Wiener-Hopf Solution

Adaptive Linear Combiner

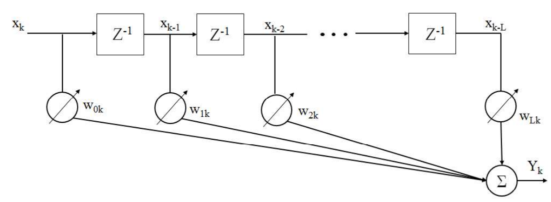


Figure: Single-input adaptive linear combiner with bias weight w_{0k}

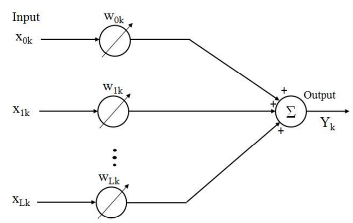


Figure: Multiple-input adaptive linear combiner with bias weight w_{0k}

Desired Response and Error

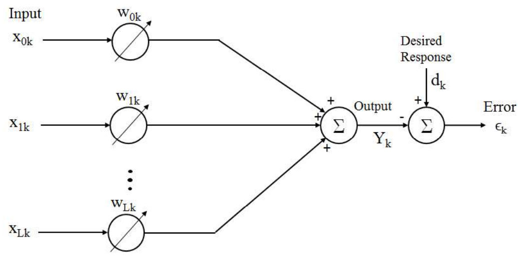


Figure: Multiple-input adaptive linear combiner with desired response and error signals.

Performance Function

$$\epsilon_k = d_k - y_k \tag{1}$$

$$\epsilon_k = d_k - \mathbf{X}_k^T\mathbf{W} = d_k - \mathbf{W}^T\mathbf{X}_k \tag{2}$$

Desired Response and Error

Instantaneous Squared Error

$$\epsilon_k = d_k - \mathbf{W}^T\mathbf{X}_k \tag{3}$$

$$\epsilon_k^2 = d_k^2 - \mathbf{W}^T\mathbf{X}_k\mathbf{X}_k^T\mathbf{W} - 2d_k\mathbf{X}_k^T\mathbf{W} \tag{4}$$

Expected Error Squared Equation

$$E\left[\epsilon_k^2\right] = E\left[d_k^2\right] + \mathbf{W}^T E\left[\mathbf{X}_k\mathbf{X}_k^T\right]\mathbf{W} - 2E\left[d_k\mathbf{X}_k^T\right]\mathbf{W} \tag{5}$$

$$\xi = E\left[d_k^2\right] + \mathbf{W}^T\mathbf{R}\mathbf{W} - 2\mathbf{P}^T\mathbf{W} \tag{6}$$

where, $\mathbf{R} = E\left[\mathbf{X}_k\mathbf{X}_k^T\right]$ is a input correlation matrix and $\mathbf{P} = E\left[d_k\mathbf{X}_k^T\right]$ is a cross correlation between the desired response and input components.

Gradient of Mean Square Error Performance

Gradient of Mean Square Error Performance

$$\xi = E\left[d_k^2\right] + \mathbf{W}^T\mathbf{R}\mathbf{W} - 2\mathbf{P}^T\mathbf{W} \tag{7}$$

$$\nabla \cong \frac{\partial \xi}{\partial \mathbf{W}} = 2\mathbf{R}\mathbf{W} - 2\mathbf{P} \tag{8}$$

Wiener-Hopf Equation

Set **W** at its optimal value $\mathbf{W}^*$,

$$\nabla = 2\mathbf{R}\mathbf{W}^* - 2\mathbf{P} \tag{9}$$

$$\mathbf{W}^* = \mathbf{R}^{-1}\mathbf{P} \tag{10}$$

Minimum Mean Square Error Performance

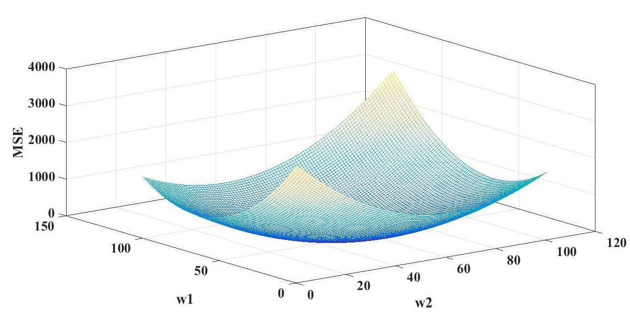


Figure: Performance surface of mean-square error

$$\xi_{min} = E\left[d_k^2\right] + \mathbf{W}^{*T}\mathbf{R}\mathbf{W}^* - 2\mathbf{P}^T\mathbf{W}^* \quad (From\ Eq.6) \tag{11}$$

$$\xi_{min} = E\left[d_k^2\right] + \left[\mathbf{R}^{-1}\mathbf{P}\right]^T\mathbf{R}\mathbf{R}^{-1}\mathbf{P} - 2\mathbf{P}^T\mathbf{R}^{-1}\mathbf{P} = E\left[d_k^2\right] - \mathbf{P}^T\mathbf{R}^{-1}\mathbf{P} \tag{12}$$

Least Mean Square (LMS) Algorithm

Least Mean Square (LMS) Algorithm

To develop adaptive algorithm, estimate the gradient of ξ such that,

$$\hat{\nabla}_k = \begin{bmatrix} \frac{\partial \epsilon_k^2}{\partial w_0} \\ \vdots \\ \frac{\partial \epsilon_k^2}{\partial w_L} \end{bmatrix} = 2\epsilon_k \begin{bmatrix} \frac{\partial \epsilon_k}{\partial w_0} \\ \vdots \\ \frac{\partial \epsilon_k}{\partial w_L} \end{bmatrix} \tag{13}$$

From Eq. 2, the error signal from the combiner output, Y_k, as a linear combination of the input samples is given as,

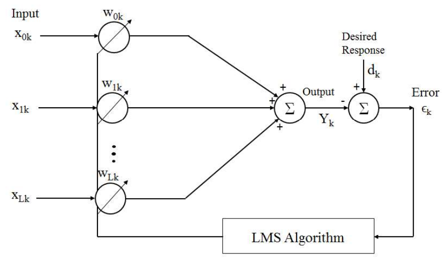
$$\epsilon_k = d_k - \mathbf{X}_k^T \mathbf{W} \tag{14}$$

$$\frac{\partial \epsilon_k}{\partial \mathbf{W}} = 0 - \mathbf{X}_k \tag{15}$$

Therefore,

$$\hat{\nabla}_k = -2\epsilon_k \mathbf{X}_k \tag{16}$$

Formulation of Least Mean Square (LMS) Equation



According to steepest-descent adaptive algorithm,

$$\mathbf{W}_{k+1} = \mathbf{W}_k - \mu \hat{\nabla}_k \tag{17}$$

LMS Algorithm

$$\mathbf{W}_{k+1} = \mathbf{W}_k + 2\mu\epsilon_k \mathbf{X}_k \tag{18}$$

References

1. Alexander Schrijver, Theory of linear and integer programming. John Wiley and Sons, 1998.
2. George Bernard Dantzig. Linear programming and extensions. Vol. 48. Princeton university press, 1998.
3. Jiri Matousek and Bernd Gärtner, Understanding and using linear programming. Springer Science and Business Media, 2007.
4. Bernad Widrow, S. D. Stearns, Adaptive Signal Processing, Prentice-Hall. Englewood Cliffs, NJ. 1985.
5. S. S. Haykin, Adaptive filter theory. Pearson Education India, 2005.

The End