Prediction and Modeling of Spectrum Occupancy for Dynamic Spectrum Access Systems

Hamed Mosavat-Jahromi, *Graduate Student Member, IEEE*, Yue Li, Lin Cai, *Fellow, IEEE*, and Jianping Pan, *Senior Member, IEEE*

***Abstract*—In a dynamic spectrum allocation (DSA) system, reliable prediction of spectrum occupancy based on a spectrum consumption model (SCM) is critical for system design, performance analysis, and evaluation. In this article, we focus on a low-level abstracted measured dataset from a massive campaign and investigate the occupancy of representative frequency bands. First, we apply an autoregressive-moving-average (ARMA) model combined with a low-pass filter, given the stationarity of the channel measurement dataset and thanks to the computational simplicity of the model. The average received power and off-state probability are extracted from the measured data. According to the results, the measured and predicted data are in good agreement. Comparing the proposed model-based ARMA with the popular long short-term memory learning algorithm, they have similar error accuracy with pre-processed data, while ARMA has a much lower training complexity. In the second step, we develop an SCM describing the spectrum usage for designing and examining the DSA system. We extract the periodic, aperiodic low-frequency, and burst components of the time series. Also, a binary sequence is extracted from a sparse occupancy channel, and modelled by a non-homogeneous Markov chain. Results show that the model-generated data can maintain the same statistics as the measured data.**

***Index Terms*—Cognitive radio, dynamic spectrum access, spectrum prediction, LSTM, ARMA.**

I. Introduction

RADIO frequency is the vital resource on which wireless communication services rely. Due to the enormous growth in the number of mobile devices in recent years, spectrum scarcity becomes a pressing issue for spectrum management agencies and wireless service providers [1]–[3]. Traditionally, the spectrum is statically allocated to particular licensees over wide geographical regions. Measurements in [4] show that roughly 80% of the lower bands are empty of any signals within 7 dB of the environmental noise. In other words, a significant portion of the licensed spectrum, called *white spaces*, is unused. The challenge of ever-increasing demands for spectrum and exploiting it efficiently, has attracted considerable research efforts [5]–[8].

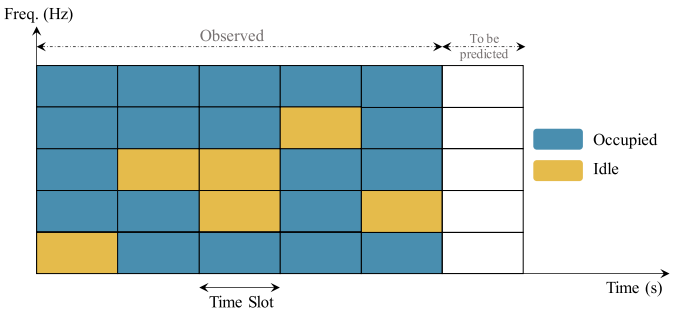


Fig. 1. Example of a DSA system.

Dynamic Spectrum Access (DSA) is promising to relieve the spectrum scarcity problem and utilize the white spaces in the spectrum. It allows the systems to share the spectrum and adjust spectrum assignments in real time [1], [9]. It is different from the other cognitive radio approach which temporarily allows the unlicensed or secondary users (SU) to use some vacant licensed bands opportunistically without interfering with the licensed or primary users (PU). DSA is technically simple and has attracted the attention of spectrum regulators recently. Traditionally, spectrum in a large area will be allocated to users for a long period of time, e.g., months to years. Using DSA, spectrum in a specific area can be dynamically allocated in the time unit of hours or tens of minutes only.

Spectrum prediction and statistical modeling are two important aspects of DSA, which are the main focuses of this article. DSA systems need to be aware of how spectrum resources are being used in different locations and situations. By deployment of the DSA system, the spectrum can be exploited efficiently through dynamic spectrum assignment, which is done by taking into account the spatial and temporal traffic statistics of different services [10]. Spectrum opportunity identification and exploitation are two key components of the DSA systems. While the former one is responsible for intelligently identifying and tracking the spectrum both temporally and spatially, the latter one decides when and how a secondary user uses the spectrum. Spectrum prediction is a complementary approach to foresee the behavior of a frequency band. It will facilitate the spectrum management and optimize user assignment to improve the DSAs' performance [11], [12]. Fig. 1 shows an example of a DSA system in which the channel states of the

Manuscript received July 23, 2020; revised November 12, 2020; accepted December 18, 2020. Date of publication January 1, 2021; date of current version September 9, 2021. This work was supported in part by Natural Sciences and Engineering Research Council of Canada (NSERC), CRC, MITACS, and Compute Canada. The associate editor coordinating the review of this article and approving it for publication was F. Gao. *(Corresponding author: Lin Cai.)*

Hamed Mosavat-Jahromi, Yue Li, and Lin Cai are with the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC V8P 5C2, Canada (e-mail: hamedmosavat@uvic.ca; liyue331@uvic.ca; cai@ece.uvic.ca).

Jianping Pan is with the Department of Computer Science, University of Victoria, Victoria, BC V8W 2Y2, Canada (e-mail: pan@uvic.ca).

Digital Object Identifier 10.1109/TCCN.2020.3048105

future time slots are predicted to be occupied by the current spectrum owner or idle. Existing spectrum prediction methods can be classified into model-based and non-model-based methods, represented by linear regression and neural network (NN), respectively. Generally, NNs are used to learn and model complex non-linear patterns, and their performance is highly dependent on the dataset instead of the mathematical model. In other words, this method is appropriate for the case where there is lack of a reasonable model a priori. Long short-term memory (LSTM) which is an artificial NN method uses the historical and current data to predict the future [13]. Therefore, it may capture the data's temporal correlation. On the other hand, the linear regression is typically a linear model with a computational complexity of $O(N)$, where N is the length of the time series, and the model coefficients can be estimated fast with less computation. Accordingly, the time series can be predicted much easier [14]. From the DSA systems' point of view, especially considering the large number of users to handle in real time, a simple yet effective model is of a high priority.

Further, the statistical modeling of the spectrum usage, i.e., spectrum consumption model (SCM), plays a pivotal role in DSA systems design. Moreover, it provides a common method to assess the spectrum usage compatibility. SCMs provide a way to capture the spectral, spatial, and temporal characteristics of any wireless systems. The collected data and captured parameters by the SCMs allow the development of spectrum management policy [1], [15], and pave the way for smart spectrum [16]. The SCMs may also help regulators to determine the spectrum rights and its boundaries, allow designers to design DSA systems efficiently, and assist markets by defining the allowed amount of spectrum that is traded as well as a better spectrum sharing policy. The IEEE 1900.5.2 standard [17] focuses on the usage of the SCMs to evaluate the compatibility, interference, and coexistence of the devices in the DSA systems. It has also specified different elements for an SCM, including the total reference power, spectral power density, power map, starting time, and ending time [1], [15].

In this article, we focus on the low-level abstracted measured data from a massive measurement campaign and investigate the occupancy of representative frequency bands. The main contributions of this article are twofold: first, because of the stationarity of the channel measurement results and the DSA's requirement of low complexity, we apply autoregressive-moving-average (ARMA) and LSTM models combined with a low-pass (LP) filter to the dataset for one-hour ahead occupancy prediction, and compare the results accordingly. Second, we provide a useful SCM depicting the spectrum occupancy for designing and examining the DSA system. For the occupancy prediction, a priori information of the dataset is considered. The Gauss-Newton (GN) algorithm is used to estimate the parameters of the ARMA model, where the initial values are obtained by the long autoregressive (AR) model. Two different types of data are extracted from the raw measurements which are the average received power and the probability of the off state. According to the simulation, high-precision predictions are obtained for both types of the data. For the spectrum occupancy modeling, we extract the periodic components, bursts, and aperiodic low-frequency part inspired from the nature of the dataset, and model them separately. We examine the model-generated data from different perspectives, including the time/frequency domains, probability density function (PDF), autocorrelation function (ACF), and partial autocorrelation function (PACF). Results show that the model-generated data can well approximate the measured data in all of the above aspects. Furthermore, an ON/OFF sequence is extracted from the measurements of the channel with sparse occupancy and then modeled by a non-homogeneous Markov chain. High similarities to the measured dataset are achieved according to the modeling results. The channel prediction and occupancy model are all performed in a centralized way on a server which collects different sensors data. The occupancy state or power level of different frequency channels in the sensors' locations will be reported to the network's users for the following time slots.

The organization of the rest of this article is as follows. In Section II, the related works are presented. Section III is dedicated to the dataset description and presentation. The spectrum prediction models and results are explained in Sections IV and V, respectively. Section VI describes the statistical spectrum modeling of the dataset. Finally, Section VII concludes this article.

II. Related Work

A. Spectrum Measurement

Several spectrum measurement campaigns have been done around the world. In [18], based on the spectrum measurements in the Netherlands, it has been shown that the totally vacant channel state exists with a high probability. Also, a stochastic model for the duty cycle distribution, i.e., the average occupancy of a channel, has been introduced according to the 200 KHz channel measurements. The proposed duty cycle is useful for DSA systems and vacant channel determination. The spectrum occupancy of some communication bands in the range of 50–860 MHz has been investigated in Leicester, U.K., in [19]. It has been demonstrated that the average occupancy of 50–470 MHz band is about 3.71%, and this band can be the candidate for the dynamic spectrum applications. A cloud-based system including the architecture and an initial system prototype was presented in [20]. Another spectrum measurement has been done for an outdoor application in [21]. The frequency range of the measurement was between 75 MHz and 3 GHz. The resolution bandwidth was 10 KHz which evenly divides the frequency blocks. The results from the measurements show remarkable spectrum opportunities in frequencies above 1 GHz.

B. Spectrum Prediction

In [22], the authors concentrated on spectrum prediction as an alternative approach to increase the efficiency of the spectrum sensing. Jacob *et al.* proposed two fast schemes based on Bayesian inference, i.e., a method in which Bayes' rule is used to update the probability distribution. According to the predicted probabilities, the channels can be sorted and the one

in a lower rank will be neglected in the spectrum sensing process. The performances of the proposed approaches in [22] are compared with the statistical methods in this work.

NN and recurrent NN (RNN) are two learning approaches to predict a dataset. Some kinds of datasets such as traffic data in large-scale networks have a non-linear nature. Azzouni and Pujolle in [23] utilized LSTM in RNN to train the prediction model, which was shown to outperform the traditional methods in terms of accuracy. In [24], the RNN method was applied to the RF measurements of Land Mobile Radio (LMR) to improve the performance, assuming that traffic data was multi-modal and non-stationary. Channel utilization efficiency and aggregate waiting time were considered as two comparison metrics due to the dataset's non-linearity. LSTM and fully-connected multi-layer perceptron methods have been applied to two different frequency bands, 1.82–1.875 GHz and 5.27–5.29 GHz in [25], and it is concluded that the LSTM model predicts more accurately.

Convolutional neural networks (CNNs) are a kind of neural networks with capability of features extraction from data with convolution structures [26]. They have been designed for image datasets and they can extract spatially invariant local relationships [27], [28]. Despite their competitive classification accuracy, they suffer from slow training speed and high computational cost. CNNs are applicable for time series forecasting problems as well where the temporal and spatial patterns of the dataset are extracted simultaneously. In multi-variate time series, an input block contains parallel observations in different time slots, and they are fed into the CNN model as a 2D data. CNNs are applicable to univariate time series as well. However, each causal convolutional filter acts as a finite impulse response (FIR) filter. Temporal CNNs assume that relationships are time-invariant and a fixed set of filter weights are derived at different time steps [27]. On the other hand, RNNs with the help of internal memory gates are a better fit for univariate time series forecasting.

Hidden Markov model (HMM) can also be utilized in DSA systems. This model can be helpful in the identification of the observation sequences with the same pattern [29]. Since HMMs are good at reproducing the training sequences, they can be used in channel occupancy prediction in DSAs. The spectrum occupancy prediction error was studied in [30], where the next time slot prediction error was modeled as a function of channel detection error and PU's state transition probability. It showed that HMM is a good candidate for duty cycle prediction. However, the HMM-based prediction methods do not show good accuracy on all kinds of data especially on non-Markovian real-world spectrum data [31].

The received power prediction may be utilized in scheduling and channel access procedure. Several researches have been done on channel access problem using deep reinforcement learning in the medium access control (MAC) layer. In [32], a dynamic multi-channel access problem has been studied in which correlated channels follow an unknown joint Markov model. Using the concept of reinforcement learning, an optimal access policy which maximizes the expected number of successful transmissions in the long run is derived via online learning. In [33], a multi-agent spectrum prediction approach is proposed for a multi-channel wireless network. Reinforcement learning has been applied to the problem by modeling the problem as a partially observable stochastic game with a continuous action space. The results can be fed into the network schedulers to avoid collisions with other neighbouring users.

Different parameters such as location, time, and frequency can be taken into account in studying the spectrum occupancy. The mixed effects of the aforementioned parameters were modeled as a linear regression in [34]. Three different frequency bands in the range of 88 MHz–3 GHz in five different locations in the USA have been chosen and studied in this work. Based on the results, the authors claimed that a good linear fit can be obtained using the spectrum measurement data.

The AR model has been applied to other datasets as well. For instance, the spectrum has been measured at four different locations in [35] with a perfect synchronization of the measuring devices. An AR model has been proposed and applied to the measured data in order to predict the binary time series. Akaike information criterion (AIC) and mean of the response residual magnitudes were considered as the criteria for determining the AR parameters. Also, a four-state Markov chain was used to store the last two bits corresponding to the occupancy states. It has been shown that the AR model can outperform the Markov chain type prediction.

In addition to the AR model, the ARMA approach can be used to predict time series more accurately by considering the current and various past values in a stochastic term using the moving-average (MA) model. In [36], the authors have studied different prediction methods including the ARMA model in content delivery networks. It has been shown that the ARMA model can improve the accuracy of the prediction. However, there is no ARMA model that can predict all contents accurately over the content's lifetime. In [14], Deng *et al.* have used the ARMA model in conjunction with Euclidean distance in order to model the nearest neighbor classification. They argued that the computational training cost of the RNN and HMM is much higher than the training cost of the ARMA model. The performance is promising according to the results. The ARMA model can be used in the spectrum occupancy prediction field as well. In [37], the frequency range of 100 MHz–2.4 GHz has been measured in an outdoor location for a week. The ARMA model has been applied to the measured data. The authors have shown that the fitted data show a good agreement with the measured one in terms of AIC.

We note that directly applying ARMA and LSTM models to the raw spectrum sensing dataset may result in poor accuracy. Different from the previous approaches, in this work, we investigate the features of the different channels in different frequency bands considering the typical types of their usage. Applying the domain knowledge, we study how to pre-process the data and then build reasonable and accurate ARMA and LSTM models that can accurately predict the channel occupancy of the channels experiencing a high temporal correlation. Furthermore, studying SCMs, especially based on the measurements, can provide us the pattern of the data which is useful in designing and examining the DSA systems.

C. Spectrum Consumption Model

SCM's goal is to capture the temporal, spatial, and spectral characteristics which specify how the spectrum is consumed by any transmitter or receiver. Total power, spectrum mask, location, schedule, etc. are several parameters in the SCM. In [15], how the SCMs can be exploited in spectrum sharing and management has been presented and discussed. The IEEE 1900.5.2 standard's preliminaries and ongoing efforts have been described in this work. An overview of the IEEE 1900.5.2 standard which defines the SCMs data model and devices' compatibility in spectrum systems has been provided in [1] by using example use cases. An open source tool for analysis and construction of the SCMs has been proposed in [38], where the SCMs' potential for spectrum sharing and DSA is highlighted.

III. Dataset Description

The raw dataset was collected at three sites in an urban area by a spectrum awareness system deployed by Communications Research Centre (CRC) in Ottawa, Canada, starting from June 1, 2016, to September 1, 2016 by a B200 USRP Ettus Research radio frequency front end. The coordinates of these sites are $45°22'10.0''$N $75°42'15.7''$W, $45°26'37.9''$N $75°38'53.8''$W, and $45°20'44.6''$N $75°53'02.0''$W. The system includes a network of sensors with high resolution in time and frequency for the measurements which are reported to a cloud-based platform for data processing and analysis [24]. The frequency range spans from 138–941 MHz. The measurements are channel and time-specific. The resolution bandwidth is 15 KHz and the channels are scanned every 300 ms. The scanned results, i.e., the received powers, have been converted to the histogram by counting the times the received power falling in each determined interval (-120 to -20 dBm, 1-dBm granularity). Different from many existing spectrum access works in the literature that focused on dynamic access of channel anytime by the SUs, in practice, spectrum sharing or leasing to the SUs will be arranged with much longer time duration to minimize the disturbance to the PUs and the spectrum management overhead. This is the reason that the histogram was stored **hourly** and converted to a probability mass function (PMF) for each channel.

According to the PMF, we can obtain two performance metrics. One is the average received power and the other is the probability of the off state, denoted as P_{off}. When the received power in a channel is lower than a threshold, it is defined as the off state. In this article, the threshold is set to -120 dBm, the lower-bound of the histogram. Regardless of the channel of each data file, the last 500 hours are separated as the test dataset to evaluate the performance of prediction (Section V), and the rest of the data is the training dataset. For modeling (Section VI), all the collected data are used as the training dataset.

The measurements mainly cover the LMR spectrum bands that are serving first responder organizations such as police, fire stations, ambulance services, and public works organizations such as utility companies. One typical traffic over LMR is from regular daily human usage, where transmissions occur in fixed locations such as office buildings and certain service areas. Therefore, the 300 ms scanning span can capture the channel behavior well. Considering the work pattern of service providers, the traffic density increases during the daytime and decreases in the night. Given the fixed location of the sensing device, this type of traffic generates more stable and predictable received power. Besides, LMR also includes dispatch services such as taxis, or companies with large vehicle fleets. When a mobile transmitter passes by the sensing device, the received power will increase first because of the reduced transmission distance, and then decrease as the transmitter leaves. It typically happens within one-hour duration and generates a sudden jump in the training/test dataset.

We select a channel within the range of 145–155 MHz as a representative, which is denoted as the type-A channel in this article. It contains regular daily usage and also has jumps in measurements. Its average received power is relatively low, which is promising for cognitive radio to reuse the spectrum. Other channels in 850–860 MHz have similar features too. Augmented Dickey Fuller (ADF) and Kwiatkowski-Phillips-Schmidt-Shin (KPSS) are two tests which determine the stationarity by defining a null or alternative hypothesis. We use both of the tests to check the stationarity of the dataset. Both the average received power and off-state probability are stationary with a confidence level of 95% due to channel behaviors, based on which the ARMA model is applied to the type-A channel in the following sections.

Another typical channel within the range of 460–470 MHz is selected for modeling and denoted as the type-B channel. In the type-B channel, the spectrum does not have a regular daily usage and the received power is below the sensing sensitivity for most of the time. Other channels in 159–169 MHz and 410–420 MHz also have the similar features. Because of its sparseness, we convert the raw dataset to an ON/OFF (1–0) sequence and then, model it as a non-homogeneous Markov chain in Section VI-C.

IV. Spectrum Prediction Models

A. ARMA Model and Parameters Estimation

An ARMA(p, q) model is a combination of AR(p) and MA(q) models and this model is fit for univariate time series [39]. The AR(p) part is responsible for modeling the future value by a linear combination of the past observations as well as a random error. On the other hand, MA(q) takes into account the past errors as explanatory variables. The ARMA model is given by

$$X_t - \sum_{k=1}^{p} a_k X_{t-k} = \epsilon_t - \sum_{s=1}^{q} m_s \epsilon_{t-s}, \tag{1}$$

where $t \geq \max(q, p)$. $X_i,\ i = t-p, \ldots, t$ and $\epsilon_j,\ j = t-q, \ldots, t$ are observed values and prediction errors at the i-th and j-th time slots, respectively. $a_i,\ i = 1, \ldots, p$ and $m_j,\ j = 1, \ldots, q$ are parameters of the AR and MA parts, respectively. In order to estimate the ARMA model's parameters, we apply the least square (LS) estimation where the squared difference

between the actual data and predicted one is minimized. The objective function of LS estimation is given by

$$\min_{a_1,\ldots,a_p,m_1,\ldots,m_q} \sum_{i=r}^{N} \epsilon_i^2, \tag{2}$$

where N is the size of the training dataset, and $r = \max(q, p)$. By rewriting (2) in a vector form, we have

$$\min_{\boldsymbol{\beta}} \boldsymbol{\epsilon}^T \boldsymbol{\epsilon} = (\boldsymbol{Y} - \boldsymbol{F}(\boldsymbol{\beta}))^T (\boldsymbol{Y} - \boldsymbol{F}(\boldsymbol{\beta})), \tag{3}$$

where

$$\boldsymbol{\beta} = \left[a_1, \ldots, a_p, \ -m_1, \ldots, -m_q\right]^T,$$

$$\boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_r \\ \epsilon_{r+1} \\ \vdots \\ \epsilon_N \end{bmatrix}, \ \boldsymbol{Y} = \begin{bmatrix} X_r \\ X_{r+1} \\ \vdots \\ X_N \end{bmatrix}, \ \boldsymbol{F}(\boldsymbol{\beta}) = \begin{bmatrix} f_r(\boldsymbol{\beta}) \\ f_{r+1}(\boldsymbol{\beta}) \\ \vdots \\ f_N(\boldsymbol{\beta}) \end{bmatrix},$$

$$f_i(\boldsymbol{\beta}) = \sum_{k=1}^{p} a_k X_{i-k} - \sum_{s=1}^{q} m_s \epsilon_{i-s}, \ i = r, \ldots, N.$$

Since the objective function in (3) is a non-linear LS problem, it is not straightforward to obtain a closed-form solution. Therefore, we apply the GN algorithm to approximate the optimal solution. In the GN algorithm, without a good initial point, the rugged gradient leads to step length fluctuation and makes it difficult to converge. In order to obtain a closer initial point to the optimal value, we first use the long AR model and estimate its parameters according to the LS method. Then we find the approximated initial point by equalization of the AR and ARMA models. The long AR model is given by making $m_j = 0$, $j = t - q, \ldots, t$ in (1) as follows,

$$X_t = l_1 X_{t-1} + l_2 X_{t-2} + \cdots + l_{p_L} X_{t-p_L} + \epsilon_t, \tag{4}$$

where $t \geq p_L$, and l_i, $i = 1, \ldots, p_L$, are parameters of the long AR model. Since the same parameter estimation method, i.e., LS estimation, is applied to obtain the parameters, the objective function is the same as (3), but $\boldsymbol{F}(\boldsymbol{\beta})$ is replaced by $\boldsymbol{Xl}$, where $\boldsymbol{l} = [l_1, \ldots, l_{p_L}]^T$. By taking derivative of the objective function w.r.t. $\boldsymbol{l}$ and making it equal to zero, we have

$$\frac{\partial (\boldsymbol{Y} - \boldsymbol{Xl})^T (\boldsymbol{Y} - \boldsymbol{Xl})}{\partial \boldsymbol{l}} = -2\boldsymbol{Y}^T \boldsymbol{X} + 2\boldsymbol{l}^T \boldsymbol{X}^T \boldsymbol{X} = 0$$

$$\Longrightarrow \boldsymbol{l} = \left(\boldsymbol{X}^T \boldsymbol{X}\right)^{-1} \boldsymbol{X}^T \boldsymbol{Y}. \tag{5}$$

With the back-shift operator, B, the long AR model in (4) and ARMA model in (1), can be rewritten as (6) and (7), respectively.

$$\mathcal{I}_1(B) = \left(1 - l_1 B - l_2 B^2 - \cdots - l_{p_L} B^{p_L}\right) X_t = \epsilon_t, \tag{6}$$

$$\mathcal{I}_2(B) = \frac{\left(1 - a_1 B - a_2 B^2 - \cdots - a_p B^p\right)}{\left(1 - m_1 B - m_2 B^2 - \cdots - m_q B^q\right)} X_t = \epsilon_t. \tag{7}$$

Given that the inverse functions, $\mathcal{I}(\cdot)$, in (6) and (7) are equivalent, we can obtain the parameters of the ARMA model by solving $\mathcal{I}_1(B) = \mathcal{I}_2(B)$ [40]. By comparing operator B's exponents on the both sides of the equation $\mathcal{I}_1(B) = \mathcal{I}_2(B)$ and writing them in matrix form, we have

TABLE I
PARAMETERS' DEFINITION

Parameter	Description
X_i	Observed value at the i-th time slot
ϵ_j	Prediction error at the j-th time slot
a	AR parameters
m	MA parameters
σ_p^2	Variance prediction error in the test dataset
σ_ϵ^2	Variance prediction error in the training dataset
N	Size of the training dataset
p	Number of parameters in the AR part
q	Number of parameters in the MA part
P_{off}	off-state probability
ReLU	LSTM activation function
Adam	LSTM optimization algorithm

$$\begin{bmatrix} m_1 \\ m_2 \\ \vdots \\ m_q \end{bmatrix} = \begin{bmatrix} l_p & l_{p-1} & \cdots & l_{p+1-q} \\ l_{p+1} & l_p & \cdots & l_{p+2-q} \\ \vdots & \vdots & \ddots & \vdots \\ l_{p+q-1} & l_{p+q-2} & \cdots & l_p \end{bmatrix}^{-1} \begin{bmatrix} l_{p+1} \\ l_{p+2} \\ \vdots \\ l_{p+q} \end{bmatrix}, \tag{8}$$

$$\begin{bmatrix} a_1 \\ a_2 \\ a_3 \\ \vdots \\ a_p \end{bmatrix} = \begin{bmatrix} m_1 \\ m_2 \\ m_3 \\ \vdots \\ m_p \end{bmatrix} + \begin{bmatrix} 1 & 0 & \cdots & 0 \\ -m_1 & 1 & \cdots & 0 \\ -m_2 & -m_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -m_{p-1} & -m_{p-2} & \cdots & 1 \end{bmatrix} \begin{bmatrix} l_1 \\ l_2 \\ l_3 \\ \vdots \\ l_p \end{bmatrix}, \tag{9}$$

where $m_i = 0$ for $i > q$, and we set $p_L = p + q$. Given $\boldsymbol{l}$ in (5), a_i, $i = 1, \ldots, p$ and m_j, $j = 1, \ldots, q$ can be obtained by solving (8) and (9). We consider parameters as a vector the same as (3) and denote it as $\boldsymbol{\beta}_0$. The calculated $\boldsymbol{\beta}_0$ is considered as the initial point of the GN algorithm.

Given $\boldsymbol{\beta}_k$, the parameter vector in the k-th iteration of the GN algorithm, prediction errors are obtained by

$$\epsilon_i = X_i - \left[X_{i-1} \ \cdots \ X_{i-p} \quad \epsilon_{i-1} \ \cdots \ \epsilon_{i-q}\right] \boldsymbol{\beta}_k, \tag{10}$$

where $X_i = \epsilon_i = a_i = m_i = 0$ for $i < 0$. We apply the following GN updating equation to approximate the optimal solution recursively until all of the parameters are stabilized, i.e., the difference of each parameter between two adjacent iterations is smaller than a threshold [41].

$$\boldsymbol{\beta}_{k+1} = \boldsymbol{\beta}_k + \left(\nabla \boldsymbol{F}(\boldsymbol{\beta}_k)^T \nabla \boldsymbol{F}(\boldsymbol{\beta}_k)\right)^{-1} \nabla \boldsymbol{F}(\boldsymbol{\beta}_k)^T \times [\boldsymbol{Y} - \boldsymbol{F}(\boldsymbol{\beta}_k)]. \tag{11}$$

Table I summarizes a few important notations in this article for convenient reference.

B. LSTM Framework Implementation

The LSTM framework is implemented in Python, and Keras library [42] is used to build the LSTM network architecture. Keras is a high-level neural network programming interface that provides powerful building blocks for deep learning networks. Stacked LSTM layers are exploited for the spectrum prediction, which consists of four hidden layers. The first two and the last two layers have 80 and 50 LSTM cells,

respectively. The activation function and the optimization algorithm that are used in the training phase are *ReLU* and *Adam*, respectively. The *Adam* optimization algorithm [43] is an adaptive learning rate optimization algorithm designed for training deep neural networks. It computes adaptive learning rates for different parameters by estimation of the first and the second moments of the gradients. We split the dataset into two parts. 85% of the dataset is used for training the model, and the remaining will be utilized for testing the model's performance. The time step and the batch size of the model are set to 15 and 64, respectively. The model is trained for about 400 iterations. The optimization algorithm's parameters including the learning parameter, β_1, β_2, and ϵ are 0.001, 0.9, 0.999, and 10^{-8}, respectively, with an adaptive learning rate. A dropout of 0.5 was applied to the first two hidden layers of the model to avoid over-fitting. Mean squared error is chosen as the loss function of the model. In order to predict the received power in the current time slot, the focus is on the previous time slots whose total number is equal to the time step parameter mentioned earlier. For a certain time step, the prediction model is trained in a way that minimizes the mean squared error of the model's output and the current time slot's received power.

Generally, the LSTM model has more parameters than the ARMA one. The total number of parameters in the LSTM model is $N_p = 4 \times n_c \times n_c + 4 \times n_i \times n_c + n_c \times n_o + 3 \times n_c$ where n_c, n_i, and n_o are the numbers of memory cells, input units, and output units, respectively. Therefore, the computational complexity of learning the LSTM model per time step is proportional to the number of parameters, i.e., $O(N_p)$ [44], [45]. Accordingly, as it was mentioned in Section I, the ARMA model parameters are easier to find. In order to have a rough idea on the complexity of these two models, we can compare their running time. We run both of the models on a computer with 3 GB RAM and a 3.40 GHz Intel Core i7–3770 CPU. The training and prediction phases of the LSTM model take around 1400 s and 2 s, respectively, while the running time of the ARMA model is 16 s.

V. Spectrum Prediction Results

We apply the ARMA model and LSTM learning method to the spectrum data and compare the results in this section.

A. Average Received Power

We apply the aforementioned method to the training dataset of the type-A channel in order to estimate the parameters of the ARMA model. Given certain p and q, parameters and corresponding prediction errors can be obtained by (10) and (11). Denoting the variance of the prediction error in the training dataset as $\sigma_\epsilon^2(p, q)$, the Bayesian information criterion (BIC) is defined as

$$\mathrm{BIC}(p, q) = N \ln\{\sigma_\epsilon^2(p, q)\} + (p + q)\ln(N), \qquad (12)$$

where N is the size of the training dataset. The best p and q are selected when BIC(p, q) is minimized. In BIC, a function of the posterior probability of a model truthfulness is estimated. Obviously, a model with a lower BIC means that it can describe the original data better [46]. Fig. 2 shows the different values of BIC corresponding to different values of p and q. As it can be observed, the minimum BIC is achieved when $p = q = 2$, i.e., the model uses the predicted power values and the corresponding errors of the last two hours, and the corresponding LS estimation of the ARMA model is

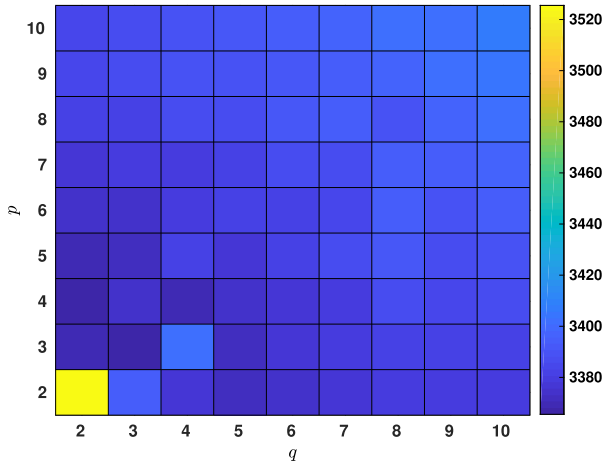


Fig. 2. BIC metric to determine (p, q).

$$X_t - 1.5529X_{t-1} + 0.5529X_{t-2} = \epsilon_t - 1.2202\epsilon_{t-1} + 0.2222\epsilon_{t-2}. \qquad (13)$$

Figs. 3 (a) and (b) show the fitting results of the above ARMA model and the LSTM method inside the training set and the prediction results in the test set, respectively. We use the variance of the prediction error, σ_p^2, and the mean absolute percentage error (MAPE) as the metrics to evaluate the prediction performance in the test dataset. The MAPE's formula is

$$\mathrm{MAPE} = 100 \times \frac{1}{N}\sum_{t=1}^{N}\left|\frac{y_t - \hat{y}_t}{y_t}\right|, \qquad (14)$$

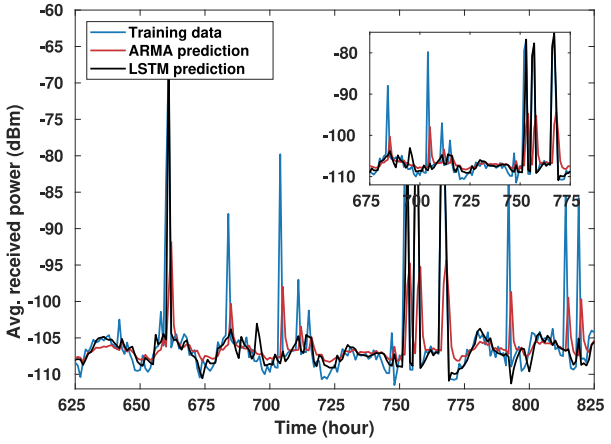
where y_t, $\hat{y}_t$, and N are the real data, the predicted one, and its length, respectively. The results of different methods are summarized in Table II.

As shown in Fig. 3, neither of the ARMA model nor LSTM fits the training dataset very well. Further increasing (p, q) in the ARMA model cannot substantially reduce $\sigma_\epsilon^2(p, q)$, and thus results in a relatively low ARMA order according to the minimum BIC and a relatively high σ_p^2. We also applied the AIC to determine the order [37]. The same result, i.e., (2, 2), is obtained in this case. The training dataset contains many sudden jumps that may be caused by mobile users as mentioned in Section III, which are denoted as bursts in this article. A burst typically happens in an hour and disappears in the next one or two hours. Besides, there is no obvious pattern observed for the occurrence of bursts, which means that they have weak temporal correlations with the other data. Therefore, it is difficult for the ARMA and LSTM models to predict them.

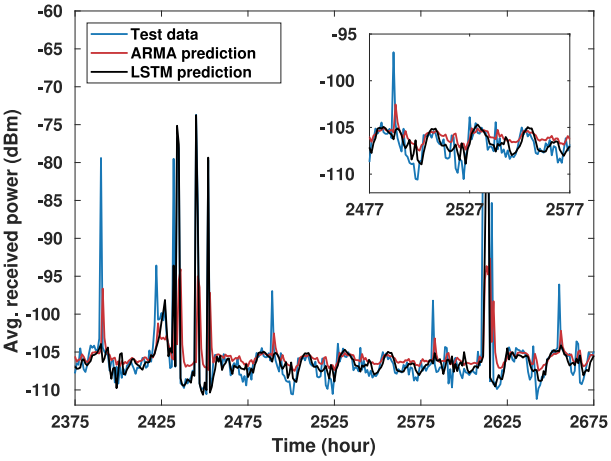
Pre-Processing the Dataset by an LP Filter: A burst is similar to an impulse signal in the time domain, which includes lots of high-frequency components in the power spectrum. It

TABLE II
ERROR COMPARISON OF DIFFERENT PREDICTION METHODS

Method	ARMA [19]		LSTM		ARMA + Pre-processing		LSTM + Pre-processing	
Metric	MAPE	σ_p^2	MAPE	σ_p^2	MAPE	σ_p^2	MAPE	σ_p^2
Power prediction (regular days)	2.32%	24.19	1.4%	10.30	1.37%	8.054	1.45%	8.064
Power prediction (working days)	1.3%	7.43	1.2%	6.26	0.82%	2.97	0.9%	2.77
P_{off} prediction (regular days)	0.76%	1.05×10^{-4}	0.47%	4.53×10^{-5}	0.29%	3.62×10^{-5}	0.35%	2.4×10^{-5}
P_{off} prediction (working days)	0.61%	6.26×10^{-5}	0.49%	3.92×10^{-5}	0.28%	4.43×10^{-5}	0.36%	2.7×10^{-5}



(a) Training dataset



(b) Test dataset

Fig. 3. One-hour ahead prediction using ARMA (2,2) and LSTM.

inspires us to apply a low-pass filter to remove high-frequency components and increase temporal correlation for the training dataset. The filter's cutoff frequency is a trade-off between temporal correlation and the power of error residue, i.e., the difference between the filtered and unfiltered data. From the testing results, 0.5π is chosen as the cutoff frequency. This frequency was examined for each month separately. The results showed that it is stable and a proper setting for the LP filter.

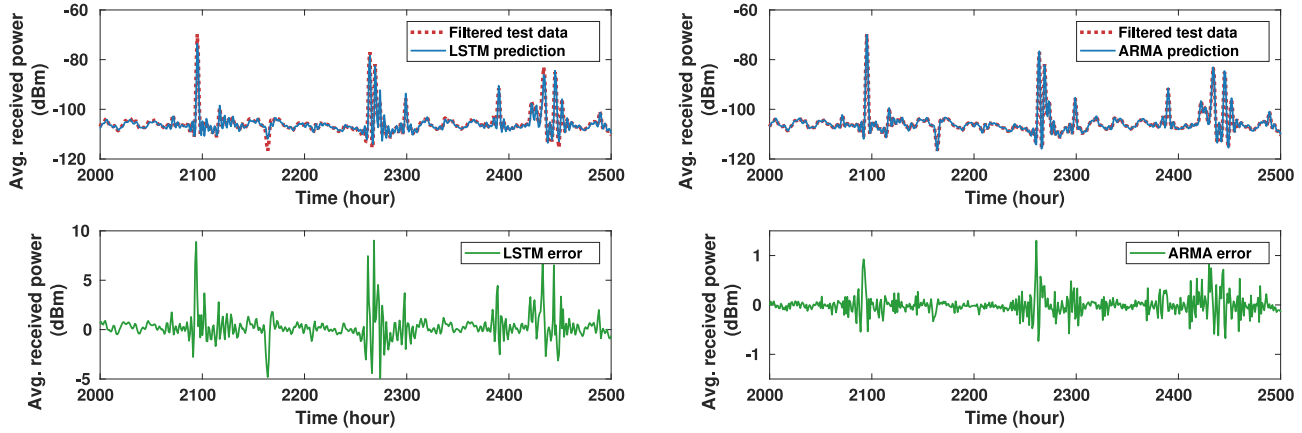
The results from applying the ARMA and LSTM models to the dataset of regular and working days are shown in Figs. 4 and 5. The ARMA model can receive a better fitting result on the filtered data as shown in Fig. 4 (b). Unlike the best order in the unfiltered case, i.e., $(p, q) = (2, 2)$, as shown in Fig. 2, further increasing the order of the ARMA model can greatly reduce $\sigma_\epsilon^2(p, q)$ for the filtered data and thus, results in a lower BIC. For the filtered average received power data of the regular days, the variance of the prediction error without considering the processing error is $\sigma_\epsilon^2(10, 9) = 0.094$, while the corresponding error variance is 1.175 for the LSTM one.

To have a fair comparison, the error residue from the LP filter should be further combined to obtain the overall prediction error of the original data. The variance of the prediction error and the MAPE are 8.054 and 1.37%, respectively. Comparing with the ARMA-only model, an appropriate LP filter can significantly improve the precision of the prediction. The variance of the prediction error and the MAPE for the data of regular days are 1.45% and 8.064, respectively. Obviously, the ARMA model can achieve a better result in comparison to the sophisticated LSTM learning approach.

Besides, we examine the training dataset day by day and find that the weekend data is slightly different from the working-day one. This phenomenon can be observed more clearly in Fig. 7, where an obvious low-frequency component with the period of one week exists. The highest frequency of the data in the Fig. 7 corresponds to $F_s/2$, where $F_s = 1$ hour^{-1}. The dominant frequencies in the spectrum happen roughly on $\{\pi/6, \pi/12, \pi/28, \pi/42, \pi/82\}$ which correspond to periods of {0.5, 1, 2.3, 3.5, 6.83} days, respectively. We apply the same method, i.e., ARMA and LSTM combined with the LP filter, to the training and test datasets containing the data of working days only. The results are shown in Fig. 5. The average received power prediction error variance of the working days is 0.195 and 0.302 for ARMA and LSTM, respectively. The pre-processing based on the a priori information, such as holidays, special events, and user behaviors, can substantially improve the accuracy. Considering both prediction and processing errors, the power prediction of working days results in a smaller error residue and a better fitting in comparison to the regular days power prediction. In this case, $\sigma_p^2 = 2.97$, and the MAPE of prediction is 0.82% only. The same metrics are $\sigma_p^2 = 2.77$ and 0.9% for the LSTM learning method.

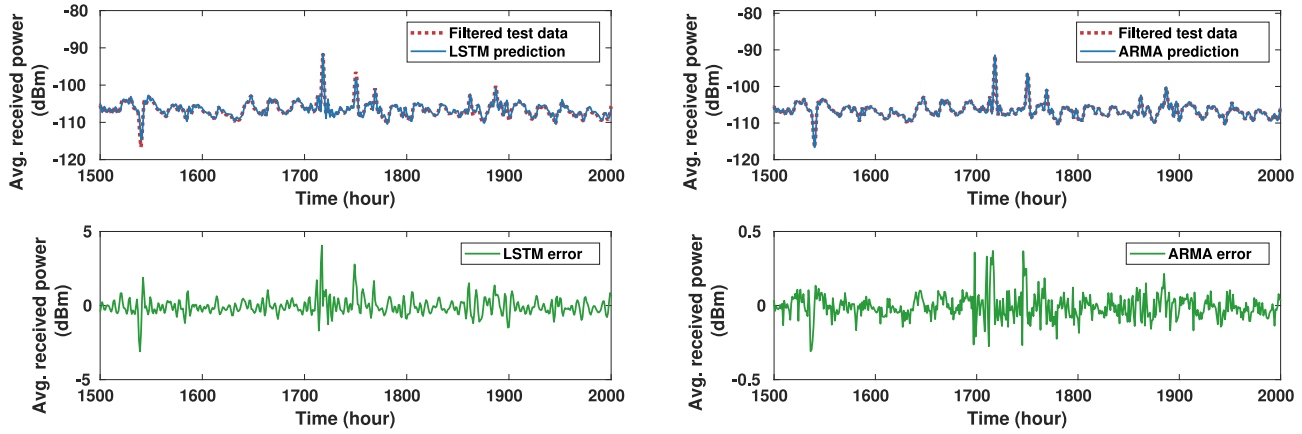
B. Off-State Probability (P_{off})

The same methods are applied to P_{off} as well. The one-hour ahead prediction results for regular days only are shown



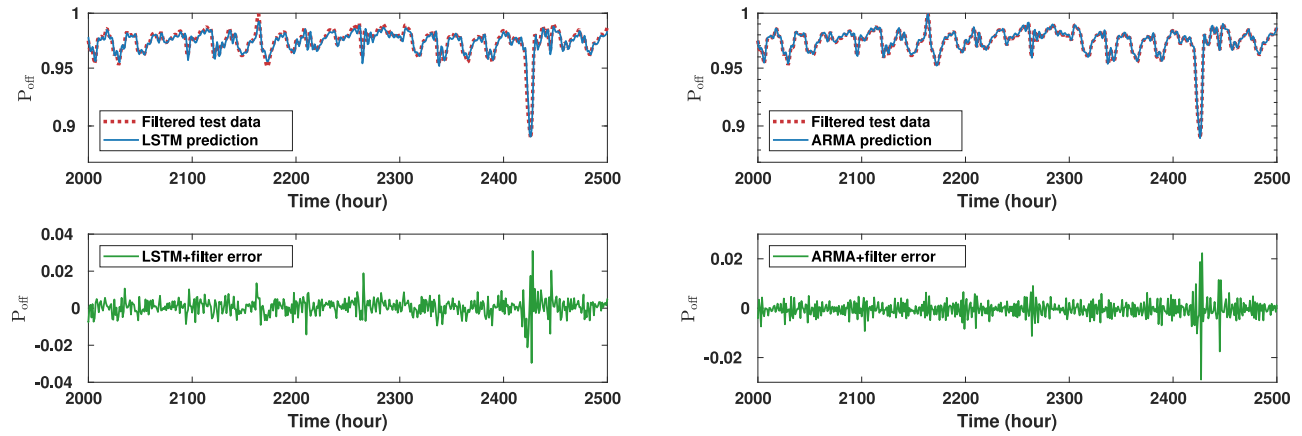
(a) LSTM prediction (b) ARMA(10,9) prediction

Fig. 4. Prediction of the average received power by applying ARMA(10,9) and LSTM combined with a low-pass filter to the dataset of regular days.



(a) LSTM prediction (b) ARMA(9,8) prediction

Fig. 5. Prediction of the average received power by applying ARMA(9,8) and LSTM combined with a low-pass filter to the dataset of working days.



(a) LSTM prediction (b) ARMA(10,10) prediction

Fig. 6. Prediction of P_{off} by applying ARMA(10,10) and LSTM combined with a low-pass filter to the dataset of regular days.

in Fig. 6. In this case, $\sigma_p^2 = 3.62 \times 10^{-5}$, and the MAPE of prediction is 0.29%. The LSTM learning method gives roughly similar results as $\sigma_p^2 = 2.4 \times 10^{-5}$ and 0.35%. We repeated the tests for other type-A channels, such as those in 850–860 MHz band, and the results are similar. Due to the space limit, we do not report them here.

The error comparisons between different approaches are summarized in Table II. It can be concluded from the results that the LSTM method outperforms the pure ARMA model for both power and P_{off} prediction, while the ARMA has a slightly better performance on the pre-processed data in terms of accuracy. The MAPE and total error variance are

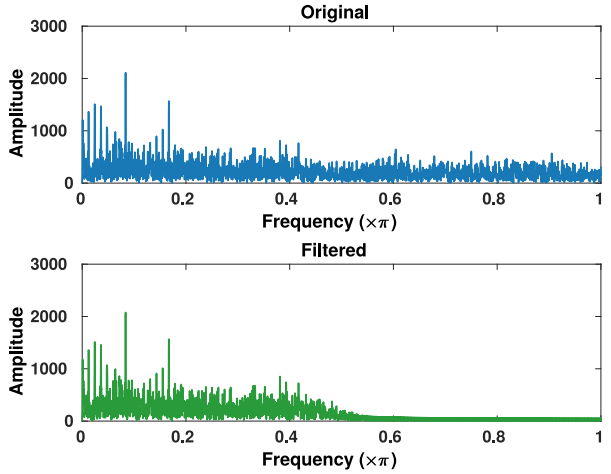


Fig. 7. Spectrum of the dataset of regular days.

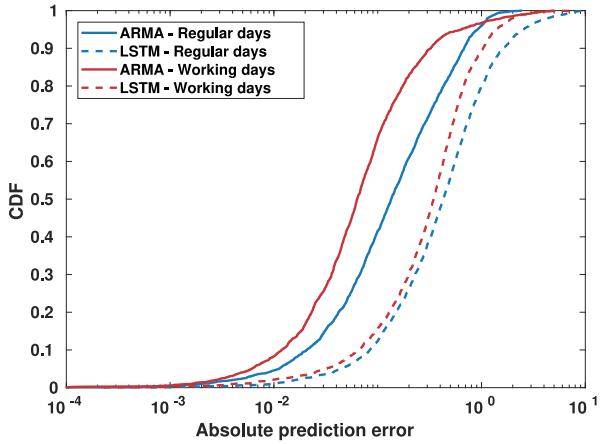


Fig. 8. CDF of the absolute prediction error for the dataset.

smaller for the power prediction by deploying the ARMA model. According to the results presented in Figs. 4 and 5, the LSTM cannot predict the bursts in the test dataset, while ARMA can predict with a better precision. The cumulative distribution functions (CDF) of the absolute prediction error of the regular and working days are shown in Fig. 8. The probability of a smaller prediction error is higher in the ARMA model compared to the LSTM one both for regular and working days.

VI. Statistical Spectrum Modeling

A. Average Received Power for Type-A Channels

In addition to prediction, it is desirable to have a model to generate time sequences having similar statistical characteristics to the training dataset of the working days. The ARMA and LSTM models may capture the temporal correlation, but cannot reconstruct periodic and burst components, so they are insufficient for the modeling purpose. Therefore, we extract and model them separately.

In Fig. 7 (a), it can be seen that the major periodic components are at one-day period ($\frac{\pi}{12}$) and its doubled frequency ($\frac{\pi}{6}$). We use two sine functions to model them. The amplitudes, initial phases, and frequencies of sine functions are obtained via fast Fourier transform (FFT).

The i-th measurement in the dataset is denoted as x_i whose unit is dBm. If $|x_i - E(x)| > 2\sigma_x$, x_i is considered as a burst, where $E(x)$ and σ_x are the expectation and standard deviation of the dataset, respectively. This threshold is an empirical parameter, by which bursts are identified as shown in Fig. 9(a). Given that the number of mobile users is large and the probability of any given user approaching the sensing device is low, we use a Poisson process to model the bursts arrivals. The interval between two adjacent bursts should follow the exponential distribution. The fitting result for $\lambda = 19.25$ is shown in Fig. 9 (b). According to the shape of the PDF curve, gamma distribution is used to model the amplitude of the bursts. Fig. 9 (c) shows the results and its tuned parameters.

We remove the fitted sine functions from the training data and use $E(x)$ to replace all the bursts. Then, an LP filter with a cutoff frequency of 0.2π is applied. The cutoff frequency is selected based on the FFT of the training data in Fig. 10(d). After removing two impulses of the fitted sine functions, the spectrum is similar to the spectrum of white noise except for the low-frequency part ($< 0.2\pi$). The ARMA model is used only for capturing the temporal correlation of the low-frequency components. We apply a high-pass filter ($> 0.2\pi$) to the generated white noise to model the high-frequency components. All of the model related parameters are summarized in Appendix A.

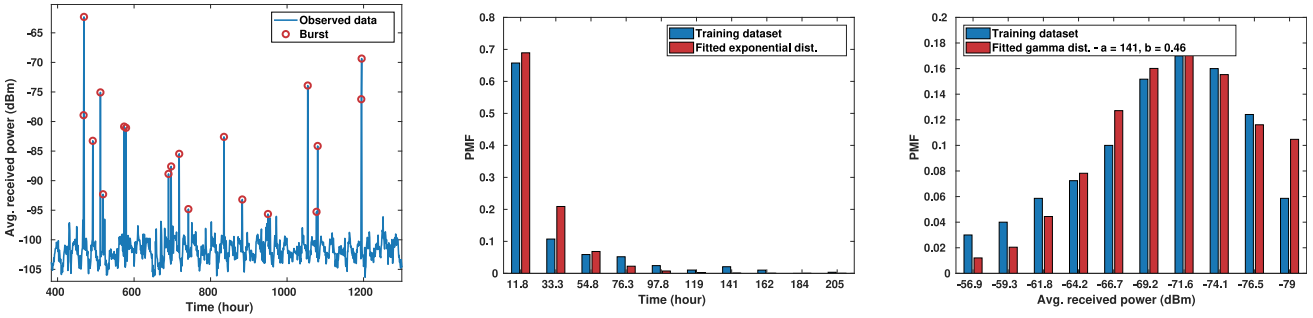
The comparison between the model-generated data and the training data from different points of view is then made as follows. We compare the PDFs in Fig. 10 (a). The model-generated data are consistent with the training data. ACF and PACF are used to depict the temporal correlation which are shown in Fig. 10 (b) and (c). Besides, a similar FFT is obtained as well, where the major periodic components are well preserved in the model-generated data as shown in Fig. 10 (d). Fig. 10 (e) shows a comparison in the time domain. The similarities in terms of period, amplitude, and bursts are illustrated. Similar patterns are observed in the model-generated data and the training data. It shows that the model can reconstruct the temporal correlation of the training data.

B. Off-State Probability (P_{off}) for Type-A Channels

For the training dataset in terms of P_{off} (working days only), the same modeling is applied. In Fig. 11, the results show that the model-generated data are consistent with the training data in terms of PDF, temporal correlation, and periodicity. The occurrences of bursts are reconstructed as well. All of the model related parameters are summarized in Appendix B. The proposed modeling approach was applied to other type-A channels, and similar results were achieved.

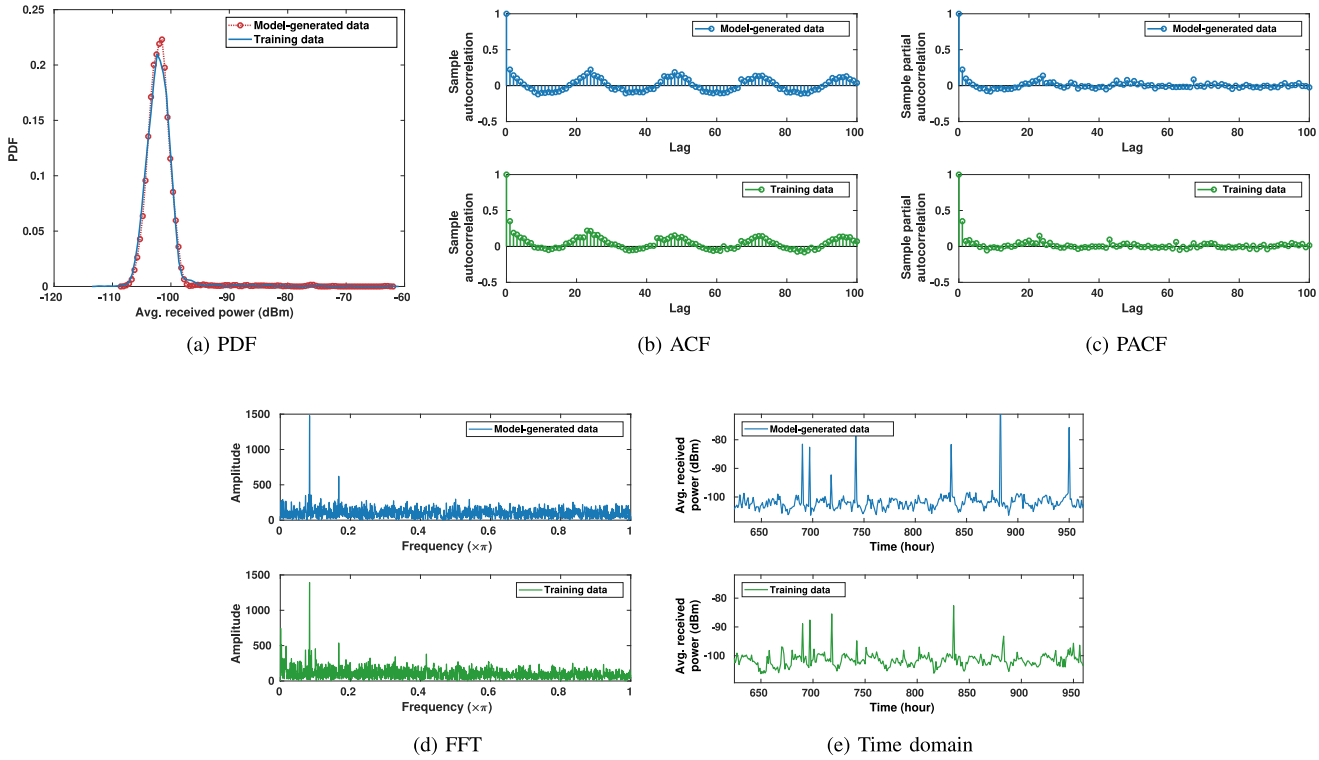
C. Non-Homogeneous Markov Model for Type-B Channels

Different from type-A channels, $P_{off} = 1$ for most of the time in type-B channels. Thus, we convert the dataset (working



(a) Burst (b) Burst modeling-interval (c) Burst modeling-amplitude

Fig. 9. Burst parameters of the dataset of working days.



(a) PDF (b) ACF (c) PACF

(d) FFT (e) Time domain

Fig. 10. Comparisons between the model-generated data and the training data on the average received power.

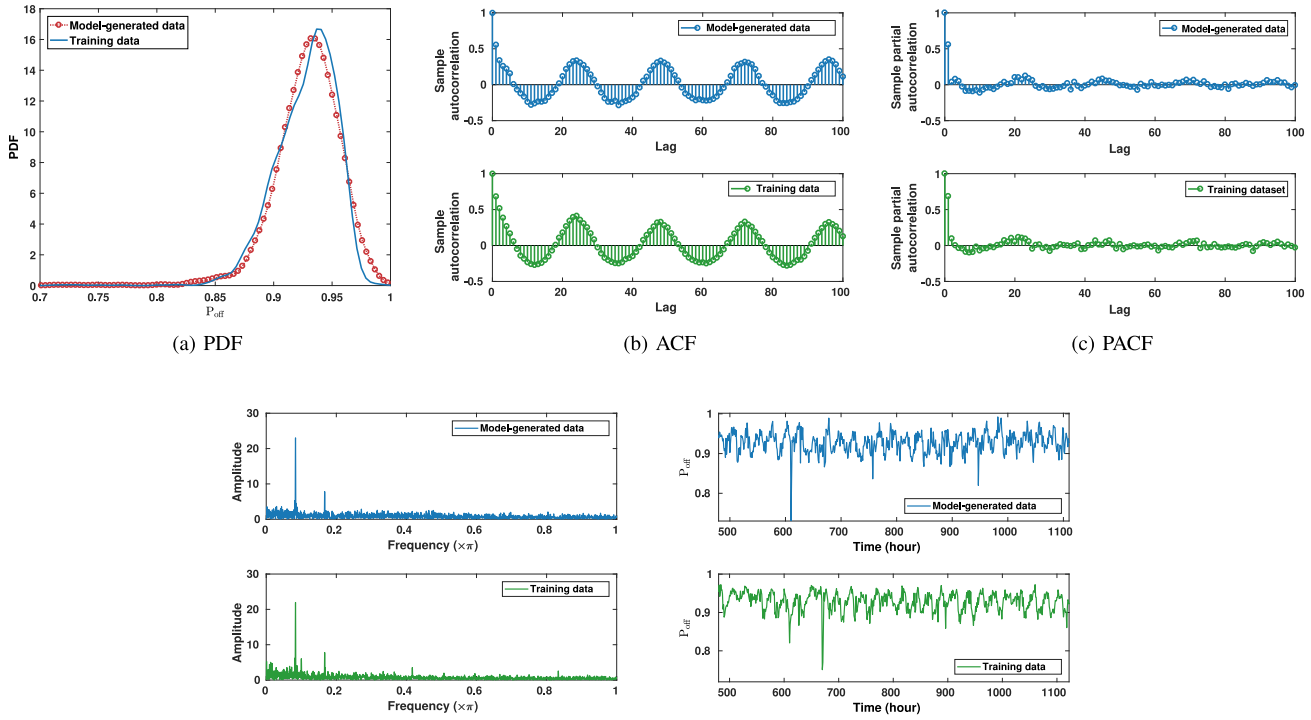
days only) into an ON/OFF (1–0) sequence by rounding them using the following indicator function.

$$\mathbf{1}\big(P_{\mathit{off}}(t)\big) = \begin{cases} 1, & P_{\mathit{off}}(t) < 1, \\ 0, & \text{otherwise}. \end{cases} \tag{15}$$

We apply a two-state discrete-time non-homogeneous (time-varying) Markov chain to model this ON/OFF training data as shown in Fig. 13. At each hour, the system may either transfer to another state or stay in the current one.

Obvious periodic components, e.g., the 24-hour period, can be found in the FFT of the training data (1–0 sequence) too, so we use time-varying transition probabilities, i.e., $P_{01}(t), P_{10}(t)$, to capture the temporal correlation and periodicity. Because the training data has been collected from 6 am in each day, we define $P_{10}(t) = P$[**off** at (t + 6)am/pm|**on** at (t + 5)am/pm], t $= 1, \ldots, 12$. $P_{01}(t)$ is defined in a similar way. In total, we can obtain 24 sets of transition probabilities. The arrival and departure probabilities at the same hour for each day remain the same, by which the periodicity is reconstructed. The transition probabilities are estimated based on the training data. Detailed parameters are summarized in Appendix C.

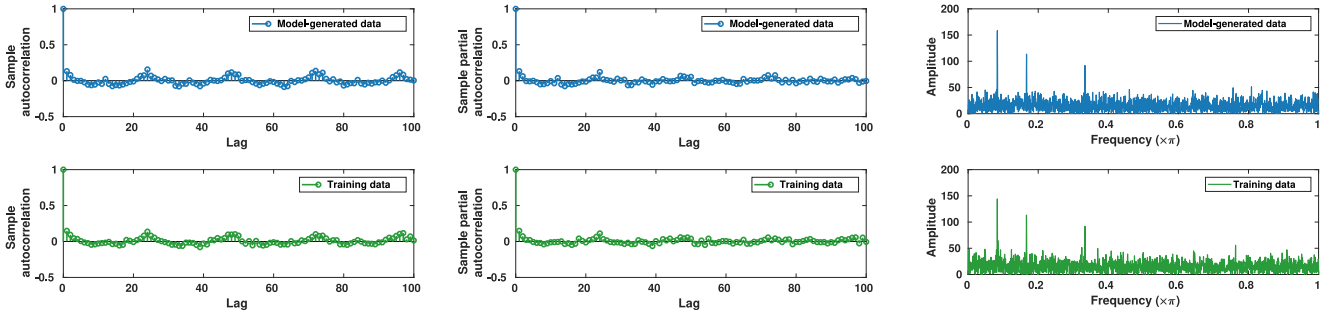
We generate a 1–0 sequence by using the above model and compare it with the training data. In Figs. 12 (a) and (b), similar patterns are reconstructed in both ACF and PACF. The periodic components are also well preserved as shown in Fig. 12 (c). The duty ratio of the model-generated data is 25.93%, while that of the training data is 25.88%, which are statistically similar. The modeling was repeated for other type-B channels, and the results were similar.



(a) PDF (b) ACF (c) PACF

(d) FFT (e) Time domain

Fig. 11. Comparisons between the model-generated data and the training data on P_{off}.



(a) ACF (b) PACF (c) FFT

Fig. 12. Comparisons between the model-generated data (1–0 sequence) and the training dataset.

VII. Conclusion

In this article, we focused on the low-level abstracted measured data and investigated the occupancy of the representative frequency bands to utilize the white spaces using the DSA systems. For the prediction of the average received power, we applied the model-based ARMA approach and the model-free LSTM learning method to the dataset. Because of the weak temporal correlation of the burst which may mislead the parameter estimation process, we used an LP filter to remove the high-frequency components to focus on the temporal correlation of the training data. We applied both of the aforementioned methods to the pre-processed data, and the simulation results show that a higher precision of the prediction can be achieved. A priori information of the dataset has taken into account as well. The accuracy can be further improved if only the dataset of working day is used. High-precision prediction is achieved for P_{off} by applying the same methods. The results from the ARMA model were compared to the results from the LSTM one. The ARMA model can achieve higher accuracy than LSTM in the burst prediction. However, the overall accuracy of these two methods is quite similar, and the ARMA model's parameters can be trained faster with the benefit of less complexity.

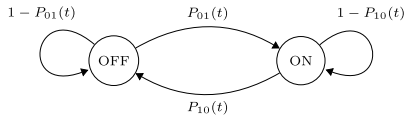


Fig. 13. Two-state non-homogeneous Markov chain.

Furthermore, we provided an SCM depicting the spectrum occupancy for designing and examining the DSA systems. Based on the nature of the training dataset, we have extracted the periodic, bursts, and aperiodic low-frequency components

from the data and modeled them separately. We examined the model-generated data from different perspectives, including the time/frequency domains, PDF, and (P)ACF. Results show that the model-generated data can approximate the measured data well in all of the above aspects. An ON/OFF sequence was extracted from the channel with sparse occupancy and then, was modeled by a non-homogeneous Markov chain. High similarities to the measured data have been achieved according to the results, which is substantial for DSA systems.

There are many open issues beckoning for further investigation. In this article, we predict and model two typical types of channels. How to deal with the occupancy for possibly non-stationary channels waits for further exploration, possibly using other tools such as autoregressive-integrated-moving-average or HMM. Given that different types of channels may use different models, a simple method guiding users to select an appropriate model is desirable yet an open issue. Bursty channels are non-predictable given the current approach due to low temporal correlation. How to apply other domain knowledge to predict bursty events is an open issue. For instance, the behaviors of licensees may facilitate the prediction. Overall, the proposed prediction approach, i.e., considering the usage of each channel to pre-process the training data to enhance the temporal correlation for more accurate spectrum prediction, points to a promising direction. The proposed simple yet effective spectrum occupancy models based on non-homogeneous Markov chains provide a powerful tool for both analysis and simulation of DSA systems. Moreover, the spatial correlation of the collected data on different locations can be taken into account by machine-learning networks such as CNNs to improve the prediction accuracy.

APPENDIX A
AVERAGE RECEIVED POWER MODELING PARAMETERS

The model is presented as follows,

$$X_t^{(\mathrm{mod})}(\mathrm{dBm}) = E(x) + c_1 \sin\left(\frac{\pi t}{12} + \phi_1\right) + c_2 \sin\left(\frac{\pi t}{6} + \phi_2\right) + X_t^{(\mathrm{a})} + n_t^{(\mathrm{hp})} + B_t, \tag{16}$$

where $E(x)$ is the mean value of the training data, and $E(x) = -102.17$ dBm. For two sine functions, $c_1 = 1.4$, $c_2 = 0.5$, $\phi_1 = 4.41$, and $\phi_2 = 2.67$. $X_t^{(\mathrm{a})}$ is the output of an ARMA model as shown below.

$$X_t^{(\mathrm{a})} = \left[X_{t-1}^{(\mathrm{a})} \ \cdots \ X_{t-7}^{(\mathrm{a})} \ \ \epsilon_t \ \ \epsilon_{t-1} \ \cdots \ \epsilon_{t-4}\right]\boldsymbol{\beta'},$$
$$\boldsymbol{\beta'} = [5.84 \ -15.28 \ 23.17 \ -21.95 \ 13.01 \ -4.46 \ 0.69 \ 1 \ -1.62 \ 1.86 \ -0.90 \ 0.31]^T. \tag{17}$$

In the above ARMA model, ϵ_t follows an independent and identical normal distribution $N(0, 0.0034)$ for all t. $n_t^{(\mathrm{hp})}$ is a white noise n_0, $n_0 \sim N(0, 0.96)$, filtered by a high-pass filter with the cutoff frequency of 0.2π. We use MATLAB to generate the FIR filter. B_t is the burst at t-th time slot. Its arrival and amplitude follow Poisson and Gamma distributions, as shown in Figs. 9 (b) and (c), respectively.

APPENDIX B
PARAMETERS FOR MODELING P_{off}

The model for P_{off} adopts the same expression as shown in (16), but with different parameters which are summarized as follows, $E(x) = 0.927$, $c_1 = 0.021, c_2 = 0.006, \phi_1 = 1.397, \phi_2 = 5.591$. $X_t^{(\mathrm{a})}$ is given by

$$X_t^{(\mathrm{a})} = \left[X_{t-1}^{(\mathrm{a})} \ \cdots \ X_{t-10}^{(\mathrm{a})} \ \ \epsilon_t \ \ \epsilon_{t-1} \ \cdots \ \epsilon_{t-6}\right]\boldsymbol{\beta'},$$
$$\boldsymbol{\beta'} = [4.71 \ -9.26 \ 9.12 \ -4.39 \ 2.62 \ -6.34 \ 8.95 \ -6.46 \ 2.40 \ -0.37 \ 1 \ -0.50 \ 0.58 \ 0.43 \ 0.79 \ -0.58 \ 0.76]^T, \tag{18}$$

where ϵ_t follows an independent and identical normal distribution $N(0, 3.70 \times 10^{-5})$ for all t. $n_t^{(\mathrm{hp})}$ is a white noise n_0, $n_0 \sim N(0, 0.011)$, filtered by a high-pass filter with the cutoff frequency of 0.2π. Both the interval and the amplitude of bursts follow exponential distribution, with λ equal to 31.62 and 0.87, respectively.

APPENDIX C
PARAMETERS FOR ON/OFF MODEL

The transition probabilities are given by

$$[P_{01}(1), \ldots, P_{01}(24)] = [12 \ 0 \ 1 \ 17 \ 46 \ 26 \ 19 \ 26 \ 08 \ 21 \ 26 \ 38 \ 22 \ 30 \ 30 \ 36 \ 55 \ 63 \ 37 \ 20 \ 17 \ 7 \ 9 \ 7] \times 10^{-2},$$
$$[P_{10}(1), \ldots, P_{10}(24)] = [90 \ 80 \ 100 \ 100 \ 36 \ 64 \ 77 \ 82 \ 76 \ 60 \ 75 \ 50 \ 60 \ 48 \ 58 \ 45 \ 44 \ 62 \ 52 \ 72 \ 75 \ 81 \ 88 \ 81] \times 10^{-2}.$$

ACKNOWLEDGMENT

The authors would like to thank the editor, the anonymous reviewers, Dr. Li Li, and Dr. Humphrey Rutagemwa from CRC for their helpful and constructive comments, which have helped them improve the manuscript significantly. They would also like to thank CRC for providing the dataset.

REFERENCES

[1] C. E. C. Bastidas *et al.*, "IEEE 1900.5.2: Standard method for modeling spectrum consumption: Introduction and use cases," *IEEE Commun. Stand. Mag.*, vol. 2, no. 4, pp. 49–55, Dec. 2018.

[2] H. Mosavat-Jahromi, Y. Li, and L. Cai, "A throughput fairness-based grouping strategy for dense IEEE 802.11ah networks," in *Proc. IEEE 30th Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Istanbul, Turkey, 2019, pp. 1–6.

[3] M. M. Sohul, M. Yao, T. Yang, and J. H. Reed, "Spectrum access system for the citizen broadband radio service," *IEEE Commun. Mag.*, vol. 53, no. 7, pp. 18–25, Jul. 2015.

[4] S. W. Ellingson, "Spectral occupancy at VHF: Implications for frequency-agile cognitive radios," in *Proc. IEEE 62nd Veh. Technol. Conf.*, vol. 2. Dallas, TX, USA, Sep. 2005, pp. 1379–1382.

[5] W. Cui, C. Liu, H. Mosavat-Jahromi, and L. Cai, "SigMix: Decoding superimposed signals for IoT," *IEEE Internet Things J.*, vol. 7, no. 4, pp. 3026–3040, Apr. 2020.

[6] H. Eltom, S. Kandeepan, R. J. Evans, Y. C. Liang, and B. Ristic, "Statistical spectrum occupancy prediction for dynamic spectrum access: A classification," *EURASIP J. Wireless Commun. Netw.*, vol. 2018, no. 1, p. 29, Feb. 2018.

[7] A. M. Kuzminskiy, P. Xiao, and R. Tafazolli, "Spectrum sharing with decentralized occupation control in rule regulated networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 5, no. 2, pp. 281–294, Jun. 2019.

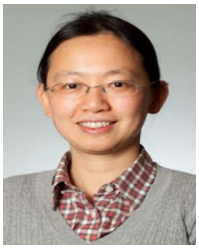
[8] H. Mosavat-Jahromi, Y. Li, Y. Ni, and L. Cai, "Distributed and adaptive reservation MAC protocol for beaconing in vehicular networks," *IEEE Trans. Mobile Comput.*, early access, May 4, 2020, doi: 10.1109/TMC.2020.2992045.
[9] M. Song, C. Xin, Y. Zhao, and X. Cheng, "Dynamic spectrum access: From cognitive radio to network radio," *IEEE Wireless Commun.*, vol. 19, no. 1, pp. 23–29, Feb. 2012.
[10] Q. Zhao and B. M. Sadler, "A survey of dynamic spectrum access," *IEEE Signal Process. Mag.*, vol. 24, no. 3, pp. 79–89, May 2007.
[11] X. Xing, T. Jing, W. Cheng, Y. Huo, and X. Cheng, "Spectrum prediction in cognitive radio networks," *IEEE Wireless Commun.*, vol. 20, no. 2, pp. 90–96, Apr. 2013.
[12] H. Eltom, S. Kandeepan, Y.-C. Liang, and R. J. Evans, "Cooperative soft fusion for HMM-based spectrum occupancy prediction," *IEEE Commun. Lett.*, vol. 22, no. 10, pp. 2144–2147, Oct. 2018.
[13] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997.
[14] K. Deng, A. W. Moore, and M. C. Nechyba, "Learning to recognize time series: Combining ARMA models with memory-based learning," in *Proc. IEEE Int. Symp. Comput. Intell. Robot. Autom. Towards New Comput. Principles Robot. Autom.*, Monterey, CA, USA, Jul. 1997, pp. 246–251.
[15] J. A. Stine and C. E. Caicedo Bastidas, "Enabling spectrum sharing via spectrum consumption models," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 4, pp. 725–735, Apr. 2015.
[16] T. Fujii, "Smart spectrum management for V2X," in *Proc. IEEE Int. Symp. Dyn. Spectr. Access Netw. (DySPAN)*, Seoul, South Korea, 2018, pp. 1–8.
[17] *IEEE Standard for Definitions and Concepts for Dynamic Spectrum Access: Terminology Relating to Emerging Wireless Networks, System Functionality, and Spectrum Management*, IEEE Standard 1900.1-2019, pp. 1–78, 2019.
[18] M. Wellens and P. Mahonen, "Lessons learned from an extensive spectrum occupancy measurement campaign and a stochastic duty cycle model," in *Proc. 5th Int. Conf. Testbeds Res. Infrastruct. Develop. Netw. Commun. Workshops*, Washington, DC, USA, Apr. 2009, pp. 1–9.
[19] S. Iliya, E. Goodyer, J. Gow, M. Gongora, and J. Shell, "Spectrum occupancy survey in Leicester, U.K., for cognitive radio application," *Int. J. Sci. Eng. Res.*, vol. 6, no. 8, pp. 1–7, Aug. 2015.
[20] L. Li *et al.*, "A cloud-based spectrum environment awareness system," in *Proc. IEEE 28th Annu. Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Montreal, QC, Canada, Oct. 2017, pp. 1–6.
[21] M. Lopez-Benitez, A. Umbert, and F. Casadevall, "Evaluation of spectrum occupancy in Spain for cognitive radio applications," in *Proc. IEEE 69th Veh. Technol. Conf.*, Barcelona, Spain, Apr. 2009, pp. 1–5.
[22] J. Jacob, B. R. Jose, and J. Mathew, "Bayesian analysis of spectrum occupancy prediction in cognitive radio," *Smart Sci.*, vol. 4, no. 2, pp. 52–61, May 2016.
[23] A. Azzouni and G. Pujolle, "A long short-term memory recurrent neural network framework for network traffic matrix prediction," 2017. [Online]. Available: arXiv:1705.05690.
[24] H. Rutagemwa, A. Ghasemi, and S. Liu, "Dynamic spectrum assignment for land mobile radio with deep recurrent neural networks," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Kansas City, MO, USA, May 2018, pp. 1–6.
[25] L. Yu, J. Chen, G. Ding, Y. Tu, J. Yang, and J. Sun, "Spectrum prediction based on Taguchi method in deep learning with long short-term memory," *IEEE Access*, vol. 6, pp. 45923–45933, 2018.
[26] Z. Li, W. Yang, S. Peng, and F. Liu, "A survey of convolutional neural networks: Analysis, applications, and prospects," 2020. [Online]. Available: arXiv:2004.02806.
[27] B. Lim and S. Zohren, "Time series forecasting with deep learning: A survey," 2020. [Online]. Available: arXiv:2004.13408.
[28] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep Learning*, vol. 1. Cambridge, MA, USA: MIT Press, 2016.
[29] A. He *et al.*, "A survey of artificial intelligence for cognitive radios," *IEEE Trans. Veh. Technol.*, vol. 59, no. 4, pp. 1578–1592, May 2010.
[30] H. Eltom, S. Kandeepan, B. Moran, and R. J. Evans, "Spectrum occupancy prediction using a hidden Markov model," in *Proc. 9th Int. Conf. Signal Process. Commun. Syst. (ICSPCS)*, Cairns, QLD, Australia, Dec. 2015, pp. 1–8.
[31] Y. Zhang, J. Hou, V. Towhidlou, and M. R. Shikh-Bahaei, "A neural network prediction-based adaptive mode selection scheme in full-duplex cognitive networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 5, no. 3, pp. 540–553, Sep. 2019.
[32] S. Wang, H. Liu, P. H. Gomes, and B. Krishnamachari, "Deep reinforcement learning for dynamic multichannel access in wireless networks," *IEEE Trans. Cogn. Commun. Netw.*, vol. 4, no. 2, pp. 257–265, Jun. 2018.
[33] R. Mennes, F. A. P. De Figueiredo, and S. Latré, "Multi-agent deep learning for multi-channel access in slotted wireless networks," *IEEE Access*, vol. 8, pp. 95032–95045, 2020.
[34] S. Pagadarai and A. Wyglinski, "A linear mixed-effects model of wireless spectrum occupancy," *EURASIP J. Wireless Commun. Netw.*, vol. 2010, no. 1, pp. 1–7, Aug. 2010.
[35] A. Gorcin, H. Celebi, K. A. Qaraqe, and H. Arslan, "An autoregressive approach for spectrum occupancy modeling and prediction based on synchronous measurements," in *Proc. IEEE 22nd Int. Symp. Pers. Indoor Mobile Radio Commun. (PIMRC)*, Toronto, ON, Canada, Sep. 2011, pp. 705–709.
[36] J. Yin, L. Li, H. Zhang, X. Li, A. Gao, and Z. Han, "A prediction-based coordination caching scheme for content centric networking," in *Proc. 27th Wireless Opt. Commun. Conf. (WOCC)*, Hualien, Taiwan, Apr. 2018, pp. 1–5.
[37] Z. Wang and S. Salous, "Spectrum occupancy statistics and time series models for cognitive radio," *J. Signal Process. Syst.*, vol. 62, no. 2, pp. 145–155, Feb. 2011.
[38] C. E. C. Bastidas and A. Mohan, "Spectrum consumption model builder and analysis tool: A software tool to enhance spectrum sharing," *Analog Integr. Circuits Signal Process.*, vol. 91, no. 2, pp. 217–226, 2017.
[39] R. Adhikari and R. K. Agrawal, "An introductory study on time series modeling and forecasting," 2013. [Online]. Available: arXiv:1302.6613.
[40] X. Cao, R. Shan, J. Fan, and P. Li, "One new method on ARMA model parameters estimation," *Mod. Appl. Sci.*, vol. 3, no. 5, p. 204, 2009.
[41] R. A. Davis, "Gauss-Newton and M-estimation for ARMA processes with infinite variance," *Stochastic Process. Appl.*, vol. 63, no. 1, pp. 75–95, 1996.
[42] F. Chollet *et al.*. (2018). *Keras*. [Online]. Available: https://keras.io
[43] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014. [Online]. Available: arXiv:1412.6980.
[44] H. Sak, A. W. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proc. 15th Annu. Conf. Int. Speech Commun. Assoc. (INTERSPEECH)*, Sep. 2014, pp. 338–342.
[45] Y. Hua, Z. Zhao, R. Li, X. Chen, Z. Liu, and H. Zhang, "Deep learning with long short-term memory for time series prediction," *IEEE Commun. Mag.*, vol. 57, no. 6, pp. 114–119, Jun. 2019.
[46] J. J. Dziak, D. L. Coffman, S. T. Lanza, R. Li, and L. S. Jermiin, "Sensitivity and specificity of information criteria," Dept. Health Hum. Develop., Pennsylvania State Univ., State College, PA, USA, Rep. 12-119, Jun. 2012.

Hamed Mosavat-Jahromi (Graduate Student Member, IEEE) received the B.Sc. degree in electrical engineering from the Iran University of Science and Technology in 2012, the M.Sc. degree in electrical engineering from the University of Tehran in 2015, and the Ph.D. degree in electrical engineering from the Department of Electrical and Computer Engineering, University of Victoria, Victoria, BC, Canada.

His research interests include vehicular networks, Internet of Things, machine learning, and optimization with applications in networking. He was recipient of the Transportation Electronics Fellowship Award from the IEEE Vehicular Technology Society in 2020.

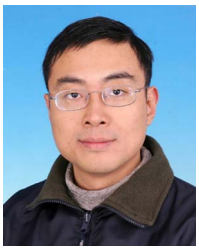
Yue Li received the Ph.D. degree in electrical and computer engineering from the University of Victoria, Victoria, BC, Canada, in 2018.

From 2008 to 2013, he worked as a Standard Preresearch Engineer with the Wireless Research Department, Huawei, Shenzhen, China. He has been closely involved in 3GPP standards evolvement and has held numerous patents in WCMDA, LTE-A, and 5G systems. He is currently a Postdoctoral Research Fellow with the Department of Electrical and Computer Engineering, University of Victoria. His research interests include next-generation cellular systems, wireless network design and optimization, wireless system modeling, and performance analysis.



Lin Cai (Fellow, IEEE) received the M.A.Sc. and Ph.D. degrees (awarded Outstanding Achievement in Graduate Studies) in electrical and computer engineering from the University of Waterloo, Waterloo, Canada, in 2002 and 2005, respectively.

Since 2005, she has been with the Department of Electrical and Computer Engineering, University of Victoria, where she is currently a Professor. Her research interests span several areas in communications and networking, with a focus on network protocol and architecture design supporting emerging multimedia traffic and the Internet of Things. She was a recipient of the NSERC Discovery Accelerator Supplement Grants in 2010 and 2015, and the Best Paper Awards of IEEE ICC 2008 and IEEE WCNC 2011. She has co-founded and chaired the IEEE Victoria Section Vehicular Technology and Communications Joint Societies Chapter. She has served as a TPC Co-Chair for IEEE VTC2020-Fall, and a TPC Symposium Co-Chair for IEEE Globecom'10 and Globecom'13. She is a registered Professional Engineer in British Columbia, Canada. She has been elected to serve the IEEE Vehicular Technology Society Board of Governors from 2019 to 2021. She has served as an Area Editor for the IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, a Member of the Steering Committee of the IEEE TRANSACTIONS ON BIG DATA and IEEE TRANSACTIONS ON CLOUD COMPUTING, an Associate Editor of the IEEE INTERNET OF THINGS JOURNAL, IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON COMMUNICATIONS, *EURASIP Journal on Wireless Communications and Networking*, *International Journal of Sensor Networks*, and *Journal of Communications and Networks*, and as the Distinguished Lecturer of the IEEE VTS Society. In 2020, she was elected as a Member of the Royal Society of Canada's College of New Scholars, Artists and Scientists. She was also elected as a 2020 "Star in Computer Networking and Communications" by N2Women. She is an NSERC E. W. R. Steacie Memorial Fellow.



Jianping Pan (Senior Member, IEEE) received the bachelor's and Ph.D. degrees in computer science from Southeast University, Nanjing, China.

He is currently a Professor of Computer Science with the University of Victoria, Victoria, BC, Canada. He did his postdoctoral research with the University of Waterloo, Waterloo, ON, Canada. He also worked with Fujitsu Labs and NTT Labs. His area of specialization is computer networks and distributed systems, and his current research interests include protocols for advanced networking, performance analysis of networked systems, and applied network security. He received the IEICE Best Paper Award in 2009, the Telecommunications Advancement Foundation's Telesys Award in 2010, the WCSP 2011 Best Paper Award, the IEEE Globecom 2011 Best Paper Award, the JSPS Invitation Fellowship in 2012, the IEEE ICC 2013 Best Paper Award, and the NSERC DAS Award in 2016, and has been serving on the technical program committees of major computer communications and networking conferences, including IEEE INFOCOM, ICC, Globecom, WCNC, and CCNC. He was the Ad Hoc and Sensor Networking Symposium Co-Chair of IEEE Globecom 2012 and an Associate Editor of IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY. He is a Senior Member of ACM.