

Handout Solutions: Markov Decision Processes

Which of the Following are MDPs?

Most single-agent situations can be construed as MDPs with a sufficiently complicated state description. Note, however, that (1) if some elements of the state needed to specify the reward function and transition model are not *observable to the agent* then the situation is a more complicated partially-observable markov decision process (POMDP). Different techniques apply to POMDPs (see textbook). If there are multiple agents, then MDP techniques do not apply. You need minimax search or game theory.

Blackjack This is an MDP with a complicated transition model. See next section handout for details.

Rock, Paper, Scissors This is a multi-agent environment, which is not an MDP environment. You could, however, imagine formulating the game as an MDP by specifying some transition model that determined the opponents play. Such an agent might play well, but would likely have exploitable predictability.

Elevator Control The state includes what floor the elevator is on and what buttons are pressed inside and outside an elevator. Actions include moving up or down a floor, or opening the door. A suitable reward function would penalize the elevator for having unserved requests (buttons lit up). The transition model would need to model the probability that each button was pressed at each time step. Note that if the reward were tied to some unobservable aspect of the state (such as how many people are in the elevator), then this would be a POMDP, for which the optimal policy would be more difficult to learn.

Tightrope-Walking Robot MDP techniques do extend to the continuous domain, although this course won't investigate such scenarios. The state description necessary to model tightrope-walking robot would likely include the robot's position, velocity, and balance.

Golf as an MDP

Recall the golf MDP we looked at in section:

- State Space: $\{Tee, Fairway, Sand, Green\}$
- Actions: $\{Conservative, Power\ shot\}$
- Initial State: Tee
- Transition Model: $T(s, a, s')$, the probability of going from s to s' with action a .

s	a	s'	$T(s, a, s')$
Tee	Conservative	Fairway	0.9
Tee	Conservative	Sand	0.1
Tee	Power shot	Green	0.5
Tee	Power shot	Sand	0.5
Fairway	Conservative	Green	0.8
Fairway	Conservative	Sand	0.2
Sand	Conservative	Green	1.0

- Reward Function:

s	$R(s)$
Tee	-1
Fairway	-1
Sand	-2
Green	3

For the conservative (c) policy, what is the utility of starting at the tee? First, some details: let us say that there is a *done* state with no rewards that can only be reached from the *green* via the action *finish*. Then, $R(\text{green}, \text{finish}, \text{done}) = 3$. Note the reward here only depends upon where you began (s), not where you end up (s'). Also, let's assume that the discount rate $\gamma = 1$.

Now, we can compute $V_C(\text{tee})$ in terms of other utilities using the recursive definition of utility:

$$\begin{aligned}
 V_C(\text{tee}) &= \sum_{s'} T(\text{tee}, c, s') [R(\text{tee}, c, s') + \gamma V_C(s')] \\
 V_C(\text{tee}) &= T(\text{tee}, c, \text{fairway}) [R(\text{tee}, c, \text{fairway}) + \gamma V_C(\text{fairway})] \\
 &\quad + T(\text{tee}, c, \text{sand}) [R(\text{tee}, c, \text{sand}) + \gamma V_C(\text{sand})] \\
 &= 0.9[-1 + V_C(\text{fairway})] + 0.1[-1 + V_C(\text{sand})] \\
 V_C(\text{fairway}) &= 0.8[-1 + V_C(\text{green})] + 0.2[-1 + V_C(\text{sand})] \\
 V_C(\text{sand}) &= 1.0[-2 + V_C(\text{green})] \\
 V_C(\text{green}) &= 1.0[3 + V_C(\text{done})] = 3
 \end{aligned}$$

Armed with a value for $V_C(\text{green})$, we can now compute that $V_C(\text{sand}) = 1$, $V_C(\text{fairway}) = 1.6$ and $V_C(\text{tee}) = 0.54$ via the equations above, working from bottom to top.

Computing utilities using value iteration In value iteration, we build a table of utility estimates and iteratively improve them until they converge to the true optimal utilities for the game. We use the following update equation:

$$V_{i+1}(s) = \max_a \sum_{s'} T(s, a, s') [R(s, a, s') + \gamma V_i(s')] \quad (1)$$

Recall that we can always initialize $V_0(s) = 0$ for all s . Then, we can fill in the following table one row at a time, using the previous row to supply values for $V_t(s)$.

i	$V_i(\text{tee})$	$V_i(\text{fairway})$	$V_i(\text{sand})$	$V_i(\text{green})$	$V_i(\text{done})$
0	0	0	0	0	0
1	-1	-1	-2	3	0
2	-0.5	1	1	3	0
3	1	1.6	1	3	0

For instance, we compute $V_2(\text{tee}) = -0.5$ as follows :

$$\begin{aligned} V_2(\text{tee}) &= \max(0.5[-1 + V_1(\text{sand})] + 0.5[-1 + V_1(\text{green}), \\ &\quad 0.9[-1 + V_1(\text{fairway})] + 0.1[-1 + V_1(\text{sand})]) \\ &= \max(0.5[-3] + 0.5[2], 0.9[-2] + 0.1[-3]) = \max(-0.5, -2.1) = -0.5 \end{aligned}$$