Enhancing the Fuel-Economy of V2I-Assisted Autonomous Driving: A Reinforcement Learning Approach

Xiao Liu, *Student Member, IEEE,* Yuanwei Liu, *Senior Member, IEEE,*
Yue Chen, *Senior Member, IEEE,* and Lajos Hanzo, *Fellow, IEEE,*

***Abstract*—A novel framework is proposed for enhancing the driving safety and fuel economy of autonomous vehicles (AVs) with the aid of vehicle-to-infrastructure (V2I) communication networks. The problem of driving trajectory design is formulated for minimizing the total fuel consumption, while enhancing driving safety (by obeying the traffic rules and avoiding obstacles). In an effort to solve this pertinent problem, a deep reinforcement learning (DRL) approach is proposed for making collision-free decisions. Firstly, a deep Q-network (DQN) aided algorithm is proposed for determining the trajectory and velocity of the AV by receiving real-time traffic information from the base stations (BSs). More particularly, the AV acts as an agent to carry out optimal action such as lane change and velocity change by interacting with the environment. Secondly, to overcome the large overestimation of action values by the Q-learning model, a double deep Q-network (DDQN) algorithm is proposed by decomposing the max-Q-value operation into action selection and action evaluation. Additionally, three practical driving policies are also proposed as benchmarks. Numerical results are provided for demonstrating that the proposed trajectory design algorithms are capable of enhancing the driving safety and fuel economy of AVs. We demonstrate that the proposed DDQN based algorithm outperforms the DQN based algorithm. Additionally, it is also demonstrated that the proposed fuel-economy (FE) based driving policy derived from the DRL algorithm is capable of achieving in excess of 24% of fuel savings over the benchmarks.**

***Index Terms*—Autonomous driving, deep reinforcement learning, fuel consumption, safe driving, trajectory design, V2I communications**

I. Introduction

Autonomous driving, which promises both increased traffic safety and traffic efficiency [1] has been the focal point of the automotive research field for a while [2, 3], since it is capable of improving the fuel economy, whilst minimizing travel time [4]. At the time of writing, the driving safety of autonomous vehicles (AVs) is mainly based on onboard sensors and radars [5, 6]. Even though the feasibility of this approach has already been demonstrated in field tests by Alphabet Inc, some impediments still persist. For instance, costly onboard units (OBUs) limit commercialization for the mass-market, while the AVs are unable to cooperate in complex traffic environments [1]. On the other hand, invoking shared roadside units (RSUs) to supplement OBUs is excepted to promote the large-scale spreading of autonomous driving.

Vehicle-to-infrastructure (V2I) communications support the interaction of road users, RSUs, pedestrians and other local paraphernalia for enhancing road safety and traffic efficiency [7]. In V2I-assisted autonomous driving, the V2I components either complement or replace the OBUs, thus allowing AVs to receive reliable real-time traffic information from the roadside base stations (RBSs), which facilitates the interaction among AVs and road users, hence enhancing their safety and traffic efficiency. The RBSs report to the AVs the presence of obstacles that AVs cannot directly sense in non-line-of-sight scenarios. Whilst fog and sun glare limit the performance of certain onboard sensors, V2I-assisted autonomous driving remains reliable in arbitrary weather or lighting conditions.

V2I-aided autonomous driving has hence received tremendous research interests both in academia and industry [8, 9]. In 2011, Japanese engineers have implemented V2I through the deployment of the intelligent transportation system (ITS) of [10]. Similarly, scientists in the Netherlands, Germany, and Austria are working on developing a European smart corridor that will provide drivers with information on road-works and oncoming traffic [11]. Furthermore, the vehicle manufacturers General Motors, Toyota, Tesla, and Nissan, among others, are also actively promoting the development of communication-based solutions to support safe driving.

A. State-of-the-art

Again, autonomous driving has attracted remarkable attention in recent years in diverse traffic scenarios, including lane changing [12, 13], reverse parking [14], parallel parking [15],

This paper will be presented in part at the IEEE International Conference on Communications, June 2020.

X. Liu, Y. Liu, and Y. Chen are with the School of Electronic Engineering and Computer Science, Queen Mary University of London, London E1 4NS, UK. (email: x.liu@qmul.ac.uk; yuanwei.liu@qmul.ac.uk; yue.chen@qmul.ac.uk).

L. Hanzo is with the School of Electronics and Computer Science, University of Southampton, SO17 1BJ, UK (e-mail:lh@ecs.soton.ac.uk).

L. Hanzo would like to acknowledge the financial support of the Engineering and Physical Sciences Research Council projects EP/N004558/1, EP/P034284/1, EP/P034284/1, EP/P003990/1 (COALESCE), of the Royal Society's Global Challenges Research Fund Grant as well as of the European Research Council's Advanced Fellow Grant QuantCom.

Future research ideas: 1) designing the multi-agent deep reinforcement learning based algorithm for enhancing the driving safety and fuel economy of multiple AVs; 2) designing the aerial-ground cooperative vehicular networks, where vehicle-to-infrastructure (V2I) communications, vehicle-to-vehicle (V2V) communications and UAV-to-vehicle (U2V) communications are considered for enhancing the driving safety of the AVs.

longitudinal control [16], computing resource allocation [17] and motion/trajectory design [18–22]. Among all these challenges, the planning of the AVs' trajectory is fundamental.

1) Trajectory Design of Autonomous vehicles: To design trajectories for AVs while ensuring safety, the authors of [19] proposed a Reachability-based Trajectory Design (RTD) method, and demonstrated that RTD was eminently applicable to passenger vehicles, where the powertrain, chassis, and tires jointly obey nonlinear dynamics. The authors of [20] proposed a beneficial technique of optimizing the vehicular trajectories, while considering the vehicle's kinematic limits and ensuring collision avoidance. Both continuous-time and discrete-time vehicle trajectory planning were considered.

In terms of fuel-efficiency and eco-driving oriented designs for autonomous vehicles, the authors of [23, 24] proposed cooperative look-ahead control strategy based on the classic distributed model predictive control (DMPC) approach. The fuel-efficiency of a vehicular platoon was maximized by optimizing the speed and motion trajectories of the autonomous vehicles with the aid of a particle swarm optimization algorithm. In [25], a switched control strategy of a heterogeneous vehicular platoon was proposed for improving the safety, passenger comfort, formation control and fuel economy of intelligent vehicles.

2) V2X-Aided Autonomous Driving: In order to reap the full benefits of communication-aided autonomous driving, the authors of [26] considered the control of the instantaneous velocity of the AV under realistic communications constraints. The authors of [27] and [28] invoked an 802.11n/g wireless network for transferring sensed data amongst two to three vehicles. As a benefit, the AV became capable of receiving real-time traffic information from other vehicles. The authors of [29] considered the problem of cooperative driving for connected AVs to reduce traffic congestion. The connected AVs became capable of exchanging information as well as driving intentions with the surrounding vehicles through vehicle-to-vehicle (V2V) communications. To analyze the performance of a heterogeneous V2X communication network operating in the downlink direction, the authors of [30] proposed a network slicing method for supporting the connectivity of the next-generation devices. their simulation results demonstrated huge improvements in terms of reliability and throughput, which is due to the utilization of high-quality V2V and V2I links.

3) Reinforcement Learning in Autonomous Driving: Machine learning has gained remarkable attention in wireless communication networks [31, 32]. Reinforcement learning algorithms have been shown to be capable of tackling problems in autonomous driving systems [33–35]. The authors of [36] invoked a beneficial combination of deep reinforcement learning (DRL) and numerical solutions for controlling the instantaneous velocity of the AV. As a result, the maximal possible speed was obtained, while collisions were avoided. The authors of [37] learned the optimal driving policy for an AV in a pair of practical intersection scenarios with the aid of reinforcement learning. The authors of [38] invoked reinforcement learning for training an AV for safe driving. In order to design a suitable cost function and to solve the associated optimal control problem in real-time, the authors of [39] designed a control architecture based deep RL framework for safe decision making in autonomous driving. By appropriately modifying the learning algorithm, the resultant continuous adaptation framework became capable of reducing the number of unnecessary safety triggers.

B. *Motivations*

As mentioned above, the costly OBUs of the autonomous driving system limit the attainable cost reduction of AVs. As an impediment, the vehicles are unable to cooperate in complex propagation and traffic scenarios [1]. As a remedy, we consider a V2I-assisted autonomous driving scenario, in which shared RSUs are invoked for complementing the OBUs for collecting real-time traffic information. This, in turn, is expected to promote the large-scale private ownership of AVs.

The aforementioned research contributions conceived solutions for enhancing both the safety and fuel-economy of AVs by both conventional and machine learning schemes. However, predominantly the driving safety of the AV was considered. By contrast, only a few papers analyzed the energy efficiency of the AV by considering their energy management [37, 39]. However, since the fuel consumption is a function of both the velocity and acceleration, their control has to be considered, when aiming for fuel economy. Unfortunately, there is still a paucity of research contributions on studying the fuel economy of AVs relying on V2I communications. Thus, we formulate the trajectory design problem of the AV for minimizing fuel consumption while enhancing driving safety.

In terms of the methodology, the V2I-assisted autonomous driving scenario is naturally a highly dynamic one, which constitutes quite a challenge for conventional optimization algorithms. But fortunately, machine learning (ML) comes to rescue. For example, reinforcement learning can be used for empowering agents by interacting with the environment and by learning from their mistakes. Since it is non-trivial to pose autonomous driving as a supervised learning problem due to strong interactions with the environment including other vehicles, RBSs pedestrians, and RSUs. The RL model is capable of monitoring the reward resulting from its actions, thus it is chosen for solving the trajectory design problem of V2I-assisted autonomous driving systems.

C. *Contributions*

Against this background, the primary contributions of this paper are as follows:

- We propose a novel framework for optimizing the fuel economy and driving safety of the AV with the aid of V2I communications, where the AV receives real-time traffic information from the RBSs. Based on the proposed framework, we formulate a fuel economy optimization problem by jointly designing the path, motion and instantaneous velocity of the AV.
- We adopt a deep Q-network (DQN) based algorithm for acquiring the trajectory of the AV under the quality-of-connectivity constraint of the communication link. In the DQN model, the AV acts as a learning agent and the trajectory of the AV corresponds to the actions taken

by itself. At each timeslot, the AV receives a reward or penalty according to the specific driving safety and fuel consumption encountered. Finally, the AV is capable of attaining a collision-free driving policy for obtaining fuel economy by interacting with the environment and by learning from its mistakes.

- We conceive a double deep Q-network (DDQN) based algorithm for preventing the over-estimation of the action values by the conventional Q-learning model. The max operation process in the conventional DQN model is decomposed into the twinned processes of action selection and action evaluation in the DDQN model. More particularly, the first DQN is used for selecting an action to interact with the environment, while the second for evaluating the action selected. Additionally, we prove that the proposed DDQN algorithm is capable of converging under mild conditions.
- We demonstrate that the proposed DDQN algorithm outperforms the DQN algorithm both in terms of its convergence rate and performance. Additionally, the fuel-economic driving policy derived from the DDQN algorithm saves about 24% fuel compared to three other practical driving policies, which are also proposed as benchmarkers.

The rest of the paper is organized as follows. In Section II, the problem formulation of fuel economy is presented for AVs. In Section III, the proposed DQN and DDQN based algorithms conceived for solving the problem formulated are demonstrated. Our numerical results are presented in Section V, which is followed by our conclusions in Section VI.

II. System Model

Again, we consider a V2I-assisted autonomous driving model, which mainly includes RBSs, AVs, OBUs, and RSUs. The OBUs which are installed in vehicles, receive real-time traffic information from RBSs and instantaneously report their velocity, driving directions and other vehicular information. The RSUs, relying on the static facilities at the roadside, are invoked for collecting traffic information concerning the complex scenarios of vehicles, pedestrians and the road. Meanwhile, the traffic information is sent by the RSUs to RBSs via the Internet. The RBSs also adopt a high-performance processor for processing both the information and data. As illustrated in Fig. 1, the RBSs provide network coverage for a particular section of the highway. Low-latency communication for supporting safe driving for the AV is served by the RBSs[1].

We aim for designing the trajectory (including the path, motion and real-time velocity) of the AV by jointly considering the driving safety and fuel consumption. In terms of safety, collisions have to be avoided, while the traffic rules also have to be observed. Finally, fuel economy has to be maintained.

A. Transmission Model

We assume that $u(t) = [x(t), y(t)]$ represents the coordinates of the AV at timeslot t. The target of the AV is that of driving from the initial position u_0 to the final destination u_f. We assume that the location of each RBS is fixed and known. The coordinates of the m-th RBS are (x_m, y_m, h_m), $m \in \mathcal{M} \triangleq \{1, 2, \cdots M\}$, while h_m represents the height of the m-th RBS. Then, the distance between the m-th RBS and the AV can be expressed as

$$d_m(t) = \sqrt{[x(t) - x_m]^2 + [y(t) - y_m]^2 + h_m^2}. \tag{1}$$

In a highway scenario, a dominant Line-of-Sight (LoS) component is expected, associated with the most appropriate transmitter antenna height. A basic signal propagation model capturing channel gain from the m-th RBS to the AV is formulated as [40]

$$g_m(t) = {K_0}^{-1} d_m^{-\alpha}(t) |h_m(t)|^2, \tag{2}$$

where we have $K_0 = \left(\frac{4\pi f_c}{c}\right)^2$ and α represents the path loss exponent, while $|h_m(t)|$ corresponds to small-scale Rician fading component [41].

Let s_m represent the vehicle-RBS indicator, where $s_m = 1$ indicates that the AV is served by the m-th RBS, while $s_m = 0$ otherwise. Thus, the received signal-to-noise ratio (SNR) of the AV is expressed as follows

$$\mathrm{SNR}(t) = \frac{s_m(t) g^m(t) P_m}{\sigma^2}, \tag{3}$$

where P_m represents the power transmitted from the m-th RBS to the AV, while $\sigma^2 = B_{\mathrm{AV}} N_0$ with B_{AV} and N_0 denoting the bandwidth and power spectral density of the additive white Gaussian noise (AWGN), respectively.

The Doppler frequency f_d depends on the relative velocity between the AV and the connected RBS [42]. Thus, the Doppler frequency of the AV with a velocity of $v(t)$ can be expressed as

$$f_d = \frac{v(t)}{c} \cdot f_c \cdot \cos\alpha_v, \tag{4}$$

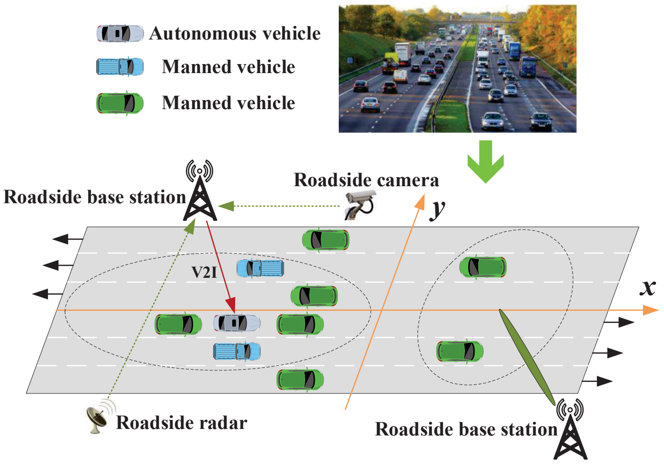


Fig. 1: Illustration of V2I assisted autonomous driving over a section of highway.

[1] In this paper, we assume that OBUs only receive traffic information from the RBSs. In our future research, OBUs are capable of receiving traffic signals, road construction, safety accidents information from both RSUs and RBSs. Meanwhile, the OBUs can exchange information with RSUs, RBSs or other OBUs in a highly dynamic environment.

where α_v represents the angle between the direction of arrival of the considered propagation path and the AV's movement.

Remark 1. *Given the location of BSs, increasing the instantaneous velocity of the AV leads to reduced SNR. In an effort to enhance driving safety, the instantaneous velocity of the AV has to be reduced, when the propagation environment is poor.*

B. Safe Driving Model

We consider delay-sensitive communication between the RBS and the AV under a specific quality-of-connectivity constraint specified by a minimum received SNR requirement, which has to be satisfied at all times along the vehicle's motion trajectory [1]. Further enhancement for the received SNR by leveraging non-orthogonal multiple access (NOMA) [43, 44] will be left for our future research. The minimum SNR target can be expressed as

$$\Gamma(t) = \mathrm{SNR}(t) \geq \overline{\Gamma}. \tag{5}$$

Lemma 1. *In order to ensure that every AV is capable of receiving real-time traffic information from the RBS, the upper bound for the distance between two RBS has to satisfy*

$$d_{GB} \leq \frac{2v^m(t)P_m|h_m(t)|^2}{K_0\overline{\Gamma}\sigma^2}. \tag{6}$$

Lemma 1 sets out the upper bound of the distance between two RBSs for guaranteeing that the AV's instantaneous transmit rate is sufficiently high for receiving real-time traffic information, since this is a vital precondition of safe driving in autonomous driving scenarios.

To maintain safe driving, we assume that the distance between the AV and the manned vehicles in the same lane has to obey the Three-Second Rule[2]. Hence, the safe driving distance has to satisfy

$$\begin{aligned} d_k &= x(t) - x_k(t) \\ &\geq 3 \cdot v(t) + 1/2(l + l_k), \end{aligned} \tag{7}$$

where $x_k(t)$ and l_k represents the coordinate and length of the k-th manned vehicle on the road, respectively, and $k \in \mathcal{K} \triangleq \{1, 2, \cdots K\}$. Furthermore, l denotes the length of the AV.

C. Fuel Consumption Function

We utilize a fuel consumption rate model, which considers both the energy required for overcoming all the forces of resistance and for episodic accelerations. The conception of a general fuel consumption model will be left for our future research.

The instantaneous energy expenditure $e(t)$ on a flat road is expressed as a sum of $e(t) = e_1(t) + e_2(t)$, where $e_1(t)$ is the energy required for overcoming all the forces of resistance, and $e_2(t)$ is the kinetic energy required for episodic acceleration.

Explicitly, $e_1(t)$ is given by Equation (8) at the top of the next page, where η_T is the efficiency of the power-train transmission; m_v is the car-mass; f_r is the rolling resistance coefficient; c_D is the coefficient of aerodynamic resistance of the car; and A_f is the characteristic area of the car. Furthermore, $\eta_j\,(P, n)$ is the engine efficiency, which depends on the degree of power utilization and on the engine speed model; ρ_a is the air density; g is the acceleration coefficient; $v_i(t)$ is the instantaneous vehicular speed at the i-th acceleration subinterval; v_j is the vehicular speed at the j-th constant speed subinterval; T_j is the j-th constant speed subinterval duration; T_i is the i-th acceleration subinterval duration. If the rolling resistance coefficient and the vehicle's frontal area are not provided by the manufacturer, their approximate values may be estimated from the following empirical equations [45]

$$\begin{aligned} f_r &= 0.0136 + 0.40 \cdot 10^{-7} \cdot v^2(t), \\ A_f &= 1.6 + 0.00056 \cdot (m_v - 756). \end{aligned} \tag{9}$$

The energy required for increasing the kinetic energy during accelerations is determined via the following relationship [45]

$$e_2(t) = \frac{m_v \cdot \gamma_m}{2} \sum_{i=1}^{I_2} \sum_{t=0}^{T_k} \frac{a_i(t)}{\eta_i(P, n, t)}, \tag{10}$$

where γ_m represents the mass factor of the car, which equivalently converts the rotational inertia of the rotating components into translation mass; and $a_i(t)$ denotes the instantaneous vehicular acceleration at the i-th acceleration subinterval. The instantaneous engine efficiency $\eta(P, n, t)$ is also calculated in [45].

The energy consumption of a vehicle also depends on the road condition such as the road's slope. When taking the road slope into consideration, the fuel consumption of a vehicle can be obtained as [46]

$$\begin{aligned} \hat{e}(t) = b_0 + b_1 v(t) + b_2 v^2(t) + b_3 v^3(t) \\ + \hat{a}\left[c_0 + c_1 v(t) + c_2 v^2(t)\right], \end{aligned} \tag{11}$$

where b_0, b_1, b_2, b_3, c_0, c_1 and c_2 are constant values given in [46], $\widehat{a} = a_V + a_\theta$ is the sum of the apparent acceleration of the vehicle and the acceleration required to counteract the road slope ($a_\theta = g\sin(\theta)$). The apparent acceleration of the vehicle can be expressed as [46]

$$a_V = -(1/2M)C_D\rho_a A_f v^2(t) - f_r g\cos(\theta) - g\sin(\theta) + u(t), \tag{12}$$

where $u(t)$ denotes the control input, while the other parameters are the same as in Equation (8).

It is worth noting that the fuel consumption model is invoked for formulating the fuel consumption of the AV. Neither the design of the communication protocol nor the proposed algorithms will be affected by the specific choice of the fuel consumption model. Hence, any fuel consumption model can be invoked in our proposed approach. In the scenario of this paper, it is assumed that the road slope is 0, and the fuel consumption model derived in [45] is invoked.

[2] The three-second rule, also known as the two-second rule in some states of USA, is a rule of thumb by which a driver may maintain a safe trailing distance at any speed. The rule is that a driver should ideally stay at least three seconds behind any vehicle that is directly in front.

$$
\begin{aligned}
e_1(t) = \tfrac{1}{\eta_T} \sum_{j=1}^{I_1} \tfrac{1}{\eta_j(P,n)} \left[m_v \cdot g \cdot f_r \cdot \tfrac{v_j}{3.6} + 0.5 \cdot \rho_a \cdot C_D \cdot A_f \cdot \left(\tfrac{v_j}{3.6}\right)^3 \right] T_j + \\
\tfrac{1}{\eta_T} \sum_{i=1}^{I_2} \int_{T_i} \tfrac{1}{\eta_j(P,n,t)} \left[m_v \cdot g \cdot f_r \cdot \tfrac{v_i(t)}{3.6} + 0.5 \cdot \rho_a \cdot C_D \cdot A_f \cdot \left(\tfrac{v_i(t)}{3.6}\right)^3 \right] dt,
\end{aligned}
\tag{8}
$$

Therefore, the cumulative fuel consumption E_{total} for the AV during the driving period is expressed as

$$E_{\text{total}} = \sum_{t=1}^{T} e(t), \tag{13}$$

where T is the mission completion time.

Remark 2. *According to equation* (8),(10) *and* (13)*, it can be observed that minimizing the mission completion time directly affects the total fuel consumption. However,* $e(t)$ *is not a linear function of the instantaneous velocity and acceleration. Indeed, in practice, maximizing the instantaneous velocity tends to increase the total fuel consumption for all vehicles. This phenomenon emphasizes the importance of velocity-control, when aiming for fuel economy.*

III. Problem Formulation and Proposed Algorithm

In this section, we formulate the optimization problem as minimizing the total fuel consumption of the AV during the mission completion time, while enhancing driving safety. A deep reinforcement learning based trajectory design algorithm is proposed for solving the problem formulated.

A. Problem Formulation

Let $U = \{u(t), 0 \le t \le T\}$, $V = \{v(t), 0 \le t \le T\}$. Again, we aim for minimizing the total fuel consumption E_{total} by optimizing the discrete position $u(t)$ and instantaneous velocity $v(t)$ by receiving real-time traffic information from a RBS. Thus, the optimization problem is formulated as

$$
\begin{aligned}
\min_{U,V} \quad & E_{\text{total}} = \sum_{t=1}^{T} e(t) && \text{(14a)}\\
\text{s.t.} \quad & V_{\min}^{l} \le v(t) \le V_{\max}, \forall t, \forall l, && \text{(14b)}\\
& 0 \le |a_v(t)| \le a_{\max}, \forall t, && \text{(14c)}\\
& \Gamma(t) \ge \overline{\Gamma}, \forall t, && \text{(14d)}\\
& u(0) = u_0, && \text{(14e)}\\
& u(T) = u_f, && \text{(14f)}\\
& d_k \ge 3 \cdot v(t) + 1/2(l + l_k), \forall t, && \text{(14g)}
\end{aligned}
$$

where $\Gamma(t)$ represents the received SNR of the AV, $v(t)$ denotes the velocity at time slot t, $v_{\min}^{l}$ is the minimal velocity stipulation in the l-th lane, d_k represents the distance between the AV and other manned vehicles driving in front of it, and finally $a_v(t)$ denotes the acceleration of the AV at time slot t. Equation (14b) formulates the velocity limitation of the AV; (14c) indicates the acceleration limitation of the AV; (14d) represents the quality-of-connectivity of the communication link; (14e) and (14f) denote the initial position and the final position of the AV, respectively; finally, (14g) represents the safe driving constraint for the AV. Again, we aim for designing the trajectory (including the path, motion, and real-time velocity) of the AV, which ensures that the AV is at the optimal position and optimal instantaneous velocity during each time slot. This, in turn, leads to the reduction of fuel consumption for the AV, while enhancing driving safety.

Problem (14a) is challenging, since the objective function (OF) is combinatorial and non-convex. Additionally, the conventional Q-learning algorithm cannot be directly applied, since the state-action space of the proposed scenario is excessive. The reason is that a large number of states will be visited infrequently, hence the corresponding Q-value will only be rarely updated. This phenomenon leads to an excessive time for the model to converge. To solve this problem at a low complexity, a deep reinforcement learning algorithm will be invoked in the following section to obtain an efficient solution.

B. Proposed DQN Based Trajectory Design Algorithm

In this section, a deep Q-learning based algorithm is proposed for determining the trajectory of the AV, while enhancing driving safety with the aid of V2I communication networks. It is assumed that the AV starts from a random lane. When an obstruction (such as a vehicle in front of the AV with a lower velocity) appears, the AV changes lane (both fast-traffic-lane and carriage lane) to finish overtaking. In this scenario, the AV considers its choices according to the resultant fuel consumption. Since the velocity lower bound of different lanes varies, the AV has to observe the traffic rules for enhancing driving safety for all the surrounding manned vehicles.

In the deep Q-network based model, the AV acts as the agent. At each time slot t throughout the iterations, the AV observes a state, s_t, from the state space, S. The state space

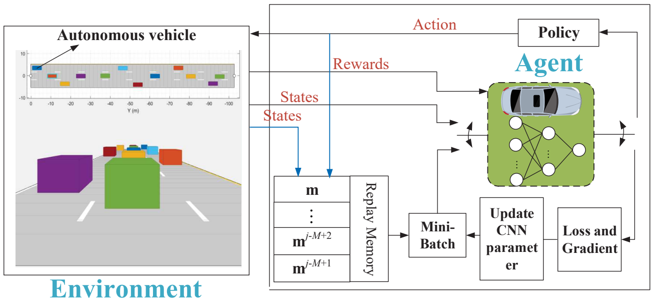


Fig. 2: Deep Q-network (DQN) based trajectory design for autonomous driving on a highway.

consists of the coordinates and instantaneous velocities of both the AV and of other manned vehicles. Accordingly, the AV carries out an action, a_t, from the action space, A, selecting the optimal choice based on policy, J. The action space consists of changing lanes and varying the velocity. The decision policy J is determined by a Q-function, $Q(s_t, a_t)$. The principle of the policy at each time slot is to choose a specific action that results in the maximum Q-value. Following the chosen action, the state of the AV traverses to a new state s_{t+1} and the AV receives a reward, r_t, determined by its safe driving condition and fuel consumption.

Remark 3. *Deep reinforcement learning models may be trained by interacting with the environment (states). They can be expected to find the optimal behaviors (actions) of the AV (agent) by iteratively exploring the environment and by learning from its mistakes. The model is capable of monitoring the reward resulting from its actions.*

As for the state space in the DQN model, it consists of four parts: the current coordinate of the AV, the current velocity of the AV, the current coordinates of other manned vehicles, and the current fuel consumption of the AV. As for the action space of the DQN model, the main actions of autonomous driving are represented by assigning both lane changes and velocity changes to the AV in a way that driving safety is enhanced. On the one hand, when the AV changes lane, (0; $-w$) means that the AV moves to the right lane, (0; w) indicates that the AV moves to the left lane, (0; 0) represents that the AV continues driving straight. In a practical driving scenario, the AV is capable of adopting the steering angle of $[-90^o, 90^o]$, which makes the problem non-trivial to solve. However, by constraining its mobility to as few as 3 directions, a tradeoff between the accuracy and complexity may be struck. On the other hand, the AV may also change its instantaneous velocity in order to satisfy the restrictions imposed. By constraining the acceleration and deceleration to as few as 5 levels (-10m/s^2, -5m/s^2, 0m/s^2, 5m/s^2, 10m/s^2), the mobility of the AV is further simplified. It is assumed that only one action will be carried out at each timeslot, while composite actions are not allowed.

The beneficial design of the associated reward/penalty function requires a sophisticated methodology for accelerating the convergence of the model, which is directly related to the safety, expected velocity and fuel consumption of the AV. In this case, the reward/penalty in the DQN model is a function of the AV's position, acceleration, velocity and fuel consumption rate, which is expressed as $r(t) = f[y(t), v(t), a(t), e(t)]$. When the AV is driving at a low fuel consumption rate while guaranteeing safe driving, it receives a positive reward. By taking any other actions, which may lead to an increase of the fuel consumption, collision or traffic violation, the AV receives a penalty. In terms of the penalty function of collision avoidance, since the collision damage of vehicles is proportional to the square of the relative velocity when an accident happens, we design the penalty value as a function of the square of the relative velocity. Using an appropriate penalty function is also essential for enhancing compliance with the specific traffic rules mentioned above. To this end, the penalty function of traffic rule violations is designed as a function of the AV's velocity. By contrast, the penalty function of fuel economy is designed based on the fuel consumption derived from Equations (8), (10) and (13). As for the reward function, when the AV carries out an action for fuel-saving and safe driving, a constant reward value is achieved. A typical reward function can be formulated as (15) at the top of the this page.

Remark 4. *The penalty function for collision avoidance has to have a high value, so that the potentially unsafe motions are suppressed.*

Algorithm 1 Deep Q-network based trajectory design algorithm for autonomous driving

```
Input:
  Q-network structure, environment simulator, replay memory
  $D$, minibatch size $n$.
  Initialize the replay memory $D$, Q-network weights $\theta$,
  weights of the target network $\theta^* = \theta$, and $Q(s, a)$.
    The AV starts at random points.
  repeat
    for each step of episode:
        The AV chooses $a_t$ uniformly from $A$ with probability of $\varepsilon$, while chooses $a_t$ such that $Q_\theta(s_t, a_t) = \max_{a \in A} Q_\theta(s_t, a_t)$ with probability of $(1 - \varepsilon)$.
      The AV carries out action $a_t$, and observes reward $r_t$,
      The model updates state $s_{t+1}$;
      Store transition $(s_t, a_t, r_t, s_{t+1})$ and sample random minibatch of transitions $(s_i, a_i, r_i, s'_i)_{i \in n}$ from $D$;
      For each $i \in I$, we can obtain
          $y_i = r_i + \gamma \cdot \max_{a \in A} Q_{\theta^*}(s'_i, a)$;
      Perform a gradient descent step
      $\theta \leftarrow \theta - a_t \cdot \frac{1}{I} \sum_{i \in n} [y_i - Q_\theta(s_i, a_i)] \cdot \nabla_\theta Q_\theta(s_i, a_i)$;
      $\theta \leftarrow \theta^*$.
    end
  until $s$ is terminal
Return: Action-value function $Q_\theta$ and policy $J$.
```

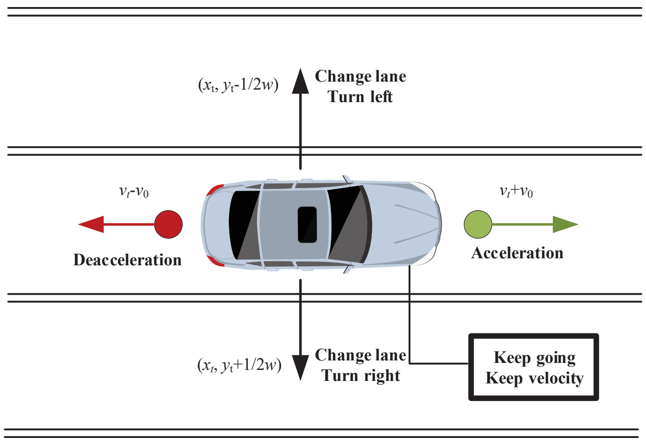


Fig. 3: Action definition for the AV.

$$r_t = \begin{cases} c_1 |v(t) - v_0(t)|^2, & \text{collision}, \\ c_2 \left[100 - v(t)\right], & \text{no collision}, y(t) = -1/2w, v(t) \leq 100km/h, \\ c_2 \left[80 - v(t)\right], & \text{no collision}, y(t) = -3/2w, v(t) \leq 80km/h, \\ c_2 \left[60 - v(t)\right], & \text{no collision}, y(t) = -5/2w, v(t) \leq 60km/h, \\ -\int_t^{t+1} e(t)dt, & \text{no collision}, y(t) = -1/2w, v(t) \geq 100km/h, E(t+1) \geq E(t), \\ -\int_t^{t+1} e(t)dt, & \text{no collision}, y(t) = -3/2w, v(t) \geq 80km/h, E(t+1) \geq E(t), \\ -\int_t^{t+1} e(t)dt, & \text{no collision}, y(t) = -5/2w, v(t) \geq 60km/h, E(t+1) \geq E(t), \\ c_3, & \text{otherwise}. \end{cases} \tag{15}$$

During the process of learning, the state-action value function for the agent can be iteratively updated as follows

$$\begin{aligned} Q_{t+1}(s_t, a_t) \leftarrow & (1 - \alpha) \cdot Q_t(s_t, a_t) \\ & + \alpha \cdot \left[r_t + \gamma \cdot \max{}_a Q_t(s_{t+1}, a)\right], \end{aligned} \tag{16}$$

where α represents the learning rate, while γ denotes the discount factor.

Remark 5. *The learning rate α in the DQN model determines the rate of information updates, while striking a compromise between averaging over possible randomness in the rewards and allowing for transitions in order to converge to the optimal action-value function. Explicitly, it has to be appropriately adopted according to the specific environment, such as $\alpha = 1$ for fully deterministic environments, and $\alpha < 1$ for stochastic scenarios.*

In (16), the reward r_t is drawn from a fixed reward distribution $R : S \times A \rightarrow \mathbb{R}$, where $E\left\{r_t \left|(s, a, s') = (s_t, a_t, s_{t+1})\right.\right\} = R_{sa}^{s'}$. The optimal value function is the solution to the following equation

$$Q^*(s, a) = \mathbb{E}_{s'} \left[r + \gamma max_{a'} Q^*(s', a') \left|s, a\right.\right], \tag{17}$$

where $Q^*(s, a)$ is the desired value function such that $Q \rightarrow Q^*$ when $i \rightarrow \infty$.

The DQN algorithm improves the conventional Q-learning algorithm by combining it with the Convolutional Neural Networks (CNN). The Q-table is approximated by a CNN having weights $\{\theta\}$ as a Q-function. Once $\{\theta\}$ is determined, the Q-values $Q(s_t, a_t)$ constitute the outputs of the CNN.

The DQN model updates its weights, θ, at each iteration for minimizing the following loss function derived from the same Q-network with the aid of the previous weights,

$$\text{Loss}(\theta) = \sum \left[y - Q\left(s_t, a_t, \theta\right)\right]^2, \tag{18}$$

where we have $y = r_t + \gamma \cdot \max\limits_{a \in A} Q_{\text{old}}\left(s_t, a_t, \theta\right)$.

As illustrated in Fig.2, the CNN includes a pair of convolutional layers and two fully connected layers. The AV's moving experience characterized by $ex(t) = y(t), v(t), a(t), e(t)$ is stored in the memory denoted by D, with $D = \{ex(0), \cdots ex(T)\}$. The AV then invokes an experience replay technique to extract the moving experience $ex(t)$ from the memory at each time slot. To elaborate a little further, the AV randomly selects a selection of M experiences from the memory D during the experience replay, which we refer to as a minibatch of experience. The weights of the CNN $\theta(t)$ are updated according to the classic stochastic gradient descent (SGD) algorithm, and the M minibatch elements are chosen similar to [47].

We strike a balance between the exploration and exploitation in DQN by using ϵ-greedy exploration [48]. More specifically, the policy J that maximizes the Q-value is chosen with a high probability of $(1 - \epsilon)$, while other actions are selected with a low probability to avoid getting trapped in a local optimum, which is formulated as

$$Pr(J = \widehat{J}) = \begin{cases} 1 - \epsilon, & \widehat{a} = \text{argmax} Q\left(s, a\right), \\ \epsilon / \left(|a| - 1\right), & otherwise. \end{cases} \tag{19}$$

C. The Proposed DDQN Based Trajectory Design Algorithm

One of the limitations of the Q-learning algorithm is that this model may suffer from the overestimation of action values, because it invokes the maximum action value as an approximation of the maximum expected action value. It is noted in [48] that the use of the same network weight θ for both the policy decision and for the Q value approximation leads to a nonzero lower limit of $\max_a Q(s, a)$

$$\max{}_a Q(s, a) \geq V^*(s) + \sqrt{\frac{C}{m - 1}}, \tag{20}$$

where V^* represents a specific state, in which all the true optimal action values are equal at $V^*(s) = Q^*(s, a)$; C is the variance of the state value and m denotes the number of optional actions in state s. Therefore, the idea of double Q learning is applied, in which the original weights θ_t are still applied, when the DQN is deciding about a particular action, and the second set of weights ${\theta_t}'$ is adopted for approximating the objective function.

In the double Q-learning model, two Q functions are independently learned, namely Q_1 for determining the maximizing action and Q_2 for estimating the Q-value. Both Q_1 and Q_2 are updated randomly by $Q_1(s, a) \rightarrow r + \beta Q_2[s', argmax_a Q_1(s', a)]$ and $Q_2(s, a) \rightarrow r + \beta Q_1[s', argmax_a Q_2(s', a)]$.

Double DQN is based on the Double Q-learning model. Instead of finding the maximum of all Q-values when computing the target-Q value for the training step, the primary network is invoked for choosing an action, and the target network is also

Algorithm 2 Double Deep Q-network based trajectory design algorithm for autonomous driving

```
Input:
  Q-network structure, environment simulator, replay memory $D$, minibatch size $n$.
  Initialize the replay memory $D$, Q-network weights $\theta$, weights of the target network $\theta^* = \theta$, and $Q(s,a)$.
   The AV starts at random points.
  repeat
    for each step of episode:
      With the same process with DQN model, the AV carries out action $a_t$ and observes reward $r_t$. The model updates state $s_{t+1}$, stores transition $(s_t, a_t, r_t, s_{t+1})$ in $D$ and sample random minibatch of transitions $(s_i, a_i, r_i, {s'}_i)_{i \in n}$ from $D$;
        For each $i \in n$, compute the target
          $y_i = r_i + \gamma \cdot Q_{\theta^*}\left({s'}_i, \arg\max\left(Q({s'}_i, {a'}_i; \theta_i)\right)\right)$;
        Perform a gradient descent step for updating the Q-network
          $\theta \leftarrow \theta - a_t \cdot \frac{1}{n} \sum_{i \in n} [y_i - Q_\theta(s_i, a_i)] \cdot \nabla_\theta Q_\theta(s_i, a_i)$;
          $\theta \leftarrow \theta^*$.
    end
  until $s$ is terminal
Return: Action-value function $Q_\theta$ and policy $J$.
```

adopted for generating the target Q-value for that particular action. Hence, the equation of the Q-target can be written as

$$y_i = r_i + \gamma \cdot Q_{\theta^*}\left[{s'}_i, \arg\max\left(Q({s'}_i, {a'}_i; \theta_i)\right)\right]. \tag{21}$$

Two CNN models of the same architecture are established: the Q estimation network and the Q target network. A target network is used for generating the target-Q values that will be used to compute the loss for every action during training. Although not fully decoupled, the target network in the DQN architecture provides a natural candidate for the second value function, without having to introduce additional networks.

D. Comparison of the Proposed Fuel-Economy-Based Driving Policy to Other Practical Driving Policies

In order to illustrate the advantages of the proposed driving policy derived from our DDQN algorithm, three practical driving policies are proposed as benchmarks, namely, the fast-lane-based (FL) driving policy, smooth-trajectory-based (SMT) and straight-driving-based (ST) policy.

1) Fast-Lane-Based Driving Policy: As illustrated in Fig.1, it is assumed that each direction has three traffic lanes. According to the traffic rule, the lower bound constraint of the instantaneous velocity in the different lanes varies. More particularly, the velocity constraint for the fast-lane is no lower than $100km/h$, while that of the carriage lanes is no lower than $80km/h$ and $60km/h$, respectively.

In terms of the FL driving policy, it is assumed that the AV starts from the fast-lane. When an obstacle (such as a manned vehicle of lower velocity) appears in front, the AV carries out the action of changing lane for overtaking while maintaining safe driving. After overtaking, the AV changes lane back to the fast-lane to travel at a higher velocity.

According to this driving policy, the AV is mainly driving in the fast-lane, and the mission completion time will be reduced due to the high velocity. However, according to the fuel consumption model, fuel consumption is not linearly proportional to the time and velocity. In this case, the proposed FL driving policy is not optimal in terms of fuel economy. However, it is capable of striking a tradeoff between fuel economy and mission completion time.

2) Smooth-Trajectory-Based Driving Policy: As for the SMT driving policy, changing lanes on the highway is one of the highest risk maneuvers that a vehicle has to perform, because it involves changes in both the longitudinal and lateral velocity as well as movement in the presence of other moving vehicles. Based on the proposed DDQN algorithm, we improve the reward function design to fulfill the aforementioned tasks, while reducing the number of lane-change maneuvers.

In the DDQN model, the specific reward that the agent acquires is known as an immediate reward/penalty at a particular state-action pair. At each episode, the AV carries out an action and in return receives a reward/penalty. In the reward/penalty function design of the SMT driving policy, when the action carried out by the AV is lane change (either left and right), the reward derived from this action is lower than that derived from changing velocity. In this case, the reward function is redefined as (22) at the top of the next page.

It can be observed from this redefined reward function that, the SMT driving policy derived from DDQN algorithm contains fewer lane changes than the FE driving policy. This is because the DDQN model aims at maximizing the total reward of each episode, while the weight of the velocity-control values is higher than that of lane changes in the reward function. Overall, the SMT driving policy strikes a tradeoff between fuel economy and passenger comfort.

3) Straight-Trajectory-Based Driving Policy: In this driving policy, the vehicle is driven from the initial position to the destination in a straight line, while varying the velocity. Since no lane changes will be carried out by the AV in this driving policy, the cardinality of the action space is reduced from 5 to 3. At each timeslot, the AV has to choose an action from acceleration, deceleration and no extra action. In this case, the reward function is designed as

$$r_t^{\mathrm{ST}} = \begin{cases} c_1 |v(t) - v_0(t)|^2, & \text{collision}, \\ c_2 \left[100 - v(t)\right], & \text{no collision} v(t) \leqslant 100km/h, \\ -\int_t^{t+1} e(t) dt, & \text{no collision,} v(t) \geqslant 100km/h, \\ & E(t) \geqslant E(t+1), \\ c_3, & \text{otherwise}. \end{cases} \tag{23}$$

Since lane changes are forbidden in this driving policy, the AV has to change the instantaneous velocity for avoiding collision and for keeping a safe distance from other manned vehicles. It can be observed from the fuel consumption model that frequent acceleration or deceleration leads to increased fuel consumption. However, we have demonstrated in our previous work [49] that the computing complexity of the

$$r_t^{\mathrm{SMT}} = \begin{cases} c_1 |v(t) - v_0(t)|^2, & \text{collision}, \\ c_2 \left[100 - v(t)\right], & \text{no collision}, y(t) = -1/2w, v(t) \leq 100km/h, \\ c_2 \left[80 - v(t)\right], & \text{no collision}, y(t) = -3/2w, v(t) \leq 80km/h, \\ c_2 \left[60 - v(t)\right], & \text{no collision}, y(t) = -5/2w, v(t) \leq 60km/h, \\ -2\int_t^{t+1} e(t)dt, & a_t \rightarrow \text{lane change, no collision}, y(t) = -1/2w, v(t) \geq 100km/h, E(t) \geq E(t+1), \\ -2\int_t^{t+1} e(t)dt, & a_t \rightarrow \text{lane change, no collision}, y(t) = -3/2w, v(t) \geq 80km/h, E(t) \geq E(t+1), \\ -2\int_t^{t+1} e(t)dt, & a_t \rightarrow \text{lane change, no collision}, y(t) = -5/2w, v(t) \geq 60km/h, E(t) \geq E(t+1), \\ -\int_t^{t+1} e(t)dt, & a_t \neq \text{lane change, no collision}, y(t) = -1/2w, v(t) \geq 100km/h, E(t) \geq E(t+1), \\ -\int_t^{t+1} e(t)dt, & a_t \neq \text{lane change, no collision}, y(t) = -3/2w, v(t) \geq 80km/h, E(t) \geq E(t+1), \\ -\int_t^{t+1} e(t)dt, & a_t \neq \text{lane change, no collision}, y(t) = -5/2w, v(t) \geq 60km/h, E(t) \geq E(t+1), \\ c_3, & \text{otherwise}. \end{cases} \tag{22}$$

algorithm increases polynomially with the number of action options. Since the action space in this driving policy is reduced, the computational complexity is also reduced. Overall, the ST driving policy is capable of striking a tradeoff between fuel economy and computational complexity.

E. Analysis of the Proposed Algorithm

1) Complexity Analysis of the Proposed Algorithm: The complexity of the Algorithm 1 is commensurate with that of the CNN model. As demonstrated in [50], the total complexity of the CNN model is on the order of $O\left(\sum\limits_{i=1}^{2} f_{i-1} f_i n_i^2 m_i\right)$, where f_0 is the number of input channels for the CNN model, and m_i is the spatial size of the output feature map of the Conv layer i. The first Conv layer has a single input channel and f_1 filters. Each filter has a size $(n_2 \times n_2)$, and outputs a $f_1(n_1 - n_2 + 1)^2$ feature map. Similarly, the second Conv layer has f_1 filters each of size $(n_3 \times n_3)$. Each filter in the second Conv inputs a $(n_2 \times n_2)$ element matrix and outputs a $f_2(n_1 - n_2 - n_3 + 2)^2$ feature map. Thus, the CNN's computational complexity is $O\left[f_1\left(n_2^2(n_1 - n_2 + 1)^2 + f_2 n_3^2 (n_1 - n_2 - n_3 + 2)^2\right)\right]$.

Table I illustrates the performance of the proposed algorithms. The DDQN algorithm is capable of obtaining optimal results at the stable complexity as the DQN algorithm, which is far lower than that of the Q-learning algorithm. In terms of the Q-learning model's complexity, we have demonstrated in our previous work [49] that the size of the model is increased linearly with the number of states and polynomially with the number of actions.

2) Convergence Analysis of the Proposed Algorithm: Two steps are taken for analyzing the convergence of the DDQN algorithm. The first step is proving that the general Q-learning approach is indeed capable of converging to the optimal state. The second step is showing that the neural network approach succeeds in identifying the nonlinear Q-values generated by the general Q-learning iteration in equation (16).

Proposition 1. *The double Q-learning approach is capable of converging to the optimal Q value.*

As discussed in [49], the conventional Q-learning model converges to the optimal Q-function as long as $0 \leq \alpha_t \leq 1$, $\sum\limits_t \alpha_t = \infty$ and $\sum\limits_t \alpha_t^2 < \infty$. Apart from the conditions required for the convergence of the conventional Q-learning model, the extension from the conventional Q-learning model to the double Q-learning model requires another condition: the Q-values have to be stored in a lookup table. Assuming that

$$F_t\left(s_t, a_t\right) = F_t^Q\left(s_t, a_t\right) + \gamma\left(Q_t^2\left(s_{t+1}, a^*\right) - Q_t^1\left(s_{t+1}, a^*\right)\right), \tag{24}$$

where $F_t^Q\left(s_t, a_t\right) = r_t + \gamma Q_t^1\left(s_{t+1}, a^*\right) - Q_t^*\left(s_t, a_t\right)$ under the condition that the conventional Q-learning model converges to an optimal state. Upon writing $c_t = \gamma Q_t^2\left(s_{t+1}, a^*\right) - \gamma Q_t^1\left(s_{t+1}, a^*\right)$ we can readily find that $\Delta_t^{2,1} = Q_t^2 - Q_t^1$ converges to zero according to Lemma 2 in [51].

The update of $\Delta_t^{2,1}$ at timeslot t then obeys either $\Delta_{t+1}^{2,1}\left(s_t, a_t\right) = \Delta_t^{2,1}\left(s_t, a_t\right) + \alpha_t\left(s_t, a_t\right) F_t^2\left(s_t, a_t\right)$ or $\Delta_{t+1}^{2,1}\left(s_t, a_t\right) = \Delta_t^{2,1}\left(s_t, a_t\right) - \alpha_t\left(s_t, a_t\right) F_t^1\left(s_t, a_t\right)$, where $F_t^1\left(s_t, a_t\right) = r_t + \gamma Q_t^2\left(s_{t+1}, a^*\right) - Q_t^1\left(s_t, a_t\right)$ and $F_t^2\left(s_t, a_t\right) = r_t + \gamma Q_t^1\left(s_{t+1}, b^*\right) - Q_t^2\left(s_t, a_t\right)$. We define $\zeta_t^{2,1} = \frac{1}{2} a_t$, thus, we have

$$\begin{aligned} & E\left\{\Delta_{t+1}^{2,1}\left(s_t, a_t\right) | P_t\right\} = \Delta_t^{2,1}\left(s_t, a_t\right) \\ & + E\left\{a_t\left(s_t, a_t\right) F_t^2\left(s_t, a_t\right) - a_t\left(s_t, a_t\right) F_t^1\left(s_t, a_t\right) | P_t\right\} \\ & = \left(1 - \zeta_t^{2,1}\left(s_t, a_t\right)\right) \Delta_t^{2,1}\left(s_t, a_t\right) \\ & + \zeta_t^{2,1}\left(s_t, a_t\right) E\left\{F_t^{2,1}\left(s_t, a_t\right) | P_t\right\}, \end{aligned} \tag{25}$$

where $E\left\{F_t^{2,1}\left(s_t, a_t\right) | P_t\right\} = \gamma E\left\{Q_t^1\left(s_{t+1}, b^*\right) - Q_t^2\left(s_{t+1}, a^*\right) | P_t\right\}$.

Assuming that $E\left\{Q_t^1\left(s_{t+1}, b^*\right) | P_t\right\} \geq E\left\{Q_t^2\left(s_{t+1}, a^*\right) | P_t\right\}$, we have $Q_t^1\left(s_{t+1}, a^*\right) = \max_a Q_t^1\left(s_{t+1}, a_t\right) \geq Q_t^1\left(s_{t+1}, b^*\right)$, thus,

$$\begin{aligned} & \left|E\left\{F_t^{2,1}\left(s_t, a_t\right) | P_t\right\}\right| \\ & = \gamma E\left\{Q_t^1\left(s_{t+1}, b^*\right) - Q_t^2\left(s_{t+1}, a^*\right) | P_t\right\} \\ & \leq \gamma\left\|\Delta_t^{2,1}\left(s_t, a_t\right)\right\|. \end{aligned} \tag{26}$$

TABLE I: Performance of proposed algorithms

Algorithm	Q-learning algorithm	Deep Q-network algorithm	Double deep Q-network algorithm
Complexity	High	Moderate	Moderate
Optimality	Suboptimal	Suboptimal	Suboptimal
Convergency	No convergence	Convergence	Convergence
Stability	Not Stable	Not Stable	Stable

Assuming that $E\left\{Q_t^2\left(s_{t+1},a^*\right)|P_t\right\} \geq E\left\{Q_t^1\left(s_{t+1},b^*\right)|P_t\right\}$, in the same way, we can achieve at $\left|E\left\{F_t^{2,1}\left(s_t,a_t\right)|P_t\right\}\right| \leq \gamma\left\|\Delta_t^{2,1}\left(s_t,a_t\right)\right\|$.

It is worth noting that one of the two assumptions must be satisfied at each timeslot. In this case, we can always obtain that $\left|E\left\{F_t^{2,1}\left(s_t,a_t\right)|P_t\right\}\right| \leq \gamma\left\|\Delta_t^{2,1}\left(s_t,a_t\right)\right\|$. According to Lemma 2 in [51], the double Q-learning model is capable of converging to an optimal state without the aid of the CNN model. As our next step, we will analyze the convergence of the CNN-aided double Q-learning model.

Proposition 2. *Since the neural network is large enough and the initial conditions are appropriately chosen, it is capable of approximating any non-linear continuous function.*

Let us assume that $\varphi(t)$ is a non-constant, bounded and monotonically increasing continuous function, and I_m denotes the m-dimensional unit hypercube $[0,1]^m$. The space of continuous functions on I_m is denoted by $C(I_m)$. Then, given any function $f \in C(I_m)$ and $\varepsilon > 0$, there exists an integer N, real constants $\nu_i, b_i \in \mathbb{R}$ and real vectors $\omega_i \in \mathbb{R}^m$, where $i = 1, \cdots, N$, so that we may define $F(x) = \sum\limits_{i=1}^{N} \nu_i \varphi\left(\omega_i^T x + b_i\right)$ as an approximate realization of the function f, where f is independent of φ; that is $|F(x) - f(x)| < \varepsilon$ or all $x \in I_m$. In other words, the functions of the form $F(x)$ are dense in $C(I_m)$. By following the Stone-Weierstrass Theorem [52], the proof of Proposition 2 has been given in [53] and formulated as the Universal Approximation Theorem. It has been demonstrated that if the neural network itself is large enough and the initial conditions are properly chosen, it can approximate any non-linear continuous function.

It can be observed from **Proposition 1** and **Proposition 2** that since the neural network is large enough and the initial conditions are appropriately chosen, it is capable of approximating any non-linear continuous function. In this case, the double deep Q-network is capable of converging.

TABLE II: Simulation parameters

Parameter	Description	Value
f_c	Carrier frequency	2GHz
N_0	Noise power spectral	-170dBm/Hz
m_v	Car mass	1000kg
η_T	Efficiency of the transmission	0.95
η_e	Common peak efficiencies	0.3
P_P	Engine rated power	100kW
ξ_{ax}	The final drive gear ratio	3.5
ζ_n	The gearbox gear ratio	1.0
r_d	Rolling radius of the tire	0.3m
A_f	The characteristic area of the car	$1.8m^2$
γ_m	The mass factor of the car	1.08

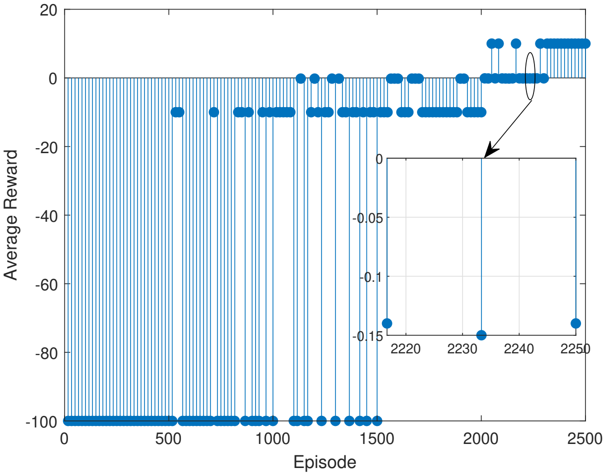


Fig. 4: Average reward of the DQN model in each episode.

IV. Simulation Results

In this section, we verify the efficiency of the proposed DDQN based trajectory design algorithm. In the simulations, the highway is assumed to have parallel lanes with the length of 2600m. Each direction has 3 traffic lanes with a width of 3.75m, which is the standard width in the highway. Two categories of vehicles are employed to represent manned vehicles: cars and trucks. The cars have a length of 4.7m, a width of 1.8m and a height of 1.4m, while the trucks have a length of 8.2m, a width of 2.5m and a height of 3.5m. Along this 2600m highway section, 60 manned vehicles having particular coordinates and velocities are assumed. The manned vehicles on the fast-lane are with a constant velocity of $100km/h$, while that of the manned vehicles on carriage lanes is $80km/h$. It is worth noting that all the manned vehicles are not allowed to change lane. The conception of a more practical model will be left for our future research.

In the proposed DDQN model, the state space consists of the current coordinates of both the AV and of other manned vehicles, as well as the current fuel consumption of the AV. Thus, the total number of the states in the state space at each timeslot is 63. Additionally, as mentioned above, the total number of actions in the DQN model is 8.

Fig. 4 characterizes the average reward of each episode. It can be observed that the average reward of the DDQN model increases with the number of training episodes, hence the AV becomes capable of carrying out the right actions after training for about 2000 episodes. The average reward increases from a negative scale to a positive scale after the DDQN model has learned the strategy of maximizing the rewards and avoiding penalties. This phenomenon is also confirmed by the insights

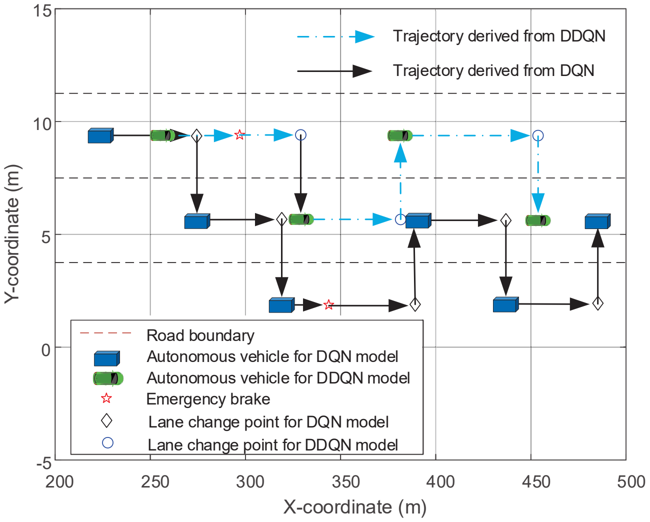

Trajectory derived from DDQN
Trajectory derived from DQN
Road boundary
Autonomous vehicle for DQN model
Autonomous vehicle for DDQN model
Emergency brake
Lane change point for DQN model
Lane change point for DDQN model
Y-coordinate (m)
X-coordinate (m)
15
10
5
0
-5
200
250
300
350
400
450
500

Fig. 5: Illustration of the AV's trajectory and emergency braking.

provided in **Remark 3**.

Fig. 5 characterizes the trajectory of the AV as well as the emergency braking actions (An emergency braking action is declared when the distance between the AV and the car in front is less than that given by the three-second rule, while the velocity gap is higher than 20km/h) derived from both the DQN and DDQN algorithm. Compared to practical driving policies, the FE driving policy derived from both the DQN and DDQN algorithms contains more lane changes for overtaking instead of deceleration. This is because of the acceleration and deceleration process results in higher fuel consumption according to our fuel consumption model. It can also be observed that compared to the trajectory derived from the DQN algorithm, invoking the DDQN algorithm results in a different trajectory for the AV.

Fig. 6 characterizes the average fuel consumption rate vs the number of training episodes. It can be observed that the average fuel consumption rate of the AV for each episode decreases gradually vs the episode indeed. This means that the proposed DDQN algorithm is capable of converging after training. This phenomenon is also confirmed by the insights in **Proposition 1** and **Proposition 2**. Additionally, the value of the learning rate determines the updating rate, thus affects the performance of the algorithm, which is also demonstrated by **Remark 5**. It can also be observed from Fig. 6 that the learning rate of 0.0080 for the DQN model outperforms that of 0.0070, 0.0075 and 0.0090 in both the convergence rate and fuel consumption.

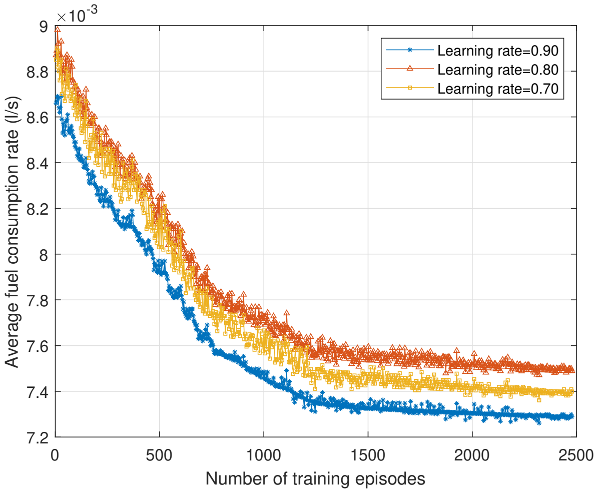


Fig. 6: Average fuel consumption vs the number of training episodes.

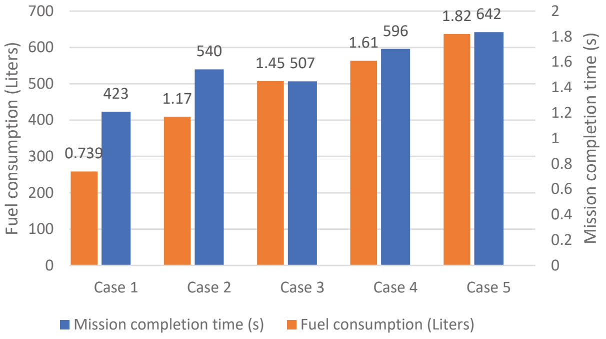


Fig. 7: Comparison of fuel consumption and mission completion time for different cases. (Case 1 is the baseline, Case 2 is the FE driving policy. Case 3 is the FL driving policy, Case 4 is the SMT driving policy, Case 5 is the ST driving policy.)

Fig. 7 characterizes the total fuel consumption and mission completion time of the FE driving policy and of the other three benchmarkers. It can be observed that obtaining a lower mission completion time tends not to be associated with decreased total fuel consumption. The reason is that carrying out the action of acceleration and deceleration consumes far more fuel than driving at a constant speed. Frequently accelerating and decelerating results in a lower mission completion time, while leads to a higher fuel dissipation. It can also be observed that the proposed FE driving policy (case 2) derived from the DDQN algorithm consumes 24% less fuel than benchmarks (case 3-5). However, the mission completion time of this scheme is higher than that of the FL driving policy (case 3). The FL driving policy (case 3) consumes more fuel, but the mission completion time is shortened. It strikes a tradeoff between the fuel consumption and mission completion time. In the ST driving policy (case 4), the AV consumes the most fuel, while simultaneously requiring the most time. However, changing lane is avoided in this driving policy, hence providing a more comfortable and safe driving experience.

Fig. 8 characterizes the fuel consumption of the AV vs the different traffic conditions (density of manned vehicles). It can be observed that the fuel consumption of the AV rises sharply, as the number of manned vehicles increases. The reason is that, as the density of manned vehicles increases, the AV has to carry out more actions (lane-change, acceleration and deceleration) for avoiding the collision. Additionally, the

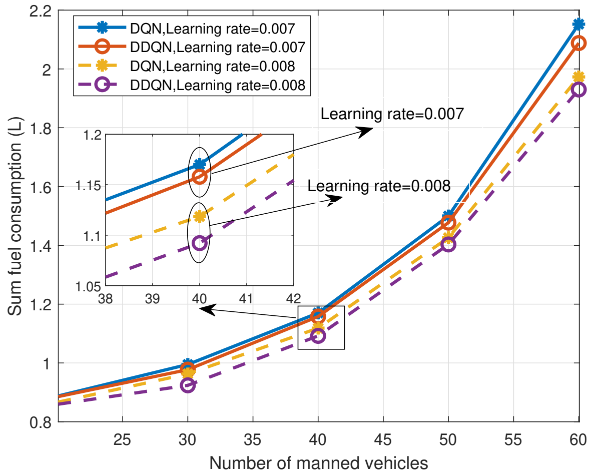


Fig. 8: Total fuel consumption vs the traffic conditions (Traffic condition refers to the density of vehicles on this particular 2600m highway section).

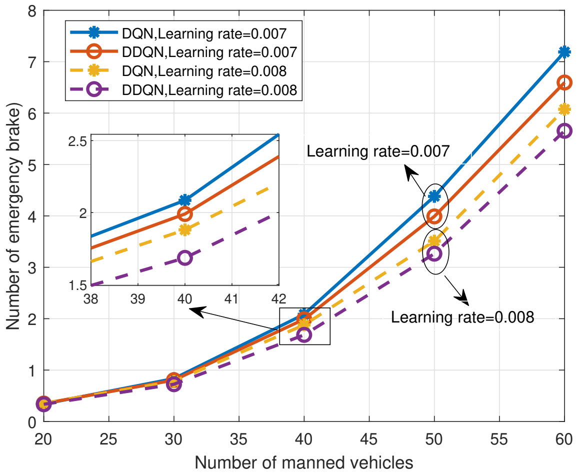


Fig. 9: Number of emergency braking event vs traffic conditions (Traffic condition refers to the density of vehicles on this particular 2600m highway section).

proposed DDQN algorithm outperforms the DQN algorithm in terms of the fuel consumption. As it has been demonstrated in **Table I**, the proposed DDQN algorithm is capable of achieving more stable results compared to the DQN algorithm.

Fig. 9 characterizes the number of emergency braking event for different traffic conditions. It can be observed that when the density of traffic increases, the AV is more likely to break. The trajectory derived from the DDQN algorithm suffers from less emergency braking event than that derived from the DQN algorithm. Additionally, for the trajectory derived from both the DDQN algorithm and the DQN algorithm, the learning rate of 0.008 results in less emergency braking event than that of 0.007.

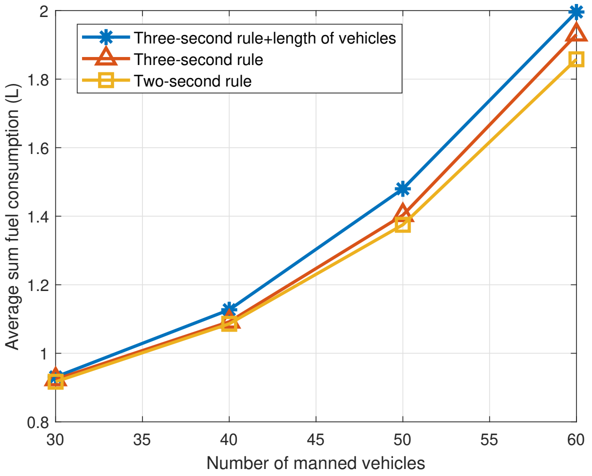


Fig. 10: Total fuel consumption vs the number of manned vehicles parameterized by three different driving distances.

Fig. 10 characterizes the fuel consumption of the AV vs the number of manned vehicles parameterized by three different safe driving distance scenarios under the proposed DDQN-based algorithm, when the learning rate is 0.008. It can be observed that when the value of safe driving distance increases, the AV consumes more fuel. The reason for this trend is that when the distance between the AV and other manned vehicles in the same lane is lower than the safe driving distance, the AV has to carry out preventive actions (slow down or change lane). Thus, as the value of safe driving distance increases, the AV is more likely to accelerate or decelerate, which results in increased fuel consumption.

V. Conclusions

The joint design of paths, motion and instantaneous velocity of the AV on the fuel economy of the AV was considered while enhancing driving safety. To tackle the problem formulated, a deep Q-network based algorithm was proposed for determining the position and motion of the AVs. By receiving real-time traffic information from ground base stations, the AV (acting as an agent in the DQN model) was shown to find a policy guaranteeing both fuel economy and safe driving. It was also demonstrated that the proposed DDQN algorithm was capable of converging after appropriate training. It outperformed the DQN algorithm by overcoming the large overestimation of action values caused by the Q-learning model. Additionally, the proposed fuel-economy based driving policy derived from the DDQN algorithm consumed less fuel than benchmark driving policies.

References

[1] L. Hobert, A. Festag, I. Llatser, L. Altomare, and F. Visintainer, “Enhancements of V2X communication in support of cooperative autonomous driving,” *IEEE Commun. Mag.*, vol. 53, no. 12, pp. 64–70, 2015.

[2] R. P. D. Vivacqua, M. Bertozzi, P. Cerri, F. N. Martins, and R. F. Vassallo, “Self-localization based on visual lane marking maps: An accurate low-cost approach for autonomous driving,” *IEEE Trans. Intell. Transport. Syst*, vol. 19, no. 2, pp. 582–597, 2018.

[3] G. Williams, P. Drews, B. Goldfain, J. M. Rehg, and E. A. Theodorou, “Information-theoretic model predictive control: Theory and applications to autonomous driving,” *IEEE Trans. Robot*, vol. 34, no. 6, pp. 1603–1622, 2018.

[4] R. Michelmore, M. Kwiatkowska, and Y. Gal, "Evaluating uncertainty quantification in end-to-end autonomous driving control," *arXiv:1811.06817*, 2018.

[5] J. Wang, J. Liu, and N. Kato, "Networking and communications in autonomous driving: A survey," *IEEE Commun. Surveys Tutorials*, vol. 21, no. 2, pp. 1243–1274, 2018.

[6] Y. S. Son, W. Kim, S.-H. Lee, and C. C. Chung, "Robust multirate control scheme with predictive virtual lanes for lane-keeping system of autonomous highway driving," *IEEE Trans. Veh. Technol.*, vol. 64, no. 8, pp. 3378–3391, 2014.

[7] L. Yao, J. Wang, X. Wang, A. Chen, and Y. Wang, "V2X routing in a VANET based on the hidden Markov model," *IEEE Trans. Intell. Transport. Syst*, vol. 19, no. 3, pp. 889–899, 2018.

[8] J. Ji, A. Khajepour, W. W. Melek, and Y. Huang, "Path planning and tracking for vehicle collision avoidance based on model predictive control with multiconstraints," *IEEE Trans. Veh. Technol.*, vol. 66, no. 2, pp. 952–964, Feb. 2017.

[9] C. Chen, A. Seff, A. Kornhauser, and J. Xiao, "Deep driving: Learning affordance for direct perception in autonomous driving," in *2015 IEEE International Conference on Computer Vision (ICCV)*, Dec. 2015, pp. 2722–2730.

[10] M. Alam, J. Ferreira, and J. Fonseca, "Introduction to intelligent transportation systems," in *Intelligent Transportation Systems*. Springer, 2016, pp. 1–17.

[11] M. Lu, O. Turetken, E. Mitsakis, R. Blokpoel, R. Gilsing, P. Grefen, and A. Kotsi, "Cooperative and connected intelligent transport systems for sustainable european road transport," in *the 7th Transport Research Arena*, 2018, pp. 1–10.

[12] Y. Zhang, Q. Lin, J. Wang, S. Verwer, and J. M. Dolan, "Lane-change intention estimation for car-following control in autonomous driving," *IEEE Trans. Intell. Veh*, vol. 3, no. 3, pp. 276–286, 2018.

[13] L. Li, K. Ota, and M. Dong, "Humanlike driving: empirical decision-making system for autonomous vehicles," *IEEE Trans. Veh. Technol.*, vol. 67, no. 8, pp. 6814–6823, 2018.

[14] C. Jang, C. Kim, S. Lee, S. Kim, S. Lee, and M. Sunwoo, "Re-plannable automated parking system with a standalone around view monitor for narrow parking lots," *IEEE Trans. Intell. Transport. Syst*, pp. 1–14, 2019.

[15] T. S. Li, M. Lee, C. Lin, G. Liou, and W. Chen, "Design of autonomous and manual driving system for 4WIS4WID vehicle," *IEEE Access*, vol. 4, pp. 2256–2271, 2016.

[16] S. Lefvre, A. Carvalho, and F. Borrelli, "A learning-based framework for velocity control in autonomous driving," *IEEE Transactions on Automation Science and Engineering*, vol. 13, no. 1, pp. 32–42, 2016.

[17] Z. Su, Y. Hui, and T. H. Luan, "Distributed task allocation to enable collaborative autonomous driving with network softwarization," *IEEE J. Sel. Areas Commun.*, vol. 36, no. 10, pp. 2175–2189, 2018.

[18] D. González, J. Pérez, V. Milanés, and F. Nashashibi, "A review of motion planning techniques for automated vehicles," *IEEE Trans. Intell. Transport. Syst*, vol. 17, no. 4, pp. 1135–1145, 2015.

[19] S. Vaskov, U. Sharma, S. Kousik, M. Johnson-Roberson, and R. Vasudevan, "Guaranteed safe reachability-based trajectory design for a high-fidelity model of an autonomous passenger vehicle," in *2019 American Control Conference (ACC)*, 2019, pp. 705–710.

[20] L. Li and X. Li, "Parsimonious trajectory design of connected automated traffic," *Transportation Research Part B: Methodological*, vol. 119, pp. 1–21, 2019.

[21] X. Li, Z. Sun, D. Cao, Z. He, and Q. Zhu, "Real-time trajectory planning for autonomous urban driving: Framework, algorithms, and verifications," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 2, pp. 740–753, Apr. 2016.

[22] B. Paden, M. Čáp, S. Z. Yong, D. Yershov, and E. Frazzoli, "A survey of motion planning and control techniques for self-driving urban vehicles," *IEEE Transactions on intelligent vehicles*, vol. 1, no. 1, pp. 33–55, 2016.

[23] C. Zhai, F. Luo, Y. Liu, and Z. Chen, "Ecological cooperative look-ahead control for automated vehicles travelling on freeways with varying slopes," *IEEE Trans. Veh. Technol.*, vol. 68, no. 2, pp. 1208–1221, 2018.

[24] C. Zhai, F. Luo, and Y. Liu, "Cooperative look-ahead control of vehicle platoon for maximizing fuel efficiency under system constraints," *IEEE Access*, vol. 6, pp. 37 700–37 714, 2018.

[25] C. Zhai, Y. Liu, and F. Luo, "A switched control strategy of heterogeneous vehicle platoon for multiple objectives with state constraints," *IEEE Trans. Intell. Transport. Syst*, vol. 20, no. 5, pp. 1883–1896, May 2019.

[26] M. K. Pal, R. Bhati, A. Sharma, S. K. Kaul, S. Anand, and P. Sujit, "A reinforcement learning approach to jointly adapt vehicular communications and planning for optimized driving," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, 2018, pp. 3287–3293.

[27] S.-W. Kim, B. Qin, Z. J. Chong, X. Shen, W. Liu, M. H. Ang, E. Frazzoli, and D. Rus, "Multivehicle cooperative driving using cooperative perception: Design and experimental validation," *IEEE Trans. Intell. Transport. Syst*, vol. 16, no. 2, pp. 663–680, 2015.

[28] S.-W. Kim, Z. J. Chong, B. Qin, X. Shen, Z. Cheng, W. Liu, and M. H. Ang, "Cooperative perception for autonomous vehicle control on the road: Motivation and experimental results," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 5059–5066.

[29] N. Wang, X. Wang, P. Palacharla, and T. Ikeuchi, "Cooperative autonomous driving for traffic congestion avoidance through vehicle-to-vehicle communications," in *IEEE Vehicular Networking Conference (VNC)*, 2017, pp. 327–330.

[30] H. Khan, P. Luoto, S. Samarakoon, M. Bennis, and M. Latva-Aho, "Network slicing for vehicular communication," *arXiv preprint arXiv:1905.09578*, 2019.

[31] Y. Liu, S. Bi, Z. Shi, and L. Hanzo, "When machine learning meets big data: A wireless communication perspective," *arXiv preprint arXiv:1901.08329*, 2019.

[32] C. Jiang, H. Zhang, Y. Ren, Z. Han, K. Chen, and L. Hanzo, "Machine learning paradigms for next-generation wireless networks," *IEEE Wireless Commun.*, vol. 24, no. 2, pp. 98–105, Apr. 2017.

[33] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "Safe, multi-agent, reinforcement learning for autonomous driving," *CoRR*, 2016.

[34] S. Nageshrao, E. Tseng, and D. Filev, "Autonomous highway driving using deep reinforcement learning," *arXiv preprint arXiv:1904.00035*, 2019.

[35] Z. Qin, H. Ye, G. Y. Li, and B.-H. F. Juang, "Deep learning in physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 93–99, 2019.

[36] G. Hartman, Z. Shiller, and A. Azaria, "Deep reinforcement learning for time optimal velocity control using prior knowledge," *arXiv preprint arXiv:1811.11615*, 2018.

[37] Z. Qiao, K. Muelling, J. M. Dolan, P. Palanisamy, and P. Mudalige, "Automatically generated curriculum based reinforcement learning for autonomous vehicles in urban environment," in *IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 1233–1238.

[38] B. Tan, N. Xu, and B. Kong, "Autonomous driving in reality with reinforcement learning and image translation," *arXiv preprint arXiv:1801.05299*, 2018.

[39] P. Wang, C.-Y. Chan, and A. de La Fortelle, "A reinforcement learning based approach for automated lane change maneuvers," in *IEEE Intelligent Vehicles Symposium (IV)*, 2018, pp. 1379–1384.

[40] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[41] B. Sklar, "Rayleigh fading channels in mobile digital communication systems. I. characterization," *IEEE Commun. Mag.*, vol. 35, no. 7, pp. 90–100, 1997.

[42] B.-S. Chen, C.-L. Tsai, and C.-S. Hsu, "Robust adaptive MMSE/DFE multiuser detection in multipath fading channel with impulse noise," *IEEE Trans. Signal Process.*, vol. 53, no. 1, pp. 306–317, Jan. 2005.

[43] Y. Liu, Z. Qin, M. Elkashlan, Z. Ding, A. Nallanathan, and L. Hanzo, "Non-orthogonal multiple access for 5G and beyond," *Proc. IEEE*, vol. 105, no. 12, pp. 2347–2381, 2017.

[44] Y. Liu, Z. Qin, M. Elkashlan, Y. Gao, and L. Hanzo, "Enhancing the physical layer security of non-orthogonal multiple access in large-scale networks," *IEEE Trans. Wireless Commun.*, vol. 16, no. 3, pp. 1656–1672, 2017.

[45] M. Ben-Chaim, E. Shmerling, and A. Kuperman, "Analytic modeling of vehicle fuel consumption," *Energies*, vol. 6, no. 1, pp. 117–127, 2013.

[46] M. A. S. Kamal, M. Mukai, J. Murata, and T. Kawabe, "Ecological vehicle control on roads with up-down slopes," *IEEE Trans. Intell. Transport. Syst*, vol. 12, no. 3, pp. 783–794, 2011.

[47] L. Xiao, C. Xie, M. Min, and W. Zhuang, "User-centric view of unmanned aerial vehicle transmission against smart attacks," *IEEE Trans. Veh. Technol.*, vol. 67, no. 4, pp. 3420–3430, 2018.

[48] H. Van Hasselt, A. Guez, and D. Silver, "Deep reinforcement learning with double Q-learning," in *Thirtieth AAAI Conference on Artificial Intelligence*, 2016, pp. 1–8.

[49] X. Liu, Y. Liu, Y. Chen, and L. Hanzo, "Trajectory design and power control for multi-UAV assisted wireless networks: A machine learning approach," *IEEE Trans. Veh. Technol.*, vol. 68, no. 8, pp. 7957–7969, 2019.

[50] K. He and J. Sun, "Convolutional neural networks at constrained time cost," in *IEEE conference on computer vision and pattern recognition (CVPR)*, 2015, pp. 5353–5360.

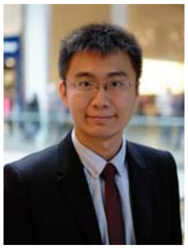
[51] H. V. Hasselt, "Double Q-learning," in *Advances in Neural Information Processing Systems*, 2010, pp. 2613–2621.

[52] V. Timofte, A. Timofte, and L. A. Khan, "Stone-Weierstrass and extension theorems in the nonlocally convex case," *Journal of Mathematical Analysis and Applications*, vol. 462, no. 2, pp. 1536–1554, 2018.

[53] A. Sannai, Y. Takai, and M. Cordonnier, "Universal approximations of permutation invariant/equivariant functions by deep neural networks," *arXiv preprint arXiv:1903.01939*, 2019.

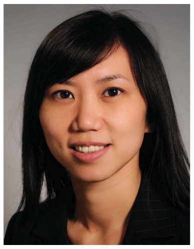
Xiao Liu (S'18) received the B.S. degree and M.S. degree in 2013 and 2016, respectively. He is currently working toward the Ph.D. degree with Communication Systems Research Group, School of Electronic Engineering and Computer Science, Queen Mary University of London.

His research interests include unmanned aerial vehicle (UAV) aided networks, machine learning, non-orthogonal multiple access (NOMA) techniques, reconfigurable intelligent surface (RIS).



Yuanwei Liu (S'13-M'16-SM'19) received the B.S. and M.S. degrees from the Beijing University of Posts and Telecommunications in 2011 and 2014, respectively, and the Ph.D. degree in electrical engineering from the Queen Mary University of London, U.K., in 2016. He was with the Department of Informatics, King's College London, from 2016 to 2017, where he was a Post-Doctoral Research Fellow. He has been a Lecturer (Assistant Professor) with the School of Electronic Engineering and Computer Science, Queen Mary University of London, since 2017.

His research interests include 5G and beyond wireless networks, Internet of Things, machine learning, and stochastic geometry. He received the Exemplary Reviewer Certificate of the IEEE wireless communication letters in 2015, the IEEE transactions on communications in 2016 and 2017, the IEEE transactions on wireless communications in 2017. Currently, He is in the editorial board of serving as an Editor of the IEEE Transactions on Communications, IEEE communication letters and the IEEE access. He also serves as a guest editor for IEEE JSTSP special issue on "Signal Processing Advances for Non-Orthogonal Multiple Access in Next Generation Wireless Networks ". He has served as the Publicity Co-Chairs for VTC2019-Fall. He has served as a TPC Member for many IEEE conferences, such as GLOBECOM and ICC.



Yue Chen (S'02-M'03-SM'15) is a Professor of Telecommunications Engineering at the School of Electronic Engineering and Computer Science, Queen Mary University of London (QMUL), U.K.. Prof Chen received the bachelor's and master's degree from Beijing University of Posts and Telecommunications (BUPT), Beijing, China, in 1997 and 2000, respectively. She received the Ph.D. degree from QMUL, London, U.K., in 2003.

Her current research interests include intelligent radio resource management (RRM) for wireless networks; cognitive and cooperative wireless networking; mobile edge computing; HetNets; smart energy systems; and Internet of Things.

Lajos Hanzo (M'91-SM'92-F'04) (http:www-mobile.ecs.soton.ac.uk, https:en.wikipedia.org/wiki/Lajos_Hanzo) FREng, FIEEE, FIET, Fellow of EURASIP, DSc holds an honorary doctorate by the Technical University of Budapest (2009) and by the University of Edinburgh (2015). He is a Foreign Member of the Hungarian Academy of Sciences and a former Editor-in-Chief of the IEEE Press. He has served as Governor of both IEEE ComSoc and of VTS. He has published 1900+ contributions at IEEE Xplore, 19 Wiley-IEEE Press books and has helped the fast-track career of 119 PhD students. Over 40 of them are Professors at various stages of their careers in academia and many of them are leading scientists in the wireless industry.