

Adversarial Search Project Report

Introduction

The objective of this report is to analyze the performance of a custom heuristic on a modified version of Isolation board game. An iterative deepening search technique is used in both the baseline agent and the custom agent. The only difference was the heuristic being used.

Baseline Heuristic

The baseline heuristic used is the same as the one used in lecture.

$$H_{baseline} = \text{Number of agent's moves} - \text{Number of opponent's moves}$$

Custom Heuristic

It is a combination of three types of heuristics. Center, boundary and baseline heuristics.

Center Heuristic

Center heuristic measures how far the player's location is from the center. The closer the player is, the larger the evaluation value. For this reason, I have subtracted the opponent's distance from the agent's distance.

$$H_{center} = \text{agent's distance from the center} - \text{opponent's distance from the center}$$

Boundary Heuristic

Boundary heuristic measures how far the player's location is from the 4 walls and takes the minimum one.

$$B(p)$$

$$= \min \left(\begin{array}{l} \text{distance of } p \text{ from left wall, distance of } p \text{ from right wall} \\ \text{distance of } p \text{ from top wall, distance of } p \text{ from bottom wall} \end{array} \right), \text{ where } p \text{ is the player name}$$

$$H_{boundary} = B(\text{agent}) - B(\text{opponent})$$

Combined Heuristic

This heuristic used in the custom agent is a linear combination of the above heuristics. The weights added in the below equation is a result of trial and error.

$$H_{custom} = H_{baseline} + 1.5H_{center} + 2.25H_{boundary}$$

Results

Both custom and baseline agents played against Random, Greedy, Minimax and Self 200 games each (fair matches).

The following commands were used to get the results below:

```
python run_match.py -f -o RANDOM -r 50 -p 10
```

```
python run_match.py -f -o GREEDY -r 50 -p 10
```

```
python run_match.py -f -o MINIMAX -r 50 -p 10
```

```
python run_match.py -f -o SELF -r 50 -p 10
```

Opponent	Baseline (%win)	Custom (%win)
Random	97.5	97
Greedy	78.0	98.5
Minimax	76.5	81.5
Self	47.5	50.5

As we can see in the above chart, the custom agent outperformed the baseline agent when it played against Greedy and Minimax agent. The win ratio is almost the same against the Random agent.

Answers to Questions

What features of the game does your heuristic incorporate, and why do you think those features matter in evaluating states during search?

It focuses on giving high values for having the agent at the center of the board and getting away from the wall boundaries. The features do matter in evaluating states because as long as we stay close to center, you have more choices to move and lower probability to get stuck. Similarly, if you get closer to the boundaries, you will have fewer moves and the opponent may have higher chance to win. Moreover, the heuristic gives a higher value if the opponent stays from the center and gets closer to boundaries so he will have a higher probability to lose.

Does search speed matter more or less than accuracy to the performance of your heuristic?

Agent	Average maximum depth in a game calculated over 40 games
Custom	1115.2
Baseline	1904.2

Average maximum depth in a game was calculated for both agents and, as shown, above the custom agent has a fewer average maximum depth. This means that in custom agent, the heuristics focus more on the accuracy rather speed while in baseline agent prunes more parts of the tree and hence reaching a higher depth.