

Laboratory 5

Prediction with Lattices

SPW does not have a Lattice filter, so we have to build one. This requires the building of stages, as well as the representation of an algorithm for updating the reflection coefficients. The stochastic gradient algorithm (equation (4.5.3) in the Text) is used for the updates.

A. SET UP

1. Copy `mirchandani-lplattice3.system`, `lupdate1.detail` and `lattice.detail` into your directory. Evaluate the whole algorithm and make sure that it is consistent with what you expect. Understand the Estimator (SPW) which is used (there are three of them) and how they are used. The “C” coefficient which is added to the estimate is probably one of the more significant factors in establishing convergence.
2. Generate the second order filter,

$$\frac{1}{1 + 0.35z^{-1} - 0.3z^{-2}}$$

and generate a second order AR-process by passing a 8 khz sampled, gaussian distributed white noise through it.

3. While we need only a two-stage Lattice to whiten this process, use the 3-stage that you have. Calculate values of reflection coefficients k_1 , k_2 and k_3 that you should expect, given the transversal filter coefficients that you are using. (Equations 4.3.8 and 4.3.10 in the Text.)

B. SIMULATIONS:

Simulate for 20000 iterations, using the exponential Estimator for signal power. Use different values of the “C” coefficient, from 10 to 1000. Each time that you wish to make changes to the Lattice sections, it is easier to edit the `lupdate1.detail`, make your changes and save (resave) it, along with creating a new symbol (which is done automatically when you resave). Then when you call in (edit) `lplattice.system`, *all* the lattice sections will be updated.

You may wish to run your transversal filter estimator (3rd order) and compare.

C. HAND-IN

1. Output from SDE showing the convergence of the lattice filter coefficients.
2. Comments on what you tried and results you got.