AN EXPONENTIATED GRADIENT ADAPTIVE ALGORITHM FOR BLIND IDENTIFICATION OF SPARSE SIMO SYSTEMS

Jacob Benesty

Université du Québec, INRS-EMT
800 de la Gauchetière Ouest, Suite 6900
Montréal, Québec, H5A 1K6, Canada
benesty@inrs-emt.uquebec.ca

Yiteng (Arden) Huang, Jingdong Chen

Bell Laboratories, Lucent Technologies
600 Mountain Avenue
Murray Hill, New Jersey 07974, USA
{arden, jingdong}@research.bell-labs.com

ABSTRACT

Sparse impulse responses are encountered in many acoustic and wireless channels. Recently, a class of exponentiated gradient (EG) algorithms has been proposed. One of the algorithms belonging to this class, the so-called EG± algorithm, converges and tracks much better than the classical stochastic gradient, or LMS, algorithm for sparse impulse responses. In this paper, we apply this technique to blind identification of a sparse SIMO system and develop the multichannel EG± algorithm. A simple experiment demonstrates its advantage in convergence compared to the MCLMS algorithm.

1. INTRODUCTION

Blind channel identification (BCI) is an important technique with extensive applications in signal processing and communications. For a real-time implementation in practice, an adaptive BCI algorithm is apparently desirable. In [1], we found a systematic way to design adaptive algorithms for BCI and proposed the multichannel LMS (MCLMS) method. But such an LMS-based method converges slowly particularly when the single-input multiple-output (SIMO) system to be identified has long impulse responses, which is a common drawback of the LMS algorithm [2].

Recently, another variant of the LMS algorithm, called the exponentiated gradient algorithm with positive and negative weights (EG± algorithm), was proposed by Kivinen and Warmuth [3]. This new algorithm converges much faster than the LMS algorithm when the impulse response that we need to identify is sparse [4], which is often the case in acoustic and wireless channels. The EG± algorithm has the nice feature that its update rule takes advantage of the sparseness of an impulse response to speed up its initial convergence and to improve its tracking abilities compared to LMS. In this paper, we intend to apply this idea to the BCI problem and will propose a multi-channel EG± (MCEG±) algorithm.

2. GENERAL DERIVATION OF ADAPTIVE ALGORITHMS FOR SYSTEM IDENTIFICATION

In designing a gradient-based adaptive algorithm for system identification, there are different ways to define the distance between the old and new weight vectors which lead to different update rules. Here we present a general derivation of these adaptive algorithms.

We begin with defining the *a priori* error signal $e(n+1)$ at time $n+1$ as:

$$e(n+1) = y(n+1) - \hat{y}(n+1), \tag{1}$$

where

$$y(n+1) = \mathbf{h}_{\mathrm{t}}^T \mathbf{x}(n+1) \tag{2}$$

is the system output,

$$\mathbf{h}_{\mathrm{t}} = [h_{\mathrm{t},0} \;\; h_{\mathrm{t},1} \;\; \cdots \;\; h_{\mathrm{t},L-1}]^T \tag{3}$$

is the true (subscript t) impulse response of the system, $(\cdot)^T$ denotes transpose of a vector or a matrix,

$$\mathbf{x}(n+1) = [x(n+1) \;\; x(n) \;\; \cdots \;\; x(n-L+2)]^T$$

is a vector containing the last L samples of the input signal x,

$$\hat{y}(n+1) = \mathbf{h}^T(n)\mathbf{x}(n+1) \tag{4}$$

is the model filter output, and

$$\mathbf{h}(n) = [h_0(n) \;\; h_1(n) \;\; \cdots \;\; h_{L-1}(n)]^T \tag{5}$$

is the model filter.

One easy way to find adaptive algorithms that adjust the new weight vector $\mathbf{h}(n+1)$ from the old one $\mathbf{h}(n)$ is to minimize the following function [3]:

$$J\left[\mathbf{h}(n+1)\right] = d\left[\mathbf{h}(n+1), \mathbf{h}(n)\right] + \eta\epsilon^2(n+1), \tag{6}$$

where $d\left[\mathbf{h}(n+1), \mathbf{h}(n)\right]$ is a measure of distance from the old to the new weight vector,

$$\epsilon(n+1) = y(n+1) - \mathbf{h}^T(n+1)\mathbf{x}(n+1) \tag{7}$$

is the *a posteriori* error signal, and η is a positive constant. The magnitude of η represents the importance of correctiveness compared to the importance of conservativeness [3]. If η is very small, minimizing $J\left[\mathbf{h}(n+1)\right]$ is close to minimizing $d\left[\mathbf{h}(n+1), \mathbf{h}(n)\right]$, so that the algorithm makes very small updates. On the other hand, if η is very large, the minimization of $J\left[\mathbf{h}(n+1)\right]$ is almost equivalent to minimizing $d\left[\mathbf{h}(n+1), \mathbf{h}(n)\right]$ subject to the constraint $\epsilon(n+1) = 0$.

To minimize $J\left[\mathbf{h}(n+1)\right]$, we need to set its L partial derivatives $\partial J\left[\mathbf{h}(n+1)\right] / \partial h_l(n+1)$ to zero. Hence, the different weight coefficients $h_l(n+1)$, $l = 0, 1, \cdots, L-1$, will be found by solving the equations:

$$\frac{\partial d\left[\mathbf{h}(n+1), \mathbf{h}(n)\right]}{\partial h_l(n+1)} - 2\eta x(n+1-l)\epsilon(n+1) = 0. \tag{8}$$

Solving (8) is in general very difficult. However, if the new weight vector $\mathbf{h}(n+1)$ is close to the old weight vector $\mathbf{h}(n)$, replacing the *a posteriori* error signal $\epsilon(n+1)$ in (8) with the *a priori* error signal $e(n+1)$ is a reasonable approximation and the equation

$$\frac{\partial d\left[\mathbf{h}(n+1), \mathbf{h}(n)\right]}{\partial h_l(n+1)} - 2\eta x(n+1-l)e(n+1) = 0 \quad (9)$$

is much easier to solve for all distance measures d.

The LMS algorithm is easily obtained from (9) by using the squared Euclidean distance

$$d_{\mathrm{E}}\left[\mathbf{h}(n+1), \mathbf{h}(n)\right] = \|\mathbf{h}(n+1) - \mathbf{h}(n)\|_2^2. \quad (10)$$

The EG algorithm with positive weights results from using for d the *relative entropy*, also known as *Kullback-Leibler divergence*,

$$d_{\mathrm{re}}\left[\mathbf{h}(n+1), \mathbf{h}(n)\right] = \sum_{l=0}^{L-1} h_l(n+1) \ln \frac{h_l(n+1)}{h_l(n)}, \quad (11)$$

with the constraint $\sum_l h_l(n+1) = 1$, so that (9) becomes:

$$\frac{\partial d_{\mathrm{re}}[\mathbf{h}(n+1), \mathbf{h}(n)]}{\partial h_l(n+1)} - 2\eta x(n+1-l)e(n+1) + \gamma = 0, \quad (12)$$

where γ is the Lagrange multiplier. Actually, the appropriate constraint should be $\sum_l h_l(n+1) = \sum_l h_{\mathrm{t},l}$ but $\sum_l h_{\mathrm{t},l}$ is not known in practice, so we take the arbitrary value 1 instead. This will have an effect on the adaptation step of the resulting adaptive algorithm.

The algorithm derived from (12) is valid for positive weights only. To deal with both positive and negative coefficients, we can always find two vectors $\mathbf{h}^+(n+1)$ and $\mathbf{h}^-(n+1)$ with positive coefficients, in such a way that the vector

$$\mathbf{h}(n+1) = \mathbf{h}^+(n+1) - \mathbf{h}^-(n+1) \quad (13)$$

can have positive and negative components. In this case, the *a posteriori* error signal can be written as:

$$\epsilon(n+1) = y(n+1) - [\mathbf{h}^+(n+1) - \mathbf{h}^-(n+1)]^T \mathbf{x}(n+1) \quad (14)$$

and the function (6) will change to:

$$\begin{aligned} & J\left[\mathbf{h}^+(n+1), \mathbf{h}^-(n+1)\right] \\ &= d[\mathbf{h}^+(n+1), \mathbf{h}^+(n)] + d[\mathbf{h}^-(n+1), \mathbf{h}^-(n)] \\ &\quad + \frac{\eta}{u}\epsilon^2(n+1), \end{aligned} \quad (15)$$

where u is a positive scaling constant. Using the same approximation as before and choosing the Kullback-Leibler divergence plus the constraint $\sum_l [h_l^+(n+1) + h_l^-(n+1)] = u$, the solutions of the equations

$$\frac{\partial d_{\mathrm{re}}\left[\mathbf{h}^+(n+1), \mathbf{h}^+(n)\right]}{\partial h_l^+(n+1)} - 2\frac{\eta}{u}x(n+1-l)e(n+1) + \gamma = 0, \quad (16)$$

$$\frac{\partial d_{\mathrm{re}}\left[\mathbf{h}^-(n+1), \mathbf{h}^-(n)\right]}{\partial h_l^-(n+1)} + 2\frac{\eta}{u}x(n+1-l)e(n+1) + \gamma = 0, \quad (17)$$

give the so-called EG± algorithm [3], where

$$e(n+1) = y(n+1) - \left[\mathbf{h}^+(n) - \mathbf{h}^-(n)\right]^T \mathbf{x}(n+1). \quad (18)$$

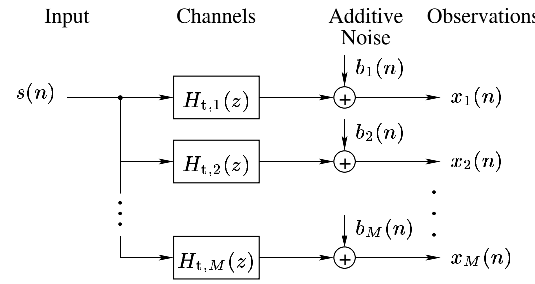

Figure 1: Illustration of the relationships between the input $s(n)$ and the observations $x_i(n)$ in an FIR SIMO system.

3. THE PRINCIPLE OF ADAPTIVE BLIND IDENTIFICATION OF A SIMO SYSTEM

In an FIR SIMO system as shown in Fig. 1, the i-th channel output $x_i(n)$ is the result of a linear convolution between the source signal $s(n)$ and the corresponding true channel impulse response $h_{\mathrm{t},i}$, corrupted by the additive background noise $b_i(n)$:

$$x_i(n) = h_{\mathrm{t},i} * s(n) + b_i(n), \quad i = 1, 2, ..., M, \quad (19)$$

where $*$ stands for linear convolution and M is the number of channels. In a vector form, (19) can be expressed as:

$$\mathbf{x}_i(n) = \mathbf{H}_{\mathrm{t},i} \cdot \mathbf{s}(n) + \mathbf{b}_i(n), \quad (20)$$

where

$$\begin{aligned} \mathbf{x}_i(n) &= [x_i(n) \;\; x_i(n-1) \;\; \cdots \;\; x_i(n-L+1)]^T, \\ \mathbf{H}_{\mathrm{t},i} &= \begin{bmatrix} h_{\mathrm{t},i,0} & \cdots & h_{\mathrm{t},i,L-1} & \cdots & 0 \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ 0 & \cdots & h_{\mathrm{t},i,0} & \cdots & h_{\mathrm{t},i,L-1} \end{bmatrix}, \\ \mathbf{s}(n) &= [s(n) \;\; s(n-1) \;\; \cdots \;\; s(n-2L+2)]^T, \\ \mathbf{b}_i(n) &= [b_i(n) \;\; b_i(n-1) \;\; \cdots \;\; b_i(n-L+1)]^T, \end{aligned}$$

L is set to the length of the longest channel impulse response by assumption. The channel parameter matrix $\mathbf{H}_{\mathrm{t},i}$ is of dimension $L \times (2L-1)$ and is constructed from the channel's impulse response:

$$\mathbf{h}_{\mathrm{t},i} = [h_{\mathrm{t},i,0} \;\; h_{\mathrm{t},i,1} \;\; \cdots \;\; h_{\mathrm{t},i,L-1}]^T. \quad (21)$$

The noise signals in the different channels are assumed to be white, independent of each other, and uncorrelated with the source signal.

A BCI algorithm is to estimate the channel impulse responses $\mathbf{h}_i$ $(i = 1, 2, \cdots, M)$ from the observations $x_i(n)$ without utilizing the source signal $s(n)$. The following two assumptions are made in the remainder of this paper to guarantee an identifiable system using only the second-order statistics [5]:

1. The polynomials formed from $\mathbf{h}_{\mathrm{t},i}, i = 1, 2, \cdots, M$, are co-prime, i.e., the channel transfer functions $H_{\mathrm{t},i}(z)$ do not share any common zeros;
2. The autocorrelation matrix $\mathbf{R}_{ss} = E\left\{\mathbf{s}(n)\mathbf{s}^T(n)\right\}$ of the source signal is of full rank.

The adaptive algorithms including the multichannel LMS and Newton methods proposed by the authors in an earlier study [1] blindly identifies a SIMO system by exploiting the cross relations among system outputs. By following the fact that

$$x_i(n) * h_{t,j} = s(n) * h_{t,i} * h_{t,j} = x_j(n) * h_{t,i}, \quad (22)$$

a cross-relation between the i-th and j-th channel outputs, in the absence of noise, can be formulated as

$$\mathbf{x}_i^T(n)\mathbf{h}_{t,j} = \mathbf{x}_j^T(n)\mathbf{h}_{t,i}, \;\; i,j = 1,2,...,M, \; i \neq j. \quad (23)$$

Then similar to (1) and (7), we can define respectively the *a priori* error

$$e_{ij}(n+1) = \mathbf{x}_i^T(n+1)\mathbf{h}_j(n) - \mathbf{x}_j^T(n+1)\mathbf{h}_i(n), \quad (24)$$

and the *a posteriori* error

$$\epsilon_{ij}(n+1) = \mathbf{x}_i^T(n+1)\mathbf{h}_j(n+1) - \mathbf{x}_j^T(n+1)\mathbf{h}_i(n+1), \quad (25)$$

where $\mathbf{h}_i(n)$ is the model filter for the i-th channel at time n and

$$\mathbf{h}(n) = \left[\mathbf{h}_1^T(n) \;\; \mathbf{h}_2^T(n) \;\; \cdots \;\; \mathbf{h}_M^T(n)\right]^T.$$

Following (6), we have the cost function for such a multichannel (subscript mc below) system:

$$J_{\rm mc}\left[\mathbf{h}(n+1)\right] = d\left[\mathbf{h}(n+1), \mathbf{h}(n)\right] + \frac{\eta}{u}\Gamma(n+1), \quad (26)$$

where η and u again are positive constants, and

$$\Gamma(n+1) \triangleq \sum_{i=1}^{M-1}\sum_{j=i+1}^{M} \epsilon_{ij}^2(n+1). \quad (27)$$

Using the squared Euclidean distance and enforcing a unit-norm constraint on the model filter (to avoid a trivial solution with all zero elements), we deduced the multichannel LMS algorithm and showed that the model filter $\mathbf{h}_i$ converges to $\mathbf{h}_{t,i}/\|\mathbf{h}_t\|$ ($i = 1,2,...,M$) [1].

4. THE MULTICHANNEL EG± ALGORITHM FOR BCI

In the previous section, we have briefly reviewed the process of developing an adaptive algorithm for blind identification of a SIMO system. In this section, we will demonstrate how to apply the concept of exponentiated gradient presented in Section 2 to BCI and will propose a multichannel EG± algorithm with appropriate constraints.

Taking the derivative of (26) produces

$$\frac{\partial d[\mathbf{h}(n+1), \mathbf{h}(n)]}{\partial \mathbf{h}(n+1)} + \frac{\eta}{u}\nabla\Gamma(n+1) = \mathbf{0}. \quad (28)$$

It can be shown that [1]:

$$\nabla\Gamma(n+1) = 2\tilde{\mathbf{R}}(n+1)\mathbf{h}(n+1), \quad (29)$$

where

$$\tilde{\mathbf{R}}(n) = \begin{bmatrix} \sum_{i\neq 1}\tilde{\mathbf{R}}_{x_ix_i}(n) & -\tilde{\mathbf{R}}_{x_2x_1}(n) & \cdots & -\tilde{\mathbf{R}}_{x_Mx_1}(n) \\ -\tilde{\mathbf{R}}_{x_1x_2}(n) & \sum_{i\neq 2}\tilde{\mathbf{R}}_{x_ix_i}(n) & \cdots & -\tilde{\mathbf{R}}_{x_Mx_2}(n) \\ \vdots & \vdots & \ddots & \vdots \\ -\tilde{\mathbf{R}}_{x_1x_M}(n) & -\tilde{\mathbf{R}}_{x_2x_M}(n) & \cdots & \sum_{i\neq M}\tilde{\mathbf{R}}_{x_ix_i}(n) \end{bmatrix},$$

and

$$\tilde{\mathbf{R}}_{x_ix_j}(n) = \mathbf{x}_i(n)\mathbf{x}_j^T(n), \;\; i,j = 1,2,...,M.$$

For positive weights, we can use the Kullback-Leibler divergence as before. With the constraint $\sum_l h_l(n+1) = u$ (note that this is different from the unit-norm constraint used in the multichannel LMS algorithm) and substituting (29) into (28), we get

$$\frac{\partial d_{\rm re}[\mathbf{h}(n+1), \mathbf{h}(n)]}{\partial \mathbf{h}(n+1)} + \frac{2\eta}{u}\tilde{\mathbf{R}}(n+1)\mathbf{h}(n+1) + \gamma\mathbf{1} = \mathbf{0}, \quad (30)$$

where γ is again a Lagrange multiplier and $\mathbf{1} = [1 \; 1 \; \cdots \; 1]^T$ is a vector of ones. For simplicity in solving (30), we approximate $\mathbf{h}(n+1)$ in the second term of (30) with $\mathbf{h}(n)$ and deduce the multichannel EG algorithm:

$$h_l(n+1) = u\frac{h_l(n)r_l(n+1)}{\sum_{k=0}^{ML-1} h_k(n)r_k(n+1)}, \quad l = 0,1,...,ML-1, \quad (31)$$

where

$$r_l(n+1) = \exp\left[-\frac{2\eta}{u}f_l(n+1)\right],$$

and $f_l(n+1)$ is the l-th element of the vector

$$\mathbf{f}(n+1) = \tilde{\mathbf{R}}(n+1)\mathbf{h}(n). \quad (32)$$

For a system with both positive and negative filter coefficients, we can decompose the model filter impulse responses $\mathbf{h}(n+1)$ into two components $\mathbf{h}^+(n+1)$ and $\mathbf{h}^-(n+1)$ with positive coefficients, as used in Section 2. Therefore the cost function (26) becomes:

$$\begin{aligned} & J_{\rm mc}\left[\mathbf{h}^+(n+1), \mathbf{h}^-(n+1)\right] \\ &= d_{\rm re}\left[\mathbf{h}^+(n+1), \mathbf{h}^+(n)\right] + d_{\rm re}\left[\mathbf{h}^-(n+1), \mathbf{h}^-(n)\right] \\ &\quad + \frac{\eta}{u_1+u_2}\chi(n+1), \end{aligned} \quad (33)$$

where u_1 and u_2 are two positive constants. Since $\mathbf{h}(n+1) = \mathbf{0}$ is an undesired solution, it is necessary to ensure that $\mathbf{h}^+(n+1)$ and $\mathbf{h}^-(n+1)$ would not be equal to each other from initialization and throughout the process of adaptation. Among many methods that can be used to enforce that $\mathbf{h}^+(n+1)$ and $\mathbf{h}^-(n+1)$ would not be identical, we propose the following constraints:

$$\sum_{l=0}^{ML-1} h_l^+(n+1) = u_1 = \kappa\|\mathbf{h}_t\|_1, \quad (34)$$

$$\sum_{l=0}^{ML-1} h_l^-(n+1) = u_2 = (1-\kappa)\|\mathbf{h}_t\|_1, \quad (35)$$

where $0 < \kappa < 1$ and $\kappa \neq 1/2$. Utilizing these constraints and taking derivatives of (33) with respect to $\mathbf{h}^+(n+1)$ and $\mathbf{h}^-(n+1)$ respectively gives

$$\ln\frac{h_l^+(n+1)}{h_l^+(n)} + 1 + \frac{2\eta}{u_1+u_2}f_l(n+1) + \gamma_1 = 0, \quad (36)$$

$$\ln\frac{h_l^-(n+1)}{h_l^-(n)} + 1 - \frac{2\eta}{u_1+u_2}f_l(n+1) + \gamma_2 = 0, \quad (37)$$

where γ_1 and γ_2 are two Lagrange multipliers. Solving (36) and (37) for $h_l^+(n+1)$ and $h_l^-(n+1)$ respectively leads to the multichannel EG± algorithm:

$$h_l^+(n+1) = u_1 \frac{h_l^+(n) r_l^+(n+1)}{\sum_{k=0}^{ML-1} h_k^+(n) r_k^+(n+1)}, \quad (38)$$

$$h_l^-(n+1) = u_2 \frac{h_l^-(n) r_l^-(n+1)}{\sum_{k=0}^{ML-1} h_k^-(n) r_k^-(n+1)}, \quad (39)$$

where

$$r_l^+(n+1) = \exp\left[-\frac{2\eta}{u_1+u_2} f_l(n+1)\right],$$

$$r_l^-(n+1) = \exp\left[\frac{2\eta}{u_1+u_2} f_l(n+1)\right] = \frac{1}{r_l^+(n+1)}.$$

5. SIMULATIONS

In this section, we compare by way of simulation, the multichannel LMS and EG± algorithms for blindly identifying a sparse SIMO system. The system consists of $M = 3$ channel and the impulse response of each channel has $L = 15$ taps. Figure 2 shows the three impulse responses. In each channel, a dominant component makes the impulse response sparse. A white Gaussian random sequence is used as source signal to excite the system. The channel output is intentionally corrupted by additive white Gaussian noise at 50 dB SNR. In the problem of BCI, a properly aligned vector is still a valid solution to the impulse response $\mathbf{h}_t$ even though their gains may differ. Therefore, *normalized projection misalignment* (NPM) is used as the performance measure and is given at time n by

$$\text{NPM}(n) \triangleq \frac{\|\boldsymbol{\delta}(n)\|_2}{\|\mathbf{h}_t\|_2}, \quad (40)$$

where

$$\boldsymbol{\delta}(n) = \mathbf{h}_t - \frac{\mathbf{h}_t^T \mathbf{h}(n)}{\mathbf{h}^T(n)\mathbf{h}(n)} \mathbf{h}(n)$$

is the projection misalignment vector [6]. By projecting $\mathbf{h}_t$ onto $\mathbf{h}(n)$ and defining a projection error, we take into account only the intrinsic misalignment of the channel estimate, disregarding an arbitrary gain factor. A comparison of the MCEG± and the MCLMS algorithm is given in Fig. 3. For the MCLMS algorithm, the step size $\mu = 1.6 \times 10^{-4}$. For the MCEG± algorithm, $\eta = 3.5 \times 10^{-2}$ and $\kappa = 0.75$. As clearly shown in the results, both the MCEG± and the MCLMS algorithm are able to determine the channel impulse responses while the MCEG± algorithm converges much faster than the MCLMS algorithm.

6. CONCLUSION

Sparsity in a channel impulse response can be exploited in adaptive algorithms to accelerate their convergence. In this paper, we developed the multichannel exponentiated gradient algorithm with positive and negative weights (MCEG± algorithm) for blind identification of a sparse SIMO system. A feasible constraint on the decomposed positive components of the channel impulse responses was proposed to avoid a trivial solution with all zero elements. It was shown with a simulation that the proposed MCEG± algorithm converges faster than the MCLMS algorithm for sparse impulse responses.

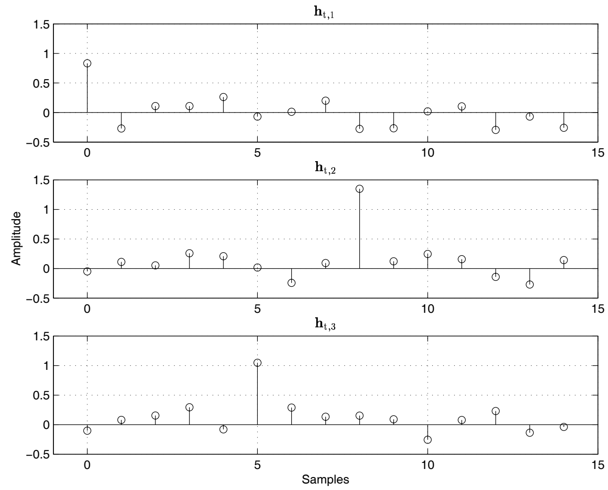

Figure 2: Impulse responses of a single-input three-output system used in the simulation for blind identification.

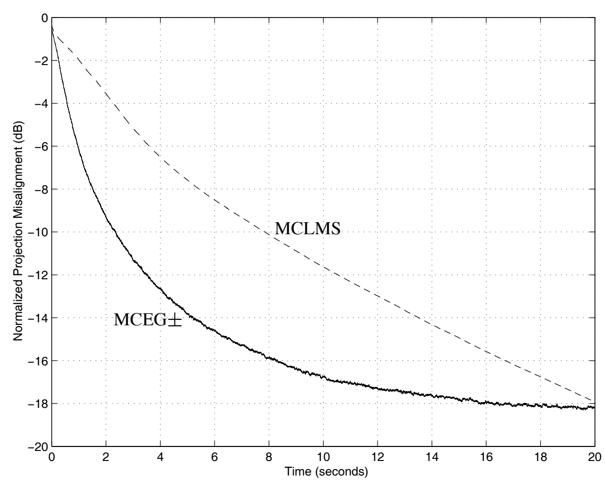

Figure 3: Normalized projection misalignment of the MCLMS and MCEG± algorithms for identifying the single-input three-output system excited by a white Gaussian source and with additive white Gaussian noise at 50 dB SNR.

7. REFERENCES

[1] Y. Huang and J. Benesty, "Adaptive multi-channel least mean square and Newton algorithms for blind channel identifi cation," *Signal Processing*, vol. 82/8, pp. 99–110, Aug. 2002.

[2] S. Haykin, *Adaptive Filter Theory*. Fourth Edition, Upper Saddle River, NJ: Prentice Hall, 2002.

[3] J. Kivinen and M. K. Warmuth, "Exponentiated gradient versus gradient descent for linear predictors," *Inform. Comput.*, vol. 132, pp. 1–64, Jan. 1997.

[4] S. I. Hill and R. C. Williamson, "Convergence of exponentiated gradient algorithms," *IEEE Trans. Signal Processing*, vol. 49, pp. 1208–1215, June 2001.

[5] G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel identifi cation," *IEEE Trans. Signal Processing*, vol. 43, pp. 2982–2993, Dec. 1995.

[6] D. R. Morgan, J. Benesty, and M. M. Sondhi, "On the evalution of estimated impulse responses," *IEEE Signal Processing Lett.*, vol. 5, no. 7, pp. 174–176, July 1998.