

Markov Decision Processes

[Submit Assignment](#)

Due Nov 22 by 11:59pm **Points** 100 **Submitting** a file upload
File Types txt and pdf **Available** Aug 21 at 8pm - Nov 23 at 12:15am 3 months

Numbers

The assignment is worth 15% of your final grade.

Read everything below carefully!

Why?

In some sense, we have spent the semester thinking about machine learning techniques for various forms of function approximation. It's now time to think about using what we've learned in order to allow an agent of some kind to act in the world more directly. This assignment asks you to consider the application of some of the techniques we've learned from reinforcement learning to make decisions.

The same ground rules apply for programming languages as with the previous assignments.

The Problems Given to You

You are being asked to explore Markov Decision Processes (MDPs):

1. Come up with two interesting MDPs. Explain why they are interesting. They don't need to be overly complicated or directly grounded in a real situation, but it will be worthwhile if your MDPs are inspired by some process you are interested in or are familiar with. It's ok to keep it somewhat simple. For the purposes of this assignment, though, make sure one has a "small" number of states, and the other has a "large" number of states. I'm not going to go into detail about what large is, but 200 is not large. Furthermore, because I like variety no more than one of the MDPs you choose should be a so-called grid world problem.
2. Solve each MDP using value iteration as well as policy iteration. How many iterations does it take to converge? Which one converges faster? Why? How did you choose to define convergence? Do they converge to the same answer? How did the number of states affect things, if at all?
3. Now pick your favorite reinforcement learning algorithm and use it to solve the two MDPs. How does it perform, especially in comparison to the cases above where you knew the model, rewards, and so on? What exploration strategies did you choose? Did some work better than others?

Coding Resources

The algorithms used in this assignment are relatively easy to implement. Existing implementations are easy to find too, below is just one example.

[Brown-UMBC Reinforcement Learning and Planning \(BURLAP\).java code library](http://burlap.cs.brown.edu/)
(<http://burlap.cs.brown.edu/>)

What to Turn In

You must submit:

- a file named *README.txt* that contains instructions for running your code
- your code (link only in the README.txt)
- a file named *yourgtaccount-analysis.pdf* that contains your writeup.

The file *yourgtaccount-analysis.pdf* should contain:

- A description of your MDPs and why they are interesting.
- A discussion of your experiments.

Note: Analysis writeup is limited to 10 pages total.

Grading Criteria

As always you are being graded on your analysis more than anything else.