PAPER

Equalizer-Aided Time Delay Tracking Based on L_1-Normed Finite Differences

Jonah GAMBA[†a)], ***Student Member*** **and Tetsuya SHIMAMURA**[†], ***Member***

SUMMARY   This paper addresses the estimation of time delay between two spatially separated noisy signals by system identification modeling with the input and output corrupted by additive white Gaussian noise. The proposed method is based on a modified adaptive Butler-Cantoni equalizer that decouples noise variance estimation from channel estimation. The bias in time delay estimates that is induced by input noise is reduced by an IIR whitening filter whose coefficients are found by the Burg algorithm. For step time-variant delays, a dual mode operation scheme is adopted in which we define a normal operating (tracking) mode and an interrupt operating (optimization) mode. In the tracking mode, only a few coefficients of the impulse response vector are monitored through L_1-normed finite forward differences tracking, while in the optimization mode, the time delay optimized. Simulation results confirm the superiority of the proposed approach at low signal-to-noise ratios.

key words: *time delay estimation, adaptive Butler-Cantoni equalizer, delay optimization, L_1-norm, finite difference*

1. Introduction

1.1 Background

Time delay estimation (TDE) between noise-corrupted signals incident on two spatially separated sensors is important in various fields such as radar, sonar and geophysics [3]. The TDE problem considers two discrete-time signals incident on two sensors, which are sampled at time $t = kT_s$ where T_s is the sampling period and expressed as

$$x_k = s_k + v_k \tag{1}$$
$$y_k = s_{k-D} + n_k \tag{2}$$

where x_k and y_k are the noisy reference and delayed signals, respectively. The s_k corresponds to the noise-free source signal and s_{k-D} is its delayed one. The n_k and v_k are the uncorrelated zero-mean white Gaussian noise of variances σ_n^2 and σ_v^2, respectively. The index parameter D represents the unknown time delay to be estimated, which is approximated to an integer closest to the true delay in the discrete-time model. The TDE problem from the observations of two signals as given by (1) and (2) is very well-established and various solutions can be found in the literature [2]–[9]. Among them, we focus in this paper on adaptive TDE and assume sinusoidal source signals.
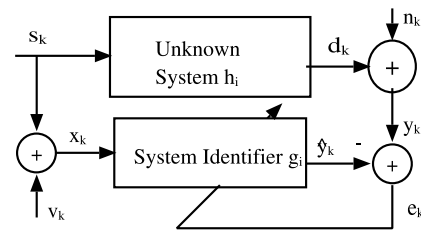
Fig. 1   Modeling time delay estimation as system identification. The sequences $\{h_i\}$ and $\{g_i\}$ are the system impulse response and identifier coefficients, respectively.

The LMS algorithm is popularly used for TDE due to its simplicity [7], [12]. In employing the LMS algorithm, the TDE is modeled as a system identification problem with the sampled input x_k and output y_k as shown in Fig. 1. The problem here is to estimate the system impulse response which is ideally equal to z^{-D}. Commonly, the solution minimizes the mean-square error (MSE) at the output of the system identifier. However, in the form depicted in Fig. 1, the use of the LMS algorithm has been known to result in a biased estimate of the time delay due to the presence of the input noise v_k [6].

In this paper we propose an LMS-based TDE method, referred to as the ABCTDE method. The method uses a modified version of the adaptive Butler-Cantoni (ABC) equalizer derived in [13]. For the ABC equalizer, the unknown system is a communication channel. The ABC method decouples the noise variance estimation from the channel estimation process and uses indirect updating of equalizer filter coefficients. We utilize this decoupling property to obtain the time delay. The use of the adaptive equalizer affords us the opportunity to use the output error and simultaneously control the effects of the input noise.

To allow for time-variant delays to be tracked, a dual mode operation scheme is considered in which we define a normal operating mode and an interrupt operating mode. In the normal operating mode, only a few coefficients of the impulse response vector are monitored, while in the interrupt operating mode, delay optimization is performed. This dual mode provides a superior performance in the case where the delay changes abruptly after some period of stability.

1.2 Motivation of the Proposed Method

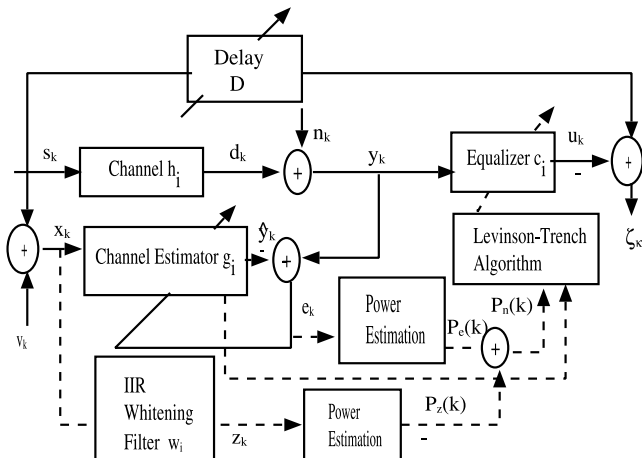It was argued and shown in [12] that two popular measures

of performance for adaptive filters, namely the time constant and mean square error (MSE), are not good performance indicators for TDE because the weights, from which the time delay is extracted, converge to the required time delay much faster than the MSE (for example in a fraction of the time constant). The signal-to-noise ratio (SNR) becomes an important factor in adaptive TDE, and hence noise elimination could speed-up the convergence to the correct delay. The initial gradient of MSE convergence characteristics also becomes very important. Hence a steep gradient in the first iterations will be desired. A two-stage approach to TDE utilizing steep initial gradient MSE convergence characteristics and removal of the effects of noise in both the input and output could be more effective. The adaptive equalizer based on the Butler-Cantoni method [1] derived in [13] suits the two-stage approach at reduced computational complexity. The two-stage approach has been actually employed in [14] for TDE, and shown to perform well as a preliminary work.

A recent method based on system identification modeling of TDE has been presented in [15], which has good convergence properties and robustness against noise. However, the method assumes that the ratio of the input to output noise variances needs to be known *a priori* and to be identically equal to unity. If the noise variance ratio is not known *a priori*, then it may be necessary to perform singular value decomposition [17] at each iteration and hence it would be computationally expensive.

The proposed method brings two new points to the conventional methods. Firstly, the delay optimization is based on noise-compensated coefficients, thus reducing the bias in the time delay estimate. Secondly, the time delay is estimated under less constraints on the input and output noise. The proposed method can provide better performance in low input SNR environments, and also in situations where the time delay and/or noise variance statistics can be variable with time. One such application where the noise power could be equal to or greater than the signal power is in underwater acoustics [4].

This paper is organized as follows. Section 2 describes the ABC equalizer. In Sect. 3, we describes how unbiased time delay estimates can be found. Section 4 gives details of the proposed method and discuss computational complexity. Section 5 gives simulation results for tracking of time delay at different SNR's. In Sect. 6 we consider the tracking of time variant delays. Finally, we conclude this paper in Sect. 7.

2. ABC Equalization Scheme

This section describes the ABC equalizer. The ABC equalizer coefficient-update equations and its properties are given.

2.1 ABC Equalizer

Figure 2 shows the ABC equalizer configuration [13]. The equalizer coefficient vector at the kth iteration,

Fig. 2 Configuration of the ABC equalizer (h_i is the channel impulse response, g_i the estimator coefficients and c_i the equalizer coefficients).

$\mathbf{c}(k) = [c_0(k), \ldots, c_{M-1}(k)]^T$, can be expressed as

$$\mathbf{c}(k) = \mathbf{A}(k)^{-1}\mathbf{b}(k) \tag{3}$$

where $\mathbf{A}(k)$ is an $M \times M$ matrix whose elements are given by

$$a_{ij}(k) = \sum_{m=0}^{L-1} h_m(k)h_{m+|i-j|}(k) + \sigma^2\delta(i-j)$$
$$i, j = 0, \ldots, M-1 \tag{4}$$

and $\mathbf{b}(k)$ is an $M \times 1$ vector

$$b_i(k) = h_{l-i}(k), \; i = 0, \ldots, M-1 \tag{5}$$

where l is the system delay for nonminimum phase channels, and σ^2 the variance of the channel output noise n_k and $\delta(\cdot)$ denotes the Kronecker delta function. The vector $\mathbf{h}(k) = [h_0(k), \ldots, h_{L-1}(k)]^T$ is the channel impulse response at time k.

Using the channel estimator output error e_k, the variance of the channel output noise is estimated based on

$$\hat{\sigma}^2(k) = \frac{1}{P}\sum_{i=0}^{P-1}(e_{k-i})^2 \tag{6}$$

where P is the number of samples required to be averaged.

The vector $\mathbf{g}(k) = [g_0(k), \ldots, g_{N-1}(k)]^T$ where T denotes transpose is the channel estimate at the kth iteration. The channel estimator parameters are updated by the LMS algorithm using the equation

$$\mathbf{g}(k + 1) = \mathbf{g}(k) + \mu e_k \mathbf{r}(k) \tag{7}$$

where μ is the step-size parameter to be suitably chosen and $\mathbf{r}(k)$ is the input vector given by $\mathbf{r}(k) = [r_k, \ldots, r_{k-N+1}]^T$.

In [10], (3) is solved by the Levinson-Trench algorithm where $g_i(k)$ obtained from (7) and $\hat{\sigma}^2(k)$ obtained from (6) are used instead of $h_i(k)$ and σ^2, respectively, in (4) and (5). The equalizer output o_k is produced by the convolution of the equalizer input q_k with c_i.

The equalizer is primarily used for the elimination of intersymbol interference (ISI) of band-limited dispersive channels. Various equalizer configurations can be found in the literature [21]. One such configuration is the ABC equalizer [13]. For the ABC equalizer, minimizing the MSE of the channel estimator leads to the equalizer design.

Fig. 3 Configuration of the proposed ABCTDE method (v_k is the input noise, $P_e(k)$ is the channel estimator output error power and $P_n(k)$ channel output noise power, z_k and $P_z(k)$ are the IIR filter output error and the error power at time k, respectively).

2.2 The ABC Equalizer for TDE

The proposed TDE scheme, ABCTDE method, is shown in Fig. 3. The channel output in the ABCTDE method is the delayed input signal, s_{k-D}. The input to the equalizer becomes $y_k = s_{k-D} + n_k$. The equalizer coefficients can be obtained by using the output error defined by

$$\zeta_k = d_k - \hat{d}_k \tag{8}$$

where $d_k = s_{k-D}$ denotes the desired signal and $\hat{d}_k = u_k$ the equalizer output. By employing the orthogonality principle, we seek the equalizer coefficients that make the error ζ_k orthogonal to the equalizer input sequence, that is,

$$E[\zeta_k y_{k-m}] = 0 \tag{9}$$

for $m = 0, \ldots M - 1$, where $E[\cdot]$ denotes expectation. Expanding the left hand side of (9) results in

$$
\begin{aligned}
E[\zeta_k y_{k-m}] &= E[(d_k - \hat{d}_k)(d_{k-m} + n_{k-m})] \\
&= E[d_k d_{k-m}] \\
&\quad - \sum_{j=0}^{M-1} c_j E[(d_{k-j} + n_{k-j}) d_{k-m}] \\
&= E[d_k d_{k-m}] \\
&\quad - \sum_{j=0}^{M-1} c_j E[d_{k-j} d_{k-m}] \tag{10}
\end{aligned}
$$

where we used the fact that the desired sequence and the noise n_k are uncorrelated to arrive at (10). The term under the summation in (10) can further be simplified if we

consider the fact that for time delay estimation problem under consideration, the channel impulse response is such that $h_i(k) = 1$ for $i = D$ and $h_i(k) = 0$ for $i \neq D$. If we further put the constraint that $\mathbf{h}^T(k)\mathbf{c}(k) = 1$, then we can write (10) as

$$
\begin{aligned}
E[\zeta_k y_{k-m}] &= E[d_k d_{k-m}] \\
&\quad - \sum_{j=0}^{M-1} c_j E[s_{k-j-D} s_{k-m-D}] \\
&= E[d_k d_{k-m}] - E[s_{k-2D} s_{k-m-D}] \\
&= R_{dd}(m) - R_{ss}(m - D) \tag{11}
\end{aligned}
$$

where $R_{dd}(l) = E[d_k d_{k-l}]$ and $R_{ss}(m - D) = E[s_{k-2D} s_{k-m-D}]$. The orthogonality principle is satisfied if

$$R_{dd}(m) = R_{ss}(m - D). \tag{12}$$

By appropriately setting the delay D, the equalizer coefficients can be found for which (12) is satisfied, and hence, the error ζ_k is minimized. Since the desired channel output is unknown and only the noise-corrupted version is available, the left hand side of (12) cannot be evaluated. For the ABC equalizer, the optimum delay D is obtained during the training period of delay tracking for which a training sequence is required. In the tracking mode, the delay is fixed for the equalizer but changes in the error ζ_k are monitored since the changes can be used to indicate a change in the impulse response of the channel or a change in the delay. However, the ABC method cannot be used without modification when both the input and output are corrupted by additive white noise as described in the next section.

3. Unbiased Estimate for Time Delay

In this section the effect of input noise on the estimated channel coefficients is shown. We also show how to get unbiased time delay estimates from the estimated channel coefficients.

3.1 Relationship with the Conventional Techniques

As shown in Fig. 3, the observed noise-corrupted desired sequence is given by

$$y_k = \sum_{i=0}^{L-1} h_i(k) s_{k-i} + n_k = d_k + n_k. \tag{13}$$

The output of the channel estimator when it is excited by the noise-corrupted input x_k of (1) is

$$\hat{y}_k = \sum_{i=0}^{N-1} g_i(k) s_{k-i} + \sum_{i=0}^{N-1} g_i(k) v_{k-i}. \tag{14}$$

The channel estimator output error is given by

$$e_k = y_k - \hat{y}_k. \tag{15}$$

Lets define here as follows:

$$d_k = \sum_{i=0}^{L-1} h_i s_{k-i}$$

$$\tilde{d}_k = \sum_{i=0}^{N-1} g_i s_{k-i}$$

$$f_k = \sum_{i=0}^{N-1} g_i v_{k-i}.$$

With the above definitions it can be shown from (15) that

$$
\begin{aligned}
e_k &= (d_k + n_k) - (\tilde{d}_k + f_k) \\
&= (d_k - \tilde{d}_k) + (n_k - f_k) \\
&= e_d(k) + e_n(k)
\end{aligned}
$$ (16)

with obvious notation. The error power, which includes the additive noise variances, becomes

$$(e_k)^2 = (e_d(k))^2 + (e_n(k))^2 + 2e_n(k)e_d(k).$$

When the input noise for the channel estimator, v_k, is absent, the channel estimator could eliminate the noise effects as

$$E[(e_k)^2] = E[(e_d(k))^2 + (n_k)^2] = \sigma_n^2(k).$$ (17)

This is because the channel estimator output error is predominantly due to the additive noise. In this case, the channel estimator provides an unbiased estimate of the vector $\mathbf{h}(k)$. When the input noise for the channel estimator, v_k, is present, on the other hand,

$$
\begin{aligned}
E[(e_k)^2] &= E[(e_d(k))^2 + (e_n(k))^2] \\
&= \sigma_n^2(k) + \hat{\sigma}_v^2(k).
\end{aligned}
$$ (18)

This means that $\hat{\sigma}_v^2(k)$ remains a part of $E[(e_k)^2]$ and could result in a biased estimate of the vector $\mathbf{h}(k)$. In fact we can show the degree of bias as follows. Substituting for $\hat{y}_k$ in (15) yields

$$e_k = y_k - \sum_{i=0}^{N-1} g_i(k) x_{k-i}.$$ (19)

Squaring both sides of (19), substituting for y_k and x_{k-i}, setting $L = N$, and taking expectations gives the MSE as

$$
\begin{aligned}
E[(e_k)^2] &= \sigma_s^2(k) \sum_{i=0}^{N-1} [h_i(k) - g_i(k)]^2 \\
&\quad + \sigma_n^2(k) + \sigma_v^2(k) \sum_{i=0}^{N-1} g_i(k)^2
\end{aligned}
$$ (20)

where $\sigma_s^2(k)$ is the variance of the channel input signal. The bias in the parameters can be seen if we differentiate (20) with respect to $g_i(k)$ and equating the result to zero, that is,

$$\sigma_s^2(k) \sum_{i=0}^{N-1} h_i(k) = (\sigma_s^2(k) + \sigma_v^2(k)) \sum_{i=0}^{N-1} g_i(k)$$

$$
\begin{aligned}
\sum_{i=0}^{N-1} g_i(k) &= \frac{SNR_i(k)}{SNR_i(k) + 1} \sum_{i=0}^{N-1} h_i(k) \\
&= \beta(k) \sum_{i=0}^{N-1} h_i(k)
\end{aligned}
$$ (21)

where $SNR_i(k) = \frac{\sigma_s^2(k)}{\sigma_v^2(k)}$ denotes the input signal-to-noise ratio and $\beta(k) = \frac{SNR_i(k)}{SNR_i(k)+1}$. It is clearly seen that the input noise variance introduces bias in the estimated channel coefficients determined by the value of $\beta(k)$. Let us write (18) as

$$P_e(k) = \sigma_n^2(k) + \hat{\sigma}_v^2(k).$$ (22)

Dividing $P_e(k)$ in (22) by $\hat{\sigma}_v^2(k)$ gives

$$
\begin{aligned}
P_e(k)/\hat{\sigma}_v^2(k) &= \sigma_n^2(k)/\hat{\sigma}_v^2(k) + \hat{\sigma}_v^2(k)/\hat{\sigma}_v^2(k) \\
&= R(k) + 1
\end{aligned}
$$ (23)

where $R(k) = \sigma_n^2(k)/\hat{\sigma}_v^2(k)$. From the above equation (23) we see that if the ratio $R(k)$ is known *a priori*, then $P_e(k)/\hat{\sigma}_v^2(k)$ can be easily found. In this case, we can obtain $\hat{\sigma}_v^2(k)$ because it is possible to measure $P_e(k)$ from the channel estimator output. The problem can be solved by the approach taken in [15] where the knowledge of $R(k)$ can be used to minimize the channel output error. In practice, it may be difficult to know the ratio $R(k)$ *a priori* and we therefore the need to find a way to deal with the unknown ratio $R(k)$. Unless the input noise is completely eliminated, the channel estimator parameters will always be biased.

3.2 Unbiased Estimate

If we take a closer look at (4) and again use the knowledge that $h_i(k) = 0$ for $i \neq D$, then we can further simplify (4) to

$$
\begin{aligned}
a_{ij}(k) &= h_D(k)h_{D+|i-j|}(k) + \sigma^2(k)\delta(i-j) \\
&\qquad i, j = 0, \dots, M-1.
\end{aligned}
$$ (24)

Equation (24) suggests that $a_{ij}(k) = 0$ for $|i-j| \neq 0$ otherwise matrix $\mathbf{A}(k)$ in (3) would be an all-zero matrix. We see that $\mathbf{A}(k)$ is diagonal in our case. Without any other modification to the ABC equalizer except for the inclusion of channel estimator input noise we get elements of $\mathbf{A}(k)$ as

$$
\begin{aligned}
\hat{a}_{ij}(k) &= h_D(k)h_{D+|i-j|}(k) \\
&\quad + (\sigma_n^2(k) + \hat{\sigma}_v^2(k))\delta(i-j).
\end{aligned}
$$ (25)

In the case where we can compensate for the channel estimator input noise, then (25) reduces to

$$\hat{a}_{ij}(k) = h_D(k)h_{D+|i-j|}(k) + (\sigma_n^2(k))\delta(i-j).$$

Since these coefficients must be calculated before updating the equalizer coefficients, when the input noise level is high, reducing the bias in $a_{ij}(k)$ could improve the rate at which the equalizer coefficients attain their true values.

The proposed method is based on the principle that we extract the noise power from the noisy input signal of the

channel estimator and subtract it from $P_e(k)$ prior to updating the equalizer coefficients. Since the channel estimator coefficients and equalizer coefficients are decoupled, we can also compensate for the bias due to the input noise before adjusting the equalizer coefficients. In this way, changes in the output error ζ_k would be primarily due to the changes in the delay.

Based on the above-mentioned principle a whitening filter is used to estimate the noise in the input. As shown in Fig. 3, the z_k is the output of the IIR whitening filter and is given by

$$z_k = \sum_{i=0}^{N_1} w_i(k)x_{k-i}, \quad w_0 = 1$$

with a transfer function equals to $1/W(z)$ and whose order N_1. Let us denote the signal power of z_k as $P_z(k) = E[z_k^2]$. When the input signal consists of sinusoids corrupted by white noise (which is the same situation as assumed in this paper), the whitening filter behaves as a notch filter and could produce an output whose power approximates the noise power [19], [22], [23]. This is achieved when the output power of the whitening filter is minimized. This behavior can be explained by the innovations approach to optimal Wiener filtering, in which it is possible to use an IIR filter to predict the input signal [20]. The Burg algorithm [11] minimizes P_z and finds the filter coefficients of $1/W(z)$ which provides $P_z(k) = \hat{\sigma}_v^2(k)$. When the Burg algorithm is used, the stability of the whitening filter is assured.

The proposed method is expected to work well if the input noise is a white process. However, when the input noise is colored (correlated), the whitening filter attempts to whiten the input signal with the result that the estimate of the input noise variance, $\hat{\sigma}_v^2(k)$, becomes generally less than the expected value depending on the bandwidth and total power of the correlated input noise. The proposed method may, therefore, be unable to give satisfactory performance under these circumstances.

4. ABCTDE Method

In this section we give details of how the parameters in the ABCTDE method are updated.

4.1 Implementation

Since the set of channel coefficients $\{h_i(k)\}$ is unavailable, the first step is to obtain an estimate, $\{g_i(k)\}$, through the channel estimator. The channel estimator parameters are updated by the LMS algorithm, with the input vector $\mathbf{x}(k) = [x_k, \ldots, x_{k-N+1}]^T$. That is, the resulting update scheme is given by

$$\mathbf{g}(k + 1) = \mathbf{g}(k) + \mu e_k \mathbf{x}(k) \tag{26}$$

where the step parameter μ is appropriately chosen. Having obtained the biased channel parameters $\{g_i(k)\}$, the next step would be the estimate of the input noise variance using the

output error and the IIR filter. The output error power is estimated as

$$\hat{P}_e(k) = \frac{1}{P_1} \sum_{i=0}^{P_1-1} (e_{k-i})^2 \tag{27}$$

where P_1 is the number of data points to be averaged. We can similarly obtain the average input error power $P_z(k)$ as

$$\hat{P}_z(k) = \frac{1}{P_1} \sum_{i=0}^{P_1-1} (z_{k-i})^2. \tag{28}$$

From (27) and (28), the output noise variance to be used in the estimation of equalizer coefficients can be found as

$$\hat{P}_n(k) \approx \sigma_n^2(k) = \hat{P}_e(k) - \hat{P}_z(k). \tag{29}$$

4.2 Delay Optimization

In order to estimate the delay D in the tracking mode, we propose to use an optimization scheme as follows. By using an exhaustive search method for a permissible delay range of $[0, l_m]$, which in practice is generally known [16], and based on (8), we define the output error associated with each delay i in the range $[0, l_m]$ as

$$\zeta_{ik} = s_{k-i} - \hat{d}_k, \quad 0 \leq i \leq l_m. \tag{30}$$

The delay estimate is initially set to zero. At each iteration, we minimize the error with respect to each delay and use the delay corresponding to the minimum error as the delay estimate at that time. Denote this delay as $D_{est}(k)$. The estimate of the equalizer coefficients can be written as

$$\hat{\mathbf{c}}(k) = \tilde{\mathbf{A}}(k)^{-1}\hat{\mathbf{b}}(k) \tag{31}$$

where $\tilde{\mathbf{A}}(k)$ is an $M \times M$ matrix whose elements are given by

$$\begin{aligned}\tilde{a}_{ij}(k) = g_{D_{est}}(k)g_{D_{est}+|i-j|}(k) \\ + \hat{P}_n(k)\delta(i - j)\end{aligned} \tag{32}$$

and $\hat{\mathbf{b}}(k)$ has elements

$$\hat{b}_i(k) = g_{D_{est}-i}, \quad i = 0, \ldots M - 1. \tag{33}$$

This calculation of equalizer coefficients is done for the first Q (small) iterations. Since the delay estimate at each iteration is $D_{est}(k - 1)$, at the $(Q + 1)$st iteration an estimate of the optimal delay $\hat{D}$ is calculated as

$$\hat{D} = D_{est}(Q). \tag{34}$$

4.3 Computational Complexity

The computational complexity of the ABC equalizer in multiplications per iterations was analyzed in [18] and can be shown to be $2M^2 - M + 4(N + 1)$. The additional step of noise variance estimation employing the Burg algorithm has the computational complexity of order $O(N^2)$. If N is assumed

to be equal to M, then the overall computational complexity would be of order $O(M^2)$. On the other hand, if the input and output noise variances are not known *a priori*, the method proposed in [15] may require performing singular value decomposition at each iteration. Hence the computational complexity would be of order $O(M^3)$ for an $M \times M$ matrix [10]. In this respect, the proposed method could be less demanding.

5. Simulation Example

This section gives a simulation example of TDE by the proposed method for a fixed delay at different SNR's.

Computer simulation was carried out to evaluate the performance of the proposed method. The source signal s_k consisted of a sinusoid and the input signal x_k was corrupted by white Gaussian noise v_k as given by

$$x_k = A \cos(2\pi f_0 k) + v_k \qquad (35)$$

where the frequency of the input signal, f_0, was set to 120 Hz per sample and the amplitude A was set to unity. The signal was sampled at 1 kHz. The corresponding output signal y_k was generated according to the relationship (2). In the simulation, the ratio of the input to output noise variance was assumed to be 1 for the So (SOLMS) method [15] while no assumptions were made for both the proposed method and the least mean square TDE (LMSTDE) method [12]. For all methods, a fixed step-size of 0.001 was used and 100 independent runs were averaged. Only integer delays were simulated because it was considered that non-integer delays can be obtained by use of a variety of interpolation techniques already well known. The filter orders for the proposed method were set to $N = M = 16$. The IIR filter order N_1 was set to 8. This filter order was sufficient for whitening the noisy single-sinusoid input signal. The estimate of $P_e(k)$ was found with $P_1 = 20$. The Q was set to 100 samples and l_m to 4. It was observed that the power P_n at time k used to update the Levinson-Trench algorithm sometimes become negative. To avoid the negative power, the following update strategy for the noise power was used in the simulation

$$P_n(k) = \begin{cases} P_e(k) - P_z(k), & \text{if } P_e(k) > P_z(k) \\ P_e(k), & \text{otherwise.} \end{cases}$$

Throughout the simulation, the SNR denotes the input SNR in decibels (dB) and is defined as $10 \log(1/2\sigma_v^2)$. The output SNR, defined by $10 \log(1/2\sigma_n^2)$, was set to 10 dB.

Figures 4 and 5 show that the delay convergence properties of the simulated methods. Figure 4 shows that at a high SNR all the methods converge to the true delay at about the same rate. However, at a low SNR the robustness of the proposed method can clearly be seen in Fig. 5 where the proposed method converges to the true delay of $2 \times T_s$ much faster than the other two methods. The choice of the input SNR's for illustration was based on the following consideration. Above 10 dB, most TDE methods perform well since the signal power is dominant. However, below 0 dB, when
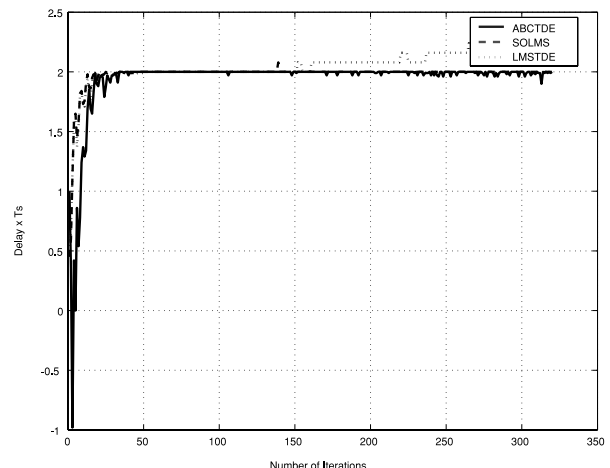
Fig. 4 Comparison of the proposed method with the SOLMS and LM-STDE methods for a time delay of $2 \times T_s$ at an SNR of 10 dB averaging 100 independent runs.
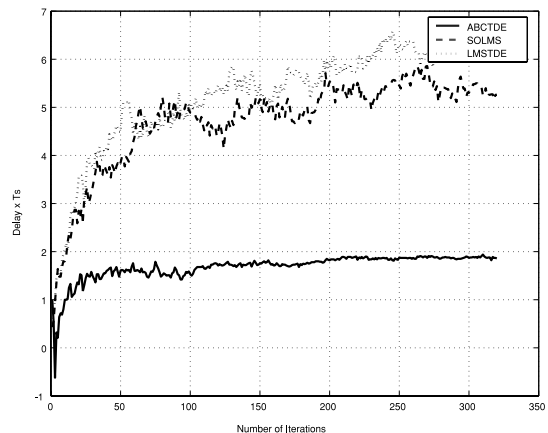

Fig. 5 Comparison of the proposed method with the SOLMS and LM-STDE methods for a time delay of $2 \times T_s$ at an SNR of 0 dB averaging 100 independent runs.

the noise power becomes dominant, most algorithms fail to perform well. Thus, these two input SNR points could provide a good performance indication. From Figs. 4 and 5, we can observe that the proposed method and the other two methods perform well at high input SNR as expected. However, at low SNR, the proposed method performs much better than the other methods.

In order to assess the effect of the statistical characteristics of the input-side noise process on the performance of the proposed method, colored additive Gaussian noise was used in (35). The colored noise was generated by a first-order autoregressive (AR) model. The output of the AR filter, v_k, was given by $v_k = -0.8v_{k-1} + u_k$, where u_k represents the driving white Gaussian noise process of unit variance. With this input noise model, the true delay was set to $2 \times T_s$ and the tracking ability of the proposed method was compared with that of the SOLMS and the LMSTDE methods. The simulation results are shown in Fig. 6. The colored noise tends to have the effect of slowing down the convergence
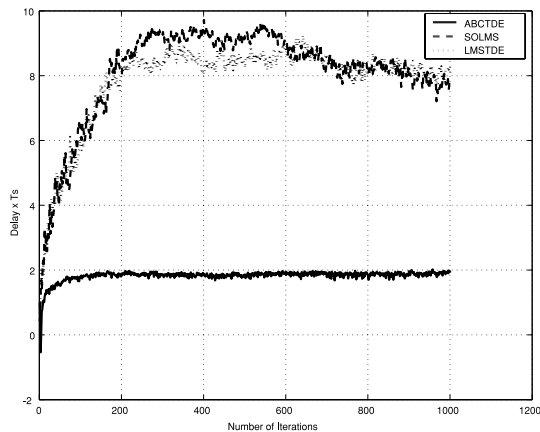
Fig. 6 Comparison of the proposed method with the SOLMS and LM-STDE methods for a time delay of $2 \times T_s$ when the input noise is a colored Gaussian process, averaging 100 independent runs.

speed of the proposed method. This could be because in an attempt to whiten the input process, the estimate of the input noise variance comes closer to that of u_k. The result is an increase in the bias of the channel estimator coefficients that reduces the rate at which the coefficients attain their true values. However, the convergence speed of the proposed method still appears to be faster than that of the other two methods. The LMSTDE method works well only for white input noise processes while the SOLMS method does not explicitly control the input noise.

6. Time-Variant Delay Tracking

This section describes how the ABCTDE method can be used to track a time variant delay.

6.1 Dual Mode Technique

Time-variant delay tracking can be considered as operating in two modes: the normal mode and the interrupt mode. The interrupt mode occurs when the time delay being tracked suddenly changes. The tracking of time delay proceeds as follows. At the $(k-1)$th iteration, three transitions are possible to the (k)th iteration for the channel impulse response vector. Assume that the channel impulse response vector at the $(k-1)$th iteration is

$$\mathbf{h}(k-1) = [0, \ldots, 0, h_D(k-1), 0, \ldots, 0]^T \tag{36}$$

where $h_D(k-1)$ is a non-zero value (ideally equal to one) occurring in the position of the vector corresponding to the time delay. The subscript D is equivalent to the optimal delay. Then, at the kth iteration the following vectors are possible in the event of a change in the time delay

$$_{-1}\mathbf{h}(k) = [0, \ldots, h_{D-1}(k), 0, 0, \ldots, 0]^T$$
$$_0\mathbf{h}(k) = [0, \ldots, 0, h_D(k), 0, \ldots, 0]^T$$
$$_{+1}\mathbf{h}(k) = [0, \ldots, 0, 0, h_{D+1}(k), \ldots, 0]^T. \tag{37}$$

The possible transitions are $\mathbf{h}(k-1) \rightarrow_0 \mathbf{h}(k)$, $\mathbf{h}(k-1) \rightarrow$

$_{-1}\mathbf{h}(k)$ and $\mathbf{h}(k-1) \rightarrow_{+1} \mathbf{h}(k)$. These transitions correspond to a decrease in time delay by one sample period, no change in the time delay, and an increase in the time delay by one sample period, respectively. In (37) we recognize that

$$E[h_D(k)] = E[h_{D-1}(k)] = E[h_{D+1}(k)] = 1.$$

Let $_T\mathbf{H}(k)$ be the matrix formed from the possible channel impulse response vectors after transition to iteration k defined by

$$_T\mathbf{H}(k) = [_{-1}\mathbf{h}(k), _0\mathbf{h}(k), _{+1}\mathbf{h}(k)]. \tag{38}$$

We can also define the matrix at the $(k-1)$th iteration as

$$\mathbf{H}(k-1) = [\mathbf{h}(k-1), \mathbf{h}(k-1), \mathbf{h}(k-1)] \tag{39}$$

which is an $M \times 3$ matrix with all its columns consisting of $\mathbf{h}(k-1)$. Using (38) and (39) we formulate a sum matrix at the kth iteration defined as

$$_s\mathbf{H}(k) =_T \mathbf{H}(k) + \mathbf{H}(k-1). \tag{40}$$

If the matrices $_T\mathbf{H}(k)$ and $\mathbf{H}(k-1)$ are sparse, then the matrix $_s\mathbf{H}(k)$ will also be sparse. Furthermore, if the largest elements of $_T\mathbf{H}(k)$ are very close in magnitude to those of $\mathbf{H}(k-1)$, the largest element of $_s\mathbf{H}(k)$ may be easily found. We can employ the matrix norm properties to get the largest element of $_s\mathbf{H}(k)$. The maximum absolute column sum norm of a matrix $\mathbf{C}^{m \times n}$ is defined from the L_1-norm as

$$\|\mathbf{C}\|_1 = \max_j \sum_{i=1}^{n} |c_{ij}| \tag{41}$$

where c_{ij} are the elements of $\mathbf{C}$. From (41), we can define the column number satisfying (41) as $\alpha(k)$ so that

$$\alpha(k) = \arg \ \|_s\mathbf{H}(k)\|_1. \tag{42}$$

From (40) we can see that it is possible to utilize the minimum matrix norm, if the matrix $\mathbf{H}(k-1)$ has its first and third column shifted by one row in opposite directions such that the maximum element of each of the vectors lies on the same diagonal. Referring to the resulting matrix as $\mathbf{H}'(k-1)$ and taking the difference instead of the sum in (40), the minimized argument corresponding to the delay can be found. By similarly defining the other matrices, the set of equations corresponding to (40)–(42) becomes

$$_s\mathbf{H}'(k) = _T\mathbf{H}(k) - \mathbf{H}'(k-1) \tag{43}$$

$$\min\|_s\mathbf{H}'(k)\|_1 = \min_j \sum_{i=1}^{n} |[_s\mathbf{H}'(k)]_{ij}| \tag{44}$$

$$\alpha'(k) = \arg[\min \ \|_s\mathbf{H}'(k)\|_1] \tag{45}$$

$$D(k) = \underset{i}{\mathrm{argmax}}[_s\mathbf{H}'_{\alpha'(k)}(k)]_i \tag{46}$$

where notation $[\mathbf{X}_i(k)]_j$ denotes the jth element of the ith column of matrix $\mathbf{X}(k)$. We recognize $_s\mathbf{H}'(k)$ as a matrix that is a function of the finite forward differences of the channel impulse response. The finite forward difference Δh_k is defined by $\Delta h_k = h_{k+1} - h_k$.

From (40)–(42) and (44)–(46), we also observe that if the time delay could be constant for a long period, continuous system delay optimization could be unnecessary. In this case the vector norms can be used instead of matrix norms by setting a threshold value for the equalizer error for which the delay optimization should be performed. It can be observed that the changes in the L_1-norm between $\mathbf{h}(k-2)$, $\mathbf{h}(k-1)$ and $\mathbf{h}(k)$ remain approximately constant unless the time delay changes. These changes can be captured by the ratio

$$\gamma(k) = \frac{\|\mathbf{h}(k)\|_1 - \|\mathbf{h}(k-1)\|_1}{\|\mathbf{h}(k-1)\|_1 - \|\mathbf{h}(k-2)\|_1}. \quad (47)$$

This ratio is approximately equivalent to one in the normal operation mode. By setting a threshold η as

$$\gamma(k) \geq \eta, \quad (48)$$

the delay optimization can be started every time. Equations (47) and (48) result in the condition

$$\|\mathbf{h}(k)\|_1 \geq (1 + \eta)\|\mathbf{h}(k-1)\|_1 - \eta\|\mathbf{h}(k-2)\|_1. \quad (49)$$

From (49) we have the condition for which the operation mode changes to the interrupt mode during the normal operation. As previously discussed, changes in ζ_k will be predominantly due to delay changes. A possible threshold could be determined by the ratio

$$\kappa(k) = \frac{|\zeta_k| - |\zeta_{k-1}|}{|\zeta_{k-1}| - |\zeta_{k-2}|} \geq \rho \quad (50)$$

for changing to the interrupt mode. Therefore, by monitoring changes in the magnitudes of both $\gamma(k)$ and ζ_k, changes in delay can be tracked.

In the tracking mode, the following set of equations are used to get the time $\hat{D}(k)$ as given by (46):

$$_T\hat{\mathbf{H}}(k) = [_{-1}\mathbf{g}(k), _0 \mathbf{g}(k), _{+1} \mathbf{g}(k)]$$
$$\hat{\mathbf{H}}(k-1) = [\mathbf{g}(k-1), \mathbf{g}(k-1), \mathbf{g}(k-1)]$$
$$_s\hat{\mathbf{H}}'(k) = {}_T\hat{\mathbf{H}}(k) - \hat{\mathbf{H}}'(k-1)$$
$$\hat{\alpha}'(k) = \arg[\min \|_s\hat{\mathbf{H}}'(k)\|_1]$$
$$\hat{D}(k) = \underset{i}{\operatorname{argmax}}[_s\mathbf{H}'_{\hat{\alpha}'(k)}(k)]_i. \quad (51)$$

The above set of equations is derived directly from (38), (39), (43), (44), and (46), respectively.

The parameter $\gamma(k)$ is estimated as

$$\hat{\gamma}(k) = \frac{\|\mathbf{g}(k)\|_1 - \|\mathbf{g}(k-1)\|_1}{\|\mathbf{g}(k-1)\|_1 - \|\mathbf{g}(k-2)\|_1}. \quad (52)$$

6.2 Simulation

Some simulation examples of time-variant delay tracking are shown. The settings in Sect. 5 were used in the simulation with the following additional requirements. The parameter η was set to 300 while ρ was set to 10. These two values were experimentally determined. The delay was suddenly
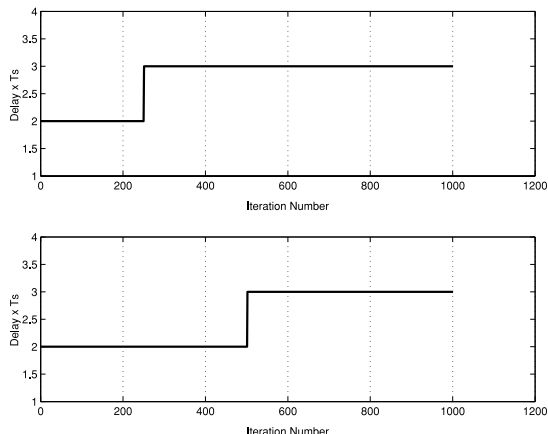

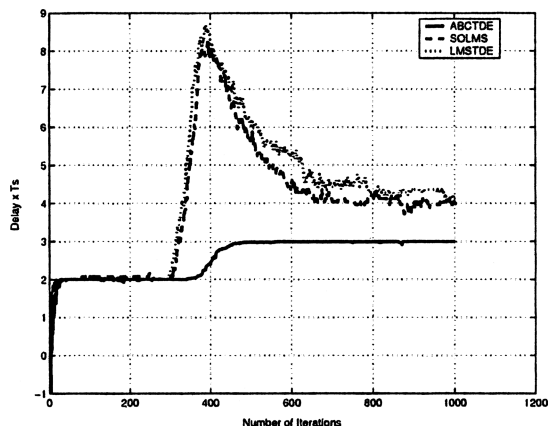**Fig. 7** The step delay schemes.

Fig. 8 Comparison of the proposed method with the SOLMS and LM-STDE methods for time-variant delay changing from $2 \times T_s$ to $3 \times T_s$ at the 250th iteration at an SNR of 10 dB, and averaging 100 independent runs.
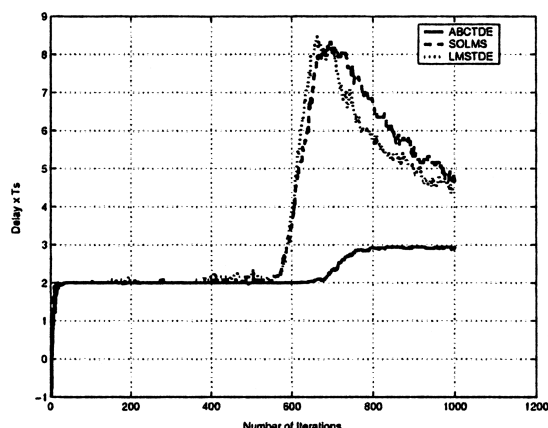

Fig. 9 Comparison of the proposed method with the SOLMS and LM-STDE methods for time-variant delay changing from $2 \times T_s$ to $3 \times T_s$ at the 500th iteration at an SNR of 10 dB, and averaging 100 independent runs.

changed at the 250th and 500th iterations, respectively, as shown in Fig. 7. Figures 8 and 9 show the performance of the proposed time variant delay estimation method compared to the other two methods. The results show that the
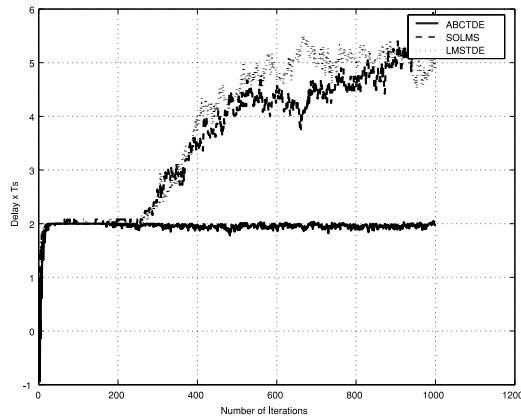
Fig. 10   Comparison of the proposed method with the SOLMS and LM-STDE methods for sudden change of noise variance from 10 dB to 0 dB at the 250th iteration with the delay held at $2 \times T_s$, and averaging 100 independent runs.
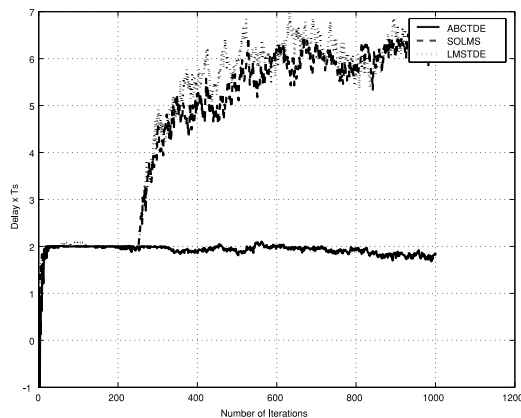

Fig. 11   Comparison of the proposed method with the SOLMS and LM-STDE methods for sudden change of noise variance from 10 dB to −5 dB at the 250th iteration with the delay held at $2 \times T_s$, and averaging 100 independent runs.

proposed method performs much better for sudden changes in delay by adapting to the new delay faster than the other methods. The bias in delay for the proposed method is evidently low at the 1000th iteration while the other two methods show considerable bias. This could be attributed to the ability of the proposed method to accurately and rapidly track changes in the channel impulse response and thus adapt to new delays faster. The rate at which the proposed method adapts to the new delay would depend on the η and ρ. Smaller values of η and ρ would be expected to give a more rapid convergence to the new delay. The slow change to the new delay by the proposed method in Figs. 8 and 9 is because during the period immediately after the change in the delay occurs, the equalizer output error and the change in the channel estimator coefficients are not big enough to make $\gamma(k)$ and $\kappa(k)$ initiate the optimization of the delay. If minor variations in the delay estimate at each iteration are taken as changes in the delay to be estimated, then delay optimization is necessary more often. In the simulation, we

determined the values of η and ρ that make the delay tracking unaffected by the iteration-to-iteration variance in the delay estimate.

We also conducted an investigation on the tracking ability of the proposed method in time-variant noise environments. The noise variance was changed from 10 dB to 0 dB at the 250th iteration. The results as depicted in Fig. 10 show a superior tracking ability for the proposed method. It is also shown in Fig. 11 that the proposed method still performs better than the other methods even for sub-zero SNR's, where in this case the input SNR changes from 10 dB to −5 dB at the 250th iteration.

7.   Conclusion

We have proposed a new ABCTDE method. The method is based on noisy input-output system identification modeling of TDE using the ABC equalizer. The attractiveness of the ABC approach is rapid convergence, reduced computational complexity and noise robustness when compared to the other competing methods. A dual mode operation approach has been proposed for step time-variant delay tracking using L_1-normed finite forward differences. Simulation results show that the proposed method converges to the true delay faster than the other methods with no constraints on the input and output noise variances. The proposed method's superiority is shown at low SNR's. Since the two spatially separated sensors may have different channel responses in addition to delay, it would be proper to consider this case in our future work. Future work also will aim to investigate the performance of the proposed method for multipath delays and how further reductions in computational complexity can be achieved.

Acknowledgment

References

[1] P. Butler and A. Cantoni, "Noniterative automatic equalization," IEEE Trans. Commun., vol.COM-23, no.6, pp.621–633, June 1975.
[2] C.H. Knapp and G.C. Carter, "The generalized correlation method for estimation of time delay," IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-24, no.4, pp.320–327, Aug. 1976.
[3] G.C. Carter, "Time delay estimation for passive sonar signal processing," IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-29, no.3, pp.463–470, June 1981.
[4] M.-A. Pallas and G. Jourdain, "Active high resolution time delay estimation for large BT signals," IEEE Trans. Signal Process., vol.39, no.4, pp.781–788, April 1991.
[5] Y.T. Chan, J.M. Riley, and J.B. Plant, "Modeling time delay and its application to nonstationary delays," IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-29, no.4, pp.577–581, Aug. 1981.
[6] Z. Cheng and T.T. Tjhung, "A new time delay estimator based on ETDE," IEEE Trans. Signal Process., vol.51, no.7, pp.1859–1869, July 2003.

[7] P.L. Feintuch, N.J. Bershad, and F.A. Reed, "Time delay estimation using LMS algorithm-dynamic behavior," IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-29, no.3, pp.571–576, June 1981.

[8] M.J. Hinich and G.R. Wilson, "Time delay estimation using the cross bispectrum," IEEE Trans. Signal Process., vol.40, no.1, pp.106–113, Jan. 1992.

[9] J. Li, "An efficient algorithm for time delay estimation," IEEE Trans. Signal Process., vol.46, no.8, pp.2231–2235, Aug. 1998.

[10] G.H. Golub and C.F. Van Loan, Matrix Computations, Johns Hopkins University Press, Baltimore and London, 1989.

[11] S.M. Kay, Modern Spectral Estimation: Theory and Application, Prentice Hall, Englewood Cliffs, NJ, 1988.

[12] F.A. Reed, P.L. Feintuch, and N.J. Bershad, "Time delay estimation using LMS algorithm-static behavior," IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-29, no.3, pp.561–571, June 1981.

[13] T. Shimamura and J. Suzuki, "Adaptalisation of Butler-Cantoni method for fast start-up equalization," IEICE Trans. Fundamentals (Japanese Edition), vol.J81-A, no.4, pp.622–630, April 1998.

[14] J. Gamba, Y. Tsuda, and T. Shimamura, "An adaptive Butler-Cantoni based time delay estimation (ABCTDE) method—IIR whitening filtering approach," Proc. IEEE International Symposium on Circuits and Systems, vol.3, pp.265–268, Vancouver, Canada, 2004.

[15] H.C. So, "Noisy input-output system identification approach for time delay estimation," Signal Process., vol.82, no.10, pp.1471–1475, Oct. 2002.

[16] H.C. So, "Time delay estimation for sinusoidal signals," IEE Proc., Radar Sonar Navig., vol.148, no.6, pp.318–324, Dec. 2001.

[17] K.V. Fernando and H. Nicholson, "Identification of linear systems with input and output noise: The Koopmans-Levin method," IEE Proc. D, vol.132, no.1, pp.31–36, Jan. 1985.

[18] T. Shimamura, S. Semnani, and C.F.N. Cowan, "Equalization of time-variant communications channels via channel estimation based approaches," Signal Process., vol.60, no.2, pp.181–193, July 1997.

[19] D.V.B. Rao and S.Y. Kung, "Adaptive notch filtering for the retrieval of sinusoids in noise," IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-32, no.4, pp.791–802, Aug. 1984.

[20] J.G. Proakis and D. Manolakis, Digital Signal Processing: Principles, Algorithms and Applications, 3rd ed., Prentice Hall, 1995.

[21] J.G. Proakis, Digital Communications, 4th ed., McGraw-Hill, 2001.

[22] D.R. Morgan, "Steady-state response of a delay-constrained adaptive linear predictor filter to a sinusoid in white noise," IEEE Trans. Acoust. Speech Signal Process., vol.ASSP-31, no.4, pp.1039–1043, Aug. 1983.

[23] J. Makhoul, "Linear prediction: A tutorial review," Proc. IEEE, vol.63, pp.561–580, April 1975.

Tetsuya Shimamura received the B.E., M.E., and Ph.D. degrees in electrical engineering from Keio University, Yokohama, Japan, in 1986, 1988, and 1991, respectively. In 1991, he joined Saitama University, Saitama City, Japan, where he is currently an Associate Professor. His interests are in digital signal processing and applications to speech and communication systems. He is a member of IEEE, EURASIP, and the Acoustical Society of America.

Jonah Gamba received BSc. (Hons) degree in electrical and electronics engineering from the University of Zimbabwe, Harare, Zimbabwe, in 1994 and an MSc. degree in computer science and engineering from Zhejiang University, Hangzhou, China, in 2000. He is currently with Saitama University, Saitama City, Japan, pursuing a doctor degree course in mathematical information systems. His present research interests include spectrum estimation, time delay estimation and adaptive channel equalization.