A Novel Switching Scheme between Adaptive Information Algorithms

Seungju Han, Sudhir Rao, Deniz Erdogmus, and Jose Principe

***Abstract*—Switching approaches can improve the performance of adaptive schemes, however a data driven criterion to accomplish the task is unclear. In this paper, we propose a new optimization criterion for switching which is estimated directly from data. We apply the method to the recently introduced MEE and MEE-SAS algorithms. Using this novel switching scheme, we develop a single algorithm which effectively combines the strengths of MEE and MEE-SAS without sacrificing the simplicity and stability properties of MEE. We explain these results analytically, and through simulations.**

I. Introduction

THE mean square error (MSE) has been used as the fundamental performance criterion in adaptive filtering theory [1]. The main reason for the wide use of MSE lies in the various analytical and computational simplicities it brings coupled with the minimization of the error energy, which makes sense in the framework of linear signal processing. The least mean square (LMS) [2] has become the core algorithm in the minimization of the error energy and one of its variants is the least mean fourth (LMF) [3].

The least mean mixed-norm (LMMN) was introduced as an approach towards the combination of the advantages of LMS and LMF [4][5]. More precisely, it aimed at exploiting the faster initial convergence of the LMF algorithm, while retaining the desirable LMS characteristic of low misadjustment. This combination algorithm used a constant mixing parameter, which may be adapted to match appropriately the properties of the measured signals. However, the optimal value of the mixing parameter is hard to estimate.

In a statistical learning sense, especially for nonlinear signal processing, a better approach would be to constrain directly the information content of signals rather than simply their energy, if the designer seeks to achieve the best performance in terms of information filtering. In this regard, Renyi's entropy criterion applied to the error signal has been utilized as an alternative to MSE in the supervised adaptive system training. It uses a nonparametric estimator based on Parzen windowing with Gaussian kernels to estimate entropy directly from the data samples [6]. The motivation for pursuing the application of Renyi's entropy was the existence of an analytically and computationally simple estimator for Renyi's quadratic entropy, as well as the fact that the commonly used Shannon's entropy is a special case of Renyi's definition [7]. For instance, minimum error entropy (MEE) had been shown as a more robust criterion for dynamic modeling [8] and an alternative to MSE in other supervised learning applications using nonlinear systems.

We also proposed a Minimum Error Entropy with self adjusting step-size for faster convergence as compared to MEE algorithm [9]. MEE-SAS provides a "Target" to automatically control the algorithm step size. However, one disadvantage of MEE-SAS is the insensitivity of the algorithm due to the "flatness" of the surface near the optimal solution. When small changes in the desired signal need to be tracked, a shallow surface would hinder the tracking ability of MEE-SAS. The loss of "sensitivity" of MEE-SAS can be attributed to the extremely small value of $\left[V(\mathbf{0})-V(\mathbf{e})\right]$ near the optimal solution which suppresses the transfer of information from the information potential gradient to the weight vectors.

In this paper, we propose an automatic switching mechanism between the MEE and MEE-SAS algorithm, directly evaluated from the data as a method to improve their performance (faster convergence and lower misadjustment) without sacrificing their simplicity and stability properties. This method can in principle also be applied to the LMMN family.

The paper is organized as follows. Section II summarizes the Information Potential and presents the concept of MEE and MEE-SAS. We introduce our novel switching scheme in Section III. Section IV deals with simulation results and finally we conclude in Section V.

II. Information potential as a criterion for adaptation

The order-α Renyi's entropy of $\mathbf{X}$ is defined as

$$H_\alpha(\mathbf{X})=\frac{1}{1-\alpha}\log\int p^\alpha(\mathbf{x})d\mathbf{x}. \quad (1)$$

This lead to the generalized definitions of entropy providing the flexibility of a parametric family, while maintaining Shannon's definitions as the special case $\alpha=1$.

The probability distribution function (pdf) $p(\mathbf{x})$ is estimated using the nonparametric technique that can be employed for entropy estimation. For a given set of iid samples $\{\mathbf{x}_1,\ldots,\mathbf{x}_N\}$ drawn from the original distribution,

Seungju Han, Sudhir Rao, and Jose Principe are with the Computational Neuroengineering Laboratory (CNEL), Department of Electrical and Computer Engineering, University of Florida, Gainesville, FL 32601, USA (phone: 352-392-2682; fax:352-392-0044; email: han@cnel.ufl.edu).

Deniz Erdogmus is with Department of Computer Science and Electrical Engineering, Oregon Health and Science University, Portland, OR 97006, USA(e-mail: derdogmus@ieee.org).

the Parzen window estimate for the distribution, assuming a fixed-size kernel function $K_{\alpha}(\xi)$ for simplicity, is given by

$$p(\mathbf{x}) = \frac{1}{N}\sum_{i=1}^{N} K_{\sigma}(\mathbf{x} - \mathbf{x}_i)\,. \tag{2}$$

The kernel function and its size can be optimized in accordance with the maximum likelihood (ML) principle or other rules-of-thumb could be employed to obtain approximate optimal parameter selections.

We will treat the nonparametric estimation of Renyi's quadratic entropy ($\alpha = 2$) with Parzen windows. Substituting (2) in Renyi's entropy definition (1), we obtain the following nonparametric Gaussian kernel entropy estimator,

$$H_2(\mathbf{X}) = -\log V(\mathbf{X}) \tag{3}$$

$$V(\mathbf{X}) = \int p^2(\mathbf{x})d\mathbf{x} = \frac{1}{N^2}\sum_{j=1}^{N}\sum_{i=1}^{N} K_{\sigma\sqrt{2}}(\mathbf{x}_j - \mathbf{x}_i) \le V(\mathbf{0}) \tag{4}$$

where $V(\mathbf{X})$ is called the quadratic information potential.

For online adaptation algorithms, approximating the expectation by the most recent sample $\mathbf{x}_k$ and utilizing a small set of previously available samples for Parzen windowing, the instantaneous cost [10] is

$$\begin{aligned} V(\mathbf{X}) &= E[p(\mathbf{x})] \approx p(\mathbf{x}_k) \\ &= \frac{1}{L}\sum_{i=1}^{L} K_{\sigma}(\mathbf{x}_k - \mathbf{x}_{k-i})\,. \end{aligned} \tag{5}$$

Given samples from an input-output mapping, in order to extract the most information from the data, the error entropy over the training data set must be minimized [6]. When the error entropy is minimized, all moments of the error pdf (not only the second moments) are constrained.

By definition (3), minimizing the entropy is equivalent to maximizing the information potential since the log is a monotonic function. Thus, the cost function $J_{MEE}(\mathbf{e})$ for MEE criterion is given by

$$J_{MEE}(\mathbf{e}) = \max_{\mathbf{w}} V(\mathbf{e})\,. \tag{6}$$

Since the information potential is smooth and differentiable by the Gaussian kernel properties, we can use its gradient vector to be used in the steepest ascent algorithm shown below,

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \mu\nabla V(\mathbf{e}) \tag{7}$$

where $\nabla V(\mathbf{e})$ denotes the gradient of the information potential.

The maximum value $V(\mathbf{0})$ of the information potential will be achieved for a Dirac $\delta -$ distributed random variable $(e(1) = e(2) = \ldots = e(N))$. As can be easily inferred from (4), $V(\mathbf{e}) \le V(\mathbf{0})$ always; hence $V(\mathbf{0})$ provides an upper bound on the achievable $V(\mathbf{e})$. Seen from a different perspective, $V(\mathbf{0})$ is the ideal "target" value to be reached in the information potential curve. Thus $[V(\mathbf{0}) - V(\mathbf{e})]$ is always a non-negative scalar quantity which does not change the direction of the weight vector but can be used to accelerate the conventional gradient search algorithm given in (7). This modified search algorithm is named MEE-SAS. The weight update in MEE-SAS becomes

$$\begin{aligned} \mathbf{w}(n+1) &= \mathbf{w}(n) + \mu[V(\mathbf{0}) - V(\mathbf{e})]\nabla V(\mathbf{e}) \\ &= \mathbf{w}(n) + \mu(n)\nabla V(\mathbf{e}) \end{aligned} \tag{8}$$

where $\mu(n) = \mu[V(\mathbf{0}) - V(\mathbf{e})]$.

We can further note that there exists a cost function which gives rise to this gradient descent algorithm which is given by,

$$J_{MEE-SAS}(\mathbf{e}) = \min_{\mathbf{w}}\, [V(\mathbf{0}) - V(\mathbf{e})]^2. \tag{9}$$

III. SWITCHING SCHEME

When far from the optimum, the MEE-SAS algorithm exhibits faster convergence over the MEE, while when close to it, the MEE algorithms track better than the MEE-SAS [9]. In order to decide the switching time to maximize convergence speed, an analytical criterion needs to be developed.

Our idea is based on the simple fact that these two algorithms have the same gradient direction. Further, even though the update looks quite different, we can show easily that the optimal solution of MEE and MEE-SAS is the same. Thus at any given location of the performance surface, the only difference between the two algorithms lies in the size of the step we take towards the optimal solution. Thus, we base our switching scheme by selecting that algorithm which gives maximum decrease in cost function.

For simplicity, suppose that adaptation is being performed in continuous-time (which could be easily approximated by the typical discrete-time update rules used in practice). We have the following MEE-SAS cost function and the continuous-time learning rule:

$$J_{MEE-SAS}(\mathbf{e}) = [V(\mathbf{0}) - V(\mathbf{e})]^2 \tag{10}$$

$$\dot{\mathbf{w}} = \frac{\partial \mathbf{w}}{\partial t} = -\mu_{MEE-SAS} \cdot \frac{\partial J_{MEE-SAS}(\mathbf{e})}{\partial \mathbf{w}} \tag{11}$$

From this, we obtain the following temporal dynamics that describes the learning rule:

$$\begin{aligned} \dot{J}_{MEE-SAS} &= -2[V(\mathbf{0}) - V(\mathbf{e})]\frac{\partial V(\mathbf{e})}{\partial \mathbf{w}}^T \cdot \dot{\mathbf{w}} \\ &= -4\mu_{MEE-SAS}[V(\mathbf{0}) - V(\mathbf{e})]^2 \left\| \frac{\partial V(\mathbf{e})}{\partial \mathbf{w}} \right\|^2 \end{aligned} \tag{12}$$

On the contrary, the regular MEE rule would have the following energy function and update rule as shown below. Note that the maximization of $V(\mathbf{e})$ in (6) has been changed to a minimization problem using (4) for direct comparison with MEE-SAS.

$$J_{MEE} = [V(\mathbf{0}) - V(\mathbf{e})] \tag{13}$$

$$\dot{\mathbf{w}} = -\mu_{MEE} \frac{\partial J_{MEE}(\mathbf{e})}{\partial \mathbf{w}} \tag{14}$$

This corresponds to the following temporal dynamics for the minimization of energy:

$$\dot{J}_{MEE} = -\mu_{MEE} \frac{\partial V(\mathbf{e})}{\partial \mathbf{w}}^T \dot{\mathbf{w}} = -\mu_{MEE} \left\| \frac{\partial V(\mathbf{e})}{\partial \mathbf{w}} \right\|^2 \tag{15}$$

From (12) and (15), the general switching time is determined as

$$\left| \dot{J}_{MEE-SAS} \right| = \left| \dot{J}_{MEE} \right| . \tag{16}$$

Therefore, in the region satisfying the condition $\left| \dot{J}_{MEE-SAS} \right| > \left| \dot{J}_{MEE} \right|$, MEE-SAS should be used since MEE-SAS converges faster than MEE, otherwise MEE is used. However, the application of the switching decision expression (16) to the stochastic gradient search would entail high computational complexity (the computational complexity of both MEE and MEE-SAS). Instead, we can modify (16) simply as

$$\begin{aligned} & \left| \dot{J}_{MEE-SAS} \right| > \left| \dot{J}_{MEE} \right| \\ & \Leftrightarrow 4\mu_{MEE-SAS} [V(\mathbf{0}) - V(\mathbf{e})]^2 \left\| \frac{\partial V(\mathbf{e})}{\partial \mathbf{w}} \right\|^2 > \mu_{MEE} \left\| \frac{\partial V(\mathbf{e})}{\partial \mathbf{w}} \right\|^2 \\ & \Leftrightarrow V(\mathbf{e}) < V(\mathbf{0}) - \frac{1}{2} \sqrt{\frac{\mu_{MEE}}{\mu_{MEE-SAS}}} . \end{aligned} \tag{17}$$

In (17), we need to check just the information potential at each iteration and compare it with a constant, which is evaluated with the learning rates of MEE and MEE-SAS.

IV. Results

Here, we perform a detailed set of simulations to clarify our idea. We start with single step prediction of non-stationary time series where we highlight the weakness of MEE and MEE-SAS and show how the switching scheme solves this problem. We then extend it to a more complex problem of switching between two non-stationary systems. Finally we show the improved performance of our switching algorithm in a realistic scenario of acoustic echo cancellation.

A. *Single-step Prediction*

First, we consider a FIR filter for single-step prediction of the Mackey-Glass (MG) time series using the SIG estimation of information potential. The MG time series is generated by an MG system with delay parameter $\tau = 30$. The input vector consists of 6 (tap) consecutive samples of the MG time series.

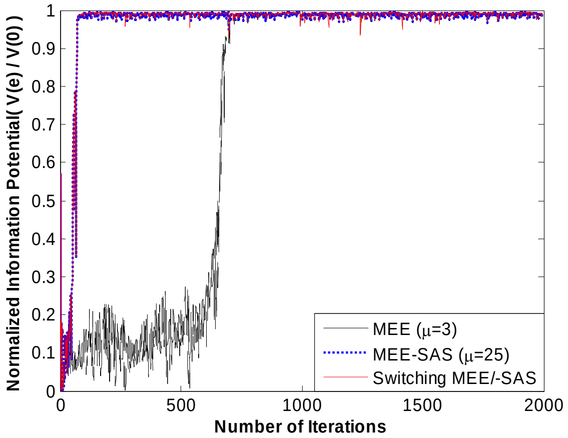

Fig.1. Information potential for online prediction of Mackey Glass time series

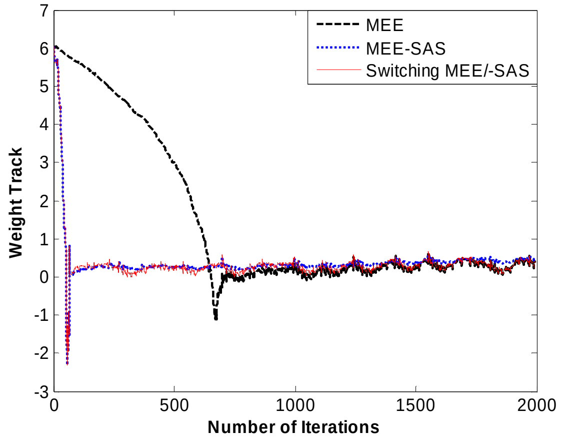

Fig.2. One of the six weight tracks (W_3)

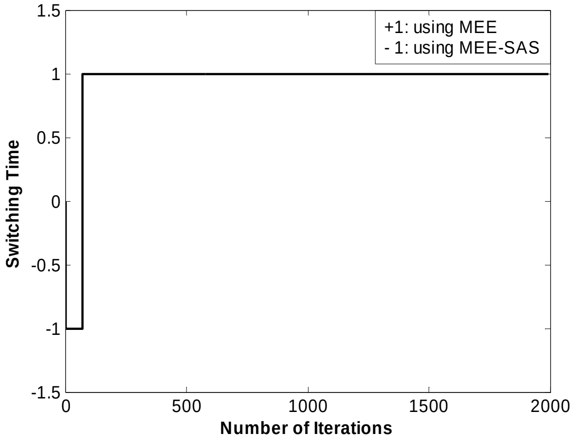

Fig.3. MEE or MEE-SAS used time on the switching MEE and MEE-SAS

We used the non-stationary MG time series to compare the weight tracking ability of MEE, MEE-SAS and the switching MEE and MEE-SAS. Due to online mode of simulation, SIG results in some misadjustment and variation about the optimal solution. We choose a proper kernel size ($\sigma = 0.707$) based on Silverman's rule and set the window length to $L = 50$.

In Fig.1, the drawback of MEE becomes quite evident. MEE takes 800 iterations to converge compared to MEE-SAS which converges in about 100 iterations. On the other hand, note the large fluctuations in the information potential curve of MEE as compared to MEE-SAS. To investigate the effect of these large fluctuations, we plot the weight track in Fig. 2. The fluctuations in the information potential curve of MEE translate into ability to track the changes in optimal solution of the non-stationary MG time series. Unlike MEE, the loss of tracking ability of MEE-SAS is attributed to the small effective step size near the optimal solution. As seen from Fig. 1, the switching algorithm utilized the MEE-SAS to go quickly near the optimal solution and then switched to MEE for tracking the small change in the solution, thus effectively combining the strengths of both the algorithms. This becomes clearly in Fig. 3 where the exact switching nature of our new algorithm is depicted.

B. *System Identification*

In this experiment, we consider the problem of tracking the weights of a system which switches abruptly between two subsystems. To increase the complexity, we introduce non-stationarity by changing the subsystems slowly. The non-stationary unknown plant transfer function is given as

$$H(z) = \begin{cases} \left(2 + \dfrac{2n}{1000}\right) \cdot \begin{bmatrix} 0.1 + 0.2z^{-1} + 0.3z^{-2} + 0.4z^{-3} + 0.5z^{-4} \\ + 0.4z^{-5} + 0.3z^{-6} + 0.2z^{-7} + 0.1z^{-8} \end{bmatrix}, & 1 \le n \le 1000 \\ \left(\dfrac{n}{1000} - 1\right) \cdot \begin{bmatrix} 0.1 + 0.2z^{-1} + 0.3z^{-2} + 0.4z^{-3} + 0.5z^{-4} \\ + 0.4z^{-5} + 0.3z^{-6} + 0.2z^{-7} + 0.1z^{-8} \end{bmatrix}, & 1001 \le n \le 2000. \end{cases} \tag{18}$$

The FIR adaptive filter is selected with equal order. The input to both the plant and the adaptive filter is white Gaussian noise with unit variance. We select window length $L = 50$ and kernel size $\sigma = 0.707$. The System mismatch (weight error power) is selected as a performance measure.

Fig.4 shows the weight tracks of MEE, MEE-SAS and the switching algorithm. Note how quickly MEE-SAS tracks the abrupt change. The ability to adaptively change its step size and track large variations is one of the strengths of MEE-SAS. On the other hand MEE, even though it takes long time to track the switching between the subsystems, gives a lower weight error power on a long run as shown in Fig. 5. This is attributed once again to it ability to track small changes near the optimal solution. Fig. 5 shows how our switching scheme takes advantage of both of them giving the best performance in terms of weight error power. As seen from Fig.6, in abrupt changing part (at initial and 1000th iteration), the switching algorithm uses MEE-SAS while to track fine changes it uses MEE algorithm.

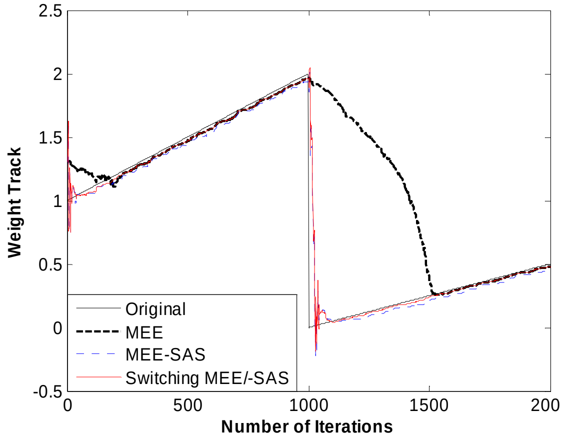

Fig.4. One of the nine weight tracks (W_5)

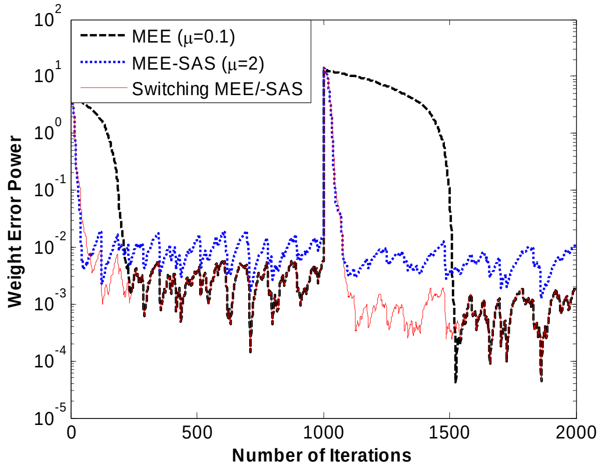

Fig.5. Weight error power

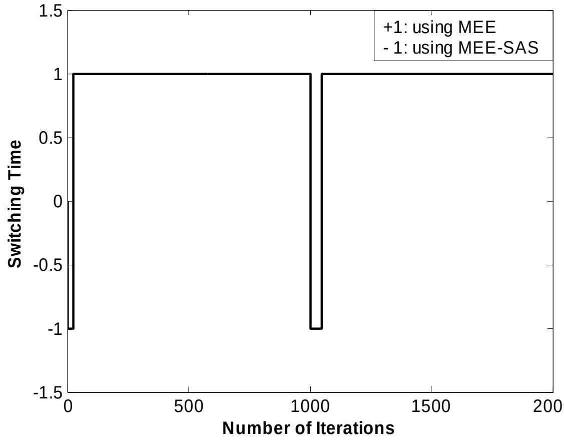

Fig.6. MEE or MEE-SAS used time on the switching MEE and MEE-SAS

C. *Acoustic Echo Cancellation*

As a practical example, we consider the acoustic echo cancellation. The aim is to minimize contribution of the echo

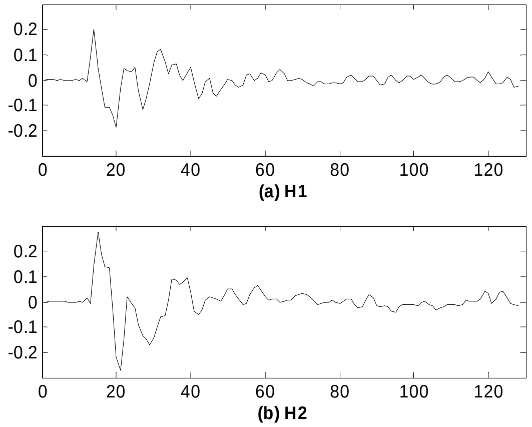

Fig.7. Room Impulse Response (H)

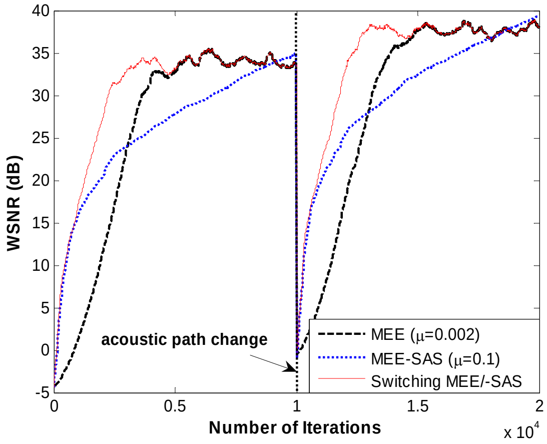

Fig.8. Weight SNR

signal by exactly estimating a room impulse response which is made by an acoustic path. In this problem, the acoustic path change occasionally occurs in the near end conference room. The changing nature is mainly due to changes in the acoustic environment. For example, these are from moving objects in the environment, and movement of the microphone within that environment. All these effect a change in the reverberation of the sound in the space. For this reason, the cancellation algorithms need to compensate for the abrupt echo path change.

We use two different impulse response of length H=128 in Fig.7. At 10000th iteration, the acoustic path changed from H1 to H2. Unlike the previous experiment, the system is stationary before and after this abrupt change. The same length is used for all the adaptive filters. The input signal is a uniform distribution signal with unit variance. The measurement noise is white Gaussian distributed with zero mean and 10^{-4} variance. We selected a kernel size of $\sigma = 0.707$ based on Silverman's rule and set the window length to $L = 200$. In order to test the ability of convergence to compensate for the abrupt echo path change, we use the weight SNR as a measure of performance.

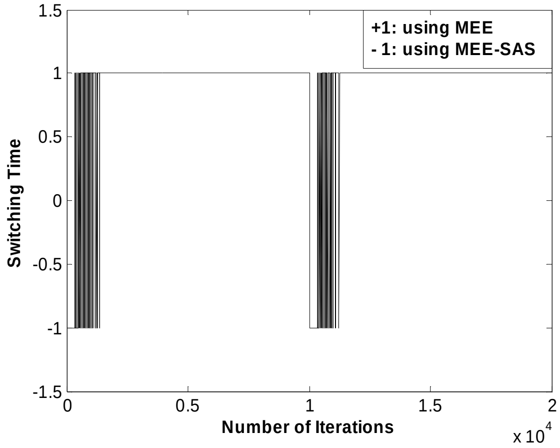

Fig.9. MEE or MEE-SAS used time on the switching MEE and MEE-SAS

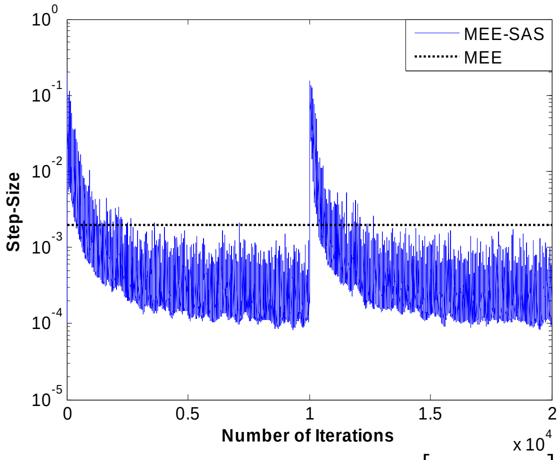

Fig.10. Step-Size of MEE-SAS ($\mu(n) = \mu[V(\mathbf{0}) - V(\mathbf{e})]$)

$$WSNR = 10\log_{10}\left[\frac{\|\mathbf{w}_*\|^2}{\|\mathbf{w}(n)-\mathbf{w}_*\|^2}\right][\mathbf{dB}] \qquad (19)$$

Fig. 8 shows the weight SNR of three algorithms. The performance of the switching algorithm is the same as that of MEE-SAS in abrupt changing part (at initial and 10000th iteration), while it is the same as that of MEE around the solution. This switching is seen clearly in Fig. 9. Notice the rattling effect around the switching time. This is due to the fluctuations in the effective step size of MEE-SAS as shown in Fig. 10. Since our switching scheme (17) utilizes this information, we continuously switch between MEE and MEE-SAS in this uncertainty region. The fluctuations in the step-size of MEE-SAS are attributed to the difficulty in getting a reliable approximation of $V(\mathbf{e})$ using stochastic estimation in this practical problem. It should be noted that this rapid switching in no way harms our switching algorithm and in fact ensures that we select the best algorithm at each iteration, thus enhancing the performance of our switching scheme. Also note the spikes in Fig. 10 at 10000th iteration reflecting a very fast step-size adjustment which helps MEE-SAS to immediately track the acoustic path change.

V. Conclusion

The switching MEE and MEE-SAS algorithm has been developed and shown to outperform MEE and MEE-SAS judging the performance of both rate of convergence and tracking. Further, we note that the total time consumed by an algorithm can be optimized considerably by simultaneously switching decision with respect to the convergence speed of MEE and MEE-SAS without sacrificing their simplicity and stability properties. We have corroborated this with extensive experiments on both artificial and real systems showing how the switching algorithm effectively combines the strengths of both MEE and MEE-SAS to give good performance.

Future work involves extending this novel multiple-switching technique to select the optimal p in the general cost function $J = [V(\mathbf{0}) - V(\mathbf{e})]^p$.

Acknowledgment

This work was partially supported by NSF grants ECS-0300340 and ECS-0601271. Deniz Erdogmus was supported by NSF grants ECS-0524835 and ECS-0622239.

References

[1] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*, Prentice Hall, New Jersey, 1985.W.-K. Chen, *Linear Networks and Systems* (Book style). Belmont, CA: Wadsworth, 1993, pp. 123–135.

[2] Simon Haykin, *Adaptive Filter Theory*, Prentice Hall, Upper Saddle River, 4th edition, 2001.

[3] E. Walach and B. Widrow, “The Least Mean Fourth (LMF) Adaptive Algorithm and its Family,” *IEEE Trans. Information Theory*, vol. IT 30, no.2, pp. 275-283, March 1984.

[4] D.I. Pazaitis and A.G. Constantinides, “LMS+F algorithm,” *Electronics Letters*, vol.31, no.17, pp. 1423-1424, August 1995.

[5] A. Zerguine, C.F.N. Cowan and M. Bettayeb, “Adaptive Echo Cancellation using Least Mean Mixed-Norm Algorithm,” *IEEE Trans. Signal Processing*, vol.45, no.5, pp. 1340-1343, May 1997.

[6] D. Erdogmus and J.C. Principe, “An Entropy Minimization algorithm for Supervised Training of Nonlinear Systems,” *IEEE Trans. Signal Processing*, vol.50, no.7, pp. 1780-1786, July 2002.

[7] A. Renyi, *Probability Theory*, Elsevier, New York, 1970.

[8] D. Erdogmus and J.C. Principe, “Generalized Information Potential Criterion for Adaptive System Training,” *IEEE Trans. Neural Networks*, vol.13, no.5, pp. 1035-1044, September 2002.

[9] S. Han, S. Rao, D. Erdogmus and J.C. Principe, “An Improved Minimum Error Entropy Criterion with Self-Adjusting Step-size,” in *Proc. IEEE Int. Workshop on Machine Learning for Signal Processing* , Mystic, Connecticut, pp. 317-322, September 2005.

[10] D. Erdogmus, J.C. Principe and K.E. Hild II, “Online entropy manipulation: Stochastic Information Gradient,” *IEEE Signal Processing Letters*, vol.10, no.8, pp.242-245, August 2003.