Baseline heuristic – number of own moves – number of opponent moves

Win rate against Minimax player **82.5%,** 120 matches (fair matches enabled)

Custom heuristics:

Win rate against Minimax player **80%**, 120 matches (fair matches enabled)

```
if state.ply_count < 30:
      # chase the opponent for the first 30 moves
      return own_moves_minus_opp_moves - player_distance
    elif state.ply_count < 45:
      # move away from the opponent and the center (presumably using
up corner space for moves) up to 45 moves
      return player_distance + own_moves_minus_opp_moves +
self.distance_to_center(self.get_coordinates(own_loc))
    else:
      # endgame get close to the opponent and the center
      return 0 - player_distance + own_moves_minus_opp_moves -
self.distance_to_center(self.get_coordinates(own_loc))
```

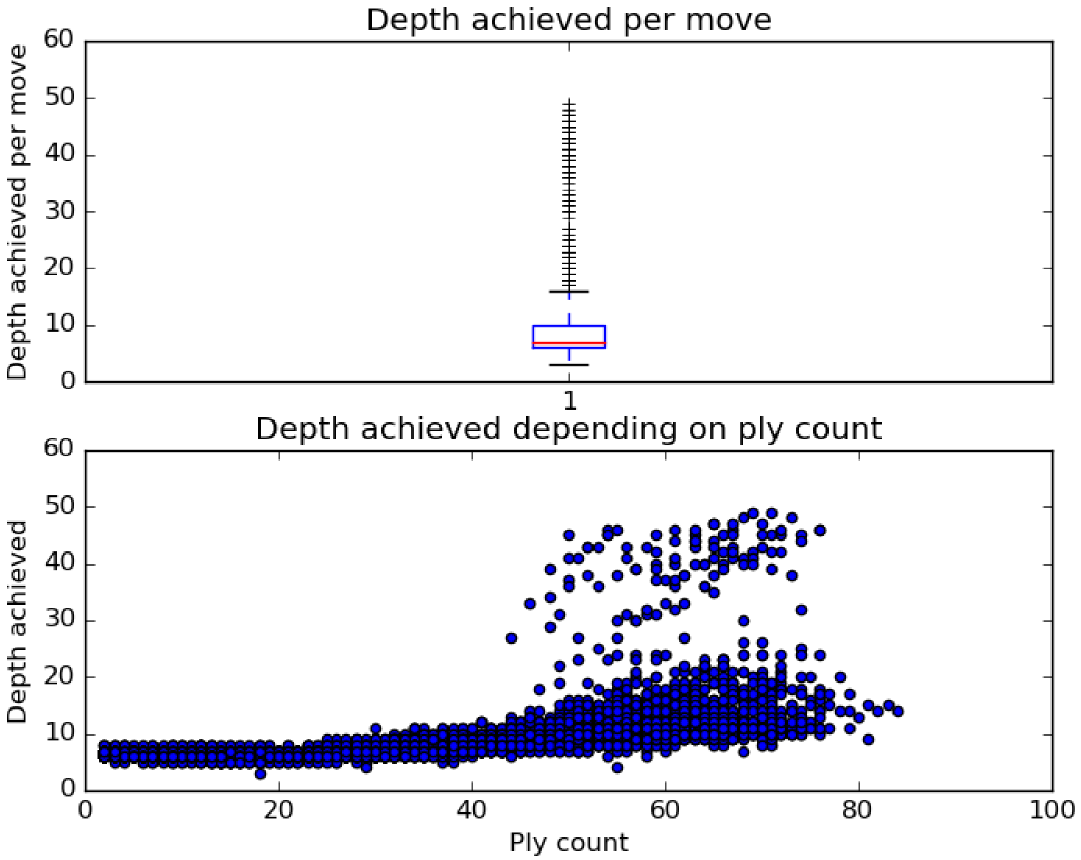
Custom heuristics 2:

Win rate against Minimax player **83.5%**, 120 matches (fair matches enabled)

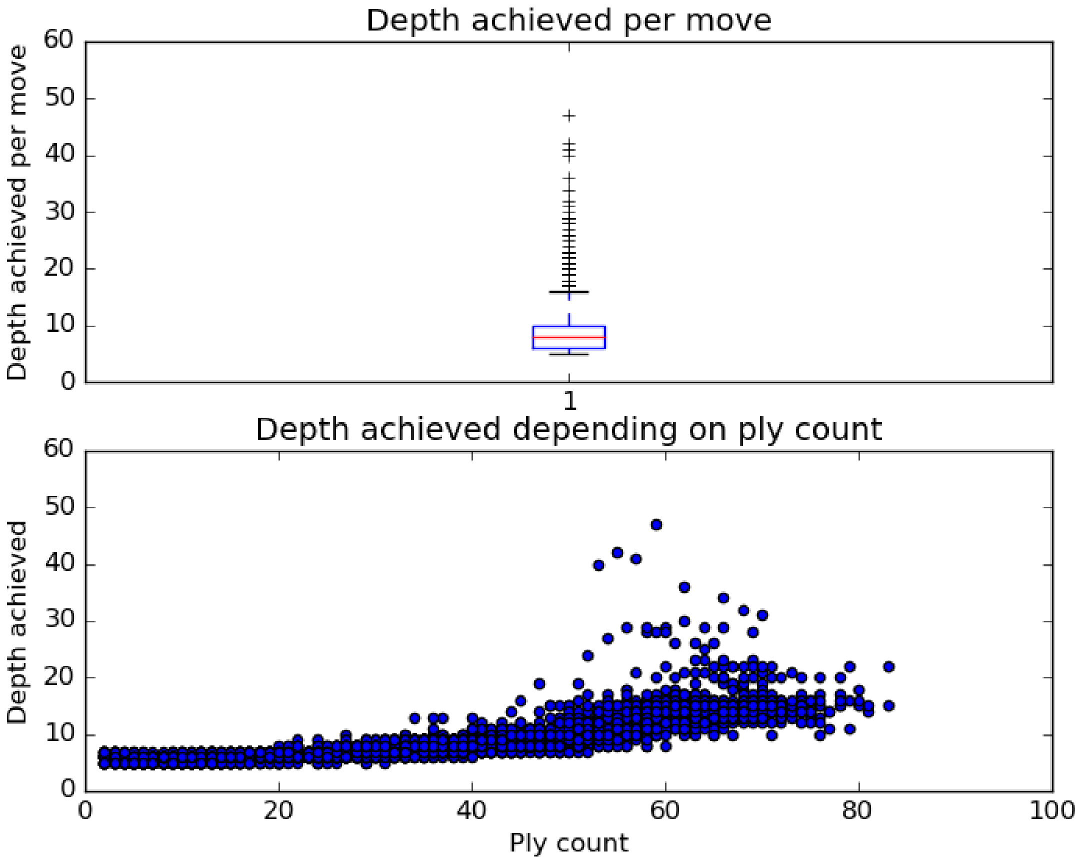
```
if state.ply_count < 30:
      # chase the opponent for the first 30 moves
      return own_moves_minus_opp_moves - player_distance
    elif state.ply_count < 50:
      # stay close to the opponent and the center up to 50 moves
      return 0 - player_distance + own_moves_minus_opp_moves +
self.distance_to_center(self.get_coordinates(own_loc))
    else:
      # endgame - stay away from the oponent but still try to stay
close to center
      # increace the effect of own_moves_minus_opp_moves value times 2
since
      # we'd like to keep its influence a bit higher in the endgame
      return player_distance + (own_moves_minus_opp_moves * 2) -
self.distance_to_center(self.get_coordinates(own_loc))
```

Data Analysis:

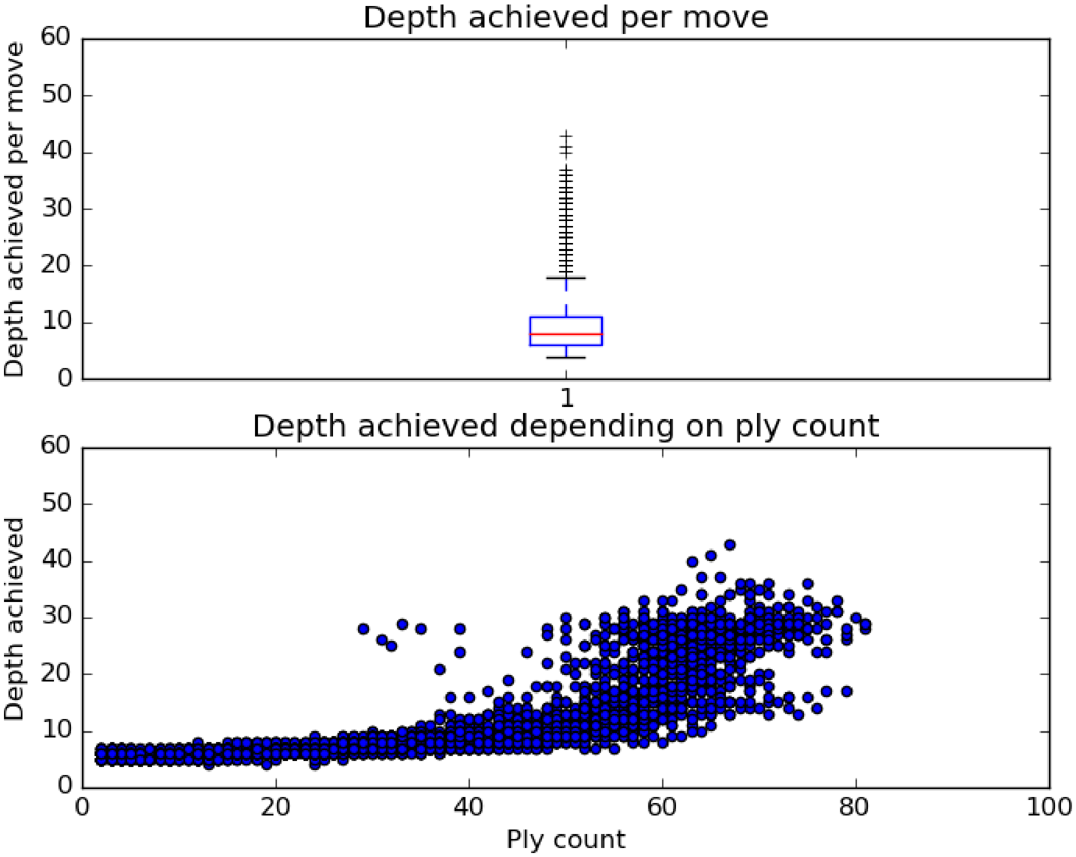
Baseline



Custom heuristics



Custom heuristics 2



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

  Both custom_heuristics and custom_heuristics_2 incorporate the distance between agents and the distance of my agent to the center of the game board. They use these slightly differently (as explained in the code and comments above), and always along with the baseline heuristics which seems pretty useful anyway. Both custom heuristics chase the opponent for the first 30 moves, but I don't think any smart heuristics can lose the game within 30 moves, so it's just a strategy in case the opponent isn't playing smart. Custom heuristics between moves 30-45 tries to stay away from both the opponent and the center, presumably filling up corner space with no risk of losing (because it stays away from the opponent), so in the endgame it gets close to the opponent and the center in an attempt to win. Custom heuristics 2 on the other hand, uses a different midgame strategy (moves 30-50 in this case) so that it stays close to the opponent and the center, while in the endgame tries to hand around the center, yet away from the opponent, with additional emphasis on the baseline heuristics (times 2).

- Analyze the search depth your agent achieves using your custom heuristic. Does search speed matter more or less than accuracy to the performance of your heuristic?

  The average search depth doesn't differ that much between all tested heuristics; median is around or below 10, with outliers above 20 and up to around 50. The scatter plot analyzing depth achieved depending on ply count, for the baseline shows a variation in the depth achieved above 45 plies, though most of the time it stays consistent with a steady increase from around 10 to 20 depth from 45 plies onwards. Custom heuristics displays a lower depth reached overall, with much less variation and outliers after 50 plies. Custom heuristics 2 on the other hand, shows a consistently higher depth reached from ply count 45-50 onwards, and it has outliers in the 30 depth range from ply count 30 onwards (when it starts using a different calculation). So, in case of custom heuristics 2 I think it can "see" the win in some cases from ply count 30-35 which might be the reason it showed slightly better win results compared to the baseline heuristics. As for what matters more (search speed or accuracy), I'd say in this case in both custom heuristics it's about the same. In custom heuristics 2 speed might matter a bit more, since it achieves consistently higher depths near the endgame.