LINEAR PREDICTIVE COEFFICIENTS

1.1. Introduction

- Main in any pattern recognition problem is reducing the number of parameters needed to represent a signal.
- One way of doing that is to use LPCs (Linear Predictive Coefficients) to represent a signal.
- Value of the signal at time t is approxsimated with linear combination of real signal values in previous moments as shown in Main where a_k are LPCs.
- LPCs are calculated so that they minimize error betwen real signal and the one calculated using LPCs over the interval of interest as defined with (1.3). We define that the signal of interest is zero outside this interval.
- To make calculations easier sumation is extended by p points as defined in (1.31) making calculated signal non zero after original interval therefore introducing errors in that reagon between real and aprocimated signal.

1.2. Calculating LPCs

Extra information about this method can be found in reference [1] in chapter 3.3 which starts on page 97.
Each function can be aproximatred with LPC (Linear Predictive Coefficients). This simply means that value of the signal at time t is approxsimated with linear combination of signal values in previous moments, which can be mathamaticly expressed as folows:

$$\tilde{f}_t = \sum_{k=1}^{p} a_k f_{t-k} \tag{1.1}$$

where:
- t – discrete time moment,
- f_t – original signal,
- $\tilde{f}_t$ – approxismation of the original signal,
- a_k – LPC (Linear Predictive Coefficient) ; $1 \le k \le p$,
- p – number of LPCs (Linear Predictive Coefficients).

Formula Main can also be written withouth sum as folows:

$$\tilde{f}_t = a_1 f_{t-1} + \ldots + a_p f_{t-p} \tag{1.2}$$

Approxsimation function $\tilde{f}_t$ will be better if we use more LPCs, but generaly, functions f_t and $\tilde{f}_t$ will be different. This difference can be messured and expresed with the folowing formula:

$$E = \sum_{t=t_s}^{t_e} \left(f_t - \tilde{f}_t\right)^2 \tag{1.3}$$

where:
- t_s – starting point in time of the interval for which the error measure is being calculated,
- t_e – ending point in time of the interval for which the error measure is being calculated,
- E – erorr measure on the interval $t \in [t_s, t_e]$

Instead of using power to 2, any other function which always returns postitive values can also be used, becaue we are only interested in the difference between two functions and we don't care which signal was bigger at certan time. Formula (1.3) is crutial in determing LPCs such that approxsimation function $\tilde{f}_t$ is the best possible approxsimation of f_t for the given time interval. This is done simply by finding LPCs for which function E will be minimal. Extrems, minimum and maximus of any function are at the points where first derivaition of the function has value zero. To find LPCs which minimize E first we have to find expression for its first derivation. Taking first derivation of (1.3) gives us:

$$E = \sum_{t=t_s}^{t_e} \left(f_t - \tilde{f}_t\right)^2 \Big/ \frac{\partial}{\partial a_K} \tag{1.4}$$

$$\frac{\partial E}{\partial a_K} = \frac{\partial}{\partial a_K} \sum_{t=t_s}^{t_e} \left(f_t - \tilde{f}_t\right)^2 \tag{1.5}$$

Using formula ????, which says that derivation of the sum is equal to sum of derivations, derivation can be put inside sum, after which we get:

$$\frac{\partial E}{\partial a_K} = \sum_{t=t_s}^{t_e} \frac{\partial}{\partial a_K} \left(f_t - \tilde{f}_t\right)^2 \tag{1.6}$$

Using formula ???? for calculating derivation, we get:

$$\frac{\partial E}{\partial a_K} = \sum_{t=t_s}^{t_e} \left[2\left(f_t - \tilde{f}_t\right) \frac{\partial}{\partial a_K} \left(f_t - \tilde{f}_t\right) \right] \tag{1.7}$$

Once again we use formula ????, which says that derivation of the sum is equal to sum of derivations to get:

$$\frac{\partial E}{\partial a_K} = 2\sum_{t=t_s}^{t_e} \left[\left(f_t - \tilde{f}_t\right) \left(\frac{\partial f_t}{\partial a_K} - \frac{\partial \tilde{f}_t}{\partial a_K} \right) \right] \tag{1.8}$$

Since f_t is not dependeble on a_K, it is just the value of original signal measured at time t, partial derivation of f_t is equal to zero. On the other hand, $\tilde{f}_t$ is dependeble on a_K, because of (1.1) so the partial derivaton of $\tilde{f}_t$ stays and we get:

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left[\left(f_t - \tilde{f}_t\right) \frac{\partial \tilde{f}_t}{\partial a_K} \right] \tag{1.9}$$

Using (1.1) to replace $\tilde{f}_t$ we get:

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left[\left(f_t - \tilde{f}_t\right) \frac{\partial}{\partial a_K} \sum_{k=1}^{p} \left(a_k f_{t-k}\right) \right] \tag{1.10}$$

Using formula ????, which says that derivation of the sum is equal to sum of derivations, derivation can be put inside sum, after which we get:

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left[\left(f_t - \tilde{f}_t\right) \sum_{k=1}^{p} \left(\frac{\partial}{\partial a_K} a_k f_{t-k} \right) \right] \tag{1.11}$$

Since only summand $a_K f_{t-K}$ is depended on a_k, partial derivations of all other summands are equal to zero. Partial derivation of $a_K f_{t-K}$ will be f_{t-K} because of ????. After this we get:

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left[\left(f_t - \tilde{f}_t\right) f_{t-K} \right] \tag{1.12}$$

If we continue to reorange (1.12) we will get:

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left[\left(f_t f_{t-K} - \tilde{f}_t f_{t-K}\right) \right] \tag{1.13}$$

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left(f_t f_{t-K}\right) + 2\sum_{t=t_s}^{t_e} \left(\tilde{f}_t f_{t-K}\right) \tag{1.14}$$

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left(f_t f_{t-K}\right) + 2\sum_{t=t_s}^{t_e} \sum_{k=1}^{p} \left(a_k f_{t-k} f_{t-K}\right) \tag{1.15}$$

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left(f_t f_{t-K}\right) + 2\sum_{k=1}^{p} \sum_{t=t_s}^{t_e} \left(a_k f_{t-k} f_{t-K}\right) \tag{1.16}$$

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left(f_t f_{t-K}\right) + 2\sum_{k=1}^{p} a_k \sum_{t=t_s}^{t_e} \left(f_{t-k} f_{t-K}\right) \tag{1.17}$$

$$\frac{\partial E}{\partial a_K} = -2\sum_{t=t_s}^{t_e} \left(f_{t-K} f_t\right) + 2\sum_{k=1}^{p} a_k \sum_{t=t_s}^{t_e} \left(f_{t-K} f_{t-k}\right) \tag{1.18}$$

Finaly, to find LPCs for which error measure E willl be minimal, (1.18) is set to be equal to zero, which yields:

$$-2\sum_{t=t_s}^{t_e} \left(f_{t-K} f_t\right) + 2\sum_{k=1}^{p} a_k \sum_{t=t_s}^{t_e} \left(f_{t-K} f_{t-k}\right) = 0 \tag{1.19}$$

$$\sum_{k=1}^{p} a_k \sum_{t=t_s}^{t_e} \left(f_{t-K} f_{t-k}\right) = \sum_{t=t_s}^{t_e} \left(f_{t-K} f_t\right) \tag{1.20}$$

1.2.1. Covariance Method

– Extra information about this method can be found in reference [1] on page 106.
– By introducing following variable, equation
(1.20) could be made simpler:

$$\Phi(K,k) = \sum_{t=t_s}^{t_e} \left(f_{t-K} f_{t-k} \right) \tag{1.21}$$

Inserting (1.21) into (1.20) we get:

$$\sum_{k=1}^{p} \left[a_k \Phi(K,k) \right] = \Phi(K,0) \quad ; \quad 1<=K<=p \tag{1.22}$$

Formula (1.20) in matrix form looks like this:

$$\begin{bmatrix} \Phi(1,1) & \Phi(1,2) & \cdots & \Phi(1,p) \\ \Phi(2,1) & \Phi(2,2) & \cdots & \Phi(2,p) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi(p,1) & \Phi(p,2) & & \Phi(p,p) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \Phi(1,0) \\ \Phi(2,0) \\ \vdots \\ \Phi(p,0) \end{bmatrix} \tag{1.23}$$

It is extremly important to notice that resulting covariance matrix is symetric. This is so because

$$\sum_{t=t_s}^{t_e} \left(f_{t-K} f_{t-k} \right) = \sum_{t=t_s}^{t_e} \left(f_{t-k} f_{t-K} \right) \tag{1.24}$$

which means that:

$$\Phi(K,k) = \Phi(k,K)$$

Because of this symetricity, formula (1.23) can be written as:

$$\begin{bmatrix} \Phi(1,1) & \Phi(1,2) & \cdots & \Phi(1,p) \\ \Phi(1,2) & \Phi(2,2) & \cdots & \Phi(2,p) \\ \vdots & \vdots & \ddots & \vdots \\ \Phi(1,p) & \Phi(2,p) & & \Phi(p,p) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} \Phi(1,0) \\ \Phi(2,0) \\ \vdots \\ \Phi(p,0) \end{bmatrix} \tag{1.25}$$

– It is even more important to notice that elements on the diagonal are not the same.
– For instance difference between Φ(1,1) and Φ(2,2) is in sumand $f_{t_e-1} f_{t_e-1}$ which exists in Φ(1,1) but not in Φ(2,2).
– This kind of matrix equation can be solved, but it would be much esear if we could make diagonal elements equal.
– This will be done by extending upper summation border in equation (1.21) as shown in the next chapter about autocorelation.

– Covariance matrix is symetric, with different elements on main diagonal as demonstrated with following example when p=4:

$$\begin{bmatrix} \Phi(1,1) & \Phi(1,2) & \Phi(1,3) & \Phi(1,4) \\ \Phi(1,2) & \Phi(2,2) & \Phi(2,3) & \Phi(2,4) \\ \Phi(1,3) & \Phi(2,3) & \Phi(3,3) & \Phi(3,4) \\ \Phi(1,4) & \Phi(2,4) & \Phi(3,4) & \Phi(4,4) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} \Phi(1,0) \\ \Phi(2,0) \\ \Phi(3,0) \\ \Phi(4,0) \end{bmatrix} \tag{1.26}$$

– Before getting into details about autocorelation method use following picture to get beter understanging of equation (1.21):

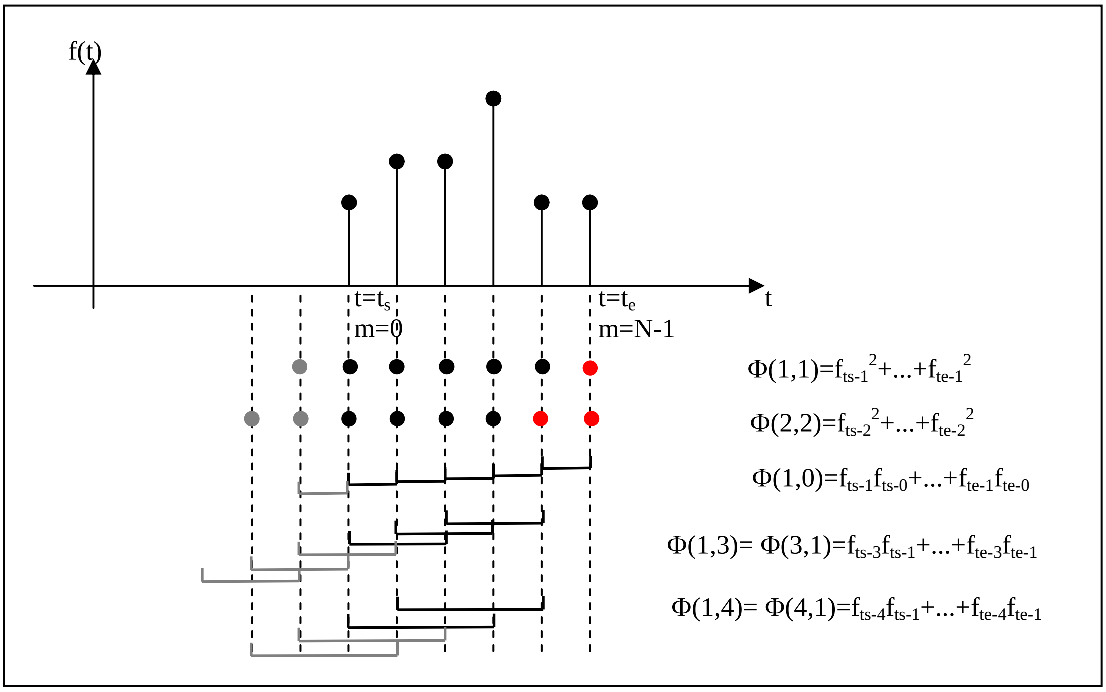

Figure 1.1

– Gray dots define values which are zero because both of the multipliers are zero.
– Gray lines also define values which are zero but this time because only one of the multipliers is zero.
– Red dots represent values which are not being calculated since they are not in the defined sumation range.
– These dots are the reason why values Φ(1,1), Φ(2,2) are not equal.
– These values are elements on the diagonal of the covariance matrix as shown before.

– If this approach of explaining equation (1.21) wasn't very helpful, go to the next chapter.
– It begins with another interpretation of that equation.

1.2.2. Autocorelation Method

– Maybe even better explanation of equation (1.21) is shown in following picture.

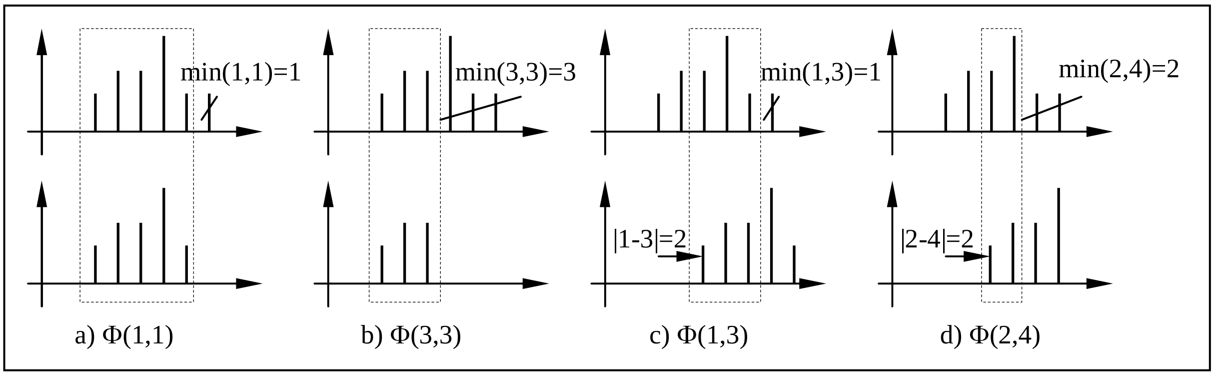

Figure 1.2

– To calculate Φ(K,k) first take the smaller value between K and k, that is min(K,k).
– Take that much points from the right of original signal.
– Copy resulting signal and shift it ti the right by absolute difference between K and k, that is |K-k|.
– Now simply multiply values which are on the same positions of those two signals as defind with (1.21).

– Figures 1.2 a) and b) clearly show why diagonal elements of (1.25) are different.
– Although there is no shifting of signal each diagonal element takes different amount of signal values while being calculated.
– This can be avoided simply by extending upper summation border in equation (1.21) by p points.
– In such case calculating any Φ(K,k) would use all of the values of original signal.
– In such case figure 1.2 would get rtansformed into following figure:

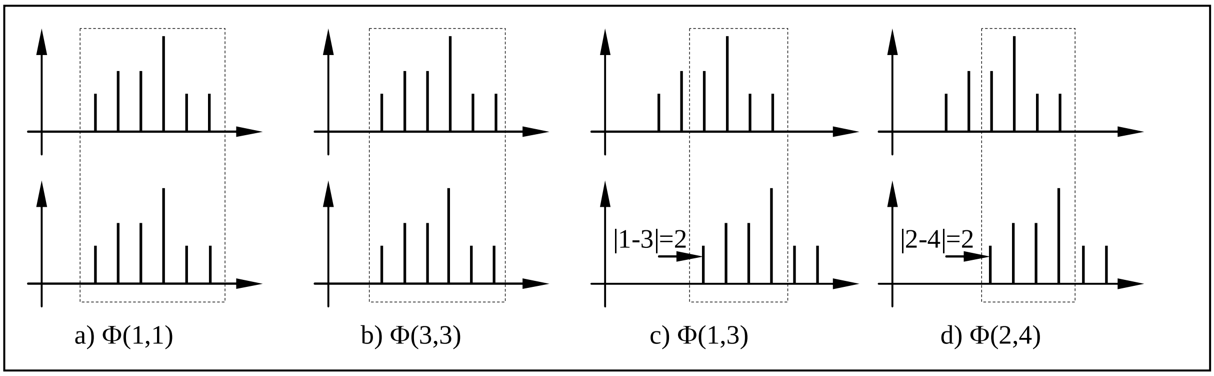

Figure 1.3

– Figure 1.3 demonstrates that value of Φ(K,k) now only depends on |K-k|.
– It also demonstrates that value of each Φ(K,k) equals to autocorelation parameter obtained by shifting signal by |K-k| steps.
– This means that (1.25) can now be written as:

$$\begin{bmatrix} r(0) & r(1) & \cdots & r(p-1) \\ r(1) & r(0) & \cdots & r(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(p-1) & r(p-2) & & r(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} r(1) \\ r(2) \\ \vdots \\ r(p) \end{bmatrix} \tag{1.27}$$

– All elements on each diagonal are the same as demonstrated with following example when number of LPCs p=4:

$$\begin{bmatrix} r_0 & r_1 & r_2 & r_3 \\ r_1 & r_0 & r_1 & r_2 \\ r_2 & r_1 & r_0 & r_1 \\ r_3 & r_2 & r_1 & r_0 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ a_4 \end{bmatrix} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{bmatrix} \tag{1.28}$$

– Now goes detailed mathematical explanation of transforming covariance matrix (1.25) into autocorelation matrix (1.27).
– Extra information about this method can be found in reference [1] on page 103.
– If we change the borders over which the estimation error is calculated like this:

$$E = \sum_{t=t_s}^{t_e+p} \left(f_t - \tilde{f}_t\right)^2 \tag{1.29}$$

which leads to:

$$\Phi(K,k) = \sum_{t=t_s}^{t_e+p} \left(f_{t-K} f_{t-k}\right) \tag{1.30}$$

then, folowing the prodecure from the previous chapter, we can get a matrix which is not only symetric but also has equal elements on the diagonal. Problem with expanding borders like this is that we will always have error for points [t_s+1,t_s+p] since these points are zero in weighted original signal, but are being aproximated using points from weighted originalsignal which are not zero, which means that estimated value will be non-zero.

Formula (1.30) can be transformd using folowing substitution:

We want that t-K could be presented as folows:

$$t\text{-}K=t_s+m \tag{1.31}$$

From (1.31) we have:

$$t=t_s+m+K \tag{1.32}$$

$$m= t\text{-}t_s\text{-}K \tag{1.33}$$

Using (1.32) folowing expressions can be calculated:

$$t\text{-}k=t_s+m+K\text{-}k \tag{1.34}$$

From (1.33) we can calculate values for m when t is t_s and t_e:

$$t=t_s \Rightarrow m= t_s\text{-}t_s\text{-}K=\text{-}K \tag{1.35}$$

$$t=t_e+p \Rightarrow m= t_e+p\text{-}t_s\text{-}K=t_e\text{-}t_s\text{-}K+p=N\text{-}1\text{-}K+p \tag{1.36}$$

Using (1.31), (1.34), (1.35) and (1.36) formula (1.30) can be written as folows:

$$\Phi(K,k) = \sum_{m=-K}^{N-1-K+p} \left(f_{t_s+m} f_{t_s+m+K-k}\right) \tag{1.37}$$

Since f_{t_s+m} is zero for all m<0 sumation can start with m=0, and therefore (1.37) can be written as folows:

$$\Phi(K,k) = \sum_{m=0}^{N-1-K+p} \left(f_{t_s+m} f_{t_s+m+K-k}\right) \tag{1.38}$$

– Equation (1.38) can be more easily understood by using folowing picture.

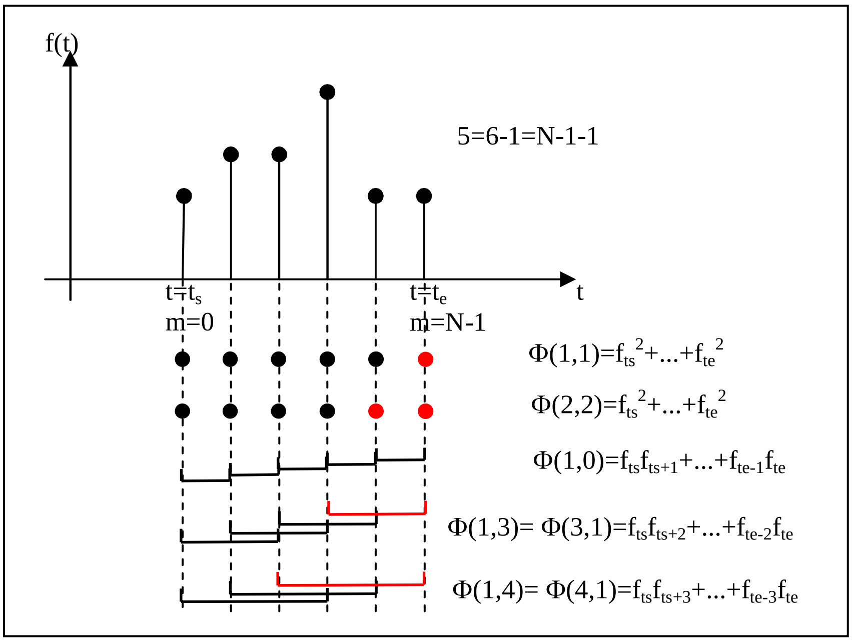

Figure 1.4

– Red parts represent values which were not calculated while original sumation borders were in place.
– Upper sumation limit can be reduced like this:

$$\Phi(K,k) = \sum_{m=0}^{N-1-|K-k|} \left(f_{t_s+m} f_{t_s+m+|K-k|} \right) \tag{1.39}$$

Autocorelation coefficient r_i is calculated by shifting signal by i steps, miltiplying such two signals and adding all results. This is exactly what we have defined in formula (1.39). Using autocorelation symbol, equation (1.39) can be written as folows:

$$r(|K-k|) = \sum_{m=0}^{N-1-|K-k|} \left(f_{t_s+m} f_{t_s+m+|K-k|} \right) \tag{1.40}$$

$$r(i) = \sum_{m=0}^{N-1-i} \left(f_{t_s+m} f_{t_s+m+i} \right) \quad ; \quad 0 \le i \le p \tag{1.41}$$

We can now write matrix equation like this:

$$\begin{bmatrix} r(0) & r(1) & \cdots & r(p-1) \\ r(1) & r(0) & \cdots & r(p-2) \\ \vdots & \vdots & \ddots & \vdots \\ r(p-1) & r(p-2) & & r(0) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} r(1) \\ r(2) \\ \vdots \\ r(p) \end{bmatrix} \tag{1.42}$$

– Equation (1.42) enables us to calculate LPC coefficients ($a_1,...,a_p$) from autocorelation coefficients (r(1),...,r(p)).
– Matrix (1.42) is Toeplitz matrix because each descending diagonal from left to right is constant as shown in example (1.28).
– Matrix equation (1.42) can be solved using Durbin algorithm.