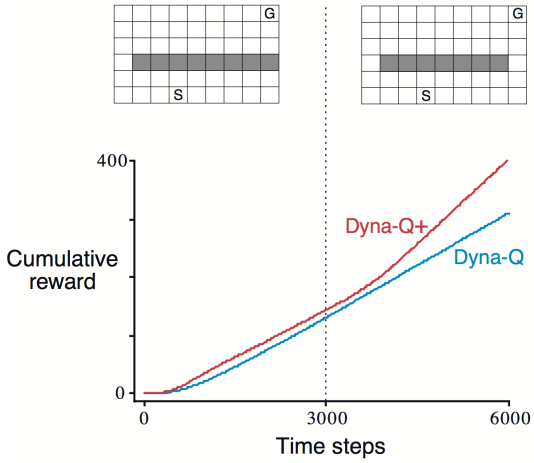
Reinforcement Learning
Exercise 6

Mathias Niepert, Vinh Tong

June 7, 2024

1 Planning and Learning (4P)

a) Why did the Dyna-Q+ (i.e., with exploration bonus) perform better in the first phase as well as in the second phase of the blocking and shortcut experiments (see figure below)? (2P)



b) Consider the tabular Dyna-Q algorithm shown on slide 9. How could you modify this algorithm in order to handle stochastic environments? Would your modification still perform well on changing environments? If not, how could you handle stochastic *and* changing environments? (2P)

2 Monte Carlo Tree Search on the Taxi environment (6P)

The code template can be found on Ilias in *ex06-plan/ex06-plan.py*. It has been tested with gym version 0.18.0 (but should also be stable with version 0.18.3).

We consider the Taxi environment from gym (`https://www.gymlibrary.dev/environments/toy_text/taxi/`). The task of the environment is to pick up a passenger at a specific location and drop him off at another.

In this task, we consider Monte Carlo Tree Search with the following properties:

- **Tree policy:** Use a ϵ-greedy policy for traversing the tree.
- **Expansion:** Expand the tree every time you hit a leaf node.
- **Simulation:** Use a random policy as a rollout policy (already given in the template).
- **Backup:** Update the values in the tree by updating the sum_values and visits of the nodes traversed by the tree policy.

a) Implement Monte Carlo Tree Search in order to solve the task. Keep track of the mean return and plot it over the number of episodes. Report the average reward for solving the problem with MCTS. Is it better than the reward from the plain code template (which does not construct a tree and thus just performs Simple Monte Carlo Search)? (4P)

b) How does the tree evolve? Keep track of the length of the tree and plot the length of the longest path over the number of iterations. (2P)