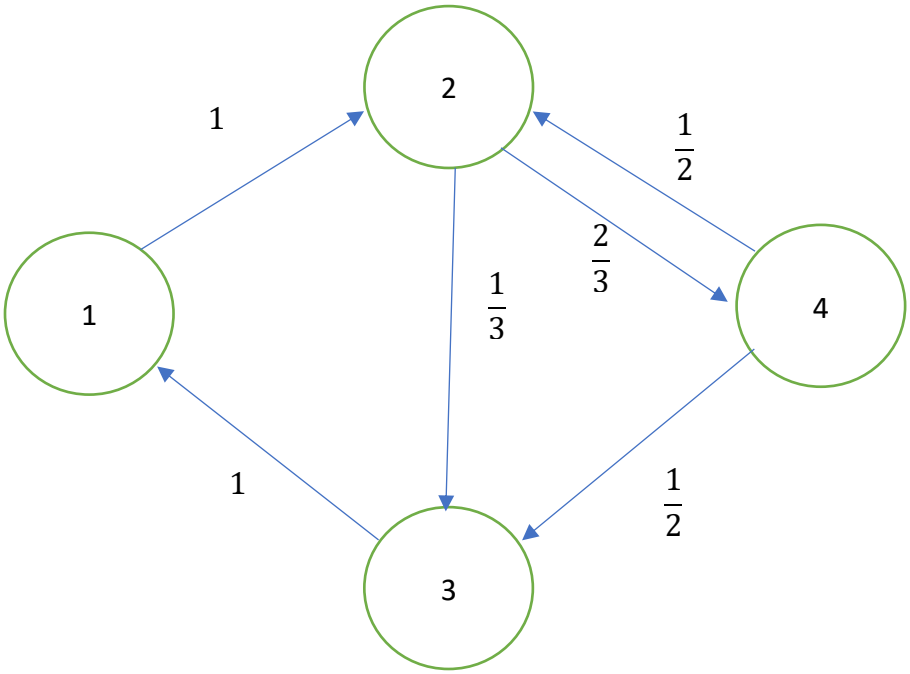
# Reinforcement Learning Assignment 2

Intro RL – EX2

Theoretical section:

Question 1:

1.




2. The communicating class is: $\{1,2,3,4\}$ and since all the states of the chain are included the chain is indeed irreducible.

3. $d_i$ is defined as follows:

$$d_i := \gcd\{m \geq 1 : p_{\{i,i\}}^m > 0\}$$

And in words, the gcd of the lengths of the different paths leading from each state to itself with probability greater than 0. Below are the different $d_i$ for our chain:

$$d_1 = \gcd\{3,4,\dots\} = 1$$

Since all the states are in the same class, this applies to all states, i.e:
$\forall i \in \{1,2,3,4\}. d_i = 1.$

The chain is aperiodic since it's irreducible and there exists a state i for which $d_i = 1$.

4. Since the transition matrix P represents an irreducible a-periodic MC, we know that P Is positive and hence there exists $\mu$ s.t.:

$$\mu^T P = \mu^T$$

So we can solve:

$$(\mu_1, \mu_2, \mu_3, \mu_4)\begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{1}{3} & \frac{2}{3} \\ 1 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & \frac{1}{2} & 0 \end{pmatrix} = (\mu_1, \mu_2, \mu_3, \mu_4)$$

And we get:

$$\mu_3 = \mu_1$$

$$\mu_1 + \frac{1}{2}\mu_4 = \mu_2$$

$$\frac{1}{3}\mu_2 + \frac{1}{2}\mu_4 = \mu_3$$

$$\frac{2}{3}\mu_2 = \mu_4$$

We also know that:

$$\mu_1 + \mu_2 + \mu_3 + \mu_4 = 1$$

This yields:

$$\mu_1 = 0.22$$
$$\mu_2 = 0.33$$
$$\mu_3 = 0.22$$
$$\mu_4 = 0.22$$

5. Since the chain is irreducible, aperiodic and has a finite state space, there is a unique invariant distribution which is defined by:

$$\mu_i = \frac{1}{E[T_i]}$$

Since we calculated $\mu_i$ *we get*:

$$E[T_1] = \frac{1}{0.22} = 4.54$$

$$E[T_2] = \frac{1}{0.33} = 3$$

$$E[T_3] = \frac{1}{0.22} = 4.54$$

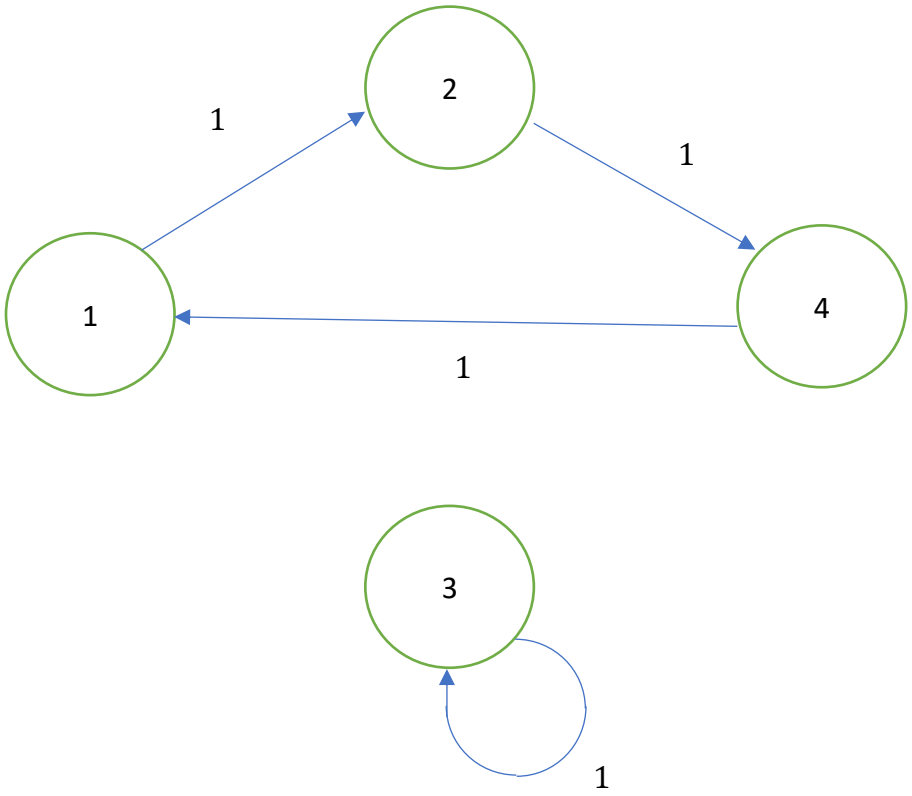
$$E[T_4] = \frac{1}{0.22} = 4.54$$

All states are positive recurrent (as expected for a irreducible aperiodic finite MC).

6. The new matrix will be:

$$P' = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

And the matching graph:




The value of $p_{1,1}^{m}$ can be calculated by calculating $P_{[1,1]}^{m}$:

$$P'^{2} = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

$$P'^3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

$$P'^4 = P', P'^5 = P'^2, P'^6 = P'^3 \ldots$$

And accordingly:

$$p_{1,1}^m = 1 \; if: m \; mod 3 = 0$$
$$p_{1,1}^m = 0 \; if \; m \; mod 3 \neq 0$$

**Question 2:**

**a)** $S = \{0,1,\dots,2k-1\}$ $A = \{CW, CCW\}$

$$\forall i, v \in S \quad P(v|i, CW) = \begin{cases} 1 \; if \; v = i + 1 mod(2k) \\ 0 \qquad\qquad\qquad else \end{cases}$$

$$\forall i, s \in S \quad P(s|i, CCW) = \begin{cases} 1 \; if \; v = i - 1 mod(2k) \\ 0 \qquad\qquad\qquad else \end{cases}$$

$$\forall s \in S, \forall a \in \{CW, CCW\} \; R(s,a) = \begin{cases} 1 \; if \; s = 0 \\ 0 \qquad else \end{cases}$$

$$s_0 = \{k\}$$

**b)** From the definition of the MDP, we can observe that the optimal policy is to get as fast as possible to state $s = 0$ and then to iterate from 0 to 1. Thus the definition of the policy is:

$$\pi^*(s) = \begin{cases} CW \;\; if k+1 \le s \le 2k-1 \\ CCW \qquad if \; 0 \le s \le k \end{cases}$$

**c)** We will demonstrate one iteration of VI algorithm:

Initiate $\forall s \in S \quad V_0(s) = 0$.

$$V_1(s) = max_{a \in A}\{r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V_0(s')\}$$

Since $V_0(s') = 0$ we get: $V_1(s) = \max\limits_{a \in A}\{r(s,a)\}$ which means that $V_1(s) = \begin{cases} 1 \; if \; s = 0 \\ 0 \qquad else \end{cases}$.

So only state 0 will change its value after one iteration from 0 to 1.

**d)** At the second iteration of the algorithm we get:

$$V_2(s) = max_{a \in A}\{r(s,a) + \gamma \sum_{s' \in S} p(s'|s,a) V_1(s')\}$$

If $s = 0$ then:

$$V_2(0) = max_{a \in A}\{r(0,a) + \gamma \sum_{s' \in S} p(s'|0,a) V_1(s')\} = \max_{a \in A}\{r(0,a)\} = 1$$

For the case where $s = 1$ we get:

$$V_2(1) = max_{a \in A}\{r(1,a) + \gamma \sum_{s' \in S} p(s'|1,a) V_1(s')\} = max_{a \in A}\{\gamma p(0|1,a)\} = \gamma$$

And similarly for $s = 2k-1$:

$$V_2(2k-1) = max_{a \in A}\{r(1,a) + \gamma \sum_{s' \in S} p(s'|2k-1,a) V_1(s')\}$$
$$= max_{a \in A}\{\gamma p(0|2l-1,a)\} = \gamma$$

For $s \neq 0,1,2k-1$ we have from the first iteration that $\forall s' \in S \;\; p(s'|s,a)V_1(s') = 0$ and $\forall a \in A \;\; r(s,a) = 0$ thus $V_2(s) = 0$.

So the states which changed their value after the second iteration are:
$V_2(1) = V_2(2k-1) = \gamma$

**e)** Continue with policy iteration algorithm:

$$V_3(0) = max_{a\in A}\{r(0,a) + \gamma \sum_{s'\in S} p(s'|0,a)V_2(s')\} = max_{a\in A}\{r(0,a) + \gamma p(1|0,a)V_2(1) + \gamma p(3|0,a)V_2(3)\}$$
$$= 1 + \gamma^2$$

$$V_3(1) = max_{a\in A}\{r(1,a) + \gamma \sum_{s'\in S} p(s'|1,a)V_2(s')\} = max_{a\in A}\{\gamma p(0|1,a)V_2(0) + \gamma p(2|1,a)V_2(2)\} = \gamma$$

$$V_3(2) = max_{a\in A}\{r(2,a) + \gamma \sum_{s'\in S} p(s'|2,a)V_2(s')\} = max_{a\in A}\{\gamma p(1|2,a)V_2(1) + \gamma p(3|2,a)V_2(3)\} = \gamma^2$$

$$V_3(3) = max_{a\in A}\{r(1,a) + \gamma \sum_{s'\in S} p(s'|1,a)V_2(s')\} = max_{a\in A}\{\gamma p(0|1,a)V_2(0) + \gamma p(2|1,a)V_2(2)\} = \gamma$$

For the fourth iteration we get:

$$V_4(0) = max_{a\in A}\{r(0,a) + \gamma \sum_{s'\in S} p(s'|0,a)V_3(s')\} = max_{a\in A}\{1 + \gamma p(1|0,a)V_3(1) + \gamma p(3|0,a)V_3(3)\}$$
$$= 1 + \gamma^2$$

$$V_4(1) = max_{a\in A}\{r(1,a) + \gamma \sum_{s'\in S} p(s'|1,a)V_3(s')\} = max_{a\in A}\{\gamma p(0|1,a)V_3(0) + \gamma p(2|1,a)V_3(2)\} = \gamma + \gamma^3$$

$$V_4(2) = max_{a\in A}\{r(2,a) + \gamma \sum_{s'\in S} p(s'|2,a)V_3(s')\} = max_{a\in A}\{\gamma p(1|2,a)V_3(1) + \gamma p(3|2,a)V_3(3)\} = \gamma^2$$

$$V_4(3) = max_{a\in A}\{r(3,a) + \gamma \sum_{s'\in S} p(s'|3,a)V_3(s')\} = max_{a\in A}\{\gamma p(0|3,a)V_3(0) + \gamma p(2|3,a)V_3(2)\} = \gamma + \gamma^3$$

Notice that due to the symmetry of the graph: $V_n(1) = V_n(3)\ and\ V_n(0) \geq V_n(2)$.

Continue for the fifth iteration:

$$V_5(0) = max_{a\in A}\{1 + \gamma p(1|0,a)V_4(1) + \gamma p(3|0,a)V_4(3)\} = 1 + \gamma^2 + \gamma^4$$
$$V_5(1) = max_{a\in A}\{\gamma p(0|1,a)V_4(0) + \gamma p(2|1,a)V_4(2)\} = \gamma + \gamma^3 = V_5(3)$$
$$V_5(2) = max_{a\in A}\{\gamma p(1|2,a)V_4(1) + \gamma p(3|2,a)V_4(3)\} = \gamma^2 + \gamma^4$$

From this point we can see the following trending of $V_n(s)$ which can be proved by induction:

For odd $n$, $V_n(0) = \gamma V_{n-1}(1) + 1$ and $V_n(2) = \gamma V_{n-1}(1)$.

For even $n$, $V_n(1) = \gamma V_{n-1}(0)$.

So taking the limit for $n \to \infty$ we get:

$$V^*(0) = \sum_{i=0}^{\infty} \gamma^{2i} = \frac{1}{1-\gamma^2}$$

$$V^*(1) = V^*(3) = \sum_{i=0}^{\infty} \gamma^{2i+1} = \frac{\gamma}{1-\gamma^2}$$

$$V^*(2) = \sum_{i=0}^{\infty} \gamma^{2i+2} = \frac{\gamma^2}{1-\gamma^2}$$

**Question 3:**

**a)** Define the state space to be $S = \{(K, d) | K \subset \{0, \dots, N-1\}, d \in \{0, \dots, 9\}\}$. Meaning that a state $s \in S$ is a pair of $K$ which is a subset of $\{0, \dots, N-1\}$ that represents the locations of $\times$, and a digit $d$.

The action space depends on the state $s = (K, d)$ is $A(s) = \{n | b \in S\}$, which represents choosing an element of $K$ that we will put the digit $d$.

For a state $s \in S$ and action $a \in A(s)$ define $p((K', d') | (K, d), a)$ to be:

$p((K', d') | (K, d), a) = \begin{cases} 0 & if\ K \backslash \{a\} \neq K' \\ \frac{1}{10} & else \end{cases}$ and the transition law is $P = \{p(s' | s, a) | s' \in S\}$.

The reward $R_t$ for $t < N-1$ will be $r_t\big((K_t, d_t), a_t\big) = 10^{a_t} d_t$. For the terminal state $t = N-1$ we must have $K = \emptyset$ and $r_{N-1}\big((\emptyset, d)\big) = 0$

Lastly, define $S_0 = \{(\{0, \dots, N-1\}, d) | d \in \{0, \dots, 9\}\}$ and $\forall s_0 \in S_0 \ \ p_0(s_0) = \frac{1}{10}$.

**b)** The fact that the optimal policy depends only on empty slots and the random digit is inherently defined in our model. For any policy $\pi$ the legal actions of state $s = (K, d)$ are elements of $K$ which represents the empty slots, and can be chosen with respect to $K$ and $d$. Moreover, it does not depend on the time $t$ since by the definition of $K$ and the transition law, at time $t$ we have that $|K| = N - 1 - t$, so the same state cannot appear in different time $t$.

Since any policy depends only on the empty slots and the random digit then in particular the optimal policy depends only on them.

**c)** We will use backward recursion in order to calculate the optimal policy.

For any terminal state $s \in S_3$ the reward $r(s) = 0$ since $K = \emptyset$. So $V_3(s) = 0$.

When $t = 2$ we have $|K| = 1$ so by definition $|A(s)| = 1$ so any policy must choose the single legal action. Hence $V_2\big((\{k\}, d)\big) = 10^k d$.

For $t = 1$ we have the cases where $K = \{0,1\}, \{0,1\}, \{1,2\}$:

If $K = \{0,1\}$ then:

$$V_1(s) = \max_{a \in A(s)} \{r_2\big((K, d), a\big) + \sum_{d'=0,..,9} p\big((K \backslash \{a\}, d') | s, a\big) V_2((K \backslash \{a\}, d')) =$$

$$= max\{\, d + \frac{1}{10}(10 + 20 + \cdots + 90), 10d + \frac{1}{10}(1 + 2 + \cdots + 9)\} =$$

$\max\{d + 45, 10d + 4.5\}$

So for such $K$ the optimal policy is $\pi^*\big((K,d)\big) = \begin{cases} 0 \; if \; d < 5 \\ 1 \; if \; d \geq 5 \end{cases}$

For $K = \{0,2\}$:

$$V_2(s) = max\{\, d + \frac{1}{10}(100 + 200 + \cdots + 900), 100d + \frac{1}{10}(1 + 2 + \cdots + 9)\} =$$
$$= \max\{d + 450, 100d + 4.5\}$$

So for such $K$ the optimal policy is $\pi^*\big((K,d)\big) = \begin{cases} 0 \; if \; d < 5 \\ 1 \; if \; d \geq 5 \end{cases}$

For $K = \{1,2\}$:

$$V_2(s) = max\{\, d + \frac{1}{10}(100 + 200 + \cdots + 900), 100d + \frac{1}{10}(1 + 2 + \cdots + 9)\} =$$
$$= \max\{10d + 450, 100d + 45\}$$

So for such $K$ the **optimal policy is** $\pi^*\big((K,d)\big) = \begin{cases} 1 \; if \; d < 5 \\ 2 \; if \; d \geq 5 \end{cases}$

To conclude the optimal policy for $t = 1$ we have:

$$\pi^*\big((K,d)\big) = \begin{cases} min\{k \in K\} \; if \; d < 5 \\ max\{k \in K\} \; if \; d \geq 5 \end{cases}$$

For $t = 0$ $K$ must equal $\{0,1,2\}$. Hence:

$$V_0(s) = \max_{a \in A(s)} \{r_2\big((K,d),a\big) + \sum_{d'=0,..,9} p\big((K\backslash\{a\},d')|s,a\big) V_1((K\backslash\{a\},d')) =$$
$$max\left\{d + \frac{1}{10}\sum_{d'=0}^{9} V_2\big((\{1,2\},d')\big), 10d + \frac{1}{10}\sum_{d'=0}^{9} V_2\big((\{0,2\},d')\big), 100d + \frac{1}{10}\sum_{d'=0}^{9} V_2\big((\{0,1\},d')\big)\right\} =$$
$$\max\{d + \frac{1}{10}6075, 10d + \frac{1}{10}5782.5, 100d + \frac{1}{10}607.5\}$$

So **the optimal policy** in this case is:

$$\pi^*\big((K,d)\big) = \begin{cases} 0 & if \; d < 4 \\ 1 \; if & 4 \leq d \leq 5 \\ 2 & if \; d > 5 \end{cases}$$

Reinforcement, H.W. 2

**(Q.4):**

- The operator: $T(V)(s) = \frac{1}{|A|} \sum_{a \in A} \left( r(s,a) + \gamma \cdot \sum_{s' \in S} P(s'|s,a) V(s') \right)$

- Proof:

$\forall V_1, V_2 \in \mathbb{R}^{|S|}, \forall s \in S$:

$$|T(V_1)(s) - T(V_2)(s)| = \frac{1}{|A|} \cdot \left| \sum_{a \in A} r(s,a) + \gamma \sum_{s' \in S} \sum_{a \in A} P(s'|s,a) V_1(s') - \sum_{a \in A} r(s,a) - \gamma \sum_{a \in A} \sum_{s' \in S} P(s'|s,a) \cdot V_2(s') \right|$$

$$= \frac{1}{|A|} \gamma \cdot \left| \sum_{a \in A} \sum_{s' \in S} P(s'|s,a) \cdot (V_1(s') - V_2(s')) \right| \leq$$

$$\leq \frac{1}{|A|} \gamma \cdot \sum_{a \in A} \sum_{s' \in S} P(s'|s,a) \left| V_1(s') - V_2(s') \right| \leq$$

$$\leq \frac{\gamma}{|A|} \sum_{a \in A} \sum_{s' \in S} P(s'|s,a) \cdot \|V_1 - V_2\|_\infty = \quad (P(s'|s,a) = 1)$$

$$= \frac{\gamma}{|A|} \sum_{a \in A} \|V_1 - V_2\|_\infty = \frac{|A| \cdot \gamma}{|A|} \cdot \|V_1 - V_2\|_\infty =$$

$$= \gamma \cdot \|V_1 - V_2\|_\infty .$$

$\Rightarrow$ Hence, $\|T(V_1) - T(V_2)\|_\infty \leq \gamma \cdot \|V_1 - V_2\|_\infty$.

□

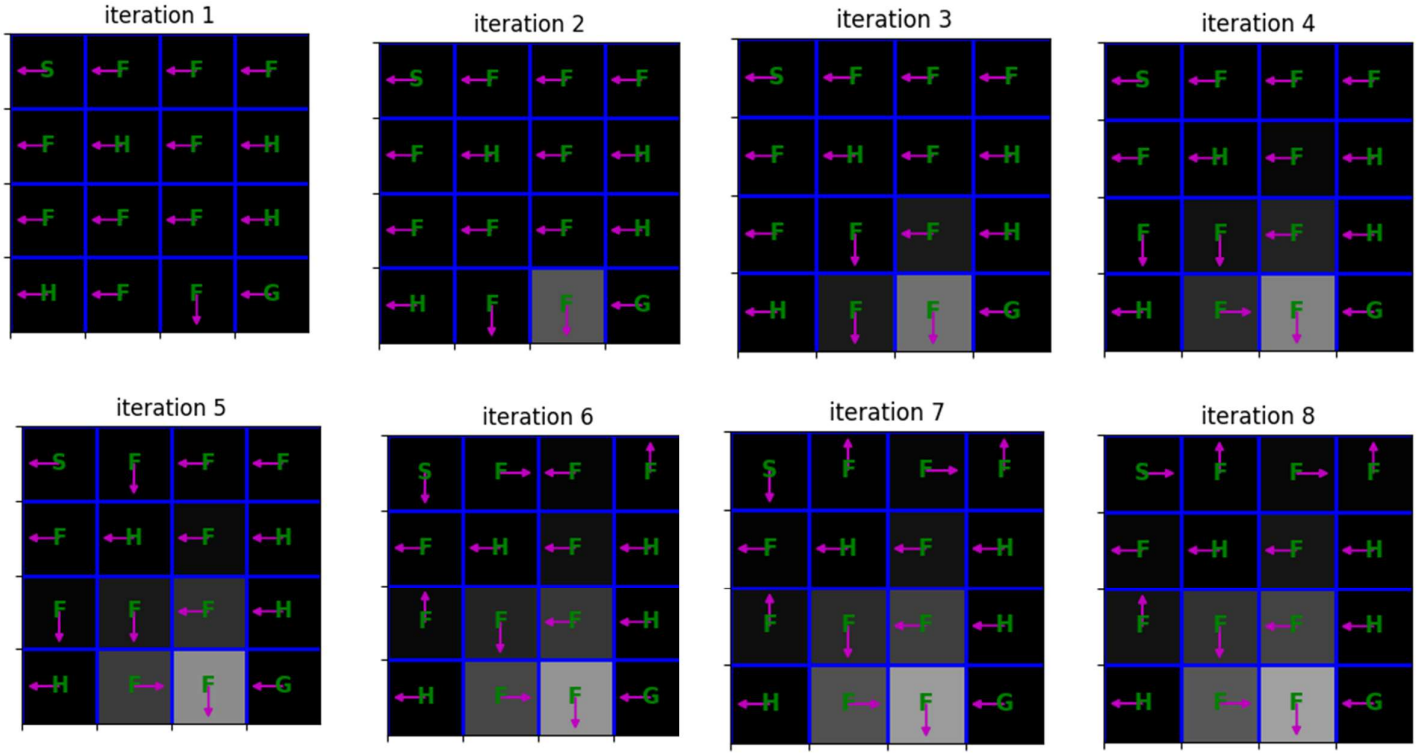
**Programming**

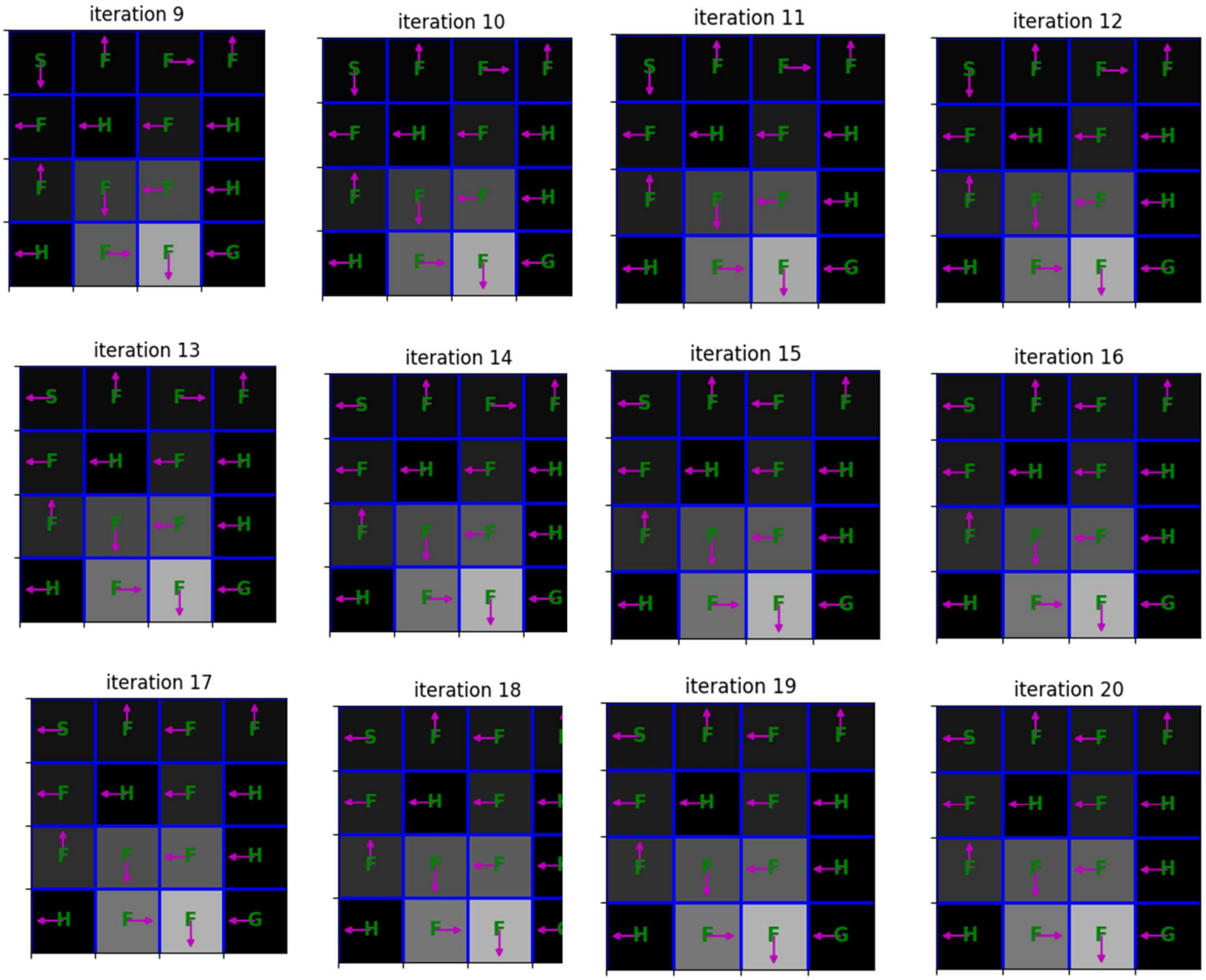
**Question 1:**

Iteration table:

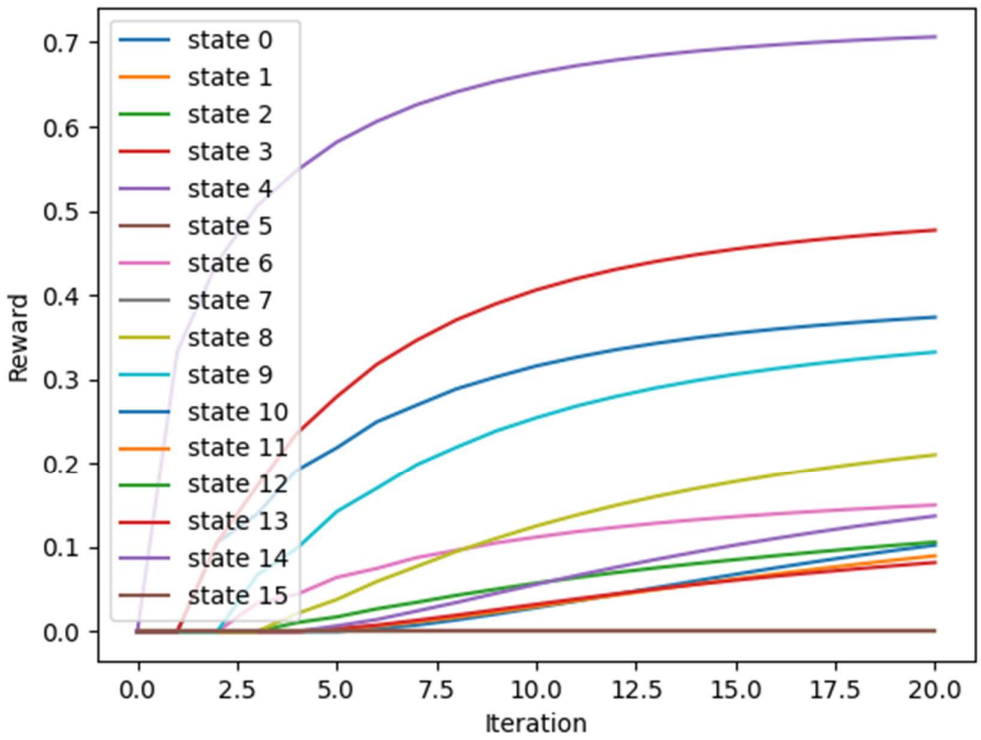
| Iteration | max\|V-Vprev\| | # chg actions | V[0] |
|---|---|---|---|
| 0 | 0.33333 | N/A | 0.000 |
| 1 | 0.10556 | 1 | 0.000 |
| 2 | 0.06685 | 1 | 0.000 |
| 3 | 0.06351 | 2 | 0.000 |
| 4 | 0.04357 | 1 | 0.000 |
| 5 | 0.03821 | 4 | 0.003 |
| 6 | 0.02857 | 2 | 0.008 |
| 7 | 0.02437 | 1 | 0.014 |
| 8 | 0.01952 | 1 | 0.021 |
| 9 | 0.01624 | 0 | 0.028 |
| 10 | 0.01384 | 0 | 0.036 |
| 11 | 0.01173 | 0 | 0.044 |
| 12 | 0.01047 | 1 | 0.052 |
| 13 | 0.00948 | 0 | 0.060 |
| 14 | 0.00852 | 1 | 0.068 |
| 15 | 0.00782 | 0 | 0.075 |
| 16 | 0.00733 | 0 | 0.083 |
| 17 | 0.00694 | 0 | 0.090 |
| 18 | 0.00656 | 0 | 0.096 |
| 19 | 0.00618 | 0 | 0.102 |

Optimal policy at each iteration:






State value per iteration:



**Question 2:**

Iteration table:

| Iteration | # chg actions | V[0] |
|---|---|---|
| 0 | 1 | 0.00000 |
| 1 | 6 | 0.00000 |
| 2 | 6 | 0.00000 |
| 3 | 5 | 0.04421 |
| 4 | 3 | 0.14475 |
| 5 | 0 | 0.18047 |
| 6 | 0 | 0.18047 |
| 7 | 0 | 0.18047 |
| 8 | 0 | 0.18047 |
| 9 | 0 | 0.18047 |
| 10 | 0 | 0.18047 |
| 11 | 0 | 0.18047 |
| 12 | 0 | 0.18047 |
| 13 | 0 | 0.18047 |
| 14 | 0 | 0.18047 |
| 15 | 0 | 0.18047 |
| 16 | 0 | 0.18047 |
| 17 | 0 | 0.18047 |
| 18 | 0 | 0.18047 |
| 19 | 0 | 0.18047 |

Value plot binberr@bgu.ac.il: