ASPAI
TENERIFE•2023

Advances in Signal Processing and Artificial Intelligence

Proceedings of the 5th International Conference on Advances in Signal Processing and Artificial Intelligence (ASPAI' 2023)

Edited by Sergey Y. Yurish

IFSA

Advances in Signal Processing and Artificial Intelligence:

Proceedings of the 5th International Conference on Advances in Signal Processing and Artificial Intelligence

7-9 June 2023
Tenerife (Canary Islands), Spain

Edited by Sergey Y. Yurish

Sergey Y. Yurish, *Editor*
Advances in Signal Processing and Artificial Intelligence
ASPAI' 2023 Conference Proceedings

E-mail (for orders and customer service enquires): ifsa.books@sensorsportal.com

Visit our Home Page on https://sensorsportal.com/ifsa_publishing.html

ASPAI Conference Website: https://aspai-conference.com/

ISSN: 2938-5350
ISBN: 978-84-09-48561-1
BN-20230602-XX
BIC: UYQ

(086)

Reliable Learning-based Controllers and How Structured Simulation is a Path towards Them

K. Kušić [1], R. Schumann [2], M. Gregurić [1], E. Ivanjko [1] and M. Šoštarić [1]
[1] University of Zagreb Faculty of Transport and Traffic Sciences,
Department of Intelligent Transportation Systems, Zagreb, Croatia
[2] HES-SO Valais-Wallis, University of Applied Sciences Western Switzerland – Valais, Switzerland
kresimir.kusic@fpz.unizg.hr

Summary: New approaches to control stochastic non-linear time-variant processes include the application of machine learning techniques. One of the problems with learning-based controllers is their reliability in a wide area of process parameters as the controller is trained using a limited set of representative scenarios, either chosen by the designer or taken from historic records. Thus, reliable controller behavior can be guaranteed only in scenarios applied during controller training. Due to the very larger number of random variables and possible scenarios, not all variations can be applied in the controller training process using simulators to guarantee good controller behavior when applied in a real system. One case is traffic control (signal programs, variable speed limit, ramp metering) having large travel patterns variety. The concept of Structured Simulations Framework (SSF) can cover most probable learning scenarios. Thus, applying SSF enables a systematic controller training approach by complementing existing scenarios with synthesized ones that evoke or replicate substantial aspects of real traffic. Such training is necessary to ensure reliable learning-based controllers. This paper discusses the concept of applying SSF to ensure the reliability of learning-based controllers and proposes the application in traffic control for the case of variable speed limits on motorways.

Keywords: Learning-based controller, Controller reliability, Structured simulation, Variable speed limit.

1. Introduction

Today, there is a need to control stochastic non-linear time-variant processes. Classical feedback controllers designed using process modeling and linearization in characteristic working points cannot cope with such processes, especially regarding a wide area of process parameters. Different controller parameters are needed for particular working points. To overcome this problem, machine learning is applied and learning-based controllers can be designed [1]. The advantage of this approach is that the control law is learned as a mapping between the input measurements and control output for a wide area of different process parameters fulfilling a defined criteria function. The drawback is that a good control output can be ensured only for scenarios presented to the controller during the training phase. In the case of large processes with many random variables and possible scenarios, the generation of learning scenarios can very fast become unfeasible if one wants to guarantee good control output in every possible situation.

Traffic control is a good example of such a problematic process. Namely, today's traffic control centers manage larger networks of motorways or connected signalized intersections [2]. Change of traffic flows is under the influence of daily human behavior demanding repeating mobility patterns related to working days and weekends or holidays, the state of the transport infrastructure, and the weather. Thus, significant stochastic non-linear time variant behavior is present in the controlled process (signal programs for intersections, and variable speed limit or ramp metering rate on motorways) creating the need for an appropriate large set of structured learning scenarios [3]. Such a set of learning scenarios has to cover most often traffic behavior and behaviors that appear not often but are related to some potential regular events (bad weather influence, traffic incidents, holidays, etc.). A good starting point for the creation of such learning scenarios sets can be obtained by analyzing collected real world measurements as shown in [4].

To ensure that the controller learns how to resolve most of the possible scenarios, which can occur in real world situations, in feasible time, a structured approach of presenting the learning scenarios to the controller during the training process is needed. This is opposite to current training approaches where only a few representative learning scenarios are used and evaluated afterwards [5]. With a structured approach to creating training scenarios, it can be ensured that the controller can successfully resolve a wider area of scenarios that can appear in the real world increasing the reliability of such a learning-based controller when applied in real world applications. This describes the aim of this paper to define such a structured framework to increase the reliability of the learned controller with the use case of traffic control i.e., Variable Speed Limit (VSL) on motorways taken as a use case.

This paper is organized as follows. The second section elaborates on the structured simulation concept. The third section explains how learning-based controllers can be designed including open problems. The fourth section describes the possible application of

the structured simulation concept for learning-based traffic controllers. The continuing fifth section concludes the paper.

2. Structured Simulation Concept

The main goal of the structured simulation approach is to automate and systematize a search process about the system's behavior by means of simulation or in a wider sense experimentation. It is assumed that system's behavior for whatever reasons cannot be described as an equation-based system. However, it is possible to analyze the behavior of the system in a controlled environment, e.g., in form of controlled experiments or computational simulations. Without loss of generality, we will address in the following that simulations are performed. For practical considerations, the total number of simulations needed to perform is relevant, as each simulation run requires resources, e.g., like computing time. Therefore, the idea of structured simulation is to maximize potential leanings about a system's behavior with a given number of simulations runs. We assume that the system's behavior can be computed as part of the simulation, and depends primarily on the input parameters of the simulation. Thus, each simulation run can provide an insight about the system. This knowledge discovery process happens in form of a guided search process, in which different states of the parameter space of the simulation system can be evaluated. The main idea of the structured search approach is that this search process needs to be a guided / informed search, taking advantage of potential domain knowledge to reduce the search time, i.e., a number of states in the parameter-space that needs to be investigated compared to an uninformed search in which the points in the state space would be visited in a purely randomized way, as it is the case in a Monte-Carlo like approach.

The main ideas of structured simulations have been implemented in the Structured Simulation Framework (SSF)[1]. The overall process is structured within three stages:

- Scenario generation in which the necessary states to be visited are computed;
- Simulation execution in which for each state the corresponding simulation is performed;
- Result handling in which results are collected and potentially analyzed.

This is visualized in Fig. 1. These stages can be executed partially overlapping. However, for the sake of simplicity we assume no parallel execution options, but a strict linear flow.

2.1. Scenario Generation

For the organization of the search process that needs to be performed, the scenario generation phase is most critical, and therefore will be detailed in the following. Depending on which phase of the controller design needs to be addressed, different search objectives, and therefore search strategies, need to be specified. The general idea that states, i.e., a particular combination of input parameters, need to be visited in a particular order to increase available information about the behavior of the system under investigation.

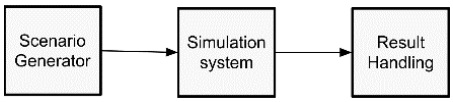

Fig. 1. Schematic structure of the Structured Simulation Framework modules [6].

For performing a state-based search approach state transitioning rules needs to be defined. These rules specify how the next state to visit can be derived. These state-transition rules are specified in form of Modifiers, which implement a particular type of state transition function. A set of modifiers can be provided, and given an initial state of the search, a number of states to evaluate can be derived. This allows for an ordering, and therefore enumerating the states to visit, which is a pre-requested for the structured search. The principle of this is shown in the following Fig. 2.

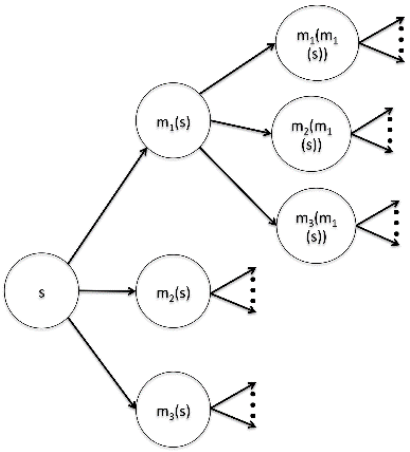

Fig. 2. Schematic representation of the state space enumeration.

The principles of structured simulation can be used in two different ways during the design phase of a learning-based controller. On the one hand, it can be used during the training phase of the controller, as it can be used to generate training data that can cover a wider range of the parameter-space, e.g., by implementing a novelty search strategy [7]. On the other hand, the framework can also be used in the validation phase of the controller, providing a systematic search for states in which the controller might not perform well, or to the contrary, provides empirical evidence that the controller is capable of

[1] http://silab.hevs.ch/structSim/structsim.html

handling various situations, it is likely to be confronted with during its operations [6]. So far, the SSF framework has been used with two different simulation environments, to ensure its general structure. A link to the commercial traffic simulator tool VISSIM, which is used by traffic researchers to validate traffic management approaches, taking advantage of learning-based traffic controllers is presented in [8]. The second simulator was a Game of life simulation [9]. In both cases, the evaluation has been done for each state in isolation. This can be considered as an alternative to the commonly used Monte-Carlo simulation approach. However, the foundation of SSF in VSL motorway analysis has not yet been explored in detail and a general concept needs to be established before the implementation of SSF in the simulations training process of the learning-based VSL controller can start.

3. Challenges in Design of Learning-based Controllers

The most used approach for learning-based controllers is the one which is based on the Reinforcement Learning (RL) approach. RL approaches can be divided into model-based and model-free methods. The model-free methods are currently most investigated since they do not require existing learning datasets. They are used in an online fashion for solving optimal control problems stated as Markov Decision Processes (MDPs). Most of the current control problems can be modeled by using the MDP approach since it enables modeling control tasks in discrete time. Thus, at each time step, the controller receives feedback from the controlled system in the form of a state signal and takes an action in response. As a result, the RL approach accounts for a change in the state signal that could lead to a change in the optimal control action. Thus, they can handle nonlinear and stochastic dynamics and nonquadratic reward functions [10]. Those features are ideal for controlling systems which are in their nature stochastic such as traffic flows. The common approach in designing the RL controllers is in the simulation loop. The system is modeled within a simulation environment and the RL controller performs learning in it until the end results converge to satisfactory values [11]. The core problem in their design is to generate enough representative states in the controlled system for desirable learning convergence of controller parameters. Particularly, this problem is related to the state-action exploitation-exploration ratio which governs when it is needed to stop the process of learning and continue just to use the learned control policies.

3.1. Latest Methodologies in Learning-based Control Design and Challenges

The model-free RL controllers can be divided into two wide categories with respect to design approaches. The first of them is value-based. They are based upon temporal difference learning in which the learn value function is computed. Typical representatives are Temporal Difference (TD), State-Action-Reward-State-Action (SARSA), and Q-learning (QL) algorithms. The second RL type of model-free controllers is related to directly learning an optimal policy or trying to approximate the optimal control policy if the true optimal policy is not attainable. The REINFORCE algorithm is the most prominent representative of that category. The policy-based controllers tend to directly optimize the control policy, which is the core goal of good control. Therefore, they are more stable and less prone to failure compared to value-based counterparts [5]. The value-based methods such as QL are less stable and suffer from poor convergence since they learn an action value function approximation usually called Q-values. Those Q-values are in their vanilla version stored in tabular format which is then used to find a corresponding policy. The advantage of value-based methods is their off-policy nature. Thus, in their operation work, they are much more sample efficient compared to policy-based methods since they exploit data from control knowledge repositories based on stored state-action function.

3.2. Open Problems

The convergence speed is problematic even for the most advanced learning-based controllers. It heavily depends on tuning the controller's hyperparameters such as learning rates, discount factor in QL, regularization parameters of ANN models, etc. Furthermore, the learning convergence depends on the complexity, differentiation rate, and persistence of learning scenarios which can be understood as outliers. Those scenarios can significantly reduce the learning convergence or make convergence unstable [12]. This is especially the case with RL approaches which are based on Q-function approximation using ANN models. Thus, the coverage of sufficiently different learning scenarios and avoiding too extreme cases have the potential to stabilize learning convergence. Regarding these mentioned issues, it is needed to develop a framework that has the ability to generate scenarios that maintain increased convergence within a simulation environment to improve learning convergence. Moreover, it is imperative to assess the reliability of all possible scenarios. Thus, it is needed to avoid learning from scenarios that are not possible to happen outside the simulation environment. It is imperative to create constraints that prevent generating such scenarios. Those learning scenarios can have a negative effect on learning convergence as well.

However, several works applied RL techniques for VSL control on motorways. For example, in [3, 13], nonlinear function approximation techniques in RL were applied to improve control of an underlying nonlinear and nonstationary traffic flow on a motorway using RL based VSL. In [3], research on the

importance of state description and simulation scenario generation on the learning process was presented. Particularly, the control policy of RL based VSL was further improved by enriching the agent's state variables with predictive information about the expected traffic (speeds and densities) of the controlled motorway segment by running parallel simulations.

4. Structured Simulations for Learning-based Traffic Controllers

In this section, we discuss the application of principles of structured simulation for the design of learning-based traffic controllers using the example of a VSL controller. Optimizing VSL on motorways is about choosing the right speed limits for the current traffic flow conditions [14]. In the event of a traffic jam, for example, the goal of the VSL system is to slow down and harmonize the traffic arriving into the downstream active bottleneck in order to relieve the congested sections and prevent further capacity drop and, if possible, re-stabilize the traffic flow (Fig. 3).

4.1. Motorway Traffic Process Characteristics

Traffic flow on a motorway can be described by the fundamental diagram relating three basic traffic parameters flow, speed, and density. Two main traffic characteristics can be distinguished: free-flow (stable) and congested (unstable) traffic flow (Fig. 6). Free flow condition is described by higher travel speeds and less dense traffic flow i.e., a lower number of vehicles per motorway segment. As the traffic volume increases, the traffic reaches the so-called critical point where the capacity is exceeded. From this point on, the traffic flow becomes unstable and significant interactions between vehicles occur characterized by higher traffic density and lower speed resulting also with lower flow rate. These three dependent variables are, thus, used either as a measure of state representation or to define objective functions for evaluating learning-based VSL controller behavior.

4.2. Learning-based VSL

Numerous approaches have been used for VSL control [15, 16]. However, nowadays, learning-based control techniques have shown great potential to improve VSL. Among them, the QL algorithm is widely studied for VSL control optimization. QL-based VSL (QL-VSL) controller (agent) perceives and interacts with its environment (motorway section) at each control time step by performing actions (speed limits) and receiving feedback signals called rewards (see Fig. 4). Thus, the QL-VSL agent learns to associate an action a with the expected long-term discounted rewards R for performing that particular action in a particular state s and by following an optimal policy π. Thus, the action-value function (1) expresses how good action is to be applied to a particular state of the environment for transition in the next improved state [17]. The QL algorithm can be expressed as:

$$q(s,a)_\pi = E[R_{t+1} + \gamma R_{t+2} + \\ +\gamma^2 R_{t+3} \cdots | S_t = s, A_t = a, A_{t+1:\infty} \sim \pi], \quad (1)$$

where $q(s,a)$ represents an action-value function. R represents rewards that an agent received where discount factor γ controls the importance of future rewards. An action a is an executed action from an available set of actions A_t, s is a possible state of the controlled process from a set of states S_t, and optimal policy π denotes the mapping function from states to optimal actions.

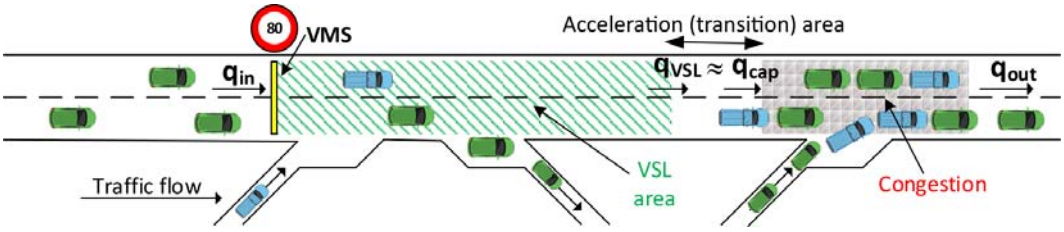

Fig. 3. Application of VSL for bottleneck control [15].

Therefore, the state variables that the VSL agent can use must be carefully selected to provide much information about the dynamics of the traffic flow, but uniquely. For example, densities uniquely define the traffic state on a motorway, but traffic volume does not (Fig. 6). The similar applies to the reward function when RL is used for modeling of a VSL controller [3], [5]. Fig. 4 visualizes the RL-VSL control process which can improve control policy during the operation. At every time step, the RL-VSL agent senses the world (traffic flow parameters on the controlled motorway section) and takes actions (speed limits) in it, and receives rewards or punishments based on the consequences of the taken actions and accordingly adjust the learning model in order to achieve better reward score.

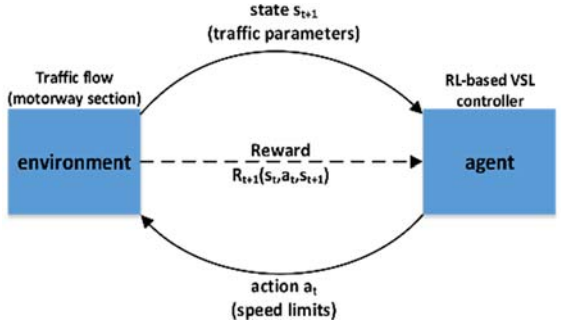

Fig. 4. Reinforcement learning framework for VSL control [15].

4.3. Structured Simulation-based VSL Controller Design

The important prerequisite for applying RL-VSL is appropriate training and evaluation processes in a traffic simulator. Thus, the quality of the learned control law strongly depends on the generated training dataset, which must provide relevant traffic scenarios for the motorway area where RL-VSL is to be used. For that, SSF can be used to create an appropriate training dataset complementing existing traffic scenarios with synthesized ones that evoke or replicate substantial aspects of real traffic.

1) Strategy for the training data generation: The overall training data has to be generated in a way that on the one hand, it provides enough data points so that the learning controller can actually generalize its behavior based on the training sample. This requires on the one hand a sufficient detailing of the parameter space in which the controller has to operate most of the time, and where the performance is critical, while on the other hand, the controller must have seen during its training phase a sufficient number of cases, which only have a low probability to occur in reality, to ensure that has obtained sufficient knowledge to react at least operative, even though not optimal in such situations. The search strategy therefore most balance between exploitation elements, here it focuses on certain areas (Fig. 5), and exploration elements in which it covers the typically larger part of the space, which will have a significantly lower probability of occurrence. In the following, we outline the strategy for computing a training set that aims to balance these two aspects. However, this approach requires an a-priori data analysis, e.g., of historic data, to provide insights into what parameter combinations can be considered typical, and to what probability they occur. For this, a theoretical model is proposed that relies on so-called modifiers that systematically change the states of the running simulation to reproduce the desired dataset for the training process. For simplicity, we allow the state domain contains two regions of interest, with each region assigned a modifier m1 and m2 computed by (2) and (3), respectively.

$$m1 = (max([e1(n)], [p1(n)]))k, \quad (2)$$

$$m2 = = (\max([e2(M - n)], [p2(M - n)]))k, \quad (3)$$

Precisely, m1 is referred to as the modifier indicating the sufficient number of data points needed from the exploitation region [b, c] (Fig. 5), while m2 defines points in the exploration region [a, b) and (c, d]. The parameter δ defines the step size of the search strategy. Accordingly, $n = (c - d)/\delta$ and $M = (d - a)/\delta$ represent the number of discrete states within the exploitation and exploration regions, respectively. Similarly, the probabilities p1 and p2 define the probability of the occurrence of a state within the mentioned regions. The numbers e1 and e2 define the designed efficiency of the learning controller within the exploitation and exploration regions. If the controller has to deal with a large dataset (m1+m2 too large), one can use the scaling factor k = (0, 1]. By scaling the training dataset, the initial ratio between m1 and m2 remains the same, so there is no bias.

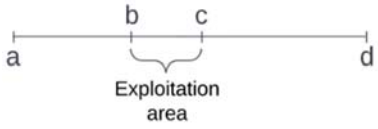

Fig. 5. State space domain.

For example, the low-frequency states (fewer probable events) are evident from the fundamental diagram in Fig. 6 (synthetic traffic data obtained from a microscopic road traffic simulator [18]). If we look at the x-axis (densities uniquely define traffic condition), we can roughly relate some density values to the boundaries (a, b, c, d) in the state space domain (Fig. 5).

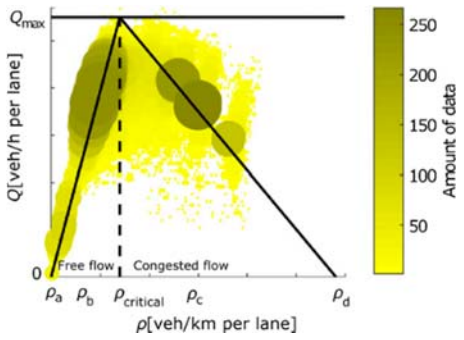

Fig. 6. Fundamental traffic flow diagram [18].

In reality, the fundamental diagram could have a different distribution of data, e.g., the most frequent data could be measurements that occurred during the night, which is characterized by low traffic intensity (longer period, more measurements received), while rush hours are less present. Thus, data density and the period during which the data are retrieved are correlated.

2) Technical integration of the SSF: Functional integration in the block diagram shown in Fig. 7 for

automatic configuration of control strategies learned by RL-VSL consists of the SSF framework and the microscopic traffic simulator Simulation of Urban MObility (SUMO) [19].

At the heart of SSF is a scenario generator that is used to systematically change states to ensure a ratio of exploitation/exploration states, e.g. how much m1 and m2 data to present in the training process of RL-VSL. The output of the simulator is used to evaluate the controller itself; accordingly, some strategies where poor performances are detected need to be re-trained more frequently. For example, each simulation took T hours to cover all relevant scenarios. Of course, since RL-VSL does not require only one simulation for training, multiple runs are required. This is where the simulation controller comes in, whose job is to automate the training process; e.g. start the traffic simulation, stop the running simulation, insert a new value of the modifier into the running simulation, and finally collect the results of the running simulation for performance analysis and strategy evaluation. Therefore, there is in general a glue code between scenario generator, controller, and simulator which wiring blocks into SSF framework. Thus, in our case, we need to ensure communication between the SSF traffic scenario generator (that computes modifiers) and SUMO simulator together with the RL-VSL controller into one loop so that such a client-server architecture enables direct modification of simulated scenario and results analytics needed to compute modifiers for structure simulation scenario modification.

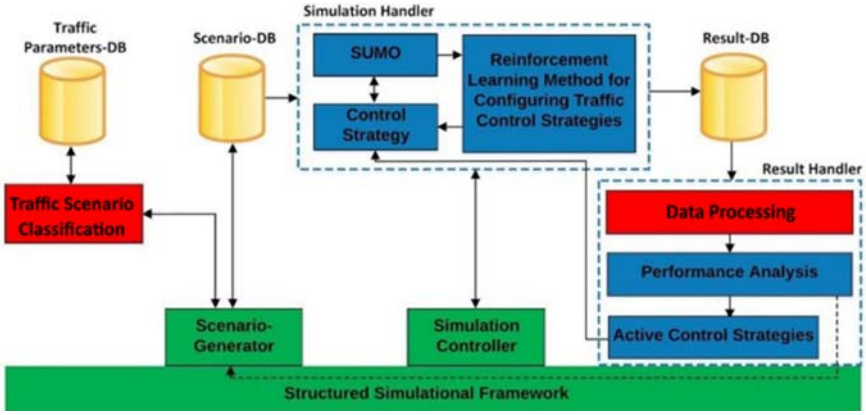

Fig. 7. Block scheme for automated configuration of control strategies learned by VSL learning controller.

5. Conclusion and Future Work

This paper presents a conceptual framework to obtain a representative dataset for training a learning-based VSL controller through the concept of applying SSF. Thus, the functional integration, i.e. closing the loop between SSF, RL-VSL controller, and simulator SUMO is explained. In general, the appropriate datasets used for training of RL-VSL should be non-biased, i.e., the data most likely to occur on the real motorway should be in a certain proportion to those less likely to occur. In this way, the required number of data points is minimized, so that the simulations and training require less computing power, but still ensure a reliable behavior of the controller in possible traffic cases, i.e., a satisfactory generalization. For this, we presented the detailed model within SSF used to calculate the state modifiers to systematically change/generate the states in the running simulation scenario in SUMO. Although the presented work is a rather conceptual study, it is based on some assumptions that should be addressed in future work. The theoretical model for calculating the modifiers, i.e., how often and in what direction the state space should be searched is presented in this paper. An important factor is the mapping from the fundamental diagram to the state space domain used in computing the modifiers. We have demonstrated the concept of mapping based on specific domain knowledge of the controlled process, i.e., from the fundamental traffic diagram by partitioning regions of the traffic densities according to traffic conditions: free flow, critical flow, and congested flow. However, this is not sufficient because it does not take into account the density of data (the occurrence of data), which, especially on motorways, depends on the time period in which the data are collected. Therefore, further analysis is needed to develop a suitable method for mapping the real traffic state space to the state space domain used in the SSF logic along with the simulation evaluation.

Acknowledgements

This work has been partly supported by the Croatian Science Foundation under the project IP-2020-02-5042, and by the European Regional Development Fund under the grant KK.01.1.1.01.0009 (DATACROSS). The author and Ph.D. student Krešimir Kušić received the 2021-2022 Swiss Government Excellence Scholarship to visit HES-SO Valais-Wallis, Switzerland. This research has 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. Yu, S. Chai, A survey of direct learning control, in *Proceedings of the Chinese Control Conference (CCC'19)*, Guangzhou, China, 2019, pp. 2536-2541.

[2]. J. A. Calvo, I. Dusparic, Heterogeneous multi-agent deep reinforcement learning for traffic lights control, in *Proceedings of the 26th Irish Conference on Artificial Intelligence and Cognitive Science (AICS'18)*, 2018.

[3]. E. Walraven, M. T. Spaan, B. Bakker, Traffic flow optimization: A reinforcement learning approach, *Engineering Applications of Artificial Intelligence*, Vol. 52, 2016, pp. 203-212.

[4]. F. Vrbanić, M. Miletić, E. Ivanjko, Z. Majstorović, Creating representative urban motorway traffic scenarios: Initial observations, in *Proceedings of the 63rd International Symposium ELMAR-2021*, 2021, pp. 183-188.

[5]. M. Gregurić, K. Kušić, E. Ivanjko, Impact of deep reinforcement learning on variable speed limit strategies in connected vehicles environments, *Engineering Applications of Artificial Intelligence*, Vol. 112, 104850, 2022.

[6]. R. Schumann, C. Taramarcaz, Towards systematic testing of complex interacting systems, *CEUR Workshop Proceedings*, Vol. 2397, 2019, pp. 55-63.

[7]. M. Boussaa, O. Barais, G. Sunyé, B. Baudry, A novelty search approach for automatic test data generation, in *Proceedings of the* IEEE/ACM *8th International Workshop on Search-Based Software Testing*, 2015, pp. 40-43.

[8]. M. Gregurić, E. Ivanjko, R. Schumann, C. Taramarcaz, Structured simulation: A framework for the automated analysis of adaptive systems, in *Proceedings of the TUD 1102 COST ARTS Final Conference*, Bordeaux, France, 2015.

[9]. D. Kraft, Game of life simulation, Bachelor Thesis, *University of Applied Sciences*, Western Switzerland, 2017.

[10]. L. Busoniu, T. de Bruin, D. Tolić, J. Kober, I. Palunko, Reinforcement learning for control: Performance, stability, and deep approximators, *Annual Reviews in Control*, Vol. 46, 2018, pp. 8-28.

[11]. M. Gregurić, M. Vujić, C. Alexopoulos, M. Miletić, Application of deep reinforcement learning in traffic signal control: An overview and impact of open traffic data, *Applied Sciences*, Vol. 10, Issue 11, 2020, 4011.

[12]. A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, Y. LeCun, The loss surfaces of multilayer networks, in *Proceedings of the 18th International Conference on Artificial Intelligence and Statistics (AISTATS'15)*, 2015, pp. 192-204.

[13]. E. Vinitsky, K. Parvate, A. Kreidieh, C. Wu, A. Bayen, Lagrangian control through Deep-RL: Applications to bottleneck decongestion, in *Proceedings of the 21st International Conference on Intelligent Transportation Systems (ITSC'18)*, 2018, pp. 759-765.

[14]. G. Iordanidou, C. Roncoli, I. Papamichail, M. Papageorgiou, Feedback-based mainstream traffic flow control for multiple bottlenecks on motorways, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, Issue 2, 2015, pp. 610-621.

[15]. K. Kušić, E. Ivanjko, M. Gregurić, M. Miletić, An overview of reinforcement learning methods for variable speed limit control, *Applied Sciences*, Vol. 10, Issue 14, 2020, 4917.

[16]. E. R. Müller, R. C. Carlson, W. Kraus, M. Papageorgiou, Microsimulation analysis of practical aspects of traffic control with variable speed limits, *IEEE Transactions on Intelligent Transportation Systems*, Vol. 16, Issue 1, 2015, pp. 512-523.

[17]. R. S. Sutton, A. G. Barto, Reinforcement Learning: An Introduction, *The MIT Press*, 1998.

[18]. E. Ivanjko, K. Kušic, M. Greguric, Simulation analysis of two controllers for variable speed limit control, *Proceedings of the Institution of Civil Engineers – Transport*, Vol. 175, Issue 7, 2022, pp. 413-425.

[19]. D. Krajzewicz, J. Erdmann, M. Behrisch, L. Bieker, Recent development and applications of SUMO – Simulation of Urban MObility, *International Journal on Advances in Systems and Measurements*, Vol. 5, 2012, pp. 128-138.