Distributed DRL-Based Downlink Power Allocation for Hybrid RF/VLC Networks

Bekir Sait Ciftler, Abdulmalik Alwarafy, and Mohamed Abdallah

***Abstract*—Hybrid radio frequency (RF) and visible light communication (VLC) networks can provide high throughput and energy efficiency with VLC access points (APs) while ensuring ubiquitous coverage with RF APs. Due to dynamic channel conditions and limited resources, the hybrid RF/VLC networks' resource allocation problem is complex and challenging. Conventional resource allocation techniques fail to overcome these challenges. Heuristic methods can solve high complexity problems; however, they are not robust against changes such as dynamic channel conditions or alternating user requirements. Heuristic methods require centralized control for stability which adds communication overhead between APs. Deep Reinforcement Learning (DRL) based solutions can solve high complexity, dynamic channel conditions, and alternating user requirements while not requiring centralized control. In this paper, we formulate a distributed downlink power allocation problem to optimize the transmit power for users to reach target data rates in hybrid RF/VLC networks. Then, we propose a distributed DRL-based algorithm Deep Deterministic Policy Gradient (DDPG), to solve the formulated computationally-intensive problem. We implement a simulation environment to benchmark the proposed distributed DRL-based method against other methods such as Q-Learning (QL) and Deep Q-Networks (DQN), and centralized heuristic power allocation algorithms. Our simulation results show that the distributed DDPG-based algorithm learns to adapt against changes in the channel or user requirements, while centralized Genetic Algorithm and Particle Swarm Optimization-based algorithms fail to endure against these changes even with coordination between APs. Additionally, we quantify the performance of the DDPG-based algorithm to prevail amid DRL-based algorithms at the expense of higher implementation complexity.**

***Index Terms*—DRL, heuristic algorithms, hybrid RF/VLC networks, resource allocation.**

I. Introduction

THE spectrum scarcity problem is becoming more critical for communication networks as the variety and the quantity of devices increase rapidly. In recent years, Visible Light Communication (VLC) gained prominence thanks to its ubiquitous capacity to yield high data rates with utilization of energy-efficient light-emitting diodes (LEDs) [2] and solve spectrum scarcity by exploiting visible spectrum [3]. However, the VLC coverage is limited because it requires line-of-sight between the transmitter and the receiver [4]. To overcome this challenge, VLC is used in conjunction with Radio Frequency (RF) technologies in hybrid RF/VLC networks to provide ubiquitous connectivity. One of the challenges of hybrid RF/VLC networks is the allocation of limited radio resources while providing users' Quality-of-Service (QoS) requirements such as target data rate [5].

Conventional power allocation methods for hybrid RF/VLC networks require perfect knowledge of the channel state information (CSI) and coordination between access points (APs) and users. Additionally, limited resources and users with QoS requirements make the power allocation problem nonconvex and highly complex [6]. Thus, the solution becomes intractable with the increasing number of users. There are heuristic optimization techniques, which can solve these complex problems, such as Genetic Algorithm (GA) [7] and Particle Swarm Optimization (PSO) [8]. However, these techniques are not robust against changes in the system such as dynamic channel conditions [9] and alternating user requirements, and they require frequent reiterations.

Reinforcement learning (RL) is a subarea of machine learning (ML), where agents learn by experience. The agents take actions based on the environment's state to maximize cumulative reward over time. These agents learn iteratively by trial and error using reward as a feedback mechanism. RL can solve problems that conventional or heuristic methods cannot solve due to limited knowledge of the environment dynamics or high complexity of the system [10], [11].

Q-Learning (QL) is a well-known RL algorithm widely used in the optimization of wireless networks [12], [13]. However, the QL's discrete state and action spaces make it incapable of solving the optimization problem for higher dimensions, i.e., a large number of users, and obtain an optimal QL strategy. To overcome this problem, a combination of Deep Learning (DL) and RL, where DL is the RL agent's main structure, is called Deep Reinforcement Learning (DRL). DRL-based solutions can be applied to problems with discrete and continuous state and action spaces along with multivariable optimization problems. The Deep Q-Networks (DQN) can interpret observations of the environment with better performance with the help of neural networks [14], [15]. DQN's continuous state space allows it to understand higher-dimension complex problems. However, its discrete action space with a finite set of transmit power levels

Manuscript received October 12, 2021; revised December 16, 2021; accepted December 25, 2021. Date of publication December 31, 2021; date of current version June 7, 2022. This work was supported by NPRP-Standard 13th Cycle under Grant # NPRP13S-0201-200219 from Qatar National Research Fund (a Member of Qatar Foundation). This work's extended version was published in the IEEE International Conference on Communications (ICC) 2021 [DOI: 10.1109/ICC42927.2021.9500564]. *(Corresponding author: Bekir Sait Ciftler.)*

The authors are with the Division of Information Computing Technology, College of Science Engineering, Hamad Bin Khalifa University, Doha, Qatar (e-mail: bsciftler@gmail.com; aalwarafy@hbku.edu.qa; moabdallah@hbku.edu.qa).

Digital Object Identifier 10.1109/JPHOT.2021.3139678

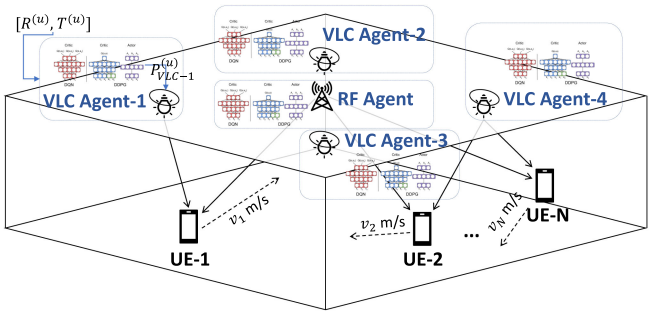

Fig. 1. Distributed Multiagent DRL-based power allocation for hybrid RF/VLC networks.

hinders its ability to find the true optimal policy in the solution space [16]. Recently, Deep Deterministic Policy Gradient (DDPG) is being considered for solving highly complex problems in wireless networks [17]–[19]. The DDPG has continuous action space where agents can allocate the transmit power with continuous variables. Thus it avoids the transmit power quantization pitfall and can obtain true optimal policy.

In this paper, we design and implement a novel distributed DDPG-based algorithm for the problem of power allocation for hybrid RF/VLC networks to provide multiple users' QoS requirements regarding downlink data rates. We assume users have multihoming capability, where they connect and receive data from an RF AP and a VLC AP simultaneously, as shown in Fig. 1. We implement three distributed RL-based algorithms, QL, DQN, and DDPG, to solve the multiagent optimization problem where each AP is an agent. There is no coordination between APs regarding their actions, i.e., they do not communicate their power allocations. We have defined the reward function and the penalty function for algorithms based on the actual data rate and the target data rate of users based on their QoS requirements. To our best knowledge, there is no paper considering a distributed DDPG-based algorithm to solve power allocation for hybrid RF/VLC networks that can cope with a large number of users with mobility and dynamic channel conditions. In [1], we had applied DQN-based algorithm to solve power allocation optimization problem with two static users with static QoS requirements. In this work, we extend our system model with more than two users, and varying QoS requirements. Additionally, the users have mobility with variable speeds. The contributions of this work beyond the conference version [1] and other literature are summarized as follows:

- We propose, design, and implement a *distributed* DDPG-based algorithm, to solve the power allocation optimization problem for a multiuser hybrid RF/VLC network with limited radio resources and dynamic channel conditions.
- We design proper *reward* and *penalty* functions for faster convergence and improved multiagent DDPG-based power allocation stability.
- We investigate the proposed algorithm's performance under dynamic channel conditions and *alternating QoS (i.e., target data rates)* for users with *mobility*.
- We benchmark performance of our proposed distributed DDPG-based algorithm amongst other distributed DRL algorithms, QL and DQN and against centralized heuristic methods, GA and PSO.
- We show that the DDPG-based algorithm has the best performance in terms of QoS and convergence performance compared to other DRL-based algorithms and heuristic algorithms.

This paper is structured as follows. In Section II, a brief overview of existing literature on resource allocation is provided, including conventional methods and learning-based techniques. We provide the system model for the hybrid RF/VLC network in Section III. We formulate the power allocation problem in IV Subsequently, we explain the DRL-based multiuser power allocation algorithms and our proposed DDPG-based method in Section V. Numerical results for the provided methodologies are given in Section VI. We present our concluding remarks in Section VII.

II. Related Work

In this section, we provide a brief survey of resource allocation for hybrid RF/VLC networks. Resource allocation has been one of the critical aspects of these networks to mitigate interference, increase throughput, lower latency, and provide a better QoS for overall performance [20]. Below, we provide a summary of the literature on various resource allocation techniques for hybrid RF/VLC networks.

Conventional resource allocation mechanisms are based on simple uniform allocation [21] or optimization techniques use numerical methods [22]. The system is modeled in closed-form equations in conventional techniques. These system models lead to computationally intensive optimization problems resulting in intractable solutions with a large number of users and their QoS requirement constraints. The conventional techniques for improving the performance of VLC networks are summarized in [6].

In [22], the authors propose a joint load balancing and power allocation problem to maximize the system capacity while keeping the fairness between users. They proposed two methods to solve this problem. The first approach assumes full CSI and interference information and is highly complex, while the second proposed algorithm assumes partial interference information and reaches a suboptimal solution. Later, an iterative power allocation algorithm by averaging the interference over links provides faster convergence and better performance. In another work, frequency reuse patterns and cell formation is studied for hybrid RF/VLC networks in the context of load balancing in [3]. The optimization problem is stated as a mixed-integer nonlinear programming (MINLP) model. The problem model is approximated to a discrete linear programming optimization for simplification. The model grow into an intractable problem as the number of users increases. In [23], the authors propose integrating the existing RF network to a hybrid PLC/VLC system to optimize power consumption while achieving the required QoS. The problem is formulated as a transmission power minimization of the hybrid system. It is solved as a convex optimization problem, a function of transmit powers under QoS requirements.

Heuristic methods are proposed for handling complexity when the optimization problem is nonconvex. In [7], the authors propose GA to solve subcarrier and power allocation for VLC systems. The problem is defined as a nonconvex system and defined as a log utility function to improve user fairness. The increase in the fairness and the throughput of users are significant. The proposed algorithm has shown better performance than round-robin, best channel quality information, and GA-based methods. The authors propose a PSO-based unified resource allocation and mobility management algorithm for indoor VLC networks in [8]. They assume a centralized controller with the knowledge of CSI and QoS requirements of users and AP status.

RL-based methods have recently gained attention due to the promises of DRL algorithms in control mechanisms [24] such as high complexity and adaptability to the dynamic environment and alternating constraints. The authors in [25] present a comprehensive survey of the DRL-based approaches for radio resources allocation and management in future heterogeneous wireless networks. A QL-based power allocation algorithm for hybrid RF/VLC systems is proposed in [26]. Multihoming users are considered in hybrid RF/VLC networks, where RF channel has more variations than VLC channels due to small-scale fading. The simulation results confirm that their proposed methodology supports up to two users while it cannot cope up with increasing number of users. In [13], the authors propose a hybrid WiFi-VLC system with RL implemented at WiFi AP for offloading users from one AP to another. The numerical results show that the total system throughput is increased significantly. Fairness between users is also improved with the proposed algorithm compared to the conventional method. In [14], the authors propose a centralized DQN-based post-decision state experience replay and transfer resource allocation scheme for hybrid RF/VLC networks. In this work, the authors consider both a joint uplink and downlink resource management problem. The experience replay and transfer model allowed the system to converge earlier and provide better performance by utilizing each others' experiences. Another DQN-based resource allocation scheme is proposed in [15] for hybrid RF/VLC networks. The numerical results show that the sum rate is 10% higher for the proposed DQN-based method, and the number of iterations is 54% less compared to the conventional methods for convergence [15]. However, in both works [14], [15], the use of the DQN-based causes to lose precision due to discrete action space and the DQN-based algorithm is designed as a centralized controller, which brings additional complexity and communication overhead.

To our best knowledge, there is no other work proposing to use the distributed DDPG-based power allocation algorithm for hybrid RF/VLC networks. In our work, we propose a distributed DDPG-based algorithm to utilize its continuous state and action spaces, which shows superiority to the other DRL-based algorithms and heuristic approaches such as GA and PSO.

III. System Model

In the rest of the paper, we consider a hybrid RF/VLC network with multiple users as shown in Fig. 1. The system model consists of N mobile users with multihoming capability uniformly distributed within the room, only one RF AP and four VLC APs. The users are moving according to a random waypoint model with a speed uniformly distributed between 0 and 1 (m/s). A random destination is determined for each user, and once a user reaches its destination, the user determines a new random location as its destination.

The link channel gain between uth user and kth VLC AP at iteration i can be represented as [1]:

$$G_{\mathrm{V}}^{(u,k)}(i) = \frac{(m+1)A_{pd}\lambda \cos^{m}(\theta_{tx}^{(u,k)}(i))}{2\pi\left(\left(x^{(u,k)}(i)\right)^2 + y^2\right)} \times H_f(\theta_{rx}^{(u,k)}(i)) H_c(\theta_{rx}^{(u,k)}(i)) \cos(\theta_{rx}^{(u,k)}(i)), \tag{1}$$

where x and y are the distances in the horizontal and the vertical axes between the kth AP and the uth user device. The effective detection area of PD is A_{pd}, responsivity of the PD is λ. θ_{tx} and θ_{rx} represents angle of irradiance and angle of incidence, respectively [1]. The PDs of all users are oriented towards ceiling (i.e. $\theta_{tx}^{(u,k)} = \theta_{rx}^{(u,k)}$). The optical filter gain is $H_f(\theta_{rx}^{(u,k)}(i))$ and the optical concentrator gain is $H_c(\theta_{rx}^{(u,k)}(i))$. Additionally, $m = -1/\log_2(\cos(\Psi_{1/2}))$, where $\Psi_{1/2}$ is the LED half-power semi-angle. The optical filter gain assumed to be 1. The optical concentrator gain is defined as:

$$H_c(\theta_{rx}^{(u,k)}(i)) = \frac{n_c^2}{\sin^2(\Psi_{\mathrm{fov}})} \mathbb{1}\left(0 \le \theta_{rx}^{(u,k)}(i) \le \Psi_{\mathrm{fov}}\right), \tag{2}$$

where n_c is the optical concentrator reflective index, Ψ_{fov} is the PD half Field-of-View (FoV), $\mathbb{1}(\cdot)$ is binary indicator [1].

The VLC AP coverage is exclusive with the assumption of orthogonal frequencies allocation for bandwidth. The total bandwidth of each AP is equally allocated to each user within its coverage range. The VLC AP determines transmit power for each user at the beginning of each iteration i. The achievable rate of the link between the user u and the VLC AP k in iteration i is defined as [2], [27]:

$$C_{\mathrm{V}}^{(u,k)}(i) = \frac{W_{\mathrm{V}}}{2}\log_2\left(1 + \frac{\left(\kappa m_d P_{\mathrm{V}}^{(u,k)}(i) G_{\mathrm{V}}^{(u,k)}(i)\right)^2}{W_{\mathrm{V}}\sigma_{\mathrm{V}}^2}\right), \tag{3}$$

where W_{V} is the VLC link bandwidth, κ is the efficiency of optical to electric conversion, m_d is the modulation depth. σ_{V}^2 defined as the VLC link's noise power spectral density (PSD). $P_{\mathrm{V}}^{(u,k)}(i)$ is the transmit power of kth VLC AP for uth user in iteration i[1].

The gain of the RF link in iteration i is defined as below:

$$G_{\mathrm{R}}^{(u)}(i) = 10^{-L(d^{(u)}(i))/10}|h_{\mathrm{R}}^{(u)}(i)|^2, \tag{4}$$

where the small-scale fading is represented with $h_{\mathrm{R}}^{u}(i)$ and modeled as an exponential random variable with 2.46 dB mean. The small-scale fading is randomly generated at every iteration i, hence the channel changes every iteration. The path loss component $L(d^{(u)})$ is defined as in the below equation:

$$L(d^{(u)}) = 47.9 + 10\nu\log_{10}(d^{(u)}/d_0) + X \text{ (dB)}, \tag{5}$$

where $d^{(u)}$ is the distance between the RF AP and the uth user, ν is the path loss exponent, d_0 is the reference distance, and X is shadowing component defined as a Gaussian random variable with zero mean and variance of 1.8 dB.

The achievable rate at the user u from the RF link in iteration i is given as:

$$C_{\mathrm{R}}^{(u)}(i) = W_{\mathrm{R}} \log_2 \left(1 + \frac{P_{\mathrm{R}}^{(u)}(i) G_{\mathrm{R}}^{(u)}(i)}{W_{\mathrm{R}} \sigma_{\mathrm{R}}^2}\right), \tag{6}$$

where the RF link bandwidth is W_{R}, and $P_{\mathrm{R}}^{u}(i)$ is the transmit power for the user u at RF AP in iteration i. The additive white Gaussian noise (AWGN) for RF links is represented with PSD value of σ_{R}^2. The users can use the RF and VLC links simultaneously thanks to their multihoming capability. The data rate capacity for the user u at iteration i becomes the summation of both link capacities:

$$C^{(u)}(i) = C_{\mathrm{R}}^{(u)}(i) + C_{\mathrm{V}}^{(u,k)}(i), \tag{7}$$

where the associated VLC AP is k at iteration i.

IV. Problem Formulation

In this section, we formulate a distributed transmit power allocation problem each for the given hybrid RF/VLC network in Fig. 1 to minimize the difference between the target and the actual data rate for all users. The optimization problem for adjusting the transmit powers accordingly defined as in the following:

$$\max_{\{P_{\mathrm{R}}^{(u)}(i)\},\{P_{\mathrm{V}}^{(u,k)}(i)\}} \sum_{t=1}^{\infty} \sum_{u=1}^{N} -|C^{(u)}(i) - T^{(u)}| \tag{8}$$

$$\text{s.t.} \quad \sum_{u=1}^{N} P_{\mathrm{R}}^{(u)}(i) \leq P_{\mathrm{R}}^{\max}, \tag{9}$$

$$\sum_{u=1}^{N} P_{\mathrm{V}}^{(u,k)}(i) \leq P_{\mathrm{V}}^{\max}, \ \ \forall l, \tag{10}$$

$$P_{\mathrm{V}}^{(u,k)} \geq 0, \ P_{\mathrm{R}}^{(u)} \geq 0, \ \ \forall u, l, \tag{11}$$

$$|C^{(u)}(i) - T^{(u)}(i)| \leq \delta^{(u)}, \ \ \forall u, \tag{12}$$

where $T^{(u)}(i)$ is the target data rate requirement of user u for the iteration i, and $\delta^{(u)}$ is defined as the acceptable data rate difference for user u. In (9) and (10), the power allocation is limited by total power available at each RF or VLC AP, respectively. The transmit powers of APs cannot be below zero as defined in (11). In (12), the acceptable difference between the target data rate of the user and the total achievable rate for the user u is given as the final constraint on the optimization problem. In our simulations, we defined the $\delta^{(u)}$ as the 5% of the target data rate. The total achievable rate ($C^{(u)}(i)$) should be within proximity of 5% of the target data rate ($T^{(u)}(i)$).

The proposed optimization problem (8) is nonconvex as a large number of users with QoS requirements and limited resources in the system. Conventional algorithms cannot solve this optimization problem due to complexity [7], [8], [15]. As we mentioned in Section II, there are some methods such as GA [7] or PSO [8] which can handle complexity; however, these methods are not robust against dynamic systems and require frequent re-iteration for each change in the system. We propose RL-based algorithms to learn the policy to solve the optimization problem with better convergence performance and faster adaptability. We benchmark the proposed method against other algorithms proposed in [7], and PSO as in [8]. In the following section, we explain details of the RL architecture used for the solution.

V. RL-Based Multiagent Power Allocation

In this section, we propose a distributed DDPG-based power allocation solution to control the transmit powers at each AP to achieve the target data rate for each user. We compare our results against other distributed RL-based algorithms (i.e., QL and DQN [1]) and centralized heuristic methods (i.e., GA [7] and PSO [8]). In DRL-based algorithms, separate agents do not communicate regarding their actions, but they gather the achieved rate at the user for the previous iteration. Each agent (i.e., each AP) has only the knowledge of the target and the estimated total rate capacity of the users. The distributed DDPG-based agents have continuous state-space to utilize the difference of the estimated total capacity and the target data rate of users. The difference of estimated total capacity and the target rate of the user provides necessary state information to DDPG agents to apply the policy.

The state space of DDPG agents is defined as follows:

$$\mathbf{s}_t = [s_t^{(1)}, \ \ldots, \ s_t^{(u)}, \ \ldots, \ s_t^{(N)}], \tag{13}$$

where $s_t^{(u)}$ consists the information for user u as follows:

$$s_t^{(u)} = [C^{(u)}(i-1), \ T^{(u)}]^T. \tag{14}$$

The difference between the total capacity of the user and the target data rate, hence the state-space of the DDPG agents is affected by the channel uncertainty due to the noise on the VLC and the RF channels, and the small-scale fading as in (4) and the shadowing in (5). However, the distributed DDPG agents learn to cope with the dynamic and uncertain channel conditions and user mobility.

The action space is defined by a noisy policy function with Ornstein-Uhlenbeck (OU) process as follows:

$$\begin{aligned} \mathbf{a}_t &= \mu(\mathbf{s}_t|\theta^{\mu}) + \mathbf{n}_t \\ &= [a_t^{(1)}, \ldots, a_t^{(u)}, \ldots, a_t^{(U)}] \end{aligned} \tag{15}$$

where the policy function of the agent is represented with μ, and $a_t^{(u)}$ is a scalar value representing the power allocation of user u at that AP. θ^{μ} is the vector for the weights of neural network, and $\mathbf{n}$ is the Ornstein-Uhlenbeck (OU) process-based action noise. The continuous action space of the DDPG allows it to be more precise in contrast to discrete action space of QL and DQN [1]. The exploration vs. exploitation trade-off for the agent is handled by the OU process. The OU process uses a correlated normal distribution [28] for sampling the noise. Each agent's action (RF and VLC APs) is the vector of transmit power

for each user in the system. If the user is not in the line-of-sight for VLC AP, transmit power for that user is equal to zero. For others, the output of the actor-network is used. All the users are connected to RF AP, and its actor network's output is used as its action.

The objective of the problem defined in (8) is minimizing every user's target and actual data rate difference given limited resources. The reward function based on optimization problem defined in (8) as follows:

$$r_t = \sum_{u=1}^{N} \delta^{(u)} - |C^{(u)}(t) - T^{(u)}| - \varrho^{(u)}(t), \tag{16}$$

where $\varrho^{(u)}(t)$ is the penalty for not satisfying the target data rate for user u. This penalty function is an additional measure to avoid leaving individual users unsatisfied while achieving a positive reward. The penalty function is defined as follows:

$$\varrho^{(u)}(t) = \begin{cases} 0, & \text{if } C^{(u)}(t) \geq T^{(u)} - \delta^{(u)} \\ \Gamma, & \text{if } C^{(u)}(t) < T^{(u)} - \delta^{(u)} \end{cases}, \tag{17}$$

where Γ is the constant value of penalty for agents.

We explain the multiagent DDPG-based power allocation in Algorithm 1. We initialize the empty replay buffer D of each agent in step 1. The capacity of each buffer is M state-action-reward-next state transactions. The actor and critic network weights are initialized in step 2 and 3. The target networks of actor and critic networks are initialized with the same weights in step 4 and 5. From step 6 to 25, represents each iteration (i.e., iteration i). In each iteration, we observe the state $\mathbf{s}$ for agents, select an action (i.e., transmit power values) with exploration noise based on the OU process in steps 8 and 10. The agents (i.e., APs) determine their actions and they are executed simultaneously. The shared reward r_t defined in (16) is received at every AP. The new state for the environment $\mathbf{s}_{t+1}$ is observed and the transactions are stored in replay memories of every AP in steps 11 and 12. The sampled random mini-batch of transitions at each agent are used in the Bellman equation. The targets for actors and critic networks' are determined in steps 15 and 21. The weights of the critic network are adjusted by minimizing the loss using targets in steps 16 and 22. The weights of the actor network are updated with the policy gradient samples in steps 17 and 23. Each agent's target networks are updated according to the update rate (τ) for stability in steps 18 and 24. This iteration is repeated until convergence is reached. In our simulations, we defined convergence as having a constant positive reward for at least 100 consecutive iterations as an early stopping mechanism [29].

Algorithm 1: DDPG-Based Power Allocation.

1: **Initialization:** Set $t = 0$ and initialize replay buffers of VLC AP agents $\mathcal{D}_{\mathrm{V}}^{(k)}$ and RF AP agent $\mathcal{D}_{\mathrm{R}}$, with capacity M.
2: Randomly initialize the weights of actor networks θ_k^{μ} and critic networks θ_k^{Q} for VLC APs.
3: Randomly initialize the weights of actor network $\theta_{\mathrm{R}}^{\mu}$ and critic network θ_{R}^{Q} for RF AP.
4: Initialize VLC target networks: $\theta_k^{\mu'} \leftarrow \theta_k^{\mu}$ and $\theta_k^{Q'} \leftarrow \theta_k^{Q}$.
5: Initialize RF target networks: $\theta_{\mathrm{R}}^{\mu'} \leftarrow \theta_{\mathrm{R}}^{\mu}$ and $\theta_{\mathrm{R}}^{Q'} \leftarrow \theta_{\mathrm{R}}^{Q}$.
6: **for** $t = 1$ to ∞ **do**
7: **for** $k = 1$ to K **do**
8: Observe state $\mathbf{s}_t^{(k)}$ (data rates of users) and determine an action (transmit power) for VLC AP k $\mathbf{a}_t^{(k)} = \mu(\mathbf{s}_t^{(k)}|\theta^{\mu}) + \mathbf{n}_t^{(k)}$
9: **end for**
10: Observe state $\mathbf{s}_t$ and determine an action with exploration noise for RF AP $\mathbf{a}_t^{RF} = \mu(\mathbf{s}_t^{RF}|\theta_{\mathrm{R}}^{\mu}) + \mathbf{n}_t^{RF}$
11: Execute all actions $\mathbf{a}_t^{(k)}$ at all VLC APs and $\mathbf{a}_t^{RF}$ at RF AP.
12: Receive the reward r_t, and observe next state s_{t+1}, store transition $(\mathbf{s}_t, \mathbf{a}_t^{(k)}, r^t, \mathbf{s}_{t+1})$ in $D_{\mathrm{V}}^{(k)}$ for VLC APs and $(\mathbf{s}_t, \mathbf{a}_t^{RF}, r^t, \mathbf{s}_{t+1})$ in D_{R} for RF AP.
13: **for** $k = 1$ to K **do**
14: Randomly sample mini-batch transitions from $\mathcal{D}_{\mathrm{V}}^{(k)}$: $B^l = \{(\mathbf{s}_i, \mathbf{a}_i^{(k)}, r_i, \mathbf{s}_{i+1})\}$.
15: Compute the targets for actor and critic networks: $\tilde{Q}_k(\mathbf{s}_i, \mathbf{a}_i^{(k)}|\theta_k^{Q}) = r_i + \gamma Q_k(\mathbf{s}_{i+1}, \mu(\mathbf{s}_i|\theta_k^{\mu'})|\theta_k^{Q'})$
16: Update the θ_k^{Q} in critic network by minimizing the loss: $L = \frac{1}{|B^{(k)}|}\sum_{i=1}^{|B^{(k)}|}(\tilde{Q}_k(\mathbf{s}_i, \mathbf{a}_i^{(k)}|\theta_k^{Q'}) - Q_k(\mathbf{s}_i, \mathbf{a}_i^{(k)}|\theta_k^{Q}))^2$
17: Update the θ^{μ} in actor network according to the sampled policy gradient: $\nabla_{\theta^{\mu}}\boldsymbol{J} \approx \frac{1}{|B^{(k)}|}\sum_{i=1}^{|B^{(k)}|}\nabla_a Q_k(\mathbf{s}_i^{(k)}, \mathbf{a}_i^{(k)}|\theta_k^{Q})\nabla_{\theta^{\mu}}\mu(\mathbf{s}_i^{(k)}|\theta_k^{\mu})$
18: Update the VLC target networks: $\theta_k^{Q'} \leftarrow \tau\theta_k^{Q} + (1-\tau)\theta_k^{Q'}$ $\theta_k^{\mu'} \leftarrow \tau\theta_k^{\mu} + (1-\tau)\theta_k^{\mu'}$
19: **end for**
20: Randomly sample mini-batch transitions from $\mathcal{D}_{\mathrm{R}}$: $B^{RF} = \{(\mathbf{s}_i, \mathbf{a}_i^{RF}, r_i, \mathbf{s}_{i+1})\}$.
21: Compute the targets: $\tilde{Q_{\mathrm{R}}}(\mathbf{s}_i, \mathbf{a}_i^{RF}|\theta_{\mathrm{R}}^{Q}) = r_i + \gamma Q_{\mathrm{R}}(\mathbf{s}_{i+1}, \mu(\mathbf{s}_i|\theta_{\mathrm{R}}^{\mu'})|\theta_{\mathrm{R}}^{Q'})$
22: Update the θ_{R}^{Q} in critic network by minimizing the loss: $L = \frac{1}{|B^{RF}|}\sum_{i=1}^{|B^{RF}|}(\tilde{Q_{\mathrm{R}}}(\mathbf{s}_i, \mathbf{a}_i^{RF}|\theta_{\mathrm{R}}^{Q'}) - Q_{\mathrm{R}}(\mathbf{s}_i, \mathbf{a}_i^{RF}|\theta_{\mathrm{R}}^{Q}))^2$
23: Update the $\theta_{\mathrm{R}}^{\mu}$ in actor network according to the sampled policy gradient: $\nabla_{\theta_{\mu}}\boldsymbol{J} \approx \frac{1}{|B^{RF}|}\sum_{i=1}^{|B^{RF}|}\nabla_a Q_{\mathrm{R}}(\mathbf{s}_i, \mathbf{a}_i^{RF}|\theta_{\mathrm{R}}^{Q})\nabla_{\theta_{\mu}}\mu(\mathbf{s}_i|\theta_{\mathrm{R}}^{\mu})$
24: Update the RF target networks: $\theta_{\mathrm{R}}^{Q'} \leftarrow \tau\theta_{\mathrm{R}}^{Q} + (1-\tau)\theta_{\mathrm{R}}^{Q'}$ $\theta_{\mathrm{R}}^{\mu'} \leftarrow \tau\theta_{\mathrm{R}}^{\mu} + (1-\tau)\theta_{\mathrm{R}}^{\mu'}$
25: **end for**

A. Complexity Discussion

In this subsection, we discuss the complexity of the proposed algorithms. The Q-table of each agent consists of 3^N rows representing environment state considering the table structure [1] and $|\mathcal{P}|^N$ (i.e., 31^N in our simulations) columns representing agent actions. The number of entries in the Q-Table is $3^N \times |\mathcal{P}|^N$, which shows that the system grows exponentially with an increasing number of users.

The neural network structure of DQN agents includes a single neural network with 2 hidden layers and $2N$ hidden nodes in each layer. The input of the DQN is the state of the environment, and its size is $2N$. However, $|\mathcal{P}|^N$ distinct possible agent actions, where N is the number of users. There are 31^N possible action

TABLE I
SIMULATION PARAMETERS

Parameter	Value
P_{R}^{max}	0.1 W
σ_{R}^2	-57 dBm/MHz
W_{R}	5 MHz
P_{V}^{max}	2 W
σ_{V}^2	-100 dBm/MHz
W_{V}	20 MHz
Ψ_{fov}	45°
$\Psi_{1/2}$	60°
A_{pd}	10^{-4}
λ	0.4
H_f	1
n_c	1.5
κ	1
ν	1.6
d_0	1 Meter
$\lvert\mathcal{P}\rvert$	31
N	2 to 15
Γ	100
γ	0.9
τ	0.05

combinations in our simulations [1]. The DQN agent uses softmax to select the best possible action as the output of its neural network. Although the agents are improved in understanding the state of the environment with its continuous form, the action space limits scalability due to exponentially increasing possible discrete action combinations with an increasing number of users.

There are 4 neural networks at each DDPG agent (i.e., APs), including actor and critic networks and target networks for increased stability. Each actor and critic network consist of 2 hidden layers, with $2N$ hidden nodes in each layer. The input of actor networks is size $2N$, and the output is size N (i.e., a transmit power value for each user). The DDPG-based algorithm allows having larger number of users to a more significant extent with linearly increasing complexity. Although the implementation of DDPG is more complex compared to QL and DQN, its convergence performance is better as the number of users (N) increases in the system since its network size grows linearly while QL and DQN size grow exponentially.

B. Simulation Parameters and Stability Discussion

The multiagent scheme caused stability problems from time to time. However, with hyperparameter optimization of DRL-based algorithms, these issues were resolved for all of the algorithms. The parameters for the simulation are provided in Table I. The DDPG-based algorithm is the most sensitive algorithm concerning parameters. However, setting a lower learning rate and update rate and use of target networks increased the stability of DDPG significantly. The initialization of each algorithm plays a significant role as well. The QL-based algorithm uses Q-table values, while DQN and DDPG are the approximations of Q-function, all of them depend on initial Q-values of actions. In QL-based and DQN-based algorithms, higher values in the initialization increase the exploration, while it causes DDPG to show asymptotic behavior. We initialize DDPG neural networks with small weights, uniformly distributed between -3×10^{-4} to 3×10^{-4} to avoid asymptotic behavior.

TABLE II
MEDIAN CONVERGENCE (ITERATIONS) OF ALGORITHMS WITH ALTERNATING TARGET DATA RATE FOR USERS WITH **NO INITIAL TRAINING**

No. of Users	QL	DQN	DDPG	GA	PSO
$N = 2$	1160	256	357	408	366
$N = 3$	92973*	1632	857	844	491
$N = 4$	-	192698*	3732	1514	1409
$N = 5$	-	-	4236	2576*	1879
$N = 10$	-	-	8519	3438*	2687*
$N = 15$	-	-	12793	6296*	5189*

*Converged less than 50% of the time.

TABLE III
MEDIAN CONVERGENCE (ITERATIONS) OF ALGORITHMS **AFTER TRAINING** WITH ALTERNATING TARGET DATA RATE FOR USERS

No. of Users	DDPG	GA	PSO
$N = 2$	17	408	366
$N = 3$	132	844	491
$N = 4$	169	1514	1409
$N = 5$	236	2576*	1879
$N = 10$	338	3438*	2687*
$N = 15$	376	6296*	3989*

* Converged less than 50% of the time. ** GA and PSO do not have memory; their performance is equal to before training.

Additionally, we have realized that using a proper reward function for RL-based algorithms is one of the dominant forces for success. In our reward function, which is defined in (16), the use of the target band plays a critical role in the stability of QL and DQN algorithms, whereas it does not affect DDPG that significantly. The QL-based and DQN-based algorithms try to find a single solution that works for all users at once due to their discrete action space, while the DDPG-based algorithm finds a feasible solution and converges through improving the actions for each user one-by-one.

VI. NUMERICAL RESULTS

In this section, we present our numerical results for benchmarking multiagent RL-based power allocation schemes. We consider a square room with 12 m width. The Cartesian system origin (0,0) is defined as the center of the room. The RF AP is located at the origin. There are four VLC APs in the room positioned at $(-3,-3),(-3,3),(3,-3)$ and (3,3). All APs has the height of 3 meters. The system parameters for the simulations are provided in Table I. We have simulated the given system in 100 Monte Carlo experiments for each setting in Table II and Table III.

We benchmark QL, DQN, and DDPG-based multiagent power allocation algorithms and GA-based and PSO-based power allocation techniques for the number of users $N = 2$ to $N = 15$ using our simulation results before and after the training phase for learning. We consider convergence performance as the primary performance metric in our numerical results. Additionally, as another performance metric, the convergence rate is defined as the ratio of the cases system converges to a stable situation. Finally, we provide the average normalized distance of actual rates of users to the target data rate as our final performance metric. The performance of the proposed algorithm

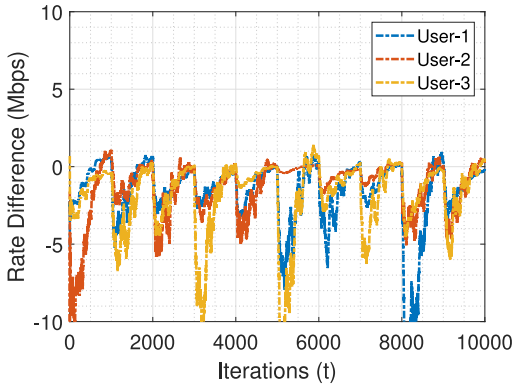

Fig. 2. A sample instance for Distributed DQN-based algorithm serving 3 users with no initial training.

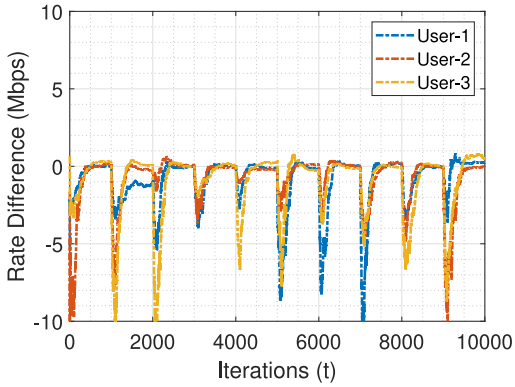

Fig. 3. A sample instance for Distributed DDPG-based algorithm serving 3 users with no initial training.

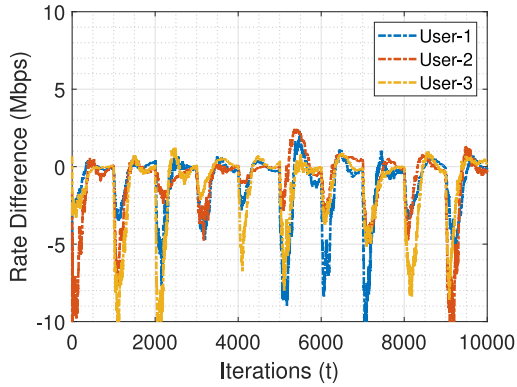

Fig. 4. A sample instance for Centralized GA-based algorithm serving 3 users.

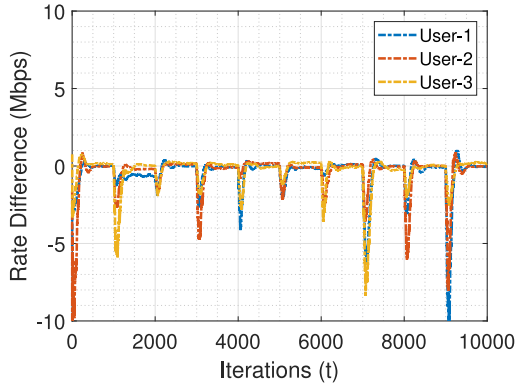

Fig. 5. A sample instance for Centralized PSO-based algorithm serving 3 users.

and other techniques are compared before and after training. The performance with no initial training proves the adaptability speed of the DDPG-based algorithm. The performance after training shows its robustness, while other techniques do not hold such adaptability and robustness.

A. Performance Evaluation Before Training

As an example, we set the number of users to $N = 3$, and we present the performance of DQN and DDPG-based power allocation algorithms in a sample training instance in Fig. 2 and Fig. 3, respectively. The target data rates alternate to feasible value randomly every 1000 iteration, which makes the problem dynamic and more complex. The power allocation actions for all users are entangled by their discrete action space for the DQN-based algorithm. Any action at an AP changes all users' rates at once. However, in the DDPG-based algorithm, each user's power allocation is continuous and separate outputs of each agent's neural network. This property allows the DDPG-based algorithm to act stable and reach a solution for the whole system while not hindering one user for another.

In Fig. 2, we show the rate difference for all users according to their target data rate for the DQN-based algorithm. The DQN-based multiagent algorithm search for a possible action set for achieving convergence. The discrete action space causes ripples in the rate difference between the target data rate and users' actual rate since each action changes power allocation for every user. In Fig. 3, the same setup is simulated for the DDPG-based algorithm. The DDPG-based multiagent algorithm gradually improves the absolute rate difference with its continuous action space, where each user's power is allocated separately and individually. Continuous action space allows the DDPG-based algorithm to learn which individual power allocations are responsible for increasing or decreasing the reward. The DDPG-based algorithm is more stable compared to the DQN-based algorithm. In Fig. 4 and Fig. 5, we demonstrate the same sample instance for GA and PSO algorithms. Note that they assume coordination between APs and control their transmit power for downlink in a centralized manner. Centralized control allows them to converge sooner than DQN and DDPG before training. However, we show that Multiagent DDPG convergence performance surpasses their performance after the training, even in a distributed setting.

In Fig. 6, we present the performance of each algorithm for the same sample instance in terms of the reward function. The PSO-based algorithm has the best performance before training. We compare the median convergence performance for five algorithms with no initial training in Table II. Note that the central premise of DRL-based algorithms is learning by experience. These numbers are provided to compare the training

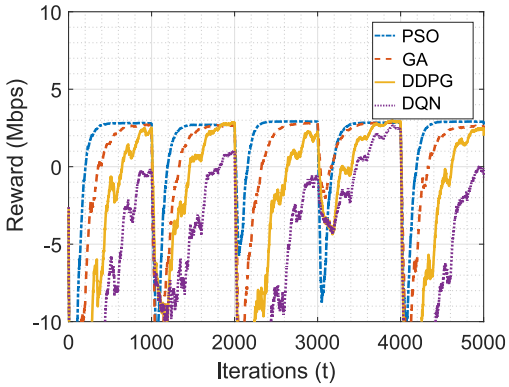

Fig. 6. Reward function plot of a sample instance for 3 users with alternating target data rate with no initial training.

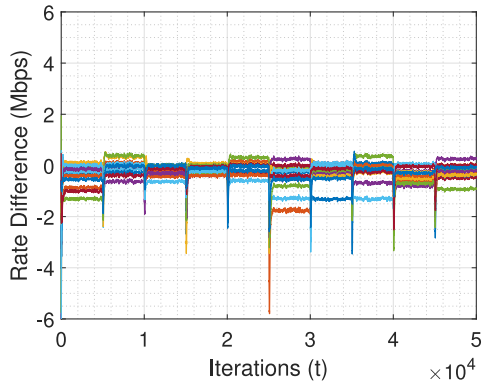

Fig. 7. A sample instance for DDPG-based algorithm serving 15 users with alternating target data rate after training.

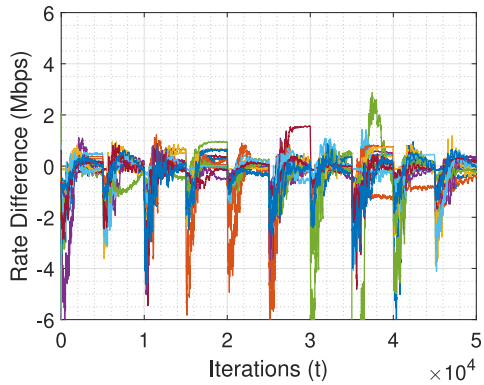

Fig. 8. A sample instance for GA-based algorithm serving 15 users with alternating target data rate.

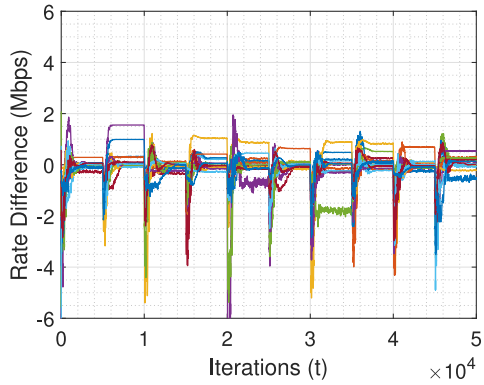

Fig. 9. A sample instance for PSO-based algorithm serving 15 users with alternating target data rate.

performance of distributed DQN and DDPG and centralized GA and PSO algorithms. As the number of users increases, QL-based and DQN-based algorithms struggle to reach convergence within a reasonable time. On the other hand, the correlation between convergence performance and the number of users is close to linear in the DDPG-based algorithm. GA and PSO algorithms have better convergence performance, but they do not have experience memory. Once the system dynamics such as target data rate for users change, they may not adapt with an increasing number of users. DDPG, on the other hand, learns with experience and becomes more robust at every instance. We provide its performance after training and compare it to others in the following subsection.

B. Performance Evaluation After Training

We benchmark QL, DQN, and DDPG-based multiagent power allocation algorithms for the number of users $N = 2$ to $N = 15$ using our simulation results for steady-state (after training) performance. We train our RL agents under different settings, multiple locations of users, and various target data rate requirements for 100 thousand iterations and use the trained models to benchmark their steady-state performance. The steady-state performances of GA and PSO are equal to their training performance due to their memoryless architecture.

The median convergence performance of algorithms is given in Table III. The robustness and scalability of DDPG allow it to outperform GA and PSO once it is trained. More than 50% of the time, GA-based algorithm is not able to converge for more than 4 users, and PSO-based algorithm is not able to converge for 10 or more users. On the other hand, the Multiagent DDPG-based algorithm is the most robust and fastest to converge as it has the median convergence time of 376 iterations for 100 Monte Carlo experiments for $N = 15$ users, which shows trained distributed DDPG-based algorithm is able to cope with larger number of users.

In Figs. 7–Fig. 9, we represent a sample instance after training with $N = 15$ for Multiagent DDPG-based algorithm, GA-based algorithm, and PSO-based algorithm serving 15 users with mobility and alternating target data rate. In this case, the target data rate of users is changing every 5000 iterations for the sake of stability and convergence of heuristic methods. The multiagent DDPG-based algorithm achieves convergence within 300 to 400 iterations, while GA-based and PSO-based algorithms struggle to reach convergence before the user changes its target data rate.

In Fig. 10, the reward function plot for the same sample instance with $N = 15$ is provided. The stability and robustness of multiagent DDPG are visible, and it has the best convergence

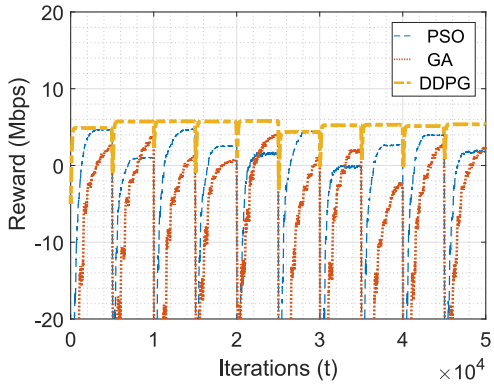

Fig. 10. Reward function plot of a sample instance for 15 users with alternating target data rate after training.

performance compared to GA and PSO algorithms and the highest steady-state performance in reaching the target data rate.

As a result, the multiagent-DDPG based algorithm performs the best amongst DRL-based algorithms or heuristic methods thanks to its neural network architecture and continuous state and action space after training. It does not require coordination between APs as conventional methods and centralized power control as heuristic approaches. DDPG-based agents learn how to cope with dynamic channel conditions, mobility of multiple users with alternating target data rate requirements even with a large number of users. The distributed DDPG-based algorithm is the least affected by the increase in the number of users in terms of training and inference.

VII. Conclusion

In this paper, we propose a *distributed* DDPG-based power allocation algorithm for hybrid RF/VLC networks with *limited resources* and multiple *mobile* users with *varying* QoS requirements. Three distributed multiagent DRL-based algorithms, namely QL, DQN, and DDPG-based approaches are implemented. The proposed DDPG-based algorithm is benchmarked amongst distributed DRL algoritmhs and against centralized heuristic methods GA and PSO. It is shown that the convergence time of the distributed DDPG-based algorithm's has a linear correlation with the number of users, while the convergence time for QL-based and DQN-based algorithms increases exponentially thanks to its continuous action space. The performance of DDPG-based algorithm after training on reaching the target data rate is better than other RL-based algorithms and heuristic methods. Additionally, the DDPG-based algorithm is the most robust algorithm against user mobility and alternating user requirements. As a result, the DDPG-based algorithm shows superiority to other algorithms (QL, DQN, GA, and PSO) in all performance metrics when the number of users is large.

Acknowledgment

The findings achieved herein are solely the responsibility of the authors.

References

[1] B. S. Çiftler, M. M. Abdallah, A. Alwarafy, and M. Hamdi, "DQN-based multi-user power allocation for hybrid RF/VLC networks," in *Proc. IEEE Int. Conf. Commun.: Opt. Netw. Syst. Symp.*, Montreal, Canada, 2021, pp. 1–6.

[2] M. Kashef, M. Ismail, M. Abdallah, K. A. Qaraqe, and E. Serpedin, "Energy efficient resource allocation for mixed RF/VLC heterogeneous wireless networks," *IEEE J. Sel. Areas Commun.*, vol. 34, no. 4, pp. 883–893, Apr. 2016.

[3] X. Li, R. Zhang, and L. Hanzo, "Cooperative load balancing in hybrid visible light communications and WiFi," *IEEE Trans. Commun.*, vol. 63, no. 4, pp. 1319–1329, Apr. 2015.

[4] J. Al-Khori, G. Nauryzbayev, M. Abdallah, and M. Hamdi, "Secrecy capacity of hybrid RF/VLC DF relaying networks with jamming," in *Proc. Int. Conf. Comput., Netw. Commun.*, 2019, pp. 67–72.

[5] X. Li, J. Fang, W. Cheng, H. Duan, Z. Chen, and H. Li, "Intelligent power control for spectrum sharing in cognitive radios: A deep reinforcement learning approach," *IEEE Access*, vol. 6, pp. 25463–25473, 2018.

[6] M. Obeed, A. M. Salhab, M. S. Alouini, and S. A. Zummo, "On optimizing VLC networks for downlink multi-user transmission: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 3, pp. 2947–2976, Jul.–Sep. 2019.

[7] G. Wang, Y. Shao, L.-K. Chen, and J. Zhao, "Subcarrier and power allocation in OFDM-NOMA VLC systems," *IEEE Photon. Technol. Lett.*, vol. 33, no. 4, pp. 189–192, Feb. 2021.

[8] M. S. Demir, S. M. Sait, and M. Uysal, "Unified resource allocation and mobility management technique using particle swarm optimization for VLC networks," *IEEE Photon. J.*, vol. 10, no. 6, Dec. 2018.

[9] J. Wang *et al.*, "A machine learning framework for resource allocation assisted by cloud computing," *IEEE Netw.*, vol. 32, no. 2, pp. 144–151, Mar./Apr. 2018.

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

[11] F. Hussain, S. A. Hassan, R.Hussain, and E. Hossain, "Machine learning for resource management in cellular and IoT networks: Potentials, current solutions, and open challenges," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 2, pp. 1251–1275, Apr.–Jun. 2020. [Online]. Available: http://arxiv.org/abs/1907.08965

[12] E. Ghadimi, F. D. Calabrese, G. Peters, and P. Soldati, "A reinforcement learning approach to power control and rate adaptation in cellular networks," in *Proc. IEEE Int. Conf. Commun.*, 2017, pp. 1–7.

[13] A. M. Alenezi and K. A. Hamdi, "Reinforcement learning approach for hybrid WiFi-VLC networks," in *Proc. IEEE 91st Veh. Technol. Conf.*, 2020, pp. 1–5.

[14] H. Yang, A. Alphones, W. -D. Zhong, C. Chen, and X. Xie, "Learning-based energy-efficient resource management by heterogeneous RF/VLC for ultra-reliable low-latency industrial IoT networks," *IEEE Trans. Ind. Informat.*, vol. 16, no. 8, pp. 5565–5576, Aug. 2020.

[15] S. Shrivastava, B. Chen, C. Chen, H. Wang, and M. Dai, "Deep Q-network learning based downlink resource allocation for hybrid RF/VLC systems," *IEEE Access*, vol. 8, pp. 149412–149434, 2020.

[16] Y. Wei, F. R. Yu, M. Song, and Z. Han, "User scheduling and resource allocation in HetNets with hybrid energy supply: An actor-critic reinforcement learning approach," *IEEE Trans. Wireless Commun.*, vol. 17, no. 1, pp. 680–692, Jan. 2018.

[17] P. C. Chen, Y. C. Chen, W. H. Huang, C. W. Huang, and O. Tirkkonen, "DDPG-based radio resource management for user interactive mobile edge networks," in *Proc. 2nd 6G Wireless Summit*, 2020, pp. 1–5.

[18] F. Meng, P. Chen, L. Wu, and J. Cheng, "Power allocation in multi-user cellular networks: Deep reinforcement learning approaches," *IEEE Trans. Wirel. Commun.*, vol. 19, no. 10, pp. 6255–6267, Oct. 2020.

[19] Y. H. Xu, C. C. Yang, M. Hua, and W. Zhou, "Deep deterministic policy gradient (DDPG)-Based resource allocation scheme for NOMA vehicular communications," *IEEE Access*, vol. 8, pp. 18797–18807, 2020.

[20] L. Liang, H. Ye, G. Yu, and G. Y. Li, "Deep learning based wireless resource allocation with application to vehicular networks," *Proc. IEEE*, vol. 108, no. 2, pp. 341–356, Feb. 2020. [Online]. Available: http://arxiv.org/abs/1907.03289

[21] D. A. Basnayaka and H. Haas, "Hybrid RF and VLC systems: Improving user data rate performance of VLC systems," in *Proc. IEEE 81st Veh. Technol. Conf.*, 2015, pp. 1–5.

[22] M. A. M. Obeed, S. A. S. Zummo, and M.-S. Alouini, "Joint optimization of power allocation and load balancing for hybrid VLC/RF networks," *J. Opt. Commun. Netw.*, vol. 10, no. 5, pp. 553–562, May 2018. [Online]. Available: http://jocn.osa.org/abstract.cfm?URI=jocn-10-5-553

[23] M. Kashef, M. Abdallah, and N. Al-Dhahir, "Transmit power optimization for a hybrid PLC/VLC/RF communication system," *IEEE Trans. Green Commun. Netw.*, vol. 2, no. 1, pp. 234–245, Mar. 2018.

[24] N. C. Luong *et al.*, "Applications of deep reinforcement learning in communications and networking: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 4, pp. 3133–3174, Oct.–Dec. 2019.

[25] A. Alwarafy, M. Abdallah, B. S. Çiftler, A. Al-Fuqaha, and M. Hamdi, "The frontiers of deep reinforcement learning for resource management in future wireless HetNets: Techniques, challenges, and research directions," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 322–365, 2022, doi: 10.1109/OJ-COMS.2022.3153226.

[26] J. Kong, Z. Wu, M. Ismail, E. Serpedin, and K. A. Qaraqe, "Q-learning based two-timescale power allocation for multi-homing hybrid RF/VLC networks," *IEEE Wireless Commun. Lett.*, vol. 9, no. 4, pp. 443–447, Apr. 2020.

[27] D. A. Basnayaka and H. Haas, "Design and analysis of a hybrid radio frequency and visible light communication system," *IEEE Trans. Commun.*, vol. 65, no. 10, pp. 4334–4347, Oct. 2017.

[28] M. Plappert *et al.*, "Parameter space noise for exploration," *CoRR*, vol. abs/1706.01905, 2017. [Online]. Available: http://arxiv.org/abs/1706.01905

[29] T. Domhan, J. T. Springenberg, and F. Hutter, "Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves," in *Proc. 24th Int. Joint Conf. Artif. Intell.*, 2015, pp. 3460–3468.