CS3243 Tutorial 6

Eric Han

March 17, 2022

Annoucements

1. Assignment 4 scores are now on Gradebook, please check.
2. Assignment 5 is *currently* being marked, check back on turnitin to check its status.
3. Today's tutorial is super short, so we can return back to discuss Midterms, wordle.

Question 2

We have an attacker looking at three targets: t_1, t_2, t_3. A defender must choose which of the two targets it will guard; however, the attacker has an advantage: it can observe what the defender is doing before it chooses its move. If an attacker successfully attacks it receives a payoff of 1 and the defender gets a payoff of -1.

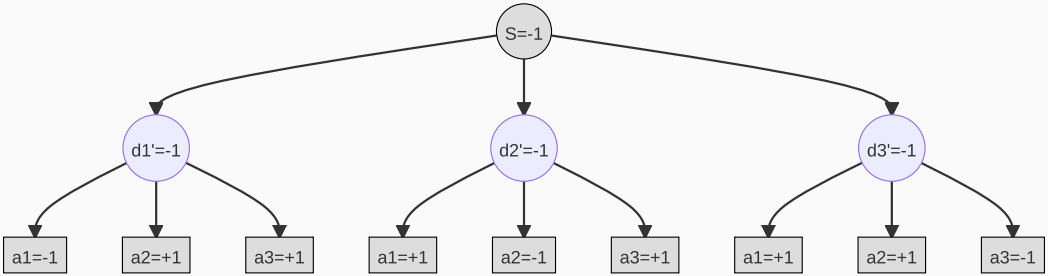
a. Model this problem as a minimax search problem.
b. Draw out the search tree.
c. What is the defender's payoff in this game?

Recap

1. What is the minimax algorithm?
2. Why is it used?
3. What are the ingredients needed to setup?

Answer

1. **Actors**: Min - Attackers, Max - Defenders
2. **Leaf Cost**: -1 if attack successful, $+1$ otherwise



No matter how the attackers choose, the value for the defenders would always be -1.

Question: How to get around this issue?

Question 3

With the MINIMAX algorithm, we know that the value v, computed at the root (i.e, the utility for the MAX player), is a worst-case value. This means that if the opponent MIN does not act optimally, the actual outcome v' for MAX can only be better, and never worse than v. That said, the MINIMAX algorithm may not select the optimal move given sub-optimal play from the MIN player.

Recap

1. What is the impact of choosing min/max in our computation?
2. When was MINIMAX famously used in AI?

Question 3

With the `MINIMAX` algorithm, we know that the value v, computed at the root (i.e, the utility for the MAX player), is a worst-case value. This means that if the opponent MIN does not act optimally, the actual outcome v' for MAX can only be better, and never worse than v. That said, the `MINIMAX` algorithm may not select the optimal move given sub-optimal play from the `MIN` player.

Recap

1. What is the impact of choosing min/max in our computation?
2. When was `MINIMAX` famously used in AI?

IBM Deep Blue versus Garry Kasparov in Chess.

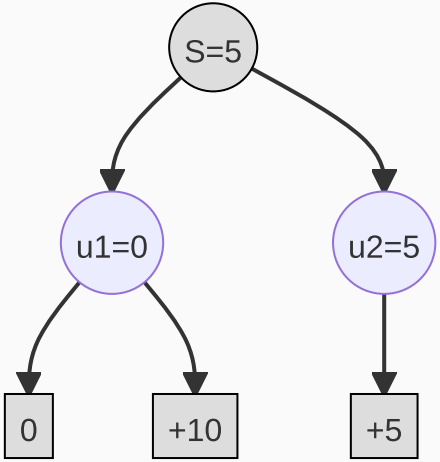


Figure 1: Sub Optimal Example

Limitations of `MINIMAX`:

- Assumes that Min player playes optimally: ie. when it sees 0 and $+10$ it will choose 0.
- So to avoid the case of Min player choosing 0
- It selects the action that eliminates the option of choosing 0.

Additional qn:

1. Comment on the use of `MINIMAX` on chess using the same technique as IBM Deep Blue.
2. What are the limitations and how to mitigate them?