

GREEDY SPARSE SPECTRAL FACTORIZATION USING REDUCED-SIZE GRAM MATRIX PARAMETERIZATION

Bogdan C. Sicleru¹ and Bogdan Dumitrescu^{1,2}

¹Dept. of Automatic Control and Systems Engineering ²Tampere Int. Center for Signal Processing
 Faculty of Automatic Control and Computers Tampere University of Technology
 Politehnica University of Bucharest P.O.Box 553, SF-33101
 313 Spl. Independenței, 060042 Bucharest, Romania Tampere, Finland
 e-mail: bogdan.sicleru@acse.pub.ro, bogdan.dumitrescu@tut.fi

ABSTRACT

In this paper we deal with retrieving the spectral factor for an autocorrelation polynomial with only a few nonzero elements. The algorithm is based on the representation of polynomials using sparse bases. We search in a greedy way for a basis by removing elements from the basis of the autocorrelation polynomial and extracting the spectral factor, using a semidefinite program. The algorithm stops when no other solution can be obtained with a smaller basis. Our algorithm appears to be faster and can be more accurate than previous methods.

Index Terms— spectral factorization, autocorrelation, semidefinite programming, greedy algorithm

1. INTRODUCTION

Spectral factorization [1] aims to recover a signal from its autocorrelation sequence. The problem is also known as phase retrieval. Its applications vary from X-ray crystallography [2], imaging in astronomy [3] to filter design [4] and speech recognition [5]. When dealing with sparse spectral factorization one has to extract the spectral factor of an autocorrelation sequence with only a few nonzero coefficients.

Spectral factorization has been approached in various ways, e.g. Schur algorithm, Riccati equations, Kalman filtering [1]. The first paper that studied the possibility of recovering the signal from its sparse autocorrelation and also giving an iterative algorithm, using singular values projections, to find the spectral factor, is [6]. Later, in [7], a convex rank minimization method was proposed which was later updated in [8]. Note that the convex optimization methods from [7] and [8] start with the lowest possible sparsity and increase it until a solution is found. A method based on combinatorial analysis was also proposed in [8], with high probability of success if $s = \mathcal{O}(n^{1/3})$, where s and n are the number of

nonzero coefficients of the autocorrelation and its degree, respectively.

Our algorithm employs a greedy method over the basis of the autocorrelation polynomial. We start with the basis of the autocorrelation and extract one element. With the new basis for the autocorrelation we solve a semidefinite programming (SDP) problem to find a rank-1 matrix from which we can extract the spectral factor. If no solution is obtained we insert the element back in the basis and extract the next one. If we obtain a solution, the support of the spectral factor becomes the new basis of the autocorrelation polynomial. The process of removing elements from the basis stops when we cannot find a solution with a smaller basis. We present two algorithms, named GreedyR and GreedyM, based on different rank minimization heuristics. While the objective function of the GreedyR algorithm uses a random matrix, as typically prescribed, the one for GreedyM is fixed. This difference is crucial—GreedyM performs better than GreedyR. We show that the GreedyM algorithm can find the solution, up to a sign change and time reversal, in more cases than previous algorithms. Also, our method appears to be faster than other algorithms from the literature.

We present the SDP problem involved in our method in Section 2. The algorithm is discussed in Section 3, while the results are presented in Section 4. We conclude in Section 5.

Notations

Multivariate variables (vectors, matrices) are written with bold letters. We denote with a vector \mathbf{h} the coefficients of the polynomial $H(z)$ and with $\text{supp}(H(z))$ its support. We denote $\text{Tr } \mathbf{M}$ the trace of the matrix \mathbf{M} and with $\det \mathbf{M}$ its determinant. $M(i, j)$ is the element on the (i, j) position of the matrix \mathbf{M} . $\text{abs}(\mathbf{x})$ is the element-wise absolute value of the vector \mathbf{x} . \mathbb{N} and \mathbb{R} are the sets of nonnegative integers and real numbers, respectively. The cardinal of a set \mathcal{I} is denoted by $|\mathcal{I}|$. By $\mathcal{I} - \mathcal{I}$ we understand the set of all differences $i - j$ where $i, j \in \mathcal{I}$.

This work was supported by the Romanian National Authority for Scientific Research, CNCS-UEFISCDI, project PN-II-ID-PCE-2011-3-0400.

2. SPARSE SPECTRAL FACTORIZATION

Denote

$$\psi_n(z) = [1 z z^2 \dots z^n]^T \quad (1)$$

the canonical base for polynomials of degree n in z . Then, any polynomial with real coefficients

$$H(z) = h_0 + h_1 z^{-1} + \dots + h_n z^{-n} \quad (2)$$

can be written as $H(z) = \mathbf{h}^T \psi_n(z^{-1})$.

Let $R(z)$ be an autocorrelation polynomial. Hence, there exists a polynomial (2) such that

$$R(z) = H(z)H(z^{-1}) = \sum_{k=-n}^n r_k z^{-k}, \quad (3)$$

where $r_k, k = -n : n$, are the coefficients of the autocorrelation polynomial. The polynomial $H(z)$ is called a spectral factor of the polynomial $R(z)$. Note that in light of (3) the powers of z in the polynomial $R(z)$ are given by all the differences between the powers of z in the polynomial $H(z)$. We denote \mathcal{I}_R the indices of the causal part of the support of $R(z)$, i.e. those with nonnegative values.

Considering (3), there exists a symmetric positive definite matrix $\mathbf{Q} \in \mathbb{R}^{(n+1) \times (n+1)}$ such that [9]

$$R(z) = \psi_n^T(z^{-1}) \cdot \mathbf{Q} \cdot \psi_n(z), \quad (4)$$

i.e. the unique coefficients of the symmetric polynomial $R(z)$ are $r_k = \mathbf{Tr}[\Theta_k \mathbf{Q}]$, $k = 0 : n$, where Θ_k is an elementary Toeplitz matrix with ones on the k -th diagonal and zeros elsewhere. The matrix \mathbf{Q} is called Gram matrix [10].

The SDP problem [11, Section 2.6.1]

$$\begin{aligned} \max_{\mathbf{Q}} \quad & \mathbf{Q}(1, 1) \\ \text{s.t.} \quad & \mathbf{Q} \succeq \mathbf{0} \\ & r_k = \mathbf{Tr}[\Theta_k \mathbf{Q}], \quad k = 0 : n \end{aligned} \quad (5)$$

can be used to extract the minimum-phase spectral factor. The solution of problem (5), \mathbf{Q}^* , has rank 1 and the minimum-phase spectral factor is

$$\mathbf{h} = \frac{1}{\sqrt{\mathbf{Q}^*(1, 1)}} \mathbf{Q}^*(:, 1). \quad (6)$$

In the case where the autocorrelation polynomial $R(z)$ is sparse, it is natural to appeal to sparse bases. For a set of indices $\mathcal{I} \subseteq \{k \in \mathbb{N} \mid 0 \leq k \leq n\}$, we define the sparse basis vector

$$\psi_{\mathcal{I}}(z) = [\dots z^k \dots]^T, \quad \text{with } k \in \mathcal{I}. \quad (7)$$

The basis from (7) is a subset of the basis from (1).

Consider a sparse autocorrelation polynomial $R(z)$ with the support $\mathcal{I} - \mathcal{I}$. Similarly to (4), it can be written

$$R(z) = \psi_{\mathcal{I}}(z^{-1}) \cdot \mathbf{Q}_{\mathcal{I}} \cdot \psi_{\mathcal{I}}(z), \quad (8)$$

where $\mathbf{Q}_{\mathcal{I}} \in \mathbb{R}^{|\mathcal{I}| \times |\mathcal{I}|}$ is a positive semidefinite matrix. Note that the supports of a sparse autocorrelation polynomial $R(z)$ and its spectral factor $H(z)$ always contain the coefficients of the powers 0 and n of z .

We aim now to find a sparse spectral factor $H(z)$ with support indices \mathcal{I} for the autocorrelation polynomial $R(z)$. We propose solving a similar problem to (5), where the autocorrelation polynomial is defined as in (8). Sparse autocorrelation polynomials don't necessarily have a sparse minimum-phase spectral factor. The criterion of (5) could be replaced by a typical rank minimization heuristic. For example, one could use in search for the sparse spectral factor the problem

$$\begin{aligned} \max_{\mathbf{Q}} \quad & \mathbf{Tr}[\mathbf{A}\mathbf{Q}] \\ \text{s.t.} \quad & \mathbf{Q} \succeq \mathbf{0} \\ & r_k = \mathbf{Tr}[\Theta_k(\mathcal{I}) \cdot \mathbf{Q}], \quad k \in \mathcal{I} - \mathcal{I} \cap \{0 : n\} \end{aligned} \quad (9)$$

where $\Theta_k(\mathcal{I}) = \mathbf{C} \cdot \Theta_k \cdot \mathbf{C}^T$ and \mathbf{C} is a selection matrix having on the k -th row a single value of 1 on the position denoted by the k -th element of \mathcal{I} and \mathbf{A} is a random matrix with normally distributed elements. The SDP problem (9) is much less complex than (5) since the size of the matrix \mathbf{Q} is $|\mathcal{I}| \times |\mathcal{I}|$ instead of $(n+1) \times (n+1)$ and thus the number of operations needed to solve it is $\mathcal{O}(|\mathcal{I}|^4) \ll \mathcal{O}(n^4)$. Moreover, the complexity of (9) depends only on the sparsity level, not on the degree of the autocorrelation polynomial.

The objective function used in the problem (9) is an heuristic used for rank minimization, see e.g. [7]. Unlike problem (5), when solving the problem (9) there is no guarantee that the obtained solution matrix has rank 1. If a rank-1 solution is obtained then the spectral factor is retrieved as in (6). If the obtained solution is not a rank-1 matrix, then we cannot retrieve the spectral factor.

Although in principle all matrices \mathbf{A} should give similar performance, we have found out that better results are given by the problem

$$\begin{aligned} \min_{\mathbf{Q}} \quad & \mathbf{Q}(1, 1) \\ \text{s.t.} \quad & \mathbf{Q} \succeq \mathbf{0} \\ & r_k = \mathbf{Tr}[\Theta_k(\mathcal{I}) \cdot \mathbf{Q}], \quad k \in \mathcal{I} - \mathcal{I} \cap \{0 : n\} \end{aligned} \quad (10)$$

As we will show, the improvement in the success rate for the problem (10) over the problem (9) can be up to 15%. Note that the criterion of (10) is the opposite of the criterion for finding the minimum-phase spectral factor. However, this is not the criterion that would give the maximum-phase spectral factor (which would mean maximizing $\mathbf{Q}(n+1, n+1)$). The minimum or maximum-phase criteria are both worse than (10) for sparse spectral factorization.

3. THE ALGORITHM

Recall the problem: given an autocorrelation polynomial $R(z)$, we seek to compute the sparsest $H(z)$ such that (3)

holds. We denote \mathcal{I} the set that in the end will contain the support of the solution $H(z)$ of the algorithm. We start with $\mathcal{I} = \mathcal{I}_R$ (the causal part of the support of $R(z)$) as the initial candidate set for the support of $H(z)$. Note that \mathcal{I}_R is also sparse, even though it may contain much more indices than the support of $H(z)$. We define the complement of an element i in the set \mathcal{I} as the difference between the maximum of \mathcal{I} and the element referred to, i.e. $n - i$. Since for a power i of z in $H(z)$, $R(z)$ contains both the power i and its complement, we remove all the elements from \mathcal{I} whose complements do not exist in the set \mathcal{I} . Then, we solve the problem (10). If we obtain a rank-1 solution, we retrieve the spectral factor $H(z)$ via (6) and set all coefficients smaller in absolute value than a threshold ε_h to zero. We define this factor as the current solution and set \mathcal{I} as its support.

Next, we try to eliminate indices from \mathcal{I} one by one and we start with an index i . If $\mathcal{I}_R \subseteq \mathcal{I} - \mathcal{I}$, i.e. the current support can generate the support of the autocorrelation, we solve problem (10). If a rank-1 solution is obtained, the spectral factor is extracted from it; all coefficients smaller than ε_h in absolute value are set to zero. We set \mathcal{I} the support of the new $H(z)$ and we continue eliminating indices from \mathcal{I} . If we do not have a rank-1 solution or \mathcal{I} can not generate \mathcal{I}_R we insert the index i back to \mathcal{I} and we eliminate the next index. The algorithm stops when no rank-1 solution can be obtained with a smaller support. Note that we solve problem (10), before eliminating any index, due to the fact that the support of the autocorrelation can also be the exact support of the spectral factor.

We name GreedyM this algorithm, which is listed below. An autocorrelation polynomial $R(z)$ stands as input variable for the algorithm. The algorithm returns a polynomial $H(z)$ which is a sparse spectral factor, possibly the sparsest. We name algorithm GreedyM for which we replace the problem (10) with problem (9) as Algorithm GreedyR.

The case $s = 2$ (the support contains only the endpoints) is treated separately, in analytic fashion. Considering the equality constraints from (10) the Gram matrix \mathbf{Q} has the form

$$\mathbf{Q}^* = \begin{bmatrix} q & r_n \\ r_n & r_0 - q \end{bmatrix} \quad (11)$$

where $q \in \left[\frac{r_0 - \sqrt{r_0^2 - 4r_n^2}}{2}, \frac{r_0 + \sqrt{r_0^2 - 4r_n^2}}{2} \right]$. The matrix \mathbf{Q}^* has rank 1 if and only if

$$\det \mathbf{Q}^* = 0, \quad (12)$$

which is equivalent to

$$q^2 - qr_0 + r_n^2 = 0. \quad (13)$$

Using the minimum solution of the equation (13) to determine the matrix \mathbf{Q}^* , we retrieve the spectral factor with (6).

Algorithm GreedyM Search for sparse spectral factor

Input: the autocorrelation $R(z)$

Output: the spectral factor $H(z)$

$\mathcal{I} \leftarrow \mathcal{I}_R$, the indices of the causal part of $R(z)$

remove the indices that do not have the complement in \mathcal{I}

if the problem (10) has a rank-1 solution **then**

extract the spectral factor using (6)

set coefficients of $\text{abs}(\mathbf{h}) < \varepsilon_h$ to 0

the candidate sparse spectral factor is $H(z)$

$\mathcal{I} \leftarrow \text{supp}(H(z))$

while we find rank-1 solutions **do**

$n' \leftarrow |\mathcal{I}|$

for $i = 2 \rightarrow n' - 1$ **do**

remove β , the i -th element from \mathcal{I}

if $\mathcal{I}_R \subseteq \mathcal{I} - \mathcal{I}$ **then**

if the problem (10) has a rank-1 solution **then**

extract the spectral factor using (6)

set coefficients of $\text{abs}(\mathbf{h}) < \varepsilon_h$ to 0

the candidate sparse spectral factor is $H(z)$

$\mathcal{I} \leftarrow \text{supp}(H(z))$

break

add β back to \mathcal{I}

4. RESULTS

We start our tests on the autocorrelation polynomial depicted by Figure 1a, which is similar to the one used in [6, Example 2]. Using algorithm GreedyM we have found that the spectral factor is a six coefficient polynomial, as shown in Figure 1b, which has the same support as the spectral factor found in [6].

The GreedyM algorithm starts with the support of the autocorrelation

$$\mathcal{I} = [0 \ 2 \ 3 \ 9 \ 11 \ 19 \ 22 \ 30 \ 33 \ 39 \ 41 \ 42 \ 44 \ 52 \ 61 \ 63]$$

from which we remove the elements that do not have the complement in \mathcal{I} , namely 3, 9, 39, 42. Solving the algorithm with the current support we do not obtain a rank-1 solution. Next, we eliminate element 2 from \mathcal{I} and solve the problem (10). We obtain a rank-1 solution and the support of the current spectral factor has 6 elements. Note that the support size cannot decrease because a smaller support for the spectral factor cannot generate the support of the autocorrelation. Hence, the last solution is the final one.

The total running time of the GreedyM algorithm is 5 seconds using a Pentium 4 with a 2 GHz processor, while the running time of the algorithm from [6] on a 2.2 GHz computer is 16 seconds. Hence, our algorithm appears to be faster. An extensive comparison with [6] is not possible due to the lack of more examples in [6].

Next, we test our algorithms on sparse polynomials $H(z)$ with coefficients chosen to be normally distributed numbers.

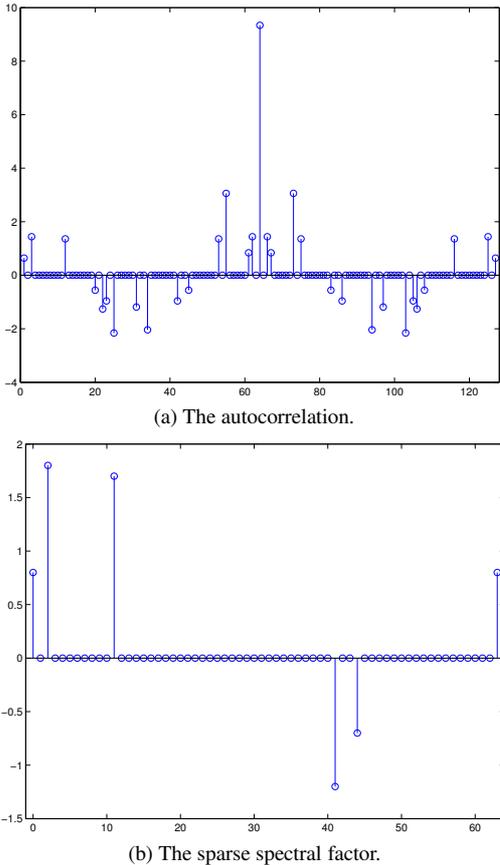


Fig. 1: The autocorrelation and its sparse spectral factor.

For each polynomial $H(z)$ we compute its autocorrelation $R(z)$ on which we test our algorithms. To validate a spectral factor we allow a maximum difference of ε_h from the original one, in absolute value. For obtained spectral factors different than the original one, but with the same number of coefficients, we have compared the difference between their autocorrelations in absolute value up to ε_o .

We test the two algorithms—GreedyM and GreedyR—on sets of $N = 1000$ autocorrelation polynomials of length 16, i.e. of degree $n = 15$, as in [7]. The tests are made with random endpoints, hence the polynomials can have smaller order than n . Table 1 shows the results we have obtained for different levels of sparsity. In Figure 2 we make a comparison between different algorithms; the results for [7] are extracted from the paper. The results show that the choice of the objective function is of great importance; the variant used in the problem (10) performs better than the one used in problem (9), i.e. GreedyM offers better results than GreedyR.

We move now to filters of larger support. We test our algorithm GreedyM on polynomials of size 64 and 8192 and compare the results with the ones extracted from [8]. Figures 3a and 3b show the rate of success for our algorithm in comparison with the results reported in [8]. We ran $N = 1000$ and $N = 100$ tests for $n = 63$ and $n = 8191$, respectively.

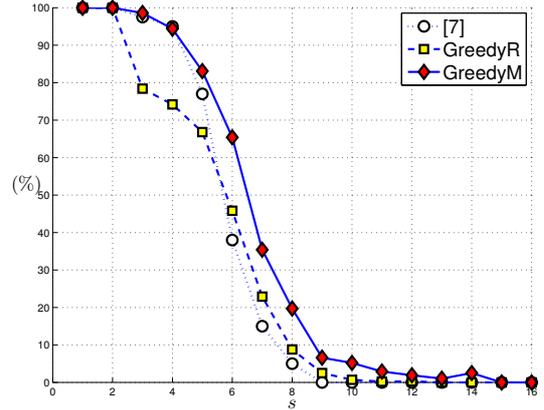


Fig. 2: Percentages of success using different algorithms for $n = 15$, $N = 1000$.

The figures show that our algorithm obtains the spectral factor in more cases than the algorithms used in [8]. Note that the results reported in [8] for $n = 63$ are made using a convex optimization algorithm, an updated version of the one from [7], while the results reported for $n = 8191$ are made using an algorithm based on combinatorial analysis, with high probability to succeed if $s = \mathcal{O}(n^{1/3})$. Moreover, the convex optimization algorithms from [7] and [8] cannot run on an average PC for the $n = 8191$ case, due to high memory usage. Our method runs faster than the method from [7] too, e.g. for $n = 63$, $s = 6$, for a single autocorrelation polynomial, our algorithm provides the spectral factor in 1.6 seconds, while the algorithm from [7] needs 25 seconds to find the spectral factor, with given s , using a 2.7 GHz computer.

We have used $\varepsilon_h = 10^{-5}$ and $\varepsilon_o = 10^{-4}$.

5. CONCLUSIONS

We have proposed a spectral factorization algorithm for autocorrelations of sparse polynomials based on SDP. We characterize sparse polynomials using sparse bases. The algorithm aims to find the sparse spectral factor for the autocorrelation polynomial. The method seeks the support of the spectral factor by removing an element from the current basis and attempting to extract the spectral factor using an SDP problem. If the problem has no solution, we insert the element back into the basis and remove the next one. If a solution is obtained, the support of the spectral factor gives the new basis. The greedy algorithm stops reducing the size of the basis when a smaller size basis can no longer generate the support of the autocorrelation. Comparing our algorithm with previous methods we have shown that our algorithm can be more reliable and its running time is smaller.

Table 1: Percentages of success for $n = 15$, $N = 1000$.

| s | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|-------------|------|------|------|------|------|------|-----|-----|-----|-----|-----|-----|
| GreedyM (%) | 97.2 | 89.4 | 83.8 | 61.8 | 37.5 | 17.5 | 7.3 | 2.4 | 2 | 2.5 | 1.2 | 1.8 |
| GreedyR (%) | 78.4 | 74.2 | 66.8 | 45.8 | 22.9 | 8.8 | 2.5 | 0.7 | 0.2 | 0.3 | 0 | 0 |

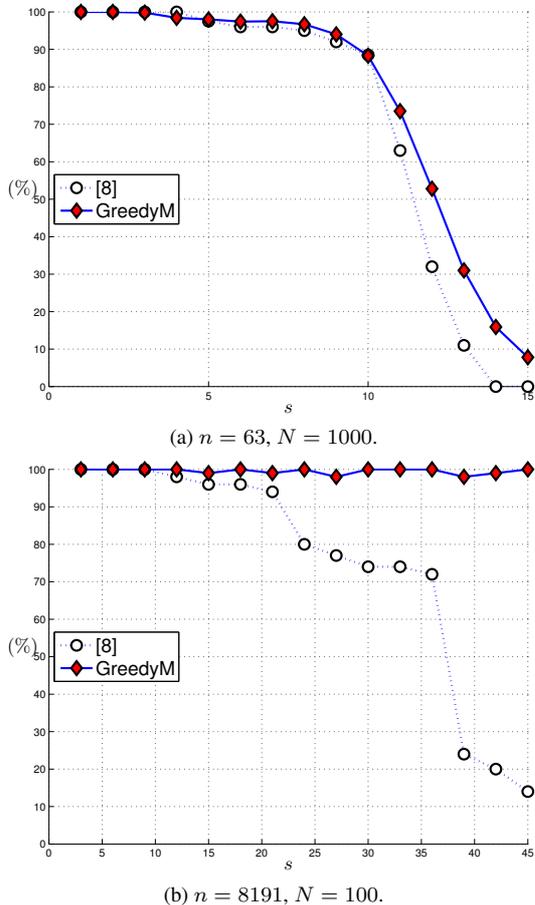


Fig. 3: Percentages of success using different algorithms.

6. ACKNOWLEDGEMENT

We thank Kishore Jaganathan for providing the source codes for the algorithm from [7].

7. REFERENCES

- [1] A.H. Sayed and T. Kailath, “A survey of spectral factorization methods,” *Num. Lin. Alg. Appl.*, vol. 8, pp. 467–496, 2001.
- [2] R.P. Millane, “Phase retrieval in crystallography and optics,” *J. Opt. Soc. Am. A*, vol. 7, pp. 394–411, 1990.
- [3] J.C. Dainty and J.R. Fienup, *Phase Retrieval and Image Reconstruction for Astronomy*, chapter 7, pp. 231–

275, *Image Recovery: Theory and Application*. Academic Press, 1987.

- [4] S.P. Wu, S. Boyd, and L. Vandenberghe, *Applied and Computational Control*, chapter FIR filter design via spectral factorization and convex optimization, pp. 215–245, *Signals and Circuits*. Birkhäuser, 1998.
- [5] L. Rabiner and B.H. Juang, *Fundamentals of speech recognition*, *Signal Processing*. Prentice Hall, 1993.
- [6] Y.M. Lu and M. Vetterli, “Sparse spectral factorization: unicity and reconstruction algorithms,” in *Proc. Int. Conf. Ac. Speech and Sign. Proc. (ICASSP)*, Prague, Czech Republic, May 2011, pp. 5976–5979.
- [7] K. Jaganathan, S. Oymak, and B. Hassibi, “Phase retrieval for sparse signals using rank minimization,” in *Proc. Int. Conf. Ac. Speech and Sign. Proc. (ICASSP)*, Kyoto, Japan, March 2012, pp. 3449–3452.
- [8] K. Jaganathan, S. Oymak, and B. Hassibi, “Recovery of sparse 1-D signals from the magnitudes of their fourier transform,” in *IEEE Int. Symp. on Information Theory*, 2012.
- [9] B. Dumitrescu, I. Tăbuș, and P. Stoica, “On the parameterization of positive real sequences and MA parameter estimation,” *IEEE Trans. Signal Proc.*, vol. 49, no. 11, pp. 2630–2639, Nov. 2001.
- [10] J.W. McLean and H.J. Woerdeman, “Spectral factorizations and sum of squares representations via semidefinite programming,” *SIAM J. Matr. Anal. Appl.*, vol. 23, no. 3, pp. 646–655, 2002.
- [11] B. Dumitrescu, *Positive Trigonometric Polynomials and Signal Processing Applications*, Springer, 2007.