

**332:525 - Optimum Signal Processing
Computer Experiment 10 - Due May 5, 2011**

It is desired to design an adaptive Wiener filter to enhance a sinusoidal signal buried in noise. The noisy sinusoidal signal is given by

$$x_n = s_n + v_n, \quad \text{where } s_n = \sin(\omega_0 n)$$

with $\omega_0 = 0.075\pi$. The noise v_n is related to the secondary signal y_n by

$$v_n = y_n + y_{n-1} + y_{n-2} + y_{n-3} + y_{n-4} + y_{n-5} + y_{n-6}$$

The signal y_n is assumed to be an order-4 AR process with reflection coefficients:

$$\{\gamma_1, \gamma_2, \gamma_3, \gamma_4\} = \{0.5, -0.5, 0.5, -0.5\}$$

As in an earlier experiment, the variance σ_ϵ^2 of the driving white noise of the model must be chosen in such a way as to make the variance σ_v^2 of the noise component v_n approximately one.

- a. For a Wiener filter of order $M = 6$, determine the theoretical direct-form Wiener filter coefficient vector:

$$\mathbf{h} = [h_0, h_1, \dots, h_6]$$

for estimating x_n (or, rather v_n) from y_n . Determine also the theoretical lattice/ladder realization coefficients:

$$\boldsymbol{\gamma} = [\gamma_1, \gamma_2, \dots, \gamma_6], \quad \mathbf{g} = [g_0, g_1, \dots, g_6]$$

- b. Generate input pairs $\{x_n, y_n\}$ (making sure that the transients introduced by the modeling filter have died out), and filter them through the LMS algorithm to generate the filter output pairs $\{\hat{x}_n, e_n\}$. On the same graph, plot e_n together with the desired signal s_n .

Plot also a few of the adaptive filter coefficients such as $h_4(n)$, $h_5(n)$, and $h_6(n)$. Observe their convergence to the theoretical Wiener solution.

You must generate enough input pairs in order to achieve convergence of the LMS algorithm and observe the steady-state converged output of the filter.

Experiment with the choice of the adaptation parameter μ . Start by determining λ_{\max} , λ_{\min} , the eigenvalue spread $\lambda_{\max}/\lambda_{\min}$ of R and the corresponding time constant.

- c. Repeat (b), using the gradient lattice adaptive filter. Plot all of the adaptive reflection coefficients $\gamma_p(n)$ versus n , and a few of the ladder coefficients, such as $g_4(n)$, $g_5(n)$, and definitely $g_6(n)$.

(Because theoretically $g_6 = h_6$ (why?), plotting $h_6(n)$ and $g_6(n)$ will let you compare the convergence speeds of the LMS and lattice adaptive filters.)

You must experiment with a couple of values of λ (use $\beta = 1$). You must work of course with exactly the same set of input pairs as in part (b).

- d. Next, we change this experiment into a non-stationary one. Suppose the total number of input pairs that you used in parts (b) and (c) is N . And suppose that at time $n = N$, the input statistics changes suddenly so that the primary signal is given now by the model:

$$x_n = s_n + v_n, \quad \text{where } v_n = y_n + y_{n-1} + y_{n-2} + y_{n-3}$$

and y_n changes from a fourth-order AR model to a second-order model with reflection coefficients (use the same σ_e^2 as before):

$$\{\gamma_1, \gamma_2\} = \{0.5, -0.5\}$$

Repeat parts (a,b,c), keeping the filter order the same, $M = 6$. Use $2N$ input pairs, such that the first N follow the original statistics and the second N follow the changed statistics. Compare the capability of the LMS and lattice adaptive filters in tracking such changes.

Here, the values of μ for the LMS case and λ for the lattice case, will make more of a difference in balancing the requirements of learning speed and quality of estimates.

- e. Finally, feel free to “tweak” the statements of all of the above parts as well as the definition of the models in order to show more clearly and more dramatically the issues involved, namely, LMS versus lattice, learning speed versus quality, and the effect of the adaptation parameters, eigenvalue spread, and time constants. One other thing to notice in this experiment is that, while the adaptive weights tend to fluctuate a lot as they converge, the actual filter outputs \hat{x}_n, e_n behave better and are closer to what one might expect.