STOCHASTIC SYSTEMS LAB

Adaptive Traffic Signal Control Using Multi-Agent RL

Sonu Dixit
Systems Engineering
Advisor – Prof. Shalabh Bhatnagar
Stochastic Systems Lab

Motivation

- Most Traffic junctions follow fixed time cycle, and it alternates its sign configuration in round robin manner. Vehicle flow rate changes a lot based on time of the day.
- Fixed time solutions don't adapt to current level of congestion, hence leading to long queues of vehicles, long delays in travel. If we manually tune cycle time based on duration of the day, still there are a lot of variations during a fixed time interval.

Objective

- To reduce congestion in a Traffic Network by effectively controlling and coordinating green signal timing at Traffic junctions in the network. Thus leading to decrease in average delay, increase in average speed.
- We need the solution to be scalable, and implementable with little extra infrastructure requirements.
- Building new roads to reduce congestion is a very costly and time consuming solution, Our solution is about utilizing(efficiently) the resources that we already have.

Problem Formulation as Markov Decision Process

- We model this problem as a multi-agent reinforcement Learning problem. In a network all junctions are represented as a RL agent. Each of these agents has access to local part of the environment(whole network).
- Environment = Whole traffic Network
- Every junction is an Agent, working in a common environment.
- Every agent can experience a part(local data) of environment.

Agent Description

- State:
  - Congestion at each incoming lane
  - Congestion at immediate Neighbors incoming lanes
  - Current Phase
- Action : Green signal duration {20,25,30,….65,70}
- Reward : Change in congestion at its lanes and neighbors lanes after performing action
- Objective : Maximize discounted sum of rewards
- Since reward(state) structure is dependent, local optimization, leads to global optimization.

Algorithm

- Advantage Actor critic, Proximal Policy Optimization (PPO)
- We can include entropy Loss to increase exploration at every state
- PPO is an On policy algorithm, On policy algorithms are sample inefficient
- Buffer of 100 latest samples, gives some off policy characteristic to PPO

Actor Critic Network Details

- Both are simple Dense Network, 2 Hidden layers (64 nodes each)
- PPO loss epsilon clipping = 0.2
- Entropy loss coefficient = 0.01
- Two actor networks(old, new), 1 critic network for every agent

Algorithm

```
Start Simulation
    For every Agent that needs to take action
        1. Put s,a,r,s'  into buffer, handle Synchronisation of all agents
        2. if buffer has enough samples from current policy
            3. K times:
            4.     Sample minibatch from buffer
            5.     Train actor(current_policy) on this data …. (a)
            6. Train Critic on whole buffer ….(b)
            7. Set old_policy = current_policy
End Simulation
```

$$\mathrm{r}_t(\theta) = \frac{\pi_\theta(a_t|s_t)}{\pi_{\theta_{old}}(a_t|s_t)} \qquad \mathrm{A}^\pi(s_t,a_t) = Q^\pi(s_t,a_t) - v^\pi(s_t)$$

$$\mathrm{L}^{CLIP}(\theta) = E_t\left[min(r_t(\theta)\hat{A}_t, clip(r_t(\theta), 1-\epsilon, 1+\epsilon)\hat{A}_t)\right]$$

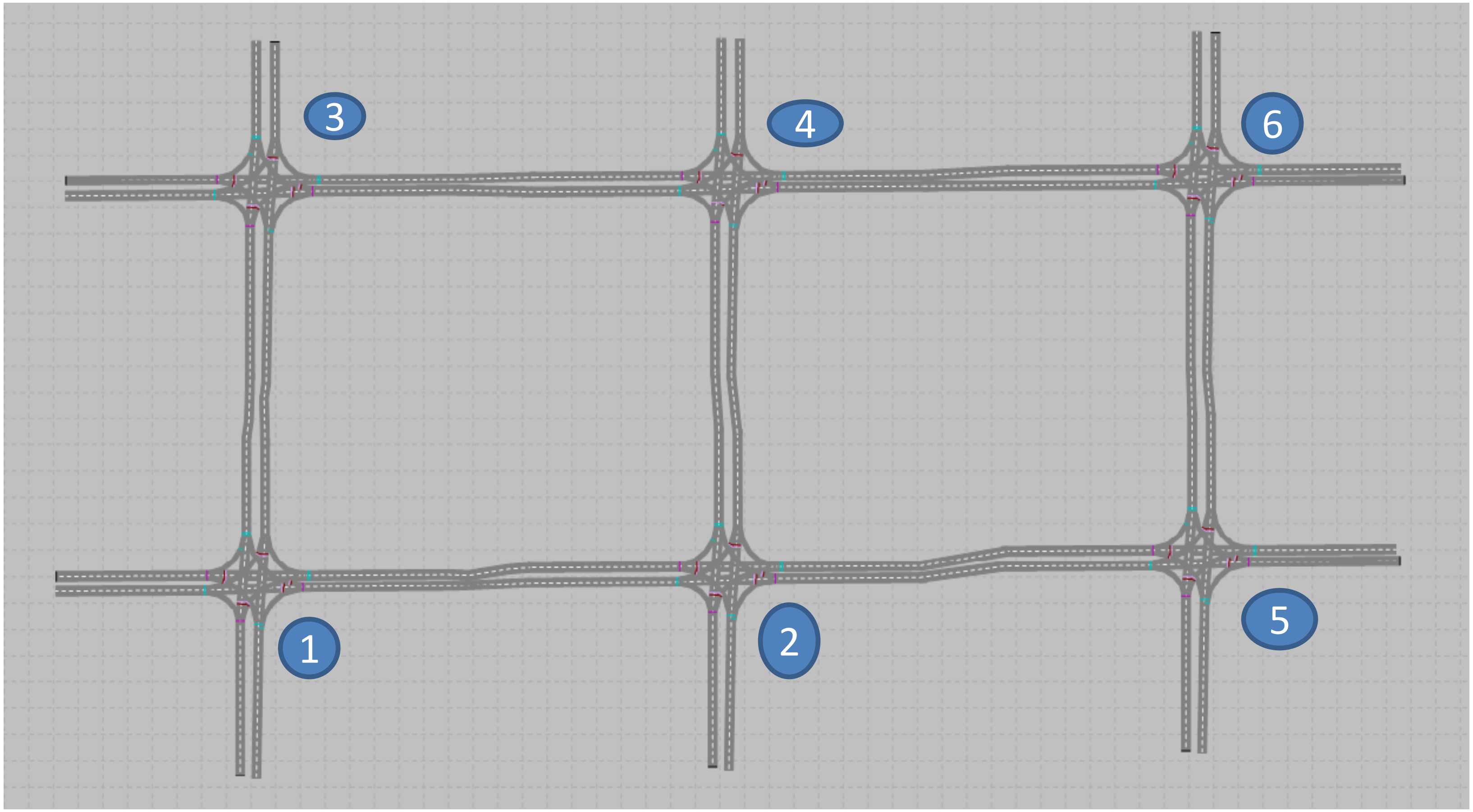
$$\mathrm{H}(\theta) = \sum_a -\pi_\theta(a_t|s_t) log(\pi_\theta(a_t|s_t))$$

$$\max\ [\mathrm{L}^{CLIP}(\theta) + \alpha H(\theta)] \qquad \text{(a)}$$

$$min(\sum_{k=t}^{k=T} \gamma^{k-t} r(s_k,a_k) + V(s_{T+1}) - V(s_t))^2 \qquad \text{(b)}$$

Traffic Network for experimentation

- PTV Vissim Simulation for modelling Traffic behaviour
- 6 junctions,
- 2 x 4 = 8 signals per junction, Queue counter behind every signal
- 10 Vehicle Input points
- Outward flow rate at each lane on incoming junction(Left, Straight, Right) = (0.25, 0.50, 0.25)



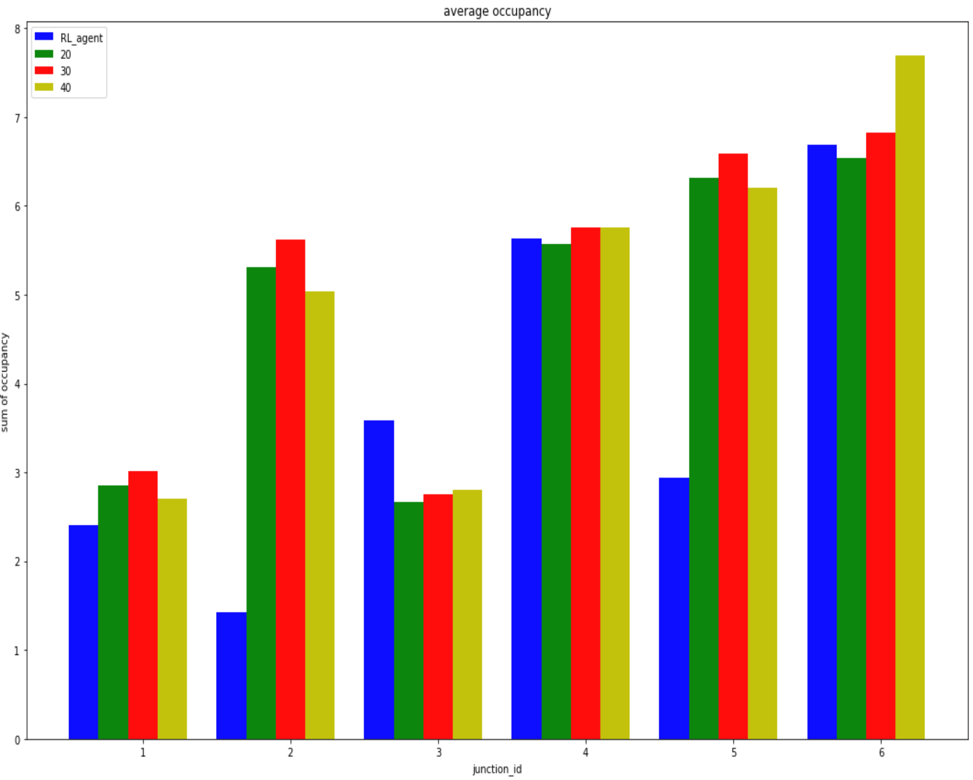
Experimentation and Results

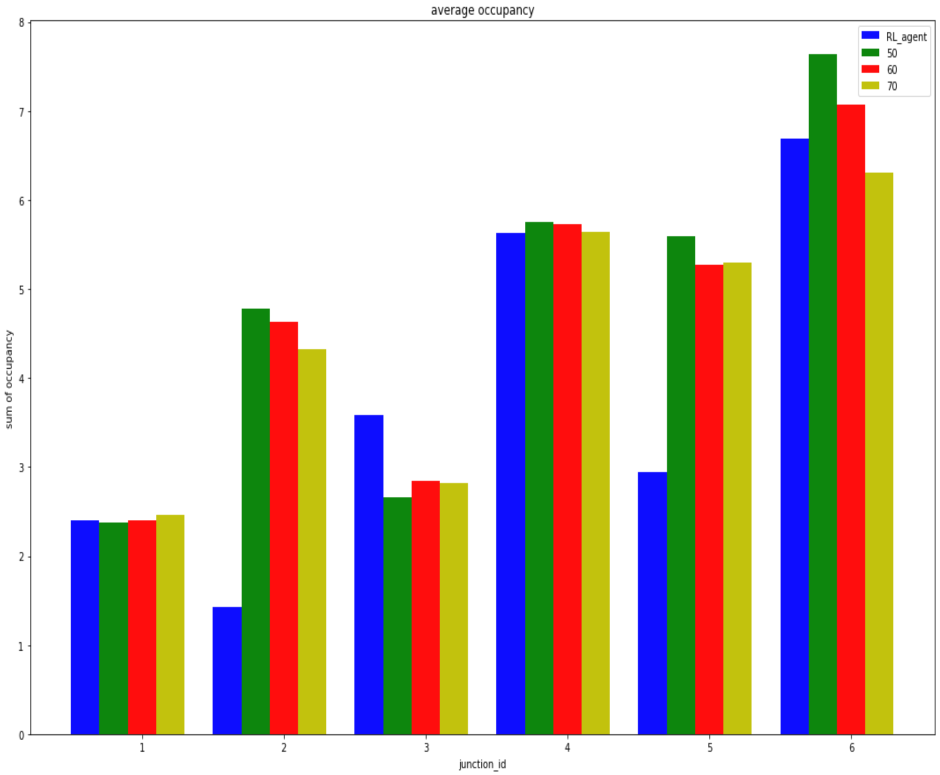
- Random vehicle input density:

  Changing after every 50 minutes

  Selected in range {500,1000,………,3000,3500} vehicles per hour

  Asymmetric vehicle incoming density at every junction
- Average Occupancy Comparison with fixed time algorithms





- If we combine the difference at all the junctions, RL agents are performing better.

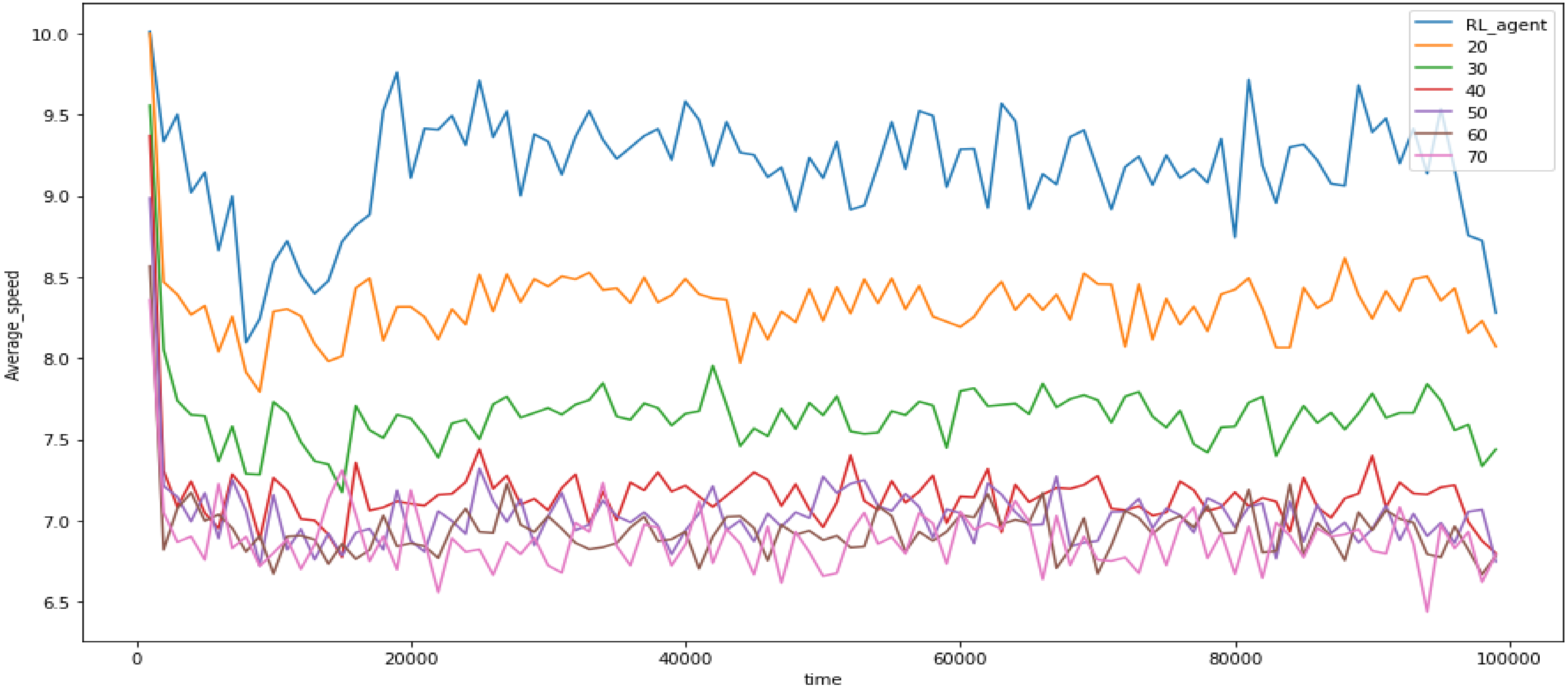
Experimentation and Results

- Random vehicle input density:

  Changing after every 50 minutes

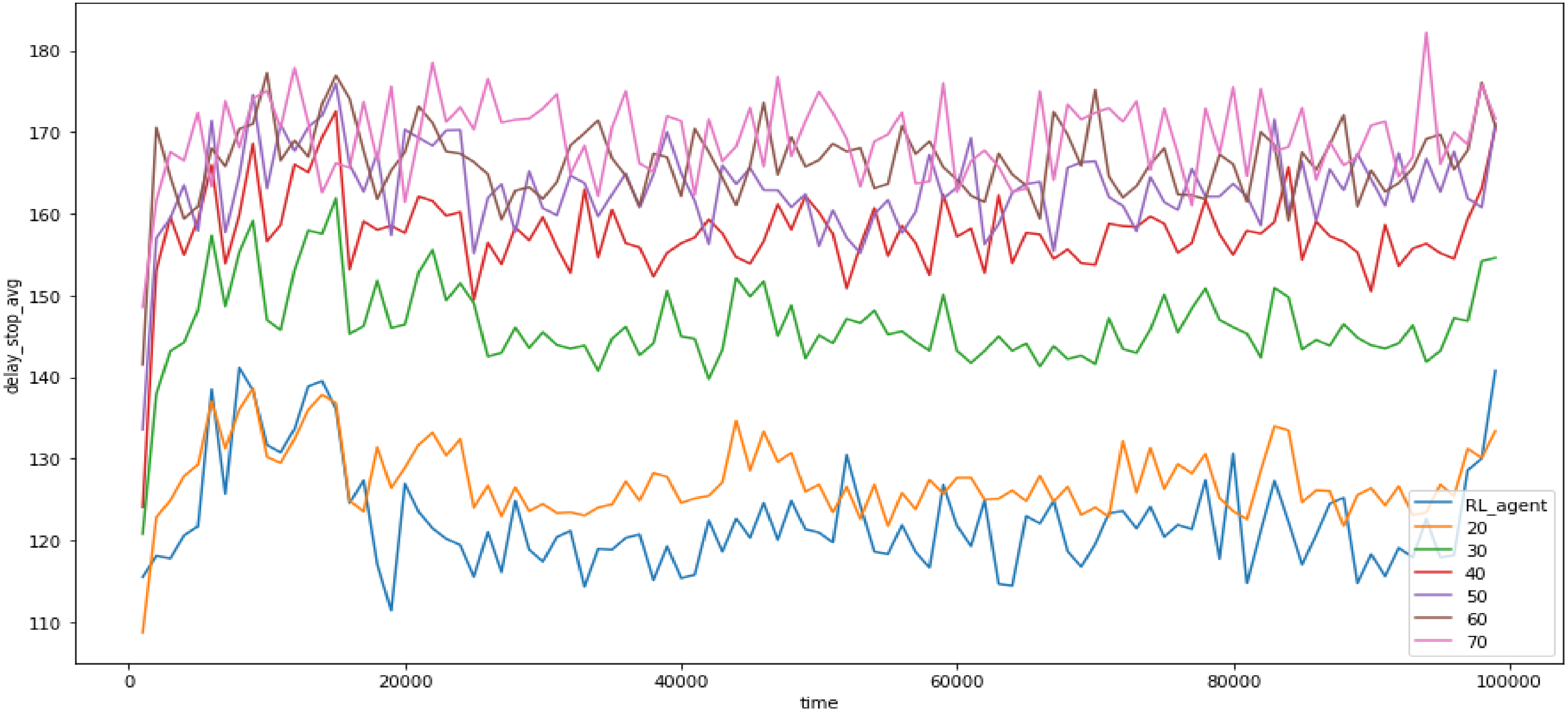
  Selected in range {500,1000,………,3000,3500} vehicles per hour

  Asymmetric vehicle incoming density at every junction
- Average Speed Comparison with fixed time algorithms



Experimentation and Results

- Random vehicle input density:
  - Changing after every 50 minutes
  - Selected in range {500,1000,………,3000,3500} vehicles per hour
  - Asymmetric vehicle incoming density at every junction
- Average Delay Stop time Comparison with fixed time algorithms



References

- Proximal Policy Optimization Algorithms (John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, Oleg Klimov)

- Multi-agent reinforcement learning for traffic signal control, Prabhuchandran K.J., Hemanth Kumar A.N, Shalabh Bhatnagar

- Vijay R. Konda and John N. Tsitsiklis. On actor-critic algorithms.IAM J. Control Optim., April 2003.