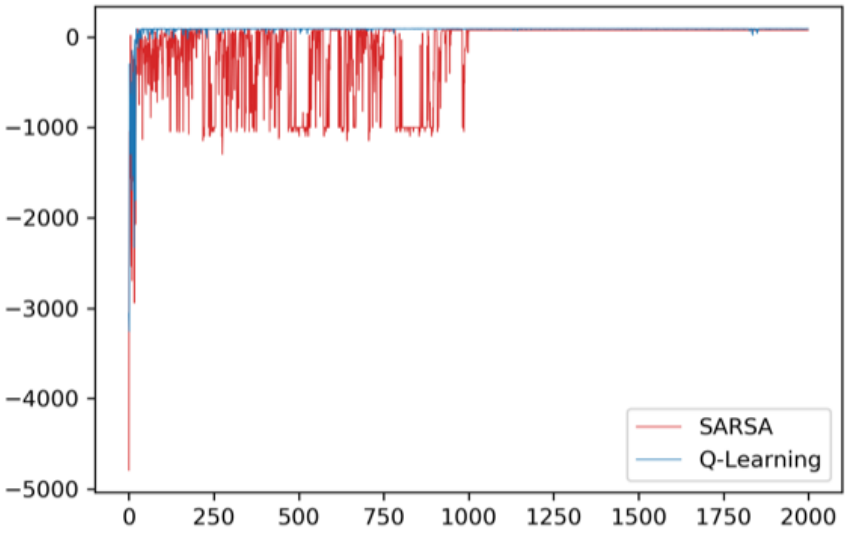
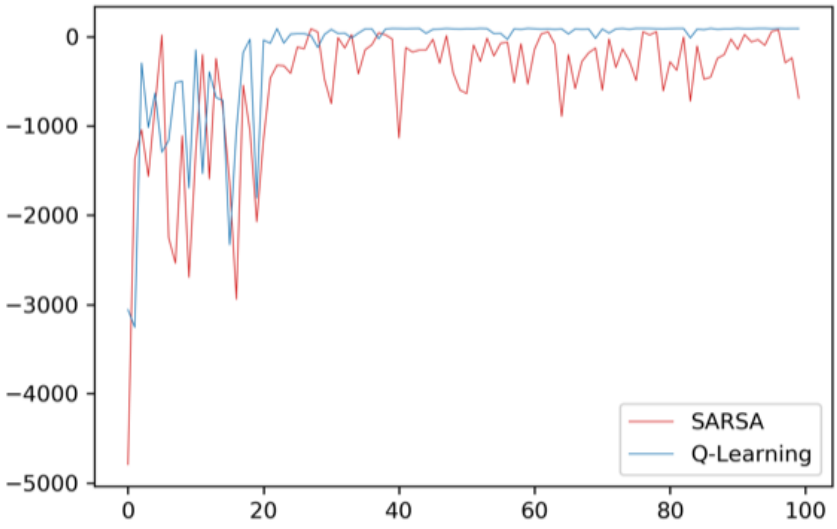
Overall, Q-learning goes through the icy and faster path (occasionally slipping because of ice then going back up or down). SARSA goes safer paths to avoid the holes with the longer duration trade-off. Here is the plot that compares reward values between Q-learning and SARSA after 2000 episodes.



The red line shows SARSA's reward values. The blue line shows Q-learning's reward values. We can see that the blue line is always above the red line. That means Q-learning earns higher reward values than SARSA after initial periods.

This is because for Q-learning, the next action a' is chosen to maximize the next state's Q-value instead of following the current policy. By contrast, SARSA learns based on the action performed by the current policy instead of the greedy policy.

Let's zoom in the first 100 periods to have a clearer view. It's obvious that the blue line dominates the red line.



However, in terms of fluctuations, SARSA's reward values fluctuate much more dramatically than Q-learning's reward values within the first 100 periods. This is because at first, SARSA explores a lot of different paths. Its action a and ϵ-greedily are based on the current policy, which can vary a lot between periods. It does not go through ice surface, which eliminates probabilistic paths. Its paths converge to a deterministic one, making rewards values constant after 1000 periods.

For Q-learning, its ϵ may still vary a little in later periods. Therefore, when it comes to the final episodes, Q-learning's reward values still fluctuate, because it may still go through the ice surface.

Another observation is that setting alpha to 1 makes the reward values become unstable. This is because an alpha equal to 1 makes the algorithms update the new solutions all the time. Even with a very small change, their moves are totally switched. The recommendation is that setting alpha smaller but very close to 1 to guarantee stability of process.