PAPER Special Section on Spread Spectrum Techniques and Applications

Prefiltering for LMS Based Adaptive Receivers in DS/CDMA Communications

Teruyuki MIYAJIMA†, *Member and* Kazuo YAMANAKA†, *Nonmember*

SUMMARY In this paper, three issues concerning the linear adaptive receiver using the LMS algorithm for single-user demodulation in direct-sequence/code-division multiple-access (DS/CDMA) systems are considered. First, the convergence rate of the LMS algorithm in DS/CDMA environment is considered theoretically. Both upper and lower bounds of the eigenvalue spread of the autocorrelation matrix of receiver input signals are derived. It is cleared from the results that the convergence rate of the LMS algorithm becomes slow when the signal power of interferer is large. Second, fast converging technique using a prefilter is considered. The LMS based adaptive receiver using an adaptive prefilter adjusted by a Hebbian learning algorithm to decorrelate the input signals is proposed. Computer simulation results show that the proposed receiver provides faster convergence than the LMS based receiver. Third, the complexity reduction of the proposed receiver by prefiltering is considered. As for the reduced complexity receiver, it is shown that the performance degradation is little as compared with the full complexity receiver.

key words: *multiple-access interference suppression, fast convergence, complexity reduction, Hebbian learning algorithm, KL transform*

1. Introduction

In direct-sequence/code-division multiple-access (DS/CDMA) communication systems, the performance degradation of the receiver for single-user demodulation is caused by the multiple-access interference (MAI) due to the cross-correlation between the spreading sequences of simultaneously accessing users. There have been many low-complexity receivers to suppress the MAI [1]. Recently, adaptive receivers with a linear adaptive filter have been investigated actively [2]–[8]. It is expected that these can be used for time-variant and/or unknown channels. One of these receivers is designed based on the minimum mean square error (MMSE) criterion and can be realized by the least mean square (LMS) algorithm [7]. In this paper, three issues concerning with the receiver using the LMS algorithm are considered: 1) analysis of slow convergence of the LMS algorithm; 2) fast convergence by prefiltering; and 3) complexity reduction by prefiltering.

The LMS algorithm has an advantage for its simplicity. Recent experimental results have shown that it takes a long time for the training convergence in DS/CDMA environment [7], [8]. However, it has not been cleared that what conditions lead to the slow convergence. It is well known that the convergence rate of the LMS algorithm can be evaluated by the eigenvalue spread of the autocorrelation matrix of receiver input signals [9]. In this paper, first, we try to derive the eigenvalue spread theoretically and consider the conditions which lead to the slow convergence.

In this paper, second, we consider the use of a prefilter for rapid convergence in the LMS based receiver. The prefilter works as a preprocessor placed before the conventional LMS based linear adaptive filter. So far, in the field of the transform-domain adaptive filter technique, fixed (non-adaptive) prefilters have been used to provide rapid convergence by decorrelating the input signals [9], [10]. Because the cause of the slow convergence of the LMS algorithm is nonzero cross-correlation between input signals. However, since prefilters used previously, e.g. discrete cosine transformation (DCT), cannot decorrelate the input signals completely, the resulting convergence rate is not always acceptable. The Karhunen-Loeve transform (KLT) is known as an ideal transform for the complete decorrelation. However, the KLT is a signal-dependent transform and requires the autocorrelation matrix, the diagonalization of this matrix and the construction of the basis vector. These computations make the KLT impractical for real-time applications.

On the contrary, it is known in the neural networks literature that a Hebbian learning algorithm (HLA) can provide the KLT by online signal processing [11]–[13]. In this paper, we propose to use an adaptive prefilter adjusted by the HLA. Rapid convergence is expected since the input signals are decorrelated by the prefilter which converges to the KLT by using the HLA. A disadvantage of receivers using a prefilter including the proposed receiver is the increase of complexity for prefiltering. As described later, the complexity of the proposed receiver can be easily reduced with small performance degradation. On the other hand, although Lee [8] has proposed to use an adaptive prefilter adjusted by the Gram-Schmidt (GS) orthogonalization for the decorrelation, its complexity cannot be reduced with small performance degradation.

It is generally known that the recursive least squares (RLS) algorithm provides faster convergence than the

Manuscript received April 11, 1997.
Manuscript revised July 9, 1997.
†The authors are with the Faculty of Engineering, Ibaraki University, Hitachi-shi, 316 Japan.

LMS algorithm [4] if high precision arithmetic is available. The RLS algorithm, however, has the numerical instability problem due to finite-precision effects [9]. The problem is caused by the computation of the difference between two nonnegative definite matrices. Lee [8] pointed out that the RLS algorithm cannot be employed in DS/CDMA environment because of the numerical problem. This paper also presents examples where the RLS algorithm fails to converge.

The employment of the HLA based prefilter increases the complexity of the receiver. In this paper, third, we consider the complexity reduction by reducing the input signal dimension using a prefilter. The input signal dimension, i.e., the number of the taps, is reduced by truncating the KLT obtained by the HLA. Usually, the truncated KLT is used for the feature extraction or the data compression [14]. Thus, the performance degradation due to the reduction is expected to be little since the truncated KLT can extract only the important components. On the other hand, although Madhow et al. [5] have proposed a non-adaptive prefilter for the reduction, its performance degrades since the prefilter does not reflect the statistics of the input signal.

In the next section, the communication model is described. Section 3 presents a linear adaptive receiver using the LMS algorithm. The eigenvalue spread is considered theoretically in Sect. 4. In Sect. 5, the receiver using the HLA based prefilter is proposed, and the complexity reduction of the receiver is discussed in Sect. 6. Section 7 presents numerical examples to demonstrate the performance of the proposed receiver. Finally, a summary is presented in Sect. 8.

2. Communication Model

Consider an asynchronous DS/CDMA communication system. Through this paper, a chip-asynchronous system is assumed. It is assumed that the channel is time-invariant and there is only a direct wave, but no delayed waves. For simplicity, a baseband model is considered.

The received signal of a BPSK-DS/CDMA system with K users can be written in the form

$$r(t) = \sum_{p} \sum_{k=1}^{K} A_k b_k(p) s_k(t - pT_b - \tau_k) + n(t) \quad (1)$$

where A_k is the signal amplitude of the kth user, $b_k(p) \in \{+1, -1\}$ is the pth data bit, $s_k(t)$ is the signature waveform, τ_k is the timing, T_b is the bit duration and $n(t)$ is zero-mean white Gaussian noise with PSD $N_0/2$. The bits $\{b_k(p)\}$ are assumed to be uncorrelated for all k and p. For ease of presentation, the timings τ_k are assumed to satisfy $0 \leqq \tau_1 < \tau_2 < \cdots < \tau_K < T_b$. The signature waveform $s_k(t)$ can be expressed as

$$s_k(t) = \sum_{l=1}^{L} a_{kl} P_{T_c}(t - (l-1)T_c) \quad (2)$$

where $\{a_{kl}\}$ is the spreading sequence, L is the length of the sequence, T_c is the chip duration and $P_{T_c}(t)$ is the rectangular chip waveform: $P_{T_c}(t) = 1$ for $0 \leqq t < T_c$; and 0 for otherwise.

3. Linear Adaptive Receivers

Taking the first user to be the desired user, the objective is to demodulate its ith data bit. It is assumed that both the spreading sequence and the timing of the desired user, i.e., $\{a_{1l}\}$ and τ_1, are known and the training signals are available at the receiver. Figure 1 shows the receiver structure. In the figure, the matrix $\mathbf{T}$ represents the prefilter which will be discussed in Sects. 5 and 6 in detail. If the matrix $\mathbf{T}$ is fixed, we call it the fixed prefilter. If the matrix $\mathbf{T}$ is constructed adaptively, we call it the adaptive prefilter. In this section, the conventional LMS based receiver where $\mathbf{T} = \mathbf{I}(L \times L)$ is described.

First, the received signal is fed into the chip-matched filter and its output is sampled at the chip-rate. The output for the lth chip of the ith bit is

$$r_l(i) = \frac{1}{T_c} \int_{iT_b+\tau_1+(l-1)T_c}^{iT_b+\tau_1+lT_c} r(t)dt, \quad l = 1, \ldots, L. \quad (3)$$

The serial-to-parallel converter produces the chip-matched filter outputs for the ith bit $\mathbf{r}(i) = (r_1(i), r_2(i), \ldots, r_L(i))^T$. The superscript T represents the transpose of vector or matrix. Then, the received signal vector can be expressed as

$$\mathbf{r}(i) = A_1 b_1(i)\mathbf{a}_1 + \sum_{k=2}^{K} A_k \{b_k(i-1)\check{\mathbf{c}}_k + b_k(i)\hat{\mathbf{c}}_k\} + \mathbf{n}(i) \quad (4)$$

where $\mathbf{a}_k = (a_{k1}, a_{k2}, \ldots, a_{kL})^T$ is the spreading sequence vector and $\mathbf{n}(i) = (n_1(i), n_2(i), \ldots, n_L(i))^T$ is the noise vector. Defining $\tau_k - \tau_1 = \nu_k T_c + \delta_k$, we get

$$\check{\mathbf{c}}_k = \frac{\delta_k}{T_c} \tilde{\mathbf{a}}_k^{(\nu_k+1)} + \left(1 - \frac{\delta_k}{T_c}\right) \tilde{\mathbf{a}}_k^{(\nu_k)}, \quad (5)$$

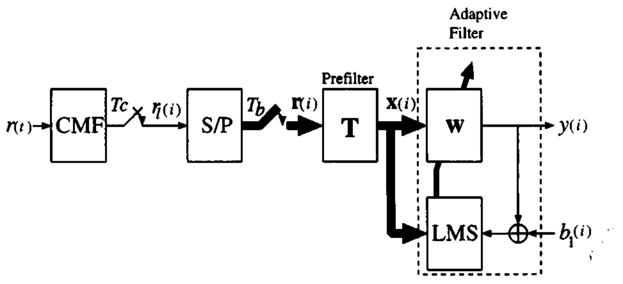

Fig. 1 Receiver structure.

$$\hat{\mathbf{c}}_k = \frac{\delta_k}{T_c}\hat{\mathbf{a}}_k^{(\nu_k+1)} + \left(1 - \frac{\delta_k}{T_c}\right)\hat{\mathbf{a}}_k^{(\nu_k)} \tag{6}$$

where

$$\tilde{\mathbf{a}}_k^{(\nu)} = (a_{kL-\nu+1}, a_{kL-\nu+2}, \ldots, a_{kL}, 0, \ldots, 0)^T, \tag{7}$$

$$\hat{\mathbf{a}}_k^{(\nu)} = (0, \ldots, 0, a_{k1}, a_{k2}, \ldots, a_{kL-\nu})^T. \tag{8}$$

Moreover, (4) can be rewritten as a synchronous CDMA system where virtual $2K-1$ users exist as follows:

$$\mathbf{r}(i) = \sum_{j=1}^{2K-1} A'_j b'_j(i)\mathbf{c}_j + \mathbf{n}(i) \tag{9}$$

where defining $\iota(j) = \lfloor \frac{j}{2} + 1 \rfloor$, $\lfloor x \rfloor$ represents the maximum integer which does not exceed x, we get $A'_j = A_{\iota(j)}$ for all j; $b'_j(i) = b_{\iota(j)}(i-1)$, $\mathbf{c}_j = \tilde{\mathbf{c}}_{\iota(j)}$ for $j \bmod 2 = 0$; and $b'_j(i) = b_{\iota(j)}(i)$, $\mathbf{c}_j = \hat{\mathbf{c}}_{\iota(j)}$ for $j \bmod 2 = 1$.

The output of the linear adaptive filter is given by

$$y(i) = \mathbf{w}^T\mathbf{r}(i) \tag{10}$$

where $\mathbf{w}$ is the $L \times 1$ weight vector. The estimation of the data bit can be obtained from the adaptive filter output as

$$\hat{b}_1(i) = \mathrm{sgn}(y(i)). \tag{11}$$

The mean square error (MSE) is defined as

$$J_0(\mathbf{w}) = E[(b_1(i) - \mathbf{w}^T\mathbf{r}(i))^2] \tag{12}$$

where $E[\cdot]$ represents the expectation operator. The minimum MSE (MMSE) receiver has the weight vector which minimizes the MSE [5]. Then, the optimum weight vector is given by

$$\mathbf{w}_{lms} = A_1\mathbf{R}^{-1}\mathbf{a}_1 \tag{13}$$

where $\mathbf{R} = E[\mathbf{r}(i)\mathbf{r}^T(i)]$ is the $L \times L$ autocorrelation matrix of the input signals.

The optimum weight vector in (13) can be obtained by a simple iterative procedure, i.e., the LMS algorithm:

$$\mathbf{w}(i+1) = \mathbf{w}(i) + \mu(b_1(i) - \mathbf{w}^T(i)\mathbf{r}(i))\mathbf{r}(i) \tag{14}$$

where μ is the step size which is chosen such that $0 < \mu < 2/\lambda_1$ where λ_1 is the maximum eigenvalue of $\mathbf{R}$. In the sequel, the eigenvalues are arranged in decreasing order, i.e., $\lambda_1 \geqq \lambda_2 \geqq \cdots \geqq \lambda_L$. As for the autocorrelation matrix $\mathbf{R}$, the eigenvalue spread is defined as [9]

$$\chi(\mathbf{R}) = \frac{\lambda_1}{\lambda_L}. \tag{15}$$

It is clear that $\chi(\mathbf{R}) \geqq 1$. If there exists strong correlation between the elements of the received signal vector $\mathbf{r}(i)$, the eigenvalue spread becomes large and the convergence rate of the LMS algorithm becomes slow [9]. In DS/CDMA systems, the experimental results which show the slow convergence of the LMS algorithm have been reported [7], [8]. However, the conditions which lead to the slow convergence have not been cleared theoretically.

4. Evaluation of Eigenvalue Spread

Let us consider the eigenvalue spread theoretically. Since it is difficult to obtain the eigenvalue spread directly, both upper and lower bounds are derived. Since the bits $\{b'_j(i)\}$ and the noise components $\{n_l(i)\}$ in the received signal vector $\mathbf{r}(i)$ are uncorrelated, the autocorrelation matrix can be written as

$$\mathbf{R} = \sum_{j=1}^{2K-1} A'^2_j \mathbf{R}_j + \sigma^2\mathbf{I} \tag{16}$$

where $\mathbf{R}_j = \mathbf{c}_j\mathbf{c}_j^T$ is the autocorrelation matrix of the jth user's sequence and $\sigma^2 = N_0/2T_c$ is the noise variance. It is clear that all L eigenvalues of the second term in (16) are σ^2. On the other hand, since $\mathbf{R}_j$ is a positive semi-definite matrix and its rank is unity, its smallest $L-1$ eigenvalues are zero [15]. Since the eigenvector corresponding to the remaining eigenvalue, i.e., the largest eigenvalue, is $\mathbf{c}_j$, the eigenvalue is

$$\lambda_{j1} = \sum_{l=1}^{L} c_{jl}^2. \tag{17}$$

By using formulas concerning the eigenvalues of the sum of Hermitian matrices [16], [17], we can obtain the following results about the maximum and the minimum eigenvalues:

$$\max_j E'_j + \sigma^2 \leqq \lambda_1 \leqq \sum_{j=1}^{2K-1} E'_j + \sigma^2 \tag{18}$$

and

$$\sigma^2 \leqq \lambda_L \leqq \sum_{j=1}^{2K-1} E'_j - \max_j E'_j + \sigma^2 \tag{19}$$

where $E'_j = A'^2_j\lambda_{j1} = A'^2_j\sum_{l=1}^{L} c_{jl}^2$ which represents the signal energy of the virtual jth user when $T_c = 1$. From (18) and (19), we can obtain both upper and lower bounds of the eigenvalue spread as follows:

$$\max\left\{1, \frac{\max_j E'_j + \sigma^2}{\sum_{j=1}^{2K-1} E'_j - \max_j E'_j + \sigma^2}\right\} \leqq \chi(\mathbf{R}) \leqq \frac{\sum_{j=1}^{2K-1} E'_j + \sigma^2}{\sigma^2}. \tag{20}$$

It is noted that the result is effective regardless of the number of users.

Furthermore, let us consider the case where the rank of the first term in (16) r is smaller than L. Although the rank r depends also on the spreading sequences and the timings, it depends strongly on the number of users. The rank r is clearly smaller than $2K-1$, and may be equal to $2K-1$ for most cases. Thus, in the considering case, L may be larger than

$2K-1$, i.e., $L > 2K-1$. Then, since the $L-r$ smallest eigenvalues of the first term in (16) are zero, the minimum eigenvalue of $\mathbf{R}$ is σ^2. Using $\lambda_L = \sigma^2$ instead of (19), the eigenvalue spread becomes

$$\frac{\max_j E'_j + \sigma^2}{\sigma^2} \leq \chi(\mathbf{R}) \leq \frac{\sum_{j=1}^{2K-1} E'_j + \sigma^2}{\sigma^2}. \quad (21)$$

The result in (21) suggests that there are two cases where the eigenvalue spread becomes large. The first is the case where the maximum signal energy becomes large, while both the other energies and the noise variance are fixed. If the user who has the largest energy is an interfering user, both the upper and the lower bounds increase as the energy of the interference becomes large. This fact agrees with the experimental results reported in [7], [8]. The second is the case where the noise variance becomes small, while all signal energies are fixed. It may also be observed from (21) that when the number of users increases the eigenvalue spread does not always increase. Because the upper bound increases but the lower bound does not always increase. In our preliminary simulation, we have confirmed that the eigenvalue spread decreases when the number of users is over a certain number.

5. Adaptive Receiver Using HLA Based Prefilter

Consider the prefiltering of the received signal vector using a $M \times L$ matrix $\mathbf{T}$, where $M \leq L$, as follows:

$$\mathbf{x}(i) = \mathbf{T}\mathbf{r}(i). \quad (22)$$

The prefilter is introduced to decorrelate the input signals for fast convergence and to reduce the input signal dimension for complexity reduction. If $M < L$, input signal dimension is reduced, i.e., M represents the reduction order. Then the filtered vector $\mathbf{x}(i)$ is fed into a linear adaptive filter whose weights are adjusted by using the LMS algorithm.

Now, (22) is rewritten by dividing the prefilter $\mathbf{T}$ into two parts as

$$\mathbf{x}(i) = \mathbf{T}\mathbf{r}(i) = \mathbf{P}^{-1/2}\mathbf{Q}^T\mathbf{r}(i) = \mathbf{P}^{-1/2}\mathbf{v}(i) \quad (23)$$

where $\mathbf{T} = \mathbf{P}^{-1/2}\mathbf{Q}^T$ is a divided prefilter, $\mathbf{v}(i) = \mathbf{Q}^T\mathbf{r}(i)$ is a transformed vector, a prefilter $\mathbf{Q}$ is a unitary matrix with rank M and $\mathbf{P} = \mathrm{diag}(p_1, \ldots, p_M) = \mathrm{diag}(E[\mathbf{v}(i)\mathbf{v}^T(i)])$ whose diagonal element represents the power of the element of $\mathbf{v}(i)$. Then, if elements of the transformed vector $\mathbf{v}(i)$ are decorrelated, the eigenvalue spread of the autocorrelation matrix of the normalized signal $\mathbf{x}(i)$ becomes unity. In the field of transform-domain adaptive filters, fixed prefilters $\mathbf{Q}$, e.g., DCT, are used for the decorrelation [10]. However, since these fixed prefilters cannot decorrelate the input signals completely, sufficient convergence rate cannot be obtained. The complete decorrelation is achievable by the KLT [9]. The KLT is given by $\bar{\mathbf{Q}} = (\bar{\mathbf{q}}_1, \bar{\mathbf{q}}_2, \ldots, \bar{\mathbf{q}}_L)$ where $\bar{\mathbf{q}}_l$ is the eigenvector corresponding to the lth eigenvalue λ_l of $\mathbf{R}$.

To obtain the eigenvectors by online signal processing, we propose to use a Hebbian learning algorithm (HLA). HLAs are unsupervised learning algorithms which have been developed in the filed of neural networks [13]. It is known that the weight vector of a neuron converges to a eigenvector by using a HLA [13]. Therefore, it can be expected that an adaptive prefilter $\mathbf{Q}$ converges to the KLT $\bar{\mathbf{Q}}$ by adjusting the prefilter $\mathbf{Q}$ using a HLA. Then, since the input signals can be decorrelated by the prefilter $\mathbf{Q}$, rapid convergence of the LMS algorithm can be expected.

In this paper, the generalized Hebbian algorithm by Sanger [11] and the learning parameter setting technique by Chen [12] are employed. The proposed algorithm is summarized by the following steps:

Step1: Initialization: As for $m = 1, \ldots, M$, $l = 1, \ldots, L$, filter tap weights $w_m(0)$ are set to zero, estimation of eigenvectors $\hat{q}_{ml}(0)$ are set to small random values, and estimation of outputs power $p_m(0)$ are set to small positive values. Set the step size μ to a small positive value. Set $i = 1$.

Step2: Prefiltering: The received signal vector is transformed by the prefilter as

$$\mathbf{v}(i) = \mathbf{Q}^T(i-1)\mathbf{r}(i). \quad (24)$$

Compute the power estimation

$$p_m(i) = p_m(i-1) + (v_m^2(i) - p_m(i-1))/i. \quad (25)$$

Then, the power estimation matrix is constructed as $\mathbf{P}(i) = \mathrm{diag}(p_1(i), \ldots, p_M(i))$ and the transformed vector is normalized as

$$\mathbf{x}(i) = \mathbf{P}^{-1/2}(i)\mathbf{v}(i). \quad (26)$$

Step3: Eigenvector estimation: Compute the time decreasing coefficient $\beta(i)$, e.g. [12]

$$\beta(i) = \max\{\beta_f, \gamma \times \epsilon^{\frac{i}{\tau}}\}, \quad (27)$$

where β_f, γ, ϵ, τ are predetermined positive constants. The eigenvectors are estimated by using a HLA.

$$\eta_m(i) = \frac{\beta(i)}{p_m(i-1)}, \quad (28)$$

$$\hat{\mathbf{q}}_m(i) = \hat{\mathbf{q}}_m(i-1) + \eta_m(i)\left(\mathbf{r}(i) - \sum_{j=1}^{m-1} v_j(i)\hat{\mathbf{q}}_j(i)\right). \quad (29)$$

Step4: Eigenvector modification: The estimated eigenvectors are modified by the modified Gram-Schmidt orthogonalization [18] to be orthonormal basis.

for $m := 1$ to M do

begin

$r_{mm} := \|\hat{\mathbf{q}}_m(i)\|;\ \mathbf{q}_m(i) := \hat{\mathbf{q}}_m(i)/r_{mm}$ (30)

for $k := m + 1$ to M do

begin

$$r_{mk} := \mathbf{q}_m^T(i)\hat{\mathbf{q}}_k(i); \quad \hat{\mathbf{q}}_k(i) := \hat{\mathbf{q}}_k(i) - \mathbf{q}_m(i)r_{mk} \tag{31}$$

end

end

Then, the prefilter is constructed as $\mathbf{Q}(i) = (\mathbf{q}_1(i), \mathbf{q}_2(i), \ldots, \mathbf{q}_M(i))$.

Step5: Adaptive filtering: Compute the filter output, and update the weight vector using the LMS algorithm.

$$y(i) = \mathbf{w}^T(i-1)\mathbf{x}(i), \tag{32}$$

$$\mathbf{w}(i) = \mathbf{w}(i-1) + \mu(b_1(i) - \mathbf{w}^T(i-1)\mathbf{x}(i))\mathbf{x}(i). \tag{33}$$

Step6: Increment i and repeat from Step2 to Step5 until the weight vector converges.

In the algorithm, two little modifications are imposed. First, although Chen et al. [12] used an estimation of an eigenvalue for the denominator in (28), we use the power estimation in (25) instead since the output power of the prefilter becomes an eigenvalue [13]. This modification eliminates the computation for eigenvalue estimation. Second, we use the modified Gram-Schmidt orthogonalization to force the transformation matrix $\mathbf{Q}(i)$ to be always a unitary matrix. In Sanger's GHA [11], the weight vectors automatically become orthonormal basis by (29) with an additional term on the right-hand side. In our preliminary simulation, the explicit orthogonalization by the Gram-Schmidt procedure provided faster convergence than the original Sanger's GHA. Thus, we employed the Gram-Schmidt procedure although the computational complexity per iteration increases, i.e., $O(LM^2)$.

The above algorithm does not have the numerical problem appeared in the RLS algorithm since there is no computation of the difference between two nonnegative definite matrices. This fact will be confirmed by computer simulation in Sect. 7.

Next, let us discuss the relation between the receiver using the prefilter and the MMSE receiver. Consider the case where the transformation $\mathbf{T} = \mathbf{P}^{-1/2}\mathbf{Q}^T$ in (22) is a $L \times L$ ($M = L$) matrix. The optimum weight obtained by the above procedure may be expressed as

$$\mathbf{w}_o = A_1\mathbf{P}^{1/2}\mathbf{Q}^T\mathbf{R}^{-1}\mathbf{a}_1. \tag{34}$$

Then, since the filter output becomes

$$y = \mathbf{w}_o^T\mathbf{x} = \mathbf{w}_{lms}^T\mathbf{r}, \tag{35}$$

the receiver is equivalent to the conventional MMSE receiver. Thus, the performance of the proposed receiver is equal to that of the conventional MMSE receiver. Note that in the case of $M < L$, it is not always equal to the performance of the MMSE receiver.

6. Complexity Reduction

The adoption of the HLA described in the previous section increases the receiver complexity. Now, consider to reduce the input signal dimension, i.e., the number of adaptive taps, by a $M \times L$ matrix $\mathbf{T}'$, instead of the $L \times L$ matrix. The matrix is obtained by omitting the lowest $L - M$ row vectors in the matrix $\mathbf{T} \in R^{L\times L}$. In other words, the lowest $L - M$ column vectors of the KLT $\bar{\mathbf{Q}} \in R^{L\times L}$ is omitted. We call it the truncated KLT.

The KLT is the optimum transform in the sense that the residual variance on the reconstruction of the original vector from its transformed vector is minimized [14]. Thus, it is expected that when the input signal dimension is reduced by the truncated KLT the performance degradation is little. In particular, if M is set to be equal to the rank of the signal component of $\mathbf{R}$, it is expected that its performance is equal to that of the full complexity case since all the signal components can be extracted.

If the transformation matrix $\mathbf{T}'(M \times L)$ is used, the optimum weight vector is written as

$$\mathbf{w}_{rc} = A_1(\mathbf{T}'\mathbf{R}\mathbf{T}'^T)^{-1}\mathbf{T}'\mathbf{a}_1 \tag{36}$$

Then, the bit error probability is given by

$$P_e = 2^{-2(K-1)} \sum_{b_2' \in \{-1,1\}} \cdots \sum_{b_{2K-1}' \in \{-1,1\}} Q\left(\frac{\mathbf{w}_{rc}^T\mathbf{T}'(A_1\mathbf{a}_1 + \sum_{j=2}^{2K-1} A_j' b_j' \mathbf{c}_j)}{\sigma'}\right) \tag{37}$$

where $\sigma' = \|\mathbf{w}_{rc}^T\mathbf{T}'\|\sigma$ and $Q(x) = (2\pi)^{-1/2} \int_x^\infty e^{-t^2/2}dt$.

For performance comparison, we consider the cyclically shifted filter bank (CSFB) proposed by Madhow et al. [5]. The CSFB is given by

$$\mathbf{T}' = (\mathbf{a}_1, \mathbf{f}_1, \ldots, \mathbf{f}_{M-1})^T \tag{38}$$

where $f_{ik} = a_{1(k+i\Delta)\bmod L}$ and $\Delta = \lfloor N/M \rfloor$. Since the CSFB does not reflect the input signal statistics, it is anticipated that its performance degrades.

7. Numerical Examples

The Gold sequences with period $L = 31$ are used as the spreading sequences. The number of users is $K = 5$. The desired user is the 1st user and its signal energy is represented by E_b. Although the signal energy in practical channels may vary with time, two typical time-invariant scenarios employed by Miller [7] and Lee [8] are considered as shown in Fig. 2. The signal energy of only the desired user differs from the all interfering users

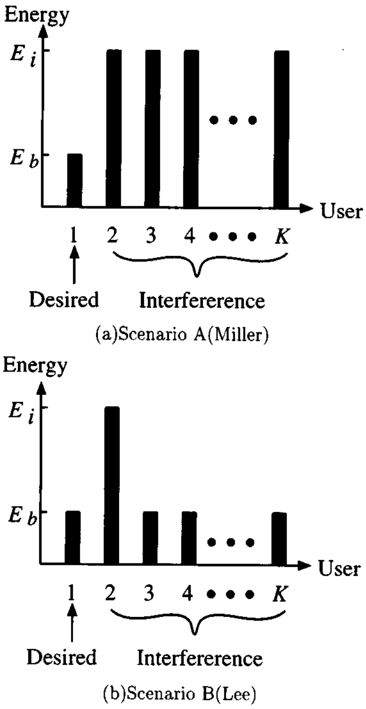

Fig. 2 Communication scenarios.

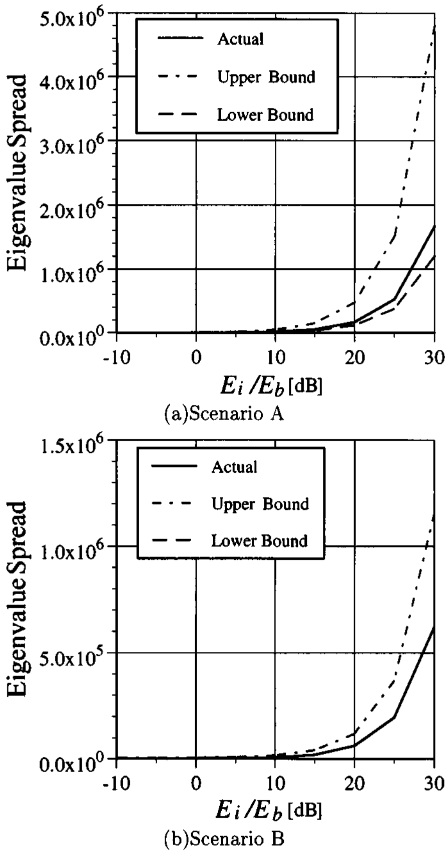

Fig. 3 Eigenvalue spread.

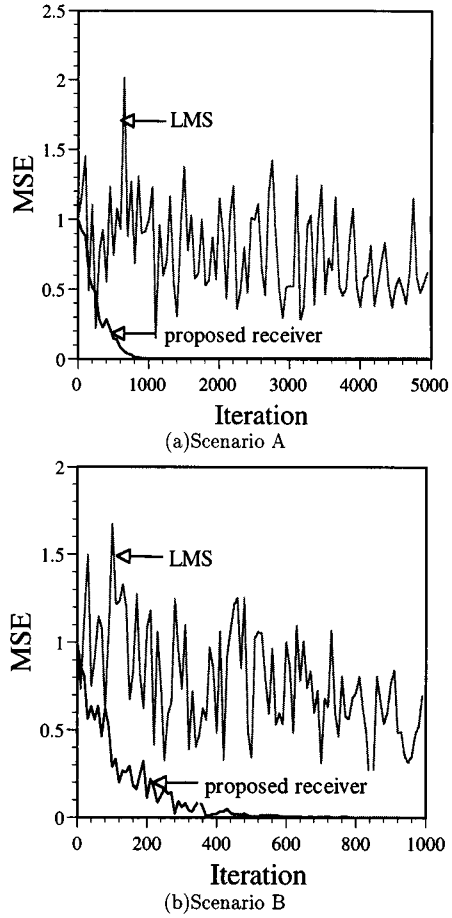

Fig. 4 Convergence rate comparison with LMS algorithm (M = 31, 52 bit mantissa).

in Miller's scenario, and the signal energy of only an interfering (2nd) user differs from other users in Lee's scenario. In both scenarios, the situations where the LMS algorithm becomes slow can be arisen by setting the interference ratio E_i/E_b large, where E_i represents the maximum interference signal's energy.

First, the bounds of the eigenvalue spread in (21) are evaluated. Figure 3 shows the relation between the eigenvalue spread versus the interference ratio E_i/E_b, where $E_b/N_0 = 30$ dB. The actual eigenvalue spreads are obtained by computing the eigenvalues of $\mathbf{R}$. It is noted that the lower bound curve overlaps with the actual eigenvalue spread curve in Fig. 3 (b). It can be found that the actual eigenvalue spread and both the upper and the lower bounds increase as the interference energy increases. This result means that when the interference energy increases the convergence rate of the LMS algorithm becomes slow since large eigenvalue spread results in slow convergence of the LMS algorithm.

Next, the convergence rate of the proposed receiver is compared with the conventional LMS based receiver via computer simulation. In Fig. 4, the learning curves are shown where $E_b/N_0 = 30$ dB, $E_i/E_b = 30$ dB, $M =$

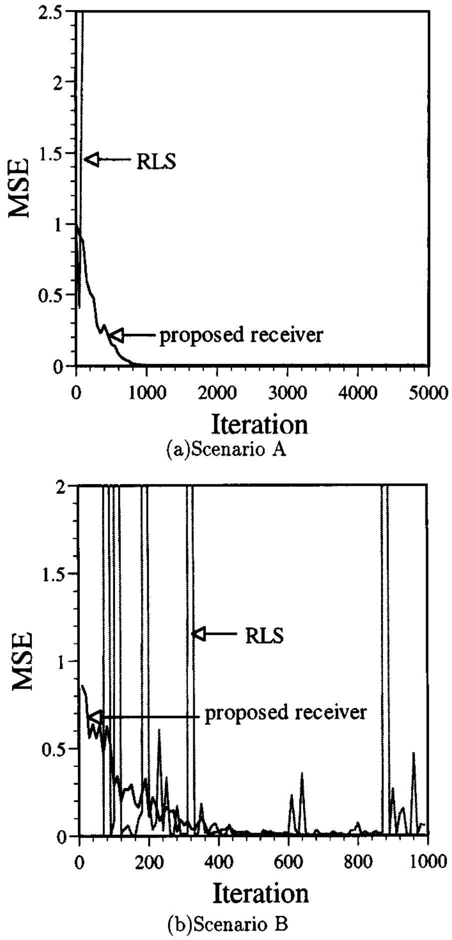

Fig. 5 Finite-precision effect in RLS algorithm and proposed receiver (M = 31, 16 bit mantissa).

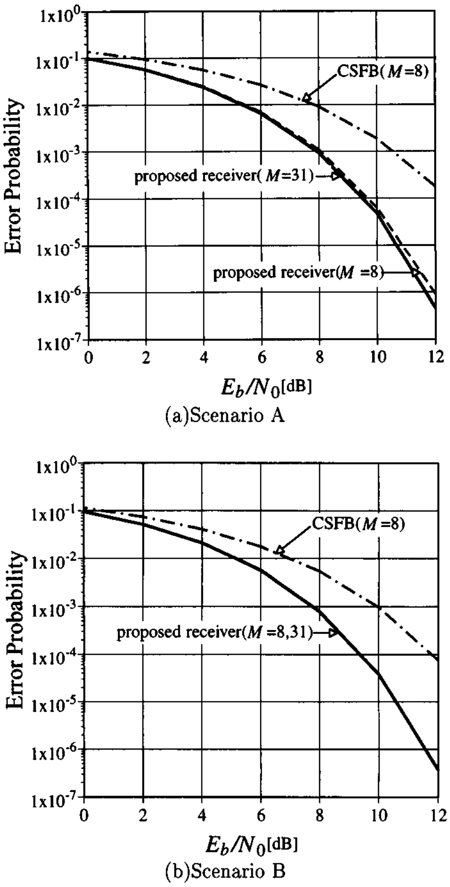

Fig. 6 Performance degradation due to complexity reduction.

31 and the mantissa of floating-point numbers is 52 bit. The learning curves were obtained by averaging the distinct 100 trials. The step size μ is set to the quarter of the reciprocal of the total input power [8]. The coefficient $\beta(i)$ in (28) is set to $\beta(i) = \max\{0.001, 0.1 \times 0.01^{i/1000}\}$ for both scenarios. In our simulation, this coefficient was determined empirically. From this figure, one can observe that the proposed receiver shows faster convergence rate than the LMS based receiver (denoted as LMS). Although the computational complexity required per an iteration of the proposed receiver is larger than the LMS based receiver, the number of training data required is small since the proposed receiver needs fewer iterations for the convergence. Although the results of the RLS algorithm based receiver are not shown in the figures, the RLS algorithm converged quickly within several tens of iterations. As shown below, however, the results of the RLS algorithm are true only if high precision arithmetic is available.

Next, the finite-precision effect is considered via computer simulation. In Fig. 5, the learning curves of the RLS algorithm based receiver (denoted as RLS) and the proposed receiver are shown where $E_b/N_0 = 30$ dB, $E_i/E_b = 30$ dB, $M = 31$ and the mantissa is 16 bit. One can observe that the RLS algorithm fails to converge. On the other hand, the results of the proposed receiver with 16 bit mantissa is almost the same as the results with 52 bit mantissa in Fig. 4. Therefore, unless high precision arithmetic is available, the RLS algorithm based receiver cannot be applied, but the proposed receiver is useful because of its robustness to the finite-precision effect.

Next, the performance degradation due to the complexity reduction is evaluated on the steady state. Since the number of users is 5, the rank corresponding to the signal component in (16) is about 9. The bit error performances of the proposed receiver with $M = 31$ (full complexity) and the receiver with $M = 8$ (reduced complexity) are shown in Fig. 6. The results of the CSFB ($M = 8$) are also shown in the figures for the purpose of comparison. The performance degradation of the proposed receiver due to complexity reduction is smaller than the CSFB method. Figure 7 shows the relation between the error probability and the reduced order M. As for the CSFB, the performance becomes better with increasing M. On the other hand, as for the proposed receiver, no performance degradation can be observed when M is larger than 9. However, when M is small, the performance degrades suddenly. Thus, it is neces-

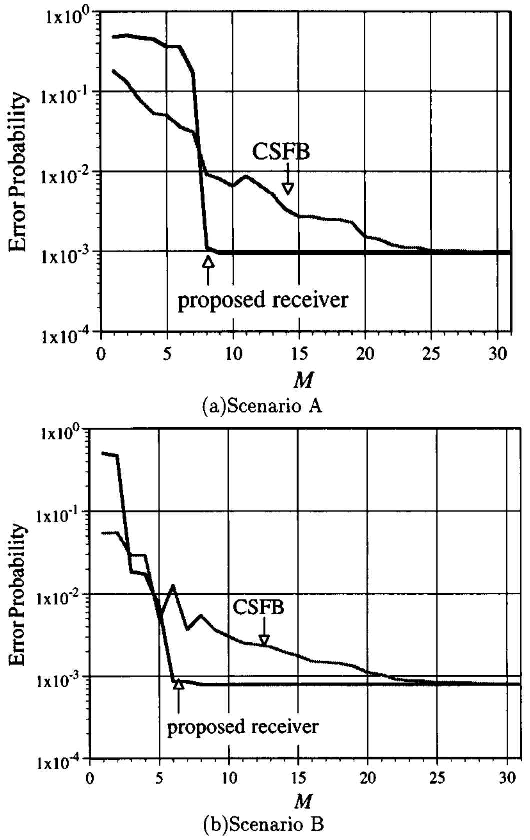

Fig. 7 Bit error probability versus reduction order M.

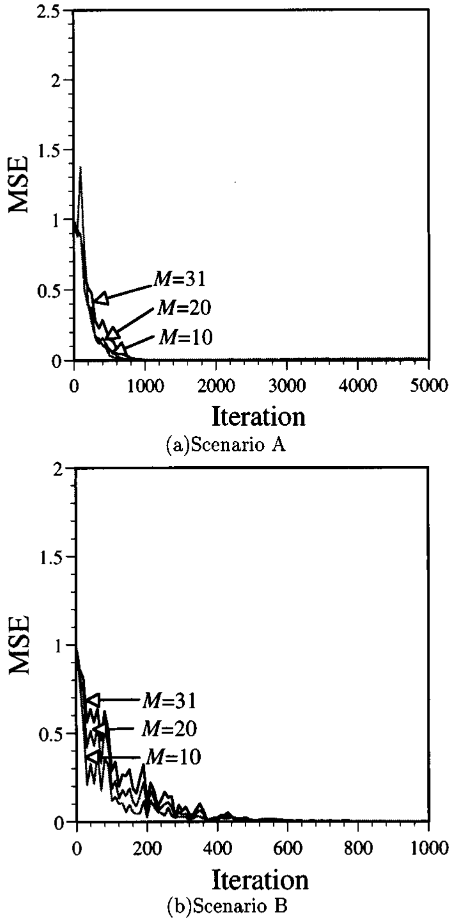

Fig. 8 Learning curves of reduced complexity receivers (52 bit mantissa).

sary to determine the reduced order M appropriately. If the number of users K is known, then set $M = 2K - 1$. If K is unknown, then M is set to the dimension of the signal subspace which is needed to estimate.

One of rank estimation methods is presented. It is known that the transformed signal power p_m in (25) converges to the eigenvalue λ_m [9]. As shown in Sect. 4, if $L - r > 0$, the $L - r$ smallest eigenvalues are equal to the noise variance σ^2. Then, there exists M such that $p_M > p_{M-1} = \cdots = p_1 (= \sigma^2)$. The number M may be used as the reduction order.

Finally, the convergence rates are evaluated via computer simulation when the input dimension is reduced. The learning curves of the reduce complexity cases $M = 10$ and 20, and the full complexity case $M = 31$ are shown in Fig. 8. It can be observed that the convergence rates of the reduced complexity cases are slightly faster than that of the full complexity case. The reason is described below. The time constant of learning curve in the LMS algorithm is given by [9]

$$\tau_{lms} \approx \frac{1}{2\mu\lambda_{av}} \tag{39}$$

where $\lambda_{av} = 1/M \sum_{i=1}^{M} \lambda_i$. In this simulation, since $\mu = 1/(4M)$ and $\lambda_{av} = 1$, the convergence rate becomes faster as the reduced order M becomes smaller.

8. Conclusions

This paper has been considered three issues concerning the linear adaptive receiver using the LMS algorithm for DS/CDMA systems. First, the eigenvalue spread of the autocorrelation matrix has been considered theoretically and it has been cleared that the convergence rate of the conventional LMS algorithm becomes slow when the interference signal energy becomes large. Second, we have proposed the use of an adaptive prefilter adjusted by a Hebbian learning algorithm to decorrelate the input signals. Fast convergence of the proposed receiver has been confirmed via computer simulation. Third, the complexity reduction by the truncated KLT has been considered, and it has been shown that performance degradation is little.

Finally, let us compare the proposed receiver with the RLS based receiver. If high precision arithmetic is available, the RLS algorithm provides faster convergence than the proposed receiver. Unless high precision arithmetic is available, however, the RLS algorithm fails to converge. On the other hand, the proposed receiver is robust to the finite-precision effect. The computational complexity required for the RLS algorithm is $O(L^2)$ and that for the proposed receiver is $O(LM^2)$. In summary, the RLS algorithm based receiver is useful when high precision arithmetic is available, and the length of spreading sequence L is not so long. On the contrary, the proposed receiver is useful when high precision arithmetic is not available, a fast CPU is available, and the number of users is small compared with the length of spreading sequence since M may be about $2K$.

Although the coefficient $\beta(i)$ in (28) was determined empirically in our simulation since the optimum setting depends on each problem, refining the coefficient may lead to better results. The optimum setting of the coefficient is worth investigating in the future. The performance evaluation of the proposed receiver in time variant channels is our future problem.

References

[1] S. Verdú, "Adaptive multiuser detection," Proc. ISSSTA '94, pp.43–50, July 1994.
[2] P.B. Rapajic and B.S. Vucetic, "Adaptive receiver structures for asynchronous CDMA systems," IEEE JSAC, vol.12, no.4, pp.685–697, May 1994.
[3] S. Yoshida, A. Ushirokawa, S. Yanagi, and Y. Furuya, "DS/CDMA adaptive interference canceler on differential detection in fast fading channel," Proc. 44th Vehicular Tech. Conf., pp.780–784, June 1994.
[4] K. Fukawa and H. Suzuki, "Orthogonalizing matched filter (OMF) detection for DS-CDMA mobile radio systems," Proc. IEEE Globecom 1994, pp.385–389, Nov. 1994.
[5] U. Madhow and M.L. Honig, "MMSE interference suppression for direct-sequence spread spectrum CDMA," IEEE Trans. Commun., vol.42, no.12, pp.3178–3188, Dec. 1994.
[6] S.L. Miller, "An adaptive direct-sequence code-division multiple-access receiver for multiuser interference rejection," IEEE Trans. Commun., vol.43, no.2/3/4, pp.1746–1755, Feb./March/April 1995.
[7] S.L. Miller, "Training analysis of adaptive interference suppression for direct-sequence code-division multiple-access systems," IEEE Trans. Commun., vol.44, no.4, pp.488–495, April 1996.
[8] K.B. Lee, "Orthogonalization based adaptive interference suppression for direct-sequence code-division multiple-access systems," IEEE Trans. Commun., vol.44, no.9, pp.1082–1085, Sept. 1996.
[9] S. Haykin, "Adaptive Filter Theory," 3rd-ed., Prentice Hall, New Jersey, 1996.
[10] F. Beaufays, "Transform-domain adaptive filters: An analytical approach," IEEE Trans. Signal Process., vol.43, no.2, pp.422–431, Feb. 1995.
[11] T.D. Sanger, "Optimal unsupervised learning in a single-layer linear feedforward neural network," Neural Networks, vol.2, pp.459–473, 1989.
[12] L.-H. Chen and S. Chang, "An adaptive learning algorithm for principal component analysis," IEEE Trans. Neural Networks, vol.6, no.5, pp.1255–1263, Sept. 1995.
[13] S. Haykin, "Neural Networks—A Comprehensive Foundation," Macmillan, New York, 1994.
[14] I.T. Jolliffe, "Principal Component Analysis," Springer-Verlag, New York, 1986.
[15] R. Zurmühl, "MATRIZEN," Springer-Verlag, 1964.
[16] M. Iri, "Linear Algebra II," Iwanami Shoten, Tokyo, 1994.
[17] S. Kodama and N. Suda, "Matrix Theory for System Control," Soc. of Instrument and Control Eng., Tokyo, 1978.
[18] M. Mori, M. Sugihara, and K. Murota, "Linear Computation," Iwanami Shoten, Tokyo, 1994.

Teruyuki Miyajima was born in 1966 in Japan. He received his B.E., M.E. and Ph.D. degrees in Electrical Engineering from Saitama University in 1989, 1991 and 1994, respectively. In 1994 he joined the Faculty of Engineering, Ibaraki University as Research Associate. His research interests are in communication systems, neural networks and signal processing. He is a member of IEEE and JNNS (The Japanese Neural Network Society).

Kazuo Yamanaka was born in 1950 in Japan. He received his B.S., M.S. and Doctor's degrees from Waseda University, Tokyo, Japan in 1974, 1976 and 1979, respectively. In 1979 he joined Ibaraki University, Hitachi, Japan as Research Associate. He has been a Professor of Systems Engineering, Ibaraki University since 1993. His research interests include estimation theory and system theoretical aspects of neural networks.