I learned about DEEP RL:

https://homes.cs.washington.edu/~todorov/courses/amath579/reading/Continuous.pdf

I found:

https://github.com/matthiasplappert/keras-rl

OpenAI Gym - https://gym.openai.com/docs

Good articel kerasrl: https://oshearesearch.com/index.php/2016/06/14/kerlym-a-deep-reinforcement-learning-toolbox-in-keras/

Traffic light sim: https://www.youtube.com/watch?v=s6tfcSIBjsU

http://sumo.dlr.de/wiki/Simulation/Traffic_Lights

action/tl= {r|g|G|y|o|O|u}

tls=#connections

Character	Description
r	'red light' for a signal - vehicles must stop
y	'amber (yellow) light' for a signal - vehicles will start to decelerate if far away from the junction, otherwise they pass
g	'green light' for a signal, no priority - vehicles may pass the junction if no vehicle uses a higher priorised foe stream, otherwise they decelerate for letting it pass
G	'green light' for a signal, priority - vehicles may pass the junction
u	'red+yellow light' for a signal, may be used to indicate upcoming green phase but vehicles may not drive yet (shown as orange in the gui)
o	'off - blinking' signal is switched off, blinking light indicates vehicles have to yield
O	'off - no signal' signal is switched off, vehicles have the right of way

Reward design:

https://deepblue.lib.umich.edu/bitstream/handle/2027.42/89705/jdsorg_1.pdf?sequence=1

pybrain explained: http://simontechblog.blogspot.de/2010/08/pybrain-reinforcement-learning-tutorial_21.html

DQN: https://jaromiru.com/2016/10/03/lets-make-a-dqn-implementation/

All connections with the same tl id are on the same junction

Convert to readable data with: netconvert -s rilsa1.net.xml --plain-output-prefix

User doc http://www.sumo.dlr.de/wiki/Networks/Building_Networks_from_own_XML-descriptions#Node_Descriptions

- `traffic_light`: The junction is controlled by a traffic light (priority rules are used to avoid collisions if conflicting links have green light at the same time).

tlLogic.id==junction.id

---

theano:

added nano .theanorc in home dir – brauch ich doch nich

wrote shell scripts to run sumo

---

with script.sh 2>&1 | tee log.file

modded the lust and cgn scenario files

lust to start from 20k and cgn to print full statistics

---

get emergency stops from traci:https://sourceforge.net/p/sumo/mailman/message/34393147/

Hello,

you would have to track the velocity for each vehicle and compare the previous time step with the current. If the difference is larger than a threshold (4.5m/s is the default maximum deceleration) it constitutes an emergency stop.
regards,
Jakob

---

found too late: https://arxiv.org/pdf/1611.01142.pdf

https://esc.fnwi.uva.nl/thesis/centraal/files/f632158773.pdf

http://cs229.stanford.edu/proj2016spr/report/047.pdf

https://github.com/bstriner/gym-traffic

interesting intersections to watch: 271358878, 277433131

61794247