Deep Reinforcement Learning for Practical Phase-Shift Optimization in RIS-Aided MISO URLLC Systems

Ramin Hashemi, *Graduate Student Member, IEEE*, Samad Ali, *Member, IEEE*, Nurul Huda Mahmood, and Matti Latva-Aho, *Senior Member, IEEE*

***Abstract*—We study the joint active/passive beamforming and channel blocklength (CBL) allocation in a nonideal reconfigurable intelligent surface (RIS)-aided ultrareliable and low-latency communication (URLLC) system. The considered scenario is a finite blocklength (FBL) regime and the problem is solved by leveraging a deep reinforcement learning (DRL) algorithm named twin-delayed deep deterministic policy gradient (TD3). First, assuming an industrial automation system, the signal-to-interference-plus-noise ratio and achievable rate in the FBL regime are identified for each actuator. Next, the joint active/passive beamforming and CBL optimization problem (OP) is formulated where the objective is to maximize the total achievable FBL rate in all actuators, subject to nonlinear amplitude response at the RIS elements, BS transmit power budget, and total available CBL. Since the formulated problem is highly nonconvex and nonlinear, we resort to employing an actor–critic policy gradient DRL algorithm based on TD3. The considered method relies on interacting RIS with the industrial automation environment by taking actions which are the phase shifts at the RIS elements, CBL variables, and BS beamforming to maximize the expected observed reward, i.e., the total FBL rate. We assess the performance loss of the system when the RIS is nonideal, i.e., with nonlinear amplitude response, and compare it with ideal RIS without impairments. The numerical results show that optimizing the RIS phase shifts, BS beamforming, and CBL variables via the TD3 method with deterministic policy outperforms conventional methods and it is highly beneficial for improving the network total FBL rate considering finite CBL size.**

***Index Terms*—Block error probability (BLER), deep reinforcement learning (DRL), finite blocklength (FBL), industrial automation, reconfigurable intelligent surface (RIS), ultrareliable low-latency communications (URLLC).**

I. Introduction

INDUSTRIAL wireless systems involving devices, actuators, and robots that require ultrareliable and low-latency communications (URLLCs) are anticipated to grow in the future sixth generation of wireless communications (6G) [2], [3]. Industrial Internet of Things (IIoT) is the industrial application of IoT connectivity along with networking and cloud computing based on data analytics collected from IoT devices. Industrial environments are diverse and heterogeneous as they are characterized by a large number of use-cases and applications [4], [5]. An underlying commonality among these diverse applications is that the wireless industrial automation connectivity solutions envisioned in Industry 4.0 (initialized in 5G) [6] will leverage cloud computing and machine learning throughout the manufacturing process. The expected URLLC key performance indicators (KPIs) in 6G networks are *reliability* up to $1 - 10^{-9}$, *latency* around 0.1–1-ms round-trip time, and *jitter* in the order of 1 μs for industrial control networks [3]. There is also high data rate demand due to the increased number of sensors and their resolution, e.g., for robots. In URLLC both the data and meta data sizes are small while both parts need to be very robust and have minimal error [7]. Thus, joint encoding of data and meta data is beneficial in terms of coding gain [8]. In addition, as the packet lengths in URLLC are usually small, the finite blocklength (FBL) theory is leveraged to investigate the achievable rate [9].

Reconfigurable intelligent surface (RIS) has been recognized as a promising technology to enhance the energy efficiency, and spectral efficiency of wireless communications [10]. An RIS is composed of metamaterials where the phase and amplitude of each element can be adjusted. This allows the reflected signal to have a desired effect, e.g., enhance the received signal-to-interference-plus-noise ratio (SINR) at a given location. Because of this feature, the distribution of the received signal, in the case of a blocked transmitter–receiver channel, has very little variation. The performance of such systems depends on the quantization levels at each phase-shift element [11], [12] or circuitry impairments [13]. Thus, the application of the RIS technology in industrial automation in ensuring high reliability is very promising [11]. Furthermore, since there is no processing overhead at the RIS and the increase in the delay spread caused by an RIS is rather small, unlike conventional relays, URLLC latency requirements can be satisfied as well by a suitable design in the higher layer. Therefore, the RIS technology has high potential in URLLC applications [14].

There are a number of challenges when deploying RIS technology in practical industrial automation use cases. For

Manuscript received 19 September 2022; revised 24 November 2022; accepted 23 December 2022. Date of publication 29 December 2022; date of current version 9 May 2023. This work was supported by the Academy of Finland, 6G Flagship Program under Grant 346208. Ramin Hashemi would like to acknowledge the support of the Nokia Scholarship Foundation. This article was presented in part at the Joint European Conference on Networks and Communications & 6G Summit, 2022, pp. 518–523 [DOI: 10.1109/EuCNC/6GSummit54941.2022.9815804]. *(Corresponding author: Ramin Hashemi.)*

The authors are with the Centre for Wireless Communications, University of Oulu, 90014 Oulu, Finland (e-mail: ramin.hashemi@oulu.fi; samad.ali@oulu.fi; nurulhuda.mahmood@oulu.fi; matti.latva-aho@oulu.fi).

Digital Object Identifier 10.1109/JIOT.2022.3232962

instance, efficient physical-layer design techniques, e.g., channel estimation, phase shift, and amplitude response control, and system-level optimizations, are still challenging and considered active research topics. Toward this goal, optimization-oriented approaches relying on exhaustive alternating optimization methods have been introduced in the existing literature. Note that due to the unit modulus phase-shifting constraint, the associated optimizations in the existing literature are highly nonconvex and nonlinear [15]. Thus, achieving a suboptimal phase-shift design is highly complicated and time-consuming. Additionally, since the radio channel characteristics vary over time or frequency, optimization-based methods need to be continuously tuned/re-executed to find the optimized phase-shift values at the RIS which is impractical in mission-critical and sensitive industrial automation scenarios. Furthermore, the complexity of phase-shift design optimizations increases considering the practical RIS in which the amplitude response changes by the value of phase shift in a nonlinear manner [13]. This poses new challenges to the existing optimization-based approaches which are still sophisticated and hard to solve even for ideal RISs [16].

In recent years, machine learning methods, particularly deep reinforcement learning (DRL) algorithms, have been considered a reliable and powerful framework in wireless communications [16], [17]. The DRL methods rely on taking action and receiving a certain reward based on the action and interacting with the environment, which constructs the agent's experience. Thus, these methods usually do not require large training data set, which is highly beneficial in practical resource allocation problems in wireless communications. Therefore, the applicability of DRL toward more reliable and faster solutions in the next generations of URLLC is highlighted with the advent of efficient new algorithms [17], [18], [19]. In this article, our aim is to investigate practical phase-shift design and optimization of a RIS-assisted URLLC system in industrial automation by employing a novel and sophisticated DRL algorithm named as twin delayed deep deterministic policy gradient (TD3) [20].

A. Related Work

Considering total available channel blocklengths (CBLs) as a constraint in various URLLC systems that incorporate short packet transmission is an active area of research. For example, Ranjha and Kaddoum [21] minimized the total transmit power of the IoT devices by assuming a finite available CBL budget. The resource allocation problems in RIS-assisted URLLC systems over short packet communications is a relatively new topic and have only been investigated in a few papers [22], [23], [24], [25], [26]. Ghanem et al. [22] studied an OP for beamforming and phase-shift control in a RIS-enabled orthogonal frequency-division multiple access (OFDMA) URLLC system where the cooperation of a set of base stations (BSs) to serve the URLLC traffic was discussed. In [23], the unmanned aerial vehicles (UAVs) trajectory and CBL allocation in FBL regime as well as phase-shift optimization in a RIS-aided network to minimize the total error probability was investigated. In [24] and [25], a CBL allocation and the RIS reflective phase-shift OP (with user grouping in [24]) was studied in a URLLC system where a dedicated RIS assists the BS in transmitting short packets in FBL scenario. The proposed OPs were tackled by a semi-definite relaxation method and the user grouping problem in [24] was solved by a greedy algorithm. Almekhlafi et al. [26] studied the applicability of the RIS in joint multiplexing of enhanced mobile broadband (eMBB) and URLLC traffic to optimize the admitted URLLC packets while minimizing the eMBB rate loss to ensure the quality of service of the two traffic types by designing RIS phase-shift matrices. It is worth noting that in all of the aforementioned works, the proposed problems were tackled by complex optimization-based algorithms as they usually are based on iterative algorithms. Particularly, even with an appropriate method that considers the nonlinear amplitude response at the RIS elements, the computational complexity of such algorithms will still be significant. Several existing works such as [27], [28], [29], [30], [31], [32], [33], [34], and [35] elaborated recent advances in DRL techniques on phase-shift design at the RIS. In [27], the secrecy rate of a wireless channel with RIS technology was maximized with quality of service (QoS) constraints on the secrecy rate and data rate requirements of the users. The resulting problem is solved by a novel DRL algorithm based on post-decision state and prioritized experience replay methods. In [28], deep deterministic policy gradient (DDPG) method was employed to maximize the received signal-to-noise ratio (SNR) in a downlink (DL) multiple-input–single-output (MISO) system via adjusting the phase shifts at the RIS. Huang et al. [29], Zhang et al. [30], and Zhu et al. [31] studied a RIS-assisted MISO system to adjust the BS transmit beamforming and the passive beamforming at the RIS in order to optimize the total achievable rate in infinite blocklength regime, i.e., assuming Shannon capacity, via DDPG [29], [30] or soft actor–critic (SAC) [31] methods. The half-duplex and full-duplex operating modes were compared in [32] for a RIS-aided MISO system. Joint relay selection and RIS reflection coefficient optimization in cooperative networks were studied in [33]. The work in [34] considered maximizing the total achievable rate in infinite blocklength regime over a multihop multiuser RIS-aided wireless terahertz communication system. A recent study in [35] investigated the applicability of distributed proximal policy optimization (PPO) technique in active/passive beamforming at the BS/RIS in a multiuser scenario. It is worth noting that the considered problem was defined in infinite CBL regime under the Shannon rate formula and the optimization of CBL was not the topic of interest.

Despite the interesting results in the aforementioned works on joint active/passive beamforming design in RIS-aided communications, the optimization of the CBL and beamforming at the BS/RIS while considering the impact of impairments in practical RIS with nonlinear amplitude response on the performance of a URLLC system over FBL regime has not been investigated before. In addition, most of the prior studies assumed that the RIS is ideal and the scenario is infinite blocklength regime while the conventional DDPG algorithm was utilized to solve the proposed resource allocation problem.

TABLE I
NOTATIONS AND SYMBOLS USED IN THIS ARTICLE

Notation	Description	Notation	Description
M	Number of BS antennas	N	Number of RIS elements
K	Number of actuators	d	Antenna/element spacing ($\mathsf{d} \leq \frac{\lambda}{2}$)
β^{inc}	Large-scale fading coefficient between BS and RIS	β_k^{RIS}	Large-scale fading coefficient between RIS and actuator k
$\mathbf{H}_{\text{LoS}}$	LoS channel matrix between BS and RIS	$\mathbf{H}_{\text{NLoS}}$	NLoS channel matrix between BS and RIS
$\bar{\mathbf{h}}_k^{\text{RIS}}$	LoS channel vector between RIS and actuator k	$\tilde{\mathbf{h}}_k^{\text{RIS}}$	NLoS channel vector between RIS and actuator k
ζ	Rician factor for BS-RIS path	ζ_k^{RIS}	Rician factor between RIS and actuator k
$\phi_1^{a/e}$	Azimuth/elevation angle between RIS elements and BS surface	$\phi_2^{a/e}$	Azimuth/elevation angle between BS antennas and RIS surface
$\phi_3^{a,k}$, $\phi_3^{e,k}$	Azimuth/elevation angle between RIS and actuator k	λ	The wavelength
$x_k[\cdot]$	Transmitted symbol to actuator k	$n_k[\cdot]$	Additive white noise in actuator k
p_k	Transmit power of BS for actuator k	p_{total}	Total transmit power of BS
N_0	White noise spectral efficiency	W	Bandwidth
$\beta_{\min}$	Minimum amplitude value in RIS phase shift model	α	The steepness in RIS phase shift model
ϕ	Horizontal distance between $-\frac{\pi}{2}$ and $\beta_{\min}$	ε_k	Target error probability for actuator k
c_k	The CBL allocated for actuator k	$c_k^{\min}$	Minimum CBL value to ensure validity of FBL rate
L_k	Number of information bits in FBL regime	L_{total}	Total number of information bits in FBL regime
C	Total number of available CBLs	$\boldsymbol{\omega}_k$	Beamforming vector for actuator k
$\mathcal{A}$	Set of actions	$\mathcal{S}$	Set of states
$\mathcal{P}_{s\to s'}$	Set of transition probabilities from state s to s'	γ	Discount factor
$\pi(a\|s)$	The probability of choosing action a given state s	$a = \mu(s;\boldsymbol{\xi})$	Deterministic policy network parameterized by $\boldsymbol{\xi}$ in state s
$\mathcal{R}$	Set of rewards	τ	Polyak averaging hyperparameter
N_{episode}	Number of episodes	N_{steps}	Number of steps in an episode
κ, κ'	Normal Gaussian noise added to action	t'	Policy network updating period
$\boldsymbol{\Upsilon}_t^k$, $\Upsilon_t^{kk'}$	Auxiliary vector, variables, respectively for each k, k' at step t	$\boldsymbol{\theta}$	RIS phase shift vector (in matrix form $\boldsymbol{\Theta} = \text{diag}(\boldsymbol{\theta})$)

However, several drawbacks are associated with this method, i.e., overestimation of the action-value function, unexpected actions, and sudden performance degradation due to frequent policy network update which are addressed meticulously in the novel twin-delayed DDPG, i.e., TD3 method. Motivated by the compelling works on resource allocation via DRL methods in RIS communications, we aim to extend our prior work in [1] and elaborate the joint active/passive beamforming and CBL allocation problem where the objective is to maximize the total FBL rate subject to nonlinear equality constraint for amplitude-phase response at the RIS. The numerical results demonstrate that while the TD3 algorithm is well-suited to the proposed problem compared to typical SAC schemes, optimizing CBLs between actuators and performing active/passive beamforming design in the practical RIS systems with imperfections improves the network total FBL rate and reduces the transmission duration significantly. Furthermore, the performance reduction gap between an ideal RIS with continuous phase shift and the nonideal RIS considering nonlinear amplitude response is elaborated. Also, we show that by optimizing CBLs among actuators the transmission duration reduces by 17% compared with equal CBL allocation.

B. Notations and Structure of This Article

In this article, $\mathbf{h} \sim \mathcal{CN}(\mathbf{0}_{N\times 1}, \mathbf{C}_{N\times N})$ denotes an N-dimensional circularly symmetric (central) complex normal distribution vector with N-dimensional zero mean vector $\mathbf{0}$ and covariance matrix $\mathbf{C}$. The operations $[\,\cdot\,]^{\text{H}}$, $[\,\cdot\,]^{\text{T}}$ denote the transpose and conjugate transpose of a matrix or vector, respectively. Also, the operators $\mathbb{E}[\,\cdot\,]$ and $\mathbb{V}[\,\cdot\,]$ denote the statistical expectation and variance, respectively. A summary of the notations and symbols used in this article is shown in Table I.

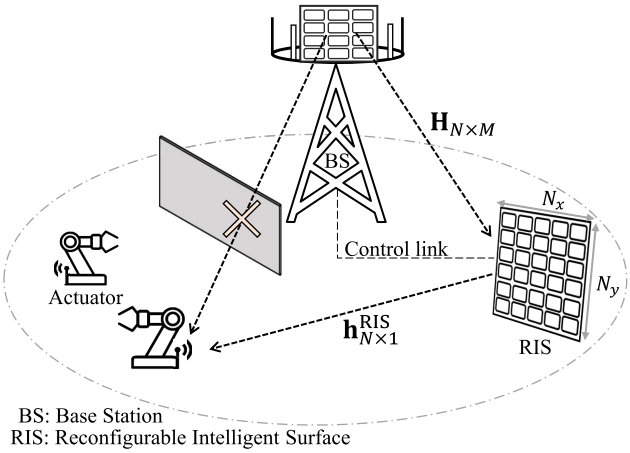

Fig. 1. Considered system model.

The structure of this article is organized as follows. In Section II, the system model and the FBL rate is proposed, then the optimization framework of joint active/passive beamforming design and CBL allocation is presented. In Section III the DRL preliminaries and exploited solution approach are studied. The numerical results are presented in Section IV. Finally, Section V concludes this article.

II. SYSTEM MODEL AND PROBLEM FORMULATION

A. System Model

Consider the DL of an RIS-assisted wireless network in a factory setting which consists of a BS with $M = M_x \times M_y$ uniform planar array (UPA) antennas and K single antenna actuators as illustrated in Fig. 1. The RIS which has $N = N_x \times N_y$ phase-shift elements constructs a communication channel between the actuators and multiantenna BS.

It is assumed that the direct channels between the BS and actuators are blocked by possible obstacles in the factory and only reflected channels exist. Thus, the total channel response between the BS and an actuator is established by the reflected path from the RIS. The channel matrix $\mathbf{H} \in \mathbb{C}^{N\times M}$ between BS and the RIS is denoted by

$$\mathbf{H} = \sqrt{\frac{\zeta}{\zeta+1}}\overline{\mathbf{H}}_{\text{LoS}} + \sqrt{\frac{1}{\zeta+1}}\mathbf{H}_{\text{NLoS}} = \left[\mathbf{h}_1^{\text{inc}}, \ldots, \mathbf{h}_M^{\text{inc}}\right] \quad (1)$$

with the column vectors $\mathbf{h}_m^{\text{inc}} = \sqrt{(\zeta/\zeta+1)}\overline{\mathbf{h}}_m^{\text{inc}} + \sqrt{(1/\zeta+1)}\tilde{\mathbf{h}}_m^{\text{inc}}$ for $\forall m \in \{1, \ldots, M\}$ where each nonline-of-sight (NLoS) channel vector is distributed as $\tilde{\mathbf{h}}_m^{\text{inc}} \sim \mathcal{CN}(\mathbf{0}_{M\times 1}, \beta^{\text{inc}}\mathbf{I}_M)$ in which β^{inc} is the path loss from BS to the RIS, and $\mathbf{I}_M$ is an identity matrix of size M. The proportion of line-of-sight (LoS) to the NLoS channel gain is defined as the Rician parameter ζ. Additionally, the LoS channel $\overline{\mathbf{H}}_{\text{LoS}} = [\overline{\mathbf{h}}_1^{\text{inc}}, \ldots, \overline{\mathbf{h}}_M^{\text{inc}}]$ is defined as [36]

$$\overline{\mathbf{H}}_{\text{LoS}} = \sqrt{\beta^{\text{inc}}}\mathbf{a}^{\text{H}}\left(\phi_1^a, \phi_1^e, N_x, N_y\right) \times \mathbf{a}\left(\phi_2^a, \phi_2^e, M_x, M_y\right) \quad (2)$$

where $\phi_1^{a/e}$ denotes the azimuth/elevation angle of a row/column of the UPA at the RIS with respect to the BS antenna surface. Similarly, $\phi_2^{a/e}$ is the azimuth/elevation angle between the direction of a row/column of the UPA at the BS with respect to the RIS plane. In addition, the vector $\mathbf{a}(x, y, N_1, N_2)$ is defined by [36]

$$\mathbf{a}(x, y, N_1, N_2) = \text{rvec}(\boldsymbol{\mathcal{H}}) \quad (3)$$

where rvec$(\cdot)$ denotes the row vectorization of a matrix, and

$$\boldsymbol{\mathcal{H}} = \left(e^{\text{j}\mathcal{G}(x,y,n_1,n_2)}\right)_{n_1=1,2,\ldots,N_1, n_2=1,2,\ldots,N_2} \in \mathbb{C}^{N_1\times N_2} \quad (4)$$

such that each element row n_1 and column n_2 are constructed by means of [36]

$$\mathcal{G}(x, y, n_1, n_2) = 2\pi\frac{\mathsf{d}}{\lambda}[(n_1-1)\cos x + (n_2-1)\sin x]\sin y \quad (5)$$

in which λ is the operating wavelength, and $\mathsf{d} \leq (\lambda/2)$ is the antenna/element spacing. Similarly, the channel between RIS and actuator k is

$$\mathbf{h}_k^{\text{RIS}} = \sqrt{\frac{\zeta_k^{\text{RIS}}}{\zeta_k^{\text{RIS}}+1}}\overline{\mathbf{h}}_k^{\text{RIS}} + \sqrt{\frac{1}{\zeta_k^{\text{RIS}}+1}}\tilde{\mathbf{h}}_k^{\text{RIS}} \quad (6)$$

where the Rician parameter ζ_k^{RIS} controls the proportion of LoS to the NLoS channel gain in actuator k. The NLoS channel is distributed as $\tilde{\mathbf{h}}_k^{\text{RIS}} \sim \mathcal{CN}(\mathbf{0}_{N\times 1}, \beta_k^{\text{RIS}}\mathbf{I}_N)$ such that β_k^{RIS} is the path-loss coefficient from RIS to actuator k. Furthermore, the LoS channel $\overline{\mathbf{h}}_k^{\text{RIS}} \in \mathbb{C}^{N\times 1}$ is modeled by

$$\overline{\mathbf{h}}_k^{\text{RIS}} = \sqrt{\beta_k^{\text{RIS}}}\mathbf{a}\left(\phi_3^{a,k}, \phi_3^{e,k}, N_x, N_y\right) \quad \forall k \in \mathcal{K} \quad (7)$$

in which $\mathcal{K} = \{1, 2, \ldots, K\}$, and $\phi_3^{a,k}, \phi_3^{e,k}$ are the azimuth (elevation) angles between RIS and the actuator k assuming the center of the coordinate system is at the RIS.

We assume that full channel state information (CSI) is available, i.e., the individual coefficients of the product channel response $\mathbf{h}_k^{\text{RIS}^{\text{H}}}\boldsymbol{\Theta}\mathbf{H}$ are obtainable at BS. First, as the location of BS/RIS is fixed, the BS-RIS channel matrix $\mathbf{H}$ remains approximately unchanged over a long period, hence, it is considered to be quasi-static by ignoring unlikely perturbations. Second, the overall channel response $\mathbf{h}_k^{\text{RIS}^{\text{H}}}\boldsymbol{\Theta}\mathbf{H}$ can be estimated by sending pilot symbols from BS toward actuators. Given that $\boldsymbol{\Theta}$ nd $\mathbf{H}$ are known, a matrix/vector arithmetic manipulation will result in $\mathbf{h}_k^{\text{RIS}^{\text{H}}}$ for each actuator. In this article, we have ignored the delay/overhead incurred in the CSI estimation phase at BS though in practice exists as the CSI acquisition and its challenges have been investigated thoroughly, e.g., in [37], [38], [39], and [40]. As an example, recently, the authors in [37] investigated a thorough comparison of algorithms to estimate the composite channels in RIS-aided systems with various assumptions, e.g., with/without LoS links and multiple antenna receiver/transmitter set-ups. Additionally, deep learning has also achieved an exemplary performance in reducing the dimension of the CSI feedback [41].

In this work we assume *single-shot transmissions*, i.e., retransmissions are not considered [42], [43]. Thus, the transmission latency is equal to one transmission time interval, which can be as low as ~ 0.1 ms when adopting the flexible numerology introduced in 5G New Radio [44]. This assumption allows us to investigate the lower bound performance of the proposed URLLC system as retransmissions improve the system's reliability while at the cost of increasing latency [45]. Nevertheless, some studies have compared the retransmission schemes with single-shot transmission [43], [46]. As an example, the study in [46] employed an incremental redundancy hybrid automatic repeat request (IR-HARQ) and concluded that the energy saving of the system enhances in comparison with the single-shot transmission.

For the considered system model, the received signal at the actuator k in time instance t is

$$y_k[t] = \overbrace{\left(\mathbf{h}_k^{\text{RIS}^{\text{H}}}\boldsymbol{\Theta}\mathbf{H}\right)\boldsymbol{\omega}_k x_k[t]}^{\text{Actuator } k \text{ signal}} + \underbrace{\left(\mathbf{h}_k^{\text{RIS}^{\text{H}}}\boldsymbol{\Theta}\mathbf{H}\right)\sum_{i=1, i\neq k}^{K}\boldsymbol{\omega}_i x_i[t] + n_k[t]}_{\text{Interference plus noise}} \quad (8)$$

where $\boldsymbol{\omega}_k \in \mathbb{C}^{N\times 1}$ is the beamforming vector applied at the transmitter to the symbol $x_k[\cdot]$ of actuator k with $\mathbb{E}[|x_k|^2] = 1$. Also, $\|\boldsymbol{\omega}_k\|_2^2 = p_k$ in which p_k is the transmit power allocated for actuator k such that $\sum_{k=1}^{K} p_k = p_{\text{total}}$ is the BS transmit power, and $n_k[t]$ is the additive white Gaussian noise (AWGN) with $\mathbb{E}[|n_k[t]|^2] = N_0 W = \sigma^2$ where N_0 and W are the noise spectral density and the system bandwidth, respectively. The complex reconfiguration matrix $\boldsymbol{\Theta}_{N\times N}$ indicates the phase-shift setting of the RIS which is defined as

$$\boldsymbol{\Theta}_{N\times N} = \text{diag}\left(\beta_1 e^{j\theta_1}, \beta_2 e^{j\theta_2}, \ldots, \beta_N e^{j\theta_N}\right) \quad \beta_n \in [0, 1], \quad \theta_n \in [-\pi, \pi) \quad \forall n \in \mathcal{N} \quad (9)$$

where $\mathcal{N} = \{1, 2, \ldots, N\}$. Note that in our model we have assumed that the RIS elements have no coupling or there is

no joint processing among elements [10]. However, practical RIS phase shifters have phase-dependent amplitude response which is given by [13]

$$\beta_n(\theta_n) = (1-\beta_{\min})\left(\frac{\sin(\theta_n-\phi)+1}{2}\right)^{\alpha}+\beta_{\min} \quad (10)$$

where $\beta_{\min} \geq 0$ (minimum amplitude), $\alpha \geq 0$ (the steepness) and $\phi \geq 0$ (the horizontal distance between $-(\pi/2)$ and $\beta_{\min}$) are circuit implementation parameters. Note that, $\beta_{\min} = 1$ results in an ideal phase shifter.

Based on the received signal at actuator k in (8), the corresponding SINR achieved at time instance t is given by

$$\mathrm{SINR}_k = \frac{\left|\mathbf{h}_k^{\mathrm{RIS}^{\mathrm{H}}}\boldsymbol{\Theta}\mathbf{H}\boldsymbol{\omega}_k\right|^2}{\sum_{i=1,i\neq k}^{K}\left|\mathbf{h}_k^{\mathrm{RIS}^{\mathrm{H}}}\boldsymbol{\Theta}\mathbf{H}\boldsymbol{\omega}_i\right|^2+\sigma^2} \quad (11)$$

to cast the channel coefficients into one single matrix, and defining $\boldsymbol{\theta} = [\beta_1 e^{j\theta_1}, \beta_2 e^{j\theta_2}, \ldots, \beta_N e^{j\theta_N}]^{\mathrm{H}}$ the SINR expression in (11) can be rewritten as

$$\mathrm{SINR}_k = \frac{\left|\boldsymbol{\theta}^{\mathrm{H}}\tilde{\mathbf{H}}_k\boldsymbol{\omega}_k\right|^2}{\sum_{i=1,i\neq k}^{K}\left|\boldsymbol{\theta}^{\mathrm{H}}\tilde{\mathbf{H}}_k\boldsymbol{\omega}_i\right|^2+\sigma^2} \quad (12)$$

where $\tilde{\mathbf{H}}_k = \mathrm{diag}(\mathbf{h}_k^{\mathrm{RIS}^{\mathrm{H}}})\mathbf{H}$ and $\mathrm{diag}(\cdot)$ refers to constructing a diagonal matrix based on a vector input as the diagonal elements. Herein, we concatenate the beamforming vectors such that $\bar{\boldsymbol{\omega}} = [\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \ldots, \boldsymbol{\omega}_K] \in \mathbb{C}^{N\times K}$. According to the FBL theory, the number of information bits that can be transmitted through c_k channel uses over a quasistatic AWGN channel is given by [9]

$$L_k = c_k\mathrm{C}(\mathrm{SINR}_k) - \mathcal{Q}^{-1}(\varepsilon_k)\sqrt{c_k\mathrm{V}(\mathrm{SINR}_k)} + \log_2(c_k) \quad (13)$$

where $\mathrm{C}(\mathrm{SINR}) = \log_2(1+\mathrm{SINR})$ is the Shannon capacity which is defined in infinite blocklength regime and ε_k is the target error probability for actuator k while $\mathcal{Q}^{-1}(\cdot)$ is the inverse of Q-function defined as $\mathcal{Q}(x) = (1/\sqrt{2\pi})\int_x^{\infty} e^{-v^2/2}dv$. The channel dispersion is defined as

$$\mathrm{V}(\mathrm{SINR}_k) = \frac{1}{(\ln 2)^2}\left(1-\frac{1}{(1+\mathrm{SINR}_k)^2}\right). \quad (14)$$

Solving (13) in order to find the decoding error probability ε_k at the actuator k yields

$$\varepsilon_k = \mathcal{Q}(f(\mathrm{SINR}_k, c_k, L_k)) \quad (15)$$

where

$$f(\mathrm{SINR}_k, c_k, L_k) = \sqrt{\frac{c_k}{\mathrm{V}(\mathrm{SINR}_k)}}\left(\log_2(1+\mathrm{SINR}_k) - \frac{L}{c_k}\right). \quad (16)$$

Also, note that from (13) when the blocklength c_k asymptotically goes infinity, the achievable rate simplifies to the conventional Shannon capacity formula.

B. Problem Formulation

Optimizing the total FBL rate of the actuators while ensuring the transmission target error probability by configuring the phase matrix of the RIS, beamforming matrix at the BS under optimized CBL vector $\mathbf{c} = [c_1, c_2, \ldots, c_K]$ is essential in factory environments to meet URLLC stringent requirements. Toward this goal, we formulate the following OP:

$$\mathbf{P1}\ \max_{\bar{\boldsymbol{\omega}},\boldsymbol{\theta},\mathbf{c}}\ L_{\mathrm{tot}} = \sum_{k=1}^{K}\left[\mathcal{V}_k(\bar{\boldsymbol{\omega}},\boldsymbol{\theta},\mathbf{c}) - \mathcal{Q}^{-1}\left(\varepsilon_k^{\mathrm{th}}\right)\mathcal{W}_k(\bar{\boldsymbol{\omega}},\boldsymbol{\theta},\mathbf{c})\right]$$

$$\mathbf{s.t.}\ C_1\colon \theta_n \in [-\pi,\pi)\ \ \forall n\in\mathcal{N}$$

$$C_2\colon \beta_n = (1-\beta_{\min})\left(\frac{\sin(\theta_n-\phi)+1}{2}\right)^{\alpha}+\beta_{\min}\ \ \forall n\in\mathcal{N}$$

$$C_3\colon \sum_{k=1}^{K}||\boldsymbol{\omega}_k||_2^2 \leq p_{\mathrm{total}}$$

$$C_4\colon \sum_{k=1}^{K} c_k \leq C,\ \ c_k \geq c_k^{\min}\quad \forall k\in\mathcal{K}$$

where $\mathcal{V}_k(\bar{\boldsymbol{\omega}},\boldsymbol{\theta},\mathbf{c}) = c_k\mathrm{C}(\mathrm{SINR}_k) + \log_2(c_k)$, and $\mathcal{W}_k(\bar{\boldsymbol{\omega}},\boldsymbol{\theta},\mathbf{c}) = \sqrt{c_k\mathrm{V}(\mathrm{SINR}_k)}$. The objective is to maximize the total number of information bits across all actuators and the variables are the reflective phase-shift values of each element in $\boldsymbol{\theta}$ at the RIS. The aim of transmission in the FBL regime is to ensure the block error probability (BLER) at a target value which is equal to $\varepsilon_k^{\mathrm{th}}\ \forall k \in \mathcal{K}$ in the objective function. Thus, by maximizing the objective in **P1** while transmitting with the specified FBL rate, the target error probability can be ensured. The constraint C_1 denotes that the phase adjustment variable is chosen from the specified interval. C_2 implies the practical phase-shift model which affects the amplitude response of the RIS. The maximum transmit power at BS is expressed in C_3. Also, C_4 is the constraint for total available number of CBLs at each transmission interval which is limited to maximum value of C. In addition, the CBL variable for each actuator k must be at least $c_k^{\min}$ so that the FBL regime rate is valid.

It is observed from **P1** that it belongs to a class of nonlinear OP which is thoroughly challenging to solve due to presence of equality constraint C_2. It is rational to use DRL for such problems since in DRL, the solution to the problem is the output of the forward pass to the neural network, which is a computationally simple process since it is often a set of simple operations. Further, the training of the neural networks that is done in different steps is performed in the background. Once the training is completed, the neural networks are updated. Therefore, the process to find the optimized variables in our problems is only an inference of the neural networks [28]. Consequently, we employ a model-free DRL algorithm based on the TD3 algorithm described in the following section.

III. DRL-Based Formulation

A. Review on the Preliminaries

The goal of the agent in reinforcement learning (RL) is to *learn* to find an optimal policy that maps states to actions based on its interaction with the environment so that the accumulated discounted reward function over a long time is

maximized. A state contains all useful information from the sequence of observations, actions, and rewards. These kinds of problems are tackled by representing them as a Markov decision process (MDP) framework. An MDP is characterized by $(\mathcal{S}, \mathcal{A}, \mathcal{R}, \mathcal{P}_{s\to s'})$ in which $\mathcal{S}$ is the set of environment states, $\mathcal{A}$ denotes the set of possible actions, which for this case is defined in terms of the RIS phase-shift values, $\mathcal{R}$ is the reward function, and $\mathcal{P}_{s\to s'}$ is the transition probabilities from current state s to the next state s', $\forall s, s' \in \mathcal{S}$. Mathematically, a Markov property means that the probability of the next state (future state) is independent of the past given the present state. In RL algorithms, the environment can be fully or partially observable. In a fully observable environment, the agent directly observes the environment [47]. The aim of the agent is to find an optimal policy to maximize the accumulated and discounted reward function over time steps, i.e., to find π^* in which the set of states $\mathcal{S}$ is mapped into the set of actions $\mathcal{A}$ as $\pi^* : \mathcal{S} \to \mathcal{A}$. The optimal policy π^* maximizes the action-value function defined as

$$Q_\pi(s,a) = \mathbb{E}_\pi\left[\sum_{t=0}^{\infty} \gamma^t r_{t+k+1} | S_t = s, A_t = a\right] \tag{17}$$

where the variable $0 \le \gamma \le 1$ is the discount factor to uncertainty of future rewards, r_i is the acquired reward in step i and $\mathbb{E}_\pi[\,\cdot\,]$ denotes the expectation with respect to policy π. By invoking the Markov property and Bellman equation, (17) will be reformulated into

$$Q_\pi(s,a) = \mathbb{E}_\pi\left[r_{t+1} + \gamma \sum_{a'\in\mathcal{A}} \pi(a'|s') Q_\pi(s',a') | S_t = s, A_t = a\right] \tag{18}$$

which $\pi(a'|s')$ gives the probability of choosing action a' given that the agent is in state s', the optimal value for the action-value function can be achieved by [48]

$$Q_{\pi^*}(s,a) = \sum_{s'\in\mathcal{S}, r\in\mathcal{R}} \Pr(s'|s,a)\left(r + \gamma \max_{a'} Q_{\pi^*}(s',a')\right) \tag{19}$$

where $\Pr(s'|s,a)$ is the probability of transition to next state s' given that the agent is in the current state s and the taken action is a. In order to find the optimal policy in (19), one must have knowledge about the transition probabilities that are usually unknown due to the complicated environment structure.

One of the efficient model-free and off-policy actor–critic methods that deals with the continuous action-space is DDPG [49]. Four deep neural networks (DNNs) are employed in DDPG, two of them are for actor/critic networks and the other two are called target networks. The actor network directly gives the action by giving the states as inputs to a DNN with parameter set $\boldsymbol{\xi}^{\text{act}}$, i.e., $a = \mu(s; \boldsymbol{\xi}^{\text{act}})$ where $\mu(\cdot)$ denotes the deterministic policy meaning that the output is a value instead of a distribution. The critic network which is usually a DNN with weights $\boldsymbol{\xi}^{\text{crit}}$ evaluates the action-value function based on the action given by the policy network and the current state. Additionally, the target networks estimate the target action-values to avoid instabilities when minimizing the mean-squared Bellman error (MSBE) which is defined by [47]

$$\mathcal{L}\left(\boldsymbol{\xi}^{\text{crit}}, \mathcal{B}\right) \triangleq \mathbb{E}\left[\left(Q\left(s,a;\boldsymbol{\xi}^{\text{crit}}\right) - \texttt{target}\right)^2\right]$$
$$\texttt{target} = r + \gamma Q\left(s', \mu\left(s';\boldsymbol{\xi}^{\text{targ-act}}\right); \boldsymbol{\xi}^{\text{targ-crit}}\right) \tag{20}$$

where the expectation is performed over $(s,a,s',r) \sim \mathcal{B}$ in which $\mathcal{B}$ is the experience replay memory which stores the set of current states, actions, rewards, and the next states as a tuple (s,a,r,s') over previous steps. Also, $Q(s,a;\boldsymbol{\xi}^{\text{crit}})$ represents the action-value function parameterized by neural network weights $\boldsymbol{\xi}^{\text{crit}}$. From (20) the next target action $a' = \mu(s';\boldsymbol{\xi}^{\text{targ-act}})$ is used to calculate the target action-value $Q(s',a';\boldsymbol{\xi}^{\text{targ-crit}})$ with network weights $\boldsymbol{\xi}^{\text{targ-act}}$. Typically, the two target networks' weights are copied over from the main networks every some-fixed-number of steps by polyak averaging which is

$$\boldsymbol{\xi}^{\text{targ-act}} \leftarrow \tau\boldsymbol{\xi}^{\text{act}} + (1-\tau)\boldsymbol{\xi}^{\text{targ-act}} \tag{21}$$
$$\boldsymbol{\xi}^{\text{targ-crit}} \leftarrow \tau\boldsymbol{\xi}^{\text{crit}} + (1-\tau)\boldsymbol{\xi}^{\text{targ-crit}} \tag{22}$$

where $\tau << 1$ is the hyperparameter used to control the updating procedure.

B. Twin Delayed DDPG

Before proceeding with the TD3 method, we restate the following lemma from [20].

Lemma 1: For the true underlying action-value function which is not known during the learning process, i.e., $Q_\pi(s,a)$ and the estimated $Q(s,a;\boldsymbol{\xi}^{\text{crit}})$ the following inequality holds:

$$\mathbb{E}\left[Q\left(s, a = \mu\left(s;\boldsymbol{\xi}^{\text{act}}\right);\boldsymbol{\xi}^{\text{crit}}\right)\right] \ge \mathbb{E}\left[Q_\pi\left(s, a = \mu\left(s;\boldsymbol{\xi}^{\text{act}}\right)\right)\right]. \tag{23}$$

Based on Lemma 1, since the DDPG algorithm leverages the typical Q-learning methods, it overestimates the Q-values during the training which propagates throughout the next states and episodes. This effect deteriorates the policy network as it utilizes the Q-values to update its weights and hyperparameters and results in poor policy updates. The impact of this overestimation bias is even problematic with the feedback loop that exists in DRL methods where suboptimal actions might be highly rated by biased suboptimal critic networks. Thus, the suboptimal actions will be reinforced in the next policy updates. The TD3 algorithm introduces the following assumptions to address the challenges [20].

1) As illustrated in Fig. 2, TD3 recruits two DNNs for estimating the action-value function in the Bellman equation, then the minimum value of the output of Q-values is used in (20).
2) In this method, the target and policy networks are being updated less frequently than critic networks.
3) A regularization of the actions that can incur high peaks and failure to the Q-value in DDPG method is leveraged so that the policy network will not try these actions in the next states. Therefore, the action will be chosen based

on adding a small amount of clipped random noise to the selected action as given by

$$a' = \text{clip}\big(\mu(s'; \boldsymbol{\xi}^{\text{targ-act}}) + \text{clip}(\kappa', -c, +c), a_{\text{Low}}, a_{\text{High}}\big) \tag{24}$$

where $\kappa' \sim \mathcal{N}(0, \tilde{\sigma}_a^2)$ is the added normal Gaussian noise and a_{Low} and a_{High} are the lower and upper limit values for the selected action that is clipped to ensure a feasible action which may not be in the determined interval due to added noise. Also, the constant c truncates the added noise at inner stage to keep the target action close to the original action.

The detailed description of the TD3 is given in Algorithm 1. A central controller at the BS is collecting and processing the required information for the algorithm execution. First, the six DNNs are initialized by their parameter weights, i.e., the actor network $\boldsymbol{\xi}^{\text{act}}$, the critic networks $\boldsymbol{\xi}_i^{\text{crit}}$, $i \in \{1, 2\}$ coefficients are initialized randomly while the target actor and critic networks' parameters are determined by replicating the primary actor and critic networks' coefficients, respectively. Also, the empty experience replay memory with specified capacity is prepared and the discount factor γ, learning rates, soft update hyperparameter τ, maximum step size N_{steps} and episodes N_{episode} are determined. In the training stage, the reflective phase matrix at the RIS is randomly initialized. The current channel coefficients of the actuators is acquired and the state set is formed, correspondingly. Next, the action, i.e., the phase-shift matrix is collected from the output of the actor DNN with parameter set $\boldsymbol{\xi}^{\text{act}}$ by importing the current state vector as the input. Next, the observed reward, taken action, the current state s, and the next state s', i.e., the modified channels' coefficients in terms of the phase-shift values given by the actor network are recorded at the experience replay buffer. To update the DNNs, a mini-batch of stored experience memory is randomly selected, then, the target actions are computed via target actor DNN with weights $\boldsymbol{\xi}^{\text{targ-act}}$ and the target values are evaluated by selecting the minimum value of target critic DNNs' output which correspond to minimizing the loss function by performing gradient descent method. In addition, when it is time to update the actor and target networks, e.g., out of t' steps where typically $t' = 2$ (once in every two steps), the gradient ascent is employed to compute the new coefficients of DNNs, i.e., renewal of $\boldsymbol{\xi}^{\text{targ-act}}$, $\boldsymbol{\xi}^{\text{targ-crit}}$, and $\boldsymbol{\xi}^{\text{act}}$.

Algorithm 1: Twin-Delayed DDPG Algorithm

Input: The number of actuators, the RIS amplitude-phase response model, position of the BS and actuators in 2D-plane.
Output: Trained agent with DNNs' weight coefficients.

Initialization: Initial values for weights $\boldsymbol{\xi}^{\text{act}}$, $\boldsymbol{\xi}_1^{\text{crit}}$ and $\boldsymbol{\xi}_2^{\text{crit}}$, empty replay memory $\mathcal{B}$. Let $\boldsymbol{\xi}^{\text{targ-act}} \leftarrow \boldsymbol{\xi}^{\text{act}}$, $\boldsymbol{\xi}_1^{\text{targ-crit}} \leftarrow \boldsymbol{\xi}_1^{\text{crit}}$ and $\boldsymbol{\xi}_2^{\text{targ-crit}} \leftarrow \boldsymbol{\xi}_2^{\text{crit}}$, soft update coefficient τ, the discount factor γ, the learning rates, the maximum steps N_{steps}, and maximum episodes N_{episode};
for $e = 1, 2, ..., N_{episode}$ **do**
  Randomly initiate CBLs, and beamforming at RIS/BS;
  Collect current channel coefficients $\left\{\mathbf{H}, \mathbf{h}_k^{\text{RIS}}, \forall k\right\}$;
  for $t = 1, 2, ..., N_{steps}$ **do**
    Select action $a = \text{clip}(\mu(s; \boldsymbol{\xi}^{\text{act}}) + \kappa, a_{\text{Low}}, a_{\text{High}})$, where $\kappa \sim \mathcal{N}(0, \sigma_a^2)$;
    Perform the action a selected above;
    Observe next state s' and the reward value r;
    Store the tuple (s, a, s', r) in the replay memory $\mathcal{B}$;
    Sample a batch of tuple $\mathbb{B} \subset \mathcal{B}$ from experience replay memory;
    Compute target actions given as $a' = \text{clip}(\mu(s'; \boldsymbol{\xi}^{\text{targ-act}}) + \text{clip}(\kappa', -c, +c), a_{\text{Low}}, a_{\text{High}})$ where $\kappa' \sim \mathcal{N}(0, \tilde{\sigma}_a^2)$;
    Compute the target value $\texttt{target}(r, s') = r + \gamma \min_{i \in \{1,2\}} Q(s', a'; \boldsymbol{\xi}_i^{\text{targ-crit}})$;
    Update the critic networks by performing gradient descent for $i \in \{1, 2\}$ using

$$\frac{1}{|\mathbb{B}|} \nabla_{\boldsymbol{\xi}_i^{\text{crit}}} \sum_{(s,a,s',r) \in \mathbb{B}} \Big(Q(s, a; \boldsymbol{\xi}_i^{\text{crit}}) - \texttt{target}(r, s')\Big)^2,$$

    if *time to update policy network* $(t \ mod \ t')$ **then**
      Update the policy network by performing gradient ascent with

$$\frac{1}{|\mathbb{B}|} \sum_{s \in \mathbb{B}} \nabla_a Q(s, a = \mu(s; \boldsymbol{\xi}^{\text{act}}); \boldsymbol{\xi}_1^{\text{crit}}) \nabla_{\boldsymbol{\xi}^{\text{act}}} \mu(s; \boldsymbol{\xi}^{\text{act}}),$$

      Update the target networks with

$$\boldsymbol{\xi}^{\text{targ-act}} \leftarrow \tau \boldsymbol{\xi}^{\text{act}} + (1 - \tau) \boldsymbol{\xi}^{\text{targ-act}},$$

$$\boldsymbol{\xi}_i^{\text{targ-crit}} \leftarrow \tau \boldsymbol{\xi}_i^{\text{crit}} + (1 - \tau) \boldsymbol{\xi}_i^{\text{targ-crit}}, \quad \text{for } i \in \{1, 2\}.$$

    end
  end
end

C. Applying TD3 to Solve *P1*

A preliminary step to solve the problem **P1** with TD3 is to map the components and properly define the algorithm states, actions and the reward function. In this section, we investigate them in detail as follows.

1) States: The agent interacts with the environment to optimize the FBL rate performance while ensuring a target BLER. Hence, the agent only has knowledge about the local information about actuators, e.g., the channel coefficients. Consequently, the DRL agent state space is defined as the aggregation of the angle and magnitude components of the composite channel coefficients, previous step beamforming vectors, and interference terms. First, it is useful to denote the interference and the inner terms as

$$\boldsymbol{\Upsilon}_t^k = \boldsymbol{\theta}^{\text{H}}(t-1) \tilde{\mathbf{H}}_k \tag{25}$$

$$\Upsilon_t^{kk'} = \boldsymbol{\theta}^{\text{H}}(t-1) \tilde{\mathbf{H}}_k \boldsymbol{\omega}_{k'}(t-1) \tag{26}$$

where $\boldsymbol{\Upsilon}_t^k \in \mathbb{C}^{1 \times M}$ and $\Upsilon_t^{kk'} \in \mathbb{C}$. The current state s_t is constructed as follows:

$$\begin{aligned}
s_t &= s_t^1 \cup s_t^2 \cup s_t^3 \cup s_t^4 \\
s_t^1 &= \left\{ \left| \Upsilon_t^{kk'} \right|, \angle \Upsilon_t^{kk'} \mid \forall k, k' \in \mathcal{K} \right\} \\
s_t^2 &= \left\{ \|\boldsymbol{\Upsilon}_t^k\|_2, \|\boldsymbol{\omega}_k(t-1)\|_2, \angle \boldsymbol{\Upsilon}_t^k, \angle \tilde{\mathbf{H}}_k, \angle \boldsymbol{\omega}_k(t-1) \mid \forall k \in \mathcal{K} \right\} \\
s_t^3 &= \{\theta_n(t-1) \mid \forall n \in \mathcal{N}\} \\
s_t^4 &= r_{t-1}
\end{aligned} \tag{27}$$

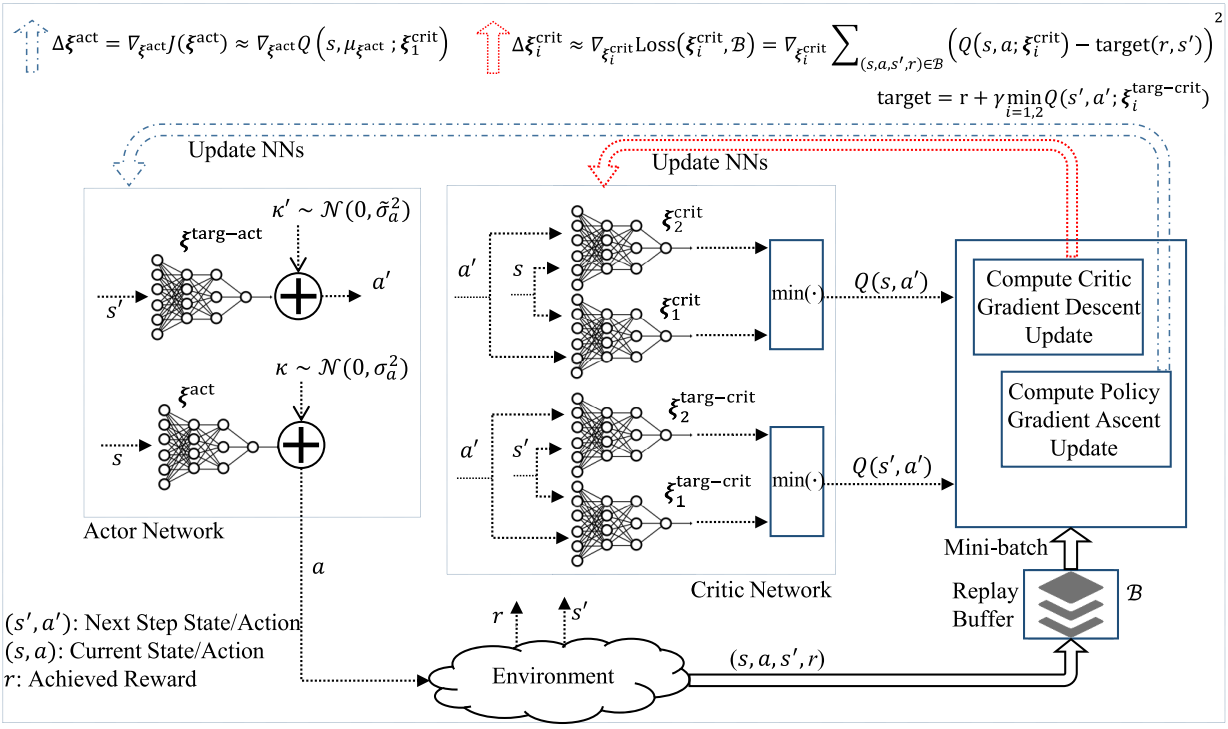

Fig. 2. Agent diagram of the TD3 method.

where $\mathcal{K} = \{1, 2, \ldots, K\}$ and $\mathcal{N} = \{1, 2, \ldots, N\}$. Note that the operators $\angle\mathbf{X}$ and $|\mathbf{X}|$ denote the angle and magnitude of each complex element in $\mathbf{X}$, respectively. The size of state space in (27) is determined based on $2K^2$ interference terms in s_t^1, $KM(N+2)+2K$ active beamforming coefficients and composite channel response from BS to the actuators in s_t^2, and N RIS reflection variables in s_t^3. Also, the previous reward achieved in the last step is considered as s_t^4 which will be defined in subsequent sections. Thus, the total size of the state space is given by $|s_t| = 2K(K+1)+(N+1)(KM+1)+KM$.

2) Actions: The action is determined as the value of phase shift at each element ($\theta_n(t)$, $\forall n$) and the action set in time t is given by

$$a_t = \{c_k(t), |\boldsymbol{\omega}_k(t)|, \angle\boldsymbol{\omega}_k(t) \mid \forall k \in \mathcal{K}\} \bigcup \{\theta_n(t) \mid \forall n \in \mathcal{N}\} \tag{28}$$

such that each phase-shift element value is chosen from the interval $\theta_n(t) \in [-\pi, \pi)$ $\forall n$ by multiplying the corresponding outputs of tanh(·) layer by π. Also, each beamforming vector is generated by producing complex numbers with separate magnitude values and angle components, then scaling the resultant vectors such that the total transmit power at the BS is satisfied, i.e., $\sum_{k=1}^{K} \|\boldsymbol{\omega}_k\|_2^2 = p_{\text{total}}$. To construct the actions corresponding to the CBLs, K elements of tanh(·) output layer in actor network are selected as

$$a_t^{\mathbf{c}} = \{a_1^{\mathbf{c}}, a_2^{\mathbf{c}}, \ldots, a_K^{\mathbf{c}}\} \tag{29}$$

where $-1 \le a_k^{\mathbf{c}} \le 1$ $\forall k$. Considering $\mathbf{c}^{\min} = [c_1^{\min}, c_2^{\min}, \ldots, c_K^{\min}]$ as the minimum CBL vector, the actions in (29) are scaled as follows to construct $\mathbf{c}(t)$:

$$\tilde{a}_t^{\mathbf{c}} \leftarrow \frac{a_t^{\mathbf{c}} + 1.0}{2}$$

$$\mathbf{c}(t) \leftarrow \frac{C - \mathbf{c}^{\min}}{\sum_{k=1}^{K} \tilde{a}_k^{\mathbf{c}} + \zeta} \tilde{a}_t^{\mathbf{c}} + \mathbf{c}^{\min} \tag{30}$$

where $\zeta << 1$ is a small value to avoid possible division by zero as $0 \le \tilde{a}_k^{\mathbf{c}} \le 1.0$ $\forall k$. Consequently, from (30) and the procedure to generate beamforming vectors, we can easily confirm that C_1–C_4 are satisfied. Finally, given (28), the output size of the actor network will be $K + 2KM + N$.

3) Reward Function: The objective function in **P1** has to be maximized over time steps t, i.e., L_{tot}. In addition, as explained in the previous section, by scaling the procedure of the raw actions, the constraints in **P1** can be met to produce feasible actions without reflecting their violation penalty into the reward function. Thus, the agent's reward function at each time step t is designed to be

$$r_t = \sum_{k=1}^{K} \Big[\mathcal{V}_k(\bar{\boldsymbol{\omega}}(t), \boldsymbol{\theta}(t), \mathbf{c}(t)) - \mathcal{Q}^{-1}\left(\varepsilon_k^{\text{th}}\right) \mathcal{W}_k(\bar{\boldsymbol{\omega}}(t), \boldsymbol{\theta}(t), \mathbf{c}(t)) \Big]. \tag{31}$$

In the following, we discuss the convergence proof for the TD3 algorithm in a finite MDP setting with discrete action-space referred to as clipped double Q-learning. It is worth noting that generalization to continuous action and actor–critic networks is straightforward. First, given Q^1 and Q^2 as the action-value estimator functions, the best action is determined based on $a^* = \arg\max_a Q^1(s', a)$. Also, the target value is found by the Bellman equation as $y = r + \gamma \min\{Q^1(s', a^*), Q^2(s', a^*)\}$. In double Q-learning, the action-value tables are updated as $Q^i(s, a) = Q^i(s, a) + \alpha_t(y -$

$Q^i(s,a))$, $i \in \{1,2\}$. Given this knowledge, the following theorem investigates the conditions for the convergence of clipped double Q-learning [20].

Theorem 1: The clipped double Q-learning will theoretically converge to the optimal action-value function Q^* with probability 1 if the following assumptions hold.

1) The MDP is of finite size and the action space is sampled infinite number of times.
2) The discount factor should be $\gamma \in [0,1)$ and the Q-values are stored in a look-up table.
3) The learning rate should meet $\alpha_t \in [0,1]$, $\sum_t \alpha_t = \infty$, $\sum_t \alpha_t^2 < \infty$.
4) Q^1 and Q^2 receive an infinite number of updates and $\mathbb{V}[r(s,a)] < \infty \ \forall s,a$.

Consequently, from the conditions in Theorem 1, we can ensure that to solve **P1** by utilizing the TD3 method, with proper selection of the learning rates, discount factor, and finite variance of the reward function the algorithm will converge to the optimized policy π^*. Since the reward function is the objective in **P1**, it is needed to verify that $\mathbb{V}[r_t(s_t,a_t)] < \infty$, therefore, we have

$$\mathbb{V}\Bigg[\overbrace{\sum_{k=1}^{K} \mathcal{V}_k(\bar{\boldsymbol{\omega}}(t),\boldsymbol{\theta}(t),\mathbf{c}(t))}^{A} - \overbrace{\Big(\sum_{k=1}^{K} \mathcal{Q}^{-1}\big(\varepsilon_k^{\text{th}}\big)\mathcal{W}_k(\bar{\boldsymbol{\omega}}(t),\boldsymbol{\theta}(t),\mathbf{c}(t))\Big)}^{B}\Bigg] \tag{32}$$

$$= \mathbb{V}[A-B] = \mathbb{V}[A] + \mathbb{V}[B] - \text{COV}[A,B]. \tag{33}$$

Given that the number of RIS elements is finite $N < \infty$, the BS has finite transmit power, and the CBL variables c_k $\forall k$ are bounded, then, the SINR values will have finite variance $\mathbb{V}[\text{SINR}] < \infty$ [11]. Thus, the reward function has finite variance $\mathbb{V}[A-B] < \infty$.

D. Complexity Analysis

In this section, we discuss the computational complexity of the TD3 to solve the **P1**. Let n_L be the number of layers in each DNN and z_l be the number of neurons in layer l. Then, in the training mode, the evaluation and update in one time step is $\mathcal{O}(|\mathbb{B}| \times \sum_{l=1}^{n_L-1} z_l z_{l+1})$ [27] where $|\mathbb{B}|$ denotes the size of the batch tuple. Since the TD3 algorithm has a finite number of DNNs and it takes $N_{\text{episode}} \times N_{\text{steps}}$ iterations to complete the training phase in which N_{steps} is the number of steps in each episode and N_{episode} is the total number of episodes. Therefore, the total computational complexity will be $\mathcal{O}(|\mathbb{B}| N_{\text{episode}} N_{\text{steps}} \sum_{l=1}^{n_L-1} z_l z_{l+1})$.

IV. Numerical Results

In this section, we numerically evaluate the considered joint active/passive beamforming and CBL allocation optimization via the TD3 method. A generic channel model is chosen to obtain insights about the proposed approach's performance trends independent of the operating frequency and employed channel model. Evaluations under specific channel models, including indoor factory scenarios and millimeter-wave channels, are left for future studies. Since the components and robots in industrial automation are usually stationary or have low mobility, we have considered four actuators in a factory environment located in 2-D plane coordinates at [16, 40] m, [32, 40] m, [48, 40] m, and [64, 40] m where a BS is positioned at [0, 0] m and the RIS is located at [40, 0] m. The large-scale path-loss fading is modeled as PL(dB) = $\text{PL}_0 - 10\nu \log_{10}(D[\text{m}])$ where $\text{PL}_0 = -30$ dB, $\nu = 2.2$ is the path-loss coefficient and D is the distance between the transmitter and the receiver [28]. For the sake of training tractability, the default number of RIS elements is set to $N = 16$ similar to other works in [28] and [32] in which by considering the number of users $K = 4$ and BS antennas $M = 4$ the total size of the output action in policy DNN will be $K + 2KM + N = 52$. Additionally, we have studied the impact of increasing RIS elements in the next figures for completeness. Table II shows the summary of the selected parameters for the network components during simulations.

TABLE II
Simulation Parameters

Parameter	Default value
Number of actuators (K)	4
Number of BS antennas (M)	4
Number of RIS elements (N)	16
BS transmit power (p_{total})	1.0 mW
Target error probability ($\varepsilon_k^{\text{th}}, \forall k$)	10^{-8}
Receiver noise figure (NF)	3 dB
Noise power density (N_0)	-174 dBm/Hz
Total available CBL (C)	100
Minimum CBL ($c_k^{\min}$, $\forall k$)	10
Bandwidth (W)	0.1 MHz
Rician factors (ζ and ζ_k^{RIS} $\forall k$)	10
BS height	12.5 m
BS location in 2D plane	[0, 0] m
RIS position in 2D plane	[40, 0] m
RIS phase shifter parameters	$\beta_{\min} = 0.4$ $\alpha = 1.9$ $\phi = 0.43\pi$

The learning rate in actor and critic networks of the TD3 agent is set to $\alpha_t = 10^{-4}$. The actor network DNN has three hidden dense layers with [800, 400, 200] neurons. The activation functions in all hidden layers are considered as rectified linear unit ReLU($\cdot$) except for the last layer in which the actor network is assumed to be tanh($\cdot$) to provide a better gradient. Since the output of tanh($\cdot$) is limited to the interval $[-1,1]$, it might get saturated for large inputs in most of the time. To avoid such saturation of the actions in the output of the actor network, the input state and action in the architecture of the critic networks are first processed by two dense layers with 800 neurons, separately. The implication behind this is that the actor network is being updated in the direction suggested by the critic, thus, proper estimation of Q-values is of paramount importance to avoid such occurrence. Next, the resultant outputs are added and are given to dense layers with size [600, 400] to estimate the current Q-value at

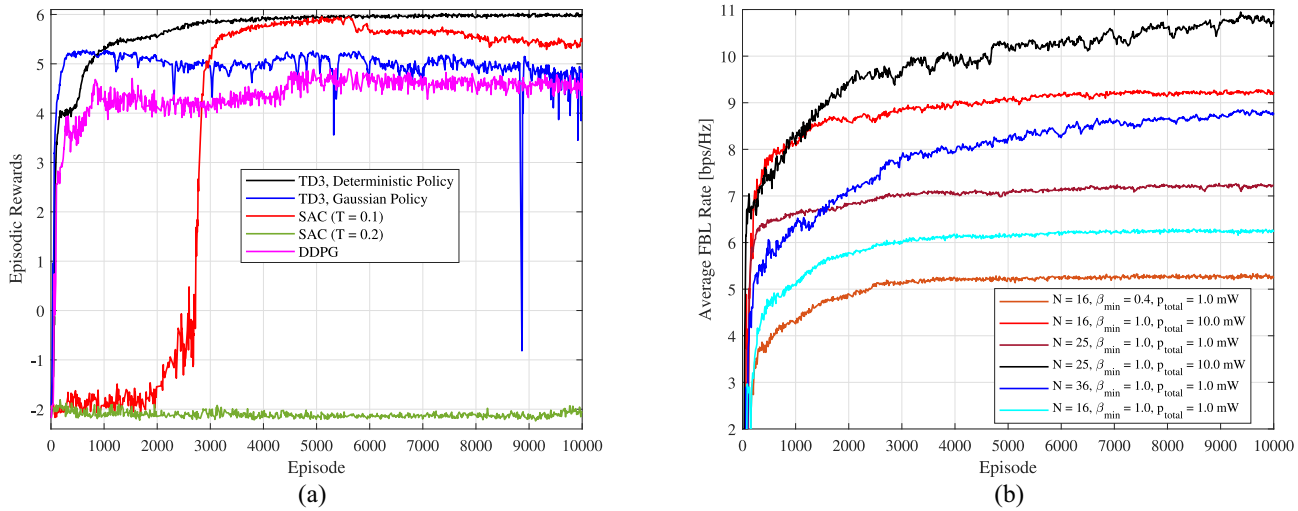

Fig. 3. (a) Comparison between SAC and TD3 with Gaussian/Deterministic policies. (b) FBL rate behavior versus episode for a different number of elements at the RIS and BS transmit power budget.

final stage. Also, extensive simulations revealed that employing Layer Normalization [50] helps to prevent the action value saturation, thus, we used this normalization technique before activation functions in dense layers.

The experience replay buffer capacity is 10 000 with batch size 64 such that the samples are uniformly selected from the buffer data. Furthermore, the exploration noises κ, κ' in TD3 actor networks are zero-mean normal random variables with variance $\sigma_a^2 = 0.1, \tilde{\sigma}_a^2 = 0.1$. The target actor/critic networks' soft update coefficient is $\tau = 0.005$. During the updating procedure, the policy network is updated every $t' = 4$ step. In all of the episodic illustrations, the agent is being evaluated over 100 independent realizations of the network channels to assess its performance, i.e., the illustrations are generalized results over 100 realizations.

In Fig. 3(a), the TD3 method is compared with the SAC algorithm with different entropy regularization coefficients ($T = 0.1, 0.2$) and DDPG. As observed, the DDPG has higher fluctuations in the curve of episodic average reward value compared to the TD3 algorithm. The fluctuations in the DDPG method occurred due to frequent policy network updates and the overestimation bias which are eliminated in TD3. In addition, TD3 outperforms the DDPG method in both final performance and learning speed in phase control. It can be observed that the SAC with higher regularization value cannot learn the optimal policy corresponding to too much exploration, however, for lower values of the coefficient, the agent started learning in around 3000 episodes, then the reward drops in around 5000 episodes. Also, the performance of employing Gaussian policy randomization at the output of the actor network is illustrated as well as utilizing deterministic policy. Basically, in deterministic sampling, the agent uses the mean action instead of a sample from fitting a Gaussian distribution with mean and variance dense layers. From illustrated curves, it is perceived that deterministic policy outperforms randomized policy as the agent has reached a higher reward value in the deterministic policy method. In addition, employing the Gaussian policy leads to some sudden drops in the reward function even in higher episodes and after training. This can be a harmful effect in our specific application scenario in factory automation where ensuring high reliability is of paramount importance. Also, Fig. 3(b) shows the convergence of the TD3 method with deterministic policy in terms of a different number of RIS elements and BS transmit power. It is observed that for either a higher number of RIS elements or a higher BS transmit power budget, the agent needs more episodes to learn the optimized policy.

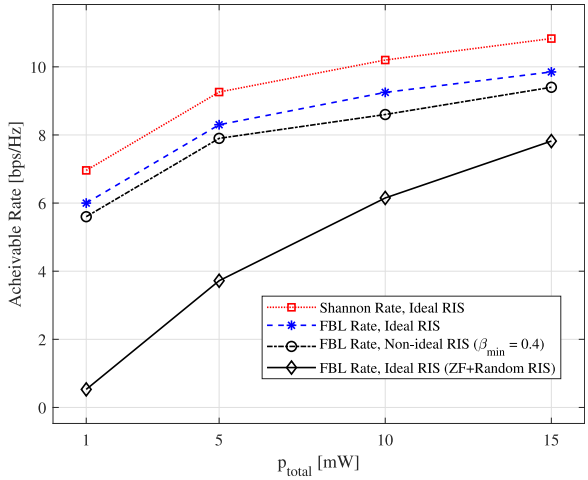

Fig. 4. Impact of increasing the BS transmit power on the converged average rate in FBL and Shannon regimes.

Fig. 4 shows the impact of increasing the BS transmit power on the average achievable rates in the Shannon/FBL regime. As it is demonstrated, the uppermost red curve shows the case that the RIS is ideal and the Shannon capacity expression is leveraged illustrating the upperbound performance of the network in the infinite CBL regime. It is also observed that increasing the transmit power budget at the BS leads to a higher total rate in all scenarios. On the other side, the performance of the system in the FBL regime with/without nonideal RIS is illustrated in the lower curves. The achievable

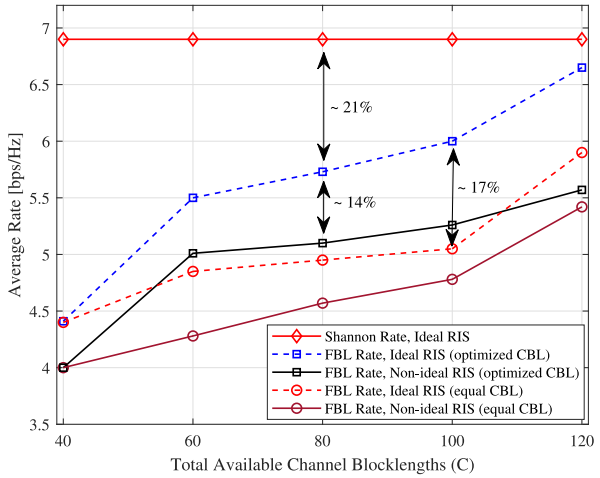

Fig. 5. Impact of increasing the total available CBL on the achievable FBL rate.

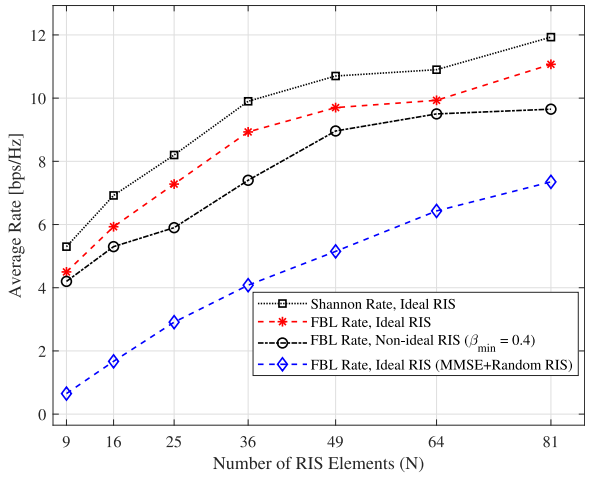

Fig. 6. Effect of increasing the number of the RIS elements on the total achievable rate of the system.

FBL rate by employing zero-forcing (ZF) precoding at the BS and uniformly distributed random phase shift at the RIS is also shown in the lowermost curve for comparison. The optimized CBLs are obtained by training an agent with significantly reduced action dimension, i.e., only $K = 4$ actions are generated while considering only constraint C_4 which is met by scaling of outputs as in (30). It is worth noting that the suggested optimized CBL vector via this agent is the same as employing the difference of concave optimization method as studied in [25]. Note that the ZF performs better in higher SNR regimes as the gap between curves reduces, i.e., the ZF precoder and optimized CBL and active/passive beamformers curves get closer as the total transmit power p_{total} increases. This highlights the applicability of our resource allocation framework in system-level design considerations to establish reliable communications in industrial environments.

Fig. 5 shows the achievable rate performance comparison in terms of the total available CBL. Since achievable rate expression in the Shannon regime is independent of varying total CBL, the uppermost curve has no variations versus changing C. The performance gap between working in the FBL regime and Shannon with either ideal or nonideal RIS is also highlighted. There is a 21% gap in the ideal RIS case and a 14% extra penalty due to having nonideal RIS. In addition, we have shown the case where the CBL variables are equally assigned between actuators, however, the active and passive beamforming vectors are being optimized. There is around a 17% gap between CBL optimization and equal CBL allocation. From another perspective, the CBL can be expressed in terms of transmission duration T and available bandwidth W as $c = TW$. Thus, utilizing fewer CBLs results in decreasing the transmission duration. This shows the importance of optimizing the CBL to preserve the possible FBL rate loss and reduce the transmission time to meet URLLC KPIs. Note that when $C = 40$, the optimized and nonoptimized curves overlap as the considered minimum CBL during simulations are $c_k^{\min} = 10\ \forall k$, $\sum_{k=1}^{4} c_k^{\min} = 40$.

Similarly, in Fig. 6, the network sum rate is assessed in terms of increasing the total number of reflective elements at the RIS. A gap is also observed between the Shannon achievable rate and FBL rate with either ideal or nonideal RIS. The Shannon and FBL regimes with nonideal RIS curves demonstrate that the system's actual performance will lie between these two curves. The performance of the TD3 method is compared with state-of-the-art linear minimum mean square error (MMSE) precoding at the BS. The optimized CBLs are obtained by applying a similar approach as in Fig. 4. The total achievable rate in all cases increases with the number of RIS elements, i.e., with/without ideal/nonideal RIS. A similar performance is also shown in FBL and Shannon rates. On the other hand, the slope of the curves is quite similar when the number of RIS elements starts to increase which additionally shows the practicality of the TD3 algorithm in ideal/nonideal reflective phase-shift design problems.

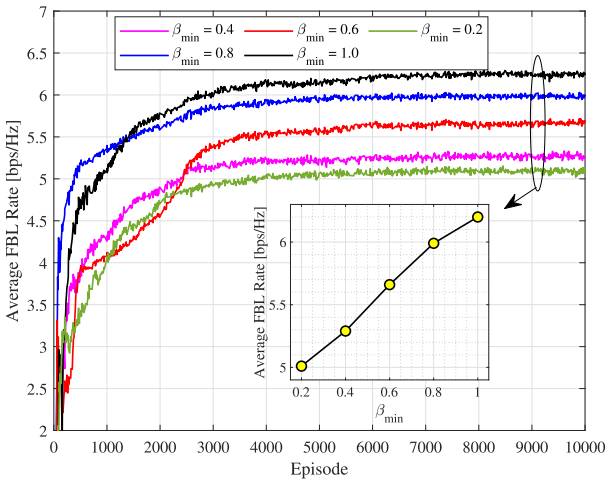

Fig. 7. FBL rate in terms of $\beta_{\min}$.

Finally, Fig. 7 shows the effect of $\beta_{\min}$ on the learning behavior of the TD3 agent. As can be seen from the curves, the agent reward function has converged in around 4000 episodes for all scenarios. In addition, there is a performance gap between $\beta_{\min} = 0.2$ and $\beta_{\min} = 1.0$ where the latter corresponds to the ideal RIS without amplitude attenuation. More

precisely, the achievable FBL rate has increased from 5 bps/Hz ($\beta_{\min} = 0.2$) to 6.2 bps/Hz ($\beta_{\min} = 1$) which is a 20% improvement.

V. Conclusion

We have studied the reflective phase-shift design, BS beamforming and CBL allocation problem in practical RIS-aided URLLC systems over short packet communications. The RIS impairments are modeled as the nonlinear amplitude response in terms of the phase-shift values, and the considered problem has been solved by utilizing a DRL algorithm, i.e., TD3 method. Since the proposed problem has highly nonlinear constraints due to considering practical phase-shift response, it is challenging to solve via optimization-based algorithms that are usually computationally inefficient even in ideal scenarios. Thus, we have employed a policy gradient DRL algorithm based on unsupervised actor–critic methods to optimize the active/passive beamforming and CBL allocation which concurrently learns a Q-function and a policy. The numerical results have demonstrated the applicability of the used DRL method in practical RIS phase-shift design problems in time-sensitive applications that exploit short packets in URLLC systems. Moreover, the TD3 method with deterministic policy outperformed other considered DRL algorithms such as SAC and Gaussian policy randomization in terms of final reward values and generalization of the policy network for different channel coefficients. In addition, we investigated the importance of optimizing the CBL in short packet communications and showed that the system total FBL can increase by 17% when the CBL variables are optimized for each actuator. As interesting future research, the formulated problem in this article can be studied under generalized assumptions, e.g., considering either uncertainties in channel coefficients and/or adaptability to change in actuators' positions by training an agent such that different network configurations are also reinforced.

References

[1] R. Hashemi, S. Ali, E. MoeenTaghavi, N. H. Mahmood, and M. Latva-Aho, "Deep reinforcement learning for practical phase shift optimization in RIS-assisted networks over short packet communications," in *Proc. Joint Eur. Conf. Netw. Commun. 6G Summit (EuCNC/6G Summit)*, Grenoble, France, 2022, pp. 1–5.

[2] C. She et al., "A tutorial on ultrareliable and low-latency communications in 6G: Integrating domain knowledge into deep learning," *Proc. IEEE*, vol. 109, no. 3, pp. 204–246, Mar. 2021.

[3] N. H. Mahmood et al., *White Paper on Critical and Massive Machine Type Communication Towards 6G* (6G Research Visions, No. 11), N. H. Mahmood, O. Lopez, O.-S. Park, I. Moerman, K. Mikhaylov et al., Eds. Oulu, Finland: Univ. Oulu, Jun. 2020. [Online]. Available: http://jultika.oulu.fi/files/isbn9789526226781.pdf

[4] A. Ranjha, G. Kaddoum, and K. Dev, "Facilitating URLLC in UAV-assisted relay systems with multiple-mobile robots for 6G networks: A prospective of agriculture 4.0," *IEEE Trans. Ind. Informat.*, vol. 18, no. 7, pp. 4954–4965, Jul. 2022.

[5] N. H. Mahmood, G. Berardinelli, E. J. Khatib, R. Hashemi, C. de Lima, and M. Latva-Aho, "A functional architecture for 6G special purpose industrial IoT networks," *IEEE Trans. Ind. Informat.*, early access, Jun. 14, 2022, doi: 10.1109/TII.2022.3182988.

[6] G. Aceto, V. Persico, and A. Pescapé, "A survey on information and communication technologies for industry 4.0: State-of-the-art, taxonomies, perspectives, and challenges," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3467–3501, 4th Quart., 2019.

[7] N. H. Mahmood, O. A. Lopez, H. Alves, and M. Latva-Aho, "A predictive interference management algorithm for URLLC in beyond 5G networks," *IEEE Commun. Lett.*, vol. 25, no. 3, pp. 995–999, Mar. 2021.

[8] P. Popovski et al., "Wireless access in ultra-reliable low-latency communication (URLLC)," *IEEE Trans. Commun.*, vol. 67, no. 8, pp. 5783–5801, Aug. 2019.

[9] Y. Polyanskiy, H. V. Poor, and S. Verdú, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inf. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[10] M. D. Renzo et al., "Smart radio environments empowered by reconfigurable intelligent surfaces: How it works, state of research, and road ahead," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 11, pp. 2450–2525, Nov. 2020.

[11] R. Hashemi, S. Ali, N. H. Mahmood, and M. Latva-Aho, "Average rate and error probability analysis in short packet communications over RIS-aided URLLC systems," *IEEE Trans. Veh. Technol.*, vol. 70, no. 10, pp. 10320–10334, Oct. 2021.

[12] R. Hashemi, S. Ali, N. H. Mahmood, and M. Latva-aho, "Average rate analysis of RIS-aided short packet communication in URLLC systems," in *Proc. IEEE Int. Conf. Commun. Work. (ICC Work.)*, Montreal, QC, Canada, 2021, pp. 1–6.

[13] S. Abeywickrama, R. Zhang, Q. Wu, and C. Yuen, "Intelligent reflecting surface: Practical phase shift model and beamforming optimization," *IEEE Trans. Commun.*, vol. 68, no. 9, pp. 5849–5863, Sep. 2020.

[14] H. Ren, K. Wang, and C. Pan, "Intelligent reflecting surface-aided URLLC in a factory automation scenario," *IEEE Trans. Commun.*, vol. 70, no. 1, pp. 707–723, Jan. 2022.

[15] Q. Wu and R. Zhang, "Beamforming optimization for wireless network aided by intelligent reflecting surface with discrete phase shifts," *IEEE Trans. Commun.*, vol. 68, no. 3, pp. 1838–1851, Mar. 2020.

[16] J. Wang et al., "Interplay between RIS and AI in wireless communications: Fundamentals, architectures, applications, and open research problems," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2271–2288, Aug. 2021.

[17] S. Ali et al., "6G white paper on machine learning in wireless communication networks," Apr. 2020, *arXiv:2004.13875*.

[18] S. Ali, N. Rajatheva, and W. Saad, "Fast uplink grant for machine type communications: Challenges and opportunities," *IEEE Commun. Mag.*, vol. 57, no. 3, pp. 97–103, Mar. 2019.

[19] M. Bennis, M. Debbah, and H. V. Poor, "Ultra-reliable and low-latency wireless communication: Tail, risk, and scale," *Proc. IEEE*, vol. 106, no. 10, pp. 1834–1853, Oct. 2018.

[20] S. Fujimoto, H. Hoof, and D. Meger, "Addressing function approximation error in actor-critic methods," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1587–1596.

[21] A. Ranjha and G. Kaddoum, "URLLC-enabled by laser powered UAV relay: A quasi-optimal design of resource allocation, trajectory planning and energy harvesting," *IEEE Trans. Veh. Technol.*, vol. 71, no. 1, pp. 753–765, Jan. 2022.

[22] W. R. Ghanem, V. Jamali, and R. Schober, "Joint beamforming and phase shift optimization for multicell IRS-aided OFDMA-URLLC systems," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Nanjing, China, 2021, pp. 1–7.

[23] A. Ranjha and G. Kaddoum, "URLLC facilitated by mobile UAV relay and RIS: A joint design of passive beamforming, blocklength and UAV positioning," *IEEE Internet Things J.*, vol. 8, no. 6, pp. 4618–4627, Mar. 2021.

[24] H. Xie, J. Xu, Y.-F. Liu, L. Liu, and D. W. K. Ng, "User grouping and reflective beamforming for IRS-aided URLLC," *IEEE Wireless Commun. Lett.*, vol. 10, no. 11, pp. 2533–2537, Nov. 2021.

[25] R. Hashemi, S. Ali, N. H. Mahmood, and M. Latva-Aho, "Joint sum rate and blocklength optimization in RIS-aided short packet URLLC systems," *IEEE Commun. Lett.*, vol. 26, no. 8, pp. 1838–1842, Aug. 2022.

[26] M. Almekhlafi, M. A. Arfaoui, M. Elhattab, C. Assi, and A. Ghrayeb, "Joint resource allocation and phase shift optimization for RIS-aided eMBB/URLLC traffic multiplexing," *IEEE Trans. Commun.*, vol. 70, no. 2, pp. 1304–1319, Feb. 2022.

[27] H. Yang, Z. Xiong, J. Zhao, D. Niyato, L. Xiao, and Q. Wu, "Deep reinforcement learning-based intelligent reflecting surface for secure wireless communications," *IEEE Trans. Wireless Commun.*, vol. 20, no. 1, pp. 375–388, Jan. 2021.

[28] K. Feng, Q. Wang, X. Li, and C. K. Wen, "Deep reinforcement learning based intelligent reflecting surface optimization for MISO communication systems," *IEEE Wireless Commun. Lett.*, vol. 9, no. 5, pp. 745–749, May 2020.

[29] C. Huang, R. Mo, and C. Yuen, "Reconfigurable intelligent surface assisted multiuser MISO systems exploiting deep reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 38, no. 8, pp. 1839–1850, Aug. 2020.

[30] J. Zhang, H. Zhang, Z. Zhang, H. Dai, W. Wu, and B. Wang, "Deep reinforcement learning-empowered beamforming design for IRS-assisted MISO interference channels," in *Proc. Int. Conf. Wireless Commun. Sign. Proc.*, Changsha, China, 2021, pp. 1–5.

[31] Y. Zhu et al., "Deep reinforcement learning based joint active and passive Beamforming design for RIS-assisted MISO systems," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Austin, TX, USA, 2022, pp. 477–482.

[32] A. Faisal, I. Al-Nahhal, O. A. Dobre, and T. M. N. Ngatched, "Deep reinforcement learning for optimizing RIS-assisted HD-FD wireless systems," *IEEE Commun. Lett.*, vol. 25, no. 12, pp. 3893–3897, Dec. 2021.

[33] C. Huang, G. Chen, Y. Gong, M. Wen, and J. A. Chambers, "Deep reinforcement learning based relay selection in intelligent reflecting surface assisted cooperative networks," *IEEE Wireless Commun. Lett.*, vol. 10, no. 5, pp. 1036–1040, May 2021.

[34] C. Huang et al., "Multi-hop RIS-empowered terahertz communications: A DRL-based hybrid beamforming design," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 6, pp. 1663–1677, Jun. 2021.

[35] X. Liu, H. Zhang, K. Long, M. Zhou, Y. Li, and H. V. Poor, "Proximal policy optimization-based transmit beamforming and phase-shift design in an IRS-aided ISAC system for the THz band," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 7, pp. 2056–2069, Jul. 2022.

[36] Y. Jia, C. Ye, and Y. Cui, "Analysis and optimization of an intelligent reflecting surface-assisted system with interference," *IEEE Trans. Wireless Commun.*, vol. 19, no. 12, pp. 8068–8082, Dec. 2020.

[37] 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.

[38] J. Mirza and B. Ali, "Channel estimation method and phase shift design for reconfigurable intelligent surface assisted MIMO networks," *IEEE Trans. Cogn. Commun.*, vol. 7, no. 2, pp. 441–451, Jun. 2021.

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

[40] L. Wei, C. Huang, G. C. Alexandropoulos, C. Yuen, Z. Zhang, and M. Debbah, "Channel estimation for RIS-empowered multi-user MISO wireless communications," *IEEE Trans. Commun.*, vol. 69, no. 6, pp. 4144–4157, Jun. 2021.

[41] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 748–751, Oct. 2018.

[42] N. H. Mahmood, N. Pratas, T. H. Jacobsen, and P. E. Mogensen, "On the performance of one stage massive random access protocols in 5G systems," in *Proc. 9th Int. Symp. Turbo Codes Iterative Inf. Process. (ISTC)*, Brest, France, Sep. 2016, pp. 340–344.

[43] A. Anand and G. De Veciana, "Resource allocation and HARQ optimization for URLLC traffic in 5G wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2411–2421, Nov. 2018.

[44] J. Sachs, G. Wikstrom, T. Dudda, R. Baldemair, and K. Kittichokechai, "5G radio network design for ultra-reliable low-latency communication," *IEEE Netw.*, vol. 32, no. 2, pp. 24–31, Mar./Apr. 2018.

[45] P. Popovski et al., "Wireless access for ultra-reliable low-latency communication: Principles and building blocks," *IEEE Netw.*, vol. 32, no. 2, pp. 16–23, Mar./Apr. 2018.

[46] A. Avranas, M. Kountouris, and P. Ciblat, "Energy-latency tradeoff in ultra-reliable low-latency communication with retransmissions," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 11, pp. 2475–2485, Nov. 2018.

[47] C. Szepesvári, "Algorithms for reinforcement learning," *Synth. Lectures Artif. Intell. Mach. Learn.*, vol. 4, no. 1, pp. 1–103, 2010.

[48] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 2018.

[49] T. P. Lillicrap et al., "Continuous control with deep reinforcement learning," 2015, *arXiv:1509.02971*.

[50] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," 2016, *arXiv:1607.06450*.