Notes on the Levinson Algorithm

Given is
See the notes on llse. At this moment we assume that we work on a time invariant, zero mean process, that we know the covariance data and that we restrict ourselves to second order statistics and least squares estimation. In practice, the estimator will work on actual data. How this is done is explained in the next section.

The Levinson idea
Normally the inversion of a system of equations requires $O(n^3)$ operations, where $n$ is the number of equations. In the time invariant case we assume here (over one frame), the covariance matrix has a special structure, it is Toeplitz, meaning that elements on NE-SW diagonals are equal. This structural property can be exploited to obtain a more efficient inversion algorithm, namely one that requires only $O(n^2)$ operations. This will be achieved by the Levinson algorithm. It will turn out that the special structure allows the matrix inversion by solving just two equations instead of $n$.

Some immediate properties
The Levinson equations are written in a somewhat special form, with a '1' in the first position of $A_p(z)$ and a positive number in the first position of the right hand vector, with a similar property for $B_p(z)$ w.r. the last position. There is in general no guarantee that such solutions exist, but here they will if the covariance matrix is strictly positive definite. The special way of writing (which is historical!) has the merit that there is a special interpretation for both $F_p$ and $R_p$ - they are variances of llse prediction errors or innovations, called respectively the forward and the reverse innovation. The forward predictor estimates $X[n]$ from $X[n-1],\cdots,X[n-p]$ with innovation $X[n] - \hat{X}_p[n]$ while the backward predictor estimates $X[n-p]$ from $X[n],\cdots,X[n-p+1]$ with appropriate innovation (we don’t give it a name since we are not going to use it later). The positivity of $F_p$ and $R_p$ is of course immediate from their physical meaning, but it follows also directly from the positive definiteness of the matrix $K_p$.

The Levinson recursion
The Levinson idea is to use the $p^{th}$ order forward and backward Levinson estimation vectors as a basis to construct those of order $p+1$. This can be done if the backward vector $B_p$ is shifted one (time) notch down. Because the entries in the covariance matrix repeat themselves along main diagonals, the same matrix is found one notch down those diagonals, which results in the pattern of zeros shown on the slide.
The reflection coefficient

The Levinson game is played by making the matrix \[ \begin{bmatrix} F_p & E_p \\ D_p & R_p \end{bmatrix} \] diagonal through the application of a column transformation (matrix to the right). The same transformation will hold on the Levinson vectors. Since we want to keep the standard '1' entries in the Levinson vectors, the 1,1 and 2,2 entries in the transformation matrix have to be '1' as well. This leaves the 1,2 and 2,1 entries to be defined. The entries will be each other's conjugates, and they will be necessarily of magnitude less that one (still to be shown!). For reasons that will become clear later (the scattering interpretation), they are called 'reflection coefficients'.

Proofs of properties

The Levinson vectors appear to be each other's reflection. Also, the forward and backward reflection coefficients are equal. The nice symmetry properties do not carry over to the block matrix case, however, that is when the process is not scalar but consists of stochastic vectors.

The normalized case

Here we modify the Levinson vectors slightly so that they become concordant with the definition of so called Szegö polynomials. \( F_p \) and \( R_p \) are covariances of innovations, in particular: \( F_p = \mathbb{E}(\{X[n] - \hat{X}_p[n]\}^2) \). The normalized innovation 
\[
\varepsilon_p[n] = \frac{1}{\sqrt{\mathbb{E}(X[n] - \hat{X}_p[n])}} (X[n] - \hat{X}_p[n])
\]
has unit covariance. Dividing both members of the Levinson equation with \( \sqrt{F_p} = \sqrt{R_p} \) produces recursions for the normalized innovations.

Normalized recursion

If we want to use the Levinson recursion for the normalized case we must adapt it. This is achieved by multiplying it with the coefficient \( \frac{1}{\sqrt{1 - |\rho_{p+1}|^2}} \), in fact,
\[
\frac{1}{\sqrt{1 - |\rho_{p+1}|^2}} = \frac{\sqrt{F_p}}{\sqrt{F_{p+1}}} \] (de-normalize order p and renormalize order p+1, check!). The transformation matrix obtained has a very special property, it is a so called 'complex hyperbolic transformation' - we will explore its properties in detail in the section on scattering.

The Szegö polynomials

We now convert the normalized Levinson vectors to a polynomial form. They are called 'Szegö polynomials in honor of the mathematician who discovered them and their remarkable properties. The normalized Levinson recursion produces a recursion for the polynomials when translated to polynomial form.
One further remark: the Szegö polynomials are normally defined in mathematics with a 'z' as running variable, engineers use 'z^{-1}'. The classical theory is of course recovered by replacing the engineer's $z^{-1}$ with $z$. If $z$ is interpreted as a complex variable, then the transformation maps the unit circle to outside and vice versa.

**Szegö ladder filter**

The recursion can also be interpreted as filter, with signal flow diagram as shown on the slide. Some attention has to be devoted to the initiation of the recursion. The zero\textsuperscript{th} order polynomials have to be solutions of the zero\textsuperscript{th} order equations

\[
\begin{align*}
\kappa_0 a_{0,0} &= \frac{1}{a_{0,0}} \\
\kappa_0 b_{0,0} &= \frac{1}{b_{0,0}}
\end{align*}
\]

leading to $a_{0,0} = b_{0,0} = \kappa_0^{-1/2}$.

In the next section we derive a method to construct the Levinson-Szegö recursion in an even more efficient and numerically better way, it is called the 'Schur algorithm'.