Robust Online Prediction of Spectrum Map With Incomplete and Corrupted Observations

Xi Li, *Student Member, IEEE*, Xin Wang, *Member, IEEE*, Tiecheng Song, *Member, IEEE*, and Jing Hu, *Member, IEEE*

Abstract—Spectrum map is an essential tool for a range of emerging applications of 5G and 6G networks. Despite the great efforts that have been put on the construction of spectrum maps, access to accurate and valid spectrum data in dynamically changing environments emphasizes the need for more advanced solutions tailored to such rapidly varying scenarios. To this end, the idea of spectrum map prediction is introduced. In this paper, we address the problem of spectrum map prediction from historical spectrum observations in the dynamically changing environments. The problem is particularly challenging when the available historical spectrum observations are incomplete and corrupted by anomalies. We propose three techniques to solve the problem. First, we combine the spectrum map with prediction functionalities so as to offer a huge potential for efficient resource management and flexible sharing of resources in dynamically changing environments. Second, by fully exploiting the hidden spatial-temporal-spectral structures of the spectrum data and the sparsity of anomalies and missing data, we model the spectrum map as a 3rd-order spectrum tensor and formulate the spectrum map prediction problem as a low-rank tensor completion problem. Third, we design a robust online spectrum map prediction (ROSMP) algorithm based on the alternating direction minimization method, which derives the tensor decomposition factors for a new timeslot based on the update of existing ones rather than re-computing from the scratch. By gradually learning the hidden spatial-temporal-spectral structures of the spectrum data, ROSMP is able to predict and obtain the complete spectrum map with high accuracy. Finally, extensive numerical evaluations using a real spectrum measurement dataset confirm the efficacy and efficiency of ROSMP and show the superiority of ROSMP over the baselines.

Index Terms—Spectrum map, spectrum prediction, tensor completion, online learning

1 Introduction

Spectrum map is a database integrated technique that could provide rich and accurate knowledge of the radio context. It is expected to be an essential tool for unlocking more opportunities in 5G and 6G networks [1]. A range of emerging applications, from adaptive spectrum sensing in cognitive radios [2], proactive resource allocation in anticipatory networks [3], to channel quality prediction in 5G automotive [4], depends crucially on the availability of the spectrum map. Existing studies on the spectrum map mainly focus on spectrum map construction [5]. The main purpose of spectrum map construction is to estimate the unknown elements of the spectrum map based on the spectrum measurements of sparsely distributed and heterogeneous sensors. Usually, updating the spectrum map frequently by construction is expensive and practically inefficient. The updating speed of the spectrum map is relatively slow (e.g., in the time scale of hours or days) [6], which significantly degrades the usefulness of the spectrum map, especially for the time-sensitive applications in dynamically changing environments. The requirement to access to accurate and valid spectrum data in dynamically changing environments emphasizes the need for more advanced solutions. Spectrum map prediction, which combines the spectrum map with prediction functionalities, finds its great potential in solving the problem. Different from spectrum map construction, which relies on the sensing capability of the spectrum sensors, spectrum map prediction can forecast the spectrum map at any time by mining the inherent structure of historical observations.

As studied in our previous work [7], spectrum map prediction heavily depends on the quality of historical observations. However, missing data and anomalies are common in real-world spectrum measurements. In practical environments, the low-cost spectrum sensors easily experience an issue of missing readings due to unexpected hardware failures or communication interruptions [8]. Besides the data loss, the spectrum measurements are also easily polluted by anomalies due to activity from malicious operations, or misconfigurations and failures of network equipments [9]. Consequently, how to perform efficient and dependable spectrum map prediction in the presence of missing data and anomalies remains an open challenge. Yet another major challenge is the requirement of real-time processing capability in the design of spectrum map prediction algorithms. Outdated spectrum map

- *Xi Li is with the School of Information Science and Engineering, Southeast University, Nanjing 211111, China, and with the National Mobile Communication Research Lab, Nanjing 211111, China, and also with the Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794 USA. E-mail: lixixuzhou@seu.edu.cn.*
- *Xin Wang is with the Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY 11794 USA, and also with the Center of Excellence in Wireless and Information Technology, Stony Brook, NY 11794 USA. E-mail: x.wang@stonybrook.edu.*
- *Tiecheng Song and Jing Hu are with the School of Information Science and Engineering, Southeast University, Nanjing 211111, China, and also with the National Mobile Communication Research Lab, Nanjing 211111, China. E-mail: {songtc, louy}@seu.edu.cn.*

Manuscript received 26 Nov. 2019; revised 3 Feb. 2021; accepted 16 May 2021. Date of publication 19 May 2021; date of current version 3 Nov. 2022. (Corresponding author: Xi Li.)
Digital Object Identifier no. 10.1109/TMC.2021.3081715

prediction is generally useless. The existing batch algorithms significantly limit the scalability to analyze massive observations in real time and the computational complexity of the batch algorithms can easily become large with the size of the spectrum map. Thus, it is necessary to design a spectrum map prediction algorithm of high efficiency.

To address these issues, we develop a robust online spectrum map prediction (ROSMP) framework to enable efficient and dependable spectrum map prediction with incomplete and corrupted historical observations. The main contributions of the paper are summarized as follows.

- We formulate the problem of spectrum map prediction based on incomplete and corrupted historical observations as a joint optimization problem of tensor completion and subspace learning by effectively integrating the time series forecasting techniques.
- We design a robust online spectrum map prediction algorithm by taking an alternating direction minimization procedure to efficiently solve the optimization problem.
- We evaluate our designed ROSMP algorithm based on real-world spectrum data and the results show that our algorithm outperforms the baseline algorithms, especially in eliminating the negative effects of anomalies and missing data.

The remainder of the paper is organized as follows. In Section 2, we discuss the related work. The preliminaries are presented in Section 3. We present our system model and analyses on the impact of missing data and anomalies in Section 4. The proposed robust online spectrum map prediction algorithm is presented in Section 5. We conduct extensive experiments to evaluate the performance of our proposed algorithm in Section 6, and finally conclude the work in Section 7.

2 Related Work

In the literature, extensive researches have been carried out for spectrum map construction. Most of the existing studies focus on creating a single spectrum map from sparse sensor measurements via spatial interpolation techniques. In particular, Kriging-based approach [4], Shepard's interpolation technique [10] and Kriged Kalman Filter [11] are used to generate spectrum maps. These construction methods implicitly assume a temporally static spectrum map of one single frequency band, ignoring the inherent temporal and spectral characteristics of the spectrum observations on the construction of a spectrum map. A few studies have also been conducted on multi-dimensional spectrum map construction. In [12], a novel spatial-temporal approach is introduced for securely constructing a spectrum map from a small number of spectrum measurements in the presence of false spectrum measurements. In [13], deep neural network is first utilized for spectrum map construction. A novel model-free approach is proposed for spectrum map construction where propagation phenomena are learned from data. In [14], the spectrum map construction problem is investigated from the spatial-temporal-spectral view. It models the multi-dimensional spectrum map from the perspective of a spectrum tensor and transforms the missing filling problem into a tensor completion problem. By invoking Kalman filtering, a joint tensor completion and prediction scheme is proposed in [15] to enable multi-dimensional spectrum map construction. Despite the great efforts that have been put on the construction of spectrum maps, the relatively slow updating speed of the constructed spectrum map makes it infeasible to be applied in dynamically changing environments, which leads to the investigation of spectrum map prediction in this paper.

Prediction techniques have found their potential in numerous emerging applications, where their availability could be crucial in energy planning, management and conservation [16], natural disaster risk reduction [17], and disease prevention and diagnosis [18], to name just a few. Prediction techniques have also been introduced and developed in wireless communications. As one of the fundamental functions in cognitive radios, spectrum prediction can provide forecast views and has made a quick progress recently. When we only consider the time-varying characteristics of a single frequency band, the spectrum prediction problem can be transformed into a time series forecasting problem to predict the availability of the frequency band using various time series forecasting algorithms [19], including linear regression (LR), recurrent neural network (RNN), and k-nearest-neighbor (KNN). Besides the time-variation characteristics of each frequency band, the correlations among different frequency bands can also be considered for spectrum prediction. The spectrum prediction problem is transformed into a matrix completion and matrix recovery problem [20], which can be effectively solved by integrating the time series forecasting techniques into the matrix completion. In [21], when the periodic characteristics among different days are further considered, long-term spectrum prediction can be performed by solving a tensor completion problem. In terms of the relationship with spectrum maps, prediction techniques can also be utilized to generate a spectrum map in a proactive and energy-efficient manner. In [22], spectrum map prediction is first investigated from the spatial-temporal view. A novel hybridized approach is proposed to first estimate the missing observations of previous timeslots from the spatial view by using the Ordinary Kriging (OK) method, and then predict the spectrum map from the temporal view by establishing a RNN. In our previous work [23], we propose a novel end-to-end deep-learning based model, entitled spatial-temporal-spectral prediction network (STS-PredNet), for spectrum map prediction. Not only the joint spatial-temporal-spectral dependencies but also the temporal diversity of the spectrum observations are considered for the design of STS-PredNet. Different from the above studies, in this paper, we focus on the prediction of spectrum map from the following two aspects: (1) We concurrently consider the impact of missing data and anomalies on spectrum map prediction. (2) We propose an efficient online solution, taking into account the cost for storage and computation.

In the technical aspect, our work benefits from the realm of tensor completion, which is a tensor-based data recovery methodology for extracting and exploiting the inherent structure and redundancy in data. It has been proven to be effective in various fields, including traffic flow prediction [24], internet anomaly detection [25] and image recovery

[26]. However, existing batch tensor completion algorithms cannot be directly applied to spectrum map prediction due to the unique features extracted from the spectrum map. Specifically, the tensor representative of the spectrum map is usually of a large size. The employment of the batch tensor completion algorithms will incur significant time complexity, which will degrades the usefulness of the spectrum map in dynamically changing environments. Besides, the spectrum measurements used to establish the spectrum map are acquired sequentially in time and easily polluted by anomalies. Keeping these features in mind, we are motivated to develop an efficient and robust way to handle the partial missing anomaly-injected noisy tensor and an online tensor completion algorithm is designed in this paper. Online algorithms update previously obtained estimates rather than re-compute new ones from scratch each time a new datum becomes available. The tracking ability to the non-stationary evolution motivates the deployment of online algorithms in different fields. In [27], a sophisticated online learning scheme is designed for green resource allocation in 5G heterogeneous cloud radio access networks, which mitigates interference and maximizes energy efficiency while maintaining QoS requirements for all users. In [28], an online time-structured traffic tensor tracking scheme is designed based on the hankelized structure of the tensor to infer network-level anomalies from indirect link measurements. Different from the above works, in this paper, we propose a robust online spectrum map prediction algorithm based on the alternating direction minimization method by fully considering the unique features of the spectrum map.

3 Preliminaries

In this section, some basic notations, operations and necessary definitions are presented. The following notations are used throughout this paper. Tensors are denoted by uppercase calligraphic letters $(\mathcal{A}, \mathcal{B}, \ldots)$, matrices are written as uppercase boldfaced letters $(\boldsymbol{A}, \boldsymbol{B}, \ldots)$, vectors are represented by lowercase boldfaced letters $(\boldsymbol{a}, \boldsymbol{b}, \ldots)$, and scalars are denoted by normal lowercase letters $(a, b, \ldots)$. An Nth-order tensor can be denoted by $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$, whose elements can be represented by its symbolic name with indexed as subscripts as $x_{i_1,i_2,\ldots,i_N}$, $i_n \in \{1, 2, \ldots, I_n\}$, $n \in \{1, 2, \ldots, N\}$. Accordingly, when $N = 2$, a matrix can be obtained as $\boldsymbol{X} \in \mathbb{R}^{I_1 \times I_2}$, where each element can be denoted by x_{i_1,i_2}. Similarly, when $N = 1$, a vector can be obtained as $\boldsymbol{x} \in \mathbb{R}^{I_1}$ and the elements of the vector can be denoted by x_{i_1}. The key variables used in this paper are summarized in Table 1.

Some operations are quite essential and are applied in this paper. Symbol "$\circ$" represents the vector outer product, while symbol "$\otimes$" represents the matrix Hadamard product. $(.)'$ represents the transposition operation. $\|.\|_0$ stands for the l_0-norm, which counts the number of non-zero elements in the tensor. $\|.\|_1$ represents the l_1-norm, which calculates the sum of the magnitudes of the tensor. $\|.\|_2$ stands for the l_2-norm, which calculates the square root of the sum of the square of all the elements of the vector. $\|.\|_F$ denotes the Frobenius norm, which calculates the square root of the sum of the absolute square of the elements of the tensor.

TABLE 1
Key Variables Used in This Paper

Variables	Explanation
L	Number of grids
F	Number of frequency bands
T	Number of timeslots
$\mathcal{Y}_T$	A 3rd-order tensor constructed based on the historical observations collected during the past $T-1$ timeslots, whose x-axis is the indices of the consecutive grids, y-axis is the indices of frequency bands, and z-axis is the indices of timeslots
$\boldsymbol{Y}_t$	The t^{th} horizontal slice matrix of tensor $\mathcal{Y}_T$
$y_{l,f,t}$	The element of tensor $\mathcal{Y}_T$, denoting the average received signal strength of the f^{th} frequency band in the l^{th} grid at the t^{th} timeslot
$\mathcal{X}_T$	The signal component of tensor $\mathcal{Y}_T$
$\mathcal{A}_T$	The anomaly component of tensor $\mathcal{Y}_T$
$\mathcal{V}_T$	The noise component of tensor $\mathcal{Y}_T$
$\mathcal{P}_{\boldsymbol{\Omega}_T}(.)$	The sampling operator, which sets the entries of a tensor argument not in $\boldsymbol{\Omega}_T$ to zero, and keeps the rest unchanged
$\mathcal{P}_{\boldsymbol{\omega}_t}(.)$	The sampling operator, which sets the entries of a matrix argument not in $\boldsymbol{\omega}_t$ to zero, and keeps the rest unchanged
R	The estimated rank
$\boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}$	The subspace factor matrix of the tensor $\mathcal{Y}_T$, obtained via CP decomposition
$\boldsymbol{b}_r, \boldsymbol{c}_r, \boldsymbol{d}_r$	The rank-one vector in the subspace factor matrices
$\boldsymbol{b}^l, \boldsymbol{c}^f, \boldsymbol{d}^t$	The row vector in the subspace factor matrices of size $\mathbb{R}^{1 \times R}$
$TSF(.)$	The time series forecasting technique
$\bar{\boldsymbol{Y}}_T$	The prefilled elements obtained via time series forecasting
$\hat{\boldsymbol{X}}_T$	The predicted spectrum map
λ	The forgetting factor
p_{miss}	The missing data percentage
$p_{anomaly}$	The anomaly data percentage

$\|.\|_*$ represents the nuclear norm, which calculates the sum of the singular values (SVs) of the tensor.

Definition 1 (Slices of a tensor [25]). *A Tensor can be represented by a set of two-dimensional slices, which are defined by fixing all but two indexes of the tensor. For a 3rd-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$, three kinds of slices, i.e., the frontal, lateral and horizontal slices can be obtained and denoted by $\mathcal{X}_{i_1,:,:}$, $\mathcal{X}_{:,i_2,:}$ and $\mathcal{X}_{:,:,i_3}$, respectively.*

Definition 2 (Mode-n matricization [25], [29]). *Tensor matricization is to unfold a tensor into a matrix format following a predefined rule. As the most commonly used tensor matricization method, mode-n matricization is to unfold a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ along its n^{th} mode into a matrix denoted as $\boldsymbol{X}_{(n)} \in \mathbb{R}^{I_n \times \prod_{k \neq n} I_k}$. An arbitrary element $x_{i_1,i_2,\ldots,i_N}$, of tensor $\mathcal{X}$ can be found in its unfolding matrix $\boldsymbol{X}_{(n)}$ at the position of (i_n, i_c), where $i_c = 1 + \sum_{m=1, m \neq n}^{N} \left[(i_m - 1) \prod_{l=1, l \neq n}^{m-1} I_l \right]$.*

Definition 3 (Rank-1 tensor). *Rank-1 tensor is also called simple tensor or decomposable tensor. The Nth-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}$ is a rank-1 tensor as long as it can be written as the outer product of N vectors as $\mathcal{X} = \boldsymbol{v}^{(1)} \circ \boldsymbol{v}^{(2)} \circ \cdots \circ \boldsymbol{v}^{(N)}$.*

Definition 4 (Tensor rank [30]). *The tensor rank is defined as the minimum number of rank-1 tensors that can be summarized to generate the tensor as*

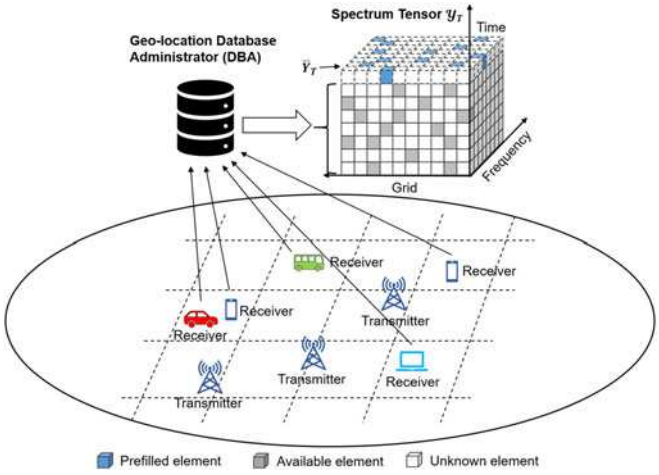


Fig. 1. An exemplary database-driven spectrum map prediction system.

$$rank(\mathcal{X}) = \min\left\{R \in \mathbb{Z}^+ : \exists \boldsymbol{v}_r^{(n)}, s.t. \mathcal{X} = \sum_{r=1}^{R} \boldsymbol{v}_r^{(1)} \circ \cdots \circ \boldsymbol{v}_r^{(N)}\right\}. \tag{1}$$

Definition 5 (Tensor n-rank [31]). *The tensor n-rank is defined as the tuple of the ranks of the unfolding matrices of a tensor, i.e.,* $rank_n(\mathcal{X}) = \big(rank\big(\boldsymbol{X}_{(1)}\big), rank\big(\boldsymbol{X}_{(2)}\big), \ldots, rank\big(\boldsymbol{X}_{(N)}\big)\big)$.

Definition 6 (CANDECOMP/PARAFAC (CP) decomposition [32]). *The idea of CP decomposition is to express a tensor as the sum of a finite number of rank-1 tensors. By this way, the Nth-order tensor* $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ *can be expressed as* $\mathcal{X} = \sum_{r=1}^{R} \boldsymbol{v}_r^{(1)} \circ \cdots \circ \boldsymbol{v}_r^{(N)}$, $\boldsymbol{v}_r^{(n)} \in \mathbb{R}^{I_n}$, $n \in \{1, 2, \ldots, N\}$. *By introducing the factor matrices* $\boldsymbol{V}^{(n)} = \left[\boldsymbol{v}_1^{(n)}, \boldsymbol{v}_2^{(n)}, \ldots, \boldsymbol{v}_R^{(n)}\right] \in \mathbb{R}^{I_n \times R}$, *tensor* $\mathcal{X}$ *can be further expressed as* $\mathcal{X} = [\![\boldsymbol{V}^{(1)}, \boldsymbol{V}^{(2)}, \ldots, \boldsymbol{V}^{(N)}]\!]$.

4 Model and Problem Statement

4.1 System Model

As shown in Fig. 1, we consider a heterogeneous cellular network formed by a set of transmitters and a network of receivers, which consists of sensors placed on vehicles, pedestrian phones and fixed access nodes, located in a geographical region of interest. The transmitters operate on F cellular frequency bands, which are simultaneously monitored by the receivers. The measurements of the receivers are collected and stored in a geo-location database administrator (DBA). Considering that accurately measuring the spectrum at every location in parallel is difficult in practice, we divide the geographical region of interest into a set of small square grids and establish the spectrum map from the perspective of the average received signal strength (ARSS) in the grids. The mean of the collected measurements in each grid can be taken as an estimation of the ARSS. If there is no measurement collected in a grid, the ARSS of the grid can be viewed as unknown. Considering the measurements collected in the past $T-1$ timeslots as historical observations, we model the historical observations stored in the DBA from the perspective of a 3rd-order tensor, $\mathcal{Y}_T \in \mathbb{R}^{L \times F \times T}$, whose x-axis is the indices of the consecutive grids, y-axis is the indices of frequency bands, and z-axis is the indices of timeslots. Each element $y_{l,f,t}$, $l \in \{1, \ldots, L\}$, $f \in \{1, \ldots, F\}$, $t \in \{1, \ldots, T\}$, denotes the ARSS of the f^{th} frequency band in the l^{th} grid at the t^{th} timeslot.

Many factors contribute to the elements in the spectrum tensor, including signals, anomalies, and inherent noise. By assumption, some observations are corrupted by anomalies and are sparsely and randomly distributed. Thus, a general expression can be given as

$$y_{l,f,t} = x_{l,f,t} + a_{l,f,t} + v_{l,f,t}, \tag{2}$$

where $x_{l,f,t}$ denotes the signal of interest, $a_{l,f,t}$ corresponds to the anomaly component, and $v_{l,f,t}$ is the additive noise component. By introducing the tensor $\mathcal{X}_T := \left[x_{l,f,t}\right]$, $\mathcal{A}_T := \left[a_{l,f,t}\right]$, and $\mathcal{V}_T := \left[v_{l,f,t}\right] \in \mathbb{R}^{L \times F \times T}$, the expression in (2) can be batch formulated and further rewritten in a compact tensor form as

$$\mathcal{Y}_T = \mathcal{X}_T + \mathcal{A}_T + \mathcal{V}_T. \tag{3}$$

It should be emphasized that $\mathcal{Y}_T$ can also be viewed as a combination of a set of horizontal slices, each of which represents the spectrum map for a single timeslot. Accordingly, $\mathcal{Y}_T$ can be written as $\mathcal{Y}_T = [\boldsymbol{Y}_1, \boldsymbol{Y}_2, \ldots, \boldsymbol{Y}_T]$. Similarly, $\mathcal{X}_T$, $\mathcal{A}_T$ and $\mathcal{V}_T$ can also be written as a set of horizontal slices as $\mathcal{X}_T = [\boldsymbol{X}_1, \boldsymbol{X}_2, \ldots, \boldsymbol{X}_T]$, $\mathcal{A}_T = [\boldsymbol{A}_1, \boldsymbol{A}_2, \ldots, \boldsymbol{A}_T]$ and $\mathcal{V}_T = [\boldsymbol{V}_1, \boldsymbol{V}_2, \ldots, \boldsymbol{V}_T]$.

Unknown elements are common in the tensor due to the limited number of the receivers, their mobility, as well as the transmission loss in the observation collection processes. More critically, in the case of spectrum map prediction, the last horizontal slice of $\mathcal{Y}_T$ (i.e., $\boldsymbol{Y}_T$) corresponds to the spectrum map to be predicted for the new timeslot T and it is completely unknown. To model the unknown elements in the spectrum tensor, we first collect the tuples corresponding to the available observations in the set $\boldsymbol{\Omega}_T$, and then introduce the sampling operator $\mathcal{P}_{\boldsymbol{\Omega}_T}(.)$ to reset the entries in tensor $\mathcal{Y}_T$ as

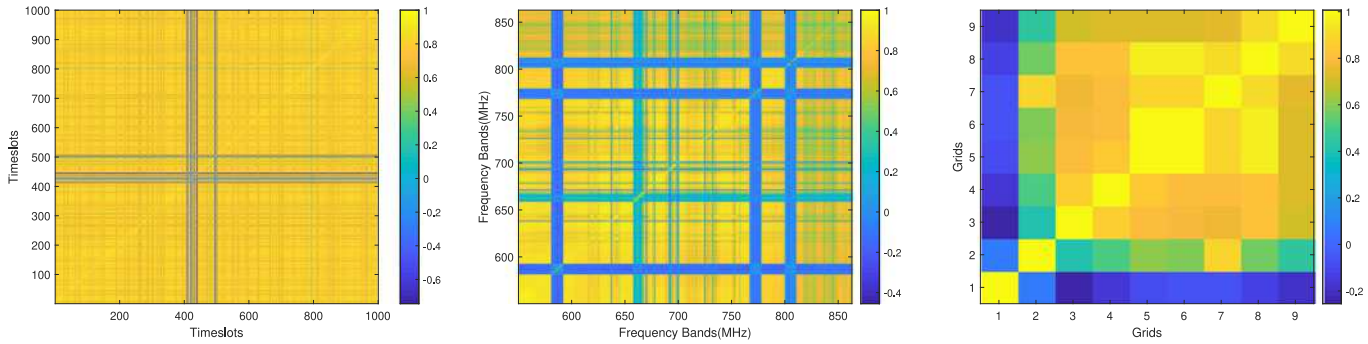
$$\left(\mathcal{P}_{\boldsymbol{\Omega}_T}(\mathcal{Y}_T)\right)_{l,f,t} = \begin{cases} y_{l,f,t}, & (l,f,t) \in \boldsymbol{\Omega}_T \\ 0, & otherwise \end{cases}. \tag{4}$$

4.2 Problem Statement

The objective of spectrum map prediction in this paper is to infer the spectrum map of the forthcoming T^{th} timeslot (i.e., $\boldsymbol{X}_T$), based on the incomplete and corrupted historical observations $\mathcal{P}_{\boldsymbol{\Omega}_{T-1}}(\mathcal{Y}_{T-1})$. Intuitively, this problem mathematically falls into the field of tensor completion. However, two critical issues have to be considered.

First, as the last horizontal slice of the $\mathcal{Y}_T$ is completely unknown, it is infeasible to apply tensor completion directly.

Second, the low-rank feature is the prerequisite for using the tensor completion. According to linear algebra, high correlation of a tensor generally implies that it is of low-rank. To ascertain whether tensor completion can be used here, it is essential to perform correlation analysis of the spectrum tensor. For demonstration purpose, we establish a spectrum tensor based on the real-world measurements of cellular frequency bands (600MHz-850MHz) sensed by Electrosense sensors distributed in an area about 6K sq-meters centering at [-3.77047 40.33701] in Madrid, Spain [33] for a 1000 timeslots duration. As stated in Section 4.1, we divide the whole



(a) Temporal correlation (b) Spectral correlation (c) Spatial correlation

Fig. 2. Temporal, spectral and spatial correlations of the considered spectrum tensor.

area into 3×3 equal-sized square grids and calculate the ARSS in each grid. The temporal/spectral/spatial correlation of the established spectrum tensor is measured using a widely used metric [34], i.e., Pearson correlation coefficient. As shown in Fig. 2, it can be evidently observed that there indeed exists high temporal/spectral/spatial correlation of the spectrum tensor. It is highly likely that the spectrum tensor has a good low-rank structure. To characterize the impact of anomalies, in the following, we re-establish the spectrum tensor in consideration of the sparsely distributed anomalies. Specifically, we artificially inject anomalies to a portion of the entries in the original spectrum tensor, following the standard anomaly injection method used in existing work [35]. Then, we study the distribution of the SVs from the perspective of the unfolding matrices of the new established spectrum tensor. According to the definition of mode-n matricization, three unfolding matrices can be obtained as $\boldsymbol{Y}_{(1)} \in \mathbb{R}^{L\times(F\times T)}$, $\boldsymbol{Y}_{(2)} \in \mathbb{R}^{F\times(L\times T)}$ and $\boldsymbol{Y}_{(3)} \in \mathbb{R}^{T\times(L\times F)}$. The ranks of the matrices can be calculated and collected to denote the tensor n-rank as $rank_n(\mathcal{Y}_T) = \big(rank\big(\boldsymbol{Y}_{(1)}\big), rank\big(\boldsymbol{Y}_{(2)}\big), rank\big(\boldsymbol{Y}_{(3)}\big)\big)$. It can be generally established that $\max(rank_n(\mathcal{Y}_T)) \leq rank(\mathcal{Y}_T)$ [21], which means that the rank of tensor $\mathcal{Y}_T$ is greater than the ranks of the corresponding unfolding matrices and will increase as long as the ranks of the unfolding matrices becomes relatively large. By applying singular value decomposition (SVD) [36], the SVs of $\boldsymbol{Y}_{(1)}$ can be calculated as

$$\boldsymbol{Y}_{(1)} = \boldsymbol{U}\boldsymbol{\Sigma}\boldsymbol{V}', \tag{5}$$

where $\boldsymbol{\Sigma} \in \mathbb{R}^{L\times(F\times T)}$ is a diagonal matrix with the diagonal elements (i.e., the SVs) organized in the decreasing order. Similarly, the SVs of $\boldsymbol{Y}_{(2)}$ and $\boldsymbol{Y}_{(3)}$ can also be calculated by applying SVD, and the cumulative distribution functions (CDFs) of the normalized SVs for different percentages of injected anomalies are plotted in Fig. 3. From Fig. 3, we can observe that more SVs are required to capture the same fraction of the total variance with the increase of the number of anomalies injected into the tensor, which indicates that the ranks of the unfolding matrices increase, so does the rank of the spectrum tensor. Therefore, it can be concluded that the sparsely distributed anomalies have destructive effects on the approximate low-rank structure of the spectrum tensor.

The above findings motivate us in the next section to develop a robust spectrum map prediction framework to support the spectrum tensor that with one whole horizontal slice unknown and may contain anomalies.

5 Problem Formulation and Algorithm Design

5.1 Problem Formulation

To tackle the aforementioned issues, we propose to effectively integrate the time series forecasting technique into tensor completion, so that a prefilling operation is performed in the first step to estimate some notable elements of the forthcoming T^{th} timeslot as

$$\bar{y}_{l,f,T} = \begin{cases} TSF\big(\mathcal{P}_{\boldsymbol{\Omega}_{T-1}}(\mathcal{Y}_{T-1})\big), & (l,f) \in \boldsymbol{S}_{PRE} \\ 0, & otherwise \end{cases}, \tag{6}$$

where $TSF(.)$ represents the time series forecasting technique, and all the indices tuple (l, f) selected for prefilling (i.e., the blue items in Fig. 1) make up the indicator set $\boldsymbol{S}_{PRE}$.

To make the prefilled elements as accurate as possible, we keep the definition of $\boldsymbol{S}_{PRE}$ consistent with the evolution law [37]. Specifically, We calculate the approximate entropy of the time series of each tuple (l, f), and denote it by $En(l, f)$. In general, a lower entropy implies a higher predictability, and vice versa. Thus, $\boldsymbol{S}_{PRE}$ can be constructed by selecting the top p tuples of relative low entropies as

$$\boldsymbol{S}_{PRE} := \arg\min_{F\times L}^{p} En(l, f). \tag{7}$$

Besides, we evaluate the performance of several popular time series forecasting techniques, including statistical models (i.e., autoregressive integrated moving average (ARIMA) [38] and mean [39]), machine learning models (i.e., long short-term memory (LSTM) and KNN) [40], and the prophet model developed by Facebook's Core Data Science team [41], over various (l, f) tuples as shown in Fig. 4. We specify the prediction errors by using the normalized root mean square error (NRMSE) in dB [20]. It is clear in Fig. 4a that even though there is no clear cut winner that consistently outperforms the others across the five techniques, KNN, mean and ARIMA are observed to be the top 3 to provide the best performance for 31, 27 and 18 percent of the instances, respectively. From Fig. 4b, we can see that the KNN model achieves the smallest median NRMSE value, as well as the second shortest interquartile range compared with the other four models, indicating its better performance in terms of both prediction accuracy

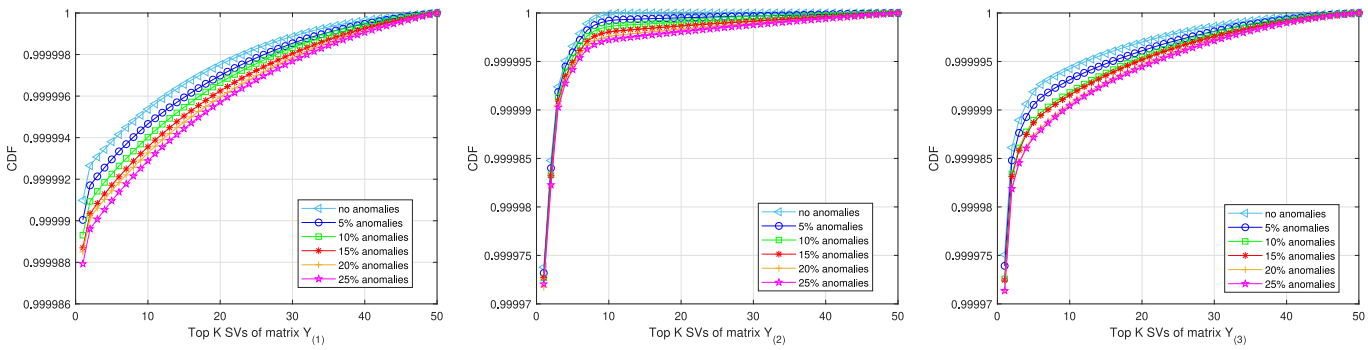


Fig. 3. Illustration of anomaly impact by the CDFs of the top K normalized SVs.

and robustness. Therefore, we define the $TSF(.)$ as a KNN model, namely $\bar{y}_{l,f,T} = KNN(\mathcal{P}_{\mathbf{\Omega}_{T-1}}(\mathcal{Y}_{T-1}))$. Based on the prefilled elements, a new tensor is obtained for further processing as

$$\mathcal{P}_{\mathbf{\Omega}_T}(\bar{\mathcal{Y}}_T) = [\mathcal{P}_{\mathbf{\Omega}_{T-1}}(\mathcal{Y}_{T-1}), \bar{\boldsymbol{Y}}_T], \tag{8}$$

where $\bar{\boldsymbol{Y}}_T = \left[\bar{y}_{l,f,T}\right], l \in \{1,\ldots,L\}, f \in \{1,\ldots,F\}$.

With the new obtained tensor $\mathcal{P}_{\Omega_T}(\bar{\mathcal{Y}}_T)$, tensor completion can then be implemented for spectrum map prediction, i.e., recovering the signal component $\boldsymbol{X}_T$ of the forthcoming T^{th} timeslot. Accordingly, an optimization problem can be formulated by leveraging the inherent low-rank property of $\mathcal{X}_T$ and the sparsity property of $\mathcal{A}_T$ as

$$\begin{aligned} &\min_{\mathcal{X},\mathcal{A}} rank(\mathcal{X}) + \rho\|\mathcal{A}\|_0 \\ &s.t. \left\|\mathcal{P}_{\mathbf{\Omega}_T}(\bar{\mathcal{Y}}_T - \mathcal{X} - \mathcal{A})\right\|_F^2 \leq \varepsilon, \end{aligned} \tag{9}$$

where $rank(.)$ stands for the rank of a tensor, ρ is a positive rank-sparsity controlling parameter, and ε is a noise-related parameter.

It should be noted that both rank and l_0-norm criteria are in general NP-hard to optimize. To make the optimization problem in (9) trackable, one efficient way is the convex relaxation, i.e., replacing $rank(.)$ with its tightest convex surrogate, the nuclear norm $\|.\|_*$, and replacing $\|.\|_0$ with its tightest convex surrogate, the l_1-norm. Accordingly, the optimization problem in (9) can be transformed as

$$\min_{\mathcal{X},\mathcal{A}} \frac{1}{2}\left\|\mathcal{P}_{\mathbf{\Omega}_T}(\bar{\mathcal{Y}}_T - \mathcal{X} - \mathcal{A})\right\|_F^2 + \mu_r\|\mathcal{X}\|_* + \mu_s\|\mathcal{A}\|_1, \tag{10}$$

where $\mu_r, \mu_s \geq 0$ are rank- and sparsity-controlling parameters.

However, it is also easily observed that the replacements result in a non-smooth optimization problem in (10) since the nuclear norm $\|.\|_*$ is not differentiable at the origin, and the size of the optimization problem can become quite large with time. To tackle the above challenges, CP decomposition is introduced in this paper to decompose the signal component tensor $\mathcal{X}$ in (10) by leveraging its low-rank property as [28]

$$\mathcal{X} = \sum_{r=1}^{R} \boldsymbol{b}_r \circ \boldsymbol{c}_r \circ \boldsymbol{d}_r = [[\boldsymbol{B}, \boldsymbol{C}, \boldsymbol{D}]], \tag{11}$$

where R is the rank of $\mathcal{X}$, $\boldsymbol{b}_r \in \mathbb{R}^L$, $\boldsymbol{c}_r \in \mathbb{R}^F$ and $\boldsymbol{d}_r \in \mathbb{R}^T$ denote the rank-one vectors. $\boldsymbol{B} = [\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_R] \in \mathbb{R}^{L\times R}$, $\boldsymbol{C} = [\boldsymbol{c}_1, \boldsymbol{c}_2, \ldots, \boldsymbol{c}_R] \in \mathbb{R}^{F\times R}$ and $\boldsymbol{D} = [\boldsymbol{d}_1, \boldsymbol{d}_2, \ldots, \boldsymbol{d}_R] \in \mathbb{R}^{T\times R}$ represent three factor matrices, and they can also be denoted by row vectors as $\boldsymbol{B} = \left[(\boldsymbol{b}^1)', (\boldsymbol{b}^2)', \ldots, (\boldsymbol{b}^L)'\right]'$, $\boldsymbol{C} = \left[(\boldsymbol{c}^1)', (\boldsymbol{c}^2)', \ldots, (\boldsymbol{c}^F)'\right]'$ and $\boldsymbol{D} = \left[(\boldsymbol{d}^1)', (\boldsymbol{d}^2)', \ldots, (\boldsymbol{d}^T)'\right]'$, where $\{\boldsymbol{b}^l, \boldsymbol{c}^f, \boldsymbol{d}^t\} \in \mathbb{R}^{1\times R}$. Based on these row vectors, each horizontal slice of $\mathcal{X}$ can be decomposed as $\boldsymbol{X}_t = \boldsymbol{B}diag(\boldsymbol{d}^t)\boldsymbol{C}'$, $t \in [1, 2, \ldots, T]$.

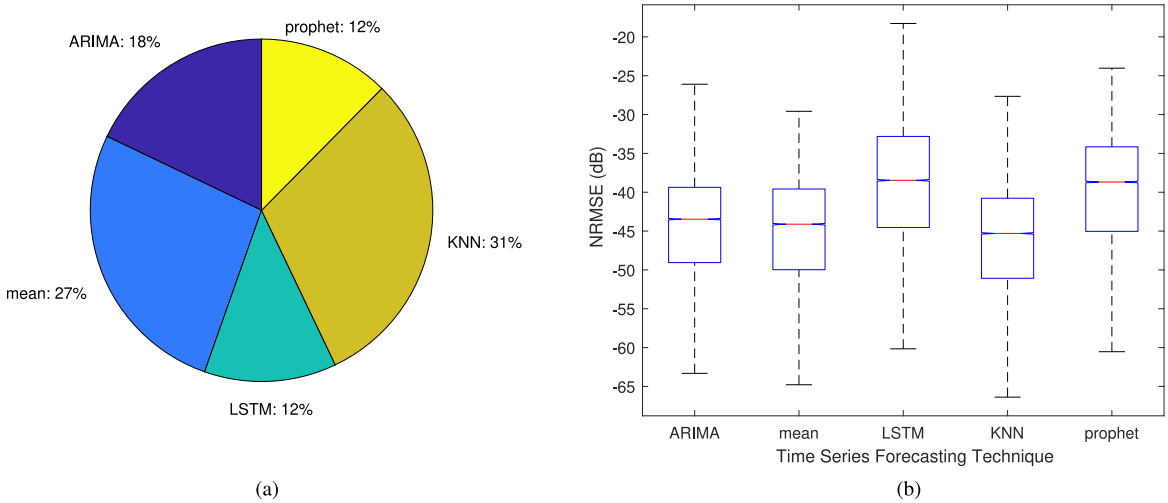


Fig. 4. (a). Pie chart of the percentage a particular forecasting technique performs best (smallest NRMSE). (b). Box plot of the NRMSE for all the forecasting techniques.

Leveraging formula (11), we can then reformulate the optimization problem in (10) as

$$\begin{aligned}\min_{\boldsymbol{B},\boldsymbol{C},\boldsymbol{D},\mathcal{A}} \frac{1}{2}\Bigg\| \underbrace{\mathcal{P}_{\boldsymbol{\Omega}_T}}_{\text{Missing data}} \underbrace{(\bar{\mathcal{Y}}_T - \mathcal{X} - \mathcal{A})}_{\text{Approximation error}} \Bigg\|_F^2 \\ + \underbrace{\frac{\mu_r}{2}\left(\|\boldsymbol{B}\|_F^2 + \|\boldsymbol{C}\|_F^2 + \|\boldsymbol{D}\|_F^2\right)}_{\text{Frobenius-norm regularizer}} + \underbrace{\mu_s\|\mathcal{A}\|_1}_{\text{Sparsity regularizer}}, \quad (12) \\ s.t. \quad \underbrace{\boldsymbol{X}_t = \boldsymbol{B}diag(\boldsymbol{d}^t)\boldsymbol{C}'}_{\text{CP decomposition}}, t \in \{1,\ldots,T\}.\end{aligned}$$

Finally, considering the high storage cost and computation complexity involved in solving the batch optimization problem in (12), we propose to develop online-based setting of subspace learning in this paper. In practice, spectrum measurements are acquired sequentially in time. The requirement of real-time processing capability motivates us to update the previously obtained estimates rather than recompute new ones each time a new datum becomes available. To this end, one possible online counterpart to the batch optimization problem in (12) can be modeled as

$$\begin{aligned}\min_{\boldsymbol{B},\boldsymbol{C},\boldsymbol{d}^T,\boldsymbol{A}_T} \underbrace{\sum_{t=1}^{T}\lambda^{T-t}}_{\text{Online-based}} \Bigg[\underbrace{\frac{1}{2}\left\|\mathcal{P}_{\boldsymbol{\omega}_t}\left(\bar{\boldsymbol{Y}}_t - \boldsymbol{B}diag(\boldsymbol{d}^t)\boldsymbol{C}' - \boldsymbol{A}_t\right)\right\|_F^2}_{\text{Approximation error by low-rank CP decomposition}} \\ + \underbrace{\frac{\mu_r[t]}{2\sum_{t=1}^{T}\lambda^{T-t}}\left(\|\boldsymbol{B}\|_F^2 + \|\boldsymbol{C}\|_F^2\right) + \frac{\mu_r[t]}{2}\|\boldsymbol{d}^t\|_2^2}_{\text{Frobenius and } l_2 \text{ norm regularizer}} \\ + \underbrace{\mu_s[t]\|\boldsymbol{A}_t\|_1}_{\text{Sparsity regularizer}} \Bigg],\end{aligned} \quad (13)$$

where set $\boldsymbol{\omega}_t$ contains the indices of the available elements of the t^{th} horizontal slice, while the corresponding sampling operator $\mathcal{P}_{\boldsymbol{\omega}_t}(.)$ set the entries not in $\boldsymbol{\omega}_t$ as zero and keeps the rest unchanged. $0 < \lambda < 1$ is the so-termed forgetting factor, which enables observations in distant past to be forgotten with exponentially decaying and more importance to be assigned to recent observations. In the case of infinite memory, i.e., $\lambda = 1$, the formula (13) coincides with the batch estimator (12). In the following, we solve the online optimization problem in (13) to estimate the CP factor matrices $\{\boldsymbol{B}, \boldsymbol{d}, \boldsymbol{C}\}$ and the anomaly component matrix $\boldsymbol{A}$ at timeslot T, and finally obtain the predicted spectrum map as $\hat{\boldsymbol{X}}_T = \boldsymbol{B}diag(\boldsymbol{d}^T)\boldsymbol{C}'$.

5.2 Solutions and Online Algorithm Design

In this subsection, we give the detailed solutions of the optimization problem in (13), where the optimization variables are $\{\boldsymbol{B}, \boldsymbol{d}, \boldsymbol{C}\}$ and $\boldsymbol{A}$. Although the optimization problem in (13) is not jointly convex in all the variables, it is convex in each individual variable when the other variables are fixed. Thus, we develop an online and computationally efficient solver based on the alternating direction minimization method to successively solve the lower-dimensional convex problems by updating the unknown variables alternatively in this paper.

5.2.1 Update $\boldsymbol{d}[T]$

With fixed $\boldsymbol{B}[T-1]$ and $\boldsymbol{C}[T-1]$, we first calculate $\boldsymbol{d}[T]$ via an l_2-norm regularized least square problem, i.e.,

$$\begin{aligned}\boldsymbol{d}[T] &= \arg\min_{\boldsymbol{d}} \frac{1}{2}\Big[\left\|\boldsymbol{\omega}_T \otimes \left[\bar{\boldsymbol{Y}}_T - \boldsymbol{B}[T-1]diag(\boldsymbol{d})(\boldsymbol{C}[T-1])'\right]\right\|_F^2 \\ &\quad + \mu_r[T]\|\boldsymbol{d}\|_2^2\Big] \\ &= \Bigg(\Bigg[\mu_r[T]\boldsymbol{I}_R + \sum_{l=1}^{L}\sum_{f=1}^{F}(\boldsymbol{\omega}_T)_{l,f}\left(\boldsymbol{g}_{l,f}[T]\right)'\boldsymbol{g}_{l,f}[T]\Bigg]^{-1} \\ &\quad \cdot \Bigg[\sum_{l=1}^{L}\sum_{f=1}^{F}(\boldsymbol{\omega}_T)_{l,f}(\bar{\boldsymbol{Y}}_T)_{l,f}\left(\boldsymbol{g}_{l,f}[T]\right)'\Bigg]\Bigg)',\end{aligned} \quad (14)$$

where $\boldsymbol{g}_{l,f}[T] = \boldsymbol{b}^l[T-1] \otimes \boldsymbol{c}^f[T-1] \in \mathbb{R}^{1\times R}$. $\boldsymbol{\omega}_T \in \mathbb{R}^{L\times F}$ denotes a binary $\{0,1\}$-matrix with $(\boldsymbol{\omega}_T)_{l,f} = 1$ for the prefilled elements, and $(\boldsymbol{\omega}_T)_{l,f} = 0$ otherwise. $\boldsymbol{I}_R$ is an identity matrix of size R.

5.2.2 Update $\boldsymbol{A}[T]$

Based on the updated $\boldsymbol{d}[T]$, we introduce $\boldsymbol{M}[T] = \bar{\boldsymbol{Y}}_T - \boldsymbol{B}$ $t[T-1]diag(\boldsymbol{d}[T])(\boldsymbol{C}[T-1])'$ into (13), and carry out the minimization with respect to $\boldsymbol{A}[T]$ as

$$\boldsymbol{A}[T] = \arg\min_{\boldsymbol{A}} \frac{1}{2}\Big[\left\|\boldsymbol{\omega}_T \otimes [\boldsymbol{M}[T] - \boldsymbol{A}]\right\|_F^2 + \mu_s[T]\|\boldsymbol{A}\|_1\Big]. \quad (15)$$

Here, we decompose the inner objective to be minimized in (15) into a parallel set of smaller sub-objectives, one for each column of $\boldsymbol{A}[T]$ as

$$\begin{aligned}\boldsymbol{a}_f[T] &= \arg\min_{\boldsymbol{a}_f} \frac{1}{2}\left\|diag(\boldsymbol{\varpi}_f[T])(\boldsymbol{m}_f[T] - \boldsymbol{a}_f)\right\|_2^2 + \mu_s[T]\left\|\boldsymbol{a}_f\right\|_1 \\ &= \arg\min_{\boldsymbol{a}_f} \frac{1}{2}\left\|\boldsymbol{\vartheta}_f[T] - diag(\boldsymbol{\varpi}_f[T])\boldsymbol{a}_f\right\|_2^2 + \mu_s[T]\left\|\boldsymbol{a}_f\right\|_1,\end{aligned} \quad (16)$$

where $\boldsymbol{a}_f[T]$, $\boldsymbol{\varpi}_f[T]$ and $\boldsymbol{m}_f[T]$ represents the f^{th} column of matrixes $\boldsymbol{A}[T]$, $\boldsymbol{\omega}_T$ and $\boldsymbol{M}[T]$ respectively. Obviously, these subproblems are typical Lasso regression problems, and an efficient solver can be found in [42].

5.2.3 Update $\boldsymbol{B}[T]$ and $\boldsymbol{C}[T]$

In the next step of the alternating direction minimization procedure, we update $\boldsymbol{B}[T]$ and $\boldsymbol{C}[T]$ alternatively by addressing a second-order stochastic gradient based on the recursive least square method with forgetting parameters. Since $\boldsymbol{B}$ and $\boldsymbol{C}$ are coupled with each other in the optimization problem, i.e., the calculation of $\boldsymbol{B}[T]$ requires $\boldsymbol{C}[T-1]$, and the calculation of $\boldsymbol{C}[T]$ requires $\boldsymbol{B}[T]$, we first fix $\{\boldsymbol{d}, \boldsymbol{C}, \boldsymbol{A}\}$ and carry out the minimization in (13) with respect to $\boldsymbol{B}$ as

$$\min_{\boldsymbol{B}} \frac{1}{2}\sum_{t=1}^{T}\lambda^{T-t}\left\|\boldsymbol{\omega}_t \otimes \left[\boldsymbol{Z}_t - \boldsymbol{B}diag(\boldsymbol{d}^t)left(\boldsymbol{C}[T-1])'\right]\right\|_F^2 + \frac{\mu_r[T]}{2}\|\boldsymbol{B}\|_F^2, \quad (17)$$

where $\boldsymbol{Z}_t = \bar{\boldsymbol{Y}}_t - \boldsymbol{A}_t$. The objective function in (17) can be decomposed into a parallel set of smaller problems, one for each row of $\boldsymbol{B}$ as

$$\min_{\boldsymbol{b}^l\in\mathbb{R}^{1\times R}}\frac{1}{2}\sum_{t=1}^{T}\lambda^{T-t}\sum_{f=1}^{F}\Big[(\boldsymbol{\omega}_t)_{l,f}\Big((\boldsymbol{Z}_t)_{l,f}-\boldsymbol{b}^l\boldsymbol{\alpha}_f[t]\Big)\Big]^2+\frac{\mu_r[T]}{2}\left\|\boldsymbol{b}^l\right\|_2^2, \tag{18}$$

where $\boldsymbol{\alpha}_f[t]=diag\big(\boldsymbol{d}^t\big)\big(\boldsymbol{c}^f[t-1]\big)'$. Defining $F\big(\boldsymbol{b}^l[T]\big)$ as the inner objective to be minimized in (18), we can obtain the optimal $\boldsymbol{b}^l[T]$ by differentiating $F\big(\boldsymbol{b}^l[T]\big)$ such that $\frac{\partial F\left(\boldsymbol{b}^l[T]\right)}{\partial\left(\boldsymbol{b}^l[T]\right)}=0$, i.e.,

$$\boldsymbol{b}^l[T]=\Big(\big[\boldsymbol{RB}_l[T]\big]^{-1}\boldsymbol{S}_l[T]\Big)', \tag{19}$$

where $\boldsymbol{RB}_l[T]$ and $\boldsymbol{S}_l[T]$ are defined by

$$\boldsymbol{RB}_l[T]=\sum_{t=1}^{T}\sum_{f=1}^{F}\lambda^{T-t}(\boldsymbol{\omega}_t)_{l,f}\boldsymbol{\alpha}_f[t]\big(\boldsymbol{\alpha}_f[t]\big)'+\mu_r[T]\boldsymbol{I}_R, \tag{20}$$

$$\boldsymbol{S}_l[T]=\sum_{t=1}^{T}\sum_{f=1}^{F}\lambda^{T-t}(\boldsymbol{\omega}_t)_{l,f}(\boldsymbol{Z}_t)_{l,f}\boldsymbol{\alpha}_f[t]. \tag{21}$$

It is easy to find both $\boldsymbol{RB}_l[T]$ and $\boldsymbol{S}_l[T]$ can be updated based on their previous values as

$$\begin{aligned}\boldsymbol{RB}_l[T]=\lambda\boldsymbol{RB}_l[T-1]&+\sum_{f=1}^{F}(\boldsymbol{\omega}_T)_{l,f}\boldsymbol{\alpha}_f[T]\big(\boldsymbol{\alpha}_f[T]\big)'\\&+(\mu_r[T]-\lambda\mu_r[T-1])\boldsymbol{I}_R,\end{aligned} \tag{22}$$

$$\boldsymbol{S}_l[T]=\lambda\boldsymbol{S}_l[T-1]+\sum_{f=1}^{F}(\boldsymbol{\omega}_T)_{l,f}(\boldsymbol{Z}_T)_{l,f}\boldsymbol{\alpha}_f[T]. \tag{23}$$

Therefore, $\boldsymbol{b}^l[T]$ can be calculated by updating its previous value as

$$\begin{aligned}\boldsymbol{b}^l[T]=&\Big(\big[\boldsymbol{RB}_l[T]\big]^{-1}\boldsymbol{RB}_l[T]\big(\boldsymbol{b}^l[T-1]\big)'\\&-\big[\boldsymbol{RB}_l[T]\big]^{-1}\left(\sum_{f=1}^{F}(\boldsymbol{\omega}_T)_{l,f}\boldsymbol{\alpha}_f[T]\big(\boldsymbol{\alpha}_f[T]\big)'\right)\big(\boldsymbol{b}^l[T-1]\big)'\\&-\big[\boldsymbol{RB}_l[T]\big]^{-1}[(\mu_r[T]-\lambda\mu_r[T-1])\boldsymbol{I}_R]\big(\boldsymbol{b}^l[T-1]\big)'\\&+\big[\boldsymbol{RB}_l[T]\big]^{-1}\sum_{f=1}^{F}(\boldsymbol{\omega}_T)_{l,f}(\boldsymbol{Z}_T)_{l,f}\boldsymbol{\alpha}_f[T]\Big)'.\end{aligned} \tag{24}$$

Meanwhile, a similar updating process can be performed to calculate each row of $\boldsymbol{C}$ as

$$\boldsymbol{c}^f[T]=\Big(\big[\boldsymbol{RC}_f[T]\big]^{-1}\boldsymbol{U}_f[T]\Big)', \tag{25}$$

where $\boldsymbol{RC}_f[T]$ and $\boldsymbol{U}_f[T]$ are defined by

$$\boldsymbol{RC}_f[T]=\sum_{t=1}^{T}\sum_{l=1}^{L}\lambda^{T-t}(\boldsymbol{\omega}_t)_{l,f}(\boldsymbol{\beta}_l[t])'\boldsymbol{\beta}_l[t]+\mu_r[T]\boldsymbol{I}_R, \tag{26}$$

$$\boldsymbol{U}_f[T]=\sum_{t=1}^{T}\sum_{l=1}^{L}\lambda^{T-t}(\boldsymbol{\omega}_t)_{l,f}(\boldsymbol{Z}_t)_{l,f}(\boldsymbol{\beta}_l[t])', \tag{27}$$

and $\boldsymbol{\beta}_l[T]=\boldsymbol{b}^l[T]diag\big(\boldsymbol{d}^T\big)$. Similarly, both $\boldsymbol{RC}_f[T]$and $\boldsymbol{U}_f[T]$ can be updated based on their previous values as

$$\begin{aligned}\boldsymbol{RC}_f[T]=\lambda\boldsymbol{RC}_f[T-1]&+\sum_{l=1}^{L}(\boldsymbol{\omega}_T)_{l,f}(\boldsymbol{\beta}_l[T])'\boldsymbol{\beta}_l[T]\\&+(\mu_r[T]-\lambda\mu_r[T-1])\boldsymbol{I}_R,\end{aligned} \tag{28}$$

$$\boldsymbol{U}_f[T]=\lambda\boldsymbol{U}_f[T-1]+\sum_{l=1}^{L}(\boldsymbol{\omega}_T)_{l,f}(\boldsymbol{Z}_T)_{l,f}(\boldsymbol{\beta}_l[T])'. \tag{29}$$

Thus, $\boldsymbol{c}^f[T]$ can be calculated by updating its previous value as

$$\begin{aligned}\boldsymbol{c}^f[T]=&\Big(\big[\boldsymbol{RC}_f[T]\big]^{-1}\boldsymbol{RC}_f[T]\big(\boldsymbol{c}^f[T-1]\big)'\\&-\big[\boldsymbol{RC}_f[T]\big]^{-1}\left(\sum_{l=1}^{L}(\boldsymbol{\omega}_T)_{l,f}(\boldsymbol{\beta}_l[T])'\boldsymbol{\beta}_l[T]\right)\big(\boldsymbol{c}^f[T-1]\big)'\\&-\big[\boldsymbol{RC}_f[T]\big]^{-1}[(\mu_r[T]-\lambda\mu_r[T-1])\boldsymbol{I}_R]\big(\boldsymbol{c}^f[T-1]\big)'\\&+\big[\boldsymbol{RC}_f[T]\big]^{-1}\sum_{l=1}^{L}(\boldsymbol{\omega}_T)_{l,f}(\boldsymbol{Z}_T)_{l,f}(\boldsymbol{\beta}_l[T])'\Big)'.\end{aligned} \tag{30}$$

After the above mathematical derivations, the overall algorithm to solve the optimization problem in (13) is summarized in Algorithm 1 and named as ROSMP, where an estimated spectrum map $\hat{\boldsymbol{X}}_T$ of the forthcoming timeslot T is finally accomplished by eliminating the negative effects of missing data and anomalies.

Algorithm 1. Robust Online Spectrum Map Prediction (ROSMP)

Input: $\mathcal{Y}_{T-1}, \boldsymbol{\Omega}_{T-1}, \mu_r, \mu_s, \lambda$;
Output: The predicted spectrum map $\hat{\boldsymbol{X}}_T$;
1: **Initialize:** Initialize $T=1$. Set $\boldsymbol{B}[1]$ and $\boldsymbol{C}[1]$ at random, while $\boldsymbol{d}[1]=\boldsymbol{0}$. The temporally used variables are set as $\boldsymbol{RB}_l[1]=\boldsymbol{0}$, $\boldsymbol{RC}_f[1]=\boldsymbol{0}$, $\boldsymbol{S}_l[1]=\boldsymbol{0}$ and $\boldsymbol{U}_f[1]=\boldsymbol{0}$;
2: **while** Spectrum map prediction is ongoing **do**
3: $T=T+1$;
4: Prefill the spectrum map and obtain $\bar{\boldsymbol{Y}}_T$; (6)
5: Construct tensor $\mathcal{P}_{\boldsymbol{\Omega}_T}(\mathcal{Y}_T)$ for further processing; (8)
6: Calculate the projection coefficient vector $\boldsymbol{d}[T]$; (14)
7: Obtain the anomaly component $\boldsymbol{A}[T]$; (16)
8: Update the subspace factor matrices:
9: **for** $l=1,2,\ldots,L$ **do**
10: Update $\boldsymbol{RB}_l[T]$; (22)
11: Update $\boldsymbol{S}_l[T]$; (23)
12: Obtain $\boldsymbol{b}^l[T]$; (19)
13: **end for** ;
14: **for** $f=1,2,\ldots,F$ **do**
15: Update $\boldsymbol{RC}_f[T]$; (28)
16: Update $\boldsymbol{U}_f[T]$; (29)
17: Obtain $\boldsymbol{c}^f[T]$; (25)
18: **end for** ;
19: Return the predicted spectrum map $\hat{\boldsymbol{X}}_T=\boldsymbol{B}diag\big(\boldsymbol{d}^T\big)\boldsymbol{C}'$;
20: **end while** ;

5.3 Computational Complexity Analysis

It is found that the computational complexity per iteration of the proposed ROSMP algorithm keeps consistent with time. The calculation of $\boldsymbol{d}[T]$ in (14) corresponds to the computation in line 6 of Algorithm 1, where the dominant cost includes a $R\times R$ matrix inversion with a computational

cost $\mathcal{O}(R^3)$, the matrix multiplication with a computational cost $\mathcal{O}(|\boldsymbol{\omega}_T|R^2)$, and the matrix addition with a computational cost $\mathcal{O}(|\boldsymbol{\omega}_T|)$. The estimation of $\boldsymbol{A}[T]$ in (16) corresponds to the computation on line 7 of Algorithm 1, where a Lasso estimator is used to calculate each column of $\boldsymbol{A}[T]$ with a computational cost $\mathcal{O}(L^3)$, thus a total of $\mathcal{O}(FL^3)$ is required. As for the calculation of $\boldsymbol{B}[T]$ on lines 9 $\sim$ 13 of Algorithm 1, each row of $\boldsymbol{B}[T]$ requires $\mathcal{O}\Big(\Big(\Big|(\boldsymbol{\omega}_T)_{l,:}\Big|+1\Big)R^2\Big)+\mathcal{O}\Big(\Big|(\boldsymbol{\omega}_T)_{l,:}\Big|R\Big)+\mathcal{O}(R^3)$ for matrix addition, multiplication, transposition and inverse in (24). Thus, a total of $\mathcal{O}((|\boldsymbol{\omega}_T|+L)R^2)+\mathcal{O}(|\boldsymbol{\omega}_T|R)+L\mathcal{O}(R^3)$ is required for the calculation of $\boldsymbol{B}[T]$, where $|\boldsymbol{\omega}_T|$ denotes the number of prefilled elements. Similarly, the calculation of $\boldsymbol{C}[T]$ on lines 14 $\sim$ 18 of Algorithm 1 requires $\mathcal{O}((|\boldsymbol{\omega}_T|+F)R^2)+\mathcal{O}(|\boldsymbol{\omega}_T|R)+F\mathcal{O}(R^3)$. Consequently, the total computational complexity at each iteration in Algorithm 1 results in $\mathcal{O}((L+F+1)R^3)+\mathcal{O}((L+F+3|\boldsymbol{\omega}_T|)R^2)+\mathcal{O}(2|\boldsymbol{\omega}_T|R)+\mathcal{O}(|\boldsymbol{\omega}_T|)+\mathcal{O}(FL^3)$, and indicates that the number of locations, L, is dominant since rank R is assumed to be small.

6 Experimental Evaluations

In this section, we evaluate the performance of our proposed ROSMP algorithm based on the real-world spectrum measurements from ElectroSense [33]. We run all the experiments on a laptop computer with 2.20 GHz Intel Core i7 CPU and 16 GB RAM using MATLAB R2018b.

Data Preparation. An aggregated spectrum measurement dataset of 750 - 800 MHz frequency bands sensed by ElectroSense[1] sensors located in an area about 6K sq-meters centering at [-3.77047 40.33701] in Madrid, Spain is used for the evaluations. The statistics of the dataset is described in Table 2. We divide the whole area into 3×3 equal-sized square grids, and construct a spectrum tensor based on the measurement data as described in Section 4.1. As the measurements are collected every 10 minutes from 14th January, 2021 to 21st January, 2021 and the resolution bandwidth is set as 1 MHz, the size of the spectrum tensor can be easily obtained as (9,50,1008). The complete and anomaly-free tensor can be viewed as the ground truth for spectrum map prediction. Especially, its last horizontal slice matrix corresponds to the spectrum map to be predicted. To characterize the impact of missing data and anomalies on spectrum map prediction, we set up another spectrum tensor by artificially injecting missing data and anomalies into the original spectrum tensor. We first inject anomalies to a portion of randomly selected elements following the standard anomaly injection method used in existing work [35]. To be specific, we use the exponential weighted moving average method to predict the future entries based on their past values (i.e., $y_t = \alpha x_t + (1-\alpha)y_{t-1}$, where $\alpha = 0.8$ in our experiments) and use the maximum difference between the actual and predicted values as the anomaly size to be injected. We vary the fraction of entries to injected anomalies from 5 to 30 percent. Then we randomly drop a portion of elements to simulate the missing data. We vary the fraction of elements to be deleted from 10 to 90 percent.

Parameter Setup. For the proposed ROSMP algorithm, we set the number of prefilled elements as $p = 0.1 \times 9 \times 50$. The chosen ranges of the rank- and sparsity-controlling parameters are derived from the optimality conditions for (13), which indicates that for μ_r and μ_s selected from the corresponding ranges, an optimal solution can always be found. Along the lines of [7], [28], we empirically set the penalty parameters as $\mu_r = 10^{-3}$ and $\mu_s = 6 \times 10^{-2}$. We also set the forgetting factor $\lambda = 0.7$ to enable measurements in the past to be forgotten with exponentially decaying. The estimated rank is set as $R = 10$.

TABLE 2
Statistics of the Spectrum Measurement Dataset

Dataset	Electrosense spectrum measurement data
Data type	Received signal power in dBm
Location	Madrid, Spain
Number of sensors	7 (test_yago, imdea_6ghz, imdea_adsb, madrid_city_1, cvientos_rooftop2, donostia, test_yago_3)
Time span	14th Jan. 2021 - 21st Jan. 2021
Resolution time interval	10 mins
Frequency span	750 - 800 MHz
Resolution bandwidth	1 MHz

Evaluation Metric. To quantitatively evaluate the spectrum map prediction performance of our proposed ROSMP algorithm, we adopt one of the most widely recognized evaluation metrics, i.e., the normalized root mean square error (NRMSE) in dB to quantify the magnitude of error as

$$NRMSE(dB) = 10\log_{10}\left(\frac{\left\|\hat{\boldsymbol{X}}_T - \boldsymbol{X}_T\right\|_F^2}{\|\boldsymbol{X}_T\|_F^2}\right), \tag{31}$$

where $\boldsymbol{X}_T$ and $\hat{\boldsymbol{X}}_T$ are the ground truth and the predicted spectrum map, respectively.

Baselines. For the comparison purpose, there are three candidate baselines in the literature, which are:

- Kriging-RNN [22]: A novel hybridized approach combining the Kriging-based statistical spatial interpolation and the recurrent neural network based temporal prediction. To the best of our knowledge, Kriging-RNN is the first algorithm dedicated for spectrum map prediction.
- RLSP [21]: A joint spectrum prediction scheme, which integrates time series forecasting technique and tensor completion for long-term spectrum prediction. Notably, our proposed ROSMP algorithm is quite different from the RLSP algorithm since ROSMP works in an online manner, while RLSP is a batch algorithm.
- KNN: To better illustrate the effectiveness of our designed framework, we also compare our proposed ROSMP algorithm with the approach that only applies the time series forecasting technique KNN to predict all the elements of the spectrum map.

6.1 Impact of Missing Data and Anomalies

Fig. 5 illustrates the performance of our proposed ROSMP algorithm under different percentages of missing data, as well as different percentages of injected anomalies in terms

1. https://electrosense.org/

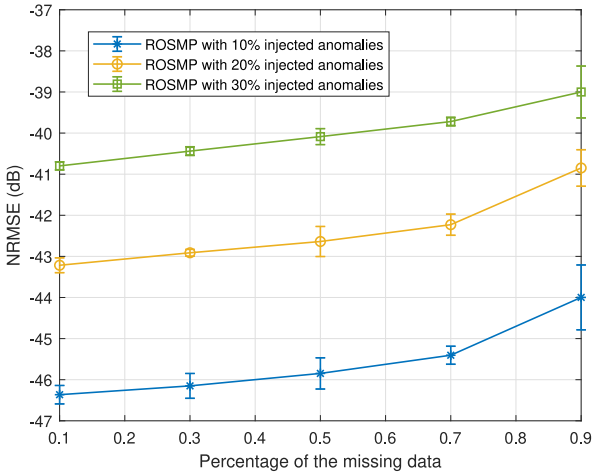


Fig. 5. NRMSE performance of our proposed ROSMP algorithm versus different percentages of missing data, as well as different percentages of injected anomalies.

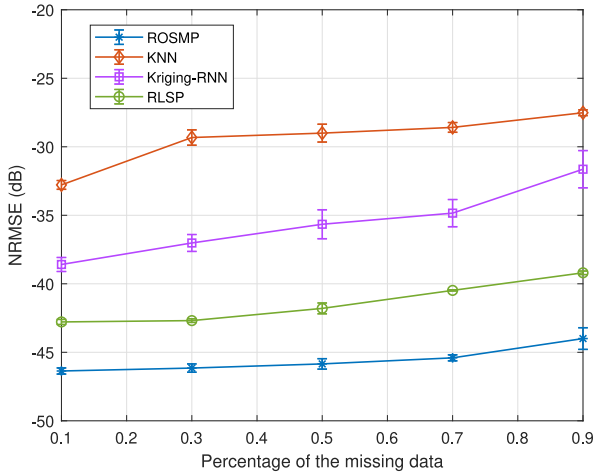


Fig. 6. Comparison of spectrum map prediction accuracy with 10 percent injected anomalies and different percentages of missing data.

of NRMSE, i.e., the prediction error of the spectrum map. We evaluate the cases where the fraction of entries to inject anomalies is set as $p_{anomaly} = 10\%, 20\%, 30\%$ respectively, and the missing data percentage varies from 10 to 90 percent. We can see that while the NRMSE value of our proposed ROSMP algorithm increases with the increase of the percentage of missing data, it remains at a relatively low level and only increases sharply when the missing rate is above 70 percent. Meanwhile, the NRMSE value of our proposed ROSMP algorithm also increases with the increase of the percentage of injected anomalies due to their destructive effect on the rank of the tensor.

To better illustrate the performance of our proposed ROSMP algorithm, we further compare it with the three baselines as shown in Figs. 6 and 7. Fig. 6 compares the spectrum map prediction error of different algorithms under a missing data percentage from 10 to 90 percent in terms of NRMSE. The fraction of entries to inject anomalies is set as $p_{anomaly} = 10\%$. It is obvious that our proposed ROSMP algorithm has a higher prediction accuracy than the three baseline algorithms for any percentage of missing data. As the missing data percentage increases, the NRMSE values of all peer algorithms raise sharply, while the errors of our ROSMP algorithm is well controlled thanks to the effectiveness of our tensor completion scheme in filling the missing data. Particularly, the gap between our ROSMP algorithm and the three baselines increases significantly with the increase of missing data percentage, which demonstrates the robustness of our ROSMP scheme in handling the missing data. Compared with the KNN model which is more sensitive to the missing data, our proposed ROSMP scheme again shows its superiority in handling the problem by incorporating the tensor completion for spectrum map prediction.

Fig. 7 compares the spectrum map prediction error of different algorithms under various percentages of injected anomalies. We evaluate the case where the percentage of injected anomalies varies from 5 to 30 percent, while the missing data percentage is set as 20 percent. Similar as the observations in Fig. 6, it is shown in Fig. 7 that the proposed ROSMP always outperforms the baseline algorithms. However, as the percentage of injected anomalies increases, the prediction accuracy of our ROSMP algorithm, the RLSP algorithm and the Kriging-RNN algorithm degrade slightly, while the prediction error of the KNN model always keeps at a relatively high level. The lower NRMSE of ROSMP under any ratio of injected anomalies demonstrates the robustness of our ROSMP algorithm against the injected anomalies and the effectiveness in improving the quality of prediction by alleviating the side effects of anomalies for spectrum map prediction.

6.2 Anomaly Detection Capability

To the best of our knowledge, RLSP algorithm and our proposed ROSMP algorithm are among the first that take into account the impact of anomalies and provide the anomaly detection function in spectrum map prediction. The anomaly detection capability is evaluated in Fig. 8, using the metrics false positive rate (FPR) and true positive rate (TPR) [43], with various percentages of anomalies injected randomly. While TPR degrades and FPR increases with the increase of percentage of injected anomalies thus the number of outliers, our ROSMP algorithm always achieves better anomaly detection performance with lower FPR and higher TPR under all scenarios compared with the RLSP algorithm. Even when 10 percent anomalies exist, the FPR of our ROSMP is below 2 percent and the TPR is above

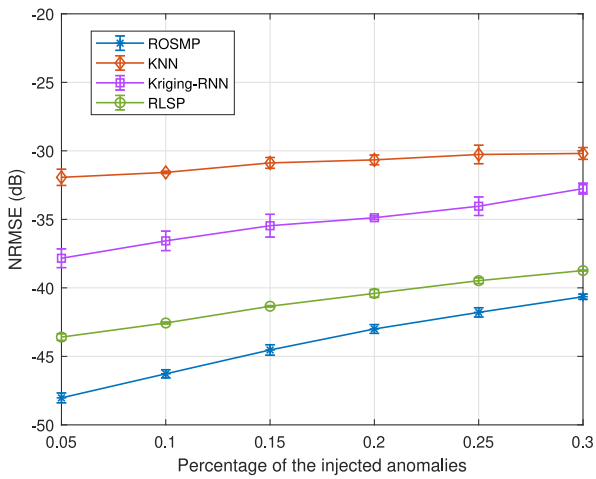


Fig. 7. Comparison of spectrum map prediction accuracy with 20 percent missing data and different percentages of injected anomalies.

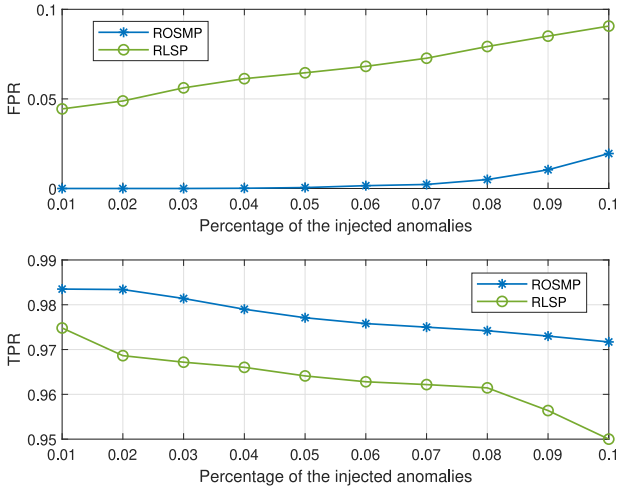


Fig. 8. Comparison of anomaly detection accuracy with 20 percent missing data and different percentages of injected anomalies.

TABLE 3
Comparison of Prediction Efficiency

Model	Memory usage	Time consumption
KNN	8.4MB	3.9s
Kriging-RNN	92.7MB	277.5s
RLSP	76.2MB	170.3s
ROSMP	**18.1MB**	**10.2s**

97 percent. The result explains why our ROSMP scheme is robust to anomalies and can be used to predict the spectrum map with corrupted measurements.

6.3 Prediction Efficiency Analysis

Table 3 presents the memory usage and the time consumption required to complete a spectrum map prediction of different algorithms. We evaluate the case where $p_{anomaly} = 10\%$ and $p_{miss} = 20\%$. Compared with RLSP and Kriging-RNN, the computation time and memory usage of our proposed ROSMP algorithm is largely reduced by updating the unknown variables to obtain their values at T from values at $T-1$, rather than recomputing new ones each time by processing all the past measurements together. While comparative computation time and memory usage are required by our ROSMP algorithm and the KNN model to enable spectrum map prediction, our ROSMP achieves much better prediction accuracy than the KNN (as indicated in Figs. 6 and 7). The result shows the higher efficiency of our ROSMP scheme in predicting the spectrum map compared with the baselines.

7 Conclusion

In this paper, we propose a robust online spectrum map prediction (ROSMP) algorithm to address the existing challenges associated with the spectrum map. First of all, to satisfy the requirements to access to accurate and valid spectrum data in dynamically changing environments, spectrum map prediction is investigated by combining the spectrum map with prediction functionalities. Then, to address the challenging problem of spectrum map prediction from incomplete and corrupted historical observations, we model the observations as a spectrum tensor by effectively integrating the time series forecasting techniques, and exploit tensor completion and subspace learning to fill in the missing values and separate anomalies in the process of spectrum map prediction. Moreover, to reduce the computation cost for tensor completion, an alternating direction minimization procedure is developed to efficiently solve the optimization problem in an online manner. We have conducted extensive experiments using a real-world spectrum measurement dataset. The results demonstrate that our proposed ROSMP algorithm leads to a superior performance than the state-of-the-art spectrum map prediction algorithms in terms of both prediction accuracy and computation efficiency in the presence of missing data and anomalies.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 61771126, in part by the Key Research and Development Plan of Jiangsu Province under Grant BE2018108, and in part by the U.S. NSF ECCS under Grants 1731238, 2030063, and CCF 2007313.

References

[1] A. Kliks *et al.*, "Beyond 5G: Big data processing for better spectrum utilization," *IEEE Veh. Technol. Mag.*, vol. 15, no. 3, pp. 40–50, Sep. 2020.
[2] M. Höyhtyä *et al.*, "Spectrum occupancy measurements: A survey and use of interference maps," *IEEE Commun. Surv. Tut.*, vol. 18, no. 4, pp. 2386–2414, Sep.-Dec. 2016.
[3] N. Barman, S. Valentin, and M. G. Martini, "Predicting link quality of wireless channel of vehicular users using street and coverage maps," in *Proc. IEEE 27th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun.*, 2016, pp. 1–6.
[4] V. Chowdappa, C. Botella, J. J. S.-Zapater, and R. J. Martinez, "Distributed radio map reconstruction for 5G automotive," *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 2, pp. 36–49, Apr. 2018.
[5] H. B. Yilmaz, T. Tugcu, F. Alagöz, and S. Bayhan, "Radio environment map as enabler for practical cognitive radio networks," *IEEE Commun. Mag.*, vol. 51, no. 12, pp. 162–169, Dec. 2013.
[6] G. Ding *et al.*, "Spectrum inference in cognitive radio networks: Algorithms and applications," *IEEE Commun. Surv. Tut.*, vol. 20, no. 1, pp. 150–182, Jan.–Apr. 2018.
[7] X. Li, G. Chen, Y. Xu, X. Wang, and T. Song, "Recovering missing values from corrupted historical observations: Approaching the limit of predictability in spectrum prediction tasks," *IEEE Access*, vol. 8, pp. 180379–180393, Sep. 2020.
[8] J. J. Meng, W. Yin, H. Li, E. Hossain, and Z. Han, "Collaborative spectrum sensing from sparse observations in cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 2, pp. 327–337, Feb. 2011.
[9] L. Zhang, G. Ding, Q. Wu, Y. Zou, Z. Han, and J. Wang, "Byzantine attack and defense in cognitive radio networks: A survey," *IEEE Commun. Surv. Tut.*, vol. 17, no. 3, pp. 1342–1363, Sep.–Dec. 2015.
[10] S. Debroy, S. Bhattacharjee, and M. Chatterjee, "Spectrum map and its application in resource management in cognitive radio networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 1, no. 4, pp. 406–419, Dec. 2015.
[11] S. Kim, E. Dall'Anese, and G. B. Giannakis, "Cooperative spectrum sensing for cognitive radios using Kriged Kalman filtering," *IEEE J. Sel. Top. Signal Process.*, vol. 5, no. 1, pp. 24–36, Feb. 2011.
[12] Y. Hu and R. Zhang, "A spatiotemporal approach for secure crowdsourced radio environment map construction," *IEEE/ACM Trans. Netw.*, vol. 28, no. 4, pp. 1790–1803, Aug. 2020.
[13] Y. Teganya and D. Romero, "Data-driven spectrum cartography via deep completion autoencoders," in *Proc. IEEE Int. Conf. Commun.*, 2020, pp. 1–7.
[14] M. Tang, G. Ding, Z. Xue, J. Zhang, and H. Zhou, "Multi-dimensional spectrum map construction: A tensor perspective," in *Proc. 8th Int. Conf. Wirel. Commun. Signal Process.*, 2016, pp. 1–5.
[15] M. Tang, G. Ding, Q. Wu, Z. Xue, and T. A. Tsiftsis, "A joint tensor completion and prediction scheme for multi-dimensional spectrum map construction," *IEEE Access*, vol. 4, pp. 8044–8052, Nov. 2016.

[16] K. Amasyali and N. EI-Gohary, "A review of data-driven building energy consumption prediction studies," *Renewable Sustain. Energy Rev.*, vol. 81, pp. 1192–1205, 2018.
[17] A. Mosavi, P. Ozturk, and K. Chau, "Flood prediction using machine learning models: Literature review," *Water*, vol. 10, no. 11, 2018, Art. no. 1536.
[18] J. Beneciuk, M. Bishop, and S. George, "Clinical prediction rules for physical therapy interventions: A systematic review," *Phys. Ther.*, vol. 89, no. 2, pp. 114–124, 2009.
[19] X. Xing, T. Jing, and W. Cheng, "Spectrum prediction in cognitive radio networks," *IEEE Wirel. Commun.*, vol. 20, no. 2, pp. 90–96, Apr. 2013.
[20] G. Ding *et al.*, "Robust online spectrum prediction with incomplete and corrupted historical observations," *IEEE Trans. Veh. Technol.*, vol. 66, no. 9, pp. 8022–8036, Sep. 2017.
[21] C. Ge, Z. Wang, and X. Zhang, "Robust long-term spectrum prediction with missing values and sparse anomalies," *IEEE Access*, vol. 7, pp. 16655–16664, Jan. 2019.
[22] A. Agarwal and R. Gangopadhyay, "Predictive spectrum occupancy probability-based spatio-temporal dynamic channel allocation map for future cognitive wireless networks," *Trans. Emerg. Telecommun. Technol.*, vol. 29, no. 8, pp. 1–18, 2018, Art. no. e3442.
[23] X. Li, Z. Liu, G. Chen, Y. Xu, and T. Song, "Deep learning for spectrum prediction from spatial-temporal-spectral data," *IEEE Commun. Lett.*, vol. 25, no. 4, pp. 1216–1220, Apr. 2021.
[24] J. Liao, J. Tang, and W. Zeng, "Efficient and accurate traffic flow prediction via incremental tensor completion," *IEEE Access*, vol. 6, pp. 36897–36905, Jun. 2018.
[25] K. Xie, X. Li, and X. Wang, "Fast tensor factorization for accurate internet anomaly detection," *IEEE/ACM Trans. Netw.*, vol. 25, no. 6, pp. 3794–3807, Dec . 2018.
[26] J. Bengua, H. Phien, and H. Tuan, "Efficient tensor completion for color image and video recovery: Low-rank tensor train," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2466–2479, May 2017.
[27] I. Alqerm and B. Shihada, "Sophisticated online learning scheme for green resource allocation in 5G heterogeneous cloud radio access networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 10, pp. 2423–2437, Oct. 2018.
[28] H. Kasai, W. Kellerer, and M. Kleinsteuber, "Network volume anomaly detection and identification in large-scale networks based on online time-structured traffic tensor tracking," *IEEE Trans. Netw. Serv. Manage.*, vol. 13, no. 3, pp. 636–650, Sep. 2016.
[29] Z. Fang, X. Yang, and L. Han, "A sequentially truncated higher order singular value decomposition-based algorithm for tensor completion," *IEEE Trans. Cybern.*, vol. 49, no. 5, pp. 1956–1967, May 2019.
[30] Q. Song, H. Ge, J. Caverlee, and X. Hu, "Tensor completion algorithms in big data analytics," *ACM Trans. Knowl. Discov. Data*, vol. 13, no. 1, 2019, Art. no. 6.
[31] K. Xie, L. Wang, and X. Wang, "Accurate recovery of internet traffic data: A sequential tensor completion approach," *IEEE/ACM Trans. Netw.*, vol. 26, no. 2, pp. 793–806, Apr. 2018.
[32] K. Xie, X. Li, X. Wang, G. Xie, J. Wen, and D. Zhang, "Graph based tensor recovery for accurate internet anomaly detection," in *Proc. IEEE Conf. Comput. Commun.*, 2018, pp. 1502–1510.
[33] S. Rajendran, R. Calvopalomino, and M. Fuchs, "Electrosense: Open and big spectrum data," *IEEE Commun. Mag.*, vol. 56, no. 1, pp. 210–217, Jan. 2018.
[34] K. Pearson, "Notes on regression and inheritance in the case of two parents," *Proc. Roy. Soc. London Ser. I*, vol. 58, pp. 240–242, 1895.
[35] Y. Chen, L. Qiu, and Y. Zhang, "Robust network compressive sensing," in *Proc. 20th Annu. Int. Conf. Mobile Comput. Netw.*, 2014, pp. 545–556.
[36] G. Golub and C. Reinsch, "Handbook series linear algebra singular value decomposition and least squares solution," *Numerische Mathematik*, vol. 14, pp. 403–420, 1970.
[37] J. Sun, L. Shen, G. Ding, R. Li, and Q. Wu, "Predictability analysis of spectrum state evolution: Performance bounds and real-world data analytics," *IEEE Access*, vol. 5, pp. 22760–22774, Oct. 2017.
[38] K. Baddour, A. Ghasemi, and H. Rutagemwa, "Spectrum occupancy prediction for land mobile radio bands using a recommender system," in *Proc. 88th Veh. Technol. Conf.*, 2018, pp. 1–6.
[39] R. Nau, "Review of basic statistics and the simplest forecasting model: The sample mean," 2014. [Online]. Available: http:// people_ duke_ edu/_ rnau/review_ of_ basic_ statistics_ and_ the_ mean_ model_ for_ forecasting–robert_ nau.pdf
[40] N. Ahmed, A. Atiya, N. ElGayar, and H. El-Shishiny , "An empirical comparison of machine learning models for time series forecasting," *Econometric Rev.*, vol. 29, pp. 594–621, 2010.
[41] S. Taylor and L. Benjamin, "Forecasting at scale," *Amer. Statistician*, vol. 72, pp. 37–45, 2018.
[42] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy Stat. Soc. Ser. B*, vol. 58, no. 1, pp. 267–288, 1996.
[43] A. Ashok, M. Govindarasu, and V. Ajjarapu, "Online detection of stealthy false data injection attacks in power system state estimation," *IEEE Trans. Smart Grid*, vol. 9, no. 3, pp. 1636–1646, May 2018.

Xi Li (Student Member, IEEE) received the BE degree from the School of Information Science and Engineering, Southeast University, China, where she is currently working toward the PhD degree. From November 2018 to October 2019 she was a visiting student with the Department of Electrical and Computer Engineering, Stony Brook University, Stony Brook, NY, USA. Her research interests include cognitive radio networks, vehicular ad hoc networks, and application of machine learning to wireless communication.

Xin Wang (Member, IEEE) received the BS degree in telecommunications engineering, the MS degree in wireless communications engineering from the Beijing University of Posts and Telecommunications, Beijing, China. and the PhD degree in electrical and computer engineering from Columbia University, New York, NY. She is currently an associate professor at the Department of Electrical and Computer Engineering, State University of New York, Stony Brook, NY. Before joining Stony Brook, she was a member of Technical Staff in the area of mobile and wireless networking with Bell Labs Research, Lucent Technologies, NJ. Her research interests include sparse signal processing, machine learning, and Big Data analysis. She was on the executive committee and a technical committee of numerous conferences and funding review panels. She was the recipient of the NSF Career Award in 2005 and the ONR Challenge Award in 2010. She is currently an associate editor for the *IEEE Transactions on Mobile Computing*.

Tiecheng Song (Member, IEEE) received the BS and MS degrees from the Department of Radio Engineering, Southeast University, Nanjing, China, in 1989 and 1992, respectively, and the PhD degree from the School of Information Science and Engineering, Southeast University, in 2006. In 1992, he joined as a research associate with the Department of Radio Engineering, Southeast University, where he is currently a professor at the National Mobile Communication Research Laboratory, School of Information Science and Engineering. He was the executive vice president of the Nanjing Institute of Communication Technology. His research interests include 5G wireless systems, optical wireless communication technologies, and cognitive radio.

Jing Hu (Member, IEEE) was born in 1975. She received the PhD degree from Southeast University in 2011. She is currently an associate professor and the master instructor at the National Mobile Communication Research Lab, Southeast University. Her research interests include cognitive radio, wireless sensor networks, and vehicle networks.