RIS-Empowered MEC for URLLC Systems with Digital-Twin-Driven Architecture

Sravani Kurma, *Student Member, IEEE*, Mayur Katwe, *Member, IEEE*, Keshav Singh, *Member, IEEE*, Cunhua Pan, *Member, IEEE*, Shahid Mumtaz, *Senior Member, IEEE*, and Chih-Peng Li, *Fellow, IEEE*

***Abstract*—This paper investigates a digital twin (DT) and reconfigurable intelligent surface (RIS)-aided mobile edge computing (MEC) system under given constraints on ultra-reliable low latency communication (URLLC). In particular, we focus on the problem of total end-to-end (E2E) latency minimization for the considered system under the joint optimization of beamforming design at the RIS, power, bandwidth allocation, processing rates, and task offloading parameters using DT architecture. To tackle the formulated non-convex optimization problem, we first model it as a Markov decision process (MDP). Later, we adopt deep deterministic policy gradient (DDPG) based deep reinforcement learning (DRL) algorithm to solve it effectively. We have compared the DDPG results with proximal policy optimization (PPO), modified PPO (M-PPO), and conventional alternating optimization (AO) algorithms. Simulation results depict that the proposed DT-enabled resource allocation scheme for the RIS-empowered MEC network using DDPG algorithm achieves up to 60% lower transmission delay and 20% lower energy consumption compared to the scheme without an RIS. This confirms the practical advantages of leveraging RIS technology in MEC systems. Results demonstrate that DDPG outperforms M-PPO and PPO in terms of higher reward value and better learning efficiency, while M-PPO and PPO exhibit lower execution time than DDPG and AO due to their advanced policy optimization techniques. Thus, the results validate the effectiveness of the DRL solutions over AO for dynamic resource allocation w.r.t. reduced execution time.**

***Index Terms*—Deep reinforcement learning, reconfigurable intelligent surface, mobile edge computing, digital twin, ultra-reliability and low-latency communication.**

I. Introduction

Mobile edge computing (MEC) has emerged as a potential solution to enable ultra-reliable low latency communication (URLLC) for various real-time processing and mission-critical applications such as autonomous driving, heterogeneous internet of things (IoT), remote surgery, and industrial automation [1]–[3]. URLLC requires low latency and high reliability, which are difficult to achieve in traditional centralized architectures [4]–[6]. MEC helps overcome these challenges by providing a more distributed and efficient network structure. With MEC, data can be processed locally at the edge, reducing the amount of data that needs to be transmitted over long distances. In particular, MEC allows decentralization of processing and storage, i.e., brings computing and storage resources closer to the end user, which enables faster and more reliable communication [7], [8]. Moreover, MEC provides a flexible and scalable infrastructure that can adapt to changing demands and support multiple applications with low-latency communication and high reliability, and thus constitute a driving technology for the fifth-generation and beyond wireless networks [9]–[13]. Although MEC offers energy-efficient and agile cloud services; however, radio access coverage and reliable task offloading become sensitive in line-of-sight (LoS) blockages and harsh fading environments.

Thanks to another frontier technology, called reconfigurable intelligent surface (RIS), which has been recently identified as the key enabler for smart propagation environment, especially under the poor channel and energy-constrained scenarios [14]–[16]. Generally, RIS is composed of a large number of passive elements made up of meta-surface, which can dynamically control the reflection of incident radio signals. These passive elements are connected to a control unit that can adjust their properties in real-time, and thus provide a highly flexible and efficient way to reconfigure the radio wave environment [17], [18]. RIS offers a number of benefits for MEC and URLLC, including improved communication quality and coverage, enhanced reliability, low latency, customizable and scalable solutions. These benefits make RIS a promising technology for 5G and beyond, enabling new and innovative communication scenarios and improving the performance and capabilities of future communication systems.

Besides, Metaverse has been envisaged to synergize and promote next-generation web and social networking applications by realizing a blended space of the physical and seamless virtual worlds [19], [20]. Interestingly, DT-enabled Metaverse architectures are recently contemplated as a holistic digital mapping technology of physical entities for intelligent resource allocation and network management in the system [7]–[11], [21], [22]. By incorporating DT into MEC, we can create virtual replicas of physical entities for real-time monitoring, analysis, and prediction [8]–[10]. This integration enables informed decision-making, improves resource management, and enhances operational efficiency. MEC's real-time analytics and low-latency processing amplify scalability, flexibility, and cost-efficiency. It also reduces network congestion, strengthens data privacy and security, and enhances user experiences.

This work was supported by the National Science and Technology Council (NSTC), Taiwan under Grants NSTC 112-2221-E-110-038-MY3 and NSTC 112-2221-E-110-029-MY3. (*Corresponding author: Keshav Singh.*)

Sravani Kurma, Mayur Katwe, Keshav Singh, and Chih-Peng Li are with the Institute of Communications Engineering, National Sun Yat-sen University, Kaohsiung 804, Taiwan (Email: sravani.phd.nsysu.21@gmail.com, mayurkatwe@gmail.com, keshav.singh@mail.nsysu.edu.tw, cpli@faculty.nsysu.edu.tw).

Cunhua Pan is with the National Mobile Communications Research Laboratory, Southeast University, Nanjing, China (Email: cpan@seu.edu.cn).

Shahid Mumtaz is with the Department of Applied Informatics, Silesian University of Technology Akademicka Gliwice, Poland and the Department of Engineering, Nottingham Trent University, U.K. (Email: dr.shahid.mumtaz@ieee.org).

Offline functionality ensures continuity in limited connectivity environments, making DT with MEC a compelling solution for process optimization and innovation [7], [8]. Overall, DT can provide a scalable, immersive, and efficient platform for MEC by enabling real-time interaction between users and digital objects and reducing the reliance on central nodes [7]–[13], [22], [23].

A. Background

Recent studies in [3], [24]–[29] reveal that the RIS deployment significantly boosts channel gain diversity which remarkably improves the MEC performance w.r.t. energy and spectral efficiency, task offloading rates or E2E delay, computations capabilities, and others when compared to the case without an RIS. The authors of [30] examined a simultaneously transmitting and reflecting RIS (STAR-RIS) system to minimize energy consumption in an MEC, and the optimization problem focused on transmission and reflection time and coefficients, as well as transmit power and data offloading size, to reduce total energy consumption. The authors of [27] studied the potential use cases of RIS for MEC systems and confirmed the intelligent beamforming design and resource allocation for RIS-aided MEC networks could effectuate the stringent requirements of emerging applications. For instance, the work in [3] demonstrated that the optimal beamforming design for the RIS-aided MEC system efficiently ameliorates the energy efficiency by 30-50% under given strict regulations on URLLC parameters. The authors of [2] proposed a new system design for an MEC that balances resources for local computation and task offloading while minimizing users' power consumption. It also introduces a user-server association policy and a two-time scale mechanism that improves reliability and delay performance.

Besides, recent works in [17], [18], [31]–[34] focused on developing deep reinforcement learning algorithms (DRLs) for optimizing the phase shift matrices of the RIS-based networks. For URLLC systems under a finite blocklength (FBL) regime, a novel twin-delayed deep deterministic policy gradient algorithm was used to maximize the total achievable FBL rate, considering feedback delay and phase shift constraints of the RIS [17]. A deep learning-based channel extrapolation was implemented over both antenna and time domains to reduce the pilot overhead, considering the acquisition of the time-varying cascaded channels in RIS-assisted communication systems [18]. The authors of [8]–[10] investigated the problem of latency minimization for task offloading associated with MEC for the industrial IoT. Similarly, a latency minimization problem for DT-assisted MEC was studied in [22] under the given constraints on the quality of services and computation resources in multi-MEC servers-based industrial IoT networks. Further, a DT framework was investigated in [7] for aerial vehicular networks to maximize the overall energy efficiency of roadside units while satisfying the dynamic requirements of resource demand. The authors of [12] studied the task offloading problem in UAV-enabled MEC using DT to minimize energy consumption. The optimization includes mobile terminal units association, UAV trajectory, transmission power, and computation capacity allocation using Double deep Q network (DDQN), closed-form expression, and an iterative algorithm.

B. Motivation and Contributions

Undoubtedly, the integration of DT and MEC offers numerous advantages, enhancing system performance and capabilities as demonstrated in [8]. While the considered DT-enabled MEC system in [8] is commendable, there exist several challenges while dealing with it, which are highlighted as follows:

1) **Resource Allocation Computational Complexity:** Moreover, integrating DT with MEC introduces complexities in terms of system integration. For example, MEC system comprises various components, including edge servers (ESs), network infrastructure, and user devices, which must seamlessly interact with the DT. Hence, the straightforward implementation of general alternating optimization (AO) solution presented in [8] may not be effective for large-scale networks.
2) **Worst-channel conditions:** In real-time scenarios, severe blockages often prevent the possibility of maintaining line-of-sight communication between the base station (BS) and user terminals (UTs). In such cases, relying solely on the direct link as considered in [8] for communication becomes impractical.
3) **Detailed Performance Analysis**: Indeed, the performance results for DT-enabled MEC in [8] are commendable and interesting; however, the detailed investigation of convergence analysis, time complexities of all algorithms, and examining the impact of factors such as bandwidth, number of BS antennas, number of users are majorly missing.

Essentially, the investigation of effective resource allocation schemes and catering of better channel conditions for users are imperative for optimizing MEC networks, serving as the primary motivation behind this work. Indeed, RIS-aided communication can enhance the efficiency and performance of the MEC by smartly reconfiguring the radio environment and enabling more reliable communication at the edge of the network [3], [25]–[30] while overcoming the challenges posed by worst-channel conditions. Noteworthy, the adoption of RIS with DT can tackle the aforementioned limitation of DT-MEC systems and can render a promising innovative communication solution for future MEC systems. The unconventional integration can result in an enhanced user experience by reducing latency, improving communication quality, and enabling real-time interaction in the virtual environment. The potential benefits of the integration of RIS and DT for MEC-driven URLLC are detailed below:

1) Improved Network Performance and User Experience: The integration of RIS and DT can lead to improved network performance by leveraging the ability of RIS to manipulate the radio wave environment and the virtual nature of the DT to provide a flexible and scalable network infrastructure.

TABLE I: Comparative summary of state-of-the-art and our work

Paper	RIS	MEC	Conventional algorithm	DRL	URLLC	DT	Performance metric
[3]	✓	✓	✓	✗	✓	✗	Energy efficiency maximization
[7]	✗	✓	✗	✓	✗	✓	Average Latency optimization
[8]	✗	✓	✓	✗	✓	✓	Total E2E latency minimization
[9], [10]	✗	✓	✓	✗	✓	✓	Latency minimization
[11]	✗	✓	✗	DTA, DDQN	✗	✓	Latency minimization
[12]	✗	✓	✗	DDQN	✗	✓	Energy consumption minimization
[13]	✗	✓	✗	DDN	✗	✓	Energy consumption minimization
[17]	✓	✗	✗	TD3	✓	✗	Total achievable FBL rate maximization
[18]	✓	✗	✗	✓	✗	✗	Normalized MSE for channel exploration
[21]	✗	✓	✓	✗	✓	✗	Overall latency optimization
[22]	✗	✓	✓	✗	✗	✓	Worst-case Latency minimization
[25]	✓	✓	✗	DDPG	✗	✗	The long-term computation offloading delay minimization
[27]	✓	✓	✓	✗	✗	✗	Latency minimization
[28]	✓	✓	✗	DDQN, DQN	✗	✗	Computing sum rate maximization
[29]	✓	✓	✓	✗	✗	✗	Average energy consumption minimization
[30]	✓	✓	✓	✗	✗	✗	Minimization of sum energy consumption
Our paper	✓	✓	✓	DDPG, PPO, M-PPO	✓	✓	Total E2E latency minimization

2) Moreover, it can result in an enhanced user experience by reducing latency, improving communication quality, and enabling real-time interaction in the virtual environment.
3) Increased Efficiency and Scalability: RIS and Metaverse can improve the efficiency of communication networks by reducing the reliance on central nodes and enabling faster and more reliable communication.
4) Improved Latency and Reliability: The URLLC constraints on FBL transmission can be made tighter (such as a reduction in block-length/packet size or packet error probability) with RIS-aided communication such that it can achieve much lower latency and higher reliability, along with the guaranteed QoS for all users when compared to the scheme without an RIS.

Despite its potential merits, there exist distinct research contributions in RIS-empowered MEC and Metaverse-empowered MEC as illustrated in Table I. To the best of the authors' knowledge, the detailed investigation of DT-driven architecture for RIS-assisted MEC systems for URLLC has not been addressed remarkably in the literature, which is quite interesting. Although interesting, the inclusion of RIS-aided communication in the DT-MEC systems makes the problem much more challenging to solve the original optimization problem for DT-MEC architecture in dynamic channel environments. Motivated by this background, we investigate an RIS-empowered DT-MEC-URLLC system to ensure stringent requirements of ultra-low latency and high-reliability task offloading for edge users with better energy efficiency. The major contributions of this paper are listed as follows:

1) Unlike [8], we investigate the RIS-empowered DT-MEC-URLLC system to captivate improved network performance of the MEC system. In particular, we focus on the total E2E latency minimization problem for the devised system subject to the given constraints on edge caching, task-offloading policies, transmit power, energy consumption at the UT, allocated bandwidth, data size, the processing rates of UT and ES, and RIS phase shifts matrices.
2) To leverage dynamic channel conditions, we model the formulated optimization problem as a Markov decision process (MDP) and later solve it using deep deterministic policy gradient (DDPG) based DRL algorithm and compare it with proximal policy optimization (PPO) and Modified PPO (M-PPO) algorithms. In particular, the DRL algorithms allow our system to learn and adapt to changing conditions, leveraging real-time data from the DT to make informed decisions and optimizations. This approach effectively tackles the complexities associated with the dynamic nature of DT and the integration challenges within the MEC system, demonstrating our strong motivation to overcome these obstacles.
3) Finally, we compare and analyze the computational complexity and latency performance of the DDPG, PPO, and M-PPO w.r.t. to the conventional AO approach. Our extensive results demonstrate that the DDPG based DRL algorithm exhibits lower execution time consumption than AO and achieves better minimum latency performance compared to the PPO and M-PPO algorithms. Moreover, M-PPO, followed by PPO has lower execution time than DDPG and AO due to their advanced policy optimization techniques. These findings highlight the effectiveness of DRL algorithms in optimizing the latency in the RIS-empowered DT-MEC-URLLC system. Moreover, we validate that the proposed RIS-empowered DT-MEC-URLLC system with optimal phase significantly reduces E2E latency and energy consumption compared to only DT-MEC system. This confirms the benefits and practicality of leveraging RIS technology in dynamic MEC systems.

C. Organization

The rest of this paper is organized as follows. Section II describes the considered RIS-empowered DT-MEC-URLLC system. The optimization problem formulation is presented in Section III. Section IV and Section V outline the proposed solution using the DRL and AO approaches, respectively.

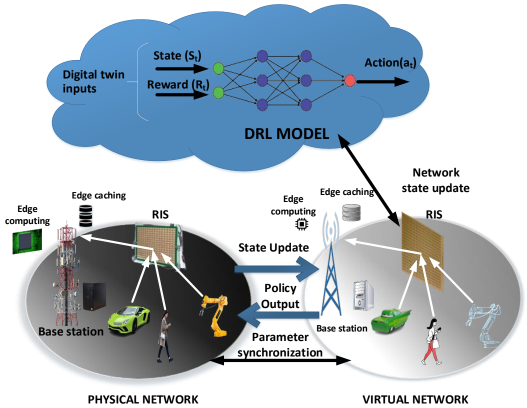

Fig. 1: An illustration of RIS-empowered DT-MEC-URLLC system.

Numerical simulations are presented in Section VI to verify the theoretical results. Finally, Section VII concludes this paper.

II. System Model

We consider an RIS-empowered DT-MEC-URLLC system where a set of M single-antenna UTs communicate with a multi-antenna BS for task offloading under given constraints on FBL transmission. It is assumed that all UTs lie in the dead zone such that there is no direct link between BS and the UT due to severe blockages, and the task offloading is carried out via an RIS consisting of N_R passive elements as shown in Fig. 1. The edge caching and task computations are carried out by the ES located at the BS, which renders low E2E latency for intensive task offloading from UTs. Table II summarizes the list of symbols. Note that the symbols associated with the algorithms are defined in the description of the algorithm itself.

Denoting $\mathbf{G}_1 = \zeta d_{BR}^{-\gamma}\mathbf{A}_1 \in \mathbb{C}^{L\times N_R}$ and $\mathbf{g}_{2,m} = \zeta d_{UR,m}^{-\gamma}\mathbf{a}_{2,m} \in \mathbb{C}^{N_R\times 1}, \forall m \in \mathcal{M} \triangleq 1,\ldots,M$ as the channel gain for RIS-BS and the m^{th} UT-RIS , respectively. Here d_{BR} and $d_{UR,m}$ are the distance between the RIS-BS and the m^{th} UT-RIS, $\{\zeta,\gamma\}$ are the large-scale model coefficients, and $\mathbf{A}_1$ and $\mathbf{a}_{2,m}$ are the corresponding small-scale fading coefficients. We assume that perfect channel state information (CSI) [1]of the whole system is available at the BS for resource allocation design [37]. Let us define the reflection-coefficient matrix, i.e., phase-shifter matrix at RIS as

$$\mathbf{\Phi} = \mathrm{diag}\left\{\Phi_1,\ldots,\Phi_{N_R}\right\}, \tag{1}$$

where $\phi_n \triangleq e^{j\theta_n}, \forall n \in \mathcal{N} \in \mathcal{N} \triangleq \{1,\ldots,N_R\}$ is the reflection coefficient and θ_n is the phase shift induced by the n^{th} RIS element. Overall, the effective channel gain for the m^{th} UT can be given as

$$g_m = \mathbf{G}_1\mathbf{\Phi}\mathbf{g}_{2,m}, \forall m \in \mathcal{M}, \tag{2}$$

For the sake of simplicity[2], we consider maximum-ratio combining (MRC) based beamforming at the BS owing to its low-computational complexity and near-optimal performance with a large number of BS antennas such that $\mathbf{z}_m = \mathbf{G}_1\mathbf{\Phi}\mathbf{g}_{2,m}, \forall m \in \mathcal{M}$, where $\mathbf{z}_m$ is the active receive beamformer at the BS. Now, the signal-to-noise (SNR) for the m^{th} UT can be given as

$$\Gamma_m\left(p_m, b_m, \mathbf{\Phi}\right) = \frac{p_m\left\|\mathbf{G}_1\mathbf{\Phi}\mathbf{g}_{2,m}\right\|^2}{b_m BW_0}, \tag{3}$$

where B is the bandwidth of the system, p_m denotes the power transmitted by the m^{th} UT, W_0 is the single-side noise spectral density, and b_m is the allocated bandwidth coefficient of the m^{th} UT.

Under FBL coding, the explicit relation between the maximum achievable rate (in bit/sec) and transmission latency (in secs) for the considered URLLC MEC system can be expressed as [13], [38]

$$\begin{aligned} R_m\left(p_m, b_m, \mathbf{\Phi}\right) \approx & \frac{B}{\ln 2}\Bigg[b_m \ln\left(1 + \Gamma_m\left(p_m, b_m, \mathbf{\Phi}\right)\right) \\ & - \sqrt{\frac{b_m V_m\left(p_m, b_m, \mathbf{\Phi}\right)}{\psi B}} Q^{-1}\left(\epsilon_m\right)\Bigg], \end{aligned} \tag{4}$$

$$T_m^{\mathrm{co}}\left(p_m, b_m, \mathbf{\Phi}\right) = \frac{D_m}{R_m\left(p_m, b_m, \mathbf{\Phi}\right)}, \tag{5}$$

where ψ and ϵ_m denote the transmission time interval and decoding error probability respectively, $Q(x)^{-1}$ is the inverse Q-function such that $Q(x) = \frac{1}{\sqrt{2\pi}}\int_x^\infty \exp^{\left(\frac{-t^2}{2}\right)} dt$, $\Gamma_m\left(p_m, b_m, \mathbf{\Phi}\right)$ represents the SNR of the m^{th} UT, V_m is the channel dispersion given by $V_m\left(p_m, b_m, \mathbf{\Phi}\right) = 1 - \left[1 + \Gamma_m\left(p_m, b_m, \mathbf{\Phi}\right)\right]^{-2}$, and D_m is the data size (in bits).

A. DT architecture

DT projects the physical world into the virtual world by replicating the physical objects with 3D digital facsimiles and companions using tools such as Automod, DELMIA, Modelica, FlexSim, etc. The DT communicates and controls the physical system effectively in real-time[3]. The proposed system is modeled by DT is presented as $DT = \{\hat{\mathcal{M}}, \hat{\mathcal{K}}\}$, where $\hat{\mathcal{M}}$, $\hat{\mathcal{K}}$ represent the virtual DT notation of M UTs and K ESs respectively. The automatic control (analyzing and optimizing the data) and management (collecting and visualizing the data) of the system are performed by DT based on real-time updated data from physical objects. DT architecture renders optimized solutions concerning estimated processing rates, allocated transmit power, and task offloading, improving overall system performance. Denote the processing rate estimated at the UT by f_m^{ut} and the difference between the real value and the estimated value of the processing rate is given by $\hat{f}_m^{\mathrm{ut}}$. The local processing at the m^{th} UT is served by DT and is defined by $\mathrm{DT}_m^{\mathrm{ut}} = \left(f_m^{\mathrm{ut}}, \hat{f}_m^{\mathrm{ut}}\right)$.

[1] The perfect channel state estimation is obtained by performing channel estimation techniques as given in [35], [36].

[2] Factly, the adopted MRC technique may not render optimal beamforming under many scenarios such as high SINR regime, imperfect CSI estimation, multi-user deployment, and others. However, the primary focus of this work is to study the impact of RIS on the MEC system in terms of latency, and the detailed study of the involved active receive beamforming has been excluded from this work.

[3] DTs of ES and MEC systems can estimate the performance of physical systems by constantly interacting with them and updating themselves with actual network topology and requests from mobile devices. The DTs can establish a digital representation system similar to the physical environment for obtaining the estimated performance value of the system without learning about the implementation details of mobile devices and ESs in the system [22], [39].

TABLE II: List of symbols

Symbols	Description	Symbols	Description
$\mathbf{G}_1$	Channel gain of RIS-BS link	M	Number of UTs
$\mathbf{g}_{2,m}$	Channel gain of m^{th} UT-RIS link	d_{BR}	Distance between the RIS and BS
$\{\mathbf{A}_1$ and $\mathbf{a}_{2,m}\}$	small-scale fading coefficients	T_m^{co}	Transmission latency (in secs)
ρ	Power parameter constant	$d_{UR,m}$	Distance between m^{th} UT and RIS
$\{\zeta, \gamma\}$	Large-scale model coefficients	ϕ_n	Reflection coefficient
N_R	Number of reflecting elements	θ_n	Phase shift induced by the n^{th} RIS element
g_m	Effective channel gain for the m^{th} UT	$\mathbf{z}_m$	Active receive beamformer at the BS
Γ_m	Signal-to-noise (SNR) for the m^{th} UT	B	Bandwidth of the system
p_m	Power transmitted by the m^{th} UT	W_0	Single-side noise spectral density
b_m	Allocated bandwidth coefficient of the m^{th} UT	R_m	Maximum achievable rate (in bit/sec)
T_m^{E2E}	Total end-to-end latency (in secs)	ψ	Transmission time interval
ϵ_m	Decoding error probability	Q^{-1}	Inverse Q-function
V_m	Channel dispersion	D_m	Data size (in bits)
$\hat{\mathcal{M}}$, $\hat{\mathcal{K}}$	Virtual DT notation of M UTs and K ESs	f_m^{ut}	Processing rate estimated at the UT
$\hat{f}_m^{\text{ut}}$	Difference between the real value and the estimated value of the processing rate	DT_m^{ut}	Local processing at the m^{th} UT that is served by DT
J_m	A tuple representing the task at the m^{th} UT	ς_m	Required cycles for computation
T_m^{max}	Maximum task latency	ϱ	Part of tasks that are locally performed at the UTs
$(1 - \varrho_m)$	Portion of the UT offloaded tasks executed by the ES	T_m^{ut}	Latency incurred in accomplishing a task at the m^{th} UT locally
$\tilde{T}_m^{\text{ut}}$	Estimated processing latency	ΔT_m^{ut}	Deviation latency
T_m^{es}	Latency incurred at the ES to perform the task offloaded from the m^{th} UT	f_m^{es}	Processing rate estimated at the ES
$\hat{f}_m^{\text{es}}$	Difference between the real processing rate and estimated processing rate at ES	$\mathbf{s} \triangleq \{s_m\}$	Integer decision variables
T_m	Edge caching enabled total E2E latency	E_m^{tot}	Total energy consumption
E_m^{cp}	Energy consumption for computation	E_m^{cm}	Energy consumption for communication
$F_{\text{max}}^{\text{ut}}$	Maximum processing rate at UT	$F_{\text{max}}^{\text{es}}$	Maximum processing rate at ES
$E_{\text{m}}^{\text{max}}$	Maximum energy budget at m^{th} UT	$\mathcal{S}_{\text{max}}^{\text{es}}$	Maximum edge computing capacity of ES
R_{min}	Minimum uplink rate	$\mathcal{S}$	Sequence of states
$\mathcal{A}$	Sequence of actions	$\mathcal{P}_{ss'}(a)$	State transition probability
$\mathcal{R}$	Sequence of rewards	ρ	Power parameter constant

B. Computation Model

A tuple representing the task at the m^{th} UT is defined as $J_m = \left(D_m, \varsigma_m, T_m^{\text{max}}\right)$, where ς_m and T_m^{max} denote the required cycles for computation and the maximum task latency, respectively. The part of tasks that are locally performed at the UTs is represented by $\varrho \triangleq \{\varrho_m\}_{\forall m}$ and the portion of the UT offloaded tasks executed by the ES is given by $(1 - \varrho_m)$.

The latency incurred in accomplishing a task at the m^{th} UT locally is expressed as

$$T_m^{\text{ut}}\left(\varrho_m, f_m^{\text{ut}}\right) = \frac{\varrho_m \varsigma_m}{f_m^{\text{ut}} - \hat{f}_m^{\text{ut}}}, \tag{6}$$

where $T_m^{\text{ut}} = \tilde{T}_m^{\text{ut}} + \Delta T_m^{\text{ut}}$. Here, $T_m^{\text{ut}} = \varrho_m \varsigma_m / f_m^{\text{ut}}$ and $\Delta T_m^{\text{ut}} = \varrho_m \varsigma_m \hat{f}_m^{\text{ut}} / \left[f_m^{\text{ut}}\left(f_m^{\text{ut}} - \hat{f}_m^{\text{ut}}\right)\right]$ represent the estimated processing latency and the deviation latency, respectively. Note that the deviation latency represents the additional time required due to the variability in the processing rate of the UTs.

Consequently, the latency incurred at the ES to perform the task offloaded from the m^{th} UT is presented as

$$T_m^{\text{es}}\left(\varrho_m, f_m^{\text{es}}\right) = \frac{(1 - \varrho_m)\,\varsigma_m}{f_m^{\text{es}} - \hat{f}_m^{\text{es}}},$$

where f_m^{es} is the processing rate estimated at the ES and, $\hat{f}_m^{\text{es}}$ denotes the difference between the real processing rate and estimated processing rate at the ES.

C. Total E2E Latency and Energy Consumption

In MEC, tasks are generated at UT and offloaded to ES for processing, which can cache data related to the task to reduce latency. Using integer decision variables, $\mathbf{s} \triangleq \{s_m\} \mid s_m \in \{0, 1\}, \forall m$, we characterize task caching techniques which specify the status of the task J_m. Two possible operations are observed based on the status of the cache at the ES, i.e., 1) if $s_m = 0$, then proceed with task offloading, and 2) if $s_m = 1$, then calculate edge processing latency.

Here, we consider only uplink transmission latency, as UTs receive the small controlled messages with more power from BS. Therefore, the edge caching enabled total E2E latency and the total energy consumption are formulated as

$$\begin{aligned} &T_m^{\text{E2E}}\left(\varrho_m, s_m, p_m, b_m, \mathbf{\Phi}, f_m^{\text{ut}}, f_m^{\text{es}}\right) \\ &\qquad = \frac{s_m \varsigma_m}{f_m^{\text{es}} - \hat{f}_m^{\text{es}}} + (1 - s_m)\left[T_m^{\text{ut}}\left(\varrho_m, f_m^{\text{ut}}\right)\right. \\ &\qquad\quad \left. + T_m^{\text{co}}\left(p_m, b_m, \mathbf{\Phi}\right) + T_m^{\text{es}}\left(\varrho_m, f_m^{\text{es}}\right)\right]. \end{aligned} \tag{7}$$

$$\begin{aligned} &E_m^{\text{tot}}\left(s_m, \varrho_m, f_m^{\text{ut}}, p_m, b_m, \mathbf{\Phi}\right) \\ &\quad = (1 - s_m)\left(E_m^{\text{cp}} + E_m^{\text{cm}}\right), \\ &\quad = (1 - s_m)\left[\varrho_m \frac{\rho}{2} \varsigma_m \left(f_m^{\text{ut}} - \hat{f}_m^{\text{ut}}\right)^2 + \frac{(1 - \varrho_m)\, p_m D_m}{R_m\left(p_m, b_m, \mathbf{\Phi}\right)}\right]. \end{aligned} \tag{8}$$

where E_m^{cp} and E_m^{cm} are the energy consumption for computation and communication, respectively, and ρ is the power parameter constant [8].

III. Problem Formulation

The prime objective of this work is to minimize the total E2E latency among M UTs for the considered RIS-empowered DT-MEC-URLLC system by jointly optimizing the RIS phase-shift matrix, portions of offloading tasks, allocated bandwidth at each UT, processing rate estimates at the UT and ES w.r.t. URLLC, energy consumption at the UTs, and edge caching parameters, i.e., caching policies and computing capacity. The problem for the total E2E latency can be formulated as

$$\min_{\substack{\varrho_m, s_m, b_m, p_m \\ \mathbf{\Phi}, f_m^{\mathrm{ut}}, f_m^{\mathrm{es}}}} \sum_{m=1}^{M} T_m^{\mathrm{e2e}}\left(\varrho_m, s_m, p_m, b_m, \mathbf{\Phi}, f_m^{\mathrm{ut}}, f_m^{\mathrm{es}}\right), \tag{9a}$$

$$\text{s.t.} \quad T_m^{\mathrm{e2e}}\left(\varrho_m, s_m, p_m, b_m, \mathbf{\Phi}, f_m^{\mathrm{ut}}, f_m^{\mathrm{es}}\right) \le T_m^{\max}, \forall m, \tag{9b}$$

$$\sum_{m=1}^{M} \left[s_m f_m^{\mathrm{es}} + (1 - s_m)(1 - \varrho_m) f_m^{\mathrm{es}}\right] \le F_{\max}^{\mathrm{es}}, \tag{9c}$$

$$E_m^{\mathrm{tot}}\left(s_m, \varrho_m, f_m^{\mathrm{ut}}, p_m, b_m, \mathbf{\Phi}\right) \le E_m^{\max}, \forall m, \tag{9d}$$

$$R_m\left(p_m, b_m, \mathbf{\Phi}\right) \ge R_{\min}, \forall m, \tag{9e}$$

$$\sum\nolimits_{m=1}^{M} b_m \le 1, \forall m, \tag{9f}$$

$$|\phi_n| \in 1, \quad \forall n \in \mathcal{N}, \tag{9g}$$

$$\sum\nolimits_{m=1}^{M} s_m D_m \le S_{\max}^{\mathrm{es}}, \tag{9h}$$

$$\begin{aligned} &\boldsymbol{\varrho} \in \mathscr{A}, \mathbf{p} \in \mathscr{P}, \mathbf{f} \in \mathscr{F}, \forall \boldsymbol{\varrho} \in \{\varrho_1, \ldots, \varrho_m\}, \\ &\mathbf{p} \in \{p_1, \ldots, p_m\}, \mathbf{f} \in \{f_1, \ldots, f_m\}, \end{aligned} \tag{9i}$$

where $\mathscr{P} \triangleq \left\{p_m, \forall m \mid 0 \le p_m \le P_m^{\max}, \forall m\right\}$, $\mathscr{A} \triangleq \{\varrho_m, \forall m \mid 0 \le \varrho_m \le 1, \forall m\}$, $E_m^{\max}$, and $\mathscr{F} \triangleq \left\{\mathbf{f} = \left\{f_m^{\mathrm{ut}}, f_m^{\mathrm{es}}\right\}, \forall m \mid 0 \le f_m^{\mathrm{ut}} \le F_{\max}^{\mathrm{ut}}, \forall m; 0 \le f_m^{\mathrm{es}} \le F_{\max}^{\mathrm{es}}\right\}$ are the uplink transmission power, the collection of offloading decisions constraints, the maximum energy consumption at UT, and the processing rates, respectively. The reflection coefficient of the n^{th} RIS element is denoted by the equation $\phi_n = e^{j\theta_n}$ with $0 \le \theta_n < 2\pi, \forall n$, where θ_n signifies the n^{th} RIS phase shift. Here, the constraints (9b) and (9c) indicate maximum latency requirements and the maximum computing capacity of ES, respectively. The maximum energy consumption requirement of the UT is described in constraint (9d). The constraints (9e) and (9f) represent the QoS for the uplink rate and the bandwidth allocation requirement, respectively. The constraint involving the RIS phase shift matrix is presented by (9g). Finally, constraint (9h) represents the maximum caching capability of ES and ensures that it does not exceed capacity, which affects E2E latency and system performance if exceeded [8], [22].

Primarily, the formulated resource allocation design problem in (9) is a nondeterministic polynomial time (NP)-hard problem that is intractable in closed-form due to the strong coupling of continuous and discrete variables, i.e., power allocation, edge caching parameters, bandwidth allocation, and others. In other words, the objective function and the respective constraint exhibit an implicit and coupled relationship with the optimization variables, which is hard to realize. Owing to the strong coupling of variables in the (9b),(9c), (9d), and (9e) of the formulated problems, the global optimal solution is hard to obtain and thus exhibits non-convex behaviour. In general, there exists no standard and systematic mathematical optimization schemes, which can provide the globally optimal or near-optimal solution for these non-convex problems in polynomial time. Although the exhaustive search may solve it, the implementation of a generic exhaustive search algorithm is not practically feasible as its computational complexity grows exponentially over the number of variables. Overall, it is imperative to solve it using sophisticated machine learning algorithms or necessary to transform the problem in (9) into some tractable sub-problems that can be solved separately and alternately over multiple iterations. In the sequel, we aim to tackle the aforementioned challenges and develop efficient and effective approaches to provide resource allocation for the problem in (9).

IV. DT-driven DRL Algorithms

The utilization of Deep Reinforcement Learning (DRL) in our research is motivated by its advantages for addressing the challenges of online task offloading in a fast fading channel environment. Conventional methods like AO or heuristic local search techniques have limitations, such as the risk of local optima and the impracticality of adapting to changing environments. The DRL-based methods offer computational efficiency advantages over AO due to its parallelizable architecture, end-to-end learning capability, model-free approach, and experience replay mechanism. DRL algorithms can leverage parallel computing architectures, learn directly from raw data, adapt to varying conditions, and achieve sample-efficient learning. While various DRL algorithms have been explored, including value-based methods like DQN and policy-based methods like PPO, we have chosen to adopt the Deep Deterministic Policy Gradient (DDPG) algorithm due to its importance in our problem. DDPG offers several key advantages for our task offloading scenario. Unlike DQN-based approaches, DDPG avoids the computational expense associated with exponentially growing wireless devices. Additionally, compared to PPO, DDPG demonstrates higher sampling efficiency in addressing the offloading problem, making it more practical for real-world applications. Moreover, DDPG is well-suited for problems with large state-action spaces, ensuring efficient exploration and decision-making. While the Soft Actor-Critic (SAC) algorithm has been considered for task offloading and offers advantages such as improved exploration and stability, its higher computational complexity limits its efficiency for large state-action spaces. In contrast, DDPG strikes a balance between efficiency and effectiveness, making it the preferred choice for our problem, which involves precise control actions in an evolving environment.

Here, we adopt the proposed DT-driven DDPG based DRL Algorithm as shown in Fig. 1, which reflects physical world objects and network parameters on the virtual ground and transmits the state of the network ($S(t)$), and later solves the

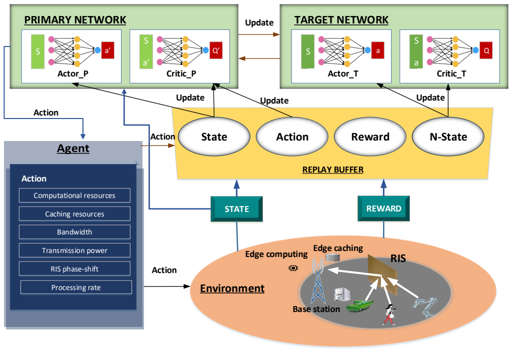

Fig. 2: An illustration of proposed DT-driven DDPG based DRL Algorithm

digitally-mapped dynamic optimization problem using DRL algorithm[4]. It leverages the benefits of both DT and DRL techniques to optimize the network parameters and physical world objects. This algorithm solves the MDP[5] by modeling the optimization problem as a sequence of states ($\mathcal{S}$), actions ($\mathcal{A}$), state transition probability ($\mathcal{P}_{ss'}(a)$ with $s = S_t, s' = S_{t+1} \in \mathcal{S}), a \in \mathcal{A}$, and rewards $\mathcal{R}$, i.e., $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R})$. Nevertheless, the use of the DRL algorithm is crucial in this case because it enables the agent to learn from its actions and the environment and adjust its strategy accordingly to get optimal policy π^* for minimizing the total E2E latency.

A. Proposed DDPG based DRL algorithm

The proposed DT-driven DDPG based DRL algorithm as illustrated in Fig. 2 is an innovative approach to solve dynamic optimization problems in a virtual environment. DDPG based DRL algorithm uses deep neural network (DNN) technique onto the deterministic policy gradient algorithm, which approximates deterministic policy function μ and action value function Q with neural network. There are two networks, namely, the primary network and the target network as shown in Fig. 2. The primary network adopt a critic network $\left(\text{Critic\_P} = Q'\left(S_t, a'_t \mid \theta^{Q'}\right)\right)$ and a actor network $\left(\text{Actor\_P} = \mu'\left(S_t \mid \theta^{\mu'}\right)\right)$ to represent the DNN model used for training, while the target network adopts a critic network $\left(\text{Critic\_T} = Q\left(S_t, a_t \mid \theta^{Q}\right)\right)$ and a actor network $\left(\text{Actor\_T}= \mu\left(S_t \mid \theta^{\mu}\right)\right)$ to represent the copy of the primary network used for making predictions during testing. Here $S(t)$ and $a \in \{a(t), a'(t)\}$ denote the state and the action of the networks, respectively. Generation of Q- learning targets requires the weights of the primary network and target network, which are given as $\theta^{Q'}, \theta^{\mu'}$ and θ^{Q}, θ^{μ} , respectively. The target network parameters are softly updated with the target update rate (τ) in each iteration as given in 12 and 13 steps of the Algorithm 1.

Algorithm 1 Proposed DT-driven DDPG-based DRL algorithm to solve the problem (9)

```
 1: Initialize:
        - Primary network: $Q'\left(S_t, a'_t \mid \theta^{Q'}\right)$ and
        $\mu'\left(S_t \mid \theta^{\mu'}\right)$ with weights $\theta^{Q'}$ and $\theta^{\mu'}$ .
        - Target network: $Q\left(S_t, a_t \mid \theta^{Q}\right)$ and
        $\mu\left(S_t \mid \theta^{\mu}\right)$ with weights $\theta^{Q} \leftarrow \theta^{Q'}$ and
        $\theta^{\mu} \leftarrow \theta^{\mu'}$.
        - Replay buffer $R_B$, phase shift matrix of the
        RIS $\mathbf{\Phi}$, learning rate $\tau$.
 2: for each episode do
 3:     Receive initial observation state $S_1$ with DT;
 4:     for each step [6] do
 5:         Select action $a_t$;
 6:         Observe reward $R_t$ and new state $S_{t+1}$;
 7:         Store $L_t$ transitions in $R_B \leftarrow (S_t, a_t, R_t, S_{t+1})$;
 8:         Sample a part of $R_B$ transition randomly
            $(S_i, a_i, R_i, S_{i+1})$;
 9:         Set $y_i$ according to (12);
10:         Update soft $Q$ by minimizing $L$ according to (11);
11:         Update the policy parameter $\mu$ according to (10);
12:         Update $\theta^{Q} : \theta^{Q} \leftarrow \tau\theta^{Q'} + (1-\tau)\theta^{Q}$;
13:         Update $\theta^{\mu} : \theta^{\mu} \leftarrow \tau\theta^{\mu'} + (1-\tau)\theta^{\mu}$;
14:     end for
15: end for
```

Algorithm 2 Proximal Policy Optimization (PPO) / Modified PPO (M-PPO)

```
 1: Initialize:
        - Policy $\pi$ with a parameter $\theta^{\pi}$.
        - Clipping parameter = 0.2.
 2: for each episode do
 3:     Receive initial observation state $S_1$ with DT;
 4:     for each step do
 5:         Select action $a_t$ based on $S_t$ using the current
            policy and execute;
 6:         Observe reward $R_t$ and new state $S_{t+1}$;
 7:         Collect set of trajectories with $L_t$ transitions$\leftarrow$
            $(S_t, a_t, R_t, S_{t+1})$;
 8:         Compute advantage function according to (20);
 9:         for each epoch do
10:             Update actor network:
11:                 Compute surrogate objective function ac-
                    cording to (19).
12:                 Update policy parameter using (21)
13:             if using Modified PPO then
14:                 Compute $\epsilon_t$ according to (22);
15:                 Compute surrogate objective function in
                    (19) by replacing $\epsilon$ with $\epsilon_t$;
16:             end if
17:         end for
18:     end for
19: end for
```

In particular, the proposed algorithm is designed as an actor-

[4] In the DRL approach, we assume that the same computation task with the same set of task parameters is processed in each time step over the optimization horizon.

[5] To tackle the formulated non-convex optimization problem, by considering the current state, action, and immediate reward, our approach adheres to the Markovian property, enabling us to formulate the problem as an Markov decision process (MDP). Later, we adopt DDPG based deep reinforcement learning (DRL) algorithm to solve it effectively.

critic network, where the primary actor network represents the agent, learns the optimal policy through trial and error, and outputs an action a_t based on the input environment state S_t. The primary critic network evaluates the actions taken by the agent and provides feedback in the form of a reward value r_t and the next state S_{t+1}. Then, the transitions (S_t, a_t, R_t, S_{t+1}) are collected in replay buffer R_B and a random mini-batch of L_t transitions (S_t, a_t, R_t, S_{t+1}) is selected for training the model using the replay buffer technique.

Further, the policy gradient is determined from the selection of L_t transition samples using the replay buffer technique, and thus, it is given by

$$\nabla_{\theta^{\mu'}} J \approx \frac{1}{L_t} \sum_{i=1}^{L_t} \nabla_a Q'\left(S, a \mid \theta^{Q'}\right)\Bigg|_{S=S_i, a=a_i} \nabla_{\theta^{\mu'}} \mu'\left(S \mid \theta^{\mu'}\right)\Bigg|_{S=S_i}. \tag{10}$$

To update the primary critic network, the loss function has to be minimized. By minimizing the loss function, the primary critic network is updated. The loss function can be presented as [40]

$$L_F\left(\theta^{Q'}\right) = \frac{1}{L_t} \sum_{i=1}^{L_t} \left(y_i - Q'\left(S_i, a_i \mid \theta^{Q'}\right)\right)^2, \tag{11}$$

where y_i is defined as

$$y_i = R_i + \lambda Q\left(S_{i+1}, \mu'\left(S_{i+1} \mid \theta^{\mu}\right) \mid \theta^{Q}\right). \tag{12}$$

Here, the discount factor $\lambda \in [0, 1]$ determines the trade-off between immediate and future rewards by exponentially decreasing the importance of future rewards.

The combination of DT architecture and DRL algorithm makes the proposed algorithm highly adaptable to changing environments and provides better learning and stability capabilities. Overall, the proposed algorithm has the potential to significantly improve the performance of dynamic optimization problems in virtual environments.

We define the framework for the proposed algorithm as follows:

- State Space: The state space components in DDPG are the variables that the agent uses to make decisions, and they depend on the environment. In this problem, as there exists no direct path, the state space is given by the indirect link channel gain as follows

$$S_t = \Big\{\mathbf{G}_1, \mathbf{g}_{2,1}, \mathbf{G}_1, \mathbf{g}_{2,2}, \cdots, \mathbf{G}_1, \mathbf{g}_{2,M}\Big\}. \tag{13}$$

- Action Space: The action space is defined as follows:

$$a_t = \{\varrho'_m, \mathbb{B}, \mathbf{\Phi'}, \mathbb{P}, \mathbb{S}, \mathbb{F}^{ut}, \mathbb{F}^{es}\}, \tag{14}$$

where all the terms associated with (14) are defined respectively as

$$\begin{aligned}
\varrho'_m &= \{\varrho_1(t), \varrho_2(t), \cdots, \varrho_M(t)\}, \\
\mathbb{B} &= \{b_1(t), b_2(t), \cdots, b_M(t)\}, \\
\mathbf{\Phi'} &= \{\mathbf{\Phi}_1(t), \mathbf{\Phi}_2(t), \cdots, \mathbf{\Phi}_M(t)\}, \\
\mathbb{P} &= \{p_1(t), p_2(t), \cdots, p_M(t)\}, \\
\mathbb{S} &= \{s_1(t), s_2(t), \cdots, s_M(t)\}, \\
\mathbb{F}^{ut} &= \{f_1^{ut}(t), f_2^{ut}(t), \cdots f^{ut}(t), f_M^{ut}(t)\}, \\
\mathbb{F}^{es} &= \{f_1^{es}(t), f_2^{es}(t), \cdots f^{es}(t), f_M^{es}(t)\}.
\end{aligned}$$

- Reward: The reward function of the proposed optimization problem is presented as

$$R_t = T_m^{\mathrm{E2E}}\left(\varrho_m, s_m, p_m, b_m, \mathbf{\Phi}, f_m^{\mathrm{ut}}, f_m^{\mathrm{es}}\right). \tag{15}$$

The state-value function V_S is defined by following the policy π at the state S as follows: $V_S^{\pi} = \mathbb{E}\{\mathcal{R} \mid S, \pi\}$, where $\mathbb{E}$ is the expectation operation. The state-action value Q is obtained when the agent at the state S takes action a following the policy π as follows:

$$Q^{\pi}(S, a) = \mathbb{E}(R(S, a)) + \lambda \sum_{s' \in \mathcal{S}} P_{ss'}(a) V_S(s'), \tag{16}$$

where π represents the agent policy. The notation $\mathbb{E}\{\cdot\}$ denotes the expectation function, which calculates the average value over all possible outcomes.

Algorithm 1 summarizes the proposed DT-driven DDPG-based DRL algorithm, which solves the total E2E latency minimization problem with multiple iterations until the convergence for the objective is achieved. Note that an episode in Algorithm 1 refers to a sequence of interactions between the agent and the environment. The initial observation state of S_1 is randomly chosen.

B. Proximal Policy Optimization (PPO)/Modified PPO (M-PPO)

The Algorithm 2 is a combination of the Proximal Policy Optimization (PPO) and M-PPO. PPO is a policy optimization algorithm that aims to find an improved policy by iteratively updating the policy based on observed trajectories [41], [42]. M-PPO extends PPO by incorporating modifications to the value function update. We define the policy by π with the parameter θ_π. Here, we train the policy and adjust the parameter to find an optimal policy π^* by running the SGD over a mini-batch of L_t transitions (S_i, a_i, R_i, S_{i+1}). The policy parameters are updated for optimizing the objective function as follows:

$$\theta_{i+1}^{\pi} = \underset{\theta^{\pi}}{\mathrm{argmax}} \frac{1}{L_t} \sum_{i=1}^{L_t} \nabla_{a_i} \mathcal{L}\left(S_i, a_i; \theta^{\pi}\right). \tag{17}$$

In both PPO and M-PPO algorithm, the agent interacts with the environment to find the optimal policy π^* with the parameter θ^{π^*} that maximizes the reward as

$$\mathcal{L}\left(S, a; \theta^{\pi}\right) = \mathbb{E}\left[\frac{\pi_{\theta^{\pi}}\left(S, a\right)}{\pi_{\theta^{old}}\left(S, a\right)} A^{\pi}(S, a)\right]. \tag{18}$$

Here, if we use only one network for the policy, the excessive modification occurs during the training stage. Thus, we use the clipping surrogate method as follows:

$$\begin{aligned}
\mathcal{L}^{\mathrm{clip}}\left(S, a; \theta^{\pi}\right) =& \mathbb{E}\big[\min\big(\frac{\pi_{\theta^{\pi}}\left(S, a\right)}{\pi_{\theta^{old}}\left(S, a\right)} A^{\pi}(S, a), \\
& \mathrm{clip}\left(\frac{\pi_{\theta^{\pi}}\left(S, a\right)}{\pi_{\theta^{old}}\left(S, a\right)}, 1-\epsilon, 1+\epsilon\right) A^{\pi}(S, a)\big)\big],
\end{aligned} \tag{19}$$

where $\pi_{\theta^{\pi}}\left(S, a\right)$ denotes the policy distribution of the actor network, $\pi_{\theta^{old}}\left(S, a\right)$ represents the policy distribution of the

TABLE III: Complexity Analysis of DDPG, PPO, and M-PPO

Algorithm	Complexity
DDPG	$O[(10N+9)H_1 + H_1H_2 + H_2 + (9N+9)H_1$ $+ H_1H_2 + H_2N]$
PPO	$O[2((9N+9)H_1 + H_1H_2) + H_2(N+1) + \epsilon]$
M-PPO	$O[2((9N+9)H_1 + H_1H_2) + H_2(N+1) + \epsilon_t]$

old actor network, and ϵ is a clipping parameter. The advantage estimate A^π is formulated as

$$A^\pi = R_t + \lambda V(S_{t+1}) - V(S_t), \tag{20}$$

where R_t is the observed return and $V(S_t)$ is the estimated value of the state S_t.

The policy is then trained by a mini-batch L_t and the parameters are updated by

$$\theta^{i+1} = \underset{\theta_\pi}{\operatorname{argmax}} \mathbb{E}\left[\mathcal{L}^{\text{clip}}(s, a; \theta_\pi)\right]. \tag{21}$$

In M-PPO with adaptive clipping, the adaptive clipping parameter ϵ_t is introduced, which is dynamically adjusted based on the policy's behavior. The adaptive clipping parameter is derived from the Kullback-Leibler (KL) divergence between the current policy and the old policy:

$$\epsilon_t = \epsilon \cdot \operatorname{sign}\left(\operatorname{KL}\left(\pi_{\theta_{\text{old}}} \| \pi_{\theta_t}\right) - \delta\right). \tag{22}$$

Here, δ is a small constant and $\operatorname{KL}\left(\pi_{\theta_{\text{old}}} \| \pi_{\theta_t}\right)$ represents the KL divergence between the old policy and the current policy. In M-PPO, the adaptive clipping parameter ϵ_t replaces the fixed threshold ϵ used in the surrogate objective function for policy update of the standard PPO. By dynamically adjusting the ϵ_t based on the KL divergence, M-PPO ensures that the policy updates are controlled and aligned with the current policy's behavior.

This adaptive clipping mechanism allows for more accurate and stable policy updates. It prevents excessive updates when the policy deviates significantly from the old policy, and it allows larger updates when the policy is closer to the old policy. By dynamically adjusting the clipping parameter, M-PPO can adapt to different scenarios and improve the stability and convergence properties of the optimization process. The algorithm continues to iterate over episodes and steps, updating the networks and improving the policy through PPO and M-PPO until convergence is achieved.

C. Computational Complexity

In the table III, we provide the complexity analysis for DDPG, PPO, and M-PPO algorithms. The table includes the dimensions of input layer, hidden layers, and output layer for each network involved in the algorithms. The complexity analysis table provides an overview of the computational complexity of the DDPG, PPO, and M-PPO algorithms. In the table, N represents the number of RIS elements. For DDPG, the dimensions of the input layer, the first hidden layer, the second hidden layer, and the output layer in the critic network are $10N + 9, H_1, H_2$, and 1, respectively. In the actor-network, the dimensions of the input layer, the first hidden layer, the second hidden layer, and the output layer are $9N + 9$, H_1, H_2, and N, respectively. Consequently, the overall complexity of the DDPG algorithm can be expressed as $O[(10N+9)H_1 + H_1H_2 + H_2 + (9N+9)H_1 + H_1H_2 + H_2N]$.

For PPO, the actor network and critic network share similar dimensions. The input layer, the first hidden layer, and the second hidden layer of both networks are defined as follows: $9N + 9$, H_1, and H_2. The output layer for the actor and the critic networks are N and 1, respectively. Consequently, the complexity of the PPO algorithm is estimated as $O[2((9N + 9)H_1 + H_1H_2) + H_2(N + 1) + \epsilon]$, where ϵ represents the clip factor.

For M-PPO, the network dimensions remain the same as PPO, including the actor and critic networks. Therefore, except the clip factor ϵ_t, the complexity analysis for M-PPO is the same as that for PPO : $O[2((9N+9)H_1+H_1H_2)+H_2(N+1)+\epsilon_t]$.

V. Baseline Alternating optimization Framework

In this section, we use the conventional AO method to solve the optimization problem of minimizing the total E2E latency for the proposed system. Note that the problem in (9) is a non-convex optimization problem as its objective function and constraints are non-convex and exhibit strong coupling of coupled integer and continuous variables, thus making it highly computationally complex to solve using exhaustive search methods. In order to solve it, we decouple the joint optimization problem in (9) into different sub-problems. Then, a unified solution based on the AO method is proposed which jointly solves these sub-problems in an alternating and iterative manner. Clearly, the joint solution is developed in the following subsections by solving four sub-problems: caching policy optimization, offloading policy optimization, joint communication and computation resource optimization, and RIS phase-shift matrix optimization.

In our work, we employ a similar analysis to that described in [8] for optimizing the variables $\mathbf{s}$, ϱ, and $\{\mathbf{b}, \mathbf{p}, \mathbf{f}\}$. We refer to these optimization problems as subproblems, namely SP1, SP2, and SP3, respectively, as outlined in [8]. Here, we only provide the optimization solution for passive beamforming design, i.e., phase-shift optimization for RIS, and refer to it as SP4 problem. The details are as follows:

A. Phase-shift optimization

By fixing other variables, the problem of the phase-shift optimization is given by

$$\text{SP4:} \quad \min_{\substack{\mathbf{\Phi}^{(i)} | \mathbf{s}^{(i+1)}, \varrho^{(i+1)}, \\ \mathbf{b}^{(i+1)}, \mathbf{p}^{(i+1)}, \mathbf{f}_m^{\text{ut}\,(i+1)} \mathbf{f}_m^{\text{es}\,(i+1)}}} \quad \sum\nolimits_{m=1}^{M} T_m^{\text{e2e}}(\mathbf{\Phi}), \tag{23a}$$

$$\text{s.t.} \quad (9\text{b}), (9\text{d}), (9\text{e}), (9\text{g}). \tag{23b}$$

The problem in (23) is non-convex due to the non-convex objective function and constraints (9b) and (9e). Next, we approximate the non-convex parts $\sqrt{V_m}$ in the constraint by the first-order Taylor series. We introduce the new variable

γ_m as given as follows, where γ_{min} is the lower bound of the rate γ_m.

$$\begin{aligned}\gamma_m &= \frac{p_m \left\|\mathbf{G}_1 \mathbf{\Phi} \mathbf{g}_{2,m}\right\|^2}{b_m B W_0}, \\ &= \{\frac{p_m}{b_m B W_0}\}(\mathbf{G}_1 \mathbf{\Phi} \mathbf{g}_{2,m})^H \mathbf{G}_1 \mathbf{\Phi} \mathbf{g}_{2,m}, \\ &= K_3 (\mathbf{G}_1 \mathbf{\Phi} \mathbf{g}_{2,m})^H \mathbf{G}_1 \mathbf{\Phi} \mathbf{g}_{2,m}, \\ &\geq 2\Re\left\{\left(\Phi^{[i]}\right)^H \mathbf{H}_{mm} \mathbf{H}_{mm}^H \mathbf{\Phi}\right\} - ||\mathbf{H}_{mm} \mathbf{\Phi}^{[i]}||^2, \end{aligned} \tag{24}$$

where $\mathbf{H}_{mm} = \mathbf{G}_1 \mathbf{G}_{2,m}$ and $\mathbf{G}_{2,m} = \text{diag}\{\mathbf{g}_{2,m}\}$.

Morover, the constraint (9e) is approximated using $\gamma_m^{[i]}$ as

$$\frac{B}{\ln 2}\Bigg[b_m \ln\left(1+\gamma_m\right) - D_m Q^{-1}\left(\epsilon_m\right)\{[1-(1+\gamma_m^{[i]})^{-2}]^{-\frac{1}{2}} \\ \left[\left(1+\gamma_m^{[i]}\right)^{-3} \times \left(\gamma_m - \gamma_m^{[i]}\right) - \left(1+\gamma_m^{[i]}\right)^{-2} + 1\right]\}\Bigg] \geq R_{\min}, \tag{25}$$

where $D_m = \sqrt{\frac{b_m}{\phi B}} Q^{-1}\left(\epsilon_m\right)$.

Lastly, the non-convex objective function (23a) which includes $T_m^{\text{e2e}}\left(\varrho_m^{(i+1)}, s_m^{(i+1)}, b_m^{(i+1)}, p_m^{(i+1)}, f_m^{(i+1)}, \mathbf{\Phi}^i\right)$ can be approximately represented as follows, by using an inner approximation.

$$T_m^{\text{e2e}} \leq \mathcal{T}_{m1}^{(i+1)} \triangleq \left(1 - s_m^{(i+1)}\right)\Bigg[\frac{\varrho_m^{(i+1)} \varsigma_m}{f_m^{\text{ut}(i+1)} - \hat{f}_m^{\text{ut}(i+1)}} \\ + D_m \tau_m\left(b_m^{(i+1)}, p_m^{(i+1)}, \mathbf{\Phi}^i\right) + \frac{\left(1 - \varrho_m^{(i+1)}\right)\varsigma_m}{f_m^{\text{es}(i+1)} - \hat{f}_m^{\text{es}(i+1)}}\Bigg] \\ + \frac{s_m^{(i+1)} \varsigma_m}{f_m^{\text{es}(i+1)} - \hat{f}_m^{\text{es}(i+1)}}. \tag{26}$$

Overall, we approximate the problem in (23) into its equivalent convex form as

$$\begin{aligned} \text{SP4:} \quad & \min_{\substack{\mathbf{\Phi}, \boldsymbol{\gamma} | \mathbf{s}^{(i+1)}, \varrho^{(i+1)}, \\ \mathbf{b}^{(i+1)}, \mathbf{p}^{(i+1)}, \mathbf{f}_m^{\text{ut}\,(i+1)} \mathbf{f}_m^{\text{es}\,(i+1)}}} \quad T_{m1}^{(i))}, \\ \text{s.t.} \quad & \text{(9b), (9d), (9g), (24), (25),} \end{aligned} \tag{27}$$

which is solved iteratively until convergence.

Since the problem in (27) involves NM scalar decision variables and $4M+N$ linear or quadratic constraints. The average worst-case computational complexity of solving this problem at each iteration is on the order of $O\left((NM)^2\sqrt{4M+N}\right)$.

VI. Numerical Results And Discussions

Here, we investigate the system performance of the proposed DDPG based DRL algorithm for the RIS-empowered DT-MEC-URLLC system through extensive numerical simulation. Note that the simulation results for the DRL algorithm are carried out using Python 3.10.7 and TensorFlow 1.13.0, while the AO is implemented using the CVX toolbox [43] in MATLAB. We fix the locations of the BS and the RIS at [0,0] and [50,40], respectively, while UTs ($M = 15$ or $M = 30$) are deployed in 100 m × 100 m square area [2].

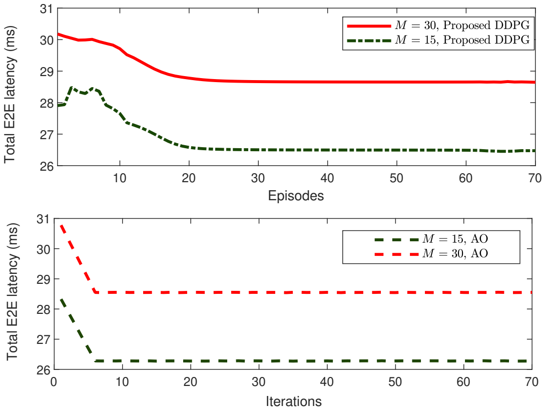

Fig. 3: Convergence of total E2E latency

TABLE IV: Simulation parameters

Hyper-parameter	**Value**
Learning rate for the critic-network	0.0002
Soft update coefficient	0.0005
Learning rate for actor-network	0.0001
Mini-batch size	64
Discount factor	0.9
Number of neurons for two hidden layers	[512, 512]
Replay buffer capacity	1000000
Variance of the action noise	0.1

The number of RIS elements and the number of antenna elements are set as $N_R = 32$ and $L = 8$, respectively. Furthermore, other simulation parameters are set as $B = 5$ MHz, W_0= -163 dBm/Hz , E_m^{max}= 3 mJ, S_{es}^{max}= 40 Kb, D_m = 1354 bytes, $\eta_m \triangleq \frac{\varsigma_m}{D_m}$ = [100,300] cycles/byte, F_{lol}^{max} = 1.5 GHz, F_{es}^{max}= 30 GHz, T_m^{max} = 10 ms, P_m^{max} = 23 dB, $\theta = 10^{-16}$ Watt-sec^3/cycle^3, $\epsilon = 10^{-7}$ [8]. The path loss parameters are set as $\zeta = 0.5$ and $\gamma = 2$ and the small-scale fading coefficients are assumed to be Rayleigh distributed. Moreover, we model the neural network setup with various hyper-parameters as [44], which are given in Table IV. As a performance benchmark, we compare the performance of the proposed DDPG based DRL algorithm in Algorithm 1 for the considered RIS-empowered DT-MEC-URLLC system with random-phase shift design and without RIS case. Although, we consider that the UTs are severely blocked due to blockages, however, in order to evaluate the benefit of RIS links, we consider that there exists a weak LOS path between the BS and UTs such that the path loss parameters for the direct link are set as $\zeta = 0.6$ and $\gamma = 1.8$.

Firstly, we examine the convergence behavior of the proposed algorithm with varying numbers of episodes in the proposed DDPG based DRL framework, and the number of iterations in the AO approach[7] is illustrated in Fig. 3. For

[7] Note that the input parameters for AO are chosen based on the values at which the DRL algorithm tends to converge. This approach allows us to compare the performance of AO and DRL, although it may not be a completely fair comparison due to the different input parameter settings. It is important to note that the comparison between AO and DRL may not be entirely fair, as the input parameters for AO are chosen based on the convergence behavior of the DRL algorithm. However, despite this limitation, our work fills a critical research gap as there is no existing literature that specifically addresses the system model we consider.

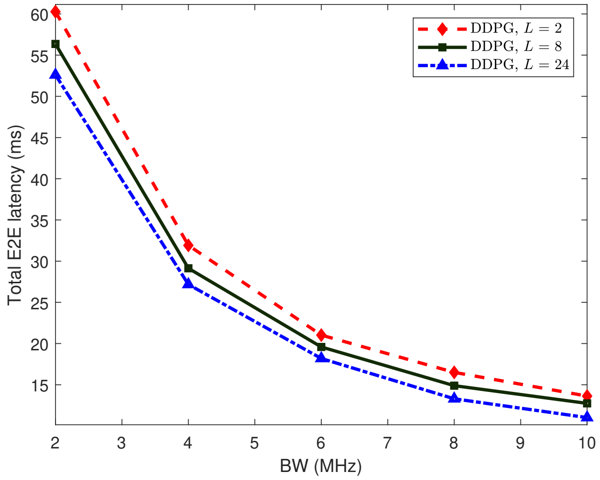

Fig. 4: Impact of bandwidth and number of BS antennas

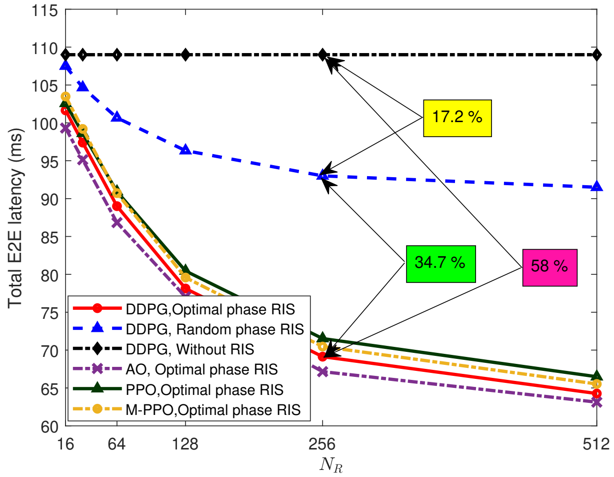

Fig. 5: Impact of number of RIS elements

TABLE V: Execution time (ms)

Algorithm	N_R =32	N_R =64	N_R =128	N_R =256
DDPG	24.52	24.85	25.53	26.50
AO	1283	1562	1985	3571
PPO	18.05	19.6	21.7	22.8
M-PPO	16.1	17.4	18.7	19.8

this analysis, we set S_{es}^{max} = 50kb and the task complexity parameter $\eta_m = 200$. Interestingly, the proposed DDPG based DRL algorithm requires a significantly less number of episodes for training as the results converge within 20 episodes only. This rapid convergence is due to several factors, such as the effectiveness of the DDPG algorithm in handling continuous action spaces, the use of target networks for stability, and the utilization of experience replay to break the correlation between consecutive samples, allowing for more efficient exploration and learning. The relatively small number of steps in each episode and the efficient exploration strategy contribute to the algorithm's ability to converge quickly. However, the convergence using the AO approach is attained within only 10 iterations. Importantly, the increase in the number of UTs increases the overall computational tasks, which further increases the total E2E latency for the considered system. Overall, the proposed scheme offers a cost-effective resource allocation with an acceptable offloading decision even with large numbers of UTs. Mainly, the AO-based algorithm is advantageous when the system parameters, such as channel state information, computation task parameters, and system constraints, are known or easily estimated. In such cases, AO can provide a computationally efficient solution by optimizing transmission powers, processing rates, and other variables in a single shot. This makes AO suitable for scenarios requiring real-time or near real-time optimization. Nevertheless, the AO algorithm serves as a valuable benchmark scheme for DRL-based algorithms in our work.

Table V presents a comparison of the execution times of various algorithms (DDPG, AO, PPO, and M-PPO) as the number of RIS elements varies. The DRL-based approaches dynamically allocate channels based on real-time traffic demands, potentially resulting in lower latency. In contrast, AO adopts predetermined rules for static channel allocation and follows an iterative and sequential optimization approach, which can be computationally expensive. The DRL-based method offers the advantage of learning policies directly from observations without explicit optimization problems, making it computationally efficient. Additionally, the utilization of parallelization and learning from experience enables faster convergence and improved performance. Table V clearly demonstrates that the running time of the AO algorithm increases with the number of RIS elements (N_R) due to its polynomial computation complexity, while the DRL algorithm's running time remains approximately constant. The DRL-based algorithm exhibits lower running time consumption and demonstrates better minimum latency performance, thus highlighting its effectiveness in comparison to AO. Moreover, M-PPO and PPO outperform DDPG in terms of execution times due to their enhanced optimization capabilities and convergence speed. These algorithms employ advanced policy optimization techniques, resulting in faster convergence to optimal solutions and reduced execution times. In summary, the table and figure provide strong evidence supporting the superiority of DRL algorithms in terms of running time efficiency and performance compared to AO.

Fig. 4 illustrates the impact of total transmission bandwidth on the system performance of the considered RIS-empowered DT-MEC-URLLC system w.r.t. latency. As bandwidth increases, the maximum number of tasks that can be executed in a specific time slot increases, which in turn results in a decrease in the computation latency of the system. Interestingly, the considered system initially undergoes high-performance amelioration up to 8 MHz, however, the performance gain becomes nearly trivial with a gradual increase of bandwidth. Moreover, we discuss the impact of increasing the number of BS antennas on the total E2E latency. Intuitively, the increase in the number of antennas increases the available phase-shifter at the BS, which improves the channel gain diversity, and this further decreases the E2E latency. Conclusively, the increase in available resources significantly increases the MEC per-

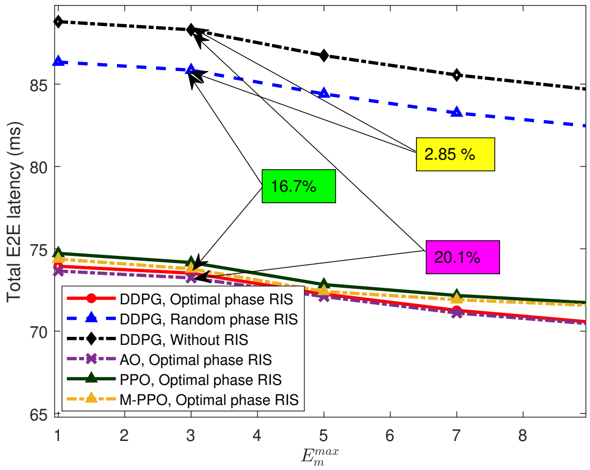

Fig. 6: Energy-consumption budget

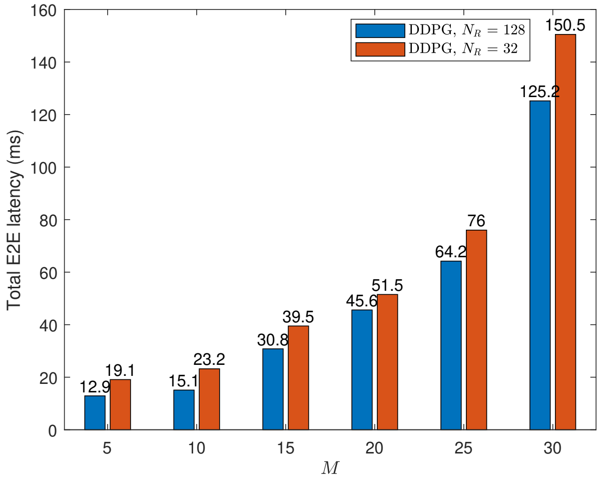

Fig. 7: Impact of number of UTs (M)

formance [28]. However, regarding the impact of increasing the number of BS antennas on the total E2E latency, we find that the decrease in latency is not significant with a gradual increase in bandwidth. As the number of antennas increases, the complexity of signal processing and coordination between antennas increases, which can introduce delays and overhead that offset the gains from improved channel gain diversity. Therefore, while increasing the number of BS antennas does improve the channel gain diversity and contribute to latency reduction, the diminishing returns from signal processing complexity and system overhead lead to a relatively modest decrease in the total latency.

Fig. 5 discusses the impact of a number of RIS elements on the system's latency performance. The increase in the number of passive elements at RIS results in improved channel gain diversity, i.e., high beamforming, which in turn improves the SNR-associated computational offloading and decreases the latency of the considered MEC system. The proposed beamforming design attains better performance gain when compared to the random phase-shift design. Moreover, the consideration of RIS deployment for the MEC URLLC system significantly outperforms the conventional MEC URLLC system without RIS. Conclusively, the optimal beamforming design with $N_R = 256$ renders approximately 35% better performance than random beamforming, moreover, the proposed RIS-empowered DT-MEC-URLLC system achieves 58% better performance than the case without an RIS. Primarily, the performance gain, i.e., low latency offloading, becomes dominant for large-scale RIS deployment ($N_R > 100$) and sophisticated beamforming design.

Moreover, we compare the latency performance of the considered MEC system w.r.t. optimal and random beamforming design at RIS and the case without RIS for varying energy consumption budget (E_m^{max}) as shown in Fig. 6. For the given data size, the increase in the maximum allowed energy consumption at the UTs decreases for all the considered schemes. Obviously, task offloading latency or rate can be improved at the expense of high transmit power consumption. Primarily, RIS-aided communication can ensure improved system performance under the given URLLC restrictions. Overall, the MEC system with RIS-aided communication ascertains lower latency while ensuring low energy consumption and high reliability when compared to those without the RIS scheme. Importantly, Fig. 5 and Fig. 6 depict that the total E2E latency of the proposed DDPG based DRL algorithm attains performance closer to the AO-based algorithm. We further investigated the performance of the DRL algorithms, specifically DDPG, PPO, and M-PPO with optimal phase, for the RIS-empowered DT-MEC-URLLC system. The results reveal that DDPG achieves the lowest latency among the three algorithms, followed by M-PPO and PPO. This can be attributed to DDPG's ability to effectively explore and exploit the action space, allowing it to converge to optimal policies more efficiently. Moreover, the adaptive clipping method employed in M-PPO improves performance compared to PPO by dynamically adjusting the clipping range, enabling better exploration and preventing policy divergence.

Fig. 7 illustrates the impact of the number of UTs on the total E2E latency. For this analysis, we set $S_{es}^{max} = 5$kb and the task complexity parameter $\eta_m = 900$. Intuitively, the amount of data that needs to be transmitted and processed increases with UTs, which in turn increases overall latency drastically. In other words, the increased demand for network resources results in a higher load on the network and processing resources, thus leading to increased E2E latency. Moreover, the increased number of UTs may also result in more network congestion, leading to longer waiting times for data transmission and processing. For $N_R = 32$, the latency starts at approximately 12.6 ms for five UTs and goes up to approximately 125.8 ms for 30 UTs. When compared to $N_R = 128$, it is observed that latency attains 19.1 ms for five UTs and reaches 150.5 ms for 30 UTs. The comparative results of $N_R = 32$ and $N_R = 128$ reveal that latency decreases with increased N_R, which is due to improved signal quality, reduced interference, and increased signal strength, as well as the ability to dynamically control the radio environment in real-time.

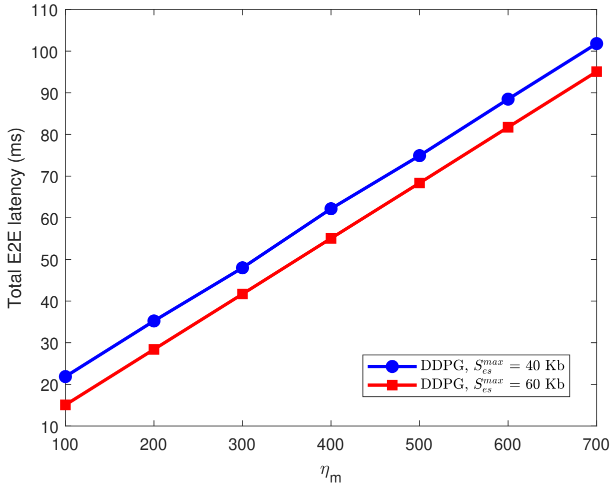

Fig. 8: Impact of task complexity and edge caching capacity

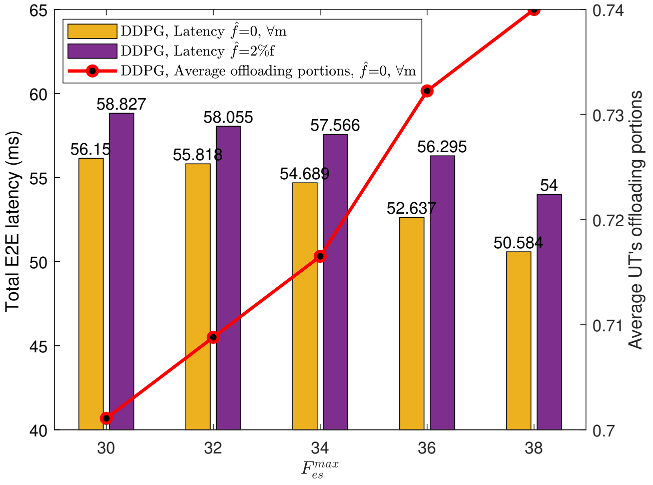

Fig. 9: Impact of average UT offloading and ES processing rate

Fig. 8 reveals the impact of task complexity and edge caching capacity on the system's latency performance. Although there is no implicit relation between task complexity and edge caching capacity, they are interdependent in many situations. A complex task that requires a lot of data may benefit from a larger edge caching capacity to reduce the amount of data that needs to be transferred over the network. Conversely, a smaller edge caching capacity results in higher latency as more data must be retrieved from a distant server. The simulation results indicate that, for any given task complexity, there is a linear increase of 33.3% in the latency when the maximum edge caching capacity (S_{es}^{max}) increases from 40 Kb to 60 Kb. The proposed algorithm's latency increases with an increase in task complexity, whereas it decreases with an increase in the S_{es}^{max}. Conclusively, a network with high edge caching capacity and low task complexity aids in achieving minimum latency, as shown in Fig. 8.

Finally, the joint impact of average UT offloading, ES processing rate, and deviation values significantly on the latency performance of the system are depicted in Fig. 9. Here, "ES processing rate" refers to the ability of the ES to process data and respond to requests. A higher processing rate results in lower latency as the ES can process requests faster. Conversely, a lower processing rate results in higher latency. This results show that as the maximum processing rate of the ES increases, the total end-to-end latency of UT decreases. The total latency decreases by nearly 5.6 ms when the maximum processing rate of the ES (F_{es}^{max}) increases to 38 GHz. A smaller deviation value, i.e., variability in the processing time for a given task, leads to lower latency, and an increase in deviation results in higher latency. Further, an increase in average UT offloading will result in reduced latency performance, and the proposed task offloading model is effective when the percentage of tasks being offloaded by UTs increases [8]–[10]. Overall, the behavior of average UT offloading, ES processing rate, and deviation values are interrelated and affect the latency performance of the system.

VII. Conclusions

This paper studied an unconventional RIS-aided MEC system for URLLC using DT driven framework. Primarily, we focused on the problem of total E2E latency minimization for the task offloading for the considered MEC system subject to the joint design of RIS beamforming, resource (power and bandwidth) allocation, processing rates and task offloading parameters associated with the DT framework. We proposed DT-driven DDPG-based DRL algorithm to solve the formulated problem. We conducted a comparison and analysis of different approaches, including DDPG, PPO, M-PPO, and the conventional AO, to assess their computational complexity and latency performance under various network parameters. The findings indicate that the proposed DDPG-based DRL algorithm outperforms AO in terms of execution time consumption and achieves better minimum latency performance compared to PPO and M-PPO algorithms. Furthermore, M-PPO exhibits lower execution time than DDPG and AO due to their advanced policy optimization techniques. These results highlight the effectiveness of DRL algorithms in optimizing latency in the devised system. Additionally, the simulation results validate the benefits of the proposed RIS beamforming technique, showing a 30-40% performance gain over random-beamforming design. The RIS-assisted MEC system also achieves a 60% lower transmission delay and 20% lower energy consumption compared to the MEC system without RIS. This confirms the practical advantages of leveraging RIS technology in MEC systems.

References

[1] Y. Wang, Z. Li, Y. Hu, and M. Chen, "Joint allocations of radio and computational resource for user energy consumption minimization under latency constraints in multi-cell MEC systems," *IEEE Trans. Veh. Technol.*, pp. 1–16, Oct. 2022.

[2] C.-F. Liu *et al.*, "Dynamic task offloading and resource allocation for ultra-reliable low-latency edge computing," *IEEE Trans. Commun*, vol. 67, no. 6, pp. 4132–4150, June 2019.

[3] Y. Yang, Y. Hu, and M. C. Gursoy, "Energy efficiency analysis in RIS-aided MEC networks with finite blocklength codes," in *Proc. IEEE WCNC*, Apr. 2022, pp. 423–428.

[4] Z. Zhou *et al.*, "Learning-based URLLC-aware task offloading for internet of health things," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 2, pp. 396–410, Feb. 2021.

[5] S. Kurma *et al.*, "Adaptive AF/DF two-way relaying in FD multiuser URLLC system with user mobility," *IEEE Trans. Wireless Commun.*, vol. 21, no. 12, pp. 10 224–10 241, Dec. 2022.

[6] ——, "URLLC-based cooperative industrial IoT networks with nonlinear energy harvesting," *IEEE Trans. Industr. Inform.*, vol. 19, no. 2, pp. 2078–2088, Feb. 2023.

[7] W. Sun, H. Zhang, R. Wang, and Y. Zhang, "Reducing offloading latency for digital twin edge networks in 6G," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12 240–12 251, Oct. 2020.

[8] D. Van Huynh *et al.*, "Edge intelligence-based ultra-reliable and low-latency communications for digital twin-enabled metaverse," *IEEE Wireless Commun. Lett.*, vol. 11, no. 8, pp. 1733–1737, Aug. 2022.

[9] Van Huynh *et al.* , "Digital twin empowered ultra-reliable and low-latency communications-based edge networks in industrial IoT environment," in *Proc. IEEE ICC*, May 2022, pp. 5651–5656.

[10] D. Van Huynh, V.-D. Nguyen, S. R. Khosravirad, V. Sharma, O. A. Dobre, H. Shin, and T. Q. Duong, "URLLC edge networks with joint optimal user association, task offloading and resource allocation: A digital twin approach," *IEEE Trans. Commun.*, vol. 70, no. 11, pp. 7669–7682, Nov. 2022.

[11] T. Liu *et al.*, "Digital-twin-assisted task offloading based on edge collaboration in the digital twin edge network," *IEEE Internet Things J.*, vol. 9, no. 2, pp. 1427–1444, Jan. 2022.

[12] B. Li, Y. Liu, L. Tan, H. Pan, and Y. Zhang, "Digital twin assisted task offloading for aerial edge computing and networks," *IEEE Trans. Veh. Technol.*, vol. 71, no. 10, pp. 10 863–10 877, Oct. 2022.

[13] R. Dong, C. She, W. Hardjawana, Y. Li, and B. Vucetic, "Deep learning for hybrid 5G services in mobile edge computing systems: Learn from a digital twin," *IEEE Trans. Wireless Commun.*, vol. 18, no. 10, pp. 4692–4707, Oct. 2019.

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

[15] A. A. Nasir, H. D. Tuan, H. H. Nguyen, M. Debbah, and H. V. Poor, "Resource allocation and beamforming design in the short blocklength regime for URLLC," *IEEE Trans. Wireless Commun.*, vol. 20, no. 2, pp. 1321–1335, Feb. 2021.

[16] R. Allu, O. Taghizadeh, S. K. Singh, K. Singh, and C.-P. Li, "Robust beamformer design in active RIS-assisted multiuser MIMO cognitive radio networks," *IEEE Trans. Cogn. Commun. Netw.*, pp. 1–1, Jan. 2023.

[17] Hashemi *et al.* , "Deep reinforcement learning for practical phase shift optimization in RIS-assisted networks over short packet communications," in *Proc. EuCNC/6G Summit*, July 2022, pp. 518–523.

[18] M. Xu, S. Zhang, J. Ma, and O. A. Dobre, "Deep learning-based time-varying channel estimation for RIS assisted communication," *IEEE Commun. Lett.*, vol. 26, no. 1, pp. 94–98, Jan. 2022.

[19] F. Tang, X. Chen, M. Zhao, and N. Kato, "The roadmap of communication and networking in 6G for the metaverse," *IEEE Wireless Commun. Lett.*, pp. 1–15, June 2022.

[20] Y. Wang, Z. Su, N. Zhang, R. Xing, D. Liu, T. H. Luan, and X. Shen, "A survey on metaverse: Fundamentals, security, and privacy," *IEEE Commun. Surv, Tutor.*, pp. 1–1, Sep. 2022.

[21] Zhou *et al.* , "Offloading optimization for low-latency secure mobile edge computing systems," *IEEE Wireless Commun. Lett.*, vol. 9, no. 4, pp. 480–484, Apr. 2020.

[22] T. Do-Duy *et al.*, "Digital twin-aided intelligent offloading with edge selection in mobile edge computing," *IEEE Wireless Commun. Lett.*, vol. 11, no. 4, pp. 806–810, Apr. 2022.

[23] Zhou *et al.* , "Secure and latency-aware digital twin assisted resource scheduling for 5G edge computing-empowered distribution grids," *IEEE Trans. Industr. Inform.*, vol. 18, no. 7, pp. 4933–4943, July 2022.

[24] Mithun Mukherjee *et al.* , "The interplay of reconfigurable intelligent surfaces and mobile edge computing in future wireless networks: A win-win strategy to 6G," *arXiv preprint arXiv:2106.11784*, May 2021.

[25] J. Xu, B. Ai, L. Chen, and L. Wu, "Deep reinforcement learning for communication and computing resource allocation in RIS aided MEC networks," in *Proc. IEEE ICC*, Aug. 2022, pp. 3184–3189.

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

[27] T. Bai, C. Pan, C. Han, and L. Hanzo, "Reconfigurable intelligent surface aided mobile edge computing," *IEEE Wireless Commun. Lett.*, vol. 28, no. 6, pp. 80–86, June 2021.

[28] A. Li, Y. Liu, M. Li, Q. Wu, and J. Zhao, "Joint scheduling design in wireless powered MEC IoT networks aided by reconfigurable intelligent surface," in *Proc. IEEE ICCC Workshop*, July 2021, pp. 159–164.

[29] Z. Luo and G. Huang, "Energy-efficient mobile edge computing in RIS-aided OFDM-NOMA relay networks," *IEEE Trans. Veh. Technol.*, pp. 1–16, Nov. Early Access Article, 2022.

[30] Q. Zhang, Y. Wang, H. Li, S. Hou, and Z. Song, "Resource allocation for energy efficient STAR-RIS aided MEC systems," *IEEE Wireless Commun. Lett.*, pp. 1–1, Jan. 2023.

[31] 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, June 2021.

[32] Z. Yang, M. Chen, Z. Zhang, and C. Huang, "Energy efficient semantic communication over wireless networks with rate splitting," *IEEE J. Sel. Areas Commun.*, vol. 41, no. 5, pp. 1484–1495, May 2023.

[33] 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.

[34] J. Xu, B. Ai, L. Chen, and L. Wu, "Deep reinforcement learning for communication and computing resource allocation in RIS aided MEC networks," in *Proc. IEEE ICC*, Aug. 2022, pp. 3184–3189.

[35] L. Wei *et al.*, "Channel estimation for RIS-empowered multi-user MISO wireless communications," *IEEE Trans. Commun*, vol. 69, no. 6, pp. 4144–4157, Mar. 2021.

[36] Wei *et al.* , "Joint channel estimation and signal recovery for RIS-empowered multiuser communications," *IEEE Trans. Commun*, vol. 70, no. 7, pp. 4640–4655, June 2022.

[37] A. Faisal, I. Al-Nahhal, O. A. Dobre, and T. M. N. Ngatched, "Deep reinforcement learning for RIS-assisted FD systems: Single or distributed RIS?" *IEEE Commun. Lett.*, vol. 26, no. 7, pp. 1563–1567, July 2022.

[38] C. She, C. Yang, and T. Q. S. Quek, "Radio resource management for ultra-reliable and low-latency communications," *IEEE Commun. Mag.*, vol. 55, no. 6, pp. 72–78, June 2017.

[39] W. Sun, H. Zhang, R. Wang, and Y. Zhang, "Reducing offloading latency for digital twin edge networks in 6G," *IEEE Trans. Veh. Technol.*, vol. 69, no. 10, pp. 12 240–12 251, Aug. 2020.

[40] Z. Li, X. Wen, Z. Lu, and W. Jing, "A DDPG-based Transfer Learning Optimization Framework for User Association and Power Control in HetNet," in *Proc. IEEE ICC*, July 2022, pp. 343–348.

[41] J. Wang *et al.*, "Computation offloading in multi-access edge computing using a deep sequential model based on reinforcement learning," *IEEE Commun. Mag.*, vol. 57, no. 5, pp. 64–69, May 2019.

[42] S. Li, X. Hu, and Y. Du, "Deep reinforcement learning and game theory for computation offloading in dynamic edge computing markets," *IEEE Access*, vol. 9, pp. 121 456–121 466, Aug. 2021.

[43] M. Grant and S. Boyd, *CVX: Matlab software for disciplined convex programming, version 2.1*, 2014.

[44] L. You *et al.*, "Energy efficiency and spectral efficiency tradeoff in RIS-aided multiuser MIMO uplink transmission," *IEEE Trans. Signal Process.*, vol. 69, pp. 1407–1421, Dec. 2021.

Sravani Kurma (Graduate student member, IEEE) received the B.Tech. degree in Electronics and Communication Engineering from the JNTUH college of Engineering, Jagtial, India, in 2017, and Master's degree (Gold Medalist) in Communication System Engineering from Visvesvaraya National Institute of Technology, Nagpur, India, in 2019. She is currently pursuing Ph.D in Institute of Communications Engineering (ICE) in National Sun Yat-sen University, Taiwan. Her current research interests include 5G, 6G, Industrial internet of things (IIoT), wireless energy harvesting (EH), cooperative communications, Reconfigurable intelligent surfaces (RIS), Full-duplex communication, cell-free MIMO, ultra-reliable and low latency communication (URLLC), resource allocation, and machine learning for communication.

Mayur Katwe (Member, IEEE) received the B.E. degree in electronics and telecommunication from the SGBAU University, Amravati, India, in 2013, the M.Tech degree in Digital System from the Govt. College of Engineering, Pune, India, in 2016, and the Ph.D. degree in Electronics and Communication Engineering from Visvesvaraya National Institute of Technology (VNIT), Nagpur, India, in 2021. Since April 2023, he is working as a Research Scientist in Nanyang Technological University (NTU), Singapore. He held the position of a Postdoctoral Researcher with the Institute of Communications Engineering, National Sun Yat-sen University (NSYSU), Taiwan from 2021 to 2023. His current research interests include radio localization, full duplex radios, non-orthogonal multiple access, rate-splitting multiple access, eMBB-URLLC traffic multiplexing, reconfigurable intelligent surfaces (RIS), integrated sensing and communication, simultaneous transmission and reflecting RIS (STAR-RIS) and unmanned aerial vehicles assisted communication.

Keshav Singh (Member, IEEE) received the M.Sc. degree in Information and Telecommunications Technologies from Athens Information Technology, Greece, in 2009, and the Ph.D. degree in Communication Engineering from National Central University, Taiwan, in 2015. He currently works at the Institute of Communications Engineering, National Sun Yat-sen University (NSYSU), Taiwan as an Assistant Professor. Prior to this, he held the position of Research Associate from 2016 to 2019 at the Institute of Digital Communications, University of Edinburgh, U.K. From 2019 to 2020, he was associated with the University College Dublin, Ireland as a Research Fellow. He leads research in the areas of green communications, resource allocation, full-duplex radio, ultra-reliable low-latency communication, non-orthogonal multiple access, wireless edge caching, machine learning for communications, and large intelligent surface-assisted communications.

Cunhua Pan received the B.S. and Ph.D. degrees from the School of Information Science and Engineering, Southeast University, Nanjing, China, in 2010 and 2015, respectively. From 2015 to 2016, he was a Research Associate at the University of Kent, U.K. He held a post-doctoral position at Queen Mary University of London, U.K., from 2016 and 2019.From 2019 to 2021, he was a Lecturer in the same university. From 2021, he is a full professor in Southeast University.

His research interests mainly include reconfigurable intelligent surfaces (RIS), intelligent reflection surface (IRS), ultra-reliable low latency communication (URLLC) , machine learning, UAV, Internet of Things, and mobile edge computing. He has published over 120 IEEE journal papers. He is currently an Editor of IEEE Transactions on Vehicular Technology, IEEE Wireless Communication Letters, IEEE Communications Letters and IEEE ACCESS. He serves as the guest editor for IEEE Journal on Selected Areas in Communications on the special issue on xURLLC in 6G: Next Generation Ultra-Reliable and Low-Latency Communications. He also serves as a leading guest editor of IEEE Journal of Selected Topics in Signal Processing (JSTSP) Special Issue on Advanced Signal Processing for Reconfigurable Intelligent Surface-aided 6G Networks, leading guest editor of IEEE Vehicular Technology Magazine on the special issue on Backscatter and Reconfigurable Intelligent Surface Empowered Wireless Communications in 6G, leading guest editor of IEEE Open Journal of Vehicular Technology on the special issue of Reconfigurable Intelligent Surface Empowered Wireless Communications in 6G and Beyond, and leading guest editor of IEEE ACCESS Special Issue on Reconfigurable Intelligent Surface Aided Communications for 6G and Beyond. He is Workshop organizer in IEEE ICCC 2021 on the topic of Reconfigurable Intelligent Surfaces for Next Generation Wireless Communications (RIS for 6G Networks), and workshop organizer in IEEE Globecom 2021 on the topic of Reconfigurable Intelligent Surfaces for future wireless communications. He is currently the Workshops and Symposia officer for Reconfigurable Intelligent Surfaces Emerging Technology Initiative. He is workshop chair for IEEE WCNC 2024, and TPC co-chair for IEEE ICCT 2022. He serves as a TPC member for numerous conferences, such as ICC and GLOBECOM, and the Student Travel Grant Chair for ICC 2019. He received the IEEE ComSoc Leonard G. Abraham Prize in 2022, IEEE ComSoc Asia-Pacific Outstanding Young Researcher Award, 2022.

Shahid Mumtaz (**Senior Member, IEEE**) is a Nottingham Trent University (NTU), UK professor. He is an IET Fellow, founder, and EiC of IET ”Journal of Quantum Communication,” Vice-Chair: Europe/Africa Region- IEEE ComSoc: Green Communications & Computing Society. He authorizes four technical books, 12 book chapters, and 300+ technical papers (200+ IEEE Journals/transactions, 100+ conferences, 2 IEEE best paper awards) in mobile communications. Most of his publication is in the field of Wireless Communication. He is a Scientific Expert and Evaluator for various research funding agencies. In 2012, he was awarded an ”Alain Bensoussan fellowship.” China awarded him the young scientist fellowship in 2017.

Chih-Peng Li (Fellow, IEEE) received the B.S. degree in Physics from National Tsing Hua University, Hsin Chu, Taiwan, and the Ph.D. degree in Electrical Engineering from Cornell University, NY, USA. Dr. Li was a Member of Technical Staff with Lucent Technologies. Since 2002, he has been with National Sun Yat-sen University (NSYSU), Kaohsiung, Taiwan, where he is currently a Distinguished Professor. Dr. Li has served various positions with NSYSU, including the Chairman of Electrical Engineering Department, the VP of General Affairs, the Dean of Engineering College, and the VP of Academic Affairs. His research interests include wireless communications, baseband signal processing, and data networks. He is now the Director General with the Engineering and Technologies Department, National Science and Technology Council, Taiwan.

Dr. Li is currently the Chapter Chair of IEEE Broadcasting Technology Society Tainan Section. Dr. Li has also served as the Chapter Chair of IEEE Communication Society Tainan Section, the President of Taiwan Institute of Electrical and Electronics Engineering, the Editor of IEEE Transactions on Wireless Communications, the Associate Editor of IEEE Transactions on Broadcasting, and the Member of Board of Governors with IEEE Tainan Section. Dr. Li has received various awards, including the Outstanding Research Award of Ministry of Science and Technology. Dr. Li is a Fellow of the IEEE.