Abstract

abstract in German in at least 500 words is a requirement!!!

optionally also in English

write this last with introduction

Contents

1 Introduction

write this last with abstract

2 Background on Formal Languages

write after two other chapters (15-20 pages)

preliminaries to cover:

- regular languages (LTL, regular expressions)
- (deterministic) finite state automata (DFA)
- mealy/moore machines
- büchli automata

3 Background on Reinforcement Learning

(10-15 pages)

cover:

- MDP
- policy
- q function
- Bellman equations
- q learning

The goal of reinforcement learning is to learn a task by maximizing the total rewards along sequences of decisions called episodes. The difficulty is that decisions can have delayed consequences.

We have an agent that acts in an environment. How the environment behaves is unknown to the agent, i.e. how the environment goes from state to state is a black box, its transition probabilities are not known. Furthermore for each action the agent takes it receives a reward from the environment but how these rewards are determined is also not visible to the agent. So the environment is a complete black box to the agent.

Consider the following example as a demonstration of the reinforcement learning framework. In a grid-based office environment a robot is tasked with bringing coffee from the kitchen to a particular office, see figure 1 for the layout.

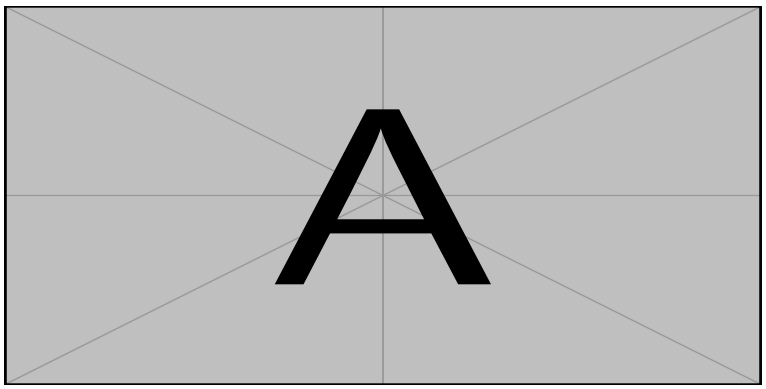

Figure 1: office world...

The environment starts in state $s_0 \in S$ and for each state s_t at time $t \in \mathbb{N}$ the agent has to decide on an action $a_t \in A$ like going up, down, left, right or use the coffee machine. After executing the action the environment changes to a new state s_{t+1} and rewards the robot with a reward $r_{t+1} \in \mathbb{R}$. This feedback loop is shown as a diagram in figure 2.

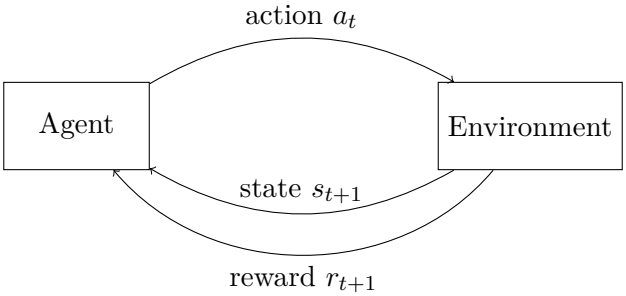

Figure 2: Diagram of the reinforcement learning feedback loop.

The usual way to model this is via a Markov decision process. What represents what? what is the idea, why markov process?? describe probability measure $\mathbb{P}$ and stochastic processes to more rigorously define the quantities and then give the implicit definition using (S, A, P, R, gamma)

Definition 3.1. A *Markov Decision Process (MDP)* is given by a tuple (S, A, P, R, γ), where S is a finite set of *states*, A is a finite set of *actions*, $P : S \times A \times S \to [0, 1]$ is the *transition probability distribution*, $R : S \times A \times S \to \mathbb{R}$ is the *reward function* and $\gamma \in [0, 1]$ is a *discount factor*.

This MDP models the diagram in figure 2 from above as follows: From a given state $s \in S$ the agent chooses an action $a \in A$ and the environment changes to state $s' \in S$ with probability $P_a(s, s')$ and gives reward $R_a(s, s')$. The discount factor will become relevant in a moment but in essence a lower discount factor would motivate the agent to take actions based on the reward sooner rather than later as with a higher discount factor.

In the office world example ...

Definition 3.2. A policy $\pi : S \times A \to [0, 1]$ is a probabiliy distribution that prescribes an action $a \in A$ to be taken for a given state $s \in S$ with probability $\pi(a|s)$.

In the office world example a policy could be to go left when in room 1 and go right when in room 2 or stand still when not in either room. Of course this would not be a good policy, but what constitutes "good" will be defined with the Q-function.

Definition 3.3. The Q-function $q_\pi : S \times A \to \mathbb{R}$ under a policy π is

prove existance of optimal policy

prove Bellman optimality equation for MDP (write about the dynamical programming approach

4 Reward Machines

(15-20 pages)

papers:

- original reward machine paper from 2018 [1]
- connection to LTL from 2019 [2]
- newer reward machine paper from 2022 [3]

include updated RL framework graph to better differentiate that reward machines are not in the environment black box

include same task from previous chapter and how a reward machine for that might look like as a graph

central theorems:

- construction/correspondence to LTL and such
- convergence proof

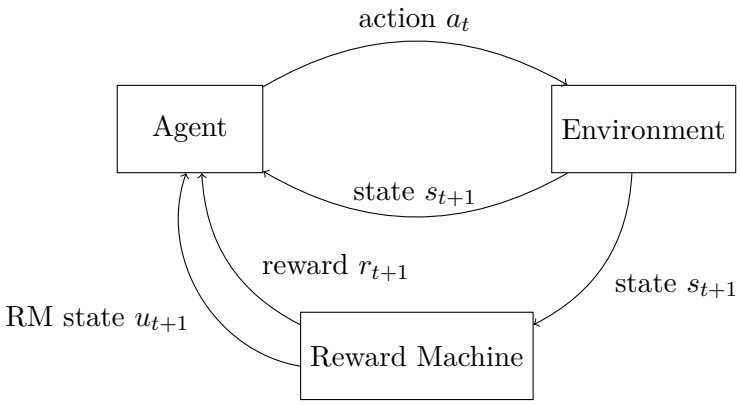

Figure 3: Diagram of the reinforcement learning feedback loop with a reward machine. The agent now not only gets an environment state but also a reward machine state.

The mathematical model for this is ...

Definition 4.1. A *Markov Decision Process with Reward Machine (MDPRM)* is ...

5 Conclusion & Outlook

write this after the main chapters

References

[1] Rodrigo Toro Icarte et al. "Using Reward Machines for High-Level Task Specification and Decomposition in Reinforcement Learning". In: *Proceedings of the 35th International Conference on Machine Learning (ICML)*. 2018, pp. 2112–2121. URL: http://proceedings.mlr.press/v80/icarte18a.html.

[2] Alberto Camacho et al. "LTL and Beyond: Formal Languages for Reward Function Specification in Reinforcement Learning". In: *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*. 2019, pp. 6065–6073. URL: https://www.ijcai.org/proceedings/2019/0840.pdf.

[3] Rodrigo Toro Icarte et al. "Reward Machines: Exploiting Reward Function Structure in Reinforcement Learning". In: *Journal of Artificial Intelligence Research (JAIR)* 73 (2022), pp. 173–208. URL: https://doi.org/10.1613/jair.1.12440.

A Appendix

experiments and code go here convergence graphs of qrm/crm and standard