Unconventional reinforcement learning on traffic lights with SUMO

Master Degree in Computer Science

Francesco Refolli

Supervisor: Prof. Giuseppe Vizzari

Outline

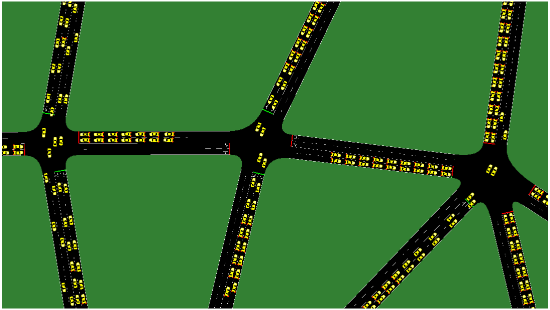
Introduction

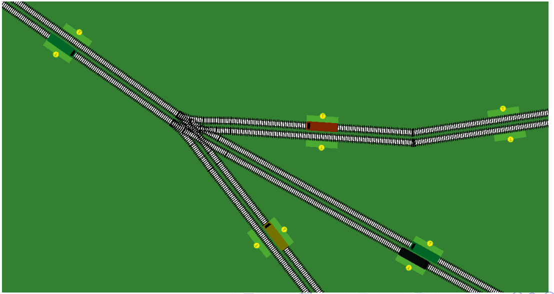
Problem statement

This thesis dealt with the Traffic Light Control problem (TCL) applying uncommon reinforcement learning techniques. In particular, the following research objective were pursued in the performed experiments:

- Evaluating the effectiveness of Curriculum Learning
- Evaluating the effectiveness of Multi Agent Learning
- Evaluating the effectiveness of Self-Adaptive agents
- Comparing observation/reward functions
- Comparing tabular and deep learning models
- Comparing Reinforcement Learning with existing solutions
- Evaluating the role of hyperparameters

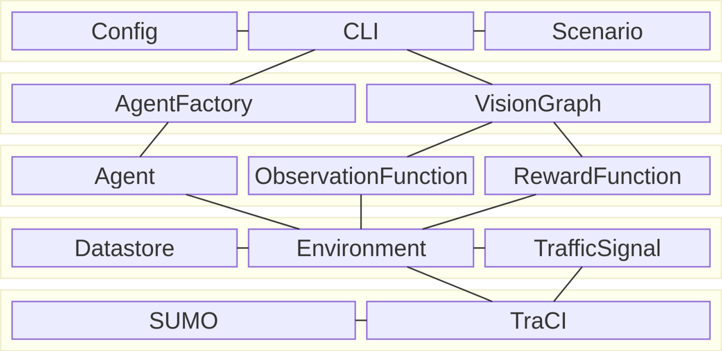
- Free and Open Source microscopic traffic simulator
- Developed at German Aerospace Center (DLR)
- Multimodal: cars, trams, bikes, pedestrians ...
- Highly customizable



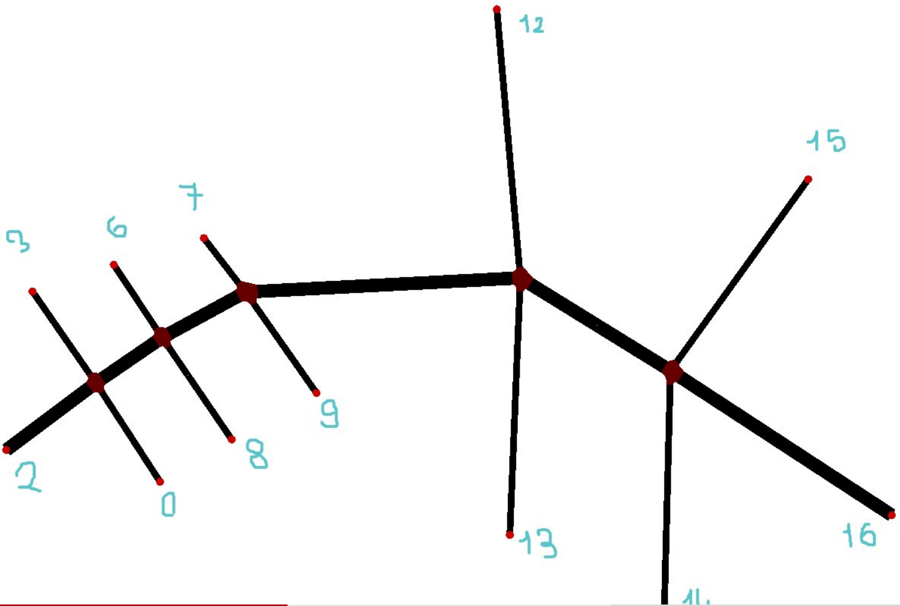


SUMO-RF: SUMO + Reinforcement Learning

A FOSS framework for Reinforcement Learning with SUMO developed as fork of *LucasAlegre/sumo-rl* with a focus on modularity, flexibility and Multi Agent Learning. It also contains several utilities for format conversions, metrics analysis and plot, schematic-based demand generation and more.

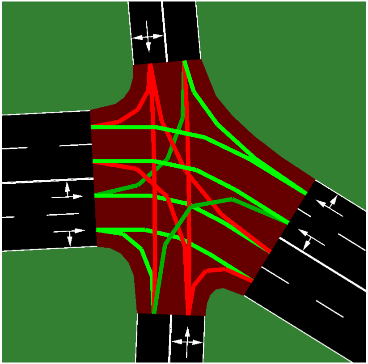


The scenario

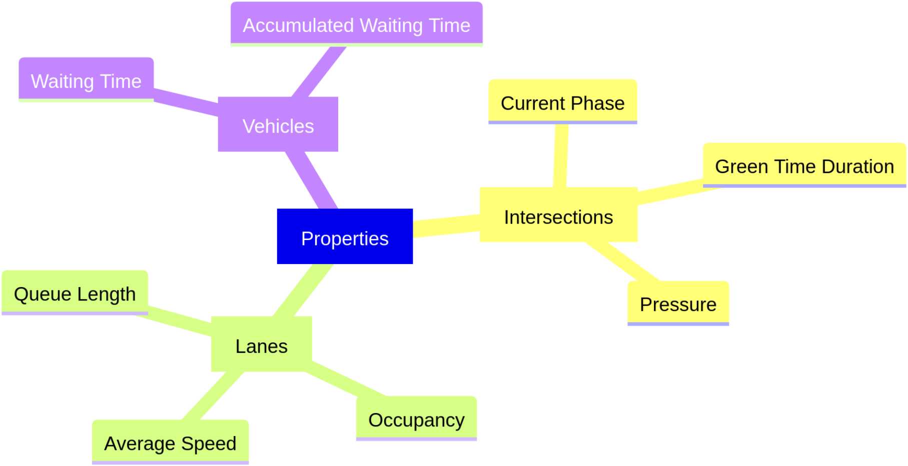


The Agent Model

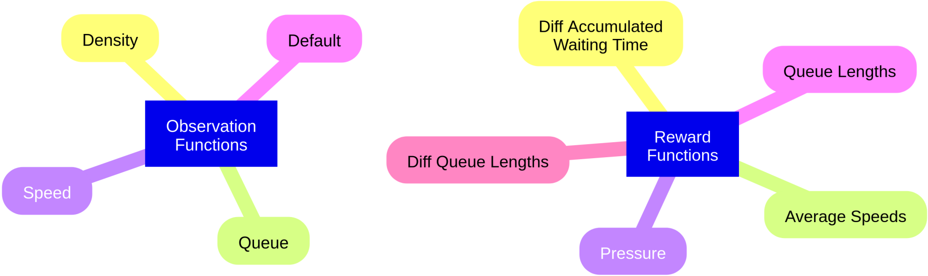
- Each agent can control one intersection and at each step (every 5 seconds) it can choose the next phase of the intersection.
- Every action is automatically enforced by TrafficSignal with also an intermediate yellow phase.
- It receives an observation of the current condition and a reward proportional to the goodness of its behaviour.
- If the agent is "smart", it uses the collected data to improve itself!



The Global State



Observing and Rewarding



Curriculum Learning

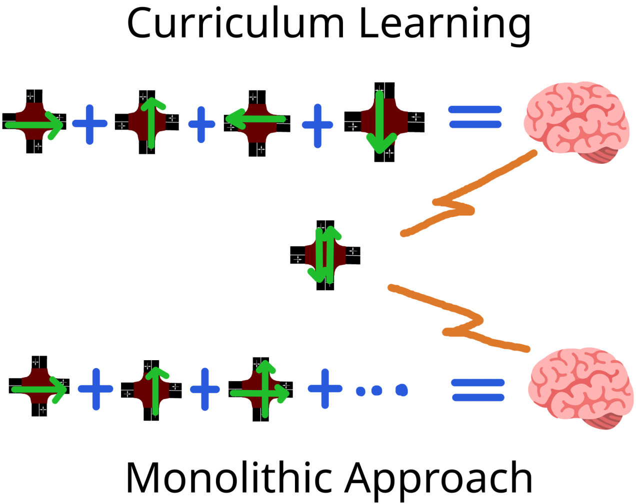
Experience Engineering

- Unlike classic RL problems like CartPole, here the configured demand over time of the system is actually a hyperparameter.
- In order to train and evaluate the agents a schematic of demand ("dataset") shall be created resembling both daily conditions and abnormal conditions.
- Since there are many junctions, many entry/exit points and each flow intensity can vary significantly, the complexity of a scenario is high.

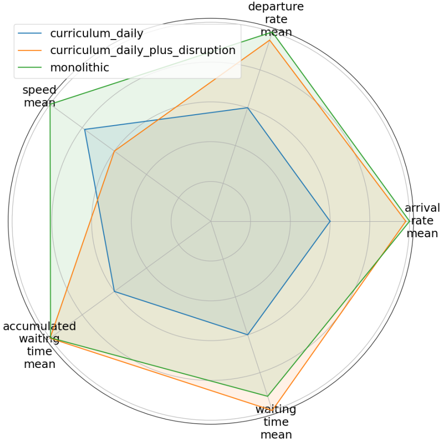
Two approaches:

- **Monolithic**: systematically train the agents against almost all combinations of subtaks.
- **Curriculum**: train the agents against each subtask in isolation expecting the agents to generalize their knowledge.

A modular learning framework



Experimental results



- TODO

Qualitative results

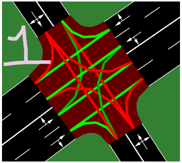
TODO

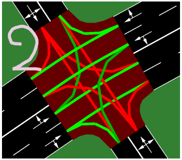
questa slide mostra un confronto qualitativo dell'efficacia dei tre dataset

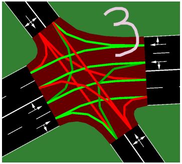
Multi Agent Learning

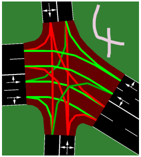
Multi Agent Learning

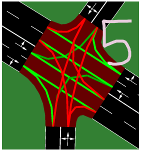
- No agent is isoled, they are all part of a whole and they influence each other with their own behaviour.
- What if an agent can *sense* its surrounding area by sharing observations with neighbours?
- What if an agent is *rewarded* for its influence on surrounding area by sharing rewards with neighbours?







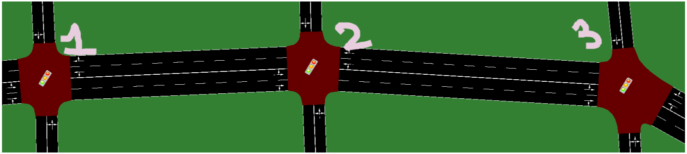




Observation sharing

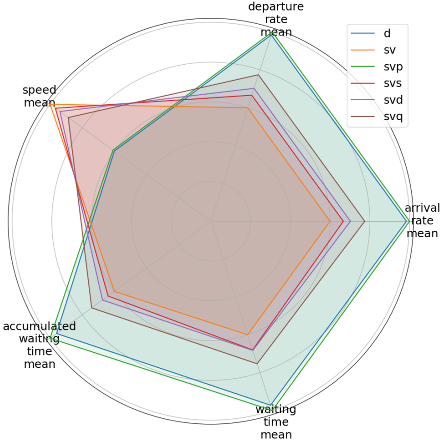
- The state S of an agent is computed through two dinstict observation functions f_{int}, f_{ext} meaning respectively the internal state and external state.
- The state of an agent x is a concatenation of $f_{int}(x)$ and $f_{ext}(y)$ for all $y \in N(x)$, where N is a function mapping an agent with its neighbours.
- It ws chosen to use the Density function as f_{int} because, among the observation functions where the agent can only see itself, it produced the best results.

Example:



$S(TLS_1) = f_{int}(TLS_1) \bullet f_{ext}(TLS_2) \qquad S(TLS_2) = f_{int}(TLS_2) \bullet f_{ext}(TLS_1) \bullet f_{ext}(TLS_3)$

Experimental results

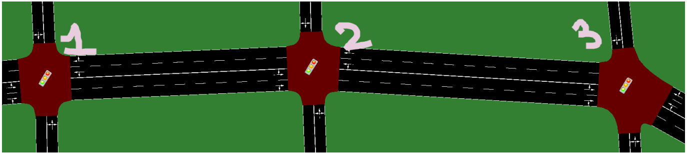


- TODO

Reward sharing

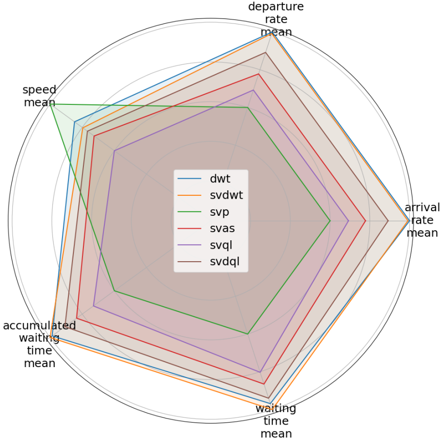
- The reward R of an agent is computed through one single reward function f representing the condition of an intersection.
- The state of an agent x is a sum of $f(x)$ and $f(y)$ for all $y \in N(x)$, where N is a function mapping an agent with its neighbours.

Example:



$R(TLS_1) = f(TLS_1) + f(TLS_2)$ $\qquad$ $R(TLS_2) = f(TLS_2) + f(TLS_1) + f(TLS_3)$

Experimental results



- TODO

Qualitative results

TODO
questa slide mostra un confronto qualitativo dell'efficacia di usare o meno lo sharing su obs/rew

Conclusions

A matter of perspective

TODO

questa slide contiene un confronto quantitativo tra priority, fixed, dql .. etc

l'idea e' mostrare che il dql e' aktually migliore degli altri e che i miglioramenti non sono seghe mentali

A matter of perspective

TODO
questa slide contiene un confronto qualitativo tra priority, fixed, dql .. etc
l'idea e' mostrare che il dql e' aktually migliore degli altri e che i miglioramenti non sono seghe mentali

Thank You