SESE Projects 2020 (EES + MPSEES)

Safe Reinforcement Learning

Project Topic Overview

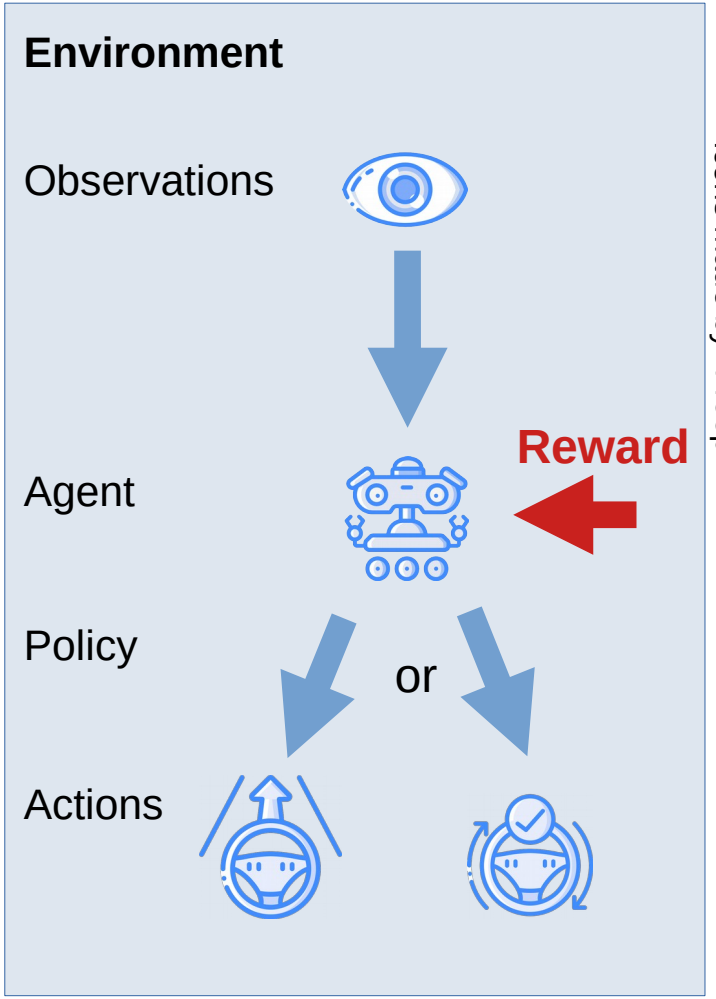
Sabine Glesner, Verena Klös, Paul Kogel,
Willie Szollmann

Technische Universität Berlin

SESE

Reinforcement Learning

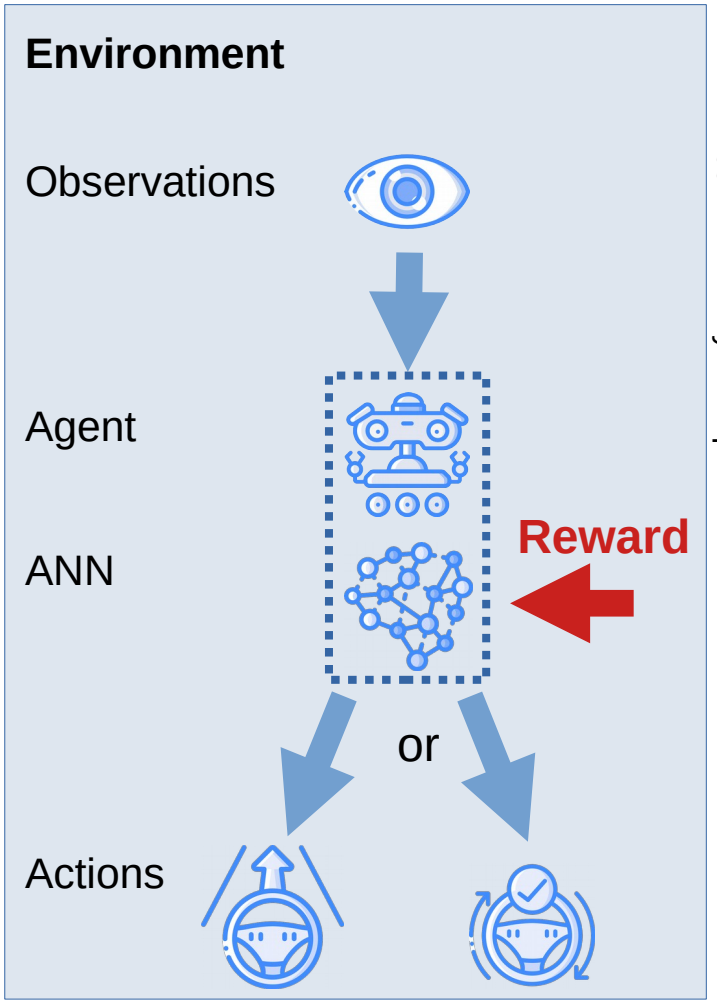
- Reinforcement learning is a subarea of **machine learning**
- Solving Problems with **statistical methods** and **collected data**
- How it works (simply put):
  - An **agent** makes **observations** in an **environment** and then chooses an **action** according to his **policy**
  - Whether the action has brought the agent closer to his **goal**, it will receive a **reward**
  - The agent changes his policy to **maximize** the reward



Icons made by "Freepik" from "www.flaticon.com"

Deep Reinforcement Learning

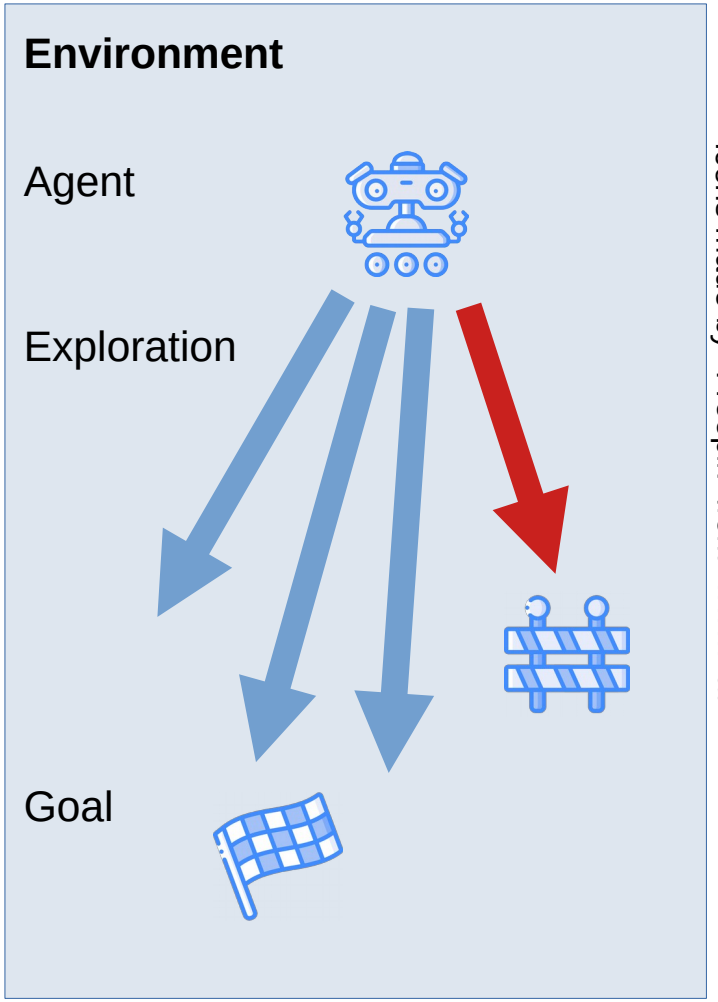
- In large state spaces classical RL quickly reaches its limits
- Combining RL with **deep learning**
  - The policy is approximated with an **artifical neural network**
  - After enough training, actions can be inferred even for **unknown observations**
- Many recent successes in the field of AI are based on deep RL:
  - AlphaZero (playing Go)
  - AlphaStar (playing Starcraft II)



Icons made by "Freepik" from "www.flaticon.com"

Safety Issues

- The disadvantage with deep RL is that it is no longer possible to trivially trace which observations lead to which action
- This is particularly problematic if the agent is to operate not only in a simulation but in real environments and the goals are **safety critical**
- Safety critical situations can already occur during data collection (**exploration**)



Icons made by "Freepik" from "www.flaticon.com"

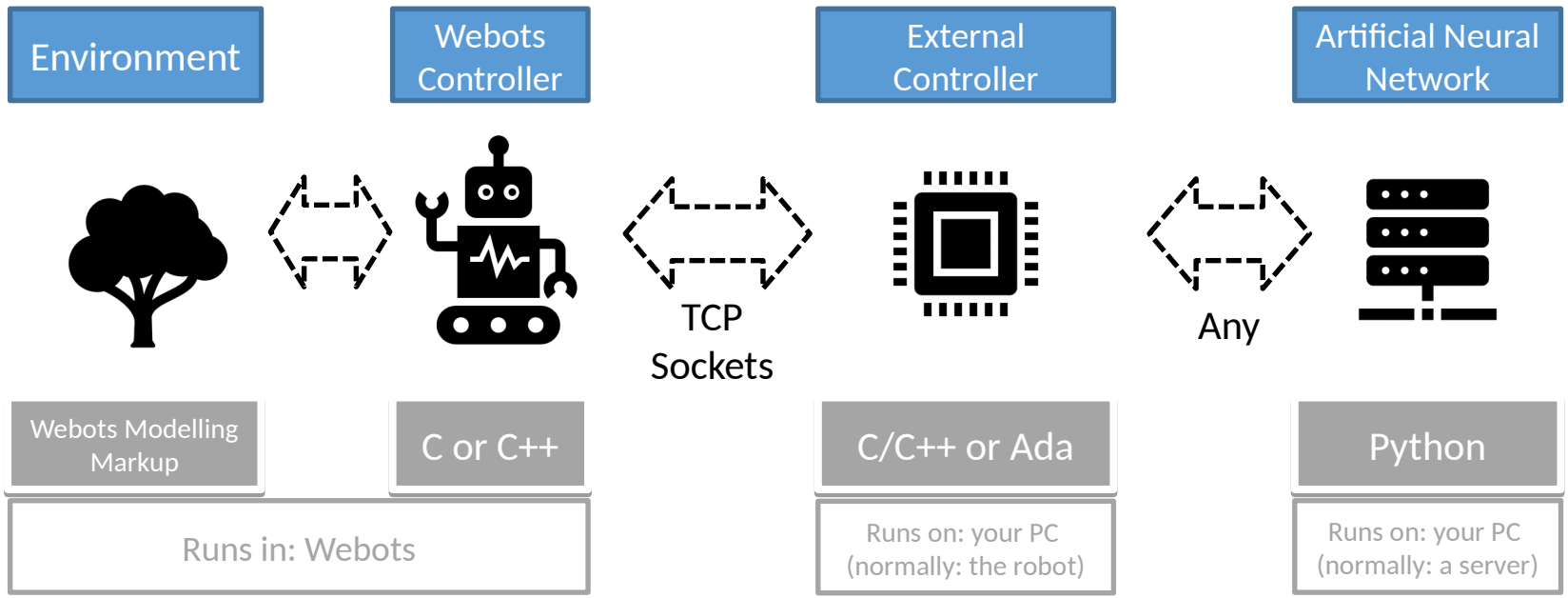
Objective

- Make a robot **autonomously reach a given target area** in its environment
- **Without endangering** itself or other participants during exploration
- Optional: Add a second robot and let both **cooperate** to reach their goal faster

Credit: NASA/Dominic Hart

Requirements: Overview

Your project **must** use the following architecture:



Requirements: Environment

- The **target zone** is a small area inside this environment
- The **position** of the target zone is given by a **Webots GPS node**

- **These problems need to be addressed:**
  - Through which terrain should the robot move
  - (consider your available **actors**)?
  - What should the obstacles look like
  - (consider your available **sensors**)?
  - Should the **test environment** differ from the **training environment**?

Requirements: Robots

- It is **advisable** to restrict yourself to the **Tinkerbots subset** of the Webots simulator
- Use a **Webots GPS node** to determine the **position** of the robot

- **These problems need to be addressed:**
  - Which **sensors** and **actors** should the robot use
  - (how does your environment look like)?
  - Should the Webots controller rely entirely on the external controller or integrate **own safety measures**?
  - What **information** should the Webots controller **exchange** with the external controller?

Requirements: Neural Network

- You are free to use different machine and reinforcement learning frameworks (**Tensorflow**, **pyTorch**, **RLlib**)
- Working with these frameworks can be done in **Python**

- **These problems need to be addressed:**
  - Should one robot contain **multiple agents/networks**?
  - How to evaluate the **efficiency** and **safety** of your trained network?

Further Readings: Paper

- **Concrete Problems in AI Safety**, Dario Amodei, Chris Olah, Jacob Steinhardt, Paul F. Christiano, John Schulman and Dan Man, https://arxiv.org/pdf/1606.06565.pdf

- **Benchmarking Safe Exploration in Deep RL**, Joshua Achiam, Alex Ray and Dario Amodei, https://d4mucfpksywv.cloudfront.net/safexp-short.pdf

- **Algorithms for Reinforcement Learning**, Csaba Szepesvári, https://sites.ualberta.ca/~szepesva/papers/RLAlgsInMDPs.pdf

Further Readings: RL Tutorials

- **With Tensorflow:** https://spinningup.openai.com/en/latest
- **With Rllib:** https://ray.readthedocs.io/en/latest/rllib.html
- **With pyTorch:** https://pytorch.org/tutorials/intermediate/reinforcement_q_learning.html