Autonomous Traffic Intersection Management Systems: Exploring the Future of Urban Mobility

Kwuan Wei Chee
kwuanwei@usc.edu

Junxi Liao
junxilia@usc.edu

1. INTRODUCTION

Autonomous Intersection Management Systems, encompassing a range of advanced fields such as Artificial Intelligence (AI), the Internet of Things (IoT), Vehicle-to-Everything (V2X) communications, and various networking and scheduling protocols [3], address the pressing needs of urban transportation. These systems can become especially significant in cities where rising living standards lead to more private vehicles, contributing to challenges like traffic congestion and a higher rate of accidents [1]. Autonomous systems thus emerge as a key solution to optimize traffic flow, enhance road safety, and support environmental sustainability by potentially reducing vehicle emissions. With these motivations, our project focuses on investigating and simulating innovative approaches to improve intersection traffic flow and evaluate their efficacy in real-world scenarios.

The operational dynamics of traffic autonomous systems have a spectrum of centralized and distributed models, with varying roles assigned to traffic lights and vehicles as controlling agents. Centralized models typically have central managers that receive data from multiple agents to make decisions on the state. Distributed and more decentralized models have multiple agents that sense or communicate with their surroundings or other agents to make localized decisions [2]. Each configuration, be it centralized, distributed, vehicle-centric, or traffic light-based, carries its own set of advantages and limitations, and their practicality hinges on the existing road infrastructure and the pace at which urban environments are embracing smart technologies.

1.1 CENTRALIZED MODELS

Centralized systems are adept at offering a macro-level management of traffic across multiple intersections or vehicles, leading to smoother city-wide traffic flow and reduced congestion. Examples of methods include Autonomous Intersection Management (AIM), one of the first methods with a First Come First Serve (FCFS) scheduling policy and Connected Autonomous Vehicles (CAVs) to query safe reservations at upcoming intersections that are either accepted or rejected by a central intersection manager [4]; Algorithms to find predefined trajectories, utilizing positioning communication and for optimal time-sensitive traffic management [6, 7]; Red-Black Tree conflict scheduler, to send query requests to the intersection manager [8]; Platoon based management, to schedule packs of vehicles in optimizing flow [9, 10]; Traffic light controllers, to change the state of intersections based on sensing vehicle states [11, 12]; and a plethora of alternatives. Their dependency on single control points introduces vulnerabilities, such as potential system-wide failures and a lack of responsiveness to localized traffic conditions. These methods also have varying degrees of state awareness, that range from features across entire road grids to single intersections. As the central manager has a wider influence its theoretical potential to be more efficient increases but it also becomes less realistic due to the limitations of communication, network latency, information access across intersections, and robustness to system failures.

1.2 DISTRIBUTED MODELS

Conversely, distributed systems enhance local traffic decision-making, offering greater resilience to systemic failures and agility in responding to immediate traffic changes. Examples of these systems include: Dynamic request-response protocols for conflict resolution of vehicles [13], Spatio-Temporal Intersection Protocol (STIP), where an FCFS priority is used with communication between nearby vehicles [14]; Model-Predictive-Controller (MPC) based solutions, where vehicles share trajectories and control [15]; broadcast methods that communicate to all nearby vehicles arrival and departure times [16]; BATCH scheduling, that processes batch vehicle requests to form platoons of vehicles moving in the same direction [17] etc. The trade-off of distributed systems lies in their potential lack of overarching coordination which can lead to significant inefficiencies on a larger scale. This is most noticeable when compared to theoretical models where central managers such as network-wide traffic controller systems can outperform distributed systems through greater knowledge of the overall road state [18]. The full integration of V2X communications also requires standardized communication protocols which may pose a challenge due to the lack of incentives and competitive economic landscape of the current automotive industry.

1.3 VEHICLE CONTROLLERS

When vehicles are the primary agents, the system allows for direct and precise management of traffic, particularly with the integration of autonomous vehicles. This approach, however, is contingent on the widespread adoption of such vehicles, which remains a future prospect. The challenge is in integrating conventional vehicles within this framework, potentially limiting its immediate applicability. In contrast, systems where traffic lights act as agents align more closely with existing road infrastructure, offering a more feasible option for current urban settings. These systems, though, might lack the dynamism and efficiency in high-congestion scenarios that vehicle-centric approaches promise.

1.4 HYBRID SYSTEMS

Hybrid systems, combining elements of both centralized and distributed models, and incorporating both traffic lights and vehicles as agents, offer a comprehensive solution. They aim to balance localized responsiveness with city-wide traffic management. The complexity of these systems, encompassing intricate technology and coordination, presents a notable challenge, particularly in terms of implementation and maintenance costs.

In the current context of urban transportation, systems with centralized traffic light controllers, decentralized across each intersection appear to be the most pragmatic initial step. This approach complements existing traffic infrastructure and the realistic integration of human-CAV systems that can be progressively enhanced to integrate more sophisticated elements over time. As urban landscapes evolve and autonomous vehicles become more prevalent, it is conceivable that a transition towards more integrated, hybrid systems will occur.

Our project aims to evaluate and improve an autonomous traffic intersection system, specifically employing traffic lights as agents and leveraging reinforcement learning, due to its strong potential for practical implementation in real-world scenarios. After our initial investigation of various autonomous traffic management systems, we have determined that this would be the most effective in its feasibility and solving integration challenges to improve urban traffic flow, safety, and environmental sustainability. Through this endeavor, we seek to offer a tangible solution to enhance the efficiency of urban traffic intersection systems. We attempt to improve upon current methods of autonomous traffic intersection management systems that can be realistically implemented, bridging the gap between theoretical innovation and its practical application.

2. PROBLEM DEFINITION

Our problem space will be dealing with the traffic flow of a single 4-way intersection system. In our preliminary investigation, we implemented and tested several methods including the AIM approach and distributed vehicle based controllers using FLOW [20] that did not utilize traffic lights. Despite being very efficient, we found these systems to be unlikely outside of theoretical models in the near future as they require full CAV integration.

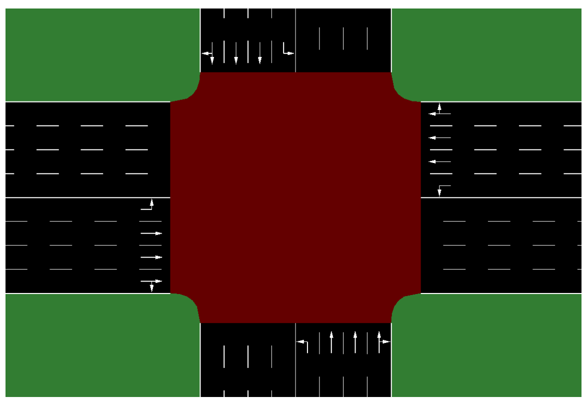

Figure 1. Traffic light controlled intersection with 4 750-meter arms and 4 incoming lanes in each direction.

On the other hand, almost all major intersections are controlled by traffic lights. While traffic light controlled intersections have different configurations of various complexities, we want to define a single configuration to better investigate the efficiency of strategies we want to implement in this problem space. To do so, we utilized Simulation of Urban MObility (SUMO) to simulate traffic flow that occurs in a normal intersection. Using starter code from this github repository, we define an intersection with traffic light controls as shown in Figure 1, where each arms are perpendicular to each other and are 750 meters long. There are 4 lanes in the direction that approaches the intersection in each arm, where the leftmost lane is the left turn only lane, the middle 2 lanes are straight ahead only lanes, and the rightmost lane is for straight and right turns. We also define there are 4 possible phases of traffic lights of which are symmetric phases of left-turn-only and straight-only for the two horizontal directions and the same for the two vertical directions.

According to [22], a Weibull distribution approximates traffic demand during rush hour, where the number of cars rises quickly until peak traffic, and then slowly decreases afterwards spanning 5400 time steps. In SUMO, each time step is equivalent to 1 second. We generated 1000 vehicles at the end of each incoming arm using a Weibull distribution with a shape equal to 2, which is depicted in Figure 2. Each generated car is placed randomly at an incoming edge, where 75% of them goes straight and 25% turns left or right equally probable at the intersection. We used TraCI to gather statistics of a baseline run of the simulation using this set up. In the baseline run, the traffic lights act in their SUMO default strategy, which cycles through each light phase, where each green light lasts for 10 time steps and yellow for 4 time steps.

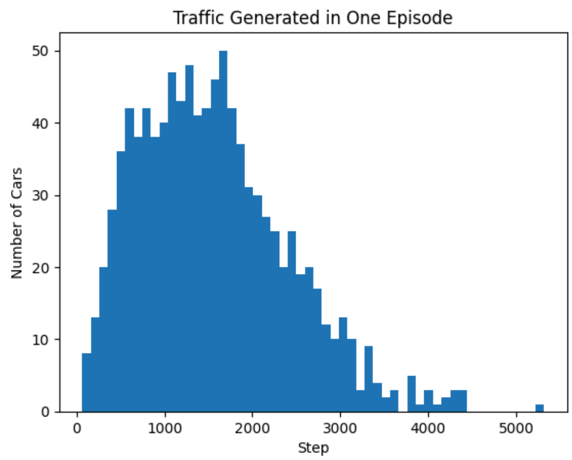

Figure 2. Traffic generated in the test episode using shape 2 Weibull distribution. Bin width = 100 steps.

We define the following STL formula:

$$\varphi \equiv G_{[0,5400]}(F_{[0,\,64]}(QL(t)) \leq Incoming(t)), \tag{1}$$

where $QL(t)$ is the queue length of vehicles at the intersection at time t, and Incoming(t) is the number of cars generated on incoming lanes at time t. φ is satisfied if the number of vehicles waiting in queue at a red light is less than or equal to the number of cars being generated within a full light cycle of the intersection. We calculated the sum of queue lengths at each timestep as followed:

$$QL(t) = \sum_{i=1}^{4}\sum_{n=1}^{4} H_{i,n}(t) \tag{2}$$

where $H_{i,n}$ is the number of halting vehicles at lane n at arm i, where there are 4 lanes in each arm and 4 incoming arms. Theoretically, if φ is satisfied, there will be minimal to no traffic congestion, since the number of cars halting being less than the number of cars introduced to the network within a short time frame means they have already passed the intersection and left the network.

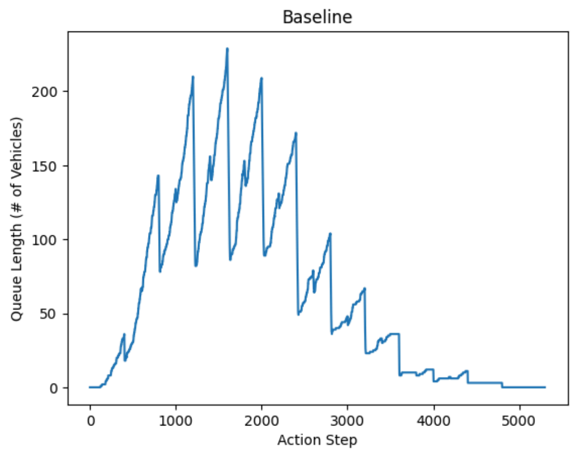

Figure 3. Simulated traffic using the problem definition.

With zero optimization, we can see that the overall shape of queue length resembles the traffic demand distribution, that is, the more traffic that is coming into the intersection, the more congested it is. As seen from figure 3, the maximum queue length, the number of vehicles stopped at a red light, reaches over 200 at a point at peak traffic. We computed a robustness value of -186 for the baseline simulation, meaning the STL formula is not satisfied, and hence there is significant wait time for drivers and poor traffic throughput and efficiency at this intersection. With this in mind, we want to find solutions that can improve the average queue length at this intersection. Formally, our objective now becomes to reduce the queue length at this intersection to satisfy the STL formula in the given problem space.

3. SOLUTION STRATEGY

There are many ways to improve the traffic flow through an intersection, most of which involve adaptive methods to change the phase of the traffic lights. Many traffic lights are adaptive [23], where they adjust the timing of red, yellow and green lights to accommodate changing traffic patterns and ease traffic congestions. They observe traffic patterns using presence sensors positioned along the roadway to gauge traffic flow and react to it using pre-defined strategies. While these methods are effective in some cases, their ability to improve traffic flow is limited to their designers, and is by nature unable to handle all possible traffic scenarios. To come up with robust algorithms that can ease traffic congestion also require extensive study of traffic patterns in and around the location of interest. Since we have no academic background on optimizing traffic flow, we want to solve this problem with methods that we are more familiar with. Namely, we want to train machine learning models to act as a traffic light agent to cope with different traffic scenarios.

While there are many machine learning models that can accomplish such tasks, we narrowed down our focus on improving an adaptive traffic light system with reinforcement deep Q learning, as suggested by [22]. A reinforcement learning agent is perfectly suitable for this problem because the agent can improve simply based on the reward function we define and act on a given state of the intersection without having expertise in the complexities of vehicle traffic flow. An RL agent is also infinitely scalable as long as the dimensions of the input state and output actions are compatible. With traffic simulators like SUMO, we can train the RL model as much as our computational resources permit. In the case of an adaptive traffic light using a deep Q RL model, our agent can be as intelligent as we want or as simple as we want.

To build an RL agent, we first define the state space of the intersection by utilizing the existing presence sensors. Each incoming edge is divided into 20 zones, and each zone can be identified as occupied or not. This translates to a total of 80 different state variables for a 4-way intersection. Note that we can change the state representation to fit different configurations of roadways, as long as we match the model inputs. The RL agent is able to control the traffic light phase via TraCI with 4 possible actions corresponding to each light phase. We implemented a minimum limit of 10 timesteps in between each act from the agent, meaning that if the agent changes the light phase, it has to wait for at least 10 seconds before it can change again, if it desires to. The 10 second limit corresponds to the minimum duration of a green light. As shown in Figure 4, the deep Q model consists of 4 hidden dense layers, takes an input vector of length 80, and outputs an integer between 0 and 3 representing one of the 4 possible actions.

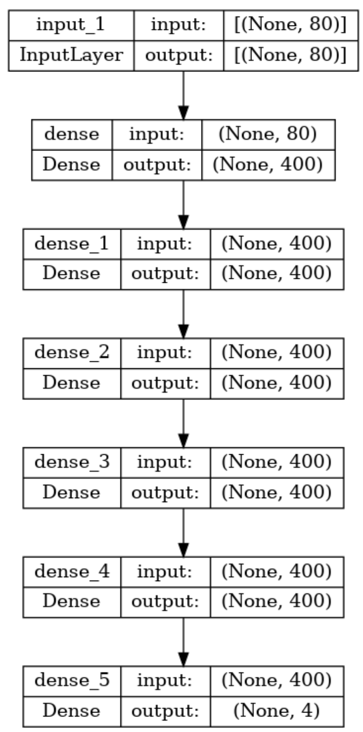

Figure 4. Q-learning model structure: deep neural networks.

Due to limited resources and time, we investigated the performance of select reward functions for the RL model. One option for the reward function is based on the average speed of vehicles on the incoming lane:

$$R_1(t) = AvgSpeed(t) - AvgSpeed(t-1) \tag{3}$$

If the average speed of all vehicles in the current time step is greater than the average speed of the previous step, this creates a positive reward. The model is then trained to take actions that can maximize the reward. If the model can maintain a higher average speed, that means the vehicles are having to stop less and thus generate a higher throughput. Another option for reward function is to calculate the total wait time, which is the sum of duration that a vehicle is stationary in the simulation:

$$R_2(t) = TWT(t - 1) - TWT(t) \tag{4}$$

If the total wait time of the previous step is greater than the total wait time of the current step, the reward is positive. If the model can maintain a low total wait time during the simulation, this means there are less vehicles stationary, or that they are stationary for less time, thus increasing the traffic throughput.

Another approach that we tried was to include more input information passing the number of vehicles in each detection zone in the input state space in addition to presence sensing. All of these approaches yield different models with varying performance, which we will discuss in the following section.

4. RESULTS

We trained 3 different models for this paper. Model 1 uses the reward function given by equation (3) trained on information of presence of cars in each detection zone. Model 2 uses the reward function given by equation (4) trained on information of presence of cars in each detection zone. Model 3 uses the reward function given by equation (4) and acquired information of the count of cars in each detection zone. We initially trained the models with 100 episodes with training epochs of over 100, but our computer crashed around episode 60. Due to limited computational power, we are only able to train the deep Q learning models with the following parameters:

Total Episodes	Width Layers	Batch Size	Learning Rate	Training Epochs	Memory Size Min	Memory Size Max
50	400	100	0.001	30	600	50000

Table 1. Parameters used for training.

Below are the training performance of models using different reward functions and input states with limited training episodes and epochs. We anticipate the models to perform poorer than they can due to these parameter constraints. Nevertheless, we can still see improvements in agent performance in higher training episodes. It is important to note that it is apparent that equation (4) is a better candidate for reward function when compared to equation (3). As shown in Figure 5, the cumulative reward using average speed initially improves from episode 0 to around episode 30, but drastically decreases in later episodes. We suspect this is due to the traffic configuration where there are less cars introduced towards the end of the simulation.

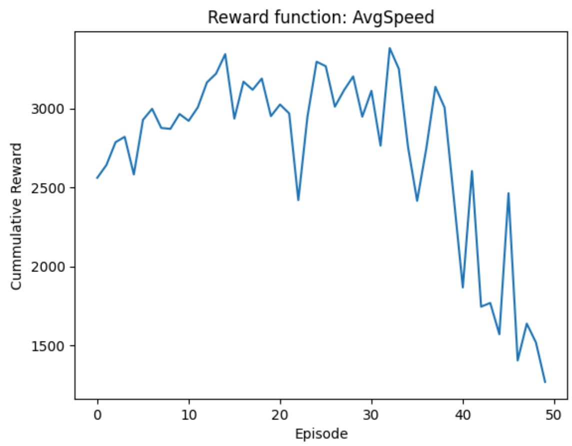

Figure 5. Training rewards for the model using reward function (3).

There is no significant difference in training reward trend between the two models using different input information, as suggested in Figure 6. While both models have weird spikes in their cumulative reward graphs, the general performance is comparable, and they are expected to keep improving with more training episodes as there is no sign of plateauing.

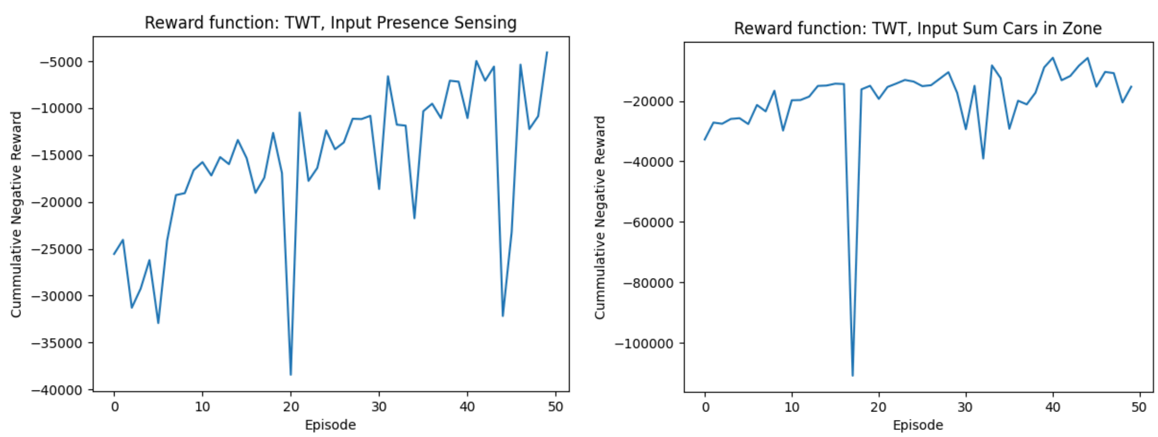

Figure 6. Training rewards for models using reward function (4). Agent in the first graph only takes in information on the presence of traffic, while the agent in the second graph counts the number of cars in each zone.

However, the performance of these agents differ drastically during the test episode. To ensure the results are comparable, the traffic generated according to the Weibull distribution during the test episode uses the same pseudo random seed. The baseline run also uses the same seed, so the results are directly comparable. To our surprise, although model 1 performed worse than model 3 during training, it outperformed model 3 during the test run. Model 1 also significantly improved traffic congestion compared to baseline, where the maximum queue length throughout the episode peaks at around 50. We analyzed its trace and computed it to have a robustness value of -23 for the STL formula (1), which is a significant improvement from baseline. Model 2 performed the best out of the three, with a maximum queue length of 17 throughout the run, and a calculated robustness of 29. Model 3 performed the worst in the test run, with a robustness of

-32. We suspect the poor performance of model 3 is due to overfitting during training. Coupled with the deficient training episode number, it learns information of the number of cars in each detection zone and then makes optimized decisions that are highly biased towards the specific traffic generations in those limited training episodes.

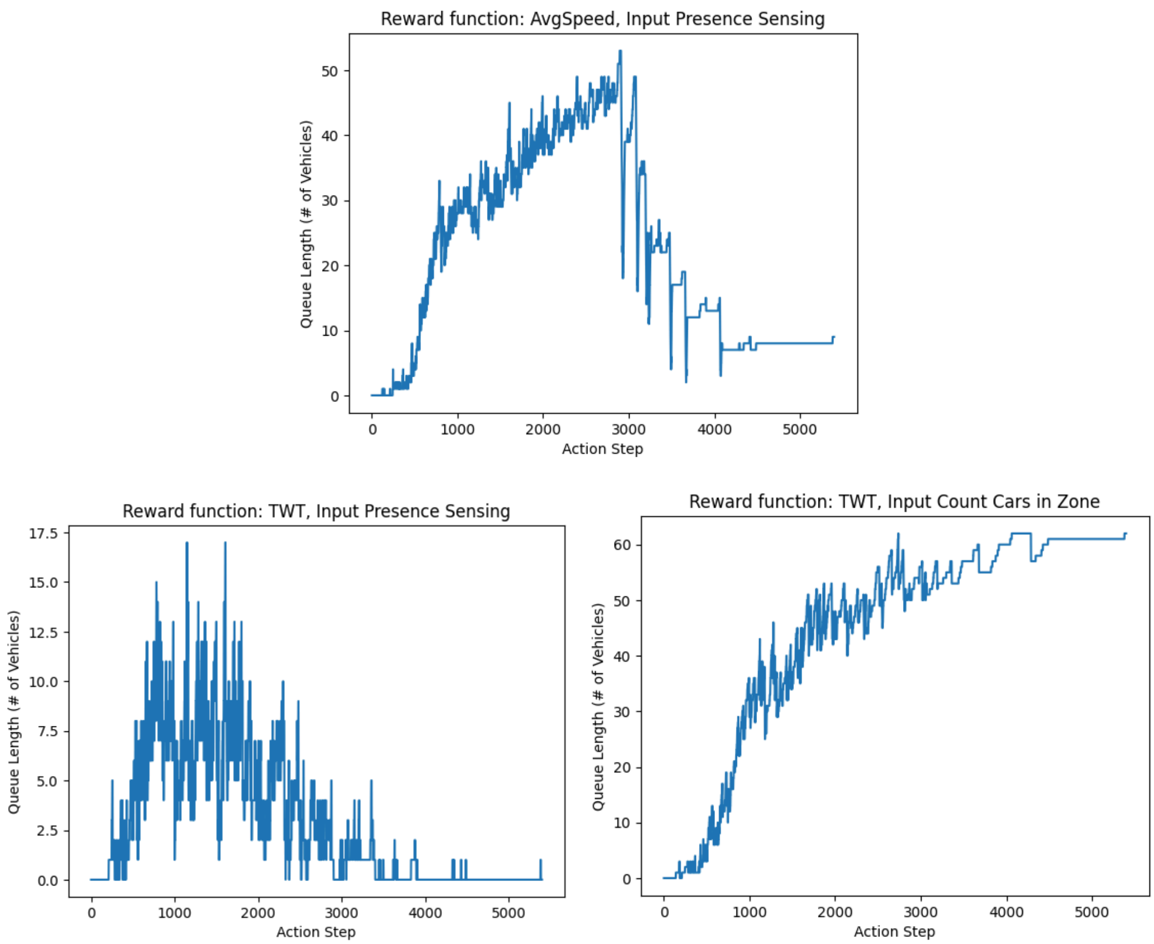

Figure 7. Queue length in test episodes using different agents.

Even though model 1 does not perform nearly as good as model 3, both of these models are improvements over the baseline. These results suggest that we may improve model performance in test runs by incorporating more complex reward functions that consider multiple state variables.

5. DISCUSSION

In our investigation, we observed that the no traffic light method, while highly efficient in theory, presents significant practical challenges, particularly in mixed traffic scenarios. Its reliance on highly sophisticated vehicle-to-vehicle communication and real-time data processing makes it less feasible in the short term, especially in environments not exclusively composed of CAVs. Since SUMO simulation does not allow for collisions, we did not measure the robustness of safety of such a system. In a system where safety is always prioritized over efficiency, this implies it would be extremely difficult to audit for safety before the system can be open for use to the public.

The traffic light method enhanced with reinforcement learning offers a more realistic approach. It adeptly manages both autonomous and human-driven vehicles and is adaptable to be extended with other autonomous systems. From our experimental results, we can see the effectiveness of traffic flow improvement with an RL agent trained on a simple reward function. However, its effectiveness is also contingent upon the precision and reliability of real-time data processing and network latency. The scope of our project also only involved improving traffic flow without considerations of pedestrian movement across these intersections, which is a significant factor especially in densely populated and walkable cities. The reinforcement learning method may reduce traffic outflow at the expense of creating excessive pedestrian wait times or having the inability to properly cross the road on time. An area for future enhancement could be to include pedestrian dynamics into the traffic management system, thereby enriching the model's applicability. Theoretically integration of broader data sets, including environmental conditions and traffic patterns from neighboring intersections, could further optimize traffic flow predictions and light phase adjustments. This system also has the drawback that processing larger amounts of data from a larger number of agents can be computationally expensive and can increase the challenges brought from networked communications, latency, and sensor reliability. Overall, we found that our approach suggests a potential first step towards an efficient traffic management system, where data from various sources are utilized to create a more fluid and responsive traffic control environment.

It is worth noting that there are many different intersection configurations and traffic flow distributions in the real world. For simplicity, we have only investigated one distribution on one intersection configuration. Our model is only trained on the given problem space, and it is not guaranteed or expected to perform better than baseline on a different intersection configuration or different traffic flow models. Nevertheless, with enough computational resources and statistical studies, it is entirely possible to train a model that can handle different traffic flows on different intersection configurations, or to implement models with different training and learning parameters on different intersection locations. This remains a topic to be investigated in the future.

6. CONCLUSION

Our project, navigating through the intricate dynamics of autonomous traffic intersection management, investigates various autonomous system methods and works towards not only implementing a model but also discussing the feasibility of its implementation. We have found that while traffic light-less systems exhibit theoretical efficiency, their practical deployment in mixed traffic environments is fraught with challenges. The incorporation of reinforcement learning into traffic light systems emerges as a more viable and adaptable solution, adeptly balancing the intricacies of both autonomous and human-driven vehicles while also providing significantly improved traffic flow over the baseline. However, there are limitations and future work that can be done to explore the integration of pedestrian dynamics into the simulation, testing real-world networked interfaces, and building systems with broader shared state spaces across traffic light controllers in proximity. This project maps out the present landscape of urban mobility and subtly hints at the potential of realistic future intersections, where advanced technology can harmonize with existing infrastructure, leading to a more optimized and intelligent transportation ecosystem.

7. REFERENCES

[1] Angus Eugene Retallack and Bertram Ostendorf.. 2019. Current Understanding of the Effects of Congestion on Traffic Accidents. 16, 18 (September 2019).. DOI:https://doi.org/10.3390/IJERPH16183400

[2] Isa Ravanshadi, Elham Amini Boroujeni, and Mahdi Pourgholi.. 2022. Centralized and distributed model predictive control for consensus of non-linear multi-agent systems with time-varying obstacle avoidance. 133, (June 2022).. DOI:https://doi.org/10.1016/j.isatra.2022.06.043

[3] Mohammad Khayatian, Mohammadreza Mehrabian, Edward Andert, Rachel Dedinsky, Sarthake Choudhary, Yingyan Lou, and Aviral Shirvastava.. 2020. A Survey on Intersection Management of Connected Autonomous Vehicles. 4, 4 (August 2020).. DOI:https://doi.org/10.1145/3407903

[4] Kurt Dresner and Peter Stone.. 2008. A multiagent approach to autonomous intersection management. (March 2008).. DOI:https://doi.org/10.5555/1622655.1622672

[5] Zijia Zhong, Mahyar Movahed Nejad, and Earl E. Lee.. 2020. Autonomous and Semi-Autonomous Intersection Management: A Survey. (June 2020).. DOI:https://doi.org/10.1109/MITS.2020.3014074

[6] Mohammad Khayatian, Yingyan Lou, Mohammadreza Mehrabian, and Aviral Shirvastava.. 2019. Crossroads+: A Time-aware Approach for Intersection Management of Connected Autonomous Vehicles. 4, 2 (November 2019).. DOI:https://doi.org/10.1145/3364182

[7] Kaidi Yang, S. Ilgin Guler, and Monica Menendez.. 2016. Isolated intersection control for various levels of vehicle technology: Conventional, connected, and automated vehicles. 72, 72 (November 2016).. DOI:https://doi.org/10.1016/J.TRC.2016.08.009

[8] Myungwhan Choi, Areeya Rubenecia, and Hyo-Hyun Choi.. 2019. Reservation-based traffic management for autonomous intersection crossing. 15, 12 (December 2019).. DOI:https://doi.org/10.1177/1550147719895956

[9] Q. Jin, G. Wu, K. Boriboonsomsin and M. Barth, "Platoon-based multi-agent intersection management for connected vehicle," 16th International IEEE Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, Netherlands, 2013, pp. 1462-1467, doi: 10.1109/ITSC.2013.6728436.

[10] Weiming Zhao, Dong Ngoduy, Simon Shepherd, Ronghui Liu, and Markos Papageorgiou.. 2018. A platoon based cooperative eco-driving model for mixed automated and human-driven vehicles at a signalised intersection. 95, (October 2018).. DOI:https://doi.org/10.1016/J.TRC.2018.05.025
[11] Yanfeng Geng and Christos G. Cassandras.. 2015. Multi-intersection Traffic Light Control with blocking. 25, 1 (June 2015).. DOI:https://doi.org/10.1007/S10626-013-0176-0
[12] Yuksel Celik and Alper Talha Karadeniz.. 2018. Urban Traffic Optimization with Real Time Intelligence Intersection Traffic Light System. 6, 3 (September 2018).. DOI:https://doi.org/10.18201/IJISAE.2018644780
[13] S. Aoki and R. Rajkumar, "Dynamic Intersections and Self-Driving Vehicles," 2018 ACM/IEEE 9th International Conference on Cyber-Physical Systems (ICCPS), Porto, Portugal, 2018, pp. 320-330, doi: 10.1109/ICCPS.2018.00038.
[14] R. Azimi, G. Bhatia, R. R. Rajkumar and P. Mudalige, "STIP: Spatio-temporal intersection protocols for autonomous vehicles," 2014 ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS), Berlin, Germany, 2014, pp. 1-12, doi: 10.1109/ICCPS.2014.6843706.
[15] A. Katriniok, S. Kojchev, E. Lefeber and H. Nijmeijer, "Distributed Scenario Model Predictive Control for Driver Aided Intersection Crossing," 2018 European Control Conference (ECC), Limassol, Cyprus, 2018, pp. 1746-1752, doi: 10.23919/ECC.2018.8550526.
[16] Li Li and Fei-Yue Wang.. 2006. Cooperative Driving at Blind Crossings Using Intervehicle Communication. 55, 6 (November 2006).. DOI:https://doi.org/10.1109/TVT.2006.878730
[17] Remi Tachet, Paolo Santi, Stanislav Sobolevsky, Luis Ignacio Reyes-Castro, Emilio Frazzoli, Dirk Helbing, and Carlo Ratti.. 2016. Revisiting Street Intersections Using Slot-Based Systems. 11, 3 (March 2016).. DOI:https://doi.org/10.1371/JOURNAL.PONE.0149607
[18] Zhenning Li, Hao Yu, Guohui Zhang, Shangjia Dong, and Cheng-Zhong Xu.. 2021. Network-wide traffic signal control optimization using a multi-agent deep reinforcement learning. 125, (April 2021).. DOI:https://doi.org/10.1016/J.TRC.2021.103059
[19] Dongjiang Liu and Leixiao Li.. 2023. A traffic light control method based on multi-agent deep reinforcement learning algorithm. 13, 1 (June 2023).. DOI:https://doi.org/10.1038/s41598-023-36606-2
[20] C. Wu, A. Kreidieh, K. Parvate, E. Vinitsky, A. Bayen, "Flow: Architecture and Benchmarking for Reinforcement Learning in Traffic Control," CoRR, vol. abs/1710.05465, 2017. [Online]. Available: https://arxiv.org/abs/1710.05465
[21] Martin Treiber, Ansgar Hennecke, and Dirk Helbing.. 2000. Congested traffic states in empirical observations and microscopic simulations. 62, 2 (August 2000).. DOI:https://doi.org/10.1103/PHYSREVE.62.1805

[22] A. Vidali, L. Crociani, Giuseppe Vizzari, S. Bandini.. 2019. A Deep Reinforcement Learning Approach to Adaptive Traffic Lights Management. Retrieved from https://www.semanticscholar.org/paper/A-Deep-Reinforcement-Learning-Approach-to-Adaptive-Vidali-Crociani/9c1abe22641981f9caadb8074c7878ef7bb9d81a

[23] 2023. EDC-1: Adaptive Signal Control Technology | Federal Highway Administration. Retrieved December 13, 2023 from https://www.fhwa.dot.gov/innovation/everydaycounts/edc-1/asct.cfm

8. APPENDICES

8.1 NO TRAFFIC LIGHT METHOD WITH FLOW

FLOW utilizes SUMO + TraCI to create a benchmarking framework that allows to easily integrate and build vehicle network simulations and apply reinforcement learning agents. In the preliminary exploration of our project, we focused on creating a simple 1-lane 4-way intersection network with different routes. In the experiment setup, the vehicle inflows had pre-planned trajectories with equal probability for each route. The vehicle model used was the Intelligent Driver Model (IDM) Controller developed by Treiber et al. [2000] and the default SUMO intersection collision avoidance algorithm [21]. For the experiment we defined one with and without actuated traffic lights and recorded the resulting average vehicle velocities and outflows. The parameters were set where each route had a distance of 50, an inflow probability of 10% per second at every start point, for a total of 3000 time steps.

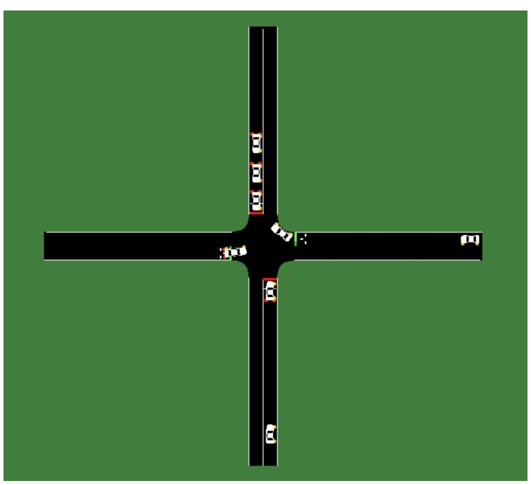

Figure 8 - FLOW Actuated Traffic Light Network

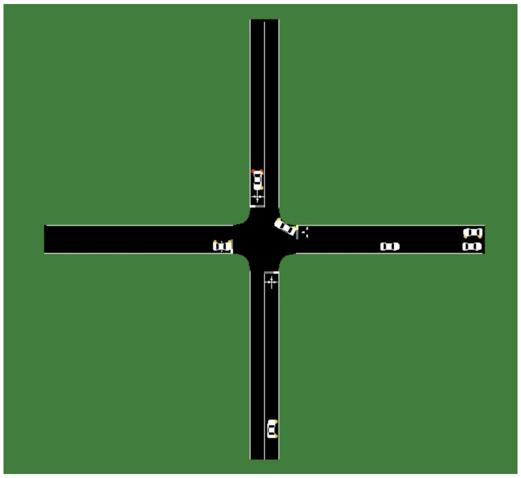

Figure 9 - FLOW No Traffic Light Network

	Actuated Traffic Lights	No Traffic Lights
Average Outflow	1068	1224
Average Velocity	2.50	4.34

Figure 8.2.3 - FLOW Experiment Results Table

8.2 GITHUB REPOSITORY

https://github.com/cheekw/traffic-intersection-management