In the Name of God

Iran University of Science and Technology

Reinforcement Learning in Control

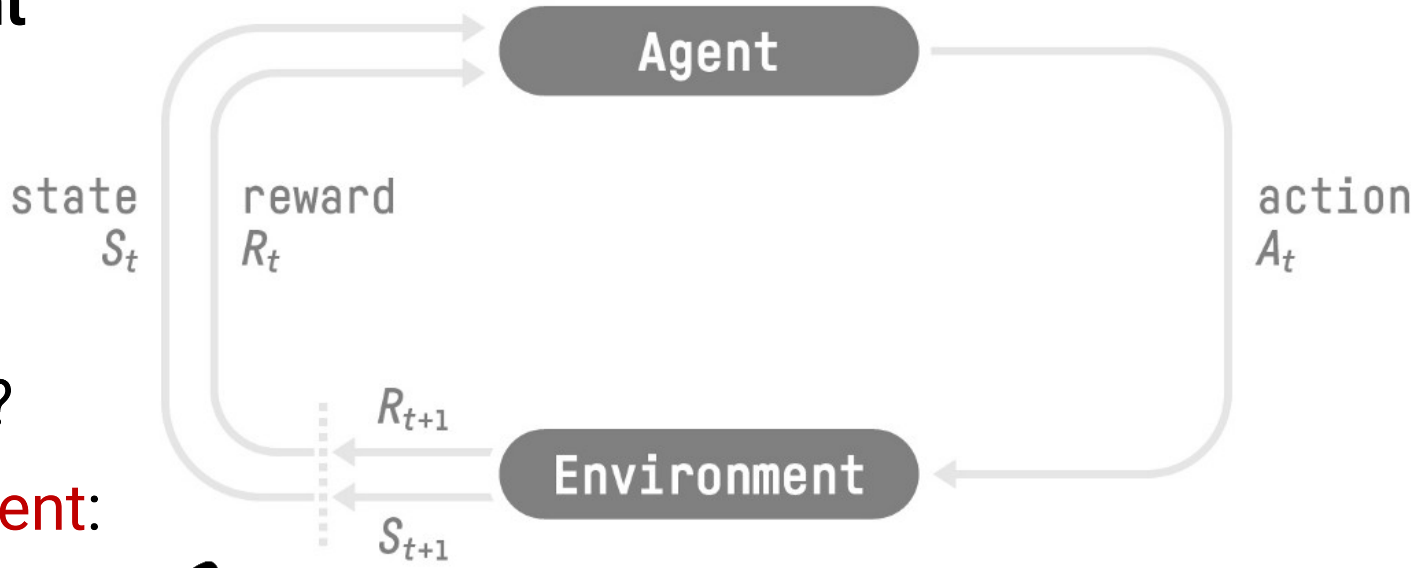
Dr. Saeed Shamaghdari

Electrical Engineering Department
Control Group

Fall 2025 | 4041

Monte Carlo Methods

Knowledge of the Environment



Lack of complete information?

Experience from the environment:

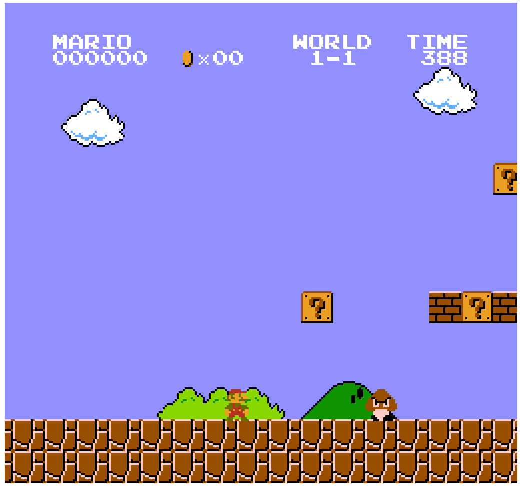
$S \to A \to R$

- Real environment vs. simulated environment

Solution?

- Estimating the dynamics of the environment → Solving with DP
- Estimating the value function from measurements (Learning)

Assumption



Episodic Task

Estimating the value at the end of each episode?

One approach: Averaging the returns experienced from each state

Monte Carlo

Observing more returns → Better estimation of expected return

Definition

First Visit: The first time state s is visited in an episode

- **Monte Carlo First-Visit Method**:
  Estimate $V_\pi(s)$ based on the average return following the **first visit** to s in each episode

- **Monte Carlo Every-Visit Method**:
  Estimate $V_\pi(s)$ based on the average return following **every visit** to s in each episode

Algorithm

First-visit MC prediction, for estimating $V \approx v_\pi$

Input: a policy π to be evaluated

Initialize:

$V(s) \in \mathbb{R}$, arbitrarily, for all $s \in \mathcal{S}$

$Returns(s) \leftarrow$ an empty list, for all $s \in \mathcal{S}$

Loop forever (for each episode):

Generate an episode following π: $S_0, A_0, R_1, S_1, A_1, R_2, \ldots, S_{T-1}, A_{T-1}, R_T$

$G \leftarrow 0$

Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:

$G \leftarrow \gamma G + R_{t+1}$

Unless S_t appears in $S_0, S_1, \ldots, S_{t-1}$:

Append G to $Returns(S_t)$

$V(S_t) \leftarrow \text{average}(Returns(S_t))$

Example: Blackjack

Rewards of $+1$, -1, and 0

Example: Blackjack

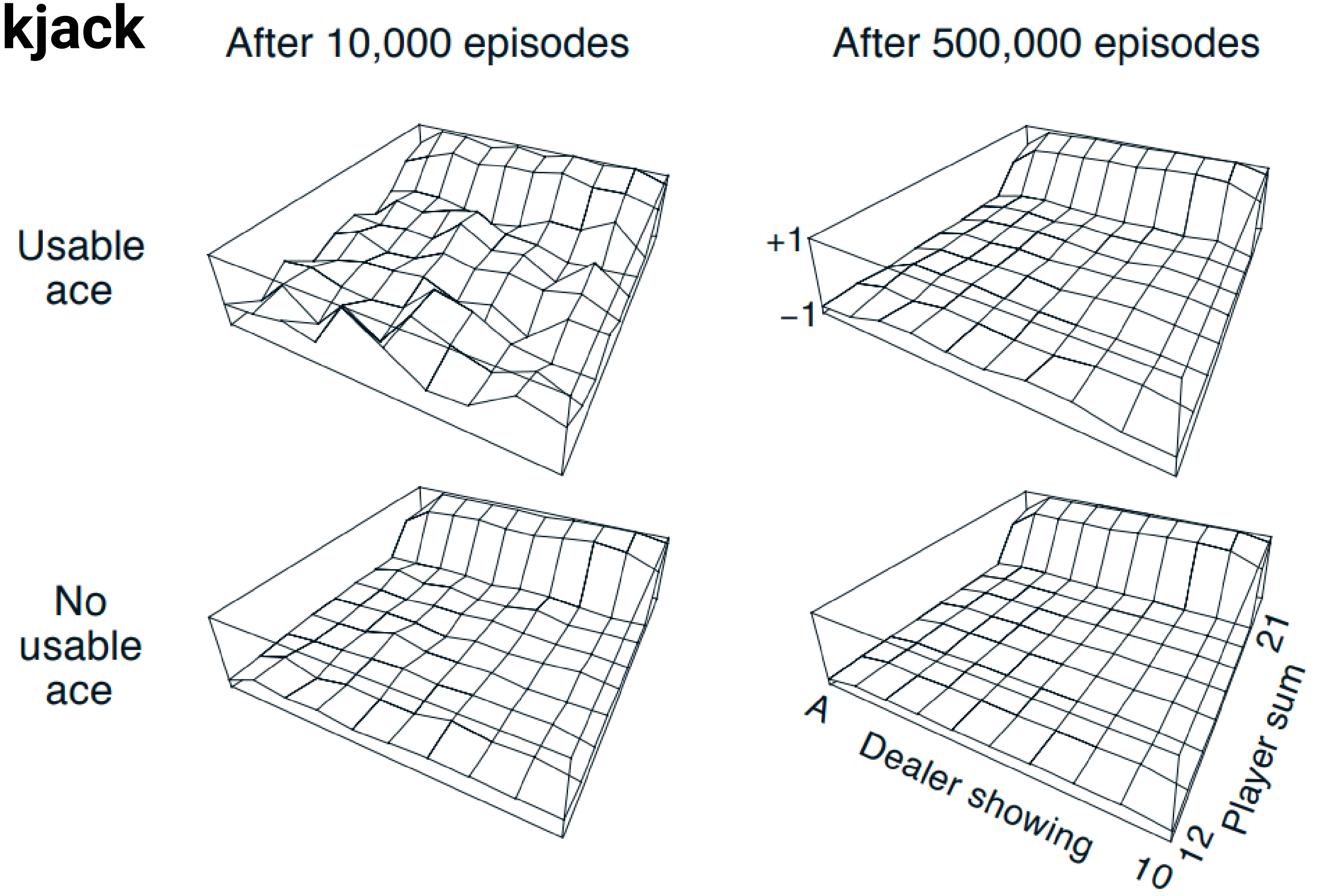


Figure 5.1: Approximate state-value functions for the blackjack policy that sticks only on 20 or 21, computed by Monte Carlo policy evaluation. ■

Monte Carlo Estimation of Action Values

Action Value or State Value?

$$q_\pi(s, a)$$

- **Estimation idea**:
  Visiting state s and taking action a (First Visit)
- **Challenge in estimating q?**
  Not all (s, a) pairs are visited! (Especially under a deterministic policy)

Solution for visiting different *(s, a)* pairs?
Continuous **exploration**

Monte Carlo Estimation of Action Values

Solution for visiting different *(s, a)* pairs?
Continuous **exploration**

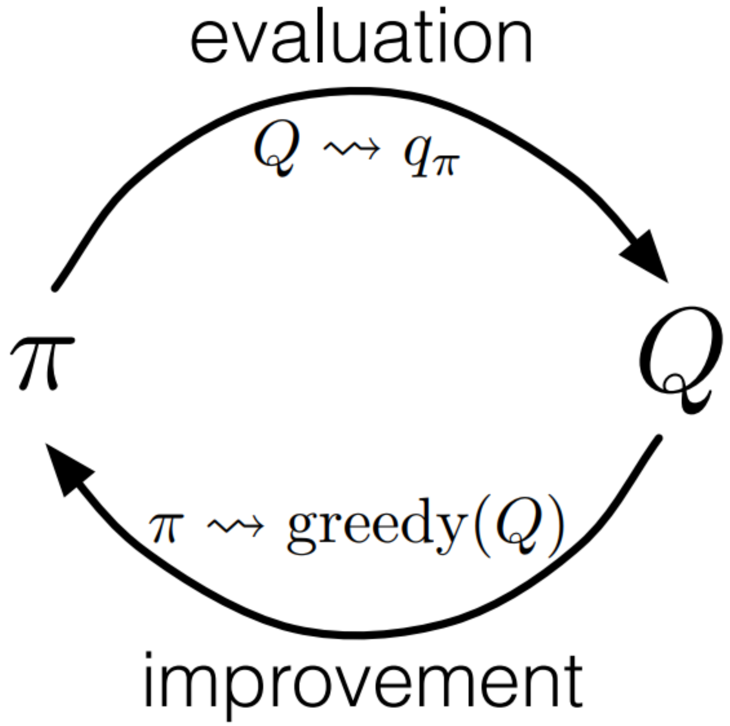
Solutions:

1. Exploring Starts:
Start each episode with a state-action pair (s, a) that has a non-zero probability of being selected.

Q: Number of visits in infinite episode repetitions?
All *(s, a)* pairs will be visited infinitely often (assuming proper exploration)

2. Stochastic Policy

Monte Carlo Control



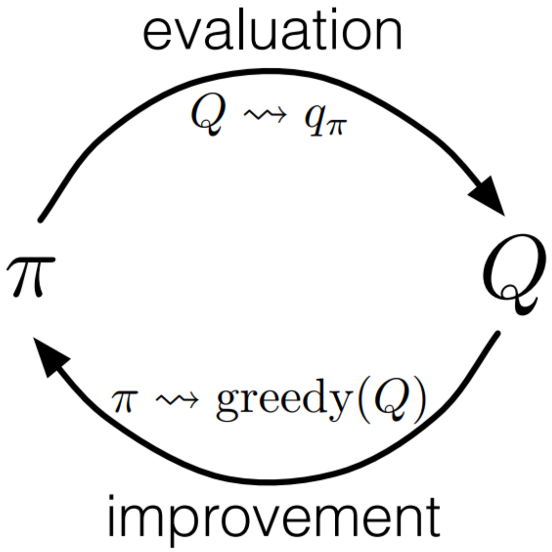
generalized policy iteration

$$\pi_0 \xrightarrow{\text{E}} q_{\pi_0} \xrightarrow{\text{I}} \pi_1 \xrightarrow{\text{E}} q_{\pi_1} \xrightarrow{\text{I}} \pi_2 \xrightarrow{\text{E}} \cdots \xrightarrow{\text{I}} \pi_* \xrightarrow{\text{E}} q_*$$

Q: Problem with deterministic policy?

Monte Carlo Control

- At the end of each episode, both **Policy Evaluation (PE)** and **Policy Improvement (PI)** are performed
- Since only a part of the (s, a) space is updated at the end of each episode, this is considered **Generalized Policy Iteration (GPI)**



generalized policy iteration

Greedy policy selection:

$$\pi(s) \doteq \arg\max_a q(s, a)$$

Monte Carlo Control

Theorem (Policy Improvement)

for all $s \in \mathcal{S}$

$$
\begin{aligned}
q_{\pi_k}(s, \pi_{k+1}(s)) &= q_{\pi_k}(s, \arg\max_a q_{\pi_k}(s, a)) \\
&= \max_a q_{\pi_k}(s, a) \\
&\geq q_{\pi_k}(s, \pi_k(s)) \\
&\geq v_{\pi_k}(s).
\end{aligned}
$$

Two requirements:

- Episodes with **Exploring Starts (ES)**, **Infinite number of episodes**

Monte Carlo Control

Two requirements:

- Episodes with **Exploring Starts (ES)**, **Infinite number of episodes**

Solving the infinite episodes problem:

$\longrightarrow$ Use **Value Iteration**

At the end of each episode:

- **Policy Evaluation**
- **Policy Improvement**

Algorithm

Monte Carlo ES (Exploring Starts), for estimating $\pi \approx \pi_*$

Initialize:
  $\pi(s) \in \mathcal{A}(s)$ (arbitrarily), for all $s \in \mathcal{S}$
  $Q(s,a) \in \mathbb{R}$ (arbitrarily), for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$
  $Returns(s,a) \leftarrow$ empty list, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

Loop forever (for each episode):
  Choose $S_0 \in \mathcal{S}$, $A_0 \in \mathcal{A}(S_0)$ randomly such that all pairs have probability > 0
  Generate an episode from S_0, A_0, following π: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
  $G \leftarrow 0$
  Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:
    $G \leftarrow \gamma G + R_{t+1}$
    Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1 \ldots, S_{t-1}, A_{t-1}$:
      Append G to $Returns(S_t, A_t)$
      $Q(S_t, A_t) \leftarrow \text{average}(Returns(S_t, A_t))$
      $\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$

Monte Carlo Control without Exploring Starts

Solution:

- Stochastic Policy

Guaranteeing selection of all actions in infinite repetitions

Methods:

- On-Policy
- Off-Policy

Which type is the MC-ES method?

Monte Carlo Control without Exploring Starts

Soft-Policy:

$$\pi(a|s) > 0$$

Epsilon Soft-Policy:

$$\pi(a|s) \geq \frac{\varepsilon}{|\mathcal{A}(s)|}$$

Epsilon Greedy Policy:

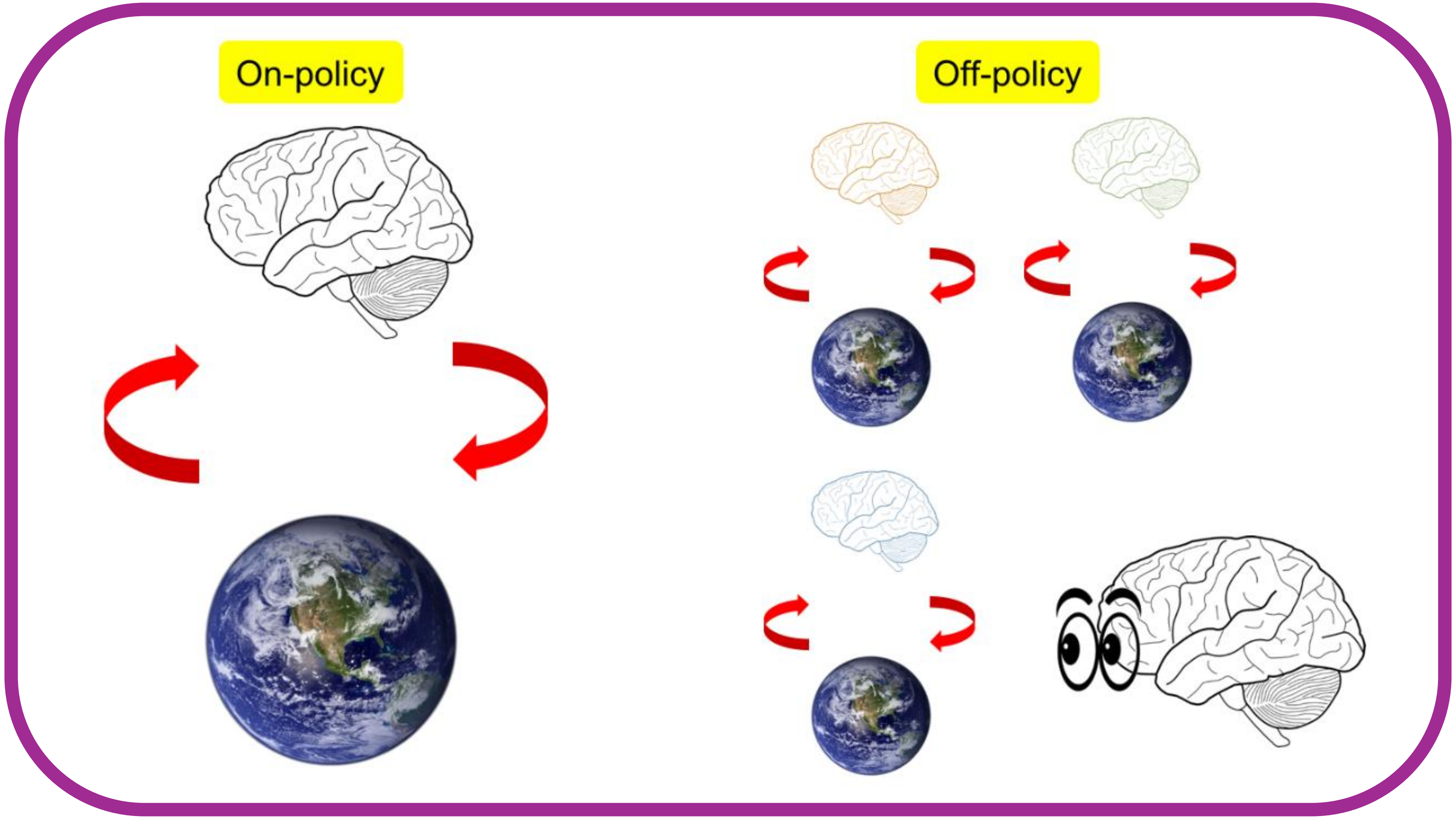
$$\text{probability of selection of} \begin{cases} \text{all nongreedy actions} & \frac{\varepsilon}{|\mathcal{A}(s)|} \\ \text{greedy actions} & 1 - \varepsilon + \frac{\varepsilon}{|\mathcal{A}(s)|} \end{cases}$$

Monte Carlo Control without Exploring Starts

Proof of the superiority of the ε-greedy policy π' over soft policies π:

$$
\begin{aligned}
q_\pi(s, \pi'(s)) &= \sum_a \pi'(a|s) q_\pi(s,a) \\
&= \frac{\varepsilon}{|\mathcal{A}(s)|} \sum_a q_\pi(s,a) \;+\; (1-\varepsilon) \max_a q_\pi(s,a) \\
&\geq \frac{\varepsilon}{|\mathcal{A}(s)|} \sum_a q_\pi(s,a) \;+\; (1-\varepsilon) \sum_a \frac{\pi(a|s) - \frac{\varepsilon}{|\mathcal{A}(s)|}}{1-\varepsilon} q_\pi(s,a) \\
&= \frac{\varepsilon}{|\mathcal{A}(s)|} \sum_a q_\pi(s,a) \;-\; \frac{\varepsilon}{|\mathcal{A}(s)|} \sum_a q_\pi(s,a) \;+\; \sum_a \pi(a|s) q_\pi(s,a) \\
&= v_\pi(s).
\end{aligned}
$$

On-policy RL vs Off-policy RL



Algorithm

On-policy first-visit MC control (for ε-soft policies), estimates $\pi \approx \pi_*$

Algorithm parameter: small $\varepsilon > 0$

Initialize:

  $\pi \leftarrow$ an arbitrary ε-soft policy

  $Q(s,a) \in \mathbb{R}$ (arbitrarily), for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

  $Returns(s,a) \leftarrow$ empty list, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$

Repeat forever (for each episode):

  Generate an episode following π: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$

  $G \leftarrow 0$

  Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:

    $G \leftarrow \gamma G + R_{t+1}$

    Unless the pair S_t, A_t appears in $S_0, A_0, S_1, A_1 \ldots, S_{t-1}, A_{t-1}$:

      Append G to $Returns(S_t, A_t)$

      $Q(S_t, A_t) \leftarrow \text{average}(Returns(S_t, A_t))$

      $A^* \leftarrow \arg\max_a Q(S_t, a)$ (with ties broken arbitrarily)

      For all $a \in \mathcal{A}(S_t)$:

$$\pi(a|S_t) \leftarrow \begin{cases} 1 - \varepsilon + \varepsilon/|\mathcal{A}(S_t)| & \text{if } a = A^* \\ \varepsilon/|\mathcal{A}(S_t)| & \text{if } a \neq A^* \end{cases}$$

Monte Carlo Control without Exploring Starts

Second approach for the proof:

Transfer the randomness of the policy to the environment

For action *a*:

- $P = 1 - \varepsilon \longrightarrow$ Same as in the original environment
- $P = \varepsilon \longrightarrow$ Equivalent to the environment's response to a randomly selected action with a uniform distribution

Monte Carlo Control without Exploring Starts

Second approach for the proof:

Transfer the randomness of the policy to the environment

$\tilde{V}_*$: value function for the new environment

$$\begin{aligned}\widetilde{v}_*(s) &= (1-\varepsilon)\max_a \widetilde{q}_*(s,a) + \frac{\varepsilon}{|\mathcal{A}(s)|}\sum_a \widetilde{q}_*(s,a) \\ &= (1-\varepsilon)\max_a \sum_{s',r} p(s',r|s,a)\Big[r+\gamma\widetilde{v}_*(s')\Big] \\ &\quad + \frac{\varepsilon}{|\mathcal{A}(s)|}\sum_a \sum_{s',r} p(s',r|s,a)\Big[r+\gamma\widetilde{v}_*(s')\Big]\end{aligned}$$

Monte Carlo Control without Exploring Starts

Second approach for the proof:

Transfer the randomness of the policy to the environment

$\tilde{V}_*$: value function for the new environment

Under convergence conditions:

$$
\begin{aligned}
v_\pi(s) &= (1-\varepsilon)\max_a q_\pi(s,a) + \frac{\varepsilon}{|\mathcal{A}(s)|}\sum_a q_\pi(s,a) \\
&= (1-\varepsilon)\max_a \sum_{s',r} p(s',r|s,a)\Big[r+\gamma v_\pi(s')\Big] \\
&\quad + \frac{\varepsilon}{|\mathcal{A}(s)|}\sum_a \sum_{s',r} p(s',r|s,a)\Big[r+\gamma v_\pi(s')\Big]
\end{aligned}
$$

Result:

$$v_\pi = \widetilde{v}_*$$

Off-Policy Prediction via Importance Sampling

Generating episodes using policy μ

Estimating policy π

Where:

$$\mu \neq \pi$$

Target Policy: π

Behavior Policy: μ

Implicit requirement:

Stochastic μ

Example:

Epsilon Greedy

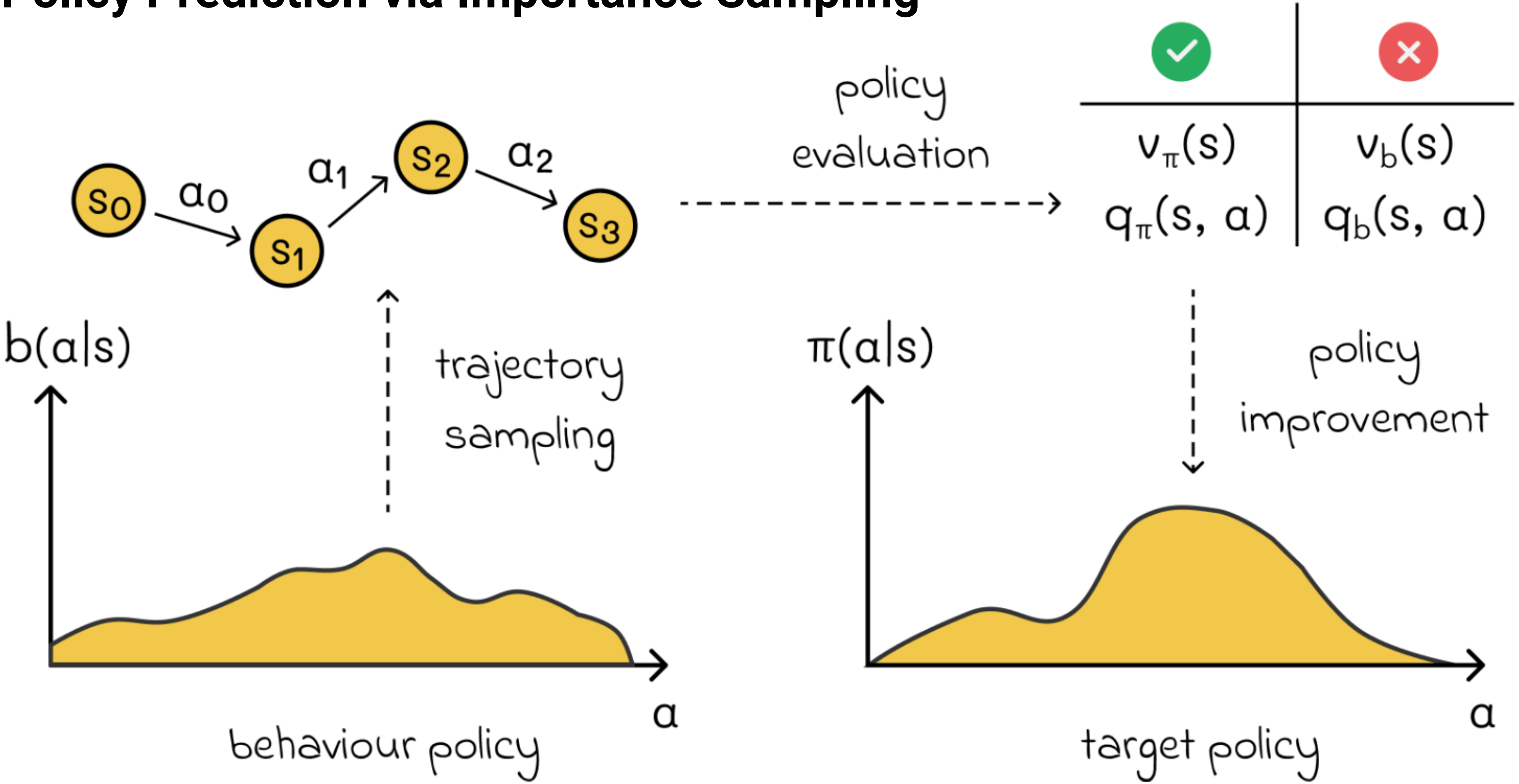
Off-policy

Requirement of estimating V_π using episodes from μ?

Coverage assumption

$$\mu(a|s) > 0 \ \rightarrow \ \pi\,(a|s) > 0$$

Off-Policy Prediction via Importance Sampling



Importance Sampling

Estimating V_π using episodes from μ

Weighting returns obtained under μ by the likelihood ratio (ρ) of trajectories under μ and π

Starting state S_t

Probability of s-a pairs under policy: π

$$\text{trajectory, } A_t, S_{t+1}, A_{t+1}, \ldots, S_T$$

$$\begin{aligned}
&\Pr\{A_t, S_{t+1}, A_{t+1}, \ldots, S_T \mid S_t, A_{t:T-1} \sim \pi\} \\
&\quad = \pi(A_t|S_t)p(S_{t+1}|S_t, A_t)\pi(A_{t+1}|S_{t+1}) \cdots p(S_T|S_{T-1}, A_{T-1}) \\
&\quad = \prod_{k=t}^{T-1} \pi(A_k|S_k)p(S_{k+1}|S_k, A_k),
\end{aligned}$$

Importance Sampling Ratio

$$\rho_{t:T-1} \doteq \frac{\prod_{k=t}^{T-1} \pi(A_k|S_k)p(S_{k+1}|S_k, A_k)}{\prod_{k=t}^{T-1} b(A_k|S_k)p(S_{k+1}|S_k, A_k)} = \prod_{k=t}^{T-1} \frac{\pi(A_k|S_k)}{b(A_k|S_k)}$$

First Visit – Every Visit

$$\{\rho_t^{T(t)}\}_{t\in\mathcal{T}(s)}$$

Note: Independence from $\boldsymbol{p}$

$\mathcal{T}(s)$: set of all time steps in which state s is visited

Also, let T denote the first time of termination following time t.

Example of Importance Sampling: Estimating average household income

Importance Sampling Ratio

Estimation method for V_π (Ordinary):

$$V(s) = \frac{\sum_{t\in\mathcal{T}(s)} \rho_t^{T(t)} G_t}{|\mathcal{T}(s)|}$$

Another estimation method for V_π (Weighted):

$$V(s) = \frac{\sum_{t\in\mathcal{T}(s)} \rho_t^{T(t)} G_t}{\sum_{t\in\mathcal{T}(s)} \rho_t^{T(t)}}$$

Q Box!

Comparing two estimation methods for a single observation?

Comparing two estimation methods for ρ = 10?

Variance analysis:

- The ordinary method is unbounded due to the unbounded variance of ρ
- Assuming bounded return, estimation variance is bounded and $\rightarrow 0$

Importance Sampling Ratio

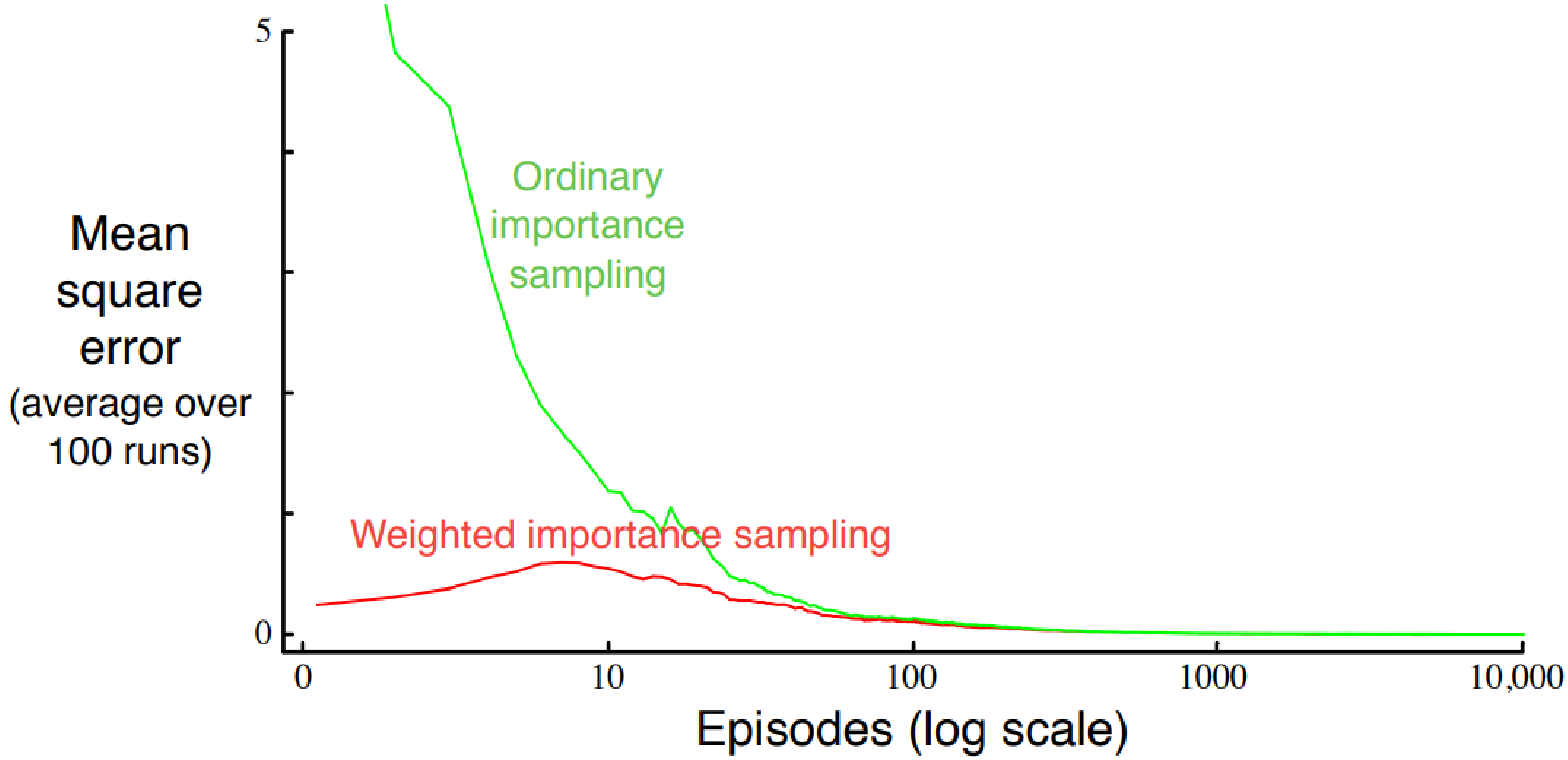
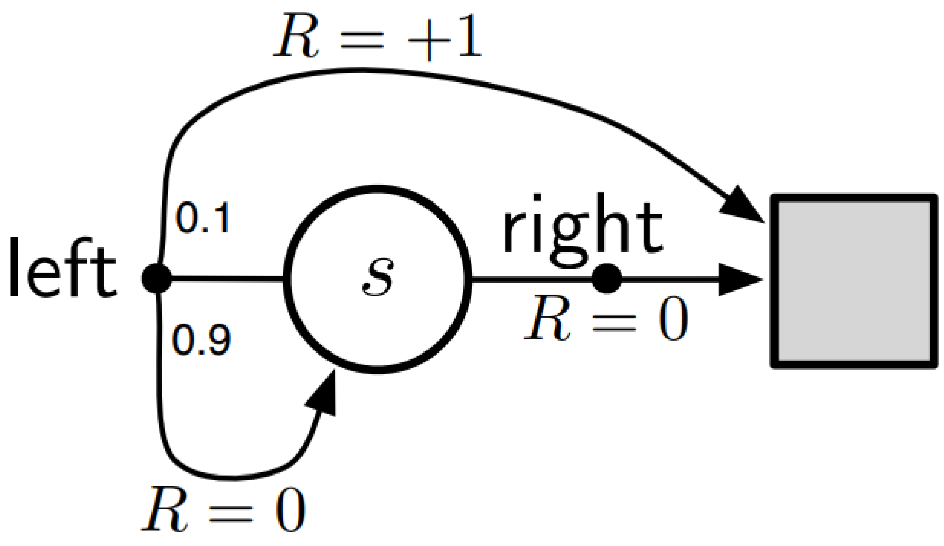


Figure 5.3: Weighted importance sampling produces lower error estimates of the value of a single blackjack state from off-policy episodes. ■

Ten Independent Runs of the First Visit MC Algorithm using ordinary importance sampling



$$\pi(\text{left}|s) = 1$$

$$b(\text{left}|s) = \frac{1}{2}$$

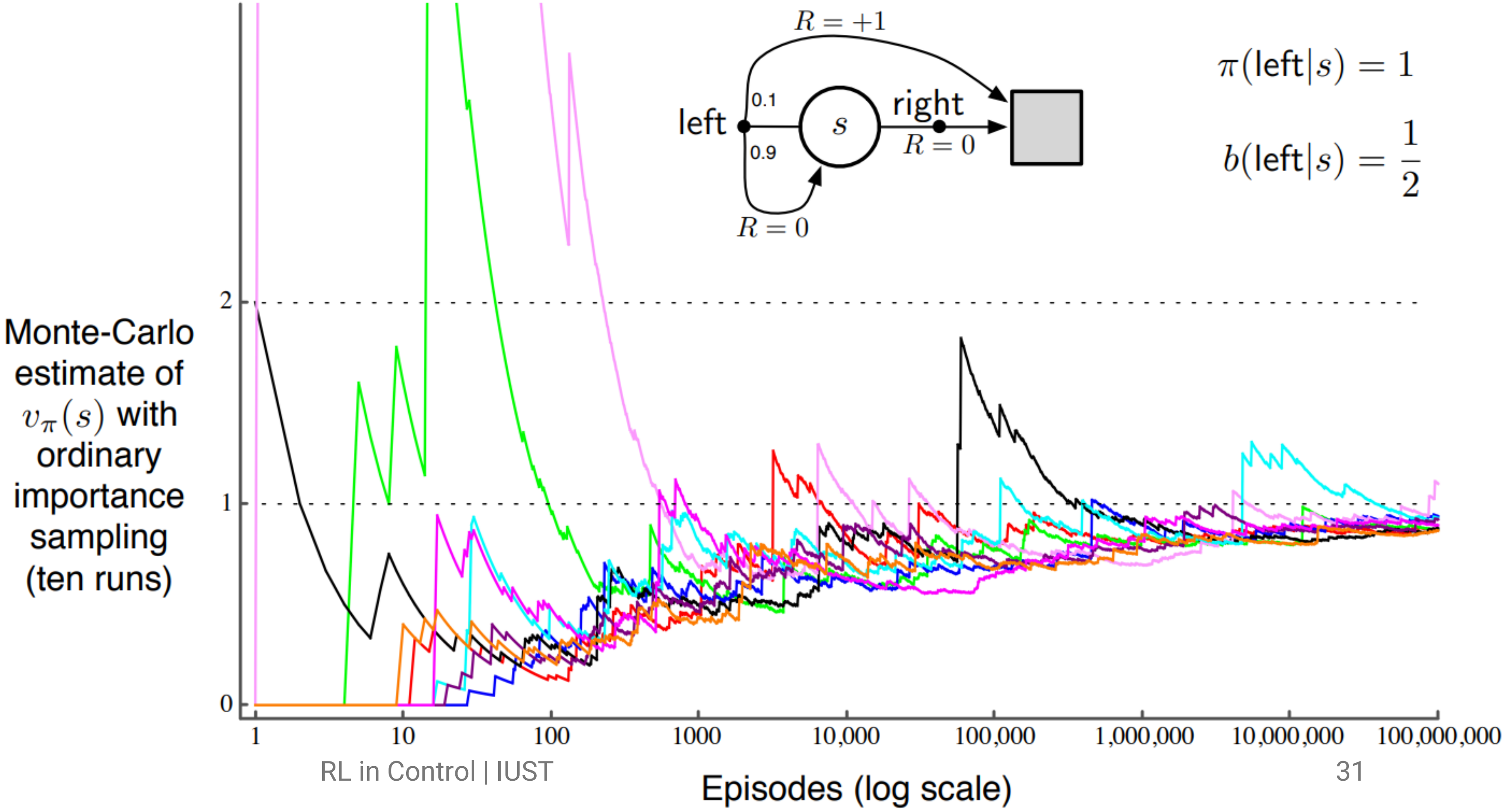
the target policy that always selects left.

behavior policy that selects right and left with equal probability

Ten Independent Runs of the First Visit MC Algorithm

Lack of convergence after 10^6 episodes!

variance of the importance-sampling-scaled returns is infinite



Ten Independent Runs of the First Visit MC Algorithm

$$\mathrm{Var}[X] \doteq \mathbb{E}\left[\left(X-\bar{X}\right)^2\right] = \mathbb{E}\left[X^2 - 2X\bar{X} + \bar{X}^2\right] = \mathbb{E}\left[X^2\right] - \bar{X}^2$$

$$\mathbb{E}_b\left[\left(\prod_{t=0}^{T-1} \frac{\pi(A_t|S_t)}{b(A_t|S_t)} G_0\right)^2\right]$$

$$\begin{aligned}
&= \frac{1}{2} \cdot 0.1 \left(\frac{1}{0.5}\right)^2 && \text{(the length 1 episode)} \\
&\quad + \frac{1}{2} \cdot 0.9 \cdot \frac{1}{2} \cdot 0.1 \left(\frac{1}{0.5}\frac{1}{0.5}\right)^2 && \text{(the length 2 episode)} \\
&\quad + \frac{1}{2} \cdot 0.9 \cdot \frac{1}{2} \cdot 0.9 \cdot \frac{1}{2} \cdot 0.1 \left(\frac{1}{0.5}\frac{1}{0.5}\frac{1}{0.5}\right)^2 && \text{(the length 3 episode)} \\
&\quad + \cdots \\
&= 0.1 \sum_{k=0}^{\infty} 0.9^k \cdot 2^k \cdot 2 = 0.2 \sum_{k=0}^{\infty} 1.8^k = \infty. && \blacksquare
\end{aligned}$$

Incremental Implementation

$$V_n \doteq \frac{\sum_{k=1}^{n-1} W_k G_k}{\sum_{k=1}^{n-1} W_k}, \qquad n \geq 2,$$

$$V_{n+1} \doteq V_n + \frac{W_n}{C_n}\Big[G_n - V_n\Big], \qquad n \geq 1,$$

and

$$C_{n+1} \doteq C_n + W_{n+1},$$

Algorithm

Off-policy MC prediction (policy evaluation) for estimating $Q \approx q_\pi$

Input: an arbitrary target policy π
Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
    $Q(s,a) \in \mathbb{R}$ (arbitrarily)
    $C(s,a) \leftarrow 0$

Loop forever (for each episode):
    $b \leftarrow$ any policy with coverage of π
    Generate an episode following b: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
    $G \leftarrow 0$
    $W \leftarrow 1$
    Loop for each step of episode, $t = T-1, T-2, \ldots, 0$, while $W \neq 0$:
        $G \leftarrow \gamma G + R_{t+1}$
        $C(S_t, A_t) \leftarrow C(S_t, A_t) + W$
        $Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{W}{C(S_t, A_t)} [G - Q(S_t, A_t)]$
        $W \leftarrow W \frac{\pi(A_t|S_t)}{b(A_t|S_t)}$

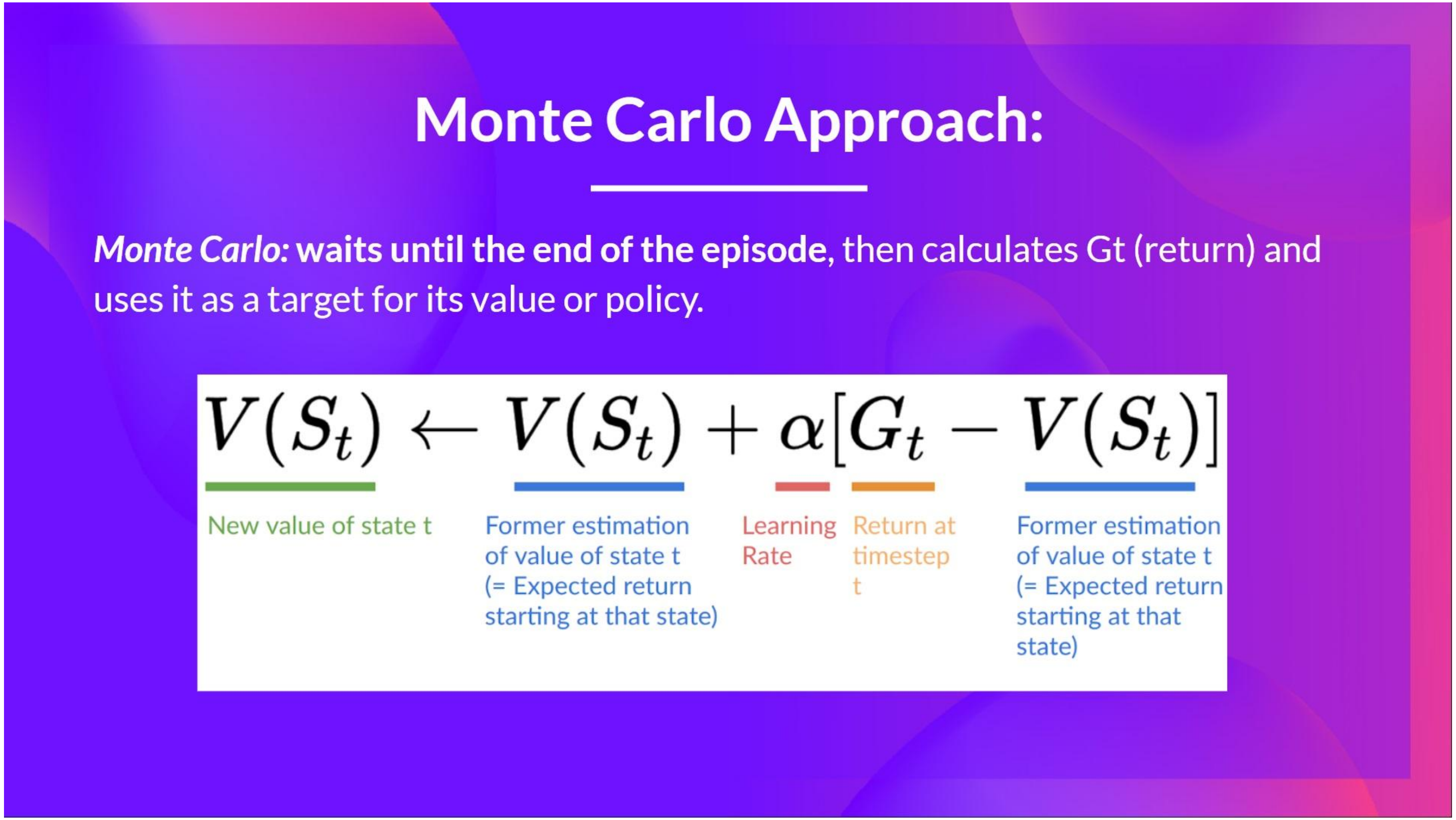
Algorithm

Off-policy MC control, for estimating $\pi \approx \pi_*$

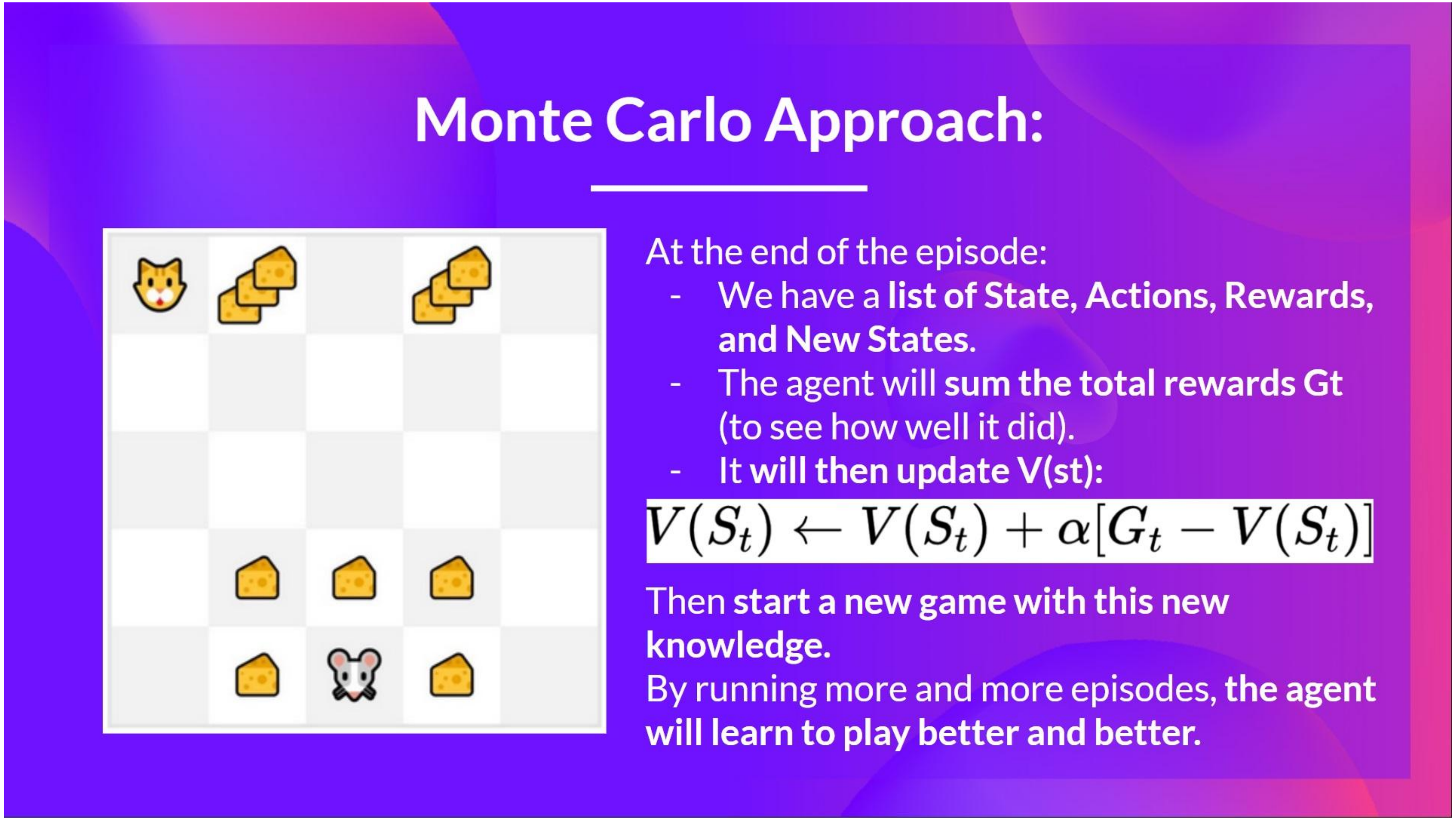
Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
  $Q(s,a) \in \mathbb{R}$ (arbitrarily)
  $C(s,a) \leftarrow 0$
  $\pi(s) \leftarrow \arg\max_a Q(s,a)$ (with ties broken consistently)

Loop forever (for each episode):
  $b \leftarrow$ any soft policy
  Generate an episode using b: $S_0, A_0, R_1, \ldots, S_{T-1}, A_{T-1}, R_T$
  $G \leftarrow 0$
  $W \leftarrow 1$
  Loop for each step of episode, $t = T-1, T-2, \ldots, 0$:
    $G \leftarrow \gamma G + R_{t+1}$
    $C(S_t, A_t) \leftarrow C(S_t, A_t) + W$
    $Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \frac{W}{C(S_t,A_t)} [G - Q(S_t, A_t)]$
    $\pi(S_t) \leftarrow \arg\max_a Q(S_t, a)$ (with ties broken consistently)
    If $A_t \neq \pi(S_t)$ then exit inner Loop (proceed to next episode)
    $W \leftarrow W \frac{1}{b(A_t|S_t)}$

Recap …


Monte Carlo Approach:
Monte Carlo: **waits until the end of the episode**, then calculates Gt (return) and uses it as a target for its value or policy.
$V(S_t) \leftarrow V(S_t) + \alpha[G_t - V(S_t)]$
New value of state t
Former estimation of value of state t (= Expected return starting at that state)
Learning Rate
Return at timestep t
Former estimation of value of state t (= Expected return starting at that state)

Recap …


Monte Carlo Approach:
At the end of the episode:
- We have a list of State, Actions, Rewards, and New States.
- The agent will sum the total rewards Gt (to see how well it did).
- It will then update V(st):
$V(S_t) \leftarrow V(S_t) + \alpha[G_t - V(S_t)]$
Then start a new game with this new knowledge.
By running more and more episodes, the agent will learn to play better and better.