Reinforcement Learning Homework 01

Qiu Yihang

Feb 2023

1 Convergence of Policy Iteration

Proof. Let the state space be $\mathcal{S}$, the action space be $\mathcal{A}$.

To prove the new policy produced by policy iterations will be at least as good as the original one, we need to prove

$$V^{\pi_{i+1}}\left(s\right) \geq V^{\pi_i}\left(s\right).$$

For $V^{\pi_i}(s)$, by the process of policy iteration, we know

$$\sum_{s'} P\left(s, \pi_i(s), s'\right)\left[R\left(s, \pi_i(s), s'\right) + \gamma V^{\pi_i}(s')\right] \leq \sum_{s'} \max_{a \in \mathcal{A}} P\left(s, \pi_i(s), s'\right)\left[R\left(s, \pi_i(s), s'\right) + \gamma V^{\pi_i}(s')\right]$$

Thus,

$$\begin{aligned}
V^{\pi_i}(s) &\leq Q(s, \pi_{i+1}(s)) \\
&= \mathbb{E}_{\pi_{i+1}}\left[R(s_0, \pi_{i+1}(s_0), s_1) + \gamma V^{\pi_i}(s_1) \mid \pi_i, s_0 = s\right] \\
&\leq \mathbb{E}_{\pi_{i+1}}\left[R(s_0, \pi_{i+1}(s_0), s_1) + \gamma Q(s_1, \pi_{i+1}(s_1)) \mid \pi_i, s_0 = s\right] \\
&= \mathbb{E}_{\pi_{i+1}}\left[R(s_0, \pi_{i+1}(s_0), s_1) + \gamma R(s_1, \pi_{i+1}(s_1), s_2) + \gamma^2 \cdot V^{\pi_i}(s_1) \mid \pi_i, s_0 = s\right] \\
&\leq \mathbb{E}_{\pi_{i+1}}\left[R(s_0, \pi_{i+1}(s_0), s_1) + \gamma R(s_1, \pi_{i+1}(s_1), s_2) + \gamma^2 Q(s_2, \pi_{i+1}(s_2)) \mid \pi_i, s_0 = s\right] \\
&= \dots \\
&\leq \mathbb{E}\left[\sum_{t=0}^{\infty} R(s_t, \pi_{i+1}(s_t), s_{t+1})\right] = V^{\pi_{i+1}}(s)
\end{aligned} \tag{1}$$

Moreover, $V^{\pi_{i+1}}(s) = V^{\pi_i}(s)$ **iff.**

$$\begin{aligned}
&\forall s' \in \mathcal{S}, \max_a P(s, a, s')\left[R(s, a, s') + \gamma V^{\pi_i}(s')\right] = P\left(s, \pi_i(s), s'\right)\left[R\left(s, \pi_i(s), s'\right) + \gamma V^{\pi_i}(s')\right] \\
&\qquad \text{i.e. } \pi_i(s) \in \arg\max_a \sum_{s'} Q^{\pi_i}(s, a), \text{ i.e. } \pi_i(s) \text{ is already an optimal policy.}
\end{aligned} \tag{2}$$

Thus, the new policy is at least as good as the original one. □

We prove the convergence of policy iterations as follows.

Let the set of optimal policies be Π^*.

Given that the state space, the action space and the reward function is all finite, we have

$$\forall s, s' \in \mathcal{S}, a \in \mathcal{A},\ R_- \leq R(s, a, s') < R_+.$$

Then for any possible policy π, since $0 \leq \gamma < 1$,

$$\frac{R_-}{1-\gamma} \leq V^\pi(s) < \sum_{t=0}^{\infty} \gamma^t \cdot R_0 = \frac{R_+}{1-\gamma}$$

By 1 and 2, we know

$$\begin{cases} V^{\pi_i} = V^{\pi_{i+1}} & \textbf{iff. } \forall s \in \mathcal{S}, \pi_i(s), \pi_{i+1}(s) \in \arg\max_a \sum_{s'} Q^{\pi_i}(s,a). \\ \exists s \in \mathcal{S}, V^{\pi_i}(s) < V^{\pi_{i+1}}(s) & \text{otherwise} \end{cases}$$

i.e.

$$\begin{cases} V^{\pi_i} = V^{\pi_{i+1}} & \textbf{iff. } \pi_i, \pi_{i+1} \in \Pi^*. \\ \exists s \in \mathcal{S}, V^{\pi_i}(s) < V^{\pi_{i+1}}(s) & \text{otherwise} \end{cases}$$

Meanwhile, there is at most $|\mathcal{S}| \times |\mathcal{A}|$ polices. Then for $N \geq |\mathcal{S}| \times |\mathcal{A}| + 1$, we know

$$\forall s \in \mathcal{S}, \exists k_s \in \mathbb{N}, V^{\pi_1}(s) < V^{\pi_2}(s) < ... < V^{\pi_{k_s - 1}}(s) = V^{\pi_{k-s}}(s) = ... = V^{\pi_N}(s).$$

i.e.

$$V^{\pi_{N-1}} = V^{\pi_N} \Longrightarrow \pi_{N-1}(s) \in \Pi^*, \pi_N(s) \in \Pi^*.$$

Moreover, by the definition of how π_{i+1} derives from π_i, we know when $\pi_N \in \Pi^*$, $\pi_{N+1} = \pi_N$.

Therefore, when $i \to \infty, \pi_i \to \pi_N \in \Pi^*$.

Thus, the policy iteration converges to an optimal policy. ■

2 Grid World

2.1 Value Iteration

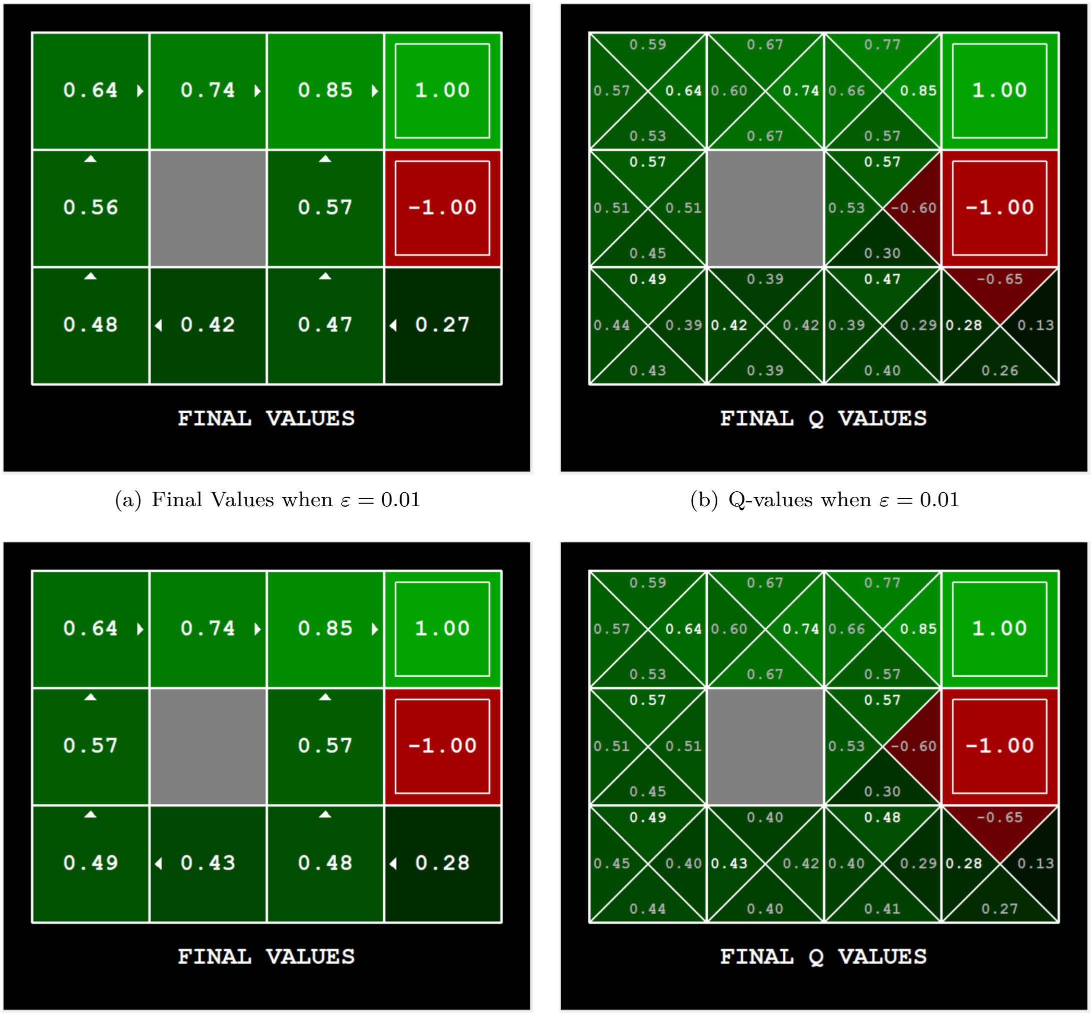
The value iteration is implemented in `valueIterationAgent.py`.

The results given by value iteration under $\varepsilon = 0.01$ or 0.001 are shown in Fig. 1.

2.2 Policy Iteration

The policy iteration is implemented in `policyIterationAgent.py`.

The results given by policy iteration under $\varepsilon = 0.01$ or 0.001 are shown in Fig. 2.



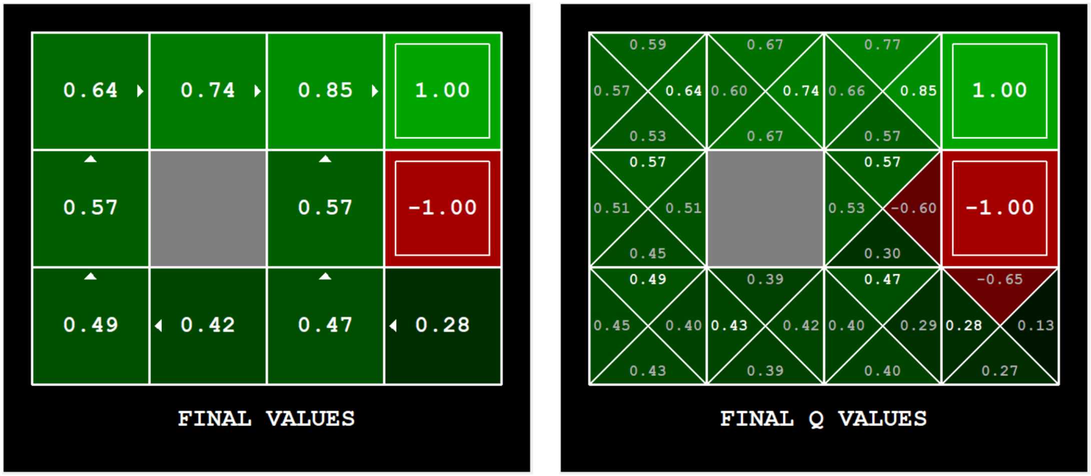
(a) Final Values when $\varepsilon = 0.01$

(b) Q-values when $\varepsilon = 0.01$

(c) Final Values when $\varepsilon = 0.001$

(d) Q-values when $\varepsilon = 0.01$

Figure 1: Value Iteration



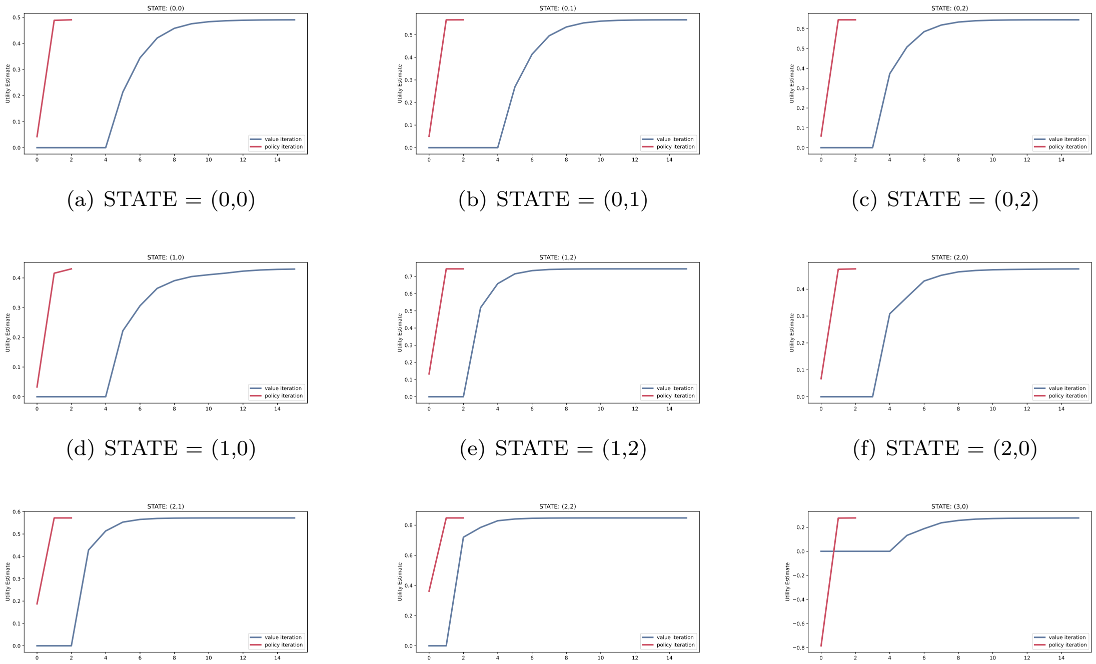
(a) Final Values when $\varepsilon = 0.01$ or 0.001

(b) Q-values when $\varepsilon = 0.01$ or 0.001

Figure 2: Policy Iteration

2.3 Comparison of Speed of Convergence

The maximal change of value function during the iteration process is depicted in Fig. 3.



(a) STATE = (0,0) (b) STATE = (0,1) (c) STATE = (0,2)

(d) STATE = (1,0) (e) STATE = (1,2) (f) STATE = (2,0)

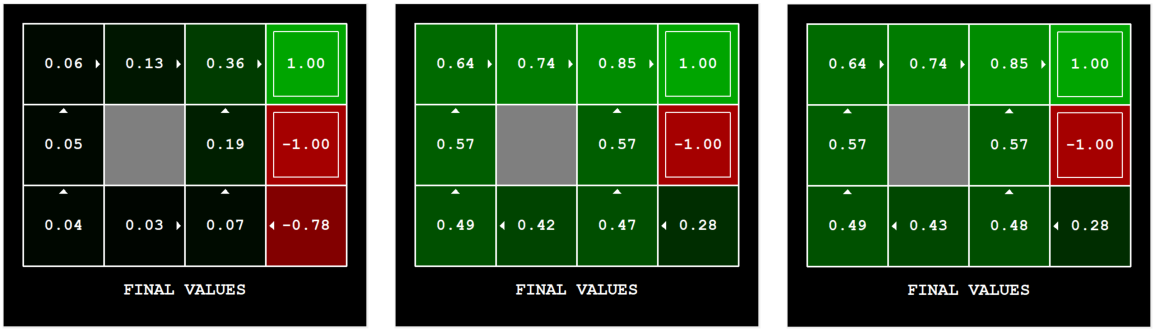
(g) STATE = (2,1) (h) STATE = (2,2) (i) STATE = (3,0)

Figure 3: Convergence of utilities of all states under value iteration and policy iteration

As shown above, in the aspect of utility, policy iteration **converges faster than** value iteration. Moreover, closer a state is to the terminal state, faster the value iteration of the utility of it converges.

* Still, it seems a little unfair to compare in such ways. In the process of policy iteration, value estimate also iterates for several times to estimate the value function of the current policy. But it is also possible to estimate the value by solving a linear function, which won't lead to large time consumption.

The policy actions of all states against the number of iterations are as follows. For policy iteration, policy actions are shown in Fig. 4. For value iteration, policy actions are shown in Fig. 5 and Fig. 6.



0.06 ▸	0.13 ▸	0.36 ▸	1.00
▴ 0.05		▴ 0.19	-1.00
▴ 0.04	0.03 ▸	▴ 0.07	◂ -0.78

FINAL VALUES

0.64 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.57		▴ 0.57	-1.00
▴ 0.49	◂ 0.42	▴ 0.47	◂ 0.28

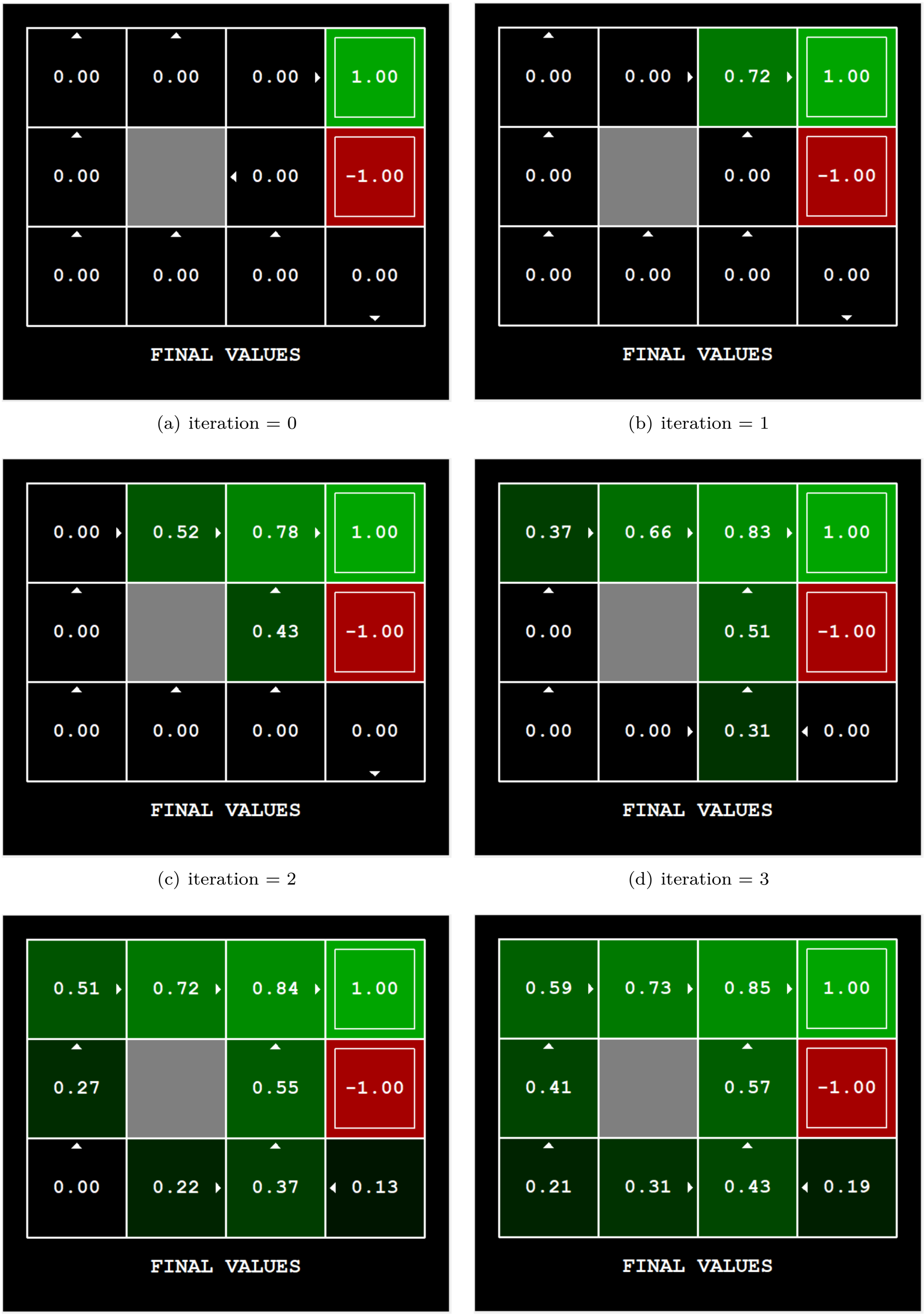
FINAL VALUES

0.64 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.57		▴ 0.57	-1.00
▴ 0.49	◂ 0.43	▴ 0.48	◂ 0.28

FINAL VALUES

(a) iteration=0 (b) iteration=1 (c) iteration=2

Figure 4: Policy Actions given by Policy Iterations



▲ 0.00	▲ 0.00	0.00 ▶	1.00
▲ 0.00		◀ 0.00	-1.00
▲ 0.00	▲ 0.00	▲ 0.00	0.00 ▼

FINAL VALUES

(a) iteration = 0

▲ 0.00	0.00 ▶	0.72 ▶	1.00
▲ 0.00		▲ 0.00	-1.00
▲ 0.00	▲ 0.00	▲ 0.00	0.00 ▼

FINAL VALUES

(b) iteration = 1

0.00 ▶	0.52 ▶	0.78 ▶	1.00
▲ 0.00		▲ 0.43	-1.00
▲ 0.00	▲ 0.00	▲ 0.00	0.00 ▼

FINAL VALUES

(c) iteration = 2

0.37 ▶	0.66 ▶	0.83 ▶	1.00
▲ 0.00		▲ 0.51	-1.00
▲ 0.00	0.00 ▶	▲ 0.31	◀ 0.00

FINAL VALUES

(d) iteration = 3

0.51 ▶	0.72 ▶	0.84 ▶	1.00
▲ 0.27		▲ 0.55	-1.00
▲ 0.00	0.22 ▶	▲ 0.37	◀ 0.13

FINAL VALUES

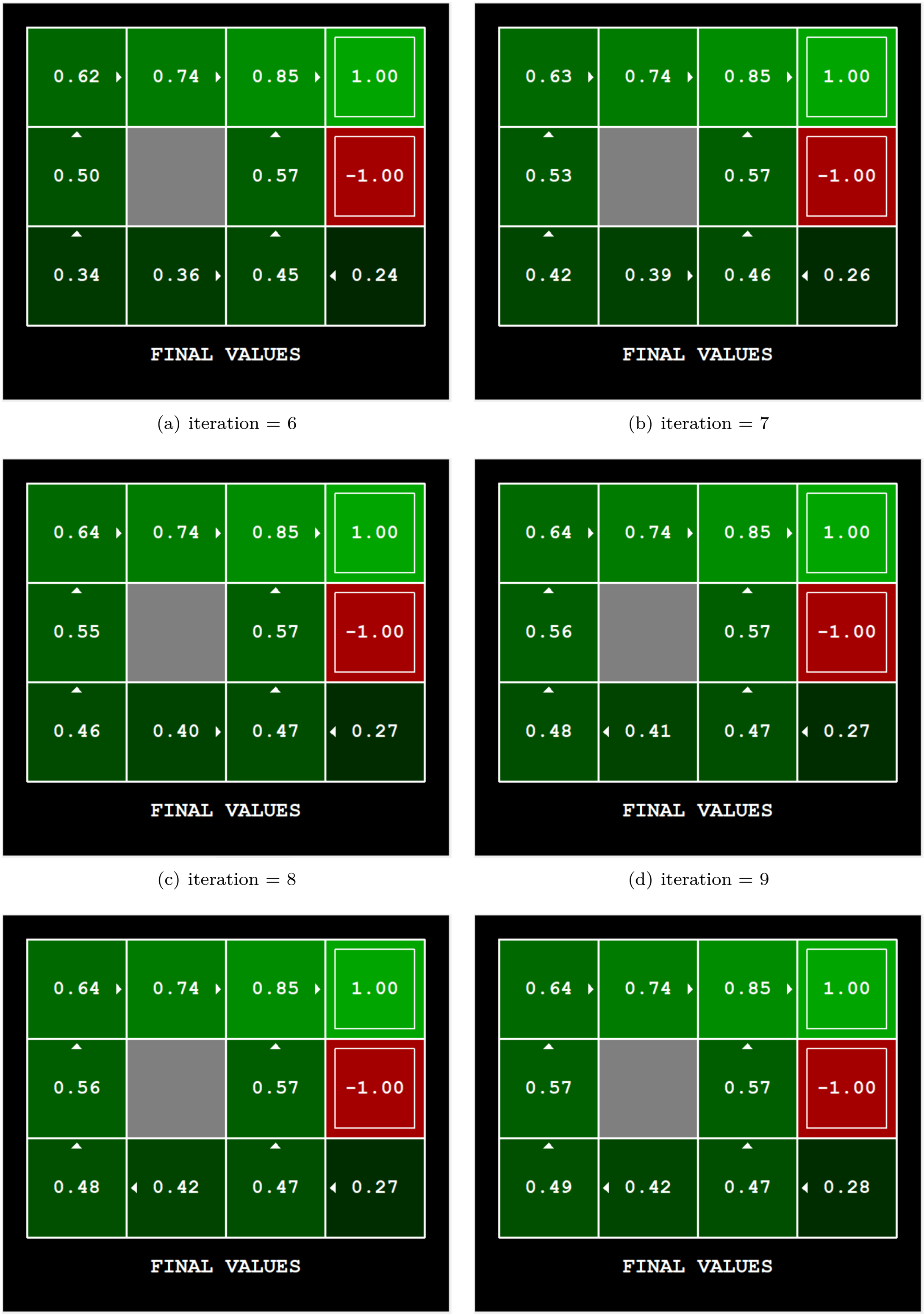
(e) iteration = 4

0.59 ▶	0.73 ▶	0.85 ▶	1.00
▲ 0.41		▲ 0.57	-1.00
▲ 0.21	0.31 ▶	▲ 0.43	◀ 0.19

FINAL VALUES

(f) iteration = 5

Figure 5: Policy Actions given by Value Iterations (Part. I)



0.62 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.50		▴ 0.57	-1.00
▴ 0.34	0.36 ▸	▴ 0.45	◂ 0.24

FINAL VALUES

(a) iteration = 6

0.63 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.53		▴ 0.57	-1.00
▴ 0.42	0.39 ▸	▴ 0.46	◂ 0.26

FINAL VALUES

(b) iteration = 7

0.64 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.55		▴ 0.57	-1.00
▴ 0.46	0.40 ▸	▴ 0.47	◂ 0.27

FINAL VALUES

(c) iteration = 8

0.64 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.56		▴ 0.57	-1.00
▴ 0.48	◂ 0.41	▴ 0.47	◂ 0.27

FINAL VALUES

(d) iteration = 9

0.64 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.56		▴ 0.57	-1.00
▴ 0.48	◂ 0.42	▴ 0.47	◂ 0.27

FINAL VALUES

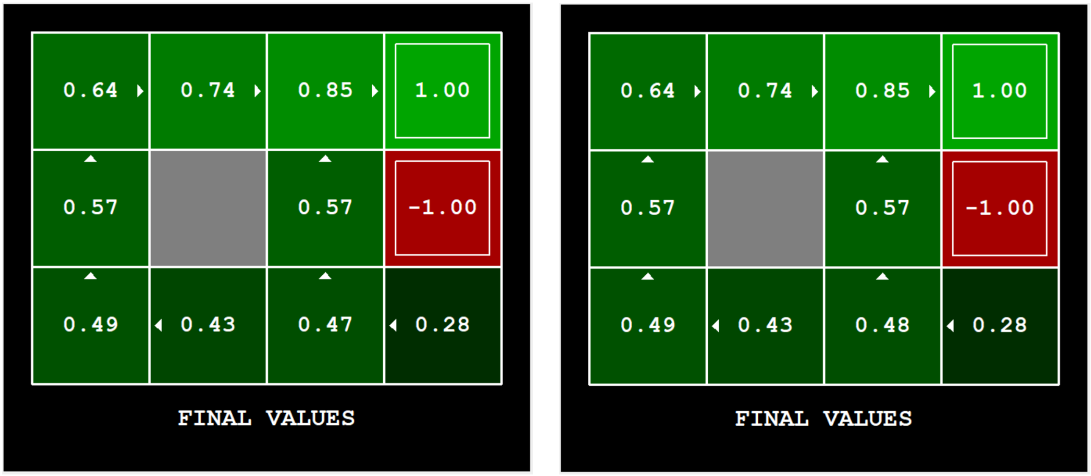
(e) iteration = 10

0.64 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.57		▴ 0.57	-1.00
▴ 0.49	◂ 0.42	▴ 0.47	◂ 0.28

FINAL VALUES

(f) iteration = 11

Figure 6: Policy Actions given by Value Iterations (Part. II)



0.64 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.57		▴ 0.57	-1.00
▴ 0.49	◂ 0.43	▴ 0.47	◂ 0.28

FINAL VALUES

(a) iteration = 12 & 13

0.64 ▸	0.74 ▸	0.85 ▸	1.00
▴ 0.57		▴ 0.57	-1.00
▴ 0.49	◂ 0.43	▴ 0.48	◂ 0.28

FINAL VALUES

(b) iteration = 14 & 15

Figure 7: Policy Actions given by Value Iterations (Part. III)

The policy is stable and optimal after the 2nd policy iteration. Meanwhile, even the policy given by the 1st iteration is quite close to the final optimal policy. On the other hand, the policy converges to the optimal policy in the 9th value iteration. In fact, the 10th to 15th value iteration is trying to approximate the value function, though the policy is already stable and optimal.

Thus, value iteration **converges slower than** policy iteration.