TORCS rally simulator and RL

Presented by: Ardi Tampuu

Institute of Computer Science, University of Tartu

April 2016
The Open Racing Car Simulator
http://torcs.sourceforge.net/

- works on Windows and Linux
- easy to install (at least in Linux)

Simulated Car Racing Championship Server
https://sourceforge.net/projects/cig/files/SCR%20Championship/

- competition every year to develop the best racing algorithm
- easy access to relevant sensor data
What does it look like?
Tambet has written step-by-step installation instructions:

https://courses.cs.ut.ee/2016/scml-seminar/spring/Main/VirtualRacing

6 steps in Linux, 2 steps in Windows.
Championship Server Code

Figure 1: The architecture of the competition software.
Championship Server Code

- Server and Bots are separated and communicate through UDP
- This means Bots can be written in any language
- real time - server does not wait for the Bot to calculate
- can be run in text mode
Sensors and actuators

Important stuff:

- Sensors sense the state of the car
- Actuators play out our commands
Most useful sensors:

- distance from center of track
- angle to the direction of the road
- distance from the edge of the road with 10 degree steps
- speed
- distance raced (to calculate reward in RL)
Let’s see the list of available sensors and actuators
This seems so damn complicated...

https://github.com/lanquarden/pyScrcClient

A driver in 100 lines of code (hardcoded, not RL)
Does anyone still think this is beyond your coding skills?
BotMobile

An unnecessarily complicated version of RL for TORCS was made by us in a hackathon.

https://github.com/tambetm/botmobile

It has parts of the code needed for controlling a driving wheel. Also code for overriding AI commands.
Tricks and tips:

• turn off damage to your car by -nodamage when launching torcs
• -nolaptime should remove max time allowed per lap
• guess what -nofuel does?
DISCUSSION
Thank you for your attention!