6CCS3AIN, Tutorial 04 (Version 1.0)

1. I have half an hour to spare in my busy schedule, and I have a choice between working quietly in my office and going out for a coffee.

   If I stay in my office, three things can happen: I can get some work done ($Utility = 8$), or I can get distracted looking at the US mid-term election forecast ($Utility = 1$), or a colleague might stop by to talk about some work we are doing on revising the curriculum ($Utility = 5$).

   If I go out for coffee, I will most likely enjoy a good cup of smooth caffienation ($Utility = 10$), but there is also a chance I will end up spilling coffee all over myself ($Utility = -20$).

   The probability of getting work done if I choose to stay in the office is 0.5, while the probabilities of getting distracted, and a colleague stopping by are 0.3 and 0.2 respectively.

   If I go out for a coffee, my chance of enjoying my beverage is 0.95, and the chance of spilling my drink is 0.05.

   (a) Compute the expected utility of staying in my office and of going out for a coffee.

   (b) By the principle of maximum expected utility, which action should I choose?

   (c) Would this decision change if I use the maximin or maximax decision criteria?

2. Consider the simple world that we studied in the lecture (Figure 1).

   (a) Write down a formal description of this as a Markov Decision Process (as in the slides).

   (b) Assume that actions are deterministic (so the agent moves with probability 1 in the direction it is trying to move) write down a version of the Bellman equation that would work in this case.

   Hint: Take the Bellman equation from the slides and simplify it so that for each a, $P(s'|s,a) = 1$ for one pair of s and s' and 0 for all other pairs.

   (c) Use this deterministic version of Bellman to run value iteration on the world, and obtain utility values for each state.

   You will need to run value iteration until it stabilises. Assume $\gamma = 1$ and set $U(s) = 0$ initially.

   (d) Write down the optimum policy given your solution to the deterministic version of value iteration.

3. Now consider the same world, but now assume that actions are non-deterministic. For any action, the action succeeds with probability 0.9 and the action completely fails with probability 0.1 (so that the agent does not move).

   For example, if the action is Up, the agent moves up with probability 0.9, and stays in the same place with probability 0.1.

   (a) Use the non-deterministic version of Bellman to run value iteration on the world, and obtain utility values for each state.

   Note that this question is asking for the values after the values for all states have converged.

   Hint: A spreadsheet is a good way to simplify the calculation.

   (b) Write down the optimum policy. How does this differ from the optimum policy for non-deterministic model that you calculated in Q2? What does this suggest?

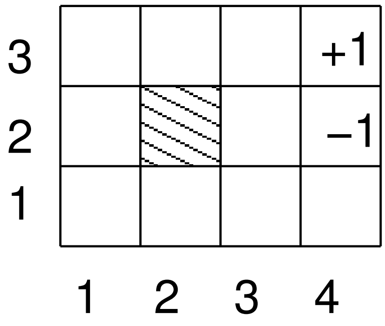
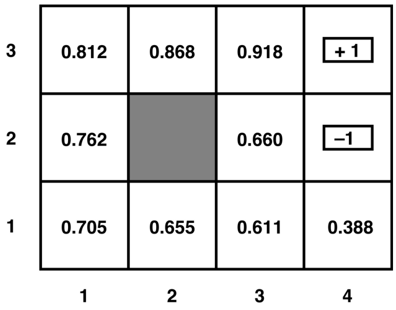


Figure 1: The simple world that we studied in the lecture.



3	0.812	0.868	0.918	+1
2	0.762	▨	0.660	–1
1	0.705	0.655	0.611	0.388
	1	2	3	4

Figure 2: The utility value for the simple world that we studied in the lecture where the problem has the rewards and the transition model from the lecture.

4. Now consider the version of this problem from the lecture — recall that in this version the transition model is such that when the agent tries to move in a given direction, 10% of the time it moves to the left of the chosen direction[1], and 10% of the time it moves to the right. Figure 2 shows the utilities that are obtained under the optimal policy for this version ofthe problem.

   (a) Using the utility values in Figure 2, compute the expected utility of state (3, 1). What do you notice about this value?

   (b) Now compute the action that maximises expected utility.

   (c) Given the utility values of each state in Figure 2, what would the policies be if we chose actions using:

      i. maximin

      ii. maximax

   Compare these to the policy chosen by MEU (this is available in the slides).

5. There is no additional computational part for this tutorial. Partly this is because the spreadsheets in questions 2 and 3 are a form of computation; and partly this is because you will have to implement an MDP solver for the coursework, so you'll get a chance to engage with the computational aspects of Bellman then.

[1] Here "left" and "right" are relative to the direction that the agent is trying to move in. If you find that hard to visualize, you might try thinking of "left" as meaning "90 degress to anticlockwise of" and "right" as "90 degrees to clockwise of".

6CCS3AIN, Tutorial 04 Answers (Version 1.0)

1. (a) We have a decision between staying in the `office` and going `out`. Both options will lead to one or more states.

   Staying in the `office` means that I will either `work`, get `distracted`, or talk with `colleague`. These states have the following utilities:

$$\begin{aligned} U(work) &= 8 \\ U(distracted) &= 1 \\ U(colleague) &= 5 \end{aligned}$$

   and the probabilities of these happening, given I stay in the `office` are:

$$\begin{aligned} P(work|office) &= 0.5 \\ P(distracted|office) &= 0.3 \\ P(colleague|office) &= 0.2 \end{aligned}$$

   Thus the expected utility of staying in the office is:

$$\begin{aligned} EU(office) &= 0.5 \cdot 8 + 0.3 \cdot 1 + 0.2 \cdot 5 \\ &= 5.3 \end{aligned}$$

   Going out can result in the states `coffee` and `spill`, with utilities:

$$\begin{aligned} U(coffee) &= 10 \\ U(spill) &= -20 \end{aligned}$$

   and the relevant probabilities are:

$$\begin{aligned} P(coffee|out) &= 0.95 \\ P(spill|out) &= 0.05 \end{aligned}$$

   Thus $EU(out) = 8.5$.

   (b) The principle of maximum expected utility says to pick the option with the greatest expected utility, in this case that is to go `out`.

   (c) The maximin principle says to pick the options based on the utillty of their worst outsomes, and to pick the option with the largest utility for its worst outcome.

   Formally we have:

$$\begin{aligned} a &= \arg \max_{s \in \{office, out\}} \{min(U(s))\} \\ &= \arg \max_{s \in \{office, out\}} \{min^{office}(8, 1, 5), min^{out}(10, -20)\} \\ &= \arg \max_{s \in \{office, out\}} \{1^{office}, -20^{out}\} \\ &= office \end{aligned}$$

   so the maximin principle would say I should stay in the `office`.

   The maximax principle says to pick the option based on the best outcome of each action, and to pick the one with the largest utility for the best outcome, in other words:

$$a = \arg \max_{s \in \{office, out\}} \{max(U(s))\}$$

   which will pick `out`.

2. The grid world is:

3				+1
2		▨		–1
1				
	1	2	3	4

(a) The formal description of this as an MDP is the following:

- States: (1, 1), (1, 2), (1, 3), . . . (2, 1), (2, 3), . . . (4, 3)
  Note that the location (2, 2), where there is an obstacle, does not correspond to a state.
- Initial state: (1, 1).
- Actions: Up, Down, Left Right for all states.
- Reward:

$$R(s) = \begin{cases} 1 & \text{for } s = (4,3) \\ -1 & \text{for } s = (4,2) \\ -0.04 & \text{otherwise} \end{cases}$$

- Transition model, $P(s'|s,a)$:

$$\begin{aligned} P((1,2)|(1,1), Up) &= 0.8 \\ P((1,1)|(1,1), Up) &= 0.1 \\ P((2,1)|(1,1), Up) &= 0.1 \\ P((1,1)|(1,1), Down) &= 0.9 \\ &\vdots \end{aligned}$$

(b) For a deterministic model, we no longer have to worry about the expected utility of an action, because we know it succeeds, and we have:

$$U(s) = R(s) + \gamma max_{a \in A(s)} U(s')$$

where s' is the state that results from action a.

(c) We will assume that $\gamma = 1$, and set $U(s)$ to 0 initially. Then, after the first round of value iteration using the above formula, we get:

3	–0.04	–0.04	0.96	+1
2	–0.04	▨	–0.04	–1
1	–0.04	–0.04	–0.04	–0.04
	1	2	3	4

After a second round we get:

3	–0.08	0.92	0.96	+1
2	–0.08	▨	0.92	–1
1	–0.08	–0.08	–0.08	–0.08
	1	2	3	4

so after 3 more rounds we will end up with:

3	0.88	0.92	0.96	+1
2	0.84	▨	0.92	−1
1	0.8	0.84	0.88	0.84
	1	2	3	4

when the values will change no more.

See the spreadsheet (on KEATS) for detail on the calculations.

(d) At each state, the optimum policy is to pick the action with the highest expected utility. Since actions are deterministic in this case, that equates to deciding to move to the state with the highest utility (make sure you understand why this is the case). As a result, we have:

3	→	→	→	
2	↑	▨	↑	
1	↑ →	→	↑	←
	1	2	3	4

Note that we assume the agent does not move once it has reached $(4,3)$ or $(4,2)$ as in the lecture.

3. (a) My spreadsheet tells me that when the values have converged, we have (rounding to two decimal places):

3	0.87	0.91	0.96	+1
2	0.82	▨	0.91	−1
1	0.78	0.82	0.87	0.82
	1	2	3	4

For $\gamma = 1$.

The calculation for the first iteration, for $(3,3)$ is:

$$\begin{array}{rlll} U(s) & = -0.04 + \gamma max(& (1 \cdot 0) & \textit{Up} \\ & & (0.9 \cdot 0 + 0.1 \cdot 0) & \textit{Down} \\ & & (0.9 \cdot 0 + 0.1 \cdot 0) & \textit{Left} \\ & & (0.9 \cdot 1 + 0.1 \cdot 0) & \textit{Right} \end{array}$$

and so on for the rest of the states.

See the spreadsheet (on KEATS) for the rest of intermediate calculations.

(b) The computation of the best action in each state parallels the calculation above. For $(3,3)$ we want to establish the maximum of the expected values for the 4 possible actions:

$$\begin{array}{ll} (1 \cdot 0.96) & \textit{Up} \\ (0.9 \cdot 0.91 + 0.1 \cdot 0.96) & \textit{Down} \\ (0.9 \cdot 0.91 + 0.1 \cdot 0.96) & \textit{Left} \\ (0.9 \cdot 1 + 0.1 \cdot 0.96) & \textit{Right} \end{array}$$

so Right will be the best action.

Repeating this for all the states, we quickly find that the optimum policy is the same as in the deterministic case (though the expected utility of the policy will be lower since the agent will spend more time in lower utility states, and will take more steps to get to the goal).

What that suggests is that for a motion model like the one we have here, the deterministic version of value iteration is a pretty good approximation of the non-deterministic version (though the difference in the utilities is obvious from the figures).

4. (a)

$$\begin{array}{rlll} U((3,1)) & = -0.04 + \gamma max(& 0.8 \cdot 0.660 + 0.1 \cdot 0.655 + 0.1 \cdot 0.388 & Up \\ & & 0.8 \cdot 0.611 + 0.1 \cdot 0.655 + 0.1 \cdot 0.388 & Down \\ & & 0.8 \cdot 0.655 + 0.1 \cdot 0.660 + 0.1 \cdot 0.611 & Left \\ & & 0.8 \cdot 0.388 + 0.1 \cdot 0.660 + 0.1 \cdot 0.611 & Right \end{array}$$

which reduces to:

$$\begin{array}{rlll} U((3,1)) & = -0.04 + \gamma max(& 0.6323 & Up \\ & & 0.5931 & Down \\ & & 0.6511 & Left \\ & & 0.4375 & Right \end{array}$$

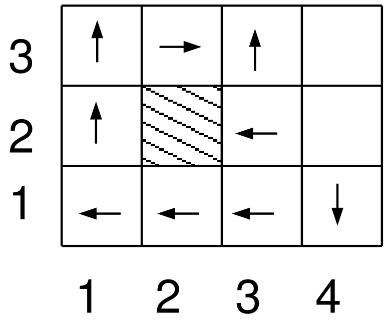
which comes to 0.611.

This is the same as the utility of the state. Which is exactly what we would expect from the Bellman equation once we have the state utility that is the one under the optimum policy (which is what happens when value iteration terminates).

(b) The numbers in the equation above are the expected utilities of the actions. So the action with the maximum expected utility is Left.

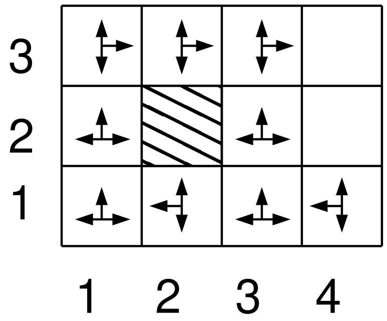
Note that this is not the same as moving towards the state with the maximum utility.

(c) i. For the maximin policy, we look at the worst outcome of each action, and pick the action which maximises this worst outcome. We get:



The policy makes sure that in every state the agent has no chance of entering $(4,2)$ while making sure that it will eventually get to $(4,3)$.

ii. For the maximax policy, we look at the best outcome of each action and pick the action which maximises this. Because every action has three possible outcomes, there are always three actions which can lead to the maximum utility. Since maximax ignores the probability of outcomes, it treats all of these actions as equally good:



The only action that is not selected by maximax is the one that points away from the state with the maximum utility.

Of course, an agent can't pick three actions, so any agent that wanted to employ maximax in this environment would have to pick one of the three actions.

5. There is no question 5.