

Prony Methods for Recovery of Structured Functions

Gerlind Plonka^{1,*} and Manfred Tasche^{2,**}

¹ Institute of Numerical and Applied Mathematics, University of Göttingen, Lotzestraße 16-18, 37083 Göttingen, Germany

² Institute of Mathematics, University of Rostock, Ulmenstr. 69, 18057 Rostock, Germany

Received 25.10.2013, revised 03.06.2014, accepted ???

Published online ???

Key words Prony methods, Padé approximation, annihilating filter method, linear prediction, times series, exponential sum, extended exponential sum, Hankel matrix, Prony polynomial, sparse expansion of eigenfunctions, recovery of structured functions, reconstruction of sparse vectors.

Subject classification 65D10, 41A45, 65F15, 65F20, 62M10, 94A12

In this survey, we describe the classical Prony method and whose relatives. We sketch a frequently used Prony-like method for equispaced sampled data, namely the ESPRIT method. The case of nonequispaced sampled data is discussed too. For the reconstruction of a sparse eigenfunction expansion, a generalized Prony method is presented. The Prony methods are applied to the recovery of structured functions (such as exponential sums and extended exponential sums) and of sparse vectors. The recovery of spline functions with arbitrary knots from Fourier data is also based on Prony methods. Finally, some numerical examples are given.

Copyright line will be provided by the publisher

1 Introduction

The recovery of a structured function from noisy sampled data is a fundamental problem in applied mathematics and signal processing. In this survey, we describe the Prony methods and present the numerical solution of three recovery problems. The first problem arises in electrical engineering, signal processing, and mathematical physics and is known as frequency analysis problem (see [6, 23]):

(i) Recover the positive integer M , distinct numbers $f_j \in [-\alpha, 0] + i[-\pi, \pi]$ with $\alpha > 0$, and complex coefficients $c_j \neq 0$, $j = 1, \dots, M$, in the *exponential sum of order M*

$$h(x) := \sum_{j=1}^M c_j e^{f_j x}, \quad x \geq 0, \quad (1)$$

if noisy sampled data $h_k := h(k) + e_k$ ($k = 0, \dots, 2N - 1$) with $N \geq M$ are given, where e_k are small error terms. Note that the real part of f_j is the damping factor and that

* Corresponding author E-mail: plonka@math.uni-goettingen.de, Phone: +00 49 551 3922115, Fax: +00 49 551 3933944

** E-mail: manfred.tasche@uni-rostock.de

Copyright line will be provided by the publisher

the imaginary part of f_j is the angular frequency of the exponential $e^{f_j x}$. Supposed that a function $h(x)$ is *a priori* known to be a sparse linear combination of exponentials from the set $\{e^{f x} : f \in [-\alpha, 0] + i[-\pi, \pi]\}$, the problem (i) can be seen as a nonlinear approximation problem to recover the best M -term approximation of h .

The reconstruction of a compactly supported, structured function from the knowledge of samples of its Fourier transform is a common problem in several scientific areas such as radio astronomy, computerized tomography, and magnetic resonance imaging. If the structured function is a piecewise polynomial, then the second problem appears (see [2, 3, 27]):

(ii) Determine the breakpoints and the associated jump magnitudes of a compactly supported, piecewise polynomial, if finitely many values of its Fourier transform are given.

In different applications as e.g. seismic exploration and nondestructive testing of materials, one is concerned with the problem whether a signal vector \mathbf{x} can be completely reconstructed from a small amount of suitable linear measurements if it satisfies the *a priori* assumption to be sparse. This problem is of special interest if the measurements are time-consuming or very expensive. This approach can be seen as a new paradigm, called compressive sensing (see [10]), and has been extensively investigated within the last years. A vector $\mathbf{x} \in \mathbb{C}^D$ is called *M-sparse*, if $M \ll D$ and if only M components of \mathbf{x} are different from zero. Then the third problem reads as follows (see [24]):

(iii) Recover an M -sparse vector $\mathbf{x} \in \mathbb{C}^D$ with $M \ll D$, if only few scalar products $\mathbf{a}_k^T \mathbf{x}$ with suitable chosen vectors $\mathbf{a}_k \in \mathbb{C}^D$ are given.

In this paper, we will show that all three problems can be solved by deterministic Prony methods. The outline of this paper is as follows. In Section 2, we describe the classical Prony method and some equivalent procedures. In Section 3, we sketch a frequently used stable Prony-like method, namely the ESPRIT method [29] (ESPRIT = Estimation of Signal Parameters via Rotational Invariance Technique). The standard application of this Prony-like method is the solution of the problem (i) with equispaced sampled data. The case of non-equispaced sampled data is discussed in Section 4. The reconstruction of a sparse sum of eigenfunctions of a linear operator is solved by a generalized Prony method in Section 5. A solution of problem (ii) is presented in Section 6, where a spline is recovered from given Fourier data. Section 7 is devoted to the solution of problem (iii). Finally, some numerical examples are given in Section 8.

In the following we use standard notations. By \mathbb{R} resp. \mathbb{C} , we denote the set of all real resp. complex numbers. The set of all integers is \mathbb{Z} . By \mathbb{N}_0 resp. \mathbb{N} we denote the set of all nonnegative resp. positive integers. The linear space of all column vectors with N complex components is denoted by \mathbb{C}^N , where \mathbf{o} is the corresponding zero vector. The linear space of all complex M -by- N matrices is denoted by $\mathbb{C}^{M \times N}$, where $\mathbf{0}_{M,N}$ is the corresponding zero matrix. A superscript $*$ denotes conjugate transpose. For a matrix $\mathbf{A}_{M,N} \in \mathbb{C}^{M \times N}$, its Moore-Penrose pseudoinverse is denoted by $\mathbf{A}_{M,N}^\dagger$. A square matrix $\mathbf{A}_{M,M}$ is abbreviated by \mathbf{A}_M . By \mathbf{I}_M we denote the M -by- M identity matrix. Further we use the known submatrix notation. For example, $\mathbf{A}_{M,M+1}(1 : M, 2 : M+1)$ is the quadratic M -by- M submatrix of $\mathbf{A}_{M,M+1}$ obtained by extracting rows 1 through M and columns 2 through $M+1$. Note that the first row or column of $\mathbf{A}_{M,N}$ can be indexed by zero. Other notations are introduced when needed.

2 Classical Prony method

The classical Prony method works with exactly sampled data of the exponential sum (1) in the case of known order M . Following an idea of G.R. de Prony from 1795 (see [9]), we recover all parameters of the exponential sum (1), if sampled data

$$h(k) := \sum_{j=1}^M c_j e^{f_j k} = \sum_{j=1}^M c_j z_j^k, \quad k = 0, \dots, 2M - 1 \quad (2)$$

are given, where $z_j := e^{f_j}$ are distinct values in \mathbb{D} . Here $\mathbb{D} := \{z \in \mathbb{C} : e^{-\alpha} \leq |z| \leq 1\}$ denotes a circular ring. We introduce the *Prony polynomial*

$$p(z) := \prod_{j=1}^M (z - z_j) = \sum_{k=0}^{M-1} p_k z^k + z^M, \quad z \in \mathbb{C} \quad (3)$$

with corresponding coefficients p_k . Further we define the *companion matrix* $\mathbf{C}_M(p) \in \mathbb{C}^{M \times M}$ of the Prony polynomial $p(z)$ by

$$\mathbf{C}_M(p) := \begin{pmatrix} 0 & 0 & \dots & 0 & -p_0 \\ 1 & 0 & \dots & 0 & -p_1 \\ 0 & 1 & \dots & 0 & -p_2 \\ \vdots & \vdots & & \vdots & \vdots \\ 0 & 0 & \dots & 1 & -p_{M-1} \end{pmatrix}. \quad (4)$$

It is known that the companion matrix $\mathbf{C}_M(p)$ has the property

$$\det(z \mathbf{I}_M - \mathbf{C}_M(p)) = p(z), \quad z \in \mathbb{C}.$$

Hence the zeros of the Prony polynomial (3) coincide with the eigenvalues of the companion matrix $\mathbf{C}_M(p)$. Setting $p_M := 1$, we observe the following relation for all $m \in \mathbb{N}_0$,

$$\begin{aligned} \sum_{k=0}^M p_k h(k+m) &= \sum_{k=0}^M p_k \left(\sum_{j=1}^M c_j z_j^{k+m} \right) \\ &= \sum_{j=1}^M c_j z_j^m \left(\sum_{k=0}^M p_k z_j^k \right) = \sum_{j=1}^M c_j z_j^m p(z_j) = 0. \end{aligned} \quad (5)$$

Using the known values $h(k)$, $k = 0, \dots, 2M - 1$, this assertion implies that the homogeneous linear difference equation

$$\sum_{k=0}^{M-1} p_k h(k+m) = -h(M+m), \quad m = 0, \dots, M - 1 \quad (6)$$

is fulfilled. In matrix–vector notation, we obtain the linear system

$$\mathbf{H}_M(0) (p_k)_{k=0}^{M-1} = - (h(M+m))_{m=0}^{M-1} \quad (7)$$

with the *square Hankel matrix*

$$\mathbf{H}_M(0) := \begin{pmatrix} h(0) & h(1) & \dots & h(M-1) \\ h(1) & h(2) & \dots & h(M) \\ \vdots & \vdots & & \vdots \\ h(M-1) & h(M) & \dots & h(2M-2) \end{pmatrix} = (h(k+m))_{k,m=0}^{M-1}. \quad (8)$$

The matrix $\mathbf{H}_M(0)$ is invertible, since using the structure of $h(k)$ we have

$$\mathbf{H}_M(0) = \mathbf{V}_M(\mathbf{z}) (\text{diag } \mathbf{c}) \mathbf{V}_M(\mathbf{z})^T,$$

where the diagonal matrix $\text{diag } \mathbf{c}$ with $\mathbf{c} = (c_j)_{j=1}^M$ contains the nonzero coefficients of the exponential sum (1), and where $\mathbf{V}_M(\mathbf{z}) := (z_k^{j-1})_{j,k=1}^M$ denotes the *Vandermonde matrix* generated by $\mathbf{z} := (z_j)_{j=1}^M$. We summarize:

Algorithm 2.1 (*Classical Prony method*)

Input: $M \in \mathbb{N}$, sampled values $h(k)$, $k = 0, \dots, 2M - 1$, of the exponential sum (1).

1. Solve the linear system (7).
2. Compute all zeros $z_j \in \mathbb{D}$, $j = 1, \dots, M$, of the Prony polynomial (3), i.e., calculate all eigenvalues of the associated companion matrix (4), and form $f_j := \log z_j$ for $j = 1, \dots, M$, where \log is the principal value of the complex logarithm.
3. Solve the Vandermonde system

$$\mathbf{V}_M(\mathbf{z}) (c_j)_{j=1}^M = (h(k))_{k=0}^{M-1}.$$

Output: $f_j \in [-\alpha, 0] + i[-\pi, \pi)$, $c_j \in \mathbb{C}$, $j = 1, \dots, M$.

As shown, Prony's idea is mainly based on the separation of the unknown exponents f_j from the unknown coefficients c_j .

Remark 2.2 The Prony method can be also applied to the recovery of an *extended exponential sum*

$$h(x) := \sum_{j=1}^M c_j(x) e^{f_j x}, \quad x \geq 0,$$

where $c_j(x)$ are polynomials of low degree (see [1]). For simplicity, we sketch only the case of linear polynomials $c_j(x) = c_{j,0} + c_{j,1}x$. An application is described in Section 6. With distinct $z_j = e^{f_j}$, $j = 1, \dots, M$, the corresponding Prony polynomial reads as follows

$$p(z) := \prod_{j=1}^M (z - z_j)^2 = \sum_{k=0}^{2M-1} p_k z^k + z^{2M}.$$

Assuming that the sampled values $h(k)$, $k = 0, \dots, 4M - 1$ are given, one has to solve the linear system (6) (where M is replaced by $2M$) and to compute all double zeros $z_j \in \mathbb{D}$ of the above Prony polynomial (resp. all double eigenvalues of the corresponding companion

matrix (4)). Introducing the *confluent Vandermonde matrix*

$$\mathbf{V}_{2M}^c(z_1, \dots, z_M) := \begin{pmatrix} 1 & 0 & \dots & 1 & 0 \\ z_1 & 1 & \dots & z_M & 1 \\ z_1^2 & 2z_1 & \dots & z_M^2 & 2z_M \\ \vdots & \vdots & & \vdots & \vdots \\ z_1^{2M-1} & (2M-1)z_1^{2M-2} & \dots & z_M^{2M-1} & (2M-1)z_M^{2M-2} \end{pmatrix},$$

one has to solve finally the confluent Vandermonde system

$$\mathbf{V}_{2M}^c(z_1, \dots, z_M) (c_{0,1}, z_1 c_{1,1}, \dots, c_{M,0}, z_1 c_{M,1})^T = (h(k))_{k=0}^{2M-1}.$$

Remark 2.3 The Prony method is closely related to *Padé approximation* (see [33]). Let $(f_k)_{k \in \mathbb{N}_0}$ be a complex sequence with $\rho := \limsup_{k \rightarrow \infty} |f_k|^{1/k} < \infty$. The *z-transform* of such a sequence is the Laurent series $\sum_{k=0}^{\infty} f_k z^{-k}$ which converges in the neighborhood $\{z \in \mathbb{C} : |z| > \rho\}$ of $z = \infty$. Observe that the *z-transform* of each sequence $(z_j^k)_{k \in \mathbb{N}_0}$ with $z_j \in \mathbb{D}$ is equal to $\frac{z}{z-z_j}$, $j = 1, \dots, M$. Since the *z-transform* is linear, the *z-transform* maps the data sequence $(h(k))_{k \in \mathbb{N}_0} = (\sum_{j=1}^M c_j z_j^k)_{k \in \mathbb{N}_0}$ into the rational function

$$\sum_{k=0}^{\infty} h(k) z^{-k} = \sum_{j=1}^M c_j \frac{z}{z-z_j} = \frac{a(z)}{p(z)}, \quad (9)$$

where $p(z)$ is the Prony polynomial (3) and $a(z) := a_M z^M + \dots + a_1 z$. Now we substitute z for z^{-1} in (9) and form the *reverse Prony polynomial* $\text{rev } p(z) := z^M p(z^{-1})$ of degree M with $\text{rev } p(0) = 1$ as well as the reverse polynomial $\text{rev } a(z) := z^M a(z^{-1})$ of degree at least $M-1$. Then we obtain that

$$\sum_{k=0}^{\infty} h(k) z^k = \frac{\text{rev } a(z)}{\text{rev } p(z)}$$

in a certain neighborhood of $z = 0$. In other words, the rational function $\frac{\text{rev } a(z)}{\text{rev } p(z)}$ is an $(M-1, M)$ *Padé approximant* of the power series $\sum_{k=0}^{\infty} h(k) z^k$ (with vanishing $\mathcal{O}(z^{2M})$ term) and it holds

$$\left(\sum_{k=0}^{\infty} h(k) z^k \right) \text{rev } p(z) = \text{rev } a(z)$$

in a neighborhood of $z = 0$. Equating the coefficients of like powers of z yields

$$\begin{aligned} \sum_{k=M-m}^M p_k h(k+m-M) &= a_{M-m}, \quad m = 0, \dots, M-1, \\ \sum_{k=0}^M p_k h(k+m) &= 0, \quad m \in \mathbb{N}_0. \end{aligned} \quad (10)$$

Now the equations (10) for $m = 0, \dots, M-1$ coincide with (6). Hence the Prony method may also be regarded as a Padé approximation.

Remark 2.4 In signal processing, the Prony method is also known as the *annihilating filter method*, see e.g. [11, 32]. For distinct $z_j \in \mathbb{D}$ and complex coefficients $c_j \neq 0, j = 1, \dots, M$, we consider the discrete signal $\mathbf{h} = (h_n)_{n \in \mathbb{Z}}$ with

$$h_n := \sum_{j=1}^M c_j z_j^n, \quad n \in \mathbb{Z}. \quad (11)$$

For simplicity, we assume that M is known. Then a discrete signal $\mathbf{a} = (a_n)_{n \in \mathbb{Z}}$ is called an *annihilating filter* of the signal \mathbf{h} , if the discrete convolution of the signals \mathbf{a} and \mathbf{h} vanishes, i.e.

$$(\mathbf{a} * \mathbf{h})_n := \sum_{\ell=-\infty}^{\infty} a_\ell h_{n-\ell} = 0, \quad n \in \mathbb{Z}.$$

For the construction of \mathbf{a} we consider

$$a(z) := \prod_{j=1}^M (1 - z_j z^{-1}) = \sum_{n=0}^M a_n z^{-n} \quad (z \in \mathbb{C} \setminus \{0\}),$$

then $\mathbf{a} = (a_n)_{n \in \mathbb{Z}}$ with $a_n := 0$ ($n \in \mathbb{Z} \setminus \{0, \dots, M\}$) is an annihilating filter of \mathbf{h} in (11). Note that $a(z)$ is the z -transform of the annihilating filter \mathbf{a} . Furthermore, $a(z)$ and the Prony polynomial $p(z)$ in (3) have the same zeros $z_j \in \mathbb{D}, j = 1, \dots, M$, since $z^M a(z) = p(z)$ for all $z \in \mathbb{C} \setminus \{0\}$. Hence the Prony method and the method of annihilating filters are equivalent. For details see e.g. [11, 32].

Remark 2.5 Prony methods arise also from problems of science and engineering, where one is interested in predicting future information from previous ones using a linear model. Let $\mathbf{h} = (h_n)_{n \in \mathbb{N}_0}$ be a discrete signal. The *linear prediction method*, see e.g. [5, 21], aims at finding suitable predictor parameters $p_j \in \mathbb{C}$ so that the signal value $h_{\ell+M}$ can be expressed as a linear combination of the previous signal values $h_j, j = \ell, \dots, \ell + M - 1$, i.e.

$$h_{\ell+M} = \sum_{j=0}^{M-1} (-p_j) h_{\ell+j}, \quad \ell \in \mathbb{N}_0.$$

Therefore these equations are also called linear prediction equations. Setting $p_M := 1$, we observe that this representation is equivalent to the homogeneous linear difference equation (6). Assuming that

$$h_k = \sum_{j=1}^M c_j z_j^k, \quad k \in \mathbb{N}_0,$$

we obtain the problem (i), i.e., the Prony polynomial (3) coincides with the negative value of the forward predictor polynomial. The associated companion matrix $\mathbf{C}_M(p)$ in (4) is hence equal to the forward predictor matrix. Thus the linear prediction method can also be considered as a Prony method.

Unfortunately, the classical Prony method has some numerical drawbacks. Often the order M of the exponential sum (1) is unknown. Further the classical Prony method is known to perform poorly when noisy sampled data are given, since the Hankel matrix $\mathbf{H}_M(0)$ as well

as the Vandermonde matrix $\mathbf{V}_M(\mathbf{z})$ are usually badly conditioned. We will show in Section 3 that one can attenuate these problems by using more sampled data. But then one has to deal with rectangular matrices.

3 Prony-like method for equispaced sampling

In practice, the order M of the exponential sum (1) is often unknown and only noisy sampled data $h_k = h(k) + e_k, k = 0, \dots, 2N - 1$ are given. Let $L \in \mathbb{N}$ be a convenient upper bound of M and $M \leq L \leq N$. In applications, such an upper bound L of M is often known *a priori*. With the $2N$ sampled data $h_k \in \mathbb{C}, k = 0, \dots, 2N - 1$, we form the *rectangular Hankel matrix*

$$\mathbf{H}_{2N-L, L+1} := (h_{\ell+m})_{\ell, m=0}^{2N-L-1, L} \in \mathbb{C}^{(2N-L) \times (L+1)}. \quad (12)$$

For exactly sampled data, it follows from (5) that $\mathbf{H}_{2N-L, L+1}$ is rank deficient with rank M , see [28].

In the following, we sketch a frequently used Prony-like method, namely the ESPRIT method (see [29]), based on the singular value decomposition (SVD) of the rectangular Hankel matrix (12). For a detailed description of this method see Section 3 in [28].

Going back to the case of exactly sampled data for a moment, we can observe the following relations. Using the common submatrix notation, let us also consider the matrix

$$\mathbf{H}_{2N-L, L+1}(1) := (\mathbf{H}_{2N-L, L+1}(0 : 2N - L - 1, 1 : L), \mathbf{o}) \in \mathbb{C}^{(2N-L) \times (L+1)},$$

where, compared with $\mathbf{H}_{2N-L, L+1}$, the first column is removed and a zero vector \mathbf{o} is added as a last column. According to (5), we have for exact data $h_k = h(k)$

$$\mathbf{H}_{2N-L, L+1} \tilde{\mathbf{p}} = -(h_{\ell+M})_{\ell=0}^{2N-L-1}, \quad (13)$$

where $\tilde{\mathbf{p}} = (p_0, p_1, \dots, p_{M-1}, 0, \dots, 0)^T \in \mathbb{C}^{L+1}$ contains the coefficients of the Prony polynomial (3). We introduce the modified companion matrix

$$\mathbf{C}_{L+1} = \mathbf{C}_{L+1}(p) := \begin{pmatrix} \mathbf{C}_M(p) & \mathbf{0}_{M, L+1-M} \\ \mathbf{0}_{L+1-M, M} & \mathbf{V}_{L+1-M} \end{pmatrix}.$$

with $\mathbf{V}_1 := (0)$ and

$$\mathbf{V}_{L+1-M} := \begin{pmatrix} \mathbf{o}^T & 0 \\ \mathbf{I}_{L-M} & \mathbf{o} \end{pmatrix}$$

for $L > M$. Then \mathbf{C}_{L+1} possesses the zeros of $p(z)$ as eigenvalues and $L + 1 - M$ additional eigenvalues zero. By (13) we observe that

$$\mathbf{H}_{2N-L, L+1} \mathbf{C}_{L+1} = \mathbf{H}_{2N-L, L+1}(1). \quad (14)$$

Now equation (14) leads us to the following procedure. We consider the singular value factorization

$$\mathbf{H}_{2N-L, L+1} = \mathbf{U}_{2N-L} \mathbf{D}_{2N-L, L+1} \mathbf{W}_{L+1},$$

where \mathbf{U}_{2N-L} and \mathbf{W}_{L+1} are unitary matrices and where $\mathbf{D}_{2N-L,L+1}$ is a rectangular diagonal matrix. Observe that this decomposition also implies

$$\mathbf{H}_{2N-L,L+1}(1) = \mathbf{U}_{2N-L} \mathbf{D}_{2N-L,L+1} \mathbf{W}_{L+1}(1)$$

for the modified matrix $\mathbf{H}_{2N-L,L+1}(1)$, where $\mathbf{W}_{L+1}(1) := (\mathbf{W}_{L+1}(1 : L+1, 2 : L+1), \mathbf{o})$.

Let the diagonal entries of $\mathbf{D}_{2N-L,L+1}$, i.e. the singular values of $\mathbf{H}_{2N-L,L+1}$, be arranged in nonincreasing order

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_M > \sigma_{M+1} = \dots = \sigma_{L+1} = 0.$$

Thus one can determine the rank M of the Hankel matrix (12) which coincides with the order of the exponential sum (1). For noisy sampled data, we arrange the singular values of (12) in the form

$$\sigma_1 \geq \sigma_2 \geq \dots \geq \sigma_M \geq \sigma_{M+1} \geq \dots \geq \sigma_{L+1} \geq 0.$$

Then we determine the numerical rank M of $\mathbf{H}_{2N-L,L+1}$ by fixing the largest integer M with $\sigma_M/\sigma_1 \geq \varepsilon$. Depending on the noise level, there is usually an obvious gap in the singular value distribution such that ε can be suitably chosen.

Introducing the submatrices

$$\begin{aligned} \mathbf{D}_{2N-L,M} &:= \mathbf{D}_{2N-L,L+1}(1 : 2N-L, 1 : M) = \begin{pmatrix} \text{diag}(\sigma_j)_{j=1}^M \\ \mathbf{0}_{2N-L-M,M} \end{pmatrix}, \\ \mathbf{W}_{M,L+1} &:= \mathbf{W}_{L+1}(1 : M, 1 : L+1), \\ \mathbf{W}_{M,L+1}(1) &:= \mathbf{W}_{L+1}(1)(1 : M, 1 : L+1), \end{aligned}$$

we replace the original matrices $\mathbf{H}_{2N-L,L+1}$ and $\mathbf{H}_{2N-L,L+1}(1)$ above by $\tilde{\mathbf{H}}_{2N-L,L+1}$ resp. $\tilde{\mathbf{H}}_{2N-L,L+1}(1)$ with exact rank M ,

$$\begin{aligned} \tilde{\mathbf{H}}_{2N-L,L+1} &= \mathbf{U}_{2N-L} \mathbf{D}_{2N-L,M} \mathbf{W}_{M,L+1}, \\ \tilde{\mathbf{H}}_{2N-L,L+1}(1) &= \mathbf{U}_{2N-L} \mathbf{D}_{2N-L,M} \mathbf{W}_{M,L+1}(1). \end{aligned}$$

Hence (14) implies

$$\mathbf{D}_{2N-L,M} \mathbf{W}_{M,L+1} \mathbf{C}_{L+1} = \mathbf{D}_{2N-L,M} \mathbf{W}_{M,L+1}(1).$$

Multiplying the conjugate transposed equation with $(\mathbf{D}_{2N-L,M}^*)^\dagger$, it follows that

$$\mathbf{C}_{L+1}^* \mathbf{W}_{M,L+1}^* = \mathbf{W}_{M,L+1}(1)^*.$$

Setting

$$\mathbf{W}_{M,L}(s) := \mathbf{W}_{M,L+1}(1 : M, 1 + s : L + s), \quad s = 0, 1, \quad (15)$$

we remove the zero columns in the last equation and arrive at

$$\mathbf{C}_L^* \mathbf{W}_{M,L}(0)^* = \mathbf{W}_{M,L}(1)^*$$

with $\mathbf{C}_L := \mathbf{C}_{L+1}(1 : L, 1 : L)$. Since $\text{rank}(\mathbf{C}_L) = M$, the nonzero eigenvalues of the modified companion matrix \mathbf{C}_L are now equal to the eigenvalues of the square matrix

$$\mathbf{F}_M := (\mathbf{W}_{M,L}(0)^*)^\dagger \mathbf{W}_{M,L}(1)^*, \quad (16)$$

where $(\mathbf{W}_{M,L}(0)^*)^\dagger$ denotes the Moore–Penrose pseudoinverse of $\mathbf{W}_{M,L}(0)^*$. Thus we can determine the wanted nodes $z_j \in \mathbb{D}$, $j = 1, \dots, M$, as eigenvalues of the matrix \mathbf{F}_M . Hence the ESPRIT algorithm reads as follows:

Algorithm 3.1 (*ESPRIT method for equispaced sampling*)

Input: $L, N \in \mathbb{N}$, $N \gg 1$, $3 \leq L \leq N$, L is upper bound of the order M of the exponential sum (1), noisy sampled values h_k , $k = 0, \dots, 2N - 1$, of the exponential sum (1).

1. Compute the SVD of the rectangular Hankel matrix (12). Determine the numerical rank of (12) as largest integer M with $\sigma_M/\sigma_1 \geq \varepsilon$ and form the matrices (15).
2. Compute all eigenvalues $z_j \in \mathbb{D}$, $j = 1, \dots, M$, of the square matrix (16) and evaluate $f_j := \log z_j$, $j = 1, \dots, M$.
3. Compute the coefficients $c_j \in \mathbb{C}$, $j = 1, \dots, M$, as least squares solution of the overdetermined linear Vandermonde–like system

$$\mathbf{V}_{2N,M}(\mathbf{z}) (c_j)_{j=1}^M = (h_k)_{k=0}^{2N-1}$$

with the rectangular Vandermonde matrix $\mathbf{V}_{2N,M}(\mathbf{z}) := (z_j^{k-1})_{k,j=1}^{2N,M}$.

Output: $M \in \mathbb{N}$, $f_j \in [-\alpha, 0] + i[-\pi, \pi]$, $c_j \in \mathbb{C}$, $j = 1, \dots, M$.

Remark 3.2 For various numerical examples as well as for a comparison between Algorithm 3.1 and another Prony–like method see [26]. The Algorithm 3.1 is very similar to the Algorithm 3.2 in [28]. Note that one can also use the QR decomposition of the rectangular Hankel matrix (12) instead of the SVD. In that case one obtains an algorithm that is similar to the matrix pencil method [18, 31], see also Algorithm 3.1 in [28]. The matrix pencil method has been also applied to reconstruction of shapes from moments, see e.g. [13].

In [4], the condition number of a rectangular Vandermonde matrix is estimated. It is shown that this matrix is well conditioned, provided the nodes z_j are close to the unit circle, but not extremely close to each other and provided N is large enough.

Remark 3.3 The given data sequence $\{h_0, h_1, \dots, h_{2N-1}\}$ can be also interpreted as *time series*. A powerful tool of time series analysis is the *singular spectrum analysis* (see [14, 15]). Similarly as step 1 of the Algorithm 3.1, this technique is based on the singular value decomposition of a rectangular Hankel matrix constructed upon the given time series h_k . By this method, the original time series can be decomposed into a sum of interpretable components such as trend, oscillatory components, and noise. For further details and numerous applications see [14, 15].

Remark 3.4 The considered Prony–like method can also be interpreted as a model reduction based on low–rank approximation of Hankel matrices, see [22]. The *structured low–rank approximation problem* reads as follows: For a given structure specification $\mathcal{S} : \mathbb{C}^K \rightarrow \mathbb{C}^{L \times N}$ with $L < N$, a parameter vector $\mathbf{h} \in \mathbb{C}^K$ and an integer M with $0 < M < L$, find a vector

$$\hat{\mathbf{h}}^* = \underset{\hat{\mathbf{h}}}{\text{argmin}} \|\mathbf{h} - \hat{\mathbf{h}}\| \quad \text{subject to} \quad \text{rank}(\mathcal{S}(\hat{\mathbf{h}})) \leq M,$$

where $\|\cdot\|$ denotes a suitable norm in \mathbb{C}^K . In the special case of a Hankel matrix structure, the Hankel matrix $\mathcal{S}(\mathbf{h}) = (h_{\ell+k})_{\ell=0, k=0}^{L-1, N-1}$ is rank-deficient of order M if there exists a nonzero vector $\mathbf{p} = (p_k)_{k=0}^{M-1}$ so that

$$\sum_{k=0}^{M-1} p_k h(m+k) = -h(M+m)$$

for all $m = 0, \dots, N+L-M-1$. Equivalently, the values $h(k)$ can be interpreted as function values of an exponential sum of order M in (1). The special kernel structure of rank-deficient Hankel matrices can already be found in [17].

4 Prony-like method for nonequispaced sampling

In the following we generalize the Prony-like method to the case of nonequispaced sampled data. More precisely, as in Sections 2 and 3 we recover all parameters of the exponential sum (1) of order M , but now we assume that the sampled data $h(x_k)$ are given at nonequispaced distinct nodes

$$0 \leq x_0 < x_1 < \dots < x_{2N-2} < x_{2N-1} = 2L-1, \quad M \leq L \leq N. \quad (17)$$

While the Prony method is reliant on equispaced data, and sensitive to data errors, this problem is quite delicate. Fortunately, the exponential sum (1) is a smooth function with a moderate oscillatory part with bounded frequency. Note that a Prony-like method for nonequispaced sampling was already proposed in [7]. There the unknown parameters of the exponential sum (1) were estimated by a linear regression equation using filtered signals.

Assuming that the given data are (almost) exact, we can apply another technique based on interpolation and we compute approximate values h_j of $h(j)$, $j = 0, \dots, 2L-1$ in a preprocessing step. This can be done by one of the following procedures.

The first method is based on interpolation with integer translates of a window function. Let M_{2m} be the centered cardinal B-spline of order $2m$ with support $[-m, m]$, where $m \in \mathbb{N}$ and $1 < m \leq N-L+1$. Then we approximate the exponential sum (1) by a linear combination of integer translates of M_{2m} ,

$$g(x) := \sum_{\ell=1-m}^{2L+m-2} g_\ell M_{2m}(x-\ell).$$

For this purpose, we compute the coefficients g_ℓ as (least squares) solution of the (overdetermined) sparse linear system

$$\sum_{\ell=-m+1}^{2L+m-2} g_\ell M_{2m}(x_k - \ell) = h(x_k), \quad k = 0, \dots, 2N-1.$$

We assume here that the coefficient matrix $(M_{2m}(x_k - \ell))_{k=0, \ell=-m+1}^{2N-1, 2L+m-2}$ has full rank $2(L+m-1)$. By the Schoenberg-Whitney theorem on spline interpolation this is true if there is

a subsequence $\{x_{k_\ell}\}_{\ell=-m+1}^{2L+m-2}$ of $\{x_k\}_{k=0}^{2N-1}$ so that $M_{2m}(x_{k_\ell} - \ell) \neq 0$ for all $\ell = -m + 1, \dots, 2L + m - 2$, see [8]. Then we set $h_k := g(k)$, $k = 0, \dots, 2L - 1$.

The second method is based on piecewise cubic polynomial interpolation. Let p_k , $k = 1, \dots, 2N - 3$ be cubic interpolation polynomials being uniquely determined by

$$p_k(x_j) = h(x_j), \quad j = k - 1, \dots, k + 2.$$

Then each polynomial p_k has the form

$$p_k(x) = h(x_k) + a_k(x - x_k) + b_k(x - x_k)^2 + c_k(x - x_k)^3,$$

where the coefficients a_k, b_k, c_k solve the linear system

$$\begin{aligned} a_k + b_k(x_{k-1} - x_k) + c_k(x_{k-1} - x_k)^2 &= \frac{h(x_{k-1}) - h(x_k)}{x_{k-1} - x_k}, \\ a_k + b_k(x_{k+1} - x_k) + c_k(x_{k+1} - x_k)^2 &= \frac{h(x_{k+1}) - h(x_k)}{x_{k+1} - x_k}, \\ a_k + b_k(x_{k+2} - x_k) + c_k(x_{k+2} - x_k)^2 &= \frac{h(x_{k+2}) - h(x_k)}{x_{k+2} - x_k}. \end{aligned}$$

We set for each $j = 0, \dots, 2L - 1$,

$$h_j := \begin{cases} p_1(j) & j \in [0, x_2), \\ p_k(j) & j \in [x_k, x_{k+1}) \text{ for } k \in \{2, \dots, 2N - 4\}, \\ p_{2N-3}(j) & j \in [x_{2N-3}, 2L - 1]. \end{cases}$$

Observe that in this way not each polynomial p_k will be used. For $N \gg L$ one may therefore replace the procedure by a local cubic polynomial approximation that involves more than four data points.

Algorithm 4.1 (*ESPRIT method for nonequispaced sampling*)

Input: $L, N \in \mathbb{N}$, $N \gg 1$, $3 \leq L \leq N$, L is an upper bound of the order M of the exponential sum (1), nonequispaced sampling nodes (17), sampled data $h(x_k)$, $k = 0, \dots, 2N - 1$ of the exponential sum (1).

1. Precompute the approximate values h_k of $h(k)$ for $k = 0, \dots, 2L - 1$ by one of the above methods.

2. Use the Algorithm 3.1 (with $L = N$) in order to determine the order M , all exponents f_j and all coefficients c_j , $j = 1, \dots, M$, of the exponential sum (1).

Output: $M \in \mathbb{N}$, $f_j \in [-\alpha, 0] + i[-\pi, \pi)$, $c_j \in \mathbb{C}$, $j = 1, \dots, M$.

5 Generalized Prony method

The Prony method can be simply transferred to a more general setting, namely to recover an element f of a given complex vector space \mathcal{V} , if f can be represented as an M -sparse expansion of eigenfunctions of a linear operator $\mathcal{A} : \mathcal{V} \rightarrow \mathcal{V}$. More precisely, let $W = \{v_j : j \in I\}$ be a given set of eigenfunctions (resp. eigenvectors) of \mathcal{A} to distinct eigenvalues λ_j , i.e.,

$$\mathcal{A}v_j = \lambda_j v_j$$

for all $j \in I$. Further, let $F : \mathcal{V} \rightarrow \mathbb{C}$ be a linear functional with $Fv_j \neq 0$ for all $j \in I$. Then, according to [24], an M -sparse representation

$$h = \sum_{j \in J} c_j v_j \quad \text{with } J \subset I \text{ and } |J| = M \quad (18)$$

can be uniquely reconstructed from the values $F(\mathcal{A}^k h)$, $k = 0, \dots, 2M - 1$, i.e., the active eigenfunctions v_j as well as the coefficients $c_j \in \mathbb{C}$, $j \in J$ in (18) can be uniquely determined, see [24]. Similarly as in Section 2 this can be seen as follows.

We define the Prony polynomial

$$p(z) := \prod_{j \in J} (z - \lambda_j) = \sum_{k=0}^{M-1} p_k z^k + z^M, \quad z \in \mathbb{C}, \quad (19)$$

where the roots λ_j , $j \in J$, are the (unknown) eigenvalues corresponding to the active eigenfunctions v_j in the representation of h . With $p_M := 1$, we observe for $m \in \mathbb{N}_0$ that

$$\begin{aligned} \sum_{k=0}^M p_k F(\mathcal{A}^{k+m} h) &= \sum_{k=0}^M p_k F\left(\sum_{j \in J} c_j \lambda_j^{k+m} v_j\right) = \sum_{j \in J} c_j \lambda_j^m \left(\sum_{k=0}^M p_k \lambda_j^k\right) Fv_j \\ &= \sum_{j \in J} c_j \lambda_j^m p(\lambda_j) Fv_j = 0, \end{aligned}$$

yielding the linear system

$$\sum_{k=0}^{M-1} p_k F(\mathcal{A}^{k+m} h) = -F(\mathcal{A}^{M+m} h), \quad m = 0, \dots, M-1.$$

As for the classical case, the coefficient matrix $\mathbf{G}_M := (F(\mathcal{A}^{k+m} h))_{k=0, m=0}^{M-1, M-1}$ is an invertible Hankel matrix, since

$$\mathbf{G}_M = \mathbf{V}_M(\lambda) \text{diag}(c_j)_{j \in J} \text{diag}(Fv_j)_{j \in J} \mathbf{V}_M(\lambda)^\top$$

with $\mathbf{V}_M(\lambda) := (\lambda_j^k)_{k=0, j \in J}^{M-1}$. Hence the algorithm for the generalized Prony method can be summarized as follows.

Algorithm 5.1 (*Generalized Prony method*)

Input: $M \in \mathbb{N}$, values $F(\mathcal{A}^k h)$, $k = 0, \dots, 2M - 1$, of the sparse eigenfunction expansion (18).

1. Solve the linear system $\mathbf{G}_M(p_k)_{k=0}^{M-1} = -(F(\mathcal{A}^{M+m} h))_{m=0}^{M-1}$.
2. Compute all zeros λ_j , $j = 1, \dots, M$, of the Prony polynomial (19), i.e., calculate all eigenvalues of the associated companion matrix (4).
3. Solve the generalized Vandermonde system

$$(\lambda_j^k)_{k=0, j \in J}^{2M-1} \text{diag}(Fv_j)_{j \in J} (c_j)_{j \in J} = (F(\mathcal{A}^k h))_{k=0}^{2M-1}.$$

Output: $v_j \in W$, $c_j \in \mathbb{C}$, $j \in J$.

Assume now that the number M of terms in the eigenfunction representation of h is not known *a priori* and can only be bounded by L while the data $F(\mathcal{A}^k h) \in \mathbb{C}$, $k = 0, \dots, 2N-1$ are given with $M \leq L \leq N$. Then we can follow the lines of Section 3 to derive an ESPRIT algorithm for the generalized Prony method by replacing $\mathbf{H}_{2N-L, L+1}$ by $\mathbf{G}_{2N-L, L+1} := (F(\mathcal{A}^{\ell+m} h))_{\ell=0, m=0}^{2N-L-1, L}$ and regarding its singular value decomposition. The matrix \mathbf{F}_M as in (16) has to be computed similarly as before. The last step in Algorithm 3.1 needs to be replaced by solving the overdetermined linear system

$$F(\mathcal{A}^k h) = \sum_{j \in J} c_j \lambda_j^k F(v_j), \quad k = 0, \dots, 2N-1.$$

Remark 5.2 The classical Prony method in Section 2 follows from the generalized Prony method for example by taking $\mathcal{V} = C(\mathbb{R})$, the vector space of continuous functions, and with the shift operator $\mathcal{A} = \mathcal{S} : C(\mathbb{R}) \rightarrow C(\mathbb{R})$, given by $\mathcal{S}h := h(\cdot + 1)$. Then the circular ring \mathbb{D} defined in Section 2 is a set of distinct eigenvalues of \mathcal{S} . Indeed, for $f \in [-\alpha, 0] + i[-\pi, \pi)$ with $\alpha > 0$ we have $e^f \in \mathbb{D}$ and

$$\mathcal{S}e^{fx} = e^{f(x+1)} = e^f e^{fx}, \quad x \in \mathbb{R}.$$

Thus we can choose $W = \{e^{fx} : e^f \in \mathbb{D}\}$ as the set of corresponding eigenfunctions. Hence the exponential sum (1) with unknowns $e^{f_j} \in \mathbb{D}$ and $c_j \in \mathbb{C}$ can be completely recovered using $F(\mathcal{S}^k h) = F(h(\cdot + k))$, $k = 0, \dots, 2M-1$. Taking the point functional $F(h) := h(0)$, we obtain the classical Prony method, where the unknowns f_j and c_j , $j = 1, \dots, M$, can be reconstructed from given equispaced sampled data $h(k)$, $k = 0, \dots, 2M-1$.

Remark 5.3 As shown in [24], the generalized Prony method can be applied to a lot of eigenfunction systems as e.g. to monomials being eigenfunctions of the dilatation operator $\mathcal{A} = \mathcal{D}_a$ with $\mathcal{D}_a h(x) := h(ax)$ thereby generalizing the sparse interpolation of polynomials (see [19]), and to orthogonal polynomials that are eigenfunctions of the Sturm–Liouville operator or of special difference operators (see also [25]). The approach also applies to finite dimensional vector spaces, where one can derive a deterministic reconstruction method for M -sparse vectors from only $2M$ “measurements”, see Section 8. The required measurements are generally of the form $F(\mathcal{A}^k h)$ and hence depend strongly on the used operator \mathcal{A} as well as on the used functional F . Particularly, the freedom in choosing F enables us to solve the reconstruction with different input data.

6 Recovery of splines from Fourier data

The reconstruction of a compactly supported, structured function from the knowledge of samples of its Fourier transform is a common problem in several scientific areas such as radio astronomy, computerized tomography, and magnetic resonance imaging. Here we determine the breakpoints and the jump magnitudes of a spline s from few equidistant samples of the Fourier transform \hat{s} . Let $-\infty < t_1 < t_2 < \dots < t_{m+1} < \infty$ be given. A function $s : \mathbb{R} \rightarrow \mathbb{R}$ is a *spline of degree* $d \in \mathbb{N}_0$ with the support $[t_1, t_{m+1}]$ and breakpoints t_j , $j = 1, \dots, m+1$, if $s(x) = 0$ for all $x \in (-\infty, t_1) \cup (t_{m+1}, \infty)$ and if on each interval (t_j, t_{j+1}) it is a polynomial of degree $\leq d$ and at least on one of them of degree d . Thus, splines of degree 0 are step functions, those of degree 1 are piecewise linear.

First we consider the step function (see [27])

$$s(x) := \sum_{j=1}^m s(t_j+) \mathbf{1}_{[t_j, t_{j+1})}(x), \quad x \in \mathbb{R}, \quad (20)$$

where $\mathbf{1}_{[t_j, t_{j+1})}(x)$ is the characteristic function of the interval $[t_j, t_{j+1})$ and where $s(t_j+) - s(t_j-) \neq 0, j = 1, \dots, m+1$. Here $s(t_j+)$ and $s(t_j-)$ denote the one-sided limits of $s(x)$ at the breakpoint t_j . Forming the Fourier transform

$$\hat{s}(v) := \int_{-\infty}^{\infty} s(x) e^{-ixv} dx, \quad v \in \mathbb{R},$$

we obtain that

$$iv \hat{s}(v) = \sum_{j=1}^{m+1} (s(t_j+) - s(t_j-)) e^{-ivt_j}, \quad v \in \mathbb{R}$$

with $s(t_{m+1}+) := 0, s(t_1-) := 0$, and $s(t_j-) := s(t_{j-1}+)$ for $j = 2, \dots, m+1$. Hence $iv \hat{s}(v)$ is an exponential sum of order $m+1$. Now we choose $\tau > 0$ so that $x_j \tau \in [-\pi, \pi)$ for $j = 1, \dots, m+1$. For given Fourier samples $\hat{s}(\ell\tau), \ell = 1, \dots, N$ with $N \geq m+1$, we can determine the breakpoints t_j and the associated jump magnitudes $c_j = s(t_j+) - s(t_j-)$ by Algorithm 3.1. Observe that here indeed $m+1$ Fourier samples corresponding to the $m+1$ unknown knots are sufficient for complete recovery, since the function s is real. By $\hat{s}(v) = \hat{s}(-v)$, the values $(i\ell\tau) \hat{s}(\ell\tau)$ are available for $\ell = -N, \dots, N$, and Algorithm 3.1 can be simply adapted to this case. Hence, the step function (20) can be completely reconstructed (see [27]).

Now we consider the (not necessarily continuous) spline of degree 1,

$$s(x) := \sum_{j=1}^m (s(t_j+) + s'(t_j+) (x - t_j)) \mathbf{1}_{[t_j, t_{j+1})}(x), \quad x \in \mathbb{R}. \quad (21)$$

Applying Fourier transform, we obtain that

$$(iv)^2 \hat{s}(v) = \sum_{j=1}^{m+1} [iv (s(t_j+) - s(t_j-)) + (s'(t_j+) - s'(t_j-))] e^{-ivt_j}, \quad v \in \mathbb{R}.$$

Hence $(iv)^2 \hat{s}(v)$ is an extended exponential sum with linear polynomials as coefficients. Now we choose $\tau > 0$ so that $x_j \tau \in [-\pi, \pi)$ for $j = 1, \dots, m+1$. For given Fourier samples $\hat{s}(\ell\tau), \ell = 1, \dots, 2N$ with $N \geq m+1$, we can determine the breakpoints t_j and the associated jump magnitudes of $s(x)$ and $s'(x)$ by the method explained in Remark 2.2. Hence, the spline of degree 1 in (21) can also be reconstructed by a Prony method.

In [27], the above approach is transferred to the reconstruction of a linear combination

$$s(x) = \sum_{j=1}^m c_j B_j^d(x), \quad x \in \mathbb{R},$$

of B-splines $B_j^d(x)$ of degree $d - 1$, $d \in \mathbb{N}$, where the coefficients $c_j \in \mathbb{R}$ and the knots t_j, \dots, t_{j+d} with $-\infty < t_1 < t_2 < \dots < t_{m+d} < \infty$ are unknown. Note that $B_j^1(x) = \mathbf{1}_{[t_j, t_{j+1})}(x)$. In this case, the reconstruction is based on the idea that the $(d - 1)$ -th derivative of B_j^d is a linear combination of step functions of the form (20), and that the $m + d$ Fourier samples $\hat{s}(\ell\tau)$, $\ell = 1, \dots, m + d$ are sufficient for complete reconstruction.

Remark 6.1 A similar technique can be applied, if the support $[t_1, t_{m+1}]$ of the spline $s(x)$ is contained in $[-\pi, \pi]$ and some Fourier coefficients

$$c_k(s) := \frac{1}{2\pi} \int_{-\pi}^{\pi} s(x) e^{-ikx} dx, \quad k \in \mathbb{Z}$$

are given. For a step function (20) we obtain

$$2\pi ik c_k(s) = \sum_{j=1}^{m+1} (s(t_j+) - s(t_j-)) e^{-ikt_j}, \quad k \in \mathbb{Z}.$$

Thus one can determine the breakpoints t_j and the associated jump magnitudes by the Algorithm 3.1 and reconstruct the step function (20) using only the Fourier coefficients $c_k(s)$, $k = 1, \dots, m + 1$.

Remark 6.2 The method of Remark 6.1 is closely related to the Krylov–Lanczos method of accelerating convergence of Fourier expansions [20] and to the reconstruction of a 2π -periodic, piecewise C^d -smooth function from given Fourier data (see [12, 2, 3]). A 2π -periodic function $f(x)$ is called *piecewise C^d -smooth* with $d \in \mathbb{N}$, if there exist finitely many points t_j , $j = 1, \dots, m$, with $-\pi \leq t_1 < t_2 < \dots < t_m < \pi$ and $t_{m+1} := t_1 + 2\pi$ so that $f(x)$ restricted to (t_j, t_{j+1}) belongs to $C^d([t_j, t_{j+1}])$ for each $j = 1, \dots, m$. By $C^d([t_j, t_{j+1}])$ we mean the set of all functions $g(x)$ whose derivatives up to the order d are continuous on (t_j, t_{j+1}) and have continuous extensions on $[t_j, t_{j+1}]$, i.e., there exist all one-sided limits $g^{(\ell)}(t_j+)$ and $g^{(\ell)}(t_{j+1}-)$ for $\ell = 0, \dots, d$.

If the 2π -periodic, piecewise C^d -smooth function $f(x)$ possesses only one breakpoint $t_1 = 0$ within $[-\pi, \pi)$, i.e. $m = 1$, then by the *Krylov–Lanczos method* $f(x)$ is split into the sum

$$f(x) = s(x) + r(x), \quad x \in \mathbb{R}, \quad (22)$$

where

$$s(x) := \sum_{\ell=0}^d (f^{(\ell)}(0-) - f^{(\ell)}(0+)) \frac{(2\pi)^\ell}{(\ell+1)!} b_{\ell+1}\left(\frac{x}{2\pi}\right), \quad x \in \mathbb{R},$$

is a 2π -periodic spline of degree $d + 1$ and $r(x)$ is a 2π -periodic, d -times continuously differentiable function with a rapidly convergent Fourier expansion (see [20]). Note that $f^{(\ell)}(0-) = f^{(\ell)}(2\pi-)$, $\ell = 0, \dots, d$, by periodicity. By $b_{\ell+1}(x)$ with $\ell \in \mathbb{N}$, we denote the 1-periodic continuation of the Bernoulli polynomial $B_{\ell+1}(x)$ restricted to the interval $[0, 1]$, where $B_{\ell+1}(x)$ is recursively defined by $B'_{\ell+1}(x) = (\ell + 1) B_\ell(x)$, $\int_0^1 B_{\ell+1}(x) dx = 0$, $\ell \in \mathbb{N}_0$, with $B_0(x) := 1$. Note that $b_1(x)$ is the 1-periodic continuation of $B_1(x) = x - \frac{1}{2}$ restricted on $(0, 1)$ with $b_1(0) = b_1(1) := 0$.

If the 2π -periodic, piecewise C^d -smooth function $f(x)$ possesses some breakpoints t_j , $j = 1, \dots, m$, within $[-\pi, \pi)$, then $f(x)$ can be represented analogously in the form (22) with the 2π -periodic spline of degree $d + 1$

$$s(x) := \sum_{j=1}^m \sum_{\ell=0}^d (f^{(\ell)}(t_{j-}) - f^{(\ell)}(t_{j+})) \frac{(2\pi)^\ell}{(\ell+1)!} b_{\ell+1} \left(\frac{x - t_j}{2\pi} \right), \quad x \in \mathbb{R},$$

and a 2π -periodic, d -times continuously differentiable function $r(x)$ with a rapidly convergent Fourier expansion (see [12, 2, 3]). If $r(x)$ is sufficiently smooth, then the contribution of $c_k(r)$ to $c_k(f)$ is negligible for large $|k|$. The reconstruction of $f(x)$ from given Fourier coefficients $c_k(f)$ is based on the fact that $c_k(f) \approx c_k(s)$ for large $|k|$ and that the Fourier coefficients $c_k(s)$ can be computed using

$$c_k \left(b_{\ell+1} \left(\frac{\cdot - t_j}{2\pi} \right) \right) = \begin{cases} -\frac{(\ell+1)!}{(2\pi i k)^{\ell+1}} e^{-i k t_j} & k \in \mathbb{Z} \setminus \{0\}, \\ 0 & k = 0 \end{cases}$$

so that

$$2\pi (i k)^{d+1} c_k(s) = \sum_{j=1}^m e^{-i k t_j} \sum_{\ell=0}^d (i k)^{d-\ell} (f^{(\ell)}(t_{j+}) - f^{(\ell)}(t_{j-})), \quad k \in \mathbb{Z}.$$

Hence the points t_j and the associated jump magnitudes can be determined by Prony methods. For details see [12, 2, 3].

7 Recovery of sparse vectors

Now we consider the recovery problem of M -sparse vectors. Let $\mathbf{x} \in \mathbb{C}^D$ be M -sparse, i.e., only M components of $\mathbf{x} = (x_j)_{j=0}^{D-1}$ are different from zero. We want to reconstruct \mathbf{x} from only $2N$ measurements, where L with $L \leq N$ is a known upper bound of the sparsity M . For this purpose we apply the generalized Prony method introduced in Section 5. Let $\mathbf{A} : \mathbb{C}^D \rightarrow \mathbb{C}^D$ be a linear operator that can be represented by a diagonal matrix $\mathbf{A} = \text{diag}(d_j)_{j=0}^{D-1}$ with distinct entries d_j , $j = 0, \dots, D-1$. Then the unit vectors $\mathbf{e}_j = (\delta_{j,\ell})_{\ell=0}^{D-1}$, $j = 0, \dots, D-1$, form a system of eigenvectors of \mathbf{A} with $\mathbf{A} \mathbf{e}_j = d_j \mathbf{e}_j$. Further, let $F : \mathbb{C}^D \rightarrow \mathbb{C}$ be a linear functional given by

$$F \mathbf{x} = \mathbf{b}^T \mathbf{x} := \sum_{j=0}^{D-1} b_j x_j,$$

where $\mathbf{b} = (b_j)_{j=0}^{D-1}$ satisfies $b_j \neq 0$ for $j = 0, \dots, D-1$. Hence, the condition $F \mathbf{e}_j \neq 0$ holds for all $j = 0, \dots, D-1$.

In order to reconstruct a sparse vector \mathbf{x} of the form

$$\mathbf{x} = \sum_{j=1}^M c_{n_j} \mathbf{e}_{n_j}$$

with unknown support indices $0 \leq n_1 < n_2 < \dots < n_M \leq D - 1$ and unknown coefficients $c_{n_j} \in \mathbb{C}$, we require by Algorithm 5.1 at least the values

$$y_k = F(\mathbf{A}^k \mathbf{x}) = \mathbf{b}^T \mathbf{A}^k \mathbf{x} = \mathbf{a}_k^T \mathbf{x}, \quad k = 0, \dots, 2M - 1, \quad (23)$$

where $\mathbf{a}_k := (b_j d_j^k)_{j=0}^{D-1}$.

For noisy measurements, the reconstruction is more stable, if we adapt the ESPRIT method also here. This is particularly necessary if the exact sparsity M is unknown and we know only an upper bound $L \geq M$.

Algorithm 7.1 (*ESPRIT method for recovery of a sparse vector*)

Input: $L, N \in \mathbb{N}$, $L \leq N$, L is an upper bound of the sparsity M of \mathbf{x} , (noisy) sampled values y_k , $k = 0, \dots, 2N - 1$ as in (23).

1. Compute the SVD of the rectangular Hankel matrix $\mathbf{H}_{2N-L, L+1} = (y_{\ell+m})_{\ell, m=0}^{2N-L-1, L}$. Determine the numerical rank of $\mathbf{H}_{2N-L, L+1}$ as largest integer M with $\sigma_M / \sigma_1 \geq \varepsilon$ and form the matrices $\mathbf{W}_{M, L}(0)$ and $\mathbf{W}_{M, L}(1)$ as in (15).
2. Compute all eigenvalues λ_j , $j = 1, \dots, M$, of the square matrix \mathbf{F}_M in (16). The set of eigenvalues $\{\lambda_1, \dots, \lambda_M\}$ is a subset of the set of eigenvalues $\{d_0, \dots, d_{D-1}\}$ of \mathbf{A} . Determine the corresponding eigenvectors \mathbf{e}_{n_j} , $j = 1, \dots, M$, resp. the indices n_j that correspond to the eigenvalues $\lambda_1, \dots, \lambda_M$.
3. Compute the coefficients $c_{n_j} \in \mathbb{C}$, $j = 1, \dots, M$, as least squares solution of the overdetermined linear system

$$\sum_{j=1}^M c_{n_j} b_{n_j} d_{n_j}^k = y_k, \quad k = 0, \dots, 2N - 1.$$

Output: $M \in \mathbb{N}$, $\mathbf{x} = \sum_{j=1}^M c_{n_j} \mathbf{e}_{n_j}$.

For example, the linear operator \mathbf{A} can be chosen as

$$\mathbf{A} = \text{diag}(\omega_D^j)_{j=0}^{D-1},$$

where $\omega_D := e^{-2\pi i/D}$ denotes the D -th root of unity. Further, taking the functional F of the form $F\mathbf{x} = \sum_{\ell=0}^{D-1} x_\ell$, the needed vector of input values $\mathbf{y} = (y_k)_{k=0}^{2N-1}$ for Algorithm 7.1 is given by

$$\mathbf{y} = \mathbf{F}_{2N, D} \mathbf{x},$$

where $\mathbf{F}_{2N, D} = (\omega_D^{k\ell})_{k, \ell=0}^{2N-1, D-1} \in \mathbb{C}^{2N \times D}$ contains the first $2N$ rows of the Fourier matrix of order D , where $N \geq M$. In other words, the knowledge of the first $2M$ DFT coefficients of \mathbf{x} is sufficient to recover the M -sparse vector \mathbf{x} .

Remark 7.2 In contrast to l_1 -recovery algorithms in compressed sensing [10] based on structured matrices, where the measurements are obtained using scalar products of \mathbf{x} with at least $\mathcal{O}(M \log D)$ random rows of the Fourier matrix [30], we have just taken here the measurements y_k arising from the first $2N$ rows of the Fourier matrix. As proposed in [16], it might be advantageous to use slightly different measurements instead. Obviously, we can also choose the operator \mathbf{A} of the form

$$\mathbf{A} = \text{diag}(\omega_D^{\sigma j})_{j=0}^{D-1},$$

where $\sigma \in \mathbb{N}$ is invertible modulo D so that all eigenvalues of \mathbf{A} are indeed distinct. Further, we can define the functional F using the τ -th row of the Fourier matrix, i.e., $F\mathbf{x} = (\omega_D^0, \omega_D^\tau, \dots, \omega_D^{\tau(D-1)})\mathbf{x} = \sum_{\ell=0}^{D-1} \omega_D^{\tau\ell} x_\ell$, where $\tau \in \{0, \dots, D-1\}$. Then the input values for Algorithm 7.1 are of the form

$$y_k = \hat{x}_{\sigma k + \tau} = \sum_{\ell=0}^{D-1} x_\ell \omega_D^{(\sigma k + \tau)\ell}, \quad k = 0, \dots, 2N-1,$$

and the reconstruction of \mathbf{x} from y_k , $k = 0, \dots, 2N-1$ is ensured for $N \geq M$.

8 Numerical examples

We illustrate the behavior of the proposed Algorithms 3.1 and 7.1. Using IEEE standard floating point arithmetic with double precision, we have implemented these algorithms in Matlab.

Example 8.1 We consider the parameter reconstruction of the exponential sum (1) with $M = 6$, $c_j = j$, $z_j = e^{f_j}$ for $j = 1, \dots, 6$, where

$$(z_j)_{j=1}^6 = \begin{pmatrix} 0.9856 - 0.1628i \\ 0.9856 + 0.1628i \\ 0.8976 - 0.4305i \\ 0.8976 + 0.4305i \\ 0.8127 - 0.5690i \\ 0.8127 + 0.5690i \end{pmatrix}.$$

We choose noisy sampled data $h_k = h(k) + e_k$ for $k = 0, \dots, 2N-1$, where the error terms $e_k \in [-1, 1] \cdot 10^{-\delta}$ are uniformly distributed. In the case $\delta = \infty$, we consider exact sampled data. By \tilde{f}_j resp. \tilde{c}_j we denote the exponents resp. coefficients computed by Algorithm 3.1. The errors of Algorithm 3.1 are measured by

$$e(\mathbf{f}) := \frac{\max_{j=1, \dots, 6} |f_j - \tilde{f}_j|}{\max_{j=1, \dots, 6} |f_j|}, \quad e(\mathbf{c}) := \frac{\max_{j=1, \dots, 6} |c_j - \tilde{c}_j|}{\max_{j=1, \dots, 6} |c_j|}$$

with $\mathbf{f} := (f_j)_{j=1}^6$ and $\mathbf{c} := (c_j)_{j=1}^6$. For noisy data with $\delta < \infty$ we show in Table 1 the average errors over 10 runs of Algorithm 3.1 (with different data realizations for fixed δ). Using Algorithm 3.1, we obtain excellent parameter reconstructions for relatively few sampled data (without preprocessing of the data). For further applications of Algorithm 3.1 see [28].

Example 8.2 We consider the recovery of a sparse vector from (noisy) samples of its Fourier transformed data. We use Algorithm 7.1 with the operator matrix $\mathbf{A} = \text{diag}(\omega_D^{\sigma j})_{j=0}^{D-1}$ proposed in Remark 7.2. Let $\mathbf{x} \in \mathbb{C}^{1024}$ with sparsity $M = 9$, and \mathbf{x} has the nonzero entries

$$x_1 = 7, x_5 = 5, x_9 = -7, x_{19} = 3, x_{42} = 10, x_{45} = 5, x_{71} = -5, x_{115} = 7, x_{132} = -5.$$

Let $y_k = \hat{x}_{\sigma k} + e_k$, $k = 0, \dots, 2N-1$ be the observed data, where $N = L$ is an upper bound for the sparsity, $\sigma \in \mathbb{N}$ is prime to D , and where the error terms $e_k \in [-\tilde{\delta}, \tilde{\delta}]$ are uniformly

N	L	δ	$e(\mathbf{f})$	$e(\mathbf{c})$
7	7	∞	8.491e-11	6.614e-11
10	10	∞	6.604e-12	6.494e-12
10	10	8	2.510e-06	2.386e-06
20	10	8	4.701e-09	1.431e-08
40	20	8	2.036e-10	8.052e-10
10	10	4	2.192e-02	2.910e-02
20	10	4	4.386e-05	1.027e-04
40	20	4	2.064e-06	7.851e-06
10	10	2	9.456e-01	3.312e-01
20	10	2	5.331e-03	1.264e-02
40	20	2	2.011e-04	8.245e-04

Table 1 Average errors over 10 runs of Algorithm 3.1 for Example 8.1.

distributed. Analogously as in Algorithm 3.1, we use $\epsilon = 0.0005$ in order to estimate the sparsity M . Let us first consider the case $\sigma = 1$. The resulting positions of nonzero entries given in Table 2 are rounded to the next integer. Interestingly, we see the following behavior of the algorithm. If the number of given Fourier values is too small, then the number of nonzero entries of \mathbf{x} is underestimated. Particularly, while “single peaks” (with large distance to the next nonzero position) are found accurately, it happens that peaks with smaller separation distance are “accumulated” where also the size of the true component values plays a role. All nonzero positions of \mathbf{x} are found accurately if $N = L$ is chosen large enough, and the considered uniform noise does not effect this behavior.

The situation changes, if σ is chosen differently. Taking $\sigma = 7$ and using the 40 Fourier samples $\hat{x}_{7k} + e_k$, $k = 0, \dots, 39$, we obtain all 9 nonzero positions of \mathbf{x} accurately for $N = L = 20$. Taking $\sigma = 11$ and using the 20 Fourier values $\hat{x}_{11k} + e_k$, $k = 0, \dots, 19$, we can completely recover the nonzero positions of \mathbf{x} already for $N = L = 10$, even for noisy data with $\tilde{\delta} \leq 2$. Having found the correct positions n_j , $j = 1, \dots, M$, of nonzero entries of \mathbf{x} , the corresponding coefficients can be very accurately computed by a least squares approach.

Acknowledgements The research in this paper is partially funded by the project PL 170/16–1 of the Deutsche Forschungsgemeinschaft (DFG). This is gratefully acknowledged. The authors thank Daniel Potts for critical remarks and the numerical tests for Example 8.1.

References

- [1] R. Badeau, B. David, and G. Richard, IEEE Trans. Signal Process. **54**, 450 – 458 (2006).
- [2] D. Batenkov and Y. Yomdin, Math. Comput. **81**, 277 – 318 (2012).
- [3] D. Batenkov and Y. Yomdin, Abel Symposia 2013, to appear.
- [4] F.S.V. Bazán, SIAM J. Matrix Anal. Appl. **21**, 679 – 693 (2000).
- [5] F.S.V. Bazán and P.L. Toint, Mech. Systems Signal Process. **15**, 667 – 683 (2001).
- [6] G. Beylkin and L. Monzón, Appl. Comput. Harmon. Anal. **28**, 131 – 149 (2010).
- [7] L. Coluccio, A. Eisnberg, and G. Fedele, Signal Process. **87**, 2484 – 2490 (2007).

$N = L$	$\tilde{\delta}$	M	obtained positions of nonzero entries																	
10	0	4	2					40		107		141								
30	0	6	1					15	43		71	115	132							
50	0	8	1			9	19	42	45	71	115	132								
70	0	9	1	5	9	19	42	45	71	115	132									
10	1	4	1					40		107		141								
30	1	6	1					15	43		71	115	132							
50	1	8	1			11	20	41	45	71	115	132								
70	1	9	1	5	9	19	42	45	71	115	132									
70	2	9	1	5	9	19	42	45	71	115	132									

Table 2 Obtained positions of nonzero entries of the sparse vector $\mathbf{x} \in \mathbb{C}^{1024}$ in Example 8.2 (with $\sigma = 1$) using Algorithm 7.1.

- [8] C. de Boor, *A Practical Guide to Splines* (Springer, New York, 2001).
- [9] G.R. de Prony, *J. Éc. Polytechn.* **1**, 24 – 76 (1795).
- [10] D. Donoho, *IEEE Trans. Inform. Theory* **52**, 1289 – 1306 (2006).
- [11] P.L. Dragotti, M. Vetterli, and T. Blu, *IEEE Trans. Signal Process.* **55**, 1741 – 1757 (2007).
- [12] K.S. Eckhoff, *Math. Comput.* **64**, 671 – 690 (1995).
- [13] G.H. Golub, P. Milanfar, and J. Varah, *SIAM J. Sci. Comput.* **21**, 1222 – 1243 (1999).
- [14] N. Golyandina, V. Nekrutkin, and A. Zhigljavski, *Analysis of Time Series Structure: SSA and Related Techniques* (Chapman & Hall/CRC, Boca Raton 2001).
- [15] N. Golyandina and A. Zhigljavski, *Singular Spectrum Analysis for Time Series* (Springer, Berlin 2013).
- [16] S. Heider, S. Kunis, D. Potts, and M. Veit, *Proceedings of the 10th International Conference on Sampling Theory and Applications*, pp. 572 – 575 (2013).
- [17] G. Heinig and K. Rost, *Algebraic Methods for Toeplitz-like Matrices and Operators* (Akademie-Verlag, Berlin, 1984).
- [18] Y. Hua and T.K. Sarkar, *IEEE Trans. Acoust. Speech Signal Process.* **38**, 814 – 824 (1990).
- [19] E. Kalfoten and W.-s. Lee, *J. Symbolic Comput.* **36**, 365 – 400 (2003).
- [20] C. Lanczos, *Discourse on Fourier Series* (Hafner, New York, 1966).
- [21] D.G. Manolakis, V.K. Ingle, and S.M. Kogon, *Statistical and Adaptive Signal Processing* (McGraw-Hill, Boston, 2005).
- [22] I. Markovsky, *Automatica* **44**, 891 – 909 (2008).
- [23] V. Pereyra and G. Scherer, In: *Exponential Data Fitting and its Applications*, edited by V. Pereyra and G. Scherer (Bentham Sci. Publ., Sharjah, 2010), pp. 1 – 26.
- [24] T. Peter and G. Plonka, *Inverse Problems* **29**, 025001 (2013).
- [25] T. Peter, G. Plonka, and D. Roşca, *J. Symbolic Comput.* **50**, 159 – 169 (2013).
- [26] T. Peter, D. Potts, and M. Tasche, *SIAM J. Sci. Comput.* **33**, 314 – 334 (2011).
- [27] G. Plonka and M. Wischerhoff, *J. Appl. Math. Comput.* **42**, 117 – 137 (2013).
- [28] D. Potts and M. Tasche, *Linear Algebra Appl.* **439**, 1024 – 1039 (2013).
- [29] R. Roy and T. Kailath, *IEEE Trans. Acoust. Speech Signal Process.* **37**, 984 – 994 (1989).
- [30] M. Rudelson and R. Vershynin, *Comm. Pure Appl. Math.* **61**, 1025 – 1045 (2008).
- [31] T.K. Sarkar and O. Pereira, *IEEE Antennas Propagation Magazine* **37**, 48 – 55 (1995).
- [32] M. Vetterli, P. Marziliano, and T. Blu, *IEEE Trans. Signal Process.* **50**, 1417 – 1428 (2002).
- [33] L. Weiss and R.N. McDonough, *SIAM Rev.* **5**, 145 – 149 (1963).