Optimizing Multi-Agent Reinforcement Learning for Adaptive Traffic Control

by Evan Liu (evanliu) & Debnil Sur (debnil)

Motivation

Transportation systems form the backbone of any country's economic vitality and fulfills critical security objectives. Bottlenecks in their performance therefore threaten significant stresses on economic growth and can hurt responses in times of crisis. In America, traffic signal timing is estimated to cost billions as a drag to national commerce per year. As a result, artificial intelligence techniques have been tested in simulation environments and deployed at certain traffic lights with the hope of reducing delay at critical junctions. Scaling could alleviate stress on the entire network.

Project Scope

Thus, we hope to generate a heuristic for coordination in the traffic system that can minimize delay times compared to similarly motivated recent work. The most successful results in the field of late have modeled the traffic system as a multi-agent Markov Decision Process and applied Reinforcement Learning techniques [2]. We will follow a similar approach and hope to improve upon the results of this work. To do this, we will model a small traffic network of nine lights in a 3x3 grid according to the literature [2]. We will train our model with SUMO, an open source traffic simulator used in similar research projects.

Challenges

Traffic signal optimization presents several key challenges. Two significant ones are optimization of a multi-agent system and constraining the state space. First, it is not sufficient to greedily optimize a single traffic light, as this will decrease waiting times for that single light, but could potentially have no effect on or even increase the overall waiting time of the traffic light network. As a result, traffic lights must be optimized in significantly large enough batches to coordinate the light timings. We address this issue by modeling the traffic light network as a multi-agent system, where an agent controls each traffic light. This ensures that the entire traffic system will be optimized, rather than just a single light within the system. Second, the state space is by nature enormous, and some constraint must be determined. Two useful limits are queue length and time signal length. The first switches to green if the line of cars becomes too long (say, over 20); the second when the light has been red for too long (30 seconds). These help manage the state space and quicken analysis.

Metric

Several metrics exist for evaluating traffic signal system performance. We choose to use minimize the average square waiting time of all cars, as given by the equation $\frac{\sum_{i=1}^{N} w_i^2}{N}$, where each w_i is a car's wait time in the system and N is the number of cars that have entered and exited.

Literature reviews suggest that this metric accelerates convergence [2]. It is of note that this metric does bias the model to prevent the case of a single car waiting a long period of time in favor of heavy traffic moving quickly in a different direction. Though this case appears favorable for the system, it is impractical in reality, as nobody wishes to wait multiple minutes for a light.

Baseline

At worst, we would like our work to represent an improvement to current non-intelligent techniques. The most common heuristic for timing traffic lights currently in use is Webster's formula (results seen in the below graph) [1] [3]. In a fixed time control setting, this calculates the green and cycle time from traffic data collected from the road network. Though it can be optimized to different times of day, any variation in traffic from the training patterns (due to accidents, special events, and similar unforeseen events) will result in increased travel time delay. Because this is the naivest and most common approach, we will utilize Webster's formula as the baseline for our project; that is, the algorithm we devise must perform better than a trained but non-adaptive approach.

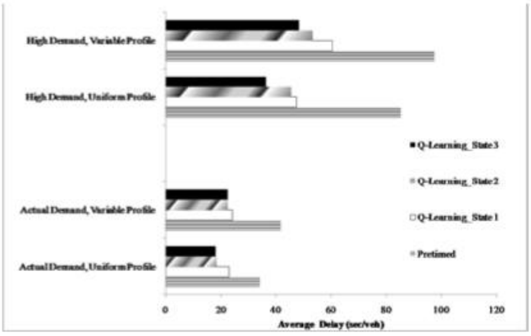

Fig. 4. Average Delay per Vehicle for Different Demand Levels and Profiles

Oracle

With respect to the traffic network, the performance of a light in the network will be worse than an isolated light with similar traffic. To see this, consider the cars at a single light. Optimizing its signals for the system as a whole will result in longer average wait times than letting traffic through at the best rate for that light. As a result, the optimal policy for a single light will be worse than if it were acting in isolation. Thus, finding the average waiting time for an isolated light serves as an oracle–the upper bound on performance. Unfortunately, we don't have the data for this average wait time: we have the code for this simulation but have not been able to interface successfully with the simulation software. Once we do so, we will be able to measure this result.

Input & Output

We model the traffic system using multi-agent reinforcement learning. As described above, we must model the system with multiple agents, otherwise a single agent may greedily optimize one light at the cost of the entire system. We represent the states of our model as ordered quintuples: $(\sum_{i=1}^{n} w_i^2$, $[t_1, t_2, ...t_k]$, T, d, $n)$. The first term represents the total (squared) waiting time of all cars that have gone past the light; $[t_1, t_2, ..., t_k]$ is a vector of wait times of cars currently at the light; T is the current time of the system; d is the direction that has a green light (N-S or E-W); n is the number of cars that have exited the light so far.

At each second, time time increments, and the possible actions are to change lights from North-South to East-West or vice-versa, or to retain the same lights. The cost of changing lights is the delay in traffic caused by the moments that both lights are red, and when either light is yellow. There is no cost associated with maintaining the same lights. When the time reaches the terminal time, which is determined as a parameter to the system, the cost is the $-\sum_{i=1}^{n} w_i^2$. The transition probabilities represent the probability that new cars enter the queue of waiting cars. The time, current light and total square waiting times change deterministically.

The reinforcement learning algorithm takes in three inputs and returns a policy. First, it takes in a terminal time, which determines the end state. Next, it takes in a distribution function which determines which lane the next car to enter the traffic network will drive on. Finally, it takes in a median rate of cars that enter the network. The model will use a normal distribution of rate of cars around this median. The output of the reinforcement learning algorithm will be a policy that determines what the light should be given a current state.

Conclusion

Traffic signal control is a field with significant economic impact that is ripe for multi-agent learning techniques. The multitude of possible states and magnitude of input make it intractable without an intelligent approach. We hope to contribute a potentially high-impact solution.

References

1. Webster, F.V. "Traffic signal settings." Road Research Technical Paper No. 39, Her Majesty's Stationery Office, London (1959).

2. Mannion, Patrick, Jim Duggan, and Enda Howley. "An experimental review of reinforcement learning algorithms for adaptive traffic signal control." Autonomic Road Transport Support Systems, Autonomic Systems. Birkhauser/Springer(2015).

3El-Tantawy, Samah, and Baher Abdulhai. "An agent-based learning towards decentralized and coordinated traffic signal control." Intelligent Transportation Systems (ITSC), 2010 13th International IEEE Conference on. IEEE, 2010.