Traffic Flow Optimization with Reinforcement Learning

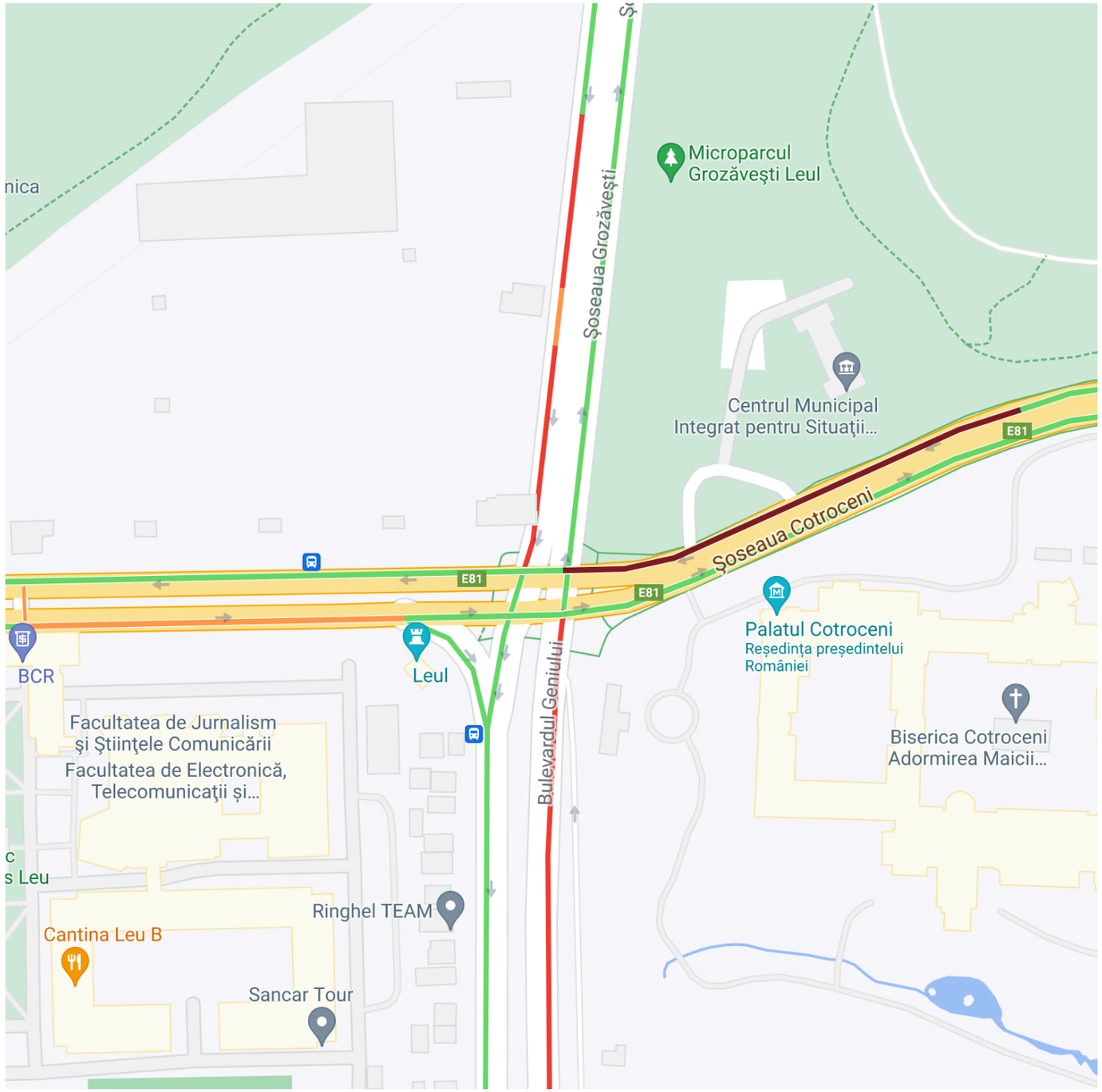
Using AI to solve the traffic problem

Dragos Giumanca

Lect. dr. Ciprian Paduraru

Motivation

Motivation


Microparcul
Grozăveşti Leul
Şoseaua Grozăveşti
Centrul Municipal
Integrat pentru Situaţii...
E81
Şoseaua Cotroceni
E81
Palatul Cotroceni
Reşedinţa preşedintelui
României
Biserica Cotroceni
Adormirea Maicii...
BCR
Leul
Bulevardul Geniului
Facultatea de Jurnalism
şi Ştiinţele Comunicării
Facultatea de Electronică,
Telecomunicaţii şi...
Ringhel TEAM
Cantina Leu B
Sancar Tour

Motivation

According to the Global Congestion Impact score, Bucharest has the worst traffic in the world in 2020.

One of the key factors of congestion is bad traffic lights systems.

Motivation

- Traffic flow is dynamic changing from hour to hour
- There are too many variable and cases to hardcode a good programme for red-green phases, ex. :ambulance in mission
- Multiple traffic lights need to respond to incoming cars to improve overall waiting time

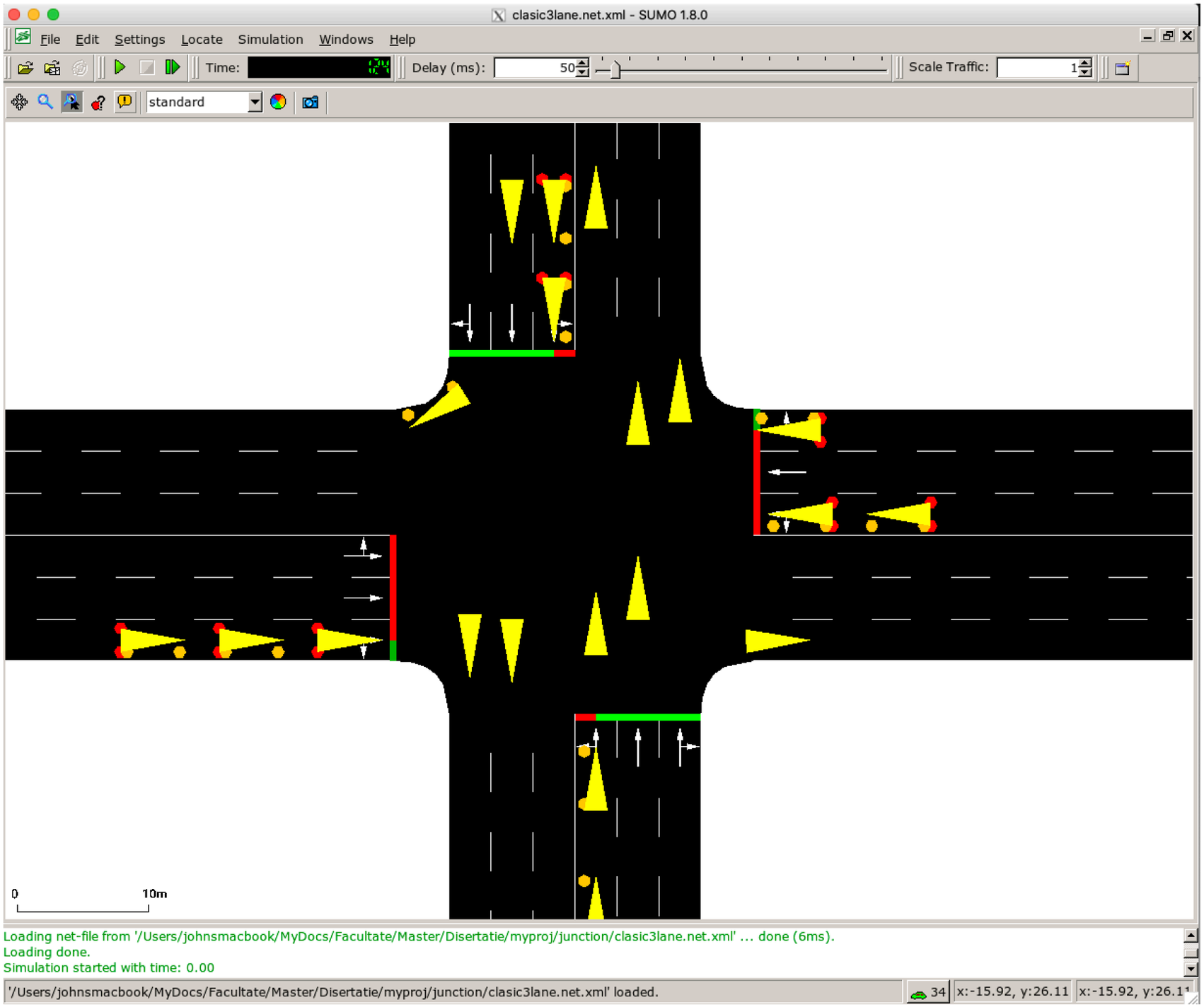
Related Work

- SUMO, OSRM, Carla
- State-of-the-art Reinforcement Learning approaches

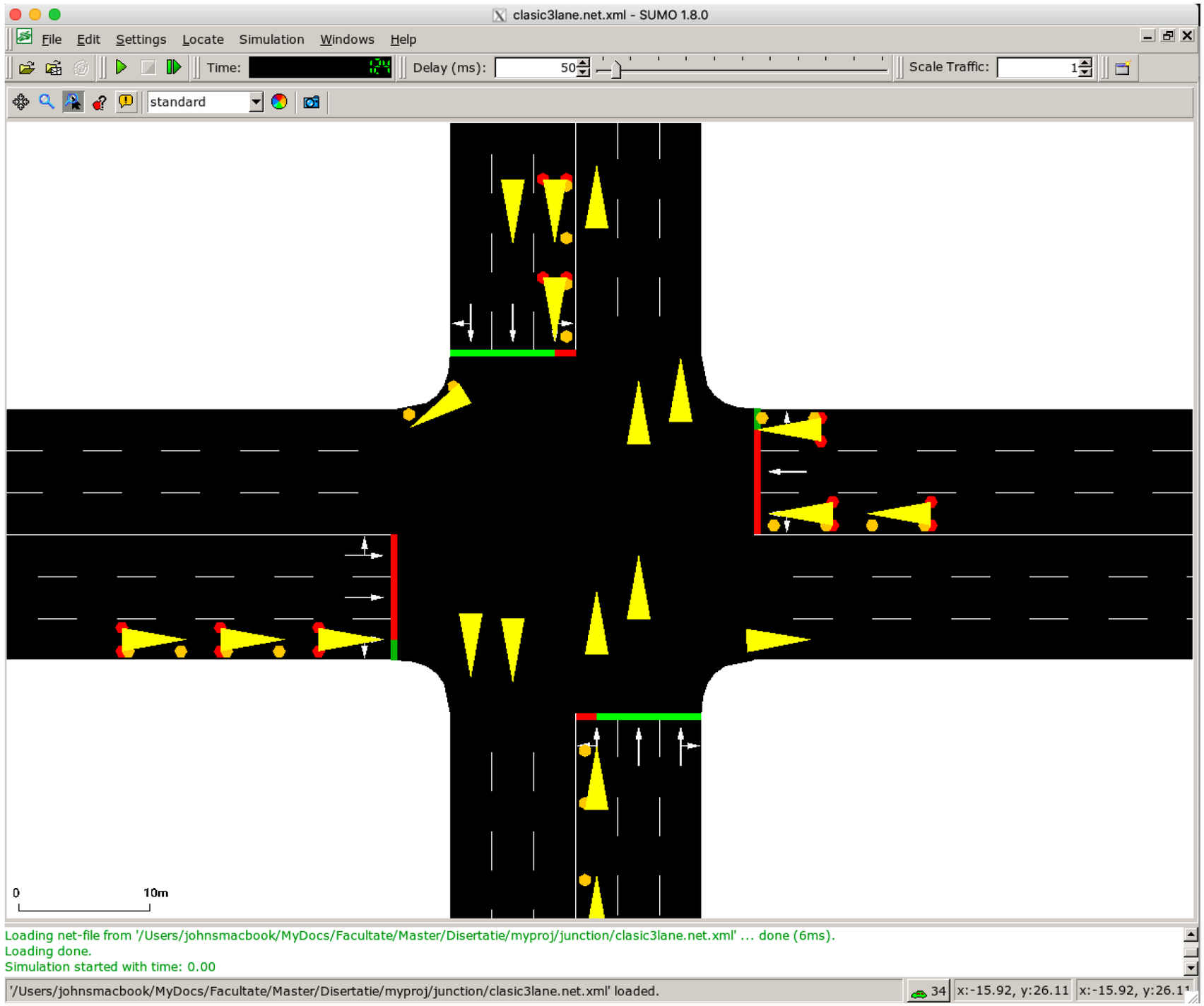
Our Work

- How does the RL methods compare and what is the simplest configuration from which we get good results?
- How can we configure our data input to mimic real world situations, ex.: using sensors to get incoming traffic data?
- How the results compare, what exactly is enough for the RL agent to learn something?

SUMO

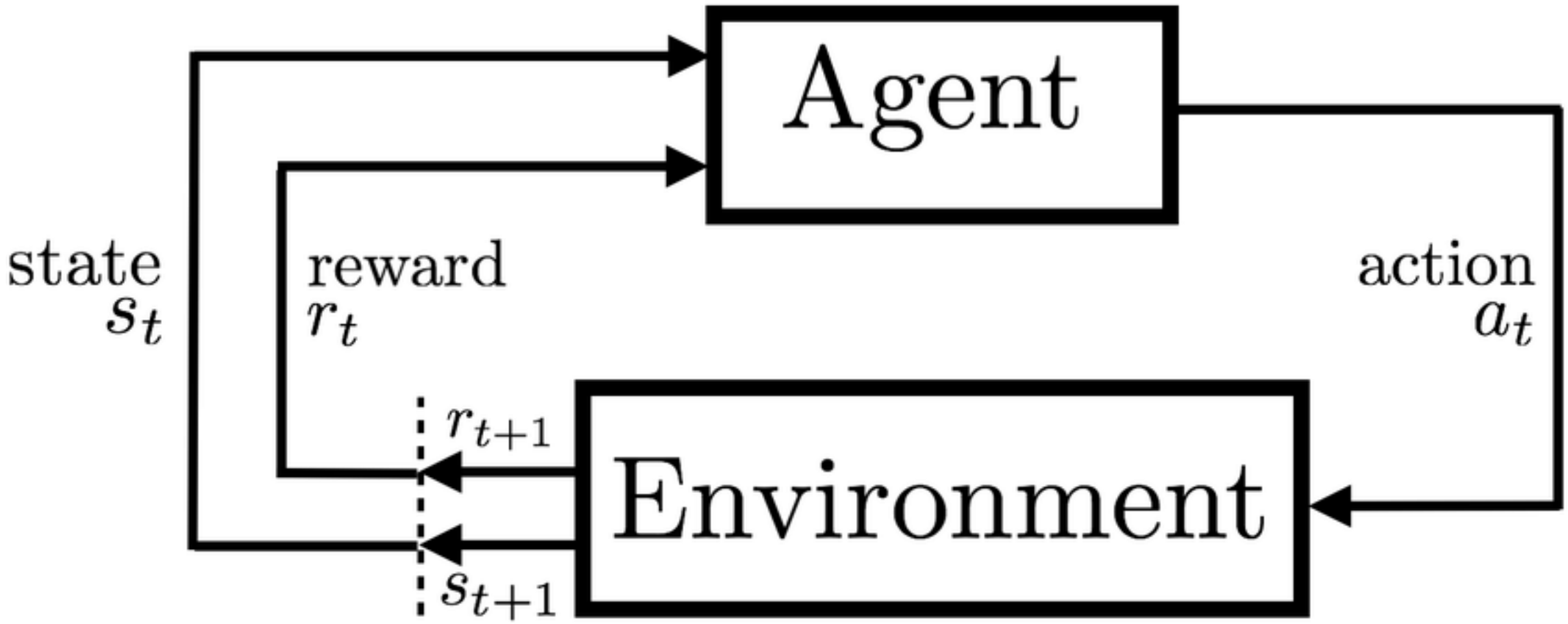

clasic3lane.net.xml - SUMO 1.8.0
File Edit Settings Locate Simulation Windows Help
Time:
Delay (ms):
50
Scale Traffic:
1
standard
0
10m
Loading net-file from '/Users/johnsmacbook/MyDocs/Facultate/Master/Disertatie/myproj/junction/clasic3lane.net.xml' ... done (6ms).
Loading done.
Simulation started with time: 0.00
'/Users/johnsmacbook/MyDocs/Facultate/Master/Disertatie/myproj/junction/clasic3lane.net.xml' loaded.
34
x:-15.92, y:26.11

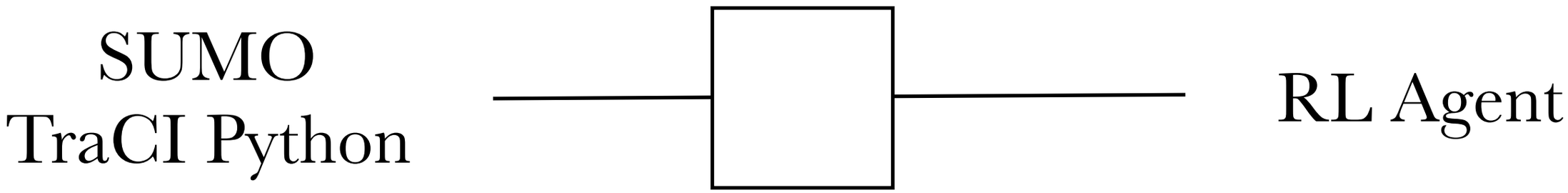
SUMO



- TraCI Python
- Nets
- Routes for cars
- Special vehicles
- Many stats
- Good Docs

Deep Reinforcement Learning




SUMO
TraCI Python
RL Agent

Our Work

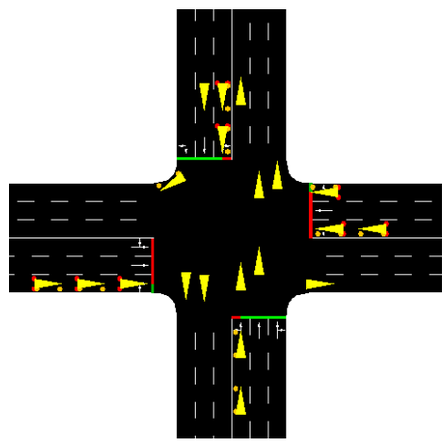
SUMO
TraCI Python

Custom Environment

RL Agent

- Reward
- Observation
- Action

Experiments



Setup:

- Classic 2 roads junction with 3 lanes
- Traffic generated by custom distribution of probabilities
- Default TL Programme: 42s Green 3s Yellow 10s Left-Green Cycle

Action:

- [0,3] - 4 actions for each Green and Left-Green Phase

Observation:

- Array of size 13, first digit [0,3] for TL phase and 12 digits for each lane stopped nr of cars (1+12)
- One-hot encoding and normalized values (4+12)
- Previous normalized values + values for occupancy for each lane (4+12+12)

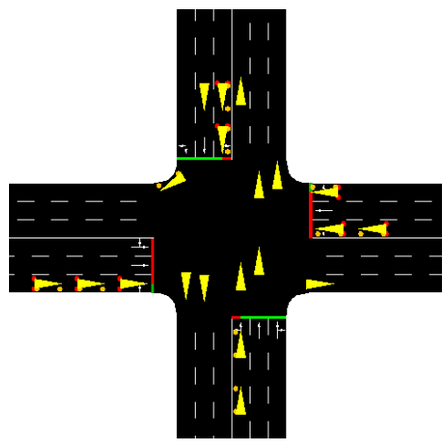
Reward:

- -1 for each stopped car for each lane and -1000 if the queue is bigger than the th.
- Negative values added up for normalized observation
- Negative values for accumulated waiting time added up

Results

Not much…

Experiments



Setup:

- Classic 2 roads junction with 3 lanes
- Traffic generated by custom distribution of probabilities
- Default TL Programme: 42s Green 3s Yellow 10s Left-Green Cycle

Action:

- [0,3] - 4 actions for each Green and Left-Green Phase

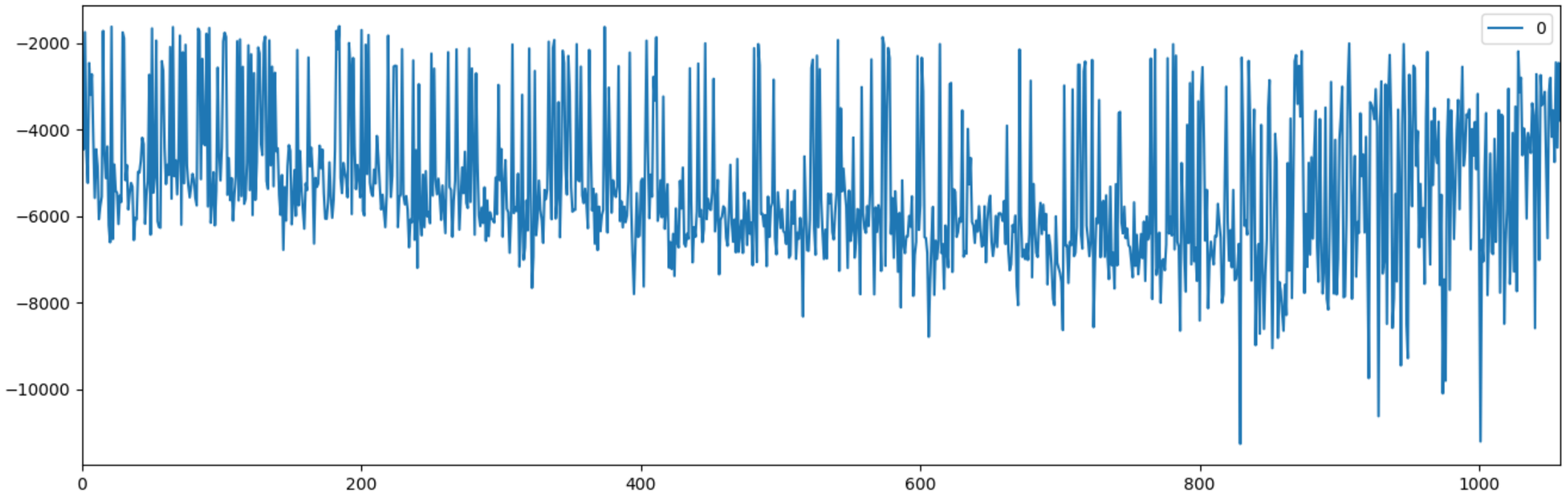
Observation:

- Array of size 13, first digit [0,3] for TL phase and 12 digits for each lane stopped nr of cars (1+12)
- One-hot encoding and normalized values (4+12)
- Previous normalized values + values for occupancy for each lane (4+12+12)

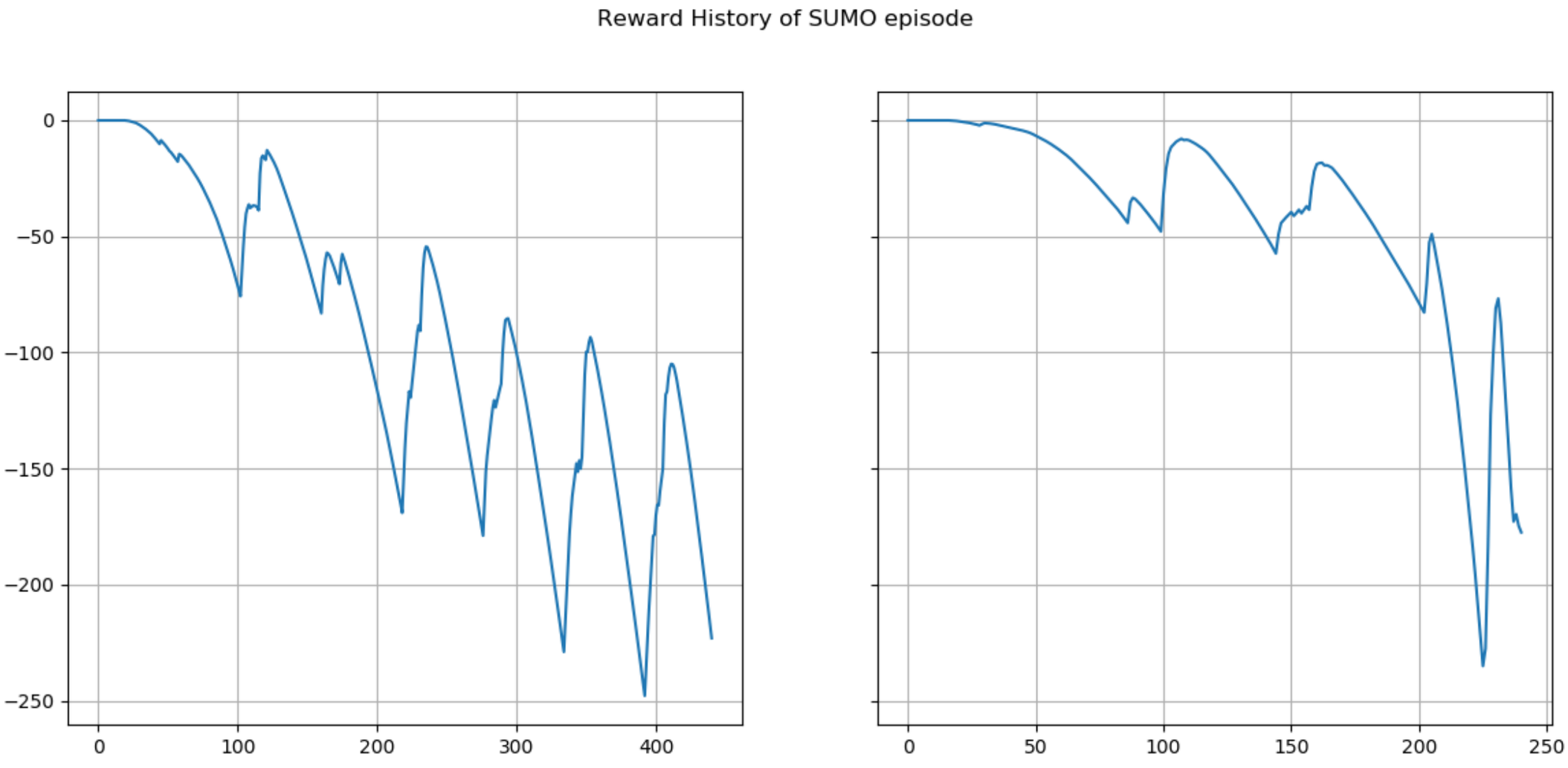
Reward:

- -1 for each stopped car for each lane and -1000 if the queue is bigger than the th.
- Negative values added up for normalized observation
- Negative values for accumulated waiting time added up

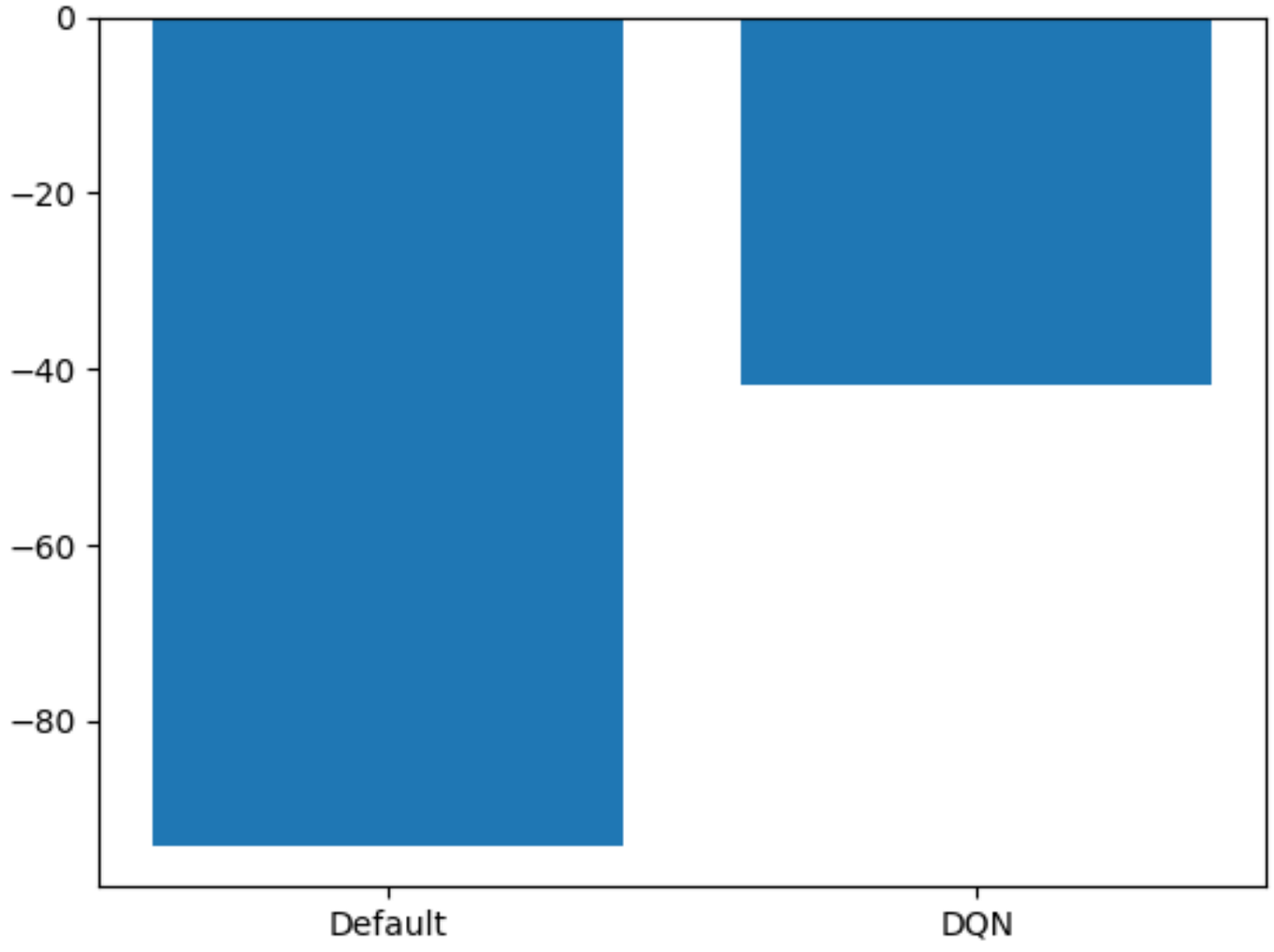
Results


Training over 1000 episodes
0
−2000
−4000
−6000
−8000
−10000
0
200
400
600
800
1000

Results


Reward History of SUMO episode
0
−50
−100
−150
−200
−250
0
100
200
300
400
0
50
100
150
200
250

Results



What is Next?

- More experiments with light and heavy traffic
- More Observation methods with sensors to mimic real life situations

Thank you!