1 Reflection Coefficients

Reflection coefficients (RCs) are related to the Levinson-Durbin algorithm used to solve the augmented Wiener-Hopf equations for a prediction error filter. For reasons of simplicity only the forward prediction error filter will be used in this worksheet.

1.1 Basics

The matrix formulation of the Levinson-Durbin algorithm can be expressed as follows:

$$\mathbf{a}_m = \begin{bmatrix} \mathbf{a}_{m-1} \\ 0 \end{bmatrix} + \kappa_m \begin{bmatrix} 0 \\ \mathbf{a}_{m-1}^{B*} \end{bmatrix} \tag{1.1}$$

where $\mathbf{a}_m$ are the weights of the forward prediction error filter with the subscript denoting the order m. $\mathbf{a}_m$ is made up of the tap weights from the Wiener filter, i.e.

$$\mathbf{a}_m = [1 \ -\omega_{m,1} \ -\omega_{m,2} \ \dots \ -\omega_{m,m}]^T$$

The superscript $B*$ in Equation 1.1 denotes the complex conjugated weights of the corresponding backward prediction errors filter. κ_m is refered to as the reflection coefficient and is recursively defined too.

$$\kappa_m = \frac{-\Delta_{m-1}}{P_{m-1}} \tag{1.2}$$

As can be seen from Equation 1.2 κ_m is a scalar made up of the the scaler quantities Δ_{m-1} and P_{m-1}. The scalar Δ_{m-1} can be interpreted in two ways;

- as a cross-correlation between the forward prediction error $f_{m-1}(n)$ and the unit delayed backward prediction error $b_{m-1}(n-1)$
- or as a series of multiplications between $\mathbf{a}$ and the autocorrelation sequence of the input, $r_{ii}(\tau)$.

Both ways of interpretation is given in Equation 1.3

$$\begin{aligned}\Delta_{m-1} &= E\left\{b_{m-1}(n-1)f^*_{m-1}(n)\right\} \\ &= \sum_{k=0}^{m-1} r_{ii}(k-m)a_{m-1,k}\end{aligned} \tag{1.3}$$

Where the forward and backward prediction error filter denoted $f_{m-1}(n)$ and $b_{m-1}(n-1)$ respectively, is defined as the output of the transversal filter with tap weights $\mathbf{a}_{m-1}$ to an input sequence $u(n)$, i.e.

$$\begin{aligned}f_{m-1}(n) &= \sum_{k=0}^{m-1} \mathbf{a}^*_{m-1,k} \cdot u(n-k) \\ b_{m-1}(n-1) &= \sum_{k=0}^{m-1} \mathbf{a}_{m-1,(m-1)-k} \cdot u(n-(k+1))\end{aligned} \tag{1.4}$$

The scalar P_{m-1} in Equation 1.2 corresponds to the power of a forward prediction error filter of order *m-1* and is given as:

$$P_{m-1} = E\left\{|f_{m-1}(n)|^2\right\} \tag{1.5}$$

As Δ_{m-1} in Equation 1.3 can never exceed the forward prediction error given by Equation 1.5, it follows that the reflection coefficient κ is bounded between -1 and 1.

By the recursive use of Equations 1.1, 1.2, 1.3 and 1.5 the Levinson-Durbin algorithm offers both the reflection coefficients, the prediction error power and the filter weights sequence $\mathbf{a}$. The initial conditions are $P_0 = r(0)$ and $\Delta_0 = r^*(1)$. Furthermore it is worth noting that $a_{m,0} = 1 \forall m$ and $a_{m,k} = 0 \forall k > m$. If κ_m is found, Equation 1.1 readily determines the corresponding weights. The relation between the reflection coefficients and the tap weights makes the reflection coefficients a possible way a describing the LPC coefficients.

1.2 Alternative Representations of κ

There are various ways to present the reflection coefficients. An often used presentation is partial correlation coefficients, abbreviated PARCOR, and defined by Equation 1.6

$$\begin{aligned}PARCOR &= \frac{E\left\{b_{m-1}(n-1)f^*_{m-1}(n)\right\}}{\left(E\left\{|b_{m-1}(n-1)|^2\right\}E\left\{|f_{m-1}(n)|^2\right\}\right)^{1/2}} \\ &= \frac{\Delta_{m-1}}{\left(E\left\{|b_{m-1}(n-1)|^2\right\}P_{m-1}\right)^{1/2}}\end{aligned} \tag{1.6}$$

Under the assumption of wide-sense stationarity, the forward prediction error power P_{m-1} is equal to the backward prediction error power $E\left\{|b_{m-1}(n-1)|^2\right\}$, hence Equation 1.6 simplifies to

$$PARCOR = \frac{\Delta_{m-1}}{\sqrt{(P_{m-1})^2}} = -\kappa_m \tag{1.7}$$

Thus PARCOR is simply the negative of the reflection coefficients given in the previous section under WSS conditions. Both representations are subject to poor quantization properties when the magnitude of κ approach unity. Therefore more appropiate representations have been investigated.

The Log Area Ratio (LAR) is one way of obtaining a more robust representation of κ. The LAR transforms a given reflection coefficient by the transformation stated below.

$$LAR = \frac{1}{2} \cdot \log\left(\frac{1+\kappa}{1-\kappa}\right) = \operatorname{arctanh}(\kappa) \tag{1.8}$$

Another useful representation is by the Inverse Sine parameters, which transforms κ according to Equation 1.9.

$$IS = \frac{2}{\pi} \cdot \arcsin(\kappa) \tag{1.9}$$

Both transformations warps the amplitude scale for values of κ near unity to avoid the high sensivity towards quantization. Figure 1.1 on the following page depicts the LAR and IS parameters as a function of reflection coefficients in the interval]-1;1[. Notice that the IS representation is still bounded by -1 and 1.

Matlab can easily convert LPC coefficients into both reflection coefficients, LAR or IS and vice versa. Table 1.1 displays the commands. The table is read row-wise, e.g. to get from RC to LAR type rc2lar(RC).

-	**LPC**	**RC**	**LAR**	**IS**
LPC	-	poly2rc(**a**)	poly2lar(**a**)	poly2is(**a**)
RC	rc2poly(κ)	-	rc2lar(κ)	rc2is(κ)
LAR	lar2poly(LAR)	lar2rc(LAR)	-	lar2is(LAR)
IS	is2poly(IS)	is2rc(IS)	is2lar(IS)	-

Table 1.1: Matlab commands.

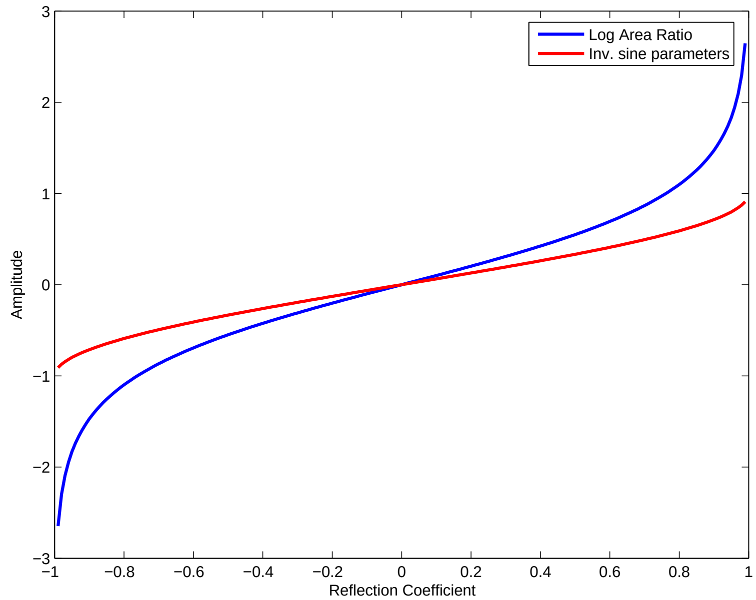

Figure 1.1: Log Area Ratios and Inverse Sine parameters as a function of reflection coefficients.

1.3 Remark on κ and LSP

The two polynomials P(z) and Q(z) used in the representation of the Line Spectrum Frequencies are related to the reflection coefficients. Equation 1.1 on page 1 can be expressed as a polynomium A(z) in terms of the variable z as follows:

$$A(z) + \kappa_m z^{-(p+1)} A(z^{-1}) = \begin{cases} P(z) \; ; \kappa_m = 1 \\ Q(z) \; ; \kappa_m = -1 \end{cases} \tag{1.10}$$

If the reflection coefficient is either 1 or -1, the Levinson-Durbin algorithm yields the symmetric and antisymmetric polynomials used in connection with the LSF. The worksheet on Line Spectrum Pairs/Frequencies is in Chapter **??** on page ??.