Optimization of Passive Beyond-Diagonal RIS via Relaxation, Randomization, and Autoencoding

Anna Valeria Guglielmi, *Member, IEEE*, Mattia Scarin Callegaro, Yaser Dorrazehi, *Graduate Student Member, IEEE*, and Stefano Tomasin, *Senior Member, IEEE*

***Abstract*—We consider beyond-diagonal reconfigurable intelligent surfaces (BD-RISs) whose elements are connected in groups and aim at optimizing their configuration to maximize the achievable rate of the cascade channel. We propose two suboptimal approaches (i.e., semidefinite programming (SDP) and projected gradient ascent (PGA) solutions) to first find the BD-RIS configuration that maximizes the composite channel trace and then locally maximizes the achievable rate by a randomization approach. We impose a constraint on the choice of the coefficients to ensure that the BD-RIS is passive, i.e., it does not emit more power than that received. Still, our solution has a high communication overhead for a large number of connections among the BD-RIS elements. We then propose a dynamic mapping between the BD-RIS configuration and a small number of control variables. The mapping is provided by the encoder part of an autoencoder, trained to minimize a suitable loss function on the optimal configurations in the specific deployment. We also design the BD-RIS configuration directly in the latent space of the autoencoder, reducing the complexity. By simulations in a typical cellular communication scenario, we show that the group-connected BD-RIS can achieve up to 95% of the rate obtained for a fully-connected BD-RIS with two orders of magnitude lower complexity, while the autoencoder compression and configuration optimization in the latent space reduces the control rate by 90% with negligible rate loss.**

***Index Terms*—Autoencoder, beyond diagonal reconfigurable intelligent surfaces (BD-RISs), semidefinite programming relaxations (SDP), projected gradient ascent (PGA).**

I. Introduction

BEYOND-DIAGONAL reconfigurable intelligent surfaces (BDRISs) are devices composed of multiple elements, each emitting a linear combination of the (complex baseband equivalent) signals captured by several elements, with tunable coefficients. BD-RISs are an evolution of the diagonal reconfigurable intelligent surfaces (RISs), where each element emits only a phase-shifted version of the signal captured by the same element. A BD-RIS enables smarter wave manipulation and increased coverage at the expense of increased circuit complexity [1]. Such components have been the subject of several studies to model BD-RISs [2] and design their architecture [3], [4]. A BD-RIS can be organized into group- and fully-connected architectures, according to the circuit topology of connections between elements [5].

Several works in the literature propose solutions for the optimization of the configuration of diagonal RISs (see the survey [6] and recent papers [7], [8], [9]). Few works, instead, have been published on the BD-RIS. In [10], a theoretical performance bound for communication systems using the BD-RIS is obtained. A graph-theoretic optimization design for BD-RIS is proposed in [11]. In [12], a general BD-RIS optimization algorithm in the presence of mutual coupling among BD-RIS elements is analyzed. In [13], the authors investigate the effect of BD-RIS non-reciprocity (i.e., considering an asymmetric BD-RIS matrix) and identify the theoretical conditions for reciprocal and non-reciprocal BD-RISs to maximize the receive signal-to-noise ratio (SNR) in both uplink and downlink scenarios. Moreover, in [14], an optimal solution for a BD-RIS configuration is proposed, under the constraint that the BD-RIS is unitary. Closed-form fully connected BD-RIS solutions that maximize the equivalent channel gain have been obtained for single-input single-output (SISO) and multiple-input single-output (MISO)/single-input multiple-output (SIMO) channels [15]. In [16], the authors propose an algorithm for maximizing the achievable rate of a multiple-input multiple-output (MIMO) link supported by a BD-RIS with a fully-connected architecture. They apply Takagi's factorization to ensure that the BD-RIS matrix is unitary and symmetric. In [17], the author proposes the joint design pilot sequence and BD-RIS design to estimate the cascade channel, achieving the minimum mean squared error (MSE) of the least square channel estimator. The unitary constraint on the BD-RIS configuration is considered. In [18], the authors consider a constraint on the maximum eigenvalue of the BD-RIS reflection matrix to model its passive nature.

Besides the optimization of BD-RISs, a second problem is the complexity of the control of its tunable parameters [4], [11]. While in diagonal RIS each element has a single tunable parameter, in BD-RIS we have a tunable parameter for each *connection* of the element with other elements. Thus, the number of controls grows quadratically with the number of BD-RIS elements, yielding a significantly high communication overhead for the BD-RIS control. On the other hand, all the

Received 5 June 2025; revised 22 October 2025 and 18 December 2025; accepted 20 January 2026. Date of publication 28 January 2026; date of current version 6 February 2026. 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"). The associate editor coordinating the review of this article and approving it for publication was H. Du. *(Corresponding author: Anna Valeria Guglielmi.)*

The authors are with the Department of Information Engineering, University of Padova, 35122 Padua, Italy (e-mail: annavaleria.guglielmi@unipd.it; mattia.scarincallegaro@studenti.unipd.it; yaser.dorrazehi@studenti.unipd.it; stefano.tomasin@unipd.it).

Digital Object Identifier 10.1109/TCOMM.2026.3658615

flexibility of BD-RIS may not be needed once it is installed in a specific location. Therefore, the trade-off between the complexity of the control and the obtained improvement of the communication channel should be investigated. Recent works have addressed the issue of control and feedback overhead in (diagonal) RIS-assisted systems through the compression of the potentially high-dimensional configuration. In [19], the authors propose a convolutional autoencoder-based approach for efficient configuration feedback compression. In [20], a deep learning-based phase shift compression scheme is proposed using global attention mechanisms, which significantly reduces the overhead while maintaining accurate reconstruction of the RIS configuration. Finally, an autoencoder-based approach has also been designed in [21] for active sensing via RIS.

In this paper, we consider a passive BD-RIS with continuous tunable phase shifts and gains, and we aim at maximizing the achievable rate of the cascade channel between two communicating devices. Unlike prior works that assume a unitary or symmetric BD-RIS matrix to ensure lossless or reciprocal operation, we explicitly enforce a passive constraint based on the spectral norm of the BD-RIS configuration matrix. This formulation ensures that the surface does not emit more power than it receives, while allowing for a realistic representation of lossy and non-reciprocal impedance networks. We first cast the BD-RIS configuration design into a quadratic optimization problem that maximizes the trace of the composite channel. To make the problem tractable, we relax the original passivity constraint (expressed in terms of the spectral norm) to express both the objective function and the constraints in quadratic form, enabling the use of SDP relaxation and iterative projected gradient methods. We then perform a randomization approach [22]. In particular, when recovering a feasible solution from the trace maximization problem, we explore several random feasible solutions and select the one maximizing the achievable rate of the cascade channel rescaled to satisfy the original spectral-norm constraint, thereby restoring the physical passive nature of the BD-RIS. Note that [18] also considers our same model of BD-RIS; however, it is used to orthogonalize the channel rather than maximizing the sum-rate.

We then introduce a machine learning approach based on an autoencoder (AE) with two objectives. First, the AE is designed to learn a compressed (i.e., latent) representation of the optimal BD-RIS configuration obtained through the SDP relaxation. This reduces the number of control signals for the BD-RIS by transferring only the latent space parameters required by the decoder part of the AE to obtain the desired configuration. Second, we solve the optimization problem in the latent space with a reduced number of variables and lower computational complexity.

A. Contributions

The main contributions of this work are summarized as follows:

- We propose a BD-RIS optimization model under passive constraints. Unlike prior works that consider unitary and symmetric assumptions on the BD-RIS configuration, we formulate the optimization problem by enforcing the constraint that the emitted signal power cannot exceed the power of the impinging signals.
- We first cast the optimization as a quadratic problem based on the evaluation of the trace of the end-to-end cascade channel. Then, we propose a randomization step to select among the feasible solutions the one that maximizes the achievable rate of the cascade channel, restoring the original physical passive BD-RIS constraint.
- To mitigate the overhead due to the large number of control parameters, we introduce an autoencoder with a twofold objective: (i) it learns a compressed latent representation of the optimized BD-RIS configuration, thereby reducing the number of control signals to be transmitted; (ii) it enables solving the optimization problem directly in the latent space, significantly lowering the computational complexity. Note that prior works have reduced signaling overhead in diagonal RIS only through phase-shift compression.
- Simulation results show that for a fully-connected BD-RIS architecture, the proposed solution achieves performance comparable to the optimal design, while achieving a reduction of the control rate with negligible performance loss when the optimization is carried out in the latent space. We also show that comparing the results obtained for the group-connected and fully-connected architectures, it can be seen that the former requires significantly fewer computational operations compared to the latter, while still achieving comparable performance in terms of rate.

The rest of this paper is organized as follows. Section II introduces the considered system model. Section III describes the problem formulation and the proposed solutions for BD-RIS optimization. In Section IV, we design a machine learning approach to reduce both the computational complexity of the BD-RIS optimization and the control communication overhead. In Section V, we detail the BD-RIS optimization in a reduced space. The simulation results are discussed in Section VI. The main conclusions are drawn in Section VII.

Notation: Throughout the paper, $(\cdot)^T$, $(\cdot)^*$, $(\cdot)^H$, $(\cdot)^{-1}$, and $(\cdot)^\dagger$ represent matrix transpose, conjugate, conjugate transpose, inverse and pseudo inverse, respectively. Also, $\otimes$, $\diamond$, and $\|\cdot\|$ denote the Kronecker product, Khatri-Rao product by columns, and Frobenius norm, respectively. $\mathrm{vec}(\boldsymbol{X})$ converts matrix $\boldsymbol{X}$ into a column vector, by reading $\boldsymbol{X}$ row-wise from top left to bottom right, and $\mathrm{unvec}(\boldsymbol{x})$ is the inverse of $\mathrm{vec}(\cdot)$ operation. Finally, $\mathrm{diag}(\boldsymbol{X})$ represents the vector of the main diagonal of a diagonal matrix $\boldsymbol{X}$, while $||\boldsymbol{X}||_2$ the spectral norm of the matrix $\boldsymbol{X}$, $||\boldsymbol{x}||_2$ the Euclidean norm of vector $\boldsymbol{x}$, and $[\boldsymbol{X}]_{a:b,c:d}$ denotes the submatrix of $\boldsymbol{X}$ formed by selecting the rows from index a to b and the columns from index c to d, inclusive.

II. System Model

We consider a narrowband MIMO communication system where the line-of-sight (LoS) between the transmitter and the

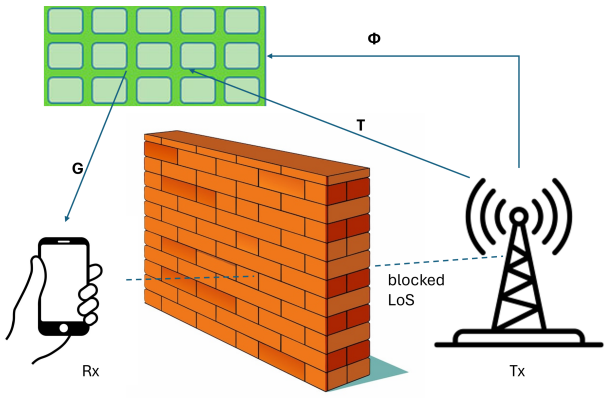

Fig. 1. System model.

receiver is blocked and the link is supported by a passive BD-RIS. The transmitter controls the BD-RIS, as occurs, for example, in the downlink of a cellular system, see Fig. 1.

The transmitter and the receiver have K and M antennas, respectively. The BD-RIS has N elements that emit signals obtained as a linear combination with controllable complex coefficients (as better explained below) of the signal received by all the elements of the BD-RIS. Let the full $N \times N$ matrix $\boldsymbol{\Phi}$ be the BD-RIS configuration, where entry $[\boldsymbol{\Phi}]_{i,j}$ is the complex gain on the signal received by element j and reflected by element i of the BD-RIS.

The baseband-equivalent narrowband matrices of the transmitter-BD-RIS and BD-RIS-receiver channels are $\boldsymbol{T} \in \mathbb{C}^{N\times K}$ and $\boldsymbol{G} \in \mathbb{C}^{M\times N}$, respectively. For a transmitted signal column vector $\boldsymbol{x}$ of K symbols, the M-long received signal column vector is

$$\boldsymbol{y} = \boldsymbol{G}\boldsymbol{\Phi}\boldsymbol{T}\boldsymbol{x} + \boldsymbol{n} = \boldsymbol{H}(\boldsymbol{\Phi})\boldsymbol{x} + \boldsymbol{n}, \tag{1}$$

where $\boldsymbol{n} \sim \mathcal{CN}(\boldsymbol{0}, \sigma^2)$ is the complex additive Gaussian noise vector with independent, zero-mean entries, each with variance σ^2. The resulting achievable rate of the cascade channel is

$$C(\boldsymbol{\Phi}) = \max_{\boldsymbol{R}} \log_2 \det\left(\boldsymbol{I} + \frac{1}{\sigma^2}\boldsymbol{H}(\boldsymbol{\Phi})\boldsymbol{R}\boldsymbol{H}^H(\boldsymbol{\Phi})\right), \tag{2}$$

where $\boldsymbol{R} = E[\boldsymbol{x}^H\boldsymbol{x}]$ is the correlation matrix of the transmit vector signal. Note that $\boldsymbol{R}$ is obtained from the singular value decomposition (SVD) of $\boldsymbol{H}(\boldsymbol{\Phi})$ and applying the waterfilling algorithm on the singular values to satisfy the transmit power constraint, [23].

A. BD-RIS Model

For a passive BD-RIS impedance network, it holds $\boldsymbol{x}^H\boldsymbol{\Phi}^H\boldsymbol{\Phi}\boldsymbol{x} \leq ||\boldsymbol{x}||^2$, i.e., the output power is smaller than or equal to the input power regardless the input signal $\boldsymbol{x}$. This leads to a constraint on the largest eigenvalue of $\boldsymbol{\Phi}^H\boldsymbol{\Phi}$ that must be smaller than or equal to 1, or equivalently $||\boldsymbol{\Phi}||_2^2 \leq 1$. Recent BD-RIS literature [5], [24] focuses on the lossless case, according to which all the incident power is reflected, enforcing the unitary constraint on $\boldsymbol{\Phi}$. In this paper, we assume that the BD-RIS impedance network might include lossy components, and then $\boldsymbol{\Phi}$ is not forced to be unitary, while still satisfying the spectral norm constraint

$$||\boldsymbol{\Phi}||_2^2 \leq 1. \tag{3}$$

Note that such passive BD-RIS can be achieved by considering a passive non-reciprocal network including elements such as passive circulators or isolators [18], [25].

We consider two architectures for the connection of the elements in the BD-RIS: group-connected and fully-connected architectures. For comparison purposes, we also consider the diagonal architecture.

1) Group-Connected Architecture: The N elements of the BD-RIS are connected to an N-port group-connected reconfigurable impedance network [24]. The N ports are uniformly partitioned into G groups, each consisting of $\overline{N} = \frac{N}{G}$ ports connected to each other.[1] From a mathematical standpoint, the BD-RIS matrix is block-diagonal, i.e., $\boldsymbol{\Phi} = \text{blkdiag}(\boldsymbol{\Phi}_1, \ldots, \boldsymbol{\Phi}_G) \in \mathbb{C}^{N\times N}$ where each matrix $\boldsymbol{\Phi}_g$, $g = 1, \ldots, G$, is full and with size $\overline{N} \times \overline{N}$.

2) Fully-Connected Architecture: The fully connected architecture is achieved by connecting each port of the reconfigurable impedance network to all other ports. It can be seen as a special case of the group-connected architecture with $G = 1$ and $\overline{N} = N$.

3) Diagonal Architecture: The diagonal architecture is obtained when there are no connections between the elements. It can be seen as a special case of the group-connected architecture with $G = N$ and $\overline{N} = 1$.

Since both the fully-connected and diagonal architectures are special cases of group-connected architecture, in the following we will consider the general group-connected BD-RIS model.

Remark: The practical feasibility of dynamically reconfigurable coupling in BD-RIS is supported by the recent experimental prototype presented in [26]. Technologies such as PIN diodes allow for controlling the inter-element coupling of a BD-RIS in a realistic, rich-scattering environment, showing measurable performance gains over the conventional diagonal architecture.

B. Transmit Channel State Information

Although the cascade channel depends on both channels to and from the BD-RIS as well as the BD-RIS configuration, it can be fully described by a single effective matrix [24]. Let us define $\mathcal{G} = \{1, \ldots, G\}$ and

$$\boldsymbol{G}_g = [\boldsymbol{G}]_{:,(g-1)\overline{N}+1:g\overline{N}} \in \mathbb{C}^{M\times\overline{N}}, \tag{4a}$$

$$\boldsymbol{T}_g = [\boldsymbol{T}]_{(g-1)\overline{N}+1:g\overline{N},:} \in \mathbb{C}^{\overline{N}\times K}. \tag{4b}$$

From the definition of $\boldsymbol{H}(\boldsymbol{\Phi})$ in (1), we can rewrite the cascade channel as

$$\boldsymbol{H}(\boldsymbol{\Phi}) = \sum_{g\in\mathcal{G}} \boldsymbol{G}_g\boldsymbol{\Phi}_g\boldsymbol{T}_g. \tag{5}$$

Let us define

$$\boldsymbol{Q} = [\boldsymbol{Q}_1, \ldots, \boldsymbol{Q}_G] \in \mathbb{C}^{MK\times\overline{N}^2G} \tag{6a}$$

$$\text{with} \quad \boldsymbol{Q}_g = (\boldsymbol{T}_g^T \otimes \boldsymbol{G}_g) \in \mathbb{C}^{MK\times\overline{N}^2} \tag{6b}$$

$$\boldsymbol{\phi} = [\boldsymbol{\phi}_1^T, .., \boldsymbol{\phi}_G^T]^T \in \mathbb{C}^{\overline{N}N\times 1} \quad \text{with} \quad \boldsymbol{\phi}_g = \text{vec}(\boldsymbol{\Phi}_g), \tag{6c}$$

[1]For a simpler notation, we assume here that N is an integer multiple of G.

then, the vectorial form of the cascade channel matrix is

$$\boldsymbol{h}(\boldsymbol{\Phi}) = \mathrm{vec}[\boldsymbol{H}(\boldsymbol{\Phi})] = \sum_{g\in\mathcal{G}} \boldsymbol{Q}_g \phi_g = \boldsymbol{Q}\boldsymbol{\phi}\,. \tag{7}$$

For a general unstructured channel, a separate estimate of $\boldsymbol{T}_g$ and $\boldsymbol{G}_g$ cannot be obtained, due to an ambiguity of rows or columns of these matrices [27]. From (7) we note that the cascade channel is fully determined by matrix $\boldsymbol{Q}$, which indeed can be easily estimated with conventional MIMO channel estimation techniques [28]. Thus, in the following, we assume that the receiver knows only matrix $\boldsymbol{Q}$. This channel state information (CSI) can be either fed back to the transmitter in uplink, or estimated at the transmitter by swapping the role of the devices (uplink transmission) to obtain transmit CSI.

1) Fully-Connected BD-RIS: In this case we have $G = 1$ and obtain $\boldsymbol{h}(\boldsymbol{\Phi}) = \boldsymbol{Q}\boldsymbol{\phi}'$, with $\boldsymbol{Q} = \boldsymbol{T}^T \otimes \boldsymbol{G}$ and $\boldsymbol{\phi}' = \mathrm{vec}(\boldsymbol{\Phi})$.

2) Diagonal BD-RIS: For $G = N$, we have $\boldsymbol{h}(\boldsymbol{\Phi}) = \boldsymbol{Q}\,\mathrm{diag}(\boldsymbol{\Phi})$, with $\boldsymbol{Q} = \boldsymbol{T}^T \diamond \boldsymbol{G}$.

Remark: We assume ideal CSI at the transmitter to focus on the optimization and control aspects of the BD-RIS configuration. In practical deployments, CSI estimation may be imperfect or partially available due to limited pilot signaling and feedback compression. Efficient channel estimation methods have recently been proposed in [17], [18], [29], and [30] and can be integrated in the configuration design.

III. Randomized Configuration Optimization

We first consider the optimization of the BD-RIS configuration $\boldsymbol{\Phi}$ to maximize the achievable rate of the cascade channel. In formulas, we obtain the following optimization problem

$$\boldsymbol{\Phi}_{\mathrm{opt}} = \arg\max_{\boldsymbol{\Phi}} C(\boldsymbol{\Phi}), \quad \text{s.t. } (3). \tag{8}$$

This problem is non-convex in the objective function and hard to solve, and it also includes, as a hidden variable to be optimized, the correlation matrix $\boldsymbol{R}$. Therefore, we solve the optimization problems of the BD-RIS configuration and the correlation matrix in cascade: first, we optimize the configuration for $\boldsymbol{R} = \boldsymbol{I}$. Then, we explore random configurations close to that found and jointly optimize the correlation matrix $\boldsymbol{R}$ to maximize the achievable rate. Moreover, we consider the alternative problem of maximizing the Frobenius norm of $\boldsymbol{H}(\boldsymbol{\Phi})$, i.e.,

$$\boldsymbol{\Phi}_{\mathrm{TR}} = \arg\max_{\boldsymbol{\Phi}} \mathrm{tr}[\boldsymbol{H}(\boldsymbol{\Phi})^H \boldsymbol{H}(\boldsymbol{\Phi})], \quad \text{s.t. } (3). \tag{9}$$

This problem has already been considered as an alternative to the achievable rate maximization problem, in a multipath MIMO system supported by a RIS with a general architecture and assuming perfect channel knowledge [6]. Indeed, (9) promotes the BD-RIS configurations that increase the sum of the squared singular values of $\boldsymbol{H}$. In formulas, $\boldsymbol{A}$ be a $n \times n$ matrix, then

$$\log_2(1 + \mathrm{tr}(\boldsymbol{A})) \le \log_2 \det(\boldsymbol{I} + \boldsymbol{A}) \le n \log_2(1 + n\mathrm{tr}(\boldsymbol{A}))\,. \tag{10}$$

Thus, increasing $\mathrm{tr}(\boldsymbol{A})$ moves both the upper and lower bound of $\log_2 \det(\boldsymbol{I} + \boldsymbol{A})$ in the proper direction. Note that a joint optimization of the configuration and the correlation matrix should, in general, lead to better performance, but it is not considered here for its entailed computational complexity. Still, in the randomization step, the correlation matrix is updated for each explored configuration; thus, the two variables are jointly optimized.

To further simplify the resolution of (9), we first observe that the objective function in (9) can be rewritten in a quadratic form. Specifically, from (7), it follows that

$$\begin{aligned}
&\mathrm{tr}(\boldsymbol{H}^H(\boldsymbol{\Phi})\boldsymbol{H}(\boldsymbol{\Phi})) = \boldsymbol{h}^H(\boldsymbol{\Phi})\boldsymbol{h}(\boldsymbol{\Phi}) = \\
&= \left(\sum_{g\in\mathcal{G}} [\boldsymbol{T}_g^T \otimes \boldsymbol{G}_g]\boldsymbol{\phi}_g\right)^H \left(\sum_{g\in\mathcal{G}} [\boldsymbol{T}_g^T \otimes \boldsymbol{G}_g]\boldsymbol{\phi}_g\right) \\
&= \sum_{g\in\mathcal{G}}\sum_{j\in\mathcal{G}} \boldsymbol{\phi}_g^H [\boldsymbol{T}_g^T \otimes \boldsymbol{G}_g]^H [\boldsymbol{T}_j^T \otimes \boldsymbol{G}_j]\boldsymbol{\phi}_j \\
&= \sum_{g\in\mathcal{G}}\sum_{j\in\mathcal{G}} \boldsymbol{\phi}_g^H [\boldsymbol{T}_g^* \boldsymbol{T}_j^T \otimes \boldsymbol{G}_g^H \boldsymbol{G}_j]\boldsymbol{\phi}_j \\
&= \sum_{g\in\mathcal{G}}\sum_{j\in\mathcal{G}} \boldsymbol{\phi}_g^H \boldsymbol{M}_{gj}\boldsymbol{\phi}_j,
\end{aligned} \tag{11}$$

with $\boldsymbol{M}_{gj} = [\boldsymbol{T}_g^* \boldsymbol{T}_j^T \otimes \boldsymbol{G}_g^H \boldsymbol{G}_j]$. Note that $\boldsymbol{M}_{gj} = \boldsymbol{Q}_g^H \boldsymbol{Q}_j$, $g, j = 1, \ldots, N$. By using matrix $\boldsymbol{M}_{gj}$ as block (g, j) of matrix $\boldsymbol{M}$

$$\mathrm{tr}(\boldsymbol{H}^H(\boldsymbol{\Phi})\boldsymbol{H}(\boldsymbol{\Phi})) = \boldsymbol{\phi}^H \boldsymbol{M}\boldsymbol{\phi}\,. \tag{12}$$

Note that this can be easily adapted to fully-connected and diagonal BD-RIS by considering

$$\boldsymbol{M} = (\boldsymbol{T}^T \otimes \boldsymbol{G})^H (\boldsymbol{T}^T \otimes \boldsymbol{G}) \tag{13}$$

and $\boldsymbol{\phi} = \boldsymbol{\phi}'$ in the first case, and

$$\boldsymbol{M} = (\boldsymbol{T}^T \diamond \boldsymbol{G})^H (\boldsymbol{T}^T \diamond \boldsymbol{G}), \tag{14}$$

and $\boldsymbol{\phi} = \mathrm{diag}(\boldsymbol{\Phi})$ in the second case.

However, the passive constraint (3) cannot be directly expressed as a quadratic constraint. Then, to obtain a more tractable problem, we introduce a surrogate set of constraints that are weaker than the spectral norm constraint but admit a quadratic representation. Specifically, we impose that the norm of each column of $\boldsymbol{\Phi}$ is less than or equal to 1, i.e.,

$$\sum_{k=1}^{N} |\boldsymbol{\Phi}_{k,j}|^2 \le 1, \quad j = 1, \ldots, N\,. \tag{15}$$

These column-wise constraints are easier to handle but do not imply passivity, because the energy of each column is individually controlled without limiting the overall amplification that may occur across different linear combinations of the columns. Therefore, $\boldsymbol{\Phi}$ can satisfy all column-norm constraints while still having a spectral norm larger than one, introducing then amplification.

Let us define the $(\overline{N} \cdot N) \times (\overline{N} \cdot N)$ matrices $\boldsymbol{A}_n$, $n = 1, \ldots, N$, having $\overline{N}$ ones on the diagonal in positions

$\overline{N}(n-1)+1, \overline{N}(n-1)+2, \ldots, \overline{N}n$, and zeros in all other entries, i.e.,

$$\boldsymbol{A}_n = \begin{bmatrix} \mathbf{0}_{\overline{N}\times\overline{N}} & 0 & \ldots & \ldots & \ldots & 0 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots \\ 0 & \ldots & \boldsymbol{I}_{\overline{N}\times\overline{N}} & 0 & \ldots & 0 \\ 0 & \ldots & \ldots & \ddots & \ldots & 0 \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & \ldots & \ldots & \ldots & 0 & \mathbf{0}_{\overline{N}\times\overline{N}}, \end{bmatrix}, \tag{16}$$

where $\boldsymbol{I}_{\overline{N}\times\overline{N}}$ is the $\overline{N}\times\overline{N}$ identity matrix, $\mathbf{0}_{\overline{N}\times\overline{N}}$ is the $\overline{N}\times\overline{N}$ with all zero entries matrix. Then, constraints (15) can be reformulated in a quadratic form as

$$\boldsymbol{\phi}^H \boldsymbol{A}_n \boldsymbol{\phi} \leq 1 \quad n = 1, \ldots, N\,. \tag{17}$$

The resulting optimization problem can be cast in a quadratic form as

$$\boldsymbol{\phi}_{\mathrm{TR}} = \arg\max_{\boldsymbol{\phi}} \boldsymbol{\phi}^H \boldsymbol{M} \boldsymbol{\phi}, \quad \text{s.t. } (17)\,. \tag{18}$$

Both the objective function and the constraints in (18) are convex functions of $\boldsymbol{\phi}$. However, maximizing a convex function is non-convex. Then, we propose two approaches to solve (18): the SDP- and the projected gradient ascent (PGA)-based techniques.

SDP-Based Solution: Both the objective function and the constraints in (18) can be written as a function of $\boldsymbol{\phi}^H\boldsymbol{\phi}$ as

$$\boldsymbol{\phi}^H \boldsymbol{M} \boldsymbol{\phi} = \mathrm{tr}(\boldsymbol{\phi}^H \boldsymbol{M} \boldsymbol{\phi}) = \mathrm{tr}(\boldsymbol{M} \boldsymbol{\phi}^H \boldsymbol{\phi}) \tag{19a}$$

$$\boldsymbol{\phi}^H \boldsymbol{A}_n \boldsymbol{\phi} = \mathrm{tr}(\boldsymbol{\phi}^H \boldsymbol{A}_n \boldsymbol{\phi}) = \mathrm{tr}(\boldsymbol{A}_n \boldsymbol{\phi}^H \boldsymbol{\phi}). \tag{19b}$$

Therefore, by defining $\tilde{\boldsymbol{\Phi}} = \boldsymbol{\phi}^H\boldsymbol{\phi}$ and noting that $\tilde{\boldsymbol{\Phi}}$ is a positive semi-definite matrix, we obtain the following equivalent formulation of (18)

$$\tilde{\boldsymbol{\Phi}}_{\mathrm{SDP}} = \arg\max_{\tilde{\boldsymbol{\Phi}}} \ \mathrm{tr}(\boldsymbol{M}\tilde{\boldsymbol{\Phi}}) \tag{20a}$$

$$\text{s.t.} \quad \mathrm{tr}(\boldsymbol{A}_n\tilde{\boldsymbol{\Phi}}) \leq 1, \qquad n = 1, \ldots, N, \tag{20b}$$

$$\tilde{\boldsymbol{\Phi}} \succeq 0, \tag{20c}$$

$$\mathrm{rank}(\tilde{\boldsymbol{\Phi}}) = 1\,. \tag{20d}$$

The rank constraint (20d) is non-convex, whereas the objective function and the other constraints are convex in $\tilde{\boldsymbol{\Phi}}$. Thus, by dropping the rank constraint, we obtain a relaxed version of (20) that is an SDP formulation of (18), which can be solved by well-known convex optimization algorithms [31]. The solution of the SDP formulation of (18) may not be feasible for (8) since we drop the rank constraint, we consider the Frobenius norm of $\boldsymbol{H}(\boldsymbol{\Phi})$ as a surrogate objective function, and we relax (3) into (15). A feasible solution of (8) can be obtained by first computing the eigenvalue decomposition $\tilde{\boldsymbol{\Phi}}_{\mathrm{SDP}} = \boldsymbol{W}\boldsymbol{E}\boldsymbol{W}^T$, where $\boldsymbol{W}$ contains the eigenvectors $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_N$, and $\boldsymbol{E}$ is the diagonal matrix of the eigenvalues $\xi_1, \ldots, \xi_N$ of $\tilde{\boldsymbol{\Phi}}_{\mathrm{SDP}}$. Then, we apply a rank-one approximation on $\tilde{\boldsymbol{\Phi}}_{\mathrm{SDP}}$, obtaining $\boldsymbol{\phi}^{(0)} = \sqrt{\xi_1}\boldsymbol{w}_1$, i.e., eigenvector $\boldsymbol{w}_1$ scaled to its eigenvalue ξ_1. A feasible solution of (8) is then

$$\tilde{\boldsymbol{\phi}}^{(0)} = \frac{\boldsymbol{\phi}^{(0)}}{\max\left(\|\,\mathrm{unvec}(\boldsymbol{\phi}^{(0)})\|_2, 1\right)}\,, \tag{21}$$

where the normalization restores the original passive constraint (3).

PGA-Based Solution: A computationally lighter approach to solve (18) is based on the PGA method [32]. The PGA iteratively updates $\boldsymbol{\phi}$ as

$$\boldsymbol{\phi}^{(t+1)} = \mathcal{P}\left(\boldsymbol{\phi}^{(t)} + \mu^{(t)} \nabla f(\boldsymbol{\phi}^{(t)})\right), \tag{22}$$

where $\nabla f(\boldsymbol{\phi})$ is the gradient of the objective function in (18) (i.e., $\nabla f(\boldsymbol{\phi}){=}2\boldsymbol{M}\boldsymbol{\phi}$), $\mu^{(t)}$ is the step size, and $\mathcal{P}(\cdot)$ represents the projection onto the feasible set given by constraints (15). Specifically, the projection normalizes each subvector $\boldsymbol{\phi}_g$ of $\boldsymbol{\phi}$ corresponding to a column of $\boldsymbol{\Phi}$ whenever its Euclidean norm exceeds 1. The iterations continue until convergence or when a maximum number of iterations T is reached, yielding a locally optimal $\boldsymbol{\phi}^{(0)} = \boldsymbol{\phi}^{(T)}$ that satisfies (15). However, a further normalization as that in (21) is needed to enforce (3).

A. Randomized Algorithm for Achievable Rate Maximization

Since the optimization problem (18) does not directly maximize the achievable rate and does not enforce (3) as the passive constraint, both SDP- and PGA-based solutions are refined to improve the resulting capacity. To this end, we resort to a randomized approach [33]. In particular, we generate P random solutions, $\boldsymbol{\phi}^{(p)}$, $p = 1, \ldots, P$. To obtain feasible solutions satisfying (3), we apply a rescaling providing

$$\tilde{\boldsymbol{\phi}}^{(p)} = \frac{\boldsymbol{\phi}^{(p)}}{\max\left(\|\boldsymbol{\Phi}^{(p)}\|_2, 1\right)}\,, \tag{23}$$

with $\boldsymbol{\Phi}^{(p)} = \mathrm{unvec}(\boldsymbol{\phi}^{(p)})$.

Finally, the feasible solution that maximizes the cascade channel achievable rate is selected as the BD-RIS configuration, i.e.,

$$\boldsymbol{\phi}^\star = \underset{\{\tilde{\boldsymbol{\phi}}^{(p)},\, p=0,\ldots,P\}}{\arg\max} \quad C(\tilde{\boldsymbol{\phi}}^{(p)})\,, \quad \boldsymbol{\Phi}^\star = \mathrm{unvec}(\boldsymbol{\phi}^\star). \tag{24}$$

The solutions $\{\boldsymbol{\phi}^{(p)}\}$ are typically drawn from a Gaussian distribution with zero mean and covariance matrix $\tilde{\boldsymbol{\Phi}}_{\mathrm{R}}$ built from the output of the chosen algorithm, i.e., the matrix $\tilde{\boldsymbol{\Phi}}_{\mathrm{SDP}}$ or an empirical covariance matrix $\tilde{\boldsymbol{\Phi}}_{\mathrm{PGA}}$ estimated from the PGA trajectory obtained by taking the sequence of solutions explored during the iterations and computing their sample covariance matrix. However, to explore solutions that are spread out (in terms of Euclidean distance) from $\tilde{\boldsymbol{\phi}}^{(0)}$, we modify the covariance matrix $\tilde{\boldsymbol{\Phi}}_{\mathrm{R}}$ by adding a random component to its eigenvalues (diagonal of $\boldsymbol{E}_{\mathrm{R}}$ obtained from the eigenvalue decomposition $\tilde{\boldsymbol{\Phi}}_{\mathrm{R}} = \boldsymbol{W}_{\mathrm{R}}\boldsymbol{E}_{\mathrm{R}}\boldsymbol{W}_{\mathrm{R}}^T$) as follows

$$\tilde{\boldsymbol{E}}_{\mathrm{R}} = \boldsymbol{E}_{\mathrm{R}} + \boldsymbol{E}_1\,, \tag{25}$$

where $\boldsymbol{E}_1$ is a diagonal matrix such that $\mathrm{diag}(\boldsymbol{E}_1) = \frac{1}{2}\boldsymbol{r}$, with $\boldsymbol{r}$ being a vector whose entries are independently drawn from a normal distribution, i.e., $r_i \sim \mathcal{N}(0,1)$ for $i = 1, \ldots, N$.[2] The perturbed covariance matrix is then reconstructed as

$$\tilde{\boldsymbol{\Phi}}_{\mathrm{R},2} = \boldsymbol{W}_{\mathrm{R}}\tilde{\boldsymbol{E}}_{\mathrm{R}}\boldsymbol{W}_{\mathrm{R}}^T. \tag{26}$$

[2]To ensure that $\tilde{\boldsymbol{E}}$ remains positive semi-definite, we clip any negative eigenvalues to small positive values of the order of 10^{-10}.

Then, we generate P random feasible solutions $\boldsymbol{\phi}^{(p)}$ drawn from a Gaussian distribution with zero mean and covariance matrix $\tilde{\boldsymbol{\Phi}}_{\mathrm{R},2}$.

1) Computational Complexity for SDP-Based Solution: The computational complexity of the SDP-based solution depends primarily on solving the SDP formulation of (18), followed by the eigenvalue decomposition and the randomized refinement. The SDP formulation involves optimizing over a positive semidefinite matrix $\tilde{\boldsymbol{\Phi}}$ of size $(\overline{N}N) \times (\overline{N}N)$, yielding a problem with $(\overline{N}N)^2$ variables. The computational complexity of solving the SDP scales as $\mathcal{O}((\overline{N}N)^6)$. The eigenvalue decomposition of $\tilde{\boldsymbol{\Phi}}_{\mathrm{SDP}}$ requires $\mathcal{O}((\overline{N}N)^3)$ operations. The randomization procedure requires first the perturbation of the $\tilde{\boldsymbol{\Phi}}_{\mathrm{SDP}}$ eigenvalues and then the generation of P random solutions. The former incurs a complexity of $\mathcal{O}((\overline{N}N)^3)$, while the latter of $\mathcal{O}(P(\overline{N}N)^3)$, due to the SVD for the rate computation. Summing all these contributions, the overall complexity is dominated by the SDP solution, with order of $\mathcal{O}((\overline{N}N)^6)$.

2) Computational Complexity for PGA-Based Solution: The computational complexity of the PGA-based solution is mainly due to the iterative gradient updates, the projection step, and the randomized refinement. At each iteration, the gradient cost is $\mathcal{O}((\overline{N}N)^2)$, while the projection cost is $\mathcal{O}(\overline{N}N)$. Assuming a total of T iterations, the overall complexity of the iterative phase is $\mathcal{O}(T(\overline{N}N)^2)$, which is significantly lower than the complexity of solving the SDP formulation of (18), making PGA more attractive for large-scale BD-RIS configurations. The randomization procedure has the same computational complexity as in the SDP-based solution, since it still requires generating P random candidate solutions and evaluating their achievable rates through SVD computations. Therefore, the overall complexity of the PGA-based solution is due to the terms $\mathcal{O}(T(\overline{N}N)^2)$ and $\mathcal{O}(P(\overline{N}N)^3)$, where the second term is typically dominating.

IV. Reduced BD-RIS Parameter Vector

Although the solutions described in the previous section can be applied to determine the proper BD-RIS configuration, both approaches operate on the full parameter vector $\boldsymbol{\phi}$ whose dimension $\overline{N}N$ grows rapidly with the number of BD-RIS elements N. This growth increases the computational burden of randomization and feasibility checks of the provided solutions, and also enlarges the amount of information that must be transmitted over the control link. In fact, a larger BD-RIS also requires a higher rate control link from the transmitter to the BD-RIS to pass all parameters $\boldsymbol{\phi}$.

To address these challenges, we introduce a reduced BD-RIS parameter vector by mapping the full configuration vector onto a lower-dimensional representation. This compact representation aims at capturing a subset of BD-RIS elements that have the largest impact on the system performance, while reducing control overhead and enabling the configuration optimization in a reduced space. Specifically, we consider only a subspace of the BD-RIS parameter vector, represented by $c \leq \overline{N}N$ variables $\overline{\boldsymbol{\phi}} = [\overline{\phi}_1, \ldots, \overline{\phi}_c] \in \mathbb{C}^c$, obtained with suitable mapping function from the solution $\boldsymbol{\phi}^\star$, i.e.,

$$\overline{\boldsymbol{\phi}} = f(\boldsymbol{\phi}^\star) \in \mathbb{C}^c\,. \tag{27}$$

The mapping function is tailored to the specific device position and propagation environment where the BD-RIS is deployed. Indeed, depending on the typical user positions, the most relevant BD-RIS elements have the greatest impact on properly shaping the wireless channel, which is governed by the device positions and electromagnetic properties of the environment. The mapping function then captures these complex dependencies, approximating $\boldsymbol{\phi}^\star$ with c parameters, enabling an environment-aware compact representation that reduces the control overhead with limited impact on the performance.

Similarly, we define a de-mapping function that maps the reduced space of dimension c back to the original space of dimension $\overline{N}N$, reconstructing the complete BD-RIS configuration from its compressed representation as

$$\hat{\boldsymbol{\phi}}^\star = f'(\overline{\boldsymbol{\phi}}) \in \mathbb{C}^{\overline{N}N}\,, \quad \hat{\boldsymbol{\Phi}}^\star = \mathrm{unvec}(\hat{\boldsymbol{\phi}}^\star). \tag{28}$$

Remark 1: The mapping function $f(\cdot)$ is not invertible, thus applying the de-mapping $f'(\cdot)$ we do not obtain again the original BD-RIS configuration and in general $\boldsymbol{\phi}^\star \neq \hat{\boldsymbol{\phi}}^\star$.

Remark 2: The de-mapping function (28) provides the effective phase shifts set on the BD-RIS. Therefore, we must ensure that the de-mapping satisfies constraints (15).

The use of mapping and de-mapping functions allows for an implementation of the BD-RIS control with a limited data rate. In this implementation the $f'(\cdot)$ is implemented at the BD-RIS, and over the control channel only the compressed representation $\overline{\boldsymbol{\phi}}$ of size c is transmitted to the BD-RIS, instead of $\boldsymbol{\phi}$ of size $\overline{N}N$.

A. Mapping and De-Mapping by Autoencoder

To design the mapping and de-mapping functions, we resort to a machine-learning approach, where first we frame the mapping and de-mapping functions into the decoder and encoder of an AE model, and then we detail a training process to learn the coefficients.

The AE consists of an encoder and a decoder, where the encoder compresses the input data into a lower-dimensional representation (i.e., latent representation), and the decoder reconstructs the input from this compressed representation. Indeed, the layers of the AE learn an effective latent-space representation of the input data and reconstruct it efficiently.

The architecture of the encoder includes L layers, each in turn composed of B_ℓ neurons, $\ell \in \{1, \ldots, L\}$, typically followed by a non-linear activation function $a_{\mathrm{enc},\ell}(\cdot)$. Specifically, the $B_\ell \times 1$ vector $\boldsymbol{v}_\ell$ is the input of layer ℓ, where B_ℓ is the input dimension of layer ℓ. Since the encoder compresses the input data, the output of each layer typically has a smaller dimension than its input. These dimensions can be adjusted as hyperparameters to control the desired level of compression and information loss. For the first layer ($\ell = 1$) we have $\boldsymbol{v}_1 = \boldsymbol{\phi}^\star \in \mathbb{C}^{\overline{N}N}$.

The output layer $1 \leq \ell \leq L$ is computed as

$$\boldsymbol{v}_{\ell+1} = a_{\mathrm{enc},\ell}(\boldsymbol{C}_{\mathrm{enc},\ell}\boldsymbol{v}_\ell + \boldsymbol{b}_{\mathrm{enc},\ell}), \tag{29}$$

where $\boldsymbol{C}_{\mathrm{enc},\ell}$ (size $B_{\ell+1} \times B_\ell$) and $\boldsymbol{b}_{\mathrm{enc},\ell}$ (size $B_{\ell+1} \times 1$) are the trainable encoder weights matrix and bias vector, respectively.

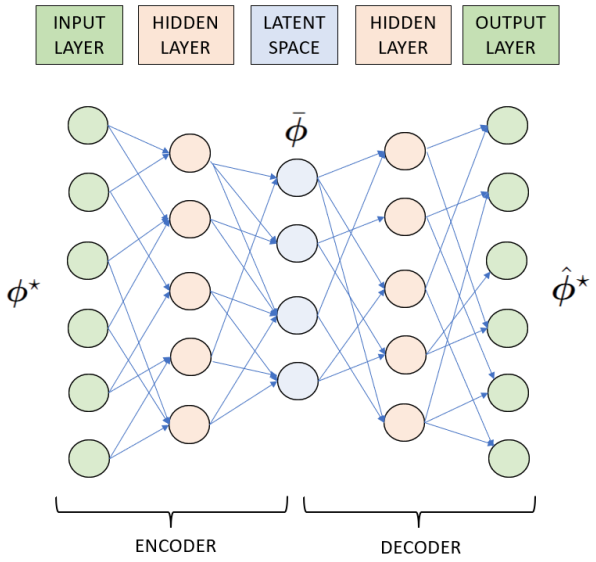

Fig. 2. Example of the considered autoencoder, with $L = 2$, $\overline{N}N = 6$, and $c = 4$.

The output of the last layer ($\ell = L$) provides the latent representation

$$\bar{\phi} = \boldsymbol{v}_{L+1}\,. \tag{30}$$

Note that $c \leq \overline{N}N$.[3]

The decoder aims at reconstructing the original input ϕ_{SDP} from the latent representation $\bar{\phi}$ through L layers that progressively expand its dimensionality. In this case, the output dimensions of each layer are higher than the corresponding input dimensions. Specifically, the decoder architecture mirrors that of the encoder, where each layer has B'_ℓ neurons, $\ell \in \{1,\dots,L\}$, and is typically followed by a non-linear activation function $a_{\text{dec},\ell}$. The input to the first layer of the decoder ($\ell = 1$) is the latent representation, i.e., $\boldsymbol{v}'_1 = \bar{\phi} \in \mathbb{C}^c$. The output of the layer $1 \leq \ell \leq L$ is

$$\boldsymbol{v}'_{\ell+1} = a_{\text{dec},\ell}(\boldsymbol{A}_{\text{dec},\ell}\boldsymbol{v}'_\ell + \boldsymbol{b}_{\text{dec},\ell}), \tag{31}$$

where $\boldsymbol{A}_{\text{dec},\ell}$ (size $B'_{\ell+1}\times B'_\ell$) and $\boldsymbol{b}_{\text{dec},\ell}$ (size $B'_{\ell+1}\times 1$) are the trainable decoder weights matrix and bias vector parameters, respectively. The activation function $a_{\text{dec},\ell}(\cdot)$, $\ell = 1,\dots,L-1$, is typically chosen to be the same as in the encoder for intermediate layers. The final layer ($\ell = L$) provides the reconstructed vectorized BD-RIS configuration

$$\hat{\phi}^\star = \boldsymbol{v}'_{L+1}\,. \tag{32}$$

To ensure that the used BD-RIS configuration satisfies the constraints (3), we consider an approach based on the decoder output normalization described in the remainder of this section. Fig. 2 depicts the general autoencoder structure.

B. Decoder Output Normalization (DON)

The activation function of the last layer enforces constraint (3). In particular, let $\boldsymbol{v}'' = \boldsymbol{A}_{\text{dec},L}\boldsymbol{v}'_L + \boldsymbol{b}_{\text{dec},L}$. After having converted $\boldsymbol{v}''$ into matrix form as $\hat{\boldsymbol{\Phi}}'' = \text{unvec}(\boldsymbol{v}'')$, we normalize it to obtain

$$\hat{\boldsymbol{\Phi}}^\star = \frac{\hat{\boldsymbol{\Phi}}''}{\max\left(\|\hat{\boldsymbol{\Phi}}''\|_2, 1\right)}\,. \tag{33}$$

The reconstructed vectorized BD-RIS configuration is then $\hat{\phi}^\star = \text{vec}(\hat{\boldsymbol{\Phi}}^\star)$.

C. Training Loss Function

To train the AE, we consider two loss functions to be minimized: the mean square error and the negative achievable rate.

1) Mean Square Error: The MSE loss function is

$$\text{MSE}(\phi^\star, \hat{\phi}^\star) = \mathbb{E}\left[||\hat{\phi}^\star - \phi^\star||^2\right], \tag{34}$$

where the expectation is taken on the dataset of $\phi^\star$.

2) Negative Achievable Rate: In this case, we directly aim at maximizing the achievable rate obtained with the selected BD-RIS configuration. Thus, the loss function (to be minimized) is the opposite of the achievable rate (2), i.e.,

$$\rho(\hat{\phi}^\star) = \mathbb{E}\left[-\min_{\boldsymbol{R}} \log_2\det\left(\boldsymbol{I} + \right.\right.$$
$$\left.\left. + \frac{1}{\sigma^2}\boldsymbol{H}(\hat{\boldsymbol{\Phi}}^\star)\boldsymbol{R}\boldsymbol{H}^H(\hat{\boldsymbol{\Phi}}^\star)\right)\right]. \tag{35}$$

Training with the negative achievable rate as a loss function entails a significantly higher computational complexity than MSE. Indeed, to obtain the rate for a specific channel and configuration, we need to optimize $\boldsymbol{R}$, 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 AE loss function. Note that the training should be performed whenever the channel conditions change significantly.

Remark: When the AE is trained offline using the achievable rate as loss, perfect CSI is assumed during training; therefore, deploying the decoder with imperfect or quantized CSI may introduce a mismatch that results in rate degradation. Conversely, if the AE was trained or periodically fine-tuned online using the same imperfect CSI available at deployment, the learned mapping may in part adapt to the CSI errors and yield improved robustness.

V. Optimization In The Reduced Space

We now propose to optimize the BD-RIS configuration in the latent space to reduce the complexity of the proposed solution. Such a solution provides vector $\bar{\phi}$ to be passed to the BD-RIS. Moreover, by optimizing directly $\bar{\phi}$, we better control the effects of BD-RIS parameter modification introduced by the reduction to the latent space.

To reformulate the optimization problem (18) in the latent space, the constraint (28) is added, obtaining

$$\bar{\phi}_{\text{TR}} = \arg\max_{\bar{\phi}} \phi^H\boldsymbol{M}\phi\,, \tag{36a}$$

$$\text{s.t.} \quad \phi^H\boldsymbol{A}_n\phi \leq 1\,, n = 1,\dots,N\,, \tag{36b}$$

$$\phi = f'(\bar{\phi})\,. \tag{36c}$$

[3]In a practical implementation, the real and imaginary parts of ϕ are processed separately by neurons having only real coefficients, and the non-linear function operates on the two components separately.

Note that the problem also includes constraints (15) on the decompressed vector $\boldsymbol{\phi}$.

To apply the solutions described in Section III to problem (36), it is necessary that the de-mapping function $f'(\cdot)$ is linear. This is because both solutions rely on the problem being expressed as a quadratic form in the optimization variables. If $f'(\cdot)$ is linear, then the objective and constraint functions remain quadratic in the new variable $\overline{\boldsymbol{\phi}}$, allowing the standard lifting procedure to a matrix variable and the subsequent convex relaxation. Conversely, if $f'(\cdot)$ is non-linear, the objective and constraints would no longer be quadratic, and the approaches discussed in Section III could not be applied directly. We focus then on a *linear AE*, i.e., an AE where the encoder and the decoder have each a single layer ($L = 1$), without activation functions that provide the following linear mapping and demapping functions

$$\overline{\boldsymbol{\phi}} = \boldsymbol{C}_{\text{enc},1}\boldsymbol{\phi}^{\star} + \boldsymbol{b}_{\text{enc},1}\,, \tag{37}$$

and

$$\hat{\boldsymbol{\phi}}^{\star} = \boldsymbol{A}_{\text{dec},1}\overline{\boldsymbol{\phi}} + \boldsymbol{b}_{\text{dec},1}\,, \tag{38}$$

respectively. Note that a linear AE consisting of a single linear encoder and decoder, trained with MSE loss function, is mathematically equivalent to the principal component analysis (PCA), [34].

Defining

$$\begin{aligned}
&\boldsymbol{M}_1 = \boldsymbol{A}_{\text{dec}}^H \boldsymbol{M} \boldsymbol{A}_{\text{dec}} \in \mathbb{C}^{c\times c}, \quad \boldsymbol{M}_2 = \boldsymbol{A}_{\text{dec}}^H \boldsymbol{M}\boldsymbol{b} \in \mathbb{C}^{c\times 1},\\
&\boldsymbol{M}_3 = \boldsymbol{b}^H \boldsymbol{M} \boldsymbol{A}_{\text{dec}} \in \mathbb{C}^{1\times c}, \quad M_4 = \boldsymbol{b}^H \boldsymbol{M}\boldsymbol{b} \in \mathbb{C},\\
&\boldsymbol{A}_n^{(1)} = \boldsymbol{A}_{\text{dec}}^H \boldsymbol{A}_n \boldsymbol{A}_{\text{dec}} \in \mathbb{C}^{c\times c}, \quad \boldsymbol{A}_n^{(2)} = \boldsymbol{A}_{\text{dec}}^H \boldsymbol{A}_n \boldsymbol{b} \in \mathbb{C}^{c\times 1},\\
&\boldsymbol{A}_n^{(3)} = \boldsymbol{b}^H \boldsymbol{A}_n \boldsymbol{A}_{\text{dec}} \in \mathbb{C}^{1\times c}, \quad A_n^{(4)} = \boldsymbol{b}^H \boldsymbol{A}_n \boldsymbol{b} \in \mathbb{C},
\end{aligned} \tag{39}$$

where we set, to simplify the notation, $\boldsymbol{A}_{\text{dec}} = \boldsymbol{A}_{\text{dec},1} \in \mathbb{C}^{\overline{N}N\times c}$ and $\boldsymbol{b} = \boldsymbol{b}_{\text{dec},1} \in \mathbb{C}^{\overline{N}N\times 1}$, problem (36) can be then reformulated as

$$\overline{\boldsymbol{\phi}}_{\text{TR}} = \arg\max_{\overline{\boldsymbol{\phi}}} \overline{\boldsymbol{\phi}}^H \boldsymbol{M}_1 \overline{\boldsymbol{\phi}} + \overline{\boldsymbol{\phi}}^H \boldsymbol{M}_2 + \boldsymbol{M}_3 \overline{\boldsymbol{\phi}} \tag{40a}$$

$$\text{s.t.} \quad \overline{\boldsymbol{\phi}}^H \boldsymbol{A}_n^{(1)} \overline{\boldsymbol{\phi}} + \overline{\boldsymbol{\phi}}^H \boldsymbol{A}_n^{(2)} + \boldsymbol{A}_n^{(3)} \overline{\boldsymbol{\phi}} + A_n^{(4)} \le 1\,, \quad n = 1,\dots,N. \tag{40b}$$

Note that we are ignoring $\boldsymbol{M}_4$ in the objective function due to its independence from $\overline{\boldsymbol{\phi}}$.

A. SDP-Based Solution

The quadratic problem in (40) can be put in a conventional form suitable for SDP relaxation as follows. First, we define

$$\overline{\boldsymbol{R}} = \begin{bmatrix} \boldsymbol{M}_1 & \boldsymbol{M}_2 \\ \boldsymbol{M}_3 & 0 \end{bmatrix}, \quad \overline{\boldsymbol{A}}_{1,n} = \begin{bmatrix} \boldsymbol{A}_n^{(1)} & \boldsymbol{A}_n^{(2)} \\ \boldsymbol{A}_n^{(3)} & A_n^{(4)} \end{bmatrix},$$
$$\overline{\boldsymbol{A}}_2 = \begin{bmatrix} \boldsymbol{0}_{c\times c} & \boldsymbol{0}_{c\times 1} \\ \boldsymbol{0}_{1\times c} & 1 \end{bmatrix}, \quad \boldsymbol{\psi} = \begin{bmatrix} \overline{\boldsymbol{\phi}} \\ t \end{bmatrix}, \tag{41}$$

and then we rewrite (40) as

$$\boldsymbol{\psi}_{\text{TR}} = \arg\max_{\boldsymbol{\psi}} \boldsymbol{\psi}^H \overline{\boldsymbol{R}} \boldsymbol{\psi} \tag{42a}$$

$$\text{s.t.} \quad \boldsymbol{\psi}^H \overline{\boldsymbol{A}}_{1,n} \boldsymbol{\psi} \le 1\,, \quad n = 1,\dots,N, \tag{42b}$$

$$\boldsymbol{\psi}^H \overline{\boldsymbol{A}}_2 \boldsymbol{\psi} \le 1, \quad |t| = 1\,. \tag{42c}$$

By applying an SDP relaxation and removing (once again) the rank-one constraint, we obtain the convex optimization problem

$$\boldsymbol{\Psi}_{\text{SDP}} = \arg\max_{\tilde{\boldsymbol{\Psi}}} \ \text{tr}(\overline{\boldsymbol{R}}\tilde{\boldsymbol{\Psi}}) \tag{43a}$$

$$\text{s.t.} \quad \text{tr}(\overline{\boldsymbol{A}}_{1,n}\tilde{\boldsymbol{\Psi}}) \le 1, \quad n = 1,\dots,N, \tag{43b}$$

$$\text{tr}(\overline{\boldsymbol{A}}_2\tilde{\boldsymbol{\Psi}}) \le 1\,, \tag{43c}$$

$$\tilde{\boldsymbol{\Psi}} \succeq 0\,, \tag{43d}$$

which can still be solved by well-known convex optimization algorithms [31]. To obtain a feasible solution of (40) from the solution of the SDP-relaxed problem (43), we first compute the eigenvalue decomposition $\boldsymbol{\Psi}_{\text{SDP}} = \boldsymbol{U}\boldsymbol{D}\boldsymbol{U}^T$, where $\boldsymbol{U}$ contains the eigenvectors $\{\boldsymbol{u}_1,\dots,\boldsymbol{u}_{c+1}\}$ and $\boldsymbol{D}$ is a diagonal matrix of the eigenvalues $\{d_1,\dots,d_{c+1}\}$. Then, we apply a rank-one approximation on $\boldsymbol{\Psi}_{\text{SDP}}$, defining $\hat{\boldsymbol{z}}^{(0)} = \sqrt{d_1}[\boldsymbol{u}_1]_{1:c}$, which is a scaled version of the first c elements of the eigenvector $\boldsymbol{u}_1$ corresponding to the largest eigenvalue d_1 of $\boldsymbol{\Psi}_{\text{SDP}}$. We then compute

$$\hat{\boldsymbol{\phi}}^{(0)} = \boldsymbol{A}_{\text{dec}}\hat{\boldsymbol{z}}^{(0)} + \boldsymbol{b}\,, \tag{44}$$

which is then normalized (similarly to (21)) to restore the original passive constraint (3). Finally, we apply the randomization described in Section III-A.

B. PGA-Based Solution

Alternatively, (40) can also be solved by using the PGA method as discussed in Section III. The PGA iteratively updates $\bar{\boldsymbol{\phi}}$ as

$$\bar{\boldsymbol{\phi}}^{(t+1)} = \mathcal{P}\left(\bar{\boldsymbol{\phi}}^{(t)} + \mu^{(t)}\nabla f(\bar{\boldsymbol{\phi}}^{(t)})\right), \tag{45}$$

where $\nabla f(\bar{\boldsymbol{\phi}})$ is the gradient of the objective function (40a) computed with respect to the complex conjugate of $\bar{\boldsymbol{\phi}}$, $\mu^{(t)}$ is the step size, and $\mathcal{P}(\cdot)$ denotes the projection that enforces (40b). Since (40b) are quadratic constraints, for a more practical and scalable approach, the projection can be done by introducing a penalty term in the objective that discourages constraint violations. After convergence, we apply the de-mapping function (38) on the obtained solution as previously discussed for the SDP solution in (44). Finally, we perform the randomization as detailed in Section III-A.

C. Complexity for BD-RIS Optimization in Latent Space

When considering the optimization in the latent space, the number of variables to optimize is reduced from $\overline{N}N$ to c, the latent space dimension. For the SDP-based solution, the complexity of solving the semidefinite relaxation scales as $\mathcal{O}(c^6)$, significantly improving upon the original complexity of $\mathcal{O}((\overline{N}N)^6)$, if $c \ll \overline{N}N$. Furthermore, the eigenvalue decomposition of $\boldsymbol{\Psi}_{\text{SDP}}$ also benefits from the reduced dimensionality, resulting in $\mathcal{O}(c^3)$. However, due to the reconstruction of the feasible solution into the original space via (44), the randomization procedure is still $\mathcal{O}(P(\overline{N}N)^3)$. Summing these contributions, the overall complexity of the SDP-based solution in the latent space is mainly

TABLE I
MAIN PARAMETERS OF THE AE

Symbol	Description
$\boldsymbol{\phi}^\star$	Target cfg. (opt sol in original space.)
$f(\cdot)$, $f'(\cdot)$	AE encoder / decoder maps.
$\boldsymbol{C}_{\text{enc}}$, $\boldsymbol{A}_{\text{dec}}$	Encoder / decoder weights
$\boldsymbol{b}_{\text{enc}}$, $\boldsymbol{b}_{\text{dec}}$	Encoder / decoder bias
$\overline{\boldsymbol{\phi}}=f(\boldsymbol{\phi}^\star)$	Latent repr. (of length $c<\overline{N}N$)
$\hat{\boldsymbol{\phi}}^\star=f'(\overline{\boldsymbol{\phi}})$	Reconstructed cfg. via AE
$\text{MSE}(\boldsymbol{\phi}^\star,\hat{\boldsymbol{\phi}}^\star)$, $\rho(\hat{\boldsymbol{\phi}}^\star)$	MSE and rate-based losses

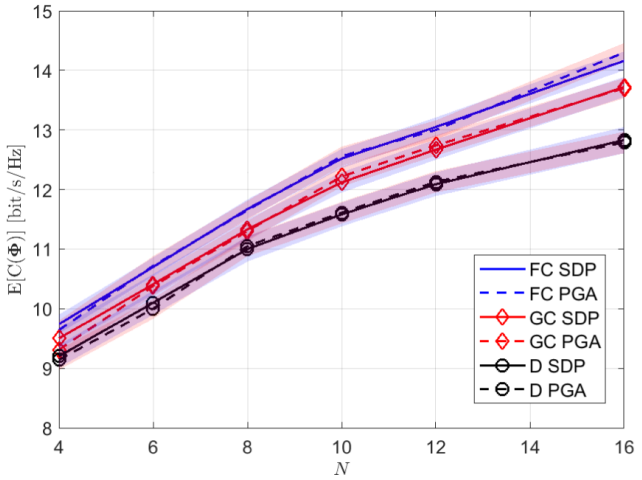

Fig. 3. Average achievable rate for SDP and PGA solution as a function of the number of BD-RIS elements for SNR=10 dB with $P=0$.

due to the terms $\mathcal{O}(c^6)$ and $\mathcal{O}(P(\overline{N}N)^3)$. For the PGA-based solution, the gradient computation and projection steps are performed now with complexity $\mathcal{O}(TNc^2)$, where T is still the number of iterations. The randomization step again dominates the total complexity, as it still depends on the original configuration size, yielding a total complexity of $\mathcal{O}(TNc^2+P(\overline{N}N)^3)$, where the second term is typically dominating. Hence, the PGA-based solution remains considerably lighter than the SDP-based one, particularly for large-scale BD-RIS configurations or small latent dimensions c.

Note that Table I summarizes the notation introduced in Sections IV and V.

VI. NUMERICAL RESULTS

In this section, we evaluate and discuss the proposed algorithm's performance for optimizing the BD-RIS configuration. We first describe the considered scenario, then we detail the compared solutions used for the fully-connected and diagonal architectures. Lastly, we also discuss the impact of different system parameters on the performance of the proposed solutions.

In obtaining numerical results, we have considered both the SDP and PGA solutions. Since the performance (in terms of capacity) obtained by both approaches is the same (see Fig. 3 where the shaded regions indicate the confidence interval), we do not report them separately. In Section VI-F, we offer a comparison in terms of complexity between the two approaches. Therefore, in the following, we refer to randomized configuration optimization (RCO) and RCO-LS as the BD-RIS optimization obtained in the original and latent space, respectively, regardless of the used method.

A. Channel Model and Parameter Setting

Signals are transmitted in the millimeter-wave (mmWave) band. We assume that both transmitter-BD-RIS and BD-RIS-receiver links exhibit a multipath condition, while the direct link between transmitter and receiver is obstructed. All devices (including BD-RIS) have uniform linear arrays (ULAs) (or elements). With reference to the downlink transmission of a cellular system, the transmitter (the base station) and the receiver (the user equipment) are each equipped with an ULA array of $K=6$ and $M=2$ antennas, respectively.

Since channels in the mmWave band have only a few relevant paths, we use a geometric model to describe them. We define the array response column vector of size N for AoA α as

$$\boldsymbol{s}_N(\alpha)=\left[1,e^{-j\frac{2\pi}{\lambda_c}d\sin\alpha},\ldots,e^{-j(N-1)\frac{2\pi}{\lambda_c}d\sin\alpha}\right]^T, \quad (46)$$

where λ_c is the carrier wavelength and $d=\lambda_c/2$ is the spacing between the BD-RIS elements and the antenna arrays at the transmitter and receiver [35].

We consider a channel model to and from the BD-RIS with Z paths and define the Z-paths array response matrix with AoA angles $\boldsymbol{\alpha}=[\alpha_1,\ldots,\alpha_Z]^T$ as

$$\boldsymbol{S}_i(\boldsymbol{\alpha})=[\boldsymbol{s}_i(\alpha_1),\ldots,\boldsymbol{s}_i(\alpha_Z)], \quad (47)$$

where i is the number of antennas or BD-RIS elements.

Let $\varphi_{T,l}$, $\theta_{T,l}$, and $\gamma_{T,l}$ be the AoD at the transmitter, the AoA at the BD-RIS, and the complex path gain, respectively, for the path $z=1,\ldots,Z$. Let us also define $\boldsymbol{\varphi}_F=[\varphi_{T,1},\ldots,\varphi_{T,Z}]^T$ and $\boldsymbol{\theta}_T=[\theta_{T,1},\ldots,\theta_{T,Z}]^T$. Moreover, $\boldsymbol{S}_K^H(\boldsymbol{\varphi}_T)$, $\boldsymbol{S}_N(\boldsymbol{\theta}_T)$, and $\boldsymbol{\Gamma}_T=\text{diag}([\gamma_{T,1},\ldots,\gamma_{T,Z}]^T)$ are the transmitter array response matrix, the BD-RIS array response matrix, and the diagonal path gain matrix, respectively. We model the baseband channel matrix between the transmitter and the BD-RIS as

$$\begin{aligned}\boldsymbol{T}&=\sqrt{\frac{KN}{Z}}\sum_{l=1}^{Z}\gamma_{T,l}\mathbf{s}_N(\theta_{T,l})\mathbf{s}_K^H(\varphi_{T,l})\\&=\boldsymbol{S}_N(\boldsymbol{\theta}_T)\boldsymbol{\Gamma}_F\boldsymbol{S}_K^H(\boldsymbol{\varphi}_T)\in\mathbb{C}^{N\times K}.\end{aligned} \quad (48)$$

All angles are randomly generated according to a uniform distribution in the interval $[0,\pi]$, while path gains are zero-mean and unitary variance independent complex Gaussian variables. We consider $Z=2$ paths, unless otherwise specified.

Similarly, we model the matrix of the channel between the BD-RIS and the receiver as

$$\begin{aligned}\boldsymbol{G}&=\sqrt{\frac{NM}{Z}}\sum_{l=1}^{Z}\gamma_{G,l}\mathbf{s}_M(\theta_{G,l})\mathbf{s}_N^H(\varphi_{G,l})\\&=\boldsymbol{S}_M(\boldsymbol{\theta}_G)\boldsymbol{\Gamma}_G\boldsymbol{S}_N^H(\boldsymbol{\varphi}_G)\in\mathbb{C}^{M\times N},\end{aligned} \quad (49)$$

where angles vectors $\boldsymbol{\theta}_G$ and $\boldsymbol{\varphi}_G$ are now those for the paths between the BD-RIS and the receiver.

B. Compared Solutions

We consider the following solutions for comparison.

1) Fully-Connected BD-RIS With Normalized Dominant Eigenvector (FC-NDE): In [36] it is proposed to optimize the BD-RIS with fully-connected architecture by solving the max-trace problem (18) and constraint (3) replaced by the power constraint $\mathrm{tr}(\boldsymbol{\Phi}^H\boldsymbol{\Phi}) = N$. This constraint is less restrictive since BD-RIS may amplify the received signal before reflection. The FC-NDE solution is a reshaped and scaled version of the eigenvector $\boldsymbol{m}$ corresponding to the largest eigenvalue of $\boldsymbol{M}$ defined in (13), i.e., $\boldsymbol{\Phi}_{\mathrm{FC-NNDE}} = \sqrt{N}\mathrm{unvec}(\boldsymbol{m})$. However, for a fair comparison with the proposed solutions, we normalize $\boldsymbol{\Phi}_{\mathrm{FC-NNDE}}$ so that (3) holds. The complexity of FC-NDE is $\mathcal{O}(N^6)$, due to the eigenvalue decomposition.

2) Fully-Connected BD-RIS With Unitary Constraint (FC-UC): In this case the BD-RIS is optimized as proposed in [14], where SVD is applied to $\boldsymbol{G}$ and $\boldsymbol{T}$ as $\boldsymbol{G} = \boldsymbol{U}_G\boldsymbol{\Sigma}_G\boldsymbol{V}_G^H$ and $\boldsymbol{T} = \boldsymbol{U}_T\boldsymbol{\Sigma}_T\boldsymbol{V}_T^H$, respectively, with the constraint $\boldsymbol{\Phi}\boldsymbol{\Phi}^H = \boldsymbol{I}_N$. Such a constraint not only ensures that $\boldsymbol{\Phi}$ does not amplify the reflected signal but also captures the lossless case, where *all* the incident power is reflected. By diagonalizing the cascade channel matrix, it is proved that the channel achievable rate is achieved by using the configuration of the fully connected BD-RIS

$$\boldsymbol{\Phi}_{\mathrm{FC-UC}} = \boldsymbol{V}_G\boldsymbol{U}_T^H. \tag{50}$$

Note that this solution represents the optimal solution of the original (non-relaxed) problem (8) for a fully-connected BD-RIS. The complexity of FC-UC is $\mathcal{O}(N^3)$, due to the SVD.

3) Diagonal BD-RIS With Normalized Dominant Eigenvector (D-NDE): In [36] it is also proposed to optimize the BD-RIS with diagonal architecture by solving max-trace problem (18) with constraint $\mathrm{tr}(\boldsymbol{\Phi}^H\boldsymbol{\Phi}) = N$. Let $\boldsymbol{g}_i$ be the i-th column of $\boldsymbol{G}$ and $\boldsymbol{t}_i^H$ as the i-th row of $\boldsymbol{T}^H$. Let also define $[\boldsymbol{K}]_{ji} = \boldsymbol{g}_j^H\boldsymbol{g}_i\boldsymbol{t}_i^H\boldsymbol{t}_j$. The D-DE solution $\boldsymbol{\Phi}_{\mathrm{D-NDE}}$ is a diagonal matrix whose diagonal $\boldsymbol{\phi}_{\mathrm{D-NDE}}$ is a scaled version of the eigenvector $\boldsymbol{v}$ of $\boldsymbol{K}$ corresponding to the largest eigenvalue, i.e., $\boldsymbol{\phi}_{\mathrm{D-NDE}} = \mathrm{diag}(\boldsymbol{\Phi}_{\mathrm{D-NDE}}) = \sqrt{N}\boldsymbol{v}$. Also in this case, we normalize $\boldsymbol{\Phi}_{\mathrm{D-NDE}}$ so that (3) holds. The complexity of D-NDE is still $\mathcal{O}(N^6)$, due to the eigenvalue decomposition.

4) Diagonal BD-RIS With Low-Complexity (D-LC): In [37], it is proposed to optimize the BD-RIS with a diagonal architecture. Let us define the following vectors related to $\boldsymbol{g}_n$ and $\boldsymbol{t}_n$ as

$$\tilde{\boldsymbol{g}}_n = [|g_{1,n}|, \ldots, |g_{M,n}|]^T,\ \tilde{\boldsymbol{t}}_n^H = [|t_{n,1}|, \ldots, |t_{n,K}|], \tag{51}$$

for $n = 1, 2, \ldots, N$. The low-complexity solution for passive diagonal BD-RIS is a diagonal matrix $\boldsymbol{\Phi}_{\mathrm{D-LC}}$ whose diagonal $\boldsymbol{\phi}_{\mathrm{D-LC}} = \mathrm{diag}(\boldsymbol{\Phi}_{\mathrm{D-LC}}) = [e^{j\rho_{LC,1}}, \ldots, e^{j\rho_{LC,N}}]$ is obtained from [37] as

$$\rho_{LC,n} = -\cos^{-1}\left(\frac{\mathrm{Re}(\tilde{\boldsymbol{t}}_n^H\boldsymbol{t}_n)}{||\boldsymbol{t}_n|| \cdot ||\tilde{\boldsymbol{t}}_n||}\right) - \cos^{-1}\left(\frac{\mathrm{Re}(\tilde{\boldsymbol{g}}_n^H\boldsymbol{g}_n)}{||\boldsymbol{g}_n|| \cdot ||\tilde{\boldsymbol{g}}_n||}\right), \tag{52}$$

for $n = 1, \ldots, N$. The complexity of the D-LC solution is $\mathcal{O}(6N(M+K))$.

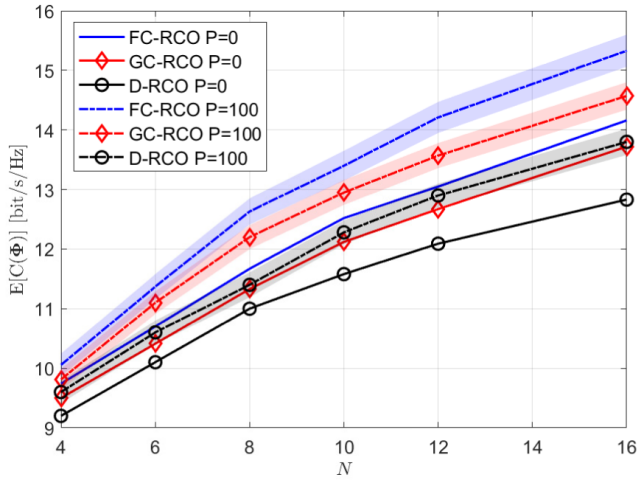

Fig. 4. Average achievable rate for RCO solution as a function of the number of BD-RIS elements for SNR=10 dB without (solid curves) and with (dot-dashed curves) randomization, i.e., $P = 0$ and $P = 100$, respectively. The shaded regions indicate the confidence interval.

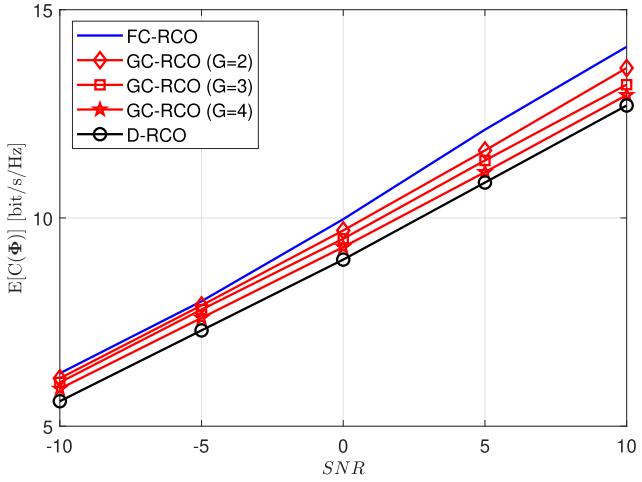

Fig. 5. Average achievable rate for RCO solution as a function of the SNR for $N = 12$, and randomization with $P = 100$. For the GC architecture, we consider $G = \{2, 3, 4\}$.

C. RCO Design and Performance

We first assess the performance of the RCO solution. In the following GC, FC, and D refer to group-connected, fully-connected, and diagonal BD-RIS architecture, respectively. The SNR is defined as the inverse of the per-antenna noise power. Fig. 4 shows the achievable rate averaged over the random channels (named $\mathbb{E}[C(\boldsymbol{\Phi})]$) as a function of N for SNR= 10 dB for the three BD-RIS architectures without (solid curves) and with (dot-dashed curves) randomization, i.e., $P = 0$ and $P = 100$, respectively. The curves for the GC architecture are obtained for $G = 2$. It can be seen that the relevance of the randomization depends on N: as N increases, randomization leads to greater performance improvements. Moreover, as expected, all techniques achieve a higher achievable rate when N is large. Lastly, the FC and GC architectures achieve similar performance, while the D architecture suffers from a small number of controllable parameters.

Fig. 5 shows $\mathbb{E}[C(\boldsymbol{\Phi})]$ as a function of the SNR for the three BD-RIS architectures with $N = 12$ and randomization

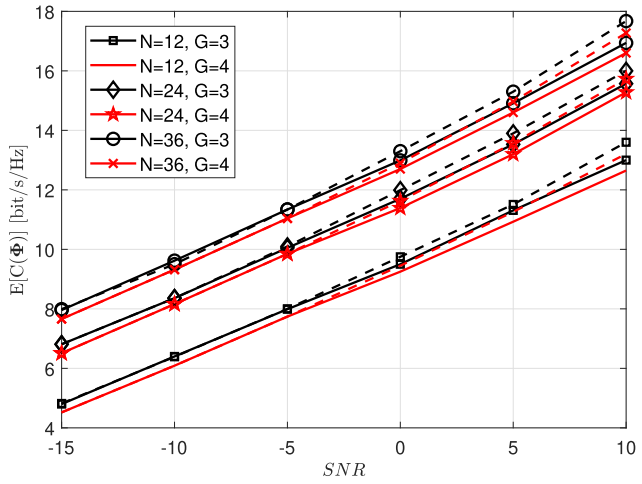

Fig. 6. Average achievable rate obtained with RCO as a function of the SNR for a group-connected architecture varying $N \in \{12, 24, 36\}$, for $G = 3$ and $G = 4$. Solid curves refer to results obtained without randomization, whereas dashed curves refer to results obtained when the randomization procedure is performed with $P = 100$.

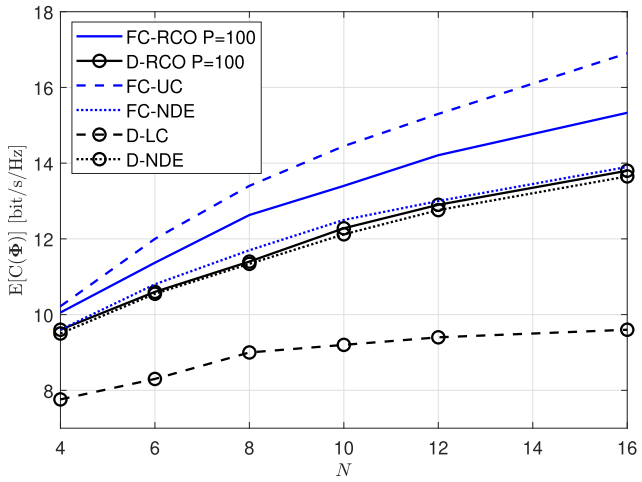

Fig. 7. Average achievable rate for RCO solution for fully-connected and diagonal architectures as a function of the number of BD-RIS elements N for SNR = 10 dB. The randomization is performed with $P = 100$ and the compared solutions are those described in Section VI-B.

($P = 100$). For the GC architecture, we consider $G = \{2, 3, 4\}$. It can be observed that increasing G decreases the rate. Indeed, as G grows, the GC architecture increasingly resembles the diagonal one. This is further confirmed by Fig. 6 that shows the average achievable rate as a function of the SNR for the GC architecture with $G = 3$ (black curves), and $G = 4$ (red curves), and $N = 12$ (squared- and solid curves), $N = 24$ (curves with diamonds and pentagrams), $N = 36$ (curves with circles and stars). Solid curves refer to results obtained without performing the randomization procedure ($P = 0$), while dashed curves refer to results obtained with randomization ($P = 100$).

Finally, Fig. 7 shows $\mathbb{E}[C(\mathbf{\Phi})]$ as a function of N for SNR= 10 dB and aims at comparing RCO solutions for the fully-connected and diagonal architectures with the existing solutions detailed in Section VI-B. Specifically, RCO is performed with randomization ($P = 100$). It can be seen that RCO solution for the FC architecture significantly outperforms FC-NDE solution. However, its performance remains below

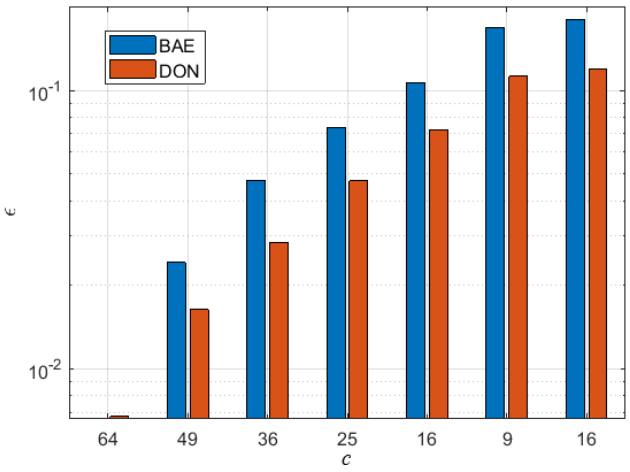

Fig. 8. The average achievable rate loss for a fully-connected architecture with $N = 8$ and SNR = 10 dB. BAE and DON approaches are considered for the AE design. The MSE is used as a loss function.

that of FC-UC, which represents the optimal solution of the original (non-relaxed) problem (8) for the fully-connected architecture. Moreover, it can be observed that the RCO solution for a diagonal architecture outperforms D-LC, while providing results quite similar to those of D-NDE.

D. Autoencoder Design

We now evaluate the impact of the AE design. Recall that for the optimization in the latent space, the AE is constrained to be linear (as discussed in Section V). Furthermore, given that the fully-connected architecture among the considered architectures has the highest computational and control rate requirements, our analysis here will focus exclusively on that architecture. Moreover, we consider $N = 8$ and SNR = 10 dB. However, these investigations can be easily extended to the other architectures, all with different values of N and SNR. The considered loss function is the MSE, unless otherwise specified. Performance is evaluated in terms of the MSE (34) and the average achievable rate loss ϵ with respect to the RCO solution, i.e.,

$$\epsilon = \mathbb{E}\left[\frac{C(\mathbf{\Phi}^\star) - C(\hat{\mathbf{\Phi}}^\star)}{C(\mathbf{\Phi}^\star)}\right], \tag{53}$$

where the average is taken over the channel realizations and thus the BD-RIS configurations. Note that ϵ measures the performance degradation when using the AE's output instead of the original input (i.e., RCO solution) as the BD-RIS configuration. We split the dataset into $8 \cdot 10^4$ samples for training, 10^4 for both validation and testing. The AE is trained considering an Adam optimizer with a learning rate of 10^{-4}, a batch size of 128, and 50 epochs.

1) Impact of Forcing the Constraints Guarantee: Figs. 8 and 9 show ϵ and MSE, respectively, as a function of the latent space dimension $c \in \{8, 9, 16, 25, 36, 49, 64\}$. We consider a basic AE (labeled as BAE) where both the encoder and decoder have one layer each, without activation functions. We compare this solution with an AE designed according to the DON approach, where the encoder is designed with only

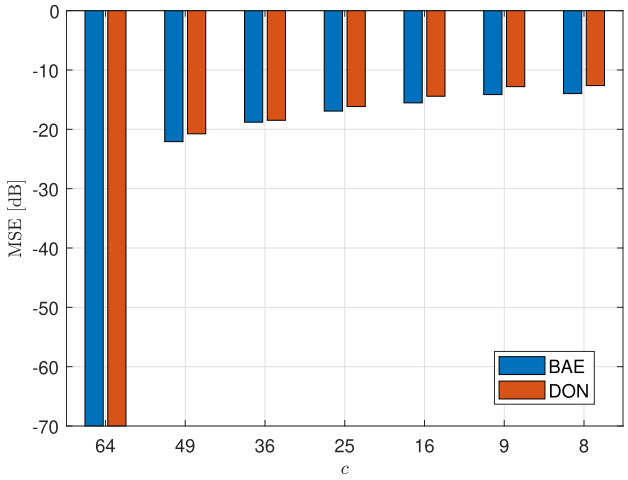

Fig. 9. MSE for a fully-connected architecture with $N = 8$ and SNR = 10 dB. BAE and DON approaches are considered for the AE design. The MSE is used as a loss function.

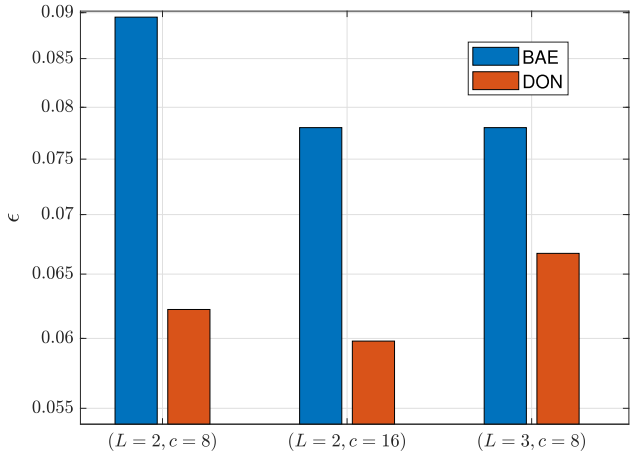

Fig. 10. The average achievable rate loss for a fully-connected architecture with $N = 8$ and SNR = 10 dB. The encoder and decoder are designed with one or more nonlinear hidden layers. The MSE is used as a loss function.

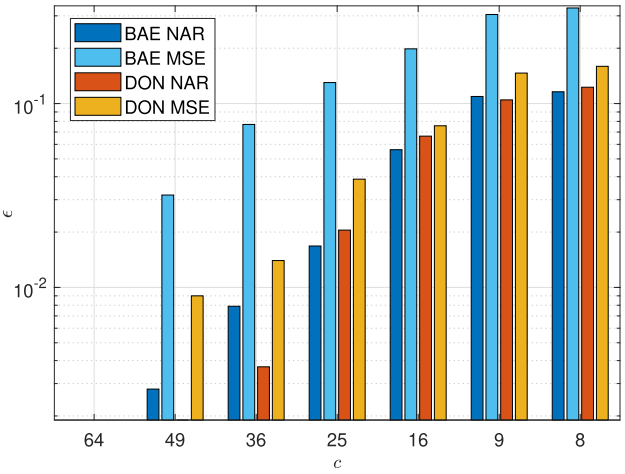

Fig. 11. The average achievable rate loss for a fully-connected architecture with $N = 8$, $K = M = 1$, and SNR = 10 dB. The encoder and decoder are designed with only one linear layer. Both the MSE and the negative achievable rate (NAR) are used as loss functions.

one layer without an activation function, while the decoder activation function is that discussed in Section IV-B. It is worth noting that, unlike for DON, the outputs for BAE do not inherently satisfy constraint (3). Therefore, before evaluating the resulting achievable rate, the BD-RIS matrix provided by the AE is normalized to ensure that its spectral norm is smaller than or equal to 1. Fig. 8 shows that the DON approach outperforms BAE, providing the lowest ϵ values. This suggests that explicitly enforcing the unit-norm constraints during the AE training, as done in DON, leads to a more effective configuration in terms of communication performance. However, when looking at the MSE between the input and reconstructed BD-RIS configuration vector in Fig. 9, a different pattern emerges. Indeed, DON provides a higher MSE compared to BAE. This is due to the explicit enforcement of the normalization constraint on the decoder output: by imposing this constraint, we restrict the solution space, preventing the AE from freely learning any transformation that minimizes the MSE. This led to a higher reconstruction error. By jointly considering ϵ and MSE results, it can be stated that while enforcing the constraints might slightly degrade reconstruction accuracy, it leads the model toward solutions that better fit the system's optimization goals.

2) Impact of Deep Autoncoders: To assess the impact of using a deep AE, we analyze the performance of BAE and DON approaches. Note that for the numerical results, each complex-valued input provided to the AE is represented by concatenating its real and imaginary parts. Accordingly, both BAE and DON employ ReLU as the activation function for the hidden layers, and Tanh as the activation function on the real and imaginary parts of the output layer for BAE. Training is performed using an Adam optimizer with a learning rate of 10^{-4}, a batch size of 128, and 50 epochs. Fig. 10 shows ϵ as a function of a different number of hidden layers and latent space dimension. We consider the cases in which the encoder and decoder have two hidden layers ($L = 2$), and $c = 8$ and 16. The first layer of the encoder maps the input of size 64 to a size of 32, and the second layer then reduces the size from 32 to c. The decoder mirrors this process. We also consider the case in which the encoder and decoder have three hidden layers ($L = 3$) and $c = 8$: the output size of the first encoder layer is 32, and the output size of the second encoder layer is 16. Once again, the decoder mirrors this process as previously discussed. Comparing Fig. 8 and Fig. 10, it can be seen that for the same value of c, the introduction of non-linearity and additional hidden layers reduces ϵ. This is particularly true for $c = 8$, while for $c = 16$, designing the encoder and decoder without a deep architecture does not significantly affect the AE performance.

3) Impact of the Loss Function: Lastly, we have also considered the negative achievable rate as the loss function for the training of the AE (see Section IV-B). However, due to the computation of the gradient for a specialized loss function (rather than the conventional MSE), the training time significantly increases. Thus, we focus on a simpler scenario with single-antenna devices. Fig. 11 shows the obtained ϵ with the MSE and the negative achievable rate loss functions (labeled as MSE and NAR, respectively) when considering BAE and DON approaches. We observe that using the rate loss function improves the performance of the AE, at the expense

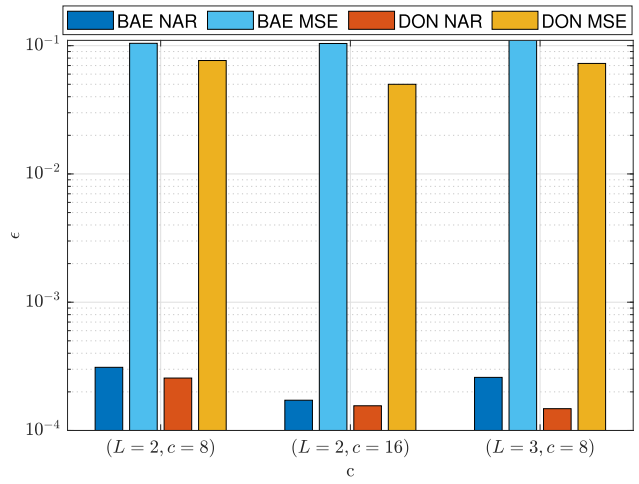

Fig. 12. The average achievable rate loss for a fully-connected architecture with $N = 8$, $K = M = 1$, and SNR = 10 dB. The encoder and decoder are designed with one or more nonlinear hidden layers. Both the MSE and NAR loss functions are used.

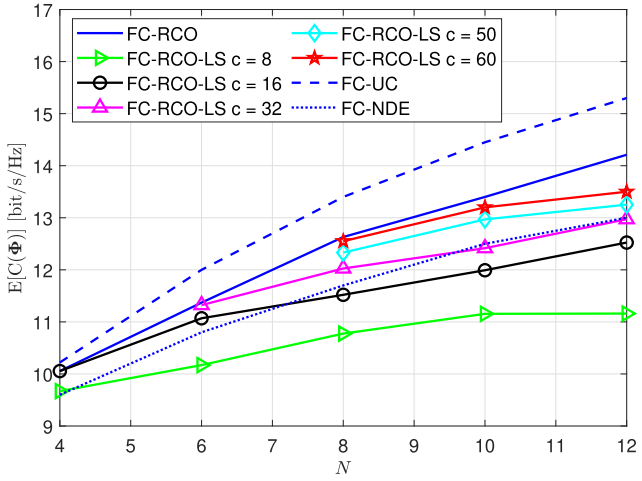

Fig. 13. Average achievable rate as a function of N for a BD-RIS with fully-connected architecture, SNR=10 dB. We consider RCO, RCO-LS with $c \in \{8, 16, 32, 50, 60\}$, and the compared solutions described in Section VI-B. For both RCO and RCO-LS, we perform the randomization with $P = 100$.

of higher computational complexity. This is further confirmed by Fig. 12, where a deep AE architecture is considered for both BAE and DON. Notably, for the same c values, the system performance is significantly improved when a deep AE and the negative achievable rate loss function are used.

E. RCO-LS Design and Performance

Here, we design and assess the performance of RCO-LS solution. The AE decoder has been obtained considering a linear AE (i.e., BAE) trained with the MSE loss function, following the BAE approach to maintain linearity. Fig. 13 shows $\mathbb{E}[C(\mathbf{\Phi})]$ as a function of N for a BD-RIS with fully-connected architecture and SNR= 10 dB. The solid, dotted, and dashed lines represent the performance of the RCO, FC-NDE, and FC-UC solutions, respectively, while the remaining curves (with markers) correspond to the RCO-LS approach varying the latent dimension $c \in \{8, 16, 32, 50, 60\}$. For both RCO and RCO-LS, we perform the randomization procedure with $P = 100$. Comparing RCO and RCO-LS solutions, it can be seen that for smaller c (e.g., $c = 8$),

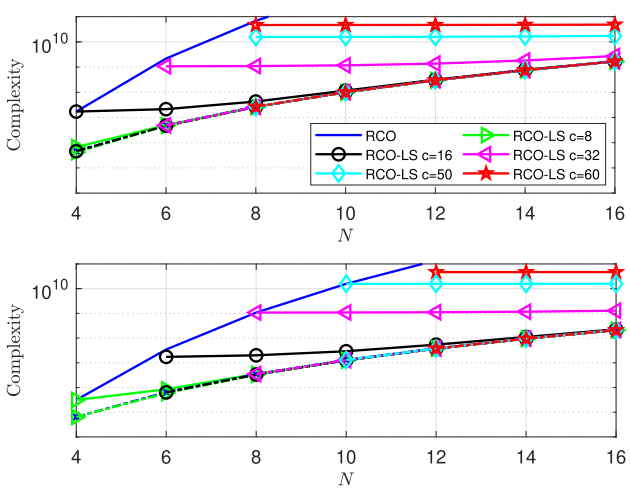

Fig. 14. Overall complexity as a function of N for a BD-RIS with a fully-connected architecture (on the top) and group-connected ($G = 2$) architecture (on the bottom), SNR=10 dB. We consider SDP (solid curves) and PGA (dot-dashed curves) for both the optimization in the original and latent space. We consider $c \in \{8, 16, 32, 50, 60\}$. We assume $T = 200$ as the number of PGA iterations, and we perform randomization with $P = 100$.

the average achievable rate obtained with RCO-LS is notably lower than that of RCO, indicating that a limited latent space worsens performance. As c increases, the RCO-LS average achievable rate improves gradually, approaching the RCO. This improvement occurs because a larger c enables better optimization within the latent space, thereby reducing the gap with RCO. Depending on the number of BD-RIS elements N, a sufficiently large c value is needed to capture the relevant features for effective optimization. Comparing RCO-LS with FC-NDE, it can be seen that for $c \geq 32$, RCO-LS outperforms the latter. Therefore, an inherent trade-off arises between dimensionality reduction in the BD-RIS optimization problem and performance.

F. Complexity Comparison

Fig. 14 shows the overall complexity as a function of N for a BD-RIS with a fully-connected architecture (on the top) and group-connected ($G = 2$) architecture (on the bottom), SNR= 10 dB. We compare the performance of SDP (solid curves) and PGA (dot-dashed curves) for both the optimization in the original and latent space with $c \in \{8, 16, 32, 50, 60\}$. We consider $T = 200$ as the number of iterations for PGA, and we perform randomization with $P = 100$. For SDP-based solution, as long as the randomization procedure is less computationally demanding than SDP (whether in the latent space or not), the optimization in the latent space exhibits a significantly lower computational complexity than that in the original space. Furthermore, for small c (i.e., $c = 8$ and $c = 16$) and N values, the complexity of SDP solution in the latent space is primarily dominated by the randomization procedure. In contrast, for $c \geq 32$, its complexity is almost due to solving the SDP relaxation in (43), which explains the flat behavior observed in the resulting curves. The overall complexity of the PGA-based solution is dominated by randomization, yielding the same complexity for different c, thus confirming that the PGA-based solution is computationally less demanding than the SDP-based one. Comparing the results obtained for the

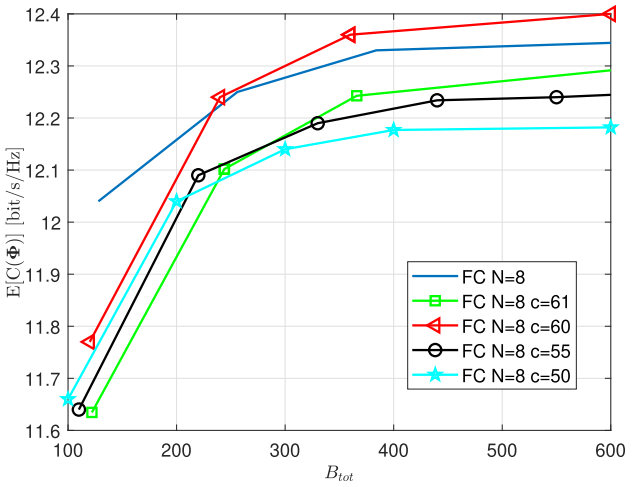

Fig. 15. Achievable rate as a function of total control rate B_{tot} for a fully-connected BD-RIS with $N = 8$. The curve labeled FC $N = 8$ corresponds to the uncompressed configuration, while the curves labeled FC $N = 8$, $c = 61, 60, 55, and 50$ represent latent-space compressed representations obtained via the autoencoder with various values of c.

group-connected and fully-connected architectures, it can be seen that the former requires significantly fewer computational operations compared to the latter, while still achieving comparable performance in terms of rate (see Fig. 5).

G. Rate Control Comparison

Fig. 15 shows the average achievable rate obtained after quantization as a function of the total control rate B_{tot} for different latent space dimensions c and the original space dimension when a fully-connected architecture with $N = 8$ is considered. For a fair comparison, both the latent and baseline configurations are quantized to ensure the same total number of bits B_{tot} used to describe each BD-RIS configuration. As it can be seen that, as B_{tot} increases, the curves tend to saturate around the same achievable rate value, indicating that finer quantization only provides marginal additional gains once the quantization noise becomes negligible. For $c = 50$ and $c = 55$, a slight rate degradation is obtained for low control rates, which progressively vanishes at higher B_{tot}. This confirms that the latent representation effectively preserves the most significant information. Furthermore, the case with $c = 60$ achieves a slightly higher rate than the uncompressed solution for most B_{tot} values. This behavior can be due to the regularization effect implicitly introduced by the autoencoder training: by constraining the representation to a lower dimension, the latent space model filters out redundant or noisy components of the original configuration, thus improving the robustness to quantization. Interestingly, for $c = 61$, the achievable rate is slightly lower than the uncompressed case, despite the larger latent space dimension. This non-monotonic behavior can be due to several factors: (i) the interaction between latent-space dimension and optimization, where slightly larger latent representations do not always lead to better configurations; (2) the effect of quantization, since increasing c reduces the number of bits per component for a fixed total control rate; and (iii) the implicit regularization of the autoencoder, where certain compressed dimensions (e.g., $c = 60$) happen to capture the most relevant information more effectively than larger latent space representations.

VII. Conclusion

In this paper, we proposed a procedure for designing the BD-RIS phase shifts to maximize the achievable rate. We first proposed a solution based on a relaxation of the trace-maximization problem, with a randomization approach for achievable rate maximization. However, this solution still has a high communication overhead to control the large number of BD-RIS elements. Therefore, we introduce a dynamic mapping between the BD-RIS configuration and a small number of control variables. The mapping is provided by an autoencoder, trained on the effectively used (optimal) configurations in the specific deployment. Specifically, the encoder is used to obtain the value of the small set of control variables, while the decoder is used at the BD-RIS for the tuning of the impedances from the control variables. Moreover, we also design the BD-RIS configuration directly in the latent space of the autoencoder, reducing the complexity. Numerical results show that, at the same total control rate, the latent-space representation of a BD-RIS configuration with a fully-connected architecture achieves up to a 5% rate improvement over the uncompressed configuration, confirming that the proposed AE-based mapping preserves the most significant information while reducing the number of control bits. Moreover, when comparing the group-connected and fully-connected architectures, the former requires two orders of magnitude fewer computational operations, while still achieving $85 - -95\%$ of the achievable rate of the fully-connected counterpart.

References

[1] H. Li, S. Shen, M. Nerini, and B. Clerckx, "Reconfigurable intelligent surfaces 2.0: Beyond diagonal phase shift matrices," *IEEE Commun. Mag.*, vol. 62, no. 3, pp. 102–108, Mar. 2024.

[2] H. Li, S. Shen, and B. Clerckx, "Beyond diagonal reconfigurable intelligent surfaces: From transmitting and reflecting modes to single-, group-, and fully-connected architectures," 2022, *arXiv:2205.02866*.

[3] H. Li, S. Shen, and B. Clerckx, "Beyond diagonal reconfigurable intelligent surfaces: A multi-sector mode enabling highly directional full-space wireless coverage," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 8, pp. 2446–2460, Aug. 2023.

[4] M. Nerini, S. Shen, and B. Clerckx, "Discrete-value group and fully connected architectures for beyond diagonal reconfigurable intelligent surfaces," *IEEE Trans. Veh. Technol.*, vol. 72, no. 12, pp. 16354–16368, Dec. 2023.

[5] S. Shen, B. Clerckx, and R. Murch, "Modeling and architecture design of reconfigurable intelligent surfaces using scattering parameter network analysis," *IEEE Trans. Wireless Commun.*, vol. 21, no. 2, pp. 1229–1243, Feb. 2022.

[6] 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," *IEEE Commun. Surveys Tuts.*, vol. 26, no. 2, pp. 781–823, Jan. 2024.

[7] A. V. Guglielmi and S. Tomasin, "Joint RIS optimization and channel estimation with decision tree-based adaptive reconfiguration," *IEEE Trans. Commun.*, vol. 73, no. 6, pp. 4049–4060, Jun. 2025.

[8] D. Wijekoon, A. Mezghani, and E. Hossain, "Phase shifter optimization in RIS-aided MIMO systems under multiple reflections," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 8969–8983, Aug. 2024.

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

[10] M. Nerini, S. Shen, and B. Clerckx, "Closed-form global optimization of beyond diagonal reconfigurable intelligent surfaces," *IEEE Trans. Wireless Commun.*, vol. 23, no. 2, pp. 1037–1051, Feb. 2024.

[11] M. Nerini, S. Shen, H. Li, and B. Clerckx, "Beyond diagonal reconfigurable intelligent surfaces utilizing graph theory: Modeling, architecture design, and optimization," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 9972–9985, Aug. 2024.
[12] H. Li, S. Shen, M. Nerini, M. Di Renzo, and B. Clerckx, "Beyond diagonal reconfigurable intelligent surfaces with mutual coupling: Modeling and optimization," *IEEE Commun. Lett.*, vol. 28, no. 4, pp. 937–941, Apr. 2024.
[13] H. Li and B. Clerckx, "Non-reciprocal beyond diagonal RIS: Multiport network models and performance benefits in full-duplex systems," *IEEE Trans. Commun.*, vol. 73, no. 11, pp. 12221–12234, Nov. 2025.
[14] G. Bartoli, A. Abrardo, N. Decarli, D. Dardari, and M. Di Renzo, "Spatial multiplexing in near field MIMO channels with reconfigurable intelligent surfaces," *IET Signal Process.*, vol. 17, no. 3, p. 12195, Mar. 2023.
[15] I. Santamaria, M. Soleymani, E. Jorswieck, and J. Gutiérrez, "SNR maximization in beyond diagonal RIS-assisted single and multiple antenna links," *IEEE Signal Process. Lett.*, vol. 30, pp. 923–926, 2023.
[16] I. Santamaria, M. Soleymani, E. Jorswieck, and J. Gutiérrez, "MIMO capacity maximization with beyond-diagonal RIS," in *Proc. IEEE 25th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Sep. 2024, pp. 936–940.
[17] H. Li, S. Shen, Y. Zhang, and B. Clerckx, "Channel estimation and beamforming for beyond diagonal reconfigurable intelligent surfaces," *IEEE Trans. Signal Process.*, vol. 72, pp. 3318–3332, 2024.
[18] J. V. Alegría, J. Thunberg, and O. Edfors, "Channel orthogonalization with reconfigurable surfaces: General models, theoretical limits, and effective configuration," *IEEE Trans. Wireless Commun.*, vol. 24, no. 6, pp. 5179–5195, Jun. 2025.
[19] X. Yu, D. Li, Y. Xu, and Y.-C. Liang, "Convolutional autoencoder-based phase shift feedback compression for intelligent reflecting surface-assisted wireless systems," *IEEE Commun. Lett.*, vol. 26, no. 1, pp. 89–93, Jan. 2022.
[20] X. Yu and D. Li, "Phase shift compression for control signaling reduction in IRS-aided wireless systems: Global attention and lightweight design," *IEEE Trans. Wireless Commun.*, vol. 23, no. 8, pp. 8528–8541, Aug. 2024.
[21] Z. Zhang and W. Yu, "Codebook learning for active sensing with reconfigurable intelligent surface," in *Proc. IEEE 25th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Sep. 2024, pp. 821–825.
[22] A. M.-C. So, J. Zhang, and Y. Ye, "On approximating complex quadratic optimization problems via semidefinite programming relaxations," *Math. Program.*, vol. 110, no. 1, pp. 93–110, Mar. 2007.
[23] N. Benvenuto, G. Cherubini, and S. Tomasin, *Algorithms for Communications Systems and Their Applications*. Hoboken, NJ, USA: Wiley, 2020.
[24] H. Li, Y. Zhang, and B. Clerckx, "Channel estimation for beyond diagonal reconfigurable intelligent surfaces with group-connected architectures," in *Proc. IEEE 9th Int. Workshop Comput. Adv. Multi-Sensor Adapt. Process. (CAMSAP)*, Dec. 2023, pp. 21–25.
[25] A. Kord, D. L. Sounas, and A. Alu, "Microwave nonreciprocity," *Proc. IEEE*, vol. 108, no. 10, pp. 1728–1758, Oct. 2020.
[26] J. Tapie, M. Nerini, B. Clerckx, and P. del Hougne, "Beyond-diagonal RIS prototype and performance evaluation," 2025, *arXiv:2505.13392*.
[27] 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, Sep. 2022.
[28] Y. Dorrazehi, A. V. Guglielmi, and S. Tomasin, "MUSIC-based channel estimation with adaptive reconfiguration of diagonal RIS," in *Proc. 15th Int. Conf. Ubiquitous Future Netw. (ICUFN)*, Jul. 2024, pp. 429–434.
[29] G. T. de Araújo and A. L. F. de Almeida, "Semi-blind channel estimation for beyond diagonal RIS," in *Proc. 58th Asilomar Conf. Signals, Syst., Comput.*, Oct. 2024, pp. 1586–1590.
[30] N. Ginige, A. S. de Sena, N. H. Mahmood, N. Rajatheva, and M. Latva-Aho, "Efficient channel prediction for beyond diagonal RIS-assisted MIMO systems with channel aging," *IEEE Trans. Veh. Technol.*, vol. 74, no. 8, pp. 12658–12672, Aug. 2025.
[31] M. Grant, S. Boyd, and Y. Ye, "CVX: MATLAB software for disciplined convex programming; CVX research," Stanford, CA, USA, Tech. Rep., 2008.
[32] L. M. G. Drummond and A. N. Iusem, "A projected gradient method for vector optimization problems," *Comput. Optim. Appl.*, vol. 28, no. 1, pp. 5–29, Apr. 2004.
[33] Z.-Q. Luo, W.-K. Ma, A. M. So, Y. Ye, and S. Zhang, "Semidefinite relaxation of quadratic optimization problems," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 20–34, May 2010.
[34] E. Plaut, "From principal subspaces to principal components with linear autoencoders," 2018, *arXiv:1804.10253*.
[35] D. Tse and P. Viswanath, *Fundamentals of Wireless Communication*. Cambridge, U.K.: Cambridge Univ. Press, May 2005.
[36] A. M. Sayeed, "Optimization of reconfigurable intelligent surfaces through trace maximization," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC Workshops)*, Jun. 2021, pp. 1–6.
[37] 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, Dec. 2020.

Anna Valeria Guglielmi (Member, IEEE) received the B.Sc., M.Sc., and Ph.D. degrees in information engineering from the University of Padova, Italy, in 2012, 2014, and 2018, respectively. Since 2017, she has been a Visiting Scientist with 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.

Mattia Scarin Callegaro received the B.S. degree in computer engineering from the University of Padova, Italy, where he is currently pursuing the master's degree in ICT.

Yaser Dorrazehi (Graduate Student Member, IEEE) received the B.Sc. degree in electrical engineering from Chabahar Maritime University (CMU), Chabahar, Iran, in 2016, and the M.Sc. degree in telecommunication engineering from the University of Sistan and Baluchestan (USB), Zahedan, Iran, in 2019. He is currently pursuing the Ph.D. degree with the University of Padova. In 2025, he was a Visiting Researcher with Ruhr University Bochum (RUB), Germany. His current research interests include signal processing for RIS/metasurface-based wireless communication systems, and integrated sensing and communication systems.

Stefano Tomasin (Senior Member, IEEE) received the Ph.D. degree from the University of Padova, Italy, in 2003. During his career, he has visited the IBM Research, Switzerland; the Philips Research, The Netherlands; Qualcomm, CA, USA; Polytechnic University, Brooklyn, NY, USA; and Huawei, France. He is currently a Full Professor with the University of Padova. 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 member of EURASIP. He is or has been an Editor of IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON SIGNAL PROCESSING (2017–2020), *EURASIP Journal on Wireless Communications and Networking*, and IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY.

Open Access funding provided by 'Università degli Studi di Padova'
within the CRUI CARE Agreement