Chapter 2
Classical Adaptation Algorithms

2.1 Least Mean-Squares (LMS) Algorithm

We review two classical adaptation algorithms in this chapter. The first one is the *least-mean-squares (LMS) algorithm*. In this algorithm, the coefficient vector $w_k \in \mathbb{R}^n$ is updated as

$$\begin{aligned} w_{k+1} &= w_k + \mu \Delta w_k, \\ \Delta w_k &= e(k) x_k, \end{aligned} \tag{2.1}$$

where

$$x_k \stackrel{\triangle}{=} (x(k), x(k-1), \ldots, x(k-(n-1)))^t$$

is the signal vector,

$$e(k) \stackrel{\triangle}{=} d(k) - z(k) = d(k) - \langle x_k,\ w_k \rangle$$

is the error signal, and μ is a parameter called the *step-size*. The integer n is the filter length throughout the book. The step-size can be time varying as $\mu(k)$. We will treat such a *variable step-size* case in the last chapter. The expression $\langle x_k,\ w_k \rangle \stackrel{\triangle}{=} x_k^t w_k$ means the inner product of x_k and w_k. The signal vector is also called a *regressor*.

How to time-index the coefficient vector is a matter of convention. In the literature, we sometimes encounter a formulation like

$$\begin{aligned} w_k &= w_{k-1} + \mu \Delta w_{k-1}, \\ \Delta w_{k-1} &= e(k) x_k, \\ e(k) &= d(k) - \langle x_k,\ w_{k-1} \rangle \end{aligned} \tag{2.2}$$

K. Ozeki, *Theory of Affine Projection Algorithms for Adaptive Filtering*,
Mathematics for Industry 22, DOI 10.1007/978-4-431-55738-8_2

instead of (2.1). However, (2.1) and (2.2) are essentially the same. The important point is that the time-index of the coefficient vector is increased by 1 after it is updated, showing that the update is performed sample-by-sample.

The motivation of the LMS algorithm can be explained by an argument based on the steepest-descent method (Appendix 1: "Steepest-Descent Method"). Let us assume that $x(k)$ and $d(k)$ are realizations of random variables $\boldsymbol{x}(k)$ and $\boldsymbol{d}(k)$, respectively. Then, $z(k)$ and $e(k)$ are also realizations of random variables defined by

$$\begin{aligned} \boldsymbol{z}(k) &\stackrel{\triangle}{=} \langle \boldsymbol{x}_k,\ w_k \rangle = \boldsymbol{x}_k^t w_k, \\ \boldsymbol{e}(k) &\stackrel{\triangle}{=} \boldsymbol{d}(k) - \boldsymbol{z}(k), \end{aligned}$$

where $\boldsymbol{x}_k = (\boldsymbol{x}(k), \boldsymbol{x}(k-1), \ldots, \boldsymbol{x}(k-(n-1)))^t$. In this stochastic framework, we can consider the mean-square error between $\boldsymbol{d}(k)$ and $\boldsymbol{z}(k)$:

$$\mathrm{E}\left\{\boldsymbol{e}^2(k)\right\} = \mathrm{E}\left\{(\boldsymbol{d}(k) - \boldsymbol{z}(k))^2\right\} = \mathrm{E}\left\{(\boldsymbol{d}(k) - \boldsymbol{x}_k^t w_k)^2\right\}. \tag{2.3}$$

We differentiate (2.3) with respect to w_k to obtain the gradient of the mean-square error at w_k:

$$\begin{aligned} \nabla_{w_k} \mathrm{E}\left\{\boldsymbol{e}^2(k)\right\} &= \nabla_{w_k} \mathrm{E}\left\{\boldsymbol{d}^2(k) - 2\boldsymbol{d}(k)\boldsymbol{x}_k^t w_k + (\boldsymbol{x}_k^t w_k)^2\right\} \\ &= \nabla_{w_k}\left(\mathrm{E}\left\{\boldsymbol{d}^2(k)\right\} - 2\mathrm{E}\{\boldsymbol{d}(k)\boldsymbol{x}_k^t\} w_k + w_k^t \mathrm{E}\{\boldsymbol{x}_k \boldsymbol{x}_k^t\} w_k\right) \\ &= -2\mathrm{E}\{\boldsymbol{d}(k)\boldsymbol{x}_k\} + 2\mathrm{E}\{\boldsymbol{x}_k \boldsymbol{x}_k^t\} w_k \\ &= -2(\mathrm{E}\{\boldsymbol{d}(k)\boldsymbol{x}_k\} - \mathrm{E}\{\boldsymbol{x}_k \boldsymbol{x}_k^t\} w_k). \end{aligned}$$

However, the expectations $\mathrm{E}\{\boldsymbol{d}(k)\boldsymbol{x}_k\}$ and $\mathrm{E}\{\boldsymbol{x}_k \boldsymbol{x}_k^t\}$ are not known. Therefore, we employ *instantaneous approximation*: we replace $\mathrm{E}\{\boldsymbol{d}(k)\boldsymbol{x}_k\}$ with the realization $d(k)x_k$, and $\mathrm{E}\{\boldsymbol{x}_k \boldsymbol{x}_k^t\}$ with $x_k x_k^t$. Then, the gradient is approximated by

$$\begin{aligned} \nabla_{w_k} \mathrm{E}\left\{\boldsymbol{e}^2(k)\right\} &\approx -2(d(k)x_k - x_k x_k^t w_k) \\ &= -2x_k(d(k) - x_k^t w_k) \\ &= -2x_k(d(k) - z(k)) \\ &= -2e(k)x_k. \end{aligned}$$

Since a small change of w_k along the direction $-\nabla_{w_k} \mathrm{E}\left\{\boldsymbol{e}^2(k)\right\}$ decreases $\mathrm{E}\left\{\boldsymbol{e}^2(k)\right\}$, if we replace w_k with $w_{k+1} = w_k + \mu e(k)x_k$ for a small positive μ, then $\mathrm{E}\left\{\boldsymbol{e}^2(k+1)\right\}$ will be smaller than $\mathrm{E}\left\{\boldsymbol{e}^2(k)\right\}$. This is just the LMS algorithm (2.1). The origin of the name "least-mean-squares (LMS)" algorithm will be apparent. The LMS algorithm is also called the *Widrow-Hoff algorithm* after its originators [1]. The geometrical meaning of the LMS algorithm will become clearer in the next section.

2.2 Normalized LMS (NLMS) Algorithm

Let us consider the system identification problem in Fig. 2.1. We assume that the unknown system is represented by a transversal filter with the coefficient vector $w^o = (w_0^o, w_1^o, \ldots, w_{n-1}^o)^t$. We also assume $\{v(k)\} = 0$[1] throughout this chapter. This is a more specific version of Fig. 1.2. In the LMS algorithm, the effective magnitude of the step-size μ depends on the volume of the input signal as explained below. If the input signal $\{x(k)\}$ is multiplied by a scalar a, then other signals are also multiplied by a. Let the new signals be denoted by

$$
\begin{aligned}
\tilde{x}(k) &\stackrel{\triangle}{=} ax(k), \\
\tilde{x}_k &\stackrel{\triangle}{=} (\tilde{x}(k), \tilde{x}(k-1), \ldots, \tilde{x}(k-(n-1)))^t \\
&= a(x(k), x(k-1), \ldots, x(k-(n-1)))^t \\
&= ax_k, \\
\tilde{y}(k) &\stackrel{\triangle}{=} \langle \tilde{x}_k, w^o \rangle = a\langle x_k, w^o \rangle = ay(k), \\
\tilde{d}(k) &\stackrel{\triangle}{=} \tilde{y}(k) = ay(k) = ad(k), \\
\tilde{z}(k) &\stackrel{\triangle}{=} \langle \tilde{x}_k,\ w_k \rangle = \langle ax_k,\ w_k \rangle = a\langle x_k,\ w_k \rangle = az(k), \\
\tilde{e}(k) &\stackrel{\triangle}{=} \tilde{d}(k) - \tilde{z}(k) = ad(k) - az(k) = ae(k).
\end{aligned}
$$

For the new signals, $\mu \Delta w_k$ can be written as

$$
\begin{aligned}
\mu \Delta w_k &= \mu \tilde{e}(k) \tilde{x}_k \\
&= (a^2 \mu) e(k) x_k. \qquad (2.4)
\end{aligned}
$$

Thus, multiplying a scalar a to $\{x(k)\}$ has the same effect as multiplying a^2 to the step-size μ. This phenomenon makes it difficult to set an optimal value for μ: just

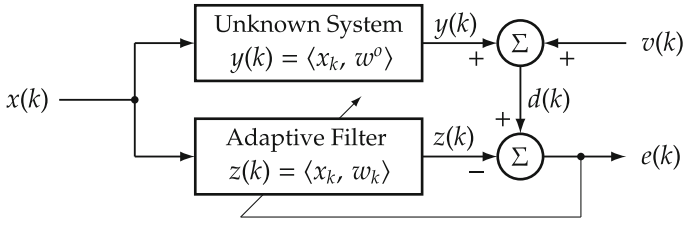

Fig. 2.1 Identification of unknown linear system

[1] This means $v(k) = 0$ for $k \geq 0$.

changing the gain of the amplifier for the input signal affects the effective value of μ.

In the *normalized LMS (NLMS) algorithm*, the coefficient vector is updated as

$$\begin{aligned} w_{k+1} &= w_k + \mu \Delta w_k, \\ \Delta w_k &= e(k) \frac{x_k}{\|x_k\|^2}. \end{aligned} \tag{2.5}$$

Note that Δw_k is normalized by $\|x_k\|^2$. It is obvious from the above arguments that Δw_k is invariant under multiplication of a scalar to the input signal $\{x(k)\}$. Thus, in the NLMS algorithm, μ has a definite meaning that is independent of the volume of the signal $\{x(k)\}$.

The NLMS algorithm can be interpreted from a geometrical point of view. If $\mu = 1$, then as shown in Fig. 2.2, w_{k+1} is the affine projection (Chap. 3, Appendix 1: "Affine Projection") of w_k onto Π_k: $w_{k+1} = P_{\Pi_k} w_k$, where Π_k is the hyperplane defined by

$$\Pi_k \stackrel{\triangle}{=} \{w;\ d(k) - \langle x_k,\ w \rangle = 0\},$$

and P_{Π_k} denotes the affine projection onto Π_k. In fact,

$$\begin{aligned} \langle x_k,\ w_{k+1} \rangle &= x_k^t \left(w_k + e(k) \frac{x_k}{\|x_k\|^2} \right) \\ &= x_k^t w_k + e(k) \\ &= d(k). \end{aligned}$$

This shows $w_{k+1} \in \Pi_k$. Also, since x_k is orthogonal to Π_k by Theorem 3.12,

$$w_{k+1} - w_k = \frac{e(k)}{\|x_k\|^2} x_k$$

is orthogonal to Π_k.

If $\mu \neq 1$, w_{k+1} is a point somewhere on the straight line connecting w_k and $P_{\Pi_k} w_k$, the location of which depends on μ as illustrated in Fig. 2.2. In fact,

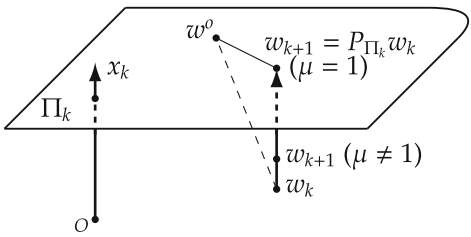

Fig. 2.2 Geometrical interpretation of the NLMS algorithm. If $\{v(k)\} = 0$, $w^o \in \Pi_k$

$$\begin{aligned} w_{k+1} - w_k &= \mu \Delta w_k \\ &= \mu (w_k + \Delta w_k - w_k) \\ &= \mu (P_{\Pi_k} w_k - w_k). \end{aligned}$$

Thus, the step-size μ determines the ratio

$$\frac{\|w_{k+1} - w_k\|}{\|P_{\Pi_k} w_k - w_k\|} = |\mu|. \tag{2.6}$$

Since the hyperplane Π_k is invariant under the change of the volume of the input signal $\{x(k)\}$, the vector w_{k+1} is also invariant. In the LMS algorithm, this is not true. Although w_{k+1} is also a point on the straight line connecting w_k and $P_{\Pi_k} w_k$, the location depends not only on μ but also on the volume of the signal $\{x(k)\}$ even if its waveform is kept unchanged.

Under the present assumption that $\{v(k)\} = 0$, the coefficient vector w^o of the unknown system satisfies $d(k) - \langle x_k,\ w^o \rangle = 0$. Thus, w^o is an element of Π_k as shown in Fig. 2.2.

Theorem 2.1 *If* $0 < \mu < 2$, *then* $\|w_{k+1} - w^o\| \leq \|w_k - w^o\|$. *If* $\mu \leq 0$ *or* $\mu \geq 2$, *then* $\|w_{k+1} - w^o\| \geq \|w_k - w^o\|$.

Proof In the triangle $\{w^o,\ w_k,\ P_{\Pi_k} w_k\}$, the vectors $(w^o - P_{\Pi_k} w_k)$ and $(P_{\Pi_k} w_k - w_k)$ are orthogonal as shown in Fig. 2.2. Therefore, by the Pythagorean theorem,

$$\begin{aligned} \|w^o - w_k\|^2 &= \|w^o - P_{\Pi_k} w_k\|^2 + \|P_{\Pi_k} w_k - w_k\|^2 \\ &= \|w^o - P_{\Pi_k} w_k\|^2 + \|\Delta w_k\|^2. \end{aligned} \tag{2.7}$$

Also, in the triangle $\{w^o,\ w_{k+1},\ P_{\Pi_k} w_k\}$, the vectors $(w^o - P_{\Pi_k} w_k)$ and $(P_{\Pi_k} w_k - w_{k+1})$ are orthogonal. Therefore,

$$\begin{aligned} \|w^o - w_{k+1}\|^2 &= \|w^o - P_{\Pi_k} w_k\|^2 + \|P_{\Pi_k} w_k - w_{k+1}\|^2 \\ &= \|w^o - P_{\Pi_k} w_k\|^2 + \|(1 - \mu) \Delta w_k\|^2 \\ &= \|w^o - P_{\Pi_k} w_k\|^2 + (1 - \mu)^2 \|\Delta w_k\|^2. \end{aligned} \tag{2.8}$$

From (2.7) and (2.8), we have

$$\|w^o - w_k\|^2 - \|w^o - w_{k+1}\|^2 = \mu (2 - \mu) \|\Delta w_k\|^2.$$

Since

$$\mu (2 - \mu) \begin{cases} > 0, & \text{if } \ 0 < \mu < 2; \\ \leq 0, & \text{otherwise}, \end{cases}$$

we have proved the theorem. □

This theorem guarantees that if $0 < \mu < 2$, then $\|w^o - w_k\|^2$ decreases monotonously. Since a bounded, monotone sequence always converges, $\lim_{k\to\infty} \|w^o - w_k\|^2$ exists. However, this does not mean $\lim_{k\to\infty} w_k = w^o$.

For arbitrary μ, we have from (2.8)

$$\|w^o - P_{\Pi_k} w_k\| \le \|w^o - w_{k+1}\|,$$

with equality only if $\mu = 1$ unless $\Delta w_k = 0$. Therefore, $\mu = 1$ is the best choice for fast convergence. However, in the presence of the observation noise $\{v(k)\}$, we have to consider a trade-off between the convergence rate and the adaptation error. In such a case, $\mu = 1$ may not be the best choice.

In the above argument, we see that the meaning of the step-size μ in the NLMS algorithm is quite different from that in the LMS algorithm. In the LMS algorithm, μ is a small positive number for approximating differential with finite difference. In the NLMS algorithm on the other hand, μ is a parameter to prevent over-fitting to noisy data. In this sense, it may be more appropriate to call it the *relaxation factor* [2].

The progress from the LMS algorithm to the NLMS algorithm, also known as the *learning method* [3], is an important step toward the affine projection algorithm (APA) [2], the main subject of the subsequent chapters.

Appendix 1: Steepest-Descent Method

Let f be a real differentiable function defined on $\mathbb{R}^n$. The *steepest-descent method* searches for $\mathrm{argmin}_x f(x)$ by iterating the following computation starting from an initial point x_0:

$$x_{k+1} = x_k - \mu \nabla_x f(x_k), \tag{2.9}$$

where $\mu > 0$ is the step-size. The step-size may depend on the iteration index k as $\mu(k)$.

This algorithm is motivated by the following fact. Let $x(t) \in \mathbb{R}^n$ be a function of t, and consider a differential equation

$$\frac{\mathrm{d}x(t)}{\mathrm{d}t} = -\nabla_x f(x(t)). \tag{2.10}$$

Using the chain rule for differentiation and (2.10), we have

$$\begin{aligned}\frac{\mathrm{d}f(x(t))}{\mathrm{d}t} &= (\nabla_x f(x(t)))^t \frac{\mathrm{d}x(t)}{\mathrm{d}t} \\ &= -(\nabla_x f(x(t)))^t \nabla_x f(x(t)) \\ &= -\|\nabla_x f(x(t))\|^2 \\ &\le 0.\end{aligned}$$

with equality only if $\nabla_x f(x(t)) = 0$. This shows that if the curve $x(t)$ is a solution of (2.10), $f(x(t))$ is a decreasing function of t.

The differential equation (2.10) can be approximated by a difference equation as

$$\frac{x(t+\Delta t) - x(t)}{\Delta t} \approx -\nabla_x f(x(t)).$$

If we let $x_k \stackrel{\triangle}{=} x(t)$, $x_{k+1} \stackrel{\triangle}{=} x(t+\Delta t)$, and $\mu \stackrel{\triangle}{=} \Delta t$, we obtain (2.9). Under a certain condition, the vector sequence $x_0, x_1, x_2, \ldots$ converges to a local minimum point. If the initial point x_0 and the step-size μ are appropriately chosen, the sequence converges to $\mathrm{argmin}_x f(x)$.

References

1. Widrow, B., Hoff, M.E.Jr.: Adaptive switching circuits. IRE WESCON Conv. Rec. Pt.**4**, 96–104 (1960)
2. Ozeki, K., Umeda, T.: An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties. IEICE Trans. **J67-A**(2), 126–132 (1984) (Also in Electron. Commun. Jpn. 67-A(5), 19–27 (1984))
3. Nagumo, J., Noda, A.: A learning method for system identification. IEEE Trans. Autom. Control **AC-12**(3), 282–287 (1967)

http://www.springer.com/978-4-431-55737-1

Theory of Affine Projection Algorithms for Adaptive Filtering

Ozeki, K.

2016, XII, 223 p. 32 illus., Hardcover
ISBN: 978-4-431-55737-1