VLC and D2D Heterogeneous Network Optimization: A Reinforcement Learning Approach Based on Equilibrium Problems With Equilibrium Constraints

Neetu Raveendran, *Member, IEEE*, Huaqing Zhang, *Member, IEEE*, Dusit Niyato, *Fellow, IEEE*, Fang Yang, *Senior Member, IEEE*, Jian Song, *Fellow, IEEE*, and Zhu Han, *Fellow, IEEE*

***Abstract*— The radio frequency spectrum crunch has triggered the harnessing of other sources of bandwidth, for which visible light is a promising candidate. Even though visible light communication (VLC) ensures high capacity, coverage is limited. This necessitates the integration of VLC and device-to-device (D2D) technologies into heterogeneous networks. In particular, mobile users which are accessible by the VLC transmitters can relay data to mobile users which are not, by means of D2D communication. However, due to the distributed behaviors of mobile users, determining optimal data transmission routes from VLC transmitters to end mobile devices is a major challenge. In this paper, we propose a reinforcement learning (RL)-based approach to determine multi-hop data transmission routes in an indoor VLC-D2D heterogeneous network. We obtain the rewards for the RL-based method dynamically, by formulating the interactions between the mobile users relaying the data as an equilibrium problem with equilibrium constraints and using alternating direction method of multipliers to solve it. The proposed technique can achieve optimal data transmission routes in a distributed manner. The simulation results demonstrate the effectiveness of the proposed approach, showing that transmission routes with low delays and high capacities can be achieved through the learning algorithm.**

***Index Terms*— Visible light communication, device-to-device, heterogeneous network, reinforcement learning, equilibrium problem with equilibrium constraints, alternating direction method of multipliers.**

Manuscript received June 26, 2018; revised November 5, 2018; accepted December 17, 2018. Date of publication January 4, 2019; date of current version February 11, 2019. This work was supported in part by US MURI AFOSR MURI under Grant 18RT0073, in part by the NSF under Grant CNS-1717454, Grant CNS-1731424, Grant CNS-1702850, Grant CNS-1646607, and Grant ECCS-1547201, in part by the National Natural Science Foundation of China under Grant 61871255, in part by the Natural Science Foundation of Guangdong Province under Grant 2015A030312006, in part by the Guangdong Key Laboratory Project under Grant 2017B030314147, in part by WASP/NTU under Grant M4082187 (4080), in part by the Singapore MOE Tier 1 under Grant 2017-T1-002-007 RG122/17, in part by the MOE Tier 2 under Grant MOE2014-T2-2-015 ARC4/15 and Grant NRF2015-NRF-ISF001-2277, and in part by EMA Energy Resilience under Grant NRF2017EWT-EP003-041. The associate editor coordinating the review of this paper and approving it for publication was J. Tang. *(Corresponding author: Neetu Raveendran.)*

N. Raveendran is with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77004 USA (e-mail: nraveendran@uh.edu).

H. Zhang is with Petuum, Inc., Sunnyvale, CA 94087 USA (e-mail: markasjunior@gmail.com).

D. Niyato is with the School of Computer Science and Engineering, Nanyang Technological University, Singapore 639798 (e-mail: dniyato@ntu.edu.sg).

F. Yang and J. Song are with the Department of Electronic Engineering, Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China, and also with the Key Laboratory of Digital TV System of Guangdong Province and Shenzhen City, Research Institute of Tsinghua University in Shenzhen, Shenzhen 518057, China (e-mail: fangyang@tsinghua.edu.cn; jsong@tsinghua.edu.cn).

Z. Han is with the Department of Electrical and Computer Engineering, University of Houston, Houston, TX 77004 USA, and also with the Department of Computer Science and Engineering, Kyung Hee University, Seoul 446-701, South Korea (e-mail: zhan2@uh.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TWC.2018.2890057

I. Introduction

THE global mobile data traffic has grown 18-fold over the past 5 years, and is expected to increase 7-fold between 2016 and 2021 [1]. This tremendous increase of mobile broadband has resulted in a Radio Frequency (RF) spectrum crunch in wireless communications. This has made exploring and exploiting alternative sources of bandwidth inevitable. With a vast bandwidth of approximately 300 THz, visible light spectrum can facilitate high data rate communication, called Visible Light Communication (VLC). Apart from the immense bandwidth, some of the advantages of using visible light for communication are: (i) it is unlicensed and hence, provides free spectrum, (ii) it is secure as its propagation is limited, and (iii) VLC is easy to be implemented through already existing ubiquitous and inexpensive visible light sources such as Light Emitting Diodes (LEDs) [2], [3]. LEDs are expected to be the major sources of illumination, and also the transmitters for VLC, using which data rates in the range of hundreds of $Mbps$ can be achieved [4].

Although VLC ensures high data transmission rates, it is hindered from serving users in strong sunlight areas or shaded areas, resulting in limited coverage. Taking this into consideration, combining cellular communication with VLC has been proposed. A heterogeneous network integrating cellular and VLC communications guarantees good coverage from the cellular network and high capacity from the VLC network [5]. In such a heterogeneous network, the traffic congestion in the cellular network is minimized by offloading some of the traffic to the VLC network. This improves the spectrum reuse in the heterogeneous network, by utilizing both the licensed and the visible spectrums for communication [6]. In these networks, the mobile users which are able to access the VLC network can relay the data to the mobile users which cannot be served by the VLC network. This can be realized using Device-to-Device (D2D) communication.

D2D communication enables mobile devices to transmit data directly between each other, without relaying it through a base station, thus improving the spectral and energy efficiencies, and minimizing the latency [7]. D2D communication can be integrated with VLC to serve mobile devices that are inaccessible by the VLC transmitters [6]. The mobile users being served can pay the mobile users which act as relays, for their services. However, there are many issues in such a VLC-D2D heterogeneous network. One of the most important issues is the determination of data transmission routes from the VLC transmitters to the end mobile devices. Traditionally, the VLC Service Provider (VLCSP) can determine this data transmission route based on maximizing its revenue and minimizing the overall latency, in a centralized manner [6].

However, the data transmission environment between the VLC transmitters and the mobile devices, and amongst the mobile devices themselves, in a VLC-D2D heterogeneous network, is highly dynamic. There can also exist a competition among the mobile devices accessible by the VLC transmitters, to relay the data to the mobile devices inaccessible by the VLC transmitters, and obtain revenue. A centralized solution available for the VLCSP and the mobile devices is difficult in such cases. Additionally, due to the random nature of the positions and data requirements of the mobile devices, the parameters in the network cannot be fully predicted. This calls for techniques to determine optimal distributed data transmission routes for a stochastic communication environment, which indeed is very challenging.

Recently, Machine Learning (ML) paradigms have gained wide popularity in several decision making scenarios, and Reinforcement Learning (RL) is a general framework that helps to learn behavior through trial-and-error interactions with a dynamic environment [8]–[10]. Consequently, taking into account the dynamic and unpredictable nature of our considered communication environment, in this paper, we propose an RL based method to determine data transmission routes in the VLC-D2D heterogeneous network.

Being a learning method that does not require a model of the considered environment makes RL suitable for our scenario, using which we can deduce optimal data transmission routes through trial-and-error. In a typical RL scenario, a learning agent which is capable of taking *actions*, observes the environment after each of its actions [11]. Each action influences the *state* reached by the agent in the future. The success in RL scenarios is measured through total *rewards*, and hence, the agent basically aims to maximize its accumulated rewards through its actions. Therefore, in order to perform learning in our heterogeneous network scenario, we need a method to calculate the rewards.

Accordingly, we consider a network where the VLCSP, the Cellular Service Provider (CSP), and the mobile users, are independent entities. In the heterogeneous network environment under consideration, the independent entities can use their experiences from interacting with the environment, to improve their behaviors [12]. In order to obtain the rewards, we also model the competition among the mobile devices accessible by the VLC transmitters in relaying the data. Subsequently, we propose an Alternating Direction Method of Multipliers (ADMM) based Equilibrium Problem with Equilibrium Constraints (EPEC) approach for determining the optimal rewards dynamically, which is discussed in detail later. The interconnection between RL, EPEC, and ADMM in the proposed algorithm is shown in Fig. 1.

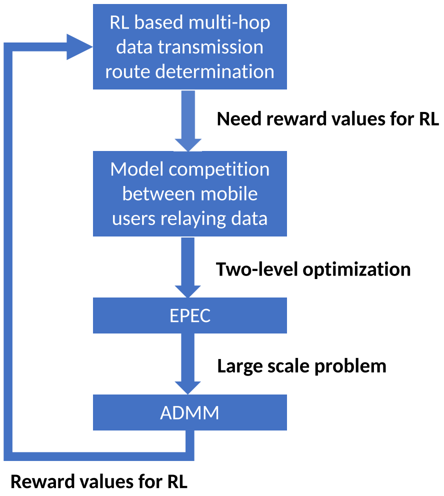
RL based multi-hop data transmission route determination
Need reward values for RL
Model competition between mobile users relaying data
Two-level optimization
EPEC
Large scale problem
ADMM
Reward values for RL

Fig. 1. Proposed multi-hop route selection algorithm for VLC-D2D heterogeneous network.

The major contributions of this paper are summarized as follows:

- We propose a multi-hop data transmission route determination method for an indoor VLC-D2D heterogeneous network, utilizing an RL based technique. This is a distributed method that can determine the route based on local information, unlike the conventional, centralized method where the VLCSP determines the route based on data transmission path delays.
- The proposed RL method utilizes Q-learning, which is a direct RL technique, and works by continuously improving the knowledge of the consequences of certain actions at certain states. In our scenario, it enables the transmitted data to learn from its interactions with the environment to find an optimal route in a distributed fashion.
- In order to compute the rewards dynamically for the RL based route determination method, we formulate the interactions between the mobile users which relay data using D2D communication, as an EPEC optimization problem. An EPEC is a hierarchical optimization problem that contains equilibrium constraints at two levels. Here, the utilities of the mobile users which send and receive the relayed data are simultaneously optimized by using EPEC.

- Then, we utilize the properties of ADMM as a large scale optimization tool to solve the EPEC problem, given the large number of entities (mobile users), in the considered scenario. Using ADMM, we achieve optimal solutions for the EPEC, which contribute to the rewards for the Q-learning based route determination approach.
- The effectiveness of the proposed algorithm is then validated through simulations, which emphasize the effect of the RL method on transmission capacity and latency. The simulation results highlight the effect of the number of learning steps and the importance of future rewards, on the data transmission rates and path delays.

We organize the rest of this paper as follows. We discuss previous work related to this paper in Section II. In Section III, we present the system architecture and the key parameters used in our VLC-D2D heterogeneous network optimization model. Then, in Section IV, we formulate the optimization problem of the proposed model, and introduce the RL and EPEC formulations, which are used in our proposed algorithms. The proposed algorithms and their detailed analyses are provided in Section V. Here, we discuss the Q-learning based algorithm for optimal data transmission route determination in Section V-A, and the ADMM based EPEC algorithm for the determination of rewards in Section V-B. We present the performance of the proposed Q-learning based optimal route determination method, which also incorporates the ADMM based EPEC technique, in Section VI, where we firstly discuss the simulation results in Section VI-A, and then, discuss a few aspects of the results in Section VI-B. Finally, conclusions are drawn in Section VII.

II. Literature Review

A promising solution to the wireless capacity crunch issue is the deployment of heterogeneous networks. Reference [13] shows 90% offloading from the macrocell base station by using heterogeneous small cell-based networks. A survey on the state-of-the-art and challenges of Long Term Evolution-Advanced (LTE-A) heterogeneous networks is given in [14], where the elements of LTE-A heterogeneous networks introduced in different LTE releases of 3GPP are also summarized, among which D2D communication is introduced in Release 12. Reference [15] utilizes game-theoretic approaches to provide distributed solutions to the resource allocation issues in D2D communication underlaying cellular networks, where the complex strategies of the D2D and cellular users to maximize their own utilities are modeled using the tools from game theory. Improving the coverage for mobile users at the cell edge is of key importance in wireless communication, and is facilitated using D2D range extension in [16]. Reference [17] deals with the energy efficiency maximization in a D2D-assisted heterogeneous network, and considers optimal power allocation, along with user equipment association.

Owing to the remarkable wireless traffic offloading potential of VLC, the research on inclusion of VLC in heterogeneous networks is tremendously growing [18]–[22]. Reference [23] elaborates on user-centric VLC heterogeneous networks from the signal coverage, system control, and service provision perspectives, and discusses a few open challenges. An indoor heterogeneous network combining VLC and RF have been proposed in [24], where a new VLC frame, multiple access mechanism, and a novel handover scheme have been introduced, and the capacity performance of the network has been improved compared to an RF only system. Reference [25] proposes an indoor hybrid system integrating WiFi and VLC, which utilizes the bandwidth benefits of VLC, and proposes the optimal resource allocation among users. An indoor VLC and RF heterogeneous network is investigated from the energy efficiency point of view in [26]. Reference [27] considers an indoor VLC attocell and RF femtocell network, where the optimal resource allocation is achieved through a distributed algorithm.

In order to implement such complex heterogeneous networks and to ensure proper allocation of resources, we need efficient algorithms that can handle complicated scenarios. Research on ML algorithms is quite prominent in this regard, of which RL research has gained a lot of momentum in the recent years. Since it is a learning method that does not need a model of the environment, RL is well suited for scenarios where the environment changes during learning [28]. It has been discussed widely in areas like machine learning, neural networks, operations research, control theory, and so on [29]. RL is the learning algorithm behind AlphaGo, the first computer program to defeat a professional player at Go, one of the most challenging classical games [30]. Reference [10] performs a basic survey of RL, and also discusses a few classic model-free algorithms. A survey of RL in robotics is given in [31], through behavior generation in robots. A comprehensive survey on the recent developments in RL with function approximation, and a comparison of the performance of different RL algorithms is provided in [29]. Reference [32] demonstrates a simple two-player soccer game using RL techniques, and a model for a route planning system based on multi-agent RL is proposed in [33].

Due to the flexibility and effectiveness of RL in learning, it has been applied to solve many issues in wireless communication. Reference [34] proposes an RL based sub-band selection policy to anti-jamming communications with Wideband Autonomous Cognitive Radios (WACRs) in a scenario with multiple policy-learning agents. The sub-band selection problem in wideband Cognitive Radio (CR) is dealt with in [35], using an extension of the Q-learning algorithm, called the replicated Q-learning. Some other applications of RL can be found in [36]–[43], which are in wireless sensor networks, LTE-A networks, cognitive network and radio resource management, wireless communication in healthcare, and energy-efficient wireless communication.

In order to harness the benefits of heterogeneous networks as well as the potentials of the VLC and D2D technologies, our scenario of interest is an indoor VLC-D2D heterogeneous network. An indoor VLC-D2D heterogeneous network scenario has been considered in [6], where the mobile devices accessible by the VLC transmitters, relay the data to the mobile devices which are not accessible by the VLC transmitters. Reference [6] proposes a hierarchical game, where the interactions between the VLCSP and the CSP, the CSP and the

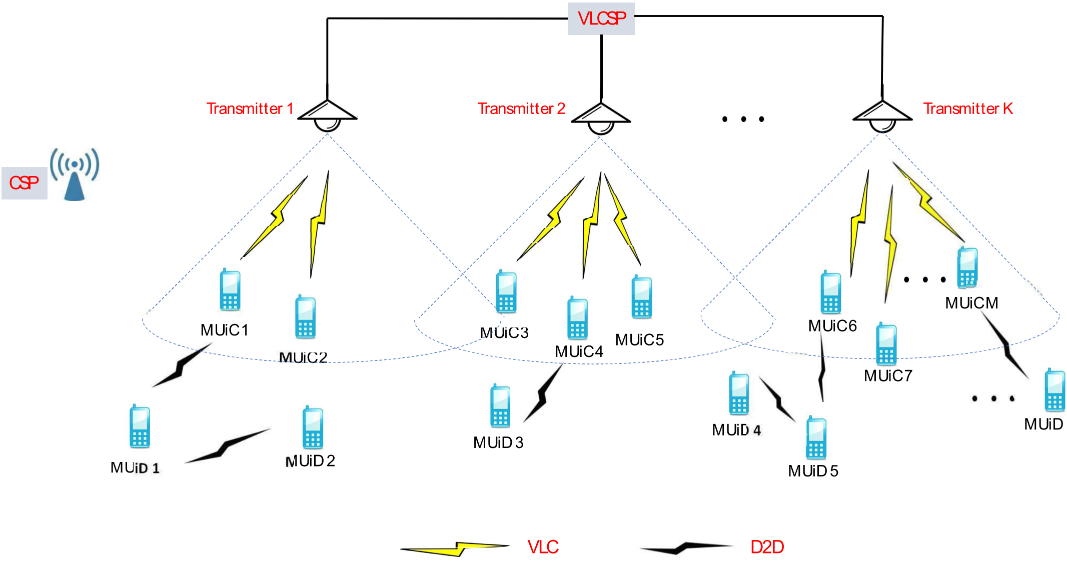

Fig. 2. System architecture of VLC-D2D heterogeneous network.

mobile users, and between the VLCSP and the mobile users are modeled as Stackelberg games. The network arrives at the solution when the mobile users achieve a Nash equilibrium among them, and the above mentioned Stackelberg games achieve Stackelberg equilibriums. Nevertheless, the data transmission route is determined by the VLCSP, by considering only the transmission path delays.

In the case of the indoor VLC-D2D heterogeneous network considered in this paper, we adopt an RL based route determination method, as discussed in the previous section. To the authors' best knowledge, an RL based method to determine data transmission routes in a dynamic VLC-D2D heterogeneous network environment has not been employed in any previous work. Also, determining the rewards for RL dynamically during the learning process has not been considered before. Hence, in this paper, we utilize the potential of RL as a powerful learning algorithm for a stochastic environment. The rewards are computed during the RL process using EPEC, which is solved by making use of the capacity of ADMM as an efficient large-scale optimization tool.

III. System Model

We consider an indoor downlink scenario consisting of K VLC transmitters of a VLCSP, a CSP, and T mobile users, following the settings in [6]. Out of the T mobile users, there are M Mobile Users in Coverage area (MUiCs) of the VLC transmitters and N Mobile Users in Darkness (MUiDs). As shown in Fig. 2, MUiCs are the mobile users which are under the visible light cones of the VLC transmitters, and MUiDs are the mobile users which are not directly under the visible light coverage of the VLC transmitters. For data transmission, as the MUiDs can only receive weak signals from the cellular base stations, D2D communication is adopted in which some MUiCs and MUiDs can act as relays for the end MUiDs [6]. The MUiCs and MUiDs that relay the service from the VLC transmitter to the end MUiD are called Mobile Users as Relays (MUaRs). As shown in Fig. 2, MUiC 1 is able to receive traffic from the VLC transmitter. Then, MUiC 1 acts as a relay for MUiD 1 which also acts as a relay for the end MUiD 2.

For the visible light communication between VLC transmitter κ, for $\kappa \in \{1, 2, \ldots, K\}$, and MUiC m, for $m \in \{1, 2, \ldots, M\}$, the transmission rate can be defined as

$$C_{\kappa m} = S_\kappa \log_2\left(1 + \frac{P_\kappa G_{\kappa m}}{\sigma_I^2 + \sigma_N^2}\right), \quad (1)$$

where S_κ is the amount of spectrum for each of the K VLC transmitters, P_κ is the transmit power of each of the K VLC transmitters, $G_{\kappa m}$ is the channel gain between VLC transmitter κ and MUiC m, σ_I^2 is the interference from other visible light sources, and σ_N^2 is the channel noise.

For the D2D communication between MUaR i, for $i \in \{1, 2, \ldots, T\}$, and MUaR j, for $j \in \{1, 2, \ldots, T\}$, $i \neq j$ (or between MUaR i, for $i \in \{1, 2, \ldots, T\}$, and end MUiD j, for $j \in \{1, 2, \ldots, N\}$), the transmission rate can be defined as

$$C_{ij} = S_{ij} \log_2\left(1 + \frac{P_{ij} G_{ij}}{I_c + N_0}\right), \quad (2)$$

where S_{ij} is the amount of allocated wireless spectrum for MUaR i to transmit data to MUaR j (or for MUaR i to transmit to the end MUiD), P_{ij} is the transmit power from MUaR i to MUaR j (or from MUaR i to the end MUiD), G_{ij} is the channel gain between MUaR i and MUaR j (or between MUaR i and the end MUiD), I_c is the interference from cellular uplink, and N_0 is the additive noise.

We consider the VLC transmitters, the MUaRs and the end MUiDs in our network as nodes in a graph. For the multihop wireless data transmission from a VLC transmitter to an end MUiD, the data packet size is denoted by $\mathcal{M}$. The penalty due to the delay between node i and node j can be expressed

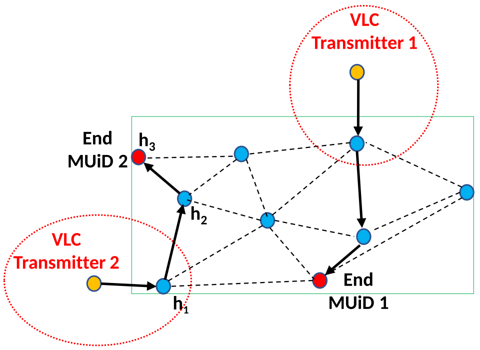

Fig. 3. L-hop data transmission route.

as [6]

$$D_{ij} = \alpha \frac{\mathcal{M}}{C_{ij}}, \tag{3}$$

where α denotes the penalty of unit service delay. The penalty due to the total delay is the sum of the penalties between all the nodes in the data transmission service route.

Accordingly, we consider an L-hop route, $\{h_1, h_2, \ldots, h_L\}$, for the data transmission from VLC transmitter κ, for $\kappa \in \{1, 2, \ldots, K\}$, to the end MUiD, where node h_l, $\forall l \in \{1, 2, \ldots, L-1\}$, is an MUaR, and node h_L is the end MUiD.

The utility for node h_l in the transmission route, is given by the revenue received by the node minus the delay penalty and the cost of the wireless spectrum. The utility function can be expressed as

$$U_l^{MUaR} = \beta\mathcal{M} - \gamma D_l - r_l S_l + r_{l-1} S_{l-1}, \tag{4}$$

where β and γ are the weight factors. D_l has the same physical meaning as D_{ij} from (3), which is the penalty of delay from node h_l to node h_{l+1} in the transmission route, r_l is the price per unit of the wireless spectrum resources, and S_l has the same physical meaning as S_{ij} from (2), which is the amount of allocated wireless spectrum for node h_l. Thus, $\beta\mathcal{M}$ is the revenue obtained by node h_l by relaying a data packet of size $\mathcal{M}$, γD_l is the delay penalty, $r_l S_l$ is the price paid for the spectrum, and $r_{l-1} S_{l-1}$ is the revenue obtained by providing spectrum to the upstream node h_{l-1}.

In this paper, we propose to determine the L-hop data transmission route, as shown in Fig. 3, from a VLC transmitter to the end MUiD, utilizing an RL technique, which is discussed in detail in the next section. In order to formulate the model for the RL technique, let us consider the transmitted data from a particular VLC transmitter to a particular end MUiD as the learning *agent*, the current location (node) of the transmitted data as the *state*, and the transmission direction of the agent from the current node to the next accessible node as an *action*. Here, as there can be different data transmitted from different VLC transmitters to the same end MUiD or different end MUiDs at the same time, we consider a scenario wherein a number of single agent RL tasks take place simultaneously.

Another key factor in this learning technique is *reward*, which is simply the payoff obtained by the agent, by performing an action at a given state. In RL, the actions may not only affect the immediate reward, but also the rewards obtained in future [11]. In our scenario, the transmission data rate can be considered as the reward. As such, we define the reward matrix $\mathbf{R}$, which contains the rewards obtained by the agent, by selecting every possible action (transmitting to all accessible nodes), from every state (current position of transmitted data). We express the reward matrix for our single agent RL scenario as

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_{11} & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{R}_{1(K+T)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \mathbf{R}_{K1} & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{R}_{K(K+T)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \mathbf{R}_{(K+M)1} & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{R}_{(K+M)(K+T)} \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \mathbf{R}_{(K+T)1} & \cdot & \cdot & \cdot & \cdot & \cdot & \mathbf{R}_{(K+T)(K+T)} \end{bmatrix},$$

where $T = M + N$ is the total number of mobile devices in our indoor scenario. Here, the rows indicate the states, and the columns indicate the actions. For example, $\mathbf{R}_{1(K+1)}$ is the reward obtained by the agent, by taking the action of transmitting data to state $K+1$ (MUiC 1) from state 1 (VLC transmitter 1), $\mathbf{R}_{(K+M+1)(K+T)}$ is the reward obtained by the agent, by taking the action of transmitting data to state $K+T$ (MUiD N) from state $K+M+1$ (MUiD 1), and so on.

Here, $\mathbf{R}_{11}, \mathbf{R}_{22}, \ldots, \mathbf{R}_{(K+T)(K+T)}$ can be taken as 0, since none of the nodes transmits data to themselves. $\mathbf{R}_{11}, \ldots, \mathbf{R}_{1K}, \ldots, \mathbf{R}_{K1}, \ldots, \mathbf{R}_{KK}$ will be 0, as the VLC transmitters will not transmit data to each other. $\mathbf{R}_{(K+1)1}, \ldots, \mathbf{R}_{(K+1)K}, \ldots, \mathbf{R}_{(K+T)1}, \ldots, \mathbf{R}_{(K+T)K}$ will also be 0, as the mobile devices do not transmit data to the VLC transmitters. Additionally, as the VLC transmitters do not send data to the N MUiDs, $\mathbf{R}_{1(K+M+1)}, \ldots, \mathbf{R}_{1(K+T)}, \ldots, \mathbf{R}_{K(K+M+1)}, \ldots, \mathbf{R}_{K(K+T)}$ will be 0. In addition, since an MUiC will not transmit data to another MUiC, columns $K+1$ through $K+M$ of rows $K+1$ through $K+M$ will be 0. Finally, since an MUiD will not transmit data to any MUiC, columns $K+1$ through $K+M$ of rows $K+M+1$ through $K+T$ will also be 0.

IV. Problem Formulation

In our considered scenario, the initial problem for VLC transmitter κ, for $\kappa \in \{1, 2, \ldots, K\}$, is determining an optimal route, $\{h_1, h_2, \ldots, h_L\}$, from the VLC transmitter to the end MUiD. Clearly, this can be considered as a single agent RL scenario, where the agent needs to learn the environment through its experience and take actions to maximize its rewards. In order to realize this, the agent has to *exploit* what is already known, and simultaneously, has to *explore* to be able to choose better actions in the future [11].

Q-learning is a model-free RL technique [44], in the sense that the agent can directly learn about its optimal policies, i.e., a mapping from a state to an action of the agent, without knowing the future rewards [28]. Q-learning works by learning an *action-value* function that constructs the optimal *policy*, by selecting the action with the highest *value*. This optimal policy gives the maximum achievable expected value of the total reward. In this paper, we utilize single agent Q-learning to decide the optimal route for each VLC transmitter to the end MUiD. The detailed analysis of the Q-learning algorithm is included in the next section.

For utilizing Q-learning to decide the data transmission route for VLC transmitter κ, for $\kappa \in \{1, 2, \ldots, K\}$, we need to generate the reward matrix, $\mathbf{R}$, as shown in the previous section. This is performed as an iterative update process, while modeling firstly, the interactions between the VLC transmitters and the MUaRs, and secondly, the interactions between the MUaRs and the end MUiDs.

Initially, we model the interactions between each of the K VLC transmitters and the set of M MUiCs, which is the set of candidates for the first hop, h_1, using (1). This generates some of the rewards in the $\mathbf{R}$ matrix, which are the reward values associated with all the possible first hops (MUiCs) for the agent, from each of the VLC transmitters.

In our scenario, we consider the first hops, h_1 to be MUiCs, and the rest of the hops in the L-hop route to be MUiDs, including the MUiDs acting as MUaRs, and the end MUiDs. Hence, we generalize and formulate the utility function of the MUaRs as

$$U_t^{MUaR} = \beta\mathcal{M} - \gamma D_t - r_t S_t + r_{t-1} S_{t-1}, \tag{5}$$

where all the symbols have the same physical meanings as given in (4), except that the subscript t denotes any MUaR.

When the current MUaR node transmits data to the next MUaR nodes, we consider S_t to be the spectrum allocated to the current node by the next nodes. An optimal value of S_t ensures a successful transmission from the current node to the next nodes. Hence, we express the utility function of the current MUaR node, with respect to the allocated spectrum resource, as

$$U_t^{MUaR}(S_t) = \beta\mathcal{M} - \gamma D_t - r_t S_t. \tag{6}$$

Accordingly, the utility of the next MUaR nodes is the revenue obtained by allocating the spectrum to the current MUaR node. Hence, we formulate the utility function of the next MUaR nodes, with respect to the price per unit of the spectrum resources, r_t, as

$$U_{t+1}^{MUaR}(r_t) = r_t S_t. \tag{7}$$

We model the interactions between the MUaRs as an optimization problem, in order to maximize the utilities of the next MUaR nodes as in (7), while maximizing the utility of the current node as in (6). The optimization problem is expressed as

$$\max_{r_t} \; U_{t+1}^{MUaR}(r_t) = r_t S_t,$$
$$s.t. \begin{cases} r_t > 0, \\ \mathbf{S}_t = \arg\max \left(U_t^{MUaR}(\mathbf{S}_t) = \beta\mathcal{M} - \gamma D_t - r_t S_t \right), \\ s.t. \; S_t > 0, \end{cases} \tag{8}$$

for $t \in \{1, 2, \ldots, T\}$, where $\mathbf{S}_t$ is the vector containing the optimal values of S_t for the current MUaR to transmit data to the next MUaRs.

The values in $\mathbf{S}_t$ form the rest of the rewards in the $\mathbf{R}$ matrix, which are the reward values associated with all the possible next hops (MUaRs) for all the possible first hops (MUiCs), and then, the reward values associated with all the possible hops after that in the L-hop route, and so on. This helps generate the values in the $\mathbf{Q}$ matrix, which is the core of the Q-learning algorithm for the L-hop data transmission route calculation, and is discussed in detail in the next section.

Here, (8) is a two-level optimization problem. Such a hierarchical optimization problem that contains equilibrium constraints at two levels is called an EPEC [45], [46]. These equilibrium problems have equilibrium constraints at both the upper and lower levels. Hence, rather than just optimizing real-valued functions subject to equilibrium constraints, there exist equilibrium criteria at two levels. (8) is an optimization problem, where the MUaRs at two levels have their sets of equilibrium constraints. A centralized solution for all parties is difficult in such cases. We need a solution that can maximize the utility of the current MUaR node, while maximizing the utilities of the next MUaR nodes.

Stackelberg games can be employed in similar cases [47]. However, they work well only in scenarios with one leader and multiple followers. If we need to coordinate multiple conflicting utilities at both the levels, it might demand high complexity to provide optimal results. In addition, if we consider scenarios with a large number of entities, we need an algorithm that can converge for large networks. On that account, we consider ADMM for the optimization problem discussed above, for the VLC-D2D scenario. ADMM for EPEC is discussed in detail in the next section.

V. Algorithm Analysis

As mentioned before, Q-learning is a model-free technique used in RL scenarios, where the environment behavior is not fully known. In our scenario, the positions and data requirements of the mobile devices are random in nature and cannot be fully predicted. Therefore, Q-learning is a suitable tool to perform learning. Specifically, we propose to utilize the Q-learning technique to determine the L-hop data transmission routes from the VLC transmitters to the end MUiDs.

Here, the Q-learning based algorithm for optimal data transmission route determination is discussed in Section V-A, and the ADMM based EPEC algorithm for the determination of rewards is discussed in Section V-B.

A. Q-Learning for Route Selection

The general Q-learning algorithm works by evaluating $\mathbf{Q}$ values for different $(state, action)$ pairs, denoted as $\mathbf{Q}(S, A)$. In order to implement the Q-learning algorithm, we need the immediate reward values associated with various actions from different states, i.e., $\mathbf{R}$ values for different $(state, action)$ pairs, denoted as $\mathbf{R}(S, A)$. For recording these reward values, we create a rewards matrix, $\mathbf{R}$, as shown in Section III.

However, the reward values are not known beforehand, in our scenario. We compute the reward values dynamically during Q-learning, and record them in the $\mathbf{R}$ matrix. In our scenario, the VLC controller can implement the Q-learning algorithm, since it can act as an entity to co-ordinate and update the values of $\mathbf{R}(S, A)$ provided by the other entities.

Algorithm 1 RL Based Multi-Hop Route Selection Algorithm in VLC-D2D Heterogeneous Network

```
1: Initialization:
   i) Initialize the reward and Q-learning matrices:
   $\mathbf{R} = \mathbf{0}$, and $\mathbf{Q} = \mathbf{0}$.
   ii) Set the $\eta$ parameter, the maximum number of learning
   steps, $L_{max}$, the maximum number of hops, $L$, and the
   minimum threshold for the D2D transmission data rate $C_{th}$.
   iii) Set the current number of learning steps, $s = 0$, and
   the current number of hops, $l = 0$.
2: Generating the Q matrix:
3: while $s \leq L_{max}$ do
4:   Select an initial state for each agent as one of the $K$ VLC
     transmitters.
5:   while current state $\neq$ end MUiD do
6:     if current state = VLC transmitter then
7:       Compute $\mathbf{R}(S, A)$ using Algorithm 2 and update.
8:     else
9:       Compute $\mathbf{R}(S, A)$ using Algorithm 3 and update.
10:    end if
11:    Select one of all possible actions for the current state.
12:    Using this possible action, go to the next state.
13:    Get the maximum $\mathbf{Q}$ value for this next state based on
       the stored $\mathbf{Q}$ values for all possible actions.
14:    Compute $\mathbf{Q}(S, A)$ for the current state using (9) and
       update.
15:    Update current state = next state.
16:  end while
17:  $s = s + 1$.
18: end while
19: Utilizing the Q matrix to find the best route:
20: Set current state = initial state.
21: From current state, find the action with the highest $\mathbf{Q}$ value.
22: $l = l + 1$.
23: if $l > L$ or $C_{ij} < C_{th}$ then
24:   Routing failure, no available route.
25: end if
26: Set current state = next state.
27: Repeat steps 21 to 26 until current state = end MUiD.
28: Output $L$-hop transmission route.
```

As our Q-learning route selection algorithm runs, depending on the current state of the agent, we can have two different scenarios.

Current State = VLC Transmitter: At the start of the algorithm, the transmitted data will be at a VLC transmitter, and hence, the next state will be an MUiC. Therefore, we can compute the reward values using (1), which gives the capacity for VLC transmission between the VLC transmitter and the MUiC.

Current State = MUaR In our considered L-hop route, the first hops are MUiCs, and the rest of the hops are MUiDs acting as MUaRs. These interactions between the MUaRs are modeled in Section IV, using the EPEC in (8). In this case, we invoke the ADMM technique to solve this EPEC, which is discussed in detail in Section V-B. Thus, for the current state and action as MUaRs, we obtain and record the reward values.

Algorithm 2 Computation of Rewards for VLC

```
1: For VLC transmitter $\kappa$, for $\kappa \in \{1, 2, \ldots, K\}$, and MUiC
   $m$, for $m \in \{1, 2, \ldots, M\}$, compute the reward, $\mathbf{R}(S, A)$
   as
   $C_{\kappa m} = S_\kappa \log_2 \left(1 + \frac{P_\kappa G_{\kappa m}}{\sigma_I^2 + \sigma_N^2}\right)$, as in (1).
2: Output $\mathbf{R}(S, A)$.
```

Now, we add a similar matrix $\mathbf{Q}$, to represent the agent's learning through experience. As in the case of the $\mathbf{R}$ matrix, the rows of the $\mathbf{Q}$ matrix represent the current state of the agent, and the columns represent the actions. The agent begins the learning by knowing nothing, and hence, the $\mathbf{Q}$ matrix is initialized to $\mathbf{0}$.

The Q-learning route selection algorithm implemented by the VLC controller is described in detail in Algorithm 1. The algorithm has two main parts: generating the $\mathbf{Q}$ matrix, and utilizing the $\mathbf{Q}$ matrix to find the best data transmission route.

1) Generating the $\mathbf{Q}$ Matrix: The generation of the $\mathbf{Q}$ matrix is performed for a number of iterations, which we set as the maximum number of learning steps, as in the $while$ loop from steps 3-18. Initially, one of the K VLC transmitters is chosen to be the initial state of the agent. The $while$ loop in steps 5-16 runs for as long as the current state of the agent becomes the end MUiD. Within each learning step, the agent continues the learning until it reaches the end MUiD within the maximum number of hops.

If the current state of the agent is a VLC transmitter, then the VLC transmitter computes the $\mathbf{R}(S, A)$ values for the $\mathbf{R}$ matrix using Algorithm 2. If the current state of the agent is not a VLC transmitter, then the current state of the agent will be an MUaR, which results in an EPEC as in (8). The corresponding $\mathbf{R}(S, A)$ values are evaluated by the MUaRs, using the ADMM for EPEC algorithm, which is discussed in detail in Algorithm 3.

Once the $\mathbf{R}(S, A)$ values are evaluated, steps 11-15 are executed, which form the core of the learning part. The determination of the $\mathbf{Q}$-value for each $(state, action)$ pair, $\mathbf{Q}(S, A)$, for a particular $\mathbf{R}(S, A)$ in the reward matrix, is given by

$$\mathbf{Q}(S, A) = \mathbf{R}(S, A) + \eta * \max[\mathbf{Q}(S', \mathbf{A}')], \tag{9}$$

where S is the current state of the agent, A is the selected action for the current state, S' is the next state for state S, and $\mathbf{A}'$ is the action space of the agent in state S'. Here, η is the discount factor, and is the weight given to the future rewards. An η value closer to 0 indicates more preference given to immediate rewards, and if η is closer to 1, future rewards are considered with greater weight.

2) Utilizing the $\mathbf{Q}$ Matrix to Find the Best Route: Once the agent learns from its experience and obtains an optimized $\mathbf{Q}$ matrix from the above mentioned steps, we put this learning to our use. The remaining steps in the algorithm explain how the agent navigates through the obtained $\mathbf{Q}$ matrix, to find an optimum route. It starts from the initial state, which is a VLC transmitter, and finds the action with the highest $\mathbf{Q}$-value for this initial state, to be the next state. Now, for this next

Algorithm 3 Computation of Rewards for D2D Using ADMM for EPEC

```
Initialization:
Set the $\varepsilon$ parameter, and number of iterations, $p = 0$.
ADMM for EPEC between MUaRs:
while $\left\| \sum_{t=1}^{T_1} U_{t+1}^{MUaR}(r_t^{(p)}) - \sum_{t=1}^{T_1} U_{t+1}^{MUaR}(r_t^{(p-1)}) \right\| \geq \varepsilon$ do
    Optimization for the current MUaR node (inner loop):
    The current MUaR node uses $r_t$ to evaluate optimal $\mathbf{S}_t$, such that its utility is maximized.
    Optimization for the next MUaR nodes (outer loop):
    The next MUaR nodes predict the behavior of the current MUaR node, i.e., they deduce the optimal $\mathbf{S}_t$ obtained by the current MUaR node, and evaluate optimal $r_t$, such that their utilities are maximized.
    $p = p + 1$.
end while
For MUaR $i$, for $i \in \{1, 2, \ldots, T\}$, and MUaR $j$, for $j \in \{1, 2, \ldots, T\}$, $i \neq j$ (or between MUaR $i$, for $i \in \{1, 2, \ldots, T\}$, and end MUiD $j$, for $j \in \{1, 2, \ldots, N\}$), compute the reward, $\mathbf{R}(S, A)$ as
$C_{ij} = S_{ij} \log_2 \left(1 + \frac{P_{ij} G_{ij}}{I_c + N_0}\right)$, as in (2), using optimal values from $\mathbf{S}_t$ as $S_{ij}$.
Output $\mathbf{R}(S, A)$.
```

state, the action with the highest $\mathbf{Q}$-value is found, to be the next state. This process is continued until the current state of the agent becomes the end MUiD. Each time when the next state is updated, and thus, one more hop is added to the route, it is checked if the number of hops, l has not exceeded the maximum number of hops, L. This procedure results in an L-hop data transmission route from the VLC transmitter to the end MUiD.

Here, an important issue is regarding the choice of actions during learning. The agent can either choose an action to maximize its current $\mathbf{Q}$-value, or choose an action randomly from among all of its possible actions. The approach in which the agent chooses to maximize its current $\mathbf{Q}$-value is called the *greedy* approach [28]. The agent can be trapped in a local optimum in this case, and the solution is for the agent to explore other possible actions. In this paper, we face this issue by employing an ϵ-greedy method, where ϵ is a probability factor. The agent chooses the action with the maximum $\mathbf{Q}$-value with a probability of $1 - \epsilon$, and a random action with a probability of ϵ.

Next, we discuss the reward calculation procedure using the ADMM for EPEC algorithm.

B. ADMM for EPEC

In this subsection, we first discuss the concept of ADMM [48]. Then, we explain the iterated process of ADMM based EPEC algorithm, used to model the interactions between the MUaRs. Finally, we discuss the convergence of the proposed ADMM based EPEC algorithm.

1) ADMM: To demonstrate the ADMM, let us consider an example scenario with one current MUaR and T' next MUaRs, where the current MUaR wants to maximize its utility as

$$\max\ U_t^{MUaR}(\mathbf{S}_t) = \beta\mathcal{M} - \gamma D_t - r_t S_t, \quad s.t.\ S_t > 0, \tag{10}$$

where S_t is a real, scalar variable.

Here, the values of $\mathbf{S}_t$ can be updated by the current MUaR as

$$\mathbf{S}_{ti}(t+1) = \arg\max\big(U_t^{MUaR}(\mathbf{S}_{ti})\big) + \lambda_i(t)\mathbf{S}_{ti} + \Psi, \tag{11}$$

$\forall i \in \{1, 2, \ldots, T'\}$, and

$$\Psi = \frac{\rho}{2}\|\mathbf{S}_{ti}\|_2^2. \tag{12}$$

Here, $\rho > 0$ is a damping factor, t is the iteration step index, and $\|\cdot\|_2$ denotes the Frobenius norm [49]. λ is the dual variable, and it is updated as

$$\lambda_i(t+1) = \lambda_i(t) + \rho\,(\mathbf{S}_{ti}(t+1)). \tag{13}$$

Due to its quick convergence property, ADMM is used for large-scale optimization problems in large networks [49].

*2) ADMM for EPEC to Obtain **R**(S, A) for MUaRs:* In the Q-learning based route selection algorithm, if the current state of the agent is an MUaR, modeling the interaction between the current state (MUaR) and the next hop (MUaR) results in an EPEC as in (8). Each of the accessible next MUaRs provides spectrum, S_t, to the current MUaR, for which the current MUaR pays a price, r_t to the next MUaR node which it transmits to. The determination of the optimal values of $\mathbf{S}_t$ and r_t by the current and next MUaRs, respectively, forms the core of the ADMM based EPEC method in our scenario. It is an iterative process as shown in Algorithm 3, in which each iteration can be explained in two steps as given below:

a) Optimization problem of the current MUaR node: Initially, the next MUaR nodes announce the prices for the bandwidth that they are providing. The current MUaR uses the announced prices at the start of each iteration, p, to calculate the values in $\mathbf{S}_t$, the amount of spectrum to be purchased from each accessible next MUaR, to maximize its profit $U_t^{MUaR}(\mathbf{S}_t)$. Here, the superscript p denotes the value at the p^{th} iteration of the method, and the superscript $p-1$ denotes the value at the $p-1^{th}$ iteration of the method. This is the *inner loop* of ADMM. t is the iteration step index of the inner loop.

We described $\mathbf{S}_t$ in (8) as $\mathbf{S}_t = \arg\max\big(U_t^{MUaR}(\mathbf{S}_t)\big)$. For the current MUaR, maximizing its utility, $\big(U_t^{MUaR}(\mathbf{S}_t)\big)$ forms a set of values, $\mathbf{S}_t$. Hence, the values in $\mathbf{S}_t$ are updated at each iteration of the inner loop by the current MUaR as

$$\mathbf{S}_{t_i}^{(p)}(t+1) = \arg\max\ \big(U_t^{MUaR}(\mathbf{S}_{ti})\big) + \lambda_i^{(p)}(t)\mathbf{S}_{ti} + \Psi, \tag{14}$$

where

$$\Psi = \frac{\rho}{2}\|\mathbf{S}_{ti}\|_2^2, \tag{15}$$

where $\rho > 0$ is a damping factor as mentioned above, and λ is the dual variable, which is updated as

$$\lambda_i^{(p)}(t+1) = \lambda_i^{(p)}(t) + \rho\left(\mathbf{S}_{t_i}^{(p)}(t+1)\right). \tag{16}$$

At the end of the inner loop during each iteration, p, of the outer loop, the current MUaR arrives at a vector of S_t values, $\mathbf{S}_t$, which maximizes its utility. At the same time, these values are predicted by the next MUaRs, and are used to update the values of r_t.

b) Optimization problem of the next MUaR nodes: The next MUaRs are able to predict the behavior of the current MUaR and the values in $\mathbf{S}_t$. The next MUaRs then execute ADMM as

$$r_t(i)^{(p)}(t+1) = \arg\max\left(U_{t+1}^{MUaR}\left(r_t(i)\right)\right) + \lambda^{(p)}\left(t\right)r_t(i) + \Psi, \tag{17}$$

where

$$\Psi = \frac{\rho}{2}\|r_t(i)\|_2^2. \tag{18}$$

Here, $\rho > 0$ is the damping factor, and λ is the dual variable, which is updated as

$$\lambda^{(p)}\left(t+1\right) = \lambda^{(p)}\left(t\right) + \rho\left(r_t(i)^{(p)}\left(t+1\right)\right). \tag{19}$$

Thus, the next MUaRs recalculate the values of r_t that maximize their profits.

The updated values, ${r_t}^{(p+1)}$, are then provided to the current MUaR for the $(p+1)^{th}$ iteration. This is the *outer loop* of ADMM. The outer loop terminates when

$$\left\|\sum_{t=1}^{T_1} U_{t+1}^{MUaR}(r_t^{(p)}) - \sum_{t=1}^{T_1} U_{t+1}^{MUaR}(r_t^{(p-1)})\right\| < \varepsilon, \tag{20}$$

where T_1 is the number of next MUaR nodes, and ε is a predetermined threshold. ADMM algorithm for EPEC is shown in Algorithm 3.

3) Convergence: ADMM is a widely used large-scale optimization tool for both convex and nonconvex objective functions. From (8), the utility functions of the current and next MUaR nodes in our considered scenario, $U_t^{MUaR}(\mathbf{S}_t)$ and $U_{t+1}^{MUaR}(r_t)$, respectively, are linear. The convergence of ADMM in the case of nonconvex objective functions is discussed in [50]. ADMM is guaranteed to converge to the set of stationary solutions, in the case of nonconvex objective functions, if the penalty parameter, ρ, is chosen to be sufficiently large. In our scenario, ADMM converges to the optimal values of $\mathbf{S}_t$ and r_t, which simultaneously maximize the utilities for the current and next MUaR nodes, respectively. Reference [50] provides the detailed convergence analysis of nonconvex objective functions.

Lemma 1: The L-hop route for data transmission from the VLC transmitter to the end MUiD, obtained through the proposed RL based method is optimal.

Proof: In the Q-learning algorithm, the reward for the first hop in the L-hop route is the data rate for the communication between the VLC transmitter and the MUiC, as determined using Algorithm 2. According to the received $\mathcal{M}$ and the known values of r_t, the current MUaR calculates the optimal $\mathbf{S}_t$ that maximizes its utility, which is predicted by the next MUaRs, and is utilized in calculating the optimal values of r_t that maximize their utilities. This process between the MUaRs is repeated until the ADMM converges, as shown in Algorithm 3. The utilities of the MUaRs computed using the optimal values of $\mathbf{S}_t$ and r_t form the rewards for the next hops in the L-hop route. Since the utilities of the MUaRs in the L-hop route are obtained using the optimal values of $\mathbf{S}_t$ and r_t, none of the MUaRs can deviate from these optimal values for better utilities. Clearly, the L-hop route deduced by the Q-learning algorithm which consists of these MUaRs is optimal. ■

VI. Performance Evaluation

Here, we discuss the simulation results in detail in Section VI-A and discuss a few important aspects of the results in Section VI-B.

A. Simulation Results

In this section, we evaluate the performance of the proposed route selection algorithm using MATLAB. We consider a 5 $metre$ × 5 $metre$ room for the indoor VLC-D2D scenario. The number of VLC transmitters, K, and the number of mobile users, T, are varied to study different cases for evaluation purposes. The mobile users are placed randomly in the room, making some of them MUiCs and some of them MUiDs. The VLC transmitter(s) are placed equidistant along a non-diagonal line through the center of the room (if there is only one VLC transmitter, it is placed at the exact center of the room). Here, the spectrum for each VLC transmitter, S_κ is set as 1 GHz. The transmit power of each VLC transmitter, P_κ is set as 1 W, the typical output power of a white LED. Both the interference and noise for the VLC transmission, σ_I^2 and σ_N^2 are set to -20 dB. The interference and noise for the D2D transmission, I_c and N_0, are also set to -20 dB. The transmit power of each mobile user, P_{ij} is set as 300 mW. The data packet size, $\mathcal{M}$ is set as 1 Kb, and α is set to 1 s^{-1}. The values of β and γ are set to 10^6 b^{-1} and 10^6, respectively. We set the maximum number of hops for the data transmission route, L as 3. For Q-learning, we vary L_{max}, the maximum number of steps between 1 and 100, to study the effect of learning on different parameters. We set the default value of η as 0.8, to emphasize future rewards, and the greedy factor, ϵ is set as 0.1. The values of ε and ρ for ADMM are set as 10^{-6} and 1.5, respectively.

Fig. 4 shows the effect of Q-learning on the VLC transmission data rate for two cases: 1 VLC transmitter and 5 mobile users, and 2 VLC transmitters and 10 mobile users. The figure shows the transmission capacity for the VLC communication between the VLC transmitter and the MUiC. We can see that as the number of learning steps increases from 1 to 100, the VLC transmission (first hop) capacity from the VLC transmitter to the MUiC increases, which demonstrates the impact of RL on improving the VLC transmission capacity. Specifically, we observe that as the number of learning steps doubles from 50 to 100, the VLC data rate improves by around 16 % for the case with 2 VLC transmitters and 10 mobile users. This is due to a better $\mathbf{Q}$ matrix achieved via better learning through more interactions with the environment.

Fig. 5 shows the effect of Q-learning on the delay in VLC transmission for two cases: 1 VLC transmitter and 5 mobile

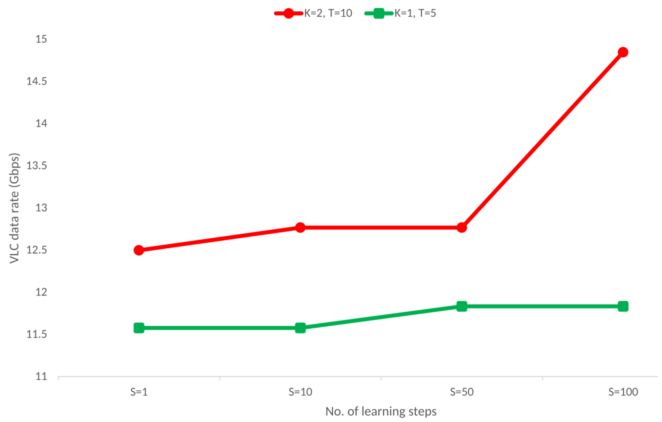

Fig. 4. Effect of Q-learning on VLC transmission data rate.

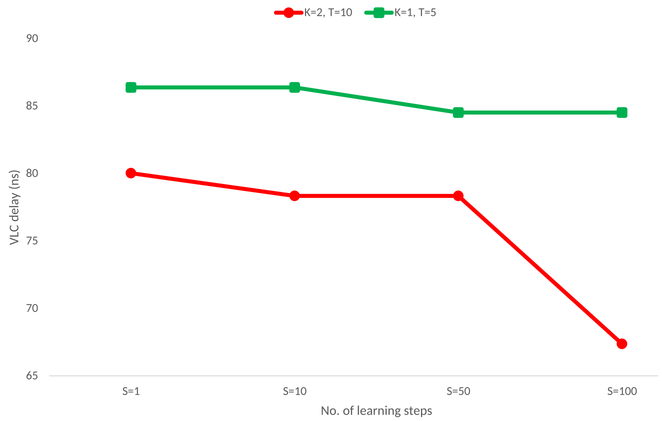

Fig. 5. Effect of Q-learning on VLC transmission delay.

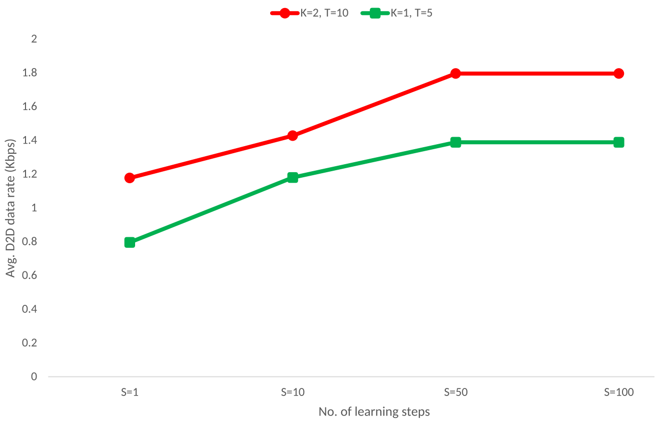

Fig. 6. Effect of Q-learning on D2D transmission data rate.

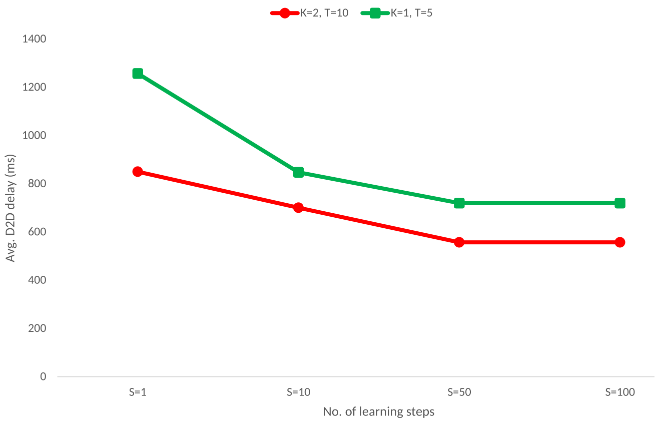

Fig. 7. Effect of Q-learning on D2D transmission delay.

users, and 2 VLC transmitters and 10 mobile users. The figure shows the delay for the VLC communication between the VLC transmitter and the MUiC. We can see that the VLC transmission (first hop) delay from the VLC transmitter to the MUiC decreases, as the number of learning steps increases from 1 to 100. This shows how RL helps in minimizing the VLC transmission delay in a VLC-D2D heterogeneous network. Here, we can see in particular that as the number of learning steps doubles from 50 to 100, the VLC delay decreases by around 14 % for the case with 2 VLC transmitters and 10 mobile users, which is also the result of better learning.

Fig. 6 shows the effect of Q-learning on the average D2D communication data rate for two cases: 1 VLC transmitter and 5 mobile users, and 2 VLC transmitters and 10 mobile users. The figure shows the average transmission capacity for the D2D communication between the MUaRs. We can see that as the number of learning steps increases from 1 to 100, the average D2D communication (second and third hops) data rate between the MUaRs increases. There is a significant increase in data rate (48%), when the number of learning steps increases from 1 to 10 for the case with 1 VLC transmitter and 5 mobile users. This shows the impact of RL on improving the D2D communication data rate in a VLC-D2D heterogeneous network.

Fig. 7 shows the effect of Q-learning on the average delay in D2D communication for two cases: 1 VLC transmitter and 5 mobile users, and 2 VLC transmitters and 10 mobile users. The figure shows the average delay for the D2D communication between the MUaRs. We can see that the average D2D communication (second and third hops) delay between the MUaRs decreases, as the number of learning steps increases from 1 to 100. The average D2D delay is decreased by around 33% in the case with 1 VLC transmitter and 5 mobile users, for an increase in the number of learning steps from 1 to 10. This shows the impact of RL in minimizing the D2D communication delay in a VLC-D2D heterogeneous network.

The effects of the discount factor for Q-learning, η, on the average D2D communication data rate and delay are shown in Fig. 8. The proposed method is compared with a centralized method where the data transmission route is determined by the VLCSP based only on the transmission path delays. The average D2D data rate and delay does not change much for the centralized allocation. It can be observed that the average capacity of D2D communication increases, and the average delay decreases as η increases for the proposed method. This indicates better performance as the emphasis on future rewards grows, which in turn underlines the benefits of learning from experience.

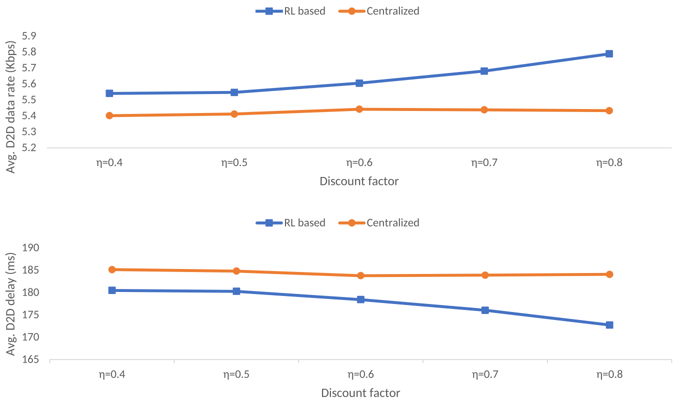

Fig. 8. Effect of discount factor on D2D parameters.

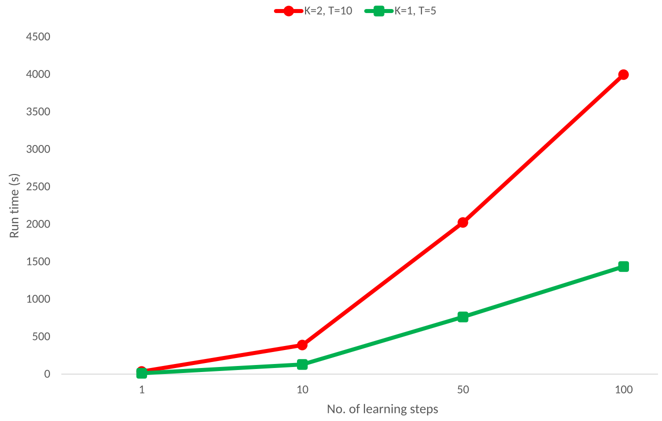

Fig. 9. Effect of Q-learning on overall algorithm run time.

The effects of Q-learning on the overall algorithm run time are shown in Fig. 9, for two cases: 1 VLC transmitter and 5 mobile users, and 2 VLC transmitters and 10 mobile users. It is obvious that as the number of learning steps increases, or as the number of entities increases, the algorithm run time will increase. Specifically, we can observe that the run time increases proportionally with the number of learning steps as well as the number of entities. Hence, the key observation here is that the time complexity of the proposed RL based multi-hop route determination algorithm using ADMM for EPEC, is linear.

B. Discussion

Here, as we consider an indoor communication scenario, the mobility speed of the cellular users is significantly lesser compared to the learning speed of the RL algorithm. Thousands of data packets are transmitted in each second, and this enables the RL algorithm to perform repeated interactions and arrive at an optimal L-hop route at a rate much faster than the changes in locations of the mobile users. The RL algorithm can be run to capture any changes in the network topology.

Also, the run times in Fig. 9 are obtained using a small-scale 16 Gb $Intel$ $Core$ $i7$ processor. The algorithm's run times can be improved tremendously by using a better processor, which would be the case in a practical VLC-D2D heterogeneous network. Accordingly, the algorithm can be run every few minutes or seconds in order for the RL part to accommodate the changes in users' locations, and optimal multi-hop routes can be determined dynamically. Also, in the considered indoor scenario, L can be as small as 2, with just an MUiC relaying the data from the VLC transmitter to the end MUiD. In this case, the RL algorithm for route determination converges even faster, dismissing the effects of user mobility.

VII. Conclusion

In this paper, we have proposed an RL based data transmission route determination method for an indoor VLC-D2D heterogeneous network. We have utilized the model-free Q-learning technique to determine L-hop data transmission routes from the VLC transmitters to the end MUiDs. We have determined the rewards for the Q-learning method dynamically during each learning step, by formulating the interactions between the MUaRs as an EPEC optimization problem, and then, solving it using ADMM. We have evaluated the performance of the proposed algorithm through MATLAB simulations. It can be observed from the simulations that RL improves the parameters of a typical VLC-D2D heterogeneous indoor downlink scenario. As the number of learning steps increases, the agent learns more through its interactions with the environment, and hence, the data transmission rate is improved, and the delay is minimized. We can observe from the simulation results that when the number of learning steps increases from 1 to 100, the VLC and D2D data rates are increased and the delays are reduced. It can also be noticed that the D2D data rate is improved and the delay is minimized as the discount factor for Q-learning increases, which highlights the importance of future rewards. The simulation results also demonstrated the time complexity of the proposed algorithm to be linear.

References

[1] "Cisco visual networking index: Global mobile data traffic forecast update, 2016–2021," Cisco Syst., San Jose, CA, White Paper 1454457600805266, 2017. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/mobile-white-paper-c11-520862.html

[2] C. Pohlmann. *Visible Light Communication*. Accessed: 2010. [Online]. Available: https://pdfs.semanticscholar.org/cd51/c741f013885834a5d05df6abf4213399f3e7.pdf

[3] T. Komine and M. Nakagawa, "Fundamental analysis for visible-light communication system using LED lights," *IEEE Trans. Consum. Electron.*, vol. 50, no. 1, pp. 100–107, Feb. 2004.

[4] E. T. Won *et al. Visible Light Communication: Tutorial*. Accessed: 2008. [Online]. Available: http://www.ieee802.org/802_tutorials/2008-03/15-08-0114-02-0000-VLC_Tutorial_MCO_Samsung-VLCC-Oxford_2008-03-17.pdf

[5] A. Vavoulas, H. G. Sandalidis, T. A. Tsiftsis, and N. Vaiopoulos, "Coverage aspects of indoor VLC networks," *J. Lightw. Technol.*, vol. 33, no. 23, pp. 4915–4921, Dec. 1, 2015.

[6] H. Zhang, W. Ding, J. Song, and Z. Han, "A hierarchical game approach for visible light communication and D2D heterogeneous network," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Washington, DC, USA, Dec. 2016, pp. 1–6.

[7] P. Gandotra, R. K. Jha, and S. Jain, "A survey on device-to-device (D2D) communication: Architecture and security issues," *J. Netw. Comput. Appl.*, vol. 78, pp. 9–29, Jan. 2017.

[8] D. Silver. *Introduction to Reinforcement Learning*. Accessed: 2015. [Online]. Available: http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching_files/intro_RL.pdf

[9] D. Silver. *Deep Reinforcement Learning*. Accessed: 2015. [Online]. Available: http://www0.cs.ucl.ac.uk/staff/d.silver/web/Resources_files/deep_rl.pdf

[10] L. P. Kaelbling, M. L. Littman, and A. W. Moore, "Reinforcement learning: A survey," *J. Artif. Intell. Res.*, vol. 4, p. 237–285, May 1996.

[11] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.

[12] M. L. Littman and C. Szepesvári, "A generalized reinforcement-learning model: Convergence and applications," in *Proc. 13th Int. Conf. Mach. Learn.*, Feb. 1996, pp. 310–318.

[13] T. Z. Oo, N. H. Tran, W. Saad, D. Niyato, Z. Han, and C. S. Hong, "Offloading in HetNet: A coordination of interference mitigation, user association, and resource allocation," *IEEE Trans. Mobile Comput.*, vol. 16, no. 8, pp. 2276–2291, Aug. 2017.

[14] M. Ali, S. Mumtaz, S. Qaisar, and M. Naeem, "Smart heterogeneous networks: A 5G paradigm," *Telecommun. Syst.*, vol. 66, no. 2, pp. 311–330, Oct. 2017.

[15] L. Song, D. Niyato, Z. Han, and E. Hossain, "Game-theoretic resource allocation methods for device-to-device communication," *IEEE Wireless Commun.*, vol. 21, no. 3, pp. 136–144, Jun. 2014.

[16] A. Abrardo, G. Fodor, and B. Tola, "Network coding schemes for device-to-device communications based relaying for cellular coverage extension," in *Proc. IEEE 16th Int. Workshop Signal Process. Adv. Wireless Commun. (SPAWC)*, Stockholm, Sweden, Jun./Jul. 2015, pp. 670–674.

[17] M. Ali, S. Qaisar, M. Naeem, and S. Mumtaz, "Energy efficient resource allocation in D2D-assisted heterogeneous networks with relays," *IEEE Access*, vol. 4, pp. 4902–4911, 2016.

[18] R. Liu and C. Zhang, "Dynamic dwell timer for vertical handover in VLC-WLAN heterogeneous networks," in *Proc. 13th Int. Wireless Commun. Mobile Comput. Conf. (IWCMC)*, Valencia, Spain, Jun. 2017, pp. 1256–1260.

[19] M. S. Saud, H. Chowdhury, and M. Katz, "Heterogeneous software-defined networks: Implementation of a hybrid radio-optical wireless network," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, San Francisco, CA, USA, Mar. 2017, pp. 1–6.

[20] M. S. Saud and M. Katz, "Implementation of a hybrid optical-RF wireless network with fast network handover," in *Proc. Eur. Wireless 23th Eur. Wireless Conf.*, Dresden, Germany, May 2017, pp. 1–6.

[21] W. O. Popoola, E. Pikasis, and I. Osahon, "Hybrid polymer optical fibre and visible light communication link for in-home network," in *Proc. 26th Wireless Opt. Commun. Conf. (WOCC)*, Newark, NJ, USA, Apr. 2017, pp. 1–6.

[22] C.-X. Wang *et al.*, "Cellular architecture and key technologies for 5G wireless communication networks," *IEEE Commun. Mag.*, vol. 52, no. 2, pp. 122–130, Feb. 2014.

[23] R. Zhang, J. Wang, Z. Wang, Z. Xu, C. Zhao, and L. Hanzo, "Visible light communications in heterogeneous networks: Paving the way for user-centric design," *IEEE Wireless Commun.*, vol. 22, no. 2, pp. 8–16, Apr. 2015.

[24] X. Bao, J. Dai, and X. Zhu, "Visible light communications heterogeneous network (VLC-HetNet): New model and protocols for mobile scenario," *Wireless Netw.*, vol. 23, no. 1, pp. 299–309, Jan. 2017.

[25] M. B. Rahaim, A. M. Vegni, and T. D. C. Little, "A hybrid radio frequency and broadcast visible light communication system," in *Proc. IEEE GLOBECOM Workshops (GC Wkshps)*, Houston, TX, USA, Dec. 2011, pp. 792–796.

[26] 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.

[27] F. Jin, R. Zhang, and L. Hanzo, "Resource allocation under delay-guarantee constraints for heterogeneous visible-light and RF femtocell," *IEEE Trans. Wireless Commun.*, vol. 14, no. 2, pp. 1020–1034, Feb. 2015.

[28] J. Hu and M. P. Wellman, "Multiagent reinforcement learning: Theoretical framework and an algorithm," in *Proc. 15th Int. Conf. Mach. Learn.*, Jul. 1998, pp. 242–250.

[29] X. Xu, L. Zuo, and Z. Huang, "Reinforcement learning algorithms with function approximation: Recent advances and applications," *Inf. Sci.*, vol. 261, pp. 1–31, Mar. 2014.

[30] DeepMind. *AlphaGo*. Accessed: 2016. [Online]. Available: https://deepmind.com/research/alphago/

[31] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1238–1274, Aug. 2013.

[32] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proc. 11th Int. Conf. Mach. Learn.*, New Brunswick, NJ, USA, Jul. 1994, pp. 157–163.

[33] M. Zolfpour-Arokhlo, A. Selamat, S. Z. M. Hashim, and H. Afkhami, "Modeling of route planning system based on Q value-based dynamic programming with multi-agent reinforcement learning algorithms," *Eng. Appl. Artif. Intell.*, vol. 29, p. 163–177, Mar. 2014.

[34] M. A. Aref, S. K. Jayaweera, and S. Machuzak, "Multi-agent reinforcement learning based cognitive anti-jamming," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, San Francisco, CA, USA, Mar. 2017, pp. 1–6.

[35] M. A. Aref, S. Machuzak, S. K. Jayaweera, and S. Lane, "Replicated Q-learning based sub-band selection for wideband spectrum sensing in cognitive radios," in *Proc. IEEE/CIC Int. Conf. Commun. China (ICCC)*, Chengdu, China, Jul. 2016, pp. 1–6.

[36] R. Ghasemaghaei, M. A. Rahman, W. Gueaieb, and A. El Saddik, "Ant colony-based reinforcement learning algorithm for routing in wireless sensor networks," in *Proc. IEEE Instrum. Meas. Technol. Conf.*, Warsaw, Poland, May 2007, pp. 1–6.

[37] M. Hasan, E. Hossain, and D. Niyato, "Random access for machine-to-machine communication in LTE-Advanced networks: Issues and approaches," *IEEE Commun. Mag.*, vol. 51, no. 6, pp. 86–93, Jun. 2013.

[38] M. Lee, D. Marconett, X. Ye, and S. J. B. Yoo, "Cognitive network management with reinforcement learning for wireless mesh networks," in *Proc. 7th IEEE Int. Workshop IP Oper. Manage.*, San Jose, CA, USA, Oct. 2007, pp. 168–179.

[39] Z. Liu and I. Elhanany, "RL-MAC: A QoS-aware reinforcement learning based MAC protocol for wireless sensor networks," in *Proc. IEEE Int. Conf. Netw., Sens. Control*, Apr. 2006, pp. 768–773.

[40] A. R. Syed, K.-L. A. Yau, J. Qadir, H. Mohamad, N. Ramli, and S. L. Keoh, "Route selection for multi-hop cognitive radio networks using reinforcement learning: An experimental study," *IEEE Access*, vol. 4, pp. 6304–6324, 2016.

[41] L. Giupponi, R. Agusti, J. Pérez-Romero, and O. Sallent, "A novel joint radio resource management approach with reinforcement learning mechanisms," in *Proc. 24th IEEE Int. Perform., Comput., Commun. Conf. (IPCCC)*, Phoenix, AZ, USA, Apr. 2005, pp. 621–626.

[42] R. S. H. Istepanian, N. Y. Philip, and M. G. Martini, "Medical QoS provision based on reinforcement learning in ultrasound streaming over 3.5G wireless systems," *IEEE J. Sel. Areas Commun.*, vol. 27, no. 4, pp. 566–574, May 2009.

[43] N. Mastronarde and M. van der Schaar, "Fast reinforcement learning for energy-efficient wireless communication," *IEEE Trans. Signal Process.*, vol. 59, no. 12, pp. 6262–6266, Dec. 2011.

[44] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Mach. Learn.*, vol. 8, nos. 3–4, pp. 279–292, May 1992.

[45] B. S. Mordukhovich, "Equilibrium problems with equilibrium constraints via multiobjective optimization," *Optim. Methods Softw.*, vol. 19, no. 5, pp. 479–492, Oct. 2004.

[46] S. Leyffer and T. Munson, "Solving multi-leader–common-follower games," *Optim. Methods Softw.*, vol. 25, no. 4, pp. 601–623, Aug. 2010.

[47] Z. Han, D. Niyato, W. Saad, T. Basar, and A. Hjorungnes, *Game Theory in Wireless and Communication Networks: Theory, Models, and Applications*. Cambridge, U.K.: Cambridge Univ. Press, 2011.

[48] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed optimization and statistical learning via the alternating direction method of multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jan. 2011.

[49] Z. Zheng, L. Song, and Z. Han, "Bridge the gap between ADMM and Stackelberg game: Incentive mechanism design for big data networks," *IEEE Signal Process. Lett.*, vol. 24, no. 2, pp. 191–195, Feb. 2017.

[50] M. Hong, Z.-Q. Luo, and M. Razaviyayn, "Convergence analysis of alternating direction method of multipliers for a family of nonconvex problems," *SIAM J. Optim.*, vol. 26, no. 1, pp. 337–364, Jan. 2016.

Neetu Raveendran (M'17) received the bachelor's degree in electronics and communication engineering from the College of Engineering, Trivandrum, India, in 2011, and the master's degree in electrical engineering from the University of Houston, Houston, TX, USA, in 2016, where she is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering. Her research interests include wireless networking, game theory, and deep learning.

Huaqing Zhang (M'14) received the B.S. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2013, and the Ph.D. degree from the Department of Electronic and Computer Engineering, University of Houston, Houston, TX, USA, in 2017. He is currently a Machine Learning Engineer with Petuum, Inc. His research interests include wireless communications and networking, zero-determinant strategy, hierarchical game theory, multi-agent reinforcement learning, and deep reinforcement learning.

Dusit Niyato (M'09–SM'15–F'17) received the B.Eng. degree from the King Mongkuts Institute of Technology Ladkrabang, Thailand, in 1999, and the Ph.D. degree in electrical and computer engineering from the University of Manitoba, Canada, in 2008. He is currently a Professor with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. His research interests are in the areas of energy harvesting for wireless communication, the Internet of Things, and sensor networks.

Fang Yang (M'11–SM'13) received the B.S.E. and Ph.D. degrees in electronic engineering from Tsinghua University, Beijing, China, in 2005 and 2009, respectively. He is currently an Associate Professor with the Department of Electronics Engineering, Tsinghua University. He has published over 120 peer-reviewed journal and conference papers. He holds over 40 Chinese patents and two PCT patents. His research interests lie in the fields of channel coding, channel estimation, interference cancellation, and signal processing techniques for communication system, especially in power line communication, visible light communication, and digital television terrestrial broadcasting. He received the IEEE Scott Helt Memorial Award (Best Paper Award from the IEEE TRANSACTIONS ON BROADCASTING) in 2015. He is the Secretary General of Sub-Committee 25 of the China National Information Technology Standardization (SAC/TC28/SC25). He currently serves as an Associate Editor for the IEEE ACCESS.

Jian Song (M'06–SM'10–F'16) received the B.Eng. and Ph.D. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1990 and 1995, respectively, and worked for the same university upon his graduation. He was with The Chinese University of Hong Kong and the University of Waterloo, Canada, in 1996 and 1997, respectively. He was with Hughes Network Systems, USA, for seven years, before joining the Faculty Team, Tsinghua University, in 2005, as a Professor. He is currently the Director of the Tsinghua University's DTV Technology R&D center. He has been working in different areas of fiber-optic, satellite and wireless communications, and the power line communications. He has published over 200 peer-reviewed journal and conference papers. He holds two U.S. and over 40 Chinese patents. His current research interest is in the area of digital TV broadcasting. He is a fellow of the IEEE and IET.

Zhu Han (S'01–M'04–SM'09–F'14) received the B.S. degree in electronic engineering from Tsinghua University, in 1997, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Maryland at College Park, College Park, MD, USA, in 1999 and 2003, respectively.

From 2000 to 2002, he was an R&D Engineer of JDSU, Germantown, MD, USA. From 2003 to 2006, he was a Research Associate with the University of Maryland. From 2006 to 2008, he was an Assistant Professor with Boise State University, Boise, ID, USA. He is currently a Professor with the Electrical and Computer Engineering Department and with the Computer Science Department, University of Houston, Houston, TX, USA. His research interests include wireless resource allocation and management, wireless communications and networking, game theory, big data analysis, security, and smart grid. He received the NSF Career Award in 2010, the Fred W. Ellersick Prize of the IEEE Communication Society in 2011, the EURASIP Best Paper Award for the *Journal on Advances in Signal Processing* in 2015, the IEEE Leonard G. Abraham Prize in the field of communications systems (Best Paper Award in IEEE JSAC) in 2016, and several best paper awards in IEEE conferences. He is currently an IEEE Communications Society Distinguished Lecturer.