IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received December 9, 2018, accepted January 11, 2019, date of publication January 23, 2019, date of current version February 8, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2894784

Spectrum Occupancy Reconstruction in Distributed Cognitive Radio Networks Using Deep Learning

MDUDUZI C. HLOPHE, (Student Member, IEEE), AND SUNIL B. T. MAHARAJ
Department of Electrical, Electronic and Computer Engineering, University of Pretoria, Pretoria 0028, South Africa
Corresponding author: Mduduzi C. Hlophe (u16250444@tuks.co.za)

This work was supported in part by the Association of Commonwealth Universities under Grant FE-2015-26, and in part by the Sentech Chair in Broadband Wireless Multimedia Communications.

ABSTRACT Spectrum occupancy reconstruction is an important issue often encountered in collaborative spectrum sensing in distributed cognitive radio networks (CRNs). This issue arises when the spectrum sensing data that are collaborated by secondary users have gaps of missing entries. Many data imputation techniques, such as matrix completion techniques, have shown great promise in dealing with missing spectrum sensing observations by reconstructing the spectrum occupancy data matrix. However, matrix completion approaches achieve lower reconstruction resolution due to the use of standard singular value decomposition (SVD), which is designed for more general matrices. In this paper, we consider the problem of spectrum occupancy reconstruction where the spectrum sensing results across the CRN are represented as a plenary grid on a Markov random field. We formulate the problem as a magnetic excitation state recovery problem, and the stochastic gradient descent (SGD) method is applied to solve the matrix factorization. SGD is able to learn and impute the missing values with a low reconstruction error compared with SVD. The graphical and numerical results show that the SGD algorithm competes favorably SVD in the matrix factorization by taking advantage of correlations in multiple dimensions.

INDEX TERMS Cognitive radio networks, Ising model, matrix factorization, Metropolis-Hastings algorithm, missing values, stochastic gradient descent.

I. INTRODUCTION AND BACKGROUND

The distributed spectrum sensing technique in CRNs was introduced in order to have a complete and reliable spectrum sensing data set that is needed for reliable resource allocation. But spectrum occupancy measurements for all the frequency channels might not be available at all times partially due to energy limitation and network failures [1]. The new motivation is to recover the whole spectrum sensing data matrix from a sampling of its entries. As a motivating example, consider collaborative spectrum sensing where individual SUs collaborate their spectrum sensing results with neighbors. This is expected to improve the ability of ascertaining the time granularity of complete spectrum usage states. Spectrum sensing techniques are discussed extensively by Aneja *et al.* [2].

In distributed spectrum sensing, each SU receives and has to process spectrum occupancy data from other SUs in order to obtain accurate channel occupancy information. However, due to issues such as: (i) bad reporting channel conditions caused by wireless channel fading, the detection performance of SUs and local communications become unreliable [3]; (ii) energy efficient spectrum sensing techniques whereby SUs that have no information to transmit in the next time slot may decide not to share their spectrum sensing results. Due to these issues, the quality of spectrum sensing information obtained from other SUs is far from sufficiently and directly assuring the occupied/unoccupied channels. If not handled well, these issues may lead to spectrum sensing errors that may interrupt primary user (PU) activities and reduce SU spectral efficiency due to wrong statistical inferences. Therefore, in order to circumvent this problem, SUs need to perform some data mining into the spectrum sensing data, impute the missing data and reconstruct the spectrum sensing data matrix and obtain accurate spectrum occupancy information. In this way, qualitative and precise information about the current channel occupancy will be obtained, which will lead to accurate inference on the current user context and channel states.

Many spectrum occupancy reconstruction studies have been conducted in both centralized and distributed collaboration approaches. However, their practical applicability depends on how much available information each SU can obtain about the spectrum. In centralized collaborative spectrum sensing, multiple SUs individually observe the spectrum band concerned and then send their sensing results to the fusion center (FC). This is an infrastructure based approach where spectrum sensing results are sent to the FC where they are processed and analyzed. The FC derives a global decision regarding the status of the sensed channels and takes a single decision that is relayed back to SUs [4], [5].

In the centralized approach, it is not difficult and tricky to deal with the missing value problem and reconstruct the spectrum occupancy data matrix. This is due to the existence of the FC which can perform missing value imputation by leveraging the low-rank nature of the spectrum occupancy data matrix using general methods such as matrix completion. Matrix completion is method that is defined as a low-rank matrix completion technique that utilizes SVD for learning and imputing missing values under the low-rank constraint [6]. For example, Meng *et al.* [7], used the matrix completion method to perform imputation of missing values in the reconstruction of accurate channel occupancy results. In order to model incomplete measurements in their work, the authors allowed some SU collaboration transmissions to fail while others are allowed to be successful. In this case, the FC receives channel occupancy results from only a subset of the total number of SU which just a percentage of all the spectrum sensing results. The reconstructed data matrix, the occupied channels together with their fading values are obtained by solving a joint-sparsity reconstruction problem using matrix completion. Candes and Plan [8] also applied a similar matrix completion technique for a channel occupancy task where the spectrum sensing data matrix was correlated and low-rank. Numerous other research works have been published in centralized spectrum occupancy reconstruction and some fine results have been achieved, including the OR FC rule in a distributed manner by Ghasemi and Sousa [9], compressive sensing by Tian [10], and multi-dimensional correlation by Xue *et al.* [11].

In the distributed spectrum sensing approach where SUs perform individual spectrum sensing and share their observation among themselves, it is particularly difficult to deal with the missing value problem. The distributed spectrum sensing approach is particularly useful in wide area ad-hoc networks where the PU transmission energy can only be detected by only a subset of SUs, Hlophe *et al.* [12]. The idea is, by knowing the spectrum occupancy at one location and sharing it with nearest neighbors, remote SUs also get that information and are able to dynamically reuse any idle channels. However, as opposed to the centralized approach, here SUs exchange their spectrum sensing results in order to construct individual views of spectrum occupancy. This makes the case of missing value imputation and spectrum occupancy reconstruction a little bit tricky to perform. missing value imputation, thus spectrum occupancy reconstruction is difficult since it requires high computational capability within each SU. There are, however, numerous works that have tried to alleviate the computational burden that comes with each SU acting as an FC. For example, the study by Song *et al.* [13] where a novel support fusion-based distributed compressive spectrum sensing was proposed. In this technique, a local compressed reconstruction and adaptive learning of support knowledge among SUs was proposed. During each spectrum occupancy reconstruction iteration, each SU reconstructs the local sparse spectrum occupancy through a truncated l_1 minimization technique and incorporating support information obtained from previous iterations. Each SU then obtains a local support detection via the thresholding of its local reconstruction before exchanging the information with its nearest neighbors. After several rounds it also receives fused support information from other SUs. In this way, the spatial diversity and reliable spectrum occupancy information is obtained and all SUs can update their support information for use in the next iteration.

The problem of distributed spectrum occupancy reconstruction has also been approached from the perspective of empirical measurements. Several studies have been reported, for example; Riihijarvi *et al.* [36] modeled the problem by characterizing spectrum maps with spatial statistics and random fields. The mathematical premises presented in their work outline, in a way, generating useful statistics from real measurement data. Wellens *et al.* [15] also modeled spectrum occupancy using spatial statistics where they used random fields and a semi-variogram to describe the spatial correlation of spectrum usage. They extracted parameters from extensive real-life measurements for multiple wireless technologies which gave useful insights on the extraction of useful spectrum sensing parameters. Li [16] and Li *et al.* [17] also studied simulation on distributed reconstruction using random fields where the entire CRN was modeled as a plenary grid. In their work, both one-dimensional and two-dimensional Ising models were applied in the modeling of joint distributions of spectrum occupancies, respectively. The phase transition in network connectivity was studied using the random field model and the spectrum occupancy reconstruction was performed using the matrix completion technique. The numerical results obtained indicated the effectiveness of the Ising model in predicting connectivity, average delay and jitter. Potier *et al.* [18] also studied the spectrum situation reconstruction problem where the spectrum occupancy problem was viewed as an image recovery problem which was solved using total variation inpainting. This method takes the advantage of correlations in multiple dimensions (2-D, 3-D) and proves prowess over belief propagation method which only focuses on one dimension (i.e., 2-D).

All the above mentioned studies evaluate spectrum occupancy and spectrum occupancy reconstruction by utilizing either probabilistic or statistical techniques which are limited to the assumptions required in their derivations. For example, one has to determine whether the value is random variable or a

random process in order decide either to use probabilistic or statistical approaches. This happens while machine learning (ML) has received increasing attention as a powerful tool in solving such problems. ML approaches with their heuristic background do not require any assumptions on the data or even any prerequisite information about the problem at hand usually provide better accuracy that the probabilistic and statistical approaches. There are very few works ML works on spectrum occupancy, mostly on the spectrum occupancy variation and traffic load prediction such as [19]. Deep Learning (DL), which is also known as Deep Neural Learning (DNL) and sometimes as deep neural network (DNN), is a subset of Machine Learning (ML), which is also a subset of artificial intelligence (AI) [20]. DL is an unsupervised ML technique which consists of networks that are capable of learning unsupervised from unstructured data as well as unlabeled data by utilizing a hierarchical level of artificial neural networks to carry out task specific ML processes.

DL has been studied for several decades, however, there were only a few practical applications in communication systems until a few years back when the field changed. The renaissance is due to the increased ownership of smart gadgets and an increased demand for network services. This increased the size of data sets that needs to be processed by the base stations as well as the gadgets themselves. The application of DL techniques in communication systems has been a hive of research, more especially in CRNs. A few examples include PU classification where Cui *et al.* [21] applied DL through the use of deep belief networks in order to improve the accuracy rate of user agents' recognition. Interesting results were obtained where it was shown that the deep belief network model provides a PU agents' classification. This result is the foundation of the prediction of both idle frequency spectrum and time slots which has attracted huge interest in CRN research. Another example is based on PU modulation classification which is another way of getting closer to knowing the technology used by PUs. Here, Mendis *et al.* [22] applied DBNs for pattern recognition and classification for automated modulated classification and achieved high accuracy in modulation detection and classification in the presence of environmental noise. The results that they obtained illustrate high efficiency in classifying 4 Frequency Shift Keying ($4FSK$), 16 Quadrature Amplitude Modulation ($16QAM$), Binary Phase Shift Keying ($BPSK$), Quadrature Phase Shift Keying ($QPSK$) techniques in various environments. Also, Paisana *et al.* [23] utilized spectrum sensing, DL and dynamic optimization to design a high throughput SU system that can coexist with a PU with minimal interference by detecting its channel usage pattern. Their method involved training a Deep Convolutional Neural Network (DCNN) model to perform the classification directly from spectrograms, where the detection of PU scenarios was treated as an image classification task that was solved using CNNs to obtain a calibration table for the SU transmitter.

The above literature demonstrates that ML algorithms perform significantly better in enhancing the standard probabilistic or statistical algorithms in CRNs. The designing of new adaptive methods that enhance the standard signal processing seems to be an encouraging area to be investigated for providing fast and accurate spectrum sensing and resource allocation techniques. In this paper, we are motivated by the new advances in knowledge discovery and inference in CRNs. Our aim is to provide a new way of knowledge discovery from few samples of information in distributed spectrum sensing using computational deep learning tools.

A. MOTIVATION

New mathematical models and methods that can address the dynamic behavior of components of CR systems are required. These are specifically new methodologies that need to provide (i) accurate link physics models in order to precisely model and characterize the set of link states in the CRN; (ii) distributed and low complexity algorithms for dynamically optimizing CRN performance by providing it with the required computational capability. We have been motivated by the recent advances in spectrum occupancy reconstruction, where matrix completion-based methods indicate that it is possible to reconstruct the entire spectrum sensing data matrix from a few observations [24]. It has been shown that if the matrix to be reconstructed has a low-rank and satisfies the incoherence property, it can be reconstructed exactly with high probability from just a few entries [25]. However, we observe that matrix completion approaches achieve lower reconstruction resolution due to the use of standard SVD which is designed for more general matrices. Therefore, instead of using the standard SVD algorithms for matrix factorization, stochastic matrix factorization techniques such as SGD can be exploited. The basic idea is to impute the observations that have not been received due to reasons such as: (i) imperfect reporting channel [26], (ii) energy saving cooperative spectrum sensing [27], (iii) spectrum sensing data falsification attacks [28], which prevent spectrum sensing data from reaching its destination.

B. SUMMARY OF CONTRIBUTIONS

In this paper, we make two contributions to the spectrum occupancy reconstruction problem. They are discussed as follows:

- We introduce an energy model for the characterization of communication links between SUs in MRFs. This link energy function is founded on the canonical Ising model and is used to study the phase transitions in SU observations using the electron spin model from quantum physics [29]. The electron spin model is used to model the energy stored within an SU regarding the $[1, 0]$ or ON/OFF channel status. Since communication links need to be available for SUs to collaborate their spectrum sensing results, we have developed a mathematical framework in which they can be modeled as physical systems. Thus, the network connectivity and SU interactions are founded on molecular models and are characterized by the potential energy stored in the

bonds forming a molecule. We create initial observations of spectrum occupancy for the spectrum sensing data matrix and using the Metropolis-Hastings algorithm. We sample spectrum observations using the Gibbs bivariate exponential distribution and create a spectrum occupancy data matrix with missing values by allowing some SU transmissions to fail while others succeed.

- We explore the applicability of the SGD algorithm for solving a matrix factorization problem in spectrum occupancy reconstruction. The proposed SGD algorithm resolves the problem of falling into the local minimum by scaling the spectrum occupancy data matrix in small ranks and deals with complexity by selecting instances randomly. This approach is similar to computational deep learning (CDL) due to its capability to allow for the addition of depth into the spectrum occupancy data matrix which is a very important step towards discovering the missing matrix entries. Thus, we utilize SGD to improve the accuracy of the missing value imputation and the convergence speed of spectrum occupancy reconstruction. We observe that the SGD algorithm performs better than traditional SVD algorithm due to the fact that as the degree of correlation between SUs drop, the performance of SVD degrades.

The rest of the paper is organized as follows: In Section **II** we introduce the CRN model showing a random deployment of PUs and SUs. We also model the CRN topology using MRFs. In Section **III**, we define the spectrum occupancy data matrix as a matrix factorization problem with assumptions that conform to CRNs. In Section **IV**, we discuss the proposed solution by first setting up the CRN on an MRF using the Ising model and Gibbs distributions. We sample the spectrum occupancy data using Gibbs distribution and the Metropolis-Hastings algorithm, and then perform missing data imputation by solving the matrix factorization problem using SGD. In Section **V**, we perform simulations and obtain graphic and numeric performance measures for the proposed algorithm and compare it with the SVD algorithm in terms of the mean square error (MSE) for both the computational cost and convergence performance. Section **VI** gives concluding remarks on the performance of our proposed algorithm.

II. PROPOSED SYSTEM MODEL

A. NETWORK MODEL

Consider a CRN where M SUs coexist with three PUs where both the PUs and SUs share the same base station (i.e., the PUBS) as shown in Fig. 1. We assume that SUs form a two-dimensional Poisson point process (PPP) and PUs interrupts a small area during their transmission in order to allow for the use of MRFs. SUs perform individual spectrum sensing using energy detection discussed in [12] and share their spectrum sensing results with neighboring SUs. Both PUs and SUs are assumed to use a time slotted system such that spectrum occupancy results are shared by SUs in every time slot.

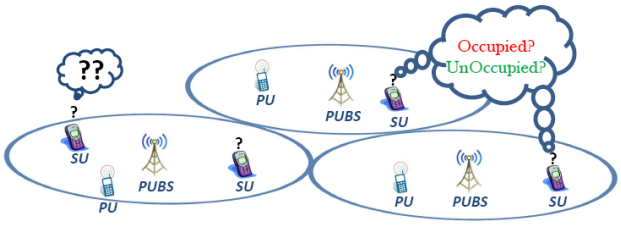


FIGURE 1. Illustration of a CRN showing a random deployment of SUs and PUs.

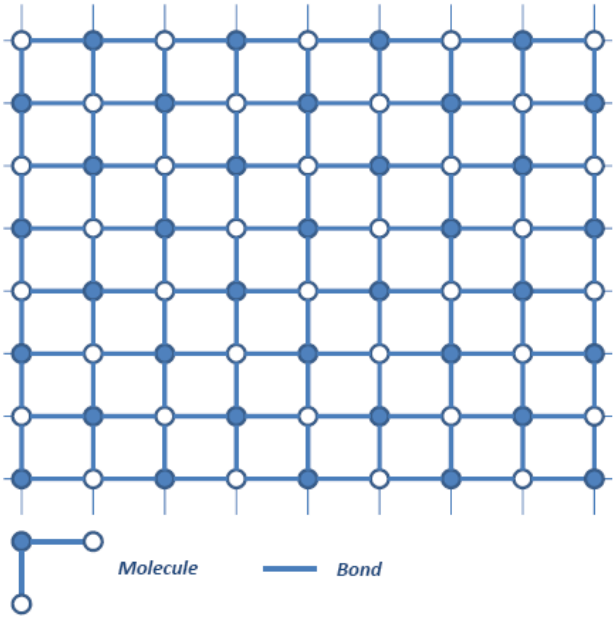


FIGURE 2. Illustration of a CRN topology on an $1500m \times 1500m$ area, SUs, neighbors and their cliques, where an SU-SU interaction is treated as a molecule.

B. CRN TOPOLOGY ON MRFS

We illustrate the CRN topology using the plenary grid given in Fig. 2, where the distribution of SUs in the CRN can be factorized into a cluster within a $1500m \times 1500m$ geographical area. This area is further divided into 50×50 subareas such that each grid is a 30×30 square area. Tagged SUs (solid circles) and their neighbors (open circles) show 2-point cliques represented by the edges connecting them. This set up completes the MRF model and the whole CRN bears resemblance of a magnetic lattice. These prominent features of the CRN topology can be realistically captured by the MRF model. Thus, we model the spatial correlation of the spectrum occupancy using the classical Ising model [31], where each grid point possesses either a spin-down ($\downarrow$) or a spin-up ($\uparrow$) representing the binary logic states 0 - idle and 1- busy, respectively. Assuming that each SU can only detect one channel within its location at any given time slot, we let each grid to only have two states; ($\downarrow$) or ($\uparrow$). Each SU in each grid represents spectrum occupancy at a specific time, and

the series of grid points detail the spatial-temporal spectrum usage in CRNs. However, we need to keep in mind that in CRNs, the first-order Markov property that comes with the MRF and Ising model hold rigorously when the PU interruption ranges are small; as the interruption range increases it may no longer be valid. Thus, each PU interrupts a circular range with a radius of 500m, within which the channels used by PUs are not available to SUs.

C. SU COMMUNICATION AND PAIRWISE INTERACTIONS

When SUs communicate and exchange their spectrum sensing results, they are exchanging electromagnetic energy. Therefore, the spatial distribution of this electromagnetic energy used for these communications and its temporal variations are used to model the flow of energy between neighboring SUs. This analogy is shown in Fig. 2 where the whole CRN is characterized as a magnetic lattice. The small circles indicate the positions of SUs in the CRN (similar to lattice points on the surface on a square ferromagnet), whereby each SU has four nearest neighbors on the four immediately adjacent grids and their interactions are represented as edges called 2-point cliques. The idea is that SUs collaborate their spectrum sensing results with neighbors such that their pairwise interactions can be realized using as an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$. Here, $\mathcal{V}$ represents the set of all vertices where $v \in \mathcal{V}$ represent the locations of the SU nodes; $\mathcal{E}$ represents the set of all edges where $e \in \mathcal{E}$ defines pairwise interactions between SUs thus forming social connections that define their collaboration paths. Together, the overall collection of $v \in \mathcal{V}$ and $e \in \mathcal{E}$ results in an undirected graph.

III. PROBLEM FORMULATION

In this section, we assume that all SUs in the CRN are the honest with their observations. After performing individual spectrum sensing, SUs collaborate their spectrum sensing results such that each SU acts as an FC in its own right such that the spectrum sensing fusion happens in a completely distributed manner. This means that each SU constructs its own spectrum sensing data matrix.

A. THE SPECTRUM SENSING DATA MATRIX

Consider a probability distribution on $\mathbb{X} = \{0, 1\}^{\mathcal{V}}$ where the variables 0 and 1 have been explained in the previous section. The random variable X stand for spectrum occupancy observation of each SU located at the vertex $v \in \mathcal{V}$ of the graph $\mathcal{G}$. Let us consider the system to be composed of K sensing slots such that the spectrum sensing data is represented by a random process $X^{M\times K}$, given as

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} & \cdots & X_{1,K} \\ X_{2,1} & X_{2,2} & \cdots & X_{2,K} \\ \vdots & \vdots & \ddots & \vdots \\ X_{M,1} & X_{M,2} & \cdots & X_{M,K} \end{bmatrix}, \tag{1}$$

which is the spectrum sensing data matrix, where $X_{m,k} \in [0, 1]$, $m = 1, 2, \cdots, M$ and $k = 1, 2, \cdots, K$ represent the sensing reports from the m^{th} SU at sensing slot k.

B. CAUSES OF MISSING SPECTRUM SENSING RESULTS

1) REPORTING CHANNEL CONDITIONS

When the reporting channel condition is poor, the neighboring SUs might not detect the spectrum sensing data sent at time slot t. In this case, $X_{m,k}$ is dependent on the reporting channel and independent on the missing or observed values, therefore $X_{m,k} = 0$. Thus, the missingness mechanism is missing completely at random (MCAR).

2) COLLABORATIVE SPECTRUM SENSING SCHEMES

If some SUs adopt energy-saving spectrum sensing schemes, the spectrum sensing data may also be incomplete. This happens when certain SUs decide not to report their spectrum sensing results if they do not have any information to transmit in the upcoming time slot. In this case, the value of $X_{m,k}$ depends on the transmission probability of the SUs, not on the values of the spectrum sensing matrix. The missing values are thus dependent on the value of $X_{m,k} = 0$, and the missingness mechanism is missing not at random (MNAR). This means that in the case of MNAR, the missing values are known and they can just be imputed by 0's. Therefore, the problem now remains with the MCAR mechanism which is now turning to be the focus of this paper. Thus, we now focus on the preprocessing of spectrum sensing data that is MCAR. However, according to [34], MCAR is stronger that missing at random (MAR) because the missingness of the data is still random but it is entirely dependent on the observed variables. Therefore, the key is that the missingness is not due to the values that are not observed. Therefore, we conclude that MCAR implies MAR, but not the other way round. Therefore, in order to depict the missingness of the spectrum sensing data matrix $\mathbf{X}$, we rewrite (1) as

$$X = \begin{bmatrix} X_{1,1} & X_{1,2} & X_{1,3} & \cdots & X_{1,K} \\ X_{2,1} & X_{2,2} & \emptyset & \cdots & X_{2,K} \\ X_{3,1} & \emptyset & X_{3,3} & \cdots & X_{3,K} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ X_{M,1} & \emptyset & X_{M,3} & \cdots & X_{M,K} \end{bmatrix}. \tag{2}$$

Thus, in the spectrum sensing data matrix in (2), there are missing values at positions $X_{3,2}$, $X_{2,3}$ and $X_{M,2}$ represented by $\emptyset$.

C. IMPUTATION OF INCOMPLETE SPECTRUM SENSING DATA

Missing value imputation is a process of plugging in plausible values in a data set where none exists [32]. The objective of imputation is not just to find out what the missing value is, but to restore the important characteristics of the data set as a whole by estimating the missing values. For example, there is random value imputation and majority rule imputation where the missing data $X_{m,k}$ in time slot k is assigned the value that

is reported by most SUs in that time slot. Here, in order to utilize the additional information, a maximum a posteriori (MAP) criterion is considered as an imputation algorithm. The spectrum sensing reports of SUs in a given time slot are dependent on each other. This is because the set of spectrum sensing data is generated based on the same truth (i.e. the status of the PU).

We thus introduce matrix factorization as a technique for approximating a matrix $\mathbf{X}$ by the product of two smaller matrices $\mathbf{Y}$ and $\mathbf{Z}$ such that $\mathbf{X}$ can be reconstructed from the two smaller matrices. This is given as

$$\boldsymbol{X} = \boldsymbol{YZ}, \tag{3}$$

where $\mathbf{Y} \in \mathbb{R}^{m\times r}$ and $\mathbf{Z} \in \mathbb{R}^{r\times k}$. Matrix $\mathbf{Y}$ is a component matrix whose columns $\{\boldsymbol{Y}_{:,k}\}_{k=1}^{r}$ are vertices in a hypercube $[0, 1]^m$ and $\mathbf{Z}$ is a matrix of coefficients whose distribution does not depend on the observed or on the missing data. The illustration of this technique is presented in Fig. 3:

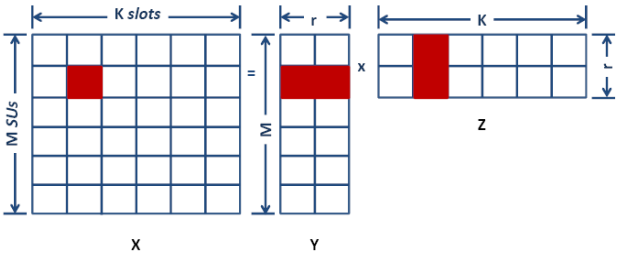


FIGURE 3. Illustration of matrix factorization.

In the context of spectrum sensing data, the matrix $\mathbf{X}$ represents the partially observed spectrum sensing data matrix; $\mathbf{Y} \in \mathbb{R}^{M\times r}$ is the matrix where each row m is a vector that contains r latent factors where $r \ll M, r \ll K$ that describe SU m. $\mathbf{Z} \in \mathbb{R}^{r\times K}$ is the matrix that contains the r factors that describe k. Therefore, matrices $\mathbf{Y}$ and $\mathbf{Z}$ are latent matrices and can be learned by optimizing an objective function given a criteria such as the root mean square error (RMSE), given in (18). Therefore, by factorizing $\mathbf{X}$ into a product $\mathbf{YZ}$ will result to the approximation, which is a problem of finding

$$\begin{aligned} \boldsymbol{Y} &\in \{0, 1\}^{m\times r}, \\ \boldsymbol{Z} &\in \mathbb{R}^{r\times k}, \\ \boldsymbol{Z}^T e_r &= e_k, \\ r &\ll \min\{m, k\}, \end{aligned} \tag{4}$$

where the optimal rank assumption r implies that the rank constraint correlates the known entries with the unknown ones; $\mathbf{Z}e_r = e_n$ entails that the columns of $\mathbf{X}$ are affine instead of being linear combinations of columns of $\mathbf{Y}$, or simply the signal direction. However, in our approach, this is a weak constraint, which is imposed only for presentation purposes in order to avoid the origin from being treated differently from the other vertices of $[0, 1]^m$. This is because if there are zero vectors, they could be dropped from $\mathbf{Y}$, leaving the factorization unchanged. Therefore, the optimization problem can thus be cast as a fixed-rank optimization problem as follows:

$$\hat{\boldsymbol{X}} = \min_{\boldsymbol{X}\in\mathbb{R}^{m\times k}} ||\boldsymbol{X} - \boldsymbol{YZ}||, \tag{5}$$

subject to

$$rank(\boldsymbol{X}) = r, \tag{6}$$

where rank r is from the assumption of the optimal rank, which has to be known a priori.

IV. SOLUTION FORMULATION

In order to allow each SU to generate spectrum observation, we employ generative modeling based on probabilistic deep learning studied in [33]. With this model, we capture the underlying data distribution as well as the mechanisms used to generate synthetic data during the data generation process. Therefore, the probability distribution of SUs in the CRN can be factorized into a cluster within 1500×1500 geographical area. This is a Bayesian network representation of a CRN which will support the following claim:

Claim: We claim that this is an efficient method for representing the independence structure of CRNs from traditional wireless networks, therefore the 1500×1500 geographical area is divided into a 50×50 grid and the existence of an SU cluster formed by a collaborating group of SUs located at each of the grid points.

A. THE ISING MODEL AND MARKOV RANDOM FIELDS

The Ising model defines a probability measure on the set of all possible configurations. We define the sample space Ω as a collection of all possible realizations of random variables, X which stand for the spectrum occupancy in the CRN. The energy function corresponding to a specific spectrum occupancy $X = \{x_0, \cdots, x_i, \cdots, x_n\} \in \Omega$ is defined using the Hamiltonian

$$\mathcal{H} = -I \sum_{u} x_u - J \sum_{u<v} A_{uv} x_u x_v, \tag{7}$$

where the first term describes the effect of the exogenous inputs and the second sum represents the endogenous effect. Each random variable x_i represents a grid in the topology and has two possible values, -1 and 1. $x_i = -1$ means busy and $x_i = 1$ means idle. I and J are two parameters of the Ising model. Intuitively, the Ising model can address the fact that the spectrum occupancy in the cognitive radio network depends on the natural (exogenous) and manufactured (endogenous) inputs. The second summation is taken over all pairs u, v of points that are neighbors. Thus, the Ising model makes the simplifying assumption that only interactions between neighboring points need to be taken into account. This is called first order Markovian property. The case $J > 0$ is called the attractive case, and the $J < 0$ is called the repulsive case. A_{uv} denotes the adjacency matrix entry that indicates the existence of an edge $e \in \mathcal{E}$ between SU_u and SU_v such that a full duplex communication between neighbors is represented by $A_{uv} = A_{vu} = 1$, and 0 otherwise.

B. MARKOV RANDOM FIELDS AND GIBBS DISTRIBUTION

In order to solve this problem, we need to model the SU observations of the ON/OFF behavior of the PU channel. Here, we assume that each SU positioned at each vertex $v \in \mathcal{V}$ of the random graph $\mathcal{G}$ will exhibit the ON/OFF behavior through spectrum sensing results $X^{m\times k}$. PU channel occupancy is based on the binary logic '1' - ON and '0' - OFF. We assume that the CRN is a 2-dimensional lattice where at each grid point we have an SU with observations '1' or '0'. Therefore, in order to model this behavior, we use the electron spin model. The electron spin model is a form of intrinsic angular momentum on a magnetic field with two orientations: up and down $\uparrow$ and $\downarrow$. Combining these magnetic orientations with the ON/OFF states' binary logic, we have a system with two states.

So, given the random variables on a MRF, it is not easy to determine the probability of any labeling. This is because figuring out what set of conditional probabilities to use on an MRF is not straightforward. Also, an arbitrary set of conditional probabilities for different sites and neighborhoods may not be mutually consistent and it is not obvious how to determine this. Therefore, the most likely labeling of an MRF is made easier by the use of Gibbs distributions. Gibbs distributions turn out to be equivalent to MRFs and are much easier to work with because the cliques capture the dependencies between neighborhoods [37].

Therefore, let define a set of sites $\{v_1, v_2, \cdots, v_m\}$ form a clique. Given a probability distribution defined for a set of sites and labels, we may say that it is a Gibbs distribution if the distribution takes the form:

$$P(X) = \frac{e^{-E(X)}}{\sum_{x\in X} e^{-E(X)}}, \tag{8}$$

where the denominator is the normalizing value required to make all probabilities sum up to 1, and $E(X)$ is the energy function given as

$$E(X) = I\sum_{u=1}^{N} x_u + J\sum_{\langle u\ v\rangle} x_u x_v. \tag{9}$$

Here, I represents the external effects and J is the energy exchange between neighboring SUs; $x(u)$ and $x(v)$ represent the configuration of the spectrum states at vertices u and v, respectively. The first summation describes the effect of exogenous inputs while the second summation is taken over all pairs (u, v) of SUs that are neighbors (nearest neighbor interactions only) such that the sum $\sum_{\langle u,v\rangle} x_u x_v$ means that the exchange energy is counted only for the neighboring SUs.

In CRNs, the first order Markovian property holds rigorously when the interruption ranges of primary users are small. As the interruption range increase, this may no longer be valid. However, some useful properties can still be quantified by the Ising model [36]. The main reason why Gibbs distributions are important in this work is that they turn out to be equivalent to MRFs. This means that for any MRF we can be able to write the probability distribution in the form of (8).

From (8), we can now define a sample space Ω for specific spectrum occupancy such that the random variable $X \in \Omega$ is assigned to a probability according to the probability distribution defined as the Gibbs distribution [37] for the 2-dimensional Ising model. Now, suppose that we can change the system from state X to X' by toggling the value of state X_v. We can use the transition function

$$\begin{aligned} T(X \to X') &= \frac{P(X')}{P(X)} \\ &= e^{-(E(X')-E(X))} \\ &= e^{-\Delta E(X)}, \end{aligned} \tag{10}$$

where $T(X \to X')$ is the probability that the system transitions from state X to X', $\Delta E(X)$ is the change in energy of the system when going from X to X'. Therefore, in order to sample spectrum occupancy data from the whole $1500m \times 1500m$ CRN, we artificially create an initial state of each SU observation of spectrum occupancy. This will be our initial CRN state over the whole area. Each SU is arbitrarily given a state from the earlier description of $\uparrow, \downarrow$. Then, combining the Gibbs distribution function $P(X)$ with the transition function $T(X)$ that will specify how much time each state $X \in [0, 1]$ lasts; then we will be able to obtain the time-series ON/OFF toggling of the occupancy states.

C. THE GIBBS DISTRIBUTION AND THE METROPOLIS-HASTINGS ALGORITHM

In the context of CRN, we utilize the Metropolis-Hastings algorithm based on two ideas: (i) To conduct a search of states through an ergodic Markov chain so that it could possibly visit every site in the CRN, not just randomly, and (ii) the Metropolis-Hastings algorithm uses the transition function $T(X \to X')$ that satisfies the relation [38]

$$P(X)T(X \to X') = P(X')T(X' \to X). \tag{11}$$

Equation (11) expresses the idea of state equilibrium in the reversible transition $X \leftrightarrow X'$. However, the initial state of the system may be far from the equilibrium that is being sought when the distribution function $P(X)$ is large. This means that during the simulation procedure, we have to go through a number of steps at first before we can start taking measurements (that is, before reaching the burn-in period). Since equilibrium is an essential part of the process, in order to apply the Metropolis algorithm, a transition function must be found that satisfies the detailed balance equation. To achieve this, we employ the canonical Ising model. We do this by supposing the initial state sampled is X^t, and later we draw a new proposal state X^* with the probability $Q(X^*|X^t)$ and then we compute a ratio α:

$$\alpha(X^t, X^*) = \frac{P(X^*)Q(X^t|X^*)}{P(X^t)Q(X^*|X^t)}. \tag{12}$$

Here, $P(X^*)/P(X^t)$ is the likelihood between the proposed sample X^* and the previous sample X^t, $Q(X^t|X^*)/Q(X^*|X^t)$ is the ratio of the proposal density in two directions. Each time

Algorithm 1 Procedure for Generating Spectrum Occupancy Using the Metropolis-Hastings Algorithm

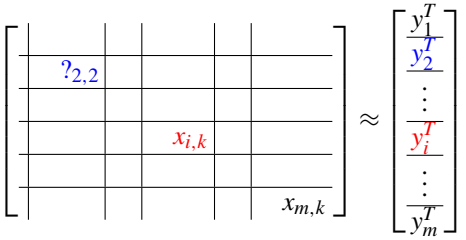
Input: M, T, I, J
Output: X
Initialization: Starting with a complete graph with un-directed edges between each pair of nodes.
Set $t = 1$
Generate an initial value u and set $X^t = u$
For $t < T$
  $t = t + 1$
  Draw a sample X^* from a proposal distribution $Q(X^*|X^t)$.
  Evaluate acceptance probability [39] $\alpha = \min\left\{1, \frac{P(X^*)Q(X^t|X^*)}{P(X^t)Q(X^*|X^t}\right\}$
  Generate u from a uniform distribution [0, 1]
  If $u \leq \alpha$ **then**
    Accept X^* as the next sample X^{t+1} with probability $\alpha(X^t|X^*)$ and keep X^t as the next sample X^{t+1} with probability $1 - \alpha(X^t|X^*)$
  Else
    Set $X^t = X^{t-1}$
  End If
  Share complete spectrum occupancy results with nearest neighbors.
End For

the new state X^{t+1} is chosen according to the following rules: when $\alpha \geq 1$, we assign X^* to X^{t+1}; when $\alpha < 1$, $X^{t+1} = X^*$ with probability α, and $X^{t+1} = X^t$ with probability $(1 - \alpha)$. Thus, once the I and J for the Ising model are specified, we can obtain the sequence of random samples from (8), which represents the spectrum occupancy during a period of time. The application of this algorithm in a CRN scenario is described in **Algorithm 1**.

D. MODELING INCOMPLETE SPECTRUM SENSING DATA

Using the Metropolis-Hastings algorithm described in the previous subsection, we are now able to sample spectrum sensing data from the whole CRN. Now, we model incomplete spectrum sensing data by allowing SUs to share their spectrum sensing results. However, we allow some transmissions to succeed and others to fail so that each SU has an incomplete spectrum sensing data matrix.

Let X to represent the spectrum occupancy results reported by SUs for the whole CRN. We divide the reporting subframe into K subdivisions such that from time t_1 state to the t_k state we have K reported observations. This is such that the spectrum sensing result reported by the i^{th} SU at time slot t_k is represented as $X_{i,k}$. Given that the observed spectrum occupancy data matrix $\mathbf{X}$ is incomplete, each SU has a task of imputing the missing values of the observation matrix (1) in a way that resembles a matrix factorization problem depicted as



$$\times \left[z_1 \middle| z_2 \middle| \cdots \middle| z_r \middle| \cdots \middle| z_k \right]. \tag{13}$$

Equation (13) shows the incomplete spectrum occupancy observation matrix $\mathbf{X}$ shown in Fig. 3 where $\mathbf{X}$ is being approximated with a product of two smaller matrices $\mathbf{Y}$ and $\mathbf{Z}$. Here, the matrix entries $\mathbf{X}$, $x_{i,k}$ and $x_{m,k}$ are MNAR, while the entry denoted by $?_{2,2}$ is of MCAR type. The way in which the imputation of the missing values of $\mathbf{X}$ can be performed is by performing a data mining technique that approximates $\mathbf{X}$ from the product of $\mathbf{YZ}$ with high resolution of correctness. Therefore, using the objective function in (5), we perform the minimization of the cost function defined by the difference $\boldsymbol{X} - \boldsymbol{YZ}$ constraining it with the matrix rank r. The matrix factorization method would be enough to solve the missing value imputation problem.

E. MISSING DATA IMPUTATION USING STOCHASTIC GRADIENT DESCENT

Stochastic gradient descent (SGD), which is also referred to as incremental *gradient descent (GD)*, as defined by Agarwal *et al.* [41] is a stochastic approximation of a gradient optimization and iterative method for minimizing an objective function that is written as a sum of differentiable functions. This is a scaled iterative procedure where we will have the incomplete spectrum occupancy data matrix $\boldsymbol{X} \in \mathbb{R}^{m \times k}$ scaled by a rank r. We perform matrix factorization to obtain two matrices $\boldsymbol{Y} \in \mathbb{R}^{m \times r}$ and $\boldsymbol{Z} \in \mathbb{R}^{r \times k}$ such that $x_{i,k} \approx y_i^T z_k$, $\forall i, k \in I$ with a probability closest to 1. Here, I denotes the indices of the existing elements in $\mathbf{X}$, such that the problem is represented in the following convex form:

$$\hat{\boldsymbol{X}} = \min_{\boldsymbol{Y},\boldsymbol{Z}} \frac{1}{2} \mathcal{P}_I ||\boldsymbol{X} - \boldsymbol{YZ}||_F^2. \tag{14}$$

Equation (14) is represented as one half of an averaged squared Frobenius norm of (5), where I is the complete set of indices $(i, j) \in I$ such that $\{(i, j) : i \in \{1, \cdots, m\}, j \in \{1, \cdots, k\}\}$. In this way, the operator $\mathcal{P}_I(\boldsymbol{X}_{i,j})$ behaves as follows:

$$\mathcal{P}_I(\boldsymbol{X}_{i,j}) = \begin{Bmatrix} \boldsymbol{X}_{i,j}, & \text{if} \quad (i,j) \in I \\ 0, & \text{otherwise} \end{Bmatrix}, \tag{15}$$

In the stochastic setup of the gradient descent for solving (14), we pick β known entries at a time and then take a gradient descent step that updates the matrices $\mathbf{Y}$ and $\mathbf{Z}$. Because of the cost structure, this means that the technique ends up updating only a maximum of β rows of $\mathbf{Y}$ and $\mathbf{Z}$ at a time. The procedure is outlined in **Algorithm 2**.

Algorithm 2 Proposed SGD method for Distributed Spectrum Occupancy Reconstruction in CRNs

Input: $\boldsymbol{X} \in \mathbb{R}^{m\times k}$: step size, θ: weight factor, μ
Input: matrix rank, r, learning rate
Output: $\hat{\boldsymbol{X}} \in \mathbb{R}^{m\times k}$
01: Partition $\boldsymbol{X} \in \mathbb{R}^{m\times k}$ into $\boldsymbol{Y} \in \mathbb{R}^{m\times r}$ and $\boldsymbol{Z} \in \mathbb{R}^{r\times k}$
02: Initialize step size θ and weight factor μ.
03: Find indices for non-missing entries
04: **While** termination condition not reached **do**
05: Pick β known entries with their indices.
06: Set up a completion sub-problem by finding the indices corresponding to the sub-matrices $\boldsymbol{Y}_\beta$ and $\boldsymbol{Z}_\beta$ which need to be modified. Consequently, find the subset I_β of indices out of the $\beta_{\boldsymbol{Y}}\beta_{\boldsymbol{Z}}$ indices.
07: Compute the residual $\boldsymbol{S}_\beta = \mathcal{P}_{I_\beta}(\boldsymbol{Y}_\beta\boldsymbol{Z}_\beta - \boldsymbol{X}^*_\beta)$.
08: Given a step size θ, update $\boldsymbol{Y}_\beta$ and $\boldsymbol{Z}_\beta$ as $Y_{\beta+1}$ and $Z_{\beta+1}$.
09: Fix $\mathbf{Y}$, find $\mathbf{Z}$ that minimizes $\frac{1}{2}||\boldsymbol{X} - \boldsymbol{Y}\boldsymbol{Z}||_F$
10: Fix $\mathbf{Z}$, find $\mathbf{Y}$ that minimizes $\frac{1}{2}||\boldsymbol{X} - \boldsymbol{Y}\boldsymbol{Z}||_F$
11: Update $\boldsymbol{Y}^T\boldsymbol{Y}$ and $\boldsymbol{Z}^T\boldsymbol{Z}$
12: Compute the MSE using (18).
13: **If** termination condition is reached
14: **Go to** 18:
15: **Else**
16: **Go to** 04:
17: **End If**
18: **End While**

F. SGD ALGORITHM DESCRIPTION

When performing an update of the spectrum sensing data matrix, we let $\beta_{\boldsymbol{Y}}$ rows of $\mathbf{Y}$ and $\beta_{\boldsymbol{Z}}$ be updated when β known entries are picked, where $\beta_{\boldsymbol{Y}} \leq \beta$ and $\beta_{\boldsymbol{Z}} \leq \beta$ [40]. This is shown in **Step 05:**. Then, in **Step 07:**, we have to define sub-matrices, firstly let $\boldsymbol{Y}_\beta$ be the sub-matrix corresponding $\mathbf{Y}$ with $\beta_{\boldsymbol{Y}}$ rows. Similarly, we let $\boldsymbol{Z}_\beta$ be the sub-matrix corresponding to $\mathbf{Z}$ with $\beta_{\boldsymbol{Z}}$ rows. At each iteration step β known entries are picked such that we have a sub-problem of completing matrix $\boldsymbol{X}_\beta$ of size $\beta_{\boldsymbol{Y}} \times \beta_{\boldsymbol{Z}}$. Therefore, we have β known entries at indices I_β, which need to be approximated by $\boldsymbol{Y}_\beta\boldsymbol{Z}^T_\beta$. In **Step 06:**, we define a residual matrix and denote it by $\mathbf{S}$ for updating the SGD algorithm. Therefore, if $\boldsymbol{S}_\beta$ is the residual matrix of this sub-problem, then the partial derivatives at $(\boldsymbol{Y}_\beta, \boldsymbol{Y}_\beta)$ are $(\boldsymbol{S}_\beta\boldsymbol{Z}_\beta, \boldsymbol{S}_\beta\boldsymbol{Y}_\beta)$. Here, $\boldsymbol{S}_\beta = \mathcal{P}_{I_\beta}(\boldsymbol{Y}_\beta\boldsymbol{Z}^T_\beta - \boldsymbol{X}^*_\beta)$ is of size $\beta_{\boldsymbol{Y}} \times \beta_{\boldsymbol{Z}}$, such that the updates for the proposed stochastic gradient descent are:

$$\boldsymbol{Y}_{\beta+1} = \boldsymbol{Y}_\beta - \theta\boldsymbol{S}_\beta\boldsymbol{Z}\left(\frac{\beta\mu}{\min(m,k)}(\boldsymbol{Z}^T\boldsymbol{Z}) + (1-\mu)(\boldsymbol{Z}^T_\beta\boldsymbol{Z}_\beta)\right)^{-1} \tag{16}$$

and

$$\boldsymbol{Z}_{\beta+1} = \boldsymbol{Z}_\beta - \theta\boldsymbol{S}_\beta\boldsymbol{Y}\left(\frac{\beta\mu}{\max(m,k)}(\boldsymbol{Y}^T\boldsymbol{Y}) + (1-\mu)(\boldsymbol{Y}^T_\beta\boldsymbol{Y}_\beta)\right)^{-1}, \tag{17}$$

where θ is the step size, $\frac{\beta}{\min(m,k)}$ is the normalization constant, and $\mu \in [0, 1]$ is a non-negative scalar that is used as a weight in order to weigh $\boldsymbol{Y}^T\boldsymbol{Y}$ and $\boldsymbol{Y}^T_\beta\boldsymbol{Y}_\beta$ differently. The normalization constant term $\frac{\beta}{\max(m,k)}$ ensures that the Frobenius norm $\frac{\beta(\boldsymbol{Y}^T\boldsymbol{Y})}{\max(m,k)}$ and $(\boldsymbol{Y}^T_\beta\boldsymbol{Y}_\beta)$ are of the same order as well as for the terms $\boldsymbol{Z}^T\boldsymbol{Z}$ and $\boldsymbol{Z}^T_\beta\boldsymbol{Z}_\beta$. In our case, we chose to compute $\boldsymbol{Y}^T\boldsymbol{Y}$ and $\boldsymbol{Z}^T\boldsymbol{Z}$ after every update until the whole matrix is reconstructed; that is, until the minimum of the cost function is obtained.

The flow diagram for the proposed SGD technique is illustrated in Fig. 4:

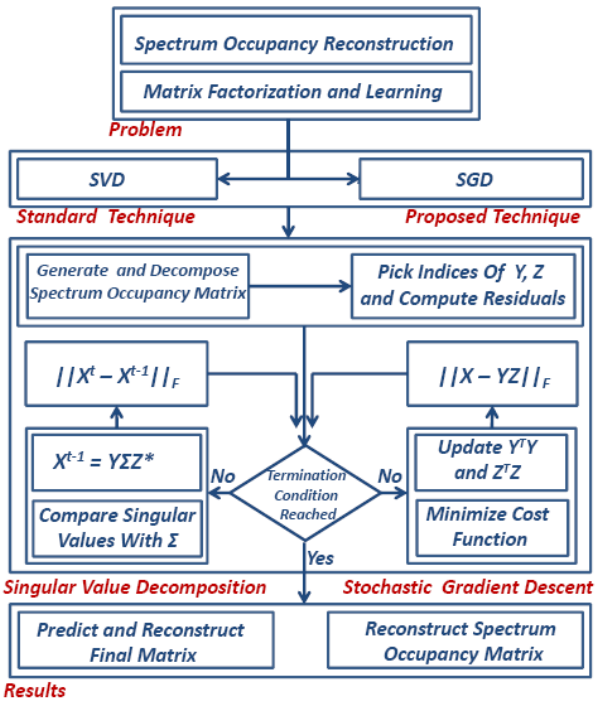


FIGURE 4. Comparison Flow Chart for the SGD Algorithm and SVD Algorithm for Matrix Factorization and Learning.

G. COMPARISON OF BOTH ALGORITHMS

Both algorithms are matrix factorization algorithms, the traditional matrix completion uses SVD while the proposed one uses SGD. The idea of the traditional matrix completion is that if the spectrum occupancy matrix is complete, SVD can be directly applied as a decomposition method and the method works very fine. However, having a complete matrix is very less likely in collaborative spectrum sensing. In the case whereby the spectrum occupancy matrix has a lot of

missing entries, which is typical for spectrum occupancy matrices, SVD can be applied with the assumption that the missing entries are zeros. This is intuitively incorrect in spectrum sensing since it implies the certain SU observed that the channel is not occupied, which creates some bias.

1) SVD

The general idea of using SVD is to approximate a low-rank matrix $\mathbf{X}$ by decomposing it to $\hat{\boldsymbol{X}} = \boldsymbol{Y}\boldsymbol{\Sigma}\boldsymbol{Z}^T = \boldsymbol{Y}\boldsymbol{T}^T$, where $\boldsymbol{T}$ is just $\boldsymbol{\Sigma}\boldsymbol{Z}^T$ and $\mathbf{Y}$ and $\mathbf{T}$ are low-rank matrices. This can be approximated as minimizing the Frobenius norm $||\boldsymbol{X} - \hat{\boldsymbol{X}}||$.

2) SGD

The idea of SGD is to use a similar technique to SVD for low-rank approximation of $\hat{X}$ by finding $\mathbf{Y}$ and $\mathbf{Z}$ that minimizes $f(\boldsymbol{Y}, \boldsymbol{Z}) = ||\boldsymbol{X} - \boldsymbol{Y}\boldsymbol{Z}||$. The difference with SGD is that it only considers the entries of $\mathbf{X}$ that are not missing which is not a convex function to minimize. However, SGD still works well since it makes this function convex by fixing alternately $\mathbf{Y}$ and $\mathbf{Z}$ and optimizing for $\mathbf{Y}$ and $\mathbf{Z}$ respectively before computing the final reconstruction error. This is shown in **Steps** 09:, 10: and 11: in **Algorithm 2**.

V. SIMULATION RESULTS AND DISCUSSION

This section consists of two subsections: i.e., simulation parameters and simulation set up, and simulation results and discussions. The simulation parameters subsection contains simulation parameters used for simulations using both algorithms, and preliminary results for checking the sanity of the proposed algorithm. The results subsection presents graphical simulation results and analyze the impacts of the matrix size, matrix rank, and step size on the performance of both the imputation algorithms. We also exemplify the performance of the algorithms using convergence tables.

A. SIMULATION SET UP AND PARAMETERS

The data set used in the simulation has been created artificially using MRFs, Gibbs distribution and Metropolis-Hastings algorithm, as discussed in **Section IV**. A fraction of SU nodes p_c are randomly picked at a time within the set of SUs to collaborate their spectrum sensing results with neighbors. This is done by making some transmissions fail while others succeed. In this way, we obtain a fraction p_c of successful collaboration which means that there are $1 - p_c$ latent values. Therefore, $p_c = X$ is the spectrum occupancy data matrix. This is the data set that forms the input spectrum sensing data matrix $\mathbf{X}$ which is very sparse in its composition and is decomposed and reconstructed by the two algorithms; matrix completion and stochastic gradient descent. The decomposition/factorization of $\mathbf{X}$ results in factor matrices $\mathbf{Y}$ and $\mathbf{Z}$ which provide the $1-p_c$ latent factors for the missing values, learned by the two algorithms separately. The two algorithms are evaluated by reconstructing the input data matrix with the help of their learned latent factors and finding the mean square error (MSE) of the original spectrum sensing data matrix with the constructed ones. The MSE is given by:

$$MSE = \sqrt{\frac{\sum_{m,k,x\in X}(x_{m,k} - \hat{x}_{mk})^2}{|X|}}. \tag{18}$$

By letting $y_{m,r}$ and $z_{r,k}$ be the elements of $\mathbf{Y}$ and $\mathbf{Z}$, respectively, a missing value at matrix position SU m and time slot k is predicted by

$$\hat{x}_{m,k} = \sum_{r=1}^{R} y_{m,r} z_{k,r}. \tag{19}$$

The performance of these two algorithms is tested for different values of latent values k, with different matrix ranks r, and with different step sizes (learning rates) θ. Unless stated otherwise, the rest of the simulation parameters are listed in **Table 1**.

TABLE 1. This Table Shows the Simulation Parameters Used for Setting up the CRN.

Parameter	Value
Maximum number of SUs, M	300
Frequency, f_c	2.1GHz
Total bandwidth, B	8MHz
Base station transmission power, P_{tx}	24dBm [250mW]
Communications channel (Noise)	Gaussian (Additive white)
Noise power density	-174dBm
Receiver noise figure	9dB
Path loss exponent, α	3
Minimum duration of each state, Δt	1 min
Average percentage of busy channels	5%
Average SNR for busy channel	$1 - 14$ dB
Test matrix ranks, r	4,5, and 6
Scalar weight, μ	0.5
Learning rate, θ	0.01, 0.1

our simulations, we a probabilistic channel modeling framework which characterizes the distribution of a sample of the channel as well as its spatial correlation. We simulate a CRN such that we are able to sample uniformly from the whole network formed by a 50 × 50 grid. This is shown if Fig. 5, where the yellow spots on the grid represent sites

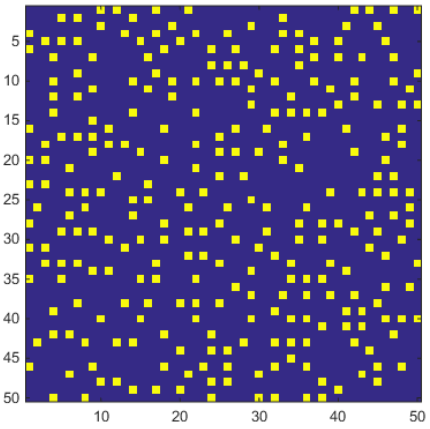


FIGURE 5. Results show SU opinions about spectrum occupancy on a 50 × 50 grid.

occupied by PU activities. As we sample from the network, we notice that the sample mean of the fraction of occupied sites to the unoccupied sites on this graph decreases with the increase in network size. For example, Fig. 5 shows a 50×50 network with a sample mean of 0.1129 compared to 0.0838 when a 1000×1000 grid is simulated. The result in Fig. 5 is obtained from toggling the states between ON and OFF and thus shows different sites on the grid showing spectrum occupancy states, denoted by yellow (bright) spots indicating sites occupied by SUs having an opinion of an occupied spectrum. The blue (dark) patches indicate the sites that SUs perceive as vacant and open for opportunistic transmission. As it can be seen in Fig. 5, the Metropolis-Hastings algorithm together with the Ising model is able to sample from the whole CRN with a probability closer to 1. Therefore, we use a proper target distribution in the form of an exponential bivariate distribution, defined inside a hyper-cube (four-dimensional cube).

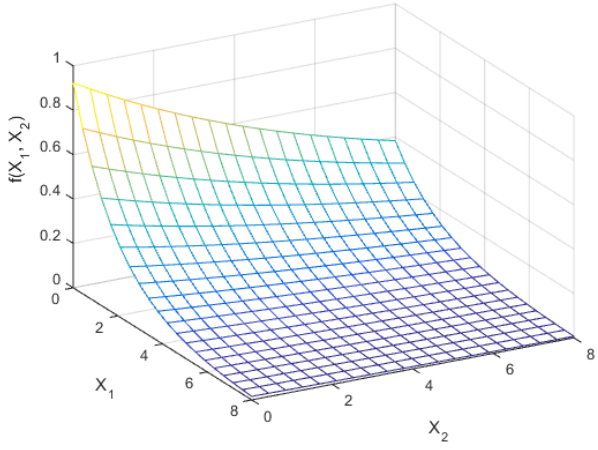


FIGURE 6. Target distribution in the bivariate exponential distribution that the algorithm samples from.

Fig. 6 indicates that the algorithm is able to sample from the whole proposal distribution, even at its tails, which shows that all the sites of the grid are covered by the exponential bivariate distribution. It is worth noting that the states sampled from the Markov chain are also samples drawn from the target distribution using the proposal distribution where the proposals X_1 and X_2 are sampled from a uniform (0,8) distribution. That is, we are sampling proposals for $\mathbf{X}$ from within a box $[0, 10]^m$ as stated in **Section III**.

B. SIMULATION RESULTS AND DISCUSSION

In this subsection, we perform simulations for the performance of the two algorithms. We assume that there are 300 SUs distributed in the CR network grid, where each SU gathers the samples from other SU nodes once per time slot. Thus, we define the sampling rate as the ratio of the collected measurements to the total number of SUs. For example, suppose that an SU receives spectrum occupancy measurements from 49 other SUs, which makes it 50 opinions. This makes $p_c = 50/300 = 16.67\%$ which is then treated as the sampling rate. Similarly, where 249 measurements have been received, the sampling rate is 83.33%. However, we have been careful not to approach much closer to the total number of SUs in the CRN because then we would not have any missing values and thus defeat the purpose of this work.

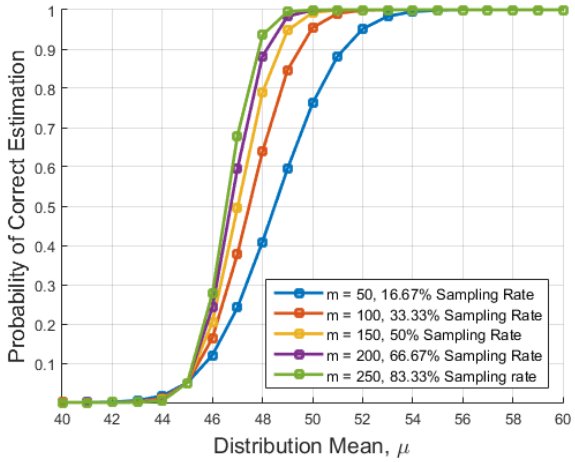


FIGURE 7. This shows the probability of the correct estimation of missing values during the reconstruction process using a hypothesis test with a different number of observations *m*.

In Fig. 7, as expected, we can observe that the curves become more steeper as the sampling rate, p_c increases. When the sampling rate is 16.67%, which is the lowest we have chosen, the estimation performance is poor, as it reaches 100% at a distribution mean of 55. As the sampling rate increases, the probability of correct estimation also increases. For example, when the sampling rate is at 83.33%, the estimation reaches 100% at a distribution mean of 49.

Now, assuming that the observation matrix has been evaluated to be MAR as the mechanism of missingness, we run simulations for the proposed spectrum occupancy reconstruction and learn the missing values using a stochastic gradient descent algorithm. We use the estimation error in (5) as our evaluation metric to test the estimation performance of the stochastic gradient descent method. The overall size of the spectrum usage data matrix is $M \times K$ with partial entries of spectrum usage known owing to the limited number of SUs reporting.

The result in Fig. 8 has been plotted as the amount of error in computing the gradient of the algorithm for different values of matrix ranks r, varying between 4 and 6. These are the results for computational error when using the stochastic gradient descent method for spectrum occupancy reconstruction. We observe from Fig. 8 that with a smaller rank (e.g., $r = 4$), the computational error is very small compared to $r = 6$. However, both computations take the same number of iterations to converge. When taking $r = 5$, the computational error initially falls between that of $r = 4$ and that of $r = 6$, but eventually becomes much better than both of them and converges earlier.

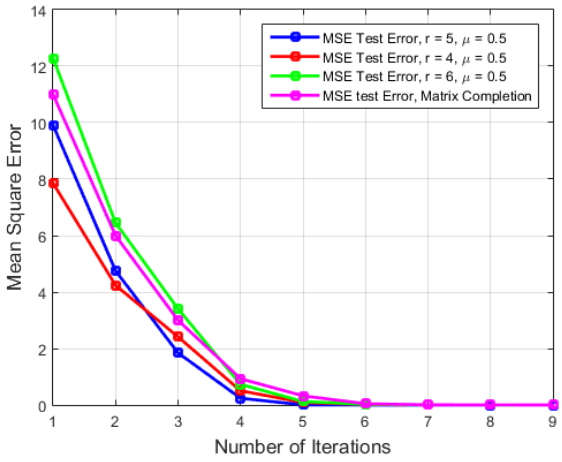


FIGURE 8. Convergence of the algorithm with computational test errors in the spectrum occupancy reconstruction against number of iterations.

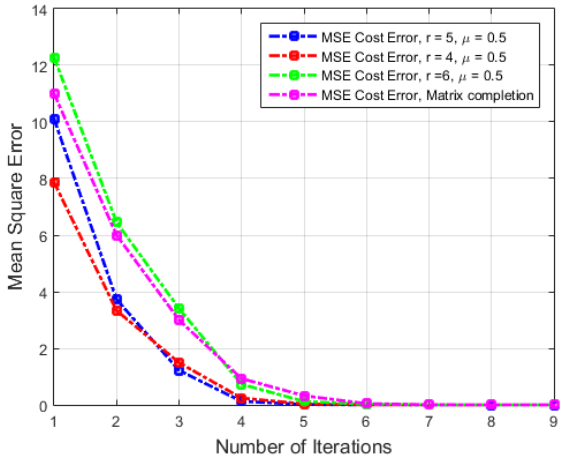


FIGURE 9. Convergence of the algorithm with computational cost errors in the spectrum occupancy reconstruction against the number of iterations.

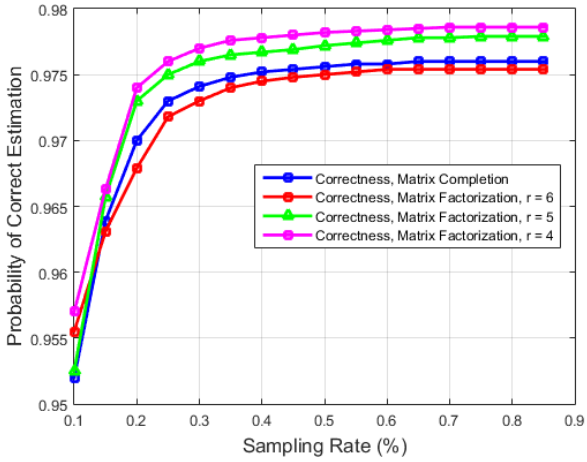


FIGURE 10. Performance of correct estimation against sampling rate in the distributed reconstruction of the spectrum occupancy data matrix compared with the matrix completion algorithm.

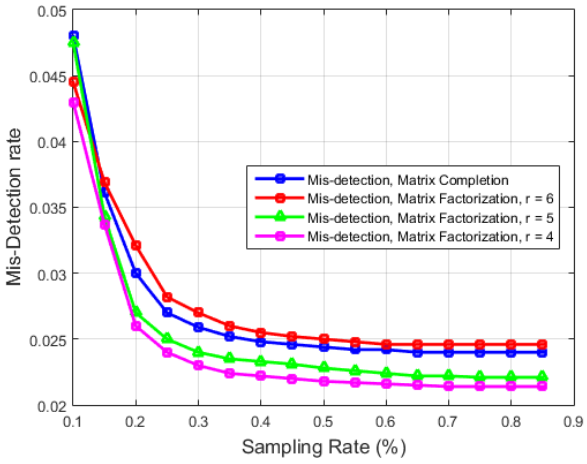


FIGURE 11. Probability of incorrect estimation in reconstructing the spectrum occupancy data matrix. The performance comparison is done with the matrix completion algorithm.

For the same values that we used in the simulation of Fig. 8, we perform simulations for the cost error in running the algorithm. This result is shown in Fig. 9, where we observe that using rank $r = 6$, the behavior is the same as the computational error shown in Fig. 8. However, the same behavior seen in Fig. 8 could not be repeated for $r = 4$ and $r = 5$. The demand for space in memory becomes almost the same for both $r = 4$ and $r = 5$.

From the results illustrated in Fig. 8 and Fig. 9, we can conclude that in order to perform reconstruction using the stochastic gradient descent, the rank of the sample matrix should optimally be $r = 5$, not more.

we fix duration Δt to a minimum of 1 minute to avoid the spectrum occupancy status from changing rapidly. The sampling rate is varied from 10% to 83.33%. By sampling rate we refer to the fraction $p_c = X$. The crossover probability between H_0 and H_1 is fixed 47.50%, as obtained in Fig. 7. We consider the case where the number of collaborating SUs is between 30 and 250 out of a total of 300 SUs. Every spectrum occupancy state lasts for a maximum of 5 minutes to keep it from toggling between ON and OF before the algorithm finishes computing.

In Fig. 10 and Fig. 11, it can be seen that the proposed algorithm can reconstruct the spectrum occupancy matrix with a high correctness rate even at a 10% sampling rate. As the sampling rate increases, there is a corresponding decrease in the miss detection rate. As seen in Fig. 10, when the sampling rate reaches 60%, the occupancy data matrix reconstruction correctness rate has already reached 97.86% when $r = 4$. A corresponding decrease in the mis-detection rate of 2.14% is seen in Fig 11. This means that as the sampling rate reaches 60%, the proposed algorithm guarantees a mis-detection rate that is less than 3%, when the reconstruction rate $r = 4$. There is a performance improvement that is noticeable with the matrix reconstruction rate; for example, when $r = 6$, the proposed algorithm performs worse than matrix completion, but

when the value of r decreases, the performance improves. This can be explained in terms of introducing errors and reducing the errors. When the matrix rank $r = 6$, more errors are introduced into the algorithm than when the matrix rank r is equal to 5 or 4. However, there is always a price to pay for every improvement since when reducing the rate of reconstruction, it means that the number of computations, and hence the cost of the algorithm.

C. OVERALL PERFORMANCE COMPARISON

The following presentation of the convergence tables seeks to summarize the performance of both algorithms. We begin by varying the size of the spectrum occupancy matrix $m \times k$, matrix ranks r, while keeping the learning rate θ constant and evaluate the total error and convergence time. The number of iterations is kept at 10 for all our investigations

The results shown in Table 2 and Table 3 show that the computation error (MSE) is dependent on the size of the spectrum occupancy matrix. Reducing the size of the spectrum occupancy matrix by a factor of 10 actually improves the reconstruction performance in terms of the MSE and convergence time. Now, we reduce the learning rate from 0.1 to 0.01 and evaluate the response of the algorithms in terms of reconstruction performance.

NB: * In SVD sampling with matrix rank is not performed, matrix rotation about an arbitrary axis is performed instead.

The results shown in Table 4 and Table 5 indicate that reducing the learning rate increases the reconstruction error (MSE) but reduces the convergence time for both algorithms. This behavior is due to weight decay which is easier to understand from intuition. Weight decay means that when a smaller learning rate is used, more and more epochs are used in training and the training weights are suppressed more and more. This as well suppresses the result due to too much training. When this suppression happens, the model becomes weak such that it underfits the model and both training and testing loss gets larger.

TABLE 2. Convergence results for matrix size 3000 × 10000.

Algorithm	m	k	r	θ	MSE	time
SGD	3000	10000	4	0.1	3.59e-08	2.59 sec
SGD	3000	10000	5	0.1	2.35e-07	3.52 sec
SGD	3000	10000	6	0.1	4.80e-07	3.12 sec
SVD	3000	10000	*	0.1	3.55e-07	3.20 sec

TABLE 3. Convergence results for matrix size 300 × 1000.

Algorithm	m	k	r	θ	MSE	time
SGD	300	1000	4	0.1	NaN	0.55 sec
SGD	300	1000	5	0.1	NaN	2.13 sec
SGD	300	1000	6	0.1	NaN	0.21 sec
SVD	300	1000	*	0.1	NaN	1.05 sec

TABLE 4. Convergence results for matrix size 3000 × 10000.

Algorithm	m	k	r	θ	MSE	time
SGD	3000	10000	4	0.01	1.65e-01	3.46 sec
SGD	3000	10000	5	0.01	2.27e-01	2.89 sec
SGD	3000	10000	6	0.01	2.23e-01	3.26 sec
SVD	3000	10000	*	0.01	2.25e-01	3.15 sec

TABLE 5. Convergence results for matrix size 300 × 1000.

Algorithm	m	k	r	θ	MSE	time
SGD	300	1000	4	0.01	NaN	4.08 sec
SGD	300	1000	5	0.01	NaN	2.11 sec
SGD	300	1000	6	0.01	NaN	0.56 sec
SVD	300	1000	*	0.01	NaN	1.30 sec

VI. CONCLUSION

In this paper, we envisioned a solution for spectrum occupancy reconstruction in CRNs. Spectrum occupancy reconstruction is often encountered in distributed spectrum sensing when the spectrum sensing data collaborated by SUs has gaps of missing entries. In this paper, we considered the problem of spectrum occupancy reconstruction across the CRN as a plenary grid on an MRF. We formulated the problem as an magnetic excitation state recovery problem, and an SGD algorithm was applied to solve the matrix factorization. SGD, which is able to learn and impute the missing values with a low reconstruction error compared to SVD. The graphical results indicate that with more collaborating SUs (less missing entries), the reconstruction has a high resolution, while the numerical results indicate that the convergence time for both compared algorithms is dependent on the matrix size and learning rate. It is also observed that the SGD algorithm performs better than the SVD algorithm, save for when the matrix rank becomes greater than 5. The results also show that in as much as factorizing a matrix with missing entries can be done easily using SVD, when an optimization technique is needed to solve the factorized matrix, SGD comes handy. This is on considering the fact that when the data matrix grows the complexities to be dealt with also increase. As the results in the above tables shows, the complexity grows with the number of rows (number of SUs). Then, better performance depends on the choice of algorithm, but SGD is able to deal with such complexities since it deals with the non-missing values which are chosen randomly every instance.

REFERENCES

[1] A. Zaeemzadeh, M. Joneidi, B. Shahrasbi, and N. Rahnavard. (Aug. 2015). "Missing spectrum-data recovery in cognitive radio networks using piecewise constant nonnegative matrix factorization." [Online]. Available: https://arxiv.org/abs/1508.07269v1

[2] B. Aneja, K. Sharma, and A. Rana, "Spectrum sensing techniques for a cognitive radio network," in *Proc. Adv. Syst. Optim. Control*. Singapore: Springer, Jun. 2018, pp. 133–144.

[3] J. W. Yufan, "Cooperative spectrum sensing for cognitive radio networks under Nakagami-M fading channels," in *Proc. IEEE Int. Conf. Ind. Control Electron. Eng. (ICICEE)*, Aug. 2012, pp. 1770–1772.

[4] S. C. Shinde and A. N. Jadhav, "Performance comparison of centralized cooperative spectrum sensing based on voting rules in cognitive radio," *Int. J. Adv. Eng. Res. Sci.*, vol. 3, no. 5, pp. 72–76, 2016.

[5] W. Na, J. Yoon, S. Cho, D. Griffith, and N. Golmie, "Centralized cooperative directional spectrum sensing for cognitive radio networks," *IEEE Trans. Mobile Comput.*, vol. 17, no. 6, pp. 1260–1274, Jun. 2018.

[6] J. Gillard and K. Usevich. (Feb. 2018). "Structured low-rank matrix completion for forecasting in time series analysis." [Online]. Available: https://arxiv.org/abs/1802.08242
[7] J. Meng, W. Yin, H. Li, E. Houssain, and Z. Han, "Collaborative spectrum sensing from sparse observations using matrix completion for cognitive radio networks," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 2, pp. 327–337, Feb. 2011.
[8] E. J. Candès and Y. Plan, "Matrix completion with noise," *Proc. IEEE*, vol. 98, no. 6, pp. 925–936, Jun. 2010.
[9] A. Ghasemi and E. S. Sousa, "Collaborative spectrum sensing for opportunistic access in fading environments," in *Proc. 1st IEEE Int. Symp. New Frontiers Dyn. Spectr. Access Netw.*, Baltimore, MD, USA, Nov. 2005, pp. 131–136.
[10] Z. Tian, "Compressed wideband sensing in cooperative cognitive radio networks," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, New Orleans, LA, USA, Nov. 2008, pp. 1–5.
[11] D. Xue, E. Ekici, and M. C. Vuran, "Cooperative spectrum sensing in cognitive radio networks using multidimensional correlations," *IEEE Trans. Wireless Commun.*, vol. 13, no. 4, pp. 1832–1843, Apr. 2014.
[12] M. C. Hlophe, B. T. Maharaj, and S. Hamouda, "Distributed spectrum sensing for cognitive radio systems using graph theory," in *Proc. IEEE AFRICON*, Capetown, South Africa, Sep. 2017, pp. 267–272.
[13] Z. Song, H. Jijun, L. Peiguo, and H. Jianguo, "Distributed compressed spectrum sensing via cooperative support fusion," *Int. J. Distrib. Sensor Netw.*, vol. 9, no. 12, p. 862320, Dec. 2013.
[14] J. Riihijarvi, P. Mahonen, M. Wellens, and M. Gordziel, "Characterization and modelling of spectrum for dynamic spectrum access with spatial statistics and random fields," in *Proc. IEEE 19th Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Cannes, France, Sep. 2008, pp. 1–6.
[15] M. Wellens, J. Riihijarvi, M. Gordziel, and P. Mahonen, "Spatial statistics of spectrum usage: From measurements to spectrum models," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Dresden, Germany, Jun. 2009, pp. 1–6.
[16] H. Li, "Reconstructing spectrum occupancies for wideband cognitive radio networks: A matrix completion via belief propagation," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Capetown, South Africa, May 2010, pp. 1–6.
[17] H. Li, Z. Han, and Z. Zhang, "Communication over random fields: A statistical framework for cognitive radio networks," in *Proc. IEEE Globe Commun. Conf. (Globecom)*, Houston, TX, USA, Dec. 2011, pp. 1–5.
[18] P. Potier, C. Sorrells, Y. Wang, L. Qian, and H. Li, "Network-wide spectrum situation reconstruction using total variation inpainting in cognitive radio ad hoc networks," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, Kathmandu, Nepal, Dec. 2011, pp. 1–5.
[19] D. Willkomm, S. Machiraju, J. Bolot, and A. Wolisz, "Primary users in cellular networks: A large-scale measurement study," in *Proc. 3rd IEEE Symp. New Frontiers Dyn. Spectr. Access Netw.*, Chicago, IL, USA, Oct. 2008, pp. 1–11.
[20] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, p. 436, 2015.
[21] Y. Cui, X. Jing, S. Sun, X. Wang, D. Cheng, and H. Huang, "Deep learning based primary user classification in cognitive radios," in *Proc. 15th IEEE Int. Symp. Commun. Inf. Technol. (ISCIT)*, Nara, Japan, Oct. 2015, pp. 165–168.
[22] G. J. Mendis, J. Wei, and A. Madanayake, "Deep learning-based automated modulation classification for cognitive radio," in *Proc. IEEE Int. Conf. Commun. Syst. (ICCS)*, Shenzhen, China, Dec. 2016, pp. 1–6.
[23] F. Paisana *et al.*, "Context-aware cognitive radio using deep learning," in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, Baltimore, MD, USA, Mar. 2017, pp. 1–2.
[24] E. J. Candès and B. Recht, "Exact matrix completion via convex optimization," *Found. Comput. Math.*, vol. 9, no. 6, pp. 717–772, 2009.
[25] R. Keshavan, S. Oh, and A. Montanari, "Matrix completion from a few entries," in *Proc. ISIT*, Jun./Jul. 2009, pp. 324–328.
[26] J. Ma, G. Zhao, and Y. Li, "Soft combination and detection for cooperative spectrum sensing in cognitive radio networks," *IEEE Trans. Wireless Commun.*, vol. 7, no. 11, pp. 4502–4507, Nov. 2008.
[27] S. Maleki, A. Pandharipande, and G. Leus, "Energy-efficient distributed spectrum sensing for cognitive sensor networks," *IEEE Sensors J.*, vol. 11, no. 3, pp. 565–573, Mar. 2011.
[28] J. Yao, J. Cao, Q. Zheng, and J. Ma, "Pre-processing of incomplete spectrum sensing data in spectrum sensing data falsification attacks detection: A missing data imputation approach," *IET Commun.*, vol. 10, no. 11, pp. 1340–1347, Jul. 2016.
[29] J. Rochner, L. Balents, and K. P. Schmidt, "Spin liquid and quantum phase transition without symmetry breaking in a frustrated three-dimensional Ising model," *Phys. Rev. B, Condens. Matter*, vol. 94, no. 20, p. 201111, Nov. 2016.
[30] A. Hamlili, "Intelligibility of Erdös-Rényi random graphs and time varying social network modeling," in *Proc. ACM Int. Conf. Smart Digit. Environ.*, Jul. 2017, pp. 201–206.
[31] H. Chen, H. Zhang, and C. Shen. (Feb. 2018). "Pseudo-double transition in the networked Ising model with core-periphery structures." [Online]. Available: https://arxiv.org/abs/1802.00547
[32] B. Andrew and A. Selamat, "Systematic literature review of missing data imputation techniques for effort prediction," in *Proc. Int. Conf. Inf. Knowl. Manage.*, Singapore, 2012, pp. 222–226.
[33] S. Nie, M. Zheng, and Q. Jing, "The deep regression Bayesian network and its applications: Probabilistic deep learning for computer vision," *IEEE Signal Process. Mag.*, vol. 35, no. 1, pp. 101–111, Jan. 2018.
[34] R. W. Krause, M. Huisman, C. Steglich, and T. A. Sniiders, "Missing network data: A comparison of different imputation methods," in *Proc. IEEE/ACM Int. Conf. Adv. Social Netw. Anal. Mining (ASONAM)*, Aug. 2018, pp. 159–163.
[35] L. H. Rubin, K. Witkiewitz, J. S. Andre, and S. Reilly, "Methods for handling missing data in the behavioral neurosciences: Don't throw the baby rat out with the bath water," *J. Undergraduate Neurosci. Educ.*, vol. 5, no. 2, p. A71, Jun. 2007.
[36] J. Riihijarvi, P. Mahonen, M. Wellens, and M. Gordziel, "Characterization and modelling of spectrum for dynamic spectrum access with spatial statistics and random fields," in *Proc. IEEE 19th Int. Symp. Pers., Indoor Mobile Radio Commun. (PIMRC)*, Cannes, France, Sep. 2008, pp. 1–6.
[37] T. Hazan, S. Maji, and T. Jaakkola, "On sampling from the Gibbs distribution with random maximum *a-posteriori* perturbations," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 1268–1276.
[38] S. Chib and E. Greenberg, "Understanding the metropolis-Hastings algorithm," *Amer. Stat.*, vol. 49, no. 4, pp. 327–335, 1995.
[39] I. Yildirim, "Bayesian inference: Metropolis-Hastings sampling," Dept. Brain Cogn. Sci., Univ. Rochester, Rochester, NY, USA, 2012. [Online]. Available: https://www.mit.edu/~ilkery/papers/MetropolisHastings-Sampling.pdf.
[40] A. Ma and D. Needell. (Feb. 23, 2017). "Adapted stochastic gradient descent for linear systems with missing data." [Online]. Available: https://arxiv.org/abs/1702.07098
[41] N. Agarwal, Z. Allen-Zhu, B. Bullins, E. Hazan, and T. Ma, "Finding approximate local minima faster than gradient descent," in *Proc. 49th Annu. ACM SIGACT Symp. Theory Comput.*, Jun. 2017, pp. 1195–1199.

MDUDUZI C. HLOPHE was born in Nhlangano, Shiselweni Region, Swaziland, in 1986. He received the bachelor's degree in electronic engineering from the University of Swaziland, Matsapha, Swaziland, in 2012, and the master's degree in wireless communications from the University of Johannesburg, Johannesburg, South Africa, in 2015. He is currently pursuing the Ph.D. degree in engineering with the the University of Pretoria, Pretoria, South Africa. His research interests include mathematical modeling of multivariate statistics, classification methods, knowledge discovery, reasoning with uncertainty and inference, and predictive analytics and inference with applications in wireless communications, finance, health, and robotics.

SUNIL B. T. MAHARAJ received the Ph.D. degree in engineering, with a minoring in wireless communications, from the University of Pretoria, where he is currently a Full Professor and holds the research position at the Sentech Chair in Broadband Wireless Multimedia Communications, Department of Electrical, Electronic and Computer Engineering. His research interests include OFDM-MIMO systems, massive MIMO, cognitive radio resource allocation, and 5G cognitive radio sensor networks.