Two-Dimensional Capon Spectrum Analysis *

Andreas Jakobsson† S. Lawrence Marple, Jr.‡ Petre Stoica†

Abstract

We present a computationally efficient algorithm for computing the 2-D Capon spectral estimator. The implementation is based on the fact that the 2-D data covariance matrix will have a *Toeplitz-Block-Toeplitz structure*, with the result that the inverse covariance matrix can be expressed in closed form by using a special case of the Gohberg-Heinig formula that is a function of strictly the forward 2-D prediction matrix polynomials. Furthermore, we present a novel method, based on a 2-D lattice algorithm, to compute the needed forward prediction matrix polynomials and discuss the difference in the so-obtained 2-D spectral estimate as compared to the one obtained by using the prediction matrix polynomials given by the Whittle-Wiggins-Robinson algorithm. Numerical simulations illustrate the clear computational gain in comparison to both the well-known classical implementation and the method recently published by Liu *et al.*

*This work was supported in part by the Swedish Foundation for Strategic Research. Corresponding author: Andreas Jakobsson. Phone +46 18 4717846. Fax +46 18 503611. Email: aj@syscon.uu.se

†Andreas Jakobsson and Petre Stoica are with the Dept of Systems and Control, Uppsala University, P.O. Box 27, SE-751 03 Uppsala, Sweden.

‡S. Lawrence Marple, Jr. is with Orincon Corp., 9363 Towne Centre Drive, San Diego, CA 92121, USA.

1 Introduction

The problem of two-dimensional (2-D) high resolution spectral estimation has been widely studied in the past literature (see, e.g., [1, 2, 3]), as well as in more recent contributions such as [4, 5, 6]. Applications occur in a wide variety of fields, such as geophysics, radio astronomy, biomedical engineering, sonar, and radar, to mention a few. In many of these applications, it is of key importance to obtain computationally efficient high resolution estimates, as for example it is in synthetic aperture radar (SAR) image formation and target feature extraction [7, 8], which are becoming increasingly important in many civilian and military applications [9]. Another important application is 2-D nuclear magnetic resonance (NMR) spectral estimation, where both resolution and computational complexity are of uttermost importance [10]. Popular approaches include the 2-D Periodogram, and in the higher resolution cases, the 2-D AR and the 2-D Capon spectral estimators. A number of approaches has been suggested for efficient estimation of the 2-D AR spectrum (see, e.g., [11, 12, 3]), whereas only limited efforts have been made to simplify the 2-D Capon estimator [5, 3].

In this work we present an efficient implementation of the 2-D Capon spectral estimator, which will depend on the (forward) prediction matrices and the (forward) prediction error covariance matrix. The derivation is based on a special case of the Gohberg-Heinig formula [13] (see also [14]) for the inverse of a Block-Toeplitz matrix. This closed form expression of the inverse covariance matrix enables significant computational reduction to form the 2-D Capon spectral estimator due to the highly structured problem formulation, and the algorithm can be seen as a 2-D extension of the algorithm derived by Musicus [15] for the 1-D case. The final form of the algorithm was proposed, although the algorithm was neither specified nor derived, in [3]. Here, we present the full derivation of the algorithm as well as a complexity comparison with the standard 2-D Capon approach as well as with the recently published implementation by Liu *et al* [5].

Furthermore, we present a novel 2-D lattice algorithm to estimate the needed (forward) prediction matrices and the (forward) prediction error covariance matrix. The presented algorithm is an improvement over the previous attempts to extend the linear prediction lattice-based 1-D techniques to two dimensions [16, 11, 12, 17, 18]. Following the 1-D Burg recursion procedure [19], Strand [16] extended the Whittle-Wiggins-Robinson Algorithm (WWRA) [20, 21] to compute the AR parameters directly from the multichannel data without first computing the autocorrelation sequence. Therrien and El-Shaer [12] used this technique to compute the 2-D AR spectral

estimates directly from the data by treating the 2-D signal as being made of strips of multichannel signals. In [17], Kuduvalli and Rangayyan further simplified Therrien and El-Shaer's algorithm by forcing the structure of the 2-D autocorrelation matrices on the multichannel version of the Burg algorithm. A similar approach was used by McGuffin and Liu in [11].

Our improvement makes further use of the matrix structure, yielding an alternative minimization criterion. In the 1-D case, the algorithm will reduce to the well-known Burg algorithm. In the numerical section, we compare the computational complexity of the 2-D lattice algorithm with alternative methods for estimating the prediction error matrices. We also compare the relative quality of the spectral estimates obtained from these estimates. It is found that the spectral estimates obtained by using the new 2-D lattice algorithm are noticeably better than the ones based on using the WWRA with estimates of the 2-D covariance matrix.

The paper is organized as follows. In Section 2 we review the classical 2-D Capon spectral estimator. Next, we describe the suggested efficient algorithm in Section 3. In Section 4 we review the extended Yule-Walker equations, and in Section 5 we introduce the novel 2-D lattice algorithm. Section 6 contains our numerical examples and, finally, Section 7 contains our conclusions.

2 Problem Formulation

In this section, we will briefly review the well-known 2-D Capon spectral estimation method and formulate the problem of interest. The Capon spectral estimator was, unlike many other spectral estimation techniques, originally proposed directly in a multi-dimensional setting as an array-processing technique [22, 23]. It is a non-parametric adaptive matched-filterbank approach [24, 4], and follows two main steps:

a) Pass the data through a 2-D bandpass filter with varying center frequencies (ω_1, ω_2), and

b) estimate the power at (ω_1, ω_2) for all $\omega_1 \in [0, 2\pi), \omega_2 \in [0, 2\pi)$ of interest from the filtered data.

Let the $N_1 \times N_2$ data matrix $\mathbf{Z}$ be a part of a 2-D stationary discrete data sequence, and assume that the bandpass filter used, $\mathbf{H}_{\omega_1,\omega_2}$, is an $(M_1+1) \times (M_2+1)$-tap 2-D finite impulse response (FIR) filter. The filterlengths M_1 and M_2 are parameters specified by the user.

Define the $(M_1+1)(M_2+1)\times 1$ filter vector

$$\mathbf{h}_{\omega_1,\omega_2} \stackrel{\triangle}{=} \mathrm{vec}(\mathbf{H}_{\omega_1,\omega_2}), \tag{1}$$

where $\mathrm{vec}(\cdot)$ denotes the operation of stacking the columns of a matrix on top of each other. Form the $(M_1+1)\times(M_2+1)$ submatrices

$$\mathbf{Y}_{t,s} = \begin{bmatrix} \mathbf{Z}_{t,s} & \dots & \mathbf{Z}_{t,s+M_2} \\ \vdots & \ddots & \vdots \\ \mathbf{Z}_{t+M_1,s} & \dots & \mathbf{Z}_{t+M_1,s+M_2} \end{bmatrix} \tag{2}$$

for $t=0,\dots,L_1, s=0,\dots,L_2$, where $L_1=N_1-M_1-1, L_2=N_2-M_2-1$, and define the $(M_1+1)(M_2+1)\times 1$ snapshot vector

$$\mathbf{y}_{t,s} = \mathrm{vec}(\mathbf{Y}_{t,s}). \tag{3}$$

Furthermore, define the covariance matrix, $\mathbf{R}$, as

$$\begin{aligned} \mathbf{R} &\stackrel{\triangle}{=} E\left\{\mathbf{y}_{t,s}\mathbf{y}_{t,s}^*\right\} \\ &= \begin{bmatrix} \mathbf{R}_0 & \mathbf{R}_1 & \dots & \mathbf{R}_{M_1} \\ \mathbf{R}_1^* & \mathbf{R}_0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{R}_1 \\ \mathbf{R}_{M_1}^* & \dots & \mathbf{R}_1^* & \mathbf{R}_0 \end{bmatrix}, \end{aligned} \tag{4}$$

where $E\{\cdot\}$ and $(\cdot)^*$ denote the expectation and the complex conjugate transpose, and where the block matrices $\mathbf{R}_k$ are

$$\mathbf{R}_k = \begin{bmatrix} r_{k,0} & r_{k,1} & \dots & r_{k,M_2} \\ r_{k,1}^* & r_{k,0} & \ddots & \vdots \\ \vdots & \ddots & \ddots & r_{k,1} \\ r_{k,M_2}^* & \dots & r_{k,1}^* & r_{k,0} \end{bmatrix}, \tag{5}$$

where

$$r_{k,l} \stackrel{\triangle}{=} E\left\{\mathbf{Z}_{t+k,s+l}\ \mathbf{Z}_{t,s}^*\right\} = r_{-k,-l}^*. \tag{6}$$

It can be seen from (4) and (5) that the $(M_1+1)(M_2+1)\times(M_1+1)(M_2+1)$ covariance matrix $\mathbf{R}$ has a (Hermitian) Toeplitz-Block-Toeplitz structure.

The adaptive Capon bandpass filter is designed to minimize the power of the filter output, as well as pass the frequencies (ω_1, ω_2) without any attenuation, i.e., to satisfy, for each (ω_1, ω_2),

$$\min_{\mathbf{h}} \mathbf{h}^*_{\omega_1,\omega_2} \mathbf{R} \mathbf{h}_{\omega_1,\omega_2} \quad \text{subject to} \quad \mathbf{h}^*_{\omega_1,\omega_2} \mathbf{a}_{\omega_1,\omega_2} = 1, \tag{7}$$

where $\mathbf{a}_{\omega_1,\omega_2}$ is the 2-D Fourier vector, defined as

$$\mathbf{a}_{\omega_1,\omega_2} \triangleq \mathbf{a}_{\omega_1} \otimes \mathbf{a}_{\omega_2} \tag{8}$$

$$\mathbf{a}_{\omega_k} \triangleq \begin{bmatrix} 1 & e^{-i\omega_k} & \dots & e^{-iM_k\omega_k} \end{bmatrix}^T. \tag{9}$$

Hereafter, $\otimes$ and $(\cdot)^T$ denote the Kronecker product and the transpose, respectively. When the filter minimizing (7) is inserted into the expression for the power of the filter output, for the bandpass filter centered at the frequencies of interest, (ω_1, ω_2), i.e.,

$$\mathbf{S}_{\omega_1,\omega_2} = \mathbf{h}^*_{\omega_1,\omega_2} \mathbf{R} \mathbf{h}_{\omega_1,\omega_2}, \tag{10}$$

the Capon spectral estimate at (ω_1, ω_2) is obtained as

$$\hat{\mathbf{S}}_{\omega_1,\omega_2} = \frac{1}{\mathbf{a}^*_{\omega_1,\omega_2} \mathbf{R}^{-1} \mathbf{a}_{\omega_1,\omega_2}}. \tag{11}$$

The problem of interest is to compute (11) efficiently. We note that, the primary computational burden to evaluate (11) is not, as it might first seem, that associated with the inverse of the covariance matrix $\mathbf{R}$. If that were the case, an efficient algorithm for the inversion of a Toeplitz-Block-Toeplitz matrix could have been used [25, 26]. Rather, it is the computation of (11) over all frequencies that is normally more time consuming. In the following section, we propose an efficient way to do this using the 2-D Fast Fourier Transform.

3 Proposed Algorithm

By making use of the fact that the inverse of a Block-Toeplitz matrix can be formulated in a closed-form expression by using the Gohberg-Heinig formula as well as the relationship between the forward and backward prediction matrix polynomials that holds for Toeplitz-Block-Toeplitz covariance matrices, we are now ready to state the first main result of this paper. For the reader's convenience, the Gohberg-Heinig formula is summarized in Appendix A.

Theorem 1
The denominator of the 2-D Capon spectral estimator

$$f_{\omega_1,\omega_2} \stackrel{\triangle}{=} \mathbf{a}^*_{\omega_1,\omega_2}\mathbf{R}^{-1}\mathbf{a}_{\omega_1,\omega_2}, \tag{12}$$

can be equivalently expressed as

$$f_{\omega_1,\omega_2} = \sum_{s=-M_1}^{M_1}\sum_{p=-M_2}^{M_2}\theta(s,p)e^{-i\omega_1 s}e^{-i\omega_2 p} \tag{13}$$

where $\theta(s,p) = \bar{\theta}(-s,-p)$,

$$\theta(s,p) = \sum_{k=0}^{M_1-s}(M_1+1-2k-s)\,\Psi(s,p,k) \tag{14}$$

$$\Psi(s,p,k) = \sum_{l=\max(0,-p)}^{\min(M_2-p,M_2)}\left[\mathbf{A}_k^*\mathbf{Q}^{-1}\mathbf{A}_{k+s}\right]_{l,l+p} \tag{15}$$

and where $\{\mathbf{A}_i\}$ *and* $\mathbf{Q}$ *denote the (forward) prediction matrices and the (forward) prediction error covariance matrix as defined in (17), with* $\mathbf{R}$ *being given by (4) and* $n = M_1$. *Here,* $\bar{x}$ *and* $[\mathbf{X}]_{i,j}$ *denote the complex conjugate of* x *and the* i,j*:th entry of the matrix* $\mathbf{X}$.

Proof: See Appendix B. □

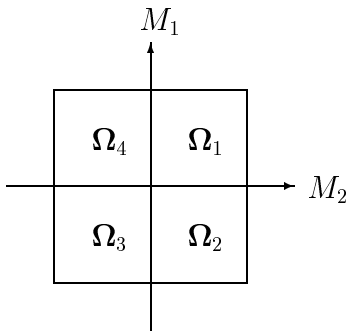


Figure 1: The support of the θ parameters

It is worth noting that the expression for f_{ω_1,ω_2} in (13) can be efficiently computed by making use of the periodic properties of the 2-D FFT algorithm. Denote the support of the θ parameters with $\mathbf{\Omega}_k$ as is shown in Figure 1, i.e.,

$$\mathbf{\Omega}_1 = \begin{bmatrix} \theta_{M_1,0} & \dots & \theta_{M_1,M_2} \\ \vdots & \ddots & \vdots \\ \theta_{0,0} & \dots & \theta_{0,M_2} \end{bmatrix} \qquad \mathbf{\Omega}_2 = \begin{bmatrix} \theta_{-1,0} & \dots & \theta_{-1,M_2} \\ \vdots & \ddots & \vdots \\ \theta_{-M_1,0} & \dots & \theta_{-M_1,M_2} \end{bmatrix}$$

$$\mathbf{\Omega}_3 = \begin{bmatrix} \theta_{-1,-M_2} & \dots & \theta_{-1,-1} \\ \vdots & \ddots & \vdots \\ \theta_{-M_1,-M_2} & \dots & \theta_{-M_1,-1} \end{bmatrix} \qquad \mathbf{\Omega}_4 = \begin{bmatrix} \theta_{M_1,-M_2} & \dots & \theta_{M_1,-1} \\ \vdots & \ddots & \vdots \\ \theta_{0,-M_2} & \dots & \theta_{0,-1} \end{bmatrix}.$$

Next, introduce the $N_{\omega_1} \times N_{\omega_2}$ $\tilde{\mathbf{\Omega}}$, where N_{ω_1} and N_{ω_2} denote the number of frequencies for which the spectrum should be computed, and let

$$\tilde{\mathbf{\Omega}} = \begin{bmatrix} \mathbf{\Omega}_2 & \mathbf{0} & \mathbf{\Omega}_3 \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{\Omega}_1 & \mathbf{0} & \mathbf{\Omega}_4 \end{bmatrix}. \tag{16}$$

Note that the different zero matrices, $\mathbf{0}$, in (16) have different dimensions. A positive real-valued f_{ω_1,ω_2} can then be efficiently computed as the $N_{\omega_1} \times N_{\omega_2}$ 2-D FFT of $\tilde{\mathbf{\Omega}}$.

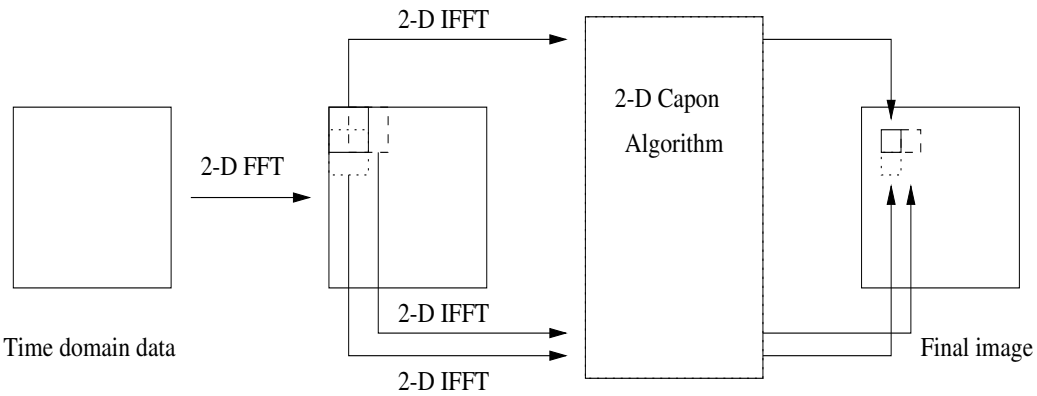


Figure 2: Applying the 2-D Capon algorithm to a data matrix with large dimensions.

It is also worthwhile to note that, as the algorithm in Theorem 1 requires the computation of the $(M_2+1)\times(M_2+1)$ $\{\mathbf{A}_i\}$ and $\mathbf{Q}$ matrices from the data, as well as the matrix inversion of the $\mathbf{Q}$ matrix in (15), it may be computationally prohibitive to apply the algorithm directly to data matrices with very large dimensions. One can then proceed in a different way as was suggested in [27]. Break up each frequency domain image (obtained by taking the 2-D FFT of the time domain data) into overlapping chips of size $N_{s_1}\times N_{s_2}$ (where N_{s_1} and N_{s_2} are smaller than the original data dimensions), as shown in Figure 3. Take the 2-D inverse FFT (IFFT) of the frequency domain chips to obtain the time domain chips and apply the algorithm in Theorem 1 to the time domain chips. This approach will not produce any mosaicing or tiling effect.

As (15) depends on the (forward) prediction matrices and the (forward) prediction error covariance matrix, the problem of computing these still remains. In the following two sections we first review the extended Yule-Walker solution and then introduce the novel 2-D lattice algorithm for computing $\{\mathbf{A}_i\}$ and $\mathbf{Q}$ from the data.

4 The Extended Yule-Walker Equations

The extended Yule-Walker equations can be expressed as [3, 28]

$$\begin{aligned}
\left[\begin{array}{cccc}\mathbf{I} & \mathbf{A}_1 & \ldots & \mathbf{A}_n\end{array}\right]\mathbf{R} &\stackrel{\triangle}{=} \left[\begin{array}{cccc}\mathbf{Q} & \mathbf{0} & \ldots & \mathbf{0}\end{array}\right] \\
\left[\begin{array}{cccc}\mathbf{B}_n & \ldots & \mathbf{B}_1 & \mathbf{I}\end{array}\right]\mathbf{R} &\stackrel{\triangle}{=} \left[\begin{array}{cccc}\mathbf{0} & \ldots & \mathbf{0} & \mathbf{S}\end{array}\right]
\end{aligned} \tag{17}$$

where $\mathbf{R}$ is defined as in (4), with $n = M_1$. In the case that $\mathbf{R}$ has a Toeplitz-Block-Toeplitz structure, it is known that the backward prediction matrices and the backward prediction error covariance matrix can be found as [3, 11, 29]

$$\mathbf{B}_i = \mathbf{J}\bar{\mathbf{A}}_i\mathbf{J} \tag{18}$$

$$\mathbf{S} = \mathbf{J}\bar{\mathbf{Q}}\mathbf{J} \tag{19}$$

where $\mathbf{J}$ denotes the so-called exchange matrix, whose anti-diagonal elements are ones and all other elements are zero, i.e.,

$$\mathbf{J} = \begin{bmatrix} 0 & \ldots & 0 & 1 \\ 0 & \ldots & 1 & 0 \\ \vdots & \iddots & \iddots & \vdots \\ 1 & 0 & \ldots & 0 \end{bmatrix}. \tag{20}$$

In fact, the above identities will hold if $\mathbf{R}$ is simply persymmetric [3], i.e.,

$$\mathbf{R} = \mathbf{J}\mathbf{R}^T\mathbf{J}. \tag{21}$$

Several approaches to solve the extended Yule-Walker equations in (17), making use of the identities in (18) and (19), have been proposed in the literature (see, e.g., [3, 11, 29]), improving computationally on the Whittle-Wiggins-Robinson algorithm (WWRA). The resulting algorithms reduce the computational burden by approximately one-half as compared to the WWRA. However, this will hold only for larger matrices, whereas for smaller matrices of the dimensions considered in this paper it might still be more efficient to use the WWRA. For this reason, we have in the numerical section only used the WWRA. For easy reference the WWRA is summarized in Appendix C.

As a first step in using any of these algorithms, the covariance matrix, $\mathbf{R}$, needs to be estimated. Just as in the 1-D case, there are several different approaches to do so, which will yield different estimates of the prediction matrices. In this paper we will consider and compare the following different approaches of estimating the covariance matrix:

- The Toeplitz-Block-Toeplitz estimate.
- The forward-backward (FB) Block-Toeplitz estimate.
- The outer product estimate.

For completeness, we will in the following subsections quickly review the different estimates.

4.1 The Toeplitz-Block-Toeplitz Estimate

The Toeplitz-Block-Toeplitz covariance matrix estimate is obtained by building the full matrix from the autocorrelation estimates, $\hat{r}_{k,l}$, as defined in (6). These are obtained as

$$\hat{r}_{k,l} = \begin{cases} \frac{1}{N_1 N_2} \sum_{m=0}^{N_1-1-k} \sum_{n=0}^{N_2-1-l} \mathbf{Z}_{m+k,n+l}\mathbf{Z}^*_{m,n} & \text{for } k \geq 0, l \geq 0 \\ \frac{1}{N_1 N_2} \sum_{m=0}^{N_1-1-k} \sum_{n=-l}^{N_2-1} \mathbf{Z}_{m+k,n+l}\mathbf{Z}^*_{m,n} & \text{for } k \geq 0, l < 0 \\ \hat{r}^*_{-k,-l} & \text{for } k \leq 0, \text{any } l \end{cases}$$

over a lag range of $|k| \leq M_1$ and $|l| \leq M_2$.

4.2 The FB Block-Toeplitz Estimate

The FB Block-Toeplitz covariance matrix is obtained by estimating the forward-backward block sample covariance matrices, $\hat{\mathbf{R}}_k^{fb}$, and then proceeding by building the full covariance matrix using its Block-Toeplitz structure to obtain $\hat{\mathbf{R}}^{fb}$. The k:th block forward covariance matrix is estimated as (for $k = 0, \ldots, M_1$)

$$\hat{\mathbf{R}}_k^f = \frac{1}{N_1 N_2} \sum_{t=0}^{N_1-1-k} \sum_{s=0}^{N_2-1-M_2} \begin{bmatrix} \mathbf{Z}_{t,s} \\ \vdots \\ \mathbf{Z}_{t,s+M_2} \end{bmatrix} \begin{bmatrix} \mathbf{Z}_{t+k,s}^* & \cdots & \mathbf{Z}_{t+k,s+M_2}^* \end{bmatrix}.$$

The forward-backward sample covariance matrix is obtained as

$$\hat{\mathbf{R}}_k^{fb} = \frac{1}{2} \left(\hat{\mathbf{R}}_k^f + \mathbf{J} \left(\hat{\mathbf{R}}_k^f \right)^T \mathbf{J} \right). \tag{22}$$

Note that, although $\hat{\mathbf{R}}_k^{fb}$ in (22) is not Toeplitz, it is persymmetric which also makes $\hat{\mathbf{R}}^{fb}$ persymmetric. Since $\mathbf{R}_k$ is persymmetric, one would expect that $\hat{\mathbf{R}}_k^{fb}$ is a better estimate of $\mathbf{R}_k$ than the non-persymmetric $\hat{\mathbf{R}}_k^f$. We refer the reader to [30] for an analysis and a comparison between the use of the forward-only and the forward-backward sample covariance matrices for the 1-D Capon spectral estimator. Similar conclusions are expected to hold in the 2-D case considered here.

4.3 The Outer Product Estimate

The outer product sample covariance matrix is obtained by estimating the covariance matrix without taking its internal structure into account, i.e.,

$$\hat{\mathbf{R}} = \frac{1}{L_1 L_2} \sum_{k=0}^{L_1} \sum_{l=0}^{L_2} \mathrm{vec}(\mathbf{Y}_{k,l}) \, \mathrm{vec}(\mathbf{Y}_{k,l})^*.$$

5 The 2-D Lattice Algorithm

In this section we present a novel 2-D lattice algorithm, that obtains $\{\mathbf{A}_i\}$ and $\mathbf{Q}$ directly from the data $\{\mathbf{Z}_{k,l}\}$. This algorithm, which does not depend on an estimate of the covariance matrix $\mathbf{R}$, is an improvement over previous attempts to extend the linear prediction lattice-based 1-D techniques to two dimensions [16, 11, 17, 18]. In the 1-D case, the algorithm will reduce to the well-known Burg algorithm [19].

The generalized $(M_2+1)\times(N_2-M_2)$ 2-D forward and backward linear prediction errors for a n:th order model at sample index k, for $n \le k \le N_1-1$, are defined as

$$\tilde{\boldsymbol{\epsilon}}_n^f(k) \triangleq \mathbf{X}_k + \sum_{l=1}^{n} \mathbf{A}_l^{(n)} \mathbf{X}_{k-l} \tag{23}$$

$$\tilde{\boldsymbol{\epsilon}}_n^b(k) \triangleq \mathbf{X}_{k-n} + \sum_{l=1}^{n} \left(\mathbf{J}\bar{\mathbf{A}}_l^{(n)}\mathbf{J}\right) \mathbf{X}_{k-n+l}, \tag{24}$$

where $\{\mathbf{A}_i\}$ are the $(M_2+1)\times(M_2+1)$ forward prediction matrices defined in (17), and where the $(M_2+1)\times(N_2-M_2)$ Toeplitz data matrix at index k is defined as

$$\mathbf{X}_k \triangleq \begin{bmatrix} \mathbf{Z}_{k,M_2} & \cdots & \mathbf{Z}_{k,2M_2} & \cdots & \mathbf{Z}_{k,N_2-1} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ \mathbf{Z}_{k,0} & \cdots & \mathbf{Z}_{k,M_2} & \cdots & \mathbf{Z}_{k,N_2-M_2-1} \end{bmatrix}, \tag{25}$$

where the dimensional constraint $N_2 - M_2 \ge M_2 + 1$ enables the product $\mathbf{X}_k\mathbf{X}_k^*$ to be non-singular. Note that at order $n=0$, $\tilde{\boldsymbol{\epsilon}}_0^f(k) = \tilde{\boldsymbol{\epsilon}}_0^b(k) = \mathbf{X}_k$. Define the estimated 2-D forward and backward linear prediction squared errors as

$$\mathbf{P}_n^f \triangleq \frac{1}{N_1-n} \sum_{k=n}^{N_1-1} \tilde{\boldsymbol{\epsilon}}_n^f(k) \left(\tilde{\boldsymbol{\epsilon}}_n^f(k)\right)^* \tag{26}$$

$$\mathbf{P}_n^b \triangleq \frac{1}{N_1-n} \sum_{k=n}^{N_1-1} \tilde{\boldsymbol{\epsilon}}_n^b(k) \left(\tilde{\boldsymbol{\epsilon}}_n^b(k)\right)^*. \tag{27}$$

A recursive 2-D lattice relationship may be developed between the 2-D forward and backward linear prediction matrix errors by using the order update of the WWRA, for $k = 1, \ldots, n$, (c.f. (C.1) and (18))

$$\mathbf{A}_k^{(n+1)} = \mathbf{A}_k^{(n)} + \mathbf{K}_{n+1}\left(\mathbf{J}\bar{\mathbf{A}}_{n+1-k}^{(n)}\mathbf{J}\right) \tag{28}$$

where $\mathbf{K}_{n+1} = \mathbf{A}_{n+1}^{(n+1)}$, the so-called 2-D reflection coefficient matrix [31], is the gain coefficient matrix of the 2-D generalized lattice filter. By substituting (28) into (23) and (24), we obtain

$$\begin{aligned} \tilde{\boldsymbol{\epsilon}}_{n+1}^f(k) &= \tilde{\boldsymbol{\epsilon}}_n^f(k) + \mathbf{K}_{n+1}\tilde{\boldsymbol{\epsilon}}_n^b(k-1) \\ \tilde{\boldsymbol{\epsilon}}_{n+1}^b(k) &= \tilde{\boldsymbol{\epsilon}}_n^b(k-1) + \left(\mathbf{J}\bar{\mathbf{K}}_{n+1}\mathbf{J}\right)\tilde{\boldsymbol{\epsilon}}_n^f(k). \end{aligned} \tag{29}$$

Making use of (17), as well as the identities (18) and (19), we note that

$$\begin{aligned} E\left(\mathbf{P}_n^f\right) &= \left[\begin{array}{cccc} \mathbf{I} & \mathbf{A}_1^{(n)} & \ldots & \mathbf{A}_n^{(n)} \end{array}\right] \mathbf{R} \left[\begin{array}{cccc} \mathbf{I} & \mathbf{A}_1^{(n)} & \ldots & \mathbf{A}_n^{(n)} \end{array}\right]^* \\ &= E\left(\mathbf{J}\bar{\mathbf{P}}_n^b\mathbf{J}\right), \end{aligned} \tag{30}$$

where $\mathbf{R}$ is given in (4), with $n = M_1$. From (30), it can be seen that both $E(\mathbf{P}_n^f)$ and $E(\mathbf{J}\bar{\mathbf{P}}_n^b\mathbf{J})$ will have the prediction error autocovariances along their main diagonals. Thus, a least squares estimate of $\mathbf{K}_{n+1}$ can be obtained by minimizing

$$\mathrm{tr}\left\{\mathbf{P}_{n+1}^{f+b}\right\}, \tag{31}$$

where

$$\mathbf{P}_n^{f+b} \triangleq \mathbf{P}_n^f + \mathbf{J}\bar{\mathbf{P}}_n^b\mathbf{J}, \tag{32}$$

as this will be equivalent with minimizing the sum of the prediction error autocovariances. This should be compared with the minimization criteria used in [11, 17], where $\mathrm{tr}(\mathbf{P}_n^f + \mathbf{P}_n^b)$ is instead minimized.

After substituting (29) into (26) and (27), the minimization in (31) only depends on the gain matrix, $\mathbf{K}_{n+1}$. The value of $\mathbf{K}_{n+1}$ that minimizes (31) can be shown to be [32]

$$\mathbf{K}_{n+1} = -\left[\mathbf{P}_n^{fb} + \mathbf{J}\left(\mathbf{P}_n^{fb}\right)^T\mathbf{J}\right]\left[\mathbf{P}_n^b + \mathbf{J}\bar{\mathbf{P}}_n^f\mathbf{J}\right]^{-1} \tag{33}$$

$$= -\hat{\Delta}_n\left[\mathbf{J}\bar{\mathbf{P}}_n^{f+b}\mathbf{J}\right]^{-1}, \tag{34}$$

where

$$\hat{\Delta}_n \triangleq \mathbf{P}_n^{fb} + \mathbf{J}\left(\mathbf{P}_n^{fb}\right)^T\mathbf{J} \tag{35}$$

$$\mathbf{P}_n^{fb} \triangleq \frac{1}{N_1 - n}\sum_{k=n}^{N_1-1}\tilde{\boldsymbol{\epsilon}}_n^f(k)\left(\tilde{\boldsymbol{\epsilon}}_n^b(k-1)\right)^*. \tag{36}$$

Here, $\mathbf{P}_n^{fb}$ is an estimate of the cross-correlation between the generalized forward and backward 2-D linear prediction errors, and (the persymmetric) $\hat{\Delta}_n$ is an approximation for the partial correlation matrix of the 2-D Yule-Walker solution. The sum of forward and backward 2-D squared errors

that minimizes the trace will then have the following recursive relationship between orders

$$\begin{aligned}\mathbf{P}_{n+1}^{f+b} &= \left(\mathbf{P}_{n+1}^{f+b}\right)^{*} \\ &= \mathbf{P}_{n}^{f+b} + \mathbf{K}_{n+1}\hat{\Delta}_{n}^{*} \\ &= \mathbf{P}_{n}^{f+b} - \hat{\Delta}_{n}\left[\mathbf{J}\bar{\mathbf{P}}_{n}^{f+b}\mathbf{J}\right]^{-1}\hat{\Delta}_{n}^{*}. \end{aligned} \tag{37}$$

By finally combining (26), (27), (36), (34), (37), (28) and (29), in that order, the algorithm is obtained. This algorithm constitutes the paper's second main contribution.

6 Numerical Examples

We first study the computational complexity of the different methods. In our first example, the data matrix has been generated as the sum of four 2-D complex sinusoids (cisoids) corrupted by additive complex Gaussian white noise. The computational complexity is evaluated as the data matrix size, $N_1 = N_2$, varies. Figure 3(a) illustrates the computational complexity of the proposed spectral estimators for varying data matrix dimensions, as compared to the direct solution for the classical 2-D Capon spectral estimator, as given in (11), and the method in [5]. The figure shows the number of floating point operations (flops) as measured by MATLAB for the calculation of the 2-D spectrum. As a comparison Figure 3(b) shows the computational requirements for computing the outer product sample covariance matrix for the classical 2-D Capon spectral estimator and the estimator in [5], and the $\{\mathbf{A}_i\}$ and $\mathbf{Q}$ matrices using the different approaches for all the other approaches based on Theorem 1. We note that there is a clear computational gain in using any of the spectral estimates based on Theorem 1 as compared to both the classical approach and the algorithm presented in [5]. In the example, the filter lengths were $M_1 = M_2 = N_1/4$, and the spectrum is zeropadded to length $N_{\omega_1} = N_{\omega_2} = 4N_1$ (which means that the spectrum is evaluated using a $4N_1 \times 4N_1$-point 2-D FFT).

We will now proceed by studying the spectral resolutions achieved by the different methods. First, we use a 32×32 data matrix that consists of a sum of four cisoids corrupted by additive complex Gaussian white noise. The cisoids all have unit amplitude, a phase of $\pi/4$, and are located at $f = (-0.25, 0.25), (0, 0), (-0.25, 0), (0, -0.25)$. Figures 4(b)-(f) illustrate the resulting resolutions obtained by using the classical 2-D Capon estimate, the Toeplitz-Block-Toeplitz 2-D Capon estimate, the FB Block-Toeplitz 2-D

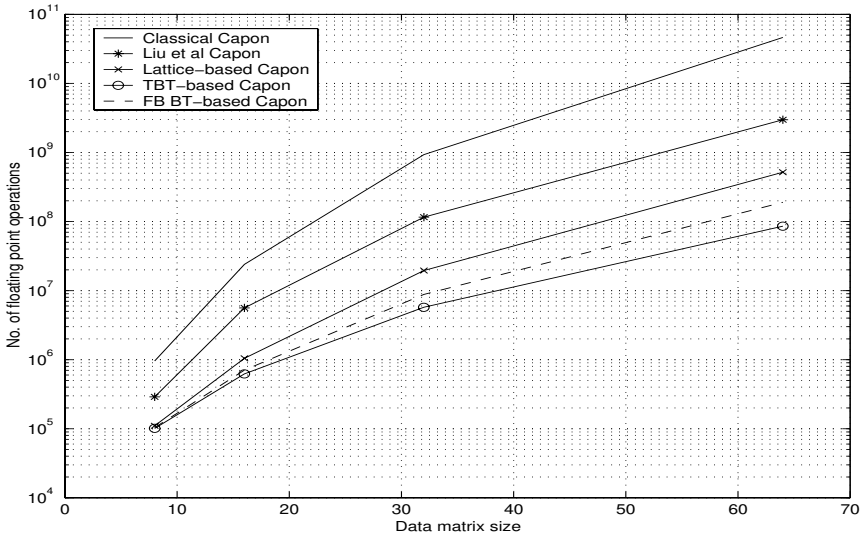
Capon estimate, the 2-D lattice Capon estimate and the 2-D Capon estimate of [5], respectively.

We note that there are significant differences between the different estimates. First, note that the 2-D classical Capon spectral estimate, based on the outer-product covariance matrix estimate, as shown in Figure 4(b), is seen to have a high frequency resolution but to provide a slightly biased amplitude estimates. This is well in accordance with the known fact that Capon will yield an amplitude estimate that is biased downwards [4, 24]. Next, the spectral estimate based on the Toeplitz-Block-Toeplitz estimated covariance matrix, as shown in Figure 4(c), is seen to give fairly wide peak estimates. Figure 4(d) shows that the spectrum based on FB Block-Toeplitz estimated covariance yields high resolution *for one of the frequencies*[1] although with significantly worse amplitude estimates. The Capon estimate of [5] is shown in Figure 4(e), and the spectral estimate based on the 2-D lattice estimates is shown in Figure 4(f). The methods are seen to have similar resolution.

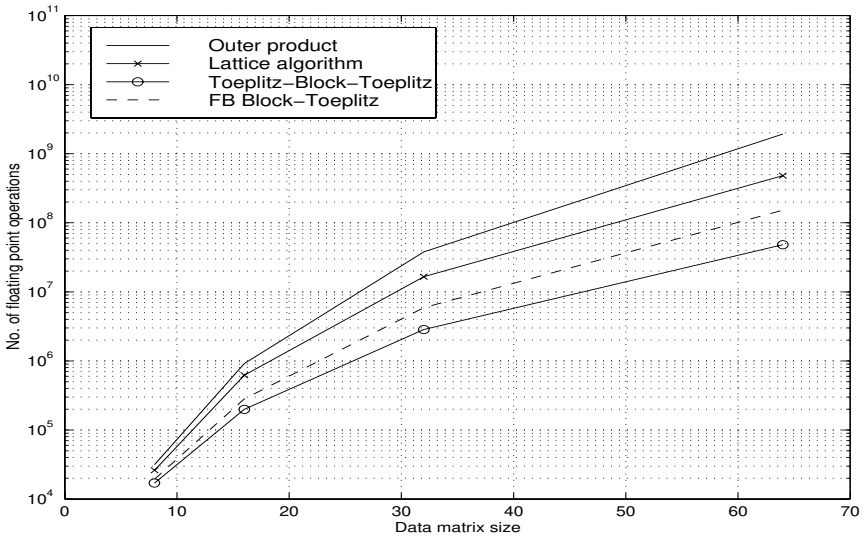
We conclude that the spectral estimate based on the $\mathbf{A}_i$ matrices as given by the 2-D lattice algorithm seems to give a better spectral estimate than the estimate given by the extended Yule-Walker solutions. This is well in accordance with the 1-D case, where the Yule-Walker algorithm's $\mathbf{A}_i$ estimates will result in lower resolution than estimates obtained by the Burg algorithm [35]. In the example, the filter lengths were $M_1 = M_2 = N_1/4$, and the spectrum is zeropadded to length $N_{\omega_1} = N_{\omega_2} = 4N_1$.

Finally, we consider synthetic aperture radar (SAR) imaging of a simulated MIG-25 airplane. The 32×32 data matrix was provided by the Naval Research Laboratory. Figures 5(a)-(d) show the 128×128 SAR images as obtained by the different methods, using $M_1 = M_2 = 15$. We note that the proposed lattice-based 2-D Capon spectral estimate seems to give the clearest image.

[1] This is due to the "row-averaging" in (3). See [33] for a further discussion on this phenomena, as well as [3, 34] for suggested remedies.

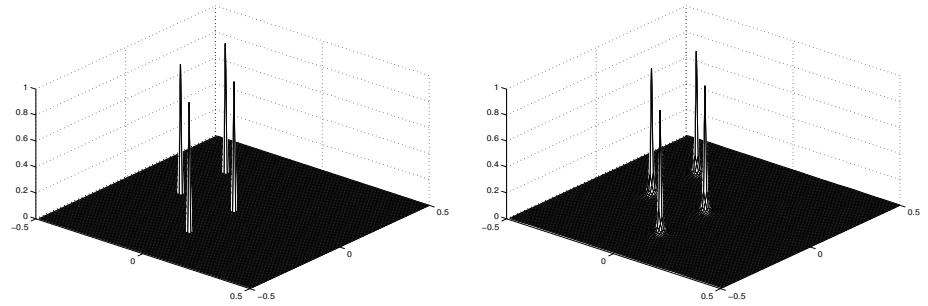


(a)



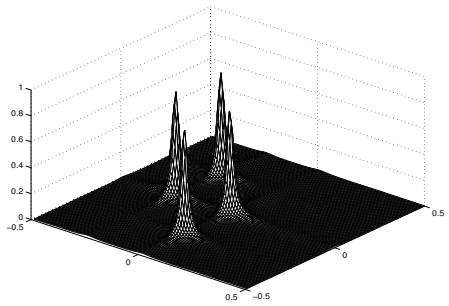
(b)

Figure 3: Computational complexity vs data matrix size ($N_1 = N_2$) (a) Computation of the spectral estimates. (b) Computation of the outer product covariance matrix for the classical and the Liu *et al* Capon spectral estimators, and the $\mathbf{A}_i$ matrices for the others.

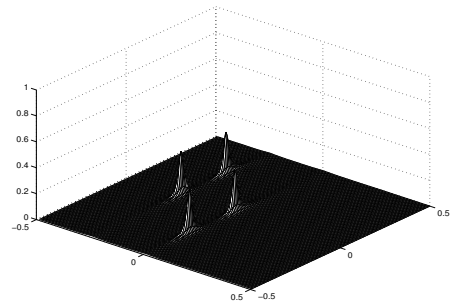


(a)

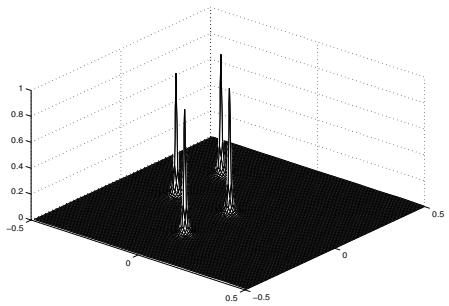
(b)



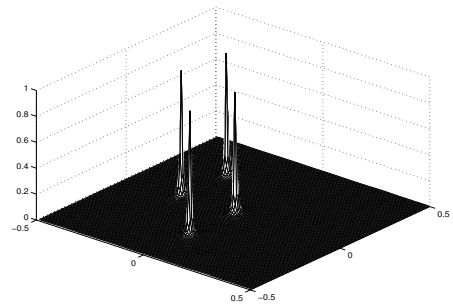
(c)



(d)

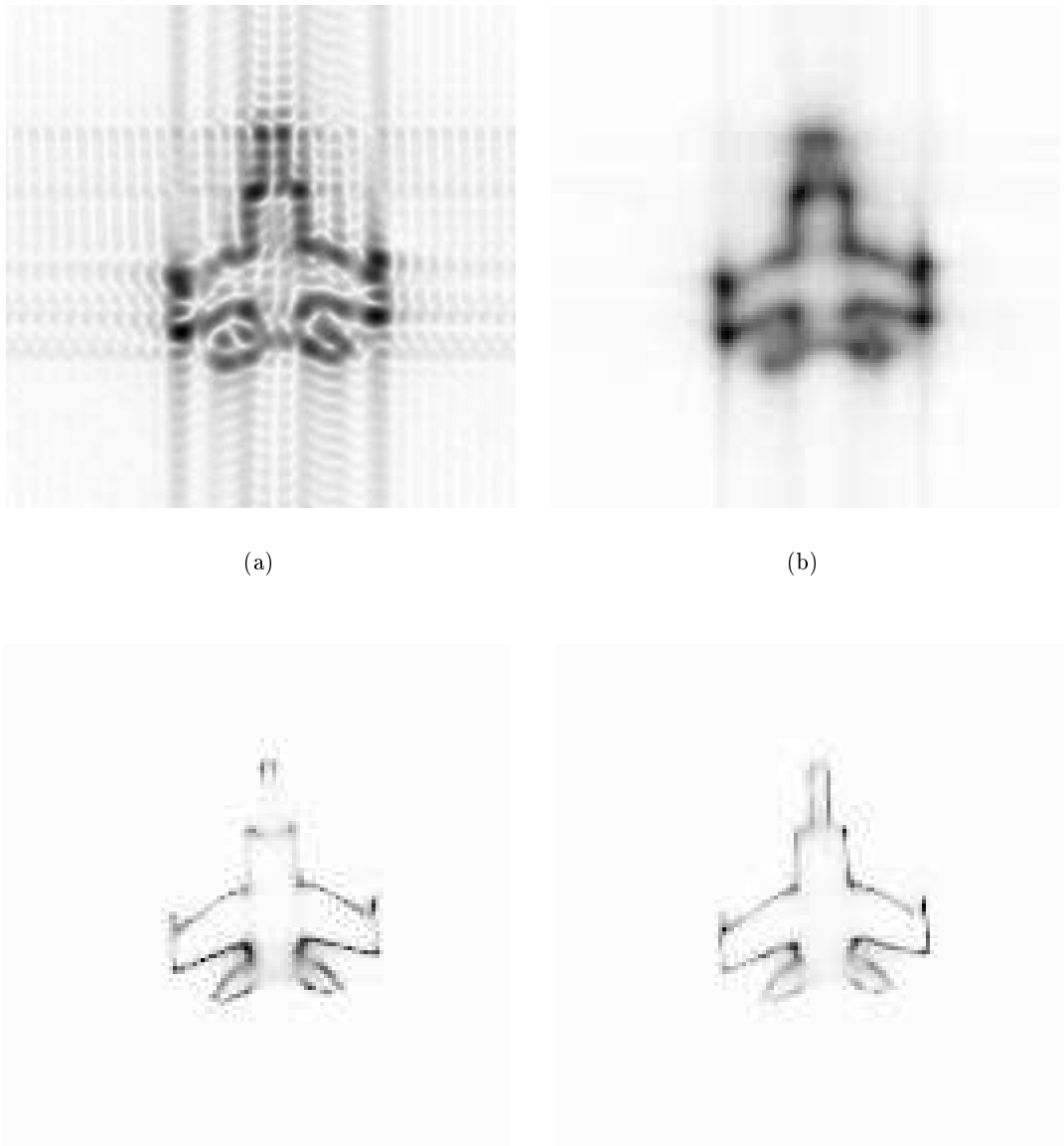


(e)



(f)

Figure 4: Illustration of the resolution and accuracy for the different spectral estimators. The estimates are plotted for fractions of the sampling frequency. (a) The true spectrum. (b) The classical 2-D Capon estimate. (c) The Toeplitz-Block-Toeplitz 2-D Capon estimate. (d) The FB Block-Toeplitz 2-D Capon estimate. (e) The 2-D Capon estimate of [5]. (f) The 2-D lattice Capon estimate.



(a) (b)

(c) (d)

Figure 5: Illustration of the resolution and accuracy for the different spectral estimators. (a) The 2-D Periodogram. (b) The Toeplitz-Block-Toeplitz 2-D Capon estimate. (c) The 2-D Capon estimate of [5]. (d) The 2-D lattice Capon estimate.

7 Conclusions

In this work we have presented an efficient implementation of the 2-D Capon spectral estimator. The derivation is based on the Gohberg-Heinig formula for the inverse of a Block-Toeplitz matrix. The so-obtained closed form expression of the inverse covariance matrix enables further simplifications of the 2-D Capon spectral estimator due to the highly structured problem formulation and the algorithm can be seen as a 2-D extension of the algorithm derived by Musicus [15] for the 1-D case. Our numerical simulations indicate a significant decrease in computational complexity as compared to the classical approach as well as the approach recently proposed by Liu *et al* [5], especially for larger matrix sizes.

Furthermore, we have presented a novel 2-D lattice algorithm to estimate the forward prediction matrices as well as the forward prediction error covariance matrix. The so-obtained estimates are found to produce a spectral estimate with high frequency resolution. Our numerical examples illustrate the improved resolution.

8 Acknowledgement

We would like to thank Dr. Hongbin Li for supplying us with the MIG data as well as the MATLAB code for the algorithm in [5].

A The Gohberg-Heinig formula

The Gohberg-Heinig formula, presented in [13] (see also [14]), is an extension of the well-known Gohberg-Semencul formula [36] (see also, e.g., [37]) to the Block-Toeplitz case. The formula is summarized below.

Theorem 2 (The Gohberg-Heinig formula)
Let $\mathbf{R}$ *be the* $(M_1+1)(M_2+1) \times (M_1+1)(M_2+1)$ *non-singular Hermitian Block-Toeplitz matrix in (4). Then it holds that*

$$\mathbf{R}^{-1} = \mathbf{L}_{\mathbf{A}}^{*}\, \mathbf{D}(\mathbf{Q}^{-1})\, \mathbf{L}_{\mathbf{A}} - \mathbf{L}_{\mathbf{B}}^{*}\, \mathbf{D}(\mathbf{S}^{-1})\, \mathbf{L}_{\mathbf{B}} \tag{A.1}$$

where $\mathbf{D}(\mathbf{Q}^{-1})$ *is a block diagonal matrix with the matrix* $\mathbf{Q}^{-1}$ *along its diagonal,*

$$\mathbf{L}_{\mathbf{A}} = \begin{bmatrix} \mathbf{I} & \mathbf{A}_1 & \dots & \mathbf{A}_{M_1} \\ \mathbf{0} & \mathbf{I} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{A}_1 \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{I} \end{bmatrix}$$

$$\mathbf{L}_{\mathbf{B}} = \begin{bmatrix} \mathbf{0} & \mathbf{B}_{M_1} & \dots & \mathbf{B}_1 \\ \mathbf{0} & \mathbf{0} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{B}_{M_1} \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \end{bmatrix},$$

and where $\{\mathbf{A}_i\}$, $\{\mathbf{B}_i\}$, $\mathbf{Q}$ *and* $\mathbf{S}$ *are the* $(M_2+1) \times (M_2+1)$ *forward and backward prediction matrices and the forward and backward prediction error covariance matrices of order* M_1 *as defined in (17).*

Proof: See [13, 14]. □

Note that, by making use of the identities (18) and (19), the Gohberg-Heinig formula can, for a Toeplitz-Block-Toeplitz matrix, be expressed in terms of only the forward prediction matrices and the forward prediction error covariance matrix as

$$\mathbf{R}^{-1} = \mathbf{L}_{\mathbf{A}}^{*}\, \mathbf{D}(\mathbf{Q}^{-1})\, \mathbf{L}_{\mathbf{A}} - \mathbf{L}_{\tilde{\mathbf{A}}}^{*}\, \mathbf{D}\left((\mathbf{J}\bar{\mathbf{Q}}\mathbf{J})^{-1}\right)\, \mathbf{L}_{\tilde{\mathbf{A}}}$$

where

$$\mathbf{L}_{\tilde{\mathbf{A}}} = \begin{bmatrix} \mathbf{0} & (\mathbf{J}\bar{\mathbf{A}}_{M_1}\mathbf{J}) & \dots & (\mathbf{J}\bar{\mathbf{A}}_1\mathbf{J}) \\ \mathbf{0} & \mathbf{0} & \ddots & \vdots \\ \vdots & \ddots & \ddots & (\mathbf{J}\bar{\mathbf{A}}_{M_1}\mathbf{J}) \\ \mathbf{0} & \dots & \mathbf{0} & \mathbf{0} \end{bmatrix}.$$

B Proof of Theorem 1

Making use of the Gohberg-Heinig formula, as stated in Appendix A, we rewrite f_{ω_1,ω_2} as

$$\begin{aligned} f_{\omega_1,\omega_2} &= \sum_{p=0}^{M_1}\sum_{k=0}^{M_1}\sum_{l=0}^{M_1} \mathbf{a}_{\omega_2}^* \left(\mathbf{A}_{k-p}^* \mathbf{Q}^{-1} \mathbf{A}_{l-p} - \right. \\ &\quad \left. - \mathbf{B}_{M_1+1+p-k}^* \mathbf{S}^{-1} \mathbf{B}_{M_1+1+p-l}\right) \mathbf{a}_{\omega_2} e^{i\omega_1(k-l)} \\ &= \sum_{p=0}^{M_1}\sum_{l=0}^{M_1}\sum_{s=l-M_1}^{l} \mathbf{a}_{\omega_2}^* \left(\mathbf{A}_{l-p-s}^* \mathbf{Q}^{-1} \mathbf{A}_{l-p} - \right. \\ &\quad \left. - \mathbf{B}_{M_1+1+p-l+s}^* \mathbf{S}^{-1} \mathbf{B}_{M_1+1+p-l}\right) \mathbf{a}_{\omega_2} e^{-i\omega_1 s}, \end{aligned} \tag{B.1}$$

where the last equality has been obtained by substitution $s = l - k$. Next, make substitution $j = l - p$ in (B.1) to obtain

$$\begin{aligned} f_{\omega_1,\omega_2} &= \sum_{l=0}^{M_1}\sum_{s=l-M_1}^{l}\sum_{j=l-M_1}^{l} \mathbf{a}_{\omega_2}^* \left(\mathbf{A}_{j-s}^* \mathbf{Q}^{-1} \mathbf{A}_{j} - \right. \\ &\quad \left. - \mathbf{B}_{M_1+1+s-j}^* \mathbf{S}^{-1} \mathbf{B}_{M_1+1-j}\right) \mathbf{a}_{\omega_2} e^{-i\omega_1 s} \\ &= \sum_{l=0}^{M_1}\sum_{s=l-M_1}^{M_1}\sum_{j=0}^{l} \mathbf{a}_{\omega_2}^* \left(\mathbf{A}_{j-s}^* \mathbf{Q}^{-1} \mathbf{A}_{j} - \right. \\ &\quad \left. - \mathbf{B}_{M_1+1+s-j}^* \mathbf{S}^{-1} \mathbf{B}_{M_1+1-j}\right) \mathbf{a}_{\omega_2} e^{-i\omega_1 s}, \end{aligned} \tag{B.2}$$

where we have made use of the fact that the summand is zero for $j < 0$, and hence we can truncate the summation over j to the interval $[0, l]$. Furthermore, we have made use of the fact that since $\mathbf{A}_{j-s} = \mathbf{B}_{M_1+1+s-j} = \mathbf{0}$ for $s > j$, we can extend the summation over s up to $s = M_1$. We proceed by writing (B.2) as

$$f_{\omega_1,\omega_2} \stackrel{\triangle}{=} T_1 + T_2,$$

where

$$\begin{aligned} T_1 &= \sum_{s=0}^{M_1} \mu_s e^{-i\omega_1 s} \\ T_2 &= \sum_{s=-M_1}^{-1} \rho_{-s} e^{-i\omega_1 s} \end{aligned}$$

and, where

$$\mu_s \triangleq \sum_{l=0}^{M_1} \sum_{j=0}^{l} \mathbf{a}_{\omega_2}^* \left(\mathbf{A}_{j-s}^* \mathbf{Q}^{-1} \mathbf{A}_j - \mathbf{B}_{M_1+1+s-j}^* \mathbf{S}^{-1} \mathbf{B}_{M_1+1-j} \right) \mathbf{a}_{\omega_2}. \tag{B.3}$$

As $f_{\omega_1,\omega_2} = T_1 + T_2 \in \mathbb{R}$ for all $\omega_1, \omega_2 \in [-\pi, \pi]$, it holds that

$$\begin{aligned} 0 &= T_1 + T_2 - T_1^* - T_2^* \\ &= \sum_{k=1}^{m} (\mu_k - \rho_k^*) e^{-i\omega_1 k} + (\rho_k - \mu_k^*) e^{i\omega_1 k} \\ &= \sum_{k=1}^{m} \alpha_k z^{-k} + \alpha_k^* z^k \\ &= \sum_{k=0}^{2m} \gamma_k z^k \end{aligned} \tag{B.4}$$

where $z \triangleq e^{i\omega_1}$, $\alpha_k \triangleq \mu_k - \rho_k^*$, and

$$\gamma_k = \begin{cases} -\alpha_{k-M_1}^* & k = M_1 + 1, \ldots, 2M_1 \\ 0 & k = M_1 \\ \alpha_{M_1-k} & k = 0, \ldots, M_1 - 1 \end{cases}$$

The polynomial in (B.4) has, according to the fundamental theorem of algebra [38], exactly $2M_1$ zeros, and as (B.4) holds for all $\omega_1 \in [-\pi, \pi]$, and thus providing an infinite number of roots, it must hold that $\gamma_k = 0$. Thus,

$$\mu_k = \rho_k^*$$

which concludes that

$$f_{\omega_1,\omega_2} = \sum_{s=-M1}^{M_1} \mu_s e^{-i\omega_1 s}. \tag{B.5}$$

Next, we proceed by noting that the summand in (B.3) does not depend on l and can thus be written as

$$\begin{aligned}
\mu_s &= \sum_{j=0}^{M_1}(M_1+1-j)\mathbf{a}_{\omega_2}^*\left(\mathbf{A}_{j-s}^*\mathbf{Q}^{-1}\mathbf{A}_j - \mathbf{B}_{M_1+1+s-j}^*\mathbf{S}^{-1}\mathbf{B}_{M_1+1-j}\right)\mathbf{a}_{\omega_2} \\
&= \sum_{k=-s}^{M_1-s}(M_1+1-k-s)\,\mathbf{a}_{\omega_2}^*\left(\mathbf{A}_k^*\mathbf{Q}^{-1}\mathbf{A}_{k+s}\right)\mathbf{a}_{\omega_2} - \\
&\quad - \sum_{k=-s}^{M_1-s}(M_1+1-k-s)\,\mathbf{a}_{\omega_2}^*\left(\mathbf{B}_{M_1+1-k}^*\mathbf{S}^{-1}\mathbf{B}_{M_1+1-k-s}\right)\mathbf{a}_{\omega_2} \\
&= \sum_{k=0}^{M_1-s}(M_1+1-k-s)\,\mathbf{a}_{\omega_2}^*\left(\mathbf{A}_k^*\mathbf{Q}^{-1}\mathbf{A}_{k+s}\right)\mathbf{a}_{\omega_2} - \\
&\quad - \sum_{l=1}^{M_1+1} l\,\mathbf{a}_{\omega_2}^*\left(\mathbf{B}_{l+s}^*\mathbf{S}^{-1}\mathbf{B}_l\right)\mathbf{a}_{\omega_2} \\
&= \sum_{k=0}^{M_1-s}(M_1+1-k-s)\,\mathbf{a}_{\omega_2}^*\left(\mathbf{A}_k^*\mathbf{Q}^{-1}\mathbf{A}_{k+s}\right)\mathbf{a}_{\omega_2} - \\
&\quad - k\,\mathbf{a}_{\omega_2}^*\left(\mathbf{B}_{k+s}^*\mathbf{S}^{-1}\mathbf{B}_k\right)\mathbf{a}_{\omega_2} \qquad \text{(B.6)}
\end{aligned}$$

where the second equality has been obtained by substituting $j = k+s$. Furthermore, the third equation is obtained by again noting that $\mathbf{A}_k = \mathbf{0}$ for $k < 0$, as well as substituting $l = M_1+1-k-s$ in the second summation, and the fourth equality is obtained by noting that $\mathbf{B}_k = \mathbf{0}$ for $k > M_1$. Next, we make use of (18) and (19) to rewrite

$$\begin{aligned}
\mathbf{a}_{\omega_2}^*\left(\mathbf{B}_{k+s}^*\mathbf{S}^{-1}\mathbf{B}_k\right)\mathbf{a}_{\omega_2} &= \mathbf{a}_{\omega_2}^*\left(\mathbf{J}\mathbf{A}_{k+s}^T\bar{\mathbf{Q}}^{-1}\bar{\mathbf{A}}_k\mathbf{J}\right)\mathbf{a}_{\omega_2} \\
&= \mathbf{a}_{\omega_2}^*\mathbf{J}\left(\mathbf{A}_k^*\mathbf{Q}^{-1}\mathbf{A}_{k+s}\right)^T\mathbf{J}\mathbf{a}_{\omega_2} \\
&= \mathbf{a}_{\omega_2}^T e^{i\omega_2 M_2}\left(\mathbf{A}_k^*\mathbf{Q}^{-1}\mathbf{A}_{k+s}\right)^T e^{-i\omega_2 M_2}\bar{\mathbf{a}}_{\omega_2} \\
&= \mathbf{a}_{\omega_2}^*\left(\mathbf{A}_k^*\mathbf{Q}^{-1}\mathbf{A}_{k+s}\right)\mathbf{a}_{\omega_2}. \qquad \text{(B.7)}
\end{aligned}$$

Making use of (B.7), we then rewrite (B.6) as

$$\mu_s = \sum_{k=0}^{M_1-s}(M_1+1-2k-s)\,\mathbf{a}_{\omega_2}^*\left(\mathbf{A}_k^*\mathbf{Q}^{-1}\mathbf{A}_{k+s}\right)\mathbf{a}_{\omega_2}. \qquad \text{(B.8)}$$

Next, we make use of the following result:

Theorem 3
Let $\mathbf{\Lambda} = \{\mathbf{\Lambda}_{i,j}\} \in \mathbb{C}^{(M+1)\times(M+1)}$ *and let* $\mathbf{a}_{\omega_2} = \left[\begin{array}{cccc} 1 & e^{-i\omega_2} & \dots & e^{-iM\omega_2} \end{array}\right]^T$, *then*

$$\mathbf{a}^*_{\omega_2}\mathbf{\Lambda}\mathbf{a}_{\omega_2} = \sum_{s=-M}^{M} \psi_s e^{is\omega_2},$$

where

$$\psi_s = \sum_{k=\max(0,s)}^{\min(M+s,M)} \mathbf{\Lambda}_{k,k-s}.$$

If $\mathbf{\Lambda}$ *is Hermitian, then* $\psi_s = \bar{\psi}_{-s}$.

Proof: See [39]. □

We can thus rewrite μ_s in (B.8) as

$$\begin{aligned} \mu_s &= \sum_{k=0}^{M_1-s} (M_1+1-2k-s) \sum_{p=-M_2}^{M_2} \Psi(s,k,p) e^{i\omega_2 p} \\ &= \sum_{p=-M_2}^{M_2} \Theta(s,p) e^{i\omega_2 p} \end{aligned} \tag{B.9}$$

where

$$\Theta(s,p) = \sum_{k=0}^{M_1-s} (M_1+1-2k-s)\, \Psi(s,p,k) \tag{B.10}$$

and where

$$\Psi(s,p,k) = \sum_{l=\max(0,p)}^{\min(M_2+p,M_2)} \left[\mathbf{A}_k^* \mathbf{Q}^{-1} \mathbf{A}_{k+s}\right]_{l,l-p}. \tag{B.11}$$

Finally, by combining (B.11) with (B.5), (B.9) and (B.10), the proof is concluded. □

C The Whittle-Wiggins-Robinson algorithm

The Whittle-Wiggins-Robinson algorithm (WWRA) [20, 21] is the extension to the multivariate case of the Levinson-Durbin algorithm. For more details on the WWRA algorithm, see, e.g., [28]. The algorithm is summarized below:

(a) Initialization:

$$\begin{aligned} \mathbf{Q}_0 &= \mathbf{S}_0 = \mathbf{R}_0 \\ \mathbf{P}_1 &= \mathbf{R}_1 \end{aligned}$$

(b) Order update: (for $n = 0, \ldots, M_1$)

$$\begin{aligned} \mathbf{P}_n &= \mathbf{R}_{n+1} + \mathbf{A}_1^{(n)}\mathbf{R}_n + \ldots + \mathbf{A}_n^{(n)}\mathbf{R}_1 \\ &\quad \begin{pmatrix} \mathbf{I} & \mathbf{A}_1^{(n+1)} & \ldots & \mathbf{A}_n^{(n+1)} & \mathbf{A}_{n+1}^{(n+1)} \\ \mathbf{B}_{n+1}^{(n+1)} & \mathbf{B}_n^{(n+1)} & \ldots & \mathbf{B}_1^{(n+1)} & \mathbf{I} \end{pmatrix} = \qquad \text{(C.1)} \\ &= \begin{pmatrix} \mathbf{I} & -\mathbf{P}_n\mathbf{S}_n^{-1} \\ -\mathbf{P}_n^*\mathbf{Q}_n^{-1} & \mathbf{I} \end{pmatrix} \begin{pmatrix} \mathbf{I} & \mathbf{A}_1^{(n)} & \ldots & \mathbf{A}_n^{(n)} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_n^{(n)} & \ldots & \mathbf{B}_1^{(n)} & \mathbf{I} \end{pmatrix} \\ \mathbf{Q}_{n+1} &= \mathbf{Q}_n - \mathbf{P}_n\mathbf{S}_n^{-1}\mathbf{P}_n^* \\ \mathbf{S}_{n+1} &= \mathbf{S}_n - \mathbf{P}_n^*\mathbf{Q}_n^{-1}\mathbf{P}_n \end{aligned}$$

References

[1] D. E. Dudgeon and R. M. Mersereau, *Multidimensional Digital Signal Processing*, Prentice Hall, Englewood Cliffs, N. J., 1984.

[2] J. H. McClellan, "Multidimensional Spectral Estimation", *Proc. IEEE*, **70**:1029–1037, September 1982.

[3] S. L. Marple, Jr., *Digital Spectral Analysis with Applications*, Prentice-Hall, Englewood Cliffs, N.J., 1987.

[4] H. Li, J. Li, and P. Stoica, "Performance Analysis of Forward-Backward Matched-filterbank Spectral Estimators", *IEEE Trans. Signal Processing*, **46**(7):1954–1966, July 1998.

[5] Z.-S. Liu, H. Li, and J. Li, "Efficient Implementation of Capon and APES for Spectral Estimation", *IEEE Trans. Aero. and Elec. Syst.*, **34**(4):1314–1319, October 1998.

[6] S. L. Marple, Jr., *Digital Signal Analysis*, Prentice-Hall, Upper Saddle River, N.J., In preparation.

[7] S. R. DeGraaf, SAR Imaging via Modern 2-D Spectral Estimation Methods, *IEEE Trans. on Image Processing*, **7**(5):729–761, May 1998.

[8] C. V. Jakowatz, Jr., D. E. Wahl, P. H. Eichel, D. C. Ghiglia, and P. A. Thomson, *Spotlight-Mode Synthetic Aperture Radar: A Signal Processing Approach*, Kluwer, Norwell, MA, 1996.

[9] R. Klemm, "Current Trends in SAR Technology", *IEEE Aerosp. Electron. Syst. Mag.*, **12**:3–8, March 1997.

[10] R. E. Hoffman, A. Kumar, K. D. Bishop, P. N. Borer, and G. C. Levy, "Application of the Maximum Likelihood Method to a Large 2D NMR Spectrum Using a Parallel Computer", *Journal of Magnetic Resonance*, **83**:586–594, 1989.

[11] B. McGuffin and B. Liu, "An Efficient Algorithm for Two-Dimensional Autoregressive Spectrum Estimation", *IEEE Trans. ASSP*, **37**(1):106–117, January 1989.

[12] C. W. Therrien and H. T. El-Shaer, "A Direct Algorithm for Computing 2-D AR Power Spectrum Estimates", *IEEE Trans. on ASSP*, **37**(11):1795–1798, November 1989.

[13] I. C. Gohberg and G. Heinig, "Inversion of finite Toeplitz matrices made of elements of a non-commutative algebra", *Rev. Roumaine Math. Pures Appl.*, **XIX**(5):623–663, 1974, (In Russian).

[14] B. Friedlander, M. Morf, T. Kailath, and L. Ljung, "New Inversion Formulas for Matrices Classified in Terms of Their Distance from Toeplitz Matrices", *Linear Algebra and its applications*, **27**:31–60, 1979.

[15] B. Musicus, "Fast MLM Power Spectrum Estimation from Uniformly Spaced Correlations", *IEEE Trans. ASSP*, **33**(4):1333–1335, October 1985.

[16] O. N. Strand, "Multichannel Complex Maximum Entropy (Autoregressive) Spectral Analysis", *IEEE Trans. Automatic Control*, **22**(4):634–640, August 1977.

[17] G. R. Kuduvalli and R. M. Rangayyan, "An Algorithm for Diect Computation of 2-D Linear Prediction Coefficients", *IEEE Trans. on Signal Processing*, **41**(2):996–1000, Febr. 1993.

[18] I. J. Gupta, "High-Resolution Radar Imaging Using 2-D Linear Prediction", *IEEE Trans. Antennas and Propagation*, **42**(1):31–37, January 1994.

[19] J. P. Burg, *Maximum Entropy Spectral Analysis*, PhD thesis, Stanford University, 1975.

[20] P. Whittle, "On the Fitting of Multivariate Autoregressions, and the Approximative Canonical Factorization of a Spectral Density Matrix", *Biometrika*, **50**:129–134, 1963.

[21] R. A. Wiggins and E. A. Robinson, "Recursive Solution of the Multichannel Filtering Problem", *J. Geophys. Res.*, **70**:1885–1891, April 1965.

[22] J. Capon, "High Resolution Frequency Wave Number Spectrum Analysis", *Proc. IEEE*, **57**:1408–1418, 1969.

[23] J. Capon, *"Maximum-Likelihood Spectral Estimation"*, Chapter 5 in *Nonlinear Methods of Spectral Analysis*, 2nd ed., S. Haykin,ed. Springer-Verlag, New York, USA, 1983.

[24] P. Stoica, A. Jakobsson, and J. Li, "Matched-Filterbank Interpretation of Some Spectral Estimators", *Signal Processing*, **66**(1):45–59, April 1998.

[25] M. Wax and T. Kailath, "Efficient Inversion of Toeplitz-block-Toeplitz Matrix", *IEEE Trans. ASSP*, **31**(2):1218–1221, Oct. 1983.

[26] N. Kalouptsidis, G. Carayannis, and D. Manolakis, "Fast Algorithms for Block Toeplitz Matrices with Toeplitz Entries", *Signal Processing*, **6**:77–81, 1984.

[27] M. R. Palsetia and J. Li, Using APES for Interferometric SAR Imaging, *IEEE Trans. on Image Processing*, **7**(9):1340–1353, September 1998.

[28] T. Söderström and P. Stoica, *System Identification*, Prentice Hall International, London, UK, 1989.

[29] H. Kimura and Y. Honoki, "A Hybrid Approach to High Resolution Two-Dimensional Spectrum Analysis", *IEEE Trans. ASSP*, **35**(7):1024–1036, 1987.

[30] M. Jansson and P. Stoica, "Forward-Only and Forward-Backward Sample Covariances – A Comparative Study", *Signal Processing*, **77**(3):235–245, 1999.

[31] J. Makhoul, "Linear Prediction: A Tutorial Review", *Proc. of the IEEE*, **63**:561–580, 1975.

[32] S. L. Marple, Jr., "Two-Dimensional Fast Computational Lattice Algorithm", In *Proc. 33rd Asilomar Conf. on Signals, Systems, and Computers*, 1999.

[33] J. W. Odendaal, E. Barnard, and C. W. I. Pistorius, "On the Asymmetry in the PSD for 2-D Linear Prediction", *Signal Processing*, **60**:43–49, 1997.

[34] L. Jackson and H. Chien, "Frequency and Bearing Estimation by Two-Dimensional Linear Prediction", In *Proc. ICASSP '79*, pages 665–668, 1979.

[35] T. J. Ulrych and T. N. Bishop, Maximum entropy spectral analysis and autoregressive decomposition, *Review of Geophysics ans Space Physics*, **13**(1):183–200, February 1975.

[36] I. C. Gohberg and A. A. Semencul, "On the inversion of finite Toeplitz matrices and their continuous analogs", *Mat. Issled.*, **2**:201–233, 1972, (In Russian).

[37] P. Stoica and R. Moses, *Introduction to Spectral Analysis*, Prentice Hall, Upper Saddle River, NJ, 1997.

[38] J. C. F. Gauss, *Demonstratio Nova Theorematis Omnem Functionem Algebraicam Rationalem Integram Unius Variabilis in Factores Reales Primi Vel Secundi Gradus Resolve Posse*, PhD thesis, University of Helmstedt, Germany, 1799.

[39] H. Li, P. Stoica, and J. Li, "Capon Estimation of Covariance Sequences", *Circuits Systems Signal Processing*, **17**(1):29–49, 1998.