**6.825**
**Techniques in Artificial Intelligence**
2005/11/22

20/15

**Dan Ports**
drkp@mit.edu

# Project 3 Exercise 2

We begin by implementing the value iteration algorithm[1], and running it for a few iterations. The results are shown below.

A few minor implementation notes: we assume that if a move, either intentional or unintentional, moves into a wall (e.g. going west from state 1), the agent stays in the same state just as if it had done nothing. Also, we do not calculate utilities for taking actions from state 3, since it is a termination state; we assume that the expected utility of moving to that state is always the reward function $R(3) = 1$.

| t | State | North | East | South | West | Nothing | Best Action | Exp. Util. | Updated Util. |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | Nothing | 0.000000 | -0.100000 |
| 0 | 2 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | Nothing | 0.000000 | -0.100000 |
| 0 | 3 | - | - | - | - | - | - | 1.000000 | 1.000000 |
| 0 | 4 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | Nothing | 0.000000 | -0.100000 |
| 0 | 5 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | Nothing | 0.000000 | -0.100000 |
| 0 | 6 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | 0.000000 | Nothing | 0.000000 | -0.050000 |
| 1 | 1 | -0.100000 | -0.100000 | -0.100000 | -0.100000 | -0.100000 | Nothing | -0.100000 | -0.199900 |
| 1 | 2 | -0.045000 | 0.890000 | -0.045000 | -0.100000 | -0.100000 | East | 0.890000 | 0.789110 |
| 1 | 3 | - | - | - | - | - | - | 1.000000 | 1.000000 |
| 1 | 4 | -0.100000 | -0.100000 | -0.100000 | -0.100000 | -0.100000 | Nothing | -0.100000 | -0.199900 |
| 1 | 5 | -0.097500 | -0.055000 | -0.097500 | -0.100000 | -0.100000 | East | -0.055000 | -0.154945 |
| 1 | 6 | -0.052500 | 0.002500 | 0.892500 | -0.042500 | -0.050000 | South | 0.892500 | 0.841608 |
| 2 | 1 | -0.150450 | 0.690209 | -0.150450 | -0.199900 | -0.199900 | East | 0.690209 | 0.589519 |
| 2 | 2 | -0.099446 | 0.931708 | 0.750204 | -0.148202 | 0.789110 | East | 0.931708 | 0.830777 |
| 2 | 3 | - | - | - | - | - | - | 1.000000 | 1.000000 |
| 2 | 4 | -0.197652 | -0.159441 | -0.197652 | -0.199900 | -0.199900 | East | -0.159441 | -0.259281 |
| 2 | 5 | -0.107365 | 0.789155 | 0.742284 | -0.148202 | -0.154945 | East | 0.789155 | 0.688366 |
| 2 | 6 | 0.791780 | 0.849527 | 0.934333 | -0.047370 | 0.841608 | South | 0.934333 | 0.883399 |
| 3 | 1 | -0.162338 | 0.764211 | 0.601582 | 0.547079 | 0.589519 | East | 0.764211 | 0.663447 |
| 3 | 2 | 0.699005 | 0.975957 | 0.827175 | 0.606524 | 0.830777 | East | 0.975957 | 0.874981 |
| 3 | 3 | - | - | - | - | - | - | 1.000000 | 1.000000 |
| 3 | 4 | -0.211899 | 0.636041 | 0.552021 | -0.216841 | -0.259281 | East | 0.636041 | 0.535405 |
| 3 | 5 | 0.650735 | 0.871016 | 0.778905 | -0.157396 | 0.688366 | East | 0.871016 | 0.770145 |
| 3 | 6 | 0.873647 | 0.889229 | 0.978588 | 0.713699 | 0.883399 | South | 0.978588 | 0.927610 |

After three iterations, we see that the best policy is to move south when in state 6, and east otherwise.

10/10

**Extra credit.** The state utilities converge to the values shown below after 12 iterations. Though the correct policy is reached after only three iterations, the state utilities do not converge to within $\epsilon = 0.01$ until the 12th iteration.

[1] Python source code for the implementation is available at `http://www.ambulatoryclam.net/svn/classes/6.825/proj3exercises/ex1`

| t | State | North | East | South | West | Nothing | Best Action | Exp. Util. | Updated Util. |
|---|---|---|---|---|---|---|---|---|---|
| 12 | 1 | 0.729882 | 0.870637 | 0.775509 | 0.767229 | 0.769764 | East | 0.870637 | 0.769766 |
| 12 | 2 | 0.833938 | 0.985647 | 0.884683 | 0.778434 | 0.884661 | East | 0.985647 | 0.884661 |
| 12 | 3 | - | - | - | - | - | - | 1.000000 | 1.000000 |
| 12 | 4 | 0.724528 | 0.819891 | 0.770155 | 0.721602 | 0.719067 | East | 0.819891 | 0.719072 |
| 12 | 5 | 0.828268 | 0.929208 | 0.879013 | 0.732807 | 0.828278 | East | 0.929208 | 0.828279 |
| 12 | 6 | 0.931839 | 0.940425 | 0.988278 | 0.842314 | 0.937290 | South | 0.988278 | 0.937290 |