Lecture 18 - March 21, 2023

- Temporal Difference Learning
  - TD(0)
  - SARSA
  - Q-Learning
  - On-Policy Vs. Off-Policy
  - Expected SARSA
  - Double Q-Learning
  - Multi-step Bootstrapping
  - SARSA-Lambda
  - Actor-Critic Method

HW4 → Due March 31

Exam 2 → Tues, April 4

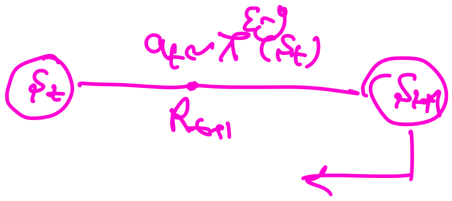
Project 3 → Due April 14

TA's office hour: Wendsdays, 2pm-3pm (in-person)
Fridays, 2pm-3pm (virtual)

SARSA & Q-Learning

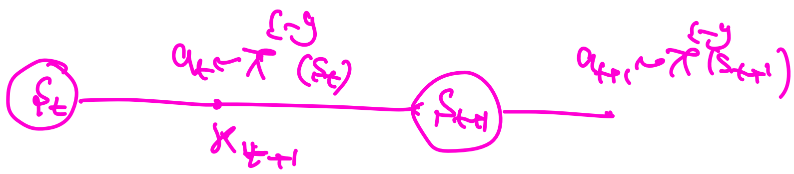
Q-Learning

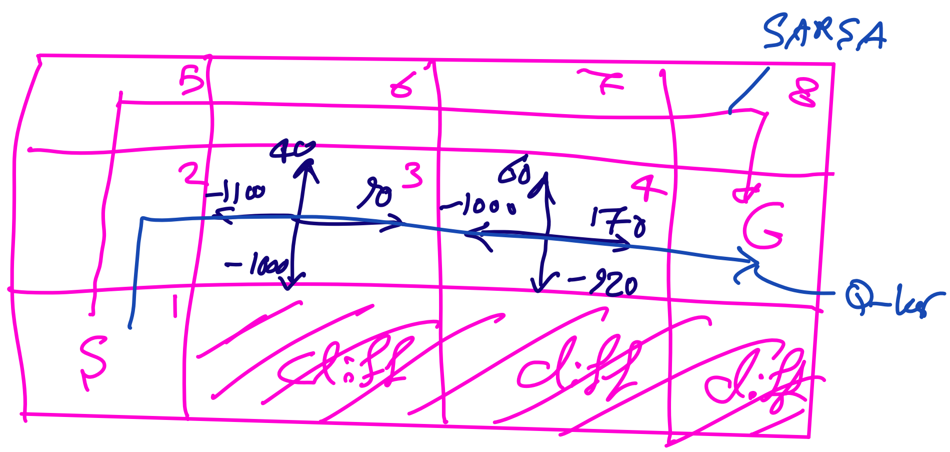
$$Q(s_t,a_t) = Q(s_t,a_t) + \alpha\left[R_{t+1} + \gamma \max_{a\in A} Q(s_{t+1},a) - Q(s_t,a_t)\right]$$

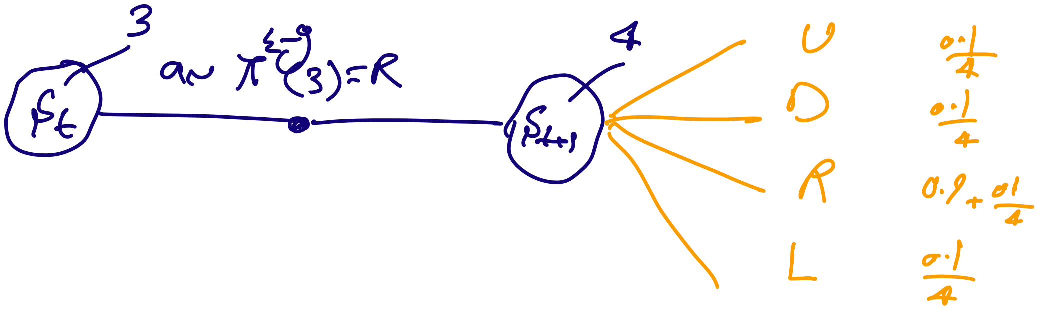


SARSA:

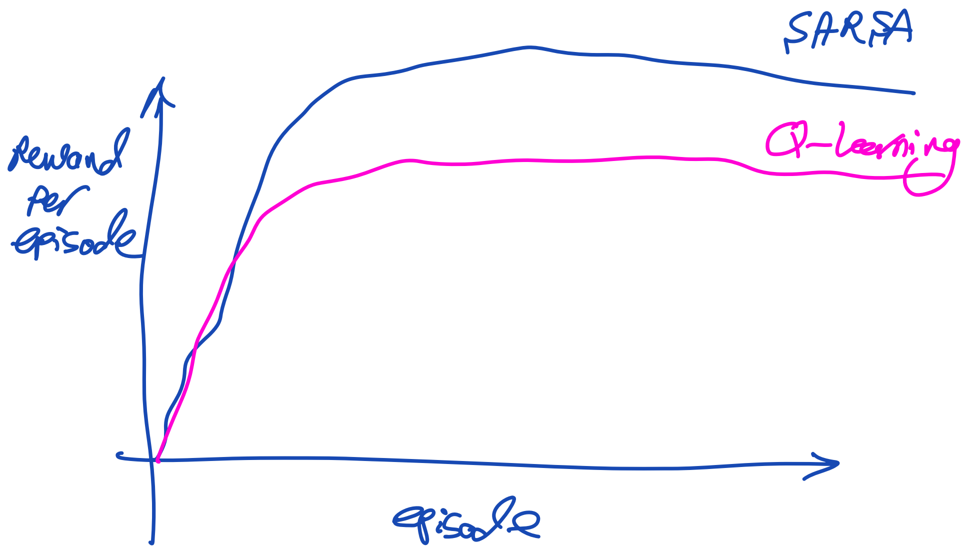
$$Q(s_t,a_t) = Q(s_t,a_t) + \alpha\left[R_{t+1} + \gamma\, Q(s_{t+1},a_{t+1}) - Q(s_t,a_t)\right]$$

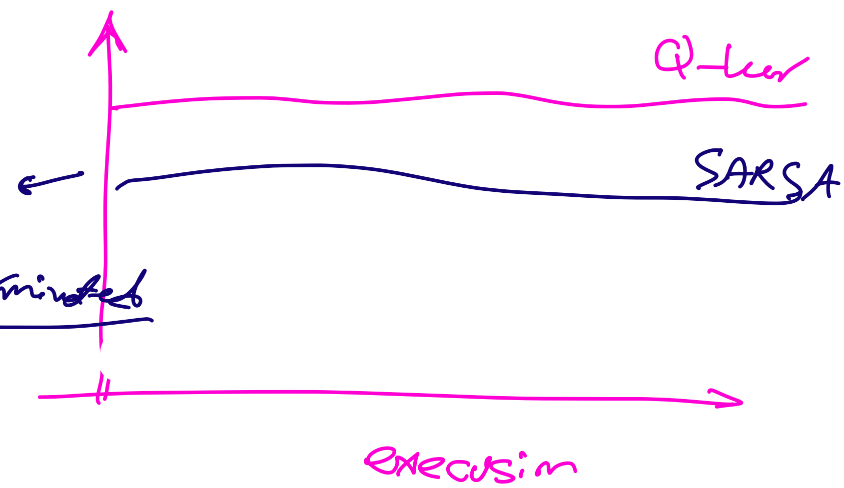






$$Q(3,R) = Q(3,R) + \alpha\left[-1 + \gamma \frac{\text{Q-learn } Q(4,R)\ -170}{\text{SARSA } Q(4,a)} - Q(3,R)\right]$$





ε=0

Training is terminated

Can we have replacement of ϵ-greedy?

Boltzman policy = Softmax policy

$$\pi^{\epsilon\text{-}g}(s_t) = \begin{cases} \arg\max_a Q(s_t, a) & 1-\epsilon \\ \text{Random} & \epsilon \end{cases}$$

$$\pi^{Bolt}(a \mid s_t) = \frac{e^{\frac{Q(s_t, a)}{\tau}}}{\sum_{a' \in A} e^{\frac{Q(s_t, a')}{\tau}}}$$: Differentiability

$a \in \{a^1, a^2\}$

$Q(s_t, a^1) = 10 \longrightarrow \pi(a^1 \mid s_t) = \dfrac{e^{10}}{e^{10} + e^{100}} = 0.001$

$Q(s_t, a^2) = 100 \longrightarrow \pi(a^2 \mid s_t) = \dfrac{e^{100}}{e^{100} + e^{10}} = 0.999$

Consider stochastic & Reward is Stochastic

Q-Learning
- Works very well is high stochasticity
- " " large state spaces
- → Batch Data (ε → as you want)

Issues:
- Overestimation of Q- Values (Double Q-Learning)
- Risky in training

SARSA:
- Conservative (both training & Execution)
- Coverge Fast

Issues →
- Performs poorly in high rewards with high variance
- on-policy → can not arbitrary ε X

Expected SARSA:

(Robust but slower)

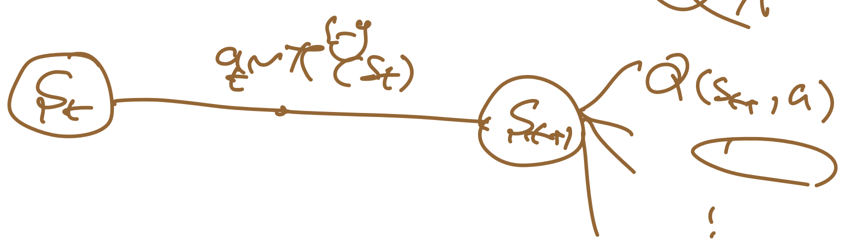
$$\begin{cases} Q(4, R) = 170 & 0.925 \\ Q(4, D) = -920 & 0.025 \\ Q(4, L) = -1600 & 0.025 \\ Q(4, U) = 60 & 0.025 \end{cases}$$

$$Q(S_t, a_t) = Q(S_t, a_t) + \alpha \left[R_{t+1} + \gamma \underbrace{\qquad\qquad}_{0.925\, \underbrace{Q(4,R)}_{170} + 0.025\, Q(4,D) + 0.025\, Q(4,L) + 0.025\, Q(4,U)} - Q(S_t, a_t) \right]$$

$$Q(S_t, a_t) = Q(S_t, a_t) + \alpha \left[R_{t+1} + \gamma \sum_{a \in A} \pi^{\varepsilon\text{-}g}(a \mid S_{t+1})\, Q(S_{t+1}, a) - Q(S_t, a_t) \right]$$

on-policy $\longrightarrow$ you can not set ε arbitrary

$Q^{\pi^{\varepsilon\text{-}g}}$



$\frac{\varepsilon}{4}$

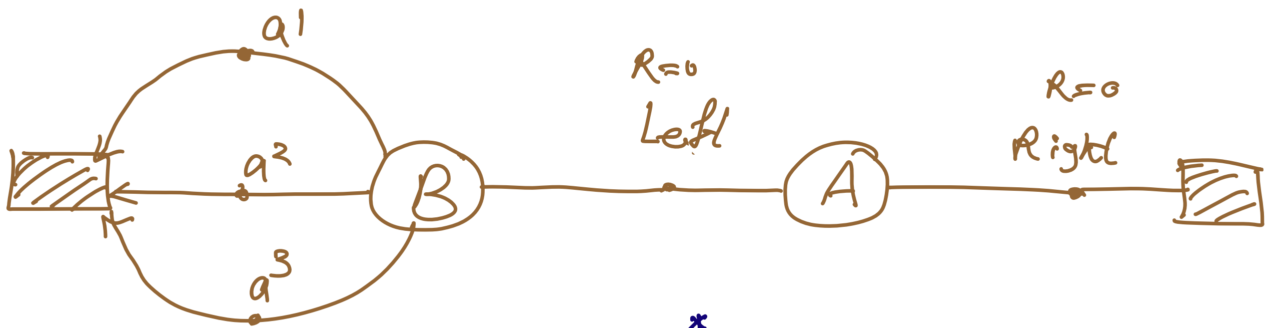
$1 - \varepsilon + \frac{\varepsilon}{4}$

$\frac{\varepsilon}{4}$

$\frac{\varepsilon}{4}$

Double Q-Learning

(overestimation of Q-Values)



$R(B,a^1) \sim \mathcal{N}(0, 1)$

$R(B, a^2) \sim \mathcal{N}(-0.2, 1)$

$R(B,a^3) \sim \mathcal{N}(0.3, 1)$

$Q^*(A,R) = 0$

$\pi^*(A) = L$

$\pi^*(B) = a^3$

$Q^*(A, L) = 0.3$

Q-Learning

episode 1

Random state: B $\xrightarrow[R \sim \mathcal{N}(-0.2,\, 1) = 0.8]{a^2 \sim \pi^{\varepsilon\text{-}g}(B) = \begin{cases} 0.33 & a^1 \\ 0.33 & a^2 \\ 0.33 & a^3 \end{cases}}$ T

$$Q(B, a^2) = \underbrace{Q(B, a^2)}_{0} + \alpha\left[0.8 + \gamma \underbrace{\max Q(T, a)}_{0} - \underbrace{Q(B, a^2)}_{0}\right] = 0.4$$

$$Q(B, a^2) = E[\overset{\mathcal{N}(-0.2, 1)}{R_{t+1}} + \underbrace{\gamma R_{t+2} + \cdots}_{0} \mid S_t = B, a_t = a^2] = -0.2$$

Episode 2

Random State: $A \xrightarrow[R=0]{L \sim \pi^{\varepsilon\text{-}g}(A)} B$ $\quad \pi^{\varepsilon\text{-}g}(A) = \begin{cases} 0.5 & L \\ 0.5 & R \end{cases}$

$$Q(A,L) = Q(A,L) + \alpha\left[\underbrace{R}_{0} + \gamma \underbrace{\max Q(B,a)}_{0.4} - Q(A,L)\right]$$

$$= 0.2$$

$a^2 = \arg\max Q(B,a)$

$B \xrightarrow[R \sim N(-0.2,\,1)]{\pi^{\varepsilon\text{-}g}(B)} T$ $\quad \pi^{\varepsilon\text{-}g}(B) = \begin{cases} \frac{\varepsilon}{3} & a^1 \\ 1-\varepsilon+\frac{\varepsilon}{3} & a^2 \\ \frac{\varepsilon}{3} & a^3 \end{cases}$

R: 0

$$Q(B,a^2) = \underbrace{Q(B,a^2)}_{0.4} + \alpha\left[\underbrace{R}_{0} + \gamma \underbrace{\max Q(B,a)}_{0.4} - \underbrace{Q(B,a^2)}_{0.4}\right] = 0.4$$

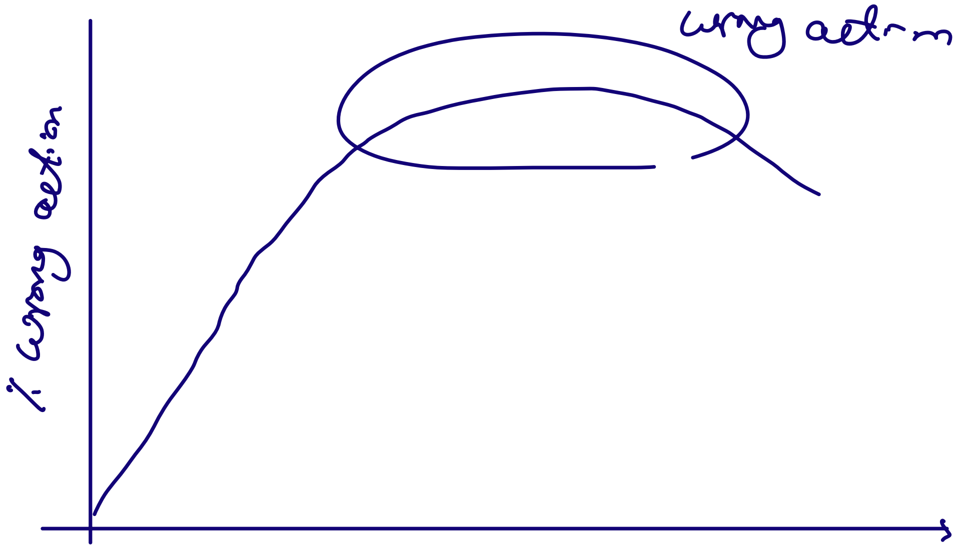
$R^{a1} \sim \mathcal{N}(-0.5, 5)$

$R^{a2} \sim \mathcal{N}(-0.5, 5)$

R^{a1}	μ^{a1} (Q^{a1})	R^{a2}	μ^{a2}	$\max_a \mu^a$
3	3	-8	-8	3
-2	0.5	-5	-6.5	0.5
4	1.67	-8	-7	1.67
⋮	⋮	⋮	⋮	⋮
	-0.5		-0.5	1.72
				$E[\max_a \mu]$

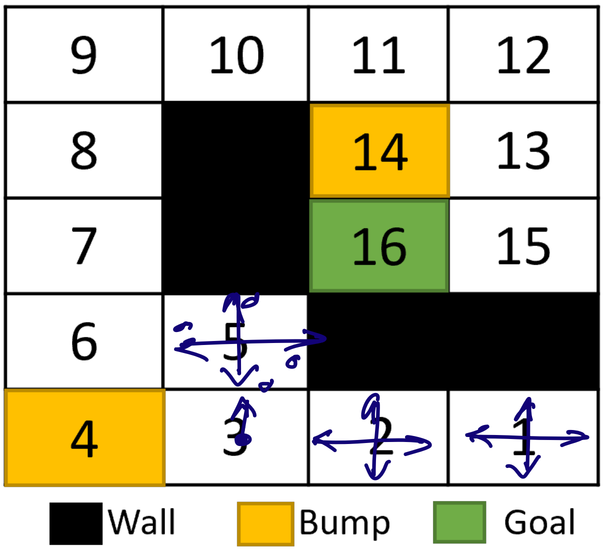
$$Q = Q + \alpha\left[R + \gamma \underbrace{\max_a Q(s', a)}_{\mu} - Q\right]$$

Q-Learning: $Q_\pi(s,a) = E_\pi\left[R + \gamma \max_{a'} Q(s', a') \mid s_t = s, a_t = a\right]$

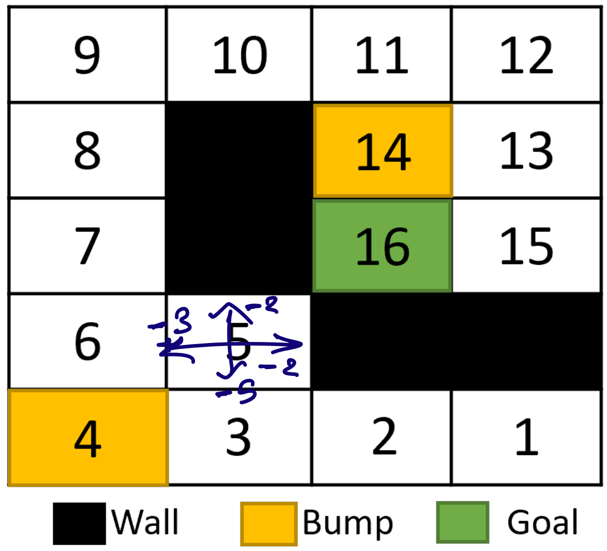


% wrong action
wrong action

Q_1



Q_2



$$3 \qquad a \sim \pi^{\{\theta\}}(3) = \begin{cases} \arg\max_{a \in A} Q_1(3,a) + Q_2(3,a) & 1-\varepsilon \quad \sim U \\ \text{Random} & \varepsilon \end{cases}$$

w.p. 0.5

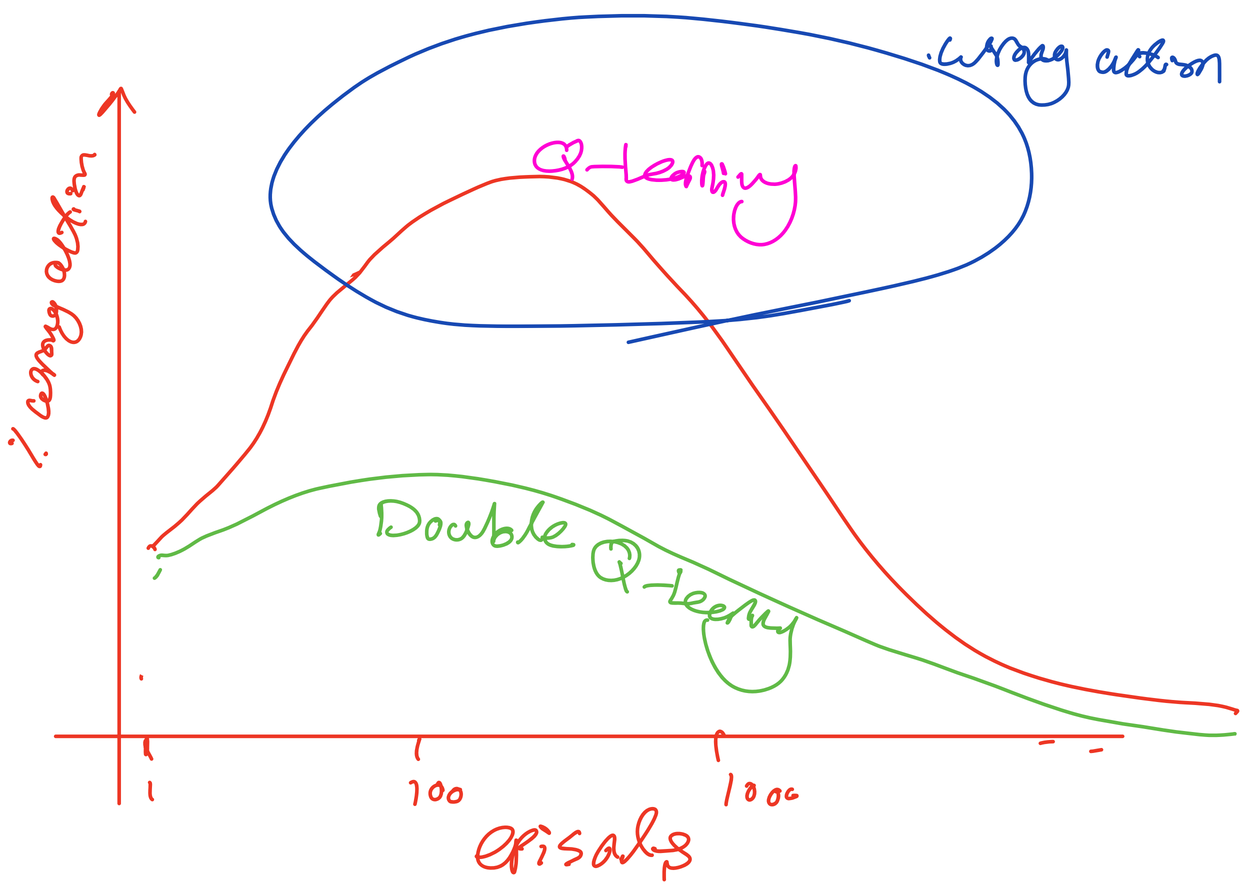
$$Q_1(3,U) = Q_1(3,U) + \alpha\left[R + \gamma Q_2\left(5, \arg\max_{a \in A} Q_1(5,a)\right) - Q_1(3,U)\right]$$

w.p. 0.5

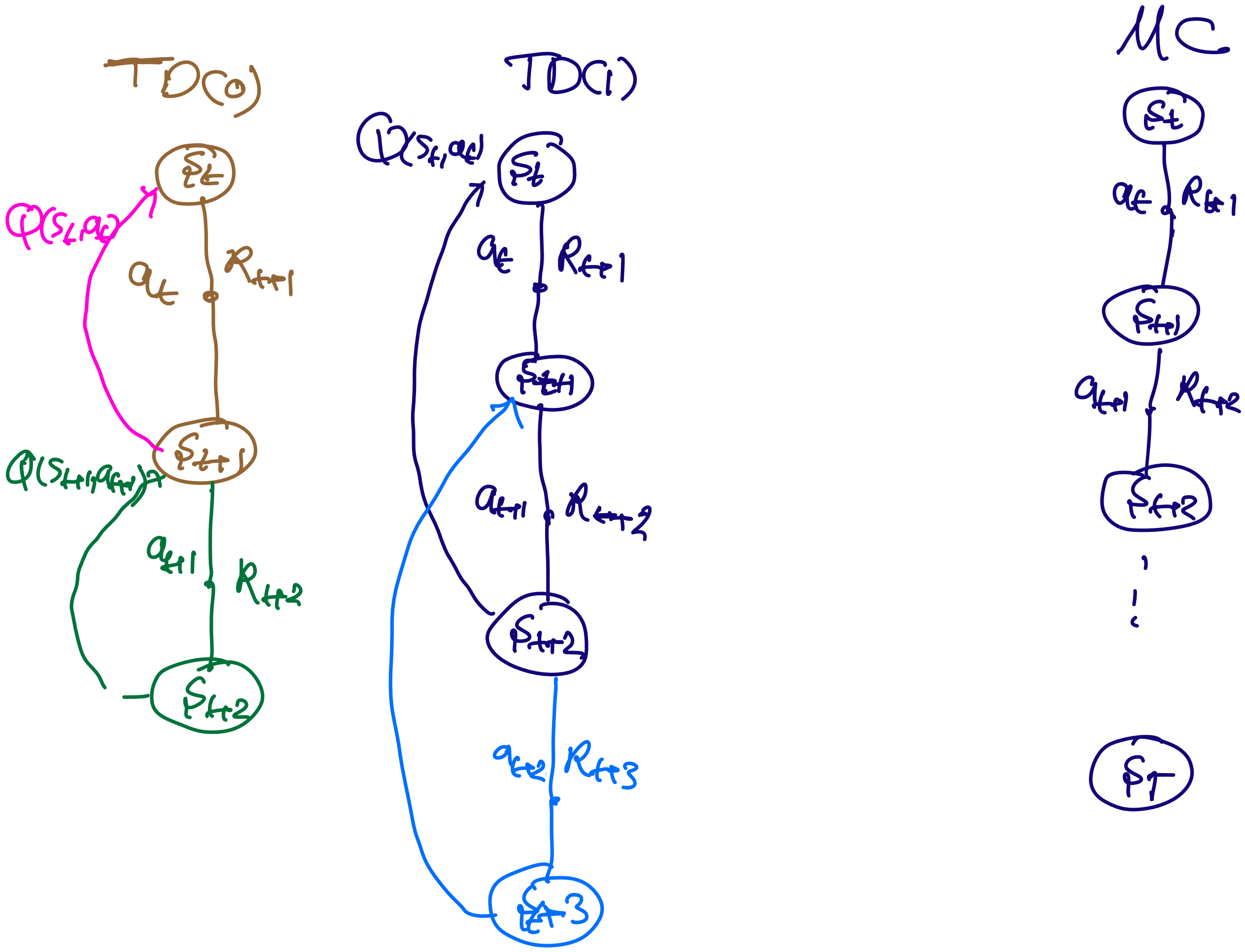
$$Q_2(3,U) = Q_2(3,U) + \alpha\left[R + \gamma Q_1\left(5, \arg\max Q_2(5,a)\right) - Q_2(3,U)\right]$$

Double Q-Learning

Initialize $Q_1(s,a)$ and $Q_2(s,a), \forall s \in \mathcal{S}, a \in \mathcal{A}(s)$, arbitrarily
Initialize $Q_1(\textit{terminal-state}, \cdot) = Q_2(\textit{terminal-state}, \cdot) = 0$
Repeat (for each episode):
    Initialize S
    Repeat (for each step of episode):
        Choose A from S using policy derived from Q_1 and Q_2 (e.g., ε-greedy in $Q_1 + Q_2$)
        Take action A, observe R, S'
        With 0.5 probabilility:
            $Q_1(S,A) \leftarrow Q_1(S,A) + \alpha\Big(R + \gamma Q_2(S', \arg\max_a Q_1(S',a)) - Q_1(S,A)\Big)$
        else:
            $Q_2(S,A) \leftarrow Q_2(S,A) + \alpha\Big(R + \gamma Q_1(S', \arg\max_a Q_2(S',a)) - Q_2(S,A)\Big)$
        $S \leftarrow S'$;
    until S is terminal



Multi-step Bootstrapping



TD(0):

$$Q(S_t,a_t) = Q(S_t,a_t) + \alpha\left[R_{t+1} + \gamma \max Q(S_{t+1},a) - Q(S_t,a_t)\right]$$

TD(1)

$$Q(S_t,a_t) = Q(S_t,a_t) + \alpha\left[R_{t+1} + \gamma R_{t+2} + \gamma^2 \max_a Q(S_{t+2},a) - Q(S_t,a_t)\right]$$

TD(n) larger variance ⟸ Variance

TD(n) less Biased ⟸ Biased
max $n>0$