad 1 \leq i \leq p$$

$$E_n = \phi_n(0,0) - \sum_{k=1}^p \alpha_k \phi_n(0,k)$$

$$= R_n(0) - \sum_{k=1}^p \alpha_k R_n(k)$$

In matrix form, then:

$$\begin{bmatrix} R_n(0) & R_n(1) & \cdot & \cdot & R_n(p-1) \\ R_n(1) & R_n(0) & \cdot & \cdot & R_n(p-2) \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot \\ R_n(p-1) & R_n(p-2) & \cdot & \cdot & R_n(0) \end{bmatrix} \begin{bmatrix} \alpha_1 \\ \alpha_2 \\ \cdot \\ \cdot \\ \alpha_p \end{bmatrix} = \begin{bmatrix} R_n(1) \\ R_n(2) \\ \cdot \\ \cdot \\ R_n(p) \end{bmatrix}$$

$$\mathfrak{R}\alpha = r$$

The solution : $\alpha = \mathfrak{R}^{-1}r$

- Matrix R (pxp) is a Toeplitz matrix that allows an efficient method of obtaining the solution (e.g. the Levinson-Durbin Algorithm).

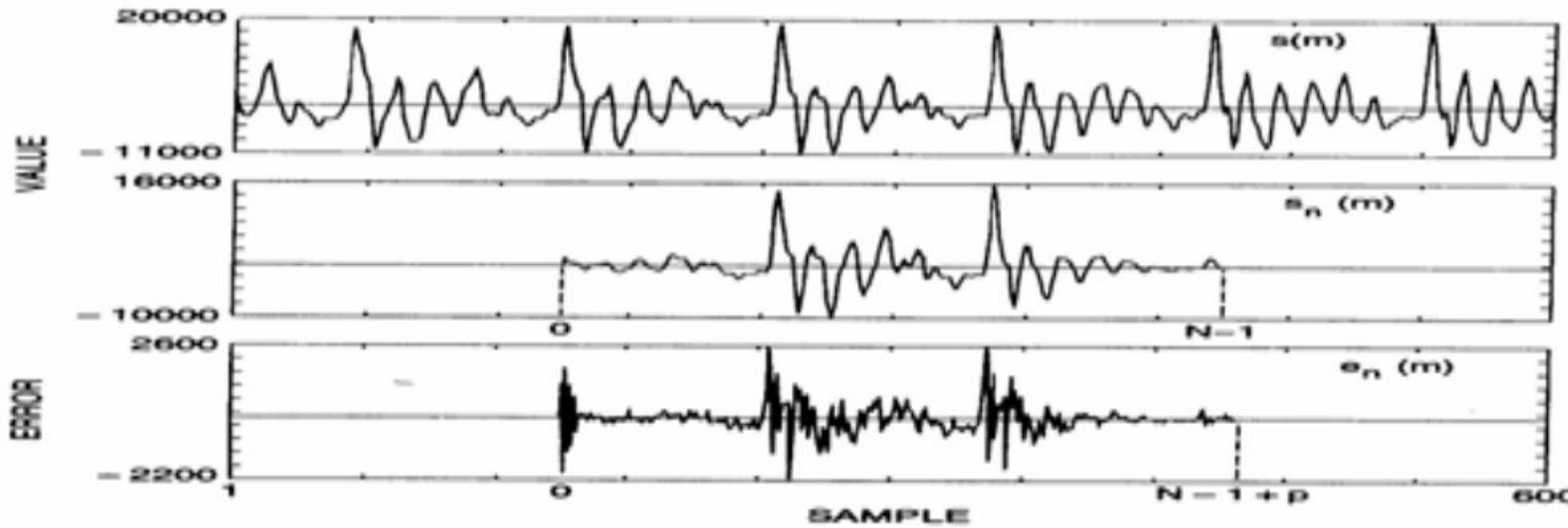


Figure 3.29 Illustration of speech sample, weighted speech section, and prediction error for voiced speech where the prediction error is large at the beginning of the section.

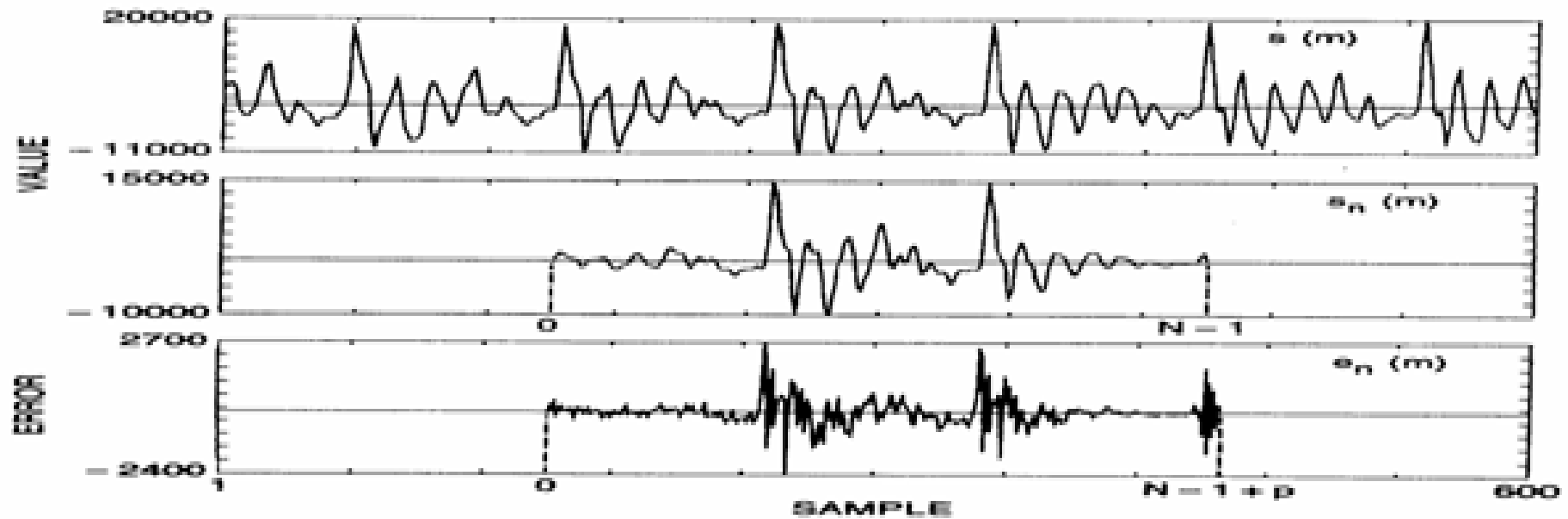


Figure 3.30 Illustration of speech sample, weighted speech section, and prediction error for voiced speech where the prediction error is large at the end of the section.

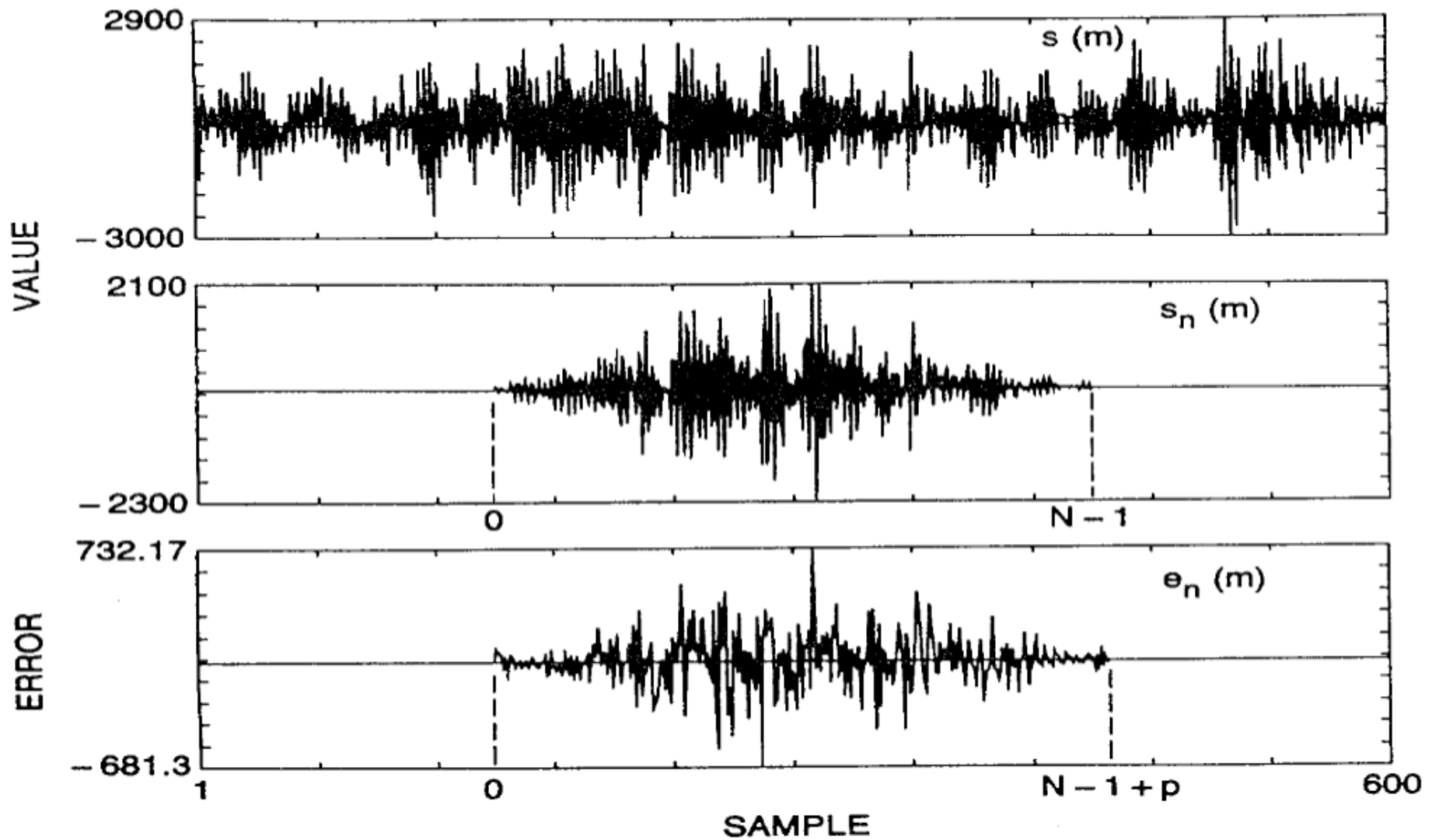
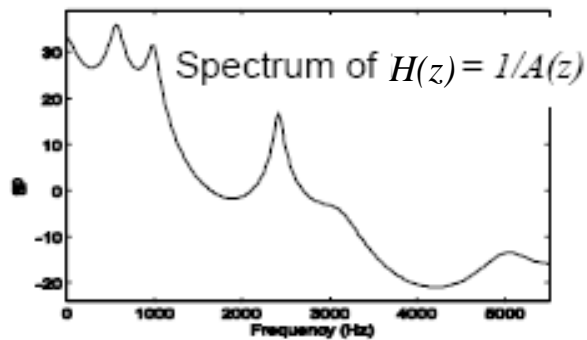
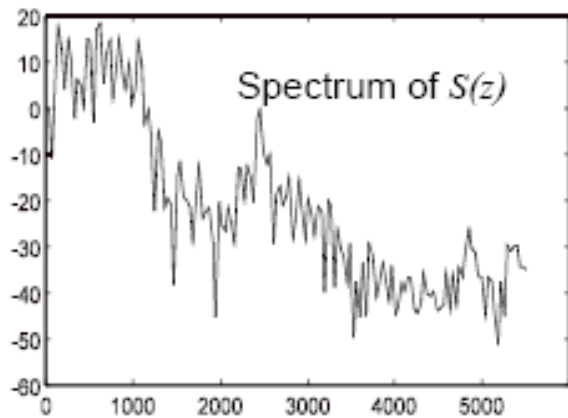
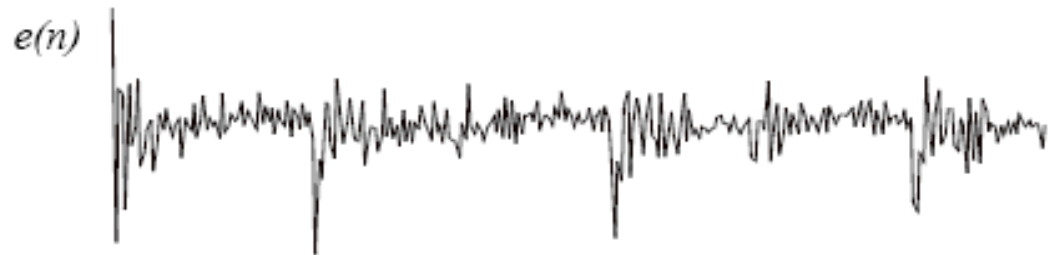
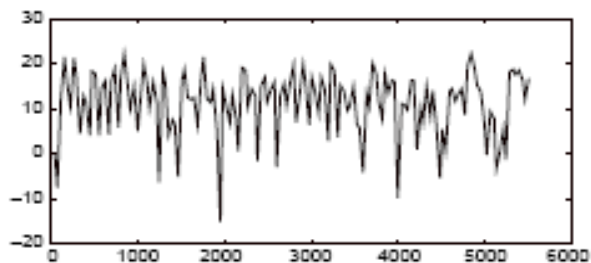


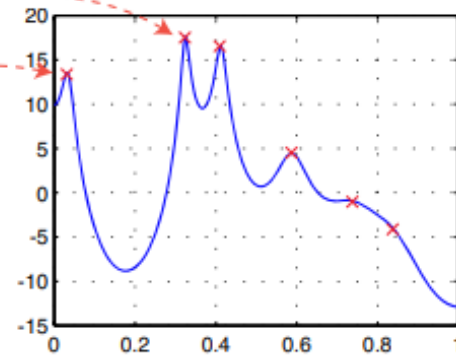
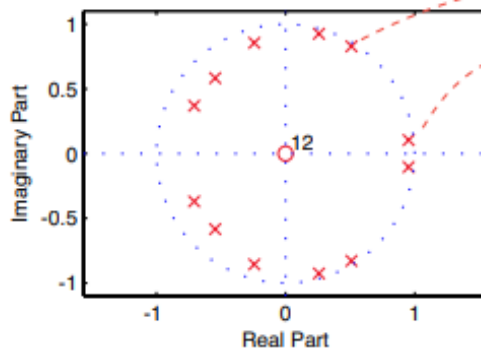
Figure 3.31 Illustration of speech sample, weighted speech section, and prediction error for unvoiced speech where there are almost no artifacts at the boundaries of the section.



Spectrum of $E(z) = S(z)A(z)$

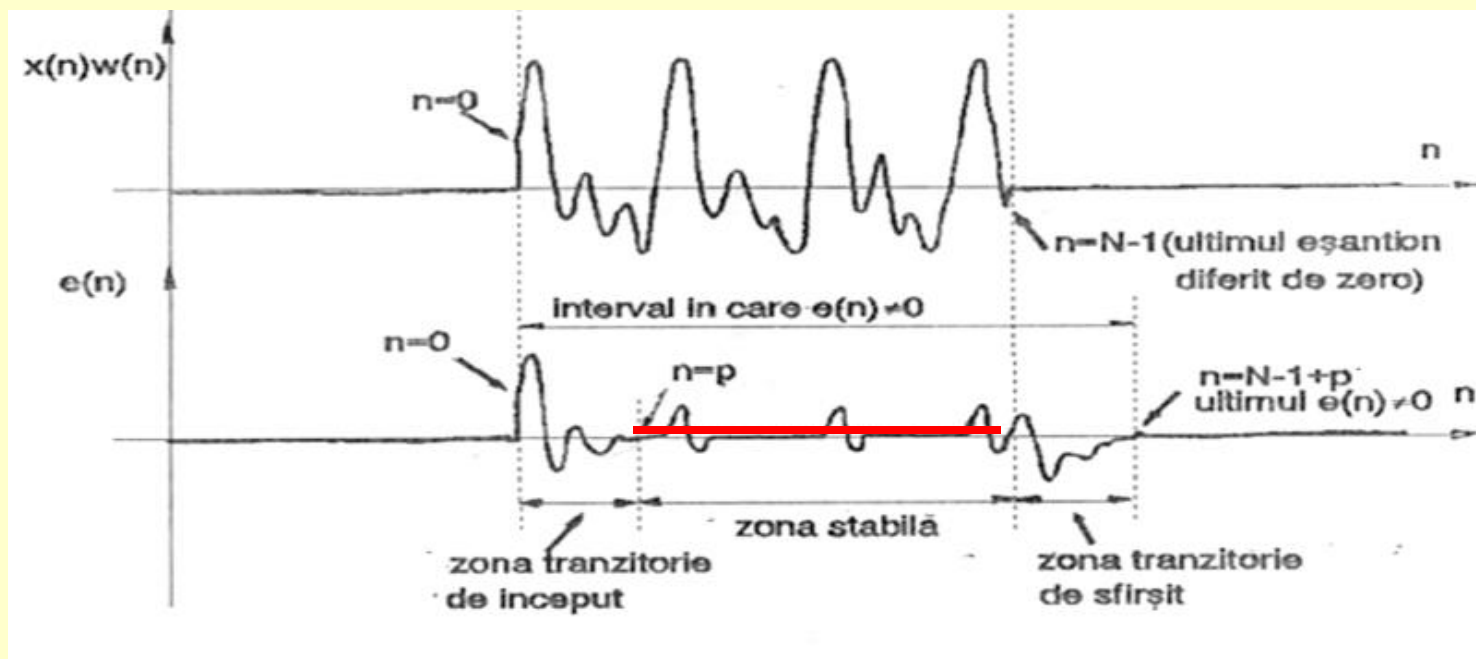


Poles of $H(z)$



2. The covariance method

We fix the interval for which the error is stable: $(n + p) \dots (n + N-1)$, $N-p$ - samples



- Symmetric matrix, but not Toeplitz; => other solutions

$$\sum_{k=1}^p \alpha_k \phi_n(i, k) = \phi_n(i, 0), \quad i = 1, 2, \dots, p$$

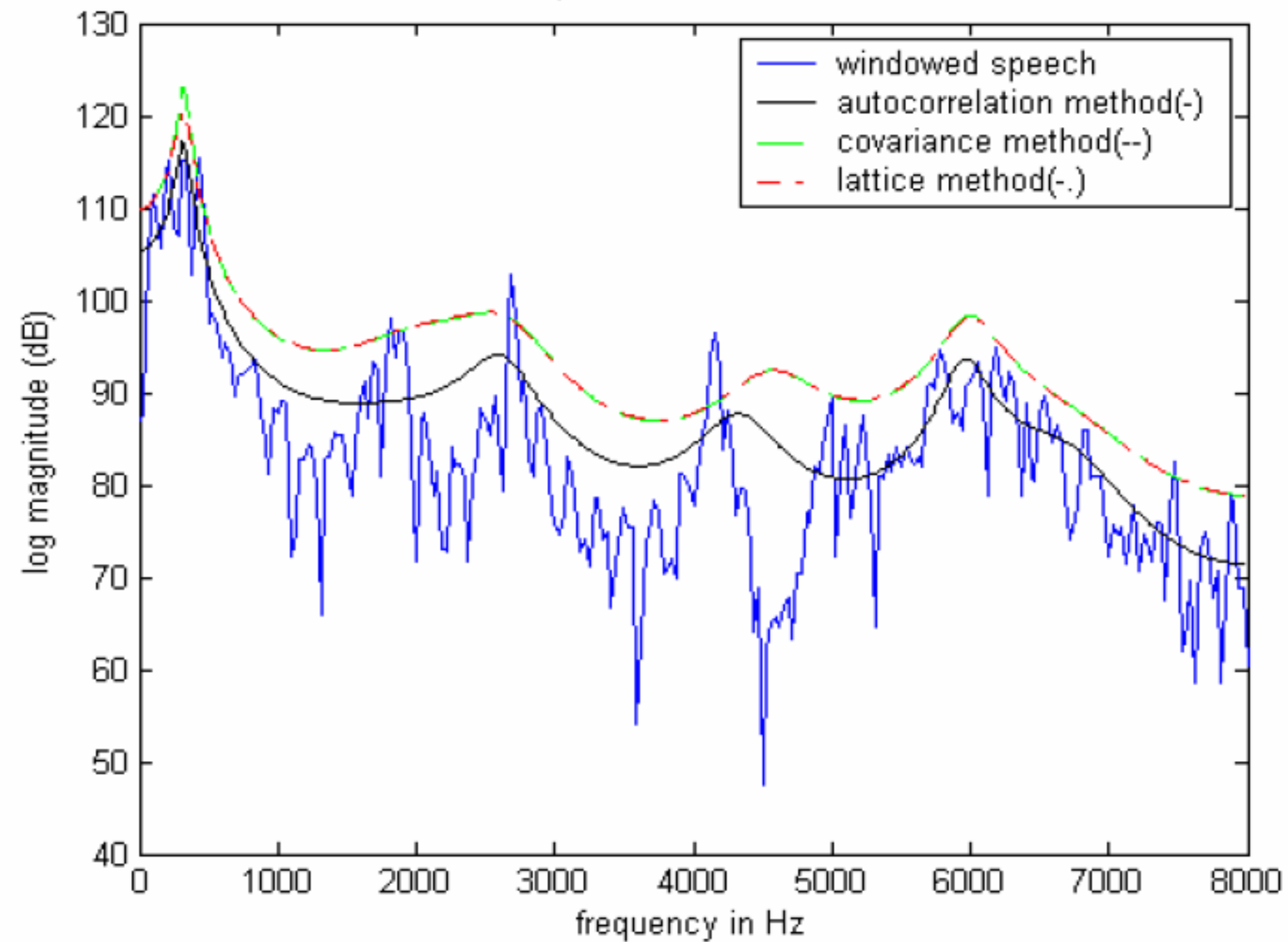
$$E_n = \phi_n(0, 0) - \sum_{k=1}^p \alpha_k \phi_n(0, k)$$

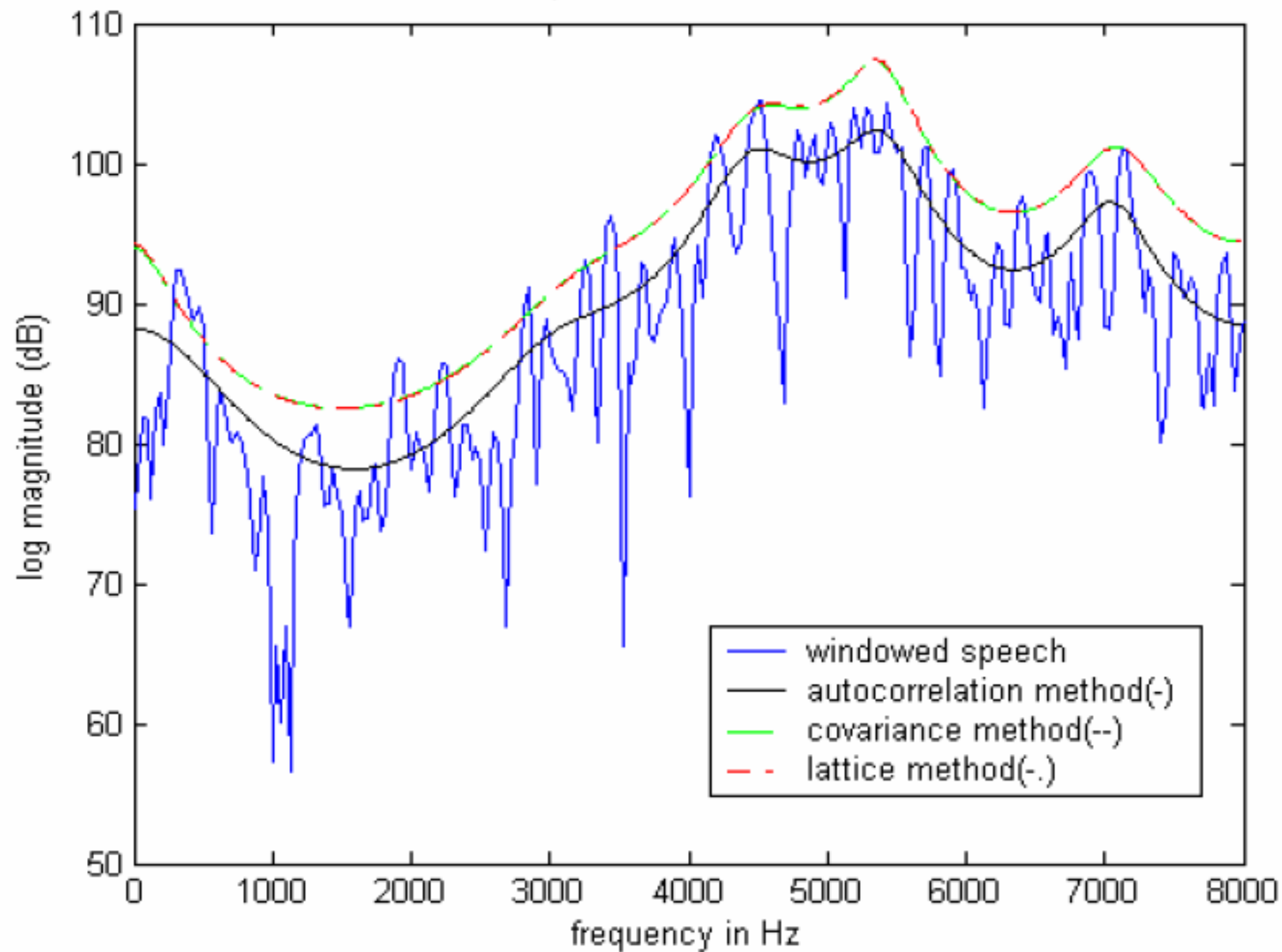
The covariance and autocorrelation methods use two steps to find the solution:

1. Calculation of the matrix of correlation values
2. Finding efficient solutions to a set of linear equations

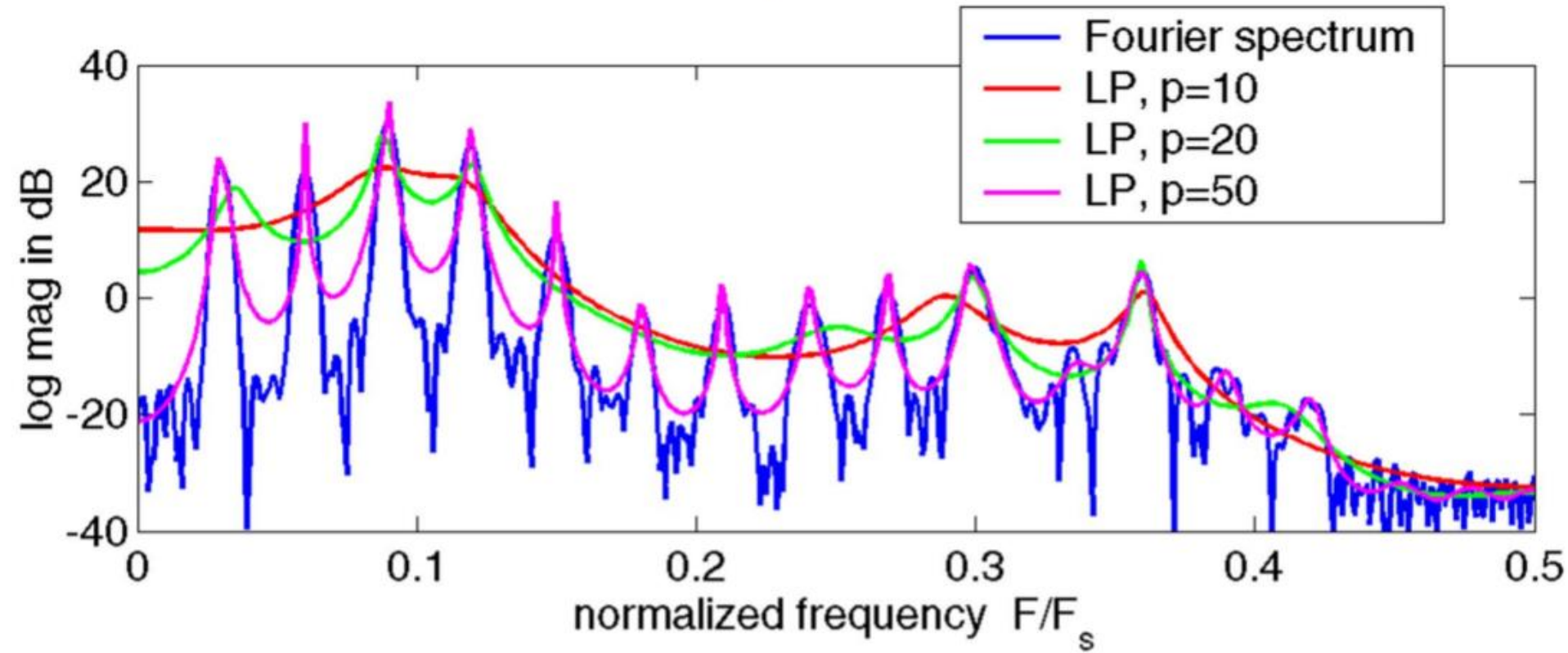
- Another class of LP methods has been developed, the lattice methods, in which the two steps are combined in a recursive algorithm to determine the LP parameters (start with the Levinson-Durbin algorithm).

file: test_6k, ss: 1000 N: 400 p: 12





LPC Spectrum



<https://www.ece.ucsb.edu/Faculty/Rabiner/ece259/>

Gain (G) estimation

- G is determined from the energy equality of the predicted signal and sample.

$$Gu(n) = s(n) - \sum_{k=1}^p a_k s(n-k) \Rightarrow \text{model}$$

Minimum error:

$$e(n) = s(n) - \sum_{k=1}^p \alpha_k s(n-k)$$

$$e(n) = Gu(n)$$

to equality

$$\alpha_k = a_k$$

- In the absence of a perfect fit, we use the equality between prediction error energy and excitation energy.
- We assume that for *voiced segments* $u(n) = \delta(n)$ and for *unvoiced segments* $u(n) = \text{white noise}$, ($m=0$, $\text{var}=1$).

$$G^2 = R_n(0) - \sum_{k=1}^p \alpha_k R_n(k) = E_n$$

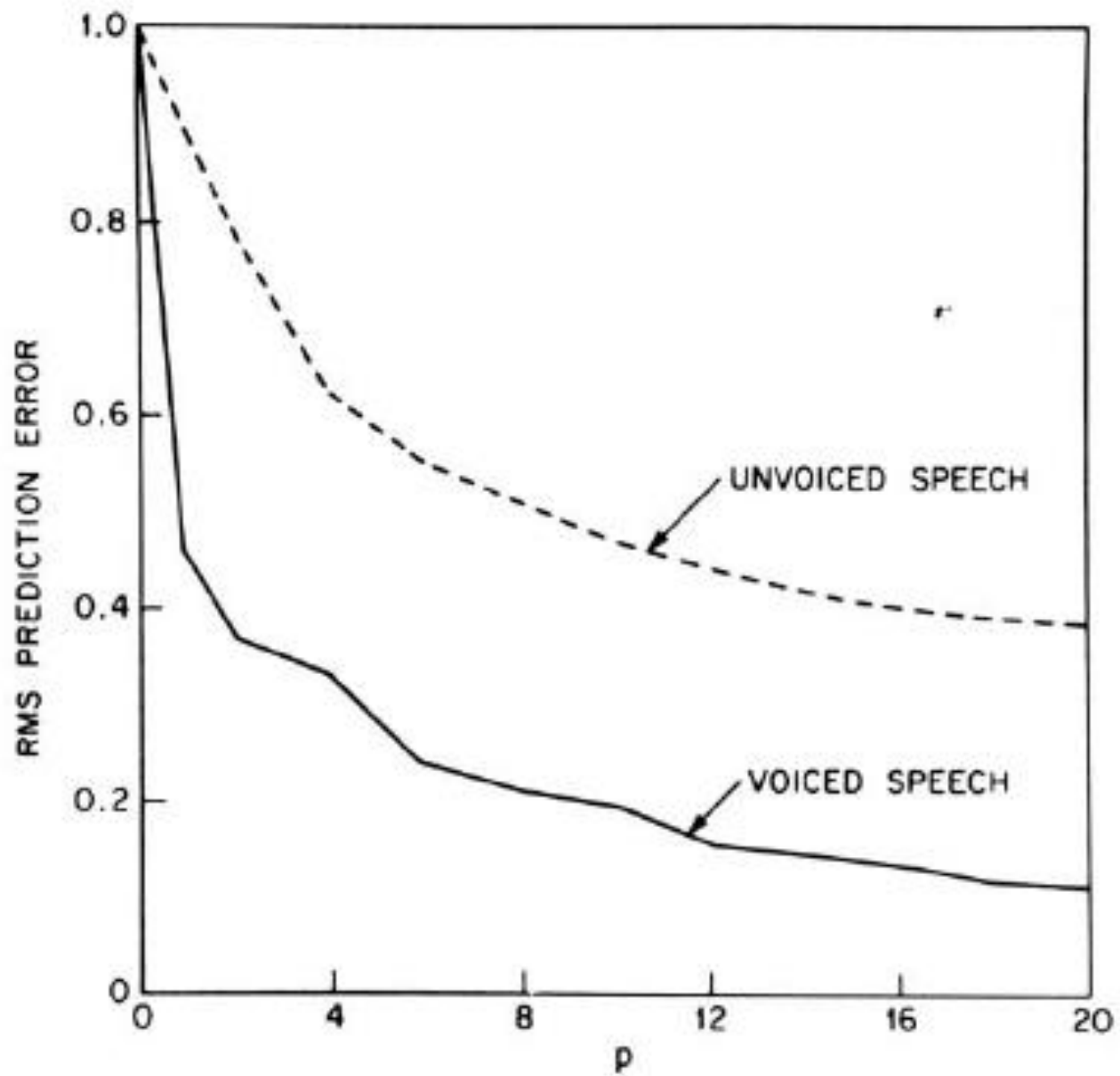


Fig. 8.4 Variation of the RMS prediction error with the number of predictor coefficients, ρ . (After Atal and Hanauer [3].)

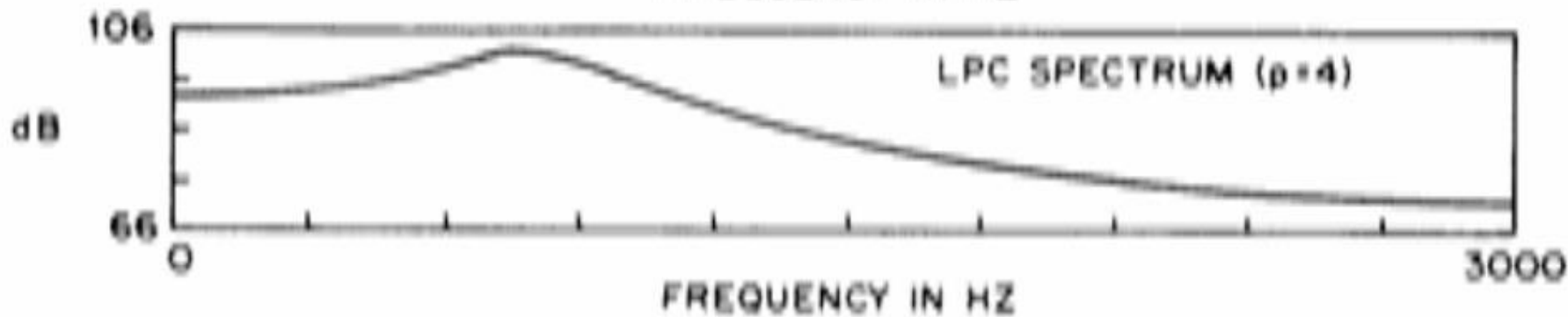
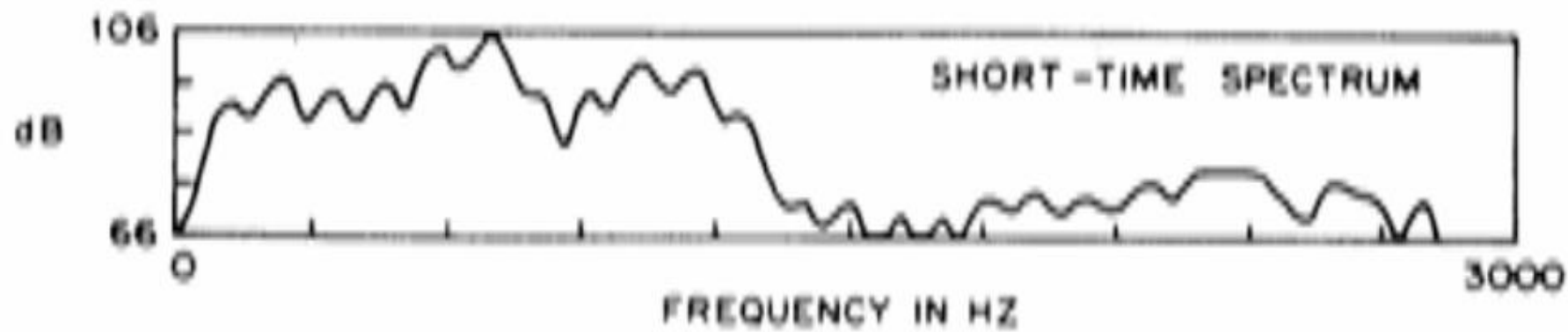
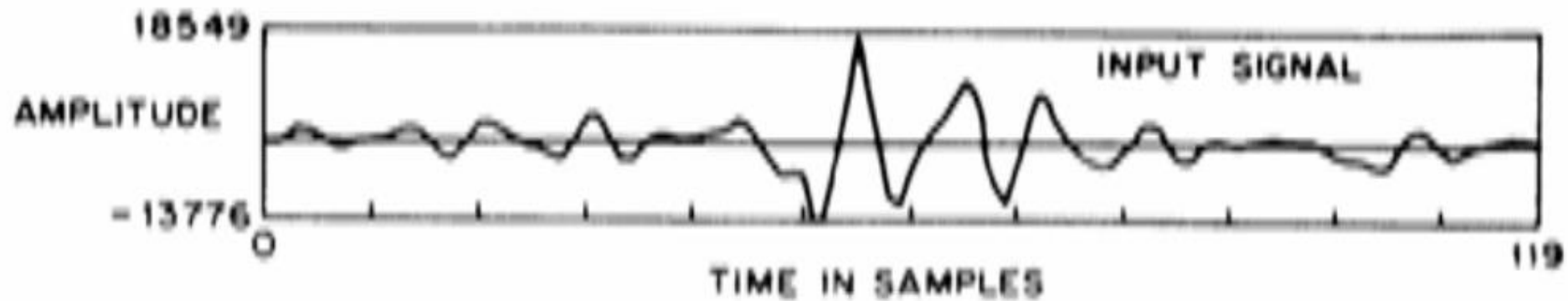
Comparisons between LP solutions

- stability guaranteed for autocorrelation and lattice method
- No guarantee for covariance method; (window size)
- Choice of analysis parameters for LP
- Works with 2 poles for each tract resonance under $F_s/2$
- Requires 3-4 poles to represent source shape and radiation load
- As an empiric/usual rule, $p \approx 12-16$

Empirically: predictor memory (p) must be at least equal to the time for sound to travel twice the distance of the vocal tract

$$T_p = 2 L_{\text{tract}} / V_{ss} = (2 * 0.17 \text{ m} / 340 \text{ m/s}) \sim 1 \text{ ms}$$

$$F_s = 8 \text{ kHz} \quad (T_s = 125 \mu\text{s}); \quad T_p / T_s = 1 \text{ ms} / 0.125 \text{ ms} \Rightarrow p = 8$$



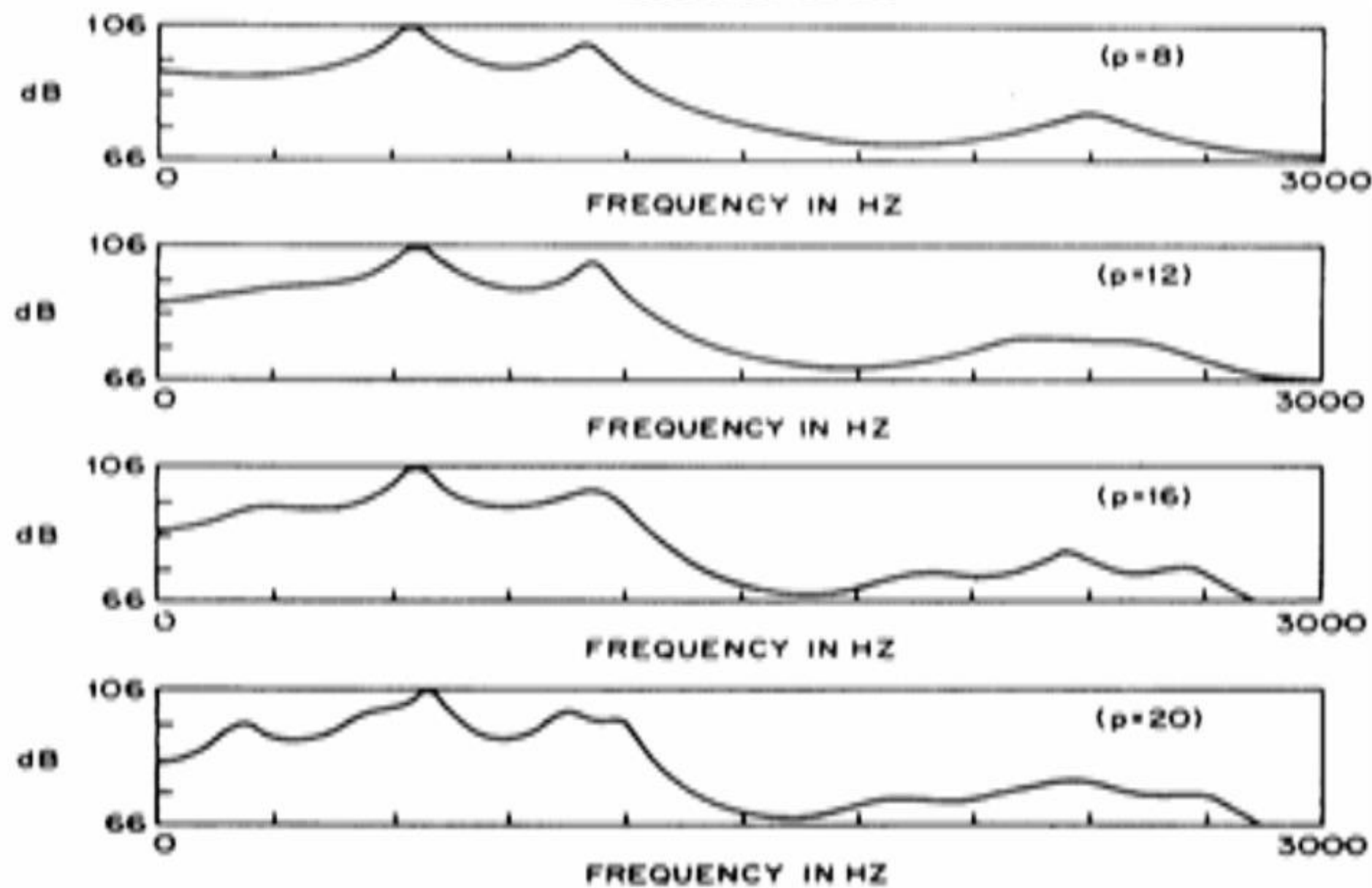


Figure 3.36 Spectra for a vowel sound for several values of predictor order, p .

LPC processing for recognition

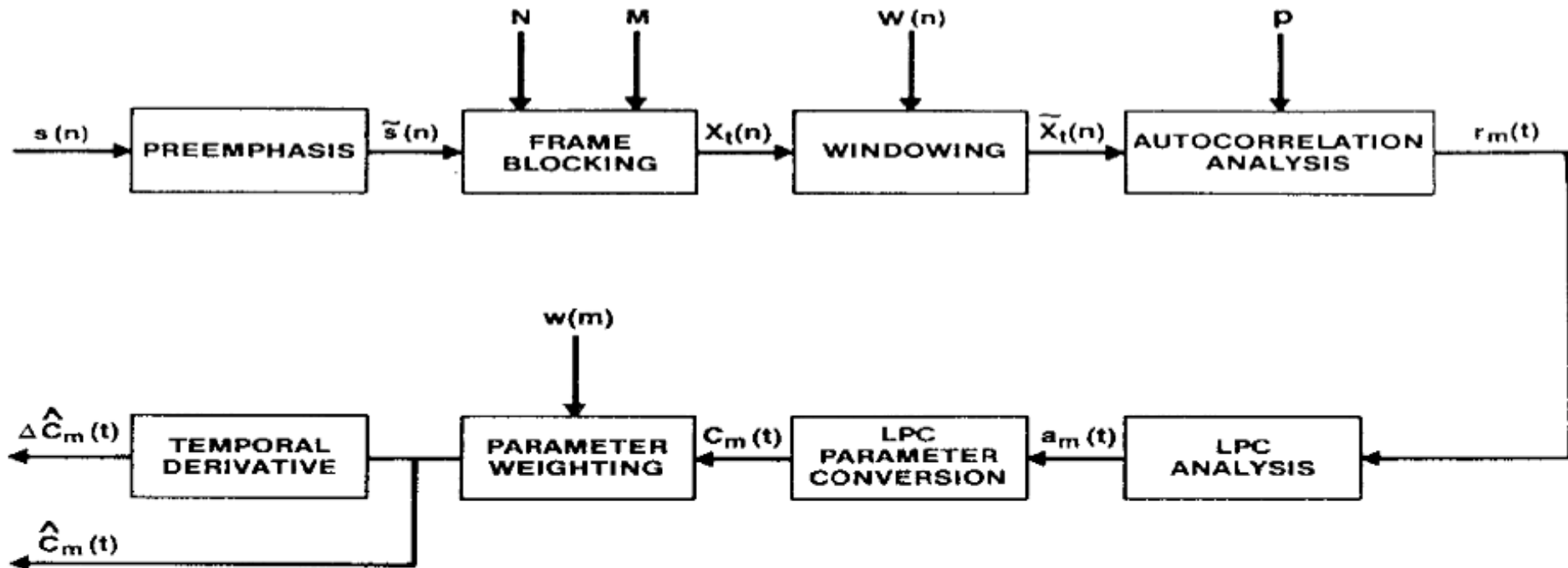


Figure 3.37 Block diagram of LPC processor for speech recognition.

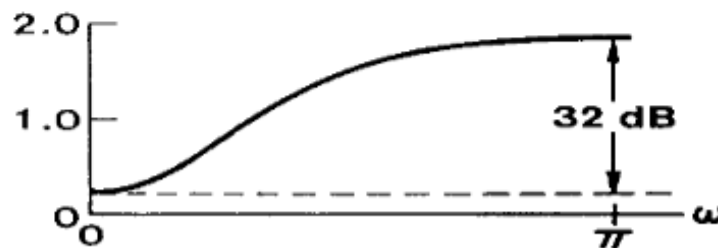


Figure 3.38 Magnitude spectrum of LPC preemphasis network for $\tilde{a} = 0.95$.

LPC processing for recognition

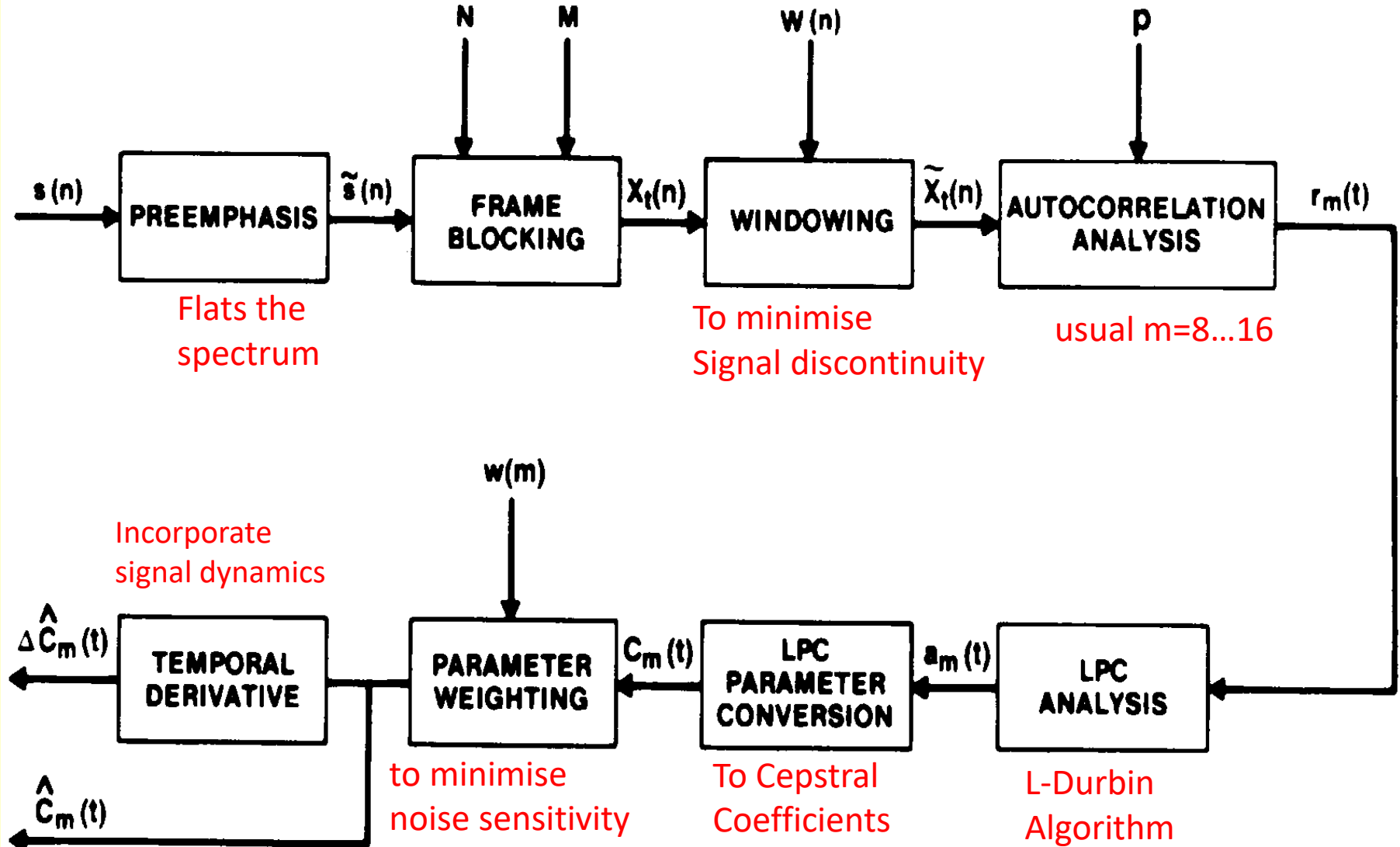


Figure 3.37 Block diagram of LPC processor for speech recognition.

1. Preemphasis

- The glottis transfer function can be modeled as follows:

$$U_g(z) = \frac{1}{(1 - \mu_1 z^{-1})(1 - \mu_2 z^{-1})} \quad 0.9 \leq \mu_1, \mu_2 \leq 1.0$$

- The radiation effect can be modeled as follows

$$R(z) = 1 - \mu_1 z^{-1}, \quad 0.9 \leq \mu_1 \leq 1.0$$

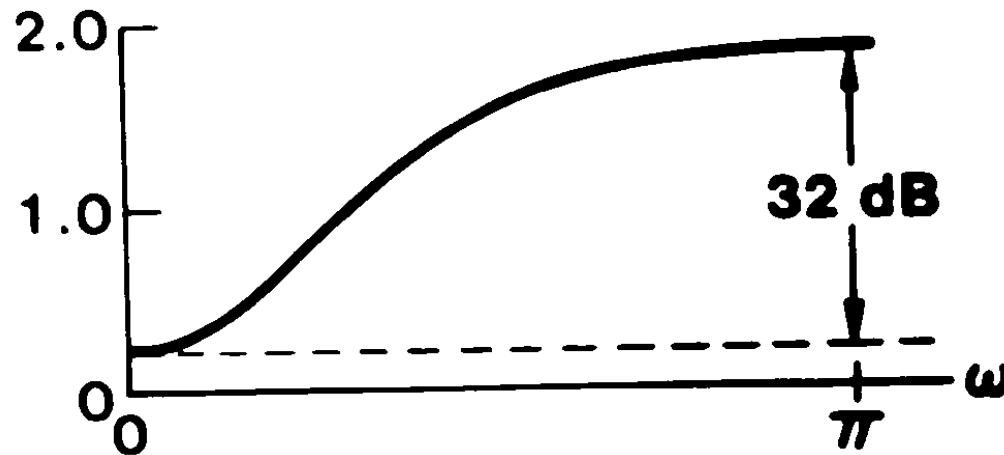
- Therefore, to obtain the transfer function of the vocal tract, the other pole must be canceled as follows:

$$H(z) = 1 - \mu_2 z^{-1}, \quad 0.9 \leq \mu_2 \leq 1.0$$

- The acquired signal $s(n)$ is filtered with a first-order HPF to smooth the signal spectrum and raise the level of high-frequency components to that of low-frequency components.

$$H(z) = 1 - \tilde{a} z^{-1}, \quad 0.9 \leq a \leq 1.0.$$

$$\tilde{s}(n) = s(n) - \tilde{a} s(n-1).$$

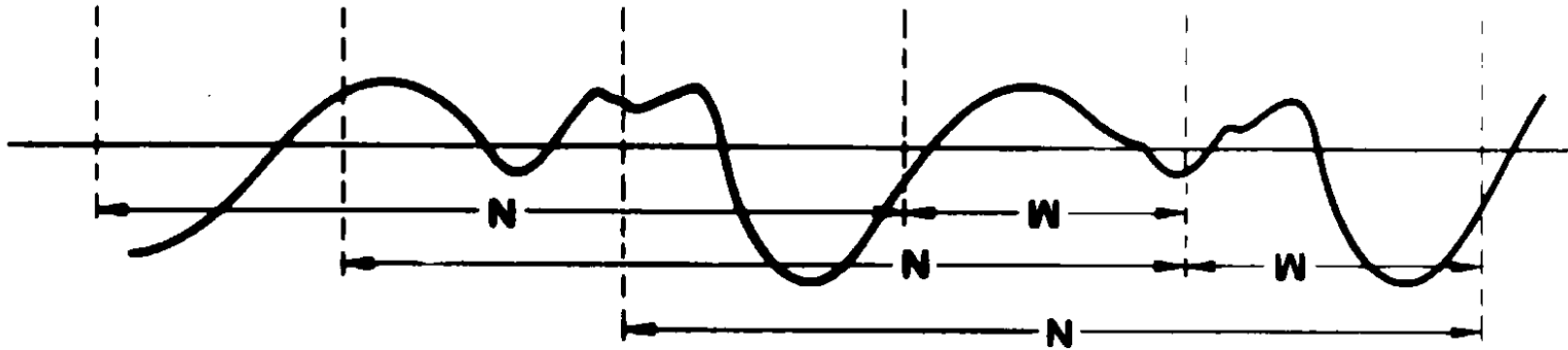


Magnitude spectrum of
LPC preemphasis network for $\tilde{a} =$
0.95.

2. Dividing the signal into blocks

- The filtered signal is divided into frames of N samples. The frames may be adjacent or overlap ($M < N$). A frame l is noted :

$$x_l(n) = \check{s}(Ml+n), \quad n = 0, 1, \dots, N-1 \quad \text{și} \quad l = 0, 1, \dots, L-1$$



Blocking of speech into overlapping frames.

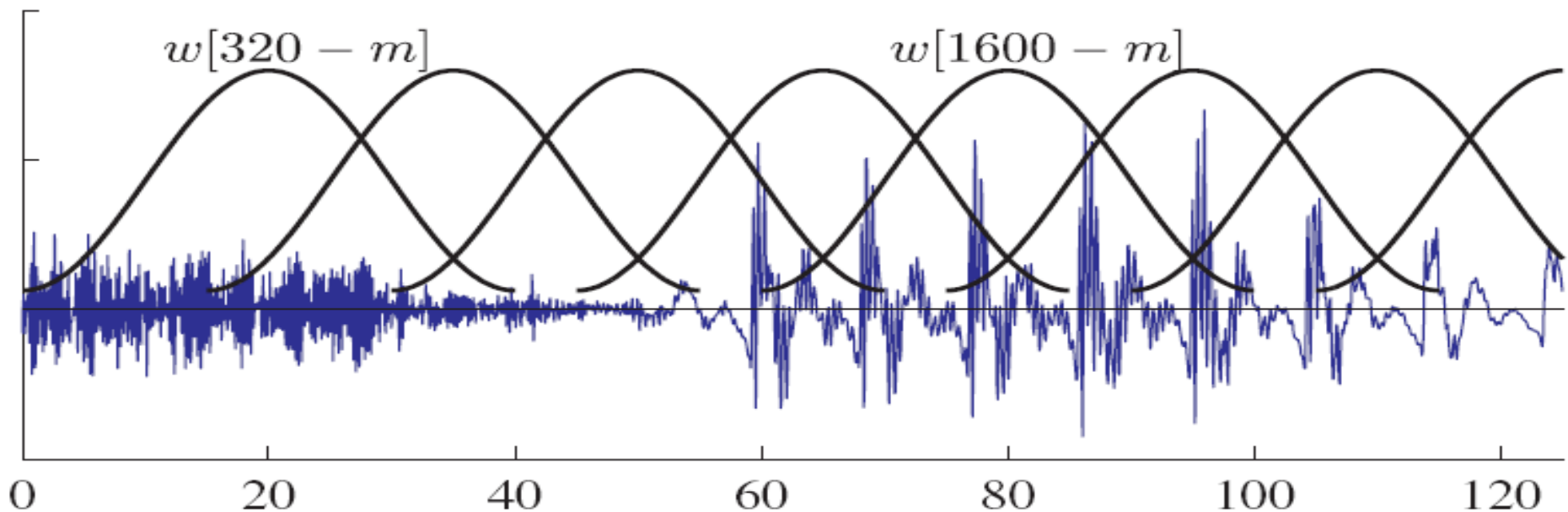
3. Windowing

- To reduce discontinuities at the ends of frames, they are weighted using a weighting/window function $w(n)$:

$$x_1(n) = x_1(n) w(n) \quad 0 \leq n \leq N-1$$

- Usual is employed Hamming window:

$$w(n) = 0.54 - 0.46 \cos(2\pi n / (N-1)) \quad 0 \leq n \leq N-1$$



4. Autocorrelation analysis

- Each frame is then weighted and autocorrelated to obtain:

$$r_1(m) = \sum_{n=0}^{N-1-m} x_1(m) x_1(m+n) \quad m = 0, 1, \dots, p$$

- The largest value of the autocorrelation is p , the order of the LPC analysis
- As a rule, $p = 8 \dots 16$
- Note that the function $r_1(0)$, is the frame energy!!
- Frame energy is an important parameter in speech detection systems.

5. LPC analysis

- converts each frame of $p+1$ autocorrelation into a set of parameters that can be:
 - LPC coefficients,
 - reflection coefficients (PARCOR),
 - log area ratio coefficients,
 - cepstral coefficients
 - Other derived coefficients
- A process for converting autocorrelation coefficients => LPC parameters (for the autocorrelation process) Levinson – Durbin Method

$$E^{(0)} = r(0)$$

$$k_i = \left\{ r(i) - \sum_{j=1}^{L-1} a_j^{(i-1)} r(|i-j|) \right\} / E^{(i-1)}, 1 \leq i \leq p$$

$$a_i^{(i)} = k_i$$

$$a_j^{(i)} = a_j^{(i-1)} - k_i a_{i-j}^{(i-1)}, 1 \leq j \leq i-1$$

$$E^{(i)} = (1 - k_i^2) E^{(i-1)}$$

$$a_m = \alpha_m^{(p)}, 1 \leq m \leq p, \quad k_m, \quad \text{si} \quad g_m = \log \frac{1 - k_m}{1 + k_m}$$

coefficients **LPC**

coefficients **PARCOR**

coefficients **log area ratio (LAR)**

The LPC smooth spectrum

- From the prediction coefficients obtained, we can determine the LPC (smoothed) spectrum

$$H(z) = \frac{G}{1 - \sum_{k=1}^p a_k z^{-k}}$$

Tract transfer function

Note:

$$A(z) = 1 - \sum_{k=1}^p a_k z^{-k}$$

- This polynomial in z has the time response $f(n)$

$$f(n) = \begin{cases} 1, & n = 0 \\ a_n, & 1 \leq n \leq p \\ 0, & n > p \end{cases}$$

- To determine $A(j\omega)$, we can use the FFT on several samples ($N=256/512$):

$$\text{FFT} (1, a_1, a_2, \dots, a_n, 0, \dots, 0)$$

to obtain the $H(j\omega)$ spectrum, where $\omega=2\pi k/N$, $k=0, \dots, N-1$:

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

$$G = \sqrt{\sum_{i=0}^p a_i r_1(i)} = \sqrt{r(0) \prod_{i=1}^p (1 - k_i^2)}$$

Converting prediction coefficients into cepstral coefficients

- Cepstrum \gg IFFT(log(f(t))), and for an only poles filter, we have :

$$\ln H(z) = \ln \frac{G}{1 + \sum_{i=1}^p a_i z^{-i}} = \sum_{n=1}^q c_n z^{-n}$$

-The coefficients $c(m)$ are obtained if both parts of the expression are derived to z^{-1} , resulting from the recurrence formulas :

$$c_1 = -a_1$$

$$c_m = -a_m - \sum_{k=1}^{m-1} \left(1 - \frac{k}{m}\right) c_{m-k} a_k, 1 < m \leq p$$

$$c_m = -\sum_{k=1}^{m-1} \left(1 - \frac{k}{m}\right) c_{m-k} a_k, m > p$$

- typical parameter values used in LPC analysis for speech recognition [Rab93]:

| parametru | fe=6.67KHz | fe=8KHz | fe=10kHz |
|-----------|------------|---------|----------|
| N | 300 | 240 | 300 |
| M | 100 | 80 | 100 |
| p | 8 | 10 | 12 |
| q | 12 | 12 | 12 |

N - number of samples in an analysis frame

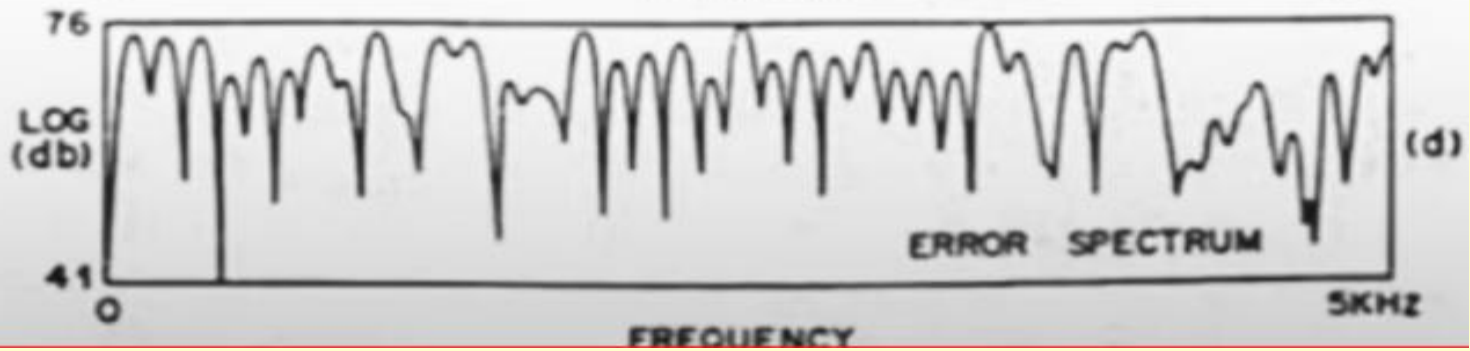
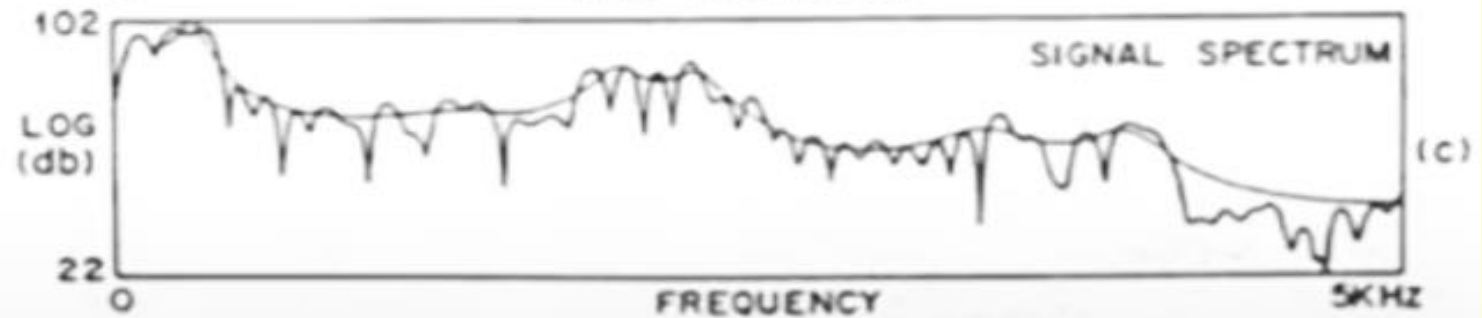
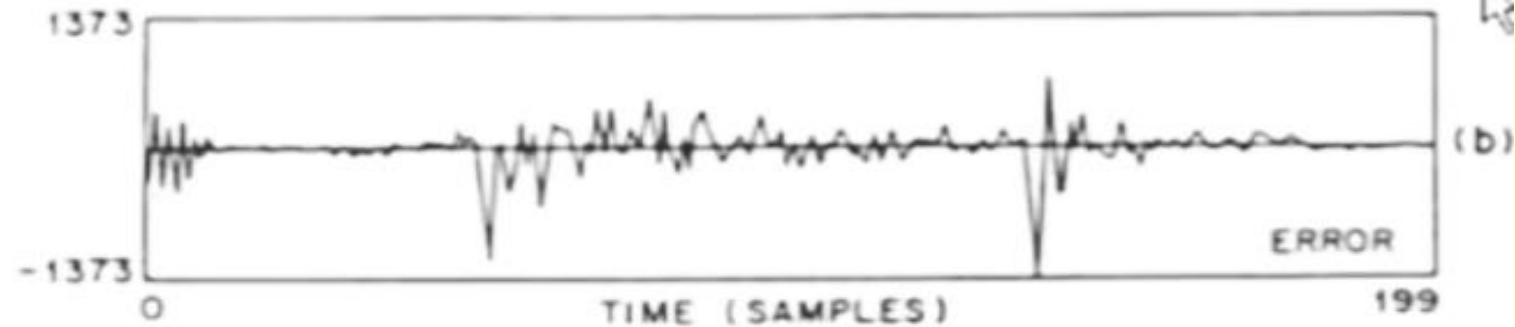
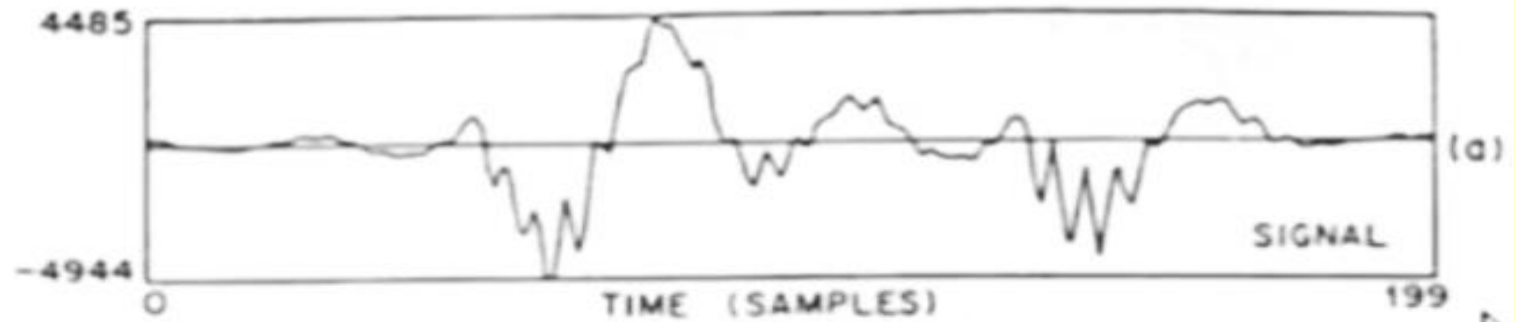
M - number of frame displacement samples

p - LPC analysis order

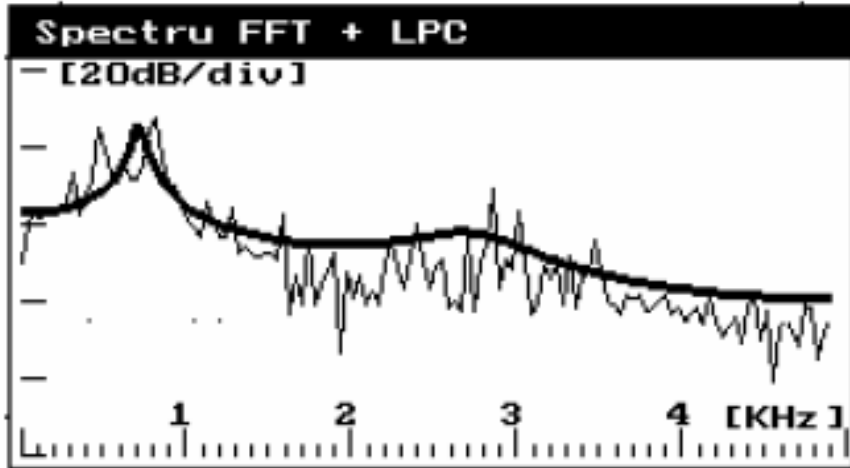
q- vector size of LPC-derived cepstral coefficients (LPCC)

Obs. In general, using a representation with $q > p$, where $q \sim 1.5p$

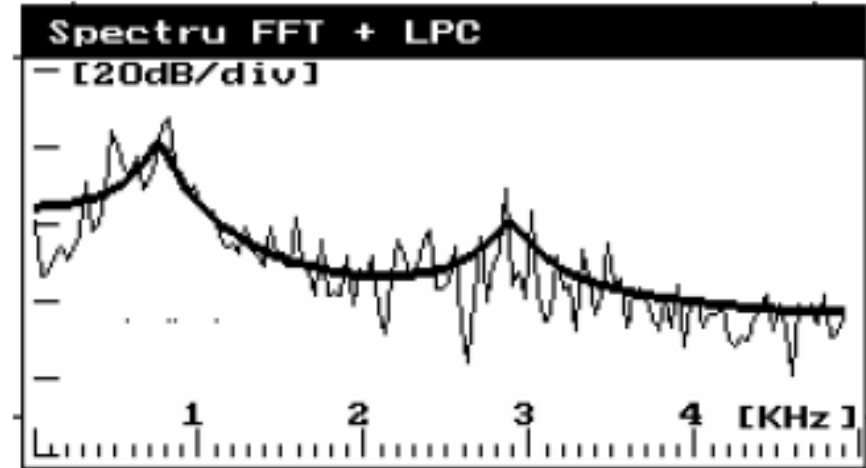
Examples of LPC analysis:



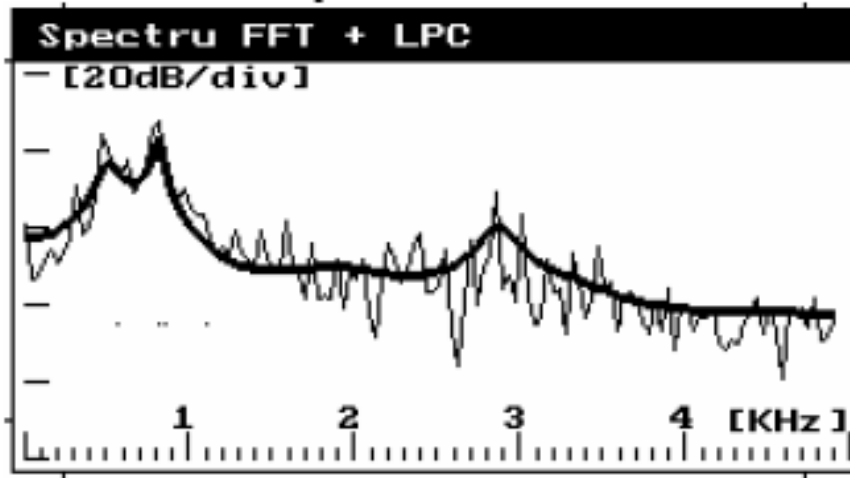
LPC spectrum examples



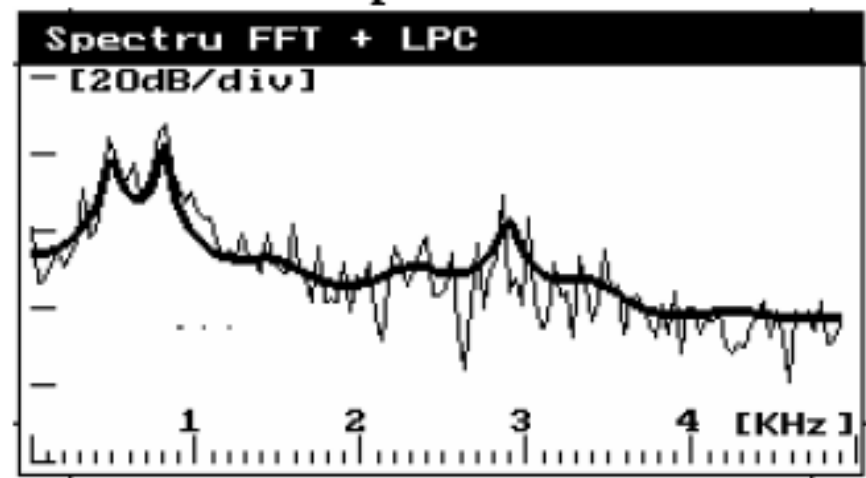
$p=4$



$p=8$

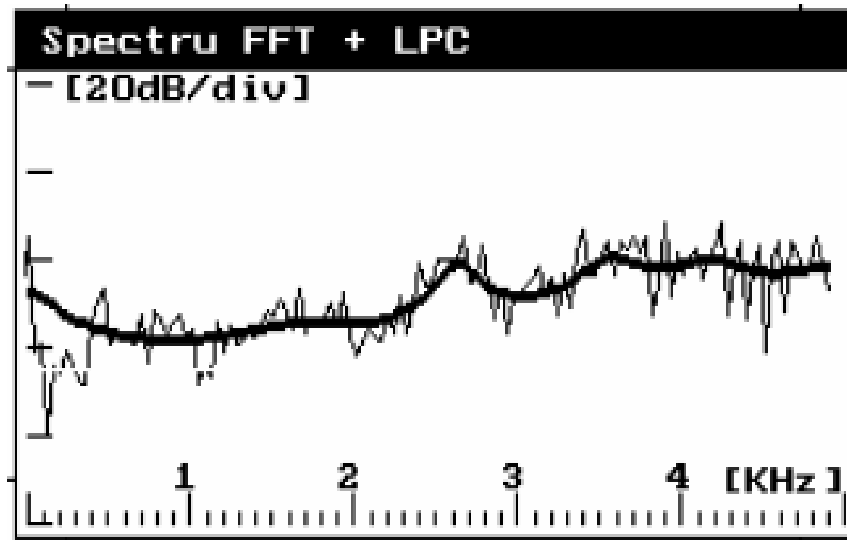
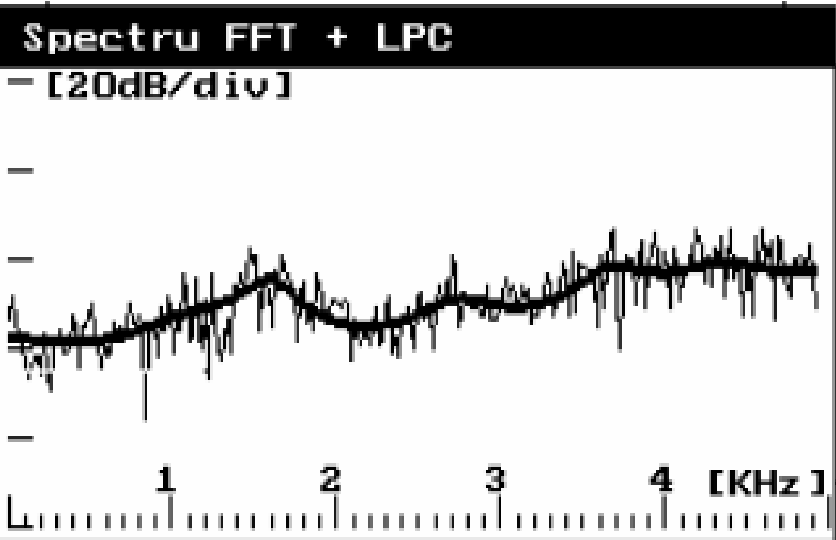


$p=12$

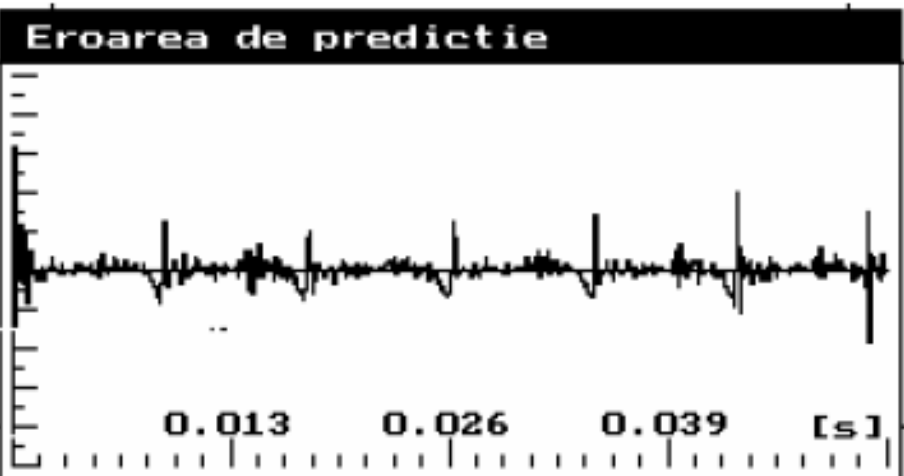


$p=16$

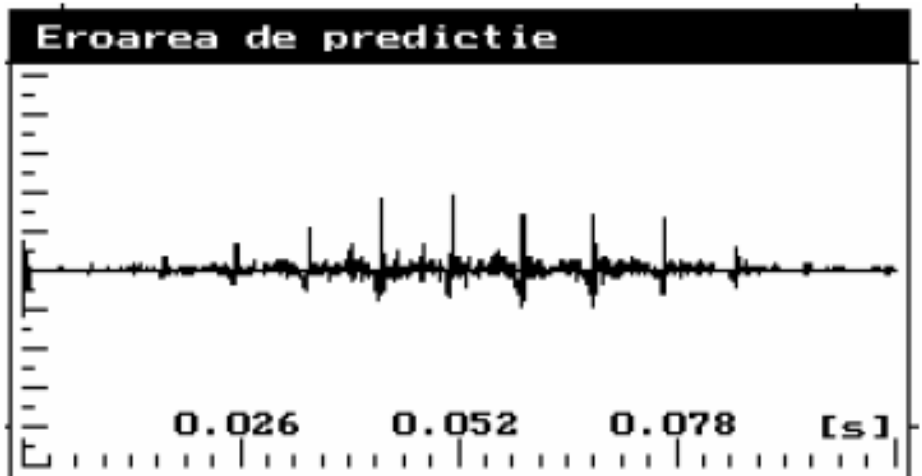
Log Fourier and LPC spectrum of the vowel "a" for $p=4,8,12,16$
and a 256-point Hamming window



Log Fourier and LPC spectra for the consonants "s" and "ș" (p=12)

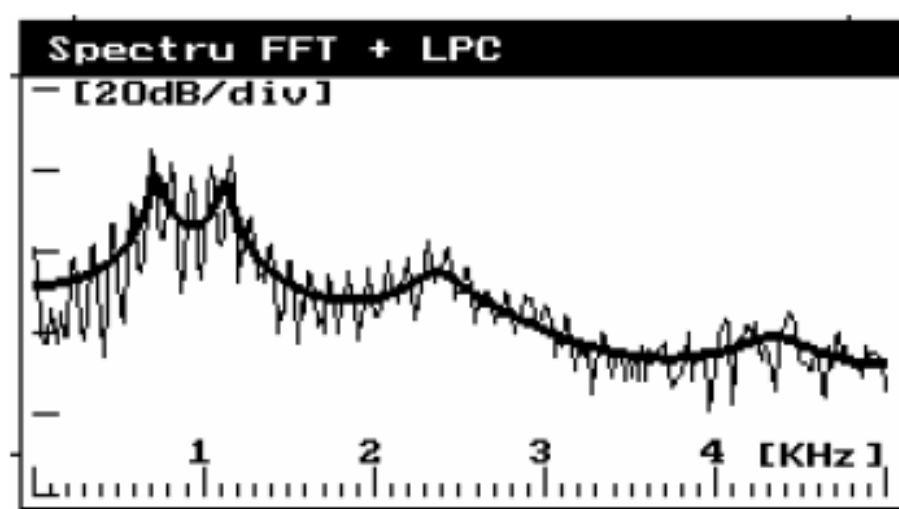


(a)

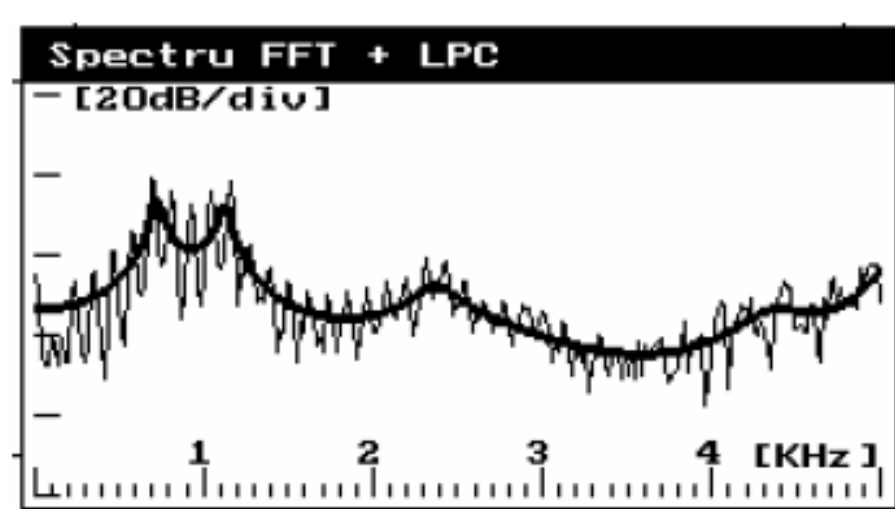


(b)

Prediction error for a voiced segment ("a"), rectangular window (a) and Hamming (b) (p=12)

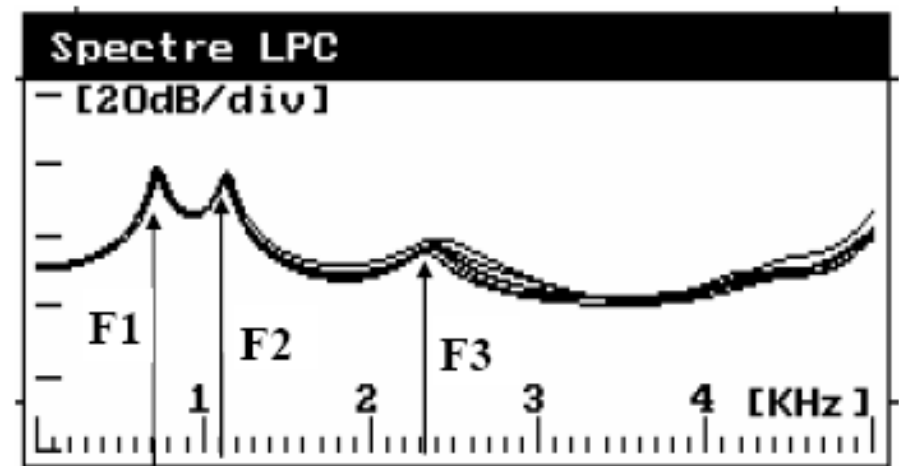


(a)

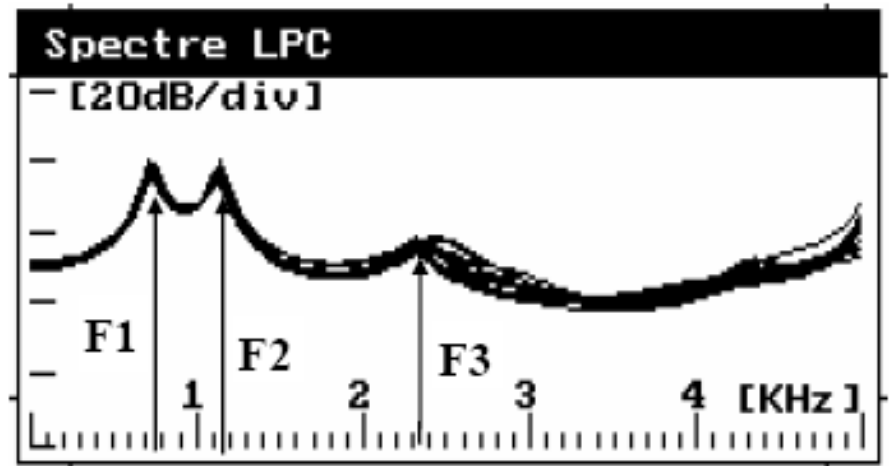


(b)

Log Fourier and LPC spectra of the vowel “a” without pre-emphasis (a) and with pre-emphasis (b) ($p=12$)



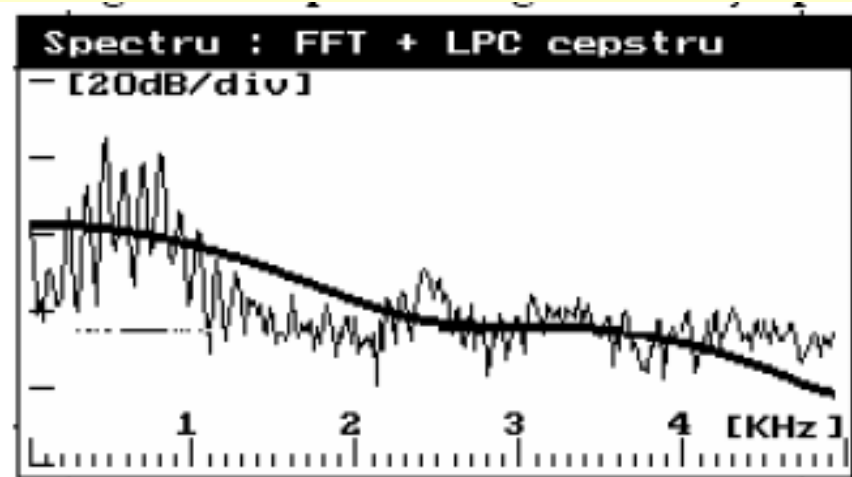
(a)



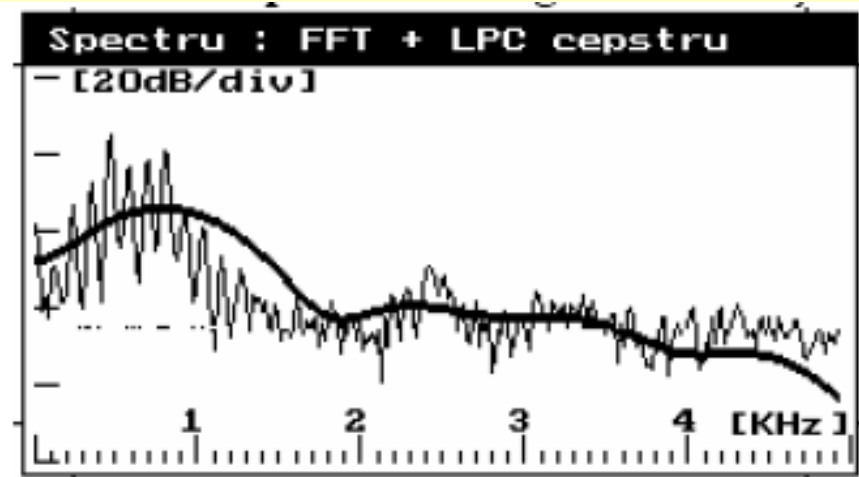
(b)

Spectral evolution of the LPC in successive frames on the utterance of the vowel “a” by a male speaker: (a) for 512-point windows and (b) for 256-point windows.

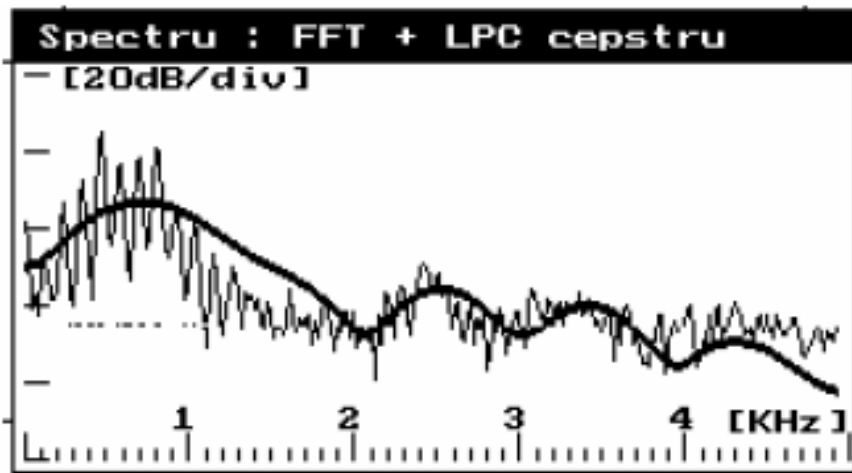
LPC Spectrum



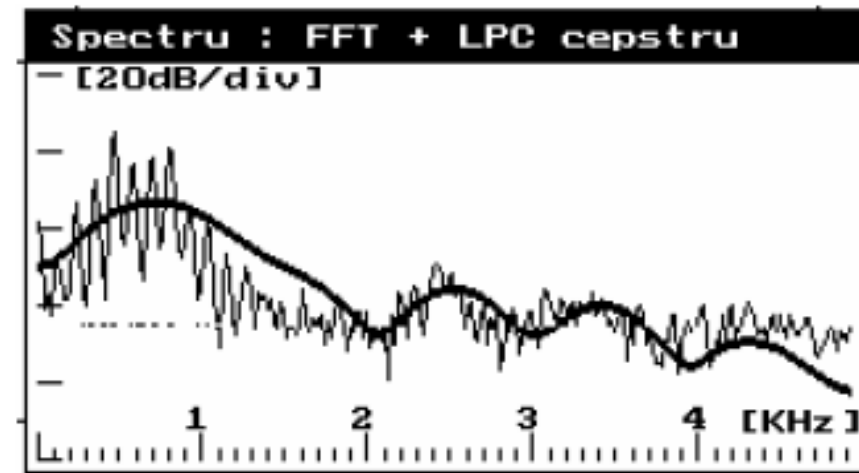
q=4



q=8



q=12



q=16

The log Fourier spectrum and LPC cepstral spectrum of the vowel "o"
for $q=4,8,12,16$ ($p=12$)

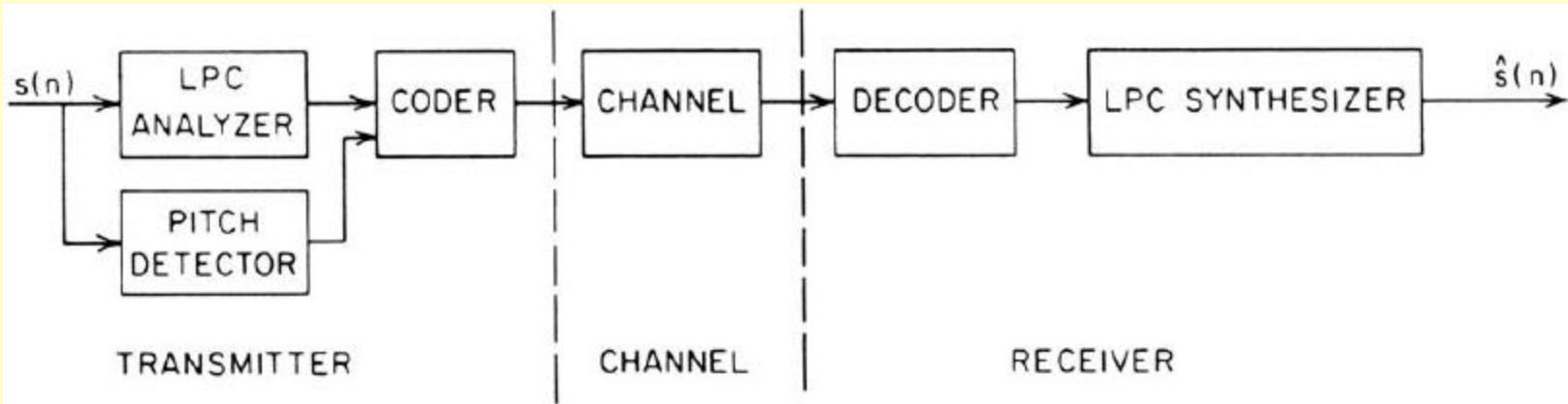
Homework.

A string of speech samples has the values $[s_0, s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8] = [1, 3, 2, 1, 4, 1, 2, 4, 3]$. The frame size is 4.

- Find the pre-emphasized values of the samples if $\tilde{a} = 0.98$.
- Find the autocorrelation parameters r_0, r_1, r_2 .
- If we use $p=2$ prediction order for the feature extraction system, find the LPC coefficients: a_0, a_1, a_2 .
- If the number of overlapping samples for two successive frames is 2, find the LPC coefficients in the second frame.

Exercises on LPC

1. Calculate LPC parameters of Speech
 - Use *LPC* function and perform pre-emphasis
2. Determination of Formants using LPC
3. Formant Tracking Using LPC
4. Inverse Filtering to determine $e(n)$
5. Determination of Pitch Period from $e(n)$
6. Pitch Period Tracking



Vocoder LPC

References.

Cours L.Rabiner **Linear Predictive Coding (LPC)-Introduction** ECE 259

<http://engineering.purdue.edu/VISE/ee438L/>

<http://www.ee.ic.ac.uk/hp/staff/dmb/voicebox/voicebox.html>