Event-Triggered Action-Delayed Reinforcement Learning Control of a Mixed Autonomy Signalised Urban Intersection

Erica Salvato, Arnob Ghosh, Gianfranco Fenu and Thomas Parisini

***Abstract*— We propose an event-triggered framework for deciding the traffic light at each lane in a mixed autonomy scenario. We deploy the decision after a suitable delay, and events are triggered based on the satisfaction of a predefined set of conditions. We design the trigger conditions and the delay to increase the vehicles' throughput. This way, we achieve full exploitation of autonomous vehicles (AVs) potential. The ultimate goal is to obtain vehicle-flows led by AVs at the head. We formulate the decision process of the traffic intersection controller as a deterministic delayed Markov decision process, i.e., the action implementation and evaluation are delayed. We propose a Reinforcement Learning based model-free algorithm to obtain the optimal policy. We show - by simulations - that our algorithm converges, and significantly reduces the average wait-time and the queues length as the fraction of the AVs increases. Our algorithm outperforms our previous work [1] by a quite significant amount.**

I. Introduction

With the introduction of autonomous vehicles (AVs), traffic intersection control technologies can be completely revamped. In an AVs-only scenario, the controller can properly manage traffic, thus avoiding queues and collisions between vehicles, without the use of traffic lights [2]. However, this perspective is still quite far away, and, in the near future, AVs and human-driven vehicles (HDVs) are expected to coexist, resulting in the so-called *mixed autonomy* scenario. HDVs are not directly controllable through the traffic intersection controller, meaning that traffic lights still need to be used for a proper traffic management. The primary research question in a mixed-autonomy setting becomes: *Can a traffic intersection controller be designed to minimize congestion by taking advantage of AVs, when HDVs are also present?*

Reinforcement Learning (RL), with its data-driven ability to learn and adapt controllers, can be a useful solution for this purpose.

E. Salvato is with the Dept. of Engineering and Architecture at the University of Trieste erica.salvato@dia.units.it

A. Ghosh is with the Dept. of Electrical and Computer Engineering at the Ohio State University, Columbus, USA. He was with the Dept. of Electrical and Electronic Engineering at the Imperial College of London, UK, when a part of the work was completed. ghosh.244@osu.edu

G. Fenu is with the Dept. of Engineering and Architecture at the University of Trieste fenu@units.it

T. Parisini is with the Department of Electrical and Electronic Engineering, Imperial College London, London SW7 2AZ, UK, with the Department of Engineering and Architecture, University of Trieste, 34127 Trieste, Italy, and also with the KIOS Research and Innovation Center of Excellence, University of Cyprus, CY-1678 Nicosia, Cyprus t.parisini@imperial.ac.uk

This work has been partially supported by European Union's Horizon 2020 research and innovation program under grant agreement no. 739551 (KIOS CoE) and by the Italian Ministry for Research in the framework of the 2017 Program for Research Projects of National Interest (PRIN), Grant no. 2017YKXYXJ.

Dynamic programming-based, and optimization-based algorithms are for example proposed in [3]–[7] to control the traffic-light duration at urban intersection. In [8]–[11], RL-based approaches have instead been used to address the same problem. However, all the above works focus in AVs-only scenarios, and do not care about the traffic management problem in a mixed autonomy context. In [12], [13] decentralized RL-based approaches are designed to control traffic at intersections in presence of mixed autonomy.

In this paper, we consider an event-driven centralized traffic-light controller, in which events occur when some conditions (Section II-B) are met. The traffic controller decides the traffic lights across the lanes based on the number of vehicles when an event occurs. The time interval between two consecutive controller decisions is not of fixed duration, and depends on the distance, in time, between two consecutive events. The decision is implemented after a certain delay of d_a seconds. The d_a value changes depending on the triggering condition of the event. In particular, the trigger conditions are designed to bring AVs in leading each queue in each lane. In such a condition, the intersection controller can communicate to each AV leader the future instance when the respective traffic light will be green again. Each AV can adjust its dynamics by solving an optimal control problem to minimize the fuel cost while entering the intersection at the maximum speed. The dynamics of HDVs is modeled, as in [1], following the Intelligent Driver Model (IDM) [14].

The decision process of the traffic intersection controller is modeled as a deterministic delayed Markov Decision Process (DDMDP) due to the presence of the action delay (d_a). Determining an optimal decision is computationally challenging. The decision affects the dynamics of the vehicles in a non-linear and non-smooth manner. Further, the dynamics of the vehicles vary based on whether they are AVs or HDVs.

We use a Reinforcement-Learning (RL) based *model-free* algorithm to learn the optimal policy for the traffic intersection controller. Our approach stabilizes the cumulative reward in a quite reasonable time, thereby obtaining a reduction in queues and waiting time for vehicles. *Empirical analysis suggests that the average wait-time and the queues length decrease as the fraction of the AVs increases.* Experimental results also show that the proposed solution outperforms our recent work [1] on traffic-light control for mixed autonomy, where, conversely, the traffic-light cycle is imposed of fixed length.

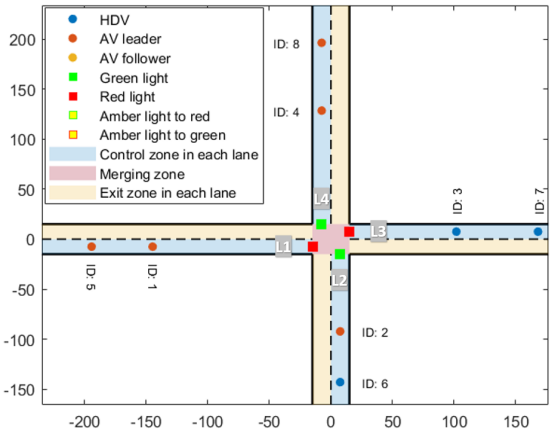

Fig. 1. Traffic-light controlled intersection in a mixed autonomy scenario

II. System Model

In the following, we, first, describe the urban intersection system which we consider (Section II-A). Subsequently, we illustrate the event-driven intersection controller (Section II-B), the dynamics of AVs, and how the decision of the intersection controller impacts them (Section II-C).

A. The Urban Intersection system

We consider a signalized urban intersection consisting of 4 lanes (Figure 1). We partition it in three main parts: 1) a *Merging Zone* (MZ) of size $L_M \times L_M$, delimiting the area where vehicles of different lanes converge; 2) a *Control Zone* (CZ) of length L_C for each lane, where vehicles travel before entering the MZ; 3) an *Exiting Zone* (EZ) of length L_E for each lane, where vehicles travel after crossing the MZ. A vehicle is considered to exit the intersection when it covers a distance of $L_C + L_M + L_E$.

A traffic-light is placed at the junction between the CZ and the MZ of each lane (4 traffic-lights in total). Each vehicle enters the MZ when the respective traffic-light is green. The vehicle stops to enter the MZ when the traffic-light is red. Once the vehicles enter the MZ they cross the intersection and do not stop.

We now introduce some notations which we use throughout this paper. We denote by $c(i,j)$ the i-th vehicle at lane j, where $i \in \{1,2,\dots,N_{\text{max,j}}\}$, and $N_{\text{max,j}}$ is maximum number of vehicles at the j-th lane, and $j \in \{1,2,\dots,4\}$ (cf. Figure 1).

We denote by $C^{(t_k)}$ the set of N vehicles in all the lanes of the intersection system at $t = t_k$. The HDVs' set and the AVs' set, respectively are $C_H^{(t_k)}$ and $C_A^{(t_k)}$, and $C_A^{(t_k)}, C_H^{(t_k)} \subseteq C^{(t_k)}$ s.t. $C_A^{(t_k)} \cap C_H^{(t_k)} = \emptyset$, $C_A^{(t_k)} \cup C_H^{(t_k)} = C^{(t_k)}$. $C_j^{(t_k)}$ denotes the set vehicles in the j-th lane of the intersection system at $t = t_k$.

We denote with $p_{i,j}(t_k)$, $v_{i,j}(t_k)$, and $u_{i,j}(t_k)$ respectively the position, the speed, and the acceleration of the $c(i,j)$-th vehicle in the intersection at $t = t_k$. Each i-th vehicle entering the control zone of the j-th lane at $t = t_{i,j}^0$ will be initialized with an initial position $p_{i,j}(t_{i,j}^0) = 0$. A new vehicle $c(i',j)$ entering the CZ of the j-th lane right after $c(i,j)$ will have $i' = i+1$, i.e., the more recent the vehicle access to the intersection, the higher the index i associated with it will be.

Definition 1. $c(i,j)$ is behind $c(k,j)$ if $k < i$. When $k = i-1$, $c(k,j)$ is the *front* vehicle of $c(i,j)$, i.e., the immediately preceding vehicle of $c(i,j)$.

Definition 2. A vehicle $c(i,j) \in C_A$ in the CZ of the j-th lane ($p_{i,j}(t) < L_C$) is the *leader* vehicle of the j-th lane if $\nexists\, c(k,j) \in C$ front vehicle of $c(i,j)$ (cf. Definition 1) such that $p_{k,j}(t) < L_C$.

Definition 3. A vehicle $c(i,j) \in C_A$ in the CZ of the j-th lane ($p_{i,j}(t) < L_C$) is a *follower* vehicle of $c(k,j)$ if $p_{k,j}(t) < L_C$ and $k < i$.

Definition 4. A pair of lanes (j,k) are non-conflicting if there are no intersection points that can lead to vehicles crashes. We denote by $\mathcal{L}$ the set of non-conflicting lane pairs.

In the proposed scenario (cf. Figure 1) the set of non-conflicting lane pairs is $\mathcal{L} = \{(1,3),(2,4)\}$. A traffic-controller can only simultaneously assign green traffic-lights for non-conflicting pair of lanes.

We assume the followings:

Assumption 1. *A vehicle $c(i,j) \in C$ can only go forward or stay still; i.e., no turning, backward gears, or lane changing are allowed.*

Assumption 2. *A vehicle $c(i,j) \in C_A$ is considered sensors equipped. $c(i,j)$ is able to estimate $p_{i-1,j}(t)$ and $v_{i-1,j}(t)$ if $c(i-1,j) \in C_H$, while can access the actual values of $p_{i-1,j}(t)$ and $v_{i-1,j}(t)$ if $c(i-1,j) \in C_A$.*

The first assumption can be relaxed by considering more complicated dynamics. The second assumption entails that an AV can adapt to the motion of the preceding vehicle.

We denote by v_{free} the maximum allowable speed within the intersection system. AVs are assumed to travel with a constant speed v_{free} once they cross the MZ.

We interchangeably also use the notation (i,j) for $c(i,j)$. We denote by $t_{i,j}^m$ the time at which the vehicle (i,j) enters the MZ (i.e., leaves the CZ). Formally,

Definition 5. $t_{i,j}^m = \inf\{t | p_{i,j}(t) > L_C\}$.

B. Event-driven Traffic-Intersection controller

We assume that the traffic-intersection controller constantly observes and receives information about the current traffic condition in the intersection, and triggers the decision-making process of a RL-based traffic-light controller when certain conditions are met:

(C1) at time t_k the CZ of the j-th lane is empty ($C_j^{(t_k)} = \emptyset$) and the traffic-light status is green at the j-th lane;

(C2) at time t_k a vehicle $c(i,j) \in C_A$ enters the intersection and the traffic-light status is green at lane j;

(C3) a trigger did not occur for a T_{silence} time interval.

With condition (C1) we are enabling the traffic-light controller to possibly close empty lanes. Condition (C3), instead,

is particularly useful in traffic intersections with a low percentage of AVs, and highly congested traffic. Specifically, if the traffic-light status has not been changed for a long-time, the condition (C3) will ensure that the controller would try to see whether it is required to change the status.

The motivation behind condition (C2) is less intuitive and needs the introduction of other notations and assumptions, provided in the following.

We denote by $\phi^j(t)$ the traffic-light status at time t for the j-th lane imposed by a traffic-light controller. $\phi^j(t) = 1$ indicates that the traffic-light is green, while $\phi^j(t) = 0$ means that the traffic-light is red. We impose a $\phi^j(t) = -1$ condition, corresponding to a yellow traffic-light, of fixed duration T_{alert} each time that a traffic-light switch occurs.

Whenever the traffic-light controller is triggered, it selects the future traffic-light status ϕ^j_{new} for each lane j, and enables a *sleep mode* of T_{sleep} during which no triggers can occur. T_{silence} is counted after each trigger occurrence.

In general, we assume that ϕ^j_{new} will be applied only d_a seconds later $\phi^j(t_k + d_a) = \phi^j_{new}$, with $d_a \leq T_{\text{sleep}}$. The motivation behind this choice is to provide AVs with future traffic-light information in advance, thus allowing their dynamic optimization, and ensuring their immediate access with the maximum allowed speed to the MZ, when granted.

However, a different d_a delay is considered in the following:

Assumption 3. *If condition (C2) is met and $\phi^j_{new} = 0$, then the traffic-light becomes red when the front vehicle of the $c(i,j)$ enters the MZ.*

This last assumption forces the AV to be the leader in that lane. When the traffic-controller again informs the $c(i,j)$ AV of a future green traffic-light, the vehicle can schedule an optimal acceleration profile, approaching the MZ as soon as the traffic-light turns green, thus increasing the throughput.

The ultimate goal of the RL-based traffic-light controller is to minimize the total number of vehicles queuing at the intersection, thus maximizing the rate of vehicles outflow.

We assume that the behavior of HDVs is modeled according to the Intelligent Driver Model (IDM) [14]. For a detailed description of the HDVs dynamics according to the scheduled traffic light choices see [1].

In the followings, we describe the dynamics of the AVs based on the decision of the traffic-intersection controller.

C. Autonomous vehicle

We assign to the generic $c(i,j) \in C_A$ the following dynamics:

$$\dot{v}_{i,j}(t) = u_{i,j}(t), \quad \dot{p}_{i,j}(t) = v_{i,j}(t). \tag{1}$$

When $c(i,j) \in C_A$ enters the intersection at $t^0_{i,j}$ with an initial speed $v_{i,j}(t^0_{i,j}) = v_0$ and an initial position $p_{i,j}(t^0_{i,j}) = 0$, it can assume the role of leader or follower according to Definition 2 and 3.

If the AV $c(i,j)$ is the *leader* at the CZ, the traffic-intersection controller would communicate with the AV.

If at $t = t_k$ the traffic intersection controller selects $\phi^j_{\text{new}} = 0$, the leader AV schedules a uniform deceleration profile $u_{i,j}(t)$ leading to stop its cruise δ distance away from L_C. We will refer to δ as stopping distance.

Conversely, if at $t = t_k$ the traffic intersection controller selects $\phi^j_{\text{new}} = 1$, the leader AV solves the following optimization problem:

$$\begin{aligned}
&\mathcal{P}_1 : \min_{u_{i,j}(\cdot)} \ \frac{1}{2}\int_{t_k}^{t^m_{i,j}} u^2_{i,j}(t)dt, \\
&\text{subject to: } (1),\ p_{i,j}(t_k) = p_k,\ v_{i,j}(t_k) = v_k, \\
&\qquad v_{i,j}(t) \leq v_{\text{free}},\ v_{i,j}(t) \geq 0 \ \forall\, t \in \left[t_k, t^m_{i,j}\right], \\
&\qquad t^m_{i,j} \geq t_k + d_a + T_{\text{alert}}, \\
&\qquad t^m_{i,j} \leq t_k + 2d_a + T_{\text{alert}}, \\
&\qquad p_{i,j}(t^m_{i,j}) = L_C,\ v_{i,j}(t^m_{i,j}) = v_{free}.
\end{aligned} \tag{2}$$

Note that the latter constraint enforces that at the time where the vehicle enters the intersection, it would enter at the maximum speed. For $t > t^m_{i,j}$ the vehicles are assumed to travel with the maximum speed $v_{i,j}(t) = v_{free}$. In general, we impose $d_a = T_{\text{sleep}}$, the duration after which the action is implemented. However, when condition (C2) is met for the vehicle (q,h), in the lane h conflicting with the lane j, and the traffic-controller decides to enforce a red-light after all the vehicles preceding the AV (q,h) cross the MZ, the vehicle (i,j) can only enter the intersection after $t^m_{q-1,h}+T_{\text{alert}}$. Note that the traffic-intersection controller may not compute the exact time $t^m_{q-1,h}$, but an estimation of the above would be sufficient.

The existence of a solution of (2) depends on d_a. Specifically, by properly setting d_a and L_C, we can ensure the feasibility of $\mathcal{P}_1$.

Note that an already scheduled AV will not update its profile if it receives a new traffic light controller action equal to the one of the previous scheduling.

The AV $c(i,j) \in C_A$ decelerates uniformly, such that it will stop δ distance away from L_C, when the condition (C2) is met and the traffic-intersection controller decides to switch on a red-light after all the vehicles in front of $c(i,j)$ enter the intersection. In this case, the AV $c(i,j)$ will be subsequently promoted to leader. If at t_k it is informed of a future green light status $\phi^j(t_k + d_a) = 1$, it will solve the optimization problem $\mathcal{P}_1$ (Equation (2)).

In all the remaining scenarios, the dynamics of AVs follow the IDM model behavior presented in [1]. This assumption takes place also when the sleep mode of T_{sleep} duration is active due to a previous trigger occurrence.

III. Traffic-light's Decision Process

We now characterize how the traffic-light controller takes its decision using a RL-based algorithm. Henceforth, we assume that the dynamics of the vehicles are discretized with a sampling time T_S, while the traffic-light controller is triggered by the events described in Section II-B, thus resulting in an event-driven discrete time controller.

We model the decision process for the urban intersection traffic-light controller as a discrete-time DDMDP [14] with

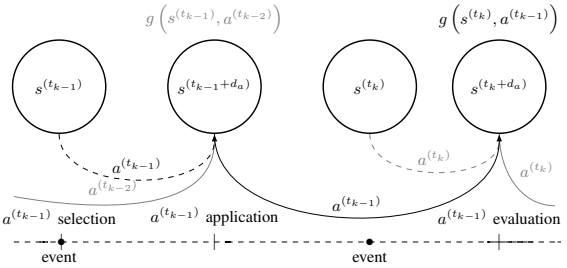

Fig. 2. Schematic of the proposed event-driven DDMDP model.

action delays. We denote by t_{k-1} and t_k two successive time at which two different events occur. The DDMDP is a tuple $\langle S, A, p, g, d_a, d_c\rangle$ where S is the state set, A is the control input (*action*) set, p is the transition probability, $g: S \times A \times S \to \mathbb{R}$ the reward function, d_a is the action delay, and $d_c = d_a$ the delay to observe the reward. Here we consider a constant d_a except in the case in which Assumption 3 occurs. In this later case, d_a depends on the time at which the front vehicle of the AV meeting condition (2) enters the MZ.

Throughout the rest of the paper we use the following notations:

- $s^{(t_k)} \in S$ the state of the RL system at t_k;
- $a^{(t_k)} \in A$ the RL control input (*action*) at t_k;
- $A_T^{(t_k)}$ the set of action applied in $[t_{k-1} + d_a;\ t_k]$;
- $I^{(t_k)} = \left(s^{(t_k)}, A_T^{(t_k)}\right) \in I$ the information needed for optimal action selection at t_k;
- $g_a\left(I^{(t_k)}, a^{(t_k)}\right) = g\left(s^{(t_k)}, a^{(t_{k-1})}\right)$ the reward function.

Note that the reward function evaluated at $t_k + d_a$ does not depend on the action chosen at t_k, but rather on the action decided at the preceding trigger instant t_{k-1}, and applied at $t_{k-1} + d_a$ (Figure 2). We assume that the action chosen at t_k is applied at $t_k + d_a$, only after the reward of the action decided at t_{k-1} is evaluated. Moreover, in our setting, no event is triggered within T_{sleep} time interval.

A. State, Action, and Reward

We now characterize the state, the action and the reward in our setting. First, we introduce some general notations.

Hereinafter we will refer to $\mathbb{N} \cup \{0\}$ as $\mathbb{N}_+$, and to $\mathbb{B} = \{0; 1\}$ as the Boolean domain. We denote by n_l the total number of lanes at the intersection.

1) State: The state $s^{(t_k)} \in S$ at t_k is equal to $X^{(t_k)}$, where $X^{(t_k)} \in {\mathbb{N}_+}^{n_l+1}$ is a vector in which the first n_l elements $x_1^{(t_k)}, \ldots, x_{n_l}^{(t_k)}$ represent the number of vehicles in the CZ of each lane, while the $n_l + 1$-th element is a counter of the number of triggers corresponding to a same consecutive choice of traffic-light controller.

2) Action: The action $a^{(t_k)} \in A$ at t_k is a vector having the number of elements equal to the number of lanes $A := \mathbb{B}^{n_l}$. At t_k, the traffic controller decides which lane to be open, i.e., for which lane the traffic-light will be green at the $t_k + d_a$. Obviously, t_k arises as as the trigger instant of an event. If $a_j^{(t_k)} = 1$, the traffic-light will be green, if $a_j^{(t_k)} = 0$, the traffic-light will be red for lane j at $t_k + d_a$. Note that only those lanes which are non-conflicting can be open simultaneously. Therefore, we reduce the problem imposing that the actions of non-conflicting lanes are equal. Hence, $a_j^{(t_k)} = a_l^{(t_k)}$ if the pair (j, l) are non-conflicting. Thus, the action space can be reduced to only choosing elements for the set $\mathcal{L}$, i.e., the non-conflicting lanes.

3) Reward function: The traffic-intersection controller wants to minimize the queues length at each lane, thus maximizing the outflow of vehicles at a given instance. Hence, we consider the reward-function $g(s^{(t_k)}, a^{(t_{k-1})}) = \| W \odot X_{[1:n_l]}^{(t_{k-1}+d_a)} \|_1 - \| W \odot X_{[1:n_l]}^{(t_k+d_a)} \|_1$, where $W \in \mathbb{R}^{n_l}$ is a weight vector, $X_{[1:n_l]}^{(\cdot)}$ the former n_l elements of $X^{(\cdot)}$, and $\odot$ denotes the element wise multiplication of two vectors. The weights vector W allows to assign to each lane a relative priority. If we want to impose a higher priority for the j-th lane, we will assign to the j-th element of W (w_j) a higher value than the others ($w_i < w_j,\ \forall i \neq j$). Imposing $W = [1, 1, \ldots, 1]$ implies that there is the same relevance for each lane in the optimization.

B. Optimal Policy and Q-Learning

The traffic-intersection controller needs to compute an optimal policy $\pi : I \to A$, i.e., the policy able to maximize the expected value of the discounted cumulative reward

$$\mathbb{E}\left[\sum_{k=0}^{H} \gamma^k g\left(s^{(t_k)}, a^{(t_{k-1})}\right)\right], \tag{3}$$

where $\gamma \in [0, 1]$ is a discounted factor (a constant real value quantifying how much important the future reward is compared to the immediate one), H is the horizon of the optimization problem to be solved, and t_k is the k-th trigger time instant.

We use a tabular Q-Learning algorithm [15] in a non-episodic framework: an off-policy value function approach. Since we performed only simulations, off-policy evaluation is not costly.

Note that in order to find the optimal policy π^* relying on the Q-Learning, we need to evaluate the Q-function for the modified DDMDP. Hence, we will compute $Q(I^{(t_k)}, a^{(t_k)})$ for all $(I^{(t_k)}, a^{(t_k)})$ and then finding the optimal policy as $\pi^* = \arg\max_{a^{(t_k)}} Q(I^{(t_k)}, a^{(t_k)})$.

The reward inherently depends on the dynamics of each vehicle at the intersection, described in II-C, which in turn depends on the decision of the traffic intersection controller. The global dynamics is non-linear and discontinuous.

Being a model-free approach, our proposed method learns the optimal decision without using the model explicitly.

IV. Implementation

A. Set Up

To evaluate the proposed approach we design a MATLAB framework. We consider a merging zone of size $L_M = 30\,\text{m}$, the length of the CZ and the EZ are both equal to $200\,\text{m}$. The maximum speed limit is set to $v_{\text{free}} = 13\,\text{m}\,\text{s}^{-1}$. The vehicles' arrivals follow a Poisson process where we vary the arrival rates. Each vehicle's initial speed $v_{i,j}(t_0)$ is randomly sampled from $[9\,\text{m}\,\text{s}^{-1}, 11\,\text{m}\,\text{s}^{-1}]$, and an initial acceleration $u_{i,j}(t_0)$ is randomly sampled from $[0\,\text{m}\,\text{s}^{-2}, 0.5\,\text{m}\,\text{s}^{-2}]$. We

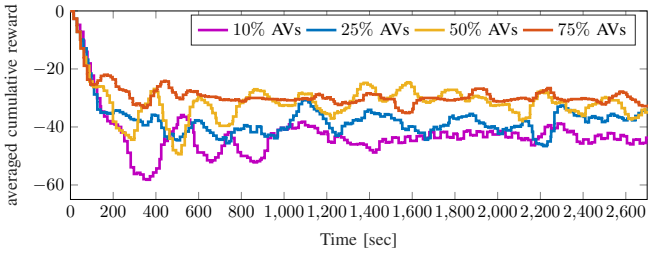

Fig. 3. Averaged cumulative reward during simulations with a vehicles arrival rates of $1125\,\mathrm{veh\,h^{-1}}$ per lane.

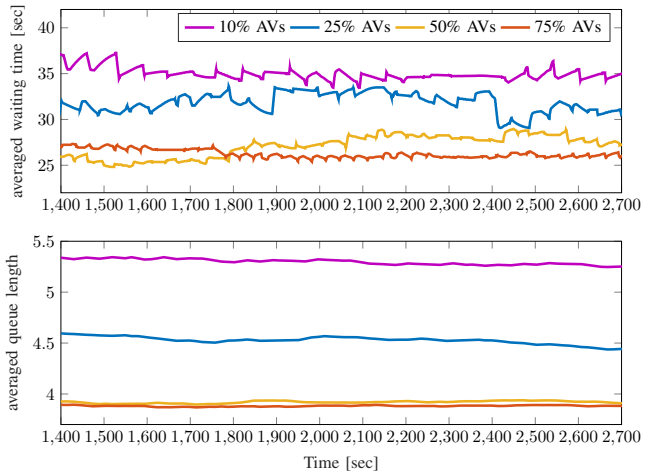

Fig. 4. Moving average of the vehicles waiting time (top), and of queues length in the intersection lanes (bottom) during simulations with a vehicles arrival rates of $1125\,\mathrm{veh\,h^{-1}}$ per lane.

set the capacity of the intersection equal to $N_{\max} = 100$. We investigate four different scenarios, where the fraction of the two categories of vehicles are: 1) 10% of AVs and 90% of HDVs; 2) 25% of AVs and 75% of HDVs; 3) 50% of AVs and 50% of HDVs; 4) 75% of AVs and 25% of HDVs.

We perform the experiments in all scenarios with a fixed arrival rate of $1125\,\mathrm{veh\,h^{-1}}$ at each lane. For the scenario (4), we evaluate our algorithm for two different arrival rates settings ($1800\,\mathrm{veh\,h^{-1}}$, and $1125\,\mathrm{veh\,h^{-1}}$) at each lane. The jam-distance, and the safety time-gap are set at $2\,\mathrm{m}$ and $5\,\mathrm{s}$ respectively for the IDM model, with $\xi = 1.6\,\mathrm{m}$. The complete equations governing the HDVs dynamics are in [1]. Recall that when the traffic-intersection controller informs the AV of a next red traffic-state, the AV stops at a δ-distance away from the intersection. We set $\delta = 20\,\mathrm{m}$, $d_{\text{follow}} = 50\,\mathrm{m}$, $T_{\text{silence}} = 30\,\mathrm{s}$, $T_{\text{sleep}} = 15\,\mathrm{s}$, and $T_{\text{alert}} = 3\,\mathrm{s}$.

The RL controller is trained according to an infinite horizon Q-Learning problem ($H = \infty$ in Equation (3)), and starts with an intersection having zero vehicles. Each simulation stops when the 2000-th vehicle enters the intersection.

For those events in which Assumption 3 doesn't occur, we consider $d_a = T_{\text{sleep}} = 15\,\mathrm{s}$. For simplicity, we consider a framework in which if an event occurs at t_k, the next event shall occur only at $t_{k+1} \geq t_k + T_{\text{sleep}}$. We assume an ϵ-greedy policy with an exploration decays of $1/k$, with k number of performed decision steps (or events). Moreover, following the definition provided in [16], we set $\gamma^k = c_1/(k + c_2)$,

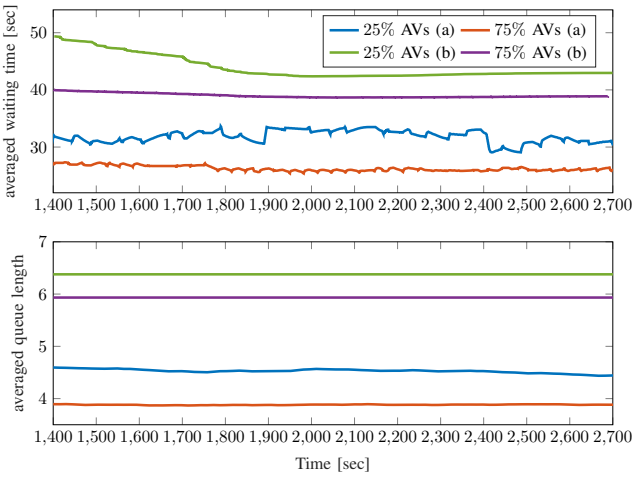

Fig. 5. Comparison between the moving average of vehicles waiting time (top), and of averaged queues length in the intersection lanes (bottom) using (a) the proposed approach, and (b) the approach described in [1], with a vehicles arrival rates of $1125\,\mathrm{veh\,h^{-1}}$ per lane.

where c_1 and c_2 both equal to 1.

B. Results

In Figure 3 we compare the averaged cumulative reward in the performed experiments with a constant arrival rate of $1125\,\mathrm{veh\,h^{-1}}$ per lane. We can observe that a scenario with an higher value of AVs leads to a faster convergence and to an higher averaged cumulative reward. In Figure 4, we can also observe that both the moving average of the vehicles waiting time and of the queues length, at the reward convergence, is lower in a scenario with 75% of AVs than in the remaining scenarios. The average waiting time and queue lengths of vehicles decreases as the fraction of the AVs increases. However, note that reduction of the average queue length is the highest when the AV penetration rate increases from 10% to 25%. The reduction is very small when the penetration rate increases from 50% to 75%. Hence, lower is the percentage of AVs at intersection, higher is the queue reduction as the fraction of AVs increases.

In Figure 5, we compare the results obtained applying the proposed approach in the scenarios (2) and (4), with those obtained performing the approach of [1] in the same settings. The proposed approach appears to be more effective in the management of both the waiting time and the queues length. The reason is that, here, the traffic-light cycle duration can be adapted based on the nature of the vehicle. For example, when the event is triggered because of the condition (C2), and the traffic-controller decides to select the red-light, the red-light will be switched only after all the vehicles preceding to the newly entered AV cross the MZ. Thus, the traffic-light cycle duration is adapted based on the nature of the vehicle, and can be of variable length unlike in [1]. Further, in the above scenario, the traffic-intersection controller coordinates with that AV as it becomes the leader. Such a provision was not there in [1].

In Figure 6 we show the averaged cumulative reward of two simulations performed with a fixed percentage of AVs (75%) and two different values of vehicles arrival

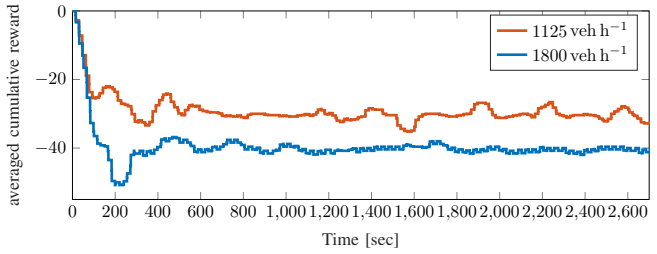

Fig. 6. Averaged cumulative reward during simulations having 75% of AVs.

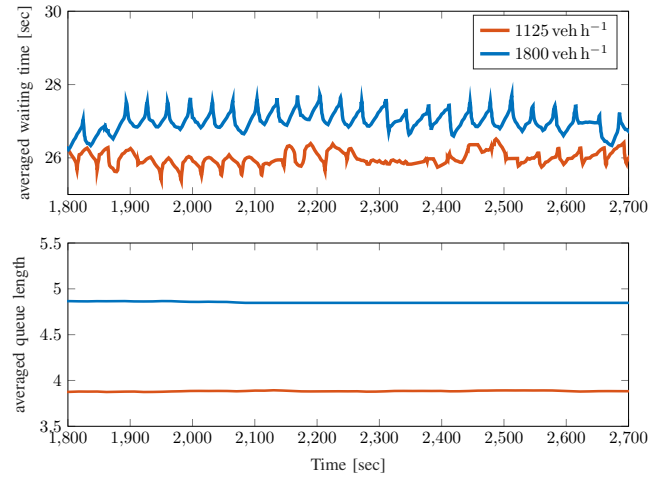

Fig. 7. Moving average of vehicles waiting time (top), and of queues length in the intersection lanes (bottom) during simulations having 75% of AVs.

rate (1125 veh h^{-1}, and 1800 veh h^{-1}per lane). Results show that, with a higher vehicles arrival rate, we observe a lower average cumulative reward than in a scenario with a lower vehicles arrival rate. However, the settling time is comparable, thus highlighting the ability of our approach to converge even with a more congested traffic at the intersection. This is confirmed in Figure 7, where both the moving average of the queues length, and of the vehicles waiting time are reported at the reward convergence of both traffic congestion scenarios. Again, with a higher vehicles arrival rate, queues length and variations in waiting time are at slightly higher values. This is expected. However, the difference is not significant, and still provides evidence of the approach's ability to properly manage traffic as the rate of vehicle flow through the intersection increases.

V. Conclusions and Future Work

We consider an event-driven decision process for a traffic-intersection controller. When an event occurs, the traffic-intersection controller decides whether the traffic-light will be green or red. The action is deployed after a delay which depends on the nature of the event. Specifically, if the event is triggered because an AV enters at a lane where the traffic-light is green, and the traffic-intersection controller decides to put a red-light after all the vehicles in front of the AV enters the intersection, the traffic-controller conveys the decision to the AV. The AV then adapts its dynamics based on the decision. The traffic-intersection controller also informs the AV when the traffic-light will be again green which helps the AV to enter the intersection at the highest speed. We model the decision framework of a traffic intersection controller as a DDMDP, and propose a model-free RL-based algorithm to compute the optimal policy to decide whether the traffic-light will be green or red. Numerical results show that our algorithm outperforms the approach proposed in [1], and is able to properly adapt its policy with different traffic congestion. The investigation of the impact of different reward functions on the policies, along with comparison with other state-of-the art solutions, has been left for future works.

References

[1] E. Salvato, A. Ghosh, G. Fenu, and T. Parisini, "Control of a mixed autonomy signalised urban intersection: An action-delayed reinforcement learning approach," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*. IEEE, 2021, pp. 2042–2047.

[2] Y. Zhang, C. G. Cassandras, W. Li, and P. J. Mosterman, "A discrete-event and hybrid traffic simulation model based on simevents for intelligent transportation system analysis in mcity," *Discrete Event Dynamic Systems*, vol. 29, no. 3, pp. 265–295, 2019.

[3] T. Tettamanti, T. Luspay, B. Kulcsar, T. Peni, and I. Varga, "Robust Control for Urban Road Traffic Networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 385–398, 2014.

[4] J. L. Fleck, C. G. Cassandras, and Y. Geng, "Adaptive Quasi-Dynamic Traffic Light Control," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 3, pp. 830–842, 2016.

[5] S.-W. Chiou, "A robust signal control system for equilibrium flow under uncertain travel demand and traffic delay," *Automatica*, vol. 96, pp. 240–252, 2018.

[6] G. Nilsson and G. Como, "A Micro-Simulation Study of the Generalized Proportional Allocation Traffic Signal Control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 4, pp. 1705–1715, 2020.

[7] D. Liu, W. Yu, S. Baldi, J. Cao, and W. Huang, "A Switching-Based Adaptive Dynamic Programming Method to Optimal Traffic Signaling," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 11, pp. 4160–4170, 2020.

[8] H. Wei, G. Zheng, H. Yao, and Z. Li, "Intellilight: A reinforcement learning approach for intelligent traffic light control," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, ser. KDD '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 2496–2505. [Online]. Available: https://doi.org/10.1145/3219819.3220096

[9] M. A. Wiering, "Multi-agent reinforcement learning for traffic light control," in *Machine Learning: Proceedings of the Seventeenth International Conference (ICML'2000)*, 2000, pp. 1151–1158.

[10] M. Abdoos, N. Mozayani, and A. L. C. Bazzan, "Traffic light control in non-stationary environments based on multi agent q-learning," in *2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2011, pp. 1580–1585.

[11] T. Chu and J. Wang, "Traffic signal control by distributed Reinforcement Learning with min-sum communication," in *2017 American Control Conference (ACC)*, 2017, pp. 5095–5100.

[12] E. Vinitsky, N. Lichtle, K. Parvate, and A. Bayen, "Optimizing mixed autonomy traffic flow with decentralized autonomous vehicles and multi-agent rl," 2020.

[13] E. Vinitsky, A. Kreidieh, L. Le Flem, N. Kheterpal, K. Jang, C. Wu, F. Wu, R. Liaw, E. Liang, and A. M. Bayen, "Benchmarks for reinforcement learning in mixed-autonomy traffic," in *Conference on Robot Learning*. PMLR, 2018, pp. 399–409.

[14] K. V. Katsikopoulos and S. E. Engelbrecht, "Markov decision processes with delays and asynchronous cost collection," *IEEE Transactions on Automatic Control*, vol. 48, no. 4, pp. 568–574, 2003.

[15] C. J. Watkins and P. Dayan, "Q-learning," *Machine learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[16] D. Bertsekas, *Reinforcement learning and optimal control*. Athena Scientific, 2019.