Unitary PUMA Algorithm for Estimating the Frequency of a Complex Sinusoid

Cheng Qian, *Student Member, IEEE*, Lei Huang, *Senior Member, IEEE*, Hing Cheung So, *Fellow, IEEE*, Nicholas D. Sidiropoulos, *Fellow, IEEE*, and Junhao Xie, *Senior Member, IEEE*

Abstract—One-dimensional (1-D) and two-dimensional (2-D) frequency estimation for a single complex sinusoid in white Gaussian noise is a classic signal processing problem with numerous applications. It is revisited here through a new unitary principal-singular-vector utilization modal analysis (PUMA) approach, which is realized in terms of real-valued computations. The 2-D unitary PUMA is first formulated as an iteratively weighted least squares optimization problem. Recognizing that only one iteration is sufficient when 2-D unitary PUMA is initialized using least squares, a computationally attractive closed-form solution is then obtained. A variant of 2-D unitary PUMA is also developed for the 1-D case. Due to the real-valued computations and closed-form expression for the frequency estimate, the unitary PUMA is more computationally efficient than a number of state-of-the-art methods. Furthermore, the asymptotic variances of 1-D and 2-D unitary PUMA estimators are theoretically derived, and numerical results are included to demonstrate the effectiveness of the proposed methods.

Index Terms—Complex sinusoid, frequency estimation, subspace method, weighted least squares.

I. INTRODUCTION

FREQUENCY estimation for a single complex sinusoid in additive white Gaussian noise is a fundamental and well-studied problem in science and engineering. In many applications, such as wireless communications [1], synthetic aperture radar imaging [2], [3], source localization in radar and sonar systems [4]–[6], and software radio receivers [7], fast and accurate frequency estimation is crucial for proper system operation.

Numerous methods have been developed for single-tone frequency estimation, attaining different trade-offs between estimation accuracy and computational complexity. Among them,

maximum likelihood (ML) methods [8]–[10] are best (at least in theory) from an accuracy point of view, since their mean square error (MSE) performance attains the Cramér-Rao bound (CRB) under certain conditions, i.e., for a wide signal-to-noise ratio (SNR) range. The main drawback of the ML approach is its high computational complexity, as it entails solving a nonlinear and nonconvex optimization problem. To overcome this issue, suboptimal algorithms such as the iterative quadratic ML (IQML) [11], linear prediction (LP) [12], [13] and Pisarenko harmonic decomposer (PHD) based methods [14]–[17] were proposed to reduce the complexity. Since the IQML requires an iterative optimization procedure while the LP based methods usually yield closed-form formulas for frequency estimation, the former is much more computationally intensive than the latter. The PHD algorithm for single-tone frequency estimation [14] can be derived in a simpler manner using the sample autocorrelation at lags 1 and 2 [18]. However, the PHD is statistically inefficient, as its MSE departs from the CRB especially for large sample sizes [19]. Improved PHD based estimators [15]–[17] have been proposed to alleviate this inefficiency, but these require additional computations. In [20], a Markov estimator (ME) that utilizes a sample correlation sequence with length J to estimate the frequency has been proposed. However, it relies on the assumption of known SNR, which is often not available in practice. Furthermore, a large J is needed for high estimation accuracy, but this also implies higher computational complexity.

Recently, an accurate and low-complexity algorithm called principal-singular-vector utilization for modal analysis (PUMA) has been proposed for one-dimensional (1-D) and two-dimensional (2-D) frequency estimation of a single-tone [21], [22]. The main idea of the PUMA algorithm is to use the rank-1 property of the noisy data matrix, and then employ the LP and weighted least squares (WLS) techniques to perform parameter estimation. For 1-D frequency estimation, the PUMA scheme first rearranges the data samples into a suitable $M \times N$ matrix, and then uses a similar procedure as the 2-D PUMA method [22] to find the frequency. However, the PUMA algorithm estimates the signal parameters in an iterative manner. Furthermore, it is realized in terms of complex-valued operations. Iterations and complex-valued processing increase the computational complexity, especially for hardware implementation. To alleviate the computational load, we propose a version referred to as unitary PUMA, which is more computationally attractive since only real-valued calculations are involved and only one iteration is required. As a bonus, unitary PUMA improves the threshold performance of frequency estimation, as we will see.

The rest of this paper is organized as follows. We introduce the signal model for 2-D complex sinusoidal frequency estimation in Section II, and derive the new 2-D unitary PUMA algorithm in Section III. The main idea is to exploit the centro-symmetric property of the data model. We first construct a centro-Hermitian data matrix and then utilize a unitary transformation to convert the complex data matrix into real-valued form. Thus, all the calculations including the singular value decomposition (SVD) of the 2-D unitary PUMA scheme only involve real-valued operations. As a result, its computational burden can be reduced. We then employ the WLS technique to estimate the frequencies. In Section IV, we analyze the theoretical performance of the 2-D unitary PUMA method and derive its MSE expressions. In Section V, we explain how the 2-D unitary PUMA method can be exploited for 1-D frequency estimation. The resulting 1-D unitary PUMA algorithm is also accompanied by theoretical performance analysis. In Section VI, we present simulation results to assess the performance of 2-D and 1-D unitary PUMA, and validate our theoretical performance analysis. We summarize our findings and conclusions in Section VII.

Notation

Throughout the paper, we use boldface lowercase letters for vectors and boldface uppercase letters for matrices. Superscripts $(\cdot)^T$, $(\cdot)^*$, $(\cdot)^H$, $(\cdot)^{-1}$ and $(\cdot)^\dagger$ represent transpose, complex conjugate, Hermitian transpose, matrix inverse and pseudo inverse, respectively. The $\hat{a}$ denotes an estimate of a, $\mathbb{E}\{\cdot\}$ is the expectation operator and $\mathrm{diag}\{\cdot\}$ is the diagonal matrix operator. The $[\mathbf{A}]_{m,n}$ is the (m,n) entry of $\mathbf{A}$, $\mathbf{0}_m$ is the $m \times 1$ zero vector, $\mathbf{1}_m$ is the $m \times 1$ vector of all ones, $\mathbf{I}_m$ is the $m \times m$ identity matrix, and $\mathbf{\Pi}_m$ is the $m \times m$ exchange matrix with ones on its anti-diagonal and zeros elsewhere.

II. PROBLEM FORMULATION

A. Data Model

The observed 2-D data model is

$$\mathbf{R} = \mathbf{S} + \mathbf{Q} \tag{1}$$

where $\mathbf{Q} \in \mathbb{C}^{M \times N}$ denotes the noise matrix and $\mathbf{S} \in \mathbb{C}^{M \times N}$ denotes the signal component with elements:

$$[\mathbf{S}]_{m,n} = \kappa e^{j(\mu m + \nu n)}. \tag{2}$$

We assume that $[\mathbf{Q}]_{m,n}$ is white Gaussian noise with mean zero and variance σ^2, κ is the complex amplitude, and $\mu \in (-\pi, \pi)$ and $\nu \in (-\pi, \pi)$ are the frequency parameters. Varying m from 1 to M and n from 1 to N yields the matrix form of $\mathbf{S}$ as

$$\mathbf{S} = \kappa \mathbf{s}_\mu \mathbf{s}_\nu^T \tag{3}$$

where

$$\mathbf{s}_\mu = [\, e^{j\mu} \quad \cdots \quad e^{jM\mu} \,]^T \tag{4}$$

$$\mathbf{s}_\nu = [\, e^{j\nu} \quad \cdots \quad e^{jN\nu} \,]^T. \tag{5}$$

Let the SVD of $\mathbf{S}$ be

$$\mathbf{S} = \tilde{\mathbf{U}}_\mu \tilde{\mathbf{\Lambda}} \tilde{\mathbf{U}}_\nu^H \tag{6}$$

where $\tilde{\mathbf{\Lambda}} = \mathrm{diag}\{\tilde{\lambda}_1 \cdots \tilde{\lambda}_N\}$ with $\tilde{\lambda}_1 > \tilde{\lambda}_2 = \cdots = \tilde{\lambda}_N = 0$, $\tilde{\mathbf{U}}_\mu = [\tilde{\mathbf{u}}_{\mu,1} \cdots \tilde{\mathbf{u}}_{\mu,M}]$ and $\tilde{\mathbf{U}}_\nu = [\tilde{\mathbf{u}}_{\nu,1} \cdots \tilde{\mathbf{u}}_{\nu,N}]$. Due to the rank-1 property of $\mathbf{S}$, it can be expressed using its SVD as

$$\mathbf{S} = \tilde{\lambda}_1 \tilde{\mathbf{u}}_{\mu,1} \tilde{\mathbf{u}}_{\nu,1}^H \tag{7}$$

and approximated by taking the SVD of $\mathbf{R} = \mathbf{U}_\mu \mathbf{\Lambda} \mathbf{U}_\nu$ as

$$\mathbf{S} \approx \lambda_1 \mathbf{u}_{\mu,1} \mathbf{u}_{\nu,1}^H \tag{8}$$

where λ_1 is the largest singular value of $\mathbf{R}$, and $\mathbf{u}_{\mu,1}$ and $\mathbf{u}_{\nu,1}$ are the corresponding left and right singular vectors, respectively. Note that, in the noiseless case, it is shown in [22] that $\tilde{\lambda}_1 = \sqrt{MN}|\kappa|$, $\tilde{\mathbf{u}}_{\mu,1} = \mathbf{s}_\mu e^{j\psi_\mu}/\sqrt{M}$ and $\tilde{\mathbf{u}}_{\nu,1} = \mathbf{s}_\nu e^{j\psi_\nu}/\sqrt{N}$ where ψ_μ and ψ_ν are unknown parameters.

B. Unitary Transformation

Let us introduce the definition of a centro-Hermitian matrix [27]:

Definition 1: An $M \times N$ complex matrix $\mathbf{F}$ is called centro-Hermitian if $\mathbf{F} = \mathbf{\Pi}_M \mathbf{F}^* \mathbf{\Pi}_N$.

It is shown in [27]–[30] that for an arbitrary matrix $\mathbf{G} \in \mathbb{C}^{M \times N}$, $[\mathbf{G} \quad \mathbf{\Pi}_M \mathbf{G}^* \mathbf{\Pi}_N]$ is always centro-Hermitian. Thus, we can transform the complex matrix $\mathbf{G}$ into its real-valued counterpart by using

$$\mathcal{T}(\mathbf{G}) = \mathbf{T}_M^H [\mathbf{G} \quad \mathbf{\Pi}_M \mathbf{G}^* \mathbf{\Pi}_N] \mathbf{T}_{2N} \tag{9}$$

where $\mathcal{T}(\mathbf{G}) \in \mathbb{R}^{M \times 2N}$, $\mathcal{T}(\cdot)$ denotes the unitary transformation, and

$$\mathbf{T}_M = \begin{cases} \dfrac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_m & j\mathbf{I}_m \\ \mathbf{\Pi}_m & -j\mathbf{\Pi}_m \end{bmatrix}, & M = 2m \\[4mm] \dfrac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_m & \mathbf{0}_m & j\mathbf{I}_m \\ \mathbf{0}_m^T & \sqrt{2} & \mathbf{0}_m^T \\ \mathbf{\Pi}_m & \mathbf{0}_m & -j\mathbf{\Pi}_m \end{bmatrix}, & M = 2m+1. \end{cases} \tag{10}$$

III. 2-D UNITARY PUMA

The conventional 2-D PUMA algorithm [22] exploits the WLS to estimate μ using an iterative procedure. This typically requires at least three iterations to converge, and all calculations are complex-valued. As a result, the 2-D PUMA algorithm can still be computationally intensive, especially when M and N are large. To overcome these shortcomings, we propose a 2-D unitary variant which is realized in terms of real-valued operations. Moreover, by properly choosing an initial value, the suggested approach is able to avoid the iterative optimization and provide a closed-form solution.

Let $\mathbf{C}_\mu = \mathcal{T}(\mathbf{R})$ and decompose $\mathbf{C}_\mu$ using SVD as

$$\mathbf{C}_\mu = \mathbf{E}_\mu \mathbf{\Gamma}_\mu \mathbf{F}_\mu^T \tag{11}$$

where $\mathbf{C}_\mu \in \mathbb{R}^{M \times 2N}$, $\mathbf{\Gamma}_\mu$ contains the singular values and

$$\mathbf{E}_\mu = [\mathbf{e}_{\mu,1} \cdots \mathbf{e}_{\mu,M}] \tag{12}$$

$$\mathbf{F}_\mu = [\mathbf{f}_{\mu,1} \cdots \mathbf{f}_{\mu,2N}]. \tag{13}$$

Define two selection matrices as

$$\mathbf{J}_{\mu,1} = [\mathbf{I}_{M-1} \quad \mathbf{0}_{M-1}] \tag{14}$$

$$\mathbf{J}_{\mu,2} = \begin{bmatrix} \mathbf{0}_{M-1} & \mathbf{I}_{M-1} \end{bmatrix}. \tag{15}$$

Employing the shift invariance property and assuming that the noise perturbation is small relative to the signal power, we have

$$\mathbf{J}_{\mu,2}\mathbf{u}_{\mu,1} \approx e^{j\mu}\mathbf{J}_{\mu,1}\mathbf{u}_{\mu,1}. \tag{16}$$

It is easy to verify that

$$\mathbf{T}_{M-1}^{H}\mathbf{J}_{\mu,2}\mathbf{T}_{M} = \left(\mathbf{T}_{M-1}^{H}\mathbf{J}_{\mu,1}\mathbf{T}_{M}\right)^{*}. \tag{17}$$

Let $\mathbf{K}_{\mu,1}$ and $\mathbf{K}_{\mu,2}$ be the real and imaginary parts of $\mathbf{T}_{M-1}^{H}\mathbf{J}_{\mu,2}\mathbf{T}_{M}$, respectively. Then, (16) can be rewritten as

$$(\mathbf{K}_{\mu,1} + j\mathbf{K}_{\mu,2})\mathbf{e}_{\mu,1} \approx e^{j\mu}(\mathbf{K}_{\mu,1} - j\mathbf{K}_{\mu,2})\mathbf{e}_{\mu,1}. \tag{18}$$

After some straightforward manipulations, we obtain

$$\mathbf{x}_{\mu,2} \approx a_{\mu}\mathbf{x}_{\mu,1} \tag{19}$$

where $a_{\mu} = \tan(\mu/2)$, $\mathbf{x}_{\mu,1} = \mathbf{K}_{\mu,1}\mathbf{e}_{\mu,1}$ and $\mathbf{x}_{\mu,2} = \mathbf{K}_{\mu,2}\mathbf{e}_{\mu,1}$.

Adopting the WLS approach, (19) leads to

$$\min_{a_{\mu}} (\mathbf{x}_{\mu,2} - a_{\mu}\mathbf{x}_{\mu,1})^{T}\mathbf{W}^{-1}(\mu)(\mathbf{x}_{\mu,2} - a_{\mu}\mathbf{x}_{\mu,1}) \tag{20}$$

where $\mathbf{W}$ is the optimal weighting matrix that is defined through the Gauss-Markov theorem [20], [22], [28], [29]:

$$\begin{aligned}\mathbf{W}(\mu) &= \mathbb{E}\left\{(\mathbf{x}_{\mu,2} - a_{\mu}\mathbf{x}_{\mu,1})(\mathbf{x}_{\mu,2} - a_{\mu}\mathbf{x}_{\mu,1})^{T}\right\} \\ &= \mathbf{A}(\mu)\mathbb{E}\{\mathbf{e}_{\mu,1}\mathbf{e}_{\mu,1}^{T}\}\mathbf{A}^{T}(\mu) \end{aligned} \tag{21}$$

with

$$\mathbf{A}(\mu) = \mathbf{K}_{\mu,2} - a_{\mu}\mathbf{K}_{\mu,1}. \tag{22}$$

The solution to this unconstrained optimization problem is

$$\hat{a}_{\mu} = \frac{\mathbf{x}_{\mu,1}^{T}\mathbf{W}^{-1}(\mu)\mathbf{x}_{\mu,2}}{\mathbf{x}_{\mu,1}^{T}\mathbf{W}^{-1}(\mu)\mathbf{x}_{\mu,1}}. \tag{23}$$

Substituting $\mathbf{e}_{\mu,1} = \tilde{\mathbf{e}}_{\mu,1} + \Delta\mathbf{e}_{\mu,1}$ into (21) yields

$$\mathbf{W}(\mu) = \mathbf{A}(\mu)\mathbb{E}\{\Delta\mathbf{e}_{\mu,1}\Delta\mathbf{e}_{\mu,1}^{T}\}\mathbf{A}^{T}(\mu). \tag{24}$$

It is shown in [22], [31], [32] that[1]

$$\mathbb{E}\{\Delta\mathbf{e}_{\mu,1}\Delta\mathbf{e}_{\mu,1}^{T}\} = \frac{\sigma^{2}}{\tilde{\gamma}_{\mu,1}^{2}}\left(\mathbf{I}_{M} - \tilde{\mathbf{e}}_{\mu,1}\tilde{\mathbf{e}}_{\mu,1}^{T}\right) \tag{25}$$

where $\tilde{\gamma}_{\mu,1}^{2}$ is the largest singular value of $\mathcal{T}(\mathbf{S})$. Substituting (25) into (24), using $\mathbf{A}(\mu)\tilde{\mathbf{e}}_{\mu,1} = \mathbf{0}_{M-1}$ and ignoring the constant term $\sigma^{2}/\tilde{\gamma}_{\mu,1}^{2}$, we have

$$\mathbf{W}(\mu) = \mathbf{A}(\mu)\mathbf{A}^{T}(\mu). \tag{26}$$

[1]Equation (25) is directly computed from Lemma 1 of [32], albeit it looks a bit different from that in [32] because $\tilde{\gamma}_{\mu,1}$ is the singular value of $\mathcal{T}(\mathbf{S})$ which is actually a forward-backward averaging matrix. It is easy to follow [32] to verify that the singular values of $\mathcal{T}(\mathbf{S})$ are the same as those of $[\mathbf{S} \quad \mathbf{\Pi}_{M}\mathbf{S}\mathbf{\Pi}_{N}]$, whereas the singular values of $[\mathbf{S} \quad \mathbf{\Pi}_{M}\mathbf{S}\mathbf{\Pi}_{N}]$ are $\sqrt{2}$ times the eigenvalues in $\mathbf{S}$. Using these tips, the reader can obtain (25).

TABLE I
2-D UNITARY PUMA ALGORITHM.

Step 1	Use the unitary transformation to obtain $\mathbf{C}_{\mu}$ and $\mathbf{C}_{\nu}$, and perform their SVD.
Step 2	Employ (27), (29) and (28) to calculate $\hat{a}_{\mu,\text{init}}$, $\hat{\mathbf{A}}(\mu)$ and $\hat{\mathbf{W}}(\mu)$, respectively.
Step 3	Estimate $\hat{\mu}$ using (30).
Step 4	Calculate $\hat{a}_{\nu,\text{init}} = \mathbf{x}_{\nu,1}^{T}\mathbf{x}_{\nu,2}/(\mathbf{x}_{\nu,1}^{T}\mathbf{x}_{\nu,1})$ and then substitute it into (37) and (38) to obtain the initial estimates of $\hat{\mathbf{W}}(\nu)$ and $\hat{\mathbf{A}}(\nu)$, respectively.
Step 5	Estimate $\hat{\nu}$ via (34).

Since a_{μ} is unknown, we can perform the estimation in a relaxation manner. It follows from (19) that we can use the least squares (LS) estimate of a_{μ} as an initial estimate, i.e.,

$$\hat{a}_{\mu,\text{init}} = \mathbf{x}_{\mu,1}^{\dagger}\mathbf{x}_{\mu,2} \tag{27}$$

where $\hat{a}_{\mu,\text{init}} \rightarrow a_{\mu}$ for sufficiently high SNR (See Appendix A). Substituting (27) into (26) leads to

$$\hat{\mathbf{W}}(\mu) = \hat{\mathbf{A}}(\mu)\hat{\mathbf{A}}^{T}(\mu) \tag{28}$$

where

$$\hat{\mathbf{A}}(\mu) = \mathbf{K}_{\mu,2} - \hat{a}_{\mu,\text{init}}\mathbf{K}_{\mu,1}. \tag{29}$$

Recalling that $a_{\mu} = \tan(\mu/2)$, the estimate of μ is then computed as

$$\hat{\mu} = 2\tan^{-1}\left(\frac{\mathbf{x}_{\mu,1}^{T}\hat{\mathbf{W}}^{-1}(\mu)\mathbf{x}_{\mu,2}}{\mathbf{x}_{\mu,1}^{T}\hat{\mathbf{W}}^{-1}(\mu)\mathbf{x}_{\mu,1}}\right). \tag{30}$$

The next step is to estimate ν. Let $\mathbf{C}_{\nu} = \mathcal{T}(\mathbf{R}^{T})$ and its SVD be

$$\mathbf{C}_{\nu} = \mathbf{E}_{\nu}\mathbf{\Gamma}_{\nu}\mathbf{F}_{\nu}^{T} \in \mathbb{R}^{N \times 2M} \tag{31}$$

where $\mathbf{\Gamma}_{\mu}$ holds the singular values and

$$\mathbf{E}_{\nu} = [\mathbf{e}_{\nu,1} \cdots \mathbf{e}_{\nu,N}] \tag{32}$$

$$\mathbf{F}_{\nu} = [\mathbf{f}_{\nu,1} \cdots \mathbf{f}_{\nu,2M}]. \tag{33}$$

Applying the same idea for estimating μ and according to (20)–(30), it is easy to obtain the closed-form expression of $\hat{\nu}$:

$$\hat{\nu} = 2\tan^{-1}\left(\frac{\mathbf{x}_{\nu,1}^{T}\hat{\mathbf{W}}^{-1}(\nu)\mathbf{x}_{\nu,2}}{\mathbf{x}_{\nu,1}^{T}\hat{\mathbf{W}}^{-1}(\nu)\mathbf{x}_{\nu,1}}\right) \tag{34}$$

where

$$\mathbf{x}_{\nu,1} = \mathbf{K}_{\nu,1}\mathbf{e}_{\nu,1} \tag{35}$$

$$\mathbf{x}_{\nu,2} = \mathbf{K}_{\nu,2}\mathbf{e}_{\nu,2} \tag{36}$$

$$\hat{\mathbf{W}}(\nu) = \hat{\mathbf{A}}(\nu)\hat{\mathbf{A}}^{T}(\nu) \tag{37}$$

$$\hat{\mathbf{A}}(\nu) = \mathbf{K}_{\nu,1}\hat{a}_{\nu,\text{init}} - \mathbf{K}_{\nu,2}. \tag{38}$$

Here, $\hat{a}_{\nu,\text{init}} = \mathbf{x}_{\nu,1}^{\dagger}\mathbf{x}_{\nu,2}$ where $\mathbf{x}_{\nu,1} = \mathbf{K}_{\nu,1}\mathbf{e}_{\nu,1}$ and $\mathbf{x}_{\nu,2} = \mathbf{K}_{\nu,2}\mathbf{e}_{\nu,2}$, $\mathbf{K}_{\nu,1}$ and $\mathbf{K}_{\nu,2}$ are the real and imaginary parts of $\mathbf{T}_{N-1}^{H}\mathbf{J}_{\nu,2}\mathbf{T}_{N}$, respectively, and $\mathbf{J}_{\nu,2} = [\mathbf{0}_{N-1} \quad \mathbf{I}_{N-1}]$. The steps for the 2-D unitary PUMA algorithm are summarized in Table I.

Remark 1: Note that following [22], $\hat{a}_\mu$ can be refined via an iterative step, i.e.,

  i) *Initialize* $\mathbf{A}(\mu)$ *using* $\hat{a}_{\mu,\text{init}}$ *in* (27);
  ii) *Construct* $\mathbf{W}(\mu)$ *using* (26);
  iii) *Calculate* $\hat{a}_\mu$ *via* (23);
  iv) *Compute* $\mathbf{A}(\mu)$ *by substituting* $\hat{a}_\mu$ *into* (22);
  v) *Repeat ii) to iv) until a stopping criterion is reached.*

However, simulation results in Section VI demonstrate that when compared to the unitary PUMA scheme with one iteration, there is almost no performance improvement for the unitary PUMA with additional iterations. Moreover, we have shown in Appendix A that $\hat{a}_{\mu,\text{init}}$ converges to the true a_μ with SNR approaching infinity, which in turn indicates that the initial estimate of a_μ is sufficiently accurate to reach the point of diminishing returns in the WLS estimate of a_μ in one iteration. As a result, only one iteration is enough for the unitary PUMA to achieve a significant performance improvement in terms of estimation accuracy, such that the closed-form expressions in (30) and (34) can be used for frequency estimation.

Remark 2: The implementation of the 2-D unitary PUMA algorithm requires two major steps:

  i) *SVD of* $\mathcal{T}(\mathbf{R})$ *and* $\mathcal{T}(\mathbf{R}^T)$;
  ii) *Calculation of* $\hat{a}_\mu$ *and* $\hat{a}_\nu$.

The total flops to calculate the SVD of $\mathcal{T}(\mathbf{R})$ and $\mathcal{T}(\mathbf{R}^T)$ are approximately $\mathcal{O}(4MN^2 + 4NM^2)$ where a flop is defined as a real floating-point multiplication. The computations of $\mathbf{W}^{-1}(\mu)$ and $\mathbf{W}^{-1}(\nu)$ take about $\mathcal{O}(M^3 + N^3)$ flops. Therefore, the 2-D unitary PUMA scheme requires about $\mathcal{O}(4(MN^2 + NM^2) + M^3 + N^3)$ where the computations needed in the remaining steps are negligible[2].

IV. PERFORMANCE ANALYSIS

In this section, we study the statistical properties of the proposed 2-D unitary PUMA method for large SNR regime. First, it is not difficult to obtain the following result on the consistency of the 2-D unitary PUMA estimates.

Lemma 1: As SNR tends to infinity, the frequency estimate $\hat{\mu}$ obtained by (30) approaches the true parameter w.p. 1.

  Proof: This lemma can be proved in a similar manner as [25] and [26]. The cost function in (20) achieves the minimum if and only if μ is the true parameter, and thus $\hat{\mu}$ approaches the true value of μ w.p. 1 as SNR $\to \infty$. ∎

The asymptotic variance of the proposed method is given in the following proposition.

Proposition 1: The asymptotic variance of $\hat{\mu}$ as SNR $\to \infty$ is given by

$$\text{var}(\hat{\mu}) = \frac{4\cos^4(\mu/2)\sigma^2}{\tilde{\gamma}_{\mu,1}^2 \left(\tilde{\mathbf{x}}_{\mu,1}^T \mathbf{W}^{-1}(\mu) \tilde{\mathbf{x}}_{\mu,1} \right)} \tag{39}$$

and the asymptotic variance of $\hat{\nu}$ is likewise given by

$$\text{var}(\hat{\nu}) = \frac{4\cos^4(\nu/2)\sigma^2}{\tilde{\gamma}_{\nu,1}^2 \left(\tilde{\mathbf{x}}_{\nu,1}^T \mathbf{W}^{-1}(\nu) \tilde{\mathbf{x}}_{\nu,1} \right)}. \tag{40}$$

  Proof: See Appendix B. ∎

[2]Since the unitary PUMA algorithm only needs the principal eigenvectors of the data matrix in both 1-D and 2-D cases, its computational complexity can be further reduced if we employ the power method [35] to calculate the principal eigenvectors. That is, the complexity order is $\mathcal{O}(M^2 + N^2 + M^3 + N^3)$.

It is often interesting to compare the variance of an estimator with the CRB. It is shown in Appendix C that (39) and (40) are equivalent to the following simplified expressions

$$\text{var}(\hat{\mu}) = \frac{6\sigma^2}{(M^2 - 1)MN|\kappa|^2} \tag{41}$$

$$\text{var}(\hat{\nu}) = \frac{6\sigma^2}{(N^2 - 1)MN|\kappa|^2} \tag{42}$$

which are identical to the CRB [11], [22].

V. REFORMULATION OF 2-D UNITARY PUMA FOR 1-D CASE

A. Data Model

The 1-D signal model is

$$r_l = s_l + q_l, \quad l = 1, \dots, L \tag{43}$$

where

$$s_l = \bar{\kappa} e^{j\omega l}, \tag{44}$$

with $\bar{\kappa}$ being the complex amplitude and ω being the frequency, q_l is the additive white Gaussian noise with mean zero and variance σ^2, and L is the number of samples.

B. 1-D Unitary PUMA Algorithm

Let us choose the first MN ($MN \leq L$) samples from the data sequence and arrange them into an $M \times N$ data matrix:

$$\mathbf{R}_\omega = \begin{bmatrix} r_1 & r_{M+1} & \cdots & r_{M(N-1)+1} \\ r_2 & r_{M+2} & \cdots & r_{M(N-1)+2} \\ \vdots & \vdots & & \vdots \\ r_M & r_{2M} & \cdots & r_{MN} \end{bmatrix}$$
$$= \mathbf{S}_\omega + \mathbf{Q}_\omega \in \mathbb{C}^{M \times N} \tag{45}$$

where $\mathbf{S}_\omega$ and $\mathbf{Q}_\omega$ share the same structure as $\mathbf{R}_\omega$ and are constructed by s_l and q_l, respectively. Indeed, $\mathbf{S}_\omega$ can be expressed as the product of two Vandermonde vectors, i.e.,

$$\mathbf{S}_\omega = \bar{\kappa} \mathbf{s}_\omega \mathbf{s}_\phi^T \tag{46}$$

with $\phi = M\omega$ and

$$\mathbf{s}_\omega = \begin{bmatrix} e^{j\omega} & e^{j2\omega} & \cdots & e^{jM\omega} \end{bmatrix}^T \tag{47}$$

$$\mathbf{s}_\phi = \begin{bmatrix} 1 & e^{j\phi} & \cdots & e^{j(N-1)\phi} \end{bmatrix}^T. \tag{48}$$

From shift invariance, we have

$$\mathbf{J}_{\omega,2} \mathbf{s}_\omega = e^{j\omega} \mathbf{J}_{\omega,1} \mathbf{s}_\omega \tag{49}$$

where $\mathbf{J}_{\omega,1} = \begin{bmatrix} \mathbf{I}_{M-1} & \mathbf{0}_{M-1} \end{bmatrix}$ and $\mathbf{J}_{\omega,2} = \begin{bmatrix} \mathbf{0}_{M-1} & \mathbf{I}_{M-1} \end{bmatrix}$. From (46), $\mathbf{S}_\omega$ is a rank-1 matrix, in the high SNR regime, it can be approximated as

$$\mathbf{S}_\omega \approx \zeta_1 \mathbf{u}_{\omega,1} \mathbf{u}_{\phi,1}^H \tag{50}$$

where ζ_1 is the largest singular value of $\mathbf{R}_\omega$ while $\mathbf{u}_{\omega,1}$ and $\mathbf{u}_{\phi,1}$ are the corresponding left and right singular vectors, respectively. It follows from (49) that

$$\mathbf{J}_{\omega,2} \mathbf{u}_{\omega,1} \approx e^{j\omega} \mathbf{J}_{\omega,1} \mathbf{u}_{\omega,1}. \tag{51}$$

Therefore, we can employ **Step 1** to **Step 3** in Table I to determine an estimate $\hat{\omega}_L$ of ω (in place of μ). However, $\hat{\omega}_L$ is not necessarily a reliable estimate, especially in low SNR scenarios, since the information in the row space of $\mathbf{R}_\omega$, i.e., $\phi = M\omega$, has not been utilized. To address this issue, we first find ϕ from the row space of $\mathbf{R}_\omega$ and then compute another frequency estimate $\hat{\omega}_R$ from $\hat{\phi}$. Finally, combining $\hat{\omega}_L$ and $\hat{\omega}_R$ leads to the final estimate of ω.

To reduce the computational complexity and improve accuracy, we use the unitary transformation to generate a unitary matrix counterpart of $\mathbf{R}_\omega$, i.e., $\mathbf{C}_\phi = \mathcal{T}(\mathbf{R}_\omega^T)$. According to (20)–(29), by solving a similar WLS problem as (20) and utilizing the initialization in (27), we obtain the estimate of ϕ as

$$\hat{\phi} = 2\tan^{-1}\left(\frac{\mathbf{x}_{\phi,1}^T \hat{\mathbf{W}}^{-1}(\phi)\mathbf{x}_{\phi,2}}{\mathbf{x}_{\phi,1}^T \hat{\mathbf{W}}^{-1}(\phi)\mathbf{x}_{\phi,1}}\right) \qquad (52)$$

where $\mathbf{x}_{\phi,1} = \mathbf{K}_{\phi,1}\mathbf{e}_{\phi,1} \in \mathbb{R}^{N-1}$, $\mathbf{x}_{\phi,2} = \mathbf{K}_{\phi,2}\mathbf{e}_{\phi,1} \in \mathbb{R}^{N-1}$ with $\mathbf{e}_{\phi,1}$ being the eigenvector corresponding to the largest eigenvalue of $\mathbf{C}_\phi$, and $\hat{\mathbf{W}}(\phi) = \hat{\mathbf{A}}(\phi)\hat{\mathbf{A}}^T(\phi)$ has the same definition as $\hat{\mathbf{W}}(\nu)$. Moreover, $\mathbf{K}_{\nu,1}$ and $\mathbf{K}_{\nu,2}$ are the real and imaginary parts of $\mathbf{T}_{N-1}^H \mathbf{J}_{\phi,2}\mathbf{T}_N$, respectively, and

$$\hat{\mathbf{A}}(\phi) = \mathbf{K}_{\phi,1}\hat{a}_{\phi,\text{init}} - \mathbf{K}_{\phi,2}. \qquad (53)$$

After obtaining $\hat{\phi}$, we cannot simply estimate ω via $\hat{\phi}/M$, because $\phi + 2\pi i = M\omega$ where $i = -\lfloor M/2 \rfloor, \ldots, \lfloor M/2 \rfloor$. In other words, there are $(2\lfloor M/2 \rfloor + 1)$ possible candidates for $\hat{\omega}_R$. We first calculate these $(2\lfloor M/2 \rfloor + 1)$ candidates for $\hat{\omega}_R$ using

$$\hat{\omega}_{R,i} = \frac{\hat{\phi} + 2\pi i}{M}. \qquad (54)$$

Then, we compute the absolute error between $\hat{\omega}_{R,i}$ and $\hat{\omega}_L$ and choose $\hat{\omega}_R = \hat{\omega}_{R,k}$, where index k is obtained from [21]

$$k = \arg\min_{i \in} \{-\lfloor M/2 \rfloor, \ldots, \lfloor M/2 \rfloor\} |\hat{\omega}_{R,i} - \hat{\omega}_L|, \qquad (55)$$

and

$$\hat{\omega}_L = 2\tan^{-1}\left(\frac{\mathbf{x}_{\omega,1}^T \hat{\mathbf{W}}^{-1}(\omega)\mathbf{x}_{\omega,2}}{\mathbf{x}_{\omega,1}^T \hat{\mathbf{W}}^{-1}(\omega)\mathbf{x}_{\omega,1}}\right). \qquad (56)$$

The two estimates $\hat{\omega}_L$ and $\hat{\omega}_R$ are finally combined to form the final estimate $\hat{\omega}$, as explained next.

C. Performance Analysis of 1-D Unitary PUMA Algorithm

According to Appendix C, the asymptotic variances of $\hat{\omega}_L$ and $\hat{\omega}_R$ are

$$\text{var}(\hat{\omega}_L) \approx \frac{6\sigma^2}{(M^2-1)MN|\bar{\kappa}|^2} \qquad (57)$$

$$\text{var}(\hat{\omega}_R) \approx \frac{6\sigma^2}{(N^2-1)M^3N|\bar{\kappa}|^2}. \qquad (58)$$

Therefore, the final estimate of ω is calculated as [21]

$$\hat{\omega} = \frac{\frac{\hat{\omega}_L}{\text{var}(\hat{\omega}_L)} + \frac{\hat{\omega}_R}{\text{var}(\hat{\omega}_R)}}{\frac{1}{\text{var}(\hat{\omega}_L)} + \frac{1}{\text{var}(\hat{\omega}_R)}}$$

$$\approx \frac{(M^2-1)\hat{\omega}_L + M^2(N^2-1)\hat{\omega}_R}{M^2N^2-1}. \qquad (59)$$

The asymptotic variance of $\hat{\omega}$ is

$$\text{var}(\hat{\omega}) = \left(\frac{1}{\text{var}(\hat{\omega}_L)} + \frac{1}{\text{var}(\hat{\omega}_R)}\right)^{-1}$$

$$\approx \frac{6\sigma^2}{(M^2N^2-1)MN|\bar{\kappa}|^2}. \qquad (60)$$

D. Determining M and N

It worth noting that the CRB for ω is [8]

$$\text{CRB} = \frac{6\sigma^2}{(L^2-1)L|\bar{\kappa}|^2}. \qquad (61)$$

Comparing (60) and (61), we find that when $L = MN$, the asymptotic variance of $\hat{\omega}$ is the same as the CRB. However, if $L > MN$, (60) is larger than the CRB. Meanwhile, we should point out that the computational complexity of the 1-D unitary PUMA algorithm is $\mathcal{O}(4(MN^2 + M^2N) + M^3 + N^3)$ flops, so different combinations of M and N entail different flop counts. Thus, how to determine appropriate values for M and N becomes an interesting problem, as computational complexity inevitably comes into play in practical applications.

Let $N = \rho M$ where $\rho \geq 1$. The computational complexity of 1-D unitary PUMA scheme is almost the same as that of the 2-D unitary PUMA. Substituting $N = \rho M$ into $4(MN^2 + NM^2) + M^3 + N^3$ leads to
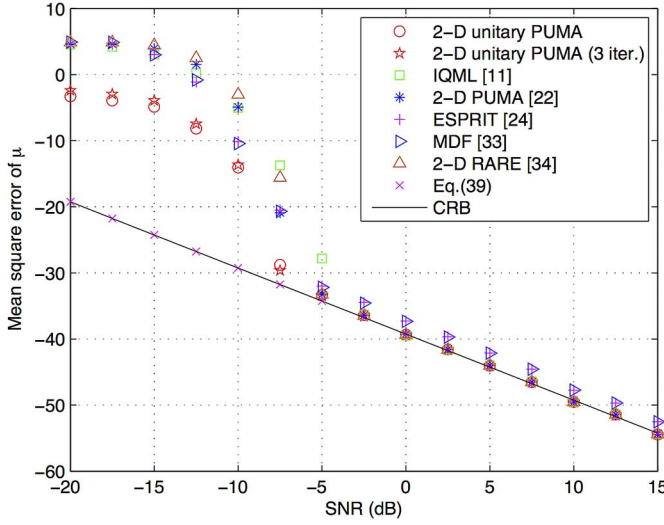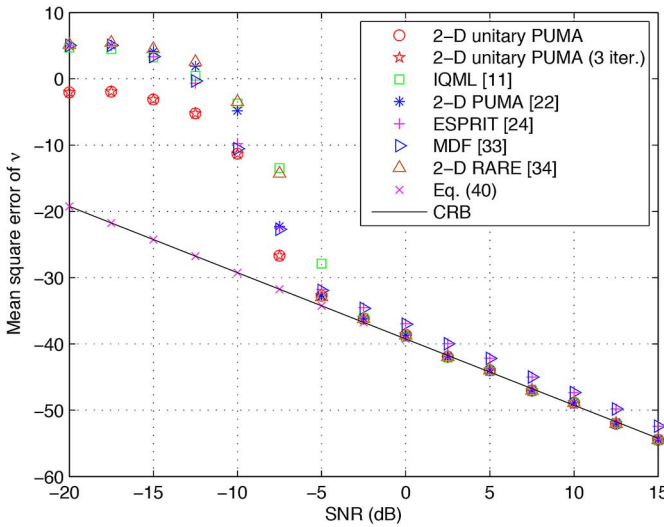
$$f(\rho) = (\rho^3 + 4\rho^2 + 4\rho + 1)M^3, \quad \rho \geq 1. \qquad (62)$$

It is not difficult to find that the minimum of $f(\rho)$ is achieved if and only if $\rho = 1$, which indicates that among all the combinations, $M = N = \sqrt{L}$ is the best choice to achieve optimal performance with the lowest computational burden. This choice is also helpful from an accuracy point of view, as will be verified in our simulations – note that whereas *asymptotic* variance (in the high SNR regime) only depends on the product of M and N and SNR, in the low SNR regime we want large M and large N for higher accuracy.

In some cases, M and N might not satisfy $L = MN$. If L is a prime number, then $M = 2$ and $N = (L-1)/2$ yields $MN = L - 1$ which maximizes the usage of samples. Although this combination has the smallest variance, it is not appropriate for real applications because it is the most computationally demanding one. From our simulations, we also find that $[M, N] = [2, (L-1)/2]$ has the worst threshold performance due to the fact that the degree of freedom for ω_L is only one. As a result, it is more appropriate to delete more samples to enable using $M \approx N$, so that a better trade-off between estimation accuracy and complexity can be attained.

VI. NUMERICAL RESULTS

Numerical experiments have been carried out to evaluate the performance of the unitary PUMA algorithm in the presence of additive white Gaussian noise. All results provided are averages over 500 independent runs using a computer with 2.6 GHz dual-core Intel i5 processor and 8 GB RAM.

Fig. 1. MSE of μ versus SNR. ($M = N = 15$, $\mu = -0.12\pi$ and $\nu = 0.2\pi$).

Fig. 3. MSE of μ versus SNR. ($M = N = 5$, $\mu = -0.12\pi$ and $\nu = 0.2\pi$).

Fig. 2. MSE of ν versus SNR. ($M = N = 15$, $\mu = -0.12\pi$ and $\nu = 0.2\pi$).

Fig. 4. MSE of ν versus SNR. ($M = N = 5$, $\mu = -0.12\pi$ and $\nu = 0.2\pi$).

A. 2-D Frequency Estimation

The 2-D unitary PUMA algorithm is compared with the IQML [11], 2-D PUMA [22], ESPRIT [24], multi-dimensional folding algorithm (MDF) [33], 2-D rank reduction estimator (2-D RARE) [34], and the CRB [11]. For the ESPRIT algorithm, we use a $MN/4 \times (N/2 + 1)(M/2 + 1)$ Hankel block-Hankel matrix constructed from the $M \times N$ data matrix for parameter estimation (the best possible setting for ESPRIT [24]). The theoretical asymptotic variance of the proposed estimator is also included to validate our performance analysis.

In the first example, we consider a complex tone where the signal parameters are $M = N = 15$, $\kappa = e^{j1}$, $\mu = -0.12\pi$ and $\nu = 0.2\pi$. To demonstrate the effectiveness of the unitary PUMA with closed-form expressions, we include a three iterations based unitary PUMA for comparison. The MSEs of μ and ν as a function of SNR are shown in Figs. 1 and 2, respectively. We observe that the proposed method is clearly advantageous compared to its counterparts. With increasing SNR, all MSEs of the 2-D unitary PUMA, 2-D PUMA, IQML
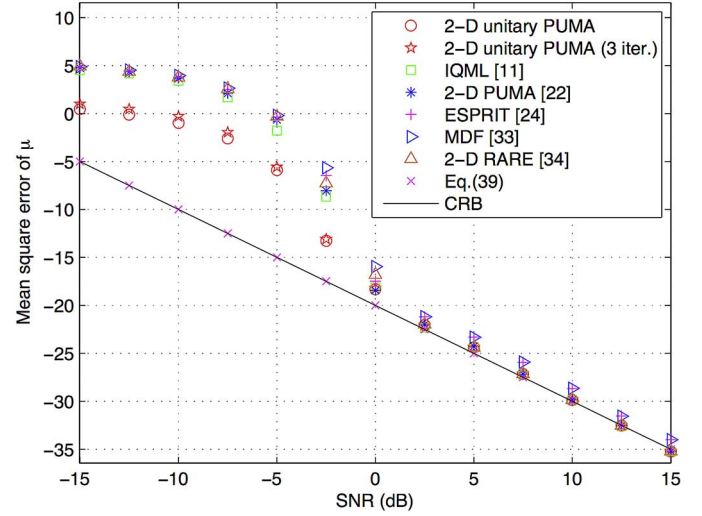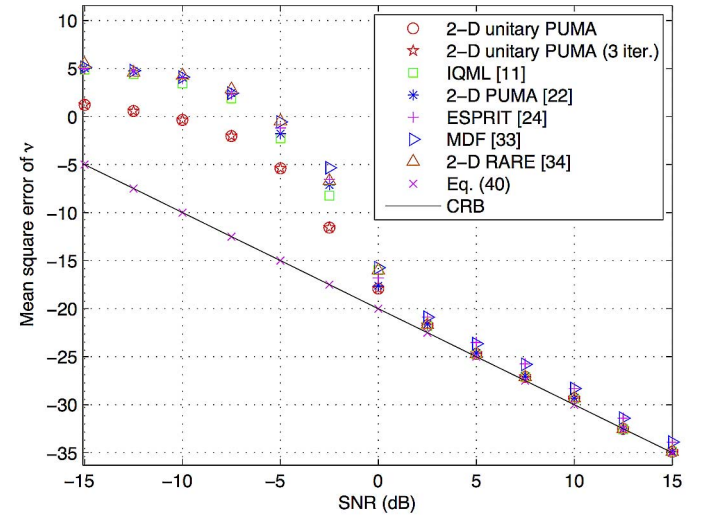
and 2-D RARE algorithms attain the CRB, while the ESPRIT and MDF methods (which are designed to handle multiple tones) are always suboptimal throughout the whole SNR range. This is mainly because both ESPRIT and MDF utilize the LS method to solve a rotational invariance equation (RIE). However, both sides of the RIE contain noise which is highly correlated, and this makes LS inappropriate. Furthermore, the computation times of 2-D unitary PUMA, 2-D unitary PUMA with 3 iterations, 2-D PUMA, ESPRIT, IQML, MDF and 2-D RARE algorithms are 6.57×10^{-4} s, 1.05×10^{-3} s, 2.14×10^{-3} s, 7.59×10^{-3} s, 2.52×10^{-1} s, 7.27×10^{-3} s and 7.45×10^{-3} s, respectively. Note that the unitary PUMA with three iterations has no performance improvement compared to the unitary PUMA with one iteration while the computational time of the former is about 1.6 times larger than the latter. Figs. 3 and 4 provide a comparison for small data size, i.e., $M = N = 5$, while the other parameters are the same as those in the first example. In this situation, it is seen that the proposed method has the highest estimation accuracy and outperforms the 2-D PUMA scheme when SNR < 0 dB. This is due to the
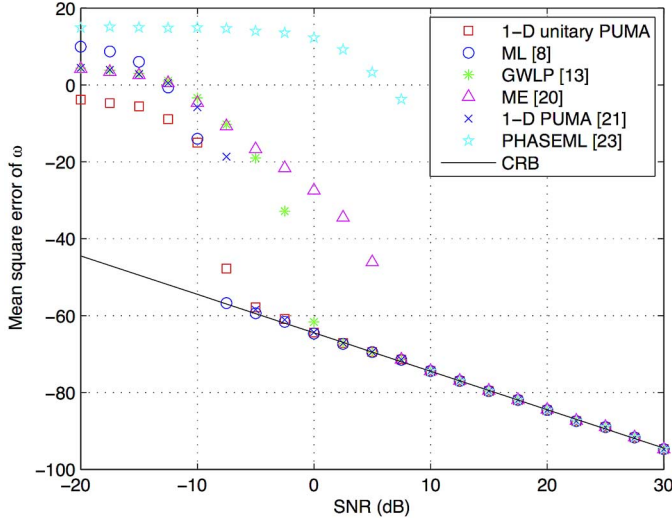
Fig. 5. MSE of ω versus SNR. ($L = 256$, $\omega = 0.1\pi$ and $\bar{\kappa} = e^{j1}$).

Fig. 6. MSE of ω versus SNR. ($L = 1024$, $\omega = 0.2\pi$ and $\bar{\kappa} = e^{j1}$).

forward-backward averaging technique used to construct $\mathbf{C}_\mu$ and $\mathbf{C}_\nu$. The theoretical MSEs of the proposed method align well with the CRB.

B. 1-D Frequency Estimation

We now evaluate the performance of the 1-D unitary PUMA algorithm. The MSEs of the ML [8], generalized weighted linear predictor (GWLP) [13], ME [20], 1-D PUMA [21] and phase-based ML (PHASEML) [23] schemes as well as the CRB [8] are included for comparison. Here, we consider a complex exponential with $\bar{\kappa} = e^{j1}$ and $\omega = 0.1\pi$. The number of samples is $L = 256$. We choose $M = N = 16$ for the 1-D PUMA and 1-D unitary PUMA algorithms. For the ME, we assume that the SNR is known and the sample sequence length is $J = 100$. It is observed in Fig. 5 that the proposed method has the smallest MSE in extremely low SNR regime, i.e., SNR < -10 dB. However, as SNR increases, the ML scheme outperforms the 1-D unitary PUMA and achieves the best threshold performance. On the other hand, the PHASEML method is suboptimal and it does not attain the CRB until SNR > 10 dB because errors can be accumulated in the phase unwrapping step if SNR is not large enough [23]. Besides comparing MSE performance, we note that the computation times of the 1-D unitary PUMA, ML, GWLP, ME, 1-D PUMA and PHASEML estimators are 5.96×10^{-4} s, 1.49×10^{-3} s, 9.29×10^{-3} s,, 1.18×10^{-2} s, 1.13×10^{-3} s and 7.53×10^{-3} s, respectively. It is obvious that the 1-D unitary PUMA scheme is the fastest, while 1-D PUMA and ML require twice the time or more in this case.

Next, let us examine the performance of the 1-D unitary PUMA algorithm for different combinations of M and N. We vary M from 2 to 32 such that the corresponding N is changed from 512 to 32. All combinations satisfy $L = MN = 1024$ and the other parameters are the same as those in Fig. 5, except that $\omega = 0.2\pi$. We observe from Fig. 6 that the threshold performance is significantly affected by the choice of M and N, but asymptotic performance is not, as expected from our analysis. The combination $[M, N] = [32, 32]$ yields the best threshold performance, while $[M, N] = [2, 512]$ and $[4, 256]$ result in
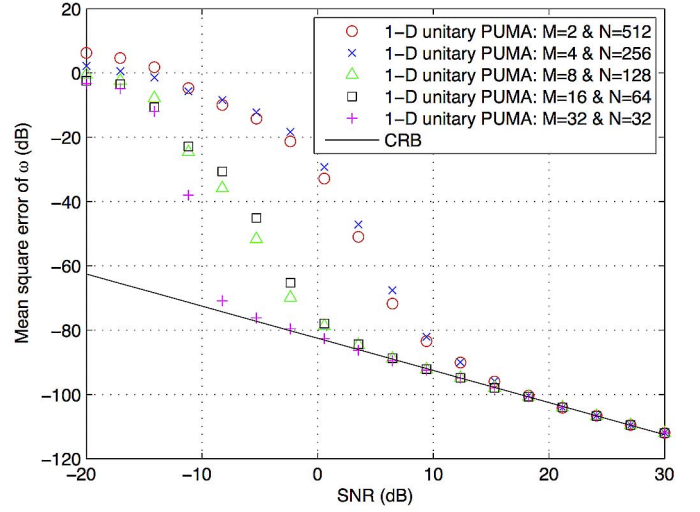
TABLE II
CPU TIME OF 1-D UNITARY PUMA FOR $L = 1024$.

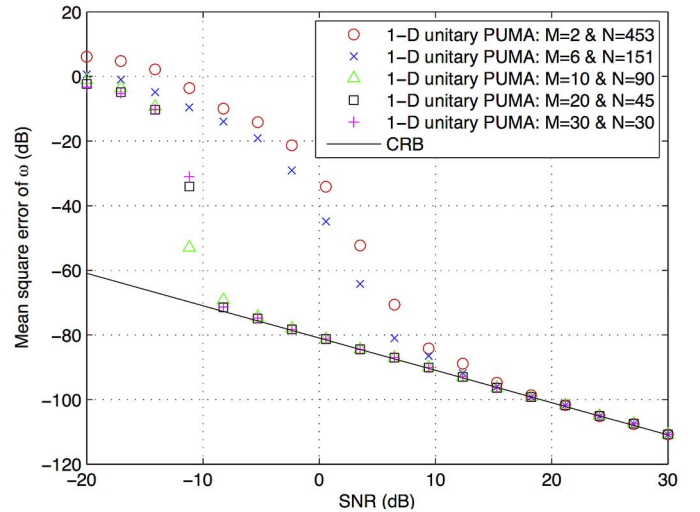| Parameter setting	CPU time (s)
$M = 2, N = 512$	0.4975
$M = 4, N = 256$	0.0563
$M = 8, N = 128$	0.0093
$M = 16, N = 64$	0.0021
$M = 32, N = 32$	0.0012

Fig. 7. MSE of ω versus SNR. ($L = 907$, $\omega = 0.2\pi$ and $\bar{\kappa} = e^{j1}$).

the worst performance. Together with the CPU times shown in Table II, we can observe that $[M, N] = [32, 32]$ corresponds to the smallest computation time and best threshold performance. This in turn verifies the correctness of our theoretical analysis in Section V-D.

Fig. 7 demonstrates the effectiveness of the 1-D PUMA method for some special cases where we cannot find M and N satisfying $L = MN$. In this test, we set $L = 907$ such that L is now a prime number and there does not exist $M = N = \sqrt{L}$. Therefore, we need to delete some samples and choose $M \approx N$. We consider four different settings of M and N for comparison,

TABLE III
CPU TIME OF 1-D UNITARY PUMA FOR $L = 907$.

Parameter setting	CPU time (s)
$M = 2, N = 453$	0.3082
$M = 6, N = 151$	0.0149
$M = 10, N = 90$	0.0040
$M = 20, N = 45$	0.0015
$M = 30, N = 30$	0.0011

i.e., $[M, N] = [2, 453], [6, 151], [10, 90], [20, 45]$ and $[30, 30]$. For $[M, N] = [2, 453]$ and $[6, 151]$, we only delete one sample while for the other three combinations, we delete seven samples. It is noticeable that the MSEs of all the combinations are larger than the CRB because they do not utilize the whole data. It is seen in Fig. 7 that the performance of $[M, N] = [2, 453]$ and $[6, 151]$ is inferior to the other three combinations when SNR < 10 dB. This is because the estimate of ω_L is not reliable in this case, which may lead us to pick a bad $\hat{\omega}_{R,i}$. We conclude that if the difference between M and N is not too large, the 1-D unitary PUMA estimator can provide better threshold performance. Among all the combinations, we conclude that $[M, N] = [30, 30]$ is the best choice since it has a good threshold performance but also the lowest complexity, as shown in Table III.

VII. CONCLUSION

Unitary versions of 2-D and 1-D PUMA frequency estimators have been devised here, along with asymptotic performance analysis leading to closed-form MSE expressions in the high SNR regime. Unitary PUMA techniques provide the same estimation accuracy as their conventional counterparts in the high SNR regime, but the threshold performance of unitary PUMA is significantly better. Furthermore, unitary PUMA entails lower complexity, requiring only real-valued computations. Computer simulations confirm our theoretical derivations, indicating that unitary PUMA approach outperforms many other single-tone frequency estimators in the 2-D case. In the 1-D case, ML frequency estimation features better threshold performance, but the 1-D unitary PUMA can be computationally simpler, and it appears to have an edge in the very low SNR regime – where there is no guarantee that ML will work well.

APPENDIX A
PROOF OF $\hat{a}_{\mu,\text{init}} \to a_\mu$

Expressing $\hat{a}_{\mu,\text{init}} = a_\mu + \Delta a_\mu$, $\mathbf{x}_{\mu,1} = \tilde{\mathbf{x}}_{\mu,1} + \Delta\mathbf{x}_{\mu,1}$ and $\mathbf{x}_{\mu,2} = \tilde{\mathbf{x}}_{\mu,2} + \Delta\mathbf{x}_{\mu,2}$, it follows from (19) that

$$(\tilde{\mathbf{x}}_{\mu,1} + \Delta\mathbf{x}_{\mu,1})(a_\mu + \Delta a_\mu) \approx \tilde{\mathbf{x}}_{\mu,2} + \Delta\mathbf{x}_{\mu,2}. \quad (63)$$

Since $\tilde{\mathbf{x}}_{\mu,1}\mathbf{a}_\mu = \tilde{\mathbf{x}}_{\mu,2}$, by neglecting $\Delta\mathbf{x}_{\mu,1}\Delta a_\mu$, (63) is simplified as

$$\Delta\mathbf{x}_{\mu,1}a_\mu + \tilde{\mathbf{x}}_{\mu,1}\Delta a_\mu \approx \Delta\mathbf{x}_{\mu,2}. \quad (64)$$

Thus, we obtain

$$\Delta a_\mu \approx \tilde{\mathbf{x}}_{\mu,1}^\dagger \mathbf{A}(\mu)\Delta\mathbf{e}_\mu. \quad (65)$$

According to [31], [32] that, for sufficiently high SNR, $\Delta\mathbf{e}_\mu$ is jointly Gaussian distributed with mean ero, namely,

$$\mathbb{E}\{\Delta\mathbf{e}_\mu\} = \mathbf{0}_M. \quad (66)$$

Taking the expectation at both sides of (65) and using (66) yield

$$\mathbb{E}\{\Delta a_\mu\} \approx 0 \quad (67)$$

which implies that at high SNR,

$$\hat{a}_{\mu,\text{init}} \to a_\mu. \quad (68)$$

APPENDIX B
PROOF OF PROPOSITION 1

As the quantity of interest is μ, we take the first-order derivative of $\mu = 2\tan^{-1}(a_\mu)$ as

$$\Delta\mu \approx \frac{2\Delta a_\mu}{1 + a_\mu^2}. \quad (69)$$

Thus, the variance of μ becomes

$$\text{var}(\hat{\mu}) \approx \frac{4}{(1 + a_\mu^2)^2}\text{var}(\hat{a}_\mu)$$
$$= 4\cos^4(\mu/2)\text{var}(\hat{a}_\mu). \quad (70)$$

It is seen in (70) that to calculate the variance of $\hat{\mu}$, we first need to determine the variance of $\hat{a}_\mu$.

The $\hat{a}_\mu$ is obtained by minimizing

$$f(\breve{a}_\mu) = (\mathbf{x}_{\mu,1}\breve{a}_\mu - \mathbf{x}_{\mu,2})^T\mathbf{W}^{-1}(\mu)(\mathbf{x}_{\mu,1}\breve{a}_\mu - \mathbf{x}_{\mu,2}). \quad (71)$$

It is shown in Lemma 1 that $\hat{a}_\mu$ is an unbiased estimate. Thus, for sufficiently high SNR, we are able to approximate the derivative of $f(a_\mu)$ using the first- and second-order terms in its Taylor series expansion about the true value a_μ as

$$0 = f'(\hat{a}_\mu) \approx f'(a_\mu) + f''(a_\mu)\Delta a_\mu. \quad (72)$$

The first- and second-order derivatives of $f(\breve{a}_\mu)$ with respect to a_μ can be simplified as

$$f'(a_\mu) \approx 2\tilde{\mathbf{x}}_{\mu,1}^T\mathbf{W}^{-1}(\mu)\mathbf{A}(\mu)\Delta\mathbf{e}_{\mu,1} \quad (73)$$
$$f''(a_\mu) \approx 2\tilde{\mathbf{x}}_{\mu,1}^T\mathbf{W}^{-1}(\mu)\tilde{\mathbf{x}}_{\mu,1} + 2\tilde{\mathbf{x}}_{\mu,1}^T\mathbf{W}^{-1}(\mu)\Delta\mathbf{x}_{\mu,1}$$
$$+ 2\Delta\mathbf{x}_{\mu,1}^T\mathbf{W}^{-1}(\mu)\tilde{\mathbf{x}}_{\mu,1}. \quad (74)$$

From (72) to (74), the variance of $\hat{a}_\mu$ becomes [36]

$$\text{var}(\hat{a}_\mu)$$
$$\approx \frac{\mathbb{E}\{f'(a_\mu)^2\}}{\mathbb{E}\{f''(a_\mu)^2\}} \approx \frac{\mathbb{E}\{f'(a_\mu)^2\}}{[\mathbb{E}\{f''(a_\mu)\}]^2}$$
$$= \frac{\tilde{\mathbf{x}}_{\mu,1}^T\mathbf{W}^{-1}(\mu)\mathbf{A}(\mu)\mathbb{E}\{\Delta\mathbf{e}_{\mu,1}\Delta\mathbf{e}_{\mu,1}^T\}\mathbf{A}^T(\mu)\mathbf{W}^{-1}(\mu)\tilde{\mathbf{x}}_{\mu,1}}{\left(\tilde{\mathbf{x}}_{\mu,1}^T\mathbf{W}^{-1}(\mu)\tilde{\mathbf{x}}_{\mu,1}\right)^2}$$

$$(75)$$

where

$$\mathbb{E}\{f''(a_\mu)\} \approx \tilde{\mathbf{x}}_{\mu,1}^T\mathbf{W}^{-1}(\mu)\tilde{\mathbf{x}}_{\mu,1}. \quad (76)$$

Substituting (25) into (75) and using $\mathbf{A}(\mu)\tilde{\mathbf{e}}_{\mu,1} = \mathbf{0}_{M-1}$ again, (75) is reduced to

$$\mathrm{var}(\hat{a}_\mu) \approx \frac{\sigma^2}{\tilde{\gamma}_{\mu,1}^2 \left(\tilde{\mathbf{x}}_{\mu,1}^T \mathbf{W}^{-1}(\mu) \tilde{\mathbf{x}}_{\mu,1} \right)}. \tag{77}$$

Substituting (77) into (70) results in the variance of $\hat{\mu}$, i.e.,

$$\mathrm{var}(\hat{\mu}) \approx \frac{4\cos^4(\mu/2)\sigma^2}{\tilde{\gamma}_{\mu,1}^2 \left(\tilde{\mathbf{x}}_{\mu,1}^T \mathbf{W}^{-1}(\mu) \tilde{\mathbf{x}}_{\mu,1} \right)}. \tag{78}$$

After a similar derivation, it is easy to obtain the variance of $\hat{\nu}$ as

$$\mathrm{var}(\hat{\nu}) \approx \frac{4\cos^4(\nu/2)\sigma^2}{\tilde{\gamma}_{\nu,1}^2 \left(\tilde{\mathbf{x}}_{\nu,1}^T \mathbf{W}^{-1}(\nu) \tilde{\mathbf{x}}_{\nu,1} \right)}. \tag{79}$$

This completes the proof of Proposition 1.

APPENDIX C
PROOFS OF (41) AND (42)

Since $\mathbf{K}_{\mu,1}$ and $\mathbf{K}_{\mu,2}$ are the real and imaginary parts of $\mathbf{T}_{M-1}^T \mathbf{J}_{\mu,2} \mathbf{T}_M$, according to (17), they can be rewritten as

$$\mathbf{K}_{\mu,1} = \frac{1}{2} \mathbf{T}_{M-1}^H (\mathbf{J}_{\mu,1} + \mathbf{J}_{\mu,2}) \mathbf{T}_M \tag{80}$$

$$\mathbf{K}_{\mu,2} = \frac{1}{2} \mathbf{T}_{M-1}^H j(\mathbf{J}_{\mu,1} - \mathbf{J}_{\mu,2}) \mathbf{T}_M. \tag{81}$$

Substituting (80) and (81) into (22), we obtain

$$\mathbf{A}(\mu) = \frac{1}{2} \mathbf{T}_{M-1}^H \big((\mathbf{J}_{\mu,1} + \mathbf{J}_{\mu,2}) a_\mu - j(\mathbf{J}_{\mu,1} - \mathbf{J}_{\mu,2}) \big) \mathbf{T}_M. \tag{82}$$

Using

$$a_\mu = \tan(\mu/2) = \frac{e^{j\mu} - 1}{j(e^{j\mu} + 1)} \tag{83}$$

Expression (82) can be written as

$$\mathbf{A}(\mu) = \frac{1}{j(e^{j\mu} + 1)} \mathbf{T}_{M-1}^H \left(e^{j\mu} \mathbf{J}_{\mu,1} - \mathbf{J}_{\mu,2} \right) \mathbf{T}_M. \tag{84}$$

We then have

$$\begin{aligned}
\mathbf{W}(\mu) &= \mathbf{A}(\mu)\mathbf{A}^T(\mu) = \mathbf{A}(\mu)\mathbf{A}^H(\mu) \\
&= \frac{1}{2(1 + \cos(\mu))} \mathbf{T}_{M-1}^H \left(e^{j\mu} \mathbf{J}_{\mu,1} - \mathbf{J}_{\mu,2} \right) \\
&\quad \times \left(e^{j\mu} \mathbf{J}_{\mu,1} - \mathbf{J}_{\mu,2} \right)^H \mathbf{T}_{M-1}.
\end{aligned} \tag{85}$$

In order to proceed, we set $\mathbf{V} = (e^{j\mu}\mathbf{J}_{\mu,1} - \mathbf{J}_{\mu,2})(e^{j\mu}\mathbf{J}_{\mu,1} - \mathbf{J}_{\mu,2})^H$ which can be further expressed as

$$\mathbf{V} = \mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \bar{\mathbf{V}} \left[\mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \right]^H \tag{86}$$

where $\bar{\mathbf{V}} = 2\mathbf{I}_{Q-1} - \mathbf{J}_{\mu,1}\mathbf{J}_{\mu,2}^T - \mathbf{J}_{\mu,2}\mathbf{J}_{\mu,1}^T$ with its inverse being

$$\mathbf{V}^{-1} = \mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \bar{\mathbf{V}}^{-1} \left[\mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \right]^H. \tag{87}$$

By using (87), the inverse of $\mathbf{W}(\mathbf{a}_\mu)$ becomes

$$\mathbf{W}^{-1}(\mu) = 2(\cos(\mu) + 1)$$

$$\left(\mathbf{T}_{M-1}^H \mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \bar{\mathbf{V}}^{-1} \left[\mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \right]^H \mathbf{T}_{M-1} \right). \tag{88}$$

Let us now determine $\tilde{\mathbf{x}}_{\mu,1}^T \mathbf{W}^{-1}(\mu) \tilde{\mathbf{x}}_{\mu,1}$. To begin with, $\tilde{\mathbf{x}}_{\mu,1}$ and its transpose are calculated as

$$\begin{aligned}
\tilde{\mathbf{x}}_{\mu,1} &= \mathbf{K}_{\mu,1}\mathbf{e}_{\mu,1} = \mathbf{K}_{\mu,1}\mathbf{T}_M^H \mathbf{u}_{\mu,1} \\
&= \frac{1}{2} \mathbf{T}_{M-1}^H (\mathbf{J}_{\mu,1} + \mathbf{J}_{\mu,2}) \mathbf{u}_{\mu,1}
\end{aligned} \tag{89}$$

$$\begin{aligned}
\tilde{\mathbf{x}}_{\mu,1}^T &= \tilde{\mathbf{x}}_{\mu,1}^H \\
&= \frac{1}{2} \mathbf{u}_{\mu,1}^H (\mathbf{J}_{\mu,1} + \mathbf{J}_{\mu,2})^T \mathbf{T}_{M-1}.
\end{aligned} \tag{90}$$

Combining (88)–(90) yields

$$\begin{aligned}
&\tilde{\mathbf{x}}_{\mu,1}^T \mathbf{W}^{-1}(\mu) \tilde{\mathbf{x}}_{\mu,1} \\
&= \frac{1}{2}(\cos(\mu) + 1) \\
&\quad \times \Big(\mathbf{u}_{\mu,1,1}^H \mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \bar{\mathbf{V}}^{-1} \left[\mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \right]^H \mathbf{u}_{\mu,1,1} \\
&\quad + \mathbf{u}_{\mu,1,1}^H \mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \bar{\mathbf{V}}^{-1} \left[\mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \right]^H \mathbf{u}_{\mu,1,2} \\
&\quad + \mathbf{u}_{\mu,1,2}^H \mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \bar{\mathbf{V}}^{-1} \left[\mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \right]^H \mathbf{u}_{\mu,1,1} \\
&\quad + \mathbf{u}_{\mu,1,2}^H \mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \bar{\mathbf{V}}^{-1} \left[\mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} \right]^H \mathbf{u}_{\mu,1,2} \Big)
\end{aligned} \tag{91}$$

where $\mathbf{u}_{\mu,1,1} = \mathbf{J}_{\mu,1}\mathbf{u}_{\mu,1}$ and $\mathbf{u}_{\mu,1,2} = \mathbf{J}_{\mu,2}\mathbf{u}_{\mu,1}$. It should be noted that

$$\mathbf{u}_{\mu,1,1}^H \mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} = \frac{1}{\sqrt{M}} \mathbf{1}_{M-1}^T \tag{92}$$

$$\mathbf{u}_{\mu,1,2}^H \mathrm{diag}\{\mathbf{J}_{\mu,1}\mathbf{s}_\mu\} = e^{-j\mu} \frac{1}{\sqrt{M}} \mathbf{1}_{M-1}^T. \tag{93}$$

As a result, (91) is simplified as

$$\tilde{\mathbf{x}}_{\mu,1}^T \mathbf{W}^{-1}(\mu) \tilde{\mathbf{x}}_{\mu,1} = \frac{(\cos(\mu) + 1)^2}{M} \cdot \mathbf{1}_{M-1}^T \bar{\mathbf{V}}^{-1} \mathbf{1}_{M-1}. \tag{94}$$

We are now in position to calculate $\bar{\mathbf{V}}^{-1}$, which is used to further simplify (95). After some matrix operations, we obtain

$$\bar{\mathbf{V}}^{-1} = \frac{1}{M} \begin{bmatrix}
M-1 & M-2 & \cdots & 2 & 1 \\
M-2 & 2(M-2) & \cdots & 4 & 2 \\
\vdots & \vdots & & \vdots & \vdots \\
2 & 4 & \cdots & 2(M-2) & M-2 \\
1 & 2 & \cdots & M-2 & M-1
\end{bmatrix} \tag{95}$$

which indicates that $\bar{\mathbf{V}}^{-1}$ is a symmetric matrix and its (m,n) entry is defined as

$$\left[\bar{\mathbf{V}}^{-1} \right]_{m,n} = \begin{cases} \frac{n(M-m)}{M}, & 1 \le n \le m \le M-1 \\ \frac{m(M-n)}{M}, & 1 \le m < n \le M-1. \end{cases} \tag{96}$$

Since $\bar{\mathbf{V}}^{-1}$ is symmetric, to determine the value of $\mathbf{1}_{M-1}^T \bar{\mathbf{V}}^{-1} \mathbf{1}_{M-1}$, we just need to calculate the trace and sum of the entries in the lower triangular part of $\bar{\mathbf{V}}^{-1}$. First, the sum of the entries in its lower triangular part can be computed as

$$\frac{1}{M} \sum_{j=1}^{M-1} \sum_{i=1}^{M-j} ij = \frac{(M-1)(M+1)(M+2)}{24}. \tag{97}$$

Then the trace of $\mathbf{V}^{-1}$ is

$$\frac{1}{M} \sum_{i=1}^{M-1} i(M-i) = \frac{(M-1)(M+1)}{6}. \tag{98}$$

Now (94) becomes

$$\tilde{\mathbf{x}}_{\mu,1}^T \mathbf{W}^{-1}(\mu)\tilde{\mathbf{x}}_{\mu,1} = (\cos(\mu)+1)^2 \cdot \frac{M^2-1}{12}. \tag{99}$$

Substituting (99) into (78) leads to

$$\begin{aligned}
\mathrm{var}(\hat{\mu}) &\approx \frac{\sigma^2}{\tilde{\gamma}_{\mu,1}^2} \frac{4\cos^4(\mu/2)}{(\cos(\mu)+1)^2} \frac{12}{M^2-1} \\
&= \frac{6\sigma^2}{(M^2-1)MN|\kappa|^2}
\end{aligned} \tag{100}$$

which is the CRB [11]. It is worth mentioning that in the last equality of (100) we use $\tilde{\gamma}_{\mu,1} = \sqrt{2MN}|\kappa|$, $\cos(\mu) + 1 = 2(\cos(\mu/2))^2$, and $1 + a_\mu^2 = (\cos(\mu/2))^{-2}$. Equation (42) follows in the same way, or by simply appealing to symmetry.

REFERENCES

[1] M. Morelli, L. Marchetti, and M. Moretti, "Maximum likelihood frequency estimation and preamble identification in OFDMA-based WiMAX systems," *IEEE Trans. Wireless Commun.*, vol. 13, no. 3, pp. 1582–1592, Mar. 2014.

[2] L. Borcea, T. Callaghan, and G. Papanicolaou, "Synthetic aperture radar imaging and motion estimation via robust principal component analysis," *SIAM J. Imaging Sci.*, vol. 6, no. 3, pp. 1445–1476, 2013.

[3] F. Zhou, G. Sun, X. Xia, M. Xing, and Z. Bao, "Stepped frequency synthetic preprocessing algorithm for inverse synthetic aperture radar imaging in fast moving target echo model," *IET Radar, Sonar, Navig.*, vol. 8, no. 8, pp. 864–874, 2014.

[4] A. Hassanien, S. A. Vorobyov, and J. Y. Park, "Joint transmit array interpolation and transmit beamforming for source localization in MIMO radar with arbitrary arrays," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP 2014)*, Vancouver, BC, Canada, 2013, pp. 4139–4143.

[5] C. Qian, L. Huang, W.-J. Zeng, and H. C. So, "Direction-of-arrival estimation for coherent signals without knowledge of source number," *IEEE Sensors J.*, vol. 14, no. 9, pp. 3267–3273, 2014.

[6] J. Liang, L. Xu, J. Li, and P. Stoica, "On designing the transmission and reception of multistatic continuous active sonar systems," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 50, no. 1, pp. 285–299, Jan. 2014.

[7] C. T. C. Nguyen, "MEMS-based RF channel selection for true software-defined cognitive radio and low-power sensor communications," *IEEE Commun. Mag.*, vol. 51, no. 4, pp. 110–119, Apr. 2013.

[8] D. C. Rife and R. R. Boorstyn, "Single tone parameter estimation from discrete-time observations," *IEEE Trans. Inf. Theory*, vol. IT-20, no. 5, pp. 591–598, Sep. 1974.

[9] T. Abatzoglou, "A fast maximum likelihood algorithm for frequency estimation of a sinusoid based on Newton's method," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 33, no. 1, pp. 77–89, Jan. 1985.

[10] M. P. Clark and L. L. Scharf, "Two-dimensional modal analysis based on maximum likelihood," *IEEE Trans. Signal Process.*, vol. 42, no. 6, pp. 1443–1452, Jun. 1994.

[11] H. C. So and K. W. Chan, "Approximate maximum likelihood algorithms for two-dimensional frequency estimation of a complex sinusoid," *IEEE Trans. Signal Process.*, vol. 54, no. 8, pp. 3231–3237, Aug. 2006.

[12] M. D. Rahman and K. B. Yu, "Total least squares approach for frequency estimation using linear prediction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, no. 10, pp. 1440–1454, Oct. 1987.

[13] H. C. So and K. W. Chan, "A generalized weighted linear predictor frequency estimation approach for a complex sinusoid," *IEEE Trans. Signal Process.*, vol. 54, no. 4, pp. 1304–1315, Apr. 2006.

[14] V. Pisarenko, "The retrieval of harmonics by linear prediction," *Geophys. J. Roy. Astronom. Soc.*, vol. 33, pp. 347–366, 1973.

[15] R. Elasmi-Ksibi, H. Besbes, R. L. Valcarce, and S. Cherif, "Frequency estimation of real-valued single-tone in colored noise using multiple autocorrelation lags," *Signal Process.*, vol. 90, no. 7, pp. 2303–2307, 2010.

[16] K. W. K. Lui and H. C. So, "Modified Pisarenko harmonic decomposition for single-tone frequency estimation," *IEEE Trans. Signal Process.*, vol. 56, no. 7, pp. 3351–3356, Jul. 2008.

[17] Y. Cao, G. Wei, and F. Chen, "A closed-form expanded autocorrelation method for frequency estimation of a sinusoid," *Signal Process.*, vol. 92, no. 4, pp. 885–892, 2012.

[18] Y. Xiao and Y. Tadokoro, "On Pisarenko and constrained Yule-Walker estimator of tone frequency," *IEICE Trans. Fundam. Electron. Commun. Comput. Sci.*, vol. E77-A, no. 8, pp. 1404–1406, 1994.

[19] P. Stoica and A. Nehorai, "Study of the statistical performance of the Pisarenko harmonic decomposition method," *IEE Proc. F Commun., Radar Signal Process.*, vol. 135, no. 2, pp. 161–168, 1988.

[20] P. Händel, "Markov-based single-tone frequency estimation," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 45, no. 2, pp. 230–232, Feb. 1998.

[21] H. C. So, F. K. W. Chan, and W. Sun, "Subspace approach for fast and accurate single-tone frequency estimation," *IEEE Trans. Signal Process.*, vol. 59, no. 2, pp. 827–831, Feb. 2011.

[22] H. C. So, F. K. W. Chan, W. H. Lau, and C. F. Chan, "An efficient approach for two-dimensional parameter estimation of a single-tone," *IEEE Trans. Signal Process.*, vol. 58, no. 4, pp. 1999–2009, Apr. 2010.

[23] H. Fu and P. Y. Kam, "MAP/ML estimation of the frequency and phase of a single sinusoid in noise," *IEEE Trans. Signal Process.*, vol. 55, no. 3, pp. 834–845, Mar. 2007.

[24] S. Rouquette and M. Najim, "Estimation of frequencies and damping factors by two-dimensional ESPRIT type methods," *IEEE Trans. Signal Process.*, vol. 49, no. 1, pp. 237–245, Jan. 2001.

[25] P. Stoica and K. Sharman, "Maximum likelihood methods for direction-of-arrival estimation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, no. 7, pp. 1132–1143, Jul. 1990.

[26] J. Xin and A. Sano, "Computationally efficient subspace-based method for direction-of-arrival estimation without eigendecomposition," *IEEE Trans. Signal Process.*, vol. 52, no. 4, pp. 876–893, Apr. 2004.

[27] A. Lee, "Centro-Hermitian and skew-centro-Hermitian matrices," *Linear Algebra Appl.*, vol. 29, no. 5, pp. 205–210, 1980.

[28] T. Soderstrom and P. Stoica, *System Identification*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1989.

[29] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1993.

[30] M. Haardt and J. A. Nossek, "Unitary ESPRIT: How to obtain increased estimation accuracy with a reduced computational burden," *IEEE Trans. Signal Process.*, vol. 43, no. 5, pp. 1232–1242, May, 1995.

[31] M. D. Zoltowski and G. M. Kautz, "Performance analysis of eigenstructure based DOA estimators employing conjugate symmetric beamformers," in *Proc. 6th SSAP Workshop Statist. Signal Array Process.*, Victoria, BC, Canada, 1992, pp. 384–387.

[32] M. Pesavento, A. B. Gershman, and M. Haardt, "Unitary root-MUSIC with a real-valued eigendecomposition: A theoretical and experimental performance study," *IEEE Trans. Signal Process.*, vol. 48, no. 5, pp. 1306–1314, May 2000.

[33] X. Liu and N. D. Sidiropoulos, "Almost sure identifiability of constant modulus multidimensional harmonic retrieval," *IEEE Trans. Signal Process.*, vol. 50, no. 9, pp. 2366–2368, Sep. 2002.

[34] M. Pesavento, C. F. Mecklenbrauker, and J. F. Boehme, "Multidimensional rank reduction estimator for parametric MIMO channel models," *EURASIP J. Appl. Signal Process.*, vol. 2004, pp. 1354–1363, 2004.

[35] P. Comon and G. H. Golub, "Tracking a few extreme singular values and vectors in signal processing," *Proc. IEEE*, vol. 78, no. 8, pp. 1327–1343, Aug. 1990.

[36] H. C. So, Y. T. Chan, K. C. Ho, and Y. Chen, "Simple formulas for bias and mean square error computation," *IEEE Signal Process. Mag.*, vol. 30, no. 4, pp. 162–165, Jul. 2013.
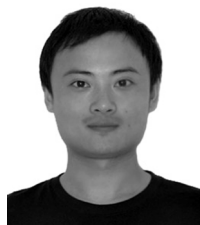
Cheng Qian (S'14) was born in Deqing, Zhejiang, China. He received the B.E. degree in communication engineering from Hangzhou Dianzi University, Hangzhou, China, in 2011, and M.E. degree in communication and information engineering from Harbin Institute of Technology (HIT), China, in 2013.

Since September 2013, he has been working towards the Ph.D. degree in the field of information and communication engineering at HIT. In addition, he is currently a visiting scholar working with the Signal and Tensor Analytics Research group at Department of Electrical and Computer Engineering, University of Minnesota. His research interests are in array signal processing, MIMO radar and phase retrieval.
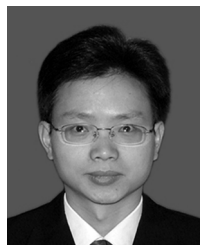
Lei Huang (M'07–SM'14) was born in Guangdong, China. He received the B.Sc., M.Sc., and Ph.D. degrees in electronic engineering from Xidian University, Xi'an, China, in 2000, 2003, and 2005, respectively.

From 2005 to 2006, he was a Research Associate with the Department of Electrical and Computer Engineering, Duke University, Durham, NC. From 2009 to 2010, he was a Research Fellow with the Department of Electronic Engineering, City University of Hong Kong and a Research Associate with the Department of Electronic Engineering, The Chinese University of Hong Kong. From 2011 to 2014, he was a Professor with the Department of Electronic and Information Engineering, Harbin Institute of Technology Shenzhen Graduate School. Since November 2014, he has joined the Department of Information Engineering, Shenzhen University, where he is currently a Chair Professor. His research interests include spectral estimation, array signal processing, statistical signal processing, and their applications in radar and wireless communication systems.

He is currently serving as an Associate Editor for IEEE Transactions on Signal Processing and Digital Signal Processing.

Hing Cheung So (S'90–M'95–SM'07–F'15) was born in Hong Kong. He received the B.Eng. degree from the City University of Hong Kong and the Ph.D. degree from The Chinese University of Hong Kong, both in electronic engineering, in 1990 and 1995, respectively. From 1990 to 1991, he was an Electronic Engineer with the Research and Development Division, Everex Systems Engineering Ltd., Hong Kong. During 1995–1996, he was a Postdoctoral Fellow with The Chinese University of Hong Kong. From 1996 to 1999, he was a Research Assistant Professor with the Department of Electronic Engineering, City University of Hong Kong, where he is currently an Associate Professor. His research interests include statistical signal processing, fast and adaptive algorithms, signal detection, robust estimation, source localization and sparse approximation.

He has been on the editorial boards of IEEE Signal Processing Magazine (2014-), IEEE Transactions on Signal Processing (2010–2014), Signal Processing (2010-), and Digital Signal Processing (2011-). In addition, he is an elected member in Signal Processing Theory and Methods Technical Committee (2011-) of the IEEE Signal Processing Society where he is chair in the awards subcommittee (2015-). He has been elected Fellow of IEEE in recognition of his contributions to spectral analysis and source localization in 2015.

Nicholas D. Sidiropoulos (S'90–M'92–SM'99–F'09) received the Diploma in electrical engineering from the Aristotelian University of Thessaloniki, Greece, and M.S. and Ph.D. degrees in electrical engineering from the University of Maryland-College Park, in 1988, 1990 and 1992, respectively.

He served as Assistant Professor at the University of Virginia (1997–1999); Associate Professor at the University of Minnesota-Minneapolis (2000–2002); Professor at the Technical University of Crete, Greece (2002–2011); and Professor at the University of Minnesota-Minneapolis (2011-), where he currently holds an ADC Chair in Digital Technology. His current research focuses primarily on signal and tensor analytics, with applications in cognitive radio, big data, and preference measurement. He received the NSF/CAREER Award (1998), the IEEE Signal Processing Society (SPS) Best Paper Award (2001, 2007, 2011), and the IEEE SPS Meritorious Service Award (2010). He has served as IEEE SPS Distinguished Lecturer (2008–2009), and Chair of the IEEE Signal Processing for Communications and Networking Technical Committee (2007–2008). He received the Distinguished Alumni Award of the Department of Electrical and Computer Engineering, University of Maryland, College Park (2013). He was elected Fellow of EURASIP in 2014.

Junhao Xie (M'02–SM'15) received the B.S. degree in electronic engineering from Harbin Institute of Shipbuilding Engineering, the M.S. degree in signal and information processing from Harbin Engineering University and the Ph.D. degree in communication and information system from Harbin Institute of Technology, in 1992, 1995 and 2001 respectively.

From 2004 to 2006, he was a visiting scholar at Curtin University of Technology, Perth, W.A., Australia, where he dealt with radar remote sensing signal processing. He is a Professor in the Department of Electronic Engineering, Harbin Institute of Technology. His past and present research interests involve radar system analysis and modeling, array signal processing, target detection and estimation, and ocean remote sensing for shipborne high-frequency surface wave radar (HFSWR) and inverse synthetic aperture radar (ISAR).