Received 3 March 2021; accepted 27 March 2021. Date of publication 30 March 2021; date of current version 9 April 2021.

Digital Object Identifier 10.1109/OJCOMS.2021.3069801

Neural Network Cognitive Engine for Autonomous and Distributed Underlay Dynamic Spectrum Access

FATEMEH SHAH-MOHAMMADI [1,2] (Member, IEEE), HATEM HUSSEIN ENAAMI [1] (Member, IEEE), AND ANDRES KWASINSKI [3] (Senior Member, IEEE)

[1]Kate Gleason College of Engineering, Rochester Institute of Technology, Rochester, NY 14623, USA

[2]Department of Population Health Science and Policy, Icahn School of Medicine at Mount Sinai, NY 10025, USA

[3]Department of Computer Engineering, Rochester Institute of Technology, Rochester, NY 14623, USA

CORRESPONDING AUTHOR: F. SHAH-MOHAMMADI (e-mail: fatemeh.shahmohammadi@mountsinai.org)

ABSTRACT Two key challenges in underlay dynamic spectrum access (DSA) are how to establish an interference limit from the primary network (PN) and how cognitive radios (CRs) in the secondary network (SN) become aware of the interference they create on the PN, especially when there is no exchange of information between the two networks. These challenges are addressed in this paper by presenting a fully autonomous and distributed underlay DSA scheme where each CR operates based on predicting its transmission effect on the PN. The scheme is based on a cognitive engine with an artificial neural network that predicts, without exchanging information between the networks, the full adaptive modulation and channel coding configuration for the primary link that is received with highest power by a transmitting CR. By managing the effect of the SN on the PN, the presented technique maintains the relative average throughput change in the PN within a prescribed maximum value, while also finding transmit settings for the CRs that result in throughput as large as allowed by the PN interference limit. Simulation results show that the ability of the cognitive engine in estimating the effect of a CR transmission on the full adaptive modulation and coding (AMC) mode leads to a very fine resolution underlay transmit power control. This ability also provides higher transmission opportunities for the CRs, compared to a scheme that can only estimate the modulation scheme used at the PN link.

INDEX TERMS Cognitive radio, underlay dynamic spectrum access, NARX neural network, adaptive modulation and coding.

I. INTRODUCTION

THE COGNITIVE radio (CR) paradigm is seen as a key solution to the radio spectrum scarcity problem stemming from the inefficient static spectrum allocation policies [1] and the ever growing wireless data traffic. A CR is a wireless device with the ability to autonomously gain awareness of its surrounding wireless network environment and to learn how to adapt its operating parameters to best meet the end-user goals [2]. As a result, CRs have frequently been considered as an enabling technology for dynamic spectrum access (DSA). Through DSA, a network of CRs, called the secondary network (SN), operates by sharing the radio spectrum with a primary network (PN) which is the incumbent owner of the spectrum band in use. Of the three main DSA approaches (overlay, underlay, and interweave DSA [3]), this paper focuses on underlay DSA. In underlay DSA, CRs in the SN (also called "secondary users" - SUs) transmit over the same spectrum band being used by the PN by limiting their transmit power level so that the interference they create on the PN remains below a tolerable threshold [4].

In order to deploy an interweave DSA scheme, the main challenges that need to be addressed are the identification of "spectrum holes," the estimation of their duration, and the coordination between terminals on common available channels. Over the years, several works have contributed approaches that address these challenges over ever more challenging scenarios, [5], [6], [7], [8], [9]. However, in

this work we focus on underlay DSA, where there is not such a robust timeline of progress in addressing the main realization challenges as for interweave DSA. In order to deploy an underlay DSA scheme in a setup where there is no exchange of information between the PN and the SN, the two key challenges that need to be resolved are: (1) how to establish the interference threshold for the PN links, and (2) how the SUs autonomously become aware of the interference they create on the PN. These two challenges remain largely unsolved, especially when following the aforementioned practical operating setup where there is no exchange of information between primary and secondary networks. The main contribution of this paper is to address these two challenges in underlay DSA.

For underlay DSA, in the past, in order for the SN to assess its effect on the PN and protect the PN transmissions, researchers have proposed different techniques that usually assume that the secondary transmitter (SU_{TX}) knows the gain of the primary channel (that between the primary transmitter, PU_{TX}, and the primary receiver, PU_{RX}) and/or the cross-channel gain from the SU_{TX} to the PU_{RX}, [10], [11], [12], [13], [14], [15], [16], [17], [18]. The common theme between these works is that they make use of the information that is sent over a feedback channel in the PN. Since feedback channels are part of most wireless communication standards [19], [20], they have often been used, under the assumption that SUs can access them, to not only estimate the primary channel gain, but also assess the effect of the SN on PN transmissions. Examples of a CR obtaining information about the primary link from a PN feedback channel are found in [11], [12], [13], [14], [18] for the case of the rate/power control feedback channel, in [15] and [16] for the ARQ feedback channel, and in [17] for the feedback of the channel state information (CSI). In general, by relying on listening to feedback channels from the PN all works mentioned above share a setup where primary and secondary networks are not completely separated and exchange information with each other. In fact, the access of a control channel from another network calls into question as to whether there are really two separate network or, as we would argue, a single network with two different types of nodes.

With a different approach, the works in [21] and [22] proposed solutions to obtain the cross-channel gain without listening to the PN feedback channels. The typical process in these works consists of the SU observing a change in the primary's waveform power and/or modulation order that results from the transmission of a probe message from the SU [16], or the SU acting as a relay by sending the amplified version of the signal received from the PU [17]. However, in these works a CR transmitter can not estimate the cross-channel gain, and later assess the effect of SN on PN, unless it observes a change in primary signal power and/or modulation order. Moreover, these works did not considered the combined effect from scenarios with multiple links in the PN and the SN, as they focused on a setup with one link in each network.

In contrast to these works, the technique to be presented here is not limited by the need to observe a change in transmit power or modulation order and is able to directly estimate the effect of an SN transmission on the PN much more accurately by estimating not only the modulation order but, in addition, also estimating the channel coding rate in a PN link. Moreover, the work herein is on scenarios consisting of multiple links in the PN and SN. In addition, our presented technique meets a key requirement by not relying on any information exchange between the primary and secondary networks.

Our proposed technique takes advantage of the use at the PN of adaptive modulation and coding (AMC), a technique where the modulation scheme and channel coding rate (a pair of parameters known as the AMC mode) are adapted based on the quality of the transmission link. The use of AMC has been part of all high performance wireless communications standards developed over the past two decades and, thus, expected to be used by a typical PN [23], [24], [25]. Since the AMC mode in use depends on the link's SINR, by estimating the AMC mode used in a primary link, it becomes possible for a CR to learn the signal-to-interference-plus-noise ratio (SINR) experienced at that link and the corresponding throughput, from which the effect of CR transmission on the PU link can be assessed. Specifically, we propose an underlay DSA technique that configures the transmit power for a SU based on estimating the throughput at a PN link corresponding to the AMC mode that would be chosen based on the interference created by the SU's transmission.

At the same time, leveraging the use of adaptive modulation in the PN will allow us not only to assess the effects of the SN on the PN (this is, the first challenge for realization of underlay DSA), but will also allow us to establish the PN interference threshold (the second challenge for realization of underlay DSA). As shown in [26], the use of adaptive modulation allows for the background noise to increase up to a certain level before the average throughput in a network starts to decrease. In the context of underlay DSA where the PN does not exchange any information with the SN, the interference imposed on the PN by an underlay-transmitting CR can be seen as a background noise for the PN, that can be increased up to a level which does not affect the average throughput in the primary network. Therefore, a CR can become aware about the interference that is creating on the primary link and decide on its transmit power by inferring the change in throughput at the primary link (details in Section IV).

Our technique addresses the two challenges in underlay DSA based on a cognitive engine (CE) at the SUs that is equipped with a non-linear autoregressive exogenous neural network (NARX-NN) that estimates the throughput on a PN link (equivalently, the full AMC mode given by modulation order and channel coding rate). The proposed CE uses as

input an estimate of the modulation scheme being used at the PN link. As such, to the best of our knowledge, our presented cognitive engine is the first to be capable of estimating the channel coding rate setting for an AMC mode, a capability well beyond the estimation of just the modulation scheme (which is a mature area of signal processing or machine learning technology). While modulation classification, the technique to estimate the modulation scheme used in a radio waveform, is applied in our DSA technique to derive an input for the cognitive engine, we deem it to be a well-researched technology and is not the subject of our work. Instead, we leverage the already large volume of existing research. In this regard, the work in [27] studied modulation classification based on second and higher order time variant periodic cumulant function of the sensed signal for which it is required a prior knowledge of the signal parameters. Authors in [28] used the same framework to perform signal pre-processing, along with utilizing artificial neural networks to address the issues associated with classification when the signal parameters are unknown. The work in [29] proposed a fully automated modulation classification scheme which employs two stages of signal processing to classify the modulation of an incoming signal.

Therefore, for an scenario of underlay DSA where networks do not tap into feedback or control channels from another network, the main contributions of this paper are the following:

1) Answering the question of how to establish the PN interference threshold in underlay DSA (the first key challenge) by asserting that for a PN using AMC the interference from the SN is indistinguishable from background noise and, thus, the threshold is the level of interference power when the AMC in the PN begins to be unable to maintain the average throughput seen in the PN when there is no SN.
2) Present a technique based on the use of a neural network, transmission of probe messages from the SN, and leveraging the use of AMC at the PN, where a CR in the SN is able to estimate the channel coding rate and modulation order (*full* AMC mode) that would be set at the link in the PN that it receives with highest power.
3) Leverage the previous contribution to introduce a technique that allows a transmitting SU to autonomously learn with a fine resolution the interference it would create on the PN, addressing the second key challenge in underlay DSA. This contribution, together with the first one, realizes a fully autonomous and distributed underlay DSA scheme. This scheme features a finer control knob for a more accurate power allocation at the SN with less harmful effect on PN transmissions as compared to techniques that rely only on the estimation of modulation order (from applying signal processing on the transmission waveform).

The above contributions are corroborated in simulations that will show that the presented technique is able to maintain

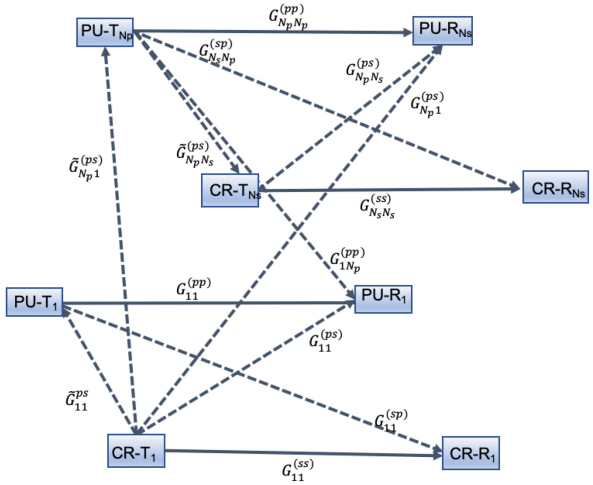

FIGURE 1. Considered network model composed of N_P primary transceivers and N_S cognitive radio users.

the relative change in PN average throughput within a prescribed fine-grained target maximum value (as an indicator of maximum allowed interference in the PN), while at the same time finding transmit settings for the SUs that will result in as large throughput in the SN as could be allowed by the PN interference limit. As such, while succeeding in its main goal of autonomously and distributively determining the transmit power of the SUs such that the interference they create remains below the PN allowed interference limit, our proposed technique is also able to manage the tradeoff between the effect of the SN on the PN and the achievable throughput at the SN. Specifically, simulation results will show that for the proposed system with a target PN maximum relative average throughput change of 2%, the achieved relative change is less than 3%, while at the same time achieving useful average throughput values in the SN between 180 and 50 kbps. In addition, it will be seen that the implementation of a variation of the proposed scheme that reduces three times the overhead from transmitting probe messages still exhibits the ability to finely control the effect on the primary network throughput, although, as is to expect, compared to the case of sending all probe messages it increases the PN relative average throughput change by at most 1%, only at the very low PN load of 0.16.

The rest of this paper is organized as follows. Section II presents the overall system setup. The rationale on how AMC can be leveraged to address the main two challenges associated with underlay DSA is outlined in Section III. Section IV describes our proposed distributed underlay DSA technique. Simulation results are presented in Section V, followed by conclusions in Section VI.

II. SYSTEM SETUP

We consider a primary network with N_P active primary links coexisting with N_S active secondary links, with both networks transmitting over the same frequency band. The system model is shown in Fig. 1, where $G^{(ps)}_{ij}$ denoted as cross-channel gain is the path gain from jth. transmitting SU

to the receiver in ith. primary link, $G_{ij}^{(ss)}$ is the path gain from the transmitter in the jth. secondary link to the receiver in the ith. secondary link, $G_{ij}^{(sp)}$ is the path gain from the transmitter in jth. primary link to the receiver in ith. SU link, $G_{ii}^{(pp)}$ is the path gain from the ith. primary transmitter to its corresponding receiver, and $G_{ii}^{(ss)}$ is the path gain from the ith. secondary transmitter to its corresponding receiver. While, $\tilde{G}_{ij}^{(ps)}$ is the path gain from the transmitter in jth. secondary link to the transmitter in ith. primary link. The model for all channel gains is discussed in Section IV-B.

In this section we focus on describing the operation of the PN, which is incumbent to the considered radio spectrum band. Section IV will present the underlay DSA scheme implemented in the SN. We assume that the PUs receive service from N_{PBS} transmitting base stations (BSs), and we call the ratio N_P/N_{PBS} as the primary network load. Each PU is assigned to the base station that presents the best channel gain. In addition, AMC is used in all transmissions (primary and secondary networks). This means that a transmitter has information about its link quality, in terms of SINR, and based on this assessment chooses, from a set of options, the modulation scheme and channel coding rate that results in highest throughput while at the same time meeting a maximum bit error rate (BER) limit.

Let $P_i^{(p)}$ denotes the transmit power in the ith. active primary link ($i = 1, 2, \ldots, N_P$). Then, in the absence of the SN, the SINR in the ith. primary link (which is used to decide on the AMC mode) can be written as,

$$\gamma_i^{(p)} = \frac{G_{ii}^{(pp)} P_i^{(p)}}{\sum_{j \neq i} G_{ij}^{(pp)} P_j^{(p)} + \sigma_p^2}, \quad i = 1, 2, \ldots, N_P. \qquad (1)$$

where σ_p^2 is the background noise power.

In addition to AMC, without loss of generality, we adopt for the primary network the variable transmit power allocation algorithm proposed in [26]. This is an iterative power control algorithm that converges to a global optimum solution that maximizes the product of SINRs across all active links. In the algorithm, the transmit power at the ith. primary link is updated as,

$$P_i^{(p)} \longleftarrow \left(\sum_{j \neq i} \frac{G_{ji}^{(pp)}}{\sum_{m \neq j} G_{jm}^{(pp)} P_m^{(p)} + \sigma_p^2} \right)^{-1}. \qquad (2)$$

III. LEVERAGING ADAPTIVE MODULATION AND CODING IN UNDERLAY DSA

Before presenting our proposed underlay DSA technique, in this Section we outline the main ideas on how AMC can be leveraged to address the two main challenges associated with underlay DSA: 1) How to establish the interference threshold in the PN and 2) how SUs can autonomously become aware of the interference they create on the PN. As previously noted, AMC (or, as also called, link adaptation) has been used during the past two decades in practically all high-performance wireless communications standards and, as such, is assumed to be used in both the primary and secondary networks in this work. In this section we summarize some of the main features of AMC that are relevant to our fully autonomous and distributed underlay DSA scheme. For clarity of presentation, in what follows, at times we will refer to the PU link that is received with highest power at an SU as the "nearest" PU link (although in a minority of cases, and due to the properties of the wireless channel, the PU link that is nearest in geographical terms may not be the one that is received with highest power). We will emphasize this lax wording by writing the word "nearest" between quotes. It is assumed that, while the PN has granted permission to the SN to operate in its spectrum band using underlay DSA, during regular operation conditions the PUs are not expected to track how many SUs are active, to the extent that the PUs are oblivious to the existence of the SUs and treat the interference from transmitting CRs as additional noise at their receiver.

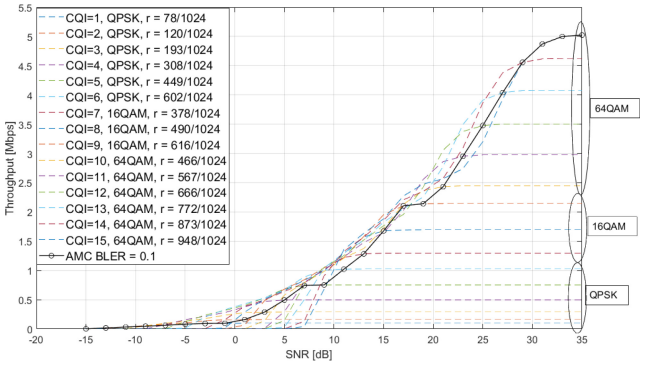

FIGURE 2. Throughput for each CQI and maximum throughput adaptive selection per RB for the AMC scheme of an LTE system under a Pedestrian B channel.

Fig. 2, obtained using the MATLAB LTE Link Level Simulator from TU-Wien [30], shows the throughput versus signal-to-noise ratio (SNR) performance for the LTE system under a Pedestrian B channel that will serve without loss of generality as the assumed AMC setup for the rest of this work (further setup details for the simulation results shown in Fig. 2 are discussed in Section IV-B and Section V). In LTE, AMC consists of 15 different modes (each for a different "Channel Quality Indicator - CQI) based on three possible modulation schemes which will be called '*type 0*' for QPSK (used for the smaller SNR regime), '*type 1*' for 16QAM (used at intermediate SNRs), and '*type 2*' for 64QAM (used for the larger SNRs). In AMC, alongside the modulation order, channel coding rate is also adapted [31], [32]. Because of this, an AMC mode is formed by a choice of both a modulation order and a channel coding rate. Fig. 2 shows the throughput of one LTE resource block[1] achieved for each AMC mode (each curve is labeled with the corresponding CQI value and AMC mode settings, formed by

1. We will use the 180 kHz of bandwidth associated with one LTE resource block as representing the spectrum band that is shared by the PN and SN. However, this does not imply that either the PN or SN are LTE networks.

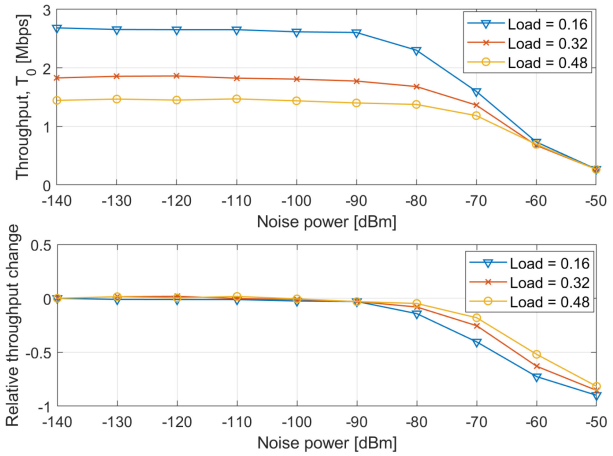

FIGURE 3. Throughput vs. noise power in the primary network.

the modulation type and channel coding rate) and the overall performance curve of the AMC scheme, where the modulation type and code rate are chosen to maximize throughput but with a constraint on the block error rate (BLER) not to exceed 10%. During transmission, the transmitter chooses the AMC mode with maximum throughput at the estimated SNR of the link.

As was discussed in [26], the use of AMC in conjunction with transmit power control allows the background noise to increase up to a maximum value without significantly affecting the network average throughput. In the context of underlay DSA, this maximum noise value can be interpreted as the maximum value for the combined powers of background noise and interference from the SN. To see this important point in detail, consider Fig. 3, which we obtained as an expanded version of [26, Fig. 5], now for different network loads (N_P/N_{PBS}) when using the LTE AMC setup just described and the power control algorithm from [26]. The figure shows the average throughput achieved in the PN by itself (the presence of an SN is not included in this result) as the background noise power increases. On the top, the figure shows the property associated with the use of adaptive modulation that for all network loads the average throughput remains approximately constant until noise power becomes sufficiently significant. This is not a trivial observation as networks with a higher load are operating in a regime more influenced by the interference rather than the noise, but it is clear from the results that adaptive modulation manages to maintain a balance between interference and noise-dominated operation. The bottom of Fig. 3 shows as a function of noise power, the change in throughput relative to the throughput at the lowest noise power. The result exposes the remarkable property that the interference that would be imposed by the SN, which can be considered by a PN that is unaware of the presence of another network as part of the background noise, will not significantly affect the average throughput in the primary network as long as the combined SN interference, the interference by other primary links and actual background noise remains below a threshold approximately equal to −80 dBm (although this number somewhat depends on the network load). Moreover, relative throughput change starts to decrease at approximately the same value of noise power for all network loads (around −90 dBm). This is a consequence of the link adaptation performed through AMC. Moreover, throughput relatively decreases faster with smaller network loads. We believe that this is because at smaller network loads, interference across the network is lower and a larger ratio of transmissions use the less resilient higher rate modulation types.

While AMC entails the adaptation of both modulation order and channel coding rate, it can be seen in Fig. 2 that the modulation order provides a coarse adaptation and that the channel coding rate enables a finer adaptation within each of the choices for modulation order. Moreover, an important difference between modulation order and channel coding rate adaptations is that while it is possible for a passive "listener" of the AMC transmission to infer the modulation order through the use of modulation classification signal processing, it is not possible to infer the channel coding rate (or equivalently the full AMC mode). On the other hand, as can be seen in Fig. 2, when the noise power increases (or equivalently the SN interference increases) the effect of AMC operation will be to change the AMC mode to one associated with a smaller CQI. At the same time, the modulation scheme with the smallest order is used for the smallest operating SINRs (because the transmission of less bits per symbol is more resilient to interference and noise). As the interference from secondary transmissions increases, primary links that are already using, in the absence of secondary transmissions, the lowest modulation order scheme will not switch to other modulation schemes because there is no other modulation scheme with fewer bits per symbols to switch to. This means that transmissions from an SU that otherwise would generate a change in modulation scheme would not result in any change when the "nearest" primary link is already transmitting with the modulation scheme with smallest bits per symbol. As a result, estimation of the modulation order provides for the SU transmitter with a control nob to infer the effect of SN transmission on the PN but with severe limitations due to the coarse information provided by the estimated modulation order. However, as seen in Fig. 2, although the increase in interference to the PU (and particularly increase in SU transmit power) may not lead to change in modulation order, it indeed results in the change in channel coding rate. Thus, the estimation of the channel coding rate (and equivalently the full AMC mode) is necessary in order to be able to finely estimate the effect of SU transmissions. Indeed, there exists a large body of research in the area of modulation classification with some representative works briefly discussed in Section I (e.g., [27], [28], [29]). Consequently, the techniques that existed before our work have been limited to use only the coarse information derived from the modulation order (e.g., [21], [22]) or to rely on the sharing of information between the PN and the SN through the SUs accessing the control feedback channel in the PN

to learn the finer information associated with the channel coding rate used in a primary link (e.g., [16]). As will be seen, our proposed technique is able to overcome the limitation of inferring the channel coding rate without exchange of information between the PN and the SN and, as such, be able to use the fine-grained information provided by channel coding rate without the SN tapping into any control channel of the PN.

In the following we develop for a transmitting CR a model to estimate the interference limit from the PN. Regarding Fig. 3 it was stated that the property associated with the use of adaptive modulation allows the background noise to increase up to a maximum value without significantly affecting the network average throughput; and that the maximum noise value can be interpreted as the maximum value for the combined powers of background noise and interference from the SN. For the purpose of underlay DSA, it is convenient to model the PN average throughput shown in the bottom of this figure as equal to the average throughput achieved without the presence of the SN minus an average throughput loss that is a function of an equivalent interference from the SN as experienced across the PN. This is, if $T_0(\sigma_p^2)$ is the PN average throughput without the SN (explicitly expressed to be dependant on the background noise power σ_p^2 as seen in the top plot of Fig. 3), we model the PN average throughput T_p as $T_p = T_0(\sigma_p^2) - T_l(I)$, where T_l is the throughput loss and I is the equivalent interference from the SN as experienced across the PN. Next, the throughput loss is modeled according to the SNR gap approximation for the Shannon's channel capacity formula, [33], yielding

$$T_p = T_0 - B\log_2(1 + \Gamma\, 10^{I/10}), \tag{3}$$

where B is the PN system bandwidth, I is the equivalent interference generated by the SN on the PN (measured in dBm), and Γ is the SNR gap that accounts for the use of practical coding and transmission mechanisms (we have also embedded into Γ a factor of 0.001 stemming from the conversion of units of I from dBm). Being the relative average throughput change in the PN $T_{\%} = (T_p - T_0)/T_0$, from (3) we have,

$$T_{\%} = -\frac{1}{\zeta}\log_2(1 + \Gamma\, 10^{I/10}), \tag{4}$$

where $\zeta = T_0/B$ is the PN spectral efficiency that is achieved without the SN's effect. Figure 4 compares the relative average throughput change from Fig. 3 and its approximation (4) for loads equal to 0.16 and 0.48 only (for clarity of the figure). The approximation uses for the spectral efficiency calculation the same system bandwidth B as in the results in Fig. 3 (180 kHz) and, of course, the values T_0 from Fig. 3 at noise power equal to -140 dBm. Fig. 4 validates our model given by (3) and (4) by showing a good approximation for the relative average throughput change (the approximation shows less accuracy when the relative average throughput change grows beyond 40% which is of no concern because these are values of relative average throughput change too large to be used in the practical operation of underlay DSA). More importantly, Fig. 4 includes a second abscissa to highlight the meaning of the equivalent interference in the context of underlay DSA where the PN uses AMC. As mentioned earlier, the curves in Fig. 3 show for a network that uses AMC the change in average throughput as the background noise power increases. In the context of underlay DSA, the absence of information exchange between the PN and the SN implies that, in principle, the PN is oblivious of the presence or not of the SN. Therefore, the PN would experience the interference from the SN as an increase in the background noise, which is what we identify as the equivalent interference I. Therefore, the magnitude T_0 by definition, becomes the PN average throughput when the equivalent interference is $I = 0$. The role of the equivalent interference on the PN is illustrated in Fig. 4 with the second abscissa, showing the increasing equivalent interference with the same value as the background noise. In this way, for example, if the actual background noise is -140 dBm, an equivalent interference of $I = 30$ dBm is performance-wise perceived by the PN as a total background noise of -110 dBm.

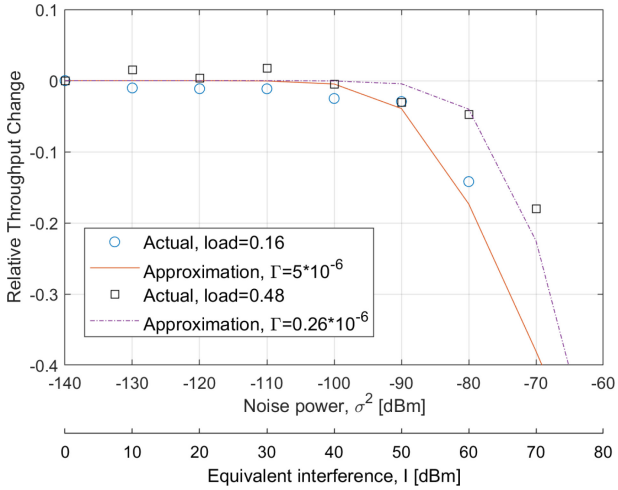

FIGURE 4. Relative average throughput change and approximation (4).

The goal of the proposed underlay DSA mechanism at the SN is to find the transmit power at the SUs that results in an equivalent interference, denoted as interference limit I_0, that is as large as possible (to increase throughput at the SN) while the relative average throughput change in the PN remains below a limit that we will denote as ϵ. From (4), this maximum equivalent interference, measured in dBm and denoted as interference limit, can be expressed as a function $I_0(\epsilon)$ of the limit ϵ as,

$$I_0 = 10 * \log\left(\frac{2^{-\epsilon\zeta} - 1}{\Gamma}\right). \tag{5}$$

Using this expression, we have calculated, for example with $\epsilon = -0.05$, values of maximum equivalent interference equal to 51.3 dBm and 61.1 dBm for load equal to 0.16 and 0.48, respectively (for a background noise of -140 dBm). The expression in (5) provides an answer to one of the

two main challenges in underlay DSA: the establishment of the interference threshold from the PN. However, the second challenge remains: how the SUs autonomously become aware of the interference they create on the PN, relative to the limit I_0. This challenge is compounded by our setup with multiple links in the PN and the SN, and where the SUs can only use information they sense individually (the modulation scheme used at the "nearest" primary link), which leads to most of the many parameters that would be needed for an analytical solution being not accessible to the SUs. To solve this challenge we propose to follow a "black-box modeling" approach, where the CR implicitly or explicitly learns this information without the support of a pre-defined analytical model. For this, we propose the use of artificial neural networks because of their state-of-the-art status within techniques capable of learning the relation between the equivalent interference and the SUs' transmissions (that is implicitly represented within the training data) through black-box modeling. Indeed, artificial neural network are known as universal function approximator capable through training to learn the interrelation between the system variables without the need for a-priori knowledge of the underlying model. Leveraging these characteristics of artificial neural networks in our proposed technique allows for the SUs to address both challenges in underlay DSA by assessing their transmission effect on the PN without the need to calculate intermediate magnitudes (e.g., the cross-channel gain(s) as seen in [21]) and, from this knowledge, identifying the transmit power setting that meets the PN interference limit.

IV. AUTONOMOUS AND DISTRIBUTED UNDERLAY DSA FOR COGNITIVE RADIO NETWORKS

We now present the main contribution of this paper: a fully autonomous and distributed underlay DSA technique for a secondary CR network. The operation of the SN is fully autonomous and ad-hoc. This means that SUs do not rely on any exchange of information with the PN and with other SUs (other than between transmitter-receiver pairs) and that the transmission control algorithm in the SN is distributed. While there is no information exchange between primary and secondary networks, it is assumed that the SN has knowledge of what is the radio access protocol used for the PN operation (through standards or publicly available information). Because of this, the SN has knowledge of the underlying timing operation in the primary network at a broad level (e.g., when frame transmission starts, etc.) so that CRs are able to sense and transmit at appropriate times. Detailed (symbol-level) knowledge of timing at the CRs is not required or assumed.

As stated, fully autonomous operation implies that the primary and secondary networks operate as being unaware of the other (except for the mild timing assumption), considering the other network transmissions as out-of-network interference akin to background noise. Then, when adding an underlay SN, the SINR at the receiver of the ith. PN link now becomes,

$$\gamma_i^{(p)} = \frac{G_{ii}^{(pp)} P_i^{(p)}}{\sum_{j\neq i}^{N_P} G_{ij}^{(pp)} P_j^{(p)} + \sum_{j=1}^{N_S} G_{ij}^{(ps)} P_j^{(s)} + \sigma_p^2}, \tag{6}$$

where $P_j^{(s)}$ is the transmit power from the jth. transmitting SU and $G_{ij}^{(ps)}$ is the path gain from a transmitting SU j to a PU i. Likewise, it is assumed that transmissions on the SN also make use of AMC, which is configured based on the corresponding link SINR. For this, the SINR, $\gamma_i^{(s)}$, at the receiver of the ith. SN link is:

$$\gamma_i^{(s)} = \frac{G_{ii}^{(ss)} P_i^{(s)}}{\sum_{j\neq i}^{N_S} G_{ij}^{(ss)} P_j^{(s)} + \sum_{j=1}^{N_P} G_{ij}^{(sp)} P_j^{(p)} + \sigma_s^2}, \tag{7}$$

The underlay DSA operation is implemented in a cognitive engine at each SU. Fig. 5 illustrates the block diagram of an SU with its cognitive engine, as well as other processing steps. The function of the cognitive engine will be to predict for different transmit settings the throughput $\hat{T}$ (equivalently the CQI or full AMC mode) of the "nearest" PN link and use this information to set power control and AMC parameters for a transmitting SU. Under our imposed practical condition of no exchange of information between the primary and secondary networks, a CR can only estimate the modulation order (after performing modulation classification signal processing on the PN transmissions) and cannot know the coding rate in use by accessing feedback or control channels in the PN. Moreover, practical limitations further dictate that the modulation classification can only be performed on the one primary transmission that it is being received with strongest power, making the other transmissions be interference. It is assumed that the estimation of modulation type does not rely on the SU accessing any information from the PN feedback channel and is error free using any of the methods existing in the literature (in Fig. 5 we assumed that modulation classification is done using the technique in e.g., [29]).

To accomplish its function, the cognitive engine at an SU follows the following procedure: First the SUs avoid transmission while the PN initially adjusts its power and AMC parameters using the iterative power control algorithm (2). During this initial listening stage, the SUs listen to the PN transmissions and infer the modulation order used by their "nearest" primary link. At this stage, each transmitting SU use any of the existing modulation classification techniques to obtain an estimate of the modulation order used by their corresponding "nearest" primary link when the SN is not transmitting. Next, each SU proceeds to send a series of short probe messages configured with different pre-defined transmit powers. Following the transmission of each probe message, the SU listen to the PN transmissions and infer the modulation order used by their "nearest" primary link in response to the probe message. After transmitting all probe messages, each transmitting SU forms two sequences: (1) a sequence $\vec{u}_1$ with the first entry equal to zero followed by the sequence of transmitted power of each probe message, and

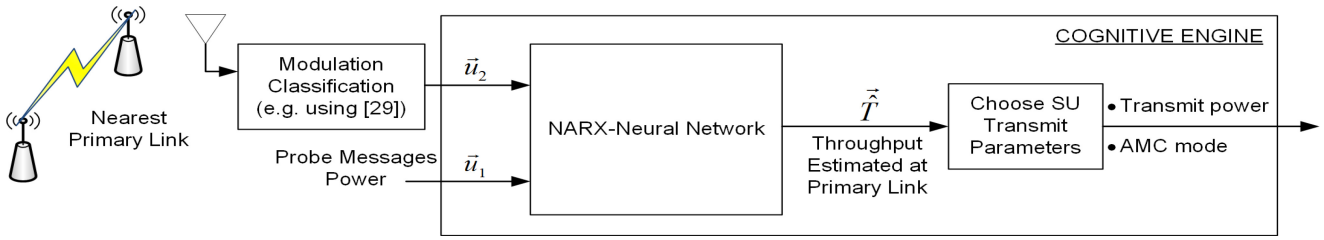

FIGURE 5. Block diagram of an SU with a cognitive engine based on the NARX neural network.

(2) a sequence $\vec{u}_2$ with the first entry equal to the modulation order of "nearest" primary link during the initial listening stage (before transmission of probe messages) followed by the sequence of modulation order in the "nearest" primary link corresponding to each of the probe messages.

As shown in Fig. 5, the two sequences $\vec{u}_1$ and $\vec{u}_2$ are fed to the cognitive engine. At the cognitive engine, a first stage is tasked with providing the sequence of corresponding estimated throughput values at the "nearest" primary link, $\hat{\vec{T}}$. Since estimating the throughput is equivalent to estimating the CQI and the corresponding full AMC mode (channel coding rate and modulation order), the cognitive engine is able to provide an estimate of the "nearest" primary link SINR with a finer resolution than what could be derived from the modulation classification alone that is present at its input. Note that this sequence of estimated throughput values includes the one with no transmissions from the SN. The SU can use the sequence of estimated throughput values to infer what would be the effect of its transmission on the nearest primary link by comparing the change in throughput value against that without the SN transmission. As seen in Fig. 5, the SU uses this information to find its own transmission parameters such that the relative change in throughput of its "nearest" primary link remains below the predetermined limit. Importantly, in our underlay DSA operation, the cognitive engine takes advantage of its ability to obtain the finer resolution full AMC mode inference (not just modulation order) on the "nearest" primary link, instead of the coarser inference on modulation order that could be achieved with a modulation classification operation only. Additionally, those SUs that estimate that their "nearest" primary link is transmitting in the lowest rate AMC mode when the SN is not transmitting (because of already experiencing a very low SINR, likely leaving no room for added interference from the SN) are prevented from transmitting. This guarantees that the reduction in the average rate of the primary link experiencing the poorest channel quality (CQI equal to 1) is minimized.

As discussed at the end of the previous Section, the many challenges associated with the task performed in the first stage of the cognitive engine (estimation of the full AMC mode from the sequences of transmit power for the probe messages and corresponding modulation order) leads to the use of a Black-box modeling approach. In the proposed underlay DSA technique, a cognitive engine at each SN transmitter leverages the use of artificial neural network to learn the functional model of the interaction between the secondary and primary networks. Often, analytical models have been used to characterize the performance of the SN. For example, in [34] the BER performance of different modulation orders have been characterized using analytical models. However, such analytical approaches are restricted to specific setups (e.g., a single link at the PN and the SN). Following an analytical approach for our more general setup, with multiple links in the PN and the SN, and our goal of predicting the full AMC mode (modulation order and channel coding rate) at a PN link, would required for the SUs knowledge of multiple variables that cannot be known without exchange of information between the PN and SN (which would contradict a key condition in the setup). As an alternative approach to analytical models, black-box modeling uses example inputs and corresponding outputs to build a predictor to estimate output values for unforeseen inputs and variations of the system configurations.

Neural Networks (NNs) have become increasingly popular as general purpose function approximators and, specifically, for dynamic system modeling [35]. Neural networks have been successfully applied to a number of black-box modeling and time series prediction tasks. Due to the inherent capability of neural networks to model nonlinear systems and their higher robustness to noise, they frequently outperform standard linear techniques when the time series are noisy and the dynamical system that generated the time series is nonlinear [36]. There is a growing number of works that have applied neural networks for various communication tasks such as channel decoding, estimating the features of the user channels and predicting the anomalies for wireless sensor networks [37], [38], [39]. For CRs, the feed forward neural network has been used in predicting the spectrum occupancy status [40] and designing a medium access control (MAC) protocol [41].

Due to the adaptation of SU's transmission to the PN interference threshold, an underlay network can be seen as an example of a dynamic system, and also the throughput in the PU can be seen as a time series with a temporal dependency. As a result, we have considered a neural network-based cognitive engine to specifically predict the throughput in a PU link and characterize the behavior of such dynamic system. In the case of one-step-ahead time series prediction tasks, only the estimation of the next sample value of a time series

is required. Therefore, the input contains only actual sample points of the time series, without feeding back the output as a new input to the model. While considering multi-step-ahead or long-term prediction, the neural network's output should be fed back to the model as a new input for a finite number of time steps [42]. In this case, the components of this input to the neural network, previously composed of actual samples of the time series, are gradually replaced by previously predicted values. As a result, the multi-step-ahead prediction task is converted to a dynamic modeling task. In this case, the neural network model behaves as an autonomous system and tries to recursively emulate the dynamic behavior of the system that generated the nonlinear time series [43]. Compared to the one-step-ahead prediction, multi-step-ahead prediction and dynamic modeling are much more complex to deal with. However, neural networks models and in particular recurrent neural architectures play an important role in dealing with these complex tasks [44]. Elman introduced in [45] a class of recurrent neural models called simple recurrent networks (SRNs) which are essentially feedforward in the signal-flow structure with a few local and/or global feedback loops. A time delay neural network (TDNN), which is an adapted version of a feedforward multilayer perceptron (MLP)-like networks with an input tapped-delay line, can be used to process time series [44].

In the case of long-term predictions, a feedforward TDNN model will eventually behave similarly to the SRN architecture, since a global loop is needed to feed back the current estimated value into the model's input. Temporal gradient-based variants of the backpropagation algorithm are usually used to train the aforementioned recurrent neural networks [46]. However, training using gradient-based learning algorithms can be quite difficult in the case of systems with a long time temporal dependencies between their input-output signals [47]. In [48], [49], the authors claimed that such training is more effective in a class of simple recurrent network models called Nonlinear Autoregressive with eXogenous input (NARX). To prove their hypothesis these works have used two signals in abstract form, without targeting any particular application. The work in [50] tested the performance of NARX neural network on time series prediction using comparative experiments with real and artificial chaotic time series from diverse domains with different memory orders. The authors claimed that NARX recurrent neural networks have the potential to capture the dynamics of nonlinear dynamic systems. Authors in [51] also used the well-known chaotic laser and real-world variable bit rate video traffic time series to empirically evaluate the performance of NARX neural networks in long-term prediction tasks. This class of neural networks were proven to be powerful in pattern recognition and classification applications as well [52], [53]. However, none of the works mentioned above have explored the usage of NARX neural networks in wireless communications domain (in the context of cognitive radio). In this paper, we employ a NARX neural network for the first time in a wireless communications setting by choosing it to implement the cognition task at the first stage of the cognitive engine in each SU.

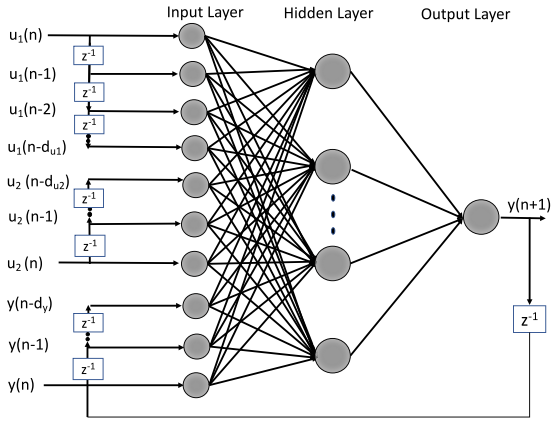

FIGURE 6. NARX neural network architecture.

A. ARCHITECTURE

With a topology as shown in Fig. 6 for the case of one hidden layer network, the NARX neural network output can be mathematically represented as, [54],

$$y(n+1) = f\big(y(n), y(n-1), \ldots, y(n-d_y); u_1(n), u_1(n-1), \ldots, u_1(n-d_{u_1}); u_2(n), u_2(n-1), \ldots, u_2(n-d_{u_2})\big), \quad (8)$$

where $u_1(n)$, $u_2(n)$ and $y(n)$ denote, respectively, the two inputs and one output of the model at discrete time step n, and $d_{u_1} \geq 1$, $d_{u_2} \geq 1$ and $d_y \geq 1$, $d_{u_1} \geq d_y$, $d_{u_2} \geq d_y$ are the inputs and output discrete delays, respectively. For its use in the cognitive engine, we configure the NARX neural network to have as inputs the sequence $\vec{u}_1$ of transmitted power of each probe message (with first element equal to zero), and the sequence $\vec{u}_2$ of modulation order in the "nearest" primary link corresponding to each of the probe messages (including the case of no SU transmission as first element). This implies that $d_{u_1} = d_{u_2}$. The output of the NARX neural network cognitive engine is the predicted throughput $\hat{T}(n)$ at the primary link "nearest" to the transmitting CR (which corresponds to a choice of AMC mode). The nonlinear mapping $f(\cdot)$ in (8) can be approximated, for example, by a standard multilayer neural network. If the non-linear mapping can be learned accurately by a neural network of moderate size (measured in terms of number of layers and number of artificial neurons in each layer), the resource allocation based on the output of the NARX neural network can be done in real time, since passing the input through the neural network only requires a small number of simple operations.

In Fig. 6 each circle represents an "artificial neuron", an elementary operation unit in the NARX neural network model. According to the input variables $u_1(n)$ and $u_2(n)$,

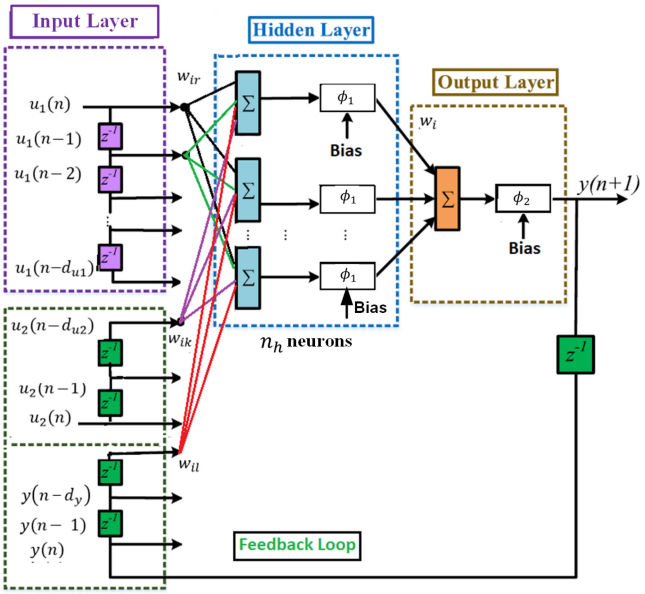

FIGURE 7. View of computing operations at NARX neural network during operation.

the output of i-th hidden layer neuron at time step n is obtained as:

$$H_i(n) = \phi_1\left(\sum_{r=1}^{d_{u_1}} w_{ir}u_1(n-r) + \sum_{k=1}^{d_{u_2}} w_{ik}u_2(n-k) + \sum_{l=1}^{d_y} w_{il}y(n-l) + a\right), \quad (9)$$

where w_{ir} is the connection weight between the input neuron $u_1(n-r)$ and i-th hidden neuron; w_{ik} is the connection weight between the input neuron $u_2(n-k)$ and i-th hidden neuron; w_{il} is the connection weight between the i-th hidden neuron and output feedback neuron $y(n-l)$; a is the bias of the hidden layer (not explicitly shown in Fig. 6) and $\phi_1(\cdot)$ is called the "activation function" for the hidden layer. Combining the hidden layer output, the final prediction can be given as follows:

$$\hat{y}(n) = \phi_2\left(\sum_{i=1}^{n_h} w_i H_i(n) + b\right), \quad (10)$$

where w_i is the connection weight between the i-th hidden neuron and predicted output; n_h is the number of neurons in the hidden layer; b is the bias of the predicted output; and $\phi_2(\cdot)$ is the output layer activation function. In our implementation of the NARX neural network, the activation function $\phi_1(\cdot)$ for the hidden layer is a sigmoid function while the activation function $\phi_2(\cdot)$ used for the output layer is linear (the input layer is not truly formed by artificial neurons but rather it is conventionally included as a representation of the connections of inputs into the neural network).

Figure 7 shows the NARX neural network but now emphasizing the elementary computing operations that are performed during operation. Table 1 shows the number of

TABLE 1. Computational complexity of NARX-NN architecture.

Elementary Operation	Number of Operations
Addition	$n_h(d_y + d_{u_1} + d_{u_2} + 4)$
Multiplication	$n_h(d_y + d_{u_1} + d_{u_2} + 4)$
Table lookup	n_h

these elementary operations that the cognitive engine needs to execute during one cycle of decision making (power allocation at one SU). In the table, the number of additions can be derived from direct inspection of Fig. 7 (including the addition of a bias at each neuron) and the number of multiplication arises from the multiplication of the input to each neuron by a weighting factor. Also, we have assumed that the nonlinear activation functions ϕ_1 is implemented through a table lookup (a common technique to implement non-linear function at low level of implementation). While more specific discussion will follow later in this same section, note that the implemented NARX neural network contrasts with many typical contemporary neural network implementations (e.g., AlexNet, [55]), which are based on tens of millions of parameters and input size in the order of thousands and, consequently, see a relatively quite high computational complexity during operation. The implemented NARX neural network has associated a number of parameters and inputs in the order of tens and, therefore, a low computational complexity. This is, of course, a combination of our neural network architectural choice but, also, the nature of the problem addressed with the neural network.

B. NARX NEURAL NETWORK TRAINING

The data set used to train the cognitive engine is an important element in designs based on black-box modeling as the one proposed here. It is from this training data that the cognitive engine learns a representation of the environment where it will operate and from which it will estimate throughput at the "nearest" primary link. In this work, we trained the cognitive engine with a data set that contains multiple examples representing an environment, with its statistical variability, that represents a common practical wireless scenario. Yet, it is noteworthy to keep in mind that an appeal of the black-box modeling approach is that the cognitive engine can learn to operate in another environments, or even in multiple concurrent environment, as long as it is trained using a data set that contains examples representative of the intended operating conditions.

Training data for the neural network cognitive engine was collected from a simulation carefully designed to reflect highly realistic scenarios. The simulation is built around a system comprising a primary and a secondary network that coexist over an area we have deemed the "playground." The PN determines the "playground" geometry, consisting of a five-by-five grid of BSs with neighboring base stations separated by a distance of 200 m. To avoid unrealistic edge effects in the playground, the grid wrapped around

all of its edges. Our strategy to implement an AMC realization that reflects realistic performance was to adopt a well-established technology deployed in the field with the backing of simulation tools that are detailed and realistic to the level of qualifying to support the standardization work associated with the deployed technology. Following this strategy, we choose the AMC subsystem from the LTE technology, [56], backed by the link level simulator developed by TU-Wien, [30]. We used in the simulation a 2x2 MIMO configuration. In the simulation, the spectrum band that is shared by the PN and the SN through our underlay DSA technique was implemented as a single LTE resource block corresponding to a bandwidth of 180 kHz. Note that references in this work to LTE technology and use of some of LTE subsystems/features (particularly the AMC subsystem and corresponding the shared spectrum band with an LTE resource block) are guided by the goal of implementing a realistic simulation. However, this should not be interpreted as indicative that either the PN or the SN are LTE networks, nor does it imply that the proposed solution is restricted to an LTE network, not even a cellular network architecture.

Following the implementation of the PN "playground" as a five-by-five grid, the total number of BSs in the PN is $N_{PBS} = 25$. However, of all N_{PBS} BSs, only N_P are active (have an established PN link involved in an active communication) at any time. Because all base stations use the same channel to communicate with their respectively assigned PU receivers the PN network load is, as previously indicated, N_P/N_{PBS}. The location of the N_P PU receivers is determined at random using a uniform distribution with the limitation that no base station could have more than one receiver assigned to it (therefore, this random placement of PU receivers also determines which BSs in the grid are active). Also, the receivers were connected to the base station from which they received the strongest signal. Transmit powers in the primary network were limited to the range between -20 and 40 dBm. The transmit power assignment for the ith. active primary link ($i = 1, 2, \ldots, N_P$) follows the same algorithm as in (2). Each primary transmission considers the other network transmissions as out of network interference akin to background noise.

The considered SN operating through underlay DSA consisted of $N_S = 4$ transmit-receive pairs of CRs, with the transmitters placed at random (also with a uniform distribution) on the PN playground. The SN receivers were placed at random (also with a uniform distribution) around their respectively assigned transmitter within a distance not exceeding 50 m. In the simulation setup we intended to reflect a situation where the PN had somewhat more capabilities (achieving larger throughput and communication range) than the SN because of being the incumbent to the spectrum band under consideration. Therefore, we assumed that the SUs were smaller devices with transmit power in the range of -30 to 20 dBm. Twenty equally spaced power levels in this range are considered as the set of allowed settings for transmission. Transmissions in both networks make use of AMC. It is also assumed that the SN has knowledge of the underlying timing operation in the primary network to the extent of allowing CRs to sense and transmit at appropriate times.

In the system, channels gains are assumed to follow a quasi-static (block) fading model. Moreover, all links assumed a path loss model for urban area given by $L = 128.1 + 37.6 \log d + 10 + S$ (in dBs), where d is the distance between transmitter and receiver in km, S is the shadowing loss (modeled as a zero-mean Gaussian random variable with 6 dB standard deviation) and the penetration loss is fixed at 10 dB, [57]. Note that it is the shadowing loss the one responsible for the possibility that a receiver may receive with larger power a transmission originated further away than a second transmission and, because of this, the reason why we use the term of "nearest" transmitter to indicate, in fact, the one received with largest power. Our assumed small scale fading follows a Pedestrian B model from [58]. This is a model that reflects time-varying channel conditions corresponding to the velocities of a person walking. Assuming 1.5 m/s as a typical pedestrian velocity (close to the 5.5 km/h usual for a person's walk), the channel coherence time is 100 ms for a carrier frequency of 2 GHz. Noting that all transmissions adopt AMC to adapt their transmission to the quality of their respective link, our assumption of a block fading channel holds with ample margin for an AMC control loop operating with rates around 1 kHz as is commonplace to see nowadays (e.g., LTE). Moreover, the coherence time of 100 ms provides ample margin of time to complete the execution of one decision-making cycle (power allocation) by the cognitive engine. In fact, since one decision-making cycle can be completed in approximately 20 ms, our assumptions hold for velocities of up to almost 30 km/h, and up to approximately 90 km/h for a scheme to be discussed in the next Section that reduces three times the number of probe messages. The noise power level was set at -130 dBm.

To generate training data, we configured the power and AMC control in the SUs so they maintained the ability to use the estimated modulation order of the "nearest" primary link but without the cognitive engine shown in Fig. 5. Instead, they implemented a distributed power control algorithm modified to incorporate the modulation order of the "nearest" primary link. A number of algorithms had been proposed for distributed power control in ad-hoc wireless networks. One of the first ones, and the precursor to many related variants, is the Foschini-Miljanic algorithm, [59], which implements an iterative distributed power control process so as to meet a target SINR. For the task at hand, we adopted this iterative power allocation algorithm for the alternative SN that generated the data set to train the NARX neural network cognitive engine at each SU. Specifically, power is calculated for a secondary link i at each iteration m using the update formula,

$$P_i^s[m+1] = \left(\frac{\beta_i}{\gamma_i^s[m]}\right) P_i^s[m], \tag{11}$$

where P_i^s is the transmit power, β_i is the target SINR and $\gamma_i^s[m]$ is the actual SINR measured in the mth. iteration which can be calculated through (7). The fact that this algorithm associates power control with a target SINR in our case is a useful feature because the target SINR, when met, also determines the modulation order to be used as follows [26],

$$T_i^{(s)} = \log_2(1 + k\,\gamma_i^{(s)}), \quad i = 1, 2, \ldots, N_S. \tag{12}$$

Consequently, we can think that instead of having a set of possible [modulation, channel code] pairs, now we have a set of target SINRs to choose from. Let $\mathfrak{B} = \{b_1, b_2, \ldots, b_K\}$ be this set, where target SINRs b_i's are assumed to be sorted in ascending order. Of course, reducing the target SINR will result in decreasing the transmit power. As a consequence, the algorithm provides the mechanisms to both adapt transmit power and AMC settings in a distributed way. Moreover, for fair comparison to the system with the proposed NARX neural network cognitive engine, and to maintain the same operating principles, the transmit power from SUs also needs to be constrained by the goal to not degrade the SINR of the "nearest" primary network link to the extent of reducing the modulation order (not having the NARX neural network, this SN used to generate training data cannot operate based on the use of throughput inferred for the "nearest" primary link and can only make use of the modulation order estimated from the modulation classification process). Modifying the Foschini-Miljanic algorithm by reducing the target SINR allows to manage this constraint by resulting in a reduction in the SU transmit power. As such, we adopted for the control of CR transmissions in the SN used for training data generation this modified version of the Foschini-Miljanic algorithm, where the SU target SINR is progressively reduced until there is no change in the modulation order of the "nearest" primary link. We note here that while it is certainly possible to use one of the many existing enhancements to the Foschini-Miljanic algorithm, we chose to use the original version without improvements because its provides a baseline performance measure and because this power control algorithm or a variation of it are not the contribution of our work. Of course, the PUs in this system maintained the power allocation algorithm proposed in [26], as explained in Section II, which provided the training data target values of throughput in each PN link.

One additional detail considered for the SN used to generate training data provided for a different operation in the case when a "nearest" primary link is already transmitting using the smallest modulation order when there is no transmissions from the SN. Note that in AMC the modulation order transmitting the smallest number of bits per symbol (type 0) is used for the smallest operating SINRs (because it is more resilient to interference and noise). When the interference from the SN increases, the primary links that were already using the smallest possible modulation orders when there were no secondary transmissions will not switch to other modulation orders because there is simply no other modulation order with fewer bits per symbols to switch to. This means that transmissions from an SU that otherwise would generate a change in modulation order would not result in any change when the "nearest" primary link is already transmitting with the modulation order with smallest bits per symbol. Moreover, primary links using this modulation order, do so because their SINR is at the lower range of the operating SINRs, which implies that they are at a link state that likely may not leave much room for added interference from SUs. Because of these reasons, and in the interest of prioritizing the protection of primary links against excessive SN interference, we configured the alternative SN so that a CR will not transmit if it senses that its "nearest" primary link is using the lowest modulation order when the SN is not transmitting (as explained in Section IV, our proposed technique implements a mechanisms with the same spirit but based on checking for the smallest CQI at the "nearest" primary link when the SN is not transmitting, instead of smallest modulation order).

Collected using the simulator described above, data sets used for training consisted of 10000 data samples for each of the primary network load values of 0.16, 0.32, 0.48 and 0.64. Each of the 10000 data samples in a data set constitutes a completely different randomly-determined realization of the receiving PU location, active BSs in the PN, SUs location and channel gain values, leading to data sets that encode a very vast array of possible practical spectrum sharing scenarios between the PN and the SN. To manage (prevent) overfitting of data during training, a subset of the data set (7000 samples or 70% of the data set) was used to train the neural network and the rest for validation (10%) and test (20%) in order to present the CRs with new, never-seen-before scenarios.

From the training process itself we determined the values of the proposed NARX neural network weights and biases. The initial values for weights and biases were selected randomly and then were updated according to the Levenberg-Marquardt optimization method. This method involves a back-propagation algorithm to compute the gradients of the prediction error corresponding to the artificial neurons [60]. It is known that in the case of function approximation problems, for the neural networks containing up to a few hundred weights, the Levenberg-Marquardt algorithm will have the fastest convergence [50]. Since the main task here is to approximate the nonlinear function $f(.)$ which maps the input variables to the continuous output variable T, we frame this task as a regression predictive modeling. For regression tasks, the mean squared error (MSE) is the most widely used metric to measure the prediction error, [61]. Therefore, MSE was chosen as the prediction error metric:

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(e_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(T_i - \hat{T}_i)^2, \tag{13}$$

where N is the size of training data set, T_i and $\hat{T}_i$ are target and predicted values, respectively. The validation set was used to fine-tune the model hyperparameters (e.g., hidden layer size, number of tap delays, etc.), and the test set was

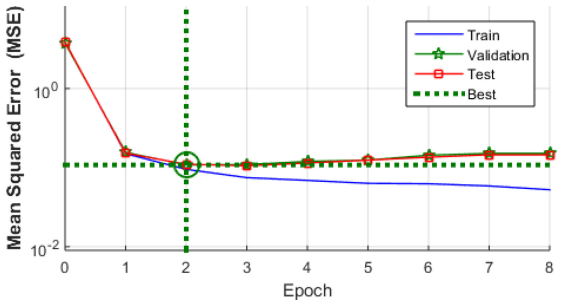

FIGURE 8. Throughput prediction performance of the NARX neural network. Best validation performance is 0.10825 at epoch 2.

used to compare the prediction performed by the trained neural network with the actual expected performance in order to encounter new, never-seen-before scenarios.

As part of the validation process, and in order to find the number of hidden neurons and depth of the tap-delay lines in the NARX neural network, we also performed a thorough set of experiments with different configurations. In [48] the authors investigated the relation between the order of the memory embedded in the NARX neural network and the performance when learning temporal dependencies in time series. They examined the performance of NARX networks with different memory orders (from 1 to 6) across different order of time dependency in signals, and demonstrated that NARX neural networks showed significant improvement when learning long-term dependencies as the order of the embedded memories is increased. Following the results in [48], we tried different delay (memory order) selected from the set $\{1, 2, 3, \ldots, 10\}$, and for each selection of delay we changed number of neurons in the hidden layer from 5 to 100 in increments of 5 while observing the loss function on the validation data set.

Fig. 8 illustrates the performance of the NARX neural network during training, validation and testing for a primary network load equal to 0.64. As is to expect, Fig. 8 shows the training error continuing to decrease with more training epochs, but the validation and test errors starting to increase after reaching a minimum with two training epochs. Therefore, two epochs marks the inflection point in the bias vs. variance tradeoff when the training process achieves the best MSE during validation and the training regime is starting to transition to overfitting. At this inflection point, marked with a vertical green dashed line in Fig. 8, the training should be stopped. Fig. 8 shows the MSE for the best performing structure that resulted in the least training error at two training epochs, which was found to be with $n_h = 50$ hidden neurons and $d_y = 7$ time delay steps.

For $n_h = 50$ hidden neurons and $d_y = 7$ time delay steps configuration, and using the formulas in Table 1, we can see that the execution of one decision-making cycle (power allocation) by the cognitive engine entails 2550 additions, 2550 multiplications and 50 table lookup operations. However, as will be seen in the next Section, we will consider an implementation of the proposed underlay DSA where the number of probe messages is reduced three-fold, in which case the number of computing operations is reduced to 1250 additions, 1250 multiplications and 50 table lookup operations. In all cases, with current existing technology these operations can be computed within a very small fraction of the time required to complete one decision-making cycle.

Going back to Fig. 2 helps to gain an intuition into the NARX neural network cognitive engine operation. The NARX neural network receives as one input the possible SU's transmit power levels organized in sequence and, as another input, the corresponding modulation order sensed from the "nearest" primary link. During training, the NARX neural network learns to predict the throughput values at the "nearest" primary link that correspond to the two input sequences. Considering (6), we can think that the sequence of transmit power values that is input to the NARX neural network will yield a range of interference values, which will correspond to a "segment" of SINR values in the abscissa of Fig. 2. The position of this segment within the range of SINR values depends on the many factors reflected in (6) (e.g., primary channel gain, interference from other SUs, etc.) but the NARX neural network has a sense of where the segment is thanks to the reference provided by the sequence of modulation orders at the input (e.g., if for the setup in Fig. 2, the sequence of modulation schemes are QPSK and 16QAM, the SINR segment is around 10 dB). During training, the NARX neural network is presented with multiple different such segments from different wireless environment scenarios, eventually learning the throughput vs. SINR AMC performance curve. During operation of the NARX neural network (the testing phase in machine learning terms), sensing the sequence of modulation schemes that results from the sequence of probe messages with different power settings allows the NARX neural network to localize the segment of SINR values for the "nearest" primary link (in effect, finding the network scenario presented during training that best matches the existing wireless environment) and, consequently, predicts the corresponding throughput from the AMC performance curve.

V. SIMULATION RESULTS

The performance of the presented technique was evaluated through 150 Monte Carlo simulation runs. These simulations were run on the same simulator described in Section IV-B, now with the addition of our proposed underlay DSA scheme. As such all the setup and models described in Section IV-B applied with no changes to performance evaluation runs.

Figs. 9–12 study the throughput performance in the primary and secondary networks for our presented technique and contrast them against other schemes. Results are shown for the four different primary network loads equal to 0.16, 0.32, 0.48, and 0.64. As indicated in Section IV, for the presented technique we considered two approaches for the SUs to choose transmit settings. The first approach, labeled

in the figures as '*PN+SN- NN Cog. Eng.- Modulation*', features our proposed cognitive engine with the capability for full AMC mode estimation at the "nearest" primary link, but it makes a limited use of this capability by making the SU choose the maximum transmit power value that is estimated to keep unchanged the modulation order (but not necessarily the channel coding rate) at its "nearest" primary link. The second approach makes full use of the cognitive engine's throughput (full AMC mode) estimation capabilities at the "nearest" primary link, by making the SU choose the maximum transmit power value that is estimated to not change the "nearest" primary link throughput beyond a maximum relative change value. This second approach is itself divided into a case where one probe message is sent for each possible power setting (for a total of twenty probe messages) and a second case that explores a reduction in the overhead from transmitting probe messages which transmits just seven messages and uses interpolation to complete the information for the rest of available transmit power settings. To show how our technique is able to leverage the estimation of the full AMC mode and provide each SU with a fine control over the interference it imposes to the "nearest" primary transmission link, we obtained results for three different limits on relative average throughput change in the PN: 2%, 5% and 10%. In the figures, we labeled the results when sending all probe messages as '*PN+SN- NN Cog. Eng- 2%- All probe msgs.*' for a limit on relative average throughput change in the PN of 2%, '*PN+SN- NN Cog. Eng- 5%- All probe msgs.*' for a limit on relative average throughput change in the PN of 5%, and '*PN+SN- NN Cog. Eng- 10%- All probe msgs.*' for a limit on relative average throughput change in the PN of 10%. Similarly, for the case when sending seven probe messages, the labels for 2%, 5% and 10% limit on relative average throughput change in the PN are respectively '*PN+SN- NN Cog. Eng- 2%- Seven probe msgs.*', '*PN+SN- NN Cog. Eng- 5%- Seven probe msgs.*' and '*PN+SN- NN Cog. Eng- 10%- Seven probe msgs.*'.

The performance of these instances of our proposed underlay DSA technique is compared in the figures against other schemes. The first such contrasting benchmark scheme, labeled in the figures as '*PN+SN- Adapted Foschini-Miljanic*', is the same system used to collect training data and described in Section IV-B. Recall that this scheme lacks the NARX neural network cognitive engine's capability to predict the full AMC mode at the "nearest" primary link. Also note that this benchmark SN constitutes an implementation of the central principles of [21] while also managing practical considerations not addressed therein (e.g., multiple links in the SN and PN, distributed, ad-hoc operation of the SN, etc.), and, as such, serves the purpose of providing an indication of the performance improvements of our proposed technique versus prior works. As additional benchmarks, we also considered two schemes that select transmit power for the SUs based on an exhaustive search across all possible setting permutations. In contradiction with our goal to avoid any exchange of information between the primary and secondary networks, these two schemes also incorporate the ability to perfectly know the CQI on the primary links as if the SN had access to the control feedback channels in the PN. The first exhaustive search scheme, labeled '*PN+SN- Exh. search-Max PU throughput*', finds across all possible SUs transmit power permutations, the setting that results in no change in modulation order at any primary link and maximum average throughput in the PN. The second exhaustive search scheme, labeled '*PN+SN- Exh. search-Min PU throughput*', favors the average throughput at the SN by finding across all possible SUs transmit power permutations, the setting that results in no change in modulation order at any primary link and minimum average throughput in the PN. Clearly, the two exhaustive search curves present extreme performance results based on ideal setups. Finally, Fig. 9 includes a curve, '*PN without SN*', which shows the average throughput achieved by the PN when the SN is not present.

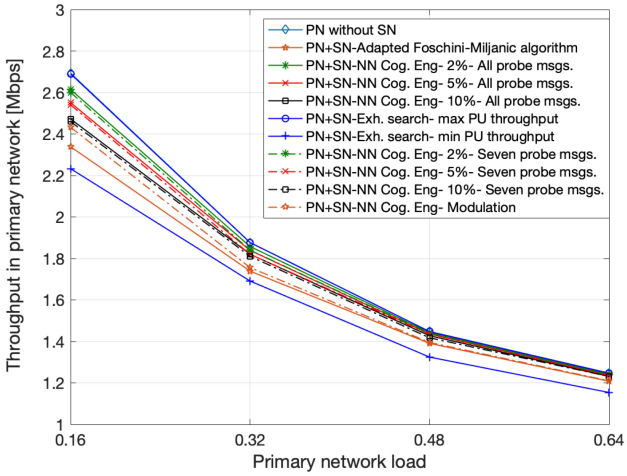

FIGURE 9. Average throughput in the primary network.

Fig. 9 shows the average throughput achieved for the PN as a function of the PN load, N_P/N_{PBS}, while Fig. 10 shows the change in this throughput relative to the 'PN without SN' case. It can be seen in both figures that the use of the proposed NARX neural network cognitive engine results in SUs transmit settings that reduces the average throughput in the PN much less than the case when the modified Foschini-Miljanic algorithm is used in the SU. Moreover, in the figures, the 'PN+SN- Exh. search-Min PU throughput' curve illustrates the extent to which the average throughput in the PN can be affected without changing the modulation order at the "nearest" primary links (as much as 17%). This indicates that considering only the modulation order provides an initial means for implementing a fully autonomous underlay DSA but with the limitations associated with the coarse indication of the primary links SINR given only by the modulation order. The 'PN+SN- Adapted Foschini-Miljanic' and the 'PN+SN- NN Cog. Eng.- Modulation' schemes, which both suffer from the

limitation of relying on considering modulation order only, show better performance because the SU transmit power is chosen in a more conservative way in terms of reducing effects to the primary network, instead of conducting an exhaustive search for the setting that results in minimum average throughput in the PU. Nevertheless, because of relying on modulation order inference only, the 'PN+SN-Adapted Foschini-Miljanic' and the 'PN+SN- NN Cog. Eng.- Modulation' schemes result in relative reduction in the PN average throughput by as much as 13.5% and 9.5%, respectively. The best performance in terms of controlling the effect of SN transmissions on the PN is achieved with our proposed scheme using the inference of the primary links' full AMC mode (in actuality, the throughput) provided by the NARX neural network cognitive engine. This is seen through the results obtained for the two cases (transmitting either all or seven probe messages) designed on the premise of limiting the maximum relative change in average throughput at the "nearest" primary link, for which we show results for 2%, 5% and 10% relative change limit. Moreover, these schemes include the means to control as desired the level of SN effect on the PN (by setting the limit maximum relative change). In fact, the 'PN+SN- NN Cog. Eng- 2%-All probe messages' along with 'PN+SN- NN Cog. Eng-2%- Seven probe messages' curves exemplify the very fine level of control that is possible to achieve with the proposed approach.

Fig. 10 shows that the very fine level of control seen with our proposed scheme is achieved at all primary network loads, except at the lowest value of 0.16, when the relative change in average PN throughput exceeds the 2% limit by only 1% when sending all probe messages and by 1.5% when sending seven probe messages. For the rest of the cases, only in the case of 5% limit, sending seven probe messages, and at the lowest primary network load, the relative change in PN average throughput is exceeded by just 0.5%. These differences are attributed to errors in estimating the throughput at the PN, which are discussed more in detail later in this Section. Also, the differences are only seen at the smallest PN network load of 0.16 because of the larger sensitivity of the relative change in PN average throughput with lower PN load as was previously highlighted for Fig. 3. Fig. 10 also shows that the scheme where a fraction of probe messages is used yields significant reduction in the transmission overhead of probe messages (roughly a threefold reduction) without much sacrifice in performance (maximum 3.5% instead of 2% actual achieved relative change in PN throughput compared to 3% maximum change with all the probe messages, and maximum 5.5% instead of 5% actual achieved relative change in PN throughput compared to 5% maximum change with all the probe messages). Finally, the curve 'PN+SN- Exh. search-Max PU throughput' coincides with the 'PN without SN' curve as the exhaustive search solution that maximizes average PN throughput is essentially the one with no transmissions in the SN (with one caveat to be discussed in Fig. 12).

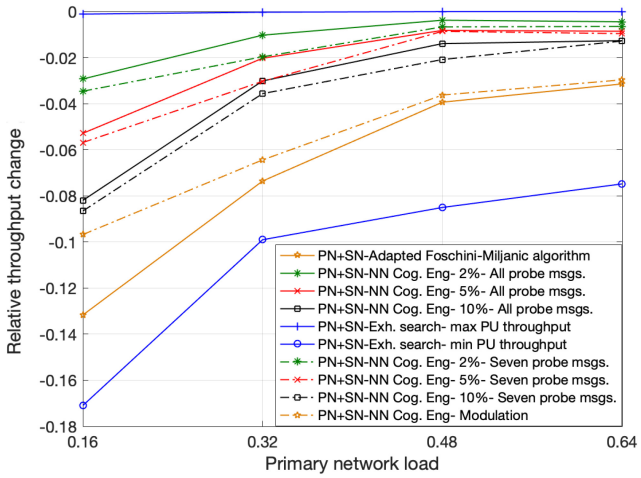

FIGURE 10. Relative throughput change in the primary network.

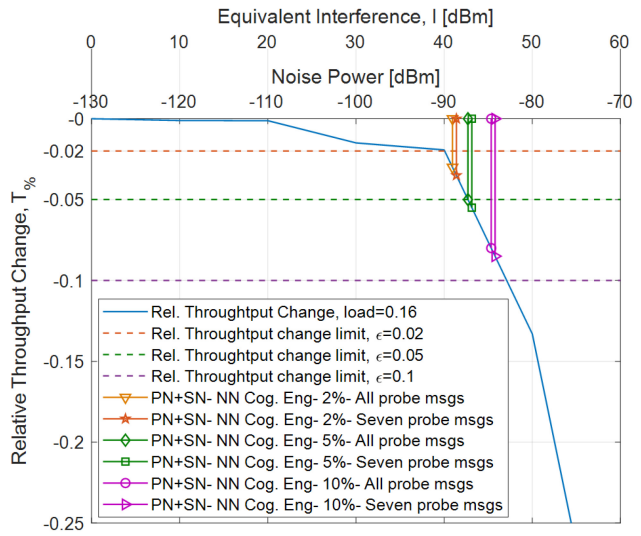

FIGURE 11. Analysis of equivalent interference generated by the SN on the PN.

Figure 11 presents results from Fig. 10 integrated into the bottom plot of Fig. 3. To avoid a cluttered plot, we restrict Fig. 11 to the case of PN load equal to 0.16 because, as we have seen, this is the only situation where the equivalent interference generated by the SN results in relative average throughput change at the PN ($T_{\%}$) that slightly exceeds the preset limit in a few cases. Same as in Fig. 3, Fig. 11 shows $T_{\%}$ as a function of increasing background noise power (when there is no active SN), or as function of the equivalent interference generated by the SN. Overlaid to these results, Fig. 11 shows three horizontal dashed lines, each representing the three limits on $T_{\%}$: $\epsilon = -0.02$, $\epsilon = -0.05$, and $\epsilon = -0.1$. The intersections of these lines with the curve for $T_{\%}$ indicate the ideal operating point for each setting for ϵ. The abscissa of these intersection points indicate the maximum equivalent interference $I_0(\epsilon)$ that has been modeled earlier as per (5). As can be seen, $I_0(-0.02) = 40$ dBm, $I_0(-0.05) = 42.7$ dBm, and $I_0(-0.02) = 47.1$ dBm. This

small range of values for $I_0(\epsilon)$ underscore the need for a fine control on the equivalent interference as provided by the proposed technique. Moreover, these results reveal an important consideration when choosing ϵ. Besides the obvious that larger ϵ implies a larger effect of the SN on the PN, Fig. 11 shows that a small value of ϵ (e.g., $\epsilon = -0.02$) corresponds to an operating point at the edge of low sensitivity to deviations in the equivalent interference, where a small excess in equivalent interference results in an small extra relative average throughput change in the PN. In contrast, larger value of ϵ (e.g., $\epsilon = -0.10$) correspond to an operating point of high sensitivity to deviations in the equivalent interference, where a small excess in equivalent interference results in a large extra relative average throughput change in the PN. Because of this, when operating with larger values of ϵ (if this is somehow acceptable for the PN) it is critical for the equivalent interference generated by the SN to be less than or at most equal to $I_0(\epsilon)$ (while there is more flexibility in this regard when operating with small values of ϵ).

Moreover, Fig. 11 depicts with vertical lines the operating points attained by the process followed by the cognitive engines in the SUs. As can be seen, the proposed underlay DSA technique manages to set transmit powers at the SN that results in equivalent interference close to the maximum $I_0(\epsilon)$. In the cases that have been noted when the limit is slightly exceeded ($\epsilon = -0.02$ and $\epsilon = -0.05$ for the case of seven probe messages), the equivalent interference is exceeded by a very small amount (1.4 dBm at then most), underscoring the very fine control allowed by the proposed technique. In the cases of the large $\epsilon = -0.1$, the proposed technique follows the needed conservative approach explain in the previous paragraph and sets transmit power levels that yield equivalent interference values below the maximum $I_0(\epsilon)$. Overall, the vertical lines shown in Fig. 11 illustrate the success of the proposed technique in addressing the two challenges associated with underlay DSA: identifying the interference limit for the PN and allowing the SN to operate with equivalent interference levels generated on the PN close to this limit (without exceeding it) by enabling the SU to become aware of the interference they generate. Finally, it is worth noting that the operation of the SN at equivalent interference limits close to the maximum $I_0(\epsilon)$ indicates that our proposed algorithm is able to find transmit settings for the SUs that will result in as large SN throughput as could be allowed by the PN interference limit.

Fig. 12 shows the average throughput achieved in the SN as a function of the PN load. Naturally, the more a scheme affects the PN throughput, the larger the SN throughput it could achieve. As such, it can be seen that our approach based on a limit maximum PN relative throughout change not only provides the means to control how much the PN is affected by the SN, but also it allows to control how large the average throughput at the SN is desired to be (at the expense of the PN). Even so, the realization with the more restrictive setting for the SN (the one with a maximum PN relative throughput change of 2%) still achieves useful

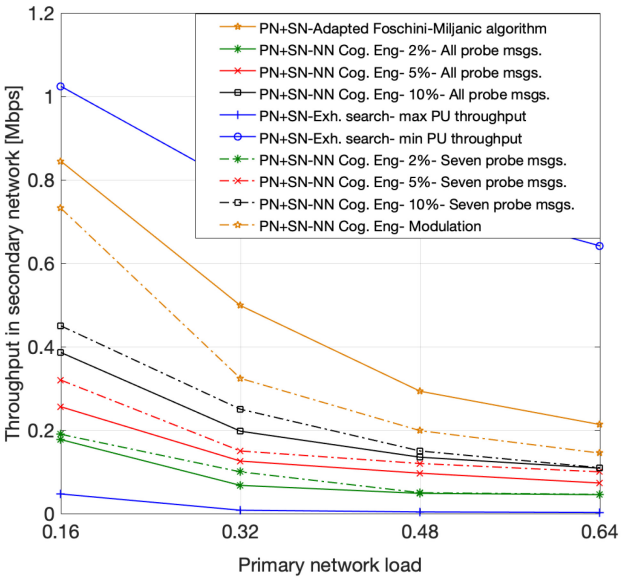

FIGURE 12. Average throughput in the secondary network.

average throughput values between 180 and 50 kbps for a channel with 180 kHz bandwidth (in case of transmitting seven probe messages the average throughput at the SN is slightly larger which is consistent with the results in Fig. 10). Also, note that at low PN loads, the 'PN+SN- Exh. search- Max PU throughput' system shows throughput values that imply transmission in the SN. This does not contradict our earlier statement that this result essentially coincides with the 'PN without SN' case. Instead, the transmissions in the SN that are seen in this case correspond to infrequent setups where the SUs are located so far away from the few active primary links (consider that this effect occurs only at very low PN loads) that they can transmit with very low power with no practical effect on the PN.

Fig. 13 depicts the cumulative distribution function (CDF) of the throughput in the SN for different primary network loads. This figure presents a perspective that explains an added advantage of the NARX neural network solution compared to the modified Foschini-Miljanic algorithm-based solution. The figure shows that in the case of the SN that uses the modified Foschini-Miljanic algorithm, around 15% of the time the SUs will be unable to transmit (throughput is zero) when the PN load equals 0.16 and this number increases to around 30% as the PN load increases. This is because the SN that uses the Foschini-Miljanic algorithm is only able to infer the modulation scheme used in the primary link and not the channel coding rate, which leads to SUs not being able to have a finer assessment of their effect on the PN when the "nearest" primary link is using a modulation '*type 0*'. As a result, and as discussed earlier, in order to protect the PN, those SUs using the modified Foschini-Miljanic scheme for which the "nearest" primary link use modulation '*type 0*' are blocked from transmitting. In contrast, the proposed technique using NARX neural network,

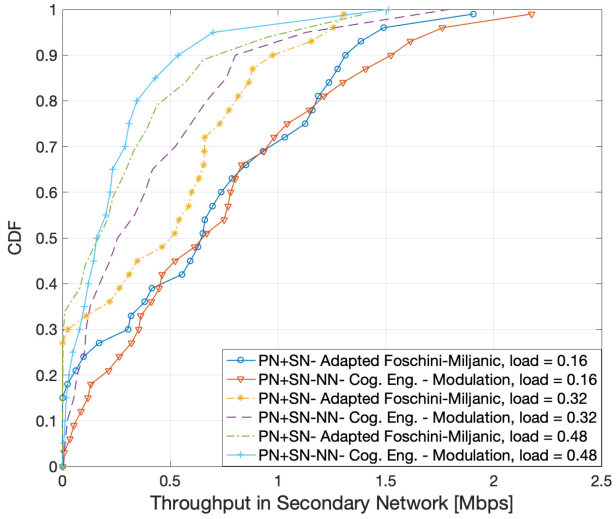

FIGURE 13. Cumulative distribution function (CDF) of the throughput in the secondary network.

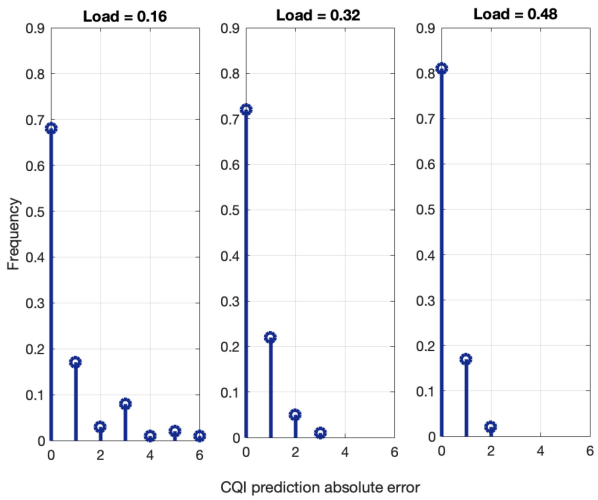

FIGURE 14. CQI estimation performance of the NARX neural network.

'*PN+SN-NN Cog.Eng.- Modulation*', is able to estimate the finer AMC configuration of coding rate setting, making this protection and the blocking of SUs unnecessary (except when the "nearest" link is using the AMC mode for a lowest rate, which corresponds to CQI=1). Consequently, as seen in Fig. 13, the proposed technique increases the transmission opportunities in the SN by the same percentage of time that the SUs are blocked in the case of using the modified Foschini-Miljanic algorithm-based solution.

As just seen, a key advantage of the proposed technique follows from the remarkable ability of the NARX neural network in the SU cognitive engine to estimate the channel coding rate used in a primary link. Therefore, we evaluated the accuracy performance of the NARX neural network in estimating the CQI in a primary link (which is equivalent to the full AMC mode consisting of modulation order and channel coding rate). Fig. 14 shows as a function of the primary network load the relative frequency (i.e., the empirical probability distribution) of the absolute error when predicting the CQI for the case of transmitting all probe messages. In this case, errors in the prediction are quantified by measuring the absolute value of the difference between the actual and the predicted CQI. This figure shows that as the network load increases, the probability of an accurate estimation (prediction absolute error equal to zero) increases and reaches more than 80% for a load equal to 0.48. The Figure shows that, overall, the probability of significant errors when predicting CQI is quite small but, nevertheless, we speculate this to be a factor in the (still small) reduction in PN average throughput and in the small difference at low PN loads between the target maximum relative change in average PN throughput and the actual achieved relative change in PN throughput for our schemes based on target maximum PN relative throughput change.

VI. CONCLUSION

In this paper we have presented a fully autonomous and distributed underlay DSA technique capable of addressing the two main challenges of how to establish an interference limit from the PN and how CRs in the SN become aware of the interference they create on the PN. The presented technique is based on a NARX neural network cognitive engine that uses a sequence of probe messages and the sensed modulation order in the primary link "nearest" to an SU to predict the effect of its transmission on that "nearest" PN link. It does this by predicting the throughput or, equivalently, both the channel coding rate and the modulation order, at the "nearest" primary link. Based on this, in the presented technique the SUs choose the maximum transmit power that is estimated to not change their respective "nearest" primary link throughput beyond a chosen maximum relative change value. Simulation results show that the proposed technique is able to accurately predict both the channel coding rate used and the modulation scheme in a primary link without the need to exchange information between the PN and the SN, and that the proposed technique succeeds in its main goal of determining the transmit power of the SUs such that their created interference remains below the maximum threshold that the primary network can sustain (with minimal effect on the average throughput). Also, our proposed algorithm is able to find transmit settings for the SUs that will result in as large throughput in the SN as could be allowed by the PN interference limit. Specifically, for a target PN maximum relative average throughput change of 2%, the proposed scheme is able to maintain the PN relative throughput change less than 3% when sending all probe messages, and less than 3.5% when reducing three times the number of transmitted probe messages, while simultaneously achieving useful average throughput values in the SN between 180 and 50 kbps for a channel with 180 kHz bandwidth. For future work we plan to investigate approaches to further reduce the number of probe messages.

REFERENCES

[1] "Spectrum policy task force," Federal Commun. Comm., Washington, DC, USA, Rep. ET Docket no. 02-135, 2002.

[2] S. Haykin, "Cognitive radio: Brain-empowered wireless communications," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 2, pp. 201–220, Feb. 2005.

[3] M. Song, C. Xin, Y. Zhao, and X. Cheng, "Dynamic spectrum access: From cognitive radio to network radio," *IEEE Wireless Commun.*, vol. 19, no. 1, pp. 23–29, Feb. 2012.

[4] A. Goldsmith, S. A. Jafar, I. Maric, and S. Srinivasa, "Breaking spectrum gridlock with cognitive radios: An information theoretic perspective," *Proc. IEEE*, vol. 97, no. 5, pp. 894–914, May 2009.

[5] E. Hossain, D. Niyato, and Z. Han, *Dynamic Spectrum Access and Management in Cognitive Radio Networks*. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[6] W. Wang and A. Kwasinski, "Feedback-based cooperative primary channel activity estimation for dynamic spectrum access," in *Proc. 3rd Int. Symp. Appl. Sci. Biomed. Commun. Technol. (ISABEL)*, 2010, pp. 1–5.

[7] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *IEEE Commun. Surveys Tuts.*, vol. 11, no. 1, pp. 116–130, 1st Quart., 2009.

[8] S. Benazzouza, M. Ridouani, F. Salahdine, and A. Hayar, "A survey on compressive spectrum sensing for cognitive radio networks," in *Proc. IEEE Int. Smart Cities Conf. (ISC2)*, 2019, pp. 535–541.

[9] X. Liu, M. Jia, X. Zhang, and W. Lu, "A novel multichannel Internet of Things based on dynamic spectrum sharing in 5G communication," *IEEE Internet Things J.*, vol. 6, no. 4, pp. 5962–5970, Aug. 2019.

[10] H. Hu and Q. Zhu, "Dynamic spectrum access in underlay cognitive radio system with SINR constraints," in *Proc. 5th Int. Conf. Wireless Commun. Netw. Mobile Comput.*, 2009, pp. 1–4.

[11] X. Kang, R. Zhang, Y.-C. Liang, and H. K. Garg, "Optimal power allocation strategies for fading cognitive radio channels with primary user outage constraint," *IEEE J. Sel. Areas Commun.*, vol. 29, no. 2, pp. 374–383, Feb. 2011.

[12] L. Musavian and S. Aïssa, "Capacity and power allocation for spectrum-sharing communications in fading channels," *IEEE Trans. Wireless Commun.*, vol. 8, no. 1, pp. 148–156, Jan. 2009.

[13] A. Ghasemi and E. S. Sousa, "Capacity of fading channels under spectrum-sharing constraints," in *Proc. IEEE Int. Conf. Commun.*, vol. 10, Jun. 2006, pp. 4373–4378.

[14] Y. Yang, H. Ma, and S. Aissa, "Cross-layer combining of adaptive modulation and truncated ARQ under cognitive radio resource requirements," *IEEE Trans. Veh. Technol.*, vol. 61, no. 9, pp. 4020–4030, Nov. 2012.

[15] K. Eswaran, M. Gastpar, and K. Ramchandran, "Bits through ARQs: Spectrum sharing with a primary packet system," in *Proc. IEEE Int. Symp. Inf. Theory*, Jun. 2007, pp. 2171–2175.

[16] J. C. F. Li, W. Zhang, A. Nosratinia, and J. Yuan, "SHARP: Spectrum harvesting with ARQ retransmission and probing in cognitive radio," *IEEE Trans. Commun.*, vol. 61, no. 3, pp. 951–960, Mar. 2013.

[17] Z. Rezki and M.-S. Alouini, "Ergodic capacity of cognitive radio under imperfect channel-state information," *IEEE Trans. Veh. Technol.*, vol. 61, no. 5, pp. 2108–2119, Jun. 2012.

[18] E. Kayalvizhi and B. Gopalakrishnan, "Estimation of optimal channel gain in cognitive radio networks using bisectional algorithm," *Int. J. Adv. Netw. Appl.*, vol. 11, no. 1, pp. 4171–4176, 2019.

[19] *3GPP Technical Specification Group Radio Access Network Physical Layer Procedures (FDD) (Release 5), V5.11.0*, 3GPP Standard TS 25.214, 2005.

[20] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Higher-Speed Physical Layer Extension in the 2.4 GHz Band*, IEEE Standard 802.11b-1999 (R2003), 1999.

[21] R. Zhang, "On active learning and supervised transmission of spectrum sharing based cognitive radios by exploiting hidden primary radio feedback," *IEEE Trans. Commun.*, vol. 58, no. 10, pp. 2960–2970, Oct. 2010.

[22] L. Zhang, M. Xiao, G. Wu, G. Zhao, Y.-C. Liang, and S. Li, "Proactive cross-channel gain estimation for spectrum sharing in cognitive radio," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 10, pp. 2776–2790, Oct. 2016.

[23] C. Tarhini and T. Chahed, "On capacity of OFDMA-based IEEE802.16 WiMAX including adaptive modulation and coding (AMC) and inter-cell interference," in *Proc. 15th IEEE Workshop Local Metropolitan Area Netw. (LANMAN)*, 2007, pp. 139–144.

[24] J. Yang, N. Tin, and A. K. Khandani, "Adaptive modulation and coding in 3G wireless systems," in *Proc. 56th Veh. Technol. Conf.*, vol. 1, 2002, pp. 544–548.

[25] F. Zhou, Y. Wu, R. Q. Hu, Y. Wang, and K.-K. Wong, "Energy-efficient NOMA enabled heterogeneous cloud radio access networks," 2018. [Online]. Available: arXiv:1801.01996.

[26] X. Qiu and K. Chawla, "On the performance of adaptive modulation in cellular systems," *IEEE Trans. Commun.*, vol. 47, no. 6, pp. 884–895, Jun. 1999.

[27] C. M. Spooner and W. A. Gardner, "The cumulant theory of cyclostationary time-series. II. Development and applications," *IEEE Trans. Signal Process.*, vol. 42, no. 12, pp. 3409–3429, Dec. 1994.

[28] A. Fehske, J. Gaeddert, and J. H. Reed, "A new approach to signal classification using spectral correlation and neural networks," in *Proc. 1st IEEE Int. Symp. New Front. Dyn. Spectr. Access Netw. (DySPAN)*, Nov. 2005, pp. 144–150.

[29] H. Abuella and M. K. Ozdemir, "Automatic modulation classification based on kernel density estimation," *Can. J. Electr. Comput. Eng.*, vol. 39, no. 3, pp. 203–209, 2016.

[30] C. Mehlführer, M. Wrulich, J. C. Ikuno, D. Bosanska, and M. Rupp, "Simulating the long term evolution physical layer," in *Proc. 17th Eur. Signal Process. Conf.*, Aug. 2009, pp. 1471–1478.

[31] J. Cavers, "Variable-rate transmission for Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 20, no. 1, pp. 15–22, Feb. 1972.

[32] B. Vucetic, "An adaptive coding scheme for time-varying channels," *IEEE Trans. Commun.*, vol. 39, no. 5, pp. 653–663, May 1991.

[33] J. M. Cioffi, *Course Notes for Digital Communications: Signal Processing*, Stanford Bookstore Custom Publ., Stanford, CA, USA, 2007.

[34] T. R. Newman, B. A. Barker, A. M. Wyglinski, A. Agah, J. B. Evans, and G. J. Minden, "Cognitive engine implementation for wireless multicarrier transceivers," *Wireless Commun. Mobile Comput.*, vol. 7, no. 9, pp. 1129–1142, 2007.

[35] M. Nrgaard, O. E. Ravn, N. K. Poulsen, and L. K. Hansen, *Neural Networks for Modelling and Control of Dynamic Systems: A Practitioner's Handbook*, 1st ed. Secaucus, NJ, USA: Springer-Verlag, 2000.

[36] G. E. Box, G. M. Jenkins, G. C. Reinsel, and G. M. Ljung, *Time Series Analysis: Forecasting and Control*. Hoboken, NJ, USA: Wiley, 2015.

[37] W. Lyu, Z. Zhang, C. Jiao, K. Qin, and H. Zhang, "Performance evaluation of channel decoding with deep neural networks," 2017. [Online]. Available: arXiv:1711.00727.

[38] S. Navabi, C. Wang, O. Y. Bursalioglu, and H. Papadopoulos, "Predicting wireless channel features using neural networks," 2018. [Online]. Available: arXiv:1802.00107.

[39] T. Luo and S. G. Nagarajan, "Distributed anomaly detection using autoencoder neural networks in WSN for IoT," *Proc. IEEE Int. Conf. Commun.*, 2018, pp. 1–6.

[40] V. K. Tumuluru, P. Wang, and D. Niyato, "A neural network based spectrum prediction scheme for cognitive radio," in *Proc. IEEE Int. Conf. Commun.*, May 2010, pp. 1–5.

[41] Y. Yu, T. Wang, and S. C. Liew, "Deep-reinforcement learning multiple access for heterogeneous wireless networks," in *Proc. IEEE Int. Conf. Commun.*, 2018, pp. 1–7.

[42] A. Sorjamaa, J. Hao, N. Reyhani, Y. Ji, and A. Lendasse, "Methodology for long-term prediction of time series," *Neurocomputing*, vol. 70, nos. 16–18, pp. 2861–2869, 2007.

[43] S. Haykin and X. B. Li, "Detection of signals in chaos," *Proc. IEEE*, vol. 83, no. 1, pp. 95–122, Jan. 1995.

[44] J. C. Principe, N. R. Euliano, and W. C. Lefebvre, *Neural and Adaptive Systems: Fundamentals Through Simulations*, vol. 672. New York, NY, USA: Wiley, 2000,

[45] J. L. Elman, "Finding structure in time," *Cogn. Sci.*, vol. 14, no. 2, pp. 179–211, 1990.

[46] B. A. Pearlmutter, "Gradient calculations for dynamic recurrent neural networks: A survey," *IEEE Trans. Neural Netw.*, vol. 6, no. 5, pp. 1212–1228, Sep. 1995.

[47] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. Neural Netw.*, vol. 5, no. 2, pp. 157–166, Mar. 1994.
[48] T. Lin, B. G. Horne, and C. L. Giles, "How embedded memory in recurrent neural network architectures helps learning long-term temporal dependencies," *Neural Netw.*, vol. 11, no. 5, pp. 861–868, 1998.
[49] T. Lin, B. G. Horne, P. Tino, and C. L. Giles, "Learning long-term dependencies in NARX recurrent neural networks," *IEEE Trans. Neural Netw.*, vol. 7, no. 6, pp. 1329–1338, Nov. 1996.
[50] E. Diaconescu, "The use of NARX neural networks to predict chaotic time series," *WSEAS Trans. Comput. Res.*, vol. 3, no. 3, pp. 182–191, 2008.
[51] J. M. P. Menezes and G. A. Barreto, "Long-term time series prediction with the narx network: An empirical evaluation," *Neurocomputing*, vol. 71, no. 16, pp. 3335–3343, 2008.
[52] H. Allende, C. Moraga, and R. Salas, "Artificial neural networks in time series forecasting: A comparative analysis," *Kybernetika*, vol. 38, no. 6, pp. 685–707, 2002.
[53] T.-N. Lin, C. L. Giles, B. G. Horne, and S.-Y. Kung, "A delay damage model selection algorithm for NARX neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2719–2730, Nov. 1997.
[54] I. J. Leontarits and S. A. Billings, "Input-output parametric models for non-linear systems—Part I: Deterministic non-linear systems," *Int. J. Control*, vol. 41, no. 2, pp. 303–328, 1985.
[55] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.
[56] "3GPP technical specification layer 1 documents TS 36.200 series (release 8)," 3GPP, Sophia Antipolis, France, Rep. TS 36.200, 2009.
[57] *LTE Physical Layer Framework for Performance Verification*, document TSG-RAN#148, R1-070674, 3GPP, Gothenburg, Sweden, 2007.
[58] "Guidelines for evaluation of radio transmission technologies for IMT-2000," Int. Telecommun. Union, Geneva, Switzerland, ITU-Recommendation M. 1225, 1997.
[59] G. J. Foschini and Z. Miljanic, "A simple distributed autonomous power control algorithm and its convergence," *IEEE Trans. Veh. Technol.*, vol. 42, no. 4, pp. 641–646, Nov. 1993.
[60] H. Mirzaee, "Long-term prediction of chaotic time series with multi-step prediction horizons by a neural network with Levenberg-Marquardt learning algorithm," *Chaos Solitons Fractals*, vol. 41, no. 4, pp. 1975–1979, 2009.
[61] G. S. Handelman *et al.*, "Peering into the black box of artificial intelligence: Evaluation metrics of machine learning methods," *Amer. J. Roentgenol.*, vol. 212, no. 1, pp. 38–43, 2019.

FATEMEH SHAH-MOHAMMADI (Member, IEEE) received the B.S. degree in electrical engineering in 2008, the M.S. degree in electrical and communication engineering Iran, in 2011, and the Ph.D. degree in engineering from the Rochester Institute of Technology, Rochester, NY, USA. She was an Assistant Professor for three years with the Electrical Engineering Department, Parsian Institute of Higher Education, Iran. She is currently with the Icahn School of Medicine at Mount Sinai. Her research interest includes cognitive radios, machine learning and data analytic applied to wireless communications, and biomedical data.

HATEM HUSSEIN ENAAMI (Member, IEEE) received the B.Sc. degree in electrical engineering from Libya, in 2002, and the M.S. degree in electrical and electronic engineering from the Rochester Institute of Technology, Rochester, NY, USA, in 2012, where he is currently pursuing the Ph.D. degree in engineering.

ANDRES KWASINSKI (Senior Member, IEEE) received the Diploma degree in electrical engineering from the Buenos Aires Institute of Technology, Buenos Aires, Argentina, in 1992, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland, College Park, Maryland, in 2000 and 2004, respectively. He is currently a Professor with the Department of Computer Engineering, Rochester Institute of Technology, Rochester, NY, USA. He has coauthored more than 80 publications in peer-reviewed journals and international conferences. He has also coauthored the books *Cooperative Communications and Networking* (Cambridge University Press, 2009) and *3D Visual Communications* (Wiley, 2013). His current areas of research include cognitive radios and wireless networks, cross-layer techniques in wireless communications, smart infrastructures and networking, and operation of the Internet-of-Things. He is currently an Area Editor for the *IEEE Signal Processing Magazine* and a Chief Editor of the IEEE Signal Processing Repository (SigPort). He has been an Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS and the IEEE WIRELESS COMMUNICATIONS LETTERS.