Traffic Light Controller using Reinforcement Learning

Mausumi Bhuyan (122201001)
Sathwika Goli(122201005

October 2024

I. Abstract

The complexity of urban traffic management systems is high and is going higher with time, and the problem with it is that these increasing complexities require adaptive and intelligent control systems. Traditional controllers of traffic lights fail to optimize real-time traffic flow, and as a result, the traffic sometimes remains congested and delayed. In this project, we try to optimise traffic flow at the intersections using Reinforcement Learning (RL), which optimizes signal timings by learning from its environment. we will create a simulated traffic environment where a Reinforcement Learning agent interact with the environment to minimize waiting times for vehicles, congestion, and travel delays . We will use the SUMO (Simulation of Urban Mobility) traffic simulator or Gazebo for simulating the environment .

The state of the traffic at an intersection, such as the number of cars in each lane, was observed, and optimal traffic signal phases that maximize traffic throughput were selected by the agent. The state-action-reward system will be modeled by the help of Q-learning and trained such that an agent makes decisions to minimize overall traffic congestion over time. Overall, the performance of RL-based traffic control is compared against traditional fixed-time traffic signal systems with significant improvements observed in vehicle wait time and the overall efficiency in traffic flow.

II. Objectives

Our objective is to design a reinforcement learning-based traffic light controller to manage traffic lights at a single four way intersection with four lanes per road(west,east,north and south) to reduce congestion and overall traffic flow. Implement a reward mechanism within the DQN model that minimize the average waiting time at the intersection.Test the DQN controller under changing traffic conditions(light, medium, and heavy) to see whether it could adapt well Develop an efficient DQN model that can produce timely decisions, good for actual deployment.This project intends to optimize the traffic with the inclusion of lane-specific data into the state representation flows more minutely, refining changes in signals according to real-time conditions of each lane

III. Motivation

This is a critical need to develop a fresh, creative approach to address the inefficiencies present in contemporary urban transportation networks. As the number of people living in cities increased and the increased usage of cars, this is especially important. Intersections are often becoming places of extreme traffic, more extended waiting periods and safety risks. Because they follow preset schedules and don't constantly adjust to changing traffic patterns, the typical traffic signal systems in use today are essentially static.

In order to create an intelligent traffic light control system, this system will use reinforcement learning. We will create an efficient system that will help to improve smooth flow for both vehicles and pedestrians based on a real-time decision with live traffic data.

IV. State-of-the-art

The fixed-timing systems of traditional traffic light control are based on regular timetables or historical traffic data, Such systems are easy to implement but are not responsive to real-time variations in traffic, which in practice will create under-utilization during off-peak hours and congestion during peak hours. An emerging technique in this adaptive traffic control is reinforcement learning, where trial-and-error functions become a means for training algorithms as to which course of actions maximizes long-term benefits. One of the popular techniques of RL are Q-learning: it enable systems controlling traffic signals to adopt those policies that will minimize waiting times on average and adapt according to changes in vehicle density. For instance, Smith et al. (2020) showed the ability of Q-learning to perform well in reduction of delay at single intersections by providing evidence of the same. This makes it challenging since typical in complex traffic environments is the state and action space with high dimensionality that can be hard to deal with for traditional Q-learning. Deep Q-Networks are advanced forms of Q-learning proposed to solve such issues, and these are allowing a method to work under conditions of more complicated situations or to deal with greater areas by approximating Q-values that can be used for different states and different types of actions with deep neural networks. Because DQNs can take into account additional variables such as the vehicle positions and queue lengths in real-time.Recent studies—in line with some by Park et al. (2022)—have demonstrated their effectiveness within dynamic traffic management. These investigations have shown that controllers based on DQN excel beyond conventional systems Despite all this advancement, there are still several issues. Due to their high computational demands, DQNs could not be applied to real-time environments of resource-constrained environments. Besides, DQN-based models cannot change according to the real-time variations of traffic flowing through multiple lanes and directions because of their hyperparameter sensitivity.

V. Methodology

Using a DQN allows for adaptive control as it will continuously learn the patterns of traffic. This has the potential to reduce congestion and improve overall traffic efficiency.

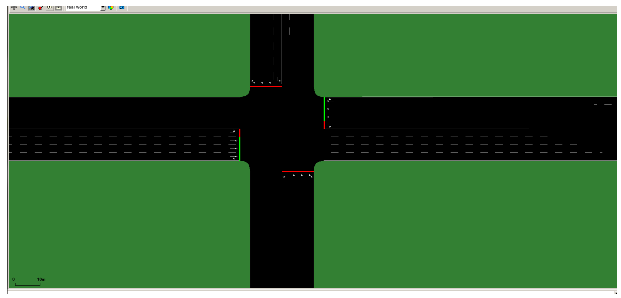

Figure 1: environment image

The DQN model is built as a deep neural network. In this architecture, it is basically an input layer followed by several layers of hidden units and finally by the output layer. The input layer consists of 80 neurons that depict the present state of the intersection. The state may include vast amounts of traffic parameters, such as queue lengths, waiting times, and other circumstances on roads, so that it depicts the environment in which the agent is functioning.

It uses six layers of dense. The first one takes the input of 80 neurons, expands it to 400 neurons, and captures all the complexities in traffic conditions. The remaining five layers are kept at the dimension of 400 neurons each, giving a deep representation of traffic conditions and allowing the model to learn data at intricacies through the relationships. The same size layers in the hidden layers will increase the capacity for the model to generalize and handle various traffic scenarios without getting overfitted to any specific situation.

The output layer, the final layer, is composed of 4 neurons corresponding to the four possible actions or signal phases that the agent can make. Each of the possible actions is one out of several traffic light arrangements, for example, a green light for north-south traffic or a green light for east-west traffic. This means the choice the agent has to make about leaving or clearing certain parts of the intersection is an integer decision, and it makes the agent decide on one of its discrete choices in the management of the flow at the intersection, thereby allowing it to learn an efficient policy mapping different states to optimal actions. An agent, the DQN agent, is trained in a simulated traffic environment through the SUMO (Simulation of Urban MObility) framework. SUMO represents an open-source traffic simulation tool that allows for the modeling of traffic dynamics under controlled conditions. To make sure the agent learns in nearly realistic environments, the simulation is tuned with realistic vehicle flows and intersection layouts coupled with parameter settings for roads.

We set up a custom reward function to train the DQN: rewarding a decrease in congestion and wait times at the intersection. We have queue length, waiting time, and counts of emergency braking events as the components of the reward function. A penalty to the agent has to suffer large queue lengths in order to incentivize actions that efficiently clear the traffic. It also suffers penalties with respect to waiting times of cars in order to encourage reductions in delay. To avoid jerky braking that may lead to accidents and inefficiencies, the agent is further penalized for events that cause abrupt braking. These reward components, thus, bring balance between the need to have smooth flow and those which ensure safety considerations as the model learns on them to achieve optimal traffic signal control decisions. The model follows during training an ϵ-greedy policy that explores new actions with probability while reducing it gradually so as to favor exploitation of learned actions as training progresses. Training is performed over multiple episodes, each representing a simulation run of the intersection for a fixed time period. During an episode, the metric values such as cumulative reward, total delay, average queue length, and average waiting time are recorded so that the performance of an agent over time can be evaluated.

VI. Results and discussion

Key Performance metrics for the DQN-based traffic controller are total reward, cumulative delay, queue length, average waiting time, and emergency braking events. These metrics altogether imply the capability of the agent in bringing forth optimal improvement in traffic flow through that intersection. The graph of the total reward throughout the training episodes continuously increases. This means that, without fail, a DQN agent learns progressive actions improving traffic efficiency. In such a manner, the upward tendency of total reward verifies that the correct alignment of the agent with the goals assigned by the reward function, as congestion reduction and waiting time minimumization, can lead to an adaptation-friendly traffic control policy. The cumulative delay, which is the total waiting time for all the vehicles in the simulation, decreases as the training progresses, thus proving that DQN can actually deal with the flow of traffic very well. It can be observed in terms of a decrease in delay, that suggests the agent is minimizing idling time and is playing a role in alleviating congestion. The number of vehicles waiting at the intersection as indicated through the queue length metric is also decaying, suggesting the agent can clear all the waiting vehicles before congestion builds up in considerable amounts. Again, this rules out bottlenecks in traffic flow and improves overall flow of traffic. we can observe them through simulation.

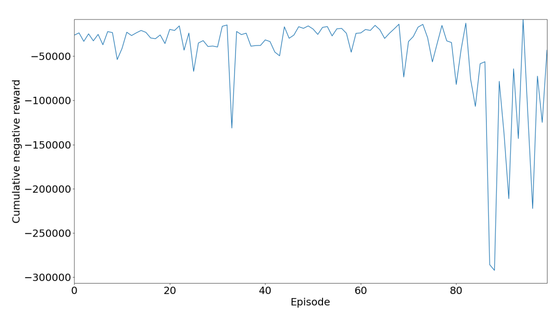

Figure 2: reward plot

Cumulative negative reward is decreasing ,this implies that positive reward is increasing.

Furthermore, the average waiting time per vehicle decreases over time with training; this corresponds to the agent's ability to reduce delays for individual vehicles. Further work might even formulate a sophisticated reward function that punishes such events or imposes some constraints against abrupt signal changes to eventually result in a smoother and safer traffic control system.

VII. Conclusion

Thus, the proposed DQN-based traffic light controller significantly improved the management of intersection traffic flows, which includes the cumulative delay, queue length, and average waiting time. This model is basically capable of learning the adaptive response policy towards real-time conditions with maximal signal timing and alleviates congestion. However, some emergency braking events during the test or case studies might imply possibilities for more design improvements. In general, though, the DQN approach shows very promising potential in the enhancement of traffic efficiency and offers a more seamless and responsive solution for the management of urban traffic.

VIII. Future Perspective

Key suggestions for further development to add value to the effectiveness and scalability of this system in the architecture of Reinforcement Learning-based Traffic Light Control are as follows: Multi-Agent Coordination. It will extend the model towards a multi-agent framework to accommodate large networks of intersections This allows the sharing of information among controllers and collaborative optimization of traffic flow across the network of multiple intersections. In the event of dealing with complexities in real life, this kind of coordination is significant in traffic load for dense urban areas. Integrating real-time traffic data from various IoT sensors, cameras, and connected vehicle data would significantly enhance the adaptability of the model. The system would then be better positioned to react more quickly to emergent changes in traffic conditions like congestion spikes or traffic incidents. Scalability and Computational Efficiency: Scaling is another important issue, because the system continues to grow. Techniques such as edge computing or distributed processing may be used to lessen the computational requirements a city-wide traffic control system so that such systems will be responsive without having high computational powers at each intersection.

Enhanced Safety and Handling Rare Events: Safe RL algorithms are developed with the capability of handling unexpected events, like emergency vehicles, pedestrian crossings, or road construction. Techniques might include robust RL or meta-learning such that the model could adapt more rapidly to the situation where unusual events occur.

IX. References

- https://link.springer.com/chapter/10.1007/978-3-031-48573-2$_7$7
- https://github.com/traffic-signal-control/RL$_{s}ignals$
- https://arxiv.org/abs/2302.03669
- https://www.youtube.com/watch?v=SK2G1z8upoU
- https://www.youtube.com/watch?v=fIHOQNhvOu4
- https://github.com/mayankiitg/TrafficLight-RL
- reference picture