Kalle Prorok/2021-05-06

# Reinforcement Learning (RL)

Lab 4 and Lab 5 in System och algoritmer för autonoma fordon.

## Intro

In this lab(s) you will play a bit with (D)RL-algorithms in a Traffic environment.

## Environment

The lab is based on work by Eleurent on

<https://github.com/eleurent/highway-env>

There are Colab-buttons in the source code to do test runs immediately. Remember to enable GPU before starting.

## Open AI

Provides free resources (except MuJoCo which requires license for Hopper-v1, often pybullet can be used instead). Elon Musk was involved initially. <https://openai.com/>

One of them are simulated environments to be used by RL-algorithms; the Gym:

<https://gym.openai.com/>

Unfortunately it doesn’t work well under Windows so we recommend Mac OS X, Linux or Colab. (Windows is possible via Xming and/or WSL(1) – on your own risk).

Another resource are courses like:

Spinning Up https://spinningup.openai.com/en/latest/index.html

and Stable baselines with Agents: <https://github.com/DLR-RM/stable-baselines3>

(the version 3 moved from Tensorflow to PyTorch in January 2020)

With the RL Baselines3 Zoo with examples/test runs with trained agents: <https://github.com/DLR-RM/rl-baselines3-zoo>

## Task

In the parking\_model\_based.ipynb the vehicle finds and navigates to a desired parking lot. The problem is the vehicle takes a shortcut crossing neighbouring lots possibly colliding with already parked cars. Please improve the reward function to make the vehicle approach the desired lots more straigthforward (or straightbackward) to avoid these possible collisions.

i.e change the code line

rewards = -torch.pow(torch.norm((states-goal)\*reward\_weigths, p=1, dim=-1), 0.5)

in reward\_model() into something more appropriate.

## Reporting

Report with a link to your (colab) ipynb-file, stored in you github or google drive in which you save some kind of performance measurments of testruns with both the original and your improved version.