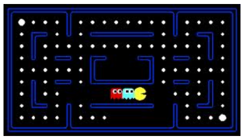
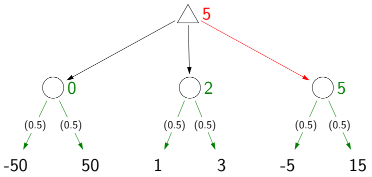
Games: expectimax



Expectimax example

Example: expectimax

$\pi_{\text{opp}}(s,a) = \frac{1}{2}$ for $a \in \text{Actions}(s)$

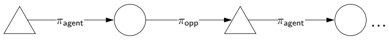


$\boxed{V_{\text{exptmax}}(s_{\text{start}}) = 5}$

- Game evaluation just gave us the value of the game with two fixed policies π_{agent} and π_{opp}. But we are not handed a policy π_{agent}; we are trying to find the best policy. Expectimax gives us exactly that.
- In the game tree, we will now use an upward-pointing triangle to denote states where the player is maximizing over actions (we call them **max nodes**).
- At max nodes, instead of averaging with respect to a policy, we take the max of the values of the children.
- This computation produces the **expectimax value** $V_{\text{exptmax}}(s)$ for a state s, which is the maximum expected utility of any agent policy when playing with respect to a fixed and known opponent policy π_{opp}.

Expectimax recurrence

Analogy: recurrence for value iteration in MDPs



$$V_{\text{exptmax}}(s) = \begin{cases} \text{Utility}(s) & \text{IsEnd}(s) \\ \max_{a \in \text{Actions}(s)} V_{\text{exptmax}}(\text{Succ}(s,a)) & \text{Player}(s) = \text{agent} \\ \sum_{a \in \text{Actions}(s)} \pi_{\text{opp}}(s,a) V_{\text{exptmax}}(\text{Succ}(s,a)) & \text{Player}(s) = \text{opp} \end{cases}$$

- The recurrence for the expectimax value V_{exptmax} is exactly the same as the one for the game value V_{eval}, except that we maximize over the agent's actions rather than following a fixed agent policy (which we don't know now).
- Where game evaluation was the analogue of policy evaluation for MDPs, expectimax is the analogue of value iteration.