Bayesian and Neural Network Schemes for Call Admission Control in LTE Systems

Biljana Bojović⋆,⋄, Giorgio Quer⋆, Nicola Baldo⋆, Ramesh R. Rao⋆
⋆Centre Tecnològic de Telecomunicacions de Catalunya (CTTC), Av. Gauss 7 – 08860 Castelldefels, Spain
⋄University of California, San Diego – La Jolla, CA 92093, USA

Abstract—Cognitive networking paradigms may help meet the challenges of operating complex wireless communications networks. In this paper, we contrast the neural network (NN) and the Bayesian network (BN) models to extract information from real-time observations and optimize network performance. In particular, we apply these two models to the problem of call admission control (CAC) for a long term evolution (LTE) system. We simulate a realistic LTE scenario with mobility in ns-3 and we select the most relevant features that can be observed by the base station. Then, we design two new CAC schemes that autonomously learn the network behavior from the observation of the selected features. Furthermore, we propose a performance comparison among these two schemes and a state-of-the-art CAC scheme, showing that the NN and the BN schemes are very promising solutions for CAC in LTE systems.

I. INTRODUCTION AND RELATED WORK

Wireless communications networks are becoming more and more complex, to the point that their operation and maintenance cannot be sustained without resorting to automation. In this respect, the use of machine learning techniques that can extract useful information from massive amounts of measurement data is a very promising line of investigation. The idea is to use these techniques to realize a cognitive network [1], i.e., a cognition process that spans the whole network, and learns from the observation of the environment in order to reconfigure the network parameters to optimize its performance.

In the recent literature, some specific machine learning techniques have been proposed for the realization of the cognitive network paradigm. In particular, the application of supervised learning via neural networks (NNs) was proposed to learn how different system configurations affect the communication performance in the presence of different environmental conditions [2]. This approach was shown to be effective in a number of cases, such as access point selection [3] in WLANs, radio admission control in long term evolution (LTE) systems [4], and optimization of cognitive radio systems [5].

In spite of their success, a potential drawback of NNs is that their output is a predicted value whose confidence is not known. In this respect, a machine learning technique that could provide not just a crisp value as output, but rather a probability distribution of the value of interest, seems more promising. A popular machine learning tool of this kind is the Bayesian network (BN). A BN is a graphical model that is used to represent the probabilistic relationships among a set of variables. In a machine learning context, this tool is used to carry out statistical inference in a computationally efficient way, e.g., to predict the probability distribution of a certain variable conditioned to the known value of some other variables. An example of the application of this approach in a cognitive network context was presented in [6], where the particular application to CAC in WLANs was considered.

To summarize, several publications have discussed the application of a particular machine learning technique to a cognitive network scenario. The goal of this paper is to do a comparative study of the application of NNs and BNs to the same scenario and with the same task, and provide an experimental comparison between the performance of the two techniques. For this purpose, we have selected a novel and relevant research topic in the field of mobile communications: the problem of call admission control (CAC) in LTE systems. This problem is very challenging due to the high complexity of the LTE technology, which can be tracked back to its advanced features such as adaptive modulation and coding, hybrid automatic repeat request (HARQ), and dynamic packet scheduling with quality of service (QoS) support. All these factors together make the performance of an LTE system very hard to predict, due to the users’ mobility, and to the variations in propagation conditions, as well as in the type and amount of traffic.

Several CAC schemes have already been proposed for LTE systems. In [7] a scheme, which models the call arrival process with queueing theory, is proposed, and the concept of resource reservation is applied to this scheme. In the presence of an incoming call, extra resources are reserved to avoid QoS degradation. The amount of these extra resources is determined a priori based on the knowledge of the user mobility patterns. The major problem is that LTE is expected to be used with a mixture of heterogeneous cells of different sizes (macro/micro/pico/femto cells), deployed in a loosely coordinated fashion, with minimum to no planning. In such conditions, the statistics of the mobility pattern of the users are expected to vary significantly among different cells, and cannot be known in advance; furthermore, overcoming these variations by a conservative estimation of the extra resources to be reserved would lead to a poor resource utilization. Other CAC schemes for LTE follow a ring-based modeling approach [8], [9]. This modeling approach assumes that users belonging to a cell can be grouped into rings according to their distance from the base station, and that the users located in the same ring consume the same amount of radio resources. Even though these schemes show significant performance improvements compared to the previous CAC models for LTE, they still rely on a priori modeling of the radio environment, which is not realistic. In particular, in the presence of frequency selective fading, as well as path loss variations due to obstacles such as walls and buildings, the prediction of the needed amount
of resources based solely on the distance between users and the base station may be highly inaccurate.

In other words, the vast majority of previously proposed CAC algorithms for LTE are based on some a priori known analytical models to predict the variations in the resource utilization. Thus, they may fail as soon as these models do not match the actual deployment conditions, which is in fact expected to occur in the vast majority of LTE deployment scenarios. For this reason, an LTE CAC scheme relying on a learning based approach, which can react to the actual conditions faced by each cell, looks more promising. In a previous work [4] we considered such a scheme based on NNs. In that preliminary work we considered a scenario without mobility and with simplified propagation conditions, thus not realistic. In this paper, we go significantly beyond our previous work by considering a much more realistic scenario with user mobility and frequency selective fading, by improving the performance of the NN-based LTE CAC scheme, and by additionally designing and evaluating a new LTE CAC scheme based on BNs.

The main contributions of this paper are:

- the design and implementation of a realistic simulation scenario representative of an LTE system with heterogeneous types of traffic, realistic propagation conditions, and user mobility;
- the definition of a new feature extraction model for measurements available in an LTE system used for a learning-based CAC scheme;
- the design of two learning-based LTE CAC schemes based on a NN and on a BN, respectively, including the parameter learning phase for both models; and
- a comparative analysis of the performance, in terms of CAC decision accuracy, of the NN, the BN and a state-of-the-art CAC scheme for LTE.

The rest of the paper is organized as follows. In Sec. II we detail the system model and the feature extraction model for the CAC schemes, then in Sec. III and in Sec. IV we describe the NN and the BN models, respectively. In Sec. V we present the CAC decision schemes for both the NN and the BN models, and in Sec. VI we describe the experimental setup and the main results. Sec. VII concludes the paper.

II. SYSTEM MODEL

A. Long Term Evolution (LTE) system overview

We consider an LTE / evolved packet system (EPS) with 1 base station (eNB) and N users (UEs). According to the LTE/EPS specifications, an EPS bearer defines the QoS requirements of a particular class of traffic for a UE, and all the packets transmitted to and from the UE should meet the QoS requirements of the EPS bearer to which they are mapped. EPS bearers are classified into two main categories: guaranteed bitrate (GBR) and non-guaranteed bitrate (non-GBR). For each GBR bearer, a certain value of bitrate is specified, which should be guaranteed by the LTE/EPS system. On the other hand, non-GBR bearers are served on a best-effort basis. Requirements other than the bitrate are normally defined by standardized QoS class identifier (QCI) values, which correspond to a pre-defined set of requirements.

For the LTE radio interface, we consider a frequency division duplex with a system bandwidth of S resource blocks (RBs). At each transmission time interval (TTI), the MAC scheduler located at the eNB decides which RBs are dedicated to the transmission of data belonging to each bearer, with the aim of satisfying the QoS requirements of the bearer. Clearly, meeting the QoS requirements for all bearers is only possible if the amount of radio resources available is sufficient in the current conditions. These conditions include the amount of data to be transmitted, and the number of bytes which can be transmitted within a RB to/from each UE, as determined by the adaptive modulation and coding (AMC) functionality. The AMC functionality depends on the propagation conditions, as well as on the distance between the UE and the eNB. The role of the CAC functionality is to determine if a new EPS bearer can be activated or not, i.e., if the available resources are sufficient to satisfy the QoS requirements of all active bearers plus the new one or, alternatively, if it is better to reject the activation of the new bearer in order to preserve the QoS of the previously activated ones [10].

B. Feature extraction for CAC

In our model, without loss of generality, we consider that there is one EPS bearer per UE. We have \( N_v \) GBR UEs and \( N_b \) non-GBR UEs. Each GBR UE is performing a VoIP call, while each non-GBR UE is transmitting a bulk file via a TCP connection. We measure the quality of service of the VoIP calls by using the R-factor defined by the E-model [11].

The R-factor after the acceptance (aa) of the incoming call, for every GBR UEs \( n = 1, \ldots, N_v \), is named \( R_{\text{aa}}^{(n)} \). This quantity depends on the future performance of the network, so it can not be measured at the moment of the CAC decision, but it can be estimated with our NN and BN models. According to the QoS constraints, all the GBR UEs should have at least a minimum R-factor, namely \( \tau_R \). The eNB should accept the incoming call if \( R_{\text{aa}}^{(n)} \geq \tau_R \) for all the GBR UEs, and drop the incoming call otherwise. Thus, we define the R-factor for the system as:

\[
R_{\text{aa}} = \min_n(R_{\text{aa}}^{(n)}).
\]  

If \( R_{\text{aa}} \geq \tau_R \), the QoS of all GBR UEs is satisfied even after the acceptance of the incoming call. On the contrary, if \( R_{\text{aa}} < \tau_R \), at least one UE does not receive a sufficient QoS. If it is possible to predict this event, the CAC scheme should drop the incoming call, thus preserving the QoS for the other GBR UEs.

In the following, we select a set of metrics that are representative of the network conditions, and that will be used as the inputs to the cognitive models to predict \( R_{\text{aa}} \) at the moment in which the CAC decision should be made.

One of the most important factors that affect the capacity in the LTE system is the radio resource allocation scheme. Radio resource allocation in an LTE system is not specified by the 3GPP standard and it is implemented by the operators. Thus, the ratio of resources that are assigned to GBR and non-GBR UEs depends on the implementation of the scheduler at the MAC layer. In our system, we model this radio resource allocation scheme as a black box. The performance of this scheme can be observed through the variables that describe...
the radio resource allocation over time, such as the ratio of resources that are assigned to GBR and non-GBR UEs, or the ratio of radio resources that are not assigned to any UE. Thus, we define the metric \( \phi_{v}^{(bd)} \) which represents the ratio of resource blocks that are dedicated to the GBR UEs during the last \( J \) time intervals before the CAC decision (bd):

\[
\phi_{v}^{(bd)} = \frac{1}{J} \sum_{j=1}^{J} K_{v}^{(j)}/K ,
\]

where \( K_{v}^{(j)} \) is the number of resource blocks assigned to GBR flows in the interval \( j \), while \( K \) is the total amount of available resource blocks. We also define the metric that describes the amount of resources that are allocated to non-GBR UEs in the following way:

\[
\phi_{b}^{(bd)} = \frac{1}{J} \sum_{j=1}^{J} K_{b}^{(j)}/K .
\]

Both \( \phi_{v}^{(bd)} \) and \( \phi_{b}^{(bd)} \) are providing information on the resource allocation, not on the actual MAC layer throughput. Furthermore, in case the resources for GBR UEs are not sufficient, a fraction of the \( \phi_{b}^{(bd)} \) can be allocated to GBR UEs.

Another important variable to describe the QoS of the system before the CAC decision is the R-factor calculated at the moment of the decision for every GBR UE \( n = 1, \ldots, N_u \), i.e., \( R_{n}^{(bd)} \). As a feature for the CAC scheme we consider:

\[
R_{n}^{(bd)} = \min_{n} \left( \frac{1}{J} \sum_{j=1}^{J} K_{v}^{(j)} / K \right) .
\]

Since the performance of non-GBR UEs may also be relevant to predict the value of \( R_{n}^{(aa)} \), we consider \( P_{n}^{(bd)} \) (with \( n = 1, \ldots, N_b \)), the successful packet rate for non-GBR UEs, and we define another feature for the CAC scheme as:

\[
P_{n}^{(bd)} = \min_{n} \left( \frac{1}{J} \sum_{j=1}^{J} K_{b}^{(j)} / K \right) .
\]

Another variable which has a high impact on overall system capacity is the channel quality indicator (CQI). The CQI is an indication of the data rate that can be supported by the channel for a certain UE. It depends on the signal to interference plus noise ratio (SINR) as well as on the characteristics of the UE receiver [12]. The CQI value is reported by each UE to the eNB and is used by the eNB to determine the modulation scheme and the code rate that can be supported by the channel for the downlink data transmissions. Since the network environment is changing quite fast in a realistic scenario, due to the mobility of the UEs, the CQI values may significantly vary over time. E.g., in the case of a wrong estimation of the code rate, packet losses and high delays for the packets received may occur. This would affect the perceived QoS. For these reasons, it is important to include some features which represent the variations in channel quality. Thus, we define a variable that represents the level of variations of the perceived channel quality by all UEs along all their subbands. For each UE \( n = 1, \ldots, N \), the eNB node collects the past \( U \) values of the CQI received from that node, \( CQI_{n}^{(u)} \) with \( u = 1, \ldots, U \), which are referred to the past \( J \) time intervals and to \( S \) different subbands, with \( J \cdot S = U \). With these values the standard deviation of the CQI for each UE \( n \) can be calculated as:

\[
\sigma_{CQI n}^{(bd)} = \left( \frac{1}{U} \sum_{u=1}^{U} \left( CQI_{n}^{(u)} - \frac{1}{U} \sum_{u=1}^{U} CQI_{n}^{(u)} \right)^2 \right)^{1/2} .
\]

By using the previous expression, we define the feature representing the CQI variation over all UEs as:

\[
\sigma_{CQI}^{(bd)} = \frac{1}{N} \sum_{n=1}^{N} \sigma_{CQI n}^{(bd)} .
\]

Another important feature related to the channel quality is the average CQI among all UEs, defined as:

\[
m_{CQI}^{(bd)} = \frac{1}{N} \sum_{n=1}^{N} \left( \frac{1}{U} \sum_{u=1}^{U} CQI_{n}^{(u)} \right) .
\]

### III. Neural Network Model

A NN is an adaptive system that changes its structure based on external or internal information that flows through the network. This adaptive system can be applied to different optimization problems in telecommunications, like pattern recognition, function approximation or classification. In this paper, a NN model is adopted to infer the future performance of the system as a function of the available network measurements. We use a Feed Forward NN (FFNN) with two layers of adaptive weights. We can write the NN function [13] as:

\[
y(x, w^{(1)}, w^{(2)}) = f \left( \sum_{h=1}^{H} w_{h}^{(2)} f \left( \sum_{m=0}^{M} w_{hm}^{(1)} x_{m} \right) + w_{0}^{(2)} \right) ,
\]

where \( x \) is the vector of the inputs \( x_{m} \), with \( m = 1, \ldots, M \), and \( x_{0} = 1; H \) is the number of nodes in the hidden layer; \( w^{(1)} \) is the vector of adaptive weights \( w_{hm}^{(1)} \) for the hidden layer, with \( h = 1, \ldots, H \); \( w^{(2)} \) is the vector of adaptive weights \( w_{h}^{(2)} \) for the output layer. The sigmoidal activation function is defined as:

\[
f(a) = \frac{1}{1 + e^{-a}} .
\]

In this paper, we select as inputs of the NN the set of metrics that are observable and available at the eNB, i.e.,

\[
x = \{ N_{v}, N_{b}, m_{CQI}^{(bd)}, \sigma_{CQI}^{(bd)}, \phi_{v}^{(bd)}, \phi_{b}^{(bd)}, P_{n}^{(bd)}, R_{n}^{(bd)} \} .
\]

The NN output is the value of \( R_{n}^{(aa)} \) obtained as a function of all the parameters in \( x \), which will be used to make the CAC decision, i.e., to accept or not the incoming call.

#### A. Neural Network training

The NN training is performed in a supervised manner. During the training phase, we collect a training dataset \( D \), i.e., for every tested scenario \( t = 1, \ldots, T \), all the metrics in the set \( x(t) \), as well as the value \( R_{n}^{(aa)}(t) \) of the R-factor after the new call has been accepted. The weights \( w^{(1)} \) and \( w^{(2)} \) of the NN are obtained by minimizing the error function, i.e.,

\[
E(w^{(1)}, w^{(2)}) = \frac{1}{2} \sum_{t=1}^{T} \| y(x(t), w^{(1)}, w^{(2)}) - R_{n}^{(aa)}(t) \|^2 .
\]
To optimize the NN training we use the improved resilient backpropagation algorithm without back-tracking (-iRPROP). This algorithm performs better than traditional RPROP, and it has a lower computational complexity than iRPROP+, while having similar accuracy performance [14]. During the training phase, we vary three training parameters to check how they affect the accuracy performance and to empirically select their optimal value: the learning rate, \( L \), the maximum number of epochs, \( E \), and the number of nodes in the hidden layer, \( H \).

In Fig. 1 we show the performance of the NN training phase as a function of the three learning parameters. The accuracy performance is represented by the normalized root mean square error, which is defined as:

\[
\varepsilon(E, H, L) = \sqrt{\frac{1}{T} \sum_{t=1}^{T} \frac{g(x(t), w^{(1)}, w^{(2)}) - R^{(aa)}(t)}{\max(R^{(aa)}) - \min(R^{(aa)})}^2},
\]

(13)

**IV. BAYESIAN NETWORK MODEL**

A BN is a probabilistic graphical model [15] that can describe in a compact way the conditional independence relationships among a set of random variables. In our case, such variables are represented by the metrics available at the moment in which we want to make the decision whether to accept or not the incoming call, i.e., the same variables selected as inputs of the NN, in the set \( x \), as well as the R-factor observed after the incoming call has been accepted, \( R^{(aa)} \). Indeed, during the learning phase of the BN model, we do not discriminate between observed variables and variables to infer, so we consider in the same way all the variables in the set \( v = \{ x, R^{(aa)} \} \).

The structure of the probabilistic relationships among these variables is represented by a directed acyclic graph (DAG). A DAG is a graphical representation of the conditional dependencies among the variables, that defines the structure of the joint probability among these variables. We use a structure learning algorithm [15] to select the DAG that best represents the probabilistic relationships among the variables, using the samples in the training dataset \( D \), described in Sec. III. A significant problem with this approach is that the size of the set of all possible DAGs grows super-exponentially with the number of variables (or nodes) in the DAG. For this reason, in order to find the DAG that best fits the observed data, we use a local search algorithm, the hill climbing (HC) random search [16], that performs a walk on the space of all the possible DAGs. This algorithm selects at each repetition of the algorithm a local minimum, i.e., a local best fitting DAG. With a sufficient number of repetitions it is possible, in our case, to reach a solution that is good enough for our inference purposes, i.e., that provides good performance results. In any case, it is not guaranteed that the optimal solution can be reached with a finite number of repetitions of the algorithm.

For every iteration of the structure learning algorithm, the best DAG is chosen among a set of DAGs to be 1) the best representation of the probabilistic relationships among the variables, and 2) as simple as possible, in order to allow the BN algorithm to learn the quantitative relationships among the variables from a limited training set. In order to meet these two goals, we score the DAGs with the Bayesian information criterion (BIC) scoring function, see e.g., [17]. This criterion assigns a score to each DAG as a function of how well the data in the training dataset \( D \) is represented by the DAG chosen (in terms of maximum likelihood), and it also penalizes based on the number of edges of the DAG, thus favoring simpler DAG structures.

In Fig. 2 we show the BN obtained with the HC random search and the BIC function to score each selected DAG. Observing this DAG, we notice that there exists a set of nodes \( z \) that separates the \( R^{(aa)} \) from the rest of the network according to the d-separation rule [15]. This means that if all the values of the nodes in \( z \) are observed, \( R^{(aa)} \) becomes independent from any other node in the network. This set of nodes is composed of:

\[
z = \{ N_v, N_b, m^{(bd)}_{CQI}, p^{(bd)}, R^{(bd)} \},
\]

(14)

and these are the nodes (features) that are observed in order to make a prediction with the BN.

Regarding the parameter learning phase, we adopt a ML model [13], coherently with the choice of the BIC as a scoring function, to infer quantitatively the probabilistic relationships
between the variables of the nodes in \( z \) and the variable we want to infer, \( R^{(aa)} \).

V. CALL ADMISSION CONTROL (CAC) DECISION

A CAC control scheme should provide a binary output every time a user (UE) is starting a new call, i.e., to accept or to drop the incoming call. The policy is that a call should always be accepted, with the exception of the case in which the acceptance of a new call may affect the quality of the calls from other UEs. In other words, the eNB should accept a new call if the \( R^{(aa)} \) will be above a certain threshold \( \tau_R \), and drop the call otherwise. The value of \( R^{(aa)} \) depends on the performance of the network in the future, so it can only be estimated.

The CAC scheme exploiting the NN approach works as follows. In the presence of an incoming call, the NN estimates the value of \( R^{(aa)} \) as a function of all the variables in \( x \), defined in Eq. (11). If the estimated value \( \hat{R}^{(aa)} \geq \tau_R \) the incoming call is accepted, otherwise if \( \hat{R}^{(aa)} < \tau_R \) the incoming call is dropped.

The CAC scheme exploiting the BN is slightly different. In the presence of an incoming call, the BN estimates the probability distribution of \( R^{(aa)} \) as a function of all the variables in \( z \), defined in Eq. (14). Then, if the probability

\[
p(R^{(aa)} \geq \tau_R) > p_T,
\]

the call is accepted, otherwise the call is dropped. The value of \( p_T \) is another parameter of the CAC scheme, that is discussed in Sec.VI-B.

For every incoming call, we should discriminate among four cases:

1) the CAC scheme suggests to accept the incoming call, and the value of the R-factor, calculated a posteriori, is \( R^{(aa)} \geq \tau_R \); this corresponds to a correct decision;
2) the CAC scheme suggests to accept the incoming call, but \( R^{(aa)} < \tau_R \); this corresponds to an incorrect decision;
3) the CAC scheme suggests to not accept the call; for testing purposes, the system accepts the incoming call anyway, and the R-factor calculated in the case the call is accepted is \( R^{(aa)} \geq \tau_R \); this corresponds to an incorrect decision;
4) the CAC scheme suggests to not accept the call; as in the previous case, the call is accepted for testing purposes and \( R^{(aa)} < \tau_R \); this corresponds to a correct decision.

In the following, we refer to case 2) as a False Negative (FN). In this case the CAC scheme erroneously decides to accept the call, but the R-factor after the call has been accepted is \( R^{(aa)} < \tau_R \). This means that the scheme cannot meet the QoS requirements.

On the other hand, we refer to case 3) as a False Positive (FP). In this case, the CAC scheme erroneously decides not to accept the call, but the R-factor if the call would be accepted is \( R^{(aa)} < \tau_R \). Thus, the radio resources are not fully exploited, since the CAC scheme drops a call that could instead be supported by the system.

VI. PERFORMANCE EVALUATION

A. Scenario and experiment setup

We model a macro cell LTE CAC scenario with the ns-3 simulator [18]. In this scenario, several users (UEs) are connected to a single base station (eNB). To simulate the UEs’ mobility, we adopt the steady state random waypoint mobility model [19], which is implemented in ns-3. With this model we can simulate random mobility of the UEs in a rectangular area, and the initial distribution of the UEs positions is the steady state distribution. For this reason, each simulation run reaches its steady state in a very short simulation time. The simulation area is square and the eNB is positioned in the center of the square area. As a radio propagation model we adopt the COST-231 path loss model [20], which is a common choice to simulate macro cell outdoor scenarios. The size of the square area is chosen such that, given the path loss model and the other network parameters, we have a wide range of SINR values. Furthermore, we use a Rayleigh multipath fading model, whose parameters are specified by 3GPP specification [21] for extended pedestrian A model (EPA). Regarding the configuration of the higher layers, we divide the UEs into two categories. The UEs in the first category are performing VoIP calls over a GBR bearer, while the other UEs are performing a TCP file download over a non-GBR bearer. The scenario setup described above is very demanding for the LTE MAC scheduler at the eNB, which has to satisfy the QoS requirements while the network conditions vary significantly as a function of time and frequency, due to the mobility and the fast fading. For this reason, we choose the priority set scheduler (PSS) which is able to successfully adapt to the changes in the channel conditions, while providing the guaranteed bit rate to the GBR UEs [22]. Other relevant experimental setup parameters are provided in Tab. I.

In the simulations, we vary the values of the number of GBR UEs, \( N_v = 12, 13, \ldots, 17 \); the number of non-GBR UEs, \( N_b = 0, 2, 4, 6 \); and the average speed of the UEs, \( v = 0.3, 3, 10, 20 \) km/h. These parameters are chosen in order to create scenarios in which the network is close to the limit of its capacity, and an incoming call may effectively deteriorate the overall QoS. For every combination of these parameters we run several independent simulations.

In each simulation, the number of UEs does not change for the first 20 seconds. Then, a new call request event occurs. If the incoming call is accepted, the new UE connects to the eNB and performs a VoIP call during the following 20 seconds of the simulation, competing for spectrum resources with the other active UEs. On the contrary, if the incoming call is not accepted, only the previously active UEs continue

<table>
<thead>
<tr>
<th>TABLE I</th>
<th>SYSTEM PARAMETERS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Transmission power (dBm)</td>
<td>40</td>
</tr>
<tr>
<td>Downlink carrier frequency (MHz)</td>
<td>2730</td>
</tr>
<tr>
<td>Uplink carrier frequency (MHz)</td>
<td>1930</td>
</tr>
<tr>
<td>Bandwidth (MHz)</td>
<td>1.4</td>
</tr>
<tr>
<td>Bandwidth (number of RBs)</td>
<td>6</td>
</tr>
<tr>
<td>eNB antenna height (m)</td>
<td>50</td>
</tr>
<tr>
<td>UE height (m)</td>
<td>2</td>
</tr>
<tr>
<td>Simulation area (m²)</td>
<td>1560x1560</td>
</tr>
<tr>
<td>Simulation time (seconds)</td>
<td>40</td>
</tr>
<tr>
<td>VoIP codec</td>
<td>G711</td>
</tr>
</tbody>
</table>
The RR scheme works as follows. It is a state-of-the-art CAC scheme, i.e., the resource reservation scheme described in Sec. V. We run $T = 6000$ testing simulations, and we can divide the results of the simulations in $T_a$ cases in which the CAC scheme is accepting the incoming call, and $T_d$ cases in which the CAC scheme is dropping the incoming call, with $T_a + T_d = T$. We define the ratio of FN as:

$$r_{FN} = \frac{1}{T_a} \sum_{t_a=1}^{T_a} \mathbb{1} \left( R^{(aa)}(t_a) < \tau_R \right) ,$$

where $R^{(aa)}(t_a)$ is the $R^{(aa)}$ relative to the $t_a$-th simulation with incoming call accepted, and $\mathbb{1} (\cdot)$ is the indicator function.

In a similar way, we define the ratio of FP as:

$$r_{FP} = \frac{1}{T_d} \sum_{t_d=1}^{T_d} \mathbb{1} \left( R^{(aa)}(t_d) \geq \tau_R \right) ,$$

where for each of the $T_d$ simulations in which the CAC is choosing to drop the incoming call, we need to simulate also the case in which the call is accepted, in the same exact conditions, in order to be able to calculate $R^{(aa)}(t_d)$.

Before comparing the performance of the two CAC schemes, it is important to make another consideration. While the NN is directly predicting the value of the $R^{(aa)}$, the BN is providing a distribution over the possible values of the $R^{(aa)}$. This gives the CAC the additional flexibility to choose the probability threshold introduced in Eq. (15). To study how $p_{T}$ is affecting the performance, in Fig. 3-(a) we plot $r_{FN}$ and $r_{FP}$ as a function of $p_T$, in the case in which the value of $R^{(bd)}$ is available and in the case in which such value is not available to the BN based CAC scheme. We observe that in both cases there exists an optimal value of $p_{T}$ to jointly minimize FN and FP. We can observe from the figure that the availability of $R^{(bd)}$ is very important, in particular because it allows the scheme to jointly keep the $r_{FN}$ and the $r_{FP}$ at a value below 0.15 even for a non-optimal choice of $p_{T}$.

In Fig. 3-(b) we use another definition of the ratio of FN:

$$r_{w}^{FN} = \frac{1}{T_a} \sum_{t_a=1}^{T_a} \mathbb{1} \left( R^{(aa)}(t_a) < \tau_R - \Delta_{r} \right) .$$

In other words, we consider as a FN (wrong CAC decision) only the case in which the incoming call is accepted and the $R^{(aa)}$ falls significantly under the threshold $\tau_R$. We have set $\Delta_{r} = 5$. Similarly, we define the FP as:

$$r_{w}^{FP} = \frac{1}{T_d} \sum_{t_d=1}^{T_d} \mathbb{1} \left( R^{(aa)}(t_d) \geq \tau_R + \Delta_{r} \right) .$$

We notice that in this case, especially when $R^{(bd)}$ is available to the CAC, the error is close to 0 for both $FN$ and $FP$, and for a wide range of $p_T$.

Finally, in Fig. 4 we show a performance comparison among the NN, the BN and the RR CAC schemes. The performance of the NN and the BN changes as a function of the length of the training set available to learn the inference engine, while the performance of the RR scheme does not depend on the training set. In order to evaluate the true performance of each scheme it is necessary to jointly observe the $r_{FN}$ and the $r_{FP}$, or $r_{w}^{FN}$ and $r_{w}^{FP}$. In Fig. 4-(a) we plot the $r_{FN}$ and the $r_{w}^{FN}$, while in Fig. 4-(b) we plot the $r_{FP}$ and the $r_{w}^{FP}$. We notice that, as expected, the RR scheme performs well in terms of $r_{FP}$, while it performs poorly in the terms of $r_{FN}$.

In other words, it can not balance well between the need to meet QoS requirements, which requires a low value of $r_{FN}$, and a full exploitation of the system resources, which requires...
when we evaluate CAC schemes perform well in both cases. In particular, for the BN, the NN and the RR CAC scheme.

Moreover, the BN scheme can also outperform the NN CAC scheme.

In this paper we designed NN and BN models for CAC in an LTE system. We studied how best to select the most relevant parameters. In future work, we plan to exploit the flexibility of these cognitive networking techniques in other challenging networking scenarios.

ACKNOWLEDGMENTS

This work at UCSD was partially supported by the U.S. Army Research Office, Grant No. W911NF-11-1-0538, and by the Qualcomm Institute. The work at CTTC was partially funded by the Spanish Ministry of Science and Innovation under grant number TEC2011-29700-C02-01 (project SYMBIOSIS), by the Catalan Regional Government under grant 2009SGR-940, by the European Commission in the framework of the COST Action IC0902 and by the FP7-ICT Network of Excellence ACROPOLIS.

REFERENCES


