The Filtered-x LMS Algorithm

L. Håkansson
Department of Telecommunications and Signal Processing,
University of Karlskrona/Ronneby
372 25 Ronneby
Sweden
Adaptive filters are normally defined for problems such as electrical noise cancelling where the filter output is an estimate of a desired signal. In control applications, however, the adaptive filter works as a controller controlling a dynamic system containing actuators and amplifiers etc. The estimate (anti-vibrations or anti-sound) in this case can thus be seen as the output signal from a dynamic system, i.e. a forward path. Since there is a dynamic system between the filter output and the estimate, the selection of adaptive filter algorithms must be made with care. A conventional adaptive algorithm such as the LMS algorithm is likely to be unstable in this application due to the phase shift (delay) introduced by the forward path [1, 2]. The well-known filtered-x LMS-algorithm is, however, an adaptive filter algorithm which is suitable for active control applications [3]. It is developed from the LMS algorithm, where a model of the dynamic system between the filter output and the estimate, i.e. the forward path is introduced between the input signal and the algorithm for the adaptation of the coefficient vector [3, 4]. Figure 1 shows an active control system with a controller based on the filtered-x LMS algorithm. The

![Figure 1: Active control system.](image)

FIR filter output is as usual given by the vector inner product:

$$y(n) = \mathbf{w}^T(n) \mathbf{x}(n)$$  \hspace{1cm} (1)

where

$$\mathbf{x}(n) = [x(n), x(n-1), \ldots, x(n-M+1)]^T$$  \hspace{1cm} (2)

is the input signal vector to the adaptive filter and

$$\mathbf{w}(n) = [w_0(n), w_1(n), \ldots, w_{M-1}(n)]^T$$  \hspace{1cm} (3)

is the adjustable filter coefficient vector. In control applications, the estimation error $e(n)$ is defined by the difference between the desired signal (desired response) $d(n)$ and the output signal from the forward path or plant under control $y_C(n)$:

$$e(n) = d(n) - y_C(n)$$  \hspace{1cm} (4)

Assuming that the transfer function for the forward path can be described with an $I$th-order FIR filter:

$$h_C(n) = \begin{cases} 
  c_n & \text{when } n \in \{0, \ldots, I - 1\} \\
  0 & \text{otherwise}
\end{cases}$$  \hspace{1cm} (5)
it follows that the estimation error $e(n)$ can be expressed as:

$$e(n) = d(n) - \sum_{i=0}^{I-1} c_i \sum_{m=0}^{M-1} w_m(n - i)x(n - i - m)$$

(6)

The Wiener (Mean Square Error) solution of the coefficient vector is obtained by minimizing the quadratic function [3, 4]:

$$J_f(n) = E[e(n)^2]$$

(7)

and this can be carried out by using the gradient vector for the mean square error $J_f(n)$:

$$\nabla_{w(n)} J_f(n) = 2E[e(n)\nabla_{w(n)} e(n)]$$

(8)

By taking advantage of the fact that the desired signal $d(n)$ is independent of the filter coefficients and by assuming that $w_m(n), m \in \{0, \ldots, M - 1\}$ is time invariant, the gradient vector for the estimation error can be expressed as:

$$\nabla_{w(n)} e(n) = \begin{bmatrix}
- \sum_{i=0}^{I-1} c_i x(n - i) \\
- \sum_{i=0}^{I-1} c_i x(n - i - 1) \\
\vdots \\
- \sum_{i=0}^{I-1} c_i x(n - i - M + 1)
\end{bmatrix}$$

(9)

By inserting this expression in Eq. 8 we obtain the following relation for the gradient vector of the mean square error:

$$\nabla_{w(n)} J_f(n) = -2E[e(n)x_C(n)]$$

(10)

where $x_C(n)$ is given by:

$$x_C(n) = \begin{bmatrix}
\sum_{i=0}^{I-1} c_i x(n - i) \\
\sum_{i=0}^{I-1} c_i x(n - i - 1) \\
\vdots \\
\sum_{i=0}^{I-1} c_i x(n - i - M + 1)
\end{bmatrix}$$

(11)

In other words, an "LMS algorithm" with a gradient estimate as in:

$$\nabla_{w(n)} J_f^{*}(n) = -2e(n)x_C(n)$$

(12)

would solve the problem of producing an estimate via a dynamic system [3, 4]. From this it follows that the conventional LMS algorithm is likely to be unstable in control applications. The conventional LMS algorithm will in some cases also find a poor solution
when it converges [1, 3, 4]. This can be explained by the fact that the LMS algorithm
uses a gradient estimate $x(n)e(n)$ which is not correct in the mean [1].

A compensated algorithm is obtained by filtering the reference signal to the coefficient
adjustment algorithm using a model of the forward path. This is illustrated in Fig. 2.
The algorithm obtained is the well-known filtered-x LMS algorithm defined by:

$$y(n) = w^T(n)x(n) \quad (13)$$

$$e(n) = d(n) - y_C(n) \quad (14)$$

$$x_{C^*}(n) = \begin{bmatrix}
\sum_{i=0}^{I-1} c_i^* x(n - i) \\
\sum_{i=0}^{I-1} c_i^* x(n - i - 1) \\
\vdots \\
\sum_{i=0}^{I-1} c_i^* x(n - i - M + 1)
\end{bmatrix} \quad (15)$$

$$w(n + 1) = w(n) + \mu x_{C^*}(n)e(n) \quad (16)$$

Here $c_i^*$ is the coefficients of an estimated FIR filter model of the forward path:

$$h_{C^*}(n) = \begin{cases} 
c_i^* & \text{if } n \in \{0, \ldots, I-1\} \\
0 & \text{else}
\end{cases} \quad (17)$$

It is in practice customary to use an estimate of the impulse response for the forward
path. As a result, the reference signal $x_{C^*}(n)$ will be an approximaton, and differences
between the estimate of the forward path and the true forward path influence both the
stability properties and the convergence rate of the algorithm [1, 3, 4]. However, the
algorithm is robust to errors in the estimate of the forward path [1, 3, 4]. The model
used should introduce a time delay corresponding to the forward paths at the dominating
frequencies [1, 4].

Figure 2: Active control system with a controller based on the filtered-x LMS-algorithm[5].
In the case of narrow-band reference signals to the algorithm, e.g. \(\sin(\omega_0t)\), the algorithm will converge with phase errors in the estimate of the forward path with up to 90°, provided that the step length \(\mu\) is sufficiently small [6, 3]. Furthermore, phase errors in the estimate of the forward path smaller than 45° will have only a minor influence on the algorithm convergence rate [6].

The filtered-x LMS algorithm relies principally on the assumption that the adaptive FIR filter and the forward path “commute”. This is approximately true if the adaptive filter varies in a time scale which is slow in comparison with the time constant for the impulse response of the forward path. We can thus write as follows:

\[
I-1 \sum_{i=0}^{M-1} c_i \sum_{m=0}^{L-1} w_m(n-i)x(n-i-m) \approx \sum_{m=0}^{M-1} w_m(n) \sum_{i=0}^{I-1} c_ix(n-m-i)
\] (18)

when

\[
w(n) = w(n-i), i \in \{1, 2, \cdots, I-1\}
\] (19)

where \(I\) is the length of the impulse response of the forward path. In practice, the filtered-x LMS-algorithm exhibits stable behaviour even when the coefficients change within the time scale associated with the dynamic response of the forward path [3, 4]. In order to ensure that the action of the LMS algorithm is stable the maximum value for the step length \(\mu\) should be given approximately by:

\[
\mu < \frac{2}{ME[x^2(n)]}
\] (20)

However, in the case of the filtered-x LMS algorithm, Elliot et al. [7] have found that the maximum step length \(\mu\) not only depends on the length of the adaptive filter and filtered reference signal but also on the delays in the forward path \(C\). If the reference signal \(x_{C^*}(n)\) is a white noise process it has thus been found that an upper limit for the step length \(\mu\) is given by:

\[
\mu_{\text{max}} \approx \frac{2}{E[x^2_{C^*}(n)](M + \delta)}
\] (21)

where \(\delta\) is the overall delay in the forward path (in samples).

In the case of a non-white reference signal Elliot et al. [7] suggest that \(\frac{1}{\mu_{\text{max}}}\) is proportional to 1.2\(M\) and not 0.5\(M\). A probable explanation is that the covariance matrix for the reference signal will have a poor conditioning [7].
Bibliography


