# A Critical Survey: Traffic Signal Control Models in an Agent-Based Simulation

Zina Abohaia, 20194621@ug.buid.ac.ae

Yousef Mamdouh , 20196448@ug.buid.ac.ae

## Abstract

Traffic congestion harms our economies, environment, and ourselves, since it increases the risk of accidents. The current Traffic Signal Control (TSC) is not efficient, and adaptive, scalable, and stable solutions are required. Agent-Based Simulations (ABS) have managed to become a way to simulate otherwise complicated scenarios, and a way to verify various systems before deploying them. This paper reviews recent literature on ABS of TSC. Papers are divided according to their aim; Using reinforcement learning and ABS to solve TSC, Introducing Novel algorithms and frameworks, and improving ABS softwares.

## Introduction

Traffic congestion is a growing issue that not only affects drivers who are tediously delayed. According to Sweet (2011), traffic congestion “slows metropolitan growth, inhibits agglomeration economies, and shapes economic geographies”. Not just that but in 2018, the United States (US) lost $87 billion to traffic congestion, and the average citizen of Boston in the US lost 164 hours from their year (Traffic congestion cost the US economy nearly $87 billion in 2018, 2022). It also has a negative impact on the environment, energy-saving agendas, and it increases the risk of accidents due to motorcycles' rash behavior during traffic congestion.

How traffic signals are calibrated, or scheduled, plays a big role in relieving traffic congestion. The introduction of Split, Cycle, Offset Optimisation Technique (SCOOT) in the 1980s, led to reductions in delays from 15% to 30% in different areas around the world (Urban traffic control systems evidence on performance, 2022). It was the first adaptive traffic signal control system, which means that according to the input it got from its sensors, it responded. According to Fang et al. (2013), “the most common type of control strategies are fixed time and actual control”. However, researchers everywhere, and in different specializations, are trying to find more efficient, adaptive, scalable, and stable scheduling methods or artificial intelligence models to optimize TSC. Recent research has looked into TSC in agent-based environments for a novel solution.

Agent-based systems are made up of entities, or agents, that interact with one another, and the environment they perceive. It has been commonly used to “understand the emergent behavior of complex systems” (Santos, Nunes and Bazzan, 2018). Building them, in the form of a simulation, is difficult, but has provided promising results to optimizing TSC.

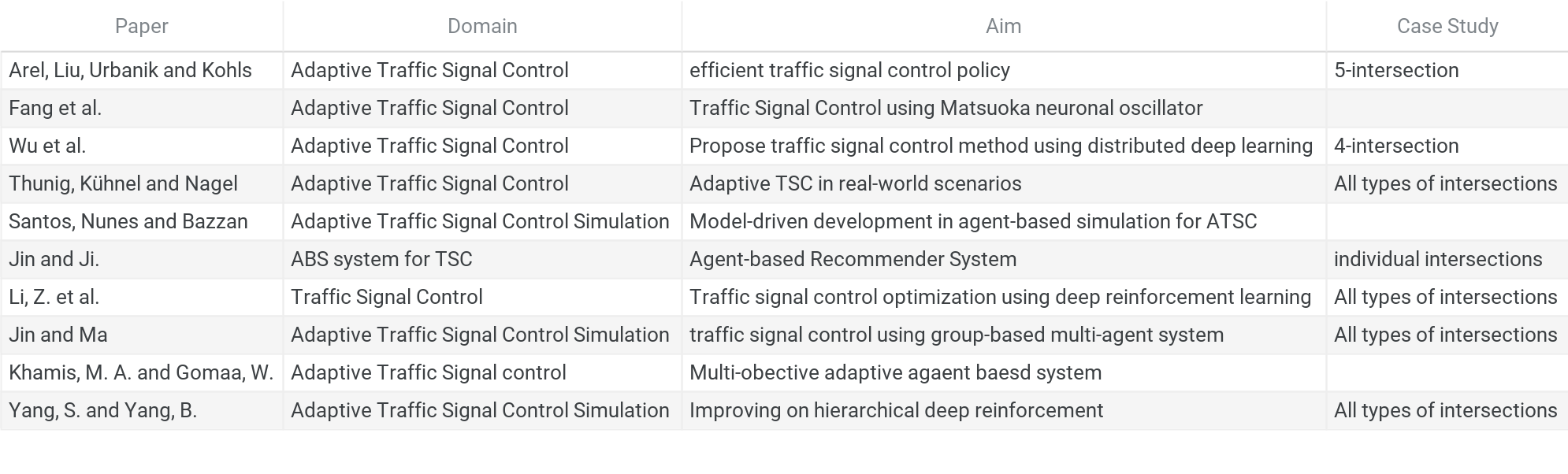
In this critical survey, we cover how reinforcement learning has been used with agent-based systems to solve conventional TSC, novel methods and frameworks, as well as improvement and development of simulation softwares. The sections are divided as listed, respectively.

## 

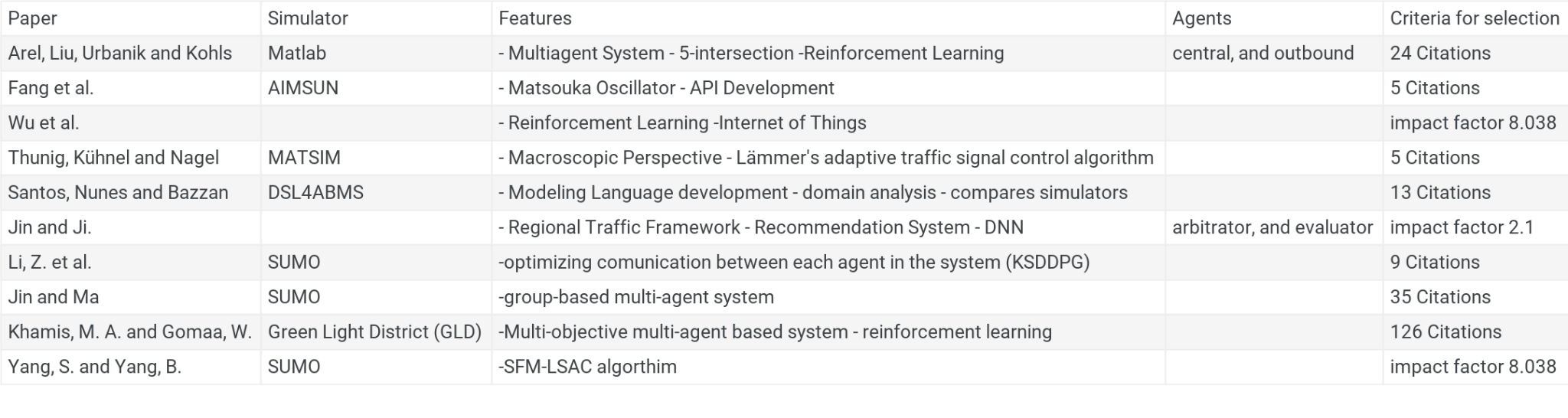
## 

## Literature review

This section summarizes the papers reviewed in the aim of highlighting advancements and research gaps in Agent-Based Traffic Signal Control (TSC). The following table summarizes the papers reviewed, their domain, aim, the case studied, simulator used, features, agents, and how its quality was evaluated. This structure is inspired by the structure of a table in the sample case study posted. Also, all the papers are open source.

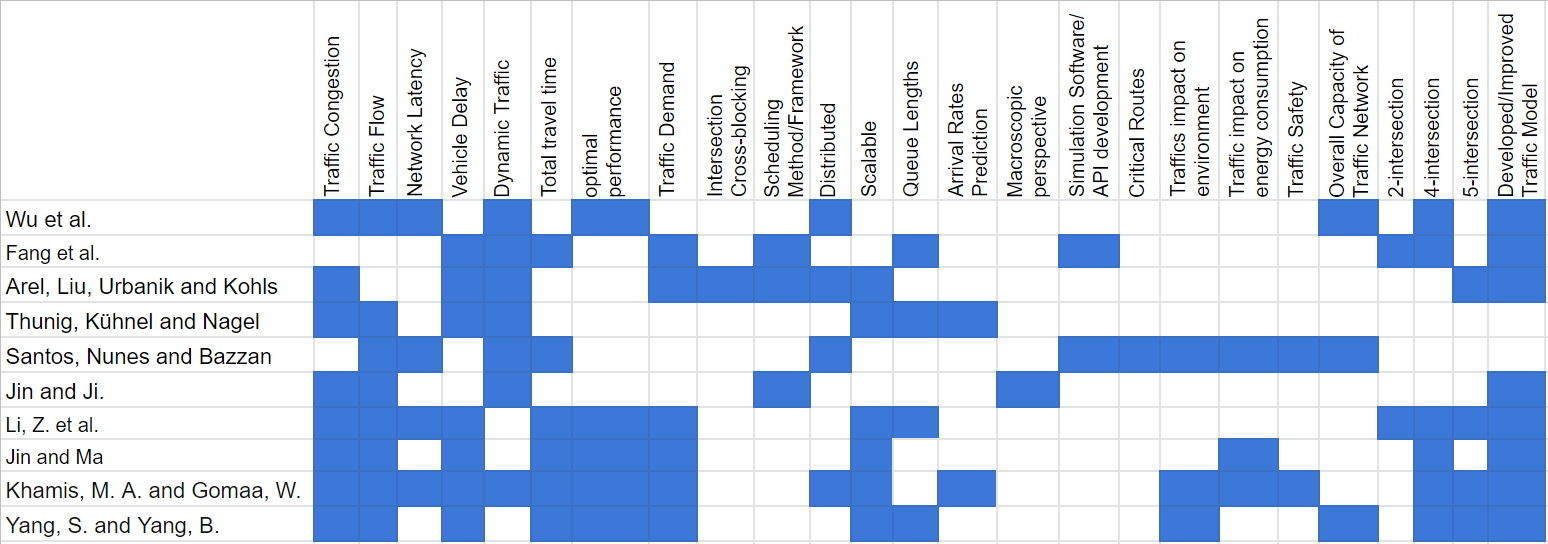


**Figure 1**: Papers Summary 1



**Figure 2**: Papers Summary 2

The following grid chart highlights the research issues mentioned in each paper, and those not mentioned:



**Figure 3**: Grid Chart

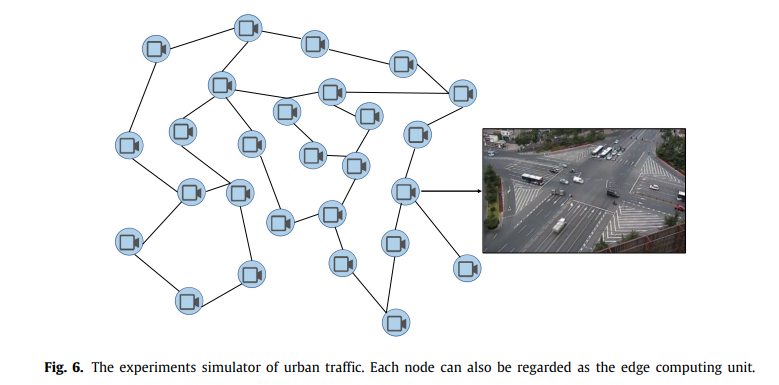
We see that Traffic congestion, and developing or improving a traffic model, are the most researched issues. This makes sense since traffic congestion is a large reason to improve the traffic model, as we highlighted in our introduction.

We discuss these research issues in detail in the following sections.

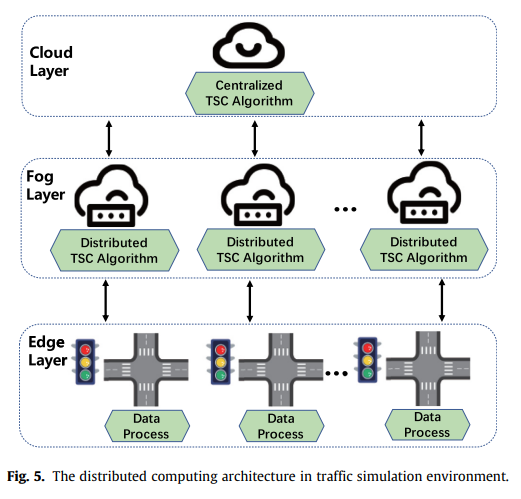
### Using Reinforcement Learning (RL) and ABS To solve traditional TSC

Multiple studies explored using RL to relieve traffic congestion, reduce the average traffic delay by maximizing traffic flow, and evaluating their algorithms performance in comparison with available algorithms.

Wu et al. (2022) presented two game theory-aided RL algorithms, Nash Advantage Actor–Critic (Nash-A2C) and Nash Asynchronous Advantage Actor–Critic (Nash-A3C), and designed an Internet of Things (IoT) distributed architecture for traffic simulation to deploy the algorithms in its fog layer. The aim from the development of these algorithms is to reduce traffic congestion time and network delay due to the large amount of video data being transmitted. The distributed IoT architecture consists of three layers: cloud layer, fog layer, and edge layer. The cloud layer has the centralized TSC algorithm, while the fog layer has the distributed TSC algorithm, and the edge layer is placed near traffic lights, and has these three functions: Detect vehicles’ information, send vehicles’ information to fog node, and receive a command, and control the traffic signal accordingly.

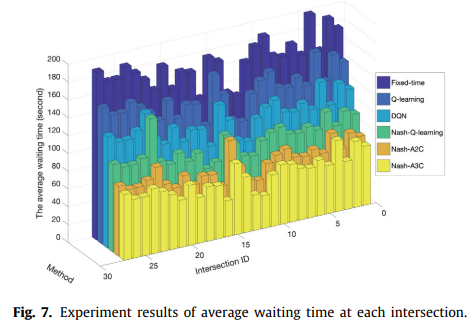


**Figure 4**: The experiments simulator of urban traffic. Each node can also be regarded as the edge computing unit (Wu et al.,2022).



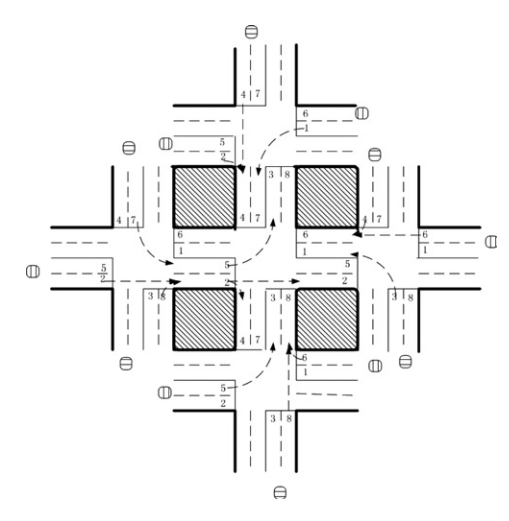
**Figure 5:** The distributed computing architecture in traffic simulation environment (Wu et al.,2022).

In the study, they built two simulation environments, one that doesn’t consider date or results delay, only collects the operating results of different algorithm models. While the second environment takes into account the network latency, and both environments have 27 edge processes. The Models tested were: Fixed-Time, Q-Learning, DQN, Nash-Q, Nash-A2C, and Nash-A3C. The algorithms were tested thoroughly, using 3000 episodes for training, and 1000 episodes for testing, each 30 minutes long. Their results showed that their algorithms, Nash-A2C, and Nash-A3C, outperformed the other models by a 22.1% reduction in traffic congestion time, and 9.7% reduction in network latency. The study only considered 4-intersection, and did not consider other intersections, but concluded that a more lightweight version of the model would be suitable for deployment.



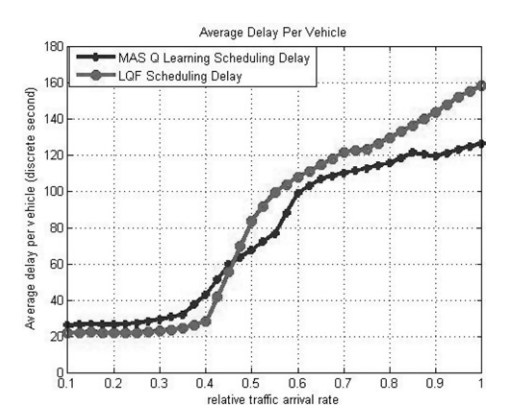
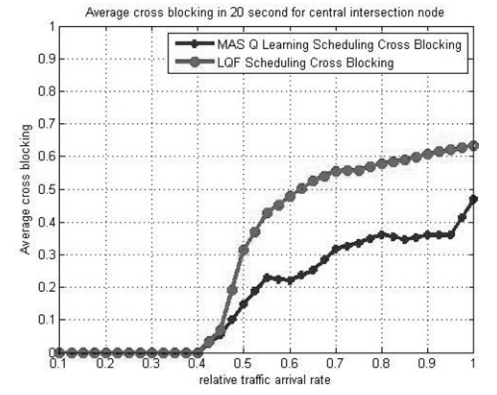
**Figure 6:** Experiment results of average waiting time at each intersection (Wu et al., 2022).

Arel, Liu, Urbanik and Kohls (2010) combined a multiagent system with a RL framework to develop a TSC policy. The multiagent system included two types of agents: central, and outbound, where the former “learns a value function driven by its local and neighbours’ traffic conditions”, and the latter follows the longest-queue-first (LQF) scheduling algorithm to schedule TSC, while providing the central agent with traffic statistics. They aimed to minimize average delay, traffic congestion, and the possibility of intersection cross-blocking by utilizing the Q-Learning algorithm with a feedforward neural network for value function approximation. The simulation was based on a five-intersection, and only the vehicles that passed the central intersection had their statistics taken into account.



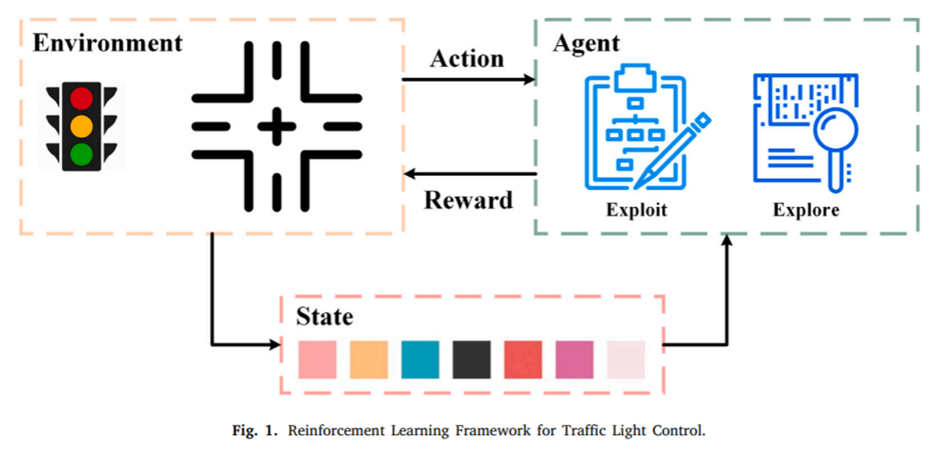
**Figure 7**: 5-intersection plan (Arel, Liu, Urbanik and Kohls, 2010).

The researchers found that combining RL with LQF, and treating this “scheduling problem as an artificial intelligence task”, instead of a traditional scheduling problem, improved the performance by reducing the average delay, and the possibility of cross-blocking. Another advantage is that the system was designed to be scalable, to accommodate future needs. However, the main disadvantage of this study is that they did not compare it to other scheduling algorithms, not even fixed-time, which is the one commonly used. They only compared it to the algorithm they improved, showing how the addition of RL gave better results, as is shown below. They also didn’t consider other intersections than the five-intersection.

**Figure 8**: Average Delay and Average Closs blocking of Researchers algorithm and LQF (Arel, Liu, Urbanik and Kohls, 2010).

Li, Yu, Zhang, Dong, and Xu (2021) proposed a novel multi agent RL method which they called “KSDDPG (Knowledge Sharing Deep Deterministic Policy Gradient)”as a solution to optimize the ATSC system. They introduced a knowledge-sharing enabled communication protocol in which each agent can access all the information about the traffic environment collected by all agents, aiming to achieve most efficient control through enhancing traffic signals cooperation, to elevate traffic congestion and energy waste. Their KSDDPG has the ability to control large scale traffic networks with high efficiency and cope with changes in traffic flow when compared to traditional methods.

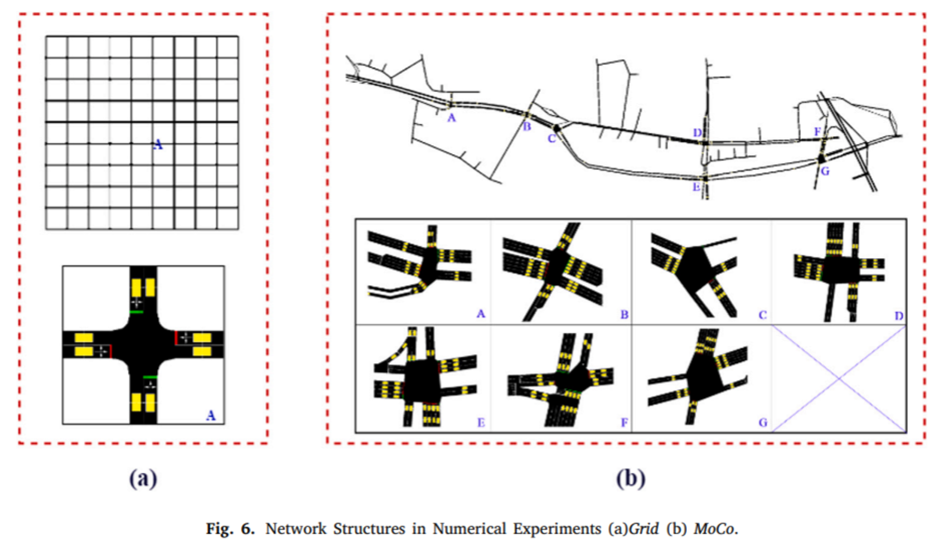


**Figure 9**: Reinforcement Learning Framework for Traffic Light Control (Li, Yu, Zhang, Dong, and Xu, 2021).

In the first figure you can see the closed “reward” loop, where the control unit is isolated and is treated as an agent.

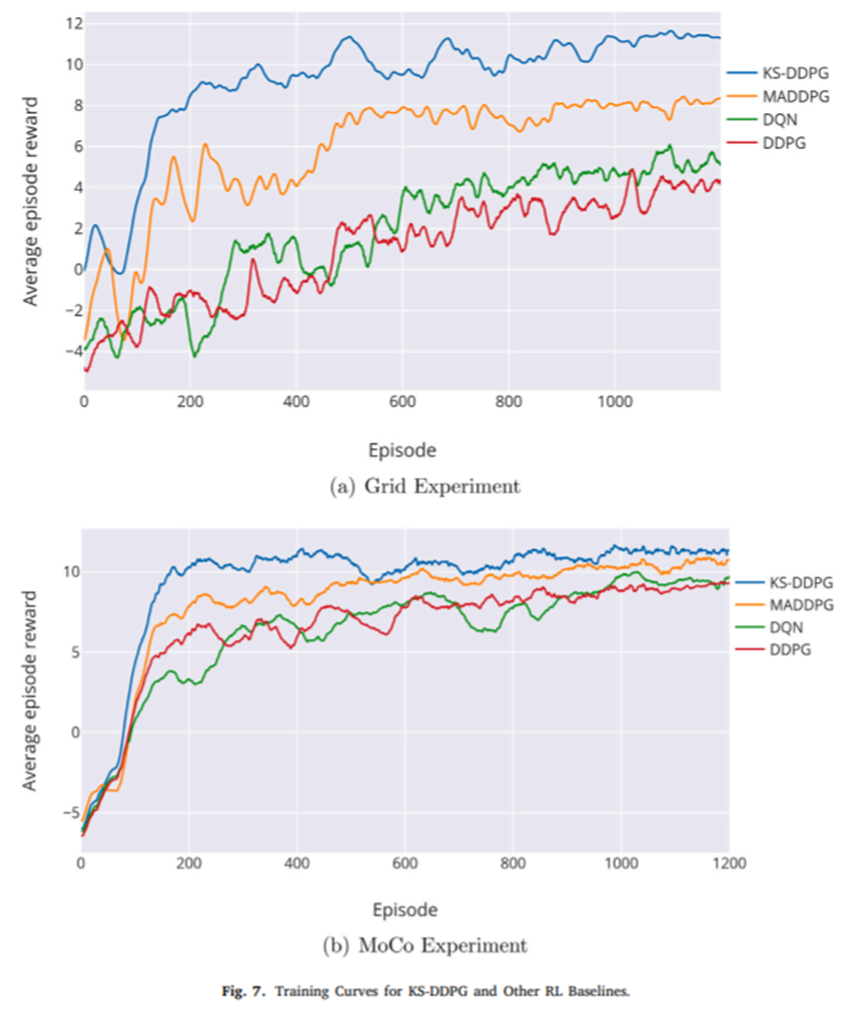
For evaluation of their KSDDPG method they used the open-source traffic simulation package called Simulation of Urban MObility (SUMO) platform, which they chose for its scalability since it has an API named TraCI (Traffic Control Interface) that allowed them to modify the status of the simulation as well as extend the current SUMO functionality, it was also compatible with their RL algorithms.

They performed two experiments over two networks, the first one was called Grid which is a 10 x 10 virtual grid network that has the same signalized intersection structures, and the second one was called MoCo which is a real world network from downtown Montgomery County, it has diverse intersections of signalized and non-signalized ones, as shown in the figure below.



**Figure 10**: Network Structures in Numerical Experiments (a)Grid (b)MoCo (Li, Yu, Zhang, Dong, and Xu, 2021).

In both experiments KSDDPG did considerably better than the other RL baselines:

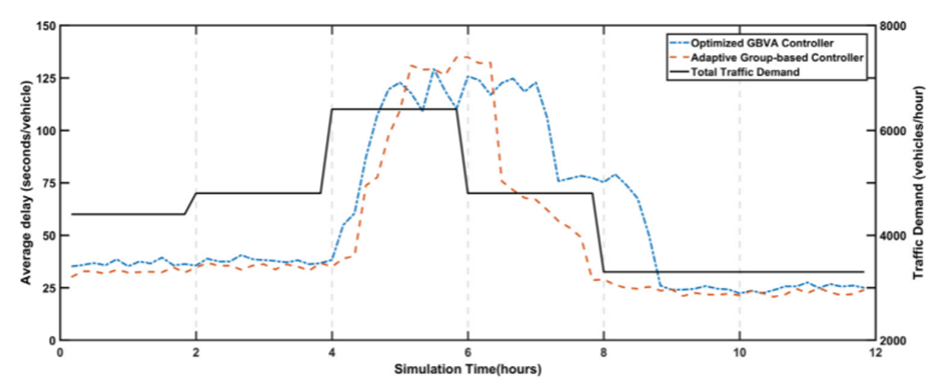


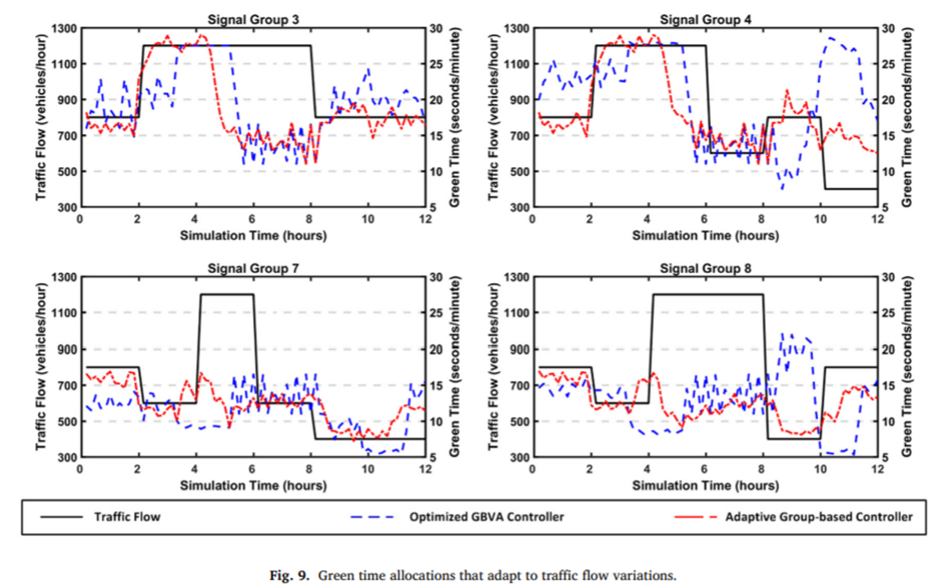
**Figure 11**: Training Curves for KS-DDPG and Other RL Baselines (Li, Yu, Zhang, Dong, and Xu, 2021).

From the results of the experiment, they concluded that KSDDPG is able to reduce queue length by about 28.9%, reduce intersection delay by 35.1%, reduce number of stops by 21%, and increase vehicle speed by about 18.3%. One obvious and main disadvantage of this algorithm is that all agents need to be communicating throughout the whole modeling process which limits the general communication efficiency.

Jin and Ma (2017) proposed a group-based signal control approach, as vehicle actuated timing and phasing schemes are usually used to deal with detected traffic. A big problem in TSC is that the parameters of the signal controller are usually predetermined, which can lead to the performance suffering in highly changing traffic scenarios. They aimed to solve that problem using their group-based control approach, which is able to make decisions depending on the traffic conditions at intersections and its understanding of it. Particularly, every group of signals is modeled as an agent in a multi-agent system, for which they use a framework of varying optimal control. They applied RL and enhanced it with multiple-step backups so that every agent updates their knowledge online. Additionally, their system is compatible with the existing signal system.

They compared the proposed controller with a benchmark controller called “group-based vehicle actuated (GBVA) controller”, with the difference between them being that GBVA’s parameters are offline optimized.

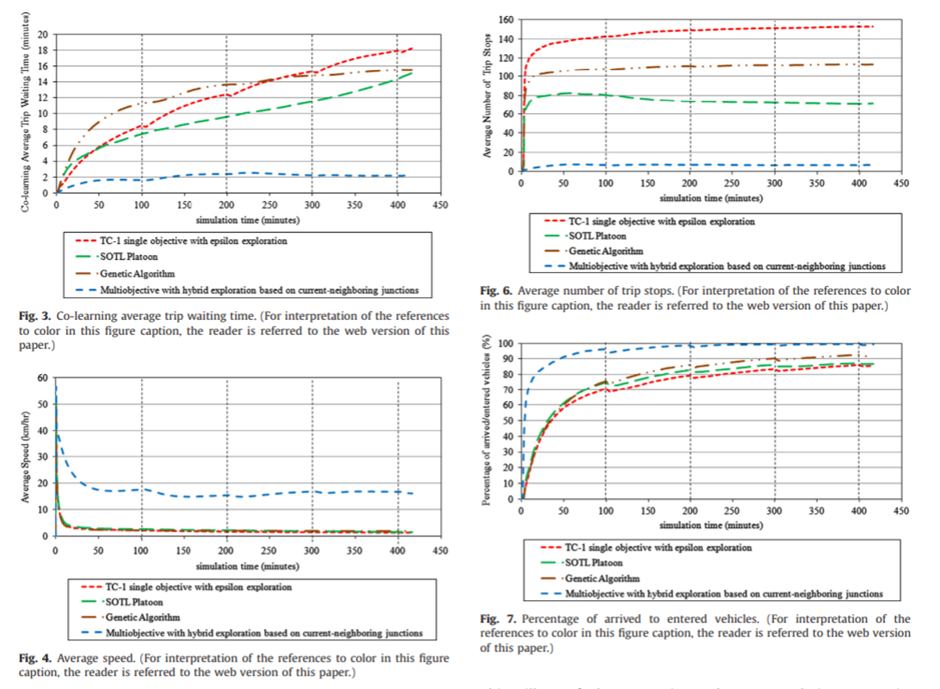




**Figure 12**: Green time allocation that adapt to traffic flow variations (Jin and Ma, 2017).

The results show that the proposed system outperformed the GBVA controller for the most part, this is mainly due to the proposed system real-time adaptivity in response to fluctuating traffic demand. While this system can be applied on a larger scale, it still needs more work and testing to reach the capability of being applied on a large-scale network.

Khamis, M. A. and Gomaa, W. (2014) proposed a multi-agent multi-objective reinforcement learning traffic signal control, instead of only focusing on a single objective multi agent system they divided the problem to objectives so then they optimize each of these objectives to improve the performance at each traffic junction. Their objectives included maximization of traffic flow rate, coordinating multiple traffic lights to allow continuous flow traffic for platoons traveling in main roads (green waves), avoiding traffic accidents mainly in residential and school areas, and minimizing fuel consumption through moderating vehicles speed rates. They developed a framework that mimics drivers’ behaviors based on a multi-objective consecutive decision-making process. They estimated the parameters of said process based on the Bayesian interpretation of probability, using the Green Light District (GLD) vehicle traffic simulator to simulate changing road dynamics. After testing it, they found that the system was effectively able to make real-time adaptations to the aforementioned road dynamics, and that the proposed system was able to highly outperform the single objective controller.



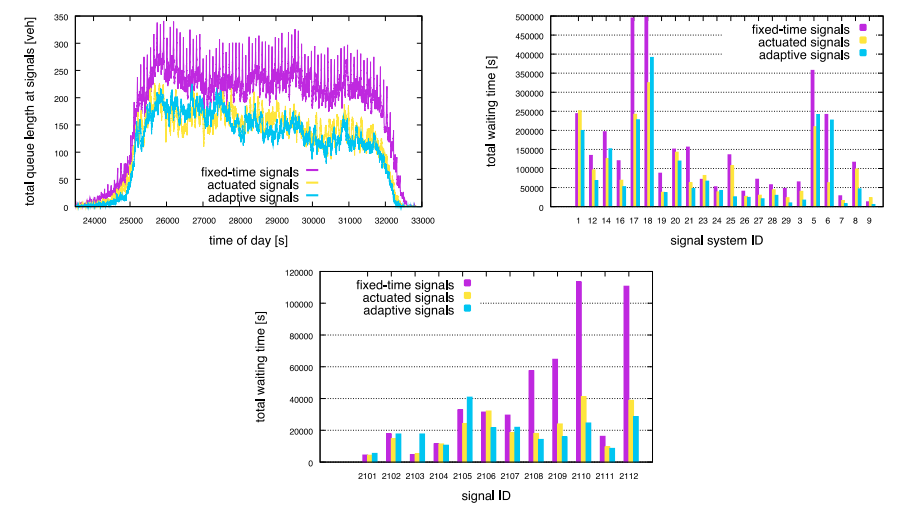
**Figure 13**: Figures comparing average trip waiting time, average number of trip stops, average speed, and percentage of arrived to entered vehicles between different algorithms (Khamis, M. A. and Gomaa, W., 2014).

And in every other performance aspect, the proposed approach has great potential, although the testing done in the study isn’t sufficient. Furthermore, it can be expanded to work on greater scales.

### Introducing Novel Algorithms and Frameworks

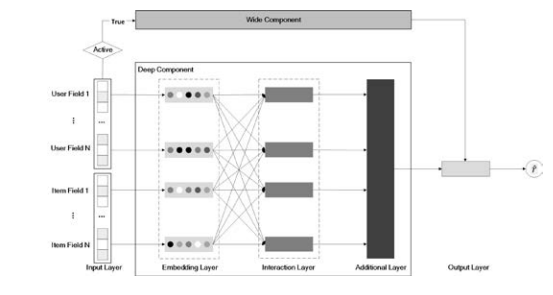
Fang et al. (2013) proposed using the Matsuoka neuronal oscillator to reduce vehicle delay times, and queues, in an isolated signaled four-intersection. They chose to research the Matsuoka neuronal oscillator due to its “stable and predictable rhythmic outputs that exploit autonomously the dynamics of the road system”. The researchers developed an API to simulate the Matsuoka Oscillator Adaptive Signal System (MOASC) and a function that can be used to implement customer-tailored applications, and which is linked to the AIMSUN simulator. The experiment used two matsuoka oscillators to analyze the inputs and outputs, which they found proportional. They compared the results of MOASC with fixed-time, and found that “the Matsuoka model improves the entire traffic performance of delay time and total travel time only when the difference of traffic demand is going high”. Even though the model didn’t achieve the aim it set out to do, the development of an API was certainly a good addition to the literature. Again, we see that the model was only compared to fixed time, and not to other algorithms too. However, it analyzed both a two-intersection, and a four-intersection, which is better than just analyzing one of them.

Thunig, Kühnel and Nagel (2019) presented an “open-source implementation of a decentralized, adaptive signal control algorithm in the agent-based transport simulation MATSim, which is applicable for large-scale real-world scenarios”. They aimed to reduce vehicle delay times, and queue lengths, and achieved that when compared to fixed-time, and traffic-actuated signal control. They improved an existing algorithm suggested by Lammer and Helbing (2008) and adapted to work in real-world scenarios. When traffic is high, the system works like the fixed-time, and so they consider that to make it stable system-wide. They also tried the system at a variety of intersections but the data they had was insufficient to allow them to test the algorithm better, especially in three–way intersections.



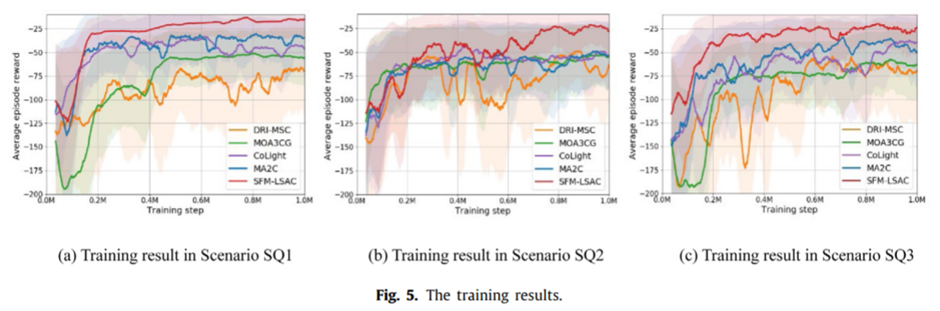
**Figure 14**: Figures comparing total queue length at signals, total waiting time by signal system IDs as well signal IDs, between various algorithms (Thunig, Kühnel and Nagel, 2019).

Jin and Ji. (2020) proposed combating traffic congestion by defining a strategic regional traffic management system at a macroscopic level. They developed a framework that utilized Deep Neural Networks (DNN) to recommend “signal cycle length at intersection-level to the traffic operators upon operational mode switch”. The DNN model has two components: a deep component, and a wide component. The deep component is a stacked DNN with four layers: input layer, embedding layer, interaction layer, and an additional layer that “represents the final vector processing step”. The wide component trains linear models with nonlinear feature transformation, and both the deep and the wide component work together to produce the recommendation of a single cycle length. The traffic state monitor, agent arbitrator, and agent evaluator produce a traffic state indicator, which determines when the operational switch occurs. However, the study only covered individual intersection-based, and was not detailed in the exact way on how this framework could be adapted, or scaled.



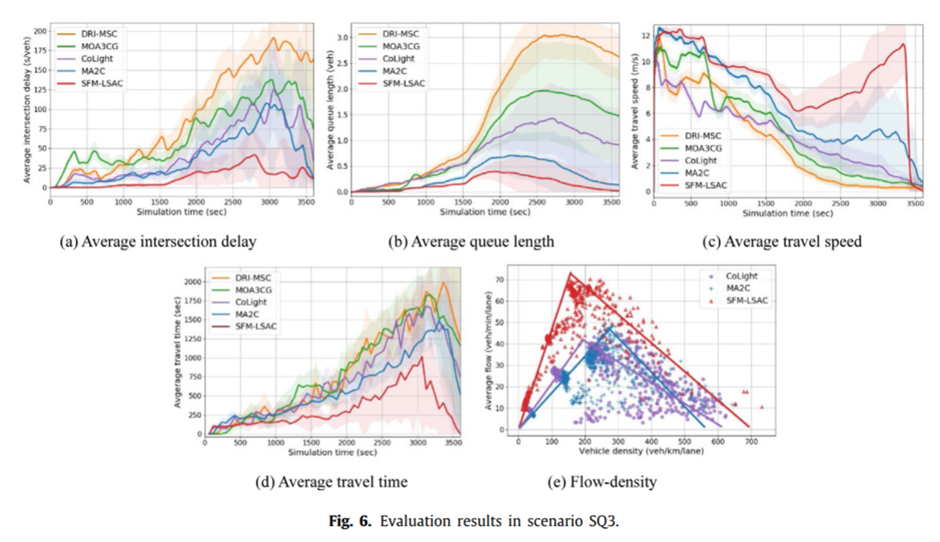
**Figure 15**: DNN System components (Jin and Ji., 2020).

Yang, S. and Yang, B. (2021) aimed to solve a main problem with the algorithms for multi-intersection traffic signal control that uses hierarchical deep RL, these types of methods usually use a two-level hierarchical structure where the lower-level employs inactive goals pushed by the higher-level policies, the problem with these methods is that these pushed goal are not optimal, to solve that they proposed two methods combined, “learned-goal soft actor–critic (LSAC) algorithm” which will automatically learn the optimal pushed goal to then use them in the lower-level, and a “semi-decentralized feudal multi-agent (SFM) framework” which can solve the problem of the increasing number of agents, creating an SFM-LSAC algorithm, they did sufficient testing using SUMO on three different real-life traffic scenarios datasets



**Figure 16**: The training results in various scenarios (Yang, S. and Yang, B., 2021).

And the results show that that their SFM-LSAC outperformed other state-of-the-art algorithms

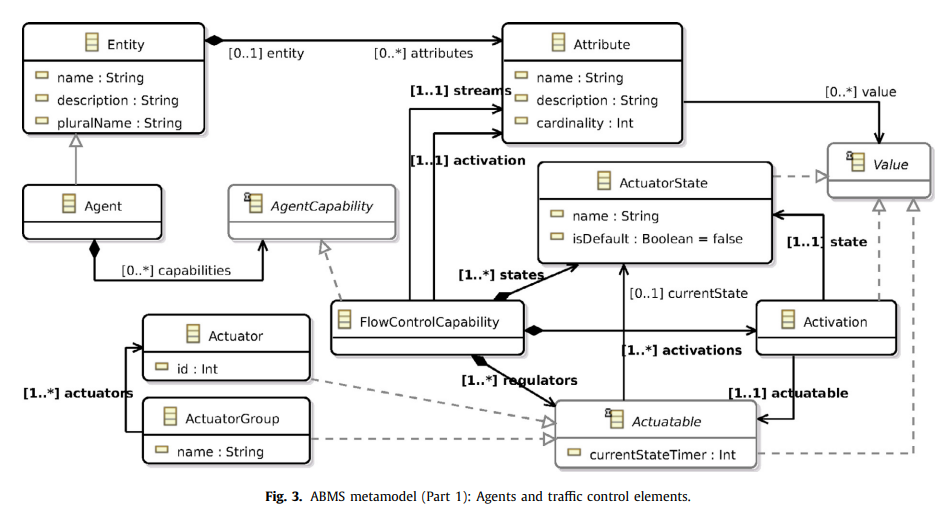


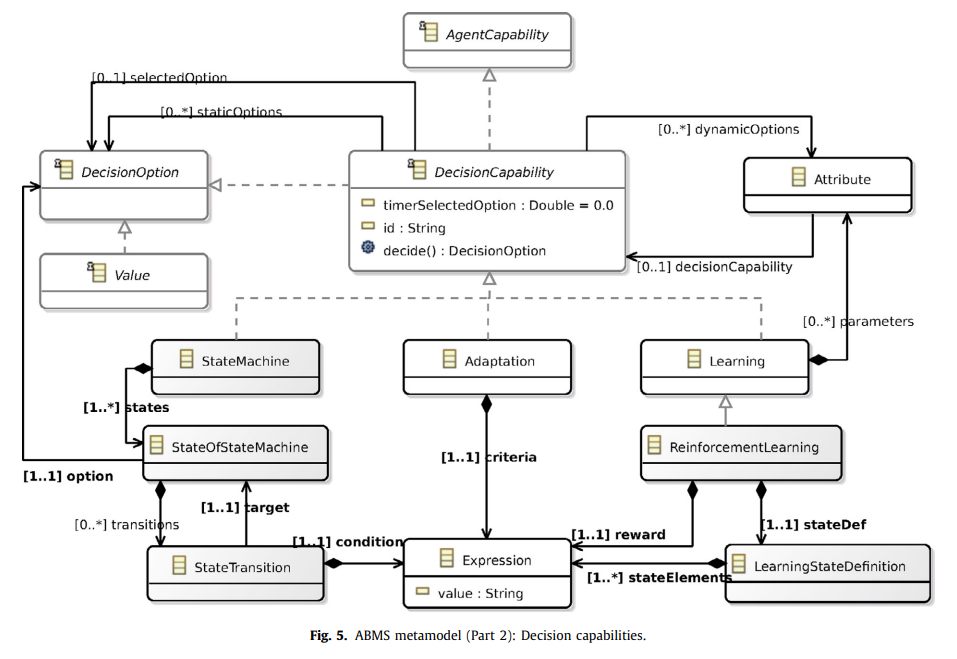
**Figure 17**: Evaluation results in scenario SQ3 (Yang, S. and Yang, B., 2021).

This approach has been getting more attention lately, but it is still a new developing idea, therefore it can be improved in various ways, and although the study seems sufficient it was still quite experimental.

### Improving ABS Software

Santos, Nunes and Bazzan (2018) employed Model-driven Development (MDD) in agent-based modeling and simulation in the adaptive TSC (ATSC) field. They aimed to optimize traffic flow, as well as reduce the workload to develop agent-based simulations in the ATSC field, by developing a modeling language, and model-to-code transformations to produce “runnable simulations automatically”. They performed domain analysis in four steps: Concept Preliminary List, Overview, Design Concepts, and Details(ODD)-based Refinement, Concept Abstractions, and Domain Modeling. The Metamodel is extensive, and it is clear how every component is necessary in the working of the system. The case study was built using the modeling language developed, and they compared the simulation results by two simulations produced by NetLogo and one simulation from ITSUMO, and found that with NetLogo, they observed a 60.00% and 81.93% respectively, and a 85.21% reduction with ITSUMO. The appendix also included the source code and XML listings of the software. The model is yet to be tested by people, and this was highlighted in their conclusion section.





**Figure 18**: ABMS Metamodel (Santos, Nunes and Bazzan, 2018).

## Conclusion

Traffic Signal Control is a complicated issue due to there being many factors, and many research issues as shown in figure 3. Although many of the papers reviewed here offer promising results, and certainly provide an improvement to the current traffic signal control systems, they are either hard to deploy or require further improvements and testing. However, it is clear that the use of agent-based simulation certainly helped researchers test these different algorithms and models, and all the researchers seem to agree that it is the most suitable way to test them. We have also found that intersection cross-blocking, approaching the problem from a macroscopic perspective, and critical routes, were the most overlooked research issues from the ones present in the different papers. Furthermore, there is no one set method for evaluation of traffic signal control algorithms and models; each researcher tested it differently, and so this makes their comparison difficult.

## Group Project Proposal

After reviewing various algorithms and models tested in an agent-based system, we decided to make a dynamic traffic signal algorithm of our own, and test it in an agent-based system. We don’t have the real time traffic flow to build a machine learning model, and train and test it. Therefore, due to our lack of resources and expertise, we will model the traffic signal as a hybrid agent, where it interacts with other smart agents in its environment to help it reach its decision. Furthermore, we will optimize the algorithm to reduce traffic congestion, average delay time, and compare it to the best performing algorithms reviewed here, as well as to fixed-time and SCOOT which are currently employed in our cities.

## References

Khamis, M. A. and Gomaa, W. (2014) “Adaptive Multi-Objective Reinforcement Learning with Hybrid Exploration for Traffic Signal Control Based on Cooperative Multi-Agent Framework,” *Engineering Applications of Artificial Intelligence*, 29, pp. 134–151. doi: 10.1016/j.engappai.2014.01.007.

Jin, J. and Ma, X. (2017) “A Group-Based Traffic Signal Control with Adaptive Learning Ability,” *Engineering Applications of Artificial Intelligence*, 65, pp. 282–293. doi: 10.1016/j.engappai.2017.07.022.

Li, Z. *et al.* (2021) “Network-Wide Traffic Signal Control Optimization Using a Multi-Agent Deep Reinforcement Learning,” *Transportation Research Part C*, 125. doi: 10.1016/j.trc.2021.103059.

Yang, S. and Yang, B. (2021) “A Semi-Decentralized Feudal Multi-Agent Learned-Goal Algorithm for Multi-Intersection Traffic Signal Control,” *Knowledge-Based Systems*, 213. doi: 10.1016/j.knosys.2020.106708.

Wu, Q., Wu, J., Shen, J., Du, B., Telikani, A., Fahmideh, M. and Liang, C., 2022. Distributed agent-based deep reinforcement learning for large scale traffic signal control. *Knowledge-Based Systems*, 241, p.108304.

Arel, I., Liu, C., Urbanik, T. and Kohls, A., 2010. Reinforcement learning-based multi-agent system for network traffic signal control. *IET Intelligent Transport Systems*, 4(2), p.128.

Fang, F., Xu, W., Lin, K., Alam, F. and Potgieter, J., 2013. Matsuoka Neuronal Oscillator for Traffic Signal Control Using Agent-based Simulation. *Procedia Computer Science*, 19, pp.389-395.

Thunig, T., Kühnel, N. and Nagel, K., 2019. Adaptive traffic signal control for real-world scenarios in agent-based transport simulations. *Transportation Research Procedia*, 37, pp.481-488.

Jin, J. and Ji., Q., 2020. Agent-based Deep Urban Traffic Recommender. *IFAC-PapersOnLine*, 53(5), pp.588-591.

Santos, F., Nunes, I. and Bazzan, A., 2018. Model-driven agent-based simulation development: A modeling language and empirical evaluation in the adaptive traffic signal control domain. *Simulation Modelling Practice and Theory*, 83, pp.162-187.

Sweet, M., 2011. Does Traffic Congestion Slow the Economy?. *Journal of Planning Literature*, 26(4), pp.391-404.

World Economic Forum. 2022. *Traffic congestion cost the US economy nearly $87 billion in 2018*. [online] Available at: <https://www.weforum.org/agenda/2019/03/traffic-congestion-cost-the-us-economy-nearly-87-billion-in-2018/> [Accessed 17 May 2022].

Its.leeds.ac.uk. 2022. *Urban traffic control systems evidence on performance*. [online] Available at: <http://www.its.leeds.ac.uk/projects/konsult/private/level2/instruments/instrument014/l2\_014c.htm> [Accessed 17 May 2022].