Expanded Applicability: Multi-Agent Reinforcement Learning-Based Traffic Signal Control in a Variable-Sized Environment

István Gellért Knáb[1][a], Bálint Pelenczei[1][b], Bálint Kővári[2,3][c],
Tamás Bécsi[2][d] and László Palkovics[1,4][e]

[1] *Systems and Control Laboratory, HUN-REN Institute for Computer Science and Control (SZTAKI), Budapest, Hungary*
[2] *Department of Control for Transportation and Vehicle Systems, Faculty of Transportation Engineering and Vehicle Engineering, Budapest University of Technology and Economics, Budapest, Hungary*
[3] *Asura Technologies Ltd., Budapest, Hungary*
[4] *Széchenyi István University, Győr, Hungary*
{*knab.istvan.gellert, pelenczei.balint*}*@sztaki.hun-ren.hu,* {*kovari.balint, becsi.tamas*}*@kjk.bme.hu,*

Keywords: Machine Learning, Reinforcement Learning, Deep Learning, Traffic Signal Control, Intelligent Transportation Systems.

Abstract: During the development of modern cities, there is a strong demand articulated for the sustainability of progress. Since transportation is one of the main contributors to greenhouse gas emissions, the modernization and efficiency of transportation are key issues in the development of livable cities. Increasing the number of lanes does not always provide a solution and often is not feasible for various reasons. In such cases, Intelligent Transportation Systems are applied primarily in urban environments, mostly in the form of Traffic Signal Control. The majority of modern cities already employ adaptive traffic signals, but these largely utilize rule-based algorithms. Due to the stochastic nature of traffic, there arises a demand for cognitive decision-making that enables event-driven characteristics with the assistance of machine learning algorithms. While there are existing solutions utilizing Reinforcement Learning to address the problem, further advancements can be achieved in various areas. This paper presents a solution that not only reduces emissions and enhances network throughput but also ensures universal applicability regardless of network size, owing to individually tailored state representation and rewards.

1 INTRODUCTION

One of the paramount contemporary challenges pertains to air quality, predominantly in densely populated regions, notably urban centers (Fenger, 1999). In order to foster the development of livable cities, it is imperative to implement various measures across multiple sectors.

As depicted in Figure 1, transportation significantly contributes to greenhouse gas emissions (Ritchie, 2020). From this observation, it is evident that substantial global progress can be achieved through the implemented interventions. The efforts touch on numerous technical areas such as vehicle propulsion and fuel issues where the direction of electricity is significant (Ritchie, 2024), but in addition,

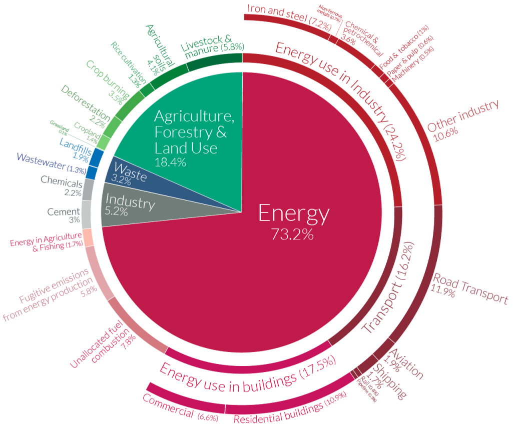

Figure 1: Emissions by sector (Ritchie, 2020).

[a] https://orcid.org/0009-0007-6906-3308
[b] https://orcid.org/0000-0001-9194-8574
[c] https://orcid.org/0000-0003-2178-2921
[d] https://orcid.org/0000-0002-1487-9672
[e] https://orcid.org/0000-0001-5872-7008

Knáb, I., Pelenczei, B., Kővári, B., Bécsi, T. and Palkovics, L.
Expanded Applicability: Multi-Agent Reinforcement Learning-Based Traffic Signal Control in a Variable-Sized Environment.
DOI: 10.5220/0012920800003822

In *Proceedings of the 21st International Conference on Informatics in Control, Automation and Robotics (ICINCO 2024) - Volume 2*, pages 15-25
ISBN: 978-989-758-717-7; ISSN: 2184-2809

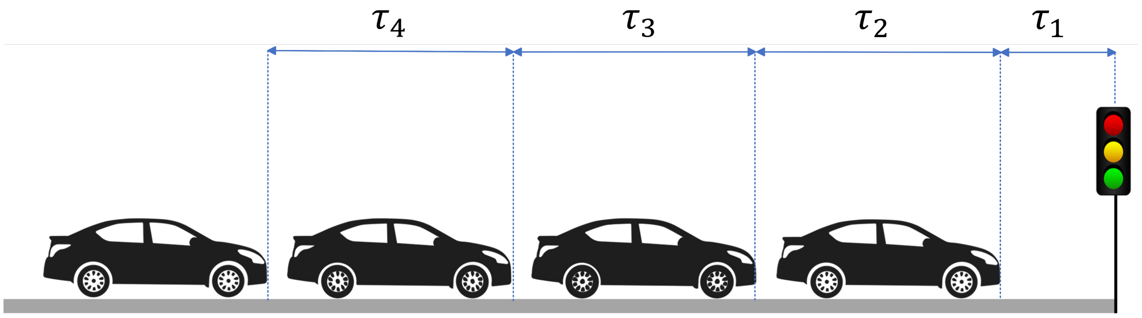

Figure 2: Illustration of reaction times in response to traffic signal changes.

infrastructure development can also lead to considerable improvements. These advancements allow for a favorable modulation of emission trends even under conditions of increasing vehicular traffic. Additionally, they can reduce both the duration of traffic exposure and the incidence of unnecessary delays (Singh and Gupta, 2015).

Given that the spatial expansion of road networks is often unfeasible (Zhang et al., 2011) or only justified during peak periods of congestion, Intelligent Transportation Systems (ITS) (Dimitrakopoulos and Demestichas, 2010) are increasingly emphasized. In urban environments, this approach achieves better utilization of the road network with minimal infrastructure changes, leading to a reduction in the aforementioned indicators.

The main guideline in traffic control is to avoid transients. The reason for this is that the dynamics of individual vehicles act as constraints on the movement of others as well. The event-driven approach of this is the ITS, which implements changes and introduces constraints based on the current traffic conditions. The primary objective is to mitigate the necessary alterations occurring within traffic, with two principal manifestations: human reaction time (Kesting and Treiber, 2008) and losses derived from the inertia of vehicles. The former presents itself as a cumulative issue, where the reaction time of vehicles in the queue behind one another can be seen in Figure 2 and delineated as follows:

$$\tau = \sum_{i=1}^{n} \tau_i, \qquad (1)$$

where τ_i denotes the reaction time of individual vehicles and n denotes the length of the queue.

Alongside reactions, inertia-derived losses suggest that in the event of anomalies, there will inevitably be a dissipation of varying degrees, manifesting as a moving jam. From this, it follows that truly efficient traffic management encompasses not only the handling of existing issues but also their prevention, which is crucial. As a result of this objective and its cognitive nature, there is a growing emphasis on the research domain concerning the application of Machine Learning (ML) and Deep Learning (DL) methodologies for the management of diverse traffic scenarios. Among these, particularly noteworthy due to their interactivity are multi-agent systems, where the goal is not merely to optimize the state of a single intersection but to seek an optimal solution at the network level.

The demand for optimal decision-making at the network level is observable in numerous cases. The increasing significance of ITS is observable both on highways and in city environments. In urban areas, one of the core implementations of ITS is Traffic Signal Control (TSC) (Qureshi and Abdullah, 2013) (Rotake and Karmore, 2012), while for highways, Variable Speed Limit Control (VSLC) (Khondaker and Kattan, 2015) (Kővári et al., 2024) often provides a versatile solution. The former encompasses demand-based switching of traffic lights, where predetermined phases can be overridden based on real-time traffic data if necessary. With this demand-based approach, the extent of idle green phases can be reduced while enhancing throughput capacity.

Behind the method, there is a repository of numerous algorithms. The applied algorithms can be divided into two main groups. Currently, rule-based solutions such as SCATS (Sydney Coordinated Adaptive Traffic System) (Kustija et al., 2023), GreenWave, and RHODES (Real-time Hierarchical Optimized Distributed Effective System) (Mirchandani and Wang, 2005) primarily dominate existing traffic networks.

The models produced during the tuning of rule-based systems serve as aids for solving traffic problems. Among these issues is the problem that the adaptability of algorithms is limited, which is a primary consideration in urban traffic management, given that the distribution of traffic network load is not uniform over time. Consequently, it is evident in the case of traffic issues that their resolution requires a predictive nature.

Predictive decision-making can be approached from multiple angles. The two primary methods are classical control theory solutions with identified systems, and implementations based on data driven solutions. While Model Predictive Control(Ye et al., 2019) is a highly popular solution among classical methods, reinforcement learning is frequently employed in the field of soft computing for solving such types of problems. The advantage of the latter is that it eliminates the need to determine approximate physical equations, which in many cases are incapable of adequately approximating reality.

Reinforcement learning (RL) has been employed in this field for several years, with numerous solutions emerging recently to address the problem (Wei et al., 2021) (Abdulhai et al., 2003). Several studies focus on the appropriate selection of different RL abstractions, as alongside the learning parameters, their proper formulation enables successful training. From articles (Wei et al., 2018) (Wiering et al., 2000) with the definition of rewards, it is evident that primarily macroscopic parameters such as speed and waiting time are enumerated. The Deep Q Network (DQN) algorithm used in the research is not unknown in the world of TSC or in multi-agent systems (Kolat et al., 2023), but its application still holds new possibilities. Deep learning-based systems offer numerous advantages over current implementations operating on networks, as demonstrated by the reviewed literature. However, in addition to online decision-making, it is also worth discussing the offline training process.

Decision-making systems founded on machine learning frequently exhibit superior performance compared to those reliant on physical models; however, several critical considerations must be addressed in this context. Among various factors, the computational costs associated with the development of contemporary models have significantly escalated, as evidenced by the data presented in Figure 3.

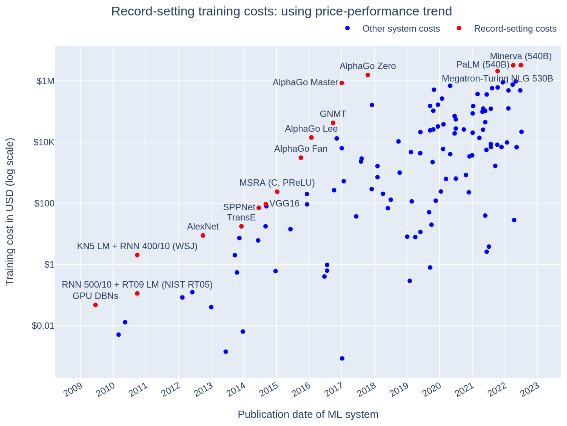

Figure 3: The change in costs over time required to train ML models (Cottier, 2023).

To address this issue, several efforts exist, such as optimizing the reward function using methods like Monte Carlo Tree Search (Kővári et al., 2022). Another approach in multi-agent systems involves formulating the state representation and the interventions of individual agents in such a way that they are reusable.

This research does not focus on the formulation of individual abstractions as its novelty; instead, the contribution lies in the use of an agent developed during the training process. By appropriately segmenting the environment, a large task can be divided into many subtasks, where each entity serves the same objective.

The study aims to demonstrate that although the performance offered by the algorithm depends on the number of intervention points, it is capable of showing improved operation in every case, even though training was conducted in only a single environment and the model derived from it is applied to networks of arbitrary size. Limiting the number of training sessions to just one environment can be a step towards sustainability, as it not only reduces emissions in the traffic network but also decreases the resources required for training.

2 METHODOLOGY

With the rising interest in artificial intelligence and the expansion of its application areas, a new industrial revolution defines the research fields of the 21st century (Ross and Maynard, 2021). AI is not only favored in the field of engineering, but its presence has also become significant in the healthcare (Reddy et al., 2020), financial (Cao, 2020), and entertainment industries over the past few years. Machine Learning represents a distinct subgroup within the domain of artificial intelligence (Nadarajan and Sulaiman, 2021). It is characterized by its capacity to execute a multitude of tasks, surpassing algorithms that rely on physical models, owing to its inherent cognitive capabilities. Within this domain, three distinct groups can be identified: supervised, unsupervised, and reinforcement learning. Although primarily the first two types are used in the automotive industry, the suitability of reinforcement learning for sequential decision-making elevates it to the forefront of modern research areas. Moreover, an attractive feature of this approach is that the data collection phase, which often represents a significant financial resource, is not part of the workflow here. Instead, the data is generated during the learning process based on the agent's own experiences.

The real breakthrough in reinforcement learning was observed when it was combined with Deep Learning (DL). In 2015, for the first time, a machine defeated a human in the highly complex game of Go. The algorithm named AlphaGo, built using neural networks and Monte Carlo Tree Search (MCTS), prevailed over the reigning world champion (Wang et al., 2016). The paper describing DQN (Mnih et al., 2015), which also appeared in the same year, shows that the desired cognitive decision-making capabilities were able to surpass human abilities in several other games as well.

2.1 Reinforcement Learning

The fundamental concept of reinforcement, which is distinct from the other two branches of machine learning, is based on the communication between two classes (Sutton et al., 1999)(Sutton and Barto, 2018). These two objects are an environment and an agent, where the agent's task is to learn a decision-making strategy that allows it to make optimal decisions over time. This is implemented through an iterative learning process that involves storing individual experiences and evaluating the success of specific decisions based on the resulting state. The numerical representation of the quality of a given decision is the reward, which the agent aims to maximize.

As shown in Figure 4, the agent receives the next state and the reward from the environment in response to a decision made in the current state. The encapsulation of these in a mathematical framework is the Markov Decision Process, which describes state transitions using these abstractions along with the transition probabilities. For most RL algorithms, during each step, the resulting data is stored in a buffer of a predetermined size in the following format:

$$Transition_t = (S_t, A_t, R_t, S_{t+1}, Done), \quad (2)$$

from which the individual elements can be seen in Figure 4, and *Done* indicates whether a training episode has ended.

As previously described, the goal of the agent is not only to make a correct decision at a given point but also to learn an optimal sequence of decisions. A cumulative reward (Szepesvári, 2022) is calculated for this, which includes the rewards attainable by subsequent actions. However, it is important to note that state transitions closer to the given state influence the cumulative reward value to a greater extent, and therefore should have a larger weight. To achieve this, the γ discount factor was introduced, which allows the formalism to be expressed in the following way:

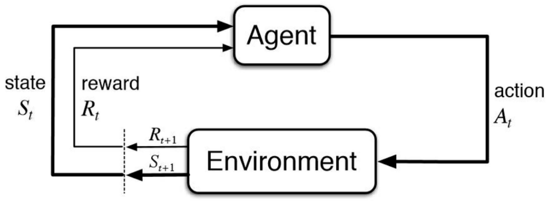

Figure 4: Reinforcement Learning (Sutton and Barto, 2018).

$$G_t = \sum_{t=0}^{T} \gamma^t \cdot r_t, \quad (3)$$

this prioritizes the higher rewards associated with state transitions that are closer over those that are further away. By introducing the cumulative reward, sequences can be generated showing the order of actions and the resulting states, to which various quality attributes can be assigned. From these, a new quality attribute can be introduced, the Q value, which is not just a sum but these values are assigned to specific state transitions, thus enabling the learning of agents. The value assigned to these state-action pairs can be calculated using the Bellman equation, of which most general form is the following:

$$q_\pi(s,a) = \mathbb{E}_\pi[G_t \mid S_t = s, A_t = a] \quad (4)$$

where $q_\pi(s,a)$ is the expected return following policy π. During training, the more training samples the system processes, the more information the algorithm can use to approximate the appropriate Q value. As a result, during training, the expected behavior is an increase in Q values in accordance with the reward trend, converging to the value that defines the boundary of the environment.

However, alongside the observed quality attributes during training, there arises a need to understand how decision-making is realized in the initial state where there is a lack of information. Since this is about experience-based decision-making, it is necessary to mention one of the main issues in reinforcement learning, the exploration-exploitation dilemma. An agent can make decisions in two ways: either by taking a random exploratory step to learn more about the environment, or by taking the best action according to its current knowledge. It is evident from the task formulation that over time, maximum rewards are achieved by fully informed decisions. However, without sufficient experience, even informed decisions will not be optimal. A commonly used method to resolve this is the ε-greedy method, which initially allows for complete exploration and gradually transitions to making fully informed decisions over time.

2.2 Deep Q Network

Reinforcement learning uses numerous algorithms. These algorithms can be categorized into two primary groups: value-based and policy-based. Additionally, there exists a hybrid category that integrates both approaches, known as actor-critic algorithms. Many of these systems use neural networks in their operation, sometimes even multiple networks.

The DQN belongs to the first group, where the model approximates optimal behavior using a value function. It consists of two value networks: one that updates at each iteration, and another network that only copies the weights of the first network after a certain number of steps. The Equation 4 for the DQN algorithm can be written as follows:

$$Q(s,a) = Q(s,a) + \alpha(r + \gamma max_{a'} Q(s,a') - Q(s,a)) \tag{5}$$

where the role of the secondary network is to extract the maximum available Q value, since a constantly changing network is not suitable for making decisions based on the experiences of previous steps, as small changes in weights could easily disrupt this value. The algorithm tunes the networks on a randomly sampled data of a predetermined size. This tuning is based on gradient descent, using the mean squared error calculated between predicted-target and the Q values provided by the network.

Although it is often appropriate to use actor-critic algorithms such as PPO or TD3 for many tasks, due to the discrete output requirement in this case, DQN was chosen, as traffic lights require binary output.

2.3 Multi Agent Reinforcement Learning

For problems that can be well-separated into subtasks, a favored approach in RL is Multi-Agent Reinforcement Learning (MARL). In this case, multiple agents share an environment (Buşoniu et al., 2010), as shown in Figure 6. Its major advantage is that complex systems composed of many small tasks can be divided into much more manageable sub-tasks, making the overall desired behavior easier to achieve.

The grouping of agents can be done based on two main principles. In terms of their structure, agents can be identical, referred to as homogeneous agents, or different, known as heterogeneous agents (Abed-Alguni et al., 2014). Additionally, based on their behavior, they can be categorized as cooperative, competitive, or independent. While in the first two cases it is clear that the goal is either to enforce common interests or individual interests, in the case of independent learners, they try to optimize their own decision-making strategies without interacting with each other. Additionally, they do not attempt to hinder each other as seen in competitive decision-making.

Among the fundamental abstractions, the reward strategy is also worth considering in multi-agent systems. Depending on the nature of the task, it is worth employing entirely different approaches for each type of agent. While in cooperative systems the network often receives a common reward or punishment, moving towards network-level optimization, which is referred to as identical payoff (Nowé et al., 2012), in competitive agent scenarios, distributed rewards are a well-functioning concept.

3 ENVIRONMENT

The environment consists of two important entities. For simulating traffic networks and assessing their load, a simulator is necessary to provide the relevant data for evaluation. In addition, there arises the demand for an environment with which agents can communicate to effect changes and alter their own states.

3.1 Simulator

The previously mentioned requirement is fulfilled by SUMO(Simulation of Urban MObility) (Behrisch et al., 2011), which is an open-source simulator. When selecting it, an important consideration was its good scalability and the ease with which modifications can be made across a wide range, including traffic generation and network sizingwindow. Additionally, it provides numerous macroscopic characteristics such as speeds, waiting times, or even emission indicators, which can be retrieved per vehicle or even per lane. Emissions are not measured using sensors; instead, SUMO employs models for emission approximation calculations. Specifically, the HBEFA v2.1

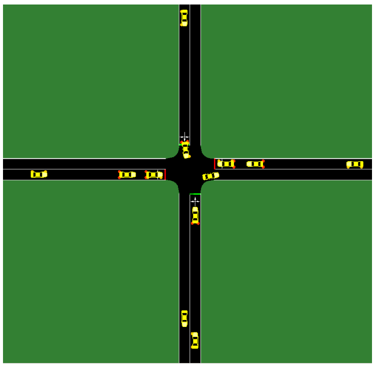

Figure 5: A SUMO intersection.

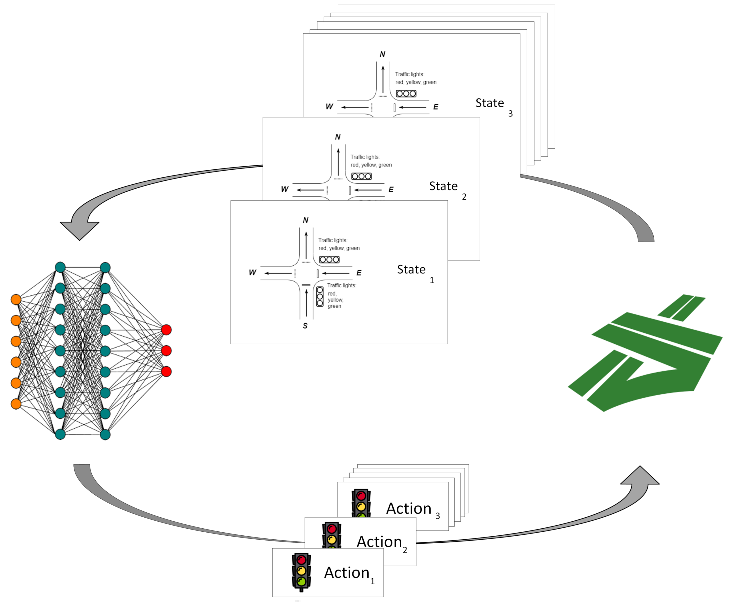

Figure 6: Multi-agent reinforcement learning in multi-intersection environments.

(Krajzewicz et al., 2014) model is utilized, which operates in the following manner:

$$E = c_0 + c_1 va + c_2 va^2 + c_3 v + c_4 v^2 + c_5 v^3,$$

denoting velocity with v, acceleration with a, and the constant related to the given emissions with c. The control is carried out with standard three-phase traffic lights, including yellow as a transient, where either the north-south direction switches to green, or the east-west direction. Movement in all directions is allowed from every lane as shown in Figure 5. During the training, a network consisting of four intersections was used, while during the evaluation, one, two and three intersection network was included.

Traffic generation is a crucial aspect of training. In all four network sizes, the load was simulated in such a way as to justify the use of TSC. In addition to the RL implementation, the built-in functions of SUMO, namely the adaptive mode and delay-based implementation, are also examined to determine how effectively they can handle saturation in given networks.

3.2 Communication Framework

In the context of RL, the gym architecture is frequently employed for environment simulations. Its standardized structure offers significant advantages, as the algorithm can be tested for correct operation on numerous pre-prepared environments before being applied to real-world problems. Its fundamental functions include step, reset, and render. In this case, the first two are essential, as visual representation is the responsibility of SUMO, which is enabled during configuration. The data extracted from the simulator are processed here for the algorithm, and this is also where Python communicates the completed interventions back to the simulator.

The environment includes the formulation of individual RL abstractions. Among these, the state is most often described by macroscopic characteristics in traffic cases. In this case, the average speeds defined on the lanes are critical values, as higher throughput is reflected in their higher values. Since the algorithm implements the multi-agent nature in a homogeneous manner, a description is necessary where each intersection receives an image of the surrounding roads. Based on these, each intersection can be described with the following state representation:

$$state_i = \begin{bmatrix} velocity_1 \\ velocity_2 \\ velocity_3 \\ velocity_4 \end{bmatrix}, \qquad (6)$$

where the individual average speeds are values measured on the incoming lanes of the intersections. This description offers a significant advantage in that intersections of identical characteristics can be described arbitrarily using the same representation, thereby facilitating the requirement articulated in the contribution for applicability across any network with just a single trained agent.

The action space has also been divided, with each intersection independently controlling either the vertical or horizontal traffic lights to turn green, in the following manner:

$$action = \begin{bmatrix} North - South \\ East - West \end{bmatrix} \quad (7)$$

The formulation of the reward strategy was conceived along analogous lines to that of the state representation. Average speeds are also visible within this level, but waiting time appears alongside it as a penalized phenomenon. The reward equation can be described as follows:

$$R = \frac{v_{avg}}{1+w}, \quad (8)$$

where v_{avg} represents the average speed defined across the network, while w denotes the total waiting time accumulated across the network at a given step. This is necessary because, although the individual agents learn based on the independent learner analogy, the goal is to master a network-level optimal decision-making strategy.

The formulation of individual abstractions facilitates the implementation of multi-agent characteristics with an arbitrary number of agents. As can be observed, thanks to effective segmentation, the independent learner concept can be applied to individual agents, and due to their identical structure, every agent can be represented by a single neural network. Despite operating on the analogy of an independent learner, the agents aim to search for network-level optimization. This is achieved in two ways. Firstly, the states overlap with each other because they are positioned next to one another. Additionally, the identical payoff between individual decisions creates a connection. As seen in Figure 6, each agent makes a decision and receives a state in return. However, since there is a common interest in rewards, the training samples at any given step will always contain the same reward across all intersections. Thus, within a single time step, a number of training samples equal to the number of intersections are introduced into the system. By studying the convergence curve after the learning process, the evaluation phase assesses the results of the training performed on the four-intersection network.

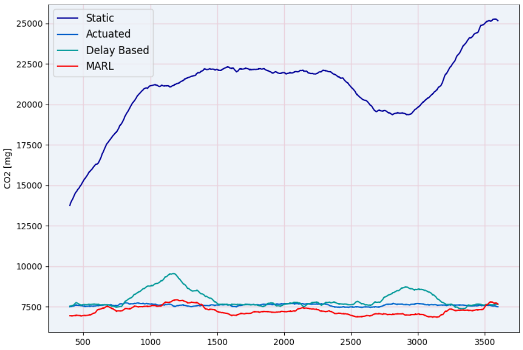

Figure 7: CO_2 Emission in a 1 intersection network.

4 RESULTS

After a successful training process, the evaluation takes place in the manner outlined in the introduction, in terms of sustainability metrics and throughput. While CO_2 and NO_x are examined in terms of emissions, the number of vehicles forced to stop and the time these vehicles spend waiting will form the basis for comparison in terms of throughput. As articulated in the contribution to the research, the training is conducted only on the largest network, but the evaluation contains 1, 2, and 3 intersection networks, thus examining the size dependency of the multi-agent system.

Naturally, the thorough assessment encompasses not merely the uncontrolled environments, but also involves comparative analyses with other event-driven methodologies implemented within SUMO. This allows the results to be compared with current solutions as well. Among these methods, the ”adaptive” and ”delay-based” approaches will be examined. The second method implements efforts similar to those of the proposed algorithm, with the primary distinction being that they are not sensitive to vehicle speeds. Instead, their primary objective is to minimize delays.

In the course of the evaluation, the load applied to the transportation system is calibrated to its maximum capacity, mirroring the conditions during the training phase. However, the traffic generated on it has a different distribution, thus taking one step further away from overfitting.

As can be seen in Figure 7-14, the formation of a shock wave is clearly manifested in both emissions and waiting times. It is also noticeable from the observations that as the scale of the network increases, the phenomenon of persistent congestion within the network becomes more prevalent. Looking at the characteristics of the graphs, it also shows that not only are the peak values lower for both waiting time and emission, but the fluctuations also occur to a lesser extent, hence stabilizing traffic flow. This, as well as the vi-

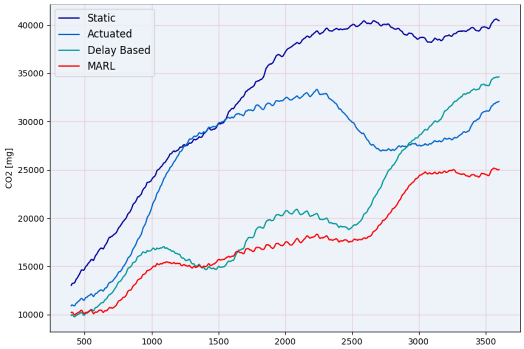

Figure 8: CO_2 Emission in a 2 intersection network.

Table 1: A 30-minute time window for testing on a one-intersection network.

	Waiting Time[s]	Average Speed [m/s]	CO2 [mg]	NOx [mg]	Halting Vehicles [1/s]
Static	67.206	4.1012	19480.115	8.66188	7406.75
Actuated	2.932	7.4978	7596.198	3.20075	1046.0
Delay Based	5.127	6.9916	8013.949	3.39340	1320.0
MARL	1.717	7.2027	7290.073	3.06179	803.25

Table 2: A 60-minute time window for testing on a one-intersection network.

	Waiting Time[s]	Average Speed [m/s]	CO2 [mg]	NOx [mg]	Halting Vehicles [1/s]
Static	72.856	3.9614	20694.588	9.21786	15991.5
Actuated	2.989	7.5899	7591.442	3.19872	2081.5
Delay Based	4.791	6.9903	7923.337	3.35206	2574.75
MARL	1.828	7.1689	7262.742	3.049596	1626.75

Table 3: A 30-minute time window for testing on a two-intersection network.

	Waiting Time[s]	Average Speed[m/s]	CO2 [mg]	NOx [mg]	Halting Vehicles [1/s]
Static	86.028	5.1520	24209.949	10.67928	8876.5
Actuated	63.670	5.4120	22015.251	9.66317	7326.125
Delay Based	27.274	6.6790	14743.874	6.31286	3723.75
MARL	10.442	7.2813	13744.383	5.85558	2585.875

Table 4: A 60-minute time window for testing on a two-intersection network.

	Waiting Time[s]	Average Speed [m/s]	CO2 [mg]	NOx [mg]	Halting Vehicles [1/s]
Static	137.594	4.7504	31886.470	14.21745	25823.75
Actuated	84.979	5.0486	26022.961	11.50417	18866.75
Delay Based	59.250	5.7190	20740.818	9.07874	13926.375
MARL	12.601	6.8569	17664.559	7.65427	6714.75

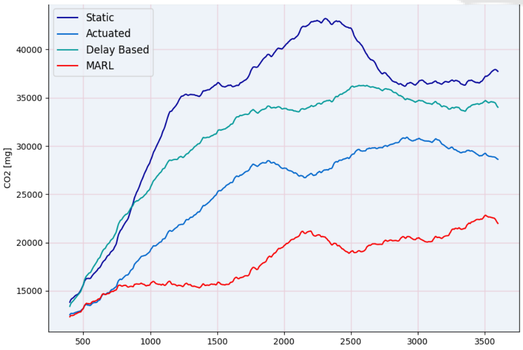

Figure 9: CO_2 Emission in a 3 intersection network.

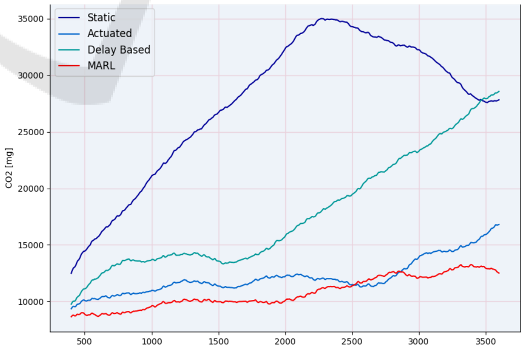

Figure 10: CO_2 Emission in a 4 intersection network.

sual verification in the simulator, confirms that the algorithm contributes positively to achieving diminishing transients. In addition, the diagrams also show a decline in the non-controlled environment, which can be explained by the network becoming so saturated that, in this case, it cannot accommodate as many vehicles as in the controlled systems.

The Tables 1-8 also list another argument for the use of the MARL-based method. Considering the baseline algorithms, it can be observed that their relative performance depends on the size of the network and the duration of the application. The results do not diverge to such an extent that one can be definitively identified as more suitable for the entire problem.

Table 5: A 30-minute time window for testing on a three-intersection network.

	Waiting Time[s]	Average Speed [m/s]	CO2 [mg]	NOx [mg]	Halting Vehicles [1/s]
Static	163.230	5.3928	27883.710	12.41850	11905.583
Actuated	96.199	6.3263	20162.268	8.84889	7260.666
Delay Based	114.739	5.7035	25289.744	11.21376	10104.166
MARL	14.065	7.5449	15173.915	6.53076	2960.583

Table 6: A 60-minute time window for testing on a three-intersection network.

	Waiting Time[s]	Average Speed [m/s]	CO2 [mg]	NOx [mg]	Halting Vehicles [1/s]
Static	323.762	4.7415	33404.804	14.96324	30599.5
Actuated	114.715	5.8889	24615.000	10.89000	18852.25
Delay Based	149.480	5.1378	29977.941	13.36289	24615.083
MARL	20.282	7.1369	17934.456	7.79766	7765.5

Table 7: A 30-minute time window for testing on a four-intersection network.

	Waiting Time[s]	Average Speed [m/s]	CO2 [mg]	NOx [mg]	Halting Vehicles [1/s]
Static	75.367	6.0456	21628.051	9.50294	7924.75
Actuated	16.962	7.7325	10934.342	4.57993	2286.875
Delay Based	32.135	7.2594	12978.820	5.51974	3435.8125
MARL	3.888	8.5647	9487.481	3.90885	1188.8125

Table 8: A 60-minute time window for testing on a four-intersection network.

	Waiting Time[s]	Average Speed [m/s]	CO2 [mg]	NOx [mg]	Halting Vehicles [1/s]
Static	103.552	5.6438	26662.658	11.81623	21105.1875
Actuated	25.294	7.3910	12265.355	5.187802	5929.75
Delay Based	68.154	6.3610	17919.312	7.78420	12042.75
MARL	5.200	8.1890	10737.716	4.47625	3080.0625

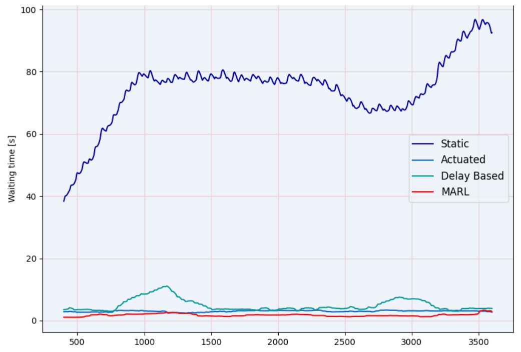

Figure 11: Waiting Time in a 1 intersection network.

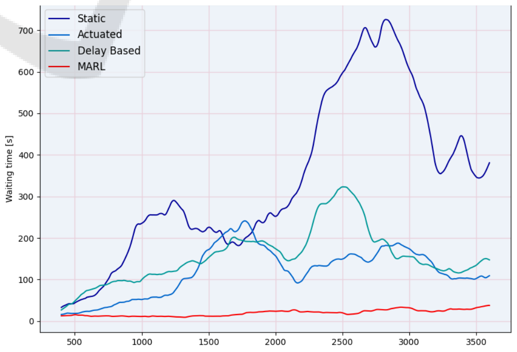

Figure 12: Waiting Time in a 3 intersection network.

This manifests in the sense that, considering waiting times and average speed, sometimes the application of one proves to be more advantageous, while at other times the other does. Of course, this shows a similar trend to the number of vehicles forced to wait, resulting in a decrease in speed for the stationary vehicles.

With these values, the trend in emissions also shows a similar favorability. Compared to these, the MARL-based solution is able to further reduce emissions in every case. In addition, it proved to be more successful in every case in reducing both the waiting time and the number of vehicles forced to wait. The increase in

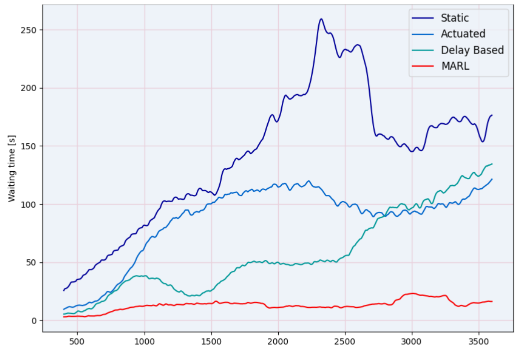

Figure 13: Waiting Time in a 2 intersection network.

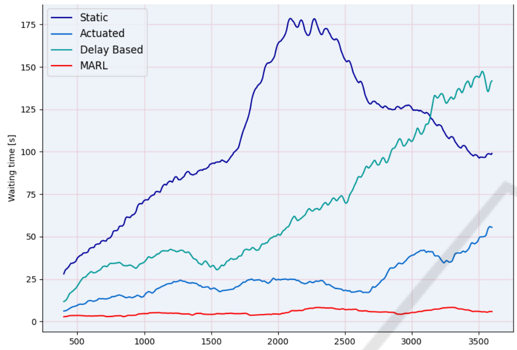

Figure 14: Waiting Time in a 4 intersection network.

speed is also almost always noticeable, with only the adaptive algorithm achieving a marginally higher average speed in the single-intersection network. However, this is most expected in this network, since, on one hand, congestion does not form between traffic lights here, as the vehicles passing through immediately flow off the network, and on the other hand, the advantages of identical payoff hardly apply, considering that there is only one intersection in the system.

In summary, based on the examination of the desired indicators, the training conducted on the largest network is suitable for controlling the other networks as well, since improvements are observed everywhere in the indicators articulated during the motivation, compared to the baseline algorithms.

5 CONCLUSION

The results prove, in a new light, that the application of MARL is justified in the case of traffic networks. The novelty, that not only environments with a predetermined number of agents can be controlled in this way, but also that appropriately formulated abstractions can apply a single model across different sizes, is able to reduce both the speed of training and the amount of resources allocated for it. With this, individual models can be deployed more quickly to real networks, allowing traffic management aimed at reducing emissions to start sooner, thus accelerating steps towards sustainability and creating more livable cities.

Cognitive decision-making still shows numerous areas for development. Among other things, choosing the size of the training network appears to be a promising area of research, as faster and thereby cheaper convergence can be achieved on a smaller network, but the suitability of the resulting model may not be sufficient. Researching this and thereby finding an optimum can provide further important advances. Additionally, from the perspective of applicability to a significant part of cities, the problem is that they do not consist of intersections with identical characteristics. This demonstrates that the deployment of homogeneous agents is not feasible in such scenarios. As a consequence, the application and scalability of heterogeneous agents also emerge as important areas of research, thereby expanding the potential for implementation in real-world environments.

ACKNOWLEDGEMENTS

This work was supported by the European Union within the framework of the National Laboratory for Autonomous Systems (RRF-2.3.1-21-2022-00002). T.B. was supported by BO/00233/21/6: the János Bolyai Research Scholarship of the Hungarian Academy of Sciences.

REFERENCES

Abdulhai, B., Pringle, R., and Karakoulas, G. J. (2003). Reinforcement learning for true adaptive traffic signal control. *Journal of Transportation Engineering*, 129(3):278–285.

Abed-Alguni, B. H. K. et al. (2014). Cooperative reinforcement learning for independent learners. *Computer Science*.

Behrisch, M., Bieker, L., Erdmann, J., and Krajzewicz, D. (2011). Sumo–simulation of urban mobility: an overview. In *Proceedings of SIMUL 2011, The Third International Conference on Advances in System Simulation*. ThinkMind.

Buşoniu, L., Babuška, R., and De Schutter, B. (2010). Multi-agent reinforcement learning: An overview. *Innovations in multi-agent systems and applications-1*, pages 183–221.

Cao, L. (2020). Ai in finance: A review. *Available at SSRN 3647625*.

Cottier, B. (2023). Trends in the dollar training cost of machine learning systems. Accessed: 2023-12-18.

Dimitrakopoulos, G. and Demestichas, P. (2010). Intelligent transportation systems. *IEEE Vehicular Technology Magazine*, 5(1):77–84.

Fenger, J. (1999). Urban air quality. *Atmospheric environment*, 33(29):4877–4900.

Kesting, A. and Treiber, M. (2008). How reaction time, update time, and adaptation time influence the stability of traffic flow. *Computer-Aided Civil and Infrastructure Engineering*, 23(2):125–137.

Khondaker, B. and Kattan, L. (2015). Variable speed limit: an overview. *Transportation Letters*, 7(5):264–278.

Kolat, M., Kővári, B., Bécsi, T., and Aradi, S. (2023). Multi-agent reinforcement learning for traffic signal control: A cooperative approach. *Sustainability*, 15(4):3479.

Kővári, B., Knáb, I., and Bécsi, T. (2024). Variable speed limit control for highway scenarios a multi-agent reinforcement learning based appraoch. Technical report, EasyChair.

Kövári, B., Pelenczei, B., and Bécsi, T. (2022). Monte carlo tree search to compare reward functions for reinforcement learning. In *2022 IEEE 16th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, pages 000123–000128. IEEE.

Krajzewicz, D., Hausberger, S., Wagner, P., Behrisch, M., and Krumnow, M. (2014). Second generation of pollutant emission models for sumo. In *SUMO2014 - Second SUMO User Conference*, Reports of the DLR-Institute of Transportation Systems.

Kustija, J. et al. (2023). Scats (sydney coordinated adaptive traffic system) as a solution to overcome traffic congestion in big cities. *International Journal of Research and Applied Technology (INJURATECH)*, 3(1):1–14.

Mirchandani, P. and Wang, F.-Y. (2005). Rhodes to intelligent transportation systems. *IEEE Intelligent Systems*, 20(1):10–15.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. (2015). Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533.

Nadarajan, R. and Sulaiman, N. (2021). Comparative analysis in execution of machine learning in breast cancer identification: A review. *Journal of Physics: Conference Series*, 1874:012032.

Nowé, A., Vrancx, P., and De Hauwere, Y.-M. (2012). Game theory and multi-agent reinforcement learning. *Reinforcement Learning: State-of-the-Art*, pages 441–470.

Qureshi, K. N. and Abdullah, A. H. (2013). A survey on intelligent transportation systems. *Middle-East Journal of Scientific Research*, 15(5):629–642.

Reddy, S., Allan, S., Coghlan, S., and Cooper, P. (2020). A governance model for the application of ai in health care. *Journal of the American Medical Informatics Association*, 27(3):491–497.

Ritchie, H. (2020). Sector by sector: where do global greenhouse gas emissions come from? *Our World in Data*. https://ourworldindata.org/ghg-emissions-by-sector.

Ritchie, H. (2024). Tracking global data on electric vehicles. *Our World in Data*. https://ourworldindata.org/electric-car-sales.

Ross, P. and Maynard, K. (2021). Towards a 4th industrial revolution.

Rotake, D. and Karmore, S. (2012). Intelligent traffic signal control system using embedded system. *Innovative systems design and engineering*, 3(5):11–20.

Singh, B. and Gupta, A. (2015). Recent trends in intelligent transportation systems: a review. *Journal of transport literature*, 9:30–34.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

Sutton, R. S., Barto, A. G., et al. (1999). Reinforcement learning. *Journal of Cognitive Neuroscience*, 11(1):126–134.

Szepesvári, C. (2022). *Algorithms for reinforcement learning*. Springer nature.

Wang, F.-Y., Zhang, J. J., Zheng, X., Wang, X., Yuan, Y., Dai, X., Zhang, J., and Yang, L. (2016). Where does alphago go: From church-turing thesis to alphago thesis and beyond. *IEEE/CAA Journal of Automatica Sinica*, 3(2):113–120.

Wei, H., Zheng, G., Gayah, V., and Li, Z. (2021). Recent advances in reinforcement learning for traffic signal control: A survey of models and evaluation. *ACM SIGKDD Explorations Newsletter*, 22(2):12–18.

Wei, H., Zheng, G., Yao, H., and Li, Z. (2018). Intellilight: A reinforcement learning approach for intelligent traffic light control. In *Proceedings of the 24th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 2496–2505.

Wiering, M. A. et al. (2000). Multi-agent reinforcement learning for traffic light control. In *Machine Learning: Proceedings of the Seventeenth International Conference (ICML'2000)*, pages 1151–1158.

Ye, B.-L., Wu, W., Ruan, K., Li, L., Chen, T., Gao, H., and Chen, Y. (2019). A survey of model predictive control methods for traffic signal control. *IEEE/CAA Journal of Automatica Sinica*, 6(3):623–640.

Zhang, J., Wang, F.-Y., Wang, K., Lin, W.-H., Xu, X., and Chen, C. (2011). Data-driven intelligent transportation systems: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 12(4):1624–1639.