

Assignment 5
Markov Decision Processes
Experiments and Report
CS 3420 – Spring, 2016
Due: Saturday, 21 May 2016

Experiments

1. How many iterations does Value Iteration need to solve this problem? How many iterations does Policy Iteration need? Use a discount factor of 0.999999, an ϵ of $1e-6$, a key loss probability of 0.5, and the values we used in class for the rewards.
2. Of course, one iteration of Policy Iteration probably took longer than one iteration of Value Iteration. So, now time both your programs to see whether Value Iteration or Policy Iteration is faster for this particular problem. Use the same parameters as in Experiment 1.
3. How many iterations of Value Iteration does it actually take before the policy is optimal? How many iterations of Policy Iteration does it actually take before the policy is optimal? Use the same parameters as in Experiment 1.
4. Using Value Iteration, measure the effect of the following factors on the number of iterations taken. When testing a given parameter, hold the other parameters constant at the values used in the previous experiments.
 - (a) the discount factor
 - (b) the step cost
 - (c) the negative reward
 - (d) the positive reward
 - (e) the key loss probability
5. Using Value Iteration, characterize *and explain* the effect of the following parameters on the resulting policy. Effects considered should include (but not necessarily be limited to) whether an attempt is made to get the key and, if so, what the paths to and from the key (not necessarily the same) look like. Again, when testing a given parameter, hold the other parameters constant at the values used in the previous experiments.
 - (a) the discount factor
 - (b) the step cost
 - (c) the negative reward
 - (d) the positive reward
 - (e) the key loss probability

Be sure to test a broad enough range of values for each parameter to capture all the possibilities. Also, feel free to explore and report on combinations of parameter value changes.

The Report

This time I am requiring that you write a complete paper. A complete paper (for a computer science conference) would be organized, roughly, as follows:

1. **Title**
2. **Abstract:** Basically, a condensed version of the Introduction.
3. **Introduction:** An overview of the work and a roadmap for the rest of the paper.
4. **Related Work:** Work other people have done that is related to the work described in the paper.
5. **Description of Algorithms:** A detailed description of the algorithms used in the work.
6. **Experimental Methodology:** What experiments were run? How many tests? On what kinds of problems? What was measured? Why?
7. **Results:** A description and analysis of the results (see below for more details).
8. **Further Work:** What could be done to push this work farther?
9. **Conclusions:** A summary of the results.

For this assignment, you need to write all of these items, except the **Related Work** section.

Clarity is *very* important. Your report should tell a connected, coherent, and self-contained story. Although this is a scientific paper, you are still telling a story and your reader needs to be able to follow the plot. You should break up the text into the sections outlined in items 3 through 9 above (not including **Related Work** section). You may want to use subsections within your sections to enhance readability.

Don't make the mistake of writing this report for an audience of me. Someone who knows nothing about what you worked on should be able to read your report and understand, at least at a high level, what you did. Assume you're writing it for a CS major who knows nothing about MDPs. Even in actual conference papers and presentations, which are being prepared for an audience of CS researchers, the author(s) will provide a basic explanation of everything, even though some readers or people in the audience will already know that. In particular, this means that your **Description of Algorithms** section should have a thorough explanation of MDPs, Value Iteration, and Policy Iteration, including the equations involved, where appropriate (e.g. the update equation in Value Iteration).

Your data is not self-explanatory. As in the SSAT report, you should use graphs, tables, and/or charts to present your data and support your conclusions. And this section should not be just a collection of captioned graphs, tables, and/or charts whose interpretation is "left as an exercise for the reader." There should be text that explains

your results and the claims in your text should refer to the graphs/table/charts in support of those claims. For example “As the table in Figure 1 shows, the effect of...” All graphs/table/charts should be captioned. Any graphs should have a title, axis labels with units, and a legend if more than one series of values is being shown.

Example Paper

I have posted a recent conference paper of mine on BlackBoard in a folder called `majercik-green-pso`. This is a good example of the type of paper I want you to be patterning your paper on. I don’t expect you to write a paper of this length, and you will notice that I don’t follow the above guidelines on sections. Like you, I had two algorithm descriptions (mine were Standard PSO and GR-PSO), but I made them subsections of a single section, rather than two separate sections. I combined **Experimental Methodology** and **Results**. And I combined **Further Work** and **Conclusions**. Things like this are fine, as long as you cover all the bases. Also, as noted above, I don’t expect you to have a **Related Work** section.

You don’t need to follow the style of my paper exactly. This just happened to be the style required by the particular conference that this paper was in. For example, although the 2-column format is widely used in conferences, some conferences (and all journals) use the standard one “column” format. In case you know Latex and you would like to use this style, I have included the style file in the folder. I have also included the `bib` file. (If you don’t know what that is, you probably won’t need to.)

Important Note on Grading

There was not much code to write and it was pretty straightforward. Also, I am specifying the experiments to run. So, the importance of the report will be much higher:

40%: your code (correctness, clarity, etc.), and

60%: your report.

What To Submit

Submit your program code and report on BlackBoard. Also, please submit a hardcopy of your report. I need only one submission from each group.