Unsupervised deep learning to solve power allocation problems in cognitive relay networks

Yacine Benatia, Anne Savard, Romain Negrel, Elena Veronica Belmega

Unsupervised deep learning to solve power allocation problems in cognitive relay networks

Yacine Benatia*†, Anne Savard†‡, Romain Negrel§, and E. Veronica Belmega*

*ETIS UMR 8051, CY Cergy Paris Université, ENSEA, CNRS, F-95000, Cergy, France
† IMT Nord Europe, Institut Mines Télécom, Centre for Digital Systems, F-59653 Villeneuve d'Ascq, France
‡Univ. Lille, CNRS, Centrale Lille, UPHF, UMR 8520 - IEMN, F-59000 Lille, France
§LIGM, Univ. Gustave Eiffel, CNRS, ESIEE Paris, Marne-la-Vallée, France
Email: {yacine.ben-atia, anne.savard}@imt-nord-europe.fr, romain.negrel@esiee.fr, belmega@ensea.fr

***Abstract*—In this paper, an unsupervised deep learning approach is proposed to solve the constrained and non-convex Shannon rate maximization problem in a relay-aided cognitive radio network. This network consists of a primary and a secondary user–destination pair and a secondary full-duplex relay performing Decode-and-Forward. The primary communication is protected by a Quality of Service (QoS) constraint in terms of tolerated Shannon rate degradation. The relaying operation leads to non-convex objective and primary QoS constraint, which makes *deep learning* approaches relevant and promising. For this, we propose a fully-connected neural network architecture coupled with a custom and communication-tailored loss function to be minimized during training in an unsupervised manner. A major interest of our approach is that the required training data contains only system parameters without the ground truth, i.e., the corresponding solutions to the non-convex optimization problem, as opposed to supervised approaches. Our numerical experiments show that our proposed approach has a high generalization capability on unseen data without overfitting. Also, the predicted solution performs close to the brute force one, highlighting the high potential of our unsupervised approach.**

***Index Terms*—Unsupervised deep learning, full-duplex relaying, Decode-and-Forward, cognitive radio**

I. Introduction

The ever increasing number of communicating devices and data-hungry applications are challenging the existing technologies and drive the transition to the next generation of wireless communications. Future communication systems target highly ambitious objectives [1] and among the promising candidate technologies envisioned to reach them are: cognitive radio, cooperative communications and full-duplexing [2].

On the one hand, cognitive radio and full-duplexing are promising technologies to enhance the network's spectral efficiency. In *cognitive radio*, a non-licensed network – the secondary or opportunistic network – is allowed to communicate in under-utilized licensed bands, provided that the impact on the licensed network – the primary network – is kept below acceptable levels [3]–[6]. *Full-duplex* nodes are capable to simultaneously transmit and receive information, having the potential to duplicate the spectral efficiency [2].

On the other hand, *cooperative communications* aim at improving the network capacity by taking advantage of the wireless medium, which makes signals sent by a source available at all receivers within range [7]. In this setting, some nodes called relays can enhance other nodes' communications by exploiting its non-intended received signals and, hence, increase the overall network capacity.

The simultaneous blend of all the above techniques holds the potential to improve both the spectral efficiency and network throughput while providing optimal resource allocation policies. In this paper, we consider a relay-aided cognitive radio network similarly to [8], where the opportunistic network is assisted by a full-duplex relay performing Decode-and-Forward (DF). Our objective is to find the optimal power allocation policy maximizing the opportunistic achievable Shannon rate under a primary quality of service (QoS) constraint by exploiting machine (deep) learning. The secondary transmitter and the relay have individual power constraints unlike in [8], where an overall power constraint was assumed.

Related works: Because of the non-linear and complex relay operations, the resulting power allocation problems in such relay-aided cognitive networks are non-convex and can be solved in closed-form only in special cases, such as: negligible interference links [5], negligible opportunistic direct links [9], Compress-and-Forward relaying [8]. Outside these very specific cases, such power allocation problems become difficult to tackle and even intractable using traditional techniques based on convex optimization and game theory.

More recently, machine learning techniques based on *deep learning* have been exploited to solve various resource allocation problems [10]–[16]. In [10], a convolutional neural network is trained to maximize the spectral and the energy efficiency of a wireless network composed of several interfering single antenna transceiver pairs under individual power constraints. The authors of [11] developed a deep neural network (DNN) approach for distributed antenna systems, which learns the nonlinear mapping between channel realizations and power allocation schemes of the traditional sub-gradient algorithm to maximize both the spectral and energy efficiency. In [12], power control policies based on DNNs in multi-user wireless networks were proposed. In [13], the authors maximized via deep learning the sum rate of a fading multi-user network under minimum rate constraints. In [14], the authors first solved a general convex functional optimization

This work has been supported IRCICA, CNRS USR 3380, Lille, France and by the ELIOT ANR-18-CE40-0030 and FAPESP 2018/12579-7 project.

problem with stochastic constraints via DNNs and then applied the general solution to maximize the rate of a point-to-point additive white Gaussian noise (AWGN) channel and of an interference channel. Unfortunately, the problem under study in this paper is not convex and the approach in [14] cannot be applied. The authors of [15] exploited DNNs based on autoencoders to solve the sum-rate maximization with respect to the sub-band power allocation in a downlink multi-cell network. In [16], the authors studied the power allocation and relay selection problem maximizing the energy efficiency of an Amplify-and-Forward relay channel with the help of a DNN.

In the context of cognitive radio networks, various resource allocation problems have also been tackled via deep learning [17]–[19]. In [17], the authors have proposed a deep learning approach for resource allocation problems in cognitive radio networks maximizing the spectrum and energy efficiency. In [18], a DNN is used to determine the power allocation over multiple channels of the opportunistic users. At last, in [19], a deep reinforcement learning approach is applied to solve a distributed resource allocation problem.

None of the above works consider *jointly* cooperative communications (via relaying) and cognitive radio networks, which significantly increases the difficulty of the problem under study. Also, these existing methods cannot be readily applied in our setting.

Our main contributions: In this paper, we propose to exploit the high potential of deep neural networks to solve non-convex power allocation problems to maximize the opportunistic achievable Shannon rate under a primary quality of service (QoS) constraint and individual device power constraints. To the best of our knowledge, machine learning techniques have not been employed yet to derive optimal power allocation policies for relay-aided cognitive radio networks.

More specifically, we leverage unsupervised learning and design a novel neural network architecture able to solve the non-convex constrained resource allocation problem. Our experimental results show that the proposed DNN architecture is able to minimize a customized loss function that takes into account both the secondary rate and the primary QoS constraint, without any overfitting effect while ensuring good generalization properties over the validation set. Furthermore, the secondary rate predicted by our proposed DNN for unseen system parameters (i.e., in the test data) is close to the optimal one obtained by exhaustive search (or brute force) while meeting the primary QoS constraint.

II. System model and problem formulation

The system under study, depicted in Fig. 1, is composed of a primary user or transmitter U_P and its destination D_P; a secondary full-duplex relay; and a secondary user U_S and its destination D_S, similarly to [5], [8], [9]. The received signals at the relay, primary and secondary destinations write as

$$Y_R = h_{PR}X_P + h_{SR}X_S + Z_R \quad (1)$$

$$Y_i = h_{Ri}X_R + h_{ii}X_i + h_{ji}X_j + Z_j, \quad (2)$$

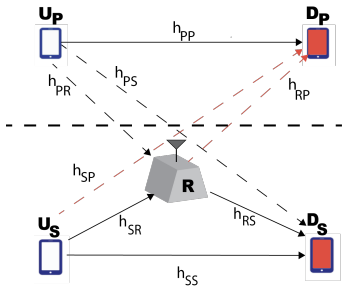


Fig. 1. Cognitive relay-aided network.

where $i \in \{P, S\}$, $j \in \{P, S\} \setminus i$; X_P, X_S and X_R, of average power P_P, P_S and P_R respectively, denote the message send by the primary user, the secondary user and the relay respectively; Z_R and Z_i denote the AWGN at the relay and at destination D_i of variance N_R and N_i respectively. Without loss of generality, we assume that all noise processes are of unit variance $N_R = N_S = N_P = 1$ or equivalently consider channel gains normalized by the receiver noise variance, defined as $g_{ij} = \frac{h_{ij}^2}{N_j}$. Furthermore, we assume that the channel gains follow a common fading and path-loss model given as $g_{ij} \sim \frac{\mathcal{N}(0,\sigma_g^2)}{\sqrt{1+d_{ij}^\gamma}}$, where d_{ij} denotes the distance between the nodes i and j and γ is the path loss factor [20]. We let $\mathbf{g} \triangleq (g_{ij}, \forall i, \forall j)$ denote the vector collecting all channel gains in the network. We assume that the relay performs full-duplex decode-and-forward (DF) and that the relay can cancel out any self-interference as in [5], [8], [9]. Also, the messages sent by the secondary user and the relay are treated as additional noise at the primary destination; and the primary message is treated as additional noise throughout the secondary network. Let $R_i, i \in \{P, S\}$ denote the achievable rate of the primary and secondary user respectively; and $\overline{R_P}$ denote the primary achievable rate in the absence of the secondary transmission:

$$\overline{R_P} = \frac{1}{2}\log_2(1 + g_{PP}P_P).$$

The primary network allows the opportunistic communication provided that the following minimum QoS constraint is met in terms of achievable primary rate [5], [8], [9]:

$$R_P \geq (1-\tau)\overline{R_P},\ \tau \in [0,1]. \quad (3)$$

As shown in [8], under DF relaying, the primary achievable rate is expressed as

$$R_P = C\left(\frac{g_{PP}P_P}{g_{RP}P_R + g_{SP}P_S + 2\alpha\sqrt{g_{SP}g_{RP}P_SP_R} + 1}\right),$$

where $C(x) = \frac{1}{2}\log_2(1+x)$ denotes the capacity of the point-to-point AWGN channel and $\alpha \in [0,1]$ comes from the used superposition coding technique and represents the tradeoff between sending the message from the previous block and sending a new one. Hence, the QoS constraint in (3) can be rewritten as $Q(\mathbf{g}, \alpha, P_S, P_R) \leq A$, with

$$Q(\mathbf{g}, \alpha, P_S, P_R) = g_{SP}P_S + g_{RP}P_R + 2\alpha\sqrt{g_{SP}g_{RP}P_SP_R},$$

$$A = \frac{g_{PP}P_P}{(1+g_{PP}P_P)^{1-\tau} - 1} - 1.$$

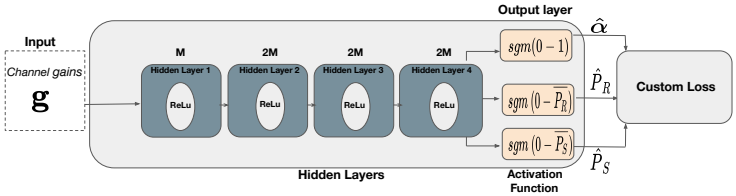


Fig. 2. Our proposed DNN architecture.

Our main objective is to maximize the opportunistic achievable rate R_S, when both the relay and secondary user are constrained by a maximum power expressed as $\overline{P_R}$ and $\overline{P_S}$ respectively. The achievable rate region is discussed in details and provided in [8]. Contrarily to [8], we consider here individual power constraints on each secondary device, i.e. secondary user and relay. To sum up, the resulting optimization problem under study writes as

$$(\mathbf{OP}) \max_{P_R,P_S,\alpha} \min \{f_R(\mathbf{g},\alpha,P_S,P_R), f_S(\mathbf{g},\alpha,P_S,P_R)\}$$
$$\text{s.t. } Q(\mathbf{g},\alpha,P_S,P_R) \le A, \quad (C1)$$
$$0 \le P_S \le \overline{P_S}, \quad 0 \le P_R \le \overline{P_R}, \quad (C2)$$
$$0 \le \alpha \le 1, \text{ with} \quad (C3)$$

$$f_R(\mathbf{g},\alpha,P_S,P_R) = \frac{g_{SR}(1-\alpha^2)P_S}{g_{PR}P_P+1},$$
$$f_S(\mathbf{g},\alpha,P_S,P_R) = \frac{g_{SS}P_S + g_{RS}P_R + 2\alpha\sqrt{g_{RS}g_{SS}P_SP_R}}{g_{PS}P_P+1}.$$

In the above, the objective function follows from the achievable rate of the secondary user:

$$R_S(\mathbf{g},\alpha,P_S,P_R) = C(\min\{f_R(\mathbf{g},\alpha,P_S,P_R), f_S(\mathbf{g},\alpha,P_S,P_R)\}),$$

as shown in [8]; constraint $(C1)$ is the primary QoS constraint; constraints $(C2)$ correspond to the individual power constraint of the relay and secondary user; constraint $(C3)$ follows from superposition coding.

III. Deep learning power allocation policy

A close analysis of the problem $(\mathbf{OP})$ reveals a non-convex objective function coupled by a non-convex QoS constraint $(C1)$, this is due to the non-linear operations performed by the relay. Hence, solving the non-convex $(\mathbf{OP})$ is a very challenging task. For this, we propose an unsupervised approach based on deep neural networks (DNN). At the opposite, a supervised approach would require computing a labeled training dataset containing the solutions to the non-convex problem $(\mathbf{OP})$ for a large number of sampled system parameters. This would be too computationally heavy and, henceforth, we opt for an unsupervised approach relying on a training dataset composed of only samples of the inputs (i.e., system parameters) of our DNN and exploits a specifically tailored communication loss to perform the training in an unsupervised manner.

A. Custom loss function

A key component of our proposed approach is the loss function that the network will be trained to minimize. Solving constrained optimization problems with DNNs is highly non-trivial, unless the constraints are of box-type such as $(C2), (C3)$. This is not the case of the QoS constraint $(C1)$ which is a difficult non-convex constraint. Nevertheless, as opposed to the power constraints $(C2)$, the primary QoS constraint is not a physical (hard) constraint but rather a requirement, which can be relaxed and included as a penalty in the objective function below

$$\mathcal{L} = \sum_{\ell=1}^{N} \left(-R_S(\mathbf{g}_\ell,\alpha,P_S,P_R) + \lambda[Q(\mathbf{g}_\ell,\alpha,P_S,P_R) - A]^+\right),$$

with $[x]^+ = \max\{0,x\}$ and N denoting the total number of channel realizations $\mathbf{g}_\ell$, $\ell \in \{1,\dots,N\}$ in the training dataset.

The hyperparameter λ denotes the unit price in bits/Watt of the primary QoS violation. A small value of λ will result in maximizing the achievable opportunistic rate without taking into account the primary QoS constraint; whereas large values of λ will strictly satisfy the primary QoS constraint but at the cost of opportunistic rate. This tradeoff between opportunistic rate and primary QoS will be further investigated via numerical results.

B. Proposed DNN architecture

Our proposed DNN architecture to solve $(\mathbf{OP})$ is composed of four fully connected hidden layers and is depicted in Fig. 2. The input consists of the channel gains vector $\mathbf{g}$ which is used to predict the outputs: $(\hat{\alpha},\hat{P}_R,\hat{P}_S)$, i.e., the solution of $(\mathbf{OP})$. The fully connected architecture is justified because of its generality and given that there is no *a priori* structural or temporal information within the input vector $\mathbf{g}$ to be exploited via more specific architectures such as convolutional or recurrent networks.

The four hidden layers are composed of $M-2M-2M-2M$ neurons with $M = 128$ and are followed by a rectified linear unit (ReLU) activation function due to its low computational complexity. This specific architecture is chosen based on extensive empirical experiments, as we discuss in the numerical section.

The final layer is followed by sigmoid activation functions: a standard one $\mathrm{sgm}_\alpha(x) = 1/(1+\mathrm{e}^{-x})$ to map the predicted α into its feasible set $[0,1]$, and two modified ones $\mathrm{sgm}_{P_i}(x) = \overline{P_i}/(1+\mathrm{e}^{-x})$, $i \in \{S,R\}$, to map the predicted powers P_R and P_S into $[0,\overline{P_R}]$ and $[0,\overline{P_S}]$ respectively. This final layer ensures that the hard constraints $(C2), (C3)$ are met.

IV. Numerical simulations

Below, we discuss our numerical setup and DNN training procedure as well as the performance evaluation of our

approach. Complete details and source codes of our experiments are available online at https://github.com/yacine074/USDNN-to-solve-powerallocation-problems-in-CRAN.git.

A. Experimental setup[1]

We consider a square cell of dimension 10×10 m in which the relay is positioned in the center whereas both primary and secondary user positions are uniformly distributed in the cell, unless specified otherwise. The path loss factor is set to $\gamma = 3$ and the channel gain standard deviation $\sigma_g = 7$. We assume that $P_P = \overline{P_R} = \overline{P_S} = 10$ W and set the threshold $\tau = 0.25$ for the maximum primary rate degradation.

Dataset: To the best of our knowledge, the majority of related works exploiting DNNs use simulated data [12], [13], [17], [18], given the lack of real data that is available and open access. Thus, unless specified otherwise to train and test our proposed DNN architecture, we use a dataset composed of three disjoint parts: i) a training set containing 10^6 channel realizations $\mathbf{g}_\ell$; ii) a validation set containing 2×10^5 channel realizations $\mathbf{g}_\ell$ jointly with the ground truth, i.e., the corresponding optimal resource allocation policies (α^*, P_R^*, P_S^*) obtained by brute force (or exhaustive search) for evaluating the generalization capability and for tuning the hyperparameters of our approach; iii) a test set containing 2×10^5 channel realizations jointly with the ground truth to evaluate the optimality gap of our predicted solution.

DNN training: In our numerical simulations, we used the ADAM optimizer to iteratively update the weights of our DNN. The batch size is set to 4096, the learning rate to 10^{-4}; these values allows the DNN weights optimization to converge within 1000 epochs.

Performance metrics and benchmark: We define here the relevant performance metrics used for the evaluation purpose. First, we define the relative gap between the predicted achievable rate via our DNN and the achievable rate obtained by brute force, our ideal benchmark, as follows:

$$\mathrm{G} = \frac{\frac{1}{N}\sum_{\ell=1}^{N} \hat{R}_{S,\ell} - R_{S,\ell}^*}{\frac{1}{N}\sum_{\ell=1}^{N} R_{S,\ell}^*} \tag{4}$$

where $\hat{R}_{S,\ell} = R_s(\mathbf{g}_\ell, \hat{\alpha}, \hat{P}_S, \hat{P}_R)$ denotes the secondary achievable Shannon rate obtained based on our DNN prediction for the ℓ-th sample in the dataset and $R_{S,\ell}^*$ denotes the optimal rate for the ℓ-th sample in the dataset obtained via brute force or exhaustive search. We chose brute force as benchmark thanks to its implementation simplicity and because it approximates the optimal solution with an adjustable precision.

Second, the degradation of the primary achievable rate caused by the opportunistic interference is defined as:

$$\Delta_\ell = 1 - \hat{R}_{P,\ell}/\overline{R}_{P,\ell}, \tag{5}$$

where $\hat{R}_{P,\ell} = R_P(\mathbf{g}_\ell, \hat{\alpha}, \hat{P}_S, \hat{P}_R)$ denotes the achievable primary rate of our predicted allocation. Based on this metric, we can also define the empirical outage as the proportion of samples in the dataset (or channel settings) for which the target primary QoS constraint is not met and the average primary rate degradation when in outage:

$$\text{Outage} = \frac{1}{N}\sum_{\ell=1}^{N} \mathbb{I}\left[\Delta_\ell > \tau\right], \tag{6}$$

$$\Delta_{\text{out}} = \frac{\sum_{\ell=1}^{N} \mathbb{I}\left[\Delta_\ell > \tau\right] \times \Delta_\ell}{\sum_{\ell=1}^{N} \mathbb{I}\left[\Delta_\ell > \tau\right]}, \tag{7}$$

where $\mathbb{I}[x]$ equals 1 when x is true and 0 otherwise.

B. DNN architecture choice

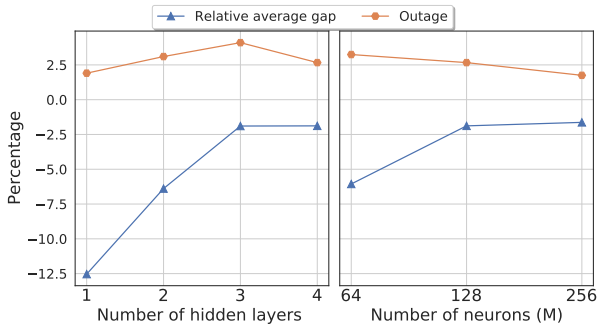


Fig. 3. Impact of the number of layers and number of neurons on the prediction performance over the validation set.

To choose the architecture in Fig. 2, we have performed extensive simulations over the validation set. In Fig. 3, we report the most significant results. On the left, we analyze the impact of the number of layers and compare four different architectures composed of one up to four hidden layers as follows: M, $M-2M$, $M-2M-2M$, and $M-2M-2M-2M$, with $M = 128$. We see that there is a significant gain in secondary rate when increasing the number of layers from 1 to 3; moving to 4 layers helps to decrease the outage. Hence, a 4-layer architecture is a good compromise between performance and computational cost. Now, on the right, we compare three different such four-layer DNNs, by varying the number of neurons per layer $M \in \{64, 128, 256\}$. We see that increasing M from 64 to 128 neurons leads to a significant gain in secondary rate; increasing further the number of neurons does not seem justified given the incurred computational cost. For these reasons, we choose $M = 128$ coupled with the 4-layer architecture in Fig. 2 henceforth.

C. No overfitting: performance on train and validation sets

In Fig. 4, we plot the evolution of our custom loss function $\mathcal{L}$ over the number of epochs within the training and validation sets for $\lambda \in \{10^{0.5}, 10^2\}$. First, notice that the DNN training converges within 1000 epochs. The superposed performance over the training and validation sets hints towards a good generalization capability of our proposed DNN. More importantly, since the loss function does not increase within the validation set, our approach does not suffer from overfitting. Finally, the decay rate of the loss function is much faster for $\lambda = 10^{0.5}$. Indeed, when λ becomes small, the custom loss is mainly rate-driven and not much emphasis is put on the primary QoS

[1]Our proposed DNN architecture in Fig. 4 is robust to changes within this experimental setup. Although the training needs to be relaunched for different setups, our core conclusions carry over in general.

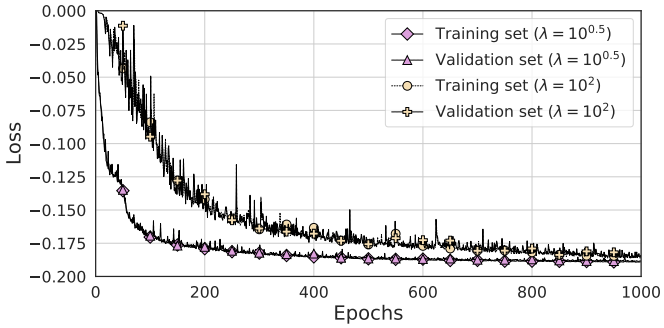


Fig. 4. Evolution of the loss function evolution over the training epochs over the training and validation sets (no overfitting effects).

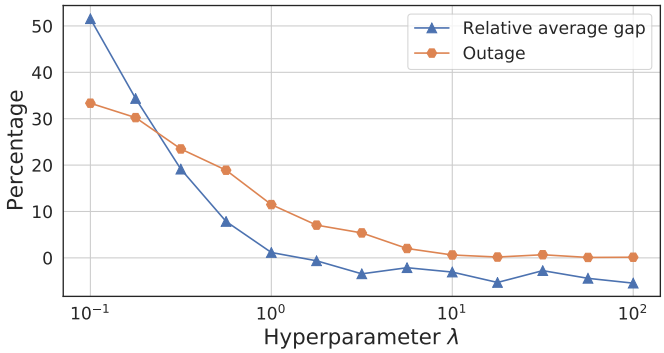


Fig. 5. Relative average gap G and outage as functions of the hyperparameter λ over the validation set.

constraint; this leads to a much easier optimization problem to solve. At the opposite, for larger values of λ, the custom loss puts an emphasis on satisfying $(C1)$, which leads to a more difficult problem. Of course, this parameter needs to be tuned carefully as discussed below.

D. Choice of the hyperparameter λ

In Fig. 5, the relative gap in (4) and the outage in (6) are depicted as functions of λ over the validation set. For small values of λ (rate-driven custom loss), the relative gap G is positive, which means that the secondary rates obtained via the DNN are larger than the optimal ones via brute force. The reason is that the primary QoS constraints are not necessarily met by our DNN solutions, as illustrated by the high outage levels. At the opposite, for large values of λ (primary QoS-driven custom loss), the outage goes to zero as expected at a cost in terms of secondary rates. Indeed, our predicted secondary rates are smaller than the optimal ones (negative relative gain G), but this gap is kept below 10 %.

In Fig. 6, we investigate closer the impact of λ on the primary rate degradation. For this, we plot the average and maximum values of the primary rate degradation as well as the average degradation when in outage Δ_{out} in (7) in Fig. 6(a). Also, in Fig. 6(b), we illustrate the histogram of the primary rate degradation (Δ) for $\lambda \in \{10^{0.5}, 10^{2}\}$. The mean primary rate degradation falls quickly below the threshold $\tau = 25$ %. For small values of λ, the worst case primary degradation can reach up to 90 %. Nevertheless, such extreme degradation is obtained only for a small number of out-layer data points. This is indicated by the curve Δ_{out} hitting the 25 % threshold reasonably fast as well as by the histogram of the degradation in Fig. 6(b).

To sum up, the value $\lambda = 10^{0.5}$ achieves a good tradeoff between the achievable secondary rate and the primary QoS degradation in our setting and will be used in the test phase below.

E. Prediction performance on test data

Henceforth, we evaluate the performance of our predicted solution $(\hat{\alpha}, \hat{P}_R, \hat{P}_S)$ obtained with new samples (i.e., test data) unseen during the training or validation phases.

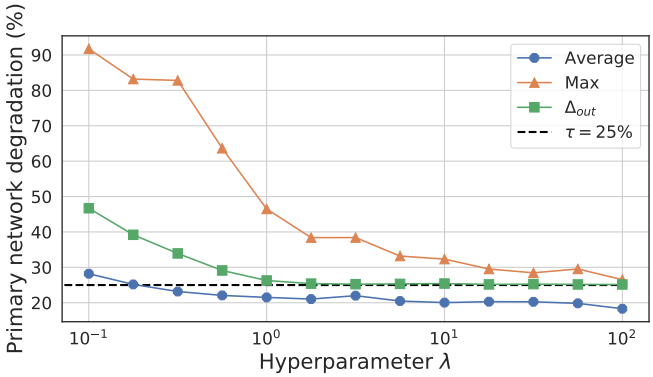
First, over our initial test set in Sec. IV-A, we obtain a relative average gap $G = -1.39$ %, Outage $= 2.60$ % and the average degradation when in outage $\Delta_{\text{out}} = 25.18$ %; this performance is coherent with the validation results demonstrating the high generalization capability of our approach on unseen data.

Impact of the relay position: We will now change the setting of Sec. IV-A and fix the positions of the primary and secondary user/destination nodes: $U_S(5, 2.5)$, $D_S(7.5, 5)$, $U_P(2.5, 5)$, $D_P(5, 7.5)$ as in Fig. 7; and consider that the relay position (x_R, y_R) varies within the cell. Hence, we generate a second test set composed of 10^4 channel realizations. In Fig. 7(a), Fig. 7(b), and Fig. 7(c), we illustrate the average primary rate degradation, the average predicted relay power $\hat{P}_R$ and the average secondary rate $\hat{R}_S$, respectively as functions of the relay position (x_R, y_R) over this new test set.

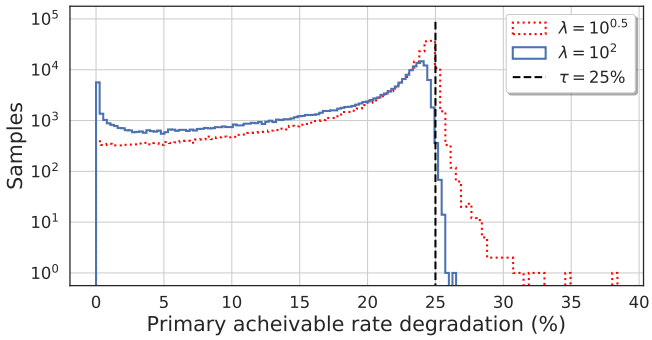
In Fig. 7(a), we see that the average degradation in the primary rate always falls below the fixed threshold of $\tau = 25$ % irrespective from the relay position. When the relay is very close to the primary nodes, the degradation drops below 20 %, since very little power is allocated to the relay as shown in Fig. 7(b). The worst case degradation arises when the relay lies between the secondary nodes, since the secondary rate improvement overcomes the damage the relay causes to the primary link. At last, in Fig. 7(b), we also notice that more power is allocated to the relay when it is close to the secondary user. This is to be expected since DF relaying is known to perform well in terms of achievable rate in these cases, which is indeed confirmed in Fig. 7(c).

V. Conclusions

In this paper, we investigated the power allocation problem in a cognitive radio network where the secondary user is assisted by a full-duplex relay node performing Decode-and-Forward. Because of the complex relaying operations, neither the secondary rate nor the primary QoS constraint are convex, leading to a difficult constrained Shannon rate maximization problem. To this aim, we proposed an unsupervised deep learning method based on fully connected architectures and a communication-tailored loss function modeling the secondary rate and the primary QoS constraint. We evaluated the performance of our approach via extensive numerical simulations. Our results show the high generalization capability of our DNN solution on unseen data and without overfitting. More importantly, we highlighted the tradeoff between the secondary

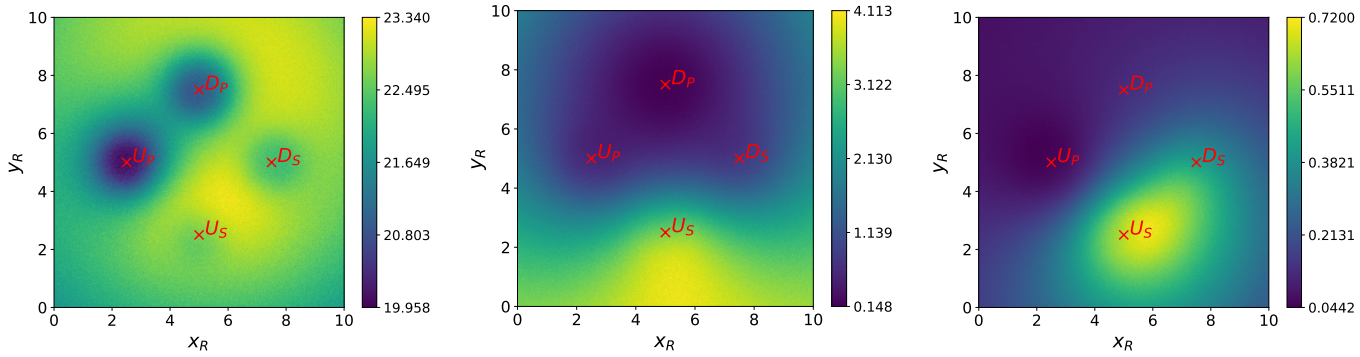


(a) Average and maximum primary rate degradation and average degradation when in outage (Δ_{out}) as functions of λ over the validation set.



(b) Histogram of the primary rate degradation over the validation set.

Fig. 6. Impact of the hyperparameter λ.



(a) Average primary rate degradation (%).

(b) Average predicted relay power (W).

(c) Average secondary rate (bpcu).

Fig. 7. Impact of the relay position (x_R, y_R).

rate and satisfying the primary QoS constraint. The primary QoS constraint is satisfied on average even though some outlayer points can be observed for secondary rate-driven losses. At last, Decode-and-Forward relaying was found to perform well when the relay is close to the secondary user.

References

[1] T. Huang, W. Yang, J. Wu, J. Ma, X. Zhang, and D. Zhang, "A survey on green 6G network: Architecture and technologies," *IEEE Access*, vol. 7, pp. 175 758–175 768, 2019.

[2] Z. Mobini, "Secure cooperative spectrum sharing in full-duplex multi-antenna cognitive radio networks with jamming," *AEU - Int. J. Electron. Commun.*, vol. 128, p. 153495, 2021.

[3] L. Zhang, M. Xiao, G. Wu, M. Alam, Y.-C. Liang, and S. Li, "A survey of advanced techniques for spectrum sharing in 5G networks," *IEEE Wireless Commun.*, vol. 24, pp. 44–51, 2017.

[4] D. Xu, X. Yu, Y. Sun, D. Ng, and R. Schober, "Resource allocation for IRS-assisted full-duplex cognitive radio systems," *IEEE Trans. Commun.*, vol. 68, no. 12, pp. 7376–7394, 2020.

[5] A. Savard and E. V. Belmega, "Optimal power allocation in a relay-aided cognitive network," in *EAI ValueTools*, 2019, pp. 15–22.

[6] H. Thampy and A. V. Babu, "Outage probability analysis and optimization of cognitive full-duplex relay networks," *Wireless Personal Commun.*, vol. 105, no. 4, pp. 1329–1352, 2019.

[7] M. Kamal, M. Kader, S. Islam, H. Yu *et al.*, "Device-to-device aided cooperative relaying scheme exploiting spatial modulation: An interference free strategy," *Sensors*, vol. 20, no. 24, p. 7048, 2020.

[8] A. Savard and E. V. Belmega, "Full-duplex relaying for opportunistic spectrum access under an overall power constraint," *IEEE Access*, vol. 8, pp. 168 262–168 272, 2020.

[9] ——, "Optimal power allocation policies in multi-hop cognitive radio networks," in *IEEE PIMRC*, 2020, pp. 1–6.

[10] W. Lee, M. Kim, and D.-H. Cho, "Deep power control: Transmit power control scheme based on convolutional neural network," *IEEE Commun. Lett.*, vol. 22, no. 6, pp. 1276–1279, 2018.

[11] Y. Liu, C. He, X. Li, C. Zhang, and C. Tian, "Power allocation schemes based on machine learning for distributed antenna systems," *IEEE Access*, vol. 7, pp. 20 577–20 584, 2019.

[12] H. Sun, X. Chen, Q. Shi, M. Hong, X. Fu, and N. D. Sidiropoulos, "Learning to optimize: Training deep neural networks for wireless resource management," in *IEEE SPAWC*, 2017, pp. 1–6.

[13] F. Liang, C. Shen, W. Yu, and F. Wu, "Towards optimal power control via ensembling deep neural networks," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1760–1776, 2019.

[14] M. Eisen, C. Zhang, L. F. Chamon, D. D. Lee, and A. Ribeiro, "Learning optimal resource allocations in wireless systems," *IEEE Trans. Signal Process.*, vol. 67, no. 10, pp. 2775–2790, 2019.

[15] K. I. Ahmed, H. Tabassum, and E. Hossain, "Deep learning for radio resource allocation in multi-cell networks," *IEEE Network*, vol. 33, no. 6, pp. 188–195, 2019.

[16] Y.-y. Guo, J. Yang, X.-l. Tan, and Q. Liu, "An energy-efficiency multi-relay selection and power allocation based on deep neural network for Amplify-and-Forward cooperative transmission," *IEEE Wireless Commun. Lett.*, 2021.

[17] F. Zhou, X. Zhang, R. Q. Hu, A. Papathanassiou, and W. Meng, "Resource allocation based on deep neural networks for cognitive radio networks," in *IEEE CIC (ICCC)*, 2018, pp. 40–45.

[18] W. Lee, "Resource allocation for multi-channel underlay cognitive radio network based on deep neural network," *IEEE Commun. Lett.*, vol. 22, no. 9, pp. 1942–1945, 2018.

[19] F. Shah-Mohammadi and A. Kwasinski, "Deep reinforcement learning approach to QoE-driven resource allocation for spectrum underlay in cognitive radio networks," in *IEEE ICC Workshops*, 2018, pp. 1–6.

[20] Z. Ding, Z. Yang, P. Fan, and H. V. Poor, "On the performance of non-orthogonal multiple access in 5G systems with randomly deployed users," *IEEE Signal Process. Lett.*, vol. 21, no. 12, pp. 1501–1505, 2014.