EURASIP Journal on
Wireless Communications and Networking
a SpringerOpen Journal

RESEARCH Open Access

CSIT: channel state and idle time predictor using a neural network for cognitive LTE-Advanced network

Adnan Shahid, Saleem Aslam, Hyung Seok Kim and Kyung-Geun Lee*

Abstract

Cognitive radio (CR) is a novel methodology that facilitates unlicensed users to share a licensed spectrum without interfering with licensed users. This intriguing approach is exploited in the Long Term Evolution-Advanced (LTE-A) network for performance improvement. Although LTE-A is the foremost mobile communication standard, future underutilization of the spectrum needs to be addressed. Therefore, dynamic spectrum access is explored in this study. The performance of CR in LTE-A can significantly be enhanced by employing predictive modeling. The neural network-based channel state and idle time (CSIT) predictor is proposed in this article as a learning scheme for CR in LTE-A. This predictive-based learning is helpful in two ways: sensing only those channels that are predicted to be idle and selecting the channels for CR transmission that have the largest predicted idle time. The performance gains by exploiting CSIT prediction in CR LTE-A network are evaluated in terms of spectrum utilization, sensing energy, channel switching rate, packet loss ratio, and average instantaneous throughput. The results illustrate that significant performance is achieved by employing CSIT prediction in LTE-A network.

Keywords: Cognitive radio; LTE-Advanced; Slot(s) state; Idle time; Neural network and multilayer perceptron

1 Introduction

Long Term Evolution-Advanced (LTE-A) is an evolving next generation mobile network that guarantees high data rates up to 1 Gbps for low mobility and 100 Mbps for high mobility [1]. International Telecommunication Union (ITU) issued requirements in terms of International Mobile Telecommunication-Advanced (IMT-Advanced). In order to achieve the requirements, various optimized wireless techniques proposed in literature for LTE Rel 8. Among them, carrier aggregation (CA) is the technique that is concerned with the escalation of bandwidth of the user equipment (UE) from 20 to 100 MHz [2]. Hence, by inculcating the concept of CA in LTE, the peak-supported transmission rate will be 1 Gbps in downlink and 500 Mbps in uplink, which is in accordance with the limits imposed by the ITU. However, despite the potential advantages of CA in terms of throughput, a major concern with the spectrum underutilization due to increased UEs in the future has to be addressed.

The Federal Communication Commission regulates the reservation of spectrum resources on a long-term basis to the primary (license) network. However, the fixed spectrum assignment policy results in the underutilization of the spectrum in the primary network that leads to unutilized portions [3]. These unutilized portions are termed white spaces or spectrum holes (SH) in the literature. This underutilization forces the development of dynamic spectrum access (DSA) in the primary network for efficient spectrum utilization. The DSA can be employed by inculcating cognitive radio (CR) (secondary) users in the primary network to enhance the capacity and improve spectrum utilization [4,5].

CR operation is illustrated as a simple 'cognitive cycle' as in [4], which consists of three tightly interconnected phases. The first phase is the *sensing*, during which CR senses the primary network channels and discovers the statistic of sensed ones. The statistics includes channel occupancy, idle time, power level, modulation schemes, etc. These statistics help in making the operation of CR possible, i.e., deciding where to operate, how much power is needed, and which modulation scheme is

* Correspondence: kglee@sejong.ac.kr
Department of Information and Communication Engineering, Sejong University, Sejong, Republic of Korea

needed, etc. Therefore, the sensing task is the most critical part in the overall cognitive cycle. Various spectrum sensing mechanisms have been proposed in the literature such as energy detection, matched filter detection, cyclo-stationary feature detection, and cooperative spectrum sensing [3]. The optimized sensing method that gives best sensing results, e.g., cooperative spectrum sensing, would be adopted because erroneous sensing results contribute to wrong decisions in the third phase, i.e., *decision*. This erroneous information will also lead to drastic interference to the licensed network. During the second phase, *predictive modeling*, the previously sensed information is stored in the database, and based upon the past history, future status about the channels is predicted and forwarded to the *decision* phase block. The third and the last phase is the decision, which concerns with setting the transmission parameters, such as channel selection, transmit power, and modulation scheme. The concern of the implication of predictive modeling in the second phase is to have optimum decisions in the third phase.

CR is termed as software-controlled radio that can adjust its parameters on run time with respect to the changing environment. These software-based capabilities are provided by the cognitive engine (CE) [6]. The CE forces the software-defined radio to adjust the parameters based upon the knowledge base. The information in the knowledge base is not only the currently observed information but also the past observations. The actual intelligence in CR is possible by learning via the information in the knowledge base. Based upon the above discussion, it is concluded that integration of CE with cognitive cycle is important for providing actual intelligence for CR. Various learning mechanisms are available for use in the CR environment ranging from simple lookup search to machine learning models, like the neural network (NN) and hidden Markov model.

Most existing works exploring DSA in LTE-A focus on the components used for inculcating CR concepts. The main contributions of this study are categorically elaborated as follows:

1. The NN-based CSIT predictor is proposed within the context of CR in LTE-A. Although channel state prediction via NN is explored in the CR domain, our proposed predictor is unique in the sense that it caters both channel state and idle time which have not been explored in the literature so far.
2. The proposed CSIT predictor is designed via multilayer perceptron (MLP), a class of NN, which does not require *a priori* knowledge of the primary network traffic characteristics. Since the primary network traffic characteristics of LTE-A is not known beforehand, we have exploited MLP for the prediction.
3. The primary user (PU) traffic is modeled in statistical domain, i.e., by *Poisson* and *Pareto* stochastic processes, as described in Section 6. The concern of employing the aforementioned stochastic processes is that they closely approximate the behavior of PU in mobile communication environment.
4. Finally, the proposed CSIT predictor is compared with nonpredictive CR in LTE-A in terms of five performance measures, i.e., spectrum utilization, sensing energy, channel switching rate, packet loss ratio (PLR), and average instantaneous throughput. In addition, the limitation of the study is also presented.

The rest of this paper is structured as follows. Related work is presented in Section 2, and the system model is outlined in Section 3. A review of NN is projected in Section 4, and the proposed MLP-based CSIT prediction is elaborated in Section 5. The simulation and analysis in terms of the accuracy of the proposed predictor and performance improvement are listed in Section 6. Finally, Section 7 concludes the article.

2 Related work

In the literature [7-9], the authors surveyed the machine learning algorithms applicable to each phase of the cognitive cycle in the CR network. The closest work relating to our study that involves prediction via NN in CR domain is given in [10-14]. Moreover, incorporating CR concepts in LTE-A is presented in [15,16].

In [10] the authors exploited the Elman recurrent neural network for spectrum prediction based on the multivariate time series. Elman recurrent is a class of NN, and the difference with feedforward NN lies in the inclusion of recurrent layer(s) within it. In their proposed model, time series is estimated using the cyclostationary feature detection and then the spectrum is predicted by applying that series to Elman recurrent NN. The authors in [11] exploited the MLP for spectrum prediction in CR. The concern of their study is to reduce the sensing time for CR, i.e., predicting the status of the channels just before brutally applying the sensing. Another approach towards reducing the sensing time and hence increasing the transmission rate is projected in [12]. In their proposed model, the sensing task is supervised by the network, i.e., by giving the intelligent sensing matrix to the CR. This supervised learning results in significant sensing time reduction and, hence, increased throughput. The NN-based throughput learner for CR is proposed in [13] where authors exploited NN for predicting the appropriate transmission rate that can be used for a concerned channel. They presented the basic and extended throughput prediction model using NN. The basic throughput prediction is a simple prediction whereas the extended one also takes into account the geographical/time location of users. Another traffic prediction for opportunistic spectrum access is presented in [14]. They proposed a selective opportunistic spectrum access in which the probability of channels appearing to be idle is computed on the statistical basis. On the basis of the predicted idle time, the best sensing order is formulated. They evaluated their results in terms of packet loss ratio and throughput.

According to the best of our knowledge, our proposed CSIT MLP prediction model is unique in the CR domain and specifically for LTE-A. However, a nominal work done in inculcating CR into LTE-A is presented here after. The DSA framework for LTE-A network is presented in [15,16]. The authors in [15] highlighted the existing blocks in LTE-A network for supporting CR concepts. According to them, DSA can be indulged just by introducing a few blocks within the existing network. The authors in [16] utilize the framework presented in [15] and introduce the opportunistic spectrum access in LTE-A network. Their work illustrates the adoption of a geo-location database within LTE-A network that gathers the cooperative information from the CR users with its surrounding environment.

3 System model

3.1 Introducing prediction-based CR in LTE-A network

The system model comprises of DSA overlay in LTE-A network as shown in Figure 1. The DSA is supported by the introduction of cognitive user equipment ($\mathrm{CU}_{\mathrm{sense}}$). We are projecting a new prediction-based CR user equipment, i.e., $\mathrm{CU}_{\mathrm{predict}}$. The difference between the two is that the former one is the normal CR whose job is to sense the primary channels and discover the opportunities in terms of SHs, whereas the latter one is the predictive CR which senses only those channels that are predicted to be idle and/or has larger predicted idle time. This predictive modeling results in significant performance improvement of the existing LTE-A network. The prediction regarding future channel statistics is carried out by having the past history on each predictive CR in the secondary network. Therefore, each $\mathrm{CU}_{\mathrm{predict}}$ houses the primary channel's past observation. Information about the past primary channels statistics are repeatedly acquired on the control channel. Furthermore, we also assume that the projected predictive secondary network nodes operate in a docitive fashion and forward their collaborative predicted results to the primary base station (eNB) where docitive network is an emerging paradigm for DSA in which the UEs teach each other that leads to reduced complexity and fine-tuned decisions [17,18].

As far as the interference to the primary network is concerned, $\mathrm{CU}_{\mathrm{predict}}$ applies sensing to the slot(s) that are predicted to be more idle, in terms of idle time slot (s), before using them for transmission. Therefore, there is a less chance of interference being inducted into the primary network due to the implication of docitive and predictive modeling approach. However, if the interference experienced by the PUs due to secondary users ($\mathrm{S_eU}$) exceeded beyond the specified threshold, then eNB alerts the concerned $\mathrm{CU}_{\mathrm{predict}}$ to change the slot(s) for transmission to avoid the interference.

3.2 System architecture of the proposed $\mathrm{CU}_{\mathrm{predict}}$

The proposed system architecture of $\mathrm{CU}_{\mathrm{predict}}$ is illustrated in Figure 2. The basic idea of our study is to predict the CSIT of the primary channels in the LTE-A network without just relying on the current sensed observations. The CSIT prediction is done by employing NN as a learning of $\mathrm{CU}_{\mathrm{predict}}$. The brief operation of $\mathrm{CU}_{\mathrm{predict}}$ for future prediction using NN as learner is as follows:

1. Firstly, the NN in $\mathrm{CU}_{\mathrm{predict}}$ is trained properly with the appropriate data set and once it is trained, its complexity is significantly reduced.

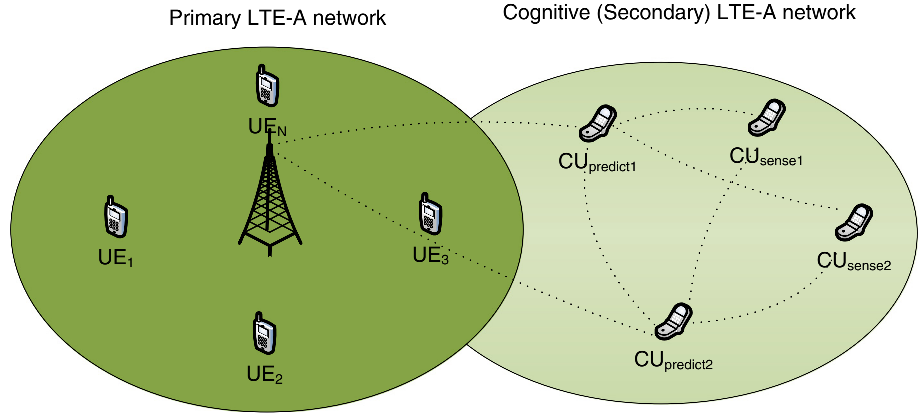
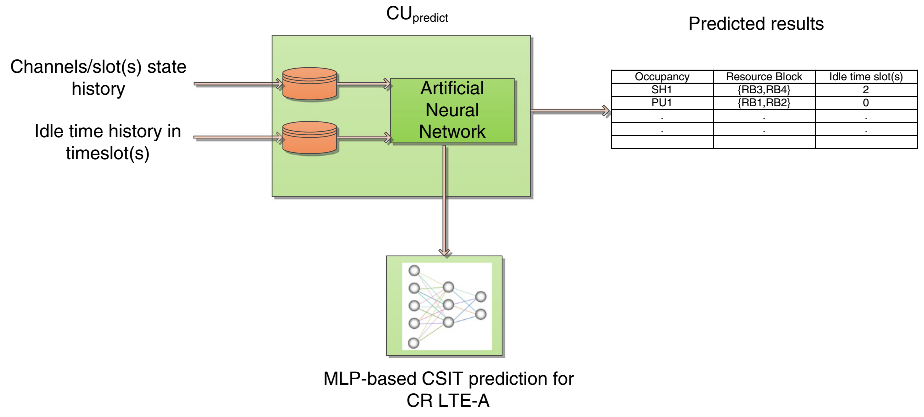


Figure 1 DSA in LTE-A network.


CU$_{predict}$
Channels/slot(s) state history
Idle time history in timeslot(s)
Artificial Neural Network
Predicted results
MLP-based CSIT prediction for CR LTE-A

Occupancy	Resource Block	Idle time slot(s)
SH1	{RB3,RB4}	2
PU1	{RB1,RB2}	0
.	.	.
.	.	.

Figure 2 System architecture of CU$_{predict}$.

2. Secondly, based upon the acquired CSIT history, the predicted results are compiled. The predicted results include slot(s) status and idle time slot(s) of the primary channels.

Here, we are considering the past observations of primary channels on a per resource block (RB) basis, where RB is the basic unit of the spectrum in LTE-A. The predicted channels statistics are grouped in to either SH or PU, where each one consists of multiple RBs, as shown in Figures 2 and 3 PU1={RB1, RB2}, SH1={RB3, RB4}, etc. However, we discussed throughout this paper in terms of the sensed/scanned slot(s). This is due to the fact that we consider the basic unit of the spectrum in LTE-A in the time-slotted fashion as shown in Figure 4a. Each CR in the LTE-A network, either CU$_{sense}$ or CU$_{predict}$, can sense slot(s) in a particular sensing time slot. The difference lies here that CU$_{sense}$ has to sense all the RBs, whereas CU$_{predict}$ applies sensing to only those slot(s) that are predicted to be idle. Furthermore, the difference between the sensing time slot and available slot (s) to CR is that the former is in accordance to measurement gap repetition period (MGRP) defined in the standard and the latter is just to have different interpretations of the RBs set in a time-slotted fashion. Noticeably, the sensing time slot is directly associated with the gap pattern defined in standard for UE in RRC_CONNECTED state, i.e., measurement gap length (MGL) and MGRP. According to the standard, MGL is reserved for capturing the samples from a certain bandwidth and is fixed at 6 ms whereas MGRP is the gap duration during which MGL (samples extraction) and then data transmission is carried out. Moreover, MGRP is configurable in multiples in frame length (i.e., 10 ms) [19]. The relationship between MGL and MGRP is also illustrated in Figure 4b where MGRP is selected to be 40 ms. Within the gap period (sensing time slot), the scheduler does not allocate channels/slot(s) to UE. Also, the channels and slot(s) are used interchangeably as licensed and primary users in this study do.

3.3 Notation and assumptions

The notations presented in Table 1 are used throughout the rest of this paper.

4 Neural network

The human brain consists of real biological neurons that are interconnected in a sense to provide stunning

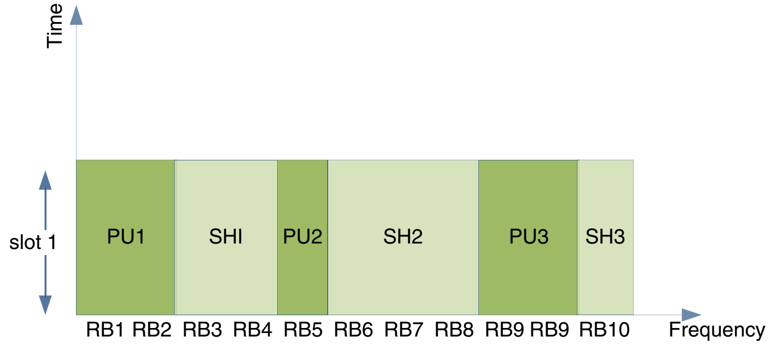


Figure 3 Sensed spectrum distribution.

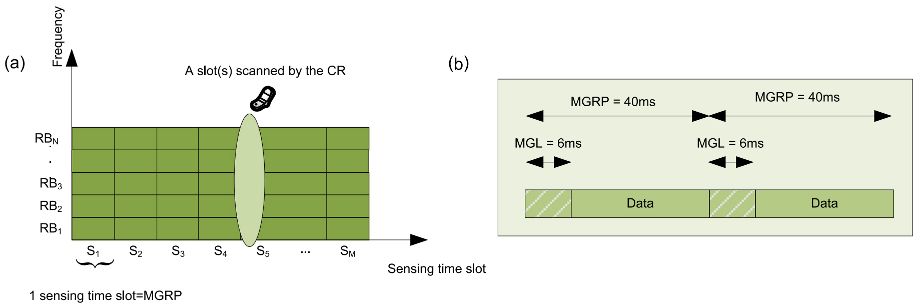


Figure 4 Interaction of S_eU and Gap pattern in LTE-A network. (a) Interaction of S_eU with RBs in LTE-A network. **(b)** Gap pattern of sensing time slot in LTE-A network.

functionality. The functionality includes classification, optimization, decision making, etc. Although there is a significant development in information technology, still the intelligence provided by the advancement can never cross the real boundaries of human brain. On the other hand, NN are composed of artificial neurons that imitate the behavior of biological neurons in human brain. These artificial neurons are interconnected in a software structure and behave the same as biological neurons in the human brain. Therefore, the NN has been extensively applied to the areas that require cognitive tasks, such as predictive modeling and pattern classifications.

The generalized structure of NN consists of the three types of neuron layers, i.e., input layer, hidden layer, and output layer. Generally, the input layer acquires the data from the outside world and the output layer returns the data after passing through the hidden layer(s). NN structure can either be of *feedforward* or *recurrent* [20]. In feedforward structure, data flow from the input layer to the output layer without having any feedback connection to the backward layer(s). The classical example of feedforward structure is the perceptron listed in [21]. However in recurrent NN, there are connections between the output neuron to the input neuron of the same or the backward layer. The example being the Elman's recurrent neural network [22] in which the concept of recurrent NN is employed. In this study, we employ MLP, a feedforward structure of NN, which has been used widely in time series prediction [23] and binary prediction [24].

In the generalized NN, each neuron communicates with one another by having a direct connection between them. Every connection is specified by the weight $v_{\beta\alpha}$ which deliberates the influence of α neuron on β neuron. Each neuron computes the output using the specified activation function. As shown in Figure 5, the β neuron has α inputs from the previous layer neurons with their specified weights, i.e., $v_{\beta\alpha}$ and a bias input b_β. The

Table 1 Notations

Parameter	Meaning
T	Past observation length in terms of number of slot(s)
y_β^k	Output of the neuron β of the kth layer
$v_{\beta\alpha}$	Connection weights, connecting the neuron β of the kth layer to the neuron α in the (k −1)th layer
z_β^k	Weighted sum of inputs coming from the output neuron in the (k − 1) th layer
b_β	Bias input in the neuron β
O	Neuron in the output layer
$s_{\tau+1}$	The actual slot(s) state from the MLP-based CSIT predictor
$s^{\wedge}_{\tau+1}$	The desired slot(s) state
$i_{\tau+1}$	Actual predictor idle time slot(s)
$i^{\wedge}_{\tau+1}$	Desired idle time slot(s)
$\varphi^{'}()$	Activation function, i.e., 'purelin' for the output layer and 'log sigmoid' for the hidden layer
δ_β^k	Local gradient of the neuron β in the kth layer
e_s	Error between the desired and actual slot(s) state of the predictor
e_i	Error between the desired and the actual idle time slot(s) of the predictor
N_is	Total number of idle time slot(s) in the system
N_i	Total number of idle time slot(s) sensed by $\mathrm{CU}_\mathrm{sense}$
N_ip	Total number of idle time slot(s) sensed by $\mathrm{CU}_\mathrm{predict}$
X	Unit energy required for sensing a slot(s)
Y	Total number of slot(s) predicted to be idle
Z	Number of slots a $\mathrm{CU}_\mathrm{sense}$ sensed in a finite duration of time slot(s)
N_s	Total number of slot(s) to be sensed
C	Sensing threshold for the event PLR to occur
p_m	Probability of the mth slot(s) appearing to be idle
Δ_s	Time duration of the slot(s)
C_0	Referred transmission capacity of the SU
η_i^K	This corresponds to how fast the a SU can find the idle slot(s) among total of N_s

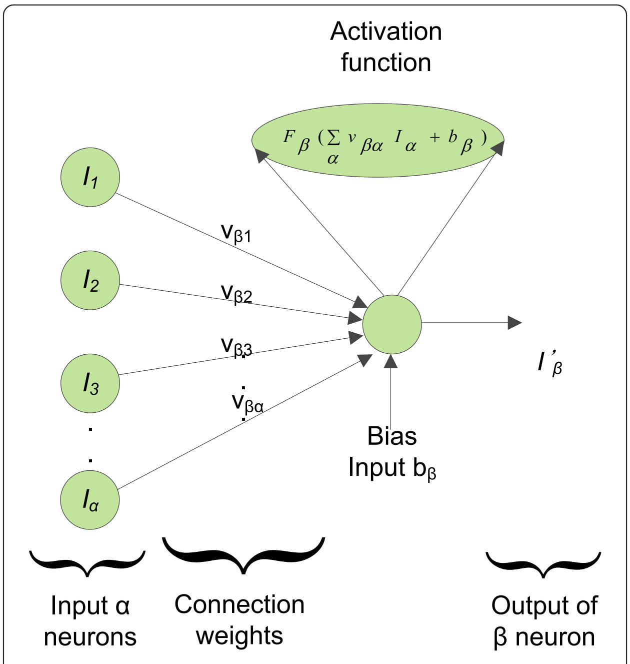


Figure 5 Generalized structure of *feedforward* NN.

activation function of the β neuron used to compute the output I'_β is given by

$$I'_\beta = F_\beta\left(\sum_\alpha v_{\beta\alpha} I_\alpha + b_\beta\right) \tag{1}$$

In any case, NN has to be configured in a manner so that it produces the desired outputs for any given set of inputs. The desired output can be achieved by adjusting the connection weights $v_{\beta\alpha}$ of all pairs of neurons (α, β). This procedure is termed as learning or training. NN should be learned in a manner so the error between seen and unseen data output is minimized, where unseen data are the data for which the NN has not been trained for. Furthermore, the learning process can be classified as either supervised or unsupervised learning. In the supervised learning mechanism, NN is given the teaching data base and on the basis of this, it adjusts connection weights to produce the desired output. This weight-changing procedure of the connection is termed as the *backpropagation* learning rule. This learning mechanism works by adjusting the connection link weights so that the error between the actual and the desired output is minimized. When this is done, NN is ready to give the output for any type of unseen data with least possible error. The other learning scheme, the unsupervised learning, is worked in the manner that does not require any real physical data set for the training but rather exploits the statistical representation of the input data for training NN. In this study, we use the supervised learning scheme for training the proposed MLP predictor.

5 CSIT prediction using MLP

The design specifications of CSIT predictor start by providing two types of inputs to MLP, i.e., primary channels/slot(s) state history and idle time history in time slot(s). The slot(s) state history is modeled as the bipolar binary series of length τ, i.e., $s^\tau = \{s_1, s_2, s_3,..., s_\tau\}$. The slot(s) can either be in one of the two states, i.e., *idle* or *busy*, where the idle state is represented by −1 and the busy state is represented by 1. The idle time history is modeled as an integral time series of length τ, i.e., $i^\tau = \{i_1, i_2, i_3,..., i_\tau\}$. Here we confined our analysis by assuming that the idle time slot(s) of the SH is to be in the integral range from {1 to 5}. Also, only the idle slot(s) in the vector **s** has an idle time slot(s) reference value in the corresponding vector **i**. The busy slot has 0 corresponding value in the idle time history vector **i**. This assumption is due to the fact that we are predicting the idle slot(s) in the RBs set and their idle duration in time slot(s). By training MLP using the above-mentioned past history vectors, i.e., **s** and **i**, it can predict the channels/slot(s) state and their idle duration in time slot(s). As far as the complexity of the proposed predictor is concerned, once the NN is trained properly, the computational complexity is significantly minimized.

5.1 MLP-based CSIT predictor design

MLP is the multilayer feedforward structure consisting of input layer, some hidden layers, and output layer. We have selected two sets of neurons in the input layer with each length τ for our proposed predictor, one set for capturing the slot(s) status, and the other one for the idle time slot(s) history. The overfitting problem in the predictor is avoided by fine tuning the neuron count in each layer, where the overfitting is a malfunction of MLP for the unseen data. The number of hidden layers and the neurons count in them depends upon the application. The best results are achieved by having two hidden layers, where the first one having 15 neurons and the other one having 20 neurons. The same specifications in terms of hidden layers and neuron count are used in [11], where the application under consideration was channel status prediction. The output layer consists of just two neurons, one for slot(s) state prediction and the other for the idle time slot(s) prediction. As far as the activation function is concerned, we have selected the log-sigmoid function for the hidden layers and pure linear function for the output layer because we want one output neuron in the range from {−1 to 1} and the other one in the range from {1 to 5}.

The output y_β^k from the neuron β in the kth layer is represented by the generalized activation function as

$$y_\beta^k = F_\beta\left(\sum_\alpha v_{\beta\alpha} z_\beta^k + b_\beta\right) \tag{2}$$

where

$$z_\beta^k = \sum_\alpha y_\alpha^{k-1} v_{\beta\alpha}^k \tag{3}$$

The parameter y_α^{k-1} corresponds to the links from α neuron in the back (k – 1)th layer.

5.2 MLP-based CSIT training

The training process of the proposed predictor is illustrated in Figure 6. The training data are organized into two streams each of length T time slot(s). Furthermore, both the streams are chopped into chunks of τ, i.e., s^τ and i^τ and their corresponding desired outputs, i.e., $s_{\tau+1}$ and $i_{\tau+1}$. Remember that s^τ incorporates slot(s) state values, whereas i^τ houses the idle time slot(s) values. Also, we are acquiring the history on a per-RB basis. This means that the chuck houses information regarding each RB. For each chuck, the output is computed by the activation functions at each hidden layer and the output layer. The achieved output from the predictor is expressed as $s_{\tau+1}$ and $i_{\tau+1}$. This step of NN is termed as *forward pass* in which the data flow from input to output by lingering with the neurons in the hidden layer. The difference between the desired output and the actual output for the two neurons of our proposed predictor is expressed as

$$e_s = s_{\tau+1} - \hat{s}_{\tau+1} \tag{4}$$

$$e_i = i_{\tau+1} - \hat{i}_{\tau+1} \tag{5}$$

We combined the two errors $\{e = e_s + e_i\}$ and fed that to the backpropagation block as shown in Figure 6. The concern of employing backpropagation for CSIT training is that we are using supervised learning, where supervised learning refers to the learning of NN in CR LTE-A in which the NN is trained with every possible data set. The job the backpropagation algorithm is to update the connection link weights, i.e., $v_{\beta\alpha}^k$ of each kth layer. This step is termed as the backpropagation. Noticeably, we are using log-sigmoid for hidden layers and pure linear for output layer as activation functions. The concern of employing pure linear function as activation function is that we want one output neuron for slot(s) state prediction having value in the range from {−1 to 1} and the other one for idle time slot(s) prediction in the range from {1 to 5}. As far as the error minimization is concerned, it is easier to minimize the mean square error E as in (6) rather than the actual combined error, i.e., e. According to the backpropagation algorithm in [20], the connection weights are updated according to (7) and (8):

$$E = \frac{1}{2}\sum e^2 \tag{6}$$

$$v_{\beta\alpha_t}^k = v_{\beta\alpha_{t-1}}^k + \Delta v_t \tag{7}$$

$$\Delta v_t = -\eta \frac{\partial E}{\partial v} + \gamma \Delta v_{t-1}, \tag{8}$$

where $v = v_{\beta\alpha_{t-1}}^k$ and this corresponds to the connection weight of neuron β of the kth layer, and the subscript (t – 1) classifies the previous instant weights. The partial derivative $\frac{\partial E}{\partial v}$ in (8) is computed successively for each

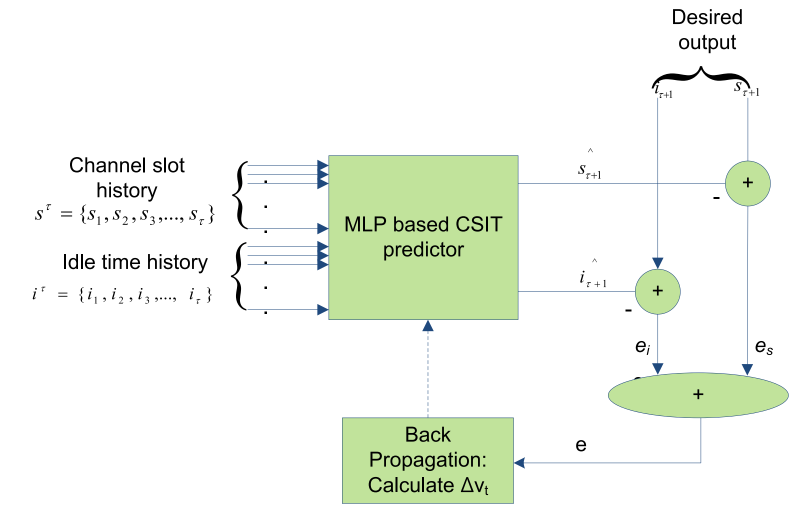


Figure 6 MLP-based CSIT predictor training.

neuron from the output to the input layer and is expressed in chain rule in terms of e, y_{β}^{k}, z_{β}^{k}, and $v_{\beta\alpha}^{k}$ as follows:

$$\frac{\partial E}{\partial v} = \frac{\partial E}{\partial v_{\beta\alpha}^{k}} = \frac{\partial E}{e}\frac{\partial e}{\partial y_{\beta}^{k}}\frac{\partial y_{\beta}^{k}}{\partial z_{\beta}^{k}}\frac{\partial z_{\beta}^{k}}{\partial v_{\beta\alpha}^{k}} \tag{9}$$

The above expression's chain rule components are represented by (2), (3), and (6) as follows:

$$\frac{\partial E}{\partial e} = e; \frac{\partial e}{\partial y_{\beta}^{k}} = -1; \frac{\partial y_{\beta}^{k}}{\partial z_{\beta}^{k}} = \varphi'\left(z_{\beta}^{k}\right); \quad \frac{\partial z_{\beta}^{k}}{\partial v_{\beta\alpha}^{k}} = y_{\alpha}^{k-1}.$$

Therefore, based upon the above partial expressions values, (9) can be represented as

$$\frac{\partial E}{\partial v} = \frac{\partial E}{\partial v_{\beta\alpha}^{k}} = -e\varphi'\left(z_{\beta}^{k}\right)y_{\alpha}^{k-1}. \tag{10}$$

To compute the rate of change of mean square error with respect to the desired output from neuron, replace k with o in the above expression, where o is the delimiter that corresponds to the output layer. Then (10) can be written as

$$\frac{\partial E}{\partial v} = \frac{\partial E}{\partial v_{\beta\alpha}^{o}} = -e\varphi'\left(z_{\beta}^{o}\right)y_{\alpha}^{o-1}$$

However, if we want to compute the partial derivative for neurons in hidden layers whose desired outputs are not known, then the partial derivative $\frac{\partial E}{\partial v}$ is calculated in terms of local gradient and is denoted by δ_{β}^{k}. The local gradient $\frac{\partial E}{\partial z_{\beta}^{k}}$ is also represented by the chain rule as follows:

$$\delta_{\beta}^{k} = \frac{\partial E}{\partial z_{\beta}^{k}} = \frac{\partial E}{\partial y_{\beta}^{k}}\frac{\partial y_{\beta}^{k}}{\partial z_{\beta}^{k}} = e\varphi'\left(z_{\beta}^{k}\right) \tag{11}$$

The above mathematical modeling is the training representation of our proposed predictor. We train the MLP with the slot(s) state and idle time slot(s) history. The training of the proposed predictor is done by changing the weights according to (7) with the aim of minimizing the mean square error, i.e., E in (6). We have repeated the above weight updating procedure until the threshold in terms of the required mean square error is achieved, where the threshold mean square error is the tolerable prediction error.

Once the CSIT predictor training has been completed, we apply the unseen data for checking the performance of the trained predictor. We predict the slot(s) state by using the binary symbol decision boundary as in (12a). The predicted idle time slot(s) is represented by taking into account the round function to the nearest integral value as in (12b) because, as already mentioned, the idle time slot(s) is represented in the integral range from {1 to 5}:

$$\begin{aligned} &\text{if} \quad \hat{s}_{\tau+1} \geq 0 \Rightarrow \hat{s}_{\tau+1} = 1 \\ &\text{if} \quad \hat{s}_{\tau+1} < 0 \Rightarrow \hat{s}_{\tau+1} = -1 \end{aligned} \tag{12a}$$

$$\hat{i}_{\tau+1} = \text{round}\left(\hat{i}_{\tau+1}\right). \tag{12b}$$

The complexity of the proposed MLP-based CSIT predictor actually lies in the training period. Remember the training period is classified as the interval in which the NN is trained for every possible data set. Once the training period has been completed, the computational complexity is significantly reduced. As mentioned in Section 6.1, once the proposed MLP-based CSIT predictor is trained with 8,000 slot(s), there is just a prediction error of about 0.09 by validating the data set of 30,000 slot(s). However, if the NN is trained with lesser slot(s), then there will be a bit larger prediction error. As far as the network operation is considered, the primary network would not be effected within the training period because the interfering cCU$_{\text{predict}}$ is alerted by the eNB to change the transmission slot(s), as mentioned in Section 3.1. Finally, the delay involved in the training of the MLP-based CSIT predictor increases the information exchange among eNB and cCU$_{\text{predict}}$ that guarantees the interference minimization to the primary network. However, this information exchange is reduced as the training of the predictor is done progressively. This is due to the fact that training reduces the prediction error and so does the interference to the primary network.

6 Simulation and analysis

In this section, we first elaborate the accuracy of our proposed MLP-based CSIT predictor, then the performance improvement by employing it in cognitive LTE-A network is illustrated. For the sake of CSIT prediction, the PU traffic is modeled here in statistical domain. We use two types of stochastic processes, i.e., Poisson and Pareto random process. The former is used for the arrival of PU and the vacancy of the channels/slot(s) in terms of time slot(s) is modeled by the latter one. We train our predictor using the statistical model of PU because we are doing analysis on the offline data. Therefore, we have used MLP for predicting future opportunities. However, MLP is applicable to any type of stochastic traffic model. The parameter, PU activity, as used later in this section, is the percentile activity of the PU on primary channels. This corresponds to the increased arrivals of PU and hence increased busy time slot(s) and is changed by adjusting the parameters of the Pareto distribution. All the simulations are carried out in the MATLAB R2010 using the Neural Network Tool Box (Mathworks, Natick, MA, USA).

6.1 Accuracy of MLP-based CSIT predictor

The accuracy of our proposed predictor is presented here by two means: slot(s) state and idle time slot(s) prediction. The *training* and *validation* of the proposed MLP are carried out for both slot(s) state and idle time slot(s). Generally speaking, the training refers to the phase of NN in which the nonlinear mapping between the inputs and the desired outputs is done by adjusting the link weights. The weights are adaptively adjusted by minimizing the mean square error between the desired and the actual MLP output. The validation refers to confirm the accuracy of the trained NN by giving the unseen data, where unseen data are the data for which the NN is not trained.

6.1.1 Training and validation of slot(s) states

The training of the MLP-based CSIT predictor in terms of slot(s) state is depicted in Figure 7. We train the predictor with a total of 8,000 slot(s) or 1,000 data points, where one data point is equivalent to the eight slot(s) that contain the slot(s) state information, either 1 or −1. Each data point actually represents a single RB which means that we are indirectly training the MLP for history of 1,000 RBs. However, due to limited space we just show 100 data points for better visual illustration.

As shown in Figure 7, there is just an error of about 0.07 for the training of 100 data points or 800 slot(s). In addition, the error is about 0.04 for the training of 1,000 data points or 8,000 slot(s). This malfunction in terms of mismatch between the proposed predictor and the targeted output is inherently induced by training the predictor not with the original input data but rather with some erroneous data, where the erroneous data are generated by minimally randomizing the original input data and the original input data correspond to the PU traffic that is modeled in the statistical domain. The concern of employing erroneous data for training is twofold. Firstly, in real world problems, it is impossible to have exact training data sets [25]; therefore, erroneous data are exploited here for close approximation to real situation. Secondly, binding of erroneous input and desired output contributes to better convergence in terms of reduced prediction error when using the NN for unseen data. The training with original input data leads to the prediction error of 0.10, whereas with the erroneous data, this error is reduced to 0.09 for the validation of 30,000 slot(s).

Figure 8 illustrates the validating data results of our proposed predictor. We have tested our predictor for the unseen data of 30,000 slot(s) and it is evaluated that there is just an error of about 0.09. However, the prediction error in the validation of 100 data points or 800 slot(s) is about 0.08 as shown in Figure 8. This reduced error is due to the fact that we have trained the NN with some erroneous data.

6.1.2 Training and validation of idle time slot(s)

The training of the MLP-based CSIT predictor for idle time slot(s) prediction is illustrated in Figure 9. The training and validation in terms of idle time slot(s) are carried out here in the same way for the slot(s) states. Although we used the same structure for the slot(s) and

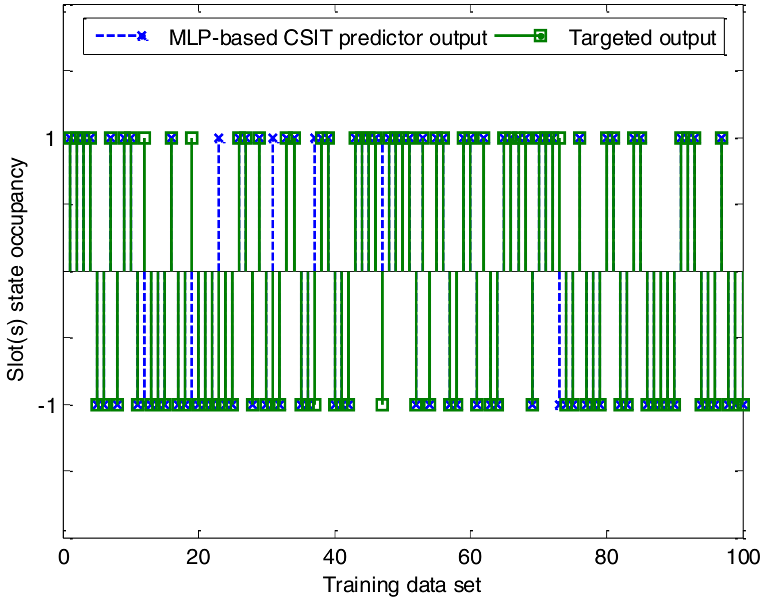


Figure 7 Slot(s) state training of the MLP-based CSIT predictor.

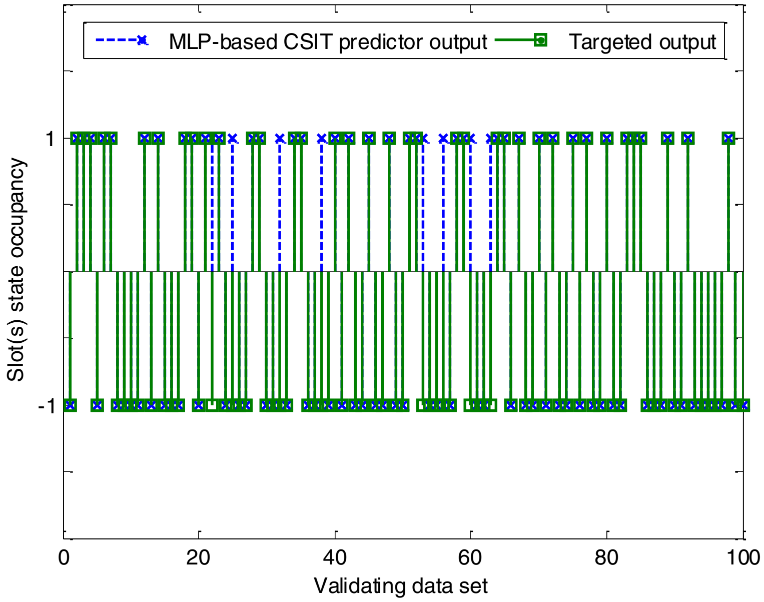


Figure 8 **Slot(s) state validation of the MLP-based CSIT predictor.**

their corresponding data points as in the previous subsection, here we used the integral idle time slot(s) history for training and validation. The training is also carried out here with erroneous data, where the same reason holds as for the slot(s) state training. There is a negligible error between the targeted output and the predictor output, i.e., about 0.07, among the training of 100 data points or 800 slot(s) as shown in Figure 9. Moreover, the error is about 0.047 for the training of 1,000 data points or 8,000 slot(s). The validation or the interaction with the unseen data of our proposed predictor is shown in Figure 10. It can be illustrated from the Figure 10 that there is an error of about 0.014, for validation of 100 data points or 800 slot(s). In addition, the error is about 0.09 for the validation of 30,000/8 data points or 30,000 slot(s).

It can be concluded from the above explanation that by training the predictor with 8,000 slot(s) or 1,000 data points, there is a prediction error of 0.09 for both slot(s) state and idle time slot(s) between the desired and the

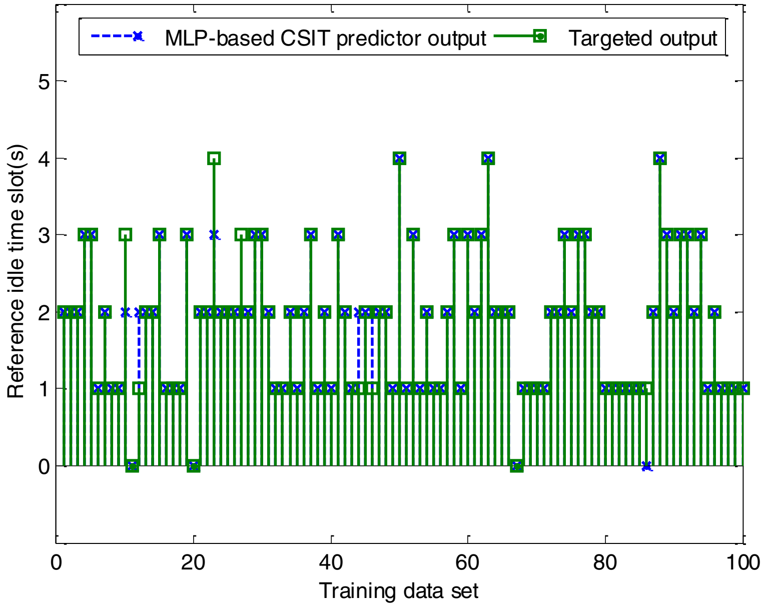


Figure 9 **Idle time slot(s) training of the MLP-based CSIT predictor.**

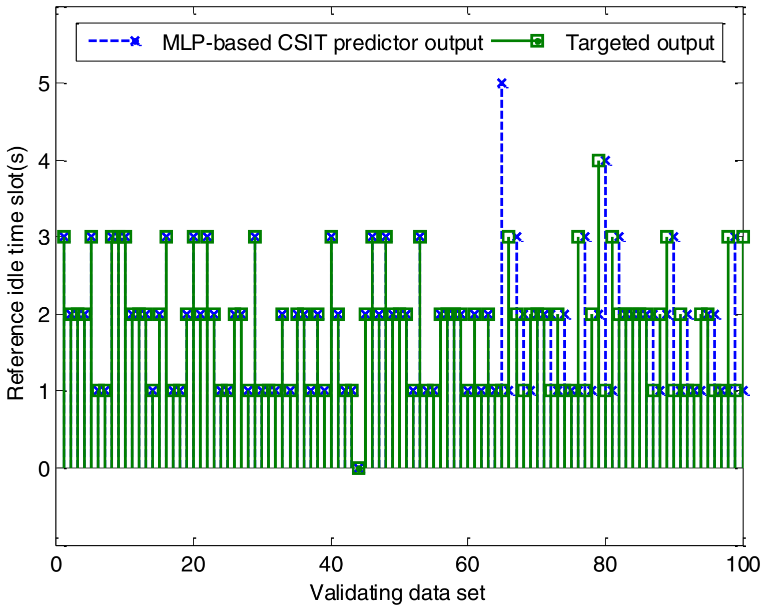


Figure 10 Idle time slot(s) validation of the MLP-based CSIT predictor.

predicted output. Furthermore, it can also be deduced that once the NN is trained for prediction with appropriate data sets, the error between the desired and actual predictor output is significantly reduced. We are not referring to the real prediction error here that resulted from using NN. However, the proposed CSIT MLP approximates the prediction error in a real environment. This is due to the fact; we are training the NN with the data set that closely approximates the behavior of PU activity in a real mobile communication environment. The data set in the form of PU traffic is explicitly described in the results in Section 6.

6.2 Performance improvement by predictive cognitive LTE-A

The advantage of exploiting CSIT prediction for cognitive LTE-A is illustrated here in terms of five performance measures, i.e., spectrum utilization, sensing energy, channel switching, PLR, and average instantaneous throughput. The comparison is carried out between predictive and nonpredictive CR, $\mathrm{CU_{predict}}$ and $\mathrm{CU_{sense}}$. Although the exploitation of $\mathrm{CU_{sense}}$ in LTE-A contributes to the significant performance improvement, e.g., spectrum utilization and throughput, the performance can further be enhanced by inculcating $\mathrm{CU_{predict}}$ along with $\mathrm{CU_{sense}}$ in LTE-A network. The job of $\mathrm{CU_{sense}}$ is just to scan the whole available band and discover the opportunities in terms of SHs. However, $\mathrm{CU_{predict}}$ works slightly different in the sense that it predicts the CSIT before actually applying sensing on the slot(s). The benefits of employing predictive CR concepts in LTE-A are illustrated following this paragraph.

6.2.1 Spectrum utilization

The performance improvement by employing CSIT prediction in terms of spectrum utilization (SU), as illustrated in Table 2, is defined as the ratio of the number of idle slot (s) sensed to the total number of idle slot(s) in the system for a limited duration in terms of time slot(s),

$$\mathrm{SU} = \frac{\text{Number of idle time slot(s) sensed}}{\text{Total number of idle slot(s) in the system}} \tag{13}$$

The number of idle slot(s) sensed by both $\mathrm{CU_{sense}}$ and $\mathrm{CU_{predict}}$ CR in LTE-A are N_i and N_p. The total number

Table 2 Spectrum utilization improvement

Number of primary slot(s)	N_i (50%)	N_i (60%)	N_ip (50%)	N_ip (60%)	$\mathrm{SU_{imp\text{-}50\%}}$	$\mathrm{SU_{imp\text{-}60\%}}$
1,000	245	198	380	287	55.10	44.94
2,000	259	201	424	310	63.70	54.22
3,000	255	215	450	360	76.47	67.44
4,000	267	214	498	376	86.52	75.70
5,000	270	216	612	467	126.67	116.20

of idle slots in the system is N_{is}. Therefore, the SU for each can be represented as

$$\text{SU}_{\text{sense}} = \frac{N_{\text{i}}}{N_{\text{is}}} \quad \text{and } \text{SU}_{\text{predictive}} = \frac{N_{\text{ip}}}{N_{is}}. \quad (14)$$

SU improvement (SU_{imp}) can be represented based upon the above two expression as

$$\text{SU}_{\text{imp}}(\%) = \frac{\text{SU}_{\text{predict}}-\text{SU}_{\text{sense}}}{\text{SU}_{\text{sense}}} = \frac{N_{\text{ip}}-N_{\text{i}}}{N_{\text{i}}}(\%). \qquad (15)$$

The numerical results of SU_{imp} are presented in Table 2. The analysis here is carried out by different primary slot(s) in the system, i.e., N_T. Also, two different PU user activity levels are incorporated here, i.e., 50% and 60%. It can be seen that by increasing N_T, SU_{imp} is going to increase regardless of the PU activity level. The is due to the fact that the increased number of PU slot (s) results in more idle slot(s) discovered by $\text{cCU}_{\text{predict}}$. In contrary, the number of slot(s) sensed by CU_{sense}, N_{i}, did not changed that much with the increase of N_T. This is due to the fact that CU_{sense} cannot discover idle slot (s) beforehand. The increased PU activity level also has a significant impact on SU_{imp}, i.e., it lessens the SU_{imp}, and this can be illustrated by the difference in the last two columns of Table 2.

6.2.2 Sensing energy

The sensing energy by employing $\text{CU}_{\text{predict}}$ within the LTE-A network is illustrated in two ways. First, the sensing energy consumption by CU_{sense} and $\text{CU}_{\text{predict}}$ with different PU activity is illustrated numerically. Second, the sensing energy reduction analysis is carried out mathematically.

The energy consumption of the CU_{sense} and $\text{CU}_{\text{predict}}$ with varying PU activity is illustrated in Figure 11. We assumed that the energy required for sensing a slot(s) is 100 (joules). It is illustrated that energy consumption of CU_{sense} rises by increasing the available primary slot(s), irrespective of the PU activity. This is due to the fact that CU_{sense} has no a prior information about the slot(s) statistics. However, $\text{CU}_{\text{predict}}$ contributes to the significant reduction in energy consumption because it applies sensing on those slot(s) that are predicted to be idle. This trend is also illustrated in Figure 11. Furthermore, it is notable that with an increase in PU activity, the energy consumption of $\text{CU}_{\text{predict}}$ is also reduced. This is due to the fact that increased PU activity corresponds to less transmission opportunities in terms of idle slot(s). Therefore, $\text{CU}_{\text{predict}}$ will not have enough slot(s) on which to apply sensing, and this results in further sensing energy reduction.

Here is a simple mathematical interpretation of sensing energy reduction. Let us assume that the total sensing energy required for classifying the occupancy of one slot(s) is X J. Then the total energy required for CU_{sense}

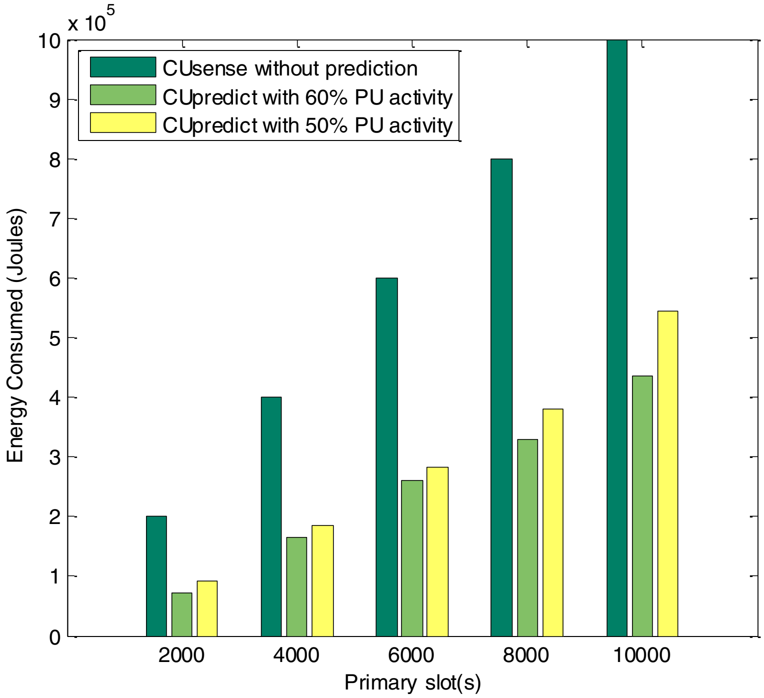


Figure 11 Sensing energy comparison between CU_{sense} and $\text{CU}_{\text{predict}}$.

for sensing Z slot(s) is given in (16), where Z is the total number of the slot(s) sensed by $\mathrm{CU_{sense}}$ in a finite duration of time slot(s):

$$E_{\mathrm{sense}} = Z^{*}X \tag{16}$$

and the total energy required for $\mathrm{CU_{predict}}$ for applying sensing on the Y slot(s) is

$$E_{\mathrm{predict}} = Y^{*}X, \tag{17}$$

where Y is the number of slot(s) predicted to be idle among the total N_T slots. It can be represented from (16) and (17) that the sensing energy of $\mathrm{CU_{sense}}$ is far greater than that of $\mathrm{CU_{predict}}$ and this can also be illustrated from the Figure 11. The reason being that the former applies sensing brutally whereas the latter one exploits slot(s) statistics by employing prediction. Furthermore, the percentile reduction in the sensing energy from (16) and (17) is represented in (18):

$$\mathrm{SE_{red}} = \frac{E_{\mathrm{sense}} - E_{\mathrm{predict}}}{E_{\mathrm{sense}}} = \frac{Z-Y}{Z} \tag{18}$$

Hence, by exploiting predictive modeling in $\mathrm{CU_{predict}}$, the sensing energy can be significantly reduced.

6.2.3 Channel switching rate

The channel switching rate is defined as the number of slot(s) switched per unit transmission time by the $\mathrm{S_eU}$ due to the result of PU arrival on the same slot(s), where $\mathrm{S_eU}$ can be either $\mathrm{CU_{sense}}$ or $\mathrm{CU_{predict}}$. $\mathrm{S_eU}$ in LTE-A network has to switch to another idle slot(s) whenever PU wants to access the slot(s) temporarily reserved by SU. The channel switching rate is an important performance bottleneck for the actual exploitation of DSA in the LTE-A network. The proposed MLP-based CSIT predictor in cognitive LTE-A network results in significant reduction in the channel switching rate. The benefit of predicting the idle time slot(s) is actually more elaborative here, in the sense that predictive $\mathrm{S_eU}$ operates on those groups of RBs that are predicted to be more idle in terms of the number of time slot(s).

The comparison of $\mathrm{CU_{sense}}$ and $\mathrm{CU_{predict}}$ in terms of slot(s) selection is carried out in LTE-A network in terms of number of primary slot(s) with varying PU traffic activity. The slot(s) selection is carried out randomly in $\mathrm{CU_{sense}}$ whereas predictive modeling is employed for the other one. Figure 12 depicts that idle time slot(s) prediction is more beneficial when we are talking in terms of the $\mathrm{S_eU}$ channel switching rate. It is seen that $\mathrm{CU_{predict}}$ with idle time slot(s) prediction results in a reduced channel switching rate as compared to the $\mathrm{CU_{sense}}$ with random slot(s) selection. Moreover, the channel switching rate is further reduced by increasing the number of available primary slot(s), specifically where $\mathrm{CU_{predict}}$ is concerned. Because of the presence of more primary slot(s), there is more probability of having longer predicted idle time slot(s), which results in a reduced channel switching rate. The effect of PU activity is also depicted such that the channel switching rate is minimized by reduction in the PU activity from 60% to 50%. Moreover, the channel switching rate of $\mathrm{CU_{sense}}$, by employing random slot(s) selection, always remains at high level as compared to the others. This is due to the lack of predictive modeling in $\mathrm{CU_{sense}}$ that leads to improper slot(s) selection as compared to its counterpart. Hence, the channel switching rate of $\mathrm{CU_{sense}}$ remains constant irrespective of the primary slot(s).

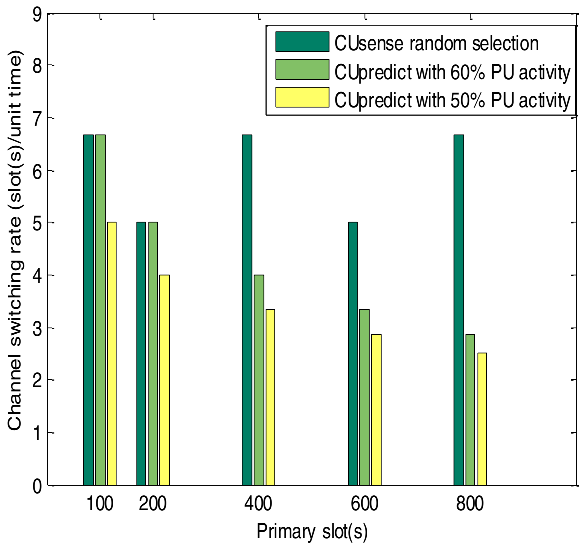


Figure 12 Channel switching rate comparison between $\mathrm{CU_{sense}}$ and $\mathrm{CU_{predict}}$.

6.2.4 Packet loss ratio

The PLR is defined as the event in which $\mathrm{S_eU}$ is unable to find the idle time slot(s) in the primary pool of slot(s). We illustrate the impact of employing CSIT prediction by $\mathrm{CU_{predict}}$ in LTE-A network in terms of reduced PLR in both theoretical and simulated means. The comparison of this predictive with nonpredictive modeling in terms of PLR is illustrated in Figure 13.

The theoretical modeling of average PLR in terms of both predictive and nonpredictive modeling is listed here. The expression of average PLR for $\mathrm{CU_{sense}}$, i.e., without prediction is listed below:

$$\mathrm{PLR}_{\mathrm{CU_{sense}}} = \prod_{m=1}^{C} (1 - p_m)\,; \qquad 1 \le C \le Z \tag{19}$$

where C is the threshold in terms of sensing slot(s) for the event PLR to occur, p_m is the probability of the mth

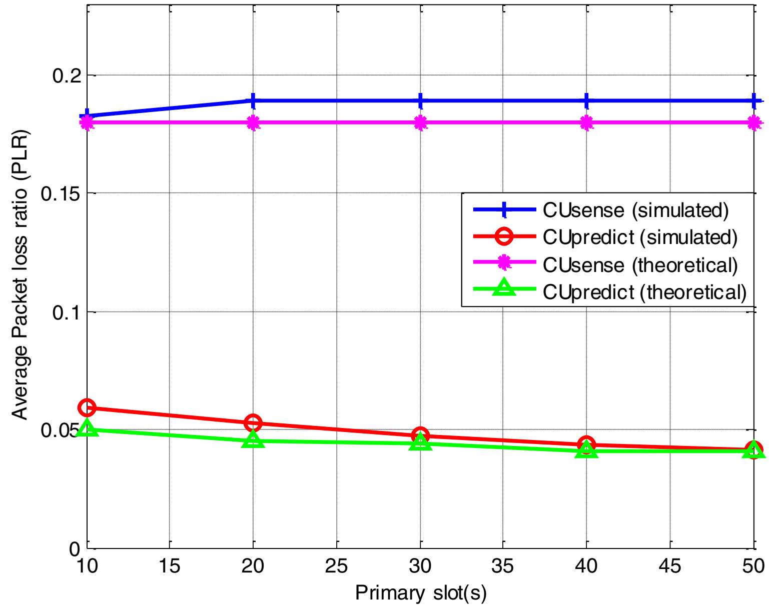


Figure 13 PLR comparison between $\mathrm{CU}_{\mathrm{sense}}$ and $\mathrm{CU}_{\mathrm{predict}}$.

slot(s) appearing to be idle and Z is the total slot(s) sensed by the $\mathrm{CU}_{\mathrm{sense}}$ in LTE-A network.

The inculcation of CSIT prediction by employing $\mathrm{CU}_{\mathrm{predict}}$ in the LTE-A in (19) can be rewritten as

$$\mathrm{PLR}_{\mathrm{CU}_{\mathrm{predict}}} = \prod_{m=1}^{C}(1-p_m + P_e p_m); \qquad 1\leq C\leq Y \tag{20}$$

For employing predictive modeling via $\mathrm{CU}_{\mathrm{predict}}$, we define that P_e, as $P_e > 0$, is the average probability that $\mathrm{CU}_{\mathrm{predict}}$ will predict the state of primary slot(s) incorrectly. The concern of its implication is to have theoretical representation for $\mathrm{PLR}_{\mathrm{CUpredict}}$. Hence, the introduction of the term P_e contributes to reduced PLR in terms of theoretical means as shown in Figure 13.

The performance gap of average PLR in Figure 13 elaborates the essence of employing CSIT prediction. The PLR for $\mathrm{CU}_{\mathrm{sense}}$ does not improve by increasing the number of primary slot(s). Although increased slots result in more transmission opportunities in terms of idle slot(s), the sensing order for $\mathrm{CU}_{\mathrm{sense}}$ cannot be improved. That is why the average PLR for the nonpredictive remains the same regardless of the number of available slot(s). However, $\mathrm{CU}_{\mathrm{predict}}$ has optimized sensing order by incorporating CSIT prediction. Therefore, $\mathrm{CU}_{\mathrm{predict}}$ results in high probability of achieving idle time slot(s) and hence reduced average PLR. This effect is depicted by both the simulated and theoretical curves of $\mathrm{CU}_{\mathrm{predict}}$ that average PLR is reduced for both cases. The increased number of primary slot(s) not only increases the transmission opportunities but also results in reduced PLR by improving the sensing order. However, there is a little difference between the simulated and theoretical curves for each $\mathrm{CU}_{\mathrm{sense}}$ and $\mathrm{CU}_{\mathrm{predict}}$, which is due to the prediction error in the trained NN that contributes to the increase simulated PLR values for each case.

6.2.5 Average instantaneous throughput

The analysis of employing $\mathrm{CU}_{\mathrm{predict}}$ in LTE-A network is carried out here in terms of average instantaneous

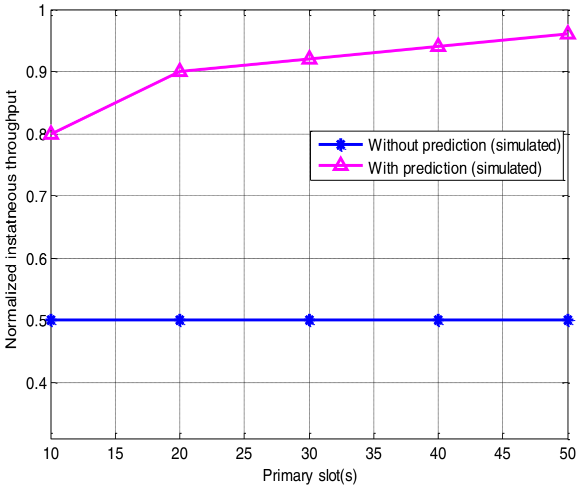


Figure 14 Average instantaneous throughput comparison between $\mathrm{CU}_{\mathrm{sense}}$ and $\mathrm{CU}_{\mathrm{predict}}$.

throughput. The comparison is carried out between $\mathrm{CU_{sense}}$ and $\mathrm{CU_{predict}}$ with respect to average instantaneous throughput by varying primary slot(s).

For the sake of simulation, we formulated the theoretical model of the instantaneous throughput per slot(s). We built the theoretical model for nonpredictive case to elaborate the simulation curves in Figure 14. The theoretical average instantaneous throughput model is formulated by assuming C_0 as the reference transmission capacity of $\mathrm{S_eU}$ and Δ_s as the time slot duration. We define a new independent parameter η_i^K which corresponds to how fast $\mathrm{S_eU}$ can find the idle time slots where $1 \le \eta_i^K \le N_s$. The throughput $R_{(i)}$ in time slot i for nonpredictive case is given by

$$R_{(i)} = C_0\Delta_s – C_0\eta_i^K. \tag{21}$$

We simulated the average instantaneous throughput for the $\mathrm{CU_{sense}}$ and $\mathrm{CU_{predict}}$ with respect to the number of primary slot(s) as illustrated in Figure 14, where the significant performance gap is achieved by employing CSIT prediction modeling. Moreover, the instantaneous throughput for the $\mathrm{CU_{sense}}$ case is same. This is due to the fact that $\mathrm{S_eU}$ $\mathrm{CU_{sense}}$ has no information about the statistics of the primary slot(s). However, $\mathrm{CU_{predict}}$ results in a bit increased throughput by increasing the primary slot(s) because more slot(s) result(s) in more available opportunities.

7 Conclusions

In this study we investigated the DSA in LTE-A network and proposed improvements by employing predictive modeling. Although LTE-A is the foremost mobile communication technology aiming to meet the requirements in terms of IMT-Advanced, inculcating CR concepts in it still needs to be addressed. Therefore, we have investigated the improvement by employing predictive CR concepts. The predictive modeling is inculcated by employing CSIT prediction using MLP NN. We train NN with the statistical model for PU traffic in mobile communication environment. Once the NN has been trained, it can predict the CSIT based on the sensed history on the control channel(s). Therefore, the projected $\mathrm{CU_{predict}}$ works by predicting the CSIT values. The improvement by employing CSIT using NN is illustrated in terms of five performance measures, i.e., spectrum utilization, sensing energy, channel switching rate, PLR, and average instantaneous throughput. Our proposed CSIT predictor results in significant performance gains in all the above-mentioned performance measures. As far as the complexity of the proposed predictor is concerned, once the NN is trained with appropriate data set, then the computational complexity is significantly minimized. This study can be extended horizontally by incorporating other prediction parameters, like throughput. The work presented here can also be extended vertically by analyzing other prediction models and compare the complexity of the suggested ones.

Competing interests

The authors declare that they have no competing interests.

Acknowledgement

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the C-ITRC (Convergence Information Technology Research Center) support program (NIPA-2013-H0401-13-0003) supervised by the NIPA (National IT Industry Promotion Agency) and the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (2012009449).

Received: 1 February 2013 Accepted: 27 July 2013
Published: 3 August 2013

References

1. G Yuan, X Zhang, W Wang, Y Yang, Carrier aggregation for LTE-advanced mobile communication systems. IEEE Commun. Mag. **48**(2), 88–93 (2010)
2. KI Pedersen, F Frederiksen, C Rosa, H Nguyen, LGU Garcia, Y Wang, Carrier aggregation for LTE-advanced: functionality and performance aspects. IEEE Commun. Mag. **49**(6), 89–95 (2011)
3. IF Akyildiz, WY Lee, MC Vuran, S Mohanty, Next generation/dynamic spectrum access/cognitive radio wireless networks: a survey. Comput. Netw. **50**(13), 2127–2159 (2006)
4. J Mitola, *Cognitive radio: an integrated agent architecture for software defined radio* (Dissertation, KTH Royal Institute of Technology, 2000)
5. S Haykin, Cognitive radio: brain-empowered wireless communications. IEEE J. Sel. Areas Commun. **23**(2), 201–220 (2005)
6. C Clancy, J Hecker, E Stuntebeck, T O'Shea, Applications of machine learning to cognitive radio networks. Wireless Communications, IEEE **14**(4), 47–52 (2007)
7. M Bkassiny, Y Li, S Jayaweera, A survey on machine learning techniques in cognitive radio networks. IEEE Comm. Survey and Tutorials **99**, 1–24 (2012)
8. A He, KK Bae, TR Newman, J Geaddert, K Kim, R Menon, LM Tirado, JJ Neel, Y Zhao, JH Reed, WH Tranter, A survey of artificial intelligence for cognitive radios. IEEE Transactions on Vehicular Technology. **59**(4), 1578–1592 (2010)
9. Y Liang, Y Sixing, H Weijun, L Shufang, Spectrum behavior learning in cognitive radio based on artificial neural network, in *Military Communications Conference, 2011 - Milcom 2011*, (Baltimore), pp. 25–30. 710 –November 2011
10. MI Taj, M Akil, Cognitive radio spectrum evolution prediction using artificial neural network based multivariate time series modeling, in *11th European Wireless Conference 2011 - Sustainable Wireless Technologies (European Wireless)*, (Vienna), pp. 1–6. 2729 –April 2011
11. VK Tumuluru, W Ping, D Niyato, A neural network based spectrum prediction scheme for cognitive radio, in *2010 IEEE International Conference on Communications (IEEE ICC), 2010* (, Cape Town), pp. 1–5. 2327 –May 2010
12. H Shokri-Ghadikolaei, R Fallahi, Intelligent sensing matrix setting in cognitive radio network. IEEE Commu. Lett. **16**(11), 1824–1827 (2012)
13. K Tsagkaris, A Katidiotis, P Demestichas, Neural network-based learning schemes for cognitive radio systems. Comp. Commu. **31**(14), 3394–3404 (2008)
14. Y Guangxiang, RC Grammenos, Y Yang, W Wenbo, Performance analysis of selective opportunistic spectrum access with traffic prediction. IEEE Vehicular. Tech. **59**(4), 1949–1959 (2010)
15. JD Deaton, RE Irwin, LA DaSilva, The effect of a dynamic spectrum access overlay in LTE-Advanced networks, in *2011 IEEE Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN)*, (Aachen), pp. 488–497. 36 –May 2011
16. O Vucenta, H Carlos, JF Monserrat, G Xavier, Implementing opportunistic spectrum access in LTE-Advanced. EURASIP JWCN **2012**, 99 (2012)
17. L Giupponi, A Galindo-Serrano, P Blasco, M Dohler, Docitive networks: an emerging paradigm for dynamic spectrum management. Wireless Communications IEEE. **17**(4), 47–54 (2010)
18. A Galindo-Serrano, L Giupponi, P Blasco, M Dohler, Learning from experts in cognitive radio networks: the docitive paradigm, in *Proceedings of the Fifth*

International Conference on Cognitive Radio Oriented Wireless Networks & Communications (CROWNCOM), (Cannes, 2010), pp. 1–6. 911 –June 2010

19. 3GPP TS 36133, Evolved Universal Terrestrial Radio Access (E-UTRA); requirements for support of radio resource management, Accessed April 2010. http://www.etsi.org/deliver/etsi_ts/136100_136199/136133/09.03.00_60/ts_136133v090300p.pdf
20. S Haykin, *Neural Networks: A Comprehensive Foundation*, 2nd edn. (Prentice Hall, Upper Saddle River, NJ, USA, 1999), pp. 161–175
21. F Rosenblatt, The perceptron: a probabilistic model for information storage and organization in the brain. Cornell Aeronautical Laboratory. Psychological Review **65**(6), 386–408 (1958)
22. JL Elman, Finding structure in time. Cognitive Science **14**, 179–211 (1990)
23. PJ Werbos, Backpropagation through time: what it does and how to do it. IEEE Proc. **78**(10), 1550–1560 (1990)
24. E Ong, A Flitman, Using neural network to predict binary outcomes, in *1997 IEEE International Conference on Intelligent Processing Systems (ICIPS 1997)*, (Beijing), pp. 427–431. 2831 –October 2010
25. P Hartono, S Hashimoto, Learning from imperfect data. Applied Soft Computing Journal **7**(1), 353–363 (2007)

doi:10.1186/1687-1499-2013-203
Cite this article as: Shahid *et al.*: **CSIT: channel state and idle time predictor using a neural network for cognitive LTE-Advanced network.** *EURASIP Journal on Wireless Communications and Networking* 2013 **2013**:203.