LETTER

A Fast Converging RLS Equaliser

Tetsuya SHIMAMURA†, ***Regular Member***

SUMMARY It is well known that based on the structure of a transversal filter, the RLS equaliser provides the fastest convergence in stationary environments. This paper addresses an adaptive transversal equaliser which has the potential to provide more faster convergence than the RLS equaliser. A comparison is made with respect to computational complexity required for each update of equaliser coefficients, and computer simulations are demonstrated to show the superiority of the proposed equaliser.

key words: *equaliser, transversal filter, RLS algorithm, channel estimator, fast convergence*

1. Introduction

One of the most important things subjected to adaptive equalisers is that of convergence speed in the training mode. This is true, in particular, for polling communication systems where fast start-up equalisation, which is accomplished by fast convergence for the equaliser in the training mode, is often required.

For the last three decades, much efforts were made toward the development of a fast converging adaptive equaliser [3]–[12]. While the least mean square (LMS) transversal equaliser [1],[2] was modified by most of researchers to achieve faster convergence than the original one, Godard [3] derived a transversal equaliser whose coefficients are adapted by the Kalman filter or the recursive least squares (RLS) algorithm. The Godard's RLS equaliser was much more complicated for the adaptation of the equaliser coefficients than the LMS equaliser, but accomplished much faster convergence in a relatively insensitive fashion to variations in the eigenvalue spread of the input correlation matrix. Also, by Gitlin et al. [5], it was shown that the RLS equaliser is the best self-orthogonalising adaptive equaliser to accomplish the fastest initial convergence. Based on these results, it is well known that the RLS equaliser provides the gradient adaptation to achieve the fastest convergence based on the structure of a linear transversal filter in stationary environments.

On the other hand, recently, Shimamura et al. [13] proposed a two-stage equaliser which behaves robustly against additive noise as well as the eigenvalue spread of the input correlation matrix. The structure of the two-stage equaliser is also suitable for tracking rapidly time variant communications channels [14]. The two-stage scheme involves two linear transversal filters; channel estimator and equalisation filter. The coefficients of the equalisation filter is updated by using the Butler-Cantoni method being a non-adaptive method [15] for each iteration, based on the results of the channel estimator. Thus, the two-stage equaliser is recognised as one realisation of adaptive version of the Butler-Cantoni method. This is why the two-stage equaliser is called adaptive Butler Cantoni (ABC) equaliser in this paper.

This paper investigates the performance for the ABC equaliser on stationary channels. We deploy a channel estimator adapted by the RLS algorithm instead of the LMS algorithm used in [13], and investigate particularly the convergence of the resulting ABC equaliser, the RLS based ABC equaliser.

Manuscript received May 12, 2000.
†The author is with the Faculty of Engineering, Saitama University, Urawa-shi, 338-8570 Japan.

2. RLS Equaliser

Let us assume the following discrete-time finite impulse response channel model:

$$x_k = \sum_{i=0}^{L-1} h_i u_{k-i} + n_k \tag{1}$$

where $h_o, h_1, ..., h_{L-1}$ is the channel coefficients, n_k is a Gaussian white noise, and u_k is the transmitted sequence being pseudo-random sequence with the values $+1$ or -1. The output of the channel, x_k, is used as an input sequence for the equaliser.

Figure 1 shows the configuration of a linear transversal equaliser in the training mode where it is assumed that a delayed transmitted sequence, u_{k-d}, is obtained at the receiver side. When the RLS adaptation is used for the linear transversal equaliser, the

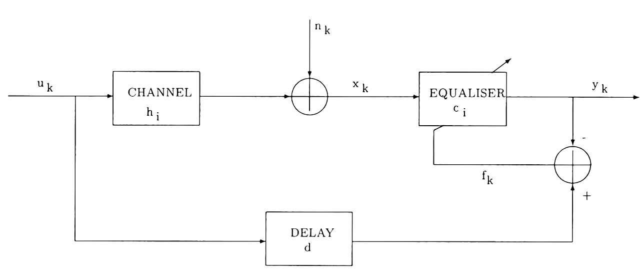

Fig. 1 Linear transversal equaliser.

coefficient update procedure is described as follows:

$$y_k = \mathbf{x}(k)^T \mathbf{c}(k-1) \tag{2}$$

$$f_k = u_{k-d} - y_k \tag{3}$$

$$\mathbf{K_c}(k) = \frac{\mathbf{P_c}(k-1)\mathbf{x}(k)^T}{\lambda + \mathbf{x}(k)^T \mathbf{P_c}(k-1)\mathbf{x}(k)} \tag{4}$$

$$\mathbf{P_c}(k) = \frac{1}{\lambda}[\mathbf{P_c}(k-1) - \mathbf{K_c}(k)\mathbf{x}(k)^T \mathbf{P_c}(k-1)] \tag{5}$$

$$\mathbf{c}(k) = \mathbf{c}(k-1) + \mathbf{K_c}(k) f_k \tag{6}$$

where $\mathbf{c}(k)$ is the equaliser coefficient vector for k iterations given by

$$\mathbf{c}(k) = [c_0(k), c_1(k), , , , c_{M-1}(k)]^T, \tag{7}$$

and $\mathbf{x}(k)$ is the input vector given by

$$\mathbf{x}(k) = [x_k, x_{k-1}, , , , x_{k-M+1}]^T. \tag{8}$$

The $\mathbf{K_c}(k)$ is the Kalman gain, and λ corresponds to the forgetting factor. The above RLS adaptation is initialised by

$$\mathbf{c}(0) = \mathbf{0} \tag{9}$$

$$\mathbf{P}(0) = \frac{1}{\alpha}\mathbf{I} \tag{10}$$

where $\mathbf{I}$ denotes the unit matrix and α is a positive real number.

The convergence of the coefficient vector for the RLS equaliser is, for $M \ll k < \infty$, subject to [16]

$$E[\mathbf{c}(k)] = \mathbf{c}_{opt} - \frac{\alpha}{k}\mathbf{A}^{-1}\mathbf{c}_{opt} \tag{11}$$

where $E[\cdot]$ denotes the expectation operation and $\mathbf{c}_{opt}$ means the optimal estimate of the coefficient vector. The $\mathbf{A}$ corresponds to the input correlation matrix. Equation (11) suggests that the RLS equaliser asymptotically provides an unbiased estimate of the coefficient vector, and that the equalisation error decreases linearly with the evolution of time or the number of iterations.

In a similar fashion with (11), the mean square error (MSE) for the RLS equaliser is given by [3]

$$E[f_k^2] = \sigma^2 + \frac{M}{k}\sigma^2 \tag{12}$$

where σ^2 denotes the variance of the output error of the equaliser whose coefficients are given by $\mathbf{c}_{opt}$. Equation (12) suggests that the excess MSE increases linearly with the number of coefficients and decreases with the number of iterations.

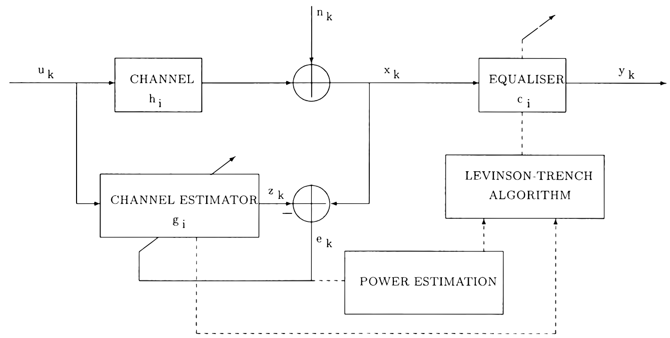

Fig. 2 Adaptive Butler-Cantoni equaliser.

3. ABC Equaliser

Figure 2 shows the configuration of the ABC equaliser where the equalisation filter has M coefficients, the output of which is given by

$$y_k = \mathbf{x}(k)^T \mathbf{c}(k). \tag{13}$$

The channel estimator has N coefficients, $g_0(k)$, $g_1(k), ..., g_{N-1}(k)$, the output of which is given by

$$z_k = \sum_{i=0}^{N-1} g_i(k) u_{k-i}. \tag{14}$$

For the channel estimation, we deploy the RLS adaptation scheme as follows:

$$e_k = x_k - z_k \tag{15}$$

$$\mathbf{K_g}(k) = \frac{\mathbf{P_g}(k-1)\mathbf{u}(k)^T}{\lambda + \mathbf{u}(k)^T \mathbf{P_g}(k-1)\mathbf{u}(k)} \tag{16}$$

$$\mathbf{P_g}(k) = \frac{1}{\lambda}[\mathbf{P_g}(k-1) - \mathbf{K_g}(k)\mathbf{u}(k)^T \mathbf{P_g}(k-1)] \tag{17}$$

$$\mathbf{g}(k) = \mathbf{g}(k-1) + \mathbf{K_g}(k) e_k \tag{18}$$

where $\mathbf{g}(k)$ is the coefficient vector for the channel estimator given by

$$\mathbf{g}(k) = [g_0(k), g_1(k), , , , g_{N-1}(k)]^T, \tag{19}$$

and $\mathbf{u}(k)$ is the input vector given by

$$\mathbf{u}(k) = [u_k, u_{k-1}, , , , u_{k-N+1}]^T. \tag{20}$$

The channel estimator invokes the estimate of noise variance as well as the estimate of channel coefficients. The variance of additive noise is estimated by the average of e_k^2 such as

$$\hat{\sigma^2}(k) = \frac{1}{P}\sum_{i=0}^{P-1}(e_{k-i})^2 \tag{21}$$

where P is the sample number to be required for the average operation.

Based on the results of the channel estimator, the coefficients of the equalisation filter is obtained by solving for

$$\mathbf{c}(k) = \mathbf{A}(k)^{-1}\mathbf{b}(k) \tag{22}$$

for each iteration, where $\mathbf{A}(k)$ and $\mathbf{b}(k)$ denote the input correlation matrix and cross-correlation vector for k iterations. The above Eq. (22) is efficiently solved by making use of the Levinson-Trench algorithm [15]. Because the elements of $\mathbf{A}$ and $\mathbf{b}$ are, in the ideal case, given by

$$a_{ij} = \sum_{m=0}^{L-1} h_m h_{m+|i-j|} + \sigma^2\delta(i-j) \qquad i,j = 0,1,,,,M-1 \tag{23}$$

$$b_i = h_{d-i} \qquad i = 0,1,,,,M-1 \tag{24}$$

respectively where $\delta(\cdot)$ denotes the Kronecker delta function, we can use the following

$$a_{ij}(k) = \sum_{m=0}^{N-1} g_m(k) g_{m+|i-j|}(k) + \hat{\sigma}^2(k)\delta(i-j) \qquad i,j = 0,1,,,,M-1 \tag{25}$$

$$b_i(k) = g_{d-i}(k) \qquad i = 0,1,,,,M-1 \tag{26}$$

for the elements of $\mathbf{A}$ and $\mathbf{b}$, respectively. Adapting the channel estimator and evaluating Eqs. (22),(25) and (26) simultaneously, we can produce the output of the equalisation filter, y_k, which turns out to be the estimate of u_{k-d}.

The equalisation filter for the ABC equaliser is indirectly adapted by the channel estimator. This means that the ABC equaliser replaces the task of channel equalisation with that of channel estimation. This replacement may be beneficial for any adaptive algorithm, because for channel estimation, the input sequence is pseudo-random, leading to the ideal state for adaptive systems, while for channel equalisation, in general, the input sequence is coloured, making a certain spread of eigenvalues of the input correlation matrix. It is well known that the convergence of an adaptive algorithm is affected by the input sequence. In particular, the LMS algorithm is rescued by the replacement, because its convergence is largely affected by the eigenvalue spread of the input correlation matrix. A large spread of that causes the LMS adaptation requiring a long time to reach the state of convergence. Due to this reason, the ABC equaliser involving the LMS channel estimator behaves insensitively to variations in the eigenvalue spread of the input correlation matrix, and provides faster convergence than the LMS equaliser [13].

If the LMS channel estimator is replaced by the RLS channel estimator, then it is expected that more advantages are obtained for the ABC equaliser. This is because for system identification problem, which is equivalent to the channel estimation problem, the RLS adaptation provides faster convergence than the LMS adaptation [18],[19].

A remaining subject is to compare the RLS based ABC equaliser with the RLS equaliser. One may be interested in knowing which provides better performance by commonly using the RLS adaptation.

With respect to the speed of convergence, the RLS based ABC equaliser may be superior to the RLS equaliser. This is because for the RLS based ABC equaliser, the adaptation is subjected to the RLS channel estimator and the convergence of the coefficient vector is given by

$$E[\mathbf{g}(k)] = \mathbf{g}_{opt} - \frac{\alpha}{k}\mathbf{g}_{opt} \tag{27}$$

where $\mathbf{g}_{opt}$ means the optimal channel coefficient vector. Equation (27) produces the best convergence curve the RLS adaptation can provide, because the convergence is not affected by the input correlation matrix for the RLS adaptive filter. On the other hand, for the RLS equaliser, the convergence is relatively insensitive to variations in the eigenvalue spread of the input correlation matrix. However, as shown by (11), the convergence of the coefficient vector is certainly affected by $\mathbf{A}^{-1}$. If the coefficient error vector

$$\epsilon(k) = \mathbf{c}_{opt} - \mathbf{c}(k) \tag{28}$$

is deployed here, then the affection is more visually shown by evaluating the coefficient error correlation matrix

$$\mathbf{R}(k) = E[\epsilon(k)\epsilon(k)^T]. \tag{29}$$

As the number of iterations approaches infinity, $\mathbf{R}(k)$ is approximated by

$$\mathbf{R}(k) \approx \frac{\sigma^2}{k}\mathbf{A}^{-1}. \tag{30}$$

Taking the norm of both sides, we get

$$||\mathbf{R}(k)|| \approx \frac{\sigma^2}{k}||\mathbf{A}^{-1}||. \tag{31}$$

If the minimum eigenvalue of the input correlation matrix is expressed by λ_{min}, then $||\mathbf{A}^{-1}||$ reduces to $1/\lambda_{min}$. Therefore, we get the following

$$||\mathbf{R}(k)|| \approx \frac{\sigma^2}{k\lambda_{min}}. \tag{32}$$

This equation obviously suggests that when λ_{min} is small, the convergence degrades and requires a long time.

Usually, the length of channel estimator may be less than that of channel equaliser. This fact leads to further two advantages for the RLS based ABC equaliser. One is the reduction in computational complexity, which will be discussed in the next section. The

Table 1 Computational complexity of adaptive equalisation algorithms. The RLS, ABC(RLS) and ABC(LMS) denote the RLS, RLS based ABC and LMS based ABC equalisers, respectively.

Algorithm	Multiplications	Divisions
RLS	$2.5M^2 + 4.5M$	2
ABC(RLS)	$2M^2 - 3M + 2.5N^2 + 4.5N + 2$	$2M - 2$
ABC(LMS)	$2M^2 - 3M + 4N + 3$	$2M - 2$

other is obtained from the excess MSE being proportional to the number of coefficients to be adapted. For the RLS based ABC equaliser, the expression of (12) is changed such as

$$E[e_k^2] = \eta^2 + \frac{N}{k}\eta^2 \tag{33}$$

where η^2 corresponds to the output error variance for the channel estimator, which is less than σ^2 in (12). When $N \ll M$, Eq. (33) suggests that the RLS channel estimation is more accurate than the RLS equalisation. This means that the equalisation filter directly derived from the channel estimator for the RLS based ABC equaliser essentially behaves more accurately than the RLS equaliser.

4. Computational Complexity

The RLS based ABC equaliser requires, at each adaptation step of the channel estimator, to solve for a set of M linear equations and to estimate the variance of the additive noise. The computational requirements are $\frac{5}{2}N^2 + \frac{9}{2}N$ multiplications and 2 divisions per iteration for the channel estimator, which are derived from the RLS adaptation results in [17], and $2M^2 - 3M$ multiplications and $2M - 3$ divisions per iteration for the Levinson-Trench algorithm. Furthermore, the calculations of the variance of additive noise is added. If we use the following operation for this purpose:

$$\hat{\sigma}^2(k) = \hat{\sigma}^2(k-1) + \frac{e_k^2 - e_{k-P}^2}{P} \tag{34}$$

the computational requirements become 2 multiplications and 1 division per iteration for the estimation of the variance of additive noise. The total computational requirements for the RLS based ABC equaliser are summarised in Table 1, in which the computational requirements for the RLS and LMS based ABC equalisers are also shown as a comparison.

Figure 3 shows a comparative result of the computational requirements for the RLS, RLS based ABC and LMS based ABC equalisers where the total operations of multiplications plus divisions are compared. For the ABC equalisers, the channel estimator is assumed to have 3 coefficients ($N = 3$). Figure 3 suggests that when the equaliser order is larger than 5, the computational requirements for the RLS based ABC equaliser become less than those for the RLS equaliser.

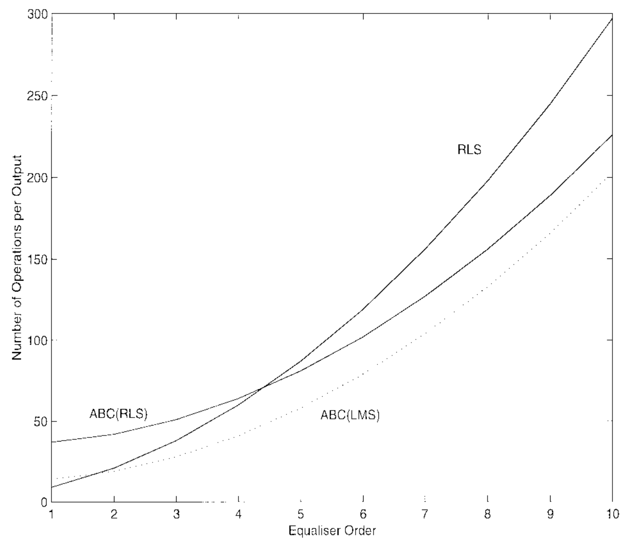

Fig. 3 Computational complexity for the ABC and RLS equalisers.

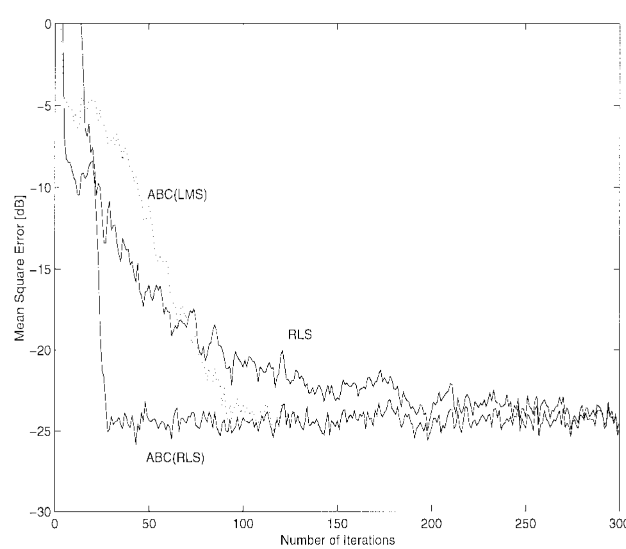

Fig. 4 Convergence of the ABC and RLS equalisers (SNR = 50 dB).

5. Simulations

To verify the performance of the RLS based ABC equaliser, we carried out computer simulations. Figure 4 shows the convergence for the channel the transfer function of which is given by

$$H(z) = 0.3482 + 0.8704z^{-1} + 0.3482z^{-2}, \tag{35}$$

where the RLS, RLS based ABC and LMS based ABC equalisers are compared. The signal-to noise ratio(SNR) is 50 dB. The three equalisers have the same equalisation filter length $M = 8$ and delay $d = 4$. For the ABC equalisers, the channel estimator length is

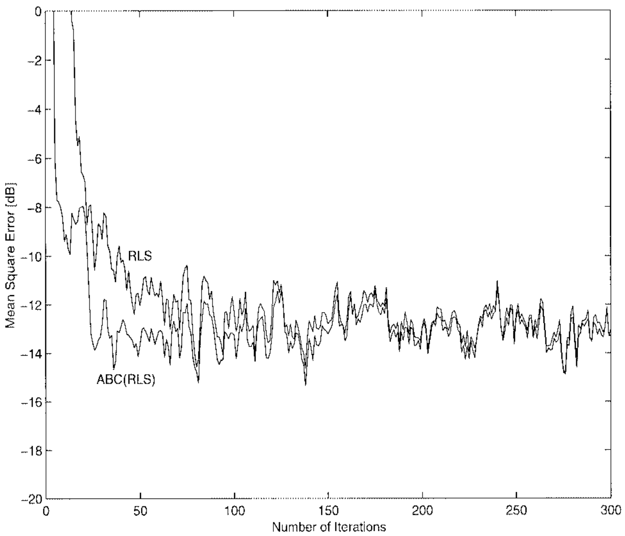

Fig. 5 Convergence of the RLS based ABC and RLS equalisers (SNR=20 dB).

$N = 3$, which is equivalent to the channel length. Constant parameter has been commonly set to the unity as the forgetting factor ($\lambda = 1.0$) for the RLS algorithm to be used in the RLS and RLS based ABC equalisers. For the LMS based ABC equaliser, the step size parameter for the LMS algorithm has been set to 0.04. Each convergence curve in Fig. 4 has been evaluated by averaging 100 individual trials. Figure 4 obviously shows that the RLS based ABC equaliser provides faster convergence than the RLS equaliser. This is due to the transient performance of the RLS estimator for the RLS based ABC equaliser, which is best in the case of the input of pseudo-random sequence or white noise. Because the channel (35) is ill-conditioned for an equaliser and produces a small minimum eigenvalue of the input correlation matrix, the convergence of the RLS equaliser degrades due to the small eigenvalue as shown by (11) and (32).

From Fig. 4, it is also observed that the use of the LMS based ABC equaliser is enough to outperform that of the RLS equaliser. This suggests how the replacement of channel equalisation with channel estimation for the ABC equaliser is beneficial to accelerate the speed of convergence.

Figure 5 shows the convergence of the RLS based ABC and RLS equalisers under the same conditions as those in Fig. 4 except for the SNR. In Fig. 5, the SNR is 20 dB. Comparing Fig. 5 with Fig. 4, we notice that the convergence speed of the RLS based ABC equaliser is not affected by the amount of additive noise, while that of the RLS equaliser is affected. This result for the RLS equaliser is obvious from (12). When the initial value of $E[f_k^2]$, $E[f_0^2]$, is much larger than σ^2, the time to reach the steady state becomes long. Thus, the case of SNR=50 dB invokes slower convergence for the RLS equaliser than the case of SNR=20 dB. On the other hand, for the RLS based ABC equaliser, the convergence behaviour is mostly determined by (27), because the estimates of $g_0(k), g_1(k), ..., g_{n-1}(k)$ are directly used for the Levinson-Trench algorithm to produce the equaliser output. Although σ^2 is also required for the Levinson-Trench algorithm, the value of η^2 is essentially less than σ^2 and thus is insensitive to the evaluation of (25). These factors preserve the insensitivity to additive noise for the RLS based ABC equaliser.

6. Conclusion

The speed of convergence of adaptive equalisers has been investigated on stationary channels. It has been shown that the ABC equaliser provides better performance than the RLS equaliser without increasing computational complexity. From the results obtained through this paper, we deduce that the RLS based ABC equaliser provides the fastest convergence based on the structure of a linear transversal filter in the sense which the convergence of the RLS based ABC equaliser outperforms that of the RLS equaliser.

References

[1] J. Proakis and J.H. Miller, "An adaptive receiver for digital signaling through channels with intersymbol interference," IEEE Trans. Inf. Theory, vol.IT-15, no.4, pp.484–497, July 1969.

[2] C. Ungerboeck, "Theory on the speed of convergence in adaptive equalizers for digital communication," IBM J. Res. Develop., pp.546–555, Nov. 1972.

[3] D. Godard, "Channel equalization using a Kalman filter for data transmission," IBM J. Res. Develop., pp.267–273, May 1974.

[4] R.W. Chang, "A new equalizer structure for fast start-up digital communication," Bell System Technical Journal, vol.50, no.6, pp.1969–2014, July–Aug. 1971.

[5] R. Gitlin and F.R. Magee Jr., "Self-orthogonalizing adaptive equalization algorithms," IEEE Trans. Commun., vol.COM-25, no.7, pp.666–672, July 1977.

[6] D.D. Falconer and L. Ljung, "Application of fast Kalman estimation to adaptive equalization," IEEE Trans. Commun., vol.COM-26, no.10, pp.1439–1446, Oct. 1978.

[7] E.H. Satorius and S.T. Alexander, "Channel equalization using adaptive lattice algorithms," IEEE Trans. Commun., vol.COM-27, no.6, pp.899–905, June 1979.

[8] E.H. Satorius and J.D. Pack, "Application of least squares lattice algorithms to adaptive equalization," IEEE Trans. Commun., vol.COM-29, no.2, pp.136–142, Feb. 1981.

[9] W.Y. Chen, "Adaptive pilot filtering for LMS algorithm," Proc. IEEE Int. Conf. Acoust., Speech & Signal Process., pp.1509–1512, 1991.

[10] K.A. Hamied, M. Rahman, and M.S. El-Hennawey, "A new channel estimator for fast start-up equalization," IEEE Trans. Commun., vol.39, no.2, pp.177–181, Feb. 1991.

[11] K. Wesolowski, C.M. Zhao, and W. Rupprecht, "Adaptive LMS transversal filters with controlled length," IEE Proc. Pt.F, vol.139, no.3, pp.233–238, June 1992.

[12] S.H. Leung and B.L. Chan, "Fast converging LMS equaliser," Electron. Lett., vol.29, no.10, pp.829–830, May 1993.

[13] T. Shimamura and J. Suzuki, "Adaptalisation of Bulter-Cantoni method for fast start-up equalisation," IEICE Trans. vol.J81-A, no.4, pp.622–630, April 1998.
[14] T. Shimamura, S. Semnani, and C.F.N. Cowan, "Equalisation of time variant communications channels via channel estimation based approaches," Signal Processing, vol.60, no.2, pp.181–193, July 1997.
[15] P. Butler and A. Cantoni, "Noniterative automatic equalization," IEEE Trans. Commun., vol.COM-23, no.6, pp.621–633, June 1975.
[16] S. Haykin, Adaptive Filter Theory, Prentice-Hall, Englewood Cliffs, 1991.
[17] J.G. Proakis, Digital Communications, McGraw-Hill, 1989.
[18] C.F.N. Cowan, "Performance comparisons of finite linear adaptive filters," IEE Proc. vol.134, Pt.F, no.3, pp.211–216, June 1987.
[19] B. Mulgrew and C.F.N. Cowan, Adaptive Filters and Equalisers, Kluwer Academic Pub., 1988.