A Neural Network Prediction Based Adaptive Mode Selection Scheme in Full-Duplex Cognitive Networks

Yirun Zhang, Jiancao Hou, Vahid Towhidlou, and Mohammad Shikh-Bahaei

Abstract—In this paper, we propose a neural network (NN) predictor for multi-slot prediction and an adaptive mode selection scheme, with the goal of improving secondary users (SUs) throughput while alleviating collision to primary user (PU) in full-duplex (FD) cognitive networks. Conventionally, FD SU can either operate in a transmission-and-reception (TR) mode to improve its throughput, or a transmission-and-sensing (TS) mode to avoid collision to PU. The difference between TR and TS modes in goal gives rise to a trade-off between higher SUs throughput and lower collision probability, which can be optimised by allowing SU to switch between these two modes. Accordingly, we design an NN predictor to predict PUs future activity which is considered as the basis of switching. In such a context, we analyse the prediction performance in terms of prediction error probability. We also compare the performance of our proposed scheme with conventional TR and TS modes in terms of SUs average throughput and collision probability, respectively. Simulation results show that our proposed scheme achieves almost the same SUs average throughput as TR mode when PU has low tolerance for collision. Meanwhile, the collision probability can be reduced by up to 92% close to that of TS mode.

Keywords—Full-duplex, neural network, cognitive radio, spectrum occupancy prediction, adaptive mode selection.

I. INTRODUCTION

THE severe scarcity of spectrum resource inspires the development of new spectrum management techniques [1]–[5]. Cognitive radio networks (CRNs), in which secondary users (SUs) are able to sense and access the spectrum white spaces by using dynamic spectrum access (DSA) technology, has become a promising approach to improve SU’s spectrum efficiency [4], [6]. For example, seven dedicated channels used in vehicular networks based on IEEE 802.11p standard could easily get congested if the vehicle density is high [7]–[10]. In this case, DSA technology can help avoid congestion by allowing vehicles to sense and access underutilised licensed frequency bands [8]–[10]. On the other hand, full-duplex (FD) operation, which allows devices to transmit and receive over the same frequency band at the same time, has also emerged as a solution to alleviate the spectrum scarcity problem [2], [11], [12]. FD operation brings new transmission mechanisms to CRNs. SUs are allowed to sense the channel while transmitting data. They can also sense the channel and then transmit-and-receive data at the same time. These mechanisms are called transmission-and-sensing (TS), and transmission-and-reception (TR), respectively [11], [13]–[16]. In TS mode, the collision probability is low because of the continuous sensing operation. SU’s throughput is also lower than that of TR mode since it cannot transmit and receive at the same time. In TR mode, SU can gain higher throughput due to the simultaneous transmission-and-reception. However, the SU cannot be aware of primary user’s (PU’s) return during its transmission-and-reception, which causes higher collision probability than that of TS mode. These two modes give rise to a trade-off between higher SU’s throughput and lower collision probability. Therefore, it is necessary for SU to adaptively switch between these two transmission modes based on different PU’s activity in order to optimise the trade-off.

Many existing works focus on only one transmission mode. The authors in [17] analyse the performance of TS mode (i.e., “listen-and-talk” (LAT) protocol) and demonstrate that it outperforms traditional half-duplex “listen-before-talk” (HD-LBT) protocol in terms of SU’s spectrum efficiency. The sensing threshold adaptation and transmit power optimisation problems of TS mode are then solved in [18]. The advantage of TR mode in cellular networks is analysed in [19]. Some recent works discuss the trade-off between TR and TS mode based on different benchmarks. The authors in [13] derive the collision probability and throughput for both transmission modes and made a trade-off between them based on the heaviness of traffic load. The authors in [15] analyse both PU’s and SU’s performance in terms of average throughput and then proposes an optimal scheme based on the number of cooperative SUs. These works assume that the SU knows PU’s activity pattern (e.g., the distribution of PU’s activity and the corresponding parameters) in advance. However, the PU’s statistics are usually unknown to SU in practice and need to be estimated [20]. A maximum likelihood estimator was proposed in [20] to estimate the average length of channel’s idle and busy time. Although the estimator gives some information on PU’s statistics (e.g., the average duration that PU occupies the channel), it cannot provide PU’s dynamics in real time in order to help SU sense and access the channel better.

In order to learn and predict PU’s dynamics, spectrum prediction for cognitive radio has been studied using different machine learning techniques, such as hidden Markov models (HMMs) [21], [22] and neural network (NN) [23], [24]. HMM-based prediction methods in [21], [22] have several...
drawbacks such as, the computationally expensive online-training process, and determining an optimal number of states in the HMM is difficult. In addition, simulation results in a recent paper [25] indicates that HMM-based prediction methods show poor performance on non-Markovian real-world spectrum data, which also motivates us to apply the so-called “universal approximator” NN prediction method to the spectrum prediction. The authors in [23] propose an NN based spectrum occupancy prediction method using off-line training, which requires less memory space and lower computational complexity. All of these existing spectrum prediction works only aim at deciding whether the very next time slot is occupied by PU or not. This one time slot, however, contains limited information on PU’s future activity, which may be outdated due to the internal hardware delays [26]. Due to the drawbacks of HMM-based prediction methods, in this paper we design an NN predictor to predict multiple time slots for transmission mode selection. Based on PU’s future activity of multiple slots, SU can adaptively switch between different transmission modes to improve its throughput and/or avoid the collision.

In recent years, researchers have paid attention to the problem of mode selection between FD and HD. For example, the authors in [27] propose a protocol for FD Wi-Fi network in which the access point (AP) can flexibly switch between FD and HD communications based on the probability of constructing FD transmission. Optimisation problems are solved to maximise the spectrum efficiency under different transmission demands. Different from [27] in which users are not allowed to communicate with each other directly, we study the transmission mode selection problem in M2M CRNs. The authors in [28] propose an adaptive mode selection scheme based on SU’s belief regarding the idleness of the PU. The belief update policy is partially based on calculating the probability of PU’s return. Such method requires a priori knowledge of PU’s activity and signal pattern, which is not practical in the case that PU is not willing to share these information.

The contributions of this paper are as follows. First, we design a multi-layer NN predictor to solve the channel status prediction problem. The NN predictor we designed can effectively classify the future channel occupancy status into two classes, one refers to the totally unoccupied status and the other represents that some slots will be occupied by PU. Second, we propose a neural network based adaptive mode selection (NN-AMS) scheme by which SU can utilise the spectrum white spaces efficiently while avoiding the collision to PU. Our proposed NN-AMS scheme is applicable to any random distribution of PU’s activity pattern. Third, we define the prediction error probabilities and analyse the robustness of prediction. Based on these metrics, we test the prediction performance and analyse the influence of sensing error on it. Fourth, we derive SU’s average throughputs and collision probabilities of our proposed NN-AMS scheme, conventional FD-TR and FD-TS modes. We find the optimal length of transmission duration for both NN-AMS scheme and TR mode via simulation of the NN and numerical analysis. Under the optimal length of transmission duration, we address the conclusion that in low sensing error probability scenario, our proposed NN-AMS scheme shows better performance than the conventional TR and TS modes in terms of collision probability and SU’s average throughput, respectively.

The rest of this paper is organised as follows. In Section II, we explain our proposed NN-AMS scheme in detail. The performance of the NN-AMS scheme in terms of prediction error probability, the SU’s average throughput and collision probability are analysed in Section III. Simulation results are then presented in Section IV. In the end, we draw the conclusion in Section V.

II. The Proposed NN-AMS Scheme

We consider a CRN consisting of a PU base station with multiple licensed PUs and a pair of unlicensed SUs which opportunistically access the PU-licensed channel, as shown in Fig. 1. SUs are equipped with FD radios with partial SIS capability. For fairness and generality, we assume that each SU is equipped with two antennas Ant₁ and Ant₂, where Ant₁ is used for data transmission and Ant₂ is used for reception. It is worth noting that when operating in TS mode, the transmitter SU (e.g., SU₁) uses both Ant₁ and Ant₂ for simultaneous data transmission and sensing, whereas the receiver SU (e.g., SU₂) uses only Ant₂ for receiving data from the transmitter SU similar to the system setup in [18]. There is a control channel in the secondary network, which is used for exchanging sensing results and mode selection decisions.

PU’s traffic is modelled as a simple ON and OFF process with average lengths of $T_0$ and $T_1$ respectively, based on whether the channel is occupied by PU or not [29]. The activity pattern of PU’s traffic follows arbitrary random distribution whose parameters are unknown to the SUs.

![Fig. 1. Structure of the system model.](image1)

![Fig. 2. The processing structure.](image2)
The proposed processing structure is illustrated in Fig. 2. The off-line training is done by a centralised power unit (e.g., cloud or network edge). After training, both SUs can use the NN predictor. The sensing function is implemented by the major SU (e.g., SU₁) who is responsible for making mode selection decision while the minor SU (e.g., SU₂) listens to its decision. Specifically, the major SU is the transmitting SU while the minor SU is the receiving SU when they are operating in TS mode. At the beginning of each transmission duration, the major SU predicts and senses the channel. Based on the prediction and sensing results, it selects a transmission mode and then sends the decision to the minor SU by using a short feedback signal through the control channel. At the end of each transmission duration, two SUs swap their roles as the major SU and minor SU. The input of the NN predictor is updated by adding new sensing results and deleting old ones.

A. Energy-Detection Based Sensing
The energy-detection based spectrum sensing technique is applied to the NN predictor due to its simplicity and ability to identify spectrum white spaces without requiring any a priori knowledge of PU’s signal pattern. The sensing result contains two states: “1” is for busy; “−1” is for idle. The probability of spectrum sensing is judged by two fundamental measures, the detection probability \( P_d \) and the false-alarm probability \( P_f \). The former refers to the probability if PU is occupying the channel, SU can successfully detect it. The latter is the probability if the channel is idle, SU falsely decides that it is occupied by PU.

The impact of SI on the sensing performance should be taken into account due to the imperfect suppression of SI in practice. Therefore, we have to consider the residual SI when deriving the false-alarm and detection probabilities. Here we do not elaborate on detail of the applied SIS methods and just quantify this capability by \( \chi_i \) := Power of residual SI / Transmit power \((0 \leq \chi_i \leq 1)\) for the \( i \)th SU [13], where \( \chi_i = 0 \) corresponds to perfect SIS (i.e., no residual SI) and \( \chi_i = 1 \) corresponds to the case where the SI is not suppressed. In this paper, we assume both SUs have the same SIS coefficient \( \chi \).

From [13], the two probabilities considering SI (e.g., the continuous sensing operations in TS mode) are given by:

\[
P_{d,s} = Q\left(\frac{\varepsilon_{th}}{\sigma^2} - \chi^2 SNR_{s,s} - SNR_{s,p} - 1\right) \times \frac{\omega_s T_s}{2\chi^2 SNR_{s,s} + 2\chi^2 SNR_{s,s} SNR_{s,p} + 2SNR_{s,p} + 1} \tag{1}
\]

\[
P_{f,s} = Q\left(\frac{\varepsilon_{th}}{\sigma^2} - \chi^2 SNR_{s,s} - 1\right) \times \frac{\omega_s T_s}{2\chi^2 SNR_{s,s} + 1} \tag{2}
\]

where \( Q(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{t^2}{2}} dt \) is the Q-function; \( \varepsilon_{th} \) is the threshold of energy detection; \( \sigma^2 \) is the variance of the circular symmetric Gaussian background noise; \( \omega_s \) is the sampling frequency. Moreover, \( SNR_{i,j} \) is the signal-to-noise ratio (SNR) transmitted by transmitter \( j \) and received at receiver \( i \). \( i \) and \( j \) can be either \( p \) or \( s \), where \( p \) and \( s \) refer to PU and SU respectively. A sensing threshold adaptation method, which is similar to the method proposed in [18], is applied in order to maintain the sensing performance under all values of \( \chi \). If \( \chi \) equals to zero, the false-alarm probability and detection probability converge to those without SI (e.g., the initial sensing operation of TR mode) which are defined as \( P_f \) and \( P_d \), respectively.

B. Neural Network Based Prediction

1) Structure Design: The main idea of our proposed NN predictor is to insert \( N \) number of past sensing results into the NN predictor and get the estimated PU’s future activity of \( M \) time slots. Generally, a small value of \( N \) cannot provide sufficient information on past channel occupancy status while a large value of \( N \) is burdensome for training [34]. A proper value of \( N \) can be set slightly higher than \( M \) so that it can contain sufficient information on PU’s past dynamics and does not burden the training. Through simulation, we found that \( N \) equal to 75 is sufficient for \( M \) smaller than 50. We introduce two parameters \( m \) and \( m_{th} \) to classify PU’s future activity of \( M \) time slots. \( m \) refers to the number of idle time slots in the future \( M \) time slots, and \( m_{th} \) is the corresponding threshold for \( m \). If \( m \geq m_{th} \), the output of the NN predictor is “−1”; Otherwise, its output is “1”.

The NN predictor contains three kinds of layers, including the input layer, the hidden layer, and the output layer, which is illustrated in Fig. 3. The output layer has one neuron. Each layer is fully-connected with adaptive weights, biases and an activation function. We use the Tan-Sigmoid function as the activation function for each layer, which is given by:

\[
\text{Tan-Sigmoid}: \ tansig(x) = \frac{2}{1+e^{-2x}} - 1. \tag{3}
\]

We define the input vector as \( x = [x_1, x_2, ..., x_N]^T \) whose elements are either “−1” (idle) or “1” (busy). The hidden layers and output layer contain multiple neurons. Each neuron in the network has a weight vector \( w \) and a bias \( b \), whose elements are all real numbers. Then we define the weight vector for neuron \( i \) in hidden layer one as \( w_{1,i} = [w_{1,1,i}, w_{1,2,i}, ..., w_{1,N,i}]^T \) which contains \( N \) elements, for neuron \( j \) in hidden layer two as \( w_{2,j} = [w_{2,1,j}, w_{2,2,j}, ..., w_{2,N,j}]^T \) which contains 15 elements, and for the neuron in output layer as \( w_3 = [w_3, w_{3,1}, ..., w_{3,20}]^T \) which contains 20 elements. The biases for neurons in each layer are defined as \( b_1 = [b_1, b_{1,1}, ..., b_{1,15}]^T \), \( b_2 = [b_2, b_{2,2}, ..., b_{2,20}]^T \) and \( b_3 = [b_3] \), receptively. The output vector for hidden layers are \( y_1 = [y_1, y_{1,1}, ..., y_{1,15}]^T \) and \( y_2 = [y_2, y_{2,2}, ..., y_{2,20}]^T \), respectively. The output of the output layer is \( y_3 \). The three outputs can be expressed by:

\[
y_1 = \text{tansig}(\sum_{k=1}^{15} w^k_{1,i}x_k + b_{1,i}) = \text{tansig}(w_{1,i}x + b_{1,i}) \tag{4a}
\]

\[
y_2 = \text{tansig}(\sum_{k=1}^{20} w^k_{2,i}y_k + b_{2,i}) = \text{tansig}(w_{2,i}y_1 + b_{2,i}) \tag{4b}
\]

\[
y_3 = \text{tansig}(\sum_{k=1}^{20} w^k_{3,i}y_k + b_3) = \text{tansig}(w_3y_2 + b_3). \tag{4c}
\]

where \( i \in [1,15] \) and \( j \in [1,20] \). By using a decision threshold at the output, the predicted value \( y_3 \) can be expressed as a binary symbol:
Determining the number of hidden layers and neurons requires numerous trials. According to the guidance provided in [35], an NN with two hidden layers is sufficient for our model as it “can represent an arbitrary decision boundary to arbitrary accuracy with rational activation functions and can approximate any smooth mapping to any accuracy.” It is worth noting that NNs with more than two hidden layers are more practical in the case that training data is sufficient (e.g., in computer vision and natural language processing). However, it is not possible for SUs to obtain numerous sensing data for training in practice. Figure. (7) in [24] also proves that NNs with more than two hidden layers suffer from poor performance for spectrum prediction due to slow convergence and lack of training data in practice. Therefore, we only compare the performance of NNs with less than three hidden layers for our model. It can be seen from Fig. 4 that the NN with two hidden layers outperforms the one with single hidden layer in terms of prediction error.

To decide the number of neurons, we follow one of the rules of thumb (see, e.g., [35]) which indicates “the number of hidden neurons should be between the size of the input layer and the size of the output layer.” Figure. 4 illustrates the prediction error probability versus number of total hidden neurons $N_{HL}$. It can be seen that the best prediction performance is achieved when $N_{HL} \geq 35$. However, a too large $N_{HL}$ also results in slow convergence and overfitting. Through simulation, we found that two hidden layers with 15 and 20 neurons respectively can provide good prediction performance with small training time.

For more complex scenarios such as multi-PU multi-SU scenario where each SU pair is allocated with a PU channel, the NN predictor needs to make mode selection decision for each SU pair. Thus, the number of neurons in both input layer and output layer should be increased (i.e., add $N$ neurons to the input layer and one neuron to the output layer for each extra PU channel). Then the number of neurons in hidden layers could be re-determined based on the following rule:

$$n < N_{HL} < nN,$$  \hspace{1cm} (6)

where $N_{HL}$ is the total number of neurons in hidden layers and $n$ is the number of PU channels. It is worth noting that the exact number of hidden neurons in each hidden layer for each scenario should be identified through a number of trials.

2) Preprocessing and Training: The general steps of preprocessing and training are described by the flow chart in Fig. 5. The NN predictor is trained by using $N_{tr}$ independent training sets. Each training set contains a pair of input vector and output value. First, SU senses the channel for a considerably long period of time and generates a sensing sequence. Then it randomly picks $N_{tr}$ sets each with length of $N + M$ from the sensing sequence. Subsequently, SU puts the first $N$ sensing results into the input vector and counts the number of idle slots in the last $M$ slots. If $m \geq m_{th}$, the SU sets the target output value to “-1”; Otherwise, it sets the target output value to “1”.

After all training sets are preprocessed, SU inputs them into the NN predictor for off-line training. The main idea of NN training is to update the weight vectors $w$ and biases $b$ iteratively in order to minimise the mean squared error between the predicted output and the target output. We use the Levenberg-Marquardt Backpropagation (LMBP) algorithm in [36] for off-line training due to its quick convergence. The detailed description of the LMBP algorithm is omitted and can be found in [36].

C. Adaptive Mode Selection Scheme

SU adaptively switches between FD-TR, FD-TS and HD modes, based on the output of NN predictor and SIS coefficient.
In this section, we first introduce these three transmission modes and then describe the adaptive mode selection algorithm.

As shown in Fig. 6, in TR mode, the transmission frame is defined as $T_p = M T_s$, where $T_s$ refers to the length of one time slot. SU first senses the primary channel for one time slot with time $T_s$ and then transmits and receives data in FD manner for $M - 1$ time slots with time $T_t = T_p - T_s$ which is the data transmission duration. The reason that we add an initial sensing slot before data transmission is to stabilise the prediction performance, which will be detailed in Section III-B. If the initial sensing gives a busy result, then the SU switches to the TS mode (i.e., keep sensing and opportunistically transmit in order to avoid collision with PU) instead of transmitting and receiving. SU can improve its throughput in TR mode because of its concurrent transmission-and-reception over the same frequency band. However, it causes a collision to PU since SU cannot be aware of the return of PU during its transmission-and-reception.

1) TS mode: In TS mode, after the initial sensing period $T_s$, SU senses the channel continuously during data transmission, as shown in Fig. 6. The duration of data transmission is divided into multiple time slots with the same length of time as the initial sensing period. The SU transmits data whenever it senses the channel to be idle. If at the end of any time slot the channel status is detected to be busy, SU immediately stops its transmission. It is worth noting that we consider each time slot of the total transmission duration as a packet duration. Thus, a total transmission duration consists of $M - 1$ packets. The receiver SU decodes whenever it receives a packet. The interruption during the transmission of a packet results in the re-transmission of it in the next transmission duration whilst other packets which have already been transmitted can still be successfully decoded. Then SU can alleviate collision to PU in TS mode at the expense of lower throughput than in TR mode because it cannot simultaneously transmit and receive data while sensing.

2) TR mode: For adaptive switching between TR and TS modes, the determination of a proper switching threshold is critical. We use $m_{th} = M$ as the switching threshold in this paper because it gives the lowest collision probability than other choices of thresholds. It is clear that collision occurs even if the prediction is perfect when $M > m_{th}$. For example, if $m_{th} = M - 1$ and the PU comes back to the channel during the last time slot, the SU will decide to transmit in TR mode in this case and then collide with PU during the last time slot. Therefore, in order to avoid collision with PU while improving the SU’s throughput, we use $m_{th} = M$ as the switching threshold. Based on this principle, SU chooses TR mode in order to improve the spectrum efficiency if the output is “-1”; Otherwise, it chooses to transmit in TS mode in order to avoid collision with PU.

The input vector is updated at the end of one transmission frame $T_p$. Depending on which transmission mode SU has chosen, different update policy is applied. If SU has chosen TR mode and the initial sensing result is “-1” (i.e., SU starts to transmit data in TR mode), SU adds the prediction results (i.e., $M$ zeros) to the end of the input vector and deletes the oldest $M$ elements. If SU has chosen TR mode and the initial sensing result is “1”, or it has chosen TS mode, it simply updates the input vector using $M$ sensing results. The NN-AMS scheme follows the general steps described in Algorithm 1.

III. PERFORMANCE ANALYSIS

In this section, we first define the prediction error probabilities and discuss the robustness in prediction. Then we derive SU’s average throughputs of NN-AMS scheme, TR and TS mode. In order to compare the spectrum efficiency of these three modes when PU has zero tolerance for collision, we
Algorithm 1 Proposed Adaptive Mode Selection Scheme

Begin  While (Next M Time Slots Prediction is NOT NULL)
Step-1: Input Vector Initialisation: Initialise the input vector \( x \) by sensing \( N \) time slots.
Step-2: Spectrum Occupancy Prediction: Input \( x \) into the NN predictor and get the output \( y_s \).
Step-3: Mode Selection Decision:
   If \( y_s = -1 \) And initial sensing = -1 Then
      SU transmits in TR mode in \( M \) slots.
   Else
      SU transmits in TS mode in \( M \) slots.
   End IF
Step-4: Input Vector Update:
   If \( y_s = -1 \) And initial sensing = -1 Then
      Use the prediction result as new sensing results (i.e.,
      All \( M \) time slots are considered to be idle).
   Else
      Use new sensing results.
   End IF
   Delete the first \( M \) elements from the input vector \( x \)
   and add the new sensing results to the end, and go to Step-2.
End While

also derive SU’s non-collision average throughputs. Finally, the collision probabilities of three modes are introduced.

A. Prediction Error Probabilities

The prediction is not perfect due to the uncertainty nature of the random distribution of PU’s activity. We can judge the accuracy of prediction by testing with \( N_{tt} \) independent testing sets and computing the prediction output. Each testing set is generated by randomly observing \( N \) time slot and predicted the spectrum occupancy status of future \( M \) time slots. We compare the prediction results of testing with true spectrum occupancy status and calculate the prediction false-alarm probability \( P_{pf} \), the prediction detection probability \( P_{pd} \), and the average prediction error probability \( P_e \). If the prediction result is different from the true spectrum occupancy status, we call it an incorrect prediction. Note that we do not need to know the true spectrum occupancy status in practice. The only purpose of doing so is to test the performance of our predictor.

The prediction false-alarm probability \( P_{pf} \) refers to the probability that the prediction result shows “-1” while the true result is “1”, which is defined as:

\[
P_{pf} = \Pr[y_s = 1 | m = M] = \frac{N_{01}}{N_0},
\]

where \( N_{01} \) refers to the number of sets that the SU falsely detects the channel will be occupied by the PU when there are no PU activities in the next \( M \) slots; \( N_0 \) is the number of sets that there are no PU activities in the next \( M \) slots among \( N_{tt} \) testing sets.

The prediction detection probability \( P_{pd} \) refers to the probability that the prediction result shows “-1” while the true result is also “-1”, which is given by:

\[
P_{pd} = \Pr[y_s = 1 | m < M] = 1 - \frac{N_e - N_{01}}{N_{tt} - N_0},
\]

where \( N_e \) refers to the total number of incorrect prediction sets.

The average prediction error probability \( P_e \), which refers to the probability that the prediction result is different from the true result, is defined as:

\[
P_e = \frac{N_e}{N_{tt}}.
\]

Although theoretically analysing \( P_e \) of NN prediction is almost impossible due to the “black box” nature of NN, we can still use conventional mathematical model based prediction method such as HMM to provide some theoretical performance guarantee. Assuming that PU’s activity pattern is a Markov model with unknown parameters, we use the derivations and analytical results of single-slot prediction error based on HMM in [37] as the prediction error lower bound. The first point of the lower bound where sensing error probabilities are zero can be calculated using the entropy rate of the Markov chain as in Eq. (16) and Fig. (1) in [37]:

\[
\mathcal{H}(P_e) = \mathbf{H}(\text{Ber}(\theta_0))\mu_0 + \mathbf{H}(\text{Ber}(\theta_1))\mu_1,
\]

where \( \mathbf{H}(\text{Ber}(x)) = -P(x)\log_2 P(x) - P(1-x)\log_2 P(1-x) \) is the entropy rate of a Bernoulli-distributed random variable; \( \theta_0 \) and \( \theta_1 \) are probabilities that the channel state changes from OFF to ON and from ON to OFF, respectively; \( \mu_0 \) and \( \mu_1 \) are the occurrence probabilities of OFF and ON state, respectively. Obtaining the relationship between sensing error and prediction error requires the calculation of probability distribution on a Borel set of measures expressed using an integral, which is hard to evaluate as mentioned in [37]. Instead, the analytical result using recursive approximation and sampling is given in Fig. (4) in [37]. Here we use the same method to obtain the lower bound of prediction error probability as illustrated in Fig. 7.

Figure 7 shows average prediction error probability \( P_e \) versus sensing error probability, assuming \( P_f = P_d \). It can be seen that \( P_e \) increases slightly when the sensing error probabilities are small. When the sensing error probabilities become larger, \( P_e \) increases dramatically. Therefore, in order to have a high-quality prediction performance, the sensing
error should be controlled to remain as low as possible. In addition, a smaller value of $M$ results in $P_e$ values closer to the lower bound. This indicates that NN can obtain a close to optimal one-slot prediction accuracy on the Markov model.

B. Robustness in Prediction

Many existing spectrum occupancy prediction works considered only one-step prediction without looking into the influence of current prediction errors on subsequent prediction as well as the influence of sensing error on prediction [21]–[23], [26]. In this section, we analyse the robustness in prediction of our proposed NN-AMS scheme.

The robustness in prediction refers to the ability of tolerating disturbances that might affect the prediction performance. For example, errors in sensing reduce the prediction accuracy. The influence of sensing errors on prediction will be analysed in Section IV-A through simulation results. In this section, we focus on analysing the influence of prediction errors on the subsequent prediction performance. Note that when discussing the influence of prediction error on robustness, we assume that there is no sensing error since these two kinds of errors are independent of each other.

In general, there is no prediction error accumulation if SU transmits in TS mode. This is because the input vector $x$ of the NN predictor will be updated by adding new sensing results. However, if SU falsely transmits in TR mode due to the prediction error, the incorrect prediction results will be used to update the input vector, which will worsen the prediction error problem to the subsequent predictions. Three possible incorrect prediction scenarios for SU transmitting in TR mode are illustrated as cases $C_3$, $C_5$ and $C_7$ in Fig. 8. In $C_3$, SU gives a wrong prediction result but a correct sensing result; PU comes back to the channel during SU’s data transmission. Case $C_5$ and $C_7$ indicate that the prediction results and the sensing result are both wrong; The PU is actually occupying the channel.

Although SU updates the input vector with incorrect prediction results in $C_3$, this harmful prediction error will have little influence in subsequent transmissions with the help of the initial sensing operation in TR mode. The reason is that SU will have a high chance to detect the presence of PU since the detection probabilities $P_d$ and $P_{d,SI}$ are very high in practice [13], [38]. This indicates that SU will keep sensing the channel and updating the input vector with new sensing results instead of transmitting in TR mode. Since $N$ is smaller than either $T_0$ or $T_1$, the input vector will have already been completely updated before $C_3$ occurs again. This keeps the prediction performance stable since the incorrect input elements will be deleted after a few updates.

C. Secondary User’s Average Throughput

In this section, we derive the SU’s average throughput for NN-AMS scheme, TR and TS mode. The average throughput contains throughputs in different cases, which can be defined as:

$$R_{avg} = \sum_{i=1}^{n} \Pr[C_i] R_i,$$

where $n$ is the number of cases; $C_i$ refers to the $i$th case; $\Pr[C_i]$ is occurrence probability of the $i$th case; $R_i$ is SU’s throughput in the $i$th case.

Our proposed NN-AMS scheme is applicable under arbitrary PU’s activity distribution. However, in order to fairly compare with the existing works [13], [15], [17], [19], [38], we assume the length of the ON and OFF state follows the negative exponential distribution with parameter $\lambda_1$ and $\lambda_0$, respectively. Note that SU’s transmission duration $T_p$ is considered small relative to $T_1$ and $T_0$ ($T_p \ll T_0$ and $T_1$). Under this assumption, the scenario where one $T_p$ contains more than one ON and OFF transition rarely happens [13].

1) NN-AMS scheme: In our proposed NN-AMS scheme, there are eight different cases of SU’s transmission, $C_1$ to $C_8$, according to different beginnings and endings of SU’s one transmission duration and different prediction results, as shown in Fig. 8. We discuss each case in turn.

- $C_1$

In $C_1$, the transmission frame $T_p = MT_s$ of SU falls into the OFF state. A sensing slot is set before data transmission, which makes the data transmission duration to be $T_t = (M-1)T_s$. Since the initial sensing operation in TR mode is not influenced by SI, sensing error probabilities $P_f$ and $P_d$ are used instead of $P_{f,SI}$ and $P_{d,SI}$. SU decides to transmit in TR mode and senses the channel to be idle with probability $\bar{P}_{pf} \cdot \bar{P}_f$, where $\bar{P}_{pf} = 1 - P_{pf}$ is the complementary probability of prediction false-alarm probability $P_{pf}$; $\bar{P}_f = 1 - P_f$ is the complementary probability of false-alarm probability $P_f$. We define the forward recurrence time of the OFF state as $L_0$, which refers to the length from the start of SU’s transmission to the end of the OFF state. According to the memoryless property of exponential distribution, $L_0$ also follows exponential distribution with parameter $\lambda_0$. The occurrence probability of this case is given by:

$$\Pr[C_1] = \mu_0 \cdot \bar{P}_{pf} \cdot \bar{P}_f \cdot \Pr[L_0 \geq T_1]$$

$$= \mu_0 \cdot \bar{P}_{pf} \cdot \bar{P}_f \cdot e^{-\frac{T_1}{\lambda_0}},$$

(12)

where $\mu_0 = \frac{\lambda_0}{\lambda_0 + \lambda_1}$ refers to the occurrence probability of OFF state. Due to the simultaneous transmission-and-reception, the achievable throughput should be twice as high as that in half-

![Fig. 8. Eight different cases of SU’s transmission in the NN-AMS scheme.](image-url)
duplex scheme [15]. Therefore, SU’s average throughput \( R_1 \) of this case can be written as:

\[
R_1 = 2 \cdot \frac{T_t}{T_p} \cdot D_{0, TR},
\]

(13)

where \( D_{0, TR} = \log_2(1 + \frac{\text{SNR}_{x,s}}{\text{SNR}_{y,s} + \text{SNR}_{x,r} + \text{SNR}_{y,r}}) \) is SU’s maximum achievable data rate in TR mode in the OFF state [38].

**C.2**

This case happens due to two situations. First, SU makes an incorrect prediction with probability \( P_{pf} \) and decides to transmit in TS mode. Second, the SU decides to transmit in TR mode initially with probability \( P_{pd} \). However, it switches to TS mode after it falsely senses the channel to be busy with probability \( P_{pf} \). The occurrence probability of this case is given by:

\[
\Pr[C_2] = \mu_0 \cdot (P_{pf} + P_{pd} \cdot P_f) \cdot \Pr[L_0 > T_t] = \mu_0 \cdot (1 - P_{pf} \cdot P_f) \cdot e^{-\frac{T_t}{T_p}}.
\]

(14)

In a conservative manner, the influence of SI on sensing in TS mode is considered over all time slots due to simultaneous sensing-and-transmission. Since the sensing operation in each time slot is independent, the event where SU transmits data follows binomial distribution with probability of occurrence given by \( P_{f,SI} \), where \( P_{f,SI} = 1 - P_{f,SI} \) is the complementary probability of false-alarm probability \( P_{f,SI} \). Thus, SU’s average throughput in this case is given by:

\[
R_2 = \sum_{j=1}^{M-1} \left( \begin{array}{c} M - 1 \\ j \end{array} \right) P_{f,SI}^j P_{f,SI}^{M-1-j} \frac{j}{M} \cdot D_{0, TS},
\]

(15)

where \( j \) refers to the number of time slots that the SU transmits data when the channel is idle; \( D_{0, TS} = \log_2(1 + \text{SNR}_{x,s}) \) is SU’s maximum achievable data rate in TS mode in the OFF state.

**C.3**

This case indicates that PU comes back during SU’s data transmission. SU decides to transmit in TR mode with probability \( P_{pd} \) and the initial sensing period gives an idle result with probability \( P_f \), where \( P_{pd} = 1 - P_{pd} \) is the complementary probability of prediction detection probability \( P_{pd} \). In this case, \( L_0 \) is larger than zero but smaller than \( T_t \). Thus, the occurrence probability of this case is given by:

\[
\Pr[C_3] = \mu_0 \cdot P_{pd} \cdot P_f \cdot \Pr[0 \leq L_0 < T_t] = \mu_0 \cdot P_{pd} \cdot P_f \cdot (1 - e^{-\frac{T_t}{T_p}}).
\]

(16)

SU’s average throughput of this case contains two parts, one without collision to PU and the other with collision to PU. The instantaneous average throughput of SU of this case is given by:

\[
R_3(L_0) = 2 \cdot \left( \frac{L_0}{T_p} \cdot D_{0, TR} + \frac{T_t - L_0}{T_p} \cdot D_{1, TR} \right)
= 2 \cdot \left( \frac{D_{0, TR} - D_{1, TR}}{T_p} \cdot \hat{L}_0 + \frac{T_t}{T_p} \cdot D_{1, TR} \right),
\]

(17)

where \( D_{1, TR} = \log_2(1 + \frac{\text{SNR}_{x,s}}{\text{SNR}_{y,s} + \text{SNR}_{x,r} + \text{SNR}_{y,r}}) \) is SU’s maximum achievable data rate in TR mode in the ON state [38]; \( \hat{L}_0 \) is the normalised recurrence time of the OFF state, whose probability density function (p.d.f) is given by:

\[
\hat{L}_0 = \frac{\frac{1}{\lambda_p} e^{-\frac{t}{\lambda_p}}}{\int_0^{\infty} \frac{1}{\lambda_p} e^{-\frac{t}{\lambda_p}} dt} = \frac{1}{\lambda_p} e^{-\frac{t}{\lambda_p}}.
\]

(18)

Since \( \hat{L}_0 \) is a random variable, eq. (17) is the SU’s throughput with a determined \( L_0 \). Its expected value is expressed as:

\[
E(R_3) = \int_0^{T_t} t \cdot R_3(\hat{L}_0) dt
= \int_0^{T_t} t \cdot 2 \left( \frac{\hat{L}_0}{T_p} D_{0, TR} + \frac{T_t - \hat{L}_0}{T_p} D_{1, TR} \right) dt
= 2 \cdot \frac{D_{0, TR} - D_{1, TR}}{T_p} \cdot \lambda_0 - (T_t + \lambda_0) e^{-\frac{T_t}{\lambda_0}} + 2 \cdot \frac{T_t}{T_p} \cdot D_{1, TR}.
\]

(19)

**C.4**

Case C.4 happens when SU makes a correct prediction that PU will come back in the next \( M \) time slots and transmit in TS mode with probability \( P_{pd} \), or it decides to transmit in TR mode but the initial sensing period gives a busy result with probability \( P_{pf} \). The occurrence probability of this case is given by:

\[
\Pr[C_4] = \mu_0 \cdot (P_{pf} + P_{pd} \cdot P_f) \cdot \Pr[0 \leq L_0 < T_t] = \mu_0 \cdot (1 - P_{pf} \cdot P_f) \cdot (1 - e^{-\frac{T_t}{T_p}}).
\]

(20)

In the analysis of SU’s average throughput of TS mode, if PU comes back during a time slot, we assume that this certain time slot is in the ON state. Since the difference of throughput in one single time slot is extremely small, we can use discrete function to express SU’s average throughput. SU’s average throughput of this case is more complicated than that of Case C.2 since SU can transmit \( j \) (\( 1 \leq j \leq \left\lfloor \frac{T_t}{T_p} \right\rfloor \)) time slots in the OFF state and \( k \) (\( 1 \leq k \leq M - 1 - \left\lfloor \frac{T_t}{T_p} \right\rfloor \)) time slots in the ON state. \( \left\lfloor \cdot \right\rfloor \) is the floor function that takes as input a real number \( x \) and gives as output the greatest integer less than or equal to \( x \). Due to the floor function, it is complicated to find the expected value of \( R_4 \). Therefore, we use \( E(L_0) \) to approximate the expected value of \( R_4 \) expressed as:

\[
E(R_4) \approx \sum_{j=1}^{\left\lceil \frac{E(L_0)}{T_p} \right\rceil} \left( \frac{E(L_0)}{T_p} \right)^j \frac{E(L_0)}{M} \cdot D_{0, TS}
+ \sum_{k=1}^{M-1-\left\lceil \frac{E(L_0)}{T_p} \right\rceil} \left( M - 1 - \left\lceil \frac{E(L_0)}{T_p} \right\rceil \right) \frac{k}{M} \cdot D_{1, TS},
\]

(21)

where \( k \) refers to the number of time slots that SU transmits in the ON state; \( P_{d,SI} = 1 - P_{d,SI} \) is the complementary probability of detection probability \( P_{d,SI} \). \( D_{1, TS} = \log_2(1 + \frac{\text{SNR}_{x,s}}{\text{SNR}_{y,s} + \text{SNR}_{x,r} + \text{SNR}_{y,r}}) \) is SU’s maximum achievable data rate in TS mode in the ON state.

**C.5**
In this case, SU’s whole transmission frame falls into the ON state. This case happens when SU makes an incorrect prediction and then falsely senses the channel to be idle with probability \( P_{pd} \cdot P_{d} \). We define the forward recurrence time of the ON state as \( L_1 \), which refers to the length from the start of SU’s transmission to the end of the ON state. \( L_1 \) follows exponential distribution with parameter \( \lambda_1 \). Thus, the occurrence probability of this case is given by:

\[
\Pr[C_5] = \mu_1 \cdot P_{pd} \cdot P_{d} \cdot \Pr[L_1 \geq T_i] = \mu_1 \cdot (1 - P_{pd} \cdot P_{d}) \cdot e^{-\frac{T_i}{\lambda_1}},
\]

where \( \mu_1 = \frac{\lambda_1}{\lambda_0 + \lambda_1} \) is the occurrence probability of the ON state. SU’s average throughput of this case can be written as:

\[
R_5 = 2 \cdot \frac{T_i}{T_p} \cdot D_{1,TR}.
\]

- \( C_6 \)

This case designates that SU decides to transmit in TS mode with probability \( P_{pd} \) while PU is occupying the channel during SU’s transmission duration, or SU falsely decides to transmit in TR mode but the initial sensing period detects the presence of PU with probability \( P_{pd} \cdot P_d \). The occurrence probability of this case is given by:

\[
\Pr[C_6] = \mu_1 \cdot (P_{pd} + P_{pd} \cdot P_d) \cdot \Pr[0 \leq L_1 < T_i] = \mu_1 \cdot (1 - P_{pd} \cdot P_d) \cdot e^{-\frac{T_i}{\lambda_1}}.
\]

SU’s average throughput of this is similar to that of case \( C_2 \) in terms of derivation, which can be written as:

\[
R_6 = \sum_{k=1}^{M-1} \left( \frac{M-1}{k} \right) P_{d,SI}^k P_{d,SI}^{(M-1-k)} \frac{k}{M} \cdot D_{1,TS}.
\]

- \( C_7 \)

This case happens when PU leaves the channel during SU’s transmission in TR mode. The occurrence probability is similar to that of case \( C_6 \), which is given by:

\[
\Pr[C_7] = \mu_1 \cdot P_{pd} \cdot P_d \cdot \Pr[0 \leq L_1 < T_i] = \mu_1 \cdot P_{pd} \cdot P_d \cdot (1 - e^{-\frac{T_i}{\lambda_1}}),
\]

Then SU’s average throughput of this case is given by:

\[
R_{7}(\hat{L}_1) = 2 \cdot \left( \frac{D_{1,TR} - D_{0,TR}}{T_p} \cdot \hat{L}_1 + \frac{T_i}{T_p} \cdot D_{0,TR} \right),
\]

where \( \hat{L}_1 \) is the normalised recurrence time of the OFF state, whose p.d.f is given by:

\[
\begin{align*}
\hat{l}_1 &= \frac{1}{\lambda_1} e^{-\frac{T_i}{\lambda_1}} \\
&= \frac{1}{\lambda_1} e^{-\frac{T_i}{\lambda_1}} dt \\
&= \frac{1}{\lambda_1} e^{-\frac{T_i}{\lambda_1}}.
\end{align*}
\]

Then we can get the expected value of eq. (27) expressed as:

\[
\mathbb{E}(R_7) = \int_0^{T_i} t \cdot R_7(\hat{l}_1) dt = 2 \cdot \frac{D_{1,TR} - D_{0,TR}}{T_p} \cdot \lambda_1 - (T_i + \lambda_1) e^{-\frac{T_i}{\lambda_1}} + 2 \cdot \frac{T_i}{T_p} \cdot D_{0,TR}.
\]

- \( C_8 \)

In this case, PU leaves the channel during SU’s data transmission in TS mode. SU can either predicts to transmit in TS mode with probability \( P_{pd} \), or falsely predicts to transmit in TR mode and then switches to TS mode through getting a busy result in the initial sensing with probability \( P_{pd} \). SU’s average throughput of this case is then given by:

\[
\Pr[C_8] = \mu_1 \cdot (P_{pd} + P_{pd} \cdot P_d) \cdot \Pr[0 \leq L_1 < T_i] = \mu_1 \cdot (1 - P_{pd} \cdot P_d) \cdot (1 - e^{-\frac{T_i}{\lambda_1}}).
\]

The expected value of SU’s average throughput of this case is similar to that of case \( C_4 \) in terms of derivation, which can be expressed as:

\[
\begin{align*}
\mathbb{E}(R_8) &\approx \sum_{k=1}^{M-1} \left( \left( \frac{R(\hat{l}_1)}{k} \right) P_{d,SI}^k P_{d,SI}^{(M-1-k)} \frac{k}{M} \cdot D_{1,TS} \\
&+ \sum_{j=1}^{M-1} \left( M - 1 - \frac{R(\hat{l}_1)}{j} \right) P_{d,SI}^j \right) \frac{M-1-R(\hat{l}_1)}{M} \cdot D_{0,TS},
\end{align*}
\]

where \([.]\) is the ceiling function that maps \( x \) to the least integer greater than or equal to \( x \).

Finally, SU’s average throughput of our proposed NN-AMS scheme taking all cases into account is given by:

\[
R_{NN-AMS} = \sum_{i=1}^{8} \Pr[C_i] R_i.
\]

2) TR mode and TS mode: The throughputs of TR and TS modes are less complicated than that of NN-AMS scheme due to the non-existence of prediction. The throughput of TR mode contains four cases which are \( C_1, C_3, C_5, \) and \( C_7 \), respectively. Thus, SU’s average throughput of TR mode are given by:

\[
R_{TR} = \frac{\Pr[C_1]}{P_{pf}} \cdot R_1 + \frac{\Pr[C_3]}{P_{pd}} \cdot R_3 + \frac{\Pr[C_5]}{P_{pd}} \cdot R_5 + \frac{\Pr[C_7]}{P_{pd}} \cdot R_7.
\]

Similar to TR mode, SU’s average throughput of TS mode can be derived by considering the rest four cases, which is given by:

\[
R_{TS} = \frac{\Pr[C_2]}{1-P_{pf} \cdot P_f} \cdot R_2 + \frac{\Pr[C_4]}{1-P_{pd} \cdot P_f} \cdot R_4 + \frac{\Pr[C_6]}{P_{pf}} \cdot R_6 + \frac{\Pr[C_8]}{P_{pd} \cdot P_d} \cdot R_8.
\]

D. SU’s Non-Collision Average Throughput

In the last section, we derived SU’s average throughput taking the collision into account, where we assumed that PU has high tolerance for the collision to SU. However, PU can be less tolerant of collision in practice in order to protect its own transmission quality. In this section, we assume that PU has zero tolerance for collision and derive SU’s non-collision average throughput.
1) NN-AMS scheme: SU’s non-collision average throughput of our proposed NN-AMS scheme contains four cases C1, C2, C4 and C8 if there is no data transmission in the ON state. The occurrence probability and SU’s average throughput of C1 have been derived in eq. (12) and eq. (13), respectively. The occurrence probability and SU’s average throughput of C2 are given in eq. (14) and eq. (15), respectively. However, the occurrence probabilities and SU’s average throughputs of C4 and C8 are different since there should be no data transmission in the ON state (i.e., every sensing slot in the ON state successfully detects the presence of the PU). The occurrence probabilities of these two cases are given by:
$$Pr[C_{4,NC}] = \mu_0 \cdot (1 - P_{pd} \cdot P_f) \cdot (1 - e^{-\frac{T_s}{\lambda_T}}) \cdot P_{d,SI}^{M-1-\left[\frac{\mu_4}{T_s}\right]} \cdot P_{d,SI},$$
(35)
$$Pr[C_{8,NC}] = \mu_1 \cdot (1 - P_{pd} \cdot P_d) \cdot (1 - e^{-\frac{T_s}{\lambda_T}}) \cdot P_{d,SI}^{M-1-\left[\frac{\mu_8}{T_s}\right]} \cdot P_{d,SI}.$$
(36)
SU’s average throughput of these two cases are the throughputs in the OFF state, which are given by:
$$E(R_{4,NC}) \approx \sum_{j=1}^{\frac{\mu_4}{T_s}} \left[\frac{e^{\lambda_T}}{T_s}\right]^j P_{f,SI}^j P_{f,SI}^{M-1-\left[\frac{\mu_4}{T_s}\right]} \cdot \frac{j}{M} \cdot D_{0,TS},$$
(37)
$$E(R_{8,NC}) \approx \sum_{j=1}^{\frac{\mu_8}{T_s}} \left[M - 1 - \left[\frac{\mu_8}{T_s}\right]\right] P_{f,SI}^j \cdot \frac{j}{M} \cdot D_{0,TS}.$$
(38)
Therefore, SU’s non-collision throughput of NN-AMS scheme taking all cases into account is given by:
$$R_{NN-AMS,NC} = Pr[C_1] \cdot R_1 + Pr[C_2] \cdot R_2 + Pr[C_{4,NC}] \cdot R_{4,NC} + Pr[C_{8,NC}] \cdot R_{8,NC}. \quad (39)$$

2) TR mode and TS mode: The non-collision throughput of TR mode and TS mode can be simply derived by removing the prediction probability from that of NN-AMS scheme. The non-collision scenario for TR mode is C1 whose corresponding throughput is given by:
$$R_{TR,NC} = \frac{Pr[C_1] \cdot R_1}{P_{pf}}. \quad (40)$$
In TS mode, the non-collision scenarios are C2, C4 and C8 whose corresponding throughput is given by:
$$R_{TS,NC} = \frac{Pr[C_2] \cdot R_2 + Pr[C_{4,NC}] \cdot R_{4,NC}}{1 - P_{pf} \cdot P_f} + \frac{Pr[C_{8,NC}] \cdot R_{8,NC}}{1 - P_{pd} \cdot P_d}. \quad (41)$$

E. Collision probability

Generally, two different events can lead to collision. First, PU comes back during the SU’s data transmission, for example, C3 in Fig. 8. This kind of event occurs when SU transmits in TR mode. Second, SU fails to detect the presence of PU and starts to transmit data as in C5 and C6 in Fig. 8. This type of collision occurs both in TR and TS mode, whose occurrence probability depends on the detection probability.

1) NN-AMS scheme: The collision probability of NN-AMS scheme consists of all cases except C1 and C2. If SU chooses to transmit in TR mode, collision probabilities are the same as the occurrence probabilities we derived before, which are given by eq. (16), eq. (22) and eq. (26), respectively. Note that if SU chooses to transmit in TS mode, we use the complementary probability of non-collision case (i.e., SU successfully detects all the busy channel state) to calculate the collision probability. Collision probabilities of C4, C6 and C8 are given by:
$$Pr[C_{4,C}] = \mu_0 \cdot (1 - P_{pd} \cdot P_f) \cdot (1 - e^{-\frac{T_s}{\lambda_T}}) \cdot P_{d,SI}^{M-1-\left[\frac{\mu_4}{T_s}\right]} \cdot P_{d,SI},$$
(42)
$$Pr[C_{6,C}] = \mu_1 \cdot (1 - P_{pd} \cdot P_d) \cdot (1 - e^{-\frac{T_s}{\lambda_T}}) \cdot P_{d,SI}^{M-1}. \quad (43)$$
$$Pr[C_{8,C}] = \mu_1 \cdot (1 - P_{pd} \cdot P_d) \cdot (1 - e^{-\frac{T_s}{\lambda_T}}) \cdot P_{d,SI}^{M-1-\left[\frac{\mu_8}{T_s}\right]} \cdot P_{d,SI}. \quad (44)$$
Then the collision probability of NN-AMS scheme considering all cases is given by:
$$P_{NN-AMS,C} = Pr[C_3] + Pr[C_{4,C}] + Pr[C_5] + Pr[C_{6,C}] + Pr[C_{7,C}] + Pr[C_{8,C}]. \quad (45)$$

2) TR mode and TS mode: The collision probability of TR mode contains three cases, which are C3, C5 and C7, respectively. However, the prediction error should be removed from these probabilities. Thus, the collision probability of TR mode is given by:
$$P_{TR,C} = \frac{Pr[C_3] + Pr[C_5] + Pr[C_7]}{P_{pd}}. \quad (46)$$
Similarly, the collision probability of TS mode containing the other three cases is given by:
$$P_{TS,C} = \frac{Pr[C_3] \cdot P_{pf} + Pr[C_5] \cdot P_{pf} + Pr[C_7]}{1 - P_{pd} \cdot P_d}. \quad (47)$$
LAT protocol, respectively. We set the sampling frequency $\omega_s = 6$ MHz, SIS coefficient $\chi = 0.1$, SNR$_{s,s} = 10$ dB, SNR$_{s,p} = 9$ dB, $T_s = 0.001$ sec, and $\lambda_0 = \lambda_1 = 0.1$ sec. The number of training sets $N_{tr}$ and testing sets $N_{tt}$ are set to be 1000 and 30000, respectively.

A. Prediction Error Probability Analysis

Figure. 9 depicts prediction error probability as a function of the number of time slots predicted $M$, given that the sensing is perfect. We tested the prediction ability of our NN predictor with $N_{tt}$ testing sets and calculated the average prediction error probabilities. As shown in Fig. 9, all the three error probabilities generally increase with $M$. The reason is that the more time slots the NN predictor predicts, the larger amount of prediction uncertainty and randomness it faces. The prediction error probability decreases when $\lambda_0$ and $\lambda_1$ increases. This indicates that the NN predictor can predict more time slots with the same prediction error probability when $\lambda_0$ and $\lambda_1$ increase. It can also be seen that the prediction false-alarm probability and prediction detection probability are related to $\lambda_0$ and $\lambda_1$, respectively. These two probabilities are close to the average prediction probability of the same $\lambda_0$ and $\lambda_1$.

Figure. 10 shows average prediction error probability $P_e$ versus the number of iterations of the input vector updating process. It can be seen from Fig. 10 that the long-term prediction performance is stable. After several iterations of update, the prediction error probability increases slightly due to the occurrence of $C_3$. However, the subsequent predictions are not influenced by the occurrence of $C_3$ since the input vector is updated frequently. Note that the prediction error probabilities in Fig. 10 are average values obtained by averaging over 30000 testing sets (i.e., 30000 channels), which means that $C_3$ may occur at any time of transmission. Thus, the prediction error probability does not decrease to the initial level (i.e., the prediction error probability of the first transmission). In practice, the prediction error probability will restore to the initial level after the incorrect inputs are deleted.

B. SU’s Average Throughput Analysis

Figure. 11 illustrates SU’s average throughput versus $M$ assuming that the PU has high tolerance for collision. It can be seen from Fig. 11 that SU’s average throughputs of all three modes increase first and then decrease. The reason is that the ratio $\frac{D_0}{T_s}$ dominates at first. However, SU’s average throughput decreases after the collision probability increases. Note that SU’s average throughput of TR mode decreases slightly because the difference between $D_0$ and $D_1$ is small in this case. The TS mode has the lowest throughput because SU cannot simultaneously transmit and receive data while sensing the channel. SU’s average throughput of TR mode is the highest among all the three modes, because SU does not stop its data transmission in TR mode during collision to PU. Compared with TR mode, our proposed NN-AMS scheme has slightly lower SU’s average throughput, because SU’s switches to TS mode and stops transmitting data when it detects the return of the PU. This indicates that the SU rarely transmits data when the PU is occupying the channel, which protects PU’s data transmission quality. From Fig. 11, we can see that SU’s average throughput in NN-AMS scheme is 91% of that...
in TR mode when $M$ equals to 10.

In Fig. 12, we compare SU’s non-collision average throughput of all the three modes assuming that PU has zero tolerance for collision. SU’s average throughput of TS mode is slightly influenced by $M$ because it only depends on the sensing error probabilities and data rates. It can be seen that SU’s average throughput of NN-AMS scheme is very close to that of TR mode with different values of $M$ and $\chi$. With large values of $M$, SU’s average throughput of NN-AMS scheme is slightly larger than that of TR mode. This indicates that our proposed NN-AMS scheme can successfully detect the return of PU and switch to TS mode to avoid collision, and thus, increase the SU’s non-collision throughput. It can also be seen that SU’s average throughputs of NN-AMS scheme and TR mode both increase at first and then decrease after they reach a maximum value. We can find the optimal $M^*$ for both NN-AMS scheme and TR mode is 10 in this case. This indicates that SU can achieve its maximum non-collision average throughput when its transmission frame is set to be $T_p = MT_s = 0.01$ sec if $\lambda_0 = 0.11$ sec. Note that for difference values of $\lambda_0$ and $\lambda_1$, the optimal value of $M$ is different. Through jointly analysing Fig. 9 and Fig. 12, we can draw a conclusion that the optimal $M^*$ is larger for larger values of $\lambda_0$ and $\lambda_1$ due to lower prediction error probabilities. Since SU’s non-collision average throughput of NN-AMS scheme is close to that of TR mode, the optimal $M^*$ can be determined during NN training by using the same estimation method of TR mode [20].

Figure. 13 depicts the relationship between SU’s average throughput and SIS coefficient $\chi$, given that $M = 10$. Two different TS mode protocols are illustrated where the one with adaptive sensing threshold is proposed in [18] and the other with fixed sensing threshold is conventional TS mode. For fair comparison, we consider the same parameter setting as those in [18] where $\text{SNR}_{e,s} = 5 \text{ dB}$, $\text{SNR}_{e,p} = -5 \text{ dB}$, $w_sT_s = 200$ and $P_d = 0.9$. It can be seen that SU’s average throughputs of all three modes decrease with the increase of $\chi$. The TS mode proposed in [18] outperforms the one without considering sensing threshold adaptation in which SU’s average throughput quickly drops to zero with the growth of residual SI. Our proposed NN-AMS scheme has higher SU’s average throughput than that in TS mode because SUs can opportunistically switch to TR mode to boost their throughput when the channel is predicted to remain idle. Compared with TR mode, NN-AMS scheme has close SU’s average throughput for all values of $\chi$ but much lower collision probability which will be illustrated in Section IV-C.

C. Collision Probability Analysis

Figure. 14 depicts the collision probabilities of NN-AMS scheme, TR mode and TS mode. It can be seen from the figure that collision probabilities increase monotonically with the number of transmission slots. The reason is that a long transmission frame increases the probability that PU will come back to the channel during SU’s data transmission. The collision probability of TS mode is extremely low (i.e., close to zero) in practice. Thanks to the adaptive switch between TR and TS mode, the NN-AMS scheme gains much lower collision probability than that of TR mode. When $M$ is small, the collision probability of NN-AMS scheme is very close to that of TS mode due to the low prediction error probabilities. It can be seen that the collision probability of NN-AMS scheme is only 8% of that of TR mode when $M = 10$. Through jointly analysing Fig. 11 to Fig. 14, we conclude that our proposed NN-AMS scheme can dramatically reduce the collision probability at the expense of slightly lower SU’s average throughput than that of conventional TR mode.

D. PU Traffic Load

Next we consider the impact of PU traffic on the performance of the proposed NN-AMS scheme. We fix $\lambda_0$ to 0.1 sec and change the value of $\lambda_1$ to adjust the average length of the ON state. Let the occurrence probability of the ON state $\mu_1$ denote PU traffic load, which also represents how frequently PU uses the channel. Figure. 15 and Fig. 16 depict SU’s throughput and collision probability versus PU traffic load, respectively. It can be seen that, with the increase of PU traffic load (i.e., PU’s channel is occupied more frequently), SU’s throughput decreases and reaches the minimum value...
when PU traffic load is close to PU traffic load. Meanwhile, the collision probabilities of NN-AMS scheme and TR mode also decrease with $\mu_1$ since the occurrence of $C_3$ and $C_4$ reduces. It can be seen that both SU’s throughput and collision probability of TS mode are hardly influenced by PU traffic load due to its simultaneous transmission and sensing. Our proposed NN-AMS scheme can have the benefit of improving SU’s throughput while reducing collision probability under different values of PU traffic load.

V. CONCLUSION

In this paper, we proposed a new application of NN in FD-CRNs. A multi-layer NN predictor and an AMS scheme were designed. By testing the prediction performance, we analysed the influence of sensing error on the prediction error. We also derived SU’s average throughputs and collision probabilities of our proposed NN-AMS scheme, conventional FD-TR and FD-TS modes. Simulation results show that SU can achieve almost the same average throughput in NN-AMS scheme as that of TR mode when PU is less tolerant of collision. Meanwhile, our proposed NN-AMS scheme reduces the collision probability by up to 92% compared with conventional TR mode.

REFERENCES


Yirun Zhang (S’18) received the B.Sc. degree in communication engineering from Tongji University, Shanghai, China, in 2016, the M.Sc degree with distinction in Mobile and Personal Communications from King’s College London, U.K., in 2017. Since 2018, he has been working towards the Ph.D. degree with the Department of Informatics, Centre for Telecommunication Research (CTR), Kings College London, U.K., where he has been strongly involved in the EPSRC SENSE Project. His current research interests lie in the area of full-duplex wireless communications, cognitive radio networks, and machine learning techniques.

Jiancao Hou (S’09–M’15) received the B.Sc. degree in information science (honor program) from China Agricultural University, Beijing, China, in 2008, the M.Sc. degree with distinction in radio frequency communication systems from the University of Southampton, Southampton, U.K., in 2009, and the Ph.D. degree in electrical engineering from the University of Surrey, Guildford, U.K., in 2014. From 2014 to 2016, he was a Research Fellow with the Institute for Communication Systems, University of Surrey. Since 2017, he has been a Research Associate with the Department of Informatics, Kings College London, London, U.K., where he has been strongly involved in the EPSRC SENSE Project. His current research interests include radio resource allocation/interference management for full-duplex dense networks, stochastic geometry theory, and MIMO beamforming techniques. He also serves as a TPC member for IEEE flagship conferences, such as ICC and GLOBECOM.

Vahid Towhidlou received the B.Sc. degree in electrical engineering from Khajeh Nasir University of Technology, Tehran, Iran, in 1997, the M.Sc. degree from Ferdowsi University, Mashhad, Iran, in 2000, and the Ph.D. degree from Kings College London, U.K., in 2016, both in telecommunication engineering. He has been working as a Research Associate with the Department of Informatics, Centre for Telecommunication Research (CTR), Kings College London, since 2016. His current research interests lie in the area of cognitive radio, communication for vehicular networks, and full duplex wireless communication.

Mohammad R. Shikh-Bahaei (S’96M’00SM’08) received the B.Sc. degree from the University of Tehran, Tehran, Iran, in 1992, the M.Sc. degree from the Sharif University of Technology, Tehran, in 1994, and the Ph.D. degree from Kings College London, U.K., in 2000. He has worked for two start-up companies, and for National Semiconductor Corp., Santa Clara, CA, USA (now part of Texas Instruments Inc.), on the design of third-generation (3G) mobile handsets, for which he has been awarded three U.S. patents as inventor and co-inventor, respectively. He joined Kings College London, as a Lecturer (2002), where he is now a Full Professor. He has been engaged in research in the area of wireless communications and signal processing for 25 years both in academic and industrial organizations. His research interests include full duplex and cognitive dense networking, visual data communications over the IoT, applications in healthcare, and secure communication over autonomous vehicle/drone networks. He is a fellow of the IET and Founder and Chair of the Wireless Advanced (formerly SPWCS) International Conference from 2003 to 2018.