Active Noise Cancellation by the modified Filtered X-LMS algorithm with online secondary path modeling

Nirav Desai
Assistant Professor, Department of ECE, ITM Universe, Vadodara, Gujarat

Abstract: **An application of the Least Mean Square Algorithm for active noise cancellation is presented here. Potential applications of the algorithm are in audio noise reduction and wireless signal jamming. Simulation results based on the filtered x-LMS algorithm are presented with the regular LMS algorithm modified to give a faster rate of convergence. Hardware experiments will be carried out eventually.**

Keywords: LMS, FXLMS, active noise cancellation

I. Introduction

In this paper, an active noise cancellation[3] scheme is presented which uses a noise microphone, error microphone and noise cancelling speakers programmed using a modified filtered x-LMS algorithm with online secondary path modeling. The system model is depicted in figure 1 below[1]. The noise from the primary noise source is captured using the Reference Noise Microphone. This signal is input to the active noise cancelling (ANC) system, which drives the noise cancelling loudspeaker. The ANC system estimates the noise signal and the loudspeaker transmits it's inverse to cancel the noise. The error signal is sampled using the error microphone and the goal of the system is to minimize this error signal. P(z) is the primary path from the noise source to the error microphone. S(z) is the secondary path from the noise cancelling loudspeaker to the error microphone and $F_z(n)$ is the feedback path from the loudspeaker to the primary noise which will be neglected in this analysis.

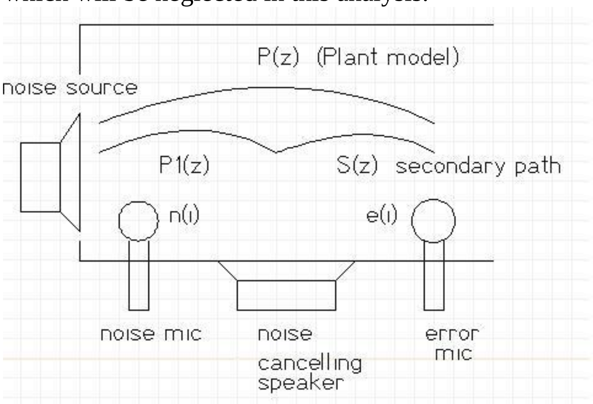

Fig.1.System diagram for active noise cancellation

The problem statement here is to eliminate background noise by transmitting an inverse signal. This problem involves 3 parts[1]: 1. Capture the background noise and estimate the plant model $P_z(n)$ that is producing this noise. The plant model will typically be an auto-regressive model which will be estimated using a modified Least Mean Square algorithm. 2. Estimate the secondary path $S_z(n)$ from noise cancelling speakers to the error microphone and include this in the algorithm estimate. 3. Adapt the plant model and secondary path model to the changing environment. The feedback path $F_z(n)$ is neglected here as it can be controlled using a directional loud speaker.

These 3 problems are incorporated into the filtered x-LMS algorithm with online secondary path modeling. The system implementation of this algorithm is shown below[1]:

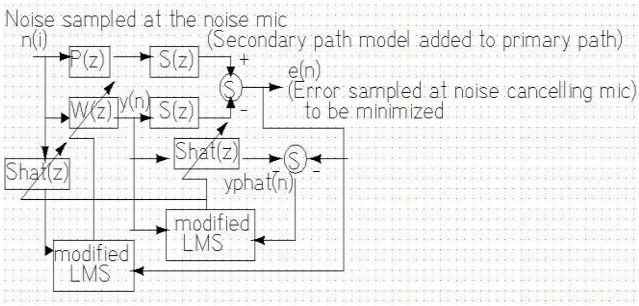

Fig.2.Block diagram of filtered x-LMS with online secondary path modeling.

Herein, $P(z)$ is the plant model from noise reference microphone to the error microphone, $W(z)$ is the estimate of the plant model $P(z)$ reached at by the FX-LMS algorithm (ANC block in previous figure), S(z) is the secondary path from noise canceling speakers to the error microphone and $\widehat{S(z)}$ is the estimate of the secondary path developed using the modified LMS algorithm. The 2 blocks indicated by LMS are implemented as a modified version of the LMS algorithm which has a high speed of convergence for rapidly changing environments. The algorithm is described later.

The signals mentioned in this system diagram are as follows: $x(n)$ is the noise signal that needs to be cancelled, $d(n)$ is the value of $x(n)$ at the noise cancelling error microphone after it passes through the channel model $P(z)$, $y(n)$ is the output of the noise cancelling speaker which passes through the secondary path $S(z)$ and reaches the error microphone, $e(n)$ is the error signal at the error microphone which needs to be minimized in the mean square error sense. $y'(n)$, $f(n)$ and $x'(n)$ are signals needed for online secondary path modeling and will be introduced eventually.

The first step in the implementation is to assume that the secondary path transfer function is unit impulse and thus neglect any real secondary path effects. In this case, the system can be reduced to the following problem where $S(z)$ is assumed to be unity:

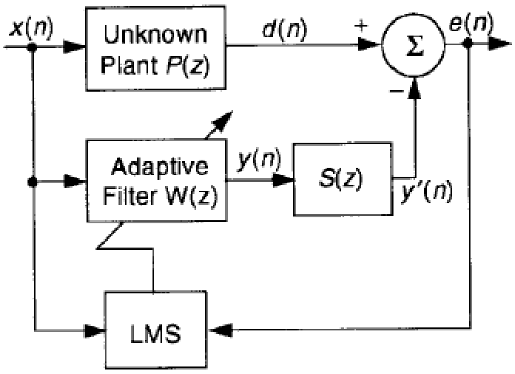

Fig.3.Block diagram for Active Noise Cancellation using LMS algorithm

This system modeling problem can be reduced to the classic least mean square algorithm where $e(n)$ is to be minimized in the mean square error sense. The mathematical formalism for this is stated as[2]:

$e(n) = d(n) - y(n) \ldots (1)$
$y(n) = x(n) * w(n) \ldots (2)$
$y(n) = y^{'}(n) \ldots \quad (3)$
$Cost\ function\ J(n) = E[e(n)e(n)]$
$J(n) = E\big[\big(d(n) - y(n)\big) * \big(d(n) - y(n)\big)\big] \ldots (4)$
$y(n) = w(n) * x(n) \ldots (5)$
$w(n+1) = w(n) + \mu x(n)e(n) \ldots (6)$

The cost function $J(n)$ needs to be minimized in the mean square error sense using the LMS algorithm and the tap weight update for this is given by equation 6. In the equation (6) for tap weight update, μ is the step size, $e(n)$ is the error, $w(n)$ are the tap weights of the filter in the present iteration and $x(n)$ is the input to the filter. The regular LMS algorithm as stated above suffers from a slow rate of convergence and does not work very well for rapidly changing environments.

A modified LMS algorithm was developed to overcome these difficulties related to the regular LMS algorithm and it is described below. The naming convention used is as follows: $d(n)$ = desired response, $e(n)$ = error, $y(n)$ = estimated response, a1, a2 and a3 are tap weights of the model, $E[x(n)]$ = Expected value of x(n) (mean of x(n)), $Rxx(n)$ is the nth-order auto-correlation function for the sequence $x(n)$.

$y(n) = a1 \cdot x(n-1) + a2 \cdot x(n-2) + a3 \cdot x(n-3) \ldots (7)$
$d(n) = x(n) \ldots (8)$
$e(n) = d(n) - y(n) = x(n) - a1 \cdot x(n-1) - a2 \cdot x(n-2) - a3 \cdot x(n-3) \ldots (9)$
$E[e(n) \cdot e(n)] = E[\{x(n) - a1 \cdot x(n-1) - a2 \cdot x(n-2) - a3 \cdot x(n-3)\} \cdot \{x(n) - a1 \cdot x(n-1) - a2 \cdot x(n-2) - a3 \cdot x(n-3)\}] \ldots (10)$
$E[e(n)^2] = \sigma_x^2 - a1R_{xx}(-1) - a2R_{xx}(-2) - a3R_{xx}(-3) - a1R_{xx}(-1) + a1^2R_{xx}(0) + a1a2R_{xx}(-1) + a1a3R_{xx}(-2) + a2a3R_{xx}(-1) + a2^2R_{xx}(0) + a1a2R_{xx}(-1) - a2R_{xx}(-2) - a3R_{xx}(3) + a1a3R_{xx}(2) + a2a3R_{xx}(1) + a3^2R_{xx}(0) \ldots (11)$
$\frac{\partial E[e(n)^2]}{\partial a1} = (2a2 - 2)R_{xx}(-1) + 2a1R_{xx}(0) + 2a3R_{xx}(2) \ldots (12)$
$\frac{\partial E[e(n)^2]}{\partial a2} = 2a2R_{xx}(0) + (2a1 + 2a3)R_{xx}(-1) - 2R_{xx}(-2) \ldots (13)$
$\frac{\partial E[e(n)^2]}{\partial a3} = 2a3R_{xx}(0) + 2a2R_{xx}(1) + 2a1R_{xx}(2) - 2R_{xx}(3) \ldots (14)$

The tap weights of the filter are updated using the equations 12 to 14 and these equations are as follows:

$-\nabla\overrightarrow{a1} = -2R_{xx}(0)\overrightarrow{a1} - R_{xx}(1)\big(2\overrightarrow{a2} - 2\widehat{a1}\big) - 2R_{xx}(2)\overrightarrow{a3} \ldots (15)$
$-\nabla\overrightarrow{a2} = -2R_{xx}(0)\overrightarrow{a2} - R_{xx}(1)\big(2\overrightarrow{a3} + 2\overrightarrow{a1}\big) + 2R_{xx}(2)\widehat{a2} \ldots (16)$
$-\nabla\overrightarrow{a3} = -2R_{xx}(0)\overrightarrow{a3} - R_{xx}(1)\big(2\overrightarrow{a2}\big) - R_{xx}(2)\big(2\overrightarrow{a1}\big) + 2R_{xx}(3)\widehat{a3} \ldots (17)$

The implementation of equations 15 – 17 gives the modified LMS algorithm. The important point of difference between this and the regular LMS algorithm is that the tap weight update for each of the filter tap weights is different and thus each tap weight moves in the individual optimum direction as required to minimize mean squared error. The step size is made proportional to the instantaneous error and this completes the feedback loop of the algorithm. If the error is less, the step size is small and thus the tap weights don't change much. Also if the filter is close to the minimum, the derivatives 15-17 will all become 0 and thus the tap weight updates will stop. The rate of convergence is faster than regular LMS algorithm and is suitable for rapidly changing environments. This algorithm is used in the LMS update step of the FXLMS for active noise cancellation.

In the present simplification, the secondary path from noise cancelling speaker to the error microphone is not being modeled. Usually this is just a delay of a finite value. This could be modeled once offline using a training sequence and then the path delay could be incorporated into the algorithm. This offline method does not adapt for changing environments and requires the offline calibration step which requires a high level of accuracy. The figure below indicates how to incorporate this path delay into the algorithm. The block $\hat{S}$(z) is the secondary path model added to system. In the simplest case, a delay identical to the secondary path is added to the input of the LMS update as indicated below.

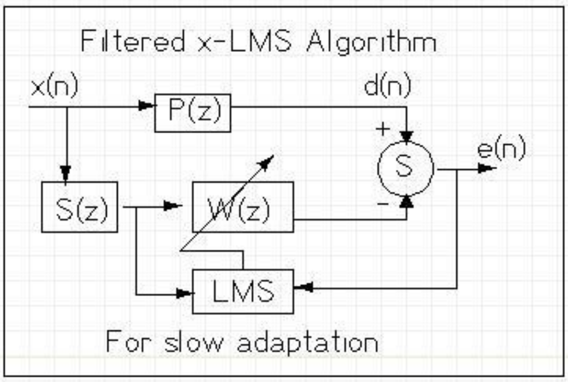

Fig.4.Block diagram of the filtered x-LMS algorithm

The online secondary path modeling technique is preferred to the offline technique as it can adapt to a changing environment and the adaptive nature of algorithm cancels calibration errors eventually. The system diagram for the FXLMS with online secondary path modeling was given in the beginning and is repeated here:

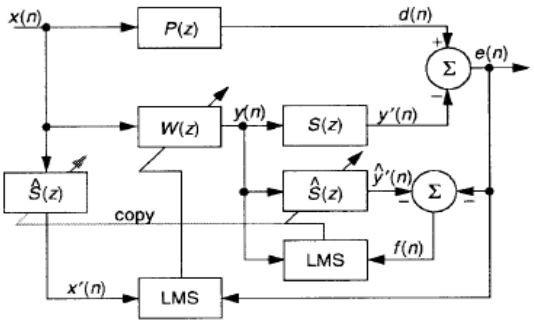

Fig.5.Block diagram of the filtered x-LMS algorithm.

As can be seen, the signal $y(n)$ is taken as the input to the secondary path modeling system and error between the output $\hat{y}'$(n) and $e(n)$ is minimized by the modified LMS algorithm to update the tap weights of $\hat{S}(z)$. An important point to note is that the input to the primary LMS update is filtered by the secondary path model. This usually works for slowly varying secondary paths.

A further change was added to this algorithm where in the secondary path transfer function $\hat{S}(z)$ was added to the primary path (fig 2). This could be achieved in practice by arranging the noise cancelling speaker in the direct path from noise source to error microphone. This arrangement would give the best performance for the noise cancelling algorithm and all other arrangements would give a somewhat higher error.

II. SIMULATION RESULTS

The above mentioned system model was implemented in SCILAB and the following figures depict the performance of the algorithm in estimating and eliminating the error.

The first figure indicates the channel correlation values for first 3 tap delays. This is the plant model P(z) in our case. This chart shows the correlation present in the error signal due to the plant noise model which is assumed to be auto regressive here.

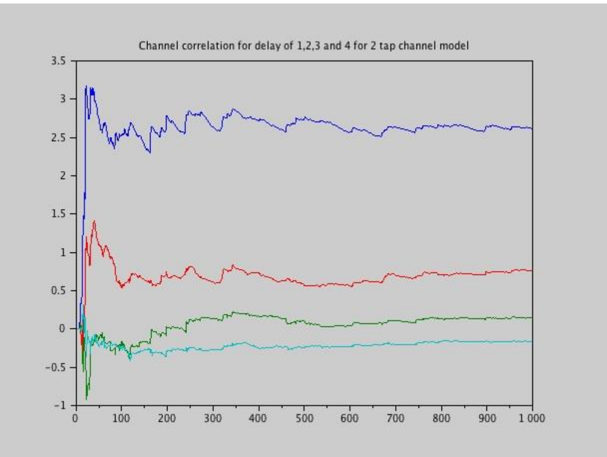

Fig.6.Computation of channel correlation for the plant model P(z).

The second figure indicates the updates of the filter tap weights based on the auto-correlation values computed above. The tap weights of the filter converge to a steady state value as soon as the channel correlation is estimated correctly.

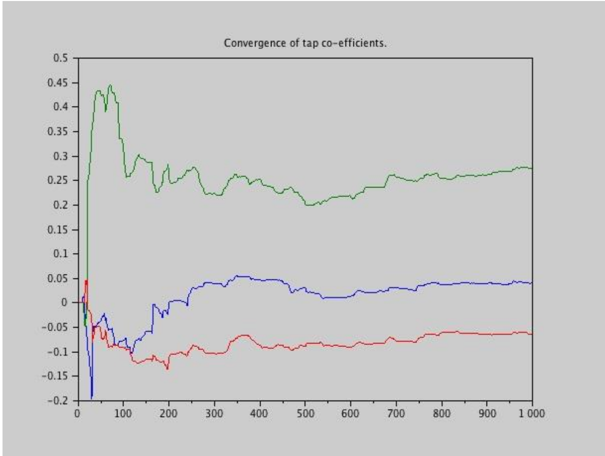

Fig.7.Estimation of the plant model P(z) from the channel correlation

This chart shows the estimate of the plant model converging as soon as the channel correlation reaches a steady state value.

Adaptive step sizes are used which are proportional to the size of the instantaneous error squared. The step size becomes 0 when the error is 0, allowing the algorithm to settle to a steady value.

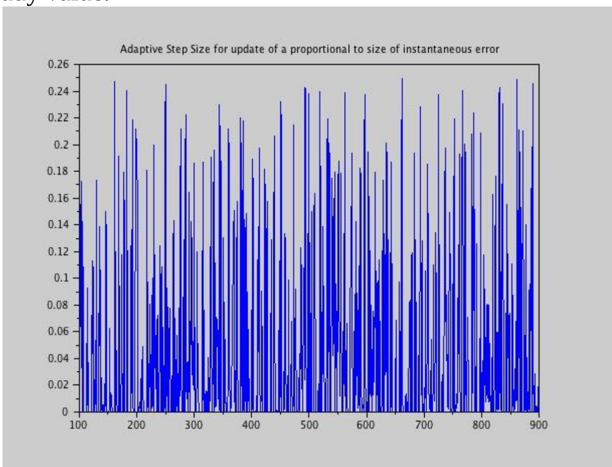

Fig.8. Plot of the adaptive step size used in the algorithm

The figure below indicates tracking of the loudspeaker output to the noise signal at the error microphone. Close tracking indicates good noise cancellation.

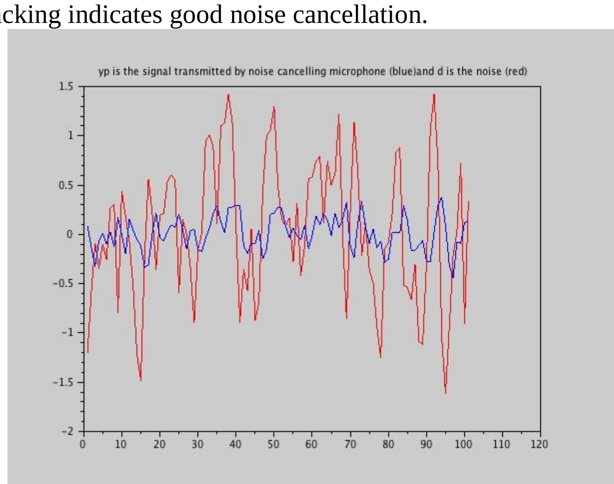

Fig.9. Overlay of channel noise over the estimated noise signal.

This figure below indicates the tracking of the signal e(n) on the secondary path modeling LMS by the signal $\hat{y}'$(n). This is the secondary error signal and thus is larger than the primary error signal.

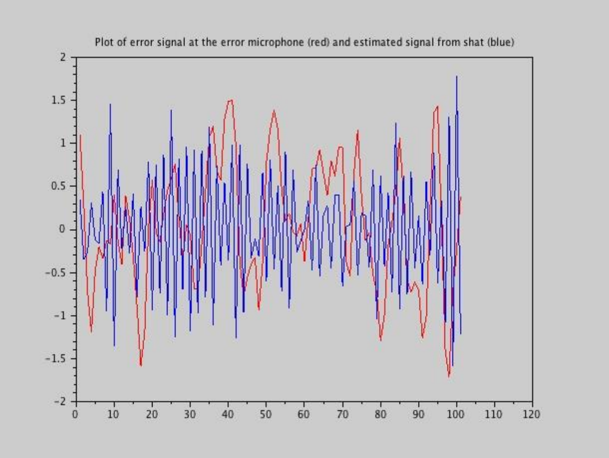

Fig.10. Overlay of channel error with error of the secondary path model.

The minimum mean square error between the signal transmitted by the noise cancelling microphone and the actual noise signal is less than 1, showing good convergence of the algorithm.

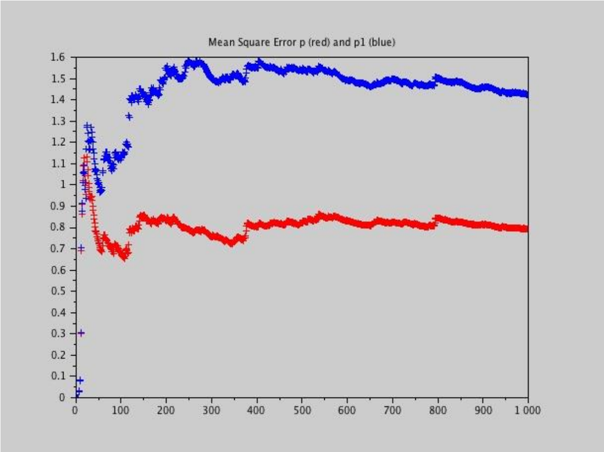

Fig.11. Convergence of the mean square error in estimate of channel noise (red) and secondary path (blue)

REFERENCES

[1] Active Noise Control: A Tutorial Review Sen M. Kuo and Dennis R. Morgan, Senior Member IEEE PROCEEDINGS OF THE IEEE, VOL. 87, NO. 6, JUNE 1999

[2] Adaptive Filter Theory by Simon Haykin

[3] Active Noise Cancellation System using DSP Processor G.U.Priyanga, T.Sangeetha, P.Saranya, Mr.B.Prasad International Journal of Scientific & Engineering Research, Volume 4, Issue 4, April-2013