

# Unrolled Compressed Blind-Deconvolution

Bahareh Tolooshams\*, *Student Member, IEEE*, Satish Mulleti\*, *Member, IEEE*, Demba Ba, and Yonina C. Eldar, *Fellow, IEEE*

**Abstract**—The problem of sparse multichannel blind deconvolution (S-MBD) arises frequently in many engineering applications such as radar/sonar/ultrasound imaging. To reduce its computational and implementation cost, we propose a compression method that enables blind recovery from much fewer measurements with respect to the full received signal in time. The proposed compression measures the signal through a filter followed by a subsampling, allowing for a significant reduction in implementation cost. We derive theoretical guarantees for the identifiability and recovery of a sparse filter from compressed measurements. Our results allow for the design of a wide class of compression filters. We, then, propose a data-driven unrolled learning framework to learn the compression filter and solve the S-MBD problem. The encoder is a recurrent inference network that maps compressed measurements into an estimate of sparse filters. We demonstrate that our unrolled learning method is more robust to choices of source shapes and has better recovery performance compared to optimization-based methods. Finally, in data-limited applications (fewshot learning), we highlight the superior generalization capability of unrolled learning compared to conventional deep learning.

## I. INTRODUCTION

In a variety of remote sensing applications (e.g., radar imaging [1], seismic signal processing [2], sonar imaging [3], and ultrasound imaging [4, 5]), an unknown source signal reflected from sparsely located targets is measured through a multichannel receiver system. The signal received at each channel can be modelled as a linear convolution of a source signal and sparse filters; the filters, modelling the targets' locations, are also known as impulse responses of the channels [6, 7]. Sparse multichannel blind-deconvolution (S-MBD) is the problem of recovering the source signal shape and sparse filters from received measurements [8, 9].

Bilen et al. [10] proposed an  $\ell_1$ -based convex optimization method to solve the S-MBD problem. Gribonval et al. [11] and Mulleti et al. [12] studied S-MBD in the context of sparse dictionary calibration, and Garcia-Cardona and Wohlberg [13]

formulated it via convolutional dictionary learning. Theoretical analysis on identifiability of S-MBD exists in the literature [8, 14, 15]. These works derived the identifiability results in terms of the number of channels by assuming that the sparse filters are random and all the measurements in each channel are available. Specifically, they showed that given  $M$  measurements in each channel,  $N = \mathcal{O}(M \log^4 M)$  channels are sufficient for identifiability [8, 14]. In addition to using a large number of channels, these results rely on having access to full measurements at each channel. None of these methods uses the sparsity of the filters to reduce the number of measurements in each channel. In a multichannel receiver system, hardware cost and power consumption increase with the number of channels, as each channel requires a dedicated analog-to-digital converter and corresponding circuitry. Further, the computational cost depends on the number of channels and the number of measurements per channel, as a larger number of measurements requires more storage and computations. To reduce the hardware implementation and computational cost, it is desired to solve S-MBD from compressed (few) measurements captured from fewer channels.

In the context of dictionary learning, Tolooshams et al. [16, 17] proposed structured inference neural networks to solve the S-MBD problem using algorithm unrolling [18, 19]; however, their method still relies on full measurements. Chang et al. [20] further extend the unrolling framework to enable dictionary learning from compressed measurements through an unstructured random matrix; this compression has two drawbacks. First, random-matrix-based compression may not achieve large compression rates in practice. Second, their practical implementation may be challenging due to their unstructured nature.

Mulleti et al. [12] addressed the later drawback and proposed a structured, linear compression operator that could be realized using a finite impulse response (FIR) filter. The authors took a frequency-domain approach for S-MBD and derived the identifiability results by using a fixed sum-of-sincs (SoS) FIR filter for compression. In addition, the authors applied the blind dictionary calibration (BDC) method [11] and the truncated-power iteration (TPI) approach [21] to recover sparse filters from the compressed Fourier measurements. This approach's limitation is that the source's frequency spectrum is assumed not to vanish on selected points. Since the source is unknown in advance, designing the filter is challenging. Furthermore, the structure of the filter is fixed and independent of the reconstruction method.

Recent work addressed this challenge by offering a data-dependent joint design of the frequency locations of a continuous-time SoS filter and the reconstruction parameters [22]. However, the filters still follow the SoS structure with

B. Tolooshams and D. Ba are with the School of Engineering and Applied Sciences, Harvard University, Cambridge, MA; S. Mulleti is with the Department of Electrical Engineering, Indian Institute of Technology (IIT) Bombay, Mumbai, India; Y. C. Eldar is with the Faculty of Math and Computer Science, Weizmann Institute of Science, Israel. Email: btolooshams@seas.harvard.edu, mulleti.satish@gmail.com, demba@seas.harvard.edu, yonina.eldar@weizmann.ac.il

\*Equal contribution. B. Tolooshams and D. Ba acknowledge the partial support of NSF under grant number DMS-2134157 and the ARO under grant number W911NF-16-1-0368. The latter is part of a collaboration between US DOD, UK MOD and UK Engineering and Physical Research Council (EPSRC) under the Multidisciplinary University Research Initiative. For S. Mulleti and Y. C. Eldar, this research was partially supported by the Israeli Council for Higher Education (CHE) via the Weizmann Data Science Research Center, by a research grant from the Estate of Tully and Michele Plesser, by the European Union's Horizon 2020 research and innovation program under grant No. 646804-ERC-COG-BNYQ, by the Israel Science Foundation under grant no. 0100101, and by the QuantERA grant C'MON-QSENS.

Manuscript accepted May, 2023. Preliminary results of this work appeared in the proceedings of IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP) 2021.

learnable frequency locations, resulting in limited flexibility. Hence, a natural question is whether a general condition on the filters, which does not require an SoS form, can be derived such that compression is possible with perfect recovery. Moreover, can we leverage data to learn such a filter together with the reconstruction method?

### A. Contributions

In this paper, we focus on answering the preceding two questions. Starting from a time-domain approach, we consider the problem of filter-based compression for S-MBD. We derive time-domain identifiability conditions on the compression filter and the S-MBD measurements to enable recovery. The time-domain approach helps in deriving general conditions on the filters compared to the frequency-domain framework. Moreover, we offer learning-based methods for designing the filters which were briefly proposed and discussed in [23]. Our main contributions are summarized as follows.

- We formulate a compressive sparse-MBD problem based on a hardware-efficient linear compression operator that can be realized using an FIR filter. Unlike prior theoretical analysis on Fourier samples [12], we provide time-domain theoretical guarantees for the recovery of sparse filters from reduced measurements. We derive identifiability conditions jointly on the compression filter and the source in terms of spark properties on the convolution matrices of the source and compression filter. Unlike non-vanishing frequency requirements of the source's Fourier spectrum in prior work [12], our framework neither enforces any specific structural requirement on the compression filter nor places vanishing spectrum conditions on the source. However, the above-mentioned flexibility on identifiability conditions comes with a trade-off, a need for more measurements than the minimum required Fourier samples achieved by [12] for identifiability.
- We propose unrolled inference networks (Fig. 2b); this is to give an unrolling structure to standard deep learning inference networks (Fig. 2a) using our compressive S-MBD problem formulation. We build upon our preliminary results in [23] and enable learning of the compression filter from data. We note that the architecture shown in Section IIIa is previously proposed by [23]. We show that unrolled learning outperforms optimization-based methods and highlight its robustness to the source shapes of Gaussian and symmetric wavelet (Fig. 5). We characterize the effect of unrolling, as compression ratio (CR) varies, e.g., only a few unrolled layers are needed to achieve highly accurate identification of the target locations (Fig. 4). Furthermore, we propose an additional linear compression, performing filtering followed by decimation; the compression is more memory efficient (Table I) and has a minimal decline in performance (Fig. 6). Our results differ from [23] as follows: [23] a) provided no theoretical analysis of their proposed unrolled network, b) considered only the noiseless setting where the source shape is Gaussian, hence, no robust characterization was performed, and c) focused on target

amplitude recovery rather than hit rate. Moreover, the new compression based on decimated filtering and the following contribution are unique to this paper.

- We propose combining a network denoiser with an unrolled network and show that this framework significantly improve performance in the task of recovery from compressed noisy measurements (Fig. 8). Finally, we highlight the advantage of unrolled learning over conventional deep learning in fewshot learning; our framework is significantly superior to generic deep networks in the low data regime (Fig. 10).

### B. Notations and Organizations

We use the following notation in the paper. Vectors and matrices are denoted by lower and uppercase boldfaced letters, respectively. A real-valued,  $N$ -length, causal sequence  $\{s[0], s[1], \dots, s[N-1]\}$  is denoted by a vector  $\mathbf{s} = [s[0], s[1], \dots, s[N-1]]^T \in \mathbb{R}^N$  where the superscript  $T$  denotes transpose. For any two sequences  $\mathbf{s} \in \mathbb{R}^{M_s}$  and  $\mathbf{x} \in \mathbb{R}^{M_x}$ , their convolution is  $\mathbf{s} * \mathbf{x} \in \mathbb{R}^{M_s+M_x-1}$ . For any vector  $\mathbf{s} \in \mathbb{R}^{M_s}$ , a convolution matrix with  $M$  columns is constructed as,

$$\mathbf{C}_s(M) = \begin{pmatrix} s[0] & & & 0 \\ & s[1] & & \vdots \\ & \vdots & \ddots & \\ s[M_s-1] & & & s[0] \\ & \vdots & & s[1] \\ & 0 & & \vdots \\ & & & s[M_s-1] \end{pmatrix}. \quad (1)$$

With this notation, the convolution  $\mathbf{s} * \mathbf{x}$  is represented as  $\mathbf{C}_s(M_x)\mathbf{x} = \mathbf{C}_x(M_s)\mathbf{s}$ . The symbol  $\ast_t$  denotes truncated convolution, with the longer vector appearing on the left side of the symbol. For  $M_s \geq M_x$ ,  $\mathbf{s} \ast_t \mathbf{x}$  denotes values  $\mathbf{s} * \mathbf{x}[m]$  for  $M_x \leq m \leq M_s$ . The spark of a matrix  $\mathbf{A}$ , that is,  $\text{Spark}(\mathbf{A})$  is defined as the minimum number of columns of  $\mathbf{A}$  that are linearly dependent.

The paper is organized as follows. In Section II, we formulate the S-MBD problem mathematically and present our assumptions. Time-domain identifiability results are derived in Section III. In Section IV, we present a data-driven unrolled-based compression filter design; we outline various unrolled architectures and discuss their training mechanism. Numerical results in Section V followed by a conclusion.

## II. PROBLEM FORMULATION

Consider a set of  $N$  signals given as,

$$\mathbf{y}^n = \mathbf{s}_0 * \mathbf{x}^n, \quad n = 1, \dots, N, \quad (2)$$

where  $*$  denotes linear convolution operation. In this model,  $\mathbf{s}_0$  represents a common source signal, and  $\mathbf{x}^n$  denotes  $n$ -th channel filter.

The problem of multichannel blind-deconvolution (MBD) is to determine the source and filters from the measurements  $\{\mathbf{y}^n\}_{n=1}^N$ . The problem is ill-posed without further assumptions on either the source, or the filters, or both. Sparse MBD

(S-MBD) is one such widely applicable model where the filters are assumed to be sparse. In addition, the source is assumed to have a finite impulse response (FIR). Specifically, we make the following structural assumptions:

(A1) **Sparse Filters:**  $\mathbf{x}^n \in \mathbb{R}^{M_x}$  and  $\|\mathbf{x}^n\|_0 \leq L$  for  $n = 1, \dots, N$ .

(A2) **FIR Source:**  $\mathbf{s}_0 \in \mathbb{R}^{M_{s_0}}$ .

(A3) **Sparse Coprime Filters:** Consider the decomposition  $\mathbf{x}^n = \mathbf{h}_0 * \bar{\mathbf{x}}^n$  for all  $n$ . If length of  $\mathbf{h}_0$  is greater than one, then  $\|\bar{\mathbf{x}}^n\|_0 > L$  for all  $n$ .

(A3') **Non-zero first and last elements of the source:**  $s_0[1] \neq 0$  and  $s_0[M_{s_0}] \neq 0$ .

The sparse filter model with FIR source is ubiquitous in applications [12]. For example, in radar imaging with  $N$  receivers,  $\mathbf{s}_0$  denotes the transmit pulse. By assuming that the target scene consists of  $L$  stationary point targets, the signal received by the  $n$ -th receiver can be modelled as  $\mathbf{y}^n$  in (2). Here, non-zero locations and amplitudes of the sparse filter  $\mathbf{x}^n$  contain the information of the distances and amplitudes of the targets for the  $n$ -th receiver. As one can generate only compactly supported transmits signals, the FIR source model is justified. As discussed in [12], either (A3) or (A3') ensure unique identifiability<sup>1</sup>. Here, identifiability implies that the source and sparse filters can be uniquely determined up to a scaling constant. As we are representing one-dimensional sequences as finite-length vectors with known support, we do not have shift ambiguity which alters the support of the sequences.

In the compressive S-MBD problem, the objective is to determine either the sparse filters or the source or both from compressed measurements of  $\{\mathbf{y}^n\}_{n=1}^N$ . In this paper, we focus on recovering the sparse filters  $\{\mathbf{x}^n\}_{n=1}^N$  from compressed measurements. This problem is equivalent to estimating the locations of the targets in radar [1] or ultrasound imaging [4, 5]. The desirable properties of the compression operator are that it should be linear, practically implementable, and importantly allows perfect recovery with the highest compression rate possible. To this end, we use linear-filtering-based compression. Specifically, to compress the measurements, we consider subsampling the filtered signal

$$\mathbf{z}^n = \mathbf{h} * \mathbf{y}^n = \mathbf{h} * \mathbf{s}_0 * \mathbf{x}^n, \quad (3)$$

where  $\mathbf{h} \in \mathbb{R}^{M_h}$  is a compression filter applied across all channels. Note that the compression mechanism is similar to that in [12]. However, a major difference is that in our framework, the filter  $\mathbf{h}$  does not require a discrete-SoS (D-SoS) model. The compressed measurements can be written as

$$\mathbf{z}^n = \mathbf{s} * \mathbf{x}^n, \quad (4)$$

where  $\mathbf{s} = \mathbf{h} * \mathbf{s}_0 \in \mathbb{R}^{M_s}$  is the *effective source* of length  $M_s = M_{s_0} + M_h - 1$ , and  $M_y = M_{s_0} + M_x - 1$ . If the number of subsampled measurements of  $\mathbf{z}^n$  is less than  $M_y$  then compression is achieved.

<sup>1</sup> Assumption (A3') is simple to verify in practice compared to assumption (A3). For identifiability, we need either (A3) or (A3') to be satisfied.

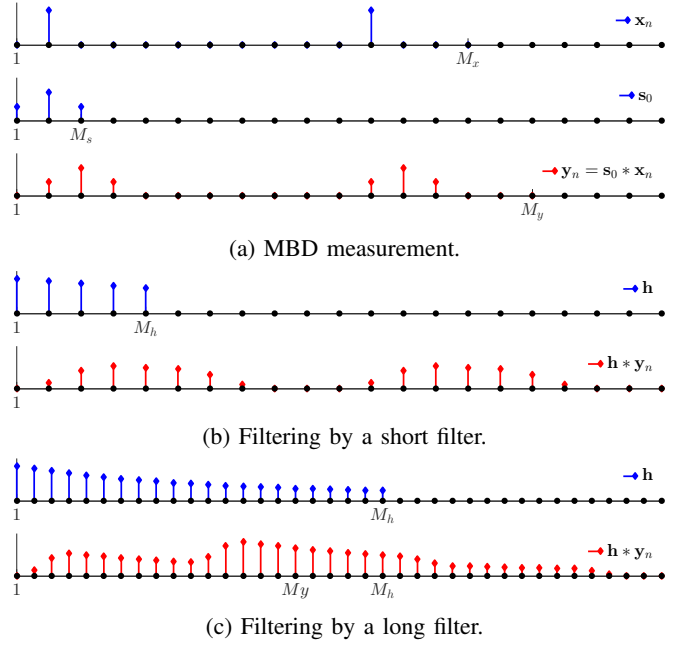


Fig. 1: A toy example to demonstrate that recovery from compression through subsampling may not be possible for  $M_h \leq M_y$ .

In the following section, we discuss conditions on the filter  $\mathbf{h}$ , or more specifically on the effective source  $\mathbf{s}$  to ensure identifiability up to a scaling ambiguity. Specifically, if there exist an alternative set of effective source  $\hat{\mathbf{s}}$  and sparse filters  $\{\hat{\mathbf{x}}^n\}_{n=1}^N$ , such that

$$\mathbf{z}^n = \hat{\mathbf{s}} * \hat{\mathbf{x}}^n, \quad n = 1, \dots, N, \quad (5)$$

on the subsampled locations then, there exists a scalar  $\beta \neq 0$  such that

$$\hat{\mathbf{s}} = \beta \mathbf{s} \quad \text{and} \quad \hat{\mathbf{x}}^n = \mathbf{x}^n / \beta, \quad n = 1, \dots, N. \quad (6)$$

If a set of alternative solutions satisfies (5) but not (6) then the problem is not uniquely identifiable.

### III. TIME-DOMAIN THEORETICAL GUARANTEES FOR COMPRESSED S-MBD

In this section, we discuss the problem of uniquely identifying the sparse filters  $\{\mathbf{x}^n\}_{n=1}^N$  from subsamples of  $\{\mathbf{z}^n\}_{n=1}^N$ . The identifiability depends on the subsampling strategy, which in turn depends on the relative lengths of the compression filter  $\mathbf{h}$ , source  $\mathbf{s}_0$ , and sparse filters.

The MBD measurements in (4) can be written as

$$\mathbf{z}^n = \mathbf{C}_h \mathbf{C}_{s_0} \mathbf{x}^n \in \mathbb{R}^{M_z}, \quad n = 1, \dots, N, \quad (7)$$

where  $\mathbf{C}_{s_0} \in \mathbb{R}^{M_y \times M_x}$  and  $\mathbf{C}_h \in \mathbb{R}^{M_z \times M_y}$  are convolution matrices corresponding to the source and filter, respectively. To identify  $\{\mathbf{x}^n\}_{n=1}^N$  from subsamples of  $\{\mathbf{z}^n\}_{n=1}^N$ , we consider a binary-compression or subsampling operator  $\mathbf{B} \in \{0, 1\}^{K \times M_z}$  where  $K < M_z$  is the number of compressed measurements. Each row of  $\mathbf{B}$  has exactly one non-zero element, and the

matrix has a full row rank. The compressed measurements are given as

$$\bar{\mathbf{z}}^n = \mathbf{B}\mathbf{z}^n = \mathbf{B}\mathbf{C}_h \mathbf{C}_{s_0} \mathbf{x}^n, \quad n = 1, \dots, N. \quad (8)$$

The identifiability depends on the number of subsamples  $K$ , subsampling pattern or matrix  $\mathbf{B}$ , and properties of  $\mathbf{C}_h \mathbf{C}_{s_0}$ .

To proceed further, we present a few basic properties of  $\mathbf{C}_s = \mathbf{C}_h \mathbf{C}_{s_0}$  that will allow subsampling. We consider two possible scenarios: (i)  $M_h \leq M_y$  and (ii)  $M_h > M_y$ . We first show that recovery from compression through subsampling may not be possible for  $M_h \leq M_y$ . Consider a toy example with  $M_x = 15$ ,  $L = 2$ , and  $M_{s_0} = 3$  (See Fig. 1a). In Fig. 1b, we chose a compression filter of length  $M_h = 4$  and plot the filtered signal  $\mathbf{h} * \mathbf{y}^n$ . We observe that a few values of  $\mathbf{z}^n = \mathbf{h} * \mathbf{y}^n$  are zero due to the short compression filter. Subsampling those values will result in a loss of information. In addition, each non-zero sample of  $\mathbf{z}^n$  only has a contribution from a few non-zero entries of  $\mathbf{x}^n$ . This implies that the information of  $\mathbf{x}^n$  is not well spread across the samples of  $\mathbf{z}^n$  which is desirable for compression.

Next, consider a long compression filter such that  $M_h > M_y$ . In the toy example, we set  $M_h = M_y + 5$ . The compression filter and  $\mathbf{z}^n = \mathbf{h} * \mathbf{y}^n$  are shown in Fig. 1c. We note that none of the values are zero unless the convolution sum is zero in that sample. Further, it can be verified that the values of  $\mathbf{z}^n[m]$  for  $M_y - 1 \leq m \leq M_h - 1$  have a contribution from all the samples of  $\mathbf{y}^n$  [23]. This implies that these samples, indexed by the set  $\mathcal{K} = \{M_y - 1, M_y, \dots, M_h - 1\}$ , carry information of all the non-zero components of the sparse filter. The set  $\mathcal{K}$  includes the sample of the convolution  $\mathbf{C}_h$  where  $\mathbf{h}$  and  $\mathbf{y}$  are fully overlapped. Hence, perfect recovery through subsampling may be possible by considering the measurements  $\{\mathbf{z}^n[m]\}_{m \in \mathcal{K}}$ . Given this intuition, we set the operator  $\mathbf{B}$  to subsample the measurements from the set  $\mathcal{K}$  where the compression ratio is controlled by the relative length of the compression filter  $M_h$  and signal  $M_y$ . For the remainder of this section, we denote  $\mathbf{T}_h(r) := \mathbf{B}\mathbf{C}_h$ , applied on a signal of length  $r$ . For a fixed  $M_y$ , the number of compressed measurements  $K = |\mathcal{K}| = M_h - M_y + 1$  increases linearly with the length of the compression filter. The measurements  $\{\mathbf{z}^n[m]\}_{m \in \mathcal{K}}$  are said to be compressed if  $K < M_y$ . We next derive conditions that ensure compression together with identifiability. Given that the subsamples  $\{\mathbf{z}^n[m]\}_{m \in \mathcal{K}}$  carry information about the sparse filters, the questions required to be answered are whether the sparse filters can be identifiable from these measurements with the assumption that  $K < M_y$ . To answer the question, we next study the problem of identifying the sparse filters from the subsamples of the truncated convolution

$$\bar{\mathbf{z}}^n = \mathbf{T}_h(M_y) \mathbf{y}^n = \mathbf{h} * \mathbf{y}^n, \quad (9)$$

where the Toeplitz matrix  $\mathbf{T}_h(r) \in \mathbb{C}^{(M_h - r + 1) \times r}$ , for  $1 \leq$

$r \leq M_h$ , is constructed as

$$\mathbf{T}_h(r) = \begin{pmatrix} \mathbf{h}[r-1] & \mathbf{h}[r-2] & \cdots & \mathbf{h}[0] \\ \mathbf{h}[r] & \mathbf{h}[r-1] & \cdots & \mathbf{h}[1] \\ \mathbf{h}[r+1] & \mathbf{h}[r] & \cdots & \mathbf{h}[2] \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{h}[M_h-1] & \mathbf{h}[M_h-2] & \cdots & \mathbf{h}[M_h-r] \end{pmatrix}. \quad (10)$$

For  $r = M_y$ , the matrix consists of  $K = M_h - M_y + 1$  rows or  $K$  compressed measurements. The compressed measurements can alternatively be written as

$$\bar{\mathbf{z}}^n = \mathbf{T}_h(M_y) \mathbf{C}_{s_0}(M_x) \mathbf{x}^n = \mathbf{A}_{(M_y, M_x)} \mathbf{x}^n, \quad (11)$$

where

$$\mathbf{A}_{(M_y, M_x)} = \mathbf{T}_h(M_y) \mathbf{C}_{s_0}(M_x) \in \mathbb{C}^{K \times M_x}, \quad (12)$$

is equivalent to the compression matrix in standard sparse recovery. The subscripts of  $\mathbf{A}$  denote the number of columns of matrices  $\mathbf{T}_h$  and  $\mathbf{C}_{s_0}$ . In the current setup,  $\mathbf{A}$  is unknown, unlike the standard sparse recovery problem where the matrix is known. For identifiability, the spark property of the matrix  $\mathbf{A}$  plays a crucial role, as summarized in the following theorem.

**Theorem 1** (Sparse-Filters Identifiability for Compressive MBD). *Consider the MBD measurements  $\mathbf{y}^n = \mathbf{s}_0 * \mathbf{x}^n$ ,  $n = 1, \dots, N$  where  $N \geq 2$ . The source  $\mathbf{s}_0$  satisfies assumption (A2) and the sparse filters  $\{\mathbf{x}^n\}_{n=1}^N$  satisfy assumption (A1) with sparsity level  $L < \min(\sqrt{M_{s_0}/2}, \sqrt{M_x})$ . Consider the measurements  $\mathbf{z}^n[m]$ ,  $m \in \mathcal{K}$  or  $\bar{\mathbf{z}}^n = \mathbf{A}_{(M_y, M_x)} \mathbf{x}^n$ ,  $n = 1, \dots, N$ . The measurements are compressed, and we have the following necessary and sufficient conditions for identifiability.*

- 1) (Necessary) If  $\text{Spark}(\mathbf{A}_{(M_y, M_x)}) \leq 2L$  then the sparse filters are not uniquely identifiable.
- 2) (Sufficient) If  $\text{Spark}(\mathbf{A}_{(M_y + M_x - 1, 2M_x - 1)}) > 2L^2$  and either (A3) or (A3') is satisfied, then the sparse filters can be identified up to a scaling constant.

*Proof.* We first focus on the necessary and sufficient conditions and then show that for both cases the measurements are compressed.

*Necessary Conditions:* From (11), we know that if  $\text{Spark}(\mathbf{A}_{(M_y, M_x)}) \leq 2L$  then for every sparse filter  $\mathbf{x}^n$  there exists another  $L$ -sparse vector  $\hat{\mathbf{x}}^n \neq \mathbf{x}^n$  such that,

$$\mathbf{A}_{(M_y, M_x)}(\mathbf{x}^n - \hat{\mathbf{x}}^n) = \mathbf{0}, \quad n = 1, \dots, N. \quad (13)$$

Hence, there exists an alternative set of sparse filters  $\{\hat{\mathbf{x}}^n\}_{n=1}^N$  satisfying the compressed measurements.

*Sufficient Conditions:* Let us assume that there exists an alternative source  $\hat{\mathbf{s}}_0$  and a set of feasible sparse filters  $\{\hat{\mathbf{x}}^n\}_{n=1}^N$  such that

$$\bar{\mathbf{z}}^n = \mathbf{h} *_{\mathbf{t}} (\mathbf{s}_0 * \mathbf{x}^n) = \mathbf{h} *_{\mathbf{t}} (\hat{\mathbf{s}}_0 * \hat{\mathbf{x}}^n). \quad (14)$$

This implies that the alternative set  $(\hat{\mathbf{s}}_0, \{\hat{\mathbf{x}}^n\}_{n=1}^N)$  results in the same compressed measurements. Since all the vectors are of finite dimension, from (14) it can be shown that

$$\mathbf{h} * \mathbf{s}_0 * \mathbf{x}^n = \mathbf{h} * \hat{\mathbf{s}}_0 * \hat{\mathbf{x}}^n, \quad n = 1, \dots, N. \quad (15)$$



Then, for any two channels, say from channel numbers  $n = 1, 2$ , we have the following cross-convolution relationship [12]:

$$\mathbf{h}_t^* (s_0 * \underbrace{(\hat{\mathbf{x}}_1 * \mathbf{x}_2 - \hat{\mathbf{x}}_2 * \mathbf{x}_1)}_{\mathbf{q}}) = \mathbf{0}, \quad (16)$$

where  $\mathbf{q} \in \mathbb{R}^{2M_x-1}$ . From the condition  $L < \sqrt{M_x}$ , we have that  $\|\mathbf{q}\|_0 \leq 2L^2$  [12]. Using (12), we rewrite (16) as

$$\mathbf{A}_{(M_y+M_x-1, 2M_x-1)} \mathbf{q} = \mathbf{0}. \quad (17)$$

Since  $\|\mathbf{q}\|_0 \leq 2L^2$ , from (17) we infer that  $\mathbf{q} = \mathbf{0}$  if  $\text{Spark}(\mathbf{A}_{(M_y+M_x-1, 2M_x-1)}) \geq 2L^2$ . Then, for  $\mathbf{q} = \mathbf{0}$ , we have that  $\hat{\mathbf{x}}_1 * \mathbf{x}_2 = \hat{\mathbf{x}}_2 * \mathbf{x}_1$  which can be written in the  $z$ -domain as

$$\hat{X}_1(z)X_2(z) = \hat{X}_2(z)X_1(z). \quad (18)$$

This relationship implies that there exists a function  $G(z)$  such that  $X^n(z) = G(z)\hat{X}^n(z)$ . Hence, it can be shown that  $G(z)$  is a constant function if one of the conditions (A3) or (A3') is satisfied (See [12, Section IV.E]). In other words,  $\hat{\mathbf{x}}^n$  is a scaled version of  $\mathbf{x}^n$  and hence a feasible solution. A similar argument can be used for any pair of sparse filters, and we conclude that identifiability holds for all sparse filters.

*Compressed Measurements:* Next, we show that the number of measurements  $K < M_y$ , i.e., the measurements  $\mathbf{z}^n$  are compressed. The identifiability results are derived in terms of spark properties of  $\mathbf{A}$ . For the spark properties to be satisfied, the length of the compression filter  $M_h$  requires satisfying certain conditions, as discussed next. In the discussion, we use the fact that for any matrix to satisfy the minimum spark property, its number of rows should be greater than or equal to the minimum spark. For example, consider a matrix  $\mathbf{B} \in \mathbb{C}^{K \times N}$ . Then, for any integer  $L \leq N$ , the condition  $\text{Spark}(\mathbf{B}) > L$  is satisfied only if  $K \geq L$ .

In the sufficiency part, the inequality  $\text{Spark}(\mathbf{A}_{(M_y, M_x)}) \leq 2L$  indicates a necessary condition. Hence, for identifiability it is necessary that  $\text{Spark}(\mathbf{A}_{(M_y, M_x)}) > 2L$  which is ensured only if the number of rows of  $\mathbf{A}_{(M_y, M_x)}$  or more specifically, the number of rows of  $\mathbf{T}_h(M_y)$  is greater than or equal to  $2L$ . Since the number of rows of  $\mathbf{A}_{(M_y, M_x)}$  determines the number of compressed measurements (see (11)), the results imply that identifiability is not possible if the number of measurements  $K$ , which is equal to the number of rows of  $\mathbf{A}_{(M_y, M_x)}$ , is less than  $2L$ . For  $K \geq 2L$ , the compression filter's length  $M_h$  should be chosen such that  $M_h \geq 2L + M_y - 1$ . Since  $L < \min(\sqrt{M_{s_0}/2}, \sqrt{M_x})$ , we have that

$$M_{s_0} > 2L^2 \quad \text{and} \quad M_x > L^2. \quad (19)$$

By using the above bounds, it can be verified that  $2L \leq K < M_y = M_x + M_{s_0} - 1 < 3L^2 - 1$  and hence the measurements are compressed.

For the sufficient conditions, it is required that  $\text{Spark}(\mathbf{A}_{(M_y+M_x-1, 2M_x-1)}) > 2L^2$  which in turn means that the number of rows of  $\mathbf{T}_h(M_y + M_x - 1)$  is greater than or equal to  $2L^2$ . This can be insured by choosing the length of the filter such that  $M_h \geq 2L^2 + 2M_x + M_{s_0} - 3$ . From the spark condition in the sufficient part, it may not be straightforward to determine the number of measurements, as the condition is derived for  $\mathbf{A}_{(M_y+M_x-1, 2M_x-1)}$  which is

different from  $\mathbf{A}_{(M_y, M_x)}$  used for compression (see (11)). However, by using the condition  $M_h \geq 2L^2 + 2M_x + M_{s_0} - 3$  and calculating the number of rows of  $\mathbf{A}_{(M_y, M_x)}$ , we note that a minimum of  $K = M_h - M_y + 1 = 2L^2 + M_x - 1$  measurements is available. For  $K = 2L^2 + M_x - 1$  and  $M_y = M_x + M_{s_0} - 1$ , it can be verified that  $K < M_y$  as  $2L^2 < M_{s_0}$ . Hence, the measurements are compressed.  $\square$

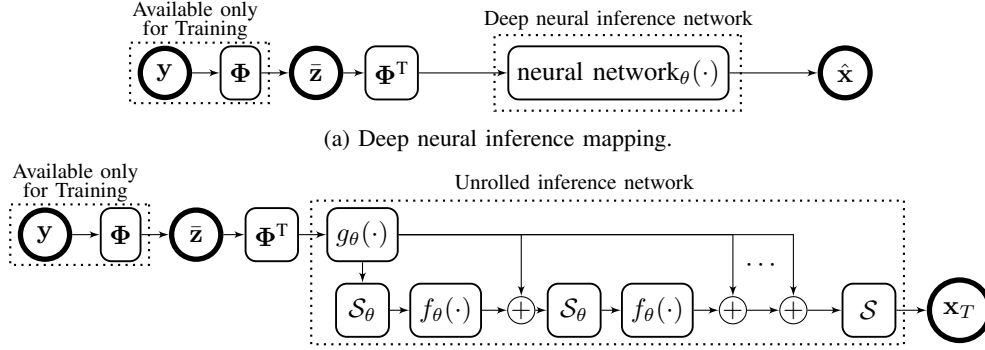
The theorem states necessary and sufficient conditions on the source and compression filter in terms of spark properties. In addition, we show that the measurements are compressible if the sparsity level is restricted in terms of the lengths of the source and the sparse filter. In the frequency-domain guarantees derived in [12], the sparsity level should satisfy the inequality  $L < \sqrt{M_x}$  for identifiability. However, in the time-domain guarantees (Theorem 1), it should satisfy the condition  $L < \min(\sqrt{M_s}, \sqrt{M_x})$  which is a stronger condition if  $M_s < M_x$ . For  $M_s < M_x$ , the time-domain results can identify less sparse signals. This is a consequence of the joint identifiability condition on both the compression filter and the MBD measurements.

Compared to the identifiability condition in [12], the results in Theorem 1 are generic. To be specific, in the frequency-domain identifiability results [12], a D-SoS filter is used together with a non-vanishing spectral support condition on the source. Whereas the time-domain results in Theorem 1 are generic and do not use any structure or condition on the compression filter or source. Hence, the conditions derived can be used to design a wider class of signals for compression. Note that the D-SoS filter, together with the non-vanishing spectral support condition, also satisfies the spark conditions in Theorem 1 (See Appendix for details). For sufficient guarantees, the approach in [12] requires minimum  $2L^2$  Fourier measurements, whereas the proposed time-domain approach requires a minimum of  $2L^2 + M_x - 1$  compressed measurements. The additional measurements in the proposed approach are due to the joint identifiability condition. Theoretical guarantees were also presented in [8, 14, 15, 24] for the S-MBD problem where they consider random sparsity models used. Specifically, these models assume that the sparse filters follow a Bernoulli-Gaussian distribution. The models may not be applicable in many applications. In comparison, the proposed approach is deterministic (no randomness) in nature and hence more widely applicable.

Given the above-mentioned identifiability results for the recovery of sparse filters from compressed measurements, we next focus on formulating the optimization problem and providing algorithms to solve S-MBD.

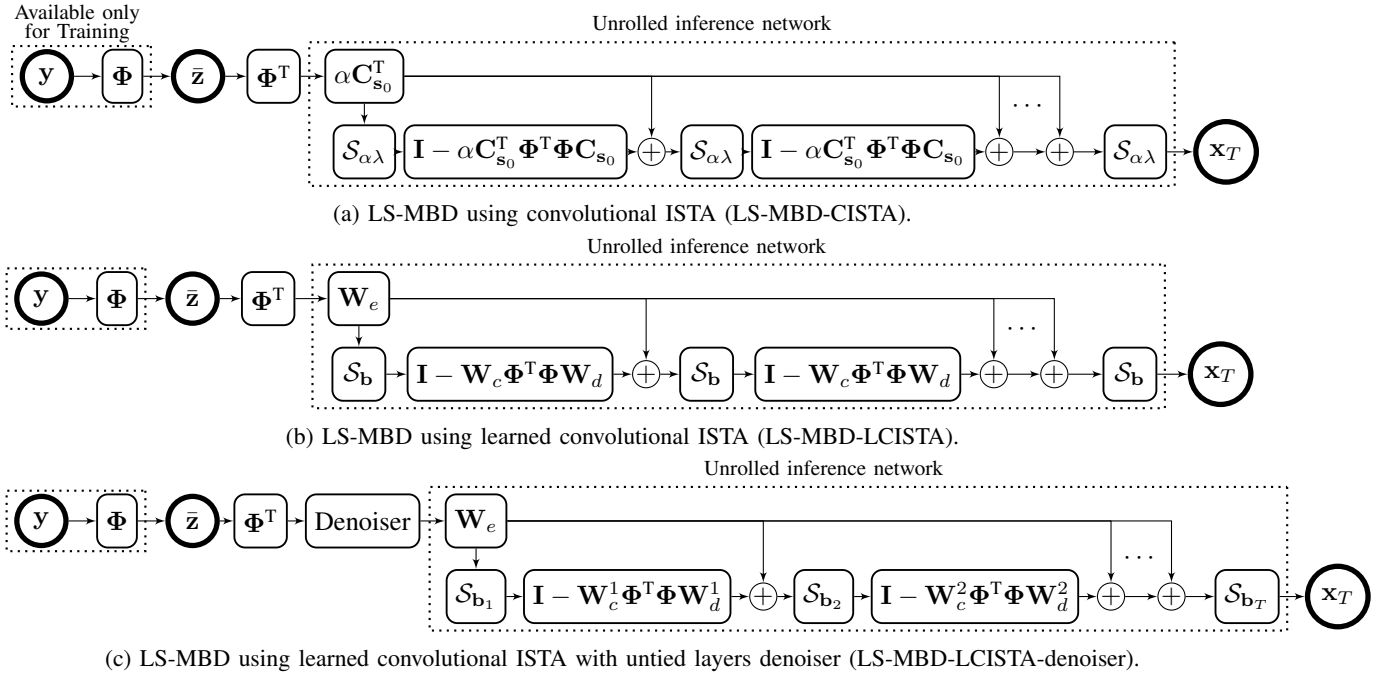
#### IV. UNROLLED LEARNING FOR COMPRESSED MULTICHANNEL BLIND DECONVOLUTION

Given the generality of the compression filter, we propose to learn the compression filter from data in the time domain. Specifically, we introduce a structured inference network based on algorithm unrolling [18] to learn a filter-based compression operator for compressed S-MBD. First, we formulate the problem as an alternating-minimization-based optimization and review inference networks in the context of compressive



(b) Model-based unrolled inference mapping. Here,  $\theta$  denotes the network parameters, and  $f_\theta$  and  $g_\theta$  are neural blocks with  $S_\theta$  representing the nonlinearity.

Fig. 2: Inference mapping for compressed multichannel blind-deconvolution.



(c) LS-MBD using learned convolutional ISTA with untied layers denoiser (LS-MBD-LCISTA-denoiser).

MBD. Then, we introduce constrained and unrolled networks to structure the inference mapping, on which our main results are based.

#### A. Learned optimization-based methods

Given the generative model, we form the following optimization problem to recover the source  $s_0$ , estimate the sparse filters  $\mathbf{x}^n$ , and learn a compression operator  $\Phi$  (i.e.,  $\mathbf{T}_h(M_y)$ ):

$$\min_{\Phi, s_0, \{\mathbf{x}^n\}_{n=1}^N} \sum_{n=1}^N \frac{1}{2} \|\bar{\mathbf{z}}^n - \Phi(s_0 * \mathbf{x}^n)\|_2^2 + \lambda \|\mathbf{x}^n\|_1 \quad (20)$$

$$\text{s.t. } \|s_0\|_2 = \|\mathbf{h}\|_2 = 1,$$

where  $\lambda$  is a regularization parameter enforcing sparsity, and  $\mathbf{h}$  is the filter corresponding to the compression operator  $\Phi$ .  $\Phi$  is implemented as a convolution operator followed by a truncation. To avoid scaling ambiguity, we impose a norm constraint on the source  $s_0$  and compression filter  $\mathbf{h}$ . We can solve this problem by alternating minimization. At update

iteration  $l$ , given an estimate of  $\mathbf{s}_0^{(l)}$  and  $\Phi^{(l)}$ , solve the sparse coding problem for each example, i.e.,

$$\mathbf{x}^{n(l)} = \arg \min_{\mathbf{x}^n} \frac{1}{2} \|\bar{\mathbf{z}}^n - \Phi^{(l)}(\mathbf{s}_0^{(l)} * \mathbf{x}^n)\|_2^2 + \lambda \|\mathbf{x}^n\|_1. \quad (21)$$

We call this step sparse coding. Then, use the newly estimated sparse filters  $\mathbf{x}^{n(l)}$  to refine the learned  $s_0$  and  $\mathbf{h}$ , i.e.,

$$\{\Phi^{(l+1)}, \mathbf{s}^{(l+1)}\} = \arg \min_{\Phi, s_0} \sum_{n=1}^N \frac{1}{2} \|\bar{\mathbf{z}}^n - \Phi(s_0 * \mathbf{x}^{n(l)})\|_2^2$$

$$\text{s.t. } \|s_0\|_2 = \|\mathbf{h}\|_2 = 1 \quad (22)$$

which is the learning step. Next, we review inference networks and discuss how to use the above alternating-minimization-based formulation to construct structured deep neural networks for S-MBD.

## B. Inference networks

Following a fully data-driven approach, one may utilize standard deep learning-based inference networks; this is to learn a mapping from compressed measurements  $\bar{\mathbf{z}}$  to the sparse filters  $\mathbf{x}$ . For training, the full measurements  $\mathbf{y}$  are compressed using the trainable compression operator  $\Phi$ , and then  $\Phi^T$  is applied to have an approximation of  $\mathbf{y}$ ; this is followed by an inference network to output  $\hat{\mathbf{x}}$ . The joint learning of  $\Phi$  and  $\mathbf{x}$  is achieved by forward and backward passes. The forward pass infers the sparse filter  $\mathbf{x}$ , and  $\Phi$  is learned in the backward pass during training. This approach is used previously by [25, 26] to recover data from compressed measurements, where the inference mapping is a deep neural network (Fig. 2a). In this paper, we propose to give an unrolling structure [18] to the inference network (Fig. 2b); the building blocks  $g_\theta$ ,  $f_\theta$ , and  $\mathcal{S}_\theta$  are determined by the optimization model, chosen to solve the problem. Prior work has demonstrated the competitive performance of unrolled networks, with an order of magnitude lower number of trainable parameters, to a state-of-the-art deep network on the supervised task of Poisson image denoising [27]. To complement this, we show in Section V-G that indeed our unrolled learning framework (Fig. 2b) outperforms the generic neural network-based inference mappings (Fig. 2a) in data-limited regimes.

1) *Constrained unrolled networks*: To construct the inference mapping, we use an approach similar to [17] to map the alternating optimization problem into an autoencoder. The forward pass is designed for sparse filter recovery separable among the examples (i.e., channels), and the backward pass is used to learn parameters shared among the dataset (i.e., source and compression filter). We design the forward pass of the encoder to solve the sparse coding step (21) by performing  $T$  iterations of iterative soft thresholding algorithm (ISTA) [28]. Each iteration implements

$$\mathbf{x}_{t+1} = \mathcal{S}_{\alpha\lambda}(\mathbf{x}_t - \alpha \mathbf{C}_{s_0}^T \Phi^T (\Phi \mathbf{C}_{s_0} \mathbf{x}_t - \bar{\mathbf{z}})), \quad (23)$$

where  $\alpha$  is the step-size,  $\mathcal{S}_b(v) = \text{ReLU}_b(v) - \text{ReLU}_b(-v)$  is the shrinkage operator with  $\text{ReLU}_b(v) = (v - b) \cdot \mathbf{1}_{v \geq b}$ , and  $\mathbf{C}_{s_0}$  is the convolution matrix corresponding to the source  $s_0$ . The decoder maps the code estimates  $\mathbf{x}_T^n$  into an estimate of the data  $\hat{\mathbf{y}}^n$  and then the compressed data  $\hat{\bar{\mathbf{z}}}^n$  using  $s_0$  and  $\mathbf{h}$ . The learning step (22) minimizes the loss

$$\begin{aligned} \min_{\Phi, \mathbf{C}_{s_0}} \sum_{n=1}^N \frac{1}{2} \|\bar{\mathbf{z}}^n - \Phi \mathbf{C}_{s_0} \mathbf{x}_T^n(\mathbf{C}_{s_0}, \Phi)\|_2^2, \\ \text{s.t. } \|\mathbf{s}_0\|_2 = \|\mathbf{h}\|_2 = 1 \end{aligned} \quad (24)$$

achieved by backpropagated gradient descent, followed by a projection step to ensure the norm constraints. Having access to the full measurements  $\mathbf{y}^n$ , the learning step (22) can be modified to  $\min_{\Phi, \mathbf{C}_{s_0}} \sum_{n=1}^N \frac{1}{2} \|\mathbf{y} - \mathbf{C}_{s_0} \mathbf{x}_T^n(\mathbf{C}_{s_0}, \Phi)\|_2^2$  or  $\min_{\Phi, \mathbf{C}_{s_0}} \sum_{n=1}^N \frac{1}{2} \|\mathbf{x}^n - \mathbf{x}_T^n(\mathbf{C}_{s_0}, \Phi)\|_2^2$ , in the supervised setting, with the knowledge of the sparse codes. Indeed, this is possible as the sparse filters are being estimated through the inference network and are the function of  $\mathbf{C}_{s_0}$  and  $\Phi$ . Hence, the gradient of the loss  $\|\mathbf{x}^n - \mathbf{x}_T^n\|_2^2$  with respect to  $\mathbf{C}_{s_0}$  and

$\Phi$  can be computed by backpropagation. We use this sparse filter loss to train the proposed architectures.

Fig. 3a shows the encoder network architecture. In the context of dense dictionary learning and given full measurements, Tolooshams and Ba [29] provide a theoretical analysis of such constrained networks; they provide conditions on the network and data under which dictionary recovery is achieved via network training.

2) *Learned unrolled networks*: The ISTA encoder architecture can be further modified by relaxing and untying the filter operator corresponding to  $\mathbf{C}_{s_0}$  and  $\mathbf{C}_{s_0}^T$  as shown in Fig. 3b. We call this architecture learned structured multichannel blind deconvolution learned ISTA (LS-MBD-LISTA). This is to increase the capacity of the model to learn from data. Unrolling ISTA iterations and similar relaxations were introduced first by [30] for dense matrices, where each iteration of ISTA is referred to as a layer of a structured neural network. Our proposed architecture is a variant of LISTA introduced in [30] where the relaxed operations  $\mathbf{W}_e$ ,  $\mathbf{W}_c$ , and  $\mathbf{W}_d$ , each has their own filters  $\mathbf{h}_e$ ,  $\mathbf{h}_c$ ,  $\mathbf{h}_d$ ;  $\mathbf{W}_e$  and  $\mathbf{W}_c$  performs correlation, and  $\mathbf{W}_d$  performs convolution. In this case, we can also let the biases of the neural network  $\mathbf{b}$  be learned from data.

In the presence of measurement noise, we propose to add a differentiable denoiser (e.g., neural network architecture) before the unrolled layers and specify different  $\mathbf{W}_c^t$ , and  $\mathbf{W}_d^t$  kernels and vector bias  $\mathbf{b}_t$  for each layer  $t$  (Fig. 3c). We use the variants of our proposed learned structured multichannel blind deconvolution network in the results discussed in the paper.

## C. Training

Similar to the two-stage approach proposed in [23], we first train the encoder along with a decoder given full measurements (i.e.,  $\Phi = \mathbf{I}$ ). This is an unsupervised training for source  $\mathbf{C}_{s_0}$  recovery by minimizing the reconstruction loss  $\|\mathbf{y} - \hat{\mathbf{y}}\|_2^2$  using the LS-MBD-CISTA network encoder architecture. This network architecture is only used for source recovery purposes. For applications where the source is known, this first stage can be skipped. In the second stage, having access to the recovered  $\mathbf{C}_{s_0}$  and following the generative model (2), we construct a training set consisting of  $\{\mathbf{y}^n, \mathbf{x}^n\}_{n=1}^N$  (See Section V-A for more info on data generation). LS-MBD-LCISTA or LS-MBD-LCISTA-denoiser are the main networks used for target recovery in this stage. For a given compression ratio  $\text{CR} = \frac{M_h - M_y + 1}{M_y}$ , we learn the compression operator  $\Phi$  and the encoder network parameters in a supervised manner by minimizing  $\|\mathbf{x} - \hat{\mathbf{x}}\|_2^2$ . We trained one network for each of the studied scenarios. For example, each CR value has its trained network. Moreover, when studying the effect of source shape, unrolling layers, or measurement noise, one network is trained for each specific scenario.

## V. RESULTS

This section evaluates the performance of our unrolled learning filter-based compression for multichannel blind deconvolution. We outline the data generation details and explain the training procedure. The section characterizes the

proposed method and the effect of unrolling on sparse filter recovery. Moreover, we evaluate unrolled learning against the optimization-based method in [12]. Specifically, given the source and sparse targets, we show the outperformance of LS-MBD-LCISTA against FS-MBD in the absence of measurement noise. Then, we offer a more efficient compression operator with minimal reduction in performance. At last, we assess our method, specifically LS-MBD-LCISTA-denoiser, in the presence of noise and highlight that using unrolled learning to structure the inference mapping makes it possible to train networks in data-limited regimes.

#### A. Simulated data generation and evaluation criteria

We generated training simulated data with the following parameters ( $N = 10,000$ ,  $M_{s_0} = 99$ ,  $M_x = 100$ ,  $x_s = 6$  where the code is  $x_s = \|\mathbf{x}\|_0$ -sparse.) For most of the experiments, we used a Gaussian source shape,  $s_0[k] = e^{-\frac{6}{5\sqrt{M_{s_0}}}}(k - \lfloor \frac{M_{s_0}}{2} \rfloor)^2$  where  $k \in \{0, 2, \dots, M_{s_0} - 1\}$ , followed by a normalization step  $\|\mathbf{s}_0\|_2 = 1$ . Gaussian source shapes are mainly concentrated in low frequencies. Hence, to highlight the robustness of our method toward source shapes containing narrow frequency bands, for a portion of the experiments, we switched to a symmetric Morlet wavelet with a scaling factor of 1.5. The choice of source shape is the same during both training and inference. The sparse filter supports are chosen uniformly at random, with amplitudes drawn from  $\text{Unif}(4, 9)$ . For the noisy experiments, we added Gaussian noise, denoted by  $\mathbf{v}$ , with various  $\sigma$  and computed the SNR of the signal as  $20 \log \frac{\|\mathbf{s}_0 * \mathbf{x}\|_2}{\|\mathbf{v}\|_2}$ . Specifically, given the same data distribution across all channels, the reported SNR is averaged over the channel (data examples).

We followed a similar approach and generated two additional test and validation sets, each of size  $N = 1,000$ . The validation set is used to choose the best-performing network across the training epochs. All datasets of training, test, and validation follow the same data distribution, i.e., they share similar sparse filter statistics and source shapes.

We refer to the sparse filters as code or the target of interest, and evaluate the performance of the methods using three metrics: normalized root-mean-squared error (NRMSE) between the estimate and ground-truth code, exact hit rate  $\frac{\text{number of correctly detected targets}}{\text{actual number of targets}}$ , and approximate hit rate which we allow a tolerance of one (e.g., for ground-truth code with a support set of  $\{7, 45, 90\}$  and estimated code with a support set of  $\{7, 46, 89\}$ , the approximate hit rate is 100%). In this definition, the actual target refers to the support (non-zero entries) of the ground-truth code, and the detected targets refers to the support of the estimated code. We expect a lower NRMSE, and a higher hit rate as the compression ratio CR increases. We report results after picking  $x_s = 6$  highest elements of the code support and setting the rest of the code entries to 0.

#### B. Training

The networks are implemented and trained in PyTorch. The convolution weights are randomly initialized using a standard

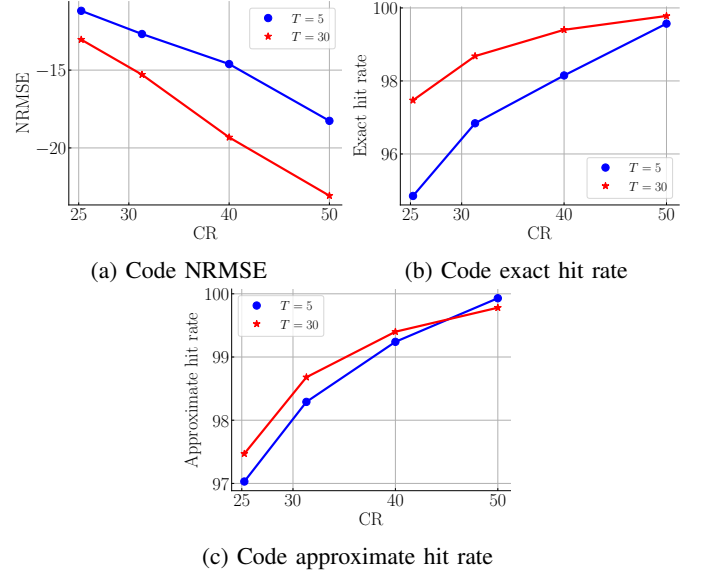


Fig. 4: Effect of Unrolling.

Normal distribution, and the biases, if trained, are set to zero. For LS-MBD-CISTA, we set  $\alpha = 0.05$ , and for the rest of the networks,  $\mathbf{W}_e$  and  $\mathbf{W}_c$  are scaled by a factor of 0.05 after initialization to adjust to the step size. For all networks except LS-MBD-CISTA, the step size  $\alpha$  is absorbed, hence learned within the weight  $\mathbf{W}_c$ . We unrolled all networks for up to  $T = 30$  layers unless stated. Given the non-negativity of the sparse filters, we set  $\mathcal{S} = \text{ReLU}$ .

All networks are trained for 10,000 epochs with a batch size of 100. We use the ADAM optimizer with a learning rate of  $10^{-3}$  and  $\epsilon = 10^{-14}$  [31]; the learning rate is decayed by a factor of 0.9 in the middle of the training. The small value of  $\epsilon$  is to ensure the exploration of the loss landscape without staying in local minima. The chosen learning rate is the common value used for unrolled networks in the literature [29].

#### C. Effect of unrolling

We characterize the proposed methods as a function of unrolling, i.e., how well the target locations and amplitudes are recovered as the network is unrolled. We used the Gaussian source shape for this section and the architecture type of LS-MBD-LCISTA. We compared unrolling networks of  $T = 5$  and  $T = 30$  for a range of compression ratios. Fig. 4 demonstrates the results: the amount of unrolling is crucial for amplitude recovery regardless of compression ratio and exact hit rate, especially when CR is low. The reported results are the average of two independent trials.

#### D. Unrolled learning vs. optimization-based methods

We compare the proposed network LS-MBD-LCISTA (Fig. 3b) against the optimization-based method of fixed and structured MBD (FS-MBD) [12]; FS-MBD uses source information in the Fourier domain to design a compression matrix  $\Phi$ , uses a blind-dictionary calibration for structured source estimation, and solve the sparse coding step using



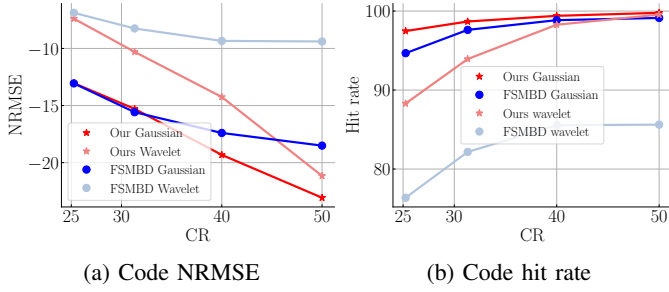


Fig. 5: Ours vs. FS-MBD. Our method is more robust to the source shape. FS-MBD only works well when the source is concentrated in low-freq and is not narrow in freq.

FISTA [32, Ch. 2][33]. Concerning the problem formulation in this manuscript, FS-MBD solves (20) over sparse targets and the source in the frequency domain with known designed compression. For a fair comparison, we assume both of the methods know the source shape; hence, the problem reduces to designing/learning the compression.

We use two different source shapes (Gaussian and symmetric wavelet). For both scenarios and in all compression ratios, LS-MBD-LCISTA outperforms FS-MBD. This result conveys two main messages. First, the learnable network parameters, especially the biases, allow LS-MBD-LCISTA to estimate the sparse filters in terms of NRMSE (Fig. 5a) much better than FS-MBD which used  $\ell_1$ -based sparse coding with fixed sparsity level. Second, our method is robust to source shape; the out-performance of LS-MBD-LCISTA against FS-MBD is significant when the source has a wavelet shape (i.e., the frequency content is not concentrated around low frequencies and contains narrow bands). Fig. 5 visualizes this result in terms of both NRMSE and exact hit rate.

#### E. Toward more efficient compression operator

Although the proposed compression operator is efficient compared to unstructured matrix compressions as discussed in [23], the method has one limitation; The length of the filter  $M_h$  must be greater than the length of the signal, and this makes the method not suitable for long data. In this section, we ameliorate this by re-designing the compression operator as follows: the compression performs a sum of  $P$  correlations (shown in (25)) where their kernel size  $M_h$  is much smaller than the size of the signal. Then, the compression is achieved by decimation (this decimation is also referred to as strides). The desired compression ratio can be achieved by changing the number of convolutional filters and the decimation factor.

$$\bar{\mathbf{z}} = \Phi \mathbf{y} = \sum_{p=1}^P \mathbf{D}_d(\mathbf{h}_p * \mathbf{y}) \quad (25)$$

where  $\mathbf{D}_d$  denotes the decimation operator matrix by a factor of  $d$ . Hence, the compression  $\Phi$  is implemented as a sum of convolutions with strides. We call this method “filter banks”, and refer to the former one as “long”. Fig. 6 shows that both methods show similar code recovery performance for various ranges of compression ratios. The data used for this experiment has a symmetric wavelet source shape, and the filter bank

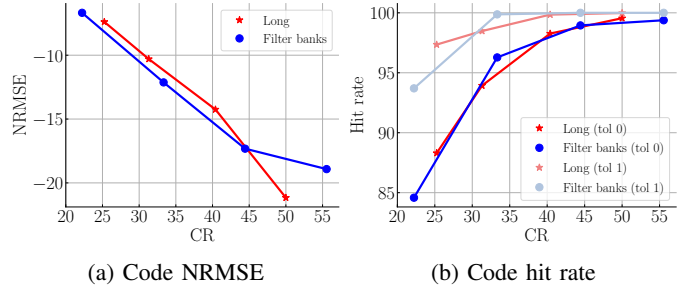


Fig. 6: Long and filter bank compression filters show similar performance.

TABLE I: Memory storage of the compression operator when it is structure long or filter banks. Memory cost is quantified by the dimensionality of the compression operator.

Long		Filter banks	
CR	Memory Cost	CR	Memory Cost
50	296	55.55	$5 \times 9 = 45$
40.4	277	44.44	$4 \times 9 = 36$
31.31	259	33.33	$3 \times 9 = 27$
25.25	247	22.22	$2 \times 9 = 18$

compression uses kernels of size  $M_h = 9$  with stride 9. For example, for  $\text{CR} = \%55.5$ , we used 5 convolutional filters, or for  $\text{CR} = \%22.2$ , only two filters are used. The architecture used in the above comparison is LS-MBD-LCISTA.

Given the comparable performance, Table I highlights the memory advantage of the filter banks method compared to the long method. For example, given the generated data (see Section V-A) for the long method when  $\text{CR} = 50$ , the compression length is  $M_h = 296$  which is almost twice as large as the signal length. However, for the filter banks methods, compression of  $\text{CR} = 55.5$  can be achieved by 5 filters of each length  $M_h = 9$  with decimation (stride) of 9 (This is indeed independent of the length of the signal). We compute the memory cost based on the dimensionality of the compression operator. This comes down to much smaller memory storage of  $5 \times 9 = 45$ . For this scenario, Figure 7 visualizes a test example of full measurements (a), the ground-truth and estimated sparse filters code (b), the source shape (c), the compressed data (d), and the learned compression filter banks (e).

#### F. Toward data-driven unrolled denoising networks

This section characterizes the performance of the proposed method in the presence of noise. The generated data contains the Gaussian source shape and the target amplitudes are following Uniform distribution  $\text{Unif}(4, 5)$ . We trained the networks using 10,000 training examples. We considered the case where the data is corrupted with additive Gaussian noise with  $\sigma = 0.161$  (i.e., SNR of approximately 15 dB). We used LS-MBD-LCISTA-freelayers-denoiser with  $T = 10$  unrolled layers and DnCNN [34] as denoiser. The denoiser has a kernel size of 3, 64 feature maps, and a depth of 4. We use validation and test sets of size  $N = 100$  where each example has its

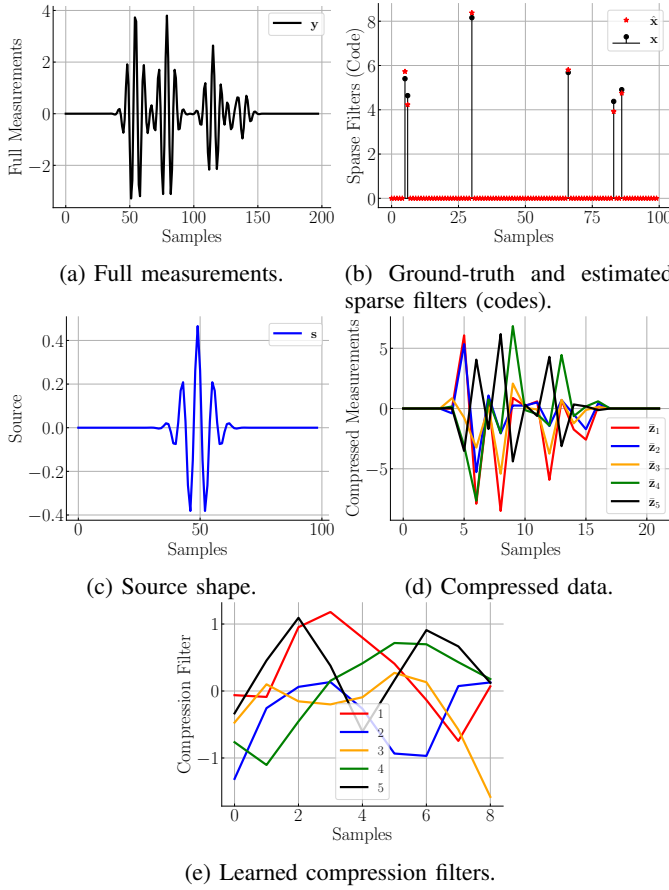


Fig. 7: Data visualizations on LS-MBD-LCISTA when the compression method of filter bank is used. The source has a wavelet shape, and the compression ratio is %55.55.

noise realization. The reported results are from the test set. For example, the reported results for the hit rate are averaged over 100 examples. Fig. 8 shows that adding a denoiser to the unrolled network results in an improved performance both in terms of NRMSE and exact hit rate.

For when SNR is 15 dB, we studied the effect of unrolling on denoising and target recovery. We trained LS-MBD-LCISTA-denoiser when  $T = 2, 5$ , and 10. Figure 9 demonstrates that increasing the unrolling layers improves the performance.

### G. Fewshot learning

In this section, we show the advantage of our unrolled learning (Fig. 2b) approach to conventional deep neural networks (Fig. 2a) such as DnCNN [34] in fewshot learning. We let  $CR = 55.5$  and choose the source shape as a symmetric wavelet, which we assume is known. We utilized the filter banks compression (i.e., using 5 compression filters with a kernel size of  $9 \times 9$  and strides of 9). For compressed DnCNN, after  $\Phi^T$ , there is an additional fully-connected layer to go from  $M_y$  to  $M_x$  before the DnCNN network which its input and output have the same dimension. DnCNN uses kernel sizes of 3, and 64 feature maps. We set the depth of DnCNN to only 3 layers lead to 32,645 trainable parameters compared to

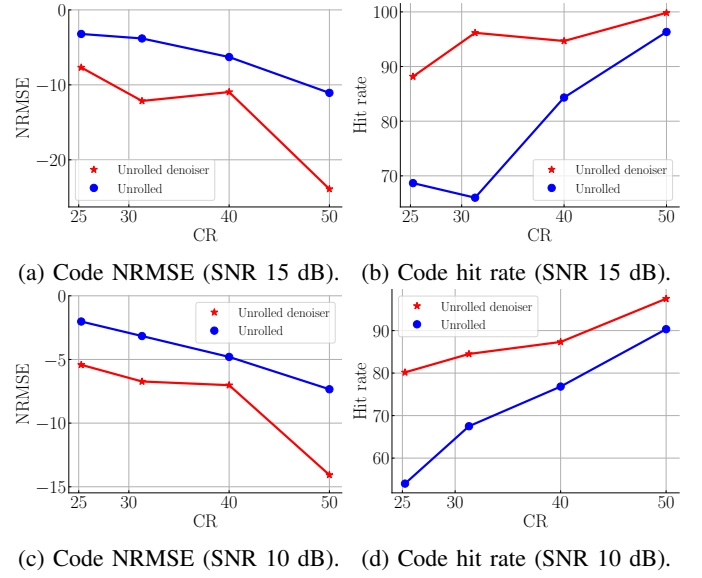


Fig. 8: Outperformance of LS-MBD denoiser against LS-MBD in the presence of additive Gaussian noise. For SNR of 15 and 10 dB,  $\sigma = 0.161$  and  $0.297$ , respectively. When SNR is 10 dB, the variance of the reported average hit rate for LS-MBD-LCISTA is 1.25, 2.52, 3.39, and 3.21 for a compression ratio of 50, 40, 31.31, and 25.25, respectively. For the same scenario, the variance over the reported hit rate of LS-MBD-LCISTA-denoiser is 0.30, 3.05, 2.76, and 2.49. Moreover, when SNR is 15 dB, the variance of the reported average hit rate for LS-MBD-LCISTA is 0.65, 3.13, 2.4, 3.49 for a compression ratio of 50, 40, 31.31, and 25.25, respectively. For the same scenario, The variance over the reported hit rate of LS-MBD-LCISTA-denoiser is 0.30, 0.96, 0.73, and 2.32.

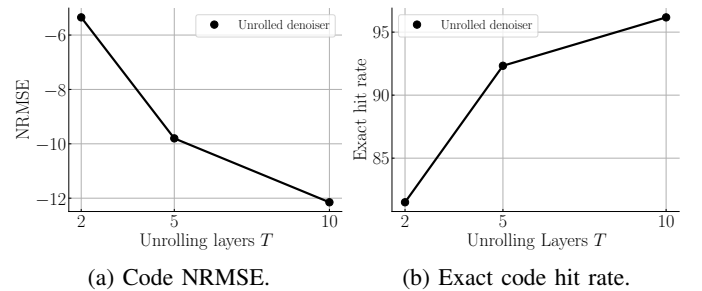


Fig. 9: Effect of unrolling layers on the performance of LS-MBD denoiser in the presence of additive Gaussian noise. For SNR of 15, with  $\sigma = 0.161$ .

442 of unrolling method; this is to alleviate the overfitting of DnCNN in low data regime.

As the amount of training data decreases from 10,000 to 100, the performance of compressed DnCNN significantly degrades, as opposed to our unrolling approach of LS-MBD-LCISTA (Fig. 10). We observed that unrolling approach outperforms DnCNN for all ranges of data. Most importantly, in a low data regime, DnCNN breaks down and reaches 0 dB NRMSE and an exact hit rate below %40 when there are only 100 training examples.

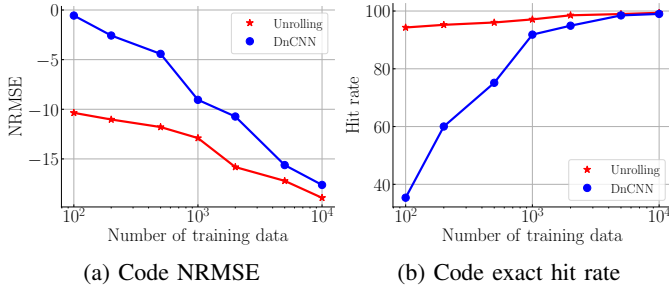


Fig. 10: Fewshot learning. Unrolling significantly outperforms conventional neural networks DnCNN in a data-limited regime.

## VI. CONCLUSION

We studied the problem of sparse multichannel blind deconvolution (S-MBD). We proposed a filter-based compression to enable recovery from a small number of measurements. We theoretically studied the identifiability problem and provide conditions under which recovery is possible. We derived the necessary and sufficient conditions on the compression filter and the source for unique recovery. Through an unrolled learning framework, we proposed to learn the compression filter, resulting in better recovery from higher compression rates. Specifically, we designed a structured inference network based on an  $\ell_1$ -based iterative sparse coding algorithm to map compressed measurements to the sparse filters of interest. We demonstrated the robustness and superior performance of our proposed framework compared to traditional optimization-based methods. Moreover, we demonstrated the generalization capability of unrolled deep learning compared to generic deep learning in applications with data-limited regimes.

## APPENDIX

In this section, we show that the D-SoS filter together with a source satisfying non-vanishing spectral properties [12] will follow the spark properties of Theorem 1. The impulse response of the D-SoS filter considered in [12] is given as

$$\mathbf{h}[m] = \sum_{k \in \mathcal{K}} e^{jk\omega_0 m}, \quad 0 \leq m \leq M_h - 1, \quad (26)$$

where  $|\mathcal{K}| = 2L^2$  for sufficiency. With the D-SoS filter, the entries of the matrix  $\mathbf{A}_{(M_y+M_x-1, 2M_x-1)}$  for any source  $\mathbf{s}_0 \in \mathbb{R}^{M_{s_0}}$  with  $M_{s_0} < M_h$  are given as

$$\mathbf{A}[m, l] = \sum_{k \in \mathcal{K}} S_0(e^{jk\omega_0}) e^{jk\omega_0(M_y+M_x+m-l-2)}, \quad (27)$$

where  $S_0(e^{j\omega})$  is the discrete-time Fourier transform of  $\mathbf{s}_0$  and  $0 \leq m \leq M_h - M_y - M_x + 1$ ,  $0 \leq l \leq 2M_x - 1$ . The non-vanishing spectral condition states that  $S_0(e^{j\omega}) \neq 0$  for  $k \in \mathcal{K}$ . Then by assuming that  $M_h \geq M_y + M_x + |\mathcal{K}|$ , the matrix  $\mathbf{A}_{(M_y+M_x-1, 2M_x-1)}$  can be decomposed as

$$\mathbf{A}_{(M_y+M_x-1, 2M_x-1)} = \mathbf{V}_1 \text{diag}(\mathbf{d}\{S_0(k\omega_0)\}_{k \in \mathcal{K}}) \mathbf{V}_2^*, \quad (28)$$

where entries of  $|\mathcal{K}|$  length vector are consists of elements in the set  $\{S_0(k\omega_0)e^{jk\omega_0(M_y+M_x-2)}\}_{k \in \mathcal{K}}$ . The matrices  $\mathbf{V}_1 \in$

$\mathbb{C}^{M_y+M_x-1 \times |\mathcal{K}|}$  and  $\mathbf{V}_2 \in \mathbb{C}^{2M_x-1 \times |\mathcal{K}|}$  are Vandermonde and with their  $(m, k)$ -th element is  $e^{jk\omega_0 m}$ . As we discussed in the proof of Theorem 1, the number of rows of  $\mathbf{A}_{(M_y+M_x-1, 2M_x-1)}$  should be greater than or equal to  $2L^2$  for the spark condition to hold in the sufficient part. Hence, the matrix  $\mathbf{V}_1$  is full column rank. In [12], it was assumed that  $S_0(k\omega_0) \neq 0, k \in \mathcal{K}$  which ensures that the matrix  $\mathbf{V}_1 \text{diag}(\mathbf{d}\{S_0(k\omega_0)\}_{k \in \mathcal{K}})$  is full rank. Hence, spark of matrix  $\mathbf{A}_{(M_y+M_x-1, 2M_x-1)}$  should be equal to that of matrix  $\mathbf{V}_2^*$ . Given that  $\omega_0$  and  $\mathcal{K}$  form universal sampling pattern [12] an  $|\mathcal{K}|$  columns of matrix  $\mathbf{V}_2^*$  are linearly independent and hence the matrix has full spark [12]. Hence,  $\text{Spark}(\mathbf{A}_{(M_y+M_x-1, 2M_x-1)}) > 2L^2$ .

## REFERENCES

- [1] M. A. Herman and T. Strohmer, "High-resolution radar via compressed sensing," *IEEE trans. signal process.*, vol. 57, no. 6, pp. 2275–2284, 2009.
- [2] K. Nose-Filho, A. K. Takahata, R. Lopes, and J. M. T. Romano, "Improving sparse multichannel blind deconvolution with correlated seismic data: Foundations and further results," *IEEE Signal Process. Mag.*, vol. 35, no. 2, pp. 41–50, Mar. 2018.
- [3] G. Carter, "Time delay estimation for passive sonar signal processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 29, no. 3, pp. 463–470, Jun. 1981.
- [4] R. Tur, Y. C. Eldar, and Z. Friedman, "Innovation rate sampling of pulse streams with application to ultrasound imaging," *IEEE Trans. Signal Process.*, vol. 59, no. 4, pp. 1827–1842, Apr. 2011.
- [5] N. Wagner, Y. C. Eldar, and Z. Friedman, "Compressed beamforming in ultrasound imaging," *IEEE Trans. Signal Process.*, vol. 60, no. 9, pp. 4643–4657, Sep. 2012.
- [6] W. U. Bajwa, K. Gedalyahu, and Y. C. Eldar, "Identification of parametric underspread linear systems and super-resolution radar," *IEEE Trans. Signal Process.*, vol. 59, no. 6, pp. 2548–2561, Jun. 2011.
- [7] O. Bar-Ilan and Y. C. Eldar, "Sub-Nyquist radar via Doppler focusing," *IEEE Trans. Signal Process.*, vol. 62, no. 7, pp. 1796–1811, Apr. 2014.
- [8] L. Wang and Y. Chi, "Blind deconvolution from multiple sparse inputs," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1384–1388, Oct. 2016.
- [9] N. Kazemi and M. D. Sacchi, "Sparse multichannel blind deconvolution," *Geophysics*, vol. 79, no. 5, pp. V143–V152, 2014.
- [10] C. Bilen, G. Puy, R. Gribonval, and L. Daudet, "Convex optimization approaches for blind sensor calibration using sparsity," *IEEE Trans. Signal Process.*, vol. 62, no. 18, pp. 4847–4856, Sep. 2014.
- [11] R. Gribonval, G. Chardon, and L. Daudet, "Blind calibration for compressed sensing by convex optimization," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, Mar. 2012, pp. 2713–2716.
- [12] S. Mulleti, K. Lee, and Y. C. Eldar, "Identifiability conditions for compressive multichannel blind deconvolution," *IEEE Trans. Signal Process.*, vol. 68, pp. 4627–4642, 2020.

- [13] C. Garcia-Cardona and B. Wohlberg, "Convolutional dictionary learning: A comparative review and new algorithms," *IEEE Trans. Comput. Imag.*, vol. 4, no. 3, pp. 366–381, 2018.
- [14] Y. Li, K. Lee, and Y. Bresler, "Identifiability in bilinear inverse problems with applications to subspace or sparsity-constrained blind gain and phase calibration," *IEEE Trans. Info. Theory*, vol. 63, no. 2, pp. 822–842, Feb. 2017.
- [15] A. Cosse, "A note on the blind deconvolution of multiple sparse signals from unknown subspaces," *Proc. SPIE*, vol. 10394, 2017.
- [16] B. Tolooshams, S. Dey, and D. Ba, "Scalable convolutional dictionary learning with constrained recurrent sparse auto-encoders," in *Proc. Workshop on Machine Learning for Signal Process. (MLSP)*, 2018, pp. 1–6.
- [17] B. Tolooshams, S. Dey, and D. Ba, "Deep residual autoencoders for expectation maximization-inspired dictionary learning," *IEEE Trans. Neural Netw. Learn. Syst.*, pp. 1–15, 2020.
- [18] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing," *arXiv:1912.10557*, 2019.
- [19] N. Shlezinger, Y. C. Eldar, and S. P. Boyd, "Model-based deep learning: On the intersection of deep learning and optimization," *arXiv:2205.02640*, 2022.
- [20] T. Chang, B. Tolooshams, and D. Ba, "Randnet: Deep learning with compressed measurements of images," in *Proc. Workshop on Machine Learning for Signal Process. (MLSP)*, 2019, pp. 1–6.
- [21] Y. Li, K. Lee, and Y. Bresler, "Blind gain and phase calibration via sparse spectral methods," *IEEE Trans. Info. Theory*, vol. 65, no. 5, pp. 3097–3123, May 2019.
- [22] S. Mulleti, H. Zhang, and Y. C. Eldar, "Learning to sample for sparse signals," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, 2022, pp. 3363–3367.
- [23] B. Tolooshams, S. Mulleti, D. Ba, and Y. C. Eldar, "Unfolding neural networks for compressive multichannel blind deconvolution," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, 2021, pp. 2890–2894.
- [24] L. Balzano and R. Nowak, "Blind calibration of sensor networks," in *In Proc. Int. Sym. Info. Process. Sensor Net.*, Apr. 2007, pp. 79–88.
- [25] A. Mousavi, G. Dasarathy, and R. G. Baraniuk, "Deep-codec: Adaptive sensing and recovery via deep convolutional neural networks," in *Annual Allerton Conf. Commu., Control, Computing (Allerton)*, 2017, pp. 744–744.
- [26] A. Mousavi and R. G. Baraniuk, "Learning to invert: Signal recovery via deep convolutional networks," in *Proc. IEEE Int. Conf. Acoust., Speech and Signal Process. (ICASSP)*, 2017, pp. 2272–2276.
- [27] B. Tolooshams, A. Song, S. Temereanca, and D. Ba, "Convolutional dictionary learning based auto-encoders for natural exponential-family distributions," in *Proc. Int. Conf. Machine Learning*, ser. Proceedings of Machine Learning Research, H. D. III and A. Singh, Eds., vol. 119. PMLR, 13–18 Jul 2020, pp. 9493–9503.
- [28] I. Daubechies, M. Defrise, and C. De Mol, "An iterative thresholding algorithm for linear inverse problems with a sparsity constraint," *Commun. Pure Applied Math.*, vol. 57, no. 11, pp. 1413–1457, 2004.
- [29] B. Tolooshams and D. E. Ba, "Stable and interpretable unrolled dictionary learning," *Trans. Machine Learning Research*, 2022.
- [30] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proc. Int. Conf. Machine Learning (ICML)*, 2010, pp. 399–406.
- [31] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [32] D. Palomar and Y. C. Eldar, *Convex Optimization in Signal Processing and Communications*. Cambridge University Press, 2009.
- [33] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.
- [34] K. Zhang, W. Zuo, Y. Chen, D. Meng, and L. Zhang, "Beyond a Gaussian denoiser: Residual learning of deep CNN for image denoising," *IEEE trans. image process.*, vol. 26, no. 7, pp. 3142–3155, 2017.