PROCEEDINGS OF ELMAR-2020

62nd International Symposium ELMAR-2020

14-15 September 2020, Zadar, Croatia

EDITED BY

Mario Muštra
Josip Vuković
Branka Zovko-Cihlar

University of Zagreb
Faculty of Electrical Engineering and Computing
Department of Wireless Communications
Unska 3 / XII
10000 Zagreb
CROATIA

Published by:	Faculty of Electrical Engineering and Computing, University of Zagreb
Edited by:	Mario Muštra, Josip Vuković, Branka Zovko-Cihlar
Front Cover:	Painting by artist Mrs Ljerka Njerš
Printed by:	Ispis Ltd., Zagreb
Print ISBN:	978-1-7281-5972-0, CFP20825-PRT
XPLORE ISBN:	978-1-7281-5973-7, CFP20825-ART

Printed in Croatia.

62nd International Symposium ELMAR-2020 is organised by:

Croatian Society Electronics in Marine - ELMAR, Zadar

University of Zagreb Faculty of Electrical Engineering and Computing, Department of Wireless Communications

ELMAR-2020 symposium Sponsors

Ministry of the Sea, Transport and Infrastructure of the Republic of Croatia

Ministry of Science and Education of the Republic of Croatia

Croatian Regulatory Authority for Network Industries

HAKOM

Town of Zadar

County of Zadar

ELMAR-2020 symposium takes place under the co-sponsorship of:

IEEE Region 8
IEEE Croatia Section
IEEE Croatia Section Chapter of the Signal Processing Society
IEEE Croatia Section Chapter of the Antennas and Propagation Society
IEEE Croatia Section Chapter of the Microwave Theory and Techniques Society
Croatian Academy of Sciences and Arts
Croatian Academy of Engineering - HATZ
University of Zagreb
University of Zagreb Faculty of Electrical Engineering and Computing
University of Zagreb, Faculty of Transport and Traffic Sciences
University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology

ELMAR-2020 symposium is supported by:

ZADAR TOURIST BOARD
ERICSSON NIKOLA TESLA
OTP BANKA HRVATSKA
FALKENSTEINER BORIK ZADAR CROATIA

ELMAR-2020 SYMPOSIUM COMMITTEES

General Chair

Branka Zovko-Cihlar, University of Zagreb, Croatia

General Co-Chair

Mislav Grgić, University of Zagreb, Croatia

Program Chair

Mario Muštra, University of Zagreb, Croatia

ELMAR-2020 SYMPOSIUM INTERNATIONAL PROGRAM COMMITTEE

Juraj Bartolić, Croatia
Narcis Behlilović, Bosnia and Herzegovina
Aura Conci, Brasil
Marek Domanski, Poland
Janusz Filipiak, Poland
Borko Furht, USA
Mohammed Ghanbari, United Kingdom
Dušan Gleich, Slovenia
Mislav Grgić, Croatia
Sonja Grgić, Croatia
Ho Yo-Sung, Korea
Bernhard Hofmann-Wellenhof, Austria
Ebroul Izquierdo, United Kingdom
Ana Katalinić Mucalo, Croatia
Ismail Khalil, Austria
Tomislav Kos, Croatia
Igor Kuzle, Croatia
Panos Liatsis, United Kingdom
Lidija Mandić, Croatia
Branka Medved Rogina, Croatia
Borivoj Modlic, Croatia
Marta Mrak, United Kingdom
Mario Muštra, Croatia
Zdeněk Němec, Slovakia
Miloš Oravec, Slovakia
Jarmila Pavlovičova, Slovakia
Fernando Pereira, Portugal
Jan Pidanič, Slovakia
Peter Planinšič, Slovenia
Pavol Podhradský, Slovakia
Darko Ratkaj, Switzerland
Fabiana Rodrigues Leta, Brasil
Gregor Rozinaj, Slovakia
Markus Rupp, Austria
Shiguang Shan, China
Thomas Sikora, Germany
Karolj Skala, Croatia
Ryszard Stasinski, Poland
Luis Torres, Spain
Frantisek Vejrazka, Czech Republic
Dijana Vitas, Croatia
Stamatis Voliotis, Greece
Krzysztof Wajda, Poland
Branka Zovko-Cihlar, Croatia

ELMAR-2020 SYMPOSIUM
INTERNATIONAL REVIEW COMMITTEE

Goran Bakalar, Croatia
Sanja Bauk, Montenegro
Alen Begović, Bosnia and Herzegovina
Marko Bosiljevac, Croatia
Jelena Božek, Croatia
Emil Dumić, Croatia
Hrvoje Gold, Croatia
Sonja Grgić, Croatia
Edouard Ivanjko, Croatia
Branimir Ivšić, Croatia
Niko Jelušić, Croatia
Juraj Kačur, Slovakia
Ana Katalinić-Mucalo, Croatia
Serđo Kos, Croatia
Tomislav Kos, Croatia
Hrvoje Leventić, Croatia
Časlav Livada, Croatia
Sadko Mandžuka, Croatia
Marko Matulin, Croatia
Štefica Mrvelj, Croatia
Mario Muštra, Croatia
Miloš Oravec, Slovakia
Jarmila Pavlovičová, Slovakia
Juraj Petrović, Croatia
Jan Pidanič, Slovakia
Snježana Rimac-Drlje, Croatia
Renata Rybárová, Slovakia
Dorsaf Sebai, Tunisia
Chakib Taybi, Morocco
Filip Turčinović, Croatia
Mario Vranješ, Croatia
Josip Vuković, Croatia

ELMAR-2020 SYMPOSIUM
ORGANISING COMMITTEE

Mislav Grgić, Croatia
Mario Muštra, Croatia
Jelena Božek, Croatia
Filip Turčinović, Croatia
Josip Vuković, Croatia

CROATIAN SOCIETY ELECTRONICS IN MARINE - ELMAR

President: *Branka Zovko-Cihlar*

Vice-president: *Ivan Pupovac*

Secretary: *Vladimir Roboz*

Members of the ELMAR Executive Committee:

Winton Afrić, Irena Galić, Mislav Grgić, Sonja Grgić, Irena Jurdana, Branka Medved Rogina, Snježana Rimac-Drlje, Vladimir Roboz, Sonja Zentner Pilinsky

Members of the ELMAR Supervisory Committee:

Tomislav Kos, Boris Kviz, Lidija Mandić

Address: Faculty of Electrical Engineering and Computing
Croatian Society Electronics in Marine - ELMAR
Unska 3 / XII, 10000 Zagreb, CROATIA

Phone: + 385 1 6129 567

Fax: + 385 1 6129 717

E-mail: elmar2020@fer.hr

URL: www.elmar-zadar.org

State Complexity Reduction in Reinforcement Learning based Adaptive Traffic Signal Control

Mladen Miletić, Krešimir Kušić, Martin Gregurić, Edouard Ivanjko
Faculty of Transport and Traffic Sciences, University of Zagreb
Vukelićeva 4, Zagreb, Croatia
mladen.miletic@fpz.unizg.hr

***Abstract*—The throughput of a signalized intersection can be increased by appropriate adjustment of the signal program using Adaptive Traffic Signal Control (ATSC). One possible approach is to use Reinforcement Learning (RL). It enables model-free learning of the control law for the reduction of the negative impacts of traffic congestion. RL based ATSC achieves good results but requires many learning iterations to train optimal control policy due to high state-action complexity. In this paper, a novel approach for state complexity reduction in RL by using Self-Organizing Maps (SOM) is presented. With SOM, the convergence rate of RL and system stability in the later stages of learning is increased. The proposed approach is evaluated against the traditional RL approach that uses Q-Learning on a simulated isolated intersection calibrated according to realistic traffic data. Presented simulation results prove the effectiveness of the proposed approach regarding learning stability and traffic measures of effectiveness.**

***Keywords*—Intelligent Transportation Systems; Adaptive Traffic Signal Control; Reinforcement Learning; Self-Organizing Maps; Machine Learning**

I. Introduction

Today the most common method for traffic signal control of intersections in urban areas is the implementation of traffic light control systems with one or more signal programs that can either be fixed or adaptive. Fixed Traffic Signal Control (FTSC) programs are predetermined regardless of the current traffic state. Adaptive signal programs actively change according to the current traffic state and can increase the operational capacity of the controlled intersection. They can also include implementation of public transport and emergency vehicle priority. The wrong choice of signal programs can have a negative impact on traffic flow and cause significant delays in the urban transport network.

The latest approaches for Adaptive Traffic Signal Control (ATSC) include fuzzy logic based controllers, evolution algorithms, Reinforcement Learning (RL), multi-agent systems, and deep reinforcement learning algorithms [1]. Commercial systems such as Split Cycle Offset Optimisation Technique (SCOOT), Sydney Coordinated Adaptive Traffic System (SCATS), and Urban Traffic Optimisation by Integrated Automation (UTOPIA) provide satisfactory ATSC but remain heavily dependent on the defined environment model [2]. Some research on ATSC is directed towards the inclusion of Connected and Autonomous Vehicle (CAV) traffic, which could allow for the implementation of different ATSC approaches without traditional constraints [3].

RL approaches in the domain of machine learning show good results for single intersection ATSC due to its capability to learn on-line and dynamically adapt to the changes in the current traffic situation [4]. Another benefit of RL is its capability to continuously improve its performance without any prior knowledge of the environment [5]. The states in RL based ATSC are usually defined by the queue length of the waiting vehicles on all intersection approaches. From all RL techniques, the Q-Learning algorithm shows most promise when applied to intersection control problems [2]. With Q-Learning, it is important to capture the real traffic state accurately. This is usually done by increasing the number of state definition variables, which can, in turn, significantly increase the number of required learning iterations because the number of state-action pairs increases exponentially [6]. Various RL augmentation methods such as coarse coding and function approximation were used to cope with the high number of states and to reduce the learning time [7]. Another possible approach is to use Self-Organizing Maps (SOM) to represent state space while preserving the topology of the input data [8]. Similar methodology to the one presented in this paper was implemented in [9], but authors assumed that input data are gathered using Global Navigation Satellite System (GNSS) trackers in each vehicle, and it was implemented on a city wide scale with only slight changes to a two phase signal program, since they classified intersections, and not traffic situations. Hence, the contribution of this paper is applying the SOM-RL based ATSC methodology in a way that can be implemented with classical infrastructure sensors and adaptation of more complex signal programs.

This paper is organized as follows. The second section describes the fundamentals of ATSC. In the third section, the state complexity reduction in RL based ATSC is explained in detail. The simulation framework, model and scenarios are given in the fourth section. Obtained results and a discussion about them are presented in the fifth section. The paper ends with a conclusion and future work section.

II. Background of ATSC

In cases of daily recurring traffic congestion, the FTSC approaches are a simple and easy to implement tool for intersection control. They are usually set up as a timetable based control that uses predetermined signal programs in predefined time intervals during the day. Usually, there is a dedicated

978-1-7281-5973-7/20/$31.00

signal program applied during morning and afternoon peak hours. However, due to the stochastic nature of urban traffic, it is challenging to model optimal FTSC. The most commonly used upgrade to FTSC is actuated signal control. It consists of a predetermined signal program and a simple ruleset (IF-THEN reasoning), which allows the vehicles to send indirect requests to the controller, which in turn alters the current signal program [10].

More advanced ATSC systems are fully adaptive and allow the controller to dynamically adapt the signal program to real-time traffic conditions [11]. While the primary goal of ATSC is the maximization of intersection throughput, it can also be applied to allow Public Transport (PT) priority or emergency vehicle preemption [12]. Additionally, the goal of such control systems can be focused on the reduction of vehicle emissions and pollution [13]. Initially, for ATSC, it is required to gather and process relevant traffic data in real-time. Traditionally used inductive loops were a basis for ATSC systems such as SCOOT and SCATS. In modern days it has become much easier to collect real-time traffic data by using video cameras positioned at intersection approaches [14]. Additionally, the emergence of CAV could also serve for data gathering due to implemented onboard sensors and vehicle to infrastructure data exchange [3]. With more traffic data available, it is possible to develop more complex ATSC systems.

III. State complexity reduction in Q-Learning based ATSC

As mentioned, the goal of ATSC is to provide an optimal signal program depending on the current intersection state. To learn the optimal control policy with RL, the ATSC controller is modeled as a Markov Decision Process (MDP) represented by a $\langle S, A, T, R \rangle$ tuple. In it, S is the set of states, A is the set of actions, R is the reward function, and T represents the transition function [5]. The goal of RL is to find the best control policy for a given MDP, which is the policy that maximizes the sum of rewards over agent lifetime. RL controllers can use model-free or model-based approaches. This paper uses a model-free based RL approach known as Q-Learning.

A. Q-Learning

It is possible to estimate optimal action selection policy by using Q-Learning given a discrete set of states and actions [2]. Q-Learning estimates a selection policy by performing actions in an environment and by receiving a reward from the environment. The final result of Q-Learning is a mapping of state-action pairs that yield the maximum reward from the environment. In the traditional Q-Learning, the following learning rule equation is used:

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha(r_{t+1} + \gamma \max_{a' \in A} Q(s_{t+1}, a') - Q(s_t, a_t)), \quad (1)$$

where $Q(s_t, a_t)$ is the Q value for state s and action a, α is the learning rate at which the Q value is updated, r_{t+1} is the reward received from the environment after an action was selected, and γ is the discount factor used for reward transfer. By increasing the learning rate α, the Q-Learning algorithm can reach the optimal control policy sooner. However, it is then more susceptible to environment noise. Especially in a non-deterministic environment, such as in the case of intersection control. The parameter γ is used to influence the behavior of an agent to take into account possible future rewards that could lead to better overall performance. The reward r_{t+1} is received from the environment one time step after the selected action has been executed.

In ATSC, the reward function is usually modeled to give positive rewards when the overall traffic situation is improved. Commonly used reference variables to determine reward in ATSC are: queue lengths, cumulative delay, and intersection throughput [2]. Furthermore, the reward function can be modeled to contain multiple weighted variables to allow specific policies such as PT priority. When applying Q-Learning to non-discrete problems, the states are usually defined by splitting the continuous state space into several discrete segments. The main identified problem with this approach is that the number of state-action pairs grows exponentially with the number of dimensions of the state space leading to a slow convergence. In ATSC, the states are best defined by the queue lengths on all intersection approaches [2]. Control knowledge of the traditional Q-Learning is usually stored in a look-up table known as the Q-matrix. It has the number of rows equal to the number of defined states, and the number of columns equal to the number of defined actions a Q-Learning agent can take. While this approach is simple in implementation, the size of Q-matrix can become a computational problem when dealing with a high number of state-action pairs, such as in the case of ATSC.

B. Self-Organizing Maps

While the regular Q-Learning can provide good results for simpler problems in ATSC, it is often difficult to learn the optimal control policy because of the large number of possible state-action pairs. To alleviate this problem, SOM can be used to reduce the dimensionality of the state space, resulting in fewer state-action pairs and faster convergence [8]. SOM is a type of artificial neural network with only one computational layer and no output layer. The basic characteristic of SOM is the interconnection between individual neurons that are usually arranged in a two dimensional rectangular or hexagonal grid. SOM is primarily used as a data visualization technique since it is capable of presenting high dimensional data in only two dimensions. It is also used as an alternative to k-means clustering. SOM is trained in an unsupervised fashion by adapting neuron weights to more closely match the input signals. Upon receiving the input signal $X = (x_1, x_2, x_3, \cdots, x_n)$ the winning neuron is identified as the neuron which has the shortest Euclidean distance from the input signal. In literature, the winning neuron is sometimes called the Best Matching Unit (BMU). The BMU weights are then updated to more closely match the input signal according to the following equation:

$$W_i(k+1) = W_i(k) + \Theta\alpha_{SOM}\big[X(k) - W_i(k)\big], \quad (2)$$

where W_i is the weight vector of a neuron i, Θ is the neighborhood function, α_{SOM} is the learning rate, and k is the current learning step. The neighborhood function defines to what extent will the neurons connected to the BMU update their weights as shown with the following equation:

$$\Theta = \exp\Big(-\frac{d^2}{2R^2}\Big), \quad (3)$$

where d is the Euclidean distance of the neuron from the BMU, and R is the radius calibration parameter. In most cases, R is gradually reduced. That means that in the initial stages of learning, the BMU will pull neurons that are far away, while in the later stages, only the BMU will be updated. After multiple training iterations SOM will have neurons scattered in a way that generalizes the topology of the input data. With neurons scattered, the state space can be split into n discrete segments by mapping each possible state to its closest neuron. By doing this, the state space can be split into any number of segments to achieve the desired resolution of the state space. With the state-space split into discrete segments, it is possible to use the defined states as input for RL based ATSC.

C. Q-Learning with Self-Organizing Maps

To combine Q-Learning with SOM, two possible approaches can be used: on-line SOM training and off-line SOM training. In on-line SOM training, the weights of the SOM neurons are continuously updated during the Q-Learning process by performing perturbations of the neuron weights [8]. In the off-line SOM training, the weights of the SOM neurons are determined by learning from existing state input data, and the weights remain unchanged during the Q-Learning process. Additionally, the SOM can be applied to both state and action space if the action space is continuous [8]. In this paper, the off-line SOM training was used with SOM applied only on the state space. After the off-line training is complete, the Q-Learning agent learns the optimal control policy according to the usual Q-Learning algorithm with SOM serving only as the state interpreter, as shown in Fig. 1.

With the SOM based Q-Learning approach (SOM-QL), the number of states the agent can learn is the same as the number of neurons defined in the SOM network, regardless of the number of input parameters. This allows for significant state complexity reduction in high dimensional state definitions. Thus, the controller can be set to learn with the desired level of detail. This is useful in real-world problems such as ATSC because such systems are required to learn the optimal control policy in finite time.

IV. Simulation Framework

In this section, the simulation framework and defined testing scenarios for the analysis of the proposed ATSC approach are described.

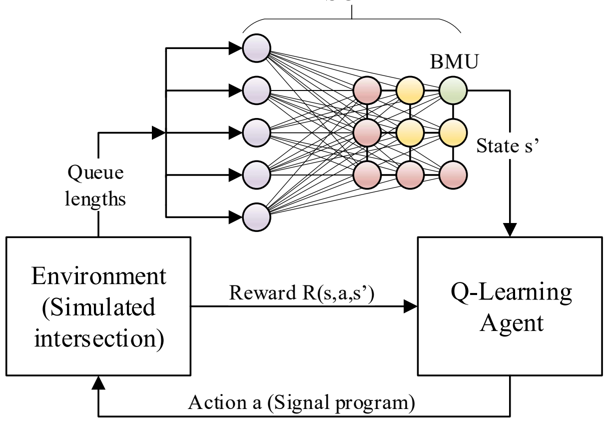

Figure 1. SOM-QL ATSC model

A. VISSIM and AForge Simulation Framework

The microscopic traffic simulator VISSIM was used for the design of an isolated signalized intersection and simulation of the behavior of traffic entities approaching the intersection [15]. It was used since it allows the simulation of individual traffic entities that results in a more realistic simulation of the traffic environment. By using the COM interface, VISSIM objects were manipulated from an external application written in C$^\sharp$ programming language. The modified AForge.NET [16] framework was used for Q-Learning and SOM implementation.

B. Simulation Model and Scenarios

The simulation model made in VISSIM was created using realistic traffic data from [17] and [18]. The modeled intersection (Fig. 2) is a part of a crucial connection to the center of Zagreb, the capital of Croatia. It is subject to congestion during morning and afternoon peak periods and is currently operated by a timetable based FTSC regime consisting of four different signal programs operating in two or three stages. The total simulation duration is 16.5 hours, and it models a typical workday period from 05:30 AM to 10:00 PM. For simplicity, the pedestrian traffic was excluded from the model, but all signal programs used followed required safety precautions regarding the minimum green light duration for pedestrians.

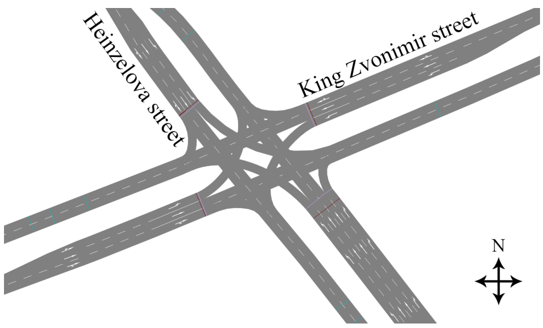

Figure 2. Simulated intersection modeled in VISSIM

The implemented ATSC approaches were analyzed in four distinct scenarios. The first scenario used the timetable based FTSC approach with four signal programs to serve as a baseline for comparison with ATSC enabled approaches. The second scenario used only traditional Q-Learning based ATSC with 9 input variables, which were the average queue lengths for each intersection movement in a 180 second interval. Each input variable was split into the three subsets (low, medium, high), resulting in a total of 19683 (3^9) defined states. The action set consisted of four distinct signal programs that were used in the FTSC approach. The third scenario used SOM-QL ATSC approach with the SOM network consisting of 19683 (81 x 243) neurons arranged in a two-dimensional grid. The fourth scenario used SOM-QL ATSC approach with the SOM network consisting of 81 (9 x 9) neurons arranged in a two-dimensional grid. Inputs for the SOM network in scenarios three and four were the same 9 intersection movements used in scenario two for state definition. The action set was the same as in scenario two.

In all ATSC based scenarios, the same Q-Learning parameters were used for the Q-Learning agent. Learning rate was fixed at $\alpha = 0.25$ and the discount factor was $\gamma = 0.8$. The exploration/exploitation trade-off was modeled by:

$$\epsilon = 0.95\epsilon_0^n + 0.05, \ \ \epsilon_0 \in \{0.98, 0.985, 0.99, 0.995\}, \tag{4}$$

where n is the current simulation number and ϵ_0 is a parameter used for testing different exploration/exploitation functions. The step time for Q function evaluation and selection of a new action was 180 seconds to give the controller enough time to affect the environment before receiving a reward. The reward function used was defined by:

$$r_{t+1} = d_t - d_{t+1}, \tag{5}$$

where d_t is the total delay of all vehicles in the time step before the selected action was executed, and d_{t+1} is the total delay of all vehicles in the time step after the action was taken.

In all SOM-QL ATSC approaches, the same parameters for SOM learning were used, with the same input data set, and were all selected according to multiple tested values. The learning rate of SOM α_{SOM} was modeled by the equation:

$$\alpha_{SOM}(Ep) = 0.9\frac{Ep_{max} - Ep}{Ep_{max}} + 0.1, \tag{6}$$

where Ep is the current epoch count. One epoch is equal to one full pass through the training set, and the total number of training epochs was $Ep_{max} = 1000$. The learning radius R is also decreased with the current epoch according to:

$$R(Ep) = 100 - \frac{Ep}{10}. \tag{7}$$

V. Simulation results

In this section, the proposed ATSC approach is evaluated using three different scenarios. The influence of the implemented approaches is analyzed concerning the obtained traffic parameters.

A. Obtained traffic parameters

The following Measures of Effectiveness (MoE) collected from VISSIM were used to analyze the impact of ATSC approaches TTT as the Total Travel Time of all vehicles in the network; LT_{avg} as the average Lost Time of all vehicles in the network; and NS_{tot} as the total Number of Stops of all vehicles in the network. Each analyzed scenario was simulated 2000 times, with 330 learning iterations per simulation. The average results of the last 200 simulations are presented in detail in Table I. Additionally, the standard deviation (σ) of the last 200 simulations is calculated to evaluate system stability. Figs. 3 and 4 show the weights of two sample neurons taken from SOM-QL (9 x 9) approach plotted topologically over the intersection map. This allows for easy visual analysis of the identified traffic states.

B. Discussion

Fig. 3 shows the sampled traffic state characteristic for early afternoon periods, with most vehicles traveling towards east. A problematic left turn is identified. Hence for this particular state, the signal program with prolonged left-turn green time should be used since other intersection approaches are not congested. Fig. 4 show the sampled traffic state characteristic for later morning periods with most vehicles traveling towards the west, and to a lesser extent, towards the east. In this state, the potentially problematic eastern approach could be handled with increased green time for the east-west approach.

Figs. 5 to 8 show the results of TTT for all analyzed scenarios and cases. In each one, the TTT was normalized according to the results of the FTSC approach. A moving average with a period of 100 was used to provide a comprehensive illustration of results. Fig. 5 shows the results for TTT with traditional Q-Learning ATSC approach. In the later learning stages, the TTT is reduced by 11.57 % when referenced to FTSC, the LT_{avg} was reduced by 18.17 % and NS_{tot} was reduced by 6.66 %. While the total number of vehicle stops was not significantly reduced, it is evident from the high reduction in LT_{avg} that the amount of time vehicles were queuing was greatly reduced. The impact of exploration/exploitation policy function is highly evident with the parameter $\epsilon_0 = 0.98$, since the Q-Learning agent is then forced to use exploitation in the earlier stages of learning, which leads to slower convergence. Fastest convergence was achieved in the case with $\epsilon_0 = 0.985$.

Fig. 6 shows the results for TTT with SOM-QL ATSC approach where SOM network consisted of 19683 (81 x 243) neurons. With the total number of neurons being the same as the number of states used in the traditional Q-Learning approach, it is expected to have similar results with the SOM-QL approach. The TTT was reduced by 11.31 % in the later learning stages, the LT_{avg} was reduced by 17.98 % and NS_{tot} by 5.40 % which is fairly close to the traditional Q-Learning approach. However, when compared, the results of the SOM-QL approach show a slower convergence rate than in the case with the traditional Q-Learning approach (Fig. 8). This is attributed to the fact that SOM neurons are positioned more densely in the more common states. By

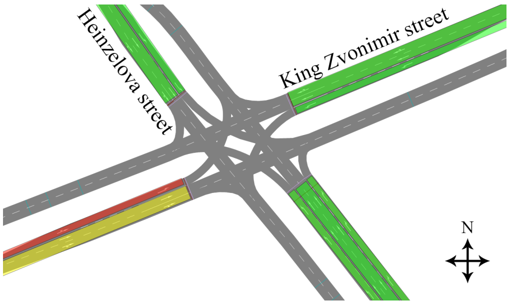

Figure 3. Topology plot of neuron 1 weights for SOM (9 x 9)

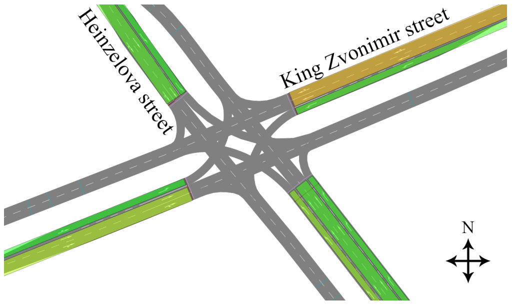

Figure 4. Topology plot of neuron 74 weights for SOM (9 x 9)

TABLE I. MEAN RESULTS FOR BEST CASE SCENARIOS FROM SIMULATION 1800 TO SIMULATION 2000

MoE	FTSC	Q-Learning $\epsilon_0 = 0.985$			SOM-QL 81x243 $\epsilon_0 = 0.980$			SOM-QL 9x9 $\epsilon_0 = 0.980$		
		Obtained	Change [%]	σ	Obtained	Change [%]	σ	Obtained	Change [%]	σ
TTT $[h]$	851.72	753.17	-11.57	14.99	755.41	-11.31	8.47	748.48	-12.12	9.13
LT_{avg} $[s]$	27.95	22.87	-18.17	0.83	22.93	-17.98	0.39	22.54	-19.34	0.40
NS_{tot}	56206	52462	$-$ 6.66	1389	53169	$-$ 5.40	1417	52432	$-$ 6.71	1440

having a finer state resolution in the most common states, the learning convergence is hindered because the Q-Learning agent is required to learn the control policy for multiple states that have similar behavior. In this case, convergence could be improved by either reducing the total number of neurons or by sharing knowledge between states, since it is expected that neurons that are connected to each other will be positioned relatively close to each other in the state space. While the convergence rate is slower, the system stability is improved with SOM-QL, as can be seen from the comparison of standard deviation in the last 200 simulations. This increase in stability is attributed to the fact that SOM neurons are positioned in a way that they more closely approximate the behavior of the intersection when compared to the traditional splitting of state space into subsets.

Fig. 7 shows the results for TTT with SOM-QL ATSC approach, where the SOM network consisted of 81 (9 x 9) neurons. Since the number of states is significantly reduced, a faster convergence rate in this scenario is expected. The TTT was reduced up to 12.12 %, which is the highest achieved result in all scenarios. The LT_{avg} was reduced up to 19.34 % and the NS_{tot} up to 6.71 %. From the results shown in Fig. 8, it can be seen that the convergence is significantly improved in comparison with the SOM-QL approach with 19683 neurons. The performance of the controller is increased even in the case with fewer states. The standard deviation in this scenario was marginally higher than in the previous SOM-QL ATSC approach, but still significantly lower than in the traditional Q-Learning ATSC approach.

VI. CONCLUSION AND FUTURE WORK

In this paper, the possibility of using SOM for state complexity reduction in Q-Learning based ATSC is analyzed and compared with traditional Q-Learning based ATSC using realistic traffic data. The results show that both traditional and SOM-QL ATSC approaches can reduce the TTT, LT_{avg} and NS_{tot} when compared to the FTSC approach. The traditional Q-Learning approach has a faster convergence rate when the number of neurons in SOM is the same as the number of states in the traditional Q-Learning. When the number of neurons in SOM is significantly reduced, the SOM-QL achieves faster convergence with better control stability, as shown by the lower σ values in the final stages of learning.

Obtained results have proven that the use of appropriately sized SOM can provide faster convergence, better control stability and improved analyzed MoEs in Q-Learning based ATSC. However, an open question regarding the optimal number of neurons in the SOM remains, and the computational cost of such systems remains. To answer this question, further work on this topic will include the analysis of different SOM sizes. Additionally, the use of Growing Neural Gas networks [19] as an upgrade to classical SOM for state complexity reduction without a predefined number of neurons in the network will also be examined.

ACKNOWLEDGEMENT

This work has been partly supported by the University of Zagreb and Faculty of Transport and Traffic Sciences under the grant "Innovative models and control strategies for intelligent mobility" and by the European Regional Development Fund under the grant KK.01.1.1.01.0009 (DATACROSS). Authors thank the companies "PTV Group" for providing a research license of the simulator VISSIM, and "Peek Promet" for providing signal program samples. This research has also been carried out within the activities of the Centre of Research Excellence for Data Science and Cooperative Systems supported by the Ministry of Science and Education of the Republic of Croatia.

REFERENCES

[1] M. Aslani, M. S. Mesgari, S. Seipel, and M. Wiering, "Developing adaptive traffic signal control by actor–critic and direct exploration methods," *Proceedings of the Institution of Civil Engineers - Transport*, vol. 172, no. 5, pp. 289–298, 2019.

[2] S. Touhbi, M. A. Babram, T. Nguyen-Huu, N. Marilleau, M. L. Hbid, C. Cambier, and S. Stinckwich, "Adaptive Traffic Signal Control: Exploring Reward Definition For Reinforcement Learning," *Procedia Computer Science*, vol. 109, pp. 513–520, Jan. 2017.

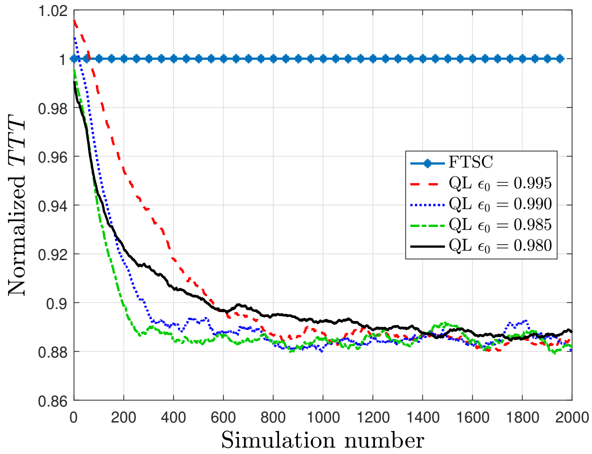

Figure 5. Normalized TTT for traditional Q-Learning scenario

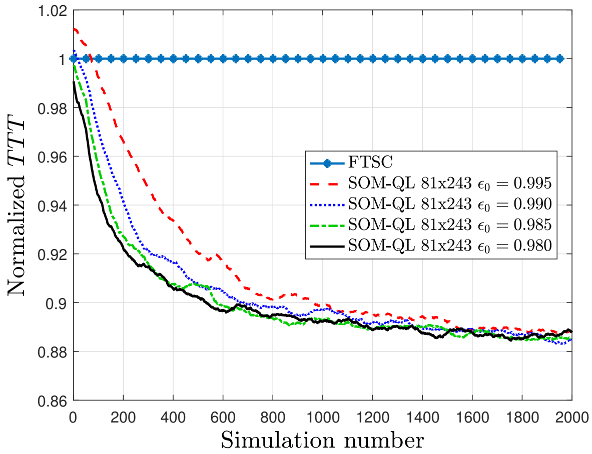

Figure 6. Normalized TTT for SOM-QL scenario with 81x243 neurons

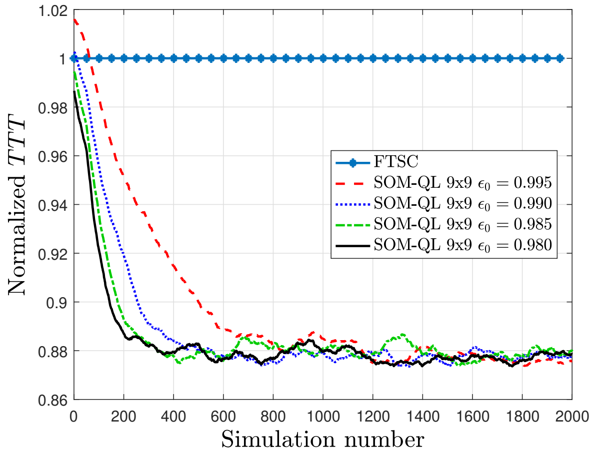

Figure 7. Normalized TTT for SOM-QL scenario with 9x9 neurons

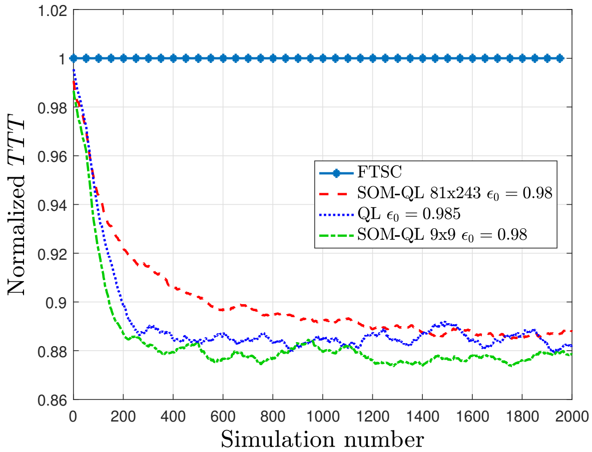

Figure 8. Normalized TTT for the best case of each scenario

[3] K. K. K. Chandan, A. J. M. Seco, and A. M. C. B. Silva, "A Real-time Traffic Signal Control Strategy Under Partially Connected Vehicle Environment," *Promet - Traffic&Transportation*, vol. 31, no. 1, 2019.

[4] B. Abdulhai, R. Pringle, and G. J. Karakoulas, "Reinforcement Learning for True Adaptive Traffic Signal Control," *Journal of Transportation Engineering*, vol. 129, no. 3, pp. 278–285, 2003.

[5] P. Mannion, J. Duggan, and E. Howley, "An Experimental Review of Reinforcement Learning Algorithms for Adaptive Traffic Signal Control," in *Autonomic Road Transport Support Systems*, Autonomic Systems, pp. 47–66, Cham: Springer International Publishing, 2016.

[6] X. Liang, X. Du, G. Wang, and Z. Han, "A Deep Reinforcement Learning Network for Traffic Light Cycle Control," *IEEE Transactions on Vehicular Technology*, vol. 68, pp. 1243–1253, Feb. 2019.

[7] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: A Bradford Book, 2018.

[8] C.-H. Uang, J.-W. Liou, and C.-Y. Liou, "Self-Organizing Reinforcement Learning Model," in *Intelligent Information and Database Systems* (J.-S. Pan, S.-M. Chen, and N. T. Nguyen, eds.), Lecture Notes in Computer Science, (Berlin, Heidelberg), Springer, 2012.

[9] Y.-C. Kao and C.-W. Wu, "A Self-Organizing Map-Based Adaptive Traffic Light Control System with Reinforcement Learning," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, pp. 2060–2064, Oct. 2018. ISSN: 1058-6393.

[10] K. L. H. Wang, Xiubin Bruce; Yin, "Vehicle actuated signal performance under general traffic at an isolated intersection," *Transportation Research Part C: Emerging Technologies*, vol. 95, 10 2018.

[11] M. J. Shirvani Shiri and H. R. Maleki, "Maximum green time settings for traffic-actuated signal control at isolated intersections using fuzzy logic," *International Journal of Fuzzy Systems*, vol. 19, Feb 2017.

[12] M. Miletić, B. Kapusta, and E. Ivanjko, "Comparison of two approaches for preemptive traffic light control," in *2018 International Symposium ELMAR*, pp. 57–62, Sep. 2018.

[13] Y. Zhang, X. Chen, X. Zhang, G. Song, Y. Hao, and L. Yu, "Assessing effect of traffic signal control strategies on vehicle emissions," *Journal of Transportation Systems Engineering and Information Technology*, vol. 9, no. 1, pp. 150 – 155, 2009.

[14] H. Wei, G. Zheng, V. Gayah, and Z. Li, "A Survey on Traffic Signal Control Methods," *arXiv:1904.08117 [cs, stat]*, Jan. 2020.

[15] PTV Group, "PTV Vissim simulation software." https://www.ptvgroup.com/en/solutions/products/ptv-vissim/. [accessed 12 February 2020].

[16] AForge, "AForge.NET Framework." http://www.aforgenet.com/framework/. [accessed 20 February 2020.].

[17] M. Vujić, *Dynamic priority systems for public transport in urban automatic traffic control*. PhD thesis, Faculty of Transport and Traffic Sciences, University of Zagreb, Croatia, December 2013. (In Croatian).

[18] M. Miletić, "Adaptive Control of Isolated Signalized Intersection using Neural Networks." Master thesis, Faculty of Transport and Traffic Sciences, University of Zagreb, Zagreb, Croatia, July 2019. (In Croatian).

[19] Y. Prudent and A. Ennaji, "An incremental growing neural gas learns topologies," in *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.*, vol. 2, pp. 1211–1216 vol. 2, 2005.