Journal of Southeast University (English Edition) Mar. 2002 Vol.18 No.1 ISSN 1003—7985

Performance of Wavelet-Transform-Domain Adaptive Equalizers*

Wu Bingyang** Chen Qifan Cheng Shixin

(National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China)

Abstract: In this paper performances of wavelet-transform-domain (WTD) adaptive equalizers based on the least-mean-square (LMS) algorithm are analyzed. The optimum Wiener solution, the condition of convergence, the minimum mean-square-error (MSE) and the steady-state excess MSE of the WTD adaptive equalizer are obtained. Constant and time-varying convergence factor adaptive algorithms are studied respectively. Computational complexities of WTD LMS equalizers are given. The equalizer in WTD shows much better convergence performance than that of the conventional in time-domain.

Key words: wavelet, transform-domain, wavelet-transform-domain, LMS, adaptive equalizer

Adaptive equalizers are widely used in communications. Equalizers of this type are generally implemented in time-domain by tapped-delay-line (TDL) forms, and the Widrow-Hoff adaptive LMS algorithm is used to obtain the equalizer parameters. One drawback of the LMS algorithm is that its convergence speed decreases as the ratio of the maximum to the minimum eigenvalue of the input auto-correlation matrix increases[1].

To increase the convergence speed of the LMS algorithm, a known self-orthogonalization algorithm can be used. An important property of the algorithm is that, in theory, it guarantees a constant rate of convergence, irrespective of the input statistics[1]. The objective may be realized by using the Karhunen-Loeve transform (KLT). However the KLT is a signal-dependent transform, which makes the KLT impractical for real-time applications.

Narayan et al. proposed transform-domain LMS (TRLMS) algorithms[2], where the discrete Fourier transform (DFT) and the discrete cosine transform (DCT) were used in the LMS algorithm. DFT and DCT provide predetermined sets of basis vectors that are good approximation to the KLT, and indeed, the TRLMS significantly improved convergence. After that, Lee and Un discussed the performance of TRLMS adaptive digital filter (ADF)[3].

Because of its flexibility in base choice and good properties of time localization and frequency selection, the discrete wavelet transform (DWT) is a good alternative of FFT and DCT in adaptive algorithms. Many researchers studied various ADF structures based on DWT[4-7], and these structures have encouraging performances. Some researchers used DWT-ADF in system identifications and equalizations and obtained good results[8-10].

Although the WTD adaptive equalizer is known to improve convergence speed significantly, its general properties including the steady-state performance and the convergence condition are not known yet. In the following sections we will discuss the general performance of the WTD adaptive equalizer.

The subsequent content of this paper is organized as the following. In section 1, we briefly review the adaptive equalizer, the related theory of DWT and the structures of DWT based adaptive equalizers. In section 2, we discuss the performance of the WTD adaptive equalizer using constant convergence factors, which includes the steady-state MSE, the condition of convergence and the speed of convergence. In section 3, we study the self-orthogonalization algorithms in WTD equalizers. In section 4, we analyze the computational complexities of WTD adaptive algorithms. Finally, in section 5 and 6, we give some simulation results and conclusions.

1 DWT and WTD Equalizer

1.1 Adaptive equalization

In this subsection we give a very brief review of adaptive equalization, mainly the linear equalizer based on the LMS algorithm.

It is known that the algorithm for adjusting the tap weight coefficients of the TDL adaptive equalizer may be expressed in the form

$$\hat{\boldsymbol{C}}_{n+1} = \hat{\boldsymbol{C}}_n + \mu \boldsymbol{X}_n e_n \tag{1}$$

Received 2001-10-09.

* The project supported by the National Natural Science Foundation of China(69982002).

** Born in 1970, male, graduate.

where $\hat{\boldsymbol{C}}_n$ denotes the estimate of the weighting coefficient vector; $\boldsymbol{X}_n$ is the vector of received signal samples; e_n is the error signal at the nth iteration and μ is a positive number chosen small enough to ensure convergence of the iterative procedure.

1.2 DWT

As we know, if $x(t)$ is any square integrable function, then it can be decomposed onto a set of square integrable basis functions, constructed by dilating and translating a single wavelet

$$x(t) = \sum_{j,k \in Z} r_{j,k} \cdot \psi_{j,k}(t) \tag{2}$$

where $\psi_{j,k}(t) = 2^{-j/2}\psi(2^{-j}t - k)$ are wavelet bases and $\psi(t)$ is called the mother wavelet base.

The coefficients of the bases can be calculated by the following equation:

$$r_{j,k} = \int x(t)\psi^*_{j,k}(t)\mathrm{d}t \tag{3}$$

which is defined as wavelet transform, where "$*$" denotes complex conjugate.

For the sequence $\{x(n)\}$ sampled from $x(t)$, the DWT coefficients of the sequence is

$$r_{j,k} = \sum_n x(n) \cdot \psi^*_{j,k}(n) \tag{4}$$

The accumulation in Eq. (4) can be written in a concision form. Assuming vector $\boldsymbol{X} = [x(n), x(n-1), \cdots, x(n-M+1)]^{\mathrm{T}}$ denotes the input sequence, where $[\cdot]^{\mathrm{T}}$ means vector transposition, the coefficient vector of DWT can be represented as

$$\boldsymbol{U} = \boldsymbol{\Psi} \cdot \boldsymbol{X} \tag{5}$$

where $\boldsymbol{U} = [U_1, U_2, \cdots, U_J]$, in which $\boldsymbol{U}_j(j = 1,2,\cdots, J)$ is the coefficient vector in the jth scalar, and $\boldsymbol{\Psi}$ is wavelet transform matrix, which is determined by scaling functions. In practice, the wavelet transform coefficients are computed using a dyadic sub-band tree structure, and so the wavelet transform matrix is constructed from wavelet filters.

Observing that we pass a periodic sequence of period N through a cascade of 2-band regular perfect reconstruction filter (RPRF) banks and critically decimate the output of every filter, we obtain the DWT coefficients of each scalar.

To compute the output of a cascade of RPRF banks we evaluate the product[11]

$$\boldsymbol{Q}_{\log_2(N)}(\boldsymbol{Q}_{\log_2(N)-1}(\cdots(\boldsymbol{Q}_2(\boldsymbol{Q}_1\boldsymbol{X})\cdots) \tag{6}$$

where

$$\boldsymbol{Q}_k = \begin{bmatrix} \boldsymbol{I}_{N-2^{\log_2(N)-k-1}} & 0 \\ 0 & \boldsymbol{D}_k \end{bmatrix} \tag{7}$$

In Eq. (7), $\boldsymbol{I}_n$ denotes an $n \times n$ identity matrix and $\boldsymbol{D}_k$ is the transform matrix in the kth stage of RPRF banks

$$\boldsymbol{D}_k = \begin{bmatrix} \boldsymbol{H}_1^{(k)} \\ \boldsymbol{H}_0^{(k)} \end{bmatrix} \tag{8}$$

and it can further be represented as an $(N/2^k) \times (N/2^k)$ matrix

$$\boldsymbol{D}_k = \begin{bmatrix} c_1^2 & c_1^3 & \cdots & 0 & 0 & 0 & c_1^0 & c_1^1 \\ c_1^0 & c_1^1 & c_1^2 & c_1^3 & \cdots & 0 & 0 & 0 \\ 0 & 0 & c_1^0 & c_1^1 & c_1^2 & c_1^3 & \cdots & 0 \\ \vdots & & & & & & & \vdots \\ c_0^2 & c_0^3 & \cdots & 0 & 0 & 0 & c_0^0 & c_0^1 \\ c_0^0 & c_0^1 & c_0^2 & c_0^3 & \cdots & 0 & 0 & 0 \\ 0 & 0 & c_0^0 & c_0^1 & c_0^2 & c_0^3 & \cdots & 0 \\ \vdots & & & & & & & \vdots \end{bmatrix} \tag{9}$$

where c_1^k and $c_0^k(k = 0, 1, 2,\cdots)$ are the coefficients of high-pass and low-pass filters respectively.

The decimation by 2 is taken into account in this matrix by shifting twice the coefficients of the filters to the right.

Let the "wavelet transform matrix" be

$$\boldsymbol{\Psi} = \boldsymbol{Q}_{\log_2(N)}\boldsymbol{Q}_{\log_2(N)-1}\cdots\boldsymbol{Q}_2\boldsymbol{Q}_1 \tag{10}$$

It is shown that for RPRF banks

$$\boldsymbol{\Psi}^{\mathrm{T}} \cdot \boldsymbol{\Psi} = \boldsymbol{I} \tag{11}$$

which means that $\boldsymbol{\Psi}$ is an orthogonal transform matrix.

1.3 WTD adaptive equalizer

From above subsection, we see that, like FFT and DCT, DWT is an orthogonal transform. Some advantages over FFT and DCT can be found in wavelet transform: ① Compact support in time-domain of wavelet bases make discontinuities and Gibbs phenomena associated with Fourier transforms applied to consecutive blocks of data be avoided; ② Good frequency selection properties make the frequency-domain be precisely divided apart; ③ Simplicity of filter structure makes it easy to implement by hardware. All of these properties make DWT an encouraging candidate in transform-domain adaptive algorithms.

The structure of DWT based equalizer is shown in Fig.1.

In Fig.1, the input sequence $\boldsymbol{X}$ is the received signal, the vector $\boldsymbol{U}$ is the DWT of $\boldsymbol{X}$ and the desired signal comes from the training sequence or decisions at the output of the detector. The output of the equalizer is

$$y_n = \boldsymbol{W}_n^{\mathrm{T}} \cdot \boldsymbol{U} \tag{12}$$

where $\boldsymbol{W}_n = [\omega_{n,0}, \omega_{n,1}, \cdots, \omega_{n,N-1}]^{\mathrm{T}}$ is the coefficient vector. The update algorithm is

$$\hat{\boldsymbol{W}}_{n+1} = \hat{\boldsymbol{W}}_n + \mu\boldsymbol{U}_n e_n \tag{13}$$

where $\hat{\boldsymbol{W}}_n$ denotes the estimate of the coefficients vector, and μ is a constant or time-varying positive number.

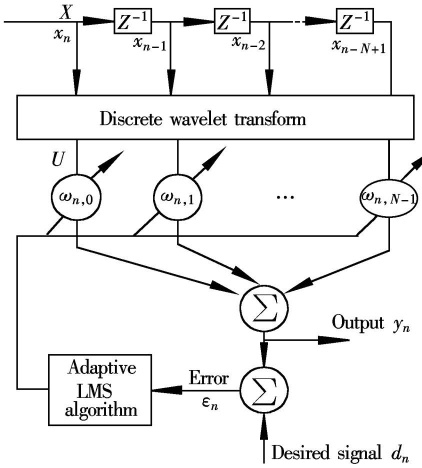

Fig. 1 Structure of WTD adaptive equalizer

1.4 Wiener solution

According to Ref. [3], it is easy to demonstrate the following results:

1) The Wiener solution of the coefficients of equalizer in WTD is

$$\omega_{opt} = \boldsymbol{\Psi}^{T} \cdot \boldsymbol{c}_{opt} \tag{14}$$

where c_{opt} is the Wiener solution in the time-domain.

2) The minimum MSE of the equalizer in WTD ε^{w}_{min} and that of the time-domain ε_{min} are identical, i.e.

$$\varepsilon_{min} = \varepsilon^{w}_{min} \tag{15}$$

2 WTD LMS Equalizer Using Constant Convergence Factor

In this section, we assume that the convergence factor μ is constant as in standard LMS algorithm. We discuss the convergence conditions, the time constant and the steady-state MSE of the WTD equalizer.

2.1 Convergence condition

It is known that, in the time-domain, the sufficient condition for the statistical expectation $E[C_n]$ converging to C_{opt} *is*

$$0 < \mu < \frac{2}{\mathrm{tr}(\boldsymbol{R}_x)} \tag{16}$$

where $\mathrm{tr}(\boldsymbol{R}_x)$ denotes the trace of the matrix $\boldsymbol{R}_x$, and

$$\boldsymbol{R}_x = E[\boldsymbol{X}_n \boldsymbol{X}_n^{T}] = N\sigma_x^2 \tag{17}$$

where $\sigma_x^2 = E[x_n x_n^{*}]$.

In WTD, the convergence condition should be

$$0 < \mu < \frac{2}{\mathrm{tr}(\boldsymbol{R}_u)} \tag{18}$$

and

$$\begin{aligned}\mathrm{tr}(\boldsymbol{R}_u) &= \mathrm{tr}(E[\boldsymbol{U}_n \boldsymbol{U}_n^{T}]) = \\ &\mathrm{tr}(E[\boldsymbol{\Psi X}_n \boldsymbol{X}_n^{T} \boldsymbol{\Psi}^{T}]) = \\ &E[\mathrm{tr}(\boldsymbol{\Psi X}_n \boldsymbol{X}_n^{T} \boldsymbol{\Psi}^{T})] = \\ &E[\mathrm{tr}(\boldsymbol{\Psi}^{T} \boldsymbol{\Psi X}_n \boldsymbol{X}_n^{T})] = \\ &E[\mathrm{tr}(\boldsymbol{X}_n \boldsymbol{X}_n^{T})] = \\ &\mathrm{tr}(E[\boldsymbol{X}_n \boldsymbol{X}_n^{T}]) = \mathrm{tr}(\boldsymbol{R}_x)\end{aligned} \tag{19}$$

So in WTD, for constant factor, the number μ can keep the same as that in the time-domain to ensure convergence.

2.2 Steady-state MSE

The steady-state MSE of the LMS algorithm is expressed in terms of the minimum MSE and the excess MSE ε_{Δ} as

$$E[\varepsilon_{ss}] = \varepsilon_{min} + \varepsilon_{\Delta} \tag{20}$$

where

$$\varepsilon_{\Delta} = \frac{1}{2}\mu\,\mathrm{tr}(\boldsymbol{R}_x)\varepsilon_{min} = \frac{1}{2}\mu N\sigma_x^2\varepsilon_{min} \tag{21}$$

From Eqs. (15) and (19), it is easily shown that $E[\varepsilon_{ss}]$ in the WTD and that of in the time-domain are identical.

2.3 Time constants

The time constants of the MSE process $E[\varepsilon_n]$, which denote the convergence speed, are

$$\tau_i = \frac{1}{2\mu\lambda_i} \qquad 1 \leqslant i \leqslant N \tag{22}$$

where $\{\lambda_i, 1 \leqslant i \leqslant N\}$ are the eigenvalues of $\boldsymbol{R}_x$ in time-domain or $\boldsymbol{R}_u$ in WTD. The convergence procedure is mainly controlled by the minimum eigenvalue. Since $\boldsymbol{R}_u = \boldsymbol{\Psi R}_x \boldsymbol{\Psi}^{T}$ and $\boldsymbol{\Psi}$ is an orthogonal matrix, the eigenvalues of $\boldsymbol{R}_u$ and $\boldsymbol{R}_x$ are identical. Therefore the WTD and the time-domain LMS algorithms have the same convergence speed.

From above discussion, it is concluded that if constant convergence factor is used in WTD LMS, the convergence performance will not be improved at all.

3 Self-Orthogonalizing WTD LMS Equalizer

Let the factor μ in Eq.(13) be

$$\mu = \gamma \boldsymbol{R}_u^{-1} \tag{23}$$

where γ is a constant number, and then we get the self-orthogonaling algorithm.

$$\hat{\boldsymbol{W}}_{n+1} = \hat{\boldsymbol{W}}_n + \gamma \boldsymbol{R}_u^{-1} \boldsymbol{U}_n e_n \tag{24}$$

The convergence condition and the excess MSE are

$$0 < \gamma < \frac{2}{\mathrm{tr}(\boldsymbol{R}_u^{-1}\boldsymbol{R}_u)} = \frac{2}{N} \tag{25}$$

and

$$\varepsilon_{\Delta} = \frac{1}{2}\gamma\,\mathrm{tr}(\boldsymbol{R}_u^{-1}\boldsymbol{R}_u)\varepsilon_{min} = \frac{1}{2}\gamma N\varepsilon_{min} \tag{26}$$

respectively. If $\sigma_x^2 = 1$, they are all the same as the constant factor case. However, for the matrix that controls the convergence is the identity matrix and all the eigenvalues are identical, the convergence speed improves significantly. The convergence is controlled by γ only no matter what the statistic of the input is.

The next work is to estimate $\boldsymbol{R}_u$ and thus compute $\boldsymbol{R}_u^{-1}$. If the inputs are white noise sequences, the covariance matrix of the inputs is diagonal and it is easy to compute $\boldsymbol{R}_u^{-1}$.

Fortunately, Ref. [11] tells us that, assuming a zero mean random process $f(n)$, DWT make the cross covariance between x_n and x_m (the nth and the mth elements of $f(n)$) decay exponentially fast to zero as the distance between m and n increases. Hence the cross covariance matrix of the samples of $f(n)$ be closer to a diagonal matrix. The following simulations in section 5 illustrate the above conclusion.

If the matrix is very closer to a diagonal one, we can set elements outside the main diagonal to zero and get an approximation to the matrix to simplify the computation. Then we can get a simple WTD LMS algorithm

$$\hat{\omega}_m(n+1) = \hat{\omega}_m(n) + \frac{\gamma}{\sigma_m^2(n)} e(n)\mu_m^*(n)$$
$$m = 0, 1, \cdots, N-1 \tag{27}$$

where $(\cdot)_m$ means the mth element in the sequence and $\sigma_m^2(n)$ is the power estimate of the mth element in vector $\boldsymbol{U}$. The estimate is updated by

$$\sigma_m^2(n+1) = \beta \cdot \sigma_m^2(n) + (1-\beta)\left|u_m(n)\right|^2 \tag{28}$$

where $0 < \beta < 1$ is called forgetting factor.

Considering the cross covariance of signals in different scales of vector space is less than that in the same scale. Then the WTD LMS adaptive mechanism can be represented as

$$\omega_{j,k}(n+1) = \omega_{j,k}(n) + \frac{\gamma}{\sigma_j^2(n)} e(n)\mu_{j,k}^*(n) \tag{29}$$

where $(\cdot)_{j,k}$ means the kth element in jth scale. $\sigma_j^2(n)$ is the estimate of the average power for the out of scale j, and is updated by

$$\sigma_j^2(n+1) = \beta \cdot \sigma_j^2(n) + (1-\beta)\frac{1}{K_j}\sum_{k=1}^{K_j}\left|u_{j,k}(n)\right|^2 \tag{30}$$

where K_j is the element number in jth scale.

A more complex algorithm to compute $\boldsymbol{R}_u^{-1}$ can be seen in Ref. [11], where M-band filter banks and Cholesky factor was used to invert matrices.

4 Computational Complexity

In the following discussion on evaluating the computational efficiency of an algorithm we consider only the number of multiplications, which is not dependent of the processor architecture. Here only real-valued case is considered.

We analyze the computational complexity of DWT from Eq. (5) first. To compute the DWT of a finite N-point segments of data$\{x(n)\}_{n=1}^{N}$, we assume that the signal $\{x(n)\}$ is periodic with period N, and the DWT of the vector $\boldsymbol{X}$ is then to compute $\boldsymbol{\Psi X}$. For the wavelet transform matrix $\boldsymbol{\Psi}$ is an $N \times N$ matrix, the number of multiplication is N^2 at most. If the not zero element number L in each row of $\boldsymbol{\Psi}$ is much smaller than N ($L \ll N$), i.e. $\boldsymbol{\Psi}$ is a sparse matrix, we need to perform LN multiplication operations to compute $\boldsymbol{\Psi X}$ only. Of course the number of multiplication LN is much smaller than N^2. Considering L is independent of N, the number of multiplication is $O(N)$.

Then from the cascade of filter banks, we discuss the computational complexity of DWT further. We know that each stage of the filter banks has the same structure and filter parameters (see Fig.2). We assume that each filter has M coefficients: c_0^0, c_0^1, $\cdots$, c_0^{M-1} or c_1^0, c_1^1, $\cdots$, c_1^{M-1}, where c_0 and c_1 are low pass and high pass filter coefficients respectively. In the first stage of the filter banks, the N-point segments are filtered and decimated. From Fig.2, it is clear that the number of multiplication is $2MN$. However we can use a more efficient structure, which is shown in Fig.3. The h^{O} and h^{E} in Fig.3 are the odd and even parts of the filter coefficients. The operation is reduced to MN. Considering the linear phase restriction $h(n) = h(N-1-n)$ and the relations between high pass and low pass $h_1(k) = (-1)^k h_0(1-k)$, we only need to perform $1/4MN$ multiplication. In the second stage, because the samples out of the low pass filter are decimated by 2, the multiplication number is $1/4M \cdot 1/2N$. So the total multiplication is

$$1/4M(N + 1/2N + 1/4N + \cdots) < 1/2MN \tag{31}$$

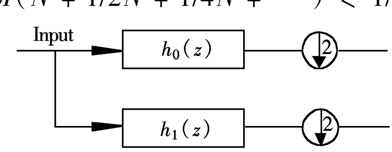

Fig.2 Low and high pass filters

Then we consider the computational complexity of the updating algorithm of Eqs.(27) and (29).

From Eqs. (27) and (28), we can see that to update $\hat{\omega}_m(n)$ we need to perform $6N + 1$ multiplications and including DWT the total multiplication of WTD LMS is

$$(1/2M + 6)N + 1 \tag{32}$$

For the case of Eq. (29), we can see that to compute each $\sigma_j^2(n+1)$ in Eq. (30) we need to

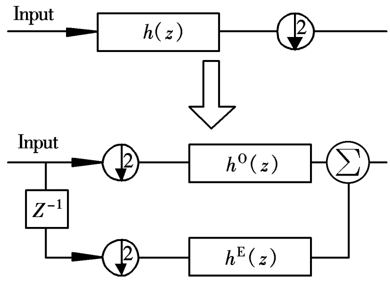

Fig.3 Efficient structure of filter bank

perform $K_j + 2$ multiplications, and to compute all the $\sigma_j^2(n+1)$ we need to perform $N + 2J$ multiplications, where $J \leqslant \log_2(N)$ is the maximum scale of DWT. So to update the WTD LMS adaptive equalizer coefficients in Eq.(29), we need to perform multiplications

$$(1/2M + 4)N + 2J + 1 \tag{33}$$

Using Eqs. (29) and (30), we need some less computation than Eqs.(27) and (28), and if $M \ll N$, the computational complexities are all $O(N)$. Compared to DCT and FFT based adaptive LMS algorithm[3], the computational complexities of DWT LMS do not increase.

5 Computer Simulation Results and Discussion

In this section, we present some computer simulation results of WTD LMS equalization. The structure we used is shown in Fig.4.

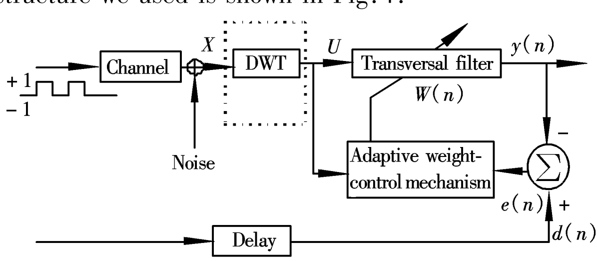

Fig.4 Block diagram of WTD LMS equalizer

In Fig.4 the input we used in the transmitting point is a sequence of random sign functions and the channel has the following impulse response, which is controlled by "w" in the equation:

$$h(n) = \begin{cases} \dfrac{1}{2}\left[1 + \cos\left(\dfrac{2\pi}{w}(n-2)\right)\right] & n = 1,2,3 \\ 0 & \text{otherwise} \end{cases} \tag{34}$$

Passing the input through the low pass channel, we got the received signal in the equalizer. Then we used a 16 points sliding window to truncate the received sequence, and got vectors $\boldsymbol{X}(n)$. After DWT processing, we got vectors $\boldsymbol{U}(n)$. And in the following parts, LMS algorithm was used to update to coefficient factor $\boldsymbol{W}(n)$.

Firstly, we illustrated the diagonalization ability of DWT. We assumed that $\boldsymbol{R}_x$ and $\boldsymbol{R}_u$ denote the covariance matrix of $X(n)$ and $U(N)$, respectively. We computed the ratios of the diagonal norm (norm of elements in the main diagonal) to the Frobenius norm of the same matrix. The more ratio means the more diagonal of course. The ratios are shown in Tab.1. In Tab.1, we used different channel parameter "w" in Eq. (34), and different Daubechies' wavelet bases, "db2", "db4" and "db8". For a stationary zero-mean, first-order Markov process, the DCT is asymptotically equivalent to the KLT. So DCT was considered in the table as a reference.

Tab.1 Ratios of diagonal norm to Frobenius norm

Base		w				
		2.9	3.1	3.3	3.5	3.7
R_x		0.875 8	0.831 3	0.795 7	0.768 8	0.749 0
R_u	db2	0.967 1	0.957 1	0.949 6	0.944 4	0.940 8
	db4	0.974 1	0.966 6	0.961 3	0.957 7	0.955 3
	db8	0.982 9	0.978 1	0.974 7	0.972 5	0.971 0
DCT		0.993 7	0.991 7	0.990 1	0.988 9	0.988 1

From Tab.1, we see that the correlation matrix is nearly diagnalized after wavelet transforms, and in these selected bases, "db8" is the best one. After wavelet transform, the diagnalization of the input sequence correlation matrix is very close to that of DCT — the approximation of KLT of low pass processes.

Then we present the computer simulation results of WTD LMS equalizer learning curves compared with that of standard LMS. In simulations the signal to noise ratio (SNR) is 30 dB, the wavelet used in our wavelet transform based adaptive filters is the Daubechies' D8 wavelet. The update algorithms we used are Eqs.(29) and (30).

For different w in Eq.(34), the spread of eigenvalues of the correlation matrix $\boldsymbol{R}$ is different. We compare the convergence properties of LMS and WLMS under some bad conditions: $w = 3.5$ and $w = 3.7$ when the spread is 54.22 and 156.32 respectively, and good conditions: $w = 2.9$, and $w = 3.1$ when the spread is 6.31 and 11.79 respectively.

The learning curves for the algorithms mentioned above can be seen in Fig.5 and Fig.6. Each of these MSE curves is obtained by taking average of 100 independent computations of the squared error data and the worst 10 samples were excluded.

The curves indicate that, as expected, for the same condition (eigenvalues spread), the WTD LMS equalizer converges faster than that of LMS. Especially when the spread is large, the convergence rate is improved distinctively.

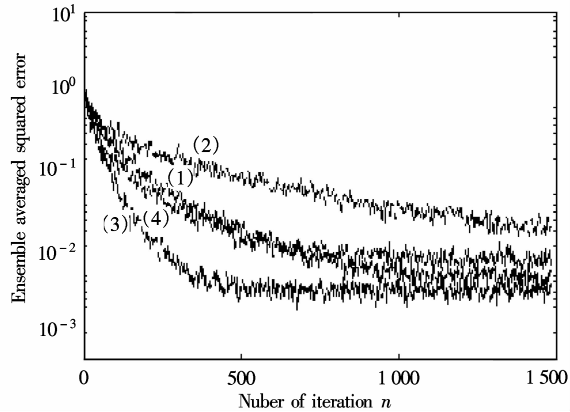

Fig.5 Learning curves of LMS. ((1) $w = 3.5$, (2) $w = 3.7$) and WTD LMS ((3) $w = 3.5$, (4) $w = 3.7$)equalizer

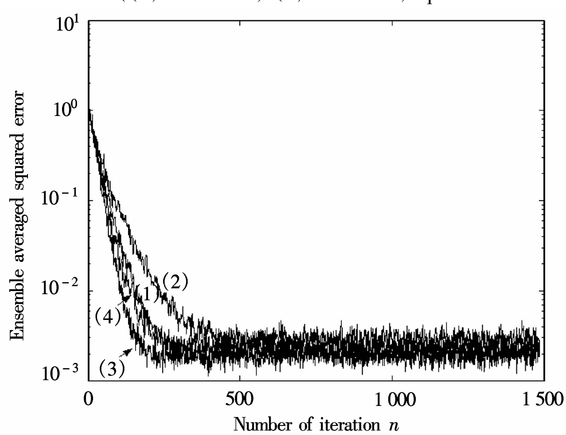

Fig.6 Learning curves of LMS ((1) $w = 2.9$, (2) $w = 3.1$) and WTD LMS ((3) $w = 2.9$, (4) $w = 3.1$)equalizer

6 Conclusions

1) Discrete wavelet transforms, under some conditions, are orthogonal transforms and it can be operated by cascade of filter banks, which results in reduced computational complexity.

2) DWT can be used in adaptive equalizer. Using constant convergence factor, the convergence condition, the steady-state MSE and the time constants of WTD LMS are all the same as those of standard LMS. Using self-orthogonal algorithm, WTD LMS largely improve the convergence performance of equalizers, especially when the eigenvalue spread of the covariance matrix of inputs is large. The computational complexity of WTD LMS using time-varying factor is $O(N)$ only.

3) Computer simulations show that WTD LMS have better convergence performance than the time-domain LMS.

References

[1] Haykin S. *Adaptive Filter Theory*[M]. 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 1996. 445 - 478.

[2] Narayan S S, Peterson A M, Narasimha M J. Transform domain LMS algorithm[J]. *IEEE Trans Acoust Speech Signal Processing*, 1983, **31**(6): 609 - 615.

[3] Lee J C, Un C K. Performance of transform-domain LMS adaptive digital filters[J]. *IEEE Trans Acoust Speech Signal Processing*, 1986, **34**(6): 499 - 510.

[4] Doroslovacki M, Fan H. Wavelet-based linear system modeling and adaptive filtering[J]. *IEEE Trans Signal Processing*, 1996, **44**(5): 1156 - 1167.

[5] Gonzalez-Prelcic N, Gonzalez F P, Gonzalez A B M. A flexible structure for wavelet packet-based subband adaptive equalization[A]. *Time-Frequency and Time-Scale Analysis on Proceedings of the IEEE-SP International Symposium*[C]. 1998. 377 - 380.

[6] Erd˙ ol N, Basbug F. Performance of wavelet transform based adaptive filters[A]. *IEEE Int Conf Acoust Speech Signal Processing*[C]. 1993, 3: 500 - 503.

[7] Hosur S, Tewfik A H. Wavelet transform domain LMS algorithm[A]. *IEEE Int Conf Acoust Speech Signal Processing*[C]. 1993, 3: 508 - 510.

[8] Zhang Y, Dill J. Comparison of equalization techniques in a wavelet packets based multicarrier modulation DS-CDMA system[A]. *GLOBECOM* '99, *IEEE*[C]. 1999, **4**: 2152 - 2156.

[9] Gracias S, Reddy V U. An equalization algorithm for wavelet packet based modulation schemes[J]. *IEEE Trans Signal Processing*, 1998, **46**(11): 3082 - 3087.

[10] Liu Feng, Cheng Jun, Xu Jinbiao, et al. Wavelet based adaptive equalization algorithm[A]. *GLOBECOM* '97, *IEEE*[C]. 1997, **3**: 1230 - 1234.

[11] Tewfik A H, Kim. Fast positive definite linear system solvers[J]. *IEEE Trans Signal Processing*, 1994, **42**(3): 572 - 585.

小波变换域自适应均衡器性能分析

吴炳洋 陈琦帆 程时昕

(东南大学移动通信国家重点实验室,南京 210096)

摘 要 介绍了基于LMS算法的小波变换域自适应均衡器,并分析了此类均衡器的性能.较为详细地研究了小波域自适应均衡器的维纳解、收敛条件、均方误差等问题.针对时不变及时变收敛因子2种情况进行了讨论.给出了小波域均衡算法的算法复杂度.仿真结果表明小波域自适应均衡比其时域算法具有更好的收敛性.

关键词 小波,变换域,小波变换域,LMS,自适应均衡器

中图分类号 TN929.533