ADAPTIVE MATCHED FILTERING OF STEADY-STATE VISUAL EVOKED POTENTIALS

Carlos E. Davila*, Richard Srebro**, Masoud Azmoodeh*, and Ibrahim Ghaleb*
*Electrical Engineering Department, Southern Methodist University, Dallas, Texas 75275 USA
**Dept. of Ophthalmology, University of Texas Southwestern Medical Center at Dallas, Dallas, Texas 75235 USA

ABSTRACT

The eigenfilter is an FIR filter that maximizes signal-to-noise ratio (SNR). It typically consists of the eigenvector associated with the maximum eigenvalue of the data covariance matrix. Alternately, the eigenfilter may incorporate a linear combination of the dominant covariance matrix eigenvectors. Expressions for the eigenfilter SNR gain are derived. An algorithm for adaptive eigenfiltering is then described which has a computational complexity of $O(Md^2)$ where M is the eigenfilter length and d is the signal covariance matrix rank. The algorithm is demonstrated via simulations to out-perform a well-known subspace averaging algorithm having similar computational complexity. The eigenfiltering algorithm is then used to obtain estimates of the single trial steady-state visual evoked potential.

1. INTRODUCTION

The steady-state visual evoked potential (SSVEP) is elicited using a periodic visual stimulus and has a variety of clinical applications [1]. It is usually measured using the ensemble averaging method, which averages an ensemble of SSVEP responses that are time locked to the stimulus. This method is based on the premise that the underlying signal and noise are deterministic, an assumption generally regarded to be unrealistic. More sophisticated estimation techniques have been devised over the past two decades, however these are primarily for the estimation of transient EP's. In this paper, we propose the use of an adaptive eigenfilter (AEF) to improve the signal-to-noise ratio (SNR) of the SSVEP. The AEF utilizes an adaptive FIR filter to maximize SNR without any knowledge of the statistical properties of the signal and noise components [2]. Adaptive filters have been used to estimate transient EP's [3]. They have also been used in an adaptive line enhancement (ALE) configuration for the estimation of the SSVEP [4]. A comb filter, tuned to the stimulus frequency and its harmonics has also been used to estimate the SSVEP [5]. Unlike comb filters, the AEF is optimized to maximize the SNR of the SSVEP and hence is useful in objective sensory thresholding applications which require detection of low-level SSVEP signals.

2. EIGENFILTERING OF COMPLEX SINUSOIDS

It is well known that the eigenfilter is the eigenvector associated with the maximum eigenvalue of the covariance matrix of the data vector [2]. The data vector is the $M \times 1$ vector consisting of the M most recent data samples, $x_n = [\ x(n) \quad x(n-1) \quad \cdots \quad x(n-M+1)\ ]^T$. It is assumed that the measured SSVEP signal consists of a signal, $s(n)$, in additive noise $z(n)$ so that $x_n = s_n + z_n$ where $s_n = [\ s(n) \quad s(n-1) \quad \cdots \quad s(n-M+1)\ ]^T$ and $z_n = [\ z(n) \quad z(n-1) \quad \cdots \quad z(n-M+1)\ ]^T$. Hence, in order to maximize the SNR, $E\left[(w^T s_n)^2\right]/E\left[(w^T z_n)^2\right]$ the eigenfilter w becomes the eigenvector associated with the maximum eigenvalue of the covariance matrix $R^x = E[x_n x_n^H]$. We will consider the general eigenfilter, $w = \sum_{i=1}^{d} \beta_i q^i$ where $q^1, \ldots, q^d$ are the eigenvectors of R^x corresponding to eigenvalues $\lambda^1 \geq \cdots \geq \lambda^d$, respectively. The SNR gain that can be obtained via eigenfiltering is analyzed for the case of complex sinusoids in white noise. The sinusoidal signals are assumed to be given by

$$x(n) = \sum_{k=1}^{d} \rho_k e^{j(\omega_k n + \phi_k)} + z(n) \tag{1}$$

where the ρ_k are real constants and the ϕ_k are random phases. The covariance matrix of x_n is given by [6]

$$R^x = R^s + \sigma_z^2 I_M \tag{2}$$

where $R^s \equiv \mathcal{E} P \mathcal{E}^H$, $\mathcal{E} = [\ e_1 \quad e_2 \quad \ldots \quad e_d\ ]$, $P = diag\,(\rho_1^2, \rho_2^2, \ldots, \rho_d^2)$, and $e_i = [\ 1 \quad e^{j\omega_i} \quad \ldots \quad e^{j(M-1)\omega_i}\ ]^T$, $i = 1, \ldots, d$. The SNR of the eigenfilter output is given by

$$SNR_{out} = \frac{w^H \mathcal{E} P \mathcal{E}^H w}{\sigma_z^2 \sum_{k=1}^{d} |\beta_k|^2} \tag{3}$$

The constants β_k in (3) can be chosen in a number of different ways, depending on the degree of SNR gain versus the desired amount of "equalization" or enhancement allotted to each sinusoidal frequency. Clearly, choosing $\beta_i = 0, \quad i = 2, \ldots, d$ maximizes the SNR at the possible expense of a loss of equalization. Throughout the remainder of this paper, the β_i are the first elements of the signal subspace eigenvectors, hence the eigenfilter effectively corresponds to an orthogonal projection onto the signal subspace, that is, the eigenfilter output is the first entry of $P_s x_n$ where P_s is the projection matrix onto the signal subspace, $P_s = [\ q^1 \quad \ldots \quad q^d\ ][\ q^1 \quad \ldots \quad q^d\ ]^H$. Hence, the eigenfilter output is perfectly equalized. The SNR of the eigenfilter output is therefore

$$SNR_{out} = \frac{\sum_{k=1}^{d} |\beta_k|^2 \left(\lambda^k - \sigma_z^2\right)}{\sigma_z^2 \sum_{k=1}^{d} |\beta_k|^2} \tag{4}$$

This work was supported in part by the Whitaker Foundation and the National Science Foundation under grant BCS-9308028.

0-7803-2431-5/95 $4.00 © 1995 IEEE

An equivalent expression is obtained by using the fact that $P_s s_n = s_n$,

$$SNR_{out} = \frac{\sum_{k=1}^{d} \rho_k^2}{\sigma_z^2 \sum_{k=1}^{d} |\beta_k|^2} \quad (5)$$

Comparing (5) with the expression for the eigenfilter input SNR,

$$SNR_{in} = \frac{\sum_{k=1}^{d} \rho_k^2}{\sigma_z^2} \quad (6)$$

it follows that the eigenfilter increases SNR to the extent that $\sum_{k=1}^{d} |\beta_k|^2 < 1$. Fig. 1 shows SNR gain as a function of $\Delta\omega = \omega_1 - \omega_2$ for $d = 2, M = 16, \rho_1^2 = 2$, and $\rho_2^2 = 1$. For the case of a single complex sinusoid, it suffices to set $\beta_1 = 1$. The signal component of the eigenfilter output is then given by the maximum eigenvalue of $\mathcal{E}P\mathcal{E}^H$, or equivalently by its trace, $M\rho_1^2$. The eigenfilter SNR is therefore,

$$SNR_{out} = \frac{M\rho_1^2}{\sigma_z^2} \quad (7)$$

giving an SNR gain of M.

3. FAST ALGORITHM FOR ADAPTIVE EIGENFILTERING

The proposed algorithm recursively computes estimates of the d dominant eigenpairs, $(q_n^i, \lambda^i(n), i = 1, \ldots d)$ of the sample covariance matrix $R_n = \sum_{i=0}^{n} \xi^{n-i} x_n x_n^H$, $0 < \xi < 1$. The algorithm updates the estimated eigenvectors as

$$\hat{q}_n^i = \sum_{k=1}^{d} \alpha_k^i(n) \hat{q}_{n-1}^k + \sum_{k=d+1}^{2d} \alpha_k^i(n) x_{n-k+d+1}, \quad i = 1, \ldots d \quad (8)$$

where the $\alpha_k^i(n)$ are chosen so that $\hat{q}_n^i$ maximizes the Rayleigh quotient associated with R_n when $i = 1$. For $i > 1$ the $\alpha_k^i(n)$ can be obtained without deflation. The motivation underlying this eigenvector update is that when there is zero noise in the data, the true eigenvector is obtained after only one update since $x_n \ldots x_{n-d+1}$ span the signal subspace [6]. This update can be reduced to a $2d$-dimensional generalized eigenvalue problem. Define the matrix

$$Q_n = \left[\hat{q}_{n-1}^1 \ \cdots \ \hat{q}_{n-1}^d \ x_n \ \cdots \ x_{n-d+1} \right] \quad (9)$$

so that

$$\hat{q}_n^i = Q_n \alpha_n^i, \ i = 1, \ldots, d \quad (10)$$

where $\alpha_n^i = \left[\alpha_1^i(n) \ \ldots \ \alpha_{2d}^i(n) \right]^T$. The Rayleigh quotient to be maximized over α_n^i is given by

$$\mu\left(R_n, Q_n\alpha_n^i\right) = \frac{\alpha_n^{iH} Q_n^H R_n Q_n \alpha_n^i}{\alpha_n^{iH} Q_n^H Q_n \alpha_n^i}, \quad i = 1 \ldots d \quad (11)$$

Hence, the problem of updating $\hat{q}_{n-1}^i$ is reduced to computing a $2d$-dimensional generalized eigendecomposition and the $\alpha_n^1 \ldots \alpha_n^d$ are the generalized eigenvectors solving

$$A\alpha_n^i = \hat{\lambda}^i(n) B \alpha_n^i \quad (12)$$

where $A = Q_n^H R_n Q_n$, $B = Q_n^H Q_n$, and $\hat{\lambda}^i(n)$ is the generalized eigenvalue. Since the α_n^i are conjugate with respect to Q_n (i.e. $\alpha_n^{iH} Q_n^H Q_n \alpha_n^j = 0, \ i \neq j$), it is easy to see that the updated eigenvectors $\hat{q}_n^i, \ i = 1, \ldots, d$ will be orthogonal. Moreover, they will also have the desired minmax properties. Suppose that $\hat{q}_n^1 = Q_n \alpha_n^1$ is an estimate of q^1, it follows that $\hat{q}_n^2 = Q_n \alpha_n^2$ will be an estimate of q^2 given that α_n^2 gives the largest generalized Rayleigh quotient (11) subject to the constraint that α_n^2 be $Q_n^H Q_n$-conjugate with α_n^1. This argument can be inductively applied to the remaining eigenvector estimates.

It remains to show that the matrix-vector products in (11) can be efficiently computed. Consider first the matrix-vector products $R_n x_n$ through $R_n x_{n-d+1}$. These products can be efficiently carried out in $O(M)$ operations by exploiting the shift-invariant property of R_n. An algorithm for doing this has been described in [7] and is listed in Table 1. The remaining matrix-vector products in (11) are $R_n \hat{q}_{n-1}^i, \ i = 1, \ldots d$. These can also be updated in $O(M)$ operations. Assume that at time n, $R_{n-1}\hat{q}_{n-2}^i$ and α_{n-1}^i are available for $i = 1, \ldots, d$. Then

$$R_n \hat{q}_{n-1}^i = \left[\xi R_{n-1} + x_n x_n^H \right] Q_{n-1} \alpha_{n-1}^i \quad (13)$$

Substituting for Q_{n-1} then leads to

$$\begin{aligned} R_n \hat{q}_{n-1}^i = \xi \big[& R_{n-1}\hat{q}_{n-2}^1 \ \cdots \ R_{n-1}\hat{q}_{n-2}^d \ \ R_{n-1}x_{n-1} \\ & \cdots \ R_{n-1}x_{n-d} \ \big] \alpha_{n-1}^i + x_n x_n^H \hat{q}_{n-1}^i \end{aligned} \quad (14)$$

where it has been assumed that α_{n-1}^i has been scaled so that $\hat{q}_{n-1}^i$ has unit norm. Since the only matrix-vector multiplications in (14) involve $R_{n-1}x_{n-1} \ldots R_{n-1}x_{n-d}$, updating $R_n \hat{q}_{n-1}^i, \ i = 1, \ldots d$ can be carried out in $O(M)$ operations. A detailed algorithm listing is given in Table 2, note the total operation count is only $O(Md^2)$. The algorithm is related to $O(M^2 d)$ methods based on the idea of subspace iteration followed by a Ritz acceleration step (SIR) [8] and is hence referred to as the Fast Subspace Iteration with Ritz Acceleration (FSIR) method. The FSIR method can therefore be expected to yield performance similar to that of the $O(M^2 d)$ complexity SIR-type algorithms.

4. EXPERIMENTAL RESULTS

First, the proposed FSIR algorithm was compared with the $O(Md^2)$ subspace averaging (SA) algorithm of Karasolo [9]. The number of complex sinusoids was set to $d = 2$ with $M = 20$, $\omega_1 = 0.6\pi, \omega_2 = 0.8\pi, \xi = 0.99, \rho_1 = \rho_2 = 1$, and $\sigma_z^2 = 1$. Each sinusoid was assigned a uniformly distributed random phase. At time $n = 300$, the frequencies were changed to $\omega_1 = 0.6\pi, \omega_2 = \pi$. The squared error norm

$$\epsilon(n) = \frac{1}{M} \left\| \left[\ q_n^1 \ \ q_n^2 \ \right] - \left[\ \hat{q}_n^1 \ \ \hat{q}_n^2 \ \right] \right\|_F^2 \quad (15)$$

was computed over 1000 iterations and averaged over 50 independent trials. The true eigenvectors q_n^1 and q_n^2 were computed from R_n directly using the Matlab routine "svd". The results are shown in Fig. 2. The FSIR algorithm is seen to have a faster rate of convergence and a lower steady-state error than the SA algorithm.

The eigenfilter, computed via FSIR, was then applied to two sinusoids with $\rho_1^2 = 2, \rho_2^2 = 1, M = 16, \sigma_z^2 = 1, \omega_1 = 0.4\pi, \omega_2 = 0.64\pi, \xi = 0.99$, and $\sigma_z^2 = 1$. At time $n =$

300, the frequencies were changed to $\omega_1 = 0.4\pi$, and $\omega_2 = \pi$. The estimated SNR versus iteration number is shown in Fig. 3 for the eigenfilter input and output, along with the theoretical SNR as predicted by (5) and (6). Close agreement between estimated and theoretical SNR is seen.

The EEG was recorded from the O_Z site referenced to F_{p2} while the subject sat watching a Grass PS22 photic flash stimulator running at 10 flashes/second. The EEG was amplified and filtered to 75Hz and then sampled at 200Hz over a 10 second interval. The FSIR AEF filter length was set to $M = 150$ with $\xi = 0.9998$ and $\beta_1 = 1, \beta_k = 0, k > 1$. The SNR for the unprocessed and enhanced data was estimated by computing the "area" of the signal and noise portions of averaged periodograms of the AEF input and output. The results are shown in Table 3 for the FSIR eigenfilter, a block-processed eigenfilter (10s block), and an adaptive line enhancer ($M = 150, \mu = 10^{-4}$)[2]. This choice of LMS convergence parameter μ provides the same adaptation rate as the FSIR AEF [7]. It can be seen that the eigenfilter produces a higher SNR gain than the ALE. Averaged periodograms for the eigenfilter, and ALE are shown in Fig. 4.

5. SUMMARY

An algorithm for efficiently tracking signal subspace eignevectors of sinusoidal data was described. The estimated eigenvectors can be incorporated into an adaptive eigenfilter for maximizing SNR. The eigenfilter was used to estimate steady-state visual evoked potentials.

6. REFERENCES

[1] D. Regan, *Evoked Potentials in Psychology, Sensory Physiology, and Clinical Medicine.* Wiley, 1972.

[2] S. Haykin, *Adaptive Filter Theory.* Prentice Hall, 1991.

[3] N. Thakor, "Adaptive filtering of evoked potentials," *IEEE Transactions on Biomedical Engineering,* vol. BME-34, pp. 6–12, 1987.

[4] C. Davila, A. Abaye, and A. Khotanzad, "Estimation of single-sweep steady state visual evoked potentials by adaptive line enhancement," *IEEE Transactions on Biomedical Engineering,* vol. BME-41, pp. 197–200, Feb. 1994.

[5] T. Collura, "Real-time filtering for the estimation of steady-state visual evoked potentials," *IEEE Transactions on Biomedical Engineering,* vol. BME-37, pp. 650–652, June 1990.

[6] S. Kay, *Modern Spectral Estimation.* Prentice Hall, 1988.

[7] C. Davila, "Line search algorithms for adaptive filtering," *IEEE Trans. Signal Proc.,* vol. SP-41, pp. 2490–2494, July 1993.

[8] P. Comon and G. Golub, "Tracking a few extreme singular values and vectors in signal processing," *Proc. IEEE,* vol. 78, pp. 1327–1343, Aug. 1990.

[9] I. Karasolo, "Estimating the covariance matrix by signal subspace averaging," *IEEE Trans. Acoust., Speech, Signal Processing,* vol. ASSP-34, pp. 8–12, Feb. 1986.

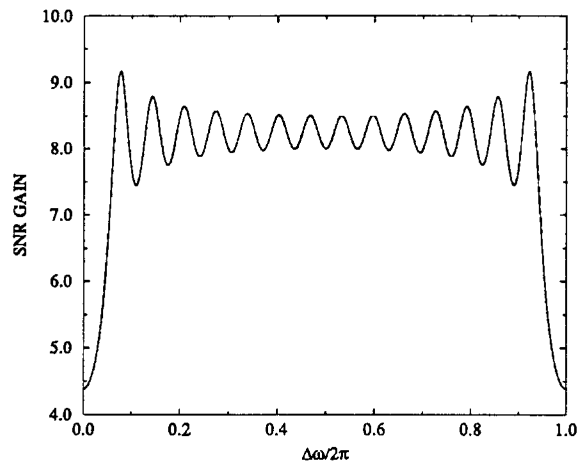

Figure 1: SNR gain versus $\Delta\omega$ for $d = 2$.

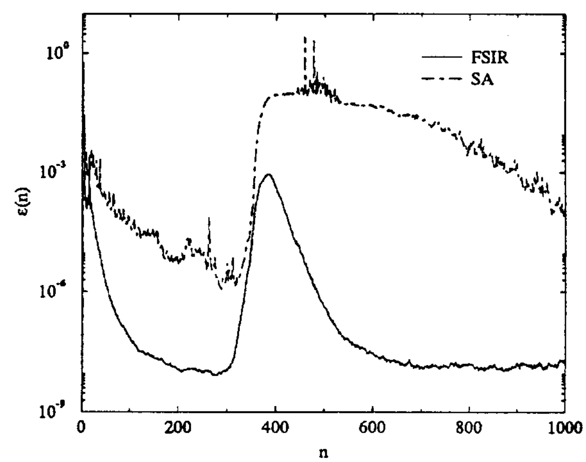

Figure 2: Eigenvector error norms for the proposed FSIR algorithm and Karasolo's subspace averaging (SA) algorithm.

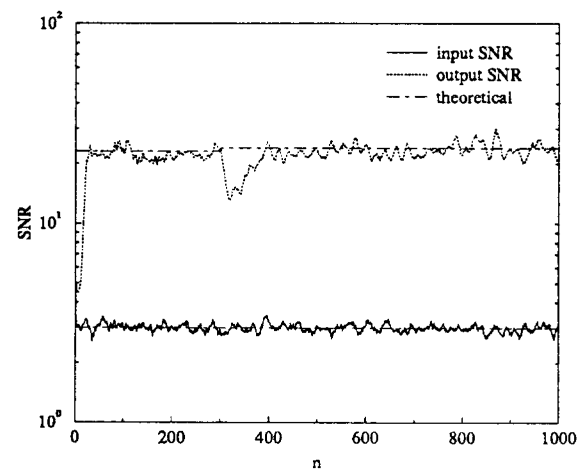

Figure 3: Eigenfilter input and output estimated SNR's for two sinusoids.

	operation count
For $n = 1, 2, \ldots$	
$r_n = \xi r_{n-1} + x(n-M)x_n$	$3M$
$\tilde{r}_n = \xi \tilde{r}_{n-1} + x(n)x_{n-1}$	$3M$
$r(n) = \xi r(n-1) + x(n)^2$	3
$[\overline{g}_b]_1 = r(n)y(n) + \tilde{r}_n^H y_{n-1}$	$2M+2$
$[\overline{g}_b]_{2,M+1} = \tilde{r}_n y(n) + g_{n-1}$	$2M$
$g_n = [\overline{g}_b]_{1,M} - r_n y(n-M)$	$\underline{2M}$
	$12M+5$
Initial Values :	
$g_0 = 0_{M\times 1}$	
$r(0) = 0.01$	
$r_0 = 0_{M\times 1}$	
$\tilde{r}_0 = 0_{M\times 1}$	

Table 1: Algorithm for updating $g_n = R_n y_n$ showing number of operations. Here, y_n is replaced with $x_n, \ldots, x_{n-d+1}$.

Algorithm	SNR Gain (dB)
FSIR AEF	8.5
Block AEF	9.0
LMS ALE	6.1

Table 3: SSVEP data SNR estimates.

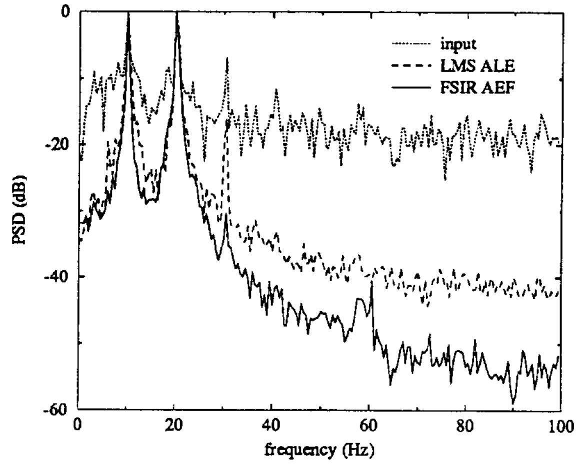

Figure 4: SSVEP PSD estimates for the proposed FSIR eigenfilter and an LMS-based ALE.

	operation count
For $n = 1, 2, \ldots$	
update $R_n x_{n-i}$, $i = 0, \ldots, d-1$ (*see Table* 1)	$12Md + 5d$
$U = \left[\begin{array}{cccccc} R_{n-1}\hat{q}_{n-2}^1 & \ldots & R_{n-1}\hat{q}_{n-2}^d & R_{n-1}x_{n-1} & \ldots & R_{n-1}x_{n-d} \end{array}\right]$	
$\theta^i = \alpha_{n-1}^i / \delta^i(n-1)$, $i = 1, \ldots, d$	$2d^2$
$\gamma^i = x_n^H q_{n-1}^i$, $i = 1, \ldots, d$	$2Md$
$R_n \hat{q}_{n-1}^i = \xi U \theta^i + x_n \gamma^i$, $i = 1, \ldots, d$	$4Md^2 + 3Md$
form the matrix A (*see* (11) − (12))	$4Md^2 + 2Md$
form the matrix B (*see* (11) − (12))	$4Md^2 + 2Md$
Solve $A\alpha_n^i = \hat{\lambda}^i(n) B \alpha_n^i$, $i = 1, \ldots, d$	$O(d^3)$
$Q_n = \left[\begin{array}{cccccccc} \hat{q}_{n-1}^1 & \hat{q}_{n-1}^2 & \ldots & \hat{q}_{n-1}^d & x_n & x_{n-1} & \ldots & x_{n-d+1} \end{array}\right]$	
$\hat{q}_n^i = Q_n \alpha_n^i$, $i = 1, \ldots, d$	$4Md^2$
$\delta^i(n) = (\hat{q}_n^{iH} \hat{q}_n^i)^{1/2}$, $i = 1, \ldots, d$	$2Md + d$
$\hat{q}_n^i = \hat{q}_n^i / \delta^i(n)$, $i = 1, \ldots, d$	$\underline{Md}$
	$16Md^2 + 24Md$
	$+2d^2 + 6d + O(d^3)$
Initial Values :	
$\hat{q}_{-1}^i$, $i = 1, \ldots, d$: *random, orthonormal*	
$R_0 \hat{q}_{-1}^i = 0.01 \times \hat{q}_{-1}^i$, $i = 1, \ldots, d$	
$\delta^i(0) = 1$, $i = 1, \ldots, d$	
$\alpha_0^i = \left[\begin{array}{ccccccc} 0 & \ldots & 0 & 1 & 0 & \ldots & 0 \end{array}\right]^T$, 1 *in the* i^{th} *position*	

Table 2: Algorithm for updating $\hat{q}_{n-1}^i$, $i = 1, \ldots, d$, showing number of operations. The $d \times d$ block matrix in B corresponding to $\hat{q}_{n-1}^{iH} \hat{q}_{n-1}^j$ can be replaced by the identity matrix, I_d.