Homework 4. Reinforcement Learning

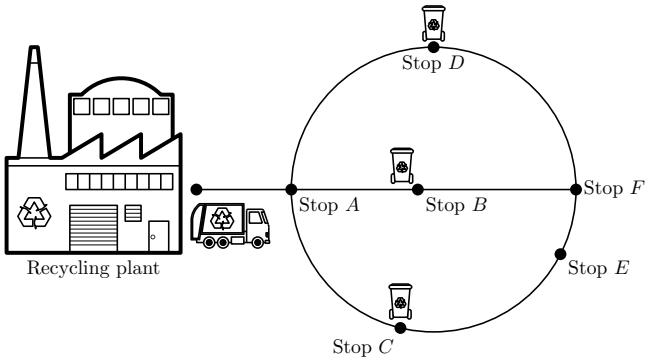

Figure 1: The garbage truck must visit B, C and D before returning to the recycling plant.

Consider the diagram in Fig. 1, representing the scenario from HW2. The building on the left corresponds to a recycling plant. A truck leaves the recycling plant and must traverse the road network depicted in the diagram, making sure to visit stops B, C and D before returning to the recycling plant, so as to collect the garbage in these locations. In each location, the driver has six actions available:

- Collect garbage. Each of the stops B, C, and D is considered visited only after this action is executed at that location. However, if the location has already been visited, the action has no effect. In the remaining locations, the action has no effect.

- Drop garbage. In stops A through F, this action has no effect. In the recycling plant, this action successfully deposits the collected garbage *only if* stops B, C, and D have been visited since the last garbage drop.[1]

[1]Note that, as soon as a successful drop takes place, the truck driver should restart the whole process, i.e., visit once again stops B, C, and D to collect garbage, then drop that garbage in the recycling plant, and so on. In other words, as soon as a successful drop takes place, the MDP should "reset" to the initial configuration (truck in the recycling plant and stops B, C and D "unvisited").

- Move up. In the recycling plant and in stops B, C, D, and E, this action has no effect. In stops A and F, it moves the truck towards stop D;
- Move down. In the recycling plant and in stops B, C, D, and E, this action has no effect. In stop A it moves the truck towards stop C; in stop F it moves the truck towards stop E.
- Move left. In the recycling plant, this action has no effect. In all other locations, it moves the truck to the adjacent location to the left.
- Move right. In stop F, this action has no effect. In all other locations, it moves the truck to the adjacent location to the right.

As seen in HW2, the driver's decision can be modeled as a Markov decision problem where the state space $\mathcal{X}$ consists of all tuples (p, c_B, c_C, c_D), with $p \in \{R, A, B, C, D, E, F\}$ and $c_B, c_C, c_D \in \{0, 1\}$. The component p indicates the *position* of the truck, while components c_B, c_C, and c_D are binary indicators of whether the garbage at stops B, C, and D has been collected, respectively (i.e., they have been "visited"). The action space, in turn, is given by $\mathcal{A} = \{\text{collect}, \text{drop}, U, D, L, R\}$.

Suppose that the driver is allowed to move around the environment for T time steps, experimenting the different actions and visiting the different states. An excerpt of its trajectory is

$$\begin{gathered}
\dots \\
x_{t-1} = (E, 1, 0, 1), \\
a_{t-1} = D, \\
c_{t-1} = 1.0, \\
x_t = (E, 1, 0, 1), \\
a_t = R, \\
c_t = 0.2, \\
x_{t+1} = (F, 1, 0, 1), \\
a_{t+1} = R, \\
c_{t+1} = 1.0, \\
\dots
\end{gathered}$$

At time step t, the Q-values estimated by the agent for state $(E, 1, 0, 1)$ are:

$$\mathbf{Q}^{(t)}_{(E,1,0,1),:} = \begin{bmatrix} 2.8 & 2.8 & 2.8 & 2.8 & 2.54 & 2.0 \end{bmatrix}$$

and for state $(F, 1, 0, 1)$,

$$\mathbf{Q}^{(t)}_{(F,1,0,1),:} = \begin{bmatrix} 2.8 & 2.8 & 2.95 & 2.0 & 3.14 & 2.8 \end{bmatrix}.$$

Exercise 1.

(a) Indicate the Q-values after a Q-learning update with step-size $\alpha = 0.1$, resulting from the transition at time step t. Use $\gamma = 0.9$.

(b) Indicate the Q-values after a SARSA update with step-size $\alpha = 0.1$, resulting from the transition at time step t. Use $\gamma = 0.9$.

(c) Explain the difference between on-policy and off-policy learning using Questions 1a and 1b to illustrate your explanation.