

# New Adaptive Algorithm for Delay Estimation of Sinusoidal Signals

Mrityunjy Chakraborty, *Senior Member, IEEE*, H. C. So, *Senior Member, IEEE*, and Jun Zheng

**Abstract**—In this letter, we address the problem of adaptively estimating the time delay of a noisy sinusoid received at two spatially separated sensors. By choosing the sampling frequency equal to four times the signal frequency, a simple adaptive algorithm for direct delay estimation is derived. Algorithm convergence in mean and mean square error is proved. Computer simulations are also included to demonstrate the effectiveness of the proposed method.

**Index Terms**—Adaptive filters, sinusoidal signals, time delay estimation.

## I. INTRODUCTION

ACCURATE time delay estimation between two or more noisy versions of the same signal received at spatially separated sensors [1] is an important topic that finds applications in positioning and tracking, speed sensing, direction finding, biomedicine, exploration geophysics, etc. Over the years, many methods have been proposed to tackle this problem effectively. For stationary time delay, the generalized cross correlator [2] is a classical estimator where the delay estimate is found by locating the peak of the cross correlation function of the filtered versions of the observed data. Under Gaussian signal and noise assumption, it can provide maximum likelihood estimation performance. On the other hand, for sinusoidal signals which commonly occur in radar, sonar, and digital communications, the quadrature delay estimator (QDE) utilizing all in-phase and quadrature-phase components of the received signals, as well as the discrete-time Fourier transform-based method [3], provide viable solutions to the delay estimation problem.

When the time delay is time-varying due to relative source/receiver motion, adaptive tracking of it is necessary. In [4], a finite impulse response (FIR) filter is used to model the time delay and its estimate is computed from interpolating filter coefficients. Alternatively, explicit delay adjustment [5]–[8] can be achieved by constraining the filter coefficients to be some functions of the time delay. A notable recent development in this respect is the adaptive QDE [8] which can be viewed as an online realization of [3].

In this letter, we present a novel adaptive filter algorithm for estimating the delay in a sinusoidal signal received at two sensors. The development is based on an appropriate sampling rate which generates a two-tap FIR filter model for the delayed

signal and also results in a diagonal autocorrelation matrix for the  $2 \times 1$  filter input vector. The delay is estimated and tracked by identifying the filter coefficients which are given as sinusoidal functions of the delay. For this, a novel adaptive algorithm is developed that updates the delay estimate explicitly. The delay update is shown to converge to the true delay value in mean, for which necessary convergence condition is established. A detailed stability analysis is also carried out where conditions for keeping the steady-state delay estimation error variance bounded are worked out. The proposed algorithm is computationally simpler than the standard least mean square (LMS) algorithm [9] as well as other existing delay estimation algorithms and, in particular, offers an elegant implementation using CORDIC processors [10], [11]. MATLAB-based simulation results also suggest that the estimation and tracking performance of the proposed algorithm is at par with both the standard LMS algorithm [9] and the adaptive QDE [8].

## II. ADAPTIVE DELAY ESTIMATION

### A. Proposed Algorithm

Consider the following model for the signals received at the two sensors, namely,  $x_a(t)$  and  $y_a(t)$ :

$$x_a(t) = s_a(t) + u_a(t) \quad (1)$$

$$y_a(t) = s_a(t - D) + z_a(t) \quad (2)$$

where  $s_a(t) = \cos(\Omega t + \phi)$  is a sinusoid with known analog frequency  $\Omega$  and random phase  $\phi$  that is uniformly distributed over  $[0, 2\pi)$ . The two terms  $u_a(t)$  and  $z_a(t)$  represent two zero mean, additive white Gaussian noise processes independent of each other and also of  $\phi$  and thus of  $s_a(t)$ . The variable  $D$  is the delay between the received copies of the signal  $s_a(t)$  at the two sensors, which is unknown and is to be estimated and tracked. It is also assumed that the net phase shift  $\Omega D$  due to the delay lies within  $[0, 2\pi)$  which eliminates possibilities of ambiguity over  $D$ .

The signals  $x_a(t)$  and  $y_a(t)$  are digitized with a sampling period  $\tau$  to generate the sequences  $x(n) \equiv x_a(n\tau) = \cos(\Omega n\tau + \phi) + u_a(n\tau) \equiv s(n) + u(n)$  and  $y(n) \equiv y_a(n\tau) = \cos(\Omega(n\tau - D) + \phi) + z_a(n\tau) = \cos(\Omega n\tau + \phi) \cos(\Omega D) + \sin(\Omega n\tau + \phi) \sin(\Omega D) + z(n)$ , where  $s(n) \equiv s_a(n\tau) = \cos(\Omega n\tau + \phi)$ ,  $u(n) \equiv u_a(n\tau)$ , and  $z(n) \equiv z_a(n\tau)$ . Our key idea is to select a specific sampling period  $\tau$  satisfying  $\Omega\tau = \pi/2$ , or equivalently,  $\Omega_s = 4\Omega$ , where  $\Omega_s = 2\pi/\tau$  is the sampling frequency. It is then possible to write  $y(n)$  as

$$y(n) = \cos(\Omega D) s(n) + \sin(\Omega D) s(n - 1) + z(n). \quad (3)$$

To estimate and track the delay, one can approach (3) as a system identification problem with noisy input and use a standard LMS-based two-tap adaptive filter with coefficients, say,  $w_0(n)$  and

Manuscript received July 2, 2007; revised July 30, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Cihan Tepedelenlioglu.

M. Chakraborty is with the Department of Electronics and Electrical Communication Engineering, Indian Institute of Technology, Kharagpur 721302, India (e-mail: mrityun@ece.iitkgp.ernet.in).

H. C. So and J. Zheng are with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong (e-mail: mrityun@ece.iitkgp.ernet.in; hcso@ee.cityu.edu.hk; junzheng@cityu.edu.hk).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/LSP.2007.908003

$w_1(n)$ . This method, however, would require additional computations in the form of  $\tan^{-1}(w_1(n)/w_0(n))$  to obtain the delay estimate at each index. Further, in the absence of a constraint like  $w_0^2(n) + w_1^2(n) = 1$ , this also may not produce sufficiently accurate delay estimates, especially under rapidly fluctuating delay condition, or for very short data records. In this context, a simpler and also more appropriate approach, in our opinion, would be to use the *a priori* knowledge of the specific forms of the system coefficients, viz.,  $\cos(\Omega D)$  and  $\sin(\Omega D)$  and precondition the adaptive filter as  $\mathbf{w}(n) = [\cos(\Omega D(n)), \sin(\Omega D(n))]^T$ , with  $T$  denoting transposition. The filter weights are then adjusted by directly updating the delay estimate  $D(n)$  in an LMS like manner, as given by

$$\begin{aligned} D(n+1) &= D(n) - \mu \frac{\partial e^2(n)}{\partial D(n)} \\ &= D(n) - 2\mu e(n) \frac{\partial e(n)}{\partial D(n)}. \end{aligned} \quad (4)$$

The constant  $\mu$  in (4) is the step size parameter and the error signal  $e(n)$  is given as  $e(n) = y(n) - \mathbf{w}^T(n)\mathbf{x}(n)$ , with  $\mathbf{x}(n) = [x(n), x(n-1)]^T$ . It is easy to verify that  $\partial e(n)/\partial D(n) = \Omega \mathbf{x}^T(n) [\sin(\Omega D(n)), -\cos(\Omega D(n))]^T$ . This results in the following delay update equation:

$$D(n+1) = D(n) + 2\mu \Omega e(n) \mathbf{x}^T(n) \begin{bmatrix} -\sin(\Omega D(n)) \\ \cos(\Omega D(n)) \end{bmatrix}. \quad (5)$$

Note that the two operations, namely, filtering as given by  $\mathbf{w}^T(n)\mathbf{x}(n)$  and delay updating as given by (5), require generation of signal pairs of the following form:  $x \cos \theta, x \sin \theta$ , where  $x$  is either  $x(n)$  or  $x(n-1)$  and  $\theta = \Omega D(n)$  at index  $n$ . Such signal pairs can be generated very efficiently by CORDIC processors [10] by means of a sequence of additions and wired shift operations. This would reduce the number of multiplications and divisions from five as in the LMS algorithm to just two. Furthermore, no additional overhead will be required to evaluate trigonometric quantities.

In the following, we show that the delay update (5) converges in mean, i.e.,  $\lim_{n \rightarrow \infty} E[D(n) - D] = 0$ , where  $E$  denotes the expectation operator, provided that  $\mu$  is chosen to satisfy  $0 < \mu < 2/\Omega^2$ .

### B. Convergence Analysis

First recall that the phase  $\phi$  is uniformly distributed over  $[0, 2\pi)$ , meaning the signal autocorrelation is given as

$$\begin{aligned} r_{ss}(k) &= E[s(n)s(n-k)] \\ &= \frac{1}{2\pi} \int_0^{2\pi} \cos(\Omega n\tau + \phi) \cos(\Omega(n-k)\tau + \phi) d\phi \\ &= \frac{1}{2} \cos\left(\frac{k\pi}{2}\right) \end{aligned}$$

where  $k$  is any integer. From this and also from the fact that  $u(n)$  is a zero mean, white process independent of  $s(n)$ , it then follows that the  $2 \times 2$  input autocorrelation matrix is given as  $\mathbf{R}_{xx} = E[\mathbf{x}(n)\mathbf{x}^T(n)] = (1/2 + \sigma_u^2) \mathbf{I}$ , where  $\sigma_u^2 = E[u^2(n)]$  and  $\mathbf{I}$  denotes the  $2 \times 2$  identity matrix. Next, define the instantaneous delay error as  $\Delta(n) = D(n) - D$ . Then, using the definition  $\mathbf{w}'(n) = [-\sin(\Omega D(n)), \cos(\Omega D(n))]^T$ , from (5), we can write

$$E[\Delta(n+1)] = E[\Delta(n)] + 2\mu \Omega E[e(n)\mathbf{x}^T(n)\mathbf{w}'(n)]. \quad (6)$$

Note that the two vectors,  $\mathbf{w}(n)$  and  $\mathbf{w}'(n)$ , are mutually orthogonal at each index  $n$ , i.e.,  $E[\mathbf{w}^T(n)\mathbf{w}'(n)] = 0$ . We now invoke the ‘‘independence assumption’’ as is common with the analysis of the LMS algorithm [9] and assume  $D(n)$  to be statistically independent both of  $s(n)$ , or, equivalently, of  $\phi$  and thus of  $\mathbf{s}(n) = [s(n), s(n-1)]^T$ , and of  $\mathbf{u}(n) = [u(n), u(n-1)]^T$ . Together, this means  $D(n)$  is statistically independent of  $\mathbf{x}(n)$ . Replacing  $e(n)$  in (6) by  $y(n) - \mathbf{w}^T(n)\mathbf{x}(n)$ , using the orthogonality between  $\mathbf{w}(n)$  and  $\mathbf{w}'(n)$  and the fact that  $\mathbf{R}_{xx} = (1/2 + \sigma_u^2) \mathbf{I}$ , we then first observe

$$\begin{aligned} &E[\mathbf{w}^T(n)\mathbf{x}(n)\mathbf{x}^T(n)\mathbf{w}'(n)] \\ &= E[\mathbf{w}^T(n)E\{\mathbf{x}(n)\mathbf{x}^T(n)\}\mathbf{w}'(n)] \\ &= \left(\frac{1}{2} + \sigma_u^2\right) E[\mathbf{w}^T(n)\mathbf{w}'(n)] = 0. \end{aligned} \quad (7)$$

In (6), on the RHS, we then have only to evaluate  $E[y(n)\mathbf{x}^T(n)\mathbf{w}'(n)]$ . For this, we first replace  $y(n)$  by  $[\cos(\Omega D), \sin(\Omega D)]\mathbf{s}(n) + z(n)$ . Next, we observe that since  $D(n)$  depends only on the past samples of  $z(n)$  and  $z(n)$  is a zero mean, white Gaussian process (i.e., samples of  $z(n)$  are i.i.d.) independent of  $x(n)$ ,  $E[z(n)\mathbf{x}^T(n)\mathbf{w}'(n)] = E[z(n)]E[\mathbf{x}^T(n)\mathbf{w}'(n)] = 0$ . Thus

$$\begin{aligned} &E[y(n)\mathbf{x}^T(n)\mathbf{w}'(n)] \\ &= E[[\cos(\Omega D), \sin(\Omega D)]E\{\mathbf{s}(n)\mathbf{x}^T(n)\}\mathbf{w}'(n)] \\ &= -\frac{1}{2} E[\sin(\Omega \Delta(n))] \end{aligned} \quad (8)$$

since  $s(n)$  and  $u(n)$  are mutually independent, zero mean processes and  $E[\mathbf{s}(n)\mathbf{s}^T(n)] = (1/2)\mathbf{I}$ . When  $D(n)$  is sufficiently close to  $D$ ,  $\Omega \Delta(n)$  is small and we can approximate  $\sin(\Omega \Delta(n))$  by  $\Omega \Delta(n)$ . Combining (6)–(8), we then have

$$E[\Delta(n+1)] = (1 - \mu \Omega^2) E[\Delta(n)]. \quad (9)$$

Clearly,  $\lim_{n \rightarrow \infty} E[\Delta(n)] = 0$  if  $0 < \mu < 2/\Omega^2$ .

Since  $D(n)$  converges to  $D$  in mean, it is essential to ensure that the steady-state value of  $\epsilon(n) = E[\Delta^2(n)]$  remains bounded. It is shown in the Appendix that this requires  $\mu$  to satisfy  $0 < \mu < 4/(3 + 4\sigma_u^2)\Omega^2$  for which

$$\lim_{n \rightarrow \infty} \epsilon(n) = \frac{\mu(\sigma_u^2 + \sigma_z^2)}{1 - (\sigma_u^2 + \frac{3}{4})\mu\Omega^2} \quad (10)$$

with  $\sigma_z^2 = E[z^2(n)]$ .

### III. SIMULATION RESULTS

Computer simulations were carried out to evaluate the delay estimation performance of the proposed method vis-a-vis the adaptive QDE [8] and the LMS algorithm [9]. For this, a unity amplitude sinusoidal signal of frequency  $\Omega = 2\pi \times 10^3$  rad s<sup>-1</sup> was used which was sampled with a sampling period of  $\tau = 2.5 \times 10^{-4}$  s. The powers of the two noise processes  $u(n)$  and  $z(n)$  were assigned the same value that results in the same signal-to-noise ratio (SNR) of 10 dB for both the received signals  $x(n)$  and  $y(n)$ . For the adaptive QDE, a fifth-order Lagrange fractional delay filter was used together with filter coefficient modulation because of the relatively low sampling rate. In the case of the LMS algorithm, a two-tap filter was employed and the corresponding delay estimate was obtained by computing arc tangent on the ratio of the filter coefficients. The results provided are based on averages of 2000 independent runs.

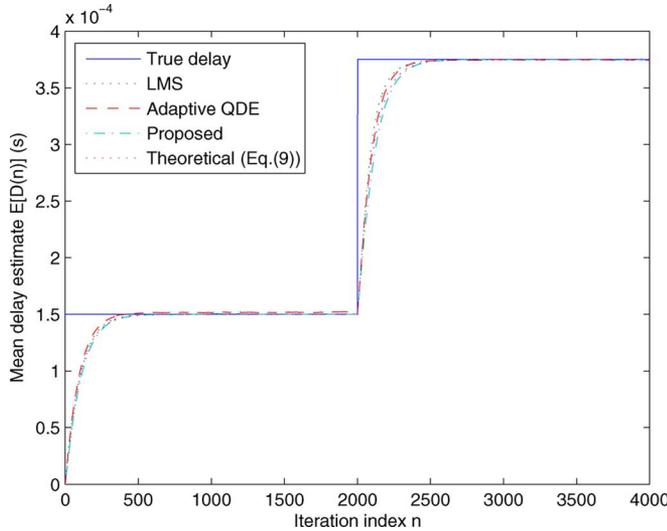


Fig. 1. Mean delay estimates for a step change in  $D$ .

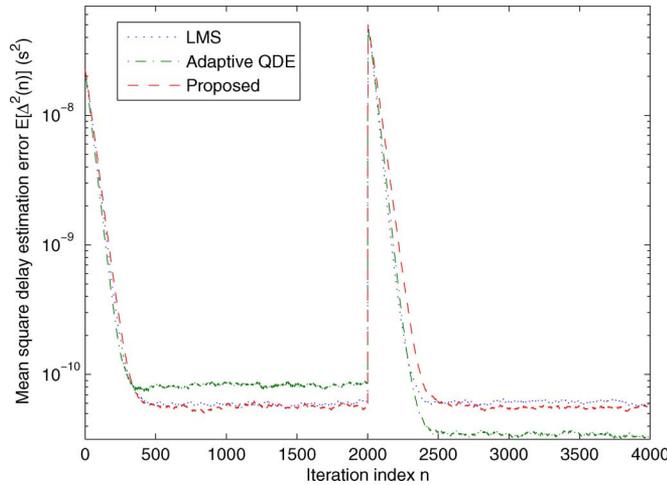


Fig. 2. Mean square delay estimation errors for a step change in  $D$ .

Fig. 1 shows the trajectories for the mean delay estimates (i.e.,  $E[D(n)]$ ) of the three methods for a step change in  $D$  which was held constant at  $1.5 \times 10^{-4}$  s during the first 2000 iterations and changed instantaneously to  $3.75 \times 10^{-4}$  s afterwards. The step size parameters for the proposed estimator, the adaptive QDE, and the LMS algorithm were chosen as  $2.5 \times 10^{-10}$ , 0.013, and 0.007, respectively, which ensures the same convergence speed for all the three schemes. It is seen from Fig. 1 that in the mean sense, all three methods accurately estimate and track the piecewise constant time delay. It is also easily seen that the simulated plot for  $E[D(n)]$  for the proposed method conforms very well to the dynamics described by (9). Next, we measured the corresponding mean square delay estimation errors and the results are plotted in Fig. 2. It is seen that while the mean square errors of the proposed scheme and the LMS algorithm are comparable, that of the adaptive QDE is lesser when  $D = 1.5 \times 10^{-4}$  s and higher when  $D = 3.75 \times 10^{-4}$  s, implying that the estimation error variance of the latter is delay dependent. Finally, we have also measured the steady-state values of  $\epsilon(n)$  at  $D = 1.5 \times 10^{-4}$  s and  $D = 3.75 \times 10^{-4}$  s and found them to be  $5.60 \times 10^{-11}$  s<sup>2</sup> and  $5.61 \times 10^{-11}$  s<sup>2</sup>, respectively, which also conforms closely to the theoretical value given by (10), namely,  $5.04 \times 10^{-11}$  s<sup>2</sup>.

## IV. CONCLUSION

A new adaptive algorithm for direct estimation of the time delay between two noisy versions of a sinusoidal signal received at spatially separated sensors has been developed. The algorithm uses a specific sampling frequency that results in a two-tap FIR filter model for the delay process which is identified by updating the delay estimate directly. Trajectories for the mean delay estimate and mean square delay error are also derived. MATLAB simulations showed comparable estimation performance of the proposed method, both in mean and mean square, with the standard LMS algorithm and the adaptive QDE. The proposed algorithm is computationally simpler than both the LMS and the adaptive QDE schemes, though the latter can operate at any sampling frequency at the expense of some additional computations like Hilbert transformation.

## APPENDIX

### CONVERGENCE OF DELAY ERROR VARIANCE

First observe from (5) that

$$\epsilon(n+1) = \epsilon(n) + 4\mu\Omega E[\Delta(n) e(n) \mathbf{x}^T(n) \mathbf{w}'(n)] + 4\mu^2\Omega^2 E[e(n) \mathbf{x}^T(n) \mathbf{w}'(n)]^2. \quad (\text{A1})$$

Using the independence between  $\Delta(n)$  and  $\mathbf{x}(n)$ , combining (7) and (8), and once again using the approximation  $\sin(\Omega\Delta(n)) \approx \Omega\Delta(n)$ , we can write

$$E[\Delta(n) e(n) \mathbf{x}^T(n) \mathbf{w}'(n)] = -\frac{1}{2}\Omega\epsilon(n). \quad (\text{A2})$$

To evaluate the third term on the RHS of (A1), we observe that  $s(n) = \cos(n\pi/2 + \phi)$  and  $s(n-1) = \sin(n\pi/2 + \phi)$ . Defining  $\gamma(n) = \mathbf{w}^T(n) \mathbf{x}(n)$ , we can then write

$$\gamma(n) = \sin\left(\frac{n\pi}{2} + \phi - \Omega D(n)\right) + \alpha(n)$$

where  $\alpha(n) = \mathbf{w}^T(n) \mathbf{u}(n)$ . Similarly, defining  $\beta(n) = \mathbf{w}^T(n) \mathbf{u}(n)$

$$\mathbf{w}^T(n) \mathbf{x}(n) = \cos\left(\frac{n\pi}{2} + \phi - \Omega D(n)\right) + \beta(n)$$

$$y(n) = \cos\left(\frac{n\pi}{2} + \phi - \Omega D\right) + z(n)$$

and thus

$$\begin{aligned} \epsilon(n) &= z(n) - 2 \sin\left(\frac{n\pi}{2} + \phi - \frac{\Omega(D + D(n))}{2}\right) \\ &\quad \times \sin\left(\frac{\Omega\Delta(n)}{2}\right) - \beta(n) \\ &\approx z(n) - \Omega\Delta(n) \sin\left(\frac{n\pi}{2} + \phi - \frac{\Omega(D + D(n))}{2}\right) - \beta(n). \end{aligned} \quad (\text{A3})$$

We can then write

$$\begin{aligned} E[e(n) \mathbf{x}^T(n) \mathbf{w}'(n)]^2 &= E[e(n) \gamma(n)]^2 \\ &= E\left[\left(z(n) - \Omega\Delta(n) \sin\left(\frac{n\pi}{2} + \phi - \frac{\Omega(D + D(n))}{2}\right) - \beta(n)\right)^2 \gamma^2(n)\right]. \end{aligned} \quad (\text{A4})$$

Note that in (A4), the cross terms involving  $z(n)$  take zero value since  $z(n)$  is a zero mean, white Gaussian process independent of  $\phi$  and  $u(n)$ , and also of  $D(n)$  as explained in Section II-B. The remaining terms in (A4) are evaluated separately each as follows.

A)  $E[z^2(n)\gamma^2(n)]$ : Since  $z(n)$  is independent of  $\phi$ ,  $u(n)$ , and  $D(n)$ ,  $E[z^2(n)\gamma^2(n)] = \sigma_z^2 E[\gamma^2(n)]$ . To evaluate  $E[\gamma^2(n)]$ , we note that 1)  $E[\sin^2(n\pi/2 + \phi - \Omega D(n))]$  = 1/2, since  $\phi$  is uniformly distributed over  $[0, 2\pi)$  and  $D(n)$  is independent of  $\phi$  because of the ‘‘independence assumption’’; 2)  $E[2\sin(n\pi/2 + \phi - \Omega D(n))\mathbf{w}^T(n)\mathbf{u}(n)] = 0$ , since  $\mathbf{u}(n)$  is a zero mean vector independent of  $\phi$  and  $D(n)$ ; and, lastly, 3)  $E[\mathbf{w}^T(n)\mathbf{u}(n)\mathbf{u}^T(n)\mathbf{w}'(n)] = E[\mathbf{w}^T(n)E[\mathbf{u}(n)\mathbf{u}^T(n)]\mathbf{w}'(n)] = \sigma_u^2$ . Thus,  $E[z^2(n)\gamma^2(n)] = (1/2 + \sigma_u^2)\sigma_z^2$ . Neglecting the fourth-order term  $\sigma_u^2\sigma_z^2$  which is very small for high SNRs, we have  $E[z^2(n)\gamma^2(n)] \approx \sigma_z^2/2$ .

B)  $E[\beta^2(n)\gamma^2(n)]$ : Substituting  $\gamma(n) = \sin(n\pi/2 + \phi - \Omega D(n)) + \alpha(n)$ , and once again using the fact that  $\mathbf{u}(n)$  is a zero mean vector independent of  $\phi$  and  $D(n)$ , we observe that 1)  $E[\sin^2(\frac{n\pi}{2} + \phi - \Omega D(n))\beta^2(n)] = E[\sin^2(\frac{n\pi}{2} + \phi - \Omega D(n))\mathbf{w}^T(n)E\{\mathbf{u}(n)\mathbf{u}^T(n)\}\mathbf{w}(n)] = \frac{\sigma_u^2}{2}$ , and 2) any cross term involving first power of  $\sin(n\pi/2 + \phi - \Omega D(n))$  results in zero after the expectation operation w.r.t.  $\phi$ . The third term,  $E[\alpha^2(n)\beta^2(n)]$ , on simplification, results in  $E\left[\left(-\frac{1}{2}\sin(2\Omega D(n))u^2(n) + \frac{1}{2}\sin(2\Omega D(n))u^2(n-1) + \cos(2\Omega D(n))u(n)u(n-1)\right)^2\right]$ .

Since  $u(n)$  is a zero mean white Gaussian process, we have  $E[u^3(n)] = 0$  and  $E[u^4(n)] = 3\sigma_u^4$ . From this and using the independence of  $\mathbf{u}(n)$  with  $D(n)$ , we observe that  $E[\alpha^2(n)\beta^2(n)]$  consists only of terms involving  $\sigma_u^4$ , which is again negligibly small for high SNR conditions. Thus,  $E[\beta^2(n)\gamma^2(n)] \approx \sigma_u^2/2$ .

C)  $E[\Omega^2\Delta^2(n)\sin^2(n\pi/2 + \phi - \Omega(D + D(n))/2)\gamma^2(n)]$ : Substituting  $\gamma(n) = \sin(n\pi/2 + \phi - \Omega D(n)) + \alpha(n)$ , we observe that this has three terms, namely, 1) a cross term involving first power of  $\alpha(n)$  which becomes zero after expectation since  $\mathbf{u}(n)$  is zero mean and is independent of  $\phi$  and  $D(n)$ , 2)  $E\left[\Omega^2\Delta^2(n)\sin^2\left(\frac{n\pi}{2} + \phi - \frac{\Omega(D+D(n))}{2}\right)\cdot\mathbf{w}^T(n)E[\mathbf{u}(n)\mathbf{u}^T(n)]\mathbf{w}'(n)\right] = \Omega^2\sigma_u^2 E\left[\frac{\Delta^2(n)}{2}\{1 - \cos(n\pi + 2\phi - \Omega(D(n) + D))\}\right] = \frac{1}{2}\Omega^2\sigma_u^2\epsilon(n)$ , and 3)  $E\left[\Omega^2\Delta^2(n)\sin^2\left(\frac{n\pi}{2} + \phi - \frac{\Omega(D+D(n))}{2}\right)\cdot\sin^2\left(\frac{n\pi}{2} + \phi - \Omega D(n)\right)\right] = \frac{3}{8}\Omega^2\epsilon(n)$ , where we use the approximation  $\cos(\Omega\Delta(n)) \approx 1$  for small values of  $\Omega\Delta(n)$ . Combining  $E\left[\Omega^2\Delta^2(n)\sin^2\left(\frac{n\pi}{2} + \phi - \frac{\Omega(D+D(n))}{2}\right)\gamma^2(n)\right] = \left(\frac{1}{2}\sigma_u^2 + \frac{3}{8}\right)\Omega^2\epsilon(n)$ .

D)  $E[2\Omega\Delta(n)\sin(n\pi/2 + \phi - \Omega(D + D(n))/2)\beta(n)\gamma^2(n)]$ : Again substituting  $\gamma(n) = \sin(n\pi/2 + \phi - \Omega D(n)) + \alpha(n)$ , we observe that this has the following three terms:

1) A cross term involving first power of  $\beta(n)$  but free of  $\alpha(n)$  which is zero for reasons explained above,

2)  $E\left[4\Omega\Delta(n)\sin\left(\frac{n\pi}{2} + \phi - \frac{\Omega(D+D(n))}{2}\right)\cdot\sin\left(\frac{n\pi}{2} + \phi - \Omega D(n)\right)\mathbf{w}^T(n)E[\mathbf{u}(n)\mathbf{u}^T(n)]\mathbf{w}'(n)\right] = 0$

(since  $\mathbf{w}(n)$  and  $\mathbf{w}'(n)$  are mutually orthogonal), and

3)  $E\left[2\Omega\Delta(n)\sin\left(\frac{n\pi}{2} + \phi - \frac{\Omega(D+D(n))}{2}\right)\beta(n)\alpha^2(n)\right] = 0$  (since  $\phi$  is independent of  $D(n)$  and  $u(n)$ ). Thus,  $E[2\Omega\Delta(n)\sin(n\pi/2 + \phi - \Omega(D + D(n))/2)\beta(n)\gamma^2(n)] = 0$ .

Combining the results of (A2) and  $E[e(n)\mathbf{x}^T(n)\mathbf{w}'(n)]^2 = E[e(n)\gamma(n)]^2$  in (A1), we then obtain

$$\epsilon(n+1) = a\epsilon(n) + b \quad (\text{A5})$$

where  $a = [1 - 2\mu\Omega^2 + 2\mu^2\Omega^4\sigma_u^2 + (3/2)\mu^2\Omega^4]$  and  $b = 2\mu^2\Omega^2(\sigma_u^2 + \sigma_z^2)$ . Equation (A5) is a first-order recursion with initial condition  $\epsilon(0) = (D(0) - D)^2$ . It is easy to verify that it has the solution

$$\epsilon(n) = a^n\epsilon(0) + \left[\frac{1-a^n}{1-a}\right]b. \quad (\text{A6})$$

To ensure that  $\lim_{n \rightarrow \infty} \epsilon(n) < \infty$ , we must choose  $-1 < a < 1$ , which, after some calculations, results in the condition:  $0 < \mu < 4/(3 + 4\sigma_u^2)\Omega^2$ . The steady-state value of  $\epsilon(n)$  is then given by  $b/(1-a)$ , which, after substitution of the values of  $a$  and  $b$  and some simplification, yields (10).

## REFERENCES

- [1] G. C. Carter, *Coherence and Time Delay Estimation: An Applied Tutorial for Research, Development, Test, and Evaluation Engineers*. Piscataway, NJ: IEEE Press, 1993.
- [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, pp. 320–327, Aug. 1976.
- [3] H. C. So, ‘‘A comparative study of two discrete-time phase delay estimators,’’ *IEEE Trans. Instrum. Meas.*, vol. 54, no. 6, pp. 2501–2504, Dec. 2005.
- [4] R. A. Reed, P. L. Feintuch, and N. J. Bershad, ‘‘Time delay estimation using the LMS adaptive filter—Static behaviour,’’ *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, pp. 561–571, Jun. 1981.
- [5] S. R. Dooley and A. K. Nandi, ‘‘Adaptive subsample time delay estimation using Lagrange interpolators,’’ *IEEE Signal Process. Lett.*, vol. 6, no. 3, pp. 65–67, Mar. 1999.
- [6] H. C. So and P. C. Ching, ‘‘A comparative study of five LMS-based adaptive algorithms for nonstationary delay estimation,’’ *Proc. Inst. Elect. Eng. Radar, Sonar, Navig.*, vol. 148, no. 1, pp. 9–15, Feb. 2001.
- [7] 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, Jul. 2003.
- [8] D. L. Maskell and G. S. Woods, ‘‘Adaptive subsample delay estimation using a modified quadrature phase detector,’’ *IEEE Trans. Circuits, Syst.*, vol. 52, pt. II, pp. 669–674, Oct. 2005.
- [9] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1986.
- [10] Y. H. Hu, ‘‘CORDIC-based VLSI architecture for digital signal processing,’’ *IEEE Signal Process. Mag.*, vol. 9, no. 7, pp. 16–35, Jul. 1992.
- [11] M. Chakraborty, A. S. Dhar, and M. H. Lee, ‘‘A trigonometric formulation of the LMS algorithm for realization on pipelined CORDIC,’’ *IEEE Trans. Circuits Syst. II*, vol. 9, pp. 530–534, Sep. 2005.