A Levinson-Type Algorithm for Modeling Fast-Sampled Data

Rajiv Vijayan, Student Member, IEEE, H. Vincent Poor, Fellow, IEEE, John B. Moore, Fellow, IEEE, and Graham C. Goodwin, Fellow, IEEE

Abstract—The standard discrete-time autoregressive model is poorly suited for modeling series obtained by sampling continuous-time processes at fairly rapid rates. Large computational errors can occur when the Levinson algorithm is used to estimate the parameters of this model, due to the ill-conditioning of the Toeplitz covariance matrix to be inverted. An alternative model is developed based on an incremental difference operator rather than the shift operator. It is shown that, as the sampling period goes to zero, unlike the standard AR parameters, the coefficients of this model converge to certain parameters that depend directly on the statistics of the continuous-time process. A Levinson-type algorithm for efficiently estimating the parameters of this model is derived. Numerical examples are given to show that, when the sampling interval is small, this algorithm is considerably less sensitive to arithmetic roundoff errors than the Levinson algorithm.

I. INTRODUCTION

Many discrete-time algorithms used in estimation and control become ill-conditioned when applied to continuous-time processes sampled at very fast rates. For example, in Kalman–Bucy filtering, the discrete-time algebraic Riccati equation (DARE) does not converge smoothly to a meaningful limit. Kalman-Bucy filtering, the discrete-time algebraic Riccati equation (DARE) does not converge smoothly to a meaningful limit. An alternative model is developed based on an incremental difference operator rather than the shift operator. It is shown that, as the sampling period goes to zero, unlike the standard AR parameters, the coefficients of this model converge to certain parameters that depend directly on the statistics of the continuous-time process. A Levinson-type algorithm for efficiently estimating the parameters of this model is derived. Numerical examples are given to show that, when the sampling interval is small, this algorithm is considerably less sensitive to arithmetic roundoff errors than the Levinson algorithm.

The success of using divided-differenced models in the Kalman–Bucy filtering and pole-assignment problems of [12], [8] suggests that a similar approach might be useful here as well. In this paper, we develop this idea. In Section II, we review the conventional autoregressive model and then introduce a linear model based on the δ operator rather than the q operator. Since any polynomial of degree n in q is equivalent to a unique polynomial of degree n in δ and vice versa, this transformation does not entail any loss of generality in the model. Even though estimating the parameters of this model involves solving a non-Toeplitz system of linear equations, we derive a Levinson-type algorithm that recursively estimates the parameters with solutions converge smoothly to continuous-time counterparts based on differential models. Similar transformations have also been used to design digital filters with low coefficient sensitivity and roundoff noise errors in [1], [9], and [14].

Similar problems are encountered when the Levinson algorithm is used to estimate the parameters of the standard autoregressive model, given the covariance sequence \({\{c_k\}}\) of a wide-sense stationary discrete-time random process \({\{X_k\}}\). The algorithm recursively solves the Yule–Walker equations of successively higher order given by the following:

\[
R_n q_n = \left[ \pi_n, 0, \cdots, 0 \right]^T \tag{1}
\]

where \(R_n\), the \((n + 1) \times (n + 1)\) Toeplitz matrix whose \(j, k\)th element is \(c_{k-j}\), is the covariance matrix of \(X_n = [X_n, X_{n-1}, \cdots, X_0]\). The numerical stability of the Levinson algorithm for solving (1) has been established by Cybenko in [2]. However, as pointed out in [2], in many cases of practical interest, the matrix \(R_n\) is ill-conditioned, which results in unacceptable errors when the algorithm is implemented. This ill-conditioning occurs when the prediction error \(\pi_n\) is very small, or, equivalently, when the reflection coefficients are close to ±1. An important case where ill-conditioning of this nature occurs is when the discrete-time process is obtained by sampling a continuous-time process at fairly rapid rates, since \(\lim_{\Delta \to 0} \text{Cov}(X_k, X_l) = \delta_0\) for all \(k\) and \(l\), if the underlying continuous-time process \({\{X(t); t \in \mathbb{R}\}}\) is sufficiently smooth (mean-square continuous). Since the problem is due to the poor conditioning of \(R_n\), it cannot be solved by using alternative algorithms, like the Schur algorithm, to solve (1), as noted in [2] and [15]. As we shall show, as \(\Delta \to 0\), if the autocovariance function of the continuous-time process has sufficiently many derivatives, the coefficients obtained by the Levinson algorithm will converge to the binomial coefficients \((-1)^i \binom{n}{i}\) independently of the underlying process. This renders the autoregressive parameters meaningless as the sampling rate increases. This phenomenon is similar to what happens to the DARE in the Kalman–Bucy filtering problem [12], and points to a need to modify the autoregressive model for rapidly sampled data.

The success of using divided-differenced models in the Kalman–Bucy filtering and pole-assignment problems of [12], [8] suggests that a similar approach might be useful here as well. In this paper, we develop this idea. In Section II, we review the conventional autoregressive model and then introduce a linear model based on the δ operator rather than the q operator. Since any polynomial of degree n in q is equivalent to a unique polynomial of degree n in δ and vice versa, this transformation does not entail any loss of generality in the model. Even though estimating the parameters of this model involves solving a non-Toeplitz system of linear equations, we derive a Levinson-type algorithm that recursively estimates the parameters with
O(n^2) complexity. It is shown that, as in the case of the AR model, the parameters of the delta model can be associated with orthogonal polynomials on certain contours in the complex plane. In Section III, we consider the limiting behavior of the delta and AR models as the sampling interval goes to zero. In particular, we see that, as \( A \to 0 \), the parameters of the delta model converge to certain regression coefficients associated with the mean-square derivatives of the underlying continuous-time process. These coefficients are intimately related to the statistics of the process, unlike the limiting values of the AR coefficients.

Also, the recursive algorithm derived in Section II is seen to have a stable limiting form that is the same as certain recursions associated with continuous-time AR processes. Section IV contains some numerical comparisons of the Levinson and delta algorithms when implemented in finite precision arithmetic. Examples are given to show that, for moderately small \( A \), the new algorithm is much less sensitive to arithmetic roundoff errors than is the Levinson algorithm. Finally, some interesting problems for further research are discussed.

II. MODEL BASED ON DIFFERENCED DATA

Suppose \( X_1, X_2, \ldots, X_{n+1} \) is a segment from a zero-mean covariance stationary random sequence with covariance sequence \( \{ c_n \} \). In the classical autoregressive modeling problem, a parameter vector \( g_n = [a_n, 0, \ldots, a_{n,n}]^T \) with \( a_{n,0} = 1 \) is chosen to minimize the mean-squared prediction error \( E(\epsilon_n^2) \) with \( \epsilon_n \) given by

\[
\epsilon_n = X_{n+1} - \sum_{k=0}^{n} a_{n,k} X_{n+1-k}
\]

where

\[
\hat{X}_n = [X_{n+1}, X_n, \ldots, X_1]^T = [q^n X_1, q^{n-1} X_1, \ldots, X_1]^T.
\]

The problem

\[
\min_{g_n} E(\hat{a}_n^T \hat{X}_n^2) = 1
\]

is solved recursively in \( n \) by the Levinson algorithm [11] as follows:

\[ g_{n+1} = \begin{bmatrix} 1_{n+1} \\ 0 \cdots 0 \end{bmatrix} g_n - \gamma_{n+1} \begin{bmatrix} 0 \cdots 0 \\ J_{n+1} \end{bmatrix} g_n, \quad n = 0, 1, 2, \ldots \]

with initialization \( g_0 = 1 \), where, for each positive integer \( k \), \( 1_k \) denotes the \( k \times k \) identity matrix and \( J_k \) denotes the \( k \times k \) matrix that has all zero entries except for 1's in its antidiagonal. The reflection coefficients \( \{ \gamma_n \} \) are given by

\[
\gamma_{n+1} = \frac{\alpha_n}{\pi_n}
\]

where

\[
\alpha_n = [0, \ldots, 0, 1] R_{n+1} [g_n]_n,
\]

and

\[
\pi_n = E(\epsilon_n^2) = [1, 0, \cdots, 0] R_n g_n.
\]

with \( R_n = E(\hat{X}_n \hat{X}_n^T) \). The sequence \( \{ \pi_n \} \) satisfies the recursion

\[
\pi_{n+1} = \pi_n - \frac{\alpha_n^2}{\pi_n}, \quad n = 0, 1, \ldots
\]

with initialization \( \pi_0 = \epsilon_0 \).

Now, let us assume that the sequence \( \{ X_k \} \) is obtained by sampling a continuous-time process \( \{ X(t); t \in \mathbb{R} \} \) at intervals \( \Delta \), i.e.,

\[
\tilde{X}_n = [X(n \Delta), X((n-1) \Delta), \ldots, X(0)]^T.
\]

We consider a model of the form

\[
\delta^n X_1 + \beta_{n,1} \delta^{n-1} X_1 + \cdots + \beta_{n,n} X_1 = \nu_n,
\]

i.e.,

\[
\beta_n^T \tilde{Z}_n = \nu_n
\]

where \( \tilde{Z}_n = [\delta^n X_1, \delta^{n-1} X_1, \ldots, X_1]^T \), \( \beta_n = [\beta_{n,0}, \beta_{n,1}, \ldots, \beta_{n,n}]^T \) with \( \beta_{n,0} = 1 \), and \( \{ \nu_n \} \) is the sequence of modeling errors.

\[
\delta^n X_1 = \frac{X_2 - X_1}{\Delta}
\]

\[
\delta^2 X_1 = \frac{X_3 - 2X_2 + X_1}{\Delta^2}
\]

\[
\delta^n X_1 = \frac{\delta^{n-1} X_1}{\Delta}
\]

Note that, as \( A \to 0 \), \( \delta^n X_1 \) converges in the mean to \( X^{(k)}(0) \), the \( k \)th mean-square derivative of \( \{ X(t); t \in \mathbb{R} \} \) at \( t = 0 \), provided this derivative exists.

The motivation for considering the model (8) is its parallelism with the continuous-time autoregressive model given by

\[
dX(t) + \delta_{n,1} X^{(n-1)}(t) dt + \cdots + \delta_{n,n} X(t) dt = dW(t)
\]

where \( \{ W(t); t \in \mathbb{R} \} \) is a Wiener process.

Another continuous-time model that has been used in [5], [4], and [3] is the following, which is based on an integral operator:

\[
dX(t) + \int_{-T}^{T} a(T; T-(t-s)) dX(s) dt = dW(t).
\]

In [3], this model is approximated by using the standard discrete-time AR model with order \( n = T/\Delta \). As \( \Delta \to 0 \), \( n \to \infty \) and the limiting values of the discrete AR parameters \( a_{n,j} \) are related to the continuous AR function \( a(T; t) \). The disadvantage of this approach is that, for small \( \Delta \), the number of parameters in the model becomes very large. In comparison, (8) gives a parsimonious parametrization that also converges to a meaningful continuous-time model.

\[
\tilde{X}_n \text{ is obtained by a linear transformation of } \tilde{X}_n \text{ that has the form } \tilde{Z}_n = T_n^{-1}\tilde{X}_n.
\]

Since \( \delta^k = (q - 1)^k/\Delta^k \), \( T_n \) is an \( (n+1) \times (n+1) \) upper-triangular matrix whose \( l \)th element is given by

\[
(T_n)_{l,k} = \frac{(-1)^{l-k}}{\Delta^{n-k}} \binom{n-k}{l-k}, \quad 0 \leq l,k \leq n.
\]
(Here, and in the sequel, we follow the convention that the binomial coefficient \( \binom{n}{k} = 0 \) for \( k < 0 \) and \( k > n \).) Thus, \( T_n \) is an invertible lower triangular matrix whose \( n \times n \) right lower submatrix is \( T_{n-1} \), with \( T_0 = I \). The inverse of this matrix is given by

\[
(T_n)^{-1}_{i,k} = \Delta^{-2}(n-k), \quad 0 \leq i, k \leq n. \tag{10}
\]

It is straightforward to see that the vector \( \beta_n \) that solves

\[
\min_{\beta_n} E(\beta_n^2) = 1
\]

is given by

\[
\beta_n = (T_n)_{i=0} g_n = \Delta^n T_n^{-1} g_n \tag{12}
\]

where \( g_n \) solves (2).

It is the Toeplitz property of \( R_n \), the covariance matrix of \( \tilde{X}_n \), that makes it possible to solve (2) recursively using \( O(n^2) \) computations. Since \( T_n \) is triangular, if we knew \( R_n \), it would be possible to solve (11) using \( O(n^2) \) computations, by first solving (2) for \( g_n \) using the Levinson or Schur algorithm, and then using (12) to obtain \( \beta_n \). However, in this procedure, any numerical errors in calculating \( g_n \) due to the ill-conditioning of \( R_n \) would carry over to the calculation of \( \beta_n \). Also, it is not recursive in \( n \).

Exploiting the special structure of \( T_n \) given by (9), we shall now obtain an \( O(n^2) \) algorithm to solve (11), that requires knowledge of only the non-Toeplitz covariance matrix of \( \tilde{Z}_n \). This algorithm has the added advantage of being recursive, like the Levinson algorithm. (In [7], Messerschmitt has derived a generalized Levinson algorithm that replaces the shift operator by any all-pass operator. However, the results of that paper do not apply to our model because the \( \delta \) operator is not all pass.)

**Proposition 1:** The argument solving (11) is given recursively (in \( n \)) by

\[
\beta_{n+1} = C_n \beta_n + \frac{\gamma_n+1}{\Delta} \left[ \begin{array}{c} 0 \\ \vdots \\ \vdots \\ 0 \\ \delta_n 
\end{array} \right] \frac{\alpha_n}{\alpha_{n-1}} \left[ \begin{array}{c} 0 \\ \vdots \\ \vdots \\ 0 \\ \Delta
\end{array} \right] \beta_{n-1}, \tag{13}
\]

with initialization \( \beta_0 = 0, \beta_1 = 1, \) and \( \gamma_0 = -\Delta^{-2} \). Here, \( C_n \) is the \( (n + 2) \times (n + 1) \) matrix defined by

\[
C_n = \left[ \begin{array}{cccccc}
\Delta & 0 & 0 & \cdots & 0 \\
1 & \Delta & 0 & \cdots & 0 \\
0 & 1 & \Delta & \cdots & 0 \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & 1 & \Delta \\
0 & 0 & \cdots & 0 & 1
\end{array} \right]
\]

\( \bar{\gamma}_n \) is defined by

\[
\bar{\gamma}_{n+1} = \bar{\alpha}_n / \bar{\gamma}_n, \quad n = 1, 2, \ldots,
\]

with

\[
\bar{\alpha}_n = [0, \ldots, 0, 1] Q_n [\beta_n], \quad n = 1, 2, \ldots,
\]

\[
\bar{\alpha}_0 = E(X_1 \cdot \delta X_1) + E(X_1^2) / \Delta
\]

and

\[
\bar{\gamma}_n = E(n_0^2) = [1, 0, \cdots, 0] Q_n [\beta_n], \quad n = 0, 1, 2, \ldots,
\]

with \( Q_n = E(\tilde{Z}_n \tilde{Z}_n^T) = T_n^T R_n T_n \).

**Proof:** By substituting for \( J_{n+1} g_n \) in (3) using (3) with \( n \) replaced by \( n - 1 \), it can be shown that the Levinson algorithm is equivalent to the following 3-term recursion:

\[
g_{n+1} = \left[ \begin{array}{c} I_{n+1} \\
0 \cdots 0
\end{array} \right] g_n + \frac{\gamma_{n+1} + 1}{\gamma_n} \left[ \begin{array}{c} 0 \cdots 0 \\
I_{n+1}
\end{array} \right] g_n
\]

\[
+ \frac{\gamma_{n+1}}{\gamma_n} \left[ \begin{array}{c} 0 \cdots 0 \\
I_n
\end{array} \right] \Delta^n T_n \beta_n,
\]

where \( \gamma_0 = -1 \) and \( \gamma_{-1} = 0 \).

Substituting for \( g_n \), we get

\[
\Delta^n T_{n+1} \beta_{n+1} = \left[ \begin{array}{c} I_{n+1} \\
0 \cdots 0
\end{array} \right] \Delta^n T_n \beta_n
\]

\[
+ \frac{\gamma_{n+1}}{\gamma_n} \left[ \begin{array}{c} 0 \cdots 0 \\
I_{n+1}
\end{array} \right] \Delta^n T_n \beta_n
\]

\[
+ \frac{\gamma_{n+1}}{\gamma_n} \left[ \begin{array}{c} 0 \cdots 0 \\
I_n
\end{array} \right] \Delta^n T_{n-1} \beta_{n-1}.
\]

It can be easily shown that

\[
\frac{1}{\Delta} T_{n+1} \left[ \begin{array}{c} T_n \\
0 \cdots 0
\end{array} \right] = C_n \tag{14}
\]

and that

\[
T_{n+1} \left[ \begin{array}{c} 0 \cdots 0 \\
T_n
\end{array} \right] = \left[ \begin{array}{c} 0 \cdots 0 \\
I_n
\end{array} \right]. \tag{15}
\]

Therefore, we get

\[
\beta_{n+1} = C_n \beta_n + \frac{\gamma_{n+1}}{\Delta} \gamma_n \left[ \begin{array}{c} 0 \\
C_n^{-1}
\end{array} \right] \beta_{n-1}. \tag{15}
\]

We have

\[
\gamma_{n+1} \left( \frac{1}{\gamma_n} - \gamma_n \right) = \frac{\alpha_n}{\alpha_{n-1}} \left( \frac{\alpha_{n-1}}{\alpha_{n-1}} - \frac{\alpha_{n-1}}{\pi_{n-1}} \right)
\]

\[
= \frac{\alpha_n}{\alpha_{n-1}} \left( \frac{\alpha_{n-1}}{\pi_{n-1}} - \frac{\alpha_{n-1}^2}{\pi_{n-1}} \right)
\]

\[
= \frac{\alpha_n}{\alpha_{n-1}} \quad \text{by (7).} \tag{16}
\]

\[
\bar{\gamma}_n = [1, 0, \cdots, 0] Q_n \beta_n
\]

\[
= [1, 0, \cdots, 0] T_n^T R_n \Delta^{-n} T_{n-1} \beta_{n-1}
\]

\[
= \Delta^{-n} T_{n-1} [1, 0, \cdots, 0] T_n^T R_n \beta_n
\]

\[
= \Delta^{-n} T_{n-1} \pi_n (T_n^T)^{0,0}
\]

\[
= \Delta^{-2n} \pi_n, \quad n = 0, 1, 2, \ldots \tag{17}
\]
Also,
\[
\alpha_n = [0, 0, \cdots, 1] Q_{n+1} [\beta_n^T]_0.
\]
Since \( T_{n+1}^T \) is upper triangular and \((T_{n+1}^T)_{n+1,n+1} = 1\), we have
\[
[0, 0, \cdots, 1] T_{n+1}^T = [0, 0, \cdots, 1].
\]
Therefore,
\[
\alpha_n = \Delta^n [0, 0, \cdots, 1] Q_n T_{n+1} [\beta_n^T]_0
\]
\[
= \Delta^n [0, 0, \cdots, 1] Q_n \Delta C_n \beta_n^T, \quad \text{by (14)}
\]
\[
= \Delta^n [0, 0, \cdots, 1] Q_{n+1} \left( \begin{bmatrix} \beta_n^T \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ \beta_n^T \end{bmatrix} \right).
\]
Since the \( n \times n \) lower right submatrix of \( Q_{n+1} \) is \( Q_n \) and
\[
Q_n \beta_n^T = \pi_n [1, 0, \cdots, 0]^T
\]
we have
\[
[0, 0, \cdots, 1] Q_{n+1} \left[ \begin{bmatrix} 0 \\ \beta_n^T \end{bmatrix} \right] = 0, \quad n = 1, 2, \cdots.
\]
Therefore,
\[
\alpha_n = \Delta^n [0, 0, \cdots, 1] Q_n \left[ \begin{bmatrix} \beta_n^T \\ 0 \end{bmatrix} \right]
\]
\[
= \Delta^n \Delta \alpha_n, \quad n = 1, 2, \cdots, \quad (18)
\]
\( \Delta \alpha_n \) is defined in such a way to ensure that (18) holds for \( n = 0 \) as well. Equations (17) and (18) yield
\[
\tilde{\gamma}_{n+1} = \Delta^{-1} \gamma_{n+1}.
\]
By substituting into (15) using (16)-(19), we obtain the desired result (13). \( \square \)

We now note that (13) can also be written as
\[
\beta_{n+1,j} = \beta_{n,j} + \frac{1}{\Delta} \left( \Delta + \frac{\tilde{\gamma}_{n+1}}{\tilde{\gamma}_n} \right) \beta_{n,j-1} - \frac{\tilde{\Delta}_n}{\tilde{\Delta}_{n-1}} 
\]
\[
\left( \beta_{n-1,j-1} + \frac{1}{\Delta} \beta_{n-1,j-2} \right), \quad j = 0, \cdots, n + 1 \quad (20)
\]
where we assume that \( \beta_{n,j} = 0 \) for \( j < 0 \) and \( j > n \). On comparing (20) with its Levinson counterpart, we see that the complexity of the new algorithm is of the same order despite the fact that the algorithm inverts a non-Toeplitz matrix.

We now examine the relation between these recursions and the theory of orthogonal polynomials. We associate polynomials \( A_n(z) \) with \( \beta_n \), defined by
\[
A_n(z) = z^n + \beta_{n,1} z^{n-1} + \cdots + \beta_{n,n}.
\]
It is well known, as described by Kailath, Vieira, and Morf in [5], that these polynomials are the Szegő orthogonal polynomials on the unit circle, i.e.,
\[
\frac{1}{2\pi} \int_{-\pi}^{\pi} A_m(e^{i\theta}) A_n(e^{i\theta}) dF(e^{i\theta}) = \pi_n \delta_{mn} \quad (21)
\]
where \( F(e^{i\theta}) \) is an increasing function of \( \theta \in [-\pi, \pi] \) called the spectral distribution function of the process. The Levinson algorithm is equivalent to the following recursion for these polynomials given by Szegő in [13]:
\[
A_{n+1}(z) = z A_n(z) - \gamma_{n+1} \hat{A}_n(z)
\]
\[
\hat{A}_{n+1}(z) = -z \gamma_{n+1} A_n(z) + \hat{A}_n(z) \quad (22)
\]
where \( \hat{A}_n(z) \) is the reverse polynomial of \( A_n(z) \) defined by
\[
\hat{A}_n(z) = z^n A_n(z^{-1}) = a_{n,n} z^n + a_{n,n-1} z^{n-1} + \cdots + 1.
\]
Similarly, we associate polynomials \( B_n(z) \) with \( \beta_n \) defined by
\[
B_n(z) = z^n + \beta_{n,1} z^{n-1} + \cdots + \beta_{n,n}.
\]
Equation (12) is then equivalent to the polynomial relation
\[
A_n(z) = \Delta^n B_n \left( \frac{z - 1}{\Delta} \right).
\]
Comparing to (21), we see that the \( B_n(z) \) are polynomials orthogonal on the circle of radius \( 1/\Delta \) centered at \((-1/\Delta, 0)\). Equation (20) is thus equivalent to the polynomial recursion
\[
B_{n+1}(z) = z B_n(z) + \frac{1}{\Delta} \left( \Delta - \frac{\tilde{\gamma}_{n+1}}{\tilde{\gamma}_n} \right) B_n(z)
\]
\[
- \frac{\tilde{\alpha}_n}{\tilde{\alpha}_{n-1}} \left( z + \frac{1}{\Delta} \right) B_{n-1}(z). \quad (23)
\]
By introducing the "backward" polynomials \( \hat{B}_n(z) \) defined by the relation
\[
\hat{A}_n(z) = \Delta^n \hat{B}_n \left( \frac{z - 1}{\Delta} \right)
\]
it can be shown that (23) has the following equivalent form that resembles the forward-backward form of the Levinson algorithm:
\[
B_{n+1}(z) = \left( z + \frac{1}{\Delta} \right) B_n(z) - \Delta^{-n} \tilde{\gamma}_{n+1} \hat{B}_n(z)
\]
\[
\hat{B}_{n+1}(z) = - \left( z + \frac{1}{\Delta} \right) \Delta^{-n} \tilde{\gamma}_{n+1} B_n(z) + \frac{1}{\Delta} \hat{B}_n(z). \quad (24)
\]
The backward-forward version of the Levinson algorithm requires fewer computations than the equivalent 3-term recursion since the backward parameters, being the forward parameters in reversed order, do not have to be computed separately. For the delta recursion, since both forward and backward parameters have to be computed, the amount of computation required is the same for both versions. Numerical calculations have shown that both versions have similar performance when implemented in finite-precision arithmetic. However, as shown in Section III, the 3-term recursion of (20) has the added advantage that, as \( \Delta \to 0 \), it has a limiting form that can be related to certain recursions for the underlying continuous-time process. For these reasons, in the remaining sections of this paper, we will restrict our attention to the 3-term recursion for estimating the parameters of the delta model.

III. LIMITING BEHAVIOR

In this section, we compare the limiting behavior, as \( \Delta \to 0 \), of the autoregressive and delta models for a discrete-time proc-
ess \{ X_k \}_{k \in \mathbb{Z}} \) obtained by sampling a zero-mean wide-sense stationary continuous-time process \{ X(t); t \in \mathbb{R} \} at interval \( \Delta \).

In order to simplify notation, the dependence of the various discrete-time quantities on \( \Delta \) will be assumed implicitly. Hence, \( a_{n,j} \) will refer to \( a_{n,j}(\Delta) \), and so forth.

First, we consider the limiting behavior of the autoregressive parameters as the sampling interval tends to zero.

**Proposition 2:** Assume that the continuous-time process satisfies the following conditions:

- A1: \( \{ X(t); t \in \mathbb{R} \} \) has \( n \) mean-square derivatives.
- A2: The random vector of derivatives \( \{ X^{(n)}(0), \ldots, X^{(n)}(0), X(0) \} \) has a nonsingular covariance matrix.

Then

\[
\lim_{\Delta \to 0} a_{n+1,j} = (-1)^j \frac{n+1}{j}.
\]

and

\[
\gamma_{n+1} = (-1)^n.
\]

**Proof:** Consider a finite segment \( \{ X_{-1}, X_0, X_1, \ldots, X_n \} \) of the discrete-time process. The \( n \)th-order forward and backward prediction errors are defined by

\[
F_n = X_n - \hat{E}[X_n | X_0, X_1, \ldots, X_{n-1}]
\]

and

\[
B_n = X_{n-1} - \hat{E}[X_{n-1} | X_0, X_1, \ldots, X_{n-1}],
\]

respectively, where \( \hat{E}[X | Y] \) denotes the linear minimum-mean-squared-error estimate of \( X \) given \( Y \).

The reflection coefficient \( \gamma_{n+1} \) is given by

\[
\gamma_{n+1} = \frac{E(F_n B_n)}{E(F_n^2)}.
\]

Now, we define the operators

\[
\delta_f = (q - 1) / \Delta
\]

and

\[
\delta_b = (1 - q^{-1}) / \Delta.
\]

Then

\[
\text{span} \{ X_0, X_1, \ldots, X_{n-1} \} = \text{span} \{ \delta_f X_0, \ldots, \delta_f X_0, X_0 \} = \text{span} \{ \delta_b X_{n-1}, \ldots, \delta_b X_{n-1}, X_{n-1} \}.
\]

Therefore, \( F_n = \Delta F_n^{(b)} \) and \( B_n = (-1)^n \Delta B_n^{(b)} \), where

\[
F_n^{(b)} = \delta_f X_0 - \hat{E}[\delta_f X_0 | \delta_f X_0, \ldots, \delta_f X_0, X_0]
\]

and

\[
B_n^{(b)} = \delta_b X_{n-1} - \hat{E}[\delta_b X_{n-1} | \delta_b X_{n-1}, \ldots, \delta_b X_{n-1}, X_{n-1}].
\]

Hence, we have

\[
\gamma_{n+1} = (-1)^n \frac{E(F_n^{(b)} B_n^{(b)})}{E((F_n^{(b)})^2)}. \tag{26}
\]

Under assumption A1,

\[
\lim_{\Delta \to 0} \delta_f X_0 = \lim_{\Delta \to 0} \delta_b X_{n-1} = X^{(j)}(0) \text{ (mean square)},
\]

\( j = 0, 1, \ldots, n \). \tag{27}

Let

\[
\tilde{Z}_n = [\delta_f^{-1} X_0, \delta_f^{-2} X_0, \ldots, X_0]^T
\]

and

\[
\tilde{Z}_{n,0} = [X^{(n-1)}(0), X^{(n-2)}(0), \ldots, X(0)]^T.
\]

Then,

\[
\hat{E}[X^{(n)}(0) | \tilde{Z}_{n,0}] = \epsilon_n \hat{P}_n^{-1} \tilde{Z}_n \text{ and } \hat{E}[X^{(n)}(0) | \tilde{Z}_{n,0}] = \tilde{Z}_{n,0} \hat{Q}_n^{-1} \tilde{Z}_{n,0},
\]

where

\[
\hat{Q}_n = \hat{E}[\tilde{Z}_n \tilde{Z}_n^T],
\]

\[
\hat{P}_n = \hat{E}[\tilde{Z}_n X(0)] = \epsilon_n \hat{Q}_n^{-1} \tilde{Z}_n,
\]

and

\[
\epsilon_n = \hat{E}[\tilde{Z}_n X(0)].
\]

Assumption A2 implies that \( \hat{Q}_{n,0} \) is positive definite.

By (27), \( \lim_{\Delta \to 0} \hat{Q}_n = \hat{Q}_{n,0} \) and \( \lim_{\Delta \to 0} \epsilon_n = \epsilon_{n,0} \).

By the Banach lemma for nonsingular matrices

\[
\|\hat{Q}_n^{-1} - \hat{Q}_{n,0}^{-1}\| \leq \|\hat{Q}_n^{-1}\| \|\hat{Q}_{n,0} - \hat{Q}_n\| \|\hat{Q}_{n,0}\|.
\]

Therefore

\[
\lim_{\Delta \to 0} \hat{Q}_n^{-1} = \hat{Q}_{n,0}^{-1}
\]

and, as a result

\[
\lim_{\Delta \to 0} \hat{E}[\delta_f X_0 | \tilde{Z}_n] = \hat{E}[X^{(n)}(0) | \tilde{Z}_{n,0}].
\]

Therefore

\[
\lim_{\Delta \to 0} E(\nabla_n^{(b)}) = X^{(n)}(0) - \hat{E}[X^{(n)}(0) | \tilde{Z}_{n,0}] \text{ (mean square)}.
\]

Similarly, it can be shown that

\[
\lim_{\Delta \to 0} B_n^{(b)} = X^{(n)}(0) - \hat{E}[X^{(n)}(0) | \tilde{Z}_{n,0}] \text{ (mean square)}.
\]

Thus, \( F_n^{(b)} \) and \( B_n^{(b)} \) have the same mean-square limit. Furthermore, assumption A2 ensures that \( \lim_{\Delta \to 0} E((F_n^{(b)})^2) > 0 \). Hence, by letting \( \Delta \to 0 \) in (26), we get (25). Equation (24) follows from (25), using the Levinson recursion and induction on \( n \).

We now consider the limiting behavior of the algorithm for estimating the parameters of the delta model. Equation (20) can be rewritten as

\[
\beta_{n,j} = \beta_{n-1,j} + K_n \beta_{n-1,j-1} - P_n \beta_{n-2,j-1} - Q_n \beta_{n-2,j-2},
\]

\( j = 0, \ldots, n \)

where

\[
K_n = \frac{1}{\Delta^2} \left[ \Delta + \frac{\gamma_n}{\tilde{r}_n} \right],
\]

\[
P_n = \tilde{a}_{n-1} \tilde{a}_{n-2}.
\]
and

\[ Q_n = \frac{\tilde{\sigma}_{n-1}}{\Delta \tilde{\sigma}_{n-2}}. \]

We now show that the coefficients \( K_n, P_n, \) and \( Q_n \) remain bounded as \( \Delta \to 0 \), if assumptions A1 and A2 hold.

\( Q_n \) can be written as

\[ Q_n = \frac{\tilde{\gamma}_n \tilde{\tau}_{n-1}}{\Delta \tilde{\tau}_{n-2}} = \frac{\gamma_n \tilde{\tau}_{n-1}}{\tilde{\tau}_{n-1} \tilde{\tau}_{n-2}}. \]

Since \( \tilde{\tau}_n = E[F_n^2] \), from the proof of Proposition 2, we have

\[ \lim_{\Delta \to 0} \tilde{\tau}_{n,0} = \tilde{\tau}_{n-1,0} E[(F_n)^2] > 0. \]

Hence

\[ \lim_{\Delta \to 0} Q_n = -\frac{\tilde{\tau}_{n-1,0}}{\tilde{\tau}_{n-2,0}}. \]

Since \( P_n = \Delta Q_n \)

\[ \lim_{\Delta \to 0} P_n = 0. \]

\( K_n \) can be written as

\[ K_n = \frac{1}{\Delta^2} \left[ \Delta + \Delta \frac{\gamma_n}{\tilde{\tau}_{n-1}} \right] \]

\[ = \frac{1}{\Delta} \left[ 1 + \frac{\gamma_n}{\tilde{\tau}_{n-1}} \right]. \]

From (7), we have \( \alpha_{n-1} = (\tau_{n-1} - \tau_n)/\gamma_n \). Hence

\[ \gamma_n = \frac{\alpha_{n-1}}{\tau_{n-1}} = \frac{1}{\gamma_n} \left( 1 - \frac{\tau_n}{\tau_{n-1}} \right), \]

i.e.,

\[ \gamma_n^2 = 1 - \frac{\tau_n}{\tau_{n-1}}. \]

By (17), \( \tau_n = \Delta^2 \tilde{\tau}_n \). Since \( \tilde{\tau}_n = O(1), \tau_n = O(\Delta^2 n) \). Thus, we have

\[ \gamma_n^2 = 1 + O(\Delta^2). \]

Using (25), we get

\[ \gamma_n = (-1)^{n-1} + O(\Delta^2) \]

and, as a result

\[ K_n = O(\Delta). \]

Thus, in the limit, (20) has the form

\[ \beta_{n,j} = \beta_{n-1,j} + \frac{\tilde{\tau}_{n-1,j-1}}{\tilde{\tau}_{n-2,0}} \beta_{n-2,j-2}, \quad j = 0, \ldots, n. \]

This limiting form is seen to be the same as a Levinson type recursion for continuous-time autoregressive models derived in [10]. Although our limiting result does not properly apply to the continuous-time AR(n) process (since it has only \( n - 1 \) derivatives), it is still interesting to note that a stable meaningful limiting form of the algorithm is obtained. However, this does of course apply to all processes, not just autoregressive processes, that satisfy the aforementioned conditions.

IV. NUMERICAL RESULTS

We now present some numerical results to compare the performance of the Levinson and delta recursions when implemented using finite-precision arithmetic.

For the first example, the underlying continuous-time process was chosen to have the smooth autocovariance function

\[ R_X(t) = e^{-t^2}. \]

The autoregressive and delta model parameters for \( n = 3 \) were computed in infinite precision and finite precision floating point arithmetic. We define the relative error due to finite precision effects by

\[ \text{Relative Error} = \frac{E\{P \} - E\{\alpha\}}{E\{\alpha\}}. \]

where \( E\{P\} \) and \( E\{\alpha\} \) are the mean-squared modeling errors when implemented in infinite and finite precision, respectively. This quantity is plotted against the number of mantissa bits in Figs. 1, 2, and 3 for sampling intervals of 0.1, 0.3, and 0.5, respectively. The figures show that the relative errors due to finite wordlength effects are significantly smaller for the delta algorithm than for the Levinson algorithm at all sampling rates. The
improvement in performance is seen to be greater at smaller sampling intervals.

For the second example, we consider a continuous-time process with the triangular spectrum given by

\[ S(f) = \begin{cases} \frac{(f_0 - |f|)}{f_0} & \text{if } |f| \leq f_0 \\ 0 & \text{otherwise.} \end{cases} \]

The relative errors for both algorithms with sampling at twice and thrice the Nyquist rate are plotted in Figs. 4 and 5, respectively. In both cases, the delta model is seen to be more stable with respect to roundoff errors, the improvement being more pronounced at the faster sampling rate.

Alternative versions of the Levinson algorithm have been derived, for example, in [6], that are more robust to finite precision effects. The advantage of such algorithms is that intermediate variables take on values in known ranges, which is important in fixed-point computations. However, such scaling considerations are not significant in floating-point calculations. Hence, for the purpose of comparison with the delta algorithm, the standard Levinson algorithm was used rather than any of its variants.

The superior numerical stability of the delta model can be explained in terms of the limiting results obtained in Section III. As shown in Section II, there is a one-to-one correspondence between the parameter vectors \( \mathbf{g}_n \) and \( \mathbf{b}_n \). However, the limiting value of \( \mathbf{b}_n \) is independent of the statistics of the process. Hence, the useful information in the parameter vector is being "compressed." As a result of this, small perturbations in the coefficients can cause large variations in the modeling error. The delta coefficients do not suffer from this problem since their limiting values contain useful information about the process.

V. CONCLUSIONS

In this paper, we have developed a linear model based on the delta operator for a rapidly sampled random process. An algorithm that recursively computes the parameter vector \( \mathbf{b}_n \) associated with this model using \( O(n^2) \) computations has been derived. If the underlying process \( \{X(t); t \in \mathbb{R}\} \) has mean-square derivatives up to order \( n \), then, as \( \Delta \to 0 \), the components of \( \mathbf{b}_n \) converge to the coefficients of the linear minimum-mean-square-error estimator of \( X^{(n)}(t) \) given all the lower order derivatives at time \( t \). Thus, for small \( \Delta \), \( \mathbf{b}_n \) gives a more useful parametrization of the underlying continuous-time process than the Levinson coefficient vector \( \mathbf{g}_n \), which converges to a vector that does not depend on the statistics of the process. Some numerical results have been given to illustrate the superior finite wordlength performance of the delta algorithm.

As a final comment, we note that, in practice, it will often be necessary to estimate the \( \mathbf{Q}_n \) matrix from noisy data. Thus, an issue that deserves further attention is the effect that different procedures for estimating \( \mathbf{Q}_n \) may have upon the estimated model parameters. Other open problems of interest include the development of delta-based algorithms analogous to the lattice and Schur algorithms available in the autoregressive case.

ACKNOWLEDGMENT

This work was performed in part while the second author was a visitor in the Department of Electrical and Computer Engineering at the University of Newcastle, NSW, Australia.

The authors are grateful to the reviewers for their comments, which have helped to improve this paper.

REFERENCES

Vijayan et al.: A Levinson-Type Algorithm


Rajiv Vijayan (S'88) was born in Thrissur, India, on November 9, 1964. He received the B. Tech. degree from the Indian Institute of Technology, Madras, in 1986, and the M.S. degree from the University of Illinois, Urbana-Champaign, in 1988, both in electrical engineering.

Since August 1986, he has been a Research Assistant in the Coordinated Science Laboratory at the University of Illinois, and is currently working towards the Ph. D. degree in electrical engineering. His research interests are in the general area of statistical signal processing.

H. Vincent Poor (S'72-M'77-SM'82-F'87) was born in Columbus, GA, in 1951. He graduated with highest honors from Auburn University, Auburn, AL, in 1972, and received the Ph.D. degree from Princeton University, Princeton, NJ, in 1977.

In 1977, he joined the University of Illinois, Urbana-Champaign, where, from 1984, he was Professor of Electrical and Computer Engineering, and Research Professor in the Coordinated Science Laboratory. In 1990, he joined Princeton University as Professor of Electrical Engineering. He has also held visiting appointments at Imperial College, London; at the University of Newcastle, Australia; and in the University of Illinois Center for Advanced Study.

Dr. Poor is the author of a number of publications in the area of statistical signal processing, including the graduate text, An Introduction to Signal Detection and Estimation (New York: Springer-Verlag, 1988). He is also a member of the editorial boards of several journals, including Mathematics of Control, Signals and Systems; and International Journal of Robust and Nonlinear Control. He has been involved in a number of IEEE activities, including as an Associate Editor of this TRANSACTIONS and of the IEEE TRANSACTIONS ON INFORMATION THEORY; as Program Chair of the 1986 Conference on Decision and Control; as General Chair of the 1989 Automatic Control Conference; and as 1990 President of the IEEE Information Theory Society. He is currently serving on the Board of Governors of the IEEE Control Systems Society and of the IEEE Information Theory Society, and on the IEEE Board of Directors.

John B. Moore (S'66-M'67-S'M'77-F'79) was born in China in 1941. He received the B.A. and M.S. degrees in electrical engineering, in 1963 and 1964, respectively, and the Ph.D. degree in electrical engineering from the University of Santa Clara, CA, in 1967.

He was appointed Senior Lecturer at the University of Newcastle, Australia, in 1967, promoted to Associate Professor, in 1968, and full Professor (Personal Chair), in 1973. In 1982, he rejoined the University of Illinois, Urbana-Champaign, where he is currently Professor of Electrical and Computer Engineering, in the Department of Systems Engineering, Research School of Physical Sciences, Australian National University, Canberra, and promoted to Professor in 1990. He has held visiting appointments at the University of Santa Clara, Santa Clara, CA, in 1968; the University of Maryland, College Park, 1970; Colorado State University, Fort Collins, and Imperial College, London, in 1974; the University of California, Davis, in 1977; the University of Washington, Seattle, in 1981; Cambridge University and the National University of Singapore, in 1985; and the University of California, Berkeley, in 1987 and 1989. He has spent periods in industry as a design engineer and as a consultant, and currently has research grants from industry and government laboratories. His current research is in control and communication systems.


Graham C. Goodwin (M'74-S'M'84-F'86) was born in Broken Hill, Australia, in 1945. He received the B.Sc. degree in physics, the B.E. degree in electrical engineering, and the Ph.D. degree, all from the University of New South Wales, Australia.

From 1970 until 1974, he was a lecturer in the Department of Computing and Control, Imperial College, London. Since 1974, he has been with the Department of Electrical Engineering and Computer Science, University of Newcastle, Australia. He is currently Professor of Electrical Engineering and Director of the Center for Industrial Control Science at the University of Newcastle.