

Heuristic analysis for the knights-isolation game

Abderrahim Kitouni

June 28, 2017

I propose three evaluation functions in addition to the one proposed in the assignment, which is defined by

$$improved_score = \begin{cases} +\infty, & \text{if player won} \\ -\infty, & \text{if player lost} \\ \#player_moves - \#opponent's_moves, & \text{otherwise} \end{cases}$$

First, *custom_score_3* is defined as follows

$$custom_score_3 = \begin{cases} +\infty, & \text{if player won} \\ -2 \times board_size + turn_number, & \text{if player lost} \\ \times \#player_moves - \#opponent's_moves, & \text{otherwise} \end{cases}$$

The difference with *improved_score* is that it gives different scores for different losing positions. When the agent determines that it's going to lose if the other player plays optimally, instead of forfeiting (or selecting a random move), it tries to make the game last as much as possible in the hope that the other player makes a mistake. This works well against weaker agents, but not so much against agents that use alpha-beta pruning as these will have searched to end-game at that point.

The second, *custom_score_2*, is defined as follows

$$custom_score_2 = \begin{cases} +\infty, & \text{if player won} \\ -2 \times board_size + turn_number, & \text{if player lost} \\ \#player_second_moves - \#opponent's_moves, & \text{otherwise} \end{cases}$$

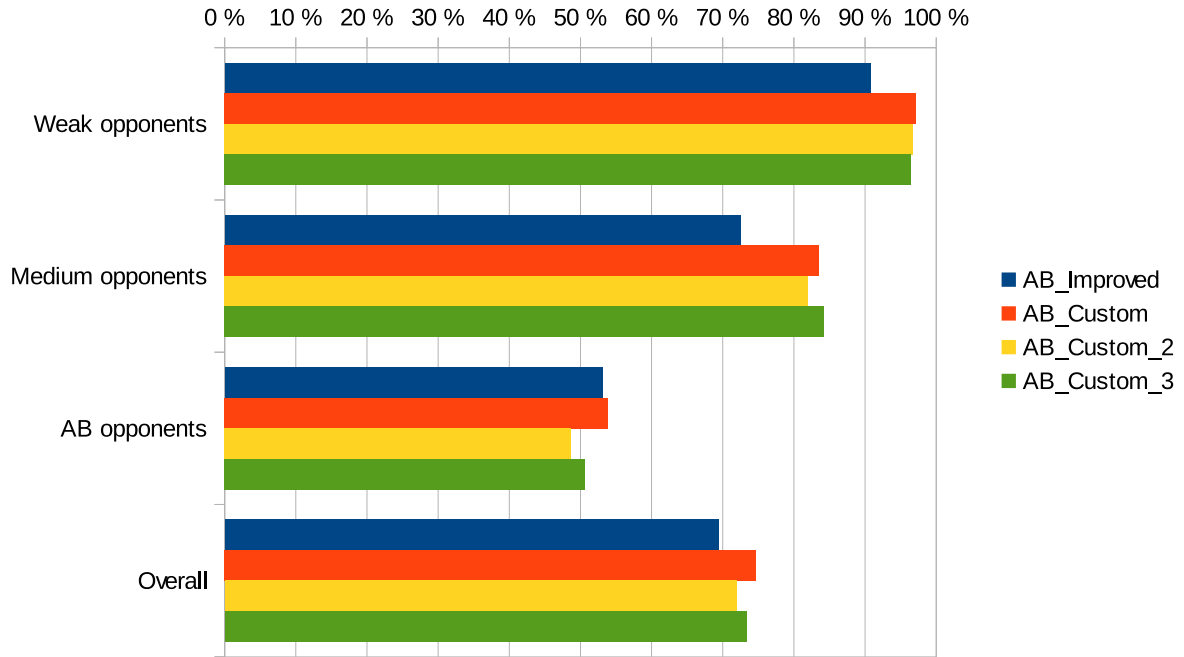
where *#player_second_moves* is the number of free squares that can be reached in two moves. The idea is to try to look a bit further on every turn.

The last proposal, *custom_score* is defined by

$$custom_score = \begin{cases} +\infty, & \text{if player won} \\ -2 \times board_size + turn_number, & \text{if player lost} \\ \#player_moves - \log(turn_number) \times \#opponent's_moves, & \text{otherwise} \end{cases}$$

The reasoning behind this is that the agent will simply look for open space at the beginning of the game, and starts being more aggressive as the game progresses.

Here the results obtained from running a tournament. The same agent using iterative deepening with alpha-beta pruning and each of the above heuristics is run against seven opponents. Each pair played 20 “fair” matches. That is, from a random starting positions each agent plays as first player then as second player. To make sure there isn't much variance, the tournament was run ten times, resulting in 400 playouts in total.



In order not to clutter the chart, I divided the opponents into three categories: weak opponents, which are the random player and the limited-depth minimax player with the distance-from-centre heuristic; medium opponents which are the limited-depth minimax players with better heuristics; and finally AB opponents which use iterative-deepening with alpha-beta pruning.

The chart shows that there isn't much difference between the different heuristics when playing against agents that search deep in the game tree. However, there is a clear advantage against weaker agents for using a heuristic score that depends on the length of the game.

The heuristic that I recommend is *custom_score* defined above because

- it is clearly better than *improved_score*, especially against weaker agents;
- it is better than the other heuristics, even if marginally, against agents using alpha-beta pruning;
- it is very simple, in contrast to *custom_score_2*, which is important to go as deep in the game tree as possible (in fact, it outperforms *custom_score_2*).