ReLight-WCTM: Multi-Agent Reinforcement Learning Approach for Traffic Light Control within a Realistic Traffic Simulation

Péter Pálos, Árpád Huszák
Dept. of Networked Systems and Services, Faculty of Electrical Engineering and Informatics
Budapest University of Technology and Economics
Budapest, Hungary
peterpalos@edu.bme.hu, huszak@hit.bme.hu

***Abstract*—Although traffic management methods are constantly evolving, traditional solutions are unable to adapt to the current dynamics of traffic. Machine learning methods provide promising results, but scientific dissertations usually work only on the theoretical basis of the models and do not take into account the legal requirements of traffic management. In the presented paper we propose a deep reinforcement learning-based multi-agent model called ReLight-WCTM that insists on maintaining reality at several points. We compared our model with the original signal setting of a real road network based on different metrics. According to the results, it can be stated that ReLight-WCTM exceeded the baseline settings in all parameters, presumably it can be an actual traffic management alternative.**

***Keywords*— Traffic Signal Control, Machine Learning, Deep Reinforcement Learning, Decentralized Multi-Agent**

I. Introduction

Humanity is increasingly confronted with the depletion of our natural resources, its negative impact on ecology, so the issue of sustainability has become the focus of many researches. We must seize every opportunity we have to reduce our ecological footprint. One of the biggest causes of which is the ever-increasing vehicle traffic. The development of technology and urban detector infrastructure allows the application of machine learning algorithms, which is the basis of the present paper.

The first traffic light was drafted in 1912 by a police officer in Salt Lake City and two years later was placed into traffic. Although more than 100 years have passed since then, contemporary goals have not changed: traffic management systems are responsible for avoiding congestion, better capacity utilization, improving safety and keep environmental considerations in focus.

Commonly used traffic control systems are not able to respond to momentary changes in traffic, which is possible by the intelligent decision making presented in this paper. We examined reinforcement learning method, a specific branch of machine learning, the essence of which is the interaction of the learning agent with the environment and the maximization of the reward based on predefined metrics. Many studies have been presented dealing with optimizing the control of a single intersection, however, far fewer studies deal with the control of complex networks. The primary goal of our research was to preserve a high level of reality, as theoretical models often oversimplify the circumstances and do not meet the expected legal regulations. We developed the presented model in cooperation with the city management and transport professionals, which is a major step forward from theoretical analysis to real industrial use. Hence the name Reality Light WCTM or ReLight-WCTM for short where WCTM stands for the reward function. The training environment, i.e. the road network, was created on the basis of the real structure of the main road of a Hungarian city, Pécs, which contains a total of nine intersections controlled by traffic lights. The nodes are managed on the basis of a cooperative multi-agent solution, in which the agents also share information with each other. The Hungarian road operator provided for us the professionally designed signal phase plan currently in use in the city, which functioned as a baseline model. Evaluation of ReLight-WCTM and baseline setting was performed based on CO_2 emission, waiting time, average speed and the number of halting vehicles.

The structure of the article is as follows. We provide a brief overview of traditional and up-to-date traffic control methods supplemented by the technical background used in section II. Section III. is about the simulation environment and model parameters. In section IV. we present the achieved results, while in the last section we summarize the research.

II. Related Works

Several approaches have been proposed in the field of traffic management. Traditional solutions work with predefined phase durations, which usually vary depending on the time of day [1]. Signal control systems, which can also be considered classical, are able to adapt to traffic dynamics based on sensor data. One of the most popular solutions is the Webster method developed by Koonce et al [2], which tries to minimize the travel time at a single intersection using the so called Webster equation. During operation, it determines cycle duration and phase split. The model created by Varaiya focuses on the pressure on the signal phase, hence the name Max-pressure [3]. The pressure is determined by the difference between the number of vehicles

waiting for a given phase and the number of vehicles leaving the corresponding roads. The SCATS [4] and SCOOT [5] systems choose from predefined phase plans to optimize congestion and saturation. There is a great advantage in generating green waves on the road network. To facilitate this, Roess, Prassas, and McShane developed GreenWave, which reduces the number of stops by taking into account offsets between intersections [1].

Reinforcement learning has long been the focus of research in terms of traffic management. One of the most effective methods is Q-learning, the success of which has been proven by numerous studies [6] [7] [8] [9]. For more complex models, its extension with deep learning has been proposed [10]. Double DQN [11] [12], Dueling DQN [13] [14], and a mixture of the two algorithms, D3QN, were also examined for optimal signal control design [15].

A. Q-learning Variants

Q-learning [16] is one of the most commonly used reinforcement learning algorithms due to its efficiency and simplicity. Although the A3C algorithm [17] is already beginning to take its place, more advanced Q-learning versions still serve as the basis for RL. Its operating principle was described in 1989 by Watkins, in which a Markov decision process (MDP) optimum search is performed by evaluating the effectiveness of possible actions for a given condition, calculated from the amount of reward or punishment and the value of the new state. This calculated effectiveness is called the Q-value, which is stored and assigned to the state-action pair in the Q-table.

The method has strong limitations in terms of state space complexity, where the size of the Q-table can cause memory shortage problems. In such cases it is necessary to introduce the Deep Q-learning (DQN) [18] solution, during which the values of decisions are not determined in a tabular form, but are estimated using a neural network. To stabilize the training, we create a so-called Replay Buffer [19] that functions as the agent's memory. At each step, we save here the values of the current state, the action, the reward and the state we have entered. The model is trained by random sampling from the Replay Buffer.

As mentioned in standard Q-learning, when evaluating the effectiveness of a decision, we take into account the state we have reached. In the DQN algorithm, the same neural network estimates the efficiency of the current as well as the next state, which creates an unstable objective function during model training, which reduces convergence. To solve the problem, Haaselt, Guez, and Silver developed the Double DQN model [11] in which target values are fixed temporarily. It achieves this with a second neural network that is an exact copy of the primary one. The training of the secondary network is frozen for the entire length of the simulation, and the weights are synchronized with the primary neural network at given intervals.

There may be cases where for a condition all actions have the same Q-value, i.e., none of the actions gives a better state than the others, they have the same result. The Dueling DQN model [13] described by Wang et al. is able to filter out such conditions, in which case it is not necessary to learn the value of the actions belonging to them, consequently stabilizing the training process. It is especially relevant in environments where the action does not always affect it substantially. In our case, this is a cardinal issue, since in traffic free periods the model must acquire the knowledge that it is not necessary to change the signal phase either. The D3QN model is a combination of these three solutions.

B. Multi-Agent Reinforcement Learning (MARL)

According to the highest level grouping of the multi-agent literature, we distinguish three types: the fully cooperative, the fully competitive, and a mixed category. The solution described in the paper belongs to the cooperative MARL systems. The aim of the agents is to maximize or minimize some optional parameter of the road network. A fully cooperative solution assumes that the common goal means a single common reward rather than separate rewards for agents, however, in our previous research [20], we concluded that a local reward may be a more effective solution. This is presumably due to the fact that for a large number of agents, the amount of the reward depends more on the decision of the other agents rather than on the specified agent. The second basic classification is whether it is a centralized system in which a central agent, either hierarchically or standalone controls the entire system, or whether it is a decentralized system with several separate agents. In this case, there is a generalized form of MDP called stochastic game. In ReLight-WCTM, we train a multitude of peer-to-peer agents without central control.

As agents learn in a parallel way, their decisions influence the traffic density in the simulation. In practice, this process reduces the stationarity of the environment due to changes in the policies of other agents. Hence, agents need to discover information not only about their environment but also about other agents. However, too much information exploration, coupled with unpredictable operation due to random actions can destabilize the learning dynamics of other agents, thereby putting the learning agent himself in a more difficult situation. The problem can be avoided by making the policy of all other agents part of the state space, however, since the policy is represented in the form of a neural network, the size of its connection composition would make learning impossible. To increase the stationarity of the environment, agents share information with each other, which can be some kind of perception, representation, action, reward, and so on [21].

III. Simulation Environment and Model Parameters

We simulated the road network and traffic with the SUMO open source traffic simulation software, in which we created an exact copy of the main road in Pécs. In addition to a wide range of configuration options, SUMO also provides the ability to visually inspect through its GUI module. The system can be easily modified on a Python basis using the TraCI control interface. The road section under study contains nine intersections controlled by traffic lights, as well as several pedestrian crossings. Each intersection is unique in its kind. The number of roads forming the intersection as well as the number of lanes of roads, the size and shape of the intersection and the number of traffic lights placed in it also differ. This is a significant change compared to generally studied "grid" network type maps. One of the nine intersections is shown in Fig 1.

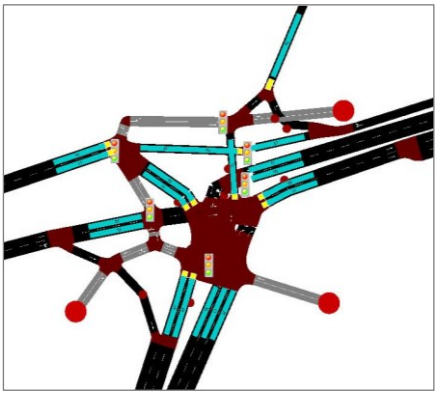

Fig. 1. The fifth intersection of the road network

On-site real-world measurements were used to simulate traffic. The incoming vehicle numbers in the given time unit were determined based on the daily rush hour. The starting points are located at the tops of the inward-facing streets at the edge of the map. In addition to the real map, this possibility further expands the reality of learning, which is a central element of the present research. The vehicle frequencies measured during rush hours represent the frequented and less used routes, so during influencing the traffic density we focused on maintaining the proportions. There are currently four traffic dynamics alternating randomly per episode (1600 steps) during the training:

- Rush hour with measured vehicle starting counts
- Average turnover reduced by 30%
- Low traffic reduced by 60%
- Extra low traffic reduced by 90% to explore the traffic-free periods

At the start of each episode, a pre-learning 180-second long traffic recovery phase begins. This ensures the agent not always start learning with an empty road network that could distort its effectiveness. The maximum permitted speed is uniformly 13.9 m/s (50 km/h) on each road section. The model checks every step for the presence of the gridlock phenomenon. In such a case, the traffic is congested to such an extent and in such a way that it also interfere the traffic of the crossing lane and blocks the entire intersection. Because nine agents are learning in parallel, the probability of the phenomenon occurring is very high. If we continue training, the traffic would remain the same regardless of the decision of the learning agent, so it would not receive adequate feedback in the form of a reward. The time to qualify as a gridlock is 200 seconds, measured with the vehicle waiting the longest. If the model detects a timeout, the agents are given a uniform penalty and then a new episode begins.

The processing of camera images provides the basis for determining the state space. Depending on the characteristics of the road network, we simulated camera images with different visibility, the length of which varies between 20 meters and 40 meters. Two types of data were extracted from the camera images: on one hand the number of vehicles seen on the camera and on the other hand their average speed. The state space also includes the current signal phase and the time spent in it in seconds. Our previous results [20] have shown that information sharing between agents effectively increases the discoverability of the simulation environment, therefore the state space was expanded with the sums of the number and average speed of vehicles, with the current signal phase, and the time spent in it in adjacent intersections

The frequency of action taking is 3 seconds, in which the agents can make two decisions: prolong the current signal phase or interrupt it. The phase plan provided by the Hungarian road operator, which the agent modifies includes a red-yellow signal in addition to the simple red, yellow, and green signals, as well as an intermediate time. The intermediate time is a full extent red phase, with the aim of avoiding possible collisions. Maximum phase time has not been introduced in the present research, which may be an advantage during off-peak periods. The reward function is a modified version of Weighted Completed Trip Maximization (WCTM) [22]. The original function calculates the number of vehicles leaving the simulation at the end of their route weighted by the length of the shortest route. To reduce the delay of the reward we counted leaving vehicles in the intersection instead of the end of the network [23].

TABLE I. PARAMETER SETTINGS USED DURING TRAINING

Parameter settings	
Parameter	*Value*
Starting value of epsilon greedy policy	1.0
Timesteps over which to anneal epsilon	10,000
Minimum epsilon value	0.02
Discount factor of Q-value update	0.9
Learning rate of the Neural Network	0.0001
Batch size	32
Target model weight updating frequency	500
Number of steps before starting learning	1000
Size of the Replay Buffer	50,000
Neural Network hidden layers	32, 128, 128

For training, we used the open source Python package called RLlib, which has high scalability and an extensive list of available reinforcement learning-based algorithms. The neural network behind the algorithms is free to build, supporting most frameworks. Their library called Ray Tune can be linked to RLlib, with which we performed hyperparameter optimization in a grid search manner. ReLight-WCTM's algorithm is D3QN with the hyperparameters in Table I..

IV. Simulation Results

ReLight-WCTM was trained in 600,000 steps, consisting of 375 episodes and immediate interruption in the case of a gridlock. The total learning time means 1,800,000 seconds of real traffic, which is nearly 21 days. The testing process consisted of three conditions, low, medium, and high traffic demand. In each condition, the model decided through 1000 steps, which means 5,000 real seconds, nearly an hour and a half for a given traffic dynamic. We compared the proposed algorithm and the baseline model by measuring CO_2 emission (g/s), average speed (m/s)(*AvSp*), number of halting vehicles (pcs)(*HaVe*) and waiting time (s)(*WaTi*).

TABLE II. SUMMARY OF METRIC AVERAGES

Type	Condition I.		Condition II.		Condition III.	
	Baseline	*ReLight-WCTM*	*Baseline*	*ReLight-WCTM*	*Baseline*	*ReLight-WCTM*
CO_2	147.949	142.517	250.154	241.038	422.776	366.954
HaVe	25.771	24.707	43.686	38.744	82.646	58.685
AvSp	12.981	12.995	12.755	12.792	12.35	12.516
WaTi	605.096	421.02	995.526	584.92	2016.40	956.658

As we can see in Table II., it was possible to improve on all four measured parameters during the three traffic dynamics presented. It can be stated the values improved the most when a large number of vehicles were traveling on the route. Although we did not optimize using the waiting time, we did see a drastic change in the metric. As can be seen from Table II. and Fig. 2., ReLight-WCTM achieved an improvement in degree of one traffic dynamic difference. Examining the visual interface, there was no visible change in its operation compared to the original setting. This dispels the concern that the model has found some anomaly that puts one group of vehicles at an advantage while another group at a disadvantage, thereby achieving good rewards during training. The agent was indeed able to achieve such a high degree of improvement by fine-tuning the signal phases.

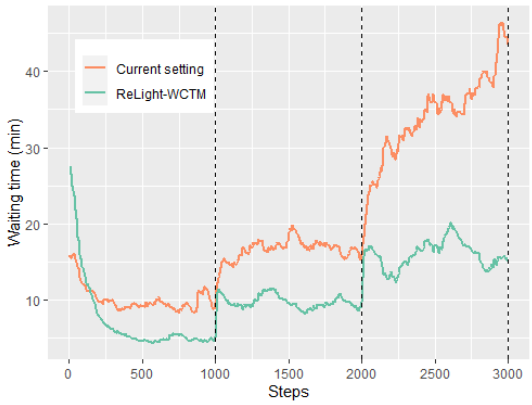

Fig. 2. Graphical comparison of aggregated waiting times

V. CONCLUSION

In the presented paper we simulated the real layout of the main road of Pécs, which contains nine intersections. In cooperation with the Hungarian road operator they provided us measured traffic data and the phase plan currently in use. Using the data provided, we trained agents based on a decentralized, information-sharing D3QN algorithm to manage the intersections. The results suggest that ReLight-WCTM has successfully mastered efficient control methods, is able to adapt to the current dynamics of traffic, and is able to greatly reduce CO_2 emission, waiting time, number of stops, and to increase average speed. By preserving verisimilitude, security risks have been reduced compared to commonly researched solutions. ReLight-WCTM can be integrated as a real traffic management tool with only a few modifications.

REFERENCES

[1] R. P. Roess, E. S. Prassas és W. R. McShane, Traffic engineering, Prentice Hall, 2004.

[2] P. Koonce, L. Rodegerdts, K. Lee, S. Quayle, S. Beaird, C. Braud, J. Bonneson, P. Tarnoff és T. Urbanik, „Traffic Signal Timing Manual," United States. Federal Highway Administration, 2008.

[3] P. Varaiya, „The max-pressure controller for arbitrary networks of signalized intersections," in *Advances in Dynamic Network Modeling in Complex Transportation Systems*, Springer, 2013, pp. 27-66.

[4] P. Lowrie, „Scats, sydney co-ordinated adaptive traffic system: A traffic responsive method of controlling urban traffic," *Roads and Traffic Authority NSW,* 1990.

[5] P. Hunt, D. Robertson, R. Bretherton és R. Winton., „SCOOT-a traffic responsive method of coordinating signals," *Transport Research Laboratory,* 1981.

[6] M. Abdoos, N. Mozayani és A. L. C. Bazzan, „Traffic light control in non-stationary environments based on multi agent q-learning," in *14th International IEEE Conference on Intelligent Transportation Systems*, 2011.

[7] S. Araghi, A. Khosravi, M. Johnstone és D. Creighton, „Q-learning method for controlling traffic signal phase time in a single intersection," in *16th International Conference on Intelligent Transportation Systems*, 2013.

[8] I. Arel, C. Liu, T. Urbanik és A. G. Kohls, „Reinforcement learning-based multi-agent system for network traffic signal control," *T Intelligent Transportation Systems,* %1. kötet4, %1. szám2, pp. 128-135, 2010.

[9] D. Zhao, Y. Dai és Z. Zhang, „Computational intelligence in urban traffic signal control: A survey," *Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews,* pp. 485-494, 2012.

[10] L. Li, Y. Lv és F. Y. Wang, „Traffic signal timing via deep reinforcement learning," *IEEE/CAA Journal of Automati,* %1. kötet3, %1. szám3, pp. 247-254, 2016.

[11] H. van Hasselt, A. Guez és D. Silver, „Deep reinforcement learning with double q-learning," in *Conference on Artificial Intelligence*, 2016.

[12] E. van der Pol, „Deep reinforcement learning for coordination in traffic light control," University of Amsterdam, 2016.

[13] Z. Wang, T. Schaul, M. Hessel, H. van Hasselt, M. Lanctot és N. de Freitas, „Dueling network architectures for deep reinforcement learning," arXiv, 2015.

[14] X. Liang és X. Du, „Deep Reinforcement Learning for Traffic Light Control in Vehicular Networks," arXiv, 2018.

[15] S. Wang, X. Xie, K. Huang, J. Zeng és Z. Cai, „Deep Reinforcement Learning-Based Traffic Signal," *Entropy,* 2019.

[16] C. Watkins, Learning from Delayed Rewards, Cambridge University, 1989.

[17] V. Mnih, A. P. Badia, M. Mirza, A. Graves, T. Harley, T. P. Lillicrap, D. Silver és K. Kavukcuoglu, „Asynchronous Methods for Deep Reinforcement Learning," 2016.

[18] V. Mnih, K. Kavukcuoglu, D. Silver, A. Graves, I. Antonoglou, D. Wierstra és M. Riedmiller, „Playing Atari with Deep Reinforcement Learning," 2013.

[19] L. Lin, „Reinforcement learning for robots using neural networks," *Computer Science,* 1992.

[20] P. Pálos, „Decentralized Multi-Agent Solutions in Traffic Management," Budapest University of Technology and Economics, 2020.

[21] L. Busoniu, R. Babuska és B. De Schutter, „A Comprehensive Survey of Multiagent Reinforcement Learning," *Transactions on Systems, Man, and Cybernetics,* p. 156–172., 2008.

[22] A. Hajbabaie és R. Benekohal, „Traffic Signal Timing Optimization Choosing the Objective Function," *Journal of the Transportation Research Board,* 2013.

[23] J. Stevens és C. Yeh, „Reinforcement Learning for Traffic Optimization," 2016.