Example 1.13. Consider a hero H who is chased by a ghost G in a maze. Assume the maze is an infinite chain of nodes, with each node labeled by an integer in $(-\infty, \infty)$. H starts at 0 and G starts at -2. H always tries to move away from G, but only succeeds with a probability p, and with a probability $1-p$ gets stuck (i.e. with probability p H moves 1 step to the right, from node i to $i+1$, and gets 1 unit of reward; with probability $1-p$, H does not change his location and gets 0 unit of reward). G always chases after H and never gets stuck. If G catches H, H incurs -10 reward immediately, and the game is over. Both H and G moves simultaneously.

1. Write down the state space and action space.
2. Calculate the expected long term future reward (value function) with $p = 0.9$ and $\gamma = 0.95$.

[Hint: it is more convenient to describe the model by considering the relative position between H and G instead of their absolute positions.]

Question 2: how to make decisions? Is it a good idea to consider $\pi^*(x) = \arg\max_\pi V^\pi(x)$?

Dozens of algorithms have been developed to search for the optimal values and the optimal policies. We will introduce two most important ones: value iteration and policy iteration.

2 MDP: algorithm

We need to introduce state-action value function.

Definition 2.1 (State-action value function)**.** The state-action value function Q associated to a (randomized) policy π is defined for all $(x, a) \in \mathcal{S} \times \mathcal{A}$ as the expected return for taking action $a \in \mathcal{A}$ at state $x \in \mathcal{S}$ and then following the policy π:

$$\begin{aligned} Q^\pi(x,a) &= \mathbb{E}[r(x,a)] + \mathbb{E}_{a_t \sim \pi(x_t)} \left[\sum_{t=1}^{\infty} \gamma^t r(x_t, a_t) \middle| x_0 = x, a_0 = a \right] \\ &= \mathbb{E}\left[r(x,a) + \gamma V^\pi(x_1) | x_0 = x, a_0 = a \right] \end{aligned}$$

Remark 2.2.

- Recall the definition of the value function for a policy π:

$$V^\pi(x) = \mathbb{E}_{a \sim \pi(x)}[r(x,a)] + \gamma \mathbb{E}_{a \sim \pi}[V^\pi(x_1) | x_0 = x].$$

- Observe that $\mathbb{E}_{a \sim \pi(x)}[Q^\pi(x,a)] = V^\pi(x)$.
- If π is a deterministic policy, then $Q^\pi(x, \pi(x)) = V^\pi(x)$.

2.1 Value Iteration

The following theorem says that a randomized policy is optimal if and only if any action with positive probability under this policy maximize the state-action value function. It is coined as *Bellman's optimality*, which is the basis of the value iteration algorithm for solving MDPs.

Theorem 2.3 (Bellman's optimality)**.** *A randomized policy π is optimal if and only if for any $(x, a) \in \mathcal{S} \times \mathcal{A}$ with $\pi(x)(a) > 0$, the following holds*

$$a \in \arg\max_{a' \in \mathcal{A}(x)} Q^\pi(x, a'). \tag{2.1}$$

Proof. **Necessary condition.** If (2.1) is not true for some $(x, a) \in \mathcal{S} \times \mathcal{A}$ with $\pi(x)(a) > 0$, define

$$\pi'(y) = \begin{cases} \pi(x), & \text{if } y \neq x, \\ \delta_{a^*}, & \text{if } y = x, \end{cases}$$

where a^* is any element in $\arg\max_{a' \in \mathcal{A}(x)} Q^\pi(x, a')$. By the definition of a^* we have:

- $Q^\pi(x, a^*) > Q^\pi(x, a)$, and

- $Q^\pi(x, a^*) \geq Q^\pi(x, a')$ for any $a' \in \mathcal{A}(x)$.

Since $\pi(x)(a) > 0$, it holds that $\mathbb{E}_{b\sim\delta_{a^*}}[Q^\pi(x,b)] = Q^\pi(x,a^*) > \mathbb{E}_{b\sim\pi(x)}[Q^\pi(x,b)]$. Moreover, when $y \neq x$, we have

$$\mathbb{E}_{b\sim\pi'(y)}[Q^\pi(y,b)] = \mathbb{E}_{b\sim\pi(x)}\mathbb{E}[Q^\pi(y,b)].$$

By Theorem 2.5 (see the detail later), $V^{\pi'}(x) > V^\pi(x)$ for at least one $x \in \mathcal{S}$ and π is not optimal.

Sufficient condition. Since $\pi(x)(a) > 0$ implies $Q^\pi(x,a) \geq Q^\pi(x,a')$ for any $a' \in \mathcal{A}(x)$, we know that $\pi(x)$ assigns positive probabilities only to actions that maximize $Q^\pi(x,a')$. Noting that $V^\pi(x) = \sum_{a\in\mathcal{A}(x)} Q^\pi(x,a)\pi(x)(a)$, we have

$$V^\pi(x) = \max_{a'\in\mathcal{A}(x)} Q^\pi(x,a').$$

Then $V^\pi(x) \geq V^{\pi'}(x)$ for any π' and any $x \in \mathcal{S}$. □

Remark 2.4.

- Bellman optimality tells π^* is optimal, if and only it solves

$$V^*(x) = \max_{a\in\mathcal{A}(x)} \left\{ \mathbb{E}[r(x,a)] + \gamma \sum_{x'\in\mathcal{S}} P(x'|x,a)V^*(x') \right\}$$

  by Definition 2.1 and Theorem 2.3. Here, $V^* := V^{\pi^*}$. It is also called Bellman equation.

- To find the optimal policy of a given state, one only needs to solve the Bellman equation.

- Generally, it is difficult to solve Bellman equations explicitly because "max" is not a linear operator. One relies one iteration: let $V_i(x)$ be the value function for state x at the ith iteration. The iteration step, called a Bellman update, reads

$$V_{i+1}(x) \leftarrow \max_{a\in\mathcal{A}(x)} \left\{ r(x,a) + \gamma \sum_{x'} P(x'|x,a)V_i(x') \right\} := \mathcal{B}(V_i(x)),$$

  where the update is assumed to be applied simultaneously to all states at each iteration. If we apply Bellman update infinitely many times, we are guaranteed to reach an equilibrium, which must be the (unique!) solution to the Bellman equations, and the corresponding value function is the optimal one:

$$\sup_{x\in\mathcal{S}} |\mathcal{B}(V(x)) - \mathcal{B}(V'(x))| \leq \gamma \max_{x\in\mathcal{S}} |V(x) - V'(x)|. \tag{2.2}$$

  This algorithm is called value iteration.

Algorithm 1 Value iteration

Input: *mdp*, an MDP with states $\mathcal{S}$, actions $\mathcal{A}$, transition model $P(x'|x,a)$, rewards r, discount γ
Local variables: V, V', vectors of value functions for states in $\mathcal{S}$, initially zero δ, the maximum change in the value function of any state in an iteration
  repeat
    $V \leftarrow V'$, $\delta \leftarrow 0$
    for each x in $\mathcal{S}$ **do**
      $V'(x) \leftarrow \max_{a\in\mathcal{A}(x)} \left\{ r(x,a) + \gamma \sum_{x'} P(x'|x,a)V(x') \right\}$
      if $|V'(x) - V(x)| > \delta$, **then** $\delta \leftarrow |V'(x) - V(x)|$
  until $\delta < \epsilon(1-\gamma)/\gamma$
  return V

- (Convergence rate) Let the solution to the Bellman equation (equivalently, the limit of the iteration) be V^*. Then $\mathcal{B}(V^*) = V^*$. By (2.2),

$$\|V_{i+1} - V^*\|_\infty = \|\mathcal{B}(V_i) - V^*\|_\infty \leq \gamma \|V_i - V^*\|_\infty \leq \cdots \leq \gamma^{i+1} \|V_0 - V^*\|_\infty.$$

  The difference $\|V_i - V^*\|_\infty$ is defined as the error of the estimate. The contraction property tells that the value iteration converges exponentially fast.

  Moreover, it may be overly conservative to use the error bound $\|V_i - V^*\|_\infty$ to determine the stopping time. Alternatively, we can use the difference between the value functions in two consecutive iterations to determine the stopping time. This is because the following result holds

$$\|V_{i+1} - V_i\|_\infty \leq \frac{\epsilon(1-\gamma)}{\gamma} \quad \Rightarrow \quad \|V_{i+1} - V^*\|_\infty \leq \epsilon.$$

  This is the termination condition used in Algorithm 1.

2.2 Policy Iteration

The policy iteration algorithm[4] alternates the following two steps, starting with some initial policy π_0:

- Policy evaluation: given a policy π_i, calculate V^{π_i}, the value function of each state if π_i is executed.
- Policy improvement: calculate a new maximum expected utility (MEU) policy π_{i+1}, i.e. for each $x \in X$

$$\pi_{i+1}(x) = \arg\max_{a \in A(x)} \left\{ r(x,a) + \sum_{x'} P(x'|x,a) V^{\pi_i}(x') \right\}.$$

Question 3: why does policy iteration work? The second step improve the value function, i.e. $V^{\pi_{i+1}} \geq V^{\pi_i}$. This is a corollary of the following *policy improvement theorem*.

Proof. Note that $r(x,a) + \sum_{x'} P(x'|x,a)V^{\pi}(x') = r(x,a) + \mathbb{E}[V^{\pi}(x_1)|x_0 = x, a_0 = a] := Q_{\pi}(x,a)$. By Lemma 1.9, $V^{\pi}(x) = Q_{\pi}(x, \pi(x))$.

Since $\pi_{i+1}(x) = \arg\max_a Q_{\pi_i}(x,a)$, we have $Q_{\pi_i}(x, \pi_{i+1}(x)) \geq Q_{\pi_i}(x,a)$ for all $a \in A(x)$. In particular, $Q_{\pi_i}(x, \pi_{i+1}(x)) \geq Q_{\pi_i}(x, \pi_i(x)) = V^{\pi_i}(x)$. Moreover,

$$\begin{aligned}
& Q_{\pi_i}(x, \pi_{i+1}(x)) \\
&= r(x, \pi_{i+1}(x)) + \gamma \mathbb{E}[V^{\pi_i}(x_1)|x_0 = x, a_0 = \pi_{i+1}(x)] \\
&= r(x, \pi_{i+1}(x)) + \gamma \mathbb{E}[Q_{\pi_i}(x_1, \pi_i(x_1))|x_0 = x, a_0 = \pi_{i+1}(x)] \\
&\leq r(x, \pi_{i+1}(x)) + \gamma \mathbb{E}[Q_{\pi_i}(x_1, \pi_{i+1}(x_1))|x_0 = x, a_0 = \pi_{i+1}(x)] \\
&= r(x, \pi_{i+1}(x)) + \mathbb{E}\left[\mathbb{E}\left\{\gamma r(x_1, \pi_{i+1}(x_1)) + \gamma^2 V^{\pi_i}(x_2)|x_0 = x, x_1, a_0 = \pi_{i+1}(x), a_1 = \pi_{i+1}(x_1)\right\}\right] \\
&\leq \cdots \\
&= \mathbb{E}\left[\sum_{k=0}^{N} \gamma^k r(x_k, \pi_{i+1}(x_k)) + \gamma^{N+1} V^{\pi_i}(x_{N+1})|x_0 = x\right].
\end{aligned}$$

Let $N \to \infty$ and use Exercise 1. □

Theorem 2.5 (Policy improvement theorem)**.** *For any two randomized policies π and π', we have*

$$\Big(\forall x \in X,\ \mathbb{E}_{a \sim \pi'(x)}[Q^{\pi}(x,a)] \geq \mathbb{E}_{a \sim \pi(x)}[Q^{\pi}(x,a)]\Big) \Rightarrow \Big(\forall x \in X,\ V^{\pi'}(x) \geq V^{\pi}(x)\Big).$$

Furthermore, a strict inequality for at least one state x in the l.h.s. implies a stric inequality for at least one x in the r.h.s..

[4] Note that the algorithm depends on the initial policy π_0; refer to Exercise 8.

Proof. Let π' and π satisfy the l.h.s.. For any $x \in X$,

$$\begin{aligned}
V^{\pi}(x) &= \mathbb{E}_{a\sim\pi(x)}[Q^{\pi}(x,a)] \\
&\leq \mathbb{E}_{a\sim\pi'(x)}[Q^{\pi}(x,a)] \\
&= \int_A \mathbb{E}[r(x,a)+\gamma V^{\pi}(x_1)|x_0=x,a_0=a]\pi'(x)(da) \\
&= \mathbb{E}_{a\sim\pi'(x)}[r(x,a)+\gamma V^{\pi}(x_1)] \\
&= \mathbb{E}_{a\sim\pi'(x)}\left[r(x,a)+\gamma\mathbb{E}_{a_1\sim\pi(x_1)}[Q^{\pi}(x_1,a_1)]|x_0=x\right] \qquad \text{(by Remark 1.9)} \\
&\leq \mathbb{E}_{a\sim\pi'(x)}\left[r(x,a)+\gamma\mathbb{E}_{a_1\sim\pi'(x_1)}[Q^{\pi}(x_1,a_1)]|x_0=x\right] \\
&= \mathbb{E}_{a\sim\pi'(x),a_1\sim\pi'(x_1)}\left[r(x,a)+\gamma r(x_1,a_1)+\gamma^2 V^{\pi}(x_2)|x_0=x\right].
\end{aligned}$$

Repeating the above argument for T times leads to

$$V^{\pi}(x) \leq \mathbb{E}_{a_t\sim\pi'(x_t)}\left[\sum_{t=0}^{T}\gamma^t\mathbb{E}[r(x_t,a_t)]+\gamma^{T+1}V^{\pi}(x_{T+1})\middle| x_0=x\right].$$

Letting $T\to\infty$ and using Exercise 1, we get

$$V^{\pi}(x) \leq \mathbb{E}_{a_t\sim\pi'(x_t)}\left[\sum_{t=0}^{\infty}\gamma^t\mathbb{E}[r(x_t,a_t)]\middle| x_0=x\right] = V^{\pi'}(x).$$

□

Example 2.6. Consider an undiscounted MDP with three states, $(1, 2, 3)$ and the corresponding rewards $(-1, -2, 0)$. State 3 is a terminal state. In states 1 and 2 there are two possible randomized actions: a and b. The transition model is as follows:

- In state 1, action a moves the agent to state 2 with probability 0.8 and makes the agent stay put with probability 0.2.
- In state 2, action a moves the agent to state 1 with probability 0.8 and makes the agent stay put with probability 0.2.
- In both state 1 and state 2, action b moves the agent to state 3 with probability 0.1 and makes the agent stay put with probability 0.9.

1. Apply policy iteration, showing each step in full, to determine the optimal policy and the values of states 1 and 2. Assume that the initial policy has action b in both states.
2. What happens to policy iteration if the initial policy has action a in both states?

3 Acknowledgement

The following references are referred to and acknowledged:

References

[1] Andreas Krause, *Lecture notes*.

[2] Mehryar Mohri, Afshin Rostamizadeh, Ameet Talwalkar, *Foundations of Machine Learning*.

[3] Stuart J. Russell and Peter Norvig, *Artificial Intellegence: A Modern Approach*.

Final Exam

1. Time: Dec 14, 12:30-14:30
2. Venue: SPH

   https://www.polyu.edu.hk/ar/docdrive/polyu-students/exam-timetable/Location-Map-Seating-Plan.pdf
3. Focus on the second half of the semester.
4. One piece of double-sided A4 paper with handwritten notes. The notes must be written directly on the paper, not on an iPad or printed.

   It will be checked seriously during exam.
5. Calculators.