# Heuristic Introductions

In this analysis I will look at four types of heuristics: heuristics to minimize distance from the center of the board, heuristics to maximize the potential move difference between the player and opponent, opening move heuristics, and combination heuristics. In the distance heuristic category I'll look at using the euclidean distance, Manhattan distance, and Chebyshev distance. In the move difference category I'll be looking at the performance of a "normal" move difference heuristic and an "aggressive" move difference heuristic which chases the opponent. I'll also look at an opening move heuristic of for the first N moves instead of running the minimax or alphabeta pruning algorithms just making moves that limit the distance from center. Finally, I'll also be looking at the combinations of all the aforementioned heuristics.

# Motivations, Data, and Analysis of Heuristics

## *Distance Heuristics*

The motivation behind investigating the performance of minimizing the distance from center comes from the intuition that knight pieces have the highest freedom of movement for their subsequent moves if they start from center-most square (non-repeating). Therefore the negation of the distance calculations is used for scoring. Figures 1 and 2 illustrate this for a 9x9 board for two different starting positions with a depth of 2.

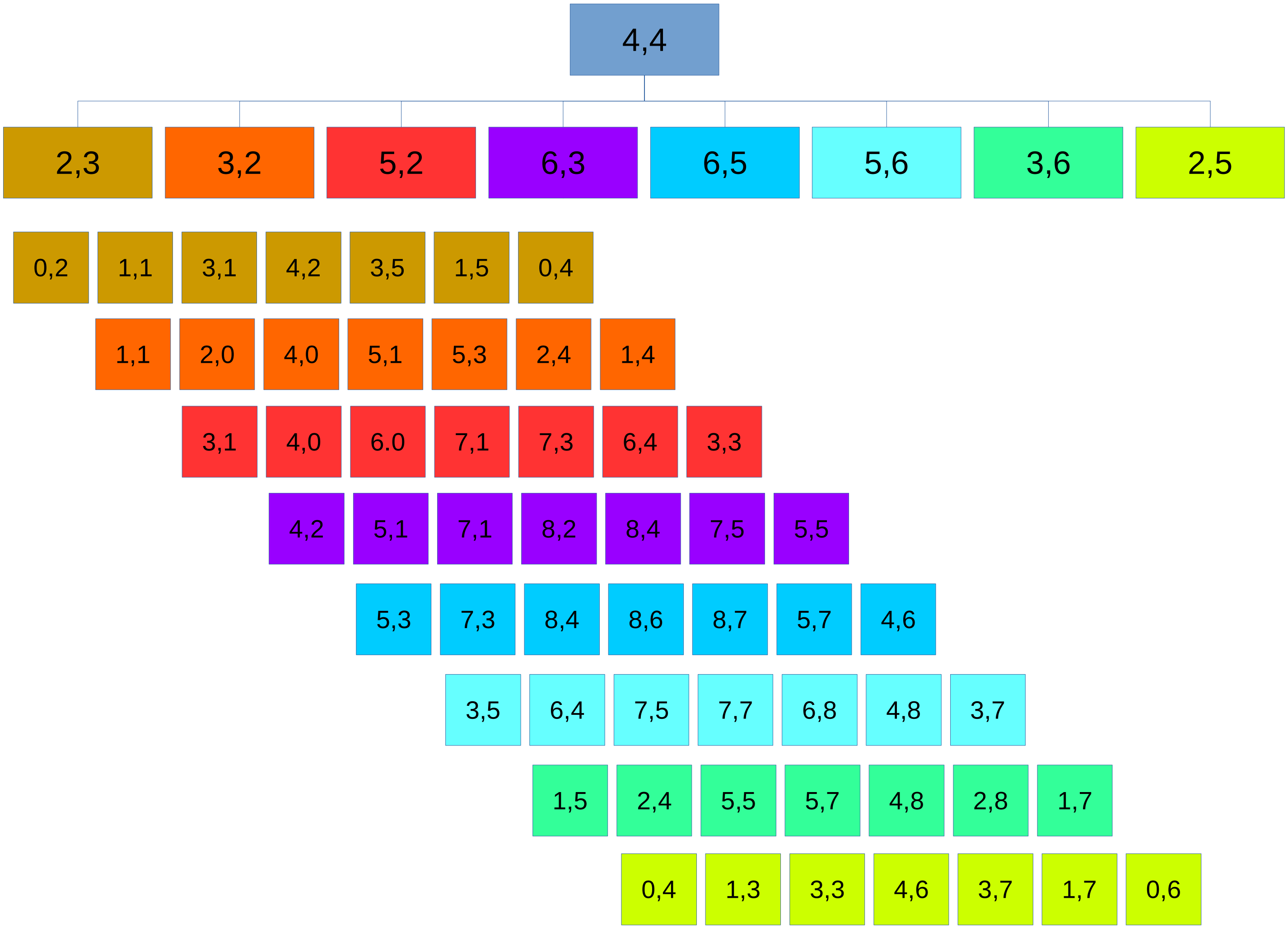

*Figure 1: Move Tree Starting from (4,4) Position*

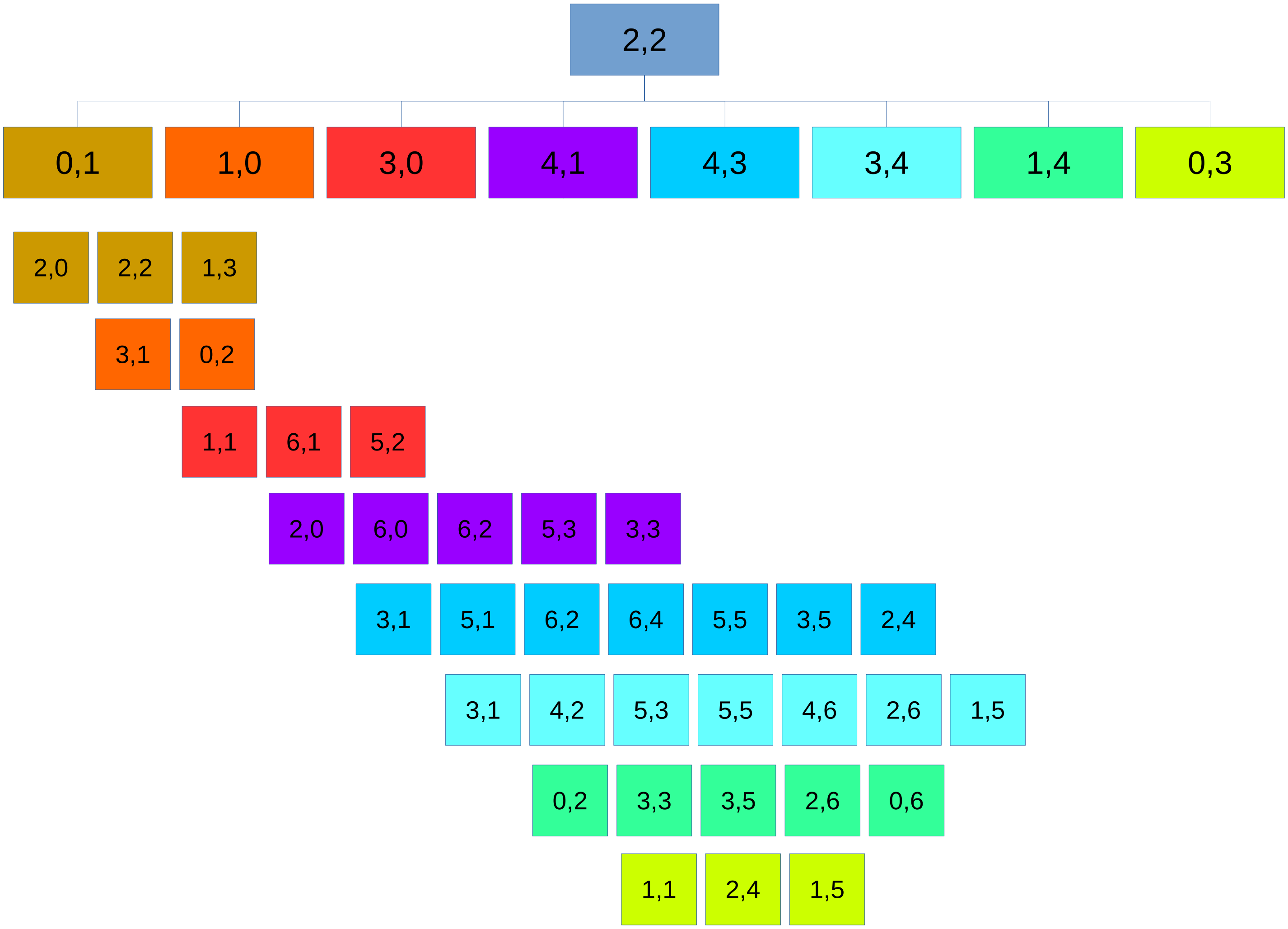

*Figure 2: Move Tree Starting From (2,2) Position*

Generally speaking, once you move to within two cells of an edge or corner of a grid your freedom of movement is limited. The closer you are to the center the more moves you can make before being within two of an edge or corner. Three common distance calculations were evaluated. Euclidean distance calculates a straight line between 2 points. Manhattan Distance calculates the distance between two points in the sum as the absolute difference of the coordinates. Chebyshev distance calculates the difference assuming all adjacent cells have a distance of 1. In Manhattan distance a diagonally adjacent cell would have a distance of two but in Chebyshev distance it would have a distance of only 1.

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

*Figure 3: Euclidean Distance Formula*

$$|x_1 - x_2| + |y_1 - y_2|$$

*Figure 4: Manhattan Distance Formula*

$$\max(|x_1 - x_2|\,, |y_1 - y_2|)$$

*Figure 5: Chebyshev Distance Formula*

*Results:*

*Table 1: AB_Improved Agent vs Distance Heuristics Wins (out of rounds of 10)*

| Match # | Opponent | AB_Improved Wins | -Euclidean Wins | -Manhattan Wins | -Chebyshev Wins |
|---|---|---|---|---|---|
| 1 | Random | 10 | 7 | 10 | 8 |
| 2 | MM_Open | 6 | 6 | 7 | 7 |
| 3 | MM_Center | 9 | 9 | 9 | 9 |
| 4 | MM_Improved | 7 | 8 | 9 | 6 |
| 5 | AB_Open | 5 | 4 | 5 | 4 |
| 6 | AB_Center | 5 | 5 | 6 | 6 |
| 7 | AB_Improved | 6 | 5 | 2 | 5 |
|  | % Win Rate | 68.6 | 62.9 | 68.6 | 64.3 |

*Analysis:*

Manhattan distance performed the best and equaled the AB_Improved agent’s win %. This is not surprising since it best aligns with what cells provide the most freedom of movement.

## *Move Difference Heuristics*

Looking at the #player_moves-#opponent_moves was motivated by the games win/lose condition (first player out of moves loses). I’ll refer to this evaluation heuristic as “MoveDiff”. I also looked at an aggressive version of this MoveDiff that I’ll call AgressiveMoveDiff. AgressiveMoveDiff’s equation is #player_moves-2*#opponent_moves. This causes the player to effectively chase the opponent.

*Results:*

*Table 2: AB_Improved Agent vs Move Difference Heuristic Wins (out of rounds of 10)*

| Match # | Opponent | AB_Improved Wins | MoveDiff Wins | AggressiveMoveDiff |
|---|---|---|---|---|
| 1 | Random | 10 | 10 | 9 |
| 2 | MM_Open | 7 | 7 | 8 |
| 3 | MM_Center | 7 | 9 | 10 |
| 4 | MM_Improved | 7 | 5 | 7 |

| | | | | |
|---|---|---|---|---|
| 5 | AB_Open | 4 | 4 | 6 |
| 6 | AB_Center | 5 | 7 | 5 |
| 7 | AB_Improved | 6 | 6 | 5 |
| | % Win Rate | 65.7 | 68.6 | 71.4 |

*Analysis:*
AgressiveMoveDiff performed the best and had a win percentage 5.7% better than AB_Improved. AggressiveMoveDiff's heuristic works better than MoveDiff because it can increase the likelihood of boxing the opponent into a corner through chasing the opponent.

## *Opening Moves Heuristics*

Different heuristics may prove more useful at different stages of a game. I looked at an opening move heuristic that tries keep the player close to the middle of the board for reasons outlined in the Distance Heuristics section above. The goal was to find the optimal # of moves to use the opening move heuristic before switching back to the evaluation heuristics (and minimax/alphabeta prunning). I'll be using Manhattan distance to calculate the distance from center based on the results from the Distance Heuristics section.

*Results:*

*Table 3: Opening Moves Heuristics = 1*

| Match # | Opponent | AB_Improved Wins | MoveDiff Wins | AggressiveMoveDiff |
|---|---|---|---|---|
| 1 | Random | 10 | 10 | 9 |
| 2 | MM_Open | 9 | 7 | 8 |
| 3 | MM_Center | 8 | 9 | 10 |
| 4 | MM_Improved | 7 | 9 | 7 |
| 5 | AB_Open | 7 | 5 | 6 |
| 6 | AB_Center | 7 | 6 | 5 |
| 7 | AB_Improved | 5 | 3 | 5 |
| | % Win Rate | 75.7 | 70.0 | 71.4 |

*Table 4: Opening Moves Heuristics = 5*

| Match # | Opponent | AB_Improved Wins | MoveDiff Wins | AggressiveMoveDiff |
|---|---|---|---|---|
| 1 | Random | 9 | 10 | 10 |
| 2 | MM_Open | 9 | 7 | 8 |
| 3 | MM_Center | 9 | 9 | 8 |
| 4 | MM_Improved | 8 | 5 | 7 |
| 5 | AB_Open | 5 | 5 | 6 |

| | | | | |
|---|---|---|---|---|
| 6 | AB_Center | 7 | 6 | 6 |
| 7 | AB_Improved | 4 | 5 | 4 |
| | % Win Rate | 72.9 | 67.1 | 70.0 |

*Table 5: Opening Moves Heuristics = 10*

| Match # | Opponent | AB_Improved Wins | MoveDiff Wins | AggressiveMoveDiff |
|---|---|---|---|---|
| 1 | Random | 9 | 9 | 10 |
| 2 | MM_Open | 7 | 6 | 6 |
| 3 | MM_Center | 7 | 8 | 7 |
| 4 | MM_Improved | 10 | 5 | 9 |
| 5 | AB_Open | 5 | 5 | 5 |
| 6 | AB_Center | 8 | 6 | 5 |
| 7 | AB_Improved | 5 | 5 | 3 |
| | % Win Rate | 72.9 | 62.9 | 64.3 |

*Table 6: Opening Moves Heuristics = 15*

| Match # | Opponent | AB_Improved Wins | MoveDiff Wins | AggressiveMoveDiff |
|---|---|---|---|---|
| 1 | Random | 10 | 8 | 9 |
| 2 | MM_Open | 7 | 8 | 7 |
| 3 | MM_Center | 6 | 8 | 8 |
| 4 | MM_Improved | 6 | 7 | 7 |
| 5 | AB_Open | 5 | 7 | 4 |
| 6 | AB_Center | 5 | 6 | 6 |
| 7 | AB_Improved | 4 | 6 | 5 |
| | % Win Rate | 61.4 | 71.4 | 65.7 |

*Table 7: Opening Moves Heuristics = 20*

| Match # | Opponent | AB_Improved Wins | MoveDiff Wins | AggressiveMoveDiff |
|---|---|---|---|---|
| 1 | Random | 9 | 10 | 8 |
| 2 | MM_Open | 6 | 7 | 7 |
| 3 | MM_Center | 6 | 5 | 4 |
| 4 | MM_Improved | 8 | 6 | 6 |

| | | | | |
|---|---|---|---|---|
| 5 | AB_Open | 5 | 8 | 3 |
| 6 | AB_Center | 4 | 8 | 7 |
| 7 | AB_Improved | 4 | 4 | 6 |
| | % Win Rate | 60.0 | 68.6 | 58.6 |

*Analysis:*

Based on the above analysis the optimal number of moves to run the opening moves near the center heuristic near 15. This is rather longer than I initially expected. However, the technique of staying near the center would be valid until the center positions get too full. Based on the game parameters from the tournament.py program and the competing agents heuristics, this appears to not happen until after 15 player moves. Another interesting result is that MoveDiff performs better than AggressiveMoveDiff when we use the opening moves heuristic > 10 moves. This points to a “rough handover” between the two different heuristics where the player, while potentially far from the opponent, moves towards the opponent and get stuck. Applying opening 15 moves heuristic results in a 10% improvement over AB_Improved agent which means it’s relative performance is greater than MoveDiff or AgressiveMoveDiff alone.

## *Combination Heuristics*

Will combining the best results from the above heuristics sections provide an overall better performing player? I’ve seen in the previous section combining the Move Difference heuristics with a opening moves heuristic that this isn’t always the case. In this section I’ll look at the performance of using a -Manhattan Distance+MoveDiff and a -Manhattan Distance+AgressiveMoveDiff evaluation heuristic. Since one result of the opening move heuristic analysis is that some evaluation heuristics do better than others with the stay in center heuristic, I’ll look at these heuristics without the opening move heuristic and with a 15 move opening heuristic.

*Results:*

*Table 8: Combination Heuristics Without Opening Moves Heuristics*

| Match # | Opponent | AB_Improved Wins | -Manhattan+ MoveDiff Wins | -Manhattan+ AggressiveMoveDiff |
|---|---|---|---|---|
| 1 | Random | 9 | 9 | 10 |
| 2 | MM_Open | 8 | 7 | 8 |
| 3 | MM_Center | 8 | 9 | 10 |
| 4 | MM_Improved | 6 | 7 | 5 |
| 5 | AB_Open | 4 | 7 | 6 |
| 6 | AB_Center | 7 | 4 | 6 |
| 7 | AB_Improved | 5 | 6 | 4 |
| | % Win Rate | 67.1 | 70.0 | 70.0 |

*Table 9: Combination Heuristics with 15 Opening Moves Heuristics*

| Match # | Opponent | AB_Improved Wins | -Manhattan+ MoveDiff Wins | -Manhattan+ AggressiveMoveDiff |
|---|---|---|---|---|
| 1 | Random | 8 | 10 | 8 |
| 2 | MM_Open | 7 | 6 | 6 |
| 3 | MM_Center | 10 | 9 | 8 |
| 4 | MM_Improved | 6 | 4 | 9 |
| 5 | AB_Open | 5 | 6 | 6 |
| 6 | AB_Center | 6 | 7 | 5 |
| 7 | AB_Improved | 4 | 4 | 6 |
|  | % Win Rate | 65.7 | 65.7 | 68.6 |

*Analysis:*

Combining these techniques didn’t result in any improvement over previously discussed heuristics.

## *Recommendation:*

The best performing heuristic is to use the opening move heuristic minimizing the Manhattan distance from center for the first 15 moves and then switching to the MoveDiff evaluation heuristic which maximizes the number of moves the player has over the opponent. Manhattan distance was the best performing heuristic according to the results shown in the Distance Heuristic section and hence used in the opening move heuristic. The opening move heuristic was determined to be beneficial when used for a significant number of moves (around 15 or greater). It’s relative % win rate performance against AB_Improved was 10% higher. Also, it’s results were more consistent and it’s much less complex than running iterative deepening with an evaluation heuristic in the beginning of the game. Finally, MoveDiff outperforms AggressiveMoveDiff when (and only when) the opening move heuristic is used for a significant number of opening moves. This heuristic combination results in a full 10% win percentage improvement over AB_Improved agent based on my tournament.py run.