

Article

Sum Rate Optimization for Multi-IRS-Aided Multi-BS Communication System Based on Multi-Agent

Maha Fathy , Zesong Fei , Jing Guo *  and Mohamed Salah Abood 

School of Information and Electronics Engineering, Beijing Institute of Technology, Beijing 100081, China; maha.fathy@bit.edu.cn (M.F.); feizesong@bit.edu.cn (Z.F.); mohammedsalah@bit.edu.cn (M.S.A.)

* Correspondence: jingguo@bit.edu.cn

Abstract: Intelligent reflecting surface (IRS) is a revolutionizing technology for improving the spectral and energy efficiency of future wireless networks. In this paper, we consider a downlink large-scale system empowered by multi-IRS to aid communication between the multiple base stations (BSs) and multiple user equipment (UEs). We target maximizing the sum rate by jointly optimizing the UE association, the transmit powers of BSs, and the configurations of the IRS beamforming. Due to the applicability restrictions of conventional optimization methods and their high complexity with large-scale networks in dynamic environments, deep reinforcement (DRL) learning is adopted as an alternative approach to finding optimal solutions. First, we model the optimization problem as a multi-agent Markov decision problem (MAMDP). Then, because large-scale wireless networks are naturally complex and changeable, and because many entities interact and affect how the whole system works, it is important to use a multi-agent approach to understand the complex dependencies and relationships between the different parts. In order to solve the problem, we propose a cooperative multi-agent deep reinforcement learning (MADRL)-based algorithm that works for both continuous and discrete IRS phase shifts. Simulation results validate that the proposed algorithm surpasses iterative optimization benchmarks regarding both sum rate performance and convergence.

Keywords: intelligent reflecting surface; passive beamforming; power control; user association; multi-agent



Citation: Fathy, M.; Fei, Z.; Guo, J.; Abood, M.S. Sum Rate Optimization for Multi-IRS-Aided Multi-BS Communication System Based on Multi-Agent. *Electronics* **2024**, *13*, 735. <https://doi.org/10.3390/electronics13040735>

Academic Editor: Fernando De la Prieta Pintado

Received: 29 December 2023

Revised: 6 February 2024

Accepted: 8 February 2024

Published: 11 February 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

In light of the escalating need for enhanced data rates, improved energy and spectrum efficiency, and pervasive network access in the upcoming network generations [1], researchers have suggested numerous enhanced solutions encompassing various technologies. Among these, the intelligent reflecting surface (IRS) has emerged as a highly appealing and attractive technology that reconfigures the wireless propagation environment via software-controlled reflections [2]. In particular, an IRS is made up of many passive reflecting elements, and each one can change the amplitude and phase angles of the signals it receives on its own [3]. The IRS can create an extra communication path between the BSs and the UEs, which keeps communication from getting stuck and improves signal reception at the intended UEs and attenuating it at the non-intended UEs [4]. Moreover, due to its inherent properties and notable benefits in the aspects of spectral and energy efficacy, along with its economically viable deployment, the IRS is anticipated to surpass alternative technologies like backscatters and relays [5], making it a superior technology.

Numerous recent works have employed traditional optimization techniques for the joint optimization design within wireless networks, aided by the IRS. Specifically, in the study [4], the semidefinite (SDR)-based relaxation and alternating techniques were utilized to address the BS and IRS beamforming. Expanding on this joint design concept, the authors in [6] introduced a difference-of-convex approach within an IRS-aided system, overcoming the SDR limitations and efficiently handling the problem. Similarly, the

work [7] concurrently configured the power allocation and IRS phases, employing gradient descent, fractional programming, and alternating maximization-based methods. The study [8] delved into optimizing the system's weighted sum rate, employing an alternating approach-based Lagrangian transform. Additionally, certain works explored systems assisted by multi-IRS. In the study [9], Riemannian manifold and Lagrangian techniques were employed to configure the IRS phase shifts and transmit powers. The above works focused on a single BS scenario and ignored the interference from other BSs.

For the joint optimization problem in the multi-BS scenario, the UE association is important; hence, some works investigated joint optimization by including association design. Specifically, the researchers in [10] jointly designed passive beamforming and UE association in IRS-aided heterogeneous networks, where fixed point iteration and priority-based swapping algorithms were proposed. In [11], the researchers analyzed the average SINR for multiple-UE in a distributed IRS-assisted system. They used the successive refinement algorithm to find the best IRS–UE association coefficients and the best IRS configuration to serve the UEs that were connected to them. The proposed strategy maintained the association between the IRS and the UE for long periods, which minimized performance degradation and channel estimation overhead. In [12], an alternating-based approach was employed to optimize both the UE association and the passive beamforming of the assisting IRS for rate maximization. The work in [13] addressed the joint association problem in a multi-IRS-aided downlink system, where the association design between the BSs, UEs, and IRSs was designed to maximize the UEs' utility via both optimal and sub-optimal presented solutions. In [14], the authors addressed the IRS–UE association problem to find the best balance of passive beamforming gains between various BS–UE links. Furthermore, the authors derived an average SINR (ASAINR) expression for each UE to formulate the ASAINR balancing problem, solving the IRS–UE association and power control. Here, all involved channels were assumed to be Rayleigh-fading in the analysis. The work [15] investigated an IRS-assisted multi-BS multi-UE millimeter wave downlink communication system. The authors formulated a strategy to maximize the sum rate by concurrently optimizing association design, power allocation, and IRS passive beamforming. They introduced an iterative algorithm grounded in alternating optimization, sequential fractional programming, and forward-reverse auction mechanisms to tackle this formulated problem. The study conducted in [16] examined a mmWave cellular network assisted by an IRS. In this configuration, an IRS was deployed to improve the signals to the UEs within each cell. Moreover, the UE association was optimized by a matching game approach to balance the BS loads and increase the network utility.

Although the aforementioned conventional optimization-based methods in various approaches and algorithms have shown satisfactory performance, the system complexity often hinders their practical implementation [7]. Consequently, researchers have turned their focus towards machine learning (ML) techniques, which have proven to be effective at overcoming the challenges faced by traditional methods and reducing optimization complexity in implementing IRS-aided systems [2,17]. Among the ML approaches, deep reinforcement learning (DRL) is commonly applied to solve complex dynamic problems in IRS-aided systems due to its exceptional ability to learn the hidden patterns in the data. Within the works of optimizing IRS reflection phases, a study by the authors [18] introduced a DRL-based algorithm. This algorithm aimed to predict IRS reflection phases while minimizing the training overhead, in contrast to the supervised-based approach presented in [19]. In the study [20], an approach based on DRL was introduced for IRS reflection. This approach aimed for throughput maximization in an imperfect channel state information (CSI) scenario. In the context of the joint beamforming design, the study presented in [21] developed a DRL model. The overarching objectives were to enhance the system's sum rate while maintaining a low level of complexity. The study outlined in [22] revolved around a secure wireless communication system incorporating an IRS. Enhancement of the system's rate was achieved via the setup of both BS beamforming and IRS reflecting beamforming over various qualities of service (QoS). The work in [23]

formulated the energy efficiency maximization problem constrained to UEs' data rates by jointly setting up the transmit powers, IRS configuration design, and deployment. To solve this joint problem, the authors adopted a long, short-term memory-based echo state network algorithm and a double-deep Q-network. Additionally, the authors of [24,25] optimized the designed ML-based algorithms for multi-IRS-aided communication systems. Specifically, the proposed GRNN-based and DDPG-based algorithms were presented for static and dynamic UE scenarios.

The above-mentioned works predominantly center around the single-agent paradigm, emphasizing the optimization and deployment of IRS-aided systems through individual agent strategies. However, when it comes to addressing challenges in IRS-aided systems, the single-agent paradigm encounters limitations in effectively navigating the complexities, grappling with intricate interdependencies, and adapting to the dynamic nature of wireless communication networks; hence, researchers are increasingly turning to the multi-agent framework. The shift to multi-agent systems becomes crucial in IRS-rich environments, where the collaborative coordination of multi-IRS holds the potential for significant improvements. Given the dynamic nature of wireless communication networks and the intricate interdependencies between IRSs and BSs, adopting multi-agent approaches offers distinct advantages. The deployment of a multi-agent approach facilitates collaborative decision-making, enabling a more holistic exploration of the solution space for refining the optimization of efficient IRS-aided communication systems. This, in turn, prompts some researchers to leverage the multi-agent paradigm [26–30]. Specifically, in [30], researchers presented a multi-agent-based deep Q-network approach that optimized the powers of the BSs, considering discrete power levels. There is a scarcity of literature dedicated to multi-agent algorithms in IRS-assisted systems [31,32]. In [31], the considered problem of power allocation and subcarrier assignment was formulated and solved by a multi-agent-based approach, while in [32], a multi-agent reinforcement learning-based buffer was proposed for aiding relay selection in an IRS-aided system. Precisely, the agents were trained to optimize the IRS reflection phases and relay selection to achieve maximum throughput.

While the existing literature has made significant strides in enhancing communication systems through the integration of IRS with ML, it is crucial to address certain limitations in the current research landscape. Many researchers have focused on optimizing IRS-aided communication in small-scale network scenarios. However, the practical deployment of communication systems often involves large-scale networks with multiple interacting elements. Existing studies have commonly built and trained their ML-based models based on fixed associations designed between systems nodes or discretizing their variables to defined levels. This approach might be inappropriate or not relevant for the deployed systems. The need for better spectral efficiency in communication networks is growing all the time. This makes studying the sum rate maximization problem even more important, since it directly relates to the main goal of improving system performance in large networks with multiple IRSs. Thus, in our work, we study the sum rate maximization problem for a large-scale system with multiple IRSs, where the phase of the IRS can be either discrete or continuous. A MADRL-based algorithm is proposed to solve the joint optimization of the transmit powers of the BSs, the reflection beamforming of the IRSs, and the design of BS–IRS–UE associations. The main contribution of this paper can be outlined as follows:

- We develop a large-scale communication system with multi-BS assisted by multi-IRS to serve multi-UE. We formulate a comprehensive optimization problem aimed at maximizing the data rates of served UEs. The joint optimization encompasses BSs' transmit powers, IRSs' passive beamforming, and the design of BS–UE–IRS associations, all subject to the system and UEs' QoS requirements.
- Addressing the inherent non-convexity of the original optimization problem, we propose a MADRL-based algorithm under continuous and discrete phases of the IRS scenarios. The optimization problem is modeled as an MAMDP, where the agents are designed by integrating DDPG and DQN algorithms to learn a joint policy.

This approach enables the system to dynamically adapt to changing environmental conditions and maximize the long-term total system utility.

- We analyze the convergence, computational complexity, and implementation efficiency of the proposed MADRL-based algorithm, integrated with the DRL algorithms. Our numerical results not only showcase the superior performance of our algorithm over benchmark schemes (e.g., the successive refinement-based approach), but also demonstrate its efficiency in terms of implementation runtime, thus validating its practical feasibility. Moreover, the achieved rate in the IRS-aided system exhibits substantial improvements as the number of reflecting elements increases, highlighting the scalability and effectiveness of the proposed approach.

The rest of this paper is structured as follows: Section 2 outlines the system model. Section 3 presents the formulated optimization problem. Section 4 details the proposed MADRL-based algorithm for continuous and discrete phases of the IRS configurations. Following that, Section 5 presents the simulation results. Finally, Section 6 provides a summary of the work.

Notation: Italicized letters represent scalars, while boldface is used for vectors and matrices. The $(\cdot)^T$ and $(\cdot)^H$ superscripts indicate transpose and Hermitian operations, respectively, while $\mathbb{C}$ signifies the complex domain. The notation $\mathcal{CN}(\mu, \sigma^2)$ denotes a circularly symmetric complex Gaussian (CSCG) distribution with mean μ and variance σ^2 . The notations $\|\cdot\|$, $|\cdot|$, $\lfloor \cdot \rfloor$, and $\mathbb{E}[\cdot]$ represent the Euclidean norm, absolute value, floor, and expectation, respectively. Moreover, $\text{Mod}(\cdot)$ represents the modulo operation; $\text{diag}(x)$ denotes a diagonal matrix constructed with the elements of the vector x as its diagonal entries; $\log(\cdot)$ denotes the logarithm function; ∇f represents the gradient of f ; and the order of complexity is denoted by $\mathcal{O}(\cdot)$.

2. System Model

As shown in Figure 1, we consider a downlink large-scale system assisted by multi-IRS to aid the communication between the multiple BSs and multiple UEs. Let the sets of multi-BS, multi-UE, and multi-IRS be denoted as $\mathcal{S} = \{1, \dots, S\}$, $\mathcal{K} = \{1, \dots, K\}$, and $\mathcal{L} = \{1, \dots, L\}$, respectively. At each time instance, the set of UEs can be divided into K_s serving UEs and K_w waiting UEs, with sets $\mathcal{K}_s = \{1, \dots, K_s\}$, and $\mathcal{K}_w = \{1, \dots, K_w\}$, respectively, where $\mathcal{K}_s \cup \mathcal{K}_w = \mathcal{K}$. Each BS is equipped with M transmit antennas, and each UE is equipped with a single antenna. Each IRS is a 2D surface composed of $N = N_H N_V$ reflecting elements arranged in a rectangular grid. Here, N_H represents the number of elements per row, and N_V represents the number of elements per column. In addition, a wireless controller connects to the BS to configure the elements of the IRS [33]. Let the binary variables $\rho_{k,s}$, $k \in \mathcal{K}$, $s \in \mathcal{S}$, represent the association indicator between the k -th UE and the s -th BS, where $\rho_{k,s} = 1$ indicates that the k -th UE is associated with the s -th BS; otherwise, $\rho_{k,s} = 0$. It is assumed that each UE can be served from a maximum of one BS and, in turn, each BS is restricted to serving only one UE at any given time instant. The binary variables, $\lambda_{l,s}$, $l \in \mathcal{L}$, $s \in \mathcal{S}$, denote the association indicator between the l -th IRS and the s -th BS, where $\lambda_{l,s} = 1$ indicates that the l -th IRS is associated with the s -th BS; otherwise, $\lambda_{l,s} = 0$. Furthermore, it is assumed that each IRS can be associated with only one BS at a given time instant. Accordingly, we can define the constraints of BS–UE–IRS associations at each node as follows:

$$\begin{aligned}
 \sum_{s \in \mathcal{S}} \rho_{k,s} &\leq 1, \quad \forall k \in \mathcal{K} \quad (\text{UE association}), \\
 \sum_{k \in \mathcal{K}} \rho_{k,s} &\leq 1, \quad \forall s \in \mathcal{S} \quad (\text{BS association}), \\
 \sum_{s \in \mathcal{S}} \lambda_{l,s} &= 1, \quad \forall l \in \mathcal{L} \quad (\text{IRS association}).
 \end{aligned} \tag{1}$$

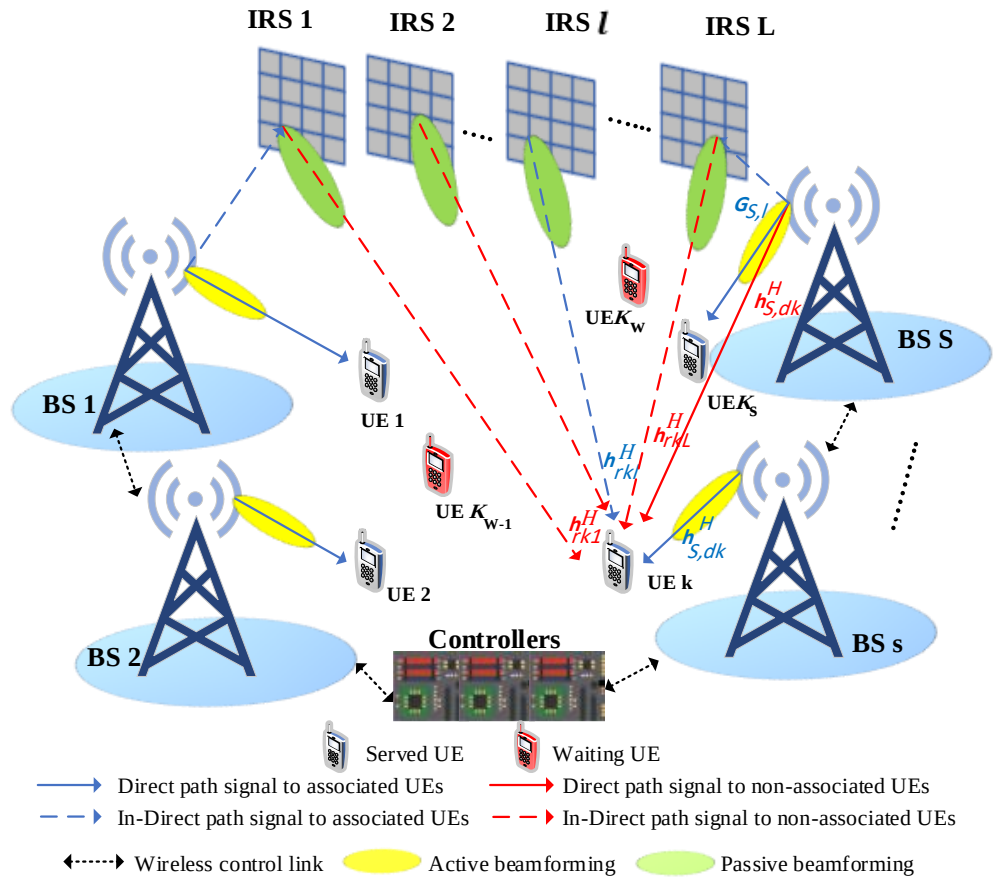


Figure 1. Architecture of multi-IRS-aided multi-BS communication system.

We assume quasi-static, flat-fading channels. Specifically, the direct channel is denoted as $\mathbf{h}_{s,dk}^H \in \mathbb{C}^{1 \times M}$, which represents the channel from the s -th BS to the k -th UE, while the indirect channels, $\mathbf{G}_{s,l} \in \mathbb{C}^{N \times M}$ and $\mathbf{h}_{rkl}^H \in \mathbb{C}^{1 \times N}$, represent the channels from the s -th BS to the l -th IRS and from the l -th IRS to the k -th UE, respectively. As for the channel models, the BS–UE channels are modeled as Rayleigh fading, $\mathbf{h}_{s,dk}^H = \text{PL}_{s,k} \tilde{\mathbf{h}}_{s,dk}^H$, where $\text{PL}_{s,k}$ is the s -th BS– k -th UE channel path loss, and $\tilde{\mathbf{h}}_{s,dk}^H$ follows $\mathcal{CN}(0, 1)$. The IRSs are positioned to maintain line-of-sight (LOS) connections between the BSs and the UEs. Consequently, the channels for BS–IRS and IRS–UE follow Rician fading, $\mathbf{G}_{s,l} = \text{PL}_{s,l} (\sqrt{\frac{\epsilon}{1+\epsilon}} \tilde{\mathbf{G}}_{s,l}^{(\text{LOS})} + \sqrt{\frac{1}{1+\epsilon}} \tilde{\mathbf{G}}_{s,l}^{(\text{NLOS})})$ and $\mathbf{h}_{rkl}^H = \text{PL}_{k,l} (\sqrt{\frac{\epsilon}{1+\epsilon}} \tilde{\mathbf{h}}_{rkl}^{H(\text{LOS})} + \sqrt{\frac{1}{1+\epsilon}} \tilde{\mathbf{h}}_{rkl}^{H(\text{NLOS})})$, respectively. Here, $\text{PL}_{s,l}$ and $\text{PL}_{k,l}$ are the path losses of the s -th BS to the l -th IRS channel and the l -th IRS to the k -th UE channel, respectively, and ϵ is the Rician factor. The NLOS components, $\tilde{\mathbf{G}}_{s,l}^{(\text{NLOS})}$ and $\tilde{\mathbf{h}}_{rkl}^{H(\text{NLOS})}$, exhibit a distribution following $\mathcal{CN}(0, 1)$. The LOS elements are formulated as per [34], that is, $\tilde{\mathbf{G}}_{s,l}^{(\text{LOS})} = \mathbf{a}_{\text{IRS}}(\vartheta_{s,l}, \psi_{s,l}) \mathbf{a}_{\text{BS}}(\vartheta_s, \psi_s)^H$ and $\tilde{\mathbf{h}}_{rkl}^{H(\text{LOS})} = \mathbf{a}_{\text{IRS}}(\vartheta_{l,k}, \psi_{l,k})^H$. Here, $\mathbf{a}_{\text{IRS}}$ and $\mathbf{a}_{\text{BS}}$ are the steering vectors for IRS and BS, respectively. The symbols ϑ and ψ represent the azimuth and elevation angles. For the l -th IRS steering vector, the n -th element can be calculated as $[\mathbf{a}_{\text{IRS}}(\vartheta_{l,k}, \psi_{l,k})]_n = e^{j \frac{2\pi d^l}{\lambda_c} \{i_1(n) \sin(\vartheta_{l,k}) \cos(\psi_{l,k}) + i_2(n) \sin(\psi_{l,k})\}}$, where λ_c denotes the carrier wavelength, d^l represents the distance between adjacent elements of the l -th IRS, $i_1(n) = \text{mod}(n-1, N_H)$, and $i_2(n) = \lfloor (n-1)/N_H \rfloor$. Let the BSs, IRSs, and UEs coordinates be denoted as (x_s, y_s, z_s) , (x_l, y_l, z_l) , and (x_k, y_k, z_k) . Accordingly, we set $\sin(\psi_{l,k}) = (z_k - z_l)/d_k^l$ and $\sin(\vartheta_{l,k}) \cos(\psi_{l,k}) = (y_k - y_l)/d_k^l$. Here, d_k^l represents the distance between the k -th UE and the l -th IRS. Also, $\mathbf{a}_{\text{BS}}(\vartheta_s, \psi_s) = [1, \dots, e^{j \frac{2\pi(M-1)d^s}{\lambda_c} \cos(\vartheta_1) \cos(\psi_1)}]$, where d^s is the distance between adjacent antennas at the s -th BS. We set

$\cos(\vartheta_s) \cos(\psi_s) = (x_l - x_s)/d_s^l$, where d_s^l represents the separating distance between the s -th BS and the l -th IRS. Likewise, the steering vector $\mathbf{a}_{\text{IRS}}(\vartheta_{s,l}, \psi_{s,l})$ can be computed, with $\sin(\vartheta_{s,l}) \cos(\psi_{s,l}) = (y_s - y_l)/d_s^l$, and $\sin(\psi_{s,l}) = (z_s - z_l)/d_s^l$. Note that reciprocity between downlink and uplink channels is assumed.

At the transmitting side, each BS transmits its signal with a beamforming vector to serve its associated UE. Hence, the transmitted signal by the s -th BS takes the form of:

$$x_s = \sqrt{p_s} \mathbf{w}_s s_s, \quad (2)$$

where p_s is the transmit power, $\mathbf{w}_s \in \mathbb{C}^{M \times 1}$ denotes the beamforming vector, and the transmit signal, denoted as s_s , is characterized by random variables with a mean of zero and a variance of $\mathbb{E}[|s_s|^2] = 1$, which holds for all UEs. We assume each BS applies the maximum ratio transmission (MRT) precoding based on both direct and indirect channels with its served UE for active beamforming design, i.e.,

$$\mathbf{w}_s = \frac{\mathbf{H}_k^H}{\|\mathbf{H}_k^H\|}, \quad (3)$$

where $\mathbf{H}_k = \mathbf{h}_{s,dk}^H + \sum_{l \in \mathcal{L}} \mathbf{h}_{rkl}^H \mathbf{\Theta}_l \mathbf{G}_{s,l}$ indicates the overall channel from the s -th BS to its served k -th UE; hence, $\mathbb{E}[|x_s|^2] = p_s$, i.e., $p_s \geq 0$.

At the reflecting side, the l -th IRS adjusts the received signals from all BSs and subsequently reflects them. In particular, each n -th element of the l -th IRS reflects the incident signal, $\hat{x}_{s,n}$, multiplied by a complex factor of reflection. Therefore, the mathematically defined expression for the adjusted reflected signal at the l -th IRS by the n -th element, resulting from the incident signal from the s -th BS, is as follows:

$$\hat{y}_{s,l,n} = \beta_{l,n} e^{j\theta_{l,n}} \hat{x}_{s,n}, \forall s, l, n, \quad (4)$$

where $\beta_{l,n} \in [0, 1]$ and $\theta_{l,n} \in F_l = [0, 2\pi)$ represent the amplitude and angle for the n -th element of the l -th IRS, respectively, and F_l denotes the set of phase-shift values at each element. Note that in practical implementation, it is considered that each n -th element n -th of the l -th IRS is limited to a finite set of discrete values. Therefore, for discrete phase shifts in the l -th IRS, the angle of the n -th element is determined as $\theta_{l,n} \in F_l = \left\{0, \frac{2\pi}{2^b}, \dots, \frac{2\pi(2^b-1)}{2^b}\right\}$, where b denotes the number of bits used to determine the number of phase shift levels at each element. As such, the l -th IRS maps from the s -th BS incident signal vector to a corresponding reflected signal vector using the reflection beamforming matrix $\mathbf{\Theta}_l$, which is the main characteristic for each IRS. Therefore, the reflected signal at the l -th IRS due to the s -th BS incident signal is:

$$\hat{\mathbf{y}}_{s,l} = \mathbf{\Theta}_l \hat{\mathbf{x}}_s, \forall s, l, \quad (5)$$

where $\hat{\mathbf{x}}_s = [\hat{x}_{s,1}, \dots, \hat{x}_{s,N}]^T$, and $\mathbf{\Theta}_l = \text{diag}(\beta_{l,1} e^{j\theta_{l,1}}, \dots, \beta_{l,N} e^{j\theta_{l,N}}) \in \mathbb{C}^{N \times N}$ and $\hat{\mathbf{y}}_{s,l} = [\hat{y}_{s,l,1}, \dots, \hat{y}_{s,l,N}]^T$. We further set $\beta_{l,n} = 1, \forall l, n$ in this system for maximizing the reflected signal by the IRSs [2,4].

At the receiving side, the k -th served UE $\in \mathcal{K}_s$ receives its desired signal both from its associated BS via the direct path as well as the indirect paths from the reflections and the scattering of the associated and non-associated IRSs, respectively. Considering the substantial path loss, only the first signal reflected by the IRSs is taken into account, while the others are ignored. Hence, the desired signal received at the k -th UE from the s -th BS is calculated as follows:

$$\begin{aligned}
y_{k,s} &= \mathbf{h}_{s,dk}^H \sqrt{p_s} \mathbf{w}_s s_s + \left(\sum_{l \in \mathcal{L}} \mathbf{h}_{rkl}^H \boldsymbol{\Theta}_l \mathbf{G}_{s,l} \right) \sqrt{p_s} \mathbf{w}_s s_s \\
&= \mathbf{h}_{s,dk}^H \sqrt{p_s} \mathbf{w}_s s_s + \underbrace{\left(\sum_{l \in \mathcal{L}} \lambda_{l,s} \mathbf{h}_{rkl}^H \boldsymbol{\Theta}_l \mathbf{G}_{s,l} \right)}_{\text{Associated IRSs}} \sqrt{p_s} \mathbf{w}_s s_s \\
&\quad + \underbrace{\left(\sum_{l \in \mathcal{L}} (1 - \lambda_{l,s}) \mathbf{h}_{rkl}^H \boldsymbol{\Theta}_l \mathbf{G}_{s,l} \right)}_{\text{Non-associated IRSs}} \sqrt{p_s} \mathbf{w}_s s_s, \quad \forall k \in \mathcal{K}_s.
\end{aligned} \tag{6}$$

Meanwhile, the k -th UE is affected by co-channel interference from the $S - 1$ non-associated BSs that serve the other $K - 1$ UEs through direct and indirect links via all deployed IRSs. Thus, the total interference signal affecting the k -th UE is expressed as:

$$\begin{aligned}
\ddot{I}_k &= \underbrace{\sum_{i \in \mathcal{S}, i \neq s} \mathbf{h}_{i,dk}^H \sqrt{p_i} \mathbf{w}_i s_i + \sum_{i \in \mathcal{S}, i \neq s} \sum_{l \in \mathcal{L}} \mathbf{h}_{rkl}^H \boldsymbol{\Theta}_l \mathbf{G}_{i,l} \sqrt{p_i} \mathbf{w}_i s_i}_{\text{Inter-cell interference (co-channel interference)} I_k} + \underbrace{n_k}_{\text{Noise}}, \quad \forall k \in \mathcal{K}_s,
\end{aligned} \tag{7}$$

where n_k is modeled as $\mathcal{CN}(0, \sigma_k^2)$, $\forall k \in \mathcal{K}$. Therefore, the signal-to-interference-plus-noise ratio (SINR) observed at the k -th UE, associated with the s -th BS, is formulated as:

$$\text{SINR}_{k,s} = \frac{|y_{k,s}|^2}{|I_k|^2 + \sigma_k^2}, \quad \forall k \in \mathcal{K}_s. \tag{8}$$

Then, the achievable rate of the k -th UE, expressed in bps/Hz, is determined as:

$$R_{k,s} = \sum_{s \in \mathcal{S}} \rho_{k,s} \log_2(1 + \text{SINR}_{k,s}), \quad \forall k \in \mathcal{K}_s. \tag{9}$$

Accordingly, we define the system utility as the sum rate of all served UEs by:

$$R_{\text{sum}} = \sum_{k \in \mathcal{K}_s} R_{k,s} = \sum_{k \in \mathcal{K}} \sum_{s \in \mathcal{S}} \rho_{k,s} \log_2(1 + \text{SINR}_{k,s}). \tag{10}$$

3. Formulation of the Optimization Problem

The target of this work is the maximization of the system utility by maximizing the sum rate of the served UEs while still guaranteeing their rate constraints by jointly optimizing BS-UE associations $\{\rho_{k,s}\}$, BS-IRS associations $\{\lambda_{l,s}\}$, IRSs beamforming matrices $\{\boldsymbol{\Theta}_l\}_{l=1}^L$, and BSs power control $\mathbf{P}$. Therefore, we formulate the optimization problem as:

$$(\text{P1}) : \max_{\{\rho_{k,s}\}, \{\lambda_{l,s}\}, \{\boldsymbol{\Theta}_l\}_{l=1}^L, \mathbf{P}} R_{\text{sum}} \tag{11a}$$

$$\text{s.t. } R_{k,s} \geq R_{\min}, \forall k \in \mathcal{K}_s, \forall s \in \mathcal{S}, \tag{11b}$$

$$\sum_{s \in \mathcal{S}} \rho_{k,s} \leq 1, \forall k \in \mathcal{K}, \tag{11c}$$

$$\sum_{k \in \mathcal{K}} \rho_{k,s} \leq 1, \forall s \in \mathcal{S}, \tag{11d}$$

$$\sum_{s \in \mathcal{S}} \lambda_{l,s} = 1, \forall l \in \mathcal{L}, \tag{11e}$$

$$\theta_{l,n} \in F_l, \forall l, n, \tag{11f}$$

$$0 \leq p_s \leq p_{\max}, \forall s \in \mathcal{S}, \tag{11g}$$

$$\rho_{k,s} \in \{0, 1\}, \lambda_{l,s} \in \{0, 1\}, \forall l \in \mathcal{L}, \forall s \in \mathcal{S}, \forall k \in \mathcal{K}. \tag{11h}$$

Note that (P1) is a challenging mixed-integer nonlinear programming (MINLP) problem due to its non-convex nature and the coupling between the objective function and constraints. The inclusion of binary variables $\rho_{k,s}$ and $\lambda_{l,s}$ introduces combinatorial aspects and NP-hardness. Generally, to optimally solve this problem, a relaxation conversion can be applied to transform the original non-convex problem into a convex formulation. To be specific, applying convex relaxation to the objective involves using the concave envelope, e.g., the exponential function, to approximate the logarithmic function. For the constraints, the problem has two types of non-convex constraints: the discrete constraints of passive beamforming and the binary constraints of association matrices. The discrete constraints can be relaxed by applying the semidefinite programming (SDP) technique by allowing continuous phase shifts. Meanwhile, the binary constraints can be replaced with continuous variables constrained between 0 and 1. Then, use a penalty function to enforce the binary nature. A convex optimization solver capable of handling mixed-integer and continuous variables, such as CVX [35], can solve the resulting convex-relaxed optimization problem. Based on our large-scale system, solving large-scale mixed-integer convex problems can be computationally demanding, especially with SDP relaxations, which can lead to computationally expensive solutions. However, with the discrete phase shift configuration and fixing BSs' power, or when relaxed to discrete counterparts, the optimization can be solved optimally by enumerating all feasible possible combinations. Nevertheless, this approach incurs high complexity, which becomes prohibitive with a large number of BSs, UEs, and IRSs. Note that in case of the continuous configurations to the IRSs and considering constraint (11d), the phases can be determined by the instantaneous CSI, as the direct and indirect channels are aligned in phase to serve the associated UE [4]. Thus, the n -th phase shift of the l -th IRS is calculated as:

$$\theta_{l,n} = -\angle h_{rkl,n} - \angle (\mathbf{g}_{s,l,n}^H \mathbf{h}_{s,dk}), \quad (12)$$

where $h_{rkl,n}$ is the n -th element of $\mathbf{h}_{rkl}^H$ and $\mathbf{g}_{s,l,n}^H$ is the n -th row vector in $\mathbf{G}_{s,l}$. Additionally, some strategies and algorithms have been applied to find sub-optimal solutions to association matrices with given other optimization variables. Specifically, the maximum reference signal received power (RSRP) association-based method is commonly used for designing BS–UE associations [36]. Let $\boldsymbol{\rho}_s = [\rho_{1,s}, \rho_{2,s}, \dots, \rho_{k,s}]$ represent the solution for BS–UE associations and the received powers from the s -th BS to UEs are denoted as $\mathbf{P}_s^r = [P_{1,s}^r, P_{1,s}^r, \dots, P_{k,s}^r]$, where $P_{k,s}^r = |y_{k,s}|^2$. Accordingly, each BS s -th BS serves k -th UE, with $k = \arg\min_i P_{i,s}^r$. This rule needs a prior design for phase shifts of the IRS elements

to determine the total received signals at the UEs from the indirect paths. Furthermore, sub-optimal solutions for BS–IRS associations can be obtained with convergent results using the successive refinement algorithm [14]. This iterative algorithm updates the associated BS for each deployed IRS successively, while keeping the associations of all other IRSs fixed until no further improvement in the defined performance can be achieved. Specifically, consider the BS–IRS association solution as $\lambda_l = [\lambda_{l,1}, \lambda_{l,2}, \dots, \lambda_{l,S}]$, $l \in \mathcal{L}$. For a given design of BS–IRS association, represented by $\{\lambda_l\}$ for all L IRSs, let $R_{\text{sum}}(\lambda_l)$ denote the corresponding maximum sum rate. During algorithm iteration, each l' -th IRS optimizes $\lambda_{l'}$ successively, while the associated BSs for all other IRSs, denoted as $\{\lambda_l\}_{l \neq l'}$, are fixed to maximize $R_{\text{sum}}(\lambda_{l'}; \{\lambda_l\}_{l \neq l'})$. Enumerating the S BSs for the l' -th IRS efficiently obtains the optimal solution to l' . Subsequently, the associated BS for the l' -th IRS is updated, and this process continues for the next IRS. Thus, this algorithm is a timely strategy since it needs to solve many iterations until there is no further improvement in its performance.

The challenges within our large-scale system stem from intricate interactions and dependencies among diverse entities, including BSs, UEs, and IRSs. These entities are intricately linked through associations, beamforming, and power control decisions, introducing complexity to the optimization of resources. Associations dictate which UEs are served by specific BSs and IRSs, while the beamforming matrices of IRSs impact signal propagation characteristics. The mobility of UEs and the non-convex nature of the optimization

problem add to the complexity. Facilitating enhanced coordination and adaptability becomes crucial for efficient resource utilization and interference minimization amid dynamic network conditions. Enhanced coordination ensures harmony among network entities, while adaptability permits dynamic adjustments to configurations and strategies. To tackle these challenges, we propose the MADRL approach. In MADRL, each agent, representing a network entity, autonomously learns and refines its decision-making policy through collaborative learning. This approach not only ensures fair resource allocation, minimum rate requirements, and efficient power usage, but also enhances the overall adaptability and coordination of the network in dynamic conditions.

4. Proposed MADRL-Based Algorithm

In this section, we present the proposed MADRL-based algorithm. We begin by reformulating the optimization problem in the form of an MMDP. Subsequently, we detail the proposed algorithm, considering two configurations of the IRS. Finally, an analysis of the algorithm's convergence and complexity is provided.

4.1. MAMDP Formulation of the Optimization Problem

To apply MADRL, the initial step involves transforming the optimization problem into an MAMDP format. The MAMDP typically comprises five essential elements for each agent: agent i , state s_t^i , action a_t^i , reward function r_t^i , and policy π_t^i [37].

Given our studied system and formulated optimization problem, the behavior of all BSs and IRSs is modeled as a MADRL MMDP, which involves a set of agents (i.e., BSs and IRSs): $\mathcal{G} = \{\mathcal{S} \cup \mathcal{L}\}$. Thus, we have a total of states sets $\{\mathcal{S}^j\}_{j=1}^{S+L}$, actions $\{\mathcal{A}^j\}_{j=1}^{S+L}$, and rewards $\{\mathcal{R}^j\}_{j=1}^{S+L}$. Under the MAMDP, at each time step t , each agent receives an immediate observation or state $s_t^i \in \mathcal{S}^i$ from the environment, where $\mathcal{S}^i$ represents the potential set of states for agent i . In response to this state, each agent chooses an action $a_t^i \in \mathcal{A}^i$, where $\mathcal{A}^i$ denotes the set of possible actions for agent i . Subsequently, the environment updates the states to the next states s_{t+1}^i and supplies a performance metric reward r_t^i for that state–action pair (s_t^i, a_t^i) . The mapping from states to actions, governed by each agent's local policy $\pi_t^i(s_t^i, a_t^i)$, signifies the probability of selecting action a_t^i for the current state s_t^i . Over time, each agent strives to learn an optimal policy by maximizing cumulative rewards in the collaborative multi-agent environment. All the agents operate independently in a shared environment and then form joint action $a_t = \{a_t^1, a_t^2, \dots, a_t^{S+L}\} \in \mathcal{A}$, where $\mathcal{A} = \{\mathcal{A}^1, \mathcal{A}^2, \dots, \mathcal{A}^{S+L}\}$. The subsequent state s_{t+1} is updated from the current $s_t = \{s_t^1, s_t^2, \dots, s_t^{S+L}\}$, providing the joint rewards $r_t = \{r_t^1, r_t^2, \dots, r_t^{S+L}\} \in \mathcal{R}$ to the joint state–action pairs. Accordingly, the joint optimal policy $\pi_t(s_t, a_t) = \{\pi_t^1(s_t^1, a_t^1), \pi_t^2(s_t^2, a_t^2), \dots, \pi_t^{S+L}(s_t^{S+L}, a_t^{S+L})\}$ is formed, which can maximize the long-term rewards received. The states, actions, rewards, and policies of our proposed MADRL-based approaches are described below.

1. States: For each BS agent $j \in \mathcal{S}$, we define its state as the current active beamforming at time step t , $(w_j)_t$, the previous transmit power at time step $t - 1$, $(p_j)_{t-1}$, the previous BS–UE associations vector at time step $t - 1$, $(\{\rho_{k,j}\}_{k=1}^K)_{t-1}$, the previous BS–IRSs associations vector at time step $t - 1$, $(\{\lambda_{l,j}\}_{l=1}^L)_{t-1}$, and the current position to the served UE, $(\{(x_k, y_k, z_k)\})_t, k \in \mathcal{K}_s$, which is mathematically given by:

$$s_t^j = \left\{ (w_j)_t, (p_j)_{t-1}, (\{\rho_{k,j}\}_{k=1}^K)_{t-1}, (\{\lambda_{l,j}\}_{l=1}^L)_{t-1}, (\{(x_k, y_k, z_k)\})_t \right\} \in \mathcal{S}, \quad (13)$$

whose dimension is $2M + K + L + 4$. The term $(w_j)_t$, determined by the CSI, serves as a surrogate for the CSI in s_t^j at time t to capture variations in wireless channel conditions. Note that directly integrating the channel estimates among all system nodes into the local observation may not be feasible due to the impracticality of handling high-dimensional feedback, while for each IRS agent $l \in \mathcal{L}$, we define its

state as the previous passive beamforming vector at time step $t - 1$, $(\theta_l)_{t-1}$, which is defined as:

$$s_t^l = \{(\theta_l)_{t-1}\} \in S, \quad (14)$$

whose dimension is $2N$.

2. Actions: For each BS agent $j \in S$, its action includes the transmit power, $(p_j)_t$, BS–UE association vector, $(\{\rho_{k,j}\}_{k=1}^K)_t$, and BS–IRSs associations vector, $(\{\lambda_{l,j}\}_{l=1}^L)_t$. However, in order to handle constraint (11e) and reduce the action space of the BS–IRS associations from $L \cdot S$ to L , we shift the BS–IRSs associations as a subfunction to IRS agents, where each IRS has to choose one BS. Moreover, in order to reduce the infeasible BS–UE association actions and handle constraints (11c) and (11d) during training, a centralized DRL element for all BS–UE associations is trained at one BS, and then a copy is delivered to all BSs. Consequently, the action of each BS agent j is denoted by:

$$a_t^j = \{(p_j)_t, (\rho'_{k,j})_t\} \in A, \quad (15)$$

where $(\rho'_{k,j})_t$ is the index of the served k -th UE by the s -th BS. Correspondingly, the dimension is simply 2, and the action space of BS–UE associations equals K , while for each IRS agent $l \in \mathcal{L}$, we define its action as the current passive beamforming, $(\theta_l)_t$, and BS–IRS association index element, $(\lambda'_{l,j})_t$, where $\lambda'_{l,j}$ is the index of the chosen j -th BS for l -th IRS. For example, if at time step t , the agent IRS l chose to associate with the fourth BS, then $(\lambda_{l,4})_t = 1$ in the BS–IRS associations matrix. The action of each IRS agent l is denoted by:

$$a_t^l = \{(\theta_l)_t, (\lambda'_{l,j})_t\} \in A. \quad (16)$$

which outputs a dimension of $2N + 1$ and an action space of BSs–IRS associations of S .

3. Rewards: As the objective function of the optimization problem is to maximize the sum rate of served UEs, establishing a connection between the reward and the objective function drives the realization of the MMDP goal to augment long-term rewards. Noting that each agent with a separate reward may behave selfishly, therefore hindering the global optimal solution. Thus, we assume all agents have the same reward after the joint action is performed. The environment sets the instant reward for all cooperative agents in this framework as the total sum rate achieved. Note that all the constraints in (P1) are handled above, except the constraint (11b), which considers the minimum QoS requirements constraints. If the resulting joint action fails to satisfy the minimum data rates of the UEs, agents are penalized to encourage them to modify inappropriate actions. The reward function includes a penalty value, which is represented by the difference between the minimum achieved rate and the specified constraint rate. Thus, the agents will receive a negative penalty if the minimum QoS is not satisfied for the UEs. Otherwise, all agents will receive positive rewards represented by the total utility achieved. Here, we set such a penalty value instead of zero to limit the sparsity of the rewards and reflect how far the agents are from achieving the minimum QoS. Therefore, each agent $i \in \mathcal{G}$ can receive an instant reward, determined as follows:

$$r_t^i = \begin{cases} (R_{\text{sum}})_t, & \text{if } (R_{k,s})_t \geq R_{\min}, \forall k \in \mathcal{K}_s, \forall s \in S, \\ -|(R_k^{\min})_t - R_{\min}|, & \text{if } (R_{k,s})_t < R_{\min}, \exists k \in \mathcal{K}_s, \forall s \in S, \end{cases} \quad (17)$$

where $R_k^{\min}$ represents the minimum achieved data rate of all the served UEs at the t -th time step.

4. Policies: In the proposed MADRL framework, effective collaboration hinges on creating an optimal joint policy, denoted as $\pi_t(s_t, a_t)$. This joint policy plays a pivotal role in guiding collective decision-making among agents, directing them to choose actions that enhance cumulative rewards in the dynamic multi-agent environment.

The process of learning and contributing to this joint policy is iterative and adaptive for each agent. In response to ever-changing environmental states, agents continually adjust their local policies by evaluating the outcomes of their actions based on observed states. Reinforcement signals aid this learning process, guiding agents toward actions that maximize system utility over the long term through reward-based feedback. Further sections will provide a closer look at how agents refine their local policies, contributing to the overall success of the collaborative MADRL framework.

4.2. The Implementation of the Proposed MADRL-Based Algorithm

In this subsection, the proposed MADRL-based algorithm frameworks are described in detail to solve the optimization problem under continuous and discrete phases of the IRSs. These proposed algorithms mainly consist of two phases, namely training and implementation. Each phase will be detailed below.

4.2.1. MADRL-Based Algorithm under Continuous Phases Case

Note that conventional one-agent-DRL-based approaches can be applied to solve the problem (P1) by exploring all possible actions. However, in this work, the problem has huge state and action spaces with both continuous and discrete optimization variables, posing challenges for applying a one-agent-based approach. Accordingly, we propose a cooperative MADRL-based framework to handle the challenges and solve the problem efficiently, where the multi-agent-based approach can work together to reduce the state and action spaces. Since the joint action includes both discrete and continuous actions, we apply different DRL algorithms to train the agents.

As shown in Figure 2, for each BS agent, we apply two DRL units: deep deterministic policy gradient (DDPG) to optimize transmit power allocation, $(p_j)_t, j \in \mathcal{S}$, and deep Q-network (DQN) to optimize BS-UE associations vector, $(\{\rho_{k,j}\}_{k=1}^K)_t$. For each IRS agent, we apply two DRL units: DDPG for continuous phases, i.e., $(\theta_l)_t$ and DQN for BSs-IRS associations vector, i.e., $(\{\lambda_{l,j}\}_{l=1}^L)_t$. Note that, according to our system design, we have a closed form (12) to obtain continuous phases by knowing the associations' designs; thus, we do not need DDPG units for the phases at the IRSs in the continuous case. In addition, we implement a centralized DDPG unit that copies the local action to each BS for controlling their transmit powers. Additionally, all the DRL algorithms are connected to replay memory to store all trained experiences, where a central control is employed to control the local and joint experiences.

The DQN algorithm integrates deep neural networks (DNNs) with Q-learning, where the DNNs are used to approximate the Q-function, replacing the tabular representation for the Q-value function. DQN consists of two networks, i.e., one is the main neural network called Q network with parameters θ_q , which estimates the Q-value for the current state and action pair. The other is a target Q network with parameters $\theta_{q'}$. Note that θ_q and $\theta_{q'}$ have identical structures, but $\theta_{q'}$ is used to estimate the Q-values of the next step state-action pair. Furthermore, the DQN algorithm stores all acquired experiences in the integrated replay memory. Figure 3a illustrates the architecture of the DQN algorithm. Meanwhile, the DDPG algorithm follows the DPG theorem, which learns a Q-function and a policy simultaneously. It can deal with continuous action spaces, addressing the restriction of the DQN network to discrete action spaces only. As illustrated in Figure 3b, DDPG consists of four networks, i.e., the actor-network with parameters θ_μ , the target actor-network with $\theta_{\mu'}$, the critic-network with parameters θ_q , and the target critic-network with $\theta_{q'}$. Specifically, the DDPG algorithm trains the actor network to select the best actions given the current state, while the performance critic network estimates Q-value performance. Similar to DQN, the target networks mirror the structure of the training networks but have distinct parameters, and they utilize experience replay memory.

Given the structure of the DRL algorithms outlined above, we employ the proposed algorithm to solve (P1), as outlined in Algorithm 1. In the beginning, the integrated networks for all DDPG and DQN units are initialized, and all training hyper-parameters

are set. Assuming the agents' interactions within the environment are segmented into I episodes, each comprising a finite number of T steps. Then, at the start of every training episode, the algorithm resets the environment with different locations for the UEs and their corresponding channel gains. Subsequently, all channels linking network nodes are computed. We apply Additionally, random initialization is applied to the transmit powers, beamforming vectors, and association matrices. Accordingly, the state s_t is defined and provided as input stimulation. The agents then provide feedback on the decisions of local actions related to the integrated DRL units at each agent to form the joint action. Then, the BSs' transmit powers, BSs' active beamforming, IRSs' passive beamforming, and BS-IRS-UE association matrices are formulated. Thereby, the current joint reward is computed, and the subsequent state s_{t+1} is defined. This information is stored as the joint experience $\{s_t, a_t, r_t, s_{t+1}\}$ to create the integrated buffer. This algorithm requires a minimum of N_B experiences to initiate network updates and learning. Consequently, the preceding steps are reiterated N_B times to establish the initial mini-batch, but this occurs exclusively in the first episode. Following the creation of N_B experiences, the algorithm proceeds to the training phase. Note that the flow chart of Algorithm 1 is figured in Figure 4 to provide a visual representation of the algorithmic steps and facilitate better understanding.

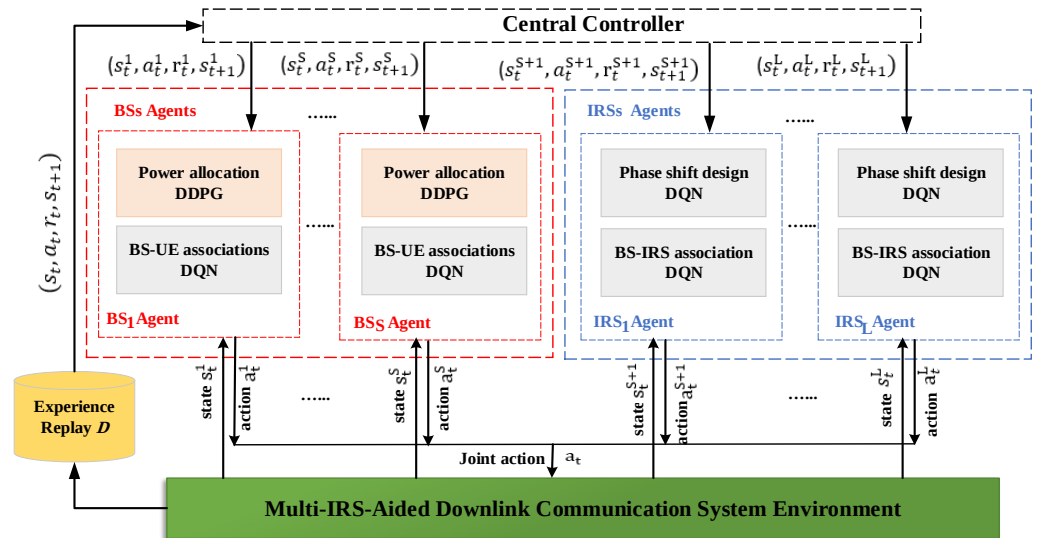


Figure 2. Proposed MADRL-based frameworks under IRS configurations cases.

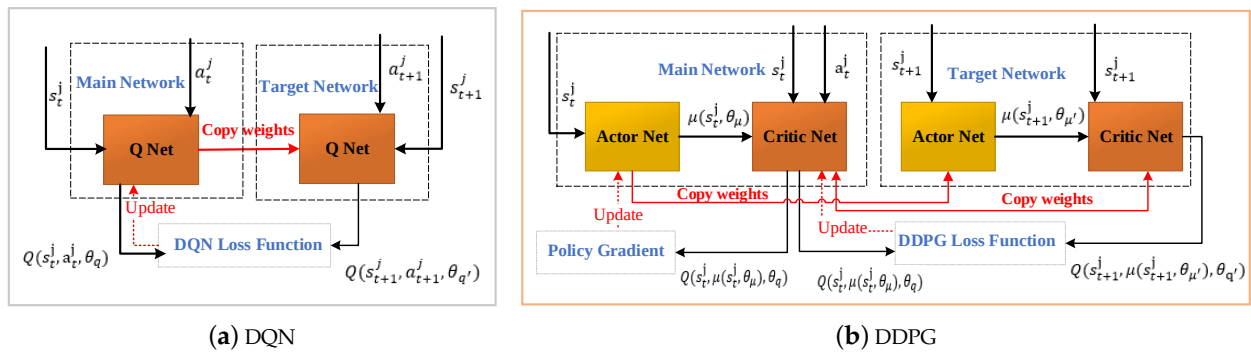


Figure 3. DRL units architecture: (a) DQN unit; (b) DDPG unit.

At each time step t , the algorithm extracts local states, actions, and rewards from the chosen N_B samples. Specifically, during the learning process, the action value function of each agent j is updated according to the Bellman equation as follows:

$$Q^j(s_t^j, a_t^j) \leftarrow Q^j(s_t^j, a_t^j) + \alpha \left(r_t^j + \gamma \max_{a^j} Q^j(s_{t+1}^j, a^j) - Q^j(s_t^j, a_t^j) \right), \quad (18)$$

where α denotes the learning rate. Therefore, each agent j obtains the optimal value function by solving:

$$Q^{*j}(s_t^j, a_t^j) = \mathbb{E}_{s_{t+1}^j} \left[r_t^j + \gamma \max_{a^j} Q^{*j}(s_{t+1}^j, a^j) \mid s_t^j, a_t^j \right], \quad (19)$$

where (s_{t+1}^j, a_{t+1}^j) represents the next state–action pair for agent j .

With the DQN algorithm structure, each agent j with a current state s_t^j selects its action a_t^j using ϵ -greedy exploration. The agent either chooses a random action with a probability of ϵ or selects an action for which the value function Q is greatest with a probability of $1 - \epsilon$.

$$a_t^j = \begin{cases} \text{random action,} & \text{with probability } \epsilon, \\ \operatorname{argmax}_{a_t^j \in \mathbf{A}_j} Q(s_t^j, a_t^j), & \text{with probability } 1 - \epsilon. \end{cases} \quad (20)$$

Since the association designs and IRS reflection phases are shared by the system nodes, fully distributed operation for the agents without environment state sharing may be infeasible for a large-sized network. In our cooperative multi-agent framework, we adopt independent learning with state and reward sharing, where all agents form a joint action and calculate the reward in (17). With this joint response, a new state s_{t+1} is formed, and a new experience $\{s_t, a_t, r_t, s_{t+1}\}$ is added to the replay. A random sample of size N_B is chosen from the replay to update the weights of the Q -network. Consequently, the target value function Q is given by:

$$y_t^j = \begin{cases} r_t^j, & \text{if } s_{t+1}^j \text{ is terminal state,} \\ r_t^j + \gamma \max_{a_{t+1}^j \in \mathbf{A}_j} Q(s_{t+1}^j, a_{t+1}^j; \theta_{q'}), & \text{otherwise.} \end{cases} \quad (21)$$

By using gradient descent, the Q network of agent j can be trained by minimizing the following DQN loss function, expressed as:

$$L(\theta_q) = (y_t^j - Q(s_t^j, a_t^j; \theta_q))^2. \quad (22)$$

Next, a smoothing update is applied to the target critic parameters, gradually adjusting them to track the changes in the critic network. This process helps stabilize the learning and prevent rapid fluctuations in the target values, as follows:

$$\theta_{q'} \leftarrow \tau_{\text{DQN}} \theta_q + (1 - \tau_{\text{DQN}}) \theta_{q'}, \quad (23)$$

where $\tau_{\text{DQN}} \ll 1$ represents the target smooth factor for the DQN algorithm. Subsequently, adjust the probability threshold ϵ for selecting a random action according to the decay rate α_{decay} . At the end of each training step, if ϵ exceeds the minimum value $\epsilon_{\min}$, it is adjusted as follows:

$$\epsilon = \epsilon(1 - \alpha_{\text{decay}}). \quad (24)$$

Note that ϵ is maintained from the end of one episode to the beginning of the next one, resulting in a gradual, uniform decrease over multiple episodes until it reaches $\epsilon_{\min}$.

Algorithm 1 The proposed MADRL-based algorithm.

```

1: Initialize the power allocation DDPG units for all the BSs agents, including actor-network  $\mu(s; \theta_\mu)$ ,
   target actor network  $\mu'(s; \theta_{\mu'})$ , critic network  $Q(s, a; \theta_{q_1})$ , target critic network  $Q'(s, a; \theta_{q_1'})$  with
   parameters  $\theta_{\mu'} = \theta_\mu$  and  $\theta_{q_1'} = \theta_{q_1}$ .
2: Initialize the BS-UE associations DQN units, including Q-network  $Q(s, a; \theta_{q_2})$ , target Q-network
    $Q'(s, a; \theta_{q_2'})$  with parameters  $\theta_{q_2'} = \theta_{q_2}$ .
3: Initialize the passive beamforming DQN units for all the IRS agents, including Q-network
    $Q(s, a; \theta_{q_3})$ , target Q-network  $Q'(s, a; \theta_{q_3'})$  with parameters  $\theta_{q_3'} = \theta_{q_3}$ .
4: Initialize the BS-IRS associations DQN units, including Q-network  $Q(s, a; \theta_{q_4})$ , target Q-network
    $Q'(s, a; \theta_{q_4'})$  with parameters  $\theta_{q_4'} = \theta_{q_4}$ .
5: Initialize the learning rates  $\alpha_{\text{DQN}}$  and  $\alpha_{\text{DDPG}}$ , discount factor  $\gamma$ , target smooth factors  $\tau_{\text{DQN}}$  and
    $\tau_{\text{DDPG}}$  mini-batch size  $N_B$  and an empty experience replay  $D$ .
6: for episode  $i = 1, 2, \dots, I$  do
7:   Reset the environment and acquire  $(x_k, y_k, z_k)$ ,  $\{\mathbf{h}_{s,dk}^H\}$ ,  $\{\mathbf{G}_{s,l}\}$ , and  $\{\mathbf{H}_{rl}\}$ ,  $\forall s, l, k$ .
8:   Initialize  $\{p_s\}_{s=1}^S$ ,  $\{w_s\}_{s=1}^S$ ,  $\{\Theta_l\}_{l=1}^L$ ,  $\{\rho_{k,s}\}$  and  $\{\lambda_{l,s}\}$ ,  $\forall s, l, k$ .
9:   Form initial state  $s_1$  to all agents.
10:  for step  $t = 1, 2, \dots, T$  do
11:    for agent  $j = 1, 2, \dots, S + L$  do
12:      if  $j \in \mathcal{S}$  then
13:        Power allocation DDPG unit selects transmit power, i.e.,  $(p_j)_t$ .
14:        BS-UE association DQN unit selects  $(\rho'_{k,j})_t$ .
15:        Form the action  $a_t^j$  of the BS  $j$  as (15).
16:      end if
17:      if  $j \in \mathcal{L}$  then
18:        Passive beamforming DQN unit selects  $(\theta_l)_t$ .
19:        BSs-IRS association DQN units select  $(\lambda'_{l,j})_t$ .
20:        Form the action  $a_t^l$  of the IRS  $l$  as (16).
21:      end if
22:    end for
23:    Form the joint action  $a_t$  by all agents.
24:    Form the BSs transmit powers  $(\{p_s\}_{s=1}^S)_t$ , the BSs active beamforming  $(\{w_s\}_{s=1}^S)_t$ , IRSs
    passive beamforming  $(\{\Theta_l\}_{l=1}^L)_t$  and BS-IRS-UE associated matrices, i.e.,  $\rho_t, \lambda_t$ .
25:    Calculate the current joint reward  $r_t$  using (17).
26:    Form the next joint state  $s_{t+1}$  for all agents.
27:    Fill the buffer  $D$  by the experience  $\{s_t, a_t, r_t, s_{t+1}\}$ .
28:    for agent  $j = 1, 2, \dots, S + L$  do
29:      if update then
30:        Select from the buffer a  $N_B$  experiences  $\{s_i, a_i, r_i, s_{i+1}\}$ .
31:        if  $j \in \mathcal{S}$  then
32:          Calculate the target values of the DDPG units according to (26).
33:          Calculate the target values of the DQN units according to (21).
34:          Update every  $U$  steps:  $\theta_\mu$  according to (28),  $\theta_{q_1}$  according to (27),  $\theta_{\mu'}$  according to
          (29), and  $\theta_{q_1'}$  according to (30).
35:          Update  $\theta_{q_2}$  according to (22), update  $\theta_{q_2'}$  according to (23).
36:        end if
37:        if  $j \in \mathcal{L}$  then
38:          Set the value function target of DQN units as (21).
39:          Update  $\theta_{q_3}$  according to (22), update  $\theta_{q_3'}$  according to (23).
40:          Update  $\theta_{q_4}$  according to (22), update  $\theta_{q_4'}$  according to (23).
41:        end if
42:      end if
43:    end for
44:    Assign the updated state as  $s_{t+1}$ .
45:  end for
46: end for

```

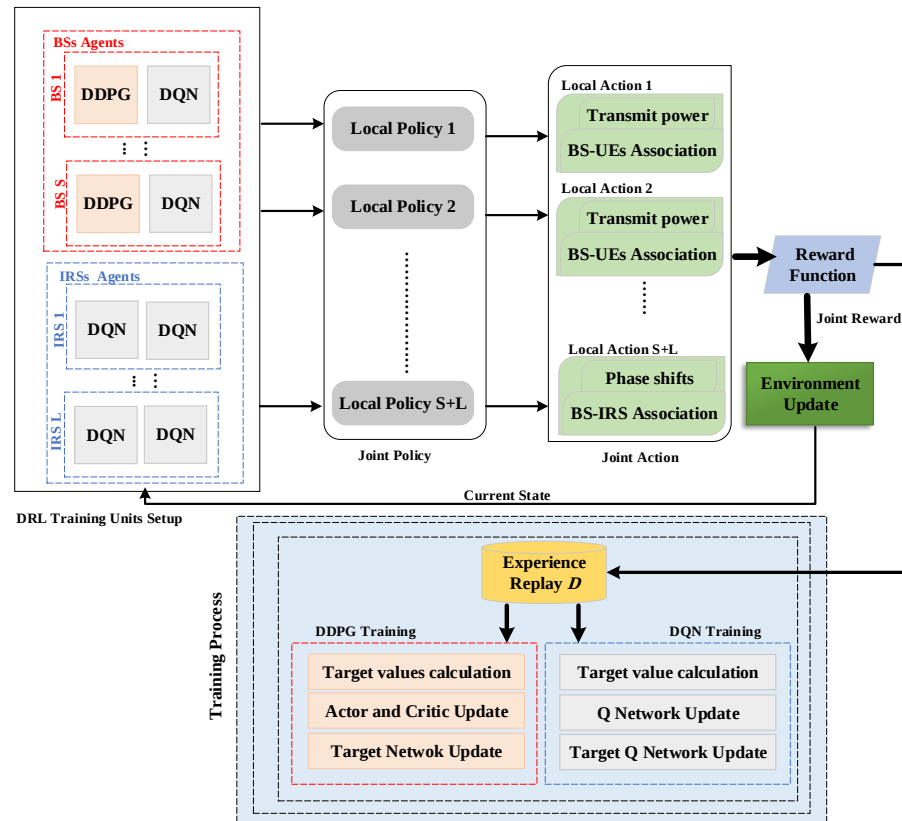


Figure 4. Flow chart of the proposed MADRL-based algorithm.

The DDPG algorithm is employed for optimizing the transmit power of the BSs. In contrast to DQN, which employs an ϵ -greedy strategy, DDPG applies the DPG theorem to update the network parameters, estimating the current action from the current state. The action is evaluated using the deterministic policy of the actor network. To ensure effective exploration in the continuous action space during the DDPG training process, an exploration policy is formed by incorporating a defined noise into the DDPG action, as follows:

$$a_t^j = \mu(s_t^j; \theta_\mu) + \mathcal{N}_t, \quad (25)$$

where $\mathcal{N}_t$ is the deployed noise model. Accordingly, the target value function for each agent j is calculated as follows:

$$y_t^j = \begin{cases} r_t^j, & \text{if } s_{t+1}^j \text{ is a terminal state,} \\ r_t^j + \gamma Q(s_{t+1}^j, \mu(s_{t+1}^j; \theta_{\mu'}); \theta_{q'}), & \text{otherwise.} \end{cases} \quad (26)$$

During training, the algorithm updates the critic network through the minimization of the critic loss function based on the selected mini-batch samples, as follows:

$$L(\theta_q) = \frac{1}{N_B} \sum_{t=1}^{N_B} (y_t^j - Q(s_t^j, a_t^j; \theta_q))^2. \quad (27)$$

Following these, a policy gradient function is applied to update the actor network, involving the calculation of the derivative of the policy's performance J with respect to θ_μ , as computed below:

$$\nabla_{\theta_\mu} J = \frac{1}{N_B} \sum_{t=1}^{N_B} \underbrace{\nabla_{a^j} Q(s_t^j, \mu(s_t^j; \theta_\mu); \theta_q)}_{G_{a_t}} \bigg| \underbrace{\nabla_{\theta_\mu} \mu(s_t^j; \theta_\mu)}_{G_{\mu_t}}, \quad (28)$$

where G_{μ_t} and G_{a_t} denote the gradient of the actor output and the critic output concerning the actor parameters and the actor action, respectively. Next, a smoothing mechanism is adopted to update the target networks, expressed as follows:

$$\theta_{\mu'} \leftarrow \tau_{\text{DDPG}} \theta_{\mu} + (1 - \tau_{\text{DDPG}}) \theta_{\mu'}, \quad (29)$$

$$\theta_{q'} \leftarrow \tau_{\text{DDPG}} \theta_q + (1 - \tau_{\text{DDPG}}) \theta_{q'}, \quad (30)$$

where $\tau_{\text{DDPG}} \ll 1$ is the smooth factor for the target networks in the DDPG algorithm.

4.2.2. MADRL-Based Algorithm under Discrete Phases Case

Due to the practical implementation and hardware constraints, the n -th element of the l -th IRS generally takes a finite set of discrete values. Considering the discrete case of IRS phase shifts, we propose a cooperative MADRL-based framework under the discrete case in this subsection. The BS agents are similar to the case with continuous cases, where each BS agent $j \in \mathcal{S}$, adopts a DDPG unit for power allocation, $(p_j)_t$, and a DQN unit for BS–UE associations, $(\{\rho_{k,j}\}_{k=1}^K)_t$. However, for the IRS agents, each IRS agent $l \in \mathcal{L}$, adopts two DQN units, i.e., one to handle the discrete phase shift actions, i.e., $(\theta_l)_t$, and the other for the BSs–IRS associations vector, i.e., $(\{\lambda_{l,j}\}_{j=1}^L)_t$, as shown in Figure 2.

During the implementation stage, the cooperative agents observe the environmental state and make decisions regarding the powers of the BSs, phases of the IRSs, and the configuration of associations based on the trained models. After that, they observe the updated state and make another decision. In this setting, we assume that the locations of the UEs are accessible to the BSs agents and then shared with IRSs agents; for example, predicting the location information involves processing the signals and channels during the uplink communication, including path loss coefficients, the angle of arrival (AoA), the angle of departure (AoD), and the time of arrival (ToA) [38]. The proposed MADRL-based algorithm integrates the contents of Sections 4.2.1 and 4.2.2, and is summarized in Algorithm 1.

4.3. Analysis of the Proposed MADRL-Based Algorithm

In this subsection, we analyze the convergence and computational complexity of the proposed cooperative MADRL-based algorithm under continuous and discrete phases. Then, the implementation complexity is compared to benchmark schemes.

4.3.1. Convergence Analysis

The proposed cooperative MADRL-based algorithm integrates DQN and DDPG algorithms, which learn policies independently. Therefore, the convergence of the proposed algorithm relies on the convergence of the DDPG and DQN algorithms. Initially, we analyze the convergence of traditional Q-learning-based algorithms, as the DQN and DDPG algorithms are considered evolutionary extensions of Q-learning algorithms. According to [39], the Q-learning-based algorithm converges to the optimal Q-function when the learning rate guarantees $0 \leq \alpha_t \leq 1$, $\sum_t \alpha_t = \infty$, and $\sum_t \alpha_t^2 < \infty$. Moreover, consider I_m as the m -dimensional unit hyper-cube $[0, 1]^m$, and $\varphi(t)$ is a continuous function that is bounded, non-constant, and monotonically increasing. Additionally, $C(I_m)$ denotes the space of continuous functions on I_m . For any provided function $f \in C(I_m)$ and $\epsilon > 0$, there exists a $F(x) = \sum_{i=1}^N v_i \varphi(w_i^T x + b_i)$ with real vector $w_i \in R^m$, $(i = 1, \dots, N)$, real constants $v_i, b_i \in R$, and integer N to obtain an approximation of the function f 's realization. Note that f is independent of φ and guarantees $|F(x) - f(x)| < \epsilon, \forall x \in I_m$. In essence, functions represented by the form $F(x)$ are densely distributed within $C(I_m)$. Following the Stone–Weierstrass theorem [40], a large neural network and the proper choice of initial conditions can be used to get close to any non-linear continuous function. Thus, it can be concluded that the DQN algorithms converge to optimal solutions by a large-size neural network and appropriately tuned free parameters and initial parameters. For the DDPG algorithm, as it applies a deterministic policy gradient theory, it shows convergence to sub-

optimal solutions [41]. In summary, the proposed cooperative MADRL-based algorithm is guaranteed to converge under both continuous and discrete phases.

4.3.2. Computational Complexity Analysis

The computational complexity of the proposed algorithm can be split into two primary parts: forward propagation and backpropagation for action selection and training models [31], respectively.

- Action selection complexity: Let $|S_{in}|$ denote the input dimension, n represent the number of nodes in the hidden layer, and $|A|$ indicate the output dimension. Specifically, $|A|$ denotes the size of the action space A^i in the DQN algorithms and the dimension of the continuous action a^i in the DDPG algorithms [28]. The complexity of a single decision for one three-layer DNN is approximately:

$$\mathcal{O}(|S_{in}| \cdot n + |A| \cdot n + n). \quad (31)$$

In our system, we have S BSs and L IRSs acting as agents, all sharing the same state. Thus, the states, actions, and related complexities under continuous and discrete cases can be defined as follows:

$$|S_{in}| = (S + L) \cdot (2M + K + L + 4 + 2N), \quad (32)$$

$$|A| = \begin{cases} S \cdot (1 + K) + L \cdot (S), & \text{if continuous phases,} \\ S \cdot (1 + K) + L \cdot (B^N L + S), & \text{if discrete phases,} \end{cases} \quad (33)$$

$$\text{Complexity} \approx \begin{cases} \mathcal{O}\left((S \cdot (2M + 2K + L + 2N + 5) + L \cdot (2M + K + L + 2N + 4 + S) + 1) \cdot n\right), & \text{if continuous phases,} \\ \mathcal{O}\left((S \cdot (2M + 2K + L + 2N + 5) + L \cdot (2M + K + L + 2N + B^N L + S + 4) + 1) \cdot n\right), & \text{if discrete phases,} \end{cases} \quad (34)$$

where B represents the quantization level with $B = 2^b$.

- Training process complexity: The computational complexity of backpropagation in the DQN and DDPG algorithms for fully connected DNN (FC-DNN) per training step is approximated as:

$$\mathcal{O}(|S_{in}| \cdot |A| \cdot n), \quad (35)$$

where $|A|$ is determined based on the types of actions. Thus, the training complexity of all DNNs is approximated as:

$$\text{Complexity} \approx \begin{cases} \mathcal{O}(I \cdot T \cdot N_B \cdot (S + L) \cdot |S_{in}| \cdot |A| \cdot n), & \text{for backpropagation,} \\ \mathcal{O}(I \cdot T \cdot N_B \cdot (S + L) \cdot (|S_{in}| + |A| + 1) \cdot n), & \text{for action selection.} \end{cases} \quad (36)$$

Here, the total complexity is approximated as the backpropagation training complexity, which has a higher order than the action selection complexity.

4.3.3. Implementation Complexity Analysis

Although the training stage requires a significant amount of time for the agents, it is conducted offline using a high-performance computational server, thereby enhancing overall time efficiency. In contrast, the implementation stage is carried out online with parallel processing, emphasizing a time-saving approach during this phase of the process. Specifically, in the implementation stage, each agent operates independently in parallel, enabling the execution of individual actions without dependence on other agents. Additionally, the DRL elements within each agent function concurrently to contribute to the final joint action of that particular agent. Consequently, the overall implementation time

of the entire MADRL algorithm is contingent upon the time taken by the DRL unit that consumes the maximum time.

Actually, the implementation run time of the DQN units in the proposed algorithm is more time-consuming than that of the DDPG units. The reason for this lies in the exploration strategy employed by DQN, where the epsilon-greedy strategy is used. During implementation, this exploration strategy may result in increased computation as the agent evaluates multiple Q-values for different actions, taking into account the exploration probability. In contrast, DDPG, being a deterministic policy gradient method, directly outputs a specific action without stochastic exploration during implementation. This deterministic nature simplifies the implementation process and can potentially lead to faster inference times. It is worth noting that we can reduce the complexity of the DQN units by dividing the function of each DQN into multiple parallel DQNs, for example, each DQN for each element or some elements. This approach to optimization could be explored in future work. Thus, we can compute the implementation complexity of the proposed algorithms under continuous and discrete phases as follows:

$$\text{Complexity} \approx \begin{cases} \mathcal{O}((2M + 2K + L + 2N + 5) \cdot n), & \text{if continuous phases,} \\ \mathcal{O}((2M + K + L + 2N + B^N L + S + 5) \cdot n), & \text{if discrete phases.} \end{cases} \quad (37)$$

It seems that the complexity of the proposed algorithms is a polynomial function of L, N, M, K, B , and n . Thus, we can represent the implementation complexity of the proposed algorithm as $\mathcal{O}(P(L, N, M, K, n))$ and $\mathcal{O}(P(L, N, M, K, B, S, n))$, under continuous and discrete cases, respectively. In Table 1, we compare the computational complexities of the prior arts with our proposed one. From the table, it shows that the prior approaches have an exponential complexity, which is higher than the polynomial complexity of the proposed approach. Here, I_t denotes the number of reached iterations until convergence of successive algorithm is met.

Table 1. Implementation complexity comparison.

Approach	Complexity
Exhaustive Approach	$\mathcal{O}(S^{K+L})$
Graph-theoretic Algorithm	$\mathcal{O}(K \cdot S^2)$ [42]
Successive Refinement Algorithm	$\mathcal{O}(I_t \cdot K \cdot S^3)$ [13,14]
Proposed MADRL Algorithm (continuous)	$\mathcal{O}(P(L, N, M, K, n))$
Proposed MADRL Algorithm (discrete)	$\mathcal{O}(P(L, N, M, K, B, S, n))$

In Tables 2 and 3, we assess the proposed average implementation run time MADRL-based algorithm under both continuous and discrete phases in comparison to the successive refinement algorithm for different numbers of reflecting elements under both continuous and discrete phase configurations, respectively. The average implementation run time is calculated from online predictions and averaged through 100 random initializations. These results are conducted on a server-grade machine featuring an Intel(R) Xeon(R) Silver 4110 CPU with a clock speed of 2.10 GHz and equipped with 16 GB of RAM. Specifically, under continuous phases, the overall implementation time of the entire MADRL algorithm is contingent upon the time taken by the DQN units for BS–UE associations at the BS agents, which consumes the maximum time. Meanwhile, under discrete phases, the overall implementation time of the entire MADRL algorithm is contingent upon the time taken by the DQN units to configure the discrete phases at the IRS agents. Additionally, it is observed that the complexity of the proposed algorithm under the discrete case is higher than in the continuous case, primarily due to the increasing number of elements, which amplifies the complexity of the DQN units. However, when compared to iterative approaches, the proposed algorithm demonstrates efficiency in run-time. As indicated in the tables, the proposed MADRL-based algorithm has a lower average prediction run time than the

successive refinement algorithm, which necessitates solving many sub-iterations alternately until the solution converges.

Table 2. Average implementation run-time (seconds) under continuous phases case.

N	5	10	15	20
Algorithm				
Successive Refinement Algorithm	44.6637	78.5485	120.3662	153.0799
Proposed MADRL Algorithm	0.4372	0.4380	0.4424	0.4669

Table 3. Average implementation run-time (seconds) under discrete phases case.

N	5	10	15	20
Algorithm				
Successive Refinement Algorithm	80.8549	94.6200	101.2413	134.2253
Proposed MADRL Algorithm	0.4478	1.2505	38.9429	1050

5. Simulation Results

In this section, we present the numerical results for the proposed algorithm under the continuous phases scenario and the discrete phases scenario. For the system parameters regarding the network deployment, we follow the latest relevant works in [14]. Figure 5 depicts the 2D deployment of the studied system, where four BSs are positioned by coordinates (x_s, y_s, z_s) at $(-30, 4, 5)$ [m], $(-30, -4, 5)$ [m], $(30, -4, 5)$ [m], and $(30, 4, 5)$ [m], respectively. Additionally, six IRSs are positioned by coordinates denoted by (x_l, y_l, z_l) as follows: $(-5, -10.5, 5)$ [m], $(-2.5, -12, 5)$ [m], $(2.5, -11.5, 5)$ [m], $(-5, -10.5, 5)$ [m], $(-2.5, -12, 5)$ [m], and $(2.5, 11.5, 5)$ [m], respectively. These IRSs share the same BS height. A total of six UEs are distributed randomly along the circumference of a circle. Specifically, the circle is centered at $(0, 0)$ with a diameter of $D = 20$ [m] and positioned in front of the IRSs. We assume the UEs change their positions at the beginning of each episode. Following a random walk, each UE has a moving angle g_k ($0 \leq g_k \leq 2\pi$) and a moving speed v_k ($0 \leq v_k \leq 1$) [m/s] within the confined region, as depicted in Figure 5. In the simulations, we set $2d^l = 2d^s = \lambda_c$, $N_H = 1$ for odd N , $N_H = 2$ for even N , and increase N_V at the IRSs. The path losses in the channels are expressed by the models $32.6 + 36.7 \log_{10}(d^{\text{direct}})$ [dB] [8] and $35.6 + 22 \log_{10}(d^{\text{indirect}})$ [dB] [8], respectively. Here, d^{direct} represents the distance between the s -th BS and the k -th UE in the direct link, while d^{indirect} signifies the corresponding distance between nodes in indirect links, namely the s -th BS to the l -th IRS or the l -th IRS to the k -th UE. Other simulation parameters include $\varepsilon = 10$, noise power $\sigma_k^2 = -80$ [dBm] and $R_{\min} = 1$ [bps].

In the proposed cooperative MADRL framework, all DRL units, including DDPG and DQN units, are structured from FC-DNNs. Each network is composed of five layers, i.e., one input layer, one output layer, and two hidden layers, with a normalization layer in between to expedite convergence and reduce training time. The dimensions of the networks are contingent on the sizes of states and actions. We configure the hidden layer to be suitable for our studied system setup [21]. This configuration strikes a balance between model complexity and performance, as confirmed through empirical testing. In addition, Tanh and Relu models are applied as activation functions, chosen for their facilitation of gradient descent and backpropagation. To explore the action set in the DDPG algorithm, we introduce a complex Gaussian noise with zero mean and variance, i.e., σ^2 . Denote T_s as the sampling time; accordingly, $\sigma^2 \cdot \sqrt{T_s}$ typically satisfy (1% to 10%) of the range of the DDPG action. The remaining hyper-parameters are listed in Table 4. Part of these parameters is aligned with [21,31], while others are chosen through extensive trials.

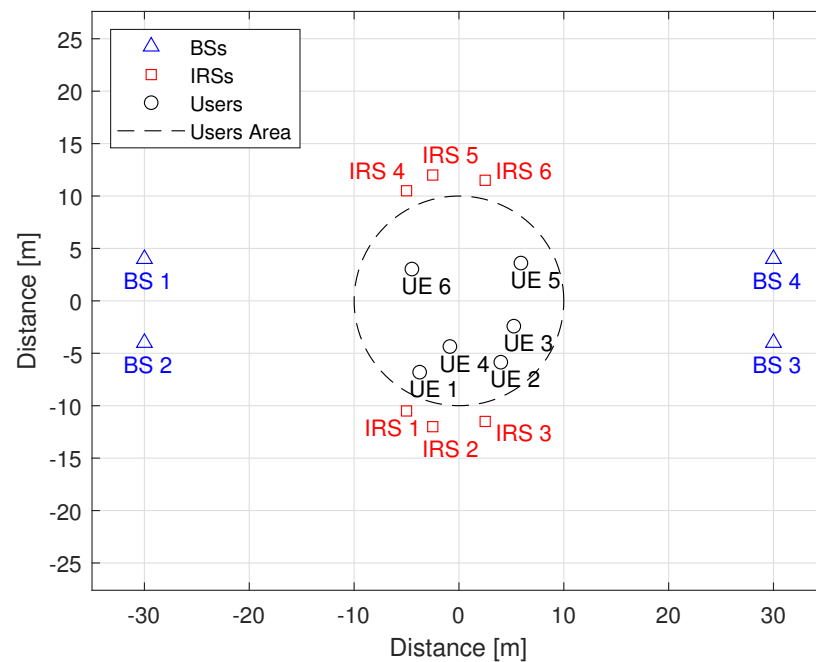


Figure 5. 2D—system deployment.

To demonstrate the performance of the proposed algorithm, we compare it with benchmark schemes. The scenarios and benchmark schemes under consideration include:

- Without IRSs: conventional system without IRSs aid, i.e., switch off all the deployed IRSs and no related associations.
- Random setup: system with the BS–IRS–UE associations are set up randomly.
- Nearest setup: system with the BS–IRS–UE associations are set up based on the nearest association rule.
- Successive refinement-based algorithm: system with the BS–IRS associations are set up based on the iterative successive refinement optimization algorithm [13,14].

Table 4. Proposed MADRL framework hyper-parameters settings.

Hyper-Parameter	Setting
Learning rates $\alpha_{\text{DQN}}, \alpha_{\text{DDPG}}$	0.001
Discount factor γ	0.95
Buffer size C	200,000
Mini-batch size N_B	16
Probability threshold ϵ	1
Min Probability threshold $\epsilon_{\min}$	0.01
Decay rate α_{decay}	0.0050
Target smooth factors $\tau_{\text{DQN}}, \tau_{\text{DDPG}}$	0.001
Target update frequency U	1
Sampling time T_s	1

5.1. Performance Analysis of the Proposed MADRL-Based Algorithm

The proposed cooperative MADRL-based algorithm should be trained to yield high sum rates and output performance. Consequently, we monitor the network during episode steps' training and investigate the effects of the learning rates and target smooth factors for both DQN and DDPG units. These parameters are considered critical hyper-parameters for the agents. It is important to note that we initially consider the continuous phases case to tune these parameters and then apply the tuned values to the discrete phases case.

Figure 6 plots the average reward versus the episode steps under various learning rates for the agents' learning units. The average rewards are computed by accumulating

instantaneous rewards over episode steps. It is observed that, for any given learning rate, agents initially receive low rewards. As the number of steps increases, these rewards grow until convergence to the episode occurs. This is primarily due to the fact that the relatively lower experience in the buffer during the early steps leads to a higher sample correlation and, consequently, poorer training performance for the agents. However, as experiences accumulate in the buffer, the agents collaborate to enhance the long-term rewards. Specifically, the figure illustrates that the curve attains the highest average rewards at a learning rate of 0.001. The improvement in performance resulting from the lower learning rate comes with the drawback of an extended convergence time. Additionally, it is worth noting that higher rates lead to increased oscillations, resulting in performance degradation. This implies that an optimal learning rate is necessary, as excessively low or high values may compromise both performance and convergence. Thus, we configure the learning rates for both DQN and DDPG units in all the agents to be 0.001, ensuring the optimization of long-term rewards and aligning with the main target of maximizing the system sum rate.

We also fine-tune the smoothing factor used in both DQN and DDPG units. Typically, it is necessary to set this value to less than 1 to ensure smoothness. In Figure 7, the average rewards are plotted against episode steps across various smooth factors. Observations reveal that utilizing a smooth value of 0.001 produces superior performance when compared to alternative values. Furthermore, it is observed that an increase in this value diminishes average rewards and degrades overall performance. This suggests that employing low smooth values for the agents is well-adapted for updating the target networks and converging towards optimal solutions for the studied system. Thus, we employ a smooth factor of 0.001 for both the DQN and DDPG units in all agents.

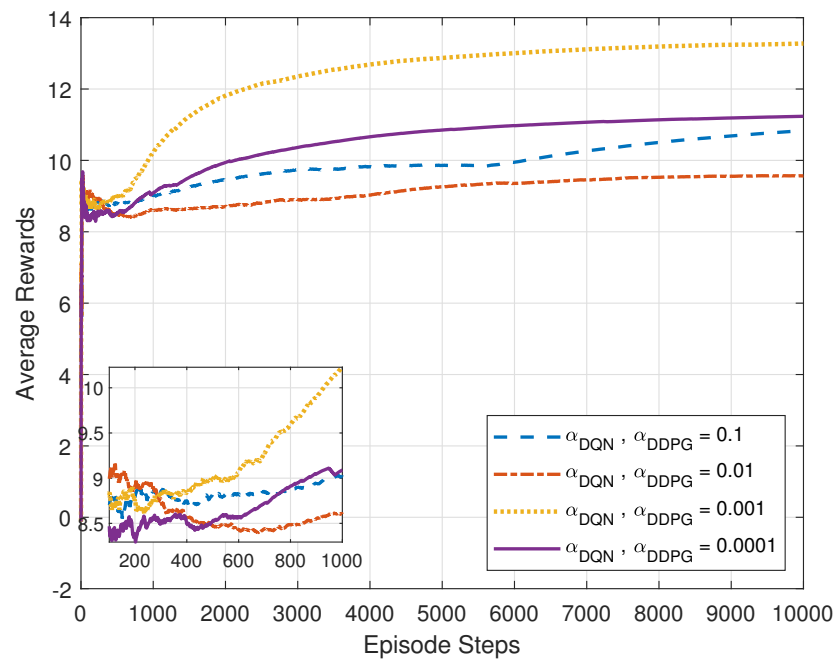


Figure 6. Average rewards versus episode steps with various learning rates.

Figure 8 depicts the average total rewards achieved during total episodes with the adjusted parameters under both continuous and discrete cases. This average is determined by summing the cumulative values across all episodes. At the beginning of each episode, it is important to note that the environment undergoes a reset to assign new channel gains and UE positions. We assume a random walk for UEs, where each UE has a moving angle g_k ($0 \leq g_k \leq 2\pi$) and a moving speed v_k ($0 \leq v_k \leq 1$) [m/s] within the confined region, as depicted in Figure 5. The plot demonstrates that the average total achieved sum rate increases with the number of training episodes until convergence is achieved. This

indicates that the agents update their policy after each episode, progressively optimizing toward the objective of maximizing the system sum rate under varying positions for UEs with different channel gains.

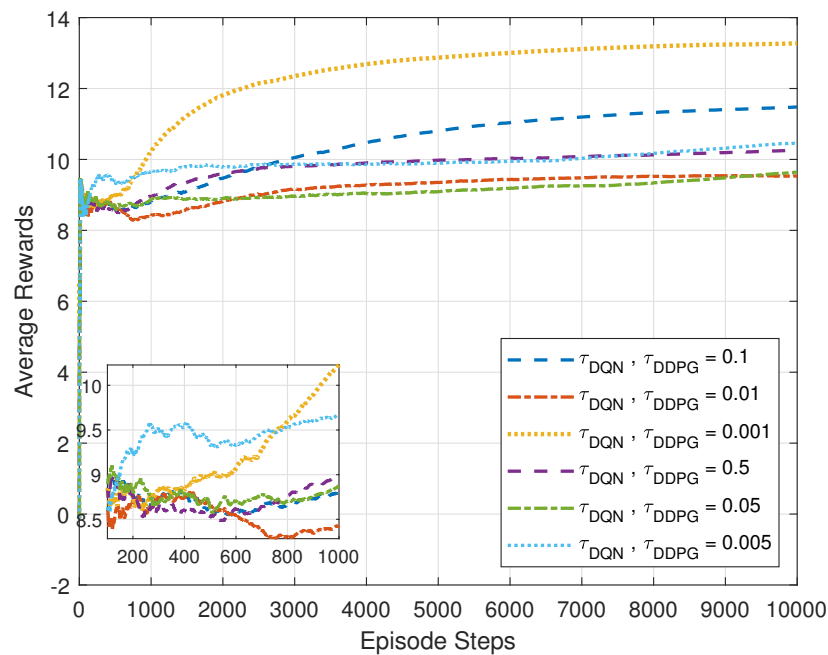


Figure 7. Average rewards versus episode steps with various target smoothing factors.

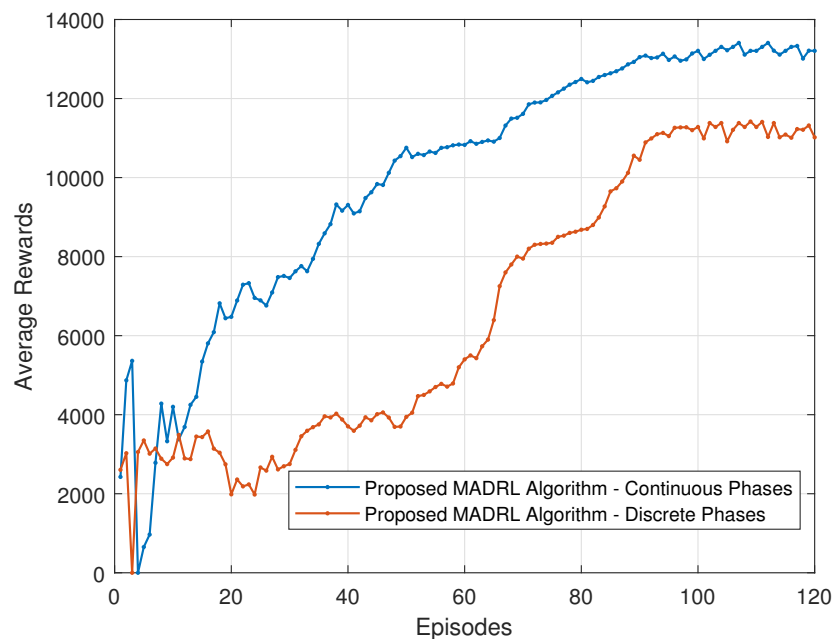


Figure 8. Average rewards versus learning episodes.

5.2. Performance Comparison of the Proposed MADRL-Based Algorithm under the Continuous Phases Case

In this subsection, we compare our proposed cooperative MADRL-based algorithm under the case of the continuous phases with the presented benchmark schemes regarding the optimized total sum rate of the served UEs.

Figure 9 plots the optimized sum instantaneous sum rate versus the BS maximum transmit power under different schemes, where UEs' locations are generated from a single

random realization, as illustrated in Figure 5. Note that all the benchmarks are enumerated on five discrete power levels to form a power space in the range $P_d = \{0, \dots, p_{\max}\}$.

From Figure 9, we note that our cooperative MADRL-based algorithm achieves the highest optimized sum rate compared to other benchmark schemes. This underscores the superiority of the proposed algorithm in reaching optimized values for the sum rate. The figure shows that the optimized sum rate is improved by increasing the BS maximum transmit power $p_{\max}$ until it reaches a limit, where the power range of BS is consistent with [4,8,23,31]. Similar to a traditional network without IRSs-aid, this behavior arises from the potential increase in transmit power from the BSs, boost impacts the strength of both the desired signal and co-channel interference at the served UEs in a comparable manner. Consequently, the SINR experiences enhancement until reaching a limit as $p_{\max}$ grows larger. In contrast, it is noted that with any given $p_{\max}$, well-optimizing the BS–IRS–UE associations can dramatically improve the optimized rate beyond the converged level. Contrasted with scenarios incorporating IRSs, the total achieved rate for the scenario without IRSs is the lowest because this scenario does not have the power gain due to the deployment of IRSs. In the conventional system without IRSs, the associations between BSs and UEs are set based on the nearest association rule. Then, the performance of the random setup scheme ranks as the second-worst. This indicates the importance of optimizing the BS–IRS–UE associations rather than randomly selecting them to maximize the benefits of IRS deployment. Applying the nearest association rule between system nodes enhances the sum rate compared to random associations. Moreover, when applying the successive refinement algorithm for optimizing the BS–IRS association, considering the BS–UE association based on nearest and RSRP setups, we can notice that the performance of the successive refinement algorithm-based RSRP is better than the successive refinement algorithm-based nearest. This is because, for any given configuration of IRS phases, the maximum RSRP association rule is a CSI-based association strategy that is determined by the strength of the UE received signal from both direct and indirect paths instead of distance-based association in the nearest setup. This is also well suited to the MRT active beamforming design in our system model. Concluding, we can find improved optimizations by applying the proposed algorithm, which jointly optimizes all optimization variables for maximizing the long-term sum rate.

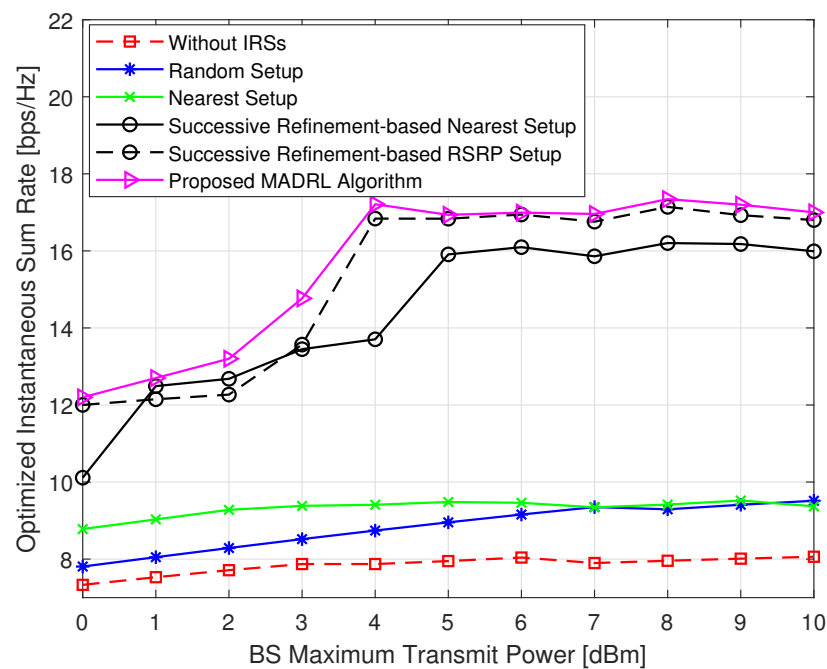


Figure 9. Optimized instantaneous sum rate versus BS maximum transmit power for different approaches under continuous phases.

Figure 10 plots the optimized BS–IRS–UE associations under different approaches for maximizing the sum rate at $p_{\max} = 4$ dBm. At each time step, the served UEs are marked by the same color as their associated BSs and IRSs, while the waiting UEs or those not served at this time step are marked in “gray”. Figure 11 shows the corresponding achieved system utility by the serving BSs. In Figure 10a, without the aid of IRSs, each BS serves the nearest UE. This has the lowest network utility compared to approaches with the IRS aid, with UE 3 and 6 identified as bottleneck users. In Figure 10b, with IRSs activated and random associations between nodes, some gains are observed, reflecting the impact of setting up the BS–IRS–UE associations to maximize the total network utility. In Figure 10c, associating the nearest IRSs to aid the transmission between the served UEs and their serving BSs enhances the rates of the bottleneck UEs in Figure 10a and increases the total sum rate.

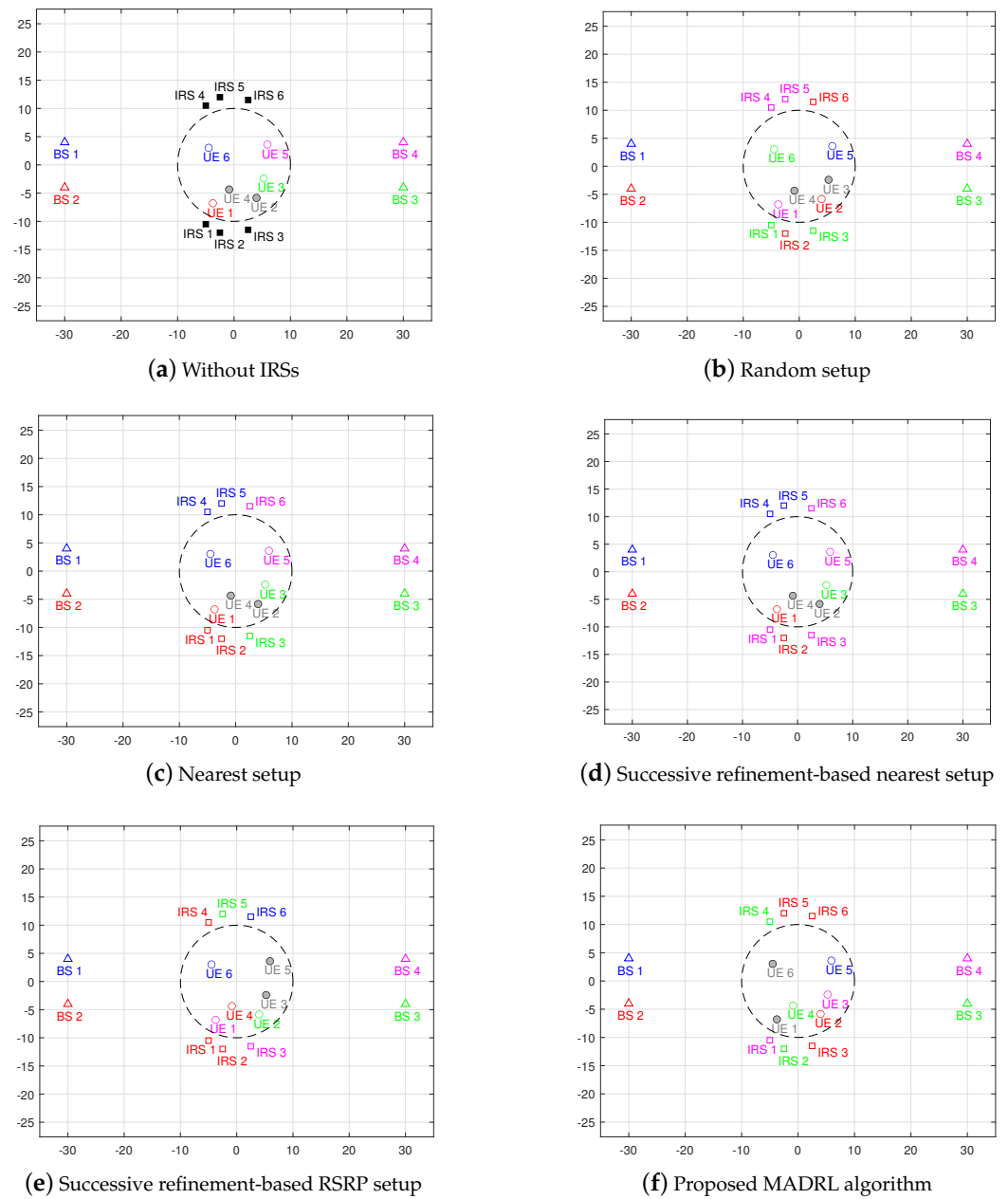


Figure 10. Optimized BS–IRS–User associations under continuous phases case for different approaches: (a) without IRSs; (b) random setup; (c) nearest setup; (d) successive refinement-based nearest setup; (e) successive refinement-based RSRP setup; (f) proposed MADRL algorithm.

In Figure 10d, the successive refinement algorithm-based nearest algorithm optimizes the BS–IRS association to maximize the achieved utility. The BS1–IRS association is kept the same, as it already achieved a high rate by serving UE6. To maximize the sum rate for other BSs and increase the achieved rates at their served UEs, re-association is performed, as shown in Figure 10d. For example, BS4 is associated with two extra-aiding IRSs to aid the transmission of the UE5. In Figure 10e, by re-associating the BS–UE based on RSRP instead of the nearest approach, the successive refinement algorithm demonstrates that it can increase the BSs' sum rates. This is because RSRP considers the direct and indirect channel conditions between BSs and UEs, instead of just the distance, as in the nearest approach case. For example, BS4 serves UE1 instead of UE5, maximizing the utility of BS4. Note that the BS–UE in the RSRP-based association is determined based on the random phases for the IRSs, which may affect the optimal BS–UE with the instant CSI. Figure 10f plots the optimized BS–IRS–UE associations by applying the proposed MADRL-based algorithm, which led to maximizing the total achieved rates, as shown in Figure 11. This reflects that the agents work together over time to improve the association matrices and, as a result, fine-tuning the phases and the BSs' power controls to maximize the overall system rate.

Figure 12 plots the optimized total average sum rate versus the number of elements per IRS in the continuous case for various schemes. Here, we average each simulated result over 100 random UE position initializations following their mobility model as described above, with a transmit power constraint of 0 dBm for each BS. The plot shows that the average achieved rate for the proposed algorithm increases with the growing number of IRS reflecting elements, surpassing the iterative successive refinement algorithm. This indicates that increasing the number of reflecting elements, along with well-optimized association design and power control, enhances the gains brought by these IRSs, thereby improving the utility of the source BSs. Furthermore, the random and nearest setup algorithms achieve comparable results and, notably, their performance degrades as N increases. This degradation is due to their static association design, which relies on the fixed locations of UEs.

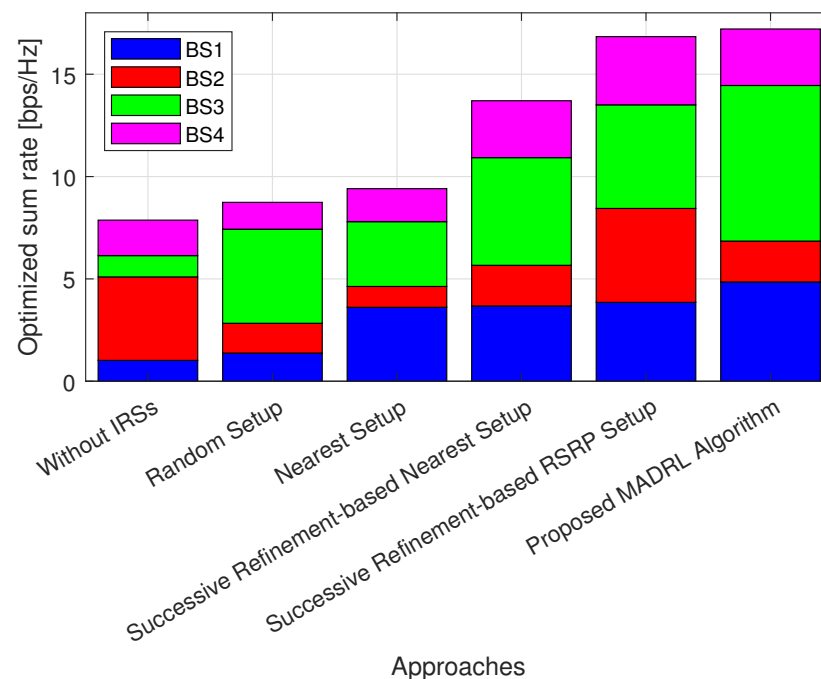


Figure 11. Optimized sum rate versus different approaches under continuous phases case.

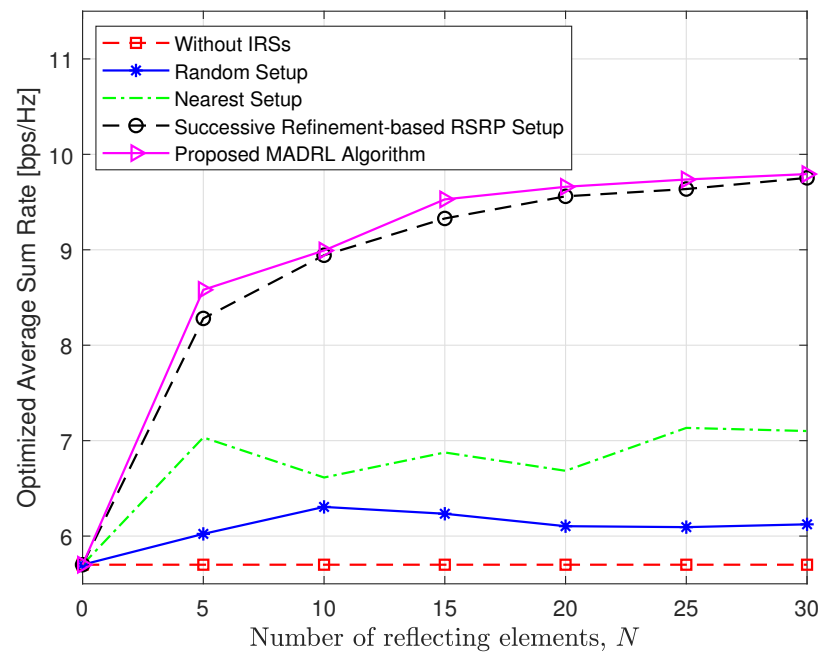


Figure 12. Optimized average sum rate versus number of reflecting elements for different approaches under continuous phases.

5.3. Performance Comparison of the Proposed MADRL-Based Algorithm under the Discrete Phases Case

In this subsection, we compare our proposed cooperative MADRL-based algorithm under the case of the discrete phases with the considered benchmark schemes in terms of the total sum rate of the served UEs.

Figure 13 plots the optimized sum instantaneous achieved rate for the served UEs versus the BS maximum transmit power under different schemes. As shown in Figure 13, the proposed cooperative MADRL-based algorithm allows the achieved sum rate to reach high values compared with other approaches, indicating the effectiveness of the proposed algorithm. Similar to Figure 9, under the case of the discrete phases, a system not relying on assistance from IRSs results in the lowest achieved rate. Moreover, the achieved rate of applying random associations between system nodes is still constrained by the association design, which is enhanced when applying the nearest association setup algorithm. Finally, the successive refinement algorithm shows sub-optimal solutions that depend on the iterative process. In the benchmark schemes, it is important to highlight that the discrete phase shifts are determined by quantizing the continuous phase shifts obtained to their nearest values in F_l . This quantization is expressed as $\theta_{l,n}^* = \underset{\theta \in F_l}{\operatorname{argmin}} |\theta - \theta_{l,n}^*|$, where $\theta_{l,n}^*$ is the optimal discrete phase shift of the n -th element at the l -th IRS.

Under the case of the discrete phases, Figure 14 depicts the optimized BS–IRS–UE associations under both the successive refinement and the proposed cooperative MADRL algorithm designed to maximize the sum rate. Note that we omit association figures for other approaches, since they are the same as those in Figure 10. Figure 15 illustrates the corresponding achieved sum rate by the serving BSs. As depicted, applying the proposed cooperative MADRL algorithm results in a higher achieved sum rate at the BSs. This verifies the superiority of our proposed approach over other benchmarks.

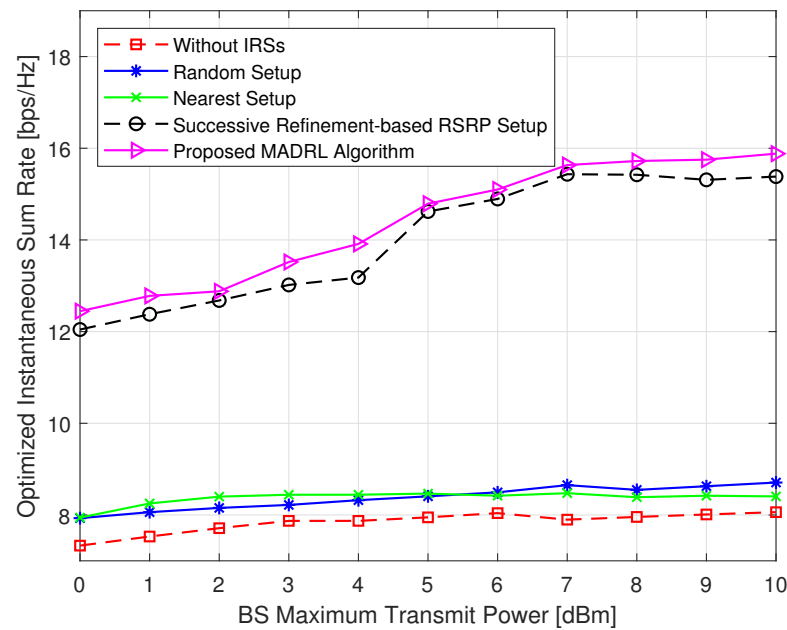


Figure 13. Optimized instantaneous sum rate versus BS maximum transmit power for different approaches under discrete phases.

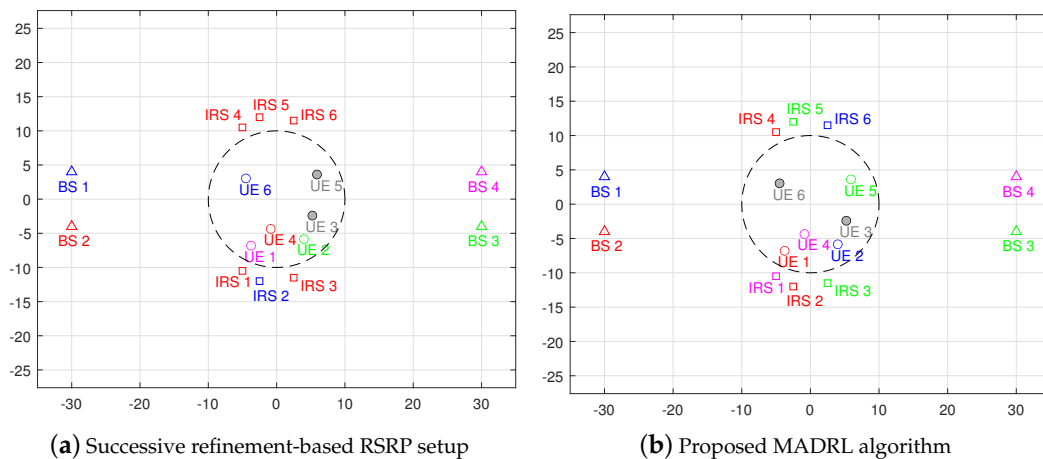


Figure 14. Optimized BS-IRS-User associations under discrete phases case for different approaches: (a) successive refinement-based RSRP setup; (b) proposed MADRL algorithm.

Figure 16 plots the optimized total average sum rate versus the number of elements per IRS under the case of the discrete phase for the optimizing schemes. We average each simulated result over 100 random UE position initializations, with a transmit power constraint of 0 dBm for each BS. Similar to Figure 12, it is observed that the average achieved rate for the proposed algorithm increases with the growing number of the IRSs reflecting units and exhibits the highest achieved sum rate compared to other schemes. The successive refinement algorithm shows the second-best performance. For comparison, the random and nearest setup algorithms also achieve comparable results, with minimal enhancement in their performance as N increases. This indicates that increasing the number of reflecting elements with well-optimized association design and power control enhances the gains brought by these IRSs, thereby helping the BSs improve the total sum rate.

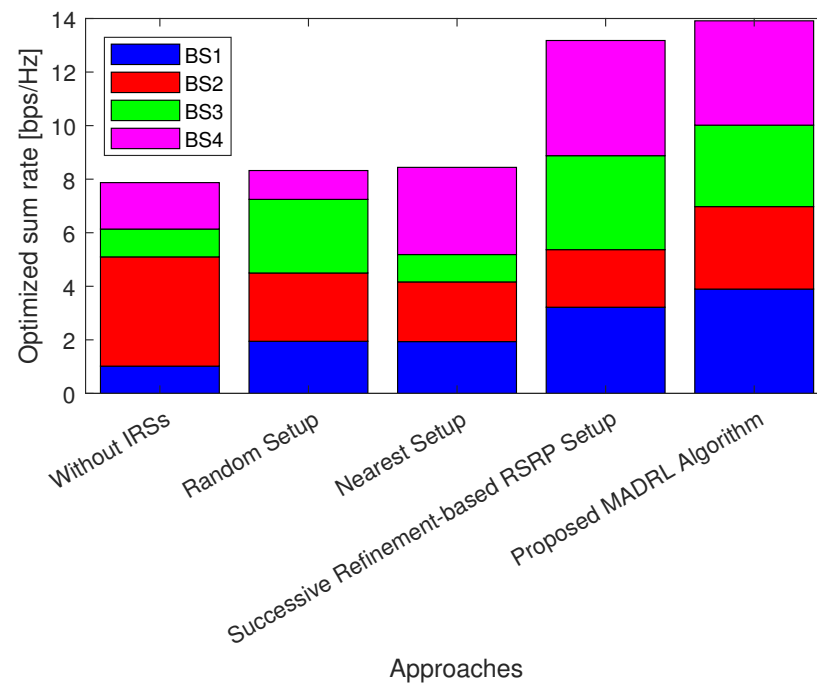


Figure 15. Optimized sum rate versus different approaches under discrete phases case.

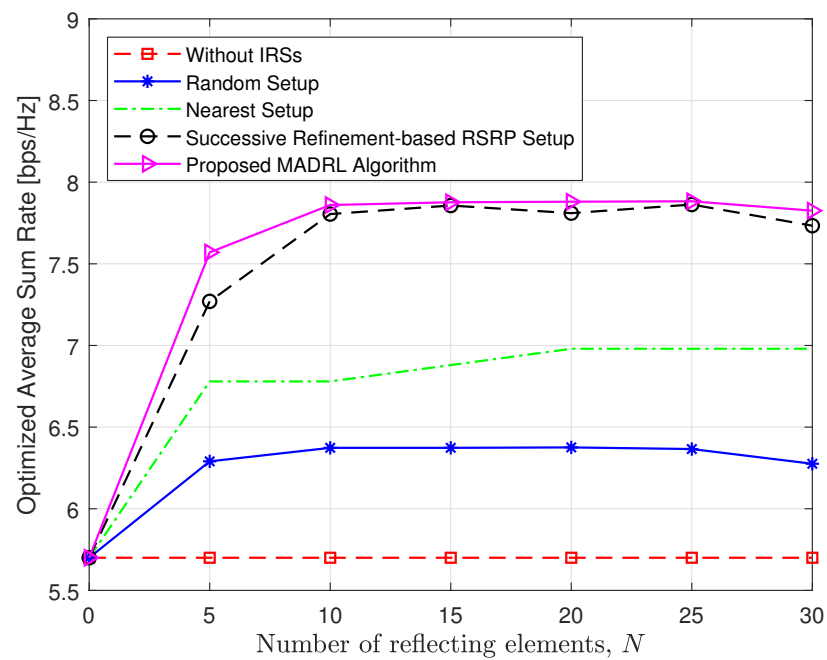


Figure 16. Optimized average sum rate versus number of reflecting elements for different approaches under discrete phases.

Figure 17 plots the optimized total average sum rate versus the number of elements per IRS under different resolutions for different schemes. In this context, we specifically consider both $b = 1$ and $b = 2$ for discrete phase shifts at each IRS. Similarly, we average each simulated result over 100 random UE position initializations, with a transmit power constraint of 0 dBm for each BS. From the figure, it is observed that as N increases, the performance gap between the considered schemes widens with increasing resolutions. This trend comes from the fact that higher resolutions provide more flexibility in designing the phases of the IRSs, leading to an enhancement in the total sum rate.

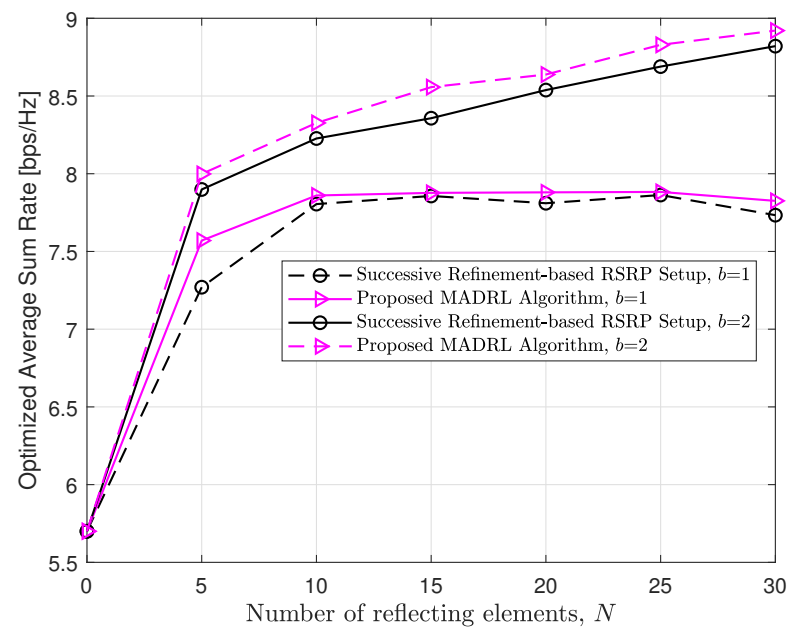


Figure 17. Optimized average sum rate versus number of reflecting elements for different approaches under discrete phases with different resolutions.

6. Conclusions

In this work, we have investigated a large-scale communication system with multi-BS assisted by multi-IRS to serve multi-UE, where the joint optimization of BS power control, IRS beamforming configuration, and associations design was studied to maximize the total sum rate. The cooperative MADRL-based algorithm was proposed for joint optimization under continuous and discrete phase cases. The demonstrated capability of these approaches includes real-time predictions in various channel conditions with different UE locations, while ensuring performance meets predefined criteria. The fine-tuning was applied to the proposed models, ensuring robust stability and convergence with a focus on maximizing long-term objectives. Our simulation results revealed that the proposed algorithm outperformed the presented iterative benchmarks with regard to the sum rate.

Author Contributions: M.F.: Conceptualization, Methodology, Software, Validation, Formal analysis, and Writing—original draft; Z.F.: Supervision and Funding acquisition; J.G.: Conceptualization, Data curation, Writing—review and editing, Visualization and Project administration; M.S.A.: Investigation and Resources. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by National Natural Science Foundation of China under Grant 62001029 and in part by Beijing Natural Science Foundation under Grant L202015.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Saad, W.; Bennis, M.; Chen, M. A vision of 6G wireless systems: Applications, trends, technologies, and open research problems. *IEEE Netw.* **2019**, *34*, 134–142. [\[CrossRef\]](#)
2. Gong, S.; Lu, X.; Hoang, D.T.; Niyato, D.; Shu, L.; Kim, D.I.; Liang, Y.C. Toward smart wireless communications via intelligent reflecting surfaces: A contemporary survey. *IEEE Commun. Surv. Tutor.* **2020**, *22*, 2283–2314. [\[CrossRef\]](#)
3. Di Renzo, M.; Debbah, M.; Phan-Huy, D.T.; Zappone, A.; Alouini, M.S.; Yuen, C.; Sciancalepore, V.; Alexandropoulos, G.C.; Hoydis, J.; Gacanin, H.; et al. Smart radio environments empowered by reconfigurable AI meta-surfaces: An idea whose time has come. *EURASIP J. Wirel. Commun. Netw.* **2019**, *2019*, 129. [\[CrossRef\]](#)
4. Wu, Q.; Zhang, R. Intelligent reflecting surface enhanced wireless network via joint active and passive beamforming. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 5394–5409. [\[CrossRef\]](#)

5. Hu, S.; Rusek, F.; Edfors, O. Beyond massive MIMO: The potential of data transmission with large intelligent surfaces. *IEEE Trans. Signal Process.* **2018**, *66*, 2746–2758. [\[CrossRef\]](#)
6. Fu, M.; Zhou, Y.; Shi, Y. Intelligent reflecting surface for downlink non-orthogonal multiple access networks. In Proceedings of the 2019 IEEE Globecom Workshops (GC Wkshps), Waikoloa, HI, UA, 9–13 December 2019; IEEE: New York, NY, USA, 2019; pp. 1–6.
7. Huang, C.; Zappone, A.; Alexandropoulos, G.C.; Debbah, M.; Yuen, C. Reconfigurable intelligent surfaces for energy efficiency in wireless communication. *IEEE Trans. Wirel. Commun.* **2019**, *18*, 4157–4170. [\[CrossRef\]](#)
8. Guo, H.; Liang, Y.C.; Chen, J.; Larsson, E.G. Weighted sum-rate maximization for reconfigurable intelligent surface aided wireless networks. *IEEE Trans. Wirel. Commun.* **2020**, *19*, 3064–3076. [\[CrossRef\]](#)
9. Li, Z.; Hua, M.; Wang, Q.; Song, Q. Weighted sum-rate maximization for multi-IRS aided cooperative transmission. *IEEE Wirel. Commun. Lett.* **2020**, *9*, 1620–1624. [\[CrossRef\]](#)
10. Han, P.; Zhou, Z.; Wang, Z. Joint user association and passive beamforming in heterogeneous networks with reconfigurable intelligent surfaces. *IEEE Commun. Lett.* **2021**, *25*, 3041–3045. [\[CrossRef\]](#)
11. Alwazani, H.; Nadeem, Q.U.A.; Chaaban, A. Performance Analysis under IRS-User Association for Distributed IRSs Assisted MISO Systems. *arXiv* **2021**, arXiv:2111.02531.
12. Zhao, D.; Lu, H.; Wang, Y.; Sun, H. Joint passive beamforming and user association optimization for IRS-assisted mmWave systems. In Proceedings of the GLOBECOM 2020—2020 IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; IEEE: New York, NY, USA, 2020; pp. 1–6.
13. Mei, W.; Zhang, R. Joint base station-IRS-user association in multi-IRS-aided wireless network. In Proceedings of the GLOBECOM 2020—2020 IEEE Global Communications Conference, Taipei, Taiwan, 7–11 December 2020; IEEE: New York, NY, USA, 2020; pp. 1–6.
14. Mei, W.; Zhang, R. Performance analysis and user association optimization for wireless network aided by multiple intelligent reflecting surfaces. *IEEE Trans. Commun.* **2021**, *69*, 6296–6312. [\[CrossRef\]](#)
15. Zhao, D.; Lu, H.; Wang, Y.; Sun, H.; Gui, Y. Joint power allocation and user association optimization for IRS-assisted mmWave systems. *IEEE Trans. Wirel. Commun.* **2021**, *21*, 577–590. [\[CrossRef\]](#)
16. Taghavi, E.M.; Alizadeh, A.; Rajatheva, N.; Vu, M.; Latva-aho, M. User association in millimeter wave cellular networks with intelligent reflecting surfaces. In Proceedings of the 2021 IEEE 93rd Vehicular Technology Conference (VTC2021-Spring), Virtual Event, 25–28 April 2021; IEEE: New York, NY, USA, 2021; pp. 1–6.
17. Liu, Y.; Liu, X.; Mu, X.; Hou, T.; Xu, J.; Di Renzo, M.; Al-Dhahir, N. Reconfigurable intelligent surfaces: Principles and opportunities. *IEEE Commun. Surv. Tutor.* **2021**, *23*, 1546–1577. [\[CrossRef\]](#)
18. Taha, A.; Zhang, Y.; Mismar, F.B.; Alkhateeb, A. Deep reinforcement learning for intelligent reflecting surfaces: Towards standalone operation. In Proceedings of the 2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC), Atlanta, GA, USA, 26–29 May 2020; IEEE: New York, NY, USA, 2020; pp. 1–5.
19. Taha, A.; Alrabeiah, M.; Alkhateeb, A. Enabling large intelligent surfaces with compressive sensing and deep learning. *IEEE Access* **2021**, *9*, 44304–44321. [\[CrossRef\]](#)
20. Zhang, Q.; Saad, W.; Bennis, M. Millimeter wave communications with an intelligent reflector: Performance optimization and distributional reinforcement learning. *IEEE Trans. Wirel. Commun.* **2021**, *21*, 1836–1850. [\[CrossRef\]](#)
21. Huang, C.; Mo, R.; Yuen, C. Reconfigurable intelligent surface assisted multiuser MISO systems exploiting deep reinforcement learning. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 1839–1850. [\[CrossRef\]](#)
22. Yang, H.; Xiong, Z.; Zhao, J.; Niyato, D.; Xiao, L.; Wu, Q. Deep reinforcement learning-based intelligent reflecting surface for secure wireless communications. *IEEE Trans. Wirel. Commun.* **2020**, *20*, 375–388. [\[CrossRef\]](#)
23. Liu, X.; Liu, Y.; Chen, Y.; Poor, H.V. RIS enhanced massive non-orthogonal multiple access networks: Deployment and passive beamforming design. *IEEE J. Sel. Areas Commun.* **2020**, *39*, 1057–1071. [\[CrossRef\]](#)
24. Fathy, M.; Fei, Z.; Guo, J.; Abood, M.S. Machine-Learning-Based Optimization for Multiple-IRS-Aided Communication System. *Electronics* **2023**, *12*, 1703. [\[CrossRef\]](#)
25. Fathy, M.; Abood, M.S.; Guo, J. A Generalized Neural Network-based Optimization for Multiple IRSs-aided Communication System. In Proceedings of the 2021 IEEE 21st International Conference on Communication Technology (ICCT), Tianjin, China, 13–16 October 2021; IEEE: New York, NY, USA, 2021; pp. 480–486.
26. Ahsan, W.; Yi, W.; Qin, Z.; Liu, Y.; Nallanathan, A. Resource allocation in uplink NOMA-IoT networks: A reinforcement-learning approach. *IEEE Trans. Wirel. Commun.* **2021**, *20*, 5083–5098. [\[CrossRef\]](#)
27. Zhang, Y.; Mou, Z.; Gao, F.; Jiang, J.; Ding, R.; Han, Z. UAV-enabled secure communications by multi-agent deep reinforcement learning. *IEEE Trans. Veh. Technol.* **2020**, *69*, 11599–11611. [\[CrossRef\]](#)
28. Zhong, R.; Liu, X.; Liu, Y.; Chen, Y. Multi-agent reinforcement learning in NOMA-aided UAV networks for cellular offloading. *IEEE Trans. Wirel. Commun.* **2021**, *21*, 1498–1512. [\[CrossRef\]](#)
29. Omidshafiei, S.; Pazis, J.; Amato, C.; How, J.P.; Vian, J. Deep decentralized multi-task multi-agent reinforcement learning under partial observability. In Proceedings of the International Conference on Machine Learning, Sydney, Australia, 6–11 August 2017; PMLR: Baltimore, MA, USA, 2017; pp. 2681–2690.
30. Nasir, Y.S.; Guo, D. Multi-agent deep reinforcement learning for dynamic power allocation in wireless networks. *IEEE J. Sel. Areas Commun.* **2019**, *37*, 2239–2250. [\[CrossRef\]](#)

31. Chen, J.; Guo, L.; Jia, J.; Shang, J.; Wang, X. Resource allocation for IRS assisted SGF NOMA transmission: A MADRL approach. *IEEE J. Sel. Areas Commun.* **2022**, *40*, 1302–1316. [[CrossRef](#)]
32. Huang, C.; Chen, G.; Wong, K.K. Multi-agent reinforcement learning-based buffer-aided relay selection in IRS-assisted secure cooperative networks. *IEEE Trans. Inf. Forensics Secur.* **2021**, *16*, 4101–4112. [[CrossRef](#)]
33. Pan, C.; Ren, H.; Wang, K.; El Kashlan, M.; Nallanathan, A.; Wang, J.; Hanzo, L. Intelligent reflecting surface aided MIMO broadcasting for simultaneous wireless information and power transfer. *IEEE J. Sel. Areas Commun.* **2020**, *38*, 1719–1734. [[CrossRef](#)]
34. Jiang, T.; Cheng, H.V.; Yu, W. Learning to reflect and to beamform for intelligent reflecting surface with implicit channel estimation. *IEEE J. Sel. Areas Commun.* **2021**, *39*, 1931–1945. [[CrossRef](#)]
35. Grant, M.; Boyd, S. CVX: Matlab Software for Disciplined Convex Programming, Version 2.2. 2020. Available online: <http://cvxr.com/cvx/> (accessed on 22 January 2024).
36. 3GPP TR 38.901. Study on Channel Model for Frequencies from 0.5 to 100 GHz. 2017. Available online: <https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3173> (accessed on 20 March 2023).
37. Sutton, R.S.; Barto, A.G. *Introduction to Reinforcement Learning*; BooksRun: Philadelphia, PA, USA, 2018.
38. Sur, S.N.; Singh, A.K.; Kandar, D.; Silva, A.; Nguyen, N.D. Intelligent Reflecting Surface Assisted Localization: Opportunities and Challenges. *Electronics* **2022**, *11*, 1411. [[CrossRef](#)]
39. Melo, F.S. *Convergence of Q-Learning: A Simple Proof*; Technical Report; Institute of Systems and Robotics. 2001; pp. 1–14. Available online: <https://www.academia.edu/download/55970511/ProofQlearning.pdf> (accessed on 20 March 2023).
40. Timofte, V.; Timofte, A.; Khan, L.A. Stone–Weierstrass and extension theorems in the nonlocally convex case. *J. Math. Anal. Appl.* **2018**, *462*, 1536–1554. [[CrossRef](#)]
41. Yang, Z.; Liu, Y.; Chen, Y.; Al-Dhahir, N. Machine learning for user partitioning and phase shifters design in RIS-aided NOMA networks. *IEEE Trans. Commun.* **2021**, *69*, 7414–7428. [[CrossRef](#)]
42. Burkard, R.; Dell’Amico, M.; Martello, S. *Assignment Problems: Revised Reprint*; SIAM: Philadelphia, PA, USA, 2012.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.