CS143: Artificial Intelligence

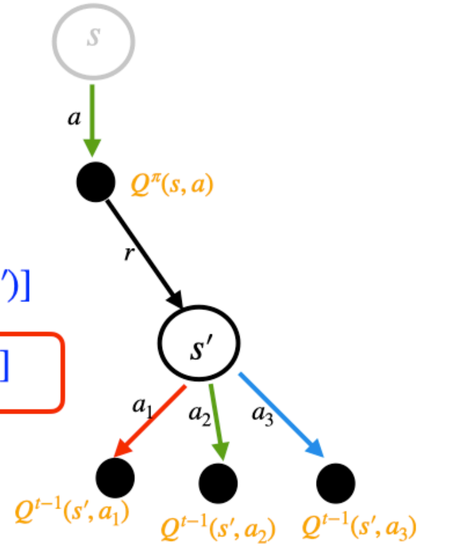
Q-Learning

$$Q^{\pi}(s,a) \leftarrow (1-\alpha)Q^{\pi}(s,a) + \alpha\ sample_i$$

$$\leftarrow Q^{\pi}(s,a) - \alpha Q^{\pi}(s,a) + \alpha\ sample_i$$

$$\leftarrow Q^{\pi}(s,a) - \alpha Q^{\pi}(s,a) + \alpha[R(s,\pi(s),s') + \gamma \max_{a'} Q^{\pi}(s',a')]$$

$$\leftarrow Q^{\pi}(s,a) + \alpha[R(s,\pi(s),s') + \gamma \max_{a'} Q^{\pi}(s',a') - Q^{\pi}(s,a)]$$



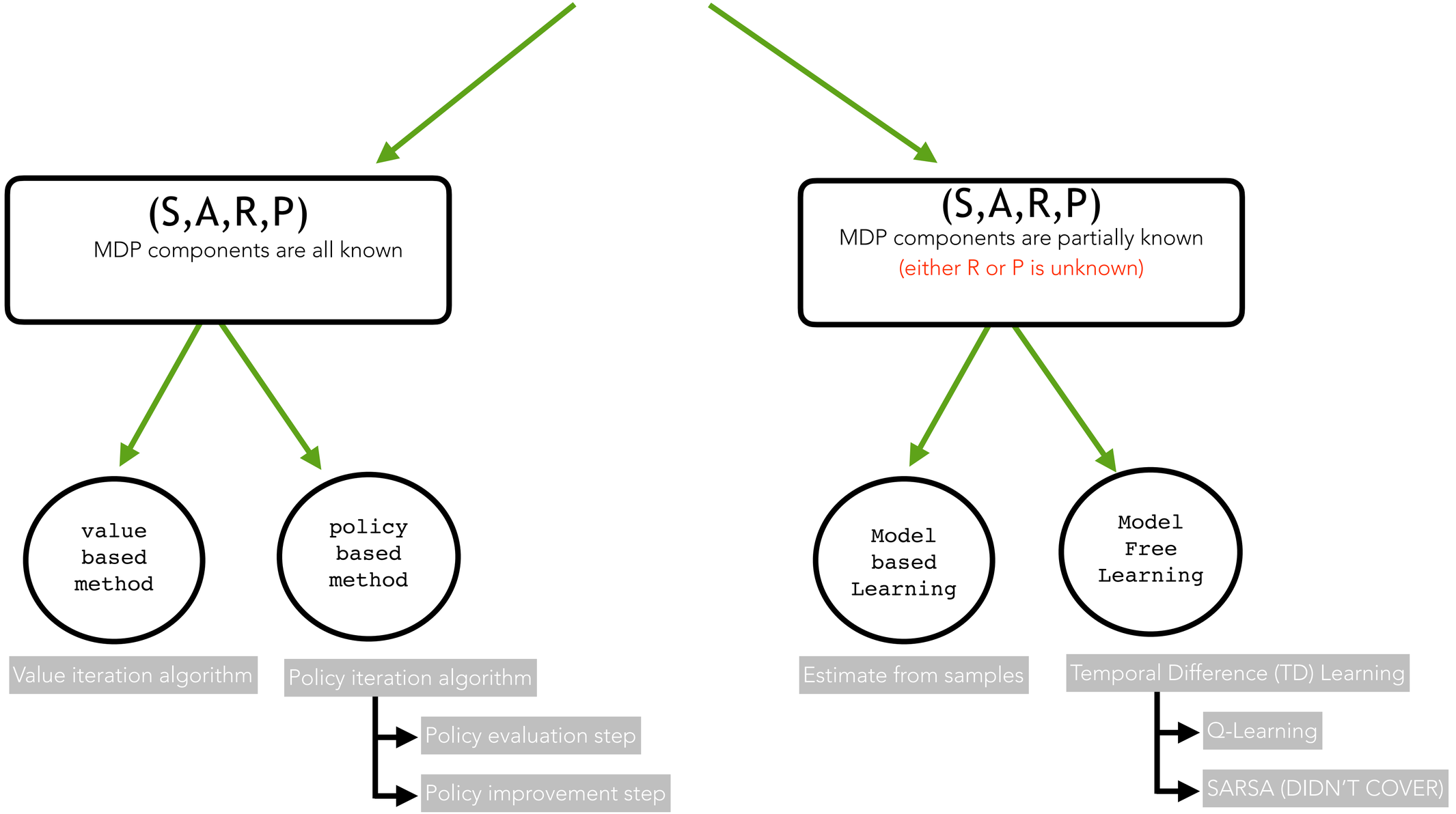
α = *learning rate which can vary between 0 to 1.0*

$sample_i = R(s,\pi(s),s') + \gamma \max_{a'} Q^{\pi}(s',a')$

Q-Learning (implementation)

Drake UNIVERSITY

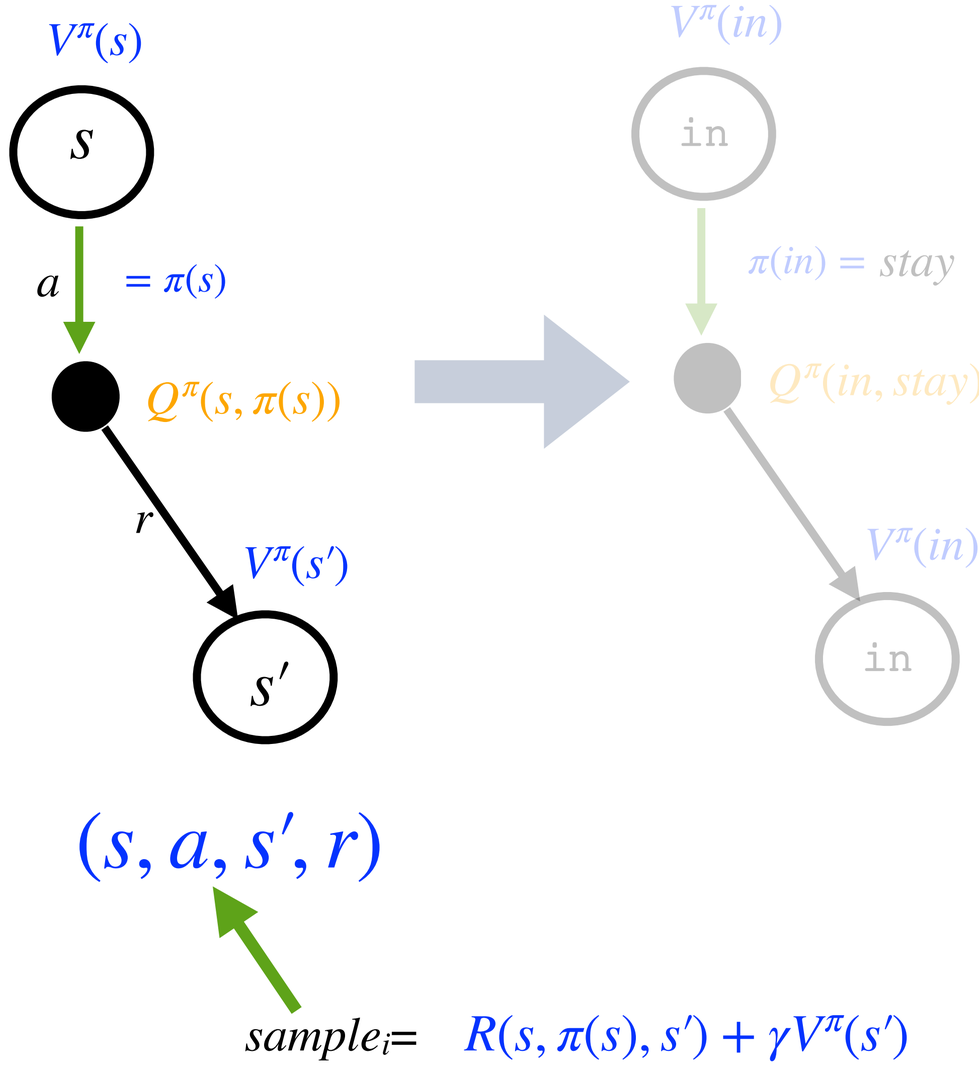
Reinforcement Learning Methods



Model-Free Approach: Generate Samples

- Components of an MDP is a tuple (S,A,R,P):
  - Either of the two components are unknown:
    - Reward function: unknown $R(s, a, s')$ *or*
    - Transition function: unknown $P(s' \mid a, s)$

- Let us interact with the environment by acting upon it. It will generate samples of the form
  - (s, a, s', r)
  - This sample is shown in the image on the right. Notice that there is only one transition from the Q(,) node because it represents a single sample. We do not have probabilities for transitions into multiple branches.



Model-Free Approach: Average for Value Function from Samples

- Calculate the average of the value function from the samples as follows:

Episode#1

$sample_1$ in, stay, in, $4

$sample_2$ in, stay, in, $4

$sample_3$ in, stay, end, $4

Episode#2

$sample_4$ in, stay, end, $4

Episode#3

$sample_5$ in, stay, in, $4

$sample_6$ in, stay, end, $4

Episode#4

$sample_7$ in, stay, in, $4

$sample_8$ in, stay, in, $4

$sample_9$ in, stay, in, $4

$sample_{10}$ in, stay, end, $4

Episode#5

$sample_{11}$ in, stay, in, $4

$sample_{12}$ in, stay, end, $4

Episode#6

$sample_{13}$ in, quit, end, $10

$$\text{sample}_1 = R(in, stay, in) + \gamma V_j^\pi(in)$$
$$\text{sample}_2 = R(in, stay, in) + \gamma V_j^\pi(in)$$
$$\text{sample}_3 = R(in, stay, end) + \gamma V_j^\pi(end)$$
$$\text{sample}_4 = R(in, stay, end) + \gamma V_j^\pi(end)$$
$$\text{sample}_5 = R(in, stay, in) + \gamma V_j^\pi(in)$$
$$\text{sample}_6 = R(in, stay, end) + \gamma V_j^\pi(end)$$
$$\text{sample}_7 = R(in, stay, in) + \gamma V_j^\pi(in)$$
$$\text{sample}_8 = R(in, stay, in) + \gamma V_j^\pi(in)$$
$$\text{sample}_9 = R(in, stay, in) + \gamma V_j^\pi(in)$$
$$\text{sample}_{10} = R(in, stay, end) + \gamma V_j^\pi(end)$$
$$\text{sample}_{11} = R(in, stay, in) + \gamma V_j^\pi(in)$$
$$\text{sample}_{12} = R(in, stay, end) + \gamma V_j^\pi(end)$$
$$\text{sample}_{13} = R(in, quit, end) + \gamma V_j^\pi(end)$$

$$V_{j+1}^\pi(in) \leftarrow \frac{1}{13} \sum_i sample_i$$

Model-Free Approach: Average for Value Function from Samples

- Calculate the average of the value function from the samples as follows:

Episode#1

$sample_1$ in, stay, in, $4

$sample_2$ in, stay, in, $4

$sample_3$ in, stay, end, $4

Episode#2

$sample_4$ in, stay, end, $4

Episode#3

$sample_5$ in, stay, in, $4

$sample_6$ in, stay, end, $4

Episode#4

$sample_7$ in, stay, in, $4

$sample_8$ in, stay, in, $4

$sample_9$ in, stay, in, $4

$sample_{10}$ in, stay, end, $4

Episode#5

$sample_{11}$ in, stay, in, $4

$sample_{12}$ in, stay, end, $4

Episode#6

$sample_{13}$ in, quit, end, $10

$$\text{sample}_1 = 4 + \gamma V_j^{\pi}(in)$$
$$\text{sample}_2 = 4 + \gamma V_j^{\pi}(in)$$
$$\text{sample}_3 = 4 + \gamma V_j^{\pi}(end)$$
$$\text{sample}_4 = 4 + \gamma V_j^{\pi}(end)$$
$$\text{sample}_5 = 4 + \gamma V_j^{\pi}(in)$$
$$\text{sample}_6 = 4 + \gamma V_j^{\pi}(end)$$
$$\text{sample}_7 = 4 + \gamma V_j^{\pi}(in)$$
$$\text{sample}_8 = 4 + \gamma V_j^{\pi}(in)$$
$$\text{sample}_9 = 4 + \gamma V_j^{\pi}(in)$$
$$\text{sample}_{10} = 4 + \gamma V_j^{\pi}(end)$$
$$\text{sample}_{11} = 4 + \gamma V_j^{\pi}(in)$$
$$\text{sample}_{12} = 4 + \gamma V_j^{\pi}(end)$$
$$\text{sample}_{13} = 10 + \gamma V_j^{\pi}(end)$$

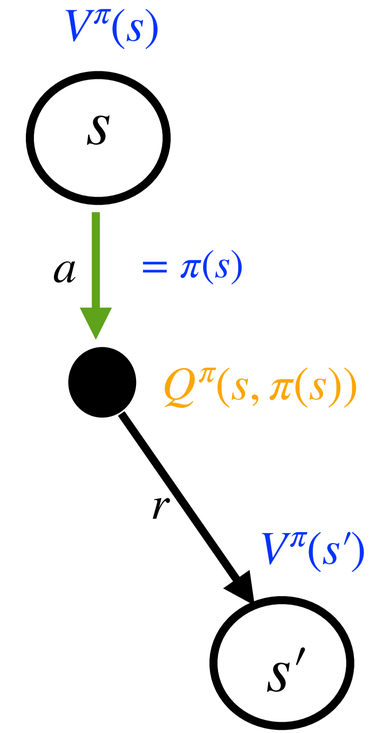
Replacing the reward function with their values

$$V_{j+1}^{\pi}(in) \leftarrow \frac{1}{13} \sum_i sample_i$$

A Model-Free Approach: Temporal Difference (TD) Learning

- Let us interact with the environment by acting upon it. It will generate samples of the form
  - (s, a, s', r)
  - We will use this sample to update value function $V^{\pi}(s)$
  - Like Policy Evaluation Algorithm, we are given a fixed policy, we will update the values from the samples until convergence
  - Rather than directly computing average (as shown in slides before), we will maintain a running average of value function $V^{\pi}(s)$ as follows:

$$V^{\pi}(s) \leftarrow (1-\alpha)V^{\pi}(s) + \alpha \ sample_i$$



$sample_i = R(s, \pi(s), s') + \gamma V^{\pi}(s')$

α = *learning rate which can vary between 0 to 1.0*

Temporal Difference (TD) Learning

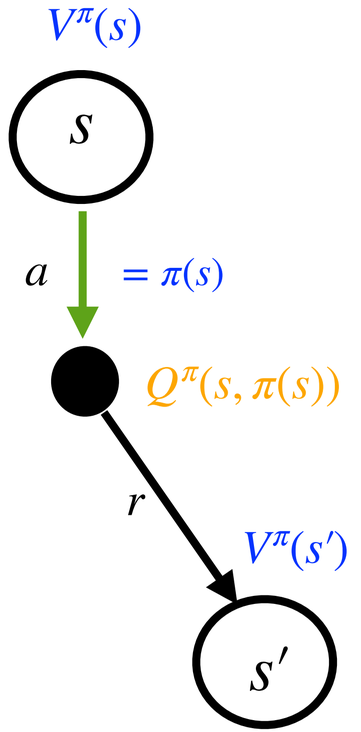
- Let us interact with the environment by acting upon it. It will generate samples of the form

$$V^{\pi}(s) \leftarrow (1-\alpha)V^{\pi}(s) + \alpha \; sample_i$$

$$\leftarrow (1-\alpha)V^{\pi}(s) + \alpha[R(s,\pi(s),s') + \gamma V^{\pi}(s')]$$

$$\leftarrow V^{\pi}(s) - \alpha V^{\pi}(s) + \alpha[R(s,\pi(s),s') + \gamma V^{\pi}(s')]$$

$$\leftarrow V^{\pi}(s) + \alpha[R(s,\pi(s),s') + \gamma V^{\pi}(s') - V^{\pi}(s)]$$



α = *learning rate which can vary between 0 to 1.0*

$sample_i = R(s,\pi(s),s') + \gamma V^{\pi}(s')$

Q-Learning

Answer: A Specific Temporal Difference Learning Algorithm (falls under **model-free** approaches)

Recall: Value Iteration Algorithm

- Turning recursive Bellman equations into update equations:

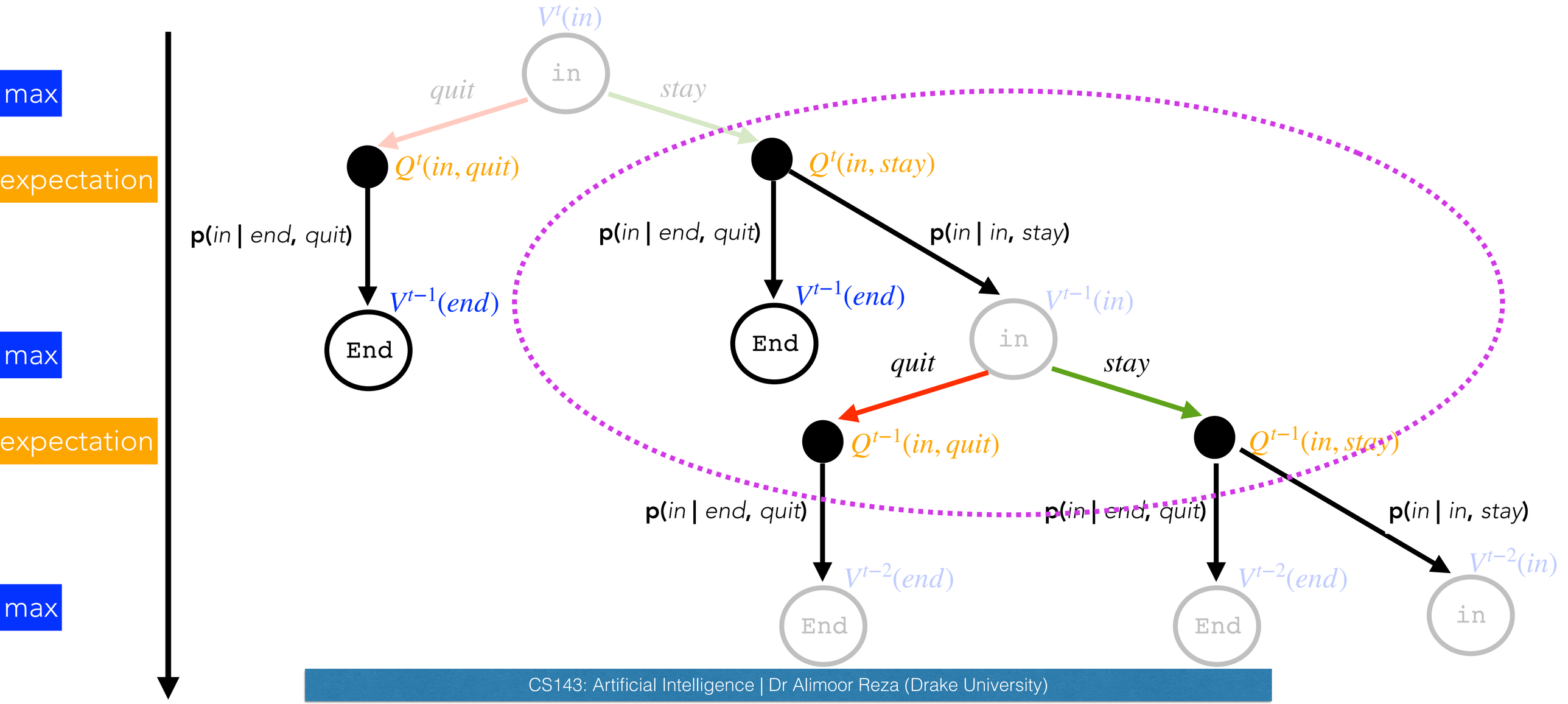
$$V^0(s) = 0$$

$$V^t(s) \leftarrow \max_{a \in actions(s)} \sum_{s'} p(s' \mid s, a)[r(s, a, s') + \gamma V^{t-1}(s')]$$

value function in time step ***t*** can be computed via retrieving values from its previous iteration ***(t-1)***

- Instead of calculating value functions, let's **directly calculate q-value function**

Infer the relationship between Q-nodes



Recall: Value Iteration Algorithm

- Instead of calculating value functions, let's directly calculate q-value function

$$Q^0(s,a) = 0$$

$$Q^t(s,a) \leftarrow \sum_{s'} p(s'|s,a)[r(s,a,s') + \gamma \max_{a' \in actions(s)} Q^{t-1}(s',a')]$$

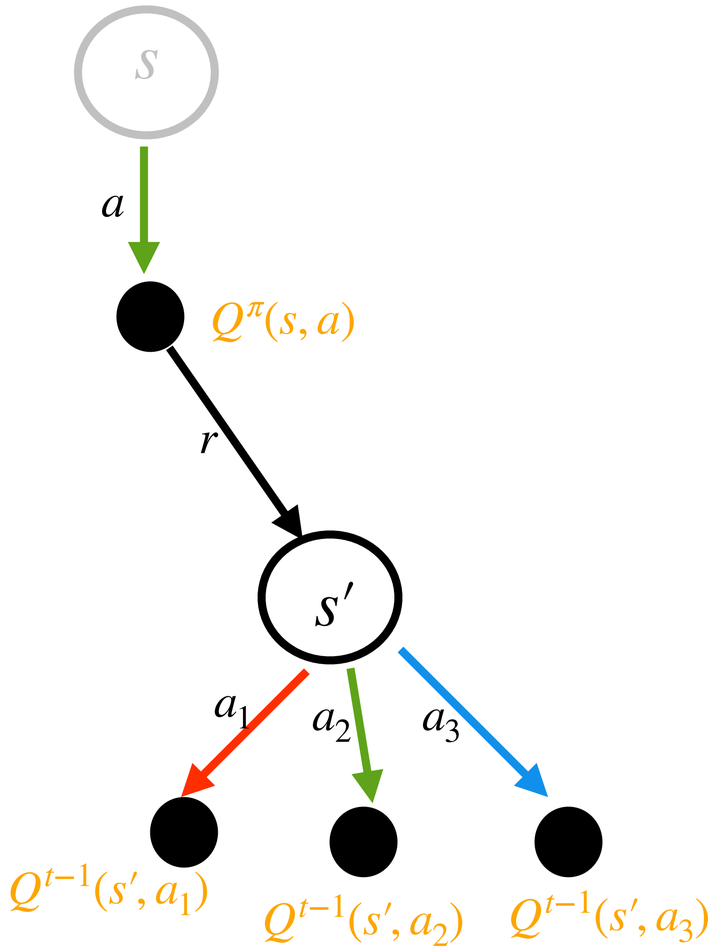
q-value in time step ***t*** can be computed via retrieving values from its previous iteration ***(t-1)***

Q-Learning

- In Q-learning, you estimate the Q(s,a) function using samples
  - (s, a, s', r)
  - We will use this sample to update value function $Q^{\pi}(s, a)$
  - Like TD learning, we will maintain a running average of q-value function $Q^{\pi}(s, a)$ as follows:

$$Q^{\pi}(s, a) \leftarrow (1 - \alpha)Q^{\pi}(s, a) + \alpha \; sample_i$$

Let's expand and shuffle around the terms of the above Equation



α = *learning rate which can vary between 0 to 1.0*

$$sample_i = R(s, \pi(s), s') + \gamma \max_{a'} Q^{\pi}(s', a')$$

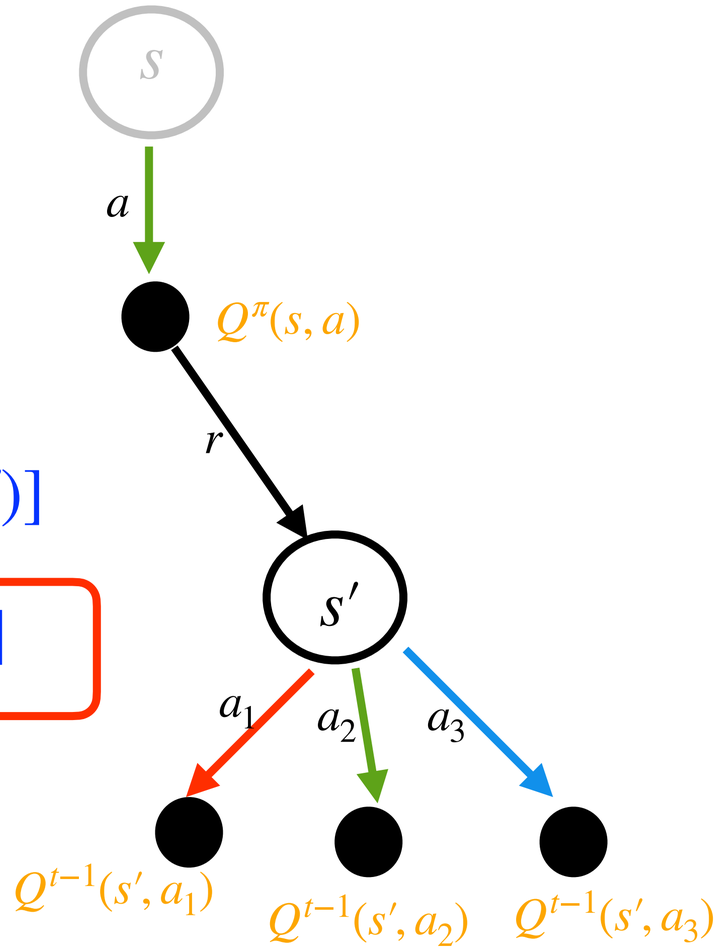
Q-Learning

$$Q^{\pi}(s,a) \leftarrow (1-\alpha)Q^{\pi}(s,a) + \alpha\ sample_i$$

$$\leftarrow Q^{\pi}(s,a) - \alpha Q^{\pi}(s,a) + \alpha\ sample_i$$

$$\leftarrow Q^{\pi}(s,a) - \alpha Q^{\pi}(s,a) + \alpha[R(s,\pi(s),s') + \gamma \max_{a'} Q^{\pi}(s',a')]$$

$$\leftarrow Q^{\pi}(s,a) + \alpha[R(s,\pi(s),s') + \gamma \max_{a'} Q^{\pi}(s',a') - Q^{\pi}(s,a)]$$



α = *learning rate which can vary between 0 to 1.0*

$sample_i = R(s,\pi(s),s') + \gamma \max_{a'} Q^{\pi}(s',a')$

Q-Learning: Implementation

Check out the link below to see the implementation of Q-learning on FrozenLake

https://github.com/alimoorreza/CS143-sp26-notes/blob/main/RL_q_learning_frozen_lake.ipynb