



# Efficient Spectral Estimation by MUSIC and ESPRIT with Application to Sparse FFT

Daniel Potts<sup>1\*</sup>, Manfred Tasche<sup>2</sup> and Toni Volkmer<sup>1</sup>

<sup>1</sup> Faculty of Mathematics, Technische Universität Chemnitz, Chemnitz, Germany, <sup>2</sup> Institute of Mathematics, University of Rostock, Rostock, Germany

In spectral estimation, one has to determine all parameters of an exponential sum for finitely many (noisy) sampled data of this exponential sum. Frequently used methods for spectral estimation are MUSIC (MULTiple Signal Classification) and ESPRIT (Estimation of Signal Parameters via Rotational Invariance Technique). For a trigonometric polynomial of large sparsity, we present a new sparse fast Fourier transform by shifted sampling and using MUSIC resp. ESPRIT, where the ESPRIT based method has lower computational cost. Later this technique is extended to a new reconstruction of a multivariate trigonometric polynomial of large sparsity for given (noisy) values sampled on a reconstructing rank-1 lattice. Numerical experiments illustrate the high performance of these procedures.

## OPEN ACCESS

### Edited by:

Charles K. Chui,  
Stanford University and Hong Kong  
Baptist University, USA

### Reviewed by:

Ronny Bergmann,  
University Kaiserslautern, Germany  
Hyenkyun Woo,  
Korea University of Technology and  
Education, South Korea

### \*Correspondence:

Daniel Potts  
potts@mathematik.tu-chemnitz.de

### Specialty section:

This article was submitted to  
Mathematics of Computation and  
Data Science,  
a section of the journal  
Frontiers in Applied Mathematics and  
Statistics

**Received:** 04 December 2015

**Accepted:** 05 February 2016

**Published:** 29 February 2016

### Citation:

Potts D, Tasche M and Volkmer T  
(2016) Efficient Spectral Estimation by  
MUSIC and ESPRIT with Application  
to Sparse FFT.  
Front. Appl. Math. Stat. 2:1.  
doi: 10.3389/fams.2016.00001

**Keywords:** Spectral estimation, ESPRIT, MUSIC, parameter identification, exponential sum, sparse trigonometric polynomial, sparse FFT

**AMS Subject Classifications:** 65T50, 42A16, 94A12.

## 1. INTRODUCTION

The problem of spectral estimation resp. frequency analysis arises quite often in signal processing, electrical engineering, and mathematical physics (see e.g., the books [1, 2] or the survey [3]) and reads as follows:

(P1) Recover the positive integer  $M$ , the distinct frequencies  $\varphi_j \in (-\frac{1}{2}, \frac{1}{2}]$ , and the complex coefficients  $c_j \neq 0$  ( $j = 1, \dots, M$ ) in the exponential sum of sparsity  $M$

$$h(x) := \sum_{j=1}^M c_j e^{2\pi i \varphi_j x} \quad (x \in \mathbb{R}), \quad (1.1)$$

if noisy sampled data  $\tilde{h}_k := h(k) + e_k$  ( $k = 0, \dots, N - 1$ ) with  $N \geq 2M$  are given, where  $e_k \in \mathbb{C}$  are small error terms with  $|e_k| \leq \frac{1}{10} \min_j |c_j|$ .

Introducing so-called left/right signal spaces and noise spaces in Section 2, we explain the numerical solution of the problem (P1) by the MUSIC method (created by Schmidt [4], see also Manolakis et al. [1, Section 9.6.3] and the references therein) and the ESPRIT method (created by Roy and Kailath [5], see also Manolakis et al. [1, Section 9.6.5], Stoica and Moses [2, Chapter 4] and the references therein). In a new unified approach to MUSIC and ESPRIT, we show that both methods are based on singular value decomposition (SVD) of a rectangular Hankel matrix of given sampled data. For the MUSIC and ESPRIT method, it is important to choose the window length

$L$  (number of rows) of the rectangular Hankel matrix in an optimal way. Based on Theorem 2.5, where we estimate the singular values and the spectral norm condition number of the rectangular Hankel matrix of noiseless data, one can see that  $L \approx \frac{N}{2}$  is a good choice. By the right choice of  $L$ , one can detect the correct sparsity  $M$  of (1.1) and avoid the computation of spurious frequencies.

The main disadvantages of MUSIC and ESPRIT are the relatively high computational cost in the case of large sparsity  $M$ , caused mainly by the SVD. The known algorithms for MUSIC and ESPRIT have moderate computational cost only for small sparsity  $M$ . Thus, following question arises: How can one improve the MUSIC and ESPRIT methods for spectral estimation of exponential sums with large sparsity  $M$ ? In this paper, we show that this is possible by a special divide-and-conquer technique. In the numerical examples of Section 5, the cases  $M = 256$  and  $M = 1024$  are handled.

The computational cost of an algorithm is measured in the number of arithmetical operations, where all operations are counted equally. Often the computational cost of an algorithm is reduced to the leading term, i.e., all lower order terms are omitted. For a unified approach to Prony-like methods for the parameter estimation of (1.1), namely the classical Prony method, the matrix pencil method [6, 7], and the ESPRIT method [5, 8], we refer also to Potts and Tasche [9] and the references therein. For a survey of the most successful methods for the data fitting problem with linear combinations of complex exponentials, we refer to Pereyra and Scherer [10].

Section 3 is the core of this paper. Here we present a new efficient spectral estimation with low computational cost for large sparsity  $M$  and a moderate number of given samples, if one has to recover a trigonometric polynomial of large sparsity  $M$ . This means we specialize the problem (P1). Let  $S > 0$  be a large even integer. Assume that  $\varphi_j = \frac{\omega_j}{S} \in (-\frac{1}{2}, \frac{1}{2}]$ , where  $\omega_j \in (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$ . Replacing the variable  $x$  by  $Sx$  in (1.1), we consider the 1-periodic trigonometric polynomial of sparsity  $M$

$$g(x) := h(Sx) := \sum_{j=1}^M c_j e^{2\pi i \omega_j x} \quad (x \in \mathbb{R}) \quad (1.2)$$

with integer frequencies  $\omega_j \in (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$ . Consequently we investigate the following spectral estimation problem:

(P2) Recover the sparsity  $M \in \mathbb{N}$ , all integer frequencies  $\omega_j \in (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$  as well as all non-zero coefficients  $c_j \in \mathbb{C}$  of the trigonometric polynomial (1.2) for noisy sampled values  $\tilde{g}_k := g(\frac{k}{S}) + e_k = h(k) + e_k$  ( $k = 0, \dots, N - 1$ ) with  $N \geq 2M$ , where  $e_k \in \mathbb{C}$  are small error terms with  $|e_k| \leq \frac{1}{10} \min_j |c_j|$ . Often one considers the modified problem (P2\*), where the sparsity  $M$  is known.

A numerical solution of problem (P2) or (P2\*) with low computational cost is called sparse fast Fourier transform (sparse FFT). Using divide-and-conquer technique, the trigonometric polynomial (1.2) of large sparsity  $M$  is split into some trigonometric polynomials of lower sparsity and corresponding samples are determined by fast Fourier transform (FFT). Here we borrow an idea from sparse FFT in Lawlor et al. [11]

and Christlieb et al. [12] and use shifted sampling of (1.2), i.e., equidistant sampling with few equidistant shifts. Then the trigonometric polynomials of lower sparsity can be recovered by MUSIC resp. ESPRIT. The computational cost of the new sparse FFT is analyzed too.

A similar splitting technique is suggested in Lawlor et al. [11] and Christlieb et al. [12], but with a different method to detect frequencies, when aliasing between two or more frequencies occurs. The method in Lawlor et al. [11] and Christlieb et al. [12] follows an idea of Iwen [13], which is based on the Chinese Remainder Theorem, see also Ben-Or and Tiwari [14]. A different method for the sparse FFT, based on efficient filters is suggested in Hassanieh et al. [15] and Gilbert et al. [16]. We remark that there are two types of methods, deterministic (see [13]) and randomized (see [11, 15, 16]). Further related randomized methods based on compressed sensing can be found in the papers [17–19] and in the monograph [20]. Please note that the sparse FFT methods mentioned before solve the problem (P2\*), i.e., one assumes that the sparsity (or an upper bound) is known, whereas our new deterministic sparse FFT also detects the sparsity  $M$ . We remark that preliminary tests of the implementation [21, 22] of the sfft version 3 algorithm [15] suggest that this method also works if the sparsity input parameter is chosen larger than the actual sparsity of the signal. For further references on sparse FFTs, we refer to Remark 3.1.

In Section 4, we extend our method to a new reconstruction of multivariate trigonometric polynomials of large sparsity, where sampled data on a convenient rank-1 lattice are given. In Section 5, several numerical experiments with noiseless resp. noisy sampled data illustrate the high performance of the sparse FFT as proposed in Section 3. Note that in the case of successful recovery of the sparse trigonometric polynomial (1.2) all frequencies are correctly detected. For the modified sparse FFT of Section 4, numerical examples for the reconstruction of six-variate trigonometric polynomials of sparsity 256 are given too. Moreover, we compare our results with preliminary tests of the implementation [21, 22] of the sfft version 3 algorithm [15].

In summary we present a splitting method, in between the well-known methods ESPRIT, MUSIC, and FFT for the problem (P2) with the parameters in **Table 1**. For the results, see the **Tables 3, 4** in Section 3. Furthermore, we use a reconstructing rank-1 lattice in order to reconstruct multivariate trigonometric polynomials, see Section 4. Here in the case of successful recovery of a sparse multivariate trigonometric polynomial, all frequency vectors are detected without errors.

**TABLE 1 | Numbers of required samples and computational costs for ESPRIT, MUSIC, and FFT, where  $M$  denotes the sparsity of (1.1) and  $S$  is a large even integer so that all frequencies  $\omega_j$  are of the form  $\frac{\omega_j}{S}$  with  $\omega_j \in (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$ .**

Method	Samples	Computational cost
ESPRIT	$\mathcal{O}(M)$	$\mathcal{O}(M^3)$
MUSIC	$\mathcal{O}(M)$	$\mathcal{O}(M^3 + M^2S + S \log S)$
FFT	$\mathcal{O}(S)$	$\mathcal{O}(S \log S)$

## 2. RECONSTRUCTION OF EXPONENTIAL SUMS

The main difficulty is the recovery of the frequency set  $\Phi := \{\varphi_1, \dots, \varphi_M\}$  in (1.1).

We introduce the rectangular Fourier-type matrix  $F_{N,M} := (e^{2\pi i \varphi_j (k-1)})_{k,j=1}^{N,M}$ . Note that  $F_{N,M}$  coincides with the rectangular Vandermonde matrix  $V_{N,M}(\mathbf{z}) := (z_j^{k-1})_{k,j=1}^{N,M}$  with  $\mathbf{z} := (z_j)_{j=1}^M$ , where  $z_j := e^{2\pi i \varphi_j}$  ( $j = 1, \dots, M$ ) are distinct nodes on the unit circle. Then the spectral estimation problem can be formulated in following matrix-vector form

$$V_{N,M}(\mathbf{z}) \mathbf{c} = \tilde{\mathbf{h}}, \tag{2.1}$$

where  $\tilde{\mathbf{h}} := (\tilde{h}_k)_{k=0}^{N-1}$  is the vector of noisy sampled data and  $\mathbf{c} := (c_j)_{j=1}^M$  the vector of complex coefficients of (1.1).

Under the natural assumption that the nodes  $z_j$  ( $j = 1, \dots, M$ ) are well-separated on the unit circle, it can be shown that  $F_{P,M}$  has a uniformly bounded spectral norm condition number for sufficiently large integer  $P > M$ .

**Theorem 2.1** (see [23, Theorem 2]) Assume that the frequencies  $\varphi_j \in (-\frac{1}{2}, \frac{1}{2}]$  ( $j = 1, \dots, M$ ) are well-separated by the separation distance

$$q := \min_{j \neq \ell} \left( \min_{n \in \mathbb{Z}} |\varphi_j + n - \varphi_\ell| \right) > 0$$

and that  $P > \max\{M, 2\pi + \frac{1}{q}\}$ .

Then the discrete Ingham inequalities related to  $F_{P,M}$  indicate that for all  $\mathbf{x} \in \mathbb{C}^M$

$$\alpha_1(P) \|\mathbf{x}\|_2^2 \leq \|F_{P,M} \mathbf{x}\|_2^2 \leq \alpha_2(P) \|\mathbf{x}\|_2^2 \tag{2.2}$$

with  $\alpha_1(P) := P \left( \frac{2}{\pi} - \frac{2}{\pi P^2 q^2} - \frac{4}{P} \right)$  and

$$\alpha_2(P) := \begin{cases} P \left( \frac{4\sqrt{2}}{\pi} + \frac{\sqrt{2}}{\pi P^2 q^2} + \frac{3\sqrt{2}}{P} \right) & \text{for even } P, \\ (P+1) \left( \frac{4\sqrt{2}}{\pi} + \frac{\sqrt{2}}{\pi(P+1)^2 q^2} + \frac{3\sqrt{2}}{P+1} \right) & \text{for odd } P. \end{cases}$$

Furthermore, the rectangular Fourier-type matrix  $F_{P,M}$  has a uniformly bounded spectral norm condition number

$$\text{cond}_2 F_{P,M} \leq \sqrt{\frac{\alpha_2(P)}{\alpha_1(P)}}.$$

*Proof.* The assumption  $P > 2\pi + \frac{1}{q}$  is sufficient for the gap condition

$$q > \frac{1}{P} \left( 1 - \frac{2\pi}{P} \right)^{-1/2} \tag{2.3}$$

to hold. The gap condition (2.3) ensures that  $\alpha_1(P) > 0$ . For a proof of the discrete Ingham inequalities (2.2) under the gap condition (2.3) see Liao and Fannjiang [23, Theorem 2]. Let  $\lambda_1 \geq \dots \geq \lambda_M > 0$  be the ordered eigenvalues of  $F_{P,M}^* F_{P,M} \in \mathbb{C}^{M \times M}$ .

Using the Raleigh–Ritz Theorem and (2.2), we obtain that for all  $\mathbf{x} \in \mathbb{C}^M$

$$\alpha_1(P) \|\mathbf{x}\|_2^2 \leq \lambda_M \|\mathbf{x}\|_2^2 \leq \|F_{P,M} \mathbf{x}\|_2^2 \leq \lambda_1 \|\mathbf{x}\|_2^2 \leq \alpha_2(P) \|\mathbf{x}\|_2^2$$

and hence

$$0 < \alpha_1(P) \leq \lambda_M \leq \lambda_1 \leq \alpha_2(P) < \infty. \tag{2.4}$$

Thus,  $F_{P,M}^* F_{P,M}$  is positive definite and

$$\text{cond}_2 F_{P,M} = \sqrt{\frac{\lambda_1}{\lambda_M}} \leq \sqrt{\frac{\alpha_2(P)}{\alpha_1(P)}}.$$

This completes the proof.  $\square$

**Corollary 2.2** Assume that the frequencies  $\varphi_j \in (-\frac{1}{2}, \frac{1}{2}]$  ( $j = 1, \dots, M$ ) are well-separated by the separation distance  $q > 0$  and that  $P > \max\{M, 2\pi + \frac{1}{q}\}$ .

Then the discrete Ingham inequalities related to  $F_{P,M}^T$  indicate that for all  $\mathbf{y} \in \mathbb{C}^P$

$$\alpha_1(P) \|\mathbf{y}\|_2^2 \leq \|F_{P,M}^T \mathbf{y}\|_2^2 \leq \alpha_2(P) \|\mathbf{y}\|_2^2. \tag{2.5}$$

*Proof.* The matrices  $F_{P,M}$  and  $F_{P,M}^T$  possess the same singular values  $\lambda_j$  ( $j = 1, \dots, M$ ). By the Rayleigh–Ritz Theorem we obtain that

$$\lambda_M \|\mathbf{y}\|_2^2 \leq \|F_{P,M}^T \mathbf{y}\|_2^2 \leq \lambda_1 \|\mathbf{y}\|_2^2$$

for all  $\mathbf{y} \in \mathbb{C}^P$ . Applying (2.4), we obtain the discrete Ingham inequalities (2.5).  $\square$

**Remark 2.3** The Riesz stability of the exponentials  $e^{2\pi i \varphi_j x}$  ( $j = 1, \dots, M$ ) in the Hilbert space  $\ell^2(\mathbb{Z}_N)$  follows immediately from the discrete Ingham inequalities (2.2), where  $\mathbb{Z}_N := \{0, \dots, N-1\}$  denotes the sampling grid. If the assumptions of Theorem 2.1 are fulfilled for  $P = N$ , then the exponentials  $e^{2\pi i \varphi_j x}$  ( $j = 1, \dots, M$ ) are Riesz stable with respect to the discrete norm of  $\ell^2(\mathbb{Z}_N)$ , i.e.,

$$\alpha_1(N) \|\mathbf{c}\|_2^2 \leq \sum_{k=0}^{N-1} |h(k)|^2 \leq \alpha_2(N) \|\mathbf{c}\|_2^2$$

for all exponential sums (1.1) with arbitrary coefficient vectors  $\mathbf{c} = (c_j)_{j=1}^M \in \mathbb{C}^M$ . Note that the Riesz stability of these exponentials related to continuous norms was formerly discussed and applied in spectral estimation in Peter et al. [24] and Potts and Tasche [25].  $\square$

In practice, the sparsity  $M$  of the exponential sum (1.1) is often unknown. Assume that  $L \in \mathbb{N}$  is a convenient upper bound of  $M$  with  $M \leq L \leq N - M + 1$ . In applications, such an upper bound  $L$  is mostly known *a priori*. If this is not the case, then one can choose  $L \approx \frac{N}{2}$ . As mentioned in Remark 2.6, the choice  $L \approx \frac{N}{2}$  is optimal in some sense. Often the sequence  $\{\tilde{h}_0, \tilde{h}_1, \dots, \tilde{h}_{N-1}\}$

of sampled data is called a *time series of length*  $N$ . Then we form the  $L$ -trajectory matrix of this time series

$$\tilde{\mathbf{H}}_{L,N-L+1} := (\tilde{h}_{\ell+m})_{\ell,m=0}^{L-1,N-L} \quad (2.6)$$

with the *window length*  $L \in \{M, \dots, N - M + 1\}$ . Analogously, we define the  $L$ -trajectory matrix of noiseless data

$$\mathbf{H}_{L,N-L+1} := (h(\ell+m))_{\ell,m=0}^{L-1,N-L}. \quad (2.7)$$

Obviously, (2.6) and (2.7) are  $L \times (N - L + 1)$  Hankel matrices. For simplicity, we consider mainly the noiseless case, i.e.  $\tilde{h}_k = h(k)$  ( $k = 0, \dots, N - 1$ ).

The main step in the solution of the frequency analysis problem (P1) is the determination of the sparsity  $M$  and the computation of the frequencies  $\varphi_j$  or alternatively of the nodes  $z_j = e^{2\pi i \varphi_j}$  ( $j = 1, \dots, M$ ). Afterwards one can calculate the coefficient vector  $\mathbf{c} \in \mathbb{C}^M$  as least squares solution of the overdetermined linear system (2.1), i.e., the coefficient vector  $\mathbf{c}$  is the solution of the least squares problem

$$\min_{\mathbf{c} \in \mathbb{C}^M} \|\mathbf{V}_{N,M}(\mathbf{z}) \mathbf{c} - (\tilde{h}_k)_{k=0}^{N-1}\|_2.$$

We denote square matrices with only one index and refer to the well known fact that the square Vandermonde matrix  $\mathbf{V}_M(\mathbf{z})$  is invertible and the matrix  $\mathbf{V}_{N,L}(\mathbf{z})$  with  $L \in \{M, \dots, N - M + 1\}$  has full column rank. Additionally we introduce the rectangular Hankel matrices

$$\tilde{\mathbf{H}}_{L,N-L}(s) := \tilde{\mathbf{H}}_{L,N-L+1}(1:L, 1+s:N-L+s) \quad (s = 0, 1). \quad (2.8)$$

In the case of noiseless data  $\tilde{h}_k = h(k)$  ( $k = 0, \dots, N - 1$ ), the related Hankel matrices (2.8) are denoted by  $\mathbf{H}_{L,N-L}(s)$  ( $s = 0, 1$ ).

**Remark 2.4** The Hankel matrix  $\mathbf{H}_{L,N-L+1}$  of noiseless data has the rank  $M$  for each window length  $L \in \{M, \dots, N - M + 1\}$  and the related Hankel matrices  $\mathbf{H}_{L,N-L}(s)$  ( $s = 0, 1$ ) possess the same rank  $M$  for each window length  $L \in \{M, \dots, N - M\}$  (see [9, Lemma 2.1]). Consequently, the sparsity  $M$  of the exponential sum (1.1) coincides with the rank of these Hankel matrices.  $\square$

By the Vandermonde decomposition of the Hankel matrix  $\mathbf{H}_{L,N-L+1}$  we obtain that

$$\mathbf{H}_{L,N-L+1} = \mathbf{V}_{L,M}(\mathbf{z}) (\text{diag } \mathbf{c}) (\mathbf{V}_{N-L+1,M}(\mathbf{z}))^T. \quad (2.9)$$

Under mild conditions, the Hankel matrix  $\mathbf{H}_{L,N-L+1}$  of noiseless data has a bounded spectral norm condition number too.

**Theorem 2.5** Let  $L, N \in \mathbb{N}$  with  $M \leq L \leq N - M + 1$  and  $\min\{L, N - L + 1\} > 2\pi + \frac{1}{q}$  be given. Assume that the frequencies  $\varphi_j \in (-\frac{1}{2}, \frac{1}{2}]$  ( $j = 1, \dots, M$ ) are well-separated by the separation distance  $q > 0$  and that the non-zero coefficients  $c_j$  ( $j = 1, \dots, M$ ) of the exponential sum (1.1) fulfill the condition

$$0 < \gamma_1 \leq |c_j| \leq \gamma_2 < \infty \quad (j = 1, \dots, M). \quad (2.10)$$

Then for all  $\mathbf{y} \in \mathbb{C}^{N-L+1}$

$$\begin{aligned} \gamma_1^2 \alpha_1(L) \alpha_1(N - L + 1) \|\mathbf{y}\|_2^2 &\leq \|\mathbf{H}_{L,N-L+1} \mathbf{y}\|_2^2 \\ &\leq \gamma_2^2 \alpha_2(L) \alpha_2(N - L + 1) \|\mathbf{y}\|_2^2. \end{aligned} \quad (2.11)$$

Further, the lowest resp. largest positive singular value of  $\mathbf{H}_{L,N-L+1}$  can be estimated by

$$\begin{aligned} 0 < \gamma_1 \sqrt{\alpha_1(L) \alpha_1(N - L + 1)} &\leq \sigma_M \leq \sigma_1 \\ &\leq \gamma_2 \sqrt{\alpha_2(L) \alpha_2(N - L + 1)}. \end{aligned} \quad (2.12)$$

The spectral norm condition number of  $\mathbf{H}_{L,N-L+1}$  is bounded by

$$\text{cond}_2 \mathbf{H}_{L,N-L+1} \leq \frac{\gamma_2}{\gamma_1} \sqrt{\frac{\alpha_2(L) \alpha_2(N - L + 1)}{\alpha_1(L) \alpha_1(N - L + 1)}}. \quad (2.13)$$

*Proof.* By the Vandermonde decomposition (2.9) of the Hankel matrix  $\mathbf{H}_{L,N-L+1}$ , we obtain that for all  $\mathbf{y} \in \mathbb{C}^{N-L+1}$

$$\|\mathbf{H}_{L,N-L+1} \mathbf{y}\|_2^2 = \|\mathbf{F}_{L,M} (\text{diag } \mathbf{c}) \mathbf{F}_{N-L+1,M}^T \mathbf{y}\|_2^2.$$

By the discrete Ingham inequalities (2.2) and the assumption (2.10), it follows that

$$\begin{aligned} \gamma_1^2 \alpha_1(L) \|\mathbf{F}_{N-L+1,M}^T \mathbf{y}\|_2^2 &\leq \|\mathbf{H}_{L,N-L+1} \mathbf{y}\|_2^2 \leq \gamma_2^2 \alpha_2(N - L + 1) \\ &\|\mathbf{F}_{N-L+1,M}^T \mathbf{y}\|_2^2. \end{aligned}$$

Using the discrete Ingham inequalities (2.5), we obtain the estimates (2.11). Finally, the estimates of the lowest resp. largest positive singular value and the spectral norm condition number of  $\mathbf{H}_{L,N-L+1}$  arise from (2.11) and the Rayleigh–Ritz Theorem.  $\square$

**Remark 2.6** For fixed  $N$ , the positive singular values as well as the spectral norm condition number of the Hankel matrix  $\mathbf{H}_{L,N-L+1}$  depend strongly on  $L \in \{M, \dots, N - M + 1\}$ . A good criterion for the choice of optimal window length  $L$  is to maximize the lowest positive singular value  $\sigma_M$  of  $\mathbf{H}_{L,N-L+1}$ . It was shown in Potts and Tasche [submitted, Lemma 3.1 and Remark 3.3] that the squared singular values increase almost monotonously for  $L = M, \dots, \lceil \frac{N}{2} \rceil$  and decrease almost monotonously for  $L = \lceil \frac{N}{2} \rceil, \dots, N - M + 1$ . Note that the lower bound (2.12) of the lowest positive singular value  $\sigma_M$  is maximal for  $L \approx \frac{N}{2}$ . Further the upper bound (2.13) of the spectral norm condition number of (2.7) is minimal for  $L \approx \frac{N}{2}$ . Therefore, we prefer to choose  $L \approx \frac{N}{2}$  as optimal window length. Thus, we can ensure that  $\sigma_M > 0$  is not too small. This property is decisively for the correct detection of the sparsity  $M$  in the first step of the MUSIC resp. ESPRIT Algorithm 2.8 resp. 2.9.  $\square$

The ranges of  $\mathbf{H}_{L,N-L+1}$  and  $\mathbf{V}_{L,M}(\mathbf{z})$  coincide in the noiseless case with  $M \leq L \leq N - M + 1$  by (2.9). If  $L > M$ , then the range of  $\mathbf{V}_{L,M}(\mathbf{z})$  is a proper subspace of  $\mathbb{C}^L$ . This subspace is called *left signal space*  $\mathcal{S}_L$ . The left signal space  $\mathcal{S}_L$  is of dimension  $M$  and is generated by the  $M$  columns  $\mathbf{e}_L(\varphi_j)$  ( $j = 1, \dots, M$ ), where

$$\mathbf{e}_L(\varphi) := (e^{2\pi i \ell \varphi})_{\ell=0}^{L-1} \quad \left( \varphi \in \left[ -\frac{1}{2}, \frac{1}{2} \right] \right).$$

Note that  $\|\mathbf{e}_L(\varphi)\|_2 = \sqrt{L}$  for each  $\varphi \in [-\frac{1}{2}, \frac{1}{2}]$ . The *left noise space*  $\mathcal{N}_L$  is defined as the orthogonal complement of  $\mathcal{S}_L$  in  $\mathbb{C}^L$ . The dimension of  $\mathcal{N}_L$  is equal to  $L - M$ .

**Remark 2.7** Let  $M \leq L < N - M + 1$  be given. If we use  $\mathbf{H}_{L,N-L+1}^*$  instead of  $\mathbf{H}_{L,N-L+1}$ , then we can define the *right signal space* as the range of  $\mathbf{V}_{N-L,M}(\bar{\mathbf{z}})$ , where  $\bar{\mathbf{z}}$  denotes the complex conjugate of  $\mathbf{z}$ . The right signal space is an  $M$ -dimensional subspace of  $\mathbb{C}^{N-L+1}$  and is generated by the  $M$  linearly independent vectors  $\mathbf{e}_{N-L+1}(-\varphi_j)$  ( $j = 1, \dots, M$ ). Then the corresponding *right noise space* is the orthogonal complement of the right signal space in  $\mathbb{C}^{N-L+1}$ .  $\square$

By  $\mathbf{Q}_L$  we denote the orthogonal projection onto the left noise space  $\mathcal{N}_L$ . Since  $\mathbf{e}_L(\varphi_j) \in \mathcal{S}_L$  ( $j = 1, \dots, M$ ) and  $\mathcal{N}_L \perp \mathcal{S}_L$ , we obtain that

$$\mathbf{Q}_L \mathbf{e}_L(\varphi_j) = \mathbf{0} \quad (j = 1, \dots, M).$$

If the number  $\varphi \in (-\frac{1}{2}, \frac{1}{2}] \setminus \Phi$ , then the vectors  $\mathbf{e}_L(\varphi_1), \dots, \mathbf{e}_L(\varphi_M), \mathbf{e}_L(\varphi) \in \mathbb{C}^L$  are linearly independent, since the square Vandermonde matrix

$$(\mathbf{e}_L(\varphi_1) \mid \dots \mid \mathbf{e}_L(\varphi_M) \mid \mathbf{e}_L(\varphi))(1 : M + 1, 1 : M + 1)$$

is invertible for each  $L \geq M + 1$ . Hence  $\mathbf{e}_L(\varphi) \notin \mathcal{S}_L = \text{span}\{\mathbf{e}_L(\varphi_1), \dots, \mathbf{e}_L(\varphi_M)\}$ , i.e.  $\mathbf{Q}_L \mathbf{e}_L(\varphi) \neq \mathbf{0}$ . Thus, the frequency set  $\Phi$  can be determined via the zeros of the *left noise-space correlation function*

$$N_L(\varphi) := \frac{1}{\sqrt{L}} \|\mathbf{Q}_L \mathbf{e}_L(\varphi)\|_2 \quad \left( \varphi \in \left(-\frac{1}{2}, \frac{1}{2}\right] \right),$$

since  $N_L(\varphi_j) = 0$  for each  $j = 1, \dots, M$  and  $0 < N_L(\varphi) \leq 1$  for all  $\varphi \in (-\frac{1}{2}, \frac{1}{2}] \setminus \Phi$ , where  $\mathbf{Q}_L \mathbf{e}_L(\varphi)$  can be computed on an equispaced fine grid. Alternatively, one can seek the peaks of the *left imaging function*

$$J_L(\varphi) := \sqrt{L} \|\mathbf{Q}_L \mathbf{e}_L(\varphi)\|_2^{-1} \quad \left( \varphi \in \left(-\frac{1}{2}, \frac{1}{2}\right] \right).$$

In this approach, we prefer the zeros resp. the lowest local minima of the left noise-space correlation function  $N_L(\varphi)$ .

In the next step we determine the orthogonal projection  $\mathbf{Q}_L$  onto the left noise space  $\mathcal{N}_L$ . Here we can use the SVD or the QR decomposition of the  $L$ -trajectory matrix  $\mathbf{H}_{L,N-L+1}$ . For an application of QR decomposition see Potts and Tasche [9]. Applying SVD, we obtain that

$$\mathbf{H}_{L,N-L+1} = \mathbf{U}_L \mathbf{D}_{L,N-L+1} \mathbf{W}_{N-L+1}^*,$$

where  $\mathbf{U}_L \in \mathbb{C}^{L \times L}$  and  $\mathbf{W}_{N-L+1} \in \mathbb{C}^{(N-L+1) \times (N-L+1)}$  are unitary matrices and where  $\mathbf{D}_{L,N-L+1} \in \mathbb{R}^{L \times (N-L+1)}$  is a rectangular diagonal matrix. The diagonal entries of  $\mathbf{D}_{L,N-L+1}$  are the singular values  $\sigma_j$  of the  $L$ -trajectory matrix arranged in non-increasing order  $\sigma_1 \geq \dots \geq \sigma_M > \sigma_{M+1} = \dots = \sigma_{\min\{L,N-L+1\}} = 0$ . Thus, we can determine the sparsity  $M$  of

the exponential sum (1.1) by the number of positive singular values  $\sigma_j$ .

Introducing the matrices

$$\mathbf{U}_{L,M}^{(1)} := \mathbf{U}_L(1 : L, 1 : M), \quad \mathbf{U}_{L,L-M}^{(2)} := \mathbf{U}_L(1 : L, M + 1 : L)$$

with orthonormal columns, we see that the columns of  $\mathbf{U}_{L,M}^{(1)}$  form an orthonormal basis of  $\mathcal{S}_L$  and that the columns of  $\mathbf{U}_{L,L-M}^{(2)}$  are an orthonormal basis of  $\mathcal{N}_L$ . Hence the orthogonal projection onto the left noise space  $\mathcal{N}_L$  has the form

$$\mathbf{Q}_L = \mathbf{U}_{L,L-M}^{(2)} (\mathbf{U}_{L,L-M}^{(2)})^*.$$

Consequently, we obtain that

$$\begin{aligned} \|\mathbf{Q}_L \mathbf{e}_L(\varphi)\|_2^2 &= \langle \mathbf{Q}_L \mathbf{e}_L(\varphi), \mathbf{Q}_L \mathbf{e}_L(\varphi) \rangle = \langle (\mathbf{Q}_L)^2 \mathbf{e}_L(\varphi), \mathbf{e}_L(\varphi) \rangle \\ &= \langle \mathbf{Q}_L \mathbf{e}_L(\varphi), \mathbf{e}_L(\varphi) \rangle \\ &= \langle (\mathbf{U}_{L,L-M}^{(2)})^* \mathbf{e}_L(\varphi), (\mathbf{U}_{L,L-M}^{(2)})^* \mathbf{e}_L(\varphi) \rangle \\ &= \|(\mathbf{U}_{L,L-M}^{(2)})^* \mathbf{e}_L(\varphi)\|_2^2. \end{aligned}$$

Hence the left noise-space correlation function can be represented by

$$N_L(\varphi) = \frac{1}{\sqrt{L}} \|(\mathbf{U}_{L,L-M}^{(2)})^* \mathbf{e}_L(\varphi)\|_2 \quad \left( \varphi \in \left(-\frac{1}{2}, \frac{1}{2}\right] \right).$$

In MUSIC, one determines the lowest local minima of the left noise-space correlation function, see e.g., [1, 4, 26, 27].

**Algorithm 2.8** (MUSIC via SVD)

*Input:*  $N \in \mathbb{N}$  ( $N \geq 2M$ ) number of samples,  $L \approx \frac{N}{2}$  window length,  $\tilde{h}_k = h(k) + e_k \in \mathbb{C}$  ( $k = 0, \dots, N - 1$ ) noisy sampled values of (1.1),  $0 < \varepsilon \ll 1$  tolerance.

1. Compute the SVD of the rectangular Hankel matrix  $\tilde{\mathbf{H}}_{L,N-L+1} = \tilde{\mathbf{U}}_L \tilde{\mathbf{D}}_{L,N-L+1} \tilde{\mathbf{W}}_{N-L+1}^*$  from (2.6), where the singular values  $\tilde{\sigma}_\ell$  ( $\ell = 1, \dots, \min\{L, N - L + 1\}$ ) are arranged in non-increasing order. Determine the numerical rank  $M$  of (2.6) such that  $\tilde{\sigma}_M \geq \varepsilon \tilde{\sigma}_1$  and  $\tilde{\sigma}_{M+1} < \varepsilon \tilde{\sigma}_1$ . Form the matrix  $\tilde{\mathbf{U}}_{L,L-M}^{(2)} = \tilde{\mathbf{U}}_L(1 : L, M + 1 : L)$ .
2. Calculate the left noise-space correlation function  $\tilde{N}_L(\varphi) := \frac{1}{\sqrt{L}} \|(\tilde{\mathbf{U}}_{L,L-M}^{(2)})^* \mathbf{e}_L(\varphi)\|_2$  on an equispaced grid  $\{-\frac{1}{2} + \frac{1}{S}, \dots, \frac{1}{2} - \frac{1}{S}, \frac{1}{2}\}$  for sufficiently large  $S$ .
3. The  $M$  lowest local minima of  $\tilde{N}_L(\frac{2k-S}{2S})$  ( $k = 1, \dots, S$ ) form the frequency set  $\tilde{\Phi} := \{\tilde{\varphi}_1, \dots, \tilde{\varphi}_M\}$ . Set  $\tilde{\mathbf{z}}_j := e^{2\pi i \tilde{\varphi}_j}$  ( $j = 1, \dots, M$ ).
4. Compute the coefficient vector  $\tilde{\mathbf{c}} := (\tilde{c}_j)_{j=1}^M \in \mathbb{C}^M$  as solution of the least squares problem

$$\min_{\tilde{\mathbf{c}} \in \mathbb{C}^M} \|\mathbf{V}_{N,M}(\tilde{\mathbf{z}}) \tilde{\mathbf{c}} - (\tilde{h}_k)_{k=0}^{N-1}\|_2,$$

where  $\tilde{\mathbf{z}} := (\tilde{z}_j)_{j=1}^M$  denotes the vector of computed nodes.

*Output:*  $M \in \mathbb{N}$  sparsity,  $\tilde{\varphi}_j \in (-\frac{1}{2}, \frac{1}{2}]$  frequencies,  $\tilde{c}_j \in \mathbb{C}$  coefficients ( $j = 1, \dots, M$ ).

Let  $L, N \in \mathbb{N}$  with  $M < L \leq N - M + 1$  be given. For noisy sampled data  $\tilde{h}_k = h(k) + e_k$  ( $k = 0, \dots, N - 1$ ), the MUSIC Algorithm 2.8 is relatively insensitive to small perturbations on the data (see [23, Theorem 3]).

In opposite to the MUSIC Algorithm 2.8, the following ESPRIT Algorithm is based on orthogonal projection onto a right signal space. For details see [5, 9, submitted].

**Algorithm 2.9** (ESPRIT via SVD)

*Input:*  $N \in \mathbb{N}$  ( $N \geq 2M$ ) number of samples,  $L \approx \frac{N}{2}$  window length,  $\tilde{h}_k = h(k) + e_k \in \mathbb{C}$  ( $k = 0, \dots, N - 1$ ) noisy sampled values of (1.1),  $0 < \varepsilon \ll 1$  tolerance.

1. Compute the SVD of the rectangular Hankel matrix  $\tilde{H}_{L,N-L+1} = \tilde{U}_L \tilde{D}_{L,N-L+1} \tilde{W}_{N-L+1}^*$  from (2.6), where the singular values  $\tilde{\sigma}_\ell$  ( $\ell = 1, \dots, \min\{L, N-L+1\}$ ) are arranged in non-increasing order. Determine the numerical rank  $M$  of (2.6) such that  $\tilde{\sigma}_M \geq \varepsilon \tilde{\sigma}_1$  and  $\tilde{\sigma}_{M+1} < \varepsilon \tilde{\sigma}_1$ . Form the submatrices

$$\tilde{W}_{N-L,M}(s) := \tilde{W}_{N-L+1}(1 + s : N - L + s, 1 : M) \quad (s = 0, 1).$$

2. Calculate the matrix

$$\tilde{F}_M := \tilde{W}_{N-L,M}(1)^* (\tilde{W}_{N-L,M}(0))^{\dagger},$$

where  $(\tilde{W}_{N-L,M}(0))^{\dagger}$  denotes the Moore–Penrose pseudoinverse.

3. Determine all eigenvalues  $z'_j$  ( $j = 1, \dots, M$ ) of  $\tilde{F}_M$ . Set

$$\tilde{\varphi}_j := \frac{1}{2\pi} \text{Arg} \frac{z'_j}{|z'_j|} \in \left(-\frac{1}{2}, \frac{1}{2}\right) \quad (j = 1, \dots, M),$$

where  $\text{Arg} z \in (-\pi, \pi]$  means the principal value of the argument of  $z \in \mathbb{C} \setminus \{0\}$ .

4. Compute the coefficient vector  $\tilde{c} := (\tilde{c}_j)_{j=1}^M \in \mathbb{C}^M$  as solution of the least squares problem

$$\min_{\tilde{c} \in \mathbb{C}^M} \|V_{N,M}(\tilde{z}) \tilde{c} - (\tilde{h}_k)_{k=0}^{N-1}\|_2,$$

where  $\tilde{z} := (\tilde{z}_j)_{j=1}^M$  denotes the vector of computed nodes  $\tilde{z}_j := e^{2\pi i \tilde{\varphi}_j}$ .

*Output:*  $M \in \mathbb{N}$  sparsity,  $\tilde{\varphi}_j \in (-\frac{1}{2}, \frac{1}{2}]$  frequencies,  $\tilde{c}_j \in \mathbb{C}$  coefficients ( $j = 1, \dots, M$ ).

Note that in Algorithms 2.8 and 2.9 the tolerance can be theoretically chosen as  $\varepsilon = (2 \text{cond}_2 \mathbf{H}_{L,N-L+1})^{-1}$ . Then by Theorem 2.5, the tolerance  $\varepsilon$  is not too small for  $L \approx \frac{N}{2}$ . A simple procedure for the practical choice of  $\varepsilon$  is described in the Example 5.1. By the right choice of the window length  $L \approx \frac{N}{2}$ , we can recover the correct sparsity  $M$  of (1.1) and avoid the determination of spurious frequencies.

The numbers of required samples and the computational costs of the Algorithms 2.8 and 2.9 are listed in Table 2. Thus, the main disadvantages of these algorithms are the high computational costs for large sparsity  $M$ , caused mainly by the SVD. Therefore, in Potts and Tasche [28], we have suggested to use a partial SVD (based on partial Lanczos bidiagonalization) instead of a complete SVD. For both Algorithms 2.8 and 2.9, one needs too many operations in the case of large sparsity  $M$ , see Table 2.

### 3. SPARSE FAST FOURIER TRANSFORM

In this section, we apply Algorithm 2.8 (MUSIC) resp. Algorithm 2.9 (ESPRIT) to the reconstruction of sparse trigonometric polynomials. Clearly, one can approximate the unknown frequencies  $\varphi_j$  of the exponential sum (1.1) by fractions. Therefore, we assume that the unknown frequencies  $\varphi_j$  of (1.1) are fractions  $\frac{\omega_j}{S}$  with  $\omega_j \in (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$ , where  $S$  is a large even integer. Replacing the variable  $x$  by  $Sx$  in (1.1), we obtain the new exponential sum (1.2). Then (1.2) is a 1-periodic trigonometric polynomial with sparsity  $M$ . Consequently we consider the spectral estimation problem (P2) as mentioned in Section 1. A fast algorithm, which solves the problem (P2) or (P2\*), is called *sparse fast Fourier transform* (sparse FFT). In recent years many sublinear algorithms for sparse FFTs were proposed, see Section 1 and Remark 3.1.

In the following, we propose a new deterministic sparse FFT for solving the problem (P2) of a trigonometric polynomial (1.2) with large sparsity  $M$ . Using divide-and-conquer technique, we split the trigonometric polynomial (1.2) of large sparsity  $M$  into some trigonometric polynomials of lower sparsity and determine corresponding samples. Here we borrow an idea from sparse FFT in Christlieb et al. [12] and use shifted sampling of (1.2). For a positive integer  $P \leq S$  and a parameter  $K \in \mathbb{N}$ ,  $K \geq 2$ , we construct a discrete array of samples of size  $P \times (2K + 1)$  via

$$g_P[s, k] := g\left(\frac{s}{P} + \frac{k}{S}\right), \quad (s = 0, \dots, P - 1; k = 0, \dots, 2K). \tag{3.1}$$

For each  $k = 0, \dots, 2K$  we form the discrete Fourier transform (DFT) of length  $P$  and obtain

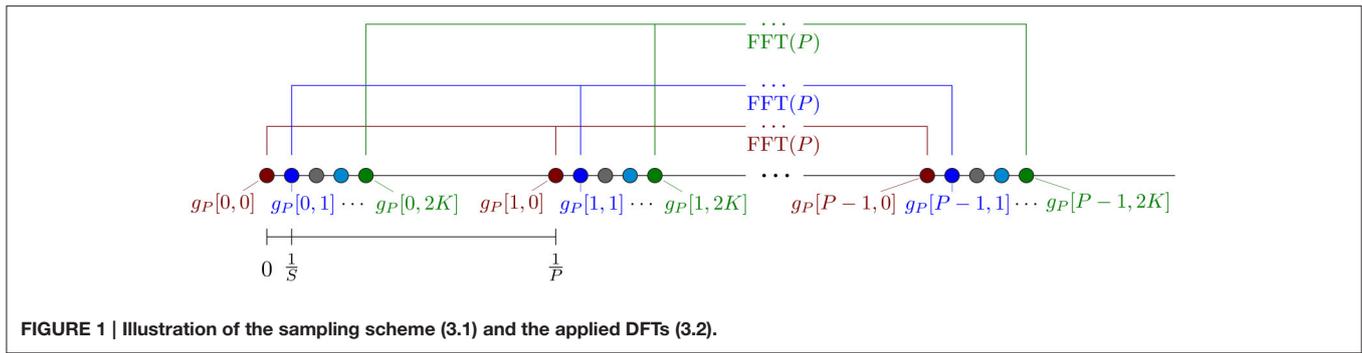
$$\hat{g}_P[\ell, k] := \sum_{s=0}^{P-1} g_P[s, k] e^{-2\pi i s \ell / P} \quad (\ell = 0, \dots, P - 1). \tag{3.2}$$

The fast Fourier transform (FFT) allows the rapid computation of this DFT of length  $P$  in  $\mathcal{O}(P \log P)$  operations. In Figure 1, the sampling scheme and the applied DFTs are visualized. Next, for each  $\ell = 0, \dots, P - 1$ , it follows that

$$\begin{aligned} \hat{g}_P[\ell, k] &= \sum_{s=0}^{P-1} \sum_{j=1}^M c_j e^{2\pi i \omega_j (s/P + k/S)} e^{-2\pi i s \ell / P} \\ &= \sum_{j=1}^M c_j e^{2\pi i \omega_j k / S} \underbrace{\sum_{s=0}^{P-1} e^{2\pi i (\omega_j - \ell) s / P}}_{=0 \text{ or } P}. \end{aligned}$$

**TABLE 2 | Computational costs for the Algorithms 2.8 and 2.9 in the case of  $N$  given samples, where  $S$  is a large even integer so that all frequencies  $\varphi_j$  are of the form  $\frac{\omega_j}{S}$  with  $\omega_j \in (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$ .**

Method	Computational cost
Algorithm 2.8 (MUSIC)	$\mathcal{O}(N^3 + N^2 S + S \log S)$
Algorithm 2.9 (ESPRIT)	$\mathcal{O}(N^3)$



Now we define the index sets

$$I_P(\ell) := \{j \in \{1, \dots, M\} : \omega_j \equiv \ell \pmod{P}\}$$

such that

$$\hat{g}_P[\ell, k] = P \sum_{j \in I_P(\ell)} c_j e^{2\pi i \omega_j k/S}.$$

Consequently, for each  $\ell = 0, \dots, P - 1$ , we may interpret  $\hat{g}_P[\ell, k]$  ( $k = 0, \dots, 2K$ ) as samples of a trigonometric polynomial with frequencies supported only on the index set  $\{\omega_j\}_{j \in I_P(\ell)} \subset \{\omega_1, \dots, \omega_M\}$ , where the samples are taken at the nodes  $k/S$  ( $k = 0, \dots, 2K$ ). In simplified terms, the trigonometric polynomial (1.2) is partitioned into  $P$  many trigonometric polynomials of smaller sparsity.

Next, we use  $K \in \mathbb{N}$ ,  $K \geq 2$ , as sparsity cut-off parameter. This means for each  $\ell = 0, \dots, P - 1$ , we apply Algorithm 2.8 resp. 2.9 to the “samples”  $\tilde{h}_k := \hat{g}_P[\ell, k]$  ( $k = 0, \dots, 2K$ ) and we check if we can uniquely identify all frequencies  $\omega_j, j \in I_P(\ell)$ , i.e., if the determined sparsity  $M_\ell := M$  from Algorithm 2.8 resp. 2.9 fulfills  $M_\ell < K$ . In this case, we use the obtained local fractions  $\tilde{\varphi}_j$  to compute the frequencies  $\omega_j = \text{round}(\tilde{\varphi}_j S)$  by rounding to nearest integer and we use the corresponding coefficients  $c_j$ .

When the condition  $|I_P(\ell)| < K$  is fulfilled for all  $\ell = 0, \dots, P - 1$ , this approach requires  $(2K + 1)P$  samples of  $g$  and  $2K + 1$  FFTs of length  $P$ . The computational costs for the corresponding algorithms are listed in **Table 3**.

If we cannot uniquely identify all frequencies, i.e., if  $|I_P(\ell)| \geq K$  for some  $\ell$ , then we form iteratively the new trigonometric polynomial

$$g_1(x) := g(x) - \sum_{j \in I} c_j e^{2\pi i \omega_j x}, \quad (3.3)$$

where  $I$  is the union of all  $I_P(\ell)$  with the property  $|I_P(\ell)| < K$ . In the next iteration step, we choose a positive integer  $P_1 \leq S$  different from  $P$  and repeat the method on the trigonometric polynomial  $g_1$ . In doing so, we can compute the values

$$\sum_{j \in I} c_j e^{2\pi i \omega_j (\frac{s}{P_1} + \frac{k}{S})} = \sum_{j \in I} \left( c_j e^{2\pi i \frac{\omega_j}{P_1} s} \right) e^{2\pi i \frac{\omega_j}{S} k} \quad (s = 0, \dots, P_1 - 1; k = 0, \dots, 2K)$$

**TABLE 3 | Computational cost of one iteration step of Algorithm 3.2 in the case of  $(2K + 1)P$  given samples (3.1).**

Method	Computational cost
Alg. 3.2 via MUSIC	$\mathcal{O}\left(K^2 P^2 + K^3 P + K^2 S + S \log \frac{S}{P}\right)$
Alg. 3.2 via ESPRIT	$\mathcal{O}\left(K^2 P^2 + K^3 P\right)$

by the non-equispaced fast Fourier transform (NFFT) [29] in  $\mathcal{O}(P_1(K \log K + |I|))$  arithmetic operations.

We perform additional iterations until all frequencies can be identified, i.e., if  $|I_{P_1}(\ell)| < K$  for all  $\ell = 0, \dots, P_1 - 1$ . Note that our algorithm is related to the sparse FFT proposed in Christlieb et al. [12]. But here we use the methods of Section 2, if aliasing with respect to modulo  $P$  occurs.

**Remark 3.1 (Relations to other sparse FFTs)** In Hassanieh et al. [15], an algorithm for the problem (P2\*) is presented, which allows to determine the (unknown) support  $I$  and the Fourier coefficients  $c_j$  from  $\mathcal{O}(M \log S)$  samples with computational cost of  $\mathcal{O}(M \log S)$  operations, as well as a second algorithm, which allows the  $M$ -sparse  $\ell_2$  best approximation of the Fourier coefficients of  $g$  from  $\mathcal{O}(M \log S) \log(S/M)$  samples with computational cost of  $\mathcal{O}(M \log S) \log(S/M)$  operations. As mentioned in the introduction, preliminary tests of the implementation [21] suggest that this method may also work for the problem (P2), if an upper bound for the sparsity of the signal is used as sparsity input parameter. In Indyk et al. [30], another variant was discussed, where the number of samples is  $\mathcal{O}(M \log S) (\log \log S)^{\mathcal{O}(1)}$  and the computational cost is  $\mathcal{O}(M \log^2 S) (\log \log S)^{\mathcal{O}(1)}$ .

Recently in Indyk and Kapralov [31], a result was presented for the multivariate case where the number of required samples is  $\mathcal{O}(M \log S)$  for constant dimensions  $d$  and the computational cost is  $\mathcal{O}(S^d \log^{\mathcal{O}(1)} S)$ . In general the exact constants, especially the dependence on  $d$ , are unknown due to missing implementations. For instance the number of samples  $\mathcal{O}(M \log S)$  of the last mentioned algorithm contains a factor of  $d^{\mathcal{O}(d)}$ , see Section [31, Section 4].

Moreover, a deterministic sparse FFT, using the Chinese Remainder Theorem, was presented in [13] for the univariate case and in Iwen [32] for the multivariate case, which takes  $\mathcal{O}(d^4 M^2 \log^4(dS))$  samples and arithmetic operations.

This means there is neither an exponential/super-exponential dependency on the dimension  $d \in \mathbb{N}$  nor a dependency on a failure probability in the asymptotics of the number of samples and arithmetic operations for this method. Besides this deterministic algorithm, there also exists a randomized version which only requires  $\mathcal{O}(d^4 M \log^4(dS))$  samples and arithmetic operations.

Recently, another sparse FFT, which is based on a multiscale approach, was presented in Christlieb et al. [12] as an extension of the method [11]. Their algorithm is able to handle (additive) noise and requires  $\mathcal{O}(M \log(S/M))$  on average.

For further references on the sparse FFT we refer to the nice webpage <http://groups.csail.mit.edu/netmit/sFFT/>. Despite the fact that the computational cost of the sparse FFT is lower, it is not clear which algorithm is indeed faster and more stable for the practical problems mentioned in the Section 5.

We stress again that it is already well known that one can use Prony-type methods for the sparse FFT, see e.g., Heider et al. [33]. Using the proposed splitting approach, one can significantly decrease the high computational cost of MUSIC and ESPRIT, but keep the numerical stable evaluation. Clearly one can combine the suggested method with a reconstruction of the non-zero Fourier coefficients in a dimension incremental way [34].  $\square$

All of the methods described in Section 2 apply an SVD and use the tolerance  $\varepsilon$  as a relative threshold parameter to determine the local sparsity  $M_\ell$  of the signal. A good choice of this parameter may depend without limitation on noise in the sampling values of the trigonometric polynomial  $g$  and on the smallest distance between two frequencies, where this distance may change for each  $\ell \in \{0, \dots, P-1\}$  in each iteration. We propose to use a (small) list of possible relative threshold parameters  $\varepsilon$ , which are tested for each  $\ell \in \{0, \dots, P-1\}$  in each iteration.

Our sparse FFT for recovery of a trigonometric polynomial (1.2) with large sparsity  $M$  reads as follows:

**Algorithm 3.2** (Sparse FFT via MUSIC resp. ESPRIT, see Algorithm 6.1 for detailed listing with extended parameter list).

*Input:*  $S \in 2\mathbb{N}$  frequency grid parameter,  $K \in \mathbb{N}$  ( $K \geq 2$ ) Hankel matrix size parameter,  $P \in \mathbb{N}$  ( $P \leq S$ ) initial FFT length,  $\tilde{g}$  1-periodic sparse trigonometric polynomial (1.2) of unknown sparsity  $M \in \mathbb{N}$  with frequencies in  $(-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$  perturbed by noise.

1. For each  $k = 0, \dots, 2K$ , sample  $\tilde{g}_P[s, k] := \tilde{g}(\frac{s}{P} + \frac{k}{S})$ ,  $s = 0, \dots, P-1$ , and compute  $(\hat{g}_P[\ell, k])_{\ell=0}^{P-1}$  by FFT of  $(\tilde{g}_P[s, k])_{s=0}^{P-1}$ .
2. **for**  $\ell = 0, \dots, P-1$ 
  - 2.1. Apply Algorithm 2.8 resp. 2.9 with  $L := K$  and  $N := 2K+1$  on the values  $(\tilde{h}_k := \hat{g}_P[\ell, k])_{k=0}^{2K}$  to obtain the local sparsity  $M_\ell := M$  and the local fractions  $\tilde{\varphi}_{\ell, m} := \tilde{\varphi}_m \in (-\frac{1}{2}, \frac{1}{2}]$  for  $m = 1, \dots, M_\ell$ . If  $M_\ell \geq K$ , then go to step 2 and

continue with next  $\ell$ . Otherwise, compute the local frequencies  $\omega_{\ell, m} := \text{round}(\tilde{\varphi}_{\ell, m} S)$  by rounding to nearest integer.

- 2.2. Compute the local coefficients  $c_{\ell, m}$  as least squares solution of the overdetermined Vandermonde system

$$\min_{(c_{\ell, m})_{m=1}^{M_\ell} \in \mathbb{C}^{M_\ell}} \left\| P \left( e^{2\pi i k \omega_{\ell, m} / S} \right)_{k=0, m=1}^{2K, M_\ell} - (c_{\ell, m})_{m=1}^{M_\ell} - (\hat{g}_P[\ell, k])_{k=0}^{2K} \right\|_2. \quad (3.4)$$

- 2.3. If the residual of (3.4) is small (see step 3.3.6 of Algorithm 6.1), then append the frequencies  $\omega_{\ell, m}$  ( $m = 1, \dots, M_\ell$ ) to the frequency set  $\Omega$ .
3. If the (global) residual

$$\max_{\substack{s=0, \dots, P-1 \\ k=0, \dots, 2K}} \left| \sum_{j=1}^{|\Omega|} C[j'] e^{2\pi i \Omega[j'] (\frac{s}{P} + \frac{k}{S})} - g\left(\frac{s}{P} + \frac{k}{S}\right) \right|$$

is small, then exit the algorithm. Otherwise, form the new trigonometric polynomial (3.3). In the next iteration step choose a positive integer  $P_1 \leq S$  different from  $P$ , sample (3.3) on  $\frac{s}{P_1} + \frac{k}{S}$  for  $s = 0, \dots, P_1-1$  and  $k = 0, \dots, 2K$ , and repeat the above method.

*Output:*  $\Omega \subset (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$  set of recovered frequencies  $\omega_j$  ( $j = 1, \dots, M$ ),  $M := |\Omega|$  detected sparsity,  $c_j \in \mathbb{C}$  coefficient related to  $\omega_j$  ( $j = 1, \dots, M$ ).

For  $(2K+1)P$  given samples (3.1), the computational cost for one iteration of Algorithm 3.2 is shown in **Table 3**.

If we take  $(2K+1)P = \mathcal{O}(M)$  samples, then we obtain minimal computational cost for the parameters  $K = \mathcal{O}(M^{1/3})$  and  $P = \mathcal{O}(M^{2/3})$ . For this case, we compare the numbers of required samples and computational costs for different methods of spectral estimation in **Table 4** such as sparse FFT via MUSIC, sparse FFT via ESPRIT, MUSIC, ESPRIT, and classical FFT. As we can see, the sparse FFT via ESPRIT is very useful for the spectral estimation by a relatively low number of samples and low computational cost.

## 4. RECONSTRUCTION OF MULTIVARIATE TRIGONOMETRIC POLYNOMIALS

Let  $d, M \in \mathbb{N}$  with  $d > 1$  be given. We consider the  $d$ -variate exponential sum of sparsity  $M$

$$g(\mathbf{x}) := \sum_{j=1}^M c_j e^{2\pi i \omega_j \cdot \mathbf{x}} \quad (4.1)$$

for  $\mathbf{x} = (x_1, \dots, x_d)^T \in \mathbb{R}^d$  with non-zero coefficients  $c_j \in \mathbb{C}$  and distinct frequency vectors  $\omega_j \in \mathbb{Z}^d$ . Here the dot in the

**TABLE 4 | Numbers of required samples and computational costs using the splitting approach for one iteration of Algorithm 3.2 as well as for Algorithms 2.8 and 2.9 in the case  $K = \mathcal{O}(M^{1/3})$ ,  $P = \mathcal{O}(M^{2/3})$  and  $M \approx L/2 \approx N/4$ .**

Method	Samples	Computational cost
Alg. 3.2 via MUSIC Alg. 2.8	$\mathcal{O}(M)$	$\mathcal{O}\left(M^{2/3}S + S \log \frac{S}{M^{2/3}}\right)$
Alg. 3.2 via ESPRIT Alg. 2.9	$\mathcal{O}(M)$	$\mathcal{O}\left(M^{5/3}\right)$
Alg. 2.8 (MUSIC)	$\mathcal{O}(M)$	$\mathcal{O}\left(M^2S + S \log S\right)$
Alg. 2.9 (ESPRIT)	$\mathcal{O}(M)$	$\mathcal{O}\left(M^3\right)$
FFT of length S	S	$\mathcal{O}(S \log S)$

exponent denotes the usual scalar product in  $\mathbb{R}^d$ . Note that the function (4.1) is a  $d$ -variate trigonometric polynomial of sparsity  $M$  which is 1-periodical with respect to each variable. Let  $\Omega := \{\omega_1, \dots, \omega_M\}$  be the set of the frequency vectors.

Assume that it is known *a priori* that  $\omega_j$  are contained in a frequency set  $\Gamma \subset \mathbb{Z}^d$ . Then the cardinality of  $\Gamma$  satisfies  $|\Gamma| \geq M$ . Examples of possible frequency sets  $\Gamma$  are the cube  $\{\mathbf{k} \in \mathbb{Z}^d : \|\mathbf{k}\|_\infty \leq N\}$  and the symmetric hyperbolic cross

$$\left\{ \mathbf{k} = (k_s)_{s=1}^d \in \mathbb{Z}^d : \prod_{s=1}^d \max\{1, |k_s|\} \leq N \right\}.$$

For given  $\mathbf{z} \in \mathbb{Z}^d$  and  $S \in \mathbb{N}$ , the set

$$\Lambda(\mathbf{z}, S) := \left\{ \mathbf{x}_k = \frac{k}{S} \mathbf{z} \bmod \mathbf{I}; k = 0, \dots, S-1 \right\} \subset \mathbb{T}^d \simeq [0, 1)^d$$

is called *rank-1 lattice*, where  $\mathbf{I} := (1, \dots, 1)^T$ . Note that  $\mathbf{x}_k = \mathbf{x}_{k+nS}$  for  $k = 0, \dots, S-1$  and  $n \in \mathbb{Z}$ . For given  $\Gamma \subset \mathbb{Z}^d$ , there exists a *reconstructing rank-1 lattice*  $\Lambda(\mathbf{z}, S)$  such that the matrix

$$\mathbf{A}_{S,|\Gamma|} := \left( e^{2\pi i \mathbf{k} \cdot \mathbf{x}} \right)_{\mathbf{x} \in \Lambda(\mathbf{z}, S), \mathbf{k} \in \Gamma}$$

fulfills the condition (see [35] and [36, Section 3.2])

$$\mathbf{A}_{S,|\Gamma|}^* \mathbf{A}_{S,|\Gamma|} = S \mathbf{I}_{|\Gamma|}. \tag{4.2}$$

Then we consider the following spectral estimation problem:

(P3) Assume that  $\omega_j \in \Gamma$  ( $j = 1, \dots, M$ ) and that  $\Lambda(\mathbf{z}, S)$  is a reconstructing rank-1 lattice with respect to  $\Gamma$ . Recover the sparsity  $M \in \mathbb{N}$ , all frequencies  $\omega_j \in \Gamma$  as well as all non-zero coefficients  $c_j \in \mathbb{C}$  of the  $d$ -variate exponential sum (4.1), if noisy sampled data

$$\tilde{g}_k := g(\mathbf{x}_k) + e_k \quad \left( |e_k| \leq \frac{1}{10} \min_j |c_j| \right)$$

for all  $k = 0, \dots, 2L-2$  are given, where  $\mathbf{x}_k \in \Lambda(\mathbf{z}, S)$ ,  $S \geq L > M$  and  $e_k \in \mathbb{C}$  are small error terms.

For simplicity we discuss only noiseless data. Let  $\mathbf{H}_L := (g(\mathbf{x}_{k+n}))_{k,n=0}^{L-1}$  be the *response matrix* of the given data. Then

$\mathbf{H}_L$  is a Hankel matrix. Further we introduce the *rectangular Fourier-type matrix*

$$\mathbf{F}_{L,M} := \left( e^{2\pi i \omega_j \cdot \mathbf{x}_k} \right)_{k=0, j=1}^{L-1, M}.$$

From (4.2) it follows in the case  $L = S$  that  $\mathbf{F}_{S,M}^* \mathbf{F}_{S,M} = S \mathbf{I}_M$  and hence for all  $\mathbf{x} \in \mathbb{C}^M$

$$\|\mathbf{F}_{S,M} \mathbf{x}\|_2^2 = \mathbf{x}^* \mathbf{F}_{S,M}^* \mathbf{F}_{S,M} \mathbf{x} = S \|\mathbf{x}\|_2^2.$$

Consequently, all positive singular values of  $\mathbf{F}_{S,M}$  are equal to  $\sqrt{S}$  and  $\text{cond}_2 \mathbf{F}_{S,M} = 1$ .

The matrix  $\mathbf{H}_L$  can be represented in the form

$$\mathbf{H}_L = \mathbf{F}_{L,M} \left( \text{diag}(c_j)_{j=1}^M \right) \mathbf{F}_{L,M}^T. \tag{4.3}$$

The ranges of  $\mathbf{H}_L$  and  $\mathbf{F}_{L,M}$  coincide in the noiseless case by (4.3). The range of  $\mathbf{F}_{L,M}$  is a proper subspace of  $\mathbb{C}^L$ . This subspace is called *left signal space*  $\mathcal{S}_L$ . The left signal space  $\mathcal{S}_L$  is of dimension  $M$  and is generated by the  $M$  columns  $\mathbf{e}_L(\omega_j)$  ( $j = 1, \dots, M$ ), where

$$\mathbf{e}_L(\omega) := \left( e^{2\pi i \omega \cdot \mathbf{x}_k} \right)_{k=0}^{L-1} \quad (\omega \in \Gamma).$$

Note that  $\|\mathbf{e}_L(\omega)\|_2 = \sqrt{L}$  for each  $\omega \in \Gamma$ . The *left noise space*  $\mathcal{N}_L$  is defined as the orthogonal complement of  $\mathcal{S}_L$  in  $\mathbb{C}^L$ . The dimension of  $\mathcal{N}_L$  is equal to  $L - M > 0$ .

By  $\mathbf{Q}_L$  we denote the orthogonal projection onto the left noise space  $\mathcal{N}_L$ . Since  $\mathbf{e}_L(\omega_j) \in \mathcal{S}_L$  ( $j = 1, \dots, M$ ) and  $\mathcal{N}_L \perp \mathcal{S}_L$ , we obtain that

$$\mathbf{Q}_L \mathbf{e}_L(\omega_j) = \mathbf{0} \quad (j = 1, \dots, M).$$

If  $\omega \in \Gamma \setminus \Omega$ , then the vectors  $\mathbf{e}_L(\omega_1), \dots, \mathbf{e}_L(\omega_M), \mathbf{e}_L(\omega) \in \mathbb{C}^L$  are linearly independent for  $S \geq L > M$ . This can be seen as follows: For distinct  $\omega, \omega' \in \Gamma$ , it follows by [36, Lemma 3.1] that

$$\omega \cdot \mathbf{z} \not\equiv \omega' \cdot \mathbf{z} \pmod{S}.$$

Consequently the vectors

$$\mathbf{e}_L(\omega_j) := \left( e^{2\pi i (\omega_j \cdot \mathbf{z}) \frac{k}{S}} \right)_{k=0}^{L-1} \quad (j = 1, \dots, M)$$

and  $\mathbf{e}_L(\omega)$  with  $\omega \in \Gamma \setminus \Omega$  are linearly independent for  $S \geq L > M$ , since the square Vandermonde matrix

$$\left( |\mathbf{e}_L(\omega_1)| \dots |\mathbf{e}_L(\omega_M)| |\mathbf{e}_L(\omega)| \right) (1 : M + 1, 1 : M + 1)$$

is invertible for each  $L \geq M + 1$ . Hence

$$\mathbf{e}_L(\omega) \notin \mathcal{S}_L = \text{span} \{ \mathbf{e}_L(\omega_1), \dots, \mathbf{e}_L(\omega_M) \},$$

i.e.  $\mathbf{Q}_L \mathbf{e}_L(\omega) \neq \mathbf{0}$ .

Thus, the frequency vectors can be determined via the  $M$  zeros resp. lowest local minima of the *left noise-space correlation function*

$$N_L(\omega) := \frac{1}{\sqrt{L}} \|\mathbf{Q}_L \mathbf{e}_L(\omega)\|_2 \quad (\omega \in \Gamma)$$

or via the  $M$  peaks of the *left imaging function*

$$J_L(\boldsymbol{\omega}) := \sqrt{L} \|\mathbf{Q}_L \mathbf{e}_L(\boldsymbol{\omega})\|_2^{-1} \quad (\boldsymbol{\omega} \in \Gamma).$$

Similar to Section 2, one can determine the left noise-space correlation function resp. the left imaging function on  $\Gamma$  by using SVD of the response matrix  $\mathbf{H}_L$ .

Now, we proceed analogously to Section 3. For a positive integer  $P \leq S$  and a parameter  $K \in \mathbb{N}, K \geq 2$ , we construct the sampling array of (4.1) of size  $P \times (2K + 1)$  via

$$g_P[s, k] := g\left(\left(\frac{s}{P} + \frac{k}{S}\right)\mathbf{z}\right) \quad (s = 0, \dots, P-1; k = 0, \dots, 2K).$$

As in the univariate case, for each  $k = 0, \dots, 2K$  we form the DFT of length  $P$

$$\hat{g}_P[\ell, k] := \sum_{s=0}^{P-1} g_P[s, k] e^{-2\pi i s \ell / P} \quad (\ell = 0, \dots, P-1).$$

For each  $\ell = 0, \dots, P-1$ , we obtain that

$$\begin{aligned} \hat{g}_P[\ell, k] &= \sum_{s=0}^{P-1} \sum_{j=1}^M c_j e^{2\pi i (s/P + k/S) \boldsymbol{\omega}_j \cdot \mathbf{z}} e^{-2\pi i s \ell / P} \\ &= \sum_{j=1}^M c_j e^{2\pi i k \boldsymbol{\omega}_j \cdot \mathbf{z} / S} \sum_{s=0}^{P-1} e^{2\pi i ((\boldsymbol{\omega}_j \cdot \mathbf{z}) - \ell) s / P}. \end{aligned}$$

Introducing the index sets

$$I_P(\ell) := \left\{ j \in \{1, \dots, M\} : \boldsymbol{\omega}_j \cdot \mathbf{z} \equiv \ell \pmod{P} \right\} \quad (\ell = 0, \dots, P-1),$$

it follows that

$$\hat{g}_P[\ell, k] = P \sum_{j \in I_P(\ell)} c_j e^{2\pi i k \boldsymbol{\omega}_j \cdot \mathbf{z} / S}.$$

This means for each  $\ell = 0, \dots, P-1$ , we may interpret  $\hat{g}_P[\ell, k]$  ( $k = 0, \dots, 2K$ ) as samples of a multivariate trigonometric polynomial with frequencies supported on the index set  $\{\boldsymbol{\omega}_j\}_{j \in I_P(\ell)} \subset \{\boldsymbol{\omega}_1, \dots, \boldsymbol{\omega}_M\}$ , where the samples are taken at the nodes  $\frac{k}{S}\mathbf{z}$  ( $k = 0, \dots, 2K$ ). In **Figure 2**, a two-dimensional

example is shown which visualizes the partitioning by the index sets  $I_P(\ell)$ .

Next, we apply Algorithm 2.8 resp. 2.9 with  $L := K$  and  $N := 2K + 1$  on the values  $\hat{h}_k := \hat{g}_P[\ell, k]$  ( $k = 0, \dots, 2K$ ) for each  $\ell = 0, \dots, P-1$  to obtain the local sparsity  $M_\ell = M$  and the local fractions  $\tilde{\varphi}_{\ell, m} := \tilde{\varphi}_m \in (-\frac{1}{2}, \frac{1}{2}]$  for  $m = 1, \dots, M_\ell$ . Afterwards, we compute the one-dimensional frequencies  $\omega_{\ell, m} := \text{round}(\tilde{\varphi}_{\ell, m} S) \in (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$  by rounding to nearest integer. We transform these one-dimensional frequencies  $\omega_{\ell, m}$  into their  $d$ -dimensional counterparts  $\boldsymbol{\omega}_{\ell, m} \in \Gamma$  using the relation  $\boldsymbol{\omega}_{\ell, m} \cdot \mathbf{z} \equiv \omega_{\ell, m} \pmod{S}$  given by the reconstructing rank-1 lattice  $\Lambda(\mathbf{z}, S)$ . Then we compute coefficients  $c_{\ell, m}$  from the samples  $\hat{g}_P[\ell, k]$  ( $k = 0, \dots, 2K$ ) by solving the corresponding overdetermined Vandermonde system. If we cannot identify all the frequencies, i.e., if  $|I_P(\ell)| \geq K$  for some indices  $\ell$ , we consider the new trigonometric polynomial

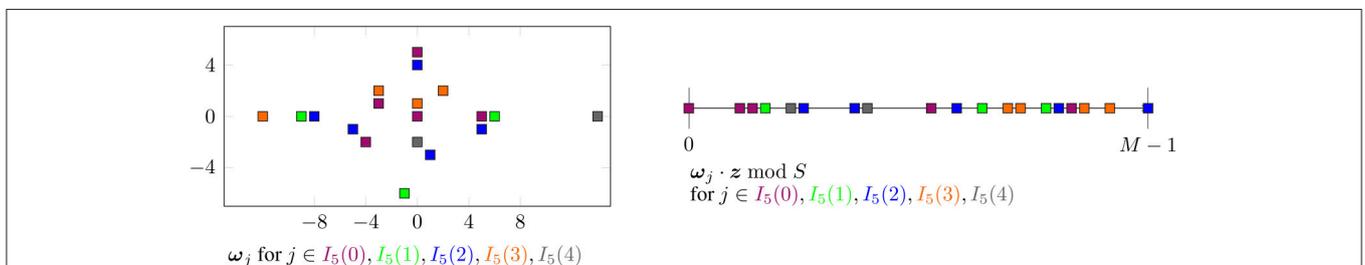
$$g_1(\mathbf{x}) := g(\mathbf{x}) - \sum_{j \in I} c_j e^{2\pi i \boldsymbol{\omega}_j \cdot \mathbf{x}} = \sum_{j=1}^M c_j e^{2\pi i \boldsymbol{\omega}_j \cdot \mathbf{x}} - \sum_{j \in I} c_j e^{2\pi i \boldsymbol{\omega}_j \cdot \mathbf{x}} \quad (\mathbf{x} \in \mathbb{T}^d) \quad (4.4)$$

in an additional iteration, where the index set  $I$  contains all index sets  $I_P(\ell)$  with  $|I_P(\ell)| < K$ . In the next iteration, we choose a positive integer  $P_1 \leq S$  different from  $P$  and repeat the method for the trigonometric polynomial  $g_1$ . In doing so, we compute the values

$$\sum_{j \in I} c_j e^{2\pi i (\frac{s}{P_1} + \frac{k}{S}) \boldsymbol{\omega}_j \cdot \mathbf{z}} = \sum_{j \in I} \left( c_j e^{2\pi i \frac{\boldsymbol{\omega}_j \cdot \mathbf{z}}{P_1} s} \right) e^{2\pi i \frac{\boldsymbol{\omega}_j \cdot \mathbf{z}}{S} k} \quad (s = 0, \dots, P_1 - 1; k = 0, \dots, 2K)$$

of the second sum in (4.4) evaluated at the nodes  $\mathbf{x} = (\frac{s}{P_1} + \frac{k}{S})\mathbf{z}$  with the univariate NFFT [29] in  $\mathcal{O}(P_1(K \log K + |I|))$  arithmetic operations. We perform additional iterations until all frequencies can be identified, i.e.  $|I_{P_1}(\ell)| < K$  for all  $\ell = 0, \dots, P_1 - 1$ .

We modify Algorithm 3.2 from Section 3 as described above and additionally in the following way. Here, we describe the changes in the detailed listing (see Algorithm 6.1) of Algorithm 3.2. In step 1, we sample the multivariate



**FIGURE 2 | Illustration of an example frequency index set  $\{\boldsymbol{\omega}_j\}_{j=1}^{19}$  and corresponding one-dimensional frequencies  $\boldsymbol{\omega}_j \cdot \mathbf{z} \bmod S$  partitioned by  $I_P(\ell)$  with parameters  $P := 5, \mathbf{z} := (1, 33)^T, S := 37$ .**

trigonometric polynomial at the nodes  $\left(\frac{s}{P} + \frac{k}{S}\right) \cdot z$  ( $s = 0, \dots, P - 1, k = 0, \dots, 2K$ ). In step 3.3.3, we compute the local frequencies  $\omega_{\ell,m} := \text{round}(\tilde{\varphi}_{\ell,m} S)$  for  $m = 1, \dots, M_\ell$ . Next, we compute the  $d$ -dimensional counterparts  $\omega_{\ell,m}$  of the one-dimensional frequencies  $\omega_{\ell,m}$  using the relation  $\omega_{\ell,m} \cdot z \equiv \omega_{\ell,m} \pmod{S}$ . In step 3.3.4, we filter the frequencies  $\omega_{\ell,m}$  by removing non-unique ones and by keeping only those with  $\omega_{\ell,m} \cdot z \equiv \ell \pmod{P}$ . We remark that we have to modify step 3.3.4 and that we have to perform the conversion of one-dimensional frequencies  $\omega_{\ell,m}$  to their  $d$ -dimensional counterparts  $\omega_{\ell,m}$  before the filtering, since the conditions  $\omega_{\ell,m} \cdot z \equiv \ell \pmod{P}$  and  $\omega_{\ell,m} \equiv \ell \pmod{P}$  are not equivalent in general if  $P$  is not a divisor of  $S$ .

## 5. NUMERICAL EXPERIMENTS

In this section, we present some numerical results for Algorithm 3.2. The related software is available from the authors' homepage. All computations are performed in MATLAB with IEEE double-precision arithmetic. First we consider noiseless sampled data and later the case, where the sampled data are perturbed by additive (white) Gaussian noise. Finally we present some numerical results for the modified Algorithm 3.2 of Section 4.

### 5.1. Noiseless Sampled Data

**Example 5.1** From noiseless sampled values, we reconstruct 100 trigonometric polynomials (1.2) of sparsity  $M = 256$  with random frequencies  $\omega_j \in \left(-\frac{S}{2}, \frac{S}{2}\right) \cap \mathbb{Z}$  and random coefficients  $c_j$  on the unit circle. We set the array of relative SVD threshold values  $\text{epsilon\_svd\_list} := [10^{-3}, 10^{-4}, \dots, 10^{-8}]$ , the parameter  $\epsilon_{\text{spatial}} := 10^{-8}$ , the absolute value of minimal non-zero coefficients  $\epsilon_{\text{fc\_min}} := 10^{-1} = 10^{-1} \cdot \min_j |c_j|$  and the maximal number of iterations  $R := 10$ , see Algorithm 6.1 for the extended parameter list. Applying the sparse FFT Algorithm 3.2 with MUSIC in the case  $S = 2^{16}$  with parameters  $K \in \{6, 12, 16\}$  and  $P \in \{16, 32, 64, 128\}$ , we can successfully detect all integer frequencies  $\omega_j$ . In Table 5, the column "iterations" depicts the maximal number of iterations actually used by the Algorithm 3.2 (computed over 100 trigonometric polynomials). The column "samples" contains the maximal number of sampled values used by the Algorithm 3.2. The column " $\ell^2$ -errors" shows the maximal relative  $\ell^2$ -error of the coefficients, which are locally computed in step 2.2 of Algorithm 3.2. The column "updated  $\ell^2$ -errors" shows the maximal relative  $\ell^2$ -error of the coefficients, which are determined by additionally solving one large Vandermonde system at the end of Algorithm 3.2 with all frequencies as well as all samples of (1.2). For comparison, the classical FFT of length  $2^{16}$  requires  $2^{16}$  samples and the resulting  $\ell^2$ -error is  $2.6e-16$ . The minimal number of samples for the cases  $K \in \{6, 12, 16\}$  and  $P \in \{16, 32, 64, 128\}$  is reached for  $K = P = 16$  with 1716 samples, the next smallest number of samples is 1725 for  $K = 12$  and  $P = 32$ . If we do not use the splitting approach ( $P = 1$  and  $R = 1$ ), we observe that the detection of some frequencies fails for exactly 1 of the 100 signals for  $K = 750$  and the detection of all frequencies of all 100 signals succeeds for  $K = 850$  requiring 1701 samples. This number of samples is very close

to the minimum of 1716 samples from above. However, a direct application of MUSIC method (entry  $K = 850$ ) has distinctly more computational cost than using the sparse FFT Algorithm 3.2. Note that  $R$  denotes the maximal number of iterations in Algorithm 6.1, the detailed description of Algorithm 3.2.  $\square$

**Example 5.2** Now we apply Algorithm 3.2 via ESPRIT with the same parameters as in Example 5.1. Then we obtain identical results for the number of iterations and samples as well as almost identical  $\ell^2$ -errors, but Algorithm 3.2 via ESPRIT has lower computational cost. If we do not use the splitting approach ( $P = 1$  and  $R = 1$ ), we observe that the detection of some frequencies fails for exactly 2 of the 100 signals for  $K = 750$  and the detection of all frequencies of all 100 signals again succeeds for  $K = 850$  requiring 1701 samples.

Additionally, we apply the implementation [21] of the sfft version 3 algorithm. The number of used samples is 7669, which is about two to four times the number of samples for the MUSIC and ESPRIT algorithm in Table 5, and the maximal relative  $\ell^2$ -error of the coefficients is  $3.6e-04$ , which is about five orders of magnitude larger.  $\square$

**Example 5.3** We generate 100 random trigonometric polynomials (1.2), where the coefficients  $c_j$  are drawn uniformly at random from  $[-1, 1] + [-1, 1]i$  with  $|c_j| \geq 10^{-2}$ . We set the absolute value of minimal non-zero coefficients  $\epsilon_{\text{fc\_min}} := 10^{-3} = 10^{-1} \cdot \min_j |c_j|$ . For the cases  $K \in \{6, 12, 16\}$  and  $P \in \{16, 32, 64, 128\}$ , we apply Algorithm 3.2 via ESPRIT. We obtain results almost identical to the ones from Example 5.2. This means, all frequencies of all 100 signals are correctly detected and the maximal relative  $\ell^2$ -errors differ slightly from

**TABLE 5 | Results for Algorithm 3.2 via MUSIC for frequency grid parameter  $S = 2^{16}$  and sparsity  $M = 256$  with random coefficients on the unit circle.**

$K$	$P$	Iterations	Samples	$\ell^2$ -errors	Updated $\ell^2$ -errors
6	16	10	3939	3.1e-09	2.8e-15
6	32	5	2600	8.5e-10	2.5e-15
6	64	3	2626	3.0e-10	2.5e-15
6	128	2	3367	5.7e-11	1.9e-14
12	16	5	2600	3.4e-10	2.4e-15
12	32	2	1725	1.7e-10	9.2e-15
12	64	2	3275	1.3e-10	7.9e-14
12	128	1	3200	1.4e-11	1.6e-14
16	16	3	1716	5.2e-10	2.3e-15
16	32	2	2277	2.3e-10	4.4e-14
16	64	2	4323	9.9e-11	3.3e-14
16	128	1	4224	7.4e-12	6.3e-15
850	1	1	1701	2.3e-13	2.3e-13

The entry with parameter  $P=1$  in the last row shows the result for Algorithm 2.8 (MUSIC). The results for Algorithm 3.2 via ESPRIT and for Algorithm 2.9 (ESPRIT) are identical for the shown parameters  $K$  and  $P$  except for slightly different  $\ell^2$ -errors.

those in **Table 5**. The maximal numbers of iterations and samples are identical.

Additionally, we apply the implementation [21] of the sfft version 3 algorithm. For each signal, the number of taken samples is 7669. Only for 16 of the 100 signals, all frequencies are detected correctly, whereas between 1 and 7 frequencies are not correctly detected for 84 signals.  $\square$

**Example 5.4** Next we apply Algorithm 3.2 via ESPRIT for signals with higher sparsity. From noiseless sampled values, we reconstruct 100 trigonometric polynomials (1.2) of sparsity  $M = 1024$  with random frequencies  $\omega_j \in (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$  and random coefficients  $c_j$  on the unit circle. The results for the frequency grid parameter  $S := 2^{22}$  are shown in **Table 6**. The minimal number of samples is about six times higher compared to the results in **Table 5**.

In general, we observe that the maximal number of used iterations decreases for increasing initial FFT length  $P \in \{64, 128, 256, 512\}$  as well as for increasing values  $K \in \{8, 10, 12\}$ . In the cases, where all frequencies of all the 100 trigonometric polynomials are correctly detected, the number of required samples first decreases and later increases again for increasing initial FFT length  $P$  and fixed values  $K$ . The reason for this is that the number of samples per iteration increases for growing FFT length, while the number of used iterations decreases until its minimum one is reached.

Again, we apply the implementation [21] of the sfft version 3 algorithm. Here, the number of used samples is 31718, which is up to three times the number of samples of Algorithm 3.2 via ESPRIT in **Table 6**, and the maximal relative  $\ell^2$ -error of the coefficients is 3.6e-04, which is about five orders of magnitude larger than in **Table 6**.  $\square$

### 5.2. Noisy Case

In this subsection, we test the robustness to noise of Algorithm 3.2. For this, we perturb the samples of the

trigonometric polynomials  $g$  from (1.2) by additive complex white Gaussian noise with zero mean and standard deviation  $\sigma$ , i.e., we have measurements  $\tilde{g}(\frac{k}{S} + \frac{s}{P}) = g(\frac{k}{S} + \frac{s}{P}) + \eta_{k,s}$ , where  $\eta_{k,s} \in \mathbb{C}$  are independent and identically distributed complex Gaussian noise values. Then, we may approximately compute the signal-to-noise ratio (SNR) in our case by

$$\text{SNR} \approx \frac{\frac{1}{S} \sum_{k=0}^{S-1} |g(\frac{k}{S})|^2}{\frac{1}{S} \sum_{k=0}^{S-1} |\eta_{k,0}|^2} \approx \frac{\sum_{j=1}^M |c_j|^2}{\sigma^2}.$$

Correspondingly, we choose  $\sigma = \|(c_j)_{j=1}^M\|_2 / \sqrt{\text{SNR}}$  for a targeted SNR value. For the numerical computations in MATLAB, we generate the noise by  $\eta_{k,s} := \sigma / \sqrt{2} * (\text{randn} + \text{li} * \text{randn})$ . Moreover, we choose the parameter  $\varepsilon_{\text{spatial}} := 5\sigma$  and this means that the absolute value of the noise  $|\eta_{k,s}| \leq \varepsilon_{\text{spatial}}$  for more than 99.9998% of the noise values  $\eta_{k,s}$ . Since we assume that the absolute value of the noise  $\leq \frac{1}{10} \min_j |c_j|$  throughout this paper, we should choose the SNR such that  $5\sigma = 5\|(c_j)_{j=1}^M\|_2 / \sqrt{\text{SNR}} \leq \frac{1}{10} \min_j |c_j|$ , which yields  $\text{SNR} \geq 50^2 \|(c_j)_{j=1}^M\|_2^2 / (\min_j |c_j|)^2$ .

**Example 5.5** As in Example 5.1, we generate 100 trigonometric polynomials (1.2) of sparsity  $M = 256$  with random frequencies  $\omega_j \in (-\frac{S}{2}, \frac{S}{2}] \cap \mathbb{Z}$  and random coefficients  $c_j$  from the unit circle. We set the frequency grid parameter  $S = 2^{16}$ , the signal sparsity  $M = 256$ , the array of relative SVD threshold values `epsilon_svd_list` :=  $[10^{-3}, 10^{-4}, \dots, 10^{-8}]$  and the maximal number of iterations  $R := 10$ . Here, we set the absolute value of minimal non-zero coefficients  $\varepsilon_{\text{fc\_min}} := 10^{-1} = 10^{-1} \cdot \min_j |c_j|$  and we use the parameters  $P \in \{32, 64, 128\}$  and  $K \in \{12, 24\}$ . For possible SNR values, we have  $\text{SNR} \geq 50 \|(c_j)_{j=1}^M\|_2^2 / (\min_j |c_j|)^2 = 6.4 \cdot 10^5$  and we consider the SNR values  $10^{10}$ ,  $10^8$ , and  $10^6$  in our numerical tests. The results of Algorithm 3.2 via ESPRIT are presented in **Table 7**. Additionally, we test the sparsity cut-off parameter  $K_2 \in \mathbb{N}$  differently from the Hankel matrix size parameter  $K$ , see Algorithm 6.1. Here, we use the parameter combinations  $(K, K_2) \in \{(12, 6), (12, 12), (24, 12)\}$ . In general, we observe that we require more samples for  $\text{SNR} = 10^6$  than for  $\text{SNR} = 10^8$  and again more for  $\text{SNR} = 10^8$  than for  $\text{SNR} = 10^{10}$ . The relative errors are about one order of magnitude larger for  $\text{SNR} = 10^6$  than for  $\text{SNR} = 10^8$  as well as for  $\text{SNR} = 10^8$  than for  $\text{SNR} = 10^{10}$ , since the maximal noise values  $\eta_{k,s}$  are larger by about one order of magnitude with high probability each time. Moreover, the maximal number of samples in the noisy case is higher than in the noiseless case, cf. **Table 5**. For some parameter combinations at  $\text{SNR} = 10^8$ , exactly one of the 100 signals is not correctly detected and this is indicated by the entry “-” in the column “ $\ell^2$ -errors” resp. “updated  $\ell^2$ -errors.” For  $\text{SNR} = 10^6$  with  $K = 12$ ,  $K_2 = 6$ , and  $P = 64$  exactly one frequency at three of the 100 signals is not correctly detected, whereas all frequencies of all 100 signals for  $\text{SNR} = 10^6$  are correctly detected in the other cases shown in **Table 7**. All parameters of (1.2) are correctly detected in the case  $\text{SNR} = 10^{10}$  for the parameter combinations  $(K, K_2) \in \{(12, 6), (12, 12), (24, 12)\}$  and  $P = 32$ . For the considered test

**TABLE 6 | Results for Algorithm 3.2 via ESPRIT for frequency grid parameter  $S = 2^{22}$  and sparsity  $M = 1024$  with random coefficients on the unit circle.**

$K$	$P$	Iterations	Samples	$\ell^2$ -errors	Updated $\ell^2$ -errors
8	64	10	14059	-	-
8	128	8	19635	1.3e-09	6.0e-15
8	256	3	13192	1.0e-09	5.0e-15
8	512	2	17561	8.3e-10	9.0e-15
10	64	10	17367	2.7e-09	5.7e-15
10	128	6	17535	1.2e-09	5.5e-15
10	256	2	10773	1.2e-09	4.7e-15
10	512	2	21693	6.2e-10	1.6e-14
12	64	10	20675	1.5e-09	6.0e-15
12	128	4	13375	1.0e-09	5.0e-15
12	256	2	12825	1.1e-09	5.1e-15
12	512	2	25825	1.1e-09	1.9e-14

**TABLE 7 | Results for Algorithm 3.2 via ESPRIT for frequency grid parameter  $S = 2^{16}$  and sparsity  $M = 256$  with noisy data.**

SNR	$K$	$K_2$	$P$	Iterations	Samples	$\ell^2$ -errors	Updated $\ell^2$ -errors
$10^{10}$	12	6	32	6	6325	1.4e-05	2.8e-06
$10^{10}$	12	12	32	5	5000	1.7e-05	3.4e-06
$10^{10}$	24	12	32	3	5390	8.9e-06	3.0e-06
$10^{10}$	3000	3000	1	1	6001	-	-
$10^{10}$	3500	3500	1	1	7001	3.3e-06	3.3e-06
$10^8$	12	6	32	7	7800	-	-
$10^8$	12	6	64	4	6875	3.9e-04	2.8e-05
$10^8$	12	6	128	3	9900	2.0e-04	2.2e-05
$10^8$	12	12	32	6	6325	-	-
$10^8$	12	12	64	4	6875	-	-
$10^8$	12	12	128	3	9900	2.0e-04	2.2e-05
$10^8$	24	12	32	4	7497	3.9e-04	2.8e-05
$10^8$	24	12	64	3	9898	1.4e-04	2.2e-05
$10^8$	24	12	128	2	12691	2.4e-05	2.3e-05
$10^8$	3000	3000	1	1	6001	4.2e-05	4.2e-05
$10^8$	3500	3500	1	1	7001	2.9e-05	2.9e-05
$10^6$	12	6	64	10	20675	-	-
$10^6$	12	6	128	8	28875	7.2e-03	1.5e-04
$10^6$	12	12	128	7	24800	5.7e-03	1.5e-04
$10^6$	24	12	32	6	12397	1.5e-02	2.0e-04
$10^6$	24	12	64	4	13475	8.5e-03	2.1e-04
$10^6$	24	12	128	3	19404	5.8e-03	1.6e-04

parameters, the choices  $(K, K_2) \in \{(12, 6), (24, 12)\}$ , which yield a higher oversampling within the ESPRIT algorithm, give slightly better results compared to the choice  $K = K_2 = 12$ .

Additionally, we apply the implementation [21] of the sfft version 3 algorithm. We remark that this particular algorithm is not suited for noisy samples. The number of taken samples is 7669 for all signals, which is about 50 percent higher than for Algorithm 3.2 at  $\text{SNR} = 10^{10}$  and similar at  $\text{SNR} = 10^8$ . In the cases  $\text{SNR} = 10^{10}$  and  $\text{SNR} = 10^8$ , all frequencies of all 100 signals are correctly detected and the maximal relative  $\ell^2$ -errors of the Fourier coefficients are about  $3.6e-04$ . For  $\text{SNR} = 10^6$ , all frequencies of 94 of the 100 signals are correctly detected and up to four frequencies at four of the 100 signals are not correctly detected.  $\square$

**Example 5.6** Additionally, we generate 100 random trigonometric polynomials (1.2), where the coefficients  $c_j$  are drawn uniformly at random from  $[-1, 1] + [-1, 1]i$  with  $|c_j| \geq 10^{-2}$ . We set the absolute value of minimal non-zero coefficients  $\varepsilon_{fc\_min} := 10^{-3} = 10^{-1} \cdot \min_j |c_j|$ .

**TABLE 8 | Results of the modified Algorithm 3.2 via ESPRIT with sparsity  $M = 256$ , frequency vectors within six-dimensional hyperbolic cross index set  $\Gamma = \{k \in \mathbb{Z}^6 : \prod_{s=1}^6 \max\{1, |k_s|\} \leq 16\}$ , and reconstructing rank-1 lattice  $\Lambda(z, S)$  with  $z = (1, 33, 579, 3628, 21944, 169230)^T$  and  $S = 1105193$ .**

SNR	$K$	$K_2$	$P$	Iterations	Samples	$\ell^2$ -errors	Updated $\ell^2$ -errors
$\infty$	8	8	8	10	3893	5.1e-08	6.2e-11
$\infty$	8	8	16	10	5151	5.1e-08	6.4e-11
$\infty$	8	8	32	10	8687	6.6e-09	9.3e-11
$\infty$	8	8	64	3	3434	2.6e-09	3.2e-10
$\infty$	8	8	128	2	4403	2.8e-09	1.8e-09
$\infty$	10	10	8	9	3948	5.0e-09	6.5e-11
$\infty$	10	10	16	10	6363	1.0e-08	6.8e-11
$\infty$	10	10	32	10	10731	3.9e-09	1.1e-10
$\infty$	10	10	64	10	17367	6.2e-09	5.5e-10
$\infty$	10	10	128	2	5439	2.3e-09	1.7e-09
$\infty$	12	12	8	10	5725	8.6e-09	6.8e-11
$\infty$	12	12	16	10	7575	1.1e-08	7.1e-11
$\infty$	12	12	32	3	2750	2.9e-09	1.3e-10
$\infty$	12	12	64	4	6875	3.0e-09	8.5e-10
$\infty$	12	12	128	2	6475	2.2e-09	1.9e-09
$10^{10}$	12	12	64	4	6875	2.2e-05	2.5e-06
$10^{10}$	24	12	32	5	9800	6.2e-05	2.4e-06
$10^{10}$	24	12	64	3	9898	8.4e-06	2.2e-06
$10^8$	24	12	32	10	25039	8.6e-05	1.4e-05
$10^8$	24	12	64	7	25774	5.1e-05	1.5e-05
$10^6$	48	24	256	10	267429	1.8e-03	4.6e-05
$10^6$	48	24	512	5	256468	7.3e-04	4.3e-05

In the case  $\text{SNR} = 10^8$ , we observe in each considered parameter combination that the correct detection of one or two frequencies fails for several of the 100 trigonometric polynomials. The most likely reason is the fact that the smallest coefficient can be very close to the noise level. If we decrease the noise by one order of magnitude, i.e.  $\text{SNR} = 10^{10}$ , the frequency detection succeeds for all considered parameter combinations.

Furthermore, we generate 100 random trigonometric polynomials (1.2), where the coefficients  $c_j$  are drawn uniformly at random from  $[-1, 1] + [-1, 1]i$  with  $|c_j| \geq 10^{-1}$ . Then we set the absolute value of minimal non-zero coefficients  $\varepsilon_{fc\_min} := 10^{-2} = 10^{-1} \cdot \min_j |c_j|$ . This means that the smallest possible coefficient as well as the parameter  $\varepsilon_{fc\_min}$  are by one order of magnitude larger than before. Now in both of the cases  $\text{SNR} = 10^{10}$  and  $\text{SNR} = 10^8$ , we observe for each parameter combination  $(K, K_2) \in \{(12, 6), (12, 12), (24, 12)\}$  and  $P \in \{32, 64, 128\}$  that all frequencies of all trigonometric polynomials are correctly detected.

For results of the implementation [21] of the sfft version 3 algorithm, we refer to Example 5.3.  $\square$

### 5.3. Reconstruction of 6-variate Trigonometric Polynomials

Finally we test the modified Algorithm 3.2 of Section 4 for the reconstruction of six-variate trigonometric polynomials with the sparsity  $M = 256$ .

**Example 5.7** We choose the index set  $\Gamma$  of possible frequency vectors as the six-dimensional hyperbolic cross  $\Gamma := \{\mathbf{k} \in \mathbb{Z}^6 : \prod_{s=1}^6 \max\{1, |k_s|\} \leq 16\}$  of cardinality 169209. Further we use the reconstructing rank-1 lattice  $\Lambda(\mathbf{z}, S)$  with generating vector  $\mathbf{z} = (1, 33, 579, 3628, 21944, 169230)^T$  and rank-1 lattice size  $S = 1105193$ , see Kämmerer et al. [37, Table 6.2]. We generate 100 random trigonometric polynomials (4.1) with sparsity  $M = 256$ , where the frequency vectors  $\omega_j$  ( $j = 1, \dots, M$ ) are chosen uniformly at random from  $\Gamma$  (without repetition) and the corresponding coefficients  $c_j$  are randomly chosen on the unit circle. We set the array of relative SVD threshold values `epsilon_svd_list` :=  $[10^{-3}, 10^{-4}, \dots, 10^{-8}]$ , the absolute value of minimal non-zero coefficients  $\varepsilon_{fc\_min} := 10^{-1}$ , and the maximal number of iterations  $R := 10$ . In the noiseless case, we set the parameter  $\varepsilon_{spatial} := 10^{-8}$ , and in the noisy case as described in Section 5.2. The results of the modified Algorithm 3.2 via ESPRIT are presented in **Table 8**.

The columns of **Table 8** have the same meaning as in Section 5.2. For the noiseless case, i.e., “SNR =  $\infty$ ,” we observe the same behavior as in the one-dimensional case in Section 5.1. The detection of all frequency vectors of all 100 trigonometric polynomials (4.1) succeeds for  $K = K_2 \in \{8, 10, 12\}$  and  $P \in \{8, 16, 32, 64, 128\}$ . For the noisy case, the results are worse

than in **Table 7** of the one-dimensional case. The reason for this is that we have a bijective mapping between six-dimensional  $\omega_j$  and one-dimensional frequencies  $\omega_j$  by means of the reconstructing rank-1 lattice,  $\omega_j \cdot \mathbf{z} \equiv \omega_j \pmod{S}$ , and the rank-1 lattice size  $S$  influences how close two distinct one-dimensional fractional frequencies  $\omega'_j/S$  and  $\omega''_j/S$  may get in the ESPRIT algorithm, see also **Figure 2**.

We tried to apply the implementation [21] of the sfft version 3 algorithm in this test setting. However, the implementation failed with an internal error caused by the used problem size  $n = S = 1105193$  and sparsity  $M = 256$ . Another test run with rank-1 lattice size  $S = 2^{21} = 2097152$  yielded 7938 samples and a maximal relative  $\ell^2$ -error of  $6.8e-04$ , where the latter is about 4 orders of magnitude larger than the results in **Table 8**. Moreover, we consider the noisy case. Again, we remark that the implementation [21] is not suited for noisy samples. For SNR =  $10^{10}$ , the detection of one frequency fails for one of the 100 signals. If we increase the SNR to  $10^8$ , then between one and 9 frequencies are not correctly detected for 81 of the 100 signals.  $\square$

### ACKNOWLEDGMENTS

The first and last named author gratefully acknowledge the funding by the European Union and the Free State of Saxony (EFRE/ESF NBest-SF). The results of this paper were first presented during the Dagstuhl Seminar 15251 on “Sparse modeling and multi-exponential analysis” (June 14 – 19, 2015). Moreover, the authors would like to thank the referees for their valuable suggestions.

### REFERENCES

- Manolakis DG, Ingle VK, Kogon SM. *Statistical and Adaptive Signal Processing*. Boston, MA: McGraw-Hill (2005).
- Stoica PG, Moses RL. *Spectral Analysis of Signals*. Upper Saddle River, NJ: Prentice Hall (2005).
- Plonka G, Tasche M. Prony methods for recovery of structured functions. *GAMM-Mitt.* (2014) **37**:239–58. doi: 10.1002/gamm.2014.10011
- Schmidt RO. Multiple emitter location and signal parameter estimation. *IEEE Trans Antennas Propag.* (1986) **34**:276–80.
- Roy R, Kailath T. ESPRIT—estimation of signal parameters via rotational invariance techniques. *IEEE Trans Acoustic Speech Signal Process.* (1989) **37**:984–94.
- Hua Y, Sarkar TK. Matrix pencil method for estimating parameters of exponentially damped/undamped sinusoids in noise. *IEEE Trans Acoust Speech Signal Process.* (1990) **38**:814–24.
- Golub GH, Milanfar P, Varah J. A stable numerical method for inverting shape from moments. *SIAM J Sci Comput.* (1999) **21**:1222–43.
- Roy R, Kailath T. ESPRIT—estimation of signal parameters via rotational invariance techniques. In: Auslander L, Grünbaum FA, Helton JW, Kailath T, Khargoneka P, Mitter S, editors. *Signal Processing, Part II. Vol. 23 of IMA Volumes in Mathematics and its Applications*. New York, NY: Springer (1990). pp. 369–411.
- Potts D, Tasche M. Parameter estimation for nonincreasing exponential sums by Prony-like methods. *Linear Algebra Appl.* (2013) **439**:1024–39. doi: 10.1016/j.laa.2012.10.036
- Pereyra V, Scherer G. Exponential data fitting. In: Pereyra V, Scherer G, editors. *Exponential Data Fitting and its Applications*. Sharjah: Bentham Science Publishers; IEEE Computer Society (2010). pp. 1–26.
- Lawlor D, Wang Y, Christlieb A. Adaptive sub-linear time Fourier algorithms. *Adv Adapt Data Anal.* (2013) **5**:25. doi: 10.1142/S1793536913500039
- Christlieb A, Lawlor D, Wang. A multiscale sub-linear time Fourier algorithm for noisy data. *Appl Comput Harmon Anal.* (in press). doi: 10.1016/j.acha.2015.04.002
- Iwen MA. Combinatorial sublinear-time Fourier algorithms. *Found Comput Math.* (2010) **10**:303–38. doi: 10.1007/s10208-009-9057-1
- Ben-Or M, Tiwari P. A deterministic algorithm for sparse multivariate polynomial interpolation. In: *Twentieth Annual ACM Symposium on the Theory of Computing*. New York, NY: ACM Press (1988). pp. 301–9.
- Hassanieh H, Indyk P, Katabi D, Price E. Nearly optimal sparse Fourier transform. In: *Proceedings of the Forty-fourth Annual ACM Symposium on Theory of Computing*. New York, NY: ACM (2012). pp. 563–78.
- Gilbert AC, Indyk P, Iwen M, Schmidt L. Recent developments in the sparse Fourier transform: a compressed Fourier transform for big data. *IEEE Signal Proc Mag.* (2014) **31**:91–100. doi: 10.1109/MSP.2014.2329131
- Rauhut H. Random sampling of sparse trigonometric polynomials. *Appl Comput Harmon Anal.* (2007) **22**:16–42. doi: 10.1016/j.acha.2006.05.002
- Kunis S, Rauhut H. Random sampling of sparse trigonometric polynomials II, orthogonal matching pursuit versus basis pursuit. *Found Comput Math.* (2008) **8**:737–63. doi: 10.1007/s10208-007-9005-x
- Gröchenig K, Pötscher BM, Rauhut H. Learning trigonometric polynomials from random samples and exponential inequalities for eigenvalues of random matrices. arXiv:math/0701781v2. (2010).

20. Foucart S, Rauhut H. *A Mathematical Introduction to Compressive Sensing*. New York, NY: Birkhäuser;Springer (2013).
  21. Schumacher J. sFFT 0.1.0 (2013). Available online at: <http://spiral.net/software/sfft.html>
  22. Schumacher J, Püschel M. High-performance sparse fast Fourier transforms. In: *Proceedings of IEEE Workshop on Signal Processing Systems (SIPS)*. Belfast, UK: IEEE (2014). pp. 1–6.
  23. Liao W, Fannjiang A. MUSIC for single-snapshot spectral estimation: stability and super-resolution. *Appl Comput Harmon Anal.* (2016) **40**:33–67. doi: 10.1016/j.acha.2014.12.003
  24. Peter T, Potts D, Tasche M. Nonlinear approximation by sums of exponentials and translates. *SIAM J Sci Comput.* (2011) **33**:314–34. doi: 10.1137/100790094
  25. Potts D, Tasche M. Parameter estimation for multivariate exponential sums. *Electron Trans Numer Anal.* (2013) **40**:204–224.
  26. Fannjiang AC. The MUSIC algorithm for sparse objects: a compressed sensing analysis. *Inverse Problems* (2011) **27**:32. doi: 10.1088/0266-5611/27/3/035013
  27. Kirsch A. The MUSIC algorithm and the factorization method in inverse scattering theory for inhomogeneous media. *Inverse Problems* (2002) **18**:1025–1040. doi: 10.1088/0266-5611/18/4/306
  28. Potts D, Tasche M. Fast ESPRIT algorithms based on partial singular value decompositions. *Appl Numer Math.* (2015) **88**:31–45. doi: 10.1016/j.apnum.2014.10.003
  29. Keiner J, Kunis S, Potts D. Using NFFT3 - a software library for various nonequispaced Fast Fourier Transforms. *ACM Trans Math Softw.* (2009) **36**: 1–30. doi: 10.1145/1555386.1555388
  30. Indyk P, Kapralov M, Price E. (Nearly) sample-optimal sparse Fourier transform. In: *Proceedings of the Forty-fourth Annual ACM Symposium on Theory of Computing*. Portland: ACM (2014). pp. 563–78.
  31. Indyk P, Kapralov M. Sample-optimal Fourier sampling in any constant dimension. In: *Foundations of Computer Science (FOCS), 2014 IEEE 55th Annual Symposium on*. Philadelphia, PA (2014). pp. 514–23.
  32. Iwen MA. Improved approximation guarantees for sublinear-time Fourier algorithms. *Appl Comput Harmon Anal.* (2013) **34**:57–82. doi: 10.1016/j.acha.2012.03.007
  33. Heider S, Kunis S, Potts D, Veit M. A sparse prony FFT. In: *10th International Conference on Sampling Theory and Applications*. Bremen (2013).
  34. Potts D, Volkmer T. Sparse high-dimensional FFT based on rank-1 lattice sampling. *Appl Comput Harmon Anal.* (in press). doi: 10.1016/j.acha.2015.05.002
  35. Kämmerer L. Reconstructing multivariate trigonometric polynomials from samples along rank-1 lattices. In: Fasshauer GE, Schumaker LL, editors. *Approximation Theory XIV: San Antonio 2013*. Cham: Springer International Publishing (2014). pp. 255–71.
  36. Kämmerer L. *High Dimensional Fast Fourier Transform Based on Rank-1 Lattice Sampling*. Dissertation. Universitätsverlag Chemnitz (2014). Available online at: <http://nbn-resolving.de/urn:nbn:de:bsz:ch1-qucosa-157673>
  37. Kämmerer L, Potts D, Volkmer T. Approximation of multivariate periodic functions by trigonometric polynomials based on rank-1 lattice sampling. *J Complex.* (2015) **31**:543–76. doi: 10.1016/j.jco.2015.02.004
- Conflict of Interest Statement:** The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.
- Copyright © 2016 Potts, Tasche and Volkmer. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) or licensor are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.

## APPENDIX

### Detailed Sparse FFT Algorithm

**Algorithm 6.1** (Detailed listing of Algorithm 3.2 with extended parameter list)

*Input:*  $S \in 2\mathbb{N}$  frequency grid parameter,  $K \in \mathbb{N}$  Hankel matrix size parameter,  $K_2 \in \mathbb{N}$  sparsity cut-off parameter (default value  $K$ ),  $P \in \mathbb{N}$  initial FFT length,  $g$  1-periodic sparse trigonometric polynomial of unknown sparsity  $M \in \mathbb{N}$  with frequencies in  $(-\frac{S}{2}, \frac{S}{2}) \cap \mathbb{Z}$ , `epsilon_svd_list` array of relative SVD threshold values  $0 < \epsilon_{\text{SVD}} < 1$  in descending order,  $\epsilon_{\text{spatial}} > 0$  estimate for maximal noise value,  $\epsilon_{\text{fc\_min}} > 0$  lower bound of absolute values of non-zero coefficients,  $R \in \mathbb{N}$  maximal number of iterations.

Create empty index set array  $\Omega$  and coefficient array  $C$ .

**for** iteration  $r := 1, \dots, R$

1. Construct the discrete array of samples of  $g$  of length  $P \times (2K + 1)$  via

$$g_P[s, k] := g\left(\frac{s}{P} + \frac{k}{S}\right) - \sum_{j'=1}^{|\Omega|} C[j'] e^{2\pi i \Omega[j']\left(\frac{s}{P} + \frac{k}{S}\right)} \quad (s = 0, \dots, P-1; k = 0, \dots, 2K).$$

2. Compute for each  $k = 0, \dots, 2K$  an FFT of length  $P$  and obtain array  $\hat{g}_P$  of length  $P \times (2K + 1)$ ,  $\hat{g}_P[\ell, k] := \sum_{s=0}^{P-1} g_P[s, k] e^{-2\pi i s \ell / P}$  for  $\ell = 0, \dots, P-1$ , if  $P > 1$ . Otherwise if  $P = 1$ , then set  $\hat{g}_P[\ell, k] := g_P[\ell, k]$  for  $\ell = 0, \dots, P-1$ .

3. **for**  $\ell := 0, \dots, P-1$

- 3.1. If  $\|\hat{g}_P[\ell, 0 : 2K]\|_\infty / P < \epsilon_{\text{spatial}}$ , then go to 3. and continue with next  $\ell$ .

- 3.2. Set variable `found_svd` := 0.

- 3.3. **for**  $\epsilon_{\text{SVD}}$  in `epsilon_svd_list`

- 3.3.1. Apply Algorithm 2.8 resp. 2.9 with  $L := K$ ,  $N := 2K + 1$  and  $\epsilon := \epsilon_{\text{SVD}}$  on the values  $(\hat{h}_k := \hat{g}_P[\ell, k])_{k=0}^{2K}$  to obtain the (local) sparsity  $M_\ell := \hat{M}$  and fractions  $\tilde{\varphi}_{\ell, m} := \tilde{\varphi}_m \in (-\frac{1}{2}, \frac{1}{2}]$  for  $j = m, \dots, M_\ell$ .

- 3.3.2. If  $M_\ell \geq K_2$  then go to 3.3. and continue with next (smaller)  $\epsilon_{\text{SVD}}$ .

- 3.3.3. Compute local frequencies  $\omega_{\ell, m} := \text{round}(\tilde{\varphi}_{\ell, m} S)$  for  $m = 1, \dots, M_\ell$ .

- 3.3.4. Filter frequencies  $\omega_{\ell, m}$  by removing non-unique ones and by keeping only those

where  $\omega_{\ell, m} \equiv \ell \pmod{P}$ . Set  $M_\ell$  to number of resulting frequencies  $\omega_{\ell, m}$ .

- 3.3.5. Compute (local) Fourier coefficients  $c_{\ell, m}$  as least squares solution from the overdetermined Vandermonde system  $(\hat{g}_P[\ell, 0 : 2K])^T \approx (e^{2\pi i k \omega_{\ell, m} / S})_{k=0; m=1}^{2K; M_\ell} (P \cdot c_{\ell, m})_{m=1}^{M_\ell}$ .

- 3.3.6. If  $\|(e^{2\pi i k \omega_{\ell, m} / S})_{k=0; m=1}^{2K; M_\ell} (c_{\ell, m})_{m=1}^{M_\ell} - (\hat{g}_P[\ell, 0 : 2K])^T / P\|_\infty \leq 10 \cdot \epsilon_{\text{spatial}}$ , then set variable `found_svd` := 1, leave **for**  $\epsilon_{\text{SVD}}$  loop and go to 3.5.

Otherwise, go to 3.3. and continue with next (smaller)  $\epsilon_{\text{SVD}}$ .

3.3. **end for**  $\epsilon_{\text{SVD}}$

- 3.4. If `found_svd`  $\neq$  1, then go to 3. and continue with next  $\ell$ .

- 3.5. If a frequency has already been found, i.e.,  $\omega_{\ell, m} = \Omega[j']$  for any  $m = 1, \dots, M_\ell$ , then update the corresponding coefficient  $C[j']$  by computing  $C[j'] := C[j'] + c_{\ell, m}$ .

- 3.6. Append new frequencies of  $\omega_{\ell, m}$ ,  $m = 1, \dots, M_\ell$ , to array  $\Omega$  and append corresponding coefficients to array  $C$ .

3. **end for**  $\ell$

4. Remove small coefficients  $|C[j']| < \epsilon_{\text{fc\_min}}$  from array  $C$  and remove corresponding frequencies from array  $\Omega$  for any  $j'$ .

5. If the residual

$$\max_{\substack{s=0, \dots, P-1 \\ k=0, \dots, 2K}} \left| \sum_{j'=1}^{|\Omega|} C[j'] e^{2\pi i \Omega[j']\left(\frac{s}{P} + \frac{k}{S}\right)} - g\left(\frac{s}{P} + \frac{k}{S}\right) \right| < 10 \cdot \epsilon_{\text{spatial}},$$

then set  $R_{\text{used}} := r$  and exit  $r$ -loop.

Otherwise, determine next prime number larger than current FFT length  $P$  and use this larger prime as  $P$  in the next iteration.

**end for** iteration  $r$

*Output:* Detected sparsity  $M := |\Omega| \in \mathbb{N}$ , array  $\Omega \subset (-\frac{S}{2}, \frac{S}{2}) \cap \mathbb{Z}$  of detected frequencies, array  $C \in \mathbb{C}^M$  of corresponding coefficients.