Estimating Two-Dimensional Frequencies by Matrix Enhancement and Matrix Pencil

Yingbo Hua, *Senior Member, IEEE*

***Abstract*—A new method, called the matrix enhancement and matrix pencil (MEMP) method, is presented for estimating two-dimensional (2-D) frequencies. The MEMP method first constructs an enhanced matrix from the data samples, and then uses the matrix pencil approach to extract out the 2-D sinusoids from the principal eigenvectors of the enhanced matrix. The MEMP method yields the estimates of the 2-D frequencies efficiently, without solving the roots of a 2-D polynomial or searching in a 2-D space. It is shown that the MEMP method can be faster than a 2-D FFT method if the number of the 2-D sinusoids is much smaller than the data set. Simulation results are provided to show that the accuracy of the MEMP method can be very close to the Cramér–Rao lower bound.**

I. Introduction

In many applications, such as synthetic aperture radar imaging, frequency and wave-number estimation in array processing, and nuclear magnetic resonance imaging, it is often desired to estimate two-dimensional (2-D) frequencies from a 2-D data set. If the data set is very large, the classical correlogram method (implementable via 2-D FFT) can be satisfactory. If the data set is relatively small, the correlogram method suffers from a resolution limit called Rayleigh limit. To overcome the Rayleigh limit, high-resolution techniques such as the 2-D autoregressive method, 2-D maximum entropy method, and 2-D minimum variance method have been developed from their 1-D versions. To obtain the estimates of the 2-D frequencies, searching for spectral peaks in a 2-D space is required by all those methods. The searching is due to the difficulty in finding the desired roots of a 2-D polynomial. It causes a very large amount of computations, and hence limits the estimation accuracy given a fixed amount of computations. Tutorial discussions on those methods are available in [1] and [17].

The computational difficulty of searching in a 2-D space also exists with other methods in [9]–[11]. This is again due to the bottleneck: 2-D polynomial. A recently published method [12] for estimating 2-D frequency, although computationally efficient, only applies to the single 2-D sinusoid case.

A computationally efficient method for estimating multiple 2-D frequencies is available in the work by Kung *et al.* [2]. This method, called the state space method, does not require searching in a 2-D space. It exploits the structure inherent in the original data matrix. The 2-D frequencies are computed by solving an eigenvalue problem. However, the state space method does not work for the case where more than one 2-D sinusoids share a common 1-D sinusoidal component. Furthermore, this method yields two sets of estimated 1-D frequencies rather than a set of estimated 2-D frequencies. How to pair the two sets of estimated 1-D frequencies into a set of estimated 2-D frequencies was not mentioned in [2].

Another 2-D frequency estimation method, called the matrix approximation method, was proposed by Shaw–Kumaresan in [3]. Similar to the state space method in [2], the matrix approximation method is based on the inherent structure of the original data matrix. The difference is, however, that the matrix approximation method tries to reconstruct such a matrix, subject to the constraint of the known data structure, that approximates the original data matrix in a least square (LS) sense. Like those of [2], the authors of [3] did not address the problem arising from multiple 2-D frequencies having a common 1-D frequency component. The pairing issue was also not addressed in [3].

In this paper, we will follow an approach similar to those in [2] and [3] to exploit the structure inherent in the original data. However, instead of relying on the original data matrix, we will form an enhanced matrix from the original data. The enhanced matrix is formed in such a way that the matrix pencil approach in [4] can be applied to efficiently estimate the 2-D frequencies. The resulting method will be called the matrix enhancement and matrix pencil (MEMP) method.

In Section II, the 2-D frequency estimation problem will be formulated. A basic structure inherent in the original data matrix will be reviewed. It will be pointed out why the methods in [2] and [3] fail to work.

In Section III, the idea of matrix enhancement will be introduced, and the structure of an enhanced matrix will be studied. It will be shown that the number of 2-D sinusoids can be obtained from the rank of the enhanced matrix, and the 2-D frequencies can be obtained from the principal eigenvectors of the enhanced matrix.

In Section IV, the matrix pencil approach will be applied to efficiently estimate the 2-D frequencies from the principal eigenvectors of the enhanced matrix. The pairing issue will also be addressed.

Manuscript received January 8, 1991; revised July 18, 1991. This work was supported in part by the Australian Research Council and the Australian Defence Science and Technology Organization.

The author is with the Department of Electrical and Electronic Engineering, University of Melbourne, Parkville, Victoria 3052 Australia.

IEEE Log Number 9201589.

In Section V, the noisy data will be assumed when the MEMP method is summarized into a step-by-step algorithm. For the case where the data set is very large and the noise covariance sequence is known except a scalar, an asymptotically consistent version will also be given.

In Section VI, an estimated order of computations required by the MEMP method will be derived It will be shown that the MEMP method can be faster than a typical 2-D FFT method if the number of 2-D sinusoids is much smaller than the data size.

In Section VII, simulation results will be provided to show the noise robustness of the MEMP method. It will be seen that the accuracy of the MEMP method can be very close to the Cramér–Rao lower bound (CRB). A set of equations useful for computing the CRB will be given in Appendix A.

II. Problem Formulation

We assume that the noiseless 2-D data samples have the following structure:

$$x(m; n) = \sum_{i=1}^{I} r_i \exp(j\phi_i + j2\pi f_{1i}m + j2\pi f_{2i}n) \quad (2.1)$$

where $0 \le m \le M-1$, and $0 \le n \le N-1$. Equation (2.1) implies that $x(m; n)$ consists of I 2-D sinusoids at the (distinct) 2-D frequencies $\{(f_{1i}, f_{2i}); i = 1, \cdots, I\}$. $\{r_i; i = 1, \cdots, I\}$ and $\{\phi_i; i = 1, \cdots, I\}$ are the (nonzero) amplitudes and phases, respectively. In the noisy case, we write $x'(m; n) = x(m; n) + w(m, n)$, where $w(m; n)$ is the 2-D noise sequence. In this paper, the prime will be used to denote the noisy quantities.

The basic problem here is to estimate $\{(f_{1i}, f_{2i}); i = 1, \cdots, I\}$ from $x'(m; n)$. r_i and ϕ_i can be straightforwardly estimated once the 2-D frequencies are obtained since $x(m; n)$ is a linear function of the complex amplitudes $r_i \exp(j\phi_i)$. Estimating r_i and ϕ_i will not be addressed. But in Appendix B, a simple algorithm for this task will be given.

To make notations simpler, we rewrite (2.1) into

$$x(m; n) = \sum_{i=1}^{I} a_i y_i^m z_i^n \quad (2.2)$$

where

$$y_i = \exp(j2\pi f_{1i}) \quad (2.3)$$

$$z_i = \exp(j2\pi f_{2i}) \quad (2.4)$$

$$a_i = r_i \exp(j\phi_i). \quad (2.5)$$

$\{a_i; i = 1, \cdots, I\}$ are the complex amplitudes, and $\{(y_i, z_i); i = 1, \cdots, I\}$ the 2-D poles. Since the 2-D frequencies can be obtained uniquely from the 2-D poles (i.e., $f_{1i} = (1/2\pi)\,\mathrm{Im}(\log(y_i))$ and $f_{2i} = (1/2\pi)\,\mathrm{Im}(\log(z_i))$), we will concentrate on the estimation of the 2-D poles.

The original (noiseless) data matrix is defined as follows:

$$\boldsymbol{X} = \begin{bmatrix} x(0; 0) & x(0; 1) & \cdots & x(0; N-1) \\ x(1; 0) & x(1; 1) & \cdots & x(1; N-1) \\ & & \cdots & \\ x(M-1; 0) & x(M-1; 1) & \cdots & x(M-1; N-1) \end{bmatrix}. \quad (2.6)$$

Using (2.2) in (2.6) yields

$$\boldsymbol{X} = \boldsymbol{YAZ} \quad (2.7)$$

where

$$\boldsymbol{Y} = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ y_1 & y_2 & \cdots & y_I \\ \vdots & \vdots & & \vdots \\ y_1^{M-1} & y_2^{M-1} & \cdots & y_I^{M-1} \end{bmatrix} \quad (2.8)$$

$$\boldsymbol{A} = \mathrm{diag}[a_1, a_2, \cdots, a_I] \quad (2.9)$$

$$\boldsymbol{Z} = \begin{bmatrix} 1 & z_1 & \cdots & z_1^{N-1} \\ 1 & z_2 & \cdots & z_2^{N-1} \\ & & \cdots & \\ 1 & z_I & \cdots & z_I^{N-1} \end{bmatrix}. \quad (2.10)$$

From (2.7)–(2.10), we know that the rank of $\boldsymbol{X}$ is no larger than I, i.e., rank $(\boldsymbol{X}) \le I$. Due to the Vandermond structure in $\boldsymbol{Y}$ and $\boldsymbol{Z}$, it can be shown that $\boldsymbol{X}$ has the rank I if and only if (iff) the two sets of the 1-D poles: $\{y_i; i = 1, \cdots, I\}$ and $\{z_i; i = 1, \cdots, I\}$ both contain distinct (nonzero) elements, provided $M \ge I$ and $N \ge I$. It was a basic condition under which the state space method [2] and the matrix approximation method [3] were developed. But the rank of $\boldsymbol{X}$ is less than I if either one of the pole sets does not contain distinct elements. Note that the assumption $(y_i, z_i) \ne (y_j, z_j)$ does not necessarily mean that $y_i \ne y_j$ and $z_i \ne z_j$. It is the ill condition (insufficient rank) of $\boldsymbol{X}$ that causes the two methods to fail.

It is important to note that a) if rank $(\boldsymbol{X})$ is less than I, $\{y_i; i = 1, \cdots, I\}$ and $\{z_i; i = 1, \cdots, I\}$ cannot be both obtained from the principal left or right singular vectors of $\boldsymbol{X}$; and b) the principal singular vectors of $\boldsymbol{X}$ do not contain sufficient information to carry out the pairing between y_i and z_i.

In the next section, we will form an enhanced matrix from the 2-D data so that the above problems can be solved.

III. Matrix Enhancement

The idea of the matrix enhancement can be seen from two simple examples as follows.

Example 1: A row vector cannot have a rank larger than one. However, if this vector is partitioned into two (overlapped or nonoverlapped) subvectors and the two subvectors are stacked into a two-row matrix, then the resulting matrix may have a rank larger than one.

Example 2: If an m-row matrix has the rank r, the matrix obtained from a similar partition-and-stacking process may have a rank larger than r.

We see that the rank condition of a matrix can be enhanced by a partition-and-stacking process. (This idea is similar, from a mathematical point of view, to the ideas of moving window [19] for a uniform linear array problem and focusing [20] for a wide-band wave direction finding problem.)

Note that in this and the next sections, only the noiseless data samples will be considered. The noise effects will be discussed in Sections V and VII.

A. Enhanced Matrix X_e

An enhanced matrix useful for the 2-D frequency estimation problem is defined through a partition-and-stacking process as follows:

$$\boldsymbol{X}_e = \begin{bmatrix} \boldsymbol{X}_0 & \boldsymbol{X}_1 & \cdots & \boldsymbol{X}_{M-K} \\ \boldsymbol{X}_1 & \boldsymbol{X}_2 & \cdots & \boldsymbol{X}_{M-K+1} \\ & & \cdots & \\ \boldsymbol{X}_{K-1} & \boldsymbol{X}_K & \cdots & \boldsymbol{X}_{M-1} \end{bmatrix} \tag{3.1}$$

where

$$\boldsymbol{X}_m = \begin{bmatrix} x(m;0) & x(m;1) & \cdots & x(m;N-L) \\ x(m;1) & x(m;2) & \cdots & x(m;N-L+1) \\ & & \cdots & \\ x(m;L-1) & x(m;L) & \cdots & x(m;N-1) \end{bmatrix}. \tag{3.2}$$

$\boldsymbol{X}_e$ is an $K \times (M-K+1)$ Hankel block matrix, and $\boldsymbol{X}_m$ is an $L \times (N-L+1)$ Hankel matrix. Each column of $\boldsymbol{X}_m$ is a windowed segment of the sequence $\{x(m;0), x(m;1), \cdots, x(m;N-1)\}$ with the window length L. Each column of $\boldsymbol{X}_e$ is a windowed segment of the matrix sequence $\{\boldsymbol{X}_0, \boldsymbol{X}_1, \cdots, \boldsymbol{X}_{M-1}\}$ with the window length K. In fact, $\boldsymbol{X}_e$ is a generalized version of $\boldsymbol{X}$. If $L = 1$ and $K = M$, $\boldsymbol{X}_m$ becomes the mth row of $\boldsymbol{X}$, and $\boldsymbol{X}_e$ becomes $\boldsymbol{X}$. We call $\boldsymbol{X}_e$ the enhanced matrix because

$$\text{rank } (\boldsymbol{X}_e) = I \geq \text{rank } (\boldsymbol{X}) \tag{3.3}$$

in some important cases. The second relation in (3.3) has been discussed in Section II. The first relation in (3.3) will be discussed as follows.

Using (2.2) in (3.2), $\boldsymbol{X}_m$ becomes

$$\boldsymbol{X}_m = \boldsymbol{Z}_L \boldsymbol{A} \boldsymbol{Y}_d^m \boldsymbol{Z}_R \tag{3.4}$$

where $\boldsymbol{A}$ is the diagonal matrix of $\{a_i;\ i = 1, \cdots, I\}$ as defined in (2.9), and

$$\boldsymbol{Z}_L = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ z_1 & z_2 & \cdots & z_I \\ \vdots & \vdots & \cdots & \vdots \\ z_1^{L-1} & z_2^{L-1} & \cdots & z_I^{L-1} \end{bmatrix} \tag{3.5}$$

$$\boldsymbol{Y}_d = \text{diag } (y_1, y_2, \cdots, y_I) \tag{3.6}$$

$$\boldsymbol{Z}_R = \begin{bmatrix} 1 & z_1 & \cdots & z_1^{N-L} \\ 1 & z_2 & \cdots & z_2^{N-L} \\ & & \cdots & \\ 1 & z_I & \cdots & z_I^{N-L} \end{bmatrix}. \tag{3.7}$$

Then, using (3.4) in (3.1), $\boldsymbol{X}_e$ becomes

$$\boldsymbol{X}_e = \boldsymbol{E}_L \boldsymbol{A} \boldsymbol{E}_R \tag{3.8}$$

where

$$\boldsymbol{E}_L = \begin{bmatrix} \boldsymbol{Z}_L \\ \boldsymbol{Z}_L \boldsymbol{Y}_d \\ \cdots \\ \boldsymbol{Z}_L \boldsymbol{Y}_d^{K-1} \end{bmatrix} \tag{3.9}$$

$$\boldsymbol{E}_R = [\boldsymbol{Z}_R, \boldsymbol{Y}_d \boldsymbol{Z}_R, \cdots, \boldsymbol{Y}_d^{M-K} \boldsymbol{Z}_R]. \tag{3.10}$$

From (3.8), we know that rank $(\boldsymbol{X}_e) = I$ iff rank $(\boldsymbol{E}_L)$ = rank $(\boldsymbol{E}_R) = I$.

B. Conditions on K and L

Now we need to find the conditions on the free parameters K and L under which rank $(\boldsymbol{E}_L)$ = rank $(\boldsymbol{E}_R) = I$. Since the structures of $\boldsymbol{E}_L$ and $\boldsymbol{E}_R$ are similar, only $\boldsymbol{E}_L$ is considered for the moment. Obviously, the rank of $\boldsymbol{E}_L$ depends on the two parameters K and L. We will show that rank $(\boldsymbol{E}_L) = I$ if

$$K \geq I \quad \text{and} \quad L \geq I. \tag{3.11}$$

To show this, we need to introduce the permutation (shuffling) matrix:

$$\boldsymbol{P} = \begin{bmatrix} \boldsymbol{p}^T(1) \\ \boldsymbol{p}^T(1+L) \\ \cdots \\ \boldsymbol{p}^T(1+(K-1)L) \\ \boldsymbol{p}^T(2) \\ \boldsymbol{p}^T(2+L) \\ \cdots \\ \boldsymbol{p}^T(2+(K-1)L) \\ \cdots \\ \cdots \\ \boldsymbol{p}^T(L) \\ \boldsymbol{p}^T(L+L) \\ \cdots \\ \boldsymbol{p}^T(L+(K-1)L) \end{bmatrix} \tag{3.12}$$

where $\boldsymbol{p}(i)$ is the $KL \times 1$ vector with one at the ith position and zero everywhere else. The superscript ^{T} denotes the transposition. Left multiplying $\boldsymbol{E}_L$ by $\boldsymbol{P}$ (i.e., shuffling the rows of $\boldsymbol{E}_L$) yields $\boldsymbol{E}_{LP} = \boldsymbol{P}\boldsymbol{E}_L$ which can be shown to be

$$\boldsymbol{E}_{LP} = \begin{bmatrix} \boldsymbol{Y}_L \\ \boldsymbol{Y}_L\boldsymbol{Z}_d \\ \cdots \\ \boldsymbol{Y}_L\boldsymbol{Z}_d^{K-1} \end{bmatrix} \tag{3.13}$$

where

$$\boldsymbol{Y}_L = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ y_1 & y_2 & \cdots & y_I \\ \vdots & \vdots & \cdots & \vdots \\ y_1^{K-1} & y_2^{K-1} & \cdots & y_I^{K-1} \end{bmatrix} \tag{3.14}$$

$$\boldsymbol{Z}_d = \text{diag}\,(z_1, z_2, \cdots, z_I). \tag{3.15}$$

Note that the position of z_i in $\boldsymbol{E}_L$ is like that of y_i in $\boldsymbol{E}_{LP}$, and the position of y_i in $\boldsymbol{E}_L$ is like that of z_i in $\boldsymbol{E}_{LP}$. Since $\boldsymbol{Z}_L$ is a submatrix of $\boldsymbol{E}_L$, and $\boldsymbol{Y}_L$ is a submatrix of $\boldsymbol{E}_{LP}$ (a shuffled version of $\boldsymbol{E}_L$),

$$\text{rank}\,(\boldsymbol{E}_L) \geq \text{rank}\begin{bmatrix} \boldsymbol{Z}_L \\ \boldsymbol{Y}_L \end{bmatrix}. \tag{3.16}$$

Since $\{(y_i, z_i);\ i = 1, 2, \cdots, I\}$ are distinct, the I columns of $\begin{bmatrix} \boldsymbol{Z}_L \\ \boldsymbol{Y}_L \end{bmatrix}$ are linearly independent provided $L \geq I$ and $K \geq I$ (so that $\boldsymbol{Z}_L$ and $\boldsymbol{Y}_L$ each have no less than I rows). Hence, the sufficient condition (3.11) is proven.

The necessary condition for $\boldsymbol{E}_L$ to be of the full rank I is that the number of rows of $\boldsymbol{E}_L$ is no less than I, i.e., rank $(\boldsymbol{E}_L) = I$ only if

$$KL \geq I. \tag{3.17}$$

If K and L satisfy the necessary condition (3.17) but not the sufficient condition (3.11), rank $(\boldsymbol{E}_L)$ may or may not be equal to I.

Due to the similarity between $\boldsymbol{E}_L$ and $\boldsymbol{E}_R$, it can be similarly shown that rank $(\boldsymbol{E}_R) = I$ if

$$M - K + 1 \geq I \quad \text{and} \quad N - L + 1 \geq I \tag{3.18}$$

or only if

$$(M - K + 1)(N - L + 1) \geq I. \tag{3.19}$$

Since rank $(\boldsymbol{X}_e) = I$ iff rank $(\boldsymbol{E}_L) =$ rank $(\boldsymbol{E}_R) = I$, combining (3.11) with (3.18) yields that rank $(\boldsymbol{X}_e) = I$ if

$$M - I + 1 \geq K \geq I \quad \text{and} \quad N - I + 1 \geq L \geq I \tag{3.20}$$

and combining (3.17) with (3.19) yields that rank $(\boldsymbol{X}_e) = I$ only if

$$KL \geq I \quad \text{and} \quad (M - K + 1)(N - L + 1) \geq I. \tag{3.21}$$

If I is unknown but less than a number $I_{\max}$, K and L must satisfy the sufficient condition (3.20) so that I can be estimated from the singular values of $\boldsymbol{X}_e$. In the sequel (except in Section V), however, the number I will be assumed to be known.

C. Eigenstructure of $\boldsymbol{X}_e$

Before we apply the matrix pencil approach, in the next section, to extract the 2-D poles from $\boldsymbol{X}_e$, we need to study the eigen structure of $\boldsymbol{X}_e$. The singular value decomposition (SVD) [5] of $\boldsymbol{X}_e$ has the form

$$\begin{aligned} \boldsymbol{X}_e &= \sum_{i=1}^{\min} \sigma_i \boldsymbol{u}_i \boldsymbol{v}_i^H \\ &= \boldsymbol{U}_s \Sigma_s \boldsymbol{V}_s^H + \boldsymbol{U}_n \Sigma_n \boldsymbol{V}_n^H \end{aligned} \tag{3.22}$$

where the superscript ^{H} denotes the conjugate transpose; min = min $(KL, (M - K + 1)(N - L + 1))$ which is the smaller dimension of $\boldsymbol{X}_e$; $\boldsymbol{U}_s$, Σ_s, and $\boldsymbol{V}_s$ contain the I principal components; and $\boldsymbol{U}_n$, Σ_n, and $\boldsymbol{V}_n$ contain the remaining nonprincipal components. Specifically,

$$\boldsymbol{U}_s = [\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_I] \tag{3.22a}$$

$$\Sigma_s = \text{diag}\,[\sigma_1, \sigma_2, \cdots, \sigma_I] \tag{3.22b}$$

$$\boldsymbol{V}_s = [\boldsymbol{v}_1, \boldsymbol{v}_2, \cdots, \boldsymbol{v}_I] \tag{3.22c}$$

$$\boldsymbol{U}_n = [\boldsymbol{u}_{I+1}, \boldsymbol{u}_{I+1}, \cdots, \boldsymbol{u}_{\min}] \tag{3.22d}$$

$$\Sigma_n = \text{diag}\,[\sigma_{I+1}, \sigma_{I+2}, \cdots, \sigma_{\min}] \tag{3.22e}$$

$$\boldsymbol{V}_n = [\boldsymbol{v}_{I+1}, \boldsymbol{v}_{I+2}, \cdots, \boldsymbol{v}_{\min}] \tag{3.22f}$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min}$. For the noiseless case, $\sigma_i > 0$ for $i = 1, \cdots, I$, and $\sigma_i = 0$ for $i > I$, and hence Σ_n is zero. Comparing (3.22) with (3.8) yields that if rank $(\boldsymbol{X}_e) = I$,

$$\text{range}\,(\boldsymbol{X}_e) = \text{range}\,(\boldsymbol{E}_L) = \text{range}\,(\boldsymbol{U}_s) \tag{3.23}$$

and

$$\text{range}\,(\boldsymbol{X}_e^H) = \text{range}\,(\boldsymbol{E}_R^H) = \text{range}\,(\boldsymbol{V}_s). \tag{3.24}$$

Since $\boldsymbol{U}_s \perp \boldsymbol{U}_n$ and $\boldsymbol{V}_s \perp \boldsymbol{V}_n$ where $\perp$ denotes that the left is (columnwise) orthogonal to the right, $\boldsymbol{E}_L \perp \boldsymbol{U}_n$ and $\boldsymbol{E}_R^H \perp \boldsymbol{V}_n$. The above properties can be used to estimate the 2-D frequencies as follows. Note that only $\boldsymbol{U}_s$ will be used to produce the 2-D frequencies in the sequel although $\boldsymbol{V}_s$ can be used similarly. It seems that using both of the two matrices might yield better estimates, but such attempt has not been successful.

From the expression of $\boldsymbol{E}_L$ in (3.9), we know that the ith column of $\boldsymbol{E}_L$ is

$$\boldsymbol{e}_{Li} = \boldsymbol{y}_{Li} \otimes \boldsymbol{z}_{Li} \tag{3.25}$$

where $\otimes$ denotes the Kronecker product, and $\boldsymbol{y}_{Li}$ and $\boldsymbol{z}_{Li}$ are the ith column of $\boldsymbol{Y}_L$ and $\boldsymbol{Z}_L$, respectively. Because of (3.25), we define a similar vector:

$$\boldsymbol{e}_L = \boldsymbol{y}_L \otimes \boldsymbol{z}_L \tag{3.26}$$

where

$$\boldsymbol{y}_L = [1, y, \cdots, y^{K-1}]^T \tag{3.27}$$

$$\boldsymbol{z}_L = [1, z, \cdots, z^{L-1}]^T \tag{3.28}$$

$$y = \exp(j2\pi f_1) \tag{3.29}$$

$$z = \exp(j2\pi f_2). \tag{3.30}$$

Clearly, e_L is a function of the 2-D frequency variable (f_1, f_2). So, we may also write $e_L(f_1, f_2)$ in place of e_L to emphasize the relationship. It can be shown that if K and L satisfy the sufficient condition (3.20), e_L belongs to span $\{e_{L1}, e_{L2}, \cdots, e_{LI}\}$ iff $(f_1, f_2) = (f_{1i}, f_{2i})$, and more importantly

$$e_L \perp U_n \tag{3.31}$$

iff $(f_1, f_2) = (f_{1i}, f_{2i})$. This suggests that $\{(f_{1i}, f_{2i}); i = 1, \cdots, I\}$ can be found from the peak positions of the 2-D frequency spectrum:

$$\frac{1}{\min\limits_{i=I+1} \sum \|u_i^H e_L(f_1, f_2)\|^2}. \tag{3.32}$$

Estimating the 2-D frequencies by searching for the peaks of the 2-D spectrum of (3.32) is very costly in computation. The above approach is similar to the idea of MUSIC [11]. In the next section, we will use the matrix pencil approach to estimate the 2-D frequencies from the principal singular vectors of X_e (i.e., using U_S).

IV. Matrix Pencil

The matrix pencil approach can be stated as constructing two matrices in such a way that the desired numbers (e.g., poles) are the rank reducing numbers (i.e , the generalized eigenvalues or G.E.s) of the corresponding matrix pencil.

A. Extracting y_i

We have shown that if the condition (3.20) is satisfied, range (U_s) = range (E_L), and hence

$$U_s = E_L T \tag{4.1}$$

where T is an unique $I \times I$ nonsingular matrix. (Note that both U_s and E_L have I independent columns.) Knowing the structure of E_L shown in (3.9), we define

$$U_1 = U_s \text{ with the last } L \text{ rows deleted} \tag{4.2}$$

$$U_2 = U_s \text{ with the first } L \text{ rows deleted} \tag{4.3}$$

Using (4.1) and (3.9), we can write

$$U_1 = E_1 T \tag{4.4}$$

$$U_2 = E_1 Y_d T \tag{4.5}$$

where

$$E_1 = E_L \text{ with the last } L \text{ rows deleted.} \tag{4.6}$$

Then, it is clear that the matrix pencil $U_2 - \lambda U_1$ becomes

$$U_2 - \lambda U_1 = E_1(Y_d - \lambda I)T \tag{4.7}$$

where I is an identity matrix of proper dimension. Since Y_d is the diagonal matrix of the poles $\{y_i; i = 1, \cdots, I\}$, (4.7) shows (see [4]) that the poles $\{y_i; i = 1, \cdots, I\}$ are the rank reducing numbers of the matrix pencil $U_2 - \lambda U_1$ (i.e., the rank of the matrix pencil decreases by one iff $\lambda = y_i$), if E_1 and T are of the full rank I.

B. Extracting z_i

In order to extract the other set of poles $\{z_i; i = 1, \cdots, I\}$, we need to exploit the structure of E_{LP} in (3.13). We define

$$U_{sP} = PU_s \tag{4.8}$$

$$U_{1P} = U_{sP} \text{ with the last } K \text{ rows deleted} \tag{4.9}$$

$$U_{2P} = U_{sP} \text{ with the first } K \text{ rows deleted.} \tag{4.10}$$

Using (4.1) in (4.8), we have $U_{sP} = PE_L T = E_{LP} T$. Using this result and (3.13) in both (4.9) and (4.10), we can write

$$U_{1P} = E_{1P} T \tag{4.11}$$

$$U_{2P} = E_{1P} Z_d T \tag{4.12}$$

where

$$E_{1P} = E_{LP} \text{ with the last } K \text{ rows deleted.} \tag{4.13}$$

Then, we can write the matrix pencil $U_{2P} - \lambda U_{1P}$ as

$$U_{2P} - \lambda U_{1P} = E_{1P}(Z_d - \lambda I)T. \tag{4.14}$$

Since Z_d is the diagonal matrix of $\{z_i; i = 1, \cdots, I\}$, (4.14) shows that $\{z_i; i = 1, \cdots, I\}$ are the rank reducing numbers of $U_{2P} - \lambda U_{1P}$, if E_{1P} and T are of the full rank I.

C. Conditions on K and L

T has been known to be of the full rank I given K and L satisfying the condition (3.20). To find the conditions under which E_1 and E_{1P} are of the full rank I, we should compare the two matrices with E_L in (3.9). It can be shown (similarly as for (3.11)) that rank $(E_1) = I$ if

$$K - 1 \geq I \quad \text{and} \quad L \geq I \tag{4.15}$$

and rank $(E_{1P}) = I$ if

$$K \geq I \quad \text{and} \quad L - 1 \geq I. \tag{4.16}$$

Combining the above two conditions with (3.20), we obtain the overall sufficient condition (for the MEMP method to work):

$$\begin{cases} M - I + 1 > K \geq I + 1 \\ N - I + 1 \geq L - I + 1. \end{cases} \tag{4.17}$$

On the other hand, the necessary condition for rank (E_1) to be I is that the number of its rows is no less than I, i.e.,

$$(K - 1)L \geq I. \tag{4.18}$$

Similarly, the necessary condition for rank (E_{1P}) to be I is

$$K(L - 1) \geq I. \tag{4.19}$$

Combining the necessary conditions (4.18) and (4.19) and (3.21) yields the overall necessary condition:

$$\begin{cases} (K-1)L \geq I \\ K(L-1) \geq I \\ (M-K+1)(N-L+1) \geq I. \end{cases} \tag{4.20}$$

D. Pairing

We have now developed the MEMP method to extract $\{y_i; i = 1, \cdots, I\}$ and $\{z_i; i = 1, \cdots, I\}$ separately. The order of poles in each set is still unknown. Note that the y_i in $\{y_i; i = 1, \cdots, I\}$ is not necessarily the y_i in the (correct) pairs $\{(y_i, z_i); i = 1, \cdots, I\}$. To obtain the (correct but not necessarily ordered) pairs $\{(y_i, z_i); i = 1, \cdots, I\}$, we need to pair the two sets $\{y_i; i = 1, \cdots, I\}$ and $\{z_i; i = 1, \cdots, I\}$ together correctly. An optimum pairing approach may be such that a cost function of $\{(y_i, z_i); i = 1, \cdots, I\}$ is minimized by a choice among $I!$ (factorial) possibles. The cost function may be the sum of squared errors between the original data samples and the reconstructed data samples. However, if I is moderately large, $I!$ can be too large to carry out the pairing process. To speed up the pairing process, we suggest to use the property shown in (3.31). Specifically, we do the following. For $i = 1, 2, \cdots, I$, we minimize

$$J_n(i, j) = \min_{l=I+1} \sum \|\boldsymbol{u}_l^H \boldsymbol{e}_L(y_i, z_j)\|^2 \tag{4.21}$$

with respect to j, where $\boldsymbol{e}_L(y_i, z_j) = \boldsymbol{e}_L(f_{1i}, f_{2j})$. After expressing the algorithm in terms of the I principal eigenvectors $\{\boldsymbol{u}_i; i = 1, \cdots, I\}$, i.e., using $\boldsymbol{U}_n\boldsymbol{U}_n^H = \boldsymbol{I} - \boldsymbol{U}_s\boldsymbol{U}_s^H$, we can equivalently maximize, for $i = 1, 2, \cdots, I$,

$$J_s(i, j) = \sum_{l=1}^{I} \|\boldsymbol{u}_l^H \boldsymbol{e}_L(y_i, z_j)\|^2 \tag{4.22}$$

with respect to j to find the proper pairing.

The above maximization can be explicitly carried out as follows.

1) Set $i = 1$.
2) Compute $J_s(i, j)$ for $j = 1, 2, \cdots, I$.
3) Search for the largest value among $\{J_s(i, j); j = 1, 2, \cdots, I\}$ to obtain the pairing index $(i, j(i))$ and hence the pair $(y_i, z_{j(i)})$.
4) Set $i = i + 1$.
5) Compute $J_s(i, j)$ for $j = 1, 2, \cdots, I$ but $j \neq j(k)$ where $k = 1, 2, \cdots, i - 1$.
6) Search for the largest value among $\{J_s(i, j); j = 1, 2, \cdots, I$ but $j \neq j(k)$ where $k = 1, 2, \cdots, I - 1\}$ to obtain the pairing index $(i, j(i))$ and hence the pair $(y_i, z_{j(1)})$.
7) Go to Step 4) unless $i = I - 1$.

This procedure does not treat the poles $\{y_i; i = 1, 2, \cdots, I\}$ equally, i.e., y_i is treated more seriously (earlier) than y_{i+1}. If we have some *a priori* information about y_i, then we may be able to order them according to their priority before the above pairing procedure is applied. If we know more about z_i than y_i, then we should interchange the order of z_i and y_i in the above pairing procedure. The overall idea here is to insure that the pole we have the best confidence with gets the best mate.

V. The MEMP Algorithms

The MEMP method for estimating 2-D frequencies has been developed in the previous two sections assuming no noise. In the following, we will first summarize the MEMP method into an algorithm which processes the noisy data. Then, another modified algorithm will be developed for the case where the data set is large and the noise covariance sequence is known.

A. Algorithm 1

Step 1: Form the $KL \times (M - K + 1)(N - L + 1)$ enhanced matrix $\boldsymbol{X}_e'$ from the noisy data $x'(m; n)$ according to (3.1). K and L must satisfy the necessary condition (4.20). For more reliable and more accurate estimation, K and L should satisfy the sufficient condition (4.17). (The effects of K and L on the noise sensitivity will be discussed in Section VII.)

Step 2: Compute the singular values and left singular vectors of $\boldsymbol{X}_e'$. Estimate the number I of 2-D sinusoids from the singular values. (e.g., see [6]). Let $\boldsymbol{U}_s'$ be the matrix of the I principal left singular vectors (as $\boldsymbol{U}_s$ in (3.22a)).

Step 3: Form $\boldsymbol{U}_1'$, $\boldsymbol{U}_2'$ from $\boldsymbol{U}_s'$ according to (4.2) and (4.3). Form $\boldsymbol{U}_{sP}'$ from $\boldsymbol{U}_s'$ according to (4.8). Form $\boldsymbol{U}_{1P}'$ and $\boldsymbol{U}_{2P}'$ from $\boldsymbol{U}_{sP}'$ according to (4.9) and (4.10).

Step 4: Compute the generalized eigenvalues (GE's) of $\boldsymbol{U}_2' - \lambda\boldsymbol{U}_1'$ and $\boldsymbol{U}_{2P}' - \lambda\boldsymbol{U}_{1P}'$. Let $\{y_i'; i = 1, \cdots, I\}$ be the first set of GE's, and $\{z_i'; i = 1, \cdots, I\}$ the second. (Several algorithms for the generalized eigenvalue problem are available in [7] or [13]–[15]. The simplest is to use the QZ algorithm [5] to solve the GE's of $\boldsymbol{U}_1'^H\boldsymbol{U}_2' - \lambda\boldsymbol{U}_1'^H\boldsymbol{U}_1'$ and $\boldsymbol{U}_{1P}'^H\boldsymbol{U}_{2P}' - \lambda\boldsymbol{U}_{1P}'^H\boldsymbol{U}_{1P}'$. The simulation results shown in Section VII were obtained by using this algorithm.)

Step 5: For $i = 1, 2, \cdots, I$, maximize the function shown in (4.22) (using y_i' and z_i' in place of y_i and z_i respectively) with respect to j to obtain the correct pairs $\{(y_i, z_i)'; i = 1, \cdots, I\}$.

Step 6: Compute $(f_{1i}, f_{2i})'$ from $(y_i, z_i)'$ by using (2.3) and (2.4).

B. Algorithm 2

If the data set is very large and the noise covariance sequence is known, then the noise can be filtered out at the covariance level as follows.

Let $\boldsymbol{R}_e'$ be the covariance matrix of the noisy enhanced matrix $\boldsymbol{X}_e'$, i.e.,

$$\boldsymbol{R}_e' = \frac{1}{c}\boldsymbol{X}_e'\boldsymbol{X}_e'^H \tag{5.1}$$

where $c = (M - K + 1)(N - L + 1)$. If $(M - K)$ and $(N - L)$ are very large, then

$$\boldsymbol{R}'_e \cong \boldsymbol{R}_e + \gamma \boldsymbol{R}_{en} \tag{5.2}$$

where $\boldsymbol{R}_e$ is the covariance matrix of the noiseless matrix $\boldsymbol{X}_e$, i.e.,

$$\boldsymbol{R}_e = \frac{1}{c} \boldsymbol{X}_e \boldsymbol{X}_e^H \tag{5.3}$$

and $\gamma \boldsymbol{R}_{en}$ is the noise covariance matrix, and γ is a scalar. Note that $\gamma \boldsymbol{R}_{en}$ is a known function, although not given here, of the 2-D covariance sequence of the 2-D noise sequence $w(m; n)$.

Assume $\boldsymbol{R}_{en}$ is known. Then the effect of $\gamma \boldsymbol{R}_{en}$ on the estimation accuracy can be removed (completely if $M - K$ and $M - L$ are infinitely large and the noise is stationary and ergodic) by using the following algorithm.

Step 1: Same as in algorithm 1. But then compute $\boldsymbol{R}'_e$ and its transformed version:

$$\boldsymbol{R}'_{et} = \boldsymbol{R}_{en}^{-1/2} \boldsymbol{R}'_e \boldsymbol{R}_{en}^{-H/2}. \tag{5.4}$$

Step 2: Compute the eigendecomposition of $\boldsymbol{R}'_{et}$, i.e.,

$$\boldsymbol{R}'_{et} = \sum_{i=1}^{KL} \lambda'_{it} \boldsymbol{u}'_{it} \boldsymbol{u}'^H_{it}. \tag{5.5}$$

Estimate I from the eigenvalues $\lambda'_{1t} \geq \lambda'_{2t} \geq \cdots \geq \lambda'_{KLt}$. (The information criteria shown in [8] can be applied.) Note that asymptotically, $\lambda_{it} > \gamma$ for $i = 1, \cdots, I$, and $\lambda_{it} = \gamma$ for $i = I + 1, I + 2, \cdots, KL$. Let $\boldsymbol{U}'_{st}$ be the matrix of the I principal eigenvectors of $\boldsymbol{R}'_{et}$. Then, compute

$$\boldsymbol{U}'_{stt} = \boldsymbol{R}_{en}^{1/2} \boldsymbol{U}'_{st}. \tag{5.6}$$

Steps 3–6: Same as in algorithm 1 but use $\boldsymbol{U}''_{stt}$ in place of $\boldsymbol{U}'_s$.

C. Remarks

If the noise is white, then it is easy to show that $\boldsymbol{R}_{en} = \boldsymbol{I}$ and hence algorithm 2 is equivalent to algorithm 1.

In some applications, not both M and N are very large, but one of them is. In this case, algorithm 2 is still applicable.

Finally, we mention that the MEMP method also applies to damped 2-D sinusoids since the data structure we have exploited so far is shown in (2.2). Given that all the poles y_i and z_i are on the unit circle for the undamped 2-D sinusoids, the enhanced matrix $\boldsymbol{X}_e$ can be replaced by a further enhanced matrix $\boldsymbol{X}_{ee}$:

$$\boldsymbol{X}_{ee} = [\boldsymbol{X}_e, \boldsymbol{P}_e \boldsymbol{X}_e^*] \tag{5.7}$$

where * denotes the complex conjunction, and $\boldsymbol{P}_e$ is a permutation matrix defined by

$$\boldsymbol{P}_e = \begin{bmatrix} & & 1 \\ & 1 & \\ \cdots & & \\ 1 & & \end{bmatrix}. \tag{5.8}$$

In the noiseless case,

$$\text{range } (\boldsymbol{X}_{ee}) = \text{range } (\boldsymbol{X}_e) = \text{range } (\boldsymbol{E}_L) \tag{5.9}$$

so that the MEMP method based on either $\boldsymbol{X}_{ee}$ or $\boldsymbol{X}_e$ yields the same results. But in the noisy case, using $\boldsymbol{X}_{ee}$ enhances the robustness to noise in a similar way as using the forward-and-backward linear prediction equations in [16] or as using the forward-and-backward matrix pencil in [4]. The simulation results shown in Section VII were obtained by using $\boldsymbol{X}_{ee}$.

VI. Computational Order of the MEMP Method

In the following, we will first derive an estimate of the order of real multiplications needed by each major computational part of the MEMP algorithm 1 for 2-D frequency estimation. The real data is assumed although the notations for complex data are used. Then, we will compare the computational order of the MEMP method against that of a 2-D FFT method.

The major computations required by the MEMP algorithm 1 are a) computing the singular values and the left singular vectors of $\boldsymbol{X}_e$ which has the dimension $KL \times (M - K + 1)(N - L + 1)$; b) computing the generalized eigenvalues (GE's) of the matrix pencils $\boldsymbol{U}_2 - \lambda \boldsymbol{U}_1$ and $\boldsymbol{U}_{2P} - \lambda \boldsymbol{U}_{1P}$ both of which have the dimension $KL \times I$; and c) computing all necessary $J_s(i, j)$ to pair $\{y_i\}$ and $\{z_i\}$.

A. Computational Order of Major Part a)

According to the Chan SVD [5], computing the singular values and left singular vectors of $\boldsymbol{X}_e$ requires

$$K^2L^2(M - K + 1)(N - L + 1) + \tfrac{17}{3}K^3L^3 \tag{6.1}$$

multiplications, where $(M - K + 1)(N - L + 1) > KL$.

However, the computational order shown in (6.1) can be reduced in the following approach is used to perform major part a):

a1) compute $\boldsymbol{R}_e = \boldsymbol{X}_e \boldsymbol{X}_e^H$ (in a fast way); and a2) compute the eigenvalues and eigenvectors of $\boldsymbol{R}_e$.

Note that the left singular vectors of $\boldsymbol{X}_e$ are the eigenvectors of $\boldsymbol{X}_e \boldsymbol{X}_e^H$, and the singular values of $\boldsymbol{X}_e$ are the square roots of the eigenvalues of $\boldsymbol{X}_e \boldsymbol{X}_e^H$. (The numerical accuracy affected by the above approach is often negligible compared to the noise effects.)

A direct computation of $\boldsymbol{X}_e \boldsymbol{X}_e^H$ requires

$$\tfrac{1}{2}KL(KL + 1)(M - K + 1)(N - L + 1) \tag{6.2}$$

multiplications, which is in an order similar to that shown in (6.1).

To compute $\boldsymbol{X}_e \boldsymbol{X}_e^H$ faster, we need to observe that the (i_n, j_n)th element of the (i_m, j_m)th block of $\boldsymbol{X}_e \boldsymbol{X}_e^H$ (see (3.1)) is

$$\sum_{k_m=-1}^{M-K-1} \sum_{k_n=-1}^{N-L-1} x(i_m + k_m; i_n + k_n) x^*(j_m + k_m; j_n + k_n) \tag{6.3}$$

where $1 \le i_m \le K$, $1 \le j_m \le K$, $1 \le i_n \le L$ and $1 \le j_n \le L$. We can see that (almost) each multiplication in (6.3) is shared in computing several elements of $\boldsymbol{X}_e\boldsymbol{X}_e^H$. In fact, this is due to the Hankel structure in $\boldsymbol{X}_e$. The distinct multiplications required in computing $\boldsymbol{X}_e\boldsymbol{X}_e^H$ are

$$x(m; n)x^*(m - t_m; n - t_n) \tag{6.4}$$

where m, n, t_m, and t_n take the following sets of integers:

$$\{t_m = 0; \{m = 0, 1, \cdots, M - 1; \{t_n = 0, 1, \cdots, L - 1; \{n = t_n, t_n + 1, \cdots, N - 1\}\}\}\}$$

and

$$\{t_m = 1, 2, \cdots, K - 1;$$
$$\{m = t_m, t_m + 1, \cdots, M - 1;$$
$$\{t_n = 0, 1, \cdots, L - 1;$$
$$\{n = t_n, t_n + 1, \cdots, N - 1\}\};$$
$$\{t_n = -(L - 1), -(L + 2), \cdots, -1;$$
$$\{n = 0, 1, \cdots, N + t_n - 1\}\}\}\}.$$

Note that, for example, $\{t_m = 1, 2, \cdots, K - 1; \{m = t_m, t_m + 1, \cdots, M - 1\}\}$ means that for each value of t_m taken from $\{1, 2, \cdots, K - 1\}$, m takes all values from $\{t_m, t_m + 1, \cdots, M - 1\}$. The Hermitian property of $\boldsymbol{X}_e\boldsymbol{X}_e^H$ and its diagonal blocks has been considered to obtain the above result.

Now, the minimum number of multiplications required to compute $\boldsymbol{X}_e\boldsymbol{X}_e^H$ can be calculated as follows:

$$\sum_{m=0}^{M-1}\sum_{t_n=0}^{L-1}\sum_{n=t_n}^{N-1} 1 + \sum_{t_m=1}^{K-1}\sum_{m=t_m}^{M-1}\left[\sum_{t_n=0}^{L-1}\sum_{n=t_n}^{N-1} 1 + \sum_{t_n=-(L-1)}^{-1}\sum_{n=0}^{N+t_n-1} 1\right] \tag{6.5}$$

which can be simplified as follows, assuming that $K >> 1$, $L >> 1$, $M - K >> 1$ and $N - L >> 1$,

$$2KL\left(M - \frac{K}{2}\right)\left(N - \frac{L}{2}\right). \tag{6.6}$$

We can see that (6.6) is much smaller than (6.2).

To compute the eigenvalues and eigenvectors of $\boldsymbol{R}_e$, the number of multiplications required (based on the symmetric QR algorithm [5] and $KL >> 1$) is in the order of

$$5K^3L^3. \tag{6.7}$$

Combining (6.6) and (6.7) yields that a1) and a2) can be carried out by using

$$2KL\left(M - \frac{K}{2}\right)\left(N - \frac{L}{2}\right) + 5K^3L^3 \tag{6.8}$$

multiplications.

It is clear that if $K >> 1$ and $L >> 1$, (6.8) is much smaller than (6.1). Equation (6.8) is the estimated order of multiplications required by the major part a).

B. Computational Order of Major Part b)

The computational order of the matrix pencils $\boldsymbol{U}_2 - \lambda\boldsymbol{U}_1$ and $\boldsymbol{U}_{2P} - \lambda\boldsymbol{U}_{1P}$ are computed by using the QZ algorithm [5] on the $I \times I$ matrix pencils $\boldsymbol{U}_1^H\boldsymbol{U}_2 - \lambda\boldsymbol{U}_1^H\boldsymbol{U}_1$ and $\boldsymbol{U}_{1P}^H\boldsymbol{U}_{2P} - \lambda\boldsymbol{U}_{1P}^H\boldsymbol{U}_{1P}$, then it requires I^2KL multiplications for each of $\boldsymbol{U}_1^H\boldsymbol{U}_2$ and $\boldsymbol{U}_{1P}^H\boldsymbol{U}_{2P}$, and $[I(I + 1)/2]KL$ multiplications for each of (Hermitian) $\boldsymbol{U}_1^H\boldsymbol{U}_1$ and $\boldsymbol{U}_{1P}^H\boldsymbol{U}_{1P}$, and $5I^3$ multiplications [5] (assuming $I >> 1$) for computing the GE's of each $I \times I$ matrix pencil. Hence, the computational order of the major part b) is

$$3I^2KL + 10I^3 \approx 3I^2KL \tag{6.9}$$

provided $K > I >> 1$ and $L > I >> 1$.

C. Computational Order of Major Part c)

For each (i, j), we need $(K - 1)(L - 1)$ complex multiplications to obtain $\boldsymbol{e}_L(y_i, z_j)$ from $\boldsymbol{y}_{Li}$ and $\boldsymbol{z}_{Li}$ (see (3.26)), and $I(KL + 1)$ complex multiplications to obtain $J_s(i, j)$ from $\{\boldsymbol{u}_t; t = 1, 2, \cdots, I\}$ and $\boldsymbol{e}_L(y_i, z_j)$ (see (4.22)).

Since $J_s(i, j)$ must be computed for $I + (I - 1) + (I - 2) + \cdots + 2 = \frac{1}{2}I(I + 1) - 1$ different values of (i, j), we need

$$(\tfrac{1}{2}I(I + 1) - 1)((K - 1)(L - 1) + I(KL + 1)) \approx \tfrac{1}{2}I^3KL \tag{6.10}$$

complex multiplications to obtain all necessary $J_s(i, j)$ from $\{\boldsymbol{y}_{Li}$ and $\boldsymbol{z}_{Li}; i = 1, 2, \cdots, I\}$, assuming that $I >> 1$, $K >> 1$, and $L >> 1$.

It takes $I(K - 2)$ complex multiplications to obtain $\{\boldsymbol{y}_{Li}; i = 1, 2, \cdots, I\}$ from $\{y_i; i = 1, 2, \cdots, I\}$, and $I(L - 2)$ complex multiplications to obtain $\{\boldsymbol{z}_{Li}; i = 1, 2, \cdots, I\}$ from $\{z_i; i = 1, 2, \cdots, I\}$. The above computations are negligible compared to $\frac{1}{2}I^3KL$ in (6.10). Hence, (6.10) is the order of complex multiplications needed by the major part c).

D. Computational Order of the MEMP Method

Combining (6.8)–(6.10) yields the order of (real) multiplications required by the MEMP algorithm 1:

$$2KL\left(M - \frac{K}{2}\right)\left(N - \frac{L}{2}\right) + 5K^3L^3 + 3I^2KL + \frac{3}{2}I^3KL \approx 2KL\left(M - \frac{K}{2}\right)\left(N - \frac{L}{2}\right) + 5K^3L^3 \tag{6.11}$$

provided $K > I >> 1$, $L > I >> 1$, $M >> 1$, and $N >> 1$. (Note that as an engineering approximation, the "much larger" notation $>>$ means at least ten times larger.) We see that the computational order for the major part a) is dominant in the overall computational order of the MEMP method. We also can see that if $M >> K$ and $N >> L$, the dominant computation is to obtain the covariance matrix $\boldsymbol{R}_e$ (or $\boldsymbol{R}_e'$ in the noisy case).

E. Comparison to 2-D FFT Method

We consider a 2-D FFT method which typically requires [18]

$$\tfrac{1}{2}(\log_2 MN)MN \tag{6.12}$$

multiplications to produce a 2-D frequency spectrum. Note that (6.12) does not include the amount of computations required to search for I 2-D (peak) frequencies. It is clear that if I is much smaller than M and N, then K and L of the MEMP method (see (4.17)) can also be much smaller than M and N, and hence (6.11) can be smaller than (6.12). In fact, the ratio of the computational order of the MEMP method over the 2-D FFT method is, assuming that $1 \ll I < K \ll M$ and $1 \ll I < L \ll N$,

$$\frac{4KL}{\log_2 MN}. \tag{6.13}$$

This means that if the number of 2-D sinusoids is much smaller than the data size, the MEMP method can be faster than the 2-D FFT method.

VII. Simulation Results

The MEMP method has been developed based on the noiseless data. Without noise, the MEMP method yields the exact 2-D frequencies. When the data is noisy, the estimated 2-D frequencies have bias and variance. To evaluate the noise sensitivity of the MEMP method, the Monte Carlo simulation has been carried out. We have found that the estimation accuracy of the MEMP method can be very close to the CRB when K and L are not close to their boundary values shown in (4.17). A set of equations useful for computing the CRB will be given in Appendix A. A typical example of our simulation results is as follows.

A set of 20×20 data samples are

$$x'(m; n) = \sum_{i=1}^{3} \exp\,(j2\pi f_{1i}m + j2\pi f_{2i}n) + w(m; n) \tag{7.1}$$

where $0 \le m \le 19$, and $0 \le n \le 19$, and $w(m; n)$ is the white noise sequence, and

$$(f_{11}, f_{21}) = (0.26, 0.24) \tag{7.2}$$

$$(f_{12}, f_{22}) = (0.24, 0.24) \tag{7.3}$$

$$(f_{13}, f_{23}) = (0.24, 0.26). \tag{7.4}$$

This data set consists of three 2-D sinusoids which are too close to each other to be resolved by 2-D FFT since the Rayleigh limit for a 20×20 data set is 0.05. The three 2-D sinusoids can neither be resolved by the state space method [2] or the matrix approximation method [3] since the first two 2-D frequencies have the second 1-D frequency component in common and the last two 2-D frequencies have the first 1-D frequency component in common.

Although other methods as in [1], [9]–[11] can be applied to this data set to estimate the three 2-D frequencies, they are too costly in computation. The simulation comparison of the MEMP method against those methods has not been obtained.

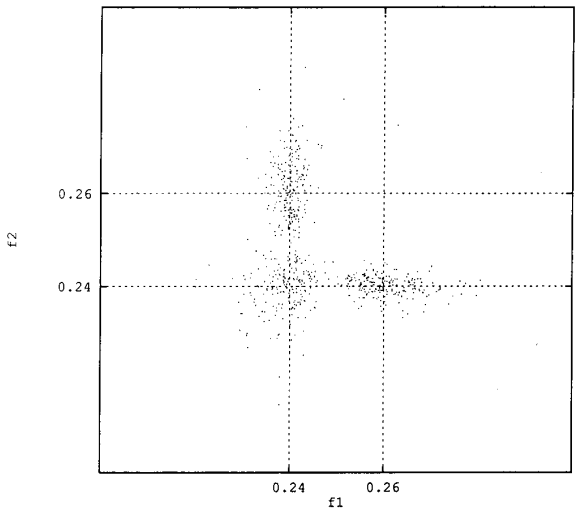

Fig. 1. Two hundred independent estimates of three 2-D frequencies. $K = L = 3$. SNR = 20 dB.

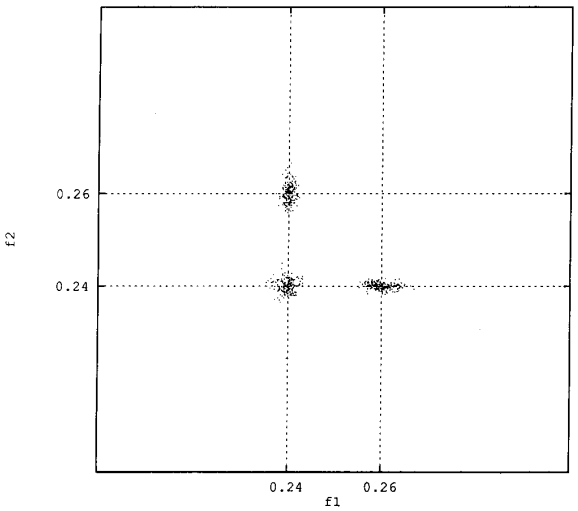

Fig. 2. Two hundred independent estimates of three 2-D frequencies. $K = L = 4$. SNR = 20 dB.

Figs. 1–4 show the estimated frequencies for 200 independent runs at SNR = 20 dB, which were obtained by using the MEMP Algorithm 1 and $\boldsymbol{X}_{ee}$. SNR is defined by

$$\text{SNR} = 10 \log_{10} \frac{1}{\gamma} \tag{7.5}$$

where γ is the variance of the complex white noise. For Figs. 1–4, the two parameters (K, L) are equal to (3, 3), (4, 4), (5, 5), and (6, 6), respectively. Due to the symmetry of the 2-D data samples, we have chosen $K = L$. It is clear from Figs. 1–4 that as K and L increase, the

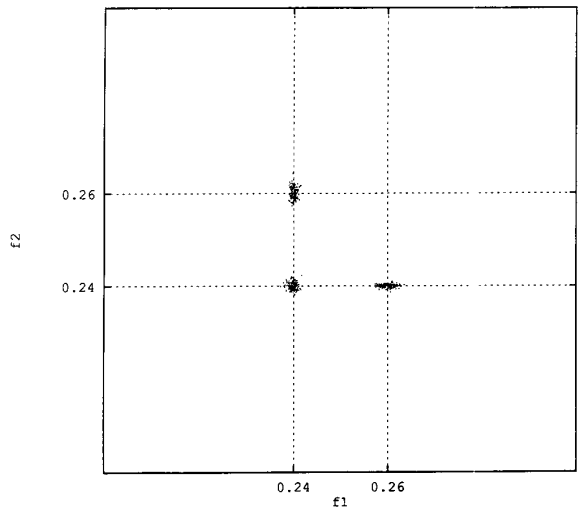

Fig. 3. Two hundred independent estimates of three 2-D frequencies. $K = L = 5$. SNR = 20 dB.

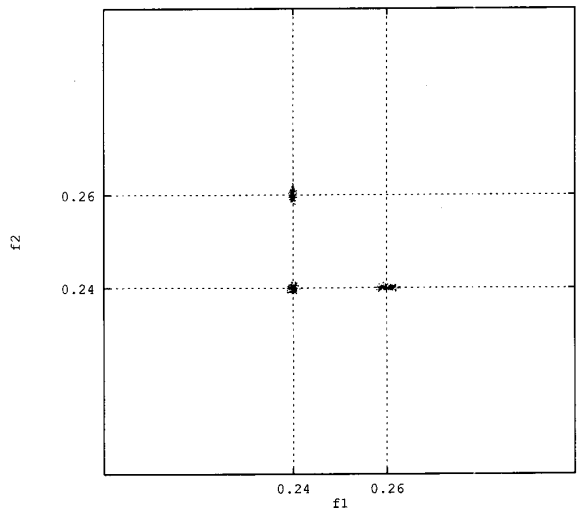

Fig. 4. Two hundred independent estimates of three 2-D frequencies. $K = L = 6$. SNR = 20 dB.

estimation accuracy increases (i.e., the three clusters in each figure become smaller). For $K = L = 6$, the biases and deviations are shown with the corresponding CRB's in Table I.

Figs. 5–9 show the estimated frequencies for 200 independent runs at SNR = 10 dB, which were obtained by using the MEMP algorithm 1 and X_{ee}. (K, L) are equal to (3, 3), (4, 4), (5, 5), (6, 6), and (7, 7), respectively. In Fig. 5 where $K = L = 3$, the 200-run estimated frequencies tend to cluster around the centroid of the three 2-D frequencies. After we increase K and L from 3 to 4, Fig. 6 shows that the dense cluster in Fig. 5 has been scattered (and the estimated frequencies start to reorganize themselves). When $K = L = 5$, Fig. 7 shows that the estimated frequencies start to cluster around each of three correct 2-D frequencies. When $K = L = 6$, Fig. 8 shows three denser clusters around each correct position. When $K = L = 7$, Fig. 9 shows that the three clusters are further compressed towards each correct position. The biases and deviations for $K = L = 7$ are shown with the corresponding CRB's in Table II.

From Tables I and II we can see that the estimation deviations are very close to the corresponding CRB's.

We also can see from Figs. 1–9 that K and L are like two tuning parameters which can be adjusted to increase the estimation accuracy. In fact, when $KL > I$, there is a $(\min - I)$-dimensional noise subspace spanned by the columns of U_n and an I-dimensional signal subspace spanned by the columns of U_s (see (3.22)). Intuitively, the larger the noise subspace is, the more noise component is absorbed into the noise subspace and the less noise component remains in the signal subspace. Since only the signal subspace is used in the MEMP method, larger noise subspace implies higher estimation accuracy. We can increase the noise subspace, and hence (intuitively) the estimation accuracy, by increasing min or equivalently K and L if $KL < (M - K + 1)(N - L + 1)$. This intuitive thinking explains the simulation results shown in Figs. 1–9. In fact, the above intuitive thinking represents a signal processing approach, which we call signal processing via inflating noise subspace (SPINS).

Note that the noise subspace is maximized when $KL = (M - K + 1)(N - L + 1)$ or when $K = \frac{1}{2}(M + 1)$ and $L = \frac{1}{2}(N + 1)$. Also note that if $KL > (M - K + 1)(N - L + 1)$, not only the estimation accuracy is reduced (due to that the noise subspace is reduced) but also the computations are increased. Hence, we should normally choose K and L such that

$$\begin{cases} \frac{1}{2}(M + 1) \geq K \geq I + 1 \\ \frac{1}{2}(N + 1) \geq L \geq I + 1. \end{cases} \tag{7.6}$$

As long as the computational burden is tolerable, K and L can be increased, from $I + 1$ to the maximum values $\frac{1}{2}(M + 1)$ and $\frac{1}{2}(N + 1)$, to reduce the noise effects.

VIII. Conclusions

We have developed the MEMP method for estimating 2-D frequencies. An enhanced matrix has been introduced to remove the ill condition of the original data matrix. The matrix pencil approach has been applied to efficiently extract out the 2-D frequencies. The MEMP method is computationally efficient mainly because searching in a 2-D space is not required. An estimated order of the computations required by the MEMP method has been derived. We have shown that if the number of 2-D sinusoids is much smaller than the size of the data set, the MEMP method can be faster than a 2-D FFT method.

The noise sensitivity of the MEMP method has been studied by the Monte Carlo simulation. The simulation

TABLE I
BIASES AND DEVIATIONS OF 200 INDEPENDENT ESTIMATES OF THREE 2-D FREQUENCIES. THE CRB'S SHOWN HERE ARE THE CRB'S ON DEVIATIONS. $K = L = 6$. SNR = 20 dB

f_1	Bias $\times 10^{-4}$	Dev $\times 10^{-3}$	CRB $\times 10^{-3}$	f_2	Bias $\times 10^{-4}$	Dev $\times 10^{-3}$	CRB $\times 10^{-3}$
0.26	0.31	1.05	0.40	0.24	0.01	0.31	0.32
0.24	−0.15	0.50	0.31	0.24	0.05	0.53	0.31
0.24	−0.20	0.31	0.32	0.26	−0.12	0.79	0.40

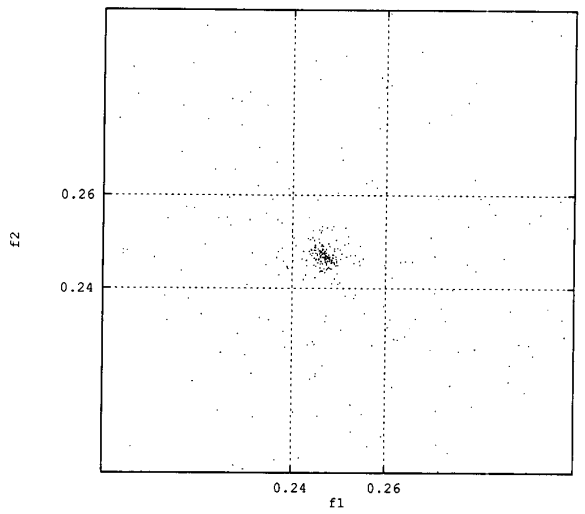

Fig. 5. Two hundred independent estimates of three 2-D frequencies. $K = L = 3$. SNR = 10 dB.

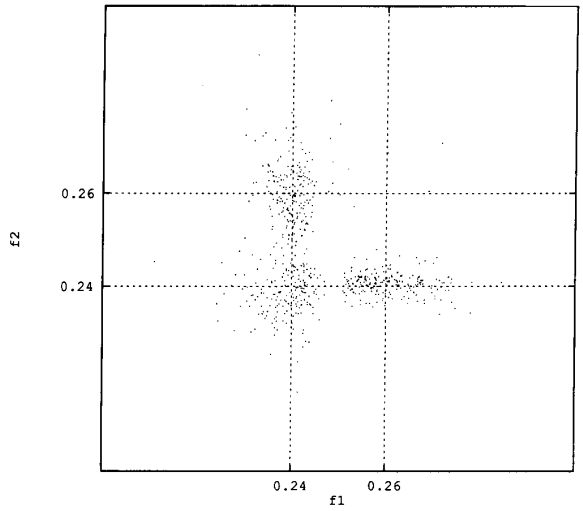

Fig. 7. Two hundred independent estimates of three 2-D frequencies. $K = L = 5$. SNR = 10 dB.

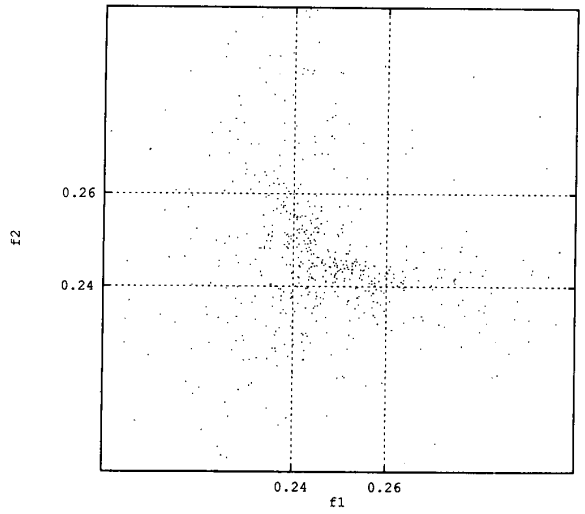

Fig. 6. Two hundred independent estimates of three 2-D frequencies. $K = L = 4$. SNR = 10 dB.

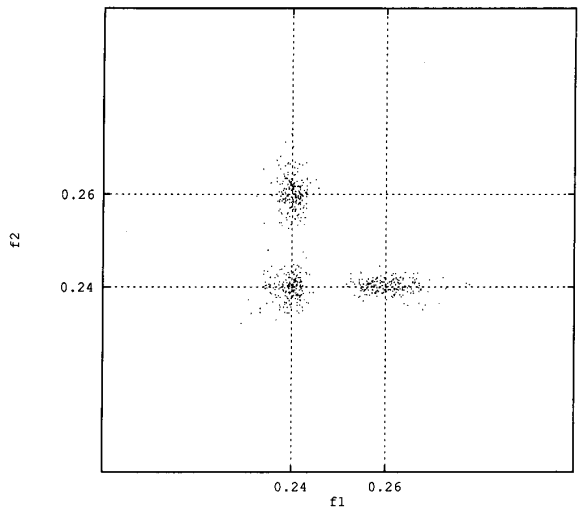

Fig. 8. Two hundred independent estimates of three 2-D frequencies. $K = L = 6$. SNR = 10 dB.

results show that the MEMP method is robust to noise and its accuracy can be very close to the CRB.

Finally, we add that the MEMP method can be extended to estimate arbitrary dimensional frequencies. Further research is under way in this direction. A severe problem in 3-D frequency estimation associated with applications such as 3-D radar imaging is data missing in the collected 3-D data set.

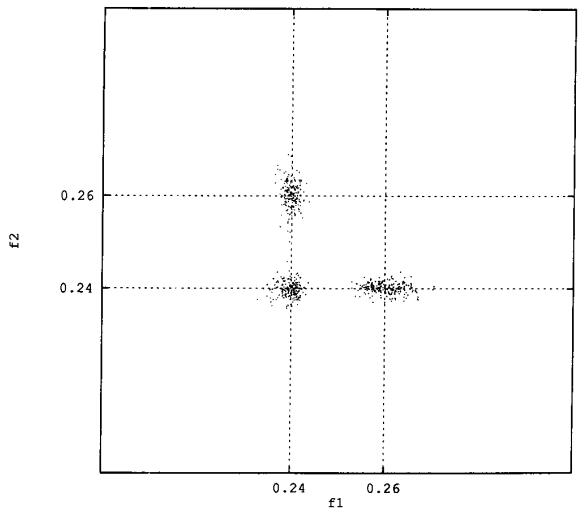

Fig. 9. Two hundred independent estimates of three 2-D frequencies. $K = L = 7$. SNR = 10 dB.

TABLE II
BIASES AND DEVIATIONS OF 200 INDEPENDENT ESTIMATES OF THREE 2-D FREQUENCIES. THE CRB'S SHOWN HERE ARE THE CRB'S ON DEVIATIONS. $K = L = 7$. SNR = 10 dB

f_1	Bias $\times 10^{-3}$	Dev $\times 10^{-2}$	CRB $\times 10^{-2}$	f_2	Bias $\times 10^{-3}$	Dev $\times 10^{-2}$	CRB $\times 10^{-2}$
0.26	0.22	0.33	0.13	0.24	0.06	0.11	0.10
0.24	−0.22	0.18	0.10	0.24	−0.29	0.16	0.10
0.24	−0.09	0.13	0.10	0.26	0.29	0.29	0.13

APPENDIX A
CRB FOR 2-D FREQUENCY ESTIMATION

For the 2-D frequency estimation problem, the CRB is not readily available in literature. A set of equations which can be easily used to compute the CRB are given in the following.

We write

$$\boldsymbol{x} = \text{vec}\ \{x(m; n);\ m, n\} \tag{A.1}$$

$$\boldsymbol{w} = \text{vec}\ \{w(m; n);\ m, n\} \tag{A.2}$$

where vec { } denotes a vector filled with the corresponding elements. Since $x'(m, n) = x(m, n) + w(m, n)$, we write

$$\boldsymbol{x}' = \boldsymbol{x} + \boldsymbol{w}. \tag{A.3}$$

Assume that the noise $\boldsymbol{w}$ is the (complex) white Gaussian, then the probability density function of $\boldsymbol{x}'$ is

$$p(\boldsymbol{x}'|\theta) = \frac{1}{(2\pi\gamma)^{MN}} \exp\left(-\frac{1}{\gamma}\|\boldsymbol{x}' - \boldsymbol{x}|^2\right) \tag{A.4}$$

where $\|\ \|$ denotes the 2-norm, γ is the noise variance, and θ is the $4I \times 1$ vector of unknown parameters, i.e.,

$$\theta = \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_I \end{bmatrix} \tag{A.5}$$

and

$$\theta_i = \begin{bmatrix} r_i \\ \phi_i \\ f_{1i} \\ f_{2i} \end{bmatrix}. \tag{A.6}$$

The corresponding $4I \times 4I$ Fisher information matrix $\boldsymbol{F}$ is defined as follows [21]:

$$F_{ij} = -\text{E}\left\{\frac{\partial^2}{\partial\theta_i\,\partial\theta_j} \log\,(p(\boldsymbol{x}'|\theta))\right\} \tag{A.7}$$

where F_{ij} is the (i, j)th element of $\boldsymbol{F}$, E{ } the expectation, $(\partial/\partial\theta_i)$ the partial derivative with respect to the ith element θ_i of θ, and log ( ) the natural logarithm. It can be shown, using (A.4) in (A.7), that

$$F_{ij} = \frac{1}{\gamma} 2\ \text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial\theta_i}\frac{\partial \boldsymbol{x}}{\partial\theta_j}\right] \tag{A.8}$$

where Re [ ] denotes the real part.

Using (2.1) in (A.1), we can show the following:

$$\frac{\partial \boldsymbol{x}}{\partial r_i} = \text{vec}\ \{\exp\,[j(\phi_i + 2\pi f_{1i}m + 2\pi f_{2i}n)];\ m, n\} \tag{A.9}$$

$$\frac{\partial \boldsymbol{x}}{\partial \phi_i} = \text{vec}\,\{jr_i \exp[j(\phi_i + 2\pi f_{1i}m + 2\pi f_{2i}n)];\, m,\, n\} \tag{A.10}$$

$$\frac{\partial \boldsymbol{x}}{\partial f_{1i}} = \text{vec}\,\{j2\pi m r_i \exp[j(\phi_i + 2\pi f_{1i}m + 2\pi f_{2i}n)];\, m,\, n\} \tag{A.11}$$

$$\frac{\partial \boldsymbol{x}}{\partial f_{2i}} = \text{vec}\,\{j2\pi n r_i \exp[j(\phi_i + 2\pi f_{1i}m + 2\pi f_{2i}n)];\, m,\, n\}. \tag{A.12}$$

By using (A.9)–(A.12), the following can be shown:

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial r_i}\frac{\partial \boldsymbol{x}}{\partial r_j}\right] = 2\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\cos(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n) \tag{A.13}$$

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial r_i}\frac{\partial \boldsymbol{x}}{\partial \phi_j}\right] = 2r_j\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\sin(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n) \tag{A.14}$$

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial r_i}\frac{\partial \boldsymbol{x}}{\partial f_{1j}}\right] = 2r_j 2\pi\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} m\sin(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n) \tag{A.15}$$

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial r_i}\frac{\partial \boldsymbol{x}}{\partial f_{2j}}\right] = 2r_j 2\pi\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} n\sin(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n) \tag{A.16}$$

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial \phi_i}\frac{\partial \boldsymbol{x}}{\partial \phi_j}\right] = 2r_i r_j\sum_{m=0}^{M-1}\sum_{n=0}^{N-1}\cos(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n) \tag{A.17}$$

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial \phi_i}\frac{\partial \boldsymbol{x}}{\partial f_{1j}}\right] = 2r_i r_j 2\pi\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} m\cos(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n) \tag{A.18}$$

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial \phi_i}\frac{\partial \boldsymbol{x}}{\partial f_{2j}}\right] = 2r_i r_j 2\pi\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} n\cos(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n) \tag{A.19}$$

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial f_{1i}}\frac{\partial \boldsymbol{x}}{\partial f_{1j}}\right] = 2r_i r_j(2\pi)^2\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} m^2\cos(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n) \tag{A.20}$$

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial f_{1i}}\frac{\partial \boldsymbol{x}}{\partial f_{2j}}\right] = 2r_i r_j(2\pi)^2\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} mn\cos(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n) \tag{A.21}$$

$$2\,\text{Re}\left[\frac{\partial \boldsymbol{x}^H}{\partial f_{2i}}\frac{\partial \boldsymbol{x}}{\partial f_{2j}}\right] = 2r_i r_j(2\pi)^2\sum_{m=0}^{M-1}\sum_{n=0}^{N-1} n^2\cos(\phi_i - \phi_j + 2\pi(f_{1i} - f_{1j})m + 2\pi(f_{2i} - f_{2j})n). \tag{A.22}$$

Using (A.13)–(A.22) in (A.8), the $4I \times 4I$ Fisher information $\boldsymbol{F}$ can be straightforwardly formed. The CRB on the variance of the unbiased estimate of the ith parameter θ_i is the ith diagonal element of the inverse $\boldsymbol{F}^{-1}$ which can be computed numerically. The CRB's on the corresponding deviations are the square roots of the diagonal elements of $\boldsymbol{F}^{-1}$. The CRB's listed in Tables I, II are the CRB's on deviations.

Appendix B
Estimating Amplitudes

The signal amplitudes $\{a_i;\, i = 1, \cdots, I\}$ can be simply obtained (among other ways) as follows. Once $\{(y_i, z_i);\, i = 1, \cdots, I\}$ are known or estimated, $\boldsymbol{E}_L$ and $\boldsymbol{E}_R$ can be reconstructed according to (3.5) and (3.7), respectively. From (3.8), we know that

$$\boldsymbol{A} = \boldsymbol{E}_L^+ \boldsymbol{X}_e \boldsymbol{E}_R^+ \tag{B.1}$$

where the superscript $^+$ denotes the Moore–Penrose inverse, and

$$\boldsymbol{E}_L^+ = (\boldsymbol{E}_L^H \boldsymbol{E}_L)^{-1}\boldsymbol{E}_L^H \tag{B.2}$$

$$\boldsymbol{E}_R^+ = \boldsymbol{E}_R^H(\boldsymbol{E}_R\boldsymbol{E}_R^H)^{-1}. \tag{B.3}$$

$\{a_i;\, i = 1, \cdots, I\}$ are the diagonal elements of $\boldsymbol{A}$.

If the noise level is high, and the noise covariance matrix is known (except a scalar), and the data set is large, then we can do the following. It can be shown, using (5.2)

and (5.5), that the signal only covariance matrix $\boldsymbol{R}_e$ can be obtained from $\boldsymbol{X}_e'$ as follows:

$$\boldsymbol{R}_e = \sum_{i=1}^{I} (\lambda_{it} - \gamma)\boldsymbol{u}_{it}\boldsymbol{u}_{it}^H \tag{B.4}$$

where the notations are defined in (5.5). Equation (B.4) is the asymptotical result so that the primes are dropped. In the asymptotical case, $\gamma = \lambda_{it}$ for $i = I - 1, \cdots, KL$. Then, using (3.8) in (5.3), we obtain

$$\frac{1}{c}\boldsymbol{E}_L\boldsymbol{A}\boldsymbol{E}_R\boldsymbol{E}_R^H\boldsymbol{A}^H\boldsymbol{E}_L^H = \boldsymbol{R}_e \tag{B.5}$$

and hence

$$\frac{1}{c}\boldsymbol{A}\boldsymbol{E}_R\boldsymbol{E}_R^H\boldsymbol{A}^H = \boldsymbol{E}_L^+\boldsymbol{R}_e(\boldsymbol{E}_L^H)^+. \tag{B.6}$$

Using (3.10), we can write

$$\frac{1}{c}\boldsymbol{E}_R\boldsymbol{E}_R^H = \frac{1}{c}\sum_{m=0}^{M-K}\boldsymbol{Y}_d^m\boldsymbol{Z}_R\boldsymbol{Z}_R^H\boldsymbol{Y}_d^{mH}. \tag{B.7}$$

Since $N - L$ is assumed to be very large, using (3.7) yields

$$\boldsymbol{Z}_R\boldsymbol{Z}_R^H = (N - L + 1)\boldsymbol{I} \tag{B.8}$$

where z_i's in $\boldsymbol{Z}_R$ are assumed to be distinct. Using (B.8) in (B.7) leads to

$$\frac{1}{c}\boldsymbol{E}_R\boldsymbol{E}_R^H = \boldsymbol{I}. \tag{B.9}$$

Hence, combining (B.9) and (B.6) yields

$$\boldsymbol{A}\boldsymbol{A}^H = \boldsymbol{E}_L^+\boldsymbol{R}_e(\boldsymbol{E}_L^H)^+. \tag{B.10}$$

The absolute amplitudes $|a_i|$ are obtained from the diagonal elements of (B.10), but the phases are lost in (B.10).

ACKNOWLEDGMENT

The author would like to acknowledge Dr. T. K. Sarkar's encouragement for this work.

REFERENCES

[1] S. L. Marple, *Digital Spectral Analysis with Applications*. Englewood Cliffs, NJ: Prentice-Hall, 1988, ch. 16.
[2] S. Y. Kung, K. S. Arun, and D. V. B. Rao, "State-space and singular-value decomposition-based approximation methods for the harmonic retrieval problem," *J. Opt. Soc. Amer.*, vol. 73, no. 12, pp. 1799–1811, Dec. 1983.
[3] A. K. Shaw and R. Kumaresan, "Frequency-wavenumber estimation by structured matrix approximation," in *Proc. IEEE ASSP 3rd Workshop Spectral Estimation Modelling* (Boston, MA), Nov. 1986, pp. 81–84.
[4] Y. Hua and T. K. Sarkar, "Matrix pencil method for estimating parameters for exponentially damped/undamped sinusoids in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, no. 5, pp. 814–824, May 1990.
[5] G. H. Golub and C. F. VanLoan, *Matrix Computations*. Baltimore, MD: Johns Hopkins, 1983.
[6] K. Konstantinides and K. Yao, "Statistical analysis of effective singular values in matrix rank determination," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, no. 5, pp. 757–763, May 1988.
[7] Y. Hua and T. K. Sarkar, "On SVD for estimating the generalized eigenvalues of singular matrix pencil in noise," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 39, no. 4, pp. 892–900, Apr. 1991.
[8] M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, no. 2, pp. 387–392, Apr. 1985.
[9] R. Kumaresan and D. W. Tufts, "A two-dimensional technique for frequency-wave number estimation," *Proc. IEEE*, vol. 69, pp. 1515–1517, Nov. 1981.
[10] S. W. Lang and J. H. McClellan, "The extension of Pisarenko's method to multiple dimensions," in *Proc. ICASSP-82*, pp. 125–128.
[11] R. O. Schmidt, "A signal subspace approach to multiple emitter location and spectral estimation," Ph.D. dissertation, Stanford University, 1981.
[12] S. Kay and R. Nekovei, "An efficient two-dimensional frequency estimator," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, no. 10, pp. 1807–1809, Oct. 1990.
[13] Y. Hua and T. K. Sarkar, "On SVD for estimating the generalized eigenvalues of singular matrix pencil in noise," in *Proc. 24th Annu. Asilomar Conf. Signals, Syst., Comput.* (Pacific Grove, CA), Nov. 1990.
[14] R. Roy and T. Kailath, "Total least squares ESPRIT," in *Proc. 21st Annu. Asilomar Conf. Signals, Syst., Comput.* (Pacific Grove, CA), Nov. 1987.
[15] M. Zoltawski and D. Stavrinides, "Sensor array signal processing via a Procrustes rotations based eigen-analysis of the ESPRIT data pencil," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 6, pp. 832–861, June 1989.
[16] D. W. Tufts and R. Kumaresan, "Estimation of frequencies of multiple sinusoids: Making linear prediction perform like maximum likelihood," *Proc. IEEE*, vol. 70, pp. 975–989, Sept. 1982.
[17] J. McClellan, "Multidimensional spectral estimation," *Proc. IEEE*, vol. 70, no. 9, pp. 1029–1039, Sept. 1982.
[18] D. E. Dudgeon and R. M. Merseresau, *Multidimensional Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
[19] T.-J. Shan, M. Wax, and T. Kailath, "On spatial smoothing for direction-of-arrival estimation of coherent signals," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, pp. 806–811, Aug. 1985.
[20] H. Wang and M. Kaveh, "Coherent signal-subspace processing for the detection and estimation of angles of arrival of multiple wideband sources," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, no. 4, pp. 823–831, Aug. 1985.
[21] H. L. Van Trees, *Detection, Estimation, and Modulation Theory*, part I. New York: Wiley, 1968.

Yingbo Hua (S'86–M'88–SM'92) was born in Wu-Xi, Jiangsu, China, on November 26, 1960. He received the B.S. degree in control engineering from Southeast University (Nanjing Institute of Technology), Nanjing, Jiangsu, China, in 1982, and the M.S. and Ph.D. degrees in electrical engineering from Syracuse University, Syracuse, NY, in 1983 and 1988, respectively.

He was a Graduate Teaching Assistant from 1984 to 1985, a University Graduate Fellow from 1985 to 1986, a Graduate Research Assistant from 1986 to 1988, and a Postdoctoral Research Associate from 1988 to 1989, all at Syracuse University. Since 1990, he has been a Lecturer with the University of Melbourne, Victoria, Australia. He has contributed over 40 publications dealing with spectral analysis and array processing.