PAPER Special Section of Selected Papers from the 14th Workshop on Circuits and Systems in Karuizawa

Multirate Repeating Method for Alias Free Subband Adaptive Filters

Kiyoshi NISHIKAWA[†a)] ***and*** **Hitoshi KIYA**[†], ***Regular Members***

SUMMARY In this paper, we propose the multirate repeating method for alias free subband adaptive filters (AFSAFs) and consider its convergence property. It is shown that we can adjust the convergence speed and the final error of the adaptive filters by varying its two parameters according to the requirements of the applications where the method is applied. The proposed method has two parameters, namely, the number of channel and the number of repetition. We show that by increasing the number of channels we can reduce the final error, and this property is preferred when the signal-to-noise ratio (SNR) is low. On the other hand, we show that the convergence speed of the AFSAF approaches to that of the affine projection algorithm (APA) by increasing the number of repetition. Through the computer simulations, we show the effect of the proposed method.

key words: *subband apaptive system, alias free, normalized LMS, multirate repeating method*

1. Introduction

This paper proposes a new method for increasing the rate of convergence of the adaptive filter based on the alias free subband adaptive filer (AFSAF) structure [2]. The proposed method uses the concept of the multirate repeating method (MRM) [1] that can increase the rate of convergence of the conventional subband adaptive filters.

The AFSAF is a recently proposed structure of subband adaptive system [2]. The major advantage of the AFSAF is that the system is not affected by the aliasing components generated in the decimation process. However, its convergence speed is slower than that of the affine projection algorithm (APA) when the input signal is highly correlated. The APA [3]–[6] is a variant of the LMS algorithm and it is known that it can improve the rate of convergence when the input signal is highly correlated. The APA uses the concept of orthogonal projection, and one can select the order of APA when implement it. By increasing the order, a higher order AR process can be decorrelated.

Although the APA enables us to decorrelate the input process, it is known that the APA is largely affected by the additive noise [7]. Moreover, the APA requires the matrix inversion to implement so that the required computational complexity becomes larger as the order increases. Note that fast APAs are proposed so far [4]–[6], however, there exist some numerical problems when the short word-length is used to implement.

In order to increase the convergence speed, in this paper, we consider applying the concept of the MRM for the AFSAF system. The MRM [1] is a method for improving the convergence speed of the subband adaptive systems using the redundant data discarded in the decimation process. However, we cannot apply the MRM directly to the AFSAF system because its decimation operations work in a different way from those in the conventional subband adaptive systems. Therefore, we propose a method that enables us to apply the concept of the MRM for AFSAF system. For that purpose, we show an equivalent representation of the AFSAF using a fullband adaptive filter. Then, the proposed method is derived by transforming the formula used in the AFSAF for adaptation based on the representation.

The proposed method has two parameters; namely the number of channels and the number of repetition. We show that by adjusting these parameters we can increase the rate of convergence, or decrease the final errors, or both. We show the following points in this paper: (1) the convergence speed of the AFSAF approaches to that of the APA as the number of repetition increases. (2) On the other hand, we can suppress the effect of additive noise by increasing the number of channels. Therefore, we can adjust the convergence characteristics by selecting a suitable value for the number of repetition and the number of channels according to the environments where the proposed method is applied.

This paper is organized as follows. In the next section, we give a brief description of the conventional methods. The proposed method is then described in Sect. 3. We provide some results of computer simulation using the proposed method in Sect. 4. From the results we confirm its effects. Further, we consider the computational complexity for implementing the proposed method.

2. Review of Conventional Methods

Here, let us review the conventional methods, namely, the alias free subband adaptive filters, multirate repeat-

Manuscript received June 28, 2001.
Manuscript revised October 9, 2001.
Final manuscript received December 20, 2001.

[†]The authors are with the Department of Electrical Engineering, Tokyo Metropolitan University, Hachioji-shi, 192-0397 Japan.
a) E-mail: kiyoshi@eei.metro-u.ac.jp

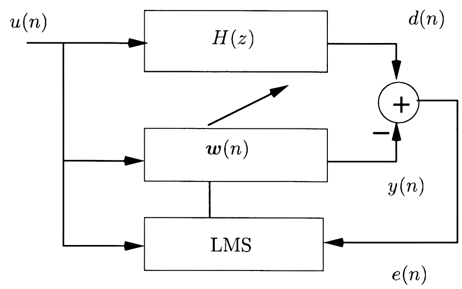

Fig. 1 Configuration of the system identification considered in this paper.

ing method, and affine projection algorithm as a preparation.

Before that, we summarize the notations used in this paper. We use $u(n)$ and $d(n)$ to indicate the input and desired signals at time n respectively. Column vectors $\boldsymbol{u}(n)$ and $\boldsymbol{w}(n)$ are used to indicate tap-input vector and the coefficients of the adaptive filter, and they are defined as

$$\boldsymbol{u}(n) = [u(n) \;\; u(n-1) \ldots u(n-N+1)]^T \quad (1)$$

$$\boldsymbol{w}(n) = [w_0(n) \;\; w_1(n) \ldots w_{N-1}(n)]^T \quad (2)$$

where $w_i(n)$ is i-th impulse response of adaptive filter $\boldsymbol{w}(n)$ and N shows the length of $\boldsymbol{w}(n)$. Note that we only consider the tapped delay line structure.

In Fig. 1, we show the system identification model considered in this paper. In this figure $H(z)$ and $d(n)$ are the transfer function of the unknown system and the desire signal respectively.

2.1 Alias Free Subband Adaptive Filter

In Fig. 2, a configuration of the alias free subband adaptive filter is shown. The figure shows the case when the number of channel M is two ($M = 2$) for simplifying the description. We used, in this paper, two time-indexes n and m where m shows the time index of down sampled signals.

In the figure, $\boldsymbol{w}_0(n)$ and $\boldsymbol{w}_1(n)$ shows the coefficient vectors of adaptive filters. Note that each channel uses the same adaptive filters, i.e., $\boldsymbol{w}_0(n)$, and $\boldsymbol{w}_1(n)$ in this case. Let us denote the filter coefficients vector $\boldsymbol{w}_i(n)$ as

$$\boldsymbol{w}_i(n) = [w_i(n,0) \; w_i(n,1) \cdots w_i(n,P-1)]^T \quad 0 \le i < M \quad (3)$$

where P shows the length of the filters and $w_i(n,m)$ shows the m-th impulse response of $\boldsymbol{w}_i(n)$. We assume that the lengths of each filter $\{\boldsymbol{w}_i(n) : 0 \le i < M\}$ are P and P is an integer multiple of M for simplifying the consideration.

We should note that $\boldsymbol{w}_i(n)(0 \le i < M)$ are the polyphase components of the adaptive filter $\boldsymbol{w}(n)$ in Fig. 1 and the relation between them are given as

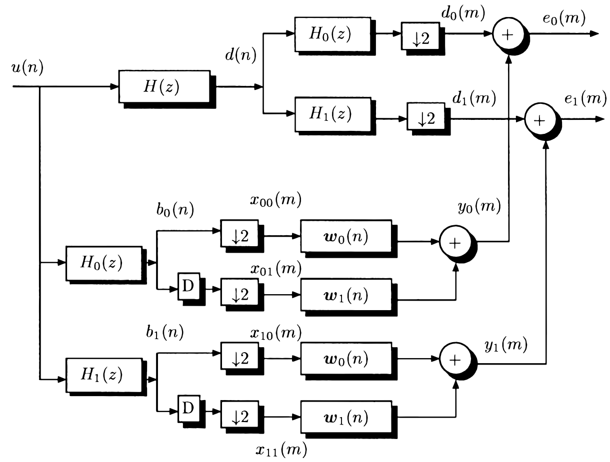

Fig. 2 Configuration of the AFSAF when the number of channel is two. D shows a unit delay.

$$w_j(n) = \begin{cases} w_0(n, j/2) & (j\text{: even}) \\ w_1(n, \lceil j/2 \rceil) & (j\text{: odd}) \end{cases} \quad (4)$$

where $w_i(n,m)$ is the m-th impulse response of $\boldsymbol{w}_i(n)$.

$H_0(z)$ and $H_1(z)$ are the transfer functions of analysis filters. The conventional method suggests [2] to use the cosine modulated paraunitary filter banks as $H_i(z)$.

Coefficients of the adaptive filters are updated using error signals $e_0(n)$, $e_1(n)$, which are expressed as

$$e_i(m) = y_i(m) - \sum_{k=0}^{M-1} \boldsymbol{x}_{ik}^T(m)\boldsymbol{w}_k(m) \quad (5)$$

where $\boldsymbol{x}_{ik}(n)$ is defined as

$$\boldsymbol{x}_{ik}(m) = [x_{ik}(m) \; x_{ik}(m-1) \cdots x_{ik}(m-P+1)]^T \quad (6)$$

and it shows the input vector of k-th polyphase components of i-th subband. The cost function of the AFSAF is defined as [2]

$$J(m) = E\left[\sum_{i=0}^{M-1} \alpha_i e_i^2(m)\right] \quad (7)$$

where $E[\cdot]$ shows the expectation operator, and α_i is a constant proportional to the inverse of the power of the input signals of $\boldsymbol{w}_i(m)$ [2].

The filters are updated based on the LMS algorithm:

$$\boldsymbol{w}_k(m+1) = \boldsymbol{w}_k(m) + \mu\left[\sum_{i=0}^{M-1} \alpha_i e_i(n)\boldsymbol{x}_{ik}(n)\right] \quad (8)$$

where μ shows the step-size parameter of the LMS algorithm, and it should be determined to ensure the convergence of $\boldsymbol{w}_i(m)$.

Readers should notice that the polyphase structure used in the AFSAF system is only for reducing its computational complexity. Also, it may not be a subband adaptive system in the sense that each subband is not independent of other subbands.

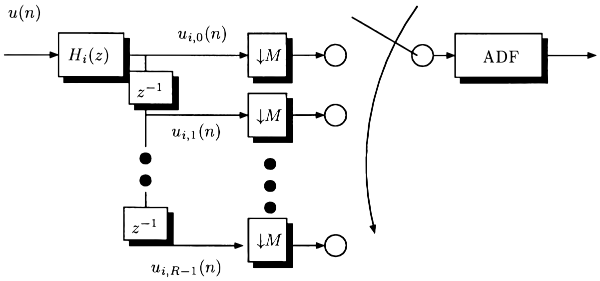

Fig. 3 Configuration of multirate repeating method (i-th channel).

2.2 Multirate Repeating Method [1]

Multirate repeating method (MRM) is a method for increasing the convergence speed of the subband adaptive filter [1]. In a subband adaptive system, the input and the desired signals are decimated before being fed into adaptive filters. The MRM uses the redundant data generated in the decimation processes. In Fig. 3, a configuration of the i-th channel of the subband system using MRM is shown.

As shown in this figure, we can obtain R different input sequences with different decimation timing where R is an integer. In the conventional subband adaptive system, only one of these sequences, say $u_{i,0}(n)$, is used for adaptation.

On the other hand, the MRM uses R sequences. Namely, the adaptive filter is updated using the first sequence $u_{i,0}(n)$, then using $u_{i,1}(n)$ and so forth. As a consequence, we can update the adaptive filter R times in a unit time compared with that of the system without the MRM. It is shown that this increment of update number results in the increase of the rate of convergence when applied to the LMS algorithm [1].

2.3 Normalized LMS Algorithm as a Special Case of APA

As mentioned in the introduction, we compare the convergence characteristics of the AFSAF with that of the affine projection algorithm (APA). Therefore, we review the APA next. As described in the following, the APA can be regarded as a generalized form of the normalized LMS (NLMS) algorithm [8]. Therefore, we first give the formula of the NLMS algorithm.

In the NLMS algorithm, the filter coefficients are updated according to the following equation:

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \boldsymbol{u}(n)e(n)\frac{\alpha}{\boldsymbol{u}^T(n)\boldsymbol{u}(n)} \tag{9}$$

where $e(n)$ is the error signal defined as

$$e(n) = d(n) - \boldsymbol{u}^T(n)\boldsymbol{w}(n). \tag{10}$$

α in (9) corresponds to the step-size parameter of the LMS algorithm and its selectable range for ensuring the convergence of the algorithm is given [9] as

$$0 < \alpha < 2. \tag{11}$$

Note that the range for α is determined independently of the input signal.

2.4 Formula of the APA

In the APA, the coefficients of an adaptive filter are updated according to the following equation [3], [5]:

$$\boldsymbol{w}(n+1) = \boldsymbol{w}(n) + \delta(n) \tag{12}$$

In this equation, $\delta(n)$ is defined as

$$\delta(n) = \boldsymbol{X}(n)\boldsymbol{g}(n) \tag{13}$$

where

$$\boldsymbol{X}(n) = [\boldsymbol{u}(n), \boldsymbol{u}(n-1), \cdots, \boldsymbol{u}(n-p+1)] \tag{14}$$

$$\boldsymbol{g}(n) = \tilde{\boldsymbol{R}}^{-1}(n)\boldsymbol{e}(n) \tag{15}$$

$$\tilde{\boldsymbol{R}}(n) = \boldsymbol{X}^T(n)\boldsymbol{X}(n) \tag{16}$$

$$\boldsymbol{e}(n) = \boldsymbol{d}(n) - \boldsymbol{X}(n)\boldsymbol{w}(n) \tag{17}$$

$$\boldsymbol{d}(n) = [d(n), d(n-1), \cdots, d(n-p+1)]^T \tag{18}$$

and p is the order of the projection. Equation (12) uses the information at time n through $n-p+1$, and therefore, we call the APA expressed by the above equations as p-th order APA algorithm, and denote it as AP(p). By comparing Eqs. (9) and (12), we can see that the normalized LMS (NLMS) algorithm can be regarded as the first order APA, or AP(1) [4], [5].

It is known that the AP(p) improves the rate of convergence provided that the input signal can be modeled as an AR(p) process [3]. For higher order AR processes, we must increase the order of AP. The computational complexity however increases as the order p increases because it requires the matrix inversion as shown in (15). Note that some fast algorithms for higher order AP have been proposed [4], [5]. However, there might be some numerical problems in those algorithms when the word-length is short. Because those fast algorithms use the similar formula to those of the fast RLS algorithms [10] that are known to be affected by the word-length used to implement the algorithm. On the other hand, as we will explain in the following section, the proposed method is based on the LMS algorithm which is known to be numerically robust [9]. Therefore, we can say that the proposed method is preferred in the view of the robustness to the word-length.

3. Proposed Method

We describe the proposed method in this section. First, we consider the selection of the analysis filters for the proposed method. Then, application of the MRM for AFSAF will be considered.

3.1 Selection of the Analysis Filters

Let us describe the proposed selection of analysis filters $H_i(z)$.

In this paper, we propose to use the elements of Hadamard matrix as the coefficients of the analysis filters $H_i(z)$. The Hadamard matrix is a square one and its elements are either 1 or -1, and its rows are orthogonal to each other. These characteristics of the Hadamard matrix enables us to implement the proposed method with small increase in computational cost.

For example, we show the Hadamard matrix of size 2×2:

$$\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix} \tag{19}$$

We can easily show that each row are mutually orthogonal. From this matrix, we can constitute analysis filters H_i as [11]

$$H_0(z) = 1 + z^{-1} \tag{20}$$
$$H_1(z) = 1 - z^{-1} \tag{21}$$

For the four channel case, we can obtain the transfer functions of the analysis filters using (20) and (21) as

$$H_{4,0}(z) = H_0(z)H_0(z^2) \tag{22}$$
$$H_{4,1}(z) = H_0(z)H_1(z^2) \tag{23}$$
$$H_{4,2}(z) = H_1(z)H_0(z^2) \tag{24}$$
$$H_{4,3}(z) = H_1(z)H_1(z^2) \tag{25}$$

where $H_{4,i}(z)$ shows the analysis filters of four channel cases. By repeating this procedure, we can easily increase the number of subbands.

3.2 Equivalent Expression of AFSAF

Next, let us consider applying the MRM to the AFSAF system. However, we cannot apply the idea of the MRM directly to the AFSAF because in the AFSAF, $b_i(n)$ is not decimated sample by sample as in the standard subband adaptive system as mentioned in the previous section. Hence, we should modify the MRM to suit for AFSAF.

For that purpose, we should express decimation operation shown in Fig. 2 in another way. First, let us express the output signal $b_i(n)$ of $H_i(z)$ defined as

$$b_i(n) = \sum_{\ell=0}^{L-1} h_{i,\ell}\, u(n-\ell) \tag{26}$$

where $h_{i,\ell}$ and L are the ℓ-th impulse response of $H_i(z)$ and its length. Using thus defined $b_i(n)$ the input vector to $\boldsymbol{w}(n)$ is given as

$$\boldsymbol{b}_i(n) = [b_i(n) \;\; b_i(n-1) \; \ldots \; b_i(n-N+1)]^T \tag{27}$$

where N is the length of $\boldsymbol{w}(n)$. In the AFSAF system, we can think the decimation process decimates the input vector $\boldsymbol{b}_i(n)$ instead of the input sample $b_i(n)$. In other word, the input vector to the adaptive filter $\boldsymbol{w}(m)$ can be expressed as the sequence:

$$\boldsymbol{b}_i(n) \;\; \boldsymbol{b}_i(n+M) \;\; \boldsymbol{b}_i(n+2M) \;\; \cdots \tag{28}$$

and this is the difference from the standard subband adaptive filters.

Thus, using (2) and (27), the output signal $y_i(m)$ of $\boldsymbol{w}(m)$ is defined as

$$y_i(m) = \boldsymbol{b}_i^T(m)\boldsymbol{w}(m) \tag{29}$$

and then the error signal in i-th channel $e_i(m)$ is given as

$$\begin{aligned} e_i(m) &= \sum_{k=0}^{L-1} h_{i,k} d(n-k) - y_i(m) \\ &= d_i(m) - y_i(m) \end{aligned} \tag{30}$$

where $d_i(m)$ is the desired signal in i-th channel.

Thus, we can express the updated formula in the conventional AFSAF using these quantities as

$$w(m+1) = w(m) + \sum_{i=0}^{M-1} \mu\alpha_i e_i(m)\boldsymbol{b}_i(m). \tag{31}$$

or we can translate this formula in terms of n, i.e., time-index of the higher rate, as

$$w(n+M) = w(n) + \sum_{i=0}^{M-1} \mu\alpha_i e_i(n)\boldsymbol{b}_i(n). \tag{32}$$

Note that $\boldsymbol{w}(n)$ is updated once per M time.

We can express the alternative representation of AFSAF shown above as in Fig. 4. Instead of sample-by-sample decimation in the standard subband adaptive system, we can express decimation operations in AFSAF as the decimation of the input vectors $\boldsymbol{b}_i(n)$. Note that, in this figure, the sum of the error signals from other channels is expressed as $\tilde{e}_i(m)$.

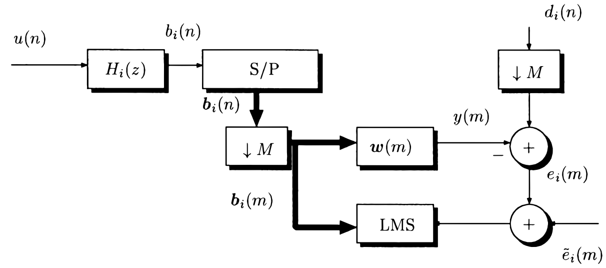

Fig. 4 An equivalent structure of the i-th channel of AFSAF. S/P shows a serial to parallel converter. A bold line shows that the signal is a vector and a thin line shows a scalar. $\tilde{e}_i(m)$ shows the sum of the error signals from other channels

3.3 MRM for AFSAF

Then, based on the expression of (32), we explain the proposed procedure. For that purpose, we first define a new time index τ as

$$\tau = \text{rem}\left[\frac{nR}{M}\right] \tag{33}$$

where rem$[x]$ shows the remainder of x. Note that τ changes cyclically. We express the number of repetition by R where R is an integer and should be a divisor of M.

When $\tau = 0$, the adaptive filters are updated using the formula given as (32). On the other hand, when $\tau \neq 0$, the filter coefficients are not modified, and they are used only for producing the output signal. When we use the formula (32) we should determine the value of the constants α_i. The reference [2] suggests a method for determining these value using the powers of the input signals $\boldsymbol{x}_{ik}(m)$. However, we cannot use the method when those powers are unknown.

To avoid this situation, we propose to use the formula proposed in [12] that is express as

$$w\left(n+\frac{M}{R}\right) = w(n) + \frac{1}{\sqrt{M}}\sum_{i=0}^{M-1}\mu_i e_i(n)\boldsymbol{u}_i(n). \tag{34}$$

In this equation, μ_i is

$$\mu_i = \frac{\alpha}{\boldsymbol{u}_i^T(n)\boldsymbol{u}_i(n)} \tag{35}$$

where α corresponds to the step-size parameter in the NLMS algorithm and its suitable range is given as $0 < \mu \leq 1$.

Note that when M is greater than N $(M > N)$ then we should slightly modify the update equation (34) [12]. The one possible selection is given in [12] and the selection modifies the second term of the right hand side of (34) as

$$\frac{1}{\sqrt{\phi M}}\sum_{i=0}^{M-1}\mu_i e_i(n)\boldsymbol{u}_i(n). \tag{36}$$

where the parameter ϕ is given as

$$\phi = M/N \tag{37}$$

As a result, the adaptive filters are updated once per M/R time. In Fig. 5, an equivalent configuration of i-th channel of the proposed method is shown. As shown in the figure, the proposed method can be implemented using decimation operations with the decimation ratio M/R.

Note that when $R = 1$ the proposed procedure is identical to the conventional AFSAF. On the other hand, when $R = M$, it is same as the system proposed in [13].

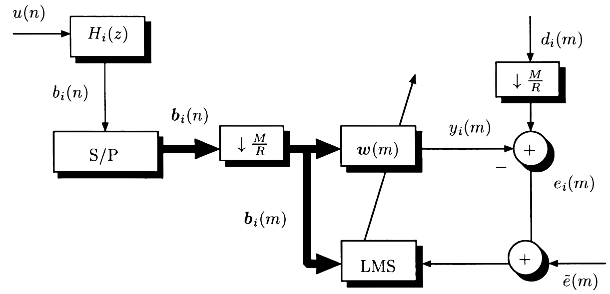

Fig. 5 Proposed configuration for i-th channel.

4. Simulation and Consideration

Here, we give some simulation results using the proposed structure and consider its computation complexity.

4.1 Simulation

We simulated the system identification problem under the following conditions:

- We used an AR(4) process as the input signal $u(n)$. The following equation [5] was used to generate it:

$$\begin{aligned} u(n) &= \gamma(n) + 0.95u(n-1) + 0.19u(n-2) \\ &\quad +0.09u(n-3) - 0.5u(n-4) \end{aligned} \tag{38}$$

  where $\gamma(n)$ is a white Gaussian process with zero mean and variance one.
- As the unknown system we used an FIR filter of length N_o that was designed using the Remez exchange algorithm. The length of adaptive filter was selected as $N = N_o$, and N_o was 61, or 151.
- We added white Gaussian noises of different variances to the desired signal $d(n)$ so that the signal-to-noise ratio (SNR) was -50 dB, or -20 dB.
- We compared the proposed structure with the affine projection algorithm [3], [6] of fifth order, or APA(5). This selection was done according to the form of input signal (38). For the proposed method, we varied the number of channel M $(M = 32,\ 128)$ and the number of repetition R $(R = 1, 2, 4)$
- The coefficients of analysis filters are selected from the Hadamard matrix [11].

We compared the NLMS, the APA, and the proposed method in terms of the impulse response error ratio (IRER). The definition of the IRER at time n is given as

$$\text{IRER}(n) = \frac{[\boldsymbol{w}_{opt} - \boldsymbol{w}(n)]^T[\boldsymbol{w}_{opt} - \boldsymbol{w}(n)]}{\boldsymbol{w}_{opt}^T\boldsymbol{w}_{opt}} \tag{39}$$

where we used $\boldsymbol{w}_{opt}$ to indicate the impulse response of the unknown system.

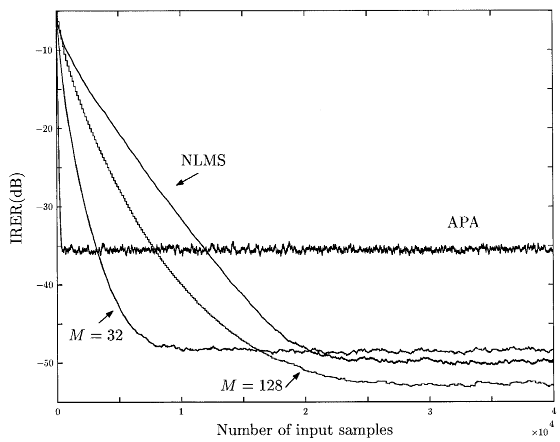

Fig. 6 Results of simulation when the signal-to-noise ratio was −50 dB.

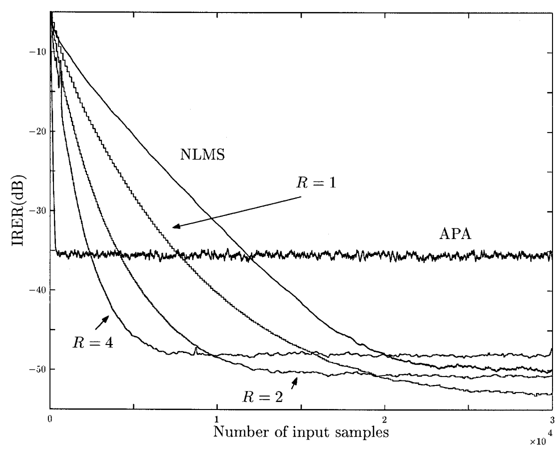

Fig. 7 Results of simulation when the signal-to-noise ratio was −50 dB.

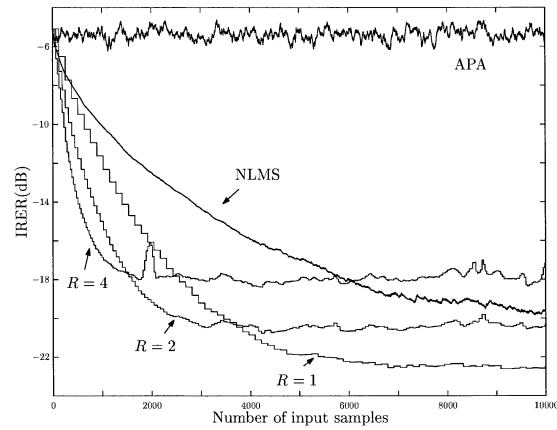

Fig. 8 Results of simulation when the signal-to-noise ratio was −20 dB. The length of adaptive filter and the unknown system were 61.

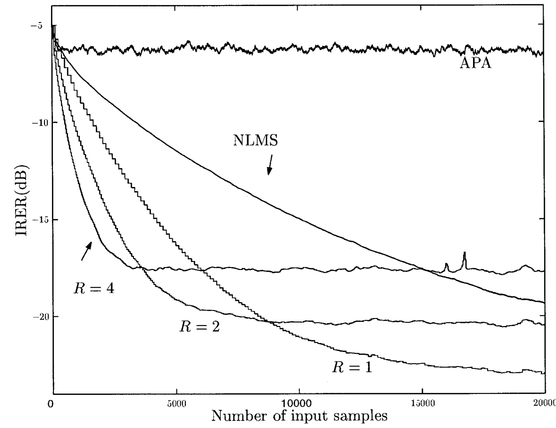

Fig. 9 Results of simulation when the signal-to-noise ratio was −20 dB. The length of adaptive filter and the unknown system were 151.

4.1.1 Case of SNR is −50 dB

As mentioned in the consideration above, the proposed method has two parameters, namely M, the number of channels, and R, the number of repetition.

We firstly show the effect of the selection of M. In Fig. 6, the results of simulations when the SNR was −50 dB are shown. We set in this simulation, the number of channels of proposed method as $M = 32$, and 128.

From the figure, it can be seen that by varying M, the rate of convergence, and the final error are adjustable. In other words, we can trade the rate of convergence and the final error level by varying the number of channel M. Note that, although the APA increases the rate of convergence greatly, the IRER was largely affected by additive noise.

Next, we consider the effect of the number of repetition R by fixing M as $M = 128$. In Fig. 7, we show the results of simulations when we varying the number of repetition as $R = 1$, 2, and 4.

From the results, we can see that the rate of convergence of the proposed method can be increased by increasing R. Moreover, the final IRER is slightly better than that of the NLMS algorithm for the case of $R = 1$, or 2.

4.1.2 Case of SNR is −20 dB

Figure 8 shows the results of simulation when SNR was −20 dB. In this case, we can say that the APA is the most affected algorithm among the three in terms of the IRER criterion. On the other hand, the proposed method can still provide better IRER performance than the NLMS algorithm when $R = 1$, or 2.

To show the dependency of the proposed method on the filter length we provide the simulation results using a different filter length, namely N was selected as 151. Other conditions for the simulation were same as those of Fig. 8. The results are shown in Fig. 9. From the figure, we can see that the characteristics of IRER are similar to those of Fig. 8 except for longer time for convergence due to the change of filter length.

4.2 Consideration on Computational Complexity

Finally, let us consider the issues regarding implementation of the proposed method. We consider in two aspects, namely, the number of computation and the amount of required hardware.

4.2.1 Computational Complexity

First, let us consider the computational complexity of the proposed method. Note that, here, we evaluate complexity in term of the amount of multiplications and divisions.

Let us begin evaluating analysis process. By selecting the Hadamard matrix as the coefficients of analysis filters, the input signal $u(n)$ can be decomposed into subband signals $b_i(n)$ without multiplications because the Hadamard matrix consists of only ones and minus ones.

Next, we see the complexity of adaptive process. When $\tau = 0$, the adaptive filter will be updated according to (34). In each channel, the second term of the right hand side of (34) is calculated. To calculate each term, the same amount of calculation as that of the NLMS algorithm is required, namely, $2N$ multiplication and one division. Hence, the following multiplications and divisions are required for updating $\boldsymbol{w}(n)$:

$$\begin{array}{ll} \text{Mul:} & M \times (2N) \\ \text{Div:} & M \end{array} \tag{40}$$

We are also required an additional multiplication and a square root calculation in (34). However, we can calculate those values when N and M are determined, so that we can treat them as a multiplication of a constant value.

We should note that the computational complexity thus evaluated are not required at each time, but M/R times for every M times. Therefore, it is natural to average (40) over M.

$$\begin{array}{ll} \text{Mul:} & \frac{R}{M} \times M(2N+1) = R(2N+1) \\ \text{Div:} & \frac{R}{M} \times M = R \end{array} \tag{41}$$

It is obvious that the required amount of multiplications and divisions are equal to those of the NLMS algorithm when $R = 1$. From Fig. 8, we have seen that, when the SNR is low, $R = 1$ provides better performance. Hence, we can improve the performance of the adaptive filter with slight increase of the computation complexity compared to the standard NLMS algorithm.

4.2.2 Hardware Requirement

We can implement the proposed method either using software on a DSP processor, or in dedicated hardware, or mixture of them.

At first, let us estimate the required amount of hardware under the condition that the proposed method is implemented in dedicated hardware. Then, in view of the required amount of hardware, those required for the proposed method can be roughly estimated as M times large than those required by the NLMS algorithm, because we should prepare M sets of the configuration shown in Fig. 5 to implement the proposed method and each one requires roughly the same amount of hardware as the NLMS algorithm.

On the other hand, when we implement using software, we need M times larger amount of memories to store the intermediate results of computations.

Hence, in the view of the amount of the hardware requirement, the requirement of the proposed method will be greater than those of the NLMS algorithm regardless of the implementation method.

5. Conclusion

In this paper, we proposed a method for applying the multirate repeating method to the alias free subband adaptive filter. The proposed method has two parameters R and M. By varying those parameters we can either increase the rate of convergence, or lower the final error in low SNR environments. The effectiveness of the proposed method was shown through the computer simulations. Besides, we evaluated the computational and hardware requirements of the proposed method. Although the increase in computational requirement can be saved, its requirement in hardware become large as the number of subband increases.

Acknowledgement

This work was supported in part by the Japan Society for the Promotion of Science under Grant-in-Aid for Scientific Research 11650389 and by Texas Instruments Japan.

References

[1] H. Kiya, K. Nishikawa, and K. Ashihara, "Improvement of convergence speed for subband adaptive digital filter using the multirate repeating method," IEICE Trans. Fundamentals (Japanese Edition), vol.J78-A, no.2, pp.194–201, Feb. 1995.

[2] S.S. Pradhan and V.U. Reddy, "A new approach to subband adaptive filtering," IEEE Trans. Signal Processing, vol.47, pp.655–664, March 1999.

[3] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," Electron. Commun. Jpn., vol.67-A, pp.19–27, 1984.

[4] S.L. Gay, "The fast affine projection algorithm," Proc. IEEE ICASSP'95, pp.3023–3026, Detroit, 1995.

[5] M. Tanaka, Y. Kameda, S. Makino, and J. Kojima, "Fast

projection algorithm and its step size control," Proc. IEEE ICASSP'95, pp.945–948, Detroit, 1995.

[6] M. Tanaka, S. Makino, and J. Kojima, "A block exact fast affine projection algorithm," IEEE Trans. Speech & Audio Proc., vol.7, pp.79–86, Jan. 1999.

[7] I. Yamada, K. Yamada, and H. Kiya, "An efficient robust adaptive algorithm based on parallel subgradient projection techniques," IEICE Technical Report, DSP 2000-138, Dec. 2000.

[8] J. Nagumo and A. Noda, "A learning method for system identification," IEEE Trans. Autom. Control, vol.AC-12, pp.282–287, June 1967.

[9] S. Haykin, Adaptive Filter Theory, 2nd edition, Prentice Hall, Englewood Cliffs, NJ, 1991.

[10] D.T.M. Slock and T. Kailath, "Fast transversal RLS algorithms," Adaptive System Identification and Signal Processing Algorithms, ed. N. Kalouptsidis and S. Theodoridis, chapter 5, pp.123–190, Prentice Hall, Englewood Cliffs, NJ, 1993.

[11] K. Nishikawa and H. Kiya, "Conditions for convergence of a delayless subband adaptive filter and its efficient implementation," IEEE Trans. Signal Processing, vol.46, pp.1158–1167, April 1998.

[12] K. Nishikawa and H. Kiya, "Normalized lms algorithm for alias free subband adaptive filters," Proc. ITC-CSCC 2001, July 2001.

[13] K. Nishikawa and H. Kiya, "New structure of affine projection algorithm based on subband adaptive filters," IEICE Technical Report, DSP 2000-140, Dec. 2000.

Kiyoshi Nishikawa received the B.E., the M.E., and the D.E. degrees in electrical engineering from Tokyo Metropolitan University in 1990, 1992 and 1996 respectively. From 1992 to 1993, he was at the Computer Systems Laboratory, Nippon Steel Corp. as a researcher. Since 1993, he has been with Tokyo Metropolitan University as an Assistant Professor and he is currently a lecturer of the Department of Electrical Engineering. His research interest includes the adaptive signal processing, hardware architecture of digital signal processing, and digital communications. He is a member of IEEE Signal Processing, Communications, and Computer Societies.

Hitoshi Kiya was born in Yamagata, Japan, on November 16, 1957. He received the B.E. and M.E. degrees in electrical engineering from Nagaoka University of Technology, Niigata, Japan, and the D.E. degree in electrical engineering from Tokyo Metropolitan University, Tokyo, Japan, in 1980, 1982, and 1987, respectively. In 1982, he joined Tokyo Metropolitan University, where he is currently a Professor of Electrical Engineering, Graduate School of Engineering. He was a visiting researcher of the University of Sydney in Australia from Oct. 1995 to Mar. 1996. His research interests are in digital signal processing, multirate systems, adaptive filtering, image processing, and efficient algorithms for VLSI implementation. He is an Associate Editor of Trans. of IEICE and served as an Associate Editor of Trans. on Signal Processing of IEEE from 1998 to 2000. Dr. Kiya is a Senior Member of the IEEE, and a member of the Image Electronics Engineers of Japan and the Institute of Television Engineers of Japan.