# Chapter 9

# Policy Gradient Methods

The idea of function approximation can be applied to represent not only state/action values but also policies. Up to now in this book, policies have been represented by tables: the action probabilities of all states are stored in a table $\pi(a|s)$, each entry of which is indexed by a state and an action. In this chapter, we show that policies can be represented by parameterized functions denoted as $\pi(a|s,\theta)$, where $\theta \in \mathbb{R}^m$ is a parameter vector. The function representation is also sometimes written as $\pi(a,s,\theta)$, $\pi_\theta(a|s)$, or $\pi_\theta(a,s)$.

When policies are represented as a function, optimal policies can be found by optimizing certain scalar metrics. Such kind of method is called *policy gradient*. Policy gradient is a big step forward in this book because it is *policy-based*. By contrast, all the previous chapters in this book consider *value-based* methods that must estimate state/action values to obtain optimal policies.

The advantages of the policy gradient methods are numerous. For example, when the state or action space is large, the tabular representation will be of low efficiency in terms of storage and policy searching. As a comparison, the dimension of the parameter in a function representation may be significantly less than the number of states. It can also handle continuous state and action spaces.

## 9.1 Basic idea of policy gradient

To use a parameterized function to represent a policy, we need to first answer some basic questions.

First, how to define optimal policies? When represented as a table, a policy $\pi$ is defined as optimal if it can maximize *every state value*. When represented by a function, a policy $\pi$ is fully determined by $\theta$ together with the function structure. The policy is defined as optimal if it can maximize certain *scalar metrics*. This is one important difference between tabular and function representations.

Second, how to update policies? When represented by a table, a policy $\pi$ can be updated by directly changing the entries in the table. However, when represented by a parameterized function, a policy $\pi$ cannot be updated in this way anymore. Instead, it can only be improved by updating *the parameter* $\theta$. This is an important difference in terms of ways of updating policies.

The basic idea of the policy gradient method is summarized below. Suppose $J(\theta)$ is a scalar metric to define optimal policies. Optimal policies can be obtained by optimizing the metric based on gradient-based algorithms:

$$\theta_{t+1} = \theta_t + \alpha \nabla_\theta J(\theta_t),$$

where $\nabla_\theta J$ is the gradient of $J$ with respect to $\theta$, $t$ is the time step, and $\alpha$ is the optimization rate. A review of gradient-based algorithms is given in Appendix B.

Although the idea of policy gradient is straightforward, the complication emerges when we try to answer the following questions.

– What appropriate metrics should be used? (Section 9.2).

– How to calculate the gradients of the metrics? (Section 9.3)

– How to use experience samples to calculate the gradients? (Section 9.4)

These questions will be answered in detail in the rest of this chapter.

## 9.2 Metrics to define optimal policies

If a policy is represented by a function, what metrics should we use to define optimality? We next present some popular metrics.

1) The first metric is the *average state value* or simply called *average value*. In particular, let

$$\begin{aligned} v_\pi &= [\dots, v_\pi(s), \dots]^T \in \mathbb{R}^{|\mathcal{S}|} \\ d_\pi &= [\dots, d_\pi(s), \dots]^T \in \mathbb{R}^{|\mathcal{S}|} \end{aligned}$$

be the vector of state values and a vector of weights, respectively. Here, $d_\pi(s) \geq 0$ is the weight for state $s$ and satisfies $\sum_s d_\pi(s) = 1$. The metric of average value is defined as

$$\begin{aligned} \bar{v}_\pi &\doteq d_\pi^T v_\pi \\ &= \sum_s d_\pi(s) v_\pi(s) \\ &= \mathbb{E}[v_\pi(S)], \end{aligned}$$

where $S \sim d_\pi$. As its name suggests, $\bar{v}_\pi$ is simply a weighted average of the state values.

How to select the weights $d_\pi(s)$? One trivial way is to treat all the states equally important and hence select $d_\pi(s) = 1/|\mathcal{S}|$ for every $s$. Another way, which is often used, is to select $d_\pi(s)$ as the *stationary distribution* satisfying

$$d_\pi^T P_\pi = d_\pi^T,$$

where $P_\pi$ is the state transition probability matrix. The interpretation of selecting $d_\pi$ is as follows. The stationary distribution reflects the long-term behavior of the Markov decision process under the given policy. If one state is frequently visited in the long run, it is more important and deserves more weight; if a state is hardly visited, then we give it less weight. Details of stationary distribution can be found in Section 8.2 in the last chapter.

2) The second metric is the *average one-step reward* or simply called *average reward* [2,34,35]. In particular, let

$$r_\pi = [\dots, r_\pi(s), \dots]^T \in \mathbb{R}^{|\mathcal{S}|}$$

be the vector of one-step immediate rewards. Here,

$$r_\pi(s) = \sum_a \pi(a|s) r(s,a)$$

is the average of the one-step immediate reward that can be obtained starting from state $s$, and $r(s,a) = \mathbb{E}[R|s,a] = \sum_r rp(r|s,a)$ is the average of the one-step immediate reward that can be obtained after taking action $a$ at state $s$. Then, the metric is defined as

$$\begin{aligned}\bar{r}_\pi &\doteq d_\pi^T r_\pi \\ &= \sum_s d_\pi(s) r_\pi(s) \\ &= \mathbb{E}[r_\pi(S)],\end{aligned} \tag{9.1}$$

where $S \sim d_\pi$. As its name suggests, $\bar{r}_\pi$ is simply a weighted average of the one-step immediate rewards. Here, the weight $d_\pi$ is the stationary distribution.

3) The third metric is the state value of a specific stating state $v_\pi(s_0)$ [3,34]. For some tasks, we can only start from a specific state $s_0$. In this case, we only care about the long-term return starting from $s_0$. The third metric can also be viewed as a weighted average of the state values. To see that,

$$v_\pi(s_0) = \sum_{s \in \mathcal{S}} d_0(s) v_\pi(s),$$

where

$$d_0(s = s_0) = 1, \quad d_0(s \neq s_0) = 0.$$

While we have introduced the definitions of the metrics above, we next give some important remarks about these metrics.

- All these metrics are functions of $\pi$. Since $\pi$ is parameterized by $\theta$, these metrics are functions of $\theta$. In other words, different values of $\theta$ can generate different metric values. Therefore, we can search for the optimal values of $\theta$ to maximize these metrics. This is the basic idea of policy gradient methods.
- Intuitively, $\bar{r}_\pi$ is more short-sighted because it merely considers the immediate rewards, whereas $\bar{v}_\pi$ considers the total reward overall steps. However, the two metrics are equivalent to each other. In the discounted case where $\gamma < 1$, it will be shown in Lemma 9.1 that

$$\bar{r}_\pi = (1-\gamma)\bar{v}_\pi.$$

- The average reward $\bar{r}_\pi$ has another important definition that is equivalent to 9.1. In particular, suppose an agent follows a given policy and generate a trajectory with the rewards as $(R_{t+1}, R_{t+2}, \dots)$. Then, the average single-step reward along this trajectory is defined as

$$\bar{r}_\pi \doteq \lim_{n\to\infty} \frac{1}{n}\mathbb{E}\left[R_{t+1} + R_{t+2} + \cdots + R_{t+n} \middle| S_t = s_0\right] = \lim_{n\to\infty} \frac{1}{n}\mathbb{E}\left[\sum_{k=1}^{n} R_{t+k} \middle| S_t = s_0\right], \tag{9.2}$$

where $s_0$ is the starting state of the trajectory. An important property is that

$$\lim_{n\to\infty} \frac{1}{n}\mathbb{E}\left[\sum_{k=1}^{n} R_{t+k} \middle| S_t = s_0\right] = \lim_{n\to\infty} \frac{1}{n}\mathbb{E}\left[\sum_{k=1}^{n} R_{t+k}\right] = \sum_s d_\pi(s) r_\pi(s). \tag{9.3}$$

The above equation implies two facts. The first is that the two definitions in (9.1) and (9.2) are equivalent. The second is that the definition in (9.2) is independent to the starting state $s_0$. The proof of the above equation is given in the following shaded box.

**Equivalence between the two definitions of $\bar{r}_\pi$ in (9.1) and (9.2)**

$$\begin{aligned}\bar{r}_\pi &= \lim_{n\to\infty}\frac{1}{n}\mathbb{E}\left[\sum_{k=1}^{n} R_{t+k}|S_t=s_0\right]\\ &= \lim_{n\to\infty}\frac{1}{n}\sum_{k=1}^{n}\mathbb{E}\left[R_{t+k}|S_t=s_0\right]\\ &= \lim_{k\to\infty}\mathbb{E}\left[R_{t+k}|S_t=s_0\right].\end{aligned}$$

The last equability in the above equation is due to the property of the Cesaro mean. In particular, if $\{a_k\}_{k=1}^{\infty}$ is a convergent sequence such that $\lim_{k\to\infty} a_k$ exists, then $\{1/n\sum_{k=1}^{n} a_k\}_{n=1}^{\infty}$ is also a convergent sequence such that $\lim_{n\to\infty} 1/n\sum_{k=1}^{n} a_k = \lim_{k\to\infty} a_k$.

We next examine $\mathbb{E}\left[R_{t+k}|S_t=s_0\right]$. By the law of total expectation, we have

$$\begin{aligned}\mathbb{E}\left[R_{t+k}|S_t=s_0\right] &= \sum_s \mathbb{E}\left[R_{t+k}|S_{t+k-1}=s, S_t=s_0\right]p^{(k-1)}(s|s_0)\\ &= \sum_s \mathbb{E}\left[R_{t+k}|S_{t+k-1}=s\right]p^{(k-1)}(s|s_0)\\ &= \sum_s r_\pi(s)p^{(k-1)}(s|s_0),\end{aligned}$$

where $p^{(k-1)}(s|s_0)$ denotes the probability for transiting from $s_0$ to $s$ using exactly $k-1$ steps. The second equality in the above equation is due to the Markov memoryless property: that is, the reward obtained the next time depends only on the current state instead of the previous states. The third equality is because $\mathbb{E}\left[R_{t+k}|S_{t+k-1}=s\right]$ is the expected value of the immediate rewards obtained starting from $s$.

Note that $\lim_{k\to\infty} p^{(k-1)}(s|s_0) = d_\pi(s)$ due to the property of stationary distribution. As a result, the initial state $s_0$ does not matter. Then, we have

$$\lim_{k\to\infty}\mathbb{E}\left[R_{t+k}|S_t=s_0\right] = \lim_{k\to\infty}\sum_s r_\pi(s)p^{(k-1)}(s|s_0) = \sum_s r_\pi(s)d_\pi(s).$$

The proof is complete.

- One complication of the policy gradient method is that the metrics can be defined in either the discounted case where $\gamma\in[0,1)$ or the undiscounted case where $\gamma=1$. We only consider the discounted case so far in this book. The undiscounted case is new and will be analyzed in detail in Section 9.3.2.

  The definitions of state values and hence $\bar{v}_\pi$ and $v_\pi(s_0)$ are different when $\gamma=1$ and $\gamma<1$. By contrast, the definition of $\bar{r}_\pi$ does not rely on $\gamma$. Nevertheless, calculating the gradient of $\bar{r}_\pi$ still needs to distinguish the discounted and undiscounted cases. We will see that the three metrics have different properties in the two cases. This might be one of the most confusing problems for beginners studying policy gradient and deserves special attention.

## 9.3 Gradients of the metrics

Given the metrics introduced in the last section, we can use the gradient-based method to maximize them. This section calculates the gradients of the metrics. The calculation is one of the most complicated parts of policy gradient methods. To introduce the results clearly, we first present the most important result and then show the proof and related results later.

**Theorem 9.1** (Policy gradient theorem). *The gradient of the average reward metric is*

$$\nabla_\theta \bar{r}_\pi(\theta) \simeq \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a), \tag{9.4}$$

*where* $\nabla_\theta \pi$ *is the gradient of* $\pi$ *with respect to* $\theta$*. Here,* $\simeq$ *refers to either strict equality or approximated equality. In particular, it is a strict equation in the undiscounted case where* $\gamma = 1$ *and an approximated equation in the discounted case where* $0 < \gamma < 1$*. The approximation is more accurate in the discounted case when* $\gamma$ *is closer to 1. Moreover,* (9.4) *has a more compact and useful form expressed in terms of expectation:*

$$\nabla_\theta \bar{r}_\pi(\theta) \simeq \mathbb{E}\Big[\nabla_\theta \ln \pi(A|S,\theta) q_\pi(S,A)\Big], \tag{9.5}$$

*where* ln *is the natural logarithm and* $S \sim d_\pi, A \sim \pi(S)$*.*

Some important remarks about Theorem 9.1 are given below.

1) This is the well-known *policy gradient theorem* [34–36], a fundamental result for policy gradient methods. This theorem incorporate both the discounted and undiscounted cases, and is a combination of Theorem 9.3 and Theorem 9.5 shown later in the following subsections. Since the two subsections are mathematically intensive, readers are advised to read selectively according to their interests.
2) The gradients of the other two metrics $v_\pi(s_0)$ and $\bar{v}_\pi$ are not given in this theorem. Their gradients in the discounted case are given in Theorem 9.2 and Theorem 9.3 as shown later. We will see that their gradients have similar expressions as (9.5).
3) The expression in (9.5) is more favorable compared to (9.4) because it is expressed in terms of an expectation that can be approximated by experience samples. Why can (9.4) be expressed as (9.5)? The proof is trivial and given below. By the definition of

expectation, (9.4) can be rewritten as

$$\begin{aligned}\nabla_\theta \bar{r}_\pi &\simeq \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a) \\ &= \mathbb{E}\left[\sum_a \nabla_\theta \pi(a|S,\theta) q_\pi(S,a)\right], \end{aligned} \tag{9.6}$$

where the state $S \sim d_\pi$. Furthermore, consider the function $\ln \pi$ where ln is the natural logarithm. Its gradient can be easily obtained as

$$\nabla_\theta \ln \pi(a|s,\theta) = \frac{\nabla_\theta \pi(a|s,\theta)}{\pi(a|s,\theta)}$$

and hence

$$\nabla_\theta \pi(a|s,\theta) = \pi(a|s,\theta) \nabla_\theta \ln \pi(a|s,\theta). \tag{9.7}$$

Substituting (9.7) into (9.6) gives

$$\begin{aligned}\nabla_\theta \bar{r}_\pi &\simeq \mathbb{E}\left[\sum_a \pi(a|S,\theta) \nabla_\theta \ln \pi(a|S,\theta) q_\pi(S,a)\right] \\ &= \mathbb{E}\Big[\nabla_\theta \ln \pi(A|S,\theta) q_\pi(S,A)\Big],\end{aligned}$$

where $S \sim d_\pi$ and $A \sim \pi(S)$.

4) It is notable that $\pi(a|s,\theta)$ must be *positive* for all $s, a$ to ensure that $\ln \pi(a|s,\theta)$ is valid. This can be archived by using *softmax functions*:

$$\pi(a|s,\theta) = \frac{e^{h(s,a,\theta)}}{\sum_{a' \in \mathcal{A}} e^{h(s,a',\theta)}},$$

where $h(s,a,\theta)$ is another function indicating the value or preference of selecting $a$ at $s$. The overall softmax function approximation can be realized by a neural network whose input is $s$ and parameter is $\theta$. The network has $|\mathcal{A}|$ outputs, each of which corresponds to $\pi(a|s,\theta)$ for an action $a$. The activation function of the output layer should be softmax.

Since $\pi(a|s,\theta) > 0$ for all $a$, the parameterized policy is *stochastic* and hence *exploratory.* The policy does not directly tell which action to take. Instead, the action should be sampled according to the probability distribution of the policy. We will see in the next chapter that there also exist *deterministic* policy gradient methods.

### 9.3.1 Gradients in the discounted case

In this section, we derive the gradients of the metrics in the discounted case where $\gamma \in [0,1)$.

We are already familiar with the discounted case since all the previous chapters consider this case. The state value and action value in the discounted case are defined as

$$\begin{aligned} v_\pi(s) &\doteq \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots | S_t = s], \\ q_\pi(s,a) &\doteq \mathbb{E}[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots | S_t = s, A_t = a], \end{aligned}$$

where $\gamma \in (0,1)$ is the discount rate. We know $v_\pi(s) = \sum_a \pi(a|s,\theta) q_\pi(s,a)$ and the state value satisfies the Bellman equation.

Before deriving the gradients of $\bar{r}_\pi$ and $\bar{v}_\pi$, we introduce some useful preliminary results. First, the next lemma shows that $\bar{v}_\pi(\theta)$ and $\bar{r}_\pi(\theta)$ are equivalent metrics.

**Lemma 9.1** (Equivalence between $\bar{v}_\pi(\theta)$ and $\bar{r}_\pi(\theta)$)**.** *In the discounted case where* $\gamma \in [0,1)$, *it holds that*

$$\bar{r}_\pi = (1-\gamma)\bar{v}_\pi. \tag{9.8}$$

*Proof.* Note that $\bar{v}_\pi(\theta) = d_\pi^T v_\pi$ and $\bar{r}_\pi(\theta) = d_\pi^T r_\pi$, where $v_\pi$ and $r_\pi$ satisfy the Bellman equation $v_\pi = r_\pi + \gamma P_\pi v_\pi$. Multiplying $d_\pi^T$ on both sides of the Bellman equation gives

$$\bar{v}_\pi = \bar{r}_\pi + \gamma d_\pi^T P_\pi v_\pi = \bar{r}_\pi + \gamma d_\pi^T v_\pi = \bar{r}_\pi + \gamma \bar{v}_\pi,$$

which implies (9.8). □

Second, the following lemma gives the gradient of $v_\pi(s)$ for any $s$.

**Lemma 9.2** (Gradient of $v_\pi(s)$)**.** *In the discounted case, it holds for any* $s \in \mathcal{S}$ *that*

$$\nabla_\theta v_\pi(s) = \sum_{s'} \Pr{}_\pi(s'|s) \sum_a \nabla_\theta \pi(a|s',\theta) q_\pi(s',a), \tag{9.9}$$

*where*

$$\Pr{}_\pi(s'|s) \doteq \sum_{k=0}^{\infty} \gamma^k \Pr(s \to s', k, \pi) = \left[(I_n - \gamma P_\pi)^{-1}\right]_{ss'}$$

*is the discounted total probability transiting from* $s$ *to* $s'$ *under policy* $\pi$*. Here,* $\Pr(s \to s', k, \pi)$ *denotes the probability for transiting from* $s$ *to* $s'$ *using exactly* $k$ *steps under* $\pi$*.*

**Proof of Lemma 9.2**

First, for any $s \in \mathcal{S}$, it holds that

$$\begin{aligned}\nabla_\theta v_\pi(s) &= \nabla_\theta \left[\sum_a \pi(a|s,\theta) q_\pi(s,a)\right] \\ &= \sum_a \left[\nabla_\theta \pi(a|s,\theta) q_\pi(s,a) + \pi(a|s,\theta) \nabla_\theta q_\pi(s,a)\right], \qquad (9.10)\end{aligned}$$

where $q_\pi(s,a)$ is the action value given by

$$q_\pi(s,a) = r(s,a) + \gamma \sum_{s'} p(s'|s,a) v_\pi(s').$$

Here, $r(s,a) = \sum_r r p(r|s,a)$ is independent of $\theta$. Therefore,

$$\nabla_\theta q_\pi = 0 + \gamma \sum_{s'} p(s'|s,a) \nabla_\theta v_\pi(s'),$$

substituting which into (9.10) gives

$$\nabla_\theta v_\pi(s) = \sum_a \left[\nabla_\theta \pi(a|s,\theta) q_\pi(s,a) + \pi(a|s,\theta)\gamma \sum_{s'} p(s'|s,a) \nabla_\theta v_\pi(s')\right]. \qquad (9.11)$$

It is notable that $\nabla_\theta v_\pi$ appears on both sides of the above equation. To calculate $\nabla_\theta v_\pi$, one way is to use the *unrolling technique* [34]. Here, we use the *matrix-vector form*, which we believe is more straightforward to understand. In particular, let

$$u(s) \doteq \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

Equation (9.11) can be written in a matrix-vector form as

$$\underbrace{\begin{bmatrix} \vdots \\ \nabla_\theta v_\pi(s) \\ \vdots \end{bmatrix}}_{\nabla_\theta v_\pi \in \mathbb{R}^{n|\mathcal{S}|}} = \underbrace{\begin{bmatrix} \vdots \\ u(s) \\ \vdots \end{bmatrix}}_{u \in \mathbb{R}^{n|\mathcal{S}|}} + \gamma (P_\pi \otimes I_m) \underbrace{\begin{bmatrix} \vdots \\ \nabla_\theta v_\pi(s') \\ \vdots \end{bmatrix}}_{\nabla_\theta v_\pi \in \mathbb{R}^{n|\mathcal{S}|}},$$

which can be written in short as

$$\nabla_\theta v_\pi = u + \gamma (P_\pi \otimes I_m) \nabla_\theta v_\pi.$$

Here, $m$ is the dimension of the parameter vector $\theta$ and $\otimes$ is the Kronecker product. The reason that the Kronecker product emerges in the equation is that $\nabla_\theta v_\pi(s)$ is already a vector. The above equation is a linear equation of $\nabla_\theta v_\pi$, which can be solved as

$$\begin{aligned}\nabla_\theta v_\pi &= (I_{nm} - \gamma P_\pi \otimes I_m)^{-1} u \\ &= (I_n \otimes I_m - \gamma P_\pi \otimes I_m)^{-1} u \\ &= \left[(I_n - \gamma P_\pi)^{-1} \otimes I_m\right] u \end{aligned} \tag{9.12}$$

The elementwise form of the solution is

$$\begin{aligned}\nabla_\theta v_\pi(s) &= \sum_{s'} \left[(I_n - \gamma P_\pi)^{-1}\right]_{ss'} u(s') \\ &= \sum_{s'} \left[(I_n - \gamma P_\pi)^{-1}\right]_{ss'} \sum_a \nabla_\theta \pi(a|s', \theta) q_\pi(s', a) \end{aligned} \tag{9.13}$$

where $[\cdot]_{ss'}$ is the entry on the $s$th row and $s'$th column. The quantity $[(I_n - \gamma P_\pi)^{-1}]_{ss'}$ has a clear probability interpretation. In particular, since $(I_n - \gamma P_\pi)^{-1} = I + \gamma P_\pi + \gamma^2 P_\pi^2 + \cdots$, we have

$$\left[(I_n - \gamma P_\pi)^{-1}\right]_{ss'} = [I]_{ss'} + \gamma [P_\pi]_{ss'} + \gamma^2 [P_\pi^2]_{ss'} + \cdots = \sum_{k=0}^{\infty} \gamma^k [P_\pi^k]_{ss'}.$$

Note that $[P_\pi^k]_{ss'}$ is the probability transiting from $s$ to $s'$ using exactly $k$ steps. Therefore, $[(I_n - \gamma P_\pi)^{-1}]_{ss'}$ is the (discounted) total probability transiting from $s$ to $s'$ using any steps. By denoting $[(I_n - \gamma P_\pi)^{-1}]_{ss'} \doteq \Pr_\pi(s'|s)$, equation (9.13) becomes (9.10).

With the results in Lemma 9.2, we are ready to derive the gradient of $v_\pi(s_0)$ with a specific $s_0$.

**Theorem 9.2** (Gradient of $v_\pi(s_0)$ in the discounted case)**.** *In the discounted case where $\gamma \in [0, 1)$, the gradient of $v_\pi(s_0)$ is*

$$\nabla_\theta v_\pi(s_0) = \mathbb{E}\big[\nabla_\theta \ln \pi(A|S, \theta) q_\pi(S, A)\big], \tag{9.14}$$

*where $S \sim \rho_\pi$ and $A \sim \pi(s, \theta)$. Here, the state distribution $\rho_\pi$ is*

$$\rho_\pi(s) = \Pr_\pi(s|s_0) = \sum_{k=0}^{\infty} \gamma^k \Pr(s_0 \to s, k, \pi) = \left[(I_n - \gamma P_\pi)^{-1}\right]_{s_0 s}, \tag{9.15}$$

*which is the discounted total probability transiting from $s_0$ to $s$ under policy $\pi$.*

By comparing with Theorem 9.1, we notice that the expression of $\nabla_\theta v_\pi(s_0)$ for any $s_0$ is the same as $\nabla_\theta \bar{r}_\pi$. However, the difference is the probability distribution of $S$. Here, $S$ obeys $\rho_\pi$ which is different from $d_\pi$. The proof of the theorem is given below.

**Proof of Theorem 9.2**

*Proof.* First of all, note that $v_\pi(s_0) = \sum_{s\in\mathcal{S}} d_0(s)v_\pi(s)$ where $d_0(s = s_0) = 1$ and $d_0(s \neq s_0) = 0$. Therefore,

$$\nabla_\theta v_\pi(s_0) = \nabla_\theta \sum_s d_0(s)v_\pi(s) = \sum_s d_0(s)\nabla_\theta v_\pi(s).$$

The last equality is because $d_0(s)$ is independent of $\pi$. Substituting the expression of $\nabla_\theta v_\pi(s)$ as in Lemma 9.2 to the above equation yields

$$\begin{aligned}
\nabla_\theta v_\pi(s_0) = \sum_s d_0(s)\nabla_\theta v_\pi(s) &= \sum_s d_0(s)\sum_{s'} \Pr{}_\pi(s'|s)\sum_a \nabla_\theta \pi(a|s',\theta)q_\pi(s',a)\\
&= \sum_{s'}\left(\sum_s d_0(s)\Pr{}_\pi(s'|s)\right)\sum_a \nabla_\theta \pi(a|s',\theta)q_\pi(s',a)\\
&\doteq \sum_{s'}\rho_\pi(s')\sum_a \nabla_\theta \pi(a|s',\theta)q_\pi(s',a)\\
&= \sum_{s'}\rho_\pi(s')\sum_a \pi(a|s',\theta)\nabla_\theta \ln\pi(a|s',\theta)q_\pi(s',a)\\
&= \mathbb{E}\left[\nabla_\theta \ln\pi(A|S',\theta)q_\pi(S',A)\right],
\end{aligned}$$

where $S' \sim \rho_\pi$ and $A \sim \pi(s,\theta)$. Furthermore, since $d_0(s \neq s_0) = 0$, we have

$$\rho_\pi(s') = \sum_s d_0(s)\Pr{}_\pi(s'|s) = \Pr{}_\pi(s'|s_0),$$

which is the discounted total probability transiting from $s_0$ to $s'$ under policy $\pi$. The proof is complete.

The above proof can also be shortened by directly rewriting (9.10) as (9.14) by considering the specific state $s_0$. The reason that we prove in the above way is to show a more general proof that can handle any fixed state distribution $d_0(s)$. □

With the results in Lemma 9.1 and Lemma 9.2, we are ready to derive the gradients of $\bar{v}_\pi$ and $\bar{r}_\pi$.

**Theorem 9.3** (Gradient of $\bar{v}_\pi$ and $\bar{r}_\pi$ in the discounted case)**.** *In the discounted case where $\gamma \in [0,1)$, the gradients of $\bar{v}_\pi$ and $\bar{r}_\pi$ are, respectively,*

$$\nabla_\theta \bar{v}_\pi \approx \frac{1}{1-\gamma}\sum_s d_\pi(s)\sum_a \nabla_\theta \pi(a|s,\theta)q_\pi(s,a)$$

*and*

$$\nabla_\theta \bar{r}_\pi \approx \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a),$$

*where the approximations are more accurate when $\gamma$ is closer to 1.*

It is interesting to note that the gradient of $\bar{r}_\pi$ has the same expression as that in the undiscounted case. This is an important result and is summarized as the policy gradient theorem. Nevertheless, it should be noted that the definition of the action value $q_\pi(s,a)$ is different from the undiscounted case.

**Proof of Theorem 9.3**

It follows from the definition of $\bar{v}_\pi$ that

$$\begin{aligned}\nabla_\theta \bar{v}_\pi &= \nabla_\theta \sum_s d_\pi(s) v_\pi(s) \\ &= \sum_s \nabla_\theta d_\pi(s) v_\pi(s) + \sum_s d_\pi(s) \nabla_\theta v_\pi(s). \end{aligned} \tag{9.16}$$

This equation contains two terms. On the one hand, substituting the expression of $\nabla_\theta v_\pi$ as in (9.12) into the second term gives

$$\begin{aligned}\sum_s d_\pi(s) \nabla_\theta v_\pi(s) &= (d_\pi^T \otimes I_m) \nabla_\theta v_\pi \\ &= (d_\pi^T \otimes I_m) \left[(I_n - \gamma P_\pi)^{-1} \otimes I_m\right] u \\ &= \left[d_\pi^T (I_n - \gamma P_\pi)^{-1}\right] \otimes I_m u. \end{aligned} \tag{9.17}$$

It is noted that

$$d_\pi^T (I_n - \gamma P_\pi)^{-1} = \frac{1}{1-\gamma} d_\pi^T,$$

which can be easily verified by multiplying $(I_n - \gamma P_\pi)$ on both sides of the equation. Therefore, (9.17) becomes

$$\begin{aligned}\sum_s d_\pi(s) \nabla_\theta v_\pi(s) &= \frac{1}{1-\gamma} d_\pi^T \otimes I_m u \\ &= \frac{1}{1-\gamma} \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a). \end{aligned}$$

On the other hand, the first term of (9.16) involves $\nabla_\theta d_\pi$. However, since the second term contains $\frac{1}{1-\gamma}$, the second term becomes dominant and the first term becomes negligible when $\gamma \to 1$. Therefore,

$$\nabla_\theta \bar{v}_\pi \approx \frac{1}{1-\gamma} \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

Furthermore, it follows from $\bar{r}_\pi = (1-\gamma)\bar{v}_\pi$ that

$$\nabla_\theta \bar{r}_\pi = (1-\gamma)\nabla_\theta \bar{v}_\pi \approx \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

The above approximation requires that the first term does not go to infinity when $\gamma \to 1$. A deeper analysis of this problem is omitted here and can be found in [36, Section 4].

### 9.3.2 Gradients in the undiscounted case

We now show how to calculate the gradients of the metrics in the undiscounted case where $\gamma = 1$. This is the first time we introduce the undiscounted case in this book. We first define and analyze state values in the undiscounted case, and then derive the metric gradients.

**State value and the Poisson equation**

If $\gamma = 1$, direct summation of the rewards, $\mathbb{E}[R_{t+1} + R_{t+2} + R_{t+3} + \dots | S_t = s]$, over infinitely long trajectories may diverge. Hence, the state value and action value under policy $\pi$ are defined in a special way as [34]

$$\begin{aligned} v_\pi(s) &\doteq \mathbb{E}[(R_{t+1} - \bar{r}_\pi) + (R_{t+2} - \bar{r}_\pi) + (R_{t+3} - \bar{r}_\pi) + \dots | S_t = s], \\ q_\pi(s,a) &\doteq \mathbb{E}[(R_{t+1} - \bar{r}_\pi) + (R_{t+2} - \bar{r}_\pi) + (R_{t+3} - \bar{r}_\pi) + \dots | S_t = s, A_t = a], \end{aligned}$$

where $\bar{r}_\pi$ is the average reward, which is fixed when $\pi$ is given. Here, $v_\pi(s)$ has different names in the literature such as differential reward [35] or bias [2, Section 8.2.1]. It follows from the above definitions that $v_\pi(s) = \sum_a \pi(a|s,\theta) q_\pi(s,a)$. More importantly, the state values satisfy the following Bellman-like equation:

$$v_\pi(s) = \sum_a \pi(a|s,\theta) \left[ \sum_r p(r|s,a)(r - \bar{r}_\pi) + \sum_{s'} p(s'|s,a) v_\pi(s') \right].$$

Hence, the action value is $q_\pi(s,a) = \sum_r p(r|s,a)(r - \bar{r}_\pi) + \sum_{s'} p(s'|s,a) v_\pi(s')$. The matrix-vector form of the above equation is

$$v_\pi = r_\pi - \bar{r}_\pi \mathbf{1}_n + P_\pi v_\pi, \tag{9.18}$$

where $\mathbf{1}_n = [1,\dots,1]^T \in \mathbb{R}^n$ and $n = |\mathcal{S}|$. Equation (9.18) is similar to the Bellman equation and it has a specific name called *Poisson equation* [35,37]. How to calculate $v_\pi$ from the Poisson equation? The solution is given in the following theorem.

**Theorem 9.4** (Solution of the Poisson equation)**.** *Let*

$$v_\pi^* = (I_n - P_\pi + \mathbf{1}_n d_\pi^T)^{-1} r_\pi. \tag{9.19}$$

*Then, $v_\pi^*$ is a solution to the Poisson equation and any solution has the form of*

$$v_\pi = v_\pi^* + c_\pi \mathbf{1}_n,$$

*where $c_\pi \in \mathbb{R}$ is a constant depending on $\pi$*

This theorem indicates that the solution of $v_\pi$ to the Poisson equation is not unique. Interested readers can find the proof of the theorem given as follows.

**Proof of Theorem 9.4**

**1) Why is the solution of $v_\pi$ not unique?**

Substituting $\bar{r}_\pi = d_\pi^T r_\pi$ into (9.18) gives

$$v_\pi = r_\pi - \mathbf{1}_n d_\pi^T r_\pi + P_\pi v_\pi \tag{9.20}$$

and consequently

$$(I_n - P_\pi) v_\pi = (I_n - \mathbf{1}_n d_\pi^T) r_\pi. \tag{9.21}$$

It is noted that $I_n - P_\pi$ is singular because $(I_n - P_\pi)\mathbf{1}_n = 0$ for any $\pi$. Therefore, the solution to (9.21) is not unique: if $v_\pi^*$ is a solution, then $v_\pi^* + x$ is also a solution for any $x \in \text{Null}(I_n - P_\pi)$. When $P_\pi$ is irreducible, $\text{Null}(I_n - P_\pi) = \text{span}\{\mathbf{1}_n\}$. Hence, any solution to the Poisson equation has the expression of $v_\pi^* + c\mathbf{1}_n$ where $c \in \mathbb{R}$.

**2) While the solution to** (9.20) **is not unique, can we at least find one specific solution?**

The answer is yes. We next show that $v_\pi^*$ in (9.19) is a solution to (9.20). For the sake of simplicity, let

$$A \doteq I_n - P_\pi + \mathbf{1}_n d_\pi^T.$$

Then, $v_\pi^* = A^{-1} r_\pi$, substituting which into (9.20) gives

$$A^{-1} r_\pi = r_\pi - \mathbf{1}_n d_\pi^T r_\pi + P_\pi A^{-1} r_\pi.$$

Recognizing the above equation gives $(-A^{-1} + I_n - \mathbf{1}_n d_\pi^T + P_\pi A^{-1}) r_\pi = 0$ and consequently

$$(-I_n + A - \mathbf{1}_n d_\pi^T A + P_\pi)A^{-1}r_\pi = 0.$$

The term in the brackets in the above equation is zero because $-I_n + A - \mathbf{1}_n d_\pi^T A + P_\pi = -I_n + (I_n - P_\pi + \mathbf{1}_n d_\pi^T) - \mathbf{1}_n d_\pi^T (I_n - P_\pi + \mathbf{1}_n d_\pi^T) + P_\pi = 0$. Therefore, $v_\pi^*$ is a solution.

**3) Why is $A = I_n - P_\pi + \mathbf{1}_n d_\pi^T$ invertible?**

Since $v_\pi^*$ involves $A^{-1}$, it is necessary to show that $A$ is invertible. The analysis is summarized as a lemma below.

**Lemma 9.3.** *The matrix $I_n - P_\pi + \mathbf{1}_n d_\pi^T$ is invertible and its inverse is*

$$\left[I_n - (P_\pi - \mathbf{1}_n d_\pi^T)\right]^{-1} = \sum_{k=1}^{\infty}(P_\pi^k - \mathbf{1}_n d_\pi^T) + I_n.$$

*Proof.* First of all, we state some preliminary results without giving proof. Let $\rho(M)$ be the spectral radius of a matrix $M$. Then, $I - M$ is invertible if $\rho(M) < 1$. Moreover, $\rho(M) < 1$ if and only if $\lim_{k\to\infty} M^k = 0$.

Based on the above facts, we next show that $\lim_{k\to\infty}(P_\pi - \mathbf{1}_n d_\pi^T)^k \to 0$ and then the invertibility of $I_n - (P_\pi - \mathbf{1}_n d_\pi^T)$ immediately follows. To do that, it is noted that

$$(P_\pi - \mathbf{1}_n d_\pi^T)^k = P_\pi^k - \mathbf{1}_n d_\pi^T, \quad k \geq 1,$$

which can be proved by induction. For instance, when $k = 1$, the equation is valid. When $k = 2$, we have

$$\begin{aligned}(P_\pi - \mathbf{1}_n d_\pi^T)^2 &= (P_\pi - \mathbf{1}_n d_\pi^T)(P_\pi - \mathbf{1}_n d_\pi^T)\\ &= P_\pi^2 - P_\pi \mathbf{1}_n d_\pi^T - \mathbf{1}_n d_\pi^T P_\pi + \mathbf{1}_n d_\pi^T \mathbf{1}_n d_\pi^T\\ &= P_\pi^2 - \mathbf{1}_n d_\pi^T,\end{aligned}$$

where the last equality is due to $P_\pi \mathbf{1}_n = \mathbf{1}_n$, $d_\pi^T P_\pi = d_\pi^T$, and $d_\pi^T \mathbf{1}_n = 1$. The case of $k \geq 3$ can be proven similarly and omitted here.

Since $d_\pi$ is the stationary distribution of the state, it holds that $\lim_{k\to\infty} P_\pi^k = d_\pi^T \mathbf{1}_n$ (the properties of the stationary distribution can be found in Section 8.2). Therefore,

$$\lim_{k\to\infty}(P_\pi - \mathbf{1}_n d_\pi^T)^k = \lim_{k\to\infty} P_\pi^k - d_\pi^T \mathbf{1}_n = 0.$$

As a result, $I_n - (P_\pi - \mathbf{1}_n d_\pi^T)$ is invertible. Furthermore, its inverse is given by

$$\begin{aligned}(I_n - (P_\pi - \mathbf{1}_n d_\pi^T))^{-1} &= \sum_{k=0}^{\infty}(P_\pi - \mathbf{1}_n d_\pi^T)^k \\ &= I_n + \sum_{k=1}^{\infty}(P_\pi - \mathbf{1}_n d_\pi^T)^k \\ &= I_n + \sum_{k=1}^{\infty}(P_\pi^k - \mathbf{1}_n d_\pi^T) \\ &= \sum_{k=0}^{\infty}(P_\pi^k - \mathbf{1}_n d_\pi^T) + \mathbf{1}_n d_\pi^T.\end{aligned}$$

□

The proof of Lemma 9.3 is inspired by [36]. However, [36] inaccurately states that $(I_n - P_\pi + \mathbf{1}_n d_\pi^T)^{-1} = \sum_{k=0}^{\infty}(P_\pi^k - \mathbf{1}_n d_\pi^T)$ (see the statement above equation (16) in [36]). The inaccuracy can also be verified by the fact that $\sum_{k=0}^{\infty}(P_\pi^k - \mathbf{1}_n d_\pi^T)\mathbf{1}_n = 0$ and hence $\sum_{k=0}^{\infty}(P_\pi^k - \mathbf{1}_n d_\pi^T)$ is singular, leading to a conflict with the nonsingularity of $A$. Lemma 9.3 corrects this inaccuracy.

### Derivation of gradients

Theorem 9.4 indicates that the value of $v_\pi$ in the undiscounted case is not unique. If we would like to get a unique value of $v_\pi$, we can add more constraints. For example, by assuming there exists a recurrent state, the state value of this recurrent state is zero [35, Section II] and hence the constant $c_\pi$ can be determined. There are also other ways. See, for example, equations (8.6.5)-(8.6.7) in [2].

Although the value of $v_\pi$ is not unique, the solution of $\bar{r}_\pi$ is still unique. In particular, it follows from the Poisson equation that

$$\begin{aligned}\bar{r}_\pi \mathbf{1}_n &= r_\pi + (P_\pi - I_n)v_\pi \\ &= r_\pi + (P_\pi - I_n)(v_\pi^* + c\mathbf{1}_n) \\ &= r_\pi + (P_\pi - I_n)v_\pi^*.\end{aligned}$$

Since $v_\pi$ is not unique and so is $\bar{v}_\pi$, we only calculate the gradient of $\bar{r}_\pi$ in the undiscounted case.

**Theorem 9.5** (Gradient of $\bar{r}_\pi$ in the undiscounted case)**.** *In the undiscounted case, the gradient of the average reward $\bar{r}_\pi$ is*

$$\nabla_\theta \bar{r}_\pi = \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).$$

**Proof of Theorem 9.5**

First of all, it follows from $v_\pi(s)=\sum_a \pi(a|s,\theta)q_\pi(s,a)$ that

$$\begin{aligned}\nabla_\theta v_\pi(s) &= \nabla_\theta\left[\sum_a \pi(a|s,\theta)q_\pi(s,a)\right]\\ &= \sum_a \left[\nabla_\theta\pi(a|s,\theta)q_\pi(s,a)+\pi(a|s,\theta)\nabla_\theta q_\pi(s,a)\right], \end{aligned} \tag{9.22}$$

where $q_\pi(s,a)$ is the action value satisfying

$$\begin{aligned}q_\pi(s,a) &= \sum_r p(r|s,a)(r-\bar{r}_\pi)+\sum_{s'} p(s'|s,a)v_\pi(s')\\ &= r(s,a)-\bar{r}_\pi+\sum_{s'}p(s'|s,a)v_\pi(s').\end{aligned}$$

Here, $r(s,a)=\sum_r rp(r|s,a)$ is independent of $\theta$. Therefore,

$$\nabla_\theta q_\pi = 0-\nabla_\theta\bar{r}_\pi+\sum_{s'}p(s'|s,a)\nabla_\theta v_\pi(s'),$$

substituting which into (9.22) gives

$$\begin{aligned}\nabla_\theta v_\pi(s) &= \sum_a\left[\nabla_\theta\pi(a|s,\theta)q_\pi(s,a)+\pi(a|s,\theta)\left(-\nabla_\theta\bar{r}_\pi+\sum_{s'}p(s'|s,a)\nabla_\theta v_\pi(s')\right)\right]\\ &= \sum_a \nabla_\theta\pi(a|s,\theta)q_\pi(s,a)-\nabla_\theta\bar{r}_\pi+\sum_a\pi(a|s,\theta)\sum_{s'}p(s'|s,a)\nabla_\theta v_\pi(s').\end{aligned} \tag{9.23}$$

Let

$$u(s)\doteq\sum_a \nabla_\theta\pi(a|s,\theta)q_\pi(s,a).$$

Equation (9.23) can be written in a matrix-vector form as

$$\underbrace{\begin{bmatrix}\vdots\\ \nabla_\theta v_\pi(s)\\ \vdots\end{bmatrix}}_{\nabla_\theta v_\pi\in\mathbb{R}^{m|\mathcal{S}|}} = \underbrace{\begin{bmatrix}\vdots\\ u(s)\\ \vdots\end{bmatrix}}_{u\in\mathbb{R}^{m|\mathcal{S}|}} -\mathbf{1}_n\otimes\nabla_\theta\bar{r}_\pi+(P_\pi\otimes I_m)\underbrace{\begin{bmatrix}\vdots\\ \nabla_\theta v_\pi(s')\\ \vdots\end{bmatrix}}_{\nabla_\theta v_\pi\in\mathbb{R}^{m|\mathcal{S}|}},$$

where $m$ is the dimension of the parameter vector $\theta$ and hence equals the dimension of $\nabla_\theta\bar{r}_\pi$ and $v_\pi(s)$. Here, $\otimes$ is the Kronecker product. The above equation can be written in short as

$$\nabla_\theta v_\pi = u - \mathbf{1}_n \otimes \nabla_\theta \bar{r}_\pi + (P_\pi \otimes I_m)\nabla_\theta v_\pi$$

and hence

$$\mathbf{1}_n \otimes \nabla_\theta \bar{r}_\pi = u + (P_\pi \otimes I_m)\nabla_\theta v_\pi - \nabla_\theta v_\pi.$$

Multiplying $d_\pi^T \otimes I_m$ on both sides of the equation gives

$$\begin{aligned}(d_\pi^T \mathbf{1}_n) \otimes \nabla_\theta \bar{r}_\pi &= d_\pi^T \otimes I_m u + (d_\pi^T P_\pi) \otimes I_m \nabla_\theta v_\pi - d_\pi^T \otimes I_m \nabla_\theta v_\pi \\ &= d_\pi^T \otimes I_m u,\end{aligned}$$

which is

$$\begin{aligned}\nabla_\theta \bar{r}_\pi &= d_\pi^T \otimes I_m u \\ &= \sum_s d_\pi(s) u(s) \\ &= \sum_s d_\pi(s) \sum_a \nabla_\theta \pi(a|s,\theta) q_\pi(s,a).\end{aligned}$$

## 9.4 Policy gradient by Monte Carlo estimation

With the gradients of the metrics presented in the previous section, we show in this section how to use the gradient-based method to optimize the metrics and hence find optimal policies.

Consider $J(\theta) = \bar{r}_\pi(\theta)$ or $v_\pi(s_0)$. The gradient-ascent algorithm maximizing $J(\theta)$ is

$$\begin{aligned}\theta_{t+1} &= \theta_t + \alpha \nabla_\theta J(\theta) \\ &= \theta_t + \alpha \mathbb{E}\Big[\nabla_\theta \ln \pi(A|S,\theta_t) q_\pi(S,A)\Big],\end{aligned} \tag{9.24}$$

where $\alpha > 0$ is a constant learning rate. The gradient-ascent algorithm can find a local minima where $\nabla_\theta \bar{r}_\pi(\theta_t) = 0$. Since the expected value on the right-hand side is unknown, we can replace the expected value with a sample:

$$\theta_{t+1} = \theta_t + \alpha \nabla_\theta \ln \pi(a_t|s_t,\theta_t) q_t(s_t,a_t), \tag{9.25}$$

where $q_t(s_t,a_t)$ is an approximation of $q_\pi(s_t,a_t)$. If $q_\pi(s_t,a_t)$ is approximated by Monte Carlo estimation, the algorithm is called *REINFORCE* [38] or *Monte Carlo policy gradient*, which is one of earliest and simplest policy gradient algorithms. Many other policy gradient algorithms such as the actor-critic methods introduced in the next chapter can be obtained by extending REINFORCE.

We examine the interpretation of (9.25) more closely. Since

$$\nabla_\theta \ln \pi(a_t|s_t,\theta_t) = \frac{\nabla_\theta \pi(a_t|s_t,\theta_t)}{\pi(a_t|s_t,\theta_t)},$$

(9.25) can be rewritten as

$$\theta_{t+1} = \theta_t + \alpha \left( \frac{q_t(s_t,a_t)}{\pi(a_t|s_t,\theta_t)} \right) \nabla_\theta \pi(a_t|s_t,\theta_t).$$

The coefficient $\frac{q_t(s_t,a_t)}{\pi(a_t|s_t,\theta_t)}$ can well *balance exploration and exploitation*. First, the coefficient is *proportional* to $q_t(s_t,a_t)$. As a result, if the action value of $(s_t,a_t)$ is great, then the algorithm intends to update the parameter $\theta$ so that the probability of taking that action can be enhanced. Therefore, the algorithm attempts to exploit actions with greater values. Second, the coefficient is *inversely proportional* to $\pi(a_t|s_t,\theta_t)$. As a result, if the probability of taking $(s_t,a_t)$ is small, then the algorithm would update the parameters $\theta$ so that the probability of taking that action can increase. This reflects that the algorithm attempts to explore actions that have a low probability to take.

Since (9.25) uses samples to approximate the expectation in (9.24), it is important to understand the correct ways to do sampling.

- How to sample $S$? In theory, $S$ in $\mathbb{E}[\nabla_\theta \ln \pi(A|S,\theta_t) q_\pi(S,A)]$ obeys either the stationary distribution $d_\pi$ or the discounted total probability distribution $\rho_\pi$ as in (9.15). Either $d_\pi$ or $\rho_\pi$ represents the long-run behavior under $\pi$. Therefore, the ideal sampling way is to execute $\pi(\theta_t)$ for sufficiently many steps and then randomly select a state as $s_t$.
- How to sample $A$? In theory, the sampling of $A$ in $\mathbb{E}[\nabla_\theta \ln \pi(A|S,\theta_t) q_\pi(S,A)]$ must follow the distribution of $\pi(A|S,\theta)$. Therefore, the ideal way to sample $A$ is to select $a_t$ following $\pi(\theta_t)$ at $s_t$. Therefore, the algorithm is on-policy.

The ideal ways for sampling $S$ and $A$ are usually not used in practice due to their low sample efficiency. For example, a more sample-efficient implementation of (9.25) is given in the pseudocode. In this implementation, an episode is generated first following $\pi(\theta_t)$. Then, every state-action pair in the episode is used to update $\{\theta_{t+1}, \theta_{t+2}, \dots\}$. Therefore, it uses the every-visit strategy and is hence more efficient in terms of sample usage. However, this implementation does not follow the ideal sampling ways. For example, the updating of $\{\theta_{t+2}, \theta_{t+3}, \dots\}$ uses the samples generated by $\pi(\theta_t)$, whereas the ideal way for updating $\{\theta_{t+2}, \theta_{t+3}, \dots\}$ is to use the samples generated by $\{\pi(\theta_{t+1}), \pi(\theta_{t+2}), \dots\}$, respectively. As a result, this implementation is off-policy. However, this implementation is more sample efficient because it fully utilizes the samples in an episode. Therefore, it is a trade-off between data efficiency and theoretical correctness. Such kind of trade-off is usually acceptable as long as the estimate of $q_\pi$ is not so inaccurate. With this

**Pseudocode: Policy Gradient by Monte Carlo (REINFORCE)**

```
Initialization: A parameterized function $\pi(a|s,\theta)$, $\gamma \in [0,1)$, and $\alpha > 0$.
Aim: Search for an optimal policy maximizing $J(\theta)$.
For each episode, do
    Select $s_0$ and generate an episode following $\pi(\theta_t)$. Suppose the episode is $\{s_0, a_0, r_1, \dots, s_{T-1}, a_{T-1}, r_T\}$.
    For $t = 0, 1, \dots, T-1$:
        Action value estimate: use $q_t(s_t, a_t) = \sum_{k=t+1}^{T} \gamma^{k-t-1} r_k$ to approximate $q_\pi(s_t, a_t)$
        Policy parameter update: $\theta_{t+1} = \theta_t + \alpha \nabla_\theta \ln \pi(a_t|s_t, \theta_t) q_t(s_t, a_t)$
```

point in mind, we know that the algorithm would not work well given a single sufficiently long episode. Instead, we should generate episodes as often as possible using the latest updated policies so that $q_\pi$ can be estimated more accurately.

## 9.5 Summary

In this chapter, we introduced policy gradient methods, which provide an efficient and powerful way to handle large or even continuous state and action spaces. Policy gradient methods are *policy-based*. It is a big step forward because all the methods in the previous chapters are *value-based*. The basic idea of policy gradient, as shown in this chapter, is very simple: that is to select an appropriate scalar metric and then optimize it by a gradient-ascent algorithm. While the expectation is the gradient-ascent algorithm is unknown, we can approximate it by using experience samples.

The most complicated part of policy gradient methods is the derivation of the gradients of the metrics. That is because the mathematical derivation is nontrivial and, in the meantime, we have to distinguish different metrics and the discounted and undiscounted cases. Fortunately, as suggested by the policy gradient theorem, the gradients of the metrics share a unified and elegant expression.

The policy gradient algorithm REINFORCE introduced in this chapter is based on Monte Carlo estimation. It is the most basic policy gradient algorithm and important to understand more advanced algorithms. In the next chapter, the algorithm will be extended to another class of policy gradient methods called actor-critic.

## 9.6 Q&A

– Q: What is the basic idea of policy gradient?

A: The basic idea of policy gradient is simple: that is to find an optimal policy by maximizing a metric using the gradient-ascent methods. The complication emerges when we want to select appropriate metrics to define optimal policies, calculate the gradients of the metric, and use samples to approximate the gradient.

– Q: How many definitions are there for the metric of average reward?

  A: There are two definitions. The first definition as shown in (9.1) is a weighted average of the expected reward that can be obtained starting from each state. The second definition as shown in (9.2) is the expected undiscounted total reward that can be obtained starting from a state. The two definitions are equivalent as shown in (9.3).

– Q: Why do we need to care about undiscounted cases?

  A: Before this chapter, we only considered the discounted case. In this chapter, we introduced the undiscounted case. The reason that we care about the undiscounted case is that it can handle continuous tasks without starting states.

– Q: While the definition and the gradient of the average reward $\bar{r}_\pi$ do not involve the discount rate, why do we need to distinguish the discounted and undiscounted cases?

  A: The reason is that, when we attempt to calculate the gradient of $\bar{r}_\pi$, we need to consider state values whose definition involves the discount rate. In particular, we need to respectively study the Bellman equation in the discounted case and the Poisson equation in the undiscounted case.