Multi-Agent Deep Reinforcement Learning for Spectral Efficiency Optimization in Vehicular Optical Camera Communications

Amirul Islam, Nikolaos Thomos, and Leila Musavian

Abstract—In this paper, we propose a vehicular optical camera communication system that can meet low bit error rate (BER) and ultra-low latency constraints. First, we formulate a sum spectral efficiency optimization problem that aims at finding the speed of vehicles and the modulation order that maximizes the sum spectral efficiency subject to reliability and latency constraints. This problem is mixed-integer programming with nonlinear constraints, and even for a small set of modulation orders, is NP-hard. To overcome the entailed high computational and time complexity which prevents its solution with traditional methods, we first model the optimization problem as a partially observable Markov decision process. We then solve it using an independent Q-learning framework, where each vehicle acts as an independent agent. Since the state-action space is large we then adopt deep reinforcement learning (DRL) to solve it efficiently. As the problem is constrained, we employ the Lagrange relaxation approach prior to solving it using the DRL framework. Simulation results demonstrate that the proposed DRL-based optimization scheme can effectively learn how to maximize the sum spectral efficiency while satisfying the BER and ultra-low latency constraints. The evaluation further shows that our scheme can achieve superior performance compared to radio frequency-based vehicular communication systems and other vehicular OCC variants of our scheme.

Index Terms—Vehicular communication, deep reinforcement learning, optical camera communication, spectral efficiency maximization, Lagrangian relaxation, low latency

1 Introduction

1.1 Background and Motivation

Vehicular communications are considered a key transforming technology for enhancing intelligent transportation systems (ITS) and overall road safety by exploiting communication between vehicles. Every year, data sharing within vehicular networks is continuously increasing, thus incurring enormous network overhead [1] [2]. As a result, the currently congested and saturated radio frequency (RF) spectrum cannot accommodate the ever-increasing data traffic. Recently, optical camera communication (OCC) has emerged as a potential technology for ITS [3], [4] and as an alternative to RF because of its license-free unlimited spectrum, longer lifespans, lower implementation cost, lower power consumption, and enhanced security [5]. OCC systems belong to the family of visible light communication (VLC) systems. In typical OCC systems, light-emitting diodes (LEDs) are typically used as transmitters, and cameras are employed as receivers. OCC overcomes the interference problems of conventional RF-based communication systems because it offers line-of-sight (LoS) and directed communication [6].

Another challenge of vehicular networks arises from the fact that they are highly dynamic, while the vast amounts of generated data (each vehicle can generate up to 750 MB per second) should be delivered reliably within stringent time constraints to ensure safety. In recent years, various technologies have been proposed in ITS using traditional optimization methods, such as [7], [8] reflecting on delay minimization and reliability guarantee to meet reliability and latency requirements for static or slow-changing vehicular networks. However, they cannot support ultra-reliability and low latency conditions because of the additional delays that are introduced due to the need to communicate with servers and the extremely high complexity of the studied problems as different decision variables are simultaneously controlled. In [7], the vehicular network transmission power is minimized by grouping vehicles into clusters and modelling reliability as queuing delay violation probability. In [8], a joint resource allocation and power control technique is suggested to maximize the communication rate considering latency and reliability constraints. However, the complexity of the system makes it difficult to guarantee reliability and low latency. The complexity arises when decision-making involves simultaneous control of various parameters, such as speed, distance, and modulation scheme. Further, these schemes involve centralized communication with the base station, servers or roadside units, where the complex problems are solved, which introduces additional latency and makes it even more difficult to meet reliability and low latency requirements. Due to their intrinsic complexity and the time required to solve them, these decision-making problems cannot be solved using conventional distributed optimization methods. Fortunately, reinforcement learning (RL) approaches can serve as an effective alternative solution to overcome the complexity of such systems [9] because they can be applied

The authors are with the School of Computer Science and Electronic Engineering, University of Essex, UK. E-mail: amirul.0903118@gmail.com, nthomos@essex.ac.uk, leila.musavian@essex.ac.uk.

distributively.

In this paper, we propose an OCC-based vehicular communication system to maximize the sum spectral efficiency while satisfying the latency requirements and respecting reliability constraints. The studied problem can be modelled as a Markov decision process (MDP). Although MDP provides an efficient way to express our framework, traditional methods used to solve them, such as value-iteration, require knowledge of the transition probability matrix that is difficult to obtain in dynamic problems, such as the one we examine in this paper. These limitations can be overcome through Q-Learning [9]. However, Q-Learning has slow convergence and cannot solve large-scale problems. To address this limitation of the Q-Learning algorithm, we use the deep RL (DRL). DRL approximates the state-action value function by adjusting the weights of the employed deep neural network.

Even though DRL has improved the scalability of RL, training a centralized RL agent is still infeasible for large-scale vehicle-to-vehicle (V2V) environments as the one considered in this paper. This is due to the fact that we need to collect all the observation states from the vehicular network and communicate them to an agent (e.g., base station), which optimizes the policies of all the vehicles centrally. After determining the policies, the central agent should communicate them back to the vehicles. This centralized decision-making is problematic as it causes higher latencies due to communicating data back and forth, worsens congestion in the network, and may lead to inefficient policies, particularly when the information is lost or delayed. Furthermore, the joint state-action space of agents grows exponentially with an increase in the number of vehicles. To avoid the above problems, we formulate the problem as a multi-agent RL (MARL), where each agent considers only local observations and does not require global communication with a central agent. In particular, we adopt independent Q-Learning [10], in which each local agent learns its policy independently by modelling other agents as part of the environment. It has been shown that independent Q-Learning can lead to well-performing solutions though there are no theoretical guarantees [11].

1.2 Our Contributions

In this paper, we propose a sum spectral efficiency maximization scheme in vehicular OCC that satisfies bit error rate (BER) and latency constraints. In doing so, we determine the optimal modulation order and speed of the vehicles using DRL. We consider a decentralized, independent and MARL scheme in solving this problem. To the best of our knowledge, we maximize sum spectral efficiency by optimizing speed and modulation in this paper by applying DRL for the first time in vehicular OCC. The major contributions of this paper are summarized as follows:

- We propose a multi-vehicular deep reinforcement learning-based approach for maximizing sum spectral efficiency in vehicular OCC;
- We formulate the maximization problem under BER and latency constraints considering a small set of modulation schemes. As the optimization function is a non-deterministic polynomial-time (NP) hard problem leading to a difficult search for the optimal solution, we model the problem as a partially observable MDP (POMDP). We design the reward function to satisfy systems' requirements;
- We transform the considered constrained problem into an unconstrained one following the Lagrangian relaxation method. This essentially simplifies the solution of the complex problem. We, then, adopt an independent learning framework and solve the sum spectral efficiency maximization problem using deep Q-Learning;
- Finally, we evaluate the performance of the proposed DRL-based optimization scheme through simulations. The results show that DRL-based optimization algorithm can effectively learn to maximize the sum spectral efficiency while meeting the constraints. Additionally, our comparisons demonstrate that our scheme significantly outperforms other communication technologies, i.e., RF systems as well as variants of our scheme.

The remainder of this paper is organized as follows. We present the vehicular OCC system model with different performance parameters in Section 3. Section 4 outlines the proposed constrained problem formulation and RL-based MDP formulation. The solution to the multi-agent problem using deep Q-Learning is presented in Section 5. The simulation setup and training procedure for the proposed DRL algorithm are explained in Section 6. Section 7 presents simulation results using the proposed DRL-based optimization scheme. Finally, concluding remarks are drawn in Section 8.

2 Related Works

The capabilities, potentials, and advantages of OCC systems have already been surveyed in [3], [5], [6], [12]. The existing works mainly target to increase the data rate, but they do not consider the ultra-low latency and BER constraints that we consider here. Based on variation in LED light intensity, a flag image was generated via communication pixels with a 10 Mbps data rate [6]. In [5], the authors proposed an image sensor-based VLC system, which achieved a 20 Mb/s/pixel data rate without LED detection and 15 Mb/s/pixel data rate with 16.6 ms real-time LED detection. In [3], the data rate was further improved to 55 Mbps, where they achieved a BER $< 10^{-5}$. The BER performance of OCC systems was investigated in [12], which experimentally demonstrated that the proposed system can deliver a data rate of 150 bits/s across a range of up to 60 m using a single commercially available RGB LED and a 50-frame/s camera. The above-mentioned schemes demonstrated the great potential of OCC systems; however, they did not consider mobility and they were not optimized considering reliability and latency constraints.

DRL has already been applied to solve various resource allocation problems [13]–[16] in vehicular networks to address their time-varying nature, making it impossible to use traditional optimization methods. In [13], a deep reinforcement learning framework was developed for spectral sharing in an RF-based centrally optimized system where each V2V link acted as an agent. In [13], the agents collectively interacted with the communication environment, received a common reward, and learned to improve spectrum and power allocation through MARL. The authors in [14] employ a centralized decision-making approach and distributed channel allocation using multi-agent deep reinforcement learning to maximize spectrum efficiency in vehicular networks. However, they did not analyze the system's reliability and latency performance. In [15], the authors addressed a resource provisioning problem in vehicular clouds to dynamically meet resource demands and stringent quality of service (QoS) requirements by minimizing overhead of reprocessing and the types and amount of resources required. The authors in [16] studied a transmission delay minimization problem in software-defined vehicular networks, where the problem was formulated as a POMDP and solved with an online distributed learning algorithm.

The above-mentioned schemes face interference issues as they use RF technology. In such systems, interference or collision between signals occurs when multiple vehicles exist in close vicinity. In RF systems, several techniques, such as frequency planning methods or machine learning-based interference cancellation approaches, can be employed to mitigate the encountered interference. Although these methods are effective, they have high computational complexity, making it challenging to reliably find the optimal solutions. Different from these approaches, our scheme is based on vehicular OCC which overcomes interference problems as it can spatially separate and process different transmitter sources independently on its image plane [17]. This happens because it has millions of pixels, which provides the freedom to handle multiple users. Moreover, previous RF-based systems did not consider reliability and latency constraints concurrently, which makes it difficult to guarantee that the information is received reliably within the shortest possible time.

To the best of our knowledge, optimizing the performance of vehicular OCC using DRL has not yet been investigated in the literature. Recently, several studies suggested the use of RL in hybrid RF and photodiodes (PD)-based VLC networks [18]–[20]. Specifically, RL for network selection considering the traffic type and the possibility of having learning records to improve the Q-Learning algorithm was applied in [18]. An RL-based energy-efficient resource management scheme to improve energy efficiency was proposed in [19] and in [20], the authors implemented MARL to determine online power allocation to enhance the user's QoS. The above-stated systems use traditional Q-Learning, which is not suitable for high-dimensional problems. More importantly, they did not consider any reliability and low latency requirements. Moreover, they used PD-based receivers, which face interference problems when dealing with multiple vehicles.

3 Proposed Vehicular OCC Modelling

In this section, we first present the considered vehicular OCC system model. We then specify the performance-defining metrics of the OCC in terms of the BER, achievable rate, and observed transmission latency.

3.1 System Model

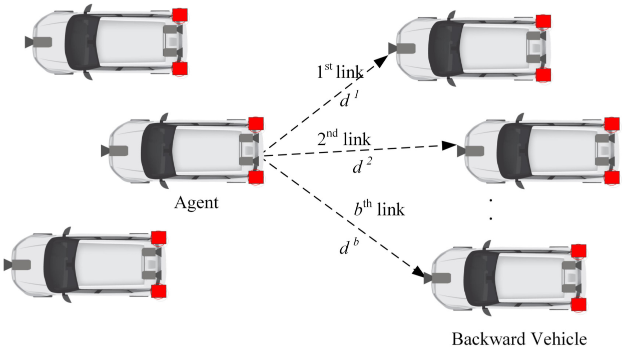

Fig. 1: Proposed system model for vehicular optical camera communication.

We consider a vehicular OCC system as shown in Fig. 1, where each vehicle acts as an individual agent. Each vehicle is equipped with a transmitting unit at the back consisting of LEDs backlights and a vision camera set and a receiving unit at the front having a high-speed camera (1000 frames per second (fps)).[1] The function of the camera in the front is twofold. First, it measures the forward distance. Second, the camera acts as a receiver that decodes the transmitted data from the LED transmitters of the front vehicle. The camera at the back measures the backward distance using a stereo-vision camera similarly to [21]. As shown in Fig. 1, each vehicle receives information from the front vehicle and transmits information to the backward vehicles. This information may include the vehicle's moving intentions (for example, braking, accelerating, changing lanes), emergency information, and so on. Let B be the number of V2V links at the back of each vehicle, where $\mathcal{B} = \{1, 2, \cdots B\}$ is the set of V2V links. We express the distance with the backward vehicles as d^b where $b \in \mathcal{B}$ and b represents the index of the backward V2V link. In our system, the number of vehicles that each vehicle interacts with equals the number of vehicles with which the vehicle can establish LoS links. This number implicitly depends on the density and the decisions of each vehicle to accelerate (and maybe overtake) or brake.

Our system employs an adaptive modulation scheme that consists of M-ary quadrature amplitude modulation (M-QAM) and time division multiple access (TDMA). The transmitter contains arrays of LEDs that transmit at different rates to different users under the adaptive modulation scheme. To support transmission at different modulation orders for different backward vehicles (links), we introduce TDMA into our system, similar to [22], where specific time slots are assigned to each vehicular link at the back. In this way, different time slots are allocated to each V2V link for either transmission or reception. However, since each link of the vehicle transmits information only at specific times, the sum spectral efficiency is divided by the number of available vehicles, B, at the back.

M-QAM is used to modulate the signal in VLC [23]. M-QAM is relatively easy to implement and offers very low BER, high-speed, and flicker-free communication [12]. At the transmitter, the data bit-streams are first mapped into symbols by splitting the amplitude and phase into in-phase and quadrature components, respectively. The resulting

[1] If cameras of low frame rate were used, e.g., 30 fps, the data rate per pixel would be limited to 15 bits per second (bps) or less to satisfy the Nyquist frequency requirement, which is low for the considered applications. Therefore, high-speed cameras should be utilized in the receiver systems to achieve higher data rates or receive high-speed optical signals.

signal is then transmitted through the optical channel by modulating the LED intensity. At the receiver, the camera captures the modulated light waveform within its exposure time. During this time, the image sensor captures the intensity of the light coming in as different LED states, e.g., on, off, and mid states. The original signal information can be extracted from the detected intensity in these pixels using an efficient M-QAM demodulation scheme [24]. In [24], a simple mathematical formulation for encoding and decoding the amplitude and phase of transmitted symbols was proposed, where the modulated symbol is sampled in three consecutive image frames by the image sensor. The LED states are identified from the captured frames and a lookup table is developed. The phase position is then retrieved using the lookup table, and the reconstructed phase is converted to radians so that it can be mapped to M-QAM. Finally, the original signal is perfectly recovered from the detected amplitude and phase.

3.2 Optical Channel Model

The transmitter and the camera receiver are assumed to have a continuous LoS link, guaranteeing that the vehicles can continuously communicate with one another without interference. OCC has a flat-fading or diffuse channel, depending on the channel characteristics. In general, the OCC channel contains two different types of light propagation components: (i) diffuse components arising from lights being reflected off of other cars or reflective surfaces, and (ii) LoS components resulting from direct light propagation from the transmitter to the receiver. The diffuse light component is neglected in this paper since it typically has far less energy than the LoS component.

The LEDs follow a Lambertian radiation pattern and can be modelled using a generalized Lambertian radiant intensity [25]. According to [25], [26], the channel gain, H_t^b, of link b in time t between the transmitter and the receiver is given by

$$H_t^b = \begin{cases} \frac{(m+1)A}{2\pi (d_t^b)^2} \ \cos^m(\phi) \ T_s(\theta) \ g \ \cos(\theta), & 0 \leq \theta \leq \theta_l \\ 0, & \theta > \theta_l \end{cases} \tag{1}$$

where m is the order of the Lambertian radiation pattern, A is the area of the entrance pupil of the camera lens, d_t^b is the agent's distance from the backward vehicles at time t, ϕ is the angle of irradiance with respect to the emitter, $T_s(\theta)$ is the transmission efficiency of the optical filter, g is the gain of the lens, θ is the angle of incidence (AoI) with respect to the receiver axis, and θ_l denotes the field of view (FoV) of the image sensor lens. m is derived as $m = \frac{-\ln(2)}{\ln(\cos(\Phi_{1/2}))}$, where $\Phi_{1/2}$ is the LED semi-angle at half luminance. In this paper, we consider fixed AoI, θ, following the analysis of our previous conference paper [27]. In particular, we fix the AoI to 60°, which can help our system to meet the latency and reliability requirements. We made this design decision as it is challenging to change the AoI continuously in a practical scenario because it would introduce additional delays caused by the need to change the AoI mechanically in the vehicle.

We would like to note that in our paper, we neglect the signal detection overhead of recognizing light sources under mobile scenarios inspired by [28], where authors proposed a statistical vehicle motion model in an image plane. In this paper, the authors showed that the vehicle motion along the vertical and horizontal axes of the image plane is limited to one pixel in most cases, which is very small compared to the entire image pixels in the captured image. Moreover, the channel gain, and as a result, the signal-to-noise ratio (SNR) at a pixel, remains constant as long as the projected image of the transmitter LED occupies several pixels. Thus, the vehicle motion and pixel illumination model are used as guidelines for our system to overlook the overhead of recognizing the desired light sources in mobile environments. We have also neglected the frame-to-frame time gap to detect the image frame using the camera. This is because we have used a high-speed camera [5].

3.3 System Performance Parameters

In order to analyze our vehicular OCC system performance, we adopt the SNR formula given in [26]. According to [26], the received SNR, γ_t^b, of the link b at time t for a single LED-camera communication can be expressed as[2]

$$\gamma^b = \begin{cases} \frac{\rho k^2 P^2}{q P_n W f^2 l^2 (d^b)^2}; & \text{if } \ d^b < d_c \ , \\ \frac{\rho k^2 P^2}{q P_n W_{\text{fps}} s^2 (d^b)^4}; & \text{if } \ d^b \geq d_c \ , \end{cases} \tag{2}$$

where $k = \frac{(m+1)A}{2\pi} \ \cos^m(\phi) \ T_{\text{s}}(\theta) \ g \ \cos(\theta)$, ρ is the receiver's responsitivity, q is the electron charge, P_n is the power in background light per unit area, P is the optical transmit power, W_{fps} is the sampling rate of the camera in fps, l is the diameter of a LED, f is the focal length and s is the edge-length of a pixel.

Motivated by the trade-off among modulation order, achieved BER, and spectral efficiency, we consider adaptive modulation that permits us to adapt the modulation order by satisfying the target BER requirement of the system. This adaptive scheme can deal with the time-varying nature of the channel while maintaining the desired link quality and maximizing the rate for the given channel conditions [29]. Furthermore, the adaptive modulation scheme enables the OCC system to transmit at high-speed under favourable channel conditions, while the transmission rate decreases when the channel conditions worsen. It is worth noting that different users might have different transmission rates since they do not have precisely the same SNR, and consequently, the users have varying BERs. For the considered system, we study uncoded M-QAM with the square constellation as an example. Still, our scheme is general and other modulation schemes can be employed. The BER of the optical wireless channel at the receiver using the M-QAM scheme is evaluated similarly to [23] as:

$$\text{BER}^b = \frac{2\left(\sqrt{M^b} - 1\right)}{\sqrt{M^b}\log_2(M^b)} \ \text{erfc}\left(\sqrt{\frac{3 \ \gamma^b \ \log_2(M^b)}{2\,(M^b - 1)}}\right), \tag{3}$$

where M^b is the available constellation points for each V2V link b, e.g., $M = 4, 8, 16, \cdots$ and erfc$(\cdot)$ is the complementary error function. For a given M^b, the spectral efficiency of the M-QAM scheme can be expressed as:

$$\text{SE}^b = \log_2(M^b). \tag{4}$$

[2] For notational simplicity, we drop t from the notation in the remainder of the paper unless it is necessary; hence, we will adopt γ^b instead of γ_t^b and so on. Also, it is clear from the context that distance is our working variable.

It is worth noting that the adaptive modulation in our system is adjusted as follows. Suppose there is any change in the modulation order during communication. In such case, the transmitter informs the receiver regarding the employed modulation by appending a small overhead, e.g., some extra bits, to each transmitted packet. This overhead can be neglected because, in practice, a small set of modulation schemes is used, e.g., 6, in our system. This requires only three bits to be appended to the transmitted data for the receiver. Hence, in our system, the overhead for notifying the modulation order is minimal compared with the transmitted packet size, i.e., 5 kbits.

The channel capacity (measured in bits/s) of a camera-based communication system depends on the employed modulation scheme, as shown in [17] where the transmission rate of link b is expressed as

$$C^b = \frac{(W_{\text{fps}}/3) N_{\text{LEDs}} w \chi}{2\ \tan\left(\frac{\theta_l}{2}\right) \cdot d^b} \cdot \log_2(M^b), \tag{5}$$

where N_{LEDs} is the number of LEDs at each row of the transmitter, w is the image width (in case the rolling axis is along the width of the image sensor), and χ is the size of LED lights in cm^2. We divide W_{fps} by three as the modulated signal must be sampled at three times of sampling frames by the camera, which is sufficient for decoding the original M-QAM signal [24]. In other words, to perfectly reconstruct the amplitude and phase, a modulated symbol is sampled in three consecutive frames. Please note that the distance d^b in (5) is affected by the relative speed of vehicle v, which in turn affects the position of the vehicle on the road. Let us assume a slotted time. The inter-vehicular distance at current time t is adjusted using $d_t = d_{t-1} + v_t \cdot \Delta t$, where d_{t-1} is the distance of the previous time instance and Δt is the time elapsed between time instants t and $t-1$. Please note that we have considered the rolling shutter camera operation to calculate the channel capacity.

In the considered vehicular OCC system, the end-to-end latency is dominated by the transmission latency and thus we neglect the computational latency. This is due to the fact that we are processing a small amount of data, i.e., the decision information from transmitter to receiver, and thus, the computational time will be short. Thus, the transmission latency of the packet size, L, can be expressed similarly to [27] as $\tau^b = \frac{L}{C^b}$.

4 Proposed Constrained and MDP Formulation

4.1 Proposed Constrained Problem Formulation

Considering the proposed framework and ultra-low latency and BER requirements, we formulate an optimization scheme that aims at maximizing the sum spectral efficiency of the vehicular OCC system by selecting the optimal modulation order from an available set and adjusting the relative speed of the vehicle to the optimal value. The BER and latency are constrained such that they meet the values imposed by the system. Mathematically, our constrained maximization problem is, hence, formulated as:

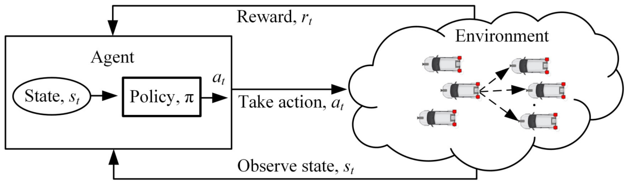

Fig. 2: An illustration of basic reinforcement learning framework for V2V communications.

$$\max_{\mathcal{M}, v} \quad \frac{1}{B} \sum_{b=1}^{B} \log_2\left(M^b\right), \tag{6}$$

$$\text{s.t.} \quad \text{BER}^b \leq \text{BER}_{\text{tgt}},\ \forall b; \tag{7}$$

$$\tau^b \leq \tau_{\max},\ \forall b; \tag{8}$$

$$M^b \in \mathcal{M},\ \forall b; \tag{9}$$

where $\mathcal{M}$ is the set of QAM modulation orders, BER_{tgt} is the maximum target BER, and $\tau_{\max}$ is the maximum affordable latency. Constraints (7) and (8) guarantee that the BER and latency thresholds are satisfied. The modulation scheme is chosen from a small set of available M-QAM schemes, as shown in (9).

4.2 MDP Formulation

The studied problem in (6) is mixed-integer programming (MIP) with nonlinear constraints for BER (7) and delay (8). This makes our problem NP-hard [30]. It is known that MIP problems have high computational complexity [31] and although it may be possible to solve them using dynamic programming or exhaustive search techniques, these methods cannot be used in dynamic systems as the one we investigate in this paper since they are extremely time-consuming or computationally demanding. As in our problem, we simultaneously control the speed and modulation for multiple links, the decision space is large. Due to the entailed computational and time complexities in solving the proposed problem, we first express the problem as an MDP problem in the next subsection. This gives us the opportunity to use other tools, such as deep RL, to solve the problem with less complexity. Note that vehicular communication must satisfy the maximum latency and BER requirements to ensure that the information is received reliably within the shortest time. We adopt an independent learning framework, where each vehicle independently decides its action, but they all affect the environment. It has been shown that this leads to well-performing solutions without requiring explicit communication [31]. Preceding to presenting our solution, we first model the optimization problem in (6) as an MDP in the next subsection.

4.2.1 Modelling of MDP

We model the proposed multi-agent RL problem as an MDP, where each vehicle acts as an agent, and everything beyond the particular vehicle is regarded as the environment. Every vehicular agent interacts with the environment to have a better understanding of it to decide its own policy. The agents explore the environment and improve the spectral efficiency maximization policies based on their observations of the environmental state. We chose to use independent Q-Learning [10] for our approach as it allows each local agent to learn its policy independently without requiring global communication. While there are methods such as [32], [33] that fall between independent and centralized learning, they still require some communication which may result in longer delays and still have significantly higher computational complexity compared to our independent learning approach. Moreover, considering only local communication significantly increases the state and action space (exponentially with the number of neighbours, i.e., vehicles), which may render the problem intractable. The deep Q-network utilized in this paper scales well with the number of vehicles.

The optimization problem (6) is modelled as an MDP with a tuple ($\mathcal{S}$, $\mathcal{A}$, p, r, ζ) [9], where $\mathcal{S}$ is the set of all possible states; $\mathcal{A}$ denotes the set of all possible actions; $p(s_{t+1}, r_t | s_t, a_t)$ denotes the transition probability which describes the probability that an agent selects an action $a_t \in \mathcal{A}$ and transits to a new state $s_{t+1} \in \mathcal{S}$ from the current state $s_t \in \mathcal{S}$; while r represents the reward. The parameter $\zeta \in [0, 1]$ is the discount factor, which gradually discounts the effect of an action on future rewards. A discount factor $\zeta = 0$ provides a short-sighted goal that maximizes the immediate reward. When ζ is close to 1, the agent focuses more on the future reward, and the scheme becomes far-sighted. In practice, a far-sighted approach is desirable as it achieves better returns by focusing on future discounted rewards. It is also notable that an algorithm with lower discount factors converges faster, especially during early learning. However, a small value of the discount factor may lead to highly suboptimal policies that are too myopic.

We present a general RL framework in Fig. 2 consisting of agents and environment. From this figure, we see that at each time t, an agent observes a state, $s_t \in \mathcal{S}$ and accordingly takes an action, $a_t \in \mathcal{A}$ based on the policy, π and receives a reward, r_t, from the environment. Next, we express the state space $\mathcal{S}$, action space $\mathcal{A}$, and reward function, r of the considered RL framework.

State Definition: In our system, the observed state from the environment by each agent couples two components: the backward distance vector, $\mathbf{d}_t^b = (d_t^1, \cdots, d_t^B)$ and the modulation orders $\mathbf{M}_t^b = (M_t^1, \cdots, M_t^B)$, selected from the set, $\mathcal{M}$ = {4, 8, 16, 32, 64}, being the transmitting modulation order for the backward vehicles. In summary, the state is expressed as $s_t = \left\{\mathbf{d}_t^b, \mathbf{M}_t^b\right\}$.

Action Definition: At each time t, the agent takes an action a_t, a decision regarding the relative speed of the agent (i.e., vehicle), v_t and selecting modulation order, $\mathbf{M}_t^b \in \mathcal{M}$, based on the current state, s_t by following a policy π. Overall, the action space is summarized as $a_t = \left\{v_t, \mathbf{M}_t^b\right\}$.

Reward Definition: At each time slot t, when the agent takes an action a_t in state s_t, it will immediately receive a reward r_t. Note that, an effective reward framework is imperative for the learning algorithm to achieve the desired goal, which is achieved through exploration. Therefore, the reward function that guides the overall learning should be consistent with the objective.[3] First, we express the reward related to distance as follows:

$$r_t^{\mathrm{d},i} = \begin{cases} -1 \times (d_{\mathrm{stop}} - d_t^b), & d_t^b < d_{\mathrm{stop}} \ , \\ \frac{1}{d_t^b - d_{\mathrm{stop}}}, & d_t^b > d_{\mathrm{stop}} \ , \end{cases} \tag{10}$$

where i is the index of the agent. Recall that, d_t^b represents the backward distance of the vehicle; however, in designing our reward, we only consider the vehicle behind residing in the same lane on the road. The priority is to avoid collision with the vehicle in the same lane. This is the decisive vehicle because it has the possibility of coming closer to the agent vehicle in the following time step or in the near future. d_{stop} is the stopping distance, which is equal to the sum of the distance covered by the vehicle to travel after the brakes are activated and the distance covered by the driver's reaction time after observing a situation [34]. In our system, each vehicle performs the same process individually. As a result, for notational simplicity, we drop i hereafter. Since our objective is to maximize the sum spectral efficiency, we design our reward function as a weighted sum of a reward related to the backward distance and the sum spectral efficiency (6). As the goal of RL is to maximize the reward, it will conclusively maximize the sum spectral efficiency while maintaining a safe distance. Hence, considering the objective function (6), the overall reward, R_t, can be expressed as

$$R_t = \omega_d \, r_t^{\mathrm{d}} + \omega_r \, \frac{1}{B} \sum_{b=1}^{B} \log_2 \left(M_t^b \right), \tag{11}$$

where ω_d and ω_r are positive weights that balance distance and sum spectral efficiency rewards. The weights are adjusted based on the system requirements. It sets the priority depending on its distance and modulation scheme changes.

5 Proposed Solution

5.1 Constrained MDP Formulation

The goal of RL is to find the optimal policy that maximizes the expected return from the state s_t, whereas the return, G_t, is defined as the cumulative discounted reward, as follows:

$$G_t = \sum_{j=0}^{\infty} \zeta^j R_{t+j+1}, \qquad 0 \leq \zeta \leq 1. \tag{12}$$

In our problem, an RL agent aims at determining the optimal policy, i.e., speed and modulation order, while respecting BER and latency constraints. This can be formally expressed as

$$\max \quad \mathbb{E}\left[G_t\left(s_t, a_t\right)\right], \ \forall t \tag{13}$$

$$\text{s.t.} \quad \mathrm{BER}_t^b \leq \mathrm{BER}_{\mathrm{tgt}}, \ \forall t; \tag{14}$$

$$\tau_t^b \leq \tau_{\max}, \ \forall t; \tag{15}$$

5.2 The Lagrangian Approach

According to [35], constrained MDP problems can be solved by recasting them as unconstrained ones via the Lagrange relaxation method. Hence, we reformulate the constrained optimization problem in (13) - (15) by introducing Lagrange

[3] From hereon, we will use backward distance and distance interchangeably though it indicates the same meaning.

multipliers associated with the BER and latency constraints, $c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t)$, as:

$$\begin{aligned} c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t) = R_t\left(s_t, a_t\right) - \sum_{b=1}^{B} \lambda^b \cdot \left(\mathrm{BER}_t^b - \mathrm{BER}_{\mathrm{tgt}}\right) \\ - \sum_{b=1}^{B} \nu^b \cdot \left(\tau_t^b - \tau_{\max}\right), \end{aligned} \tag{16}$$

where $\boldsymbol{\lambda} = (\lambda^1, \lambda^2 \cdots, \lambda^b)$ and $\boldsymbol{\nu} = (\nu^1, \nu^2, \cdots, \nu^b)$ are vectors representing the Lagrange multipliers corresponding to the constraints in (14) and (15), respectively. The optimal value of the constrained MDP problem can be computed as [36]:

$$\begin{aligned} L_\delta^{\pi^*,\boldsymbol{\lambda}^*,\boldsymbol{\nu}^*}(s) &= \min_{\pi\in\phi} \max_{\boldsymbol{\lambda},\boldsymbol{\nu}\geq 0} V^{\pi,\boldsymbol{\lambda},\boldsymbol{\nu}}(s) - \sum_{b=1}^{B} \lambda^b \delta_1 - \sum_{b=1}^{B} \nu^b \delta_2 \\ &= \max_{\boldsymbol{\lambda},\boldsymbol{\nu}\geq 0} \min_{\pi\in\phi} V^{\pi,\boldsymbol{\lambda},\boldsymbol{\nu}}(s) - \sum_{b=1}^{B} \lambda^b \delta_1 - \sum_{b=1}^{B} \nu^b \delta_2, \end{aligned} \tag{17}$$

where $\delta = \{\delta_1, \delta_2\}$, with $\delta_1 = \mathrm{BER}_{\mathrm{tgt}}$ and $\delta_2 = \tau_{\max}$. ϕ denotes the set of all possible stationary policies,

$$V^{\pi,\boldsymbol{\lambda},\boldsymbol{\nu}}(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \zeta c^{\boldsymbol{\lambda},\boldsymbol{\nu}}\left(s_t, \pi\left(s_t\right)\right) \mid s_0 = s\right]. \tag{18}$$

A policy π^* is optimal for the constrained MDP, if and only if

$$L_\delta^{\pi^*,\boldsymbol{\lambda}^*,\boldsymbol{\nu}^*}(s) = \max_{\boldsymbol{\lambda},\boldsymbol{\nu}\geq 0} V^{\pi^*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s) - \sum_{b=1}^{B} \lambda^b \delta_1 - \sum_{b=1}^{B} \nu^b \delta_2. \tag{19}$$

For a fixed $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$, the rightmost maximization of (17) is equivalent to solving the following dynamic programming equation:

$$\begin{aligned} V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t) = \min_{a_t\in\mathcal{A}} \Bigg\{ & c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t) \\ & + \zeta \sum_{s_{t+1}\in\mathcal{S}} p(s_{t+1} \mid s_t, a_t) V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_{t+1}) \Bigg\}, \forall s \in \mathcal{S}, \end{aligned} \tag{20}$$

where $V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}} : \mathcal{S} \mapsto \mathbb{R}$ is the optimal state-value function and s_{t+1} is the state at time slot $t+1$.

We also define optimal action-value function $Q^{*,\boldsymbol{\lambda},\nu} : \mathcal{S} \times \mathcal{A} \mapsto \mathbb{R}$ which represents the Q-value of action a_t in a given state s_t.

$$\begin{aligned} Q^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t) = \; & c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t) \\ & + \zeta \sum_{s_{t+1}\in\mathcal{S}} p(s_{t+1} \mid s_t, a_t) V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_{t+1}), \end{aligned} \tag{21}$$

where $V^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_{t+1}) = \max_{a_t\in\mathcal{A}} Q^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_{t+1}, a_t), \forall s \in S$. In words, $Q^{*,\boldsymbol{\lambda}\boldsymbol{\nu}}(s_t, a_t)$, is the infinite discounted cost achieved after taking action a_t in state s_t and therefore, following the optimal policy $\pi^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}$, which is given by

$$\pi^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t) = \arg\max_{a_t\in\mathcal{A}} Q^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t), \forall s \in S. \tag{22}$$

[4]In practice, the optimal policy, π^*, cannot be determined using value-iteration method [9] as it requires transition probabilities to be known beforehand. For the considered problem, continuous computation of the transition probability matrix is necessary, which is computationally demanding. To solve this problem, we adopt a model-free RL approach known as Q-Learning, which learns Q^* and π^* online, without requiring the model of the environment and computing the transition probability matrix. Q-Learning uses the $Q_t(s_t, a_t)$ values instead of the value function in (20). $Q_t(s_t, a_t)$ represents how good it is to take action a_t when starting from state s_t, and thereafter follow the policy π. To determine the optimal policy π^*, the Q-Learning algorithm employs the following recursive formula to update the $Q_t(s_t, a_t)$ values:

$$\begin{aligned} Q_{t+1}(s_t, a_t) = & (1 - \alpha_t) Q_t(s_t, a_t) + \alpha_t \Bigg[c_t(s_t, a_t) \\ & + \zeta \max_{a_{t+1}\in\mathcal{A}} Q_t(s_{t+1}, a_{t+1}) \Bigg], \end{aligned} \tag{23}$$

where $\alpha_t \in [0, 1]$ is a time-varying learning rate and a_{t+1} is the greedy action in state s_{t+1} at time slot $t+1$. The learning rate refers to the rate at which newly updated information overrides the old one.

Q-Learning can select actions using policies such as the ϵ-greedy, where $\epsilon \in [0, 1]$ [37]. It has been shown in [36] that the Q-Learning algorithm will eventually converge to the optimal Q, $Q^*(s_t, a_t)$ with probability 1 when all the state-action pairs are visited often, and the learning rate α_t respects the following conditions:

$$\alpha_t \in [0, 1], \quad \sum_{t=0}^{\infty} \alpha_t = \infty, \quad \sum_{t=0}^{\infty} \left(\alpha_t\right)^2 < \infty. \tag{24}$$

We discuss the learning of the optimal $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$ in Section 5.3.2.

5.3 Deep Q-Learning

Q-Learning is a well-known method [9] that is used to solve problems expressed as MDP. The convergence speed of this algorithm depends on the state-action space size. Q-Learning converges faster for small state-action spaces since the agent can quickly explore the state-action pairs and determine the optimal policy. For larger state-action spaces, the convergence is slow which makes the determination of the optimal actions not feasible within the stringent time constraints imposed by the dynamic nature of the environment in problems like the one we study here. Although some linear function approximation approaches exist for solving large-scale RL problems, their capabilities are limited to medium-scale problems. In high-dimensional and complex systems, conventional RL methods cannot learn the informative features of the environment quickly, despite employing effective approximation functions. This is due to the fact that most of the state-action pairs are rarely visited, and thus the corresponding Q-values are not updated regularly, leading to a much longer time to

[4] For notational simplicity, we drop the Lagrangian multipliers from the notation in the remainder of the paper unless it is necessary, for example, we will write $c(s_t, a_t)$, $Q^*(s_t)$, instead of $c^{\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t, a_t)$, $Q^{*,\boldsymbol{\lambda},\boldsymbol{\nu}}(s_t)$, respectively.

converge. More importantly, distance and speed are continuous values that lead to a large state-action space; hence, the tabular Q-learning algorithm cannot be used because it works with discrete values. Discretization may be applied, but this affects the quality of the solution.

These issues, however, can be overcome by using deep learning-based function approximators, in which deep neural networks (DNN) are trained to learn the best policy. In a deep Q-Network (DQN), a DNN function approximator with weights θ is employed as Q-network, and then Q-Learning is combined with deep learning. Once the weights are determined, the Q-values, $Q(s, a)$, will be the DNN outputs. DNN addresses sophisticated mappings between channel information and desired output via large amounts of training data, which are then used to calculate Q-values.

5.3.1 *Target Network*

In order to stabilize the learning of DQN, we follow the target network approach. The DQN is composed of two networks: the main network, which approximates the Q-function, and the target network, which serves as a target for updating the main network. In the training phase, the main network parameters $\boldsymbol{\beta}$ are adjusted after every action, and target network parameters $\boldsymbol{\beta}_{-}$ are updated after a certain period of time. The target network is not updated after each iteration because it adjusts the main network updates to control the value estimations. If both networks are updated simultaneously, the change in the main network would be exaggerated due to the feedback loop from the target network, resulting in an unstable network. To ensure stability in learning, the neural network aims to minimize the loss function, $L(\boldsymbol{\beta})$, which is expressed as

$$L(\boldsymbol{\beta}) = \mathbb{E}\left[y_t - Q\left(s_t, a_t; \boldsymbol{\beta}\right)\right]^2, \tag{25}$$

where $y_t = c(s_t, a_t) + \zeta \max_{a_{t+1} \in \mathcal{A}} Q\left(s_{t+1}, a_{t+1}; \boldsymbol{\beta}_{-}\right)$ is the target for each iteration. Note that, $\boldsymbol{\beta}_{-}$ are held fixed when optimizing the loss function $L(\boldsymbol{\beta})$.

5.3.2 *Optimal Lagrange Multipliers*

The optimal value of the Lagrange multipliers λ^b, ν^b in (16) depend on the BER constraint, BER_{tgt} and latency constraint, $\tau_{\max}$, respectively and can be learned online using a stochastic sub-gradient method as presented in [38]

$$\lambda^b_{t+1} = \Lambda\left[\lambda^b_t + \varpi_t(\text{BER}^b_t - \text{BER}_{\text{tgt}})\right], \tag{26}$$

$$\nu^b_{t+1} = \Lambda\left[\nu^b_t + \varpi_t(\tau^b_t - \tau_{\max})\right], \tag{27}$$

where we apply the projection operator Λ in order to project λ^b and ν^b onto $[0, \lambda_{\max}]$ and $[0, \nu_{\max}]$. To ensure the boundedness of $\lambda_{\max}$ and $\nu_{\max}$, we consider $\lambda_{\max}, \nu_{\max} > 0$ to be large enough. ϖ_t corresponds to a time-varying learning rate, which obeys the same conditions as α_t in (24). The following additional conditions must be jointly satisfied by α_t and ϖ_t to guarantee the convergence of (26) and (27) to $\boldsymbol{\lambda}^*$ and $\boldsymbol{\nu}^*$, respectively:

$$\sum_{t=0}^{\infty}(\alpha_t + \varpi_t) < \infty \quad \text{and} \quad \lim_{t\to\infty} \frac{\varpi_t}{\alpha_t} \to 0. \tag{28}$$

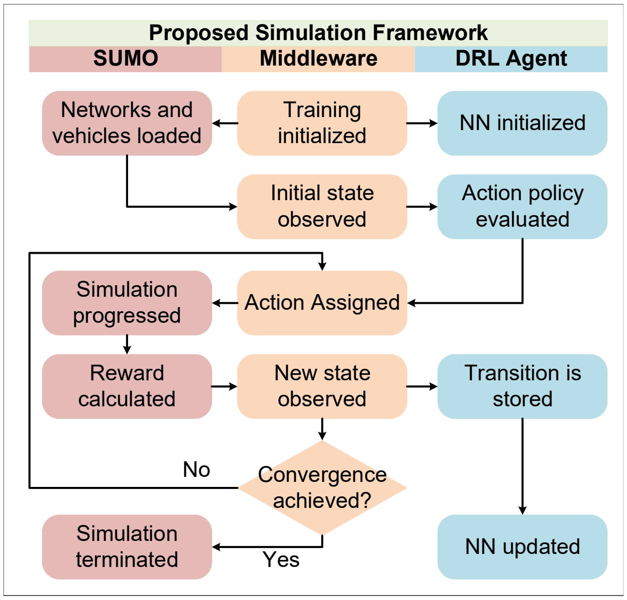

Fig. 3: Proposed simulation framework combining SUMO simulator, middleware and DRL agent for vehicular communication.

5.4 Complexity Discussion

In practice, convergence in finding the optimal value of BER and latency constraints is improved with the cost of increased complexity in finding λ^b and ν^b. The optimal constraint values can be determined by selecting constraints that are frequently violated during the sub-gradient method in (26) and (27). Additionally, the time complexity of the running process depends on the size of the state-action space and the structure of the neural networks. Please note that, a key distinction between RL and the conventional stochastic non-convex method is the inherent computational costs to link the bias in the search direction, which is determined by the technique utilized in the DRL framework. Moreover, discretization can affect the quality of the solution. If the discretization is too coarse, it may result in an inefficient sub-optimal solution; if it is too fine, it will take an enormous amount of time to find a solution with no guarantee of optimality. Therefore, the challenges are balancing the trade-off between choosing the optimal gradient methods, precise generalization or quantization approaches, and function parameterization in RL.

6 Simulation Setup

This section describes the implementation details of our proposed DRL-based vehicular OCC scheme. Specifically, we build the simulation environment upon microscopic traffic simulator Simulation of Urban Mobility (SUMO) [39] and DRL framework within SUMO.

6.1 SUMO Framework

Our simulation framework maintains the connection between the SUMO and DRL agent using Traffic Control Interface (TraCI). To simulate the proposed vehicular framework in a more practical scenario, we convert the proposed environment into a corresponding SUMO map. Each vehicle is considered an agent and modelled accordingly to test the proposed DRL method in the integrated environment. The

Algorithm 1 DQN Training Algorithm

```
Initialization: Initialize SUMO environment, DQN parameters, replay memory according to system requirements.
Output: Action-value function, loss (25).
for each episode do
  Update vehicle speed and modulation order
  for each link, $b$ do
    Observe state $s_t$
    Choose action $a_t$ according to the $\epsilon$-greedy policy
    Execute action $a_t$, observe reward $r_t$ and next state $s_{t+1}$
    Store transitions ($s_t$, $a_t$, $r_{t+1}$, $s_{t+1}$) in the replay memory
  end for
  Agent takes actions and receive reward $r_t$ using (11).
  Update Lagrange multipliers $\boldsymbol{\lambda}$ and $\boldsymbol{\nu}$ using subgradient method as in (26) and (27), respectively.
end for
Sample a mini-batch from the replay memory.
Optimize error between Q-network and target Q, defined in (25), using RMSProp optimizer gradient descent.
```

vehicles enter randomly in SUMO environment and move or leave the network following SUMO mobility models.

As shown in Fig. 3, the proposed simulation framework consists of three parts: firstly, SUMO, which is the simulator environment for creating traffic scenarios; secondly, the middleware that connects the SUMO environment with the DRL agents; and finally, the DRL agents, which maintain and update the network policies and execute actions for the simulation. After the training is initialized, the SUMO simulator is loaded, progressed, and reset with required information such as the transportation network and vehicles via TraCI. During the simulation, TraCI interacts with the SUMO environment and extracts the data from SUMO to produce observations for state space and aggregate rewards. Moreover, TraCI retrieves different features from the network, e.g., the number of vehicles on each road, the speed of the vehicle, and the current position of the agent. Based on the current observations, the DRL agent evaluates the current traffic environment and assigns an action based on the policy of the neural network. Accordingly, the agent updates the state and moves to the next step in the SUMO environment and this process continues until all the simulation steps finish. The reward is then computed and transferred to the DRL agent for optimization at the end of each simulation run. The objective is to train the policy network that ensures higher communication quality in the form of sum spectral efficiency, delay, and BER.

We have modified the SUMO environment according to the requirement of our proposed multi-agent vehicular system. We illustrate a screenshot of the simulated vehicular model represented on SUMO GUI interface in Fig. 4. As shown in this figure, we have three lanes, where vehicles move at different velocities and each vehicular agent has potentially multiple vehicles in front and back. Please recall that the agent must satisfy the constraints of the system to generate a higher reward and minimize the loss.

TABLE 1: List of DRL hyper-parameters and their values

Parameter, Notation	Value
Mini-batch size	32
Replay memory size	100000
Number of hidden layer (Neurons)	1(250)
Exploration rate, ϵ	0.05
Discount factor, ζ	0.98
Activation function	ReLU
Optimizer	RMSProp
Learning rate (used by RMSProp)	0.001
Gradient momentum (used by RMSProp)	0.95

6.2 DNN Settings

6.2.1 Network Architecture

This subsection provides the details of the employed DQN architecture as well as the training parameters we employed. The DQN consists of three fully connected layers, including an input layer, a hidden layer, and an output layer. Recall that distance and modulation order define the state space; hence, the input layer consists of $M^b + d^b$ nodes. The output layer consists of $M^b + v$ nodes, as we have $M^b + v$ actions. The hidden layer has 250 neurons. We use rectified linear unit (ReLU) as the activation function [37], defined as $f(x) = \max(0, x)$. We adopt root mean square propagation (RMSProp) optimizer [40] as the training algorithm to minimize the loss function and update DQN network parameters. We set the initial learning rate α to 0.001. It is known that a large learning rate leads to fast convergence, but at the same time, may incur a poor convergent point with unsatisfactory performance, e.g., local minima, saddle point. On the contrary, intensive training computations are required for a small α as it results in slow convergence. Therefore, an appropriate α should carefully be chosen. In our case, the RMSProp optimizer is used to vary the learning rate over time. We use TensorFlow [41] in our simulations to implement deep reinforcement learning framework. We implement ϵ-greedy policy to balance between exploration and exploitation while avoiding overfitting. According to ϵ-greedy policy, the action with maximum $Q_t(s_t, a_t)$ value is chosen with probability $1 - \epsilon$ while a random action is selected with probability ϵ.

6.2.2 Training Procedure

The training procedure of our proposed DQN algorithm is summarized in **Algorithm** 1. The input of the algorithm is the current observations (distance and modulation scheme), and the output is the actions (speed and modulation scheme) chosen by the vehicle. The agents map the actions with the corresponding action-value functions, i.e., Q-value. We train the DQN algorithm for multiple episodes, and at each training step, all the agents execute the ϵ-greedy policy to explore the state-action space. Following the environment transition due to channel variation and actions taken by all agents, each agent observes and stores the transition tuple, (s_t, a_t, r_{t+1}, s_{t+1}), in the replay memory. At each episode, a uniformly sampled mini-batch of experiences is taken from the memory for updating $\boldsymbol{\beta}$ parameters of (25) using stochastic gradient descent methods and the loss is estimated using (25).

For the simulations, we train the DQN for 10000 episodes. The exploration rate, ϵ is set to 0.05. The target Q-network parameters are updated every 400 learning

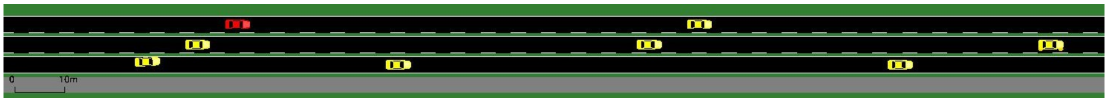

Fig. 4: Illustration of proposed scenario in SUMO GUI interface.

TABLE 2: Vehicular OCC modelling parameters

Parameter, Notation	Value
Angle of irradiance w.r.t. the emitter, ϕ	70^o
AoI w.r.t. the receiver axis, θ	60^o
FOV of the camera lens, θ_l	90^o
Image sensor physical area, A	10 cm^2
Transmission efficiency of optical filter, T_s	1
Concentrator/lens gain, g	3
Optical transmitting power, P	1.2 Watts
Constellation size, M	4, 8, 16, 32, 64
Camera-frame rate, W_{fps}	1000 fps
Number of LEDs at each row, N_{LEDs}	30
Packet size, L	5 kbits
Size of the LED, χ	15.5×5.5 cm^2
Resolution of image, w	512×512 pixels

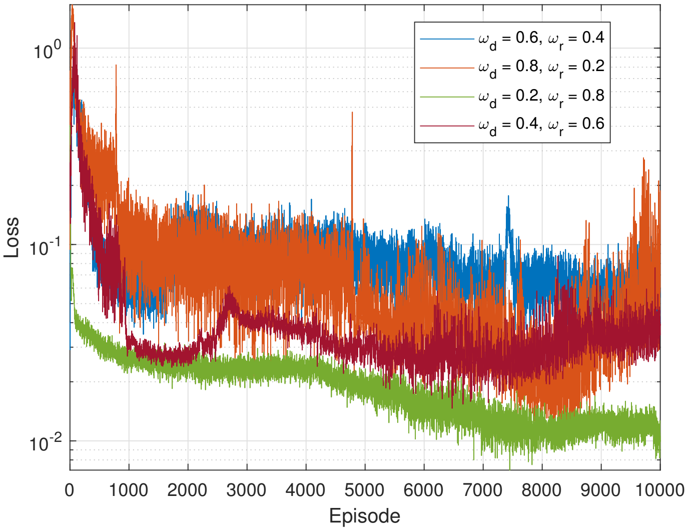

Fig. 5: Convergence of loss function for ϵ = 0.05 and α = 0.001.

steps, where each episode contains 100 steps. We choose a discount factor, $\zeta = 0.98$. For our simulation run, we use a track size of 180 m, and we measure the density of vehicles as the number of vehicles per 180 m. The total replay memory size for storing the transactions is 100000, and the mini-batch for training is 32. The training and testing parameters of the DRL are presented in Table 1.

6.2.3 Normalization

The goal of normalization is to bring the different sub-rewards corresponding to delay and spectral efficiency in (16) to be on a similar scale. This normalization improves the performance and provides training stability of the NN model. Specifically, we normalize the reward function (11), BER and latency constraints of (16) to keep the scale between 0 and 1. Please note that, we perform quantization on the continuous values of distance and speed of the vehicle to convert them into discrete values. For example, we quantize the values of distance into step length of 1 m and the speed into 0.5 ms^{-1} step.

7 Performance Evaluation

In this section, we evaluate the performance of the proposed multi-agent RL-based sum spectral efficiency maximization scheme for vehicular OCC. The simulation parameters for the OCC system model are listed in Table 2.

7.1 Overview of Comparison Schemes

We investigate the performance of the proposed multi-agent DRL-based vehicular scheme, termed hereafter as the proposed scheme against different methods for comparison. We present a brief summary of all the schemes under comparison below:

- **Proposed scheme**: By the proposed scheme, we refer to our multi-agent DRL-based vehicular OCC system, where each agent performs independent learning considering all other vehicles as environment. In this case, we employ the settings as we discuss in Sections 6.2.1 and 6.2.2. We set the discount factor to 0.98.
- **Greedy**: This method is a variant of our scheme, where we set the discount factor to $\zeta = 0$ in (25), while we keep all other parameters of the systems as reported in Table 1. In this scenario, the agent chooses the action which maximizes only the immediate reward.
- **Far-sighted**: This method is a variant of our scheme, where we set the discount factor to $\zeta = 1$ in (25), while we keep all other parameters of the systems as reported in Table 1. This scheme takes future rewards into account more strongly and ignores immediate rewards.
- **Random**: This is a scheme, where the actions are chosen randomly for all the vehicles at each time slot. In this case, the system parameters are not optimized and the agent chooses speed and modulation schemes randomly.
- **RF-based MARL [13]**: This is a multi-agent RL based resource allocation scheme presented in [13]. This method is based on RF technology. For this scheme, we adapt the hyper-parameters according to our proposed scheme while keeping the environment unchanged. This scheme considers centralized learning and distributed implementation. The system performance-related reward is available to each individual agent through a centralized base station in the cellular network. Then the agent adjusts its action towards the optimal policy by updating its DQN and utilises its local observation and trained DQN to select the best action. Finally, the agent communicates the updated DQN towards the base station.
- **RF-based SARL [13]**: This is a single agent RL based scheme proposed in [13], specified as single-agent reinforcement learning (SARL), where at each time only an agent, i.e., V2V link, updates its action based on the locally observed information, whereas other agents' action remains unchanged. A single DQN policy is shared over the vehicular network for all the vehicles.

We would like to note that we have not considered some existing methods, such as [7], [8], as baselines for

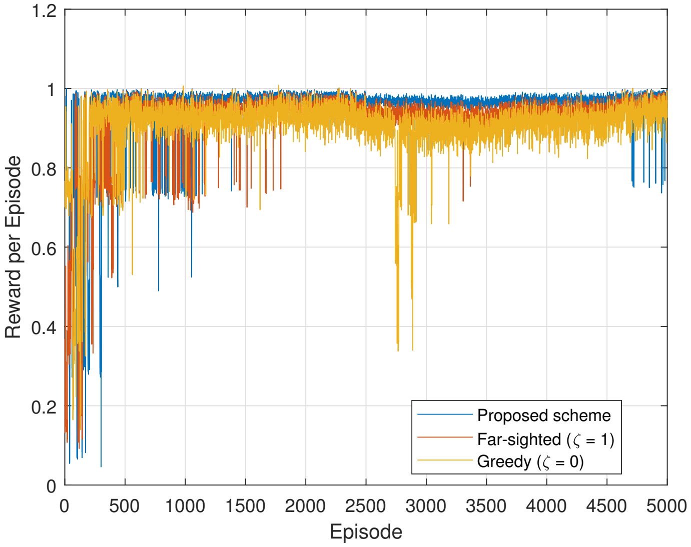

Fig. 6: Reward per training episode for three different approaches for ϵ = 0.05 and α = 0.001.

our proposed vehicular OCC problem because they rely on traditional optimization schemes and centralized communication, which cannot guarantee reliable and low-latency communication in fast-varying environments. Moreover, the extremely high complexity of our studied problem of simultaneously controlling multiple decision variables makes them inappropriate. Instead, we compared our proposed approach with an RF-based DRL scheme [13], which is regarded as state-of-the-art and shares more similarities with our proposed multi-agent vehicular problem.

7.2 Simulation Results

The convergence trend of the training algorithm confirms the suitability of the proposed scheme. To this end, we investigate the convergence of the proposed algorithm. First, we perform an ablation study to determine the weight values corresponding to distance and rate rewards in (11). In doing so, we examine our algorithm for different weight settings of distance and rate rewards, but for simplicity of representation, we only demonstrate four settings, including $\omega_d = 0.2$ and $\omega_r = 0.8$, $\omega_d = 0.4$ and $\omega_r = 0.6$, $\omega_d = 0.6$ and $\omega_r = 0.4$, $\omega_d = 0.8$ and $\omega_r = 0.2$, as shown in Fig. 5. We observe that we achieve lower loss when we allocate higher weight value to the spectral efficiency component. By observing Fig. 5, we can see that our scheme converges at around 8000 episodes for $\omega_d = 0.2$ and $\omega_r = 0.8$. On the contrary, other weight sets require longer times for convergence and show frequent variations in the loss and offer higher loss than $\omega_d = 0.2$ and $\omega_r = 0.8$ set. So, we adopt this weight setting for the rest of our evaluation.

We then present the rewards per training episode to analyze the convergence behaviour of the multi-agent vehicular OCC system at three different discount factors, i.e., the proposed scheme ($\zeta = 0.98$), greedy ($\zeta = 0$) and far-sighted ($\zeta = 1$). The results are depicted in Fig. 6. Please note that for ease of visualization, we present the reward until 5000 episodes as they follow the same trend after that. Based on this figure, we observe that the greedy and far-sighted approaches achieve better performance than the proposed scheme until 1500 episodes. However, as training progresses, the reward for the proposed scheme improves and reaches to lower loss. Instead, the rewards for greedy and far-sighted schemes vary throughout the training episodes. We can conclude that the proposed scheme achieves higher rewards than other variants of our scheme.

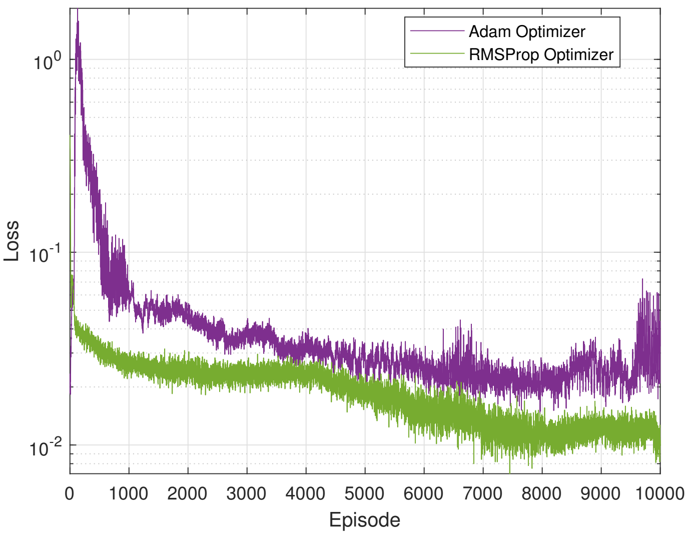

Fig. 7: Performance comparison between RMSProp and Adam gradient optimizer versus training episode.

There are various gradient descent optimizers that vary the learning rate adaptively to minimize the loss in the DQN. Here, we investigate the loss performance of two commonly used optimizers, RMSProp and adaptive moment estimation (Adam) for 10000 episodes. The results are illustrated in Fig. 7. This figure shows that the RMSProp optimizer achieves lower loss than the Adam optimizer over the training period. In particular, the Adam optimizer does not converge within 10000 episodes, whereas the RMSProp converges around 7000 episodes. Therefore, we adopt RMSProp optimizer in our framework.

To justify the superiority of the proposed multi-agent DRL-based vehicular OCC scheme, we compare its performance with MARL and SARL methods presented in [13] and a random scheme. We utilize the same DQN parameter to optimize the problem in [13]. For example, we implement a single hidden layer with 250 neurons instead of three hidden layers, a fixed discount factor, and 10000 training episodes. In addition, we formulate the spectral efficiency and latency according to our formulation. Though the system proposed in [13] did not consider latency, we estimated it to investigate how the latency requirements are satisfied. As the MARL and SARL methods require base stations to communicate with each other, they involve uplink and downlink latency in addition to processing latency. Whereas our system has only transmission latency as it is a decentralized scheme, RF-based MARL and SARL in [13] require centralized communication, which incurs additional latency.

Fig. 8 shows the maximized sum spectral efficiency performance with regard to the density of vehicles for all schemes under comparison. This figure shows that for all techniques employing our suggested framework, including the greedy, far-sighted, proposed scheme, and the random scheme, the sum spectral efficiency increases with an in-

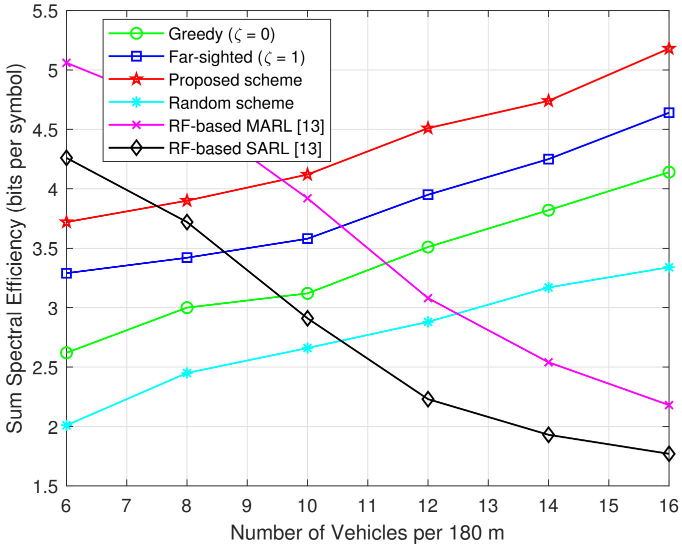

Fig. 8: Comparison of sum spectral efficiency versus number of vehicles with different approaches for $\epsilon = 0.05$.

crease in density of vehicles. On the contrary, the performance drops with increasing density of vehicles for RF-based MARL [13] and SARL systems [13]. The reason for the lower sum spectral efficiency in our proposed scheme with lower vehicle density is due to the increased distance between vehicles, which may weaken transmission links and limit the ability to communicate using higher modulation orders. In contrast, RF-based schemes experience less interference with fewer vehicles, resulting in higher sum spectral efficiency. However, as the density of vehicles increases, interference also increases, which negatively affects the performance of RF systems, as shown in Fig. 8. In our proposed OCC system, the increased density of vehicles leads to the vehicles being closer to each other and this improves the quality of the communication links. Hence, higher sum spectral efficiency is achieved. The same trend is noticed for the random scheme as for this scheme, which is a variant of our scheme, also the transmission link quality improves when the density of vehicles increases as the distance between the vehicles becomes smaller. This happens even if a suboptimal action is chosen and on average there are larger spectral gains than MARL and SARL. This is why the random scheme outperforms the MARL and SARL schemes when the density of vehicles increases. We would like to recall that the key difference between the proposed scheme and the random scheme is that in the latter scheme, the agent selects actions randomly (selection of speed and modulation schemes) for all vehicles at each time slot, whereas the proposed scheme decides the optimal actions using the proposed DRL framework. From the Fig. 8, we see that with a density of vehicles of 16 vehicles per 180 m, the proposed algorithm achieves rates that are approximately 2.4 times better than those of the MARL, 2.9 times better than those of the SARL, and roughly 1.6 times better than those of the random scheme. When vehicle density is 6 vehicles per 180m, however, the sum spectral efficiency is reduced by 0.73 times for MARL and 0.82 times for SARL. We might infer from this comparison that our OCC system operates better in urban settings or highways with dense traffic where the vehicle density is always higher.

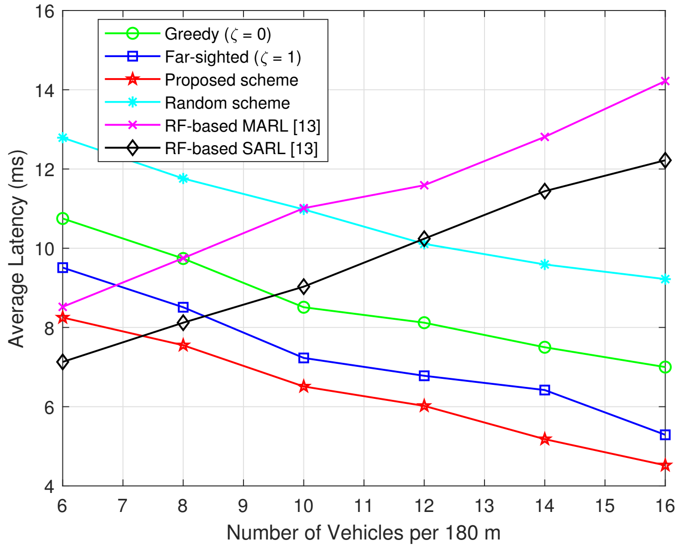

Fig. 9: Comparison of average latency versus number of vehicles with different schemes for $\epsilon = 0.05$ and $\alpha = 0.001$.

In Fig. 9, we exhibit the comparative findings of average latency versus vehicle density per 180m, which demonstrate that as the number of vehicles increases, the average latency decreases for all variants of our scheme utilizing the proposed algorithm. While it follows the opposite trend for the SARL and MARL schemes. This is because when the density of vehicles increases, the distance between vehicles reduces, and thus the experienced delay decreases. More significantly, the interference gets worse as the density of the vehicles rises. The centralized RF-based system involves latency as well because it needs to communicate with the base station and gets feedback. As a result, the latency increases with the increase of vehicle density. For our proposed scheme, there is no interference, which improves the spectral efficiency and hence, the latency with the increase of vehicle density. The random scheme follows a similar trend as the proposed scheme as the communication quality improves and the inter-vehicular distance reduces when the density of vehicles increases. As a result, the random scheme provides more significant latency benefits and performs better than the MARL and SARL schemes at higher vehicle densities. Our scheme achieves the lowest average latency of 4.5 ms and the maximum of 8.2 ms when the density of vehicles is 16 and 6, respectively. Whereas for MARL, SARL and the random scheme, the average latency is 8.5 ms and 14.2 ms, 7.1 ms and 12.2 ms, 12.9 ms and 9.2 ms, respectively, when the number of vehicles is 6 and 16 vehicles per 180m. From this comparison, it is seen that our proposed algorithm achieves lower latency compared to other schemes.

To explore whether the proposed scheme can maximize the sum spectral efficiency while also respecting the latency and BER constraints, we present the cumulative distribution function (CDF) of BER and latency for the schemes under comparison. First, we compare the CDF of the observed latency considering the maximum latency of all available links at each time slot for 10000 episodes in Fig. 10. From the figure, we observe that the proposed scheme can al-

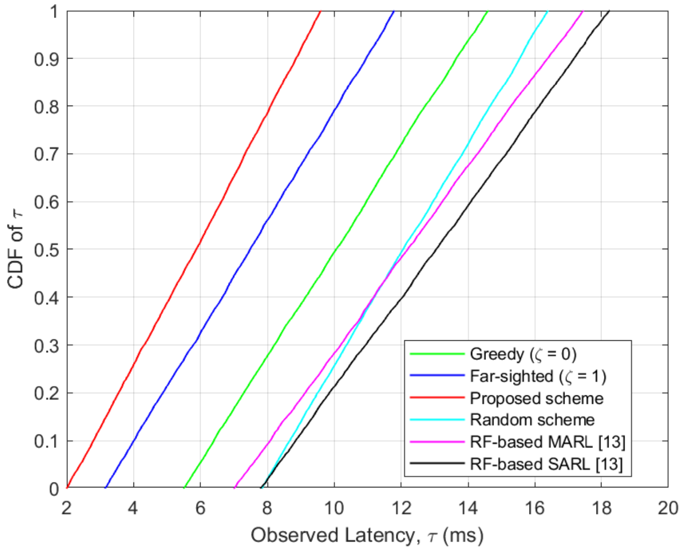

Fig. 10: CDF of observed latency while considering the maximum latency of all the available links behind the agent for $\epsilon = 0.05$ and learning rate $\alpha = 0.001$.

ways satisfy the latency requirement of 10 ms, whereas the greedy, far-sighted and random methods, satisfy the constraint only 50%, 78%, and 27% of the time, respectively. At the same time, the RF-based MARL and SARL schemes meet the latency requirements for 29%, and 20% of the time, respectively.

Finally, Fig. 11 illustrates the comparison of CDF of the observed BER for different schemes under comparison when the schemes have been optimized for 10000 episodes. In doing so, we examine only the maximum observed BER of all available links at each time slot, which will respect the minimum BER. From this figure, we note that our algorithm always satisfies the BER constraints of 10^{-4}. We can also see that the other algorithms violate the BER constraints most of the time. Specifically, far-sighted schemes satisfy BER requirements a maximum of 40% of the time, whereas greedy and random schemes meet 27% and 8% of the time, respectively. Similarly to what we observed in Fig. 10, the proposed method also respects the BER requirement when other schemes satisfy it only for some period of time.

8 Conclusion

In this paper, we present a DRL-based sum spectral efficiency optimization scheme for a multiple vehicular OCC scenario while respecting BER and latency requirements. Firstly, we model the OCC channel and several performance parameters. Then, we formulate a sum spectral efficiency maximization problem considering a small set of modulation orders, as well as the BER and latency constraints. To overcome the fact that the studied problem is NP-hard, we first formulate the optimization problem as an MDP. We design the reward function considering the objective function and the problem constraints. We then convert the constrained problem into an unconstrained problem through the Lagrangian relaxation method by relaxing the BER and latency constraints. To solve the problem, we employ deep Q-Learning to deal with large state-action spaces. We verify the performance of our proposed scheme

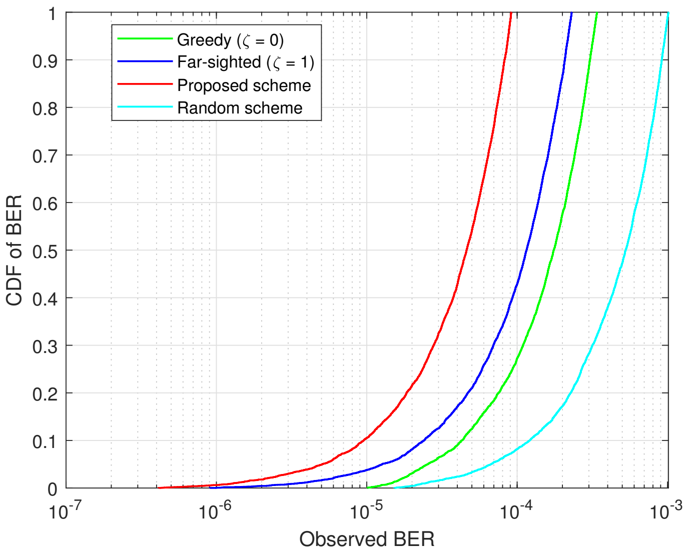

Fig. 11: CDF of BER while considering the maximum BER of all the available links behind the agent vehicle for $\epsilon = 0.05$.

through extensive simulations and compare it with various variants of our scheme as well as schemes based on RF communications. Our system achieves better sum spectral efficiency and lower average latency compared to all the schemes under comparison. By observing the CDF of the experienced latency and BER, we can conclude that our system can satisfy ultra-low latency communication and BER constraints, while the rest of the schemes fail to meet the constraints for large periods of time.

References

[1] P. Papadimitratos *et al.*, “Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation,” *IEEE Commun. Mag.*, vol. 47, no. 11, pp. 84–95, Nov. 2009.

[2] S.-h. Sun *et al.*, “Support for vehicle-to-everything services based on LTE,” *IEEE Wireless Commun.*, vol. 23, no. 3, pp. 4–8, Jun. 2016.

[3] Y. Goto *et al.*, “A new automotive vlc system using optical communication image sensor,” *IEEE Photon. J.*, vol. 8, no. 3, pp. 1–17, Apr. 2016.

[4] T. Yamazato *et al.*, “Image-sensor-based visible light communication for automotive applications,” *IEEE Commun. Mag.*, vol. 52, no. 7, pp. 88–97, Jul. 2014.

[5] I. Takai *et al.*, “LED and CMOS image sensor based optical wireless communication system for automotive applications,” *IEEE Photonics Journal*, vol. 5, no. 5, pp. 6 801 418–6 801 418, Oct. 2013.

[6] I. Takai *et al.*, “Optical vehicle-to-vehicle communication system using LED transmitter and camera receiver,” *IEEE Photon. J.*, vol. 6, no. 5, pp. 1–14, Oct. 2014.

[7] M. I. Ashraf, C.-F. Liu, M. Bennis, and W. Saad, “Towards low-latency and ultra-reliable vehicle-to-vehicle communication,” in *Proc. 2017 EuCNC*, Oulu, Finland, Jun. 2017, pp. 1–5.

[8] W. Sun, E. G. Ström, F. Brännström, Y. Sui, and K. C. Sou, “D2D-based V2V communications with latency and reliability constraints,” in *Proc. 2014 IEEE GC Wkshps*, Austin, TX, USA, Dec. 2014, pp. 1414–1419.

[9] R. S. Sutton and A. G. Barto, *Introduction to reinforcement learning*. MA, USA: MIT press Cambridge, 1998, vol. 135.

[10] M. Tan, “Multi-agent reinforcement learning: Independent vs. cooperative agents,” in *Proc. 10th ICML*, Jun. 1993, pp. 330–337.

[11] C. Claus and C. Boutilier, “The dynamics of reinforcement learning in cooperative multiagent systems,” *AAAI/IAAI*, vol. 1998, no. 746-752, p. 2, Jul. 1998.

[12] P. Luo *et al.*, “Experimental demonstration of RGB LED-based optical camera communications,” *IEEE Photon. J.*, vol. 7, no. 5, pp. 1–12, Oct. 2015.

[13] L. Liang, H. Ye, and G. Y. Li, "Spectrum sharing in vehicular networks based on multi-agent reinforcement learning," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 10, pp. 2282–2292, Aug. 2019.

[14] A. S. Kumar, L. Zhao, and X. Fernando, "Multi-agent deep reinforcement learning-empowered channel allocation in vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 71, no. 2, pp. 1726–1736, Dec. 2021.

[15] M. A. Salahuddin, A. Al-Fuqaha, and M. Guizani, "Reinforcement learning for resource provisioning in the vehicular cloud," *IEEE Wireless Commun.*, vol. 23, no. 4, pp. 128–135, Jun. 2016.

[16] Q. Zheng, K. Zheng, H. Zhang, and V. C. Leung, "Delay-optimal virtualized radio resource scheduling in software-defined vehicular networks via stochastic learning," *IEEE Trans. Veh. Technol.*, vol. 65, no. 10, pp. 7857–7867, Mar. 2016.

[17] A. Ashok *et al.*, "Capacity of screen–camera communications under perspective distortions," *Pervasive Mob. Comput.*, vol. 16, pp. 239–250, Jan. 2015.

[18] Z. Du, C. Wang, Y. Sun, and G. Wu, "Context-aware indoor VLC/RF heterogeneous network selection: Reinforcement learning with knowledge transfer," *IEEE Access*, vol. 6, pp. 33 275–33 284, Jun. 2018.

[19] H. Yang, A. Alphones, W. 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. Industr. Inform.*, vol. 16, no. 8, pp. 5565–5576, Aug. 2020.

[20] 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.

[21] A. Islam, M. T. Hossan, and Y. M. Jang, "Convolutional neural network scheme–based optical camera communication system for intelligent internet of vehicles," *Int. J. Distrib. Sens. Netw.*, vol. 14, no. 4, pp. 1–15, Apr. 2018.

[22] R. V. Terres, "Multi-user MISO for visible light communication," Ph.D. dissertation, University of Virginia, Sep. 2015.

[23] P. Deng, "Real-time software-defined adaptive MIMO visible light communications," *Visible Light Commun.*, pp. 637–640, Jul. 2017.

[24] S. A. I. Alfarozi, K. Pasupa, H. Hashizume, K. Woraratpanya, and M. Sugimoto, "Square wave quadrature amplitude modulation for visible light communication using image sensor," *IEEE Access*, vol. 7, pp. 94 806–94 821, Jul. 2019.

[25] J. M. Kahn and J. R. Barry, "Wireless infrared communication," *Proc. IEEE*, vol. 85, no. 2, pp. 265–298, Feb. 1997.

[26] A. Islam, L. Musavian, and N. Thomos, "Multi-agent deep reinforcement learning in vehicular OCC," in *Proc. 95th Veh. Technol. Conf.*, Helsinki, Finland, Jun. 2022, pp. 1–6.

[27] A. Islam, L. Musavian, and N. Thomos, "Performance analysis of vehicular optical camera communications: Roadmap to uRLLC," in *Proc. 2019 IEEE GLOBECOM*, Hawaii, USA, Dec. 2019, pp. 1–6.

[28] T. Yamazato *et al.*, "Vehicle motion and pixel illumination modeling for image sensor based visible light communication," *IEEE J. Sel. Areas Commun.*, vol. 33, no. 9, pp. 1793–1805, Sep. 2015.

[29] R. Steele and W. T. Webb, "Variable rate QAM for data transmission over Rayleigh fading channel," in *Proc. IEEE Wireless*, Calgary, Canada, Jul. 1991, pp. 1 –14.

[30] D. A. Plaisted, "Some polynomial and integer divisibility problems are NP-HARD," in *Proc. 17th SFCS*, Houston, TX, USA, Oct. 1976, pp. 264–267.

[31] C. H. Papadimitriou and K. Steiglitz, *Combinatorial optimization: algorithms and complexity*. Courier Corporation, 1998.

[32] T. Rashid *et al.*, "QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning," in *Proc. 35th ICML*, Stockholm, Sweden, July 2018, pp. 6846–6859.

[33] P. Sunehag, G. Lever, A. Gruslys, W. M. Czarnecki, V. Zambaldi, M. Jaderberg, M. Lanctot, N. Sonnerat, J. Z. Leibo, K. Tuyls *et al.*, "Value-decomposition networks for cooperative multi-agent learning," *arXiv preprint arXiv:1706.05296*, 2017.

[34] T. Zinchenko, "Reliability assessment of vehicle-to-vehicle communication," doctoralthesis, Technische Hochschule Wildau, 2014.

[35] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge university press, 2004.

[36] N. Mastronarde and M. van der Schaar, "Joint physical-layer and system-level power management for delay-sensitive wireless communications," *IEEE Trans. Mob. Comput.*, vol. 12, no. 4, pp. 694–709, Feb. 2012.

[37] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, no. 7540, pp. 529–533, Feb. 2015.

[38] N. Salodkar, A. Bhorkar, A. Karandikar, and V. S. Borkar, "An on-line learning algorithm for energy efficient delay constrained scheduling over a fading channel," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 4, pp. 732–742, Apr. 2008.

[39] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO-simulation of urban mobility," *Int. J. on Advances in Systems and Measurements*, vol. 5, no. 3&4, Dec. 2012.

[40] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2020.

[41] M. Abadi *et al.*, "Tensorflow: A system for large-scale machine learning," in *Proc. 12th USENIX Symp. Operating Syst. Design and Implementation*, Savannah, GA, Nov. 2016, pp. 265–283.

Amirul Islam received his PhD degree in Computing and Electronic Systems from the University of Essex, UK, in 2022. He is currently working as a Postdoctoral researcher at the Visual Artificial Intelligence Laboratory of Oxford Brookes University, UK. His research interests include machine learning for communication, optical camera communication, deep reinforcement learning, automotive vehicular communications, and optimization strategies.

Nikolaos Thomos (S'02,M'06,SM'16) received the Diploma and Ph.D. degrees from the Aristotle University of Thessaloniki, Greece, in 2000 and 2005, respectively. He was a Senior Researcher with the Ecole Polytechnique Federale de Lausanne (EPFL) and the University of Bern, Switzerland. He is currently a Professor with the University of Essex, U.K.. His research interests include machine learning for communications, multimedia communications, network coding, information-centric networking, source and channel coding, device-to-device communication, and signal processing. He is an elected member of the IEEE MMSP Technical Committee (MMSP-TC) for the period 2019-2024. He received the highly esteemed Ambizione Career Award from the Swiss National Science Foundation (SNSF).

Leila Musavian received her PhD degree in Telecommunications at Centre for Telecommunications Research (CTR), Kings College London, UK. She is currently working as Professor of Wireless Communications at University of Essex. She was Deputy Pro-Vice-Chancellor for Research at University of Essex between September 2018 and December 2020 and Reader in Telecommunications at the School of Computer Science and Electronic Engineering from Dec 2016-Oct 2020. Prior to that, she was Lecturer at InfoLab21, Lancaster University (Dec 2012-Aug 2016), Senior Lecturer at InfoLab21, Lancaster University (Aug 2016-Nov 2016), Research Associate at McGill University, Canada (2011-2012), research associate at Loughborough University, UK (2009-2010) and post-doctoral fellow at INRS-EMT, Canada (2006-2008).