Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 53-2 (2020) 15667–15673

An Independent Trajectory Advisory System in a Mixed-Traffic Condition: A Reinforcement Learning-Based Approach

Majid Rostami-Shahrbabaki*, Tanja Niels, Sascha Hamzehi*, and Klaus Bogenberger***

**Chair of traffic engineering and control, Technical University of Munich, Munich, Germany (Tel: +49(89)289-22438; email: majid.rostami@tum.de, sascha.hamzehi@bmw.de, klaus.bogenberger@tum.de)*

***Institute for Intelligent Transportation Systems, Bundeswehr University Munich, Neubiberg, Germany (Tel: +49(89)6004-3490; e-mail:, tanja.niels@unibw.de)*

Abstract: Achieving smooth urban traffic flow requires reduction of sharp acceleration/deceleration and accordingly unnecessary stop-and-go driving behavior on urban arterials. Traffic signals at intersections, and induced queues, introduce stops along with increasing travel times, stress and emission. In this paper, an independent reinforcement learning-based approach is developed to propose smooth traffic flow for connected vehicles enabling them to skip a full stop at queues and red lights at urban intersections. Two reward functions, i.e., a fuzzy reward engine and an emission-based reward system, are proposed for the developed Q-learning scheme. Another contribution of this work is that the necessary information for the learning algorithm is estimated based on the vehicle trajectories, and hence, the system is independent. The proposed approach is tested in a mixed-traffic condition, i.e., with connected and ordinary vehicles, via a realistic traffic simulation with promising results in terms of flow efficiency and emission reduction.

Keywords: Connected vehicles, Emission, Fuzzy, GLOSA, Traffic state estimation, Reinforcement learning.

1. INTRODUCTION

In the last decades, vehicle fuel consumption and relevant greenhouse gas (GHG) emissions have been a major concern of our society. It is assumed that around 60% of global oil consumption is consumed by the transportation sector (Jollands *et al.*, 2010). The European Union roadmap is to attain the reduction of greenhouse gas emissions by 80%-95% until 2050 (compared to 1990 levels). Decreasing the Carbon Dioxide (CO2) emissions caused by internal combustion engine vehicles could substantially help reach this goal. A great deal of these CO2 emissions can be reduced by more slow-and-go motions rather than stop-and-go motions caused by signalized intersections (Eckhoff, Halmos and German, 2013).

One important approach in the field of Intelligent Transportation Systems (ITS) which improves traffic flow efficiency is referred to as Green Light Optimal Speed Advisory (GLOSA) (van Leersum, 1985), which attempts to coordinate vehicles crossing a traffic signal with a known and usually fixed signal plan. Its goal is to provide the optimal speed advisory for every vehicle to be arrived at the intersection during the green phase, for example, via Connected Vehicle (CV) technology, which provides Vehicle-to-Infrastructure (V2I) and Vehicle-to-Vehicle (V2V) communications. Another similar strategy is called eco-driving. With increasing awareness of the relationship between speed trajectory and fuel consumption, eco-driving typically consists of changing driving behavior by accelerating slowly, driving smoothly, and reducing high speeds. The major objective of eco-driving is providing real-time driving advice to individual vehicles in order to reduce fuel consumption and CO2 emission levels while this may increase vehicle travel time in some cases (Yang, Rakha and Ala, 2017).

In an ideal case where the Signal Phase and Timing Information (SPaT) is available, such approaches can propose optimal speed advisory for timely arrival at the green phase or fuel savings. In practice, even with new developments in V2I technology, direct access to the signal timing information and real-time state of the traffic lights may be accessible only for signals on main corridors or a small-scale network, whereas collecting such information for large areas (e.g. region or nationwide) directly from controllers can be very challenging. Thus, this fact increases the need for SPaT estimation instead (Hao *et al.*, 2012).

1.1 Literature review

Over the years, more research efforts have been directed to the developments of eco-driving strategies and GLOSA systems to improve the energy efficiency for traveling along signalized intersections in urban areas. With the assumption of availability of SPaT information via V2I communication, methodologies to enhance traffic flow efficiency while approaching a signalized intersection are developed (Barth *et al.*, 2011; Rakha and Kamalanathsharma, 2011). Asadi and Vahidi (2011) proposed the use of upcoming traffic signal information within the vehicle's adaptive cruise control system to reduce idle time at stoplights and fuel consumption.

Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2020.12.2550

Performance comparison of a conventional single-segment GLOSA with a multi-segment approach is carried out in (Seredynski, Dorronsoro and Khadraoui, 2013). In (Eckhoff, Halmos and German, 2013), potentials and limitations of the GLOSA systems in a realistic, large scale simulation study are investigated. A multi-stage optimal control formulation is proposed (He, Liu and Liu, 2015) to obtain the optimal vehicle trajectory on signalized arterials, where both vehicle queue and traffic light status are considered. In (Yang, Rakha and Ala, 2017), an Eco-CACC (Cooperative Adaptive Cruise Control) algorithm is developed that computes the fuel-optimum vehicle trajectory through a signalized intersection by ensuring that the vehicle arrives at the intersection stop bar just as the last queued vehicle is discharged. A partially automated vehicle system with an eco-approach and departure feature (called the GlidePath Prototype), is developed in (Altan *et al.*, 2017). A modified GLOSA algorithm that considers the formed intersection queues and queue discharge headways for each vehicle position is considered in (Njobelo *et al.*, 2018). A consensus and optimal speed advisory model (SAM) for CV platoon at an isolated signalized intersection in the presence of a mixed traffic scenario is proposed (Yu *et al.*, 2019).

In recent years, increasing amounts of attention have been paid to the development of fuzzy systems (Bogenberger, Vukanovic and Keller, 2002) and reinforcement learning approaches in the area of intelligent transportation systems such as providing speed limit control in a stochastic traffic environment (Zhu and Ukkusuri, 2014), optimizing traffic flow in highways, (Walraven, Spaan and Bakker, 2016), maximizing the probability of arriving on time (Cao *et al.*, 2017), and real-time estimation of lane-based queue lengths (Lee et al., 2019). In a very recent work, a reinforcement learning-based car following model in order to obtain an appropriate driving behavior to improve travel efficiency at signalized intersections is proposed (Zhou, Yu and Qu, 2019). However, it is assumed that all vehicles are connected and automated, and thus, no queue is formed and all produced actions can be implemented. In case of a mixed-traffic condition, ordinary vehicles may prevent connected vehicles from speeding up or changing their lanes which consequently effect the implementation of the proposed trajectories.

Overall and to the best knowledge of the authors, current eco-driving and GLOSA systems rely on the deterministic information of the signal timing and, in some cases, queue information. Although, through current V2X technology, having access to such information is theoretically possible, in practical applications, due to many traffic control operators worldwide, different database structures, etc., this might not be indeed straightforward. Hence, it might be of great importance for car manufacturers such as BMW or navigation systems such as Google Maps to have an independent system which provides, on one hand, a real-time estimation of traffic conditions and SPaT information, and on the other hand, suggests optimal trajectories or route recommendations.

1.2 Outline

The goal of this present work is to develop an independent trajectory advisory system that provides trajectories for connected vehicles traveling in urban arterials, based on the estimated SPaT and queue information. Compared to other similar works where a deterministic optimal solution is derived, in this paper, a Q-learning approach is developed since only real-time estimates of needed information are utilized. In the first step, based on connected vehicle data, the necessary traffic states are estimated. The estimated signal timings and queue tail locations provide global information for states in the Q-learning scheme. Based on such information and the speed of any individual vehicle, the Q-learning agent provides a set of actions that determine the desired trajectory of the vehicle. At the time each vehicle leaves the intersection, its experienced trajectory is evaluated by means of the proposed reward functions. A fuzzy reward engine is developed which evaluates the vehicle trajectory and, based on its average speed and maximum acceleration, assigns high rewards to the fastest and smoothest trajectories. Trajectories encountering low speed and rapid acceleration/deceleration are given lowest rewards. Another reward system is proposed that takes the amount of CO2 emissions produced by vehicles into account as a reward criterion. Since emissions are directly related to the acceleration/deceleration patterns and the idling period, less produced emissions by vehicles during their journey indicate best eco-driving strategies which are highly rewarded, and vice versa.

In summary, at each sampling time, the reinforcement learning agent receives global states, i.e., queue tail location and SPaT information of a signalized link, and local states, i.e., vehicles' location and speed, and then proposes advisory trajectories for connected vehicles as a new action. Finally, when a vehicle passes the upcoming intersection, the set of state-action-pairs are evaluated via reward engines and the given reward is used for updating the Q-values.

The rest of this paper is organized as follows. In the next section, the proposed approach along with the related theories are presented. A realistic simulation is carried out in Section 3 where associated results and related discussion are presented. Finally, main conclusions and future works are outlined in Section 4.

2. PROPOSED APPROACH

A sketch of the proposed approach is illustrated in Fig. 1. It is assumed that some percentage of vehicles are connected, and hence, they can send their location and speed information and receive the proposed actions. Speed and location information of all connected vehicles are stored and treated in a proxy server where, mainly, a map matching algorithm maps the vehicles to their corresponding signalized link. The map-matched data are then sent to the state estimator. The estimator comprised of two main agents, i.e., the link agent and the node agent. The link agent, which corresponds to a signalized link, receives the information of all connected vehicles traveling in that link and estimates the queue information including queue tail location, the number of vehicles in the queue and the corresponding link outflow. This obtained information of all links approaching an intersection is then conveyed to a node agent that corresponds to one intersection. The node agent estimates the SPaT information of the given intersection. Each

signalized link has one learning agent which constructs the state space based on global states, i.e., queue tail and SPaT information, and local states, i.e., location and speed of every connected vehicle, and then suggests an action such as the new acceleration or target speed for each connected vehicle. As soon as each vehicle leaves that link at the intersection, its trajectory is evaluated, and a reward is assigned for the proposed action which is used for updating the Q-values of the learning agent. The complete procedure is explained in detail in the following sub-sections.

2.1 State estimation

The first step of the estimation procedure in the link agent is identifying the queue tail location. To this end, at every time step, the measurements from connected vehicles are appropriately treated. Then, based on a velocity threshold, connected vehicles are clustered to the group of (virtually) stopped or the group of moving vehicles. The distance of the farthest connected vehicle from the stop bar in the group of stopped vehicle is, in fact, the criterion for queue estimation. This first rough queue tail estimate, L_q, is calculated as in (1) and is compensated afterwards to reduce the error caused by low penetration rates since in a low penetration rate of connected vehicles, there may be farther, non-connected, vehicles queuing behind the last connected vehicle as explained in (Rostami Shahrbabaki *et al.*, 2018).

$$L_q = \max_i(d_i) \tag{1}$$

where $i \in I = \{n | v_n \leq v_{min}\}$ for $n = 1, \dots, N$. v_{min} is the speed threshold that designates vehicles to either stopped or moving groups, N is the number of connected vehicles, and d_i and v_i are the distance of the i^{th}-connected vehicle measured from the downstream end of the link and its corresponding speed, respectively.

The number of vehicles in the queue, $\hat{N}(k)$, and link outflow, $\hat{q}(k)$, are then estimated via (2)-(3).

$$\hat{N}(k) = \frac{\lambda A \hat{L}_q(k)}{L_v(v(k)+A)} \tag{2}$$

where $\hat{L}_q(k)$ is the estimate of the queue tail after compensation of the queue tail dislocation error, λ is the number of lanes, L_v is the average headway of queuing vehicles, A is the queue wave speed, and $v(k)$ is the average speed of connected vehicles inside the queue.

$$\hat{q}(k) = \hat{N}(k)v(k) \tag{3}$$

For a given intersection in the urban network, all achieved estimates for the signalized links approaching that intersection are passed to the node agent, which takes the following steps to estimate the SPaT information which are extensively described in (Rostami-Shahrbabaki *et al.*, 2020):

- Outflow enhancement via connected vehicles data crossing the intersection.
- Cycle time estimation based on autocorrelation of each outflow signal.
- Phase order estimation based on cross-correlation analysis of all approaching links' outflows.
- Outflow quantization.
- Green and red time estimation of the traffic signal based on pulse width clustering of the quantized outflow.

Finally, the queue tail estimation and SPaT information are conveyed to the reinforcement learning agent to construct the global states of the environment.

2.2 Reinforcement learning

This subsection briefly introduces the basics of reinforcement learning (RL) and describes the parameter settings used in this study.

Commonly known, the Markov Decision Process (MDP) describes dynamic and discrete decision processes within a stochastic environment in a formal manner.

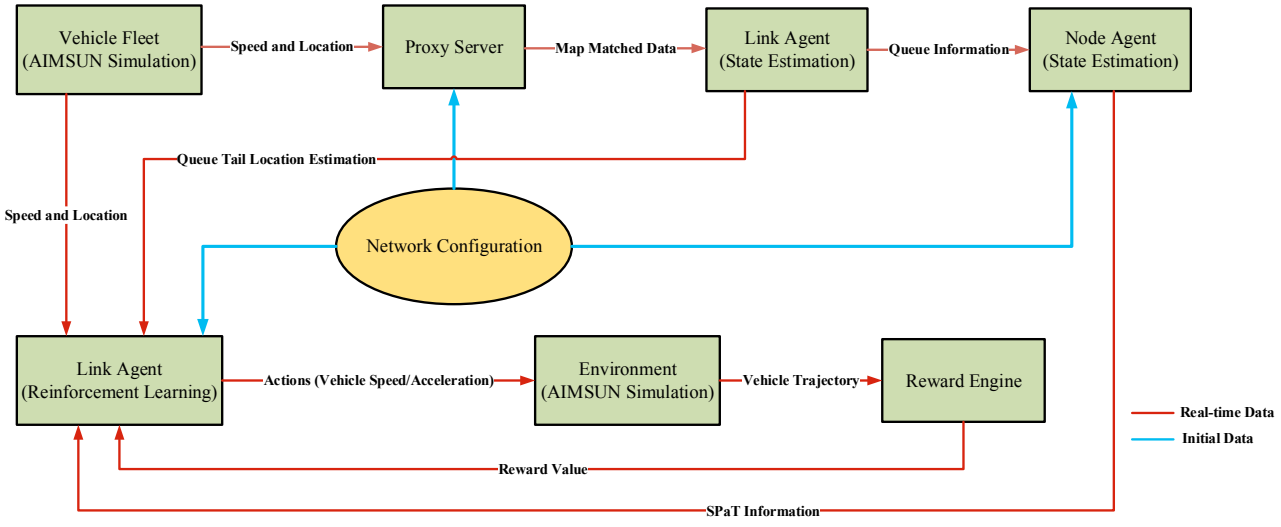

Fig. 1. Schematic of the proposed independent trajectory advisory system

The MDP framework serves as basis for reinforcement learning (Sutton, Barto and Williams, 1992) and if the existing environment is fully observable, the environment state completely characterizes the decision process. Equally to the MDP, a Reinforcement Learning (RL) algorithm considers discrete states and provides actions as output. Further detailed, the RL process is denoted by a tuple (S, A, R, T, γ) where $S = [s_0, s_1, \dots, s_n]$ denotes a set of n discrete states and $A = [a_0, a_1, \dots, a_m]$ denotes a set of m discrete action outputs. The RL algorithm often called learner or agent requires a reward mechanism denoted by the matrix R which contains all reward values for each state-action-pair tuple (s_i, a_j). During the learning phase the agent tries to maximize the accumulated probabilistic reward for a range of learning epochs containing a quantity of iteration steps k. Further, the accumulated scalar reward is denoted by $r^k(s_i, a_j) = \sum_{i=0}^{n} \sum_{i=0}^{m} R^k(s_i, a_i)$ for each iteration k. Finally, the agent tries to learn a policy π, i.e. the decision or in other words the state-action sequence that maximizes the agent's reward over time/iterations k.

Usually, the agent requires a notion of how the stochastic environment behaves, i.e. denoted by the transition matrix T which contains the transition probabilities. The transition matrix describes how likely it is to change from one state to the other and usually is formally described by $T^{n \times m}: S \times A \times S \rightarrow [0,1]$. For completely deterministic environments the transition probabilities are set to 1. In order to model the uncertainty of future and current reward updates, the RL process introduces a discount factor γ with respect to $0 \leq \gamma < 1$.

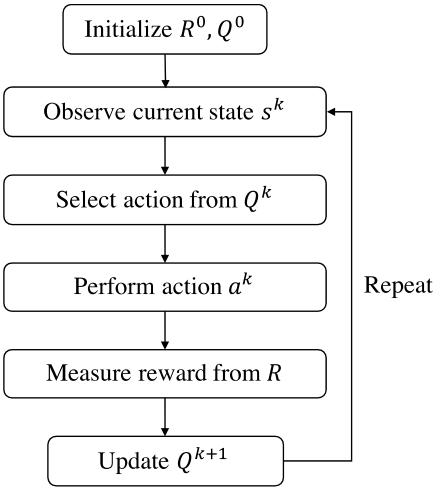

Fig. 2. Q-learning process flow chart

However, in the case of Q-learning (Watkins and Dayan, 1992), the agent learns to act optimally in a Markovian environment model by interacting with the environment. This is possible by storing the experience of the interacting agent in the matrix Q. The Q matrix thereby contains a set of scalar Q-values which again bootstrap the expected discounted reward for all executed actions a for each observed environment state s. Finally, the agent must learn to estimate the Q-values that result in the optimal policy π, where the Q-values ideally may converge while learning. The learning process ultimately can be summarized as follows. During the sequence of learning epochs and iterations the agent:

- observes the current state s_n^k
- selects and executes an action a_n^k
- observes the subsequent state s_n^{k+1}
- receives an immediate reward r^k for selecting an action and,
- updates the experience stored in Q^k, where the individual q-values are influenced by the learning rate α^k. The q-values are updated by the following equation.

$$Q^{k+1}(s_i, a_j) \leftarrow (1 - \alpha^k) \cdot Q^k(s_i, a_j) + \alpha^k \cdot \{R^k(s_i, a_j) + \gamma \max_{a^{k+1}} [Q^k(s^{k+1}, a^{k+1})] - Q^k(s^k, a^k)\}. \quad (4)$$

The process finally can be visualized via Fig. 2, where k is the quantity Fig. 2. Q-learning process flow chart.

2.3 Reward functions

As stated in the previous section, each state-action-pair used in the Q-learning framework must be evaluated. Two reward functions are defined which receive trajectories of each connected vehicle as an input argument and generate a reward value which is used in the next step for updating Q-values as depicted in Fig. 1 and Fig. 2.

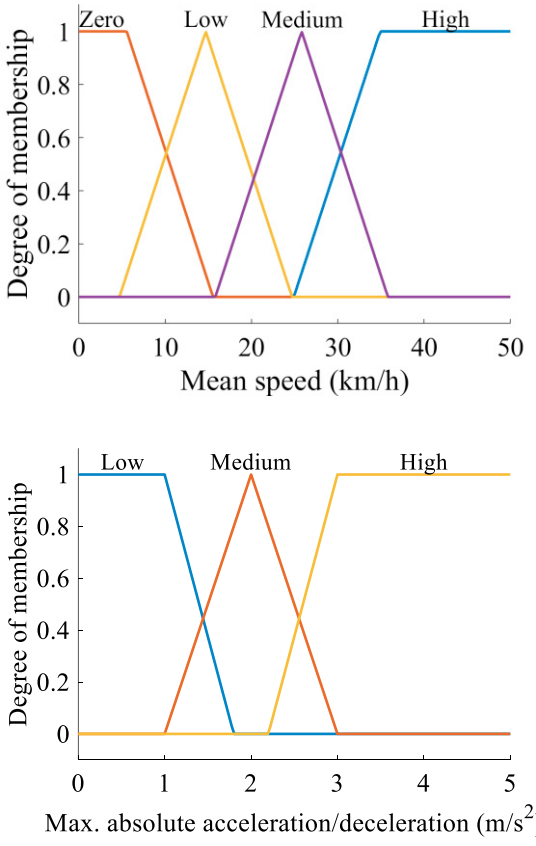

Fig. 3. Input membership functions of fuzzy reward engine

A mamdani fuzzy reward engine is designed with two inputs, i.e., average speed and maximum absolute acceleration/deceleration of the connected vehicles. Defined membership functions for each input of the fuzzy reward engine are shown in Fig. 3. Based on these membership functions, comprehensive rules as given in Table 1 are developed in order to assign higher rewards for the fastest and smoothest trajectories. The given reward decreases with lower average speed and sharper acceleration and/or deceleration rates. Output membership functions for the proposed rewards are shown in Fig. 4 which map the given fuzzy reward to a real value.

Table 1. Set of rules defined for the fuzzy reward engine

		Mean Speed			
		Zero	Low	Medium	High
Max. of abs. acc./dec.	Low	Low	High	High	VHigh
	Medium	Low	Low	Medium	High
	High	Low	Low	Low	High

Another reward function is developed that guarantees the eco-driving behaviour of the vehicles. This rewarding system is based on the amount of CO2 emissions of the vehicles. To this end, the emission model proposed by Panis et al., (2006) is utilized and the amount of CO2 emissions is calculated based on (5), over time, for each vehicle trajectory. The trajectories with less amount of produced emissions deserve higher rewards.

$$E(t) = \max[E_0, f_1 + f_2 v(t) + f_3 v(t)^2 + f_4 a(t) + f_5 a(t)^2 + f_6 v(t) a(t)] \quad (5)$$

where $v(t)$ and $a(t)$ are the instantaneous speed (km/h) and acceleration (m/s^2) of each vehicle at time t, E_0 is a lower limit of emission (g/s), and f_1 to f_6 are emission constants.

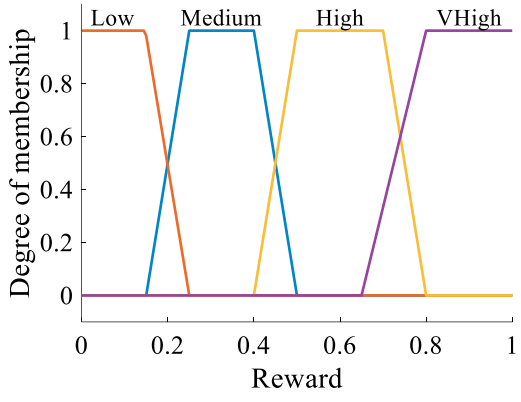

Fig. 4. Output membership functions of fuzzy reward engine

3. SIMULATION AND RESULTS

The described approach was implemented and tested using the microsimulation platform Aimsun (Barceló and Casas, 2005). Without loss of generality and for the sake of simplicity, a set of two links, each 500 m long, connected by one node is assumed. The node is assumed to be controlled by a fixed-time traffic signal with a cycle time of 90 seconds, and a green time of 45 seconds in each cycle. The signal timing is assumed to be unknown for the analysis and is estimated as explained in Section 2.1.

The learning agent is implemented using the Python programming language and the interface between the intersection control and the traffic model within the simulation is realized by using the Aimsun Application Programming Interface (API). For each connected vehicle, the link agent proposes a speed for approaching the intersection that is based on the Q-learning strategy. To this end, a five-dimensional sate vector (d, v, l, p, s, v) of the state of an approaching vehicle along with the global states is constructed, where d is the distance of the vehicle to the intersection, v is the current velocity of the vehicle, l is the estimated queue length, p is the current signal phase, and s is the number of seconds since the signal change. When the vehicle is at a distance of approximately 500 meters to the intersection, the state vector is passed to the Q-learning strategy. The action, i.e. the new desired speed for the vehicle, is chosen such that $new\_desired\_speed = argmax_{speed}\, Q(state, speed)$ for all speed values between 10 km/h and the maximum allowed speed (50 km/h). In case this value is not unique, a random desired speed all speeds $speed^*$ with $speed^* = argmax_{speed}\, Q(state, speed)$ is chosen. This speed is passed as the desired speed of the vehicle. Other than that, default driving behaviour implemented by Aimsun is not changed. This means that the vehicle accelerates or decelerates (respecting maximum acceleration and deceleration values) until the new desired speed is reached. Safety headways to vehicles in front are respected and if vehicles approach a red light they decelerate in front of the traffic signal (eventually reaching to a full stop, if necessary). Instead of updating the Q-values immediately, rewards for the algorithm can only be calculated after the vehicle has passed the traffic signal. The Q-value of the respective state-action-pair is then updated using the reward functions described in the previous section. Non-connected vehicles move using the default driving behaviour implemented by Aimsun, their desired speed lies between 0.9 and 1.3 times the speed limit. A traffic demand of 600 veh/h with 25% rate of connected vehicles is considered.

As explained in Section 2.3, two different reward functions are used for training the model. In the first approach, vehicle trajectories are rewarded based on the developed fuzzy reward engine. In the second approach, produced emissions are evaluated as in (5). Received moving average fuzzy rewards and emission-based rewards for connected vehicles are shown in Fig. 5 and Fig. 6, respectively. There is a chattering behaviour at the early stage of learning in both accumulated rewards which is mainly due to the random actions induced by the exploration nature of Q-learning. Both rewards progress over time, i.e., incrementally for fuzzy rewards and decrementally for emission rewards, which reveals the fact that vehicles are learning how to adjust their speed to achieve better trajectories and hence receive better rewards.

The trajectories of connected vehicles vs ordinary vehicles are plotted in Fig. 7. It is evident that connected vehicles have learned how to adjust their speed in order to avoid full stop at the stop bar and also to continue with their speed in case the can catch current green signal phase.

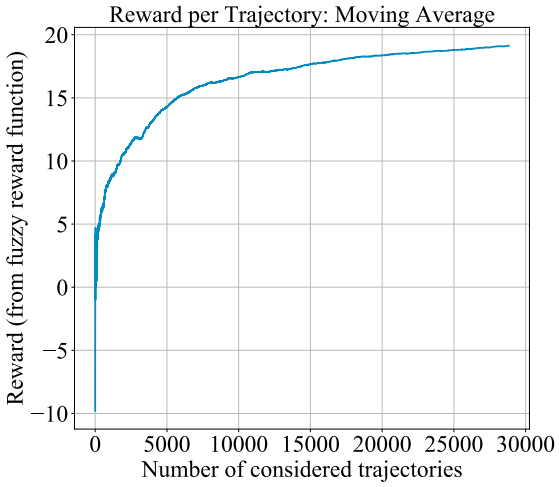

Fig. 5. Average fuzzy rewards

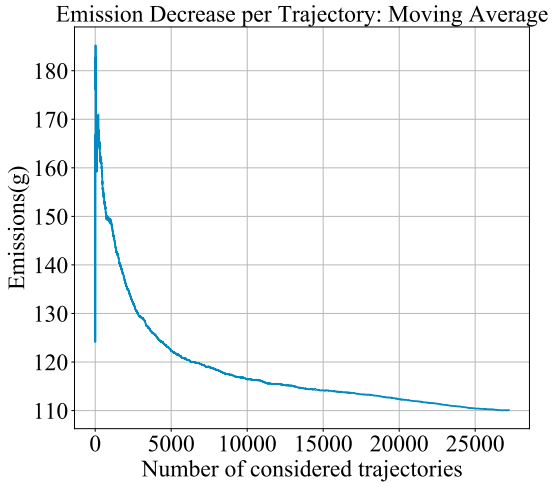

Fig. 6. Average emission rewards

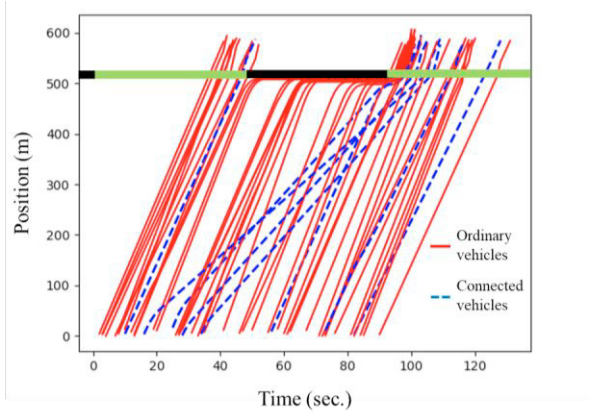

Fig. 7. Trajectories of connected vs ordinary vehicles

The presented algorithm allows for a number of possible extensions, e.g. consideration of several consecutive intersections, evaluation in oversaturated traffic conditions, and application of adaptive traffic signal timing which are parts of authors' future work.

4. CONCLUSIONS

In this paper, a Q-learning approach for proposing appropriate trajectories for connected vehicles when approaching a signalized intersection was introduced. It focuses on reducing emissions and providing smooth trajectories with low acceleration and deceleration rates. The core of the approach is SPaT estimation together with a learning agent and two reward functions. The approach was described, implemented, and tested. Proposed trajectories were evaluated along with resulting emissions and the rewards defined by the fuzzy reward engine.

The simulation results indicated that, over time, the connected vehicles learned how to adjust their speed in order to catch the current or next green time without experiencing the queue during the red signal phase. Accumulation of rewards, measured via both functions, is also an indicator of learning process implemented by proposed methodology.

In order to further improve the quality of conclusions, a sensitivity analysis will be conducted that evaluates the effect of considered vehicle parameters, percentage of connected vehicles, signal programs, and section length.

5. REFERENCES

Altan, O. D. *et al.* (2017) 'GlidePath: Eco-Friendly Automated Approach and Departure at Signalized Intersections', *IEEE Transactions on Intelligent Vehicles*, 2(4), pp. 266–277.

Asadi, B. and Vahidi, A. (2011) 'Predictive cruise control: Utilizing upcoming traffic signal information for improving fuel economy and reducing trip time', *IEEE Transactions on Control Systems Technology*. IEEE, 19(3), pp. 707–714.

Barceló, J. and Casas, J. (2005) 'Dynamic network simulation with AIMSUN', *Simulation approaches in transportation analysis*, pp. 57–98.

Barth, M. *et al.* (2011) 'Dynamic ECO-driving for arterial corridors', in *IEEE Forum on Integrated and Sustainable Transportation Systems,*. IEEE, pp. 182–188.

Bogenberger, K., Vukanovic, S. and Keller, H. (2002) 'ACCEZZ - Adaptive fuzzy algorithms for traffic responsive and coordinated ramp metering', in *Proceedings of the International Conference on Applications of Advanced Technologies in Transportation Engineering*, pp. 744–753. doi: 10.1061/40632(245)94.

Cao, Z. *et al.* (2017) 'Maximizing the Probability of Arriving on Time : a Stochastic Shortest Path Problem', *Thirty-First AAAI Conference on Artificial Intelligence*, pp. 4481–4487.

Eckhoff, D., Halmos, B. and German, R. (2013) 'Potentials and limitations of Green Light Optimal Speed Advisory systems', in *IEEE Vehicular Networking Conference*. IEEE, pp. 103–110.

Hao, P. *et al.* (2012) 'Signal timing estimation using sample intersection travel times', *IEEE Transactions on Intelligent*

Transportation Systems. IEEE, 13(2), pp. 792–804.

He, X., Liu, H. X. and Liu, X. (2015) 'Optimal vehicle speed trajectory on a signalized arterial with consideration of queue', *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd, 61, pp. 106–120.

Jollands, N. *et al.* (2010) 'The 25 IEA energy efficiency policy recommendations to the G8 Gleneagles Plan of Action', *Energy Policy*, 38(11), pp. 6409–6418. doi: 10.1016/j.enpol.2009.11.090.

Lee, S. *et al.* (2019) 'An advanced deep learning approach to real-time estimation of lane-based queue lengths at a signalized junction', *Transportation Research Part C: Emerging Technologies*, 109, pp. 117–136.

van Leersum, J. (1985) 'Implementation of an advisory speed algorithm in TRANSYT', *Transportation Research Part A: General*. Pergamon, 19(3), pp. 207–217.

Njobelo, G. *et al.* (2018) 'Enhancing the green light optimized speed advisory system to incorporate queue formation', in *Transportation Research Board 97th Annual MeetingTransportation Research Board*, pp. 1–11.

Panis, L. I., Broekx, S. and Ronghui Liu (2006) 'Modelling instantaneous traffic emission and the influence of traffic speed limits', *Science of the Total Environment*, 371, pp. 270–285.

Rakha, H. and Kamalanathsharma, R. K. (2011) 'Eco-driving at signalized intersections using V2I communication', in *IEEE Conference on Intelligent Transportation Systems*. IEEE, pp. 341–346.

Rostami-Shahrbabaki, M. *et al.* (2020) 'Intersection SPaT estimation by means of single-source connected cehicle data', in *TRB 99th Annual Meeting*.

Rostami Shahrbabaki, M. *et al.* (2018) 'A data fusion approach for real-time traffic state estimation in urban signalized links', *Transportation Research Part C: Emerging Technologies*, 92(November 2017), pp. 525–548.

Seredynski, M., Dorronsoro, B. and Khadraoui, D. (2013) 'Comparison of Green Light Optimal Speed Advisory approaches', in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, pp. 2187–2192.

Sutton, R. S., Barto, A. G. and Williams, R. J. (1992) 'Reinforcement learning is direct adaptive optimal control', *IEEE Control Systems Magazine*. Publ by American Automatic Control Council, 3(2), pp. 19–22. doi: 10.1109/37.126844.

Walraven, E., Spaan, M. T. J. and Bakker, B. (2016) 'Traffic flow optimization: A reinforcement learning approach', *Engineering Applications of Artificial Intelligence*, 52, pp. 203–212.

Watkins, C. J. C. H. and Dayan, P. (1992) 'Q-learning', *Machine Learning*. Springer Science and Business Media LLC, 8(3–4), pp. 279–292.

Yang, H., Rakha, H. and Ala, M. V. (2017) 'Eco-Cooperative Adaptive Cruise Control at Signalized Intersections Considering Queue Effects', *IEEE Transactions on Intelligent Transportation Systems*. IEEE, 18(6), pp. 1575–1585.

Yu, S. *et al.* (2019) 'Consensus and optimal speed advisory model for mixed traffic at an isolated signalized intersection', *Physica A: Statistical Mechanics and its Applications*. Elsevier B.V., 531, p. 121789.

Zhou, M., Yu, Y. and Qu, X. (2019) 'Development of an Efficient Driving Strategy for Connected and Automated Vehicles at Signalized Intersections: A Reinforcement Learning Approach', *IEEE Transactions on Intelligent Transportation Systems*. IEEE, pp. 1–11.

Zhu, F. and Ukkusuri, S. V. (2014) 'Accounting for dynamic speed limit control in a stochastic traffic environment: A reinforcement learning approach', *Transportation Research Part C: Emerging Technologies*. Elsevier Ltd, 41, pp. 30–47.