Joint RIS Optimization and Channel Estimation With Decision Tree Based Adaptive Reconfiguration

Anna V. Guglielmi[1], *Member, IEEE*, and Stefano Tomasin[1,2], *Senior Member, IEEE*

[1] Dept. of Information Engineering, University of Padova, Italy
[2] Consorzio Nazionale Interuniversitario di Telecomunicazioni, CNIT, Parma, Italy
email: {annavaleria.guglielmi, stefano.tomasin}@unipd.it

***Abstract*—Reconfigurable intelligent surfaces (RISs) are seen as a promising technology to improve cellular network coverage, due to their ability to steer the impinging signals in desired directions. The design of the RIS can be easily addressed by assuming full channel knowledge. Nevertheless, estimating the channels to and from the RIS is a challenging problem, as it requires a huge training overhead. This paper proposes an efficient configuration optimization jointly with channel estimation by exploiting deep learning tools. In particular, we propose an algorithm that works in two steps. The first step is based on a decision tree that requires few end-to-end channel estimates with different RIS configurations. The configurations are iteratively selected based on an estimate of the mutual information between the obtained rates and the optimal configuration. The second step instead provides the minimum mean-square-error estimate of the optimal RIS configuration based on the data rates estimated on the channels obtained in the first step through a neural network (NN) trained with a supervised approach. Numerical results confirm that the proposed solution provides a configuration close to the optimal, with achievable rates approaching the upper bound obtained with perfect channel knowledge.**

***Index Terms*—Channel Estimation, Decision Tree, Neural Network, Phase Configuration, Reconfigurable Intelligent Surfaces.**

I. Introduction

Reconfigurable intelligent surfaces (RISs) are among the promising technologies of next-generation wireless networks, thanks to their ability to purposely shape the wireless environment [1]. Indeed, a RIS induces a phase shift on reflected wireless signals according to the state of many nearly passive meta-material elements. The overall *RIS configuration* steers wireless signals in desired directions, with benefits on coverage and data rate [2], especially in the millimetre-wave (mmWave) band [3].

However, to fully exploit the potential of RISs, the phase shifts must be properly selected, which is challenging. To this end, two main difficulties arise: (i) the problem is non-convex, as optimization variables are phases of the complex numbers used in linear combination for the objective and the constraints; (ii) the propagation channels to and from the RIS must be estimated with a significant signaling overhead. In particular, while the receiver easily obtains an estimate of the end-to-end channel from the source for a specific RIS configuration, the separate estimate of the channels to and from the RIS remains challenging [4].

Some existing works jointly optimize the transmitter precoder and the RIS configuration through alternating optimization, successive convex optimization, and semidefinite relaxation algorithms [5], [6]. Still, the two optimization problems can be solved in cascade, by first optimizing the RIS and then the precoder, where the latter optimization leverages well-known approaches for multiple-input multiple-output (MIMO) channels. In this paper, we focus on the joint RIS optimization and overall end-to-end channel (including the RIS) estimation, while beamforming design and data detection are performed in a later stage. Another option, left for future study, sees the inclusion of this second step into a single process [7].

Existing solutions have been classified in [8] into a) model-based, b) heuristic, and c) machine learning (ML) methods.

Model-based Methods: Model-based methods refer to algorithms that approximate the RIS optimization problem by imposing specific structures to the problem; moreover it is assumed that the channel is perfectly known. [1] These structures include majorization-maximization problems [9], [10], stochastic successive convex approximation [11], alternating optimization [12], and alternating direction method of multipliers [13]. A cosine similarity theorem-based algorithm is exploited to compute suboptimal phase shifts in [14]. The maximization of the cascade channel trace is targeted in [15], and a suboptimal solution based on the dominant eigenvector of matrices obtained from the channels from and to the RIS is derived. However, these approaches achieve reasonable performance at the cost of a high computational effort due to the typically large number of RIS elements, dynamic channel conditions, and channel estimation requirements. In addition, function transformations and relaxations are typically required to solve optimization problems, resulting in methods strongly depending on specific channel models that may be unrealistic.

Heuristic Methods: Heuristic methods instead solve the RIS optimization (without specific structures) using general heuristic algorithms. These methods typically sacrifice optimality in favor of lower complexity and faster solutions. Among heuristic algorithms, greedy approaches [16], convex-concave procedure [17], and meta-heuristic algorithms [18] have been investigated. However, these approaches only find

This work was supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) Mission 4, Component 2, Investment 1.3, CUP C93C22005250001, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART").

[1] Note that machine learning methods can be based on models, but in [8] the authors use *model-based* to describe at best the common features of a type of optimization problems.

local optima that may be far from the global optimum [17]. To overcome these issues, heuristic methods might require many iterations entailing in turn many channel estimates [16].

ML Methods: Due to the recent advances in ML technology, deep-learning-based techniques are becoming attractive to optimize RISs [19]. In [20], the authors design a deep neural network (DNN) model that takes as input the channel knowledge sampled from some active RIS elements. The proposed model is trained offline to estimate the optimal RIS configuration. In [21], [22], the received signal is used to train a DNN that estimates the optimal RIS configuration and the transmit beamformer, bypassing explicit channel estimation. A DNN model that learns the implicit correlation between the optimal RIS configuration and the estimated receiver position is presented in [23]. For other similar investigations, we refer the reader to [24]–[27]. Supervised learning is considered in these works, yielding a significant computational burden.

In this paper, we propose a RIS optimization approach based on two steps. The first step estimates the achieved data rate under different RIS configurations. The configurations are selected adaptively by exploring a decision tree designed to maximize the mutual information between the estimated rates and the optimal RIS configuration. We denote this step as decision-tree configuration optimization (DTCO). The second step instead provides the minimum mean squared error (MSE) estimate of the optimal RIS configuration based on the rates estimated on the channel observed in the first step through a neural network (NN) trained with a supervised approach. We denote this step as optimal configuration estimation (OCE). The resulting algorithm is denoted as decision tree learning model (DTLM) for RIS optimization.

According to the above classification, our solution is between the *heuristic* and the *ML* methods. Indeed, the adaptive selection of the RIS configurations in DTCO serves as a baseline and supplement for a ML approach for the estimation of the optimal RIS configuration. By suitably choosing the RIS configurations in DTCO, we limit the number of channel estimates while providing an effecting input to the NN. Therefore, with respect to the existing heuristic methods, we improve the overall computational overhead by adaptively determining a sequence of RIS configurations getting closer to the optimal one with few channel estimates. Moreover, we obtain this sequence of configurations from the observed achievable rates, which is more raw information than the full channel matrix. When compared to the existing ML methods, our solution reduces the number of hidden layers while providing a data rate close to that obtained with perfect channel knowledge.

Still, with respect to the existing literature, due to the DTCO step, our solution is also codebook-based. Recently, a general framework for codebook-based RIS configurations has been provided in [28]. However, the DTLM procedure significantly differs from [28] due to the iterative and adaptive choice of the configurations to explore according to DTCO algorithm. Other works address the problem of reducing the overhead introduced by channel estimation. The deep reinforcement learning (DRL) algorithm for RIS optimization proposed in [29] employs the readily available user's location information to circumvent the channel estimation. Moreover, some other recent papers similarly deal with the adaptive design of the RIS coefficients during channel estimation [30], [31]. However, differently from the DTLM algorithm, these approaches are codebook-free and include the design of the transmit beamformer.

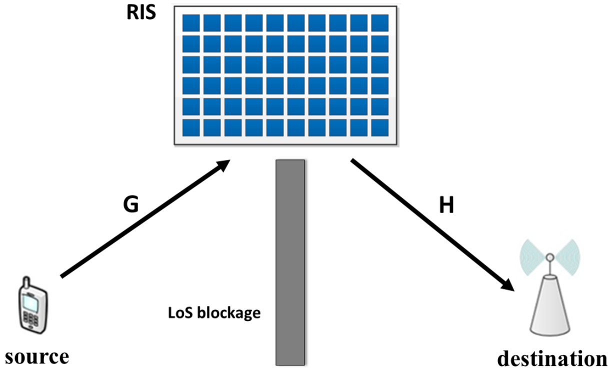

Figure 1. System model.

The main contributions of this paper are:

- A channel estimation and RIS optimization technique requiring very few channel estimates thanks to an adaptive selection of the RIS configurations used in the estimation process;
- The exchange of a small amount of information (channel data rates) between the decision tree and the NN, while still obtaining an accurate estimate of the optimal RIS configuration;
- A flexible solution that merges the advantages of heuristic and ML methods by operating an adaptive pre-processing of the estimated channel before estimating (by the NN) the optimal RIS configuration;
- The extensive validation of the performance of the DTLM algorithm for the optimal RIS design by simulations.

The rest of the paper is organized as follows. Section II introduces the considered system model. Section III describes the problem formulation, the channel estimation procedure, and the DTLM for the optimal RIS configuration: this includes the presentation of the DTCO and OCE algorithms. Section IV details the design of the main blocks of the DTLM. In Section V, we present and discuss the simulation results. The main conclusions are drawn in Section VI.

Notation: Throughout the paper, boldface capital letters and boldface lower-case letters such as $\boldsymbol{X}$ and $\boldsymbol{x}$ denote matrices and vectors, respectively. $(\cdot)^T$, $(\cdot)^*$, $(\cdot)^H$, $(\cdot)^{-1}$, and $(\cdot)^\dagger$ denote matrix transpose, conjugate, conjugate transpose, inverse, and pseudo inverse, respectively. $vec(\cdot)$ and $\mathrm{diag}(\boldsymbol{d})$ represent the vectorization and diagonal matrix whose main diagonal is $\boldsymbol{d}$, respectively. $||\cdot||$ and $|\cdot|$ represent the Frobenius norm and the cardinality of a set, respectively. Finally, $\mathbb{P}[\cdot]$ and $\mathbb{E}[\cdot]$ denote the probability and expectation operator, respectively.

II. System Model

We consider a narrowband single-user MIMO mmWave communication system between a source and a destination device, as shown in the model of Fig. 1. We assume that the direct source-destination channel is blocked, thus the connectivity between the source and the destination is maintained through an RIS controlled by the destination. Note that if the direct link is available, properly choosing the RIS configuration may be less relevant. In fact, the signal received via the direct link would be much stronger than that received via the RIS, making negligible the contribution of RIS to the overall channel. This scenario models for example the uplink of a cellular system, where the source is the user equipment and the destination is the base station.

We assume that both source and destination are equipped with uniform linear arrays (ULAs) with $N_{\rm S}$ and $N_{\rm D}$ antennas, respectively. The RIS has $N_{\rm I}$ passive reflective elements equally spaced along a line. To simplify the notation, we assume that both source and destination are on the same side of the RIS, and antennas/RIS elements are on the same plane and with the same orientation.

We indicate with $\boldsymbol{G} \in \mathbb{C}^{N_{\rm I}\times N_{\rm S}}$ and $\boldsymbol{H} \in \mathbb{C}^{N_{\rm D}\times N_{\rm I}}$ the source-RIS and RIS-destination baseband equivalent channel matrices, respectively. We adopt a block-fading channel model wherein both $\boldsymbol{G}$ and $\boldsymbol{H}$ remain constant during the channel coherence time, which is assumed to be larger than the duration of channel estimation considered in this paper. We assume that the RIS elements are implemented using only phase shifters, and define vector

$$\boldsymbol{\theta} = [\theta_1, \dots, \theta_{N_I}], \quad \theta_r \in [0, 2\pi) \tag{1}$$

and $r = 1, \dots, N_I$, collecting the phase shifts at each RIS element and also uniquely identifying the *RIS configuration*.

The end-to-end channel matrix between the source and destination for a given RIS configuration $\boldsymbol{\theta}$ is

$$\boldsymbol{Q}(\boldsymbol{\theta}) = \boldsymbol{H}\,\mathrm{diag}([e^{j\theta_1}, \dots, e^{j\theta_{N_I}}])\boldsymbol{G}. \tag{2}$$

III. Decision-Tree Learning Model for RIS Optimization

The performance of the communication channel between the source and the destination depends on $\boldsymbol{Q}(\boldsymbol{\theta})$, thus on the selected RIS configuration $\boldsymbol{\theta}$ and the channels $\boldsymbol{H}$ and $\boldsymbol{G}$. In this paper, we aim to select the RIS configuration $\bar{\boldsymbol{\theta}}$ that maximizes the achievable rate of the source-RIS-destination link, i.e.,

$$\begin{aligned}\bar{\boldsymbol{\theta}} &= \underset{\boldsymbol{\theta}}{\mathrm{argmax}}\, C(\boldsymbol{Q}(\boldsymbol{\theta})) \\ &= \underset{\boldsymbol{\theta}}{\mathrm{argmax}} \max_{\boldsymbol{P}} \log_2 \det\left(\boldsymbol{I} + \frac{1}{\sigma_R^2}\boldsymbol{Q}(\boldsymbol{\theta})\boldsymbol{P}\boldsymbol{Q}^H(\boldsymbol{\theta})\right),\end{aligned} \tag{3}$$

where σ_R^2 is the noise power per receive antenna during data transmission, $\boldsymbol{P}$ is the correlation matrix of the transmitted vector signal on the N_S antennas. Note that $\boldsymbol{P}$ is obtained from the singular value decomposition (SVD) of $\boldsymbol{Q}(\boldsymbol{\theta})$ and applying the waterfilling algorithm on the singular values to satisfy the transmit power constraint [32].

As discussed in the Introduction, the solution of (3) is complicated since channel matrices $\boldsymbol{H}$ and $\boldsymbol{G}$ are not readily available and the problem is non-convex. Indeed, the destination can easily estimate $\boldsymbol{Q}(\boldsymbol{\theta})$ for a given RIS configuration, using standard techniques for the estimation of MIMO channels. In particular, the source transmits pilot signals, and the destination uses a least square method to obtain the noisy estimate of the end-to-end channel

$$\hat{\boldsymbol{Q}}(\boldsymbol{\theta}) = \boldsymbol{Q}(\boldsymbol{\theta}) + \boldsymbol{w}, \tag{4}$$

where $\boldsymbol{w}$ is the estimation error matrix with independent Gaussian entries having zero mean and variance σ^2. Note that $\sigma^2 \leq \sigma_R^2$, since the latter is the noise power on each symbol and the channel estimate is averaged over multiple pilot symbols, thus reducing the estimation error power. However, the estimate of the end-to-end channel does not immediately provide the single channels $\boldsymbol{H}$ and $\boldsymbol{G}$, thus it is not easy to infer the end-to-end channel for other configurations.

Moreover, from (4) we note that the quality of the estimate of the channel (e.g., in terms of channel gain $||\boldsymbol{Q}(\boldsymbol{\theta})||^2$) depends on the used RIS configuration $\boldsymbol{\theta}$ through (2). This quality will also affect the estimation of the optimal RIS configuration to maximize the achievable rate. Thus the problems of channel estimation and RIS optimization are strictly interlaced.

To address the two problems jointly, we propose an adaptive strategy where the complete end-to-end channel $\boldsymbol{Q}(\boldsymbol{\theta})$ is estimated for several RIS configurations that are *adaptively* chosen during the estimation process itself to reduce the MSE of the estimate of the optimal RIS configuration. The selected RIS configuration at each iteration depends on the achievable rates estimated in previous iterations.

In detail, we propose an approach based on two steps. The first step, denoted decision-tree configuration optimization (DTCO) collects information on the end-to-end channel. Its output will be the set of estimated end-to-end channels and the set of estimated achievable rates obtained by using a suitably defined set of RIS configurations, which will also be output. The list of used RIS configurations is found iteratively, based on previously estimated rates and by exploring a decision tree trained with a supervised approach.

The second step instead obtains an estimate $\hat{\boldsymbol{\theta}}$ of the optimal RIS configuration based on the DTCO output, and is denoted optimal configuration estimation (OCE). In particular, the OCE step aims at obtaining the minimum MSE estimate of the optimal configuration, $\mathbb{E}[||\hat{\boldsymbol{\theta}} - \bar{\boldsymbol{\theta}}||^2]$ and is implemented by a NN trained in a supervised manner.

The resulting algorithm (including both steps) is denoted as decision tree learning model (DTLM) for RIS optimization. Fig. 2 shows the block scheme of the proposed DTLM solution. We observe that our solution fits also a multiuser scenario, where the joint channel estimation and RIS optimization are performed for each user separately. This can be achieved when user transmissions are transmitted over different time frames.

In the following, we provide the details of the two steps.

A. The DTCO Algorithm

The DTCO algorithm is the first step of the proposed DTLM scheme and aims at providing enough information for the best

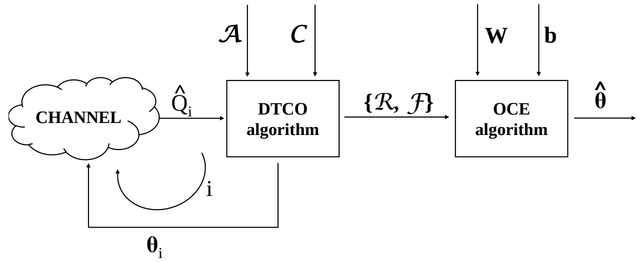

Figure 2. Block scheme of the DTLM solution, which consists of two steps: DTCO and OCE algorithms. Specifically, i is the index of the current DTCO iteration, $\boldsymbol{\theta}_i$ is the currently explored RIS configuration, $\hat{\boldsymbol{Q}}_i$ is the end-to-end estimated channel with the selected configuration $\boldsymbol{\theta}_i$, $\mathcal{A}$ and $\mathcal{C}$ are the codebooks of the achievable rate and the RIS configurations, respectively. When the maximum number of iterations K for DTCO is reached, the DTCO outputs the set of explored configurations $\mathcal{F}$ and the set of resulting (unquantized) rates $\mathcal{R}$. $\boldsymbol{W}$ and $\boldsymbol{b}$ are the weights and bias of the NN, respectively. $\hat{\boldsymbol{\theta}}$ is the estimated optimal RIS configuration.

estimation of the optimal RIS configuration in the next step.

The DTCO explores K different RIS configurations. The input of the DTCO is the estimated end-to-end channels[2] collected over K iterations, where at each iteration the RIS configuration is adaptively selected based on the achievable rates obtained with RIS configurations in the previous iterations.

At the end of the DTCO process, the output of this algorithm is a) the set of configurations that have been used at the kth channel estimation, $k = 1, \ldots, K$,

$$\mathcal{F} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}, \tag{5}$$

and b) the set of the achievable rates obtained with the K explored RIS configurations, i.e.,

$$\mathcal{R} = \{C(\hat{\boldsymbol{Q}}_1), \ldots, C(\hat{\boldsymbol{Q}}_K)\}, \tag{6}$$

where $\hat{\boldsymbol{Q}}_1, \ldots, \hat{\boldsymbol{Q}}_K$, are the end-to-end estimated channels with the selected RIS configurations.

To simplify the DTCO algorithm, the RIS configurations in the K iterations are selected from a finite codebook of Z configurations

$$\mathcal{C} = \{\boldsymbol{c}_1, \ldots, \boldsymbol{c}_Z\}, \tag{7}$$

with $[\boldsymbol{c}_z]_n \in [0, 2\pi)$, for $z = 1, \ldots, Z$, and $n = 1, \ldots, N_I$. Moreover, the selection of the RIS configuration is based on a *quantized version* of the achievable rates in the previous iterations, where the codebook of p quantized rates is

$$\mathcal{A} = \{a_1, \ldots, a_p\}. \tag{8}$$

The quantization of the rates is performed with a minimum distance criterion, as the unquantized rate C is quantized into

$$\tilde{C} = \underset{a \in \mathcal{A}}{\operatorname{argmax}}\, |C - a|. \tag{9}$$

In detail, the DTCO algorithm works as follows.

An initial configuration $\boldsymbol{\theta}_1 \in \mathcal{C}$ is selected, as better described in Section IV-B, and the end-to-end channel matrix $\hat{\boldsymbol{Q}}_1$ is estimated for this configuration. Then, the corresponding rate $C(\hat{\boldsymbol{Q}}_1)$ is computed by (3), stored in $\mathcal{R}$, i.e., the set of *unquantized* estimated rates. The rate is also quantized into $\tilde{C}_1 \in \mathcal{A}$.

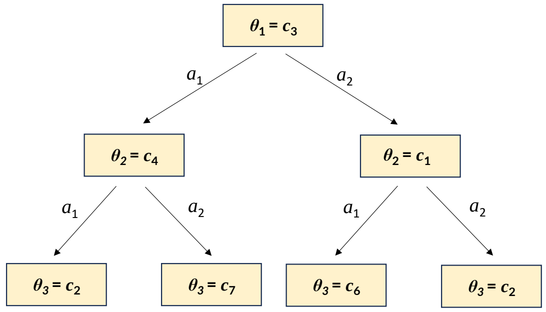

Figure 3. Example of decision tree of the DTCO algorithm for $K = 3$ explored configurations and a $p = 2$ size of the rate quantization codebook.

Then, $K-1$ iterations are performed. At the generic iteration $k = 2, \ldots, K$, let the set of previously explored configurations be

$$\mathcal{F}_{k-1} = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_{k-1}\}, \tag{10}$$

and the set of previously obtained quantized achievable rates be

$$\tilde{\mathcal{R}}_{k-1} = \{\tilde{C}_1, \ldots, \tilde{C}_{k-1}\}. \tag{11}$$

Then, the configuration selected for the next end-to-end channel estimation is taken from codebook $\mathcal{C}$ using function $f(\cdot)$ described in Section IV-B as

$$\boldsymbol{\theta}_k = f(\tilde{\mathcal{R}}_{k-1}, \mathcal{F}_{k-1}) \in \mathcal{C}. \tag{12}$$

For the selected configuration $\boldsymbol{\theta}_k$, the end-to-end channel is estimated and the obtained achievable rate is quantized into $\tilde{C}_k \in \mathcal{A}$.

After K iterations, the output of DTCO is the set of explored configurations $\mathcal{F} = \mathcal{F}_K$ and the set of unquantized rates $\mathcal{R}$.

DTCO as a Decision Tree: Since the configuration vectors and achievable rates are taken from codebooks, we can consider $f(\cdot)$ as a *decision function* among a finite set of alternatives (output set $\mathcal{C}$), given observations $\tilde{\mathcal{R}}_{k-1}$. Therefore, DTCO is a decision tree algorithm, where each level of the tree corresponds to an iteration, and each node provides the configuration to be used at the iteration given by the tree level. In particular, the set of quantized achievable rates $\tilde{\mathcal{R}}_{k-1}$ identifies the path from the root to the node of level k where the next configuration to be used is found. An example of the resulting tree for $p = 2$ and $K = 3$ is shown in Fig. 3, where we note that the index k of $\boldsymbol{\theta}_k$ is the node level (starting from the root at level $k = 1$), a_k enumerates the edges departing from each node, while in each node the selected configurations $\boldsymbol{\theta}_k$ depend on the decision process and are not related to the tree structure, as detailed in the following. By considering DTCO as a decision tree, we observe that the set of quantized achievable rates $\tilde{\mathcal{R}}_{k-1}$ uniquely identifies the set of explored configurations $\mathcal{F}_{k-1}$. Lastly, we note that the decision tree is used by the base station, which goes through it (from the root to one leaf) to select a new RIS configuration

[2] The end-to-end channels estimates are obtained using standard methods for the MIMO channel estimation as discussed at the beginning of this section.

for which the channel is estimated. Thus, the decision tree is neither explicitly communicated nor handled by the RIS.

DTCO Complexity: The computational complexity O_{DTCO} of DTCO algorithm depends on the dichotomic search performed at each node of the tree to determine the quantization value to associate with the observed achievable rate. Specifically, this requires $\log_2(2^p)$ comparisons, where 2^p is the number of quantization intervals. It follows then

$$O_{DTCO} = pK, \tag{13}$$

where the multiplication by K is due to the fact that K pilot sequences are transmitted so that K rates can be estimated.

B. The OCE Algorithm

The OCE algorithm provides the minimum MSE estimate $\hat{\boldsymbol{\theta}}$ of the optimal RIS configuration obtained from the set $\mathcal{F}$ of the used configurations and the set $\mathcal{R}$ of the resulting achievable rates, as provided by the DTCO algorithm. We aim at solving the following problem

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, \mathbb{E}[\,||\boldsymbol{\theta} - \bar{\boldsymbol{\theta}}||^2 \mid \mathcal{R}, \mathcal{F}\,]. \tag{14}$$

For the estimation, we resort to an ensemble of p^{K-1} NNs, one for each possible set of configurations $\mathcal{F}$, i.e., for each leaf of the decision tree. The set $\mathcal{F}$ given by the DTCO selects the NN that is fed with the set of corresponding unquantized rates in $\mathcal{R}$ and provides as output the estimated RIS configuration $\hat{\boldsymbol{\theta}}$. Each NN is trained on a dataset of achievable rates obtained with the specific $\mathcal{F}$ and aims at minimizing the MSE.

The NNs capture the non-linear relationships among system parameters and rates, as the optimal RIS configuration depends on the unknown $\boldsymbol{H}$ and $\boldsymbol{G}$ from (3). These channel matrices depend on the electromagnetic characteristics of surroundings, including their geometry, scatter materials, and source/destination positions. Such a dependency cannot be captured by mathematical modeling. Nevertheless, we aim at obtaining a RIS optimization process that is aware of the electromagnetic environment. Thus, we resort to the ensemble of NNs that, by suitable training, can learn the characteristics of the environment.

In this ML framework, DTCO can be considered as a pre-processing algorithm (also known as *feature selection algorithm*) of the input to a NN for the MSE estimation of the optimal RIS configuration. Note that we could also include the full estimated channel instead of only the achievable rate. However, this would imply a significant increase in complexity and here we focus on the set of achievable rates.

The ensemble of NNs is trained in a supervised manner. In particular, we assume that we have p^{K-1} datasets, one for each set $\mathcal{F}$,

$$\mathcal{D}(\mathcal{F}) = \{\mathcal{R}_n, \bar{\boldsymbol{\theta}}_n\} \tag{15}$$

of rates provided by the DTCO algorithm and corresponding optimal RIS configurations. Such datasets can be obtained by simulations or measurements in the field. The optimal RIS configuration $\bar{\boldsymbol{\theta}}$ can be computed using one of the several model-based approaches available in the literature [14], [15], [33], [34]. If the considered datasets are large enough to represent the environment where the system is then deployed, each NN will provide accurate estimations of the optimal RIS configuration. Note that the output of each NN is not constrained to be in the quantization set $\mathcal{C}$. Lastly, we compute the complexity of the OCE algorithm in Section IV-C, once the main NN architecture parameters are introduced.

NN Ensemble: The ensemble of NNs yields a complexity burden, in particular when either p or K (the number of the quantized data rates and the tree depth) are large. However, this solution ensures the highest flexibility in the estimation process, as each NN is trained on a specific set of explored configurations from which the input data rates are obtained.

Single NN With Input $(\mathcal{F}, \mathcal{R})$: A first alternative to the NN ensemble is a single NN with input both sets $\mathcal{F}$ and $\mathcal{R}$. While we still provide all the information to the NN as in the case of multiple NN, the complexity may not be significantly reduced, since the number of possible input sets $\mathcal{F}$ is finite, thus we expect that the single NN will operate internally as an ensemble of p^{K-1} NNs. Indeed, this structure becomes advantageous when p^{K-1} becomes large.

Single NN With Input $\mathcal{R}$: A second option is a single NN taking as input only the set $\mathcal{R}$ (as each NN of the ensemble). In this case the ML model does not know the set of RIS configurations selected by DTCO, thus we may expect a suboptimal performance. Moreover, since the input space becomes larger and the relation between each input with the target optimal RIS configuration becomes more complex, we expect that such single NN will require more layers and neurons than each NN of the ensemble. In Section V we compare the performance of the NN ensemble and single NNs with input $\mathcal{R}$.

Maximum-Rate Configuration Design: As an alternative to the MSE approach, we may consider the optimization of the RIS in the OCE algorithm to maximize the achievable rate. In this case (14) is replaced by

$$\hat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\operatorname{argmin}}\, \mathbb{E}\left[-\log_2 \det\left(\boldsymbol{I} + \frac{1}{\sigma_R^2}\boldsymbol{Q}(\boldsymbol{\theta})\boldsymbol{P}\boldsymbol{Q}^H(\boldsymbol{\theta})\right) \mid \mathcal{R}, \mathcal{F}\right]. \tag{16}$$

Note that we are considering the opposite of the rate (with a minus sign) and a minimization problem to frame it into a standard ML notation, where the loss function is minimized. Such an approach is more consistent with the DTCO. However, we note that using the rate as a loss function entails a significantly higher computational complexity than using the MSE. Indeed, to obtain the rate for a specific channel and configuration we need to optimize $\boldsymbol{P}$, which requires computing the SVD of the cascade channel and the water-filling solution on the singular values: such operations must be computed also to evaluate numerically the gradient of the loss function of the OCE NN.

IV. Design of DTLM Algorithm

In this section, we address the design of the blocks of the DTLM algorithm. In particular, we describe the design of a) the configuration codebook $\mathcal{C}$ for DTCO, b) the function $f(\cdot)$ to select the configuration in the decision tree, and c) the NN ensemble. We consider supervised learning for the design of both the decision tree (i.e., the function $f(\cdot)$) and the NN.

A. Design of Codebook $\mathcal{C}$

For the design of the codebook $\mathcal{C}$ we resort to the Linde-Buzo-Gray (LBG) algorithm [35] (also known to K-mean clustering) applied on the dataset $\mathcal{L}=\{\bar{\boldsymbol{\theta}}\}$ of optimal RIS configurations obtained from a set of channels $\{\boldsymbol{H}, \boldsymbol{G}\}$. Numerical methods should be used for the computation of the optimal RIS configuration unless we can exploit the structure of the channels. This occurs for example for channels with a single path, as better detailed in Section V-A.

The LBG algorithm provides the codebook that minimizes the mean square error between the optimal RIS configuration and its quantized version. Let $\boldsymbol{c}(\bar{\boldsymbol{\theta}})$ be the quantized version (in codebook $\mathcal{C}$) of the optimal RIS configuration $\bar{\boldsymbol{\theta}}$. The LBG algorithm computes the codebook $\mathcal{C}^*$ that solves

$$\mathcal{C}^* = \operatorname*{argmin}_{\mathcal{C}} \sum_{\bar{\boldsymbol{\theta}} \in \mathcal{L}} \frac{1}{||\mathcal{L}||}[||\boldsymbol{c}(\bar{\boldsymbol{\theta}}) - \bar{\boldsymbol{\theta}})||^2], \tag{17}$$

where $\boldsymbol{c}(\bar{\boldsymbol{\theta}})$ are taken from the set $\mathcal{C}$ over which the minimization is performed. The LBG algorithm is run offline (by simulations).

B. Design of Function $f(\cdot)$

The objective of the DTCO algorithm is to provide information on the channel (through the quantized achievable rate) good for the estimation of the optimal RIS configuration. At iteration k, we measure this information by conditional mutual information (CMI) between the optimal configuration $\bar{\boldsymbol{\theta}}$ and the estimated rates, given the current node in the tree (or, equivalently, $\tilde{\mathcal{R}}_{k-1}$) and the considered RIS configuration $\boldsymbol{\theta}_k$, i.e.,

$$I(\bar{\boldsymbol{\theta}}; \tilde{C}_k | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k). \tag{18}$$

Note that (18) is evaluated based on the statistics of the channel that leads to the statistics of $\bar{\boldsymbol{\theta}}$ and $\tilde{C}_k$. Then, for a given set of quantized estimated achievable rates $\tilde{\mathcal{R}}_{k-1}$, function $f(\tilde{\mathcal{R}}_{k-1}, \mathcal{F}_{k-1})$ provides the configuration $\boldsymbol{\theta}_k$ (among those not already explored) that maximizes the CMI between the optimal configuration $\bar{\boldsymbol{\theta}}$ and the quantized data rate, given previously obtained quantized rates $\tilde{\mathcal{R}}_{k-1}$, i.e., (for $k > 1$)

$$\boldsymbol{\theta}_k = f(\tilde{\mathcal{R}}_{k-1}, \mathcal{F}_{k-1}) = \operatorname*{argmax}_{\boldsymbol{c} \in \mathcal{C} \setminus \mathcal{F}_{k-1}} I(\bar{\boldsymbol{\theta}}; \tilde{C}_k | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k = \boldsymbol{c}). \tag{19}$$

For the first iteration, we have instead

$$\boldsymbol{\theta}_1 = \operatorname*{argmax}_{\boldsymbol{c} \in \mathcal{C}} I(\bar{\boldsymbol{\theta}}; \tilde{C}_1 | \boldsymbol{\theta}_k = \boldsymbol{c}), \tag{20}$$

where we drop the condition on previous quantized achievable rates from the CMI. Regarding Fig. 3, the value of $\boldsymbol{\theta}_3$ for the node achieved starting from the root and following branches a_1 and a_2 ($\boldsymbol{\theta}_3 = \boldsymbol{c}_7$ in the figure) is obtained by (19) with $\mathcal{F}_2 = \{\boldsymbol{c}_3, \boldsymbol{c}_4\}$.

For the computation of the CMI, first, let us define the following probability mass distribution (PMD) functions

$$p_{\bar{\boldsymbol{\theta}}, \tilde{C}_k | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k}(\boldsymbol{\theta}, a | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c}) = \mathbb{P}[\bar{\boldsymbol{\theta}} = \boldsymbol{\theta}, \tilde{C}_k = a | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c}], \tag{21a}$$

$$p_{\tilde{C}_k | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k}(a | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c}) = \mathbb{P}[\tilde{C}_k = a | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c}], \tag{21b}$$

$$p_{\bar{\boldsymbol{\theta}} | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k}(\boldsymbol{\theta} | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c}) = \mathbb{P}[\bar{\boldsymbol{\theta}} = \boldsymbol{\theta} | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c}], \tag{21c}$$

where for the first iteration ($k = 1$) we define $\tilde{\mathcal{R}}_0 = \emptyset$, i.e., we drop the condition on the quantized achievable data rates. Note that the randomness in the probabilities (21) is due to the randomness of the channels that yield random optimal RIS configurations and data rates. Thus the probabilities are obtained from the channel statistics, as better detailed in the following.

The CMI is then computed as

$$\begin{aligned} & I(\bar{\boldsymbol{\theta}}; \tilde{C}_k | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k = \boldsymbol{c}) = \\ & = \sum_{\boldsymbol{\theta} \in \mathcal{C}} \sum_{a \in \mathcal{A}} p_{\bar{\boldsymbol{\theta}}, \tilde{C}_k | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k}(\boldsymbol{\theta}, a | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c}) \times \\ & \log \left(\frac{p_{\bar{\boldsymbol{\theta}}, \tilde{C}_k | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k}(\boldsymbol{\theta}, a | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c})}{p_{\tilde{C}_k | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k}(a | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c}) p_{\bar{\boldsymbol{\theta}} | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k}(\boldsymbol{\theta} | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c})} \right). \end{aligned} \tag{22}$$

On the Computation of the PMDs: For the computation of PMDs (21) we first generate a dataset $\mathcal{D}$ obtained by randomly generating channel realizations, evaluating the quantized data rates for all possible RIS configurations, and identifying the optimal RIS configuration $\bar{\boldsymbol{\theta}}$. The generic entry of $\mathcal{D}$ is $[\bar{\boldsymbol{\theta}}, \tilde{Q}(\boldsymbol{c}_1), \ldots, \tilde{Q}(\boldsymbol{c}_Z)]$ for each channel realization. To compute the PMD $p_{\bar{\boldsymbol{\theta}}, \tilde{C}_k | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{\theta}_k}(\boldsymbol{\theta}, a | \tilde{\mathcal{R}}_{k-1}, \boldsymbol{c})$, we extract from $\mathcal{D}$ all entries having among the quantized data rates all the values in $\tilde{\mathcal{R}}_{k-1}$. In this sub-dataset, we evaluate the sampling frequency of the entries with optimal RIS configuration $\bar{\boldsymbol{\theta}} = \boldsymbol{\theta}$ and quantized data rate $\tilde{Q}(\boldsymbol{c}) = a$. A similar procedure is applied to compute the other PMDs in (21). The size of the dataset grows with the size of the RIS configuration codebook, Z, and the computation of the PMDs may become very complex for large codebooks.

On The Use of CMI: The choice of the (conditional) mutual information between the RIS configuration and quantized achievable rate as a target metric aims at (i) limiting the computational complexity of CMI evaluation; and (ii) being consistent with the performance metric of the proposed scheme. Indeed, the final objective of this paper is the optimization of the RIS to maximize the achievable rate of the link, and this task requires (in part) the estimation of the channel. What matters for the achievable rate is not only the value of the channel matrix entries but also their relation (through the eigenvectors and eigenvalues). The achievable rate is then more meaningful to consider as the performance metric of interest and in the evaluation of (18).

C. Design of The NNs Ensemble

As we are considering an ensemble of NNs in OCE, a different NN is considered (and then trained) for each set of explored configurations $\mathcal{F}$, i.e., for each leaf of the DTCO tree. We expect that the training of each NN would benefit from this assumption since the path traversed on the decision tree provides information on the optimal configuration. Indeed, the NN ensemble better captures the relationship between the observed achievable rates $\mathcal{R}$ and optimal RIS configuration $\bar{\boldsymbol{\theta}}$.

The architecture of each NN includes L layers, each in turn composed of B_ℓ neurons, $\ell \in \{1, \ldots, L\}$, and followed by a

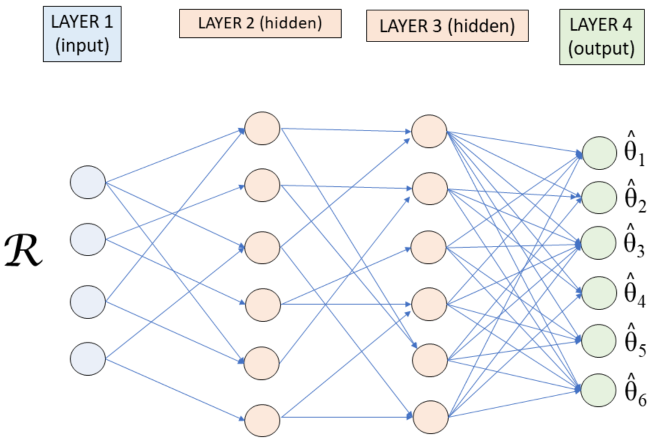

Figure 4. Example of the considered neural network, with $L = 3$ layers, $K = 4$ DTCO iterations, and $N_I = 6$ RIS elements.

non-linear activation function [36]. Fig. 4 depicts the general NN structure. It is worth noting that the layers of the designed NN do not have physical meanings. In general, multiple layers are required to provide the sophisticated mapping from the observed achievable rates to the optimal phase shifts introduced by the RIS.

Specifically, the $B_\ell \times 1$ vector $\boldsymbol{F}_\ell$ is the input of layer ℓ, where B_ℓ is the input dimension of layer ℓ. For the first layer ($\ell = 1$) we have $\boldsymbol{F}_1 = \mathcal{R}$, the set of K rates estimated by DTCO.

The output of the hidden layer $1 \leq \ell < L$ is computed as

$$\boldsymbol{F}_{\ell+1} = g_\ell(\boldsymbol{W}_\ell \boldsymbol{F}_\ell + \boldsymbol{b}_\ell), \tag{23}$$

where $\boldsymbol{W}_\ell$ (size $B_{\ell+1} \times B_\ell$) and $\boldsymbol{b}_\ell$ (size $B_{\ell+1} \times 1$) are the trainable weights and bias, with g_ℓ the activation function.

The output of the last layer ($\ell = L$) provides the estimate of the optimal RIS configuration

$$\hat{\boldsymbol{\theta}} = g_L(\boldsymbol{W}_L \boldsymbol{F}_L + \boldsymbol{b}_L). \tag{24}$$

The activation function in the last layer g_L is typically different from those used in the other layers. In fact, g_L strictly depends on the properties desired NN output, while at inner layers more generic non-linear functions are preferable. Considering that the target $\bar{\boldsymbol{\theta}}$ has real continuous entries in the interval $[0, 2\pi)$, we set g_L as the sigmoid function [37] scaled between 0 and 2π.

Since we aim at obtaining the MSE of the optimal RIS configuration from (14), we use the MSE loss function to train the NN, i.e.,

$$MSE(\bar{\boldsymbol{\theta}}, \hat{\boldsymbol{\theta}}) = \frac{1}{||\mathcal{D}(\mathcal{F})||} \sum_{(\mathcal{R}, \bar{\boldsymbol{\theta}}) \in \mathcal{D}(\mathcal{F})} ||\hat{\boldsymbol{\theta}}(\mathcal{R}) - \bar{\boldsymbol{\theta}}||^2. \tag{25}$$

OCE Complexity: The computational complexity O_{OCE} of the OCE step can be evaluated as the sum of the number of mathematical operations O_ℓ performed by all layers ℓ, $\ell = 1, \ldots, L$. In particular, O_ℓ is the sum of the number of multiplication $(O_{\ell,m})$ and additions $(O_{\ell,a})$ at each layer ℓ. Thus we have

$$O_{\ell,m} = B_{\ell+1} \cdot B_\ell, \tag{26}$$

due to the product $\boldsymbol{W}_\ell \boldsymbol{F}_\ell$, while

$$O_{\ell,a} = B_{\ell+1} \cdot (B_\ell - 1) + B_{\ell+1}, \tag{27}$$

where $B_{\ell+1} \cdot (B_\ell - 1)$ comes again from the product $\boldsymbol{W}_\ell \boldsymbol{F}_\ell$ and $B_{\ell+1}$ comes from adding the bias $\boldsymbol{b}_\ell$. It follows that

$$O_\ell = O_{\ell,m} + O_{\ell,a} = 2 \cdot B_{\ell+1} \cdot B_\ell, \tag{28}$$

and hence

$$O_{OCE} = \sum_{\ell=1}^{L} O_\ell = \sum_{\ell=1}^{L} 2 \cdot B_{\ell+1} \cdot B_\ell. \tag{29}$$

Ensemble of NNs vs Single NN With Input $\mathcal{R}$: We observe that designing a separate NN for each set $\mathcal{F}$ also requires fewer hidden layers and reduces the computational complexity of the OCE step.

V. Numerical Results

In this section, we evaluate and discuss the performance of the proposed DTLM algorithm for the optimization of the RIS configuration. We first describe the considered scenario, then we detail a benchmark approach used for comparison and the performance metric. Finally, we discuss the impact of different system parameters on the performance of the proposed solution.

A. Simulation Setup

Signals are transmitted in the mmWave band. We assume that both source-RIS and RIS-destination links exhibit a line-of-sight (LoS) condition while the direct link between source and destination is obstructed. We consider here a single-path scenario for the channels to and from the RIS. Let $\boldsymbol{\alpha}_{N_D}(\beta)$ be the array response column vector (also called steering vector) for an angle of arrival (AoA) β, with entries

$$[\boldsymbol{\alpha}_{N_D}(\beta)]_n = e^{j2\pi \frac{d}{\lambda}(n-1)\sin\beta}, \qquad n = 1, \ldots, N_D, \tag{30}$$

where λ is the wavelength at the carrier frequency. The two channel matrices can be written as

$$\boldsymbol{G} = \rho_G \boldsymbol{\alpha}_{N_I}(\eta_G) \boldsymbol{\alpha}_{N_S}^H(\gamma_G), \tag{31}$$

$$\boldsymbol{H} = \rho_H \boldsymbol{\alpha}_{N_D}(\eta_H) \boldsymbol{\alpha}_{N_I}^H(\gamma_H), \tag{32}$$

where ρ_G and ρ_H are the complex channel gains, η_G and η_H are the AoAs, and γ_G and γ_H are the angle of departures (AoDs). We assume that the AoA and AoD at the RIS are uniformly distributed in $\left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$. ρ_G and ρ_H are independent complex Gaussian variables with zero mean and unit power.

A focus on this scenario is motivated by the following main reasons. First, the considered system operates at mmWave frequencies, where one path is typically significantly dominant over a few others. The assumption of single-path transmissions is also justified by the rank of the cascade channel matrix, which is likely equal to one. This conclusion comes from the considerations reported in [38]–[40]. Moreover, in this

Table I
MAIN PARAMETERS OF NN AND ITS TRAINING.

Parameter	multiple-NN OCE	single-NN OCE
Nr. hidden layers	2	7
Nr. neurons	20	128
Activation func.	ReLu	ReLu
Optimizer	Adam	Adam
Learning rate	$1 \cdot 10^{-4}$	$1 \cdot 10^{-4}$
Batch size	128	128

scenario, we can compute the optimal RIS configuration in close form as

$$\bar{\boldsymbol{\theta}} = [0, \bar{\psi}, \ldots, (N_I - 1)\bar{\psi}]^T, \tag{33}$$

where

$$\bar{\psi} = 2\pi \frac{d}{\lambda}(\sin\gamma_H - \sin\eta_G). \tag{34}$$

Finally, for the single-path case, our solution will have a limited computational complexity.

The signal-to-noise ratio (SNR), defined as the inverse of the per antenna noise power $SNR = 1/\sigma_{\mathrm{R}}^2$, is considered in the range $[0, 30]$ dB. The channel is estimated using 100 symbols, thus $\sigma^2 = 1/(10^2 SNR)$.

With reference to the uplink transmission of a cellular system, the source (the user equipment) and the destination (the base station) are equipped with an ULA array of $N_S = 2$ and $N_D = 10$ antennas, respectively. The RIS has $N_I = 100$ elements arranged along a line and spaced by $d = \lambda/2$.

Table I summarizes the main parameter values of each NN used in OCE (both for the NN ensemble and single NN with input $\mathcal{R}$ cases) and its training procedure. Note that in the following we denote the use of the NN ensemble in OCE as *multiple-NN OCE* and the use of a single NN with input $\mathcal{R}$ in OCE as *single-NN OCE*, respectively. In Section V-G we consider the *single-NN OCE*, while all the other results of this Section are obtained with *multiple-NN OCE*.

Performance is evaluated then in terms of the average normalized achievable rate loss

$$\epsilon = \mathbb{E}\left[\frac{C(\boldsymbol{Q}(\bar{\boldsymbol{\theta}})) - C(\boldsymbol{Q}(\hat{\boldsymbol{\theta}}))}{C(\boldsymbol{Q}(\bar{\boldsymbol{\theta}}))}\right], \tag{35}$$

where the average is taken over the channel and noise realizations.

B. Benchmark

For comparison purposes, we consider a *benchmark* approach that applies the OCE on a fixed set of K RIS configurations $\mathcal{F}_B = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_K\}$, instead of considering the adaptive set of RIS configuration provided by the DTCO algorithm. The set $\mathcal{F}_B$ is obtained by maximizing the average mutual information between the set of achievable rates $\mathcal{R}(\mathcal{F}_B) = \{C(\hat{\boldsymbol{Q}}_1), \ldots, C(\hat{\boldsymbol{Q}}_K)\}$, resulting from using $\mathcal{F}_B$ and the optimal RIS configuration $\bar{\boldsymbol{\theta}}$. In formulas,

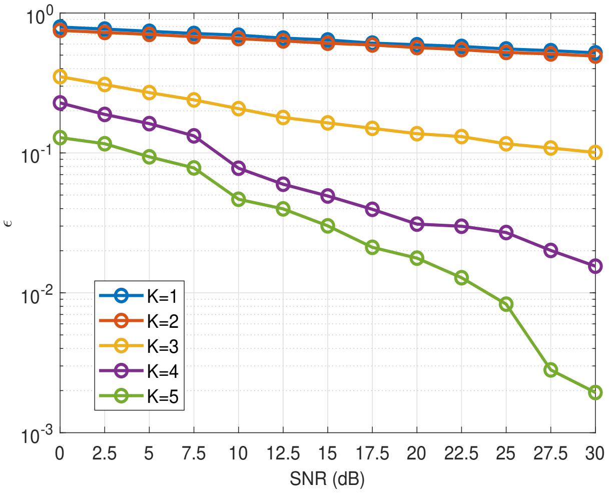

Figure 5. Average normalized achievable rate loss ϵ as a function of the SNR in $[0, 30]$ dB for $K \in \{1, 2, 3, 4, 5\}$.

$$\mathcal{F}_B = \underset{\mathcal{F} \in \tilde{\mathcal{F}}}{\operatorname{argmax}}\ \mathbb{E}[I(\bar{\boldsymbol{\theta}}; \mathcal{R}(\mathcal{F}))], \tag{36}$$

where the average is computed over the channel realizations and $\tilde{\mathcal{F}}$ represents the set of all the permutations of K elements from $\mathcal{C}$. Note that the choice of the configuration at each iteration of DTLM depends on the evaluation of the instantaneous CMI as in (19).

Remark: Although the *benchmark* is very close to the solution proposed in [28] it differs for the codebook generation. DTLM instead significantly differs from the solution in [28], as the latter has a fixed set of explored configurations while DTLM adaptively chooses the configurations for channel estimation. In Section V-E, we compare DTLM with both *benchmark* and the OCE algorithm of [28] with sum-distance maximization (SDM) codebook design.

C. DTLM Performance

We first assess the overall performance of DTLM as a function of the SNR and K, and then we examine its two steps separately.

Fig. 5 shows the average normalized achievable rate loss ϵ as a function of the SNR for the proposed DTLM. The number of iterations K is in the set $\{1, 2, 3, 4, 5\}$. As expected, a more accurate estimate of the optimal RIS configuration (which corresponds to a lower ϵ) is achieved at a higher SNR. Moreover, a larger number of channel estimates in the DTCO step of DTLM (i.e., a higher value of K) also provides a better estimate. Indeed, the proposed DTLM algorithm is particularly effective when for more than 2 iterations of the DTCO part. This is because K determines the number of input variables provided to each NN, and with less than 3 inputs the NN cannot learn the nonlinear relationship between the estimated achievable rates and the optimal RIS configuration $\bar{\boldsymbol{\theta}}$. This results in a very poor estimation capability of the trained NN. This explanation is confirmed by Fig. 6, which shows the MSE

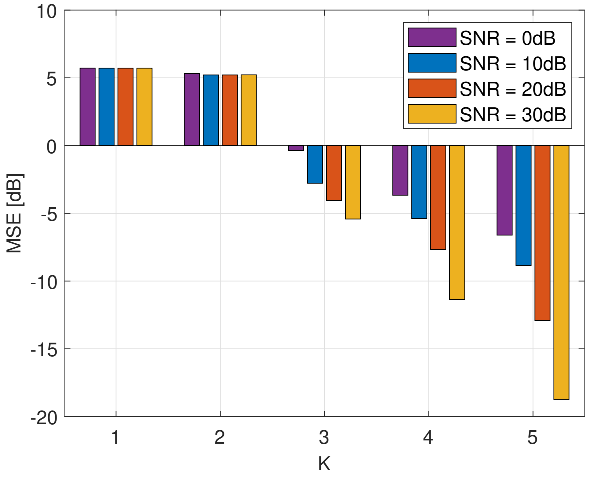

Figure 6. MSE on the estimated RIS configuration as a function of $K \in \{1, 2, 3, 4, 5\}$, for SNR in $\{0, 10, 20, 30\}$ dB.

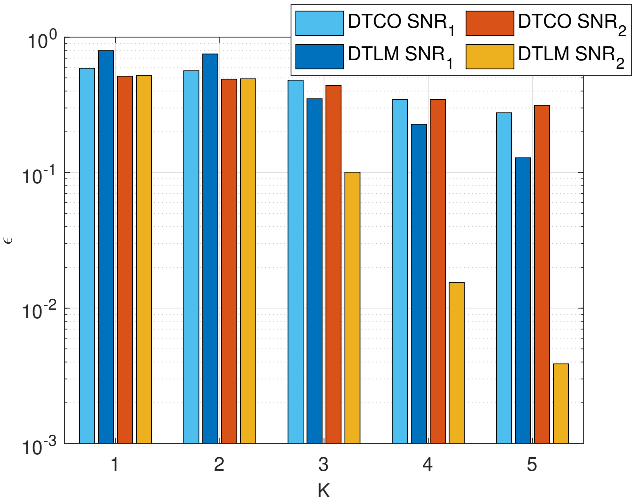

Figure 7. Average normalized achievable rate loss ϵ as a function of K considering both the output of the DTCO algorithm and the resulting achievable rate obtained using the estimated RIS configuration, for $\text{SNR}_1 = 10$ dB and $\text{SNR}_2 = 30$ dB.

on the estimated RIS configuration as a function of K for SNR $\in \{0, 10, 20, 30\}$ dB. It can be observed that the MSE is significantly higher than 0 dB for $K = 1, 2$, and then reaches a value approximately in the range $[-10, -20]$ dB when $K = 5$ according to the considered SNR. Moreover, for $K = 1, 2$, the MSE is not sensitive to the different SNR values. In fact, even when the noise power is reduced, only 1 or 2 input variables provided to the NN are not informative enough to allow the NN to be properly trained and then to estimate the optimal RIS configuration.

Nevertheless, it can be generally stated that $K = 5$ iterations of the proposed DTLM solution are sufficient to approach the optimal achievable rate with an accuracy of about 1% for low SNR and reach an accuracy of about 10^{-3} for high SNR. Furthermore, it should be emphasized that K is also the total number of end-to-end channel estimates in the proposed DTLM. This means that 5 channel estimates provide achievable rates approaching the upper bound of perfect channel knowledge. This is a remarkably low number of estimates considering the complexity of the channel and the number of optimized parameters.

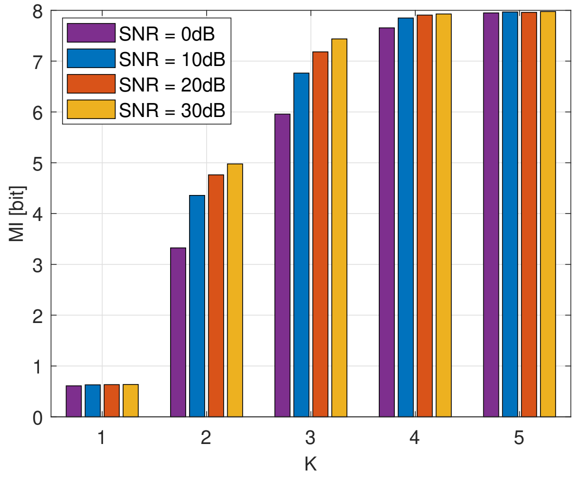

Figure 8. Mutual information between the observed rates output of the DTCO algorithm and the optimal RIS configuration as a function of $K \in \{1, 2, 3, 4, 5\}$, for SNR $\in \{0, 10, 20, 30\}$ dB.

D. Impact of DTCO and OCE

We now evaluate the contribution of the two steps (DTCO and OCE) to the DTLM for the optimal RIS design. Fig. 7 shows the average normalized achievable rate loss ϵ for different values of K at both the last DTCO RIS configuration (DTCO label in the figure) and the output of the DTLM for the optimal RIS design (DTLM label in the figure). Two values of SNR, $\text{SNR}_1 = 10$ and $\text{SNR}_2 = 30$ dB, are considered. In general, we note that the OCE step contributes significantly to the reduction of ϵ. We also notice that the improvement is more significant at higher SNRs: this is because the proposed technique is affected by the estimation noise, which is more significant at low SNRs.

To better understand the operation of the DTCO step, Fig. 8 shows the mutual information between the observed achievable rates at the output of the DTCO algorithm and the optimal RIS configuration $\bar{\boldsymbol{\theta}}$, for $K \in \{1, 2, 3, 4, 5\}$, and SNR $\in \{0, 10, 20, 30\}$ dB. The mutual information is obtained by averaging (18) over the channel realizations. The mutual information increases with K, regardless of the SNR, meaning that the explored configuration at each iteration gets closer to the optimal one $\bar{\boldsymbol{\theta}}$. This confirms the effectiveness of the feature selection by the DTCO algorithm. Finally, using the estimated RIS configuration further reduces ϵ, as shown in Fig. 7.

E. Comparison With Other Codebook-based Solutions

Fig. 9 shows the average normalized achievable rate loss ϵ as a function of K for the proposed DTLM, *benchmark* (label B), and [28] (label SDM). Specifically, in SDM we consider

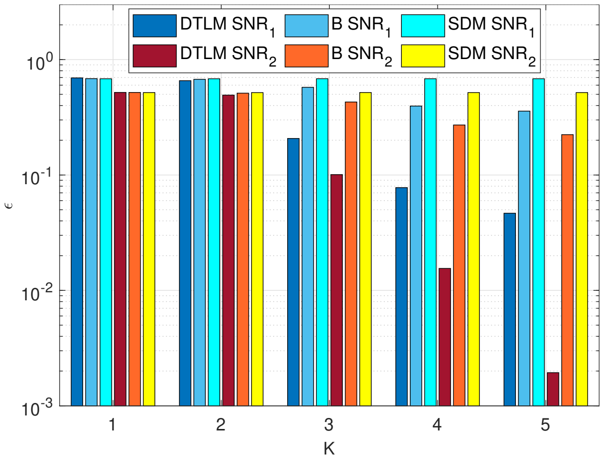

Figure 9. Average normalized achievable rate loss ϵ of the proposed DTLM, *benchmark*, and SDM approaches, as a function of K for $\text{SNR}_1 = 10$ dB and $\text{SNR}_2 = 30$ dB.

the sum-distance maximization method for the codebook generation and OCE to optimize the RIS. Moreover, we consider $K \in \{1, 2, 3, 4, 5\}$, $\text{SNR}_1 = 10$ dB, and $\text{SNR}_2 = 30$ dB. The DTLM algorithm outperforms both the *benchmark* and SDM for each value of K and SNR. In particular, the gain of DTLM over the two methods is larger for a higher K, especially for $K > 2$. This is because the proposed DTLM solution involves an initial adaptive pre-processing of the input to the NN estimating the optimal RIS configuration (i.e., the DTCO algorithm). The processing takes into account both the explored RIS configuration and the resulting quantized achievable rates for building the tree. Thus, for each new channel, a different configuration may be chosen at the same iteration according to the observed achievable rate. This does not occur in the benchmark and SDM approaches, where the sequence of explored configurations is fixed, regardless of the observed rate. Therefore, the comparison with the *benchmark* and SDM approaches is a further confirmation of the effectiveness of the adaptive configurations selection of DTCO, which yields a much more accurate estimate of the optimal RIS configuration through OCE.

F. Convergence

Here, we discuss the convergence with the training dataset size for both DTCO and OCE algorithms.

For DTCO, the dataset is used for the computation of the CMI and Fig. 10 shows the average normalized achievable rate loss ϵ as a function of the dataset size for $K \in \{1, 2, 3, 4, 5\}$, and SNR=20 dB. We observe that $4 \cdot 10^5$ samples are enough for the convergence of ϵ when $K = 3$. For $K > 3$, an even higher number of samples is required. We observe that the same intervals of ϵ can be achieved with different values of K and different dataset sizes. In general, a larger K requires smaller datasets for the same value of ϵ.

About the convergence of OCE, we stop the training of each NN when the current validation error is smaller than that of the last 6 iterations. For $K = 3$, 4, and, 5, the average number (on the NN ensemble) of epochs for convergence is 110, 115, and 125, respectively.

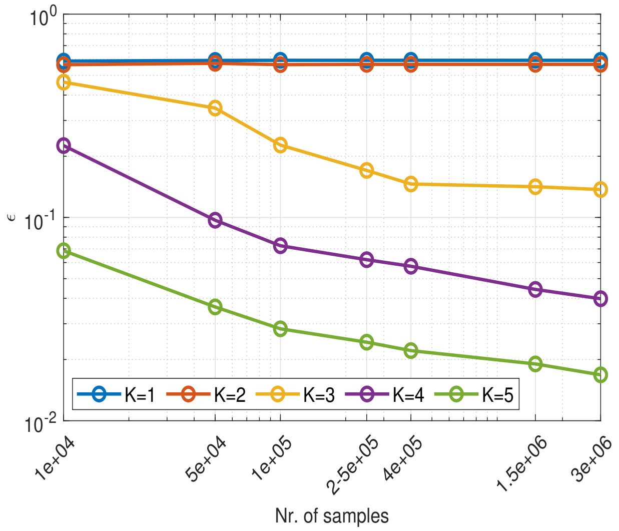

Figure 10. Average normalized achievable rate loss ϵ as a function of the number of samples considered for the CMI computation for $K \in \{3, 4, 5\}$ and SNR = 20 dB.

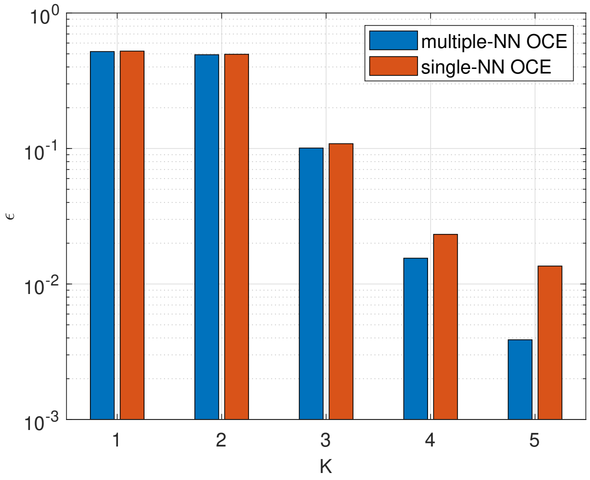

Figure 11. Average normalized achievable rate loss ϵ as a function of K when a NN for each leaf (blue bars) and only one NN for the entire dataset (orange bars) are used in OCE. In both cases, we assume SNR= 30 dB.

G. Impact of The NN Architecture

Previous figures are all obtained using the NN ensemble (multiple-NN OCE). We now consider the single NN solution to estimate the optimal RIS configuration regardless of the path traversed on the tree, i.e., the explored configuration. To this end, a deeper architecture is considered, with the parameters reported in Table I. While a *less* deep NN architecture would be sufficient when considering a different NN for each leaf node due to the higher specificity, we expect that training a NN that operates without awareness of the path traveled in the tree would require a more complex and deeper architecture for a more reliable learning process.

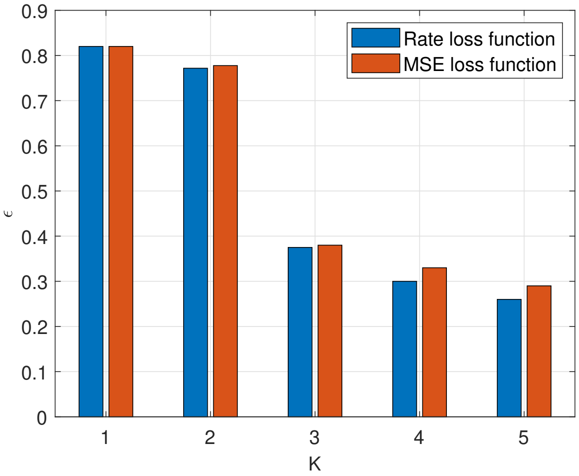

Figure 12. Average normalized achievable rate loss ϵ as a function of K for single-antenna devices and single-NN OCE when the MSE loss function (orange bar) and the rate loss function (blue bar) are used in OCE. In both cases, we assume SNR= 10 dB.

Fig. 11 shows the average normalized achievable rate loss ϵ as a function of K, when multiple-NN OCE (blue bars) and single-NN OCE (orange bars) are considered for SNR= 30 dB. We note that the solution with one NN with a deeper architecture performs closer to the multiple-NN OCE, at the expense of a higher complexity. Indeed, from (29) the computational complexity per RIS estimation is 1020 operations for the multiple-NN OCE and 10368 operations for single-NN OCE. We also find that the choice of multiple-NN OCE or single-NN OCE does not significantly affect the convergence of OCE. We then conclude that, in the considered scenario, the ensemble of NNs is indeed a better solution than the single NN.

H. Impact of the OCE loss function

Lastly, we have also considered the rate as the loss function for the training of the NN of the OCE, according to the approach described at the end of Section III-B. However, due to the computation of the gradient for a specialized loss function (rather than the standard MSE), the computation time for training increased significantly. Thus, we focus on a simpler scenario with single-antenna devices and single-NN OCE. For $K \in \{1, 2, 3, 4, 5\}$ and SNR $= 10$ dB, the obtained average normalized achievable rate loss ϵ with the MSE and loss and rate loss functions is shown in Fig. 12. We note that using the rate loss function slightly improves the performance of DTLM, at the expense of higher computational complexity, as discussed in Section III-B.

VI. Conclusions

In this paper, we have proposed a novel technique to optimize the configuration of a RIS in a wireless communication system, jointly with the channel estimation. Our proposed algorithm works in two steps. The first step collects information about the channel. Its output is the set of achievable rates obtained by using a suitably defined set of RIS configurations. The second step instead estimates the optimal RIS configuration based on the DTCO output: for this step, we resort to a NN trained with a supervised approach.

Simulation results confirm that the obtained configuration is close to the optimal one and the resulting achievable rates approach the upper bound under perfect channel knowledge. In particular, the proposed DTLM for the RIS design exhibits a fast convergence to a near-optimal solution, while requiring few end-to-end channel estimates.

As future work, it would be interesting to apply the proposed method to stacked intelligent metasurfaces [41]. Moreover, a reinforcement learning method to find the configuration that maximizes the achievable rate could be considered as an alternative approach to our solution.

References

[1] C. Huang, A. Zappone, G. C. Alexandropoulos, M. Debbah, and C. Yuen, "Reconfigurable intelligent surfaces for energy efficiency in wireless communication," *IEEE Trans. on Wireless Commun.*, vol. 18, no. 8, pp. 4157–4170, Aug. 2019.

[2] E. C. Strinati, G. C. Alexandropoulos, H. Wymeersch, B. Denis, V. Sciancalepore, R. D'Errico, A. Clemente, D.-T. Phan-Huy, E. De Carvalho, and P. Popovski, "Reconfigurable, intelligent, and sustainable wireless environments for 6G smart connectivity," *IEEE Commun. Mag.*, vol. 59, no. 10, pp. 99–105, Oct. 2021.

[3] M. D. Renzo, M. Debbah, D. T. P. Huy, A. Zappone, M. Alouini, C. Yuen, V. Sciancalepore, G. C. Alexandropoulos, J. Hoydis, H. Gacanin, J. de Rosny, A. Bounceur, G. Lerosey, and M. Fink, "Smart radio environments empowered by AI reconfigurable metasurfaces: An idea whose time has come," *CoRR*, vol. abs/1903.08925, 2019. [Online]. Available: http://arxiv.org/abs/1903.08925

[4] A. L. Swindlehurst, G. Zhou, R. Liu, C. Pan, and M. Li, "Channel estimation with reconfigurable intelligent surfaces—a general framework," *Proc. IEEE*, vol. 110, no. 9, pp. 1312–1338, May 2022.

[5] R. Liu, M. Li, Q. Liu, and A. L. Swindlehurst, "Joint symbol-level precoding and reflecting designs for IRS-enhanced MU-MISO systems," *IEEE Trans. on Wireless Commun.*, vol. 20, no. 2, pp. 798–811, Oct. 2021.

[6] H. Ur Rehman, F. Bellili, A. Mezghani, and E. Hossain, "Joint active and passive beamforming design for IRS-assisted multi-user MIMO systems: A vamp-based approach," *IEEE Trans. on Commun.*, vol. 69, no. 10, pp. 6734–6749, Jul. 2021.

[7] M. Li, S. Zhang, Y. Ge, F. Gao, and P. Fan, "Joint channel estimation and data detection for hybrid RIS aided millimeter wave OTFS systems," *IEEE Trans. on Commun.*, vol. 70, no. 10, pp. 6832–6848, Oct. 2022.

[8] H. Zhou, M. Erol-Kantarci, Y. Liu, and H. V. Poor, "A survey on model-based, heuristic, and machine learning optimization approaches in ris-aided wireless networks," *Arxiv 2303.14320*, Mar. 2023.

[9] C. Huang, A. Zappone, M. Debbah, and C. Yuen, "Achievable rate maximization by passive intelligent mirrors," in *Proc. IEEE Int. Conf. on Acoustics, Speech and Sig. Proc. (ICASSP)*, Sept. 2018, p. 3714–3718.

[10] G. Zhou, C. Pan, H. Ren, K. Wang, and A. Nallanathan, "Intelligent reflecting surface aided multigroup multicast miso communication systems," *IEEE Trans. on Sig. Proc.*, vol. 68, pp. 3236–3251, Jun. 2020.

[11] H. Guo, Y.-C. Liang, J. Chen, and E. G. Larsson, "Weighted sum-rate maximization for reconfigurable intelligent surface aided wireless networks," *Arxiv 1912.11999*, Dec. 2019.

[12] N. S. Perović, L.-N. Tran, M. D. Renzo, and M. F. Flanagan, "On the maximum achievable sum-rate of the RIS-aided MIMO broadcast channel," *IEEE Trans. on Sig. Proc.*, vol. 70, pp. 6316–6331, Jan. 2022.

[13] X. Liu, C. Sun, and E. A. Jorswieck, "Two-user SINR region for reconfigurable intelligent surface aided downlink channel," in *Proc. IEEE Int. Conf. on Commun. Work. (ICC Work.)*, Jun. 2021, p. 1–6.

[14] Z. Yigit, E. Basar, and I. Altunbas, "Low complexity adaptation for reconfigurable intelligent surface-based mimo systems," *IEEE Commun. Lett.*, vol. 24, no. 12, pp. 2946–2950, 2020.

[15] A. M. Sayeed, "Optimization of reconfigurable intelligent surfaces through trace maximization," in *Proc. IEEE Inter. Conf. on Commun. Work. (ICC Work.)*, Jun. 2021, pp. 1–6.

[16] A. V. Guglielmi and S. Tomasin, "Fast iterative configuration of reconfigurable intelligent surfaces in mmWave systems," *in Proc. 2023 IEEE Global Commun. Conf.*, pp. 631–636, Dec. 2023.

[17] Y. Chen, Y. Wang, J. Zhang, and M. D. Renzo, "QoS-driven spectrum sharing for reconfigurable intelligent surfaces (RISs) aided vehicular networks," *IEEE Trans. on Wireless Commun.*, vol. 20, no. 9, pp. 5969–5985, Sept. 2021.

[18] K. Zhi, C. Pan, H. Ren, and K. Wang, "Power scaling law analysis and phase shift optimization of RIS-aided massive mimo systems with statistical CSI," *IEEE Trans. on Commun.*, vol. 70, no. 5, pp. 3558–3574, May 2022.

[19] K. M. Faisal and W. Choi, "Machine learning approaches for reconfigurable intelligent surfaces: A survey," *IEEE Access*, vol. 10, pp. 27 343–27 367, Mar. 2022.

[20] A. Taha, M. Alrabeiah, and A. Alkhateeb, "Deep learning for large intelligent surfaces in millimeter wave and massive MIMO systems," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2019, pp. 1–6.

[21] Ö. Özdoğan and E. Björnson, "Deep learning-based phase reconfiguration for intelligent reflecting surfaces," in *Proc. 54th Asilomar Conf. on Signals, Systems, and Computers*, Sept. 2020, pp. 707–711.

[22] T. Jiang, H. V. Cheng, and W. Yu, "Learning to beamform for intelligent reflecting surface with implicit channel estimate," in *Proc. IEEE Global Commun. Conf.*, Dec. 2020, pp. 1–6.

[23] C. Huang, G. C. Alexandropoulos, C. Yuen, and M. Debbah, "Indoor signal focusing with deep learning designed reconfigurable intelligent surfaces," in *Proc. Int. Work. on Sig. Proc. Advances in Wireless Communications (SPAWC)*, Jul. 2019, pp. 1–5.

[24] A. Taha, M. Alrabeiah, and A. Alkhateeb, "Enabling large intelligent surfaces with compressive sensing and deep learning," *IEEE Access*, vol. 9, pp. 44 304–44 321, Mar. 2021.

[25] G. C. Alexandropoulos, S. Samarakoon, M. Bennis, and M. Debbah, "Phase configuration learning in wireless networks with multiple reconfigurable intelligent surfaces," in *Proc. 2020 IEEE Globecom Work.*, Dec. 2020, pp. 1–6.

[26] Y. Song, M. R. A. Khandaker, F. Tariq, K.-K. Wong, and A. Toding, "Truly intelligent reflecting surface-aided secure communication using deep learning," in *Proc. IEEE 93rd Vehic. Tech. Conf. (VTC2021-Spring)*, Apr. 2021, pp. 1–6.

[27] X. Hu, C. Masouros, and K.-K. Wong, "Reconfigurable intelligent surface aided mobile edge computing: From optimization-based to location-only learning-based solutions," *IEEE Trans. on Commun.*, vol. 69, no. 6, pp. 3709–3725, Mar. 2021.

[28] J. An, C. Xu, Q. Wu, D. W. K. Ng, M. Di Renzo, C. Yuen, and L. Hanzo, "Codebook-based solutions for reconfigurable intelligent surfaces and their open challenges," *IEEE Wireless Commun.*, vol. 31, no. 2, pp. 134–141, Apr. 2024.

[29] W. Xu, J. An, C. Huang, L. Gan, and C. Yuen, "Deep reinforcement learning based on location-aware imitation environment for RIS-aided mmWave MIMO systems," *IEEE Wireless Commun. Lett.*, vol. 11, no. 7, pp. 1493–1497, Jul. 2022.

[30] T. Jiang, F. Sohrabi, and W. Yu, "Active sensing for two-sided beam alignment and reflection design using ping-pong pilots," *IEEE Jour. on Sel. Areas in Info. Theory*, vol. 4, pp. 24–39, May 2023.

[31] F. Sohrabi, T. Jiang, W. Cui, and W. Yu, "Active sensing for communications by learning," *IEEE Jour. on Sel. Areas in Commun.*, vol. 40, no. 6, pp. 1780–1794, Jun. 2022.

[32] N. Benvenuto, G. Cherubini, and S. Tomasin, *Algorithms for Communications Systems and their Applications*. Wiley, 2020. [Online]. Available: https://books.google.it/books?id=JeESEAAAQBAJ

[33] M. Jung, W. Saad, M. Debbah, and C. S. Hong, "On the optimality of reconfigurable intelligent surfaces (RISs): Passive beamforming, modulation, and resource allocation," *IEEE Trans. on Wireless Commun.*, no. 7, Feb. 2021.

[34] J. Sanchez, E. Bengtsson, F. Rusek, J. Flordelis, K. Zhao, and F. Tufvesson, "Optimal, low-complexity beamforming for discrete phase reconfigurable intelligent surfaces," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Dec. 2021, pp. 01–06.

[35] Y. Linde, A. Buzo, and R. Gray, "An algorithm for vector quantizer design," *IEEE Trans. on Commun.*, vol. 28, no. 1, pp. 84–95, Jan. 1980.

[36] M. Alrabeiah and A. Alkhateeb, "Deep learning for TDD and FDD massive MIMO: Mapping channels in space and frequency," in *Proc. Asilomar Conf. on Signals, Systems, and Computers*, Nov. 2019, pp. 1465–1470.

[37] A. Menon, K. Mehrotra, C. K. Mohan, and S. Ranka, "Characterization of a class of sigmoid functions with applications to neural networks," *Neural Networks*, vol. 9, no. 5, pp. 819–835, Jul. 1996.

[38] Z.-Q. He and X. Yuan, "Cascaded channel estimation for large intelligent metasurface assisted massive MIMO," *IEEE Commun. Lett.*, vol. 9, no. 2, pp. 210–214, Feb. 2020.

[39] J. Rains, A. Tukmanov, Q. H. Abbasi, and M. Imran, "RIS-Enhanced MIMO Channels in Urban Environments: Experimental Insights," *techrxiv preprint techrxiv.24558475.v1*, Nov. 2023.

[40] T. S. Rappaport, S. Sun, and M. Shafi, "Investigation and comparison of 3GPP and NYUSIM channel models for 5G wireless communications," in *Proc. IEEE VTC-Fall*, Sept. 2017, pp. 1–5.

[41] J. An, C. Xu, D. W. K. Ng, G. C. Alexandropoulos, C. Huang, C. Yuen, and L. Hanzo, "Stacked intelligent metasurfaces for efficient holographic MIMO communications in 6G," *IEEE Journ. on Sel. Areas in Commun.*, vol. 41, no. 8, pp. 2380–2396, Aug. 2023.

Anna Valeria Guglielmi (Member, IEEE) received the B.Sc. degree, the M.Sc. degree, and the Ph.D. degree in information engineering from the University of Padova, Italy, in 2012, 2014, and 2018, respectively. In 2017, she has been a Visiting Scientist with the BIOTEC Technische Universität Dresden (TUD), Germany, and the University of California, Irvine. She is currently an Assistant Professor with the University of Padova. Her current research interests include machine-learning architectures and signal processing for wireless communication systems, and physical layer security.

Stefano Tomasin (Senior Member, IEEE) received the Ph.D. degree from the University of Padova, Italy (2003), where he is now a Full Professor. During his career, he has visited IBM Research (Switzerland), Philips Research (Netherlands), Qualcomm (California), the Polytechnic University in Brooklyn (New York), and Huawei (France). His current research interests include physical layer security, security of global navigation satellite systems, signal processing for wireless communications, synchronization, and scheduling of communication resources. He is a senior member of IEEE and a member of EURASIP. He is or has been an Editor of the IEEE Transactions on Vehicular Technologies, the IEEE Transactions on Signal Processing (2017-2020), the EURASIP Journal of Wireless Communications and Networking, and the IEEE Transactions on Information Forensics and Security.