Course 2, Module 3
Temporal Difference Learning Methods for Control

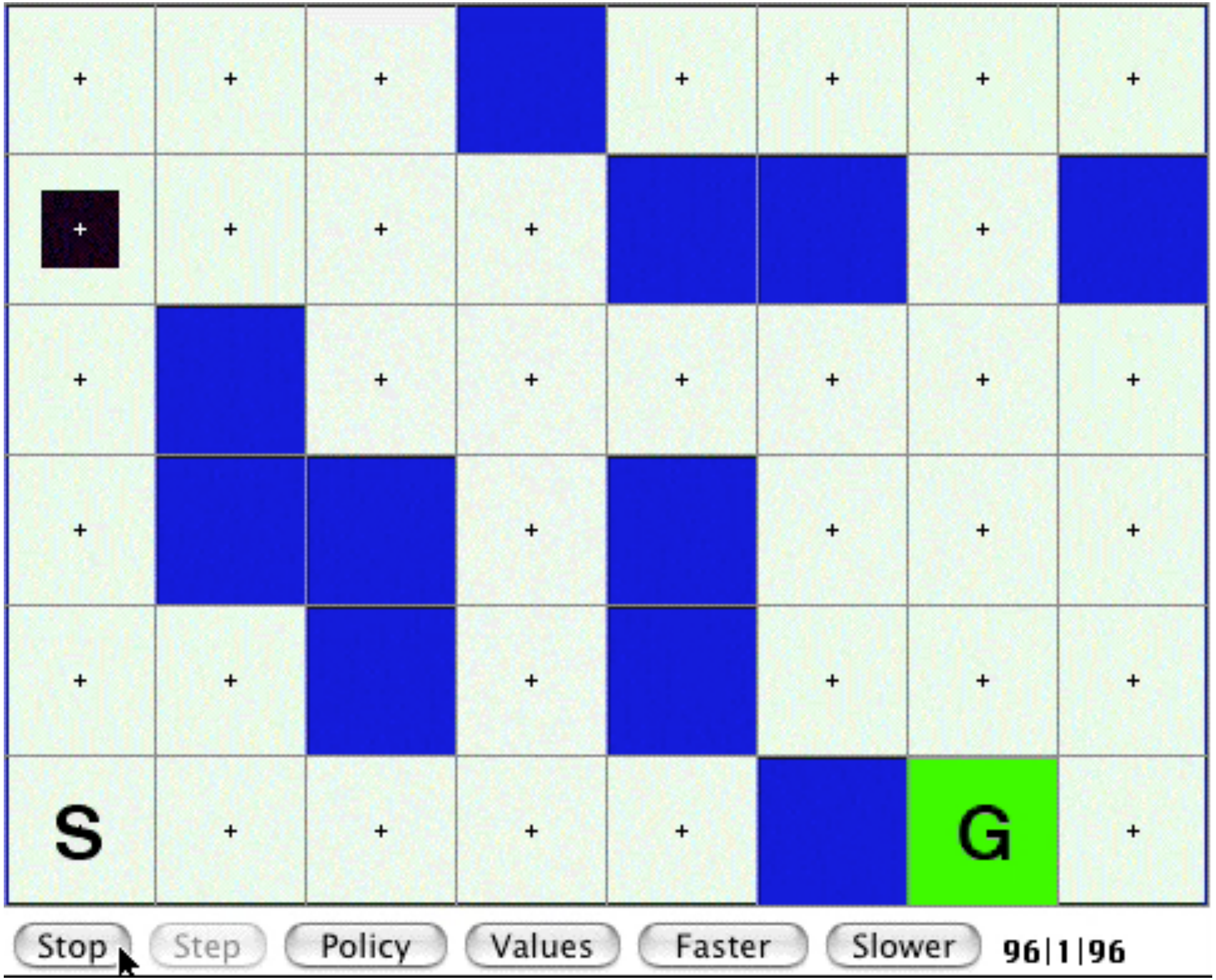
CMPUT 365
Fall 2021

Comments

- **Mini-essay due Oct 20th**
  - See previous lecture for advice on writing
  - Sample essay in the google sheet
  - Any questions about the mini essay
- Any questions?

Review of Course 2, Module 3
TD Control

GridWorld Example



Video 1: Sarsa: GPI with TD

- Building an algorithm to find *near optimal* policies: SARSA (**S**tate, **A**ction, **R**eward, Next **S**tate, **A**ction). Combining the ideas of *policy evaluation, policy improvement, TD,* and *epsilon-soft policies*

- Goals:

  - explain how **generalized policy iteration** can be used with TD to find improved policies

  - Describe the Sarsa Control algorithm

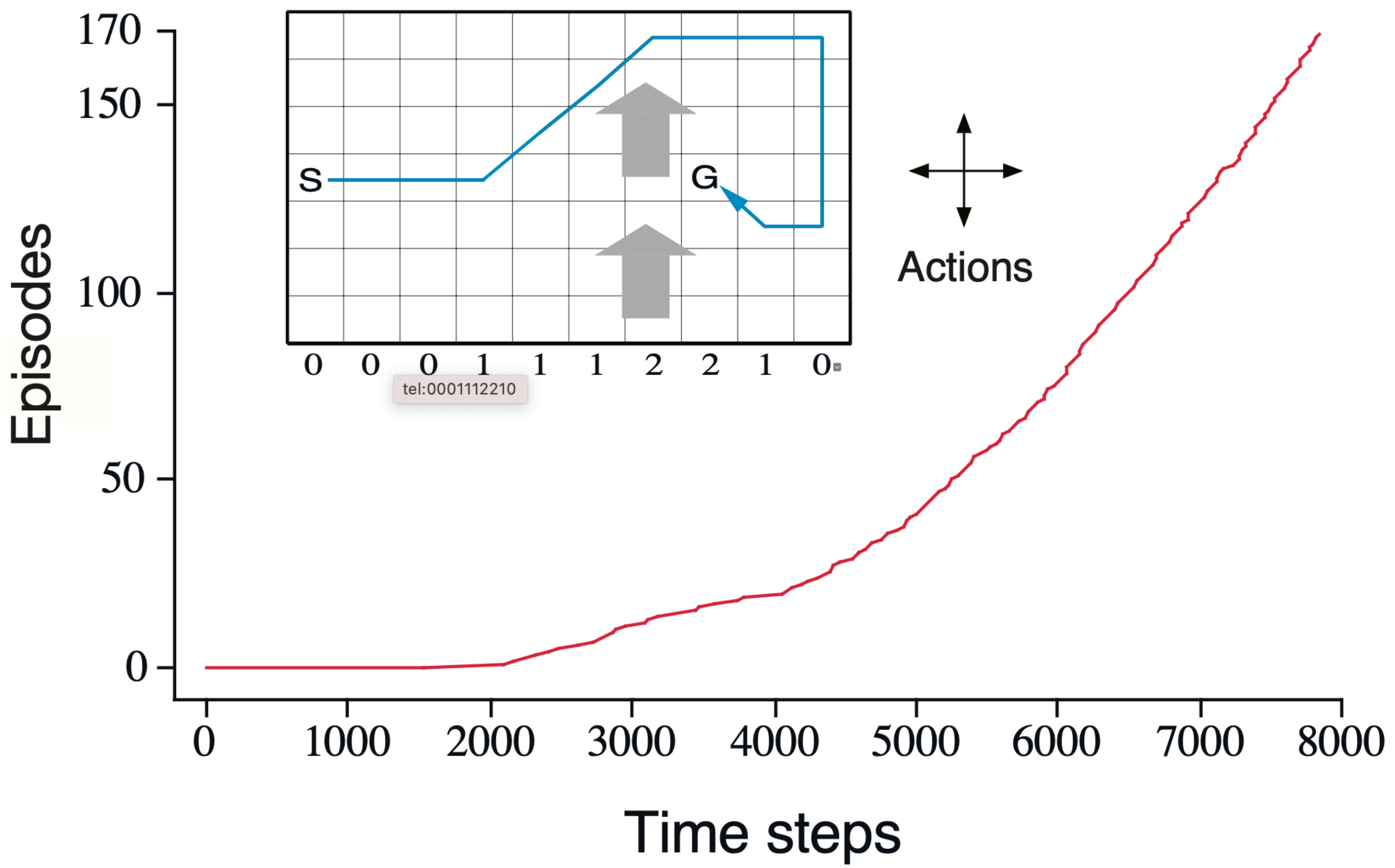
- *Is Sarsa an on-policy control algorithm or an off-policy control algorithm?*

Video 2: Sarsa in the Windy Grid World

- We ran a fun **experiment with Sarsa** on a fancy gridworld
- Goals:
  - Understand how the Sarsa control algorithm operates in an example MDP.
    - the Windy Gridworld
  - Gain experience analyzing the performance of a learning algorithm.
    - **New type of plot:** understanding the plot of cumulative episodes completed vs steps

Plotting learning

- *How can we tell that the agent is learning and getting better?*



- *What would the plot look like if we plotted steps (y-axis) per episode (x-axis)?*

Video 3: What is Q-learning

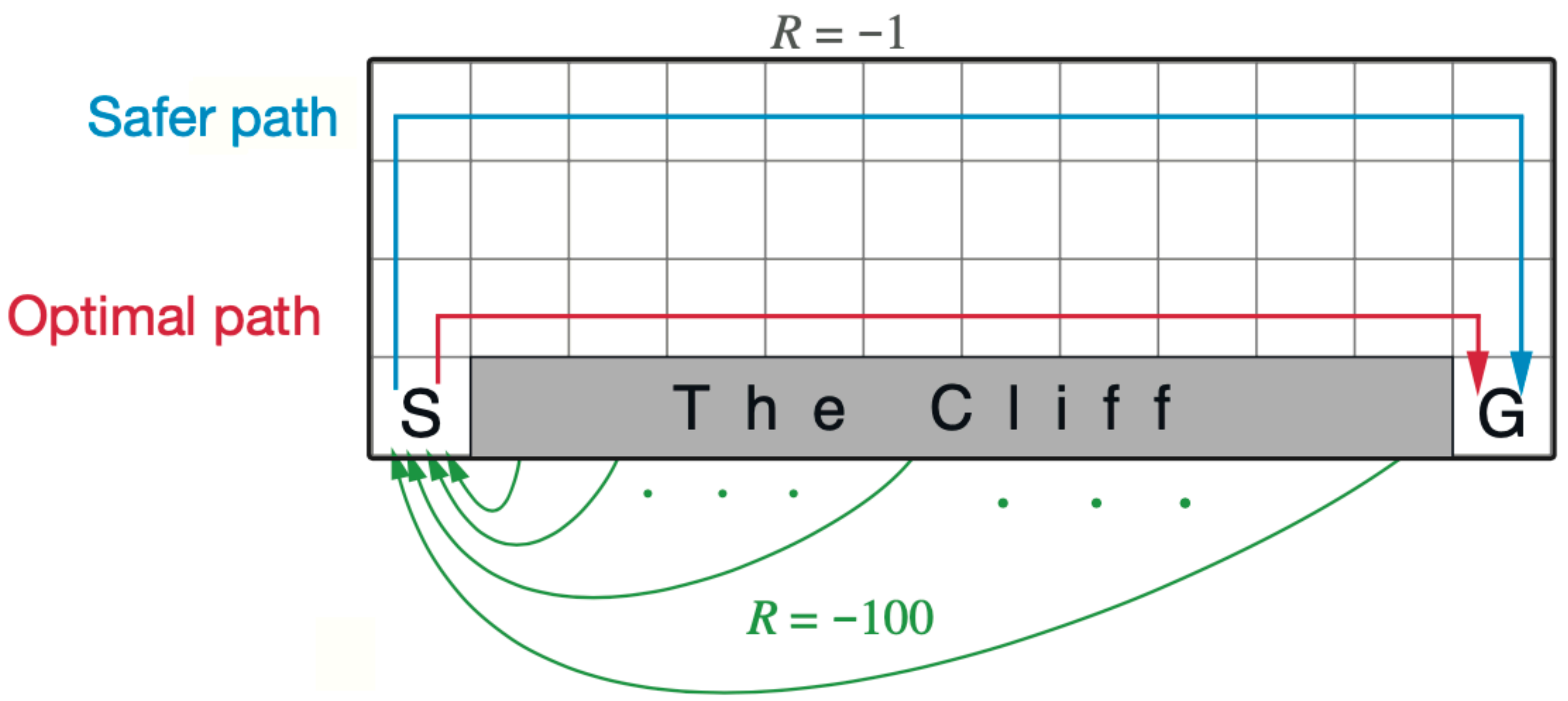
- Just the most famous RL algorithm! Similar to SARSA, but learns the **optimal policy**
- Goals:
  - Describe the Q-learning algorithm
  - Explain the relationship between Q-learning and the Bellman optimality equations
- *How does Q-learning handle exploration?*

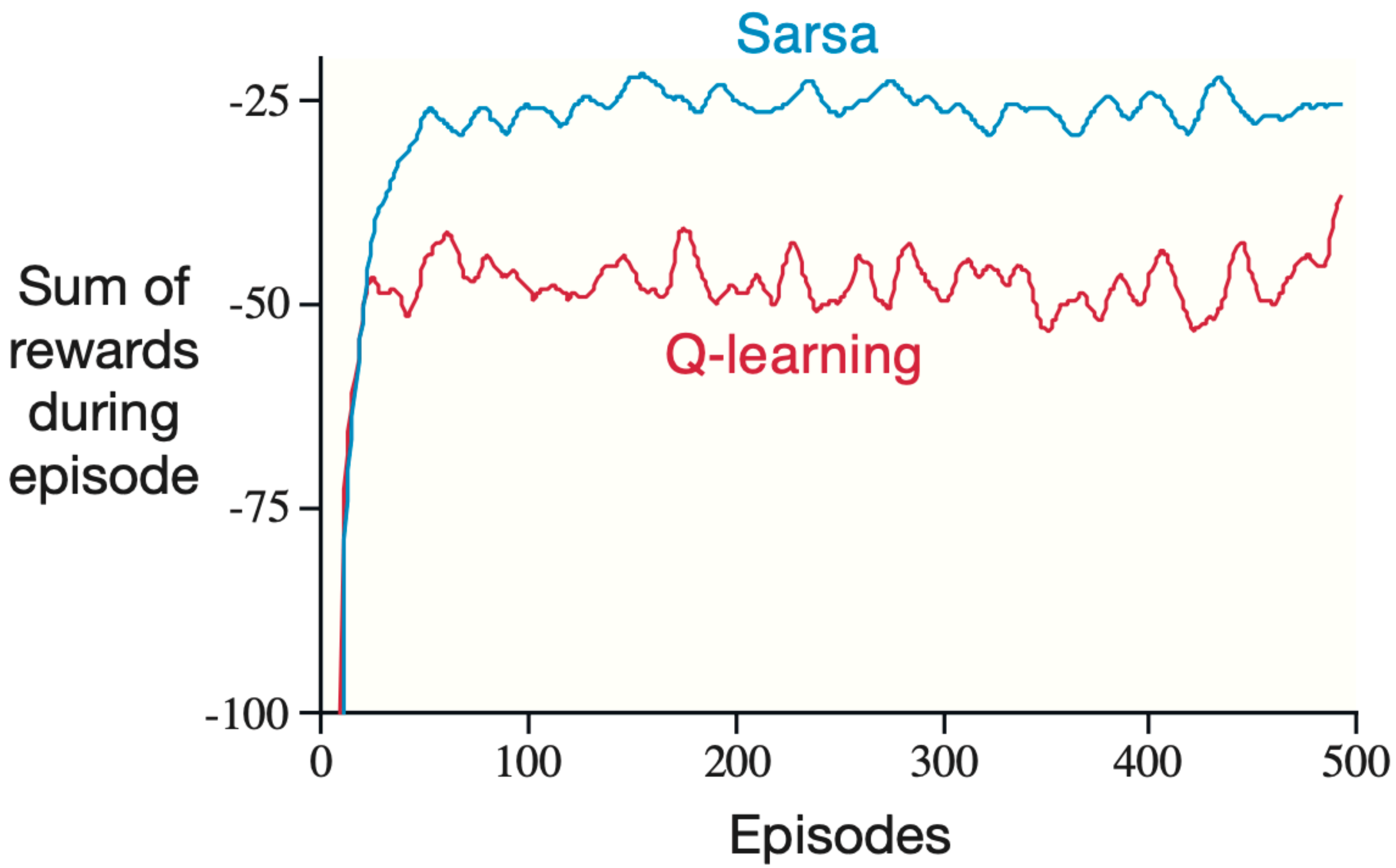
Video 4: Q-learning in the Windy Gridworld

- How does Q-learning work in practice? We get some insight with an **experiment comparing with SARSA**
- Goals:
  - Gain insight into how Q-learning performs in an example MDP
  - Gain insight into the **differences between Q-learning and Sarsa**.

Plotting learning

- *Why is Q-learning worse here?*





Video 5: How is Q-learning Off-policy?

- Q-learning **learns about** the greedy policy (which eventually becomes π^*), while **following a different policy** ε-greedy. That is off-policy, but there are no importance sampling corrections!

- Goals:

  - Understand how Q-learning can be off-policy **without using importance sampling**

  - Describe how learning on-policy or off-policy might affect performance in control.

    - SARSA (on-policy learning), can be better!

- *What is the target policy and the behavior policy in Q-learning?*

Video 6: Expected SARSA

- **A new TD Control method**! Uses the probability of each action under the current policy in its update!
- Goals:
  - explain the Expected Sarsa algorithm.
- $Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma \sum_a \pi(a \mid S_{t+1}) Q(S_{t+1}, a) - Q(S_t, A_t)]$
- *How would we compute this* $\sum_a \pi(a \mid S_{t+1}) Q(S_{t+1}, a)$ *for an epsilon-greedy policy?*

Video 7: Expected SARSA in the Cliff World

- Why all the fuss about Expected Sarsa? We find out with an **experiment** in another gridworld: The cliff world. Spoiler: **Expected Sarsa learns faster** AND **is more robust to our choice of alpha**

- Goals:

  - Describe Expected Sarsa's behaviour in an example MDP.

  - And Empirically compare Expected Sarsa and Sarsa

- *What part of the E-sarsa update accounts for the improvement in performance and why?*

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha[R_{t+1} + \gamma \sum_a \pi(a \mid S_{t+1}) Q(S_{t+1}, a) - Q(S_t, A_t)]$$

Video 8: The Generality of Expected SARSA

- Expected SARSA is pretty neat! It can perform better than either SARSA or Q-learning. In addition, the algorithm can be **used in different ways**
- Goals:
  - Understand how Expected Sarsa can do **off-policy** learning without using importance sampling
  - Explain how Expected Sarsa **generalizes Q-learning**
- *If the target policy is greedy what does Expected Sarsa become?*

Terminology Review

- TD methods we have learned about are **tabular, one-step, model-free** learning algorithms
- **Tabular:** we store the value function in a table. One entry in the table per value, so each value is stored independently of the others. We are implicitly assuming the state-space ($\mathcal{S}$) is small
- **One-step**: we update a single state or state-action value on each time-step. Only the value of Q(S,A) from S -- A --->S',R. We never update more than one value per learning step
- **Model-free**: we don't assume access to or make use of a model of the world. All learning is driven by sample experience. Data generated by the agent interacting with the environment

Clarification: Prediction vs Control

- “How is SARSA and Q-learning different than the previous TD methods we learned last week?”
- “Can you explain how learning from state-value is different from action-value and why we look at action-value learning in sarsa and q-learning instead of state-value? Is one better than the other?”
- “For off-policy methods like Q-Learning and Expected Sarsa, does these algorithms use the behaviour policy b anywhere?”

Clarification: online vs offline perf

- “It seems like Expected sarsa is better than sarsa which is better than q-learning (when measuring performance online). How do we know which method to pick when looking at different situations? Is E-Sarsa always the best?”
  - In this course we **always evaluate the agents online**—we measure the reward they get while exploring and learning. **All rewards count**
  - In **offline evaluation** we only measure rewards during **special test episodes** where learning and exploration is disabled (e.g., alpha = 0, epsilon = 0)

Clarification: Convergence

- “Why are fixed epsilon values used for greedifying TD methods when it seems like, in general, they benefit from using epsilon values that vary over time such as epsilon=1/t

- “How do we know if we have performed a sufficient number of episode iterations to obtain the optimal action-value function for Sarsa and Expected Sarsa? Is there a specific condition that will be met when they have converged?”

- “I wonder asymptotic or interim performance is more important in the real wold? I think asymptotic performance is more important, but if the asymptotic performance are close to each other, will the interim performance be a reason to choose a worse asymptotic performance?”

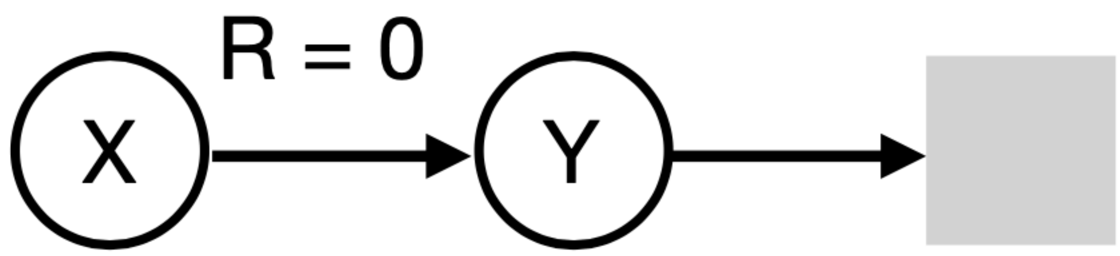
Quiz Review

Worksheet Question

- Why is Sarsa considered on-policy, but Expected Sarsa can be used off-policy?

Worksheet solution

7. Assume still that $V = v_\pi = 0$. What is the expectation and the variance of the TD update from state X? What is the expectation and the variance of the Monte-carlo update from state X?



$$P(R = r|Y) = \begin{cases} 0.5 & \text{if } r = -1000 \\ 0.5 & \text{if } r = +1000 \end{cases}$$

7. **TD update:** The expectation of the TD update is

$$\mathbb{E}[\delta|X] = \mathbb{E}[0 + \gamma v_\pi(Y) - v_\pi(X)|X] = 0,$$

since each $v_\pi(X) = v_\pi(Y) = 0$.
The variance of the TD update is

$$\begin{aligned} \mathbb{V}[\delta|X] &= \mathbb{E}[\delta^2|X] - \mathbb{E}[\delta|X]^2 \\ &= \mathbb{E}[(0 + \gamma v_\pi(Y) - v_\pi(X))^2|X] \\ &= 0. \end{aligned}$$

MC update: The expectation of the MC update is

$$\mathbb{E}[G - v_\pi(X)|X] = 0.5 \times 1000 + 0.5 \times (-1000) = 0.$$

Similarly, the variance is

$$\begin{aligned} \mathbb{V}[G - v_\pi(X)|X] &= \mathbb{E}[(G - v_\pi(X))^2|X] - \mathbb{E}[G - v_\pi(X)|X]^2 \\ &= \mathbb{E}[(G - v_\pi(X))^2|X] \\ &= 0.5 \times 1000^2 + 0.5 \times (-1000)^2 \\ &= 1000^2. \end{aligned}$$