Discrete Fast Algorithms for Two-Dimensional Linear Prediction on a Polar Raster

Wen-Hsien Fang and Andrew E. Yagle, Member, IEEE

Abstract—New generalized split Levinson and Schur algorithms for the two-dimensional linear least squares prediction problem on a polar raster are derived. The algorithms compute the prediction filter for estimating a random field at the edge of a disk, from noisy observations inside the disk. The covariance function of the random field is assumed to have a Toeplitz-plus-Hankel structure for both its radial part and its transverse (angular) part. This assumption is valid for some types of random fields, such as isotropic random fields. The algorithms generalize the split Levinson and Schur algorithms in two ways: 1) to two dimensions; and 2) to Toeplitz-plus-Hankel covariances.

I. INTRODUCTION

THE problem of computing linear least squares estimates of two-dimensional random fields from noisy observations has many applications in image processing. In particular, the two-dimensional discrete linear prediction problem is a useful formulation of problems in image smoothing and coding [1]. If the random field: 1) is defined on a rectangular lattice of points; 2) is stationary; and 3) has quarter-plane or asymmetric half-plane causality, then the two-dimensional linear prediction problem may be solved using the multichannel Levinson algorithm [2], [3] (modifications of these conditions are also possible). By exploiting the Toeplitz-block-Toeplitz structure of the covariance function of the stationary random field, this algorithm allows the linear prediction filters to be computed recursively using significantly fewer computations than direct solution of the two-dimensional discrete Wiener-Hopf equations. The multichannel Schur algorithm computes the reflection coefficient matrices from the covariance function; propagating it in parallel with the Levinson algorithm saves even more computation.

In tomographic imaging problems solved by filtered back-projection [4], and in spotlight synthetic aperture radar [5], data are collected on a polar raster of points, rather than on a rectangular lattice. Although such data can be interpolated onto a rectangular lattice, this is necessarily inexact; it also affects the covariance function. For example, the covariance of an isotropic random field on a rectangular lattice is a Toeplitz function of the ordinates and abscissae, while on a polar raster it is a Toeplitz-plus-Hankel function of the radii. For smoothing noisy images and performing image coding for images defined on a polar raster, it is clearly desirable to develop analogues of the multichannel Levinson and Schur algorithms applicable to discrete random fields defined on a polar raster.

This paper develops these analogs. They generalize previous results in three ways: 1) the random field is defined on a polar raster; 2) the random field is not required to be stationary; rather, its covariance must have Toeplitz-plus-Hankel structure in both the radial and transverse directions (some important cases of such random fields are noted in Section IV); and 3) the quarter-plane or asymmetric half-plane causality assumption is replaced by a more natural causality defined in the radial direction only. The prediction filters estimate the random field at a given point using observations from all points of smaller radius.

Three other features are worth noting here. First, the algorithms are generalized three-term recurrences, similar in structure to the split algorithms [6], [7]. The one-dimensional split algorithm recursions require only half as many multiplications as the two-component lattice recursions of the Levinson and Schur algorithms. Our two-dimensional algorithms are similarly computationally efficient, which is important in two-dimensional signal processing. Second, the smoothing filters for estimating the random field from observations at points of smaller and greater radii can be easily computed [8] from the prediction filters using a discrete multidimensional generalization of the application of the Bellman-Siegert-Krein identity to the one-dimensional smoothing problem in [9]. Indeed, the new two-dimensional algorithms of this paper are applied to arbitrary Toeplitz-plus-Hankel-block-Toeplitz-plus-Hankel systems in Section V.

Finally, we note that similar ideas have been applied to continuous-parameter isotropic [10] and homogeneous [11] random fields, and to random fields with more general Toeplitz-plus-Hankel structure in [12] and [13]; this paper can be viewed as a discrete version of the results of [13]. Although the continuous algorithm can always be discretized, an inherently discrete algorithm can be expected to perform better on a computer; there are minor yet significant differences between the results of this paper and the continuous results of [13] (see Section IV). Also, in some problems the data are sampled, or only taken at discrete points. These facts motivated us to de-

Manuscript received November 9, 1989; revised March 23, 1991. This work was supported by the Air Force Office of Scientific Research under Grant AFOSR 89-0017.

The authors are with the Department of Electrical Engineering and Computer Science, University of Michigan, Ann Arbor, MI 48109-2122. IEEE Log Number 9107667.
develop a discrete counterpart of the continuous algorithms. We also note that the one-dimensional version of this algorithm has been presented in [14], and that a summary of the results of this paper was presented in [15].

This paper is organized as follows. In Section II, the two-dimensional analog of the discrete split Levinson recurrence for the linear prediction problem on a polar raster is derived. The derivation is based on the assumption that both the radial part and the transverse part of the covariance have Toeplitz-plus-Hankel structure. Section III derives a corresponding Schur algorithm, to be propagated in parallel with the Levinson algorithm. Some examples of random fields with covariances having Toeplitz-plus-Hankel structure are discussed in Section IV, and comparison with the results of [13] are made. In Section V, the computational complexity of the proposed algorithm is evaluated, and compared with other algorithms. The solution to a general Toeplitz-plus-Hankel-block-Toeplitz-plus-Hankel system of equations is also developed. Section VI concludes with a summary.

II. DERIVATION OF THE LEVINSON-LIKE RECURRENTCE

A. Basic Problem

The problem considered is as follows. Given noisy observations \( \{y_{i,n}\} \) of a zero-mean real-valued discrete random field \( \{x_{i,n}\} \) at the points \( (i, N) \) of a polar raster on a disk, compute the linear least-squares predictions of \( x_{i,n} \) for all points on the edge of the disk using all the data inside the disk. Here \( i \) is an integer index of the argument (angle); if there are \( M \) points distributed on the circle of any radius, then \( (i, N) \) is the point at radius \( i \) and angle \( 2\pi N / M \).

The observations \( \{y_{i,n}\} \) are related to the field \( \{x_{i,n}\} \) by \( y_{i,n} = x_{i,n} + u_{i,n} \), where \( \{u_{i,n}\} \) is a zero-mean discrete white noise field with unit power, and \( \{x_{i,n}\} \) and \( \{y_{i,n}\} \) are uncorrelated (white noise with arbitrary power \( \sigma^2 \) can be easily handled by scaling). The covariance of \( \{x_{i,n}\} \) is

\[
K(i, N; j, N_2) = E[x_{i,n} x_{j,n}]
\]

which is assumed to be a positive semidefinite function with Toeplitz-plus-Hankel structure in both arguments (this is defined precisely in (13) and (14) below). Although an actual covariance would also be a symmetric function, symmetry in (1) is not required by the algorithms to follow; this permits their application to general Toeplitz-plus-Hankel-block-Toeplitz-plus-Hankel systems in Section V.

The estimates of \( x_{i,n} \) at the edge of the disk are computed from the observations \( \{y_{i,n}\} \) using

\[
\hat{x}_{i,N} = \sum_{j=0}^{i-1} \sum_{n=0}^{N_2-1} h(i, N; j, N_2) y_{j,n}.
\]

By the orthogonality principle of linear prediction, the optimal prediction filters \( h(i, N; j, N_2) \) are computed from the covariance \( K(i, N; j, N_2) \) by solving the two-dimensional discrete Wiener-Hopf equation

\[
K(i, N; j, N_2) = h(i, N; j, N_2) + \sum_{n=0}^{N_2-1} \sum_{j=0}^{i-1} M \cdot h(i, N; j, N_2) \]

for all \( 0 \leq j < i-1 \) and \( 1 \leq N_1, N_2 \leq M \).

The goal of this paper is to derive fast algorithms for solving (3) for \( h(i, N_1; j, N_2) \) when \( K(i, N_1; j, N_2) \) has Toeplitz-plus-Hankel-block-Toeplitz-plus-Hankel structure.

For convenience in the derivation, we solve not (3) but the system of equations

\[
K(i, N_1; j, N_2) = h(i, N_1; j, N_2) + \sum_{n=0}^{N_2-1} \sum_{j=0}^{i-1} M \cdot h(i, N_1; j, N_2)
\]

for all \( -(i-1) \leq j < i-1 \) and \( 1 \leq N_1, N_2 \leq M \). This modified system (4) is motivated by noting that the continuous-parameter two-dimensional Wiener-Hopf integral equation

\[
K(x, y) = h(x, y) + \int_{|z| \leq |x|} h(x, z) K(z, y) dz
\]

\[
= h(x, y) + \int_{0}^{2\pi} \int_{0}^{\sqrt{2\pi N / M}} h(x, |z|) K(|z|, |y|) |z| d\theta d|z|,
\]

\( x, y, z \in R^2, |y| \leq |x| \)

discretizes into

\[
K'(i, N_1; j, N_2) = h'(i, N_1; j, N_2) + \sum_{n=0}^{N_2-1} \sum_{j=0}^{i-1} M \cdot h'(i, N_1; j, N_2)
\]

where the radial weighting factor \( n \) in (6) reflects \( |z| \) in (5). If we let

\[
h(i, N_1; j, N_2) = \frac{\sqrt{i-j}}{2} h'(i, N_1; j, N_2)
\]

\[
= \frac{\sqrt{i-j}}{2} h'(i, N_1; -j, N_2 + M/2)
\]

\[
K(i, N_1; j, N_2) = \frac{\sqrt{i-j}}{2} K'(i, N_1; j, N_2)
\]

\[
= \frac{\sqrt{i-j}}{2} K'(i, N_1; -j, N_2 + M/2)
\]

then the sum in (4) is simply double the sum in (6), so that if \( h(i, N_1; j, N_2) \) and \( K(i, N_1; j, N_2) \) satisfy (4), then \( h'(i, N_1; j, N_2) \) and \( K'(i, N_1; j, N_2) \) satisfy (6). Note that the second equalities in (7) and (8) will hold on a polar raster, but are not required in (4). For convenience we
continue to refer to $K(i, N_1; j, N_2)$ in (4) as the covariance function.

Similarly to the approach used in [14], we decompose the update procedure into two steps by introducing an interpolated (auxiliary) system. As shown in Fig. 1, between every pair of points in the same radial direction, we insert an auxiliary point. The covariance function $K(i, N_1; j, N_2)$ is interpolated at these auxiliary points such that the Toeplitz-plus-Hankel structure (see (13), (14)) is maintained. Then the interpolated system is defined as

$$K(i, N_1; j + 1/2, N_2)$$

$$= h(i, N_1; j + 1/2, N_2) + \sum_{n = -(i - 1/2) N_3 = 1}^{i - 1} \sum_{n = -(i - 1/2) N_3 = 1}^{i - 1/2} M$$

$$\cdot h(i, N_1; n, N_3)K(n, N_3; j + 1/2, N_2)$$

for interpolation at half-integer values of $j$ and

$$K(i + 1/2, N_1; j, N_2)$$

$$= h(i + 1/2, N_1; j + 1/2, N_2) + \sum_{n = -(i + 1/2) N_3 = 1}^{i + 1/2} \sum_{n = -(i + 1/2) N_3 = 1}^{i + 1/2} M$$

$$\cdot h(i + 1/2, N_1; n, N_3)K(n, N_3; j + 1/2, N_2)$$

for interpolation at half-integer values of $i$. Note that in (10) $j$ can also take on half-integer values.

**B. Derivation of the Basic Levinson-Like Recurrence**

Define the discrete wave operators $\Delta_r$ and $\Delta_\theta$ by

$$\Delta_r f(i, N_1; j, N_2)$$

$$= h(i + 1/2, N_1; j, N_2) + f(i + 1/2, N_1; j, N_2)$$

$$- f(i, N_1; j + 1/2, N_2) - f(i, N_1; j - 1/2, N_2)$$

$$\Delta_\theta f(i, N_1; j, N_2)$$

$$= h(i, N_1; j - 1/2, N_2) + f(i, N_1; j - 1/2, N_2)$$

$$- f(i, N_1; j, N_2 + 1)$$

$$- f(i, N_1; j, N_2 - 1)$$

where $\Delta_r$ and $\Delta_\theta$ can be regarded as discrete versions of the continuous operators $(\partial^2/\partial r^2) - (\partial^2/\partial \theta^2)$ and $(\partial^2/\partial \theta_r^2) - (\partial^2/\partial \theta^2)$ for the radial part and transverse part, respectively. In (12) $N_1 \pm 1$ and $N_2 \pm 1$ are computed mod $M$, reflecting their definition as angular variables on a polar raster.

We assume that the covariance function has Toeplitz-plus-Hankel-block-Toeplitz-plus-Hankel structure, defined by

$$\Delta_r K(i, N_1; j, N_2) = 0$$

$$\Delta_\theta K(i, N_1; j, N_2) = 0.$$ 

Applying the Laplacian operator $\Delta = \Delta_r + \Delta_\theta$ to (4), we have after some algebra

$$\Delta K(i, N_1; j, N_2)$$

$$= 0 = \Delta h(i, N_1; j, N_2) + \sum_{n = -(i - 1/2) N_3 = 1}^{i - 1/2} \sum_{n = -(i - 1/2) N_3 = 1}^{i - 1/2} M$$

$$\cdot h(i, N_1; n, N_3)K(n, N_3; j, N_2)$$

for interpolation at half-integer values of $i$. Note that in (10) $i$ can also take on half-integer values.

The algebra required to derive (15) is a generalization of the algebra in [14]; the major difference is that there are no 'end effect' terms in the sums over $N_1$ when $\Delta_\theta$ applied. This is true since $h(i, N_1; j, N_2)$ is periodic with period $M$ in $N_1$ and $N_2$, since these indices represent angles on the polar raster.

Using (13) and (14) to note that the last two terms in (15) are zero, we note that (15) has the same form as the
following linear combination:
\[
\sum_{N_1=1}^{M} \left[ V_+^i(N_1, N_2) K(i - \frac{1}{2}, N_3; j, N_2) + V_-^i(N_1, N_3) K((-i - \frac{1}{2}), N_3; j, N_2) \right] \\
\sum_{N_1=1}^{M} \left[ V_+^i(N_1, N_3) h(i - \frac{1}{2}, N_3; j, N_2) + V_-^i(N_1, N_3) h((-i - \frac{1}{2}), N_3; j, N_2) \right]
\]

\[
\sum_{n=-\left(i-\frac{1}{2}\right)}^{\left(i+\frac{1}{2}\right)} \sum_{N_4=1}^{M} [V_+^i(N_1, N_3) h(i - \frac{1}{2}, N_3; n, N_4) + V_-^i(N_1, N_3) h((-i - \frac{1}{2}), N_3; n, N_4)] K(n, N_4; j, N_2)
\]

\[
h((-i - \frac{3}{2}), N_3; n, N_4) K(n, N_4; j, N_2)
\]

where we have defined the potentials
\[
V_+^i(N_1, N_2) = -h(i, N_1; j, N_2)
\]
\[
V_-^i(N_1, N_2) = -h(i, N_1; j, N_2) + h(i + \frac{1}{2}, N_1; j, N_2) + h(i - \frac{1}{2}, N_1; j, N_2)
\]

\[
\Delta h(i, N_1; j, N_2) = \sum_{N_1=1}^{M} [V_+^i(N_1, N_3) h(i - \frac{1}{2}, N_3; j, N_2) + V_-^i(N_1, N_3) h((-i - \frac{1}{2}), N_3; j, N_2)].
\]

Equation (19) is the basic recurrence that is the heart of the Levinson-like algorithm. The left side is the difference of two two-dimensional discrete Laplacian operators, analogous to the difference of one-dimensional discrete Laplacian operators appearing in the split algorithms of [6]. The right side generalizes the three-term recurrence in [6] to a multi-term recurrence; this is analogous to the matrix recurrence in [7]. However, it is applicable to nonsymmetric block Toeplitz-plus-Hankel systems, unlike that of [7]. Writing out (19) explicitly, we have
\[
h(i + \frac{1}{2}, N_1; j, N_2) = h(i, N_1; j + \frac{1}{2}, N_2) + h(i, N_1; j - \frac{1}{2}, N_2) - h(i - \frac{1}{2}, N_1; j, N_2) + h(i - \frac{1}{2}, N_1; N_2 - 1) - h(i - \frac{1}{2}, N_1 + 1; j, N_2)
\]

\[
- h(i - \frac{1}{2}, N_1 - 1; j, N_2)
\]

\[
+ \sum_{N_3=1}^{M} [V_+^i(N_1, N_2) h(i - \frac{1}{2}, N_3; j, N_2) + V_-^i(N_1, N_2) h((-i - \frac{1}{2}), N_3; j, N_2)]
\]

for all \((-i - 3/2) \leq j \leq (i - 3/2)\) and \(1 \leq N_1, N_2 \leq M\). Although we have implicitly treated \(i\) as positive throughout the derivations, the recursive equations hold for negative \(i\) as well. When \(i\) is an integer and \(j\) is a half-integer, (20) will update \(h\) from the real points to the interpolated points. When \(i\) is a half-integer and \(j\) is an integer, (20) will update \(h\) from the interpolated points to the real points.

III. DERIVATION OF THE SCHUR-LIKE ALGORITHM

A. Derivation of the Schur-Like Recurrence

We still need to calculate the potentials \(V_+^i(N_1, N_2)\) and \(V_-^i(N_1, N_2)\) at the beginning of every update so that we can use the recursive formula (20). To do this, we introduce the Schur variables (defined at integer and half-integer points)
\[
s(i, N_1; j, N_2) = \delta_{i, N_1, j, N_2} + K(i, N_1; j, N_2) - h(i, N_1; j, N_2)
\]
\[
- \sum_{n=-\left(i-\frac{1}{2}\right)}^{\left(i+\frac{1}{2}\right)} \sum_{N_4=1}^{M} h(i, N_1; n, N_4) K(n, N_4; j, N_2)
\]

(21)

where \(\delta_{i, N_1, j, N_2} = 0\) unless \(i = j\) and \(N_1 = N_2\), in which case it is unity.

Since the Schur variables are linear combinations of the prediction error filters \(\delta_{i, N_1, j, N_2} = h(i, N_1; j, N_2)\), equations (17)–(20) show that \(s(i, N_1; j, N_2)\) satisfies the recurrence (20), but now for all \(j\):
\[
s(i + \frac{1}{2}, N_1; j, N_2) = s(i, N_1; j + \frac{1}{2}, N_2) + s(i, N_1; j - \frac{1}{2}, N_2)
\]
\[
- s(i - \frac{1}{2}, N_1; j, N_2) + s(i - \frac{1}{2}, N_1; j, N_2 + 1)
\]
\[
+ s(i - \frac{1}{2}, N_1; j, N_2 - 1) - s(i - \frac{1}{2}, N_1 - 1; j, N_2)
\]
\[
+ \sum_{N_3=1}^{M} [V_+^i(N_1, N_2) s(i - \frac{1}{2}, N_3; j, N_2) + V_-^i(N_1, N_2) s((-i - \frac{1}{2}), N_3; j, N_2)].
\]

(22)

Equation (22) is the basic recurrence for the Schur-like algorithm. Note that for \(-(i - 1) \leq j \leq (i - 1)\) \(s(i, N_1; j, N_2) = 0\) by (4).
B. Computation of Potentials

Setting \( j = (i - (1/2)) \) and \(- (i - (1/2)) \) in (22), we have

\[
\sum_{N_1=1}^{M} [V^+_i(N_1, N_2)s(i - \frac{1}{2}, N_1; i - \frac{1}{2}, N_2) + V^-_i(N_1, N_2)s(- (i - \frac{1}{2}), N_1; - (i - \frac{1}{2}), N_2)] = s(i - \frac{1}{2}, N_1; i - \frac{1}{2}, N_2) - s(i, N_1; i, N_2) + \Delta s(i, N_1; i - \frac{1}{2}, N_2)
\]

(23)

Equations (23) and (24) can be written in matrix notation as

\[
\begin{align*}
\begin{bmatrix} V^+ & V^- \\ V^+ & V^- \end{bmatrix} \begin{bmatrix} N_1, N_2 \end{bmatrix} & = \begin{bmatrix} s(i - \frac{1}{2}, N_1; i - \frac{1}{2}, N_2) \\ -s(i, N_1; i, N_2) + \Delta s(i, N_1; i - \frac{1}{2}, N_2) \end{bmatrix} \\
\end{align*}
\]

(25)

where we have defined the \( M \times M \) matrices

\[
\begin{align*}
[V^+]_{N_1,N_2} & = V^+_i(N_1, N_2); [V^-]_{N_1,N_2} = V^-_i(N_1, N_2) \\
[S^\pm]_{N_1,N_2} & = s(\pm(i - \frac{1}{2}), N_1; \pm(i - \frac{1}{2}), N_2) \\
[X^\pm]_{N_1,N_2} & = s(i - \frac{1}{2}, N_1; \pm(i - \frac{1}{2}), N_2) - s(i, N_1; i, N_2) + \Delta s(i, N_1; i - \frac{1}{2}, N_2).
\end{align*}
\]

(27)

If the system matrix defined in (4) (written explicitly in (44) below) is strongly nonsingular, i.e., the leading principal submatrices are all nonsingular, then (25) and (26) can be solved in closed form as

\[
\begin{align*}
\begin{bmatrix} V^+ & V^- \\ V^+ & V^- \end{bmatrix} \begin{bmatrix} X^+ \end{bmatrix} & = \begin{bmatrix} \Delta s(i, N_1; i - \frac{1}{2}, N_2) \end{bmatrix} \\
\end{align*}
\]

(28)

\[
\begin{align*}
\begin{bmatrix} V^+ & V^- \\ V^+ & V^- \end{bmatrix} \begin{bmatrix} X^- \end{bmatrix} & = \begin{bmatrix} \Delta s(i, N_1; i - \frac{1}{2}, N_2) \end{bmatrix} \\
\end{align*}
\]

(29)

The strongly nonsingular assumption is necessary and sufficient for (25) and (26) to have a unique solution; the proof of this is a direct generalization of the one in [14]. A similar assumption is required by the standard multi-channel Levinson and Schur algorithms.

The split Schur-like algorithm consists of computing \( s(i, N_1; j, N_2) \) by propagating (22), initialized using \( K(i, N_1; j, N_2) \), while computing the \( V_i^+(N_1, N_2) \) from the \( s(i, N_1; j, N_2) \) using (28)-(31).

C. Summary of Overall Procedure

The overall procedure can be summarized as follows. Let \( I_{\text{max}} \) be the largest radius (maximum radial prediction order). Then

1) Initialization of Split Schur-Like Algorithm:

\[
H_{\pm \frac{1}{2}, 0} = K_{\pm \frac{1}{2}, 0}^{-1}, \quad H_{\pm \frac{1}{2}, 0} = K_{\pm \frac{1}{2}, 0}^{-1}
\]

where

\[
[K_{\pm \frac{1}{2}, 0}]_{N_1,N_2} = K(\pm \frac{1}{2}, N_1; 0, N_2)
\]

\[
[K_{0,0}]_{N_1,N_2} = K(0, N_1; 0, N_2)
\]

\[
s(\pm \frac{1}{2}, N_1; j, N_2)
\]

(30)

- \[
\sum_{N_1=1}^{M} h(\pm \frac{1}{2}, N_1; 0, N_2) K(0, N_2; j, N_2)
\]

for all \( j = \pm \frac{1}{2}, \cdots, \pm I_{\text{max}} \)

\[
N_1, N_2 = 1, \cdots, M
\]

\[
s(\pm \frac{1}{2}, N_1; j, N_2) = \delta_{\pm \frac{1}{2}, N_1; j} + K(\pm \frac{1}{2}, N_1; j, N_2)
\]

(31)

- \[
h(\pm 1, N_1; j, N_2) - \sum_{N_1=1}^{M} h(\pm 1, N_1; 0, N_2) K(0, N_2; j, N_2)
\]

for all \( j = \pm 1, \cdots, \pm I_{\text{max}} \) and

\[
N_1, N_2 = 1, \cdots, M.
\]

2) Propagation of Split Schur-Like Algorithm:

A) Computation of the potentials \( V_i^+(N_1, N_2) \) and \( V_i^-(N_1, N_2) \): Compute \( V_i^+(N_1, N_2) \) and \( V_i^-(N_1, N_2) \) from the available \( s(\pm(i - (1/2)), N_1; \pm(i - (1/2)), N_2) \) and \( s(\pm i, N_1; \pm i, N_2) \) using (30) and (31);

B) Update the Schur variables:

For \( j = \pm (i + (1/2)) \) to \( j = \pm I_{\text{max}}, N_1 = 1 \) to \( M, N_2 = 1 \) to \( M, \) parallel do

Update the Schur variables using (22).

End parallel do \( \{ j, N_1, N_2 \} \).

3) Propagation of Split Levinson-Like Recurrence:

A) Propagate the boundary points:

For \( N_1 = 1 \) to \( M, N_2 = 1 \) to \( M, \) parallel do

\[
h(i + \frac{1}{2}, N_1; i - \frac{1}{2}, N_2)
\]

= \[
h(i, N_1; i - 1, N_2) - V_i^+(N_1, N_2)
\]

(32)
\begin{align*}
& h(i + \frac{1}{2}, N_1; -(i - \frac{1}{2}), N_2) \\
& = h(i, N_1; -(i - 1), N_2) - V_i^{-1} (N_1, N_2)
& \quad \text{End parallel do } \{N_1, N_2\}.
& \text{B) Propagate nonboundary points:}
& \quad \text{For } j = -(i - (1/2)) \text{ to } j = -(i - 1/2),
& \quad N_1 = 1 \text{ to } M, \text{ } N_2 = 1 \text{ to } M, \text{ parallel do}
& \quad \text{Update } h(i, N_1; j, N_2) \text{ using } (20).
& \quad \text{End parallel do } \{j, N_1, N_2\}.
& \text{4) Repeat steps 2 and 3 from } i = 1 \text{ to } I_{\text{max}}, \text{ with increment 1/2.}
& \text{Note that the Levinson and Schur recurrences (20) and (22) have identical forms, with complementary supports.}
& \text{Hence they can be propagated in parallel using identical processors; this possibility was first noted for the one-dimensional case in [16].}

\text{IV. Random Fields with Block Toeplitz-Plus-Hankel Covariances}

In the above derivation, we have assumed that the covariance function is already known. If only a sequence of two-dimensional time series data are available, there are two methods for obtaining a covariance function having the desired Toeplitz-plus-Hankel structure (13), (14). The first method is to compute a data covariance matrix, and then determine a symmetric Toeplitz-plus-Hankel-block Toeplitz-plus-Hankel matrix close (in some sense) to this matrix. This is a two-dimensional Toeplitz-plus-Hankel generalization of the well-known “Toeplitization” problem [17]. Some procedures for this problem are suggested in [18]. The second method is to assume that the data are generated by some underlying model, for which unknown parameters may need to be determined.

In this section we focus on the second approach, giving some specific examples of random fields whose covariances satisfy assumptions (13) and (14). These are merely illustrative; there are of course many others. We also note how the algorithms of Section II relate to the continuous-parameter algorithms of [13].

A. Isotropic Random Fields

For an isotropic random field, the covariance is a function of distance only, i.e., if \( x \) and \( y \) are two arbitrary points in the plane, then \( K(x, y) = K(|x - y|) \). Consider the special case of a stochastic random field with covariance \( K(x, y) = \rho^{||x - y||^2} \), which is often used in image modeling [19]. In polar coordinates on a discrete polar raster, this covariance function can be represented as

\[ K(i, N_1; j, N_2) = \rho^{\frac{1}{2} \text{log}(2 \pi (N_1 - N_2) / M)} \]

\[ = \rho^{\frac{1}{2} \text{log}(2 \pi (\pi (N_1 - N_2) / M))} \]

\[ = 1 + \frac{1}{4} \left\{ (i + j)^2 + (i - j)^2 - (i + j)^2 - (i - j)^2 \right\} \]

\[ \cdot \cos \left( 2 \pi (N_1 - N_2) \right) \text{ in } \rho \]

\[ \text{if } \rho = 1. \]

Note that the exponent has the Toeplitz-plus-Hankel structure required by (13) and (14), and that it is not merely Toeplitz in \( i \) and \( j \); hence the multichannel Levinson algorithm is not applicable. If \( \rho = 1 \), the entire covariance satisfies (13) and (14). Indeed, \( \text{any slowly changing function of distance on a polar raster satisfies (13) and (14) in its radial and angular arguments.} \]

B. Separable Covariance Functions

A separable covariance function is one that can be decomposed into multiplication of a function of the radial part and a function of the transverse part, i.e., the covariance function \( K(i, N_1; j, N_2) \) can be expressed as

\[ K(i, N_1; j, N_2) = R(i, j) \times T(N_1, N_2) \]

for some functions \( R \) and \( T \). This type of covariance function satisfies (13) and (14) as long as both \( R \) and \( T \) have Toeplitz-plus-Hankel structure. Examples include:

1) 2-D Discrete Wiener Process: The 2-D discrete Wiener process on a polar raster can be defined as

\[ x_{i, N_1} = \sum_{j=0}^{M} \sum_{n=1}^{N_2} w_{j,n}, \quad x_{0, N_1} = 0 \]

where \( w_{j,n} \) is a zero-mean discrete white noise field with variance \( \sigma^2 \). Its covariance function is equal to

\[ K(i, N_1; j, N_2) = E[x_{i,N_1}x_{j,N_2}] = M\sigma^2 \text{ min } (i, j) \]

\[ = M\sigma^2 \left( \frac{1}{2} \right) \left( |i - j| \right). \]

Note that \( R(i, j) \) has Toeplitz-plus-Hankel structure and \( T(N_1, N_2) \) is a constant function.

2) 2-D Circularly Symmetric Markovian Random Field: In a first-order 2-D circularly symmetric Markovian random field, the output is a uniformly linear combination of the previous “shell” of data plus white noise, i.e.,

\[ x_{i, N_1} = a \sum_{n=1}^{M} x_{i-1,n} + w_{i,n} \]

If \( w_{i,n} \) is assumed to be zero for all \( n \), and the variance of \( w_{i,n} \) is equal to \( \sigma^2 \), then the covariance function is

\[ K(i, N_1; j, N_2) = E[x_{i,N_1}x_{j,N_2}] \]

\[ = \frac{M\sigma^2}{1 - a^2} \left( a^{|i-j|} \right). \]

Again, \( R(i, j) \) has Toeplitz-plus-Hankel structure and \( T(N_1, N_2) \) is a constant function. In the limit \( a \to 0 \) (39) reduces to (37).

C. Relations with Continuous Algorithms

It is instructive to examine the continuous-parameter limits of some of the equations of this paper. Let the intervals between points be \( \delta_t \) in the radial direction and \( \delta_{\phi} = 2\pi / M \) rad in the transverse (angular) direction. Introducing a radial weighting factor, as discussed earlier,
and taking limits as \( \delta_6 \) and \( \delta_9 \) go to zero results in the following transformations:

1) The discretized Wiener–Hopf equation (6) becomes the Wiener-Hopf integral equation (5);
2) \( \delta_{N_1; N_2} \) becomes a continuous two-dimensional impulse function, dominating the other terms in the definition (21) of the Schur variables, so that (30) and (31) may be replaced with \( \mathbf{V}^+ = \mathbf{X}^+ \) and \( \mathbf{V}^- = \mathbf{X}^- \). Using this, (29) becomes

\[
V(x, \theta_1; \theta_2) = -\left( \frac{\partial}{\partial x} + \frac{\partial}{\partial y} \right) s(x, \theta_1; y = x, \theta_2) \quad (40)
\]

where \( x \) and \( y \) are continuous radii and \( \theta_1 \) and \( \theta_2 \) are continuous angles. Equation (40) has the form of [13, (4-17)]. Similarly, the continuous version of (13) has the form of [13, (4-2)];

3) Equation (19), with its difference of discrete two-dimensional Laplacian operators on the left side, is clearly analogous to \((\Delta_x \equiv \text{Laplacian with respect to } x)\)

\[
(\Delta_x - \Delta_y)h(x, \theta_1; y, \theta_2) = \int_0^{2\pi} V(x, \theta_1; \theta_3)h(x, \theta_3; y, \theta_2) \, d\theta_3 \quad (41)
\]

which is the two-dimensional form of [13, (4-1)]. However, (41) is not the continuous limit of (19) with radial weighting, since \((1/\sqrt{x}) (\partial^2/\partial x^2) (\partial f(x)) = ((\partial^2/\partial x^2) + (1/x)(\partial/\partial x) - (1/4x^2)) f(x)\), which is not the radial part of the 2-D Laplacian. On the other hand, \((1/x)(\partial^2/\partial x^2) (xf(x)) = ((\partial^2/\partial x^2) + (2/x)(\partial/\partial x)) f(x)\), which is the radial part of the 3-D Laplacian. This shows that the results of [13], derived for the continuous 3-D case, do not apply exactly to the 2-D case (as do the results of this paper);

4) The algorithms of this paper require the differences of the radial parts and transverse parts of the Laplacian of the covariance to be separately zero: (13) and (14) must be separately zero. However, in the continuous limit, we have \( h(i, N_1; n, N_2) \approx h(i - (1/2), N_1; n, N_2) \), and the last two sums in (11) may be combined. Then it suffices for the sum \((\Delta_x + \Delta_y)K(i, N_1; j, N_2) = 0\), rather than (13) and (14) separately. This agrees with the requirement \((\Delta_x - \Delta_y)K(x, y) = 0\) for the algorithms in [13].

**D. Application to Discretized Continuous-Parameter Problems**

We can draw some important conclusions from these observations. If the algorithms of this paper are being used to solve the discretized version (6) of the Wiener-Hopf equation (5), then

1) Equations (30) and (31) may be replaced with the approximations \( \mathbf{V}^+ = \mathbf{X}^+ \) and \( \mathbf{V}^- = \mathbf{X}^- \);

2) By the chain rule, any continuous function of the distance between two points will satisfy (13) and (14), since the square of the distance itself does. Hence the algorithms may be used for any isotropic random field. Note in particular that (34) becomes

\[
K(i, N_1; j, N_2) = \rho \delta_6 \left( e^{i\theta_1^2 + j^2 - 2i\theta_1\cos(N_1 - N_2)\delta_6} \right) \quad (42)
\]

and \( \rho \delta_6 \to 1 \) as \( \delta_6 \to 0 \);

3) Conditions (13) and (14) may be replaced with the more general condition \((\Delta_x + \Delta_y)K(i, N_1; j, N_2) = 0\).

Numerical studies have shown that approximation (2) gives very good results for \( \delta_6 = 0.001 \), but approximation (4) is much more sensitive to non-infinitesimal \( \delta_6 \).

**V. COMPLEXITY AND GENERAL TOEPLITZ-PLUS-HANKEL SYSTEMS**

**A. Computational Complexity**

We determine the number of multiplications and divisions (MAD's) needed to solve (4) up to order \( i = I_{\text{max}} \). Although some current DSP chips can perform multiplications as quickly as additions, the fact remains that multiplication is a more complex operation than addition. Also, the computational savings in the number of additions is similar to that for MAD's, although we omit details.

The initialization of the Levinson-like recurrences requires \( 2M \times M \) matrix inversions and \( 4M \times M \) matrix multiplications, or \( 2(I_{\text{max}}^2/3) + (M^2/2) \) + \( 4M^2 \) MAD's. The initialization of the Schur-like recurrences requires \( 8I_{\text{max}} M \times M \) matrix multiplications, or \( 8I_{\text{max}} M^3 \) MAD's. Each Schur-like recursion update of \( s(i, N_1; j, N_2) \) for \( i \) up to \( (1/2) \) requires \( 16(I_{\text{max}} - i)M^2 \) MAD's. Computation of the potentials requires \( 4M \times M \) matrix inversions and \( 6M \times M \) matrix multiplications. Finally, updating \( h(i, N_1; j, N_2) \) for \( i \) up to \( (1/2) \) in the Levinson-like recurrence requires \( 4(2i + 1)M^2 \) MAD's. The total number of multiplications needed to solve (4) up to \( i = I_{\text{max}} \) is

\[
4M^3 + 2 \left( \frac{M^3}{3} + \frac{M^2}{2} \right) + 8I_{\text{max}} M^3
\]

\[
+ 2 \sum_{i=1}^{I_{\text{max}}} \left[ 16(I_{\text{max}} - i)M^2 \right.
\]

\[
+ \left( 4 \left( \frac{M^3}{3} + \frac{M^2}{2} \right) + 6M^3 \right) + 4(2i + 1)M^2 \left. \right] \right.
\]

\[
= 24I_{\text{max}}^2 M^2 + I_{\text{max}} \left( 68M^3 + 4M^2 \right)
\]

\[
+ 4M^3 + 2 \left( \frac{M^3}{3} + \frac{M^2}{2} \right) \quad (43)
\]

This can be seen to be \( O(I_{\text{max}}^3 M^3) \) MAD's if \( I_{\text{max}} \gg M \gg 1 \). Solution of (4) using Gaussian elimination would require \( (2I_{\text{max}} M)^3/3 + (2I_{\text{max}} M)^2/2 = O(I_{\text{max}}^3 M^3) \) MAD's. Hence the savings in MAD's over Gaussian elimination for large \( I_{\text{max}} \) and \( M \) is a factor of order \( I_{\text{max}} M \).

**B. Comparison with Reformulation as a Block-Toeplitz System**

In [20] Merchant and Parks noted that a Toeplitz-plus-Hankel system of equations can be reformulated as a
block-Toeplitz system of equations with $2 \times 2$ blocks. Although no multichannel generalizations were discussed in [20], it is not difficult to show that a system of equations in which the system matrix is the sum of a block-Toeplitz matrix and a block-Hankel matrix, where the blocks are $M \times M$, can be reformulated as a block-Toeplitz system of equations with $2M \times 2M$ blocks. This could then be solved using the multichannel Levinson algorithm. We now compare this approach, which we call the generalized Merchant-Parks procedure, to the algorithm of this paper.

If the generalized Merchant-Parks procedure is used to solve (4) up to order $i = I_{\max}$, the number of MAD’s required is $32I_{\max}^2M^2 + I_{\max}(8M^3/3 + 2M^5)$, since $2M \times 2M$ matrices are being multiplied and propagated. Hence if $I_{\max} \gg M \gg 1$ the algorithm of this paper requires roughly $3/(4M)$ as many MAD’s as the generalized Merchant-Parks procedure; for large $M$ this can be quite significant. If $M = 1$ the algorithm of this paper reduces to that of [14], which requires roughly 75% as many MAD’s as the original Merchant-Parks procedure [20].

On the other hand, the algorithm of this paper requires that the system matrix be block-Toeplitz-plus-Hankel with Toeplitz-plus-Hankel blocks, while the generalized Merchant-Parks algorithm does not require the blocks to have special structure. Thus the generalized Merchant-Parks algorithm requires more computation, but solves a more general problem.

$$\begin{bmatrix} H_m^{+} & I + K_{-i,-i} & \ldots & K_{-i,i} \\ K_{i,-i} & \ldots & \ldots & \ldots \\ \ldots & \ldots & \ldots & \ldots \\ K_{i,-i} & \ldots & \ldots & I + K_{i,i} \end{bmatrix} = - \begin{bmatrix} S_{i,-m+1,-i} & \ldots & S_{i,-m+1,-i} & 0 & \ldots & 0 & S_{i,-m+1,i-m+1} & \ldots & S_{i,-m+1,i} \\ S_{i,-m+1,i-m+1} & \ldots & S_{i,-m+1,i-m+1} & 0 & \ldots & 0 & S_{i,-m+1,i-m+1} & \ldots & S_{i,-m+1,i} \end{bmatrix}$$

C. Solution of Arbitrary Toeplitz-plus-Hankel-Block Toeplitz-plus-Hankel Systems

Equation (4) can be written as the following Toeplitz-plus-Hankel-block-Toeplitz-plus-Hankel system:

$$\begin{bmatrix} 0 & H_{-i,-i} & \ldots & H_{-i,-i} & -I \\ -I & H_{-i,-i} & \ldots & H_{-i,-i} & 0 \\ \ldots & \ldots & \ldots & \ldots & \ldots \\ I + K_{i,-i} & \ldots & K_{i,-i} & \ldots & \ldots \\ K_{i,-i} & \ldots & \ldots & \ldots & I + K_{i,i} \end{bmatrix} = - \begin{bmatrix} S_{i,-i} & 0 & \ldots & S_{i,i} \\ S_{i,-i} & 0 & \ldots & S_{i,i} \end{bmatrix}$$

where

$$\begin{bmatrix} H_{\pm i, \pm j} \end{bmatrix} \triangleq h(\pm i, N_1; \pm j, N_2),$$

$$j = -(i - 1), \ldots (i - 1), \hspace{1cm} 1 \leq N_1, N_2 \leq M$$

and

$$\begin{bmatrix} K_{j,i} \end{bmatrix} \triangleq K(j, N_1; i, N_2),$$

$$j, i = -(i - 1), \ldots (i - 1), \hspace{1cm} 1 \leq N_1, N_2 \leq M$$

In (44)-(47) $I$ is the $M \times M$ identity matrix and $0$ is an $M \times M$ matrix of zeros. Conditions (13) and (14) are equivalent to requiring that the system matrix in (44) be block-Toeplitz-plus-Hankel with Toeplitz-plus-Hankel blocks.

In this section we solve a Toeplitz-plus-Hankel-block Toeplitz-plus-Hankel system of equations having the same system matrix as (44), but with an arbitrary right side. This system is

$$[X_{-i} \ X_{-(i-1)} \ \cdots \ X_{-1} \ X_i]$$

$$= \begin{bmatrix} I + K_{-i,-i} & \ldots & K_{-i,i} \\ \ldots & \ldots & \ldots & \ldots \\ K_{i,-i} & \ldots & \ldots & I + K_{i,i} \end{bmatrix} \begin{bmatrix} B_{-i} \\ \ldots \\ B_{-(i-1)} \ \cdots \ B_{-i} \ B_i \end{bmatrix}$$

where the right side is arbitrary. Recall that the algorithms of this paper do not require the system matrix to be symmetric. To find the solution $X \triangleq [X_{-i} \ \cdots \ X_i]$, note that from the definition (21) of $s(iN_1; j, N_2)$ we have

$$\begin{bmatrix} K_{j,i} \end{bmatrix}_{N_1, N_2} \triangleq h(\pm i, N_1; \pm j, N_2),$$

$$j, i = -(i - 1), \ldots (i - 1), \hspace{1cm} 1 \leq N_1, N_2 \leq M$$

$$[S_{\pm i, \pm j}]_{N_1, N_2} \triangleq s(\pm i, N_1; \pm j, N_2),$$

$$1 \leq N_1, N_2 \leq M.$$
\[ \hat{\mathbf{x}} = \sum_{m = -1, m \neq 0}^i C_m \mathbf{H}_m \quad (\text{i.e., } X_j = \sum_{i = -j}^i C_i H_{i,j}) \]

(52)

Here \( C_m \) can be found by equating the linear combination (52) to (48), for \( 1 \leq j \leq (i-1) \):

\[
C_j S_{j,-j} + C_j S_{j,-j} = -\left( B_j + \sum_{\sigma = -1}^{j-1} C_\sigma S_{\sigma,-j} \right)
\]

(53)

\[
C_j S_{j,j} + C_j S_{j,j} = -\left( B_j + \sum_{\sigma = -1}^{j-1} C_\sigma S_{\sigma,j} \right)
\]

(54)

The overall procedure is as follows. Compute the \( \mathbf{H}_{i,j} \) and \( \mathbf{S}_{i,j} \) using the Levinson-like and Schur-like algorithms. Next, recursively compute \( C_m \) in increasing \( j \) by solving the \( 2M \times 2M \) systems (53) and (54). Finally, compute \( \hat{\mathbf{x}} \) using (52).

The procedures in (52)-(54) require roughly \( 4I_{\text{max}}^2 M^2 \) MAD’s, which for \( I_{\text{max}} \gg M \gg 1 \) dominates the \( 24I_{\text{max}}^2 M^2 \) MAD’s that are the dominant term in the number of MAD’s (43) required by the basic algorithm. For an arbitrary right side, the generalized Merchant–Parks algorithm requires \( 48I_{\text{max}}^2 M^3 \) MAD’s. Thus the algorithm of this section requires only 1/12 as many MAD’s when \( I_{\text{max}} \gg M \gg 1 \).

VI. CONCLUSION

New fast algorithms for solving the discrete two-dimensional Wiener–Hopf equation on a polar raster when the covariance function has Toeplitz-plus-Hankel structure have been derived. Since we have performed explicitly discrete derivations, instead of just discretizing the continuous versions [13], the algorithms do not require fine discretization or closely-spaced points; if adjacent points are close enough, then the algorithms reduce to the continuous case [13]. In particular, the proposed fast algorithms make full use of the Toeplitz-plus-Hankel structure of the covariance function, so that the overall computational complexity is only \( O(I_{\text{max}}^2 M^2) \) MAD’s, as opposed to \( O(I_{\text{max}}^3 M^3) \) MAD’s for the generalized Merchant–Parks algorithm discussed in the paper and \( O(I_{\text{max}}^3 M^3) \) MAD’s for Gaussian elimination. These algorithms are also highly parallelizable, making them even more favorable in a vector/parallel processor environment.

The smoothing filter for estimating the points inside the disk of observations can be computed from the prediction filters using a generalized discrete Bellman–Siegel–Krein identity, as was done for the one-dimensional continuous case in [9]. The overall complexity is reduced compared with Gaussian elimination. This is considered in the separate paper [8].

Unresolved issues include mapping of this algorithm into optimal array processor architectures, the numerical stability of the algorithm, and practical applications of this algorithm in problems such as image restoration and coding. Preliminary results on these issues have been encouraging.

REFERENCES


Wen-Hsien Fang was born in Taipei, Taiwan, Republic of China, in 1961. He received the B.S. degree from the National Taiwan University in 1983, and the M.S.E. degree from the University of Michigan, Ann Arbor, in 1988. He is currently working toward the Ph.D. degree in the Department of Electrical Engineering and Computer Science at the University of Michigan, Ann Arbor.

From 1983 to 1985, he was in the Army where he served as a Battery Control Officer. Since 1988, he has been a Research Assistant at the University of Michigan. His research interests include fast algorithms for signal processing and their VLSI hardware implementations, adaptive filtering, and speech and image coding.

Andrew E. Yagle (M'85) was born in Ann Arbor, MI, in 1956. He received the B.S.E. and B.S.E.E. degrees from the University of Michigan, Ann Arbor, in 1977 and 1978, respectively, and the S.M., E.E., and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1981, 1982, and 1985, respectively.

Since September 1985 he has been an Assistant Professor in the Department of Electrical Engineering and Computer Science, the University of Michigan, Ann Arbor. He has published over 20 papers on fast algorithms for inverse scattering problems, linear least squares estimation, and signal processing.

Professor Yagle is a member of the Society of Exploration Geophysicists, the Acoustical Society of America, Tau Beta Pi, and Sigma Xi. He received the NSF Presidential Young Investigator Award in 1988 and the ONR Young Investigator Award in 1990. He is currently an Associate Editor for the IEEE Transactions on Signal Processing.