Fast Missing-Data IAA with Application to Notched Spectrum SAR

Johan Karlsson, *Member, IEEE,* William Rowe, Luzhou Xu, *Member, IEEE,*
George-Othon Glentis, *Member, IEEE,* and Jian Li, *Fellow, IEEE*

***Abstract*—Recently the spectral estimation method Iterative Adaptive Approach (IAA) has been shown to provide higher resolution and lower sidelobes than comparable spectral estimation methods. The computational complexity is higher than other methods such as the periodogram (matched filter). Fast algorithms have been developed that considerably reduce the computational complexity of IAA by utilizing Toeplitz and Vandermonde structures. For the missing data case several of these structures are lost, and existing fast algorithms are only efficient when the number of *available* samples is small. In this work we consider the case where the number of *missing* samples is small. This allows us to use a low rank completion to transform the problem to the structured problem. We compare the computational speed of the algorithm with the state-of-the-art and demonstrate the utility in a frequency notched SAR imaging problem.**

***Index Terms*—IAA, Missing Data, FFT, Fast Algorithm, SAR, Spectral Estimation.**

I. Introduction

A fundamental task in spectral estimation is to estimate noisy sinusoids' frequencies and amplitudes from a set of measurements [1]. The common solution is to use the periodogram to estimate the spectra, a method which in general suffers from large sidelobes and poor resolution. In the case of missing data, the sidelobe problem becomes even worse. The sidelobes increase due to modulation in the sampling domain by the incomplete sampling pattern relative to the uniform sampling in the Fourier matrix. It is therefore often desirable to interpolate the data onto a uniform sampling pattern or estimate the missing data if patches of data are missing.

A recently developed high resolution nonparametric spectral estimation technique, the iterative adaptive approach (IAA) [2], can also be used in the case with missing data (MIAA) [3]. This is a method based on iterative weighted minimization, where the weight is updated to increase the resolution and suppress sidelobes. IAA provides resolution superior to the periodogram, and has the advantage that only a single snapshot is required. The major drawback of IAA is the computational costs for a direct implementation. In two recent papers ([4], [5]) implementations of IAA were developed based on FFT operations, which considerably speeds up the algorithm and makes it applicable for larger problems. These fast implementations utilize Topelitz and Vandermonde structures that arise when the sampling grid is uniform and complete. When the sampling grid is not complete the Toeplitz structure of the covariance matrix is lost, and the fast implementations are not efficient when the number of missing samples is small. To resolve this, we use a low rank completion to transform the problem to the structured problem where the covariance matrix is Toeplitz. This leads to the main contribution of this paper, a fast implementation of MIAA which is considerably faster than the state-of-the-art for the case when the missing data is small. Finally, an application to synthetic aperture radar (SAR) imaging is examined.

In many practical SAR scenarios the set of measured data is incomplete, due to, e.g., interference, jamming or data dropouts [6], [7], resulting in an incomplete data set. Here we consider the case where a proportion of the samples ($< 50\%$) are missing due to frequency notching being used to suppress interference. This is a situation which is not uncommon in ultra high frequency (UHF) or very high frequency (VHF) SAR, where the spectrum is often crowded. Here we apply MIAA for the recovery of lost fast time samples due to frequency notching in the occupied bands. This is shown to significantly improve the resulting SAR image quality.

In Section II we set up the data model, discuss the spectral estimation problem, and introduce the algorithm IAA. Section III describes computational complexities of the IAA algorithm and how to utilize the Toeplitz/Vandermonde structures in the problem to significantly decrease the computational complexity. In Section IV the missing data algorithm is discussed and we present the new fast algorithm for missing data IAA. In Section V it is shown how to utilize the results from Section IV in order to provide a computationally efficient recovery of the missing data. In Section VI we present examples that illustrate the computational benefits of the new algorithms and then apply the algorithm to data recovery in sparse SAR imaging in the presence of spectrum notches.

The notation used in this paper will be briefly defined. A vector is represented by a bold face lower case letter ($\mathbf{x}$) and a matrix is represented by a bold face upper case letter ($\mathbf{X}$). The transpose of a matrix is $(\cdot)^T$ and the conjugate

This work was supported in part by Swedish Research Council, NSF CCF-1218388, and the Department of Defense. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. Johan Karlsson is with the department of Mathematics, Royal Institute of Technology (KTH), Stockholm, 100 44, Sweden, johan.karlsson@math.kth.se. William Rowe, Luzhou Xu, and Jian Li are with the department of Electrical and Computer Engineering, University of Florida, Gainesville, Florida 32611, wrowe001@ufl.edu, xuluzhou@dsp.ufl.edu, li@dsp.ufl.edu. Luzhou Xu and Jian Li are also affiliated with the IAA, Inc. George-Othon Glentis is with the Department of Science and Technology of Telecommunications, University of Peloponnese, Tripolis, 22100, Greece, gglentis@uop.gr.

transpose is $(\cdot)^*$. The conjugate of a complex number is given by $\overline{(\cdot)}$. A matrix $\mathbf{X}$ is Hermitian if $\mathbf{X} = \mathbf{X}^*$. For Hermitian matrices $\mathbf{X}, \mathbf{Y}$, let $\mathbf{X} > \mathbf{Y}$ ($\mathbf{X} \geq \mathbf{Y}$) denote that the matrix $\mathbf{X} - \mathbf{Y}$ is positive definite (semidefinite). A fast Fourier transform operation (FFT) of size N is denoted by $\mathcal{F}(\cdot)_N$, where appropriate zero padding is performed if necessary without discussion. A subscript on a vector or matrix of g denotes the data was measured (given) while a subscript of m represents data that was not measured (missing). A denotation of $(\cdot)_{1:K}$ represents an indexing operation, i.e. elements 1 to K of a vector.

II. Spectral estimation and IAA

A. Data model

Consider the problem of recovering the spectral content from a measured signal. Let $\mathbf{y} = (y_0, y_1, \ldots, y_{N-1})^T$ denote a sampled data sequence of length N and let

$$\mathbf{A} = (\mathbf{a}(\omega_0), \ldots, \mathbf{a}(\omega_{K-1})) \tag{1}$$

be an oversampled Fourier matrix such that $K > N$. The columns of $\mathbf{A}$ are $\mathbf{a}(\omega_k) = \left(1,\, e^{j\omega_k}, \ldots,\, e^{j(N-1)\omega_k}\right)^T$ which correspond to the frequency vectors and ω_k corresponds to the frequency grid point $\omega_k = \frac{2\pi k}{K}$, for $k = 0, \ldots, K-1$. Let $\mathbf{x} = (x_0, x_1, \ldots, x_{K-1})^T$ where x_k denotes the complex spectral content at frequency ω_k of the signal $\mathbf{y}$. The data model can then be formulated as

$$\mathbf{y} = \mathbf{A}\mathbf{x} + \mathbf{e}, \tag{2}$$

where the noise contribution is $\mathbf{e}$. The goal of the problem is to estimate $\mathbf{x}$.

The most common method for solving this is the periodogram (matched filter method),

$$x_k = \frac{\mathbf{a}(\omega_k)^*\mathbf{y}}{\mathbf{a}(\omega_k)^*\mathbf{a}(\omega_k)}, \quad k = 0, 1, \ldots, K-1, \tag{3}$$

which may be calculated using the FFT. This is computationally efficient, but suffers from high sidelobes and poor resolution. One way to overcome these issues is to use data-adaptive methods such as the Capon method [8], [9], Amplitude and Phase Estimation (APES) [10], [11], or Iterative Adaptive Approach (IAA) [2]. Here we will focus explicitly on IAA as it has shown promise in the fields of radar imaging, sonar, communications [2], medical diagnostics [12], information forensics [13], and general spectral estimation [14].

B. Iterative Adaptive Approach (IAA)

IAA seeks to find a spectral estimate x_k in (2) by modeling the rest of the spectrum x_ℓ, $\ell \neq k$, as interference [2]. The inference refers to all the signals at frequency grid points other than the grid point of interest ω_k and is modelled as $x_\ell \in N(0, p_\ell)$ in (2) for $\ell \neq k$. The covariance matrix $\mathbf{Q}_k$ of the interference is then given by

$$\mathbf{Q}_k = \mathbf{R} - p_k \mathbf{a}(\omega_k)\mathbf{a}(\omega_k)^*,$$

where

$$\mathbf{R} = \sum_{\ell=0}^{K-1} p_\ell \mathbf{a}(\omega_\ell)\mathbf{a}(\omega_\ell)^* = \mathbf{A}\mathbf{P}\mathbf{A}^*. \tag{4}$$

Here $\mathbf{R}$ is the covariance matrix of the data and $\mathbf{P} = \operatorname{diag}(\mathbf{p})$, where $\mathbf{p} = (p_0, p_1, \ldots, p_{K-1})^T$, and $p_\ell = |x_\ell|^2$ denotes the power estimate at the frequency grid point ω_ℓ, for $\ell = 0, 1, \ldots, K-1$. Maximizing the likelihood of x_k then results in minimization of the weighted quadratic cost function

$$(\mathbf{y} - \mathbf{a}(\omega_k)x_k)^* \mathbf{Q}_k^{-1} (\mathbf{y} - \mathbf{a}(\omega_k)x_k), \tag{5}$$

where the optimal solution is given by

$$x_k = \frac{\mathbf{a}(\omega_k)^*\mathbf{Q}_k^{-1}\mathbf{y}}{\mathbf{a}(\omega_k)^*\mathbf{Q}_k^{-1}\mathbf{a}(\omega_k)}, \quad k = 0, 1, \ldots, K-1. \tag{6}$$

Using the matrix inversion lemma, (6) equals

$$x_k = \frac{\mathbf{a}(\omega_k)^*\mathbf{R}^{-1}\mathbf{y}}{\mathbf{a}(\omega_k)^*\mathbf{R}^{-1}\mathbf{a}(\omega_k)}, \quad k = 0, 1, \ldots, K-1. \tag{7}$$

This considerably speeds up the calculation since (7) does not require the computation of the inverse of the interference covariance matrix $\mathbf{Q}_k$ for each frequency grid point. Note that $\mathbf{R}$ depends on $\mathbf{x}$, hence solving Equations (4) and (7) is a non-trivial task. IAA handles this in an iterative manner. The algorithm starts with an initial solution which is often taken as the Periodogram (3). The following steps are then taken:

1) The covariance matrix $\mathbf{R}$ is calculated using (4),
2) x_k is calculated using (7) for $k = 0, 1 \ldots, K-1$.

Steps 1) and 2) are repeated until convergence, and the spectral estimate in the point ω_k is given by $p_k = |x_k|^2$. From empirical results usually $10 - 15$ iterations are sufficient for the algorithm to converge [2]. In the scenarios considered here, the steering matrix $\mathbf{A}$ is an oversampled Fourier matrix. However, IAA performs equally well for other applications such as imaging and channel estimation in communications where the columns of $\mathbf{A}$ consists of other basis functions such as delayed or Doppler shifted versions of probing/training signals [15].

Comparing (3) to (6) we see that the only difference between the periodogram and IAA is the weighting matrix $\mathbf{Q}_k$. Assuming that the interference covariance $\mathbf{Q}_k$ is known, (6) gives a better estimate of x_k than (3) under quite general conditions [14]. IAA utilizes this in order to achieve better resolution than the periodogram. The matrices $\mathbf{Q}_k$ and $\mathbf{R}$ are typically well-conditioned if the signal contains noise.[1] This is also the reason that IAA can work with as little as a single snapshot, which is beneficial compared to existing high resolution techniques that typically require many snapshots to estimate the covariance matrix.

III. Computational complexities and fast calculation of IAA

IAA provides high resolution estimates with low sidelobes, but it is also rather computationally demanding. In each iteration it requires evaluation of the numerator and denominator of the expression (7), denoted by

$$\Phi_{\mathrm{N}}(\omega) = \mathbf{a}(\omega)^*\mathbf{R}^{-1}\mathbf{y}, \tag{8}$$

$$\Phi_{\mathrm{D}}(\omega) = \mathbf{a}(\omega)^*\mathbf{R}^{-1}\mathbf{a}(\omega), \tag{9}$$

[1] For high SNR scenarios with sparse signal spectrum the matrices $\mathbf{Q}_k$ and $\mathbf{R}$ may be ill conditioned. This may be handled by regularizing IAA [16].

at each of the points ω_k, $k = 0, 1, \ldots, K-1$. Using a brute force approach, in each iteration the matrix $\mathbf{R}$ would be inverted and then $\mathbf{R}^{-1}\mathbf{a}(\omega_k)$ computed for $k = 0, 1, \ldots, K-1$. This takes $\mathcal{O}(N^2K)$ which is too computationally demanding in many applications [17]. In situations when N and K are large, memory requirements may prevent us from even creating the matrices $\mathbf{A}$ and $\mathbf{R}$. However, since the matrices have known structure (Toeplitz/Vandermonde), the matrices can be represented by vectors and operations performed on those vectors instead of the full matrix. Next we will review the algorithm proposed in [4], where these structures are used to calculate IAA several orders of magnitudes quicker by utilizing FFT operations. In the following discussion we frequently use fast calculations with Toeplitz matrices. These are briefly reviewed in Appendix B.

1) Calculate $\mathbf{R}^{-1}$*:* The first step in the algorithm is to calculate the matrix $\mathbf{R}$. By noting that $\mathbf{P}$ is diagonal and $\mathbf{A}$ is a Vandermonde matrix, $\mathbf{R} = \mathbf{APA}^*$ is a Hermitean Toeplitz matrix and it is sufficient to calculate the first column of $\mathbf{R}$. The first column of $\mathbf{PA}^*$ is $\mathbf{p} = [p_1, \ldots, p_K]^T$, and since $\mathbf{A}$ is the partial Fourier matrix (1), the first column of $\mathbf{R}$ is obtained by taking the first N elements in the inverse Fourier transform of $\mathbf{p}$, i.e., $\mathbf{R}_{1:N,1} = K(\mathcal{F}^{-1}(\mathbf{p})_K)_{1:N}$. Note that $\mathbf{R}$ is uniquely defined by its first row since it is Toeplitz and Hermitean.

The next step in IAA is to obtain the Gohberg-Semencul (GS) factorization,

$$\mathbf{R}^{-1} = \mathcal{L}(\mathbf{u}, \mathbf{D})\mathcal{L}(\mathbf{u}, \mathbf{D})^* - \mathcal{L}(\tilde{\mathbf{u}}, \mathbf{D})\mathcal{L}(\tilde{\mathbf{u}}, \mathbf{D})^*,$$

which allows for expressing $\mathbf{R}^{-1}$ in terms of the lower triangular Toeplitz matrices $\mathcal{L}(\mathbf{u}, \mathbf{D}) = (\mathbf{u}, \mathbf{Du}, \mathbf{D}^2\mathbf{u}, \ldots \mathbf{D}^{N-1}\mathbf{u})$ and $\mathcal{L}(\tilde{\mathbf{u}}, \mathbf{D}) = (\tilde{\mathbf{u}}, \mathbf{D}\tilde{\mathbf{u}}, \mathbf{D}^2\tilde{\mathbf{u}}, \ldots \mathbf{D}^{N-1}\tilde{\mathbf{u}})$. Here

$$\mathbf{D} = \begin{pmatrix} \mathbf{0}_{1\times(N-1)} & 0 \\ \mathbf{I}_{N-1} & \mathbf{0}_{(N-1)\times 1} \end{pmatrix}$$

denote the shift matrix and the vectors $\mathbf{u}$ and $\tilde{\mathbf{u}}$ are calculated using the Levinson-Durbin (LD) algorithm in $\mathcal{O}(N^2)$ (see Appendix B and [1] for details). Using the Gohberg-Semencul factorization, we can solve the equation $\mathbf{Rx} = \mathbf{y}$ for any $\mathbf{y}$ in $\mathcal{O}(N \log N)$. Next, the numerator and denominator of the IAA estimate are examined.

2) Calculate $\Phi_{\mathrm{N}}(\omega)$*:* Examining the numerator (8), the first problem is the evaluation of $\mathbf{R}^{-1}\mathbf{y}$. This is done using the GS factorization and efficient Toeplitz matrix-vector multiplication in $\mathcal{O}(N \log N)$ time (see Appendix B). Then the estimate, $\Phi_{\mathrm{N}}(\omega_k)$, can be computed for all k with an FFT operation on $\mathbf{R}^{-1}\mathbf{y}$,

$$\boldsymbol{\phi}_{1-\mathrm{N}} = \mathcal{F}(\mathbf{R}^{-1}\mathbf{y})_K, \tag{10}$$

where $\boldsymbol{\phi}_{1-\mathrm{N}} = (\phi_{\mathrm{N}}(\omega_0), \phi_{\mathrm{N}}(\omega_1), \ldots, \phi_{\mathrm{N}}(\omega_{K-1}))^T$. Hence the numerator can be calculated in $\mathcal{O}(K \log K)$ time.

3) Calculate $\Phi_{\mathrm{D}}(\omega)$*:* The denominator may also be calculated using FFT/IFFT by considering the trigonometric polynomial

$$\phi(z) \quad = \quad \boldsymbol{\alpha}(\bar{z}^{-1})^*\mathbf{R}^{-1}\boldsymbol{\alpha}(z) = \sum_{\ell=-N+1}^{N-1} c_\ell z^\ell, \ \text{ for}$$

where $\boldsymbol{\alpha}(z) = (1, z, z^2, \ldots, z^{N-1})^T$, and c_ℓ is the coefficient for z^ℓ. Since $\bar{z}^{-1} = z$ whenever $|z| = 1$, it follows that $\phi(z)$ is real valued at the unit circle and $c_{-\ell} = \bar{c}_\ell$. Evaluation of the denominator $\phi_{\mathrm{D}}(\omega)$ corresponds to evaluation of $\phi(z)$ at the unit circle

$$\phi_{\mathrm{D}}(\omega) = \phi(z)|_{z=e^{i\omega}} = \sum_{\ell=-N+1}^{N-1} c_\ell e^{j\ell\omega}.$$

Since the grid $\{\omega_k\}_{k=0}^{K-1}$ is uniform, $\phi_{\mathrm{D}}(\omega_k)$ may be computed using FFT from the coefficients $\{c_n\}_{n=-N+1}^{N-1}$. This may be expressed as

$$\boldsymbol{\phi}_{1-\mathrm{D}} = K\mathcal{F}^{-1}(\mathbf{c}) \tag{11}$$

where $\mathbf{c} = (c_0, c_1, \ldots, c_{N-1}, \mathbf{0}_{K-2N+1}^T, c_{-N+1}, \ldots, c_{-1})^T$ and $\boldsymbol{\phi}_{1-\mathrm{D}} = (\phi_{\mathrm{D}}(\omega_0), \phi_{\mathrm{D}}(\omega_1), \ldots, \phi_{\mathrm{D}}(\omega_{K-1}))^T$. Using the GS factorization of $\mathbf{R}^{-1}$ and noting that the elements of $\mathbf{c}$ correspond to summing up the diagonals of $\mathbf{R}^{-1}$, the vector $\mathbf{c}$ may be obtained from $\mathbf{u}$ as

$$\begin{pmatrix} c_{-N+1} \\ \vdots \\ c_{-1} \\ c_0 \end{pmatrix} = \begin{pmatrix} u_{N-1} & 0 & \cdots & 0 \\ 2u_{N-2} & u_{N-1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ Nu_0 & \cdots & 2u_{N-2} & u_{N-1} \end{pmatrix} \begin{pmatrix} \bar{u}_0 \\ \bar{u}_1 \\ \vdots \\ \bar{u}_{N-1} \end{pmatrix}$$
$$- \begin{pmatrix} \bar{u}_1 & \cdots & \cdots & 0 \\ \vdots & \ddots & & \vdots \\ (N-1)\bar{u}_{N-1} & \ddots & \ddots & \vdots \\ 0 & (N-1)\bar{u}_{N-1} & \cdots & \bar{u}_1 \end{pmatrix} \begin{pmatrix} 0 \\ u_{N-1} \\ \vdots \\ u_1 \end{pmatrix},$$

which is a set of Toeplitz matrix- vector products, and hence may be computed in $\mathcal{O}(N \log N)$ using FFT.

TABLE I
FAST IAA

Step 0 - Initialize: Calculate $\mathbf{x}^{(0)} = \mathcal{F}(\mathbf{y})_K$
Step 1: Calculate $\mathbf{R}$ from $(\mathcal{F}^{-1}(\mathbf{p})_K)_{1:N}$, where $p_k^{(i)} = |x_k^{(i)}|^2$
Step 2: Calculate GS Factorization of $\mathbf{R}^{-1}$ via LD
Step 3: Calculate $\mathbf{R}^{-1}\mathbf{y}$ using Toeplitz matrix-vector calculations
Step 4: Calculate $\phi_{1-\mathrm{N}} = \mathcal{F}(\mathbf{R}^{-1}\mathbf{y})_K$
Step 5: Calculate $\phi_{1-\mathrm{D}} = K\mathcal{F}^{-1}(\mathbf{c})_K$ obtained from $\mathbf{u}$
Step 6: Calculate $\mathbf{x} = \phi_{1-\mathrm{N}}/\phi_{1-\mathrm{D}}$ (elementwise division)
Step 7: Repeat Steps $1-6$ until a pre-specified number of iterations is reached or a pre-specified threshold is satisfied.

4) Summary: The steps for Fast IAA are summerized in Table I. The overall computational cost is $\mathcal{O}(N^2 + K \log K)$. This methodology for computing fast IAA was independently developed in [4] and [5], and they are based on earlier fast implementations of APES [18].

IV. MISSING DATA IAA AND FAST CALCULATIONS

In [3], IAA was applied to problems where data samples are missing.[2] In this section we treat the spectral estimation part of missing data IAA and provide a new algorithm that is fast when the number of missing data samples is small. In the

[2] This case may be seen as a problem with nonuniform data sampling, hence IAA still applies.

next section we utilize these results for recovering the missing part of $\mathbf{y}$.

Consider the problem of estimating $\mathbf{x}$ from a vector of available data $\mathbf{y}_{\mathrm{g}}$, which is a subset of the full data vector $\mathbf{y}$. The available and missing part of $\mathbf{y}$ may be represented as

$$\mathbf{y}_{\mathrm{g}} = \mathbf{S}_{\mathrm{g}}\mathbf{y}, \quad \mathbf{y}_{\mathrm{m}} = \mathbf{S}_{\mathrm{m}}\mathbf{y},$$

where $\mathbf{S}_{\mathrm{g}} \in \mathbb{R}^{N_{\mathrm{g}} \times N}$ and $\mathbf{S}_{\mathrm{m}} \in \mathbb{R}^{N_{\mathrm{m}} \times N}$ are the selection matrices corresponding to the available and missing samples, respectively. Here N_{g} and N_{m} denote the number of available data and missing data, and hence $N = N_{\mathrm{g}} + N_{\mathrm{m}}$.

The data model (2) is now replaced by

$$\mathbf{y}_{\mathrm{g}} = \mathbf{S}_{\mathrm{g}}\mathbf{y} = \mathbf{S}_{\mathrm{g}}\mathbf{A}\mathbf{x} + \mathbf{S}_{\mathrm{g}}\mathbf{e} \tag{12}$$

where $\mathbf{S}_g\mathbf{A}$ is the steering matrix and the vector $\mathbf{x}$ is sought. Since IAA is applicable for the missing data case, each iteration of MIAA is now to evaluate

$$x_k = \frac{\mathbf{a}_{\mathrm{g}}(\omega_k)^* {\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{y}_{\mathrm{g}}}{\mathbf{a}_{\mathrm{g}}(\omega_k)^* {\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{a}_{\mathrm{g}}(\omega_k)}, \tag{13}$$

where

$$\begin{aligned} \mathbf{a}_g(\omega_k) &= \mathbf{S}_{\mathrm{g}}\mathbf{a}(\omega_k), \\ \mathbf{R}_{\mathrm{g}} &= \mathbf{S}_{\mathrm{g}}\mathbf{A}\mathbf{P}\mathbf{A}^*{\mathbf{S}_{\mathrm{g}}}^T = \mathbf{S}_{\mathrm{g}}\mathbf{R}{\mathbf{S}_{\mathrm{g}}}^T, \end{aligned} \tag{14}$$

and as before $\mathbf{P} = \mathrm{diag}(p_k)$ where $p_k = |x_k|^2$, for $k = 0, \ldots, K-1$. Denote the numerator and denominator of (13) by

$$\begin{aligned} \Psi_{\mathrm{N}}(\omega_k) &= \mathbf{a}_{\mathrm{g}}(\omega_k)^* {\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{y}_{\mathrm{g}}, \\ \Psi_{\mathrm{D}}(\omega_k) &= \mathbf{a}_{\mathrm{g}}(\omega_k)^* {\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{a}_{\mathrm{g}}(\omega_k). \end{aligned}$$

For the full data case, where $\mathbf{S}_{\mathrm{g}} = \mathbf{I}_N$, these polynomials would be identical to Φ_{N} and Φ_{D} and may be calculated efficiently as in Section III. In the missing data case, however, $\mathbf{R}_{\mathrm{g}}$ is not Toeplitz and several of the steps in the previous section required for fast computations breaks down, including the GS factorization. A brute force solution would include a matrix inversion of $\mathbf{R}_{\mathrm{g}}$ and evaluations of ${\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{a}_{\mathrm{g}}(\omega_k)$ for all k, which have computational complexity $\mathcal{O}({N_{\mathrm{g}}}^2 K)$. A fast approach was proposed reducing the number of operations to $\mathcal{O}({N_{\mathrm{g}}}^3 + K \log K)$ [5]. The main burden here is the inversion of the matrix $\mathbf{R}_g$. If the number of available samples N_{g} is small, then this is not a problem. However, if the number of available samples N_{g} is large, then this inversion will be the bottleneck.

Next we consider the case where the number of missing data points is small compared to the total set of data. In this case inversion of $\mathbf{R}_{\mathrm{g}}$ is the bottleneck and could even be infeasible. Instead, we will show that a low rank completion can be used to transform the problem to the full data case. Using this, Ψ_{N} and Ψ_{D} in MIAA may be calculated by adjusting Φ_{N} and Φ_{D} from the full data case by terms with low rank structures.

A. Fast calculation of missing data IAA

Consider the case where the number of missing samples N_{m} is small compared to all samples N. The problem is how to utilize the structure of $\mathbf{R}_{\mathrm{g}}$ and $\mathbf{S}_{\mathrm{g}}$ for evaluating the trigonometric polynomials $\Psi_{\mathrm{N}}(\omega)$ and $\Psi_{\mathrm{D}}(\omega)$. The calculations rely on the following key proposition, which allows us to express the matrix product ${\mathbf{S}_{\mathrm{g}}}^T{\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{S}_{\mathrm{g}}$ as a sum of a low rank matrix and the inverse of a Toeplitz matrix.

Proposition 1: Let $\mathbf{R} > 0$ and $\mathbf{R}_{\mathrm{g}}$ be defined by (14) where $\mathbf{A}$ is the steering matrix defined in (1). Then

$${\mathbf{S}_{\mathrm{g}}}^T{\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{S}_{\mathrm{g}} = \mathbf{R}^{-1} - \mathbf{\Gamma}$$

where $\mathbf{\Gamma}$ is given by

$$\mathbf{\Gamma} := \mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T(\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T)^{-1}\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}.$$

Proof: Let $\mathbf{S}$ denote the permutation matrix partitioning all samples into the available and the missing samples

$$\mathbf{S} = \begin{pmatrix} \mathbf{S}_{\mathrm{g}} \\ \mathbf{S}_{\mathrm{m}} \end{pmatrix} \tag{15}$$

and note that the selection matrices satisfy

$$\begin{aligned} \mathbf{S}^T\mathbf{S} &= {\mathbf{S}_{\mathrm{g}}}^T\mathbf{S}_{\mathrm{g}} + {\mathbf{S}_{\mathrm{m}}}^T\mathbf{S}_{\mathrm{m}} = \mathbf{I}_N, & (16) \\ \mathbf{S}_{\mathrm{g}}\mathbf{S}^T &= (\mathbf{S}_{\mathrm{g}}{\mathbf{S}_{\mathrm{g}}}^T, \mathbf{S}_{\mathrm{g}}{\mathbf{S}_{\mathrm{m}}}^T) = (\mathbf{I}_{N_{\mathrm{g}}}, \mathbf{0}_{N_{\mathrm{g}} \times N_{\mathrm{m}}}). & (17) \end{aligned}$$

From (16), (17) and then by inserting (15) it follows that

$$\begin{aligned} (\mathbf{I}_{N_{\mathrm{g}}}, \mathbf{0}_{N_{\mathrm{g}} \times N_{\mathrm{m}}}) &= (\mathbf{S}_{\mathrm{g}}\mathbf{R}\mathbf{S}^T)(\mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T) \qquad (18) \\ &= (\mathbf{R}_{\mathrm{g}}, \mathbf{R}_{\mathrm{gm}}) \begin{pmatrix} \mathbf{S}_{\mathrm{g}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{g}}}^T & \mathbf{S}_{\mathrm{g}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T \\ \mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{g}}}^T & \mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T \end{pmatrix}, \end{aligned}$$

where $\mathbf{R}_{\mathrm{gm}} := \mathbf{S}_{\mathrm{g}}\mathbf{R}{\mathbf{S}_{\mathrm{m}}}^T$. By first using the first block and then the second block of (18), we get[3]

$$\begin{aligned} {\mathbf{R}_{\mathrm{g}}}^{-1} &= \mathbf{S}_{\mathrm{g}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{g}}}^T + {\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{R}_{\mathrm{gm}}\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{g}}}^T \\ &= \mathbf{S}_{\mathrm{g}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{g}}}^T - \mathbf{S}_{\mathrm{g}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T(\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T)^{-1}\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{g}}}^T \\ &= \mathbf{S}_{\mathrm{g}}(\mathbf{R}^{-1} - \mathbf{\Gamma}){\mathbf{S}_{\mathrm{g}}}^T. \end{aligned}$$

Finally note that

$$\begin{aligned} \mathbf{\Gamma}{\mathbf{S}_{\mathrm{m}}}^T &= \mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T(\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T)^{-1}\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T \\ &= \mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^T, \end{aligned}$$

and hence

$$\begin{aligned} {\mathbf{S}_{\mathrm{g}}}^T{\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{S}_{\mathrm{g}} &= {\mathbf{S}_{\mathrm{g}}}^T\mathbf{S}_{\mathrm{g}}(\mathbf{R}^{-1} - \mathbf{\Gamma}){\mathbf{S}_{\mathrm{g}}}^T\mathbf{S}_{\mathrm{g}} \\ &= (\mathbf{I}_N - {\mathbf{S}_{\mathrm{m}}}^T\mathbf{S}_{\mathrm{m}})(\mathbf{R}^{-1} - \mathbf{\Gamma})(\mathbf{I}_N - {\mathbf{S}_{\mathrm{m}}}^T\mathbf{S}_{\mathrm{m}}) \\ &= \mathbf{R}^{-1} - \mathbf{\Gamma}. \end{aligned} \tag{19}$$

■

Denote by $\mathbf{y}_{\mathrm{f}} = {\mathbf{S}_{\mathrm{g}}}^T\mathbf{y}_{\mathrm{g}}$, the vector $\mathbf{y}$ with the missing samples zeroed out. Using Proposition 1, we see that Ψ_{N} and Ψ_{D} may be written as

$$\begin{aligned} \Psi_{\mathrm{N}}(\omega) &= \mathbf{a}_{\mathrm{g}}(\omega)^*{\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{y}_{\mathrm{g}} \\ &= \mathbf{a}(\omega)^*{\mathbf{S}_{\mathrm{g}}}^T{\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{S}_{\mathrm{g}}\mathbf{y}_{\mathrm{f}} \\ &= \mathbf{a}(\omega)^*(\mathbf{R}^{-1} - \mathbf{\Gamma})\mathbf{y}_{\mathrm{f}} \\ &= \Phi_{\mathrm{N}}(\omega) - \mathbf{a}(\omega)^*\mathbf{\Gamma}\mathbf{y}_{\mathrm{f}} \end{aligned} \tag{20}$$

[3] Alternatively, use (0.12) in [19] with $\mathbf{S}\mathbf{R}^{-1}\mathbf{S}^T$ in the place of G, in which case ${\mathbf{R}_{\mathrm{g}}}^{-1}$ corresponds to $A^{\times}$ on page 4 in [19].

and

$$\begin{aligned}\Psi_{\mathrm{D}}(\omega) &= \mathbf{a}_{\mathrm{g}}(\omega)^{*}{\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{a}_{\mathrm{g}}(\omega)\\ &= \mathbf{a}(\omega)^{*}{\mathbf{S}_{\mathrm{g}}}^{T}{\mathbf{R}_{\mathrm{g}}}^{-1}\mathbf{S}_{\mathrm{g}}\mathbf{a}(\omega)\\ &= \mathbf{a}(\omega)^{*}(\mathbf{R}^{-1}-\mathbf{\Gamma})\mathbf{a}(\omega)\\ &= \Phi_{\mathrm{D}}(\omega)-\mathbf{a}(\omega)^{*}\mathbf{\Gamma}\mathbf{a}(\omega). \end{aligned}\tag{21}$$

Here Φ_{N} and Φ_{D} may be evaluated efficiently as in Section III where $\mathbf{y}_{\mathrm{f}}$ replaces $\mathbf{y}$, using the Toeplitz structure of $\mathbf{R}$. The rank of $\mathbf{\Gamma}$ is equal to N_{m}, a fact which may be used for calculating the remaining parts of Ψ_{N} and Ψ_{D}. Next we will utilize this structure in order to evaluate (20) and (21) efficiently.

1) Get $\mathbf{R}$, the GS factorization of $\mathbf{R}^{-1}$, and evaluate $\Phi_{\mathrm{N}}(\omega)$ and $\Phi_{\mathrm{D}}(\omega)$: This is done exactly as in Section III where $\mathbf{y}_{\mathrm{f}}$ is used instead of $\mathbf{y}$.

2) Get $\mathbf{L} = (\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^{T})^{-1/2}$ and $\mathbf{X} = \mathbf{\Gamma}^{1/2}$: As a first step to get $\mathbf{L}$ we calculate the matrix $\mathbf{R}^{-1}$. Let $\mathbf{R}^{-1} =: [\mathbf{r}_1^{\mathrm{inv}}, \ldots, \mathbf{r}_N^{\mathrm{inv}}]$ and recursively calculate $\mathbf{r}_k^{\mathrm{inv}}$ using the displacement structure $\nabla(\mathbf{R}^{-1})$ (see Appendix B, c.f., (18)-(20) in [20])

$$\mathbf{r}_k^{\mathrm{inv}} = \begin{cases} \mathbf{u}\overline{\mathbf{u}(1)} - \tilde{\mathbf{u}}\overline{\tilde{\mathbf{u}}(1)} & k=1\\ \mathbf{D}\mathbf{r}_{k-1}^{\mathrm{inv}} + \mathbf{u}\overline{\mathbf{u}(k)} - \tilde{\mathbf{u}}\overline{\tilde{\mathbf{u}}(k)} & k>1, \end{cases}\tag{22}$$

in $\mathcal{O}(N^2)$. Then get $\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^{T}$ by selecting the rows and columns corresponding to the missing data. Let $\mathbf{L} \in \mathbb{C}^{N_{\mathrm{m}}\times N_{\mathrm{m}}}$ be the inverse of the Cholesky factor of $\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^{T}$ (calculated in $\mathcal{O}({N_{\mathrm{m}}}^3)$), i.e.,

$$\mathbf{L}\mathbf{L}^{*} = (\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^{T})^{-1}.$$

Finally, get $\mathbf{X} \in \mathbb{C}^{N\times N_{\mathrm{m}}}$ satisfying $\mathbf{X}\mathbf{X}^{*} = \mathbf{\Gamma}$, by multiplication $\mathbf{X} = \mathbf{R}^{-1}({\mathbf{S}_{\mathrm{m}}}^{T}\mathbf{L})$ using[4] the GS factorization of $\mathbf{R}^{-1}$, in $\mathcal{O}(N_{\mathrm{m}}N\log N)$.

3) Evaluate $\mathbf{a}(\omega)^{*}\mathbf{\Gamma}\mathbf{y}_{\mathrm{f}}$: The remaining part of the numerator Ψ_N, i.e., $\mathbf{a}(\omega)^{*}\mathbf{\Gamma}\mathbf{y}_{\mathrm{f}}$, may be calculated by noting that

$$\mathbf{a}(\omega)^{*}\mathbf{\Gamma}\mathbf{y}_{\mathrm{f}} = \mathbf{a}(\omega)^{*}\mathbf{X}\mathbf{L}^{*}\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}\mathbf{y}_{\mathrm{f}}.\tag{23}$$

In this expression first the $\mathbf{R}^{-1}\mathbf{y}_{\mathrm{f}}$ multiplication is carried out using the GS-factorization, $\mathbf{S}_{\mathrm{m}}(\mathbf{R}^{-1}\mathbf{y}_{\mathrm{f}})$ by selecting the rows corresponding to the missing data, and then $\mathbf{X}\mathbf{L}^{*}(\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}\mathbf{y}_{\mathrm{f}})$ by standard matrix-vector calculations (in $\mathcal{O}(N_{\mathrm{m}}N + N\log N)$). Finally (23) is evaluated at ω_k for $k = 0, 1, \ldots, K-1$ in $\mathcal{O}(K\log K)$ by noting that

$$\begin{pmatrix} \mathbf{a}(\omega_0)^{*}\mathbf{\Gamma}\mathbf{y}_{\mathrm{f}}\\ \mathbf{a}(\omega_1)^{*}\mathbf{\Gamma}\mathbf{y}_{\mathrm{f}}\\ \cdots\\ \mathbf{a}(\omega_{K-1})^{*}\mathbf{\Gamma}\mathbf{y}_{\mathrm{f}} \end{pmatrix} = \mathcal{F}(\mathbf{X}\mathbf{L}^{*}\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}\mathbf{y}_{\mathrm{f}})_K.\tag{24}$$

4) Evaluate $\mathbf{a}(\omega)^{*}\mathbf{\Gamma}\mathbf{a}(\omega)$: The remaining part of the numerator, i.e., $\mathbf{a}(\omega)^{*}\mathbf{\Gamma}\mathbf{a}(\omega)$ may be calculated by noting that

$$\mathbf{a}(\omega)^{*}\mathbf{\Gamma}\mathbf{a}(\omega) = \mathbf{a}(\omega)^{*}\mathbf{X}\mathbf{X}^{*}\mathbf{a}(\omega) = \sum_{\ell=-N+1}^{N-1} d_\ell e^{j\ell\omega}$$

is a trigonometric polynomial. Here the coefficient d_ℓ is the ℓth diagonal of $\mathbf{X}\mathbf{X}^{*}$ ($d_\ell = \bar{d}_{-\ell}$), and may be calculated in $\mathcal{O}(N_{\mathrm{m}}N\log N)$ using Toeplitz matrices (c.f., Appendix B [21])

$$\begin{pmatrix} d_{N-1}\\ \vdots\\ d_0 \end{pmatrix} = \sum_{\ell=1}^{N_{\mathrm{m}}} \begin{pmatrix} \mathbf{X}(1,\ell) & & \\ \vdots & \ddots & \\ \mathbf{X}(N,\ell) & \cdots & \mathbf{X}(1,\ell) \end{pmatrix} \begin{pmatrix} \overline{\mathbf{X}(N,\ell)}\\ \vdots\\ \overline{\mathbf{X}(1,\ell)} \end{pmatrix}.$$

Since the grid $\{\omega_k\}_{k=0}^{K-1}$ is uniform, $\mathbf{a}(\omega_k)^{*}\mathbf{\Gamma}\mathbf{a}(\omega_k)$ may be computed using FFT from the coefficients $\{d_n\}_{n=-N+1}^{N-1}$. This may be expressed as

$$\boldsymbol{\phi}_{\mathbf{\Gamma}} = K\mathcal{F}^{-1}(\mathbf{d})\tag{25}$$

where $\mathbf{d} = [d_0, d_1, \ldots, d_{N-1}, \mathbf{0}_{K-2N+1}^{T}, d_{-N+1}, \ldots, d_{-1}]^T$ and $\boldsymbol{\phi}_{\mathbf{\Gamma}} = [\mathbf{a}(\omega_0)^{*}\mathbf{\Gamma}\mathbf{a}(\omega_0), \ldots, \mathbf{a}(\omega_{K-1})^{*}\mathbf{\Gamma}\mathbf{a}(\omega_{K-1})]^T$, which is evaluated in $\mathcal{O}(K\log K)$.

5) Get $\Psi_{\mathrm{N}}(\omega)$ and $\Psi_{\mathrm{D}}(\omega)$: Get the evaluations of $\Psi_{\mathrm{N}}(\omega)$ and $\Psi_{\mathrm{D}}(\omega)$ from (20) and (21) using (23) and (25). Note that $\Psi_{\mathrm{N}}(\omega)$ is independent of the missing samples $\mathbf{y}_{\mathrm{m}}$ since $(\mathbf{R}^{-1}-\mathbf{\Gamma}){\mathbf{S}_{\mathrm{m}}}^{T} = 0$.

TABLE II
FAST MIAA

Step 0 - Initialize: Calculate $\mathbf{x}^{(0)} = \mathcal{F}(\mathbf{y}_{\mathrm{f}})_K$
Step 1: Calculate $\mathbf{R}$ from $(\mathcal{F}^{-1}(\mathbf{p})_K)_{1:N}$, where $p_k^{(i)} = |x_k^{(i)}|^2$
Step 2: Calculate GS Factorization of $\mathbf{R}^{-1}$ via LD
Step 3: Calculate $\mathbf{R}^{-1}\mathbf{y}_{\mathrm{f}}$ using Toeplitz matrix-vector calculations
Step 4: Calculate $\phi_{1-\mathrm{N}} = \mathcal{F}(\mathbf{R}^{-1}\mathbf{y}_{\mathrm{f}})_K$
Step 5: Calculate $\phi_{1-\mathrm{D}} = K\mathcal{F}^{-1}(\mathbf{c})_K$ obtained from $\mathbf{u}$
Step 6: Calculate $\mathbf{L} = (\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^{T})^{-1/2}$ and $\mathbf{X} = \mathbf{\Gamma}^{1/2}$
Step 7: Calculate $\mathbf{a}(\omega)^{*}\mathbf{\Gamma}\mathbf{y}_{\mathrm{f}}$ from (24)
Step 8: Calculate $\mathbf{a}(\omega)^{*}\mathbf{\Gamma}\mathbf{a}(\omega)$ from (25)
Step 9: Calculate $x_k = \Psi_{\mathrm{N}}(\omega_k)/\Psi_{\mathrm{D}}(\omega_k)$
Step 10: Repeat Steps $1-9$ until a pre-specified number of iterations is reached or a pre-specified threshold is satisfied.

6) Summary: The steps for fast MIAA are shown in Table II. Each iteration is calculated in $\mathcal{O}({N_{\mathrm{m}}}^3 + N_{\mathrm{m}}N\log N + N^2 + K\log(K))$. The proposed algorithm can handle the cases where the number of missing data is small ($N_{\mathrm{m}} < N/2$). In this situation the inversion of a large $N_{\mathrm{g}} \times N_{\mathrm{g}}$ matrix is replaced by the inversion of a smaller $N_{\mathrm{m}} \times N_{\mathrm{m}}$ matrix. Compare this to the results in [5] which requires an inversion of an $N_{\mathrm{g}} \times N_{\mathrm{g}}$ matrix in each iteration and takes $\mathcal{O}({N_{\mathrm{g}}}^3 + K\log(K))$.

V. Calculating the missing data, *MIAA-t*

The spectral estimate can be utilized for estimating the missing samples. This was studied in [3], where the missing data vector of the form

$$\hat{\mathbf{y}}_{\mathrm{m}} = \mathbf{T}\mathbf{y}_{\mathrm{g}}$$

is sought which minimizes the mean squared error of $\hat{\mathbf{y}}_{\mathrm{m}} - \mathbf{y}_{\mathrm{m}}$. This is referred to as *MIAA-t*, and the minimizing $\mathbf{T}$ is (see [3])

$$\mathbf{T} = \mathbf{S}_{\mathrm{m}}\mathbf{R}\mathbf{S}_{\mathrm{g}}(\mathbf{S}_{\mathrm{g}}\mathbf{R}\mathbf{S}_{\mathrm{g}})^{-1}.$$

Also here the computational burden of calculating $(\mathbf{S}_{\mathrm{g}}\mathbf{R}\mathbf{S}_{\mathrm{g}})^{-1}\mathbf{y}_{\mathrm{g}}$ would be considerable when N_{g} is large.

[4] For small data sizes, multiplication $(\mathbf{R}^{-1}{\mathbf{S}_{\mathrm{m}}}^{T})\mathbf{L}$ is faster $\mathcal{O}(N_m^2 N)$.

However, using Proposition 1, we see that $\hat{\mathbf{y}}_{\mathrm{m}}$ can be written as

$$\begin{aligned} \hat{\mathbf{y}}_{\mathrm{m}} &= \mathbf{S}_{\mathrm{m}}\mathbf{R}\mathbf{S}_{\mathrm{g}}(\mathbf{S}_{\mathrm{g}}\mathbf{R}\mathbf{S}_{\mathrm{g}})^{-1}\mathbf{y}_{\mathrm{g}} \\ &= \mathbf{S}_{\mathrm{m}}(\mathbf{I}_N - \mathbf{R}\boldsymbol{\Gamma})\mathbf{y}_{\mathrm{f}}. \end{aligned} \tag{26}$$

Therefore the estimated data vector $\hat{\mathbf{y}}$ can be written as

$$\begin{aligned} \hat{\mathbf{y}} &= {\mathbf{S}_{\mathrm{g}}}^T\mathbf{y}_{\mathrm{g}} + {\mathbf{S}_{\mathrm{m}}}^T\hat{\mathbf{y}}_{\mathrm{m}} \\ &= (\mathbf{I}_N - {\mathbf{S}_{\mathrm{m}}}^T\mathbf{S}_{\mathrm{m}})\mathbf{y}_{\mathrm{f}} + {\mathbf{S}_{\mathrm{m}}}^T\mathbf{S}_{\mathrm{m}}(\mathbf{I}_N - \mathbf{R}\boldsymbol{\Gamma})\mathbf{y}_{\mathrm{f}} \\ &= \mathbf{y}_{\mathrm{f}} - {\mathbf{S}_{\mathrm{m}}}^T\mathbf{S}_{\mathrm{m}}\mathbf{R}\boldsymbol{\Gamma}\mathbf{y}_{\mathrm{f}} \\ &= \mathbf{y}_{\mathrm{f}} - {\mathbf{S}_{\mathrm{m}}}^T\mathbf{L}\mathbf{L}^*\mathbf{S}_{\mathrm{m}}\mathbf{R}^{-1}\mathbf{y}_{\mathrm{f}}, \end{aligned}$$

which may be evaluated in $\mathcal{O}({N_{\mathrm{m}}}^2 + N\log N)$. Note that the matrix $\mathbf{I}_N - \mathbf{R}\boldsymbol{\Gamma}$ is a projection along span($\mathbf{S}_{\mathrm{m}}$) onto span($\mathbf{R}{\mathbf{S}_{\mathrm{g}}}^T$). That is, it takes any values from missing data to 0, and any vector in span($\mathbf{R}{\mathbf{S}_{\mathrm{g}}}^T$) is unaltered. Then $\hat{\mathbf{y}}_{\mathrm{m}}$ in (26) (and consequently $\hat{\mathbf{y}}$) only depend on the available data $\mathbf{y}_{\mathrm{g}}$, and hence $\mathbf{y}_{\mathrm{f}}$ could be any vector whose available data coincides with $\mathbf{y}_{\mathrm{g}}$, i.e., $\mathbf{y}_{\mathrm{g}} = \mathbf{S}_{\mathrm{g}}\mathbf{y}_{\mathrm{f}}$.

VI. Applications

A. Computational complexity

A detailed study of the proposed algorithm shows that the leading terms of the computational cost are (see Appendix A)

$$\frac{2}{3}{N_{\mathrm{m}}}^3 + 8N_{\mathrm{m}}N\log_2 N + \frac{3}{2}N^2 + \frac{3}{2}K\log(K). \tag{27}$$

This should be compared to $(2/3){N_{\mathrm{g}}}^3 + (3/2)K\log(K)$ in [5]. The asymptotic improvement in the computational complexity, as the number of samples N go to infinity while the proportion N_{m}/N is fixed, is given by

$$\left(\frac{N}{N_{\mathrm{m}}} - 1\right)^3$$

The improvement for N ranging from 500 to 8000 is depicted in Figure 1. Here $K = 8N$ is used. Note however that the choice of K is irrelevant for the plot since the last term of (27) is negligible for relevant cases ($K < 15N$).

B. 1D sinusoid identification

Consider an example of identification of $1-$D sinusoids in noise and recovery of the missing data. Let the signal y_n be

$$y_n = \sum_{\ell=1}^{6} 2\sin(n\hat{\omega}_\ell + v_\ell) + w_n, \quad n = 0, 1, \ldots. N-1, \tag{28}$$

where w_n is Gaussian white noise with variance 1, v_ℓ is a random variable with uniform distribution on $[0, 2\pi]$, and $\hat{\omega}_\ell$ denote the frequencies $(0.8, 1.2, 1.4, 1.5, 1.55, 1.575)$ of the real sinusoids. In the first example, let the number of samples be $N = 200$ and the number of frequency points be $K = 8N = 1600$. Furthermore, let 10% (= 20) of the samples be missing in two gaps consisting of the samples $[101, 110]$ and $[121, 130]$. Spectral estimates are based on the missing data IAA and the periodogram.

Figure 2 shows the data sequence (upper plot) and a close up on the region with the missing samples (lower plot). Here we can see that the recovery is quite good– actually the estimate of the missing data is closer to the original signal (without noise $= y_n - w_n$) than the signal y_n itself. Figure 3 shows the spectral estimates, where it can be seen that MIAA has considerably lower sidelobes and better resolution than the periodogram. Since the main focus in this work is on the computational complexity of MIAA, we refer to [17] and [3] for a comprehensive comparison of MIAA with other methods such as MAPES, GAPES [22], CoSaMP [23], and SLIM [24]. Instead we compare the computational complexity of the proposed implementation of MIAA with the ones proposed in [5] and [17].

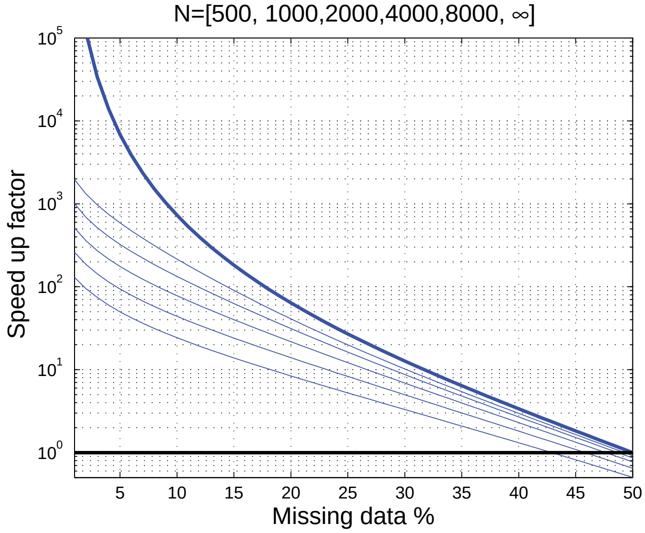


Fig. 1. The speed up (measured by comparing the number of flops) for the missing data IAA algorithm compared to inverting $\mathbf{R}_{\mathrm{g}}$, for $N = (500, 1000, 2000, 4000, 8000, \infty)$. The asymptotic speed up ($N = \infty$) is depicted with bold line.

To compare the computational complexity of the proposed implementation with the ones in [5] and [17], note that the computational bottleneck in [5] and [17] are the inversion of the matrix $\mathbf{R}_{\mathrm{g}}$. To avoid discussions regarding specifics in their implementation, we will simply compare the proposed algorithm with the inversion of the matrix $\mathbf{R}_{\mathrm{g}}$. Consider the signal $\mathbf{y}$ defined by (28) for the cases $N \in \{2000, 4000, 8000\}$ and $K = 8N$. For each of those cases we compare the average time to perform a MIAA iteration over 10 total iterations of the proposed algorithm with the average time to invert $\mathbf{R}_{\mathrm{g}}$ in each iteration.This is done for the missing data ratio going from 5% to 50%. The results are shown in Figure 4.

The times for the proposed algorithm are considerably shorter than the time required to perform the matrix inversion of $\mathbf{R}_{\mathrm{g}}$ when the proportion of missing data is low. At missing data ratio of 10%, the proposed algorithm is $9, 15$, and 30 times faster[5] than the matrix inversion of $\mathbf{R}_{\mathrm{g}}$, for the respective data sizes $2000, 4000$, and 8000. The proposed algorithm is

[5] Note that these running times based on Matlab differ from the theoretical speed up from Subsection VI-A. This is because running times in Matlab depend heavily on programming details. For example, many intrinsic Matlab functions are very efficient and are capable of taking advantage of multi-core processors. More involved algorithms implemented using Matlab scripts, on the other hand, are not always capable of making use of all the cores available.

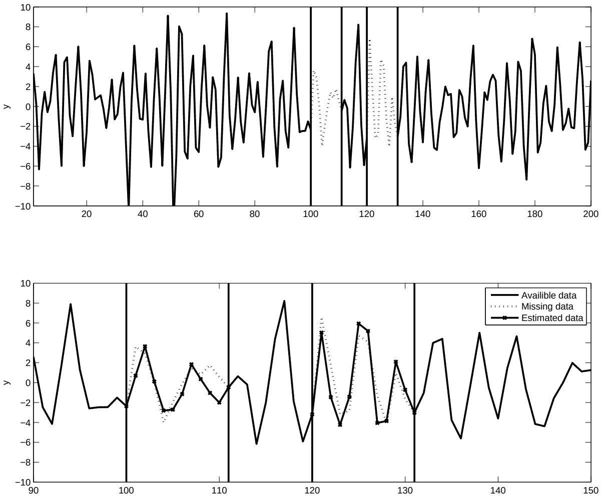


Fig. 2. The upper graph shows the full time series $\mathbf{y}$. The lower graph is an enlargement on the part where data is missing. Here the punctuated line shows the estimate $\hat{\mathbf{y}}$.

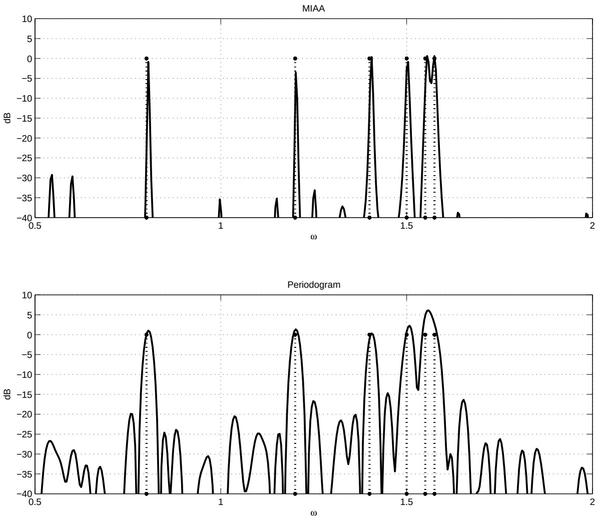


Fig. 3. The upper graph shows the missing data IAA spectral estimate. The lower graph shows the periodogram based on the available data.

about as fast as the matrix inversion when the proportion of missing data is around $40\% - 47\%$, depending on N. This is consistent with the claim that the proposed algorithm is asymptotically faster than the state-of-the-art [5] and [17] when the proportion of missing data is less than 50%.

C. Application: Sparse 2D SAR imaging

SAR imaging comes down to a 2D spectral estimation problem in the fast time and slow time domains after all of the preliminary steps [25]. If a SAR platform is working in a congested spectrum such as the UHF or VHF bands, radio frequency interference (RFI) comes from relatively narrowband sources such as FM radio (commercial and amateur) and television broadcasts [26]. When RFI interference is suppressed by frequency notching, missing data is introduced into the fast time phase history data resulting in increased sidelobe energy [27].

We use the GOTCHA volumetric SAR data in this example from the U.S. Airforce Sensor Data Managament System (SDMS)[6]. The flight pattern was a circular SAR imaging pattern, and the sensor data available had a bandwidth of approximately 600 MHz.[7] We simulate frequency notching, to prevent interference from narrowband sources, by removing three 20 MHz bands (approximately 10% of total bandwidth) from each fast time pulse. This results in three annuli of data to be missing from the polar annulus (c.f., [28]). This represents a scenario where the transmit signal is a chirp and the reference signal is a notched chirp.

The images are formed using standard SAR techniques. The data is reformatted using polar reformatting techniques, and an image is formed (via FFT or SLIM-1) using the data from four degrees of azimuth data across all 360 degrees with no overlapping look angles. The images are then rotated into alignment and each image is fused together. The fusion is done by taking the maximum of each pixel from the set of all the images created. The first subfigure is formed from the full data and is used as a benchmark. The second image is formed from frequency notched data, i.e., the missing data is zeroed out. The third image is formed from frequency notched data where the missing data is estimated for each fast-time pulse.

In the simulations where MIAA was used to recover missing data, it was applied to every fast time pulse (in the slant range) before polar reformatting.[8] For each four degree aperture (90 total apertures), this results in 469 fast time pulses and a MIAA problem of size $N = 424$ and $K = 900$. For this data size, the computational time is approximately half compared to the state of the art, resulting in a considerable reduction in the image generation time.

In the first set of images shown in Figure 5 the images are formed by applying 2D FFTs to the data on the rectangular grid. The full data set produces a baseline image in Figure 5(a). The missing data case occurs when frequency notching is utilized and the sidelobes in the image increase as shown in Figure 5(b). When MIAA is applied to the data before the imaging, the reconstruction is nearly as good as the full data set as demonstrated in Figure 5(c) compared to Figure 5(a).

Next, we illustrate the effectiveness of MIAA when applied as a preprocessing step to a sparsity promoting imaging method. In particular, we use MIAA to estimate missing frequency samples, and the sparsity promoting method to reconstruct an underlying SAR image. The sparsity promoting imaging method used in this example was Sparse Learning via Iterative Minimization with ℓ_1 norm (SLIM-1) [24]. SLIM-1 is a maximum *a posteriori* method that is based off a Bayesian

[6] This data is publicly available by request.

[7] The GOTCHA data is X-band, but the frequency notched simulated scenario could equally well represent low frequency interference for UHF and VHF data.

[8] We do not consider any joint 2-D estimation of missing data. In a 2-D data recovery scenario, the missing data should be estimated on the rectangular grid rather than the polar grid since the sinusoids should decouple between the spatial domains.

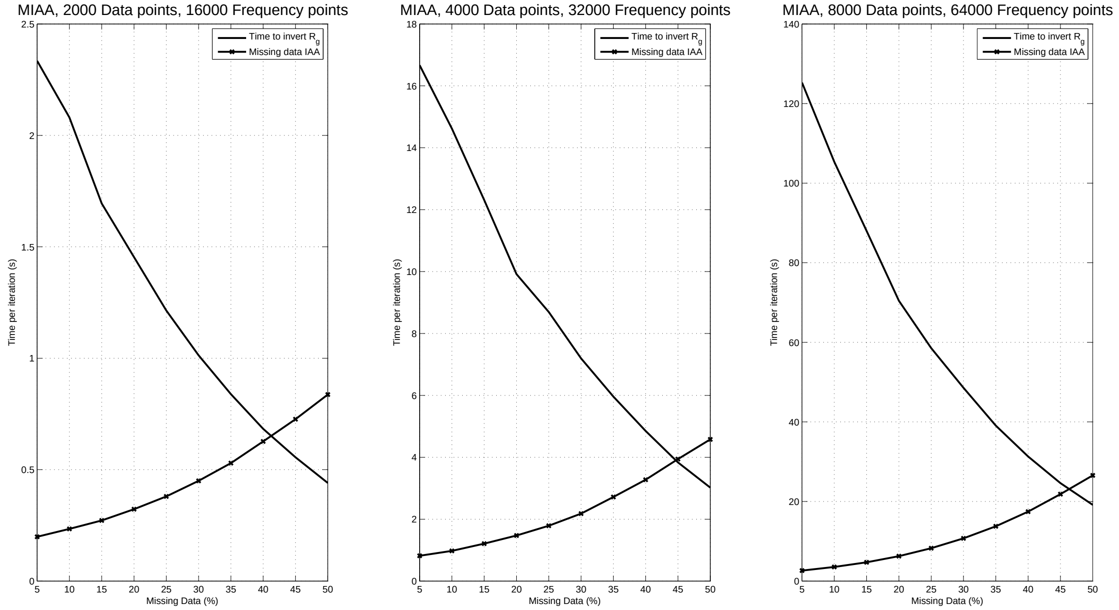


Fig. 4. Average time per iteration for running the missing data IAA compared to inverting $\mathbf{R}_\mathrm{g}$. N denotes the total number of data points and K represents the number of frequency points. Left: $N = 2000, K = 16000$. Middle: $N = 4000, K = 32000$. Right: $N = 8000, K = 64000$.

hierarcheal model. It has been shown to be an effective method for SAR imaging in full data [24] and missing data scenarios [17]. The SLIM imaging algorithm here is the 2D extension [24] where the missing data is zeroed out (not the missing data SLIM as shown in [17]).

Figure 6(a) shows the output from SLIM-1 when full data is available. The speckle noise and any sidelobes are significantly suppressed compared to Figure 5(a). The use of frequency notching to the bad bands reduces the sparse methods effectiveness at suppressing sidelobes as shown in Figure 6(b). When the missing data was estimated with MIAA before utilizing the SLIM-1 imaging algorithm, this results in the image shown in Figure 6(c) which is nearly identical to the original full data image. This example was not designed to compare different missing data algorithms, but to illustrate that reconstruction using sparsity promoting methods can be improved by first estimating missing data, e.g., by MIAA.

These empirical results show that using MIAA to estimate missing data in notched bands can be quite effective. The fast implementation allows for this method to be more practical for real world applications such as SAR imaging. In this application we have shown that fast MIAA may be utilized successfully to improve image quality in the case where frequency notching has been used to suppress interference.

VII. Conclusions

In this paper a new approach for fast missing data IAA is designed for cases where the amount of missing data is small. This method utilizes structures in the MIAA algorithm and replaces a nonstructured problem with a structured problem by a low rank completion. This allows for reducing the computational complexity from $\mathcal{O}((N - N_\mathrm{m})^3 + K\log K)$ (see [5], [17]) to $\mathcal{O}({N_\mathrm{m}}^3 + N_\mathrm{m} N \log N + N^2 + K\log(K))$. This is an improvement of the asymptotic computational complexity of $(N/N_\mathrm{m} - 1)^3$, which is an increase in performance whenever the proportion of missing data is less than 50%. We have also numerically observed that the improvement in performance is significant compared to the state-of-the art.

The method is applied to SAR imaging with frequency notching for interference suppression. Frequency notching causes missing data resulting in increased sidelobes which decreases image quality. We demonstrated that the use of missing data IAA can improve the image quality significantly.

Appendix

A. Computational complexity of MIAA algorithm

The computational cost for one iteration of the proposed algorithm is

$$\frac{2}{3}{N_\mathrm{m}}^3 + 8N_\mathrm{m}N\log_2 N + \frac{3}{2}N^2 + \frac{3}{2}K\log(K)$$

in addition to lower order terms. The first term comes from calculating $\mathbf{L}$, the inverse of the Cholesky factor of $\mathbf{S}_\mathrm{m}\mathbf{R}^{-1}{\mathbf{S}_\mathrm{m}}^T$. The second term is for calculating $\mathbf{X} = \mathbf{R}^{-1}({\mathbf{S}_\mathrm{m}}^T\mathbf{L})$ which takes $6N_\mathrm{m}$ FFT's of size $2N$ and for calculating $\mathbf{d}$ from $\mathbf{X}$ which takes $2N_\mathrm{m}$ FFT's of size $2N$. The third term results from Levinson-Durbin (N^2, see [1]) and the computation of $\mathbf{R}^{-1}$ from (22) ($N^2/2$, also utilizing that R^{-1} is Hermitian and persymmetric[9]). The final terms is $(3/2)K\log K$ and results from the three FFT/IFFT calculations of size K: 1)

[9] In this case (22) reduces to the Trenchs algorithm for the estimation of the inverse of Toeplitz matrices [29, p. 132], noting however that Trenchs method requires $\mathcal{O}(N^2)$ memory storage while (21) only $\mathcal{O}(N)$, at the expense of $(3/2)N^2$ extra computations.

$\mathbf{R}_{1:N,1} = K(\mathcal{F}^{-1}(p))_{1:N}$, 2) Ψ_{N} by synchronizing (10) and (24), 3) Ψ_{D} by synchronizing (10) and (25).

B. Toeplitz matrices and fast calculations

The celebrated fast Fourier transform has been widely used for decreasing the computational complexity of algorithms with structure. In particular, this can be used when the matrices involved are cyclic or Toeplitz. Here we summarize some of the concepts relevant to this paper. For a more comprehensive review on this, see [30].

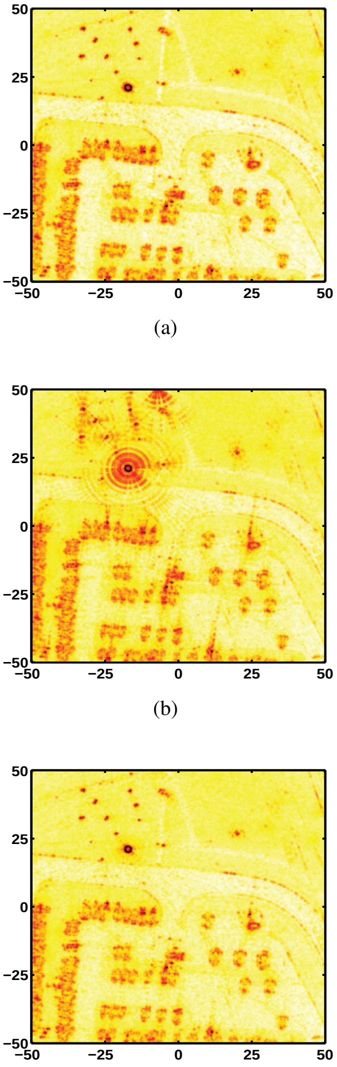
Multiplication between a Toeplitz matrix $\mathbf{R} \in \mathbb{C}^{N \times N}$ and any vector $\mathbf{y} \in \mathbb{C}^{N}$ may be computed by embedding the Toeplitz matrix into a circulant matrix

$$\begin{pmatrix} \mathbf{R} & \mathbf{U} \\ \mathbf{U} & \mathbf{R} \end{pmatrix}.$$

Since the Fourier matrix diagonalizes any cyclic matrix, $\mathbf{Ry}$ may be obtained from

$$\begin{pmatrix} \mathbf{R} & \mathbf{U} \\ \mathbf{U} & \mathbf{R} \end{pmatrix} \begin{pmatrix} \mathbf{y} \\ \mathbf{0}_{N\times 1} \end{pmatrix} = \begin{pmatrix} \mathbf{Ry} \\ \mathbf{Uy} \end{pmatrix},$$

which is computed efficiently using FFT in $\mathcal{O}(N \log N)$ [30].

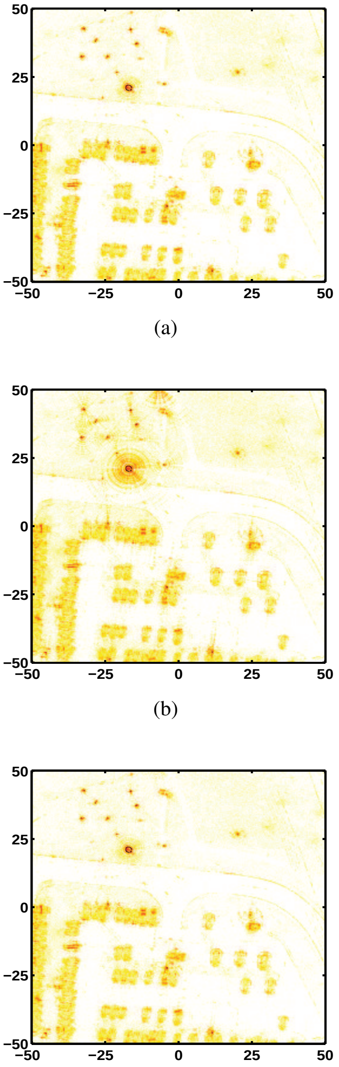


(a)

(b)

(c)

Fig. 5. FFT Imaging: (a) The image formed with full data. (b) The image formed with notched (missing) data. (c) The image formed when the notched data is estimated via MIAA.



(a)

(b)

(c)

Fig. 6. SLIM1 Imaging: (a) The image formed with full data. (b) The image formed with notched (missing) data. (c) The image formed when the notched data is estimated via MIAA.

Toeplitz systems may be solved efficiently using the displacement structures [31]. More specifically, let

$$\mathbf{D} = \begin{pmatrix} \mathbf{0}_{1\times(N-1)} & 0 \\ \mathbf{I}_{N-1} & \mathbf{0}_{(N-1)\times 1} \end{pmatrix}$$

denote the shift matrix and define the displacement of a Hermitean matrix $\mathbf{R}$ as $\nabla(\mathbf{R}) = \mathbf{R} - \mathbf{DRD}^T$. If $\mathbf{R}$ is a nonsingular Toeplitz matrix, then the displacements of $\mathbf{R}$ and its inverse $\mathbf{R}^{-1}$ both have rank bounded by 2. The displacement of $\mathbf{R}^{-1}$ may be expressed as

$$\nabla(\mathbf{R}^{-1}) = \mathbf{uu}^* - \tilde{\mathbf{u}}\tilde{\mathbf{u}}^*, \tag{29}$$

where

$$\mathbf{u} = \frac{1}{\sigma}\begin{pmatrix} 1 \\ \boldsymbol{\theta} \end{pmatrix}, \quad \tilde{\mathbf{u}} = \frac{1}{\sigma}\begin{pmatrix} 0 \\ \tilde{\boldsymbol{\theta}} \end{pmatrix},$$

$\boldsymbol{\theta} = (\theta_1, \ldots, \theta_{N-1})^T$ are the vector of AR coefficients consistent with $\mathbf{R}$, and σ^2 is the corresponding prediction error, i.e.,

$$\mathbf{R}\begin{pmatrix} 1 \\ \boldsymbol{\theta} \end{pmatrix} = \begin{pmatrix} \sigma^2 \\ \mathbf{0} \end{pmatrix},$$

and $\tilde{\boldsymbol{\theta}} = (\bar{\theta}_{N-1}, \ldots, \bar{\theta}_1)^T$ (see, e.g., [1]). Levinson-Durbin's algorithm may be used to find $\boldsymbol{\theta}$ and σ, and consequently $\mathbf{u}$ and $\tilde{\mathbf{u}}$, in $\mathcal{O}(N^2)$ [1], [29]. This may be used for solving Toeplitz systems efficiently using the so called Gohberg-Semencul (GS) factorization

$$\mathbf{R}^{-1} = \mathcal{L}(\mathbf{u}, \mathbf{D})\mathcal{L}(\mathbf{u}, \mathbf{D})^* - \mathcal{L}(\tilde{\mathbf{u}}, \mathbf{D})\mathcal{L}(\tilde{\mathbf{u}}, \mathbf{D})^*,$$

where $\mathcal{L}(\mathbf{u}, \mathbf{D}) = (\mathbf{u}, \mathbf{D}\mathbf{u}, \mathbf{D}^2\mathbf{u}, \ldots \mathbf{D}^{N-1}\mathbf{u})$ is a lower triangular Toeplitz matrix. Since $\mathcal{L}(\mathbf{u}, \mathbf{D})$ and $\mathcal{L}(\tilde{\mathbf{u}}, \mathbf{D})$ are Toeplitz matrizes, any number of systems $\mathbf{R}\mathbf{x}_j = \mathbf{y}_j$ may be solved by four Toeplitz-matrix vector multiplications.

References

[1] P. Stoica and R. L. Moses, *Spectral Analysis of Signals*. Upper Saddle River, NJ: Prentice-Hall, 2005.

[2] T. Yardibi, J. Li, P. Stoica, M. Xue, and A. B. Baggeroer, "Source localization and sensing: A nonparametric iterative adaptive approach based on weighted least squares," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 46, pp. 425 – 443, Jan. 2010.

[3] P. Stoica, J. Li, and J. Ling, "Missing data recovery via a nonparametric iterative adaptive approach," *IEEE Signal Processing Letters*, vol. 16, pp. 241–244, April 2009.

[4] M. Xue, L. Xu, and J. Li, "IAA spectral estimation: Fast implementation using the Gohberg-Semencul factorization," *IEEE Transactions on Signal Processing*, vol. 59, pp. 3251–3261, July 2011.

[5] G.-O. Glentis and A. Jakobsson, "Efficient implementation of iterative adaptive approach spectral estimation techniques," *IEEE Transactions on Signal Processing*, vol. 59, pp. 4154 –4167, September 2011.

[6] H. Hellsten and L. Ulander, "VHF/UHF synthetic aperture radar - principles and motivation," in *Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2003*, pp. 318 – 327, Dec. 2003.

[7] M. Cetin and R. Moses, "SAR imaging from partial-aperture data with frequency-band omissions," in *Proc. SPIE*, 2005.

[8] J. Capon, "High resolution frequency-wavenumber spectrum analysis," *Proceedings of the IEEE*, vol. 57, pp. 1408–1418, August 1969.

[9] R. T. Lacoss, "Data adaptive spectral analysis methods," *Geophysics*, vol. 36, pp. 661–675, August 1971.

[10] J. Li and P. Stoica, "An adaptive filtering approach to spectral estimation and sar imaging," *IEEE Transactions on Signal Processing*, vol. 44, pp. 1469–1484, June 2009.

[11] P. Stoica, H. Li, and J. Li, "A new derivation of the APES filter," *IEEE Signal Processing Letters*, vol. 6, pp. 205–206, August 1999.

[12] A. Jakobsson, G. Glentis, and E. Gudmundson, "Computationally efficient time-recursive IAA-based blood velocity estimation," *IEEE Transactions on Signal Processing*, vol. 60, pp. 3853–3858, July 2012.

[13] O. Ojowu, J. Karlsson, J. Li, and Y. Liu, "ENF extraction from digital recordings using adaptive techniques and frequency tracking," *IEEE Transactions on Information Forensics and Security*, vol. 7, pp. 1330–1338, August 2012.

[14] P. Stoica, J. Li, and H. He, "Spectral analysis of non-uniformly sampled data: A new approach versus the periodogram," *IEEE Transactions on Signal Processing*, vol. 57, pp. 843–858, March 2009.

[15] W. Roberts, P. Stoica, J. Li, T. Yardibi, and F. Sadjadi, "Super resolution SAR image formation via parametric spectral estimation methods," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 35, pp. 267–281, January 1999.

[16] W. Roberts, P. Stoica, J. Li, T. Yardibi, and F. A. Sadjadi, "Iterative adaptive approaches to MIMO radar imaging," *IEEE Journal of Selected Topics in Signal Processing: Special Issue on MIMO Radar and Its Applications*, vol. 4, no. 1, pp. 5–20, 2010.

[17] D. Vu, L. Xu, M. Xue, and J. Li, "Nonparametric missing sample spectral analysis and its applications to interrupted SAR," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, pp. 1–14, February 2012.

[18] G. Glentis, "A fast algorithm for APES and Capon spectral estimation," *IEEE Transactions on Signal Processing*, vol. 56, pp. 4207–4220, September 2008.

[19] H. Dym, *J contractive matrix functions, reproducing kernel Hilbert spaces and interpolation*, vol. 71. American Mathematical Society, 1989.

[20] J. Jensen, G. Glentis, M. Christiansen, A. Jakobsson, and S. Jensen, "Computationally efficient iaa-based estimation of the fundamental frequency," *Proc. 20th EUSIPCO*, August 2012.

[21] E. Larsson and P. Stoica, "Fast implementation of two-dimensional APES and Capon spectral estimators," *Multidimensional Systems and Signal Processing*, vol. 13, pp. 35–54, January 2002.

[22] Y. Wang, J. Li, P. Stoica, and J. Moura, *Spectral Analysis of Signals: The Missing Data Case*. San Rafael, CA: Morgan & Claypool, 2005.

[23] D. Needell and J. Tropp, "CoSaMP: Iterative signal recovery from incomplete and inaccurate samples," *Applied and Computational Harmonic Analysis*, vol. 26, pp. 301–321, May 2009.

[24] X. Tan, W. Roberts, J. Li, and P. Stoica, "Sparse learning via iterative minimization with application to MIMO radar imaging," *IEEE Transactions on Signal Processing*, vol. 59, pp. 1088–1101, March 2011.

[25] C. V. Jakowatz, D. E. Wahl, P. H. Eichel, D. C. Ghiglia, and P. A. Thompson, *Spotlight-mode Synthetic Aperture Radar: A Signal Processing Approach*. Springer Science and Business Media, Inc., 1996.

[26] M. E. Davis, P. G. Tomlinson, and R. P. Maloney, "Technical challenges in ultra-wideband radar development for target detection and terrain mapping," *Proc. 1999 IEEE Radar Conference*, pp. 1–6, April 1999.

[27] E. J. Taylor, *Ultra-Wideband Radar Technology*. CRC Press LLC, 2001.

[28] M. E. Davis, "Design alternatives for foliage penetration SAR ultra wideband waveforms," pp. 1233 –1238, May 2010.

[29] G. H. Golub and C. F. V. Loan, *Matrix Computations*. Baltimore, MD: Johns Hopkins University Press, 1989.

[30] R. Gray, *Toeplitz and circulant matrices: A review*. Foundations and Trends in Communications and Information Theory, 2006.

[31] I. Gohberg and V. Olshevsky, "Circulants, displacements and decompositions of matrices," *Integral Equations and Operator Theory*, vol. 15, no. 5, pp. 730–743, 1992.

Johan Karlsson (S'06–M'09) was born in Stockholm, Sweden, 1979. He received an M.Sc. degree in Engineering Physics from Royal Institute of Technology (KTH) in 2003 and a Ph.D. in Optimization and Systems Theory from KTH in 2008. From 2009 to 2011, he was with Sirius International, Stockholm. From 2011 to 2013 he was a postdoctoral associate at the Department of Computer and Electrical Engineering, University of Florida. Since March 2013 he is an assistant professor at the Department of Mathematics, KTH, Sweden.

His current research interests include spectral estimation, interpolation theory, and fast algorithms, for applications in signal processing and control theory.

William Rowe is a Ph.D. student at the University of Florida. He received his Bachelor's degree and Master's degree in electrical engineering from the University of Florida in 2009 and 2010 respectively. He is a SMART (Science, Mathematics, and Research for Transformation) fellowship winner and is sponsored through the U.S. Army. His research interests include active sensing signal processing, spectral estimation, and software-defined radios.

Luzhou Xu (S'04-M'06) received the B.Eng. and M.Sc. degrees in electrical engineering from Zhejiang University, Hangzhou, China, in 1996 and 1999, respectively, and the Ph.D. degree in electrical engineering from the University of Florida, Gainesville, in 2006.

He is currently the Chief Engineer of the Integrated Adaptive Applications (IAA), Inc, and an adjunct research associate professor with the University of Florida, Gainesville, FL. He was with Zhongxing Research and Development Institute, Shanghai, China, from 1999 to 2001, with Philips Research East Area, Shanghai, China, from 2001 to 2003, and with ArrayComm LLC, San Jose, CA, from 2006 to 2008. His research interests include statistical and array signal processing and their applications.

George-Othon Glentis (S'88-M'91) was born in Athens, Greece, on July 24, 1965. He received the B.Sc. degree in Physics in 1987 and the Ph.D. degree in Informatics in 1991, both from the University of Athens. From 1988-1991, he held a four years research fellowship from the Institute of Informatics of the National Center for Physical Science, DEMOCRITOS. From 1993-1995, he was with the Electrical Engineering Department of the University of Twente, The Netherlands, and with the Faculté des Sciencés Appliquées, Université Catholique de Louvain, Belgium, as a EU HCM research fellow. From 1996-1997 he was with the Department of Informatics, University of Athens, as a EU TMR research fellow. From 1998- 2005 he was an associate professor at the Department of Electronics, Technological Education Institute of Crete, Chania, Greece. During the summer 2011 he has been with the Spectral Analysis Laboratory, University of Florida, as a visiting associate professor. He is currently an associate professor at the Department of Science and Technology of Telecommunications, University of Peloponnese, Tripolis, Greece. His research interests include signal and image processing, and telecommunication applications. He has published over 60 papers in major journals and conferences.

Jian Li (S'87-M'91-SM'97-F'05) received the M.Sc. and Ph.D. degrees in electrical engineering from The Ohio State University, Columbus, in 1987 and 1991, respectively.

From April 1991 to June 1991, she was an Adjunct Assistant Professor with the Department of Electrical Engineering, The Ohio State University, Columbus. From July 1991 to June 1993, she was an Assistant Professor with the Department of Electrical Engineering, University of Kentucky, Lexington. Since August 1993, she has been with the Department of Electrical and Computer Engineering, University of Florida, Gainesville, where she is currently a Professor. In Fall 2007, she was on sabbatical leave at MIT, Cambridge, Massachusetts. Her current research interests include spectral estimation, statistical and array signal processing, and their applications.

Dr. Li is a Fellow of IEEE and a Fellow of IET. She is a member of Sigma Xi and Phi Kappa Phi. She received the 1994 National Science Foundation Young Investigator Award and the 1996 Office of Naval Research Young Investigator Award. She was an Executive Committee Member of the 2002 International Conference on Acoustics, Speech, and Signal Processing, Orlando, Florida, May 2002. She was an Associate Editor of the IEEE Transactions on Signal Processing from 1999 to 2005, an Associate Editor of the IEEE Signal Processing Magazine from 2003 to 2005, and a member of the Editorial Board of Signal Processing, a publication of the European Association for Signal Processing (EURASIP), from 2005 to 2007. She was a member of the Editorial Board of the IEEE Signal Processing Magazine from 2010 to 2012. She was a member of the Editorial Board of Digital Signal Processing – A Review Journal, a publication of Elsevier, from 2006 to 2012. She is a co-author of the papers that have received the First and Second Place Best Student Paper Awards, respectively, at the 2005 and 2007 Annual Asilomar Conferences on Signals, Systems, and Computers in Pacific Grove, California. She is a co-author of the paper that has received the M. Barry Carlton Award for the best paper published in IEEE Transactions on Aerospace and Electronic Systems in 2005. She is also a co-author of the paper that has received the Lockheed Martin Best Student Paper Award at the 2009 SPIE Defense, Security, and Sensing Conference in Orlando, Florida.