Chapter 5 Least-mean-square Adaptive Filters

5.2 Least-mean-square Adaptation Algorithm

According to Eq. (4.8), the gradient vector at iteration n, is

$$\nabla J(n) = -2\vec{p} + 2\tilde{R}\vec{w}(n).$$

The simplest gradient vector estimate is the instantaneous estimates for $\tilde{R}$ and $\vec{p}$ based on the sample values of the tap-input vector and desired response, defined by

$$\hat{\tilde{R}}(n) = \vec{u}(n)\vec{u}^H(n)$$

$$\hat{\vec{p}}(n) = \vec{u}(n)d^*(n)$$

The instantaneous estimate of the gradient vector is

$$\hat{\nabla} J(n) = -2\vec{u}(n)d^*(n) + 2\vec{u}(n)\vec{u}^H(n)\hat{\vec{w}}(n)$$

Therefore, the recursive update of the weight vector is given by

$$\hat{\vec{w}}(n+1) = \hat{\vec{w}}(n) + \mu\, \vec{u}(n) \left[d^*(n) - \underbrace{\vec{u}^H(n)\, \hat{\vec{w}}(n)}_{y(n)} \right]$$

$$= \hat{\vec{w}}(n) + \mu\, \vec{u}(n)\, e^*(n),$$

where $e(n) = d(n) - y(n)$,

and $0 < \mu < \dfrac{2}{M S_{max}}$, ($M$ is the filter length

S_{max} is the maximum value of the input power spectral density)

This LMS algorithm can be applied for both deterministic and random signals $u(n)$ and $d(n)$. It can track the nonstationary variations as well.

Read 5.3 Applications Yourselves!

5.4 Statistical LMS Theory

As discussed in Chapter 4, the weight-error vector in the LMS filter is given by

$\vec{\varepsilon}(n) = \vec{w}_o - \hat{\vec{w}}(n)$, where $\vec{w}_o$ is the optimal tap-weight vector.

According to Eq. (5.5),

$$\vec{\varepsilon}(n+1) = \left[\tilde{I} - \mu \vec{u}(n) \vec{u}^H(n)\right] \vec{\varepsilon}(n) - \mu \vec{u}(n) e_o^*(n),$$

where $\tilde{I}$ is the identity matrix and

$e_o(n) = d(n) - \vec{w}_o^H \vec{u}(n)$ is the estimation error produced by the optimum filter.

Equation (5.56) is a random difference equation. To study the convergence behavior of such a random algorithm in an average sense, we may invoke the direct-averaging method.

According to this method, the solution of Eq. (5.56), with a small step-size μ, is approximated by the ensemble average,

$$E\{\tilde{I} - \mu\, \vec{u}(n)\, \vec{u}^H(n)\} = \tilde{I} - \mu \tilde{R}.$$

Hence, $\vec{\varepsilon}_0(n+1) = (\tilde{I} - \mu\tilde{R})\, \vec{\varepsilon}_0(n) - \mu\, \vec{u}(n)\, e_0^*(n)$,

where $\vec{\varepsilon}_0(n) \approx \vec{\varepsilon}(n)$ for small μ.

※ Butterweck's Iterative Procedure

The error vector in Eq. (5.56) can be written as a sum of iterative partial error vectors such that

$$\vec{\varepsilon}(n) = \vec{\varepsilon}_0(n) + \vec{\varepsilon}_1(n) + \vec{\varepsilon}_2(n) + \cdots,$$

where $\vec{\varepsilon}_0(n)$ is the error-vector in Eq. (5.58) for $\mu \to 0$. $\vec{\varepsilon}_1(n), \vec{\varepsilon}_2(n), \ldots$ are higher order corrections for $\mu > 0$.

The zero-mean difference matrix is defined as $\tilde{P}(n) = \vec{u}(n)\, \vec{u}^H(n) - \tilde{R}$.

From Eq. (5.59), (5.60) and (5.56), we obtain

$$\vec{\varepsilon}_0(n+1) + \vec{\varepsilon}_1(n+1) + \vec{\varepsilon}_2(n+2) + \cdots$$

$$= (\tilde{I} - \mu \tilde{R})\,[\vec{\varepsilon}_0(n) + \vec{\varepsilon}_1(n) + \vec{\varepsilon}_2(n) + \cdots]$$

$$- \mu \tilde{P}(n)\,[\vec{\varepsilon}_0(n) + \vec{\varepsilon}_1(n) + \vec{\varepsilon}_2(n) + \cdots]$$

$$- \mu \vec{u}(n)\, e_0^*(n)$$

Hence we can readily deduce the set of coupled difference equations:

$$\vec{\varepsilon}_i(n+1) = (\tilde{I} - \mu \hat{R})\,\vec{\varepsilon}_i(n) + \vec{f}_i(n), \quad i = 0, 1, 2, \cdots,$$

where the driving force $\vec{f}_i(n)$ is defined as

$$\vec{f}_i(n) = \begin{cases} -\mu \vec{u}(n)\, e_0^*(n), & i = 0 \\ -\mu \tilde{P}(n)\, \vec{\varepsilon}_{i-1}(n), & i = 1, 2, \cdots \end{cases}$$

Eq. (5.61) states that the analysis of the LMS filter is reduced to a study of transmission of a stationary random process through a low-pass filter with an extremely low cut-off frequency as $\mu \to 0$.

The correlation matrix of the weight-error vector $\vec{\varepsilon}(n)$ is defined as

$$\tilde{K}(n) = E\{\vec{\varepsilon}(n)\,\vec{\varepsilon}^H(n)\}$$

$$= \sum_i \sum_k E\{\vec{\varepsilon}_i(n)\,\vec{\varepsilon}_k^H(n)\}, \quad (i,k)=0,1,2\cdots$$

$$\tilde{K}(n) = \tilde{K}_0(n) + \mu\,\tilde{K}_1(n) + \mu^2\,\tilde{K}_2(n) + \cdots,$$

$$\text{where } \mu^j\tilde{K}_j(n) = \begin{cases} E[\vec{\varepsilon}_0(n)\,\vec{\varepsilon}_0^H(n)], & \text{for } j=0 \\ \sum_i \sum_k E[\vec{\varepsilon}_i(n)\,\vec{\varepsilon}_k^H(n)], & \text{for } (i,k)\ge 0 \text{ such that } i+k = 2j-1,\ 2j \end{cases}$$

※ Small-step-size Statistical Theory

Assumption I. The step-size μ is small, so the LMS filter acts as a low-pass filter with a low cut-off frequency

From Eq. (5.58), the k^{th} element in $\tilde{V}^H\vec{\varepsilon}(n)$ can be written as

$$\varepsilon'_{0,k}(n+1) = (1-\mu\,\sigma_{u,k}^2)\,\varepsilon'_{0,k}(n) + f'_0(n)$$

where $\vec{V}$ is the eigen vector matrix of $\tilde{R}$

The zero of the low-pass filter specified in the above difference equation is

$$z = (1 - \mu \sigma_{u,k}^2) \to 1 \quad \text{as } \mu \to 0.$$

Assumption II. The physical mechanism for generating the observable data, i.e., the desired response is described by a linear multiple regression model that is matched exactly by the Wiener filter; that is

$$d(n) = \vec{w}_o^H \vec{u}(n) + e_o(n),$$

where the irreducible estimation error $e_o(n)$ is a white process independent of the input $\vec{u}(n)$.

$$E\{e_o(n)\, e_o^*(n-k)\} = \begin{cases} J_{min}, & \text{for } k=0 \\ 0, & \text{for } k \neq 0 \end{cases}$$

Assumption III. The input vector $\vec{u}(n)$ and the desired response $d(n)$ are jointly Gaussian.

※ Natural Modes of the LMS filter

Under Assumption I, Butterweck's iterative procedure reduces to the following equations:

$$\vec{\varepsilon}_0(n+1) = (\tilde{I} - \mu\tilde{R})\,\vec{\varepsilon}_0(n) + \vec{f}_0(n)$$

$$\vec{f}_0(n) = -\mu\,\vec{u}(n)\,e_0^*(n)$$

(one-zero model of error vector)

Suppose $\tilde{R} = \tilde{Q}\tilde{\Lambda}\tilde{Q}^H$ (eigen-decomposition)

We define
$$\vec{v}(n) = \tilde{Q}^H\vec{\varepsilon}_0(n);$$

then
$$\vec{v}(n+1) = (\tilde{I} - \mu\tilde{\Lambda})\,\vec{v}(n) + \vec{\phi}(n),$$

where $\vec{\phi}(n) = \tilde{Q}^H\vec{f}_0(n)$.

We can easily verify that:

1. The mean of the random force vector $\vec{\phi}(n)$ is zero: $E[\vec{\phi}(n)] = \vec{0}$, $\forall n$

2. The correlation matrix of the random force vector $\vec{\phi}(n)$ is diagonal matrix:

$$E\left[\vec{\phi}(n)\,\vec{\phi}(n)^H\right] = \mu^2 J_{min}\,\tilde{\Lambda},$$

where J_{min} is the minimum mean-square error produced by the Wiener filter.

Proof:

1. From Eq. (5.69), $\vec{\phi}(n) = \tilde{Q}^H \vec{f_0}(n)$

$$\simeq -\mu\,\tilde{Q}^H \vec{u}(n)\,e_0^*(n)$$

$$E\left[\vec{\phi}(n)\right] = -\mu\,\tilde{Q}^H E\left[\vec{u}(n)\,e_0^*(n)\right]$$

$$= \vec{0}$$

($E\left[\vec{u}(n)\,e_0^*(n)\right] = \vec{0}$) due to the principle of the orthogonality.

2. $$E\left[\vec{\phi}(n)\,\vec{\phi}(n)^H\right] = \mu^2 \tilde{Q}^H E\left[\vec{u}(n)\,e_0^*(n)\,e_0(n)\,\vec{u}^H(n)\right]\tilde{Q}$$

$$= \mu^2 \tilde{Q}^H \underbrace{E\left[|e_0(n)|^2\right] E\left[\vec{u}(n)\,\vec{u}^H(n)\right]}_{\text{Assumption II}} \tilde{Q}$$

$$= \mu^2 \tilde{Q}^H J_{min}\,\tilde{R}\,\tilde{Q}$$

$$= \mu^2 J_{min}\,\tilde{\Lambda}$$

According to Eq. (5.73), the k^{th} natural mode (the k^{th} element in $\vec{v}(n)$) is written as

$$v_k(n+1) = (1-\mu\lambda_k)\, v_k(n) + \phi_k(n),$$
$$k = 1, 2, \cdots, M.$$

The iterative change in the natural mode v_k is

$$\Delta v_k(n) \equiv v_k(n+1) - v_k(n)$$
$$= -\mu\lambda_k v_k(n) + \phi_k(n), \quad k = 1, 2, \cdots, M$$

From Eq. (5.78),

$$v_k(n) = (1-\mu\lambda_k)^n v_k(0) + \sum_{i=0}^{n-1} (1-\mu\lambda_k)^{n-1-i} \phi_k(i)$$

※ Properties of the natural mode $v_k(n)$, $k = 1, 2, \cdots, M$

1. Mean value:

$$E\{v_k(n)\} = v_k(0)\,(1-\mu\lambda_k)^n.$$

2. The mean-square value:

$$E\{|v_k(n)|^2\} = \frac{\mu J_{min}}{2-\mu\lambda_k} + (1-\mu\lambda_k)^{2n}\left(|v_k(0)|^2 - \frac{\mu J_{min}}{\quad}\right)$$

※ Learning Curves

Two kinds of learning curves regarding LMS algorithm can be identified:

1. The mean-square error (MSE) learning curve, which is based on ensemble averaging of the squared estimation error $|e(n)|^2$. This learning curve is thus a plot of the MSE,

$$J(n) = E\{|e(n)|^2\}$$

2. The mean-square deviation (MSD) learning curve, which is based on ensemble averaging of the squared error deviation $\|\vec{\varepsilon}(n)\|^2$. This learning curve is thus a plot of the MSD,

$$D(n) = E\{\|\vec{\varepsilon}(n)\|^2\}$$

L-2 norm

The estimation error is

$$e(n) = d(n) - \hat{\vec{w}}^H(n)\,\vec{u}(n)$$

$$= d(n) - \vec{w}_o^H\,\vec{u}(n) + \vec{\varepsilon}^H(n)\,\vec{u}(n)$$

$$= e_o(n) + \vec{\varepsilon}^H(n)\,\vec{u}(n)$$

$$\approx e_o(n) + \vec{\varepsilon}_o^H(n)\,\vec{u}(n) \quad \text{for small } \mu.$$

Hence, $J(n) = E\{|e(n)|^2\}$

$$= E\left\{ (e_o(n) + \vec{\varepsilon}_o^H(n)\,\vec{u}(n))(e_o^*(n) + \vec{u}^H(n)\,\vec{\varepsilon}_o(n)) \right\}$$

$$= J_{min} + 2\,\mathrm{Re}\left\{ E\left[e_o^*(n)\,\vec{\varepsilon}_o^H(n) \times \vec{u}(n) \right] \right\} + E\left\{ \vec{\varepsilon}_o^H(n)\,\vec{u}(n)\,\vec{u}^H(n)\,\vec{\varepsilon}_o(n) \right\}.$$

Under Assumption II, $e_o(n)$ is statistically independent of $\vec{u}(n)$ and $\vec{\varepsilon}_o(n)$, so $E[\,e_o^*(n)\,\vec{\varepsilon}_o^H(n)\,\vec{u}(n)]$

$$= E[\,e_o^*(n)]\,E[\,\vec{\varepsilon}_o^H(n)\,u(n)] = 0.$$

Under Assumption I, the variations of the vector $\vec{\varepsilon}_0(n)$ are slow compared with those of the input $\vec{u}(n)$,

Hence $E[\vec{\varepsilon}_0^H(n)\,\vec{u}(n)\,\vec{u}^H(n)\,\vec{\varepsilon}_0(n)]$

$$\approx E\{\vec{\varepsilon}_0^H(n)\,E[\vec{u}(n)\,\vec{u}^H(n)]\,\vec{\varepsilon}_0(n)\}$$
$$= E[\vec{\varepsilon}_0^H(n)\,\tilde{R}\,\vec{\varepsilon}_0(n)]$$
$$= tr\{E[\vec{\varepsilon}_0^H(n)\,\tilde{R}\,\vec{\varepsilon}_0(n)]\}$$
$$= E\{tr[\vec{\varepsilon}_0^H(n)\,\tilde{R}\,\vec{\varepsilon}_0(n)]\}$$
$$= E\{tr[\tilde{R}\,\vec{\varepsilon}_0(n)\,\vec{\varepsilon}_0^H(n)]\}$$
$$= tr\{E[\tilde{R}\,\vec{\varepsilon}_0(n)\,\vec{\varepsilon}_0^H(n)]\}$$
$$= tr\{\tilde{R}\,\underbrace{E[\vec{\varepsilon}_0(n)\,\vec{\varepsilon}_0^H(n)]}_{\tilde{K}_0(n)}\}$$
$$= tr[\tilde{R}\,\tilde{K}_0(n)]$$

Therefore, $J(n) = J_{min} + tr[\tilde{R}\,\tilde{K}_0(n)]$

The excess mean-square error can be written as

$$J_{ex}(n) = J(n) - J_{min}$$
$$\approx tr[\tilde{R}\,\tilde{K}_0(n)]$$

$$J_{ex}(n) \approx tr\{\tilde{R}\, E[\vec{\varepsilon}_o(n)\, \vec{\varepsilon}_o^{\,H}(n)]\}$$

$$= tr\{\tilde{R}\, E[\tilde{Q}\, \vec{V}(n)\, \vec{V}^H(n)\, \tilde{Q}^H]\}$$

$$= E\{tr[\tilde{R}\, \tilde{Q}\, \vec{V}(n)\, \vec{V}^H(n)\, \tilde{Q}^H]\}$$

$$= E\{tr[\vec{V}^H(n)\, \tilde{Q}^H \tilde{R}\, \tilde{Q}\, \vec{V}(n)]\}$$

$$= E\{tr[\vec{V}^H(n)\, \tilde{\Lambda}\, \vec{V}(n)]\}$$

$$= \sum_{k=1}^{M} \lambda_k\, E[\,|V_k(n)|^2\,]$$

According to Assumption I, Eq. (5.71) and Eq. (5.84)

$$D(n) \approx E[\,\|\vec{\varepsilon}_o(n)\|^2\,]$$

$$= E[\,\|\vec{V}(n)\|^2\,]$$

$$= \sum_{k=1}^{M} E[\,|V_k(n)|^2\,]$$

If we denote λ_{min} and λ_{max} as the minimum and maximum eigenvalues of $\tilde{R}$,

$$\lambda_{min} \le \lambda_k \le \lambda_{max}, \quad k = 1, 2, \ldots, M$$

From Eq. (5.91) and (5.92), the MSD can be bounded by

$$\lambda_{min}\, D(n) \le J_{ex}(n) \le \lambda_{max}\, D(n), \quad \forall n$$

Or $\dfrac{J_{ex}(n)}{\lambda_{min}} \geq D(n) \geq \dfrac{J_{ex}(n)}{\lambda_{max}}, \quad \forall n.$

X. Transient Behavior and Convergence Considerations

According to Eq. (5.82), (5.90), (5.91), we have

$$J(n) = J_{min} + \mu J_{min} \sum_{k=1}^{M} \frac{\lambda_k}{2-\mu\lambda_k} + \sum_{k=1}^{M} \lambda_k \left(|v_k(0)|^2 - \frac{\mu J_{min}}{2-\mu\lambda_k} \right)(1-\mu\lambda_k)^{2n}$$

$$\approx J_{min} + \frac{\mu J_{min}}{2} \sum_{k=1}^{M} \lambda_k + \sum_{k=1}^{M} \lambda_k \left(|v_k(0)|^2 - \frac{\mu J_{min}}{2} \right)(1-\mu\lambda_k)^{2n},$$

for small μ.

$$\lim_{n\to\infty} J(n) = J(\infty) = J_{min} + \mu J_{min} \sum_{k=1}^{M} \frac{\lambda_k}{2-\mu\lambda_k}$$

$$\approx J_{min} + \frac{\mu J_{min}}{2} \sum_{k=1}^{M} \lambda_k, \text{ for small } \mu.$$

$$\neq J_{min}$$

X. Misadjustment

A new parameter, the misadjustment can be defined as

$$\mathcal{M} \equiv \frac{J_{ex}(\infty)}{J_{min}}$$

$$= \frac{J(\infty) - J_{min}}{J_{min}}$$

$$= \frac{\mu}{2} \sum_{k=1}^{M} \lambda_k ,$$

where $\mathcal{M}$ is dimensionless and it provides a measure of how close the LMS algorithm to the optimality in the mean square sense

$$\mathcal{M} = \frac{\mu}{2} \, tr[\tilde{R}]$$

$$= \frac{\mu}{2} M r(0)$$

$$= \frac{\mu}{2} E[\|\vec{u}(n)\|^2]$$

$= \frac{\mu}{2}$ X (total tap-input power)

X Average Time Constant

The average eigenvalue of $\tilde{R}$ is defined as

$$\lambda_{av} = \frac{1}{M}\sum_{k=1}^{M} \lambda_k .$$

Suppose that the ensemble - average learning curve of LMS algorithm is approximated by a single exponential with time constant $\tau_{mse,av}$. According to Eq (4.32), the average time constant for LMS algorithm is defined as

$$\tau_{mse,av} = \frac{1}{2\mu\lambda_{av}}$$

According to Eq. (5.100), (5.103) & (5.104), the misadjustment can be approximated by

$$\mathcal{M} = \frac{\mu M \lambda_{av}}{2} = \frac{M}{4\tau_{mse,av}}$$

※ Remarks :

1. The misadjustment $\mathcal{M}$ increases linearly with the filter length M for a fixed $\tau_{mse,av}$.

2. The setting time of the LMS algorithm (i.e, the time taken for the transients to die out) is proportional to $\tau_{mse,av}$ and is inversely proportional to μ. Therefore there is a trade-off, in that if μ is reduced so as to reduce the misadjustment, then the setting time (time to converge) of the LMS algorithm is increased. Conversely, if μ is increased so as to reduce the setting time (faster convergence), then the misadjustment is increased.

Read 5.6 ~ 5.9 Yourselves !!