CS234 winter 2018 course project proposal:

Mastering TORCS with Risk-Averse AI Agent

Participants: Kostya Sebov <ksebov> and Adhitya Venkatesh <avenkate>

Introduction

Driving a car, even on a closed track, is a complex task requires experience and dexterity. Racing a high performance car involves advanced maneuvers like advance braking, precise cornering and operating the car under extreme conditions when the wheels may be slipping or drifting. This demands detailed understanding of the track trajectory, road conditions and involves precise planning and execution. The task is even more complex if the conditions are not uniform or changing, in which case a prompt correction is critical to achieve stable performance.

Goals

Our main goal is to build a DRL network capable of discovering advanced racing techniques in simulated environment in a risk-sensitive manner. We estimate that in order to plan and execute complex maneuvers the state space needs to include temporal dimension, that is several snapshot are used to plan the next action.

Implementation

For this we are going to use The Open Racing Car Simulator (TORCS) as a simulation platform.

Input consist of: total of 27 floating point values; 6 d.o.f of position and speed, 2 telemetry values, and 19 range finder (radar) values. Note: no visual information is used as input, although it can be used as a follow-up research. Output: 2 floating point values: steering and acceleration/breaking.

The network will be implemented in Tensorflow or Keras and Python.

Evaluation

For baseline and comparison we will use hand-programmed agents that participated in simulated TORCS competition. We plan to train the network on a tracks used in such competition, with one track selected exclusively for evaluation/validation.

Links and References

Using Keras and Deep Deterministic Policy Gradient to play TORCS
High Dimensional Planning and Learning for Off-Road Driving
Simulated Car Racing with Python and TORCS