Q-learning

Nicolas Brissonneau , UTEID:nb24488 , email:nicolasb@utexas.edu

I. Introduction

I am considering the problem of generating an *agent* able to walk across a sidewalk while avoiding *obstacles* and picking up *litter* optimally. To achieve this, we will first setup an environment defining the rules by which the agent should comply and then we will use the q-learning algorithm to generate the optimal behaviors.

II. Method

A. Environment

We have built a sidewalk of dimension 6x25 which we will visualize with a blue box, and we define functions returning the *state* of the agent for a given position on the sidewalk's map. We have thus provided the agent knowledge of his state defined as the left,up,down or right direction in which there is a presence of either nothing, either an obstacle which we desire to avoid or litter which we want to pick up. We will see later on that we can add the information of map limits to the agent. The user-defined amount of obstacles and litter is randomly distributed among the map and when the episode if finished because of a time limit or the final goal being reached, a new randomly generated map is provided.

B. Agent

The agent is defined as a set of states, possible actions and a decision table Q. It receives knowledge of its state from the environment and each state-action pair is associated to a user-defined reward R. Each reward is then used to teach the agent what policy is preferred using a user-defined learning rate α and a discount factor γ through the following equation:

$$Q(s,a) = (1-\alpha)Q(s,a) + \alpha(R + \gamma Q(s', a_{max})) \qquad (1)$$

Where a_{max} corresponds with the action associate to the maximum value of $Q(s',:)$ We have defined different types of rewards, each associated with an expected behavior. Thus, we decided to penalize an action which makes the agent encounter an obstacle, actions which go beyond the sidewalk, actions which do not contribute to the final goal of reaching the other end of the sidewalk, as well as the action which go in the direct opposite direction of the goal which we defined as the right side of the sidewalk. We defined the states as binaries, taking into account the information of the litter and obstacle position. In order to keep exploring the state-action pairs to find the optimal policy, we choose an ε greedy solution.

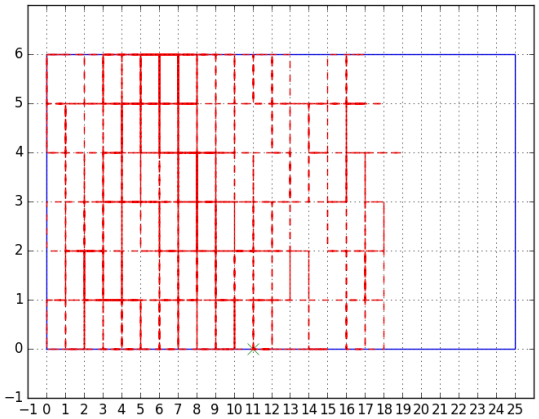

Fig. 1. Staying inside the sidewalk

III. Results

In the following results, the agent will always start on the left-side of the sidewalk at a random height and we used a learning rate $\alpha = 0.05$, a discount factor $\gamma = 0.95$ and we set the limit in the number of actions to 1000 per episode. The path chosen by the agent will be presented as red dashed lines which thickens as it is walked upon, and the agent's final position is shown using a green cross at the end of the dashed lines.
The tasks we will address in this paper are the following are defined as follows:

1) Staying on the sidewalk
2) Going to the right-side of the sidewalk
3) Picking up as much litter as possible along the way
4) Avoiding as many obstacles as possible along the way

First, in order to remain on the sidewalk, we propose 3 different solutions. We have tried defining the surrounding of the sidewalk as a set of obstacles, thus simplifying the problem as the last task, then we have defined a separate penalty gradually increasing as the agent's strays further away from the middle of the path. However, the last solution which is the most robust regarding this performance is to prevent the action of the agent of leaving the path in the first place. A demonstration of this last solution's result is shown in Fig. 1.

The second task being to head toward the right-side of the sidewalk, we simply set a positive reward as the agent decides to go to the right, and a penalty when he decides to do otherwise. The penalty grows stronger as he goes to the

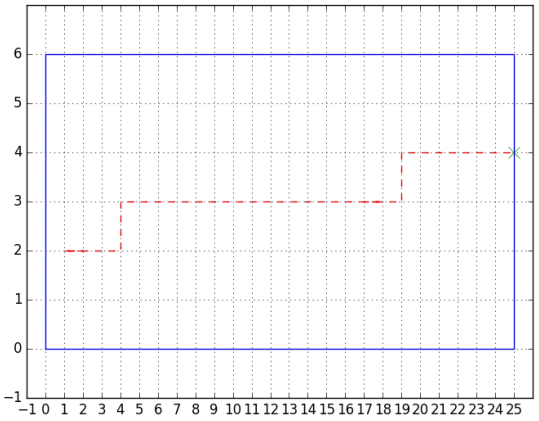

Fig. 2. Walking along the sidewalk on first trial

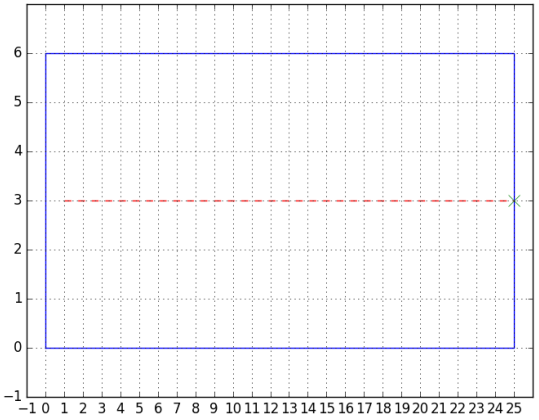

Fig. 3. Walking along the sidewalk on final trial without randomness

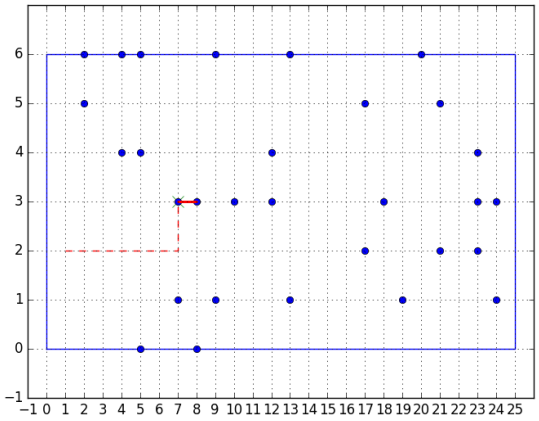

Fig. 4. Litter only - infinite litter reward case 1

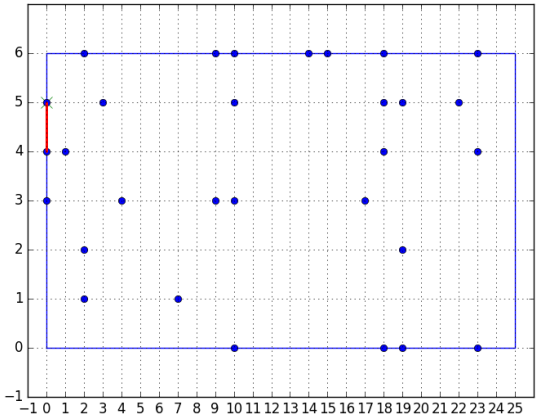

Fig. 5. Litter only - infinite litter reward case 2

left (the opposite direction to the goal) so that he remained encouraged to walk vertically for the following tasks. A demonstration of this last solution's result is shown in Fig. 2 and Fig. 3. As the state remains unchanged, the first time the agent will decide to head right it will keep the same perfect behavior for the remaining part of the testing, thus only one episode is necessary to reach that performance. The sudden changes in path in Fig. 2 are due to the random component of the learning.

The third part consists in picking up as much litter as possible on the sidewalk. We will thus introduce 30 litters in our map which we will try to pick up as optimally as possible during 10,000 episodes. The first intuitive reward to set is a positive one for going toward litter, however it proved to get the agent stuck in place as shown in Fig. 4, Fig. 5 and Fig. 6. In Fig. 4 and Fig. 5 the reward for each litter found is too high and the penalty for not following the path is too small. While adapting the reward/penalty for each state-action pair helps, it does not seem to motivate the agent enough to always end up going toward the right-side of the sidewalk. I have then tried to penalize the agent for staying on the same column for too long as shown in Fig. 6, but it still slowed down the agent considerably.

In Fig. 7 we can see the result when adding a penalty when not heading toward the goal, proportional to the time spent in the current episode. The agent is thus encouraged to shorten each episode and learn smart patterns to get a lot more individual litter than in the previous testing, but we can still notice that he is missing some litter along the path.

Fig. 8 demonstrates an excellent behavior, the modification leading to this result is an amplification of the state-actions values in the Q matrix which led to finish an episode. Indeed, as the $ith-$episode goes on, we store the actions taken and the associated states and if the episode ends properly we increase each Q value chosen for the episode and reciprocally if it fails we decrease it. Inspired from the last assignment on back-propagation, this results in amplifying the importance of the desired behavior to head toward the right, and to attenuate the importance of the

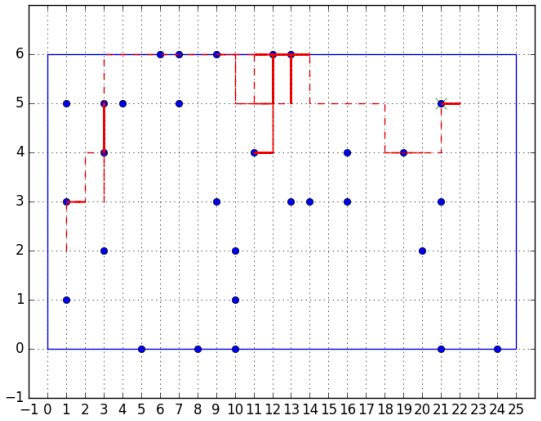

Fig. 6. Litter only - infinite litter reward case 3

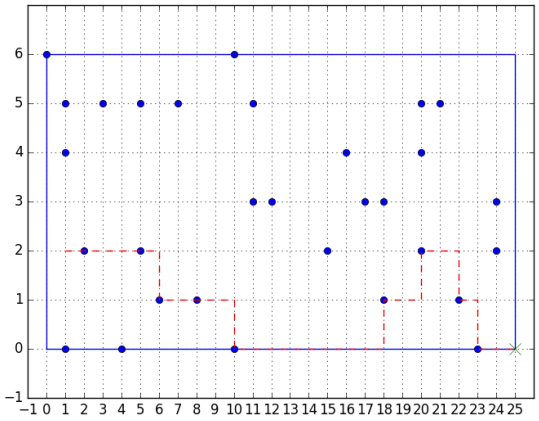

Fig. 8. Litter only - proper litter picking

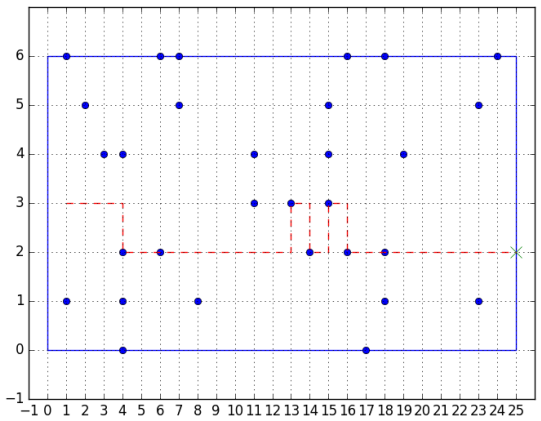

Fig. 7. Litter only - learning patterns

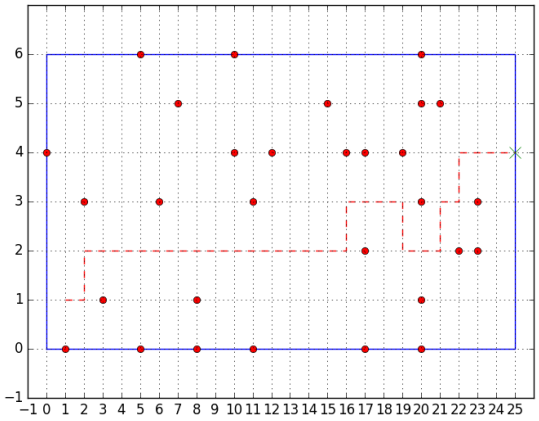

Fig. 9. Obstacles only - proper obstacle avoidance

undesired behavior of long episodes.

The same strategy was used for the last task when trying to avoid hitting obstacles as shown in Fig. 9. However some cases scenarios can prove to be very difficult such as in Fig. 10 where the agent has no other choice but to cross the obstacle at the top of the sidewalk to finally reach the end goal.

Finally, as all tasks are mixed together we can observe a good overall performance in Fig. 11 where the agent compromises between the danger of colliding and the appeal of picking up litter while staying on the sidewalk and heading as fast as possible to the right side.

IV. Conclusion

It took an important amount of time to setup the environment itself but the tuning of the agent's behavior through the reward system was a lot of fun! I have hesitated between applying a lot of different approaches to solve each task, considering the amount of work and the degree of complexity each solution added to the algorithm either by adding new states or new reward systems. I am now very curious as of what would happen if we added more states, as the accuracy should increase but the training size and time would too. Another approach I would like to explore is better tuning inspired from the backpropagation assignment, classify the "moves" which truly were problematic since in my solution I penalize or reward the whole episode for its performance instead of "surgically" improve the policy.

To end this report I would like to thank you for your help during this semester, I learned a lot and have appreciated your support and feedback on my homeworks !

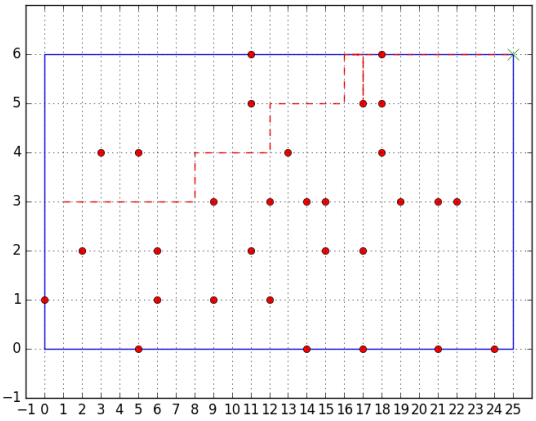

Fig. 10. Obstacles only - impossible win

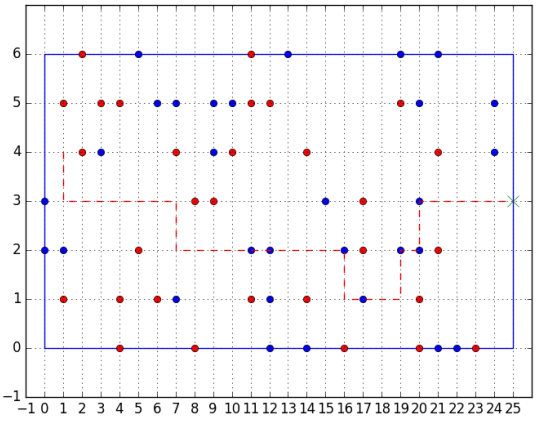

Fig. 11. Final behavior