

Deep Spectrum Cartography: Completing Radio Map Tensors Using Learned Neural Models

Sagar Shrestha, Xiao Fu[✉], Senior Member, IEEE, and Mingyi Hong[✉], Senior Member, IEEE

Abstract—The *spectrum cartography* (SC) technique constructs multi-domain (e.g., frequency, space, and time) radio frequency (RF) maps from limited measurements, which can be viewed as an ill-posed tensor completion problem. Model-based cartography techniques often rely on handcrafted priors (e.g., sparsity, smoothness and low-rank structures) for the completion task. Such priors may be inadequate to capture the essence of complex wireless environments—especially when severe shadowing happens. To circumvent such challenges, offline-trained deep neural models of radio maps were considered for SC, as deep neural networks (DNNs) are able to “learn” intricate underlying structures from data. However, such deep learning (DL)-based SC approaches encounter serious challenges in both off-line model learning (training) and completion (generalization), possibly because the latent state space for generating the radio maps is prohibitively large. In this work, an emitter radio map disaggregation-based approach is proposed, under which only individual emitters’ radio maps are modeled by DNNs. This way, the learning and generalization challenges can both be substantially alleviated. Using the learned DNNs, a fast nonnegative matrix factorization-based two-stage SC method and a performance-enhanced iterative optimization algorithm are proposed. Theoretical aspects—such as recoverability of the radio tensor, sample complexity, and noise robustness—under the proposed framework are characterized, and such theoretical properties have been elusive in the context of DL-based radio tensor completion. Experiments using synthetic and real-data from indoor and heavily shadowed environments are employed to showcase the effectiveness of the proposed methods.

Index Terms—Spectrum cartography, radio map, deep learning, tensor completion.

I. INTRODUCTION

ADIO frequency (RF) awareness is the stepping stone towards smart, intelligent, and high-efficiency wireless

Manuscript received April 30, 2021; revised November 5, 2021 and December 21, 2021; accepted January 9, 2022. Date of publication January 25, 2022; date of current version March 11, 2022. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Youngchul Sung. The work of Sagar Shrestha and Xiao Fu was supported in part by the NSF MLWiNS Program under Project NSF CNS-2003082 and in part by NSF under Grant ECCS-2024058. The work of Mingyi Hong was supported in part by the NSF MLWiNS Program under Project NSF CNS-2003082 and in part by ARO under Grant W911NF-19-1-0247. This work was also supported by Intel through the MLWiNS Program. (*Corresponding author: Xiao Fu*.)

Sagar Shrestha and Xiao Fu are with the School of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR 97331 USA (e-mail: shressag@oregonstate.edu; xiao.fu@oregonstate.edu).

Mingyi Hong is with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN 55455 USA (e-mail: mhong@umn.edu).

This article has supplementary downloadable material available at <https://doi.org/10.1109/TSP.2022.3145190>, provided by the authors.

Digital Object Identifier 10.1109/TSP.2022.3145190

communication systems. Sensing the RF environment has been at the heart of the physical layer technologies, especially since the vision of cognitive radio was advocated by the *Federal Communications Commission* (FCC) [1]. Starting from the early 2000s, a plethora of *spectrum sensing* techniques have been proposed to estimate the spectral usage; see, e.g., [2]–[4] and a recent survey [5]. A step forward is the *spectrum cartography* (SC) technique, which aims at crafting a radio power propagation map across a geographical region from (often sparsely acquired and limited) sensor measurements [6]–[8]. SC produces radio maps that reveal critical information of the RF environment across multiple domains, e.g., time, frequency, and space. Hence, SC is considered useful for a wide range of tasks in wireless communications—e.g., spectrum surveillance, opportunistic access, interference-free routing and networking, and mobile relay placement; see [9].

On the other hand, SC is a highly nontrivial task. Sensors are often sparsely deployed in the geographical region of interest, and can only acquire power measurements *locally* in space. Reconstructing the full radio map across multiple domains poses a multi-aspect high-dimensional data inverse problem using heavily down-sampled measurements—which is known to be challenging in both theory and methods [10], [11].

Many SC approaches have emerged since the late 2000s. For example, the work in [7] employed a widely used geostatistical tool, namely, Kriging interpolation [7], which treats the SC problem as a 2D field interpolation problem (implicitly) under statistical priors (e.g., the Gaussian prior). The work in [6], [12] proposed a compressive sensing-based SC framework that exploits the spatial sparsity of emitter locations—and the work in [4] additionally exploits the sparsity of spectral usage by the emitters. The methods in [13] took a dictionary learning/sparse coding perspective for SC, which exploit spatio-temporal correlations and the associated sparse representations. To further exploit the spatial smoothness/correlation of the radio maps, the approaches in [8], [14] used representation tools such as *thin plate splines* (TPS) and *radial basis functions* (RBF) to enable interpolation; also see a sensor signal-quantized version in [15]. Recently, the work in [16] models the radio map as a low-rank block-term tensor and offered a tensor recovery-based SC approach, which is again a means to advantage of the spatial smoothness of radio maps.

The existing SC approaches are effective to a certain extent, but a number of challenges remain. Some approaches may need a number of environmental parameters, e.g., the point-to-point propagation model [6], basis function for representing the

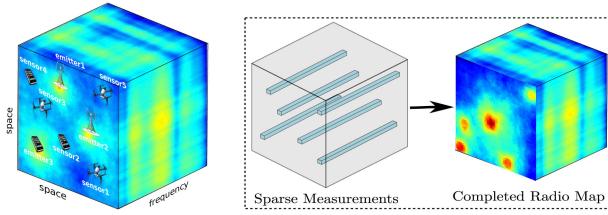


Fig. 1. (Left) Illustration of a spatio-spectral radio map tensor. (Right) The SC scenario of interest: completing the radio map from sensed tensor “fibers”.

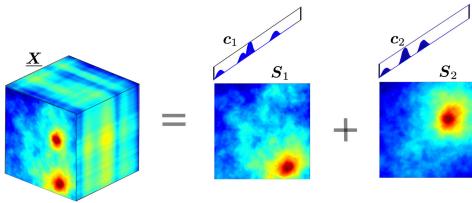


Fig. 2. Illustration of (1) for $R = 2$ case.

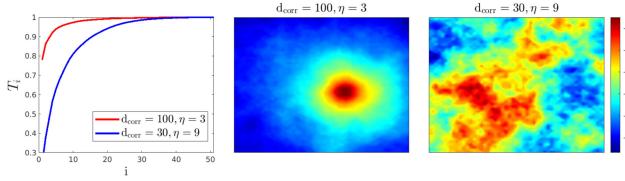


Fig. 3. Shadowing effect affects the soundness of the low-rank assumption on the 50×50 SLFs. (Left) T_i for all $i \in [50]$ where $T_i = \sum_{k=1}^i \sigma_k / \sum_{k=1}^I \sigma_k$ and σ_i is the i th singular value of the SLF. (Middle) An SLF with little shadowing effect. (Right) An SLF with heavy shadowing effect. The SLFs are generated following the model in [29]; d_{corr} is the decorrelation distance and η is the shadowing variance (see Section V).

emitters’ power spectral density (PSD) [8], or even the exact PSDs of the emitters [15]—which are not always easy to acquire, especially in fast changing environments. More importantly, handcrafted model priors (e.g., sparsity, smoothness, low rank) are often inadequate to capture the complex underlying structure of the radio maps. Such modeling mismatches could lead to substantial performance degradation. For example, the tensor-based method in [16] is built upon the premise that the spatial power propagation pattern of each emitter gives rise to a low-rank matrix. However, this assumption is often violated in geometrically complicated environments, e.g., urban and indoor areas, where many blockages and barriers exist and the shadowing effect is severe (cf. Fig. 3). Many temporal/spatial smoothness based approaches, e.g., [8], [14], share the same challenges.

To circumvent such modeling uncertainties, data-driven methods were recently proposed in [17], [18] for SC. Instead of hinging on pre-specified and handcrafted model priors, these methods “learn” from training samples and leverage the expressiveness of deep neural networks (DNNs) to represent radio data with complex underlying structures. The methods in [17], [18] learn “completion networks” offline, and then feed the sensor-acquired measurements through the network to produce an estimated radio map.

On one hand, DL-based SC has potentials in dealing with intricate RF environments. On the other hand, purely data-driven

methods have their own challenges in both the offline network learning (training) and online radio map completion (generalization) stages. Training a DNN to represent a multi-domain high-dimensional radio tensor is computationally challenging. In particular, the state space of the radio maps is often prohibitively large. The size of the state space is affected by the number of emitters, the frequency usage of them, the locations of them—and all possible combinations. Consequently, one may need an unexpectedly large number of training samples to learn the network. Hence, the training stage may involve a large number of samples to ‘cover’ representative states—which induce heavy computational loads and overall harder optimization problems. In addition, the generalization stage also has a number of challenges. Specifically, even if very large training sets have been used, many “out-of-training-distribution” cases can still easily happen in the completion stage—e.g., when R emitters are used for generating the training samples but some extra emitters move into the scene, as a consequence, the networks trained with a smaller number of emitters often produce radio maps that miss some emitters.

Contributions.

In this work, we propose a *hybrid* model and data-driven SC framework. Specifically, our method combines a radio map disaggregation model with DNN-based spatial power propagation pattern learning. The idea is only using DNNs to represent the most complex part in radio map models, instead of the entire radio map—in order to alleviate the training and generalization challenges. Our detailed contributions are as follows:

- **Emitter Disaggregated Neural Modeling.** Motivated by the fact that the overall interference radio map is an aggregation of a number of emitters’ individual radio maps, we propose to learn and represent each emitter’s spatio-spectral radio map (as opposed to the ambient radio map itself) using deep networks. Intuitively, *single* emitter’s radio maps reside in a much smaller state space compared to that of the aggregated maps. Using our disaggregated modeling also effectively overcomes the out-of-distribution generalization problem in the completion (test) stage.

- **Recoverability-Guaranteed Realizations.** After the disaggregated deep neural model is trained, we offer two algorithms to realize SC. (i) We propose a two-stage method. The first stage leverages the sparsity of spectral occupancy of the emitters and a greedy *nonnegative matrix factorization* (NMF) approach [19], [20] to decompose the sensor measurements to emitter components. Then, the second stage uses an off-line trained deep completion network to estimate the individual emitters’ radio maps. (ii) When the spectral usage of the emitters is not sparse, we propose a deep generative model regularized radio map completion criterion for our task—with the price paid on a heavier computational burden. We provide performance characterizations for our methods—including recoverability, noise robustness, and sample complexity—which have been elusive in the context of DL-based SC.

- **Synthetic and Real Data Validation.** We validate the proposed methods over a variety of synthetic data experiments, as well as a real data experiment with radio map measurements from an indoor area at University of Mannheim, Germany [21].

TABLE I
LIST OF ABBREVIATIONS

Abbreviations	Definitions
SLF	Spatial Loss Function
PSD	Power Spectral Density
SC	Spectrum Cartography
DNN	Deep Neural Network
DL	Deep Learning
NNLS	Nonnegativity constrained Least Squares
SPA	Successive Projection Algorithm
AO	Alternating optimization
NMF	Nonnegative Matrix Factorization
NAE	Normalized Absolute Error
SRE	Squared Reconstruction Error

Both the simulations and the real experiments corroborate our design, and show that our framework offers promising SC performance in challenging scenarios.

A conference version of this work has appeared in ICASSP 2021 [22], which includes the NMF-assisted SC approach under sparse emitter spectral occupancy. In this journal version, we additionally include a new iterative algorithm that deals with more challenging dense spectral occupancy cases, its radio map recoverability and stability analyses, more extensive simulations, and a real data experiment.

Notation: $x \in \mathbb{R}$, $\mathbf{x} \in \mathbb{R}^K$, $\mathbf{X} \in \mathbb{R}^{I \times K}$, $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ represent a scalar, a vector, a matrix and a tensor, respectively. \mathbf{x}_i represents the i th column of matrix \mathbf{X} . The Matlab notations $\mathbf{X}(:, j)$ and $\mathbf{X}(i, :)$ represent the j th column and i th row of matrix \mathbf{X} , respectively; similar Matlab notations are used for vectors and tensors as well. \circ and \otimes represent the outer product and Hadamard product, respectively; in particular, if $\mathbf{X} \in \mathbb{R}^{I \times J}$ and $\mathbf{y} \in \mathbb{R}^K$, then $\underline{\mathbf{Z}} = \mathbf{X} \circ \mathbf{y}$ is an $I \times J \times K$ tensor such that $\underline{\mathbf{Z}}(i, j, k) = \mathbf{X}(i, j)\mathbf{y}(k)$. $\|\mathbf{x}\|_2$ and $\|\mathbf{X}\|_2$ denote the vector 2-norm and the matrix spectral norm, respectively. $\|\cdot\|_{\text{F}}$ represents the matrix and tensor Euclidean norm. \mathbf{X}^\top denotes the transpose of \mathbf{X} . \geq , \leq , $>$ and $<$ represent element-wise inequality. $\mathbf{0}$ and $\mathbf{1}$ denote all-zero and all-one matrices/vectors with proper size, respectively. $|\mathcal{X}|$ denotes the cardinality of set \mathcal{X} . $[I] = \{1, 2, \dots, I\}$ where I is an integer. A list of the abbreviations used in this paper can be found in Table I.

II. PROBLEM STATEMENT AND BACKGROUND

In this section, we present the problem setup and briefly introduce existing approaches.

A. Problem Setup and Signal Model

We are interested in estimating a spatio-spectral radio map of a certain geographical region across a number of frequency bins. The region has R emitters that transmit their signals across certain (possibly overlapping) frequency bands, which together create a power interference field across space and frequency. We consider a rectangular region discretized into $I \times J$ grids and K frequency bins. Hence, the (discretized) *power spectral density* (PSD) measured over space and frequency gives rise to a 3D radio map that can be expressed as an $I \times J \times K$ tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$, where $\underline{\mathbf{X}}(i, j, k)$ denotes the PSD of the signal received at location (i, j) and frequency k ; see Fig. 1 (left). We should mention that although the instantaneous radio map may change quickly, the PSD captures the average spectrum usage over a certain period of time, if the signals are stationary random processes. Hence, estimating PSD-based radio maps is often considered more realistic than estimating the instantaneous situation; see, e.g., [3], [4], [8], [23], [24].

Suppose that there are a number of sensors deployed within this region. We assume that a sensor located at position (i, j) where $i \in [I]$ and $j \in [J]$ can measure the power spectral density (PSD) of its received signal locally over all K frequency bins; i.e., the sensor located at (i, j) can acquire the PSD $\mathbf{p}_{(i,j)} \in \mathbb{R}^K$ of the received signal such that $\mathbf{p}_{(i,j)} = \underline{\mathbf{X}}(i, j, :) \in \mathbb{R}^K$, where K is the number of frequency bins. Note that a vector $\underline{\mathbf{X}}(i, j, :)$ is also sometimes called a “*tensor fiber*” [25], [26]. If every location (i, j) has a sensor, a radio map tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ would be constructed. However, in practice, placing IJ sensors to “cover” the geographical region may not be an viable option. Instead, only a small number of sensors are sparsely placed within the region of interest. In other words, letting

$$\Omega = \{(i, j) | i \in [I], j \in [J]\} \subseteq [I] \times [J]$$

denote the set of sensor locations, we often have $|\Omega| \ll IJ$. Hence, the task of spatio-spectral SC is to estimate the full radio map tensor $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ using the sensor-acquired measurements (i.e., tensor fibers) $\underline{\mathbf{X}}(i, j, :) \in \mathbb{R}^K, \forall (i, j) \in \Omega$ —see Fig. 1 (right) for an illustration.

If the spectral band of interest is not large relative to the central carrier frequency (e.g., 20 MHz bandwidth at a central carrier frequency within 2–5 GHz.), the spatial propagation pattern of an emitter across different frequencies are identical up to scaling differences caused by the emitter’s PSD. Consequently, if noise is absent, the full radio map can be expressed as follows [8], [16]:

$$\underline{\mathbf{X}}(i, j, k) = \sum_{r=1}^R \mathbf{S}_r(i, j) \mathbf{c}_r(k) \iff \underline{\mathbf{X}} = \sum_{r=1}^R \mathbf{S}_r \circ \mathbf{c}_r, \quad (1)$$

where $\mathbf{S}_r \in \mathbb{R}^{I \times J}$ denotes the *spatial loss function* (SLF) of the power of emitter r , $\mathbf{c}_r \in \mathbb{R}^K$ represents the PSD of emitter r , \circ is the outer product, and R is the number of emitters in the region of interest. An illustration is shown in Fig. 2. A remark is that the factorization based expression of a radio map also makes storing $\underline{\mathbf{X}}$ easier. The memory cost of $\underline{\mathbf{X}} \in \mathbb{R}^{I \times J \times K}$ under (1) is only $O((IJ + K)R)$ other than $O(IJK)$. The memory is even lower when \mathbf{S}_r can be parameterized, e.g., using a low-rank matrix as in [16] or a deep generative model as will be seen in this work.

B. Prior Art

In essence, the SC problem is an ill-posed inverse problem. Hence, the same insights employed in classic inverse problems, e.g., compressive sensing and low-rank matrix/tensor completion may be used for SC. Indeed, early works exploited the spatial sparsity of emitter locations [6], sparse representation of emitter PSDs in a certain learned domain [4], [13], spatial smoothness of the SLFs [8], and a low-rank tensor structure [16] to come up with radio map recovery formulations. Notably, the

most recent work in [16] assumed that $\text{rank}(\underline{\mathbf{S}}_r) \leq L$ where $L \ll \min\{I, J\}$ and recast the SC problem as a low-rank *block-term tensor* [27] completion problem. This way, recoverability guarantees of the radio map from sparse sensor measurements can be established.

Structural priors/constraints-assisted inverse problem formulations are useful, but they are often inadequate in dealing with complex scenarios. For example, the low-rank assumption on $\underline{\mathbf{S}}_r$ is plausible in free space or in rural areas where electromagnetic waveforms could propagate without encountering many barriers. However, the assumption becomes unrealistic when urban or indoor regions are considered. The same challenge is also shared by methods that use spatial smoothness; see, e.g., [8], [28]. Fig. 3 shows how heavy shadowing (caused by barriers like buildings and walls) makes the low-rank and spatial smoothness assumptions on $\underline{\mathbf{S}}_r$ less accurate. In this example, the first 10 principal components of a 50×50 SLF with little shadowing contain more than 90% energy of the SLF. However, the same number of principal components only contain less than 70% energy of the SLF when the shadowing effect is severe.

Very recently, a couple of data-driven methods were proposed to circumvent the model mismatch challenge. The idea is to learn the complex data generative process (from partial observations) using DNNs through a large number of realistic simulated data samples, and then use the learned neural model to assist radio map completion [17], [18]. For example, the work in [18] generates a training set, denoted by $\{\underline{\mathbf{X}}_n\}_{n=1}^N$, where each $\underline{\mathbf{X}}_n \in \mathbb{R}^{I \times J \times K}$ is a radio map tensor following a spatial propagation model [29]. Then, a deep completion network is trained using the following:

$$\hat{\theta} \leftarrow \arg \min_{\theta} \sum_{n=1}^N \|\mathbf{f}_{\theta}(\underline{\mathbf{M}}_n \circledast \underline{\mathbf{X}}_n) - \underline{\mathbf{X}}_n\|_F^2, \quad (2)$$

where $\mathbf{f}_{\theta}(\cdot) : \mathbb{R}^{I \times J \times K} \rightarrow \mathbb{R}^{I \times J \times K}$ is a DNN whose network weights are collected in θ ; $\underline{\mathbf{M}}_n \in \mathbb{R}^{I \times J \times K}$ is a mask tensor (or, a sampler) with $\underline{\mathbf{M}}_n(i, j, k) = 1$ if the entry in the radio map is sensed, and $\underline{\mathbf{M}}_n(i, j, k) = 0$ otherwise.¹ After a network $\mathbf{f}_{\hat{\theta}}$ is learned (via any off-the-shelf DNN training algorithms, e.g., [30]), a radio map can be simply estimated via $\widehat{\underline{\mathbf{X}}} = \mathbf{f}_{\hat{\theta}}(\underline{\mathbf{M}}_{\text{sens.}} \circledast \underline{\mathbf{X}}_{\natural})$, where $\underline{\mathbf{X}}_{\natural}$ is the ground-truth radio map to estimate, $\underline{\mathbf{M}}_{\text{sens.}}$ is the mask tensor that reflects the sensor locations, and $\underline{\mathbf{M}}_{\text{sens.}} \circledast \underline{\mathbf{X}}_{\natural}$ represents the sensed measurements (i.e., tensor fibers). Specifically, we have $\underline{\mathbf{M}}_{\text{sens.}}(i, j, k) = 1$, $\forall k$, if $(i, j) \in \Omega$, and $\underline{\mathbf{M}}_{\text{sens.}}(i, j, k) = 0$ otherwise. The major idea of these methods is summarized in Fig. 4.

C. Challenges of Deep Learning-Based SC

The most appealing feature of the DL-based SC approaches is that the network $\mathbf{f}_{\hat{\theta}}$ could potentially complete complex radio maps (e.g., those with heavy shadowing) that handcrafted priors may struggle to model. However, a couple of challenges remain.

¹In practice, the idea in (2) is often carried out with different variations, e.g., slab by slab realization in [18] and [17]. The mask information is also fed to the DNNs together with the sensed data; see details in [18].

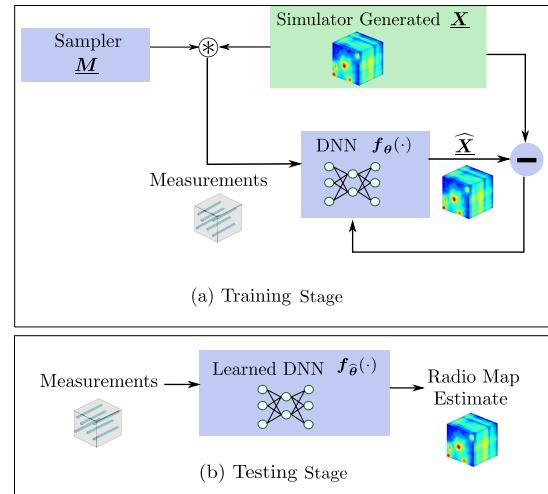


Fig. 4. Schematic of deep learning based SC methods in existing works [17], [18].

- **Training Challenge.** One premise for learning a high-quality $\mathbf{f}_{\hat{\theta}}$ is that the training samples $\{\underline{\mathbf{X}}_n\}_{n=1}^N$ should “cover” most representative radio maps under the scenario of interest; otherwise, the network could not make accurate inferences, if $\underline{\mathbf{X}}_{\natural}$ is not close to any sample in the training set. In SC, the state space contains all possible $\underline{\mathbf{X}}_n$ ’s, which may be too large. The reason is that there are too many combinations of key components, e.g., the number of emitters, their locations, transmission power levels, just to name a few. For example, if $R \leq 5$ and $I = J = 100$, the number of different emitter locations exceeds 8×10^{17} —without considering any other key parameters of the radio map. As a consequence, one may need a prohibitively large N to attain reasonable SC performance—making the training stage costly.

- **Generalization Challenge.** In addition, “out-of-training-distribution” test data can appear quite often. For instance, when R' emitters emerge in the region of area, but the network is trained with at most R emitters, where $R < R'$. This situation is particularly difficult for the off-line trained system to adapt because re-training can take a long time. However, the system designers have no control of the number of emitters, which means $R < R'$ may always happen. One may use a very large R in the training stage to circumvent this situation, but this may make the already hard and costly training problem even more challenging.

III. PROPOSED APPROACH

In this work, we propose an alternative paradigm for deep learning-assisted SC. In a nutshell, instead of using a purely data-driven method, we utilize the aggregation part in (1), as this part is intuitive and relatively reliable. In addition, we use deep networks to help learn the hard to model part in (1), i.e., the SLFs of individual emitters. As one will see, this model-assisted data-driven approach can effectively circumvent the training and generalization challenges encountered in pure DL methods, while keeping the expressive power of deep networks for accurately representing radio maps.

A. Training DNN for Completing Individual SLFs

Recall that the radio map can be modeled as $\underline{\mathbf{X}} = \sum_{r=1}^R \mathbf{S}_r \circ \mathbf{c}_r$, i.e., aggregated power intensity over space and frequency from R emitters. Due to the difficulty of modeling the \mathbf{S}_r 's using handcrafted priors, the performance of model-based methods often degrades in challenging cases, e.g., when heavy shadowing happens. Nonetheless, the model that a radio map is a superposition of individual radio maps associated with the emitters is still accurately reflecting the underlying physics. Hence, instead of learning a DNN to complete the entire aggregated radio map tensor $\underline{\mathbf{X}}$, our idea is to learn a network for completing the individual \mathbf{S}_r 's. If the \mathbf{c}_r 's can be somehow estimated, then one can assemble the estimated SLFs and PSDs to recover the entire radio map using (1).

Based on the discussion above, we propose the following process to learn a completion network for individual SLFs:

$$\hat{\theta} \leftarrow \arg \min_{\theta} \frac{1}{N} \sum_{n=1}^N \|f_{\theta}(\mathbf{M}_n \circledast \mathbf{Q}_n) - \mathbf{Q}_n\|_F^2, \quad (3)$$

where n is the training sample index, $\mathbf{M}_n \in \mathbb{R}^{I \times J}$ is the n th randomly generated 2D mask, $\mathbf{Q}_n \in \mathbb{R}^{I \times J}$ is a complete SLF of a single emitter, and $f_{\theta}(\cdot) : \mathbb{R}^{I \times J} \rightarrow \mathbb{R}^{I \times J}$ is the SLF completion network to be learned. Fig. 5 shows the training procedure of the proposed method and DL-based SC methods (e.g., [18]).

After f_{θ} is learned, we complete the SLFs by letting

$$\hat{\mathbf{S}}_r \leftarrow f_{\hat{\theta}}(\mathbf{M}_{\text{sens.}} \circledast \mathbf{S}_r), \quad (4)$$

where $\mathbf{M}_{\text{sens.}}$ is the 2D version of sensing mask, in which $\mathbf{M}_{\text{sens.}}(i, j) = 1$ if $(i, j) \in \Omega$ and $\mathbf{M}_{\text{sens.}}(i, j) = 0$ otherwise. The notation $\mathbf{M}_{\text{sens.}} \circledast \mathbf{S}_r$ represents the incomplete individual SLF of emitter r —and how to obtain it will be discussed in the next subsection.

Remark 1: Intuitively, the state space where the individual SLF resides is much smaller than that of the aggregated radio map $\underline{\mathbf{X}}$. Using the same example where 100×100 grids are considered, one can see that there are only 10^4 emitter locations if we only model a single SLF—as opposed to more than 8×10^{17} possible emitter location combinations if we consider up to 5 emitters in the same region. Such a substantial reduction of space size may make the training process more efficient and effective. In addition, if R can be properly estimated (using any model-order selection methods in signal processing, e.g., those in [31]), then the “out-of-distribution” and emitter misdetection problem encountered in existing DL-based SC approaches can be easily circumvented.

In this work, we employ an autoencoder structure for f_{θ} . Nonetheless, in principle, any generative model such as *generative adversarial network* (GAN) [32] and *variational autoencoder* (VAE) [33] can be readily employed into our framework. The autoencoder consists of two major parts, i.e., the encoder and the decoder, respectively. To be more precise, for an ideal autoencoder that is applied to high-dimensional data $\mathbf{Q} \in \mathbb{R}^{I \times J}$, we wish the following holds:

$$f_{\theta}(\mathbf{Q}) = g_{\theta_d}(p_{\theta_e}(\mathbf{Q})), \quad (5)$$

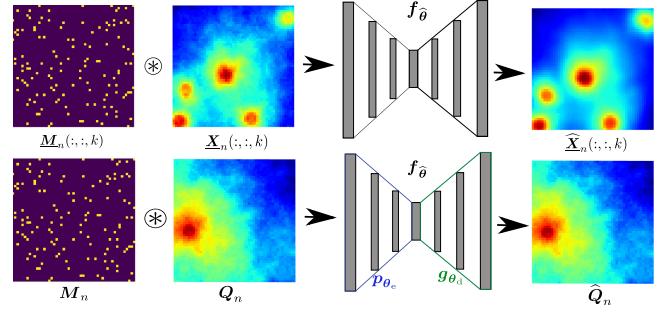


Fig. 5. Training procedures of existing DL-based SC methods (e.g., [18]) that learn DNNs to complete the aggregated $\underline{\mathbf{X}}$ (top) and the proposed method (bottom) that uses DL to complete individual SLFs.

in which θ_e and θ_d represents the network parameters of the encoder and decoder, respectively, $\theta = [\theta_e^\top, \theta_d^\top]^\top$, and

$$p_{\theta_e}(\cdot) : \mathbb{R}^{I \times J} \rightarrow \mathbb{R}^D, g_{\theta_d}(\cdot) : \mathbb{R}^D \rightarrow \mathbb{R}^{I \times J} \quad (6)$$

denote the encoder and decoder networks, respectively. Notably, if $D \ll IJ$, the model in (5) means that the data matrix $\mathbf{Q} \in \mathbb{R}^{I \times J}$ can be represented in a low-dimensional latent domain, using its “embedding” $\mathbf{z} = p_{\theta_e}(\mathbf{Q})$. The decoder can be regarded as a *generator* that maps a latent embedding \mathbf{z} to the observation domain, i.e., to “generate” the ambient data. Hence, the deep autoencoders are also considered as deep generative model learners.

B. Emitter Disaggregation-Assisted Deep Completion

The proposed individual SLF completion method could potentially reduce training costs and improve generalization. However, the inputs to such networks, i.e., $\mathbf{M}_{\text{sens.}} \circledast \mathbf{S}_r$ for all r , are not observed. Instead, if noise is absent, we observe the following incomplete tensor:

$$\underline{\mathbf{Y}} = \mathbf{M}_{\text{sens.}} \circledast \underline{\mathbf{X}}_{\natural}, \quad (7)$$

that is acquired by the sensors, where $\underline{\mathbf{X}}_{\natural} = \sum_{r=1}^R \mathbf{S}_r \circ \mathbf{c}_r$ is the target radio map that we aim to recover. If the \mathbf{c}_r 's can be somehow estimated, our objective boils down to estimating the incomplete individual SLFs, i.e., $\mathbf{M}_{\text{sens.}} \circledast \mathbf{S}_r$ from the observed measurements $\underline{\mathbf{Y}}$.

To this end, notice that the following relationship holds:

$$\mathbf{Y} = \text{unfold}(\underline{\mathbf{Y}}) = \text{unfold}(\mathbf{M}_{\text{sens.}} \circledast \underline{\mathbf{X}}_{\natural}), \quad (8)$$

where $\mathbf{X}_{\natural} = \text{unfold}(\underline{\mathbf{X}}_{\natural}) \in \mathbb{R}^{K \times IJ}$ and the tensor unfolding operator is defined as $\mathbf{X}_{\natural}(:, q) = \underline{\mathbf{X}}_{\natural}(i, j, :)$, $q = I(i-1) + j$.

Clearly, if we define $\mathbf{G} := \mathbf{Y}(:, \Omega_{\text{col}})$, where $\Omega_{\text{col}} \subseteq [IJ]$ such that $q = I(i-1) + j \in \Omega_{\text{col}}, \forall (i, j) \in \Omega$, then, we have the following expression for \mathbf{G} :

$$\mathbf{G} = \mathbf{C} \underbrace{\mathbf{S}(:, \Omega_{\text{col}})}_{\mathbf{H}} = \mathbf{C} \mathbf{H}, \quad (9)$$

where $\mathbf{C} = [\mathbf{c}_1, \dots, \mathbf{c}_R] \in \mathbb{R}^{K \times R}$ and $\mathbf{S} = [\text{vec}(\mathbf{S}_1), \dots, \text{vec}(\mathbf{S}_R)]^\top \in \mathbb{R}^{R \times IJ}$.

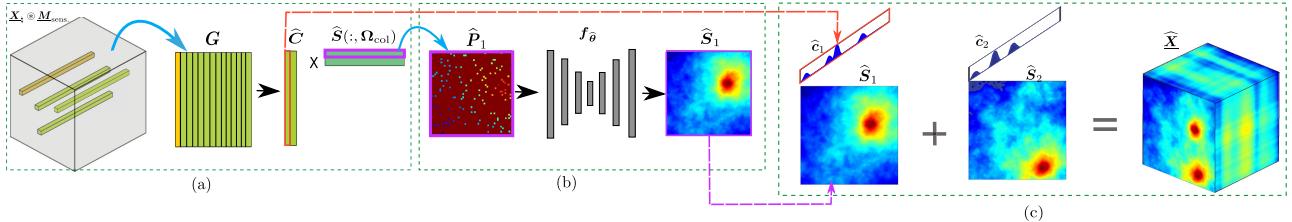


Fig. 6. Illustration for Nasdac algorithm. (a) Emitter Disaggregation model in (9). \widehat{C} and $\widehat{S}(:, \Omega_{\text{col}})$ are estimated by sparsity based NMF algorithm. (b) Deep Completion of emitter SLFs. r -th row of $\widehat{S}(:, \Omega_{\text{col}})$ is matricised and completed with completion network f_{θ} to obtain the estimate of completed r -th SLF, \widehat{S}_r . (c) Estimated \widehat{C} and \widehat{S} are aggregated to recover the estimate \widehat{X} .

If one could estimate H , then define a matrix $S_{\text{miss}} \in \mathbb{R}^{R \times IJ}$ such that $S_{\text{miss}}(r, \Omega_{\text{col}}) := H(r, :)$. Further, let $S_{\text{miss}}(:, \Omega_{\text{col}}^c) = \mathbf{0}$, where Ω_{col}^c is the complement of Ω_{col} . As a consequence, we have

$$\mathbf{P}_r = \text{mat}(S_{\text{miss}}(r, :)) = M_{\text{sens.}} \circledast S_r, \quad (10)$$

where the matricization operator $\text{mat}(\cdot) : \mathbb{R}^{IJ} \rightarrow \mathbb{R}^{I \times J}$ is the inverse operation of vectorization. Note that \mathbf{P}_r is the incomplete individual radio map that we wish to feed to $f_{\theta}(\cdot)$ in (4). The procedure for creating the matrix G is illustrated in Fig. 6(a).

Note that C and H are nonnegative matrices, per their physical meanings. Hence, our task amounts to estimating C and H from G —which is an NMF problem [34]. Then, we take the estimated H 's rows and apply deep network-based completion to estimate S_r for all r . With C and S_r 's estimated, we can recover \underline{X}_{\natural} ; see the overall procedure in Fig. 6.

Note that NMF is not always unique—i.e., estimating C and H from G may not be always possible; see discussions about NMF identifiability in the literature, e.g., [34]–[36]. Nonetheless, if the emitters transmit sparsely across the spectral band, estimating C and H via NMF is viable. To see this, we make the following assumption:

Assumption 1: There exists a frequency f_r for each emitter $r \in [R]$ such that $C(f_r, r) > 0$ and $C(f_r, k) = 0$ for $k \neq r$.

Assumption 1 is considered reasonable since emitters often take different carrier frequencies (or subbands in OFDM systems) [24]. In addition, if the emitters use the K frequencies in a sparse manner, such an assumption is often not hard to meet. Under Assumption 1, we show that:

Theorem 1: Under (1) and the construction of G in (9), assume that the entries of $S \in \mathbb{R}^{R \times IJ}$'s are drawn from any joint absolutely continuous distribution. Then, with probability one, there exists a polynomial time algorithm (e.g., the *successive projection algorithm* (SPA) [19], [20]) that estimates $\widehat{C} = C\Pi\Lambda$, $\widehat{H} = \Lambda^{-1}\Pi^\top H$ (Π and Λ being permutation and full-rank diagonal scaling matrices, respectively), given that the number of sensors satisfies $|\Omega| \geq R$.

The proof is relegated to Appendix B. Note that Assumption 1 translates to the *separability* condition in the context of NMF, and the proof of Theorem 1 is a direct application of identifiability of separable NMF [34], [36]. Note that under separability condition, NMF is solvable using either greedy algorithms [19], [20] or convex optimization [37], both of which are polynomial-time algorithms. In this work, we employ the Gram-Schmidt-like

Algorithm 1: Nasdac.

Data: $X_{\text{sens.}}, \Omega, R, f_{\theta}(\cdot)$.
Result: \widehat{X} , \widehat{S} and \widehat{C} .

// Stage 1: Emitter Disaggregation
1 $\widehat{C}, \widehat{H} \leftarrow \text{NMF_SPA}(G, R)$ (see Appendix G);
// Stage 2: SLF completion
2 **for** $r = 1 : R$ **do**
3 $\widetilde{S}(r, \Omega_{\text{col}}) \leftarrow \widehat{H}(r, :)$;
4 $\widetilde{S}(r, \Omega_{\text{col}}^c) \leftarrow \mathbf{0}$;
5 $\widehat{S}_r \leftarrow f_{\theta}(\text{mat}(\widetilde{S}(r, :)))$;
6 **end**
7 $\widehat{X} \leftarrow \sum_{r=1}^R \widehat{S}_r \circ \widehat{C}_r$;
8 **return** \widehat{X} , \widehat{S} and \widehat{C} .

successive projection algorithm (SPA) algorithm in [19], [20] due to its efficiency; see [19], [20], [34] and Appendix G for details of the SPA algorithm.

With the learned f_{θ} and the estimated \widehat{P}_r [cf. Eq. (10)] from the NMF stage, we complete the SLFs by letting $\widehat{S}_r \leftarrow f_{\theta}(\widehat{P}_r)$. The two-stage process is summarized in Algorithm 1, which we will refer to as the *nonnegative matrix factorization assisted deep emitter spatial loss completion* (Nasdac). One remark is that although all NMF algorithms admit inevitable scaling and permutation ambiguities (i.e., the Λ and Π matrices in Theorem 1 cannot be removed), these ambiguities do not affect the reconstruction, if $\widehat{C} = C\Pi\Lambda$, $\widehat{H} = \Lambda^{-1}\Pi^\top H$ hold—since the ambiguity matrices cancel each other in the final assembling stage (i.e., line 7 in Algorithm 1).

Note that in Nasdac, the number of emitters R is assumed to be previously estimated or known. Under the considered scenario, estimating R corresponds to estimating the rank of the NMF model $G \approx CH$ [cf. Eq. (29)]. This can be accomplished by existing model-order selection methods (see [38] for an overview). Many methods for rank estimation under noisy NMF models can also be used, e.g., [20], [39]–[41].

IV. PERFORMANCE-ENHANCED APPROACH

Once the network f_{θ} is learned off-line, the Nasdac method offers a computationally simple way to complete the radio map tensors. Nonetheless, the method was proposed under the premise that C satisfies Assumption 1, or roughly speaking, the emitters use the frequency bins sparsely. In this subsection, we take a step forward and address more challenging scenarios where Assumption 1 is violated. We also consider the case where

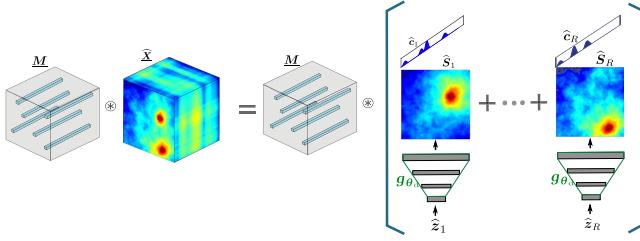


Fig. 7. Illustration of the modelling idea used for the performance enhanced approach (DowJons).

the sensor measurements are noisy, and we will offer stability guarantees.

To proceed, we denote

$$\underline{Y}(i, j, :) = \underline{X}_{\natural}(i, j, :) + \underline{N}(i, j, :), \quad \forall(i, j), \quad (11)$$

where \underline{N} is a noise tensor and \underline{X}_{\natural} is the ground-truth tensor that we aim to recover. With $\underline{Y}(i, j, :)$'s for $(i, j) \in \Omega$ collected by the sensors, our formulation to estimate \underline{X}_{\natural} is as follows:

$$\min_{\{c_r, z_r\}_{r \in [R]}} \left\| \underline{M}_{\text{sens.}} \circledast \left(\underline{Y} - \sum_{r=1}^R g_{\theta_d}(z_r) \circ c_r \right) \right\|_F^2. \quad (12)$$

In the above, the neural network $g_{\theta_d}(z_r)$ is a generative network that links a latent vector $z_r \in \mathbb{R}^D$ with the SLF $S_r \in \mathbb{R}^{I \times J}$ —see (6) and the discussions there. Fig. 7 illustrates the idea of modeling the radio map used in (12).

Denote $\{\hat{c}_r, \hat{z}_r\}_{r=1}^R$ as the solution obtained via solving (12). We reconstruct the radio map tensor by $\widehat{\underline{X}} \leftarrow \sum_{r=1}^R g_{\theta_d}(\hat{z}_r) \circ \hat{c}_r$. The idea is similar to low-rank matrix factorization based data completion, and in particular the learned generator $g_{\theta_d}(z_r)$ is used to represent the SLF of emitter r . This representation can be understood as imposing realistic data-driven constraints to the learned SLFs. Using the above, one does not need to resort to the two-stage approach as in Nasdac, which is prone to error propagation. More importantly, we will show that the criterion in (12) does not need Assumption 1 to guarantee recoverability of \underline{X}_{\natural} , and that the criterion is robust to noise.

A. Recoverability Analysis

In this subsection, we analyze the recoverability of the ground-truth radio map tensor \underline{X}_{\natural} under the criterion in (12). In our analysis, we treat \underline{X}_{\natural} as a deterministic term, while the set of sampling locations specified by Ω are random. To proceed, first, consider the following assumption on the employed neural network in our framework:

Assumption 2 (Deep Generator Structure): Let $\mathcal{Z} = \{z \in \mathbb{R}^D : \|z\|_2 \leq q\}$ and $g_{\theta_d}(\cdot) : \mathcal{Z} \rightarrow \mathbb{R}^{I \times J}$ be a deep generative model with the network structure $g_{\theta_d}(z) = \text{mat}(\zeta_L(\mathbf{A}_L(\dots \zeta_1(\mathbf{A}_1 z)))$, where $\mathbf{A}_\ell \in \mathbb{R}^{D_\ell \times D_{\ell-1}}$ is the network weight in the ℓ th layer, $\ell = 1, \dots, L$, in which $D_0 = D$ and $D_L = IJ$, $\zeta_i(\cdot) = [\zeta(\cdot), \dots, \zeta(\cdot)]^\top : \mathbb{R}^{D_\ell} \rightarrow \mathbb{R}^{D_\ell}$ is a ϕ_i -Lipschitz function, $\zeta(\cdot) : \mathbb{R} \rightarrow \mathbb{R}$ is an entry-wise activation function (e.g., ReLU or Sigmoid) [42] and $P = \prod_{i=1}^L \phi_i \|\mathbf{A}_i\|_2 < \infty$.

Note that many neural network structures (e.g., the fully connected network (FCN) and the convolutional neural network (CNN)) satisfy the structure specified in Assumption 2. Popular activation functions such as Sigmoid and ReLU are also Lipschitz continuous. We will use the following definitions:

Definition 1 (Covering Number [43]): The covering number of a set \mathcal{W} with parameter $\varepsilon > 0$, denoted as $N(\mathcal{W}, \varepsilon)$, is defined as the smallest number of ε -radius norm balls required to completely cover the set \mathcal{W} (with possible overlaps). In other words, $N(\mathcal{W}, \varepsilon) = |\overline{\mathcal{W}}|$, where $\overline{\mathcal{W}}$ is discrete, and for any $w \in \mathcal{W}$, there exists a $\bar{w} \in \overline{\mathcal{W}} \subseteq \mathcal{W}$ such that $\|\bar{w} - w\| \leq \varepsilon$. The discrete set $\overline{\mathcal{W}} \subseteq \mathcal{W}$ is called an ε -net of \mathcal{W} .

In principle, any norm can be used to define an ε -net. We use the Euclidean norm in this work. The covering number is a way of measuring the “complexity” of a continuous set using its discretized version.

Definition 2 (Solution Set): We define $\mathcal{X}_{R, g_{\theta_d}} \subset \mathbb{R}^{I \times J \times K}$ as the set that contains all possible $\widehat{\underline{X}}$ that could be expressed as $\widehat{\underline{X}} = \sum_{r=1}^R \widetilde{S}_r \circ \widetilde{c}_r$ with $\widetilde{S}_r = g_{\theta_d}(z_r)$, $z_r \in \mathcal{Z}$ (where \mathcal{Z} and g_{θ_d} are specified in Assumption 2), $\|g_{\theta_d}(z_r)\|_F \leq \beta$, and $\|\widetilde{c}_r\|_2 \leq \alpha$ for $r = 1, \dots, R$.

Definition 3: We define $\text{Gap}(\widehat{\underline{X}}, \Omega) = \sqrt{\text{Loss}(\widehat{\underline{X}})} - \sqrt{\text{Loss}(\underline{X})}$, where $\widehat{\underline{X}} \in \mathcal{X}_{R, g_{\theta_d}}$ and

$$\begin{aligned} \widehat{\text{Loss}}(\widehat{\underline{X}}) &= \frac{1}{|\Omega|K} \sum_{i,j \in \Omega} \left\| \underline{Y}(i, j, :) - \widehat{\underline{X}}(i, j, :) \right\|_2^2, \\ \text{Loss}(\widehat{\underline{X}}) &= \frac{1}{IJK} \sum_{i,j \in [I] \times [J]} \left\| \underline{Y}(i, j, :) - \widehat{\underline{X}}(i, j, :) \right\|_2^2. \end{aligned}$$

Note that $\widehat{\text{Loss}}$ can be understood as the “empirical loss” measured at a solution $\widehat{\underline{X}} \in \mathcal{X}_{R, g_{\theta_d}}$, and Loss is the “true” loss measured over the complete data (or, the generalization error). Under the above definitions, we first show the following Lemma:

Lemma 1: Define $\text{Gap}^*(\Omega) = \sup_{\widehat{\underline{X}} \in \mathcal{X}_{R, g_{\theta_d}}} |\text{Gap}(\widehat{\underline{X}}, \Omega)|$.

Assume that Assumption 2 holds. For any $\widehat{\underline{X}} \in \mathcal{X}_{R, g_{\theta_d}}$, under the sensing model in (11), it holds with probability of at least $1 - \delta$ that

$$\text{Gap}^*(\Omega) \leq \frac{2cR}{\sqrt{|\Omega|}} + \left(\frac{\xi^2 \omega}{2} \log \left(\frac{2N(\mathcal{X}_{R, g_{\theta_d}}, cR)}{\delta} \right) \right)^{\frac{1}{4}}, \quad (13)$$

where $c > 0$ is a positive scalar, $\omega = (\frac{1}{|\Omega|} - \frac{1}{IJ} + \frac{1}{IJ|\Omega|})$, and $\xi = \frac{1}{K}(\sqrt{K}(v + \nu) + R\alpha\beta)^2$ in which $v = \max_{i,j,k} |\underline{X}_{\natural}(i, j, k)|$, $\nu = \max_{i,j,k} |\underline{N}(i, j, k)|$.

We also show the following:

Lemma 2: For the set in Definition 2, the covering number of its ε -net is

$$N(\mathcal{X}_{R, g_{\theta_d}}, \varepsilon) \leq \left(\frac{3R(\alpha + \beta)}{\varepsilon} \right)^{R(K+D)} \alpha^{RK} (Pq)^{RD}. \quad (14)$$

Using the above lemmas and definitions, we present our main recoverability theorem:

Theorem 2 (Recoverability): Under the observation model in (11), assume that $\underline{\mathbf{X}}^* = \sum_{r=1}^R \mathbf{g}_{\theta_d}(z_r^*) \circ c_r^*$, where z_r^* and c_r^* are from any optimal solution of (12). Also assume that the network structure and z_r^* and c_r^* satisfy the specifications in Assumption 2 and the solution set in Definition 2. Then, the following holds with probability of at least $1 - \delta$:

$$\begin{aligned} & \frac{1}{\sqrt{IJK}} \|\underline{\mathbf{X}}^* - \underline{\mathbf{X}}_\natural\|_F \leq \frac{1}{\sqrt{IJK}} \|\underline{\mathbf{N}}\|_F + \text{Gap}^*(\Omega) \\ & + \frac{1}{\sqrt{|\Omega|K}} \left(\|\underline{\mathbf{M}}_{\text{sens.}} \circledast \underline{\mathbf{N}}\|_F + \text{Err}_{\text{rep}}(\mathcal{X}_{R,\theta_d}) \right), \end{aligned}$$

where $\text{Err}_{\text{rep}}(\mathcal{X}_{R,\theta_d}) = \|\widetilde{\underline{\mathbf{X}}}^* - \underline{\mathbf{X}}_\natural\|_F$, in which $\widetilde{\underline{\mathbf{X}}}^* = \arg \min_{\widetilde{\underline{\mathbf{X}}} \in \mathcal{X}_{R,\theta_d}} \|\widetilde{\underline{\mathbf{X}}} - \underline{\mathbf{X}}_\natural\|_F^2$, and $\text{Gap}^*(\Omega)$ is upper bounded by (13) with the covering number in (14).

Theorem 2 asserts that the criterion in (12) admits recoverability of the ground-truth radio map $\underline{\mathbf{X}}_\natural$ —even if there exists sensing noise and the learned generative model is not perfect. The recovery error consists of three main sources, namely, the sensing noise $\underline{\mathbf{N}}$, sampling-induced noise (or, generalization error) $\text{Gap}^*(\Omega)$ and the generative model’s representation error, i.e., $\text{Err}_{\text{rep}}(\mathcal{X}_{R,\theta_d})$. If a more complex (i.e., deeper and/or wider) neural network is used—which means that the generative model can approximate more complex SLFs—then the representation error can be reduced. However, using a more complex neural network will make the covering number of the solution set increase, and thus $\text{Gap}^*(\Omega)$ increases—which presents a trade-off between network expressiveness and recovery accuracy. In addition, when $|\Omega|$ approaches IJ and I, J grow large, the sampling-induced error $\text{Gap}^*(\Omega)$ approaches zero [cf. Eq. (13)].

A remark is that although the theorem statement assumes a known R , it can also be used to understand the cases where R is not exactly known (or wrongly estimated). When $\widehat{R} = R'$ is underestimated, the energy of the missed components $S_r \circ c_r$ for $R' < r \leq R$ can be absorbed into the noise term $\underline{\mathbf{N}}$. When \widehat{R} is overestimated, $\underline{\mathbf{X}}_\natural$ can still be expressed by using (1), with some redundant/spurious components. The redundant components do not create any new noise terms, but will increase the covering number of the model. This means that more samples will be required for attaining the same bound of recoverability accuracy—which is the price to pay for overestimating R .

B. Algorithm Design

We propose an alternating optimization algorithm to handle the formulated problem in (12); i.e., we tackle subproblems w.r.t. \mathbf{C} and $\mathbf{Z} = [z_1, \dots, z_R]$ alternately, until a certain convergence criterion is met.

Note that the \mathbf{C} -subproblem can be re-expressed as follows:

$$\mathbf{C}^{(k+1)} \leftarrow \arg \min_{\mathbf{C} \geq 0} \|\mathbf{G} - \mathbf{CH}^{(k)}\|_F^2, \quad (15)$$

where \mathbf{G} is defined as before, $\mathbf{H}^{(k)}(r, :) = \mathbf{S}^{(k)}(r, \Omega_{\text{col}})$, and $\mathbf{S}^{(k)}(r, :) = \text{vec}(\mathbf{g}_{\theta_d}(z_r))^\top$ is the (vectorized) current estimate for the r th SLF at iteration k . The problem in (15) is a *nonnegativity-constrained least squares* (NNLS) problem—which admits a plethora of off-the-shelf solvers. In this work,

we employ the NNLS algorithm proposed in [44] that strikes a good balance between speed and accuracy.

As for the z_r -subproblem, since it involves deep neural networks, any back-propagation-based gradient descent-related algorithms can be used [42]. To be specific, we use the update $\mathbf{Z}^{(k_{t+1})} \leftarrow \mathbf{Z}^{(k_t)} - \gamma^t \bar{\nabla}_{\mathbf{Z}} f(\mathbf{Z}^{(k_t)}; \mathbf{C}^{(k+1)})$, and we let $\mathbf{Z}^{(k_0)} \leftarrow \mathbf{Z}^{(k)}$, $\mathbf{Z}^{(k+1)} \leftarrow \mathbf{Z}^{(k_T)}$, where T is the number of gradient iterations (inner iterations) used for handling the \mathbf{Z} -subproblem in the k th outer iteration (that alternates between \mathbf{Z} and \mathbf{C}), γ^t is the step size in the t th inner iteration, and $\mathbf{Z} = [z_1, \dots, z_R]$. We have used the notation

$$f(\mathbf{Z}; \mathbf{C}^{(k+1)}) = \left\| \underline{\mathbf{M}}_{\text{sens.}} \circledast \left(\underline{\mathbf{Y}} - \sum_{r=1}^R \mathbf{g}_{\theta_d}(z_r) \circ c_r^{(k+1)} \right) \right\|_F^2,$$

and $\bar{\nabla}_{\mathbf{Z}} f$ is a gradient-based direction, e.g., momentum-based gradient or scaled gradient (such as those used in Adagrad [45] and Adam [30]), which can often be constructed from the gradient $\nabla_{\mathbf{Z}} f$. The gradient of f w.r.t. \mathbf{Z} can be computed via the chain rule and back-propagation. In this work, we employ the popularized gradient evaluation tool, namely, autograd in Pytorch [45], to carry out the computation. The step size γ^t and $\bar{\nabla}_{\mathbf{Z}} f$ follow the Adam strategy [30], and γ^t has an initial value of 0.01. The algorithm is summarized in Algorithm 2, which is referred to as the *Deep generative PriOr With Joint OptimizatioN for Spectrum cartography* (DowJons) algorithm.

The DowJons algorithm is an alternating optimization (AO) algorithm where the \mathbf{Z} block is nonconvex. This means that one often does not optimally solve the \mathbf{Z} -subproblem in practice. Nonetheless, we observe that DowJons converges well even if the \mathbf{Z} -subproblem is executed with only a few iterations. Indeed, under some conditions, one can show that such an “inexact” AO procedure produces a solution sequence that converges to the vicinity of a stationary point of (12). Discussions on DowJons’s convergence properties can be found in Appendix C.

Remark 2: A remark on the difference between Nasdac and DowJons is as follows. Nasdac is a two-stage approach. The two stages are (i) SPA-based SLF disaggregation and (ii) learned network-based SLF completion, respectively. Both stages are lightweight to compute, but there may be error propagation from stage (i) to stage (ii). Either stage alone could not guarantee recoverability of $\underline{\mathbf{X}}_\natural$, no matter how well the associated optimization problem (e.g., NMF) is solved. In contrast, for DowJons, if the optimization criterion in (12) is solved, $\underline{\mathbf{X}}_\natural$ is ensured to be recovered with a reasonable bound on the estimation error. Hence, DowJons can be regarded as a one-stage method in the sense of ensuring recoverability guarantees. Nonetheless, the recoverability advantages of DowJons come at the expense of a higher computational cost.

V. EXPERIMENTS

In this section, we use synthetic-data and real-data experiments to showcase the effectiveness of the proposed methods.

Algorithm 2: DowJons.

Data: $\underline{\mathbf{Y}}, \underline{\mathbf{M}}_{\text{sens.}}, R, \gamma, \tau, \text{MaxIter}, \theta$
Result: Estimated $\widehat{\mathbf{X}}$, $\widehat{\mathbf{S}}$ and $\widehat{\mathbf{C}}$

// Initialization via Nasdac:

- 1 $\overline{\mathbf{G}} = \text{unfold}(\underline{\mathbf{M}}_{\text{sens.}} \circledast \underline{\mathbf{Y}});$
- 2 $\mathbf{G} = \overline{\mathbf{G}}(:, \Omega_{\text{col}});$
- 3 $\mathbf{C}^{(0)}, \mathbf{S}^{(0)} \leftarrow \text{Nasdac}(\underline{\mathbf{M}}_{\text{sens.}} \circledast \underline{\mathbf{Y}}, \Omega, R, f_\theta(\cdot));$
- 4 **for** $r = 1 : R$ **do**
- 5 | $\mathbf{z}_r^{(0)} \leftarrow p_{\theta_e}(\mathbf{S}_r^{(0)});$
- 6 **end**
- 7 $k \leftarrow 0;$
- 8 **while** $(f(\mathbf{Z}^{(k+1)}; \mathbf{C}^{(k+1)}) \leq f(\mathbf{Z}^{(k)}; \mathbf{C}^{(k)}) - \tau)$ or
 $(k \leq \text{MaxIter})$ **do**
- 9 | $\mathbf{C}^{(k+1)} \leftarrow \text{NNLS}(\mathbf{G}, \mathbf{S}^{(k)}(:, \Omega_{\text{col}}));$
- 10 | $\mathbf{Z}^{(k_0)} \leftarrow \mathbf{Z}^{(k)};$
- 11 | **for** $t = 1 : T$ **do**
- 12 | $\mathbf{Z}^{(k_t)} \leftarrow \mathbf{Z}^{(k_{t-1})} - \gamma^t \nabla_{\mathbf{Z}} f(\mathbf{Z}^{(k_{t-1})}; \mathbf{C}^{(k+1)});$
- 13 | **end**
- 14 | $\mathbf{Z}^{(k+1)} \leftarrow \mathbf{Z}^{(k_T)};$
- 15 | **for** $r = 1 : R$ **do**
- 16 | $\mathbf{S}_r^{(k+1)} = g_{\theta_d}(\mathbf{z}_r^{(k+1)});$
- 17 | **end**
- 18 | $k \leftarrow k + 1;$
- 19 **end**
- 20 $\widehat{\mathbf{X}} = \sum_{r=1}^R \widehat{\mathbf{S}}_r^{(k)} \circ \widehat{\mathbf{c}}_r^{(k)}.$

A. Synthetic-Data Experiments - Settings

1) *Baselines:* We use a number of baselines to benchmark the proposed methods. To be specific, we compare the performance of our methods with that of the TPS method in [46], the LL1 method in [16], and the DeepComp method in [18]. The TPS method uses 2D splines to interpolate between the observed entries, the LL1 method is a block-term tensor based completion method, and the DeepComp method also uses deep generative models, but directly learns the aggregated radio map using a neural network; see our discussion in previous sections. We hope to remark that the baseline LL1 is implemented as a combination of emitter level disaggregation and classic interpolation (using TPS) based individual SLF completion. Therefore, it is similar to the proposed approach, but the deep network completion used in our framework is replaced by a classic SLF interpolation method. The comparison with LL1 can therefore serve as an ablation study of the effectiveness of the learned deep priors in our approach. More comparisons with such “SLF disaggregation and classic interpolation” approaches using different interpolation methods can be found in the supplementary materials.

2) *Data Generation:* Our simulations and training set are generated following the joint path loss and log-normal shadowing model [29]. Under this model, the SLF of emitter r is expressed as follows:

$$\mathbf{S}_r(\mathbf{y}) = \|\mathbf{y} - \mathbf{r}_r\|_2^{-\gamma_r} 10^{v_r(\mathbf{y})/10}, \quad (16)$$

where $\mathbf{y} = (i, j)$ denotes the spatial coordinates, \mathbf{r}_r is the location of r -th emitter, γ_r is the path loss coefficient associated with emitter r , and $v_r(\mathbf{y})$ is the correlated log-normal shadowing component. This component is sampled from zero-mean Gaussian distribution with variance η_r , and the auto-correlation between \mathbf{y} and \mathbf{y}' can be written as $\mathbb{E}(v_r(\mathbf{y}), v_r(\mathbf{y}')) = \eta_r \exp(-\|\mathbf{y} - \mathbf{y}'\|_2/d_{\text{corr}})$, in which d_{corr} is the so-called *decorrelation distance*. Smaller d_{corr} and larger

η_r indicate more severe shadowing effects and thus harsher environments. For example, for an outdoor environment, the typical values of d_{corr} and η_r range from 50 to 100 m and 4 dB to 12 dB, respectively [29].

We consider a geographical region of 50×50 m². We discretize the region into 50×50 grids, and consider K frequency bins of interest, where $K = 64$ unless specified otherwise. The PSDs of the emitters are assumed to be a combination of randomly scaled sinc functions. Specifically, the PSD of emitter r is given by $c_r(k) = \sum_{i=1}^M a_i^{(r)} \text{sinc}^2(k - f_i^{(r)}/w_i^{(r)})$, where $f_1^{(r)}, \dots, f_M^{(r)}$ are the central frequencies of M subbands that are available to emitter r , where M will be specified in different simulations; $w_i^{(r)}$ controls the width of the emitter r 's PSD's sidelobe at its i th subband, which is sampled from a uniform distribution between 2 and 4; $a_i^{(r)}$ is a random scaling of emitter r 's PSD intensity at its i th subband, which follows the uniform distribution between 0.5 and 2.5. After generating \mathbf{S}_r and c_r , $r = \{1, \dots, R\}$, we use (9) to obtain the aggregate radio map $\underline{\mathbf{X}}$. When noise is considered, we use $\underline{\mathbf{X}} = \sum_{r=1}^R \mathbf{S}_r \circ c_r + \underline{\mathbf{N}}$, where $\underline{\mathbf{N}}$ represents the noise tensor.

3) *Deep Network Configuration and Training Sample Generation:* As mentioned, we employ a deep autoencoder for learning the generative model of the SLFs. The autoencoder has a convolutional encoder-decoder architecture with the latent dimension being $D = 256$. More details can be seen in Appendix A.

To train our network, we generate 500,000 single emitter SLFs within the region of interest. The SLFs follow the model in (16). For each sample, we randomly pick a position for the emitter from the \mathbf{r}_r over the 50×50 grids. The parameters γ_r for all r follows uniform distribution from 2 to 2.5, which covers the range of path loss coefficients in scenarios of interest [47]; $\eta_r = \eta$ for all r and it is sampled uniformly from 3 to 8; and d_{corr} is sampled uniformly from 30 to 100. The selection of the latter two parameters is used to cover scenarios with moderate to severe shadowing effects.

To generate training samples for DeepComp, we use R such SLFs and randomly generated PSDs c_r 's to construct $\underline{\mathbf{X}}$ following (1). The c_r 's are generated using $M = 10$, so that most of the slabs contain at least one emitter, and other parameters set as described earlier. We collect 500,000 $\underline{\mathbf{X}}(:, :, k)$'s as the training set with R sampled randomly from $\{3, 4, 5, 6\}$ for each $\underline{\mathbf{X}}$.

4) *Performance Metric:* For an estimated $\widehat{\mathbf{X}}$ of the ground-truth $\underline{\mathbf{X}}$, we measure *squared reconstruction error* (SRE), i.e.,

$$\text{SRE} = \|\widehat{\mathbf{X}} - \underline{\mathbf{X}}\|_{\text{F}}^2 / \|\underline{\mathbf{X}}\|_{\text{F}}^2.$$

In order to evaluate the accuracy of $\widehat{\mathbf{C}}$, we use *normalized absolute error* (NAE) to evaluate the estimates of \mathbf{C} defined as follows:

$$\text{NAE}_{\mathbf{C}} = \frac{1}{R} \sum_{r=1}^R \left\| \frac{\mathbf{c}_r}{\|\mathbf{c}_r\|_1} - \frac{\widehat{\mathbf{c}}_r}{\|\widehat{\mathbf{c}}_r\|_1} \right\|_1, \quad (17)$$

where we have assumed that the column permutation between \mathbf{C} and $\widehat{\mathbf{C}}$ has been removed, e.g., using the Hungarian algorithm

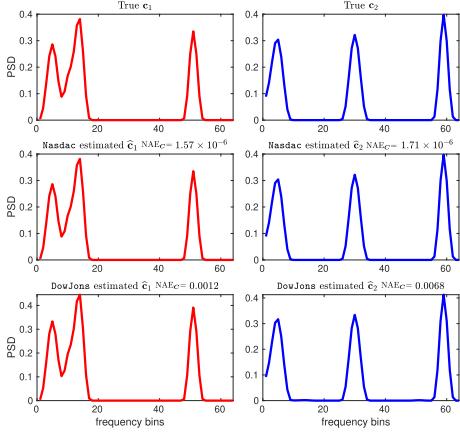


Fig. 8. Ground-truth and estimated PSDs by Nasdac and DowJons; $\rho = 10\%$, $R = 2$, $\eta = 5$, $d_{\text{corr}} = 50$.

[16], Sec. VI. For the SLFs, NAE_S for S is defined in the same way; see [16].

5) *Hyperparameters of Algorithms*: Among the algorithms under test, DowJons and LL1 have some hyperparameters to be determined in advance. For DowJons, we stop the algorithm either when the relative change of the cost function is smaller than 0.003 or when the alternating optimization procedure reaches 10 outer iterations. For the inner iterations that tackle the Z -subproblem, we set the maximum number of iterations T to be 10. For tensor-based method LL1, we set the rank of SLFs to be up to 4 using trial-and-error type tuning. The regularization parameters for all the mode latent factors to be 10^{-3} . The algorithm is terminated when the relative change of its loss function less than 10^{-3} or when the number of iterations exceeds 50. All the results are averaged from 30 Monte Carlo trials.

B. Synthetic-Data Experiments - Results

1) *Sparse Spectral Occupancy*: Under the observation model in (7), we first test the algorithms under scenarios where the emitters use the spectral bands in a sparse manner such that Assumption 1 holds. To ensure this, we first designate one frequency band for each emitter and allow to use up to $M = 16$ frequency bands out of the total remaining bands. We use this setting for all the tables and figures unless stated otherwise.

Figs. 8–9 show an illustrative example, where $R = 2$ emitters are in the region of interest. In this case, we randomly sample $\rho = |\Omega|/IJ \times 100\% = 10\%$ of the 2,500 grids; i.e., 10% of the $\underline{\mathbf{X}}(i, j, :)$'s are revealed to us by sensors that capturing the PSD over the $K = 64$ frequency bands at (i, j) . We set $\eta_r = \eta = 5$ for all r and $d_{\text{corr}} = 50$. The SREs/ NAEs of the estimated radio maps/SLFs are also displayed in the figures, so that the reader can have a better understanding of the correspondence between the performance metrics and the visual quality of the recovered maps/SLFs. One can see that Nasdac and DowJons recover the PSDs and SLFs accurately in this case—and thus the aggregated radio map as well.

Fig. 10 shows the results of a much more challenging case, where $R = 10$ emitters are present and only $\rho = 5\%$ locations

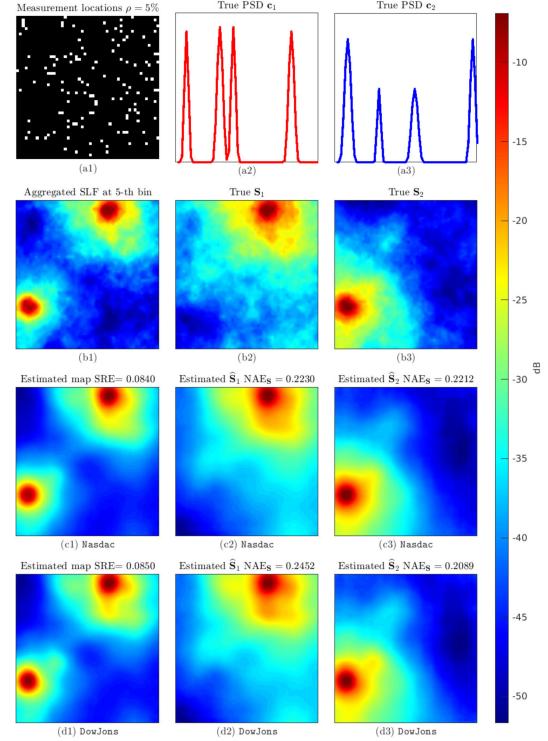


Fig. 9. Ground truth SLF ((b2), b(3)) and radio map (b1), and reconstructed SLF ((c2), (c3), (d2), (d3)) and radio map ((c1), (d1)) by Nasdac and DowJons under sparse spectral occupancy; $\rho = 5\%$, $R = 2$, $\eta = 5$, and $d_{\text{corr}} = 50$; the visualized maps are at the 5th frequency bin.

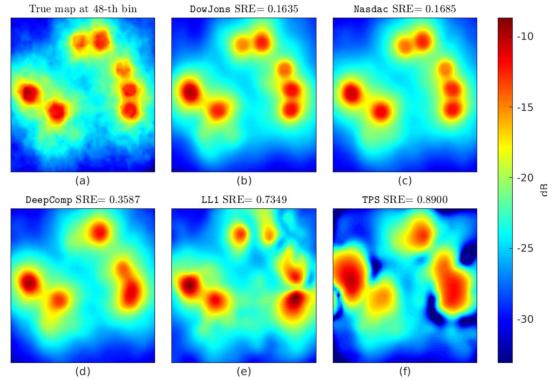


Fig. 10. Ground-truth and reconstructed radio map by proposed algorithms and baselines under $R = 10$, $\eta = 5$, $\rho = 5\%$, $d_{\text{corr}} = 50$, $M = 4$ when Assumption 1 holds (see Fig. 9 (a2) and (a3) for the PSDs when Assumption 1 holds.).

are sampled. The other settings are the same as before. Fig. 10 shows the ground truth and recovered radio maps by the algorithms at the 48th frequency band. One can see that both the proposed methods, i.e., DowJons and Nasdac, yield visually accurate maps as in the previous case. The DeepComp approach seems to miss some emitters, especially when the emitters are close to each other. This may be because $R = 10$ emitters are present in this frequency, while DeepComp was trained using up to 6 emitters. Purely model-based methods (i.e., LL1 and TPS) do not perform as well, possibly because the model assumptions

TABLE II
TEST TIME OF ALGORITHMS FOR RUNNING THE TASK IN FIG. 10

	TPS	LL1	DeepComp	Nasdac	DowJons
Runtime (sec.)	0.203	6.8	0.112	0.069	5.2

TABLE III
SRE UNDER VARIOUS ρ 'S; $R = 7$, $d_{\text{corr}} = 50$, $\eta = 6$

ρ	TPS	LL1	DeepComp	Nasdac	DowJons
1%	0.6882	0.8158	0.5257	0.4195	0.4078
5%	0.2974	0.3387	0.2047	0.1319	0.1332
10%	0.1493	0.1347	0.1162	0.0918	0.0896
15%	0.1079	0.1085	0.0929	0.0798	0.0713
20%	0.0840	0.0710	0.0898	0.1013	0.0599

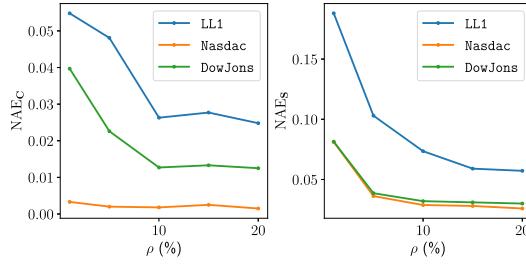


Fig. 11. NAE under various ρ 's; $R = 7$, $d_{\text{corr}} = 50$, $\eta = 6$.

(e.g., the SLFs being low-rank matrices in LL1) are grossly violated.

A remark is that although both Nasdac and DowJons work similarly in terms of recovery accuracy in Fig. 10, there is a notable difference in the runtime performance. Table II shows the runtime in the test stage of the algorithms. Best results are highlighted in bold. One can see that Nasdac takes only 0.069 seconds to accomplish the task. While DowJons takes 5.2 seconds. This is because there is no heavy optimization in the test stage of Nasdac, while DowJons involves solving a complex optimization problem in the test stage. As a tradeoff, DowJons can work with more challenging cases where Assumption 1 does not hold—which will be seen in Section V-B2.

Table III shows the performance of the algorithm under various ρ 's. One can see that all algorithms favor more samples (sensing locations). The DL-based methods (Nasdac, DowJons and DeepComp) exhibit notable advantages over the model-based methods LL1 and TPS, especially when $\rho \leq 5\%$. In addition, the proposed Nasdac and DowJons consistently outperform the baselines under most ρ 's.

Fig. 11 shows the estimation accuracy of the individual PSDs and SLFs by pertinent algorithms under the same settings. Note that only LL1, Nasdac and DowJons estimate such individual components, while other algorithms directly estimate the aggregated radio map. Interestingly, the PSDs are estimated accurately by all the algorithms under test. However, LL1 performs poorly in terms of SLF estimation. This backs our motivation for this work—the low-rank matrix model that LL1 leverages for estimating S_r does not always hold. On the other hand, Nasdac and DowJons are both able to estimate the SLFs more accurately. In particular, the proposed methods-output NAE_S's are at least improved by 50% relative to the outputs of LL1 when $\rho \leq 5\%$.

We also note that Nasdac exhibits higher estimation accuracy of the PSDs and SLFs relative to DowJons, even though the

TABLE IV
SRE UNDER VARIOUS η 'S; $R = 7$, $d_{\text{corr}} = 50$, AND $\rho = 10\%$

η	TPS	LL1	DeepComp	Nasdac	DowJons
4	0.1415	0.1259	0.0852	0.0861	0.0556
5	0.1387	0.1245	0.0934	0.0979	0.0684
6	0.1507	0.1444	0.1184	0.1141	0.0841
7	0.1679	0.1581	0.1478	0.1536	0.1225
8	0.1850	0.1798	0.1644	0.1539	0.1362

TABLE V
SRE UNDER VARIOUS d_{corr} ; $R = 7$, $\eta = 6$ AND $\rho = 10\%$

d_{corr}	TPS	LL1	DeepComp	Nasdac	DowJons
30	0.1803	0.1948	0.1586	0.1317	0.1279
50	0.1662	0.1741	0.1327	0.1058	0.0912
70	0.1397	0.1437	0.1057	0.0867	0.0735
90	0.1510	0.1463	0.1023	0.0706	0.0611

TABLE VI
SRE UNDER VARIOUS R 'S; $\eta = 6$, $d_{\text{corr}} = 50$, $\rho = 5\%$

R	TPS	LL1	DeepComp	Nasdac	DowJons
6	0.2355	0.2146	0.1426	0.0950	0.0813
7	0.2353	0.2236	0.1616	0.1095	0.0882
8	0.2237	0.2213	0.1715	0.0845	0.0872
9	0.2248	0.2459	0.1809	0.0879	0.0830
10	0.2368	0.2500	0.1899	0.1029	0.0986
11	0.2498	0.2528	0.1922	0.1081	0.0901

overall radio map reconstruction by the latter is more accurate. This may be because the optimization criterion of DowJons targets minimizing the error of the aggregated radio map, instead of the PSDs and SLFs. Also note the NAE definitions do not consider amplitudes of the quantities [c_r and $S(r, :)$] under test [cf. Eq. (17)], but amplitudes are considered in the SRE measure when evaluating the reconstruction error. Hence, low NAEs do not directly imply good SREs, although they are related and are both meaningful metrics.

Tables IV–V show the SREs under various levels of shadowing effects by changing η and d_{corr} , respectively. Note that $\eta = 4$ and $d_{\text{corr}} = 90$) correspond to relatively mild shadowing, while $\eta = 8$ and $d_{\text{corr}} = 30$ both correspond to severe shadowing. From the two tables, one can see that Nasdac and DowJons output competitive SREs relative to the baselines. We also note that Nasdac constantly produces reasonable results, but is less accurate compared DowJons in the most challenging cases, i.e., when $\eta = 8$ or $d_{\text{corr}} = 30$. This may be because Nasdac uses a two-stage approach. Challenging environments may make the first state (i.e., NMF-based disaggregation) less accurate—and the error propagates to the SLF completion stage. Nonetheless, using a one-shot learning criterion, DowJons does not have this issue. The price to pay is that DowJons's optimization process is more costly relative to Nasdac.

Table VI shows the SREs of the algorithms when R varies. An important observation is that the performance of DeepComp deteriorates when R increases. The reason is that DeepComp is trained using samples that contain up to 6 emitters (which already makes the underlying state space overwhelmingly large, as we discussed; see, e.g., Remark 1). Whenever a band contains more than 6 emitters, the DeepComp approach tends to miss the extra emitters. Nevertheless, the proposed methods that model the emitters' SLFs individually do not have this problem—which

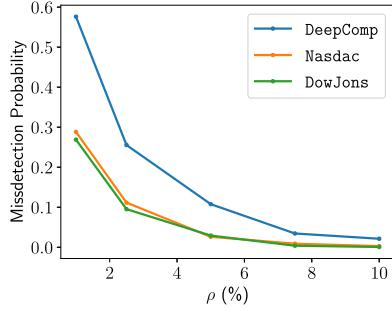


Fig. 12. Emitter misdetection probability under various ρ 's; $R = 10$, $\eta = 5$, $d_{\text{corr}} = 50$.

TABLE VII
SRE UNDER VARIOUS SNRs; $R = 7$, $\eta = 6$, $d_{\text{corr}} = 50$, AND $\rho = 10\%$

SNR(dB)	TPS	LL1	DeepComp	Nasdac	DowJons
0	0.9019	0.8092	0.7745	0.9121	0.6774
10	0.1937	0.2101	0.1676	0.2521	0.1413
20	0.1540	0.1539	0.0947	0.1371	0.0749
30	0.1398	0.1238	0.0988	0.0853	0.0747
40	0.1181	0.0999	0.0807	0.0716	0.0541

supports our design goal of improving the generalization performance.

The results in Table VI also articulate one noticeable challenge of DeepComp: It usually misses some emitters if more emitters than the number of emitters used in training samples appear in scene (also see Fig. 10). We define a misdetection case as follows: If the power of the reconstructed signal at the emitter location in a frequency drops below a certain threshold (25% of the ground-truth signal strength in our experiments), we count this as a misdetection case. The misdetection probabilities of different DL-based methods under $R = 10$, $\rho = 5\%$, $\eta = 5$ and $d_{\text{corr}} = 50$ are shown in Fig. 12. One can see that DeepComp has a misdetection probability that is often much higher than that of the proposed methods.

Table VII shows the performance of all methods under different sensing noise levels under the noisy observation model in (11). In this simulation, all entries of the noise ther \underline{N} [cf. Eq. (11)] are sampled from a uniform distribution between 0 and 1, and then scaled to satisfy the pre-specified signal-to-noise ratios (SNRs). The SNR is defined as $\text{SNR} = 10 \log_{10}(\|\underline{X}_b\|_F^2 / \|\underline{N}\|_F^2)$ (dB). One can see that Nasdac enjoys a high reconstruction accuracy when $\text{SNR} \geq 30$ dB and outperforms all the baselines. However, it is less robust to heavily noise environments—which is, again, a consequence of using a two-stage approach. On the other hand, DowJons offers stable results across all SNRs under test, which also corroborates our noise robustness proof in Theorem 2.

2) *Dense Spectral Occupancy*: We also test the algorithms under the cases where the spectral bands are crowded—i.e., where Assumption 1 does not hold. Under such circumstances, Nasdac is expected to not work, while DowJons should not be affected.

Fig. 13 shows the estimated PSDs under $\rho = 10\%$, $R = 2$, $\eta = 5$, $d_{\text{corr}} = 50$. In this case, the two emitters both use all the frequency bands and thus Assumption 1 is clearly violated.

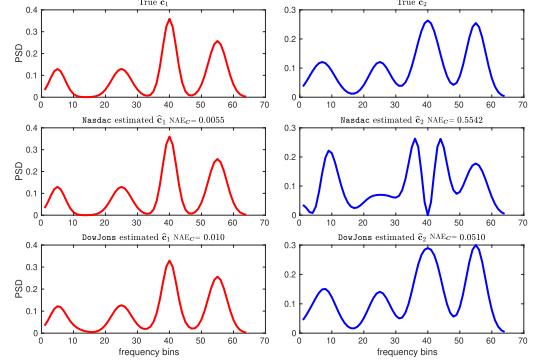


Fig. 13. Visualization of PSD reconstructed by Nasdac and DowJons from 10% sampling when C is dense (does not satisfy Assumption 1); $R = 2$, $\eta = 5$, $d_{\text{corr}} = 50$.

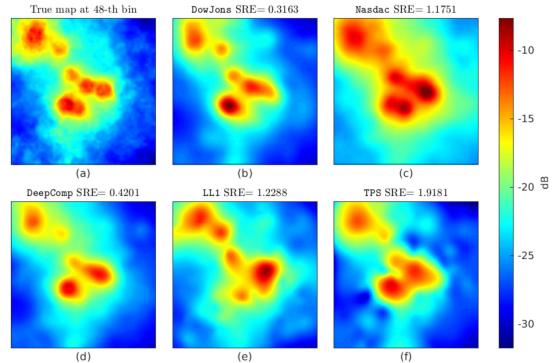


Fig. 14. Ground-truth and reconstructed radio maps under $R = 7$, $\eta = 5$, $\rho = 5\%$, $M = 8$, and $d_{\text{corr}} = 50$, when Assumption 1 does not hold.

As expected, Nasdac fails to estimate the PSDs. Nonetheless, DowJons works reasonably well in terms of reconstructing the PSDs. Fig. 14 shows the reconstructed radio maps in a case where $R = 7$, $\eta = 5$, $\rho = 5\%$, and $d_{\text{corr}} = 50$, where Assumption 1 is again violated. Similar as the previous case, DowJons is essentially not affected by the violation of Assumption 1, but Nasdac could not output reasonable results.

3) *Under Mis-Specified R* : Our algorithms use the knowledge of R to separately model the individual SLFs of the emitters. Although model order selection is a mature technique in signal processing [31], it is also of interest to observe how the algorithms perform under wrongly estimated R .

Fig. 15 shows the performance of DL-methods when R is under or over-estimated. Specifically, we test the algorithms under the case where $R = 7$, but various wrongly estimated \hat{R} 's ranging from 5 to 9 are used for the proposed algorithms. The result of DeepComp is also presented as a baseline, which does not use R and thus is a constant over all the cases. One can see that if $\hat{R} < R$, then the SREs of the proposed algorithms tend to be high. Nonetheless, over-estimated \hat{R} 's (i.e., $\hat{R} \geq R$) seem to essentially not affect the SRE. This phenomenon is understandable: under-estimated \hat{R} 's lead to significant missing components, but over-estimated \hat{R} 's may only introduce repeated or redundant components that may not affect the reconstruction.

4) *Training-Testing Mismatch*: It is natural to ask what happens if the shadowing environment in reality turned out to be very

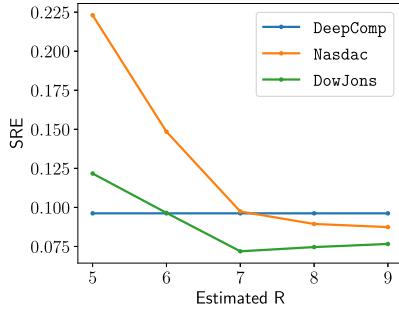


Fig. 15. SRE under various wrongly estimated \hat{R} 's; true $R = 7$, $d_{\text{corr}} = 50$, $\eta = 5$, and $\rho = 10\%$.

TABLE VIII
SRE UNDER VARIOUS η FOR TRAINING-TESTING DATA MISMATCH CASE.
 $R = 5$, $d_{\text{corr}} = 50$, $\rho = 10\%$

η	LL1	WRP models			NRP models		
		DeepComp	Nasdac	DowJons	DeepComp	Nasdac	DowJons
6	0.1521	0.1259	0.1139	0.0954	0.1610	0.1416	0.1005
7	0.1533	0.1746	0.1247	0.1412	0.2007	0.1604	0.1337
8	0.2279	0.1861	0.1602	0.1549	0.2171	0.2022	0.1499

TABLE IX
SRE UNDER VARIOUS d_{corr} FOR TRAINING-TESTING DATA MISMATCH CASE.
 $R = 5$, $\eta = 6$, $\rho = 10\%$

d_{corr}	LL1	WRP models			NRP models		
		DeepComp	Nasdac	DowJons	DeepComp	Nasdac	DowJons
30	0.1471	0.1799	0.1408	0.1316	0.2101	0.1847	0.1382
70	0.1271	0.1227	0.1290	0.0828	0.1472	0.1629	0.0803
90	0.1137	0.1060	0.1217	0.0830	0.1345	0.1460	0.0746

different from the ones that were using to train our neural models. To simulate this condition, we provide experiments where the testing scenarios are not covered in the training set of the neural models. Specifically, we train a set of neural models on simulated data generated from a limited range of shadowing parameters and test it on data generated from a wider range of shadowing parameters.

First, we use a narrow range of shadowing parameters, i.e., $\eta \sim \text{Uniform}([3, 5])$ and $d_{\text{corr}} \sim \text{Uniform}([50, 60])$ to generate training data. The learned models are categorized as “narrow range parameters-learned (NRP) models”. Second, we use a wider range of shadowing parameters, i.e., $\eta \sim \text{Uniform}([3, 8])$ and $d_{\text{corr}} \sim \text{Uniform}([30, 100])$, to generate data to train another set of neural models. We categorize this set of models as “wide range parameters-learned (WRP) models”. We test the “out-of-range” cases (i.e., the parameters used for generating test data are out of the range of the parameters used in the training data) for both η and d_{corr} under the NRP models. The results under the WRP models are used as benchmarks. The results are averaged over 30 trials.

Table VIII shows the SREs attained under various η with $\rho = 10\%$, $R = 5$, $d_{\text{corr}} = 50$, and $M = 20$. One can see that the NRP models admit a mismatch with the test data. One can see that DeepComp and Nasdac seem to have performance degradation when using NRP models. But DowJons is much more resilient to the training-testing model mismatch. This is perhaps because DowJons is an all-at-once optimization-based estimator, which is often more robust to modeling errors.

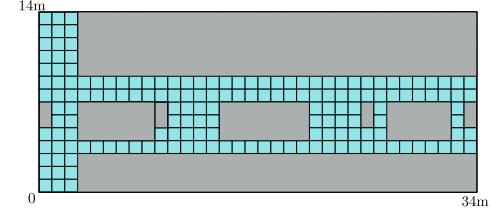


Fig. 16. Geometry of the $14 \times 34\text{m}^2$ indoor region according to [21]. The region is divided into grids that have sizes of $1 \times 1\text{m}^2$. Measurements were taken at the center of each blue shaded cell. The dark shaded regions contain rooms and pillars.

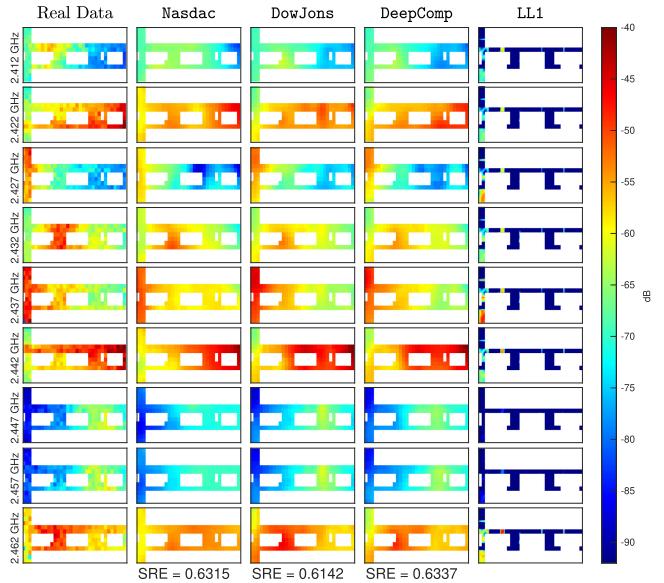


Fig. 17. First Column: 166 ground-truth measurements covering the hallway for all frequency bands [21]. The second, third, fourth, and fifth columns: reconstruction by Nasdac, DowJons, DeepComp, and LL1, respectively, across the hallway.

Table IX shows SREs attained under various d_{corr} with $\rho = 10\%$, $R = 5$, $\eta = 6$, $M = 20$. Similar to the previous case, DeepComp and Nasdac suffer performance degradation but DowJons is robust to the model mismatch.

C. Real Data Experiment

In this subsection, we test the algorithms using real data.

1) *Data Description:* The data was collected in Mannheim University in an office floor across 9 frequency bands [21]. The frequency bins are centered at 2.412 GHz, 2.422 GHz, 2.427 GHz, 2.432 GHz, 2.437 GHz, 2.442 GHz, 2.447 GHz, 2.457 GHz, and 2.462 GHz, respectively. The indoor region is a $14 \times 34\text{ m}^2$ area. The area is divided into $1 \times 1\text{ m}^2$ grids. Sensors are placed at the center of 166 of those grid cells, denoted by blue shaded cells in Fig. 16, which roughly cover the hallway. The dataset constrains more than 400 observations for each (frequency, location) pair. We take the median as the observation for each (frequency, location) pair and obtain an incomplete radio map. All 9 slabs of the radio map are shown in left column of Fig 17, each corresponding to one frequency (with the top being 2.412 GHz and the bottom being 2.462 GHz).

TABLE X
SRE ATTAINED BY DL-BASED METHODS ON REAL DATA FOR VARIOUS ρ

ρ	DeepComp	Nasdac	DowJons
5%	0.5459	0.5326	0.5080
10%	0.3985	0.4035	0.3876
15%	0.3513	0.3605	0.3452
20%	0.3046	0.3139	0.2936

2) *Training Sample Generation*: Since the region of interest is a narrow hallway in an indoor environment, we generate training samples with large η 's and small d_{corr} 's. Specifically, every training sample uses $\eta \in [8, 12]$ and $d_{\text{corr}} \in [5, 50]$, which are both sampled uniformly at random. The training samples cover a $14 \times 34 \text{ m}^2$ region that is the same as floor size. Note that emitters located outside the 14×34 region could also affect the measurements. In order to account for this, we let the emitters be located anywhere in $200 \times 200 \text{ m}^2$ region that contains the $14 \times 34 \text{ m}^2$ region at its center. We extract the SLF induced by the emitter in the $14 \times 34 \text{ m}^2$ region as our training example. The number of training examples and ρ are set to be the same as in the synthetic data experiments (see section V-A). For DeepComp, each emitter use up to 4 frequency bands, which empirically ensures the existence of at least one emitter in each frequency slab. The number of emitters in the training examples are set to be the same as in the synthetic data experiments.

3) *Result*: Since R is unknown, we estimate R using an existing model-order selection algorithm, namely, the HySime algorithm that was introduced for estimating the dimension of the signal subspace under noisy NMF models [40]. Using HySime on the measurements and the re-arranged NMF model (9), we obtain $\hat{R} = 7$. We use the estimated \hat{R} for Nasdac, DowJons, and LL1. For LL1, we set $L = 4$ based on the visual reconstruction quality across the floor. Fig. 17 shows results when 8 random samples are used. One can see that the DL-based approaches, namely, DowJons, Nasdac, and DeepComp work better than the tensor model-based method LL1. This is understandable, since the low-rank model hinged on by LL1 may be grossly violated in such an indoor environment. On the other hand, the DL-based methods may be better suited for such scenarios.

Note that using 8 locations to recover the radio map presents an extremely challenging tensor completion problem, and the proposed methods output satisfactory performance, which is quite encouraging and shows the positive prospects of learning based methods in real world problems. The performance is even more promising when 16 locations are used; see Appendix H-B. Finally, using 166 ground-truth measurements in the first column, we evaluate the SRE for the DL-based methods—the SREs produced by DowJons, Nasdac, and DeepComp are 0.6142, 0.6315, and 0.6337, respectively.

Table X shows the SREs attained by DL-based methods under various ρ 's. The result is averaged over 30 random trials. We used the estimated $\hat{R} = 7$ again. One can see that that DowJons performs consistently better than other methods for all ρ 's.

VI. CONCLUSION

In this work, a DL framework for blind spectrum cartography was proposed. Different than the existing DL-based SC methods that directly learn a data completion network for the observed incomplete radio map, our framework learns a deep network for completing the *individual* SLFs of the emitters. This way, both the offline training cost and the online prediction/completion error can be substantially reduced. A fast NMF-based data desegregation approach was proposed to extract the individual SLFs from the sensor measurements, which are used as inputs of the learned completion network. This two-stage NMF-DL approach is fast and lightweight in the prediction/completion stage. However, it only works when the spectral occupancy of the emitters is relatively sparse, and it suffers from error propagation as other two-stage approaches. To enhance performance, a one-shot optimization-based data completion criterion was proposed, using the learned deep model for the SLFs as structural constraints. This approach was shown to guarantee the recoverability of the radio map tensor under more relaxed conditions relative to the NMF-DL approach—even if noise is present. Synthetic and real data experiments corroborate our design and recovery theory.

REFERENCES

- [1] F. C. Commission, “FCC notice of proposed rulemaking FCC 04-113,” 2004.
- [2] Z. Tian and G. B. Giannakis, “Compressed sensing for wideband cognitive radios,” in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2007, pp. IV-1357–IV-1360.
- [3] X. Fu, N. D. Sidiropoulos, J. H. Tranter, and W.-K. Ma, “A factor analysis framework for power spectra separation and multiple emitter localization,” *IEEE Trans. Signal Process.*, vol. 63, no. 24, pp. 6581–6594, Dec. 2015.
- [4] J. A. Bazerque and G. B. Giannakis, “Distributed spectrum sensing for cognitive radio networks by exploiting sparsity,” *IEEE Trans. Signal Process.*, vol. 58, no. 3, pp. 1847–1862, Mar. 2010.
- [5] Y. Arjoune and N. Kaabouch, “A comprehensive survey on spectrum sensing in cognitive radio networks: Recent advances, new challenges, and future research directions,” *Sensors*, vol. 19, no. 1, p. 126, 2019.
- [6] B. A. Jaywickrama, E. Dutkiewicz, I. Oppermann, G. Fang, and J. Ding, “Improved performance of spectrum cartography based on compressive sensing in cognitive radio networks,” in *Proc. IEEE Int. Conf. Comput. Vis.*, 2013, pp. 5657–5661.
- [7] G. Boccolini, G. Hernandez-Penaloza, and B. Beferull-Lozano, “Wireless sensor network for spectrum cartography based on kriging interpolation,” in *Proc. IEEE 23rd Int. Symp. Pers., Indoor, Mobile Radio Commun.*, 2012, pp. 1565–1570.
- [8] J. A. Bazerque, G. Mateos, and G. B. Giannakis, “Group-lasso on splines for spectrum cartography,” *IEEE Trans. Signal Process.*, vol. 59, no. 10, pp. 4648–4663, Oct. 2011.
- [9] S. Bi, J. Lyu, Z. Ding, and R. Zhang, “Engineering radio maps for wireless resource management,” *IEEE Trans. Wireless Commun.*, vol. 26, no. 2, pp. 133–141, Apr. 2019.
- [10] C. I. Kanatsoulis, X. Fu, N. D. Sidiropoulos, and M. Akakaya, “Tensor completion from regular sub-Nyquist samples,” *IEEE Trans. Signal Process.*, vol. 68, pp. 1–16, Nov. 2019.
- [11] M. Yuan and C.-H. Zhang, “On tensor completion via nuclear norm minimization,” *Found. Comput. Math.*, vol. 16, no. 4, pp. 1031–1068, 2016.
- [12] B. A. Jaywickrama, E. Dutkiewicz, I. Oppermann, and M. Mueck, “Iteratively reweighted compressive sensing based algorithm for spectrum cartography in cognitive radio networks,” in *Proc. IEEE Wireless. Commun. Netw. Conf.*, 2014, pp. 719–724.
- [13] S.-J. Kim and G. B. Giannakis, “Cognitive radio spectrum prediction using dictionary learning,” in *Proc. IEEE Glob. Commun. Conf.*, 2013, pp. 3206–3211.

- [14] M. Hamid and B. Beferull-Lozano, "Non-parametric spectrum cartography using adaptive radial basis functions," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2017, pp. 3599–3603.
- [15] D. Romero, S.-J. Kim, G. B. Giannakis, and R. López-Valcarce, "Learning power spectrum maps from quantized power measurements," *IEEE Trans. Signal Process.*, vol. 65, no. 10, pp. 2547–2560, May 2017.
- [16] G. Zhang, X. Fu, J. Wang, X.-L. Zhao, and M. Hong, "Spectrum cartography via coupled block-term tensor decomposition," *IEEE Trans. Signal Process.*, vol. 68, pp. 3660–3675, Jan. 2020.
- [17] X. Han, L. Xue, F. Shao, and Y. Xu, "A power spectrum maps estimation algorithm based on generative adversarial networks for underlay cognitive radio networks," *Sensors*, vol. 20, no. 1, 2020, Art. no. 311.
- [18] Y. Teganya and D. Romero, "Data-driven spectrum cartography via deep completion autoencoders," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2020, pp. 1–7.
- [19] N. Gillis and S. A. Vavasis, "Fast and robust recursive algorithms for separable nonnegative matrix factorization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 4, pp. 698–714, Apr. 2014.
- [20] X. Fu, W.-K. Ma, T.-H. Chan, and J. M. Bioucas-Dias, "Self-dictionary sparse regression for hyperspectral unmixing: Greedy pursuit and pure pixel search are related," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 6, pp. 1128–1141, Sep. 2015.
- [21] T. King, S. Kopf, T. Haenselmann, C. Lubberger, and W. Effelsberg, "CRAWDAD dataset mannheim/compass (v. 2008-04-11)," Apr. 2008. [Online]. Available: <https://crawdad.org/mannheim/compass/20080411>
- [22] S. Shrestha, X. Fu, and M. Hong, "Deep generative model learning for blind spectrum cartography with NMF-based radio map disaggregation," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, 2021, pp. 4920–4924.
- [23] O. Mehanna and N. D. Sidiropoulos, "Frugal sensing: Wideband power spectrum sensing from few bits," *IEEE Trans. Signal Process.*, vol. 61, no. 10, pp. 2693–2703, May 2013.
- [24] X. Fu, N. D. Sidiropoulos, and W.-K. Ma, "Power spectra separation via structured matrix factorization," *IEEE Trans. Signal Process.*, vol. 64, no. 17, pp. 4592–4605, Sep. 2016.
- [25] X. Fu, S. Ibrahim, H.-T. Wai, C. Gao, and K. Huang, "Block-randomized stochastic proximal gradient for low-rank tensor factorization," *IEEE Trans. Signal Process.*, vol. 68, pp. 2170–2185, Mar. 2020.
- [26] X. Fu, N. Vervliet, L. De Lathauwer, K. Huang, and N. Gillis, "Computing large-scale matrix and tensor decomposition with structured factors: A unified nonconvex optimization perspective," *IEEE Signal Process. Mag.*, vol. 37, no. 5, pp. 78–94, Sep. 2020.
- [27] L. De Lathauwer, "Decompositions of a higher-order tensor in block terms—Part II: Definitions and uniqueness," *SIAM J. Matrix Anal. Appl.*, vol. 30, no. 3, pp. 1033–1066, 2008.
- [28] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Commun. Surv. Tut.*, vol. 11, no. 1, pp. 116–130, Jan.–Mar. 2009.
- [29] A. Goldsmith, *Wireless Communications*. Cambridge, U.K.: Cambridge Univ. Press, 2005.
- [30] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. Int. Conf. Learn. Representations*, 2015.
- [31] P. Stoica *et al.*, *Spectral Analysis of Signals*. Upper Saddle River, NJ, USA: Pearson Prentice Hall, 2005.
- [32] I. J. Goodfellow *et al.*, "Generative adversarial nets," in *Proc. Adv. Neural Inf. Process. Syst.*, 2014.
- [33] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *Proc. Int. Conf. Learn. Representations*, 2013.
- [34] X. Fu, K. Huang, N. D. Sidiropoulos, and W.-K. Ma, "Nonnegative matrix factorization for signal and data analytics: Identifiability, algorithms, and applications," *IEEE Signal Process. Mag.*, vol. 36, no. 2, pp. 59–80, Mar. 2019.
- [35] X. Fu, K. Huang, and N. D. Sidiropoulos, "On identifiability of nonnegative matrix factorization," *IEEE Signal Process. Lett.*, vol. 25, no. 3, pp. 328–332, Mar. 2018.
- [36] N. Gillis, *Nonnegative Matrix Factorization*. Philadelphia, PA, USA: SIAM, 2020.
- [37] N. Gillis and R. Luce, "Robust near-separable nonnegative matrix factorization using linear optimization," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1249–1280, 2014.
- [38] P. Stoica and Y. Selen, "Model-order selection: A review of information criterion rules," *IEEE Signal Process. Mag.*, vol. 21, no. 4, pp. 36–47, Jul. 2004.
- [39] X. Fu and W.-K. Ma, "Robustness analysis of structured matrix factorization via self-dictionary mixed-norm optimization," *IEEE Signal Process. Lett.*, vol. 23, no. 1, pp. 60–64, Jan. 2016.
- [40] J. M. Bioucas-Dias and J. M. P. Nascimento, "Hyperspectral subspace identification," *IEEE Trans. Geosci. Remote Sens.*, vol. 46, no. 8, pp. 2435–2445, Aug. 2008.
- [41] A. Ambikapathi, T.-H. Chan, C.-Y. Chi, and K. Keizer, "Hyperspectral data geometry-based estimation of number of endmembers using p-norm-based pure pixel identification algorithm," *IEEE Trans. Geosci. Remote Sens.*, vol. 51, no. 5, pp. 2753–2769, May 2013.
- [42] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016, vol. 1, no. 2.
- [43] D.-X. Zhou, "The covering number in learning theory," *J. Complexity*, vol. 18, no. 3, pp. 739–767, 2002.
- [44] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*. SIAM, 1995.
- [45] Pytorch, "torch.autograd." [Online]. Available: <https://pytorch.org/docs/stable/autograd.html>
- [46] S. Üreten, A. Yongaolu, and E. Petriu, "A comparison of interference cartography generation techniques in cognitive radio networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2012, pp. 1879–1883.
- [47] M. N. Hindia, A. M. Al-Samman, T. A. Rahman, and T. Yazdani, "Outdoor large-scale path loss characterization in an urban environment at 26, 28, 36, and 38 Ghz," *Phys. Commun.*, vol. 27, pp. 150–160, 2018.

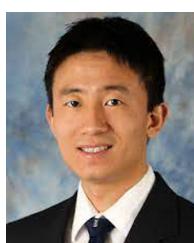


Sagar Shrestha received the B.Eng. degree in electronics and communication engineering from the Pulchowk Campus of Tribhuvan University, Kathmandu, Nepal, in 2016. He is currently working toward the Ph.D. degree with the Department of Electrical Engineering and Computer Science, Oregon State University, Corvallis, OR, USA. His current research interests include statistical machine learning and signal processing.



Xiao Fu (Senior Member, IEEE) received the B.Eng. and M.Sc. degrees from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2005 and 2010, respectively, and the Ph.D. degree in electronic engineering from The Chinese University of Hong Kong (CUHK), Hong Kong, in 2014. From 2014 to 2017, he was a Postdoctoral Associate with the Department of Electrical and Computer Engineering, University of Minnesota, Minneapolis, MN, USA. Since 2017, he has been an Assistant Professor with the School of Electrical and Computer Science, Oregon State University, Corvallis, OR, USA. His research interests include the broad area of signal processing and machine learning.

He was the recipient of the Best Student Paper Award at ICASSP 2014, and Outstanding Postdoctoral Scholar Award at University of Minnesota in 2016. His coauthored papers received the Best Student Paper Awards from IEEE CAMSAP 2015 and IEEE MLSPI 2019. He is a Member of the Sensor Array and Multichannel Technical Committee (SAM-TC) of the IEEE Signal Processing Society (SPS). He is also a Member of the Signal Processing for Multisensor Systems Technical Area Committee (SPMuS-TAC) of EURASIP. He is the Treasurer of the IEEE SPS Oregon Chapter. He is an Editor of *Signal Processing*. He also serves as an Associate Editor of IEEE TRANSACTIONS ON SIGNAL PROCESSING. He was a Tutorial Speaker at ICASSP 2017 and SIAM Conference on Applied Linear Algebra 2021.



Mingyi Hong (Senior Member, IEEE) received the Ph.D. degree from the University of Virginia, Charlottesville, VA, USA, in 2011. He is currently an Associate Professor with the Department of Electrical and Computer Engineering, the University of Minnesota, Minneapolis, MN, USA. His research interests include optimization theory and applications in signal processing and machine learning. He is on the IEEE Signal Processing for Communications and Networking Technical Committee. He is an Associate Editor for the IEEE TRANSACTIONS ON SIGNAL PROCESSING.