Q3. [9 pts] A Multiplayer MDP

Alice and Bob are playing a game on a 2-by-2 grid, shown below. To start the game, a puck is placed on square A.

At each time step, the available actions are:

- Up, Left, Down, Right: These actions move the puck deterministically. Actions that move the puck off the grid are disallowed. These actions give both players a reward of 0.
- Exit: This action ends the game. Note that the Exit action can be taken no matter where the puck is. This action gives Alice and Bob rewards depending on the puck's final location, as shown below.

Each player tries to maximize their own reward, and does not care about what rewards the other player gets.

X	Y
W	Z

Figure 1: The grid that the puck moves on.

Puck location when "exit" is taken	Alice's reward	Bob's reward
W	−1	+3
X	0	0
Y	+1	+2
Z	−1	+2

Figure 2: Exit rewards for Alice and Bob.

(a) [2 pts] Suppose Bob is the only player taking actions in this game. Bob models this game as an MDP with a discount factor $0 < \gamma \leq 1$.

For which of the following states does Bob's optimal policy depend on the value of γ? Select all that apply.

☐ W ☐ X ■ Y ■ Z ○ None of the above.

Clarification during exam: To start the game, a puck is placed on square W (not A).

Note that in this question, Bob is the only player, so all we're concerned about is maximizing Bob's utility. Alice's utility does not matter in this question.

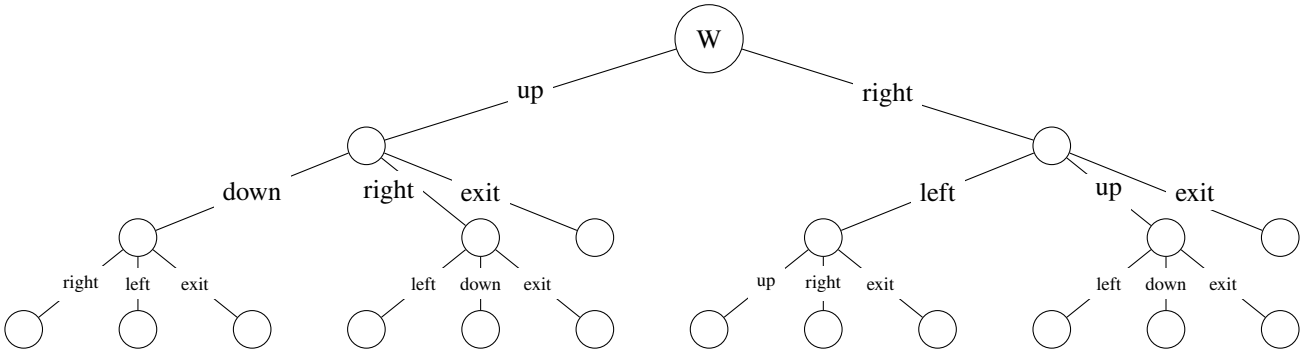
Bob gets highest reward at W, so the optimal action at W is always to exit.

The optimal policy from X is always {down, exit} to move into state W and exit there for a reward of 3. Even though there's a discount factor, we know that a reward of 3γ (one discount factor applied) is always going to be greater than the reward of 0 (if we had exited from X), because $\gamma > 0$.

At Y and Z, if the discount factor is low (i.e. future rewards are heavily discounted), the optimal action is to immediately exit and get a reward of 2. If the discount factor is high enough (i.e. future rewards are not so heavily discounted), then it's better to go to state W and get the discounted reward of 3γ for exiting from state W.

For the rest of the question, Alice and Bob alternate taking actions. Alice goes first.

Alice models this game with the following game tree, and wants to run depth-limited search with limit 3 turns. She uses an evaluation of 0 for any leaf node that is not a terminal state. (Note: Circles do not necessarily represent chance nodes.)



(b) [1 pt] This is a zero-sum game.

○ True ● False

Alice and Bob have different utilities in the tree, partially cooperative and partially competitive, as shown in the reward table.

In order for this game to be zero-sum, Alice's utility would need to be the negative of Bob's utility.

(c) [1 pt] What is Alice's value of the root node of the tree?

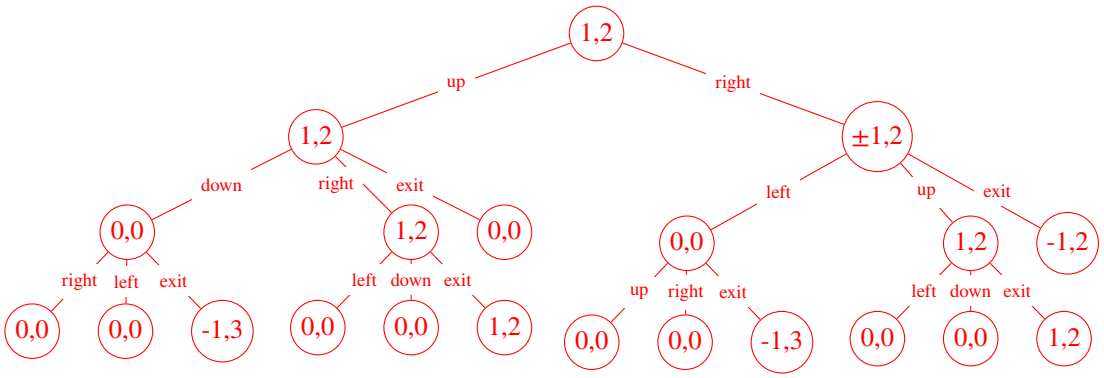
1

(d) [1 pt] What is Bob's value of the root node of the tree?

2

Clarification during exam: The left-most subtree under the game tree should say "right, up, exit" not "right, left, exit".

Filling out the tree with Alice and Bob's utility gives the following tree.



At the terminal nodes, we fill in the utilities associated with the corresponding sequence of actions. For example, going through the leaf nodes of the tree left to right:

The left-most leaf node corresponds to the actions {up, down, right}. This sequence of actions has not led to the game ending, so we cannot evaluate the value (Alice and Bob's reward) at this state. However, we note that Alice uses an evaluation of 0 for any non-terminal leaf node, so the value of this state is $(0, 0)$.

Similarly, the next leaf node corresponds to {up, down, up} (clarification corrected this from {up, down, left}, which is illegal). Again, the game has not ended, so we use the evaluation of 0 to find the value of this state is $(0, 0)$.

The next leaf node corresponds to {up, down, exit}. This ends the game with an exit from W, which gives Alice reward of -1 and Bob reward of $+3$. In the tree, we denote this as $(-1, 3)$, where the left value is Alice's reward and the right value is Bob's reward.

Going through the other leaf nodes, we can write in $(0, 0)$ for any non-terminal leaf nodes and the exit rewards for any leaf nodes where an exit action was taken.

To solve the tree, we need to evaluate which layers correspond to which player. Alice is going first, so the two branches from the root node (up and right) correspond to Alice's action (where she will maximize her own utility). Since Alice and Bob take turns, the next layer then corresponds to Bob's action (where he will maximize his own utility), and the final layer corresponds to Alice's action again.

In layers where it's Bob's turn, we pick the child with the highest right-value (second value in the tuple). In layers where it's Alice's turn, we pick the child with the highest left-value (first value in the tuple).

Note that we have a node labeled $(\pm 1, 2)$. This is because at this node, Bob is indifferent between the $(1, 2)$ and the $(-1, 2)$ nodes, since he gets a reward of 2 either way. Since we didn't specify a tiebreaker mechanism, the value here could be either $(1, 2)$ or $(-1, 2)$. However, the tiebreaker mechanism doesn't matter because at the root node, Alice has a choice between $(1, 2)$ and $(\pm 1, 2)$ and will choose $(1, 2)$. If Alice is unsure of Bob's tiebreaker strategy, she'd choose the guaranteed $(1, 2)$ over the $(\pm 1, 2)$ where she might risk getting -1.

(e) [2 pts] Assuming that Alice and Bob play optimally, what are possible sequences of actions that they might play? Select all that apply.

- [x] up, right, exit
- [ ] up, right, down
- [ ] right, left, right
- [ ] right, up, exit
- [ ] right, exit
- ○ None of the above.

Using the tree above, we see that the value $(1, 2)$ is associated with either the action sequence {up, right, exit}, or {right, up, exit}.

However, as discussed above, Alice does not know Bob's tiebreaker strategy, so the $(1, 2)$ value at the root is associated with the left branch {up, right, exit}. In other words, {right, up, exit} is suboptimal with no further knowledge of Bob's strategy, because we can't be sure if Bob will force the $(1, 2)$ or the $(-1, 2)$ outcome.

Alice instead decides to model the game as an MDP. Assumptions:

- $\gamma = 0.5$
- Alice knows Bob's policy is π.
- $D(s, a)$ represents the new state you move into when you start at state s and take action a.

(f) [2 pts] Fill in the blanks to derive a modified Bellman equation that Alice can use to compute the values of states.

Let $s' = D(s, a)$ and $s'' = D(s', \pi(s'))$.

$$V(s) = \max_a R(s, a, s') + \textbf{(i)}\ \textbf{(ii)} + \textbf{(iii)}\ \textbf{(iv)}$$

(i)	○ 1	● 0.5	○ 0.25
(ii)	○ 0	○ $R(s, \pi(s), s')$	● $R(s', \pi(s'), s'')$
(iii)	○ 1	○ 0.5	● 0.25
(iv)	○ $V(s)$	○ $V(s')$	● $V(s'')$

Recall that the Bellman equation relates the values of states $V(s)$ with the values of other states $V(s')$. In this MDP, Alice and Bob alternate choosing actions, so in order to relate Alice's value at a state to Alice's value function at some other state, we need to iterate two timesteps into the future (to reach the next time it's Alice's turn).

To consider the first time step into the future, we need to consider Alice's immediate reward $R(s, a, s')$, which is already in the answer. After the first time step, the game has transitioned from s into s'.

Then, for the second time step into the future, we need to consider the action that Bob will take, $pi(s')$. This will transition the game from state s' to another state, denoted s''. We also need to consider the reward that Alice will get from this transition (that Bob chose), which is $R(s', \pi(s'), s'')$.

Once we reach s'' two time steps later, it's Alice's turn again, so we can use the recursive definition $V(s'')$ to denote the value of Alice starting at s'' and acting optimally.

Finally, we need to make sure to apply one discount of 0.5 to the reward $R(s', \pi(s'), s'')$ one time step into the future. We need to apply two discounts to the rewards that are 2+ time steps into the future, $V(s'')$.

$V(s) = \max_a(R(s, a, s') + 0.5R(s', \pi(s'), s'') + 0.25V(s''))$

Q5. [13 pts] MDPs: Jim & Pam Part 2

Jim and Pam are on a $1x5$ grid, where Jim starts at square 1, and Pam is fixed at square 5.

Jim				Pam

In each time step, Jim chooses to either move right or to rest. Choosing to move succeeds with probability p and fails with probability $1-p$, in which case Jim stays in his original square (Jim received 0 utility regardless of success or failure). Choosing to rest always succeeds, and gives $R(d) = 4^{5-d}$ utility, where d is the distance between Jim and Pam. For example, at the start, where $d = 4$, if Jim decides to rest, he gets 4 utility. We represent this as an infinite horizon MDP with no terminal state.

(a) [3 pts] Jim is considering two policies:

Policy 1: Rest at the start forever.

Policy 2: Attempt to move right once, and then, regardless of success or failure, rest forever.

Assuming that Jim starts in square 1, for what values of p is Policy 1 superior to Policy 2 when the discount factor $\gamma = 0.5$? Hint: the sum S of an infinite geometric series with starting value a and ratio r is $S = \frac{a}{1-r}$

Show your work. $0 \leq$ [0] $\leq p \leq$ [1/3] ≤ 1

$V_{rest} = 8$ and $V_{move->rest} = 16p + 4(1-p)$, so $8 \geq 16p + 4(1-p)$

(b) [3 pts]

Now assume that $p = 1$. Still assuming that Jim starts in square 1, for what values of γ is Policy 1 superior to Policy 2?

Show your work. $0 \leq$ [0] $< \gamma <$ [1/4] ≤ 1

$V_{rest} = \frac{4}{1-\gamma}$ and $V_{move->rest} = \frac{16\gamma}{1-\gamma}$, so $\frac{4}{1-\gamma} \geq \frac{16\gamma}{1-\gamma}$

(c) [7 pts] For the following subparts, assume that $\gamma = 0.5$ and $p = 1$.

(i) [3 pts] Perform two iterations of value iteration, for the following locations of Jim. Show your work.

States	$s_J = 1$	$s_J = 2$	$s_J = 3$	$s_J = 4$	$s_J = 5$
V_0	0	0	0	0	0
V_1	4	4^2	4^3	4^4	4^5
V_2	$2^{-1} * 4^2$	$2^{-1} * 4^3$	$2^{-1} * 4^4$	$2^{-1} * 4^5$	$4^5 + 2^{-1} * 4^5$

(ii) [3 pts] Perform two iterations of policy iteration for the following locations of Jim.

States	$s_J = 1$	$s_J = 2$	$s_J = 3$	$s_J = 4$	$s_J = 5$
π_i	$a_J = right$	$a_J = rest$	$a_J = rest$	$a_J = rest$	$a_J = rest$
V^{π_i}	4^2	$2 * 4^2$	$2 * 4^3$	$2 * 4^4$	$2 * 4^5$
π_{i+1}	$a_J = right$	$a_J = right$	$a_J = right$	$a_J = right$	$a_J = rest$
$V^{\pi_{i+1}}$	$2^{-3} * 4^5$	$2^{-2} * 4^5$	$2^{-1} * 4^5$	$2^0 * 4^5$	$2^1 * 4^5$
π_{i+2}	$a_J = right$	$a_J = right$	$a_J = right$	$a_J = right$	$a_J = rest$

(iii) [1 pt] Assuming that policy iteration has converged, Jim argues it isn't guaranteed values have converged yet, so they need to run value iteration to get the correct values. Pam agrees that policy convergence doesn't guarantee value convergence, but thinks that we don't need to switch to value iteration, as if we continue running policy iteration, eventually the values will converge as well. Who is correct and why?

○ Jim ● Pam

Pam is correct. Policy iteration contains the policy evaluation step, which is essentially just value iteration once the policy has converged.

Q4. [13 pts] Slightly Different MDPs

Each subpart of this question is independent.

For all MDPs in this question, you may assume that:

- The maximum reward for any single action is some fixed number $R > 0$, and the minimum reward is 0.
- The discount factor satisfies $0 < \gamma \leq 1$.
- There are a finite number of possible states and actions.

(a) [2 pts] Which statements are always true? Select all that apply.

- ☐ $\sum_{s \in S} T(s, a, s') = 1$.
- ■ $\sum_{s' \in S} T(s, a, s') = 1$.
- ☐ $\sum_{a \in A} T(s, a, s') \leq 1$.
- ☐ For all state-action pairs (s, a), there exists some s', such that $T(s, a, s') = 1$.
- ○ None of the above

Clarification during exam: Q4 - The discount factor is $0 < \gamma < 1$.

(A): False. This adds up the probability of going from all states to a certain state, which doesn't necessarily sum to 1. As an intuitive example (disregarding the constant action a), consider a 3-state MDP with states X, Y, and Z. $P(X \rightarrow Z) + P(Y \rightarrow Z) + P(Z \rightarrow Z) \neq 1$. Maybe it's likely ($< 50\%$) to reach Z from any of the 3 states, which would make this expression add to more than 1.

(B): True. This adds up the probability of going from 1 state to all other states, which sums to 1 (because from one state, once you take an action, you have to land in some state). Using the example above, $P(X \rightarrow X) + P(X \rightarrow Y) + P(X \rightarrow Z) = 1$ because from X, once you take an action, you have to land in X, Y, or Z.

(C): False. Suppose that in the example, if you're in X, any action is guaranteed to land you in Z, no matter what action you take. Assume there are 2 actions, Left and Right. Then $T(X, \text{Left}, Z) + T(X, \text{Right}, Z) = 2$, which is not ≤ 1.

(D): False. $T(s, a, s') = 1$ would say that in state s, taking action a always lands you in s'. However, there is no guarantee that taking some action in the MDP has a guaranteed outcome. For example, consider Gridworld from lecture with the exit action removed: every action is probabilistic, and there is no action that has a guaranteed s' successor state.

(b) [2 pts] Which statements are always true? Select all that apply.

- ■ Every MDP has a unique set of optimal values for each state.
- ☐ Every MDP has a unique optimal policy.
- ☐ If we change the discount factor γ of an MDP, the original optimal policy will remain optimal.
- ■ If we scale the reward function $r(s, a)$ of an MDP by a constant multiplier $\alpha > 0$, the original optimal policy will remain optimal.
- ○ None of the above

(A): True. The optimal values are the solutions to the Bellman equations, and they exist and are finite if $0 < \gamma < 1$ and the state space is finite. In other words, from a given state, the expected discounted sum of rewards for acting optimally is a unique value.

(B): False. An MDP could have multiple optimal policies. For example, consider a state where every action results in the same successor state. Then the optimal policies could assign any action at this state.

(C): False. Consider an MDP with two states, A and B. At A we can either go to B or exit, getting a reward of 1, and at B, we can only exit, getting a reward of 10. If the discount factor is 0.9, the optimal action at A would be to go to B; whereas if the discount factor is 0.01, the optimal action at A is to exit directly.

(D): True. The optimal policy is determined by taking a argmax over values; if we scale all the values up or down by a constant, the relative ordering of values stays the same.

(c) [2 pts] Which statements are true? Select all that apply.

■ Policy iteration is guaranteed to converge to a policy whose values are the same as the values computed by value iteration in the limit.

■ Policy iteration usually converges to exact values faster than value iteration.

☐ Temporal difference (TD) learning is off-policy.

■ An agent is performing Q-learning, using a table representation of Q (with all Q-values initialized to 0). If this agent takes only optimal actions during learning, this agent will eventually learn the optimal policy.

○ None of the above

(A): True. Policy iteration is guaranteed to converge to an optimal policy. Value iteration computes the values of the optimal policy. If we compute the values of the optimal policy (from policy iteration), we'll get the same numbers as if we performed value iteration.

(B): True. Most of the time, value iteration converges towards optimal values in the limit but never reaches the exact values. However, policy iteration eventually finds the optimal policy, and then the policy evaluation step finds exact values as the solution of the linear equations.

(C): False. TD learning involves collecting samples using a particular policy $\pi(s)$ on the MDP, and thus it is on-policy, i.e., it learns values for π, the policy that is generating the samples.

(D): True.

(d) [2 pts] We modify the reward function of an MDP by adding or subtracting at most ϵ from each single-step reward. (Assume $\epsilon > 0$.)

We fix a policy π, and compute $V_\pi(s)$, the values of all states s in the original MDP under policy π.

Then, we compute $V'_\pi(s)$, the values of all states in the modified MDP under the same policy π.

What is the maximum possible difference $\left|V'_\pi(s) - V_\pi(s)\right|$ for any state s?

○ ϵ

○ $\gamma\epsilon$

● $\sum_{n=0}^{\infty} \epsilon(\gamma)^n = \epsilon/(1-\gamma)$

○ ϵR

○ None of the above

Intuitively, because the policies are the same, the future action sequences from any given state is also the same. (Formally, because an action can result in landing in multiple different states, we'd have to say something like, the distribution over futures is the same at a given state.)

Recall that the value of a state is the expected, discounted sum of rewards for acting optimally from that state for the rest of the time. So at each time step that we act, the difference in reward is at most ϵ (discounted appropriately). The sum of discounted differences at each time step is: $1 + \gamma + \gamma^2 + \ldots = \epsilon/(1-\gamma)$.