Homework **Fall 2014**
COMP150 AML - Learning, Planning and Acting in Complex Environments
Department of Computer Science, Tufts University **Roni Khardon**

Homework Assignment 4

In this assignment we continue to work with the RDDL system and simulator, this time implementing and testing the rollout algorithm. The assignment is due by Monday November 10, 1:30pm - in hardcopy (in class) and electronically (via provide).

1 Background

The intention for this assignment is to implement the rollout algorithm and experiment with it in the context of RDDL domains. In order to make your work easier we have slightly modified the simulator and provided you with a template including the function calls implementing the interface required for the rollout algorithm. Several files are provided on the course web page. Please follow the instructions in the README file to distribute these files and compile them so as to be ready to use the new simulator.

2 Policy Rollout

Recall the policy rollout algorithm discussed in class. Here we modify it slightly to simplify the implementation as explained in the following pseudocode. Given a base policy π, the algorithm dynamically decides on the action in the current state s as in the following pseudocode where W is a parameter of the algorithm controlling the number of samples.[1]

1. For $j = 1$ to W
2. Let i be a random legal action available at s.
3. Take one step from s using action i. Let the result be s'.
4. Take a h-length trajectory following π starting at s'
5. Record the return (total discounted reward) from the first step and trajectory.
6. Calculate the average return for the trajectory and update the record for action i
7. Pick i with the maximum average return

In the code above, step (2) can be implemented using a call to the RandomConcurrentPolicy from the previous assignment.

[1] In the original algorithm we take w trajectories for each legal action, so that if there are k legal actions we take $W = k * w$ trajectories in total. In our pseudocode the trajectories may not be divided equally between the legal actions, and if different states have different numbers of legal actions then the effective w is different in each state. But this helps simplify the implementation.

3 Your Task

Implement the policy rollout algorithm and evaluate it as follows.

1. Using instance `elevators_inst_mdp__5` evaluate the performance of the algorithm as a function of W where π is the random policy. In particular, for each $W \in \{10, 20, 50, 100, 150, 200\}$ and $h \in \{5, 10\}$ evaluate the performance when using W, repeating 10 times and calculating the average and standard deviation. Plot the average performance as a function of W and compare it to the performance of the random policy and the round robin policy from the previous assignment. Of course you may also compare to your improved policy from the previous assignment. Overall your plot should have two curves ($h = 5$ and $h = 10$) and two level lines for the baselines.

   Since the simulation is likely to be slow you might consider using instance number 1 when developing your code but please report on the evaluation with instance 5.

2. Repeat the previous item when π is the round robin policy.

3. Now pick $W = 50$ (or another value that makes your code run in moderate amount of time) and $h = 10$ and evaluate with π being the round robin policy on the problem instance from assignment 3.

4. Repeat the evaluation from item 1 but this time changing domain and evaluating on the instance `crossing_traffic_inst_mdp__3`.

To facilitate the experiments you might consider adding command line parameters to the Client program and passing these to your policy. But using compile time constants will be accepted.

4 Extra Credit

Implement and evaluate the recursive multi-stage rollout on some domain and problem.

5 Submitting Your Assignment

- You should submit the following items both electronically and in hardcopy:
  (1) Your code for the assignment which includes the rollout policy code and any additional files needed to run your code (please write clear code and document it as needed). In addition, please include instructions in case we want to compile and run your code.
  (2) A short report with the results from the evaluation and your observations on them. Is the rollout algorithm effective? Does it work as well for small and large problems? How does it compare with the round robin policy? and your improved policy?

- **Please submit a hardcopy** in class.

- **Please submit electronically using provide**: Put all the files from the previous item into a zip or tar archive, for example call it `myfile.zip`. Then submit using
  `provide comp150aml a4 myfile.zip`.