# Reinforcement Learning
# Exercise 1 - Solution

Jonathan Schnitzler

April 22, 2024

## 1 Formulating Problems

### a) The game of chess

**States** The position of all pieces on the board. A chess board is a 8x8 grid, which starts with 16 white and 16 black pieces on opposing sites. The state space is large (an upper bound from around $\approx 10^{45}$ [1]).

**Actions** The possible moves of the current player. The number of possible moves is limited by the number of pieces on the board and the rules of chess.

#### Reward Signal

- win, lose or draw the game (by checkmate)
- evaluate the current position of the board (e.g. material advantage, positional advantage)

### b) A pick and place robot

#### States

- position and orientation of the axes
- is holding something
- source of objects and destination

#### Actions

- pick
- place
- repeat

[1] see `https://tromp.github.io/chess/chess.html`

**Reward Signal**

- successfully pick and place an object
- time to pick and place an object
- lost an object

## c) A drone which should stabelize in air

**States**

- tilt angle

**Actions**

- adapt speed of individual rotors

**Reward Signal**

- time in air
- minimize the tilt angle
- minimize steering (and energy consumption)

## d) Playing tetris

**States**

- position of the falling block
- position of the other blocks
- preview of next block

**Actions**

- move block left/right
- rotate block

**Reward Signal**

- clear a row
- lose the game

## 2 Value Functions

**a) k-armed Bandits as MDP** The Future rewards are independent on the current state. Since there is only one state, i.e. you are about to roll a bandit, the only thing which could change is our policy based on new estimates of the true distribution - exploration.

This is not considered in the expected reward and therefore its basicly the same $R_t = R$ for all rolls. Then it holds that the sum is just an additional factor, which is not relevant for the value function

$$\sum_{k=0}^{\infty} \gamma^k R_{t+k+1} = R \sum_{k=0}^{\infty} \gamma^k = R \frac{1}{1-\gamma} \tag{1}$$

since $\gamma < 1$.

**b) Proof Relation of value and action-value function** We shall show that

$$v_\pi(s) = \sum_a \pi(a|s) q_\pi(s,a) \tag{2}$$

holds. We can use the definition of the value function

$$v_\pi(s) = \mathbb{E}_\pi[G_t | S_t = s] \tag{3}$$

and the definition of the action-value function

$$q_\pi(s,a) = \mathbb{E}_\pi[G_t | S_t = s, A_t = a] \tag{4}$$

to show the relation. We use the law of total expectation to condition on the action $A_t = a$:

$$\begin{aligned} v_\pi(s) &= \mathbb{E}_\pi[G_t | S_t = s] && (5) \\ &= \sum_a p(A_t = a | S_t = s) \mathbb{E}_\pi[G_t | S_t = s, A_t = a] \cdot && (6) \end{aligned}$$

**c) Rephraze value function**

## 3 Bruteforce the Policy Space

**a) Number of policies** There are $4 \cdot 4 = 16$ tiles, which are the states. There are four actions, namely up, down, left and right. If the policy is deterministic, then for each tile the decision tree branches into four additional options, i.e.

$$n_\pi = 4^{16} = 2^{32} \approx 4 \cdot 10^9 \tag{7}$$

which is doable for a computer but still kind of ridiculos.

**b) See Code**