Reinforcement Learning For Adaptive Traffic Signal Control With Limited Information

Machine Learning (CS 229) Final Project, Fall 2015

Jeff Glick (jdglick@stanford.edu), M.S. Candidate, Department of Management Science and Engineering, Stanford University

Motivation

Recent research has effectively applied reinforcement learning to adaptive traffic signal control problems. Learning agents learn best with a high level of intelligence about what state the environment is in to determine the right actions for a given state. Most researchers have provided this access to perfect state information. However, in a real-life deployment, this level of intelligence would require extensive physical instrumentation. This study explores the possibility of training a control agent that has only access to limited information. However, in a real environment is in to determine the right actions for a given state.

Algorithm & Key Parameters

- Q-learning develops quality-values $Q(s,a)$ for each pair $(s, a)$ which is an estimate for the true value $V(s,a)$
- Continuous asynchronous updating for on-line learning; assume infinite visits to states for convergence
- Q-learning update:
  $$Q(s,a) = (1 - \alpha)Q(s,a) + \alpha(r + \gamma \max_a Q(s',a))$$
- States: Discretized state space; number of states for problem: ($\#$ light phases) x ($\#$ queue sizes) x ($\#$ waiting times) x ($\#$ edges)
- Learning Rate $\alpha$: Initially $\alpha = 1$ ignores previous experience; As $\alpha \to 0$, we are weighting previous experience more heavily
- Discount factor $\gamma$: Use $\gamma = 0$ to prevent myopic decision making
- Control policy: Given a state $s$, try action $a$ with probability:
  $$p_a(s) = \frac{\exp(Q(s,a) / \tau)}{\sum_{a'} \exp(Q(s',a') / \tau)}$$
- Reward $r$: determined by objective function:
  $$r = \sum_{i=1}^{4} \beta_i(\text{queue size}^i) + \beta_w(\text{waiting time}^i)$$

Simulation Build & Data Generation

Simulation Setup:
- Open Street Map and Java Open Street Map Editor
- Simulation for Urban Mobility (SUMO)
- Using realistic, variable arrival turn rates for a single 8- phase 4-way intersection

Next Steps

- Continue to experiment with learning strategy, parameters and objective function; improve discretization
- Work on state detection problem (limited information); learn arrival rates or use hour of day in the state space
- Change arrival rate dynamics to test robustness of process

Initial Results

- Queues blowing up
- Learning rate $\alpha$ shrinking quickly
- Crude discretization (most of 25k states not being visited)
- Challenges with volatility
- Reward should = change in objective function (reward improvement)

Acknowledgements:
- Michael Bennon (Stanford Global Projects Center), Allen Huang (CS229 Student), Jesiska Tandy (CS229 Student)