DRL Homework 1

Hannes Erbis, Marty Schüller

Task 01

The environment is the chess board consisting of 8x8 squares and could be represented with an 8x8 array. Each field has a specific chess piece assigned to it which could be modeled by encoding the specific pieces (i. e. white pawn = 0, black pawn = 1, white knight = 2 and so on) and use them accordingly in the board array.
The state space consists of each square on the chess board. Each square can be represented by their respective x and y coordinates. Hence, the state space is the set of all possible coordinates. So it could look something like this:

$$S = \{(a,1),(a,2),...,(a,8),(b,1),(b,2),...,(b,8),...,(h,8)\} \tag{1}$$

The action space consists of all valid moves. A possible action could be '(pawn, (a,5))'. The action space could therefore look something like this:

$$A = \{(pawn,(a,1)),(pawn,(a,2)),...,(king,(h,8))\} \tag{2}$$

The state transitions can be described by checking if the action is valid and if it is we would move the respective piece to the new field. To check if the move is valid we need to determine if the move can be made based on the underlying chess rules but also if the square where we want to move the piece to is already occupied. If the move is valid we simply move the piece to its new location and see if it otherwise changes the board (e. g. if another piece is being knocked out due to the move). The board state represented with an 8x8 array is then updated accordingly to match the new state.
The reward function could evaluate how strong the current board is for the AI player. This could be done by assigning each piece a respective value (e. g. 1 for pawn, 3 for knight, 5 for rook) and summing the values up for each piece the AI controls. The less pieces the enemy has the higher the reward would be. Additionally, winning a game would yield a high reward.
The policy would be a probability distribution over the action space (equation 2). So for example the move (king, (h,8)) from (g, 7) given a specific board state would be taken with a probability 0.1 while the move (king, (f,8)) could be assigned a probability of 0.7 under the same circumstances.

Task 02

The state space are all possible coordinates of the space ship represented with vectors (x, y). Since the environment is the entire m x n game area, we have:

$$S = \{(0,0),(0,1),...,(0,n),(1,1),(1,2),...,(1,n),...,(m,n)\} \tag{3}$$

The action space consists of the four actions 'do nothing', 'fire left orientation engine', 'fire main engine' and fire right orientation engine.

The state transitions can be described by the physics of the game. The space ship will move according to the action given the games physical properties. If the space ship crashes or lands the round is over. The movement can be described by changing the coordinates of the space ship according to current velocity.
The reward is being determined by the distance from where the space ship landed and the actual landing pad. The further away it lands, the less the reward will be. Firing the main engine will be punished by reducing the reward by 0.3 points per frame it is being used. If the star ship crashed it will receive less rewards than if it landed safely. Solving the problem will yield another 200 points.

Task 03

The reward function is a feedback the agent receives from the environment to evaluate its performance after taking action a at state s. For example in a game of chess the reward function would evaluate the usefulness of each piece and sum up the values of each piece the agent owns and reduces it according to the values of each piece the enemy owns. In the space ship landing game from exercise 2, the reward function is mainly based on the distance between the ship and the landing pad and whether the ship crashed or landed (see exercise 2). The state transition function describes the probability of transitioning to a state s' from state s by taking action a. In chess the transition function would span a probability distribution over the entire chess board for each action at a specific state. In the space ship landing game, the transition function would assign a probability distribution over all game coordinates.
The environment dynamics aren't necessarily known beforehand but should be thought about by humans. When thinking about the reward function it is often required that a human decides which metrics to evaluate in order to reward or punish certain behaviours. The transition function depends on the specific environment and its physics and is therefore derived from the application.