AN ADAPTIVE VARIABLE STEP-SIZE PRE-FILTER BANK ALGORITHM FOR COLORED ENVIRONMENTS

Ting Liu and Saeed Gazor

Dept. of Electrical and Computer Engineering
Queen's University
Kingston, Ontario, Canada K7L 3N6

ABSTRACT

A Variable Step-size (VS) algorithm is proposed based on the Pre-Filter Bank (PFB) Adaptive algorithm, first introduced by Courville and Duhamel [4]. The proposed algorithm adjusts the step-sizes of the subbands by using a simplified version of the Benveniste procedure [5]. As the filter banks are commonly narrow-band filters, their non-decimated outputs are highly correlated. This correlation allows us to approximate the subband autocorrelation matrices by single rank matrices, thus permitting us to simplify and reduce the computational complexity of the VS procedures. The proposed inexpensive algorithm is very efficient in terms of tracking capabilities and initial learning for environments with colored additive noise and colored input signal.

1. INTRODUCTION

A common problem often encountered in many applications of the LMS algorithm is that it suffers from slow convergence when the input signal to the adaptive filter is correlated. A pre-filter structure is introduced in [3] to solve this problem by applying a decorrelation filter jointly to the pair of input-outputs to jointly whiten the additive noise and the input signal. This structure operates based on filtered signals to make the algorithm faster and to reduce the Mean-Square Error (MSE). One may use different decorrelation filters, either in parallel like a filter bank or in serial, to obtain different estimates of the optimal weight target W_o, and then combine these estimates. If we apply appropriate downsamplers the result can be viewed as the subband adaptive algorithm as proposed in [4]. In particular, this PFB algorithm minimizes a weighted criterion of squared errors in subbands. The step size of each subband must be adjusted to obtain fast convergence, at the expense of some minor increase in computational complexity.

The performance of the LMS algorithm using a fixed step-size may not be satisfactory in some environments involving colored noise, colored input signal and/or time-varying plant, for instance, for a time-varying communication channel or when the additive noise is colored and non-stationary. To deal with this problem, a number of VS algorithms have been developed, *e.g.*, [6, 7]. In this paper in the context of PFB, we employ a set of time-varying step sizes as proposed for the standard LMS weight update recursion to each subband. A simplification is then introduced to this procedure based on the near orthogonality of the subband input signals. Computational complexity is reduced without any perceptible loss in performance.

Corresponding author: Telephone: (613) 533-6068; Fax: (613) 533-6515; Email: liut@ee.queensu.ca.

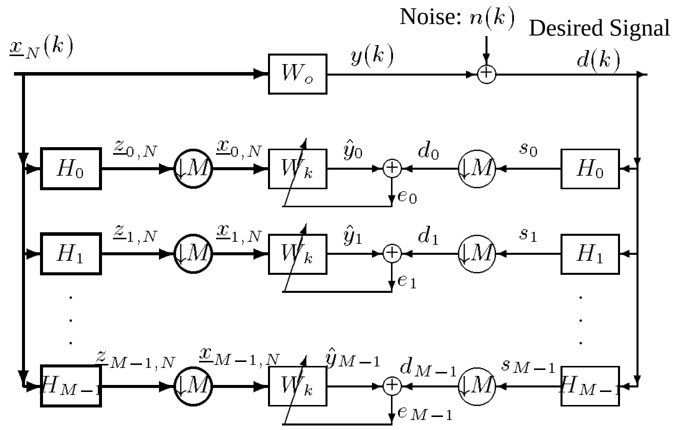

Fig. 1. The structure of an adaptive filter using filter bank.

The remainder of the paper is organized as follows. Section 2 briefly states the structure and procedures of the pre-filter bank algorithm. The new VS pre-filter bank algorithm is proposed in Section 3.1, and Section 3.2 derives its simplified version. These algorithms are compared with some simulations in Section 4. In Section 5, the conclusions of the paper are drawn.

2. PRE-FILTER BANK ALGORITHM

Figure 1 shows the block diagram of the new adaptive approach using an analysis filter bank. A group of linear filters $\{\underline{h}_i\}_{i=0}^{M-1}$ is applied jointly to the input and desired signals followed by downsamplers with a factor of M. Suppose that the tap-weight vector has length N, each filter in the filter bank has length L, and M is the number of bands. The signals $\underline{x}_{i,N}(k)$ and $d_i(k)$ are the outputs of the analysis filter banks, and $\underline{z}_{i,N}$ and $s_i(k)$ are the signals before the downsamplers. In this paper, the underline notation $\underline{y}_N(k)$ is used to denote the vector $[y(k), y(k-1), \cdots, y(k-N+1)]^T$, where the subscript N is the vector length.

From the Pre-Filter Theorem given in [3], we can conclude that if $n(k)$ and $\underline{x}_N(k)$ are uncorrelated, the Wiener filter obtained from each subband pairs $\{d_i(k), \underline{x}_{i,N}(k)\}$ and from the pair $\{d(k), \underline{x}_N(k)\}$ are identical. This means that the information from all subbands points to a unique optimal Wiener solution. Therefore, the outputs of all analysis filters can be used to obtain unbiased estimators for the unknown plant W_o.

The idea of the approach in [4] is to adaptively update $W(k)$

Table 1. Summary of the filter bank adaptive algorithm [4].

Filtering: for $l = 0, \cdots, N-1$, and $i = 0, \cdots, M-1$,
$z_i(kM-l) = \sum_{j=0}^{L-1} h_{i,j} x(kM-l-j)$, $d_i(k) = s_i(kM) = \sum_{j=0}^{L-1} h_{i,j} d(kM-j)$, $y_i(k) = W_k^H \underline{x}_{i,N}(k)$,
Error estimation:
$e_i(k) = d_i(k) - y_i(k)$,
Tap-weight vector adaptation:
$W(k+1) = W(k) + \sum_{i=0}^{M-1} \mu_i \underline{x}_{i,N}(k) e_i^*(k)$.

to minimize $J = \sum_{i=0}^{M-1} \lambda_i E[|e_i(k)|^2]$, where λ_i is the weight for each subband, which is a positive number, leading to the following recursion equation:

$$\begin{aligned} W(k+1) &= W(k) + \textstyle\sum_{i=0}^{M-1} \mu_i \underline{x}_{i,N}(k)\, e_i^*(k), \\ &= W(k) + \textstyle\sum_{i=0}^{M-1} \mu_i \underline{z}_{i,N}(kM)\, e_i^*(k), \end{aligned} \quad (1)$$

where $\underline{z}_{i,N}(kM) = [z_i(kM) \;\; z_i(kM-1) \;\; \cdots \;\; z_i(kM-N+1)]$ is the signal in subbands before the downsampler, $\mu_i = \mu\lambda_i$ is the step size of each subband, and $(.)^*$ is the conjugate operation. This adaptive filter bank algorithm is summarized in Table 1.

3. VS PRE-FILTER BANK ADAPTIVE ALGORITHMS

3.1. VS Pre-filter Bank Algorithm (VSPFB)

To get better performance for the non-stationary environment, we introduce the proposed simplified version of Benveniste's algorithm [5] to each subband LMS adaptation, and derive the corresponding VS pre-filter bank algorithm based on Benveniste's approach (VSPFB). In each subband, using the definition $\Psi_i(k) = \frac{\partial W(k)}{\partial \mu_i(k)}$ as in [5], we have

$$\Psi_i(k) = \alpha \Psi_i(k-1) + e_i^*(k-1)\underline{x}_{i,N}^H(k-1), \quad (2)$$

$$\mu_i(k) = \mu_i(k-1)[1 + \rho_i \mathrm{Re}\{e_i(k)\underline{x}_{i,N}^H(k)\Psi_i(k)\}], \quad (3)$$

where α is a constant positive number which is smaller than but close to 1. This α is used to replace $[I - \mu(n-1)\underline{x}(n-1)\underline{x}^H(n-1)]$ in Benveniste's algorithm [5]. Ψ_i can be initialized as $\underline{0}$ at the beginning of the adaptation. ρ_i is the step-size for the adaptation of μ_i. Table 2 gives the adaptation procedures of this variable step size algorithm.

If we use the normalized LMS algorithm instead of the LMS algorithm, (3) becomes

$$\Psi_i(k) = \alpha \Psi_i(k-1) + \frac{e_i^*(k-1)\underline{x}_{i,N}^H(k-1)}{\|\underline{x}_{i,N}(k-1)\|^2}. \quad (4)$$

Furthermore, to take more uniform control of the step size parameters, we can use the step-normalization technique [1], that is, normalizing the parameter ρ_i with respect to the estimate of the corresponding gradient power. This is done by replacing ρ_i with $\rho_o/\phi_i(k)$, $i = 0, \cdots, M-1$, where ρ_o is a common constant, and $\phi_i(k)$ is an estimate of the energy of the subband input signal. The estimated energy $\phi_i(k)$ may be obtained by using the recursion $\phi_i(k) = a\phi_i(k-1) + (1-a)\|\underline{x}_{i,N}(k)\|^2$, where a is a constant number smaller than but close to 1. Table 3 summarizes this normalized variable step size pre-filter bank algorithm (NVSPFB).

Table 2. The proposed VSPFB Adaptive Algorithm.

Step-size adaptation:
$\Psi_i(k) = \alpha \Psi_i(k-1) + e_i^*(k-1)\underline{x}_{i,N}^H(k-1)$, $\mu_i(k) = \mu_i(k-1)[1 + \rho_i \mathrm{Re}\{e_i(k)\underline{x}_{i,N}^H(k)\Psi_i(k)\}]$.
Tap-weight vector adaptation:
$W(k+1) = W(k) + \sum_{i=0}^{M-1} \mu_i(k) e_i^*(k) \underline{x}_{i,N}(k)$.

Table 3. The proposed NVSPFB Adaptive Algorithm.

Step-size adaptation:
$\Psi_i(k) = \alpha \Psi_i(k-1) + \frac{e_i^*(k-1)\underline{x}_{i,N}^H(k-1)}{\|\underline{x}_{i,N}(k-1)\|^2}$, $\phi_i(k) = a\phi_i(k-1) + (1-a)\|\underline{x}_{i,N}(k)\|^2$ $\mu_i(k) = \mu_i(k-1)[1 + \frac{\rho_o}{\phi_i(k)} \mathrm{Re}\{e_i(k)\underline{x}_{i,N}^H(k)\Psi_i(k)\}]$.
Tap-weight vector adaptation:
$W(k+1) = W(k) + \sum_{i=0}^{M-1} \mu_i(k) e_i^*(k) \underline{x}_{i,N}(k)$.

3.2. Simplified VSPFB (SVSPFB) Adaptive Algorithm

For each subband in the algorithm described in Table 2, it is required to perform $(2N+4)$ multiplications to update $\mu_i(k)$. For the normalized version, we need $(N+3)$ extra multiplications. This increase in the computational complexity is a significant burden. To make the algorithm more practical, some simplification is needed. Suppose ξ_i is the dominant eigenvalue of R_i, which is related to the maximum value of the power spectral density of the subband input signal. If the subband input signal is smooth enough in its pass-band (which can be achieved by adjusting the number of bands and the range of pass-bands), we can approximate

$$R_i \approx \xi_i \underline{v}_i \underline{v}_i^H, \quad (5)$$

where $\underline{v}_i$ is the eigenvector corresponding to the biggest eigenvalue ξ_i. Since the filters have different pass-bands, we can argue that $< \underline{v}_i, \underline{v}_j > \approx 0$ if $i \neq j$. This approximation allows us to do the projection to reduce the computational complexity. Under this assumption, we have

$$\underline{x}_{i,N}(k) \approx \eta_i(k)\underline{v}_i, \quad (6)$$

where $\eta_i = \underline{v}_i^H \underline{x}_{i,N}(k)$ is the projection of $\underline{x}_{i,N}$ onto the vector $\underline{v}_i$. Actually by simulation we will see in section 4 that the application of this approximation has a markable beneficial effect even

Table 4. The proposed SVSPFB Adaptive Algorithm.

Step-size adaptation:

$$\begin{aligned}\beta_i(k) &= \underline{v}_i^H e_i^*(k)\underline{x}_{i,N}(k),\\ \gamma_i(k) &= \alpha\gamma_i(k-1)+\beta_i(k-1),\\ \mu_i(k) &= \mu_i(k-1)[1+\rho\mathrm{Re}\{\beta_i^*(k)\gamma_i(k)\}].\end{aligned}$$

Tap-weight vector adaptation:

$$W(k+1)=W(k)+\sum_{i=0}^{M-1}\mu_i(k)e_i^*(k)\underline{x}_{i,N}(k).$$

Table 5. The proposed SNVSPFB Adaptive Algorithm.

Step-size adaptation:

$$\begin{aligned}\beta_i(k) &= \underline{v}_i^H e_i^*(k)\underline{x}_{i,N}(k),\\ \zeta_i(k) &= \underline{v}_i^H \underline{x}_{i,N}(k),\\ \gamma_i(k) &= \alpha\gamma_i(k-1)+\frac{\beta_i(k-1)}{\zeta_i^2(k-1)},\\ \phi_i(k) &= a\phi_i(k-1)+(1-a)\zeta_i^2(k)\\ \mu_i(k) &= \mu_i(k-1)[1+\frac{\rho_o}{\phi_i(k)}\mathrm{Re}\{\beta_i^*(k)\gamma_i(k)\}].\end{aligned}$$

Tap-weight vector adaptation:

$$W(k+1)=W(k)+\sum_{i=0}^{M-1}\mu_i(k)e_i^*(k)\underline{x}_{i,N}(k).$$

if the eigenvalues of R_i do not differ a lot in practice. The orthogonal vectors $\underline{v}_i$ are to be predetermined after the design of the filter banks by eigen-decomposition of R_i for a white input noise or for some real data available.

In detail, we first approximate $e_i^*(k)\ \underline{x}_{i,N}$ using

$$e_i^*(k)\underline{x}_{i,N} \approx \beta_i(k)\underline{v}_i, \qquad (7)$$

where

$$\beta_i(k) = < e_i^*(k)\underline{x}_{i,N}, \underline{v}_i > . \qquad (8)$$

From (3), we can conclude that $\Psi_i(k)$ should converge in the same direction as $\underline{x}_{i,N}(k)$. We can approximate

$$\Psi_i(k) \approx \gamma_i(k)\underline{v}_i. \qquad (9)$$

Substitute (7) and (9) into (3), yielding

$$\gamma_i(k)\underline{v}_i = \alpha\gamma_i(k-1)\underline{v}_i + \beta_i(k-1)\underline{v}_i. \qquad (10)$$

By multiplying (10) from the left by $\underline{v}_i^H$, we get

$$\gamma_i(k) = \alpha\gamma_i(k-1)+\beta_i(k-1). \qquad (11)$$

In the same way, the step size adaptation equation is also simplified to the following form:

$$\mu_i(k) = \mu_i(k-1)[1+\rho\mathrm{Re}\{\beta_i^*(k)\gamma_i(k)\}]. \qquad (12)$$

Drawing from all of the above, the proposed simplified VSPFB (SVSPFB) algorithm is summarized in Table 4. Once again, applying the same technique to the normalized version and defining $\zeta_i(k) = \underline{v}_i^H\underline{x}_{i,N}(k)$, the Simplified Normalized VS Pre-Filter Bank (SNVSPFB) algorithm is summarized in Table 5.

Consider the algorithm in Table 4. In each step to get a new step size in each subband we need a total of $(N+4)$ multiplications, reducing N multiplications from the VSPFB algorithm. Also, for the simplified normalized case, we get the same amount of complexity reduction. It is apparent that the complexity is reduced by almost half.

4. SIMULATION RESULTS

In this section, three examples are considered. In each of example, the length of the plant is fixed at $N = 11$. The filter bank we use has three bands with length 11. The time constant, the Mean Square Coefficients Error (MSCE), $\|W_k - W_o\|^2$ of all three algorithms are computed and listed in Table 6 for comparison. All results presented are based on an ensemble average of 15 independent simulation runs.

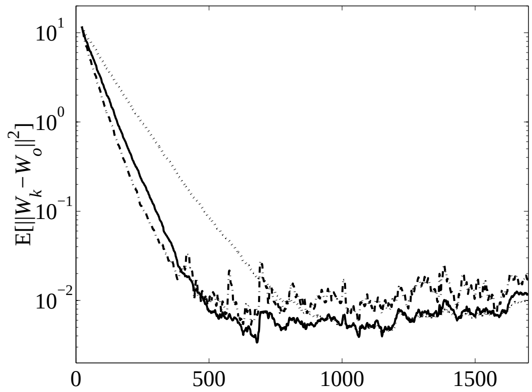

Fig. 2. Example 1: MSCE using traditional PFB algorithm (dotted), VSPFB algorithm (dashdot) and its simplified version (solid).

Example 1: In this example, the noise is white Gaussian with variance 0.1, and the input signal $x(k)$ is colored and follows the AR(1) model:

$$x(k) = \epsilon(k) - 0.98x(k-1),$$

where $\epsilon(k)$ is a Gaussian noise with variance 4. The time-varying tap-weight vector $W_o(k)$ is chosen to be a multivariate random-walk process characterized by the difference equation

$$W_o(k+1) = W_o(k) + \chi(k),$$

where $\chi(k)$ is an i.i.d. random process vector. The sequences $n(k)$, $\chi(k)$ and $x(k)$ are independent zero-mean and stationary random processes. From Figure 2 and Table 6 we find both the VSPFB and its simplified version outperform the traditional pre-filter bank algorithm. The VSPFB converges somewhat faster than its simplified version, but its MSCE is slightly higher. Considering the complexity reduction resulting from the use of the simplified VSPFB, we conclude that the SVSPFB algorithm is preferred.

Example 2: In many applications such as in the case of echo cancellation, the signal and noise have almost similar power spec-

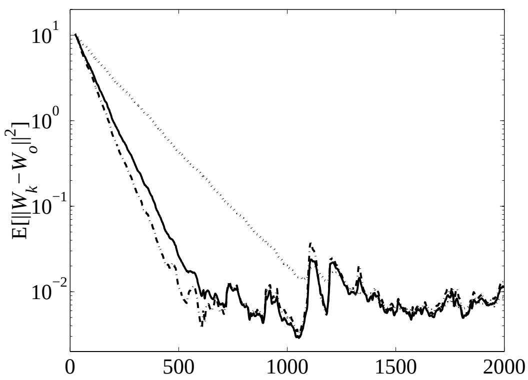

Fig. 3. Example 2: MSCE using traditional PFB algorithm (dotted), VSPFB algorithm (dashdot) and its simplified version (solid).

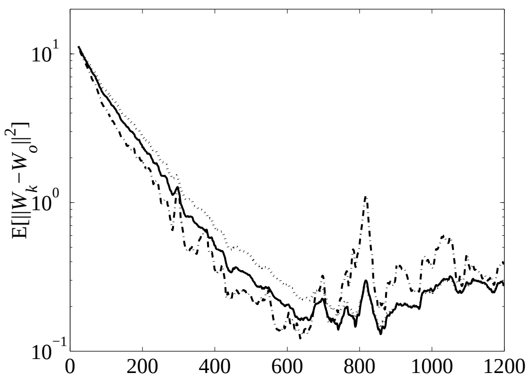

Fig. 4. Example 3: MSCE using traditional PFB algorithm (dotted), VSPFB algorithm (dashdot) and its simplified version (solid).

tra. In this example, the noise we use follows the same model as the input signal. We have

$$\begin{aligned} n(k) &= \epsilon_n(k) - 0.98n(k-1), \\ x(k) &= \epsilon_x(k) - 0.98x(k-1), \end{aligned}$$

where $\epsilon_n(k)$ and $\epsilon_x(k)$ are i.i.d. zero-mean Gaussian sequences with variance 0.1 and 2, respectively. The tap-weight vector is still time varying as in Example 1. The learning curves are plotted in Figure 3. It is obvious that the three algorithms have almost the same MSCE, but the VSPFB and its simplified version have much faster convergence.

Example 3**:** As opposed to the second example, in this example, the noise and the input signal are both colored, but with dissimilar power spectral densities, *i.e.*, the signal $x(k)$ is highpass and noise $n(k)$ has a lowpass power spectrum. The signals are generated by

$$\begin{aligned} n(k) &= \epsilon_n(k) + 0.98n(k-1), \\ x(k) &= \epsilon_x(k) - 0.98x(k-1), \end{aligned}$$

where $\epsilon_n(k)$ and $\epsilon_x(k)$ are same as in Example 2. Again, using the time varying plant as in the two previous examples, we get Figure 4. The convergence rate of the simplified algorithm lies between those of the PFB and the VSPFB. The MSCE of the simplified algorithm is almost the same as the PFB's, but the MSCE in the VSPFB algorithm is slightly higher and seems somewhat unstable.

Table 6. Performance comparison of the PFB, VSPFB and its simplified version.

	Time Constant (# of iterations)			$\|W_k - W_o\|^2$ (in dB)		
	PFB	VSPFB	SVSPFB	PFB	VSPFB	SVSPFB
Ex. 1	87	45.0	54	-22.4	-21.0	-22.4
Ex. 2	151	68	79	-22.0	-21.1	-21.8
Ex. 3	147	124	133	-6.2	-4.1	-6.1

5. CONCLUSIONS

The VS pre-filter bank algorithm (VSPFB) introduced in Section 3 speed up the initial convergence compared to the traditional pre-filter bank algorithm by making the step sizes in each subband adaptive, but the computational complexity is also increased at the same time. By using a projection, approximating the vector computations by scalar computations, we propose a simplified version of the VSPFB (SVSPFB), which reduces by almost half the complexity in the step-size adaptation without apparent performance compromise. From simulations, we conclude that the proposed simplified VS adaptive algorithm is a very efficient algorithm for environments in which the signals involved are colored and the plant is non-stationary.

6. REFERENCES

[1] B. Farhang-Borouheny, *Adaptive Filters, Theory and Applications*, John Wiley and Sons Ltd., Chichester, 1999.

[2] M. Mboup, M. Bonnet, and N. Bershad, “LMS coupled adaptive prediction and system identification: a statistical model and transient mean analysis,” *IEEE Trans. on Signal Processing*, vol. SP-42, no. 10, pp. 2607-2615, Oct. 1994.

[3] S. Gazor and T. Liu, “Fast Adaptive Filtering with Adaptive De-correlation,” submitted to *IEEE Trans. on Signal Processing*, Nov., 2001.

[4] M. Courville and P. Duhamel, “Adaptive filtering in subbands using a weighted criterion,” *IEEE Trans. on Signal Processing*, vol. SP-46, no. 9, pp. 2359-2371, Sep. 1998.

[5] W. Ang and F. Farhang-Boroujeny, “A new class of gradient adaptive step-size LMS algorithms,” *IEEE Trans. on Signal Processing*, vol. SP-49, no. 4, pp. 805-810, April 2001.

[6] T. Aboulnasr and K. Mayyas, “A robust variable step-size LMS-type algorithm: analysis and simulations,” *IEEE Trans. on Signal Processing*, vol. SP-45, no. 3, pp. 631-639, March 1997.

[7] V. J. Mathews and Z. Xie, “A stochastic gradient adaptive filter with gradient adaptive step size,” *IEEE Trans. on Signal Processing*, vol. SP-41, pp. 2075-2087, June 1993.